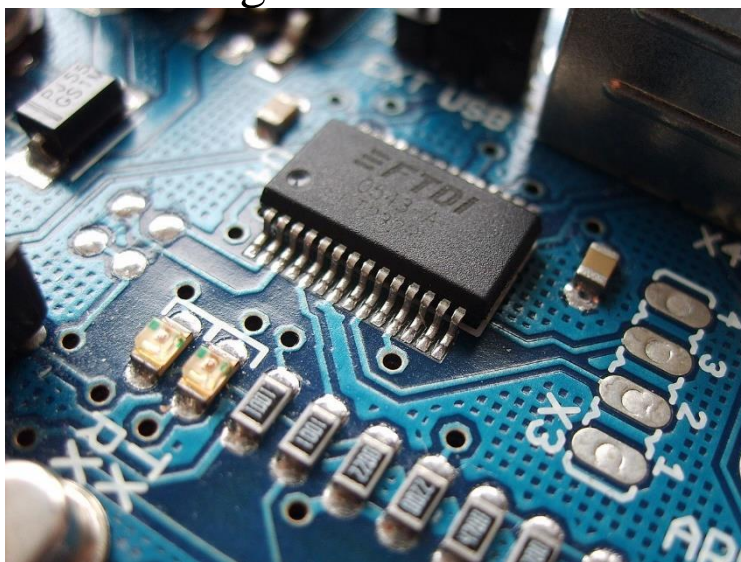




Mestrado em Engenharia Biomédica
Eletrónica Médica

2023/2024

Laboratórios Integrados de Eletrónica Médica I



Grupo VI

Luís Gonçalves
PG54011

Sandra Ferreira
PG54221

Docentes

Cristina Santos

Cristiana Pinheiro

Resumo

Este projeto retrata o desenvolvimento de um sistema capaz de dar feedback sobre os movimentos de pacientes com dificuldades motoras.

Utilizou-se diferentes sensores inercias e de EMG para captar dados que foram utilizados para criar modelos de Machine Learning capazes de indicar o tipo de movimento, bem como a definição de thresholds de atividade muscular, usado para identificar um bom ou mau movimento.

Índice

1. Introdução.....	4
2. Métodos.....	4
2.1 Obtenção de Dados	5
2.2 Criação de modelos de avaliação	7
2.2.1 Preparação de Dados.....	7
2.2.2 Criação de modelo.....	8
2.3 Programação do Microcontrolador	9
3 Resultados e Discussão.....	12
3.1 Dados recolhidos	12
3.2 Análise dos Modelos	15
3.3 Teste do Código.....	16
4. Conclusão.....	17
5. Bibliografia.....	17

1. Introdução

Em Portugal cerca de 125 a 150 pessoas por cada 100 000 anualmente têm dificuldades motoras causadas por AVC's, comprometendo a qualidade de vida. Uma forma de melhorar esse fator é através de sistemas de biofeedback, que são de extrema importância para a reabilitação motora, uma vez que os mesmos recebem informações de diversas formas sobre os seus comportamentos.

Durante este projeto foi proposto a realização de um sistema capaz de dar feedback ao utilizador dos seus diferentes movimentos, através da utilização de sensores inerciais e de eletromiograma (EMG), bem como o uso de Machine Learning para a identificação do tipo de movimentos a serem feitos pelo utilizador. Para além destas ferramentas recorreu-se ainda a um microcontrolador capaz de ativar diversos atuadores, quando corretamente feito um tipo de movimento.

2. Métodos

Este projeto foi dividido em três fases criando um sistema completo capaz de identificar o tipo de movimento e se foi feito devidamente ou não.

Para isso foi realizado um diagrama que identifica os passos a serem tomados ao longo deste projeto. Na Figura 1 é possível observar essa representação

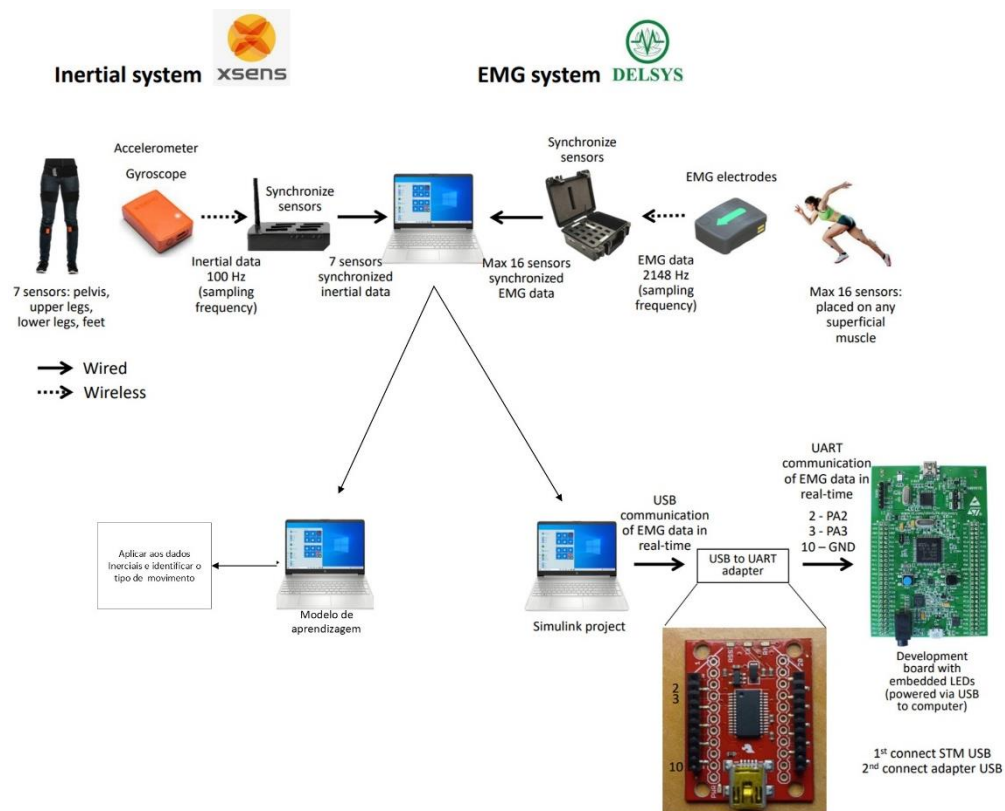


FIGURA 1 – DIAGRAMA PROPOSTO PARA O PROJETO.

2.1 Obtenção de Dados

A fim, de implementar o projeto recolheu-se dois tipos de dados, dados biomecânicos, ou dados inerciais, e dados fisiológicos, ou dados relativos ao EMG. Antes da colocação dos sensores é importante a preparação do participante, onde o sistema do EMG requer um maior cuidado, uma vez que este mede através do contacto com a pele, que é um meio propenso a ruído. Assim deve ser feita a limpeza da na zona a ser colocado, podendo ainda ser retirados os pelos da mesma, bem como a hidratação, que permite a diminuição do ruído.

Foram então usados sensores Inerciais da *XSENS*, em 7 posições diferentes, na zona dos pés (em ambos), na zona abaixo dos joelhos, na zona externa da coxa e na zona do pélvis nas costas. Na figura 2(a) e é possível ver marcada com círculos laranjas, as zonas onde estes foram colocados.

Foram também utilizados os sensores da atividade elétrica muscular da Delsys e foram colocados seguindo o guia criado pela Delsys para a medição do músculo femorotibial, ou *tiabialisanterior*, para cada perna.

Então seguindo este guia foram colocados no centro deste musculo, paralelamente às fibras musculares. Assim foram colocados a um terço deste musculo em relação ao joelho do paciente. Na figura 2(b) é possível observar a posição destes sensores marcados com círculos azuis.



FIGURA 2(A) E 2(B) – COLOCAÇÃO DOS SENSORES DA XSENS E DELSYS EM ÂNGULOS DIFERENTES.
(A) VISÃO FRONTAL. (B) VISÃO DORSAL.

É necessário ainda a calibração do sistema da Xsens a cada utilização, uma vez que cada um dos pacientes terá um modelo biomecânico diferente. Começa-se com o participante tendo os seus pés paralelos e apontando para a frente, tendo ambos os braços

e pernas esticados e junto ao corpo. Após isto através da aplicação fornecida pela Xsens é começada a calibração, onde o deve ser mantida a posição anterior por alguns segundos, e de seguida iniciar a marcha de maneira normal para frente e para trás.

Foi necessário ainda a medição de algumas dimensões sobre o corpo e posição dos sensores do participante, para que a sessão de recolha de dados pela aplicação da Xsens seja corrida e seja possível retirar desta uma pequena gravação em 3 dimensões dos movimentos do participante através das informações recolhidas pelos sensores, bem como estas informações fisiológicas referidas.

Entre cada um dos sistemas teve-se de realizar a ligação a uma base recetora de dados, através de ligações sem fios. Estas bases receberão dados com uma frequência máxima de 100 Hz e de 2148 Hz para os sistemas da XSENS e da Delsys, respetivamente. Este protocolo de ligação sem fios garante que haja uma sincronia entre sensores do mesmo sistema.

Por fim antes da recolha de dados é necessário ainda garantir a sincronia dos dados entre estes dois sistemas de recolha. Existem diferentes formas de realizar essa sincronia, como por exemplo, através de um tratamento dos dados *a posteriori*. Neste exemplo pedir-se-ia ao participante para realizar um movimento fora do procedimento, que marcaria em ambos os sistemas o momento inicial de captura.

Nesta atividade realizou-se a sincronização por hardware, desta maneira utilizou-se um equipamento que liga ambos os sistemas, e através das aplicações de recolha de dados de ambas a Delsys e Xsens, permitir que o sistema de EMG esteja a recolher dados, mas que apenas os envie no momento que exista um pulso elétrico proveniente do sistema de sensores da XSENSE. Fazendo assim com que ambos enviem dados simultaneamente, garantindo a sincronia entre sistemas.

Na preparação para aquisição de dados propôs-se ainda um protocolo a ser seguido pelo participante. Este protocolo começa por ser medido as Máximas voluntárias contrações relativas ao EMG, isto é a capacidade máxima do musculo, para que este seja comparado com os dados relativos aos movimentos. A importância deste passo é elevada uma vez que existe uma alta variabilidade entre medições, tanto interpessoais, como intrapessoais, e assim é possível fazer a normalização da atividade muscular dos participantes, e esta sim pode ser comparada entre participantes e com outros dados provenientes do mesmo.

Avançou-se então para os movimentos a serem realizados pelos participantes. Definiu-se então dois movimentos diferentes. Um deles começou-se com o participante parado e pediu-se que se ande para a frente e quando não pudesse continuar, depois virar-se e andar para trás, isto até completar 3 voltas.

O outro iniciou-se com o participante de pé, junto de uma cadeira e esperar 5 segundos e senta-se, voltar a esperar mais 5 segundos, para voltar a se levantar. Repete-se este conjunto de instruções até serem feitas 5 repetições.

Procedeu-se ainda a um pós-processamento dos dados de duas maneiras. Primeiro procedeu-se ao cálculo do Root Mean Square ou valor eficaz, uma vez que para estes dados o sinal deste é insignificativo, este simplifica o projeto no futuro fazendo uma

espécie de filtragem nos dados. Foi ainda realizado o cálculo da taxa de atividade muscular através dos valores Máximas voluntarias contrações.

2.2 Criação de modelos de avaliação

Para poder haver a reconhecimento de diferentes tipos de movimentos utilizou-se diferentes modelos de *machine learning*, através do programa *Weka*, testando qual o mais efetivo para este tipo de dados. Para isso utilizou-se um dataset [1], de 4 diferentes participantes, onde são apresentados diferentes atributos, tais como, género, idade, altura, peso, IMC e dados inerciais como os obtidos no ponto 2.

Para a criação deste tipo de modelos é necessário um grande número de dados para que sejam corretamente instruídos. É necessário também uma grande variabilidade nos dados, devendo ser balanceados a quantidade de dados nos atributos tais como género, idade, peso, altura e IMC, para que o modelo não seja mais influenciado para identificar com base apenas nos dados das pessoas com estas características, mas sim saiba identificar com base nestes diferentes dados.

2.2.1 Preparação de Dados

Após a revisão dos dados deste dataset, fez-se então a preparação dos dados do dataset. Na Figura 3(a) é possível observar a distribuição por classes dos dados desta base de dados.

Começou-se por verificar a existência de *Missing Values* e removê-los caso eles existam, mas uma vez que não existiam nesta base de dados não foi necessária a realização deste passo.

Avançou-se para a identificação e remoção de *Outliers*, para este passo utilizou-se o filtro *InterquartileRange*, que identifica quais das entradas podem ser consideradas *outliers*, para esta base de dados foram identificados 45824 valores como *outliers*. Na Figura 3(b) observa-se a distribuição após este passo. Após isto removeu-se com o filtro *RemoveWithValues*, com o atributo a remover ser o *Outlier* e o índice a ser o *last*, isto é, os que são *outliers*. A Figura 3(c) mostra a distribuição após a remoção dos *outliers*.

Continuou-se com a normalização dos dados numéricos, através do filtro *Normalise*. Através deste passo, os dados numéricos do *dataset* passam a variar entre -1 e 1 como foi definido no filtro.

Após isto, identificou-se os atributos significativos, e removeu-se os que não são significativos para a criação do modelo. Assim usou-se o filtro *AttributeSelection*, que escolhe os atributos que são significativos. Ao aplicar o filtro apenas permaneceram os atributos numéricos e a classificação original das classes.

Por fim procedeu-se ao balanceamento das classes, através do *classBalancer*, fazendo com que os pesos de cada classe sejam iguais para todas as classes, mantendo a quantidade de dados. Na figura 3(d) pode-se observar a distribuição dos pesos após este passo. Este passo é de extrema importância para que o modelo a ser criado tenha um balanceamento a reconhecer as classes, fazendo com que não exista uma maior tendência a escolher uma classe a outra.

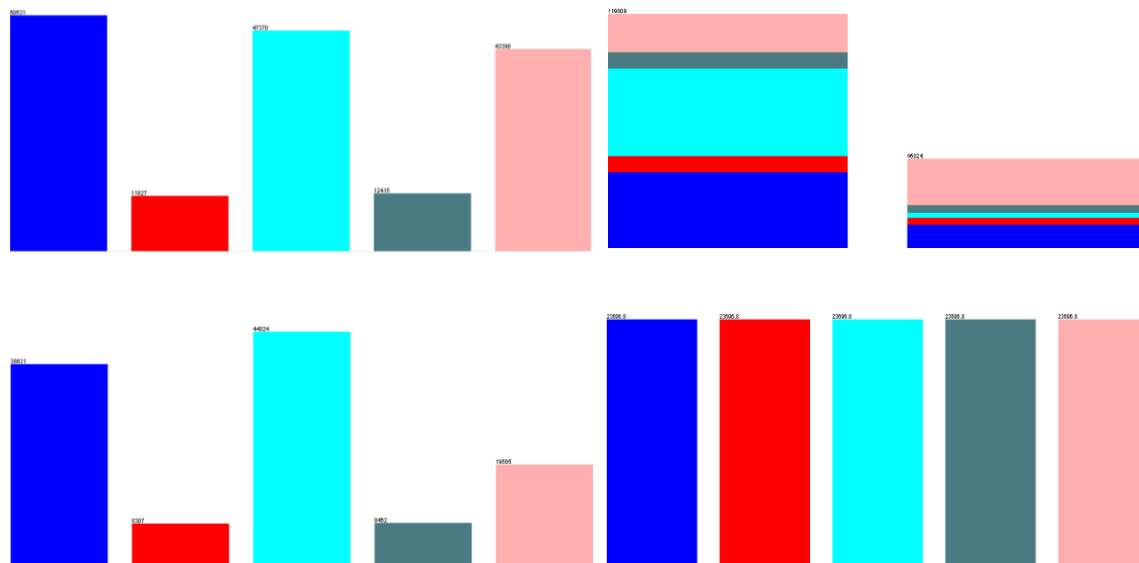


FIGURA 3 – DISTRIBUIÇÕES DOS PESOS DOS DADOS DO DATASET. (A) DISTRIBUIÇÃO DE CLASSES (CADA COR REPRESENTA UMA CLASSE DIFERENTE). (B) DISTRIBUIÇÃO DE OUTLIERS POR CLASSES (COLUNA DA ESQUERDA REPRESENTA OS DADOS QUE NÃO SÃO OUTLIERS, A DA DIREITA SÃO OUTLIERS). (C) DISTRIBUIÇÃO DE CLASSES APÓS A REMOÇÃO DE OUTLIERS. (D) DISTRIBUIÇÃO DE CLASSES APÓS O BALANCEAMENTO DE DADOS

2.2.2 Criação de modelo

Como foi referido anteriormente utilizou-se modelos de machine learning para reconhecer o tipo de movimento feito pelo participante. Para isso existem dois grandes tipos de modelos, modelos supervisionados e modelos, não supervisionados. Nos modelos supervisionados, o modelo é construído a partir de dados pré-definidos, ou dados de treinos, sendo que após isto são classificados através de outros dados, ou dados de treino. Quanto aos modelos não supervisionados os dados fornecidos não possuem os resultados a que se pretende chegar e cabe ao modelo tentar dividir o conjunto de dados em grupos.

Para a finalidade definida neste projeto usou-se os modelos supervisionados, uma vez que queremos fazer a identificação de um tipo de dados, e não a divisão e separação destes.

Assim começou-se com o modelo de *RandomForest*, um modelo que cria diversas árvores de decisão aleatórias, ou seja, um conjunto de perguntas, onde cada resposta leva a um novo nó. Este método assemelha-se a de uma árvore uma vez que cada novo nó leva a criação de um número superior de nós.

Um aspeto a ter cuidado neste e em qualquer outro método é o *Overfitting*, ou seja, o modelo estar extremamente adaptado aos dados de treino, mas de tal modo que não representa a realidade, fazendo com que quando apresentado um novo conjunto de dados

este não consiga uma precisão na determinação nestes novos dados. Assim neste método deve-se ter em consideração a quantidade de níveis presentes na árvore, uma vez que poderá levar ao *overfitting*.

Foi necessário ainda fazer a divisão dos dados em treino e teste, para isso utilizou-se e comparou-se dois métodos distintos, estes sendo o *Percentage Split* a 66% e *Cross Validation* com 10 Folds. Estas duas formas de dividir dados representam respetivamente uma simples divisão com 66% dos dados para treino e o restante para teste, e uma divisão em 10 partes em que 9 delas são usadas para treino e 1 para teste é repetido até terem sido usadas todas as partes uma vez para teste.

Foram usados ainda os métodos da Decision Table e de Multilayer Perceptron. A Tabela de Decisão tal como o nome indica e em semelhança á árvore de decisão, representa o método onde é construída uma tabela onde existem guias para as decisões que devem ser tomadas. O MultilayerPerceptron é um tipo de redes neuronais que funciona com 3 tipo de camadas uma de input, uma de output e uma camada escondida. A camada escondida submete os valores que entram nesta com input com diferentes funções de ativação e são submetidos sucessivamente a outras funções de ativação até chegarem a camada de output.

2.3 Programação do Microcontrolador

Visto que neste sistema queremos proporcionar ao utilizador um feedback do movimento que está a efetuar, utilizou-se um microcontrolador, sendo o que foi escolhido o STM32F4 Discovery development board.

A utilização deste microcontrolador levou á necessidade da utilização de um adaptador da ligação USB para UART, isto é, universal asynchronous receiver/transmitter do microcontrolador, uma vez que este microcontrolador não tem periféricos para este efeito. Teve-se em conta que é necessária a ligação através de fios alguns dos pinos de ambos, devendo ser feita a ligação do pino 2, 3 e 10 do adaptador aos pinos PA2, PA3 e ground do STM respetivamente.

Então para a realização do biofeedback, começou-se por definir um thresholds para os quais é considerado que está a ser realizada uma determinada tarefa. Para isto calculou-se através dos dados obtidos após o Root Mean Square, para cada tipo de movimento um diferente threshold, sendo este 80% da média dos valores de pico, tendo em conta possível outliers que desviem este valor.

Após isto, realizou-se o fluxograma, observável na Figura 4, do algoritmo para receber em tempo real o EMG pelo simulink e controlar os LED's através de uma máquina de estados, isto é, comparar com o threshold definido e liga ou desliga o led quando ultrapassa ou é inferior a este valor respetivamente.

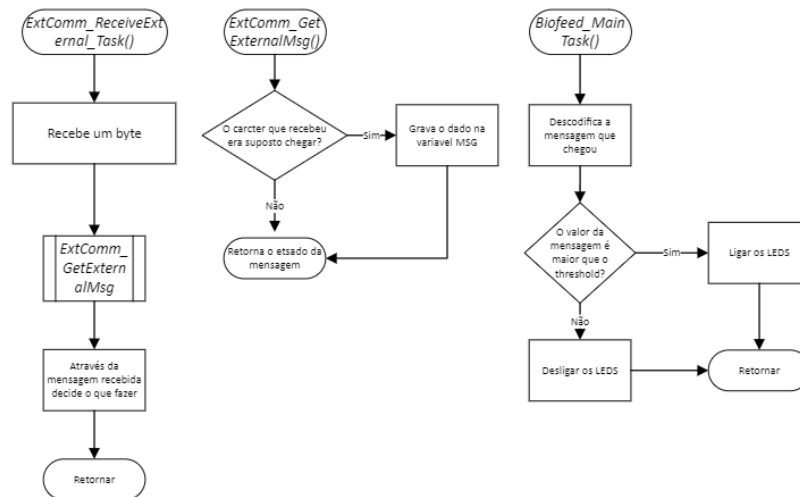


FIGURA 4- FLUXOGRAMA DO ALGORITMO A CRIAR.

Foi necessário ainda a realização de um projeto Simulink para estabelecer a ligação entre o sistema de sensores e o microcontrolador. Este projeto divide-se em três zona a zona Send START, a zona Read Data, e zona Send Data. A zona START estabelece a comunicação com o sistema de aquisição, a zona Read Data, tal como o nome indica, lê e descodifica os dados que chegam. Na zona de Send Data criou-se uma função que codifica os dados, sendo este depois enviados para o STM. Esta função pode ser observada na Figura 5 e resume-se a mandar um start_byte, tamanho dos dados, os dados codificados e um stop_byte. Definiu-se que o start_byte seria 0x0d, os dados a serem enviados teriam todos um tamanho constante de 4 bytes e que o stop_byte seria 0x0a.

Os dados codificados foram feitos através dos mesmos representados no formato científico, com os primeiros 3 dígitos dos dados e após isto o inverso do seu expoente de base 10. Por exemplo caso o dado a ser enviado seja 0,000365, teremos a sua representação científica como $3,65 \times 10^{-4}$ será enviado as representações dos valores 3, 6, 5, 4 por esta ordem.

```

function msg = teste(ch1)
    start_byte = 0x0d;
    stop_byte = 0x0a;
    data_lenght = 4;
    data=[0,0,0,0];
    i=1;
    while (i<4)
        if (fix(ch1/10)==0 && i==1)
            data(4)=data(4)+1;
        else
            data(i)=fix(ch1/10);
            i=i+1;
        end
        ch1=ch1*10-(fix((ch1/10))*10);
        disp(ch1)
    end
    msg=[start_byte,data_lenght,data,stop_byte];
end

```

FIGURA 5 – CÓDIGO PARA A FUNÇÃO QUE CODIFICA A MENSAGEM.

Utilizou-se por fim a ferramenta *Keil uvision 5* para tratar da programação do microcontrolador. Para isso criou-se um projeto na linguagem C para então ser possível realizar o feedback. Neste projeto existem dois ficheiros com maior relevância o *Extcomm.c* e o *Main.c*.

No ficheiro *Extcomm.c* é criada uma tarefa chamada, onde recebe os valores que foram mandados pelo simulink, chama a função *ExtComm_GetExternalMsg* que

identifica se o caracter que foi recebido é o esperado ou não para que seja possível dar seguimento á mensagem.

A função *ExtComm_GetExternalMsg*, que é possível ser observada na Figura 6 baseou-se na codificação criada anteriormente para fazer a verificação, isto é começar com um start byte, o tamanho dos dados, ou seja 4, os dados e um stop_byte. Para isso criou-se duas variáveis estáticas *index* e *state*, em que o *index* identifica em que posição dos dados estamos e o *state* em qual das 4 partes da mensagem estamos. Para isto criou-se um conjunto de switch-cases para a variável *state* em que:

Caso a mensagem esteja correta, mas ainda não foi terminada retornaria 1 e grava na variável *Msg*;

Caso a mensagem esteja correta e já terminou retornaria 0;

Caso o tamanho da mensagem seja demasiado grande, isto é, maior que 128 retornaria 2;

Caso esteja à espera de um stop_byte e este não foi o que foi mandado retornaria 3;

```
uint8_t ExtComm_GetExternalMsg(ExtComm_Data *Msg, uint8_t c)
{
    static uint8_t state = 0;
    static uint8_t index = 0;

    switch (state)
    {
        case 0:
            if (c == S0F_) {
                state = 1;
                return 1;
            }
        case 1:
            if (c <= 128) {
                Msg->length = c;
                state = 2;
                return 1;
            }
            else {
                return 2;
            }
        case 2:
            if (index < Msg->length) {
                Msg->data[index] = c;

                if (index == (Msg->length - 1)) {
                    state = 3;
                }
                return 1;
            }
            else {
                return 3;
            }
        case 3:
            if (c != EOF_) {
                return 3;
            }
            else {
                return 0;
            }
    }
}
```

FIGURA 6 – CÓDIGO DA FUNÇÃO AUXILIAR DA TAREFA.

Já no main começou-se por definir a variável global *threshold* onde esta é um float, de valor igual ao *threshold* definido anteriormente. Criou-se também uma tarefa chamada *Biofeed_MainTask* que usa os dados recebidos na tarefa anterior e os descodifica, para que seja comparado com o *threshold*, acendendo os LED's caso o valor recebido seja maior e os desligar quando forem menores. Na Figura 7 é possível então observar o código criado para esta tarefa.

```

void Biofeed_MainTask(void * pvParameters)
{
    ExtComm_Data ReceivedMsg;

    int EMG=0;
    while(1)
    {
        if (xQueueReceive(hQueueReceivedExternal, &ReceivedMsg, portMAX_DELAY) == pdTRUE){
            int cont=0;

            while (cont<4){
                int vartemp = ReceivedMsg.data[cont];
                if (cont==4){
                    EMG=EMG*(pow(10, (2+vartemp)));
                    EMG=sqrt(pow(EMG, 2));
                }
                else{
                    EMG=EMG*10+vartemp;
                }
            }
            if (EMG > threshold){
                HAL_GPIO_WritePin(GPIOD, GPIO_PIN_12|GPIO_PIN_13|GPIO_PIN_14|GPIO_PIN_15, GPIO_PIN_SET);
            }
            else{
                HAL_GPIO_WritePin(GPIOD, GPIO_PIN_12|GPIO_PIN_13|GPIO_PIN_14|GPIO_PIN_15, GPIO_PIN_RESET);
            }
        }
    }
}

```

FIGURA 7 – CÓDIGO DA TAREFA BIOFEED_MAINTASK.

3 Resultados e Discussão

3.1 Dados recolhidos

Após o tratamento dos dados construiu-se os gráficos relativos á taxa de atividade muscular e dos ângulos nos momentos do estudo. Começou-se por analisar os dados relativos ao primeiro protocolo, isto é, dados relativos á marcha do participante.

Na Figura 8 é possível observar os dados relativos ao EMG de ambas as pernas. Observou-se que para o participante a perna esquerda houve valores entre 7,02% e 74,87%, enquanto que, na perna direita obteve-se valores entre 1,29% e 50,31%. O facto de que os valores sejam superiores na perna esquerda, pode-se dever ao estado participante. Uma vez que o participante esteve lesionado na perna esquerda, podendo fazer com que as suas Máximas voluntarias contrações sejam menores, mantendo a força necessária para andar.

É de se notar ainda que os picos de ambos sinais estão desfasados, uma vez que o movimento e contrações das pernas são desfasados, isto é, quando uma perna esta a se levantar do chão e a fazer força a outra está em repouso. Nota-se ainda a existência alguns picos menores representados os momentos onde apenas os dedos deixam estão apoiados no solo, sendo que o pico maior representa o momento em que o pé volta a estar apoiado no solo.

Por fim é ainda possível notar os momentos onde o participante alterou a sua rota e voltou-se para continuar a andar, por exemplo nos momentos entre 13,5 segundo e 15,5 segundos. Nos passos anteriores e posteriores aos momentos de viragem á ainda uma desaceleração do movimento, o que provoca uma diminuição da atividade muscular nestes movimentos.

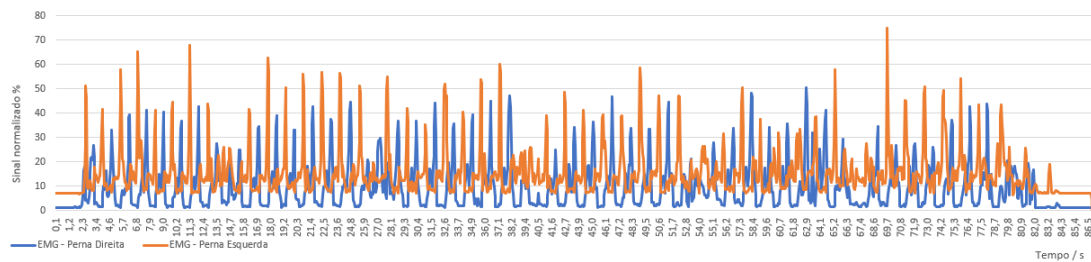


FIGURA 8 – REPRESENTAÇÃO GRÁFICA DA TAXA DE ATIVIDADE AO LONGO DO TEMPO PARA O PRIMEIRO PROTOCOLO.

Na Figura 9 é observável os dados angulares recolhidos para a perna direita do participante. Os valores medidos na cintura variam entre $-12,40^\circ$ e $24,93^\circ$, os valores do joelho entre $0,67^\circ$ e $74,03^\circ$ enquanto os do tornozelo variaram entre $18,91^\circ$ e $23,96^\circ$. É notável a existência de picos semelhantes aos encontrados na Figura 8, tendo um pequeno desfaseamento entre os picos do tornozelo e da cintura. Nestes dados é possível também reparar os momentos onde o participante voltou-se, isto verifica-se nas zonas onde há uma notável diminuição dos ângulos e uma menor frequência entre dois picos, tudo isto para os três casos.

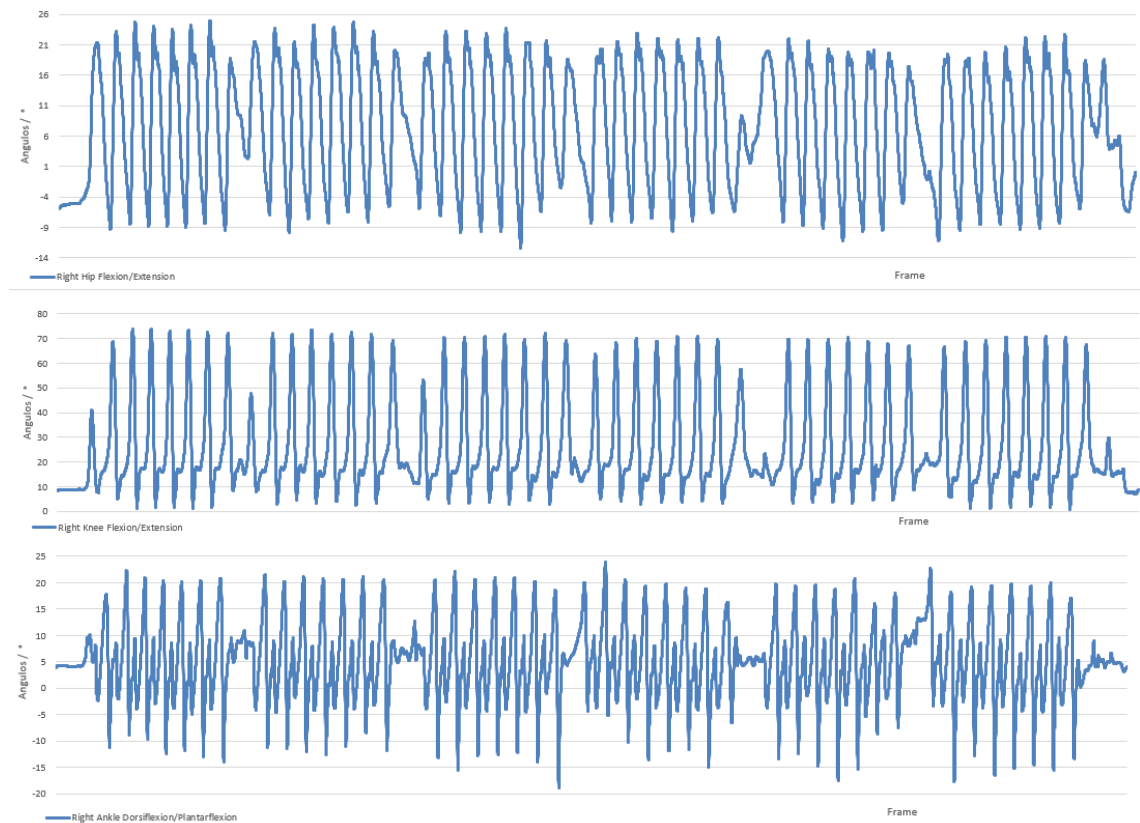


FIGURA 9 – REPRESENTAÇÃO GRÁFICA DA DOS ÂNGULOS AO LONGO DOS FRAMES PARA O PRIMEIRO PROTOCOLO. (A). PARA ÂNGULOS DA CINTURA. (B) PARA OS ÂNGULOS DO JOELHO. (C) PARA OS ÂNGULOS DO TORNOZELO.

Quando comparado com a literatura [2] dos dados angulares para a cintura notou-se que os dados são semelhantes, representando apenas um pico negativo de cerca de -10° e depois um pico positivo de cerca de 30° , sendo que no participante em que foi medido tem um pico positivo inferior em cerca de 10° .

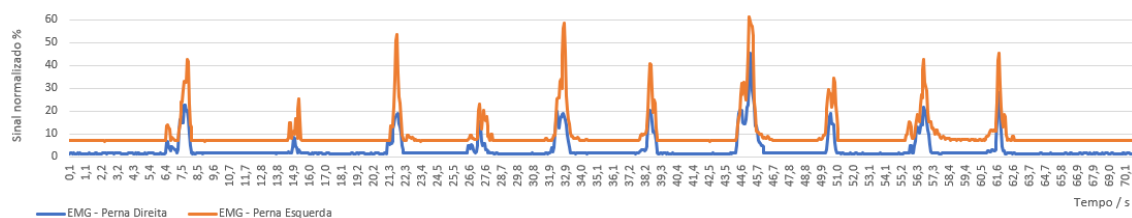
Quando ao joelho na literatura é representado com um pico inicial de cerca de 30° , após um pico negativo de cerca de 15° , vindo um segundo pico no 65° e acabando o

movimento por volta do 0°. No participante assemelha-se esta descrição á exceção do primeiro pico negativo que não se encontra presente na maioria dos passos dados.

Por fim no tornozelo têm-se apenas um pico negativo na zona dos -10°, uma subida até os 10°, e tendo outro decréscimo até os -30° voltando até ao fim do movimento aso 5°. No participante foi representado á semelhança deste, mas representado inversamente e tendo um pico máximo na zona dos 20° onde seria por volta dos 30, na representação invertida.

Na literatura em relação ao EMG [3], diz que está deve ter um pico médio por volta dos 37,95%, o que comprova que a perna direita do participante pode ser considerada correta e saudável, enquanto a perna esquerda teve a necessidade de fazer um maior esforço, ou tem uma menor capacidade de atingir forças elevadas, visto que tem um valor médio superior em cerca de 15%.

Na Figura 10 observa-se os dados de EMG recolhidos para ambas as pernas. Observou-se que ao se levantar é necessária mais força, uma vez que em cada dois picos começando no primeiro são maiores que os restantes, tendo o levantar da perna esquerda variar entre 42,73% e 59,67%, sentar da perna esquerda entre 23,17% e 45,29%, o



levantar da perna direita entre 19,15% e 45,5% e o sentar da perna direita entre 8,60% e 28,24%. Voltou-se também a notar que o participante necessitou mais força na perna esquerda, tendo picos mais elevados.

FIGURA 10 – REPRESENTAÇÃO GRÁFICA DOS DADOS DE EMG RECOLHIDOS PARA O MOVIMENTO DE SENTAR-LEVANTAR E LEVANTAR-SENTAR.

Na Figura 11 é possível analisar que no joelho direito quando sentado o participante manteve o ângulo de cerca de 95 ° e de tornozelo na zona dos 9°. Na zona da cintura o participante realizou o movimento tendo um angulo de cerca de -5° quando de pé e cerca de 60 ° quando sentado, havendo um pico de cerca de 90° quando é alterado a posição, sendo este pico ligeiramente mais acentuado quando se senta. Por fim no tornozelo tem uma linha de base na zona dos 10°, tendo uma variação de 5 ° nos momentos em que há troca de posição.

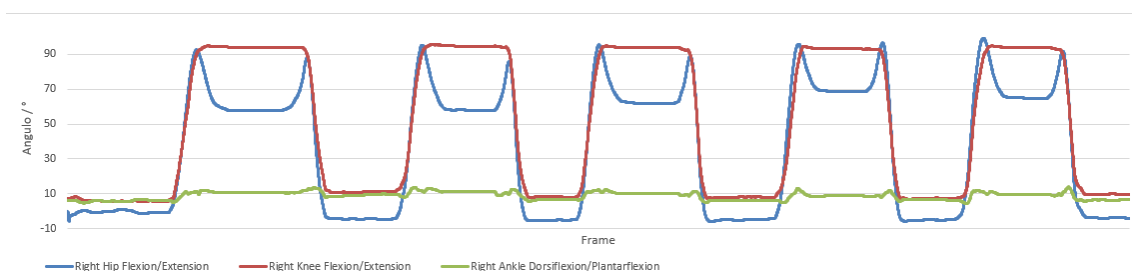


FIGURA 11 - REPRESENTAÇÃO GRÁFICA DOS DADOS INERCIAIS RECOLHIDOS PARA O MOVIMENTO DE SENTAR-LEVANTAR E LEVANTAR-SENTAR EM TRÊS POSIÇÕES (CINTURA, JOELHO, TORNOZELO) NA PERNA DIREITA.

Estes dados inerciais quando comparados com a literatura encontrada [4], diz-nos que quanto aos ângulos do tornozelo, têm uma semelhança em forma para ambos os movimentos, sentado-levantado, como levantado-sentado, mas os valores de pico para ambos são muito inferiores nos dados recolhidos, cerca de 10° menor.

Quanto aos ângulos do joelho são representados identicamente na literatura e nos recolhido pelo participante, para ambos os movimentos de levantar e sentar. O mesmo não acontece na cintura, onde tem a mesma forma, mas o participante realizou uma rotação superior em 10°, quando comparada com a literatura para ambos os movimentos.

Quanto aos EMG a literatura [5] diz que deverá ter valores de pico de media 9.22 com um desvio padrão de 4,23%, ou seja, variar entre 4,99% e 13,45%. Assim como o participante tem valores de pico de cerca de 20%, este estará acima do previsto por volta de 6% do valor das máximas contrações voluntárias.

3.2 Analise dos Modelos

Quanto ao modelo de random forest com divisão a 66% teve uma accuracy de 99.2072%, representando o número de acertos, enquanto com Cross Validation de 10 dobras teve 99.2663% de acertos. Assim o modelo com cross Validation têm uma taxa de acerto superior, podendo ser considerada um melhor modelo. Outros factos que também contribuem com esta afirmação é os Valores da taxa de True Positives que é superior na Cross Validation e representa a taxa que foi verdadeiramente representada como verdadeira. Os Falso Positivos são iguais e representa o número de dados que foram falsamente denominados como positivos. A precisão representando a taxa de verdadeiros positivos com o total de positivos identificados também apoiam a afirmação. O MCC confirma por fim que esta o Cross Validation seria o melhor método tendo um valor maior neste método. Nas Figuras 12 (a) e (b) observam-se os dados que foram usados para comparar para ambos os casos.

O facto de o *Cross Validation* ser um melhor método pode se dever ao facto que quanto maior o número de dados de treino melhor será o modelo, visto que com uma maior quantidade e variabilidade de dados o modelo é mais bem treinado para um maior número de casos.

Para os outros modelos têm resultados menores ao modelo de *random forest* no geral para todos os campos falados anteriormente. Pode-se, contudo, argumentar que poderia ser usado o método do *Multilayer Perceptron* uma vez que tem uma *accuracy* relativamente alta, 91.0422%. Para além de que para a aplicação que estamos a procurar o uso de modelos de avaliação deve ser relativamente rápida, para que seja possível uma tomada de decisão mais rápida e adequada. Visto que neste método demora 0.42 segundo a aplicar, enquanto que, no método da Random Forest demora 3.17 segundos, podemos argumentar a utilização do modelo MLP ou Multilayer Perceptron. Pode-se verificar

ainda nas Figuras 12 (c) e (d) a tabela de *accuracy* detalhada por classes para os métodos de *Decision Table* e MLP, respetivamente.

TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
0,999	0,000	1,000	0,999	0,999	0,999	1,000	1,000	sitting	0,999	0,000	1,000	0,999	0,999	0,999	1,000	1,000	sitting
0,989	0,002	0,994	0,989	0,991	0,989	1,000	0,999	sittingdown	0,990	0,002	0,994	0,990	0,992	0,990	1,000	1,000	sittingdown
0,995	0,002	0,991	0,995	0,993	0,991	1,000	1,000	standing	0,996	0,002	0,991	0,996	0,994	0,992	1,000	1,000	standing
0,982	0,001	0,994	0,982	0,988	0,985	1,000	0,999	standingup	0,983	0,002	0,994	0,983	0,988	0,986	1,000	0,999	standingup
0,996	0,005	0,982	0,996	0,989	0,986	1,000	0,999	walking	0,995	0,004	0,985	0,995	0,990	0,988	1,000	1,000	walking
0,992	0,002	0,992	0,992	0,992	0,990	1,000	1,000	Weighted Avg.	0,993	0,002	0,993	0,993	0,993	0,991	1,000	1,000	Weighted Avg.

TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
0,937	0,047	0,829	0,937	0,880	0,851	0,987	0,938	sitting	0,986	0,002	0,991	0,986	0,989	0,986	1,000	0,998	sitting
0,714	0,101	0,640	0,714	0,675	0,589	0,907	0,683	sittingdown	0,933	0,033	0,877	0,933	0,904	0,879	0,982	0,941	sittingdown
0,904	0,065	0,775	0,904	0,834	0,793	0,973	0,881	standing	0,937	0,034	0,872	0,937	0,903	0,879	0,979	0,943	standing
0,566	0,062	0,699	0,566	0,625	0,547	0,875	0,712	standingup	0,838	0,020	0,913	0,838	0,874	0,845	0,935	0,899	standingup
0,656	0,032	0,836	0,656	0,735	0,686	0,930	0,834	walking	0,861	0,022	0,907	0,861	0,883	0,855	0,970	0,908	walking
0,754	0,062	0,755	0,754	0,749	0,692	0,934	0,809	Weighted Avg.	0,910	0,022	0,912	0,910	0,910	0,888	0,973	0,937	Weighted Avg.

FIGURA 12 – TABELA DE ACCURACY DETALHADA POR CLASSES. (A) RANDOM FOREST COM PERCENTAGE SPLIT A 66%. (B) RANDOM FOREST COM CROSS VALIDATION COM 10 FOLDS. (C) DECISION TABLE COM PERCENTAGE SPLIT A 66%. (D) MULTILAYER PERCEPTRON COM PERCENTAGE SPLIT A 66%.

3.3 Teste do Código

Para o teste deste código criado utilizou-se em vez de diretamente o sistema de recolha utilizado inicialmente, utilizou-se dados recolhidos previamente, onde se calculou o threshold para este participante, sendo este de cerca de 0,000104V. Na Figura 13 observa-se então a representação com os seus valores de pico.

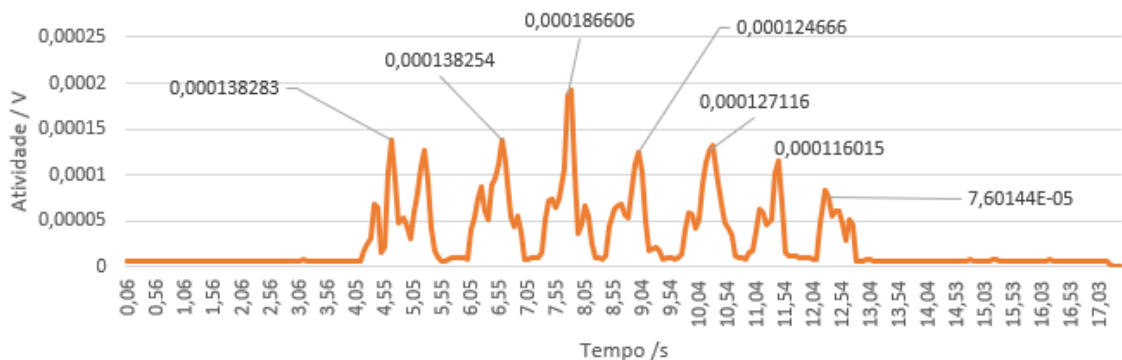


FIGURA 13 – ANÁLISE E REPRESENTAÇÃO GRÁFICA DOS DADOS DO DATASET USADO PARA TESTE

Após isto ligou-se os fios 2 do adaptador ao PA2 do microcontrolador, o 3 do adaptador ao PA3 microcontrolador e 10 do adaptador ao Ground do microcontrolador. Ligou-se também o microcontrolador e o adaptador por esta mesma ordem para haver a ligação entre o simulink e o microcontrolador. Após isto foi necessário dar switch ao manual switch do projeto simulink para alterar o fluxo de dados do sistema captador para o ficheiro Excel.

Assim correu-se o programa no microcontrolador, uma vez que não existiam erro no código ao dar build. Pode-se então correr o projeto no simulink vendo que de tempos em tempos o LED ligava por algum tempo e depois se desligava. Presumindo-se que o programa funcionava como desejado.

4. Conclusão

O presente projeto teve como objetivo a criação das partes de um sistema capaz de dar um feedback ao utilizador do sistema. Para isso começou-se através da recolha de dados que foram usados tanto para a construção de algoritmos de decisão capazes de identificar o tipo de movimento feito, para além da definição de thresholds que são usados para definir a boa ou má realização de um movimento.

A nível de futuras melhorias tem-se de identificar a clara junção dos dois últimos pontos, criando um sistema que identifique o tipo de movimento, e classificar a qualidade deste para cada tipo de movimento identificável.

5. Bibliografia

- [1] W. Ugulino, D. Cardador, K. Vega, E. Velloso, R. Milidiú, and H. Fuks, “Wearable Computing: Accelerometers’ Data Classification of Body Postures and Movements,” 2012, pp. 52–61.
- [2] Ismailidis, Petros & Hegglin, Lea & Egloff, Christian & Pagenstert, Geert & Kernen, Rolf & Eckardt, Anke & Ilchmann, Thomas & Nüesch, Corina & Mündermann, Annegret. (2021). Side to side kinematic gait differences within patients and spatiotemporal and kinematic gait differences between patients with severe knee osteoarthritis and controls measured with inertial sensors. *Gait & Posture*. 84. 24-30. 10.1016/j.gaitpost.2020.11.015.
- [3] Tyler D. Chuang, Stacey M. Acker, (2019) Comparing functional dynamic normalization methods to maximal voluntary isometric contractions for lower limb EMG from walking, cycling and running, *Journal of Electromyography and Kinesiology*, Volume 44, Pages 86-93.
- [4] Pinheiro, C., Lopes, J. M., Moreira, L., Sanz-Merodio, D., Figueiredo, J., Santos, C. P., & Garcia, E. (s.d.). Kinematic and kinetic study of sit-to-stand and stand-to-sit.
- [5] Sutcu G, Yalcin AI, Ayvat E, OnursalKılınç Ö, Ayvat F, Doğan M, Harput G, Aksu Yıldırım S, Kılınç M (2019) Electromyographic activity and kinematics of sit-to-stand in individuals with muscle disease. *Neurol Sci* 40(11):2311–2318. <https://doi.org/10.1007/s10072-019-03974-5>