



## Medição de Frequência Cardíaca com Arduino + Sensor + Buzzer e com a conexão com Wi-Fi e o uso do MQTT Explorer – Broker Mosquitto: IoT (protótipo para medir batimentos cardíacos)

Integrante Aline Cristina Garcia de Oliveira, Professor André Luís de Oliveira

<sup>1</sup> Faculdade de Computação e Informática  
Universidade Presbiteriana Mackenzie (UPM) – São Paulo, SP – Brasil

10424073@mackenzista.com.br

**Abstract.** This article presents the development of an accessible system for measuring heart rate, focusing on basic health monitoring. The project uses a pulse sensor integrated into an Arduino ESP32 microcontroller, with Wi-Fi connectivity, to measure heart rate (BPM) in real time. The data is transmitted via MQTT protocol to a broker, allowing its visualization in compatible software. In addition, a buzzer is automatically activated when BPM values are outside the healthy range, emitting audible alerts. The proposal aligns with Sustainable Development Goal (SDG) 3 – Good Health and Well-being, promoting prevention and self-care. It is a practical and educational solution, ideal for home, school, or community environments.

**Resumo.** Este artigo apresenta o desenvolvimento de um sistema acessível para medição de batimentos cardíacos, voltando ao monitoramento básico de saúde. O projeto utiliza um sensor de pulso integrado ao microcontrolador Arduino ESP32, com conectividade Wi-Fi, para medir a frequência cardíaca (BPM) em tempo real. Os dados são transmitidos via protocolo MQTT a um broker, permitindo sua visualização em softwares compatíveis. Além disso, um buzzer é acionado automaticamente quando os valores de BPM estão fora dos limites consideradas saudáveis, emitindo alertas sonoros. A proposta está alinhada com o Objetivo de Desenvolvimento Sustentável (ODS) 3 – Saúde e Bem-Estar, promovendo a prevenção e o autocuidado. Trata-se de uma solução prática e educativa, ideal para ambientes domésticos, escolares ou comunitários.

## 1. Introdução

A saúde é um dos pilares fundamentais para o desenvolvimento humano e social. Em um mundo onde o acesso à tecnologia pode transformar vidas, projetos de baixo custo e fácil implementação tem o cuidado com a saúde, especialmente em comunidades vulneráveis ou com infraestrutura limitada.

Este projeto propõe a construção de um sistema portátil de monitoramento de sinais vitais — frequência cardíaca (BPM) — utilizando o sensor de pulso, um microcontrolador Arduino com chip Wi-Fi, e um buzzer para alertas. A condição é criar uma ferramenta acessível, pode ser muito útil para controle de rotina de exercícios, atividade físicas diárias ou mesmo para fins didáticos e a prevenção de problemas cardiovasculares.

Este projeto está diretamente relacionado ao **ODS 3 – Saúde e Bem-Estar**, cujo objetivo é “assegurar uma vida saudável e promover o bem-estar para todos, em todas as idades.” Mais especificamente, ele contribui com as metas de reduzir a mortalidade por doenças não transmissíveis por meio da prevenção e tratamento e atingir a cobertura universal de saúde, incluindo acesso a serviços essenciais de saúde e medicamentos.

Ao permitir que qualquer pessoa possa monitorar seus sinais vitais de forma simples e eficaz, este projeto promove a conscientização sobre a importância da saúde preventiva e fortalece a autonomia dos indivíduos no cuidado com seu próprio corpo.

### 1.1. Aplicação do projeto

A aplicação do projeto Medição de Frequência Cardíaca + Arduino com chip Wi-Fi + Sensor + Buzzer na IoT reside em criar um dispositivo de monitoramento de saúde conectado, capaz de coletar dados de batimentos em tempo real e emitir alertas sonoros via buzzer.

## 2. Materiais e métodos

### Hardware

Todos os componentes de hardware e software estão sendo demonstrados nessa sessão, tal como seu funcionamento.



Figura 1. Arduino ESP32

É uma placa de microcontrolador para desenvolvimento de baixo custo para projetos da internet das coisas (IoT), que integra o microcontrolador com chip Wi-Fi em uma única placa. Como já possui conexão Wi-Fi integrada, consegue otimizar o uso das portas de entrada e saída da placa.



**Figura 2. Cabos de conexão**

- O cabo azul de rede crimpado que realiza a transferência de dados entre o Arduino e a máquina que estará sendo realizado o código de programação.
- Os jumpers são pequenos pedaços de fios condutores utilizados em eletrônica e engenharia de prototipagem para estabelecer conexões entre pontos em uma placa de circuito.
- O protoboard é usado para a interligação dos componentes eletrônicos em um número quase infinito para produzir circuitos eletrônicos em trabalhos de prototipagem.



**Figura 3. Sensor de pulso e frequência cardíaca SM-S4303R**

É um dispositivo óptico com a capacidade de medir à atividade elétrica do coração. Efetua a leitura das batidas do coração usando um sensor óptico amplificado.



**Figura 4. Buzzer 5v passivo BP18**

É um dispositivo destinado à emitir sinais sonoros a partir do oferecimento de diferentes frequências. O acionamento do buzzer dá-se através da placa microcontroladora.



**Figura 5. Notebook**

Esse equipamento servirá para baixar o app do Arduino para desenvolver o código. Será feita a conexão com a placa Arduino através de um cabo de rede (Figura2).

## Software

**Tabela 1. Relação das ferramentas que serão usadas para o desenvolvimento do software.**

Ferramentas	Funcionalidades
Arduino IDE 2.3.6	Ambiente para programar e carregar o código no Arduino. <ul style="list-style-type: none"><li>• Instalação das bibliotecas:<ol style="list-style-type: none"><li>1. PubSubClient (para MQTT)</li><li>2. PulseSensorPlayground.h</li><li>3. Wifi.h</li></ol></li></ul>
C/C++	Usada para escrever o código do sistema.
MQTT Explorer e Borker Mosquitto	Servidor intermediário responsável por gerenciar a comunicação entre dispositivos por meio do protocolo MQTT, implementado com o broker Mosquitto.

Autor: próprio

#### **4. Funcionamento e acionamento dos materiais**

O protótipo desenvolvido integra sensores, atuadores e conectividade para realizar o monitoramento fisiológico remoto de forma eficiente. O processo tem início com a leitura dos batimentos cardíacos, captados pelo dedo do usuário. O sensor de pulso, baseado em tecnologia óptica amplificada, detecta as variações de fluxo sanguíneo e transmite os dados ao microcontrolador ESP32 por meio de um único pino de sinal conectado à entrada analógica D32.

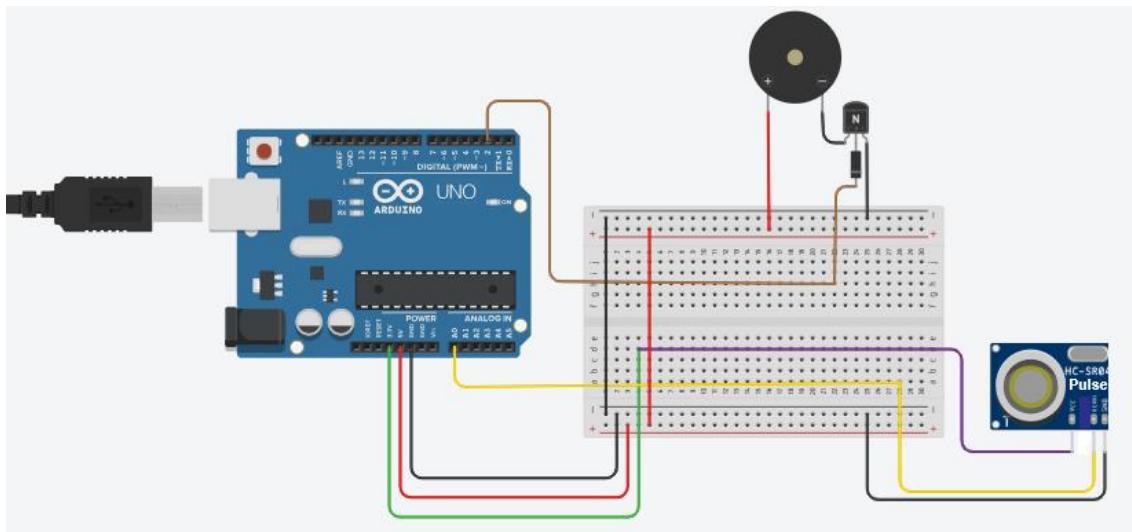
A alimentação dos componentes é fornecida via conexão USB de 5 V, sendo regulada internamente pelo ESP32 para 3.3 V, valor compatível com o sensor de pulso. O buzzer passivo é acionado automaticamente quando os valores medidos ultrapassam os limites pré-definidos, funcionando como um alerta sonoro para situações que exigem atenção imediata — como episódios de taquicardia. Quando os batimentos estão dentro do intervalo saudável de 60 a 120 BPM, o som emitido é intermitente; fora desse intervalo, o buzzer emite um som contínuo.

O microcontrolador opera em modo station, conectando-se a uma rede Wi-Fi local. A transmissão dos dados para monitoramento remoto é realizada via protocolo MQTT, utilizando sockets TCP para comunicação com o broker Mosquitto. Isso permite que os dados sejam publicados em tempo real e acessados por sistemas externos, como dashboards ou aplicativos de saúde.

Por fim, o código responsável pelo funcionamento do sistema é desenvolvido em linguagem C++ e transferido para o ESP32 por meio de cabo USB conectado ao computador, utilizando a IDE Arduino como ambiente de programação. A conexão física via USB entre o microcontrolador e o computador também permite a depuração e o monitoramento em tempo real durante o desenvolvimento.

- A conexão da via cabo USB entre o microcontrolador e notebook transferirá o código em C+ do software para a execução do projeto.

## 5. Prototipagem do projeto



**Fonte:** Projeto desenvolvido pelo próprio autor.

Foi utilizado um microcontrolador ESP32 integrado a um sensor de pulso e a um buzzer passivo. A montagem do circuito foi realizada em uma protoboard, seguindo o diagrama desenvolvido no software TinkerCad, que serviu como ferramenta de documentação visual das interconexões entre os componentes.

A alimentação do sensor foi fornecida diretamente pelo ESP32, operando em 3.3V. O sensor de pulso foi conectado ao pino analógico D32, utilizando os terminais de sinal (S) e saída (OUT). Já o buzzer passivo foi conectado ao pino digital D33 (GPIO 25) do ESP32. Por possuir um circuito integrado, o buzzer não requer componentes adicionais para seu funcionamento.

O modelo do circuito no TinkerCad teve como objetivo validar a disposição dos componentes, além de organizar o roteamento dos fios de alimentação e sinal, garantindo clareza e eficiência na montagem física do sistema.

## 6. Links

Link GitHub

<https://github.com/Aline073/proj-arduino-pulsesensorbpm/blob/main/README.md>

Link YouTube

<https://www.youtube.com/watch?v=2a8gJrmxetA>

## 7. Resultados

As quatro medições de cada sensor e atuador foram coletadas diretamente, no Serial Monitor da IDE do Arduino, conforme demonstrado na imagem a seguir.

```
sketch_nov10a.ino
42 // --- Controle de medições ---
43 const int maxMedicoes = 4;
44 int sensorCount = 0;
45 int buzzerCount = 0;
46 bool testesConcluidos = false;
47
48 // --- Setup ---
49 void setup() {
50   Serial.begin(115200);
51   analogReadResolution(12);
52   analogSetPinAttenuation(PulseWire, ADC_11db);
53
54   pinMode(LED_PIN, OUTPUT);
55   digitalWrite(LED_PIN, HIGH);
```

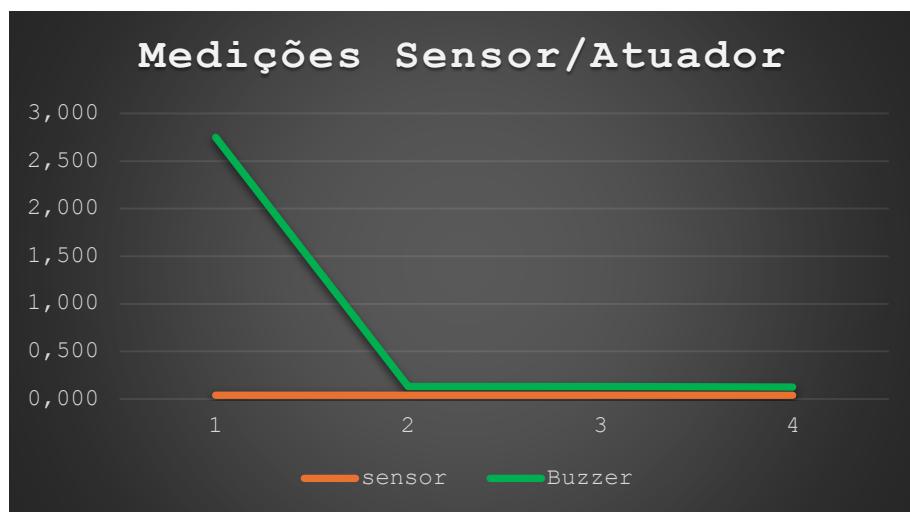
Output    Serial Monitor X

Not connected. Select a board and a port to connect automatically.

WiFi conectado. IP: 192.168.15.174  
Calibracao concluida. Threshold: 1640  
Iniciando testes de tempo...  
Sensor - Medida 1 Tempo: 0.040000 ms  
Sensor - Medida 2 Tempo: 0.040000 ms  
Sensor - Medida 3 Tempo: 0.040000 ms  
Sensor - Medida 4 Tempo: 0.040000 ms  
Buzzer - Medida 1 Tempo: 2.749000 ms  
Buzzer - Medida 2 Tempo: 0.132000 ms  
Buzzer - Medida 3 Tempo: 0.132000 ms  
Buzzer - Medida 4 Tempo: 0.126000 ms  
Testes de tempo concluidos! Iniciando leituras de BPM...  
Tentando conectar ao MOTT...

Valores coletados, apresentados na tabela abaixo, seguidos pelo gráfico correspondente.

N.º Medida	Sensor/Atuador	Tempo de resposta (ms)
1	Sensor	0,040
2	Sensor	0,040
3	Sensor	0,040
4	Sensor	0,040
	Média	0,040
1	Buzzer	2,749
2	Buzzer	0,132
3	Buzzer	0,132
4	Buzzer	0,126
	Média	0,785



## 8. Conclusões

O sistema implementado com o ESP32, o sensor de pulso e o buzzer passivo foi montado e testado em protoboard, os sinais foram validados no modelo do TinkerCad e as leituras do sensor puderam ser lidas e transmitidas via MQTT conforme esperado.

Durante à execução do projeto surgiram alguns obstáculos que atrasaram o andamento. O sensor de pulso apresentou leituras instáveis e dificuldade de calibração, exigindo ajustes no circuito e no tratamento de sinais. A configuração do broker Mosquitto também se mostrou sensível, demandou várias alterações no arquivo de configuração e garantir comunicação estável.

### Vantagens:

- Implementação simples e de baixo custo usando ESP32 e componentes comuns.
- Integração com MQTT permite monitoramento remoto e escalabilidade.
- Uso do TinkerCad acelerou a validação do circuito antes da montagem física.

### Desvantagens:

- Sensibilidade a ruído e variações na leitura do sensor que exigem tratamento por software.
- Dependência de rede e do broker para comunicação remota, acorrendo queda de rede afetando a disponibilidade.
- Montagem em protoboard é menos robusta e suscetível a falsos contatos.
- Substituir protoboard por placa de circuito impresso (PCB) ou montagem com solda para maior confiabilidade.
- Adotar autenticação/TLS no broker Mosquitto em ambientes não controlados e usar QoS apropriado no MQTT para maior confiabilidade.

## 10. Referências

MICROCONTROLLERS LAB. *ESP32 Heart Rate and Pulse Oximeter using MAX30102*. 2023. Disponível em: <https://microcontrollerslab.com/esp32-heart-rate-pulse-oximeter-max30102/>. Acesso em: 14 nov. 2025.

NAÇÕES UNIDAS NO BRASIL. *Sustainable Development Goal 3: Saúde e Bem-Estar*. Disponível em: <https://brasil.un.org/pt-br/sdgs/3>. Acesso em: 14 nov. 2025.

SARAVATI. *Sensor de pulso e frequência cardíaca*. Disponível em: <https://www.saravati.com.br/sensor-de-pulso-e-frequencia-cardiaca.html>. Acesso em: 14 nov. 2025.

USINAINFO. *Buzzer passivo módulo transdutor BP18*. Disponível em: <https://www.usinainfo.com.br/buzzer-arduino/buzzer-passivo-modulo-transdutor-bp18-acionamento-gnd-low-4708.html>. Acesso em: 14 nov. 2025.

MAKERHERO. *Jumpers para prototipagem*. Disponível em: <https://www.makerhero.com/categoria/prototipagem/jumpers/>. Acesso em: 14 nov. 2025.

CASA DA ROBÓTICA. *Protoboard 400 furos + 65 fios jumpers macho x macho*. Disponível em: <https://www.casadarobotica.com/prototipagem-ferramentas/prototipagem/protoboard/protoboard-400-furos-65-fios-jumpers-macho-x-macho>. Acesso em: 14 nov. 2025.

WJ COMPONENTES. *Placa Lolin ESP32*. Disponível em: <https://www.wjcomponentes.com.br/lollin>. Acesso em: 14 nov. 2025.

RECYCLE INFORMÁTICA. *Cabo de rede CAT5e com conector RJ45 1,5m PlusCable*. Disponível em: <https://www.recycleinformatica.com.br/redetelecom/cabo-de-rede/cabo-de-rede-cat5e-com-conector-rj45-1-5m-pluscable>. Acesso em: 14 nov. 2025.

TINKERCAD. *Protótipo ESP32 com sensor de pulso*. Disponível em: <https://www.tinkercad.com/things/gENUM3J52Ye/editel>. Acesso em: 14 nov. 2025.

INSTRUCTABLES. *DIY MAX30102 Heart Health Monitor Using ESP8266*. Disponível em: <https://www.instructables.com/DIY-MAX30102-Heart-Health-Monitor-Using-ESP8266/>. Acesso em: 14 nov. 2025.

TECNOLOGIA E PROJETOS. *Oxímetro de frequência cardíaca e pulso ESP32 com MAX30102 (SpO2 e BPM)*. Disponível em: <https://www.tecnologiaeprojetos.com.br/oximetro-esp32-max30102/>. Acesso em: 14 nov. 2025.

CLUBE DA ELETRÔNICA. Configurando o Mosquitto e liberando o Firewall do Windows para IoT: guia passo a passo [vídeo]. YouTube, s.d. Disponível em: <https://www.youtube.com/watch?v=2114dez0VjI&t=7s>. Acesso em: 16 nov. 2025.

ECLIPSE MOSQUITTO. Mosquitto — Downloads. Disponível em:  
<https://mosquitto.org/download/>. Acesso em: 16 nov. 2025.

MQTT EXPLORER. MQTT Explorer — GUI for MQTT. Disponível em: <https://mqtt-explorer.com/>. Acesso em: 16 nov. 2025.