

Projeto de Graduação



13 de dezembro de 2023

DECODIFICAÇÃO E SISTEMA IHM DE PROCESSAMENTO DE BIOSSINAIS UTILIZANDO MICROCONTROLADORES

Leonardo Farina de Souza de Botton



www.ele.puc-rio.br

Projeto de Graduação



DECODIFICAÇÃO E SISTEMA IHM DE PROCESSAMENTO DE BÍOSSINAIS UTILIZANDO MICROCONTROLADORES

Aluno: Leonardo Farina de Souza de Botton

Orientador: PhD. William de Souza Barbosa

Coorientadora: PhD. Flávia de Miranda Gonçalves

Trabalho apresentado com requisito parcial à conclusão do curso de Engenharia Elétrica na Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, Brasil.

Agradecimentos

À minha família por me apoiar na jornada de estudo durante todos esses anos, e ressaltar a importância da busca por constante aprimoramento.

Ao meu orientador William de Souza Barbosa e minha coorientadora, Flávia de Miranda Gonçalves, por todo o suporte, incentivo e mentoria no processo de escrita deste documento. Obrigado pela paciência e por serem solícitos com tantas das minhas dúvidas ao longo do caminho.

À minha namorada, Dafne Borges da Cunha, que me deu forças apesar de tudo, esteve do meu lado para todos os desafios e me ajudou a enfrentá-los de cabeça erguida. Me deu apoio e garra para não desistir de buscar soluções. Sem sua inspiração e determinação a me fazer acreditar, este trabalho não seria possível. Em todas as madrugadas de produção de texto, não poderia ter melhor companhia.

Ao Dado Sutter, que me deu a certeza da minha vocação. Sem ele não passaria pela PUC. O que começou como um *hobby* está se tornando um diploma, e sou muito grato por tudo o que construímos ao longo de todos esses anos.

À PUC-Rio por toda a extensa bagagem de aprendizado que levarei para a carreira.

Ao Instituto Tecgraf por disponibilizar componentes para testar parte do que foi desenvolvido.

Muito obrigado a todos os que me ajudaram nessa trajetória.

Resumo

O trabalho aborda dispositivos *wearables*, eletrônicos usados como acessórios junto ao corpo, que facilitam a interação integrada com a tecnologia. Esses dispositivos têm desempenhado um papel significativo na sociedade contemporânea, oferecendo oportunidades na aquisição e análise de biossinais, impactando setores como indústria, esporte e medicina. Dessa forma, o objetivo do trabalho foi desenvolver uma estrutura que realize a leitura de biossinais e apresente os dados ao usuário, possibilitando análise de terceiros em diferentes distâncias, tornando o monitoramento facilitado. Para isso, foi utilizado o *wearable* Polar H10 que é um monitor cardíaco utilizado na aquisição de biossinais, a conexão sem fio foi estabelecida pelo protocolo *Bluetooth*[®], um microcontrolador (ESP32) foi utilizado para requisitar dados do *wearable* e a exibição dos dados requisitados em uma Interface Homem - Máquina (IHM). O sistema foi desenvolvido utilizando o Python para interpretar e monitorar os dados de saúde do usuário. Com isso, foi possível criar a interface para monitoramento de biossinais trazendo agilidade no acesso e processamento dessas informações. Essa interface demonstrou ter potencial para ser empregada em uma variedade de cenários nos quais é essencial realizar monitoramento contínuo, mesmo em ambientes distintos entre o agente e a pessoa monitorada.

Palavras-chave: *Wearables; Biossinais; Microcontroladores; Interface Homem-Máquina*

DECODING AND IHM BIOSIGNAL PROCESSING SYSTEM USING MICROCONTROLLERS

Abstract

The paper looks at wearable devices, electronics worn as accessories next to the body, which facilitate integrated interaction with technology. These devices have played a significant role in contemporary society, offering opportunities in the acquisition and analysis of biosignals, impacting sectors such as industry, sport and medicine. The aim of this work was to develop a structure that reads biosignals and presents the data to the user, enabling third-party analysis at different distances, making monitoring easier. To do this, the Polar H10 wearable was used, which is a cardiac monitor used to acquire biosignals, the wireless connection was established using the *Bluetooth*® protocol, a microcontroller (ESP32) was used to request data from the wearable and display the requested data on a Human-Machine Interface (HMI). The system was developed using *Python* to interpret and monitor the user's health data. With this, it was possible to create the interface for monitoring biosignals, bringing agility to accessing and processing this information. This interface proved to have the potential to be used in a variety of scenarios in which continuous monitoring is essential, even in different environments between the agent and the person being monitored.

Keywords: *Wearables; Biosignals; Microcontrollers; Man-Machine Interface*

Lista de Figuras

Figura 1 - Vendas internacionais de fones wireless de 2018 a 2021 (adaptado de Statista, c2023)	5
Figura 2 - Bracelete smart da marca RINGLY (extraído de Ringly, c2023).....	6
Figura 3 - Eventos elétricos do coração (extraído de Silverthorn, 2017)	15
Figura 4 - Camadas de comunicação BLE (Mathworks, 2021).....	21
Figura 5 - Estrutura do sistema de agrupamento do GATT (adaptado de Kolban, 2018)	22
Figura 6 - Monitor cardíaco Polar H10 usado no projeto	23
Figura 7 - Dispositivos Bluetooth® encontrados.....	24
Figura 8 - Lista de serviços da faixa Polar H10	24
Figura 9 - Algumas características do dispositivo.....	24
Figura 10 - Log de comunicação com a característica Medição de Frequência Cardíaca.....	31
Figura 11 - Inscrição e interpretação das notificações da característica Medição de Frequência Cardíaca	31
Figura 12 - Log de comunicação com a característica Nível da Bateria	32
Figura 13 - Leitura da característica Nível da Bateria.....	32
Figura 14 - Referência dos eixos de acelerômetro da faixa Polar H10 (Polar Electro Oy, 2021c)	40
Figura 15 - Inscrição para notificação e indicação.....	42
Figura 16 - Configuração do MTU de conexão	42
Figura 17 - Cadastro de mensagens arbitrárias	42
Figura 18 - Log de pré-requisitos e dados de eletrocardiograma	42
Figura 19 - Log do acelerômetro	42
Figura 20 - Placa DOIT ESP32-DEVKIT-V1, que apresenta um chip ESP-WROOM-32 (Placa [...], [s.d.])	43
Figura 21 - Estrutura de arquivos dos programas do ESP32.....	44
Figura 22 - Início do código principal em Polar H10_Complete.ino	45
Figura 23 - Fim das declarações iniciais do código principal	46
Figura 24 - Função setup do programa, executado uma vez	46
Figura 25 - Função loop do programa, executada inúmeras vezes	47
Figura 26 - Função callback do escaneamento BLE	48
Figura 27 - Conexão com o servidor.....	48
Figura 28 - Função para estruturar a mensagem de saída do ESP32.....	49
Figura 29 - Estrutura do projeto da interface gráfica.....	52
Figura 30 - Recursos gráficos externos usados na interface	53
Figura 31 - Detecção de mensagens na interface	54
Figura 32 - Processamento de mensagens na interface	54
Figura 33 - Implementação do buffer circular	55
Figura 34 - Implementação do gráfico para o eletrocardiograma	56
Figura 35 - Conceitos iniciais para a aparência da interface	57
Figura 36 - Simulação da aquisição e decodificação do Polar H10 com ESP32.....	58
Figura 37 - Interface usada com o microcontrolador simulando a Polar H10	60
Figura 38 - Interface gráfica funcionando com dados reais	60
Figura 39 - Estrutura final do projeto (Polar Eletro Oy, c2023a) (Placa [s.d.]).....	62

Lista de Tabelas

Tabela 1 - Descrição dos serviços da Polar H10	27
Tabela 2 - Flags de notificação da característica de medição de frequência cardíaca	29
Tabela 3 - Exemplo de notificação da característica de frequência cardíaca.....	30
Tabela 4 - Estrutura de resposta da característica Nível da Bateria	32
Tabela 5 - Features da Polar H10 para streaming de dados	34
Tabela 6 - Requisição para obter configurações específicas de tipo de stream	34
Tabela 7 - Configuração de stream de dados de aceleração	35
Tabela 8 - Configuração de stream de dados de eletrocardiograma	36
Tabela 9 - Requisição de início de stream de dados de acelerômetro	37
Tabela 10 - Requisição de início de stream de dados de eletrocardiograma	37
Tabela 11 - Resposta do Ponto de Controle PMD para início de stream, dependendo do tipo de dado ..	38
Tabela 12 - Requisição de encerramento de stream de dados	38
Tabela 13 - Resposta do Ponto de Controle PMD para encerramento de stream de dados	38
Tabela 14 - Notificação de stream de dados de aceleração	39
Tabela 15 - Notificação de stream de dados de eletrocardiograma.....	40
Tabela 16 - Codificação de todas as grandezas medidas pelo SDK da Polar Electro Oy	72
Tabela 17 - Todos os tipos de frames de dados de aceleração.....	72
Tabela 18 - Único tipo de frame disponível de eletrocardiograma	73
Tabela 19 - Códigos de erros possíveis para a comunicação com Ponto de Controle PMD.....	73

Lista de Abreviações

ACC – Aceleração
ANVISA – Agência Nacional de Vigilância Sanitária
APP – Camada de Aplicação
ATT – Protocolo de Atributo
BLE – Bluetooth Low Energy
BPM – Batidas Por Minuto
CAGR – Taxa de Crescimento Anual Composto
ECG – Eletrocardiograma
EEG – Eletroencefalograma
EMG – Eletromiografia
EOG – Eletro-oculograma
FC – Frequência Cardíaca
FDA – *Federal Drug Administration*
FTDI – *Future Technology Devices International*
GAP – Perfil de Acesso Genérico
GATT – Perfil de Atributo Genérico
GUI – Interface Gráfica de Usuário
HMD – *Head-Mounted Display*
IA – Inteligência Artificial
IDE – Ambiente de Desenvolvimento Integrado
IIoT – Internet Industrial das Coisas
IHM – Interface Humano-Máquina
IoT – Internet das Coisas
IRR – Intervalo R-R
LED – Diodo Emissor de Luz
LGPL – *Lesser General Public License*
MLB – *Major League Baseball*
MCU – Unidade Microcontroladora
MTU – Unidade de Transmissão Máxima
NBA – *National Basketball Association*
PCB – Placa de Circuito Impresso
PMD – *Polar Measurement Data*
PFC – *Polar Features Configuration*
PPG – Fotopletismograma
PSFTP – *Polar Simple File Transfer Protocol*
RFC – *Request For Comments*
SDK – *Software Development Kit*
SIG – Grupo de Interesses Especiais
SO2 – Saturação de Oxigênio
UART – *Universal Asynchronous Receiver/Transmitter*
USB – *Universal Serial Bus*
UUID – Identificador Único Universal
VFC – Variabilidade da Frequência Cardíaca

Sumário

1. INTRODUÇÃO	1
a. Motivação	2
b. Objetivo geral	3
2. ESTADO DA ARTE	4
a. O que são dispositivos <i>wearables</i>?	4
b. Classificação de <i>wearable</i>	4
c. <i>Wearables</i>, IIoT e Indústria 4.0	7
d. Aplicação de <i>wearables</i> na medicina	8
e. Uso de <i>wearables</i> em esportes	10
f. Desafios enfrentados pelos <i>wearables</i>	10
g. Sinais biológicos	11
h. IHM	12
3. METODOLOGIA	14
a. Tipos de sinais aqusitados	14
i. Sinais cardíacos	14
ii. O sensor Polar H10	16
iii. Validação da cinta Polar H10	17
b. Tipo de tecnologia de aquisição	20
i. <i>Wearables</i> e o <i>Bluetooth</i> [®]	20
ii. Fundamentos e implementação do <i>Bluetooth</i> [®]	21
iii. Polar H10	23
iv. Desenvolvimento com o ESP32	43
c. Interface Homem-Máquina	50
i. Visualização dos dados na interface gráfica	50
4. RESULTADOS E DISCUSSÃO	58
5. CONCLUSÃO	63
REFERÊNCIAS BIBLIOGRÁFICAS	64

1. INTRODUÇÃO

A sociedade contemporânea está imersa em uma era marcada pela convergência da digitalização, automação e interconectividade, conhecida como Indústria 4.0. Nesse contexto, a incorporação de tecnologia em dispositivos *wearables* tem desencadeado uma transformação significativa na forma como os seres humanos interagem com o ambiente ao seu redor. Esses dispositivos, eletrônicos portáteis incorporados em itens de vestuário ou usados como acessórios, representam uma das inovações mais notáveis do Século XXI.

Desde os primórdios dos relógios de pulso até os atuais dispositivos multifuncionais, como relógios inteligentes, pulseiras fitness e óculos de realidade aumentada, os *wearables* desempenham um papel crucial em setores como indústria, esporte e medicina.

Na área da saúde, esses dispositivos desempenham um papel fundamental, oferecendo monitoramento contínuo de parâmetros vitais, coleta de dados biométricos e contribuindo para pesquisas médicas. No entanto, desafios como a precisão dos dados, privacidade e aceitação pelos usuários precisam ser abordados para uma integração mais ampla na prática clínica. Além disso, os *wearables* tornam-se ferramentas revolucionárias para o monitoramento remoto de pacientes, diagnóstico precoce e pesquisa clínica. A coleta de dados contínuos e não invasivos oferece benefícios substanciais para a saúde, embora desafios relacionados à segurança e validação dos dispositivos precisem ser superados (Athavale & Krishnan, 2017).

No contexto esportivo, o uso de *wearables* cresce exponencialmente, proporcionando avaliações em tempo real do desempenho atlético, monitoramento de carga de treinamento e prevenção de lesões. A integração de dados precisos desses dispositivos em modelos preditivos é crucial para otimizar a saúde e o desempenho dos atletas (Seshadri *et al.*, 2019).

Além disso, os biossinais captados pelos *wearables* desempenham um papel fundamental na compreensão da atividade fisiológica. A coleta precisa desses sinais é essencial para avaliar o estado de saúde e impulsionar avanços em tecnologias biomédicas. Métodos variados, como eletrodos, biossensores e medições de impedância, são empregados dependendo do contexto, oferecendo flexibilidade quanto à precisão e invasividade desejadas (Semmlow, 2018).

Os sinais elétricos do corpo humano são comumente capturados para diagnóstico médico, apresentando melhoras crescentes na captação, processamento e análise desses biossinais. Dispositivos portáteis e sistemas integrados, têm proporcionado monitoramento contínuo e diagnósticos personalizados, destacando-se como ferramentas valiosas na compreensão do metabolismo humano.

No âmbito da Interface Homem-Máquina (IHM), crucial para a interação entre humanos e tecnologia, o design ergonômico, usabilidade e integração tecnológica desempenham papéis essenciais. O desenvolvimento de uma IHM enfrenta desafios, como atender diferentes perfis de usuários e garantir acessibilidade. Três pesquisas representativas contextualizam o estado da arte, abordando monitoramento remoto, adaptação de medidores de ECG em ambientes extremos e sistemas de alerta para mergulhos offshore. A implementação bem-sucedida desses conceitos fortalece a confiança nas capacidades do projeto. Em síntese, a integração eficiente de biossinais e uma IHM bem projetada promovem avanços significativos na compreensão da fisiologia humana e no desenvolvimento de soluções médicas eficazes (Abo-Zahhad *et al.*, 2014; Almeida *et al.*, 2023; Barbosa *et al.*, 2021).

À medida que a tecnologia avança, espera-se que os *wearables* continuem redefinindo os limites da interação humana com a tecnologia, oferecendo oportunidades e desafios que moldarão o futuro dessa inovação. Este trabalho visa desenvolver uma estrutura versátil para a aquisição, processamento e exibição de biossinais por meio de dispositivos *wearables* para monitoramento de biossinais em cenários

diversos. Dessa forma, essa pesquisa apresenta-se como tecnologia inovadora e relevante para o âmbito da saúde.

O trabalho está estruturado em:

- “Motivação”, no qual será apresentado o estímulo para a execução deste projeto;
- “Objetivo” aborda o resultado que se deseja atingir após o cumprimento de todas as suas etapas;
- “Estado da arte” visa traçar um panorama dos wearables e suas aplicações, sinais biológicos e IHMs;
- “Metodologia” apresenta as ferramentas utilizadas para a execução do projeto;
- “Resultados e Discussão” apresenta e analisa todos os resultados obtidos durante a execução do projeto;
- “Conclusão” sintetiza as considerações finais e indica futuras rotas para a realização de trabalhos vindouros.

a. Motivação

O uso cada vez mais dominante de dispositivos vestíveis (relógios inteligentes, pulseiras, sistemas de rastreamento etc.), oferece a oportunidade de melhorar os cuidados de saúde. Ao incorporar tecnologias *wearable*, consegue-se estabelecer um monitoramento contínuo dos sinais vitais, proporcionando uma visão abrangente e em tempo real do estado de saúde dos usuários.

A coleta constante de dados biomédicos através dos *wearables* não apenas agrega uma quantidade valiosa de informações, mas também acelera o diagnóstico. Detectar precocemente padrões anômalos possibilita intervenções rápidas, melhorando a eficácia dos tratamentos e, conseqüentemente, os resultados clínicos. De acordo com Lin *et al.* (2021) aproximadamente um terço das mortes mundiais são em decorrência de doenças cardiovasculares, muitas das quais poderiam ser diagnosticadas precocemente se houvesse dispositivos de monitoramento facilitado. Isso mostra uma ampla oportunidade de aplicação e aprimoramento nessa área.

Além disso, a eficiência do sistema integrado, aliada à redução de custos e à simplificação dos processos, destaca sua relevância na evolução dos cuidados de saúde. O futuro aponta para uma abordagem mais personalizada e centrada no paciente, impulsionada pela sinergia entre *wearables* e sistemas de monitoramento avançados. O avanço da era tecnológica segue levando cada vez mais informações às pessoas, que se tornam mais conscientes da importância de seu bem-estar. Deste modo, e aliado ao aumento da utilização de dispositivos vestíveis para a população em geral permite uma mudança na forma de tratar a saúde, passando a priorizar a prevenção em detrimento do tratamento de doenças.

Este trabalho tem como foco elaborar uma estrutura versátil destinada à captura, processamento e apresentação de biosinais através de dispositivos vestíveis, com o propósito de monitorar tais sinais de forma desvinculada ao ponto de aquisição dos sinais. Sendo assim, a principal motivação deste projeto está na busca de um uso efetivo e futuro destes sinais para gerar uma base de dados mais extensa de pacientes e atletas como forma de prevenir doenças ou incidentes, por exemplo, em situações extremas. Além disso, a construção destes tipos de dispositivos permite o monitoramento em ambientes extremos, como submersos e *offshore*.

b. Objetivo geral

O objetivo principal deste trabalho é desenvolver uma estrutura versátil para a aquisição, processamento e exibição de biossinais por meio de dispositivos *wearables*, visando oferecer uma solução de monitoramento contínuo em diferentes ambientes entre o agente (usuário) e o monitorado.

i. Objetivos específicos:

- Justificar a escolha de um *wearable* adequado para a aquisição de biossinais;
- Justificar a escolha do microcontrolador utilizado para conectar o *wearable* a interface;
- Conectar o *wearable* selecionado a uma central responsável pela extração dos dados do *wearable*;
- Estabelecer um protocolo de comunicação customizado entre o microcontrolador e a interface para transmitir mensagens com os diferentes tipos de dados do *wearable*;
- Implementar o elemento de exibição;
- Abordar aspectos como confiabilidade do *wearable*, conhecimento das fontes de ruído e ergonomia, especialmente na aplicação à área da saúde;
- Projetar o sistema para ser fácil e intuitivo, visando atender a um amplo espectro de usuários.

2. ESTADO DA ARTE

A incorporação de tecnologia em dispositivos *wearables* tem conferido à sociedade contemporânea uma vasta gama de oportunidades na aquisição e análise de biossinais, contribuindo substancialmente para diversos setores, tais como a indústria, o esporte e a medicina. Essa evolução tecnológica é uma característica intrínseca à Indústria 4.0, marcada pela convergência da digitalização, automação e interconectividade, impactando significativamente na maneira como os seres humanos interagem com o ambiente ao seu redor.

a. O que são dispositivos *wearables*?

Os dispositivos vestíveis, também conhecidos como *wearables*, representam uma inovação tecnológica marcante do Século XXI. São definidos como eletrônicos portáteis capazes de serem incorporados em itens de vestuário ou usados no corpo como acessórios (e.g. relógio de pulso, óculos). As suas origens remontam aos primeiros relógios de pulso e dispositivos de acompanhamento de saúde. Alguns dos incentivos para o surgimento dos *wearables* foram o esforço para medir o desempenho atlético e o tempo de maneira mais precisa. No entanto, essa ideia só foi ampliada com o advento da era digital e dos conceitos de Indústria 4.0, do avanço da conectividade entre aparelhos e do desenvolvimento de sensores cada vez menores (Yasar, 2023).

A evolução tecnológica desempenhou um papel crucial no aparecimento e difusão dos vestíveis. Com a presença de componentes eletrônicos cada vez menores e o aprimoramento das baterias de longa duração, foi possível o desenvolvimento de dispositivos multifuncionais, integrando conectividade, conforto e estética. A era dos *smartphones* testemunhou uma sinergia sem precedentes entre aparelhos móveis e *wearables*, levando a um crescimento exponencial na adoção de itens como relógios inteligentes, pulseiras *fitness*, óculos de realidade aumentada e roupas com sensores integrados. Esses dispositivos são agora capazes de realizar uma vasta gama de funções, desde a comunicação básica até o monitoramento contínuo do estado de saúde.

Além disso, a tecnologia dos *wearables* foi totalmente incorporada pelos movimentos de IoT e IIoT, já que com o aumento de sua versatilidade e a melhora da qualidade de conexão, os dispositivos passaram a ser introduzidos na indústria de forma capilarizada. Alguns exemplos de setores influenciados pelos vestíveis são os de monitoramento, suporte, treinamento e rastreamento, aos quais sua adesão proporcionou robustez e confiabilidade (Svertoka *et al.*, 2021).

Atualmente os *wearables* estão presentes na vida cotidiana, auxiliando indivíduos em diversos pontos, como na saúde, nos exercícios físicos e no bem-estar. Os dispositivos fornecem *feedback* em tempo real sobre atividades diárias, padrões de sono, frequência cardíaca e muitos outros parâmetros vitais. Essa capacidade de monitoramento contínuo transformou a forma como cada um interage com a tecnologia, promovendo uma maior consciência e proatividade em relação ao cuidado pessoal. Além disso, a facilidade de uso e a capacidade de sincronização com outros dispositivos digitais ampliam sua funcionalidade e geram um maior apelo ao usuário (Smith, 2019; Yasar, 2023).

b. Classificação de *wearable*

Não existe uma norma sobre como categorizar vestíveis. Um dos motivos é o fato de muitos deles pertencerem a diversos grupos ao mesmo tempo. Após uma vasta pesquisa foram traçadas as categorias presentes nesse projeto, levando em conta popularidade, incentivo financeiro e área de uso. Mesmo assim, alguns exemplos se encaixam em mais de um conjunto.

- Entretenimento

A incorporação dos *wearables* na rotina também se reflete no entretenimento, como nos jogos eletrônicos. Nesse meio, as telas montadas na cabeça (HMD, da sigla em inglês *Head-Mounted Display*) se desenvolvem rapidamente na direção da realidade virtual e realidade aumentada. Ambas proporcionam experiências multissensoriais ao usuário, sejam visuais, auditivas ou olfativas, o que promove a sua imersão total. Dessa forma, essa tecnologia vem sendo apresentada principalmente como nova forma de divertimento, mas possibilita avanços no meio profissional. Com o seu uso, podem-se efetuar treinamento de indivíduos, simulando tarefas complexas e perigosas em ambientes controlados. Assim, a chance de erro em operações no mundo real é reduzida, e falhas podem ser detectadas antes que sistemas sejam implementados na prática. Alguns exemplos de vestíveis dessa categoria são o *Oculus Rift*, o *Samsung HMD Odyssey*, o *HTC Vive*, o *Valve Index*, entre outros aparelhos disponíveis no mercado (Angelov et al., 2020).

Vestíveis mais acessíveis para a população do que os HMDs são os *hearables*, que dividem-se em duas principais vertentes – os fones de ouvidos com e sem fio; e os aparelhos auditivos, que são focados na categoria de saúde, abordada posteriormente. Desde sua concepção inicial, os fones se consolidaram como elementos indispensáveis no cotidiano de indivíduos envolvidos com tecnologia. A adesão contínua desses usuários a tais dispositivos demonstra um crescimento progressivo e constante, principalmente atrelado à indústria da música (Smith, 2019).

A Figura 1 abaixo, por exemplo, mostra o número de vendas unitárias dos fones de ouvido sem fio (*wireless*) no mundo, de 2018 a 2021 (em unidades de milhão).

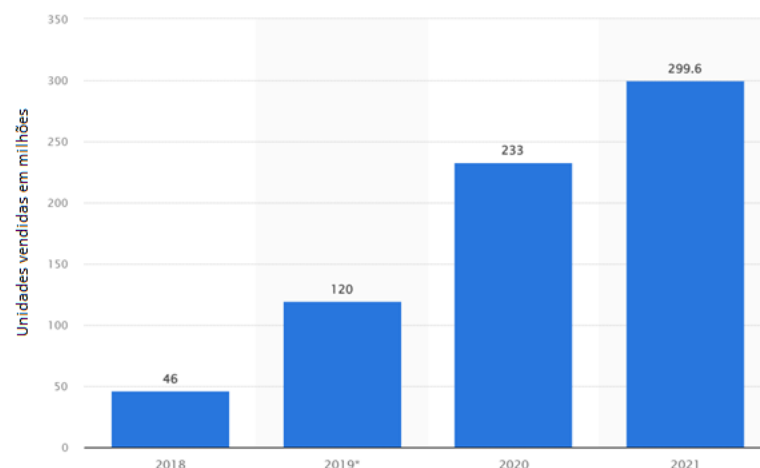


Figura 1 - Vendas internacionais de fones wireless de 2018 a 2021 (adaptado de Statista, c2023)

Os óculos inteligentes são outro item que também pode ser enquadrado na categoria de entretenimento. Operam no limite entre o mundo físico e o virtual, com óculos inteligentes (muitos dos quais também possuem o recurso da realidade aumentada) o usuário pode, por exemplo, andar na rua enquanto vê na sua frente recomendações de restaurantes, lojas, mapas com rotas para melhor locomoção, entre outros dados essenciais. Isso mostra que o seu uso difere dos HMDs, sendo mais focados em praticidade.

Dentre estes dispositivos de mercado, estão os Apple Vision Pro, desenvolvidos pela empresa de tecnologia Apple®; os HoloLens, da multinacional americana Microsoft®; os Ray-Ban® Stories, desenvolvidos em parceria pela empresa Meta® e a EssilorLuxxotica®; e os Amazon® Echo Frames, criados pela Amazon®. A lista segue por muitos nomes, tanto de inovações já disponíveis no mercado, como produtos ainda em desenvolvimento (Allison, 2023; Smith, 2019).

Para encerrar, entram ainda na classificação os sensores de movimento, como os *XSens*®. Usados na criação de jogos, na jogabilidade em si e até mesmo na concepção de animações e efeitos especiais do cinema. A chegada desses aparatos mudou de maneira definitiva o conceito de entretenimento.

- Moda

No setor da moda surge o objetivo de não só conectar os *wearables* às pessoas, mas de fazer isso considerando a estética como uma das prioridades. Além dos conhecidos *smartwatches* e *sportwatches*, os quais serão explorados mais a fundo nos subtópicos seguintes, a indústria da moda e o setor tecnológico se uniram para apresentar uma solução para indivíduos que desejam usufruir dos benefícios de dispositivos de monitoramento não-invasivos sem renunciar à aparência. As joias inteligentes, por exemplo, oferecem vantagens como rastreamento de atividades físicas, meditação guiada, alertas de telefone avisando sobre eventos e mensagens, suporte para aplicativos, entre outros.

Abaixo, imagens da marca americana *RINGLY*, focada, segundo a empresa, na venda de joias inteligentes para mulheres com o intuito de levá-las a uma vida mais leve e feliz, sem deixar de lado o bem-estar pessoal.

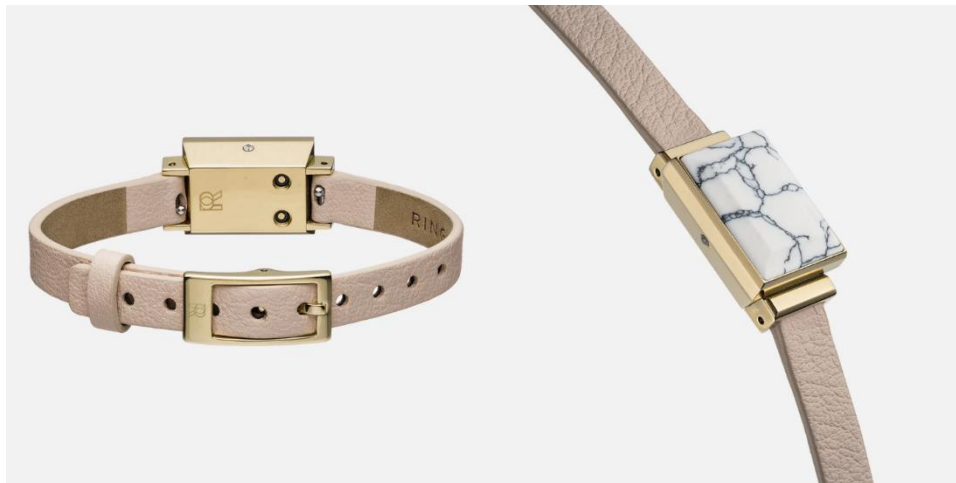


Figura 2 - Bracelete smart da marca *RINGLY* (extraído de Ringly, c2023)

Cada vez mais, roupas integradas à tecnologia vêm surgindo no mercado. Além de tecidos inteligentes com funções diversas que não estão integrados a uma rede – um assunto que não será aprofundado neste trabalho – existem hoje roupas capazes de se conectarem a telefones e computadores, ou que já contêm em si mesmas um sistema próprio, como, por exemplo, as roupas ou sapatos de LED.

Inovando nessa área, marcas como a *CuteCircuit* apresentam atualmente produtos antes nunca pensados. A camisa *HugShirt*, criada em 2002, é um exemplo de como os têxteis inteligentes são capazes de mudar a perspectiva das pessoas sobre conceitos como a distância. Com tecido *smart* equipado de micro atuadores e conectividade a um aplicativo, o objetivo é replicar a experiência de um abraço a um indivíduo localizado em quaisquer partes do mundo (Cutecircuit, c2023).

- Militar

Segundo a plataforma *Exactitude Consultancy*:

"Espera-se que o mercado global de *wearables* militares cresça 7.1% CAGR de 2022 a 2029. Espera-se que atinja acima de US\$ 8.34 bilhões até 2029, contra US\$ 4.5 bilhões em 2020." (Exactitude Consultancy, 2022)

Atualmente, os Estados Unidos da América seguem sendo quem mais alimenta o mercado de dispositivos vestíveis na área militar. Em 2021, por exemplo, o país foi responsável pela compra de 45% dos dispositivos, contra 55% (somados) do resto do mundo. Dentro dessa categoria, entram, fora os próprios têxteis inteligentes já citados, câmeras corporais, uniformes capazes de informar rastreamento e localização dos soldados, dispositivos audíveis para comunicação, óculos militares inteligentes, exoesqueletos, bem como os próprios *smartwatches*, agora adaptados para o campo de batalha.

- **Atividades físicas e saúde**

No campo da saúde os *wearables* vêm sendo cada vez mais difundidos, desde *smartwatches* e *fitnesswatches*, até monitores de glicose, eletrocardiógrafos e faixas peitorais. Em se tratando da importância dos *wearables*, esta fica particularmente evidente na coleta de dados biométricos. Com sensores mais precisos, permitem uma medição contínua e não invasiva de informações que anteriormente só poderiam ser obtidas em ambientes clínicos e de forma muitas vezes desconfortável para os pacientes. Com o uso desses novos dispositivos, tornou-se mais simples o monitoramento de condições crônicas, como a diabetes e a hipertensão, aprimorando, assim, o bem-estar individual (Iqbal *et al.*, 2021). Além disso, eles também contribuem para grandes conjuntos de dados (*big data*) que podem ser utilizados em novas pesquisas. Os profissionais de saúde, por exemplo, hoje têm a possibilidade de, através desses dados coletados, obter informações mais precisas a respeito da saúde de seus pacientes e dessa forma desenvolver tratamentos individualizados.

Em estudo voltado para a saúde (Chiang *et al.*, 2021), tinha-se como objetivo investigar a relação entre a pressão arterial elevada e o estilo de vida das pessoas para, assim, fornecer soluções personalizadas. Foi feito então, um experimento controlado com 25 pacientes, todos com pressão arterial elevada ou hipertensão em estágio I (hipertensão acima de 130 por 90 mmHg e abaixo de 160 por 100 mmHg), e que não tomavam medicamentos para tratar da condição crônica. Durante três meses consecutivos, os participantes do estudo tiveram seus dados coletados por monitores de pressão arterial domiciliares e usaram dispositivos vestíveis de rastreamento de atividade. Ao fim do estudo, foi constatado que os participantes do grupo experimental reduziram suas pressões arteriais em 3,8 e 2,3 mmHg para a pressão sistólica e diastólica, em comparação com a diminuição de 0,3 e 0,9 mmHg para os participantes sem recomendações. Isso se deu após terem seus atributos de estilo de vida mais importantes identificados e de terem recebido recomendações precisas para melhorar a pressão arterial individual. A validação da tese provou não só o que os autores intencionavam, mas também foi um estudo valioso para mostrar como os *wearables* vêm ganhando um espaço cada vez mais importante no campo da medicina.

Contudo, a integração desses dispositivos na prática médica diária ainda enfrenta desafios. Questões como a precisão de dados, a privacidade, a segurança da informação e a aceitação pelos usuários são preocupações que precisam ser consideradas. Mesmo com o surgimento de novos modelos regulatórios como o “Programa de Pré-certificação de Software de Saúde Digital” (Pre-Cert), o uso dos *wearables*, sobretudo na medicina, segue em um embate. Entre especialistas, um dos fatores para a hesitação é que não há um consenso sobre a forma de implementação e uso desses dispositivos em ambientes de pesquisa, o que pode acarretar diferenças no resultado dos dados coletados. A padronização desses dados, para que sejam úteis em contextos clínicos, é um aspecto crítico que requer colaboração entre desenvolvedores de tecnologia, profissionais de saúde e reguladores (Digital, 2022).

À medida que a era digital avança, é necessário seguir explorando e entendendo as oportunidades oferecidas por esses dispositivos inovadores. Entretanto, também é necessário encontrar formas de solucionar os problemas advindos desse processo a fim de seguir usando o melhor dos *wearables* em prol do avanço pessoal e coletivo.

c. Wearables, IIoT e Indústria 4.0

O movimento de Indústria 4.0, também conhecido como a Quarta Revolução Industrial, é caracterizado pela automação inteligente, pela integração de sistemas e pelo uso extensivo da tecnologia de informação com o objetivo de melhorar os processos industriais. Iniciada na primeira metade do Século XXI, representa a transformação digital das práticas de manufatura e produção, impulsionada por inovações

tecnológicas tais quais a Internet das Coisas (IoT), Internet Industrial das Coisas (IIoT), *big data*, sistemas ciber-físicos e Inteligência Artificial (IA). A Indústria 4.0 tem como objetivo digitalizar a produção, criar "fábricas inteligentes" onde os sistemas cibernéticos monitoram os processos de chão de fábrica e gerar uma cópia virtual de um objeto real para tomar decisões descentralizadas e autônomas. A meta é, com isso, quebrar o paradigma de construtividade (Aires *et al.*, 2017).

Segundo Meister (1999) nas décadas de 80 e 90, diversas empresas investiram em Universidades Corporativas, que tinham o intuito justamente de capacitar os trabalhadores focando em expandir seus conhecimentos. Além disso, conforme expresso por Aires *et al.* (2017):

"Davenport e Prusak (1998) e Teixeira Filho (2000) afirmaram que a verdadeira vantagem competitiva está na capacidade e na velocidade do aprendizado das pessoas nas organizações, logo, a gestão do capital intelectual faz-se necessária."

Integrados nesse contexto, os *wearables* se tornaram uma parte essencial da Indústria 4.0, especialmente quando conectados à IoT. Usados para diversas finalidades, desde monitorar a saúde e a segurança dos trabalhadores até otimizar a eficiência operacional, eles coletam, através de sensores embutidos, dados que podem ser compartilhados instantaneamente com outros dispositivos e sistemas ao longo da cadeia de produção, criando uma integração entre o operador humano e as máquinas. Essa coleta de dados em tempo real também registra variedades de métricas (como a localização dos trabalhadores); detecta gases perigosos; aponta condições ambientais adversas; identifica padrões; prevê tendências; entre outros muitos exemplos. Tais avanços otimizam a tomada de decisões para manutenção preditiva, controle de qualidade e gestão da cadeia de suprimentos (Barbosa *et al.*, 2022).

Portanto, ao equipar os trabalhadores com esses vestíveis, as empresas conseguem reduzir o tempo de inatividade das máquinas, aumentar a produtividade geral do setor e melhorar a ergonomia no local de trabalho. Para citar exemplos, um *wearable* consegue alertar um operador sobre a necessidade de ajustar uma configuração ou realizar manutenção preventiva em um equipamento, reduzindo assim as paradas não planejadas. Um outro fator interessante é que, utilizando esses dispositivos, diversas ações tornam-se menos suscetíveis a erros humanos. Assim, integrando-os a sistemas de aprendizado inteligentes, o algoritmo de processamento e análise de dados passa a estar em contínuo aprimoramento, fator que resulta em uma eficiência operacional crescente.

Com isso, conclui-se que esses aparatos são componentes vitais da Indústria 4.0 por possibilitarem a transição para práticas de trabalho mais inteligentes e conectadas. À medida que a tecnologia avança, espera-se que sua adoção se expanda ainda mais, com novas aplicações emergindo para fortalecer a sinergia entre humanos e máquinas em um ambiente industrial progressivamente automatizado. A integração dos *wearables* com a IoT e as tecnologias associadas à Indústria 4.0 é, portanto, um campo de oportunidade e crescimento e promete redefinir os limites da produção industrial moderna.

d. Aplicação de *wearables* na medicina

Na área médica, os *wearables* têm se revelado uma ferramenta revolucionária para o monitoramento de pacientes e para o diagnóstico remoto. A capacidade de coletar dados vitais, como a frequência cardíaca, de forma contínua e não invasiva, oferece benefícios substanciais para a saúde. Instituições líderes como a *Clínica Mayo* e o *Hospital John Hopkins* estão adotando-os para rastrear sinais vitais em pacientes com condições crônicas e aqueles em fase de recuperação pós-operatória, oferecendo um perfil abrangente e facilitando uma experiência de saúde personalizada (Eddy, 2021). Além disso, também é possível a detecção precoce de problemas. Esses dispositivos têm mostrado sua eficácia ao coletar múltiplas observações por segundo, integrando os dados do paciente com aplicativos de bem-estar e sistemas de registros eletrônicos de saúde (Dinh-Le *et al.*, 2019).

Estudos em universidades renomadas, como a *Universidade de Stanford*, indicam melhorias no controle glicêmico em pacientes diabéticos que utilizam monitoramento contínuo de glicose, reduzindo significativamente a incidência de episódios hiperglicêmicos (Steakley, 2015). Além disso, o acompanhamento de condições crônicas como insuficiência cardíaca e asma tem sido associado a uma melhoria na qualidade de vida das pessoas e a diminuição na frequência de readmissões hospitalares (Mualem *et al.*, 2022; Wadas, 2022).

A digitalização da saúde traz uma necessidade de profissionais que estejam preparados para lidar com as novas tecnologias. É preciso auxiliar não apenas esses profissionais, como também os pacientes, futuros usuários desses dispositivos.

Com o crescimento dos *wearables* voltados para o monitoramento de biossinais, os usuários podem armazenar dados sobre sua saúde 24h por dia. A partir dessa disponibilidade de medição em grande escala, são construídos banco de dados (*datasets*) de grande porte, úteis para o estudo de diversas condições de saúde. Apesar de existir certa confidencialidade de alguns dos dados de biossinais coletados, eles ainda podem ser usados por redes de hospitais de forma anônima para diversas aplicações, beneficiando a todos (Graña Possamai *et al.*, 2020).

A qualidade e confiabilidade dos sinais adquiridos por dispositivos vestíveis e instrumentos médicos estão cada vez mais próximas. A alta disponibilidade destes dispositivos no mercado de diversas origens faz necessária a validação rigorosa de cada *wearable* que se deseje aplicar na área da saúde (Kubota, 2021). Deste modo, em outras palavras, a precisão e validação dos *wearables* é essencial para sua aplicabilidade na prática clínica, conforme regulamentações da *Federal Drug Administration* (FDA) e da Agência nacional de Vigilância Sanitária (ANVISA), que garantem a segurança e eficácia desses dispositivos (Kinney, 2022). A pesquisa do *Instituto Politécnico de Milão*, por exemplo, aborda essa questão e sustenta a precisão dos vestíveis, comparando favoravelmente os dados de frequência cardíaca obtidos a partir deles com os eletrocardiogramas padrão (Dinh-Le *et al.*, 2019).

A coleta distribuída de dados proporciona um repositório de pesquisa e base de treinamento que, se usado da forma correta, consegue ajudar na detecção precoce e no diagnóstico de doenças. É possível traçar um perfil personalizado para cada usuário com dados suficientes, o que não só contribui para o trabalho dos profissionais envolvidos como também favorece a implementação de *softwares* de previsão para lidar com diagnóstico e tratamentos (Tana *et al.*, 2017).

Por outro lado, a segurança dos dados enviados para qualquer tipo de nuvem torna-se uma preocupação, pois o usuário deixa de ser o único a possuí-los. Uma vez que ele não mais os controla, ficam suscetíveis à venda para outras empresas, em alguns casos ilegalmente.

É imprescindível seguir padrões rigorosos para validação e uso de *wearables* na medicina, pois dados incorretos podem levar a diagnósticos equivocados, colocando a saúde dos usuários em risco. Tais resultados errôneos estão intrinsecamente relacionados com os métodos de medição e do ruído de medição envolvido, feitos sem uma precisão estabelecida. O fato de os dispositivos selecionados para a tarefa serem portáteis e o menos invasivo possível também é um fator que pode aumentar as chances de imprecisão (Tana *et al.*, 2017).

O monitoramento de biossinais permite a aquisição de dados de pacientes para acompanhar tratamentos, progressão de doenças e com o benefício de não precisar estar em uma clínica. Pacientes podem ser acompanhados de suas casas, e há disponibilidade de dados dos mais diversos lugares. Além disso, a situação é favorável especialmente para indivíduos com dificuldades de locomoção (Graña Possamai *et al.*, 2020).

Casos de estudo publicados no *Journal of Medical Internet Research* e na *Nature Medicine* ilustram o papel vital dos *wearables* em detectar condições não diagnosticadas, como arritmias cardíacas, e monitorar a progressão de doenças como o Mal de Parkinson em ensaios clínicos, fornecendo *insights* valiosos sobre a eficácia dos tratamentos (Mualem *et al.*, 2022).

À medida que a tecnologia avança, a integração dos dados coletados por *wearables* com sistemas eletrônicos de saúde promete um gerenciamento mais proativo e centrado no paciente. Espera-se que os vestíveis ofereçam funcionalidades ampliadas, como a detecção de biomarcadores para várias condições e a capacidade de administrar medicamentos automaticamente (Kinney, 2022).

A adoção desses dispositivos abre novas fronteiras na medicina, melhorando a qualidade de vida e revolucionando tanto o gerenciamento de doenças crônicas quanto a pesquisa clínica.

e. Uso de *wearables* em esportes

A adoção dos vestíveis no mercado esportivo tem crescido exponencialmente. Atualmente, são milhões de dispositivos gerando dados em contexto de treinamento e performance. A qualidade dos dados e a padronização desses dispositivos são centrais para garantir sua eficácia e utilidade na área do esporte e da medicina esportiva. Trata-se de um campo dinâmico, com desenvolvimento em múltiplos níveis. Os *stakeholders* estão buscando facilitadores e contornando barreiras para a participação em padrões de diretrizes, sem a necessidade de regulamentações prescritivas, mas buscando colaboração entre empresas, consumidores e pesquisadores (Ash *et al.*, 2021).

O monitoramento do desempenho atlético abrange a avaliação de cargas internas e externas de treinamento, possibilitando avaliações em tempo real dos níveis de fadiga e aptidão dos atletas. A carga interna de treinamento pode ser avaliada por meio de percepções de esforço da sessão e frequência cardíaca, enquanto a carga externa é quantificada usando dispositivos vestíveis que contêm GPS e acelerômetro triaxial, como o *PlayerLoad™*.

O controle e a prevenção de lesões feitos através de *wearables* são baseados na capacidade de monitorar e ajustar a carga de treinamento de cada atleta a fim de minimizar a incidência de lesões nos tecidos moles. A tecnologia de acelerômetros é usada para determinar o gasto energético e o controle do movimento e equilíbrio durante a atividade esportiva. Calvert (1976) e Morton (1997) propuseram modelos para balancear a carga de treinamento com o intuito de otimizar o desempenho, diminuir a ocorrência de lesões e a fadiga dos esportistas. Ressalta-se que, nessa definição de fadiga, inclui-se perdas de desempenho induzidas por exercício devido a fatores tanto fisiológicos como psicológicos (Seshadri *et al.*, 2019).

Pesquisas recentes têm investigado a exatidão da contagem de passos entre aplicativos e *wearables*, uma medida frequentemente usada para derivar outras métricas de atividade física, como distância percorrida e calorias gastas. Tais avanços nos vestíveis estão transformando o treinamento esportivo, com aplicações práticas variando de esportes de equipe a atividades individuais de alta performance. A revisão publicada pela revista acadêmica *Digital Medicine* fornece uma visão abrangente da literatura científica e dos dispositivos comercialmente disponíveis, ressaltando sua importância e potencial nessa área (Seçkin *et al.*, 2023; Seshadri *et al.*, 2019).

Com o avanço das tecnologias *wearable*, a ciência do esporte tem ampliado a utilização de sensores para monitorar uma gama de parâmetros biomecânicos, fisiológicos e bioquímicos dos atletas. A integração de dados precisos de sensores vestíveis em modelos preditivos está se tornando um campo de estudo crescente e necessário para monitorar a segurança e o desempenho dos atletas. Essa integração de dados é crucial para reduzir a carga de lesões e otimizar a carga de trabalho, ajustando as práticas de treinamento para melhorar a saúde e o desempenho dos atletas. O uso de sensores como o *Catapult OptimEye S5*, *Zebra RFID tag* e *Zephyr BioHarness* tem fornecido medidas quantificáveis do esforço dos atletas, conhecido como "carga de trabalho", permitindo que algumas equipes profissionais demonstrem uma relação entre a carga de trabalho e as taxas de lesões (Seshadri *et al.*, 2021).

Além disso, a aplicação de aprendizado de máquina tem surgido em ligas esportivas como a MLB (*Major League Baseball*) e a NBA (*National Basketball Association*), com ferramentas públicas como o *Statcast* rastreando dados de desempenho dos jogadores e fornecendo insights sobre a produção futura baseada em métricas de desempenho (Seshadri *et al.*, 2021).

f. Desafios enfrentados pelos *wearables*

A integração dos *wearables* na saúde revolucionou o monitoramento contínuo de biossinais, oferecendo uma gama de dados inestimáveis para o diagnóstico e gestão terapêutica. No entanto, a adoção desses dispositivos enfrenta desafios substanciais em relação a precisão, validade dos dados, privacidade, segurança, limitações técnicas e barreiras de adoção pelo usuário, bem como desafios regulatórios e normativos que precisam ser superados para maximizar seu potencial clínico e comercial (Stuart *et al.*, 2022).

A precisão e a validade dos dados coletados através de *wearables* são fundamentais para sua eficácia clínica. Fatores como dados incompletos, incorretos e heterogêneos podem levar a desafios de qualidade intrínseca dos dados, resultando em potencial de sobre-detecção de sinais benignos não clínicos, causando diagnósticos errados, exames desnecessários e nervosismo ao paciente (Tana *et al.*, 2017).

As preocupações com a privacidade e segurança dos dados se intensificam à medida que os *wearables* se tornam cada vez mais presentes no monitoramento da saúde. A ameaça de que os dados pessoais dos indivíduos sejam compartilhados com aplicativos e serviços de terceiros sem o conhecimento ou consentimento dos mesmos suscita preocupações, gerando desconfiança em torno do uso desses dispositivos (Cilliers, 2019; Zhang *et al.*, 2020; Broll *et al.*, 2021; Silva, 2023).

Do ponto de vista técnico, as limitações de *hardware* e *software*, bem como a falta de interoperabilidade dos vestíveis com as redes de comunicação existentes, representam barreiras à adoção pelos usuários. A usabilidade e a experiência do usuário com essa nova tecnologia são críticas para a aceitação do dispositivo, particularmente entre os idosos, que podem se beneficiar de seu uso, mas que frequentemente encontram barreiras psicológicas para sua adoção (Broll, 2021; Gu & Wei, 2021; Farivar *et al.*, 2020).

g. Sinais biológicos

Os biossinais, ou sinais biológicos, constituem elementos fundamentais na compreensão e monitoramento da atividade fisiológica dos organismos. Esses sinais, muitas vezes representados por variáveis elétricas, mecânicas ou químicas, representam uma série de oportunidades no campo da medicina, uma vez que com a sua análise podem ser tiradas diversas conclusões. A mensuração precisa e interpretação dos biossinais são essenciais para a avaliação do estado de saúde de um indivíduo e para o desenvolvimento de tecnologias biomédicas avançadas.

Existem diferentes métodos para medi-los, dependendo da situação em que se deseje adquirir os dados. Assim, o objetivo da coleta está atrelado com o nível de precisão e o grau com que o procedimento será invasivo. Por exemplo, em condições extremas como a de mergulho, o usuário não pode ter os seus movimentos restringidos. Por outro lado, em ambientes clínicos, há mais liberdade para medições e pontos de coleta, consequentemente possibilitando dados mais precisos (Barbosa *et al.*, 2021).

Sinais de natureza elétrica são capturados de maneira geral com eletrodos na região de interesse do corpo humano. Nesse caso, não pode haver pelos onde forem colocados e é comum a aplicação de gel condutor para melhorar a aquisição. Outra das formas de coleta se dá por biossensores, que têm sido empregados para monitorar níveis específicos de biomarcadores em fluidos biológicos, contribuindo para a detecção precoce de patologias e o acompanhamento de tratamentos médicos. Ainda existem métodos que se baseiam na impedância da pele e na movimentação de partes do corpo, o que são usados para medir a respiração e deslocamento (Lin *et al.*, 2021).

O funcionamento do corpo humano depende de uma série de processos complexos, muitos dos quais podem ser monitorados e processados. Dependendo da área de interesse, esses sinais podem ter diferentes naturezas, mas todos precisam ser passados por biotransdutores, de modo que sejam convertidos para algum sinal elétrico analógico e enfim digital, para poder ser interpretado por computadores (Semmlow, 2018).

Dentre os biossinais mais comumente investigados, destacam-se os sinais elétricos relacionados à atividade bioelétrica dos órgãos, como os potenciais elétricos cardíacos e cerebrais. A eletrocardiografia (ECG) e a eletroencefalografia (EEG) são exemplos de técnicas que exploram esses biossinais para diagnóstico médico e pesquisa científica. Tem-se alguns biossinais comuns de serem adquiridos e a forma de obtenção (Semmlow, 2018):

- Eletrocardiograma (ECG): Registra a atividade elétrica do coração, medindo a variação dos potenciais elétricos ao longo do tempo na superfície corporal. É obtido através de eletrodos colocados na pele, geralmente no peito, membros superiores e inferiores;

- Eletromiografia (EMG): Mede a atividade elétrica dos músculos durante contrações e repouso. Eletrodos de superfície ou agulhas são inseridos nos músculos, registrando a variação no potencial elétrico gerado pelas fibras musculares;
- Eletro-oculograma (EOG): Monitora os movimentos oculares através da detecção dos potenciais elétricos gerados pelos movimentos dos olhos. Para captar esses sinais, eletrodos são colocados em torno dos olhos;
- Eletroencefalograma (EEG): Aquisita a atividade elétrica do cérebro, capturando a variação nos potenciais elétricos gerados pelas células cerebrais. Eletrodos são colocados no couro cabeludo para registrar essas oscilações;
- Saturação de Oxigênio (SO₂): A medição da SO₂ avalia a quantidade de oxigênio ligada à hemoglobina no sangue. Isso é geralmente feito através de um oxímetro de pulso, que utiliza luz infravermelha para calcular a proporção de oxigênio com base na parcela refletida da luz;
- Fotopletismografia (PPG): Mede as variações no volume sanguíneo periférico, geralmente em um dedo ou lóbulo da orelha. Isso ocorre por meio de um feixe de luz que atravessa o tecido e é absorvido pela hemoglobina, fornecendo informações sobre o pulso e a perfusão sanguínea;
- Frequência Respiratória: É determinada através da contagem do número de respirações por minuto. Isso pode ser realizado observando os movimentos do tórax ou abdômen, ou usando sensores de fluxo de ar. Pode ser estimada a partir de outros sinais também;
- Glicose: A medição da glicose envolve a coleta de uma pequena amostra de sangue (por punção digital) e a análise usando um dispositivo que detecta os seus níveis.

Dispositivos portáteis e sistemas integrados têm emergido como ferramentas valiosas para o monitoramento contínuo de biossinais em ambientes ambulatoriais, proporcionando uma compreensão mais abrangente das variações fisiológicas ao longo do tempo. A integração de inteligência artificial e aprendizado de máquina também tem desempenhado um papel relevante na interpretação de biossinais, abrindo novas perspectivas para o diagnóstico personalizado e a medicina de precisão. Essas inovações convergentes têm fortalecido o campo dos biossinais, promovendo avanços substanciais na compreensão da fisiologia humana e no desenvolvimento de soluções médicas mais eficazes.

h. IHM

Uma Interface Homem-Máquina (IHM) é um componente tecnológico que proporciona a interação entre seres humanos e máquinas, sistemas computacionais ou dispositivos eletrônicos. Essa interface serve como o ponto de contato onde os usuários enviam comandos e recebem feedback do sistema. O objetivo fundamental de uma IHM é tornar a comunicação eficaz e intuitiva entre humanos e a tecnologia, tornando a interação mais acessível e compreensível. Seu desenvolvimento consiste em um processo complexo que envolve uma combinação de design ergonômico, usabilidade e integração tecnológica para otimizar a comunicação entre o usuário e o sistema (Abo-Zahhad *et al.*, 2014; Barbosa *et al.*, 2021).

A implementação de uma IHM cobre vários componentes essenciais, cada um contribuindo para a criação de um produto coeso e eficaz. O design gráfico desempenha um papel crucial na representação visual das informações, garantindo clareza e compreensão. Além disso, através da interatividade se deve proporcionar uma experiência dinâmica, integrando elementos como botões, menus e feedbacks sensoriais, por exemplo, sonoros. A usabilidade é um aspecto crítico, demandando uma compreensão profunda das necessidades e habilidades dos usuários finais. A ergonomia também desempenha um papel vital, assegurando que a interface seja adaptada às características físicas e cognitivas dos usuários, minimizando fadiga e erros.

O desenvolvimento de uma IHM enfrenta diversos desafios, incluindo a diversidade de usuários e contextos de uso, e são abordados a seguir. A criação de interfaces que atendam a diferentes perfis de usuários, desde iniciantes até especialistas, exige uma abordagem flexível e personalizável (Barbosa *et*

al., 2021). Além disso, considerações de acessibilidade são essenciais para garantir que a interface seja utilizável por pessoas com diferentes habilidades e necessidades. Por fim, quando existir principalmente a conectividade com a nuvem deve-se garantir a segurança dos dados de usuários, para que toda a sua trajetória e processamento seja feita de forma privada.

Foram selecionadas três pesquisas acadêmicas, representando o estado da arte, que implementam objetivos relacionados ao deste projeto, a título de contextualização e direcionamento para o presente texto. Serão retomadas na seção de resultados, comparando-as com o que foi desenvolvido.

O trabalho realizado por Abo-Zahhad *et al.* (2014) implementou um sistema para monitorar os dados de um ECG Holter com 5 eletrodos e enviar alertas *online*, via dados telefônicos ou redes de internet sem fio. Isso permite que médicos e familiares possam ser contatados imediatamente e ações importantes sejam tomadas a tempo. Isso é especialmente adequado para a população idosa, que ao aumentar nos últimos anos necessita de acompanhamento constante.

Foi estudado um caso de aplicação em trabalho *offshore* de sistema IHM para monitoramento de biossinais. Em um cenário alterado pelo contexto da Covid-19, o monitoramento à distância se fez necessário. Assim, buscou-se adaptar um medidor de eletrocardiograma padrão ouro, e torná-lo portátil incorporando-o em uma maleta, junto a um *notebook* para ler os seus dados. As leituras são enfim transferidas sem fio ao computador de médicos envolvidos no monitoramento. Isso possibilitou um acesso facilitado e um diagnóstico assertivo à distância, visto que os dados possuem confiabilidade (Barbosa *et al.*, 2021).

No estudo, o objetivo foi desenvolver um sistema de alertas com IHM para monitorar a condição de saúde em mergulhos *offshore*. Esse trabalho em condições extremas necessita de sistemas de segurança para prevenir acidentes, interpretando os sinais biológicos do mergulhador e interrompendo a atividade caso necessário, antes que ocorram complicações. No caso, os sinais escolhidos foram a frequência cardíaca, intervalo R-R, temperatura, acelerômetro e giroscópio. Foram utilizados métodos da biologia para verificar se os dados cardíacos assumiam valores considerados normais, porém, os diferentes metabolismos dos profissionais proporcionaram um desafio para resposta do programa. Assim, dados adicionais do usuário foram requeridos, e algumas otimizações no sistema incorporando-os permitiram uma maior precisão nas suas análises (Almeida *et al.*, 2023).

Diante disso, o desenvolvimento de uma Interface Homem-Máquina é um processo multidisciplinar que combina princípios de design, psicologia do usuário e conhecimentos tecnológicos. Uma IHM cuidadosamente projetada pode melhorar a eficiência, segurança e experiência do usuário em uma variedade de contextos.

3. METODOLOGIA

Nesta seção é apresentada a metodologia usada no estudo de decodificação e sistema IHM de processamento de biosinais utilizando microcontroladores. A princípio, são descritos os tipos de sinais adquiridos, incluindo sinais cardíacos, o sensor Polar H10 e a sua validação. A seguir, são descritos os tipos de tecnologia de aquisição, introduzindo o protocolo Bluetooth®, fundamentos e implementações do Bluetooth®, integração com a Polar H10 e o desenvolvimento com o ESP32. Finalmente, tem-se a descrição da interface HM com a visualização dos dados.

a. Tipos de sinais adquiridos

i. Sinais cardíacos

O coração é o músculo mais importante do corpo humano (Ministério da saúde, 2004). Ele funciona como uma câmara oca com dois compartimentos que bombeiam o sangue para o restante do organismo. Sendo mais ou menos do tamanho de um punho fechado e pesando entre 250g e 300g nos adultos, ele fica localizado logo acima do diafragma, no osso do tórax, entre os dois pulmões (Varella, 2014). Seu funcionamento é explicado, nas palavras de Mariana Varella (2014):

"O sangue venoso e o arterial são separados por uma membrana vertical, que divide o coração em dois lados: o direito, composto de um átrio e um ventrículo, e o esquerdo, composto de outro átrio e outro ventrículo. Os dois átrios são câmaras superiores que recebem o sangue venoso (desoxigenado); os dois ventrículos bombeiam o sangue arterial (oxigenado). A divisão horizontal é feita por válvulas atrioventriculares: a mitral, que divide o lado esquerdo em dois, e a tricúspide, que divide o lado direito."

Entendendo a composição do coração humano, é importante explicar ainda que sístole é a contração das câmaras inferiores (ventrículos) que ejetam o sangue e diástole é o relaxamento das câmaras superiores (átrios) que recebem o sangue. A cada ciclo de bombeamento é dado o nome de batida cardíaca. Assim funcionam as batidas do coração.

Para compreender o foco deste trabalho, é necessário também saber o que é um eletrocardiograma (ECG) e como interpretá-lo. Consiste em um gráfico que ilustra os impulsos elétricos do coração, e a sua estrutura será explorada a seguir (Figura 3). Segundo José Feldman e Gerson P. Goldwasser (2004):

"Ativação ou despolarização cardíaca, em condições normais, tem origem no nódulo sinusal (nódulo de Keith-Flack), região do marca-passo cardíaco, localizado no átrio direito, sendo esta a primeira área do coração a se despolarizar. O estímulo alcança, em sequência, o átrio esquerdo, o nódulo atrioventricular (nódulo de Aschoff-Tawara), o feixe de His e seus ramos (esquerdo e direito), a rede de Purkinje, os ventrículos e, por fim, se extingue."

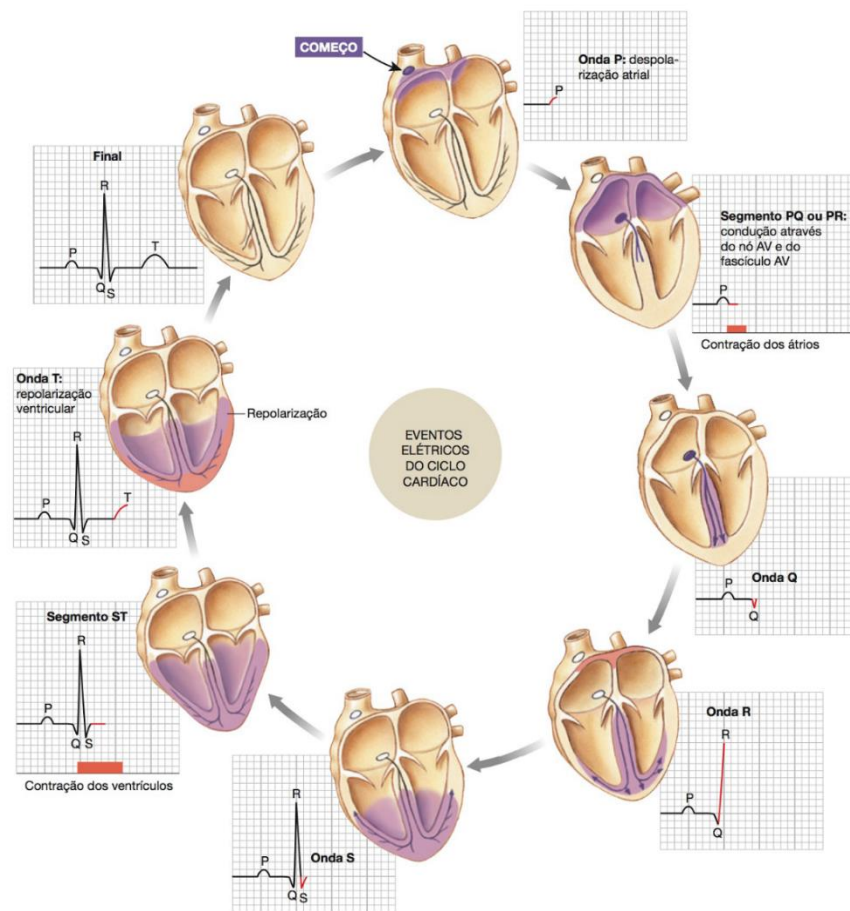


Figura 3 - Eventos elétricos do coração (extraído de Silverthorn, 2017)

Como ilustrado na Figura 3, que descreve as etapas dos sinais elétricos no coração, cada impulso nervoso está associado a um movimento para bombear o sangue. Um ciclo do eletrocardiograma, ou seja, uma batida completa, é composto pelos cinco sinais principais denominados PQRST. Ao longo desse processo, a despolarização de regiões está associada à contração dos músculos e a repolarização se refere ao relaxamento destes. O órgão possui diversos nós do sistema nervoso em seu interior, responsáveis pela ativação dos músculos. No decorrer do procedimento, são excitados em sequência, de forma coordenada para suprir as demandas do corpo humano (Silverthorn, 2017; Feldman & Goldwasser, 2004).

O ciclo será abordado apenas superficialmente, todavia o mesmo pode ser estudado de forma muito mais aprofundada, dependendo da sua utilização. Para começar, a onda P representa a despolarização dos átrios, ou seja, essas partes se contraem para bombear o sangue para os ventrículos. Isso está representado nas primeiras duas etapas da figura. Em seguida, o chamado complexo QRS representa a despolarização dos ventrículos, que são preenchidos com novo volume sanguíneo, atinge sua capacidade máxima e sua contração o expulsa para outros órgãos. Já esse processo corresponde às etapas 3 a 6 do ciclo. A última fase, associada ao sinal T, está atribuída à repolarização desta região ventricular, completando as etapas 7 e 8. Vale lembrar que a repolarização dos átrios, que não está explicitada na figura, ocorre simultaneamente ao processo descrito para o complexo QRS, que equivale ao pico mais alto do sinal (Silverthorn, 2017).

Um ponto importante do eletrocardiograma, chamado linha de base, diz respeito ao nível do sinal no gráfico entre as ondas. Está associado, por exemplo, aos segmentos PQ e ST, que também são referidos como intervalos isoeletricos.

A partir disso, o eletrocardiograma é composto por sucessivos ciclos como o descrito, com durações maiores ou menores, a depender do estado físico e psicológico de cada ser humano. Ao ser mostrado em monitores de vídeo normalmente a escala é automática, mas existe uma forma padrão de representá-lo com uma grade, principalmente no papel. O eixo vertical do gráfico representa a tensão e o horizontal o

tempo. A menor divisão da grade é representada em quadrados de 1 mm, representando 0,1 mV na altura e 40 ms com a largura. São marcados contornos de quadrados maiores, de lado equivalente a 5 vezes os menores (Becker, 2006). Assim sendo, a frequência cardíaca é calculada dividindo 1500 pelo número de quadrados pequenos do ECG entre duas ondas P consecutivas ou dois complexos (QRS). (Feldman & Goldwasser, 2004).

Acompanhar os intervalos de tempo entre ondas no eletrocardiograma permite análises importantes, tais como a avaliação do intervalo R-R. O tempo decorrido entre ondas R é denominado intervalo R-R, esse dado constitui o cálculo da frequência cardíaca e sua variação. Quanto mais precisos os dados de intervalo R-R, melhor também a estimativa dos seus dependentes (Gilgen-Ammann *et al.*, 2019).

A Variabilidade da Frequência Cardíaca depende dos valores sucessivos de intervalos R-R, que representa o tempo entre batidas do coração. Mais especificamente, está associado à despolarização do nó sinusal, localizado no átrio direito do coração. Dessa forma, essa medida pode ser relacionada com o ECG para verificar a coerência e identificar erros ou comportamentos não ideais. Esses casos incluem a discriminação de batidas sinusais ou não sinusais, mudanças não morfológicas no complexo QRS, identificação de arritmia induzida por exercício e diferenciação de taquicardia supraventricular e arritmia ventricular (Gašior *et al.*, 2022).

ii. O sensor Polar H10

O monitor de frequência cardíaca Polar H10, lançado em janeiro de 2017 pela *Polar Electro Oy*, é uma representação da inovação e qualidade que a empresa finlandesa entrega desde sua fundação em 1977. Esta introdução foi marcada por um compromisso contínuo com a evolução da tecnologia *wearable*, destacando-se por sua precisão e versatilidade.

Concebido com o objetivo de satisfazer as exigências tanto de atletas de alto rendimento quanto de amadores, o Polar H10 tem se consolidado como um instrumento de confiança no âmbito do monitoramento esportivo. A empresa é reconhecida por sua inovação, tendo introduzido no mercado os primeiros monitores de frequência cardíaca sem fio no mundo (Polar Electro Oy, c2023c). Existem antecessores respeitados no mercado da mesma empresa, também com um histórico de uso em pesquisas (Lown *et al.*, 2018), como por exemplo o Polar H7 (Gilgen-Ammann *et al.*, 2019).

As melhorias incorporadas no Polar H10 em relação ao H7 incluem um algoritmo novo de medição, a implementação de memória interna, o dobro da duração de bateria e uma faixa peitoral com proteção contra ruído para os eletrodos. Além disso, é possível fazer o update de *firmware* do dispositivo via conexão sem fio. Mais especificamente, a cinta peitoral Polar Pro, inclusa com a Polar H10, apresenta eletrodos adicionais que visam diminuir interferências, garantindo a captura precisa da frequência cardíaca com poucas distorções. Além desse fator, é equipada com um mecanismo de fivela e pontos de silicone, que ajudam a mantê-la fixa durante o uso (Polar Electro Oy, 2017).

Uma característica notável do Polar H10 é sua autonomia em armazenar dados de treino, permitindo uso independente de outros dispositivos, o que representa uma grande vantagem em esportes onde a liberdade de movimento é essencial. A faixa peitoral também é preparada para condições mais extremas, com proteção à prova d'água, suportando uma profundidade de até 30 metros. Nesses casos, a nova função de guardar medições *offline* se faz mais útil, já que na água a transmissão sem fio é comprometida (Polar Electro Oy, c2023a).

A versatilidade da faixa peitoral é reforçada por ser compatível com três tipos de protocolos de comunicação. Suportando tanto BLE, ANT+ e o protocolo *GymLink*, que utiliza 5 kHz na transmissão para dispositivos *fitness*. O sensor garante integração com uma ampla gama de aparelhos e plataformas, desde equipamentos *fitness* até aplicativos especializados. Esse suporte multiprotocolo permite aos usuários compartilharem e analisarem seus dados com facilidade, integrando-os a comunidades online voltadas para o bem-estar e expandindo o alcance de seu uso (Rainmaker, 2021).

Entendendo o contexto e a motivação por trás do desenvolvimento do Polar H10, torna-se claro como a *Polar Electro Oy* procurou criar um produto que não só atendesse aos padrões de precisão e flexibilidade exigidos por profissionais e pesquisadores, mas que também se adaptasse a condições extremas sem perder a integridade dos dados. A capacidade do dispositivo de operar em variadas temperaturas e

ambientes aquáticos é uma prova do compromisso da empresa com a pesquisa e a inovação. Com o crescimento do interesse por saúde conectada e a tendência em direção ao bem-estar otimizado, o Polar H10 foi estrategicamente introduzido a fim de atender a uma demanda por dispositivos inteligentes e integrados que ofereçam uma visão completa do desempenho e da saúde pessoal (Polar Electro Oy, 2017).

iii. Validação da cinta Polar H10

As etapas de validação de instrumentos consistem em um processo rigoroso para conhecer a fundo as capacidades do *wearable* em questão. Em primeiro lugar, escolhe-se a grandeza a ser trabalhada, e tem-se uma base de comparação a partir de medidas por um ou mais dispositivos “padrões ouros” no meio profissional. É importante descrever os procedimentos e garantir a repetibilidade dos ensaios. A partir dos dados obtidos, são feitas avaliações estatísticas para conferir erros de medição, sejam por ruído ou leituras inválidas, entre outras análises relacionando os sinais. São considerados também o curto e o longo prazo de operação, já que estes podem apresentar confiabilidade diferente. Por fim, as condições de ambiente dos testes devem estar de acordo com as especificadas pelo fabricante do dispositivo, a não ser que o objetivo do teste aborde esse limite.

Especificamente, a validação do Polar H10 e seus usos em trabalhos acadêmicos e contextos profissionais tem sido um tópico de crescente interesse na comunidade científica e no setor da saúde. O sensor é capaz de fornecer frequência cardíaca, eletrocardiograma, intervalos R-R e dados de acelerômetro. É muito usado como um dispositivo padrão para a aquisição de Variabilidade da Frequência Cardíaca (VFC) e outros dados.

Estudos (Gilgen-Ammann *et al.*, 2019; Schaffarczyk *et al.*, 2022) focados na sua validação vêm mostrando que o Polar H10 apresenta dados coerentes e pequeno viés em comparação com gravações de eletrocardiograma, posicionando-o como uma escolha confiável para praticantes e pesquisadores que necessitam de precisão em suas medições. A seguir, são analisadas implementações que corroboram com essas afirmativas, ilustrando as capacidades do dispositivo, mas que também pontuam as suas limitações.

Um dado comumente levado em conta para avaliar o treino de atletas é a VFC, que apresenta forte relação com a sua performance. Em 19 de outubro de 2022 foi publicado um estudo conduzido que tinha como objetivo a validação do Polar H10 em esquiadores profissionais mundialmente reconhecidos (Gašior *et al.*, 2022). Nele, efetuaram-se testes antes, durante e depois de exercícios físicos, com o intuito de avaliar o intervalo R-R em curto e curtíssimo prazo, respectivamente 5 minutos e 1 minuto. Nota-se que a estabilidade e qualidade do sinal varia muito entre os casos de usuário em repouso e em movimento. Ou seja, a presença de artefatos no último caso é significativamente maior. Artefatos podem ser entendidos como interferências ou outras fontes de erro em uma série temporal, e existe a questão de que não há prática padrão e difundida de como tratá-los, ainda que esse procedimento deva ser feito antes de qualquer análise. Dessa forma, mostra-se imprescindível o uso de *softwares* como o *Kubios HRV* para o pós-processamento do sinal de ECG, relacionando-o com o intervalo R-R para tratar os artefatos que podem ser administrados e conseguir um sinal mais limpo e confiável como resultado.

No trabalho de Gašior *et al.* (2022) concluíram que são obtidos dados confiáveis para os testes em curto e curtíssimo prazo, mas ressaltam que conhecimentos sobre a estacionariedade do sinal se fazem importantes para esses casos. Isso significa inferir sobre o comportamento das estatísticas de um sinal ao longo do tempo, como média, variância e estrutura de correlação. Para curtíssimo prazo, a análise estatística indicou que esses tipos de *wearable* são úteis para estudos, mas com algumas ressalvas, já que em determinados casos suas características extrapolaram os limites de acordo especificados.

A frequência cardíaca (FC) e sua variabilidade (VFC) são sinais que estão relacionados com a condição fisiológica e psicológica dos usuários (Gilgen-Ammann *et al.*, 2019). Em estudo publicado no *European Journal of Applied Physiology* (Jornal Europeu de Fisiologia Aplicada, tradução própria) por Gilgen-Ammann objetivou-se comparar o Polar H10 com um monitor *Holter* “padrão ouro”, analisando-se especialmente a qualidade do intervalo R-R.

Esses aparelhos de eletrocardiograma são a referência no meio médico para ambientes clínicos ou cenários de uso ambulatorio envolvendo intensidade física com pouco deslocamento espacial. Nas

atividades esportivas, por conta das conexões dos eletrodos e ruídos de movimento, o uso de um monitor como esse é, muitas das vezes, inviável. Entretanto, usando vestíveis, é possível realizar a medição de sinais cardíacos sem a interferência advinda da movimentação dos fios. Vale lembrar que existem monitores que utilizam 2, 3 ou até 12 eletrodos, com um cabo para cada.

A cinta Polar H10 se mostra confiável e precisa na detecção de intervalos R-R, inclusive em alta intensidade física, com taxa de erro menor do que o monitor *Holter Schiller medilog® AR12plus*, especificamente nessa categoria do estudo realizado.

A faixa na qual o medidor da Polar é acoplado possui regiões de eletrodos voltados para a proteção contra ruído na medição do ECG. O seu *design* é focado no uso em atividades físicas. A maior preocupação com esse tipo de sensor são as alterações de contato entre a pele e a seção da cinta peitoral voltada para a medição.

No estudo de Gilgen-Amman *et al.* (2019), foram testados 10 participantes em cinco atividades de demanda física variada. Estas incluíram estar em repouso, varrer o chão, andar em esteira, correr em esteira e musculação com halteres. Cada uma das ações consideradas teve intervalo de um minuto entre elas e duração de curto prazo de cinco minutos.

Os dados do ECG do *Holter* passaram por um *software* de detecção de picos no eletrocardiograma com o propósito de inferir os intervalos R-R, no caso o *Medilog Darwin Enterprise Software*. Para ambos os dispositivos (*Holter* e Polar H10) esses valores foram conferidos visualmente para classificar erros encontrados. No estudo, optou-se por uma prática usada anteriormente na literatura consistindo em considerar errôneos intervalos R-R que diferem em mais de 20% do valor anterior. Dessa forma, foram comparados os desempenhos de ambos os instrumentos de medição (Gilgen-Amman *et al.*, 2019).

Sabe-se (Gilgen-Amman *et al.*, 2019) que alguns erros principais podem ocorrer na detecção de picos R, entre eles: não detecção de picos válidos no ECG; detecção de picos errônea no ECG; e inviabilidade de detecção por conta de ruído na aquisição.

O estudo concluiu que o Polar H10 se mostrou perfeitamente capaz de efetuar as medições para as atividades, mesmo com movimento intenso por parte dos usuários. Isso indica também a possibilidade de adequação do seu uso em contextos clínicos, considerando que em casos mais próximos do repouso ambos os instrumentos apresentam taxas de erro baixas e similares (Gilgen-Amman *et al.*, 2019).

Outro trabalho, publicado por Schaffarczyk *et al.* (2022) na revista *Sensors*, sinaliza que os resultados do estudo abordado anteriormente podem ter sido influenciados pelo fato de o grupo de pessoas testadas ter características muito parecidas e o tamanho da população insuficiente de 10 pessoas. Assim, esse novo estudo buscou incluir mais indivíduos, abordando atletas recreativos, usando a Polar H10 e um dispositivo *12-channel ECG CardioPart 12 Blue* para adquirir o ECG, e com pós-processamento e interpretação pelo *software Kubios HRV*. Três situações foram analisadas: pré-atividade, exercício incremental e pós-atividade. Um foco especial direcionou-se para a comparação estatística entre os sinais, incorporando inclusive análises não lineares das séries temporais, especificamente a DFA-a1.

Esse mesmo estudo também realiza uma análise estatística dos sinais do Polar H10, comparando-o com dados extraídos de um instrumento de referência estabelecida no meio profissional. Esse dispositivo inclui eletrodos e *software* de processamento de ECG para interpretar os dados. Foram efetuados testes em um grupo de aproximadamente 25 pessoas que atendessem certos critérios de saúde e que não tivessem consumido certas substâncias (álcool, fumo, entre outros) nas últimas 24h. Determinaram-se situações de coleta de dados em repouso, durante atividade física de intensidade crescente e após o treino. O que se constatou, e que é apontado no site de suporte do Polar H10 (Polar Electro Oy, c2023b), é que diferentes pessoas podem apresentar qualidades de coleta também variadas, e um ajuste de posição do sensor tem potencial benéfico. A detecção do sinal do Polar H10 é capaz de ser favorecida se posicionada mais para a esquerda do torso do usuário, por exemplo, ou até invertida, e, em alguns casos, se posicionada nas costas.

Existem vários aplicativos para extrair e gravar os dados da faixa peitoral disponíveis nas lojas online dos *smartphones*. Além do aplicativo oficial da própria Polar, diversos outros desenvolvidos por terceiros e compatíveis – já que a abordagem para esses sinais usa técnicas conhecidas – podem ser encontrados. Uma questão que surge em relação ao uso dessas ferramentas é que na maioria das vezes não se sabe

o tratamento de dados implementado em cada uma, portanto, existem variações nos resultados. Para extrair os dados do monitor Polar H10 usou-se o aplicativo *Elite HRV* (Schaffarczyk *et al.*, 2022).

Por fim, o estudo de Schaffarczyk *et al.* (2022) concluiu que o sinal da frequência cardíaca e intervalo R-R são confiáveis, e o dispositivo é passível de ser usado comercialmente. Mais uma vez, os resultados foram a favor da validação do *wearable* em questão, principalmente na análise linear. A análise não-linear apresentou correlação um pouco mais fraca, o que pode ser justificado, em parte, pelas diferentes intensidades de sinal geradas por conta do posicionamento dos sensores em cada indivíduo.

A Variação de Frequência Cardíaca é um indicador de estresse tanto em humanos quanto em outros animais. Esse valor analisa a variação de tempo entre batimentos sucessivos do coração, e está relacionado com o funcionamento do sistema nervoso autônomo.

Em estudo publicado na revista científica *Frontiers in Veterinary Science* (Fronteiras em Ciências Veterinárias, tradução própria) analisou oito gatos – quatro machos e quatro fêmeas. Neles, instalou-se uma Polar H10 e um sensor cardíaco de referência para medir dados confiáveis de eletrocardiograma (Grigg, 2021).

Por conta de erros de aquisição atrelados tanto ao movimento dos bichos quanto à adequação da faixa para as suas fisionomias, muitos dados foram perdidos. Como a Polar H10 foi desenvolvida para ser usada em humanos, a dimensão não é a ideal para felinos. Vale lembrar que o valor de frequência cardíaca em repouso desses animais é bem maior do que o dos humanos, de 120 a 140 bpm, podendo alcançar até 220 bpm sob estresse.

No fim do estudo, observou muita discrepância entre as medições do sensor Holter e os dados da faixa peitoral. Além disso, o trabalho deixou claro que o uso da Polar H10 não seria ideal para felino, segundo os dados obtidos. Ainda assim, foi ressaltada a importância de ter acesso a um sensor que consiga análises cardíacas de gatos livres para se movimentarem, ao contrário do que foi feito no teste, que usou jaulas restritas (Grigg, 2021).

A confiabilidade do Polar H10 foi mais examinada no meio científico em termos de sua capacidade de fornecer leituras consistentes e precisas. Por exemplo, um estudo publicado pelo *PubMed* indicou que o Polar H10 exibiu uma melhor confiabilidade absoluta com o menor erro técnico de medida para a variabilidade de intervalos R-R e frequência cardíaca, sugerindo que se trata de um dispositivo adequado para a medição de HRV, inclusive em jovens crianças em ambientes educacionais reais (Schaffarczyk *et al.*, 2022).

Outro estudo, publicado na *European Journal of Investigation in Health, Psychology and Education* (Revista Europeia de Investigação em Saúde, Psicologia e Educação, tradução própria), apontou para a validade do Polar H10 em comparação com a eletrocardiografia em crianças, adicionando credibilidade ao seu uso em situações fora do laboratório, como a existência cotidiana no mundo real (Speer *et al.*, 2020).

No âmbito acadêmico, o Polar H10 tem sido implementado em diversas pesquisas para explorar seu potencial em diferentes cenários. Por exemplo, em estudos de VFC, o Polar H10 tem sido recomendado para uso devido à sua forte concordância com as gravações de ECG tradicionais, um fator determinante quando os pesquisadores pretendem aplicar os dados em intervenções clínicas ou na avaliação de estresse e recuperação em atletas.

Adicionalmente, a comparação do Polar H10 com outros dispositivos tem sido uma prática comum para validar sua precisão e confiabilidade. Diferentes estudos (Gilgen-Ammann *et al.*, 2019; Skála *et al.*, 2020) têm demonstrado que o Polar H10 supera ou está alinhado com outros dispositivos de monitoramento de frequência cardíaca no mercado, tanto em ambientes controlados quanto em configurações do mundo real. Essas comparações são fundamentais para estabelecer normas para a validação de sensores no campo da saúde e do esporte, assegurando que os dispositivos possam ser utilizados com confiança para a coleta de dados biométricos precisos.

Em suma, o Polar H10 é um dispositivo que tem sido extensivamente testado e validado em vários contextos, solidificando sua reputação como um monitor de frequência cardíaca de alta qualidade. A continuação do uso deste dispositivo em pesquisas futuras e a sua implementação em práticas profissionais depende da confirmação de sua precisão, confiabilidade e aplicabilidade em uma gama

diversificada de situações e populações. As descobertas desses estudos não apenas apoiam a utilização do Polar H10 em aplicações atuais, mas também orientam o desenvolvimento futuro de tecnologias de monitoramento esportivo e de saúde.

b. Tipo de tecnologia de aquisição

i. Wearables e o Bluetooth®

Estima-se que, em 2026 o uso de dispositivos que utilizam tecnologia *Bluetooth®* no mercado mundial ultrapasse a marca dos 7 bilhões. Considerando-se que existem, atualmente no planeta, 8 bilhões de pessoas (United Nations, [s.d.]) esta marca se aproxima cada vez mais da proporção de um dispositivo por pessoa, dos quais muitos são dispositivos *wearables*. Segundo dados do estudo *IDC Tracker Brazil Wearables Q2 2020*, o mercado de vestíveis apresentou um aumento de 21,1% no número de vendas em 2020, primeiro ano de pandemia da Covid-19 (PUCRS, 2021).

Os *wearables* se utilizam de diversos protocolos de comunicação para passar adiante os dados coletados. A forma mais comum escolhida pelos fabricantes é via *Bluetooth®*, tecnologia surgida e popularizada nas últimas décadas, com o objetivo de solucionar a troca de informação à curta distância (Bluetooth, [s.d.] a). Estimativas da *ABI Research* de 2022 mostram que, em 2026, 1.69 bilhões de aparelhos com transferência de dados por esse método serão entregues a consumidores. Desse valor, 29% correspondem a *wearables* (Marcel, 2021), o que mostra a afinidade entre a categoria de eletrônicos e o protocolo.

A tecnologia *Bluetooth®*, originada no final do Século XX, representa um marco no desenvolvimento de comunicações sem fio. Concebida inicialmente em 1994 pela *Ericsson*, uma inovadora no campo das telecomunicações, foi nomeada em homenagem ao rei Harald Bluetooth, da Dinamarca, que unificou as tribos dispersas do país. Este nome simbólico reflete a visão por trás do protocolo: a unificação de diferentes tecnologias em uma plataforma de comunicação coerente e universalmente compatível (Bluetooth, 2023).

Primeiramente, o termo “Bluetooth” era usado apenas como codinome para identificar esse tipo de comunicação sem fio. A intenção era que o nome oficial fosse “PAN” ou “RadioWare”. Contudo, o primeiro foi logo descartado com a justificativa de ser comum demais, e o segundo, por questões burocráticas, não poderia ser registrado a tempo do lançamento. Assim, o nome ficou decidido e oficializado, rapidamente passou a ser conhecido na indústria e se tornou referência no campo em todo o mundo (Bluetooth, 2023).

Desde seu advento, o *Bluetooth®* evoluiu de uma novidade técnica para um pilar da comunicação sem fio global. A primeira versão oficial, lançada em 1999, estabeleceu as bases para uma série de inovações que transformariam a conectividade pessoal e profissional. Com a introdução de melhorias significativas nas versões 1.x, o foco estava na estabilização e na robustez do protocolo. Posteriormente, com o surgimento das versões 2.x e 3.x o protocolo elevou-se, passando a ter taxas de transferência de dados mais rápidas e ampliando as possibilidades de uso (Bluetooth, [s.d.] b).

A versão 4.0 marcou a chegada do *Bluetooth® Low Energy (BLE)*, uma revolução destinada a alimentar dispositivos que necessitam de longa duração de bateria e eficiência energética, como os *wearables*. A integração do BLE facilitou a criação de uma gama de produtos inovadores, desde monitores de atividade física até dispositivos médicos portáteis, todos projetados para se integrarem perfeitamente com *smartphones* e outros equipamentos de forma a otimizar a conveniência e a eficácia no monitoramento da saúde e no condicionamento físico.

A gestão do desenvolvimento e da padronização do protocolo é administrada pelo *Bluetooth® Special Interest Group (SIG)*, um consórcio de empresas que dita as diretrizes para a evolução contínua da tecnologia. Composto por gigantes da indústria como *Ericsson®*, *Nokia®*, *Intel®* e *IBM®*, o SIG desempenha um papel crucial na definição de novos padrões que não apenas aprimoram a funcionalidade e o desempenho, mas também garantem a interoperabilidade entre milhões de dispositivos no mercado. Atualmente, o grupo *Bluetooth® SIG*, formado por milhares de empresas e contribuidores ao redor do mundo, usa o voto entre os seus constituintes para regulamentar diversos aspectos do protocolo de

comunicação, além de planejar como serão futuras versões, nas quais têm o direito de opinar a respeito de novas funcionalidades e objetivos que a tecnologia busca alcançar.

A conformidade com os padrões estabelecidos pelo SIG é essencial para a funcionalidade cruzada dessa tecnologia. Isso garante que os dispositivos de diferentes fabricantes possam se comunicar e operar juntos sem complicações, proporcionando uma experiência de usuário consistente e confiável. Essa universalidade dos padrões é o que permite uma ampla adesão ao protocolo em uma infinidade de produtos e aplicativos.

ii. Fundamentos e implementação do *Bluetooth*[®]

A tecnologia foi devidamente introduzida, mas é preciso entrar em detalhes em alguns aspectos técnicos e conceitos para a sua implementação. Esses pontos serão úteis para descrever o funcionamento da Polar H10 e fundamentarão a programação do microcontrolador. A ideia principal é que a comunicação seja universal, entre dispositivos de quaisquer fabricantes.

Praticamente toda a comunicação acontece com base na relação cliente-servidor. Entende-se como “servidor” a parte que contém certos dados, e propõe formas de disponibilizá-los. Já “cliente” é o lado que busca essas informações no servidor. A documentação oficial do *Bluetooth*[®] esquematiza arquivos de Serviços e Perfis visando cobrir as necessidades de cada um, respectivamente. Assim, existe um padrão a ser seguido para estruturar o acesso a um dispositivo, bem como uma maneira específica de requisitar seus elementos. No caso do projeto, a Polar H10 atua como o servidor e o microcontrolador como o cliente (Bluetooth, [s.d.] a).

O percurso da informação ocorre através de uma série de camadas, cada uma representando uma área envolvida na troca de mensagens. Essa divisão é comumente feita em protocolos de comunicação. A mais básica é a camada física, que está atrelada à interação com o meio no qual se transmite. O esquema exibido na Figura 4 indica de forma geral o processo, mas serão apenas abordados o *ATT*, *GATT* e *GAP*. Na imagem a divisão entre *host* e controlador é normalmente a interface entre sistemas operacionais e *chips* com antenas, respectivamente. O que está sinalizado por *APP* representa a última etapa que lida com o contato a usuários em aplicações de fato e outros dispositivos (Bluetooth, [s.d.] a).

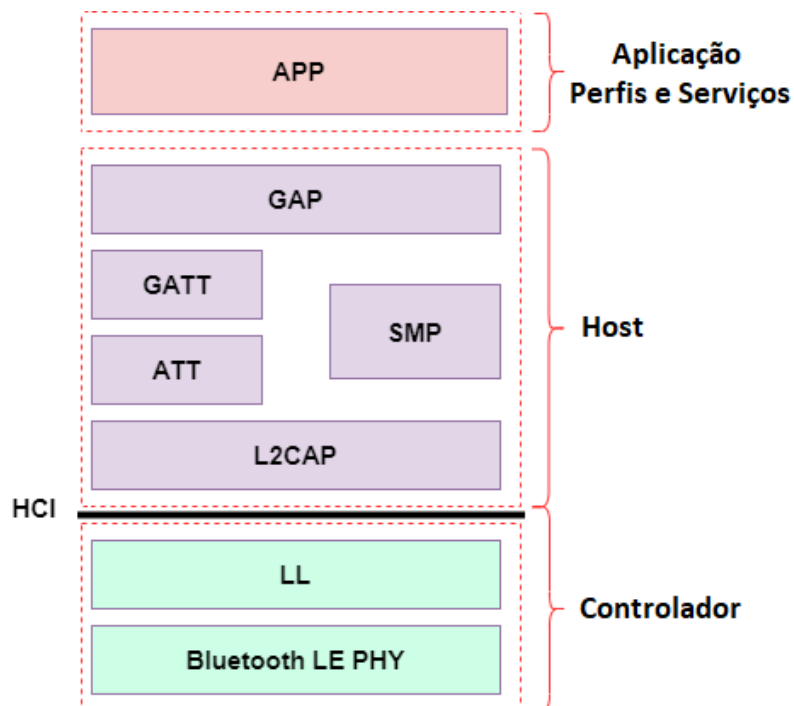


Figura 4 - Camadas de comunicação BLE (Mathworks, 2021)

O Protocolo de Atributo (ATT) ocupa uma das camadas que, além de outras funcionalidades, indexa os dados do dispositivo de forma clara para clientes, ou seja, indica como é representada a sua base de dados. São dispostos em uma estrutura referida como “tabela de atributos”, na qual cada atributo possui um índice, um Identificador Único Universal (UUID) e uma lista de permissões, e podem ser indicados em grupos relacionados a um mesmo propósito. Esse é o mecanismo que define boa parte da troca de mensagens com agentes externos (Bluetooth, [s.d.] a).

Alguns UUIDs são padronizados pelo Bluetooth® SIG, e nesse caso podem ser representados por 16 *bits*. A partir deles que se definem os documentos de Perfis e Serviços de forma universal. Esse tipo de endereço também fica disponível para ser comprado por empresas, e é disponibilizado publicamente. Programadores podem criar os seus identificadores customizados para propósitos não oficializados, e então esse tipo de UUID deve possuir 128 bits. Porém, não pode ocupar o intervalo 0000XXXX-0000-1000-8000-00805F9B34FB, já que essa é a representação completa dos UUIDs do SIG, que entram no lugar dos “X” (Bluetooth, [s.d.] a).

O ATT ainda estipula certas trocas de pacotes de dado do protocolo, o que está relacionado com as permissões de cada atributo que incluem, por exemplo, leitura e escrita. A notificação também é uma delas, consiste em mensagens enviadas pelo servidor sem a expectativa de resposta. O cliente não tem controle sobre o tempo no qual será enviada. As indicações têm funcionamento parecido, exceto pelo fato de esperarem respostas de recebimento, que devem chegar em no máximo 30 segundos.

Já o Perfil de Atributo Genérico (GATT) pode ser entendido como a abstração de algumas estruturas do ATT para representar de outra forma as capacidades de um aparelho BLE. Assim, opera em um nível mais alto. Não só isso, também estabelece procedimentos padrão para interagir com os novos elementos. Dessa forma, são apresentados os conceitos de serviços, características e descritores. Nessa ordem, existe uma relação de dependência, no sentido de que um contém os seguintes (Bluetooth, [s.d.] a).

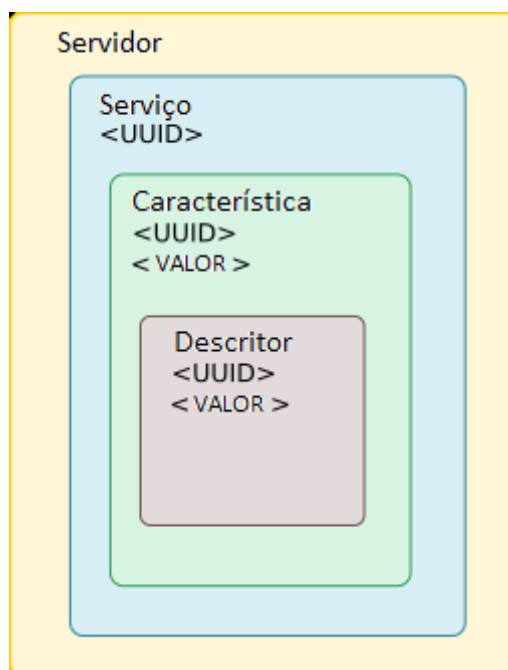


Figura 5 – Estrutura do sistema de agrupamento do GATT (adaptado de Kolban, 2018)

A Figura 5 torna clara a disposição das estruturas comentadas. Desta maneira são abordadas a maioria das interações de dispositivos com o servidor e os seus serviços. Há serviços obrigatórios para todo aparelho Bluetooth®, e há ocasiões em que a implementação de um requer outro também. Essencialmente, os serviços têm a função de agrupar funcionalidades, dependendo do contexto das suas características. Normalmente serviços primários são anunciados junto com informações básicas do dispositivo para outros nas proximidades (Kolban, 2018).

As características têm um foco maior no gerenciamento de estados do servidor. Praticamente todo o acesso a dados é feito nesse nível, e elas estão diretamente ligadas com as permissões dos atributos. Ou seja, lidam com capacidades dos atributos ATT, e modificam ou fornecem valores internos a partir de interações com os clientes. Nem toda característica possui um descritor, isso acontece apenas para permissões específicas. O papel dele não só é fornecer algum tipo de descrição sobre a funcionalidade envolvida, como o nome sugere, mas também existem tipos diferentes, também atrelados a UUIDs. O mais comum tem identificador 0x2902, e serve para gerenciar a habilitação de notificações e indicações para as conexões (Kolban, 2018).

Os conceitos vistos nessa seção são essenciais para o decorrer do projeto. A partir deles, compreende-se melhor a forma com que o *wearable* escolhido é estruturado, e como abordar a comunicação seguindo as diretrizes impostas pelo protocolo BLE.

iii. Polar H10

A Polar H10 foi introduzida como produto e como objeto de pesquisa em um contexto acadêmico. Através do que se discutiu, o seu uso como *wearable* escolhido para o projeto se justifica, e ainda será retomado. No momento é necessária uma abordagem mais técnica para se compreender na prática como trabalhar com os sinais disponíveis.



Figura 6 - Monitor cardíaco Polar H10 usado no projeto

A comunicação usada pelo dispositivo em questão pode ocorrer por BLE ou por ANT+, ambos protocolos sem fio comumente usados em *wearables*. Neste trabalho, abordou-se apenas o *Bluetooth®*, tanto por conta da disponibilidade de material para estudo e implementação quanto pela relevância de uso dessa tecnologia.

A Polar mantém um repositório no *GitHub* com um *Software Development Kit* (SDK) para a comunicação entre os seus sensores e produtos de terceiros (Polar Electro Oy, 2023b). Porém, seu uso está voltado exclusivamente para aplicações nativas em *IOS* e *Android*, que usam respectivamente as linguagens *Swift* e *Kotlin*. Esse código, por sua vez, não é compatível para a implementação direta em microcontroladores, que utilizam outras linguagens, por exemplo *MicroPython* e *C++*. Mesmo assim, esses códigos o ainda serve como referência para o seu desenvolvimento nas plataformas embarcadas em alguns pontos, ainda que portar toda a implementação seja uma tarefa complexa. Vale pontuar a existência de outros documentos importantes referentes ao detalhamento da comunicação, os quais serão abordados mais adiante.

Abaixo, estão os dados que a Polar H10 é capaz de fornecer, segundo o fabricante:

- Frequência cardíaca (BPM);

- Intervalo R-R (ms);
- Eletrocardiograma (μV);
- Aceleração nos eixos X, Y e Z (mG).

O serviço de eletrocardiografia funciona a uma taxa fixa de 130 Hz, e os valores do acelerômetro possuem maior flexibilidade na leitura. Esses últimos suportam frequências de aquisição de 25 Hz, 50 Hz, 100 Hz e 200 Hz, e intervalo de medição de 2g, 4g e 8g, sendo g a aceleração gravitacional da Terra. Além disso, existem outras funcionalidades como a gravação de dados na memória interna, e a manipulação desses dados de acordo com o objetivo do usuário. É importante ressaltar que apenas uma característica pode ser gravada de cada vez (Polar Electro Oy, 2020b; Polar Electro Oy, 2023c).

O grupo *Bluetooth® SIG* definiu uma série de perfis e serviços GATT para dispositivos BLE dependendo do seu propósito. Dessa forma, a comunicação entre dispositivos genéricos pode acontecer sem que as partes tenham conhecimento específico uma da outra, desde que sigam os padrões especificados. Para o caso da Polar H10, existem vários serviços como esses, que serão apresentados a seguir. Pode haver também funcionalidades não documentadas de maneira oficial, customizadas especialmente para um dispositivo. Nesse caso, existem boas práticas para definir UUIDs dos serviços e características, deixando pouca possibilidade de haver de conflito entre dispositivos com serviços customizados diferentes.

Com o objetivo de conhecer todos os serviços e características providenciados pela Polar H10, utilizou-se um programa para listá-los. A aplicação usada, denominada *nRF Connect*, é gratuita e foi baixada pela loja Play Store em um telefone *Android*. Dentre outras funcionalidades, a *nRF Connect* permite a busca por dispositivos *Bluetooth®* nas proximidades e mostra as capacidades de cada conexão encontrada especificadas pelo protocolo padrão. Com o sensor vestido, a procura foi efetuada e, após detectada a faixa peitoral, o pareamento foi efetuado. Como é possível ver na Figura 7, Figura 8 e Figura 9, está ilustrada a interface primeiro mostrando os dispositivos disponíveis, e em seguida após conectar à Polar H10, permitindo a interação com os seus serviços e características.

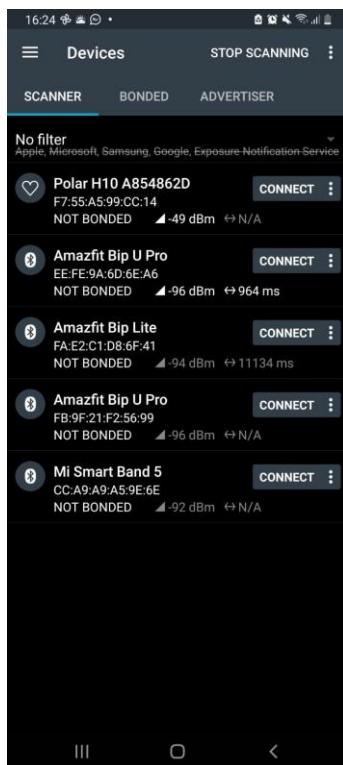


Figura 7 - Dispositivos Bluetooth® encontrados

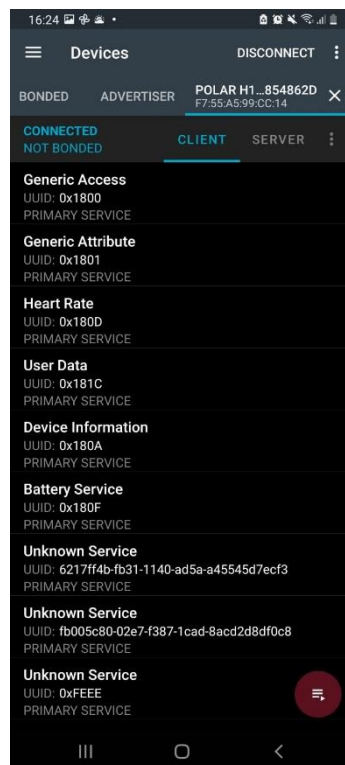


Figura 8 - Lista de serviços da faixa Polar H10

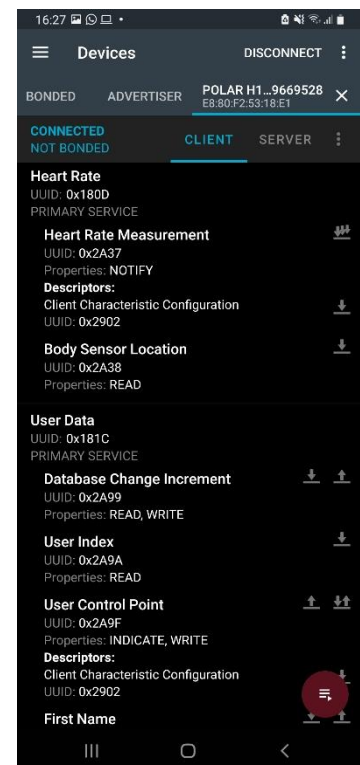


Figura 9 - Algumas características do dispositivo

Pela Figura 8 nota-se a existência de seis serviços que podem ser prontamente utilizados, mas outros três são customizados pela Polar. Sem uma documentação, é extremamente trabalhoso conhecer por completo o funcionamento desses últimos. No caso, como existe uma implementação de código aberto, seria possível fazer a adaptação do SDK para outras plataformas, se um dado ou função desejados para a aplicação se enquadrarem nesse caso. Existem os fatores de tempo e complexidade de fazê-lo, que serão levados em conta apenas caso ocorra essa situação. Em princípio, o serviço de frequência cardíaca e intervalo R-R é acessível, e vai ser abordado em detalhes mais adiante.

A Figura 9 mostra que ao selecionar um serviço, um submenu abre informando as suas características.

Assim, infere-se que o método de acesso dos dados do eletrocardiograma e do acelerômetro está relacionado com os serviços desconhecidos, já que todos os detectados estão devidamente descritos na especificação oficial do protocolo. Para conhecê-los, foi feita uma análise no repositório. Assim, em versões mais antigas do SDK, foi encontrado um arquivo que documenta um serviço customizado, denominado *Polar Measurement Data* (PMD). Sua funcionalidade é voltada para a configuração e aquisição de *streaming* (transmissão rápida e contínua) de dados, formato usado para obter os dados de eletrocardiograma e acelerômetro (Polar Electro Oy, 2021a). O UUID referente a esse serviço é o *FB005C80-02E7-F387-1CAD-8ACD2D8DF0C8*. Em decorrência disso, todas as grandezas apresentadas são passíveis de se alcançar. Posteriormente será detalhada a dinâmica de comunicação que o documento descreve.

Tenta-se conhecer os outros dois serviços faltantes, pois posteriormente podem ser úteis para a aplicação desenvolvida. Para eles, não existe documentação oficial, nem em versões antigas do repositório. Duas abordagens foram tomadas: uma delas consistiu em baixar o código fonte e pesquisar o UUID de cada um deles para descobrir os seus usos no código fonte, e a outra foi pesquisar na seção de suporte do *GitHub* da Polar.

Os arquivos de nome *BlePfcClient.swift* e *BlePfcClient.java* são os únicos que mencionam o UUID *6217FF4B-FB31-1140-AD5A-A45545D7ECF3*. A partir do conteúdo de ambos, pode-se supor, através dos comentários, nome de variáveis e lógica do programa, que o serviço em questão, referenciado como *Polar Features Configuration* (PFC), está relacionado com algumas funcionalidades específicas de dispositivos Polar. Ao buscar por esses termos na solução de problemas, em um dos resultados um desenvolvedor da Polar informa que não há uma documentação disponível para o público (Polar electro oy, 2022a). Por esse motivo, não se entrará em detalhes no seu funcionamento. Continuando a análise, nota-se que essa seção está relacionada com o suporte e controle das seguintes configurações:

- Anúncio da frequência cardíaca;
- Transmissão em 5 kHz (usado para o *GymLink*);
- Potência de transmissão adaptativa;
- Multiconexão *Bluetooth®*;
- Transmissão em ANT+;
- Modo BLE.

É importante ter em mente que existe a possibilidade de atuar sobre esses aspectos, para se ter uma ideia da capacidade total do dispositivo com o qual se trabalha. Associadas ao serviço PFC existem duas características, supostamente uma para leitura de parâmetros compatíveis, e outra para controle das mesmas.

Foi tomado um procedimento igual para identificar o serviço faltante, de UUID com valor *0xFEEE*. A julgar pelos serviços da Polar H10 que foram anunciados no repositório, mas ainda não foram encontrados, cabe assumir que são implementados por este. Existem três características associadas ao serviço, e o código é mais extenso e complexo. Os principais arquivos a serem analisados possuem os nomes *BlePsFtpClient.swift*, *BlePsFtpUtility.swift*, *BlePsFtpUtils.java* e *BlePsFtpClient.java*. Após análise, essa função customizada aparenta servir para controlar a transferência de arquivos relacionados à memória interna da Polar H10, protocolo referido como *Polar Simple File Transfer Protocol* (PSFTP). Suas características definem os dados a serem transmitidos, dependendo do sentido da informação, ou seja,

se a transmissão ocorre do dispositivo para o *host* ou do *host* para o dispositivo. Além disso, é possível configurar parâmetros da transmissão.

Nos arquivos, existem menções ao RFC 76 (Bouknight *et al.*, 1970) e ao RFC 60 (Kalin, 1970), que são documentos usados para a padronização dos protocolos de internet, e especificamente esses dois lidam com troca de informação envolvendo arquivos. Inclusive, o RFC 60 introduz uma versão simplificada de um protocolo de comunicação entre dispositivos, o que contribui para toda a hipótese, dialogando com o nome.

Assim, conclui-se que funções como a gravação de dados de treino tenham a ver com esta seção, bem como outras que dependem da memória interna, como por exemplo a referência de relógio interno do *wearable*. Infelizmente, esse último também não está abertamente documentado (Polar Electro Oy, 2021b).

A Tabela 1 dispõe tanto as informações obtidas com o auxílio do aplicativo *nRF Connect*, quanto as deduzidas a partir do repositório. Vale ressaltar que o aplicativo também permite explorar os serviços da Polar H10 para se obter mais informações. Dessa forma, foi obtida a descrição completa de todas as capacidades do dispositivo, o que constitui uma referência importante para o desenvolvimento do projeto. Cada serviço possui um endereço associado, bem como características também endereçadas e suas propriedades, que definem as formas de interação entre cada característica e os clientes BLE.

Tabela 1 - Descrição dos serviços da Polar H10

Serviços Primários	UUID do Serviço	Características	UUID da Característica	Propriedades	Descritores	UUID do Descritor
Acesso Genérico	0x1800	Nome do Dispositivo	0x2A00	LER	-	-
		Aparência	0x2A01	LER	-	-
		Parâmetros de Conexão Preferencial de Periféricos	0x2A04	LER	-	-
		Resolução de Endereço Central	0x2AA6	LER	-	-
Atributo Genérico	0x1801	Serviço Alterado	0x2A05	INDIC.	Configuração de Características do Cliente	0x2902
Frequência Cardíaca	0x180D	Medição de Frequência Cardíaca	0x2A37	NOTIF.	Configuração de Características do Cliente	0x2902
		Localização do Sensor no Corpo	0x2A38	LER	-	-
Dados do Usuário	0x181C	Incremento de Alteração de Database	0x2A99	LER, ESCR.	-	-
		Índice do Usuário	0x2A9A	LER	-	-
		Ponto de Controle do Usuário	0x2A9F	INDIC., ESCR.	Configuração de Características do Cliente	0x2902
	0x181C	Primeiro Nome	0x2A8A	PROPS ESTENDIDAS, LER, ESCR.	Propriedades Estendidas de Características	0x2900
		Último Nome	0x2A90	PROPS ESTENDIDAS, LER, ESCR.	Propriedades Estendidas de Características	0x2900
		Idade	0x2A80	LER, ESCR.	-	-
		Gênero	0x2A8C	LER, ESCR.	-	-
		Peso	0x2A98	LER, ESCR.	-	-
		Altura	0x2A8E	LER, ESCR.	-	-
		Idioma	0x2AA2	LER, ESCR.	-	-
Informações do Dispositivo	0x180A	String de Nome do Fabricante	0x2A29	LER	-	-
		String de Nome do Modelo	0x2A24	LER	-	-
		String de Número Serial	0x2A25	LER	-	-
		String de Revisão de Hardware	0x2A27	LER	-	-

Serviços Primários	UUID do Serviço	Características	UUID da Característica	Propriedades	Descritores	UUID do Descritor
		String de Revisão de Firmware	0x2A26	LER	-	-
		String de Revisão de Software	0x2A28	LER	-	-
		ID do Sistema	0x2A23	LER	-	-
Serviço de Bateria	0x180F	Nível da Bateria	0x2A19	NOTIF., LER	Configuração de Características do Cliente	0x2902
Polar Measurement Data (PMD)	FB005C80-02E7-F387-1CAD-8ACD2D8DF0C8	Ponto de Controle PMD	FB005C81-02E7-F387-1CAD-8ACD2D8DF0C8	INDIC., LER, ESCR.	Configuração de Características do Cliente	0x2902
		MTU de Dados PMD	FB005C82-02E7-F387-1CAD-8ACD2D8DF0C8	NOTIF.	Configuração de Características do Cliente	0x2902
Polar Features Configuration Service (PFC)	6217FF4B-FB31-1140-AD5A-A45545D7ECF3	Feature PFC	6217FF4C-C8EC-B1FB-1380-3AD986708E2D	LER	-	-
		Ponto de Controle PFC	6217FF4D-91BB-91D0-7E2A-7CD3BDA8A1F3	INDIC., ESCR. SEM RESPOSTA	Configuração de Características do Cliente	0x2902
Polar Simple File Transfer Protocol (PSFTP)	0xFEED	Característica de MTU PSFTP	FB005C51-02E7-F387-1CAD-8ACD2D8DF0C8	NOTIF., ESCR., ESCR. SEM RESPOSTA	Configuração de Características do Cliente	0x2902
		Notificação de Dispositivo para Host PSFTP	FB005C52-02E7-F387-1CAD-8ACD2D8DF0C8	NOTIF.	Configuração de Características do Cliente	0x2902
		Notificação de Host para Dispositivo PSFTP	FB005C53-02E7-F387-1CAD-8ACD2D8DF0C8	ESCR., ESCR. SEM RESPOSTA	-	-

Fonte: elaborado pelo autor

Para o objetivo do projeto, os principais serviços a serem considerados são o *Frequência Cardíaca* e o *Polar Measurement Data*, já que fornecem os dados desejados. Um serviço não essencial, mas também importante, é o *Serviço de Bateria*, que fornece o nível da bateria da Polar H10, informação útil para a operação. Os serviços *Acesso Genérico* e *Atributo Genérico* são obrigatórios para qualquer dispositivo BLE (Bluetooth, [s.d.] a), e no caso serão úteis para iniciar a conexão com o dispositivo Polar H10 específico em questão. Os demais serviços podem ser descartados para a etapa inicial do desenvolvimento, porém posteriormente podem ser implementados em outras funcionalidades.

- Obtenção da frequência cardíaca e intervalos R-R.

Uma vez escolhidos os serviços de interesse, o próximo passo torna-se determinar como interagir com eles para se conseguir os dados desejados. Nota-se que o serviço *Frequência Cardíaca* possui uma característica *Medição de Frequência Cardíaca* para fornecer tanto esse dado quanto o intervalo R-R. Como essa funcionalidade está definida pelo *Bluetooth® SIG*, a documentação de perfil e serviço de frequência cardíaca define como deve ser toda a interação acerca da obtenção desse dado (Bluetooth, [s.d.] c, Bluetooth; [s.d.] d). Ambos estando presentes no site oficial do *Bluetooth®*, o documento *Heart Rate Profile* aborda parâmetros de conexão e a funcionalidade geral dos serviços necessários no dispositivo. Por sua vez, o documento *Heart Rate Service* estrutura em detalhes os serviços e como suas informações serão implementadas nas mensagens de comunicação. De acordo com o estabelecido, ao se acessar a característica de *Medição de Frequência Cardíaca*, a sua resposta inclui o seguinte:

- Um campo de *flags* (marcadores), para mostrar as funcionalidades suportadas e a presença de alguns campos;
- Uma medida de frequência cardíaca;
- A depender das flags, de 0 a 8 ou 9 campos com intervalos R-R, quantidade máxima definida caso a funcionalidade de energia gasta esteja ou não presente;
- A depender das flags, campos com a energia gasta acumulada pelo usuário.

A Polar H10 não calcula a energia gasta, já que, como especificado pela documentação, em caso positivo, deve existir um serviço chamado *Ponto de Controle de Frequência Cardíaca*. Assim, esse campo nunca estará presente na resposta, e podem aparecer de 0 a 9 intervalos R-R.

As flags da resposta são encapsuladas em um byte de transmissão, no qual cada bit indica uma funcionalidade disponível ou comportamento apresentado. A Tabela 2 apresenta essas informações:

Tabela 2 - Flags de notificação da característica de medição de frequência cardíaca

Flags de Notificação de Frequência Cardíaca		
Byte de Flags	Valor	Descrição
Bit 0	0	Formato da FC é UINT8
	1	Formato da FC é UINT16
Bit 1	0	Sensor fora de contato com o corpo
	1	Sensor em contato com o corpo
Bit 2	0	Contato com a pele não é suportado
	1	Contato com a pele é suportado
Bit 3	0	Campo de Energia despendida ausente
	1	Campo de Energia despendida presente
Bit 4	0	Valores de Intervalo R-R ausentes
	1	Valores de intervalo R-R presentes
Bits 5 .. 7	-	RFU

Fonte: adaptado do documento *Heart Rate Profile Test Suite* (Bluetooth, [s.d.] d)

A partir da documentação, presente na descrição do repositório oficial (Polar Electro Oy, 2023b), confirma-se que o dispositivo não suporta a função de indicar o contato do sensor com a pele, e que não há suporte para a energia despendida. Dessa forma, os bits 1, 2 e 3 da Tabela 2 são sempre 0 no campo de *flags* da notificação da frequência cardíaca. Os que podem variar são os bits 0 e 4. Para este último, quando o valor da FC passa de 255 BPM, limite de representação de 1 byte, ele passa a ocupar 2 bytes, também com o formato de inteiro sem sinal. Porém, é um caso incomum em humanos, sendo mais voltado para o monitoramento de animais (Chhabra, *et al.*, 2012), o que não é inicialmente o objetivo deste trabalho. Portanto, para a maioria dos casos, pode-se esperar apenas um byte para esse campo.

Existe o caso referente ao outro extremo, quando a frequência cardíaca está baixa, e pode não haver intervalos R-R entre notificações. Nesse momento o bit 4 assume valor zero. Espera-se que o cliente lide com essas situações da maneira correta.

Uma vez que o funcionamento das flags é bem descrito, pode-se analisar a composição da resposta completa para a característica da frequência cardíaca. A Tabela 3 mostra um exemplo de resposta com a mensagem segmentada e detalhada.

Tabela 3 - Exemplo de notificação da característica de frequência cardíaca

Exemplo de Notificação de Frequência Cardíaca				
Byte	Dados	Descrição	Valor	Unidade
0	10	Flags	-	-
1	45	Frequência Cardíaca	69	BPM
2	44	Intervalo R-R 1	836	1/1024 s
3	03			
4	64	Intervalo R-R 2	868	1/1024 s
5	03			

Fonte: elaborado pelo autor

Através de sua análise, a quantidade máxima de dados para a mensagem pode conter 2 bytes de frequência cardíaca, e 16 bytes de intervalo R-R ou apenas um byte de frequência cardíaca e 18 bytes de intervalo R-R. Dessa forma, o limite de tamanho alcança não mais do que 19 bytes.

Existe um detalhe importante referente ao intervalo R-R, em relação à forma como é calculado. Como ilustrado na Tabela 3, a sua representação na mensagem ocupa 2 bytes, representando um inteiro sem sinal. Além disso, assim como todo o protocolo BLE, utiliza a codificação *little endian* (Bluetooth, [s.d.] b), ou seja, o byte menos significativo ocupa as primeiras posições na disposição dos dados na mensagem. A sua unidade de referência é de 1/1024 segundos, por conta de implementações antigas (Polar Electro Oy, 2022b).

Caso existam mais intervalos desde a última notificação BLE do que se encaixa em uma mensagem, como especificado pelo serviço GATT de Frequência Cardíaca (Bluetooth, [s.d.] c), esses valores podem ser transmitidos na próxima mensagem. No caso de mesmo assim não haver espaço, os dados mais antigos podem ser descartados.

A unidade de medida dos intervalos R-R sendo na escala de 1/1024 pode ter sido adotada por motivos históricos, estipulada inicialmente para evitar a perda de dados na aquisição dos sensores. Essa unidade de tempo é mais favorável para ser trabalhada para o armazenamento dos dados no formato binário, e pode ser facilmente convertida pelo usuário após a aquisição. No lançamento da Polar H10, essa especificação estava descrita nas normas do documento *Heart Service* do *Bluetooth® Low Energy* (Bluetooth, [s.d.] c).

Como foi ilustrado anteriormente com a descrição das capacidades do *wearable*, a única forma de interação com o serviço em questão é por meio de notificações. O tempo entre cada notificação também é parametrizado, e de controle exclusivo do servidor, que no caso é o monitor cardíaco. Mesmo assim, no regulamento existe certa margem para a implementação (Bluetooth, [s.d.] c). Através dos testes

feitos, constatou-se que a Polar H10 envia as notificações do serviço de 1 s a 2.5 s de intervalo, mas são disparadas em intervalos mais curtos logo após a requisição dos dados.

Foram feitos testes mais uma vez com o aplicativo *nRF Connect* para se entender melhor a interação com o serviço para a aquisição de dados. A partir disso, foi possível verificar a troca de mensagens em hexadecimal e a interpretação de alguns resultados. Ilustrado pela Figura 10 e Figura 11, é possível efetuar a inscrição para notificações na característica de frequência cardíaca, e assim analisar cada uma. Esse processo acontece pela interação com o *Client Characteristic Configuration*, presente na imagem, ou Configuração de Característica de Cliente, que é um descritor voltado para controlar a interação entre usuário e dispositivo. Na Figura 10 observa-se um histórico de mensagens da comunicação BLE e na Figura 11 está a interpretação de alguns dados pelo aplicativo, mas nem todos são identificados. A frequência cardíaca e alguns intervalos R-R são detectados, e apenas a flag da função de “contato com a pele” foi detectada pelo aplicativo. Como visto anteriormente, para esse modelo de *wearable* será sempre falso, não apresentando utilidade para o sistema. No log de comunicação, observa-se que a estrutura das mensagens BLE segue o mesmo padrão mencionado anteriormente.

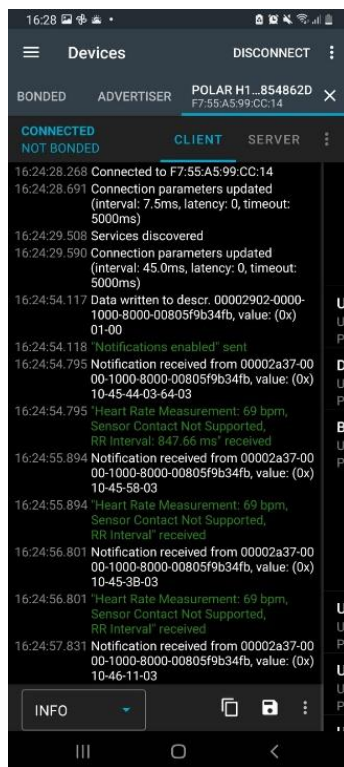


Figura 10 - Log de comunicação com a característica Medição de Frequência Cardíaca

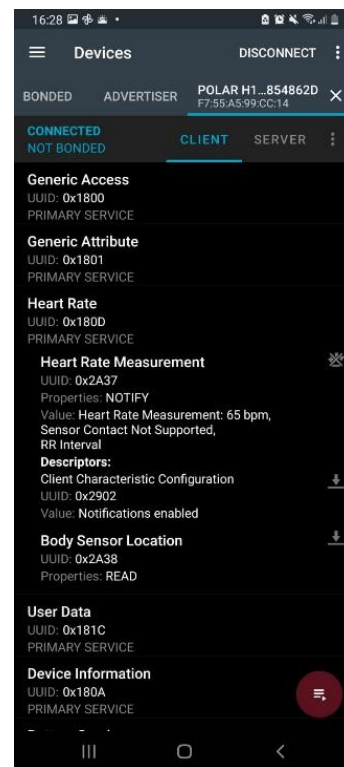


Figura 11 - Inscrição e interpretação das notificações da característica Medição de Frequência Cardíaca

A Figura 11 mostra o aplicativo após se inscrever na característica de nome *Heart Rate Measurement* na imagem, o que foi feito clicando no ícone alinhado com o seu nome. Logo abaixo, aparecem alguns valores interpretados da medição. A Figura 10 informa em um log a primeira conexão do *smartphone* com a Polar H10, seguido da inscrição para notificações, e em seguida começam a chegar mensagens do dispositivo, catalogadas com o UUID da característica e os dados em hexadecimal. Como se pode ver, existem casos com apenas um intervalo R-R e casos com dois.

Dessa forma, a aquisição de dados de frequência cardíaca teve o seu funcionamento esclarecido e testado, e é passível de ser implementada em um microcontrolador, o que será abordado mais adiante. Além de adquirir os dados, também foi explorado como interpretar as funcionalidades do dispositivo BLE.

- Obtenção do nível da bateria

A aquisição das informações a respeito da bateria de um dispositivo BLE pode ter diferentes níveis de precisão. Para o caso da Polar H10, basta se ter uma carga aproximada restante no dispositivo, para sinalizar a troca da bateria. Esse serviço também é bem documentado na referência oficial do BLE (Bluetooth, [s.d.] e). Como visto anteriormente, a faixa peitoral dispõe apenas de recursos para fornecer o nível da bateria. Segundo esse protocolo, o dado é calculado analisando a energia disponível em relação à capacidade da bateria, e é retornado na forma percentual.

Como visto na descrição dos serviços da Polar H10, apresentados anteriormente na Tabela 1, essa característica suporta leitura, que pode ser feita a qualquer momento, e notificação, que é enviada ao cliente BLE apenas quando há mudança no valor do nível da bateria. O dado recebido é um byte representando um inteiro sem sinal, que equivale à porcentagem de carga restante na bateria do dispositivo, e o *wearable* só envia notificações quando o valor varia em 10%, que é a menor mudança implementada.

A título de manter a coerência com a apresentação dos demais dados, a Tabela 4 mostra a estrutura da mensagem de interesse sobre a bateria.

Tabela 4 - Estrutura de resposta da característica Nível da Bateria

Exemplo de Leitura/Notificação da Bateria				
Byte	Dados	Descrição	Valor	Unidade
0	64	Nível da Bateria	100	%

Fonte: elaborado pelo autor

Do mesmo jeito que foi feito com o serviço de frequência cardíaca, esse dado pode ser adquirido pelo aplicativo, para serem efetuados testes e verificar o funcionamento.

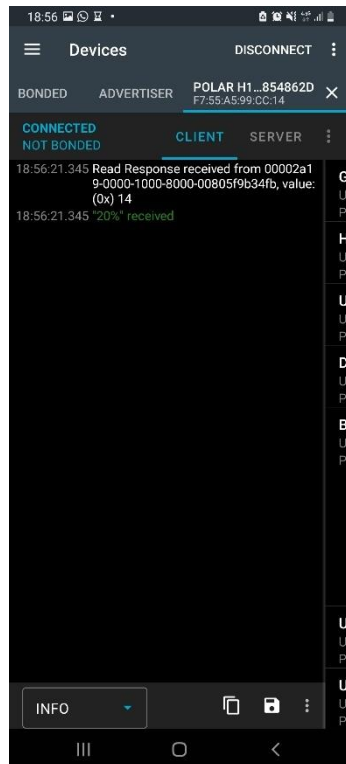


Figura 12 – Log de comunicação com a característica Nível da Bateria

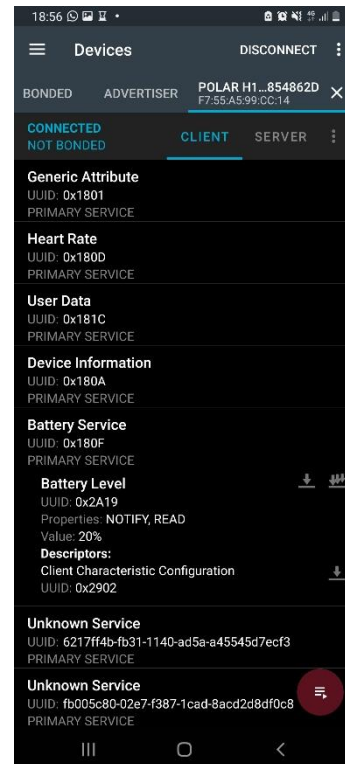


Figura 13 – Leitura da característica Nível da Bateria

Na Figura 13 foi realizada apenas uma leitura da característica, representada pelo ícone da seta unitária. A resposta é mostrada no mesmo menu, e aparece no log de comunicação, na Figura 12.

- Obtenção de dados de acelerômetro e eletrocardiograma

A aquisição dos dados de acelerômetro e eletrocardiograma deve ser feita pelo serviço customizado *Polar Measurement Data*. Como ilustrado anteriormente na Tabela 1, tem-se duas características associadas a essa funcionalidade. Uma delas é referenciada como *Ponto de Controle PMD* e, como o nome sugere, serve para controlar a transmissão dos dados e obter informações a respeito de funcionalidades disponíveis. A outra característica, *MTU de Dados PMD*, tem o propósito de enviar ao cliente os pacotes com as informações desejadas, por meio de notificações, de acordo com o que foi parametrizado na característica de controle.

Os procedimentos para a interação com esse serviço, como foi mencionado anteriormente, foram descritos em parte pela Polar em um documento chamado *Polar_Measurement_Data_Specification.pdf*. Porém, no momento de confecção deste trabalho, o arquivo não se encontra mais presente no repositório oficial do *GitHub*, podendo ser acessado em versões antigas (Polar Electro Oy, 2021a).

Antes de se iniciar a comunicação entre a Polar e os clientes, devem ser cumpridos alguns pré-requisitos. O cliente deve efetuar a inscrição para indicações na característica *Ponto de Controle PMD* e inscrição para notificações na característica *MTU de dados PMD*. Além disso, é preciso estabelecer um MTU de no mínimo 232, ou seja, a comunicação deve suportar no máximo esse tamanho de pacote em bytes. O padrão é de 23 bytes para o MTU (Punch Through, 2017). Por fim, recomenda-se uma conexão *LE 2M PHY*, ainda que não seja essencial. Trata-se de uma configuração da camada física que permite uma taxa de transferência de 2 Mb/s, uma boa melhora na conexão considerando que a velocidade padrão é de 1 Mb/s, conhecida como *LE 1M PHY* (Silicon Laboratories, c2023).

Como os dados em questão são contínuos e apresentam grande número de amostras, a Polar implementou a sua aquisição por meio de um *streaming* para enviar informações. Existem alguns comandos para configurar esse processo, e cada um será detalhado adiante. Após isso, os pacotes com amostras de aceleração e de eletrocardiograma serão descritos. Vale lembrar que, para descrever a comunicação, a mensagem é estruturada em *bytes* representados em hexadecimal. Como mencionado anteriormente, os dados compostos por mais de um *byte* são sempre representados em *little endian*, como é da especificação BLE. Essa etapa será a base de todo o desenvolvimento com o microcontrolador no projeto, logo todas as informações e formatos usados são apresentados detalhadamente.

Antes de se fazer a aquisição, é possível obter as capacidades (ou *features*) da Polar H10 para essa situação se for efetuada uma leitura na característica *Ponto de Controle PMD*. Vale notar que toda a troca de mensagens antes de se receber os valores dos sensores ocorrerá nessa característica, durante essas etapas de configuração. Além disso, como está programada com indicações, toda requisição apresentará uma resposta, diferente do que ocorre com as notificações. A resposta da leitura está descrita na Tabela 5 a seguir:

Tabela 5 – Features da Polar H10 para streaming de dados

Resposta do Ponto de Controle PMD para Obter Features			
Byte	Dados	Descrição	Valor
0	0F	Resposta de Leitura	-
1	05	Bit 0: ECG Suportado	VERDADEIRO
		Bit 1: PPG Suportado	FALSO
		Bit 2: ACC Suportado	VERDADEIRO
		Bit 3: PPI Suportado	FALSO
		Bit 4: RFU	FALSO
		Bit 5: GYRO Suportado	FALSO
		Bit 6: MAG Suportado	FALSO
		Bit 7: RFU	FALSO
2 .. 16	00	RFU	FALSO

Fonte: elaborado pelo autor, com dados adaptados de Polar Electro Oy (2021a)

Como se pode observar, a resposta consiste em 17 bytes, dos quais a maioria está reservada para uso futuro, no momento de desenvolvimento do projeto. No caso, apenas eletrocardiograma e acelerômetro são suportados, como era o esperado, mas o protocolo é feito para a compatibilidade com outros dispositivos Polar. No futuro, pode ser que haja sensores com mais funcionalidades, por isso vale a pena se atentar ao funcionamento geral do protocolo.

Uma vez confirmada a possibilidade de aquisição dos dados, o próximo passo deve ser buscar as configurações de cada tipo de transmissão. Isso pode ser feito escrevendo um comando de dois *bytes* na característica *Ponto de Controle PMD*. A Tabela 6 abaixo indica o significado de cada parte da mensagem a ser enviada.

Tabela 6 - Requisição para obter configurações específicas de tipo de stream

Requisição de Configurações de Stream ACC/ECG			
Byte	Dados	Descrição	Valor
0	01	Obter Configurações de Medição	-
1	02/00	Tipo de Medição	ACC/ECG

Fonte: elaborado pelo autor, com dados adaptados de Polar Electro Oy (2021a)

O primeiro byte da mensagem indica a operação desejada, no caso a requisição da configuração da transmissão de dados. Nota-se que dependendo do sinal escolhido, aceleração (ACC) ou eletrocardiograma (ECG), deve ser enviado 02 ou 00 como o segundo *byte*, respectivamente. O código para cada tipo de dado também está na documentação oficial (Polar Electro Oy, 2021a), contendo as representações para os outros sinais eventualmente suportados por dispositivos. Para o projeto, apenas o ACC e o ECG serão relevantes, mas a tabela completa pode ser consultada no ANEXO – Detalhamento da Comunicação da Polar. Daqui em diante, esses tipos de medida serão representados pelos mesmos códigos hexadecimais.

As respostas para cada uma das requisições estão descritas na Tabela 7 e na Tabela 8. Esses dados são importantes para se construir as mensagens de início de transmissão de dados, visto que na requisição devem estar presentes apenas parâmetros suportados pelo servidor BLE, no caso a Polar H10.

Tabela 7 - Configuração de stream de dados de aceleração

Resposta do Ponto de Controle PMD para Configurações de Stream ACC				
Byte	Dados	Descrição	Valor	Unidade
0	F0	Resposta do Ponto de Controle	-	-
1	01	Opcode	Obter Configurações de Medição	-
2	02	Tipo de Medição	ACC	-
3	00	Código de Erro	SUCESSO	-
4	00	Mais Frames	FALSO	-
5	00	Parâmetro	TAXA DE AMOSTRAS	-
6	04	Tamanho do Vetor	4	-
7	19	Elemento 0	25	Hz
8	00			
9	32	Elemento 1	50	Hz
10	00			
11	64	Elemento 2	100	Hz
12	00			
13	C8	Elemento 3	200	Hz
14	00			
15	01	Parâmetro	RESOLUÇÃO	-
16	01	Tamanho do Vetor	1	-
17	10	Elemento 0	16	bits
18	00			
19	02	Parâmetro	ALCANCE	-
20	03	Tamanho do Vetor	3	-
21	02	Elemento 0	2	g
22	00			
23	04	Elemento 1	4	g
24	00			
25	08	Elemento 2	8	g
26	00			

Fonte: elaborado pelo autor, com dados adaptados de Polar Electro Oy (2021a)

A resposta com tamanho de 27 *bytes* mostra todas as especificações disponíveis para a coleta dos dados de aceleração. O *byte* 0 da mensagem indica uma resposta da característica *Ponto de Controle PMD*, e o *byte* 1 apenas confirma a operação solicitada, que é a indicada na mensagem de requisição. Os três *bytes* seguintes indicam respectivamente o tipo de medida ACC, o código de erro e se há mais partes da mensagem, o que não é o caso. Mais especificamente, todos os códigos de erro são mencionados no documento da Polar H10, e estão sinalizados no ANEXO – Detalhamento da Comunicação da Polar. No exemplo, a requisição foi um sucesso, porém os códigos de erro são úteis para solucionar problemas de configuração de conexão e/ou requisições incorretas para o servidor.

A partir do *byte* 5, os parâmetros são estruturados de forma consistente. Existe um código para o tipo de configuração, seguido da quantidade de opções disponíveis. Em termos de programação, é fornecido o tamanho de um vetor de elementos que está por vir. Cada elemento tem um tamanho de 2 *bytes*. Dessa forma, constata-se que o parâmetro taxa de amostras pode ser 25 Hz, 50 Hz, 100 Hz ou 200 Hz.

A resolução de cada amostra é fixa em 16 *bits* e a faixa de medição pode assumir o valor de 2 g, 4 g ou 8 g, sendo “g” a aceleração da gravidade da Terra.

A seguir está a resposta da Polar H10 com as configurações de eletrocardiograma.

Tabela 8 - Configuração de stream de dados de eletrocardiograma

Resposta do Ponto de Controle PMD para Configurações de Stream ECG				
Byte	Dados	Descrição	Valor	Unidade
0	F0	Resposta do Ponto de Controle	-	-
1	01	Opcode	Obter Configurações de Medição	-
2	00	Tipo de Medição	ECG	-
3	00	Código de Erro	SUCESSO	-
4	00	Mais Frames	FALSO	-
5	00	Parâmetro	TAXA DE AMOSTRAS	-
6	01	Tamanho do Vetor	1	-
7	82	Elemento 0	130	Hz
8	00			
9	01	Parâmetro	RESOLUÇÃO	-
10	01	Tamanho do Vetor	1	-
11	0E	Elemento 0	14	bits
12	00			

Fonte: elaborado pelo autor, com dados adaptados de Polar Electro Oy (2021a)

A mensagem segue a mesma estrutura para o caso do acelerômetro, como era de se esperar. Os primeiros 5 *bytes* são idênticos, com exceção do *byte* 2 que passa a representar o ECG. De resto, existem dois parâmetros fixos para a aquisição deste tipo de sinal, sendo eles uma taxa de amostras de 130 Hz e uma resolução de 14 bits para cada uma. Por convenção, a frequência de aquisição de um eletrocardiograma se situa entre 0.05 Hz e 150 Hz (Trägårdh & Schlegel, 2007), mas existem componentes com frequências mais altas no sinal. Assim, a empresa Polar optou por se estabelecer dentro do padrão estabelecido, com certa margem dos limites.

Com o que foi observado até então, torna-se possível fazer o pedido para o início da transmissão da Polar H10. Tem-se os exemplos de pedidos para os dois tipos de dados expressos na Tabela 9 e na Tabela 10.

Tabela 9 - Requisição de início de stream de dados de acelerômetro

Exemplo de Requisição de Início de Stream ACC				
Byte	Dados	Descrição	Valor	Unidade
0	02	Iniciar Medição	-	-
1	02	Tipo de Medição	ACC	-
2	02	Parâmetro	ALCANCE	-
3	01	Tamanho do Vetor	1	-
4	08	Elemento 0	8	g
5	00			
6	00	Parâmetro	TAXA DE AMOSTRAS	-
7	01	Tamanho do Vetor	1	-
8	C8	Elemento 0	200	Hz
9	00			
10	01	Parâmetro	RESOLUÇÃO	-
11	01	Tamanho do Vetor	1	-
12	10	Elemento 0	16	bits
13	00			

Fonte: elaborado pelo autor, com dados adaptados de Polar Electro Oy (2021a)

Para o caso da aceleração, o primeiro *byte* da mensagem indica o comando de iniciar a aquisição, e logo em seguida o tipo de sinal é especificado. A partir disso, o pedido segue a estrutura fornecida pela Polar H10 anteriormente ao informar os parâmetros disponíveis. Assim, são enviados os códigos para os tipos de configuração, seguidos do tamanho do vetor de argumentos e o vetor em si, cada elemento com 2 *bytes*. O exemplo inicia um pedido com a maior faixa, de 8 g, taxa de amostragem máxima de 200 Hz e 16 bits de resolução, que é a única opção, porém precisa ser enviada.

A seguir está o pedido para a transmissão do eletrocardiograma.

Tabela 10 - Requisição de início de stream de dados de eletrocardiograma

Exemplo de Requisição de Início de Stream ECG				
Byte	Dados	Descrição	Valor	Unidade
0	02	Iniciar Medição	-	-
1	00	Tipo de Medição	ECG	-
2	00	Parâmetro	TAXA DE AMOSTRAS	-
3	01	Tamanho do Vetor	1	-
4	82	Elemento 0	130	Hz
5	00			
6	01	Parâmetro	RESOLUÇÃO	-
7	01	Tamanho do Vetor	1	-
8	0E	Elemento 0	14	bits
9	00			

Fonte: elaborado pelo autor, com dados adaptados de Polar Electro Oy (2021a)

Diferentemente do caso anterior, essa medida dispõe de apenas uma possibilidade de requisição, dado que cada parâmetro possui suporte para apenas um valor. Dessa forma, a transmissão do tipo ECG tem taxa de amostras de 130 Hz e resolução de 14 *bits*, como visto anteriormente. Existe uma resposta para o pedido de início da *stream* de dados, descrita na Tabela 11. Por apresentar a mesma estrutura para ambos os sinais ACC e ECG, foi apresentada de forma resumida.

Tabela 11 - Resposta do Ponto de Controle PMD para início de stream, dependendo do tipo de dado

Resposta do Ponto de Controle PMD para Início de Stream ACC/ECG			
Byte	Dados	Descrição	Valor
0	F0	Resposta do Ponto de Controle	-
1	02	Opcode	Iniciar Medição
2	02/00	Tipo de Medição	ACC/ECG
3	00	Código de Erro	SUCESSO
4	00	Mais Frames	FALSO
5	01	Reservado	-

Fonte: elaborado pelo autor, com dados adaptados de Polar Electro Oy (2021a)

O primeiro *byte* indica o tipo da mensagem, que é uma resposta da característica de controle. De resto, os pontos mais importantes a serem confirmados são o código do pedido, o tipo de medida e o código de erro. Para este último campo, em caso de sucesso, a característica *MTU de Dados PMD* começará a receber notificações com os dados desejados. Porém, antes que se trate da forma na qual isso ocorre, é necessário mencionar o último par comando/resposta para a característica trabalhada no momento, a *Ponto de Controle PMD*. Essa função diz respeito ao encerramento da transmissão, útil para quando se deseja interromper o processo de aquisição, seja para desconectar o cliente ou por razões do meio físico. Dependendo das condições de conexão, como baixa taxa de transferência por exemplo, pode ser interessante priorizar apenas uma *stream* para que não haja perda de dados, se ambos juntos não forem suportados por conta de interferências. Porém, em condições ideais, sabe-se que é possível adquirir ACC e ECG ao mesmo tempo (Polar Electro Oy, 2020a).

A Tabela 12 e a Tabela 13, respectivamente, indicam a requisição de encerramento da transmissão e a resposta, ambas adaptadas para os dois tipos de dados.

Tabela 12 - Requisição de encerramento de stream de dados

Requisição de Encerramento de Stream ACC/ECG			
Byte	Dados	Descrição	Valor
0	03	Encerrar Medição	-
1	02/00	Tipo de Medição	ACC/ECG

Fonte: elaborado pelo autor, com dados adaptados de Polar Electro Oy (2021a)

Tabela 13 - Resposta do Ponto de Controle PMD para encerramento de stream de dados

Resposta do Ponto de Controle PMD para Encerramento de Stream ACC/ECG			
Byte	Dados	Descrição	Valor
0	F0	Resposta do Ponto de Controle	-
1	03	Opcode	Encerrar Medição
2	02/00	Tipo de Medição	ACC/ECG
3	00	Código de Erro	SUCESSO
4	00	Mais Frames	FALSO

Fonte: elaborado pelo autor, com dados adaptados de Polar Electro Oy (2021a)

Nota-se que o Opcode de encerramento de dados tem código 03. O resto dos pontos presentes nas mensagens são análogos ao que já foi abordado.

Parte-se então para o recebimento dos pacotes da *stream* de dados, o que ocorre na característica *MTU de Dados PMD*. Antes de se entrar em detalhes nesse assunto, é importante comentar sobre os *Frame Types*. São úteis para providenciar informações sobre a estrutura da mensagem, como a unidade dos dados presentes, tamanho de amostras, entre outros. No caso da Polar H10, apenas um tipo de *frame* é suportado tanto para *streams* ACC ou ECG. Por isso, as respostas terão sempre a mesma estrutura geral, e não seria vantajoso explorar as outras variedades. Caso seja necessário em alguma circunstância, os demais *Frame Types* estão disponíveis no ANEXO – Detalhamento da Comunicação da Polar. Reiterando, a *stream* de ACC sempre usará o tipo de *frame* 1 e a *stream* de ECG o tipo 0.

A Tabela 14 mostra uma resposta com medições do acelerômetro, assumindo uma requisição exatamente como foi exemplificado.

Tabela 14 - Notificação de stream de dados de aceleração

Exemplo de Notificação de Stream ACC					
Byte	Dados	Descrição	Valor	Unidade	
0	02	Tipo de Dado	ACC	-	
1	E2	Timestamp	599616677003712000	ns	
2	D9				
3	34				
4	DA				
5	30				
6	44				
7	52				
8	08				
9	01	Tipo de Frame	1	-	
10	37	Amostra 1	Eixo X	-969	mg
11	FC		Eixo Y	120	mg
12	78		Eixo Z	-102	mg
13	00				
14	9A	Amostra 2	Eixo X	-967	mg
15	FF		Eixo Y	126	mg
16	39		Eixo Z	-87	mg
17	FC				
18	7E				
19	00				
20	A9				
21	FF				
22 ..225

Fonte: elaborado pelo autor, com dados adaptados de Polar Electro Oy (2021a)

Existem três informações importantes no início da mensagem antes das amostras de medições. O primeiro diz respeito ao tipo de dado e o último consiste no tipo do *frame*, ambos que já foram vistos. A informação que precisa ser detalhada é o *timestamp*, que se faz essencial ao se lidar com uma série temporal. Esse dado mostra o tempo decorrido em nanossegundos, utilizando 8 *bytes* para isso.

A referência para a contagem de tempo em dispositivos digitais é quase universalmente a data arbitrária de 1 de janeiro de 1970 00:00 UTC, chamada época (Hauser, 2018). Todavia, a Polar escolheu a data de 1 de janeiro de 2000 00:00 UTC como o marco zero de contagem de tempo em nanossegundos, inicialmente por restrições de memória de seus primeiros dispositivos (Polar Electro Oy, c2023d). Existe uma pequena explicação da contagem de tempo no repositório oficial, em um documento de nome *TimeSystemExplained.md*. A partir dele entende-se que apesar de se ter essa data como base, o valor padrão de fábrica do dispositivo é 1 de janeiro de 2019 00:00 UTC, o que equivale ao valor em nanossegundos de 599616000000000000. A contagem da Polar H10 é reiniciada se o sensor for desconectado da faixa e não houver conexões BLE durante 1 minuto. Esse comportamento também acontece ao trocar a bateria. Outra informação importante é a diferença de tempo entre a época universal e a escolhida, que em nanossegundos equivale a 946684800000000000, que pode ser útil na interpretação.

Por conta da resolução de 16 *bits* enviada na requisição, que é a única suportada pelo modelo H10, cada dado de aceleração ocupa 2 *bytes*. Uma amostra completa constitui-se de uma medida em cada eixo X, Y e Z, e nessa mensagem existem 36 amostras. A uma frequência de aquisição de 200 Hz, cada amostra está espaçada de 5 ms, ou seja, as mensagens chegam a cada 180 ms aproximadamente. Esse é o tempo para o sensor preencher o *buffer* interno e enviar seu conteúdo ao cliente quando esgotar a capacidade.

A Figura 14 ilustra a referência dos eixos em relação à Polar H10.



Figura 14 - Referência dos eixos de acelerômetro da faixa Polar H10 (Polar Electro Oy, 2021c)

A Tabela 15 exemplifica a mensagem de notificação de eletrocardiograma.

Tabela 15 - Notificação de stream de dados de eletrocardiograma

Exemplo de Notificação de Stream ECG				
Byte	Dados	Descrição	Valor	Unidade
0	00	Tipo de Dado	ECG	-
1	D4	Timestamp	599616529178539000	ns
2	B4			
3	23			
4	6F			
5	0E			

Exemplo de Notificação de Stream ECG				
Byte	Dados	Descrição	Valor	Unidade
6	44			
7	52			
8	08			
9	00	Tipo de Frame	0	-
10	B7	Amostra1	-73	uV
11	FF			
12	FF			
13	C0	Amostra 2	-64	uV
14	FF			
15	FF			
16 .. 228

Fonte: elaborado pelo autor, com dados adaptados de Polar Electro Oy (2021a)

O início da mensagem segue a mesma estrutura discutida para o acelerômetro. Para as amostras, cada uma ocupa 3 *bytes* de dados, o que é incomum por não ser um tamanho em potência de 2. Essa diferença será refletida na forma de interpretá-los nas seções responsáveis pela sua visualização. A tabela está em um formato resumido, e ao todo engloba 73 amostras. Como a amostragem é de 130 Hz, as amostras distam de 7,69 ms cada uma, em relação ao *timestamp* inicial. Dessa forma, uma mensagem é enviada pela faixa peitoral a cada 562 ms, algumas vezes mais lento do que o tipo anterior. A quantidade de amostras na mensagem depende do MTU da conexão. Por exemplo, para um MTU de 70, que é o mínimo suportado, as mensagens levam 19 amostras (Polar electro Oy, 2022c).

Por fim, resta apenas testar na prática a conexão com esse serviço, para garantir que a sua interface está sendo feita da maneira correta e assim se possa passar para as próximas etapas. Voltando ao aplicativo *nRF Connect*, em primeiro lugar é preciso cumprir os pré-requisitos: inscrição para indicações na característica *Ponto de Controle PMD*, inscrição para notificações na característica *MTU de Dados PMD* e definir o MTU de conexão como 232. Assim, veste-se o dispositivo, e o conecta com o programa.

A Figura 15 ilustra o início do processo. Para fazer isso no aplicativo, dentro do serviço de UUID de início *fb005c80*, inscreve-se para as indicações na característica de UUID iniciando com *fb005c81*, clicando no ícone com as duas setas em sentidos opostos. Na característica de UUID iniciando com *fb005c82*, as notificações são ativadas ativando o ícone com três setas para baixo.

Já a Figura 16 indica como mudar o MTU. No botão com os três pontos ao lado de *SERVER* na imagem, existe um menu com a opção *Request MTU*, que abre o menu ilustrado. Cadastrando o 232 desejado, basta requisitar o início das *streams* como nas mensagens vistas anteriormente. Para a construção das tabelas, toda a comunicação descrita foi testada, mas não é de interesse do trabalho ilustrar todo esse processo, apenas a extração dos dados de interesse.

O que se pode observar na Figura 17 é que se pode cadastrar mensagens arbitrárias para se enviar para os dispositivos pareados. Aqui estão salvas em hexadecimal as requisições abordadas, e os comandos foram enviados para a característica *Ponto de Controle PMD*.

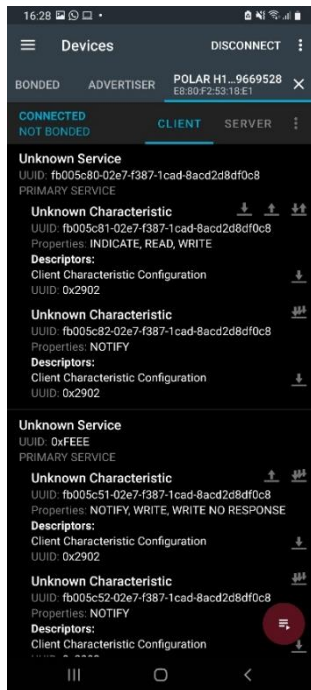


Figura 15 - Inscrição para notificação e indicação

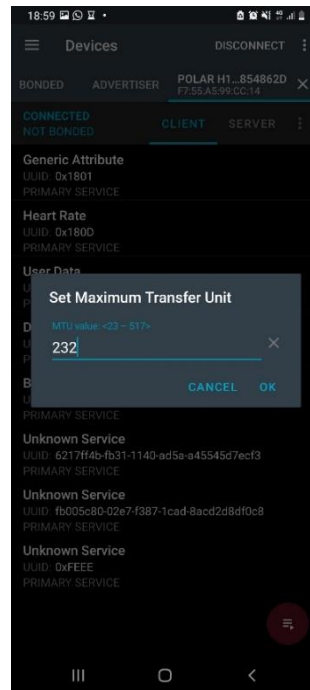


Figura 16 - Configuração do MTU de conexão

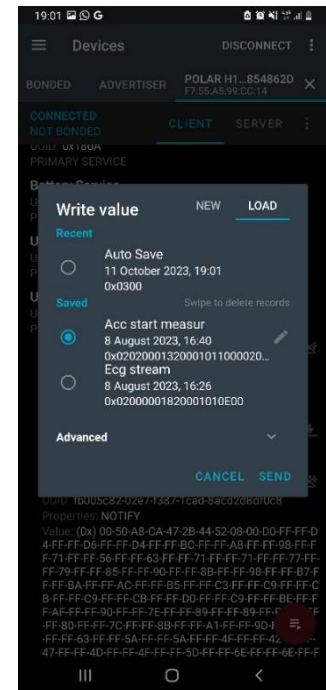


Figura 17 - Cadastro de mensagens arbitrárias

O log de comunicações com essas características está descrito na Figura 18, mostrando em verde a parte dos pré-requisitos, e a última mensagem é do tipo ECG. Na Figura 19 tem-se um exemplo de mensagem de acelerômetro. Lembrando que facilita a diferenciação entre as duas se for observado o primeiro *byte*, que indica o tipo ECG ou ACC.



Figura 18 - Log de pré-requisitos e dados de eletrocardiograma



Figura 19 - Log do acelerômetro

Logo, todo o processo de leitura de dados com a Polar H10 foi descrito e testado. Tem-se um bom conhecimento do seu comportamento e forma de operação, limites e capacidades. O próximo passo é adquirir esses valores em uma plataforma com mais liberdade para o seu uso, análise se necessário e retransmissão.

iv. Desenvolvimento com o ESP32

Todo o protocolo de comunicação com o monitor Polar H10 foi estabelecido. A próxima etapa é selecionar um dispositivo para ler os seus dados e enviá-los a um computador de alguma forma. Esse dispositivo, idealmente, deve ser compacto, acessível tanto em termos financeiros quanto de disponibilidade, ter poder de processamento suficiente e suporte a conexões sem fio. Esse é o papel da central de conexão de acordo com as diretrizes estabelecidas.

Existem várias opções que se encaixam nesses requisitos. Uma delas, que se destaca, é o ESP32. Por se encaixar perfeitamente nas necessidades do projeto, ele foi o dispositivo escolhido. Comparado com outros produtos populares, o ESP32 apresenta mais capacidade de processamento que um *Arduino* e é mais barato e compacto que um *Raspberry Pi*. Lançado como uma versão aprimorada do ESP8266, outro microcontrolador de sucesso, foi também desenvolvido pela empresa *Espressif*. Seu chip foi anunciado em 2016 (Espressif, 2016), contando com especificações que suprem necessidades de projetos pessoais e profissionais (Barbosa *et al.*, 2022).

Todas as informações a seguir foram retiradas do *datasheet* do chip (Espressif, c2023a). O modelo específico implementado foi o *ESP-WROOM-32*:

- Processador de 32 bits *dual core* com *clock* de até 240 MHz;
- Memória *Flash* de 4 MB;
- Memória *SRAM* de 520 kB;
- Compatível com especificações *Bluetooth®* v4.2 BR/EDR e *Bluetooth®* LE;
- Wi-Fi: IEEE 802.11 b/g/n/e/i.

Existem uma série de variações técnicas para as unidades microcontroladoras (MCUs), e novas versões são constantemente lançadas. Outro detalhe importante e ser ressaltado é que os módulos com chips são frequentemente vendidos em *PCBs*, os quais facilitam o processo de desenvolvimento, como o adquirido *DOIT ESP32-DEVKIT-V1*. As vantagens de comprar o kit pronto são, principalmente, a inclusão de um porta micro USB para programar o microcontrolador, o regulador de tensão e o *layout* de portas compatível com *protoboards*.

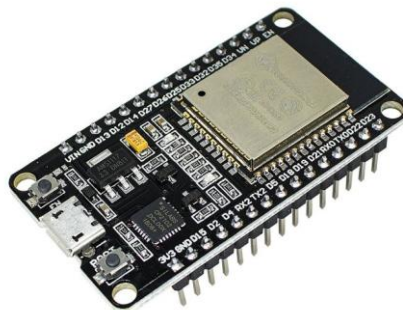


Figura 20 - Placa DOIT ESP32-DEVKIT-V1, que apresenta um chip ESP-WROOM-32 (Placa [...], [s.d.])

Na placa existe um botão com referência *EN* que serve para reiniciar o programa salvo, e o botão de nome *BOOT* tem o seu uso voltado para colocar a placa em modo de receber um novo programa. Para mais detalhes, a Figura 20 ilustra o produto escolhido. Nela percebe-se a clara diferença entre o chip e a placa. Os botões mencionados estão respectivamente à esquerda e à direita da entrada micro USB. São introduzidos também dois LEDs, um para indicar a energia e outro de uso livre pelo programa.

O chip com formato quadrado e de cor preta é o controlador FTDI, que converte o protocolo USB para o UART usado pelo chip. Inicialmente, ao ser conectado, o FTDI apresentou questões de *driver* no Windows 10, sistema operacional usado para desenvolver todas as partes do projeto. Para alcançar a sua detecção da forma correta foram consultados arquivos do site oficial da *Silicon Labs* que auxiliaram na solução do problema.

Oficialmente a *Espressif* suporta um *framework* para o desenvolvimento em suas placas chamado *esp-idf*, feito boa parte em *Python*, e voltado para o uso com C/C++. No próprio site existe a referência dos comandos suportados e a documentação (*Espressif, c2023b*). Além disso, no mesmo acesso, também há um conjunto de bibliotecas que permitem a integração dos modelos *ESP* para programação no ambiente de desenvolvimento do *Arduino*, o que é preferível por conta de certa familiaridade com a *IDE* e fácil configuração. Dito isso, os únicos passos necessários são cadastrar um link para o ambiente buscar as placas e em seguida instalá-las, o que proporciona acesso a uma série de pacotes desenvolvidos tanto pela empresa quanto pela comunidade. Guias para esse processo também são encontrados através do link.

Vale mencionar que a versão instalada do ambiente foi a *Arduino IDE 1.8.19*, que se denomina a última versão “legado” da plataforma. Ainda que a versão mais recente no momento de desenvolvimento do projeto, em dezembro de 2023, seja a 2.2.1, ela também foi testada e apresentou performance consideravelmente mais lenta do que a anterior no computador usado. Por isso, escolheu-se seguir com a primeira, com recursos mais básicos e maior responsividade.

Até então, tem-se o que é preciso para se começar a abordar a programação implementada no ESP32. A biblioteca para BLE que vem com os pacotes do microcontrolador foi inicialmente desenvolvida por um membro da comunidade, Neil Kolban, por volta de 2018 (Kolban, c2023). Posteriormente, a *Espressif* a incorporou no seu repositório oficial do *GitHub*, e algumas funcionalidades a mais foram implementadas, divergindo, mas definitivamente superando a original. O programador chegou a publicar um livro que comenta o seu desenvolvimento dessa e de outras bibliotecas, sob o nome de *Kolban's book on ESP32*. Dessa forma, como referência de material para compreender e implementar um código com objetivos do projeto, foram usados o livro (cujo capítulo sobre o BLE é disponibilizado gratuitamente no acesso do *GitHub*), o código fonte da biblioteca oficial e os seus exemplos. O intuito da implementação do Kolban ao desenvolver o pacote foi justamente desenvolver um código em C/C++ compatível com *Arduino*, que se executa por cima do *framework esp-idf*, tornando o uso do BLE mais simples, compacto e acessível. Essa abordagem é ideal para projetos que não são extremamente complexos.

O seu funcionamento ficará claro conforme for abordado o seu uso no desenvolvimento do trabalho. A Figura 21 mostra todos os arquivos desenvolvidos para o microcontrolador, ilustrando as pastas e os seus conteúdos com barra e tabulações, respectivamente. Foram desenvolvidos dois programas, um com nome *Polar H10_Complete*, que constitui o código principal para a versão finalizada do projeto, e o outro chamado *Polar H10_Simulation*, cujo intuito é testar o comportamento do mesmo.

```
ESP32/
|
+---PolarH10_Complete/
|       PolarH10_Complete.ino
|
+---PolarH10_Simulation/
|       Constants.h
|       HelperFunctions.cpp
|       HelperFunctions.h
|       PolarH10_Simulation.ino
```

Figura 21 - Estrutura de arquivos dos programas do ESP32

Começa-se então a explorar o programa principal. Para o seu desenvolvimento, o primeiro passo é importar as bibliotecas usadas e definir as constantes e variáveis globais que serão manipuladas pela lógica implementada. Todo o pacote das funções BLE são governados por classes e ponteiros, definidos no código fonte em vários arquivos dependendo do seu propósito. A Figura 22 mostra as declarações referentes a essa etapa. Nas linhas 13 e 14 do código, as funções referentes ao *scan* BLE e ao dispositivo BLE são importadas, e apenas esses são necessários, já incluindo outras estruturas. No projeto, estavam disponíveis duas faixas Polar H10 para uso, e nas linhas 16 e 17 percebe-se os nomes de cada uma. Por padrão, esses *wearables* utilizam essas identificações no anúncio BLE para outros dispositivos. O código alfanumérico se localiza impresso na parte de cima dos sensores, referenciado como *ID*, facilitando a diferenciação entre ambos. Durante o desenvolvimento deste trabalho, por usos não relacionados, o segundo dispositivo apresentou mau funcionamento, e deixou de ser usado.

Ainda na mesma figura, são descritas as suas características a partir dos seus *UUIDs*, código que pode ser identificado por apenas 2 *bytes* para as características padrão, mas que para as customizadas são necessários todos os *bytes*. Por fim, alguns dos comandos relativos ao controle das *streams* de dados foram declaradas, pelos seus valores brutos em hexadecimal.

```

13 #include "BLEDevice.h"
14 #include "BLEScan.h"
15
16 String SensorName = "Polar H10 B9669528";
17 //String SensorName = "Polar H10 A854862D";
18
19 // UUIDs dos serviços e características da Polar H10
20
21 static BLEUUID heartRateServiceUUID("180D");
22 static BLEUUID heartRateCharUUID("2A37");
23 static BLEUUID batteryServiceUUID("180F");
24 static BLEUUID batteryCharUUID("2A19");
25 static BLEUUID pmdServiceUUID ("FB005C80-02E7-F387-1CAD-8ACD2D8DF0C8");
26 static BLEUUID pmdDataCharUUID ("FB005C82-02E7-F387-1CAD-8ACD2D8DF0C8");
27 static BLEUUID pmdControlCharUUID ("FB005C81-02E7-F387-1CAD-8ACD2D8DF0C8");
28
29 // PolarH10 Comandos do Ponto de Controle PMD
30
31 const uint8_t REQUEST_ECG_SETTINGS[] = {0x01, 0x00};
32 const uint8_t REQUEST_ACC_SETTINGS[] = {0x01, 0x02};
33 // Stream ECG com 130 Hz de taxa de amostras e 14 bits de resolução
34 const uint8_t START_ECG_STREAM[] = {0x02, 0x00, 0x00, 0x01, 0x82, 0x00, 0x01, 0x01, 0x0E, 0x00};
35 // Stream ACC com 200Hz de taxa de amostras, 8g de alcance e 16 bits de resolução
36 const uint8_t START_ACC_STREAM[] = {0x02, 0x02, 0x02, 0x01, 0x08, 0x00, 0x00, 0x01, 0xC8, 0x00, 0x01, 0x01, 0x10, 0x00};
37 const uint8_t STOP_ECG_STREAM[] = {0x03, 0x00};
38 const uint8_t STOP_ACC_STREAM[] = {0x03, 0x02};

```

Figura 22 - Início do código principal em Polar H10_Complete.ino

Existem ainda valores globais do programa para serem definidos, o que está ilustrado na Figura 23. Nela, estão separados de acordo com o seu propósito, o que é indicado pelos comentários. Inicialmente, são definidos um valor para armazenar o nível da bateria do *wearable* e a duração máxima que um *scan* BLE pode assumir, ou seja, 10 segundos. Esse segundo valor é a única constante nesta seção, já que todas as outras serão atribuídas ao longo da execução do código. Em seguida, nas linhas 47 e 48, duas *flags* ditam o comportamento do programa para conectar à Polar H10 e, caso a conexão seja encerrada ou perdida, buscar por uma novamente. Esse comportamento será analisado no *loop* em breve. De resto, são declarados ponteiros para classes do cliente, do servidor e do *scan* BLE, assim como as características remotas, que exercem papéis cruciais. Essas últimas não necessariamente precisam ser globais, mas essa forma facilita o envio de comandos e leituras esporádicas que se deseje efetuar eventualmente, inclusive para testes.

```

40 //-----
41 // Parâmetros do programa
42
43 uint8_t batteryLvl = 0;
44 const int BLEScanTime = 10; // Tempo de scan BLE em segundos
45
46 // Flags
47 static boolean doConnect = false;
48 static boolean BLEConnected = false;
49
50 // Declaração das estruturas principais do BLE
51 static BLEAddress* pServerAddress;
52 static BLEClient* pClient;
53 BLEScan* pBLEScan;
54
55 // Declaração das características usadas para se interagir com os dados
56 static BLERemoteCharacteristic* pBatteryCharacteristic;
57 static BLERemoteCharacteristic* pHeartRateCharacteristic;
58 static BLERemoteCharacteristic* pPmdControlCharacteristic;
59 static BLERemoteCharacteristic* pPmdDataCharacteristic;

```

Figura 23 - Fim das declarações iniciais do código principal

Parte-se então para as duas funções mais básicas de qualquer programa em Arduino, a primeira denominada *setup*, executada uma única vez, e a outra, *loop*, chamada repetidamente enquanto o ESP32 estiver ligado. A inicialização da porta *Serial* no programa é importante para transmitir os dados do ESP para o computador na última etapa do projeto. Em seguida, o BLE é inicializado no ESP32, e são efetuadas configurações de escaneamento. O que acontece ao iniciar uma busca por dispositivos é que o microcontrolador executa uma função padrão para cada aparelho encontrado, chamada *callback*. Na linha 185 essa função é sobrescrita por uma outra, que lida com os resultados da busca da maneira ideal, e será abordada em outro momento. Normalmente, o escaneamento é feito de forma passiva, mas como se deseja requisitar informações de cada servidor BLE, a linha 186 garante que seja feito um *scan* ativo. Essa lógica é apresentada na Figura 24.

```

180 void setup() {
181
182     Serial.begin(115200);
183     BLEDevice::init("");
184     pBLEScan = BLEDevice::getScan();
185     pBLEScan->setAdvertisedDeviceCallbacks(new MyAdvertisedDeviceCallbacks());
186     pBLEScan->setActiveScan(true);
187 }

```

Figura 24 - Função *setup* do programa, executado uma vez

Ao se conectar a um dispositivo com uma conexão sem fio, o ideal é que sejam implementadas formas de se lidar com a perda de sinal. Os biossinais adquiridos podem ser encerrados por parte do dispositivo encerrando a comunicação ou por conta de falha na transmissão, que por sua vez acontece devido a ruído, falta de alcance, entre outros. O cliente implementado no ESP32 tem uma lógica para lidar com isso. As configurações básicas são inicializadas no *setup*, mas a parte de reconexão está no *loop*. Dois estados são usados para controlar os procedimentos, um que sinaliza se existe uma conexão cliente-servidor, e outro para indicar que deve ser iniciada uma conexão. Com essa ideia apenas são cobertos os casos mencionados. A lógica é a seguinte: se o estado indicar que o ESP32 estiver conectado com o monitor Polar H10, precisa-se verificar se isso continua verdade. Em caso contrário, é verificado o estado que indica se deve ser feita uma conexão. Para esse último teste, se positivo, há uma requisição, e no caso negativo o escaneamento por servidores é iniciado. A situação com todos os testes falsos é a inicial, como seria desejado.

Na Figura 25, está explicitado como esse procedimento foi implementado no código, com os estados, ou *flags* de controle definidos como *BLEConnected* e *doConnect*, com nomes autoexplicativos. A duração de um escaneamento foi arbitrada com 10 segundos, mas pode durar menos se o *wearable* for encontrado antes. Essa dinâmica dos resultados da busca e a função para iniciar uma conexão, que não estão explicadas na figura, terão o foco no momento.

```
189 void loop() {
190     // Verifica se já está conectado, e se a conexão se mantém
191     if (BLEConnected) {
192         if (!(pClient->isConnected())) {
193             BLEConnected = false;
194         }
195         // Inserir aqui lógica adicional de código após o BLE conectado, se necessário
196     }
197     else {
198         // Verifica se é necessário efetuar uma conexão
199         if (doConnect) {
200             if (connectToServer(*pServerAddress)) {
201                 BLEConnected = true;
202             }
203             doConnect = false;
204         }
205         // Escaneia dispositivos próximos
206         else {
207             pBLEScan->start(BLEScanTime);
208             delay(100);
209         }
210     }
211 }
```

Figura 25 - Função loop do programa, executada inúmeras vezes

Para cada dispositivo encontrado em uma busca, o ESP32 requisita o seu nome, e verifica se é o mesmo definido anteriormente, uma faixa Polar H10 com *ID* único. Ao confirmar o servidor desejado, o escaneamento é encerrado, o endereço do servidor se armazena em uma variável global e o microcontrolador fica pronto para iniciar a requisição de conexão. Essa prontidão é determinada pelo estado de controle mencionado anteriormente, que é modificado nesta parte. Na Figura 26 explicita-se a função que lida com o processo aqui descrito, e que foi sinalizada primeiramente na Figura 24.

```

126 class MyAdvertisedDeviceCallbacks: public BLEAdvertisedDeviceCallbacks {
127
128     // Chamada para cada servidor encontrado
129     void onResult(BLEAdvertisedDevice advertisedDevice) {
130         // Checa se o nome do servidor corresponde ao nome do dispositivo
131         if (SensorName.equals(advertisedDevice.getName().c_str())) {
132             advertisedDevice.getScan()->stop();
133             pServerAddress = new BLEAddress(advertisedDevice.getAddress());
134             doConnect = true;
135         }
136     }
137 };

```

Figura 26 - Função callback do escaneamento BLE

Uma vez obtido o endereço do servidor, inicia-se o processo de comunicação com a Polar H10, que até então se comporta no modo anúncio. Isso é feito em uma única função, a qual se divide em três etapas para esse processo. Em primeiro lugar, é definido o cliente, que é a estrutura que gerencia as conexões, e necessita apenas do endereço. Uma vez conectado, define-se imediatamente o MTU para 232.

A segunda etapa diz respeito a, partindo-se dos *UUIDs*, buscar os serviços e destes obter as características da faixa peitoral. No código a aquisição foi implementada com alguns mecanismos de detecção de erro, conferindo se cada parte foi corretamente adquirida. Por mais que erros sejam incomuns durante esse processo, blindar o programa para suas ocorrências garante que o ciclo de tentativa de conexão não seja quebrado, mantendo o funcionamento esperado.

Em seguida, a terceira etapa consiste em registrar as características para notificações e indicações, atribuindo para cada uma função *callback*, ou seja, uma forma de lidar com as respostas da Polar H10. Como forma de segurança, verifica-se se essas características oferecem essas capacidades antes de requisitá-las. Dessa forma, tem-se um tratamento específico para cada tipo de mensagem recebida pelo microcontrolador. Finalmente, é feita uma leitura inicial da bateria, já que a inscrição por si só não fornece os valores. Junto a isso, requisita-se o início de ambas as *streams* de ACC e ECG.

A biblioteca oferece métodos específicos para essas operações, deixando o usuário alheio ao que acontece implicitamente. Para habilitar notificações escreve-se (0x) 01 00 no descritor *Client Characteristic Configurator* (UUID 0x2902). Já para uma indicação, escreve-se (0x) 02 00. Para desabilitar ambas, escreve-se (0x) 00 00.

A Figura 27 mostra o começo da implementação da função descrita, apenas a primeira etapa. O resto, por ser mais extenso e seguir ideias já abordadas, pode ser conferido no programa em si enviado em conjunto com este trabalho.

```

63 bool connectToServer(BLEAddress pAddress) {
64
65     BLERemoteService* pBatteryService;
66     BLERemoteService* pHeartateService;
67     BLERemoteService* pPmdService;
68
69     // Cria cliente e conecta com o endereço do servidor encontrado na busca
70     pClient = BLEDevice::createClient();
71     pClient->connect(pAddress, (esp_ble_addr_type_t)1);
72     pClient->setMTU(232);

```

Figura 27 - Conexão com o servidor

A parte final do programa refere-se ao envio dos dados para o computador, que representa o terceiro elemento dos objetivos, e será justificado na seção seguinte. Para isso, torna-se necessário um protocolo de comunicação definido, conhecido por ambas as partes. Existem quatro tipos de mensagem recebidos pelo ESP32: frequência cardíaca, eletrocardiograma, acelerômetro e bateria. Cada uma delas possui um tempo de envio determinado pelo servidor, que não pode ser alterado. Assim, uma solução possível é tentar juntar os dados compatíveis em mesmos pacotes, e definir formas de se identificar e diferenciá-los. Como as mensagens de ACC e ECG possuem forte dependência com o tempo, apresentam tamanho grande e são recebidas com frequência, optou-se por mantê-las independentes. Já a FC com os R-Rs e a bateria não sofrem nenhum desses problemas. Assim, podem ser juntos em apenas um envio para o computador, inserindo a bateria no fim dos dados de FC, e a saída do ESP32 lida então com três mensagens.

O método escolhido foi definir uma sequência de *bytes* introduzida antes de cada mensagem, indicando informações como *bytes* de início, tipo de mensagem e tamanho da mensagem. Para a aplicação, ao menos em seu estado inicial julgou-se suficientes essas informações apenas, e não colocar *bytes* de fim. Nesse raciocínio, o tipo de mensagem dessa espécie de cabeçalho introduzido pode incorporar os mesmos tipos dos dados de *stream*, 0 e 2, e para a mensagem de FC com bateria usa-se o tipo 1. Nota-se que se está fazendo uma diferenciação entre os tipos de dado do protocolo da Polar H10 e o tipo de dado do protocolo construído no projeto, no sentido de que a representação de um não está sendo repassada adiante, mas ambas coincidem em dois sinais. Implementou-se dessa forma para não haver conflitos.

Idealmente os *bytes* de delimitação de comunicações não devem aparecer na mensagem. Escolheu-se 0x50 0x75, em hexadecimal, para esse propósito. Pode parecer arbitrário, e de certa forma o é, mas informações envolvidas possuem peculiaridades que podem ser aproveitadas. Focando nas acelerações presentes na mensagem, sabe-se que o maior valor possível é de 8000 mm/s², e como está em 2 *bytes*, a representação vai de -32,768 a 32,767. Boa parte desses valores nunca são assumidos. O mesmo acontece nas mensagens de eletrocardiograma, que apresenta 3 *bytes* disponíveis, mas os limites de medição são de -20.000 a 20.000. Nota-se que, da forma como é usado, a representação anterior poderia ser tranquilamente usada aqui também. O que acontece na prática é que o terceiro *byte* de uma amostra vale 0x00 quando ela for positiva (ou zero) ou 0xFF quando for negativa.

Assim, o início escolhido para a informação equivale a 30.032 com a notação *little endian*, ou seja, nunca poderia aparecer em uma amostra de *stream*. Não basta se contentar com apenas esse fato, pois nada garante que a interpretação será feita respeitando a região das amostras. Por exemplo, é válido supor que a referência estipulada possa aparecer entre amostras, com o 0x50 ocupando o *byte* mais significativo de uma e 0x75 como o menos significativo da seguinte. Ainda assim, não há possibilidade de erro nesse caso, pois se o segundo *byte* menos significativo em qualquer caso for 0x50 já é sinal de que o valor representado é maior ou igual a 20.480. Para completar, como o *bit* mais significativo é zero, não pode ser confundido por um número negativo na notação de complemento a dois.

Abaixo, nota-se pela Figura 28 como se dá essa comunicação. As mensagens da Polar H10 são copiadas para o final de estruturas maiores, contendo as informações estabelecidas no início. O tamanho introduzido no cabeçalho é referente à mensagem original, e foram usados dois *bytes* para a sua representação. Apenas um já daria conta das necessidades básicas, mas com uma margem pequena, então pensando em futuros tipos de mensagem optou-se por fazer dessa maneira.

```

214 void sendMsgWithHeader(uint8_t* msg, uint16_t len, uint8_t type) {
215     int totalSize = len + 5;
216     uint8_t msgTemplate[totalSize];
217     msgTemplate[0] = 0x50;
218     msgTemplate[1] = 0x75;
219     msgTemplate[2] = type;
220     memcpy(&msgTemplate[3], msg, sizeof(uint16_t));
221     memcpy(&msgTemplate[5], msg, len);
222
223     Serial.write(msgTemplate, totalSize);
224 }

```

Figura 28 - Função para estruturar a mensagem de saída do ESP32

A função recebe o valor original bruto, sua extensão e o tipo, faz o processamento já descrito e por fim envia os dados via serial pela porta USB. O objetivo é implementá-la para cada função *callback* das notificações, de modo que assim que o microcontrolador recebe a informação começa a processá-la e a repassa. Na verdade, apenas a notificação de bateria não a usa, e atualiza o valor recebido em uma variável global.

Todo o funcionamento do programa principal do ESP32 foi descrito. Porém, durante o seu desenvolvimento foram necessárias várias outras funcionalidades que não são utilizadas na sua versão final. Não só isso, como o ESP32 serve como uma ponte entre o computador e a faixa peitoral, o produto necessita ter sempre os três dispositivos, mas também foram feitos testes dois a dois com os dispositivos (ESP32 – Polar H10 e ESP32 - Computador). É com esse intuito que se desenvolveu o segundo programa, cuja estrutura é mostrada na Figura 21. Com o nome *Polar H10_Simulation*, possui dois objetivos, sendo um deles testar a interpretação de mensagens, e o outro simular o comportamento do monitor cardíaco para testar a interface gráfica quando o mesmo não estiver disponível.

Assim, o arquivo de constantes guarda mensagens genéricas conhecidas do sensor, usadas como teste. São implementadas uma série de funções para traduzir as informações de hexadecimal para valores úteis em decimal, de modo que se possa verificar o funcionamento do programa. Outras funções são usadas justamente para a visualização dos dados. Tem-se então o procedimento completo de codificação e decodificação das mensagens.

Para simular o comportamento da Polar H10 usou-se as mesmas mensagens de ACC e ECG em todos os envios para poupar tempo, com um nível de bateria fixo também. Os dados de frequência cardíaca foram gerados aleatoriamente podendo cobrir todas as possibilidades, ou seja, existem casos com a FC assumindo um ou dois *bytes*, e os intervalos R-R podem não estar presentes ou incluir nove amostras. Para o próximo passo, esse programa será bastante útil.

c. Interface Homem-Máquina

i. Visualização dos dados na interface gráfica

Com o que se desenvolveu até então, a próxima etapa consiste em definir a forma de exibição de dados, conforme os objetivos. Decidiu-se por praticidade e alcance de uso, estruturar uma Interface HM para mostrar em um computador de forma clara e intuitiva os biossinais requisitados. O aparelho foi escolhido por seu relativamente fácil acesso e pela vasta disponibilidade de material sobre visualização de dados. Após diversas pesquisas, o framework escolhido foi o Qt. Os pontos mais importantes para determinar essa escolha serão melhor comentados abaixo.

O Qt é um framework de desenvolvimento de aplicativos amplamente utilizado, que oferece um conjunto abrangente de ferramentas para criar interfaces gráficas de usuário (GUIs) e outros tipos de *software* em várias plataformas. Ele suporta uma ampla gama de sistemas operacionais, incluindo *Windows*, *MacOS*, *Linux*, *iOS*, *Android* e outros sistemas. Isso permite que os desenvolvedores escrevam um código fonte que pode ser compilado e executado em diversas plataformas, proporcionando uma experiência com componentes nativos agradáveis para o usuário. Embora seja escrito em C++, ele pode ser usado com algumas linguagens de programação, incluindo *Python* (através de *bindings* como *PySide* e *PyQt*), *Ruby*, *Perl* e outras, garantindo a sua alta performance. Desenvolvido pela *The Qt Company*, foi apresentado pela primeira vez na década de 90, mas vem sendo repaginado desde então. Atualmente, está na sua sexta versão, *Qt6*, lançada em 2020 (Qt, 2017; Fitzpatrick, 2022).

O *PySide6* e o *PyQt6* são bibliotecas de interface gráfica de usuário para *Python* que oferecem uma maneira de criar aplicativos de desktop modernos e atraentes. Eles são, respectivamente, as versões mais recentes do *PySide* e *PyQt* e são baseados no *Qt6*. Vale lembrar que o *PySide6*, framework escolhido para o projeto, é a versão oficial do Qt para *Python*. Ele é lançado sob a Licença LGPL (*GNU Lesser General Public License*), o que o torna atraente para muitos projetos, pois oferece mais flexibilidade em termos de licenciamento em comparação com o *PyQt*. Apesar de ambos os *bindings* terem bom suporte e documentação, por ser o oficial, o *PySide6* estará mais alinhado com as atualizações e mudanças

recentes do Qt. Além disso, por esta mesma razão, tem suporte direto dos desenvolvedores do Qt e uma comunidade crescente de usuários e contribuidores (Chaetognathan, 2023; Qt, 2017).

Empresas em mais de 70 áreas da indústria fazem uso do software. Entre as principais companhias e organizações que incorporaram o Qt em seus produtos estão *Siemens, Autodesk, AMD, Volvo Mobility Systems* etc. (Qt, 2017).

Em comparação com outros *frameworks* conhecidos, como o *Tkinter*, embora esse seja uma biblioteca padrão do *Python*, fornece uma gama de elementos mais limitada e uma aparência menos moderna em comparação com os outros. Além disso, o *PySide6* oferece melhores ferramentas para desenvolvimento de aplicações complexas e visualmente ricas, como o *Qt Designer* (Chaetognathan, 2023; Fitzpatrick, 2022).

A realidade é que há opções de acordo com cada caso de uso, como aplicações mais simples, rápidas de implementar, portabilidade, suporte *mobile*, foco em embarcados, entre outros. O usado no projeto tem o uso mais comum na área científica e de engenharia de *software* (Chaetognathan, 2023).

Após as devidas introduções, começa-se a descrever em aspectos técnicos a implementação da interface. A versão do *Python* usada foi a 3.10.0, em um computador com o *Windows 10*. Como ambiente de programação, escolheu-se o *VSCode*, e como ambiente virtual da linguagem o *venv*. Abaixo, na Figura 29 está a estrutura do projeto.

```

python-gui-esp32-polar/
|
| .gitignore
| app.py
| LICENSE
| README.md
| requirements.txt
|
+---.venv/
|
+---assets/
|     green-circle.svg
|     icon.ico
|     icon.svg
|     logo.svg
|     red-circle.svg
|
+---layouts/
|     main_layout.py
|
+---mocks/
|     ECG_ACC_Raw_Data.txt
|     ECG_Raw_Data.txt
|
+---modules/
|     circular_buffer.py
|     file_reader.py
|     helper_functions.py
|
+---widgets/
|     custom_charts.py
|     serial_thread.py

```

Figura 29 - Estrutura do projeto da interface gráfica

A raiz do projeto inclui alguns arquivos importantes. O *.gitignore* é um arquivo padrão do *GitHub*, mas nesse caso serve para impedir que o ambiente virtual, presente na pasta *.venv* seja vinculado ao repositório, o que é uma prática comum, e o deixa mais enxuto. O *README.md* descreve o projeto e mostra o passo a passo de como executá-lo, servindo para fornecer informações ao repositório do código. O arquivo de licença foi definido para a *GNU GENERAL PUBLIC LICENSE*, o que garante o livre acesso e uso do que foi desenvolvido. Por fim, o *requirements.txt* contém informações sobre todos os pacotes instalados, que no caso foram o *PySide6* e as suas dependências. Pode ser fornecido junto à criação do ambiente virtual para que se instalem exatamente as mesmas versões usadas neste trabalho, o que impede possíveis erros e incompatibilidades decorrentes de atualizações futuras. O script *app.py* é o mais importante, pois representa a aplicação principal.

Na pasta *assets*, estão todos os recursos gráficos, como logos e ícones usados no projeto, que são exibidos na Figura 30.



Figura 30 - Recursos gráficos externos usados na interface

Ambos os círculos foram criados para o projeto, com o propósito de indicar o estado da conexão com o microcontrolador. A terceira imagem serve como o ícone do projeto, tanto na barra de tarefas do sistema operacional quanto, ao menos no caso do *Windows*, para o ícone na janela do programa. Para esse caso, a partir da figura, foi gerado um arquivo com extensão *.ico*, que segue alguns padrões para a aplicação. Este é um ícone gratuito disponível no site da *fontawesome*, uma biblioteca que disponibiliza milhares de figuras para uma gama de ocasiões. Pode ser encontrado em (Fonticons inc., c2023). Sua cor foi ajustada para a primária usada no projeto, de código #326496. O logo do *Tecgraf* da PUC-Rio foi utilizado porque contribuiu para o projeto, disponibilizando os sensores Polar H10 para serem usados nos testes.

Os demais arquivos e pastas serão abordados conforme for descrito o funcionamento do programa.

Para compreender a lógica de desenvolvimento, são necessários dois conceitos fundamentais ao *framework*: o de *widgets* e o de *sinais-slots*. Os *widgets* são componentes base de construção de uma GUI. São elementos como rótulos, botões, janelas, caixas de texto, etc., que podem ser agrupados e inclusive contidos em outros. São fornecidos alguns prontos com funcionalidades específicas, mas podem ser totalmente customizáveis pelo programador. Já o outro conceito é responsável por conectar as unidades do código, de modo que se opere com uma dinâmica governada por eventos, e cada parte possa lidar com uma função específica. Nesse sentido, os *sinais* são indicações ativas de um componente para certos eventos, podendo enviar dados ou não. Os *slots* estão presentes em outros componentes, e são implementados para receber essas notificações.

O arquivo *app.py* detém todas as instâncias de *widgets* na aplicação, e cabe decidir quais tipos devem ser usados no caso. Já se conhecem os dados a serem exibidos, então o primeiro passo é definir como fazer isso de forma útil e agradável ao usuário. Informações tais quais a frequência cardíaca, nível da bateria e intervalos R-R podem ser mostrados em caixas de texto genéricas em um primeiro momento, com rótulos indicando o dado e a unidade. Nada impede que, posteriormente, a visualização seja aprimorada. As *streams* de dados objetivamente necessitam de gráficos para que sejam observadas com clareza.

Dessa forma, o ciclo de operação que dever ser feito começa a ficar claro. Precisa-se detectar as mensagens advindas da porta serial, processá-las e exibi-las na tela. Pela quantidade de informação, principalmente das amostras transmitidas em grandes pacotes, pode ser que a demanda computacional seja demais para uma instância lidando com todas as operações. Por isso, o funcionamento do programa foi dividido em *threads* separadas, o que significa que o processamento pode ocorrer de forma paralela. As primeiras duas etapas, ou seja, identificar e decodificar os dados do microcontrolador ocorrem simultaneamente com a exibição de dados na tela. Essas são as principais tarefas que mais demandam do processador, e a sua divisão garante um ganho de desempenho considerável, deixando a aplicação mais leve.

A partir de um *widget* pronto que cria uma *thread* separada, foi implementado um código para instanciar a comunicação serial. Nele, são definidos *sinais* para todos os tipos de dados lidos, emitidos à medida que são recebidos e decodificados. Também há eventos para a conexão e desconexão com o microcontrolador. Naturalmente, depende de informações que são enviadas pela parte gráfica, no que consiste em informações de porta, *baudrate* (taxa de símbolos) e, no caso, os *bytes* de início também. Existe um modo de testes em relação a esse componente, comentado ao final da seção.

Quando a *thread* for iniciada, a comunicação serial também é instanciada. Fica sempre analisando os bits que chegam, em especial os de início informados, 0x50 0x75, repetidos aqui por conveniência. A determinação de um *timeout* se mostrou essencial nesse momento, pois caso não existisse a *thread* se tranca indefinidamente enquanto não houver mensagens, e se tem dificuldade para encerrar a comunicação. Na versão final, um segundo é aguardado até que se passe para a próxima instrução e a serial termine corretamente.

A Figura 31 ilustra a abordagem escolhida. Essa parte do código fica sempre aguardando informação. Ao detectar uma mensagem, extrai o tipo e o tamanho a partir do cabeçalho definido no ESP32. Assim, sabe-se a quantidade certa de *bytes* a ser lida, e logo após essa etapa decodifica-se a mensagem. Qualquer erro encontrado é lido de forma que não prejudique a execução do resto do programa, e para que se possa tentar iniciar a conexão novamente.

```

42     def run_serial(self):
43         try:
44             self.serial = serial.Serial(self.port, self.baudrate, timeout=1)
45             if(self.serial.is_open):
46                 self.serial_connected.emit()
47             while self.is_running:
48                 data = self.serial.read_until(self.start_of_msg)
49                 if self.start_of_msg in data:
50                     msg_type = self.serial.read(1)
51                     msg_len_raw = self.serial.read(2)
52                     msg_len = int.from_bytes(msg_len_raw, byteorder='little')
53                     hex_data = self.serial.read(msg_len)
54                     self.process_data(hex_data, msg_type)
55
56         except Exception as e:
57             print("Erro na thread serial:", str(e)) # Lida com exceções, se ocorrerem
58         finally:
59             if self.serial and self.serial.is_open:
60                 self.serial.close()
61                 self.serial_disconnected.emit()

```

Figura 31 - Detecção de mensagens na interface

O processamento do que foi recebido pode ser visto na Figura 32. A partir do tipo, as amostras são decodificadas para seus valores decimais e, por sua vez, emitidas para a tela principal. O evento que acontece nesta seção passa os dados de uma *thread* para outra, amarrando ambas as partes. Em seguida os métodos dos gráficos são chamados para os atualizarem.

O detalhamento das funções de decodificação segue o mesmo raciocínio das usadas no ESP32, com diferença apenas na linguagem de programação. Não serão abordadas neste documento, mas também estão disponíveis junto ao resto do código para serem conferidas. A ideia é justamente seguir o processo inverso da construção das mensagens.

```

26     def process_data(self, hex_data, msg_type):
27         if msg_type == b'\x00': # ECG
28             data = extract_ecg_values(hex_data)
29             self.ecg_data_received.emit(data)
30         elif msg_type == b'\x02': # ACC
31             data = extract_acc_values(hex_data)
32             self.acc_data_received.emit(data)
33         elif msg_type == b'\x01': # FC
34             hr_data, rri_data, battery_data = extract_hr_values(hex_data)
35             self.hr_data_received.emit(hr_data)
36             self.rri_data_received.emit(rri_data)
37             self.battery_data_received.emit(battery_data)
38         else:
39             print('Tipo de mensagem não reconhecido')
40             pass

```

Figura 32 - Processamento de mensagens na interface

Antes de se entrar em detalhes nos gráficos, que são os segundos componentes mais importantes, vale mencionar outra otimização. Tem-se um fluxo de informação ligeiro e contínuo, o que justifica uma abordagem de “janela deslizante” para exibi-lo. uma técnica mais eficiente do que a do senso comum, que seria descartar o começo de uma lista e reatribuir a sua fatia mais recente que não extrapola o tamanho máximo. Para evitar essa operação custosa a cada elemento, criou-se um *buffer* circular simples, com extensão pré-definida. Foi usado o *numpy*, uma biblioteca mais eficiente que o *Python* nativo para gerenciar listas. Logo, quando está todo preenchido, ao adicionar uma amostra apenas sobrescreve os elementos mais antigos, e para o método de ler o *buffer* ajusta o ponto de início.

Dois comentários são feitos a partir da Figura 33, representando a classe discutida. O tipo guardado na lista usa 32 *bits*, independente do dado guardado, o que é um ponto a ser otimizado. Além disso, na linha 14, o argumento *axis* precisou ser especificado, porque cada amostra de ACC é uma lista com 3 elementos, e o valor padrão do argumento embaralha os termos.

O tamanho do *buffer* colocado nos gráficos é de 600 amostras, mas pode ser ajustado no início do arquivo *app.py*, junto a outros valores globais.

```

3  class CircularBuffer:
4      def __init__(self, size=(100,)):
5          self.size = size[0]
6          self.buffer = np.zeros(size, dtype=np.int32)
7          self.index = 0
8
9      def add_data(self, data):
10         self.buffer[self.index] = data
11         self.index = (self.index + 1) % self.size
12
13     def get_data(self):
14         return np.roll(self.buffer, -self.index, axis=0)

```

Figura 33 - Implementação do buffer circular

Da Figura 34 tem-se a programação do gráfico de eletrocardiograma. O componente em si herda propriedades de um *widget* genérico, mas dentro dele usa um gráfico pronto do *framework*. Implementa-se o *buffer*, e de resto as configurações visuais e comportamentais. Dentre elas: mostrar a grade, o título do gráfico, título dos eixos, escala e interação com o usuário. Um dos atributos da classe é a curva do gráfico, que é modificada com o método de atualizar os dados. Todas as amostras são adicionadas ao *buffer*, e o mesmo é lido para sobrescrever a curva.

O gráfico para o acelerômetro segue essa ideia e está definido no mesmo módulo, *custom_charts.py* dentro da pasta *widgets*. As diferenças são a escolha de não mostrar a grade, a presença da legenda e o uso de 3 curvas ao simultaneamente, mas com conceitos idênticos. Nota-se também que o alcance do eixo X depende do tamanho do *buffer*, e para o eixo Y o limite é o valor máximo possível lido da Polar H10. Quando se trabalha dentro desses limites, a visualização é ajustada automaticamente.


```

6  class ECGChartWidget(QWidget):
7      def __init__(self, parent=None, buffer_size=100):
8          super().__init__(parent)
9          self.buffer = CircularBuffer((buffer_size,))
10         self.plot = pg.PlotWidget()
11         self.plot.setBackground('w')
12         self.plot.showGrid(x=True, y=True)
13         self.plot.setLabel('left', 'uV')
14         self.plot.setLabel('bottom', 'Amostras')
15         self.plot.setYRange(-20000, 20000)
16         self.plot.setXRange(0, buffer_size)
17         self.pen = pg.mkPen(color='r', width=1)
18         self.curve = self.plot.plot(pen=self.pen)
19
20         self.plot.setMouseEnabled(x=False, y=False) # Desabilita interações do mouse e teclado
21         self.plot.enableAutoRange(pg.ViewBox.XYAxes) # Ajuste automático para ambos os eixos
22
23         layout = QVBoxLayout()
24         layout.addWidget(self.plot)
25         self.setLayout(layout)
26
27     def update_chart(self, data):
28         for el in data:
29             self.buffer.add_data(el)
30         curve_values = self.buffer.get_data()
31         self.curve.setData(curve_values)

```

Figura 34 - Implementação do gráfico para o eletrocardiograma

A aparência do projeto é a última parte a ser comentada. Inicialmente a diagramação era feita no *script* principal, mas por motivo de organização foi movida para a pasta *layouts*, presente na Figura 29. Em muitos casos se usa a aplicação voltada para facilitar o *design* do projeto, mas nesse caso o tudo foi feito direto no código, por se encontrar em um meio termo. Para avanços futuros, essa ferramenta deve ser considerada.

Dois esboços para a interface gráfica são sinalizados na Figura 35. Com tantas decisões visuais a serem feitas, buscou-se o mais natural e que aproveitasse o espaço da melhor forma. Assim, detalhes de ambos os modelos foram incorporados na versão final. Por exemplo, a divisão horizontal proporciona um maior espaço aproveitado para os gráficos, visto que o eixo X é mais valorizado do que o Y para ilustrar o histórico de dados e a sua distribuição com maior clareza.

Além disso, uma barra lateral à esquerda foi determinada para a visualização do restante dos dados e do controle. Pensando na intuição do usuário em processar as informações de cima para baixo e da esquerda para a direita, essa organização proporciona a valorização adequada para cada categoria. A posição do logo é na parte superior, os dados no meio e na parte inferior está o controle de comunicação.

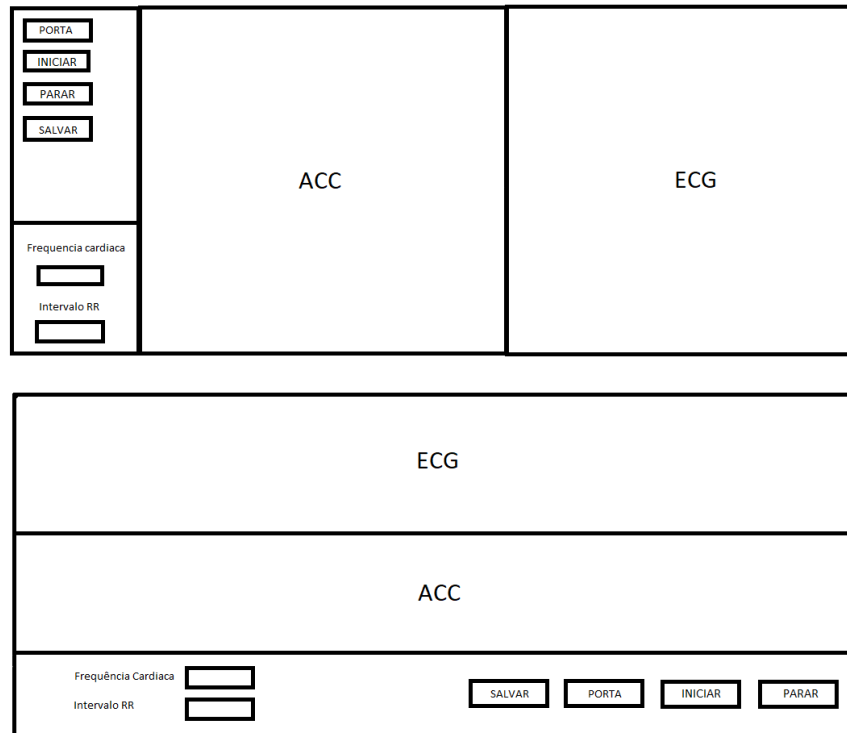


Figura 35 - Conceitos iniciais para a aparência da interface

Finalizando a seção de interface, vale mencionar sobre a fase de testes durante o seu desenvolvimento. Antes de se ter a comunicação serial pronta, o arquivo *serial_thread.py* teve dois funcionamentos configurados, um para teste e outro para a versão final. Durante os testes com o ESP32, alguns *logs* de mensagens da faixa peitoral foram salvos diretamente da serial da IDE do Arduino para arquivos de texto. Como fruto dessas sessões de aquisição, tem-se um arquivo com mais ou menos 23 minutos de dados de ECG e aproximadamente 7 minutos de coleta de ECG junto com ACC. Esses são ambos os arquivos da pasta *mocks* indicada anteriormente na Figura 29. Em cada linha existe uma mensagem, com a devida marca de tempo. A partir deles se poderia simular o comportamento do microcontrolador mesmo sem estar conectado.

No programa principal existe uma configuração global que define o uso ou não dos dados *mock*. Foi introduzido um botão para que essa funcionalidade seja melhor explorada no futuro. No entanto, vale ressaltar que, no status atual, o programa é capaz de ler sessões gravadas e salvas em arquivos de texto, o que é importante para revisar sessões de monitoramento a fim de se chegar a uma análise mais detalhada, detectar erros e identificar instabilidades, por exemplo.

4. RESULTADOS E DISCUSSÃO

A Figura 36 ilustra os resultados do teste de simulação no qual o ESP32 foi submetido à tarefa de decodificar mensagens contínuas de eletrocardiograma (ECG) e aceleração (ACC) provenientes do dispositivo Polar H10. Os resultados fornecem uma visão clara da eficácia da aquisição do ESP32 durante esse processo, evidenciando que as funções desenvolvidas para o teste ao longo do projeto desempenharam seu papel de forma precisa e planejada.

O êxito na decodificação das mensagens de dados contínuos é crucial para assegurar a integridade e precisão das informações provenientes da Faixa Peitoral Polar H10. Os resultados da simulação indicam que o ESP32 foi capaz de interpretar e processar corretamente os dados transmitidos, confirmando que a comunicação entre o microcontrolador e o dispositivo de monitoramento ocorreu conforme o planejado.

A capacidade do ESP32 em decodificar as mensagens de ECG e aceleração corretamente valida a eficiência das funcionalidades implementadas ao longo do desenvolvimento do projeto. Além disso, essa validação reforça a robustez do protocolo de comunicação estabelecido entre o ESP32 e a faixa peitoral Polar H10. Os resultados obtidos nesse teste de simulação contribuem positivamente para a confiança no funcionamento do sistema, fornecendo uma base sólida para a continuidade do projeto.

```

19:58:18.005 -> FC: 69, IRRs: 816, 847, -> Frequência cardíaca e intervalos R-R
19:58:18.005 -> ECG: -73, -64, -57, -53, -40, -42, -115, -156, 115, 753,
19:58:18.005 -> 1295, 1045, -167, -981, -503, -115, -321, -243, -71, -149,
19:58:18.005 -> -171, -84, -69, -69, -31, -20, -28, -13, 8, 24,
19:58:18.053 -> 49, 64, 60, 69, 100, 131, 160, 189, 187, 182,
19:58:18.053 -> 196, 187, 176, 200, 211, 193, 187, 0, 147, 78,
19:58:18.053 -> -11, -102, -171, -211, -225, -225, -225, -225, -216, -196,
19:58:18.053 -> -167, -147, -136, -127, -120, -115, -100, -84, -82, -84,
19:58:18.053 -> -107, -138, -138,
19:58:18.053 -> ACC: -969, 120, -102, -967, 126, -87, -972, 122, -105, -963,
19:58:18.053 -> 120, -99, -965, 123, -99, -966, 121, -98, -967, 120,
19:58:18.053 -> -92, -971, 123, -94, -968, 127, -96, -968, 123, -96,
19:58:18.053 -> -967, 120, -95, -968, 119, -95, -968, 120, -95, -967,
19:58:18.053 -> 122, -98, -968, 123, -100, -969, 125, -100, -968, 125,
19:58:18.053 -> -102, -967, 121, -100, -966, 120, -99, -965, 122, -100,
19:58:18.101 -> -967, 123, -100, -969, 121, -98, -967, 121, -96, -967,
19:58:18.101 -> 125, -98, -965, 125, -98, -969, 126, -98, -972, 125,
19:58:18.101 -> -93, -967, 124, -92, -957, 123, -95, -965, 129, -105,
19:58:18.101 -> -978, 123, -106, -968, 113, -88, -968, 111, -78, -973,
19:58:18.101 -> 126, -105, -967, 127, -107, -971, 123, -92,

```

→ ECG

→ ACC

Figura 36 – Simulação da aquisição e decodificação do Polar H10 com ESP32.

A Figura 37 apresenta os resultados da simulação dos testes utilizando a Polar H10 que mostra os sinais relacionados ao eletrocardiograma, intervalo R-R, além da bateria e frequência cardíaca. Já a Figura 38 apresenta os mesmos resultados só que com os valores reais. Portanto, resultados utilizando um indivíduo para avaliação dos sinais. A partir disso, pode-se dizer que ao analisar os resultados referentes ao teste real, o sensor apresentou ruído nas medições de ECG. Esse comportamento pode ser justificado devido ao desgaste na faixa peitoral, visto que a faixa já havia sido utilizada para testes em outras ocasiões. Assim, sugere-se que para novos estudos voltados para a avaliação do comportamento inicial dos valores do Polar H10, utilize um *wearable* dedicado exclusivamente para a pesquisa.

Também pode ser verificado que o *layout* final do projeto utilizou métodos definidos anteriormente no item 3.c.i. Dessa forma, o posicionamento das informações referente ao ECG e acelerômetro foram plotados de forma horizontal, proporcionando maior aproveitamento do espaço de amostra e facilitando a visualização detalhada da informação. Além disso, o intervalo R-R, nível de bateria e frequência cardíaca foram colocados centralizadas na aba esquerda, por serem dados importantes a serem monitorados do indivíduo em atividade sem que atrapalhe a visualização das demais informações.

A disposição da seção da interface relacionada à comunicação serial foi deliberadamente colocada no canto inferior esquerdo, uma vez que sua relevância diminui após a conclusão da conexão. Em termos simples, embora essa informação seja crucial durante a inicialização, foi decidido posicioná-la nessa área, levando em consideração outros dados essenciais.

Além disso, o uso efetivo do programa requer uma abordagem específica para garantir uma comunicação eficiente com o ESP32. Recomenda-se conectar o microcontrolador antes de iniciar o programa, uma vez que a verificação da disponibilidade das portas COM ocorre apenas uma vez durante a inicialização. A seleção da porta e a velocidade da conexão são essenciais e podem ser configuradas através do *dropdown* (seletor) localizado no canto inferior esquerdo da interface. As opções de taxa de símbolos, provenientes da IDE do Arduino, foram incorporadas, sendo a taxa de 115200 escolhida no programa. Essa seleção, além de ser comum em aplicações do ESP32, foi adotada para suportar múltiplos dispositivos de aquisição simultaneamente, sem atrasos ou perda de conteúdo.

Para usar o programa, uma vez feitas as configurações, o usuário pode iniciar a comunicação com o ESP32 clicando no botão "Iniciar". Por sua vez, a faixa Polar H10 pode ser vestida a qualquer momento, e o microcontrolador realiza automaticamente a conexão quando detecta o dispositivo. No entanto, é necessário pré-definir o ID do sensor no código. Com todos os componentes inicializados, os dados começam a ser exibidos na interface. Assim, os gráficos exibem um pacote de amostras por vez, preenchendo as 600 posições do buffer e adotando um comportamento de "janela deslizante". No caso do acelerômetro, as cores dos gráficos são uniformes, mantendo consistência com a Figura 14, enquanto a legenda esclarece claramente os eixos.

À esquerda da tela, para os dados fora dos gráficos, os dados mais recentes permanecem visíveis até que novas mensagens cheguem, atualizando-os. Para os intervalos R-R, que podem estar ausentes em um pacote, essa situação sobrescreve as informações, resultando em campos vazios.

O botão "Parar" permite interromper a conexão, refletindo o status por meio do texto e da cor do círculo na interface. O programa lida adequadamente com a desconexão do cabo USB, evitando erros e encerrando a conexão. No entanto, é importante observar que a mudança para uma nova porta serial durante a execução do programa não é viável, pois a detecção de portas disponíveis ocorre apenas durante a inicialização.

Os recursos gráficos são exibidos conforme o esperado, e o framework permite o ajuste responsivo do tamanho da janela, proporcionando uma experiência de usuário consistente. Esses resultados destacam a eficácia do programa na integração e apresentação de dados, bem como na gestão adequada da comunicação e interação com os dispositivos conectados.

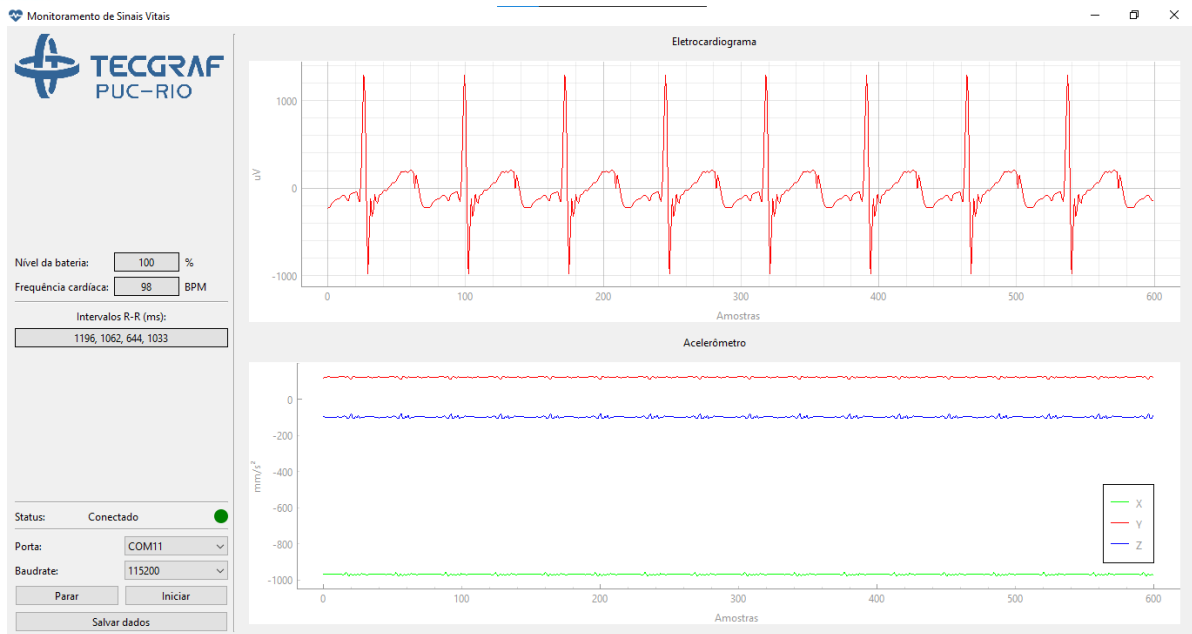


Figura 37 - Interface usada com o microcontrolador simulando a Polar H10



Figura 38 - Interface gráfica funcionando com dados reais

Abo-Zahhad *et al.* (2014) desenvolveram um sistema para monitorar os dados de um ECG Holter com 5 eletrodos e enviar alertas online através de dados telefônicos ou redes de internet sem fio. Isso possibilita que médicos e familiares sejam imediatamente contatados, permitindo ações cruciais serem tomadas a tempo. Essa abordagem é particularmente benéfica para a população idosa, que, devido ao aumento nos últimos anos, requer monitoramento contínuo.

Em um estudo relacionado, foi explorado o uso de um sistema de Interface Homem-Máquina (IHM) para monitorar biossinais em um ambiente de trabalho offshore. Diante do contexto da Covid-19, a necessidade de monitoramento remoto tornou-se imperativa. Assim, um eletrocardiograma padrão ouro foi adaptado, tornando-se portátil ao ser incorporado em uma maleta, juntamente com um notebook para a leitura dos dados. As leituras são transmitidas sem fio ao computador dos médicos envolvidos no

monitoramento, proporcionando acesso facilitado e diagnóstico preciso à distância, dada a confiabilidade dos dados (Barbosa *et al.*, 2021).

O objetivo de outro estudo foi desenvolver um sistema de alerta com IHM para monitorar a condição de saúde durante mergulhos offshore. Em ambientes extremos, a segurança é crucial para prevenir acidentes, envolvendo a interpretação dos sinais biológicos do mergulhador e a interrupção da atividade, se necessário, antes que complicações ocorram. Os sinais monitorados incluíram frequência cardíaca, intervalo R-R, temperatura, acelerômetro e giroscópio. Embora métodos biológicos tenham sido empregados para verificar se os dados cardíacos estavam dentro dos limites normais, os diferentes metabolismos dos profissionais representaram um desafio. Para superar isso, dados adicionais do usuário foram incorporados, e otimizações no sistema foram implementadas para garantir maior precisão nas análises (Almeida *et al.*, 2023).

Dessa forma, esses estudos mostram como é relevante o estudo envolvendo a utilização de interface HM para o monitoramento de biossinais. Dois desses trabalhos são voltados para aplicações extremas (alterações offshore) e o outro de uso clínico cotidiano, onde os pesquisadores utilizaram interfaces semelhantes ao desenvolvido nesse projeto. Em relação ao que foi desenvolvido, alguns sinais diferentes foram escolhidos para a mostra de formas diferentes, como a aceleração em um gráfico, o que fica mais claro de visualizar. O ECG foi um dado não presente em todos os estudos, e a bateria é um dado que é importante para ser implementado em uma interface, mas não se é sinalizado. Por fim, para futuras implementações seria ideal configurar um sistema de alertas.

A Figura 39 apresenta o diagrama completo do sistema implementando, onde possui o wearable, o microcontrolador ESP32 e o computador. Como foi visto ao longo do projeto, o fluxo de informação é unidirecional, do wearable até a interface, passando pelo BLE e por cabo USB. Buscou-se com essas escolhas o melhor comportamento possível dado o escopo do projeto, e os resultados correram como o esperado.

A faixa peitoral Polar H10 demonstrou ser altamente apta à função proposta, respaldada por diversos estudos de validação que atestam a confiabilidade e comparabilidade de seus dados com os medidores Holter convencionais utilizados em ambientes hospitalares. Em comparação com esses dispositivos, o sistema adquire o eletrocardiograma de maneira não invasiva, eliminando interferências causadas pelos movimentos dos fios e proporcionando um nível de conforto significativo para o usuário. A ausência da necessidade de depilação e aplicação de gel condutor destaca-se como uma vantagem adicional. A abertura do código fonte, associada à boa autonomia de bateria e à facilidade de vestir e retirar a faixa, reforçam a escolha desse sensor como extremamente favorável para o desenvolvimento do projeto. As taxas de envio de dados foram consideradas adequadas para o monitoramento contínuo, inclusive durante atividades físicas, evidenciando a robustez do sistema.

A placa ESP32 revelou-se uma escolha acertada, caracterizada por sua acessibilidade em termos de preço e disponibilidade no mercado. Sua capacidade de processamento mostrou-se suficiente para suportar simultaneamente mais de um dispositivo Polar H10, quando necessário. A total compatibilidade com a versão BLE do wearable contribuiu para uma integração eficiente. O código desenvolvido para a placa demonstrou comportamento conforme o esperado, gerenciando as mensagens de maneira rápida e eficiente. O protocolo estabelecido, utilizando estruturas simples para categorizar os tipos de mensagem, facilita a incorporação de novos tipos, adicionando uma camada adicional de versatilidade. Vale ressaltar que, embora a placa possua conectividade Wi-Fi, a opção de transmissão via USB foi selecionada, uma vez que o uso simultâneo com o BLE necessitaria de uma consideração cuidadosa devido ao compartilhamento da mesma antena.

O desenvolvimento da interface gráfica enfrentou desafios iniciais, particularmente em relação ao desempenho dos gráficos. No entanto, com otimizações progressivas, a versão final apresentou um comportamento superior às expectativas. A implementação do buffer circular e a separação de threads para a conexão serial e a parte gráfica resultaram em melhorias significativas, abrindo oportunidades para futuras expansões. O código modularizado facilitou a incorporação de novas medidas, mantendo-se alinhado aos objetivos iniciais do projeto. A usabilidade da interface final é destacada por sua simplicidade

e intuição, atingindo a meta estabelecida. A disposição clara das informações na tela, sinalizando todas as unidades, contribui para a facilidade de uso e acessibilidade. A execução eficiente em outras plataformas, como Linux e MacOS, amplia ainda mais a acessibilidade do sistema

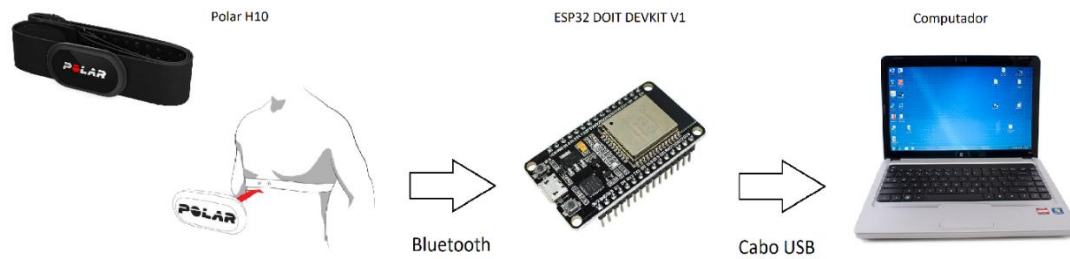


Figura 39 - Estrutura final do projeto (Polar Eletro Oy, c2023a) (Placa [s.d.]

5. CONCLUSÃO

Concluindo, a Interface HM revelou-se fundamental para a avaliação da saúde do usuário do dispositivo *wearable*, permitindo conclusões precisas mesmo em situações de distanciamento entre o usuário e o avaliador. O fluxo unidirecional de informação, utilizando os protocolos BLE e USB, demonstrou eficácia na transmissão de dados do *wearable* para a interface.

Os resultados obtidos destacaram a eficiência da faixa peitoral Polar H10, amplamente validada em estudos, proporcionando dados confiáveis comparáveis aos medidores Holter hospitalares. A escolha da placa ESP32 revelou-se vantajosa em termos de acessibilidade financeira, disponibilidade no mercado e capacidade de processamento, permitindo a gestão simultânea de múltiplos dispositivos Polar H10, se necessário.

O código desenvolvido, modular e eficiente, gerenciou as mensagens de maneira rápida, com total compatibilidade com o BLE do *wearable*. A implementação do protocolo mostrou-se flexível, facilitando a adição de novos tipos de mensagem.

Apesar dos desafios iniciais, a interface gráfica superou as expectativas finais com otimizações, como o uso de buffer circular e threads separadas para conexão serial e gráficos. O desenvolvimento modular do código facilitou a incorporação de novas medidas, alinhado com os objetivos estabelecidos.

A usabilidade da interface final foi alcançada com sucesso, sendo simples e intuitiva, atendendo a usuários de diferentes níveis. Cada componente contribuiu para um sistema final que não apenas atendeu às expectativas, mas também abriu portas para futuras oportunidades, evidenciando seu potencial contínuo.

Dessa forma, sugere-se para trabalhos futuros:

- Melhorias na interface, como por exemplo, incluir referência temporal das amostras;
- Possibilidade de salvar sessões de monitoramento;
- Acrescentar log de comunicação com o ESP32;
- Mandar comando pela Polar H10 pela interface;
- Avaliar outros dispositivos *wearables*;
- Monitorar diferentes indivíduos simultaneamente em uma mesma interface.

REFERÊNCIAS BIBLIOGRÁFICAS

- ABO-ZAHHAD, M.; AHMED, S. M.; ELNAHAS, O. A Wireless Emergency Telemedicine System for Patients Monitoring and Diagnosis. **International Journal of Telemedicine and Applications; Hindawi Limited**, 2014. Disponível em: <<http://dx.doi.org/10.1155/2014/380787>>. Acesso em: 1 dez. de 2023.
- AIRES et al. Indústria 4.0: desafios e tendências para a gestão do conhecimento. **Anais do I SUCEG**, Florianópolis, SC, Brasil, 8 dez. De 2017. Disponível em: <<https://anais.suceg.ufsc.br/index.php/suceg/article/view/49/17>>. Acesso em: 7 nov. de 2023.
- ALLISON, C. Best smart glasses and AR specs 2023: Tested picks from Snap, Meta and Amazon. **Wearable**, 2023. Disponível em: <<https://www.wearable.com/ar/the-best-smartglasses-google-glass-and-the-rest>>. Acesso em: 13 nov. de 2023.
- ALMEIDA, D. F. S. et al. In-depth Analysis and Creation of Vital Alerts System for Offshore Submerged Work 2023 9th International Conference on Control, Decision and Information Technologies (CoDIT). Em: **2023 9TH INTERNATIONAL CONFERENCE ON CONTROL, DECISION AND INFORMATION TECHNOLOGIES (CODIT)**. IEEE, 3 jul. 2023. Disponível em: <<http://dx.doi.org/10.1109/CoDIT58514.2023.10284497>>. Acesso em: 1 dez. de 2023.
- ANGELOV, V. et al. Modern Virtual Reality Headsets. **2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)**. Em: 2020 INTERNATIONAL CONGRESS ON HUMAN-COMPUTER INTERACTION, OPTIMIZATION AND ROBOTIC APPLICATIONS (HORA). IEEE, jun. 2020. Disponível em: <<http://dx.doi.org/10.1109/HORA49412.2020.9152604>>. Acesso em: 1 dez. de 2023.
- ASH, G. I. et al. Establishing a Global Standard for Wearable Devices in Sport and Exercise Medicine: Perspectives from Academic and Industry Stakeholders. **Sports Medicine Springer Science and Business Media LLC**, 1 set. 2021. Disponível em: <<http://dx.doi.org/10.1007/s40279-021-01543-5>>. Acesso em: 8 nov. de 2023.
- ATHAVALA, Y.; KRISHNAN, S. Biosignal monitoring using wearables: Observations and opportunities. **Biomedical Signal Processing and Control Elsevier BV**, set. 2017. Disponível em: <<http://dx.doi.org/10.1016/j.bspc.2017.03.011>>. Acesso em: 5 nov. de 2023.
- BARBOSA, W. S. et al. Development and Validation of a Biosignal Measurement Device to Assist in Teleoperated Diagnosis 2021 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE). **2021 IEEE ASIA-PACIFIC CONFERENCE ON COMPUTER SCIENCE AND DATA ENGINEERING (CSDE)**. IEEE, 8 dez. 2021. Disponível em: <<http://dx.doi.org/10.1109/CSDE53843.2021.9718442>>. Acesso em: 1 dez. de 2023.
- BARBOSA, W. S. et al. Industry 4.0: Construction of a Waterproof Wearable Biosignal Device to Assist in Submerged Offshore Work. 2022 8th **International Conference on Control, Decision and Information Technologies CoDIT'22**. Istanbul, Turquia, 17-20 maio de 2022.
- BECKER, D. E. Fundamentals of Electrocardiography Interpretation. **Anesthesia Progress, American Dental Society of Anesthesiology (ADSA)**, jun. de 2006. Disponível em: <[http://dx.doi.org/10.2344/0003-3006\(2006\)53\[53:FOEI\]2.0.CO;2](http://dx.doi.org/10.2344/0003-3006(2006)53[53:FOEI]2.0.CO;2)>. Acesso em: 10 nov. de 2023.
- BLUETOOTH. Origin of the Bluetooth Name. **Bluetooth®**, 2023. Disponível em: <<https://www.bluetooth.com/about-us/bluetooth-origin/>>. Acesso em: 7 nov. de 2023.
- BLUETOOTH (a). The Bluetooth® Low Energy Primer. **Bluetooth®, [s.d.]**. Disponível em: <<https://www.bluetooth.com/bluetooth-resources/the-bluetooth-low-energy-primer/>>. Acesso em: 1 dez. de 2023.

BLUETOOTH (b). Specifications and Documents. **Bluetooth®**, [s.d.]. Disponível em: <<https://www.bluetooth.com/specifications/specs/?types=specs-docs&keyword=core+specification&filter=>>>. Acesso em: 26 nov. de 2023.

BLUETOOTH (c). Heart Rate Service 1.0. **Bluetooth®**, [s.d.]. Disponível em: <<https://www.bluetooth.com/specifications/specs/heart-rate-profile-1-0/>>. Acesso em: 26 nov. de 2023.

BLUETOOTH (d). Heart Rate Profile 1.0. **Bluetooth®**, [s.d.]. Disponível em: <<https://www.bluetooth.com/specifications/specs/heart-rate-service-1-0/>>. Acesso em: 26 nov. de 2023.

BLUETOOTH (e). Battery Service 1.1. **Bluetooth®**, [s.d.]. Disponível em: <<https://www.bluetooth.com/specifications/specs/battery-service/>>. Acesso em: 26 nov. de 2023.

BOUKNIGHT, J. *et al.* Connection-By-Name: User-Oriented Protocol. **University of Illinois**, 28 out. de 1970. Disponível em: <<https://www.rfc-editor.org/rfc/rfc76.txt>>. Acesso em: 24 nov. de 2023.

BROLL, S. *et al.* Interpreting blood GLUcose data with R package iglu (L. Pyle, Ed.) **PLOS ONE Public Library of Science (PLOS)**, 1 abr. 2021. Disponível em: <<http://dx.doi.org/10.1371/journal.pone.0248560>>. Acesso em: 7 nov. de 2023.

CALVERT, T. W. *et al.* A Systems Model of the Effects of Training on Physical Performance. **IEEE Transactions on Systems, Man, and Cybernetics: Institute of Electrical and Electronics Engineers (IEEE)**, fev. de 1976. Disponível em: <<http://dx.doi.org/10.1109/TSMC.1976.5409179>>. Acesso em: 20 nov. de 2023.

CHHABRA, L. *et al.* Mouse Heart Rate in a Human: Diagnostic Mystery of an Extreme Tachyarrhythmia. **Indian Pacing and Electrophysiology Journal, Elsevier BV**, jan. 2012. Disponível em: <[http://dx.doi.org/10.1016/s0972-6292\(16\)30463-6](http://dx.doi.org/10.1016/s0972-6292(16)30463-6)>. Acesso em: 16 nov. de 2023.

CHAETOGNATHAN, P. Which Python GUI library should you use?: Comparing the Python GUI libraries available in 2023. **Python GUIs**, atualizado em 13 set. de 2023. Disponível em: <<https://www.pythonguis.com/faq/which-python-gui-library/>>. Acesso em: 30 nov. de 2023.

CHIANG, P.-H.; WONG, M.; DEY, S. Using Wearables and Machine Learning to Enable Personalized Lifestyle Recommendations to Improve Blood Pressure. **IEEE Journal of Translational Engineering in Health and Medicine Institute of Electrical and Electronics Engineers (IEEE)**, 2021. Disponível em: <<http://dx.doi.org/10.1109/JTEHM.2021.3098173>>. Acesso em: 6 nov. de 2023.

CHO, S. *et al.* Factors Affecting the Quality of Person-Generated Wearable Device Data and Associated Challenges: Rapid Systematic Review. **JMIR mHealth and uHealth JMIR Publications Inc.**, 19 mar. 2021. Disponível em: <<http://dx.doi.org/10.2196/20738>>. Acesso em: 19 nov. de 2023.

CILLIERS, L. Wearable devices in healthcare: Privacy and information security issues. **Health Information Management Journal ISAGE Publications**, 30 maio 2019. Disponível em: <<http://dx.doi.org/10.1177/1833358319851684>>. Acesso em: 12 nov. de 2023.

CUTECIRCUIT. Hugshirt. **CuteCircuit**, c2023. Disponível em: <<https://cutecircuit.com/hugshirt/>>. Acesso em: 25 nov. De 2023.

DAVENPORT, T. H.; PRUSAK, L. Conhecimento empresarial: como as organizações gerenciam o seu capital intelectual. Rio de Janeiro: **Campus**, 1998.

DIGITAL Health Software Precertification (Pre-Cert) Pilot Program. **fda.gov**, 2022. Disponível em: <<https://www.fda.gov/medical-devices/digital-health-center-excellence/digital-health-software-precertification-pre-cert-pilot-program>>. Acesso em: 10, de nov. de 2023.

DINH-LE, C.; CHUANG, R.; CHOKSHI, S. & MANN, D. Wearable Health Technology and Electronic Health Record Integration: Scoping Review and Future Directions. **JMIR mHealth and uHealth**, 2019. Vol. 7, Issue 9, p. e12861. JMIR Publications Inc.

EDDY, L. New Resource Enables Use of Data from Wearables in Clinical Studies. **John Hopkins Medicine: Insight**, 2021. Disponível em: <<https://www.hopkinsmedicine.org/news/articles/2021/08/new-resource-enables-use-of-data-from-wearables-in-clinical-studies>> Acesso em: 14 nov. de 2023.

ESPRESSIF. Espressif Announces the Launch of ESP32 Cloud on Chip and Funding by Fosun Group. **Espressif Systems Co.**, Shanghai, China, 07 set. de 2016. Disponível em: <https://www.espressif.com/en/media_overview/news/20160907-esp32briefing#:~:text=Sep%207%2C%202016-,Espressif%20announces%20the%20launch%20of%20ESP32%20Cloud%20on%20Chip%20and,MCU%20at%20Shanghai%20Parkyard%20Hotel>. Acesso em: 28 nov. de 2023.

ESPRESSIF (a). ESP32 series: datasheet. **Espressif Systems Co.**, versão 4.3, Shangai, c2023. Disponível em: <https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf>. Acesso em: 28 nov. de 2023.

ESPRESSIF (b). Official IoT Development Framework. **Espressif Systems Co.**, c2023. Disponível em: <<https://www.espressif.com/en/products/sdks/esp-idf>>. Acesso em: 28 nov. de 2023.

EXACTITUDE CONSULTANCY. Mercado de vestíveis militares por tecnologia (Comunicação e Computação, Conectividade, Navegação, Visão e Vigilância, Exoesqueleto, Monitoramento, Gestão de Energia e Têxteis Inteligentes), Usuário Final (Forças Terrestres, Forças Aerotransportadas e Forças Navais) e Tipo Wearable (Chapéus, Óculos, pulseiras, bodywear e aparelhos auditivos), região (América do Norte, Europa, Ásia-Pacífico, América do Sul, Oriente Médio e África), Tendências globais e previsões de 2022 a 2029. **Exactitude Consultancy**, 2022. Disponível em: <<https://exactitudeconsultancy.com/pt/reports/5178/mercado-de-vestiveis-militares/#table-of-content>>. Acesso em: 15 nov. 2023.

FARIVAR, S.; ABOUZAHA, M.; GHASEMAGHAEI, M. Wearable device adoption among older adults: a mixed-methods study. **International Journal of Information Management, Elsevier BV**, dez. de 2020. Disponível em: <<http://dx.doi.org/10.1016/j.ijinfomgt.2020.102209>>. Acesso em: 11 nov. de 2023.

FELDMAN, J.; GOLDWASSER, G. P. Eletrocardiograma: recomendações para a sua interpretação. **Revista de SOCERJ**, v17., n. 04, art. 03, 2004. Disponível em: <http://sociedades.cardiol.br/socerj/revista/2004_04/a2004_v17_n04_art03.pdf>. Acesso em: 22 nov. De 2023.

FITZPATRICK, M. Create GUI Applications with Python & Qt6 (PyQt6 Edition): The hands-on guide to making apps with Python. **Kindle**, 5a Ed. 2022.

FONTICONS INC. Fontawesome, c2023. **Biblioteca online de ícones**. Disponível em: <<https://fontawesome.com/icons/heart-pulse?f=classic&s=solid&pc=%23326496>>. Acesso em: 11 de nov. de 2023.

GILGEN-AMMANN, R.; SCHWEIZER, T.; WYSS, T. RR interval signal quality of a heart rate monitor and an ECG Holter at rest and during exercise. **European Journal of Applied Physiology, Springer Science and Business Media LLC**, 19 abr. 2019. Disponível em: <<http://dx.doi.org/10.1007/s00421-019-04142-5>>. Acesso em: 1 dez. de 2023.

GUYTON, A.C.; HALL, J.E., Tratado de Fisiologia Médica. 11ª. Ed. Rio de Janeiro: **Elsevier Brasil**, 2006. 1216 p. Disponível em: <<https://cssjd.org.br/imagens/editor/files/2019/Abril/Tratado%20de%20Fisiologia%20M%C3%A9dica.pdf>> Acesso em: 15 nov. de 2023.

GRANA POSSAMAI, C. *et al.* Use of wearable biometric monitoring devices to measure outcomes in randomized clinical trials: a methodological systematic review. **BMC Medicine Springer Science and Business Media LLC**, 6 nov. 2020. Disponível em: <<http://dx.doi.org/10.1186/s12916-020-01773-w>>. Acesso em: 11 nov. de 2023.

GRIGG, E. K. *et al.* Comparative Assessment of Heart Rate Variability Obtained via Ambulatory ECG and Polar Heart Rate Monitors in Healthy Cats: A Pilot Study. **Frontiers in Veterinary Science Frontiers in Veterinary Science, Frontiers Media SA**, 8 nov. 2021. Disponível em: <<http://dx.doi.org/10.3389/fvets.2021.741583>>. Acesso em: 20 nov. de 2023.

GU, Z.; WEI, J. Wearable Services Adoption Study from a Perspective of Usability. **Springer Nature**, 2021. Disponível em: <https://www.researchgate.net/publication/342822524_Wearable_Services_Adoption_Study_from_a_Perspective_of_Usability>. Acesso em: 14 nov. 2023.

GAŠIOR, J. S. *et al.* Validity of the Polar H10 for RR intervals acquisition during different conditions for heart rate variability analysis in ski mountaineers. **Research Square Platform LLC**, 19 out. 2022. Disponível em: <<http://dx.doi.org/10.21203/rs.3.rs-2167605/v1>>. Acesso em: 18 nov. de 2023.

HAUSER, E. UNIX time, UTC, and datetime: Jussivity, prolepsis, and incorrigibility in modern timekeeping Proceedings of the Association for Information Science and Technology. **Wiley**, jan. 2018. Disponível em: <<http://dx.doi.org/10.1002/pra2.2018.14505501018>>. Acesso em: 11 nov. de 2023.

HOSPITAL ISRAELITA ALBERT EINSTEIN. Saiba o que é eletrocardiograma e como se preparar para fazê-lo. **Vida Saudável**, 2 jun. de 2021. Disponível em: <<https://vidasaudavel.einstein.br/eletrocardiograma/#:~:text=O%20seu%20funcionamento%20ocorre%20a,para%20a%20tela%20ou%20papel.>>>. Acesso em: 22 nov. de 2023.

IQBAL, S. M. A. Advances in healthcare wearable devices. **Npj flexible electronics**, 12 abr. de 2021. Disponível em: <<https://www.nature.com/articles/s41528-021-00107-x>>. Acesso em: 20 nov. 2023.

KALIN, R. A Simplified NCP Protocol. **MIT**, 13 jul. de 1970. Disponível em: <<https://www.rfc-editor.org/rfc/rfc60.txt>>. Acesso em: 24 nov. de 2023.

KINNEY, M. Wearable Activity Trackers Can Be Used to Determine Health Metrics That Could Support Clinical Care: Metrics also enabled researchers to group participants based on clinical parameters. **John Hopkins Institute whiting school of engineering**, 2022. Disponível em: <<https://engineering.jhu.edu/materials/news/wearable-activity-trackers-can-be-used-to-determine-health-metrics-that-could-support-clinical-care/#:~:text=,health%20and%20cardiovascular%20health%20status>> Acesso em: 15 nov. de 2023.

KOLBAN, N. BLE C++ Guide. **GitHub**, 1 jan. de 2018. Disponível em: <<https://github.com/nkolban/esp32-snippets/blob/master/Documentation/BLE%20C%2B%2B%20Guide.pdf>> . Acesso em: 1 nov. de 2023.

KOLBAN, N. ESP32_BLE_Arduino. **GitHub**, c2023. Repositório ESP32 Arduino. Disponível em: <https://github.com/nkolban/ESP32_BLE_Arduino>. Acesso em: 24 nov. de 2023.

KUBOTA, T. Wearable technology research at Stanford. **Stanford University Communications**, 18 ago. de 2021. Disponível em: <<https://news.stanford.edu/2021/08/18/wearable-technology-research-stanford/>>. Acesso em: 16 nov. de 2023.

LIN, Q. *et al.* Wearable Multiple Modality Bio-Signal Recording and Processing on Chip: A Review. **IEEE Sensors Journal; Institute of Electrical and Electronics Engineers (IEEE)**, 15 jan. 2021. Disponível em: <<http://dx.doi.org/10.1109/JSEN.2020.3016115>>. Acesso em: 29 dez. de 2023.

LOWN, M. *et al.* Screening for Atrial Fibrillation Using Economical and Accurate Technology (From the SAFETY Study). **The American Journal of Cardiology, Elsevier BV**, out. 2018. Disponível em: <<http://dx.doi.org/10.1016/j.amjcard.2018.07.003>>. Acesso em: 12 nov. de 2023.

MARCEL, J. Bluetooth Wearables Are Driving the Future of Data Transfer Device Growth. **Bluetooth®**, 10 maio de 2021. Disponível em: <<https://www.bluetooth.com/blog/bluetooth-wearables-are-driving-the-future-of-data-transfer-device-growth/#:~:text=Thanks%20to%20a%20greater%20understanding%20of%20the%20personal,Bluetooth%20Data%20Transfer%20device%20shipments%20will%20be%20wearables.>>. Acesso em: 20 nov. de 2023.

MATHWORKS. **mathworks.com**, 2021. Bluetooth Protocol Stack. Disponível em: <<https://www.mathworks.com/help/bluetooth/ug/bluetooth-protocol-stack.html>>. Acesso em: 31 de nov. de 2023.

MEISTER, J. Educação corporativa. São Paulo: **Makron Books**, 1999.

MINISTÉRIO DA SAÚDE. Saúde do coração. **Biblioteca virtual em saúde**, 2004. Disponível em: <<https://bvsmis.saude.gov.br/saude-do-coracao/#:~:text=O%20cora%C3%A7%C3%A3o%20%C3%A9%20o%20m%C3%BAsculo,para%20mant%C3%AA%20o%20em%20forma>>. Acesso em: 22 nov. de 2023.

MORTON, R. H. Modelling training and overtraining. **Journal of Sports Sciences: Informa UK Limited**, jan. 1997. Disponível em: <<http://dx.doi.org/10.1080/026404197367344>>. Acesso em: 20 nov. de 2023.

MUALEM, W. *et al.* Utilizing Data from Wearable Technologies in the Era of Telemedicine to Assess Patient Function and Outcomes in Neurosurgery: Systematic Review and Time-Trend Analysis of the Literature. **World Neurosurgery Elsevier BV**, 2022. Disponível em: <<http://dx.doi.org/10.1016/j.wneu.2022.07.036>>. Acesso em: 15 nov. de 2023.

PLACA ESP32 WiFi / Bluetooth DEVKit V1 30 Pinos. **Saravati, [s.d.]**. Disponível em: <<https://www.saravati.com.br/placa-esp32-wifi-bluetooth-devkit-v1-30-pinos.html>>. Acesso em: 9 nov. de 2023.

POLAR ELECTRO OY. Polar H10 Heart Rate Sensor – the gold standard of heart rate measurement. **Polar Electro Oy**, 15 mar. de 2017. Atualizado 4 set. de 2020. Disponível em: <<https://www.polar.com/blog/new-polar-h10-heart-rate-sensor-2017/>>. Acesso em: 17 nov. de 2023.

POLAR ELECTRO OY (a). PMD Data Specification. **GitHub**, 2021. Repositório oficial. Disponível em: <https://github.com/polarofficial/polar-ble-sdk/tree/39c809ab4a3398d3ea88ff8a0cc6f6d44cd58bda/technical_documentation>. Acesso em: 30 jun. 2023.

POLAR ELECTRO OY (b). How to set polar OH1 time. **GitHub**, 15 jan. de 2021. Disponível em: <<https://github.com/polarofficial/polar-ble-sdk/issues/86>>. Acesso em: 24 nov. de 2023.

POLAR ELECTRO OY (c). Acceleration data with gravity eliminated. **GitHub**, 17 jun. de 2021. Disponível em: <<https://github.com/polarofficial/polar-ble-sdk/issues/167>>. Acesso em: 10 nov. de 2023.

POLAR ELECTRO OY (a). Polar Technical Documentation. **GitHub**, mar. de 2023. Disponível em: <https://github.com/polarofficial/polar-ble-sdk/tree/master/technical_documentation>. Acesso em: 22 nov. de 2023.

POLAR ELECTRO OY (b). SDK for polar sensors. **GitHub**, 2023. Disponível em: <<https://github.com/polarofficial/polar-ble-sdk>>. Acesso em: 15 nov. de 2023.

POLAR ELECTRO OY (c). Is it possible to store Accelerometer Data on H10?. **GitHub**, abr. de 2023. Disponível em: <<https://github.com/polarofficial/polar-ble-sdk/issues/357>>. Acesso em: 14 nov. de 2023.

POLAR ELECTRO OY (a). Polar H10: manual do usuário. **Polar Electro Oy**, c2023. Disponível em: <https://support.polar.com/e_manuals/h10-heart-rate-sensor/polar-h10-user-manual-portugues/manual.pdf>. Acesso em: 22 nov. de 2023.

POLAR ELECTRO OY (b). Abnormal Heart Rate Readings During Exercise. **Polar Electro Oy**, c2023. Disponível em:

<https://support.polar.com/en/support/FAQs/Abnormal_Heart_Rate_Readings_During_Exercise?product_id=89452&category=heart_rate_measurement>. Acesso em: 21 nov. de 2023.

POLAR ELECTRO OY (c). Who we are. **Polar Electro Oy**, c2023. Disponível em:

<https://www.polar.com/en/about_polar/who_we_are>. Acesso em: 19 nov. De 2023.

POLAR ELECTRO OY (d). Time system in Polar devices. **GitHub**, c2023. Disponível em:

<<https://github.com/polarofficial/polar-ble-sdk/blob/master/documentation/TimeSystemExplained.md>>. Acesso em 25 nov. de 2023.

POLAR ELECTRO OY (a). Where is the Polar Feature Configuration Service (PFC) documented? **GitHub**, 29 ago. de 2022. Repositório oficial. Disponível em: <<https://github.com/polarofficial/polar-ble-sdk/issues/296>>. Acesso em: 24 nov. de 2023.

POLAR ELECTRO OY (b). RR Interval Data Interpretation. **GitHub**, 23 jan. de 2022. Disponível em:

<<https://github.com/polarofficial/polar-ble-sdk/issues/228>>. Acesso em: 24 nov. de 2023.

POLAR ELECTRO OY (c). how to match ecg data package with HR Interval. **GitHub**, 23 jan. de 2022.

Disponível em: <<https://github.com/polarofficial/polar-ble-sdk/issues/227>>. Acesso em: 22 nov. de 2023.

POLAR ELECTRO OY (a). Accelerator stream mixed with ECG stream. **GitHub**, 15 dez. de 2020.

Disponível em: <<https://github.com/polarofficial/polar-ble-sdk/issues/109>>. Acesso em: 24 nov. de 2023.

POLAR ELECTRO OY (b). H10: is it possible to access and set the internal memory? **GitHub**, 19 out. de 2020.

Disponível em: <<https://github.com/polarofficial/polar-ble-sdk/issues/102>>. Acesso em: 14 nov. de 2023.

PUCRS Online. Wearables: um mercado em ascensão. **PUCRS Online**, 28 set. de 2021. Disponível em:

<<https://online.pucrs.br/blog/public/wearables-um-mercado-em-ascensao>>. Acesso em: 17 nov. de 2023.

PUNCH THROUGH. Maximizing BLE Throughput Part 2: Use Larger ATT MTU. **Punch Through**, 7 nov.

de 2017. Disponível em: <<https://punchthrough.com/maximizing-ble-throughput-part-2-use-larger-att-mtu-2/>>. Acesso em: 25 nov. de 2023.

QT. About Qt. **Wiki Qt**, atualizado em 18 jul. de 2022. Disponível em: <https://wiki.qt.io/About_Qt>.

Acesso em: 30 nov. de 2023.

RAINMAKER, DC. Polar H10 Heart Rate Monitor: Very Long Term In-Depth Review. **DC Rainmaker**, 17

dez. de 2021. Disponível em: <https://www.dcrainmaker.com/2021/12/polar-h10-heart-rate-monitor-very-long-term-in-depth-review.html>. Acesso em: 26 nov. de 2023.

RINGLY. RINGLY Go. **ringly.com**, c2023. Disponível em:

<<https://docs.google.com/document/d/1V3XjHXwPnQL0CaxZTNEgle9NMGuGmsSya7kn8hCGBQY/edit>>. Acesso em: 15 nov. de 2023.

ROOS, L. G.; SLAVICH, G. M. Wearable technologies for health research: Opportunities, limitations, and practical and conceptual considerations Brain, Behavior, and Immunity. **Elsevier BV**, out. 2023.

Disponível em: <<http://dx.doi.org/10.1016/j.bbi.2023.08.008>>. Acesso em: 12 nov. de 2023.

SANDHAM, J. Electrical signals of the heart. **EBME & Clinical Engineering Articles**, c1999-2023.

Disponível em: <<https://www.ebme.co.uk/articles/clinical-engineering/electrical-signals-of-the-heart>>. Acesso em: 22 nov. De 2023.

SCHAFFARCZYK, M. *et al.* Validity of the Polar H10 Sensor for Heart Rate Variability Analysis during Resting State and Incremental Exercise in Recreational Men and Women Sensors. **MDPI AG**, 30 ago. 2022. Disponível em: <<http://dx.doi.org/10.3390/s22176536>>. Acesso em: 3 nov. de 2022.

SEMMLOW, John. Circuits, Signals and Systems for Bioengineers: a matlab-based introduction. 3. ed. Amsterdã: **Elsevier**, 2018.

SESHADRI, D.R. *et al.* Wearable sensors for monitoring the internal and external workload of the athlete. **Digital Medicine**. 2, 71 (2019). <https://doi.org/10.1038/s41746-019-0149-2> Acesso em: 3 nov. de 2022.

SESHADRI, D. R. *et al.* Wearable Technology and Analytics as a Complementary Toolkit to Optimize Workload and to Reduce Injury BurdenFrontiers in Sports and Active Living. **Frontiers Media SA**, 21 jan. 2021. Disponível em: <<http://dx.doi.org/10.3389/fspor.2020.630576>>. Acesso em: 19 nov. de 2022.

SEÇKIN, A. Ç.; ATEŞ, B.; SEÇKIN, M. Review on Wearable Technology in Sports: Concepts, Challenges and Opportunities. **Applied Sciences MDPI AG**, 17 set. 2023. Disponível em: <<http://dx.doi.org/10.3390/app131810399>>. Acesso em: 6 nov. de 2023.

SILICON LABORATORIES. Using 2M and LE Coded PHY. **Silicon Labs**, c2023. Disponível em: <<https://docs.silabs.com/bluetooth/2.13/general/connections/using-2m-and-le-coded-phy-bt5>>. Acesso em: 29 nov. de 2023.

SILICON LABS. CP210x USB to UART Bridge VCP Drivers. **Silicon Labs**, c2023. Disponível em: <<https://www.silabs.com/developers/usb-to-uart-bridge-vcp-drivers>>. Acesso em: 7 nov. de 2023.

SILVA, J. Privacy Data Ethics of Wearable Digital Health Technology. **Digital Health: the Warren Alpert Medical School**, maio 2023. Disponível em: <<https://digitalhealth.med.brown.edu/news/2023-05-04/ethics-wearables>>. Acesso em: 14 nov. de 2023.

SILVERTHORN, D. U. **Fisiologia Humana: Uma Abordagem Integrada**. 7a Ed. Porto Alegre: Artmed, 2017. ISBN 978-8582714034.

SKÁLA, T. *et al.* Feasibility of evaluation of Polar H10 chest-belt ECG in patients with a broad range of heart conditions. **Cor et Vasa; Czech Society of Cardiology**, 1 set. 2022. Disponível em: <<http://dx.doi.org/10.33678/cor.2022.083>>. Acesso em: 31 out. de 2023.

SMITH, C. What is wearable tech? Everything you need to know explained, Wearable tech, fitness trackers and smartwatches explained. **Wearable**, 2017. Disponível em: <<https://www.wearable.com/wearable-tech/what-is-wearable-tech-753>>. Acesso em: 13 nov. 2023.

SPEER, K. E. *et al.* Measuring Heart Rate Variability Using Commercially Available Devices in Healthy Children: A Validity and Reliability Study. **European Journal of Investigation in Health, Psychology and Education, MDPI AG**, 10 jan. 2020. Disponível em: <<http://dx.doi.org/10.3390/ejihpe10010029>>. Acesso em: 21 nov.

STATISTA. Unit sales of true wireless hearables worldwide from 2018 to 2021. **Statista**, c2023. Disponível em: < <https://www.statista.com/statistics/985608/worldwide-sales-volume-true-wireless-hearables/>>. Acesso em: 12 nov. de 2023.

STEAKLEY, L. Harnessing mobile health technologies to transform human health. **Stanford University Communications**, 16 mar. De 2015. Disponível em: <<https://scopeblog.stanford.edu/2015/03/16/harnessing-mobile-health-technologies-to-transform-human-health/>>. Acesso em: 10 nov. De 2023.

STUART, T.; HANNA, J.; GUTRUF, P. Wearable devices for continuous monitoring of biosignals: Challenges and opportunities. **APL Bioengineering AIP Publishing**, 13 abr. 2022. Disponível em: <<http://dx.doi.org/10.1063/5.0086935>>. Acesso em: 17 nov. de 2023.

SVERTOKA, E. *et al.* Wearables for Industrial Work Safety: A Survey Sensors. **MDPI AG**, 2 jun. 2021. Disponível em: <<http://dx.doi.org/10.3390/s21113844>>. Acesso em: 10 nov. de 2023.

TANA, J., FORSS, M., HELLSTÉ, T. 2017. The use of wearables in healthcare – challenges and opportunities. **Arcada Working Papers** 6/2017, ISSN 2342-3064, ISBN 978-952-5260-83-0. . Disponível em: <https://www.researchgate.net/publication/325176555_The_use_of_wearables_in_healthcare-challenges_and_opportunities>. Acesso em: 11 nov. de 2023.

TEIXEIRA FILHO, J. **Gerenciando Conhecimento**: como a empresa pode usar a memória organizacional e a inteligência competitiva no desenvolvimento de negócios. Rio de Janeiro: SENAC, 2000

TRÄGÅRDH, E.; SCHLEGEL, T. T. High-frequency ECG. **Department of Clinical Physiology, Lund University Hospital**, 221 85 Lund, Sweden; **NASA Johnson Space Center, Human Adaptation and Countermeasures Office**, Houston, TX 77058, USA, 2007. Disponível em: <<https://ntrs.nasa.gov/api/citations/20060056493/downloads/20060056493.pdf>>. Acesso em: 9 nov. de 2023.

VARELLA, M. Coração. **Drauzio Varella**, [s.d.]. Disponível em: <<https://drauziovarella.uol.com.br/corpo-humano/coracao/>>. Acesso em: 15 nov. 2023.

UNITED NATIONS. Peace, dignity and equality on a healthy planet. **United Nations**, [s.d] [2023?]. Disponível em: <<https://www.un.org/en/global-issues/population#:~:text=The%20global%20human%20population%20reached,and%20%20billion%20since%201998>>. Acesso em: 18 nov.

WADAS, G. Wearable Activity Trackers Can Be Used to Determinate Health Metrics That Could Support Clinical Care. **John Hopkins Institute for Nanobiotechnology**, 2022. Disponível em: <<https://inbt.jhu.edu/wearable-activity-trackers-can-be-used-to-determine-health-metrics-that-could-support-clinical-care/#:~:text=,health%20and%20cardiovascular%20health%20status>>. Acesso em: 15 nov. 2023.

YASAR, K. Wearable technology. **Tech Target**, 2023. Disponível em: <<https://www.techtarget.com/searchmobilecomputing/definition/wearable-technology>>. Acesso em: 15 nov. 2023.

ZHANG, C.; SHAHRIAR, H.; RIAD, A. B. M. K. Security and Privacy Analysis of Wearable Health Device. **2020 IEEE 44th Annual Computers, Software, and Applications Conference (COMPSAC), IEEE**, jul. 2020. Disponível em: <<http://dx.doi.org/10.1109/COMPSAC48688.2020.00044>>. Acesso em: 8 nov. de 2023.

ANEXO – Detalhamento da Comunicação da Polar

As funcionalidades de transmissão contínua de dados são ditadas pelo serviço PMD, e a sua descrição completa envolve todos os dispositivos com suporte ativo pela empresa *Polar*. O modelo Polar H10 é compatível com poucos dos tipos de grandezas definidas para serem medidas, por vezes sendo compatível com única opção de parâmetros. Por isso, na sua abordagem ao longo do trabalho, não foi dado foco para os outros valores descritos na documentação geral do SDK. Porém, o conteúdo desta seção pode ser usado para esclarecer a origem de alguns códigos usados no desenvolvimento do projeto. Ademais, as tabelas a seguir são úteis para adaptar os programas para novos dispositivos da empresa, mais avançados e que sejam compatíveis com o mesmo SDK.

A Tabela 16 ilustra as diversas grandezas que podem ser medidas por dispositivos da empresa, e os códigos usados para representá-los durante a comunicação com o serviço PMD. A Polar H10 usa apenas o 0 e o 2.

Tabela 16 - Codificação de todas as grandezas medidas pelo SDK da Polar Electro Oy

Tipos de Medição PMD		
Tipo de Medição	Descrição	Unidade
0	Eletrocardiograma (ECG)	Volt (V)
1	Fotopletismograma (PPG)	-
2	Acelerações (ACC)	Força por unidade de massa (g)
3	Intervalo PP (PPI)	Segundo (s)
4	Reservado para Uso Futuro	-
5	Giroscópio (GYRO)	Graus por segundo (°/s)
6	Magnetômetro (MAG)	Gauss (G)
7 .. 255	Reservado para Uso Futuro	-

Fonte: adaptado de Polar Electro Oy (2021a)

A Tabela 17 contém as estruturas de mensagem de acelerômetro, com três níveis de precisão para valores brutos de aceleração. Também há o modo *delta frame*, que é um método de aquisição focado em fornecer apenas as variações nos valores de aceleração medidos. A Polar H10 suporta apenas o tipo 1 com amostras de 16 *bits*.

Tabela 17 - Todos os tipos de frames de dados de aceleração

Tipos de Frame ACC			
Tipo de Frame	Tamanho	Unidade	Descrição
0	3 Bytes	mG	x, y, z, 8 bits
1	6 Bytes	mG	x, y, z, 16 bits
2	9 Bytes	mG	x, y, z, 24 bits
128	n	mG	Delta Frame
3 .. 127, 129 .. 255	-	-	RFU

Fonte: adaptado de Polar Electro Oy (2021a)

O eletrocardiograma suporta apenas uma forma de aquisição, e a sua estrutura é definida na Tabela 18. Aqui é definido o padrão de 3 *bytes* por amostra, e as unidades.

Tabela 18 - Único tipo de frame disponível de eletrocardiograma

Tipos de Frame ECG			
Tipo de Frame	Tamanho	Unidade	Descrição
0	3 Bytes	μV	Eletrocardiograma
1 .. 255	-	-	RFU

Fonte: adaptado de Polar Electro Oy (2021a)

A Tabela 19 a seguir tem grande utilidade para testes e para compreender a compatibilidade com diferentes comandos e dispositivos. Estão representados os códigos de erro para diferentes casos, que são enviados pelo servidor nos momentos apropriados. Erros de parâmetros, instruções e pré-requisitos de conexão são comuns e importantes de serem lidados em aplicações que visam compatibilidade com ampla gama de *wearables*.

Tabela 19 - Códigos de erros possíveis para a comunicação com Ponto de Controle PMD

Códigos de Erro do Ponto de Controle		
Valor	Descrição	Uso
0	SUCESSO	Comando enviado ao Ponto de Controle administrado com sucesso
1	ERRO OPCODE INVÁLIDO	Comando enviado ao Ponto de Controle não é suportado pelo dispositivo
2	ERRO TIPO DE MEDIÇÃO INVÁLIDO	Tipo de medição requisitado não é reconhecido pelo dispositivo
3	ERRO NÃO SUPORTADO	Tipo de medição requisitado não é suportado pelo dispositivo
4	ERRO TAMANHO INVÁLIDO	Tamanho enviado não condiz com o tamanho dos dados
5	ERRO PARÂMETRO INVÁLIDO	Requisição contém parâmetros que impedem a sua realização
6	ERRO JÁ NO ESTADO	Dispositivo já está no estado requisitado
7	ERRO RESOLUÇÃO INVÁLIDA	Medição requisitada possui resolução não suportada pelo dispositivo
8	ERRO TAXA DE AMOSTRAS INVÁLIDA	Medição requisitada possui taxa de amostras não suportada pelo dispositivo
9	ERRO ALCANCE INVÁLIDO	Medição requisitada possui alcance não suportado pelo dispositivo
10	ERRO MTU INVÁLIDO	MTU da conexão não corresponde ao MTU requisitado pelo dispositivo
11	ERRO NÚMERO DE CANAIS INVÁLIDO	Requisição de medição contém número inválido de canais
12	ERRO ESTADO INVÁLIDO	Dispositivo está em um estado inválido
13	ERRO DISPOSITIVO NO CARREGADOR	Dispositivo está conectado ao carregador e não suporta o comando requisitado nesse estado
14 .. 255	RFU	Reservado para Uso Futuro

Fonte: adaptado de Polar Electro Oy (2021a)