International Journal of Information Technology & Computer Engineering

ISSN 2347–3657

Volume 14, Issue 1, 2026

Heart Pulse Monitoring Sensor With Notification Using Arduino

R Vyshnavi[1], Bandi. Bhavani[2], Kandi Harika[3], Kuppa Jahnavi[4]

[1]Assistant Professor, Department of Electronics and Communication Engineering, Bhoj Reddy Engineering College for Women, Hyderabad, India.
[2,3,4]B.Tech Students, Department of Electronics and Communication Engineering, Bhoj Reddy Engineering College for Women, Hyderabad, India.
bandibhavani27@gmail.com, harikakandi2005@gmail.com, kuppajahnavi2005@gmail.com

Abstract
The rapid advancement of healthcare technology has significantly increased the demand for continuous and real-time health monitoring systems. Cardiovascular diseases remain one of the leading causes of mortality worldwide, making heart rate monitoring a critical component of preventive healthcare. Conventional heart rate monitoring systems such as electrocardiograms (ECG) and hospital-based diagnostic devices are often expensive, bulky, and unsuitable for continuous personal monitoring. To address these limitations, this research presents the design and implementation of a low-cost, portable, and IoT-enabled Heart Pulse Monitoring and Notification System using the Arduino IDE development platform and the ESP8266 NodeMCU microcontroller.
The proposed system utilizes a photoplethysmography-based pulse sensor to detect blood volume changes and calculate Beats Per Minute (BPM). The measured data is displayed locally on an OLED screen and transmitted wirelessly to the Blynk cloud platform for remote monitoring. In abnormal conditions, such as tachycardia or bradycardia, automated email alerts are generated to notify caregivers or medical personnel. The system is optimized for low power consumption and compact design, making it suitable for wearable applications and remote healthcare monitoring. Experimental results demonstrate reliable real-time monitoring and effective alert generation, highlighting the feasibility of integrating embedded systems and IoT technologies in affordable healthcare solutions.

Keywords:
Heart Rate Monitoring, Pulse Sensor (HW-827), ESP8266 NodeMCU, IoT-Based Healthcare, Blynk Cloud, OLED Display, Real-Time Monitoring, BPM (Beats Per Minute), Email Notification System, Remote Patient Monitoring.

Introduction

In recent years, the emphasis on preventive healthcare and remote patient monitoring has grown significantly. Heart rate is one of the most fundamental vital signs used to evaluate cardiovascular health. Variations in heart rate can indicate stress, physical exertion, arrhythmias, tachycardia, bradycardia, and other cardiac abnormalities. Traditional monitoring methods rely on clinical instruments and manual examination, limiting accessibility for daily use.

Microcontroller-based systems have emerged as affordable alternatives for personal health tracking. Platforms such as Arduino enable rapid prototyping of biomedical devices due to their open-source architecture and wide hardware compatibility [1]. When combined with Wi-Fi-enabled microcontrollers like the ESP8266 NodeMCU [3], real-time cloud connectivity becomes feasible.

This project presents a Heart Pulse Monitoring and Notification System that integrates sensor technology, embedded processing, and IoT communication. The system detects heart pulses using photoplethysmography, calculates BPM, displays real-time results on an OLED display, and transmits data to a cloud platform for remote monitoring. If abnormal BPM values are detected (<60 BPM or >100 BPM), automated email alerts are sent through the Blynk IoT platform [2].

The proposed system is suitable for:

1. Elderly monitoring
2. Home healthcare
3. Sports performance tracking
4. Remote patient supervision
5. Academic and prototype development

Literature Review

Several researchers and technical platforms have contributed to the development of IoT-based health monitoring systems.

Arduino's official documentation [1] explains the flexibility of the Arduino IDE for embedded system programming and rapid hardware prototyping. The ESP8266 datasheet by Espressif Systems [3] highlights integrated TCP/IP stack support, enabling wireless IoT connectivity without additional hardware modules.

International Journal of Information Technology & Computer Engineering

ISSN 2347–3657

Volume 14, Issue 1, 2026

The Blynk Cloud documentation [2] describes IoT dashboards and event-based notifications, which form the backbone of remote monitoring systems. Raj and Trivedi [8] proposed an IoT-based health monitoring system using NodeMCU, demonstrating wireless transmission of physiological data to cloud servers. Their study emphasized affordability and scalability in healthcare applications.

The Smart Heartbeat Monitoring System using IoT published in IJIRCCE [16] demonstrated integration of pulse sensors with microcontrollers for remote monitoring, concluding that IoT systems significantly improve patient safety through real-time alerts. Research on wearable heart monitoring devices available on ScienceDirect [17] provides an overview of wearable sensor technology and highlights photoplethysmography (PPG) as a reliable non-invasive technique for heart rate detection.

SparkFun's Pulse Sensor Amped Guide [4] explains the working principle of PPG sensors and signal conditioning methods for accurate pulse detection. Electronics Hub [12] and Circuit Digest [10] demonstrated practical implementations of pulse monitoring systems using NodeMCU and Blynk, showing successful cloud-based visualization.

ThingSpeak documentation [9] describes IoT data logging systems that allow long-term storage and analysis of biomedical data. Hackster.io [13] and TutorialsPoint [6] further illustrate practical IoT healthcare projects integrating wireless alerts and mobile dashboards.

Research from All About Circuits [19] discusses filtering techniques to reduce noise in pulse sensor outputs, emphasizing the importance of signal smoothing algorithms for accurate BPM calculation. Electronics For You [20] provides case studies on Arduino-based health monitoring prototypes suitable for rural healthcare.

From the literature, it is evident that IoT-enabled pulse monitoring systems are increasingly popular due to affordability, ease of integration, and scalability. However, many implementations lack integrated real-time email alert systems. The proposed work addresses this gap by combining real-time visualization with automated cloud-based notifications.

System Architecture and Methodology

The proposed system consists of three major modules:

1. Signal Acquisition Module
2. Processing and Display Module
3. IoT Communication and Notification Module

The signal acquisition module uses an HW-827 pulse sensor based on photoplethysmography. The sensor emits light into the skin and measures changes in light absorption corresponding to blood flow variations. These variations produce an analog voltage waveform proportional to heartbeats [4].

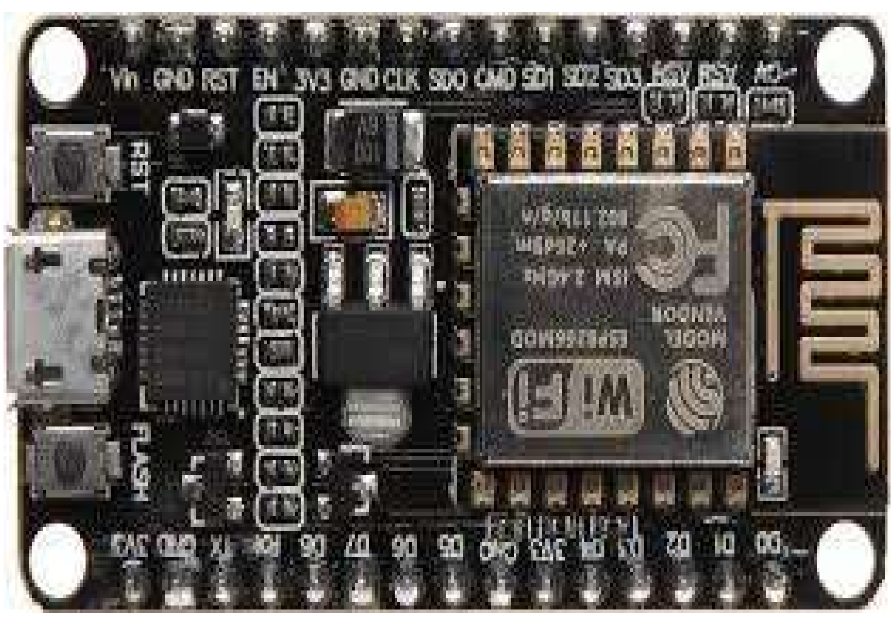

fig:1-ESP8266 NodeMCU

The processing module is built around the ESP8266 NodeMCU. The analog signal is read via the ADC pin (A0). The microcontroller processes the signal using a peak detection algorithm to identify heartbeat intervals and compute BPM using the formula:

BPM = 60 / (Time interval between beats)

To reduce noise caused by motion artifacts or ambient light interference, averaging and filtering techniques are applied [19].

The calculated BPM is displayed on a 128×64 OLED display via I2C communication. The OLED provides real-time visual feedback.

The IoT communication module connects the NodeMCU to a Wi-Fi network. Using the Blynk platform [2], BPM values are transmitted to a cloud dashboard. A threshold-based event detection algorithm checks whether BPM values fall outside the safe range:

1. BPM < 60 → Low Pulse (Bradycardia Alert)
2. BPM > 100 → High Pulse (Tachycardia Alert)

If either condition is met, the Blynk Email widget automatically sends a notification to a registered email address.

Hardware and Software Implementation

The hardware implementation includes the ESP8266 NodeMCU, HW-827 Pulse Sensor, OLED Display (128×64 I2C), breadboard, and jumper wires. The NodeMCU is powered via USB and operates at 3.3V logic levels [3].

Connections are as follows:

1. Pulse Sensor VCC → 3.3V
2. Pulse Sensor GND → GND
3. Pulse Sensor Signal → A0
4. OLED SDA → D2 (GPIO4)
5. OLED SCL → D1 (GPIO5)

The software is developed using the Arduino IDE [1]. Required libraries include ESP8266WiFi, BlynkSimpleEsp8266, Wire, and Adafruit_SSD1306. After compilation and uploading, the system continuously reads pulse data, calculates BPM, updates the OLED display, and sends data to the cloud.

Working Principle

The system operates on the principle of photoplethysmography (PPG). When the heart pumps blood, the volume of blood in capillaries changes, altering light absorption. The pulse sensor converts this variation into an analog signal.

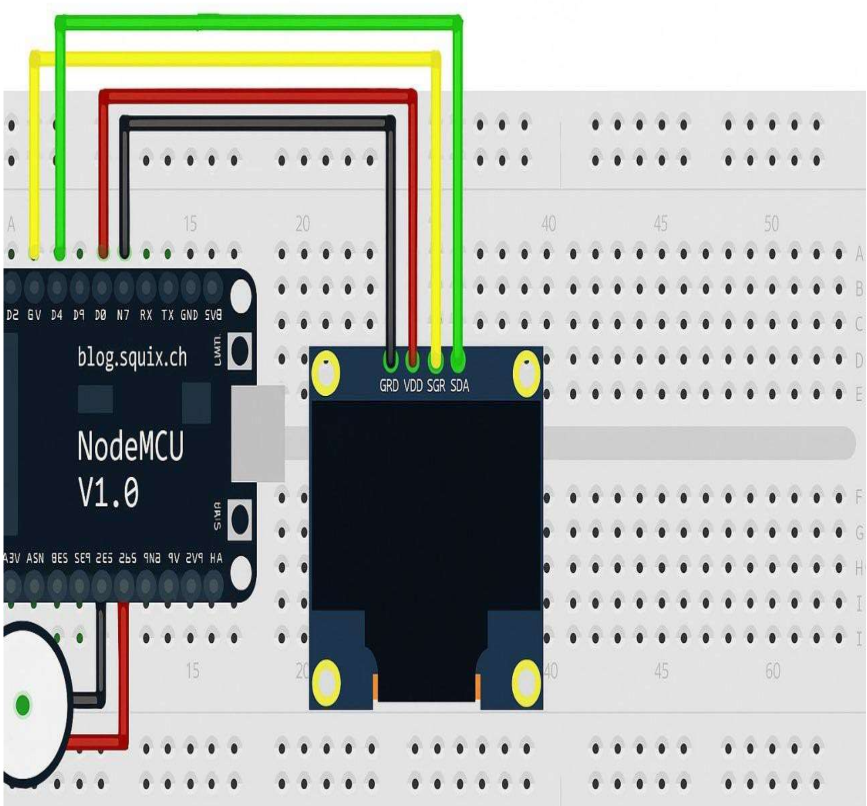

Fig:2-circuit diagram of Heart pulse Monitoring Sensor with Notification using Arduino

ISSN 2347–3657

Volume 14, Issue 1, 2026

The NodeMCU processes this signal, detects pulse peaks, and calculates BPM. The OLED screen displays the result in real time. Simultaneously, BPM data is transmitted via Wi-Fi to the Blynk cloud server [2]. If abnormal thresholds are crossed, automated email alerts are triggered.

This closed-loop system ensures continuous monitoring, local visualization, and remote notification.

The circuit is built around the **NodeMCU ESP8266**, which acts as the main controller of the heart pulse monitoring system. It is connected to a laptop through a USB cable for both power supply and code uploading using the Arduino IDE. After uploading the program, the NodeMCU reads real-time data from the pulse sensor and displays the calculated heart rate on the OLED screen.

The heart pulse sensor (HW-827) has three connections: VCC, GND, and Signal. The VCC is connected to the 3.3V pin of the NodeMCU, GND to the common ground, and the Signal pin to the analog input pin A0. The sensor works on the principle of photoplethysmography (PPG), where it emits light into the skin and measures variations in reflected light caused by blood flow changes during heartbeats. These variations are converted into an analog signal, which the NodeMCU processes to calculate Beats Per Minute (BPM).

An OLED display is used to show the BPM value. It communicates with the NodeMCU using the I2C protocol, requiring only two pins: SDA connected to D2 (GPIO4) and SCL connected to D1 (GPIO5). The OLED's VCC and GND are connected to the 3.3V and GND pins of the NodeMCU.

All components are assembled on a breadboard using jumper wires for easy prototyping without soldering. Once powered on, the system continuously reads pulse data, calculates BPM, and updates the display in real time. Since the NodeMCU has built-in Wi-Fi capability, the system can be extended to send heart rate data to IoT platforms such as Blynk or ThingSpeak for remote monitoring and notifications.

Results and Discussion

The system was tested under normal and abnormal heart rate conditions. The OLED successfully displayed real-time BPM values with minimal delay. In high pulse conditions (>100 BPM), the Blynk platform generated a "High Pulse" email alert. In low pulse conditions (<60 BPM), a "Low Pulse" notification was sent. The cloud dashboard displayed live BPM values ranging from 0–250 BPM.

The results of the proposed **IoT-based Heartbeat Monitoring System** demonstrate that the system successfully monitors heart rate in real time and generates alerts during abnormal conditions. The system integrates a pulse sensor, **ESP8266 NodeMCU**, OLED display, and the **Blynk** cloud platform for remote monitoring and notifications.

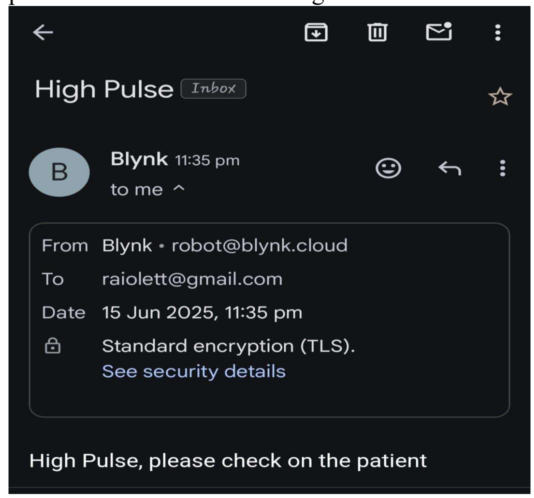

Fig:3-result of high pulse

Fig 3 High Pulse Alert When the measured heart rate exceeds the predefined threshold (e.g., 100 BPM), the system detects a **high pulse condition**. The NodeMCU processes the analog signal from the pulse sensor, calculates the BPM, and sends the data to the Blynk Cloud via Wi-Fi. An automatic email notification with the subject "High Pulse" is sent to the registered Gmail account. This confirms that the system can provide real-time emergency alerts, enabling quick medical attention in cases such as tachycardia or stress-induced high heart rate.

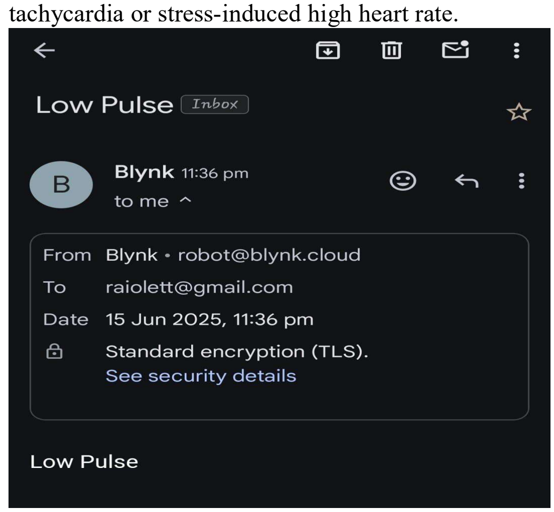

fig:4-result of low pulse

Fig 4 Low Pulse Alert When the BPM falls below the safe limit (e.g., 60 BPM), the system identifies it as a **low pulse condition**. The NodeMCU transmits the data to the Blynk platform, which triggers an automatic email alert with the subject "Low Pulse." This feature ensures that abnormal conditions such as bradycardia are immediately reported to caregivers or doctors, enhancing patient safety during remote monitoring.

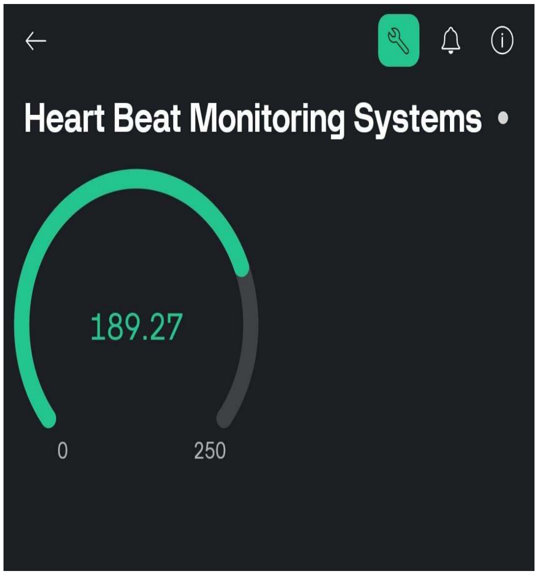

fig :5-heart rate in blynk app

Fig 5 – Heart Rate Display in Blynk App The Blynk mobile application displays live heart rate data using a gauge widget. The BPM value (e.g., 189.27 BPM) is updated in real time, allowing remote users to continuously monitor the patient's condition from anywhere. This confirms the reliability of Wi-Fi-based data transmission and cloud integration for IoT healthcare applications.

Fig:6-output pulse in OLED

Fig 6 Pulse Output on OLED Display The OLED display connected to the NodeMCU shows the real-time BPM locally. This provides instant on-device monitoring without requiring internet access. The display updates continuously, ensuring that the user or caretaker nearby can view the pulse rate directly.

The results confirm that the system performs accurate real-time pulse monitoring, successfully differentiates between normal and abnormal heart rate conditions, and generates immediate alerts through email notifications. The combination of local OLED display and remote monitoring through Blynk enhances the system's reliability and usability.

Thus, the project proves to be an effective, low-cost, and practical IoT-based healthcare monitoring solution suitable for elderly patients, remote care, and emergency health supervision.

The experimental results confirm:

1. Accurate detection of heart rate under stable conditions
2. Reliable Wi-Fi data transmission
3. Successful threshold-based email alert generation
4. Low power consumption suitable for portable applications

The system demonstrates effectiveness for non-clinical monitoring and early detection support.

Conclusion

The proposed Heart Pulse Monitoring Sensor with Notification using Arduino successfully integrates biomedical sensing, embedded processing, and IoT communication into a compact and cost-effective solution. By combining a pulse sensor with the ESP8266 NodeMCU and Blynk cloud platform, the system provides real-time monitoring, local display, and automated emergency notifications.

Although not intended as a replacement for clinical ECG equipment, the system offers a practical solution for preventive healthcare, elderly monitoring, and remote supervision. The affordability and scalability of Arduino-based systems make them ideal for widespread healthcare accessibility.

Future Scope

Future improvements may include:

1. Integration of SpO_2 and ECG sensors for multi-parameter monitoring
2. Cloud-based long-term data analytics using ThingSpeak [9]
3. AI-based anomaly prediction models
4. Mobile app development with encrypted data transmission
5. Wearable battery-powered design with deep sleep optimization
6. SMS and push notification support
7. GPS-enabled emergency tracking

With further enhancement, the system can evolve into a smart wearable health monitoring device capable of supporting modern telemedicine applications.

References

[1] Arduino. (n.d.). *Arduino IDE –Official Documentation*. Retrieved from https://www.arduino.cc/en/software

[2] Blynk IoT Platform. (n.d.). *Blynk Cloud Documentation*. Retrieved from https://docs.blynk.io

[3] Espressif Systems. (n.d.). *ESP8266 Datasheet and Technical Reference Manual*. Retrieved from https://www.espressif.com/en/products/socs/esp8266

[4] Spark Fun Electronics. (n.d.). *Pulse Sensor Amped – Hookup Guide*. Retrieved from https://learn.sparkfun.com/tutorials/pulse-sensor-amped-hookup-guide

[5] Last Minute Engineers. (n.d.). *Interfacing OLED Display with Arduino*. Retrieved from https://lastminuteengineers.com/oled-display-arduino-tutorial/

[6] Tutorials Point. (n.d.). *IoT Based Heartbeat Monitoring using Arduino*. Retrieved from https://www.tutorialspoint.com/heart-rate-monitor-using-arduino-and-pulse-sensor

[7] ResearchGate. (n.d.). *IoT Based Health Monitoring System Using NodeMCU and Pulse Sensor*. Retrieved from https://www.researchgate.net

[8] P. Raj, & M. C. Trivedi. (2021). *IoT Based Health Monitoring System Using NodeMCU. International Journal of Engineering Research & Technology (IJERT)*, Vol. 10 Issue 5.

[9] ThingSpeak. (n.d.). *IoT Data Logging Platform for Arduino and NodeMCU*. Retrieved from https://thingspeak.com

[10] Circuit Digest. (n.d.). *IoT Based Heart Rate Monitoring System using NodeMCU and Blynk*. Retrieved from https://circuitdigest.com

[11] GeeksforGeeks. (n.d.). *Introduction to IoT in Healthcare Applications*. Retrieved from https://www.geeksforgeeks.org

[12] Electronics Hub. (n.d.). *Pulse Rate Monitoring with Arduino and Pulse Sensor*. Retrieved from https://www.electronicshub.org/pulse-rate-monitoring-using-arduino/

[13] Hackster.io. (n.d.). *Remote Heart Rate Monitor using Blynk and NodeMCU*. Retrieved from https://www.hackster.io

[14] Maker Advisor. (n.d.). *Getting Started with OLED Displays on ESP8266*. Retrieved from https://makeradvisor.com

[15] GitHub. (n.d.). *Pulse Sensor Arduino Code Examples*. Retrieved from https://github.com/WorldFamousElectronics/PulseSensor_Amped_Arduino

[16] International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE). *Smart Heartbeat Monitoring System using IoT*, Vol. 8, Issue 4, 2020.

[17] ScienceDirect. (n.d.). *Wearable Heart Monitoring Devices: A Review*. Retrieved from https://www.sciencedirect.com

[18] Medium.com. (n.d.). *Sending Gmail Alerts from Blynk*. Retrieved from https://medium.com

[19] All About Circuits. (n.d.). *Pulse Sensor Operation and Filtering Techniques*. Retrieved from https://www.allaboutcircuits.com

[20] Electronics For You. (n.d.). *Arduino-based Health Monitoring Systems*. Retrieved from https://www.electronicsforu.com