

CENTRO UNIVERSITÁRIO FEI
ANA BEATRIZ DUBUGRAS PISANI
GIOVANNI ALMEIDA MATOS
LUCAS SANTEJO DOS SANTOS

EYECOMM: Controle de mouse através de Eletrooculograma.

São Bernardo do Campo

2022

ANA BEATRIZ DUBUGRAS PISANI
GIOVANNI ALMEIDA MATOS
LUCAS SANTEJO DOS SANTOS

EYECOMM: Controle de mouse através de Eletrooculograma.

Trabalho de Conclusão de Curso
apresentado ao Centro Universitário FEI,
como parte dos requisitos necessários
para obtenção do título de Bacharel em
Engenharia Elétrica. Orientado pela Prof.
Dra. Maria Claudia Ferrari de Castro.

São Bernardo do Campo

2022

Dubugras Pisani, Ana Beatriz.

EYECOMM: Controle de mouse através de Eletrooculograma. / Ana Beatriz Dubugras Pisani, Giovanni Almeida Matos, Lucas Santejo dos Santos. São Bernardo do Campo, 2022.

70 p. : il.

Trabalho de Conclusão de Curso - Centro Universitário FEI.

Orientadora: Prof.^a Dra. Maria Claudia Ferrari de Castro.

1. Mouse Ocular. 2. Acessibilidade. 3. Esclerose Lateral Amiotrófica.
I. Almeida Matos, Giovanni. II. Santejo dos Santos, Lucas. III. Ferrari de Castro, Maria Claudia, orient. IV. Título.

ANA BEATRIZ DUBUGRAS PISANI
GIOVANNI ALMEIDA MATOS
LUCAS SANTEJO DOS SANTOS

EYECOMM: Controle de mouse através de Eletrooculograma.

Trabalho de Conclusão de Curso, apresentado
ao Centro Universitário FEI, como parte dos
requisitos necessários para obtenção do título
de Bacharel em Engenharia Elétrica.

Comissão julgadora

_____ Prof. Dra. Maria Claudia F. de Castro _____

Orientador e presidente

_____ Prof. Dr. Valter F. Avelino _____

Examinador (1)

_____ Prof. Dr. Fabio Gerab _____

Examinador (2)

São Bernardo do Campo

Dezembro 2022

AGRADECIMENTO

Às nossas famílias e amigos que nos incentivaram e acolheram nos momentos difíceis e estiveram ao nosso lado para comemorar cada vitória. Em especial, agradecemos ao nosso grande amigo Greg, reconhecido por nós como o quarto integrante do grupo, que com sua alegria contagiante tornou nossos dias na faculdade memoráveis, além de ser o designer oficial do *Eyecomm*.

A nossa orientadora, Dra. Maria Cláudia, pelos ensinamentos e apoio durante todo este ano de desenvolvimento, nos ajudando não somente na área técnica, mas também estando sempre presente ao grupo.

À toda equipe do Centro de Laboratórios Elétricos (CLE), especialmente ao Douglas, Vitão e Yuri, que além de nos auxiliarem durante toda a montagem do projeto, nos apoiaram e ajudaram nas tomadas de decisões em momentos de imprevistos e desafiadores.

RESUMO

A cada 90 minutos, uma pessoa é diagnosticada com Esclerose Lateral Amiotrófica (ELA). Essa doença é caracterizada pela perda progressiva do controle motor, levando a um quadro de paralisia irreversível, enquanto as funções cognitivas são preservadas. De forma semelhante, a Síndrome de Encarceramento retira a capacidade de movimento de todo o corpo, exceto dos olhos e pálpebras. Avanços na Engenharia Biomédica possibilitam o desenvolvimento de novas tecnologias que favorecem uma melhor qualidade de vida. Neste contexto, este projeto objetiva desenvolver um sistema de controle de cursor ocular para pessoas com níveis de paralisia severos através de sinais obtidos pelo eletrooculograma. A partir de pesquisas abrangendo diferentes soluções de acessibilidade para sistemas computacionais e juntamente a testes realizados no projeto *Eye Access*, foram levantadas possíveis melhorias e novas funcionalidades, como mais alternativas de clique, incluindo clique duplo, clique e arrasta, clique com o botão direito e clique scroll, e a opção de aumento de velocidade do cursor. Foram utilizados eletrodos posicionados na face do usuário para a aquisição dos sinais, amplificador de instrumentação junto ao uso de filtros analógicos para o tratamento e o microcontrolador para a análise e transcrição do movimento ocular para o movimento do cursor. Obteve-se um sinal similar ao apresentado na literatura, e a partir dele foram definidos limiares que foram utilizados para o reconhecimento do comando. Fundamentado a este método, alcançou-se uma acurácia global superior a 90%, validando este projeto.

Palavras-chave: Mouse Ocular; Acessibilidade; Esclerose Lateral Amiotrófica.

ABSTRACT

Every 90 minutes, a person is diagnosed with Amyotrophic Lateral Sclerosis (ALS). This disease is characterized by progressive motor control loss, leading to irreversible paralysis, while cognitive functions are preserved. Similarly, Locked-in Syndrome removes the ability to move from the entire body except for the eyes and eyelids. Advances in Biomedical Engineering allow the development of new technologies that favor a better quality of life. In this context, this project aims to develop an eye cursor control system, designed for people with severe paralysis levels through signals obtained by the electrooculogram. From research covering different accessibility solutions for computer systems and together with tests carried out in the Eye Access project, improvements and new functionalities were raised, such as more click alternatives, including double-click, click and drag, right-click, and scroll, and cursor speed increase option. Electrodes positioned on the user's face were used for the signals acquisition, an instrumentation amplifier together with the use of analog filters for the treatment and the microcontroller for the analysis and transcription of the eye movement for the movement of the cursor. A signal similar to that presented in the literature was obtained, and from its thresholds were defined that were used for command recognition. Based on this method, an overall accuracy greater than 90% was achieved, validating the project.

Keywords: Ocular Mouse; Accessibility; Amyotrophic Lateral Sclerosis.

LISTA DE ILUSTRAÇÕES

Figura 1 – Estrutura fisiológica do olho	13
Figura 2 – Posicionamento dos eletrodos no rosto	14
Figura 3 – Sinal de um movimento para direita dos olhos	15
Figura 4 – Circuito comum para um amplificador de instrumentação	16
Figura 5 – Filtro Passa-Baixas (A); Filtro Passa-Faixa (B); Filtro Passa-Altas (C); Filtro Rejeita-Faixa (D).....	17
Figura 6 – Esquemático <i>Tobii</i>	18
Figura 7 – Estrutura física Eye Access	20
Figura 8 – Teclado desenvolvido pelo <i>Eye Access</i>	20
Figura 9 – Diagrama de bloco <i>Eye Access</i>	21
Figura 10 – Interface gráfica desenvolvida por Lima <i>et al.</i> (2008).....	22
Figura 11 – Sinais de EOG original e filtrado (A); Menu de configuração (B); Interface de usuário (C)	23
Figura 12 – Diagrama de blocos do sistema.....	25
Figura 13 – Estrutura de LEDs e respectiva simbologia	26
Figura 14 – Topologia Sallen-key do filtro passa-baixas projetado.....	28
Figura 15 – Diagrama de Bode do filtro passa-baixas Butterworth de segunda ordem	28
Figura 16 – Topologia Sallen-key do filtro passa-altas projetado.....	29
Figura 17 – Diagrama de Bode filtro passa-altas Butterworth de segunda ordem.....	29
Figura 18 – Sinal de EOG da movimentação dos olhos para cima com limiares.....	30
Figura 19 – Fluxograma do sistema. PRINCIPAL (A); Sub-rotina VALIDA UP (B); Sub-rotina VALIDA DOWN (C); Sub-rotina MOVE_UP (D); Sub-rotina BLINK (E)	32
Figura 20 – Esquemático <i>Eyecom</i>	35
Figura 21 – Máscara da placa de circuito impresso.....	36
Figura 22 – Placa de Circuito Impresso frente e verso.....	37
Figura 23 – Placa com o circuito completo	38
Figura 24 – Sinal obtido pelo canal vertical de movimento para cima	40
Figura 25 – Sinal obtido pelo canal vertical de movimento para baixo	40
Figura 26 – Sinal obtido pelo canal vertical de piscada	41
Figura 27 – Sinal obtido pelo canal vertical de dupla piscada	41
Figura 28 – Sinal obtido pelo canal horizontal de movimento para direita.....	42
Figura 29 – Sinal obtido pelo canal horizontal de movimento para esquerda.....	42

Figura 30 – IDE do Arduino contendo o código parcial43

LISTA DE TABELAS

Tabela 1 – Comparativo dos trabalhos apresentados.....	24
Tabela 2 – Modelo de Matriz de Confusão	34
Tabela 3 – Tabela descritiva de custos do projeto.....	39
Tabela 4 – Matriz de Confusão Usuário 01- Primeira sessão.....	44
Tabela 5 – Matriz de Confusão Usuário 01- Segunda sessão	44
Tabela 6 – Matriz de Confusão Usuário 01- Terceira sessão	45
Tabela 7 – Somatório das Matrizes de Confusão – Primeira sessão	45
Tabela 8 – Somatório das Matrizes de Confusão – Segunda sessão	45
Tabela 9 – Somatório das Matrizes de Confusão – Terceira sessão.....	46
Tabela 10 – Acurácia total a partir do conjunto de testes.....	46
Tabela 11 – Comparativo do Eyecomm com os trabalhos apresentados	49

LISTA DE SIGLAS

ADC	Analogic-Digital Converter – Conversor Analógico-Digital
CI	Circuito Integrado
CMRR	Common-Mode Rejection Rate – Razão de rejeição de modo comum
DC	Direct Current – Corrente Contínua
ELA	Esclerose Lateral Amiotrófica
EOG	Eletrooculograma
FFT	Fast Fourier Transform – Transformada Rápida de Fourier
IDE	Integrated Development Environment – Ambiente de Desenvolvimento Integrado
LED	Light Emitting Diode – Diodo Emissor de Luz
LIS	Locked-in Syndrome – Síndrome do Encarceramento
MLP	Multilayer Perceptron – Rede Neural Multicamadas
PCB	Printed Circuit Board – Placa de Circuito Impresso
SVM	Support Vector Machine - Máquina de Vetores de Suporte

SUMÁRIO

1	INTRODUÇÃO	11
1.1	OBJETIVOS	12
2	FUNDAMENTAÇÃO TEÓRICA	13
2.1	FISIOLOGIA OCULAR	13
2.1.1	ELETROOCULOGRAMA	14
2.2	AMPLIFICADOR DE INSTRUMENTAÇÃO	15
2.3	FILTROS	16
3	REVISÃO BIBLIOGRÁFICA	18
3.1	EYE TRACKING	18
3.2	PRODUTOS POR ELETROOCULOGRAMA	19
3.2.1	EYE ACCESS	19
3.2.2	OUTROS TRABALHOS	21
4	METODOLOGIA	25
4.1	VISÃO GERAL DO SISTEMA	25
4.1.1	Projeto de Hardware	26
4.1.1.1	<i>Amplificador de instrumentação</i>	26
4.1.1.2	<i>Filtros</i>	27
4.1.1.3	<i>Microcontrolador</i>	29
4.1.2	Fluxograma do firmware	30
4.1.3	Experimentos	34
5	RESULTADOS	35
6	DISCUSSÕES	47
7	CONCLUSÃO	50
8	PERSPECTIVAS FUTURAS E SUGESTÕES DE MELHORIAS	51
	REFERÊNCIAS	52
	APÊNDICE A – FLUXOGRAMA DO FIRMWARE	55
	APÊNDICE B – ESQUEMÁTICO EYECOMM	57
	APÊNDICE C – CÓDIGO DO FIRMWARE	58
	APÊNDICE D – MATRIZ DE CONFUSÃO POR USUÁRIO TESTE	63

APÊNDICE E – MANUAL DO USUÁRIO	68
---	-----------

1 INTRODUÇÃO

A Esclerose Lateral Amiotrófica, também conhecida como ELA, é uma das principais doenças neurodegenerativas, junto ao Parkinson e Alzheimer [1] e, apesar de ser considerada rara, afeta mais de 5000 pessoas por ano [2], em sua maioria de 40 a 70 anos [3]. É caracterizada pela degeneração progressiva dos neurônios motores (células do sistema nervoso central que controlam os movimentos voluntários) do cérebro, da medula espinal e do tronco cerebral, levando à fraqueza muscular global. A perda progressiva do controle motor limita a capacidade de movimento, levando à paralisia motora irreversível do indivíduo, enquanto as funções sensitivas e cognitivas são preservadas durante todo o curso da doença. Considerando que para muitas pessoas o músculo responsável pelo controle dos olhos não é afetado com a progressão da doença [4], torna-se possível que esse meio seja utilizado para a comunicação.

De forma similar, a Síndrome de Encarceramento, ou *Locked-in Syndrome* (LIS), é um estado de tetraplegia junto à paralisia da parte inferior do crânio [5], retirando a capacidade de todos os movimentos, exceto olhos e pálpebras [6], além do indivíduo ainda manter a sua memória e consciência sobre o todo [5, 6]. Geralmente, os pacientes conseguem somente movimentar os olhos para cima e para baixo, pois até mesmo os movimentos laterais dos olhos ficam comprometidos [6].

Mesmo com doenças tão complexas, a Engenharia Biomédica tem ganhado notoriedade devido ao desenvolvimento de novas tecnologias assistivas, impactando na saúde e qualidade de vida de pessoas com diferentes doenças. Essas novas tecnologias partem desde a expansão da ciência biológica molecular e celular, ciência dos materiais até estratégias interdisciplinares para a solução de problemas complexos [7]. O envelhecimento da população também tem colaborado com a aceleração do desenvolvimento de tecnologias assistivas devido a necessidade deste público em manter a qualidade de vida apesar da deterioração funcional que ocorre em função da idade [7].

Alinhado a este contexto, este projeto objetiva melhorar a qualidade de vida e acessibilidade digital e, principalmente, de comunicação de indivíduos com um nível de paralisia severa, como a ELA e a LIS.

1.1 OBJETIVOS

O projeto visa controlar um cursor de mouse por meio do movimento dos olhos obtido através do sinal de Eletrooculograma (EOG), com velocidade ajustável, agregando as seguintes funcionalidades: clique com a esquerda, clique secundário (botão da direita), *scroll* (botão do meio), clique duplo e clique e arrasta.

Para tal, foram necessários: a obtenção dos sinais de EOG nas componentes vertical e horizontal através de eletrodos; desenvolvimento de um filtro capaz de eliminar os ruídos e artefatos presentes no sinal, após sua amplificação, para que possa ser interpretado via microcontrolador; desenvolvimento de um algoritmo para tradução do sinal do EOG em movimento, bem como discernir o movimento de uma ou duas piscadas. Uma vez processado, este movimento é enviado por USB a um computador ou Tablet, de modo que seja reconhecido como um mouse e reproduza as ações processadas pelo microcontrolador.

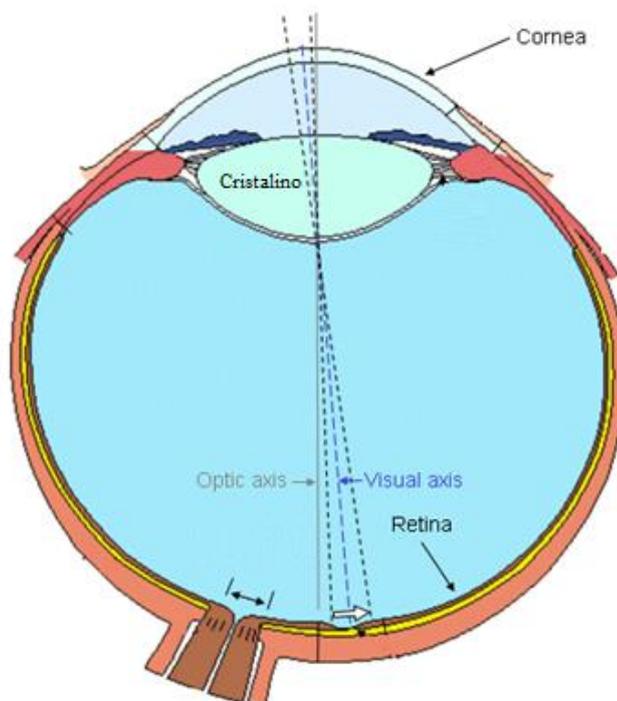
2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são abordados os conceitos fundamentais para o desenvolvimento técnico deste trabalho.

2.1 FISILOGIA OCULAR

O olho humano é o órgão capaz de transformar informação luminosa em sinais elétricos, que são enviados ao cérebro no córtex visual [8], capacitando o sentido da visão. Dividido em regiões que realizam diferentes funções no processo de visão, pode-se citar a córnea, o cristalino e a retina. A córnea é a primeira parte a entrar em contato com a luz; que passará pelo cristalino que é a lente que realiza a refração para o ajuste de foco. A retina, por sua vez, é o transdutor biológico que exerce a principal função de conversão da luz incidente em sinal elétrico que será transmitido pelo nervo óptico, sendo composta por uma grande quantidade fotorreceptores [8].

Figura 1 – Estrutura fisiológica do olho



Fonte: autor “adaptado de” [8]

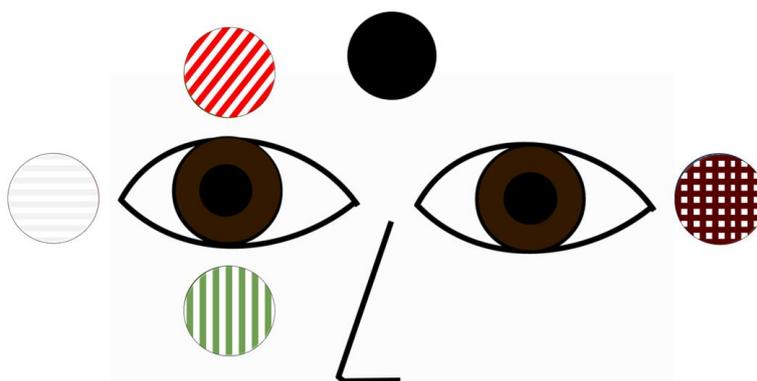
2.1.1 ELETROOCULOGRAMA

O EOG é um sinal biológico produzido pela diferença de potencial entre a retina e a córnea ocular [9], diferente da maioria dos outros sinais que são originados por tecidos excitados [8]. Este biopotencial é causado pela grande presença de nervos ativos na retina quando comparados com a região frontal dos olhos [7, 9], o qual pode ser modelado como um dipolo elétrico estático que se movimenta junto ao deslocamento ocular [7].

Em razão deste fenômeno, o campo elétrico gerado é linearmente relacionado ao deslocamento angular do olho [9], exceto quando ocorre a movimentação do piscar dos olhos [10, 12], mensurável através de eletrodos posicionados na região do rosto, independente das condições de luz do meio externo, dado que o sinal sofre uma variação insignificativa em função da luminosidade [8].

A Figura 2 mostra o posicionamento dos eletrodos para captação do EOG. Os eletrodos referenciados em marrom (hachura quadriculada) e branco (hachura listras horizontais) extraem o sinal do movimento ocular no eixo horizontal, enquanto os em vermelho (hachura listras diagonais) e verde (hachura listras verticais), no eixo vertical. Em preto, encontra-se o eletrodo utilizado para equalizar a referência dos biopotenciais, bem como a realimentação do sinal de modo comum invertido, contribuindo para minimizar o ruído induzido da rede (60 Hz).

Figura 2 – Posicionamento dos eletrodos no rosto

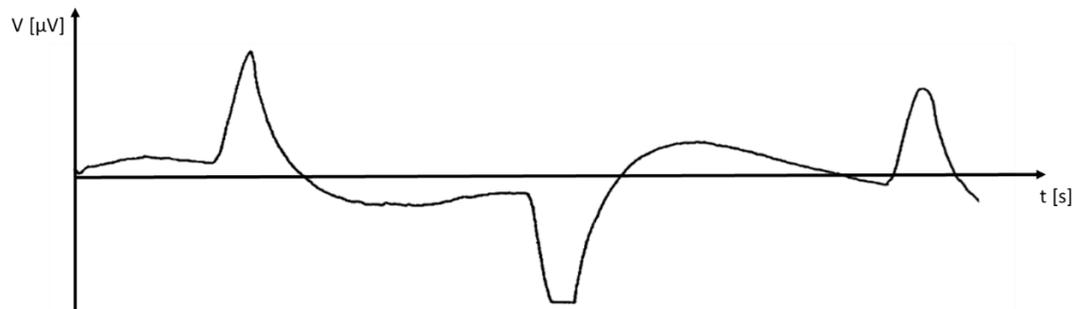


Fonte: Autor

É esperado que a leitura dos valores a partir do EOG tenha uma faixa de tensão de 15 μV até 200 μV e uma banda de frequência do nível DC até 10 Hz [10]. Uma vez que os sinais possuem baixa amplitude, é importante ter cautela com artefatos, como os provenientes da movimentação da pele [11, 12]. Um sinal EOG típico do movimento horizontal (esquerda para

direita) é mostrado na Figura 3, podendo ser observado a relação do movimento ocular para tensão, onde os picos representam variações da posição do olho.

Figura 3 – Sinal de um movimento para direita dos olhos



Fonte: Autor “adaptado de” [12]

Devido ao fato de que o sinal de EOG é caracterizado pela tensão diferencial entre os eletrodos e que, conseqüentemente, o sinal de modo comum é indesejado, há a necessidade do uso de amplificador diferencial como estágio de entrada.

2.2 AMPLIFICADOR DE INSTRUMENTAÇÃO

O amplificador de diferenças idealmente conta com uma entrada de dois sinais e obtém uma saída proporcional a diferença entre eles [13]. Entretanto, ainda permanece um pequeno ganho de modo comum na saída, podendo ser descrito pela Equação (1).

$$v_O = A_d v_{Id} + A_{cm} v_{Icm} \quad (1)$$

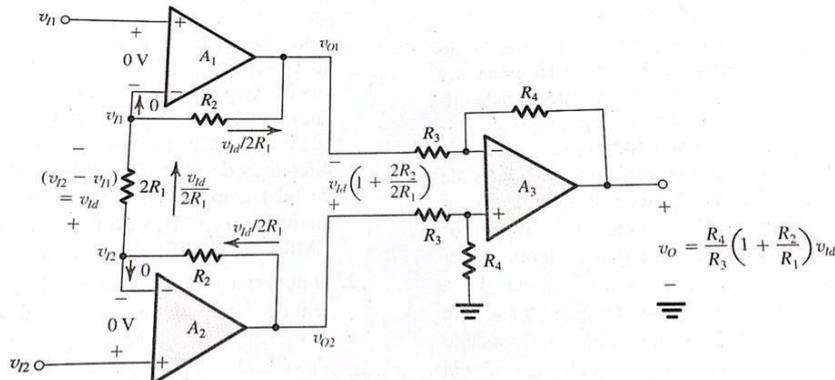
Onde v_o é a tensão de saída, A_d é o ganho diferencial, A_{cm} o ganho de modo comum, v_{Id} o sinal de entrada diferencial e o v_{Icm} a tensão de entrada de modo comum.

Um dos parâmetros que mede a eficiência de um amplificador diferencial é a razão de rejeição do sinal de modo comum (CMRR), dada pela Equação (2).

$$CMRR = 20 \log \frac{|A_d|}{|A_{cm}|} \quad (2)$$

O circuito amplificador diferencial utilizado neste trabalho é o amplificador de instrumentação (ilustrado na Figura 4), dado que apresenta maior impedância de entrada, um alto CMRR e um ganho facilmente ajustável em comparação com um amplificador diferencial simples [13].

Figura 4 – Circuito comum para um amplificador de instrumentação

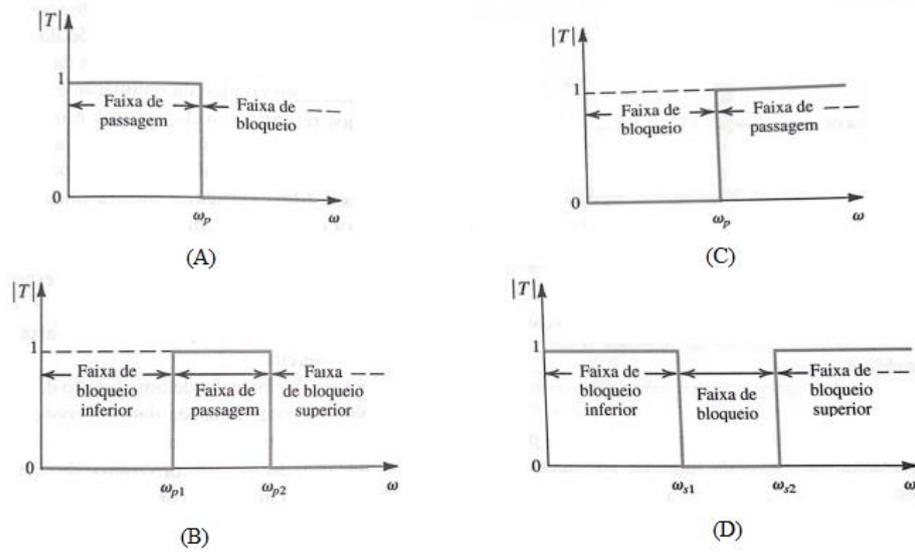


Fonte: [13]

2.3 FILTROS

Um filtro pode ser definido como um quadripolo capaz de permitir a passagem de frequências específicas do espectro do sinal de entrada, atenuando as demais [13], sendo o limiar dos estados a frequência de corte. Os filtros são subdivididos em quatro modos: filtro passa-baixas, passa-altas, passa-faixa e rejeita-faixa, sendo elas dependentes das frequências não atenuadas, também chamadas de faixa de passagem, como ilustra a Figura 5.

Figura 5 – Filtro Passa-Baixas (A); Filtro Passa-Faixa (B); Filtro Passa-Altas (C); Filtro Rejeita-Faixa (D)



Fonte: [13]

3 REVISÃO BIBLIOGRÁFICA

Existem duas vertentes de desenvolvimento para cursores controlados pela movimentação ocular: *Eye Tracking*, por meio de câmera, e o sistema por intermédio do EOG. O *Eye Tracking* é um rastreador ocular que possibilita o registro dos movimentos oculares [15], o qual, junto ao uso do *software*, permite que os dados sejam processados e utilizados para controle do cursor.

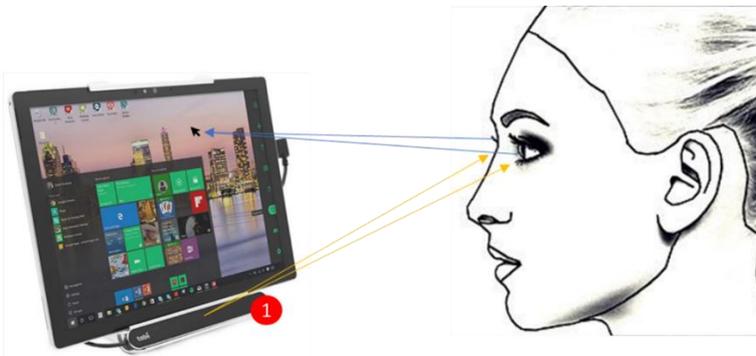
As soluções por meio de EOG, atualmente, estão concentradas na área da pesquisa. Esse tipo de solução conta com a orientação do cursor a partir dos sinais elétricos gerados pelo movimento dos olhos, conforme apresentado no Tópico 2.1.

A captação por meio de eletrodos é mais suscetível a interferências provenientes da rede elétrica de 60 Hz, podendo gerar um desconforto ao usuário devido a presença de cabos próximos ao seu rosto e o cansaço pode interferir no sinal do EOG. Entretanto, tem a vantagem de ser fracamente dependente da luminosidade do local [8] e da condição do usuário, permitindo o uso em locais escuros e/ou para usuários com acessórios e óculos [16].

3.1 EYE TRACKING

A tecnologia de rastreamento ocular é a solução presente no mercado, como os produtos desenvolvidos pela *Tobii* e *MyGaze*. Entre as variedades de produtos oferecidos por ambas as marcas, o responsável pelo rastreamento dos olhos é um equipamento em forma de barra, que acoplado a dispositivos, torna-se o responsável por executar e traduzir o movimento dos olhos para o movimento do cursor, conforme o modelo da *Tobii* ilustrado na Figura 6.

Figura 6 – Esquemático *Tobii*



Fonte: Autor, “adaptado de” [17]

A tecnologia apresenta fatores positivos quanto ao conforto para o usuário, além da agilidade de movimentação do mouse promovida pelo sistema de rastreamento. Por outro lado, essa tecnologia conta com um alto preço, na faixa de R\$ 2.900,00 à R\$ 15.000,00 (verificados em maio de 2022) [18, 19], contando com a variação de *softwares* inclusos e acessórios de *hardware*. Esse fato limita a acessibilidade apenas para um grupo específico de usuários. Além do outro fator desfavorável a esta solução, já mencionado anteriormente, a sensibilidade a interferência luminosa do meio e/ou o uso de óculos ou lentes de contato [16].

3.2 PRODUTOS POR ELETROOCULOGRAMA

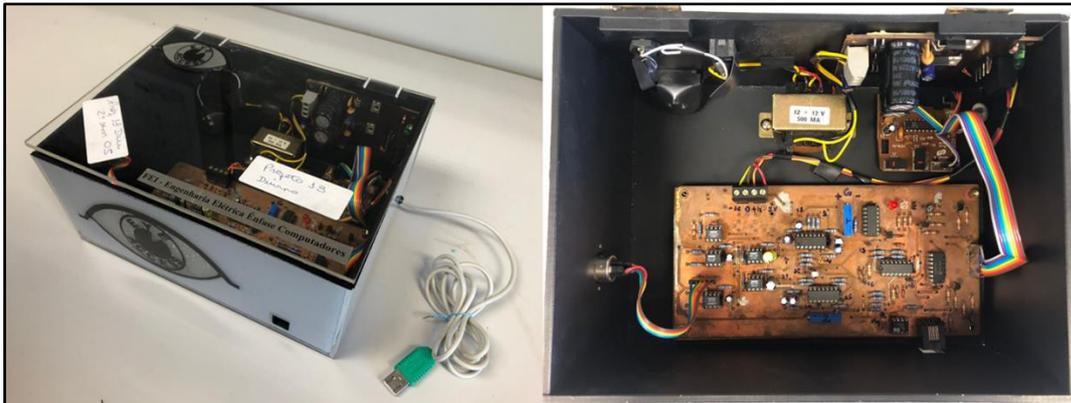
Nesse item são tratados os diferentes dispositivos que utilizam EOG como fonte de informação, visando o controle de mouse. Os dispositivos apresentados partiram de pesquisas realizadas em artigos científicos, com exceção do *Eye Access*, ao qual também foi fornecido o acesso, possibilitando a realização de testes de usuário.

3.2.1 EYE ACCESS

O *Eye Access* [20], desenvolvido no Centro Universitário FEI, obtém o sinal de EOG, o filtra, faz o reconhecimento do movimento do olho e o traduz em movimento de mouse, enviando ao computador um sinal análogo ao de um mouse comum.

O dispositivo é composto por uma caixa (24x18x10 cm), conforme Figura 7, com três cabos de saída: o cabo de alimentação, o conector USB e o cabo de EOG. Durante a experiência de uso, foi notado que o tamanho do dispositivo não é muito portátil e que o cabo de EOG utilizado era pesado para ser conectado aos eletrodos no rosto, trazendo uma sensação de desconforto.

Figura 7 – Estrutura física Eye Access

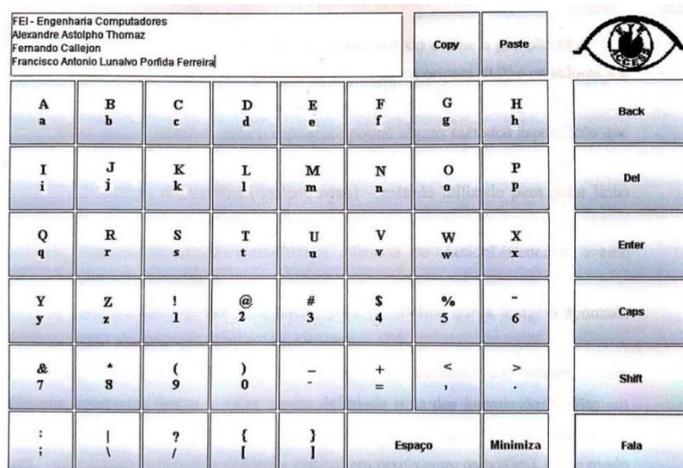


Fonte: Autor

Durante o período de testes foram observados alguns obstáculos quanto ao controle do cursor. Inicialmente, a curva de aprendizagem de uso se mostrou lenta, pois o sistema respondia apenas a uma velocidade de movimento ocular específica, além de apresentar uma alta taxa de erro. Outra dificuldade foi em relação a funcionalidade de clique, que a princípio seria acionada a partir da piscada dos olhos, mas que com os testes de uso se provou mais efetivo quando se realizava uma movimentação da esquerda para a direita.

Em paralelo ao *hardware*, é fornecido um *software* de um teclado virtual adaptado (Figura 8) de forma a otimizar a digitação do usuário, além de fornecer funções especiais, tais como: copiar, colar, *delete*, *capslock* e *shift*, juntamente com um recurso de reprodução do texto que foi escrito para pessoas cuja capacidade de fala foi comprometida. Contudo, não houve acesso ao teclado virtual, e, portanto, não foram feitos testes do seu funcionamento. Por outro lado, atualmente, existem vários aplicativos disponíveis com as mesmas funcionalidades.

Figura 8 – Teclado desenvolvido pelo Eye Access

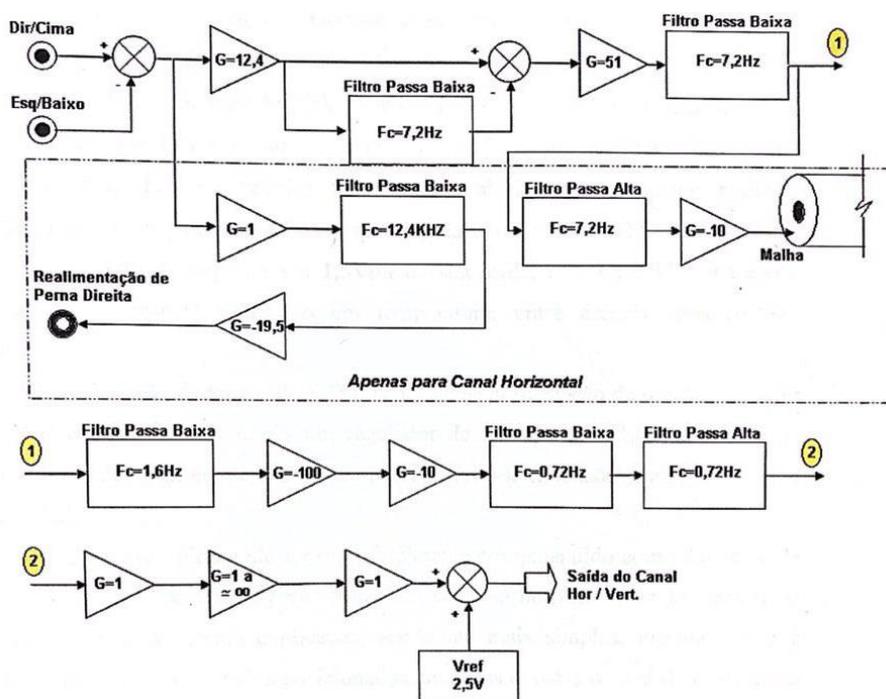


Fonte: [20]

Para o condicionamento do sinal, o projeto conta com um sistema de amplificação e filtragem, conforme o diagrama de blocos ilustrado na Figura 9. É notável a existência de grande quantidade de filtros utilizados que podem ser suprimidos, sem perda de desempenho, tornando o circuito elétrico mais compacto, mais simples e, conseqüentemente, mais barato. Essa melhoria foi levantada devido as pesquisas de faixa de frequência do EOG, que apresentaram do nível DC à 10 Hz. Com esses dados, é possível a implementação de apenas 4 filtros, sendo um passa-baixas e um passa-altas para cada orientação.

A partir da análise da arquitetura, em conjunto com as frequências de corte utilizadas nos filtros, é possível justificar as dificuldades de uso mencionadas anteriormente, como a velocidade específica necessária para o movimento do cursor durante o período de testes, a alta taxa de erro e a demora para a adaptação.

Figura 9 – Diagrama de bloco *Eye Access*



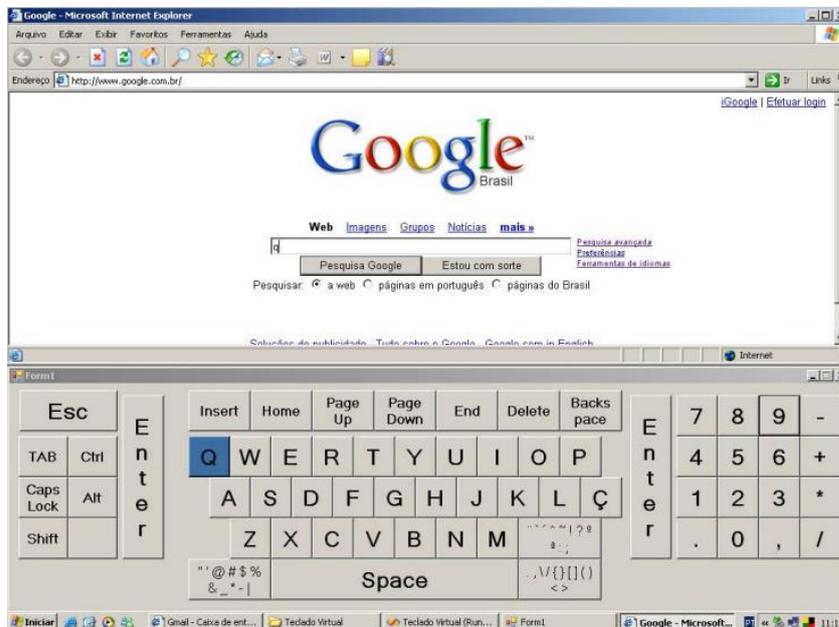
Fonte: [20]

3.2.2 OUTROS TRABALHOS

Um trabalho semelhante foi desenvolvido por Lima *et al.* (2008) cujos diferenciais foram a alimentação por bateria para a redução de ruído da rede elétrica de 60 Hz, um algoritmo de calibração e um *software* [21]. O *software* desta solução conta com uma interface de teclado

(ilustrada na Figura 10) que possibilita atalhos para digitação por meio de sugestão de texto e acesso a diversas páginas do computador.

Figura 10 – Interface gráfica desenvolvida por Lima *et al.* (2008)



Fonte: [21]

Outros trabalhos mais recentes utilizam diferentes tecnologias para o controle do cursor de mouse via EOG. Dentre eles, o desenvolvido por A. López *et al.* (2014) [22], por Zakir Hossain *et al.* (2017) [23], Ahsan-ul Kabir *et al.* (2020) [24].

O trabalho produzido por A. López *et al.*, Espanha, em 2014, desenvolveu uma plataforma que utiliza o *LabVIEW* para a escolha de algoritmos de processamento e filtragem de sinal, sendo eles *Laplace*, *FFT* e Transformada *Wavelet*, Figura 11. O circuito de *hardware* construído é semelhante aos outros trabalhos anteriormente mencionados, realizando a amplificação do sinal e um filtro passa-banda de 0.05 Hz a 34 Hz. Um ponto a ser destacado é que o projeto conta com uma alimentação de baterias que são recarregáveis por USB [22].

Figura 11 – Sinais de EOG original e filtrado (A); Menu de configuração (B); Interface de usuário (C)



Fonte: [22]

Desenvolvido em 2017 por Zakir Hossain *et al.*, Bangladesh, o projeto é capaz de processar o sinal de EOG e utilizá-lo de forma a controlar mecanismos além do cursor do mouse de computador, como braço robótico, cadeira de rodas e robôs móveis [23]. Elaborado em *Python*, o algoritmo utiliza técnicas de aprendizagem de máquina para a otimização da classificação do movimento, tanto no eixo horizontal como no vertical. O *hardware* é constituído por filtros passa-banda com frequências de corte de 0,3 Hz a 15 Hz e parte de uma alimentação por baterias simples não recarregáveis [23].

O trabalho mais recente, projetado por Ahsan-ul Kabir *et al.* (2020), se distingue dos demais por utilizar comunicação *wireless* entre o módulo de aquisição de EOG e o de processamento. O *hardware* trata o sinal com filtros passa-banda de frequências de corte de 0,67 Hz a 33,33 Hz, enviando-o a um Arduino para a digitalização e uma nova etapa de filtragem, porém digital [24]. O segundo Arduino recebe esse sinal e envia ao computador, que o analisa pelo *MATLAB* utilizando Máquina de Suporte Vetorial (SVM) e uma rede neural multicamadas (*Multilayer Perceptron* - MLP).

Em suma, com a finalidade de comparar todos os projetos estudados foi montada a Tabela 1. Nela verifica-se que os projetos mais recentes não contam com teclado virtual devido à grande oferta de aplicativos para esse fim, e passaram a utilizar inteligência artificial para tomadas de decisões. A maioria utiliza bateria como fonte de alimentação, possivelmente por questões de segurança, tendo em vista que, se conectado à rede elétrica, seria necessário

um estágio de isolamento para evitar corrente de fuga pelo corpo [25]. Por unanimidade todos possuem apenas a funcionalidade de clique simples. Destaca-se o projeto realizado por Ahsan-ul Kabir *et al.* pela sua função *wireless*, fornecendo uma melhor experiência de usuário.

Tabela 1 – Comparativo dos trabalhos apresentados

	Ahsan-ul Kabir <i>et al.</i> (2020)	Zakir Hossain <i>et al.</i> (2017)	A. López <i>et al.</i> (2014)	Lima <i>et al.</i> (2008)	Thomas <i>et al.</i> (2005)
Teclado virtual	Não	Não	Não	Sim	Sim
Alimentação	Bateria simples	Bateria simples	Bateria recarregável	Bateria simples	Rede
Funcionalidades de clique	Clique simples	Clique simples	Clique simples	Clique simples	Clique simples
Algoritmo	Inteligência Artificial	Inteligência Artificial	Simples	Simples	Simples
Wireless	Sim	Não	Não	Não	Não

Fonte: Autor

4 METODOLOGIA

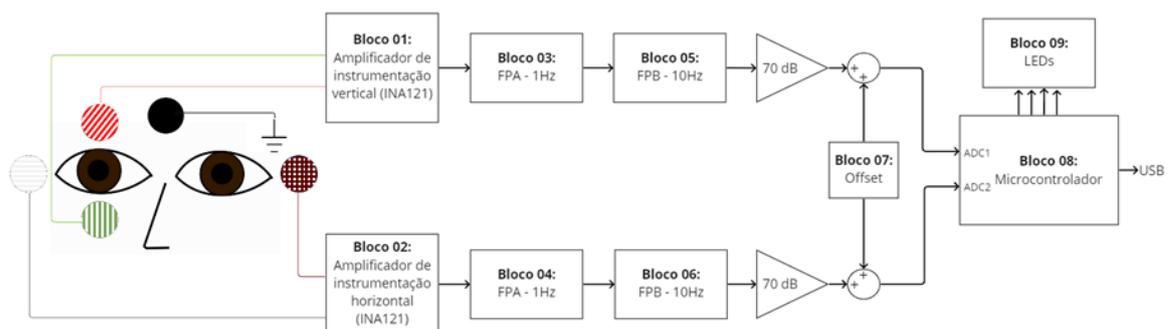
Neste capítulo é descrito o detalhamento da proposta do projeto, contando com a apresentação do diagrama de blocos, fluxograma e esquema elétrico do circuito.

4.1 VISÃO GERAL DO SISTEMA

A partir de pesquisas na literatura e de estudos realizados em dispositivos similares existentes, foi possível definir o *hardware* do projeto, o qual conta com amplificadores de instrumentação, filtros passa-baixas e passa-altas, amplificador operacional, microcontrolador e baterias. Em cima do *hardware* implementado, foi desenvolvido um sistema que abrange diferentes funções de cliques além da movimentação do cursor.

A Figura 12 apresenta o diagrama de blocos do sistema desenvolvido. O circuito foi dividido em 09 partes, sendo que os blocos 01, 03 e 05 correspondem respectivamente aos blocos 02, 04 e 06, mudando apenas os sinais de entrada gerados pela movimentação dos olhos do usuário, que, como mencionado no Capítulo 2.1.1, consideram, estrategicamente, uma análise separada das componentes horizontal e vertical do movimento.

Figura 12 – Diagrama de blocos do sistema



Fonte: Autor

Inicialmente, o biopotencial é extraído através dos amplificadores de instrumentação, conforme os blocos 01 e 02, dado que a componente relevante do sinal é o diferencial entre os dois eletrodos de entrada.

Conforme abordado no Capítulo 2.1.1, a faixa de frequência do EOG abrange desde o nível DC até 10 Hz. Considerando que a posição estática do olho, análoga ao nível DC, não é relevante ao projeto, fez-se o uso de um filtro passa-altas, cuja frequência de corte é de 1 Hz

(blocos 04 e 05) e, para a limitação da frequência máxima, foi utilizado um filtro passa-baixas com frequência de corte de 10 Hz (blocos 06 e 07).

Com o intuito de ajustar o sinal para a entrada do microcontrolador, foi necessário aumentar a amplitude até uma ordem de grandeza próxima a referência do conversor analógico-digital e adicionar um *offset*, devido ao fato do conversor analógico digital (ADC) não trabalhar com tensões negativas (bloco 08).

O microcontrolador é o responsável pelo processamento dos sinais, reconhecimento do movimento e geração do movimento do cursor. A outra função do microcontrolador é alternar entre os modos de operação de clique, mostrando o modo selecionado a partir dos LEDs, como demonstrado na Figura 13.

Figura 13 – Estrutura de LEDs e respectiva simbologia



Fonte: Autor

4.1.1 Projeto de Hardware

Esse item conta com a apresentação do projeto de *hardware* incluindo os principais componentes apresentados no diagrama da Figura 12.

4.1.1.1 Amplificador de instrumentação

O amplificador utilizado para o projeto é o INA118, o qual possui um CMRR de 110 dB e uma tensão máxima de *offset* de 50 μ V. Para alcançar o ganho de 60 dB descrito no

diagrama de blocos, foram distribuídos 30 dB para o amplificador de instrumentação e o restante para os filtros. Para o cálculo teórico do resistor de ganho R_g do INA118, foi utilizada a Equação (3.1), fornecida pela *Texas Instruments* [26], onde G representa o ganho em V/V e o R_g é o valor da resistência em Ohms.

$$G = 1 + \frac{50 \text{ k}\Omega}{R_g} \quad (3.1)$$

$$30 = 20 \log_{10}(G) \quad (3.2)$$

$$G = 31,62 \frac{\text{V}}{\text{V}} \quad (3.3)$$

$$R_g = 1,58 \text{ k}\Omega \quad (3.4)$$

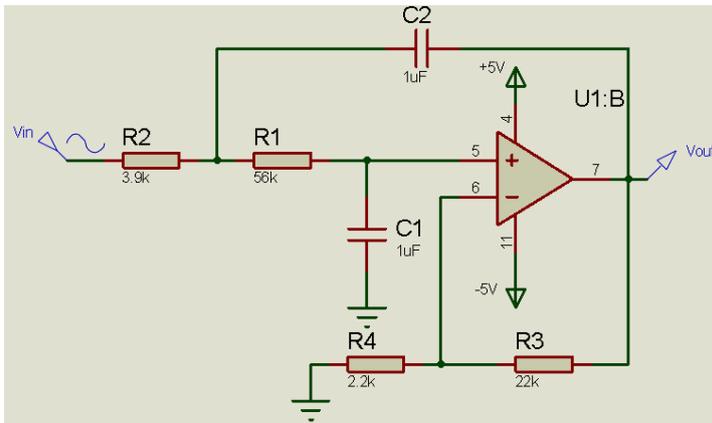
Considerando que o valor teórico calculado para R_g não é comercial, considera-se substituir por 1,5 k Ω .

4.1.1.2 Filtros

O projeto dos filtros desejados foi realizado com o uso da ferramenta de design fornecida pela *Texas Instrument*, a partir da configuração de um filtro Butterworth de segunda ordem, na topologia Sallen-key, devido a sua linearidade na faixa de passagem.

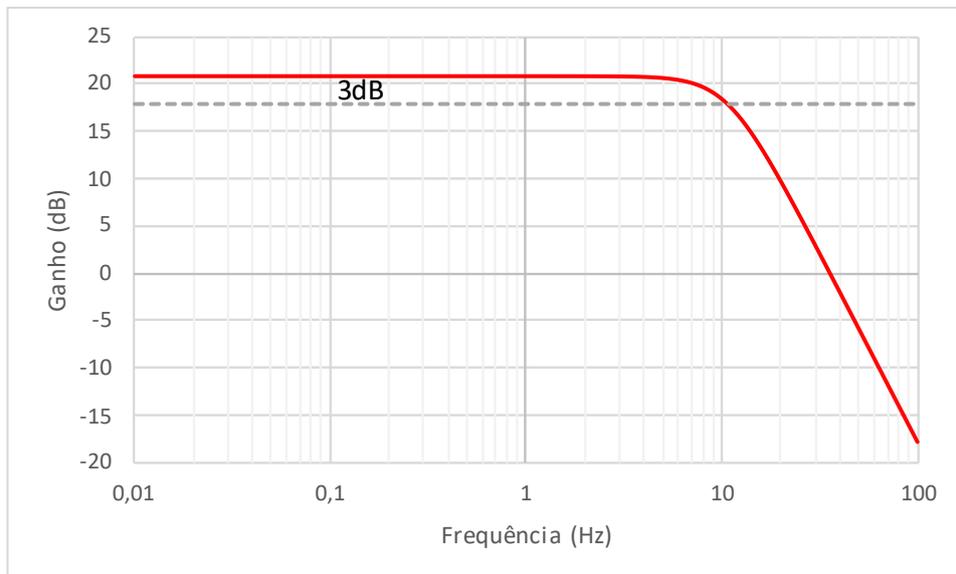
A Figura 14 mostra o circuito do filtro passa-baixas, configurado com frequência de corte de 10,8 Hz e um ganho na banda de passagem de 20 dB, enquanto a Figura 15 ilustra o diagrama de Bode da resposta simulada do filtro. Considerando que o principal ruído externo detectado é o da rede elétrica, uma atenuação de pelo menos 20 dB é suficiente para que o ruído não interfira na resposta do sistema. Essa topologia foi utilizada pela familiaridade do grupo com sua utilização e, apesar de apresentar um ganho superior a 1 devido sua estrutura, essa característica não afeta seu uso no projeto, pois os ganhos projetados são maiores do que 1.

Figura 14 – Topologia Sallen-key do filtro passa-baixas projetado



Fonte: Autor

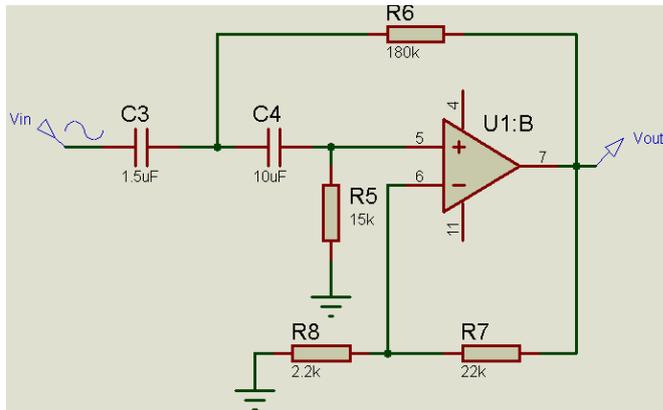
Figura 15 – Diagrama de Bode do filtro passa-baixas Butterworth de segunda ordem



Fonte: Autor

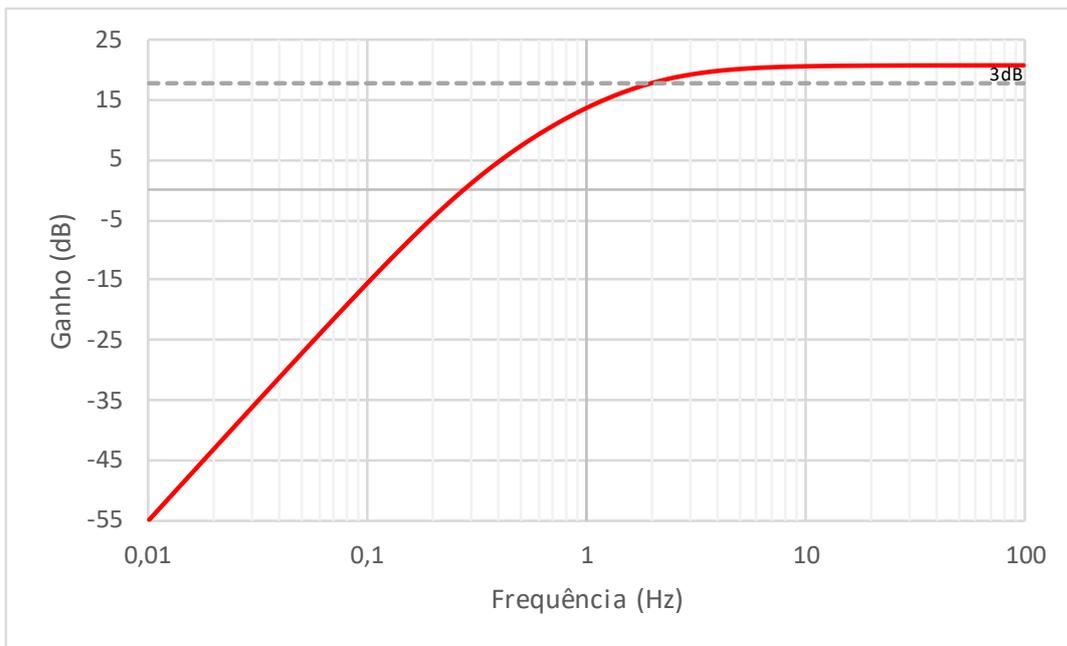
Semelhante ao filtro passa-baixas, foi projetado um filtro passa-altas com frequência de corte em 1,98 Hz e ganho de 20 dB (Figura 16), aumentando mais ainda a amplitude do sinal e eliminando o sinal DC como mostra a Figura 17 do diagrama de Bode da resposta simulada.

Figura 16 – Topologia Sallen-key do filtro passa-altas projetado



Fonte: Autor

Figura 17 – Diagrama de Bode filtro passa-altas Butterworth de segunda ordem



Fonte: Autor

4.1.1.3 Microcontrolador

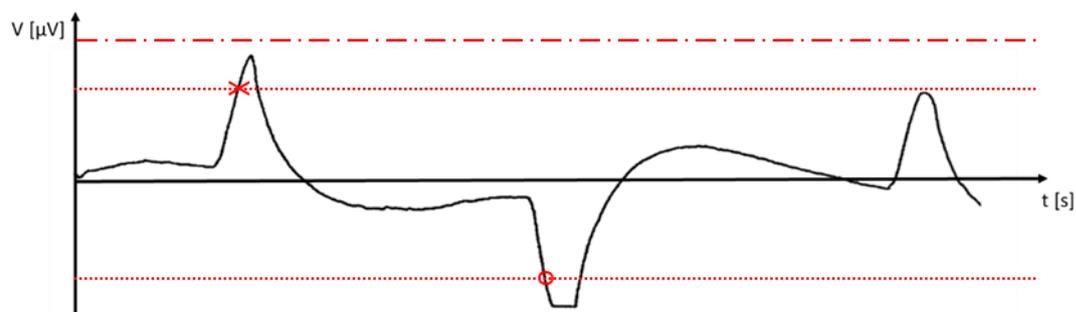
Para fazer o processamento, foi selecionada a placa Arduino Leonardo, que contém o microcontrolador *ATmega32U4*, em virtude do seu recurso de *Built-in USB* [27], o qual permite ser reconhecido pelo sistema operacional como um mouse ou teclado, eliminando a necessidade de um driver de mouse externo.

Além deste recurso, o microcontrolador contém um conversor ADC interno de 10 bits [27] o que, para uma alimentação de 5 V, entrega uma resolução absoluta de aproximadamente 5 mV.

4.1.2 Fluxograma do firmware

O firmware foi desenvolvido baseando-se no sinal de EOG, descrito na Figura 18, para a detecção de movimentos através de limiares pré-estabelecidos. No caso de uma movimentação para cima, devido à disposição dos eletrodos, espera-se que o sinal suba, atingindo o limiar superior, sinalizado por um “x”, e quando ocorre o retorno dos olhos ao repouso, o sinal alcança o limiar inferior, representado na figura por “o”. Esta ação caracteriza a validação da movimentação para cima, análogo ao sinal para a direita e oposto ao sinal para baixo e esquerda. No caso da detecção de piscada, o sinal é similar ao para cima e para direita, porém, devido a intensidade do sinal, os limiares foram estabelecidos em valores maiores, demonstrado na figura por traço e ponto.

Figura 18 – Sinal de EOG da movimentação dos olhos para cima com limiares



Fonte: Autor “adaptado de” [12]

Para dar significado e função aos sinais extraídos, foi desenvolvido um *firmware*, cujo fluxograma de alto nível conta com uma rotina principal e 09 sub-rotinas, fragmentadas em validação, movimento e piscada, descrito na Figura 19 e complementado no APÊNDICE A – FLUXOGRAMA DO FIRMWARE.

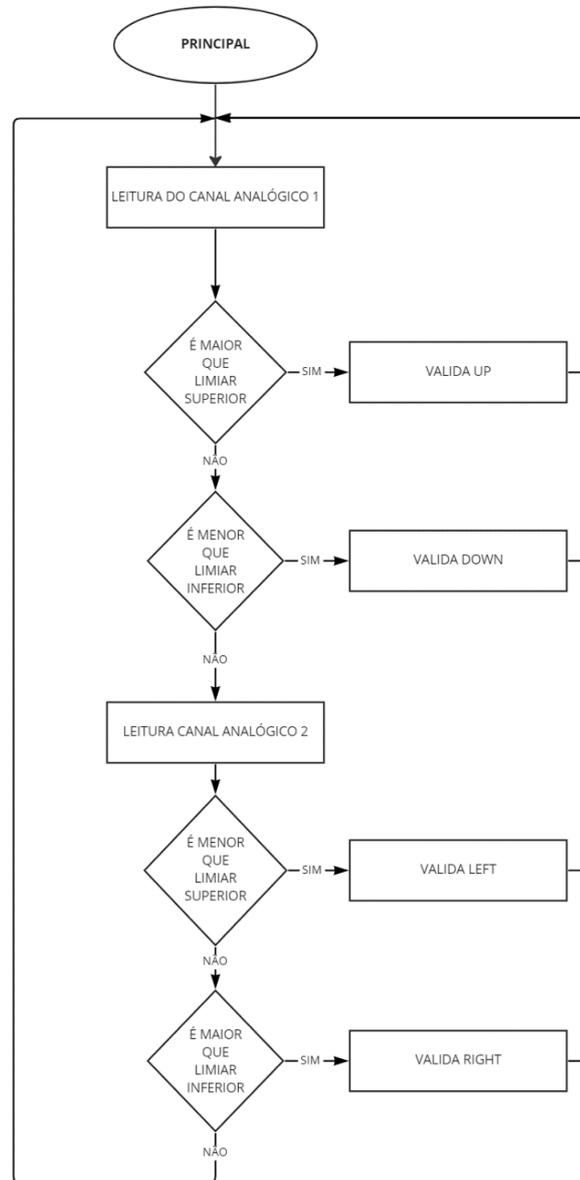
A rotina principal, Figura 19 (A), inicia com a leitura do canal analógico 1, responsável pelo sinal vertical, e verifica se o mesmo atingiu o limite superior ou inferior para entrar na respectiva sub-rotina de validação. Caso nenhum limiar tenha sido atingido, efetua-se a leitura do canal analógico 2 para executar a mesma ação, porém, direcionada ao sinal horizontal.

As sub-rotinas de validação, *VALIDA UP*, *VALIDA DOWN*, *VALIDA LEFT* e *VALIDA RIGHT*, são responsáveis pela detecção do retorno dos olhos à posição de repouso. Após o movimento, caso não tenha atingido o limiar em até 500 ms, o algoritmo assume que este movimento não é válido, o descartando. Em caso positivo, o comando será validado direcionando para a sub-rotina de movimento respectiva. Há uma exceção desta lógica para o caso do *VALIDA UP*, o qual, além das ações descritas, confere se o sinal atinge o limiar de piscada.

Caso o limiar de piscada seja alcançado, o programa chama a sub-rotina *BLINK*, a qual inicialmente aguardará 500 ms, tempo definido empiricamente, por outra piscada. Na situação em que não há uma segunda piscada, será executada a função selecionada, por exemplo um clique simples. No outro cenário, onde ocorre uma segunda piscada, o sistema irá comutar para a próxima função, seguindo a seguinte ordem: clique simples, clique duplo, clique secundário, clique *scroll* e clique e arrasta.

As sub-rotinas de movimentação (*MOVE_UP*, *MOVE_DOWN*, *MOVE_LEFT* e *MOVE_RIGHT*) contam com duas ações: aumento de velocidade e a finalização da movimentação do cursor. O controle da velocidade é dado por um flag, que define se será lenta (velocidade 1) ou rápida (velocidade 2), sendo ele alterado conforme ocorra o gesto dos olhos no mesmo sentido e direção que o cursor está realizando. Para a finalização do movimento, é necessário o deslocamento ocular no sentido contrário ao atual do cursor.

Figura 19 – Fluxograma do sistema. PRINCIPAL (A); Sub-rotina VALIDA UP (B); Sub-rotina VALIDA DOWN (C); Sub-rotina MOVE_UP (D); Sub-rotina BLINK (E)



(A)

4.1.3 Experimentos

A fim de entender e validar o tempo de adaptação dos usuários e precisão do *Eyecomm*, foram realizados testes com 5 pessoas, dentre elas homens, mulheres, jovens e idosos, por meio da metodologia de Matriz de Confusão, que correlaciona os movimentos e funções desejadas pelo usuário com o que foi efetivamente realizado pelo sistema, fazendo uma estimativa da acurácia de cada movimento e do sistema como um todo.

A tabela 2 mostra o modelo utilizado neste trabalho, onde na coluna 1 são as possibilidades de ações intencionais do usuário, sendo elas: para cima (C), para baixo (B), para esquerda (E), para direita (D), realizar a função (F), trocar de função (T). Enquanto na linha 1 estão dispostas opções que indicam a resposta do sistema à ação do usuário, mantendo a representação das letras comentadas, porém, acrescentando (N) para quando não houver resposta do sistema. Considerando que a diagonal principal da matriz corresponde à combinação certa da intenção com a resposta do sistema, é desejado que haja uma maior frequência de ocorrências.

Tabela 2 – Modelo de Matriz de Confusão

	C	B	E	D	F	T	N
C							
B							
E							
D							
F							
T							

Fonte: Autor

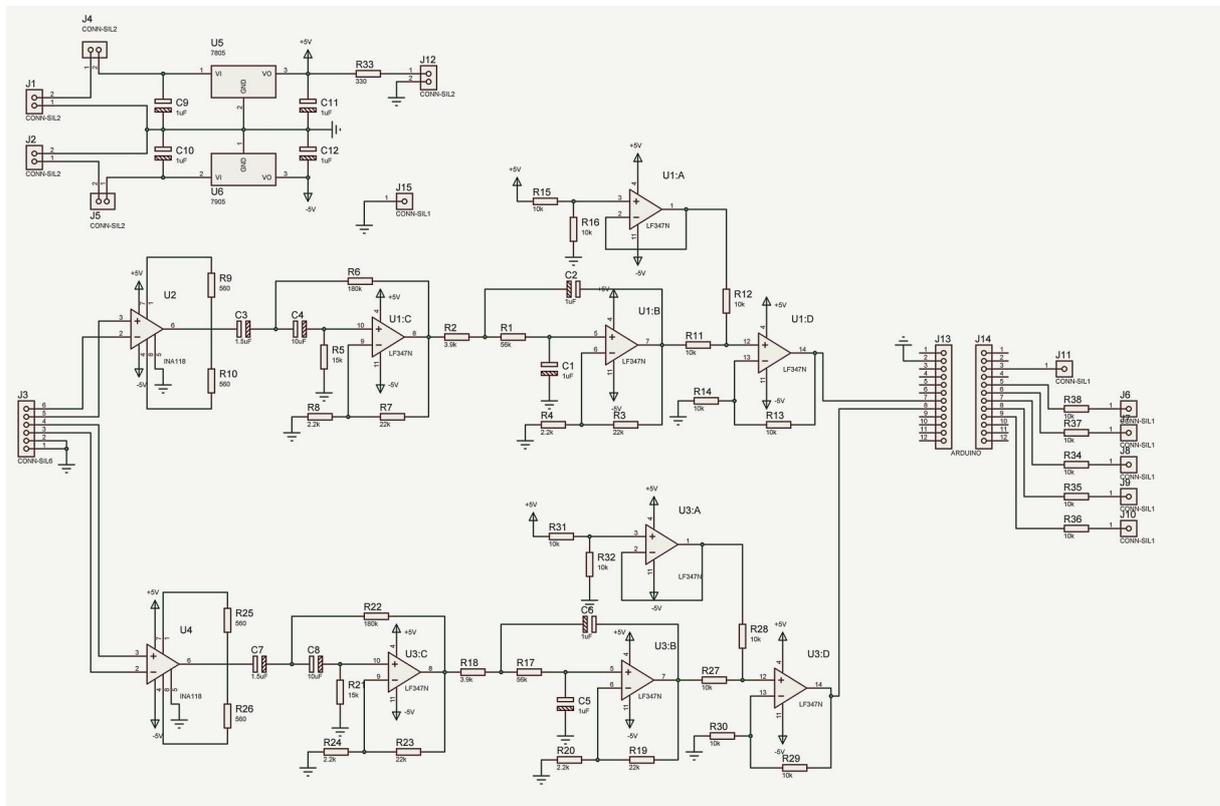
Para o preenchimento da Matriz foram executados testes em que o usuário realiza 20 vezes cada movimento, sendo eles: cima, baixo, esquerda e direita, considerando 10 para início da movimentação e 10 contrários para interrupção, e 10 vezes para função e troca de função. Esse procedimento foi realizado 3 vezes para cada usuário.

5 RESULTADOS

Neste capítulo serão abordados os resultados obtidos a partir do desenvolvimento do projeto apresentado em metodologia, sendo eles o esquemático, implementação física, código e os testes experimentais.

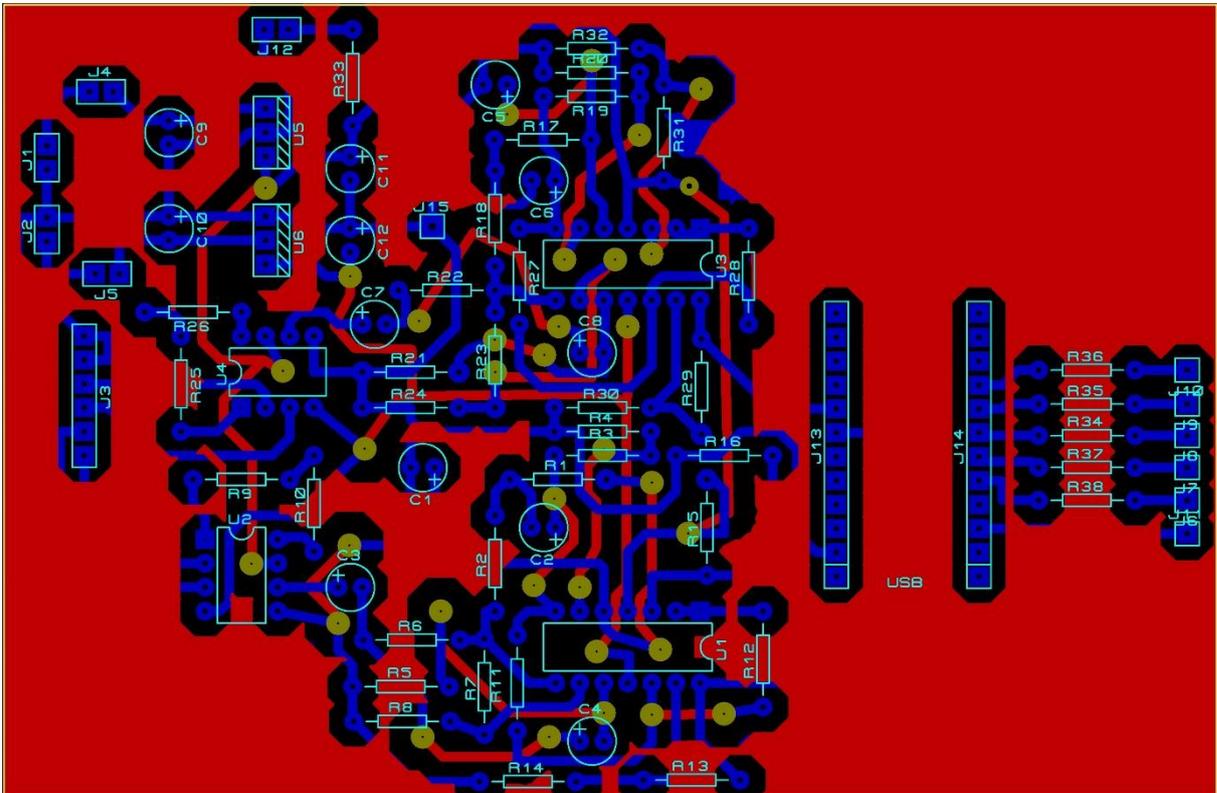
Utilizando o *software Proteus* foi feito o esquemático, ilustrado na Figura 20 (para uma maior resolução consultar o APÊNDICE B – ESQUEMÁTICO EYECOMM), e a máscara para a confecção da Placa de Circuito Impresso (PCB) ilustrada na Figura 21.

Figura 20 – Esquemático *Eyecom*



Fonte: Autor

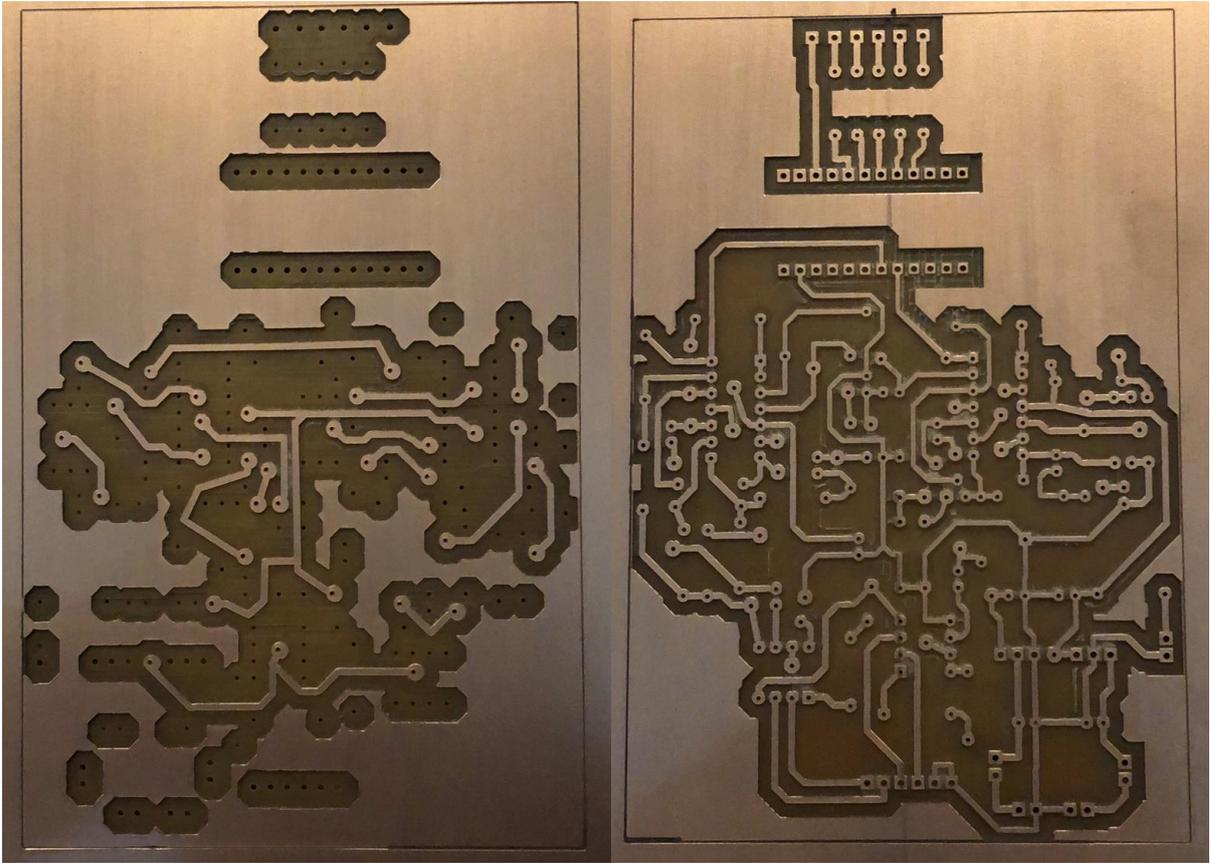
Figura 21 – Máscara da placa de circuito impresso



Fonte: Autor

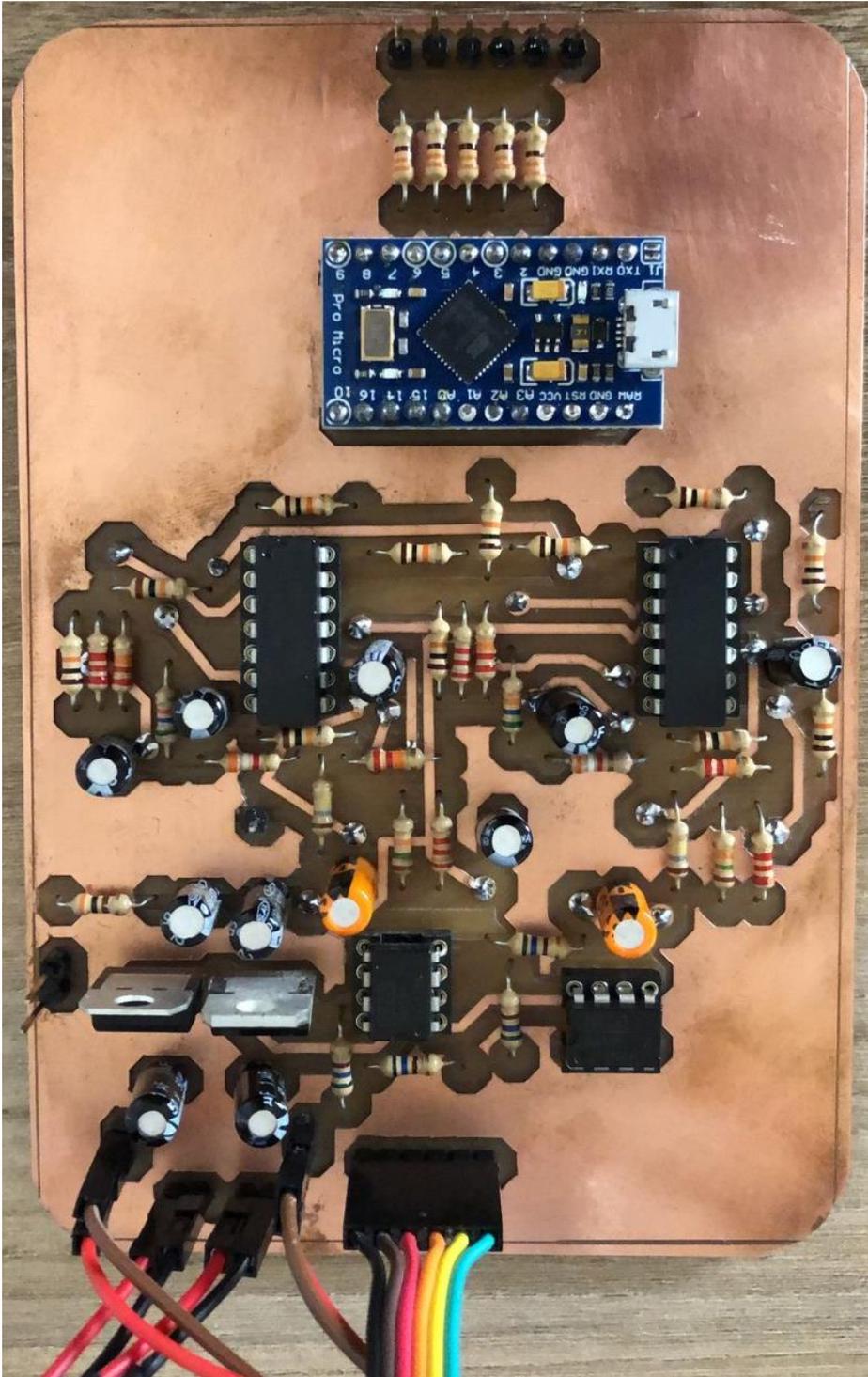
A placa foi usinada no Centro Universitário FEI com o apoio do Centro de Laboratórios Elétricos (CLE) resultando na Figura 22, e, posteriormente foram soldados os componentes, precificados na Tabela 3, conforme esquemático, obtendo a placa com o circuito completo ilustrado na Figura 23.

Figura 22 – Placa de Circuito Impresso frente e verso



Fonte: Autor

Figura 23 – Placa com o circuito completo



Fonte: Autor

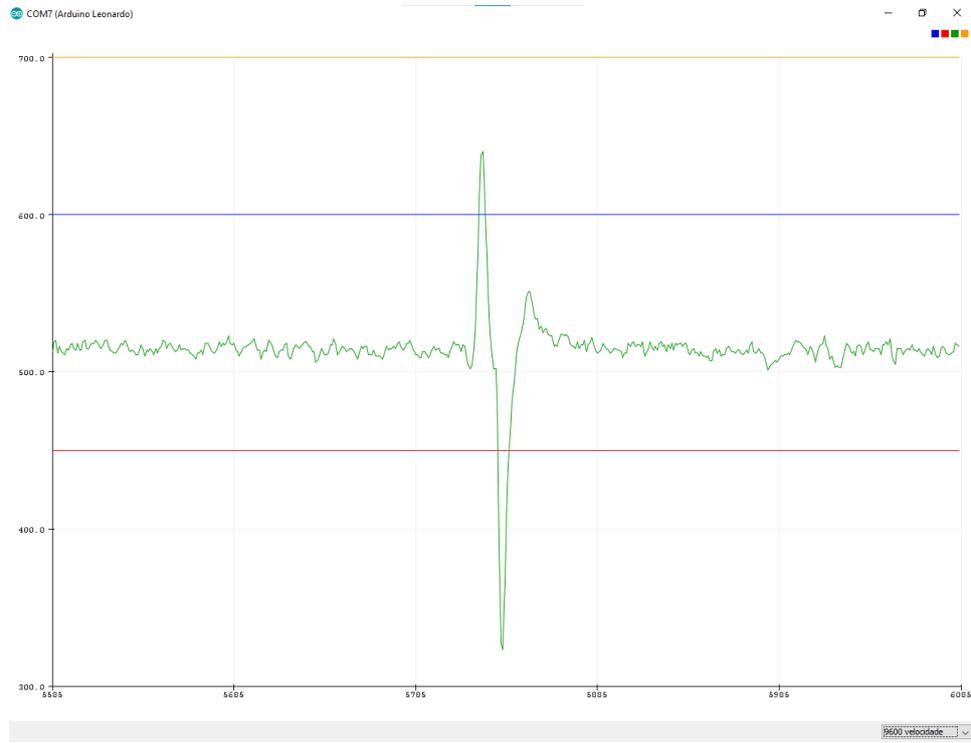
Tabela 3 – Tabela descritiva de custos do projeto (2022)

Qtde.	Descrição	Preço un.	Preço total
2	INA121	R\$ 35,00	R\$ 70,00
2	LF347N	R\$ 2,82	R\$ 5,64
1	Cabo ECG 5vias	R\$ 340,00	R\$ 340,00
1	Conector DIN	R\$ 5,40	R\$ 5,40
1	Arduino Leonardo	R\$ 79,90	R\$ 79,90
1	LM7805	R\$ 3,00	R\$ 3,00
1	LM7905	R\$ 3,00	R\$ 3,00
-	Componentes	R\$ 20,00	R\$ 20,00
TOTAL			R\$ 526,94

Fonte: Autor

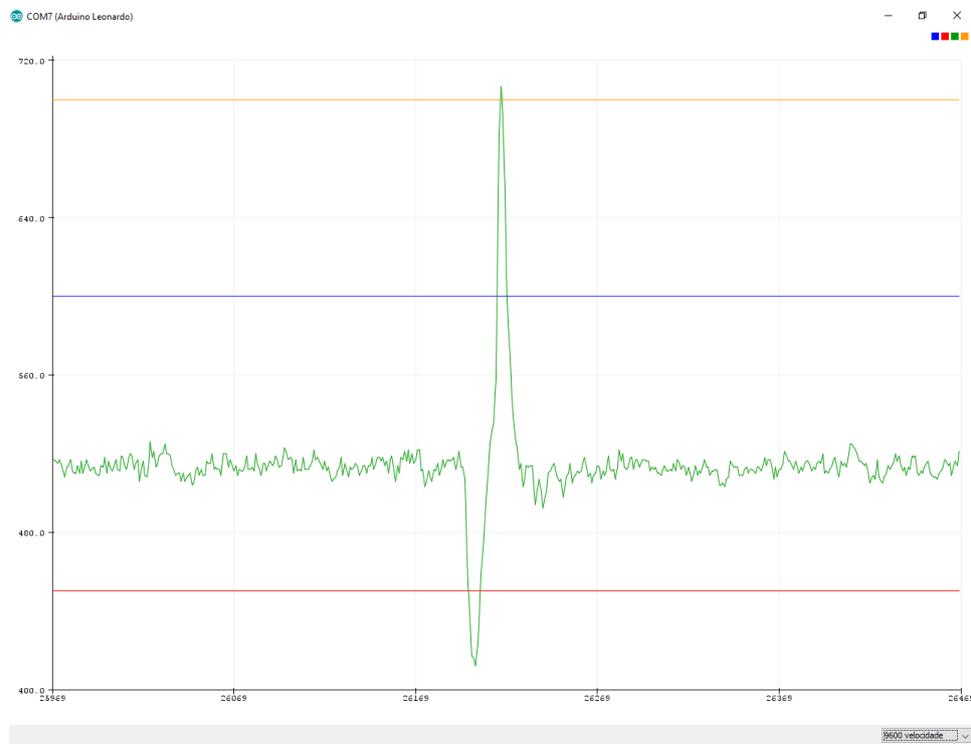
Com a placa do circuito construído, foram extraídos os sinais obtidos na saída dos circuitos de tratamento horizontal e vertical, através do conversor analógico-digital do Arduino, onde o eixo horizontal representa o número da amostra e o eixo vertical o valor amostrado. As Figuras 24 a 27 mostram a leitura do canal vertical no tempo ao realizar os movimentos de olhar para cima, baixo, uma piscada e piscada dupla, junto com os limiares: inferior, superior e de piscada, representados por linhas verticais e apresentados de baixo para cima, respectivamente. Da mesma forma nas Figuras 28 e 29, é lido o canal horizontal ao executar os movimentos de olhar para a direita e olhar para a esquerda, com os limiares inferior e superior. A definição do limiar foi feita a partir da leitura do sinal de 5 pessoas de modo que atendessem bem a todos.

Figura 24 – Sinal obtido pelo canal vertical de movimento para cima



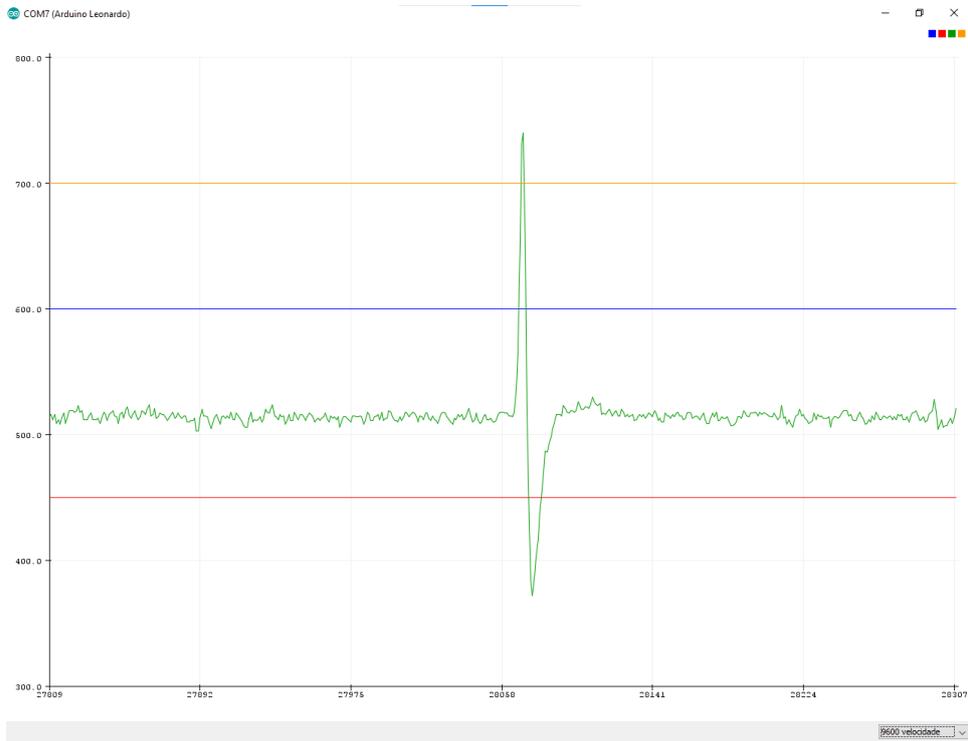
Fonte: Autor

Figura 25 – Sinal obtido pelo canal vertical de movimento para baixo



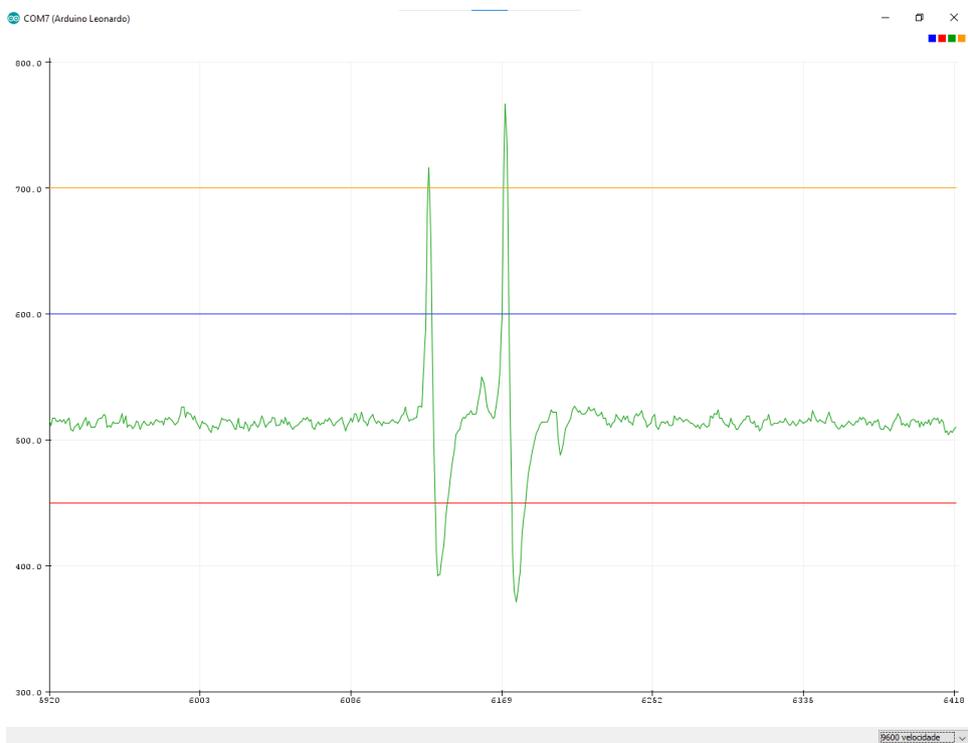
Fonte: Autor

Figura 26 – Sinal obtido pelo canal vertical de piscada



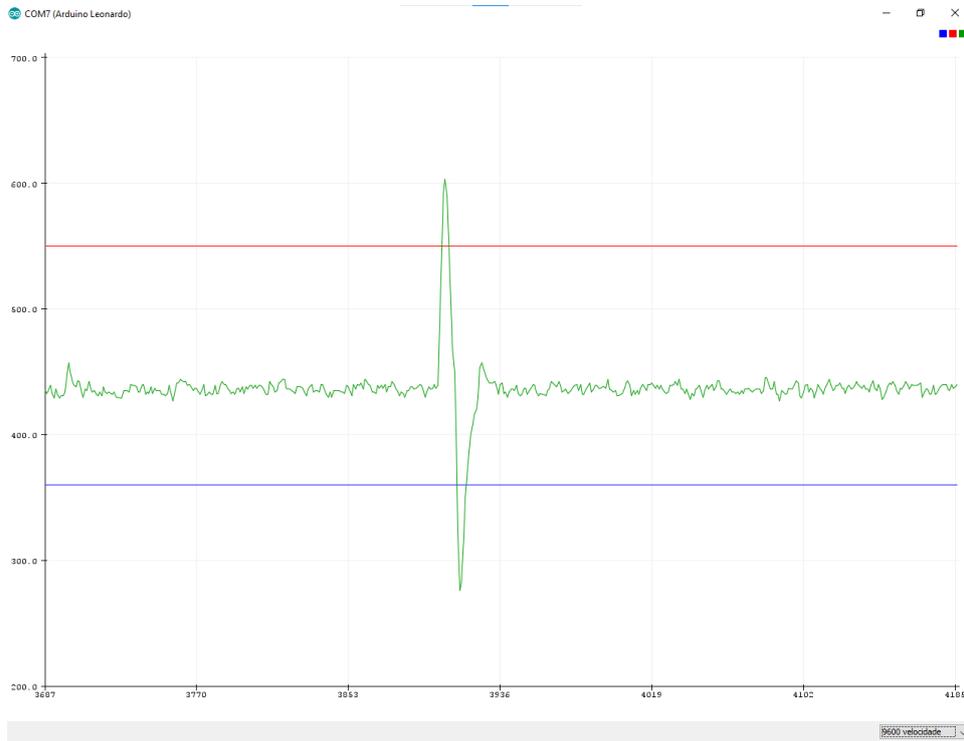
Fonte: Autor

Figura 27 – Sinal obtido pelo canal vertical de dupla piscada



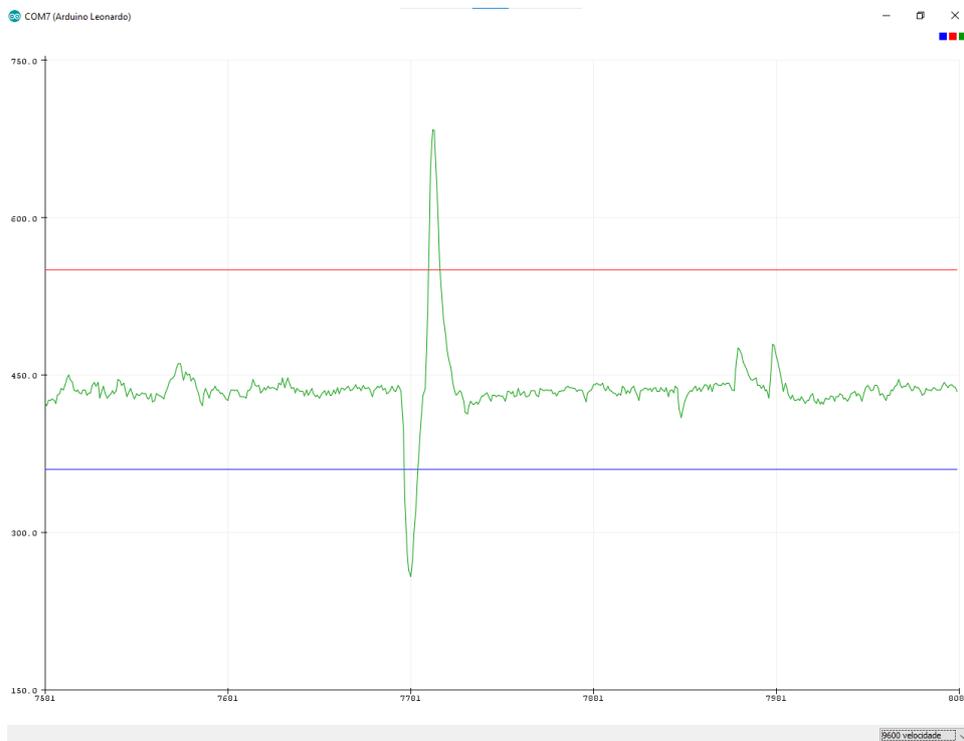
Fonte: Autor

Figura 28 – Sinal obtido pelo canal horizontal de movimento para direita



Fonte: Autor

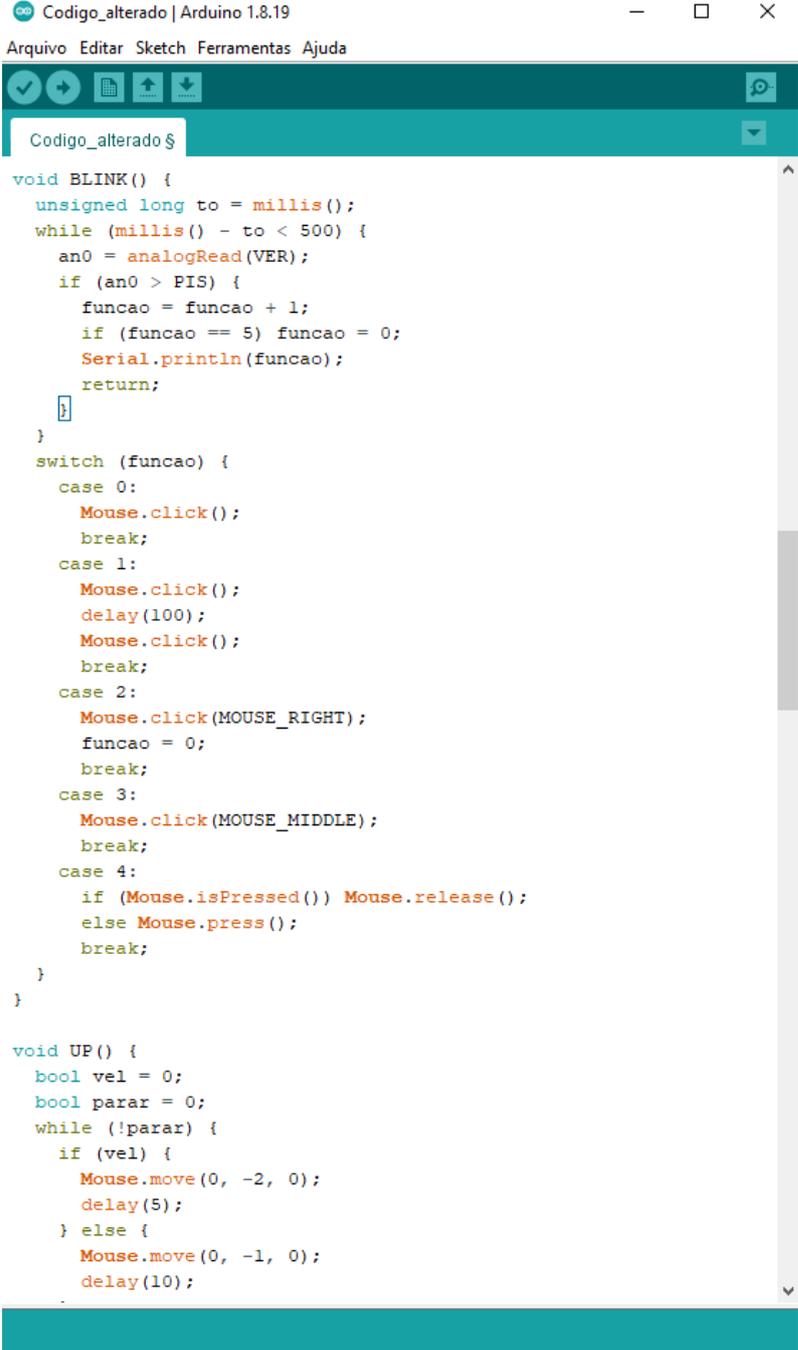
Figura 29 – Sinal obtido pelo canal horizontal de movimento para esquerda



Fonte: Autor

Para o desenvolvimento do *firmware* foi utilizado a IDE própria do Arduino conforme lógica definida no Tópico 4.1.2, conforme demonstra a Figura 30 (código completo no APÊNDICE C – CÓDIGO DO FIRMWARE).

Figura 30 – IDE do Arduino contendo o código parcial



```

Codigo_alterado | Arduino 1.8.19
Arquivo Editar Sketch Ferramentas Ajuda
Codigo_alterado $
void BLINK() {
  unsigned long to = millis();
  while (millis() - to < 500) {
    an0 = analogRead(VER);
    if (an0 > PIS) {
      funcao = funcao + 1;
      if (funcao == 5) funcao = 0;
      Serial.println(funcao);
      return;
    }
  }
  switch (funcao) {
    case 0:
      Mouse.click();
      break;
    case 1:
      Mouse.click();
      delay(100);
      Mouse.click();
      break;
    case 2:
      Mouse.click(MOUSE_RIGHT);
      funcao = 0;
      break;
    case 3:
      Mouse.click(MOUSE_MIDDLE);
      break;
    case 4:
      if (Mouse.isPressed()) Mouse.release();
      else Mouse.press();
      break;
  }
}

void UP() {
  bool vel = 0;
  bool parar = 0;
  while (!parar) {
    if (vel) {
      Mouse.move(0, -2, 0);
      delay(5);
    } else {
      Mouse.move(0, -1, 0);
      delay(10);
    }
  }
}

```

Fonte: Autor

De acordo com o Tópico 4.1.3 foram desenvolvidas as matrizes de confusão por usuário e a partir delas foi criada uma matriz que centraliza todos os resultados obtidos, visualizando a

resposta do sistema. Foram testados 5 usuários, entre eles quatro jovens (três homens e uma mulher) de 20 a 30 anos e uma mulher de 59 anos, todos usuários de óculos e nenhum diagnosticado com ELA ou LIS. A seguir, nas Tabelas 4 à 6 foram dispostos os resultados de um usuário para demonstração e apresentados na ordem que foram realizados, conforme comentado no Tópico 4.1.3, na coluna 1 são as possibilidades de ações intencionais, sendo elas: para cima (C), para baixo (B), para esquerda (E), para direita (D), realizar a função (F), trocar de função (T), e na linha 1 estão dispostas opções que indicam a resposta do sistema à ação do usuário, mantendo a representação das letras comentadas, porém, acrescentando (N) para quando não houver resposta do sistema, objetivando a visualização da adaptação do sistema pelo usuário (para maior detalhamento dos dados do restante dos usuários, consultar APÊNDICE D – MATRIZ DE CONFUSÃO POR USUÁRIO TESTE). Na sequência, da Tabela 7 à Tabela 9 foram apresentados os resultados das somatórias.

Tabela 4 – Matriz de Confusão Usuário 01- Primeira sessão

	C	B	E	D	F	T	N
C	19	0	0	0	0	0	1
B	0	20	0	0	0	0	0
E	0	0	20	0	0	0	0
D	0	0	0	19	0	0	1
F	0	0	0	0	8	0	2
T	1	0	0	0	0	9	0

Fonte: Autor

Tabela 5 – Matriz de Confusão Usuário 01- Segunda sessão

	C	B	E	D	F	T	N
C	19	0	0	0	0	0	1
B	0	20	0	0	0	0	0
E	0	0	19	0	0	0	1
D	0	0	0	20	0	0	0
F	0	0	0	0	10	0	0
T	0	0	0	0	0	10	0

Fonte: Autor

Tabela 6 – Matriz de Confusão Usuário 01- Terceira sessão

	C	B	E	D	F	T	N
C	18	0	0	0	0	0	2
B	0	19	0	0	0	0	1
E	0	0	20	0	0	0	0
D	0	0	0	20	0	0	0
F	0	0	0	0	10	0	0
T	0	0	0	0	0	10	0

Fonte: Autor

A partir dos resultados foi calculada a acurácia conforme a Equação (4).

$$\text{acurácia} = \frac{\text{total de acertos}}{\text{total de tentativas}} * 100 \quad (4)$$

Tabela 7 – Somatório das Matrizes de Confusão – Primeira sessão

	C	B	E	D	F	T	N	TOTAL (%)
C	74	0	0	0	21	0	5	74,0
B	0	97	0	0	0	0	3	97,0
E	0	0	99	0	0	0	1	99,0
D	0	0	0	96	0	0	3	97,0
F	1	0	0	0	46	0	3	92,0
T	2	0	0	0	4	42	2	84,0

Fonte: Autor

Tabela 8 – Somatório das Matrizes de Confusão – Segunda sessão

	C	B	E	D	F	T	N	TOTAL (%)
C	84	0	0	0	5	0	11	84,0
B	0	94	0	0	0	0	6	94,0
E	0	0	97	0	0	0	3	97,0
D	0	0	0	96	0	0	4	96,0
F	1	0	0	0	47	0	2	94,0
T	3	0	0	0	2	43	2	86,0

Fonte: Autor

Tabela 9 – Somatório das Matrizes de Confusão – Terceira sessão

	C	B	E	D	F	T	N	TOTAL (%)
C	82	0	0	0	14	0	4	82,0
B	0	95	0	0	0	0	5	95,0
E	0	0	99	0	0	0	1	99,0
D	0	0	0	100	0	0	0	100,0
F	2	0	0	0	46	0	2	92,0
T	2	0	0	0	2	42	4	84,0

Fonte: Autor

A partir dos dados coletados parcialmente, foi desenvolvida uma tabela para análise global das acurácias.

Tabela 10 – Acurácia total a partir do conjunto de testes

Somatório das Matrizes de Confusão – Primeira Sessão	Somatório das Matrizes de Confusão – Segunda Sessão	Somatório das Matrizes de Confusão – Terceira Sessão
91,0%	92,2%	92,8%

Fonte: Autor

6 DISCUSSÕES

Neste capítulo serão feitas as comparações entre o projeto previsto no Capítulo 4 e sua versão final, além das análises dos resultados apresentados no capítulo anterior.

A primeira adaptação realizada no circuito completo em comparação ao previsto na Metodologia foi a mudança do amplificador de instrumentação INA118 para INA121, que se tornou necessária devido ao atraso na entrega dos materiais de projeto. Optou-se pelo uso do INA121 por ter sido disponibilizado pela orientadora, além de não ter impactado na performance por sua similaridade ao INA118 [28], se diferenciando por alguns parâmetros, dentre eles o CMRR em que no INA121 é de 106 dB, que, por mais que tenha uma diferença de duas vezes menos, o ruído não foi relevante em comparação com o sinal do sistema, e a tensão máxima de *offset* é de 200 μV , que também não é impactante devido a utilização de um filtro passa-altas.

Outra adaptação realizada foi o aumento do ganho global do circuito, pois foi notado que com o ganho calculado a excursão do sinal era muito inferior ao que o conversor analógico de sinal é capaz de mensurar, de +5 V a 0 V. Com isso, decidiu-se aumentar o ganho do amplificador de instrumentação pela sua facilidade na alteração, tendo apenas que substituir o resistor R_g . Testes de bancada mostraram que para um R_g de 1,12 k Ω o sinal apresentou uma excursão próxima ao do conversor. Dessa forma, foi recalculado o ganho do amplificador de instrumentação apresentado na Equação (5).

$$R_g = 1,12 \text{ k}\Omega \quad (5.1)$$

$$G = 1 + \frac{50 \text{ k}\Omega}{R_g} \quad (5.2)$$

$$G = 45,64 \frac{\text{V}}{\text{V}} \quad (5.3)$$

$$G_{\text{db}} = 20 \log_{10}(G) \quad (5.4)$$

$$G_{\text{db}} = 33,19 \text{ dB} \quad (5.5)$$

Com as alterações feitas, o sinal obtido foi ilustrado nas Figuras 24 a 29. Foi observado a ausência do ruído de 60 Hz, validando a hipótese de que o filtro passa-baixas de segunda ordem seria suficiente para reduzi-lo. Outro fato relevante é a similaridade do sinal extraído ao apresentado na literatura, Figura 3, destacando que o sinal de piscada foi mais intenso que o de olhar para cima.

Na implementação do fluxograma em código fonte do Arduino foi utilizada a biblioteca *Mouse* para execução das funções. Além disso, foram definidos os padrões de velocidade lenta e rápida a partir de testes executados em diferentes telas, visando uma velocidade confortável a todos os dispositivos.

Analisando o somatório da matriz de confusão, Tabela 7 à Tabela 9, foi observado que os movimentos para cima apresentaram a menor das acurácias chegando à 74% no primeiro teste, devido à similaridade do sinal com o de piscada. Entretanto, no geral, obteve-se uma acurácia acima de 91,0%, aumentando conforme a sequência de testes para 92,8%, levantando a hipótese de que com o tempo o usuário tem uma melhor adaptação à resposta do sistema, ainda sim necessitando a comprovação. Tendo em vista essa dificuldade comum para novos usuários, no APÊNDICE E – MANUAL DO USUÁRIO está disponível um manual para iniciantes.

Ainda que a acurácia do sistema tenha atingido uma boa performance, um obstáculo detectado pelo grupo durante o desenvolvimento foi a interferência do sinal da piscada involuntária, que ocasionalmente é identificada como clique para o sistema. Apesar de apresentar uma curva com menor amplitude, algumas vezes ainda foi mal interpretado como uma piscada intencional.

Comparando o preço de venda de ambas as tecnologias, *Eye Tracking* e EOG, mesmo tendo ciência que os produtos por EOG estão voltados ao ramo de pesquisa e, portanto, não possuem preço de mercado, é possível estimar que ainda sim trata-se de uma tecnologia mais acessível que o *Eye Tracking* dada a grande diferença de escala de valores quando comparado ao custo do *Eyecommm*.

Ao confrontar o *Eyecommm* com o restante dos projetos estudados é possível realçar alguns fatores importantes, tais como: em trabalhos mais recentes não foi considerada a implementação de um teclado virtual possivelmente pela disponibilidade de *softwares* específicos voltados nesse nicho, sendo essa a motivação pela não inclusão neste projeto. Ao contrário de todos os projetos apresentados no estudo da arte, Tópico 3.2, foram implementadas funções de clique que abrangem todas as ações de um *mouse* regular, possibilitando uma experiência completa ao usuário. As comparações do *Eyecommm* com os diferentes projetos pesquisados foram apresentadas na Tabela 11.

Tabela 11 – Comparativo do Eyecomm com os trabalhos apresentados

	Thomas <i>et al.</i> (2005)	Lima <i>et al.</i> (2008)	A. López <i>et al.</i> (2014)	Zakir Hossain <i>et al.</i> (2017)	Ahsan-ul Kabir <i>et al.</i> (2020)	<i>Eyecomm</i> (2022)
Teclado virtual	Sim	Sim	Não	Não	Não	Não
Alimentação	Rede	Bateria simples	Bateria recarregável	Bateria simples	Bateria simples	Bateria simples
Funcionalidades de clique	Clique simples	Clique simples	Clique simples	Clique simples	Clique simples	Cliques variados
Algoritmo	Simples	Simples	Simples	Inteligência Artificial	Inteligência Artificial	Simples
Wireless	Não	Não	Não	Não	Sim	Não

Fonte: Autor

Com relação ao projeto do *Eye Access*, trabalho o qual teve-se experiência prática, observa-se uma redução significativa na quantidade de filtros utilizados, uma vez que o projeto desenvolvido por Thomas *et al.* (2005) conta com um total de 14 filtros para o tratamento do sinal e realimentação, enquanto o *Eyecomm* utiliza somente 4 sem realimentação e sem perda de performance. Além das melhorias de *hardware* e do tratamento do sinal adquirido via EOG, o projeto conta com a implementação de novas funções de clique e de aumento de velocidade, trazendo maior flexibilidade do sistema para o usuário.

7 CONCLUSÃO

A partir da análise dos resultados obtidos, foi possível concluir que o projeto cumpriu todos os objetivos previamente propostos. O grupo obteve o sinal de EOG conforme esperado e, após a etapa de filtragem, apresentou baixo ruído, praticamente desprezível, independentemente da utilização da realimentação. Juntamente com o sucesso do *hardware*, a interpretação do sinal em movimento do cursor, como também a identificação da quantidade de piscadas para a execução correta da função, foi realizada com êxito.

Uma das dificuldades enfrentadas pelo grupo foi o desenvolvimento da placa de circuito impresso, onde a mensuração das ilhas e trilhas sofreram alterações da primeira versão do projeto devido ao processo de usinagem contar com a intervenção humana para alternar entre as faces para confecção, acrescentando imprecisões. Entretanto, o problema foi contornado a partir do aumento das ilhas e trilhas para que mesmo com imprecisões durante a etapa de furos não houvesse falha de contato. Por mais que o aumento não fosse algo desejado pelo grupo pois aumentaria o tamanho do circuito final, essa modificação auxiliou durante a etapa de solda que, além de facilitar o processo, evitou o descolamento das ilhas e trilhas que poderia ocorrer devido à alta temperatura.

Os testes em usuários mostraram que a acurácia do *Eyecom* está acima de 90%, o que valida esse projeto como um sistema de alta precisão. Fazendo um estudo mais detalhado sobre a resposta de cada função, percebe-se que a maior taxa de erro está na execução do “para cima”, que devido a sua similaridade com o sinal da piscada, induz a execução do clique. Na sequência, a segunda maior taxa de erro é apresentada na troca de função, que pode ser explicada pelo fato de ser um movimento mais complexo, além de exigir um acerto no tempo entre as duas piscadas. Conforme hipótese levantada no Tópico 6, é necessário um estudo com uma população maior e mais diversificada para confirmar se há uma relação entre a repetição dos movimentos e a adaptação do usuário.

8 PERSPECTIVAS FUTURAS E SUGESTÕES DE MELHORIAS

Como perspectivas futuras, considera-se a alteração da utilização dos eletrodos descartáveis por uma estrutura em formato de tiara, além da implementação de tecnologias *Wireless*, que aprimorarão a experiência do usuário.

Considerando melhorias de *softwares*, algumas novas funcionalidades que agregariam ao sistema seriam: desativação do mouse para momentos específicos que caberia ao usuário determinar (como durante a reprodução de um vídeo) e a adição de uma janela fixa na tela que indicasse a função selecionada tendo uma melhor visualização para o usuário.

Em consonância com o que foi mencionado na conclusão, é recomendado para melhorias futuras um limiar adaptativo para a detecção de piscadas involuntárias, de modo que não interfira no clique do sistema e aprimore a performance de acordo com as singularidades de cada usuário.

REFERÊNCIAS

- [1] MINISTÉRIO DA SAÚDE. **Esclerose Lateral Amiotrófica (ELA)**. [S.l.:s.n.]: 2021. Disponível em: <<https://www.gov.br/saude/pt-br/assuntos/saude-de-a-a-z/e/esclerose-lateral-amiotrofica-ela-1>>. Acesso em: 20 mar. 2022.
- [2] AMYOTROPHIC LATERAL SCLEROSIS ASSOCIATION. **What is ALS?**. [S.l.:s.n.]. Disponível em: <<https://www.als.org/understanding-als/what-is-als>>. Acesso em: 20 mar. 2022.
- [3] AMYOTROPHIC LATERAL SCLEROSIS ASSOCIATION. **Who Gets ALS?**. [S.l.:s.n.]. Disponível em: <<https://www.als.org/understanding-als/who-gets-als>>. Acesso em: 20 mar. 2022.
- [4] AMYOTROPHIC LATERAL SCLEROSIS ASSOCIATION. **ALS Symptoms and Diagnosis**. [S.l.:s.n.]. Disponível em: <<https://www.als.org/understanding-als/symptoms-diagnosis>>. Acesso em: 25 mar. 2022.
- [5] Merck & Co., Inc., **Síndrome de Encarceramento**. Manual MSD Versão para Profissionais da Saúde. 2022. Disponível em: <<https://www.msmanuals.com/pt/profissional/dist%C3%BArbiops-neuro%C3%B3gicos/coma-e-consci%C3%Aancia-prejudicada/s%C3%ADndrome-de-encarceramento>>. Acesso em: 25 mar. 2022.
- [6] LEÓN-CARRIÓN, José; EECKHOUT, Phillippe Van; DOMÍNGUES-MORALES, Maria Del Rosario. **The locked-in syndrome: a syndrome looking for a therapy**. Brain injury. vol. 16, n. 7, pp. 555-569, 2002. DOI: 10.1080/02699050110119466
- [7] FANTINI, Sergio; BENNIS, Caoimhe; KAPLAN, David. **Biomedical Engineering Continues to Make the Future**. IEEE pulse, v. 2, n. 4, p. 70-73, 2011. DOI: 10.1109/MPUL.2011.941720
- [8] MALMIVUO, Jaakko; PLONSEY, Robert. **Bioelectromagnetism: principles and applications of bioelectric and biomagnetic fields**. Oxford University Press, USA, 1995. p. 575-580.
- [9] HOSSAIN, Zakir; SHUVO, Md Maruf Hossain; SARKER, Prionjit. **Hardware and software implementation of real time electrooculogram (EOG) acquisition system to control computer cursor with eyeball movement**. 4th International Conference on Advances in Electrical Engineering (ICAEE). IEEE, 2017. p. 132-137. DOI: 10.1109/ICAEE.2017.8255341.
- [10] LV, Zhao et al. **A novel eye movement detection algorithm for EOG driven human computer interface**. Pattern Recognition Letters, v. 31, n. 9, p. 1041-1047, 2010. DOI: 10.1016/j.patrec.2009.12.017

- [11] MCGILL UNIVERSITY. **Biological Signal Acquisition**. [S.l.:s.n.]. Disponível em: <http://www.medicine.mcgill.ca/physio/vlab/other_exps/EOG/artifacts_n.htm>. Acesso em: 14 abr. 2022.
- [12] THAKOR, Nitish V. **Biopotentials and Electrophysiology Measurement**. [S.l.]. ed. CRC Press LLC. 1999.
- [13] PERTENCE, Antônio J. **Amplificadores operacionais e filtros ativos: teoria, projetos, aplicações e laboratório**. 6. Ed. São Paulo: McGraw-Hill, 2003.
- [14] SEDRA, Adel S.; SMITH, Kenneth Carless. **Microeletrônica**. Pearson Prentice Hall, 2007. p. 53-54, 679.
- [15] BARRETO, Ana M. **EYE TRACKING COMO MÉTODO DE INVESTIGAÇÃO APLICADO ÀS CIÊNCIAS DA COMUNICAÇÃO**. Revista Comunicando, vol. 1, n. 1, pp. 168-186. 2012. Disponível em: <<https://revistas.sopcom.pt/index.php/comunicando/article/view/126/70>>. Acesso em: 21 maio 2022.
- [16] RAHNAMA-YE-MOQADDAM, Reza; VAHDAT-NEJAD, Hamed. **Designing a pervasive eye movement-based system for ALS and paralyzed patients**. 5th International Conference on Computer and Knowledge Engineering (ICCKE). IEEE, 2015. p. 218-221. DOI: 10.1109/ICCKE.2015.7365830.
- [17] TOBBI. **Eye Mobile Tobii Tablet Adaptado**. [S.l.:s.n.]. Disponível em: <<https://www.tobiibrasil.com/produto/eyemobile-tobii-tablet-adaptado/>>. Acesso em: 21 maio 2022.
- [18] SPECTRONICS. **Inclusive EyeGaze Education: Skyle for Windows**. Disponível em: <<https://www.spectronics.com.au/product/inclusive-eyegaze-education>>. Acesso em: 21 maio 2022.
- [19] MERCADO LIVRE. **Tobii Tracker Eye 5 Para Jogo Ou Comunicação Assistiva**. Disponível em: <https://produto.mercadolivre.com.br/MLB-2179466654-tobii-tracker-eye-5-para-jogo-ou-comunicacao-assistiva-_JM?matt_tool=18956390&utm_source=google_shopping&utm_medium=organic>. Acesso em: 21 maio 2022.
- [20] THOMAS, Alexandre Astolpho; CALLEJON, Fernando; FERREIRA, Francisco Antonjo Lunalvo Porfida. **EYE ACCESS: controle do cursor do mouse através do movimento dos olhos**. Trabalho de Conclusão de Curso. FEI. São Bernardo do Campo. 2005.
- [21] LIMA, Leonardo Araújo et al. **SISTEMA BASEADO EM ELETROOCULOGRAMA PARA CONTROLE DE DISPOSITIVOS COMPUTACIONAIS**. VI CEEL, 2008. UFU. Disponível em: <https://www.peteletricaufu.com.br/static/ceel/doc/artigos/artigos2008/ceel2008_25.pdf>. Acesso em: 23 maio 2022.

[22] LOPEZ, A. et al. **EOG-based system for mouse control**. In: SENSORS, 2014 IEEE. IEEE, 2014. p. 1264-1267. DOI: 10.1109/ICSENS.2014.6985240.

[23] HOSSAIN, Zakir; SHUVO, Md Maruf Hossain; SARKER, Prionjit. **Hardware and software implementation of real time electrooculogram (EOG) acquisition system to control computer cursor with eyeball movement**. In: 2017 4th international conference on advances in electrical engineering (ICAEE). IEEE, 2017. p. 132-137. DOI: 10.1109/ICAEE.2017.8255341.

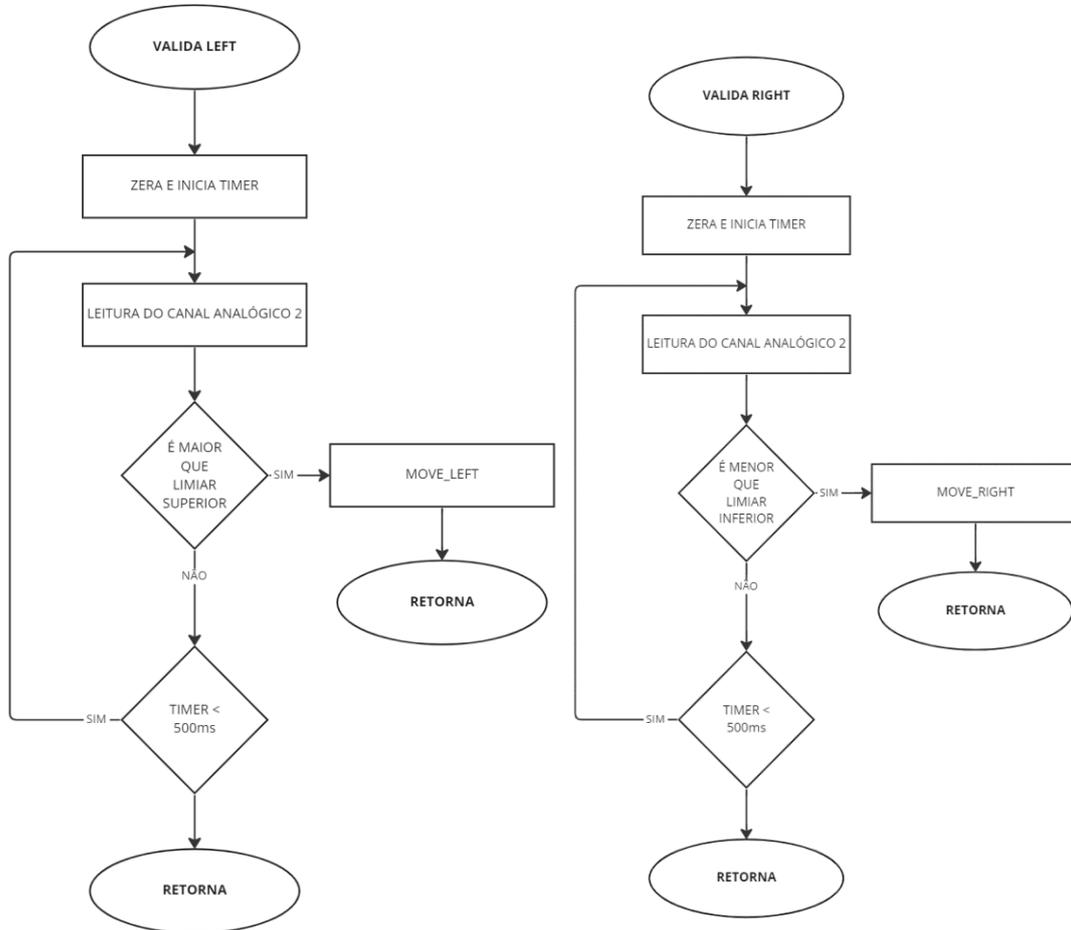
[24] KABIR, Ahsan-ul; SHAHIN, Faisal Bin; ISLAM, Md Kafiul. **Design and implementation of an EOG-based mouse cursor control for application in human-computer interaction**. In: Journal of Physics: Conference Series. IOP Publishing, 2020. DOI 10.1088/1742-6596/1487/1/012043.

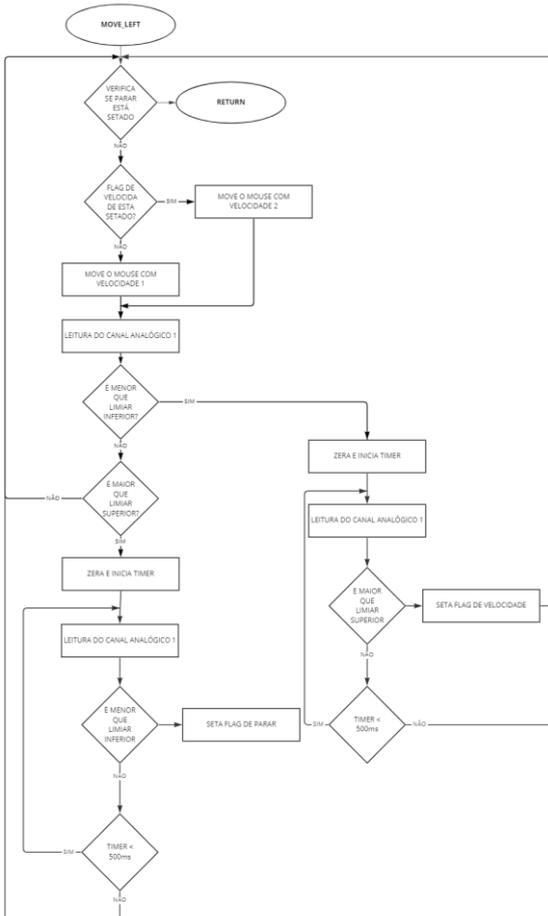
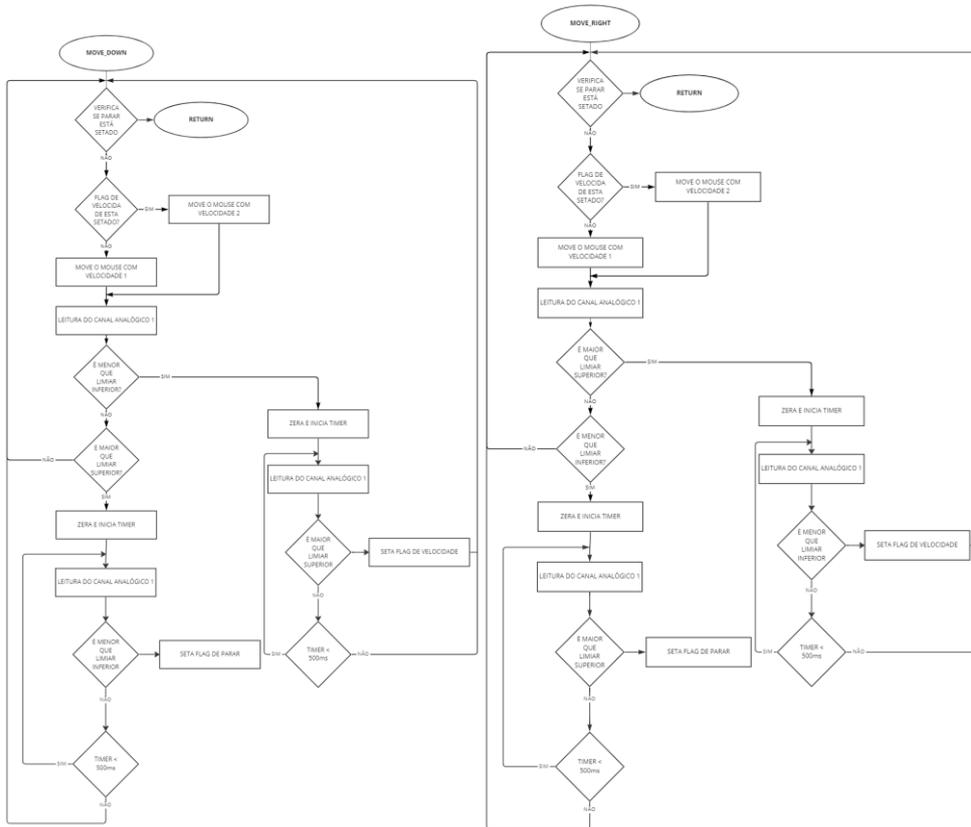
[25] ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR IEC 60601-1. **Equipamento eletromédico Parte 1: Requisitos gerais para segurança básica e desempenho essencial**. 2016. Disponível em: <<https://www.zambini.org.br/pdfs/ABNT%20NBR%20IEC%2060601-1-2010%20Emenda%201-2016%20-%20Equipamento%20eletrom%C3%A9dico%20-%20Parte%201-%20Requisitos%20gerais%20para%20seguran%C3%A7a%20b%C3%A1sica%20e%20desempenho%20essencial.pdf>>. Acesso em: 15 nov. 2022.

[26] TEXAS INSTRUMENTS. **INA118 Precision, Low-Power Instrumentation Amplifier**. [S.l.:s.n.]. 2019. Disponível em: <<https://www.ti.com/lit/ds/symlink/ina118.pdf>>. Acesso em: 28 maio 2022.

[27] ARDUINO. **Leonardo**. [S.l.:s.n.]. Disponível em: <<https://docs.arduino.cc/hardware/leonardo>>. Acesso em: 28 maio 2022.

[28] BURR BROWN. **FET-Input, Low Power INSTRUMENTATION AMPLIFIER**. Disponível em: <https://www.ti.com/lit/ds/symlink/ina121.pdf?ts=1669471980573&ref_url=https%253A%252F%252Fwww.google.com%252F>. Acesso em: 15 nov. 2022.

APÊNDICE A – FLUXOGRAMA DO FIRMWARE



APÊNDICE C – CÓDIGO DO FIRMWARE

```

1  #define HOR A0
2  #define VER A1
3  #define CLICK 2
4  #define DCLICK 3
5  #define DIREITA 4
6  #define SCROLL 5
7  #define ARRASTA 6
8
9
10 #include <Mouse.h>
11
12 const int PIS = 700;
13 const int CIMA = 600;
14 const int BAIXO = 450;
15 const int ESQ = 360;
16 const int DIR = 550;
17
18 int an0, an1;
19 int funcao = 0;
20
21 void setup() {
22     Mouse.begin();
23     pinMode(HOR, INPUT);
24     pinMode(VER, INPUT);
25     pinMode(CLICK, OUTPUT);
26     pinMode(DCLICK, OUTPUT);
27     pinMode(DIREITA, OUTPUT);
28     pinMode(SCROLL, OUTPUT);
29     pinMode(ARRASTA, OUTPUT);
30     digitalWrite(CLICK, HIGH);
31     digitalWrite(DCLICK, LOW);
32     digitalWrite(DIREITA, LOW);
33     digitalWrite(SCROLL, LOW);
34     digitalWrite(ARRASTA, LOW);
35 }
36
37 void loop() {
38     an0 = analogRead(VER);
39     if (an0 > CIMA) {
40         unsigned long to = millis();
41         bool pisque = 0;
42         while (millis() - to < 500) {
43             an0 = analogRead(VER);
44             if (an0 > PIS) {
45                 pisque = 1;
46             }
47             if (an0 < BAIXO) {
48                 if (pisque) BLINK();
49                 delay(300);
50                 if (!pisque) UP();
51                 break;
52             }
53         }
54         return;
55     }
56     if (an0 < BAIXO) {
57         unsigned long to = millis();
58         while (millis() - to < 500) {
59             an0 = analogRead(VER);
60             if (an0 > CIMA) {
61                 delay(100);
62                 DOWN();
63                 break;

```

```

64     | }
65     }
66     return;
67 }
68 an1 = analogRead(HOR);
69 if (an1 > DIR) {
70     unsigned long to = millis();
71     while (millis() - to < 500) {
72         an1 = analogRead(HOR);
73         if (an1 < ESQ) {
74             delay(100);
75             RIGHT();
76             break;
77         }
78     }
79     return;
80 }
81 if (an1 < ESQ) {
82     unsigned long to = millis();
83     while (millis() - to < 500) {
84         an1 = analogRead(HOR);
85         if (an1 > DIR) {
86             delay(100);
87             LEFT();
88             break;
89         }
90     }
91     return;
92 }
93 }
94
95 void BLINK() {
96     unsigned long to = millis();
97     while (millis() - to < 500) {
98         an0 = analogRead(VER);
99         if (an0 > PIS) {
100             digitalWrite(funcao +2, LOW);
101             funcao = funcao + 1;
102             if (funcao == 5) funcao = 0;
103             digitalWrite(funcao +2, HIGH);
104             return;
105         }
106     }
107     switch (funcao) {
108     case 0:
109         Mouse.click();
110         break;
111     case 1:
112         Mouse.click();
113         delay(100);
114         Mouse.click();
115         break;
116     case 2:
117         Mouse.click(MOUSE_RIGHT);
118         funcao = 0;
119         digitalWrite(DIREITA, LOW);
120         digitalWrite(CLICK, HIGH);
121         break;
122     case 3:
123         Mouse.click(MOUSE_MIDDLE);
124         break;
125     case 4:
126         if (Mouse.isPressed()) Mouse.release();

```

```

127     else Mouse.press();
128     break;
129 }
130 }
131
132 void UP() {
133     bool vel = 0;
134     bool parar = 0;
135     while (!parar) {
136         if (vel) {
137             Mouse.move(0, -2, 0);
138             delay(5);
139         } else {
140             Mouse.move(0, -1, 0);
141             delay(10);
142         }
143         an0 = analogRead(VER);
144         if (an0 > CIMA) {
145             unsigned long to = millis();
146             while (millis() - to < 500 && !vel) {
147                 an0 = analogRead(VER);
148                 if (an0 < BAIXO) {
149                     vel = 1;
150                     delay(300);
151                     an0 = analogRead(VER);
152                 }
153             }
154         }
155         if (an0 < BAIXO) {
156             unsigned long to = millis();
157             while (millis() - to < 500 && !parar) {
158                 an0 = analogRead(VER);
159                 if (an0 > CIMA) {
160                     delay(300);
161                     parar = 1;
162                 }
163             }
164         }
165     }
166 }
167
168 void DOWN() {
169     bool vel = 0;
170     bool parar = 0;
171     while (!parar) {
172         if (vel) {
173             Mouse.move(0, 2, 0);
174             delay(5);
175         } else {
176             Mouse.move(0, 1, 0);
177             delay(20);
178         }
179         an0 = analogRead(VER);
180         if (an0 < BAIXO) {
181             unsigned long to = millis();
182             while (millis() - to < 500 && !vel) {
183                 an0 = analogRead(VER);
184                 if (an0 > CIMA) {
185                     vel = 1;
186                     delay(300);
187                     an0 = analogRead(VER);
188                 }

```

```

189     }
190   }
191   if (an0 > CIMA) {
192     unsigned long to = millis();
193     while (millis() - to < 500 && !parar) {
194       an0 = analogRead(VER);
195       if (an0 < BAIIXO) {
196         delay(300);
197         parar = 1;
198       }
199     }
200   }
201 }
202 }
203
204 void RIGHT() {
205   bool vel = 0;
206   bool parar = 0;
207   while (!parar) {
208     if (vel) {
209       Mouse.move(2, 0, 0);
210       delay(5);
211     } else {
212       Mouse.move(1, 0, 0);
213       delay(10);
214     }
215     an0 = analogRead(HOR);
216     if (an0 > DIR) {
217       unsigned long to = millis();
218       while (millis() - to < 500 && !vel) {
219         an0 = analogRead(HOR);
220         if (an0 < ESQ) {
221           vel = 1;
222           delay(300);
223           an0 = analogRead(HOR);
224         }
225       }
226     }
227     if (an0 < ESQ) {
228       unsigned long to = millis();
229       while (millis() - to < 500 && !parar) {
230         an0 = analogRead(HOR);
231         if (an0 > DIR) {
232           delay(300);
233           parar = 1;
234         }
235       }
236     }
237   }
238 }
239
240 void LEFT() {
241   bool vel = 0;
242   bool parar = 0;
243   while (!parar) {
244     if (vel) {
245       Mouse.move(-2, 0, 0);
246       delay(5);
247     } else {
248       Mouse.move(-1, 0, 0);
249       delay(10);
250     }
251     an0 = analogRead(HOR);

```

```

213     delay(10);
214 }
215 an0 = analogRead(HOR);
216 if (an0 > DIR) {
217     unsigned long to = millis();
218     while (millis() - to < 500 && !vel) {
219         an0 = analogRead(HOR);
220         if (an0 < ESQ) {
221             vel = 1;
222             delay(300);
223             an0 = analogRead(HOR);
224         }
225     }
226 }
227 if (an0 < ESQ) {
228     unsigned long to = millis();
229     while (millis() - to < 500 && !parar) {
230         an0 = analogRead(HOR);
231         if (an0 > DIR) {
232             delay(300);
233             parar = 1;
234         }
235     }
236 }
237 }
238 }
239
240 void LEFT() {
241     bool vel = 0;
242     bool parar = 0;
243     while (!parar) {
244         if (vel) {
245             Mouse.move(-2, 0, 0);
246             delay(5);
247         } else {
248             Mouse.move(-1, 0, 0);
249             delay(10);
250         }
251         an0 = analogRead(HOR);
252         if (an0 < ESQ) {
253             unsigned long to = millis();
254             while (millis() - to < 500 && !vel) {
255                 an0 = analogRead(HOR);
256                 if (an0 > DIR) {
257                     vel = 1;
258                     delay(300);
259                     an0 = analogRead(HOR);
260                 }
261             }
262         }
263         if (an0 > DIR) {
264             unsigned long to = millis();
265             while (millis() - to < 500 && !parar) {
266                 an0 = analogRead(HOR);
267                 if (an0 < ESQ) {
268                     delay(300);
269                     parar = 1;
270                 }
271             }
272         }
273     }
274 }
275 }

```

APÊNDICE D – MATRIZ DE CONFUSÃO POR USUÁRIO TESTE

USUÁRIO 01 Teste 1

	C	B	E	D	F	T	N
C	19	0	0	0	0	0	1
B	0	20	0	0	0	0	0
E	0	0	20	0	0	0	0
D	0	0	0	19	0	0	1
F	0	0	0	0	8	0	2
T	1	0	0	0	0	9	0

USUÁRIO 01 Teste 2

	C	B	E	D	F	T	N
C	19	0	0	0	0	0	1
B	0	20	0	0	0	0	0
E	0	0	19	0	0	0	1
D	0	0	0	20	0	0	0
F	0	0	0	0	10	0	0
T	0	0	0	0	0	10	0

USUÁRIO 01 Teste 3

	C	B	E	D	F	T	N
C	18	0	0	0	0	0	2
B	0	19	0	0	0	0	1
E	0	0	20	0	0	0	0
D	0	0	0	20	0	0	0
F	0	0	0	0	10	0	0
T	0	0	0	0	0	10	0

Usuário 2 Teste 1

	C	B	E	D	F	T	N
C	16	0	0	0	3	0	1
B	0	19	0	0	0	0	1
E	0	0	20	0	0	0	0
D	0	0	0	19	0	0	1
F	1	0	0	0	8	0	1
T	1	0	0	0	1	6	2

Usuário 2 Teste 2

	C	B	E	D	F	T	N
C	17	0	0	0	1	0	2
B	0	17	0	0	0	0	3
E	0	0	20	0	0	0	0
D	0	0	0	19	0	0	1
F	1	0	0	0	7	0	2
T	3	0	0	0	0	5	2

Usuário 2 Teste 3

	C	B	E	D	F	T	N
C	18	0	0	0	2	0	0
B	0	20	0	0	0	0	0
E	0	0	20	0	0	0	0
D	0	0	0	20	0	0	0
F	2	0	0	0	6	0	2
T	2	0	0	0	1	5	2

Usuário 3 Teste 1

	C	B	E	D	F	T	N
C	13	0	0	0	6	0	1
B	0	20	0	0	0	0	0
E	0	0	20	0	0	0	0
D	0	0	0	20	0	0	0
F	0	0	0	0	10	0	0
T	0	0	0	0	1	9	0

Usuário 3 Teste 2

	C	B	E	D	F	T	N
C	14	0	0	0	4	0	2
B	0	20	0	0	0	0	0
E	0	0	19	0	0	0	1
D	0	0	0	20	0	0	0
F	0	0	0	0	10	0	0
T	0	0	0	0	1	9	0

Usuário 3 Teste 3

	C	B	E	D	F	T	N
C	15	0	0	0	5	0	0
B	0	20	0	0	0	0	0
E	0	0	20	0	0	0	0
D	0	0	0	20	0	0	0
F	0	0	0	0	10	0	0
T	0	0	0	0	1	8	1

Usuário 4 Teste 1

	C	B	E	D	F	T	N
C	12	0	0	0	8	0	0
B	0	20	0	0	0	0	0
E	0	0	20	0	0	0	0
D	0	0	0	20	0	0	0
F	0	0	0	0	10	0	0
T	0	0	0	0	1	9	0

Usuário 4 Teste 2

	C	B	E	D	F	T	N
C	15	0	0	0	0	0	5
B	0	20	0	0	0	0	0
E	0	0	19	0	0	0	1
D	0	0	0	20	0	0	0
F	0	0	0	0	10	0	0
T	0	0	0	0	0	10	0

Usuário 4 Teste 3

	C	B	E	D	F	T	N
C	14	0	0	0	4	0	2
B	0	20	0	0	0	0	0
E	0	0	20	0	0	0	0
D	0	0	0	20	0	0	0
F	0	0	0	0	10	0	0
T	0	0	0	0	0	10	0

Usuário 5 Teste 1

	C	B	E	D	F	T	N
C	14	0	0	0	4	0	2
B	0	18	0	0	0	0	2
E	0	0	19	0	0	0	1
D	0	0	0	18	0	0	1
F	0	0	0	0	10	0	0
T	0	0	0	0	1	9	0

Usuário 5 Teste 2

	C	B	E	D	F	T	N
C	19	0	0	0	0	0	1
B	0	17	0	0	0	0	3
E	0	0	20	0	0	0	0
D	0	0	0	17	0	0	3
F	0	0	0	0	10	0	0
T	0	0	0	0	1	9	0

Usuário 5 Teste 3

	C	B	E	D	F	T	N
C	17	0	0	0	3	0	0
B	0	16	0	0	0	0	4
E	0	0	19	0	0	0	1
D	0	0	0	20	0	0	0
F	0	0	0	0	10	0	0
T	0	0	0	0	0	9	1

APÊNDICE E – MANUAL DO USUÁRIO



EYECOMM Manual do usuário

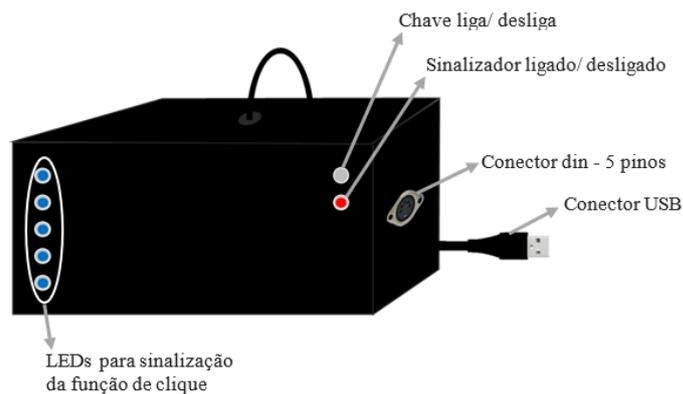


EYECOMM Manual do usuário

Sobre este manual

Este manual do usuário foi desenvolvido especialmente para detalhar como conectar os eletrodos faciais para uso do dispositivo em seu melhor desempenho, executar conexões iniciais necessária e orientar o cursor a partir dos olhos.

Layout do dispositivo



Posicionamento dos eletrodos

A conexão dos eletrodos deve ocorrer conforme ilustrado na figura abaixo e descrição detalhadamente a seguir, e após isso devem ser conectados os cabos.

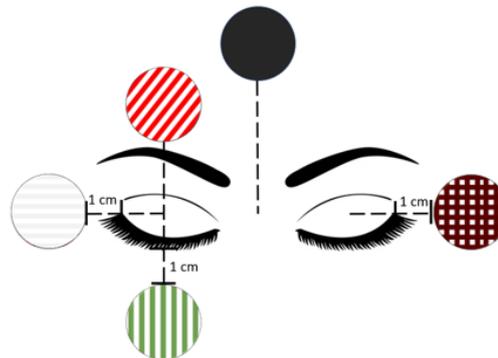
- Eletrodo marrom: Aproximadamente 1 cm a esquerda do final do olho esquerdo do usuário;
- Eletrodo branco: Aproximadamente 1 cm a direita do final do olho direito do usuário;
- Eletrodo vermelho: Logo acima da sobrancelha do usuário;
- Eletrodo verde: Aproximadamente 1 cm abaixo do final do olho direito do usuário;
- Eletrodo preto: Centralizado a testa do usuário;



EYECOMM

Manual do usuário

Posicionamento dos eletrodos



Conexão do dispositivo

- Conecte o cabo USB do dispositivo ao aparelho que será guiado pelo *mouse*.
- Na chave liga/ desliga, ligar o dispositivo.

Nota: É recomendado que o EYECOMM fique localizado próximo a tela a ser utilizada para que a conferência da função atualmente selecionada seja facilmente visualizada.

Orientação do cursor

- Movimentação para direita: Rotacione o olho para a direita e retorne ao centro.
- Movimentação para esquerda: Rotacione o olho para a esquerda e retorne ao centro.
- Movimentação para cima: Rotacione o olho para cima e retorne ao centro.
- Movimentação para baixo: Rotacione o olho para baixo e retorne ao centro.
- Movimentação para clicar: Piscada dos olhos.
- Movimentação para mudança de função: Piscada dupla.



EYECOMM

Manual do usuário

EYECOMM

Manual do usuário

Orientação do cursor

VELOCIDADE: O *Eyecom* conta com duas velocidades de uso. Inicialmente, ao iniciar o comando, o mouse se movimentará a partir da velocidade lenta, para alterar para a velocidade mais rápida, repetir o movimento inicialmente utilizado.

INTERRUPÇÃO: Para interromper a movimentação do cursor, é necessário fazer o comando contrário ao inicial. Exemplo: Para interromper o movimento do cursor para cima, deve-se rotacionar os olhos para baixo e então retornar ao centro.

Nota: Caso tenha dificuldade em obter a movimentação do cursor, é recomendado variar a amplitude da movimentação dos olhos para uma adaptação do usuário, o mesmo é indicado para a variação do tempo entre a movimentação e o retorno ao centro (posição de descanso dos olhos).

Recomendações para novos usuários

A partir do primeiro uso do sistema, caso tenha sentido dificuldade para alguma/as das movimentações do cursor, é recomendado que faça a prática repetitiva, dado que com o maior tempo de uso, melhor a adaptação e memória muscular dos movimentos.

