"DESIGN OF TELE-HEALTH CARE MODEL"

A

Minor Project Report

Submitted in partial fulfilment for the degree of

BACHELORS OF TECHNOLOGY

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

Submitted by:
B-Tech VI semester

Jayant Kumar Mishra	**(0201EC191041)**
Krishnkant Kushwah	**(0201EC191043)**
Nupur Singh	**(0201EC191050)**
Om Rastogi	**(0201EC191051)**
Shrikant Kushwaha	**(0201EC203D08)**

Under the supervision of:

DR. AGYA MISHRA

Department of Electronics and Communication Engineering

Jabalpur Engineering College, Jabalpur (M.P.)

Rajiv Gandhi Proudyogiki Vishwavidhyalaya, Bhopal (M.P.)

SESSION: JAN-JUN 2022

Department Of Electronics and Communication Engineering
Jabalpur Engineering College, M.P.

CANDIDATE'S DECLARATION:

We hereby declare that the work which is being presented in the project report entitled "DESIGN OF TELE-HEALTH CARE MODEL" is an authentic record of our own work carried under the guidance of Dr. Agya Mishra, Professor, Department of Electronics and Communication Engineering, Jabalpur Engineering College, Jabalpur. We have not submitted the matter embodied in the report for the award of any other degree.

Jayant Kumar Mishra (020EC191041)
Krishnkant Kushwah (0201EC191043)
Nupur Singh (0201EC191050)
Om Rastogi (0201EC191051)
Shrikant Kushwaha (0201EC203D08)

Date:

Place:

Department Of Electronics and Communication Engineering
Jabalpur Engineering College, Jabalpur M.P.

CERTIFICATE

It is very well certified that **Jayant Kumar Mishra, Krishnkant Kushwah, Nupur Singh, Om Rastogi, Shrikant Kushwaha** students of VI Semester Electronics and Communication Engineering has accomplished minor project work entitled **"DESIGN OF TELE-HEALTH CARE MODEL"** under my supervision in sixth semester. They have been regularly attending the college during the project work.

Dr. Agya Mishra

Professor and Guide
Dept. of Electronics and Communication Engineering
Jabalpur Engineering College

Dr. Bhavana Jharia

HOD
Dept. of Electronics and Communication Engineering
Jabalpur Engineering College

Department Of Electronics and Communication Engineering

Jabalpur Engineering College, M.P.

CERTIFICATE OF APPROVAL

This it to certify that the Project Report entitled "DESIGN OF TELE-HEALTH CARE MODEL" submitted by: Jayant Kumar Mishra, Krishnkant Kushwah, Nupur Singh, Om Rastogi and Shrikant Kushwaha may be accepted for the partial fulfilment for the award of degree of bachelor of Engineering in Electronics and Communication.

Internal Examiner

Date:

External Examiner

Date:

ACKNOWLEDGEMENT

*It is our great fortune that we have got the opportunity to carry out this project work under the supervision of **Dr. Agya Mishra**, Professor in the Department of Electronics and communication Engineering, Jabalpur Engineering College. We express our sincere thanks and deepest sense of gratitude to our guide for her constant support, unparalleled guidance and encouragement.*

*We wish to convey our gratitude to **Dr. Bhavana Jharia,** Professor & Head, Department of Electronics and Communication Engineering, Jabalpur Engineering College, and all other teachers and staff member of Electronics and Communication Engineering, for their invaluable advice and assistance during our studies at Jabalpur Engineering College, Jabalpur.*

*We extent our sincere gratitude to our principal **Dr. P.K. Jhinge**, for his encouragement and support.*

We are indebted to our friends and seniors for their encouragement and support.

Last but not the least our gratitude is devoted to our parents, almighty God and family. They have always been the source and inspiration in our lives.

Jayant Kumar Mishra (020EC191041)
Krishnkant Kushwah (0201EC191043)
Nupur Singh (0201EC191050)
Om Rastogi (0201EC191051)
Shrikant Kushwaha (0201EC203D08)

ABSTRACT

Tele-Healthcare monitoring system in hospitals and many other health centers has experienced significant growth, and portable healthcare monitoring systems with emerging technologies are becoming of great concern to many countries worldwide nowadays. The advent of Internet of Things (IoT) technologies facilitates the progress of healthcare from face-to-face consulting to telemedicine. This project report proposes a smart healthcare system in IoT environment that can monitor a patient's basic health signs in real-time. In this system, various sensors are used to capture the data by sensing the health conditions of patient. The condition of the patients is conveyed to the doctor and family via IOT portal developed at Blynk platform, where they can process and analyze the current situation of the patients. The developed prototype is well suited for healthcare monitoring that is proved by the effectiveness of the system.

Contents

CHAPTER 1

INTRODUCTION

1.1. Overview

Health is characterized as a full state of physical, mental, and social well-being and not merely a lack of illness. Health is a fundamental element of people's need for a better life. As India is growing and developing day by day and so its villages and cities. Be it the sector of education or skill development. The main aspects that still lags behind and less concentrated is health. Especially in rural areas, due to lack of awareness people tend to avoid their small problems and ignore the symptoms leading to worst heath issues. Moreover, after the COVID era it has now really become essential to have regular health checkups and visit the physician more frequently. Again, a problem arrives when we concern about rural areas that they still lack medical facilities and lack of educated practitioners. And it becomes unaffordable for the village people to visit the expensive towns again and again.

The world is changing with technology and INTERNET OF THINGs being one of the biggest contributors of this growing era. So, with this project we aim to bring the internet if things and cloud-based technology together to deal we a big problem. Especially of rural areas, where the growth pace is still lacking. IoT is making any objects internally connected in the recent decade and it has been considered as the next technological revolution. Smart health monitoring mechanism, smart parking, smart home, smart city, smart climate, industrial sites, and agricultural fields are some of the applications of IoT. The most tremendous use of IoT is in healthcare management which provides health and environment condition tracking facilities. IoT is nothing but linking computers to the internet utilizing sensors and networks. These connected components can be used on devices for health monitoring. The used sensors then forward the information to distant locations like M2M, which are machinery for computers, machines for people, handheld devices, or smartphones. It is a simple, energy-efficient, much smarter, scalable, and interoperable way of tracking and optimizing care to any health problem. Nowadays, modern systems are providing a flexible interface, assistant devices, and mental health management to lead a smart life for the human being.

1.2. Ideal Health Measures:

Heart rate and body temperature are the two most significant indicators for human health. Heart rate is the per-minute number of heartbeats, commonly known as the pulse rate. To measure the pulse rate, an increase in the blood flow volume can be used by calculating the pulses. Normal heart rate ranges between 60 and 100 beats per minute for healthy people. The typical restful heart for adult males is roughly 70 bpm and for adult females 75 bpm. Female with 12 years of age and above, typically have higher rates of heart in contrast with males. The temperature of human body is simply the heat of body and the sum of heat radiated by the body is scientifically determined. The average person's body temperature relies on different factors such as ambient temperature, the person's gender, and his eating habits. In healthy adults, it is likely to range between 97.8 °F (36.5 °C) and 99 °F (37.2 °C). Different factors such as flu, low-temperature hypothermia, or any other illness may lead to a change in body temperature. In almost all illnesses, fever is a typical indicator. Various methods exist to invasively and noninvasively assess the heart rate and body temperature. For the consumer, noninvasive approaches over a while have proven accurate and convenient. It is suggested that a healthcare should provide good room conditions to facilitate the patients

There are several fatal diseases like heart disease, diabetes, breast cancer, liver disorder, etc. in medical sector but the main concern of our developed system is to monitor the fundamental signs of all types of patients and the patient's room environment. This report proposes a customized healthcare system that monitors the pulse and body temperature of patient. Transmits the data through Wi-Fi that enables the medical staffs to get data from the server. The developed system also provides a solution for the problem of maintaining a single database of patients in hospitals using a web server.

After the search, survey and surfing of so many articles on the internet (mentioned in chapter 2) we have tried to enhance the health monitoring system. We have tried to embed many health sensors together which and detect general health issues of people and can reach out to a professional online. “Literature Review” section describes the recent development of health monitoring system in IoT environment.

Chapter three describes the latest technology used in this project. It elaborates Internet of things, cloud, Arduino development environment, which was used to create the codes and verify them. Proteus simulating is used to check the schematic and implementation. The Blynk IOT platform is used as an GUI interface to display the data. Virtual serial port emulator application is used to connect the Proteus and Blynk interface using virtual ports. COMPIM resource on proteus is the main resource to interface between Blynk and Proteus. The medical sensors used in that are temperature sensor and pulse sensors, moreover as the data is being displayed on LCD screen before being uploaded to cloud. Proteus allows us to use two types of liquid crystal display.

Proposed algorithm (chapter four) presents the major hardware components which are used to develop the prototype are illustrated in “Components used” section. “flowchart” section demonstrates the design methodology of the proposed system. The implementation details with user prototype are depicted in “Schematic” section. The experimental results are described in “Experimental Results Analysis” section. “Conclusion” section concludes the paper with some future work.

CHAPTER 2

LITERATURE REVIEW AND PROBLEM FORMULATION

2.1. Introduction

The project prototype has been developed after a systematic survey on the reviews of some previous works in the wireless sensor network area and to use Internet of Things in health monitoring and providing healthcare facilities. Following research has been made before going ahead with this project.

2.2. IOT Based Disease Detection

• Many applications are now-a-days available where use of smart phones in health monitoring is experimented. Such system consisted of a wearable device comprising of Temperature and Pulse sensors. The device will send its data to the server through the android application. This data will be available to the doctor using his android application.

• While few authors presented the development of a microcontroller-based system for wireless heartbeat and temperature monitoring using ZigBee. The system is developed for home use by patients that are not in a critical condition but need to be constant or periodically monitored by clinician or family. In any critical condition the SMS is send to the doctor or any family member. So that we can easily save many lives by providing them quick service.

• Some of the authors proposed a remote mobile health monitoring system with mobile phone and web service capabilities. It provides doctors and family members with necessary data through a web interface and enables authorized personnel to monitor the patient's condition and to facilitate remote diagnosis.

2.3. Real Time Data Sharing And Developments

• In some papers, the authors have proposed an initial prototype development for wireless transmission of ECG signals using AD8232 sensor and raspberry pi based on IoT.

• In some papers, a Smartphone based remote health monitoring system using body temperature and heart beat sensors to continuously monitor body parameters of cardiac patients, has been proposed using GPS to track location and wireless communication. The link is established between the patient's Bluetooth enabled Mobile device and sensors via a Bluetooth modem, this helps in continuous monitoring.

• An IoT-based ECG monitoring system was developed consisting of three main parts, i.e., the ECG sensing network, IoT cloud, and GUI. Through a wearable monitoring node with three electrodes, real-time ECG signals can be collected with satisfactory accuracy. The gathered data were transmitted to the IoT cloud using Wi-Fi, which supports high data rates and wide coverage areas.

• A paper developed a portable, low - cost ECG data acquisition system with overall less complex circuitry. Results were presented using MATLAB and Raspberry Pi.

2.4. Need and possibilities of IOT in the Heath-care domain

As per the data collected on the internet, the involvement of Internet of things in the healthcare domain has been ever increasing due to the immense possibilities it has. The graph below is a Projection of IoT in worldwide healthcare market site, by component 2014-2025 (USD billion).

We can derive, from this graph, a lot of possibilities that various platforms using internet of things has to provide us and a lot more that we can contribute for its development.

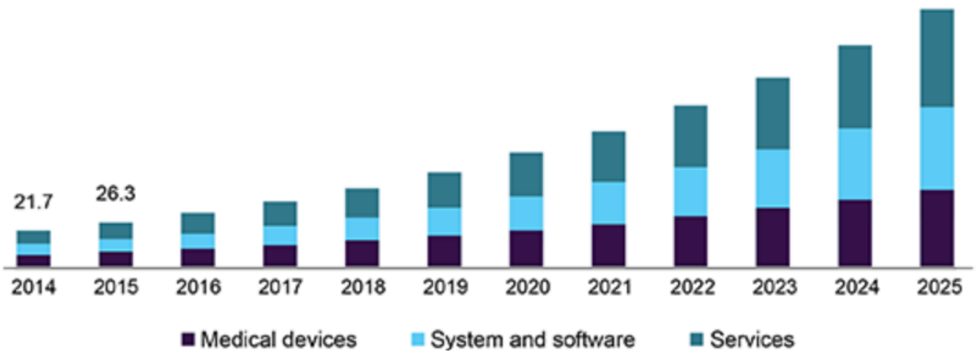


Fig 2.1: Projection of IoT in worldwide healthcare market site, by component 2014-2025 (USD billion)

2.5. Problem Formulation

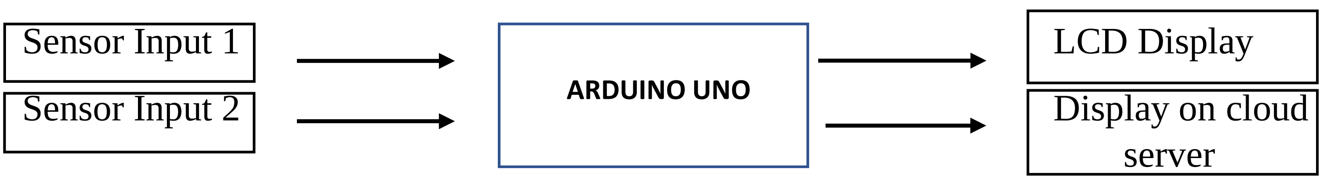
For this project we had analyzed many research papers as discussed in previous section, which were related to bio technology domain. From our research and analysis, we got to know about immense development and enhancement opportunities in the same. With the evolution of IOT it has now become need of the hour. Therefore, we have come up with our proposed simulated project on "**Design of Tele-health care Model.**"

- In this project, a prototype has been intended to be designed and developed for real time health monitoring using IOT platform i.e, Blynk. Software like Arduino, Proteus and Virtual serial port emulator has to be used for the implementation of the simulated model.

- Sensors for recording patient's health parameters like body temperature and heart rate has to be integrated with Arduino board using Proteus simulation and the data is showcased on LCD screen connected to the microcontroller board.

- It is than gathered and transmitted to the IoT cloud which can be visualized using a web-based server or any android-based application.

- Using this system, the physician can use the cloud platform to diagnose patients at remote locations. The patients can also access their medical records via this cloud service.



Chapter- 3

Introduction, Concept and working of various IOT based devices

3.1. Terminologies and Techniques

3.1.1. Internet of Things (IoT)- Internet of Things (IoT) is an ecosystem of connected physical objects that are accessible through the internet. The 'thing' in IoT could be a person with a heart monitor or an automobile with built-in-sensors, i.e, objects that have been assigned an IP address and have the ability to collect and transfer data over a network without manual assistance or intervention. The embedded technology in the objects helps them to interact with internal states or the external environment, which in turn affects the decisions taken. It enables devices/objects to observe, identify and understand a situation or the surroundings without being dependent on human help. It can connect devices embedded in various systems to the internet. When devices/objects can represent themselves digitally, they can be controlled from anywhere. The connectivity then helps us capture more data from more places, ensuring more ways of increasing efficiency and improving safety and IoT security. IoT is basically categorized into 6 major layers. They are:

a) Smart devices and Controllers
b) Connectivity and protocol Communication
c) Cloud Server
d) Data Storage and Accumulation
e) Data analysis and Computing
f) User Application and Report Generation

Fig 3.1: Internet of Things

3.1.2 Cloud- Cloud computing is a general term for anything that involves delivering hosted services over the internet. These services are divided into three main categories: infrastructure as a service (IaaS), platform as a service (PaaS) and software as a service (SaaS). A cloud can be private or public. A public cloud sells services to anyone on the internet. A private cloud is a proprietary network or a data center that supplies hosted services to a limited number of people, with certain access and permissions settings. Cloud service can be used for uploading and storing of data which can be accessed from anywhere using some specific web user end servers or applications. IoT cloud refers to any number of cloud services that power the IoT. These include the underlying infrastructure needed for processing and storing IoT data, whether in real time or not. IoT cloud also includes the services and standards necessary for connecting, managing, and securing different IoT devices and applications. As with other types of cloud services, such as software-as-a-service, organizations consume IoT cloud services as they need them, rather than building a data center or other on-premises infrastructure to deliver those services locally.

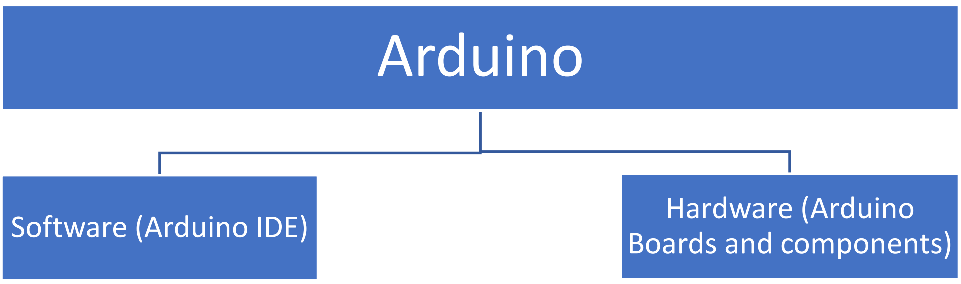
Fig 3.2: IOT Cloud

3.1.3 Tele - Health Monitoring System- Remote health monitoring is growing in popularity as both patients and healthcare professionals want health to be monitored outside of clinical settings. Remote health monitoring, also referred to as remote patient monitoring, is the process of using technology to monitor patients in non-clinical environments, such as in the home. When incorporated in the management of chronic diseases, remote health monitoring has the potential to significantly improve quality of life for patients and so it should come as no surprise that this technology is growing increasingly popular. Remote health monitoring is a diverse field but the associated technologies normally share some similar components. Firstly, a monitoring device requires a sensor which can measure specific physiological data and wirelessly communicate this information to both the patient and healthcare professionals. This system enables medical check-ups and health care to be provided remotely as well as decreases health care delivery costs.

3.2. Arduino Development Environment

3.2.1. Introduction

Arduino is an open-source electronics platform based on easy-to-use hardware and software.



Among wide ranges of Arduino boards available for various uses, Arduino Uno is the most widely used board. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so you use the Arduino programming language (based on Wiring), and the Arduino Software (IDE), based on Processing.

3.2.2. Arduino Ide

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board. A basic interface of Arduino IDE is shown here:

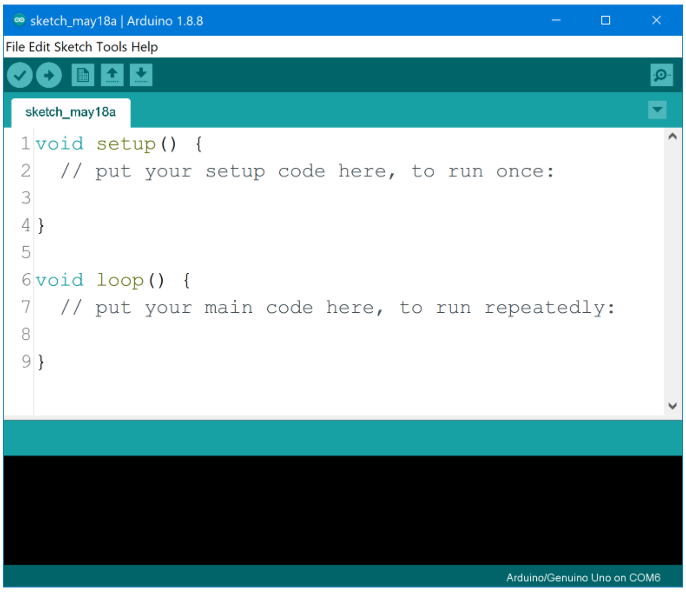


Fig 3.3: Arduino IDE

3.2.3. Arduino Hardware

Arduino uno: The Arduino Uno is an open-source microcontroller board based on the Microchip ATmega328P microcontroller

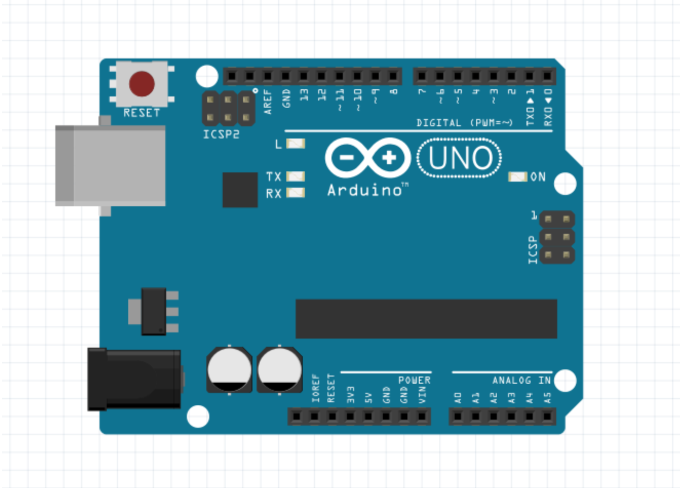


Fig 3.4: Arduino UNO

Technical specifications

- Microcontroller: Microchip ATmega328P
- Operating Voltage: 5 Volts
- Input Voltage: 7 to 20 Volts
- Digital I/O Pins: 14 (of which 6 can provide PWM output)

- PWM Pins: 6 (Pin # 3, 5, 6, 9, 10 and 11)[9]
- UART: 1
- I2C: 1
- SPI: 1
- Analog Input Pins: 6
- DC Current per I/O Pin: 20 mA
- DC Current for 3.3V Pin: 50 mA
- Flash Memory: 32 KB of which 0.5 KB used by bootloader
- SRAM: 2 KB
- EEPROM: 1 KB
- Clock Speed: 16 MHz
- Length: 68.6 mm
- Width: 53.4 mm
- Weight: 25 g
- ICSP Header: Yes
- Power Sources: DC Power Jack & USB Port

General pin functions

- **LED**: There is a built-in LED driven by digital pin 13. When the pin is high value, the LED is on, when the pin is low, it is off.
- **VIN**: The input voltage to the Arduino board when it is using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- **5V**: This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 20V), the USB connector (5V), or the VIN pin of the board (7-20V).
- **3V3**: A 3.3-volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND**: Ground pins.
- **IOREF**: This pin on the Arduino board provides the voltage reference with which the microcontroller operates. A properly configured shield can read the IOREF pin voltage and select the appropriate power source, or enable voltage translators on the outputs to work with the 5V or 3.3V.
- **Reset**: Typically used to add a reset button to shields that block the one on the board.

3.3. PROTEUS

Fig 3.5 : Proteus logo

3.3.1. Introduction to Proteus

Proteus is a widely used application favored by developers and engineers to simulate electrical circuits. It is often used when modeling and designing programmable devices such as microprocessors and microcontrollers.

- **Proteus** is used to simulate, design and drawing of electronic circuits. It was invented by the Labcenter electronic.
- By using proteus you can make two-dimensional circuits designs as well.
- With the use of this engineering software, you can construct and simulate different electrical and electronic circuits on your personal computers or laptops.
- There are numerous benefits to simulate circuits on proteus before make them practically.
- Designing of circuits on the proteus takes less time than practical construction of the circuit.
- The possibility of error is less in software simulation such as loose connection that takes a lot of time to find out connections problems in a practical circuit.
- Circuit simulations provide the main feature that some components of circuits are not practical then you can construct your circuit on proteus.
- There is zero possibility of burning and damaging of any electronic component in proteus.
- The electronic tools that are very expensive can easily get in proteus such as an oscilloscope.
- Using proteus you can find different parents of circuits such as current, a voltage value of any component and resistance at any instant which is very difficult in a practical circuit.

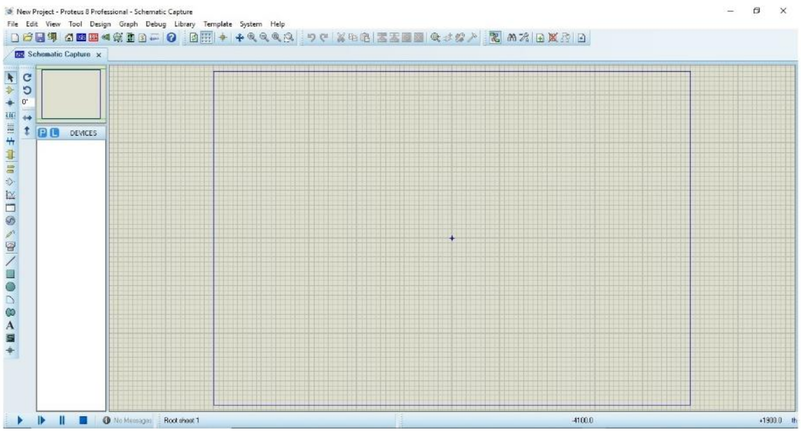


Fig 3.6: Proteus Window

3.3.2. Features of Proteus

- There are 2 main parts of proteus first is used to design and draw different circuits and the second is for designing of PCB layout.
- First is ISIS that used to design and simulate circuits. And second is ARES that used for designing of a printed circuit board.
- It also provides features related to the three-dimensional view of design in PCB.

3.4. Iot Platform (Blynk)

Fig 3.7: Blynk logo

3.4.1. Introduction

In our project, we have used Blynk as our IoT platform. It is specifically designed for the development and implementation of smart IoT devices quick and easily. Blynk is a Platform with IOS and Android apps to control Arduino, Raspberry Pi and similar microcontroller boards over the Internet. It's a digital dashboard where we can build a graphic interface for our project by simply dragging and dropping widgets. Blynk supports the connection types like Wi-Fi, Ethernet, Bluetooth, Cellular, Serial to connect our microcontroller board (hardware) with the Blynk Cloud and Blynk's personal server. The Blynk platform includes the following components:

- Blynk app builder: Allows building apps for our projects using various widgets. It is available for Android and iOS platforms.
- Blynk server: Responsible for all the communications between our mobile device that's running the Blynk app and the hardware. We can use the Blynk Cloud or run our private Blynk server locally. Its open source, could easily handle thousands of devices, and can even be launched on a Raspberry Pi.
- Blynk libraries: Enables communication with the server and processes all the incoming and out coming commands from our Blynk app and the hardware. They are available for all the popular hardware platforms.

All the afore mentioned components communicate with each other to build a fully functional IoT application that can be controlled from anywhere through a preconfigured connectivity type. We can control our hardware from the Blynk app running on our mobile device through the Blynk Cloud or Blynk's personal server. It works the same in the opposite direction by sending rows of processed data from hardware to our Blynk app.

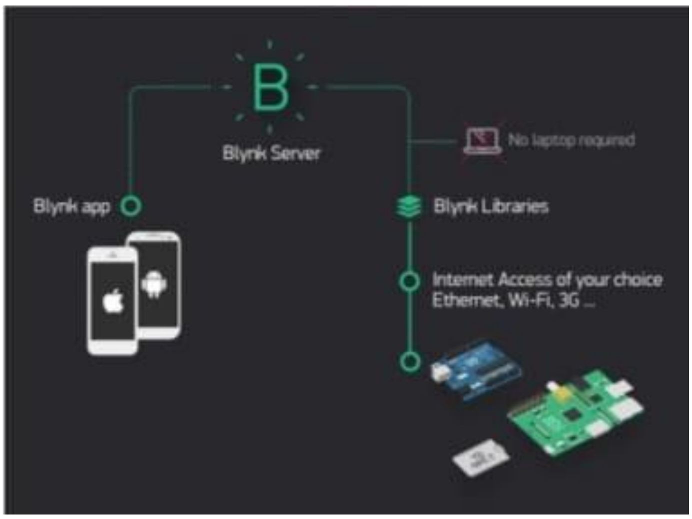


Fig 3.8: Blynk block diagram

3.4.2. UPLOADING DATA TO THE CLOUD:

- Blynk application has very easy user interface and hence can prove to be useful for family members as well as doctors to monitor the data easily and give medication accordingly.
- It uses the TCP/IP protocol which securely communicates with the respective known device and transfers the data in real time.
- Blynk application has different widgets which can be used limitedly and most importantly can be used with different microcontrollers such as Arduino, NodeMCU, Raspberry pi and the like.
- The interfacing of adding different widgets and creating its functionality is really very simple for non-tech people as well. Moreover, for long term application, one can also take the values in the csv files which can be used in the machine learning algorithm to predict and also analyse the data.
- There are two major ways of displaying sensor data in the app:
  PULL: In this case, Blynk app will request the data only when the app is open;
  PUSH: In this case the hardware will be constantly sending data to the Blynk Cloud. And when we open the app, it will be there waiting.

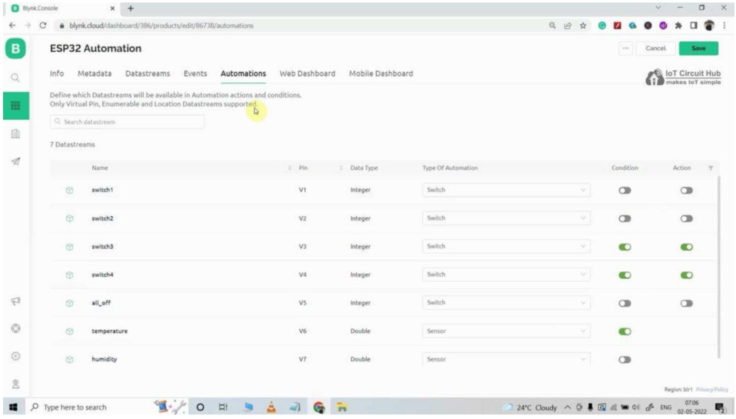


Fig 3.9: Blynk Interface

The process to collect and display the data on the Blynk application can be stated in the following simple steps:

1. Create a New Project after creating an account or log in using Face book.
2. Add Value Display Widget such as gauge and super chart
3. After clicking the widgets go to particular widget Settings
4. Set PIN to Virtual Pin or analog/digital based on your sensor.
5. Set a Frequency of interval or set it to PUSH.
6. Open the coding sketch. Change pin respectively as mentioned in Blynk
7. Send the auth taken to your email by clicking here.
8. Insert your Auth Token in the sketch
9. Flash the code to your microcontroller.
10. Go to the Blynk app - press Play button.
11. The Real time data is updating and can be seen on the application

3.5. MEDICAL SENSORS

3.5.1. Temperature Sensor

- The basic principle of working of the temperature sensors is the voltage across the diode terminals. If the voltage increases, the temperature also rises, followed by a voltage drop between the transistor terminals of base and emitter in a diode.
- The working of the LM35 Temperature Sensor is very simple and easy to understand. We just have to connect 5V and Ground to the sensor and we need to measure the output voltage from the output pin.

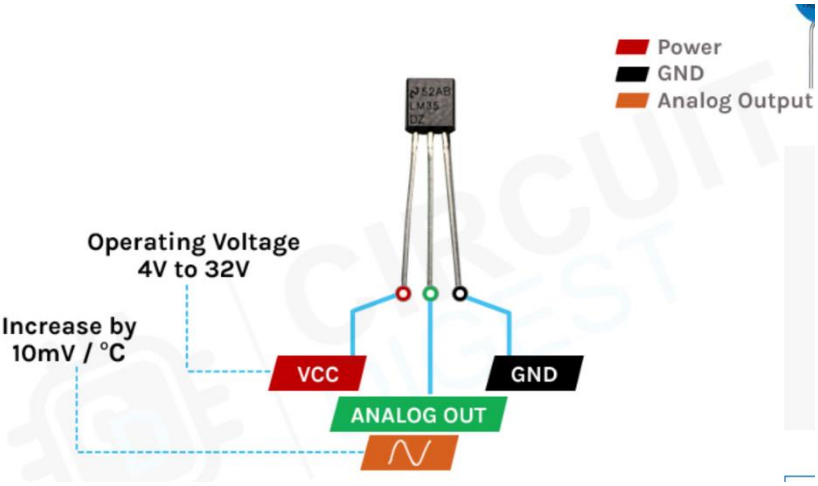


Fig 3.10: LM35 Temperature sensor

3.5.2. Pulse sensor –

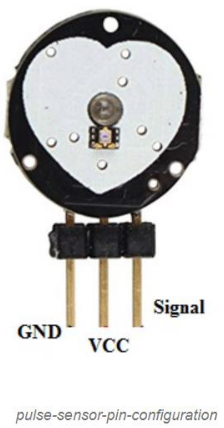


Fig 3.11: Pulse Sensor

- Pulse sensor is designed to give digital output of heat beat when a finger is placed on it.
- When the heart beat detector is working, the beat LED flashes in unison with each heart beat.
- This digital output can be connected to microcontroller directly to measure the Beats Per Minute (BPM) rate.

The heartbeat sensor is based on the principle of photoplethysmography. 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 (avascular region). In the case of applications where the 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 the blood, the signal pulses are equivalent to the heartbeat pulses.

Specifications:

- This is a hear beat detecting and biometric pulse rate sensor
- Its diameter is 0.625
- Its thickness is 0.125
- The operating voltage is ranges +5V otherwise +3.3V
- This is a plug and play type sensor
- The current utilization is 4mA
- Includes the circuits like Amplification & Noise cancellation
- This pulse sensor is not approved by the FDA or medical. So it is used in student-level projects, not for the commercial purpose in health issues applications.

- Pin-1 (GND): Black Colour Wire – It is connected to the GND terminal of the system.
- Pin-2 (VCC): Red Colour Wire – It is connected to the supply voltage (+5V otherwise +3.3V) of the system.
- Pin-3 (Signal): Purple Colour Wire – It is connected to the pulsating o/p signal.

3.6. Liquid Crystal Display:

LCD (Liquid Crystal Display) is a type of flat panel display which uses liquid crystals in its primary form of operation. LEDs have a large and varying set of use cases for consumers and businesses, as they can be commonly found in smartphones, televisions, computer monitors and instrument panels

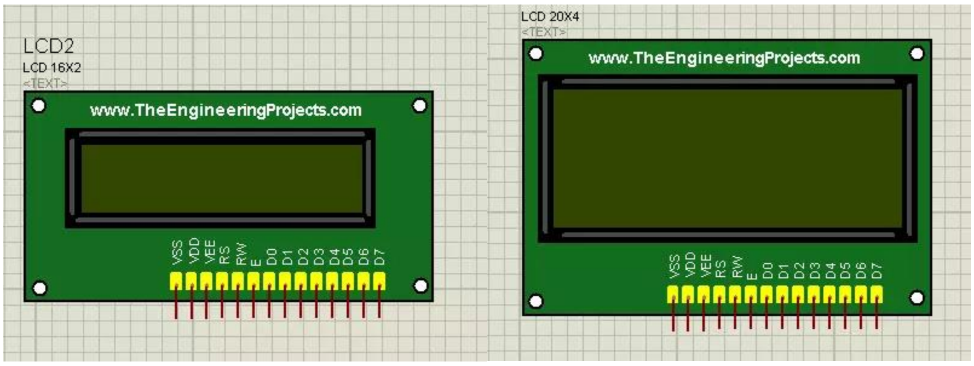


Fig 3.12: 16 x 2 & 20 x 4 Liquid Crystal Display in proteus

LiquidCrystal.h is an inbuilt file provided with the Arduino Software. So adding #include<LiquidCrystal.h> easily helps to use all LCD functions in the code. Following are some functions which automatically gets initialized once this header file is included.

- lcd.begin();
- lcd.print();
- lcd.cursor();
- lcd.home();
- lcd.write();

- lcd.display();
- lcd.blink();
- lcd.autoscroll();

3.7. Virtual Serial Port Emulator (VSPE)

3.7.1. Virtual COM Port

A virtual COM port or a virtual serial port is an ideal solution when a software application expects a connection to a serial device port (COM port or COMM port) but cannot connect, due to a physical lack of available serial ports. Instead, we reconfigure the computer to send serial port data over a local area network or the Internet as if over a true serial port. When the application attempts to send data to a serial COM port, it is actually transmitted through the virtual serial port over a TCP/IP connection. Information sent back to the application also travels over the network where it is received by the virtual serial or virtual COM port, and is then finally passed to the application.

3.7.2. VSPE

VSPE is built to help software engineers and developers to create/debug/test applications that use serial ports. It is able to create various virtual devices to transmit/receive data. Unlike regular serial ports, virtual devices have special capabilities: for example, the same device can be opened more than once by different applications, that can be useful in many cases. With VSPE we are able to share physical serial port data for several applications, expose serial port to local network (via TCP protocol), create virtual serial port device pairs and so on.

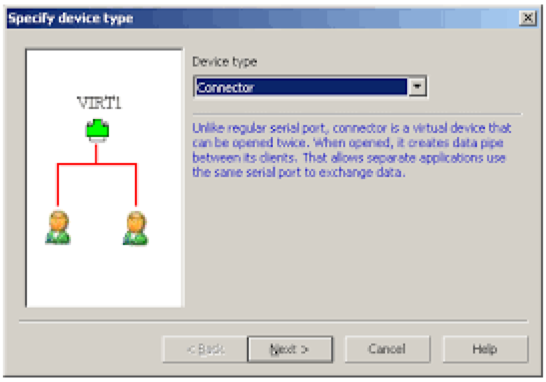


Fig 3.13: Virtual Serial Port Emulator

3.7.3. COMPIM

COMPIM is used to model physical COM interfaces in Proteus. It works by capturing and buffering serial signals which it then presents to the electrical circuit. The computer's serial ports will be used to conduct all serial data originating from the CPU or the UART model.

In case, we use serial ports, we could simply use **Serial Port Terminal** to serve as the host program and run a device simulation on Proteus to evaluate the connection. We will, however, need to employ additional software if we desire to create virtual serial ports in Proteus.

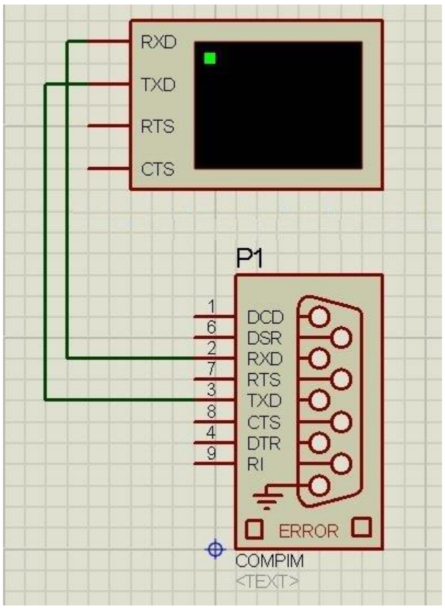


Fig 3.14: COMPIM

CHAPTER 4

PROPOSED ALGORITHM

4.1. Block Diagram

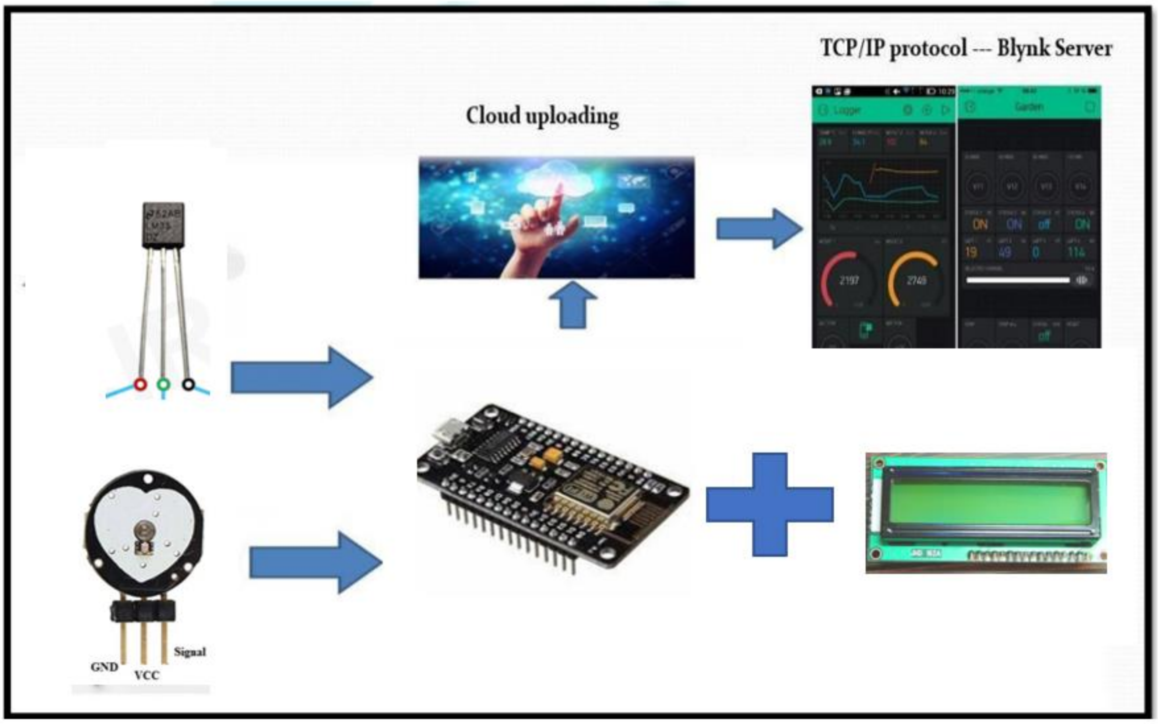


Fig 4.1: Block Diagram

4.2. Components Used

Hardware Requirements: -

- Node MCU – ESP8266 (COMPIN for proteus implementation)
- Temperature sensor – LM35
- Pulse Sensor
- Breadboard
- Jumper wires

Power source Software Requirements: -

- Arduino IDE
- Proteus
- Blynk

4.3. Source Code

SOURCE CODE FOR INTERFACING VARIOUS SENSORS WITH ARDUINO

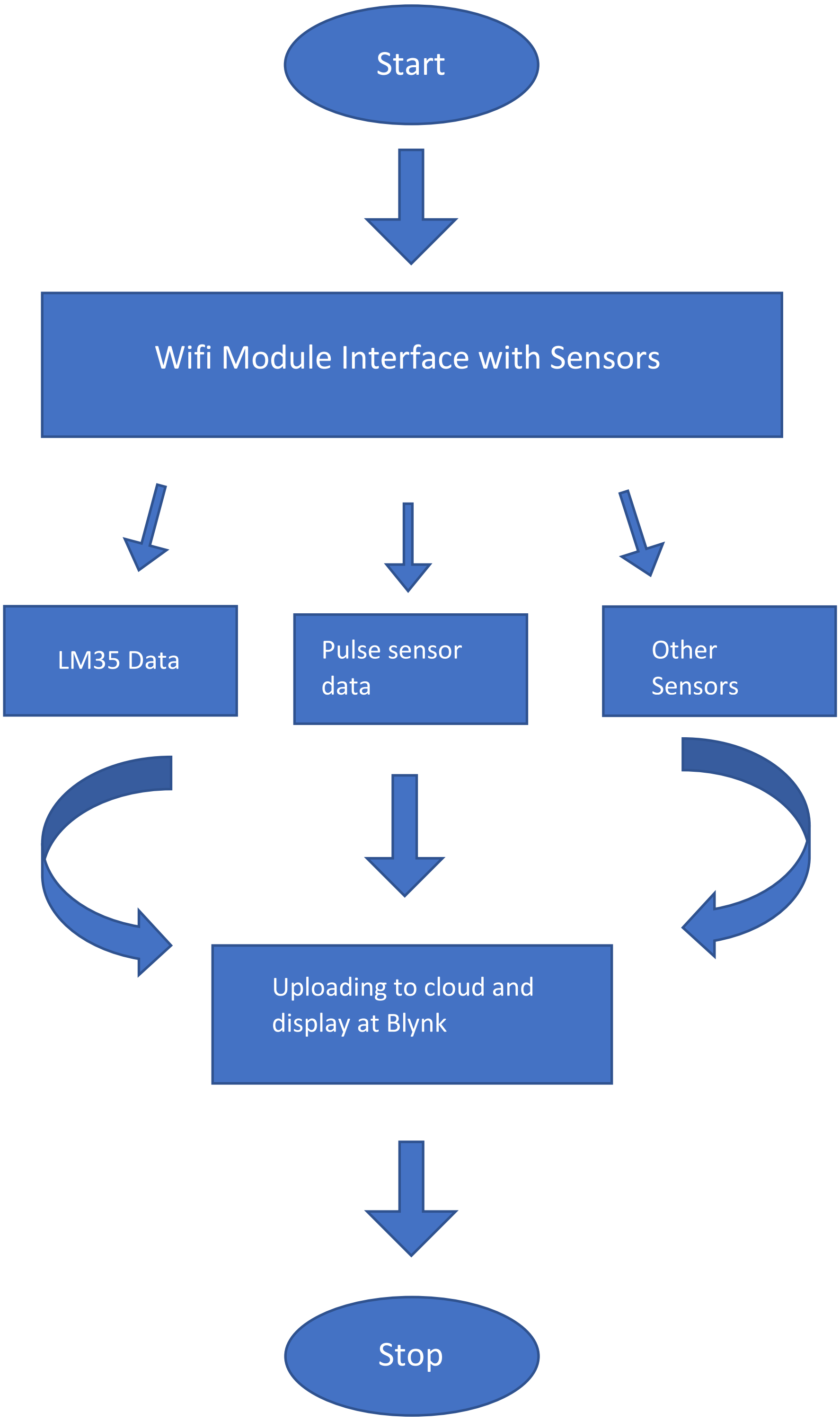
The code has been written and compiled in Arduino Ide.

```
#include <LiquidCrystal.h>
#include <TimerOne.h>
LiquidCrystal lcd(13, 12, 11, 10, 9, 8);

int val;
int tempPin = A0;// temperature Sensor Pin
int HBSensor = 4;// Sensor Pin
int HBCount = 0;
int HBCheck = 0;
int TimeinSec = 0;
int HBperMin = 0;
int HBStart = 2;
int HBStartCheck = 0;

void setup() {
// put your setup code here, to run once:
lcd.begin(20, 4);
pinMode(HBSensor, INPUT);
pinMode(HBStart, INPUT_PULLUP);
Timer1.initialize(800000);
Timer1.attachInterrupt( timerIsr );
lcd.clear();
lcd.setCursor(0,0);
lcd.print("Current HB  : ");
lcd.setCursor(0,1);
lcd.print("Time in Sec : ");
lcd.setCursor(0,2);
lcd.print("HB per Min  : 0.0");
lcd.setCursor(0,3);
```

```
lcd.print("Body Temp  : ");
}
void loop() {
if(digitalRead(HBStart) == LOW){
//lcd.setCursor(0,3);
//lcd.print("HB Counting ..");
HBStartCheck = 1;}
if(HBStartCheck == 1)
{
if((digitalRead(HBSensor) == HIGH) && (HBCheck == 0))
{
HBCount = HBCount + 1;
HBCheck = 1;
lcd.setCursor(14,0);
lcd.print(HBCount);
lcd.print(" ");
}
if((digitalRead(HBSensor) == LOW) && (HBCheck == 1))
{
HBCheck = 0;
}
if(TimeinSec == 10)
{
HBperMin = HBCount * 6;
HBStartCheck = 0;
lcd.setCursor(14,2);
lcd.print(HBperMin);
lcd.print(" ");
//lcd.setCursor(0,3);
//lcd.print("Press Button again.");
HBCount = 0;
TimeinSec = 0;
}
}

val = analogRead(tempPin);
```

```
float mv = (val/1024.0)*5000;
float cel = mv/10;
lcd.setCursor(14,3);
lcd.print(cel);
lcd.print(" ");
delay(100);
}
void timerIsr()
{
if(HBStartCheck == 1)
{
TimeinSec = TimeinSec + 1;
lcd.setCursor(14,1);
lcd.print(TimeinSec);
lcd.print(" "  }}
```

4.4. Flowchart



4.5. Schematic and connections

- Following are the circuit connections as per our proposed model, Temperature sensor, Pulse Sensor and LCD display and COMPIM are connected simultaneously to the Arduino board.

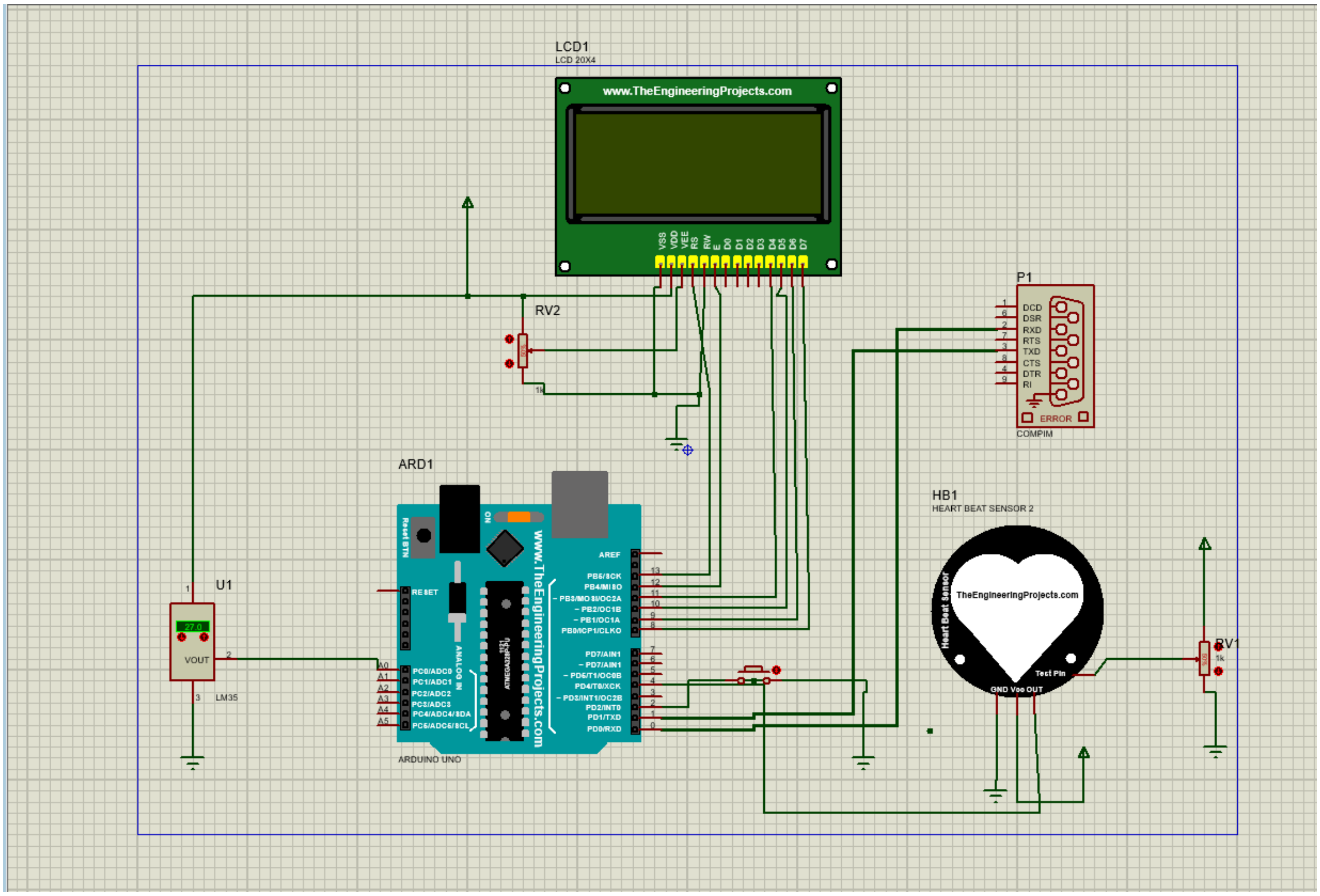


Fig 4.2: Proposed circuit schematic

- After connecting all the devices as per requirements the ".hex" file of source code written using Arduino IDE is fed in the Arduino board as well as the sensors.
- Blynk Dashboard has also been setup to receive the information.
- As soon as the program is RUN, various results are obtained, which are shown in the next chapter.

Chapter 5

Experimental Results

Results display and connections

➢ ARDUIONO—LDC

For the real time result display to the patients, LCD screen has been used and following are the connections of Arduino board with the LCD screen.

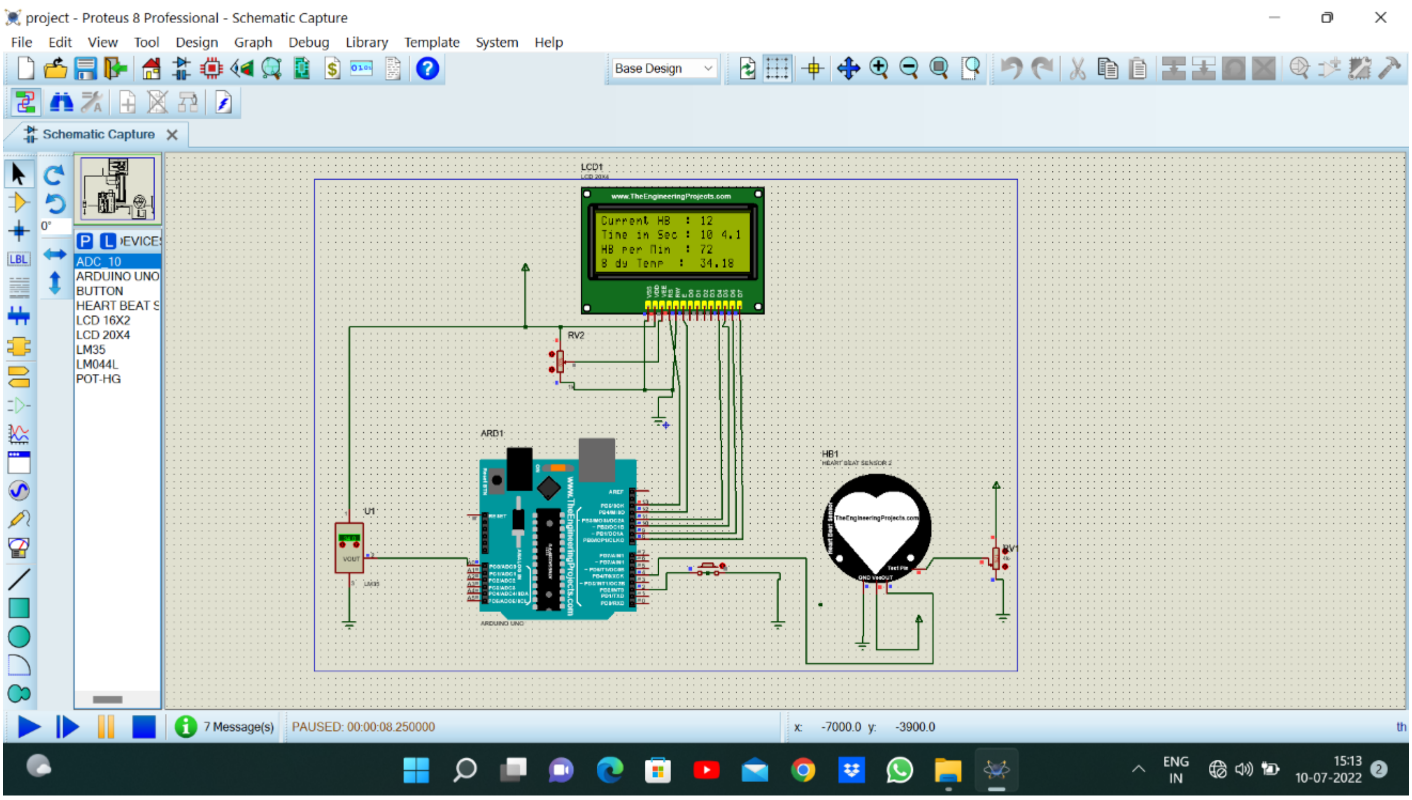


Fig 5.1: LCD display showing results

➢ Pin connections for Arduino and LCD are as follows:

- PIN 13 – RS
- PIN 12 – E
- RW – Gnd
- PIN 11 – D4
- PIN 10 – D5
- PIN 9 – D6
- PIN 8 – D7

➢ ARDUINO – PULSE SENSOR

- Vcc of pulse sensor has been connected with Arduino pin 2, via a switch S1.
- Switch S1 would be used to trigger the heartbeat sensor
- Potentiometer has been used for varying the pulse rate and feeding pseudo input.

➢ ARDUINO – LM35

- LM35 temperature sensor has been connected with Arduino using the analog pin A0 and rest reset and power circuitry has been completed.
- To vary the sensor input, an inbuilt button is provided in the LM35 sensor itself.

➢ ARDUINO – COMPIM

COMPIM has been connected to send the real time data on cloud.

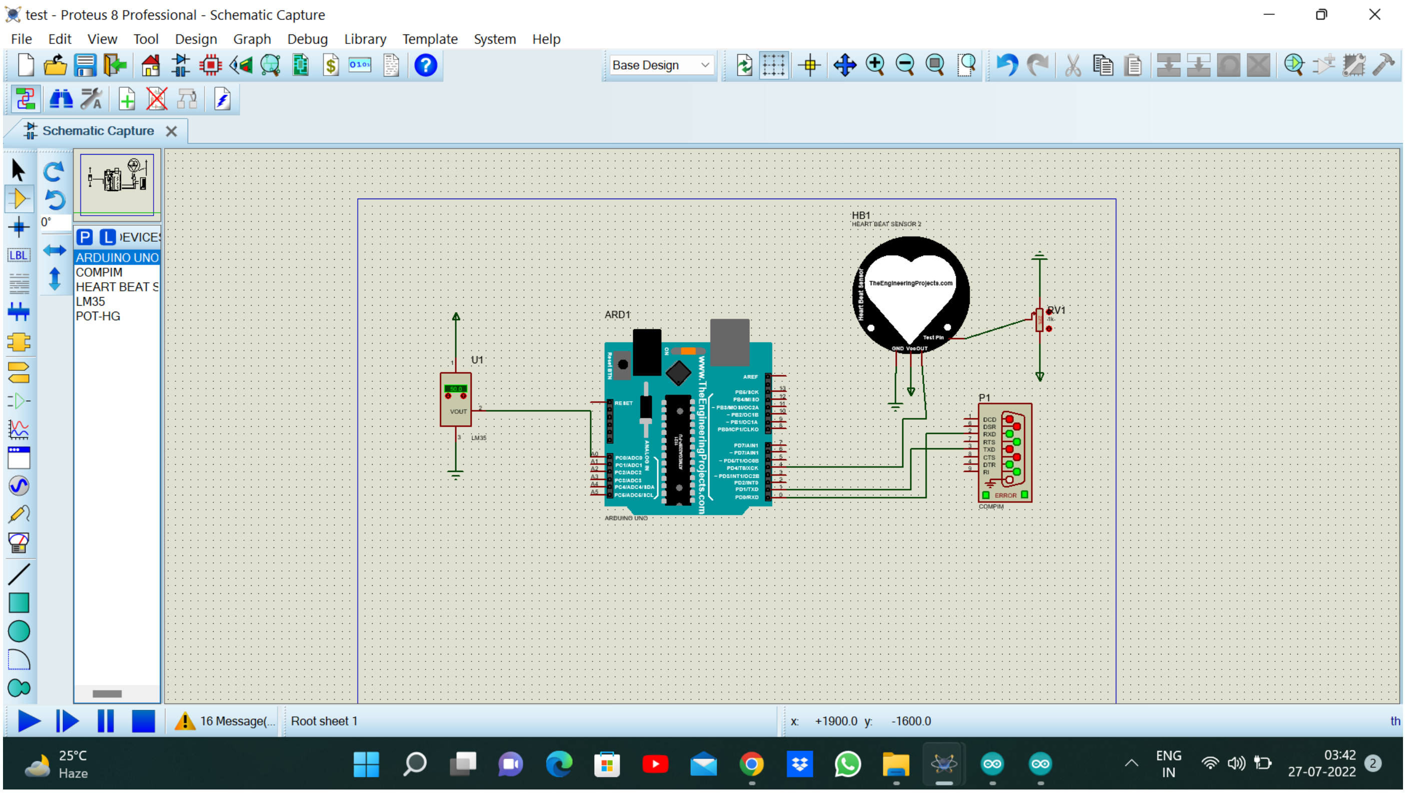


Fig 5.2: Activated COMPIM

- TXD and RXD pins of Arduino and COMPIM are connected.
- Arduino PIN 1 – COMPIM PIN 3
- Arduino PIN 0 – COMPIM PIN 2

➢ Virtual serial port Emulator Setup:

To create the virtual comports Virtual serial port Emulator has been used. Blynk-cloud platform has been connected with virtual port COM1. COMPIM used in our simulation has been connected to virtual port COM2 and both the ports are then paired using virtual serial port emulator.

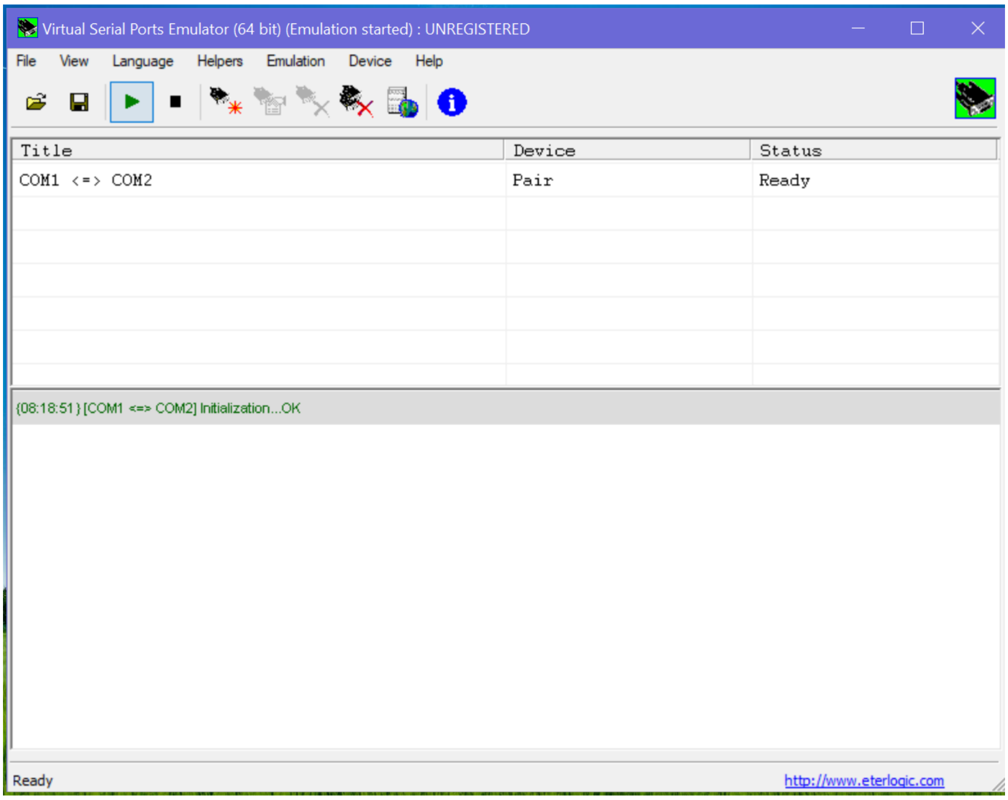


Fig 5.3: Virtual COM1 and COM2 paired

- **Blynk Interface:**

Here is the Blynk web dashboard displaying the received data. Blynk- cloud has been connected to the virtual COM1 via VSPE.

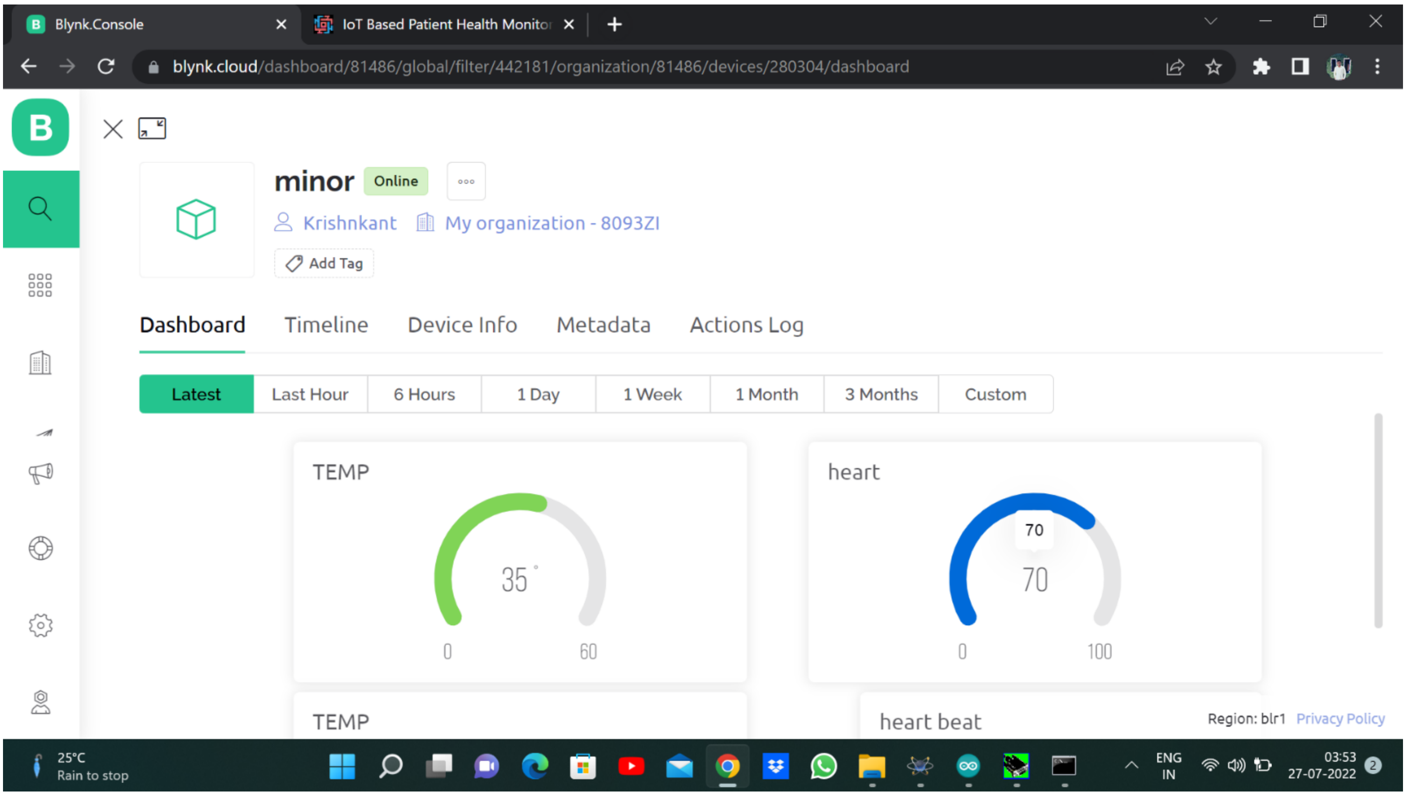


Fig 5.4: Blynk interface showing results 1

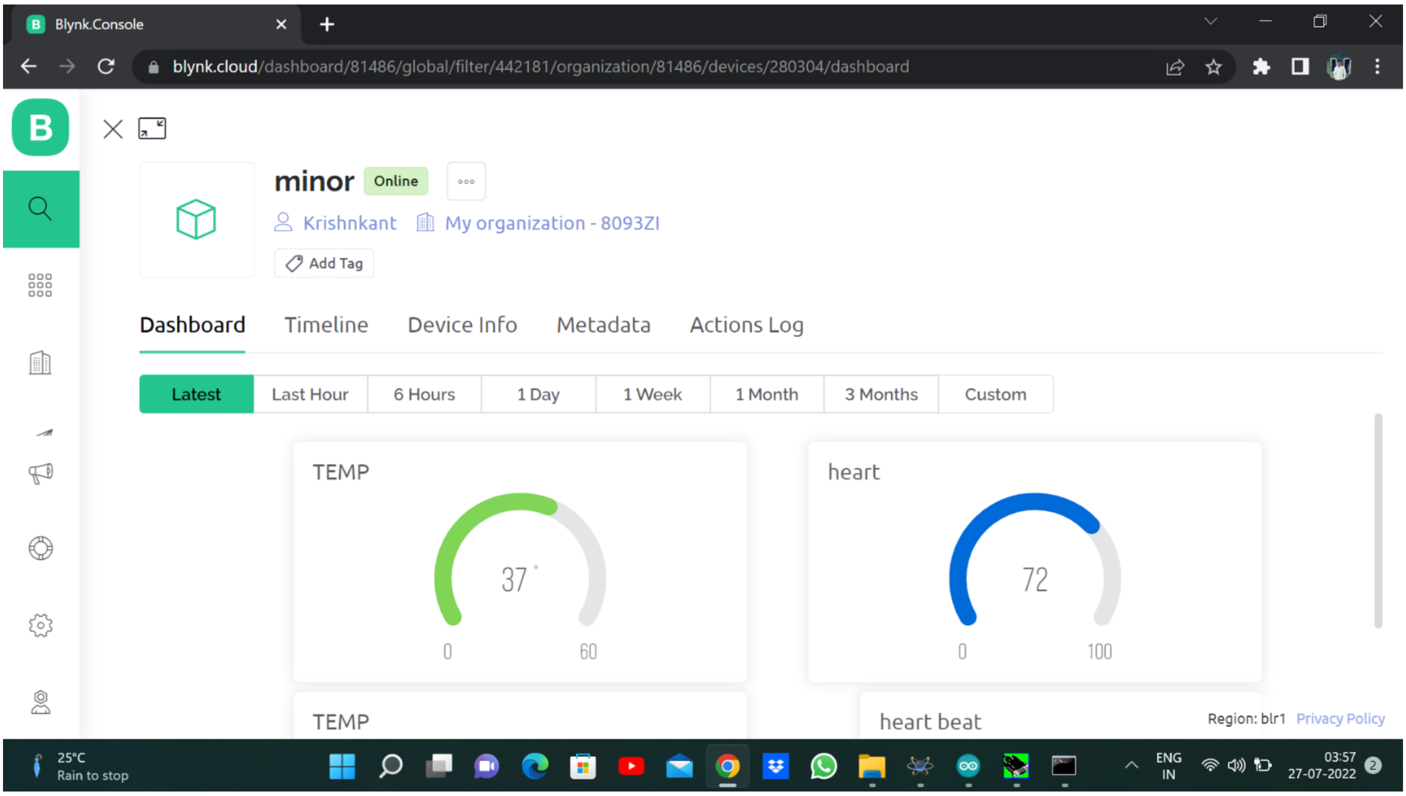


Fig 5.5: Blynk interface showing results 2

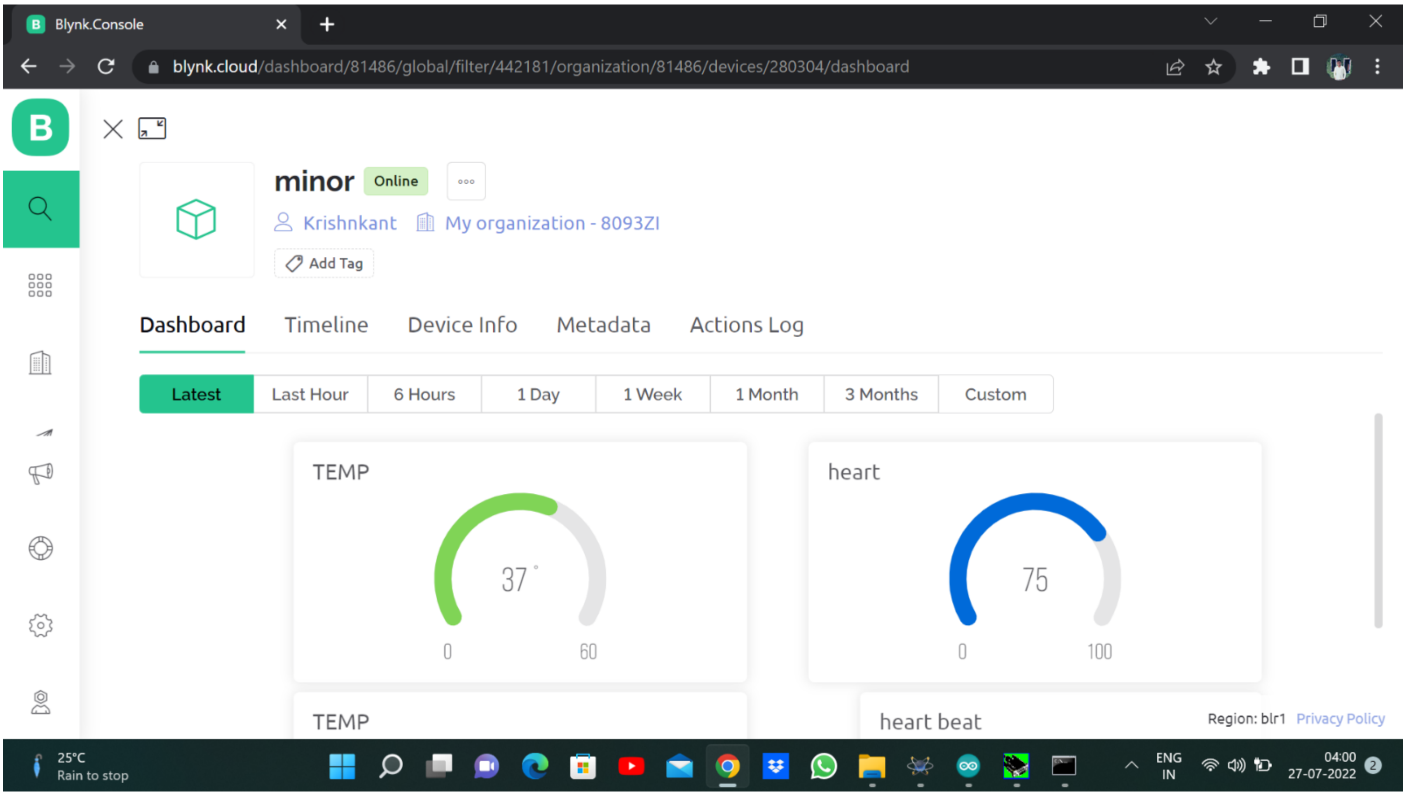


Fig 5.6: Blynk interface showing results 3

➢ **Display on Blynk Mobile Application:**

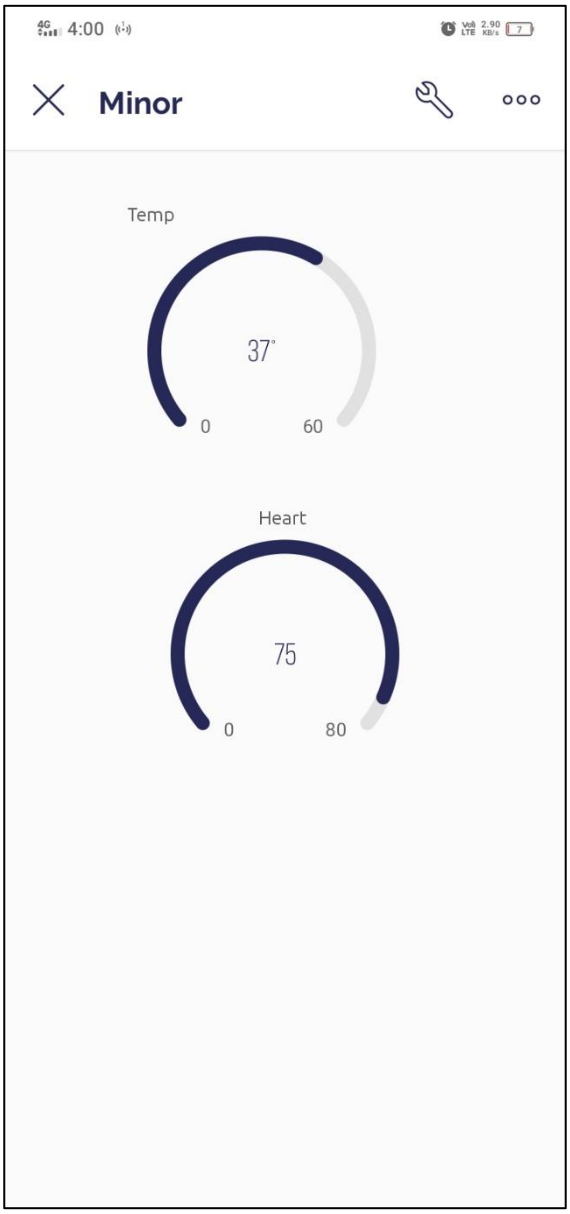


Fig 5.7: Result Display on Blynk Mobile application

OBSERVATIONS:

On varying the Potentiometer resistance, that is connected to the pulse sensor and varying the temperature of LM35 sensor, following results are obtained:

OBSERVATION	**LM35 TEMPERATURE SENSOR READINGS** (CEL)	**PULSE SENSOR READINGS** (Heart beat per min)
1.	**35**	**70**
2.	**37**	**72**
3.	**37**	**75**

Table 5.1 : Observations

Result Analysis

The simulation model is ready for the hardware implementation purpose. The results of actual hardware model could be manually calculated and compared with the obtained data. However, following errors during the implementation are expected and need to be enhanced in the upcoming projects.

Firstly, the deviation in the readings could occur due to motion artifact. Also, sometimes due to errors in the precision of sensors used. A snapshot of the continuously collected data in web server (Blynk) is illustrated in above results. Any medical staff can easily monitor the patient's condition through any type of device which has internet access. The patient data could be secured with a password and for accessing data, one needs to pass through this password protected system, i.e., only the authentic staff can monitor the system. By analyzing data, a doctor can easily decide on from a remote location. As discussed above, the ideal health conditions could be compared with the actual health conditions by the doctor. So that the far located patient could be helped in the best possible manner.

Chapter 6
Conclusion and Future Scope

In this project, we designed and developed an Tele-Health Monitoring System using IoT. This prototype was designed keeping in mind the problems faced by most of the people who deal with tight daily schedules and tend to postpone or miss their required health check-ups and also for rural people, who have very less facilities of heath care. This system can also provide assistance to the elderly people who find it inconvenient and difficult to visit clinics or diagnostic centers for their regular check-ups. With the help of the remote monitoring system provided by our prototype, health monitoring can be done by the doctors or health care providers at any time from any location in a cost-effective manner. As this system size is quite small so it can be carried at various locations with ease. Through the use of small circuitry, the heartbeat and temperature signals of a patient can be viewed rather than using large machines for it. Doctors can see data remotely and analyze the health parameters of patients. This is the most important advantage of this system. The persons living in remote locations who have no access to a doctor can be helped through a greater extent through this system, as this system sends all the values and signals on the website and the doctors which are far away can get an accurate idea of heart condition of a person.

Furthermore, this system can be used in ambulance which saves a lot of time and can save a life of a person because every second counts. It is evident that the implementation of such a system will help in early detection of abnormal conditions of cardiovascular diseases and prevention of its serious consequences.

Also, advanced features like smart health card, separate reports of individual patient, automatic sanitizer circuit can also be added for the further enhancement and future developments of the project.

Chapter 7

References

[1] Nancy Fernando E, “A Survey on Android Based Patient Monitoring System” GRD Journals | Global Research and Development Journal for Engineering | International Conference on Innovations in Engineering and Technology (ICIET) - 2016 | July 2016eISSN: 2455-5703

[2] Manisha Shelar, Jaykaran Singh, Mukesh Tiwari, “Wireless Patient Health Monitoring System”, International Journal of Computer Applications (0975 – 8887) Volume 62– No.6, January 2013

[3] Yunzhou Zhang1, Huiyu Liu, Xiaolin Su, Pei Jiang and Dongfei Wei1, “Remote Mobile Health Monitoring System Based on Smart Phone and Browser/Server Structure” 2013 International Conference on Electronic Engineering and Computer Science

[4] Uttara Gogate, Jagdish Bakal, “Smart Healthcare Monitoring System Based on. Wireless Sensor Networks.” Computing Analytics and security trends, 2016.

[5] Mehak Chhabra, Manik Kalsi, “Real Time ECG monitoring system based on Internet of Things (IoT)”. International Journal of Scientific and Research Publications, Volume7, Issue 8, August 2017

[6] Uttara Gogate, Manali Marathe, Jay Mourya, Nipunya Mohan, “Android Based Health Monitoring System for Cardiac Patients”. International Research Journal of Engineering and Technology (IRJET), Volume: 04 Issue: 04 | Apr -2017.

[7] Uttam U. Deshpande, Milan A. Kulkarni , “IoT based Real Time ECG Monitoring System using Cypress WICED”. International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, Vol. 6, Issue 2, February 2017.

[8] Zhe Yang, Qihao Zhou, Lei Lei, Kan Zheng, Wei Xiang, “An IoT-cloud Based Wearable ECG Monitoring System for Smart Healthcare”, February 2018.

[9] Ms.Gauravi.A.Yadav, Prof. Shailaja.S.Patil, “RASPBERRY PI BASED ECG DATA ACQUISITION SYSTEM”. International Journal of Advance Research in Science and Engineering, April 2017.

[10] Shivleela Patil, Dr. Sanjay Pardeshi, “Health Monitoring system using IoT.” International Research Journal of Engineering and Technology (IRJET) , Volume: 05 Issue: 04 | Apr-2018.

[11] Muriel Boulakia, Serge Cazeau, Miguel Angel Fernández, Jean-Frédéric Gerbeau, Nejib Zemzemi. Mathematical Modeling of Electrocardiograms: A Numerical Study. [Research Report] RR-6977, 2009, pp.42. ffinria-00400490v1ff