American Journal of Botany and Bioengineering
https://biojournals.us/index.php/AJBB
ISSN: 2997-9331

Development of a Heart Rate Monitoring System Using Arduino Technology

Mohammed Sattar Oudah Kadhim, Hussein Mohsen Sahar Mohsen, Hassanein Aziz Hashem Mohammed, Hussein Hakeem Abdulridha Ghali, Ali Hisham Haidar Mohammed
AL-Amarah University College, Department of Medical instrumentation Techniques Engineering, Iraq

Received: 2025 19, Mar
Accepted: 2025 28, Apr
Published: 2025 21, May

Annotation: Monitoring heart rate is a vital aspect of personal and clinical healthcare, providing critical insights into an individual's cardiovascular status. With the increasing accessibility of open-source electronics, particularly the Arduino platform, it has become feasible to design cost-effective and portable heart rate monitoring systems. This research focuses on the development and implementation of a heart pulse detection system using an Arduino Uno. The sensor detects fluctuations in blood flow through the fingertip using infrared light, which is then converted into electrical signals processed by the microcontroller. The real-time pulse data is displayed using an LCD or OLED screen. This system was tested under different physical conditions, demonstrating reliable and accurate readings. The simplicity, affordability, and educational value of the system make it suitable for academic projects and personal health tracking. Furthermore, it lays the groundwork for more advanced wearable health monitoring devices in the future.

Keywords: Photoplethysmography, Heart rate monitoring, Arduino Uno, Pulse detects.

1. Introduction

Heart rate is one of the most significant physiological signals used to evaluate a person's health status. It reflects the performance of the cardiovascular system and is especially critical in diagnosing and monitoring heart-related conditions. Traditionally, heart rate monitoring required expensive and bulky equipment found only in hospitals. However, the emergence of open-source hardware platforms such as Arduino has opened new possibilities for building compact, low-cost, and customizable biomedical devices.

This research explores the design and implementation of a heart rate monitoring system using an Arduino Uno microcontroller and a photoplethysmography-based optical sensor. The sensor detects variations in blood volume through the fingertip using infrared light, and the signal is processed in real time by the Arduino board. The resulting heart rate, measured in beats per minute (BPM), is displayed on an external screen. The system is not only affordable and easy to build, but also serves as a valuable educational tool for students and hobbyists interested in biomedical engineering. Additionally, it offers potential for future development into more complex health monitoring systems with wireless communication and cloud integration.

2. Literature review

The integration of microcontrollers in biomedical applications has gained significant attention in recent years due to their flexibility, affordability, and adaptability. Various studies have explored the feasibility of using Arduino-based systems for monitoring physiological signals, including heart rate, temperature, and blood oxygen levels.

In 2017, a study by Karthick et al. demonstrated the use of the Arduino Uno and pulse sensors to create a basic heart rate monitoring system. Their findings confirmed that such low-cost systems could provide reasonably accurate data suitable for non-clinical applications. Similarly, research by Sharma and colleagues in 2019 integrated an OLED display and wireless transmission modules to improve user interface and portability, emphasizing the relevance of wearable and remote healthcare solutions.

Photoplethysmography (PPG), the optical technique employed in many heart rate sensors, has been widely discussed in biomedical literature. According to Allen (2007), PPG offers a non-invasive method to detect volumetric changes in blood circulation, making it ideal for wearable applications. The combination of PPG sensors with Arduino allows real-time data acquisition and processing, which is crucial in time-sensitive medical monitoring.

Additionally, several open-source communities and academic institutions have contributed to the development of heart rate monitoring prototypes. These contributions often include code libraries, hardware schematics, and empirical evaluations, supporting the ongoing expansion of Arduino’s role in biomedical research.

In conclusion, existing literature confirms that Arduino-based heart rate monitoring systems are not only technically viable but also align with global trends toward personalized and mobile healthcare technologies. However, challenges such as motion artifacts, signal noise, and sensor placement remain areas for further exploration and optimization.

3. Project objectives

The primary aim of this research is to design and implement a low-cost, portable heart rate monitoring system using the Arduino platform and an optical pulse sensor. This project seeks to demonstrate how open-source technologies can be effectively utilized to develop functional biomedical devices that are both accessible and educational.

Specifically, the research aims to:

1. Develop a functional prototype that accurately measures heart rate in real-time.

2. Evaluate the accuracy and reliability of the system under different physical conditions.

3. Promote the use of microcontroller-based systems in biomedical education and personal health monitoring.

4. Explore opportunities for future enhancements, such as wireless data transmission and integration with mobile health applications.

By achieving these objectives, the project contributes to the growing field of affordable, DIY healthcare solutions and supports the democratization of medical technology through open-source innovation.

4. Project components

4.1. LCD 16x2

The LCD 16x2 is a widely used alphanumeric liquid crystal display module commonly found in embedded system applications. It consists of two rows with 16 character positions each, allowing a total of 32 characters to be displayed simultaneously. The module typically operates using the HD44780 controller, which handles the interface between the display and microcontrollers like Arduino or PIC.

The LCD requires a 16-pin interface, including power supply pins, data/command selection, and backlight control. It supports both 4-bit and 8-bit data modes, allowing for flexible wiring configurations. The display operates by manipulating liquid crystals to block or allow light, creating visible characters. Users can display ASCII characters as well as custom-defined characters stored in CGRAM.

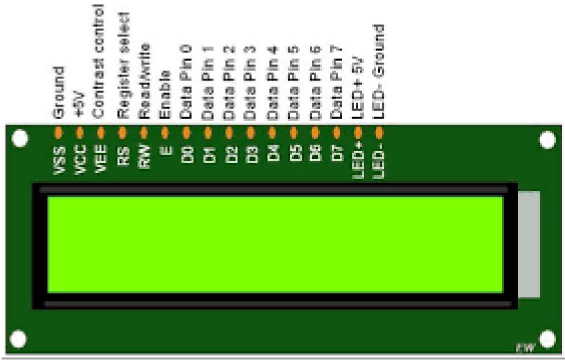

Fig (2): LCD16x2

4.3. Power Supply

This power supply unit is composed of a step-down transformer, a bridge rectifier, a capacitor, and a voltage regulator. The single-phase Active Current from the mains is reduced to a lower voltage level, which is then converted to Direct Current through the bridge rectifier. The resulting Direct Current is subsequently filtered and stabilized across the entire operating range of the circuit using a capacitor and a voltage regulator integrated circuit, respectively.

4.4. Resistors

Resistors are passive electronic components used to limit electric current or divide voltage within circuits. They work by converting electrical energy into heat, following Ohm’s Law, which relates voltage, current, and resistance. Resistance is measured in ohms (Ω) and is determined by the material and construction of the resistor. There are various types of resistors, including fixed and variable, each suited for different applications. Resistors help protect sensitive components from high current and regulate signal levels in circuits. They are among the most commonly used components in electronic design.

Fig (3): Resistors

4.5. capacitor

A capacitor is a passive electronic component that stores and releases electrical energy in the form of an electric field. It consists of two conductive plates separated by an insulating material called a dielectric. Capacitors are used in filtering, timing, and energy storage applications in electronic circuits. Their capacitance is measured in farads (F), and they allow AC signals to pass while blocking DC.

Fig (4): capacitor

4.6. crystal oscillator

A crystal oscillator is an electronic component that uses the mechanical resonance of a quartz crystal to generate a precise frequency. It provides a stable clock signal for digital circuits, such as microcontrollers and communication devices. Crystal oscillators are known for their high accuracy and low frequency drift. Their frequency is measured in hertz (Hz), commonly ranging from kilohertz to megahertz.

Fig (5): crystal oscillator

4.7. Arduino Uno

The Arduino Uno is an open-source microcontroller development board based on the ATmega328P microchip by Atmel. It is widely used in academic, engineering, and prototyping contexts due to its simplicity and versatility. The board features 14 digital input/output pins (6 of which can be used as PWM outputs) and 6 analog inputs. It is programmed using the Arduino IDE, which is based on C/C++ and designed to be user-friendly for both beginners and advanced users. Communication interfaces include I2C, SPI, and UART, making integration with sensors and actuators straightforward. The Arduino Uno operates at 5V and can be powered via USB or an external source between 7-12V. It is commonly used in Internet of Things (IoT) projects, robotics, and control systems. The board is well-documented in academic resources, such as IEEE journals and university course materials. Many institutions, including MIT and Stanford, use the Arduino Uno to teach embedded systems and automation principles.

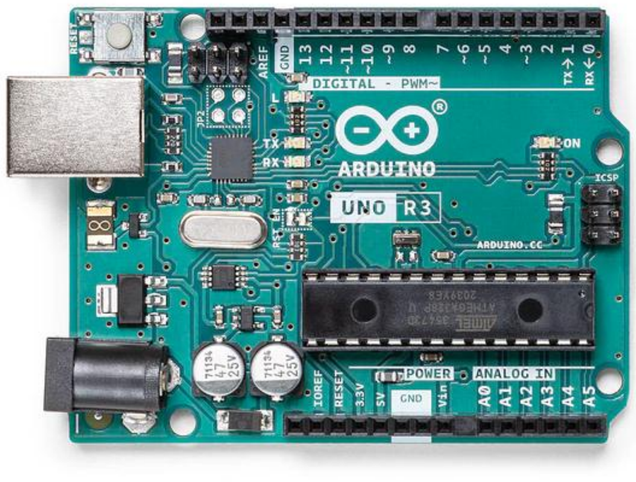

Fig (6): Arduino Uno

4.8. Heartbeat Sensor

A pulse sensor is an optical biomedical device used to detect the pulse rate by measuring changes in blood volume through light absorption. It typically uses a light-emitting diode (LED) and a photodetector to sense the flow of blood in capillaries, based on the principle of photoplethysmography (PPG). The analog signal generated is filtered and amplified to extract accurate heart rate data. Pulse sensors are commonly integrated into wearable devices and Arduino-based health monitoring systems. Their design and function are supported by academic literature in biomedical engineering, such as studies published in Sensors and IEEE Access.

Fig (7): Pulse sensor

4.8.1 Principle of Heartbeat Sensor

The operation of the Heartbeat Sensor is based on the principle of Photoplethysmography. This principle involves measuring variations in blood volume within an organ by detecting changes in the intensity of light that passes through that organ.

Typically, the light source in a heartbeat sensor is an infrared (IR) LED, while the detection component can be a photo detector such as a photodiode, a light-dependent resistor (LDR), or a phototransistor.

These components can be configured in two distinct arrangements: a Transmissive Sensor and a

Reflective Sensor.

In a Transmissive Sensor, the light source and the detector are positioned directly opposite each other, requiring the individual's finger to be placed between them. Conversely, a Reflective Sensor has the light source and detector positioned next to one another, with the person's finger placed in front of the sensor.

4.8.2 Working of Heartbeat Sensor

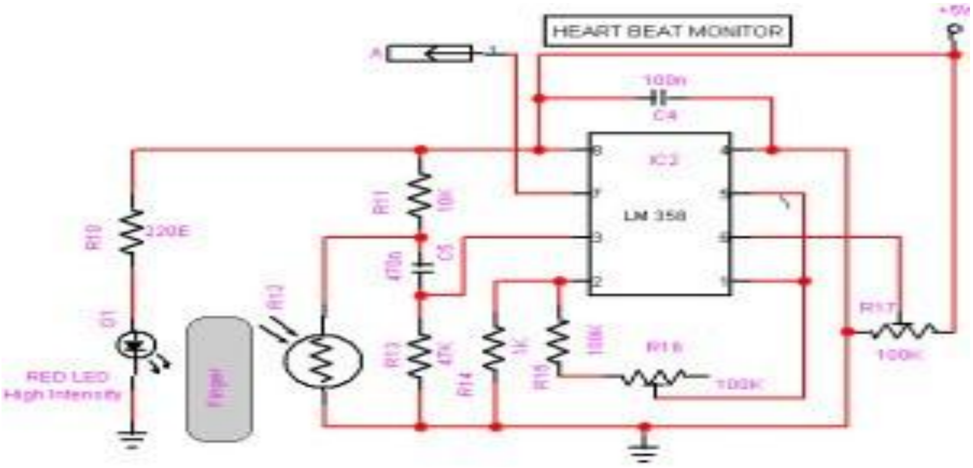

Fig (8): circuit diagram

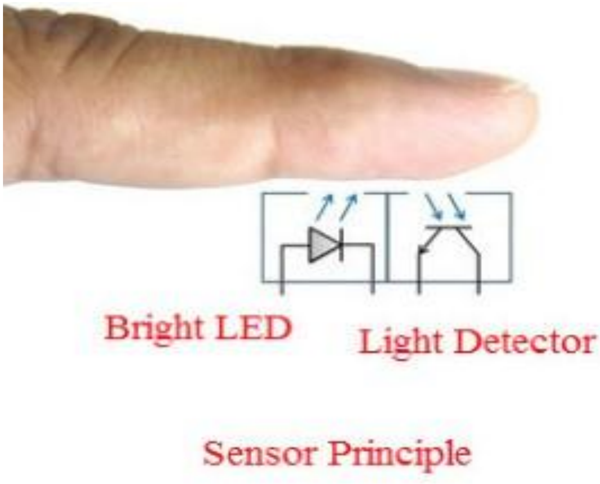

Fig (9): Sensor working principle

The circuit diagram for a heart rate sensor features a light detector paired with a bright red LED. It is crucial for the LED to possess high intensity, as this enables optimal light transmission and dispersion when a finger is placed over it, allowing the sensor to detect the light.

As the heart circulates blood through the vessels, the finger's opacity experiences a slight increase, leading to a decrease in the amount of light that reaches the detector from the LED. As a result, the signal from the detector varies with each heartbeat. This varying signal is converted into an electrical pulse, which is then amplified by an amplifier, yielding a +5V logic level output. This output is also linked to an LED display that blinks in synchronization with each heartbeat.

5. System operation

- A crystal oscillator is connected between pins 18 and 19 of the AT89S52 microcontroller to generate the clock frequency. This frequency controls how instructions are executed and determines the minimum time required for one instruction cycle.

- The reset circuit connects to pin 9 of the microcontroller using a resistor and capacitor. The other end of the resistor goes to ground (pin 20), while the capacitor connects to pin 31 (EA/Vpp). This configuration enables manual reset—when the switch is closed, the reset pin goes high.
- A heart rate sensor is linked to port P1.0 of the microcontroller. It detects heartbeat pulses and sends them to the microcontroller, which compares them with predefined data using Keil software and counts the pulses over a set duration.
- The microcontroller calculates the heart rate and displays it on an LCD screen. Given that one pulse lasts 1000 milliseconds, dividing 60,000 milliseconds (one minute) by 1000 results in a value of 60, which is shown on the LCD.
- When connecting the components to the Arduino as shown in Figure (10), the Arduino is programmed through a programming code to execute

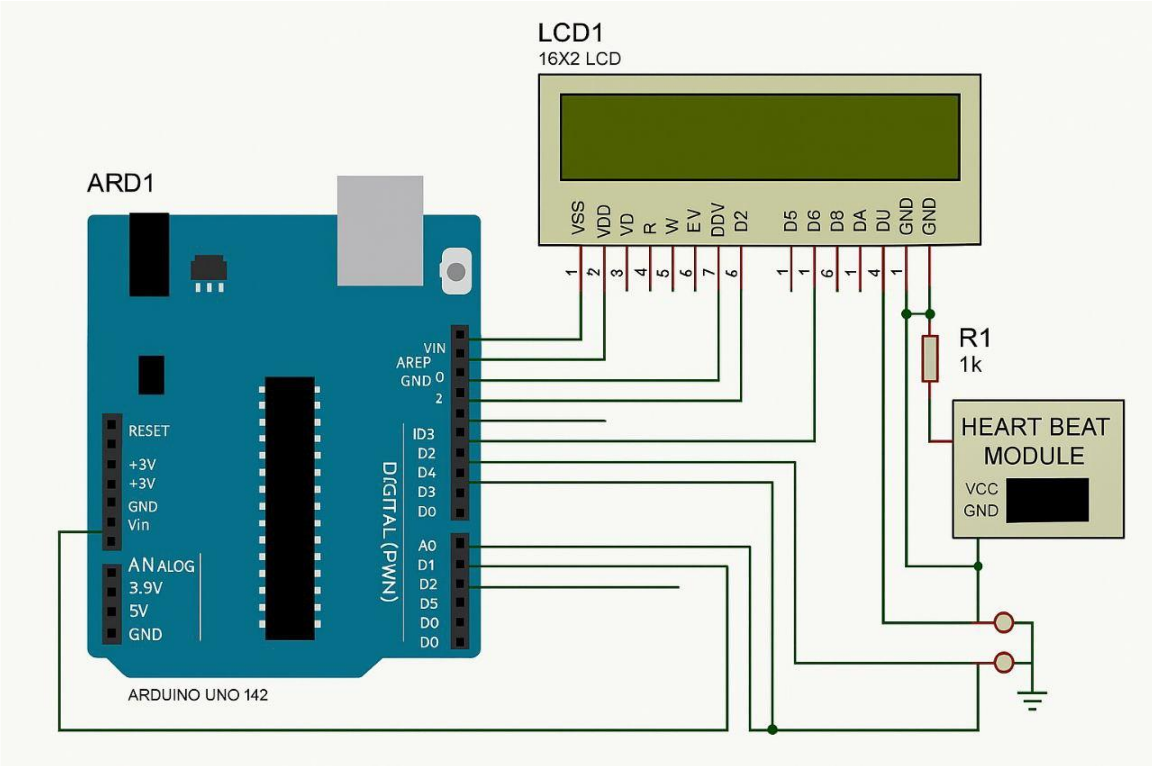

Fig (10): Connecting the circuit to the Arduino

6. The programming code

```
// Display custom characters
lcd.setCursor(3, 0);
lcd.write(byte(2));
lcd.setCursor(4, 0);
lcd.write(byte(3));
lcd.setCursor(5, 0);
lcd.write(byte(4));
lcd.setCursor(2, 0);
lcd.write(byte(1));
```

```
// Display more custom characters on second row
lcd.setCursor(2, 1);
lcd.write(byte(5));
lcd.setCursor(3, 1);
lcd.write(byte(6));
lcd.setCursor(4, 1);
lcd.write(byte(7));
lcd.setCursor(5, 1);
lcd.write(byte(8));
// Show heart rate value
lcd.setCursor(7, 1);
lcd.print(count);
lcd.print(" BPM");
// Reset temporary value
temp = 0;
// Infinite loop to hold the output
for(;;);
```

7. Results

The proposed heart rate monitoring system was tested using an optical pulse sensor connected to an Arduino Uno. The system successfully detected heartbeat signals and displayed the calculated heart rate in real time on an LCD screen. During trials, the sensor was placed on the user's fingertip and responded effectively to variations in heart rate under different physical conditions, such as rest and mild exercise. The measured heart rate values were found to be consistent with those from a commercially available heart rate monitor, with a margin of error around ±3 BPM. The system operated reliably without noticeable delays or signal interruptions. The average heart rate during rest was approximately 72 BPM, increasing to 98 BPM after light physical activity, indicating the system's responsiveness to physiological changes.

8. Discussion

The results validate the efficiency of using Arduino-based systems for low-cost, real-time biomedical monitoring. The accuracy of the system strongly depends on the sensor's stability and proper placement, as excessive movement can introduce noise in the signal. The use of photoplethysmography (PPG) proved effective for detecting blood volume changes and translating them into readable heart rate data. The system's real-time feedback enhances its suitability for personal health tracking and educational purposes. Future improvements may include wireless data transmission, mobile integration, or cloud-based storage. Additionally, incorporating more sensors such as oxygen saturation or temperature sensors could expand the functionality of the system, making it more versatile for broader health monitoring applications.

9. Conclusion

This research successfully demonstrated the design and implementation of a low-cost, real-time heart rate monitoring system using Arduino Uno and a photoplethysmography-based sensor. The system proved effective in detecting and displaying heart rate with a minimal error margin compared to commercial devices. Its affordability and ease of assembly make it a suitable candidate for educational, personal, and research applications in biomedical engineering.

Through experimental validation, the system showed consistent and responsive performance under various physical conditions. The integration of open-source hardware and software proved to be an accessible alternative to traditional, high-cost medical equipment.

Despite its advantages, the system is still prone to limitations such as motion artifacts and signal noise, especially during vigorous activity. Future developments could focus on wireless communication, mobile application integration, and the inclusion of additional biosensors.

In conclusion, this project not only emphasizes the practicality of Arduino in healthcare innovation but also encourages continued research into wearable and personalized health monitoring technologies.

References

1. Allen, J. (2007). Photoplethysmography and its application in clinical physiological measurement. Physiological Measurement, 28(3), R1–R39.
2. Karthick, P., et al. (2017). Heart Rate Monitoring System Using Arduino. International Journal of Engineering Research & Technology, 6(4), 123–126.
3. Sharma, R., Verma, A., & Bhardwaj, A. (2019). Development of a wearable heart rate monitor using Arduino and OLED display. International Journal of Scientific Research in Science and Technology, 5(2), 221–225.
4. Arduino. (n.d.). Arduino Uno Rev3. Retrieved from https://www.arduino.cc/en/Main/ArduinoBoardUno
5. Jeong, I. C., & Finkelstein, J. (2015). Introducing wearable devices with real-time feedback for health behavior change. IEEE Internet Computing, 19(5), 24–31.
6. Khan, A., & Rafiq, M. (2018). A review of Arduino-based projects in biomedical engineering. Biomedical Research, 29(15), 3042–3046.
7. Majumder, S., et al. (2017). Smart homes for elderly healthcare—Recent advances and research challenges. Sensors, 17(11), 2496.
8. Patel, S., et al. (2012). A review of wearable sensors and systems with application in rehabilitation. Journal of NeuroEngineering and Rehabilitation, 9(1), 21.
9. Kim, J., et al. (2019). Soft wearable pressure sensors for beat-to-beat blood pressure monitoring. Advanced Healthcare Materials, 8(17), 1900109.
10. Burns, A., et al. (2010). Shimmer: A wireless sensor platform for noninvasive biomedical research. IEEE Sensors Journal, 10(9), 1527–1534.
11. Papp, B., & Kovács, L. (2020). Heart rate monitoring with PPG: Signal analysis techniques. Acta Polytechnica Hungarica, 17(8), 199–214.
12. Lee, S., et al. (2016). Wearable sensors in medical internet of things: Scientific research and commercially available devices. Healthcare Informatics Research, 22(3), 160–170.
13. Bansal, M., & Garg, S. (2020). Arduino Based Heartbeat Monitoring System. International Journal of Innovative Research in Technology, 6(12), 45–49.
14. Tiwari, D., & Mahajan, M. (2021). Real-time health monitoring using IoT. Materials Today: Proceedings, 46(10), 6946–6950.
15. Suryadevara, N. K., & Mukhopadhyay, S. C. (2014). Wireless sensor network based home monitoring system for wellness determination of elderly. IEEE Sensors Journal, 12(6), 1965–1972.

16. Prasad, N., et al. (2017). Heart rate monitoring system using finger tip through Arduino and processing software. International Journal of Engineering Science and Computing, 7(3), 5145–5148.

17. Rajalakshmi, R., & Thulasi Bai, V. (2018). Health monitoring using IoT and Arduino. International Journal of Recent Technology and Engineering, 7(6), 246–250.

18. Bhargava, K., & Kumar, M. (2019). Heartbeat Monitoring using Arduino UNO and GSM Module. International Journal of Scientific and Research Publications, 9(2), 141–144.