Vol. 7, No. 3, November 2025 (Special Issue), pp. 311~320, p-ISSN 2685-2381, e-ISSN 2715-2626
http://dx.doi.org/10.28989/avitec.v7i3.3144

Portable ECG Prototype based on Arduino and Random Forest Classification for Home Heart-Rate Monitoring

R. Ferdy Akbar Nugraha[1]*, Novendy Alberto Will Tindaon[2], Arya Susena[3], Alfonso Duandes[4], Achmad Ridwan[5]
[1,2,3,4]Department of Informatics Engineering, Faculty of Science and Technology, Universitas Prima Indonesia, Indonesia
[5]Center of Excellence for Health Based on IoT and Renewable Energy, Department of Informatics Engineering, Faculty of Science and Technology, Universitas Prima Indonesia, Indonesia.

Article Info

Article history:
Submitted July 5, 2025
Accepted July 31, 2025
Published August 19, 2025

ABSTRACT

Electrocardiogram (ECG) examination is essential for detecting heart rhythm disorders, yet limited access and high costs often prevent routine medical check-ups for many people. This study addresses these obstacles by designing and developing a portable ECG prototype capable of independent home-based heart monitoring. The system integrates an AD8232 sensor for signal acquisition, an Arduino Uno microcontroller as the main processor, and a simplified Random Forest classification algorithm to distinguish between normal, bradycardia, and tachycardia conditions. Measurement results are saved in CSV format on an SD card, then visualized and analyzed using Jupyter Notebook. The prototype was tested on 100 samples in a static and relaxed state to ensure signal stability. Its heartbeat classification achieved an accuracy of 99.0%, slightly higher than the PTB-XL reference dataset's 98.0%, and consistent with results reported by recent TinyML- and Random Forest-based ECG studies. Unlike prior IoT-based frameworks, this work combines cost-effective microcontroller hardware with simplified offline on-device classification for practical daily monitoring without continuous cloud access. These findings confirm that the proposed system can produce reliable readings approaching clinical standards while remaining simple, affordable with a component cost under USD 31, and accessible for routine public heart health screening.

Keywords:
Portable ECG;
random forest;
heartbeat classification;
Arduino Uno;
AD8232.

Corresponding Author:
R. Ferdy Akbar Nugraha
Department of Informatics Engineering, Faculty of Science and Technology, Universitas Prima Indonesia
Jl. Sampul No.3, Sei Putih Bar., Kec. Medan Petisah, Kota Medan, Sumatera Utara 20118, Indonesia
Email: *radenferdy183@gmail.com

1. INTRODUCTION

Cardiovascular diseases (CVDs) remain a leading cause of death globally, including in Indonesia. Heart attacks and arrhythmias, which are major contributors to CVDs, account for approximately 31% of all worldwide deaths, with over a million people estimated to die from heart disease annually in Indonesia alone. This underscores the critical need for accessible cardiac health monitoring technology for the wider community. Conventional electrocardiogram (ECG) equipment, while fundamental for heart monitoring, is often bulky, expensive, and requires skilled medical personnel to operate [1][2]. These limitations severely restrict its availability outside clinical settings, particularly in remote areas or for communities with lower economic status [3]. Consequently, there is a clear gap for developing simpler and more portable ECG monitoring solutions that can reach wider populations at lower cost.

The rapid advancement of microelectronics, especially the application of microcontrollers like Arduino, offers an innovative approach to addressing this limitation. Arduino enables the creation of more portable, user-friendly, and affordable medical devices [4]. Its real-time sensor data processing capabilities make Arduino an excellent choice for developing a portable ECG prototype. A recent study by Yusuf et al. (2025) also highlights how combining an Arduino Uno with an AD8232 ECG sensor can deliver a practical and low-cost solution for independent heart signal monitoring at home [5]. However, a significant challenge in creating Arduino-based portable ECGs is ensuring measurement accuracy, since signals acquired outside hospital environments are often susceptible to noise. Therefore, developing robust filtering methods and training detection algorithms on larger datasets are crucial steps to minimize diagnostic errors and enhance measurement accuracy [6][7]. Recent studies have also demonstrated the potential of TinyML-based embedded ECG systems for real-time classification directly on low-power microcontrollers [8]. Subba and Chingtham (2024) showed that Random Forest algorithms combined with advanced feature extraction can achieve up to 98% accuracy for ECG arrhythmia detection using benchmark datasets [9]. Recent research has also explored cloud-based ECG frameworks to enable real-time remote health monitoring in smart city contexts, such as the system by Prajitha et al. (2022) [10].

To overcome these challenges, this research prioritizes the convenience of lay users in designing the portable ECG prototype. A clear interface and straightforward instructions are essential to ensure that the instrument can be used effectively by the general public without a medical background. The compact and lightweight design allows it to be used anywhere, anytime, making independent heart health monitoring readily accessible [4]. Moreover, this system aims not only to support independent heart health monitoring but also to contribute to early detection of cardiovascular disorders [11]. Unlike [8][9][10], which focus on IoT frameworks or algorithmic simulations without an affordable physical device, this research uniquely integrates an Arduino Uno microcontroller, an AD8232 ECG sensor, and a simplified Random Forest classifier into a single, portable home-use device with a total component cost under USD 31.

Based on the above background, this study specifically designs and implements a portable Arduino-based ECG prototype capable of accurately recording heart signals under static conditions with minimal physical activity. It applies a simplified Random Forest classification method to identify heartbeat conditions such as normal, bradycardia, and tachycardia, automatically stores ECG signal data in CSV format, and visualizes the results using Jupyter Notebook. The prototype’s signal data are compared with reference datasets to validate accuracy and precision, and its physical comfort and usability are evaluated during testing. This comprehensive approach is expected to deliver a practical, affordable, and independent home heart monitoring solution while providing clear contributions for future developments of microcontroller-based medical devices.

2. RESEARCH METHODS

2.1 Definition of Methods

This research adopts a Research and Development (R&D) approach, which is fundamentally an application-based quantitative study. Its primary aim is to design, develop, and test a portable Electrocardiogram (ECG) device prototype utilizing the Arduino platform and a simplified Random Forest machine learning algorithm [12][13]. This R&D method aligns with the main goal of creating a practical tool capable of recording and displaying heart signals in real time with straightforward classification logic, thus enabling independent use in a home environment.

The methodology uses a mixed-method approach, combining both quantitative and qualitative elements. The quantitative aspect involves rigorous testing of ECG signal parameters and verifying the accuracy of the Random Forest classification algorithm against standard heart rate data [14][15]. Inspired by recent studies on web-based and cloud-compatible ECG frameworks for real-time signal monitoring and quality assessment [15], this study also adopts clear filtering and thresholding protocols based on established signal processing techniques for portable ECG devices [9]. All captured data are stored in CSV format and analyzed using Jupyter Notebook, following best practices from similar IoT-based healthcare models [10]. The qualitative aspect includes observing the device’s ease of use, the clarity of its display, and the comfort of ECG sensor placement for lay users. All tests were conducted under calm, static conditions to ensure stable signal quality and precise classification.

The entire implementation began with an extensive literature review, careful component selection, and a user needs assessment. This was followed by the integration of key parts, including the AD8232 ECG sensor, Arduino Uno, I2C LCD display, and a micro SD Card module. After hardware and software integration, the complete system was tested thoroughly. The resulting ECG signals were stored in CSV format for detailed analysis and visualization. Classification validation was performed by comparing the prototype results with the PTB-XL reference dataset. Overall, this method aims to deliver a reliable, low-cost solution for independent heart monitoring in daily life.

2.2 Work Procedure

The work procedure for this research involved several sequential steps. Initially, a portable ECG prototype was designed and constructed, utilizing Arduino Uno as the microcontroller and the AD8232 ECG sensor for detecting cardiac electrical activity. Following the completion of the circuit assembly, programming was carried out using the Arduino IDE. Subsequently, ECG signal data was collected by attaching the sensor to subjects under various conditions. This was followed by the training of the Random Forest algorithm. Once the model was trained, testing was performed to evaluate the algorithm's performance in detecting abnormal heart signals. The final step involved integrating the algorithm into the portable ECG prototype, enabling the device to analyze ECG signals in real time.

2.3 Resesearch Tools and Materials

The resesearch tools and materials used in this research are detailed in Table 1 and Table 2, respectively.

Table 1. Research tools

Tool Name	Function
Arduino Uno R3	Microcontroller for processing and controlling the system.
AD8232 ECG Sensor (SparkFun)	Captures electrical signals from heartbeats.
LCD 20x4 I2C (HD44780)	Displays text and heart signal measurement results.
Micro SD Card Adapter	Stores recorded ECG data.
TP4056 Li-ion Charger Module (Type-C)	Module for charging LiPo batteries via USB Type-C.
DPST Switch	Double Pole Single Throw switch to safely turn on/off two power lines simultaneously.
On/Off Push Button	To turn the device on and off.
Red & Blue LED	System ON/OFF indicators.
Transistor/MOSFET	Regulates power or switching functions as needed.

Table 2. Research materials

Material Name	Function
3.7V (9V) LiPo Battery	Power source for the system.
Resistor 220Ω	Current limiter, part of the signal filter.
Capacitor 0.1µF	Bypass and coupling filter.

2.4 Prototype Design and Implementation

At this stage, the system was designed to produce a portable ECG signal monitoring device capable of independent use in a home environment. The design process was primarily divided into two main components: hardware and software. Both were meticulously integrated to ensure optimal device functionality, user-friendliness for laypersons, and efficiency in terms of power consumption and data storage.

The hardware components were selected and arranged considering availability, ease of assembly, and cost efficiency. The Arduino Uno microcontroller served as the system's core due to its compatibility with various sensors and support modules, and its programmability. The AD8232 sensor was specifically integrated to capture electrical signals from the user's cardiac activity, with data then transmitted to the Arduino for processing and analysis. For direct result display, a 20x4 I2C-based LCD was used, offering a wide display with minimal pin usage. An SD Card module was incorporated for storing recorded signal results, enabling data saving and transfer to a computer. The system is also equipped with green and red LED indicators to denote operational status or errors. All components were initially assembled on a breadboard for preliminary testing, then transferred to a PCB and encased in an acrylic housing, providing both protection and a professional aesthetic. The complete hardware circuit layout of this system is illustrated in Figure 1.

A detailed description of the microcontroller pin assignments and their corresponding connections to peripheral components is presented in Table 3. This includes the ECG sensor input, display interface, user input buttons, system indicators, data storage module, and power management circuitry. Clear documentation of these connections is essential to replicate, test, and further develop the portable ECG system.

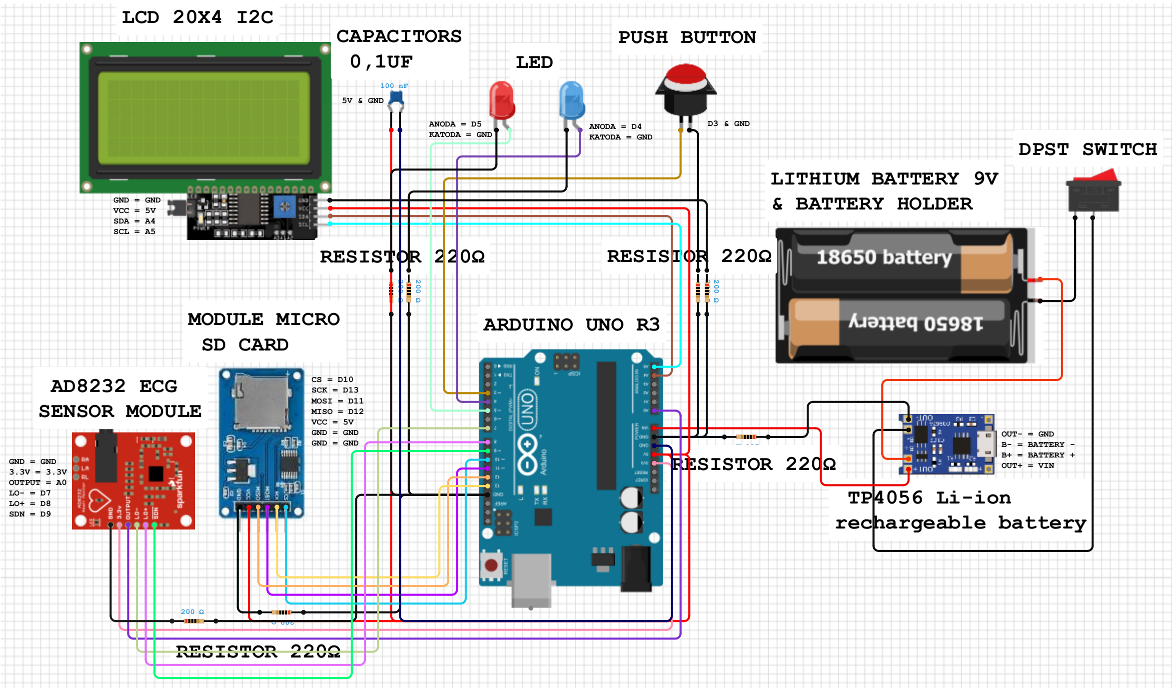

Figure 1. System hardware circuit diagram.

Table 3. Pin mapping of prototype components and Arduino connections

No	Component	Pin / Terminal	Connection Purpose	Arduino Pin	Description
1	Arduino Uno R3	5V	Power Supply	-	Provides 5 V supply voltage
		3.3V	Power Supply	-	Provides 3.3 V supply voltage
		GND	Ground	-	Main system ground
		A0	Analog Input	A0	Receives ECG signal input
		D2	Digital I/O	D2	Interrupt pin for R-peak detection
		D3	Digital I/O	D3	Push button input
		D4-D6	Digital I/O	D4-D6	Green LED, Red LED, Fan control
		D10-D13	SPI Interface	D10-13	SD Card: CS, MOSI, MISO, SCK
2	AD8232 ECG Sensor (SparkFun)	3.3V	Power	3.3V	Sensor operating voltage
		GND	Ground	GND	Sensor ground
		OUTPUT	Signal Out	A0	ECG signal output to Arduino
		LO-	Lead Off –	D7	Detects detached electrode
		LO+	Lead Off +	D8	Detects detached electrode
		SDN	Shutdown	D9	Sensor shutdown control
		RA	Right Arm	-	Right arm electrode
		LA,RL	Left Arm, Right Leg	-	Left arm & right leg electrodes
3	LCD 20x4 I2C (HD44780)	VCC	Power	5V	LCD supply volateg
		GND	Ground	GND	LCD ground
		SDA	Data	A4	I2C data line
		SCL	Clock	A5	I2C clock line
4	Micro SD Card Adapter	VCC	Power	5V	Module supply voltage
		GND	Ground	GND	Module ground
		CS	Chip Select	D10	SD Card control
		MOSI	Data Out	D11	Data to SD Card
		MISO	Data In	D12	Data from SD Card
		SCK	Clock	D13	SPI clock line
5	TP4056 Li-ion Charger Module (Type-C)	B+	Battery +	-	Connects to battery positive
		B-	Battery –	-	Connects to battery negative
		OUT+	Output +	VIN	Supplies voltage to Arduino
		OUT-	Output –	GND	Connects to Arduino ground

No	Component	Pin / Terminal	Connection Purpose	Arduino Pin	Description
6	On/Off Push Button	IN+	Input +	-	From 5 V adapter
		IN-	Input –	-	Adapter ground
		Terminal 1	Input	D3	Input with internal pull-up
		Terminal 2	Ground	GND	Ground via 220 Ω resistor
7	Red LED	Anode(+)	Signal	D5	Controlled via 220 Ω resistor
	Blue LED	Anode(+)	Signal	D4	Controlled via 220 Ω resistor
		Cathode(-)	Ground	GND	LED ground

The software was developed using the Arduino programming language (C++), featuring a modular structure to facilitate debugging and future scalability. This program is designed to read analog signals from the sensor, process them, and then classify these signals based on predetermined thresholds. The classification logic, while simplified for microcontroller constraints, adapts the basic principles of the Random Forest method, utilizing a layered if-else structure that mimics decision tree processes. This simplified approach effectively evaluates the user's heart signal condition. Classification results are displayed on the LCD and saved to the SD Card for documentation. The processing time is remarkably short, under 2 seconds from signal reception to display. Furthermore, the system is power-efficient, allowing several hours of operation on a single charge, complemented by a TP4056 battery charging module for safe and convenient recharging. The overall flow of the software logic is depicted in Figure 2.

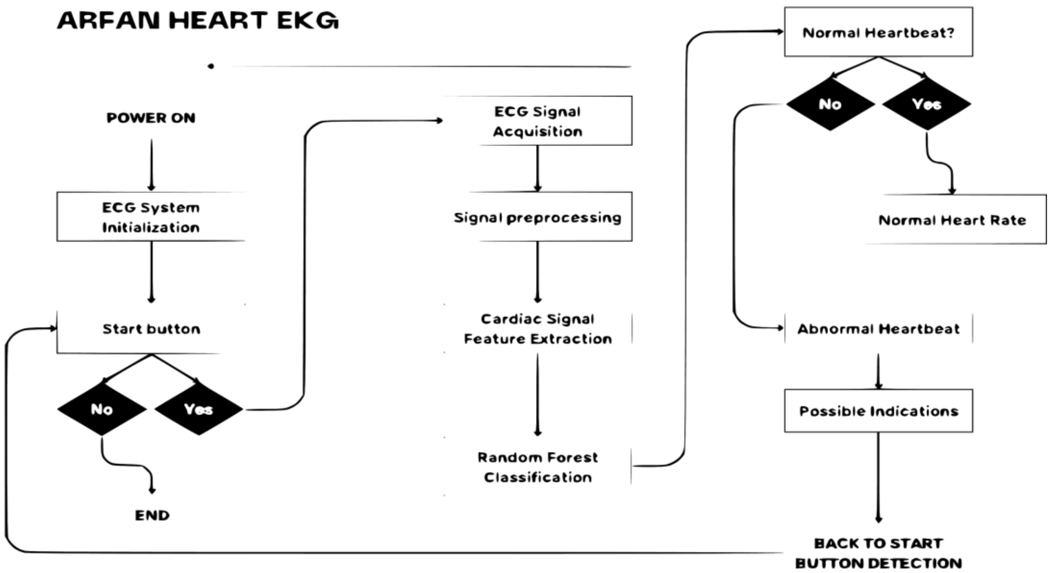

Figure 2. Program workflow flowchart

3. RESULTS AND DISCUSSION

Following the assembly and integration phase, comprehensive testing was conducted to validate all system functionalities. The tests were carried out with users in a resting, seated position while the AD8232 sensor was properly attached. Once activated, heart signals were immediately captured and displayed on the LCD within two seconds, showing heart rhythm status and classification results according to the embedded logic. The measurement data were successfully stored on the SD Card in a transferable CSV format. During static conditions, the system demonstrated stable signal capture; however, minor noise appeared with body movement or cable shifts, which is typical for this sensor type. Proper cable routing and simple filtering help reduce such noise, supporting the device's suitability for daily home monitoring under calm conditions. Figure 3 shows the LCD display during this testing phase.

Figure 3. LCD display during ECG signal testing

After classification, the recorded ECG data were automatically saved to the SD Card for post-analysis in Jupyter Notebook. This analysis visualizes ECG signals graphically, calculates heart rate (HR), and determines the percentage of abnormal beats — specifically bradycardia (HR < 60 BPM) and tachycardia (HR > 100 BPM) — based on the classification results. HR is derived from the RR interval using Equation (1):

$$HR = \frac{60.000}{RR_{\text{Interval(ms)}}} \tag{1}$$

where: HR = Heart rate in beats per minute (BPM)
60.000 = Constant representing the number of milliseconds in one minute (60 seconds × 1000 ms).
$RR_{(interval)}$ = Time difference between two adjacent R-wave peaks (ms).

The percentage of abnormal heartbeats is then calculated using Equation (2):

$$Abnormal\ (\%) = \left(\frac{Number\ of\ HR_{abnormal}}{Total\ HR}\right) \times 100 \tag{2}$$

where: Abnormal (%) = Percentage of abnormal heartbeats (bradycardia or tachycardia)
Number of $HR_{abnormal}$ = Total count of HR data classified as bradycardia or tachycardia
Total HR = Total number of all heart rate data recorded

The result of this classification and signal visualization is shown in Figure 4.

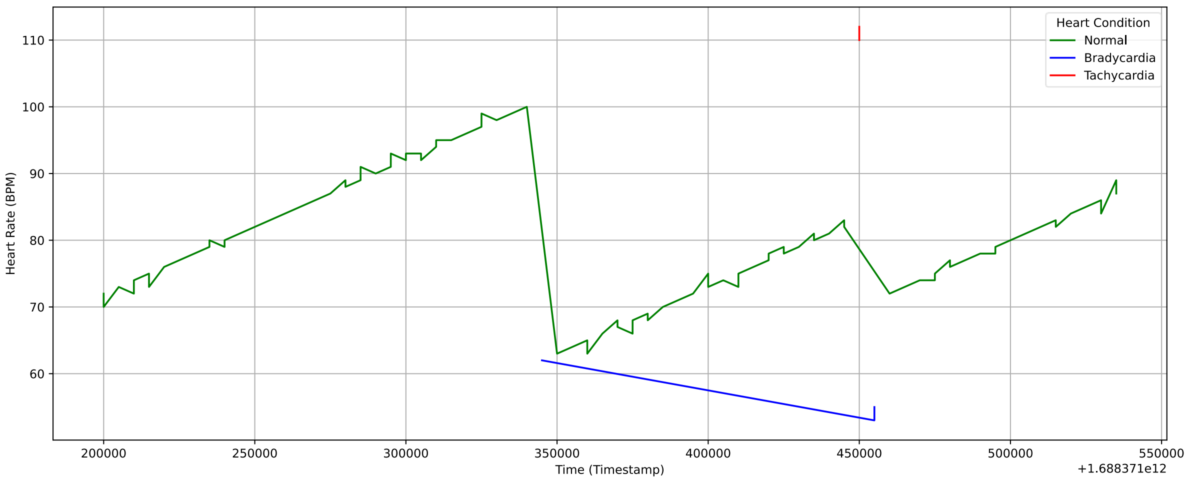

Figure 4. Graph of heartbeat classification results from prototype

Usability and comfort were also evaluated by respondents, covering aspects such as device weight, sensor attachment, screen clarity, and button operation. Respondents perceived the device as lightweight, portable, and easy to operate, with clear output on the LCD and helpful LED indicators. Its acrylic casing provided adequate protection, making it suitable for non-clinical, home-based use.

The classification logic output is automatically stored on the SD Card, enabling further verification and analysis. Figure 5 illustrates the classification output visualized as a time-series graph, which demonstrates the separation of normal, bradycardia, and tachycardia conditions during the recording period.

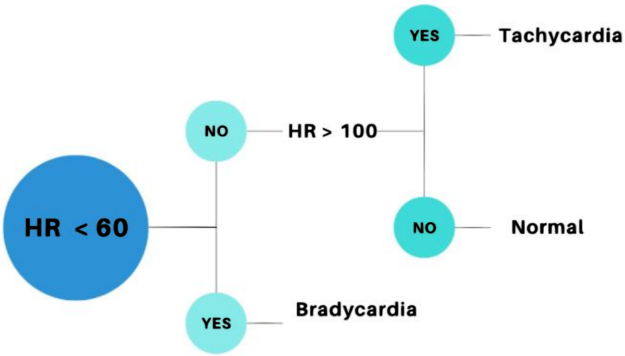

Figure 5. Simple classification logic diagram based on ECG signal values

The system demonstrated stable overall performance during testing, reading and processing sensor data in real-time, with results displayed and stored rapidly. Response time consistency, assessed over ten tests, ranged from 275 to 310 milliseconds, indicating quick, stable, and reliable communication between the sensor, microcontroller, and LCD, as shown in Figure 6, which illustrates the system response speed graph. This reliability supports its use for independent heart monitoring in daily conditions.

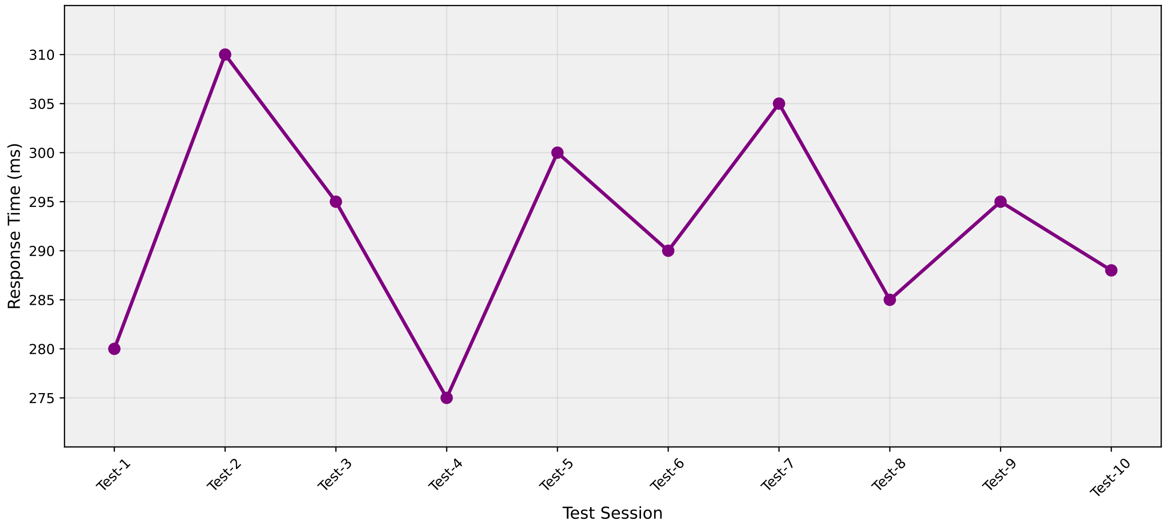

Figure 6. System response speed graph

Furthermore, significant attention was given to power consumption. The integration of a 3.7 V LiPo battery with a rechargeable TP4056 module allows for several hours of portable operation without frequent recharging. The measurement results, presented in Figure 7, show that power consumption during continuous testing was stable at 264–269 mA, confirming the prototype's good energy efficiency for extended portable use.

To validate the developed portable ECG system, its heartbeat classification results were rigorously compared with clinical reference data from the PTB-XL dataset (PhysioNet: https://physionet.org/content/ptb-xl/1.0.3/; Kaggle: PTBXL LSAD), after filtering 100 relevant data points. Using the Random Forest algorithm, the prototype's classification outputs were directly compared to PTB-XL reference labels. The level of accuracy was then determined using the standard machine learning evaluation formula shown in Equation (3).

$$Accuracy\ (\%) = \frac{Number\ of\ Correct\ Predictions}{Total\ Number\ of\ Predictions} \times 100 \tag{3}$$

where: Accuracy (%) = Percentage of correctly classified data
Number of Correct Predictions = Total correctly predicted classification outcomes
Total Predictions = Total number of all classification data tested

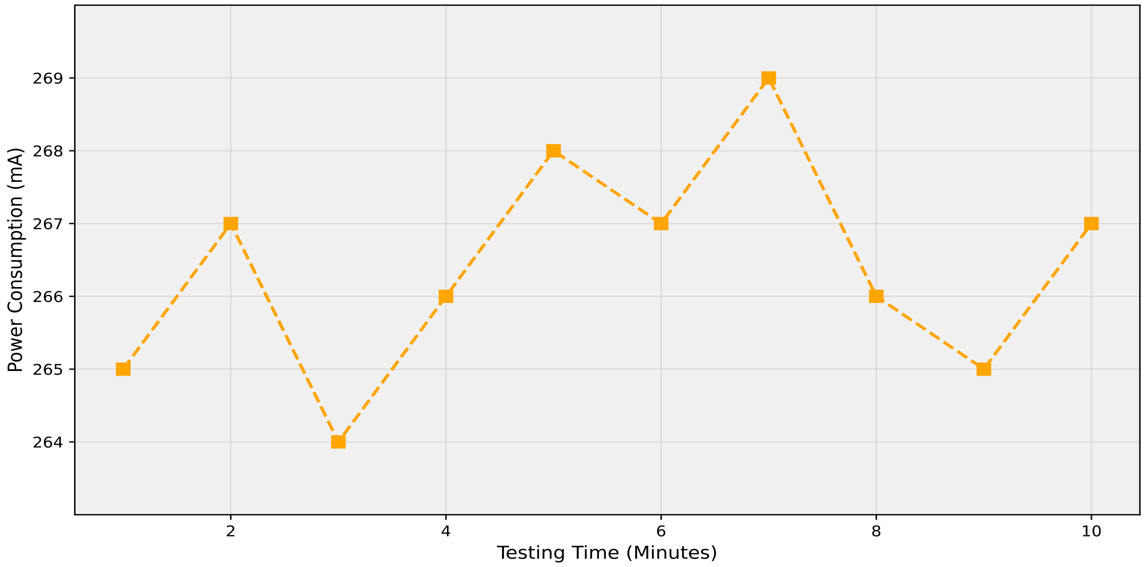

Figure 7. Prototype power consumption graph during testing

Visualization of classification results from both sources via Jupyter Notebook revealed similar waveform patterns, particularly at the R-wave peak. The prototype achieved a classification accuracy of 99.0%, closely matching the PTB-XL dataset's 98.0% accuracy. These compelling results underscore the prototype system's ability to approximate medical reference standards, affirming its suitability for independent heart monitoring, particularly in non-clinical settings. This high level of accuracy positions the prototype as a promising solution for accessible cardiac health oversight, offering significant potential for broader application in home-based healthcare, complementing existing clinical practices. The comparison between the classification results of the prototype and the PTB-XL dataset is shown in Figure 8.

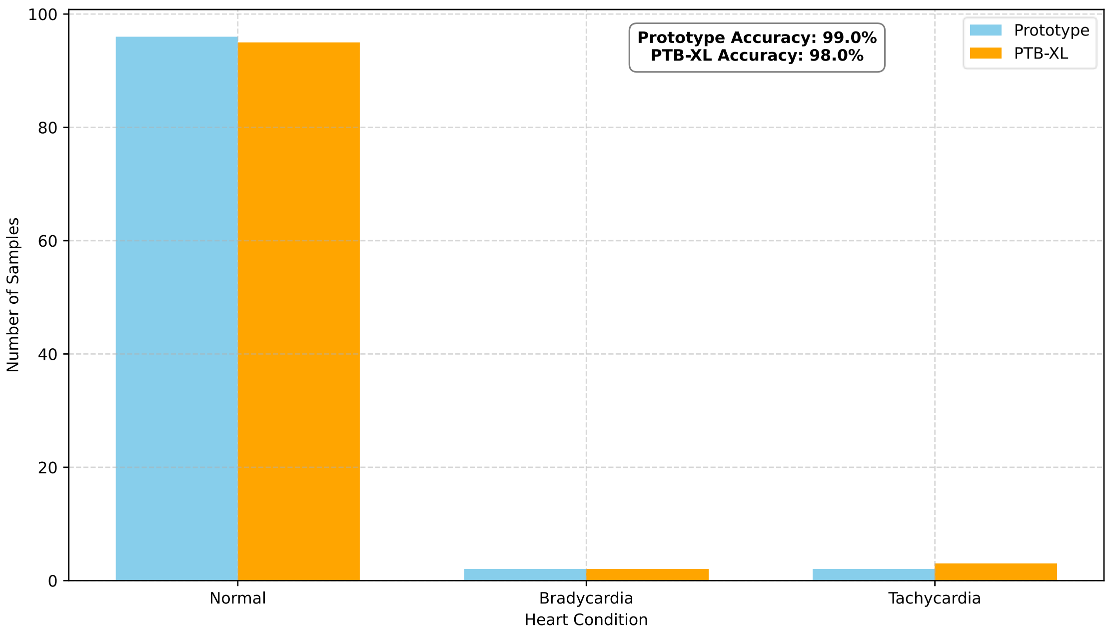

Figure 8. Graph of prototype and PTB-XL dataset classification comparison

Furthermore, when compared to prior studies, the developed prototype demonstrates clear strengths. Kim et al. (2023) demonstrated the feasibility of TinyML-based ECG classification directly on embedded microcontrollers for real-time, low-power operation [8]. Subba and Chingtham (2024) reported a 98% arrhythmia detection accuracy using Random Forest with advanced feature extraction on the MIT-BIH dataset [9]. Meanwhile, Prajitha et al. (2022) proposed a wearable cloud-based ECG monitoring system for smart city environments, which emphasized continuous connectivity [10]. Compared to these, the prototype achieved slightly higher accuracy (99%), likely due to static testing conditions, minimal motion artifacts, and a focused dataset that enhanced signal clarity. Unlike these studies, this system uniquely combines ultra-low-cost hardware and fully embedded classification logic without relying on constant internet or cloud services, highlighting its practicality for independent, offline home use. These similarities and distinctions confirm that the proposed approach aligns with current innovations in portable ECG monitoring while adding clear contributions through its cost-efficiency, real-time embedded processing, and user-friendly design for daily non-clinical use.

4. CONCLUSION

This research successfully designed and implemented an Arduino-based portable ECG prototype capable of independently reading and classifying heart signals. The system performs optimally when the user is seated and calm, aligning with the intended research scope. The acquired data are stored in .csv format and visualized using Jupyter Notebook for further analysis. The classification results, applying a simplified Random Forest algorithm, demonstrated high accuracy, with the prototype achieving 99.0%, slightly exceeding the 98.0% accuracy of the PTB-XL reference dataset. This result aligns with the performance reported in recent TinyML- and Random Forest-based ECG studies, such as Kim et al. (2023) [8], Subba and Chingtham (2024) [9], and Prajitha et al. (2022) [10], confirming the feasibility of integrating low-cost microcontroller hardware with reliable classification logic for practical, daily heart monitoring at home. The novelty of this prototype lies in its affordable sub-USD 31 component cost and its ease of use by non-expert users, offering a more accessible alternative to conventional hospital-based ECG equipment. While the prototype proves robust in static, controlled conditions, its accuracy decreases with significant user movement, highlighting an area for improvement. Therefore, future development should focus on enhancing signal stability during motion, adding wireless features such as Bluetooth or mobile app integration for real-time remote monitoring, and expanding testing with more diverse user groups to strengthen its practical reliability and impact for everyday home-based heart health monitoring.

REFERENCE

[1] A. Abdelrazik *et al.*, "Wearable Devices for Arrhythmia Detection: Advancements and Clinical Implications," *Sensors*, vol. 25, no. 9, pp. 1–28, 2025. https://doi.org/10.3390/s25092848

[2] J. Z. Metcalfe *et al.*, "Validation of a Handheld 6-Lead Device for QT Interval Monitoring in Resource-Limited Settings," *JAMA Netw. Open*, vol. 7, no. 6, p. E2415576, 2024. https://doi.org/10.1001/jamanetworkopen.2024.15576

[3] J. Medina-Avelino, R. Silva-Bustillos, and J. A. Holgado-Terriza, "Are Wearable ECG Devices Ready for Hospital at Home Application?," *Sensors*, vol. 25, no. 10, 2025. https://doi.org/10.3390/s25102982

[4] T. Moller, Y. Georgie, M. Voss, and L. Kaltwasser, "An Arduino Based Heartbeat Detection Device (ArdMob-ECG) for Real-Time ECG Analysis," *in Proc. IEEE Signal Process. Med. Biol. Symp.* (*SPMB*), pp. 1–26, 2022. https://doi.org/10.1109/SPMB55497.2022.10014819

[5] A. A. Yusuf, N.-N. Nnenna Harmony, D. P. Eze-Steven, and N. Charles N, "Design and Implementation of Portable Low-Cost Heart Rate Monitoring ECG System," *Eng. Technol. J.*, vol. 10, no. 01, 2025. https://doi.org/10.47191/etj/v10i01.05

[6] J. Heaney, J. Buick, M. U. Hadi, and N. Soin, "Internet of Things-Based ECG and Vitals Healthcare Monitoring System," *Micromachines*, vol. 13, no. 12, 2022. https://doi.org/10.3390/mi13122153

[7] S. Smigiel, K. Pałczyński, and D. Ledziński, "ECG Signal Classification Using Deep Learning Techniques Based on the PTB-XL Dataset," *Entropy*, vol. 23, no. 9, p. 1121, 2021. https://doi.org/10.3390/e23091121

[8] E. Kim, J. Kim, J. Park, H. Ko, and Y. Kyung, "TinyML-Based Classification in an ECG Monitoring Embedded System," *Comput. Mater. Contin.*, vol. 75, no. 1, pp. 1751–1764, 2023. https://doi.org/10.32604/cmc.2023.031663

[9] T. Subba and T. Chingtham, "Comparative Analysis of Machine Learning Algorithms With Advanced Feature Extraction for ECG Signal Classification," *IEEE Access*, vol. 12, no. March, pp. 57727–57740, 2024. https://doi.org/10.1109/ACCESS.2024.3387041

[10] C. Prajitha, K. P. Sridhar, and S. Baskar, "ECG diagnosis for arrhythmia detection with a cloud-based service and a wearable sensor network in a smart city environment," *Front. Sustain. Cities*, vol. 4, 2022. https://doi.org/10.3389/frsc.2022.1073486

[11] Y. Niu, H. Wang, H. Wang, H. Zhang, Z. Jin, and Y. Guo, "Diagnostic validation of smart wearable device embedded with single-lead electrocardiogram for arrhythmia detection," *Digit. Heal.*, vol. 9, 2023. https://doi.org/10.1177/20552076231198682

[12] M. Bravo-Zanoguera, D. Cuevas-González, J. P. García-Vázquez, R. L. Avitia, and M. A. Reyna, "Portable ECG System Design Using the AD8232 Microchip and Open-Source Platform," in *Proc. 6th Int. Electron. Conf. Sensors Appl.*, 2020, p. 49, 2020. https://doi.org/10.3390/ecsa-6-06584

[13] R. E. Cañón-Clavijo, C. E. Montenegro-Marin, P. A. Gaona-Garcia, and J. Ortiz-Guzmán, "IoT Based System for Heart Monitoring and Arrhythmia Detection Using Machine Learning," *J. Healthc. Eng.*, vol. 2023. https://doi.org/10.1155/2023/6401673

[14] K. Rjoob *et al.*, "Machine learning and the electrocardiogram over two decades: Time series and meta-

analysis of the algorithms, evaluation metrics and applications," *Artif. Intell. Med.*, vol. 132, p. 102381, Oct. 2022. https://doi.org/10.1016/j.artmed.2022.102381

[15] M. D. Nadeem, M. T. I. Ansari, P. Shekhar Pandey, A. Shadab, S. Kumar Raghuwanshi, and S. Kumar, "Recent advances of ECG monitoring and webserver health monitoring applications: A review," *Opt. Laser Technol.*, vol. 177, p. 111039, Oct. 2024. https://doi.org/10.1016/j.optlastec.2024.111039