



Nome do projeto: PulseCerto

Luciano Rocha Reis

¹ Faculdade de Computação e Informática
Universidade Presbiteriana Mackenzie (UPM) – São Paulo, SP – Brazil

10369897@mackenzista.com.br

Abstract. *This project aims to make a low-cost device that monitors the body's pulse, which is the heartbeat, making it useful in the early detection of health problems, contributing to the prevention of diseases and being a valuable tool to monitor our health and well-being. -Being and making decisions based on data.*

Resumo. *Este projeto tem como objetivo fazer um aparelho de baixo custo que monitora o pulso corporal que são os batimentos cardíacos fazendo com que seja útil na detecção precoce de problemas de saúde contribuindo para a prevenção de doenças e sendo uma ferramenta valiosa para acompanhar nossa saúde e bem-estar e tomar decisões embasadas em dados.*

1. Introdução

Recentemente passamos por uma pandemia global que infelizmente contágio milhões de pessoas pelo mundo e evidenciou a necessidade de aparelhos médicos de baixo custo para auxiliar no combate e manutenção da vida, sendo assim, um projeto de baixo custo e que ajude as famílias a monitorar os batimentos cardíacos fará com que seja detectado anormalidades de forma precoce podendo ter a tomada de decisão mais rápida e consequentemente aumento na saúde e bem-estar de quem dispõem do aparelho com esses propósitos.

Minha motivação é poder utilizar esse equipamento em minha família e principalmente em meu filho que recentemente entrou na escolinha e possivelmente será infectado por gripes e poderei identificar um aumento cardíaco e tomar as devidas decisões como levar ao médico de forma precoce, tenho como inspiração projetos semelhantes como já desenvolvidos e que utilizarei como base nesse projeto.

2. Materiais e métodos

Nesse projeto será utilizado alguns materiais e softwares para monitoramento da pulsação e comunicação com a internet (não será utilizado a plataforma Arduino), sendo eles:

ESP8266-01 Wifi Module:

O ESP8266-01 é um módulo WiFi que conecta projetos e sistemas à internet através da rede sem fios. O ESP8266-01 pode ser utilizado em diversas aplicações, adicionando conectividade Wi-Fi a projetos já existentes ou servindo como módulo de conexão para dispositivos que não disponham nativamente de um módulo Wi-Fi.

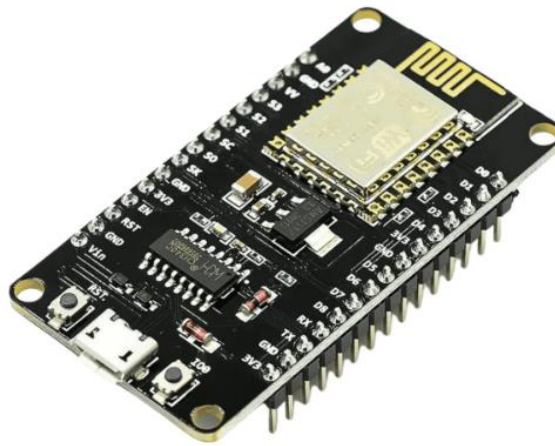


Figura 4 – ESP8266-01 Modulo Wifi

Pulse Sensor:

Um sensor de pulso é um dispositivo óptico pequeno que mede a atividade elétrica do coração. Ele combina um sensor de pulsação óptico com circuitos de amplificação e cancelamento de ruídos para ler a pulsação.



Figura 5 – Sensor de Pulso

Fios Jumpers:

Os fios jumper são fios elétricos com pinos conectores em cada extremidade que podem ser usados para conectar dois pontos sem soldar. Eles são frequentemente usados com protoboards e outras ferramentas de prototipagem. Os fios jumper vêm em três versões: macho para macho, macho para fêmea e fêmea para fêmea.

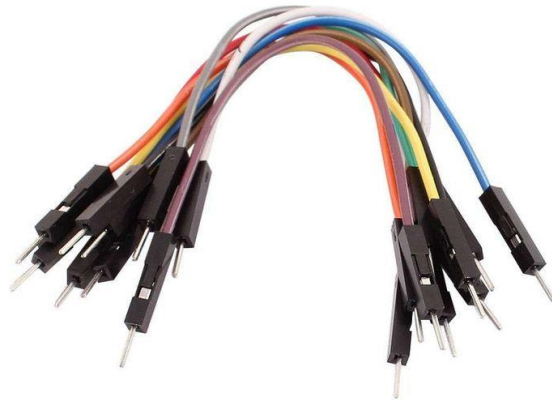


Figura 7 – Fios Jumper

Breadboard Normal:

Uma placa de ensaio é uma placa de plástico retangular com orifícios para componentes eletrônicos, como fileiras de cliques de metal que se prendem aos cabos dos componentes eletrônicos. Placas de ensaio são usadas para criar protótipos de circuitos eletrônicos, realizar experimentos de projeto de circuitos e testar novas peças.

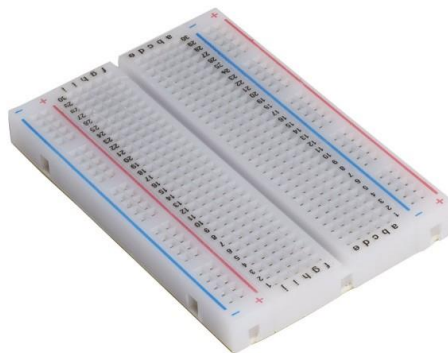


Figura 8 – Placa de Ensaio

Os métodos utilizados nesse projeto serão um Sistema de Monitoramento de Saúde baseado em IOT que registrará a pulsação do paciente. Este sistema não apenas registra esses dados, mas também os atualiza na plataforma IOT através da comunicação via protocolo MQTT.

O princípio de funcionamento deste sensor de frequência cardíaca é muito simples. Se falarmos sobre a frequência cardíaca, então a frequência cardíaca é a proporção de tempo entre dois batimentos cardíacos consecutivos. Da mesma forma, quando o sangue humano circula no corpo humano, esse sangue é espremido nos tecidos. Como resultado, o volume dos tecidos aumenta, mas diminui após cada batimento cardíaco. Esta alteração no volume dos tecidos afeta a luz LED do sensor de frequência cardíaca, que transmite luz após cada batimento cardíaco. Esta mudança na luz é muito pequena, mas pode ser medida conectando qualquer controlador a este sensor de pulso. Isso significa que a luz LED que possui cada sensor de pulso ajuda a medir a frequência de pulso.

O funcionamento deste sensor pode ser verificado colocando um dedo humano na frente deste sensor de pulso. Quando um dedo é colocado na frente deste sensor de pulso, o reflexo da luz LED é alterado com base na alteração do volume de sangue dentro dos vasos. Isso significa que durante o batimento cardíaco o volume de sangue nos vasos estará alto e diminuirá após cada batimento cardíaco. Assim, ao alterar este volume a luz do LED é alterada. Esta mudança na luz LED mede a frequência cardíaca de um dedo. Este fenômeno é conhecido como “Fotopletismograma”.

Funcionamento e Comunicação MQTT:

1. Configuração Inicial:

- O código começa definindo as credenciais Wi-Fi (SSID e senha) e o endereço do servidor MQTT (no caso, “test.mosquitto.org”).
- Ele também configura os pinos do Pulse Sensor (conectado ao pino analógico 0) e do LED embutido no Arduino (pino 2).

2. Função setup_wifi():

- Essa função é responsável por configurar a conexão Wi-Fi.
- Ela tenta se conectar à rede Wi-Fi especificada e exibe o endereço IP local quando a conexão é estabelecida.

3. Loop Principal (loop()):

- A cada iteração do loop, o código lê o valor analógico do Pulse Sensor (armazenado na variável Signal).
- Em seguida, ele armazena essa leitura no array readings para calcular a média móvel.
- Calcula a média móvel dos últimos valores lidos.
- Publica o valor da média móvel no tópico MQTT chamado “device/pulsesensor”.

- O código também verifica se a média móvel está acima do limite definido (Threshold). Se estiver, o LED embutido é desligado; caso contrário, ele é ligado.
- O loop é pausado por 50 milissegundos antes de iniciar a próxima iteração.

4. Função reconnect():

- Essa função é chamada quando a conexão MQTT não está ativa.
- Ela tenta reconectar ao servidor MQTT até que a conexão seja estabelecida.

5. Comunicação MQTT:

- O código utiliza a biblioteca PubSubClient para comunicação MQTT.
- Ele se conecta ao servidor MQTT definido no início (mqtt_server).
- Publica o valor da média móvel no tópico “device/pulsesensor” usando client.publish().
- O servidor MQTT encaminhará essa mensagem para todos os assinantes interessados nesse tópico.

Em resumo, o código envia dados do Pulse Sensor para o servidor MQTT, permitindo que outros dispositivos se inscrevam e recebam essas informações.

O diagrama de circuito do sistema de monitoramento de temperatura e saúde do paciente baseado em IOT é mostrado na figura abaixo.

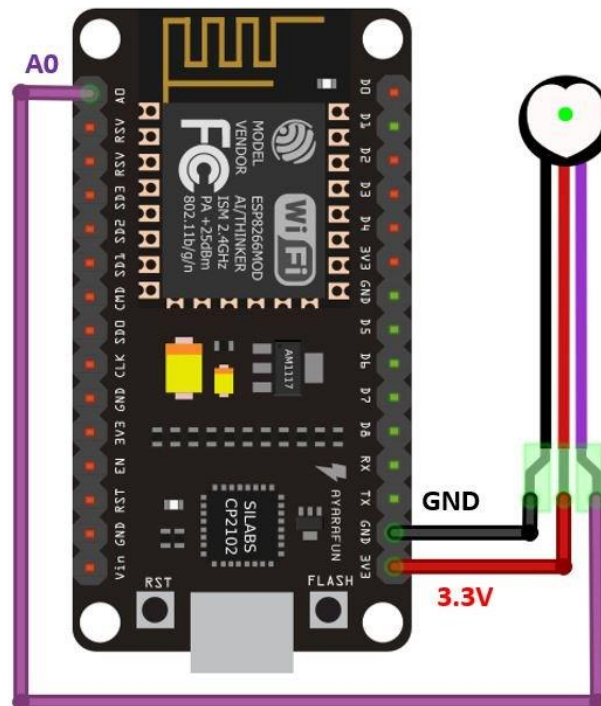


Figura 11 - Diagrama do Circuito

Modelo de Montagem:

O sensor de pulso possui três pinos que devemos conectar ao nosso ESP8266. Isso inclui GND, VCC e pino de sinal.

Siga a tabela e o diagrama esquemático abaixo para conectar seu circuito com sucesso.

ESP8266 Node MCU	Sensor de pulso SEN-11574
GND	GND
A0	A0
5V	CCV

3. Resultados

Outros gráficos, imagens e capturas de tela, demonstrando o funcionamento do protótipo e a comunicação com o *broker* MQTT:

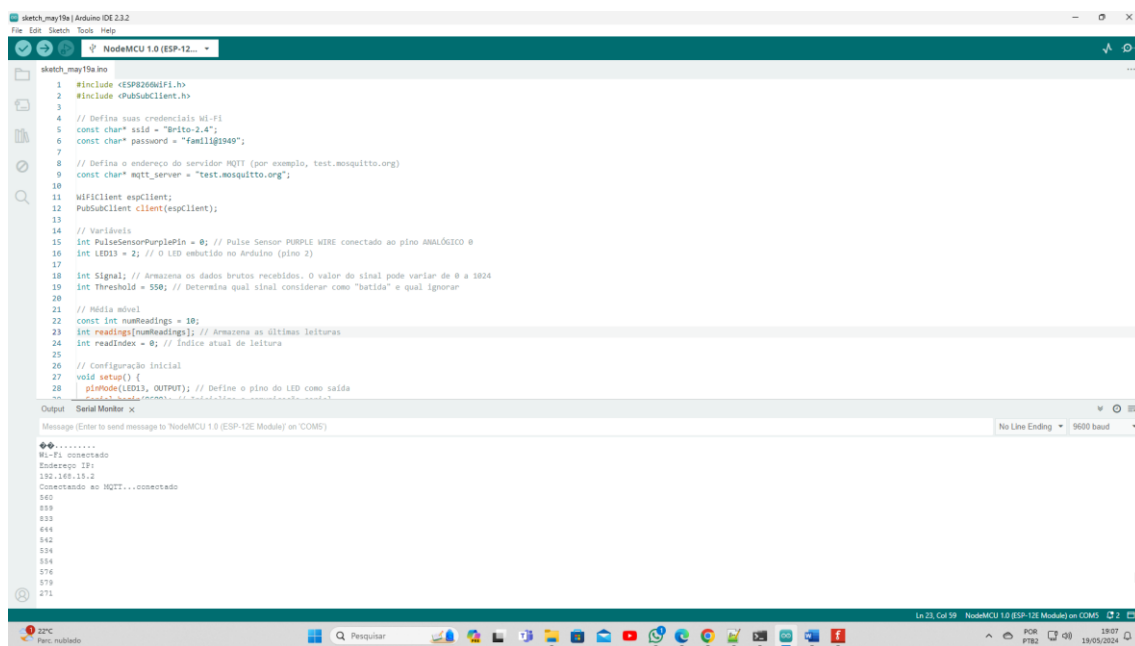


Figura 12 – Testes de conexão

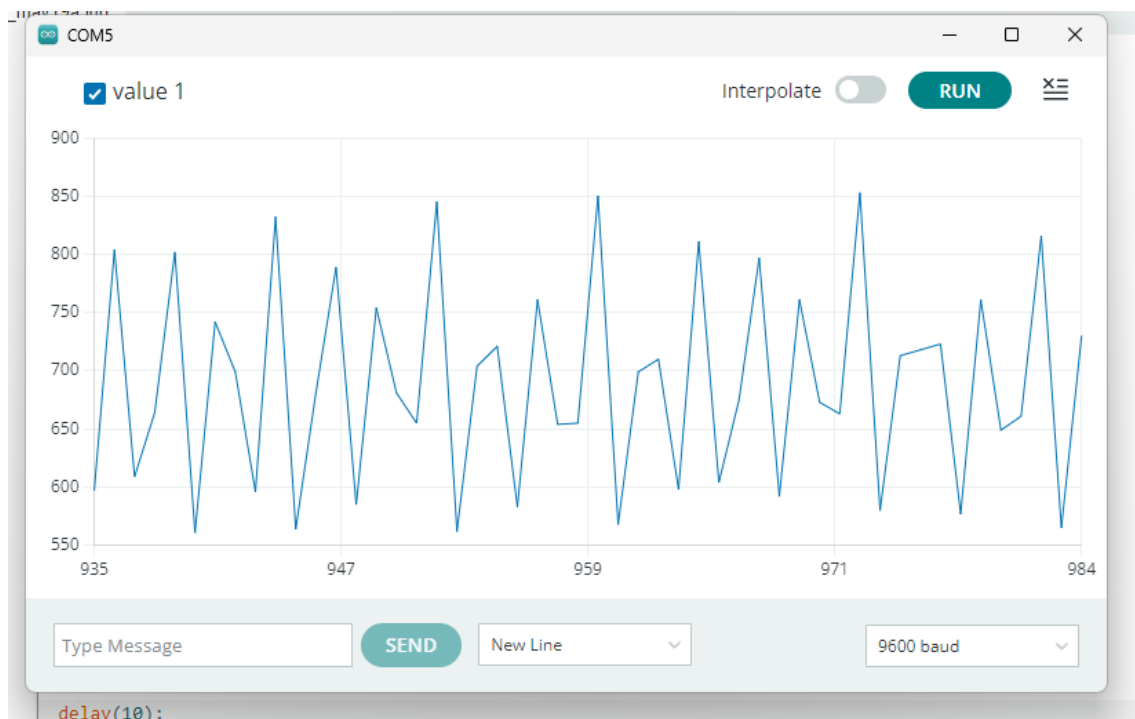


Figura 13 – ESP8266 NodeMCU Traçando Sinal de Batimento Cardíaco

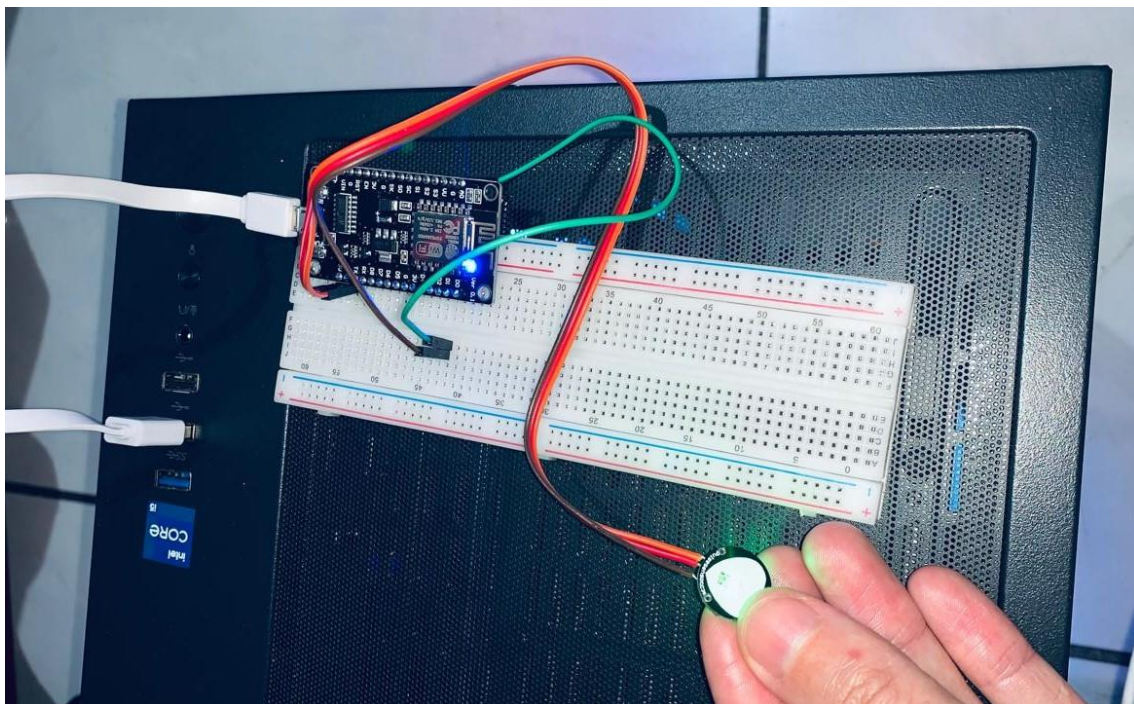


Figura 14 – Foto do Projeto em Funcionamento

Métricas e medições:

O delay atual entre as leituras é de 60 milissegundos programado no código fonte, dessa forma os tempos e respostas são influenciados pelo delay programado.

Código fonte disponibilizado no GitHub.

Link: <https://github.com/LucianoReis/SensorPulseMQTT>

Vídeo-demonstração Youtube:

Link: <https://youtu.be/37VU2c3UK4Q>

4. Conclusões

I. Os objetivos propostos foram alcançados?

Parcialmente, pois devido a especificidade do protocolo MQTT não foi possível encontrar em tempo hábil um software disponível com fácil comunicação para tratar os dados e apresentar os resultados graficamente. Os sites Ubidots, ThingSpeak foram testados a utilização, porém sem sucesso.

II. Quais são os principais problemas enfrentados e como foram resolvidos?

Os principais problemas foram a qualidade do sensor que não detecta com precisão a pulsação e a comunicação via padrão MQTT com plataformas de demonstração gráfica dos resultados, via HTTP é muito mais intuitivo e fácil a integração com outras plataformas.

III. Quais são as vantagens e desvantagens do projeto?

Vantagem baixo custo, desvantagem a qualidade não ser confiável.

IV. O que deveria/poderia ser feito para melhorar o projeto?

Utilização do protocolo HTTP e sensor com qualidade superior.

5. Referências

Electronics Workshop. (21 de 12 de 2022). *IOT based Patient Health Monitoring using ESP8266 & Arduino*. Fonte: hackster.io: <https://www.hackster.io/electronicworkshop111/iot-based-patient-health-monitoring-using-esp8266-arduino-f04ded>

<https://www.arduino.cc/>. (18 de 12 de 2023). *Home Arduino*. Fonte: arduino: <https://www.arduino.cc/>

Microcontrollerslab.com. (12 de 09 de 2021). *Monitore a frequência cardíaca usando sensor de pulso e ESP8266 NodeMCU*. Fonte: Microcontrollerslab.com: https://microcontrollerslab.com/pulse-sensor-esp8266-nodemcu-tutorial/#google_vignette

Souza, F. (29 de 11 de 2013). *Arduino UNO*. Fonte: embarcados.com.br: <https://embarcados.com.br/arduino-uno/>