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

Arduino Based Ecg & Heartbeat Monitoring Healthcare System

[1] Noufal Younis Noufal Qader, [2]Ali Mohammed Khader Ahmed,
[3] Rayan Ibrahim Hassan Ibrahim, [4]Iyad Faza Turki Al kanaan
[1, 2, 3, 4] Al-Kitab University, College, Department of Technology Engineering, Medical Devices, Iraq

Received: 2024 16, Nov
Accepted: 2024 17, Nov
Published: 2024 18, Dec

Abstract: The Arduino-based ECG and heartbeat monitoring healthcare system project aims to develop a low-cost and efficient solution for monitoring heartbeats and detecting anomalies in the ECG signal. The system uses an Arduino microcontroller and a heartbeat pulse sensor module to capture the ECG signal and gives the heart rate (BPM), which is then displayed in real-time on a ST7789 240 x 240 RGB TFT IPS LCD display module.

Signal processing techniques are applied to enhance and remove noise from the ECG signal.

The project has significant potential applications in the field of healthcare, including monitoring patients with heart conditions, tracking vital signs during medical procedures, and providing real-time feedback during rehabilitation programs. The low-cost nature of the project makes it an affordable option for low-income communities and developing countries.

1. Introduction

A heart rate monitor is a personal monitoring device that allows a subject to measure their heart rate in real time or record their heart rate for later study. Early models consisted of a monitoring box with a set of electrode leads that attached to the chest. The heart rate of a healthy adult at rest is around 72 beats per minute (bpm) & Babies at around 120 bpm, while older children have heart rates at around 90 bpm.

The heart rate rises gradually during gymnastics and returns slowly to the rest value after gymnastics. The rate when the pulse returns to normal is an indication of the fitness of the person. Lower than normal heart rates are usually an indication of a condition known as bradycardia, while higher is known as tachycardia.

Heart rate is simply measured by placing the thumb over the patient's arterial pulsation, and feeling, timing and counting the pulses usually in a 30 second period. Heart rate (bpm) of the subject is then found by multiplying the obtained number by

2. This method although simple, is not accurate and can give errors when the rate is high. More sophisticated methods to measure the heart rate utilize electronic techniques. Electro-cardiogram (ECG) is one of frequently used method for measuring the heart rate, but it is an expensive device. Low-cost devices in the form of wrist watches are also available for the instantaneous measurement of the heart rate. Such devices can give accurate measurements So, this heart rate monitor definitely a useful instrument in knowing the pulse and drawing ECG signal of the patient **[1]**.

2. Significance of Heart

The heart acts as a pump that circulates oxygen and nutrient carrying blood around the body in order to keep it functioning. When the body is exerted the rate at which the heart beats will vary proportional to the amount of effort being exerted. By detecting the voltage created by the beating of the heart, its rate can be easily observed and used for a number of health purposes. Heart pounded to pump oxygen rich blood to your muscles and to carry cell waste products away from your muscles. The heartrate gives a good indication during exercise routines of how effective that routine is improving your health.

Overall, the heart plays an essential role in maintaining the overall health and functioning of the human body, making it one of the most significant organs in the body **[1]**.

3. Why measuring heart beat is important?

Measuring heart rate is important because it provides valuable information about a person's cardiovascular health. The heart is a muscle that pumps blood throughout the body, and the rate at which it beats can give us an indication of how efficiently it is functioning.

Here are some reasons why measuring heart rate is important:

1. It can help detect heart problems: Measuring heart rate can help detect irregular heartbeats, also known as arrhythmias, which can be a sign of heart disease or other underlying health conditions.
2. It can help monitor exercise intensity: Measuring heart rate during exercise can help determine whether a person is working at the appropriate intensity for their fitness level and goals.
3. It can help monitor stress levels: Stress can have a significant impact on heart rate, and monitoring heart rate can help people identify when they are experiencing stress and take steps to manage it.
4. It can help monitor medication effectiveness: Heart rate monitoring can be used to evaluate the effectiveness of medications used to treat cardiovascular conditions, such as hypertension.
5. It can help track overall cardiovascular health: Consistently monitoring heart rate can provide a baseline for a person's cardiovascular health, allowing them to track changes over time and make adjustments to their lifestyle as needed.

Overall, measuring heart rate is a simple and non-invasive way to gather important information about a person's cardiovascular health and can help detect potential problems early on, leading to better health outcomes **[2]**.

4. Literature Review

"Real-time heart rate monitoring using Arduino and pulse sensor" by S. H. Yeom, et al. (2016) This study proposed a real-time heart rate monitoring system using an Arduino- based platform and a pulse sensor. The system was tested on 15 healthy subjects and was found to provide accurate and reliable heart rate readings.

- "Design and implementation of a low-cost heart rate monitoring system using Arduino" by K. E. Shaheen and M. S. Alam (2017)

This study developed a low-cost heart rate monitoring system using an Arduino- based platform and a heart rate sensor. The system was tested on healthy subjects and was found to provide reliable heart rate readings.

In conclusion, the use of Arduino-based heart rate monitoring systems in healthcare is a growing field of research. These systems are cost-effective, reliable, and can be used for continuous monitoring of patients. Further research is needed to improve the accuracy and efficiency of these systems and to explore their potential in different healthcare settings.

1. Methodology

Here is a methodology for building an Arduino Based Heartbeat Monitoring Healthcare System using Arduino Uno, heart beat sensor, and SPI OLED Display Module:

1. Identify the objectives of the system: In chapter (1), Paragraph (1.3) we determined the purpose of the system and the target users. Because of the importance of the heartbeat measurements the device should has accuracy and precision.
2. Select appropriate hardware components: In this chapter we'll explain in detail the items used in this project.
3. Connect the hardware components: In chapter (3) we'll explain how to connect the heart beat sensor to the Arduino Uno board using appropriate pins. And how to Connect the OLED display to the board using the SPI interface.
4. Develop the software program: In Appendix (A) we'll write our project code that reads the heart beat sensor data, process it, and display the heart rate on the OLED display by Using the appropriate libraries for the OLED display.
5. Test the system: In chapter (4) we'll explain how we tested the system to ensure that the heart beat sensor is accurately reading the heartbeat, and the OLED display is showing the correct heart rate readings.
6. Evaluate the system: Evaluate the system's accuracy and effectiveness in achieving the objectives of the project.
7. Optimize and refine the system: Based on the evaluation, refine the system to improve its performance, accuracy, and user-friendliness.
8. Deployment: Deploy the system in a healthcare setting for continuous monitoring of patients' heart rates.

2. Arduino Uno R3

Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards are able to read inputs light on a sensor, a finger on a button, or a Twitter message and turn it into an output activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so you use the Arduino programming language (based on Wiring), and the Arduino Software (IDE), based on Processing.

Over the years Arduino has been the brain of thousands of projects, from everyday objects to complex scientific instruments. A worldwide community of makers students, hobbyists, artists, programmers, and professionals has gathered around this open-source platform, their contributions have added up to an incredible amount of accessible knowledge that can be of great help to novices and experts alike.

Arduino was born at the Ivrea Interaction Design Institute as an easy tool for fast prototyping, aimed at students without a background in electronics and programming. As soon as it reached a wider community, the Arduino board started changing to adapt to new needs and challenges, differentiating its offer from simple 8-bit boards to products for Internet of Things (IoT) applications, wearable, 3D printing, and embedded environments. All Arduino boards are completely open source, empowering users to build them independently and eventually adapt them to their particular needs. The software, too, is open source, and it is growing through the contributions of users worldwide [3].

1. Arduino Uno Specifications

- Microcontroller: ATmega328
- Operating Voltage: 5V
- Input Voltage (recommended): 7-12V
- Input Voltage (limits): 6-20V
- Digital I/O Pins: 14 (of which 6 provide PWM output)
- Analog Input Pins: 6
- DC Current per I/O Pin: 40 mA
- DC Current for 3.3V Pin: 50 mA
- Flash Memory: 32 KB of which 0.5 KB used by bootloader
- SRAM: 2 KB(ATmega328)
- EEPROM: 1 KB (ATmega328)
- Clock Speed: 16 MHz

2. Arduino Uno Technology

A typical example of the Arduino board is Arduino Uno. It includes an ATmega328 microcontroller and it has 28-pins. The pin configuration of the Arduino Uno board is shown in Fig. 2.1.

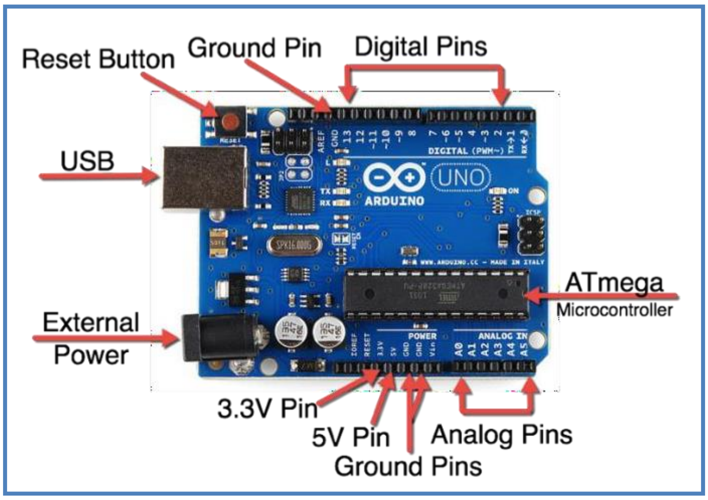

Figure (2.1) pin Configuration of Arduino Uno

It consists of 14-digital input-output pins, each pin takes up and provides 40mA current. Some of the pins have special functions like pins 0 & 1, which acts as a transmitter and receiver respectively for serial communication, pins 2 & 3 are external interrupts, 3,5,6,9,10,11 pins deliver PWM output and pin 13 is used to connect LED.

It has six analog input pins (A0-A5); each pin provides a 10-bit resolution. AREF pin is the pin that gives a reference to the analog inputs. And also has a USB connection, an DC power jack, a 16MHz crystal oscillator, a reset button, and an ICSP header.

Arduino board can be powered either from the personal computer through a USB or external source like a battery or an adaptor. This board can operate with an external supply of 7-12V by giving voltage reference through the IOREF pin or through the pin Vin.

3. Pulse Sensor

A pulse sensor shown in Fig.2.2. is a small electronic device that measures heart rate by detecting blood flow through the capillaries in the skin. It typically consists of a small sensor that is placed on the fingertip or earlobe, which shines a light through the skin and measures the amount of light that is absorbed by the blood as it flows through the vessels.

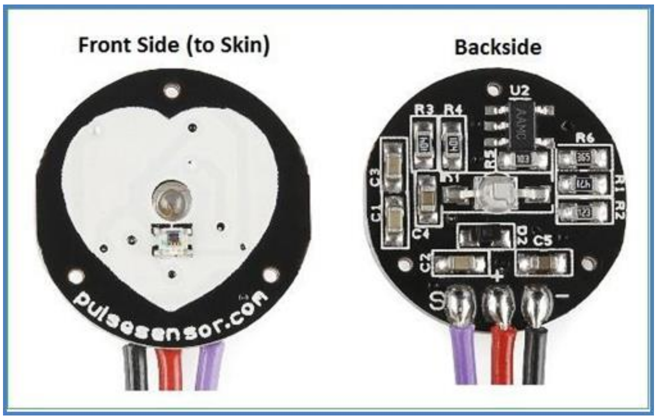

Figure (2.2) Pulse Sensor

1. Hardware Overview of Pulse Sensor

The front of the sensor, with the heart logo, is where you put your finger. You’ll also notice a tiny circular opening through which the Kingbright’s reverse mounted green LED shines.

Just beneath the circular opening is a small ambient light photo sensor – APDS-9008 from Avago. This sensor is similar to the ones used in cell phones, tablets, and laptops to adjust the screen’s brightness based on the ambient lighting conditions. As shown in Fig.2.3. below.

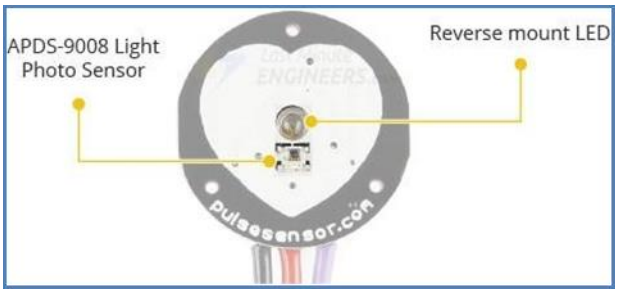

Figure (2.3) Front side (to skin) of pulse sensor

On the back of the module are an MCP6001 Op-Amp from Microchip and a few resistors and capacitors that make up the R/C filter network. Additionally, there is a reverse protection diode to prevent damage in the event that the power leads are accidentally reversed. As illustrated in Fig. 2.4. bellow.

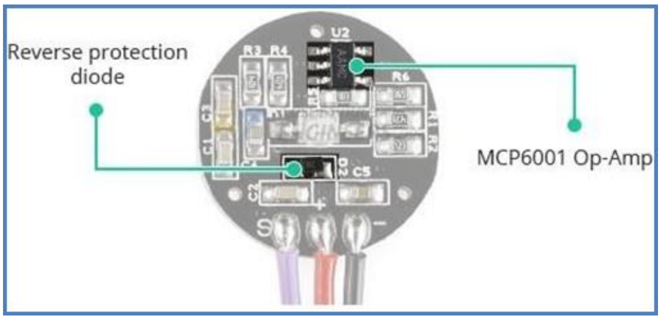

Figure (2.4) back side of pulse sensor

The module requires a DC power supply ranging from 3.3 to 5V and draws less than 4mA of current. Table (2.1) below illustrates technical specifications

Table (2.1): technical specifications of Pulse Sensor

Maximum Ratings	VCC	3.0 – 5.5V
	IMax (Maximum Current Draw)	< 4mA
	VOut (Output Voltage Range)	0.3V to Vcc
Wavelength	LED Output	565nm
	Sensor Input	525nm
Dimensions	L x W (PCB)	15.8mm (0.625″)
	Lead Length	20cm (7.8″)

2. How Does a Pulse Sensor Work?

The theory behind optical heart-rate sensors is very simple. If you've ever shined a flashlight through your fingers and observed your heartbeat pulsing, the concept of optical heart-rate pulse sensors can be easily grasped.

A pulse sensor, like any other optical heart-rate sensor, works by shining a green light (~ 550nm) on the finger and measuring the amount of reflected light with a photosensor. This optical pulse detection technique is known as a Photoplethysmogram (PPG). As shown in Fig. 2.5. below.

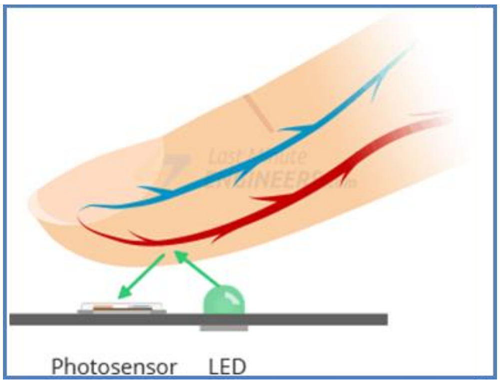

Figure (2.5) Photoplethysmogram Technique

The oxygenated hemoglobin in arterial blood has the property of absorbing green light. The redder the blood (the higher the hemoglobin), the greater the absorption of green light. With each heartbeat, blood is pumped through the finger, causing a change in the amount of reflected light, which in turn produces a waveform at the photosensor's output.

As you keep shining light and taking photosensor readings, you quickly begin to obtain a heart-beat pulse reading. As illustrated in Fig. (2.6) bellow.

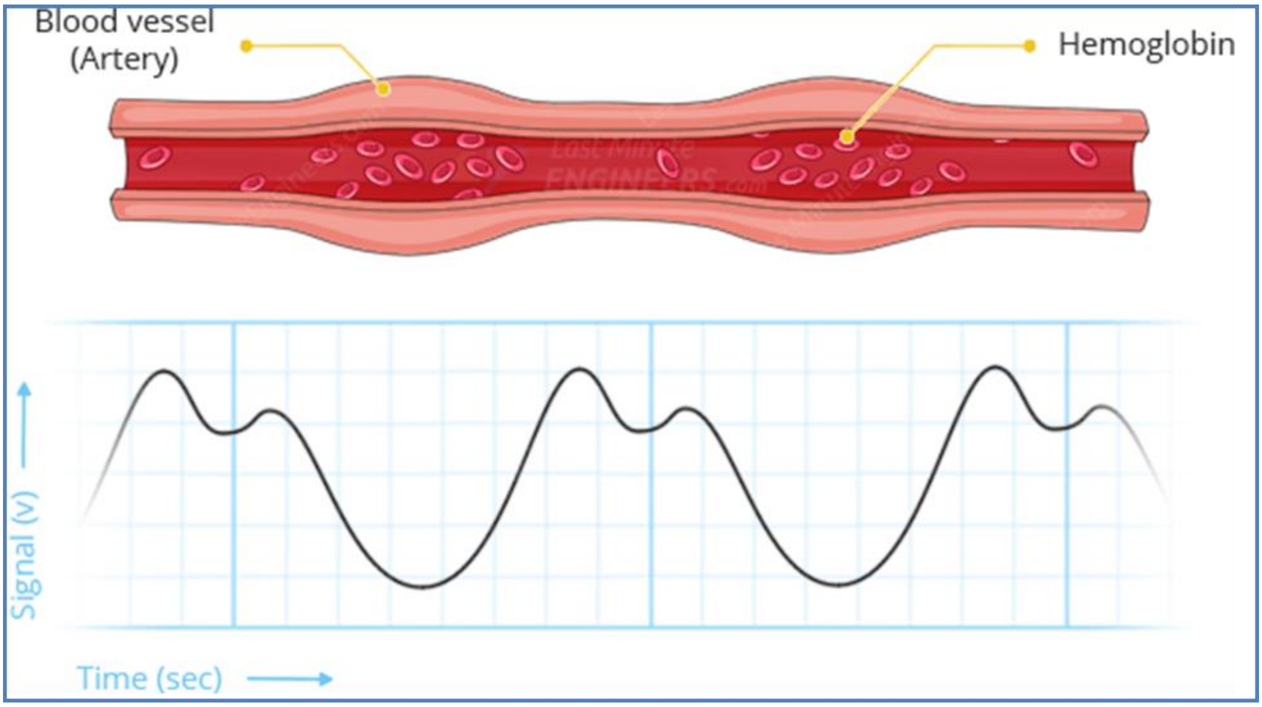

Figure (2.6) heart-beat pulse reading

This signal from the photosensor is typically small and noisy; therefore, it is passed through an R/C filter network and then amplified with an Op-Amp to create a signal that is significantly larger, cleaner, and easier to detect **[4]**.

4. RGB TFT IPS LCD display Module

The 1.3-inch 240x240 RGB TFT IPS LCD display is a small color LCD screen that measures 1.3 inches diagonally and has a resolution of 240x240 pixels. It uses an IPS (In-Plane Switching) panel, which provides wide viewing angles and good color reproduction. Here are some additional properties of this display:

- Display size: 1.3 inches
- Resolution: 240x240 pixels
- Color depth: 16-bit (65K colors)
- Display technology: TFT (Thin Film Transistor)
- Panel type: IPS (In-Plane Switching)
- Viewing angles: >160 degrees
- Interface: SPI (Serial Peripheral Interface)
- Backlight: White LED
- Operating temperature: -20 to 70 degrees Celsius

This type of display, shown in Fig. (2.7), is commonly used in small electronic devices, such as smartwatches, fitness trackers, and other wearables. Its compact size and high-resolution display make it an ideal choice for applications where space is limited but a clear and vibrant display is still required. The display uses the ST7789 driver **[5]**.

Figure (2.7) 1.3-inch 240x240 RGB TFT IPS LCD display

RESULTS AND DISCUSSION

1. Interfacing a Pulse Sensor with Arduino

Connecting the Pulse Sensor to an Arduino is a breeze. You only need to connect three wires: two for power and one for reading the sensor value.

The module can be supplied with either 3.3V or 5V. Positive voltage is connected to '+,' while ground is connected to '-.' The third 'S' wire is the analog signal output from the sensor, which will be connected to the Arduino's A0 analog input. Fig. (3.1) illustrates the wiring diagram for the Pulse Sensor experiments **[4]**.

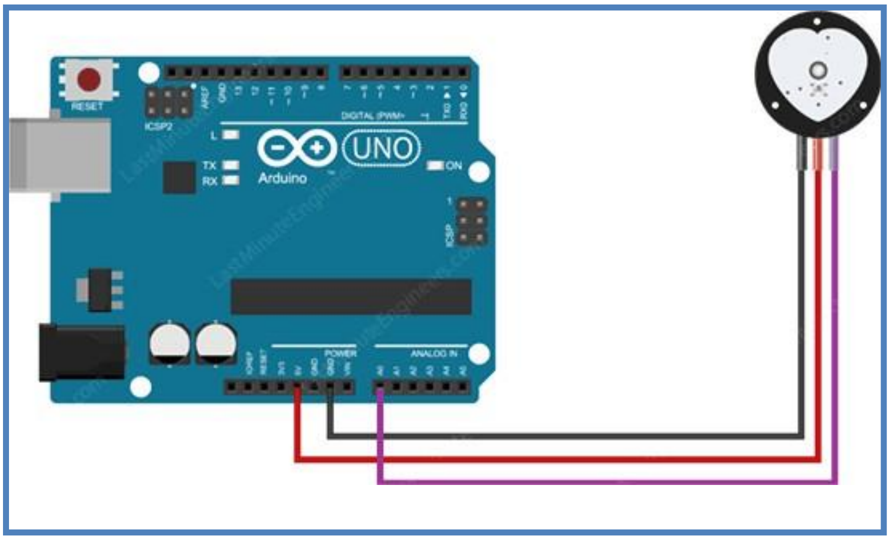

Figure (3.1) Wiring a pulse Sensor to an Arduino

After connecting pulse sensor to Arduino, we have written the following code to display the heartbeat signal on serial plotter of Arduino IDE.

```
void setup() {
Serial.begin(115200); // initialize the serial communication:
}
void loop() { Serial.println(applyfilter(analogRead(A0)));
}
delayMicroseconds(500); //Wait for a bit to keep serial data from saturating
}
#define REAL double #define NBQ 2
```

```
REAL biquada[]={0.9573350743594453,- 1.9541573210865666,0.9000141954472616,-1.8969295032005473}; REAL biquadb[]={1,2,1,2};
REAL gain=1632254.8438709665; REAL xyv[]={0,0,0,0,0,0,0,0,0};
REAL applyfilter(REAL v)
{
int i,b,xp=0,yp=3,bqp=0; REAL out=v/gain;
for (i=8; i>0; i--) {xyv[i]=xyv[i-1];} for (b=0; b<NBQ; b++)
{
int len=2; xyv[xp]=out;
for(i=0; i<len; i++) { out+=xyv[xp+len-i]*biquadb[bqp+i]-xyv[yp+len- i]*biquada[bqp+i]; }
bqp+=len; xyv[yp]=out; xp=yp; yp+=len+1;
}
return out;}
```

Fig. (3.2) illustrates the result of ECG Signal on serial plotter of Arduino IDE by using pulse sensor.

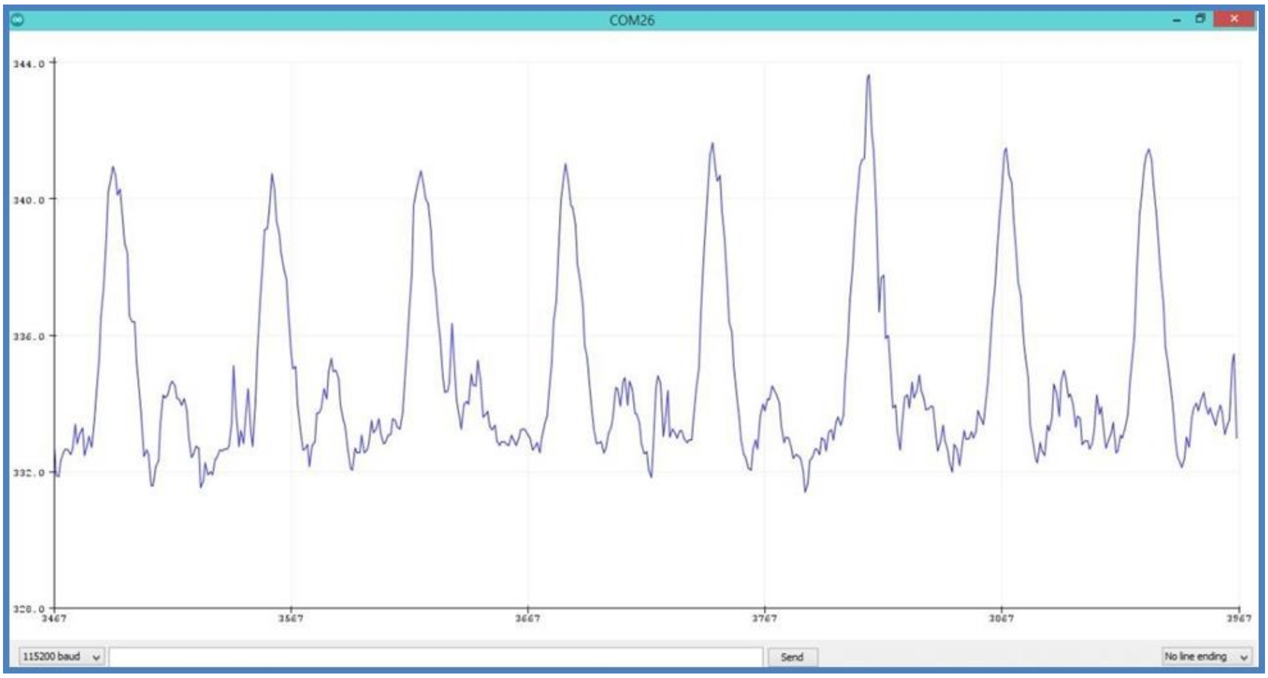

Figure (3.2) pulse sensor output signal on serial plotter of Arduino IDE.

2. Interfacing 1.3 INCH OLED IPS Display Module with Arduino

Fig. (3.3) below shows schematic diagram of connecting Arduino to ST7789 module. The ST7789 display module shown in circuit diagram has 7 pins: (from left to right): GND (ground), VCC, SCL (serial clock), SDA (serial data), RES (reset), DC (or D/C: data/command) and BLK (back light). Connecting the BLK pin is optional. The back light turns off when the BLK pin connected to the ground (GND).

The ST7789 TFT display controller works with 3.3V only (power supply and control lines). The display module is supplied with 3.3V (between VCC and GND) which comes from the Arduino board **[5]**.

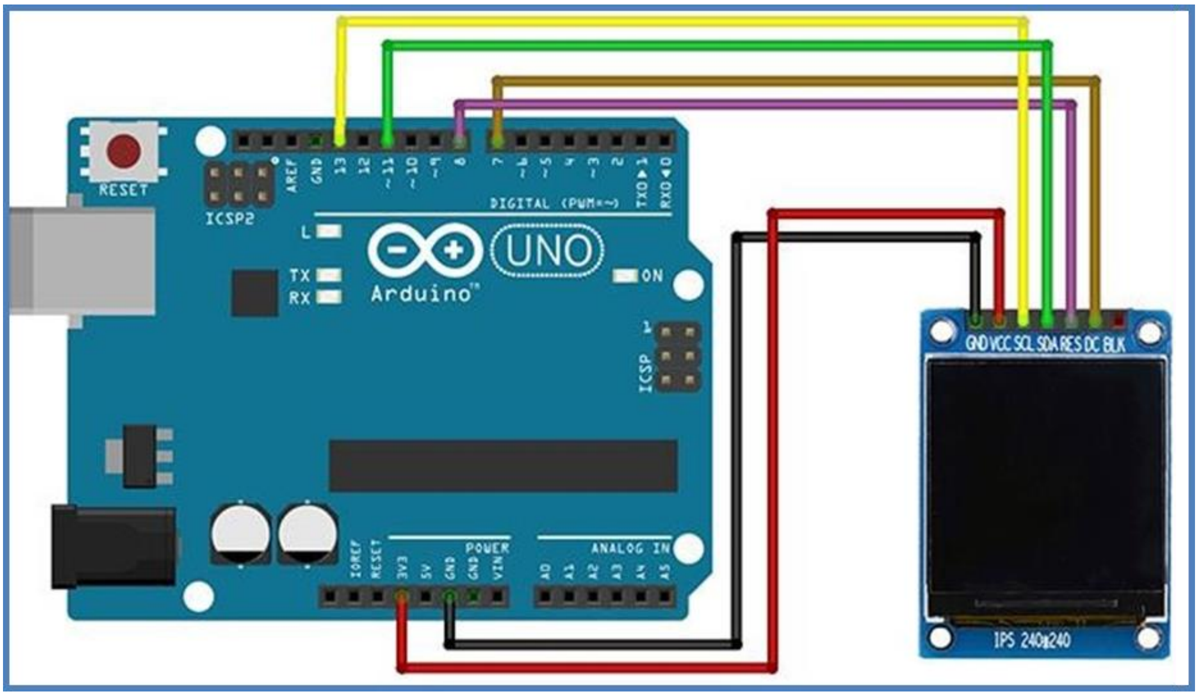

Figure (3.3) Wiring ST7789 module with Arduino

All Arduino UNO board output pins are 5V, connecting a 5V pin to the ST7789 TFT display may damage its controller. So, the ST7789 TFT display is connected to the Arduino board as illustrates in table (3-1) below.

Table (3-1) Electrical connection between ST7789 display module and Arduino

ST7789 display module pins	Arduino pins
GND	GND
VCC	3.3V
SCL	Pin 13
SDA	Pin11
RES	Pin8
DC	Pin7
BLK	Not connected

3. Project's Electrical Connection and Its Results

Fig. (3.4) below shows the electrical connection of the project, where we have combined the electrical conduction of the heartbeat sensor with the electrical conduction of the ST7789 TFT display module.

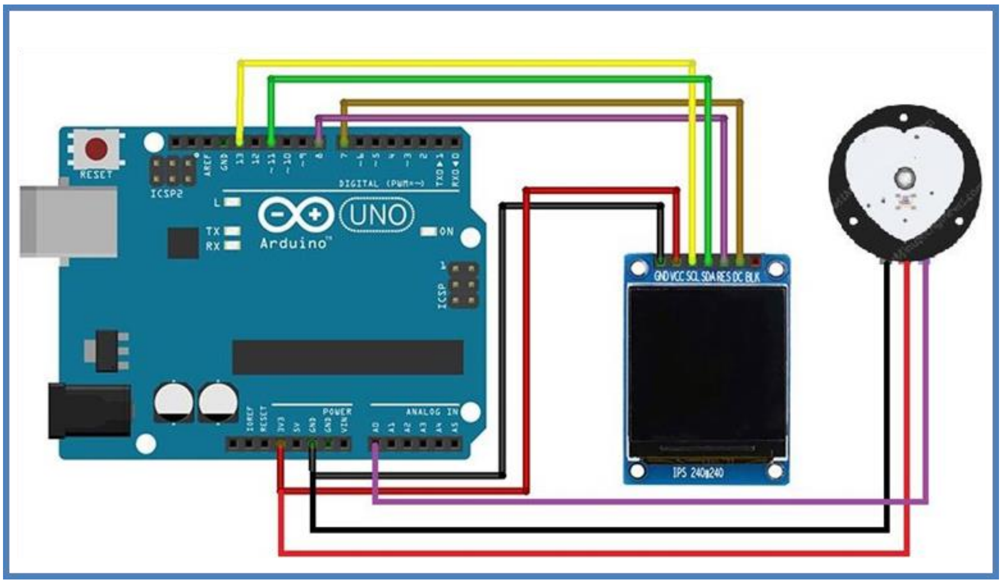

Figure (3.4) Electrical Connection of Project

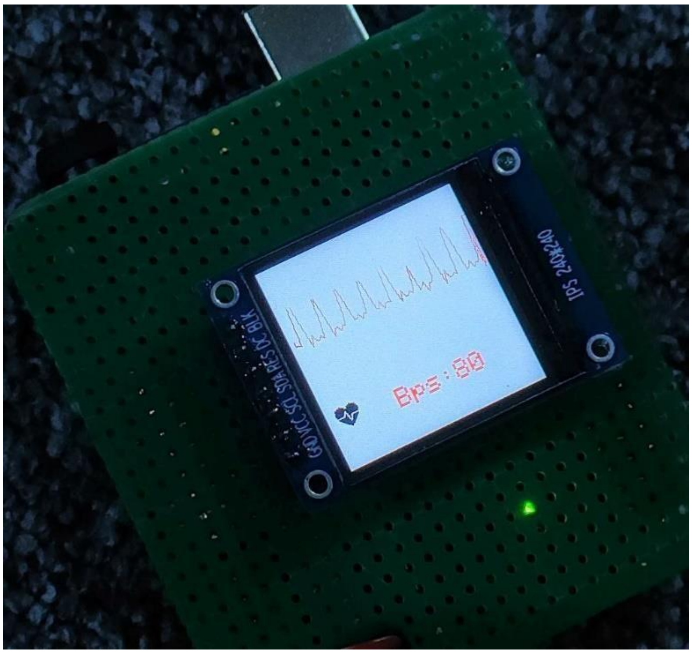

Figure (3.5) Heart beat rate and ECG Signal Result.

After combining project components on green vero-board, we wrote the programming code included in the appendix A and then checked the project results. Fig. (3.5) above shows the ECG signal as well as the heart rate taken from one of us.

Finally, we designed a box using AutoCAD software, then the box was manufactured using CNC, to appear in the wonderful form as shown in Fig. (3.6) below.

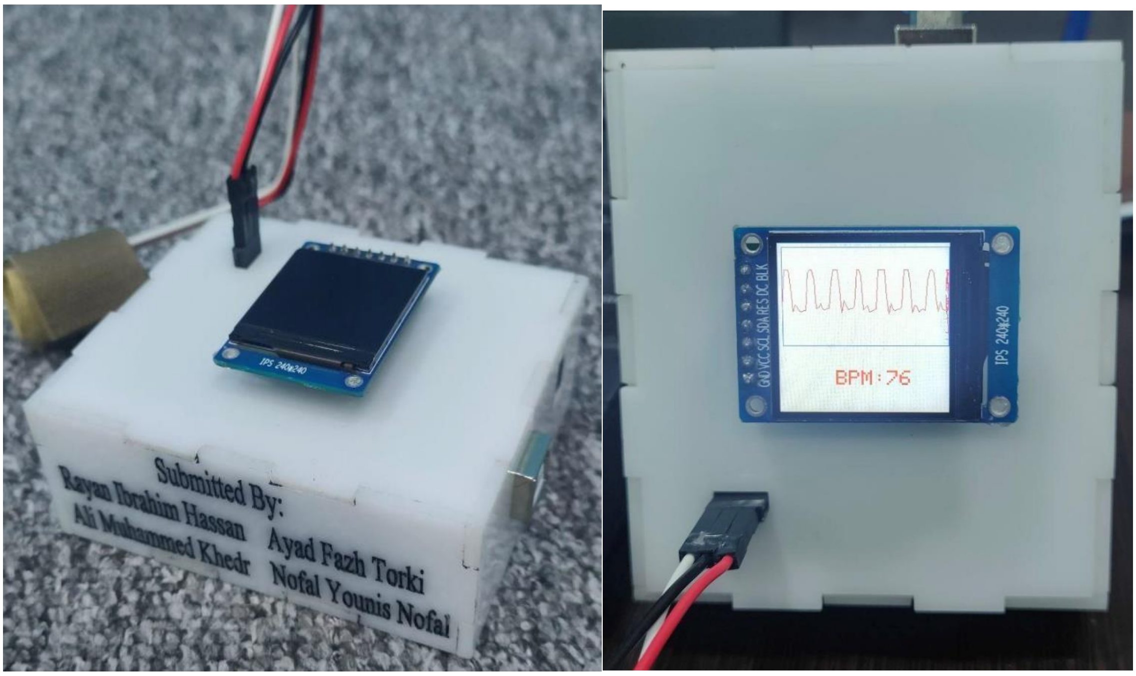

Figure (3.6) Project final form placed in a box

We compared the results obtained from our project with a device manufactured for the same purpose, and the results were satisfactory, and the convergence rate was very high.

CONCLUSIONS AND RECOMMENDATIONS

1. CONCLUSIONS

The project has several potential applications in the field of healthcare, including monitoring patients with heart conditions, tracking vital signs during medical procedures, and providing real-time feedback during rehabilitation programs. Additionally, the low-cost nature of the project makes it an affordable option for low-income communities and developing countries.

In conclusion, the Arduino-based ECG and heartbeat monitoring healthcare system project provides a low-cost and efficient solution for monitoring heartbeats and detecting anomalies in the ECG signal. The project has significant potential applications in healthcare and can improve the quality of life for patients with heart conditions.

2. RECOMMENDATIONS

Here are some recommendations for further improvements:

- Incorporate wireless connectivity: Adding wireless connectivity to the system would allow for remote monitoring and data collection. This would enable doctors and caregivers to monitor patients' ECG signals remotely and provide real-time feedback and alerts. This is done using the ESP8266 or ESP32 instead of the Arduino
- Improve signal-processing techniques: More advanced signal processing techniques could be implemented to enhance the accuracy of the ECG signal analysis. This could include algorithms for noise reduction and filtering, as well as machine learning algorithms for anomaly detection.
- Add a data logging feature: A data logging feature could be added to the system to record ECG data over time. This could be used for long-term monitoring of patients with heart conditions and for analyzing trends in their heart health.
- Integrate with other healthcare systems: The system could be integrated with other healthcare systems, such as electronic medical records or telemedicine platforms. This would enable

doctors and caregivers to access patient data in real-time and make more informed decisions about patient care.

- Conduct clinical trials: Clinical trials could be conducted to evaluate the effectiveness of the system in a real-world setting. This would help to validate the accuracy and reliability of the system and provide insights into its potential applications in healthcare.

By implementing these recommendations, the Arduino-based ECG and heartbeat monitoring healthcare system project can be improved to provide even more benefits to patients and healthcare providers.

REFERENCES

1. Bandana Mallick and Ajit Kumar Patro. “Heart Rate Monitoring System using Finger Tip Through Arduino and Processing Software” International Journal of Science, Engineering and Technology Research (IJSETR), Volume 5, Issue 1, January 2016.
2. Robert H. Shmerling, MD, “How’s your heart rate and why it matters?”, Harvard Medical School, March 25,2020 https://www.health.harvard.edu/heart- health/hows-your-heart-rate-and-why-it-matters
3. S. A. James, “What are the Different Types of Arduino Boards”, https://www.elprocus.com/different-types-of-arduino-boards , Elprocus,2019.
4. André Sam, “Monitor the Heart Rate using Pulse Sensor and Arduino”, https://lastminuteengineers.com/pulse-sensor-arduino-tutorial/ , Last Minute Engineers,2021.
5. Simple Projects Team work, “Interfacing Arduino with ST7789 TFT Display”, https://simple-circuit.com/arduino-st7789-ips-tft-display-example/ , SIMPLE PROJECTS, June 1, 2019