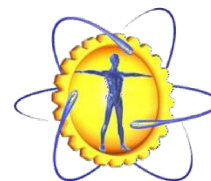




**UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FEELT – FACULDADE DE ENGENHARIA ELÉTRICA
GRADUAÇÃO EM ENGENHARIA BIOMÉDICA**



LÍVIA GABRIELLE VAZ FERREIRA

**PROPOSTA DE MODELO PARA TRANSMISSÃO E ARMAZENAMENTO DE
DADOS EM NUVEM DO EXAME *HOLTER* COM *BACKUP* LOCAL**

Uberlândia
2026

LÍVIA GABRIELLE VAZ FERREIRA

**PROPOSTA DE MODELO PARA TRANSMISSÃO E ARMAZENAMENTO DE
DADOS EM NUVEM DO EXAME *HOLTER* COM *BACKUP* LOCAL**

Trabalho apresentado como requisito parcial de avaliação na disciplina Trabalho de Conclusão de Curso do Curso de Engenharia Biomédica da Universidade Federal de Uberlândia.

Orientador: Prof. Alcimar Barbosa Soares, PhD

Assinatura do Orientador

Uberlândia
2026

AGRADECIMENTOS

Quero registrar o meu agradecimento a todos que me apoiaram durante os anos em que durou esta etapa e que contribuíram para sua conclusão.

Agradeço aos meus pais, Adriane e Jailso, por todo o encorajamento, incentivo e suporte, psicológico e financeiro. Se cheguei ao fim foi porque vocês permitiram que eu desse o primeiro passo e caminhasse ao longo desse caminho. Agradeço à minha irmã, Bruna, pela cumplicidade e apoio, que me incentivou e a quem busquei incentivar, e às minhas avós, Helena e Lenita, que sempre me encorajaram a buscar conhecimento.

Agradeço aos meus amigos, Aline Victória, Carol Carvalho, Carol Martins, Fernanda Flor, Gabriel Ribeiro, Lutiéle Santos e Samara Marques, que, mais do que isso, tornaram-se uma família para mim. Em especial à Fernanda e Gabriel, a quem tenho como irmãos. E à minha amiga Cindy, que mesmo distante foi fundamental para este processo.

À minha namorada, Júlia Tannús, que foi minha principal inspiração na reta final, e que tem me inspirado e incentivado todos os dias, me ajudando a ser melhor.

Ao meu orientador, Alcimar, pela atenção, dedicação e paciência, e que tem uma mente brilhante e nos estimula a pensar fora da caixinha e desenvolver nossa capacidade e nosso potencial.

Por fim, agradeço aos meus colegas do BioLab, por terem me ensinado tanto durante o tempo que passei lá, e me ajudado a crescer e me desenvolver. Principalmente à Ana Clara, que se tornou uma amiga e a quem admiro muito.

Meu muito obrigada a todos vocês por terem me ajudado a concluir minha jornada!

RESUMO

As doenças cardiovasculares (DCV) tornaram-se a principal causa de morte no mundo, com destaque para condições como a doença arterial coronariana (DAC), infarto do miocárdio, insuficiência cardíaca (IC) e arritmias. Diante do envelhecimento populacional, com idade sendo um fator de risco, e da importância do diagnóstico precoce, o eletrocardiograma (ECG) destaca-se como um exame primordial; sua agilidade e eficácia no registro da atividade elétrica cardíaca permitem mensurar a frequência, o ritmo e a dinâmica dos impulsos elétricos de forma precisa. Entretanto, como o ECG possui curta duração, em casos que exigem monitoramento contínuo de 24 horas ou mais, torna-se viável a utilização do *Holter*. Considerando que uma parcela da população enfrenta dificuldades de locomoção até centros preparados para esses exames, a telemedicina e o telemonitoramento surgem como ferramentas para otimizar o processo de forma remota. Assim, este trabalho visou desenvolver um sistema de transmissão de sinais de ECG em tempo real para a nuvem, propondo uma forma simples e acessível para o acompanhamento da saúde cardíaca. Para isso, foram desenvolvidos um protótipo para aquisição de sinal e um sistema de transmissão com *backup* local. A arquitetura utilizou um banco de dados para o processamento e envio de dados para a nuvem através do uso de *threads*, garantindo a agilidade do fluxo de informações. O sistema mostrou-se capaz de realizar o armazenamento híbrido (local e nuvem) e de informar eventos específicos, como condições de bradicardia, taquicardia e a taxa de batimentos por minuto. Por fim, o envio desses dados para um sistema online de acesso restrito confirmou a eficácia da ferramenta no suporte ao monitoramento remoto e seguro dos pacientes.

Palavras-chave: *backup*, ECG, envio para a nuvem, processamento de dados.

ABSTRACT

Cardiovascular diseases (CVD) have become the leading cause of death worldwide, with conditions such as coronary artery disease (CAD), myocardial infarction, heart failure (HF), and arrhythmias being particularly prominent. Given the aging population, with age being a risk factor, and the importance of early diagnosis, the electrocardiogram (ECG) stands out as a primary examination; its speed and effectiveness in recording cardiac electrical activity allow for precise measurement of the frequency, rhythm, and dynamics of electrical impulses. However, since the ECG has a short duration, in cases requiring continuous monitoring of 24 hours or more, the use of Holter monitoring becomes viable. Considering that a portion of the population faces difficulties in traveling to centers equipped for these examinations, telemedicine and telemonitoring emerge as tools to optimize the process remotely. Thus, this work aimed to develop a system for transmitting ECG signals in real time to the cloud, proposing a simple and accessible way to monitor cardiac health. To achieve this, a prototype for signal acquisition and a transmission system with local backup were developed. The architecture used a database for processing and sending data to the cloud through the use of threads, ensuring the agility of the information flow. The system proved capable of performing hybrid storage (local and cloud) and reporting specific events, such as bradycardia, tachycardia, and heart rate. Finally, sending this data to a restricted-access online system confirmed the tool's effectiveness in supporting the remote and secure monitoring of patients.

Keywords: backup, data processing, ECG, sending to the cloud.

LISTA DE ABREVIATURAS E SIGLAS

ADC – *Analog-to-Digital Converter* (Conversor Analógico-Digital)
API – *Application Programming Interface* (Interface de Programação de Aplicação)
BPM – Batimentos por minuto
DC – *Direct Current* (Corrente Contínua)
DCV – Doenças cardiovasculares
ECG – Eletrocardiograma
GND – *Ground* 6(Terra)
LA – *Left Arm* (Braço Esquerdo)
LGPD – Lei Geral de Proteção de Dados
RA – *Right Arm* (Braço Direito)
Ref – Referências
SBC – Sociedade Brasileira de Cardiologia
SPI – *Serial Peripheral Interface* (Interface Periférica Serial)
SUS – Sistema Único de Saúde
Vcc – Alimentação de entrada (*Voltage Common Collector*)
VFC – Variabilidade da frequência cardíaca
Vin – Tensão de entrada (*Input Voltage*)
Vout – Tensão de saída (*Output Voltage*)

SUMÁRIO

CAPÍTULO 1 - INTRODUÇÃO	8
2.1 Justificativa	9
2.2 Objetivos.....	9
2.2.1 Objetivo geral.....	9
2.2.2 Objetivos específicos	9
CAPÍTULO 2 - FUNDAMENTAÇÃO TEÓRICA.....	11
2.3 O coração e a atividade elétrica cardíaca	11
2.3.1 Biopotenciais	11
2.3.2 Potencial de membrana em repouso	11
2.4 Eletrodos para captação de sinais ECG.....	13
2.5 Biopotenciais cardíacos	15
2.6 Estabelecimento do ECG padrão de 12 derivações	16
2.7 <i>Holter</i>	19
2.8 Telemedicina, telemonitoramento e envio para nuvem	20
2.9 Lei Geral de Proteção de dados (LGPD).....	20
2.10 Bancos de Dados	22
CAPÍTULO 3 - METODOLOGIA	24
3.1 Projeto, construção e validação em bancada de um dispositivo <i>Holter</i>	24
3.1.1 Materiais e métodos	24
3.1.2 Protótipo.....	24
3.2 Procedimento para validação com auxílio de sinais <i>Holter</i> em banco de dados....	28
CAPÍTULO 4 - RESULTADOS E DISCUSSÃO.....	31
4.1 Testes do protótipo com sinais sintéticos	31
4.2 Validação do com sinais <i>Holter</i> de banco de dados	32
4.3 Visualização do sinal online.....	35
CAPÍTULO 5 – CONCLUSÃO.....	37
REFERÊNCIAS BIBLIOGRÁFICAS	38
APÊNDICE	41
Código em C++ para aquisição do sinal simulado.	41
Código em <i>Python</i> para a aquisição das informações do sinal, o armazenamento das amplitudes no cartão SD e o envio para a nuvem (<i>Sheets</i>)	45
Código em <i>Python</i> para a extração das amostras do sinal na nuvem, reconstrução do sinal e envio para a Web das ondas e informações.....	49

CAPÍTULO 1 - INTRODUÇÃO

No início do século XX, as doenças cardiovasculares (DCV) eram responsáveis por menos de 10% de todas as mortes no mundo. No final do século, as DCV foram responsáveis por quase metade de todas as mortes no mundo desenvolvido e 25% no mundo em desenvolvimento. Este aumento global nas DCV resulta em uma mudança drástica no estado de saúde de indivíduos em todo o mundo ao longo do século XX, graças, em grande parte, à melhoria da nutrição e das medidas de saúde pública [1].

No Brasil, as doenças cardiovasculares são as responsáveis pelo maior número de mortes da população, sendo o dobro das mortes causadas por câncer, 2,3 vezes mais que causas externas, como acidentes e atos de violência, 3 vezes mais do que doenças respiratórias e 6,5 vezes mais que todas as infecções, incluindo a *AIDS* (Síndrome da Imunodeficiência Adquirida). A cada hora, cerca de 46 pessoas perdem a vida em decorrência de alguma doença cardiovascular. Grande parte dessas mortes poderiam ser postergadas, ou até evitadas, com cuidados preventivos e medidas terapêuticas, que um diagnóstico e um acompanhamento podem proporcionar. [2]

No âmbito das doenças cardíacas, os cuidados preventivos e as medidas terapêuticas dependem de diagnóstico e identificação precoces. No entanto, a identificação precoce de tais doenças pode encontrar diversos obstáculos, como, por exemplo, a variedade de apresentações clínicas das doenças cardíacas, a sobreposição de sintomas e a falta de especificidade de certos métodos diagnósticos tradicionais, causadas pela escassez geográfica de profissionais [3].

Segundo informações do DATASUS referentes a 2024, (dados mais atuais disponibilizados pelo site) o Brasil registrou 1.532.015 óbitos, dos quais 26,01% (398.605) decorreram de doenças do aparelho circulatório. Entre as patologias com maior índice de mortalidade, destacam-se o infarto agudo do miocárdio, com 94.248 casos, seguido pela hipertensão essencial, insuficiência cardíaca, doença cardíaca hipertensiva e doença isquêmica crônica [4]. Diante desse cenário epidemiológico, o eletrocardiograma (ECG) consolidou-se como o padrão-ouro (gold standard) para a avaliação da condição cardíaca e o diagnóstico preciso dessas patologias. [4-7].

O Sistema Único de Saúde (SUS), consolidado como o maior sistema público de saúde do mundo, do qual 76% da população brasileira depende diretamente, tem um custo de aproximadamente R\$1.000.000.000,00 por ano apenas com procedimentos cardiovasculares. A maioria dessas doenças são silenciosas, evidenciando a importância de acompanhamento e investigação periódica da saúde cardiovascular. Apesar dos avanços tecnológicos e dos modernos procedimentos de atendimento, dados da Sociedade Brasileira de Cardiologia (SBC) apontam que, a cada 90 segundos, uma pessoa morre no Brasil em decorrência de DCV [8-9].

O Ministério da Saúde lançou, em 2025, o programa “Agora tem especialista”, com foco em seis áreas estratégicas, entre elas a cardiologia, visando enfrentar um dos maiores desafios históricos do SUS: o tempo de espera na rede pública. Outro eixo do programa tem como foco a expansão da telessaúde, visando descentralizar o atendimento e reduzir as barreiras geográficas que isolam pacientes de regiões desassistidas. Nesse contexto, o desenvolvimento de ferramentas de monitoramento remoto não apenas corrobora com as

diretrizes governamentais de transformação digital na saúde, como se apresenta como uma solução viável para otimizar a regulação e o fluxo de atendimento especializado, permitindo que o suporte à decisão médica ocorra de forma contínua e integrada à nuvem. Estima-se que a estratégia possa reduzir em até 30% o tempo de espera por atendimento especializado no SUS [10].

2.1 Justificativa

O crescente número de casos de doenças cardíacas no Brasil e os altos índices de mortalidade por tais enfermidades ilustram a importância de ferramentas acessíveis que possibilitem o diagnóstico e o acompanhamento da saúde cardíaca dos pacientes, independentemente da localização e da escassez geográfica de cardiologistas. Dentre os diversos exames solicitados rotineiramente por cardiologistas, destaca-se o *Holter*, cujo objetivo é acompanhar a atividade elétrica do coração por um período estendido, comumente entre 24 e 48 horas [11]. Para tal, eletrodos são fixados ao corpo do sujeito, e a atividade cardíaca é amostrada e armazenada em um cartão de memória local (SD). Após o período de aquisição definido o sujeito retorna a clínica onde os dados são descarregados num computador para posterior processamento.

Com o advento de sistemas computacionais em nuvem, pode-se utilizar tais recursos para otimizar a coleta e armazenamento dos sinais pelo equipamento *Holter*, bem como possibilitar o acompanhamento dos pacientes em tempo real, à distância, otimizando ainda o processo, em casos de danos aos cartões de memória locais, dentre outros.

Diante deste quadro, este trabalho visa propor uma solução quanto a lacuna no acompanhamento contínuo de pacientes com dificuldades de acesso a centros especializados. Espera-se contribuir para a melhoria na qualidade do atendimento ao reduzir o tempo de resposta em diagnósticos de eventos cardíacos e evitar deslocamentos desnecessários, por meio da criação de um sistema que possibilita o telemonitoramento cardíaco em tempo real, ou seu armazenamento e envio posteriormente, com base no sistema *Holter*.

2.2 Objetivos

2.2.1 Objetivo geral

Desenvolver um sistema de telemonitoramento cardíaco, com uma interface simples e uma dupla estratégia de *backup*, como forma de otimizar o acompanhamento e o diagnóstico da saúde cardíaca.

2.2.2 Objetivos específicos

- Levantar dados sobre a incidência de doenças cardíacas no Brasil;
- Revisão de literatura sobre os principais tipos de sinais eletrocardiograma e suas alterações;
- Desenvolver um sistema de transmissão de dados, com capacidade armazenamento local e envio para a nuvem em tempo real;

- Desenvolver um *Hardware* para aquisição de um sinal simulado, testando a capacidade de receber dados por um longo período de tempo e realizar armazenamento e envio das informações;
- Realizar testes preliminares para avaliar a viabilidade técnica do sistema proposto.

CAPÍTULO 2 - FUNDAMENTAÇÃO TEÓRICA

2.3 O coração e a atividade elétrica cardíaca

O coração humano é composto por quatro câmaras: dois átrios e dois ventrículos, formados por células miocárdicas. Impulsos elétricos naturais se propagam por essas câmaras, sendo os responsáveis por coordenar as contrações dessas estruturas que, por sua vez, mantêm o fluxo sanguíneo no sistema cardiovascular. O coração funciona com base nesses sinais elétricos [12].

Para que a bomba cardíaca tenha um desempenho adequado, é necessária uma sincronia perfeita entre o período em que o músculo se encontra relaxado (permitindo o enchimento das câmaras) e o período de contração, o que possibilita aplicar pressão (energia potencial) e velocidade (energia cinética) ao sangue, garantindo a circulação sanguínea. A sincronia da contração das câmaras cardíacas é garantida pela geração e propagação de potenciais elétricos (potenciais de ação) ao longo do sincício elétrico miocárdico. Alterações na atividade elétrica do coração irão apresentar repercussões deletérias na função da bomba cardíaca, de forma a comprometer seu funcionamento. O eletrocardiograma (ECG) é o exame-padrão utilizado para avaliação da geração e propagação da atividade elétrica do coração [12].

2.3.1 Biopotenciais

Biopotenciais são as diferenças de potencial elétrico provenientes das condições do meio interno e externo à célula, decorrentes de sua atividade metabólica e da distribuição desigual de íons através da membrana celular [13]. No contexto cardíaco, esses biopotenciais manifestam-se através do potencial de ação das células do miocárdio, onde o fluxo ordenado de íons (como sódio, potássio e cálcio) gera correntes iônicas que se propagam pelo tecido [12]. Essa atividade elétrica celular resulta em campos elétricos que podem ser detectados na superfície do corpo através de eletrodos, constituindo a base fundamental para a formação do sinal de eletrocardiograma. Portanto, sua compreensão depende do conhecimento de alguns aspectos referentes à fisiologia celular e à dinâmica de íons, que são dotados de partículas portadoras de carga elétrica capazes de gerar correntes sob a influência de um campo elétrico [12, 13].

Os fluidos intra e extracelulares, por sua vez, são ricos em diversos tipos de íons dotados de grande mobilidade. Essa característica permite que o movimento dessas cargas, seja por difusão ou pela presença de um campo elétrico, estabeleça as correntes iônicas [13, 14].

2.3.2 Potencial de membrana em repouso

Os fluidos são separados por uma membrana semipermeável, conhecida como membrana citoplasmática. A concentração de íons sódio (Na^+) no meio externo é maior que no meio interno, ao passo que a concentração de íons potássio (K^+) é maior no meio intracelular. Essa diferença de concentrações, ilustrada na Figura 1, origina gradientes de

difusão que impulsionam os íons Na^+ para o interior da célula e os íons K^+ para fora, movimento que ocorre exclusivamente através dos canais iônicos [13], [14].

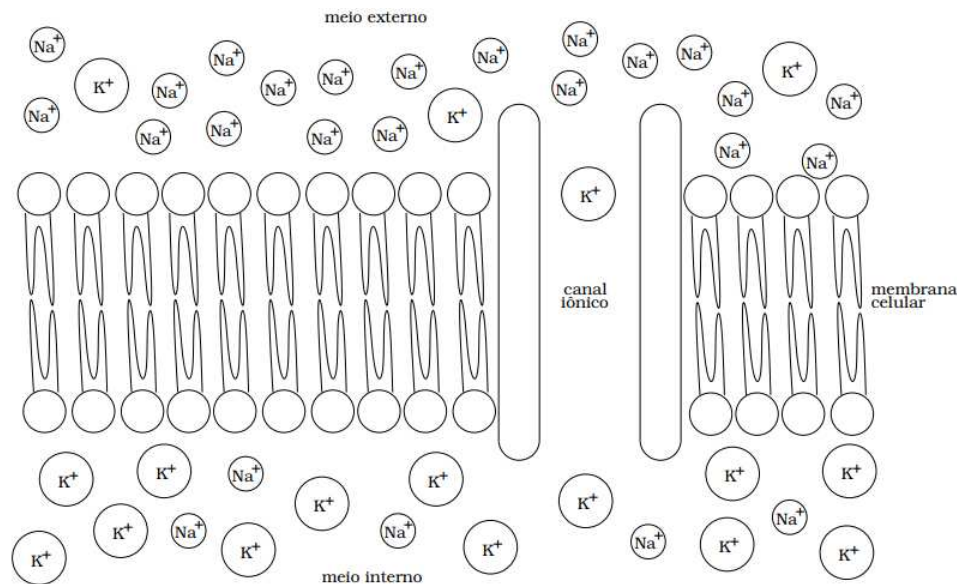


Figura 1: Ilustração esquemática da separação dos meios intra e extracelular pela membrana citoplasmática.
Fonte: [13].

Contudo, a transferência de íons entre esses meios só é possível devido à existência de canais iônicos que são capazes de transpassar a membrana celular, permitindo que alguns íons Na^+ entrem na célula, ao mesmo tempo em que permitem que alguns íons K^+ se movam para o meio externo. A quantidade de canais iônicos disponíveis determina a permeabilidade da membrana a determinados íons [13].

Bombas iônicas possibilitam a troca de íons K^+ do meio exterior por íons Na^+ do interior da célula. O tipo de bomba em questão é conhecido como bomba de sódio-potássio. Um equilíbrio é estabelecido quando as forças químicas, devido ao gradiente de difusão dos íons, se igualam às forças elétricas, devido à concentração de cargas elétricas de mesma polaridade [13], [14].

Nesta situação, existe uma diferença de potencial, denominada potencial de repouso da membrana, do interior para o exterior da célula. O valor desta diferença de potencial depende das concentrações dos vários íons envolvidos dentro e fora da célula, além da permeabilidade da membrana a cada íon em questão [13], [14].

Nas células excitáveis, ou seja, neurônios e células musculares, existem canais que, embora presentes, não se encontram acessíveis à movimentação dos íons permanentemente. Tais canais são classificados como gatilháveis, uma vez que se abrem à passagem de íons somente em situações específicas [13], [14].

Há uma grande variedade destes canais, ativados pela variação de uma grandeza física ou pela presença de uma determinada substância. Dois tipos de canais gatilháveis são particularmente interessantes ao estudo dos potenciais de ação: os canais de sódio e os canais de potássio ativados pela despolarização da membrana [13], [14].

Os canais de Na^+ , ativados pela despolarização da membrana, se abrem rapidamente quando o potencial da membrana atinge um limiar próximo a -55 mV . Uma vez abertos, o

influxo de íons Na^+ acelera o processo de despolarização, causando a abertura de um número cada vez maior dos canais, gerando uma reação em cadeia. Outra característica importante destes canais é que eles permanecem abertos por um curto período de tempo, da ordem de 1 ms. Após este período, o canal é desativado, estando novamente pronto para ativação somente após a repolarização da membrana [13], [14].

Podem ser identificadas quatro fases componentes do potencial de ação:

1. membrana em repouso;
2. despolarização: abertura dos canais de Na^+ e K^+ . Após a despolarização da membrana atingir o limiar de ativação, a abertura dos canais de Na^+ dispara o processo em cadeia;
3. repolarização: fechamento dos canais de Na^+ . A abertura dos canais de K^+ é lenta, de maneira que apenas uma pequena parcela destes canais estará aberta no momento em que todos os canais de Na^+ estiverem abertos. Fechados os canais de Na^+ , o efluxo de íons K^+ provoca a repolarização da membrana, no sentido de restabelecer o potencial de repouso; e
4. fechamento dos canais de K^+ . O efluxo de íons K^+ torna a membrana hiperpolarizada. O restabelecimento do potencial de repouso só ocorrerá após o fechamento de todos os canais gatilháveis de potássio.

Embora o ciclo de sódio e potássio fundamente o potencial de ação celular, os potenciais cardíacos apresentam uma dependência crítica dos íons cálcio (Ca^{2+}). Diferente de outras células excitáveis, o miocárdio possui canais de cálcio do tipo L que se abrem durante a despolarização, permitindo um influxo lento desse íon. Esse processo resulta na fase de platô (fase 2 do potencial de ação cardíaco), que prolonga a duração do potencial e é essencial para garantir o tempo de esvaziamento das câmaras cardíacas e a subsequente contração muscular [12].

Os potenciais de ação constituem um importante mecanismo de transferência de informação. As contrações em fibras musculares, bem como a transmissão de estímulos nervosos, são viabilizadas pela ocorrência de potenciais de ação nos tecidos envolvidos. A atividade conjunta de células componentes de um tecido, seja em uma fibra muscular ou em um nervo, resulta nos biopotenciais [14], [15].

A somatória desses potenciais de ação individuais, ao se propagarem de forma coordenada pelo tecido cardíaco, gera um campo elétrico que atravessa o volume condutor do corpo humano. Para a realização do exame de eletrocardiograma, esse sinal é captado na superfície da pele, através de eletrodos [14].

O registro destes biopotenciais contém informações sobre o funcionamento da estrutura envolvida e, por esta razão, é de grande valia no diagnóstico de doenças e identificação de anomalias [14], [15].

2.4 Eletrodos para captação de sinais ECG

O eletrodo é um componente que atua como um transdutor, transformando a corrente iônica do corpo em corrente elétrica. Essa conversão acontece graças a reações químicas de oxirredução (oxidação e redução), ocorridas na superfície do eletrodo [14].

Reações de oxirredução são reações químicas que envolvem transferência de elétrons, e sempre acontecem em conjunto [15]. Na oxidação, tem-se a perda de elétrons por átomos do eletrodo, liberando íons para o eletrólito (meio aquoso onde existem íons dissolvidos). Na redução, tem-se o recebimento de elétrons por íons contidos no eletrólito, gerando o acúmulo do material correspondente sobre o eletrodo [14].

As reações de oxirredução iniciam-se naturalmente quando um eletrodo é imerso em uma solução eletrolítica a qualquer temperatura acima do zero absoluto. A liberação de íons metálicos na solução eletrolítica e, simultaneamente, o acúmulo de elétrons no metal, indicado na Figura 2, origina um desequilíbrio eletrostático na fronteira entre eletrodo e eletrólito. Os elétrons liberados na oxidação do metal irão acumular-se na superfície do eletrodo. A carga elétrica negativa distribuída sobre a superfície do metal atrai cátions presentes no eletrólito devido a forças coulombianas, formando uma dupla camada elétrica na fronteira [13].

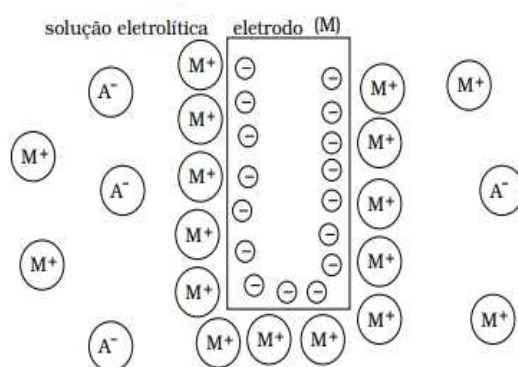


Figura 2: Formação da dupla camada elétrica na interface do eletrodo com a solução. Fonte: [13].

Assim como os átomos do metal sofrem oxidação, os cátions presentes na dupla camada elétrica, situada na fronteira entre eletrodo e eletrólito, sofrem redução. Nesse sentido, após o equilíbrio ser atingido, a taxa de oxidação torna-se igual à de redução, resultando na estabilização da dupla camada elétrica, e, conseqüentemente, do campo elétrico nessa fronteira. Por conseguinte, haverá uma diferença de potencial entre o eletrodo e a solução eletrolítica, em consequência do aparecimento da dupla camada elétrica. A formação da distribuição de cargas na fronteira origina, também, uma capacitância entre o eletrodo e a solução eletrolítica [14].

Se a taxa de redução é maior que a taxa de oxidação, nota-se que existe uma corrente resultante de elétrons deixando o eletrodo, o que significa uma corrente elétrica, em seu sentido convencional, penetrando no eletrodo. Essa corrente tem a mesma direção da corrente de íons M^+ na solução. Por outro lado, se a taxa de oxidação for maior, haverá uma corrente resultante de elétrons penetrando o eletrodo, indicando a existência de uma corrente elétrica em sentido contrário [14].

Dessa maneira, a corrente iônica pode ser convertida em corrente elétrica, conversão proveniente das reações de oxirredução na fronteira entre o eletrodo e eletrólito, o que permite a medição do potencial cardíaco. A medição de biopotenciais depende da utilização de eletrodos em contato com tecidos biológicos. Os fenômenos elétricos oriundos do metabolismo celular provocam a movimentação de íons, que formam correntes iônicas. Dessa forma, a primeira etapa da medição consiste na transformação das correntes iônicas em correntes eletrônicas, que serão conduzidas, através de cabos, ao circuito de medição. Esta transformação

consiste em um processo eletro-químico, determinante das características elétricas da interface entre tecido e eletrodo [14].

O principal eletrodo utilizado para aquisição de sinais de ECG é o eletrodo de prata/cloreto de prata (Ag/AgCl). Trata-se de um eletrodo de simples fabricação e cujas características se aproximam do tipo não-polarizável (os cátions M^+ que se movem na direção oposta ao eletrodo elevam a taxa de oxidação na fronteira, de modo que a quantidade de elétrons inseridos no eletrodo é maior que a quantidade retirada pela redução [14]).

O eletrodo de prata/cloreto de prata apresenta as seguintes características elétricas:

- baixa resistência entre eletrodo e eletrólito;
- baixa capacitância entre eletrodo e eletrólito; e
- baixo ruído comparado aos eletrodos metálicos.

2.5 Biopotenciais cardíacos

Biopotenciais produzidos na membrana celular se propagam para tecidos vizinhos devido a correntes iônicas, de maneira que podem ser medidos em tecidos distantes daqueles em que se originaram. Os sinais de ECG consistem em uma classe de biopotenciais, coletados na superfície da pele, que refletem a contração do músculo cardíaco. Portanto, registram informações sobre o funcionamento do coração [14].

O ciclo cardíaco origina-se a partir da despolarização das células do nódulo sinusal, que se propagam pelos átrios direito e esquerdo. A partir da análise da equivalência temporal entre os registros de potencial em diferentes regiões do coração e as ondas do eletrocardiograma, como exibido na Figura 3, é possível notar que a ativação atrial gera uma onda conhecida como onda P, ou seja, a ativação elétrica responsável por gerar a contração das câmaras superiores do coração [12].

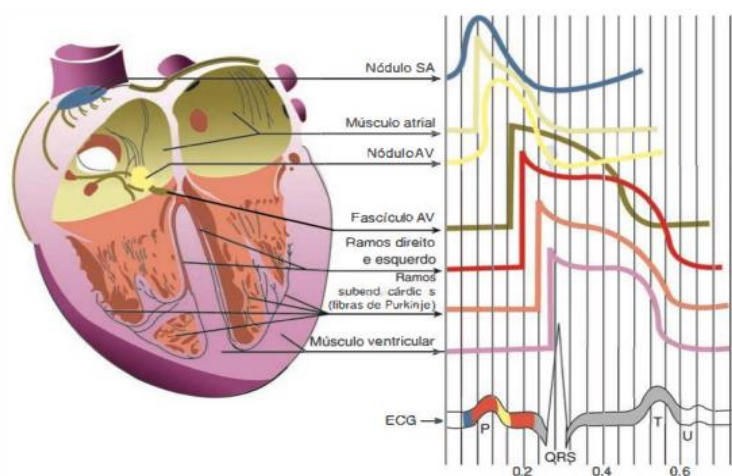


Figura 3: Potenciais de ação associados às diversas estruturas cardíacas e o potencial resultante. Fonte: [12].

Essa excitação ventricular gera um conjunto de ondas pontiagudas e de rápida inscrição, chamado de complexo QRS. As ondas QRS representam a despolarização dos ventrículos. Destes, o pico R é o mais alto e mais fácil de ser identificado [12].

Em seguida, tem-se a fase três dos potenciais de ação dos ventrículos, na qual a onda T representa a repolarização ventricular. Uma onda U, que pode aparecer depois da onda T, tende a ser determinada pela repolarização das fibras ventriculares com potenciais de ação mais longos [12]. O traçado do ECG, apresentado na Figura 4, consiste no ciclo isolado de onda, conhecido como PQRSST, que representa a atividade elétrica do coração durante um batimento.

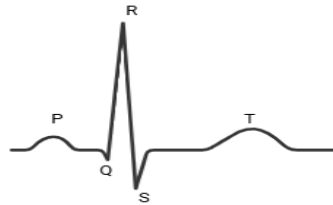


Figura 4: Formato de onda típica de um sinal ECG. Fonte: Autora.

2.6 Estabelecimento do ECG padrão de 12 derivações

O ECG de 12 derivações atual deriva de 9 eletrodos posicionados na superfície do corpo em locais específicos e previamente acordados. Três eletrodos de derivação dos membros (que levaram aos registros bipolares iniciais de ECG de 3 derivações no plano frontal, desenvolvidos por Willem Einthoven) são colocados no pulso do braço direito, no pulso do braço esquerdo e no tornozelo que une o pé esquerdo [17].

Esses posicionamentos formam um triângulo equilátero invertido imaginário, chamado triângulo de Einthoven, que é delimitado entre o ombro direito, o ombro esquerdo e o púbis, com o coração considerado a fonte elétrica localizada em seu centro, descrito abaixo na Figura 5. Os ombros e o púbis são análogos aos eletrodos posicionados (de forma mais conveniente) no braço direito, ou *left arm* (LA), um segundo eletrodo é posicionado abaixo da clavícula direita, próximo ao braço direito, ou *right arm* (RA), visto que os registros obtidos dos ombros e do púbis mostraram-se praticamente indistinguíveis. Além desses três eletrodos de ECG nos membros, outro eletrodo é colocado no tornozelo, unindo o pé direito e conectado à referência (o terra), no pé esquerdo, ou *left foot* (LF) [17].

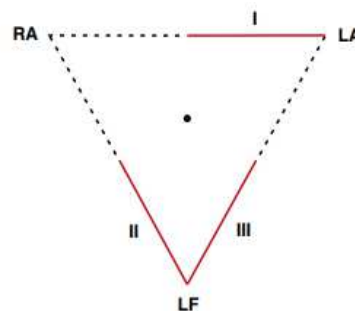


Figura 5: Triângulo Einthoven. Fonte: [16].

Os eletrodos de ECG de referência (-) e de exploração (+) e a direção do eixo da derivação correspondente no plano frontal são os seguintes:

1. O eixo da derivação I estende-se do braço direito (RA) (-) ao braço esquerdo (LF) (+).
2. O eixo da derivação II estende-se do braço direito (RA) (-) ao pé esquerdo (LF) (+).
3. O eixo da derivação III estende-se do braço esquerdo (LA) (-) ao pé esquerdo (LF) (+).

Ao reorganizar esses três eixos de derivação de forma que seus centros se sobreponham, gera-se um sistema de referência triaxial, apresentado na Figura 6 e o plano frontal é então dividido em 6 segmentos, cada um delimitando 60°. Por convenção, e com o propósito de definir o eixo elétrico médio das várias fases do ciclo cardíaco, o eixo da derivação I é marcado como 0° em sua extremidade (+). Além disso, como os eixos posicionados no sentido horário a partir dessa linha horizontal de 0° são designados como positivos (inversamente, aqueles posicionados no sentido anti-horário são designados como negativos), as extremidades (+) dos eixos das derivações II e III são orientadas a +60° e +120°, respectivamente [17].

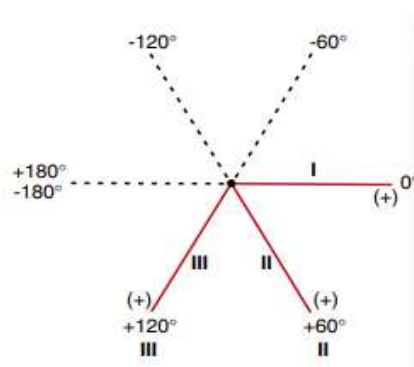


Figura 6: Sistema de referência triaxial para derivações bipolares. Fonte: [16].

Cerca de duas décadas após o estabelecimento desses registros de ECG bipolar de 3 derivações, que revelavam a atividade elétrica do coração no plano frontal, o médico cardiologista Frank N. Wilson concebeu e promoveu a noção de derivações unipolares, conhecidas como precordiais, no plano horizontal e, nesse processo, desenvolveu o que hoje conhecemos como Terminal Central de Wilson (WCT) [17].

Seis eletrodos exploratórios (+), colocados no tórax, levaram à introdução das 6 derivações precordiais unipolares (V1 a V6), seguindo as recomendações da *American Heart Association* e da *Cardiac Society of Great Britain*, publicadas em 1938 [17].

Os eletrodos de ECG exploratórios das 6 derivações precordiais unipolares (+), referenciadas ao potencial 0 do WCT (-), são posicionados conforme apresentado na Figura 7.

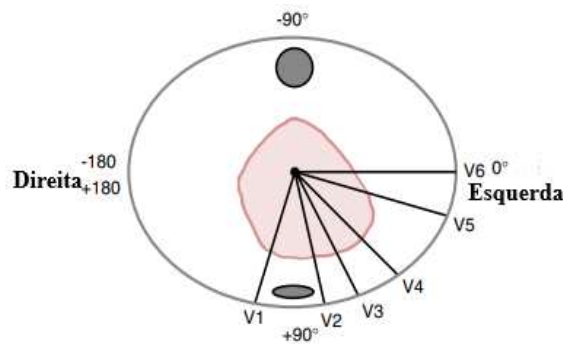


Figura 7: Posicionamento dos eletrodos associados às derivações precordiais. Fonte: [17].

Essa configuração resulta nas seguintes derivações:

- V1: 4º espaço intercostal à direita do esterno;
- V2: 4º espaço intercostal à esquerda do esterno;
- V3: Ponto médio entre V2 e V4;
- V4: 5º espaço intercostal na linha hemiclavicular;
- V5: Na linha axilar anterior, no mesmo plano horizontal que V4;
- V6: Na linha axilar média, no mesmo plano horizontal que V4 e V5

O eixo elétrico médio das várias fases do ciclo cardíaco no plano horizontal é calculado de maneira semelhante ao do plano frontal. Assim, o eixo de derivação de V6 recebe o valor de 0°, os eixos direcionados anteriormente a partir desse eixo de 0° recebem valores positivos e os eixos direcionados posteriormente recebem valores negativos [17].

Alguns anos após a introdução das derivações precordiais V1 a V6, e em um esforço para visualizar regiões do coração potencialmente não detectadas no plano frontal com apenas 3 derivações bipolares dos membros (I, II e III), o também médico cardiologista Emanuel Goldberger desenvolveu as derivações unipolares baseadas no vetor direito, ou *vector right* (VR), vetor esquerdo, ou *vector left* (VL) e vetor pé, ou *vector foot* (VF), usando os eletrodos exploradores (+) do RA, LA e LF, respectivamente, e o WCT como referência (-). No entanto, como as amplitudes dos sinais eram muito pequenas, Goldberger resolveu o problema combinando apenas dois eletrodos de referência das derivações dos membros do WCT (-) e usando o outro eletrodo do membro como explorador. Essa abordagem levou a um aumento de aproximadamente 50% na amplitude dos sinais, e as derivações foram então denominadas aVR, aVL e aVF. A nomenclatura aV, que precede as letras R, L e F, significa Vetor aumentado [17].

Com a adição dessas três derivações unipolares ao plano frontal, um sistema de referência hexaxial foi gerado e o plano frontal foi então dividido em 12 segmentos, cada um delimitando 30°. Assim, o ECG padrão de 12 derivações foi finalmente estabelecido em 1954, seguindo uma recomendação publicada pela *American Heart Association*. As diferentes ondas e intervalos característicos de um ECG humano normal são ilustrados na Figura 8 [17].

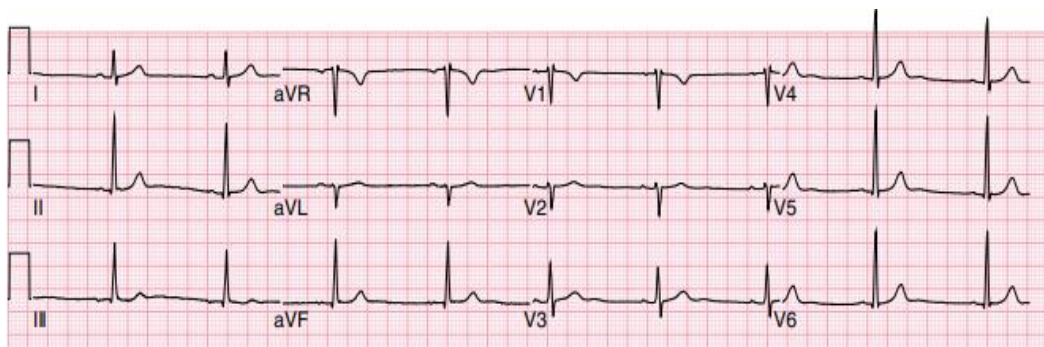


Figura 8: Exemplo de um ECG normal de 12 derivações. Fonte: [16].

O traçado de um ECG normal apresenta uma substancial variabilidade interindividual, o que significa que padrões considerados saudáveis podem variar significativamente de uma pessoa para outra. Embora sistemas computadorizados automáticos consigam medir esses dados com precisão, uma leitura completa por um médico qualificado que possa integrar todos os dados clínicos é fundamental para uma interpretação e diagnóstico final corretos do ECG [16].

2.7 Holter

Com a introdução do eletrocardiograma na prática clínica no início do século XX, seu valor na avaliação das arritmias tornou-se evidente. A eletrofisiologia cardíaca emergiu como uma disciplina clínica à medida que técnicas diagnósticas invasivas e não invasivas foram desenvolvidas, como o monitoramento por Holter [11].

Apesar da importância, o ECG tem suas limitações, uma vez que só registra a atividade cardíaca instantânea. Como certos ritmos cardíacos anormais podem ocorrer esporadicamente ou sob certas condições, como estresse ou prática de atividades físicas, o Holter tem maiores chances de detectar tais anomalias, já que vai avaliar os sinais por um período mais extenso de tempo, tendo um intervalo maior para detectar quaisquer batimentos cardíacos ou ritmos anormais que possam estar causando os sintomas [18].

O monitor Holter é um sistema de eletrocardiografia ambulatorial descoberto pelo Dr. Norman J. Holter e seu time no ano de 1957. Essa descoberta teve um grande impacto na forma como o ritmo cardíaco é avaliado fora do ambiente hospitalar. Esse dispositivo funciona de acordo com o princípio do galvanômetro para registrar sinais de ECG de um indivíduo ao longo de suas atividades diárias, de forma contínua. Desde então, ele passou a constituir a espinha dorsal da detecção e análise de ritmos em eletrofisiologia cardíaca [11].

O dispositivo passou por avanços significativos, tornando-se mais compacto, preciso e acessível desde a sua introdução, e a tecnologia foi tornando a gravação da eletrocardiografia cada vez mais confiável. No início de 1990, surgiram os primeiros discos rígidos com tamanho reduzido e um custo mais acessível, sendo substituídos, depois, pelos cartões de memória instantânea. A qualidade dos registros também avançou, possibilitando a gravação da atividade cardíaca por períodos maiores e o envio de dados pela Internet [19].

Um Holter costuma avaliar a atividade cardíaca por um período de 24 a 48 horas, possibilitando que anormalidades no eletrocardiograma sejam detectadas enquanto o paciente vive o período normalmente. Entre as aplicabilidades do exame, podemos citar: detecção de alterações do segmento ST, efeito terapêutico de antiarrítmicos e drogas anti-isquêmicas, variação da frequência cardíaca, análise de marca-passos e potenciais tardios com eletrocardiograma de alta resolução [18]. O equipamento também representa um importante

instrumento para a abordagem inicial a pacientes com síncope e representativa probabilidade de apresentar arritmias [19].

O sinal ECG é obtido, de forma padrão, ao se colocar pequenos adesivos de plástico (eletrodos) em determinados pontos da barriga (abdômen) e do tórax, com o paciente em repouso. Esses eletrodos são conectados a um aparelho de ECG através de fios, possibilitando que a atividade cardíaca possa ser medida, registrada e impressa, não enviando nenhuma eletricidade ao corpo [18].

Cada parte diferente do coração é controlada por impulsos elétricos naturais. Isso é o que mantém o sangue fluindo adequadamente pelo corpo. O ECG registra os impulsos para apontar a velocidade e o ritmo dos batimentos cardíacos, apontando irregularidades. Também permite o registro da força e o tempo dos impulsos elétricos. Dessa forma, alterações no exame podem indicar diversas condições cardíacas [20].

O exame pode ser utilizado para investigar frequência cardíaca, batimentos cardíacos irregulares e palpitações, avaliar o risco de futuros eventos cardíacos em determinadas condições, o funcionamento de um marcapasso, descobrir o desempenho do tratamento para ritmos cardíacos anormais complexos ou até para verificar o quão rápida ou lenta a frequência cardíaca fica durante o dia, além de verificar pausas no ritmo cardíaco [20].

2.8 Telemedicina, telemonitoramento e envio para nuvem

A telemedicina pode ser entendida como o uso de tecnologia de telecomunicações para diagnóstico médico, tratamento, atendimento ao paciente e monitoramento remoto. Atualmente, é reconhecida por reduzir os custos médicos, uma vez que, fora do ambiente hospitalar, permite que cuidados médicos sejam administrados à distância. Além disso, a telemedicina é utilizada por instituições clínicas em diversas áreas, como no monitoramento residencial a longo prazo de pacientes com doenças cardiovasculares [21].

Isso é possível graças a dispositivos portáteis de telemonitoramento, que são comumente usados para registrar a arritmia no momento de sua ocorrência ou para registrar sinais de ECG, de forma a permitir que os especialistas observem tendências durante o período de registro. Esses sistemas de telemonitoramento podem ser divididos em duas partes: uma em tempo real, em que os dados do paciente estão disponíveis no servidor imediatamente após a aquisição, e outra por meio de armazenamento e encaminhamento de dados, onde esses serão acessados posteriormente [21].

O sistema escolhido para envio e armazenamento dos resultados foi o Planilhas Google, em que as planilhas são armazenadas na nuvem, no *Google Drive*, no mesmo formato de nuvem utilizado por [22] em seu projeto, *Online Monitoring Health Station Using Arduino Mobile Connected to Cloud service: "Heart Monitor" System*. Nessa configuração, os dados são exibidos em tempo real e salvos, permitindo o compartilhamento e garantindo fácil acessibilidade para os profissionais de saúde que têm acesso à planilha, por meio de um *link*, disponível a partir de qualquer dispositivo com conexão à internet.

2.9 Lei Geral de Proteção de dados (LGPD)

A Lei Geral de Proteção de Dados (LGPD), de 18 de agosto de 2018, dispõe sobre o tratamento de dados pessoais, inclusive nos meios digitais, por pessoa natural ou por pessoa

jurídica de direito público ou privado, com o objetivo de proteger os direitos fundamentais de liberdade e de privacidade e o livre desenvolvimento da personalidade da pessoa natural [23].

Tal Lei tem como fundamento o respeito à privacidade, a autodeterminação informativa, a liberdade de expressão, de informação, de comunicação e de opinião, a inviolabilidade da intimidade, da honra e da imagem, o desenvolvimento econômico, tecnológico e a inovação, a livre iniciativa, a livre concorrência, a defesa do consumidor e os direitos humanos, o livre desenvolvimento da personalidade, a dignidade e o exercício da cidadania pelas pessoas naturais [23].

No âmbito do tratamento de dados pessoais, o Art. 5º da Lei nos traz algumas considerações:

I - dado pessoal: informação relacionada à pessoa natural identificada ou identificável;

II - dado pessoal sensível: dado pessoal sobre origem racial ou étnica, convicção religiosa, opinião política, filiação a sindicato ou a organização de caráter religioso, filosófico ou político, dado referente à saúde ou à vida sexual, dado genético ou biométrico, quando vinculado a uma pessoa natural;

III - dado anonimizado: dado relativo a titular que não possa ser identificado, considerando a utilização de meios técnicos razoáveis e disponíveis na ocasião de seu tratamento;

IV - banco de dados: conjunto estruturado de dados pessoais, estabelecido em um ou em vários locais, em suporte eletrônico ou físico;

V - titular: pessoa natural a quem se referem os dados pessoais que são objeto de tratamento;

Ainda sobre os dados sensíveis, a Seção II (Do Tratamento de Dados Pessoais Sensíveis), traz as seguintes informações:

Art. 11. O tratamento de dados pessoais sensíveis somente poderá ocorrer nas seguintes hipóteses:

I - quando o titular ou seu responsável legal consentir, de forma específica e destacada, para finalidades específicas;

II - sem fornecimento de consentimento do titular, nas hipóteses em que for indispensável para:

a) cumprimento de obrigação legal ou regulatória pelo controlador;

b) tratamento compartilhado de dados necessários à execução, pela administração pública, de políticas públicas previstas em leis ou regulamentos;

c) realização de estudos por órgão de pesquisa, garantida, sempre que possível, a anonimização dos dados pessoais sensíveis;

d) exercício regular de direitos, inclusive em contrato e em processo judicial, administrativo e arbitral, este último nos termos da Lei nº 9.307, de 23 de setembro de 1996 (Lei de Arbitragem);

e) proteção da vida ou da incolumidade física do titular ou de terceiro;

f) tutela da saúde, exclusivamente, em procedimento realizado por profissionais de saúde, serviços de saúde ou autoridade sanitária; ou

g) garantia da prevenção à fraude e à segurança do titular, nos processos de identificação e autenticação de cadastro em sistemas eletrônicos, resguardados os direitos mencionados no art. 9º desta Lei e exceto no caso de prevalecerem direitos e liberdades fundamentais do titular que exijam a proteção dos dados pessoais.

O Art. 7º informa que o tratamento de dados pessoais somente poderá ser realizado nas seguintes hipóteses:

I - mediante o fornecimento de consentimento pelo titular;

II - para o cumprimento de obrigação legal ou regulatória pelo controlador;

III - pela administração pública, para o tratamento e uso compartilhado de dados necessários à execução de políticas públicas previstas em leis e regulamentos ou respaldadas em contratos, convênios ou instrumentos congêneres, observadas as disposições do Capítulo IV desta Lei;

IV - para a realização de estudos por órgão de pesquisa, garantida, sempre que possível, a anonimização dos dados pessoais;

V - quando necessário para a execução de contrato ou de procedimentos preliminares relacionados a contrato do qual seja parte o titular, a pedido do titular dos dados;

VI - para o exercício regular de direitos em processo judicial, administrativo ou arbitral, esse último nos termos da Lei nº 9.307, de 23 de setembro de 1996 (Lei de Arbitragem);

VII - para a proteção da vida ou da incolumidade física do titular ou de terceiro;

VIII - para a tutela da saúde, exclusivamente, em procedimento realizado por profissionais de saúde, serviços de saúde ou autoridade sanitária;

IX - quando necessário para atender aos interesses legítimos do controlador ou de terceiro, exceto no caso de prevalecerem direitos e liberdades fundamentais do titular que exijam a proteção dos dados pessoais.

Tais considerações servem como base para o tratamento adequado das informações a respeito dos pacientes durante a realização do exame, dentro dos parâmetros da lei [23].

2.10 Bancos de Dados

O Banco de Dados *Long Term ST Database* foi desenvolvido graças ao esforço interinstitucional e internacional, coordenado pelo Prof. Franc Jager, da Faculdade de Ciência da Computação e Informação da Universidade de Ljubljana, da Eslovênia. O desenvolvimento do banco de dados teve início em 1995 e foi finalizado em 2002, e está disponível publicamente para o estudo de pesquisadores [24].

O Banco de Dados de ST (do segmento ST do sinal) de Longo Prazo contém 86 registros longos de ECG de 80 indivíduos humanos, selecionados para exibir uma variedade de eventos de alterações do segmento ST, incluindo episódios isquêmicos de ST, episódios não isquêmicos de ST relacionados ao eixo, episódios de desvio lento do nível de ST e episódios contendo misturas desses fenômenos. O banco de dados foi criado para apoiar o

desenvolvimento e a avaliação de algoritmos capazes de diferenciar com precisão eventos de ST isquêmicos e não isquêmicos, bem como a pesquisa básica sobre os mecanismos e a dinâmica da isquemia miocárdica [24].

Metade (43) dessas 86 gravações, representando 42 dos 80 indivíduos, foram disponibilizadas ao PhysioNet pelos criadores do banco de dados em fevereiro de 2003, e a metade restante do banco de dados foi disponibilizada em maio de 2007. (Uma versão corrigida do arquivo s30801.dat também foi publicada juntamente com a segunda metade do banco de dados.) Notas clínicas detalhadas e gráficos de tendência do desvio-ST são fornecidos para todos os 86 registros.

As gravações individuais do Banco de Dados de ST de Longo Prazo têm duração entre 21 e 24 horas e contêm dois ou três sinais de ECG. Cada sinal de ECG foi digitalizado a 250 amostras por segundo com resolução de 12 bits em uma faixa de ± 10 milivolts.

Cada registro inclui um conjunto de episódios de ST e anotações de qualidade de sinal meticulosamente verificados, juntamente com anotações adicionais de QRS batimento a batimento e medições de nível de ST [24].

Para cada registro, o primeiro dígito no nome do registro (2 ou 3) indica o número de sinais de ECG. Registros obtidos do mesmo indivíduo têm nomes que diferem apenas no último dígito [24].

Os arquivos .hea são um pequeno arquivo de texto, trazendo informações sobre o sinal de ECG, como nome do arquivo de sinal, número de canais, frequência e número de amostras, dados de calibração, características do digitalizador e duração do registro [24].

Os arquivos .atr são arquivos de anotação, contendo dados sobre eventos importantes no sinal, como batimentos anormais e arritmias em um momento específico do registro [24].

Por fim, os arquivos .dat trazem informações específicas sobre o tipo de arquivo (texto, vídeo, binário etc.). No caso, os arquivos estavam em binário. Além de anotações contendo o diagnóstico, idade, uso de medicamentos, histórico, tratamento e comentários sobre cada coleta.

Com base nessas considerações, o presente trabalho propõe o desenvolvimento de um sistema capaz de realizar a transmissão em tempo real de sinais de ECG, bem como seu processamento e transmissão para a nuvem, além de realizar o armazenamento seguro dos dados, garantindo as proteções necessárias.

CAPÍTULO 3 - METODOLOGIA

Inicialmente, previa-se para este trabalho o projeto e a construção de um equipamento *Holter* conectado à internet para envio de dados à repositórios na nuvem, com *backup* local. Findo o projeto, o equipamento deveria ser testado com voluntários para validação do mesmo. No entanto, tal validação requer aprovação da pesquisa por Comitê de Ética em Pesquisas (CEP), o que demandaria um tempo considerável, além daquele factível para a realização deste Trabalho de Conclusão de Curso. Assim, decidiu-se pela seguinte estratégia, que balizará as descrições da metodologia de trabalho descritas a seguir, bem como os resultados apontados no capítulo seguinte:

A – Projeto, construção e validação em bancada de um dispositivo *Holter*;

B – Projeto, implementação e validação do sistema de *software* embarcado para envio de dados à nuvem, com *backup* local e software de leitura de dados da nuvem. Esta etapa fará uso de um banco de dados de sinais ECG-*Holter*.

3.1 Projeto, construção e validação em bancada de um dispositivo *Holter*

3.1.1 Materiais e métodos

Os materiais utilizados para a construção do protótipo foram:

- Protoboard;
- 2 resistores de 3,3 k Ω ;
- Microcontrolador ESP32;
- Módulo Sensor Cardíaco AD8232;
- Módulo Cartão Micro SD Card;
- Cartão de memória;
- Cabo USB-C;
- Jumpers.

3.1.2 Protótipo

O hardware para aquisição de sinais foi baseado no Módulo Sensor Cardíaco AD8232, detalhado na Figura 9. Este dispositivo é um módulo de condicionamento de sinal projetado para captar e processar sinais de ECG. Para fins de teste e validação neste projeto, o sensor foi utilizado para receber um sinal simulado proveniente do gerador de funções, em vez de eletrodos biológicos.

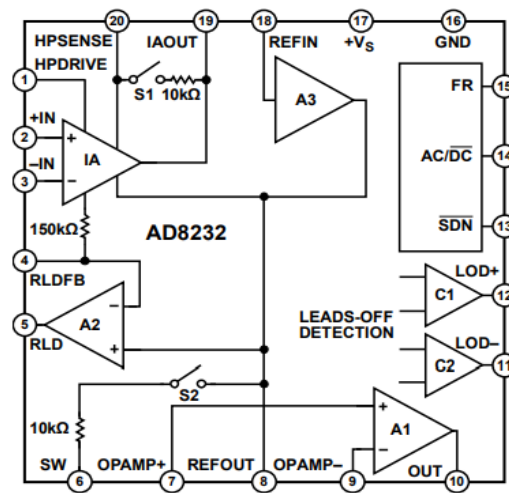


Figura 9: Diagrama de blocos funcional do módulo AD8232.Fonte: [25].

O sensor apresenta as seguintes configurações, de acordo com informações descritas no seu *Datasheet* [25]:

- filtro de dois polos passa-alta com frequência de corte de 0,5 Hz, para atenuar frequências abaixo do limiar e eliminar principalmente as componentes de baixa frequência do sinal, que podem ser ruído lento ou artefatos de movimento;
- filtro de dois polos passa-baixa com frequência de corte de 40 Hz, que elimina principalmente as componentes de alta frequência do sinal, que podem ser ruídos elétricos ou artefatos de alta frequência;
- filtro notch com frequência de corte de 60 Hz, aplicado para remover a interferência da rede elétrica (que no Brasil é da ordem de 60 Hz) e para suavizar o sinal;
- ganho fixo de 100.

Os sinais de saída do módulo AD8232 que são digitalizados em um módulo ESP32. Os arquivos de dados são então formatados, armazenados em um cartão SD local e, em seguida, enviados para a nuvem para armazenamento em uma planilha. As informações são enviadas para a planilha por meio de uma conexão internet, onde as informações são enviadas em tempo real, e somente aqueles usuários com acesso à pasta podem acessá-las e visualizá-las.

Outra questão que se mostrou importante durante o desenvolvimento do projeto foi o armazenamento dos dados coletados, em caso de interrupção do sistema de rede de *Internet* ou da falta do mesmo. Sendo assim, um cartão de memória SD foi adicionado ao circuito, visando armazenar as informações.

A coleta foi dividida em duas etapas: uma coleta direto de oito horas, com uma tensão de 100 mV aplicada continuamente e armazenada no cartão de memória SD, a uma frequência de 150 Hz, via USB, e uma coleta de 4 horas via Wi-Fi, com as mesmas tensão e frequência e com a conexão interrompida propositalmente e realizando o armazenamento simultâneo no cartão de memória.

Procedimento para geração de sinal sintético

Uma vez que não seria possível utilizar um sinal fisiológico para validar o protótipo, optou-se por utilizar sinais provenientes de gerador de funções, como detalhado na Figura 10. O dispositivo deveria operar por um período de oito horas ininterruptas. Essa fase tinha como principal objetivo avaliar a capacidade do circuito de receber os sinais provenientes do gerador de funções, testando a funcionalidade de envio das informações do *hardware* para a planilha com conexão *Wi-Fi* e com a conexão interrompida propositalmente, tendo sido esse período suficiente para observar tais eventos.

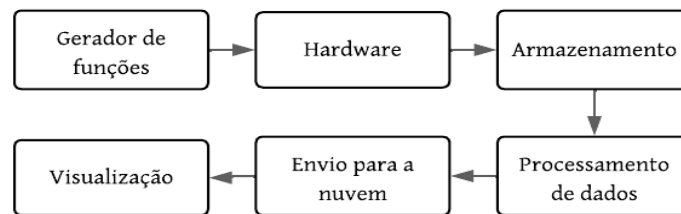


Figura 10: Diagrama de blocos das unidades gerais do sistema e sua validação com sinais de geradores de funções. Fonte: Autora.

Como a tensão mínima de saída do gerador de funções é de 200 mV e a tensão desejada para o circuito era de 100 mV, foi utilizado um divisor de tensão (Figura 11), a fim de dividir a tensão por 2, conforme a Equação descrita em (I). Assim, foi possível atingir a tensão desejada para alimentação do protótipo.

$$\begin{aligned} V_{out} &= V_{in} \times \left(\frac{R_2}{R_2 + R_1} \right) \quad (I) \\ V_{out} &= 200 \times \left(\frac{3,3k}{3,3k + 3,3k} \right) \\ V_{out} &= 200 \times \left(\frac{3,3k}{6,6k} \right) \\ V_{out} &= 200 \times (0,5) \\ V_{out} &= 100 \text{ mV}. \end{aligned}$$

Sendo:

V_{out} é a tensão de saída;

V_{in} é a tensão de entrada;

E R_2 e R_1 são, respectivamente, os valores dos resistores.

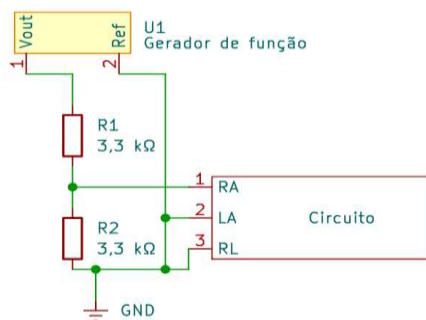


Figura 11: Divisor de tensão integrado ao circuito. Fonte: Autora.

A Figura 12 mostra o diagrama completo do circuito eletrônico do equipamento e a Figura 13 o protótipo montado. Os dados são armazenados no cartão SD em arquivos no formato binários.

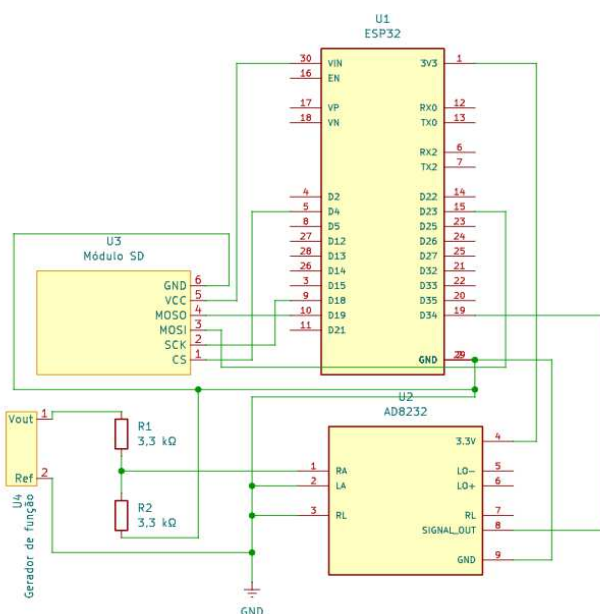


Figura 12: Esquemático do circuito eletrônico do protótipo. Fonte: Autora.

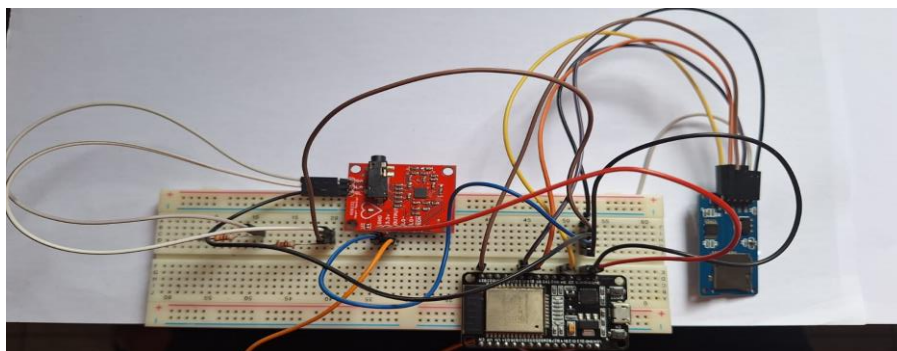


Figura 13: O Hardware do protótipo montado em protoboard. Fonte: Autora.

O gerador de funções foi configurado para gerar uma onda senoidal com amplitude de 100 mV e frequência de 150 Hz. Para fins de validação do protótipo, visando realizar um estresse de carga de dados, utilizou-se um sinal sintético de 150 Hz. Neste cenário, evidencia-se a necessidade de ajustes na frequência de aquisição ao operar com sinais de maior largura

de banda, a fim de respeitar o Teorema de Nyquist, que preconiza uma frequência de amostragem de, no mínimo, o dobro da frequência analisada [13]. Entretanto, como o escopo desta etapa focou na validação da infraestrutura de comunicação, focando especificamente na capacidade de recepção e envio de dados para a nuvem dentro do período estipulado, a taxa de amostragem empregada mostrou-se suficiente para comprovar a viabilidade do fluxo de informações do sistema



Figura 14: Gerador de funções utilizado como sinal sintético para testes do protótipo. Fonte: Autora.

Para validação da coleta do sinal, inicialmente, a interface enviava uma mensagem de “Dado gravado” a cada gravação, o que acabava ocupando uma memória maior e deixando o sistema mais lento. Pensando em uma forma de indicar que o *hardware* estava recebendo os sinais e que o cartão SD havia sido reconhecido e estava gravando e armazenando as informações, foi feita uma configuração para exibir uma luz azul, que indicava que o sinal estava sendo recebido sem problemas.

Desta forma, um código foi escrito na linguagem de programação C++, que possui uma estrutura de bibliotecas específicas para os requisitos do sistema, através do ambiente de desenvolvimento integrado (IDE) Arduino.

Quanto ao sistema de *hardware* para a coleta, o circuito, descrito na Figura 7, foi alimentado via conexão USB com uma tensão proveniente de um *notebook*, e montado em uma *proto board*.

A alimentação proveniente do cabo USB foi conectada no pino de 3,3 V do ESP32, que, por sua vez, foi conectado ao pino de 3,3 V do sensor. O pino D34, um dos pinos conversor analógico-digital (*ADC - Analog-to-Digital Converter*) do microcontrolador e ideal para sinais analógicos como os provenientes do coração, foi conectado à saída *output* do sensor e os pinos de referência terra, ou ground (GND), foram conectados para estabelecer uma referência comum.

Simultaneamente, os dados foram enviados para o Visual Studio Code (*VS Code*), onde foram processados usando scripts *Python*. O primeiro passo foi converter os dados gravados para os formatos necessários para análise, assim como o formato dos arquivos do banco de dados: .dat, .hea e .atr. Nesse formato, os arquivos .dat contêm dados brutos do sinal de ECG, armazenando as amostras digitalizadas do sinal, uma vez que os arquivos podem ser muito grandes.

3.2 Procedimento para validação com auxílio de sinais *Holter* em banco de dados

Esta fase do projeto visa validar a robustez do sistema em condições de operação prolongada, simulando o monitoramento contínuo de oito a 24 horas. O objetivo é garantir que a infraestrutura de transmissão e armazenamento em nuvem suporte o fluxo ininterrupto de uma elevada densidade de dados/registros sem perda de integridade. Para tanto, foram utilizados sinais do banco de dados público *Long Term ST Database* (Figura 15), escolhido por oferecer registros reais e extensos, amplamente validados pela literatura científica para o estudo de arritmias [24]. Essa etapa é crucial para assegurar que o protótipo seja capaz de identificar eventos cardíacos transitórios que um exame de ECG convencional de curta duração poderia omitir.

s20041.atr	246.3 KB	2002-05-20
s20041.cnt	4.3 KB	2002-05-20
s20041.dat	61.6 MB	2002-05-20
s20041.heg	2.3 KB	2008-04-05
s20041.heg-	2.3 KB	2002-05-20
s20031.atr	229.3 KB	2002-05-20
s20031.cnt	4.3 KB	2002-05-20
s20031.dat	61.4 MB	2002-05-20
s20031.heg	1.7 KB	2008-04-05
s20031.heg-	1.6 KB	2002-05-20
s20051.atr	276.0 KB	2019-04-02
s20051.cnt	4.2 KB	2019-04-02
s20051.dat	54.9 MB	2019-04-02
s20051.heg	1.5 KB	2010-09-13
s20051.heg-	1.5 KB	2019-04-02

Figura 15: Exibição dos arquivos disponíveis no Banco de Dados. Fonte: Autora.

Para emular corretamente um sistema *Holter*, os sinais do banco de dados foram coletados a uma taxa de 250Hz (taxa de amostragem original) e tratados por um software como se estivessem sendo coletados em tempo real do hardware desenvolvido (descrito anteriormente). A Figura 16 destaca as unidades do sistema, desenvolvido em linguagem Python.

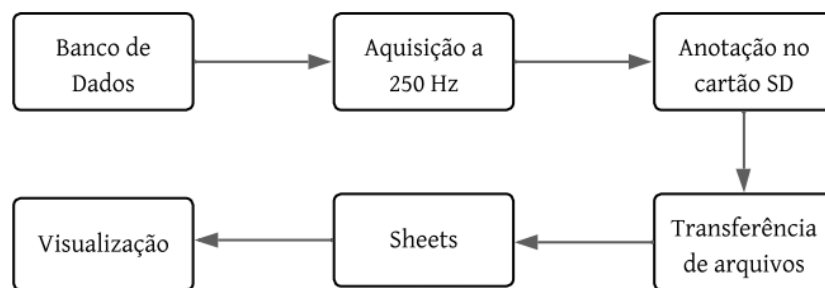


Figura 16: Diagrama de blocos das unidades de software desenvolvidas para o sistema *Holter* e emulação de funcionamento com sinais de bancos de dados. Fonte: Autora.

Dentre os diversos sinais componente no banco de dados, foram escolhidas, aleatoriamente, as amostras s20051 e s20061. Estes sinais foram coletados a uma taxa de amostragem de 250 Hz, com resolução de 12 bits e amplitudes entre ± 10 mV. Os sinais armazenados no banco de dados já se encontravam processados, tendo sido devidamente

filtrados. O tempo de gravação de cada amostra do banco foi de, respectivamente, 21 horas e 18 minutos, e 22 horas e 45 minutos. Os registros selecionados apresentam volumes distintos de dados: enquanto o sinal s20051 totaliza 19.181.000 amostras, o registro s20061 compreende 20.474.500 amostras, refletindo a extensão temporal das coletas no banco de dados.

Para viabilizar a compatibilidade estrutural e a integridade na leitura dos dados, utilizou-se a conversão da estrutura de dados original para um vetor unidimensional por meio da extração da coluna de amplitudes em um objeto do tipo *Series*. Esse procedimento eliminou a complexidade de estruturas multidimensionais, permitindo que o sistema acessasse cada amostra de forma sequencial e contínua através de um ponteiro de iteração. Tal abordagem garantiu que o sinal fosse processado e plotado como uma linha temporal única, eliminando inconsistências de dimensionalidade e assegurando a precisão na indexação temporal do sinal de ECG.

Foi utilizado um *Buffer* circular de tamanho fixo ($n=2.500$), que armazena os sinais de forma contínua utilizando aritmética modular para sobrescrever os registros mais antigos. Esse processo foi gerenciado por uma *thread* secundária dedicada, operando de forma assíncrona ao fluxo principal do programa. O uso de *threads* permitiu a aquisição de dados em tempo real com frequência de amostragem constante de 250 Hz, enquanto mecanismos de exclusão mútua (locks) garantiam a integridade dos dados, impedindo conflitos de leitura e escrita simultânea na memória.

CAPÍTULO 4 - RESULTADOS E DISCUSSÃO

4.1 Testes do protótipo com sinais sintéticos

O hardware construído foi testado com apoio de um gerador de funções, conforme descrito no capítulo anterior. Os testes de coleta e transmissão de dados para a nuvem e para armazenamento local (cartão SD) ocorreram de forma ininterrupta por um período de quatro horas. Para teste do sistema de backup, a comunicação Wi-Fi do equipamento foi interrompida aleatoriamente. Foi utilizada uma rede de *Internet* particular, sendo fornecida a partir de um smartphone. Com a conexão interrompida, o sistema buscava gravar as informações recebidas no cartão SD, voltando a enviar os sinais em tempo real assim que a conexão era restabelecida. As Figuras 17 e 18 mostram os dados gravados no cartão SD (formato binário) e no arquivo em nuvem (formato texto), respectivamente, com o formato em texto contendo os valores ADC dos dados. Houve também uma coleta ininterrupta por um período de oito horas, com a conexão estável o tempo todo. É importante destacar que o modelo de backup construído desempenhou satisfatoriamente, impedindo a perda de dados quando a internet foi interrompida – durante a interrupção, os dados continuavam a ser gravados em memória local e, com o retorno da conexão internet, os dados gravados localmente voltavam a ser transferidos para a nuvem.

```
00101011 00000111 00110101 00000111 00011111 00000111 00010111 00000111 00011010 0000
111 00110110 00000111 00110001 00000111 00110111 00000111 00101110 00000111 00100001 0
0000111 00011101 00000111 00000111 00000111 00000101 00000111 00010000 00000111 000111
00 00000111 00011101 00000111 00100011 00000111 00101010 00000111 01000110 00000111 00
101001 00000111 00101111 00000111 00100110 00000111 00001100 00000111 00000000 0000011
1 00010000 00000111 00010101 00000111 00110010 00000111 00011011 00000111 00010100 000
00111 00011010 00000111 00001110 00000111 00001010 00000111 00011111 00000111 00110000
00000111 00011011 00000111 00001101 00000111 00010111 00000111 00101001 00000111 0011
0010 00000111 00110101 00000111 00111101 00000111 00100011 00000111 00011101 00000111
00001101 00000111 00000110 00000111 00110001 00000111 00110010 00000111 00100001 00000
111 00001111 00000111 00001110 00000111
```

Figura 17: Trecho de dados armazenados em formato binário. Fonte: Autora.

1	Amostra	Valor_ADC
2	1	1809
3	2	1819
4	3	1808
5	4	1795
6	5	1796
7	6	1831
8	7	1809
9	8	1802
10	9	1809
11	10	1825
12	11	1821
13	12	1771
14	13	1814
15	14	1823
16	15	1823
17	16	1802
18	17	1795

Figura 18: Trecho de dados armazenados em formato texto, contendo o valor ADC correspondente dos números em binário. Fonte: Autora.

4.2 Validação do com sinais *Holter* de banco de dados

A Figura 19 mostra um exemplo de sinais ECG provenientes do banco de dados, abertos e plotados pelo aplicativo Python desenvolvido.

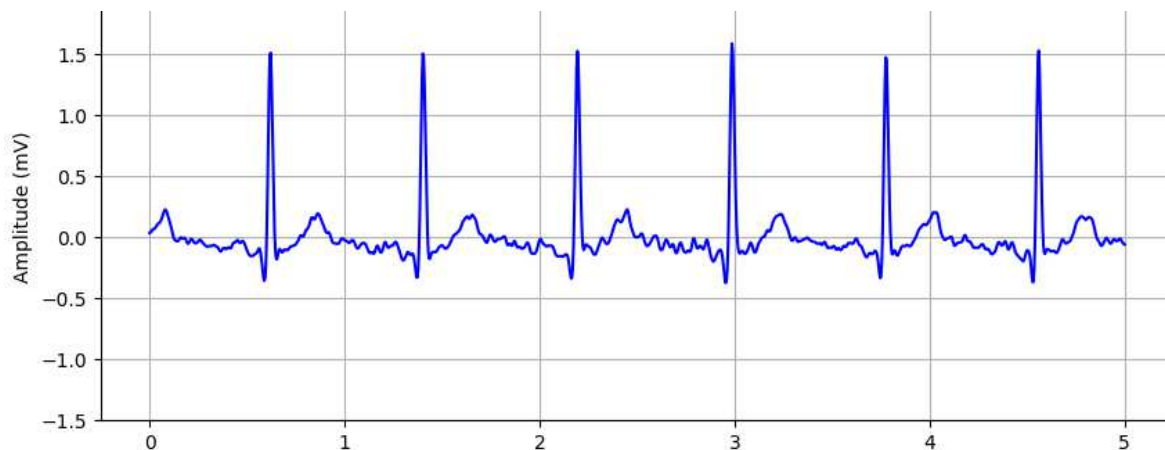


Figura 19: Formato de onda do ECG de uma das amostras. Fonte: Autora.

A título de exemplo de tratamento e processamento destes dados, a Figura 20 mostra a evolução da frequência cardíaca do sujeito (Equação II), durante um período de 45 minutos.:

$$bpm = \frac{n^{\circ} \text{ de picos de onda R na janela de tempo}}{\text{duração da janela de tempo em segundos}} \times 60 \text{ (II)}$$

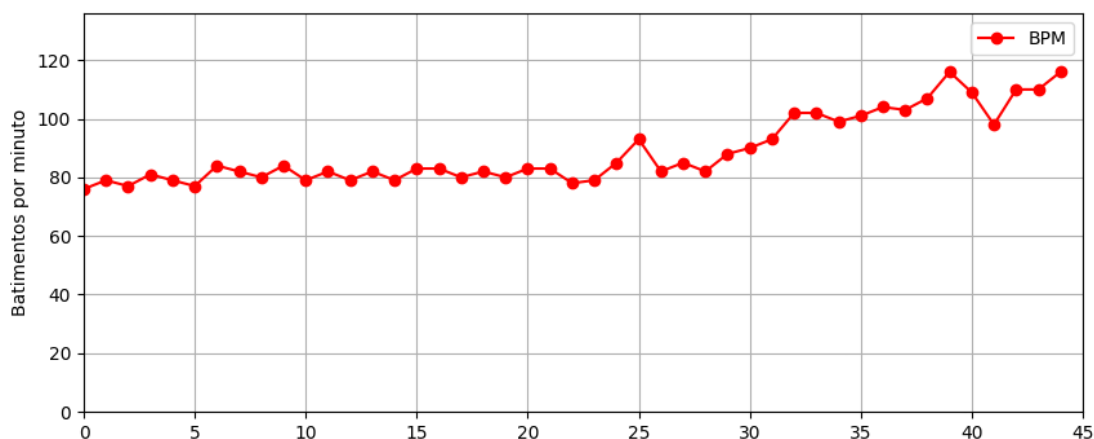


Figura 20: Gráfico de bpm no período de 45 minutos. Fonte: Autora.

Conforme descrito anteriormente, duas unidades de software compõem o sistema:

- Software embarcado na unidade de hardware para coleta de dados e envio para nuvem com redundância local (cartão SD);
- Aplicativo de operação, acessível pelos profissionais da saúde, que descarrega e processa os dados dos arquivos em nuvem.

Software embarcado

O arquivo gravado no cartão de memória incluiu o registro de 60 minutos de análise do sinal, a fim de obter um montante de dados que garantisse a validade estatística da análise de integridade. Uma vez que a frequência de amostragem era de 250 Hz, podemos calcular o número de amostras a partir da fórmula descrita na Equação III.

$$n^o = Fs \times t \text{ (III)}$$

$$n^o = 250 \times (3.600)$$

$$n^o = 900.000 \text{ amostras}$$

Sendo:

Fs = frequência de amostragem;

t = tempo (em segundos).

As 900.000 amostras ocuparam um espaço de 14,7 MB, ou 0,086% dos 16 GB do cartão. Em seguida, o processo tem início com o sinal bruto de eletrocardiograma passando por uma etapa de filtragem digital (passa-banda 0,5-40 Hz) para minimização de ruídos e artefatos, como mostra a Figura 21.

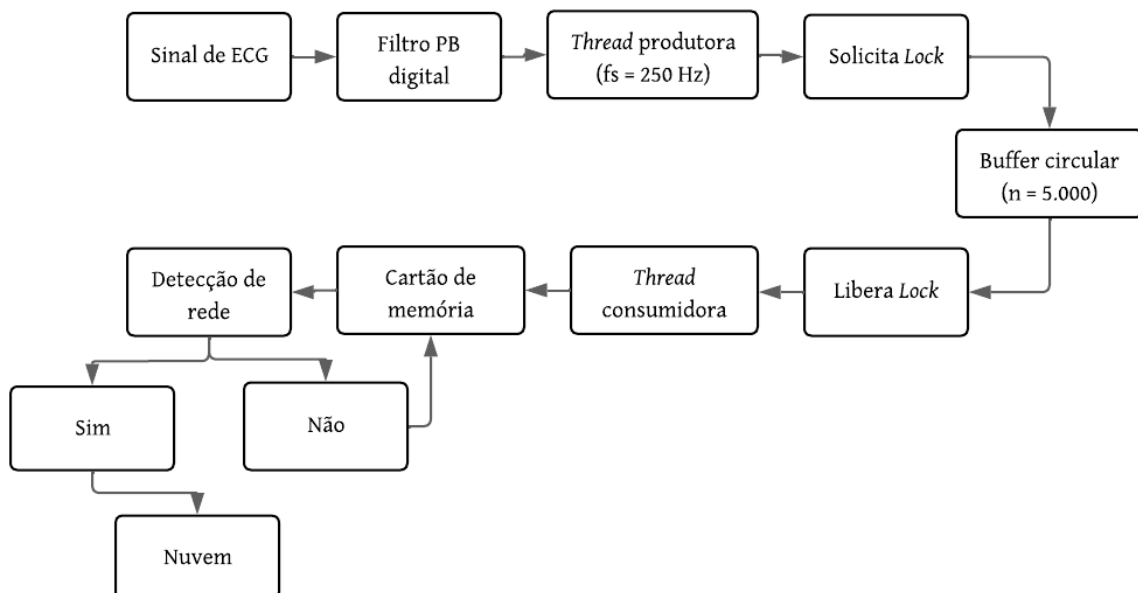


Figura 21: Diagrama de blocos da lógica do sistema. Fonte: Autora.

Para realizar as funções de leitura, armazenamento de dados e envio para nuvem, optou-se por implementar aquelas funções em concorrência. Um *buffer* circular com capacidade para cinco mil amostras é utilizado como repositório inicial de dados provenientes da unidade de leitura/aquisição de dados. Para implementação das unidades de concorrência, o sistema emprega *threads* de execução paralela. Nesse arranjo, uma *thread* produtora é responsável por coletar as informações do banco de dados na mesma taxa que estaria sendo coletada de um sistema de coleta em hardware ($f_s=250$ Hz) e os armazena no buffer circular. Paralelamente,

uma *thread* consumidora acessa o buffer circular, armazena os dados no cartão SD e os envia para a nuvem, caso a internet esteja disponível.

Para garantir a integridade dos dados durante esse processo simultâneo, utiliza-se um mecanismo de exclusão mútua baseado em *threading.Lock()*. Esse recurso de sincronização evita a chamada condição de corrida, assegurando que as operações ocorram de forma atômica, ou seja, impedindo que a *thread* consumidora leia um dado enquanto ele ainda está sendo parcialmente escrito pela *thread* produtora. A Figura 22 ilustra um exemplo do sinal de ECG reconstruído a partir das amostras gravadas no cartão de memória.

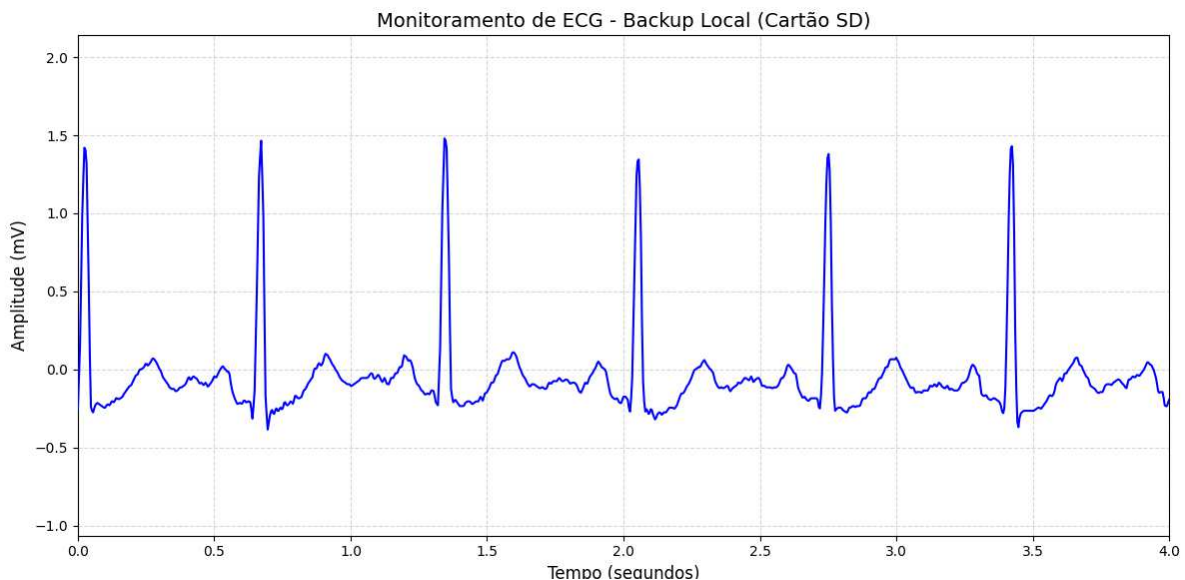


Figura 22: Reconstrução do ECG a partir de dados gravados no cartão SD local. Fonte: Autora.

Aplicativo de operação

Para validar a fidelidade do sistema de monitoramento, foi realizado um procedimento de auditoria automatizada que confronta o banco de dados local, armazenado no cartão SD, com o repositório em nuvem através de um *script* de paridade em *Python*. Esse processo consistiu na leitura simultânea de ambos os registros, seguida de uma normalização dos caracteres e separadores decimais para garantir a compatibilidade entre os diferentes ambientes de armazenamento. A análise realizou a contagem das amostras e o cruzamento ponto a ponto de cada valor de amplitude registrado, permitindo verificar a integridade da camada de aplicação e confirmar que a transmissão via rede sem fio ocorreu sem perda de pacotes ou corrupção de informações, como mostra a Figura 23.

```
=====
Comparativo de amostras (cartão de memória e nuvem)
=====
Amostras no cartão de memória: 900000
Amostras na Nuvem:           900000
Divergências de Valor:       0
-----
Status: 100% de compatibilidade entre as amostras
=====
```

Figura 23: Comparação de compatibilidade entre os arquivos local e na nuvem. Fonte: Autora.

Ao realizar o envio dos dados para uma planilha online, o sistema permite que o monitoramento possa ser realizado de forma remota, em diferentes localidades. Médicos, pesquisadores e pacientes podem acessar os dados a partir de qualquer dispositivo com acesso à *Internet*, sem necessidade de possuir os scripts em *Python* ou softwares instalados. Para o envio e exibição das informações na nuvem, foi escolhido o *Google Sheets*.

Essa plataforma oferece ferramentas de visualização que se atualizam automaticamente, em tempo real, criando um *dashboard* de baixo custo, simples e de alta eficiência para o acompanhamento das informações desejadas. Outra decisão técnica estratégica foi a separação entre o armazenamento e a visualização gráfica. Embora o sistema seja capaz de gerar gráficos interativos com os formatos de onda do ECG, na nuvem, os dados na planilha são salvos estritamente no formato de texto, registrando a amplitude de cada amostra para tornar possível a construção posterior do formato de onda.

É importante ressaltar que o acesso à planilha não é público. Somente o *script* autorizado consegue inserir dados e as informações processadas estão disponíveis apenas para usuários que possuam o *link* de acesso e as devidas permissões de visualização cedidas pelo administrador. Essas configurações permitem que o compartilhamento seja feito de forma seletiva entre a equipe médica e os usuários, respeitando normas de sigilo de dados de saúde.

Para a apresentação dos resultados e das interfaces de divulgação do projeto, foi aplicado o princípio do anonimato de dados sensíveis, conforme as diretrizes da LGPD. Todas as informações de identificação pessoal (como nome completo, detalhes de documentação, localização, dados de contato, entre outros) foram omitidas. Esse procedimento visa garantir que a demonstração técnica da arquitetura do sistema ocorra sem a exposição desnecessária de dados privados, mantendo o foco estritamente na validação da aplicação e no seu impacto tecnológico.

4.3 Visualização do sinal online

Esta unidade permite visualizar parâmetros gerais do sinal lido da nuvem, como, por exemplo: a forma de onda, a taxa de batimentos por minuto e eventos como taquicardia e bradicardia. Para tal, os dados foram extraídos da planilha online onde estavam armazenados e processados para exibir o formato de onda no sistema. Isso foi possível graças à biblioteca *Streamlit*, uma biblioteca de código aberto em *Python* que permite transformar *scripts* de dados em aplicativos *Web* interativos, de forma rápida e fácil.

O aplicativo foi armazenado na plataforma *GitHub* (<https://parte-dois-exkhztyxt7nn3lvyjf8vr.streamlit.app/>) permitindo assim que o mesmo esteja hospedado também na nuvem facilitando seu acesso (Figura 24).

Exibição do ECG em tempo real

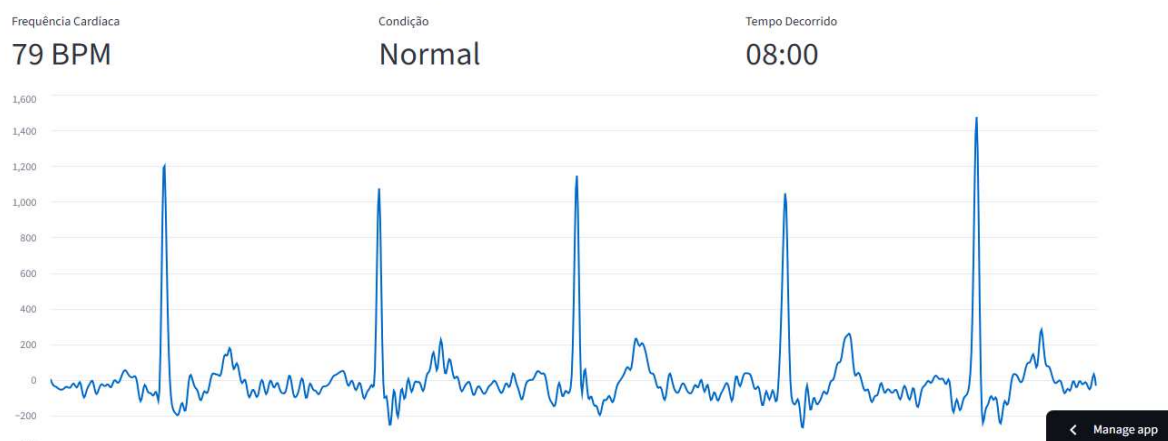


Figura 24: Interface do sistema para a visão de profissionais de saúde. Fonte: Autora.

Por fim, destaca-se que, as informações também podem ser acessadas via *smartphone*, utilizando a aplicativo acessível pelo link descrito acima.

CAPÍTULO 5 – CONCLUSÃO

Este trabalho dividiu-se em duas etapas complementares: a construção de um protótipo capaz de realizar a aquisição, leitura e armazenamento de um sinal, e o desenvolvimento de um *software* capaz de realizar desde o processamento das informações até sua disponibilidade na nuvem, de forma simples e de baixo custo.

O sistema foi projetado para priorizar a escrita local no cartão SD, assegurando que nenhum dado fisiológico seja perdido caso a conexão com a nuvem seja interrompida. O uso de *locks* garante que a tarefa de aquisição de sinais não seja bloqueada pela tarefa de envio, dessa forma, em situações de instabilidade, o envio para a nuvem é temporariamente retardado em favor da continuidade da gravação e da visualização em tempo real, evitando o congelamento da interface e garantindo a robustez do monitoramento contínuo.

O protótipo mostrou-se capaz de fazer a aquisição do sinal simulado, enviar as informações via *Wi-Fi* e *USB*, gravar as informações em um cartão de memória e receber informações por um período de até oito horas. O *software* mostrou-se eficaz no processamento dos sinais, na exibição das informações e no envio, de forma simples, para um sistema na nuvem, onde, mesmo online, tem configurações privadas de acesso.

O sistema mostrou ser possível ver índices cruciais para a avaliação da saúde cardíaca, como o número de batimentos cardíacos por minuto, formato de onda e informações sobre bradicardia e taquicardia de maneira acessível, podendo ser acessados por profissionais de saúde de diferentes regiões, diminuindo a barreira geográfica.

Entre as principais vantagens da solução proposta destacam-se o seu baixo custo de implementação, devido ao uso de ferramentas acessíveis e APIs gratuitas, a confiabilidade proporcionada pela combinação de armazenamento em nuvem, com *backup* local, que cria uma dupla camada de segurança para os dados, e a modularidade do sistema, que permite sua adaptação para o monitoramento de parâmetros biométricos, como o ECG.

Apesar dos resultados satisfatórios, o sistema apresenta limitações que devem ser consideradas para aplicações em larga escala. Primeiramente, uma taxa de amostragem adequada para um sinal cardíaco real, conforme o Teorema de Nyquist. Além disso, o uso do *Google Sheets* como banco de dados, apesar de simples e de baixo custo, não poderia ser utilizado para um sinal de 24 horas de duração, visto que a plataforma possui um limite de dez milhões de linhas. Para um monitoramento de 24 horas com uma taxa de amostragem de 250 Hz, seriam necessários 21,6 milhões de registros por canal, o que excede a capacidade da ferramenta atual e exige a migração para bancos de dados com maior volume, como NoSQL, em implementações futuras. Por fim, ressalta-se que as validações foram realizadas exclusivamente com sinais sintéticos e bases de dados públicos, carecendo de testes com voluntários humanos para avaliar a influência de ruídos biológicos e artefatos de movimento na precisão do algoritmo.

O aprimoramento do sistema em trabalhos futuros deve incluir: a melhoria do hardware para coleta de sinais ECG em 12 derivações, o desenvolvimento de um sistema de alimentação por bateria, tornando o sistema portátil, melhorias no aplicativo de uso para avaliação de outros eventos importantes, como possíveis sinais de isquemia, uma taxa de amostragem adequada, uma alternativa de banco de dados com maior capacidade, e, por fim, uma validação com voluntários humanos.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] BRAUNWALD, Eugene. **Heart Disease: A Textbook of Cardiovascular Medicine**. 6. ed. [S. I.]: W. B. Saunders Company, 2001. Disponível em: <http://www.arritmiaonline.com.br/files/Braunwald---Heart-Disease---A-Textbook-of-Cardiovascular-Medicine-6th-ed.pdf>. Acesso em: 06 ago. 2025.
- [2] SOCIEDADE BRASILEIRA DE CARDIOLOGIA. **Cardiômetro: mortes por doenças cardiovasculares no Brasil**. Rio de Janeiro: SBC; 2025. Disponível em: <http://www.cardiometro.com.br/>. Acesso em: 06 ago. 2025.
- [3] MAGALHÃES, M. I. S., *et al.* Desafios diagnósticos na identificação precoce de doenças cardíacas: avanços e perspectivas. **Revista Ibero-Americana De Humanidades, Ciências E Educação**, [s. I.], v. 10, n. 1, p. 1531–1538. Disponível em: <https://doi.org/10.51891/rease.v10i1.13093>. Acesso em: 06 ago. 2025.
- [4] BRASIL. Ministério da Saúde. **Informações de Saúde – TabNet**. Brasília, DF: Ministério da Saúde, [2025]. Disponível em: <http://tabnet.datasus.gov.br/cgi/tabcgi.exe?sim/cnv/obt10uf.def>. Acesso em 12 set. 2025.
- [5] FYE W. B. A history of the origin, evolution, and impact of electrocardiography. **Am J Cardiol**, [s. I.], v. 73, n.13, p. 937-949, 15 maio 1994. doi:10.1016/0002-9149(94)90135-X.
- [6] GRUNE, J., YAMAZOE, M.; NAHRENDORF, M. Electroimmunology and cardiac arrhythmia. **Nat Rev Cardiol**, [s. I.], v. 18, p. 547–564, 2021. Disponível em: <https://doi.org/10.1038/s41569-021-00520-9>. Acesso em 12 set. 2025.
- [7] SABER, Mohamed; ABOTALEB, Mostafa. Arrhythmia modern classification techniques: A review. **J. Artif. Intell. Metaheuristics**, [s. I.], v. 1, p. 42-53, 2022.
- [8] MORAIS, Danielle; LINS, Aretha; CAMPOS, Danielle. **O coração pede cuidado: por que as doenças cardiovasculares preocupam**. Brasília, DF: EBSEH, 1 set. 2025. Disponível em: <https://www.gov.br/ebserh/pt-br/hospitais-universitarios/regiao-norte/hdt-uf/comunicacao/noticias/o-coracao-pede-cuidado-por-que-as-doencas-cardiovasculares-preocupam>. Acesso em: 12 set. 2025.
- [9] BRASIL. Ministério da Saúde. **Viva o SUS! O maior sistema público de saúde do mundo é gratuito, universal e do Brasil**. Brasília, DF: Ministério da Saúde, 19 set. 2025. Disponível em: <https://www.gov.br/saude/pt-br/assuntos/noticias/2025/setembro/viva-o-sus-o-maior-sistema-publico-de-saude-do-mundo-e-gratuito-universal-e-do-brasil>. Acesso em: 12 set. 2025.

- [10] BRASIL. Ministério da Saúde. **Agora tem Especialistas**. Brasília, DF: Ministério da Saúde, 2025. Disponível em: <https://www.gov.br/saude/pt-br/campanhas-da-saude/2025/agora-tem-especialistas>. Acesso em: 12 set. 2025.
- [11] JOHNS HOPKINS MEDICINE. **Holter Monitor**. Baltimore, MD: Johns Hopkins Medicine, 2024. Disponível em: <https://www.hopkinsmedicine.org/health/treatment-tests-and-therapies/holter-monitor>. Acesso em: 15 set. 2025.
- [12] AIRES, Margarida de Mello. **Fisiologia**. 4. ed. Rio de Janeiro: Guanabara Koogan, 2013.
- [13] CARDOSO, Adriano Silva Vale. **Instrumentação e metodologias de medição de biopotenciais**. 2010.
- [14] WEBSTER, John G. (Ed.). **Medical instrumentation: application and design**. [S. I.]: John Wiley & Sons, 2009.
- [15] KANDEL, E. R., SCHWARTZ, J. H., e JESSEL, T. M. **Principles of Neuroscience**. 4 ed. [S. I.]: McGraw-Hill, 2000.
- [16] BROWN, L. S. e HOLME, T. A. **Química Geral Aplicada à Engenharia**. [S. I.]: Cengage Learning, 2010.
- [17] DI DIEGO, José M. ECG Waves and Signs: Ionic and Cellular Basis. In: **Management of Cardiac Arrhythmias**. Cham: Springer International Publishing, 2020. p. 117-148.
- [18] MUBARIK, Ahsan; IQBAL, Amir M. **Holter Monitor**. Treasure Island (FL): StatPearls Publishing, 2025. Disponível em: <https://www.ncbi.nlm.nih.gov/books/NBK538203/>. Acesso em: 29 set. 2025.
- [19] GALLI A.; AMBROSINI F.; LOMBARDI, F. Holter Monitoring and Loop Recorders: From Research to Clinical Practice. **Arrhythm Electrophysiol Rev.**, [s. I.], v. 5, n.2, p.136-143, ago. 2016. doi: 10.15420/AER.2016.17.2.
- [20] PAPOUCHADO, M. *et al.* Fundamental differences between the standard 12-lead electrocardiograph and the modified (Mason-Likar) exercise lead system. **Eur Heart J.**, [s. I.], v. 8, n. 7, p. 725-733, jul. 1987.
- [21] WEN, Cheng *et al.* **Real-time ECG telemonitoring system design with mobile phone platform**. Measurement, [s. I.], v. 41, n. 4, p. 463-470, 2008.
- [22] KARAJAH, E. A. A.; ISHAQ, I. **Online Monitoring Health Station Using Arduino Mobile Connected to Cloud Service: “Heart Monitor” System**. In: International Conference on Promising Electronic Technologies (ICPET), IEEE, 2020. p. 38-43.

[23] BRASIL. **Lei nº 13.709, de 14 de agosto de 2018**. Dispõe sobre a proteção de dados pessoais e altera a Lei nº 12.965, de 23 de abril de 2014 (Marco Civil da Internet). Brasília, DF: Presidência da República, [2018]. Disponível em: https://www.planalto.gov.br/ccivil_03/_ato2015-2018/2018/lei/113709.htm. Acesso em: 5 out. 2025.

[24] JAGER, Franc; MOODY, George B.; MARK, Roger G. **Long-Term ST Database v1.0.0**. 2000. Disponível em: <https://physionet.org/content/ltstdb/1.0.0/>. Acesso em: 10 out. 2025.

[25] ANALOG DEVICES. **AD8232**: Single-Lead, Heart Rate Monitor Front End (Datasheet). Revision E. Wilmington, MA, 2020. Disponível em: <https://www.analog.com/media/en/technical-documentation/data-sheets/ad8232.pdf>. Acesso em: 19 fev. 2026.

APÊNDICE

Código em C++ para aquisição do sinal simulado.

```
#include <SD.h>           // Leitura e escrita no cartão SD
#include <WiFi.h>          // Conexão com Internet
#include <HTTPClient.h>
#include <deque>

//Configurações da rede de conexão Wi-Fi
const char* ssid = "xxxx";           // Inserir nome da rede
const char* password = "xxxx";       // Inserir senha de acesso
const char* serverUrl = "xxxx";      // Endereço para acessar recursos
do computador

//Definição da configuração do Hardware
const int chipSelect = 5; // Chip Select (CS) para o barramento SPI do
Cartão SD
const int ECG_PIN = 34; // O pino 34 pertence ao ADC1 do ESP32, o que
evita conflitos de hardware com o uso do Wi-Fi e garante que a
aquisição do sinal não sofra ruídos de RF.
const int LED_STATUS = 2; // O pino 2 por estar conectado ao LED
interno do ESP32

//Configurações de tempo e frequência
unsigned long tempoInicial;
const unsigned long DURACAO_GRAVACAO = 480 * 60 * 1000; // Colocar o
primeiro valor em minutos desejados para a gravação
const int FREQUENCIA = 150;          // Frequência de amostragem utilizada
const int INTERVALO_MS = 1000 / FREQUENCIA;

bool gravacaoConcluida = false;
bool sdOk = false;
File dataFile;

std::deque<int16_t> bufferRAM; // Fila de dupla extremidade (deque) que
atua como buffer para gerenciar a diferença de velocidade entre a
leitura do sensor e a latência de envio para a nuvem.
const int MAX_BUFFER = 3000; // Limite de 3000 amostras

//Definição de tarefas
void TaskColeta(void * pvParameters); // Responsável pela aquisição do
sinal via interrupção de tempo, garantindo a taxa de amostragem
constante de 250 Hz.
void TaskComunicacao(void * pvParameters); // Gerencia o processamento
paralelo para escrita no SD e envio via Wi-Fi sem bloquear a leitura do
sensor.
```

```

void setup() {
    Serial.begin(115200);
    pinMode(ECG_PIN, INPUT);
    pinMode(LED_STATUS, OUTPUT);

    //Configuração de inicialização do SPI e do cartão SD
    SPI.begin(18, 19, 23, chipSelect); // Inicializa o barramento VSPI do
    ESP32 definindo os pinos SCK(18), MISO(19) e MOSI(23) para garantir a
    integridade da comunicação com o SD.
    if (SD.begin(chipSelect)) { // Inicia o sistema de arquivos do cartão
    SD; se detectado, assegura a camada de backup local (offline) caso
    ocorra falha na conexão Wi-Fi.
        Serial.println("SD Detectado!");

        //Criação da pasta para armazenamento dos arquivos
        if (!SD.exists("/Gravacao_binaria")) {
            SD.mkdir("/Gravacao_binaria"); // Cria a pasta dedicada para os
            arquivos de dados brutos, organizando o backup local de forma isolada
            do sistema de arquivos.
        }

        //Abre o arquivo uma única vez no início
        dataFile = SD.open("/Gravacao_binaria/ecg_data.bin", FILE_WRITE);
        if(dataFile) {
            sdOk = true;
            Serial.println("Arquivo binário pronto para gravar.");
            digitalWrite(LED_STATUS, HIGH); // LED aceso fixo indica gravação
        }
    } else {
        Serial.println("Erro: Cartão SD não encontrado."); // Função para
        alertar sobre falha de reconhecimento do cartão
    }

    WiFi.begin(ssid, password); // Inicia a tentativa de conexão com o
    ponto de acesso Wi-Fi utilizando as credenciais (SSID/Senha) para
    habilitar o envio de dados para a nuvem.
    tempoInicial = millis(); // Registra o timestamp atual em ms para o
    cálculo de timeouts de rede e sincronização temporal do fluxo de dados
    (Data Logging).

    // Configuração de coleta (Core 1)
    xTaskCreatePinnedToCore(TaskColeta, "Coleta", 4096, NULL, 3, NULL,
1);

    //Configuração de comunicação/SD (Core 0)
    xTaskCreatePinnedToCore(TaskComunicacao, "Comunicacao", 8192, NULL,
1, NULL, 0);
}

void TaskColeta(void * pvParameters) {
    TickType_t xLastWakeTime = xTaskGetTickCount();

```

```

const TickType_t xFrequency = pdMS_TO_TICKS(INTERVALO_MS);

for(;;) {
    if (!gravacaoConcluida) {
        //Verifica se o tempo estipulado acabou
        if (millis() - tempoInicial >= DURACAO_GRAVACAO) {
            gravacaoConcluida = true;
            Serial.println("Tempo de coleta atingido!");
        } else {
            //Realiza a leitura do sinal
            int16_t sinal = (int16_t)analogRead(ECG_PIN);

            //Adiciona o buffer se houver espaço
            if(bufferRAM.size() < MAX_BUFFER) {
                bufferRAM.push_back(sinal);
            }
        }
    }
    //Mantém a frequência de 150Hz de forma rígida
    vTaskDelayUntil(&xLastWakeTime, xFrequency);
}

void TaskComunicacao(void * pvParameters) {
    for(;;) {
        //Se a gravação terminou e o buffer esvaziou, fecha o arquivo e
        encerra
        if (gravacaoConcluida && bufferRAM.empty()) {
            if (sdOk && dataFile) {
                dataFile.flush();
                dataFile.close();
                sdOk = false;
                digitalWrite(LED_STATUS, LOW); // Função para apagar o LED ao
                finalizar tudo
                Serial.println("Arquivo fechado com sucesso. O cartão pode ser
                removido.");
            }
            vTaskDelay(pdMS_TO_TICKS(1000));
            continue;
        }

        //Processa os dados do buffer
        if (!bufferRAM.empty()) {
            int16_t valor = bufferRAM.front();
            bufferRAM.pop_front();
            bool enviado = false;

            //Sistema para envio de dados via Wi-Fi ou cartão SD

            //1. Tenta enviar via Wi-Fi
            if (WiFi.status() == WL_CONNECTED) {

```

```

    HTTPClient http;
    http.begin(serverUrl);
    http.addHeader("Content-Type", "application/octet-stream");
    http.setTimeout(30);

    if (http.POST((uint8_t *)&valor, sizeof(valor)) == 200) {
        enviado = true;
    }
    http.end();
}

//2. Se a conexão Wi-Fi falhar, os arquivos são salvos no SD
if (!enviado && sdOk) {
    dataFile.write((uint8_t *)&valor, sizeof(valor));

    //Flush periódico para segurança dos dados
    static int cont = 0;
    if(++cont > 150) {
        dataFile.flush();
        cont = 0;
        //Pisca LED rapidamente para indicar gravação no SD
        digitalWrite(LED_STATUS, !digitalRead(LED_STATUS));
    }
}
else {
    vTaskDelay(pdMS_TO_TICKS(10));
}
}

void loop() {}

```

Código em *Python* para a aquisição das informações do sinal, o armazenamento das amplitudes no cartão SD e o envio para a nuvem (*Sheets*)

```
import wfdb
import time
import os
import pandas as pd
import numpy as np
import gspread
import socket
from datetime import datetime
from oauth2client.service_account import ServiceAccountCredentials
from scipy.signal import find_peaks

#Configurações do cartão SD (Lembrar de checar se tá sendo chamado de
"E:/" ou "D:/")
LETRA_SD = "E:/"
#Criação de uma pasta com nome único para não sobrescrever o backup
anterior
timestamp_inicio = datetime.now().strftime("%Y%m%d_%H%M%S") #Nomeando
com o ano, dia e horário de início
arquivo_sd = os.path.join(LETRA_SD,
f"backup_ecg_{timestamp_inicio}.csv") #Salva em formato .txt

#Configuração dos diretórios e credenciais json
diretorio_pasta = r"C:/Users/Livia/xxx"
nome_registro = "xxx" #Trocar a cada amostra diferente
caminho_completo = os.path.join(diretorio_pasta, nome_registro)
json_credencial = r"C:/Users/Livia/xxx...43f78.json"

fs = 250
duracao_total_min = xx #Intervalo de tempo de 30 minutos (ajuste
conforme o pedido)
total_amstras_alvo = fs * 60 * duracao_total_min #n° total de amostras
=250 * 60 * (xx)

#Configuração de Internet
def internet_on():
    try:
        socket.create_connection(("8.8.8.8", 53), timeout=2) #Tenta
estabelecer uma conexão de baixo nível com o DNS do Google (8.8.8.8) na
porta 53 para validar o acesso real à internet
        return True
```

```

    except: #Captura qualquer falha de timeout ou rede, sinalizando que
o sistema deve operar em modo offline (apenas gravação no SD)
        return False

#Planilha
def preparar_planilha():
    try:
        escopo = ["https://spreadsheets.google.com/feeds",
"https://www.googleapis.com/auth/drive"] #Lista as URLs de escopo
(Scopes) que permitem à aplicação acessar as planilhas e o sistema de
arquivos na nuvem
        creds =
ServiceAccountCredentials.from_json_keyfile_name(json_credencial,
escopo)#Realiza a autenticação segura utilizando uma chave JSON
(Service Account), garantindo o acesso sem intervenção humana
        client = gspread.authorize(creds) #Autoriza o cliente gspread
com as credenciais obtidas, estabelecendo a sessão de comunicação
criptografada com a API do Google

        #Abre a planilha
        sh = client.open("ECG_Monitoring")
        #Timestamp no título da aba organiza os envios
        nome_aba = f"Sinal_{timestamp_inicio}"

        try:
            worksheet = sh.add_worksheet(title=nome_aba,
rows=total_amostras_alvo + 1, cols=2) #Tenta criar uma nova aba
(worksheet) dimensionada exatamente para o volume de dados esperado,
otimizando o uso de células na nuvem
        except:
            worksheet = sh.worksheet(nome_aba) #Caso a aba já exista,
recupera a instância atual para evitar erros de duplicidade e garantir
a continuidade do registro

            worksheet.clear()
            worksheet.append_row(["Amostra", "Amplitude"]) #Cabeçalho
focado na onda
            return worksheet
        except:
            return None

```

```

#Execução principal

print(f"Analisando os sinais | Tempo estimado: {duracao_total_min} minutos")

try:
    record = wfdb.rdrecord(caminho_completo)
    #Pega apenas as amostras correspondentes ao tempo estipulado
    sinal = record.p_signal[:total_amostras_alvo, 0]
except Exception as e:
    print(f"Erro ao carregar sinal!!: {e}")
    exit()

#Gravação no cartão SD, mantendo o arquivo lá mesmo após o envio para a nuvem
print(f"Gravando {total_amostras_alvo} amostras no SD")

dados_buffer = []

for i in range(len(sinal)):
    #Linha com índice da amostra e a amplitude bruta
    dados_buffer.append([i, sinal[i]])

    #A gravação no cartão SD é feita em blocos de 10.000 para não sobrecarregar o processador
    if len(dados_buffer) >= 10000:
        df_sd = pd.DataFrame(dados_buffer, columns=["Amostra", "Amplitude"])
        df_sd.to_csv(arquivo_sd, mode='a', header=not os.path.exists(arquivo_sd), index=False)
        dados_buffer = [] #Limpa o buffer após escrever

#Garante que o que sobrou no buffer também vá para o SD
if dados_buffer:
    pd.DataFrame(dados_buffer).to_csv(arquivo_sd, mode='a', header=not os.path.exists(arquivo_sd), index=False)

print(f"Backup concluído no SD: {arquivo_sd}")

#Envio para a nuvem (Sheets) após completar o ciclo
if internet_on():
    print("Conexão detectada! Iniciando transferência para o Sheets!")
    planilha = preparar_planilha()

```

```

if planilha:
    #Leitura do arquivo do SD para subir à nuvem
    df_total = pd.read_csv(arquivo_sd)

    tamanho_chunk = 10000
    for inicio in range(0, len(df_total), tamanho_chunk):
        fim = inicio + tamanho_chunk
        bloco = df_total.iloc[inicio:fim].values.tolist()

        try:
            planilha.append_rows(bloco)
            print(f"Sincronizando nuvem
{round((fim/len(df_total))*100, 1)}%")

            #Pequena pausa de 2 s para não estourar a cota requisições
            time.sleep(2)

        except gspread.exceptions.APIError as e:
            if "429" in str(e): #Erro 429 sinaliza que as amostras
estão sendo enviadas rápido demais
                print("Cota atingida! Esperando 15 segundos para
retomar.")

                time.sleep(15)
                planilha.append_rows(bloco) #Tenta de novo após a
espera

            else

                print("Dados enviados para o Sheets com sucesso.")
else:
    print("Sem internet no momento. O arquivo permanece salvo apenas no
SD.")

print("Processo finalizado.")

```

Código em *Python* para a extração das amostras do sinal na nuvem, reconstrução do sinal e envio para a Web das ondas e informações

```
import numpy as np
import scipy.signal as signal
import threading
import time
import os
import pandas as pd
import streamlit as st
import gspread
from oauth2client.service_account import ServiceAccountCredentials
from datetime import datetime
import json

#Sheets
Nome_planilha = "ECG_Monitoring"
#Credencial Json
Caminho_Json_Local = r"C:/Users/Livia/xxx...43f78.json"

def preparar_credenciais():
    #Gerencia a conexão com o Google Sheets de forma híbrida (Local ou
    Nuvem)
    escopo = ["https://spreadsheets.google.com/feeds",
    "https://www.googleapis.com/auth/drive"]

    #1. Tentativa via Secrets
    try:
        if "gcp_service_account" in st.secrets:
            info_chaves = dict(st.secrets["gcp_service_account"])
            if "private_key" in info_chaves:
                info_chaves["private_key"] =
info_chaves["private_key"].replace("\\n", "\n")

                creds =
ServiceAccountCredentials.from_json_keyfile_dict(info_chaves, escopo)
                return gspread.authorize(creds)
    except:
        pass

    #2. Tentativa via Arquivo Local (Notebook)
    try:
```

```

        if os.path.exists(Caminho_Json_Local):
            creds =
ServiceAccountCredentials.from_json_keyfile_name(Caminho_Json_Local,
escopo)

            return gspread.authorize(creds)
        except Exception as e:
            st.error(f"Erro ao carregar credenciais locais: {e}")

        return None

#Processamento do sinal (Sheets)
def aplicar_filtro_ecg(dados, fs=250):
    #Aplica filtro Butterworth de 4ª ordem (0.5Hz - 40Hz) para limpar o
sinal e estabilizar a linha de base, já que os sinais podem sofrer com
artefatos de visualização na planilha
    nyq = 0.5 * fs
    low_cutoff = 0.5 / nyq
    high_cutoff = 40.0 / nyq
    b, a = signal.butter(4, [low_cutoff, high_cutoff], btype='band')
    return signal.filtfilt(b, a, dados)

#Gerenciamento dos dados em Real Time
class ReceptorTempoReal:
    def __init__(self, freq=250, tamanho_buffer=5000):
        self.intervalo = 1.0 / freq
        self.tamanho_buffer = tamanho_buffer
        self.buffer = [0.0] * tamanho_buffer
        self.indice_escrita = 0
        self.amostra_atual = 0 #Contador para cálculo do tempo
decorrido

        self.rodando = False
        self.lock = threading.Lock() #Impede as múltiplas threadings de
acessarem os dados ao mesmo tempo
        self.fonte_de_dados = None

        def configurar_fonte(self, funcao_leitura): #Caminho para conexão
com origem dos dados
            self.fonte_de_dados = funcao_leitura

        def _loop_execucao(self): #Lê um dado -> guarda no buffer -> espera
4 ms -> repete
            proximo_tempo = time.time()
            while self.rodando:

```

```

        dado = self.fonte_de_dados() if self.fonte_de_dados else
0.0

        with self.lock:
            self.buffer[self.indice_escrita] = dado #Salva o dado
no Buffer

            self.indice_escrita = (self.indice_escrita + 1) %
self.tamanho_buffer #0 "%" faz o Buffer circular
            self.amostra_atual += 1

        proximo_tempo += self.intervalo
        pausa = proximo_tempo - time.time()
        if pausa > 0:
            time.sleep(pausa)
        else:
            proximo_tempo = time.time()

    def iniciar(self):
        self.rodando = True
        self.thread = threading.Thread(target=self._loop_execucao,
daemon=True)
        self.thread.start()

    def obter_janela_visualizacao(self, pontos=1000):
        #Retorna os dados da janela e o tempo decorrido (em segundos)
        with self.lock:
            dados = self.buffer[self.indice_escrita:] +
self.buffer[:self.indice_escrita]
            tempo_seg = self.amostra_atual / 250.0
            return dados[-pontos:], tempo_seg

class LeitorSheets:
    def __init__(self, cliente_gspread):
        try:
            sh = cliente_gspread.open(Nome_planilha)
            self.aba = sh.get_worksheet(-1)
            registros = self.aba.get_all_records()
            df = pd.DataFrame(registros)
            self.dados = df['Amplitude'].tolist()
            self.ponteiro = 0
            st.toast("Conexão com a nuvem estabelecida!")
        except Exception as e:
            st.error(f"Erro ao ler planilha: {e}")
            self.dados = [0.0] * 1000

```

```

def ler_proximo(self): #Recebe os dados sequenciais
    if self.ponteiro >= len(self.dados):
        self.ponteiro = 0
    v = self.dados[self.ponteiro]
    self.ponteiro += 1
    return v

#Streamlit

def main():
    st.set_page_config(page_title="Monitor de ECG em tempo real",
        layout="wide")
    st.title("Exibição do ECG em tempo real")

    #Receptor via Session State
    if 'receptor' not in st.session_state:
        cliente = preparar_credenciais()
        if cliente:
            with st.spinner('Sincronizando sinais da nuvem'):
                leitor = LeitorSheets(cliente)
                receptor = ReceptorTempoReal()
                receptor.configurar_fonte(leitor.ler_proximo)
                receptor.iniciar()
            st.session_state.receptor = receptor
        else:
            st.warning("Credenciais não encontradas. Verifique o
arquivo JSON ou os Secrets.")
            return

    #Loop Principal de visualização no Streamlit
    if 'receptor' in st.session_state:
        col1, col2, col3 = st.columns(3)
        placeholder_bpm = col1.empty()
        placeholder_status = col2.empty()
        placeholder_tempo = col3.empty()
        placeholder_grafico = st.empty()

        while True:
            #Coleta de dados da planilha (a cada 4 ms)
            janela, tempo_total_seg =
st.session_state.receptor.obter_janela_visualizacao(1000)

```

```

# Cronômetro de tempo corrido (em MM:SS)
minutos = int(tempo_total_seg // 60)
segundos = int(tempo_total_seg % 60)
tempo_str = f"{minutos:02d}:{segundos:02d}"

#Detecção de batimentos
sinal_limpo = aplicar_filtro_ecg(np.array(janela))
picos, _ = signal.find_peaks(sinal_limpo, distance=150,
prominence=0.5)

bpm = 0
status = "Estabilizando..."
if len(picos) >= 2:
    bpm = int(60.0 / np.mean(np.diff(picos) / 250.0))
    status = "Taquicardia" if bpm > 100 else "Bradycardia"
if bpm < 60 else "Normal"

#Definição de componentes visuais
placeholder_bpm.metric("Frequência Cardíaca", f"{bpm} BPM")
placeholder_status.metric("Condição", status)
placeholder_tempo.metric("Tempo Decorrido", tempo_str)

placeholder_grafico.line_chart(sinal_limpo, height=450,
use_container_width=True)

time.sleep(0.3)

if __name__ == "__main__":
    main()

```