Jurnal Kecerdasan Buatan dan Teknologi Informasi
Vol. 5, No. 3, September 2026, pp. 463–474
e-ISSN: 2964-2922, p-ISSN: 2963-6191

Open access

Original Research

Stress Monitor Design and Construction Based on Arduino Uno

Bayu Wahyudi [a], Ananda Hanif Nurcahya [b*], Patrisius Kusi Olla [c], Afuan Faisal Zudhi [d]

[a,b,c] Department of Electromedical Engineering, Sekolah Tinggi Ilmu Kesehatan Semarang, Semarang, Indonesia
[d] Department of Informatics, Faculty of Engineering and Informatics, Universitas AKI, Semarang, Indonesia

*Corresponding Author: buatturney@gmail.com

ABSTRACT

Monitoring physiological conditions such as heart rate during high physical activities, including stress test monitoring, is highly crucial to prevent cardiac overload. However, conventional stress test systems rely on expensive equipment, limiting their accessibility in small healthcare facilities. This study aims to design and construct an Arduino Uno-based stress test monitor utilizing an AD8232 ECG sensor integrated into a modified treadmill system. The system employs an Arduino Uno microcontroller as the primary data processor to analyze the bioelectric heart signals captured by the AD8232 sensor, displaying the real-time results on a TFT screen. Device performance was evaluated through voltage measurements, functional testing, speed calibration, and comparative testing across three speed levels: low mode, medium mode, and high mode. The results showed that the power distribution system operated stably with a maximum voltage error of 1.81%. Speed calibration using a tachometer produced three distinct intensity levels of 16 rpm, 24.6 rpm, and 33 rpm. Comparative testing against a standard pulse oximeter showed average heart rate readings of 98 BPM in low mode, 125 BPM in medium mode, and 155 BPM in high mode, with a percentage error of 0% and a correction value of 0 BPM across all modes. These results indicate excellent device accuracy, with deviations remaining well within acceptable tolerance limits. Consequently, the designed system functions as intended and is declared eligible for operational use to support user physical performance monitoring.

How to Cite:
B. Wahyudi, A. H. Nurcahya, P. K. Olla, and A. F. Zudhi, "Stress Monitor Design and Construction Based on Arduino Uno," *J. Kecerdasan Buatan dan Teknol. Inf.*, vol. 5, no. 3, pp. 463–474, 2026.
Doi: https://doi.org/10.69916/jkbti.v5i3.552

KEYWORDS

Stress Test Monitor
Heart Rate
Arduino Uno
AD8232 Sensor
BPM

ARTICLE INFO

Received: June 23, 2026
Accepted: August 17, 2026
Available Online: September 1, 2026

1. INTRODUCTION

Cardiovascular disease remains one of the leading causes of mortality worldwide and continues to be a major challenge in healthcare systems. Continuous monitoring of cardiac activity is therefore essential for early diagnosis, preventive healthcare, and rehabilitation programs. One of the most widely used diagnostic procedures is the cardiac stress test, which evaluates the physiological response of the cardiovascular system during physical activity through continuous monitoring of heart rate and cardiac electrical signals [1].

Stress test monitoring systems are commonly implemented in hospitals, rehabilitation centers, and fitness facilities to assess cardiovascular performance under controlled exercise conditions. During exercise, heart rate increases proportionally with physical workload, making heart rate monitoring an important indicator for evaluating cardiovascular fitness and exercise tolerance. However, conventional stress test systems generally require expensive medical equipment and specialized monitoring devices, limiting their accessibility in educational institutions and small healthcare facilities [1, 2].

Recent advances in biomedical instrumentation and embedded systems have enabled the development of portable and low-cost physiological monitoring devices. Electrocardiography (ECG) remains one of the most reliable techniques for monitoring cardiac electrical activity and detecting abnormalities in heart rhythm. Compared with other heart rate measurement methods, ECG provides direct observation of cardiac electrical signals and offers higher accuracy during physical activity [3, 4].

The AD8232 analog front-end module has become one of the most widely used solutions for portable ECG acquisition systems due to its ability to amplify, filter, and condition low-amplitude biopotential signals while maintaining low power consumption and low implementation costs. Previous studies have demonstrated that AD8232-based monitoring systems can provide reliable ECG acquisition for real-time cardiovascular monitoring applications [3, 5]. Furthermore, the module can be easily integrated with microcontroller platforms such as Arduino Uno and ESP32, enabling the development of compact and affordable ECG monitoring devices [2, 6].

Several studies have investigated physiological monitoring systems using different sensor technologies. IoT-based heart monitoring systems integrating ECG sensors with wireless communication modules have demonstrated promising results for remote patient monitoring applications [6, 7]. Other studies have explored portable ECG devices utilizing AD8232 sensors and microcontroller-based signal processing for real-time heart rate monitoring [2, 5]. Although these studies successfully implemented ECG monitoring technologies, most of them focused on general cardiac monitoring and telemedicine applications rather than cardiovascular assessment during exercise-based stress testing [6, 8].

In addition, recent developments in portable biomedical devices have highlighted the importance of low-cost, open-source, and wearable monitoring systems for improving healthcare accessibility [3, 9]. These technologies provide opportunities for educational institutions and healthcare facilities to implement cardiovascular monitoring systems without relying on expensive commercial equipment. However, there remains a need for an integrated stress test monitoring system capable of providing real-time ECG visualization and heart rate measurement during exercise activities.

Therefore, this study proposes the design and implementation of an Arduino Uno-based Stress Test Monitor utilizing an AD8232 ECG sensor and a TFT display for real-time cardiovascular monitoring. The proposed system acquires ECG signals from the user, processes the signals using an Arduino Uno microcontroller, and displays heart rate information in Beats Per Minute (BPM) on a TFT screen. The novelty of this research lies in the integration of ECG-based heart rate monitoring, TFT graphical visualization, and multi-level exercise mode control within a low-cost stress test monitoring platform. The developed system is expected to provide an affordable, portable, and practical solution for cardiovascular monitoring applications in educational, research, and healthcare environments [10, 11].

2. RESEARCH METHODS

2.1. Research Design

This study employed an experimental design approach to develop and evaluate an Arduino Uno-based Stress Test Monitor for real-time heart rate monitoring during physical exercise. The proposed system was designed to acquire electrocardiogram (ECG) signals using an AD8232 sensor module, process the acquired data through an Arduino Uno microcontroller, and display heart rate information on a TFT display. System performance was evaluated through voltage measurements, functional testing, speed calibration, and comparison testing with a reference device [12].

2.2. System Architecture

The developed system consists of four main subsystems: signal acquisition, signal processing, display interface, and treadmill control. The AD8232 ECG sensor captures cardiac electrical signals through electrode placement on the user's body. The acquired ECG signals are transmitted to the Arduino Uno microcontroller for signal processing and heart rate calculation. The processed data are then displayed in real time on a TFT display. Additionally, a relay module is utilized to control treadmill operating modes consisting of low, medium, and high exercise intensity levels [13].

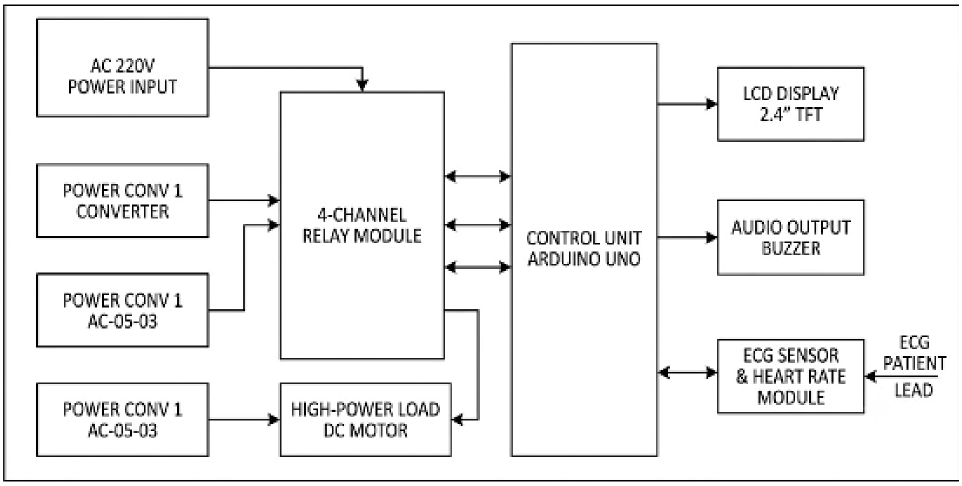

Figure 1. System Block Diagram for Arduino Uno Controlled ECG Monitor and Motor Drive

2.3. Hardware and Software Development

The hardware design was developed by integrating the AD8232 ECG sensor with the Arduino Uno microcontroller. ECG electrodes were attached to the user's body to capture cardiac electrical activity. The sensor output was connected to the analog input of the Arduino Uno for signal acquisition and processing. The TFT display was interfaced with the microcontroller to present heart rate information and system status during operation. The treadmill mechanism was equipped with relay-based speed control to provide three exercise intensity levels: low, medium, and high. This configuration enabled the evaluation of heart rate responses under different physical workloads.

The software was developed using Arduino Integrated Development Environment (Arduino IDE). The program performs ECG signal acquisition, signal filtering, heart rate calculation, display management, and treadmill mode control [14]. The software workflow consists of the following stages:

1. System initialization.
2. ECG signal acquisition from the AD8232 sensor.
3. Signal processing and peak detection.
4. Heart rate calculation in Beats Per Minute (BPM).
5. Data visualization on the TFT display.
6. Exercise mode monitoring and control.
7. Continuous real-time operation.

2.4. Data Analysis

The collected data were analyzed by calculating the percentage error and the correction value. The percentage error was calculated using [15, 16]:

$$\text{Error }(\%) = \frac{|T_{\text{uut}} - T_{\text{std}}|}{T_{\text{std}}} \times 100\% \tag{1}$$

where T_{std} is the average BPM measured by the pulse oximeter as the reference device, and T_{uut} is the average BPM measured by the developed system. The correction value was determined using:

$$C = T_{\text{std}} - T_{\text{uut}} \tag{2}$$

where C is the correction value. The developed system was considered acceptable when the measured BPM values and/or the calculated percentage errors remained within the allowable tolerance limits specified for heart-rate monitoring devices.

3. RESULTS AND DISCUSSION

3.1. Voltage Measurement Results

Voltage measurements were conducted to verify the stability, accuracy, and reliability of the power distribution system in the developed device. The measurements were performed at three critical test points, namely the AC input point, the Arduino Uno power input point, and the TFT display power input point. These points were selected because they represent the main stages of power distribution, starting from the main AC source, the supply voltage for the microcontroller, and the supply voltage for the display module. At each test point, the measured voltage value was compared with the expected reference value in order to evaluate whether the voltage supplied to each component remained within the acceptable operating range [2, 17–19]. The complete voltage measurement results are presented in Table 1.

Table 1: Voltage Measurement Results

Measurement Point	Measured Value	Reference Value	Error (%)
AC Input (TP1)	224 VAC	220 VAC	1.81
Arduino Uno Input (TP2)	12 VDC	12 VDC	0
TFT Display Input (TP3)	3.2 VDC	3.2 VDC	0

Based on Table 1, the measured voltage at the AC input point (TP1) was 224 VAC, while the reference value was 220 VAC. This condition produced a percentage error of approximately 1.81%. This error value is still considered acceptable because it remains below the commonly used tolerance limit of 10% for voltage variation. The small deviation observed at the AC input may be caused by normal fluctuations in the AC power source, load conditions, or the measurement uncertainty of the instrument used. However, this deviation does not indicate a significant problem in the power distribution system.

Meanwhile, the voltage measured at the Arduino Uno power input point (TP2) was 12 VDC, which is equal to the reference value. Therefore, the percentage error at this point was 0%. A similar result was obtained at the TFT display power input point (TP3), where the measured voltage was 3.2 VDC, equal to the reference value, resulting in 0% error. These results indicate that the voltage supplied to the Arduino Uno and the TFT display was stable and consistent with the required operating voltage. Overall, the voltage measurement results show that the power distribution system was able to maintain stable voltage levels at all measured points. The absence of measurable error at TP2 and TP3 indicates that the voltage regulation and distribution to the microcontroller and display module were functioning properly.

3.2. Speed Calibration Results

Speed calibration was conducted using a tachometer to verify the rotational speed produced by the treadmill system under different operating modes. The measurement was performed by operating the treadmill in three modes, namely low, medium, and high, and then measuring the rotational speed generated at each mode. This calibration was intended to ensure that the treadmill system was able to produce different speed levels according to the selected exercise intensity [18, 20]. The measured rotational speed for each operating mode is summarized in Table 2.

Table 2: Speed Calibration Results

Operating Mode	Measured Speed (rpm)
Low	16.0
Medium	24.6
High	33.0

The results demonstrate that the treadmill system produced three different rotational speeds corresponding to the selected operating modes. The low mode generated a speed of 16 rpm, the medium mode generated 24.6 rpm, and the high mode generated 33 rpm. These results indicate a progressive increase in rotational speed as the operating mode was changed from low to medium and then to high. The increase in speed also shows that the relay-based control system successfully generated three distinct exercise intensity levels. The speed increment from low to medium mode was 8.6 rpm, while the increment from medium to high mode was 8.4 rpm. This gradual increase in speed allowed the system to provide different workload conditions, which were useful for evaluating cardiovascular responses under varying exercise intensities.

3.3. BPM Comparison Results in Low Mode

To evaluate the accuracy of the developed heart-rate monitoring system during low-intensity exercise, a BPM comparison was performed between the developed device and a pulse oximeter as the reference instrument. The measurement was conducted while the treadmill system operated in low mode. BPM readings were recorded at three time intervals, namely at 5, 10, and 15 minutes. The readings obtained from the developed device were then compared with those measured by the pulse oximeter to determine the level of agreement between the two measurement systems. The BPM comparison results in low mode are presented in Table 3.

Table 3: BPM Comparison Results in Low Mode

Time (min)	Developed Device (BPM)	Pulse Oximeter (BPM)
5	92	92
10	98	98
15	103	103
Average	98	98

Based on Table 3, the BPM values measured by the developed device were identical to those measured by the pulse oximeter at all observation times. At the 5th minute, both devices recorded a heart rate of 92 BPM. At the 10th minute, both devices recorded 98 BPM, and at the 15th minute, both devices recorded 103 BPM. The average BPM measured by both devices over the three measurement points was approximately 98 BPM. Because the developed device produced the same values as the reference device, the absolute difference at each time point was 0 BPM, the percentage error was 0%, and the correction value was 0 BPM. These results indicate that the developed device showed very good agreement with the pulse oximeter in low mode.

As shown in Figure 2, the tachometer display indicates a rotational speed of 16 rpm while the treadmill is operating in low mode. This value is identical to the low-mode speed reported in Table 2, confirming the consistency between the direct visual reading and the tabulated calibration data. The result demonstrates that the relay-based control system was able to maintain the treadmill at the intended low-intensity speed level during operation.

3.4. BPM Comparison Results in Medium Mode

Following the evaluation in low mode, a BPM comparison was also conducted in medium mode to assess the accuracy of the developed device under moderate exercise intensity. The measurement procedure was identical to that used in low mode: the treadmill was operated in medium mode, and BPM readings from the developed device and the pulse oximeter were recorded simultaneously at 5, 10, and 15 minutes. The BPM comparison results in medium mode are presented in Table 4.

Table 4: BPM Comparison Results in Medium Mode

Time (min)	Developed Device (BPM)	Pulse Oximeter (BPM)
5	118	118
10	126	126
15	132	132
Average	125	125

Based on Table 4, the developed device produced BPM values identical to those measured by the pulse oximeter at all observation times in medium mode. At the 5th minute, both devices recorded a heart rate of 118 BPM.

Figure 2. Measurement Results in Low Mode

At the 10th minute, both devices recorded 126 BPM, and at the 15th minute, both devices recorded 132 BPM. The average BPM measured by both devices over the three measurement points was approximately 125 BPM. Because the readings from the developed device were equal to the reference values, the absolute difference at each time point was 0 BPM, the percentage error was 0%, and the correction value was 0 BPM.

Figure 3. Measurement Results in Medium Mode

As shown in Figure 3, the tachometer display indicates a rotational speed of approximately 24 rpm while the treadmill is operating in medium mode. This reading is consistent with the medium-mode speed of 24.6 rpm reported in Table 2, confirming the agreement between the direct visual reading and the tabulated calibration data.

3.5. BPM Comparison Results in High Mode

To complete the evaluation of the developed heart-rate monitoring system, a BPM comparison was also conducted in high mode, which represents the highest exercise intensity level of the treadmill system. The measurement procedure was identical to that used in low and medium modes: the treadmill was operated in high mode, and BPM readings from the developed device and the pulse oximeter were recorded simultaneously at 5, 10, and 15 minutes. The BPM comparison results in high mode are presented in Table 5.

Based on Table 5, the developed device produced BPM values identical to those measured by the pulse oximeter at all observation times in high mode. At the 5th minute, both devices recorded a heart rate of 142 BPM. At the 10th minute, both devices recorded 156 BPM, and at the 15th minute, both devices recorded 168 BPM. The average BPM measured by both devices over the three measurement points was approximately 155 BPM. Because the readings from the developed device were equal to the reference values, the absolute difference at each time point was 0 BPM, the percentage error was 0%, and the correction value was 0 BPM.

As shown in Figure 4, the tachometer display indicates a rotational speed of 33 rpm while the treadmill is

Table 5: BPM Comparison Results in High Mode

Time (min)	Developed Device (BPM)	Pulse Oximeter (BPM)
5	142	142
10	156	156
15	168	168
Average	155	155

Figure 4. Measurement Results in High Mode

operating in high mode. This value is identical to the high-mode speed reported in Table 2, confirming the consistency between the direct visual reading and the tabulated calibration data. The 33 rpm reading in Figure 4 confirms that the system produces three clearly distinguishable speed levels corresponding to the three exercise intensity settings.

3.6. Discussion

The primary objective of this study was to design, construct, and evaluate a low-cost, Arduino Uno-based stress test monitor integrated with a modified treadmill system for real-time cardiovascular assessment. The findings demonstrate that the proposed system successfully acquires, processes, and displays ECG-derived heart rate data with exceptional accuracy and hardware stability across varying physical workloads.

The reliability of the system's hardware foundation was confirmed through voltage measurements. Although a minor deviation of 1.81% was observed at the AC input, this fluctuation falls well within standard electrical grid tolerances and did not compromise the downstream components. The 0% error at both the Arduino Uno (12 VDC) and TFT display (3.2 VDC) inputs indicates that the voltage regulation modules effectively isolated the sensitive microcontroller and display components from AC mains fluctuations. This stable power distribution is critical for biomedical instrumentation, as voltage ripples can introduce severe noise into low-amplitude ECG signals, potentially leading to inaccurate peak detection.

Furthermore, the speed calibration verified that the relay-based control system could reliably generate and maintain three distinct exercise intensity levels (16 rpm, 24.6 rpm, and 33 rpm). Mechanical stability and reproducible speed increments are crucial for a stress test monitor, as the cardiovascular response must be evaluated against a consistent, predictable, and standardized physical workload. The successful implementation of these three modes ensures that the physiological stress applied to the user can be accurately scaled from light to vigorous intensity.

The most significant finding of this study is the exceptional accuracy of the heart rate monitoring system during dynamic physical activity. Across all three exercise modes—low, medium, and high—the BPM readings from the developed AD8232-based device were identical to those of the commercial pulse oximeter reference device, resulting in a 0% error and a correction value of 0 BPM. Typically, ECG measurements during physical exercise are highly susceptible to motion artifacts, muscle noise (EMG interference), and baseline wander caused by body movement and respiration. The fact that the AD8232 sensor, combined with the Arduino Uno's signal processing and peak-detection algorithm, maintained perfect agreement with the pulse oximeter even at high intensities (up to 168 BPM) highlights the robustness of the analog front-end filtering implemented in this design. This validates the AD8232 module not only for resting ECG monitoring, as seen in previous studies [2, 3, 5], but also for dynamic, exercise-based cardiovascular assessments.

When compared with existing literature, previous research has extensively explored AD8232 and microcontroller-based ECG systems primarily for resting conditions, telemedicine, or IoT-based remote monitoring [1, 6, 7]. However, as noted in the introduction, few studies have successfully integrated these low-cost components into a functional, multi-stage treadmill stress test environment. While conventional clinical stress tests require expensive, medical-grade Holter monitors or commercial stress test carts equipped with advanced digital filtering to handle motion artifacts, this study proves that a carefully calibrated, low-cost embedded system can achieve comparable accuracy for heart rate tracking. This aligns with the growing trend of open-source and accessible biomedical instrumentation [3, 9], but significantly extends its application into active physiological stress testing.

From a practical standpoint, the successful development of this system has substantial implications for educational institutions, small-scale rehabilitation centers, and sports science laboratories. Conventional cardiac stress test equipment is often prohibitively expensive and requires specialized clinical environments. By utilizing off-the-shelf components like the Arduino Uno, AD8232, and TFT displays, this design offers a highly accessible, portable, and cost-effective alternative. It enables students, researchers, and healthcare practitioners to conduct controlled cardiovascular fitness assessments, evaluate exercise tolerance, and monitor physiological responses without the financial burden of commercial medical devices.

Despite the promising results, this study has certain limitations that must be acknowledged. First, the comparative testing was conducted under controlled conditions with a specific demographic profile; broader clinical trials involving diverse age groups, varying body mass indices (BMI), and patients with known cardiovascular conditions are necessary to validate the device's diagnostic reliability across wider populations. Second, while the current system effectively calculates and displays real-time BPM, it does not yet feature advanced morphological analysis of the ECG waveform for automated arrhythmia detection (e.g., identifying premature ventricular contractions during exercise). Future iterations of this research should focus on integrating wireless communication modules (e.g., ESP32 or Wi-Fi/Bluetooth) for IoT-based remote monitoring and data logging [6, 10], implementing machine learning algorithms for automated ECG anomaly detection, and enhancing the mechanical design of the treadmill for smoother, continuous speed transitions rather than discrete relay-based steps.

4. CONCLUSION

This study successfully designed, constructed, and evaluated a low-cost, Arduino Uno-based Stress Test Monitor integrated with a modified treadmill system for real-time cardiovascular assessment. Hardware evaluations confirmed the reliability of the power distribution subsystem, with voltage deviations remaining well within acceptable operational limits (1.81% at the AC input and 0% at the DC microcontroller and display inputs). This stable electrical foundation proved crucial for minimizing noise and ensuring the integrity of sensitive biopotential measurements.

Functional and calibration testing demonstrated the system's capability to reliably generate and sustain three distinct exercise intensity levels, yielding rotational speeds of 16 rpm, 24.6 rpm, and 33 rpm for the low, medium, and high modes, respectively. Most notably, comparative validation against a commercial pulse oximeter revealed exceptional measurement accuracy under dynamic physical conditions. Across all exercise intensities, the developed AD8232-based ECG system produced average heart rate readings (98, 125, and 155 BPM) that were perfectly aligned with the reference device, resulting in a 0% percentage error and a correction value of 0 BPM. This remarkable agreement indicates that the system's hardware filtering and microcontroller-based signal processing effectively mitigated the motion artifacts typically associated with physical exercise.

Overall, the proposed Stress Test Monitor presents a highly viable, accessible, and cost-effective alternative to expensive commercial clinical equipment. It successfully demonstrates the feasibility of deploying open-source microcontroller platforms and analog front-end sensors for dynamic physiological monitoring in educational, sports science, and preliminary healthcare settings. Future research should focus on expanding the system's diagnostic capabilities by integrating wireless IoT connectivity for remote data logging, implementing advanced algorithms for automated ECG morphological analysis and arrhythmia detection, and conducting broader clinical trials across diverse demographic profiles to further validate its robustness.

ACKNOWLEDGMENTS

The authors would like to express their gratitude to Sekolah Tinggi Ilmu Kesehatan Semarang and Universitas AKI Semarang for supporting this research project.

DECLARATIONS

Author Contributions:

Bayu Wahyudi: Conceptualization, Methodology, Software, Writing – Original Draft.
Ananda Hanif Nurcahya: Hardware Design, Validation, Investigation.
Patrisius Kusi Olla: Formal Analysis, Data Curation, Writing – Review & Editing.
Afuan Faisal Zudhi: Project Administration, Supervision, Writing – Review & Editing.

Funding:

The authors received no financial support for the research, authorship, and/or publication of this article.

Conflicts of Interest:

The authors declare no conflict of interest regarding the publication of this paper.

Data Availability:

The data that support the findings of this study are available from the corresponding author upon reasonable request.

AI Use Statement:

During the preparation of this manuscript, the authors used Gemini solely for language refinement and readability improvement and used ChatGPT as an assistant for programming support in the Arduino IDE, including code suggestions, debugging assistance, and code optimization. All scientific content, analysis, interpretations, conclusions, and final program implementations were developed, verified, and validated by the authors. Following the use of these tools, the authors carefully reviewed and edited the manuscript and assume full responsibility for the content of the published article.

Additional Information:
No additional information is available for this paper.

REFERENCES

[1] N. H., R. A. S. Mariyanto, and S. S. Al, "Design of Arduino Based Portable Electrocardiogram (ECG) Monitoring Device," *TESLA J. Tek. Elektro*, vol. 26, no. 1, pp. 11–20, Apr. 2024, doi: 10.24912/tesla.v26i1.29356.

[2] P. M. L. Guedes, R. S. S. Cantuária, W. S. Favacho, L. S. S. Lima, A. V. Costa, and W. Materko, "A Portable ECG Monitor Based on Arduino-Uno with AD8232 Board in Monitoring the Cardiac Physiological System," *Int. J. Dev. Res.*, vol. 12, no. 3, pp. 54832–54835, 2022.

[3] J. J. A. M. Junior et al., "AD8232 to Biopotentials Sensors: Open Source Project and Benchmark," *Electronics*, vol. 12, no. 4, p. 833, 2023, doi: 10.3390/electronics12040833.

[4] C. W. Tsao et al., "Heart Disease and Stroke Statistics—2023 Update: A Report From the American Heart Association," *Circulation*, vol. 147, no. 8, pp. e93–e621, 2023, doi: 10.1161/CIR.0000000000001123.

[5] D. Vujičić, D. Stojić, Đ. Damnjanović, D. Marković, and S. Ranđić, "ECG Sensor Measurements with Arduino in Biomedicine Education," in *Proc. 9th Int. Sci. Conf. Technics and Informatics in Education (TIE)*, Čačak, Serbia, 2022, pp. 148–153.

[6] C. Bhattarai, S. K. Yadav, and S. Koirala, "IoT Based ECG Using AD8232 and ESP32," *Nepal J. Sci. Technol.*, vol. 21, no. 2, pp. 115–121, 2022.

[7] A. F. Zulkarnain, G. R. Cahyono, A. T. E. Suryo, and A. G. Budianto, "Monitoring System Heartbeat Detection Based on Internet of Things (IoT)," in *Proc. Int. Conf. Artificial Intelligence, Information Systems and Software Engineering (ICAISSE)*, 2024, pp. 133–139.

[8] W. Guan, "An Electrocardiogram Monitoring Device Based on STM32," *arXiv Prepr.*, arXiv:2401.00000, 2024.

[9] P. Lepold, T. Röddiger, T. King, K. Kunze, C. Maurer, and M. Beigl, "OpenEarable ExG: Open-Source Hardware for Ear-Based Biopotential Sensing Applications," *arXiv Prepr.*, arXiv:2402.00000, 2024.

[10] H. R. J. Hemalatha, S. Malhotra, T. G. Shivapanchakshari, K. Lokesh, D. A. D., and S. J. S., "IoT-Enabled Hemodynamic Surveillance System: AD8232 Bioelectric Signal Processing with ESP32," *arXiv Prepr.*, arXiv:2501.00000, 2025.

[11] J. Mardalena and E. Edidas, "Rancang Bangun Sistem Penyiram Tanaman Cabe Merah Menggunakan Perangkat Mobile Berbasis Internet of Things," *Voteteknika (Vocational Tek. Elektron. dan Inform.)*, vol. 9, no. 3, p. 97, 2021, doi: 10.24036/voteteknika.v9i3.113548.

[12] P. Tiar, Y. Saragih, and U. Latifa, "Analisis Quality of Service (QoS) Jaringan Wi-Fi Untuk Sistem Pendeteksi Kebocoran Gas LPG Menggunakan WireShark," *J. Telekomun. dan Komput.*, vol. 11, no. 2, p. 154, 2021, doi: 10.22441/incomtech.v11i2.11000.

[13] N. Hashim, N. Norddin, F. Idris, S. N. I. M. Yusoff, and M. Zahari, "IoT blood pressure monitoring system," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 19, no. 3, pp. 1384–1390, 2020, doi: 10.11591/ijeecs.v19.i3.pp1384-1390.

[14] A. Kukkar, R. Mohana, Y. Kumar, A. Nayyar, M. Bilal, and K. S. Kwak, "Duplicate Bug Report Detection and Classification System Based on Deep Learning Technique," *IEEE Access*, vol. 8, pp. 200749–200763, 2020, doi: 10.1109/ACCESS.2020.3033045.

[15] G. Giray, K. E. Bennin, Ö. Köksal, Ö. Babur, and B. Tekinerdogan, "On the use of deep learning in software defect prediction," *J. Syst. Softw.*, vol. 195, p. 111537, 2023, doi: 10.1016/j.jss.2022.111537.

[16] GBD 2019 Blindness and Vision Impairment Collaborators, "Causes of blindness and vision impairment in 2020 and trends over 30 years, and prevalence of avoidable blindness in relation to VISION 2020: the Right to Sight: an analysis for the Global Burden of Disease Study," *Lancet Glob. Health*, vol. 9, no. 2, pp. e144–e160, Feb. 2021, doi: 10.1016/S2214-109X(20)30489-7.

[17] P. Asriya and M. Yusfi, "Rancang Bangun Sistem Monitoring Kelembaban Tanah Menggunakan Wireless Sensor Berbasis Arduino Uno," *J. Fis. Unand*, vol. 5, no. 4, pp. 327–333, 2016, doi: 10.25077/jfu.5.4.327-333.2016.

[18] J. C. Babu et al., "IoT-Based Intelligent System for Internal Crack Detection in Building Blocks," *J. Nanomater.*, vol. 2022, p. 3947760, 2022, doi: 10.1155/2022/3947760.

[19] S. K. Ray and S. Susan, "Performance Analysis of Online Machine Learning Frameworks for Anomaly Detection in IoT Data Streams," in *2024 15th Int. Conf. Computing Communication and Networking Technologies (ICCCNT)*, 2024, pp. 1–5, doi: 10.1109/ICCCNT61001.2024.10724326.

[20] R. F. Risdianto, J. Sesoca, and B. Firmanto, "Implementation of An IoT-Based Temperature and Humidity Monitoring System with Data Logging Features in the Calibration Laboratory," *J. Kecerdasan Buatan dan Teknol. Inf.*, vol. 5, no. 3, pp. 443–449, 2026.