Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

ECG BASED HEART RATE MONITORING SYSTEM USING ARDUINO

V.Venkata narendra - 18BCE0515

B Sai Koushik - 18BCE0961

Sai Nikhil Reddy M -18BCE0527

Harsha Vardan Kamal -18BCE0983

Mugilan T - 18BCE0946

Under the guidance of

TEACHER : SWETHA B THOMAS

COURSE CODE : CSE2006

Theory Slot: G1

Lab SLOT : L41+L42

TABLE OF CONTENTS

ABSTRACT:

World Health Organization (WHO) researchshows that most people die due to heart disease. The basic parameters, which are monitored by a human regularly to keep track of their health, are Heart rate, body temperature and blood pressure. We can easily track/diagnose any abnormal heart activity, indicating to possible heart problems in an early stage by monitoring the ECG signal. Hence, constant tracking of heart rate rhythm can help in identifying heart problems due to any abnormal waves. This paper presents a method to create an ECG based Heart Rate Monitoring System using Arduino with a temperature sensor which will display the heart wave on a monitor (screen) and also the body temperature. This has application in portable ECG remote health monitors and fitness and heart rate monitors.

.

INTRODUCTION:

Heart rate is one of the most frequently measured parameters of the human body and plays an important role in determining an individual's health. Heart rate measurement is becoming a part of the typical consumer lifestyle, and many electronic devices. World Health Organization (WHO) research also shows that the most people was dying due to heart disease. The cardiovascular abnormality is one of the biggest causes of deaths among people of all races around the globe especially in the case of old age people. In few years, it is expected that the world population of age 65 and older would exceed the population of the world with age of 15 years. Therefore, this disease cannot be taken lightly. Hence, most health care equipment and monitoring system are designed to keep track the disease. In some countries, people still do not have access to quality health facilities owing to different barriers. This project can help people in rural and remote areas who are far away from hospital or any medical healthcare to take care of their health. Health monitoring of a patient distantly has become an easy and plausible task with the advent of this technology. A heart rate monitor is a personal monitoring device that allows a person to measure the heart rate in real time or record for later study. This also has application in portable ECG remote health monitors and fitness and heart rate monitors. Sowith the development of this system the elder patients will have the facility to get them diagnosed from the comfort of their home without traveling to hospitals.

LITERATURE REVIEW

[1]This paper explains how a single chip microcontroller can be used to analyze heart beat signals in real-time. The main feature of this paper is the use of zero crossing algorithms to compute heart rate. It process real time information to determine heart diseases. The hardware is implemented using the PIC16F876 microcontroller. The software is of two parts one for getting signals implemented in the hardware and other one is Graphic Unit Interface.

[2]This paper presents the design and development of a microcontroller based heart rate monitor using fingertip sensor. The device uses the optical technology to detect the flow of blood through the finger and offers the advantage of portability over tapebased recording systems. The important feature of this research is the use of Discrete Fourier Transforms to analyze the ECG signal in order to measure the heart rate. Then performance of HRM device was compared with ECG signal represented on an oscilloscope and manual pulse measurement of heartbeat, giving precise results.

[3] This paper describes a technique of measuring the heart rate through a fingertip and Arduino. It is based on the principal of Photo Phelthysmo Graphy (PPG), which is non-invasive method of measuring the variation in blood volume in tissue using a light source and detector. While the heart is beating, it is pumping blood throughout the body, and that makesthe blood volume inside the finger artery to change too. This fluctuation of blood can be detected through an optical sensing mechanism placed around the fingertip. Thesignal can be amplified and is sent to Arduino with the help of serial port communication. With the help of processing software heart rate monitoring and counting is performed

[4] In this paper, the propose method for ECG monitoring is based on Internet-of-Things (IoT) techniques. ECG data are gathered using a wearable monitoring node and are transmitted directly to the IoT cloud using Wi-Fi. Both the HTTP and MQTT protocols are employed in the IoT cloud in order to provide visual and timely ECG data to users.

[5]This paper describes the design of a simple 3-lead Electrocardiogram (ECG) monitoring and heart rate measurement system with LCD output.The system takes the physical pulse input using Ag/Cl sticking electrodes stuck to the arms and right leg of the patient under observation. The model encompasses of instrumentation amplifier and filter Circuits etc., which are used for signal conditioning of the pulse, input from the patient's body and displayed on CRO as the ECG waveform. Thus conditioned signal is also processed by the microcontroller AT89S52 to count the heart for duration of one minute and displays the information on LCD display.

[6]This paper presents a microcontroller ARM7 based health monitoring system intended to monitor and to early detect situations when heart rate and blood oxygen level are out of their safe ranges. This system employs a programmable ARM7 for confab the bio-signal to determine the condition of heart. If any abnormalities are discovered from patient's heart parameters, the system sends alarm to the doctor. The system ensures wireless transmission of ECG signal to the Medical Server (doctor's PC) through Bluetooth and Android platform.

[7]This paper introduces a wearable Tele-ECG and heart rate (HR) monitoring system which has a novel architecture including a stretchable singlet redesigned with textile electrodes (TEs), textile threads, snap fasteners, Velcro, sponges, and an ECG circuit. In addition, a Bluetooth low energy (BLE), a smartphone, a server, and a web page have been added to the system for remote monitoring.

[8]This paper presents a prototype for the monitoring of Heartbeat rate. A Heart Beat (HB) sensor is being developed for acquainting the input signals using Light Dependent Resistance (LDR) and Light Emitting Diode (LED). It senses the heartbeat of a person and converts it in the form of electricalsignals and pulses. The signals are amplified using a signal conditioning circuit and processed by a controller. The frequency of the signal depends on the heartbeat rate; this lays down the basic principle of the HB measuring system.

[9]This paper proposed an alert system that able to monitor the heart beat rate condition of patient. The heart beat rate is detected using photoplethysmograph (PPG) technique. This

signal is processed using PIC16F87 microcontroller to determine the heart beat rate per minute. Then, it sends SMS alert to the mobile phone of medical experts or patient's family members, or their relatives via SMS.

[10]In this paper, a basic framework is developed for a simple real-time heart-rate (HR) monitoring system using noncontact capacitive electrocardiograph (ECG). The system uses of an electrode structure, which can be integrated within a steering wheel and simple, low-cost signal processing blocks. These tests show that the proposed system gives good quality ECG signals and estimates HR with good accuracy

TABLE

S No	Year	Title and Author	Methodologies Used	Observations
1	2008	Microcontroller Based Heart Rate Monitor **Authors:** Mohamed Fezari, Mounir Bousbia-Salah, and MouldiBedda	Zero crossing algorithm to compute heart rate.	The hardware is implemented using the PIC16F876 microcontroller. The software is of two parts one, for getting signals implemented in the hardware and other one is Graphic Unit Interface.
2	2012	Microcontroller based Heart Rate Monitor using Fingertip Sensors **Authors:** Babiker, Sharief&Elrayah Abdel-Khair, Liena	Optical technology to detect the flow of blood through the finger and offers the advantage of portability over tape-based recording systems	The performance of HRM device was compared with ECG signal represented on an oscilloscope and manual pulse measurement of heartbeat, giving precise results.
3	2017	Fingertip based heart beat monitoring system using embedded systems	Principal of Photo PhelthysmoGraphy (PPG)	The signal can be amplified and is sent to Arduino with the help of serial port communication. With the help of processing

		Authors: T. S. Arulananth and B. Shilpa		software heart rate monitoring and counting is performed
4	2016	An IoT-cloud Based Wearable ECG Monitoring System for Smart Healthcare **Authors:** Yang, Qihao Zhou, Lei Lei, Kan Zheng, Wei Xiang	ECG monitoring is based on Internet-of-Things (IoT) techniques	HTTP and MQTT protocols are employed in the IoT cloud in order to provide visual and timely ECG data to users.
5	2013	Design and Implementation of ECG Monitoring and Heart Rate Measurement System **Authors:** Naazneen M. G., Sumaya Fathima, Syeda Husna Mohammadi, Sarah Iram L. Indikar, Abdul Saleem, Mohamed Jebran	The system takes the physical pulse input using Ag/Cl sticking electrodes stuck to the arms and right leg of the patient under observation.	Conditioned signal is processed by the microcontroller AT89S52 to count the heart for duration of one minute and displays the information on LCD display.
6	2016	An Android based wireless ECG monitoring system for cardiac arrhythmia **Authors:** V. Wahane and P. V. Ingole	Employs a programmable ARM7 for confab the bio-signal to determine the condition of heart	The system ensures wireless transmission of ECG signal to the Medical Server (doctor's PC) through Bluetooth and Android platform.
7	2018	A Portable Wearable Tele-ECG Monitoring System **Authors:**	A novel architecture including a stretchable singlet redesigned with textile electrodes (TEs), textile threads, snap	A wearable Tele-ECG and heart rate (HR) monitoring system with good accuracy

		H. Ozkan, O. Ozhan, Y. Karadana, M. Gulcu, S. Macit and F. Husain	fasteners, Velcro, sponges, and an ECG circuit	
8	2014	Heartbeat ratemonitoring system by pulse technique. **Authors:** J. Arora, Gagandeep, A. Singh, N. P. Singh, S. S. S. Rawat and G. Singh	Heart Beat (HB) sensor for acquainting the input signals using Light Dependent Resistance (LDR) and Light Emitting Diode (LED)	The frequency of the signal depends on the heartbeat rate; this lays down the basic principle of the HB measuring system.
9	2009	Heartbeat monitoring alert via SMS **Authors:** W. Mat Jubadi and S. F. AisyahMohdSahak	Photoplethysmograph (PPG) technique	Able to monitor the heart beat rate condition of patient and sends SMS alert to the mobile phone of medical experts or patient's family members, or their relatives.
10	2016	A real-time heart-rate monitor using non- contact electrocardiogram for automotive drivers **Authors:** R. K. Singh, A. Sarkar, R. K. Thakur and C. S. Anoop	Non-contact capacitive electrocardiograph (ECG)	Gives good quality ECG signals and estimates HR with good accuracy.

METHODOLOGY

Hardware and Software Specifications

Components:

1. Arduino UNO
2. Heart Beat sensor module
3. 16x2 LCD Display

4. Push button

5. Bread board

6. Power

7. Connecting wires

8. Resistor of 330Ω

Working of Heartbeat Monitor Project

Working of this project is quite easy but a little calculation for calculating heart rate is required. There are several methods for calculating heart rate, but here we have read only five pulses. Then we have calculated total heart beat in a minute by applying the below formula:

Five_pusle_time = time2-time1;

Single_pulse_time = Five_pusle_time /5;

Rate = 60000/ Single_pulse_time;

where time1 is first pulse counter value

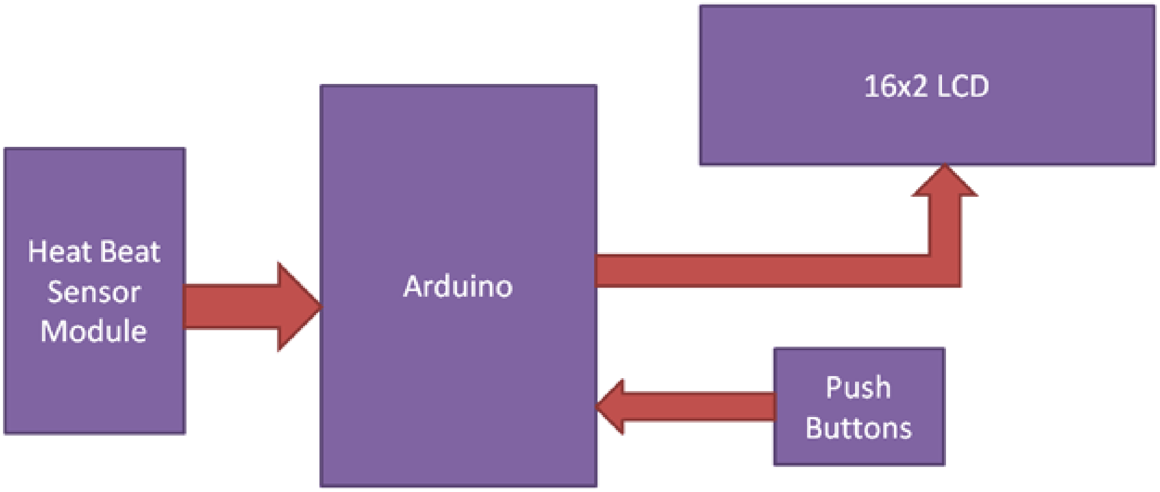
time 2 is list pulse counter valuerate is final heart rate.

When first pulse comes, we start counter by using timer counter function in arduino that is millis();. And take first pulse counter value form millis();. Then we wait for five pulses. After getting five pulses we again take counter value in time 2 and then we subtract time1 from time 2 to take original time taken by five pulses. And then divide this time by 5 times for getting single pulse time. Now we have time for single pulse and we can easily find the pulse in one minute, dividing 600000 ms by single pulse time.

Rate= 600000/single pulse time.

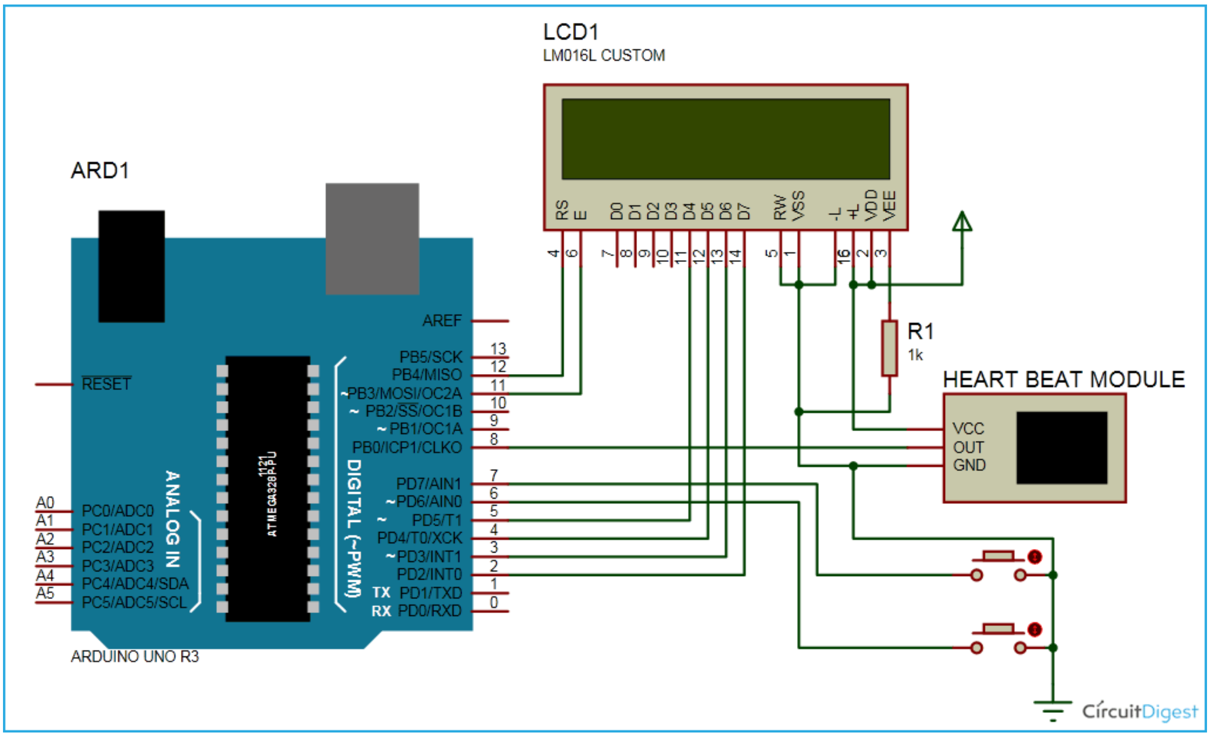
In this project we have used Heart beat sensor module to detect Heart Beat. This sensor module contains an IR pair which actually detect heart beat from blood. Heart pumps the blood in body which is called heart beat, when it happens the blood concentration in body changes. And we use this change to make a voltage or pulse electrically.

PROPOSED ARCHITECTURE DIAGRAM



Circuit Diagram and Explanation

Circuit of heartbeat monitor is shown below, which contains arduino uno, heart beat sensor module, reset button and LCD. Arduino controls whole the process of system like reading pulses form Heart beat sensor module, calculating heart rate and sending this data to LCD. We can set the sensitivity of this sensor module by inbuilt potentiometer placed on this module.



Heart beat sensor module's output pin is directly connected to pin 8 of arduino. Vcc and GND are connected to Vcc and GND. A 16x2 LCD is connected with arduino in 4-bit mode. Control pin RS, RW and En are directly connected to arduino pin 12, GND and 11. And data pin D4-D7 is connected to pins 5, 4, 3 and 2 of arduino. And one push button is added for resetting reading and another is used to start the system for reading pulses. When we need to count heart rate, we press start button then arduino start counting pulses and also start counter for five seconds. This start push button is connected to pin 7 and reset push button is connected to pin 6 of arduino with respect to ground

<u>Code</u>

```
int pulsePin = A0;                 // Pulse Sensor purple wire connected to analog pin A0
int blinkPin = 13;                // pin to blink led at each beat

// Volatile Variables, used in the interrupt service routine!
volatile int BPM;                   // int that holds raw Analog in 0. updated every 2mS
volatile int Signal;                // holds the incoming raw data
volatile int IBI = 600;             // int that holds the time interval between beats! Must be seeded!
volatile boolean Pulse = false;     // "True" when User's live heartbeat is detected. "False" when not a "live beat".
volatile boolean QS = false;        // becomes true when Arduoino finds a beat.

static boolean serialVisual = true;   // Set to 'false' by Default.  Re-set to 'true' to see Arduino Serial Monitor ASCII Visual Pulse
```

```
volatile int rate[10];                    // array to hold last ten IBI values
volatile unsigned long sampleCounter = 0;          // used to determine pulse timing
volatile unsigned long lastBeatTime = 0;           // used to find IBI
volatile int P = 512;                      // used to find peak in pulse wave, seeded
volatile int T = 512;                     // used to find trough in pulse wave, seeded
volatile int thresh = 525;                // used to find instant moment of heart beat, seeded
volatile int amp = 100;                   // used to hold amplitude of pulse waveform, seeded
volatile boolean firstBeat = true;        // used to seed rate array so we startup with reasonable BPM
volatile boolean secondBeat = false;      // used to seed rate array so we startup with reasonable BPM


void setup()
{
  pinMode(blinkPin,OUTPUT);         // pin that will blink to your heartbeat!
  Serial.begin(115200);             // we agree to talk fast!
  interruptSetup();                 // sets up to read Pulse Sensor signal every 2mS
                                   // IF YOU ARE POWERING The Pulse Sensor AT VOLTAGE LESS THAN THE BOARD VOLTAGE,
                                   // UN-COMMENT THE NEXT LINE AND APPLY THAT VOLTAGE TO THE A-REF PIN
                                   //   analogReference(EXTERNAL);
```

```
}



//  Where the Magic Happens
void loop()
{
  serialOutput();

  if (QS == true) // A Heartbeat Was Found
    {
      // BPM and IBI have been Determined
      // Quantified Self "QS" true when arduino finds a heartbeat
      serialOutputWhenBeatHappens(); // A Beat Happened, Output that to serial.
      QS = false; // reset the Quantified Self flag for next time
    }

  delay(20); //  take a break
}



void interruptSetup()
{
  // Initializes Timer2 to throw an interrupt every 2mS.
```

```
  TCCR2A = 0x02;     // DISABLE PWM ON DIGITAL PINS 3 AND 11, AND
GO INTO CTC MODE
  TCCR2B = 0x06;     // DON'T FORCE COMPARE, 256 PRESCALER
  OCR2A = 0X7C;      // SET THE TOP OF THE COUNT TO 124 FOR 500Hz
SAMPLE RATE
  TIMSK2 = 0x02;       // ENABLE INTERRUPT ON MATCH BETWEEN
TIMER2 AND OCR2A
  sei();             // MAKE SURE GLOBAL INTERRUPTS ARE ENABLED
}

void serialOutput()
{   // Decide How To Output Serial.
 if (serialVisual == true)
  {
     arduinoSerialMonitorVisual('-', Signal);   // goes to function that makes Serial
Monitor Visualizer
  }
 else
  {
      sendDataToSerial('S', Signal);     // goes to sendDataToSerial function
   }
}

void serialOutputWhenBeatHappens()
```

```
{
 if (serialVisual == true) //  Code to Make the Serial Monitor Visualizer Work
   {
     Serial.print(" Heart-Beat Found ");  //ASCII Art Madness
     Serial.print("BPM: ");
     Serial.println(BPM);
   }
 else
   {
     sendDataToSerial('B',BPM);   // send heart rate with a 'B' prefix
     sendDataToSerial('Q',IBI);   // send time between beats with a 'Q' prefix
   }
}

void arduinoSerialMonitorVisual(char symbol, int data )
{
  const int sensorMin = 0;      // sensor minimum, discovered through experiment
  const int sensorMax = 1024;        // sensor maximum, discovered through
experiment
  int sensorReading = data; // map the sensor range to a range of 12 options:
  int range = map(sensorReading, sensorMin, sensorMax, 0, 11);
  // do something different depending on the
  // range value:
```

```
}



void sendDataToSerial(char symbol, int data )
{
   Serial.print(symbol);
   Serial.println(data);
}

ISR(TIMER2_COMPA_vect) //triggered when Timer2 counts to 124
{
  cli();                                      // disable interrupts while we do this
  Signal = analogRead(pulsePin);              // read the Pulse Sensor
  sampleCounter += 2;                         // keep track of the time in mS with this
variable
  int N = sampleCounter - lastBeatTime;       // monitor the time since the last beat
to avoid noise
                                              //  find the peak and trough of the pulse wave
  if(Signal < thresh && N > (IBI/5)*3) // avoid dichrotic noise by waiting 3/5 of
last IBI
    {
      if (Signal < T) // T is the trough
      {
        T = Signal; // keep track of lowest point in pulse wave
```

```
    }
   }

  if(Signal > thresh && Signal > P)
   {         // thresh condition helps avoid noise
     P = Signal;                            // P is the peak
   }                                        // keep track of highest point in pulse wave

  //  NOW IT'S TIME TO LOOK FOR THE HEART BEAT
  // signal surges up in value every time there is a pulse
  if (N > 250)
  {                                   // avoid high frequency noise
    if ( (Signal > thresh) && (Pulse == false) && (N > (IBI/5)*3) )
      {
        Pulse = true;                              // set the Pulse flag when we think there is a pulse
        digitalWrite(blinkPin,HIGH);               // turn on pin 13 LED
        IBI = sampleCounter - lastBeatTime;        // measure time between beats in mS
        lastBeatTime = sampleCounter;              // keep track of time for next pulse

        if(secondBeat)
        {                        // if this is the second beat, if secondBeat == TRUE
          secondBeat = false;                  // clear secondBeat flag
```

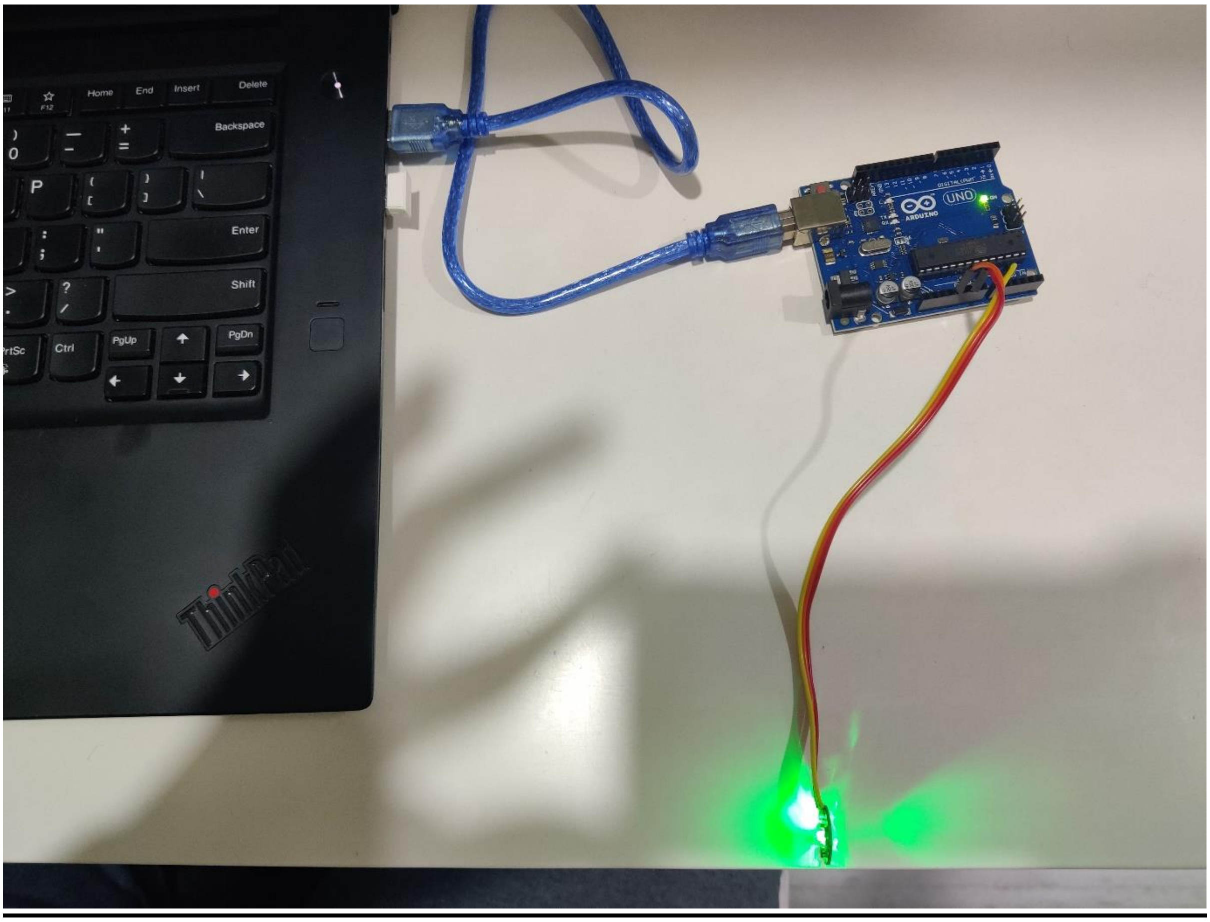
```
        for(int i=0; i<=9; i++) // seed the running total to get a realisitic BPM at startup
        {
          rate[i] = IBI;
        }
      }

      if(firstBeat) // if it's the first time we found a beat, if firstBeat == TRUE
      {
        firstBeat = false;                  // clear firstBeat flag
        secondBeat = true;                  // set the second beat flag
        sei();                              // enable interrupts again
        return;                             // IBI value is unreliable so discard it
      }
    // keep a running total of the last 10 IBI values
    word runningTotal = 0;                  // clear the runningTotal variable

    for(int i=0; i<=8; i++)
      {               // shift data in the rate array
        rate[i] = rate[i+1];                // and drop the oldest IBI value
        runningTotal += rate[i];            // add up the 9 oldest IBI values
      }
```

```
      rate[9] = IBI;                          // add the latest IBI to the rate array

      runningTotal += rate[9];                // add the latest IBI to runningTotal

      runningTotal /= 10;                     // average the last 10 IBI values

      BPM = 60000/runningTotal;               // how many beats can fit into a minute?
that's BPM!

      QS = true;                              // set Quantified Self flag

      // QS FLAG IS NOT CLEARED INSIDE THIS ISR

    }

  }


  if (Signal < thresh && Pulse == true)

    {   // when the values are going down, the beat is over

      digitalWrite(blinkPin,LOW);             // turn off pin 13 LED

      Pulse = false;                          // reset the Pulse flag so we can do it again

      amp = P - T;                            // get amplitude of the pulse wave

      thresh = amp/2 + T;                     // set thresh at 50% of the amplitude

      P = thresh;                             // reset these for next time

      T = thresh;

    }


  if (N > 2500)

    {                            // if 2.5 seconds go by without a beat

      thresh = 512;                           // set thresh default
```

```
    P = 512;                          // set P default
    T = 512;                          // set T default
    lastBeatTime = sampleCounter;        // bring the lastBeatTime up to date
    firstBeat = true;                    // set these to avoid noise
    secondBeat = false;                  // when we get the heartbeat back
  }

  sei();                               // enable interrupts when youre done!
}// end isr
```

CODE SCREENSHOT:

Hardware Screenshot:



Results Screenshot:

Conclusion and future scope:

This device allows a person to measure the heart rate in real time or record for later study. This also has application in portable ECG remote health monitors and fitness and heart rate monitors. So with the development of this system the elder patients will have the facility to get them diagnosed from the comfort of their home without traveling to hospital or in the situation where there is a dire need to visit the hospital in case of abnormalities that are detected on the device and after future developments might be able to send required information to nearby hospital facility about incoming patient and precautions to be taken and thus by eliminating potential risk of death

Reference:

[1] M. J Drinnan, J. Allen, and A. Murray (2001). "Relation between heart rate and pulse transit time during paced respiration", Physiol. Meas. vol. 22, pp425–432.

[2] "Range of Heart Rates per Minute and Average Heart Rate for Various Ages". Retrieve on 7th December 2008 (http://www.smm.org/heart/lessons/lesson1.htm)

[3] Allen J. (2007). "Photoplethysmography and its application in clinical physiological measurement", Physiol Meas. 2007 Mar; 28(3):R1-39. Epub 2007 Feb 20

[4] Shelley, K. H., R. G. Stout, et al. (1999). "The Use of Joint Time Frequency Analysis of The Pulse Oximeter Waveform to Measure The Respiratory Rate of Ventilated Patients." Anesthesiology 91(3A): A583.

[5] Péter Csordàs, Balázs Scherer (2006), "Development and Applications of A Home Health Monitoring Device", Proc. of International Carpathian Control Conference ICCC' 2006, May 29-31

[6] Mohd Fadzly Othman (2006). "Developing a Heartbeat Monitoring System Using PIC Microcontroller". KUiTTHO: Thesis B. Eng.

[7] Mohammad Faaiz Bin Jamaluddin (2008). "Wireless Heart Rate Monitor", University Teknology Malaysia, Thesis (B. Eng).

[8] Microchip Technology Inc. (2006). "MPLAB® IDE User's Guide".

[9] "PIC16F84A Microcontrollers Data Sheet". Retrieved on 5th September 2008 (http://www.microchip.com)

[10] S.Sara and Z.Bill. (1999) "Photoplethysmograph". Retrieved on 11th October 2008 (http://en.wikipedia.org/photoplethysmograph.)

[11]Jeong, In Cheol et al. (2006). "A new method to estimate arterial blood pressure using Photoplethysmographic signal", Conf. Proc. IEEE Eng Med Biol Soc 1:4667-70

[12] ECG Measurement System -Chia-Hung Chen, Shi-Gun Pan, Peter Kinget.

[13] Daniel Paulus / Thomas Meier,"ECG-Amplifier", MB Jass 2009.

[14] G.M.Patil, K.Subbarao, V.D.Mytri, A.D. Rajkumar, D.N.Reddy and K. "Embedded Microcontroller based Digital Telemonitoring System for ECG", J. Instrum.Soc. India 37(2) 134-149.

[15] Dr. P. K. Dash, "Electrocardiogram Monitoring", Indian J.Anaesth. 2002; 46 (4): 251-256.