

UNIVERSIDADE DE SÃO PAULO
ESCOLA DE ENGENHARIA DE SÃO CARLOS

JOÃO MARCOS PERON BATAGLIA

Utilização de dispositivos móveis na estimulação elétrica neuromuscular

São Carlos

2020

JOÃO MARCOS PERON BATAGLIA

Utilização de dispositivos móveis na estimulação elétrica neuromuscular

Monografia apresentada ao Curso de Engenharia Elétrica com Ênfase em Eletrônica, da Escola de Engenharia de São Carlos da Universidade de São Paulo, como parte dos requisitos para obtenção do título de Engenheiro Eletricista.

Orientador: Prof. Dr. Alberto Cliquet Jr

VERSÃO CORRIGIDA

São Carlos
2020

AUTORIZO A REPRODUÇÃO TOTAL OU PARCIAL DESTE TRABALHO, POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO, PARA FINS DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

Ficha catalográfica elaborada pela Biblioteca Prof. Dr. Sérgio Rodrigues Fontes da EESC/USP com os dados inseridos pelo(a) autor(a).

B335u Bataglia, João Marcos Peron
Utilização de dispositivos móveis na estimulação elétrica neuromuscular / João Marcos Peron Bataglia; orientador Alberto Cliquet Junior. São Carlos, 2020.

Monografia (Graduação em Engenharia Elétrica com ênfase em Eletrônica) -- Escola de Engenharia de São Carlos da Universidade de São Paulo, 2020.

1. Estimulação Elétrica Neuromuscular EENM. 2. Aplicativo para Android. 3. Microcontrolador ESP32. 4. Tetraplegia. I. Título.

Eduardo Graziosi Silva - CRB - 8/8907

FOLHA DE APROVAÇÃO

Nome: João Marcos Peron Bataglia

Título: "Utilização de dispositivos móveis na eletroestimulação neuromuscular"

Trabalho de Conclusão de Curso defendido e aprovado
em 26/6/2020,

com NOTA 10,0 (dez, zero), pela Comissão Julgadora:

Prof. Titular Alberto Cliquet Junior - Orientador - SEL/EESC/USP

Dr. Renato Varoto - UNICAMP

Mestre Renata Manzano Maria - Doutoranda - SEL/EESC/USP

Coordenador da CoC-Engenharia Elétrica - EESC/USP:
Prof. Associado Rogério Andrade Flauzino

AGRADECIMENTOS

Em primeiro lugar, agradeço a minha família e a minha namorada Paula pelo apoio e força durante a jornada da graduação. Deixo também um especial agradecimento ao LABCIBER e todos os seus integrantes pela oportunidade de desenvolver esse trabalho e por confiarem nos meus esforços. Finalmente, mas não menos importante, deixo um agradecimento à Tatiana por todo o apoio psicológico.

RESUMO

BATAGLIA, J. M. P. **Utilização de dispositivos móveis na eletroestimulação neuromuscular.** 2020. 108p. Monografia (Trabalho de Conclusão de Curso) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2020.

Existe no mundo uma quantidade grande de indivíduos com lesões medulares, em específico aquelas que apresentam como principal efeito a tetraplegia, ou seja, a perda completa ou parcial dos movimentos dos membros superiores e inferiores.

Tendo em mente esse contexto, várias abordagens são realizadas buscando o tratamento desse tipo de lesão com o intuito da redução ou até a completa recuperação dos indivíduos lesionados. Uma dessas abordagens, é a estimulação elétrica neuromuscular, que atualmente é aplicada no LABRAL - Laboratório de Biomecânica e Reabilitação do Aparelho Locomotor da UNICAMP, o qual apesar de funcionar corretamente apresenta algumas limitações.

Portanto, esse é um projeto de conclusão de curso cujo objetivo principal é apresentar uma abordagem moderna para a estimulação elétrica. Buscando complementar o desenvolvimento de um equipamento vestível, desenvolveu-se no decorrer deste trabalho um aplicativo para celular com sistema operacional Android e também um firmware para o microcontrolador ESP32, visando a realização de eletroestimulação sem fio.

Ao final do projeto, obteve-se um aplicativo completamente funcional adicionado de um firmware que se comunica de forma estável com o aplicativo, além do fato que o conjunto se mostrou capaz de realizar a estimulação elétrica seguindo o protocolo de estimulação.

Palavras-chave: Estimulação Elétrica Neuromuscular EENM, Aplicativo para Android, Microcontrolador ESP32, Tetraplegia.

ABSTRACT

BATAGLIA, J. M. P. Usage of Mobile application in Neuromuscular Electrostimulation. 2020. 108p. Monografia (Trabalho de Conclusão de Curso) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2020.

Nowadays, there is a high number of people in the world that suffered spinal cord injury (SCI), being tetraplegia in many cases its main effect. It means losing some or all of upper and lower limbs movement.

With that being said, numerous treatments were developed focusing on these kind of injury, trying in many different ways, to reduce impacts or even to completely heal the patient in such condition. One of these treatments is the Neuromuscular Electrical Stimulation - NMES, which is applied to patients in the Laboratório de Biomecânica e Reabilitação do Aparelho Locomotor - LABRAL - UNICAMP. Although the system has shown to be effective, its use has limitations such as precision, comfort and mobility.

To minimize these limitations, this project aim to contribute with a novel technology to improve electrical stimulation in treatment of SCI patients. Focusing in complementing the development of a wearable equipment to perform electrical stimulation, it was developed a mobile application (Android) and a new microcontroller firmware (ESP32) to allow wireless NMES sections.

As a result, the system was successfully integrated to the wearable and made possible to execute wireless NMES in patients.

Key-words: Neuromuscular Electrical Stimulation - NMES, Android mobile application, ESP32 microcontroller, tetraplegia.

LISTA DE ILUSTRAÇÕES

Figura 1 - EENM aplicada ao paciente no LABRAL.....	25
Figura 2 – Vestível.....	26
Figura 3 - ESP32 e plataforma Arduino.....	27
Figura 4 - Android Studio, Java e FIGMA.....	28
Figura 5 – Aplicativo.....	29
Figura 6 – Firmware.....	30
Figura 7 - Tela de boas-vindas.....	31
Figura 8 - Tela de ajuste de canais.....	32
Figura 9 - Tela de ajuste de protocolo.....	33
Figura 10 - Ajuste de receita.....	34
Figura 11 - Caixa de dialogo.....	35
Figura 12 - Card povoado.....	36
Figura 13 - Tela de resumo e envio.....	37
Figura 14 - Processamento do protocolo.....	39
Figura 15 - Execução do protocolo.....	40
Figura 16 - Circuito de teste.....	42
Figura 17 – Ondas.....	42

LISTA DE TABELAS

Tabela 1 - Lesões Medulares	22
-----------------------------------	----

LISTA DE ABREVIATURAS E SIGLAS

EESC	Escola de Engenharia de São Carlos
USP	Universidade de São Paulo
UNICAMP	Universidade Estadual de Campinas
LABCIBER	Laboratório de Biocibernética e Engenharia de Reabilitação
LABRAL	Laboratório de Biomecânica e Reabilitação do Aparelho Locomotor
EENM	Eletroestimulação Neuromuscular

SUMÁRIO

1 INTRODUÇÃO.....	22
1.1 Lesão medular.....	22
1.2 Tratamento.....	23
1.3 Protocolo de estimulação elétrica.....	24
1.4 Aparato de estimulação neuromuscular.....	24
1.5 Objetivos.....	25
2 DESENVOLVIMENTO E RESULTADOS.....	27
2.1 Materiais.....	27
2.2 Desenvolvimentos.....	28
2.2.1 Tela de boas-vindas.....	30
2.2.2 Sistema para calibração de canais.....	31
2.2.3 Sistema de ajuste do protocolo.....	33
2.2.4 Sistema de envio e execução do protocolo.....	36
2.2.4.1 Recepção do protocolo.....	38
2.2.4.2 Recepção de ciclos.....	39
2.2.4.3 Recepção de indicador de início.....	39
2.3 Implementação da comunicação WiFi.....	41
2.4 Testes.....	42
3 CONCLUSÃO.....	43
4 REFERÊNCIAS BIBLIOGRÁFICAS.....	44
APENDICE.....	45

1 INTRODUÇÃO

Paraplegia e tetraplegia são condições geradas por lesão medular, nas quais podem ser perdidos os movimentos e sensibilidade parcial ou total dos membros inferiores e superiores.

Segundo dados fornecidos pelo ministério da saúde nas diretrizes de atenção a pessoa com lesão medular, a incidência de TRM (trauma raquimedular) é de 40 casos novos/ano/milhão de habitantes, ou seja, cerca de 6 a 8 mil casos novos por ano.

1.1 Lesão medular

De acordo com o ministério da saúde em suas diretrizes de atenção a pessoa com lesão medular (2013), é considerado como uma lesão medular toda injúria às estruturas contidas no canal medular (medula, cone medular e cauda equina), podendo levar a alterações motoras, sensitivas, autonômicas e psicoafetivas.

Como referência para determinar o nível da lesão medular, utiliza-se os pares de nervos presentes na medula espinhal. A coluna vertebral humana pode ser dividida em regiões: região cervical composta por 7 vértebras (C1 a C7), a região torácica, por 12 vértebras (T1 a T12), a região lombar, por 5 vértebras (L1 a L5) e a região sacral, por 5 vértebras (S1 a S5). Vale a ressalva que as lesões são determinadas pelos pares de nervos e não propriamente pela vertebra de forma que é possível se ter uma lesão C8, sendo que não existe uma vertebra C8 (a lesão C8 é compreendida entre as vértebras C7 e T1, onde o nervo está alocado).

A gravidade da lesão medular é maior quanto mais próximo do crânio ela ocorrer. A Tabela 1 apresenta um resumo dos níveis de lesão e efeitos consequentes.

Tabela 1 - Lesões medulares

Região	Nível da lesão	Efeito	Condição
Cervical	C1 - C3	Respiração prejudicada, fala limitada, movimentos do pescoço e cabeça dificultados.	Tetraplegia

Cervical	C4	Respiração praticamente preservada, melhor movimentação da cabeça e pescoço, elevação dos ombros preservadas.	Tetraplegia
Cervical	C5	Todas as funções preservadas apresentadas em C4 adicionado da capacidade de flexão do cotovelo.	Tetraplegia
Cervical	C6	Todas as funções preservadas apresentadas em C5 adicionado da extensão do punho.	Tetraplegia
Cervical	C7	Todas as funções preservadas apresentadas em C6 adicionado da extensão dos cotovelos.	Tetraplegia
Cervical/Torácica	C8 - T1	Todas as funções preservadas apresentadas em C7 adicionado do movimento dos dedos.	Tetraplegia
Torácica	T2 - T6	Controle da cabeça, pescoço e membros superiores preservados.	Paraplegia
Torácica	T7 - T11	Músculos abdominais e extensores de tronco ativos parcial ou totalmente.	Paraplegia
Torácica/Sacral	T12 - L1	Todas as funções preservadas apresentadas em T11 adicionado do controle da pelve.	Paraplegia

Fonte: ABRAFIN (2015)

O escopo deste trabalho foi estabelecido para atender indivíduos tetraplégicos, mas que possuam algumas partes do membro superior ainda funcionais, sendo mais específico, indivíduos com níveis de lesões compreendidos entre os pares de nervos C4 e C7.

1.2 Tratamento

Sabe-se que quando ocorre uma lesão medular, o local da lesão é dominado por um ambiente predominantemente tóxico o que acarreta na morte de neurônios devido a uma baixa na quantidade de energia e oxigênio. Em um curto espaço de tempo após a lesão, células gliais recobrem neurônios intactos da medula espinhal, criando uma barreira que, apesar de oferecer proteção, impedem a regeneração.

Dessa forma, desenvolveu-se uma terapia chamada de Estimulação Elétrica Neuromuscular (EENM) a qual utiliza o princípio da neuroplasticidade e permite simular o estímulo de neurônios motores por sinais elétricos - que seriam enviados pelo cérebro em

condições normais. Este estímulo, feito de forma consistente, instiga a busca por novos caminhos para a reconexão das regiões afetadas pela lesão.

No Laboratório de Biomecânica e Reabilitação do Aparelho Locomotor - LABRAL/UNICAMP, atualmente são realizados protocolos de EENM que se utilizam da plasticidade para tratamento de lesados medulares. No procedimento, eletrodos de superfície são posicionados em regiões específicas de acordo com o movimento desejado. Uma corrente elétrica é induzida de forma que o músculo sofre contração, gerando o movimento ou sequência de movimentos, o que é determinado pelo protocolo estabelecido. No laboratório, são realizados treinamentos consecutivos de movimentos funcionais - simulam algum movimento natural.

1.3 Protocolo de estimulação elétrica

A sessão de EENM aplicada no LABRAL para membros superiores pode ser dividida em até 12 fases que são repetidas idealmente pelo paciente durante sessões de 20 minutos. Em cada uma das fases, até três canais podem ser utilizados, ativando três músculos (no caso de estimulação simultânea dos dois braços) por um período adequado de modo que a contração ocorra.

Dessa maneira, ao executar sequencialmente as 12 fases, os músculos são contraídos em uma sequência tal, que o paciente efetua o movimento funcional de levar um copo de água até a boca (copo é emulado por um cilindro com peso) e retornar até a posição inicial.

Para que tal estimulação seja efetuada, o aparato de estimulação utilizado estimula o músculo do paciente com um padrão de ondas quadradas que possui uma sequência de quatro pulsos de 100 us separados por um período entre pulsos de 100 us, os quais são reproduzidos a uma frequência de 25 Hz. Cada pulso possui amplitude que varia com a sensibilidade de cada paciente, podendo atingir até 70 V em alguns.

1.4 Aparato de estimulação neuromuscular

Atualmente, o LABRAL utiliza para suas sessões de EENM um aparato composto por:

- Eletrodos ativos de superfícies autoadesivos, fixados aos pacientes também com o auxílio de fitas;
- Unidade estimuladora, cuja finalidade é gerar os pulsos retangulares;
- Fios para realizar as conexões entre os eletrodos e a unidade estimuladora;
- Microcomputador para realizar a configuração de cada sessão de estimulação.

Figura 1 – EENM aplicada ao paciente no LABRAL



Todo esse aparato apesar de funcional apresenta limitações no que se refere ao conforto do paciente e precisão. Nesse sentido, é muito recorrente durante uma sessão a ocorrência de pinçamentos dos fios, o que causa não somente a parada do processo de estimulação como também a retirada forçada dos eletrodos fixados ao corpo do paciente. Outra limitação diz respeito ao local da estimulação, o qual devido a quantidade de equipamentos deve ser mantido sempre no mesmo local impedindo a mobilidade. Finalmente, devido à grande quantidade de aparelhos para a estimulação, essa não é capaz de abranger muitos pacientes, gerando mais uma limitação.

1.5 Objetivos

Tendo em vista os obstáculos enfrentados pelo LABRAL durante suas sessões de EENM para membros superiores, e buscando solucionar problemas relacionados ao posicionamento dos eletrodos, assim como eliminar as ocorrências de pinçamentos, foi desenvolvido um vestível, onde todos os eletrodos e fios ficam embutidos, facilitando os movimentos do paciente (VENTURA et al., 2020).

Figura 2 - Vestível



Logo, este projeto possui como objetivo principal complementar a solução vestível anteriormente apresentada, de forma a adicionar conectividade wireless aumentando assim a sua mobilidade e abrangência além de realizar a redução de custos com aparelhagem.

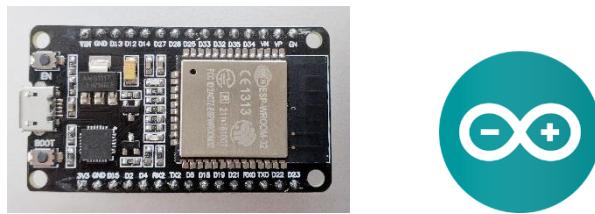
Neste sentido, no decorrer deste projeto, será desenvolvido um aplicativo para dispositivos móveis com sistema operacional Android, cuja finalidade é a de substituir a utilização do microcomputador na configuração da sessão de estimulação. Será também desenvolvido o firmware para o microcontrolador ESP32 cujo intuito é de se comunicar com o aplicativo via WiFi, receber, interpretar e executar o protocolo da estimulação gerada pelo usuário através do aplicativo, podendo ativar até 8 canais diferentes (8 músculos diferentes).

2 DESENVOLVIMENTO E RESULTADOS

2.1 Materiais

Para o desenvolvimento, utilizou-se o ESP32 DEVIKIT V1 Development Board (Shenzhen Doctors of Intelligence & Technology Co. Ltd, Shenzhen, Guangdong, China) para realizar a interface entre o aplicativo para dispositivo móvel e a unidade de amplificação. Essa placa toma por base o módulo microcontrolador ESP-WROOM-32 (Espressif Systems Co. Ltd, Pudong, Shanghai, China) contando com dois microprocessadores de 32-Bits LX6 (Cadence Design Systems, Inc., San Jose, California, SA) e conectividade Wi-Fi e Bluetooth Low Energy (BLE). Em específico, utilizou-se as portas D13, D12, D14, D27, D26, D25, D33, D32 assim como a conectividade WiFi para operá-las.

Figura 3 – ESP32 e plataforma Arduino



Esta placa de desenvolvimento foi programada utilizando a linguagem C de programação e contando com o auxílio do ambiente de desenvolvimento integrado (IDE) Arduino (Arduino S.r.l, Scarmagno, TO, Italy) adicionada de pacotes de desenvolvimento direcionados ao controlador ESP32.

Para o desenvolvimento do aplicativo Android, utilizou-se a linguagem JAVA de programação em conjunto com o ambiente de desenvolvimento Android Studio (Google LLC, Mountain View, CA, USA). Quanto ao design do aplicativo, este foi desenvolvido utilizando a ferramenta gráfica, On-Line, FIGMA (Figma Inc., San Francisco, CA, USA).

Figura 4 – Android Studio, Java e FIGMA



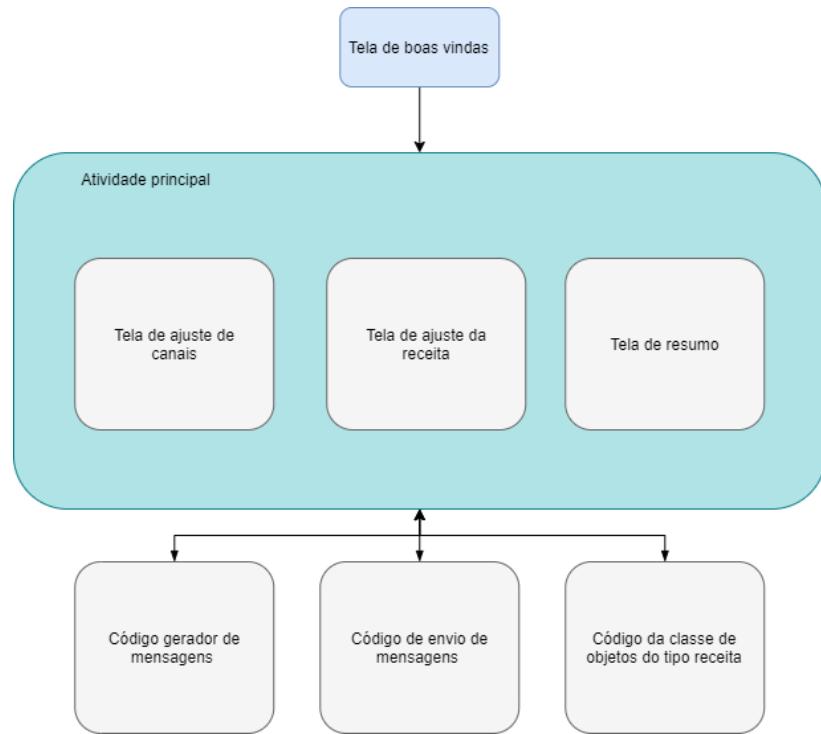
Finalmente para a realização de testes rápidos durante o desenvolvimento, utilizou-se uma Protoboard em combinação com o microcontrolador e uma circuitaria de LEDs e resistores, cuja finalidade foi a de simular os eletrodos. Para realizar testes de bancada, utilizou-se um osciloscópio digital de Tektronix modelo TDS2024 (Tektronix Inc, Beaverton, OR, USA) e também um multímetro digital Fluke (Fluke Corporation, Everett, WA, USA).

2.2 Desenvolvimentos

O desenvolvimento do aplicativo e do firmware do sistema vestível foi feita de forma concomitante. Dessa forma uma descrição de cada parte do sistema será feita da mesma forma, ou seja, apresentado tanto aquilo que foi desenvolvido para o aplicativo, quanto o código do microcontrolador que dá suporte ao anterior.

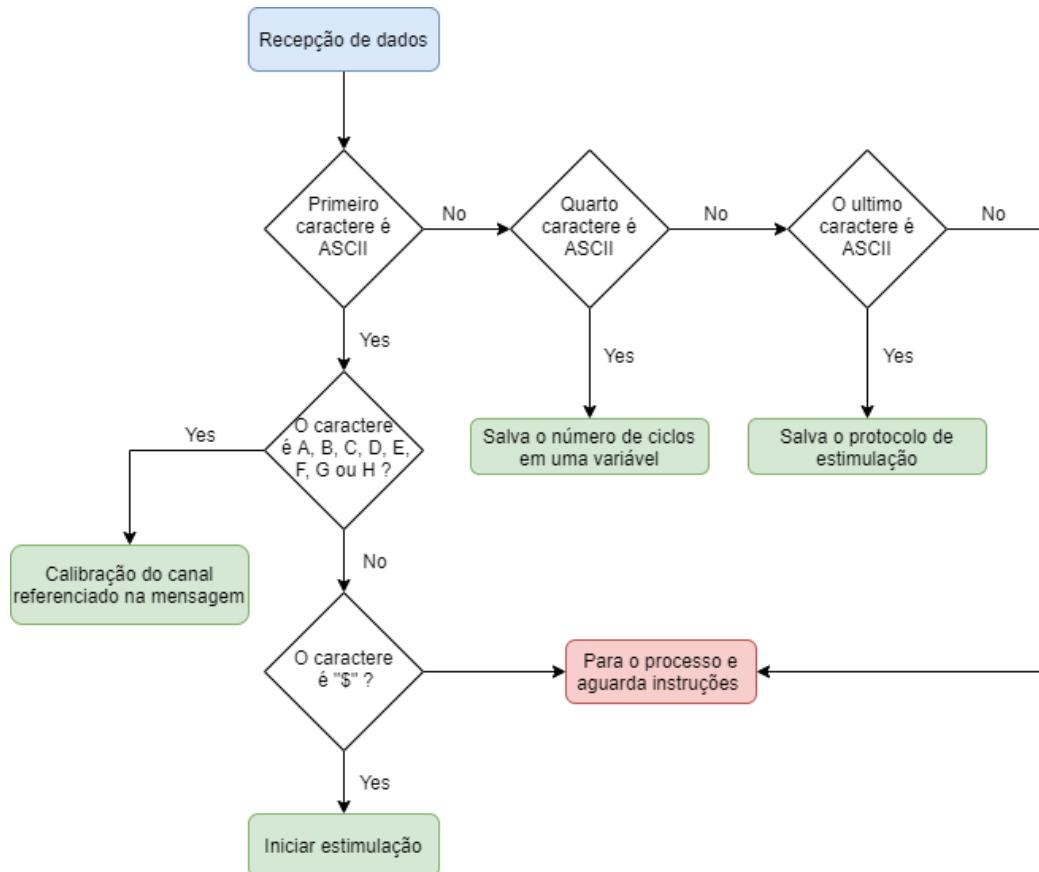
Desenvolveu-se o aplicativo segundo a estrutura apresentada na figura 5.

Figura 5 - Aplicativo



Desenvolveu-se o firmware do microcontrolador segundo a estrutura apresentada na figura 6.

Figura 6 – Firmware



2.2.1 Tela de boas-vindas

O primeiro item a ser desenvolvido foi uma tela de boas-vindas.

Figura 7 – Tela de boas-vindas



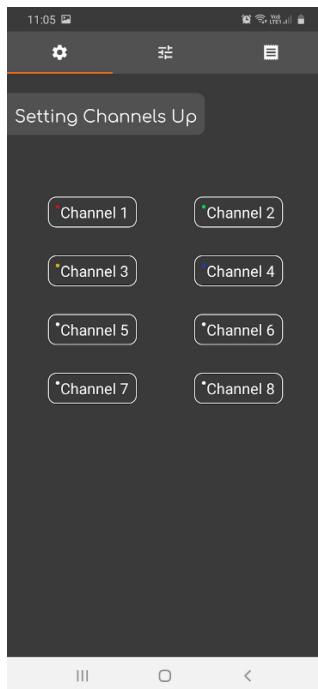
Desenvolveu-se essa tela utilizando por base uma atividade do sistema Android na qual ajustou-se uma duração de 3 segundos. Após o tempo estipulado, o aplicativo é direcionado para a próxima atividade (atividade principal) de tal forma que não é possível retornar a ela a não ser reabrindo o aplicativo.

2.2.2 Sistema para calibração de canais

Na sequência, foi desenvolvido um sistema para o procedimento de calibração dos canais, ou seja, para o aplicativo foi desenvolvido uma forma de seleção e envio de informações a respeito de canais e no microcontrolador foi implementado um sistema de interpretação de mensagens e execução do procedimento de calibração.

A tela de ajuste de canais foi desenvolvida utilizando um fragmento dentro da atividade principal. A escolha do uso de um fragmento se deve a escolha de menu utilizado (posicionado na parte superior da tela), uma vez que é mandatório a utilização de fragmentos para as operações com o tipo de menu selecionado.

Figura 8 – Tela de ajuste de canais



Conforme apresentado na figura 8, foi desenvolvido uma tela contendo 8 botões. Cada botão foi referenciado a um canal específico de forma a ser possível calibrar até 8 canais diferentes. Logo, quando pressionados, os botões sofrem uma mudança de cor indicando o pressionamento e em seguida efetua-se o envio de um caractere do tipo ASCII para o microcontrolador através de “Sockets” (um conjunto formado por um endereço e uma porta, seguindo os protocolos de comunicação TCP/IP).

As mensagens de calibração, consistem em um único caractere do tipo ASCII que pode assumir os valores A, B, C, D, E, F, G, H, os quais devem ser interpretados pelo microcontrolador como os canais de 1 a 8 e consequentemente remetem a porta do controlador ao qual deve ser aplicada o sinal de calibração.

Foi desenvolvido, na sequência, uma parte do código para o ESP32, o qual foi programado para receber um caractere, e através de uma estrutura de comparação define-se qual canal deve ser ativado, chamando uma função responsável pela ativação das portas.

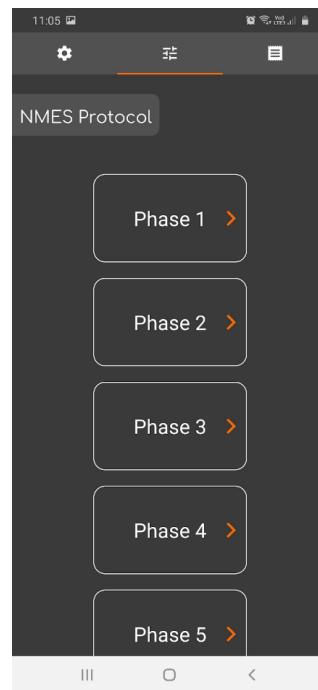
Nesse sentido criou-se uma função responsável por gerar o protocolo de estimulação, ou seja, uma função que recebe como entrada um valor de tempo e uma numeração de porta, gerando uma sequência de quatro pulsos retangulares de 100 us espaçados por um período de 100 us, repetindo esse padrão a uma frequência de 25 Hz na porta indicada, durante o tempo estipulado.

Para o caso de calibração de canais, o tempo foi mantido fixo e com valor de 15 segundos, valor esse encontrado após vários testes. É também interessante ressaltar que ao receber as mensagens, o controlador foi programado para as reenviar para o aplicativo onde sofre uma comparação, determinando se houve falhas na comunicação.

2.2.3 Sistema de ajuste do protocolo

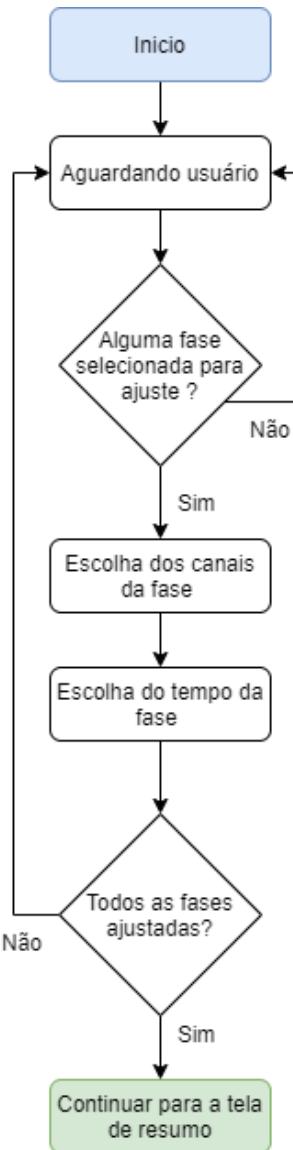
Na sequência, desenvolveu-se uma tela cuja finalidade é a de permitir o usuário realizar os ajustes necessários para uma sessão de EENM.

Figura 9 – Tela de ajuste de protocolo



Esse sistema foi construído utilizando um fragmento sobreposto a atividade principal, e para esclarecer o seu funcionamento, segue o seu diagrama:

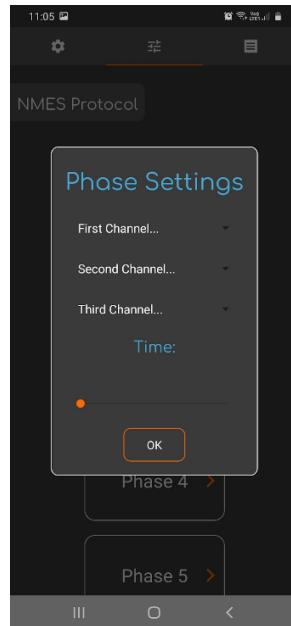
Figura 10 - Ajuste de receita



A tela é composta de 12 cards que possuem referência a qual das 12 fases pertence. Para realizar a configuração de uma fase, uma seta (em laranja no canto direito de cada card) foi implementada utilizando o conceito de “ImageButton”, ou seja, uma imagem que quando pressionada opera de forma semelhante a um botão.

Implementou-se, como resultado do aperto da imagem, uma janela do tipo “Dialog”, ou caixa de diálogo, onde as configurações para a dada fase podem ser efetuadas.

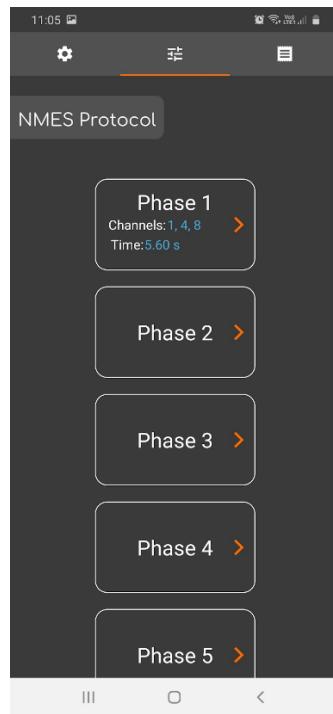
Figura 11 – Caixa de dialogo



Essa caixa de diálogo possui 3 elementos de seleção para a escolha dos canais a serem ativados, seguida de um elemento para a seleção do tempo que os dados canais devem permanecer ativados e um botão para salvar as alterações, que quando pressionado altera sua coloração afim de indicar o aperto.

Implementou-se como resultado no aperto do botão na caixa de diálogo, o fechamento da caixa de diálogo seguido do salvamento das informações configuradas, em um objeto da classe “messenger”, especialmente construída para armazenar tais valores, sendo que em seguida esse objeto é utilizado, dentre outras aplicações, para povoar o card na tela de ajuste de protocolo.

Figura 12 – Card povoado

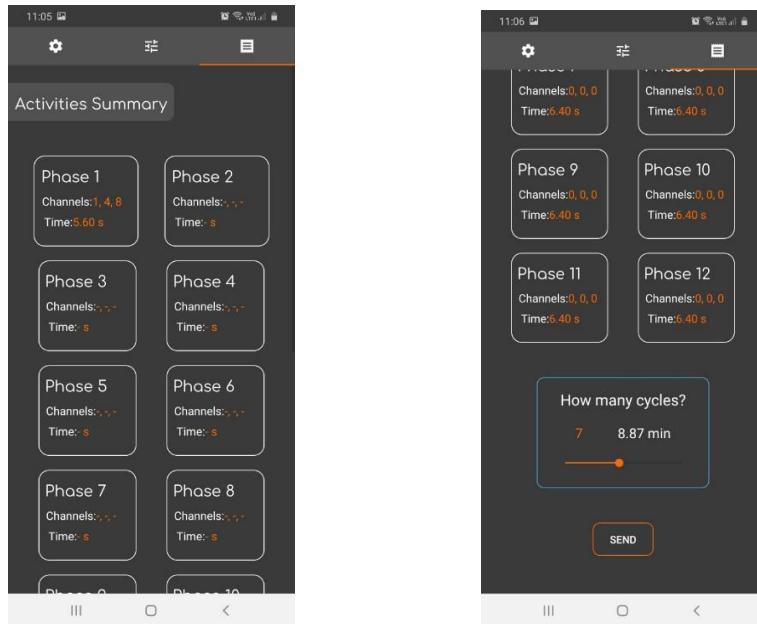


2.2.4 Sistema de envio e execução do protocolo

Na sequência, foi implementado um sistema que permite tanto a checagem visual do protocolo configurado quanto a execução do mesmo pelo controlador.

Iniciou-se com a implementação de uma tela de resumo (que contem também um elemento de ajuste de número de ciclos).

Figura 13 – Tela de resumo e envio



Desenvolveu-se essa parte do aplicativo com o intuito de apresentar todas as configurações realizadas e também para realizar a definição do número de ciclos (número de vezes que a receita deve ser executada).

Uma sequência de 12 cards que são povoados a partir dos objetos da classe “messenger” foram criados, e caso exista objetos não ajustados, o aplicativo foi programado para apresentar caracteres especiais de forma a alertar o usuário. Na sequência, foi implementada uma barra de seleção de tempo e também um botão para realizar o envio dos dados para o microcontrolador.

Ao pressionar o botão de envio, implementou-se um sistema de checagem pelo aplicativo, buscando por falhas de configuração do protocolo. Caso todos os valores estejam condizentes, o aplicativo inicia a preparação das mensagens a serem enviadas.

Primeiramente, todos os valores referentes às 12 fases, isso inclui tanto os três canais quanto seu tempo, são organizados sequencialmente na forma: (Canal 1 - Fase 1, Canal 2 - Fase 1, Canal 3 - Fase 1, Tempo - Fase 1, Canal 1 - Fase 2, Canal 2 - Fase 2, Canal 3 - Fase 2, Tempo - Fase 2 !), gerando um vetor contendo 97 caracteres.

Na sequência, prepara-se a mensagem de ciclo, a qual é composta do número de ciclos seguido do caractere “?”. E finalmente prepara-se a mensagem de sinalização de início a qual é composta de um único caractere, ou seja, o caractere “\$”.

Com todas as mensagens prontas para envio, implementou-se um sistema onde cada envio sofre uma checagem em busca de erros de transmissão, sendo que uma mensagem

somente é enviada quando a anterior foi enviada com sucesso, evitando falhas na comunicação e transmissão de dados.

Tendo a tela de resumo e envio pronta, adicionou-se funcionalidade ao firmware já implementado para o sistema de calibração de canais. Utilizando o sistema de comparação implementado, adicionou-se novas opções de comparação realizando dessa forma um sistema de triagem de dados cujo funcionamento está descrito na figura 6.

Dessa forma, ao receber um dado, primeiramente, o microcontrolador analisa o primeiro caractere da mensagem recebida. Se for constatado que se trata de um caractere do tipo ASCII (elemento específico), as rotinas de calibração dos canais serão ativadas de acordo com o caractere recebido ou então o processo de estimulação será iniciado (no caso da recepção do caractere “\$”).

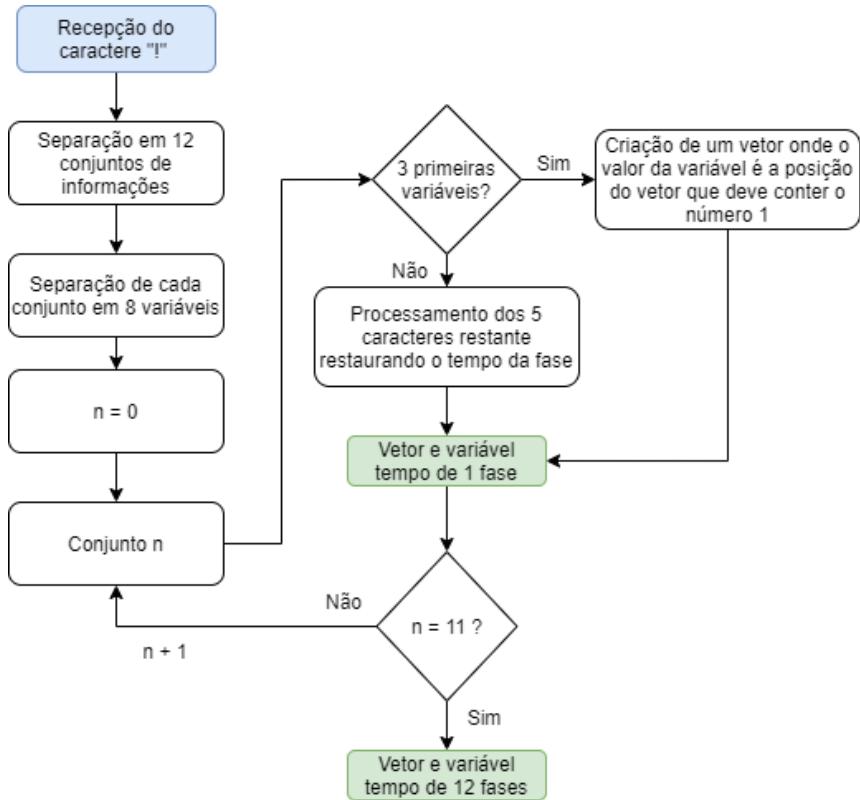
Na sequência, caso o primeiro caractere não seja do tipo ASCII esperado, o quarto caractere é analisado. Caso se trate de um caractere especial esperado, retira-se da mensagem o valor do número de ciclos de estimulação e o armazena em uma variável.

Caso o quarto caractere não seja do tipo ASCII, será analisado o último caractere em busca do tipo ASCII (também um caractere especial), e se esse for o caso, retira-se da mensagem o protocolo para a realização da estimulação.

2.2.4.1 Recepção do protocolo

Ao receber uma mensagem contendo um protocolo de estimulação, o microcontrolador foi programado para realiza o procedimento descrito no diagrama a seguir:

Figura 14 - Processamento do protocolo



Desta forma, a primeira etapa após a identificação do tipo da mensagem é o processo de extração das informações a respeito das 12 fases de estimulação, ou seja, todos os canais e tempos que devem ser estimulados em cada uma das 12 fases. Desenvolveu-se o firmware sabendo-se que no máximo 3 canais serão estimulados simultaneamente em uma fase, logo a mensagem recebida possui 3 números reais de 1 a 8 seguido de 5 números reais que representam o tempo que estes 3 canais devem ser estimulados dado em segundos.

Tendo extraído esses valores, os numerais que representam os canais a serem estimulados passam a representar posições de um vetor com 8 posições que devem ser preenchidos com o número 1 (o vetor inicialmente é um vetor de zeros). Em seguida os 4 números restantes, passam por um processamento para se reconstituir o tempo original.

Cada um desses vetores povoados com 1 são posteriormente processados, e a posição em que o número 1 estiver representará o número da porta do microcontrolador a ser estimulada.

Esse processo é executado para todos os 12 canais, de forma que ao final do processo, obtém-se 12 vetores e 12 valores de tempo, os quais sequencialmente compõem a receita para a execução da estimulação.

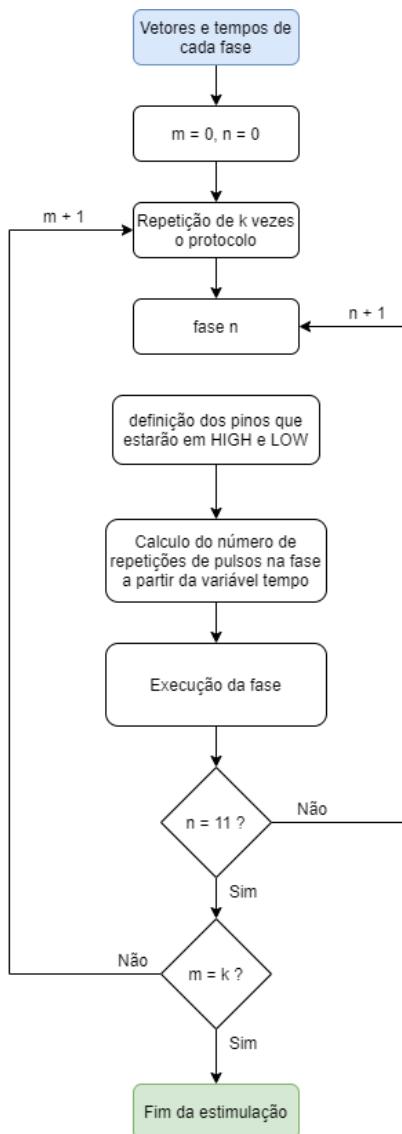
2.2.4.2 Recepção de ciclos

Ao receber uma mensagem contendo o número de ciclos, o microcontrolador foi programado para realizar o procedimento de extrair os valores da mensagem e preservá-lo em uma variável para posterior uso.

2.2.4.3 Recepção de indicador de início

Ao receber uma mensagem contendo o indicador de início, o microcontrolador foi programado para realizar a seguinte sequência de operações a fim de gerar a estimulação:

Figura 15 - Execução do protocolo



Dessa maneira o processo de estimulação se inicia com o envio dos dados de cada fase (vetor e tempo) de forma sequencial para a função que realizará a interpretação dos dados e executará a estimulação nas portas indicadas pelo vetor (função criada anteriormente para o sistema de calibração), pelo tempo indicado na variável “tempo” (as 12 fases são passadas para a função o número de vezes definido pela mensagem ciclos).

Dentro da função, o vetor é separado em 8 variáveis que são analisadas e se apresentarem o valor 1, a porta do microcontrolador correspondente receberá nível lógico alto no momento adequado e caso apresente o valor zero receberá nível lógico baixo (nível lógico alto representa uma saída de 3.3 V e nível lógico baixo representa uma saída de 0 V).

Tendo então definido para aquela fase quais portas serão estimuladas, o valor de tempo recebido é processado, onde calcula-se quantos conjuntos de 4 pulsos a 25 Hz podem ser realizados, de forma que se inicia a estimulação dos canais e realiza-se exatamente o número calculado.

2.3 Implementação da comunicação WiFi

A comunicação utilizando WiFi foi implementada utilizando como base os protocolos TCP/IP em conjunto com uma arquitetura cliente - servidor.

Dessa maneira, o microcontrolador foi programado para operar tanto em modo roteador quanto em modo servidor simultaneamente, de forma que a necessidade de um aparelho roteador para gerar uma rede interna de internet foi excluída.

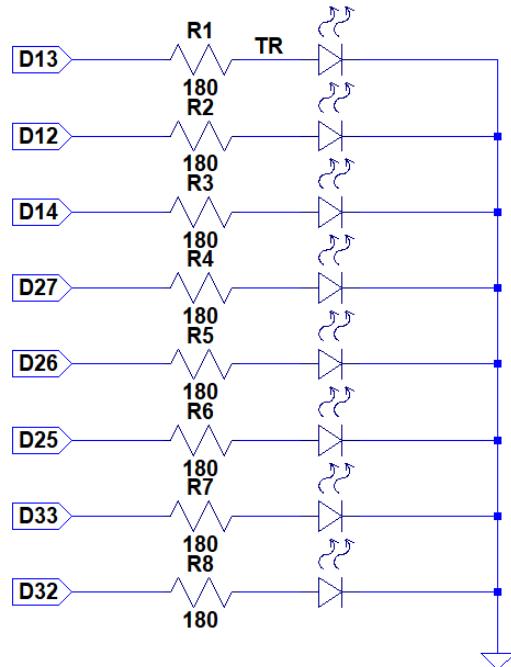
Para operar como roteador, utilizou-se uma função do microcontrolador chamada de “SoftAP”, ou “Soft Access Point” criando uma rede local de internet que é processada em uma das portas do microcontrolador (essa porta referida aqui, não é uma porta física do controlador e sim uma porta de entrada/saída de dados da internet). Para ter um maior controle sobre essa internet local, definiu-se ainda um endereço (IP) estático para a conexão (caso não seja definido dessa forma o controlador opera em modo dinâmico alocando um IP aleatório cada vez que a rede for criada) e por uma questão de segurança, definiu-se uma senha para o acesso à rede.

Ao mesmo tempo que o microcontrolador foi configurado para operar em modo roteador, em outra porta de dados (neste caso definida porta 80), o controlador foi configurado para operar como um servidor, ou seja, o microcontrolador opera na mesma rede criada por ele apenas observando portas de dados diferentes.

2.4 Testes

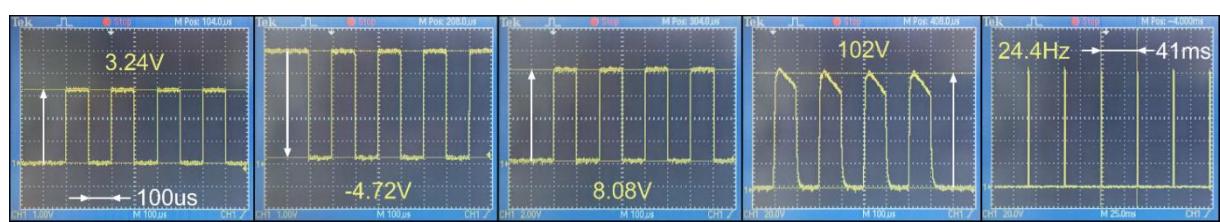
Para a realização dos testes no sistema, implementou-se em uma protoboard um circuito conforme o apresentado na figura 16.

Figura 16 - Circuito de teste



Dessa maneira, toda vez que as portas do microcontrolador são ativadas, os LEDs são acesos emulando assim as condições reais. Para a observação do formato da onda, utilizou-se um osciloscópio digital, com as pontas de prova posicionadas entre a porta do controlador e a referência, de forma a comprovar o funcionamento do sistema wireless de EENM. As ondas obtidas estão expressas a seguir.

Figura 17 - Ondas



3 CONCLUSÃO

Conforme apresentado durante este projeto, a utilização de um aplicativo em conjunto com um microcontrolador, é viável para a realização da eletroestimulação neuromuscular. Além disso, promove uma maior mobilidade ao paciente e também representa uma redução de custos relacionado a aparelhagem necessária.

O processo de desenvolvimento ressaltou também alguns pontos que podem ser melhorados, sendo o primeiro deles a substituição da comunicação WiFi pela comunicação Bluetooth, abrindo espaço para uma comunicação mais segura e estável. Na sequência, é interessante a implementação de um sistema de transmissão ao vivo do protocolo de estimulação, possibilitando a pausa do processo caso necessário. Finalmente, após diversos testes ficou claro que o esquema de cores e a forma que foi disposta as funcionalidades do aplicativo não constituem as ideais, podendo ser mais intuitivas para o usuário.

Tendo isso em mente, é possível concluir que o projeto desenvolvido se compõe como uma adição bem-vinda e vantajosa para o protótipo vestível, dando maior autonomia para o projeto, além de representar uma proposta inovadora para a execução de estimulação elétrica.

4 REFERÊNCIAS BIBLIOGRÁFICAS

ABRAFIN. Fisioterapia e Lesão Medular. Associação Brasileira de Fisioterapia Neurofuncional, 2015. Disponível em: https://abrafin.org.br/wp-content/uploads/2015/01/LESAO_MEDULAR.pdf, Acesso em 25/05/2020

Ventura, A.; Bataglia, J.; Machado, L.; Lopes da Silva, J.; Varoto, R. and Cliquet Jr., A. (2020). An Innovative Approach towards Incorporating the End User to the NMES Wearable System Development. In Proceedings of the 13th International Joint Conference on Biomedical Engineering Systems and Technologies - Volume 1: BIODEVICES, ISBN 978-989-758-398-8, pages 192-199. DOI: 10.5220/0009095401920199

MAGALHÃES, Lana. Coluna Vertebral. TodaMatéria, 2019. Disponível em: <https://www.todamateria.com.br/coluna-vertebral/>, Acesso em, 25/05/2020

Ministério da Saúde. Diretrizes de atenção a pessoa com lesão medular, 2013. Disponível em: http://bvsms.saude.gov.br/bvs/publicacoes/diretrizes_atencao_pessoa_lesao_medular.pdf, Acesso em 04/07/2020

APENDICE

Todos os códigos criados, tanto para o microcontrolador quanto para o aplicativo podem ser encontrados no link abaixo.

<https://github.com/JhonPB/TCC>