IMPLEMENTATING TREE TOPOLOGY IN ESP32 AND REDUCING LOAD IN ROUTES OF A NETWORK

ABSTRACT

The unit which administers and manages the requisition and distribution of the blood is named as a blood bank. The main objectives of the blood banks are providing blood to the patients with minimal blood transfusion error. The blood is very important medical supplies so it should be managed well. As the blood bank management consists of several manual steps, therefore it will become difficult for the blood banks to provide a high level of accuracy, reliability, automation in blood storage and transfusion process. The system proposed is divided into three segments, the first segment consists of Temperature sensor, IR sensor nodes which is installed in rack of blood bank, and the GSM Module for sending request of blood to the donors and blood banks all these are interfaced with Arduino Mega. Second segment consists of wi-fi module for data transfer to the server and third segment is displaying the status of available blood stock. All the real time status relates to the available blood stock of the blood bank is displayed on web page, so that the blood seeker can get the blood from their nearest blood bank. Key Words: Blood bank, blood transfusion, Temperature Sensor, IR sensor, GSM Module, Arduino, Wi-fi Module, Web page.

TABLE OF CONTENTS

SERIAL NO.		TITLE	PAGE NO.
		ABSTRACT ACKNOWLEDGEMENT	
1		INTRODUCTION	
	1.1	OBJECTIVES AND GOALS	
	1.2	APPLICATIONS	
	1.3	FEATURES	
	1.4	LITERARY REVIEW	
2		DESIGN	
	2.1	BLOCK DIAGRAM	
	2.2	HARDWARE ANALYSIS	
	2.3	(SNAPSHOTS-PROJECT, TEAM, RESULTS)	
3	3.1	SOFTWARE CODING AND ANALYSIS (SNAPSHOTS OF CODING AND RESULTS)	
4		CONCLUSION AND FUTURE WORK	
	4.1	RESULT, CONCLUSION AND INFERENCE	
	4.2	FUTURE WORK COST	
5	REFERENCES		

\

CHAPTER 1

INTRODUCTION

1.1 OBJECTIVES

- Design the PAINLESS MESH for IoT for managing the temperature of the blood bag that is received from the blood donation events needs a proper and systematic management that is done by the blood banks.
- Detect the change in the temperature in the blood bag and manage accordingly.
- Design the automatic temperature modifying system which has a varying intensity depending upon the number of persons and the intensity of moisture inside the room.
- Design the automatic IoT system depending on the varying temperature of the surrounding.
- Design RF module communication between Nodes and the designed system

1.2 APPLICATION

The painless mesh can be deployed in places where the data has to be transmitted to a large area/long distance. Few typical examples are

- Airport
- Shipyard
- Space station

and many more areas where large number of sensors are used and the collected data has to be sent over a long-distance communication this model can deployed.

1.3 FEATURES

- Sensors to collect the data of various things
- Wi-Fi communication module to transmit and receive data
- Use of devices such as water pump and bulb
- Microcontroller used is ESP32
- The efficient coding in the software tool called as Arduino IDE
-

1.4 LITERATURE REVIEW

International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 05 Issue: 05 | May-2018 www.irjet.net p-ISSN: 2395-0072

Smart Blood Bank System Based on IoT

Radha R. Mahalle[1], S. S. Thorat[2]

[1]*Electronics and Telecommunication Department, GCOE, Amravati, India*
[2]*Assistant Professor, Electronics and Telecommunication Department, GCOE, Amravati, India*

Abstract - *The unit which administers and manages the requisition and distribution of the blood is named as a blood bank. The main objectives of the blood banks are providing blood to the patients with minimal blood transfusion error. The blood is very important medical supplies so it should be managed well. As the blood bank management consists of a number of manual steps, therefore it will become difficult for the blood banks to provide a high level of accuracy, reliability, automation in blood storage and transfusion process. The system proposed is divided into three segments, the first segment consists Temperature sensor, IR sensor nodes which is installed in rack of blood bank, and the GSM Module for sending request of blood to the donors and blood banks all these are interfaced with Arduino Mega. Second segment consists of wi-fi module for data transfer to the server and third segment is displaying the status of available blood stock. All the real time status relates to the available blood stock of the blood bank is are displayed on web page, so that the blood seeker can get the blood from their nearest blood bank.*

Key Words: Blood bank, blood transfusion, Temperature Sensor, IR sensor, GSM Module, Arduino, Wi-fi Module, Web page.

1. INTRODUCTION

The process of managing the blood bag that is received from the blood donation events needs a proper and systematic management that is done by the blood banks. As the blood is related to someone's life so the bags of must be handled with care and treated thoroughly. The blood consists of different blood constituents and every constituent of blood is having a specific function. Statistics say that in our India every two seconds someone needs a blood transfusion. The Blood transfusions are used for trauma victims - due to accidents and burns - heart surgery, organ transplants, women with complications during childbirth, newborns and premature babies, and patients receiving treatment for leukaemia, cancer or other diseases, such as sickle cell disease and thalassemia [6].The different blood constituent is applied to the patient for different type's disease given in table1.

India today faces a blood shortage of 10% relative to its blood requirements. It means that we require covering a shortfall of over 12 lakhs of blood units. Given that the eligible donor population of our India is more than 512 million, this deficit is surprising. Every day more than 1200 road crashes occurring in our India, 60 million trauma induced surgeries are performed in the country every year. Near about 230 million major operations, 331 million cancer-related procedures like chemotherapy and 10 million pregnancy complications all these require the blood transfusion [9]. Besides all this, the patients which are for sickle cell anaemia, thalassemia and haemophilia require large quantities of blood daily. Another reason for the shortage of the blood supply is the ban on payment to blood donors that was enacted in 1995. Previously near about one-third of the blood supply was from paid donors. But with the ban, both private and government hospitals have faced a larger shortage in the blood supply.

Table -1: Different Blood Constituents with their uses for different diseases and there functions

Blood Constituents	Functions	Diseases
Plasma	Medium in which the blood cells are transported around the body	Burn patients, Shock, Bleeding disorders
Red blood cells	Carries oxygen	Surgery, Any blood loss , Blood disorders such as sickle cell
White blood cells	Part of the immune System	Infectious disease and foreign invaders
Platelets	To facilitate blood clotting	Cancer treatments, Organ transplants , lower platelet counts, suffering from leukemia

The current system that is using by the blood bank is manual system. With the manual system, there are problems in managing the records related with blood stock. There is no centralized database of volunteer donors. So, it becomes really tedious for a person to search blood in case of emergency. Without an automated management system, there are also problems in keeping track of the actual amount of each and every blood type in the blood bank. In addition, there is also no alert available when the blood quantity is below its par level or it become zero. Healthcare as a sector is a definite beneficiary of the increasing adoption of technologies such as the Internet of Things. One of the areas where IoT can be used is blood bank management. In our country thousands of litters of blood get wasted every day due to less than adequate management practices,

CHAPTER 2

IoT BASED PATIENT MONITORING SYSTEM - DESIGN

2.1 BLOCK DIAGRAM

The four main features of the basic block diagram (given below) are

- The Parent Node, serially connected to the ESP32
- The child nodes at different places on the blood bag carrier/container
- The sensors
- Wi-Fi Modules

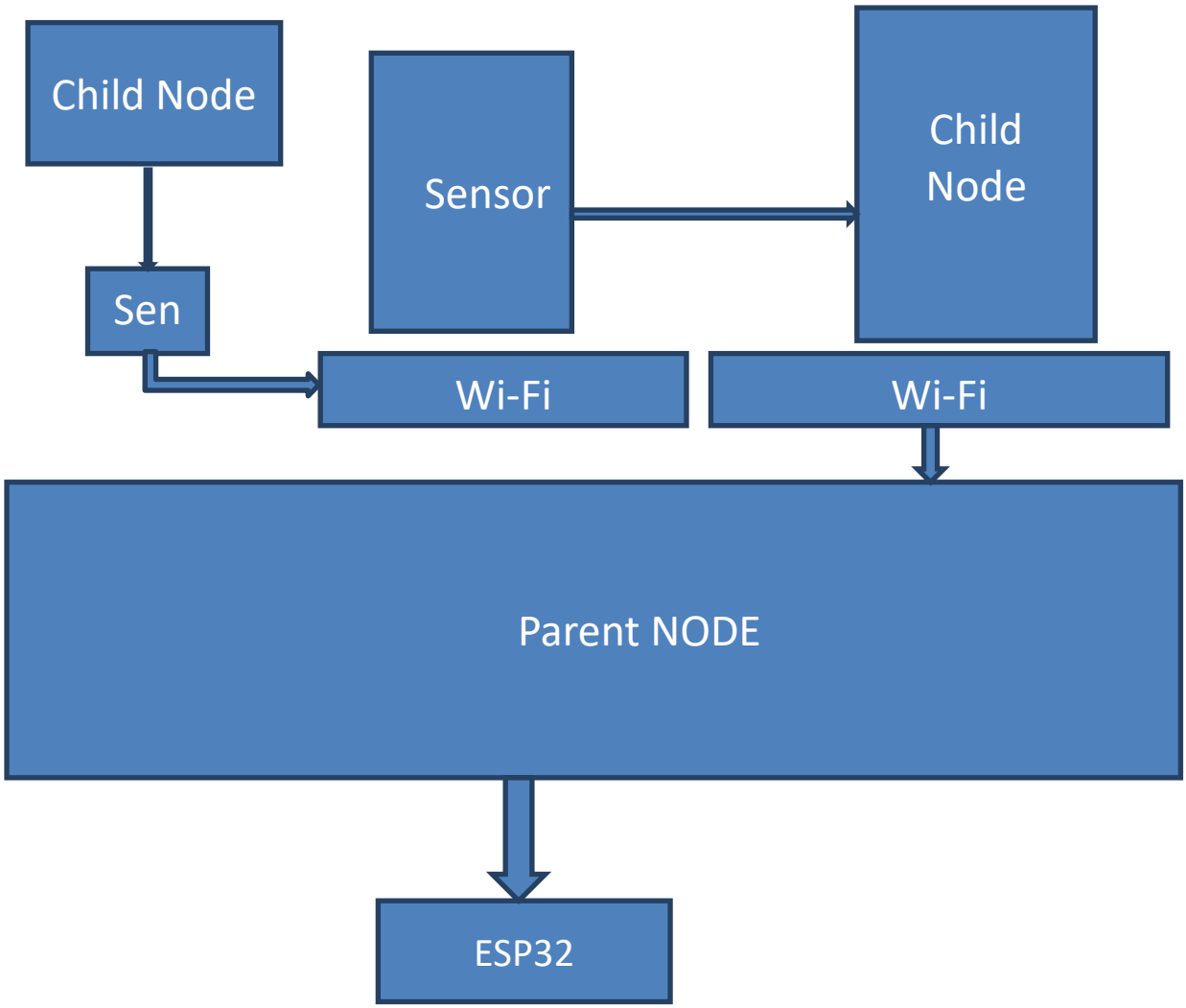


Figure 1. Block diagram with all the components used

2.2 HARDWARE ANALYSIS

The PCB consists of the power supply, Wi-Fi module, Microcontroller ESP32 Chip and the Voltage Divider circuit placed upon the PCB board. The power supply powers all the other units with the appropriate voltages. The Microcontroller will compare the data gathered from the sensors with the data from the computer and sends out the feedback signals.

2.2.1 POWER SUPPLY

The power supply will be 10,000 mAh power bank and the voltage eliminator to supply power to different devices. A low-dropout regulator will step down the voltage to 3.3V to power the microcontroller, sensors, L EDs, motors. The power supply will connect to all the other components through the PCB board and wires.

2.2.2 REAL-TIME HUMIDITY AND TEMPERATURE DETECTION SYSTEM

An DHT-11 sensor is an electronic instrument that is used to sense certain characteristics of its surroundings by either It is also capable of measuring heat of an object and detecting temperature.

A typical system for detecting Humidity sing DHT-11 sensors includes the Arduino or any compatible microcontroller with digital pins.

The output from the sensors are framed and packeted for transmission. The packets are then sent over to the master node after checking the connection to the mater node. If the master node is not in contact, the data is stored locally and flushed to the master node later, when available.

Finally, the infrared detector completes the system for detecting infrared radiation. The output from the detector is usually very small, and hence pre-amplifiers coupled with circuitry are added to further process the received signals.

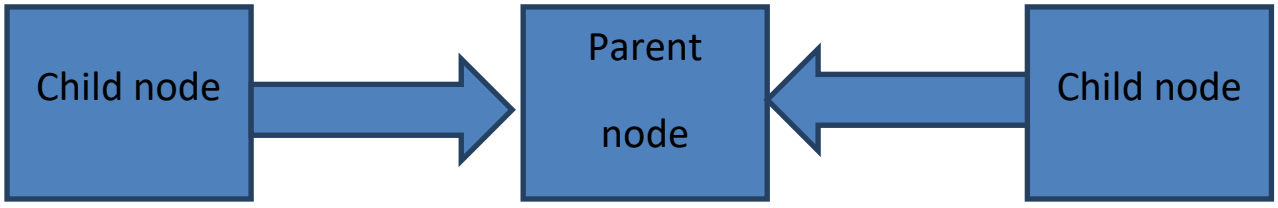


Figure 2. Sensor interfaced with Controller

Figure 2 explains that the temp sensors are attached to the two ends of a long agricultural field. It is used to detect the humidity and temperature of the atmosphere. The signal from the Humidity sensor is then transmitted to the microcontroller through Wi-Fi module which is received at the receiver end.

2.2.3 Wi-Fi MODULE

The Wi-Fi Frequency Modules used use the personal open source bandwidth of 2.4GHz. Wi-Fi transmission is stronger and more reliable than Radio Frequency (RF) transmission due to following reasons:

- Wi-Fi signals can travel longer distances than Radio Frequency.
- Infrared signals will get interfered by other IR sources but signals on one frequency band in Wi-Fi will not interfered by other frequency Wi-Fi signals.

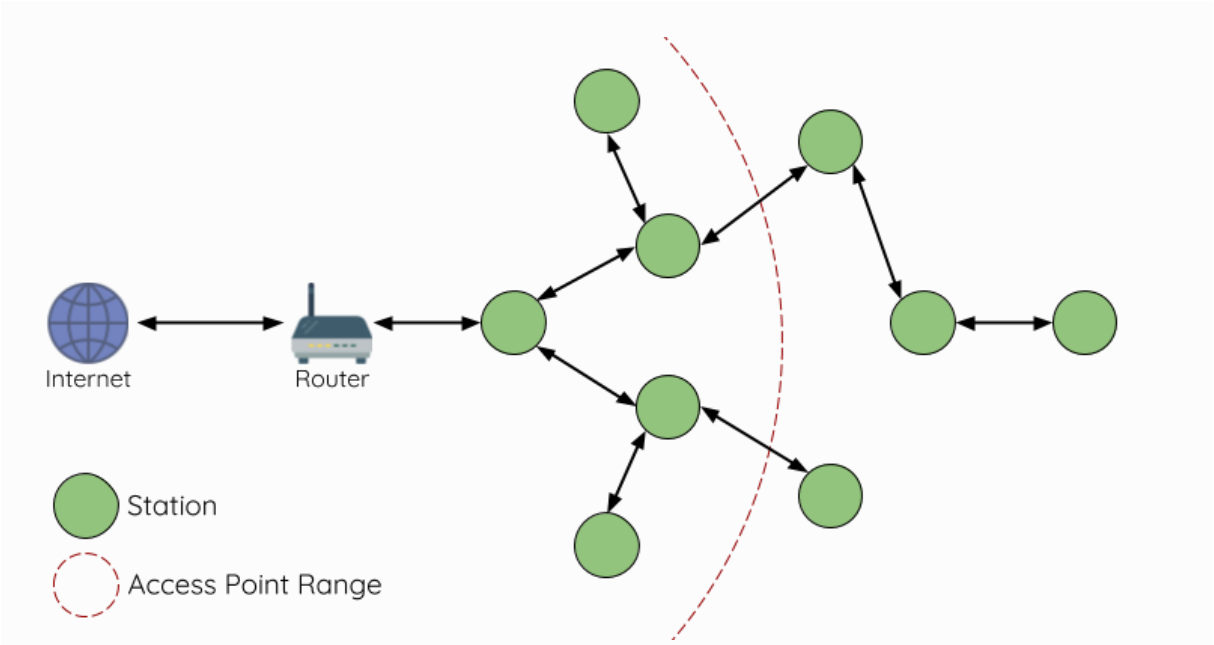
2.3 HARDWARE IMPLEMENTATION

Figure 3. HARDWARE CIRCUIT

CHAPTER 3

SOFTWARE CODING AND ANALYSIS

3.1 SOFTWARE IMPLEMENTATION



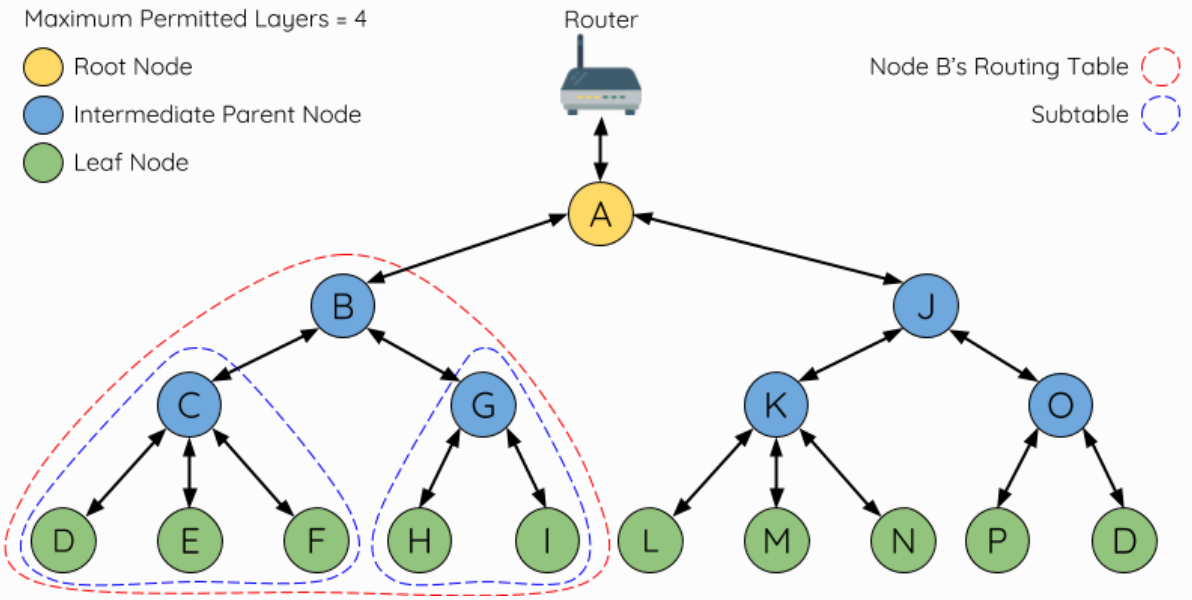
ESP-MESH Network Architecture

Figure 4. ESP-MESH Network Architecture

ESP-MESH differs from traditional infrastructure Wi-Fi networks in that nodes are not required to connect to a central node. Instead, nodes are permitted to connect with neighbouring nodes. Nodes are mutually responsible for relaying each other's transmission. This allows an ESP-MESH network to have much greater coverage area as nodes can still achieve interconnectivity without needing to be in range of the central node. Likewise, ESP-MESH is also less susceptible to overloading as the number of nodes permitted on the network is no longer limited by a single central node.

3.1.1. Routing Tables

Each node within an ESP-MESH network will maintain its individual routing table used to correctly route ESP-MESH packets (see ESP-MESH Packet) to the correct destination node. The routing table of a particular node will consist of the MAC addresses of all nodes within the particular node's **subnetwork** (including the MAC address of the particular node itself). Each routing table is internally partitioned into multiple sub tables with each sub table corresponding to the subnetwork of each child node.

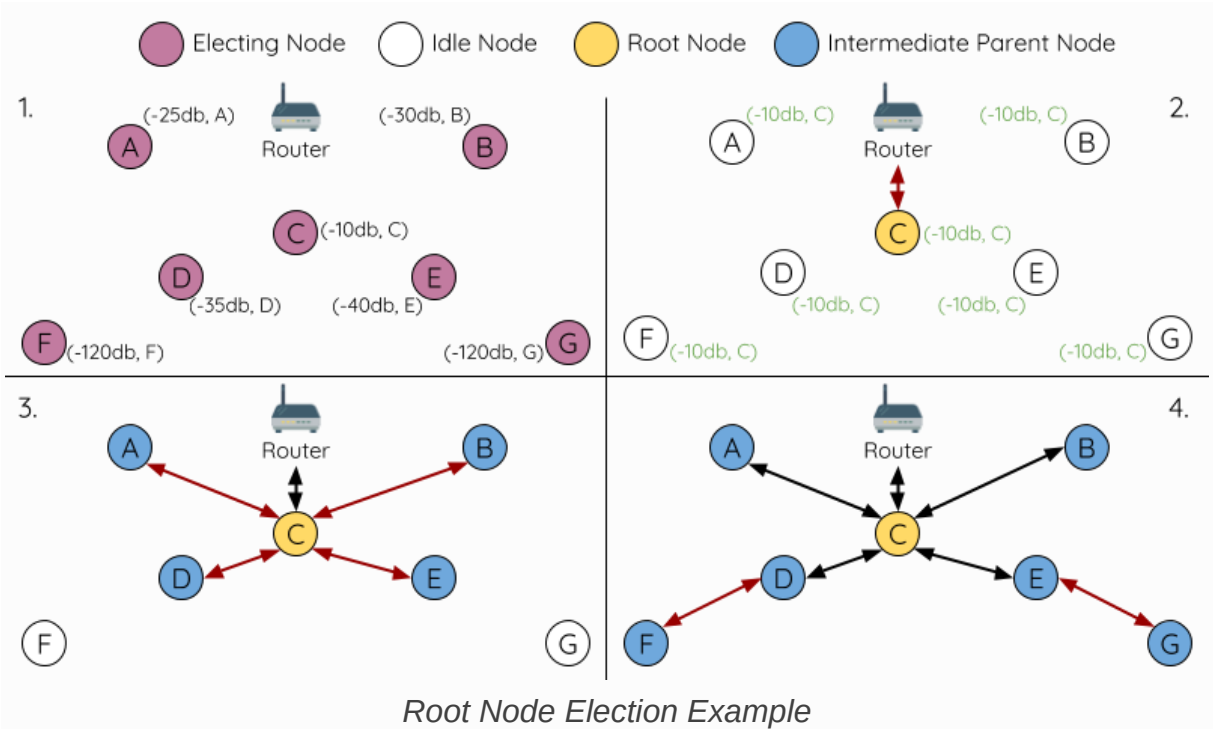


ESP-MESH Routing Tables Example

Figure 5. Routing Tables

3.1.2. Automatic Root Node Selection

The automatic selection of a root node involves an election process amongst all idle nodes based on their signal strengths with the router. Each idle node will transmit their MAC addresses and router RSSI values via Wi-Fi beacon frames. The MAC address is used to uniquely identify each node in the network whilst the router RSSI is used to indicate a node's signal strength with reference to the router.



Root Node Election Example

Figure 6. Demonstrates how an ESP-MESH network is built when the root node is automatically selected.

3.2 IMPLEMENTATION OF CODE & OBSERVATION

```
08:58:00.793 -> --> startHere: New Connection, nodeId = 2761519685
08:58:00.793 -> --> startHere: New Connection, {
08:58:00.793 ->   "nodeId": 2761376217,
08:58:00.793 ->   "subs": [
08:58:00.793 ->     {
08:58:00.793 ->       "nodeId": 2761519685
08:58:00.793 ->     }
08:58:00.793 ->   ]
08:58:00.793 -> }
08:58:00.793 -> Changed connections
08:58:00.793 -> Num nodes: 1
08:58:00.793 -> Connection list: 2761519685
08:58:00.793 -> Adjusted time 4113833. Offset = -683812
```

```
08:58:02.300 -> Changed connections
08:58:02.300 -> Num nodes: 2
08:58:02.300 -> Connection list: 2761519685 681626585
08:58:02.300 -> --> startHere: New Connection, nodeId = 681626585
08:58:02.300 -> --> startHere: New Connection, {
08:58:02.300 ->   "nodeId": 2761376217,
08:58:02.300 ->   "subs": [
08:58:02.300 ->     {
08:58:02.300 ->       "nodeId": 2761519685
08:58:02.300 ->     },
08:58:02.300 ->     {
08:58:02.300 ->       "nodeId": 681626585
08:58:02.300 ->     }
08:58:02.300 ->   ]
08:58:02.300 -> }
```

```
20:15:40.738 -> startHere: Received from 681619293 msg=Temp: 27.94681619293
20:15:41.951 -> startHere: Received from 673458505 msg=Temp: -0.37673458505
20:15:44.449 -> startHere: Received from 673458505 msg=Temp: -0.56673458505
20:15:45.943 -> startHere: Received from 681619293 msg=Temp: 27.87681619293
20:15:46.267 -> startHere: Received from 673458505 msg=Temp: -0.69673458505
20:15:48.452 -> startHere: Received from 673458505 msg=Temp: -0.81673458505
20:15:50.867 -> startHere: Received from 681619293 msg=Temp: 27.87681619293
20:15:52.906 -> startHere: Received from 673458505 msg=Temp: -1.12673458505
20:15:53.556 -> startHere: Received from 681619293 msg=Temp: 27.94681619293
20:15:55.880 -> startHere: Received from 681619293 msg=Temp: 27.87681619293
20:15:57.600 -> startHere: Received from 681619293 msg=Temp: 27.94681619293
20:15:57.785 -> startHere: Received from 673458505 msg=Temp: -1.50673458505
20:16:00.112 -> startHere: Received from 673458505 msg=Temp: -1.62673458505
20:16:02.167 -> startHere: Received from 681619293 msg=Temp: 27.94681619293
```

Figure 7,8&9. Connection status

```
#include "painlessMesh.h"
#include <OneWire.h>
#include <DallasTemperature.h>



OneWire oneWire(15);

DallasTemperature sensors(&oneWire);


#define   MESH_PREFIX     "whateverYouLike"
#define   MESH_PASSWORD   "somethingSneaky"
#define   MESH_PORT       5555

Scheduler userScheduler; // to control your personal task
painlessMesh  mesh;

// User stub
void sendMessage() ; // Prototype so PlatformIO doesn't complain

Task taskSendMessage( TASK_SECOND * 1 , TASK_FOREVER, &sendMessage );

void sendMessage() {
  sensors.requestTemperatures();
  float temperatureC = sensors.getTempCByIndex(0);
  String msg = "Temp: ";
  msg += temperatureC;
  msg += mesh.getNodeId();
  mesh.sendBroadcast( msg );
  taskSendMessage.setInterval( random( TASK_SECOND * 1, TASK_SECOND * 5 ));
}

// Needed for painless library
void receivedCallback( uint32_t from, String &msg ) {
  Serial.printf("startHere: Received from %u msg=%s\n", from, msg.c_str());
}

void newConnectionCallback(uint32_t nodeId) {
    Serial.printf("--> startHere: New Connection, nodeId = %u\n", nodeId);
}

void changedConnectionCallback() {
  Serial.printf("Changed connections\n");
}

void nodeTimeAdjustedCallback(int32_t offset) {
    Serial.printf("Adjusted time %u. Offset = %d\n", mesh.getNodeTime(),offset);
}

void setup() {
  Serial.begin(115200);
  sensors.begin();
//mesh.setDebugMsgTypes( ERROR | MESH_STATUS | CONNECTION | SYNC | COMMUNICATION | GENERAL | MSG_TYPES | REMOTE ); // all types on
  mesh.setDebugMsgTypes( ERROR | STARTUP );  // set before init() so that you can see startup messages

  mesh.init( MESH_PREFIX, MESH_PASSWORD, &userScheduler, MESH_PORT );
  mesh.onReceive(&receivedCallback);
  mesh.onNewConnection(&newConnectionCallback);
  mesh.onChangedConnections(&changedConnectionCallback);
  mesh.onNodeTimeAdjusted(&nodeTimeAdjustedCallback);

  userScheduler.addTask( taskSendMessage );
  taskSendMessage.enable();
}

void loop() {
```

Figure 10. Code

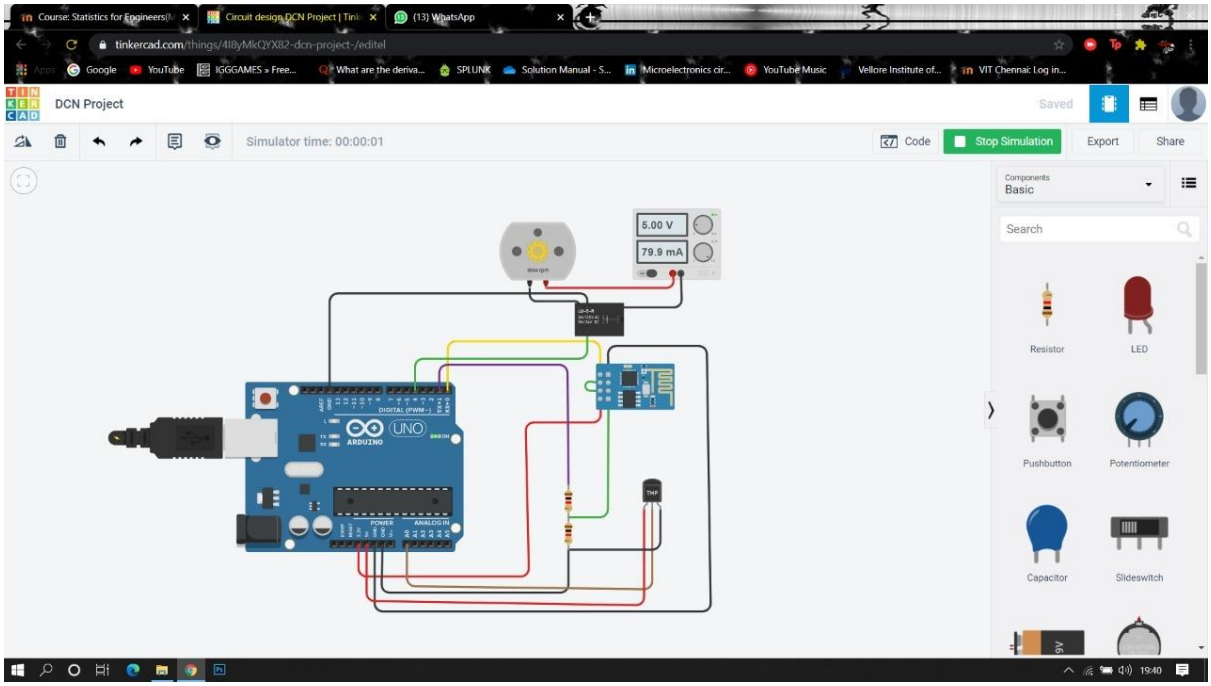


Figure 11. Software simulation

CHAPTER 4

CONCLUSION AND FUTURE WORK

4.1 RESULT, CONCLUSION AND INFERENCE

Growing population has increased the need for the blood supply for various diseases. In every two seconds, some person required blood transfusion and currently India facing problem of the blood shortage. To address the problem an effective system is designed using the Internet of things. The system provides a methodology to fulfil the requirement of blood to the patients/victims without rushing to the blood bank to know the availability of the blood. An IR Sensors are connected to the Arduino board which continuously monitors the status of the available bloodstock. The output data provided by the Arduino is displayed on the webpage using the wifi module so anyone accesses the website and obtained the information of available bloodstock in real time. It will reduce the manpower required at the blood bank to update the online data also reduces the efforts of blood seeker of searching bloodstock at each blood bank. When bloodstock reaches to zero system helps to send a request message to the donor and nearest blood bank. By using IoT the real-time available bloodstock is displaying on the website it minimizes the efforts of blood seeker.

4.2 FUTURE WORK

- Voice operated system can be implemented
- Adding confirmation commands to the voice recognition system.
- Integrating variable control functions to improve the system versatility such as providing control commands other than ON/OFF commands. For example, “Increase Temperature”, “Turn on the compressor” etc.
- Integration of GSM or mobile server to operate from a distance.
- Design and integration of an online web control panel.

4.2.1 COST

S.No	Components	Price in RS
1	**ESP32 WROOM 32d**	**490**
2	**DS18B20**	**90**
3	**RESISTOR**	**1**
4	**DESIGN**	**60**
	TOTAL	**641**

REFERENCES

LIST OF PUBLICATIONS

INTERNATIONAL JOURNALS

1. Smart Blood Bank System Based on IoT, International Research Journal of Engineering and Technology (IRJET)

INTERNATIONAL CONFERENCES

1. Smart Blood Bank System Based on IoT, International Research Journal of Engineering and Technology (IRJET) Conference.

REFERENCE JOURNALS:

1. https://github.com/gmag11/painlessMesh
2. J. S. Lee, Y. W. Su, and C. C. Shen, "A comparative study of wireless protocols:Bluetooth, UWB, ZigBee and Wi-Fi," in Proc. IEEE 33rd Annu. Conf.IECON, pp.46-51, Nov. 2007
3. Ala Al-Fuqaha, Mohsen Guizani, Mehdi Mohammadi, Mohammed Aledhari,and Moussa Ayyash, "Internet of Things: A Survey on Enabling Technologies,Protocols, and Applications" IEEE Communication Surveys Tutorials, Vol.17, No. 4, Fourth Quarter 2015,pp. 2347-2377.
4. G. Muddu Krishna; S. Nagaraju(2016), "Design and implementation of short message service (SMS) based blood bank", 2016 International Conference on Inventive Computation Technologies (ICICT).
5. Do-Sung Kim, Sun K. Yoo, HO Kim, BC Chang ,HS Bae , SJ Kim(2007), "Location Based Blood Bag Management using active RFID and Ubiquitous Sensor Network" ,6th International Special Topic Conference on ITAB, 2007, Tokyo.
6. Wijai Boonyanusith; Phongchai Jittamai (2010), "The Development of Web-Based System for Blood Requisition within Blood Supply Chain", 2010 Seventh International Conference on Information Technology: New Generations.
7. Muhammad Arif; S. Sreevas; K. Nafseer; R. Rahul(2012) "Automated online Blood bank database", 2012 Annual IEEE India Conference (INDICON)
8. Adarsh N, Arpitha J, Md. Danish Ali, Mahesh Charan N, Pramodini G Mahendrakar (2014), "Effective Blood Bank Management Based On RFID in Real Time Systems", International Conference on Embedded Systems - (ICES 2014)
9. Anish Hamlin M R, Albert Mayan J (2016), "Blood Donation And Life Saver-Blood Donation App", 2016 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT)
10. https://www.ndtv.com/health/why-is-there-a-bloodshortage-in-india-1712012