EE 3200 - Electrical and Electronic Project Design
Department of Electrical and Electronic Engineering

Khulna University of Engineering & Technology

Design and Implementation of a Wireless ECG Monitoring System Using ESP32 and Firebase Cloud Integration

Submitted By

Name: Md. Ohiduzzaman Sajol

Roll: 2003043

Year: 3rd

Term: 2nd

Date of Submission

August 24, 2025

Submitted To (Supervisor)

Dr. Mohammad Shaifur Rahman

Professor, Dept. of EEE, KUET

Abstract

Electrocardiography (ECG) is one of the most widely used non-invasive methods for monitoring heart health. Traditional ECG monitoring systems are expensive, bulky, and often restricted to clinical environments. This project presents the design and implementation of a low-cost wireless ECG monitoring system using the ESP32 microcontroller, the AD8232 biopotential sensor, and Firebase cloud services. The system acquires real-time ECG signals through disposable electrodes, transmits the data via Wi-Fi to a Firebase real-time database, and visualizes the processed signals on a web-based dashboard. The dashboard performs basic signal processing, including heart rate (BPM) calculation and condition classification (e.g., normal, bradycardia, tachycardia). This project demonstrates the feasibility of using IoT technology for continuous, remote, and cost-effective health monitoring applications, paving the way for future developments in telemedicine and wearable devices.

1. Introduction

Cardiovascular diseases (CVDs) remain the leading cause of mortality worldwide, accounting for nearly 17.9 million deaths annually. Early detection of irregularities in cardiac activity can significantly reduce risks associated with heart failure, arrhythmia, and sudden cardiac arrest. Electrocardiography (ECG) is a primary tool used for diagnosing such conditions by recording the electrical activity of the heart.

However, conventional ECG machines are costly, require skilled operators, and are not portable for everyday use. With the advent of low-cost sensors, microcontrollers, and cloud platforms, Internet of Things (IoT)-based health monitoring systems have emerged as promising alternatives. These systems can provide real-time, remote, and continuous monitoring at a fraction of the cost.

This project aims to design and implement a wireless ECG monitoring system that integrates hardware and software to provide real-time data acquisition, transmission, processing, and visualization. The system leverages ESP32 for data collection, Firebase for cloud synchronization, and a web application for user-friendly visualization and health condition analysis.

Objectives

1. To design a low-cost ECG acquisition system using the AD8232 sensor and ESP32 microcontroller.

2. To transmit ECG data wirelessly to a cloud database (Firebase).

3. To process the ECG data server-side for BPM calculation and condition classification.

4. To develop a web-based dashboard for real-time visualization and monitoring.

5. To evaluate system performance and limitations in practical scenarios.

2. Literature Review

2.1 Existing ECG Monitoring Systems

Traditional hospital-grade ECG devices typically consist of multi-lead systems connected to a processing unit that displays signals on a monitor or prints them on paper. While highly accurate, these systems are bulky and not suitable for continuous remote monitoring. Portable Holter monitors exist but are expensive and require specialized handling.

2.2 IoT-Based Health Monitoring

Recent research has focused on IoT-enabled health monitoring solutions. These systems utilize microcontrollers (such as Arduino, ESP32, or Raspberry Pi) connected to biosensors to capture physiological signals and transmit them over Wi-Fi or Bluetooth to cloud platforms. The data can then be accessed remotely via mobile apps or web interfaces.

2.3 Limitations of Existing Systems

Most commercial and research IoT ECG systems face challenges such as:

- High cost of deployment.

- Limited data security and privacy.

- Restricted usability due to poor electrode placement or signal noise.

- Lack of real-time processing for meaningful clinical interpretation.

3. System Design and Methodology

3.1 System Overview

The system consists of three primary components:

1. **Signal Acquisition:** ECG signal captured by AD8232 through disposable electrodes.

2. **Signal Transmission:** ESP32 samples the signal (at 250 Hz), converts it to digital form, and transmits it to Firebase over Wi-Fi.

3. **Signal Visualization & Processing:** A web application retrieves data from Firebase, plots the ECG waveform, calculates BPM, and identifies basic cardiac conditions.

3.2 Block Diagram

Electrodes ⇨ AD8232 Sensor ⇨ ESP32 ⇨ Wi-Fi ⇨ Firabase RTDB ⇨ Web Dashboard

3.3 Hardware Components

- **ESP32 Development Board:** Dual-core microcontroller with Wi-Fi and Bluetooth capabilities.

- **AD8232 ECG Sensor:** Low-power biopotential amplifier optimized for ECG and heart-rate monitoring.

- **Disposable Gel Electrodes:** Used for acquiring biopotential signals from the skin.

- **Power Supply:** 5V USB supply for ESP32 and sensor.

3.4 Software Components

- **Arduino IDE:** Used for ESP32 programming.

- **Firebase Realtime Database:** Cloud storage for ECG data.

- **Firebase ESP Client Library:** Facilitates ESP32-to-Firebase communication.

- **Web Application:** Developed using HTML, CSS, and JavaScript for live visualization.

3.5 Data Flow

1. Analog ECG signal acquired by AD8232.

2. ESP32 samples signal via ADC at 250 Hz.

3. Data transmitted to Firebase in small chunks.

4. Web app retrieves and plots signal in real-time.

5. Signal processing applied: BPM detection (via peak detection) and condition classification.
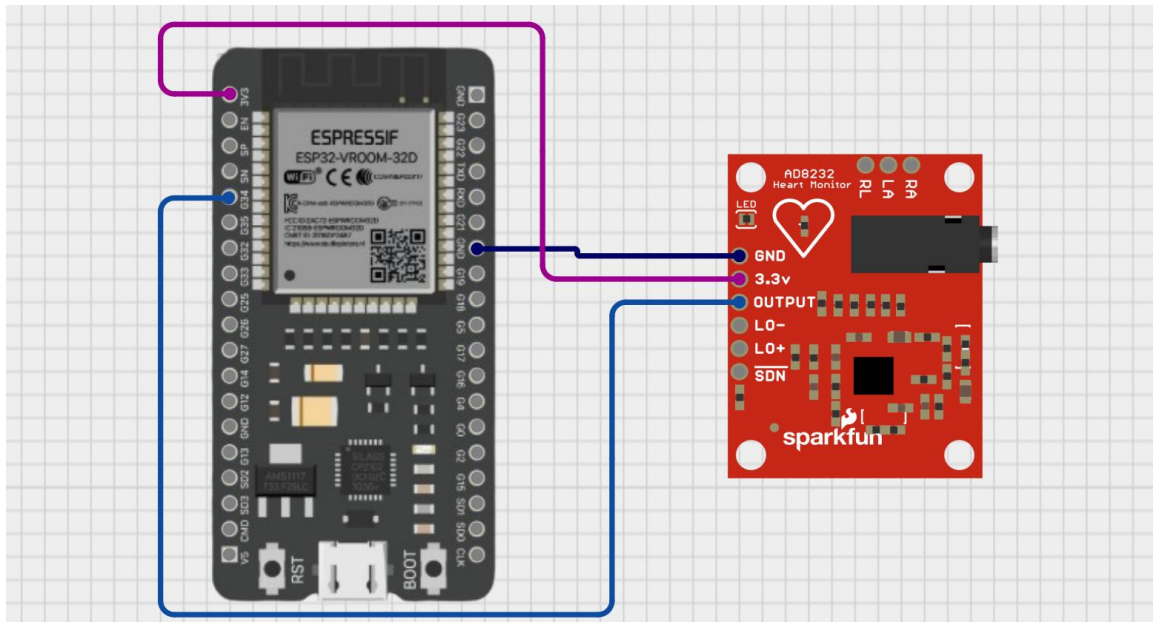
4. Implementation

4.1 Hardware Setup

The AD8232 module was connected to the ESP32 as follows:

- Output → ESP32 ADC pin (GPIO 34).

- LO+ / LO- pins connected for electrode lead-off detection.

- Power pins connected to 3.3V and GND.
  Three disposable electrodes were used: Right arm (RA), Left arm (LA), and Right leg (RL) as reference.

4.2 Firmware Development

The ESP32 code was written in Arduino IDE using the **Firebase_ESP_Client** library.

- Sampling frequency: 250 Hz.

- Data buffered in small chunks before uploading to Firebase.

- JSON format used for data transfer to maintain structured records.

4.3 Firebase Setup

- A new Firebase project was created with Realtime Database enabled.

- ESP32 authenticated using an API key and project credentials.

- Database Structured as:

```
/ecg_data
    timestamp
    signal_value
```

4.4 Web Dashboard Development

The web application retrieves ECG data in real-time using Firebase JavaScript SDK. Features include:

- **Real-time ECG plotting** using Chart.js.

- **BPM calculation** through peak detection.

- **Condition classification:**

    o Normal (60–100 BPM).

    o Bradycardia (<60 BPM).

    o Tachycardia (>100 BPM).

- **UI styling:** Responsive design, clean layout with CSS styling for readability.

5. Results and Analysis

5.1 System Functionality

- ECG waveform successfully acquired, transmitted, and displayed on **web dashboard** which can be found in this URL: ***https://ecg-monitor-ohid.web.app/.***

- Real-time BPM calculation achieved with ~1-second delay.

- Condition analysis correctly displayed based on BPM thresholds.

5.2 Observations

- Without proper electrode placement, the ECG signal flattened due to poor contact.

- The system detected consistent BPM in stable conditions but showed errors when noise or motion artifacts were present.

- A typical test showed a heart rate of ~72 BPM, correctly classified as normal.

5.3 Limitations

- Susceptible to noise from poor electrode placement.

- Single-lead ECG only (limited diagnostic accuracy).

- Basic BPM-based classification only (no advanced arrhythmia detection).

6. Discussion

The developed system demonstrates the potential of combining ESP32 and Firebase for real-time biomedical signal monitoring. While the accuracy is limited compared to clinical-grade ECG machines, it provides valuable insights for preliminary health monitoring and educational purposes.

The project highlights the balance between cost, simplicity, and functionality. With improvements in filtering, multi-lead acquisition, and AI-based classification, the system can evolve into a practical telemedicine solution.

7. Conclusion and Future Work

Conclusion

The project successfully demonstrated a wireless ECG monitoring system using ESP32, AD8232, and Firebase. It achieved real-time data acquisition, transmission, processing, and visualization through a web-based dashboard. The system offers an affordable, accessible, and portable solution for ECG monitoring, aligning with the global demand for low-cost healthcare technologies.

Source Code Files and Project Related Resources

All the resources including source code files, firebase configuration files etc. can be found in this repository: *https://github.com/projectohid/EE-3200-Electrical-and-Electronic-Project-Design*

Future Work

1. Implementation of digital filtering (bandpass, notch filters) to remove noise.

2. Multi-lead ECG support for improved diagnostic accuracy.

3. AI-based arrhythmia detection using machine learning models.

4. Development of a mobile application for increased portability.

5. Integration of data security and encryption for medical compliance.

8. References

[1] Analog Devices, "AD8232 Heart Rate Monitor Front End," Datasheet, 2019.

[2] Google Firebase, "Firebase Realtime Database Documentation," Available: https://firebase.google.com/docs/database

[3] Espressif Systems, "ESP32 Technical Reference Manual," 2020.

[4] World Health Organization, "Cardiovascular diseases (CVDs)," 2021. Available: https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds)

[5] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," Future Generation Computer Systems, vol. 29, no. 7, pp. 1645–1660, 2013.