Chapter 8 : IoT based Heart Attack Prevention System (BEAT-O-HEART)

8.1 Introduction to BEAT-O-HEART

Every year many peoples die because of heart attack only in United State 375,000 person die every year! there are many drug that calm down heart and prevent heart attack like nitroglycerin, ACE inhibitors, Beta blockers and ...! but they have to be injected to the body immediately. When an old lady sleeps at home and suddenly heart attack happens, she is really in a bad situation, because there is no doctor at home to help her and give her drug and if she is hospitalized, it takes time for nurses to hear the alarm and go to patient's room, take her drug and inject that to the her body!

So in such a case even a small delay in injecting the drug to the patient's body can cost a lot more. In such a situation an immediate solution has to be found out to save a life.
So, what can be a solution to this ?? …
IoT has got a lot of things in its store to save the human beings from such critical situations.

The first stage of heart attack is sudden change in heart beat rate. Suddenly it goes too high or too low. At this stage we detect it and predict heart attack by help of a microcontroller and a heart rate sensor, then microcontroller will inject drug to body, by controlling vale or pump that are placed at the middle of drug hose! Now controlling heart attack is faster easier and predictable.

8.2 Objectives of BEAT-O-HEART

- Calculation of the heart rate of a person.
- Capable of controlling the drug injection ,whenever the heart rate points at the case of need.
- Developing an alternative to the high cost equipment for monitoring heart rate.
- The system is capable of saving someone's life.

8.3 Project Architecture

The IoT based heart attack prevention system can be implemented in hospitals, clinics, schools, dispensaries or even at home .In this system there will be a heart

beat sensor module attached with arduino to measure the heart rate of a person in beats per minute (BPM) . Further , a 5V dc stepper motor will be interfaced along with heart beat sensor to control the injection of the fluid inside patient's body on the basis of the heart rate measured by the sensor and the micro-controller.

8.3.1 Software used in IoT based Heart Attack Prevention System

8.3.1.1 Arduino IDE

Arduino IDE (Integrated Development Environment) software is used to write and upload codes from the software to physical board. The language used in Arduino IDE is C++. Most of the libraries are written in C/C++. It runs on Windows, Mac and Linux. The software can be used with any arduino board.

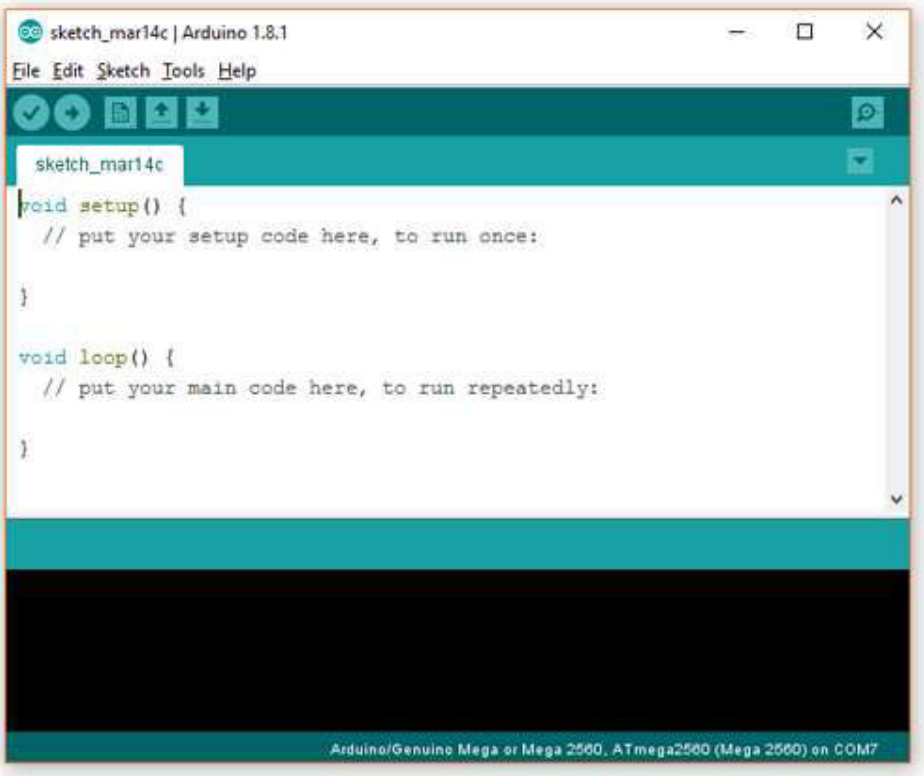

Figure 8.1 Arduino IDE Sketch

8.3.2 Hardware used in Project

8.3.2.1 Arduino Microcontroller

The Arduino Uno board is a microcontroller based on the ATmega328. It has 14 digital input/output pins in which 6 can be used as PWM outputs, a 16 MHz ceramic resonator, an ICSP header, a USB connection, 6 analog inputs, a power jack and a reset button .

Figure 8.2 Arduino Uno

8.3.2.2 HEART BEAT SENSOR MODULE

Heartbeat Sensor is an electronic device that is used to measure the heart rate i.e. speed of the heartbeat. Monitoring body temperature, heart rate and blood pressure are the basic things that we do in order to keep us healthy.

Monitoring heart rate is very important for athletes, patients as it determines the condition of the heart (just heart rate). There are many ways to measure heart rate and the most precise one is using an Electrocardiography

But the more easy way to monitor the heart rate is to use a Heartbeat Sensor. It comes in different shapes and sizes and allows an instant way to measure the heartbeat.

Heartbeat Sensors are available in Wrist Watches (Smart Watches), Smart Phones, chest straps, etc. The heartbeat is measured in beats per minute or bpm, which indicates the number of times the heart is contracting or expanding in a minute.

Figure 8.3 Heart beat sensor

8.3.2.3 Jump wires

Jumper wires are simply wires that have connector pins at each end, allowing them to be used to connect two points to each other without soldering. Jumper wires are typically used with breadboards and other prototyping tools in order to make it easy to change a circuit as needed Jumper *wires* are used for making connections between items on your breadboard and your Arduino's header pins.

Figure 8.4 Jumper Wires

8.3.2.4 Stepper Motor

Stepper motors are DC motors that move in discrete steps. They have multiple coils that are organized in groups called "phases". By energizing each phase in sequence, the motor will rotate, one step at a time.

With a computer controlled stepping you can achieve very precise positioning and/or speed control. For this reason, stepper motors are the motor of choice for many precision motion control applications.

Stepper motors come in many different sizes and styles and electrical characteristics. This guide details what you need to know to pick the right motor for the job.

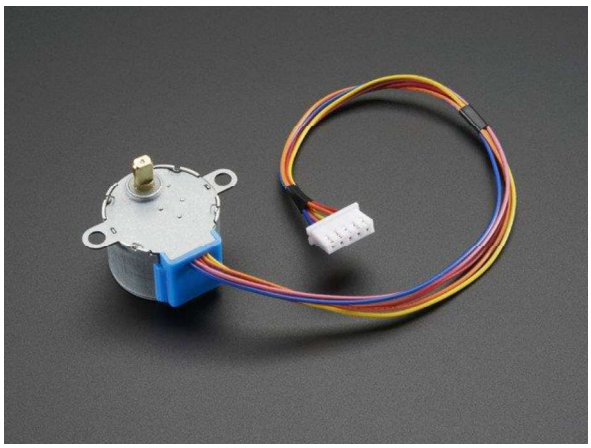

Figure 8.5 stepper motor

8.4 Project Implementation and working

In this project a heart rate sensor will be attached to patient's body and send heart rate to the microcontroller. Now microcontroller checks the data and show it on the LCD, if the data is good and normal , there will be no fluid flow but if it is bad , then the step motor that control the drug flow will rotate and increases the drug flow.

8.4.1 Interfacing of the components

8.4.1.1 Heat pulse sensor Pin Out

The heart beat sensor module has three pins

Figure 8.6 Pin-Out of heart beat sensor

Signal: This will be connected to the analog pin of the Arduino

5V: This will be connected to the 5V pin of the Arduino

GND: This will be connected to the ground of the Arduino

8.4.1.2 Heat beat sensor interfacing with arduino

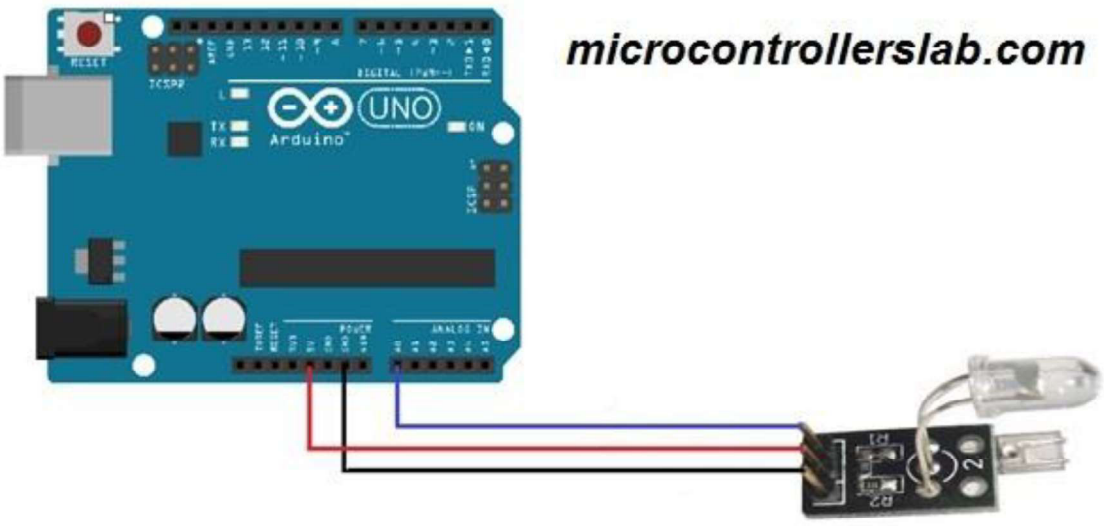

Figure 8.7 Interfacing of arduino and heart beat sensor

The connections are very easier. First connect the 5v and gnd pin of the module with the 5v and the gnd of the Arduino and then connect the sensor pin to the A0 of the Arduino.

8.4.1.3 Heart Beat sensor with LCD attached

In this example we will attached the heart pulse sensor with Arduino and a lcd. Arduino will control the whole system. Arduino will read the pulses from the heart beat sensor module and will calculate the heart rate and will show it to the LCD.The output pin of the heart beat sensor is connected to the pin 8 of Arduino. LCD is connected to Arduino in the 4-bit mode. VCC and gnd of sensor are connected to the vcc and gnd of the Arduino. When we will press the push button

the system will start to count the pulses.

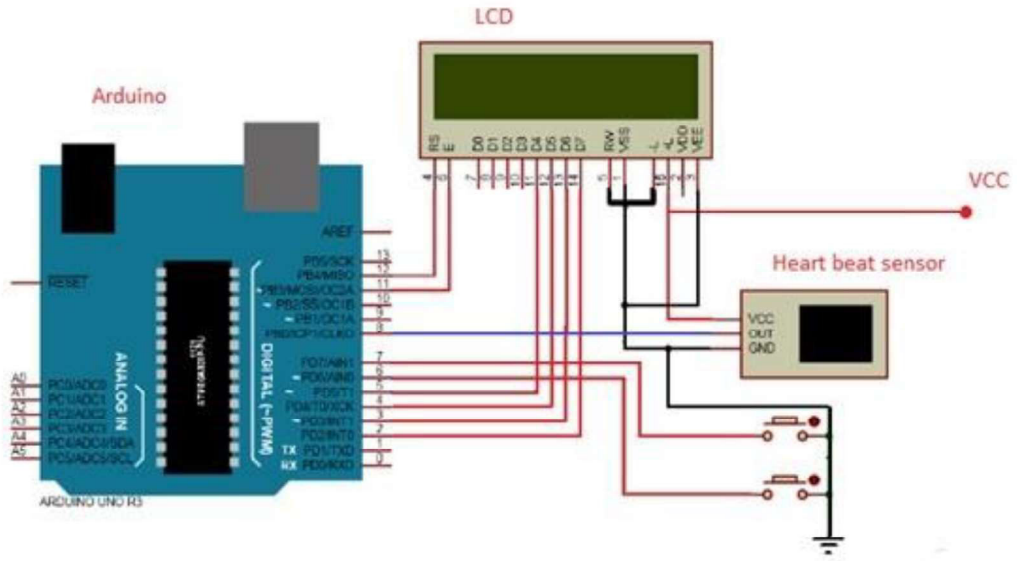

Figure 8.8 LCD interfacing with heart beat sensor and arduino

8.4.1.4 Stepper motor interfacing with arduino

Step motor (28BYJ-48) will be connected to its driver (ULN2003) easily by thair sockets. the driver's VIN to 5V of arduino, GND to GND, and IN 1,2,3,4 pins to digital pins A0,A1,A2,A3 of arduino.

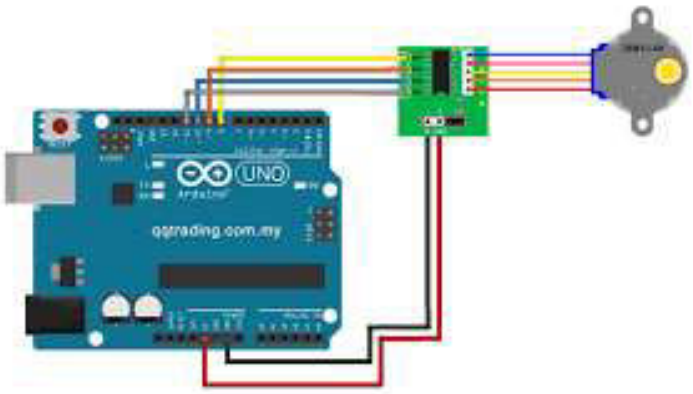

Figure 8.9 Stepper motor interfacing with arduino

8.4.2 Injection System

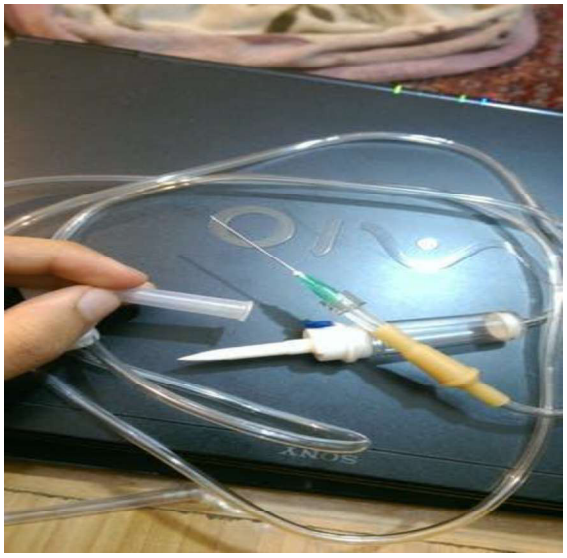

Figure 8.10 Injection system

For controlling the amount of drug that will be injected to patient's body. we use an step motor that works like an dialysis pump! This kind of pump works by pushing the liquid that is inside a hose by pressing it!

8.4.3 Working of the IoT based Heart Attack Prevention System

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

time2 is list pulse counter value

rate 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 time2 and then we substarct time1 from time2 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, deviding 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.

After the heart beat has been measured the value of heart beat is used to drive the stepper motor according to the need of the patient. If the noted heart beat is greater than LBP (60) or less than HBP(100) , then the patient is in normal condition and there will be no change in stepper motor. But if in case the detected rate is either greater than HBP or less than LBP , then the controller will automatically rotate the stepper motor to allow the flow of drug through valve and then after some delay , the stepper will fall back to its original position , thereby blocking the drug flow.

8.5 Codes used in the Project

```
#include<LiquidCrystal.h>

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

int in = 8;

int val;



#include <Stepper.h>

#define HBP 100

#define LBP 60

const int stepsPerRevolution = 32;  // change this to fit the number of steps per revolution

// for your motor



// initialize the stepper library on pins 8 through 11:

Stepper myStepper(stepsPerRevolution, A0, A1, A2, A3);




int Reset=6;

int start=7;

int count=0,i=0,k=0,rate=0;
```

```
unsigned long time2,time1;
unsigned long time;
byte heart[8] =
{
 0b00000,
 0b01010,
 0b11111,
 0b11111,
 0b11111,
 0b01110,
 0b00100,
 0b00000
};
void setup()
{
 lcd.createChar(1, heart);
 lcd.begin(16,2);

 lcd.print("Heart Beat ");
 lcd.write(1);
 lcd.setCursor(0,1);
 lcd.print("Monitering");

 pinMode(in, INPUT);

 pinMode(Reset, INPUT);
```

```
  pinMode(start, INPUT);
  digitalWrite(Reset, HIGH);
  digitalWrite(start, HIGH);



    myStepper.setSpeed(200);
  // initialize the serial port:
  pinMode(A0, OUTPUT);
  pinMode(A1, OUTPUT);
  pinMode(A2, OUTPUT);
  pinMode(A3, OUTPUT);

  Serial.begin(9600);
}

void inject()
{
  Serial.println("clockwise");
  val = 512;
  myStepper.step(val);
  delay(1000);
  myStepper.step(val);
  return;
}

void loop()
```

```
{
 if(!(digitalRead(start)))
 {
  k=0;
  lcd.clear();
  lcd.print("Please wait.....");
  while(k<5)
  {
   if(digitalRead(in))
   {
    if(k==0)
    time1=millis();
    k++;
    while(digitalRead(in));
   }
  }
    time2=millis();
    rate=time2-time1;
    rate=rate/5;
    rate=60000/rate;

    lcd.clear();
    lcd.print("Heart Beat Rate:");
    lcd.setCursor(0,1);
    lcd.print(rate);
    lcd.print(" ");
```

```
    lcd.write(1);
    if( (rate>HBP) || (rate < LBP))
    {
      inject();
    }
  Serial.println("WAIT FOR 1000 !");
  delay(1000);
    k=0;
    rate=0;
  }
 if(!digitalRead(Reset))
 {
  rate=0;
   lcd.clear();
    lcd.print("Heart Beat Rate:");
    lcd.setCursor(0,1);
    lcd.write(1);
    lcd.print(rate);
    k=0;

 }
}
```