Heartbeat Monitoring System

Nishat Anjumane Salsabila[1] Anamika Ghosh[2] Sayma Sultana[3]

Department of Electrical and Electronic Engineering

Chittagong University of Engineering and Technology, Chittagong-4349, Bangladesh

Email: [1]u1902154@student.cuet.ac.bd ,[2]u1902162@student.cuet.ac.bd, [3]u1902164@student.cuet.ac.bd

Abstract-This project illustrates a Heartbeat monitoring system using a heart beat sensor and lcd display. Heart rate is one of the most important health parameter that is directly related to human cardiovascular system. Heart rate can be described as the number of times the heart beats per minute. Heart rate is utilized by medical professionals to diagnose and track medical conditions of a person. It is also use by individuals like athletes, aerobic activity who are interesting in monitoring their heart beat in order to acquire maximum efficiency. The system discussed read, stores and analyze the heart beat in real time. Cost of the entire system has been analyzed in the paper and it is verified that the system is low-priced compared to others existing device. So, the design can be used mainly as a comfortable and low-cost heart beat monitoring device for everyone.

Keyword: Arduino Uno, lcd display, real time, heart beat monitoring , PCB ,Proteus.

I. Introduction

When we're doing an aerobic activity like trail running, hiking or wild swimming, our heart rate increases, which trains our cardiovascular system to be more efficient, burns calories and lowers cholesterol, all of which is excellent news. But if we're looking to improve our performance on the trail and go further or faster, we might want to invest in a heart rate monitor to help our training. While measuring your beats per minute tells you exactly how fast your heart is beating. A heart rate monitor not only gives you more scientific data about your heart rate but can also help you adjust your training to meet your personal goals. It also stores historical data for a bigger picture of your performance. Heart rate monitors are small devices worn in different ways – usually around your chest, wrist or bicep, or hand – that measure electrical signals from your heart, pick up information from your pulse and transmit the data. Nowadays, the information is shared in real-time, so you'll be able to access it while you're moving and use it to adjust your pace and technique as needed. This project is based on two previous works. Finnish professor Seppo Säynäjäkangas invented the first battery-operated fingertip heart rate monitor as a training aid[2]. Arduino project hub uses a heart sensor and LCD to describe the heartbeat monitoring system.[1] The objective of this project is to implement a low-cost, reliable, make easy to manage intensity workout.[3]

II. Literature review

In this project, we mainly use Arduino Uno, a heartbeat sensor, and an LCD.

Here the Arduino Uno is one kind of microcontroller board based on ATmega328. This board includes digital I/O pins-14, a power jack, analogue i/ps-6, ceramic resonator-A16 MHz, a USB connection, an RST button, and an ICSP header. All these can support the microcontroller for further operation by connecting this board to the computer. AC or DC is a USB cable; otherwise, a battery adapter can supply this board with power [4].

Heartbeat monitoring is done using heartbeat sensors. Heartbeat sensors are designed to give a digital output heartbeat when a finger is placed on it. The LED blinks simultaneously for every heartbeat when the heartbeat detector starts working. Our heart does this for a healthy person around 72 to 84 times a minute.[5]

16x2 LCD can display 16 characters per line, with two such lines. In this LCD, each character is revealed in a 5x7 pixel matrix. The 16 x 2 intelligent

alphanumeric dot matrix display is capable of displaying 224 different characters and symbols. This LCD has two registers, namely, Command and Data.[6]

To accomplish our project goal, we used a 10k potentiometer to control the brightness of the LCD.

Finally, the Printed circuit boards (PCBs) are the foundational building block of most modern electronic devices. A Printed Circuit Board (PCB) is a Sheet of Insulating Material, such as Fiberglass, with Metallic Circuit or Track Printed or Etched on it for Electrical Conductivity.

III. Methodology

A. Background Theory

At a glance the required circuit parts :

1. Arduino Uno.

2. Heart beat sensor.

3. 220Ω resistor.

4. 10k potentiometer.

5. 16×2 lcd display.

6. PCB board.

7. Jumper wires.

Here, at first, we connect the components. Firstly, we secure the sensor's positive pin with 5v of the Arduino Uno and touch the ground with the Arduino UNO's base. Then click the sensor's analogue pin with UNO's A0 pin. After that, we connect the LCD. Firstly, we secure the lcd's positive and negative pin with 5v and ground. Then click enable and rs pin with digital pin 10,9 of the Arduino Uno. After that, connect the D4, D5, D6, and d7 pin with the digital pin 4,5,6,7 of the uno. Here we also use a 10k potentiometer whose control pin is connected with an LCD Vss pin and use a 220Ωresistor with the power control pin of the LCD. Here we use a PCB board. For this, we have to make a design using the proteus app. Then we print it on the PCB board and by using heat and Fecl3 solution we have printed the design on the board. We drill board where we need to add female port and solder the board. Finally, we give the connection properly and see the result. When we touch the sensor, the LED blinks simultaneously for every heartbeat. It gives a digital output heartbeat when a finger is placed on it. The output is shown on the LCD. Firstly the LCD shows, ”we created an object” when touching the sensor; the LCD a heart sign and delivers the heartbeat.

B. Code

```
#define USE_ARDUINO_INTERRUPTS true //--> Set-up low-level interrupts for most acurate BPM math.

#include <PulseSensorPlayground.h> //--> Includes the PulseSensorPlayground Library.

#include <LiquidCrystal.h> //--> Includes the LiquidCrystal Library.

LiquidCrystal lcd(10, 9, 4, 5, 6, 7); //--> Initialize LiquidCrystal with "lcd". lcd(RS,E,D4,D5,D6,D7).

//----------------------------------------Variable Declaration

const int PulseWire = 0; //--> PulseSensor PURPLE WIRE connected to ANALOG PIN 0

const int LED_3 = 3; //--> LED to detect when the heart is beating. The LED is connected to PIN 3 on the Arduino UNO.

int Threshold = 550; //--> Determine which Signal to "count as a beat" and which to ignore.

            //--> Use the "Gettting Started Project" to fine-tune Threshold Value beyond default setting.

            //--> Otherwise leave the default "550" value.

//----------------------------------------


//----------------------------------------Draw "Heart" on LCD.

/*

            heart4 heart5

             ===  ===

            ||  || ||  ||

        = 00011 11000 00011 11000 =                 11 11      11 11

       || 00111 11100 00111 11100 ||              111 111    111 111

       || 01111 11110 01111 11110 ||               1111 1111 1111 1111

       || 11111 11111 11111 11111 ||              11111 11111 11111 11111
```

```
    heart3 ||  11111 11111 11111 11111  || heart6          11111 11111 11111 11111 11111

          ||  11111 11111 11111 11111  ||                 11111 11111 11111 11111 11111

          ||  11111 11111 11111 11111  ||                 11111 11111 11111 11111 11111

          = 01111 11111 11111 11110 =                      1111 11111 11111 11111 1111

                                  ------->

          = 00011 11111 11111 11000 =                        11 11111 11111 11111 11

          ||  00001 11111 11111 10000  ||                  1 11111 11111 11111 1

          ||  00000 11111 11111 00000  ||                  11111 11111 11111 11111

    heart2 ||  00000 11111 11111 00000  || heart7          11111 11111 11111 11111

          ||  00000 01111 11110 00000  ||                   1111 1111

          ||  00000 00111 11100 00000  ||                    111 111

          ||  00000 00011 11000 00000  ||                     11 11

          = 00000 00001 10000 00000 =                         1 1

               ||   || ||   ||

               ===   ===

              heart1 heart8
*/

byte heart1[8] = {B11111, B11111, B11111, B11111, B01111, B00111, B00011, B00001};

byte heart2[8] = {B00011, B00001, B00000, B00000, B00000, B00000, B00000, B00000};

byte heart3[8] = {B00011, B00111, B01111, B11111, B11111, B11111, B11111, B01111};

byte heart4[8] = {B11000, B11100, B11110, B11111, B11111, B11111, B11111, B11111};

byte heart5[8] = {B00011, B00111, B01111, B11111, B11111, B11111, B11111, B11111};

byte heart6[8] = {B11000, B11100, B11110, B11111, B11111, B11111, B11111, B11110};

byte heart7[8] = {B11000, B10000, B00000, B00000, B00000, B00000, B00000, B00000};

byte heart8[8] = {B11111, B11111, B11111, B11111, B11110, B11100, B11000, B10000};

//----------------------------------------

int Instructions_view = 500; //--> Variable for waiting time to display instructions on LCD.

PulseSensorPlayground pulseSensor; //--> Creates an instance of the PulseSensorPlayground object called "pulseSensor"

//--------------------------------------------------------------------------------
--void setup

void setup() {

 Serial.begin(9600);//--> Set's up Serial Communication at certain speed.

 lcd.begin(16, 2); //--> Initializes the interface to the LCD screen, and specifies the dimensions (width and height) of the display

 //----------------------------------------Create a custom character (glyph) for use on the LCD

 lcd.createChar(1, heart1);

 lcd.createChar(2, heart2);

 lcd.createChar(3, heart3);

 lcd.createChar(4, heart4);

 lcd.createChar(5, heart5);

 lcd.createChar(6, heart6);

 lcd.createChar(7, heart7);

 lcd.createChar(8, heart8);

 //----------------------------------------

 lcd.setCursor(0,0);

 lcd.print(" HeartBeat Rate ");

 lcd.setCursor(0,1);

 lcd.print("   Monitoring   ");

 //----------------------------------------Configure the PulseSensor object, by assigning our variables to it.

 pulseSensor.analogInput(PulseWire);

 pulseSensor.blinkOnPulse(LED_3); //--> auto-magically blink Arduino's LED with heartbeat.

 pulseSensor.setThreshold(Threshold);

 //----------------------------------------

 //----------------------------------------Double-check the "pulseSensor" object was created and "began" seeing a signal.

 if (pulseSensor.begin()) {

  Serial.println("We created a pulseSensor Object !"); //--> This prints one time at Arduino power-up,  or on Arduino reset.

 }

 //----------------------------------------

 delay(20);
```

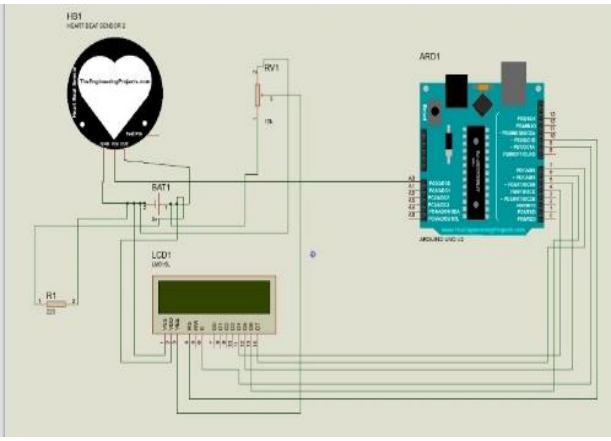
```
 lcd.clear();

}

//--------------------------------------------------------------------------------
--

//--------------------------------------------------------------------------------
--void loop


void loop() {

 int myBPM = pulseSensor.getBeatsPerMinute(); //--> Calls function on our pulseSensor object that returns BPM as an "int". "myBPM" hold this BPM value now.


//----------------------------------------Condition if the Sensor does not detect the heart rate / the sensor is not touched.

 if (Instructions_view < 500) {

   Instructions_view++;

 }



 if (Instructions_view > 499) {

   lcd.setCursor(0,0);

   lcd.print("Put your finger ");

   lcd.setCursor(0,1);

   lcd.print("on the sensor   ");

   delay(1000);

   lcd.clear();

   delay(500);

 }

 //----------------------------------------



//----------------------------------------Constantly test to see if "a beat happened".

 if (pulseSensor.sawStartOfBeat()) { //--> If test is "true", then the following conditions will be executed.

   Serial.println("♥  A HeartBeat Happened ! "); //--> Print a message "a heartbeat happened".

   Serial.print("BPM: "); //--> Print phrase "BPM: "

   Serial.println(myBPM); //--> Print the value inside of myBPM.

 //----------------------------------------Displays a "Heart" shape on the LCD.

   lcd.setCursor(1,1);

   lcd.write(byte(1));

   lcd.setCursor(0,1);

   lcd.write(byte(2));

   lcd.setCursor(0,0);

   lcd.write(byte(3));

   lcd.setCursor(1,0);

   lcd.write(byte(4));

   lcd.setCursor(2,0);

   lcd.write(byte(5));

   lcd.setCursor(3,0);

   lcd.write(byte(6));

   lcd.setCursor(3,1);

   lcd.write(byte(7));

   lcd.setCursor(2,1);

   lcd.write(byte(8));


//----------------------------------------


 //----------------------------------------Displays the BPM value on the LCD.

   lcd.setCursor(5,0);

   lcd.print("Heart Rate");

   lcd.setCursor(5,1);

   lcd.print(": ");

   lcd.print(myBPM);

   lcd.print(" ");

   lcd.print("BPM    ");

   //----------------------------------------

    Instructions_view = 0;

 }

 //----------------------------------------

 delay(20); //--> considered best practice in a simple sketch.

}
```

IV. Software Simulation

A. About simulation software

The Proteus Design Suite is a Windows application for schematic capture, simulation, and PCB (Printed Circuit Board) layout design. It can be purchased in many configurations, depending on the size of designs being produced and the requirements for microcontroller simulation.[7] Proteus is quite lenient in circuit designing and it works on ideal condition. if you don't add pull up resistors in Proteus simulation, then it won't give garbage value.

B. Circuit Connection



V. Hardware Implementation

A. About parts

In this project, we mainly use Arduino Uno, a heartbeat sensor, and an LCD.

Here the Arduino Uno is one kind of microcontroller board based on ATmega328. This board includes digital I/O pins-14, a power jack, analogue i/ps-6, ceramic resonator-A16 MHz, a USB connection, an RST button, and an ICSP header. All these can support the microcontroller for further operation by connecting this board to the computer. AC or DC is a USB cable; otherwise, a battery adapter can supply this board with power [4].

Heartbeat monitoring is done using heartbeat sensors. Heartbeat sensors are designed to give a digital output heartbeat when a finger is placed on it. The LED blinks simultaneously for every heartbeat when the heartbeat detector starts working. Our heart does this for a healthy person around 72 to 84 times a minute.[5]

16x2 LCD can display 16 characters per line, with two such lines. In this LCD, each character is revealed in a 5x7 pixel matrix. The 16 x 2 intelligent alphanumeric dot matrix display is capable of displaying 224 different characters and symbols. This LCD has two registers, namely, Command and Data.[6]

To accomplish our project goal, we used a 10k potentiometer to control the brightness of the LCD.

Finally, the Printed circuit boards (PCBs) are the foundational building block of most modern electronic devices. A Printed Circuit Board (PCB) is a Sheet of Insulating Material, such as Fiberglass, with Metallic Circuit or Track Printed or Etched on it for Electrical Conductivity.

B. Circuit Diagram

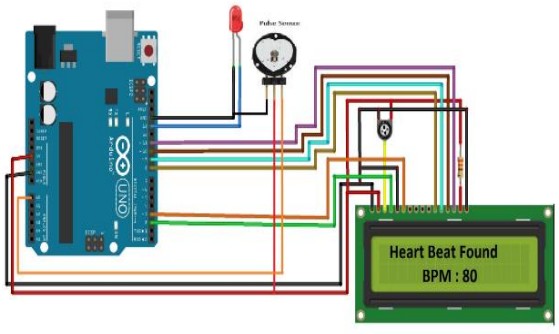


Figure: Circuit configuration

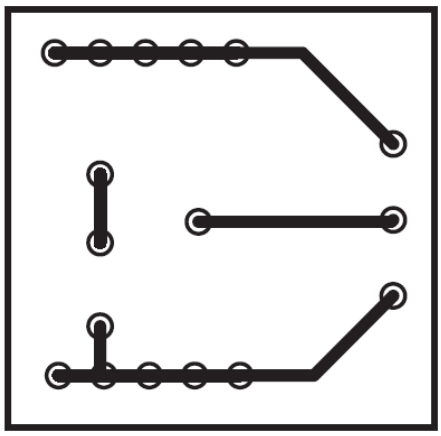


Figure: PCB implementation

VII. Result and application

A heart rate monitor (HRM) is a personal monitoring device that allows one to measure heart rate at a time or record the heart rate for later study.

The main applications are given below:

1. This system is necessary for safe exercise. Like aerobic activities like trail running, hiking and wild swimming.

2. Basically, a heart rate monitor acts as your coach. It will show you when to pull back and pump it up. This helps you determine what you want to achieve and ensure the best outcome for the time you put into your workout plan, improving fitness safety.

VIII. Drawbacks

While doing this project we have experienced some difficulties. The main difficulty is that are not able to find the perfect heart beat sensor for the circuit. Being unable to find it, we have not found the pulse rate accurately. Again, we have faced problem for lcd display. Sometimes, the lcd is not light up. This is happened for

The loose connection of the connecting wire and power pin of the lcd display. Here, the lcd display also show some block for the loose connection of the wire with the 10k potentiometer. For this, we hard to be very careful as if there's any loose connection, the circuit will not work properly. Lastly, it would have been a great project if we use some micro-controller chip for the circuit, then the circuit will be easy to implementation.

VI. Cost Analysis

This system is designed keeping in mind that it should be cheaper than other existing systems available in the market and, at the same time, sustainable perspective on the product's accuracy, stability and durability. So, project finance analysis is done. Table 1 shows the cost of the components, which justifies that this project is a really cheaper one.

Table 1: Details of Purchased Instruments

Items	Quantity	Per unit cost (BDT)	Total cost(BDT)
Arduino Uno	1	600	600
Heartbeat sensor	1	120	120
220Ω resistor	1	3	3
10k potentiometer	1	15	15
16×2 lcd display	1	150	150
PCB board	1	25	25
Jumper wires	many	30	30
			Total=943

IX. Comparative study

There are many Heart beat monitoring system is available in the market. But the prices are around 1500-5000 BDT. But our system's price is under 1000 BDT (10.44 USD) which makes this system affordable.

X. Conclusion

The experiment done with Arduino uno was done successfully. Heartrate monitoring device can measure pulse rate within very less time . We have developed the system in such a way that it can be implemented for monitoring heart beat in real time. This system is helpful for Monitoring of Heart beat for athletes and person who are doing exerciser regularly. This system can be used in home, or during travelling, or in hospitals also. For testing this system we have measures heart rate and pressure of different person. It shows the accurate value. Sometimes, we faced some problems. However, Our project is very essential so we proposed it to reduce extra effort and to add comfort in life.

Acknowledgement

The success and outcome of this project required a lot of guidance and assistance and we are extremely fortunate to have got this all along the completion of our project work. The authors also thank and owe profound gratitude to all the professors of *Department of Electrical & Electronic Engineering* for their proper guidance and motivation for this project.

References

1.https://cc.bingj.com/cache.aspx?q=heart+beat+monitoring+system+by+arduino&d=4709165495621344&mkt=en-US&setlang=en-US&w=2YNr2lQAis9de24yZun6cM-SH9IXzrk6.(project)

2. https://en.wikipedia.org/wiki/Heart_rate_monitor#:~:text=The%20first%20wireless%20EKG%20heart,heart%20monitors%20started%20in%201983..(sensor invent)

3. Julia Clarke," heart-rate-monitor", March 07, 2022

4. https://www.elprocus.com/atmega328-arduino-uno-board-working-and-its-applications/.(uno)

5. https://www.sciencedirect.com[sensor]6. https://www.thingbits.in/products/standard-lcd-16x2-display[lcd]

6.Administrator,Wikipedia. November 4, 2017

7.www.theengineeringprojects.com/2020/01/introduction-to-proteus.html