Khulna University of Engineering & Technology

Course No: CSE 3104.

Couse Title: Peripheral & Interfacing Laboratory.

Project Name: Pulse rate monitor.

Submitted to:

Dr. Muhammad Sheikh Sadi,

Professor, Dept. of CSE, KUET.

Md. Milon Islam,

Lecturer, Dept. of CSE, KUET.

Submitted by:

Khadiza Sarwar Moury.

Roll No: 1607010.

Samariya Nawrin.

Roll No: 1607028.

Nishat Anjum Lea.

Roll No: 1607009.

Year: 3rd.

Term: 1st.

*Objectives:*

* To know about pulse rate and monitor it.
* To determine healthiness of a person by indicating pulse rate measurement.
* To determine the change of blood flow.
* To give digital output of heat beat when a finger is placed on it.
* To use arduino in building a monitor machine in order to measure pulse rate.
* To study the function of the heart.

*Introduction:*

A person’s heartbeat is the sound of the valves in his heart contracting or expanding as they force blood from one region to another. The number of times the heart beats per minute (BPM), is the heart beat rate and the beat of the heart that can be felt in any artery that lies close to the skin is the pulse.

Heart Beat can be measured based on optical power variation as light is scattered or absorbed during its path through the blood as the heart beat changes.

*Principle of Heartbeat Sensor:*

The heartbeat sensor is based on the principle of photo phlethysmography. It measures the change in volume of blood through any organ of the body which causes a change in the light intensity through that organ (a vascular region). In case of applications where heart pulse rate is to be monitored, the timing of the pulses is more important. The flow of blood volume is decided by the rate of heart pulses and since light is absorbed by blood, the signal pulses are equivalent to the heart beat pulses.

Usually, the source of light in a heartbeat sensor would be an IR LED and the detector would be any Photo Detector like a Photo Diode, an LDR (Light Dependent Resistor) or a Photo Transistor. It works on the principle that when light falls on the resistor, its resistance changes. As the light intensity increases, the resistance decreases. Thus, the voltage drops across the resistor decreases.

Here a comparator is used which compares the output voltage from the LDR to that of the threshold voltage. The threshold voltage is the voltage drop across the LDR when the light with fixed intensity, from the light source falls directly on it. The inverting terminal of the comparator LM358 is connected to the potential divider arrangement which is set to the threshold voltage and the no- inverting terminal is connected to the LDR. When a human tissue is illuminated using the light source, the intensity of the light reduces. As this reduced light intensity falls on the LDR, the resistance increases and as a result the voltage drop increases. input, a logic high signal is developed at the output of the comparator and in case voltage drop being lesser a logic low output is developed. Thus, the output is a series of pulses. These pulses can be fed to the Microcontroller which accordingly processes the information to get the heart beat rate. This signal is actually a DC signal relating to the tissues and the blood volume and the AC component synchronous with the heart beat and caused by pulsatile changes in arterial blood volume is superimposed on the DC signal. Thus, the major requirement is to isolate that AC component as it is of prime importance.

*Apparatus Required:*

1. Arduino Uno.
2. Heartbeat sensor module.
3. 16\*2 LCD display.
4. 10kΩ potentiometer.
5. Push button.
6. Bread board.
7. 1kΩ resistor.
8. Red & blue LED.
9. Connecting wires.

*Block Diagram:*

*Flowchart:*

*Software Design:*

**Pseudo Code:**

1. Press the push button.
2. For 10 seconds

Loop :

1. Take the first pulse time when input reading exceeds upper thresholds.
2. Take the second pulse time when input reading below lower thresholds and take pulse interval.
3. Calculate BPM using BPM = (1/pulse\_interval) \*60.0\*1000.0;

End Loop.

3.Calculate the mean of BPM and show in the LCD display.

Code:

#include<LiquidCrystal.h>

const int rs = 12, en = 11, d4 = 5, d5 = 4, d6 = 3, d7 = 2;

LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

int UpperThreshold = 518, LowerThreshold = 490;

int reading = 0;

float BPM = 0.0;

bool IgnoreReading = false, FirstPulseDetected = false;

unsigned long FirstPulseTime = 0, SecondPulseTime = 0, PulseInterval = 0;

const unsigned long delayTime = 10, delayTime2 = 500;

unsigned long previousMillis = 0, previousMillis2 = 0;

const int data=A0, start=7, warning= 13, blue= 8;

unsigned long count=0, temp=0;

void setup() {

lcd.begin(16, 2);

Serial.begin(9600);

pinMode(data,INPUT);

pinMode(start,INPUT\_PULLUP);

pinMode(warning, OUTPUT);

pinMode(blue, OUTPUT);}

int myTimer1(long delayTime, long currentMillis){

if(currentMillis - previousMillis >= delayTime){previousMillis = currentMillis;return 1;}

else{return 0;}

}

int myTimer2(long delayTime2, long currentMillis){

if(currentMillis - previousMillis2 >= delayTime2){previousMillis2 = currentMillis;

return 1;}

else{return 0;}

}

void loop() {

digitalWrite(warning, LOW);

digitalWrite(blue, LOW);

lcd.setCursor(0, 0);

lcd.print("Place The Finger");

Serial.println("Place The Finger");

lcd.setCursor(0, 1);

lcd.print("And Press Start");

Serial.println("And Press Start");

count =0;

unsigned long c=0,sum=0;

while(digitalRead(start)>0);

lcd.clear();

temp=millis();

while(millis()<(temp+10000)){

unsigned long currentMillis = millis();

if(myTimer1(delayTime, currentMillis) == 1){

reading = analogRead(data);

if(reading > UpperThreshold && IgnoreReading == false){

if(FirstPulseDetected == false){

FirstPulseTime = millis();

FirstPulseDetected = true;

}

else{

SecondPulseTime = millis();

PulseInterval = SecondPulseTime - FirstPulseTime;

FirstPulseTime = SecondPulseTime;

}

IgnoreReading = true;

}

if(reading < LowerThreshold && IgnoreReading == true){

IgnoreReading = false;

}

if(PulseInterval>0){

lcd.setCursor(0, 0);

lcd.print("counting..");

BPM = (1.0/PulseInterval) \* 60.0 \* 1000;

sum=sum+BPM;

c=c+1;

}

}

if(myTimer2(delayTime2, currentMillis) == 1 && PulseInterval>0){

Serial.print(reading);

Serial.print("\t");

Serial.print(PulseInterval);

Serial.print("\t");

Serial.print(BPM);

Serial.println(" BPM");

Serial.flush();

}

lcd.clear();

sum=sum/c;

count=sum;

if(count<=100.0 && count>=50.0){

digitalWrite(blue, HIGH);

}

else{

digitalWrite(warning, HIGH);

}

lcd.setCursor(0, 0);

lcd.print(count);

lcd.print(" BPM");

Serial.print(sum);

Serial.println(" BPM\*\*\*\*\*");

delay(8000);

}

*Hardware Design:*

*System Testing:*

White box testing: Unit testing: In this regard, before integrating any part each module unit (like LCD display, heartbeat sensor, 10K potentiometer, register, switch) has been tested. Even the code has been tested using simulation (proteus).

[Integration Testing](http://softwaretestingfundamentals.com/integration-testing/): After attaching all the parts again the output has been checked and rechecked until all valid outcome was found.

Black box testing: The module has been tested using Equivalence Partitioning technique.  It is a software test design technique that involves dividing input values into valid and invalid partitions and selecting representative values from each partition as test data. The valid partitions were within 50 BPM to 100 BPM in the following test. All results outside this range is considered invalid.

*Experimental Result:*

The module has been tested on three people (around 21 times), based on the output the conclusion has been taken over its accuracy. The outputs were documented. They are sequentially written below:

Persson1:

1. 75 BPM

2. 75 BPM

3. 79 BPM

4. 75 BPM

5. 78 BPM

6. 77 BPM

7. 78 BPM

Person2:

8. 73 BPM

9. 73 BPM

10. 74 BPM

11. 73 BPM

12. 74 BPM

13. 73 BPM

14. 72 BPM

Person3:

15. 82 BPM

16. 82 BPM

17. 83 BPM

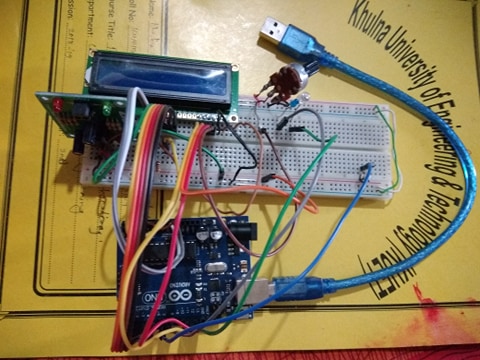
18. 82 BPM

19. 82 BPM

20. 80 BPM

21. 82 BPM

*Protype Design:*

**

Fig

*Discussion:*

In this project a pulse rate monitor which is both cheap and easy to use has been introduced. The monitor itself consists of a heartbeat sensor which can process analog data, received based on optical power variation since light is scattered or absorbed during its path through the blood as the heart beat changes and then convert the data to digital and show the output on per each pulse or heartbeat. Though the project is a complete one but it has some limitations such as the output or pulse rate it measures is largely dependent on the correct use of the module and applied pressure on it. Moreover, it requires considerable amount of technical knowledge to operate it. So, the module is generally used for athletes or for first aid treatments. But above all the module is for monitoring pulse rate and can be used for medical purpose, on top this can be quite handy and cheap. So, it may be said that the project is an important and successful one.

*Conclusion:*

A pulse rate monitor is a personal monitoring device that allows one to measure or display heart rate in real time or record the heart rate for later study. It is largely used to gather heart rate data while performing various types of physical exercise. The main purpose of this automatic health system is to monitor the body heart rate and pulse rate of a patient and display the same to the doctor.

*Reference:*

1. <https://www.techshopbd.com/product-categories/modules-98775/2960/heart-beat-sensor-module-techshop-bangladesh>.
2. <https://www.electronicshub.org/heartbeat-sensor-using-arduino-heart-rate-monitor/>
3. <https://www.medicalnewstoday.com/articles/235710.php>