SRI RAMACHANDRA
INSTITUTE OF HIGHER EDUCATION AND RESEARCH
(Category - I Deemed to be University) Porur, Chennai
SRI RAMACHANDRA ENGINEERING AND TECHNOLOGY

HEART RATE MONITORING SYSTEM

PROJECT REPORT

Quarter I (Year 1)

Submitted by

NIVEDHIDHA I E0220003

In partial fulfilment for the award of the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

(Cyber Security & Internet of Things)

Sri Ramachandra Engineering and Technology

Sri Ramachandra Institute of Higher Education and Research, Porur, Chennai -600116

JANUARY, 2021

SRI RAMACHANDRA
INSTITUTE OF HIGHER EDUCATION AND RESEARCH
(Category - I Deemed to be University) Porur, Chennai
SRI RAMACHANDRA ENGINEERING AND TECHNOLOGY

BONAFIDE CERTIFICATE

Certified that this project report **"Heart Rate Monitoring System"** is the bonafide work of **NIVEDHIDHA I Reg No. E0220003** who carried out the internship work under my supervision.

Signature of Faculty Mentor

Nandhini JJ

Assistant Professor

Sri Ramachandra Engineering and Technology

Porur

Chennai-600116

Signature of Vice-Principal

Prof. M. Prema

Vice-Principal

Sri Ramachandra Engineering and Technology

Porur

Chennai-600116

Evaluation Date:

TABLE OF CONTENTS

ACKNOWLEDGEMENT

I express my sincere gratitude to our Chancellor, Vice-Chancellor and our sincere gratitude to our Provost **Dr. Raju** and our Vice-Principal **Prof. Prema** for their support and for providing the required facilities for carrying out this study.

I wish to thank my faculty mentor, **Prof. Nandhini JJ,** Department of Computer Science and Engineering, Sri Ramachandra Engineering and Technology for extending help and encouragement throughout the project. Without his/her continuous guidance and persistent help, this project would not have been a success for me.

I am grateful to the Department of Computer Science and Engineering, Sri Ramachandra Engineering and Technology, our beloved parents and friends for extending the support, who helped us to overcome obstacles in the study.

I. OBJECTIVE:

To make a circuit for MONITORING HEART BEAT using IoT

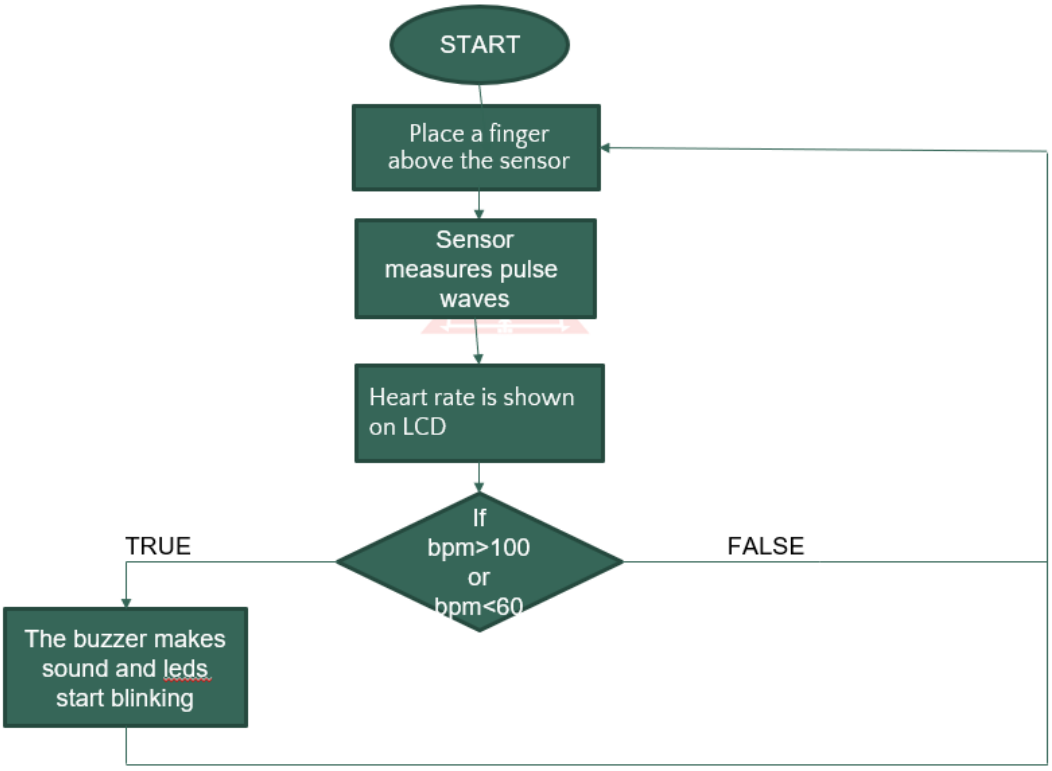
II. WORKFLOW:

The person is supposed to place his/her finger on the sensor. The sensor detects and displays “WAIT FOR 10 SECONDS”.

The sensor has a LED and an ambient light sensor. The sensor works on the principle TRANSMISSION type pulse wave measurement. When blood flows through the nerves the light is blocked and ambient light sensor cant detect light, this way KY-039 sensor works to sense heartbeat.

In the mean while time the sensor passes data to the Arduino UNO R3. Then the microprocessor will process the data and display the heart rate on the LCD panel.

If the detected heartbeat is less than 60 i.e., **Bradycardia** or heartbeat is more than 100 i.e., **Tachycardia** then “EMERGENCY” is displayed on the LCD display and LED’s and buzzer are activated in such a way that the people nearby can be alerted as an emergency.



III. ARDUINO IDE CODE:

The Arduino Integrated Development Environment (IDE) is the main text editing program used for Arduino programming. It is used to write and upload programs to Arduino compatible boards.

LiquidCrystal.h Library:

This library allows an Arduino board to control LiquidCrystal displays (LCDs) based on the Hitachi HD44780 (or a compatible) chipset, which is found on most text-based LCDs. The library works in either 4- or 8-bit mode (i.e. using 4 or 8 data lines in addition to the rs, enable, and, optionally, the rw control lines).

Main Functions:

print() - Prints text to the LCD.

write() - Write a character to the LCD.

setCursor() - Position the LCD cursor; that is, set the location at which subsequent text written to the LCD will be displayed.

clear() - Clears the LCD screen and positions the cursor in the upper-left corner.

createChar() - Create a custom character (glyph) for use on the LCD.To display a custom character on the screen, write() its number.

begin() - Initializes the interface to the LCD screen, and specifies the dimensions (width and height) of the display. begin() needs to be called before any other LCD library commands.

Project Code:

```
#include <LiquidCrystal.h>


#define Sensor A0

#define d7 2

#define d6 3

#define d5 4

#define d4 5

#define buzzer 6

#define LED 8

#define enable 11

#define rs 12


double alpha=0.75;

int period=20;

double refresh=0.0;
```

```
LiquidCrystal lcd(rs, enable, d4, d5, d6, d7);


byte mini_heart[8] = { B00000, B00000, B01010, B11111, B11111, B01110, B00100, B00000 };

byte large_heart1[8] = { 0b11111, 0b11111, 0b11111, 0b01111, 0b00111, 0b00011, 0b00001, 0b00000 };

byte large_heart2[8] = { 0b00000, 0b00000, 0b00000, 0b00100, 0b01110, 0b11111, 0b11111, 0b11111 };

byte large_heart3[8] = { 0b11111, 0b11111, 0b11111, 0b11110, 0b11100, 0b11000, 0b10000, 0b00000 };

byte large_heart4[8] = { 0b00000, 0b00000, 0b00000, 0b00100, 0b01110, 0b11111, 0b11111, 0b11111 };


void setup()

{

  pinMode(Sensor, INPUT);


  lcd.begin(16, 2);

  lcd.clear();

  lcd.createChar(0, mini_heart);

  lcd.createChar(1, large_heart1);

  lcd.createChar(2, large_heart2);

  lcd.createChar(3, large_heart3);
```

```
  lcd.createChar(4, large_heart4);


  pinMode(LED, OUTPUT);
  pinMode(buzzer,OUTPUT);
}
void loop()
{
  static double oldValue=0;
  static double oldrefresh=0;
  int beat=analogRead(A0);  //define sensor pin to AO
  double value=alpha*oldValue+(0-alpha)*beat;
  refresh=value-oldValue;


  lcd.clear();
  lcd.setCursor(3, 0);
  lcd.print("Heart Rate");
  lcd.setCursor(4, 1);
  lcd.print("Monitor");
  lcd.setCursor(11, 1);
  lcd.write((byte)0);
  delay(5000);
```

```
lcd.clear();
lcd.setCursor(2, 0);
lcd.write((byte)2);
lcd.setCursor(2, 1);
lcd.write((byte)1);
lcd.setCursor(3, 0);
lcd.write((byte)4);
lcd.setCursor(3, 1);
lcd.write((byte)3);
lcd.setCursor(4, 0);
lcd.print("HEART BEAT");
lcd.setCursor(4, 1);
lcd.print("Wait 10 sec");
delay(10000);

lcd.clear();
lcd.setCursor(2, 0);
lcd.write((byte)2);
lcd.setCursor(2, 1);
lcd.write((byte)1);
lcd.setCursor(3, 0);
lcd.write((byte)4);
```

```
lcd.setCursor(3, 1);
lcd.write((byte)3);
lcd.setCursor(4, 0);
lcd.print("HEART BEAT");
lcd.setCursor(4, 1);
lcd.print("BPM: ");
lcd.setCursor(9, 1);
lcd.print(beat/10);

if (beat/10>100)
{
  digitalWrite(buzzer,HIGH);
  digitalWrite(LED,HIGH);
  delay(1000);
  digitalWrite(buzzer,LOW);
  digitalWrite(LED,LOW);
  lcd.setCursor(4, 0);
  lcd.print("EMERGENCY ");
}

else if(beat/10<60)
{
```

```
    digitalWrite(buzzer,HIGH);
    digitalWrite(LED,HIGH);
    delay(1000);
    digitalWrite(buzzer,LOW);
    digitalWrite(LED,LOW);
    lcd.setCursor(4, 0);
    lcd.print("EMERGENCY ");
   }

  oldValue=value;
  oldrefresh=refresh;
  delay(period*10);
}
```

IV. IMPLEMENTATION:

Usage of Arduino UNO:

When the patient places his/her finger to find heart rate in the sensor, the components work as instructed by the Arduino. It processes the input from the sensor and converts into bpm, instructs the LCD to display the bpm and in case of an emergency situation it makes the LCD to display the word “emergency”, LED to glow and the buzzer to make sound.

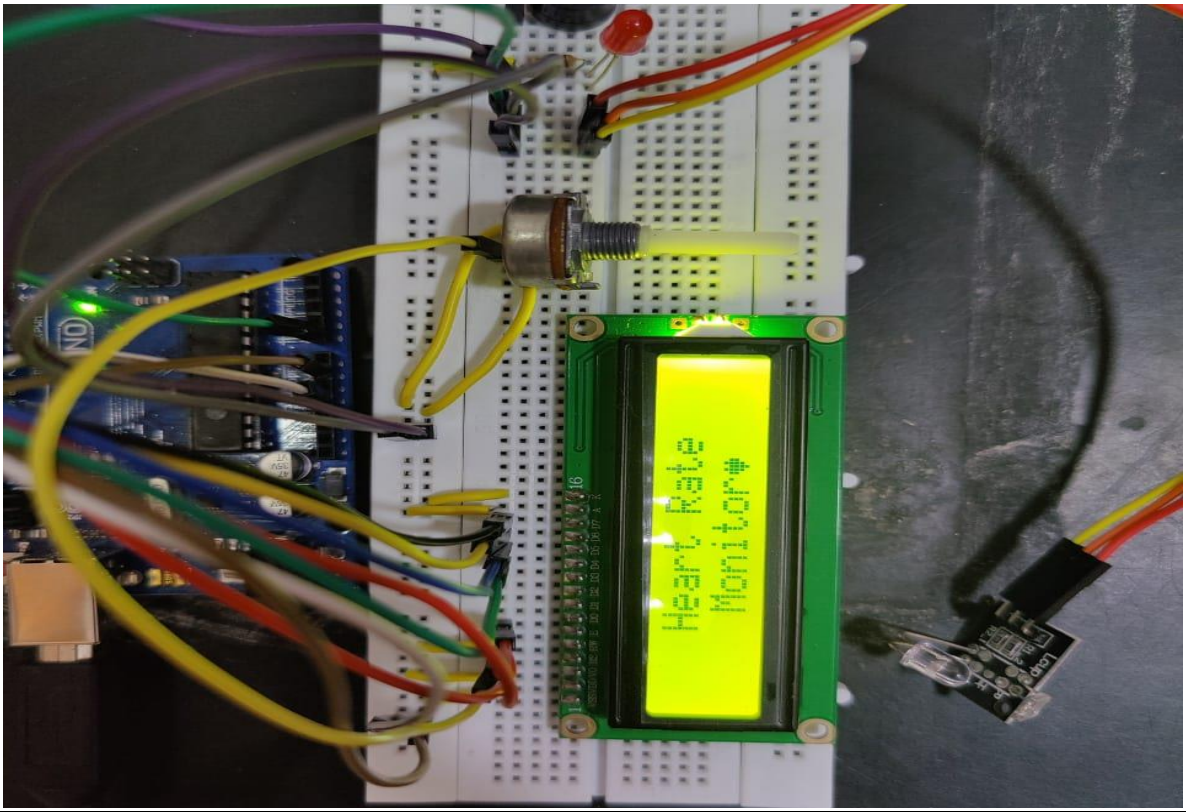
Usage of KY-039 Sensor:

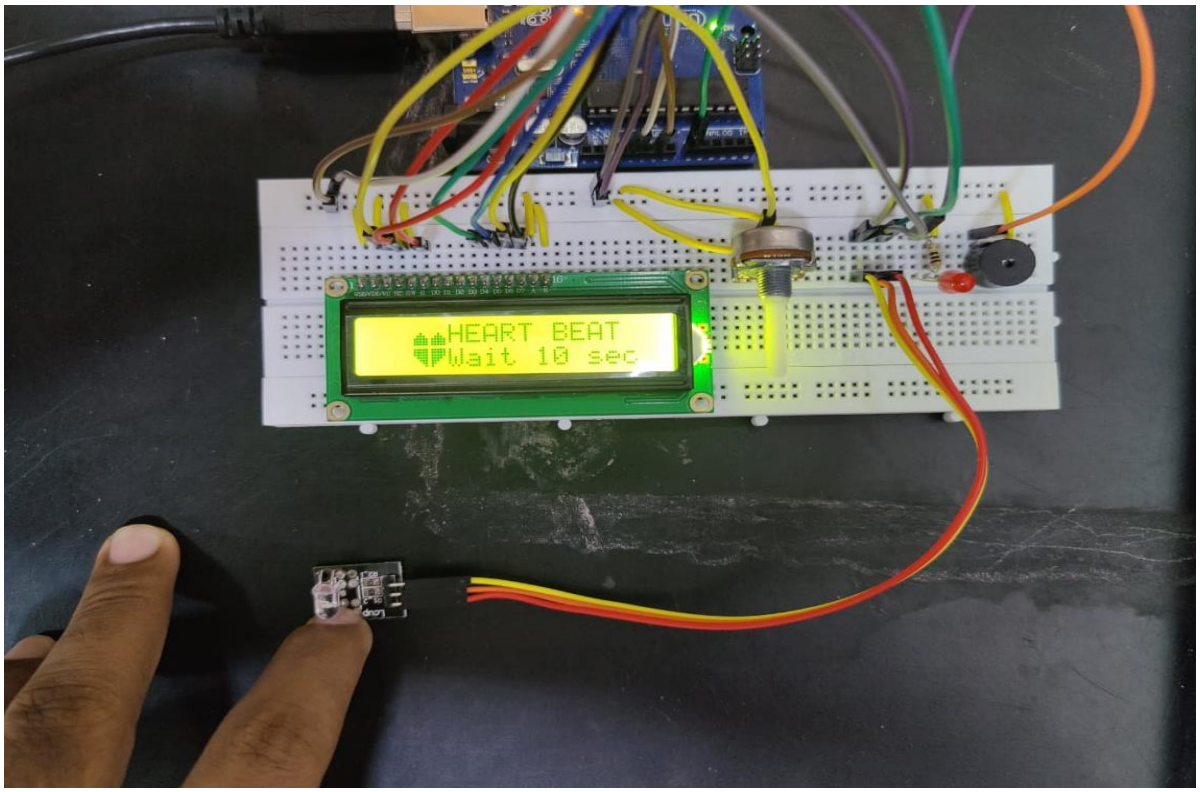
This KY-039 Finger Detection Heartbeat Measuring Sensor Module uses an infrared (IR) LED and a phototransistor to detect the pulse of the finger, a red LED flashes with each pulse. It sends the signals to the Arduino in the form of voltage. When the patient keeps his/her fingers on the sensor,the sensor works as the heart beats as the heart pumps the to all parts of the body,it takes that pumping as calculation to calculate BPM of the heart

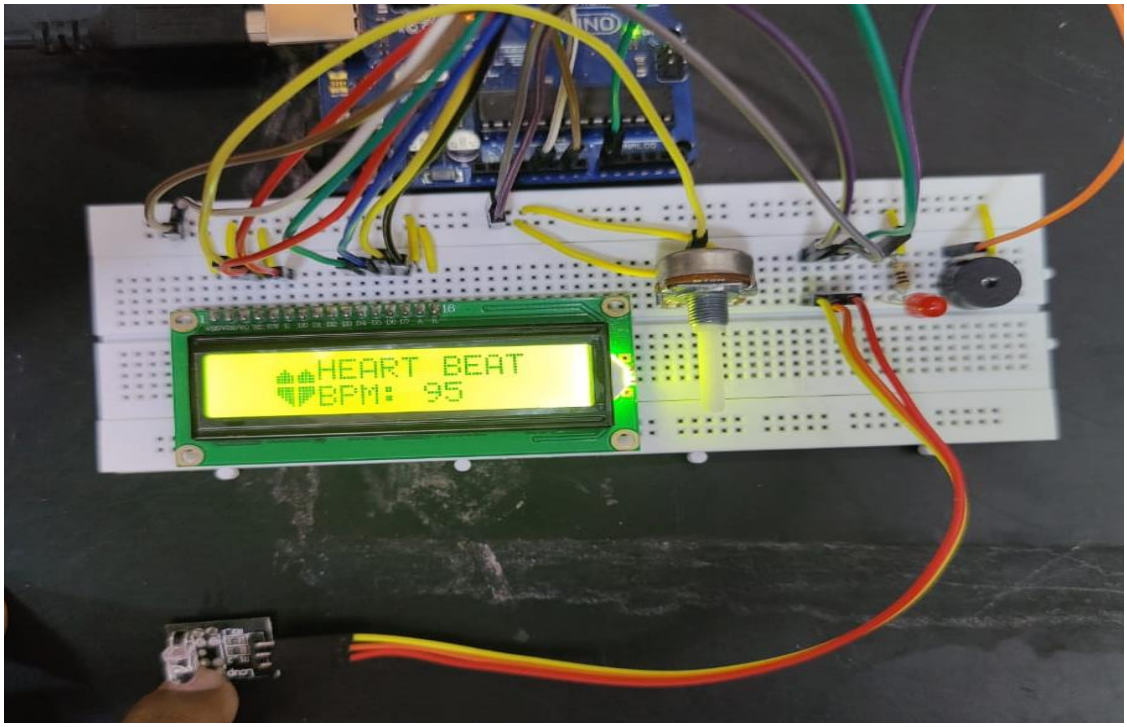
Usage of LCD:

LCD displays are used for displaying the heart rate of the patient. It displays the heart rate of the patient and in case of an emergency, i.e., when the heart rate goes below 60bpm or above 100 bpm, the LCD displays the word "EMERGENCY" as directed by the Aurduino.

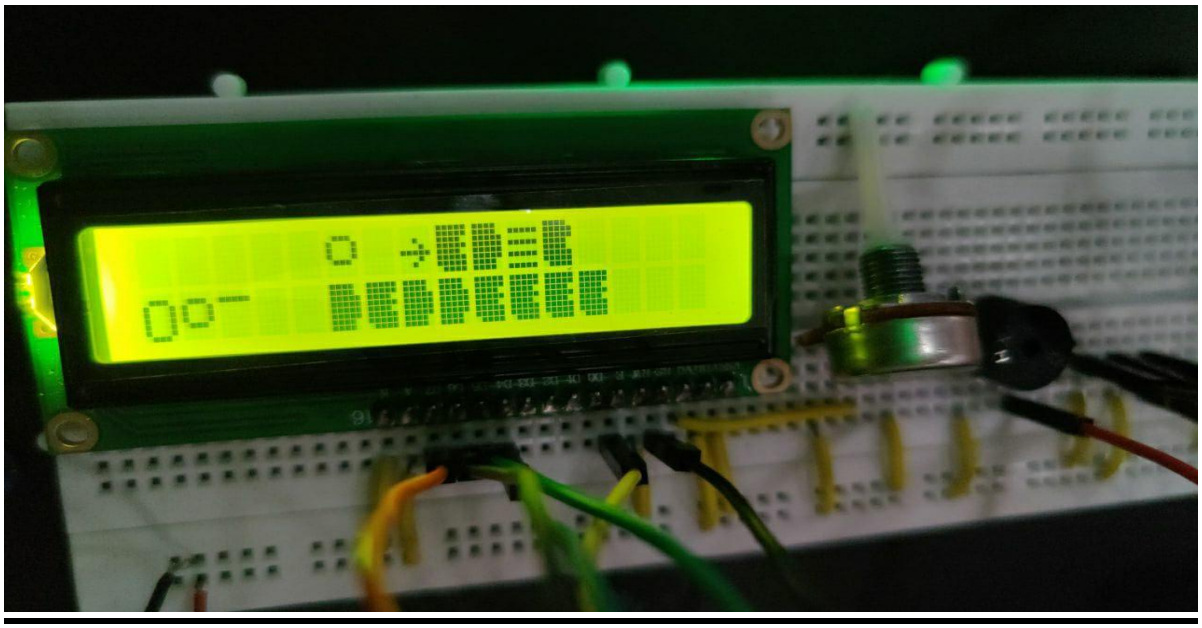
V. SAMPLE OUTPUT:







VI. TROUBLESHOOTING PHOTO:



VII. CONCLUSION:

This project will help in monitoring the heart rate of patients with heart conditions in a successful manner and will help in alarming people around for them to get immediate help. This technology will be efficient when compared to traditional ECG's.

VIII. FUTURE SCOPE:

In the future, many implementations can take place, like ECG monitoring, proper oxygen supply controlling and above all the controlling system will automatically on or off.

We are gonna add the oximeter sensor so that People with respiratory or cardiovascular conditions, very young infants, and individuals with some infections may benefit from pulse oximetry.

We are gonna add ESP866, which is a small module that allows microcontrollers to connect to a Wi-Fi network and make simple TCP/IP connections using Hayes-style commands, so that it performs certain automation tasks by using an application or software such as sending an automated message to specific phone numbers so that they get notified.

IX. REFERENCES:

Hardware Connections:

- ➤ hackerearth.com
- ➤ thegeekpub.com
- ➤ circuitstoday.com

Programming:

- ➤ electronicshub.org
- ➤ generationrobots.com
- ➤ create.arduino.cc/projecthub
- ➤ engineersgarage.com