HEALTH MONITORING SYSTEM using ESP8266 and ARDUINO

Project report submitted in full fulfillment of the requirement for the degree of

BACHELOR OF TECHNOLOGY

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

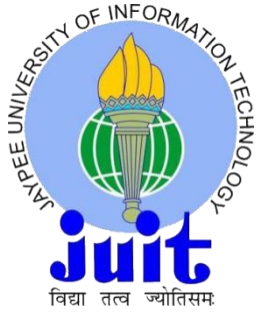
By

Sanyam Singla (171020)

Sheetanshu Saurabh (171027)

UNDER THE GUIDANCE OF

Dr.Alok Kumar



Department of Electronics and Communication Engineering

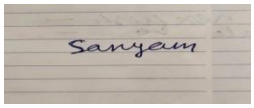
JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY WAKNAGHAT, SOLAN – 173234,
HIMACHAL PRADESH, INDIA

Candidate's Declaration

I hereby announce that the work presented in this report named "Health Monitoring System" in full fulfilment of the requirements for the honor of the degree of Bachelor of Technology in Electronics And Communication submitted in the Department of Electronics And Communication, Jaypee University of Information Technology, Waknaghat is my very own true document work completed over a period from July 2020 To June 2021 underneath the oversight of Dr. Alok Kumar.

Sanyam Singla

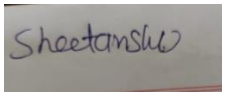
(171020)



(Student Signature)

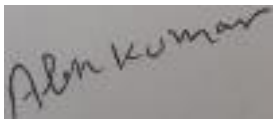
Sheetanshu Saurabh

(171027)



(Student Signature)

This is to ensure that the above confirmation made by candidates is consistent with the best of our insight.



Dr.Alok Kumar

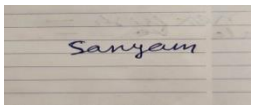
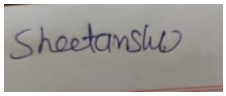
Department of Electronics and Communication

Date: 21/05/2021

ACKNOWLEDGEMENT

I should acknowledge the open way to thank and convey my deepest gratitude to our undertaking guide Alok Kumar for his tremendous help and significant direction without which it would not have been conceivable to reach at this phase of our fourth year venture.

Date:

Sanyam Singla (171020)

Sheetanshu Saurabh (171027)

TABLE OF CONTENTS

Contents

LIST OF FIGURES

ABSTRACT

Healthcare is a serious problem in this day and age. Due to the lack of official health monitoring, the patient experiences the negative effects of real medical problems. As we know there are dozens of iot devices in recent times to reveal affected person delight at the web. Obstetricians also use those sharp devices to monitor their patients. In this method, we are able to develop an iot device based totally on a health surveillance gadget that records inner and external patient beats (temperature). Pulse readings and internal temperatures were published on thingspeak and google pages in order that staying power can be seen everywhere in the international. As we can see, thingspeak is an open source net software for items with an api for storing and retrieving statistics on gadgets that use the http show at the net or on a neighborhood community.

CHAPTER 1

INTRODUCTION

1.1 Introduction

The extended use of flexible improvement and accurate devices in an green environment has created a large impact on the world. Success professionals are constantly exploiting the benefits that these developments bring, thus creating significant improvements in clinical thinking in clinical settings. Inside the same way, stop-of-life customers are provided from the blessings of m-fitness (mobile fitness) and e-fitness programs (clinical advantages maintained by ict) to improve, help and aid their prosperity.

In keeping with the arena fitness business enterprise (who) the most terrific and cheap welfare device is the high-quality man or women. As this really excites us, we try to propose a thinking framework that enhances the next resilient prosperity following a structure that uses sensors to trace irreversible and restless boundaries and uses the web to rehabilitate trained professionals so that they can help when problems quickly prevent passing costs.

Expertise health perspectives the usage of iot is a development that gives affected person care beyond the standard clinical settings (e.G. At domestic),that can increase intellectual retention, and decrease the value of handing over medical offerings. This can improve a person's health. It empowers patients to care for their well-being, prevent abuse, and pay for reunion costs. The device supports those regions by using moving direct care to the house. Similarly, patients and their own family individuals feel comforted understanding that they are being tested and that they'll be saved inside the occasion of a disaster.

1.2 Project Overview

Iot health monitoring has 2 sensors.First one is temperature and every other one is pulse charge sensor. This undertaking is crucial for the reason that professional can display screen information prosperity limits just by means of touring a site or url. Further extra nowadays diverse iot programs are furthermore being made . So now the skilled professional or own family contributors can screen or music the affected person’s prosperity through the android utility.

To work iot based totally prosperity checking shape journey, you need a wifi affiliation. The microcontroller or the arduino board partners with the wifi network using wifi module. This attempt will now not work with out an lively wifi business enterprise. You

can create a wifi area using a wifi module or you can create a wifi location the use of hotspot to your cellular. The arduino uno board examines the commitments from these . It then sends this records to the cloud by means of sending this data to a particular url/ip cope with. At that factor this procedure of sending statistics over ip is repeated after a while.For example on this activity we've got sent data after regular instances.

1.3 Block Diagram

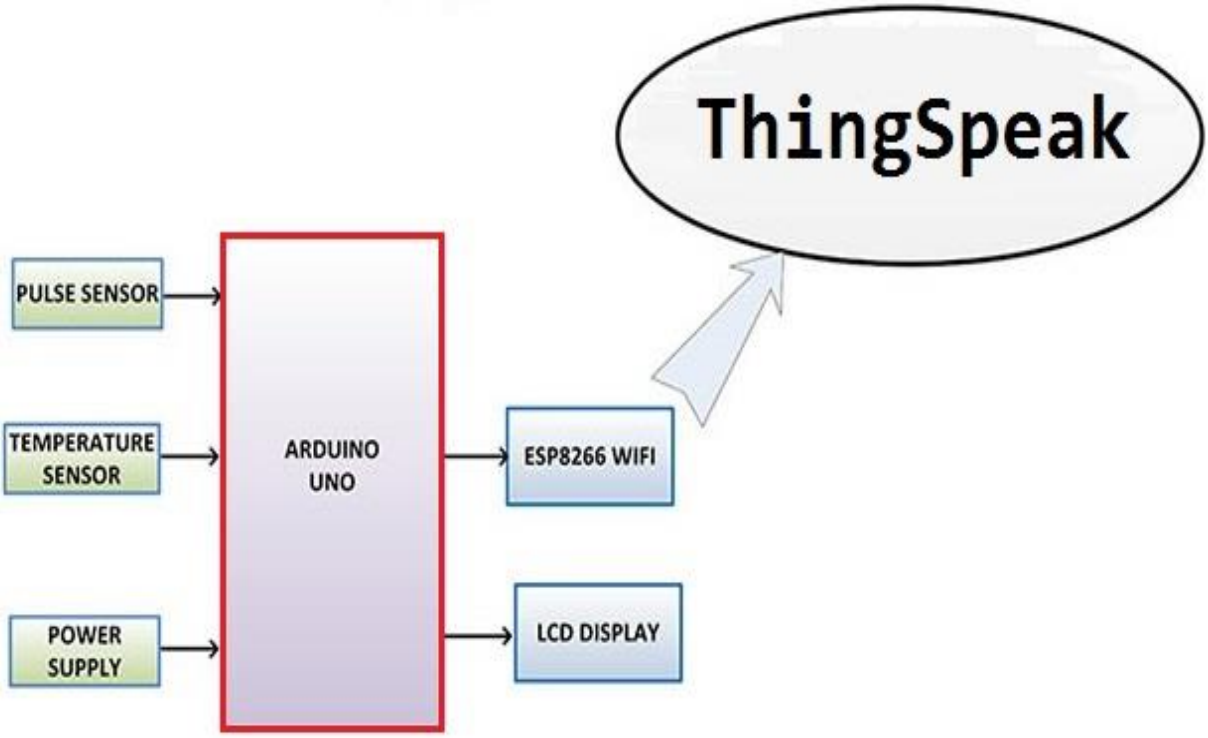


Figure 1.1:Block Diagram

1.4 Circuit Diagram:

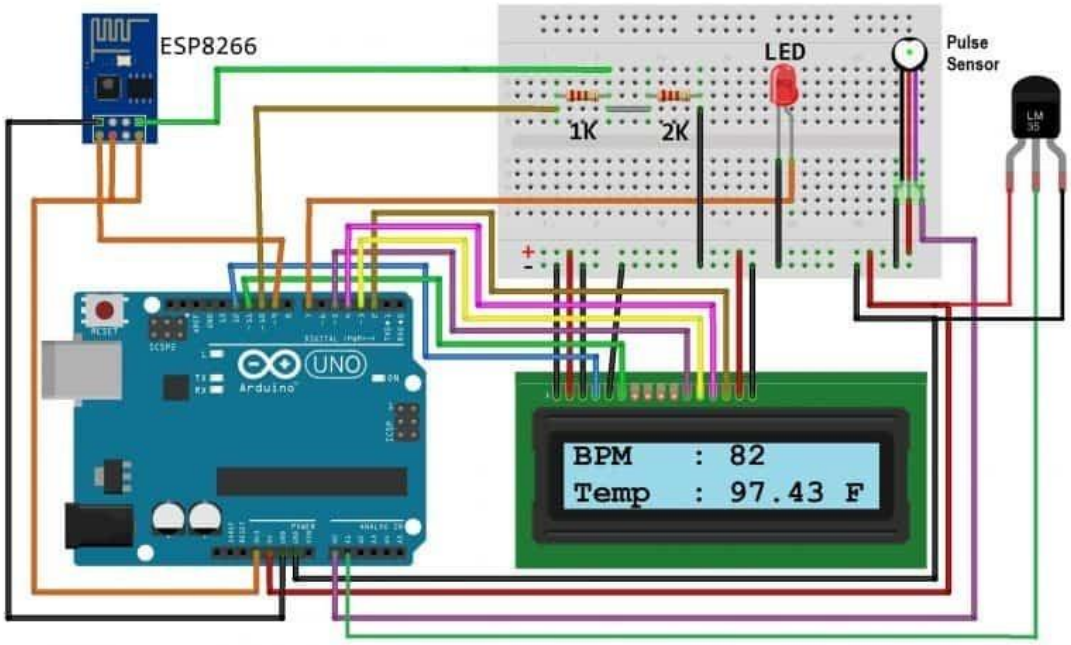


Figure 1.2:Circuit Diagram

To create an IoT primarily Based Health Monitoring gadgets using ESP8266 and Arduino, interface the circuit as validated with in the determine above.

1. Join the output pulse Sensor pin to the Arduino A3 and two other pins to the VCC and GND.

2. Join the output of the LM35 Temperature Sensor pin to the Arduino A1 and different pins to the VCC and GND.

3. Join Pin 1,5,16 for LCD and GND.

4. Join Pin 2 LCD and VCC.

5. Join Pin 4,6,11,12,13,14 for LCD and virtual Pin7,6,2,3,4,5 for arduino.

6.The rx pin of esp8266 works on the 3.3v and could now not speak to arduino whilst we well join it to arduino.In keeping with these traces,we ought to make energy separator so one can change over 5v to three.3v. On this way the rx pin of esp8266 is hooked up to eleventh arduino pin by using resistors.

7. Join the tx pin esp8266 to connect to 10 Arduino.

CHAPTER 2

2.1 Literature Review

- US. J. Jung and W. Y. Chung reviewed the dynamic and versatile patient monitoring framework in 6LoWPAN. The major benifit of this item is the ability to mix and match specific pre-programming. The Internet of Things is a collection of exercises gathered in various fields of knowledge such as media communications, informatics and gadgets.

- KS Shin and MJ Mao Kaiver explored a telephone-based health monitoring system by investigating IoT another global view that the use of articles is not only good for collecting weather data and working with the real world, but also for networking each other to trade data as data.

- Gennaro tartarisco and Tabilo Paniclo considered retaining the acquisition and availability of large-scale sensory networks that actually include information on how to integrate or construct a new computer-based device based on psychological support, data management, long-term communication and further data mining in new fields.

- Loren Schwiebert, Sandeep K.S. Gupta and Jennifer Weinmann are looking at the dynamic sensory energy created from the discovery compound next to the integrated hardware of other biological applications.

- Reza S.Dilmaghani (2016) in their trial set up a suitable Wi-Fi network system to view chronic patient illnesses in their home using a remote control. So immersion tests for new sensory testing such as heart rate, heart rate and so on can be limited but this test project puts this limit together to be limited to a single framework, and moreover all can be worn with tolerant and web-based information (IoT).

CHAPTER 3

IMPLEMENTATION

3.1 COMPONENTS USED

- Arduino Uno
- Esp8266
- Lcd Display
- Pulse Sensor
- Temperature Sensor
- Resistor
- Led
- Connecting Wires
- Bread Board

3.2 DESCRIPTION OF COMPONENTS USED

3.2.1 Arduino Uno

Arduino is an open source and company for planning, making and connecting customers planning and making singleboard microcontrollers and microcontroller packages for building modern devices. Content material is concern to the gnu lesser widespread public license(lgpl) or the gnu popular public license (gpl),which lets in for the introduction of arduino sheets and termination of applications via each person. Arduino sheets are presently available in a preassembled constructing or as you are making your personal with the assist of other gadgets (diy)devices.

Arduino board systems use chip time and controllers. Sheets are fitted with front row sets and straight data / crop (I / O) sets that can be interrupted on various continuous sheets ('shields') or bread boards (for prototyping) and various circuits. Sheets include a series of interconnected communications, including Universal Serial Bus (USB) to the undisputed models, used for installing applications on PCs. Microcontrollers can be replaced using C and C ++ vernaculars. In addition to using conventional interlocking metal chains, the Arduino sensor provides a set-up (IDE) setting that monitors processing language processing. Arduino's enjoy began in 2005 as a scholar software underneath the interplay layout institute ivrea in ivrea, italy, which required to provide insignificant attempt and simple preparation for college students and professionals to create demanding situations that discuss their situation using sensors and actuators. Frequent times of such conflict are

suggested by young professionals to join head robots, house controllers and head identifiers.

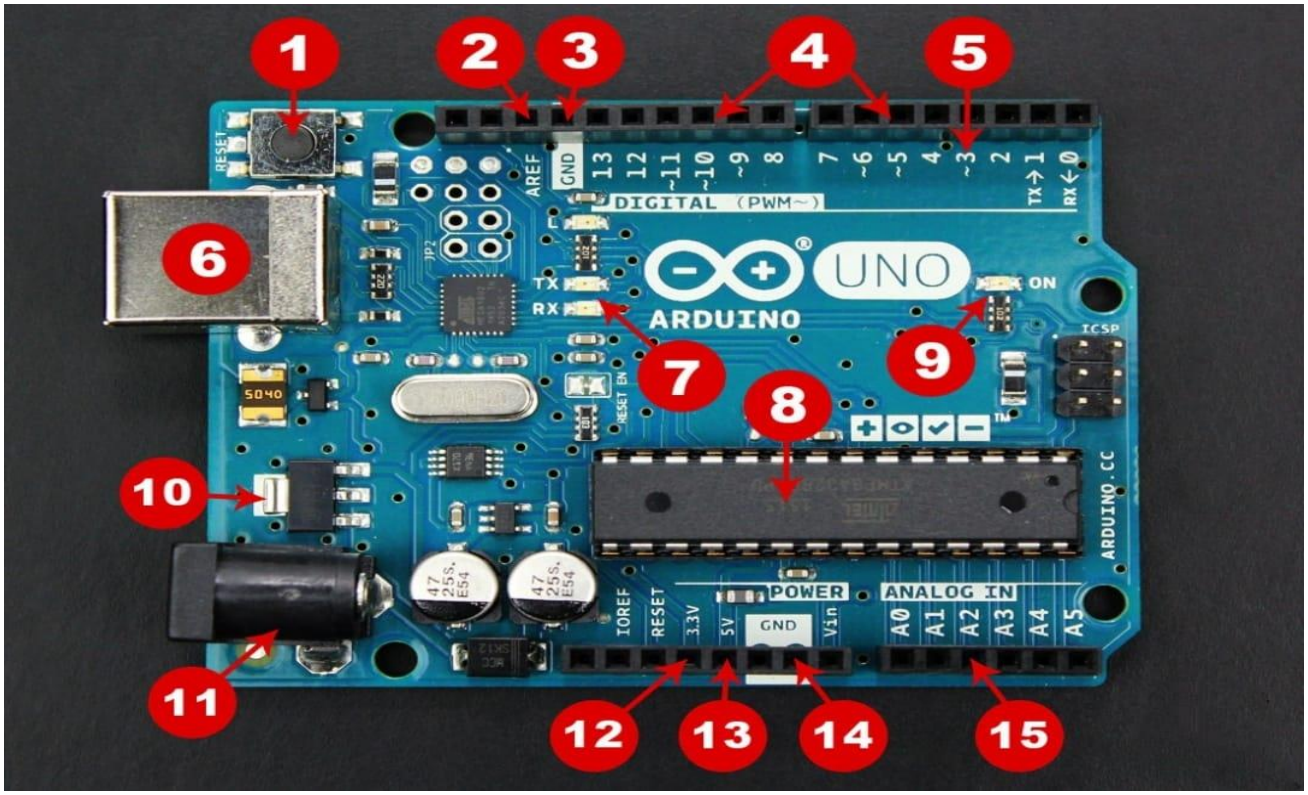


Figure 3.1:Arduino

3.2.1.1 Working of Arduino

1. **Reset Button –** It'll reset any code that is stacked to the arduino board
2. **AREF –** Approach "Analog Reference" and it's far applied to set an outer reference voltage
3. **Ground Pin –** Pins zeros-13 may be applied for virtual input or output
4. **Digital Input/Output –** Pins zeros-13 can be applied for virtual input or output
5. **PWM –** The pins marked with the (~)image can simulate analog output
6. **USB Connection –** Utilized by allowing your arduino and importing sketches
7. **TX/RX –** Those pins are used for transmit and receive facts indication LED's.
8. **ATmega Microcontroller –** This is the mind of the arduino wherein the packages are saved
9. **Power LED Indicator –** This Led bright up everytime the board is connected in a power supply
10. **Voltage Regulator -** This controls the amount of voltage going to the arduino board
11. **DC Power Barrel Jack –** This is utilized for enabling your arduino with a voltage deliver
12. **3.3V Pin –** It supply 3.3v of ability to our initiatives
13. **5V Pin –** It supply 5v of ability to our initiatives
14. **Ground Pins –** There are fewer pins of ground on the arduino and that all work something very comparable
15. **Analog Pins –** Those pins can stumble on the signal from an analog sensor and alternate it to virtual

3.2.2 Esp8266

Esp8266 is easy to use and simplifies the device to provide the web organization in your efforts. The module can serve both as an get admission to point(it is able to create an thrilling location) and as a channel (can connect with wifi) any more it could retrieve facts without the most surprising advice and take it to the web to make internet of things as straightforward as can be unexpectedly expected.It can similarly deliver data from the web using API's in a way that your effort can access any open information on the web,thus making it more efficient. Every other refreshing function of this module is that

it tends to be stepped forward the usage of the arduino ide which makes it very straightforward.

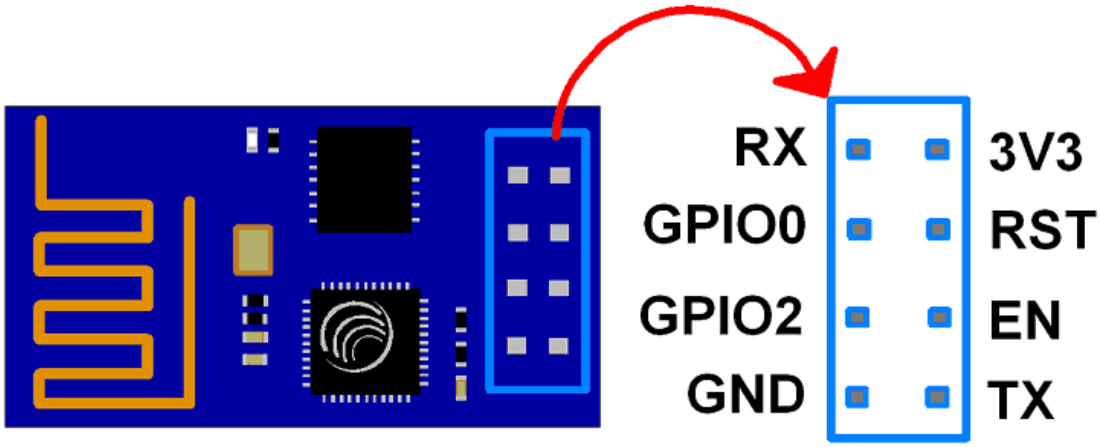


Figure 3.2:ESP8266

The esp8266 module works with just 3.3V, anything above 3.7V can kill the module hence becoming a warning to your circuits. Here is its anchor.

Pin 1:Ground: connected to the ground of the circuit

Pin 2:Tx/GPIO-1:linked to rx pin of programmer/uc to upload program

Pin 3:GPIO-2:general purpose input/output pin

Pin 4:CH_EN:chip enable/active high

Pin 5:Flash/GPIO-0:general purpose input /output pin

Pin 6:Reset:resets the module

Pin 7:RX/GPIO-3:general purpose input/output pin

Pin 8:Vcc:connect to +3.3v simplest

3.2.3 Pulse Sensor

Pulse Sensor is an internal and external programming organization for Arduino simulaton and play. The sensors cuts off the tip of the finger or ear and sticks directly to arduino. Additionally it is an open supply application that consists of your drawing constantly.

The front of the sensor is blanketed by a heart signal.This is the aspect that goes with the pores and skin.On the front you see a turning starting ,which is wherein the led passes thru the rear and there's a positive rectangular beneath the led.A rectangular is a overlaying bulb,similar to that used on cell telephones,capsules and places of work to exchange the beauty of the display screen in a variety of lighting conditions.The led illuminates mild from a finger or ear or different high-quality tissue and the sensor video display units the quantity of go back mild . It is the way it registers rhythm. The opposite facet is felt when different components are inserted.

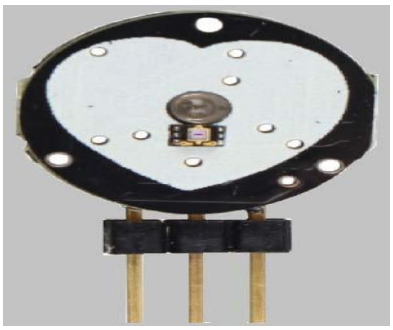


Figure 3.3:Front Pulse Rate

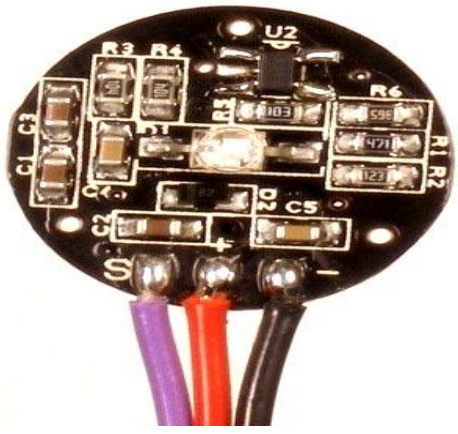


Figure 3.4:Back Pulse Rate

Earlier than using this sensor,we need to affirm the indicated aspect of the sensor of the sensor so that we can use it indiscriminately and live faraway from short discharges due to sweat. For this you could use a velcro tie or black tape. As shown within the photograph beneath.

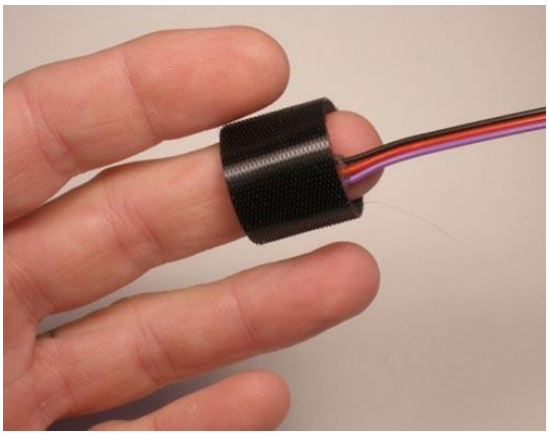


Figure 3.5:Pulse Rate

The three wires are emerging from the sensor:-

- Signal(S)
- Vcc(3 - 5 V) and
- GND.

3.2.4 Temperature Sensor

The LM35 game system specifies the combined circuit temperature with the yield voltage directly compared to the centigrade temperature range.The cracking of the Lm35 could affect the room in addition to the direct temperature sensors switching to Kelvin ,as the cutomer is not expected to turn off a reputable large-scale power supply that has received a good rate of Centigrade.The Lm35 device does not need to bother with any external configurations or to find out how it can provide standard specifications of ± ¼ ° C at room temperature and ± ¾ ° C over full temperature 55 ° C to 150 ° C range.

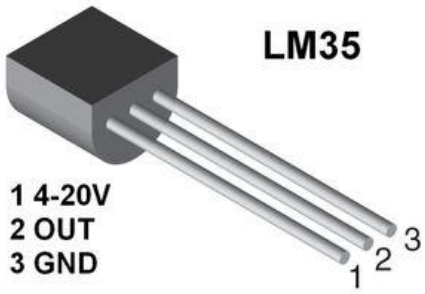


Figure 3.6:Temperature Lm35

3.2.5 LCD 16*2 Screen

Figure 3.7:Lcd

That is the lcd 16x2 parallel liquid crystal display display that offers a truthful and successful answer for including 16x2 white to rgb liquid crystal display in your paintings. The 16-line 2-line show has clear and differentiating highwhite substance over a blue/foundation light.

That is a vibrant lcd show with a blue background.Great for arduino paintings.This show of lcd 16x2 parallel lcd with back backlight is not tough to connect to arduino or different microcontroller.

3.3 Software Used

3.3.1 ThingSpeak

Thingspeak offers a savvy tool for iot sporting events.By means of this thingspeak internet site, we may additionally view our statistics and update our on-line framework ,the use of the channels and pages furnished with the aid of thingspeak . Info for thingspeak ‘stack’ as of sensors,’dissect and image’ subtleties and ‘sports’ by deleting

grievance.We used thingspeak late on the raspberry pi weather station and using arduino, trying to see thingspeak.Right here we quick introduce the usage of thingspeak for this iot impatient tracking improvement .

We make use of the thingspeak at the screen to discover coronary heart rate and the temperature using the internet electronically. Similarly we use the ifttt section to hyperlink thingspeak to an e mail/message businesses hence a cautioning message may be ship at anything factor the patient is in dangerous circumstance.

Step 1:- As a matter of first importance, person requirements to make an Account on ThingSpeak.com, then Sign In and hit it off to Start.

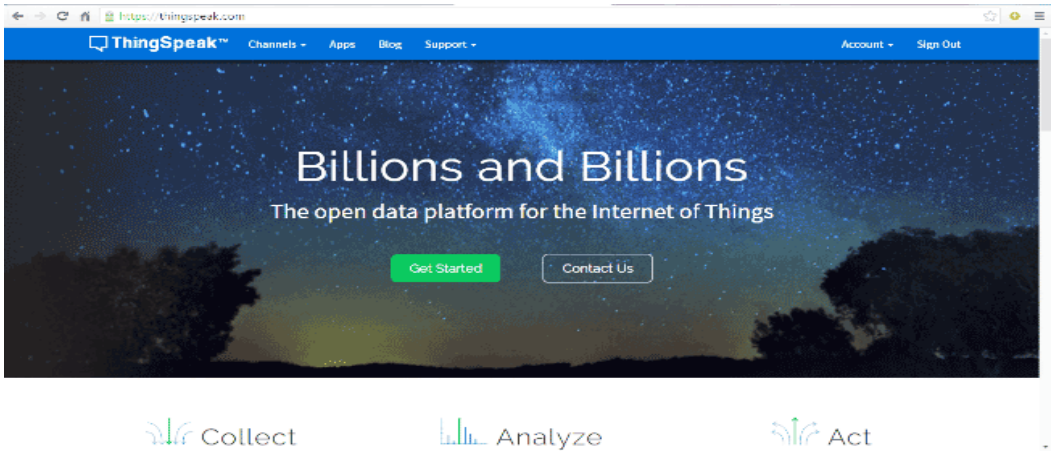


Figure 3.8:Thingspeak Account

Step 2:- Presently pass to the 'channels' menu and come to a selection on a new channel in complete agreement for the corresponding cycle.

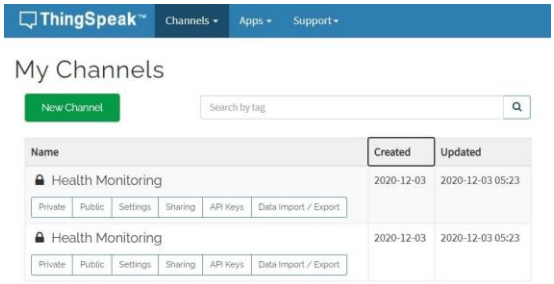


Figure 3.9:Thingspeak channel

Step3:- Inside the suggest time you wil see the channel application,fill out the call and display as shown on your decision.Top off ‘heartbeat price’ and ‘temperature’ in field 1

and subject 2 imprints, check the fields test bins.Tick the check container for ‘display’ below the properties at the end of keep channel.From now on your new channel has been created.

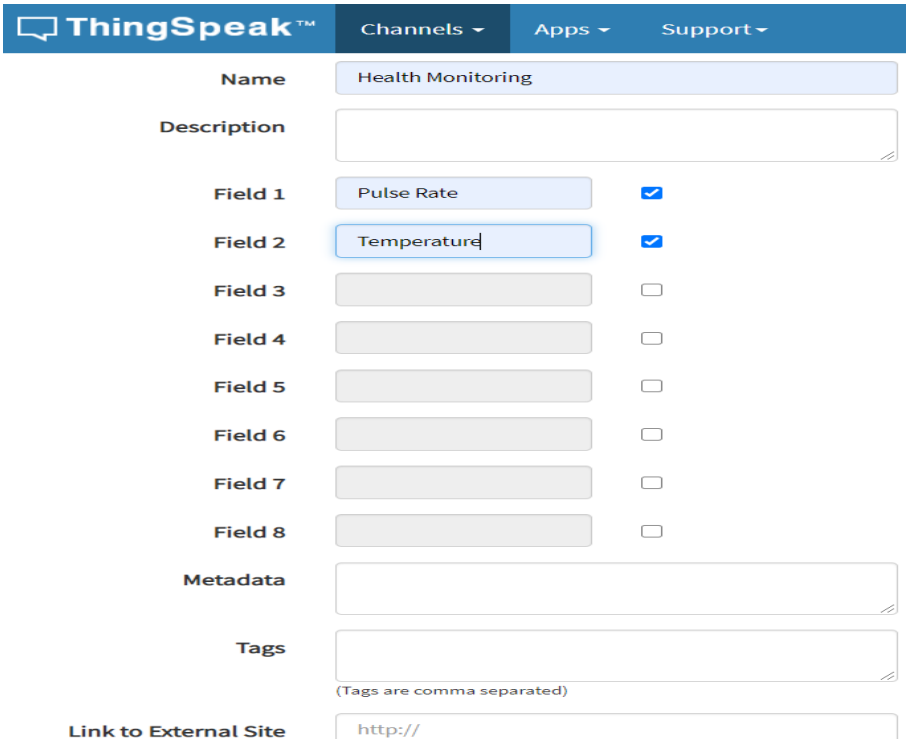


Figure 3.10:Thingspeak Creating

Step 4:- You will consider two charts as shown underneath. We will utilize this API key, in our code.

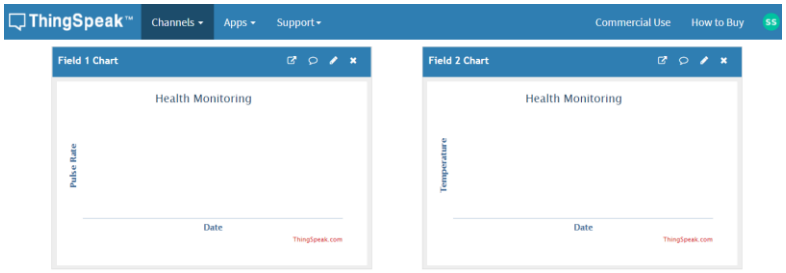


Figure 3.11:Pulse Output

3.3.2 Integrated Development Environment

Figure 3.12:Arduino Ide

Integrated Development Environment (IDE) is an operating system for Windows, MacOS, Linux, etc. With this software we can do programming in C and C ++ languages. It helps to create and transfer projects on arduino sheets.

Under general public license we can discharge source source code for the IDE. Arduino ide holds up lingos c and c ++ the usage of exceptional code integration strategies. Arduino ide gives library substances from wiring journey, providing a spread of information and strategies.The created code required only two restrictions, starting with a drawing and an important program circle, integrated and associated with the system system guide () in the username and other public authorization tool tool, what else is integrated by way of the Arduino IDE movement. Arduino IDE uses the code to change the code that can be made into a record book in a hexadecimal installation that is embedded in the Arduino board with the upload system in the firmware of the board.

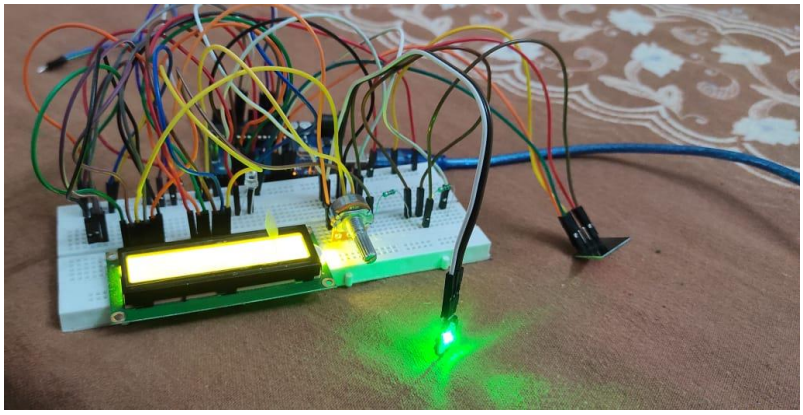


Figure 3.13:Circuit

CHAPTER 4

APPLICATIONS AND ADVANTAGES AND DISADVANTAGE

4.1 Applications

- Io Healthcare is a much-needed field in the medical field. This project is for older people in our homes. The same is true for an older citizen who lives alone or who lives with one or two people. This project proves to be very useful when relatives need to move out to work low maintenance.
- Patients with disabilities can use this function as patients who think it is difficult to go to a specialist consistently or to those patients who need further clinical evaluation.

4.2 Advantages

- **Remote monitoring**: Continuous far away seeing through related IoT devices and smart alerts can break down afflictions, treat sicknesses and extra lives if there ought to emerge an event of a health related emergency.
- **Prevention**: Smart sensors investigate medical problem, health and climate decisions and suggest preventive measures, which will reduce the incidence of disease and severe conditions.
- **Reduction of healthcare costs**: IoT reduces the cost of expensive professional visits and clinical confirmation and makes testing more modest.
- **Medical data accessibility**: Access to electronic clinical records allows patients to be considered for quality and helps medical care providers to resolve appropriate clinical decisions and complications of prevention.
- **Improved treatment management**: IoT gadgets help track medication planning and response to treatment and reduce clinical error.
- **Improved healthcare management:**Using IoT gadgets, medical professionals can obtain valuable data about hardware and staff adequacy and use it to recommend improvements.
- **Research**: Since IoT gadgets can gather and examine a gigantic measure of information, they have a high potential for clinical examination purpose.

4.3 Disadvantages

- **Security and privacy**: Security and protection stay a significant concern preventing clients from utilizing IoT innovation for clinical purposes, as medical services observing arrangements can possibly be penetrated or hacked. The hole of touchy data about the patient's wellbeing and area and interfering with sensor information can have grave outcomes, which would counter the advantages of IoT.
- **Risk of failure**: Disappointment or bugs in the equipment or even force disappointment can affect the exhibition of sensors and associated gear setting medical services tasks in danger. Also, avoiding a booked programming update might be considerably more perilous than skirting a specialist exam.
- **Integration**: There's no agreement with respect to IoT conventions and norms, so gadgets created by various producers may not function admirably together. The absence of consistency forestalls full-scale mix of IoT, subsequently restricting its possible viability.
- **Cost**: While IoT vows to diminish the expense of medical services in the long haul, the expense of its usage in clinics and staff preparing is very high.

CHAPTER 5

5.1 Conclusion

The Internet of Things is currently seen as one of the most useful answers to any long distance calling especially in the field of social security. Enabling that each achievement limit data is validated within the cloud, clinical stay is reduced to standardized tests and is so large that prosperity can be observed and disruptions are spread by any specialist at any stage. The frame is made. The system detected body temperature, heart rate, and temperature sensors, which also appear on LCDs. Sensor measurements are then sent out of clinical practice using remote communication. This data is then obtained from an approved mobile developer through the IoT category. With qualifications I found a specialist at the time to break the disease and the condition of the sufficiency of the patient.

5.2 Future development

We are able to add a gps module iot monitoring help that appears to use the arduino uno task and wifi module.This gps module will decide the location of the patient using longitude and distance.Then it will ship the location to the iot cloud the use of a wifi module.Then professionals can decide the patient’s condition in case they want to do something.

References

[1]. Electronicwings.com, https://www.electronicwings.com

[2]. Secured Smart Healthcare Monitoring System Based on Iot, International Journal on Recent and Innovation Trends in Computing and Communication Volume: 3 Issue: 7, Bhoomika.B.K, Dr. K N Muralidhara.

[3]. S. J. Jung and W. Y. Chung, "Flexible and scalable patient's health monitoring system in 6LoWPAN," Sensor Lett., vol. 9,no. 2, pp. 778–785, Apr. 2011.

[4]. S. M. Mahalle, P. V. Ingole, "Design and Implementation of Wireless Body Area Sensor Network Based Health Monitoring System", International Journal of Engineering Research & Technology, Vol. 2 Issue 6, pp. 105- 113, June 2013

[5]. Dr.A.Sabanayagam, G.Anish Girija," DESIGN AND MODELING OF MOBILE HEALTH MONITORING SYSTEM", International Journal of Innovations in Scientific and Engineering Research (IJISER),vol4,no 2,pp.63- 65,2017

[6]. Goutam Motika, Abinash Prusty," Wireless FetalHeartbeat Monitoring System Using ZigBee & IEEE 802.15.4 Standard", 2011 Second International Conference on Emerging Applications of Information Technology, 978-0- 7695-4329-1/11, 2011 IEEE DOI 10.1109/EAIT.2011.89 .

[7]. C. Kim, A. Soong, M. Tseng, and X. Zhixian, "Global wireless machineto- machine normalization," IEEE Internet Comput., vol. 15, no. 2, pp. 64–69, Mar.–Apr. 2011.

Appendix

sketch_may19a | Arduino 1.6.12

File Edit Sketch Tools Help

sketch_may19a §

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
#include <SoftwareSerial.h>
float pulse = 0;
float temp = 0;
SoftwareSerial ser(9,10);
String apiKey = "VSDI50FO97D51MDR";


int pulsePin = A0;
int blinkPin = 7 ;
int fadePin = 13;
int fadeRate = 0;

volatile int BPM;
volatile int Signal;
volatile int IBI = 600;
volatile boolean Pulse = false;
volatile boolean QS = false;


static boolean serialVisual = true;
volatile int rate[10];
volatile unsigned long sampleCounter = 0;
volatile unsigned long lastBeatTime = 0;
volatile int P = 512;
volatile int T = 512;
volatile int thresh = 525;
volatile int amp = 100;
volatile boolean firstBeat = true;
volatile boolean secondBeat = false;

void setup()
{
lcd.begin(16, 2);
pinMode(blinkPin,OUTPUT);
pinMode(fadePin,OUTPUT);
Serial.begin(11);
interruptSetup();
lcd.clear();
lcd.setCursor(0,0);
lcd.print(" Patient Health");
lcd.setCursor(0,1);
lcd.print(" Monitoring ");
delay(4000);
```

sketch_may19a | Arduino 1.6.12

File Edit Sketch Tools Help

sketch_may19a §

```
lcd.setCursor(0,1);
lcd.print(" Monitoring ");
delay(4000);
lcd.clear();
lcd.setCursor(0,0);
lcd.print("Initializing....");
delay(5000);
lcd.clear();
lcd.setCursor(0,0);
lcd.print("Getting Data....");
ser.begin(9600);
ser.println("AT");
delay(1000);
ser.println("AT+GMR");
delay(1000);
ser.println("AT+CWMODE=3");
delay(1000);
ser.println("AT+RST");
delay(5000);
ser.println("AT+CIPMUX=1");
delay(1000);

String cmd="AT+CWJAP=\"Redmi Note 9 Pro Max\",\"0987654321\"";
ser.println(cmd);
delay(1000);
ser.println("AT+CIFSR");
delay(1000);
}


void loop()
{
serialOutput();
if (QS == true)
{

fadeRate = 255;
serialOutputWhenBeatHappens();
QS = false;
}
ledFadeToBeat();
delay(20);
read_temp();
esp_8266();
}
```

sketch_may19a | Arduino 1.6.12

File Edit Sketch Tools Help

sketch_may19a §

```
}
ledFadeToBeat();
delay(20);
read_temp();
esp_8266();
}
void ledFadeToBeat()
{
fadeRate -= 15;
fadeRate = constrain(fadeRate,0,255);
analogWrite(fadePin,fadeRate);
}
void interruptSetup()
{

TCCR2A = 0x02;
TCCR2B = 0x06;
OCR2A = 0X7C;
TIMSK2 = 0x02;
sei();
}
void serialOutput()
{
if (serialVisual == true)
{
arduinoSerialMonitorVisual('-', Signal);
}
else
{
sendDataToSerial('S', Signal);
}
}
void serialOutputWhenBeatHappens()
{
if (serialVisual == true)
{
Serial.print("*** Heart-Beat Happened *** ");
Serial.print("BPM: ");
Serial.println(BPM);
}
else
{
sendDataToSerial('B',BPM);
sendDataToSerial('Q',IBI);
}
```

sketch_may19a | Arduino 1.6.12

File Edit Sketch Tools Help

sketch_may19a §

```
}
void arduinoSerialMonitorVisual(char symbol, int data )
{
const int sensorMin = 0;
const int sensorMax = 1024;
int sensorReading = data;
int range = map(sensorReading, sensorMin, sensorMax, 0, 11);

switch (range)
{
case 0:
Serial.println("");
break;
case 1:
Serial.println("---");
break;
case 2:
Serial.println("------");
break;
case 3:
Serial.println("---------");
break;
case 4:
Serial.println("------------");
break;
case 5:
Serial.println("--------------|-");
break;
case 6:
Serial.println("--------------|---");
break;
case 7:
Serial.println("--------------|-------");
break;
case 8:
Serial.println("--------------|----------");
break;
case 9:
Serial.println("--------------|----------------");
break;
case 10:
Serial.println("--------------|--------------------");
break;
case 11:
Serial.println("--------------|------------------------");
```


sketch_may19a | Arduino 1.6.12
File Edit Sketch Tools Help
sketch_may19a §
break;
case 11:
Serial.println("--------------|-----------------------");
break;
}
}
void sendDataToSerial(char symbol, int data)
{
Serial.print(symbol);
Serial.println(data);
}
ISR(TIMER2_COMPA_vect)
{
cli();
Signal = analogRead(pulsePin);
sampleCounter += 2;
int N = sampleCounter - lastBeatTime;
if(Signal < thresh && N > (IBI/5)*3)
{
if (Signal < T)
{
T = Signal;
}
}
if(Signal > thresh && Signal > P)
{
P = Signal;
}
if (N > 250)
{
if ((Signal > thresh) && (Pulse == false) && (N > (IBI/5)*3))
{
Pulse = true;
digitalWrite(blinkPin,HIGH);
IBI = sampleCounter - lastBeatTime;
lastBeatTime = sampleCounter;
if(secondBeat)
{
secondBeat = false;

sketch_may19a | Arduino 1.6.12

File Edit Sketch Tools Help

sketch_may19a §

```
{
secondBeat = false;
for(int i=0; i<=9; i++)
{
rate[i] = IBI;
}
}
if(firstBeat)
{
firstBeat = false;
secondBeat = true;
sei();
return;
}

word runningTotal = 0;
for(int i=0; i<=8; i++)
{
rate[i] = rate[i+1];
runningTotal += rate[i];
}
rate[9] = IBI;
runningTotal += rate[9];
runningTotal /= 10;
BPM = 60000/runningTotal;
QS = true;

pulse = BPM;
}
}
if (Signal < thresh && Pulse == true)
{
digitalWrite(blinkPin,LOW);
Pulse = false;
amp = P - T;
thresh = amp/2 + T;
P = thresh;
T = thresh;
}
if (N > 2500)
{
thresh = 512;
P = 512;
T = 512;
lastBeatTime = sampleCounter;
```

sketch_may19a | Arduino 1.6.12

File Edit Sketch Tools Help

sketch_may19a §

```
firstBeat = true;
secondBeat = false;
}
sei(); // enable interrupts when youre done!
}// end isr
void esp_8266()
{
String cmd = "AT+CIPSTART=4,\"TCP\",\"184.106.153.149";
cmd += "184.106.153.149";
cmd += "\",80";
ser.println(cmd);
Serial.println(cmd);
if(ser.find("Error"))
{
Serial.println("AT+CIPSTART error");
return;
}
String getStr = "GET /update?api_key=VSDI50FO97D51MDR";
getStr += apiKey;
getStr +="&field1=";
getStr +=String(temp);
getStr +="&field2=";
getStr +=String(pulse);
getStr += "\r\n\r\n";
// send data length
cmd = "AT+CIPSEND=4,";
cmd += String(getStr.length());
ser.println(cmd);
Serial.println(cmd);
delay(1000);
ser.print(getStr);
Serial.println(getStr);
delay(3000);
}
void read_temp()
{
int temp_val = analogRead(A1);
float mv = (temp_val/1024.0)*5000;
float cel = mv/10;
temp = (cel*9)/5 + 32;
Serial.print("Temperature:");
Serial.println(temp);
lcd.clear();
lcd.setCursor(0,0);
lcd.print("BPM :");
```

sketch_may19a | Arduino 1.6.12

File Edit Sketch Tools Help

sketch_may19a §

```
cmd += "\",80";
ser.println(cmd);
Serial.println(cmd);
if(ser.find("Error"))
{
Serial.println("AT+CIPSTART error");
return;
}
String getStr = "GET /update?api_key=VSDI50FO97D51MDR";
getStr += apiKey;
getStr +="&field1=";
getStr +=String(temp);
getStr +="&field2=";
getStr +=String(pulse);
getStr += "\r\n\r\n";
// send data length
cmd = "AT+CIPSEND=4,";
cmd += String(getStr.length());
ser.println(cmd);
Serial.println(cmd);
delay(1000);
ser.print(getStr);
Serial.println(getStr);
delay(3000);
}
void read_temp()
{
int temp_val = analogRead(A1);
float mv = (temp_val/1024.0)*5000;
float cel = mv/10;
temp = (cel*9)/5 + 32;
Serial.print("Temperature:");
Serial.println(temp);
lcd.clear();
lcd.setCursor(0,0);
lcd.print("BPM :");
lcd.setCursor(7,0);
lcd.print(BPM);
lcd.setCursor(0,1);
lcd.print("Temp.:");
lcd.setCursor(7,1);
lcd.print(temp);
lcd.setCursor(13,1);
lcd.print("F");
}
```


```
--------------|-
*** Heart-Beat Happened *** BPM: 50
Temperature:781.71
AT+CIPSTART=4,"TCP","184.106.153.149184.106.153.149",80
AT+CIPSEND=4,83
GET /update?api_key=VSDI50FO97D51MDRVSDI50FO97D51MDR&field1=781.71&field2=50.00


--------------|-
Temperature:781.71
AT+CIPSTART=4,"TCP","184.106.153.149184.106.153.149",80
AT+CIPSEND=4,83
GET /update?api_key=VSDI50FO97D51MDRVSDI50FO97D51MDR&field1=781.71&field2=50.00


---------
Temperature:781.71
AT+CIPSTART=4,"TCP","184.106.153.149184.106.153.149",80
AT+CIPSEND=4,83
GET /update?api_key=VSDI50FO97D51MDRVSDI50FO97D51MDR&field1=781.71&field2=50.00


--------------|---
*** Heart-Beat Happened *** BPM: 136
Temperature:781.71
AT+CIPSTART=4,"TCP","184.106.153.149184.106.153.149",80
AT+CIPSEND=4,84
GET /update?api_key=VSDI50FO97D51MDRVSDI50FO97D51MDR&field1=781.71&field2=136.00


--------------|-
Temperature:779.95
AT+CIPSTART=4,"TCP","184.106.153.149184.106.153.149",80
AT+CIPSEND=4,84
GET /update?api_key=VSDI50FO97D51MDRVSDI50FO97D51MDR&field1=779.95&field2=136.00
```