MULTI-PATIENT HEALTH MONITORING USING ARDUINO MEGA

Phaneesh U[1], H Prasanna Kumar[2]

PG – Control & Instrumentation Engineering, University Visvesvaraya College of Engineering,

Bangalore, Karnataka[1].

Associate Professor, Control & Instrumentation Engineering, University Visvesvaraya College of Engineering,

Bangalore, Karnataka[2].

Abstract: In many countries, considering majorly in underdeveloped and developing countries proper health care service is a vital concern. When compared to the requirement of the people, the health centers are far and even the medical personnel are deficient. This reason has made health services for the people who are unhealthy and need health monitoring on regular basis likely to impossible way. This has led the health monitoring of healthy people left far more behind. It is always desirable to implement some system to solve this issue, so that the citizens will not be deprived of the primary care. Smart Healthcare will be significant for the people who have to undergo continuous monitoring, which is not possible to provide outside hospitals.

It is also important that the nearby clinics at rural areas or villages, can be in touch with city hospitals, where they can get about their patient's health condition. This current work presents the smart health monitoring system that uses biomedical sensors to the check patient's condition, and it uses internet to inform the concerned. The biomedical sensors used here are connected to Arduino UNO controller to read the data which in turn is interfaced to an LCD display/serial monitor to see the output.

Keywords: SpO2, Pulse Sensor, LM35 Sensor, Arduino IDE

1. INTRODUCTION

The health of a human body is monitored in traditional approach by using thermometer, oximeter, etc. A thermometer can be used to measure the body temperature, A small changes in body temperature will be detected When, some activities are carried out. Pulse oximeters are devices that are used to measure oxygen level in human body. The device will be attached to a finger, a wrist, a foot, or any other area to read blood flow. The usual process requires a lot of time for registration, appointment and then check-up at the Hospitals. Since, it will be a lengthy process working people tend to ignore the check-ups. Doctors play a significant role in health check-up. Present-day systems use health monitoring devices which will be connected to the patients at the hospital.

The conditions of the patient's and their data is collected and transferred to connected monitoring systems by use of these machines. Doctors and staff should have to visit the patient frequently to examine his/her condition. The smart health monitoring system can be used to provide information required to monitor the health parameters of the patient. Sensors are used to collect and transmit the data and to display the data on the LCD.

2. MOTIVATION

Now a days due to rise of diseases there is a deficiency of testing centres which results to untimely treatment of patients. If patients are tested at a earlier timing then illness can be cured sooner and spreading of diseases can be prevented. To design a system which can measure body temperature, heart pulse rate and SPO2 of patients with greater efficiency and more ease of use.

3. PROBLEM STATEMENT

Existing system

In the existing system, Heartbeat Sensor is used to measure heart rate, is. But in the current proposed system, the heartbeat sensor is used to monitor the heart rate. In the existing system, the sensors to detect body temperature and water level in the body by using the sensor fusion algorithm. In the proposed system, the alert system and LCD display will get activated depending on the heart rate, body temperature and pressure. This system is accurate, flexible, portable, and efficient.

Disadvantage:

- Data availability is not ensured in case of this system.
- Implementation of this system over the wide area is difficult without the usage of cloud system.
- The Data collected by different sensors are not processed.

4. OBJECTIVES

The main objective of patient monitoring is to have a quantitative assessment of the important physiological variables of the patients during critical periods of their biological functions. For diagnostic and research purposes, it is necessary to know their actual value or trend of change. The main aim of Smart health monitoring devices determines the health condition i.e. body temperature, respiratory rate, ECG, and other things by using sensors.

5. APPLICATIONS

People having the following criteria can use this system,

1. A patient who has undergone certain treatment and due to which there is a requirement of monitoring certain health parameters prescribed.
2. A patient who has suffered from a heart attack before and to prevent it to occur in the future, the required health parameters can be monitored remotely.
3. As advised by the doctor, A patient who is suffering from paralytic can use the system to monitor the required health parameters and report the same.
4. During the training period, athletics can check their current health conditions.
5. The people at an advanced age and maybe having failing health conditions can use this system to monitor the current health parameters.

6. EXPERIMENTAL METHODS OR METHODOLOGY

Proposed system uses LM35 Sensor to check the temperature in patient body. In the system we have used two LM35 Sensor. The system makes use of heartbeat sensor which recognizes out the current heartbeat level and display it on the LCD screen. The system also includes buzzer which gives alert alarm for the person supervising the heartbeat rate of the patient and turns on the buzzer as soon as the heartbeat level of the patient varies more or less than the set level normal heartbeat. MAX30102 is an SPO2 sensor that is used to measure the blood oxygen level of the patient. The sensor data is then analysed by an Arduino mega which is connected to output devices. In this project we have to check the two patients at a time. It consumes time check- up is so faster.
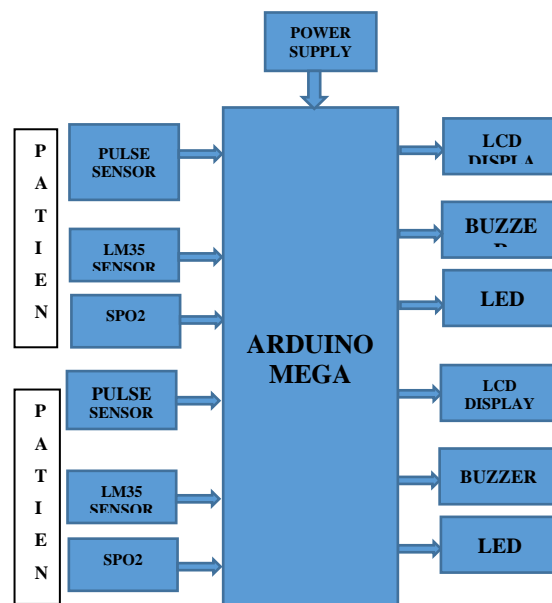
6.1 Block Diagram

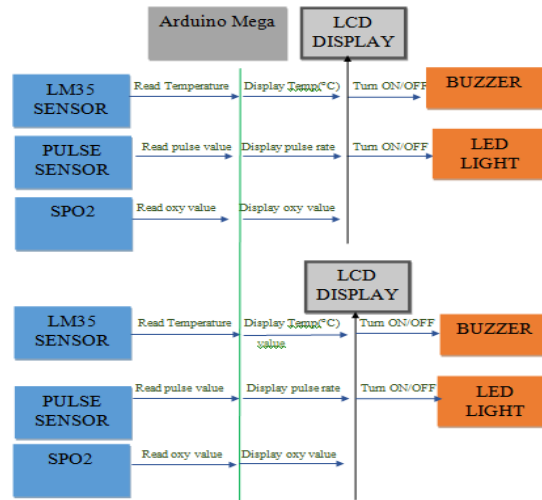**Figure-1: Block Diagram of Multi-Patient Health Monitoring System.**

6.2   Pert Chart

Figure-2: Pert Chart

7.   HARDWARE DETAILS

7.1   Arduino Mega

Figure-3: Arduino Mega

The Arduino Mega 2560, a microcontroller board based on the ATmega2560, which consists of 54 digital input/output pins, 16 analog inputs, 4 UARTs (Hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It is provided with everything needed to support the microcontroller, by simply connecting it to the computer with a USB cable or power it with AC-to-DC adapter or battery to get started. The Mega is compatible with most shields designed for the Arduino Duemilanove or Diecimila.

The Mega 2560 R3 also adds SDA and SCL pins. Along with, there are 2 new pins near to the RESET pin. One is the IOREF which allows the shields to adapt to the voltage provided from the board. The second one is not connected and can be reserved for future purposes. The Mega 2560 R3 will work with all existing shields but can be adapted to new shields which uses these additional pins.

7.2   SPO2

SpO2 also known as oxygen saturation, is a measure of the amount of oxygen-carrying hemoglobin in the blood relative to the amount of hemoglobin not carrying oxygen. Our body needs blood with certain level of oxygen to function efficiently.

Figure-4: MAX30102 Sensor

MAX30102 is a Bio-Sensor module which is an integrated Pulse oximeter and Heart rate monitor. This module integrates a red LED and an infrared LED, Photo detector, optical components, and low-noise electronic circuitry with ambient light suppression. This is a wearable device and can be worn on fingers, earlobes, and wrist.

I2C Compatible communication interface is used to transmit the obtained values to the Arduino Mega 2560 to calculate the Heart Rate and Blood Oxygen level.

Main Parameters:
o   LED peak wavelength: 660nm/880nm.
o   LED power supply voltage: 3.3~5V.
o   Detection signal type: light reflection signal (PPG).
o   Output signal interface: I2C interface Communication.
o   interface voltage: 1.8~3.3V~5V (optional).
o   Board assembly hole size: 0.5 x 8.5mm.

Pin Description: VIN: main power input terminal 1.8-5V 3-bit pad: Select the pull-up level of the bus, depending on the pin master voltage, select 1.8v or 3_3v (this terminal contains 3.3V and above) SCL: the clock connected to the I2C bus; SDA: data connected to the I2C bus; INT : Interrupt pin of the MAX30102 chip; RD: RED LED ground terminal of MAX30102 chip, generally not connected; IRD: The IR LED ground of the MAX30102 chip is generally not connected; GND: Ground wire.

Light absorption of Oxygenated and De-Oxygenated Hemoglobin is not the same. So, using this principle, light source of specific wavelength selective for Oxy-Hemoglobin (HbO2) and Hemoglobin (Hb) is transmitted on Human tissue, the reflected light is received by Photodiode. The change in the volume of the arterial pulsation changes the transmittance of the light received by the photodiode. The light that is received converted into an electrical signal.

7.3 LM35 Temperature Sensor:
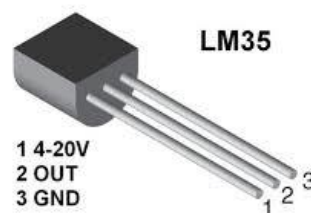
Figure-5: LM35 Temperature Sensor

LM35 is a Precision IC Temperature sensor, the output of which is directly proportional to temperature (in ºC). The temperature measurement using the LM35 is more accurate than using thermistor. The sensor is sealed such that it is not subjected to oxidation. The self-heating to air is comparatively less and doesn't cause more than 0.1 ºC temperature rise in still air.

The temperature range which the sensor can operate is from -55 ºC to 150 ºC. For every ºC rise/fall in ambient temperature the output voltage varies by 10mV, i.e., its scale factor is 0.01V/ ºC.

7.4 Pulse Sensor

Figure-6: Pulse Sensor

The Pulse sensor used detects the heartbeat by the method of a Photoplethysmography (PPG). PPG illuminates the skin optically and measures the pulse as a change of resistance to the blood flow in the skin. The sensor could be connected to the finger/earlobe. Since, the heart pulse is detected, heart rate and Inter Beat Interval (IBI) are calculated alongside.

7.5 Buzzer

Figure-7: Buzzer

It's simple, Tone(buzzer, 1000) sends a 1KHz sound signal to pin 9 of Arduino, the delay(1000) will pause the program for one second and noTone(buzzer) stops the signal sound. The loop() routine will make this run again and again making a short beeping sound.

7.6 LED

Figure-8: LED

Light Emitting Diode is a Semiconductor light source that emits light when current flows through it. Here we are using them to give indication if the Temperature of the patient is high than the normal temperature.
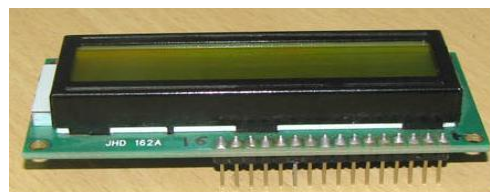
7.7 LCD display

Figure-9: Liquid Crystal Display

LCD (Liquid Crystal Display) screen is an electronic display module and find a wide range of applications. A 16x2 LCD display is very basic module and is very commonly used in various devices and circuits. These modules are preferred

over seven segments and other multi segment LEDs. The reasons being: LCDs are economical; easily programmable; have no limitation of displaying special & even custom characters (unlike in seven segments), animations and so on.

A 16x2 LCD means it can display 16 characters per line and there are 2 such lines. In this LCD each character is displayed in 5x7 pixel matrix. This LCD has two registers, namely, Command and Data.

The command register stores the command instructions given to the LCD. A command is an instruction given to LCD to do a predefined task like initializing it, clearing its screen, setting the cursor position, controlling display etc. The data register stores the data to be displayed on the LCD. The data is the ASCII value of the character to be displayed on the LCD. Click to learn more about internal structure of a LCD.
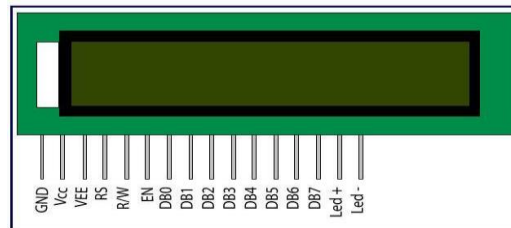

Figure-9: Liquid Crystal Display Pin Diagram

Table.1. LCD Pin Description

Pin no	Function	Name
1	Ground (0V)	Ground
2	Supply voltage; 5V (4.7V – 5.3V)	Vcc
3	Contrast adjustment; through a variable resistor	V_{EE}
4	Selects command register when low; and data register when high	Register Select
5	Low to write to the register; High to read from the register	Read/write
6	Sends data to data pins when a high to low pulse is given	Enable
7		DB0
8		DB1
9		DB2
10		DB3
11	Backlight VCC (5V)	DB4
12		DB5
13		DB6
14		DB7
15		Led+
16	Backlight Ground (0V)	Led-

8. SOFTWARE REQUIREMENT

8.1 Arduino IDE Programming:

The Arduino Integrated Development Environment - or Arduino Software (IDE) - contains, a message area, a text console, a text editor for writing code, a toolbar with buttons for common functions and a series of menus. It connects to the Genuine hardware and Arduino to upload programs and communicate with them.

Figure-2: Arduino Tool

> **Writing Sketches**

Programs composed employing Arduino Software (IDE) are called portrays. These representations are composed in the content tool and are spared with the document expansion ino. The proof-reader has highlights for cutting/gluing and for looking/supplanting content. The message zone provides criticism while sparing and sending out and additionally it shows blunders. The comfort presents the content yield by the Arduino Software (IDE), including complete blunder messages and the other data. The base righthand corner of the window displays the arranged board and serial port. The toolbar takes permit you to check and transfer programs, make, open, and spare outlines, and open the serial screen.

Upload
Accumulates the code entered and transfers it to the designed board. See transferring beneath for the points of interest.
Note: In case if you are utilizing an outer software engineer with your board, you can hold down the "move" key on your PC when utilizing this symbol. The content will change to the "Transfer utilizing Programmer"

New
Creates the new sketch.

Open
Presents a menu of the representations mentioned in your sketchbook. Clicking one will open it inside the present window and overwrites its substance. Note: Because of a bug in Java, this menu doesn't scroll; on the off chance that you must open a draw late in rundown and utilize the File | Sketchbook menu.

Save
Saves your sketch.

Serial Monitor
Opens a serial monitor.

Additional commands are found within the five menus:
- **File**
- **Edit**
- **Sketch**
- **Tools**
- **Help**.

The menus are setting sensitive, which implies only for those things significant to the work as of now being done are accessible.
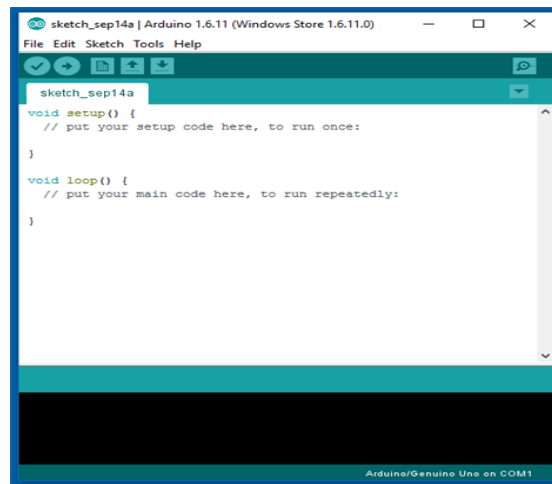
Figure-3: Arduino Software

FILE

- *New*

o   Makes another instance of the editorial manager, with the absolute minimum structure of an outline already set up.

- *Open*

Permits to stack an outline document which analyze through the PC drives and envelopes.

- *Open Recent*

Provides a short list of the most current sketches, which are ready to be opened.

- *Sketchbook*

Exhibits the current portrays inside the sketchbook envelope structure; touching on any name displayed opens the relating sketch in another proof-reader example.

- *Examples*

Any case provided by the Arduino Software (IDE) or library appears in this menu. Every one of the cases are catalogued in a tree that permits simple access by topic or library.

- *Close*

Closes the item of the Arduino Software from which it is clicked.

- *Save*

Spares the draw with the current name. On the off chance that the record which has not been named some time recently, a name will be given in a "Spare as.." window.

- *Save as*

Allows to save the present sketch with a different name.

- *Page Setup*

It displays the Page Setup window for printing.

- *Print*

Sends the present sketch to the printer as per the settings defined in Page Setup.

- *Preferences*

Opens the Preferences window where some settings of the IDE may be changed subsequently, as the language of the IDE interface.

- *Quit*

Shuts all the IDE windows. The same illustrations open when Quit was picked, will be automatically revived whenever the IDE is started again

- EDIT
- *Undo/Redo*

Backpedals of at least one stages you performed while altering; when you do backpedal, you may go ahead with Redo.

- *Cut*

Removes the chose content from the supervisor and highlights it into the clipboard.

- *Copy*

Copies the selected message in the manager and highlights it into the clipboard.

- *Copy for Forum*

Duplicates the code of your portray to the clipboard in a frame appropriate for displaying on the discussion, and finishes with sentence structure shading.

- *Copy as HTML*

Duplicates the code of your draw to the clipboard as HTML, reasonable for Implanting them in website pages.

- *Paste*

Places the substance of the clipboard at the cursor position, in the editorial manager.

- *Select All*

Chooses and highlights the complete substance of the editorial manager.

- *Comment/Uncomment*

Places or removes the remark/marker toward the start of each chose line.

- *Increase/Decrease Indent:*

Includes or removes a space toward the beginning of each chose line, moves the text   one   space   on   the privilege or dispensing with a space toward the start.

- *Find*

Opens the Find and Replace window where you can select the content to seek inside the current draw as indicated by a few choices.

- *Find Next*

Spots the following event - if any - of the string indicated as the seeking thing in the Find window, with respect to the cursor position.

- *Find Previous*

Spots the past event - if any - of the string shown as the hunt item in the Find window with respect to the cursor position.

SKETCH

- *Verify/Compile*

Checks your outline for mistakes, arranging it; it will report memory utilization for code and factors in the reassure territory.

- *Upload*

Arranges and stacks the twofold document onto the designed board by using the configured Port.

- *Upload Using Programmer:*

This will overwrite the boot loader on the board; you should utilize Tools >Burn Boot loader to reintroduce it and have the capacity to Upload to USB serial port once again. In any case, it permits you to use the complete limit of the Flash memory for your sketch. It would be constant if you take note of that, this summon doesn't consume the wires. To do this way, a Tools - >Burn Boot loader charge must be executed.

- *Export Compiled Binary*

Spares a .hex document that may be kept as chronicle or sent to the board for utilizing different tools.

- *Show Sketch Folder*

Opens the present sketch folder.

- *Include Library*

Includes a library to your portray by embedding's #include explanations towards the begin of your code. For further points of interest, see libraries underneath. Furthermore, from this menu thing you can move to the Library Manager and import new libraries from compress records.

- *Add File...*

Includes a source record to the outline (it will be duplicated from its current area). The new document displays in another tab in the portray window. Documents can be expelled from the portray uses the tab menu available, clicking on the little triangle symbol beneath the serial screen one on the correct side of the toolbar.

TOOL

- *Auto Format*

This organizes your code pleasantly: i.e. indents it so that opening and closing wavy braces line down, so that the announcements inside wavy supports are indented more.

- *Archive Sketch*

Documents a duplicate of the current draw in .zip arrange. The file is kept in the same index similar to the draw.

- *Fix Encoding & Reload*

Resolves conceivable errors seen between the supervisor burn outline and other operating frameworks scorch maps.

- *Serial Monitor*

▪ Opens the serial screen window and initiates the trading of information with any connected board on the at currently chosen Port. This as a rule resets the board, if the board bolsters, Reset over the serial port opening.

- *Board*

Selects the board that you're using. See below for depictions of the different boards.

- *Port*

This menu involves all the serial gadgets (genuine or virtual) on your machine. It should consequently recollect each time you open the top-level instruments menu.

- *Programmer*

For selecting a hardware software engineer, when programming a board or chip and not utilizing the locally available USB-serial association. Typically, you do not require this however in the event that you're consuming a bootloader to another microcontroller, you will use this.

- *Burn Boot loader*

The things in this menu allows you to consume a bootloader onto the microcontroller on an Arduino board. This is not necessary for ordinary using of an Arduino or Genuino board however it is valuable in the event that you buy another AT mega microcontroller Ensure that you've selected the right board from the Boards menu before consuming the bootloader on the objective board. This charge includes an extra set the correct circuits.

8.2 Test Results
Table.2. Results of Tests Performed

Description	LM35 Temperature Sensor
Input	5v
Expected output	Normal body temperature 37 degrees Celsius
Actual Result/Remarks	35 degrees Celsius
Passed(?)	Yes
Description	Pulse Rate Sensor
Input	5 or 3.3v
Expected output	(70-100) beats per min
Actual Result/Remarks	90 beats per min
Passed(?)	Yes
Description	SPO2 Sensor
Input	3.3v
Expected output	0-100
Actual Result/Remarks	94
Passed(?)	Yes

9. CONCLUSION

In this project with the SpO2 sensor we were able to measure and display the Blood oxygen level of the patient. And using the LM35 temperature of the patient can be measured and displayed, buzzer and the LED will be turned on when the temperature of the patient is more than the temperature setpoint in the code. Also, using the Pulse sensor we were able to measure the pulse rate of the heart. These sensors integrated project can be used in the General wards of Hospitals where multiple patient's temperature, Blood oxygen level, Pulse rate needs to be monitored continuously.

10. REFERENCES

1. Jorge Gómez, "Patient Monitoring System Based on Internet of Things: A survey", Computer Networks, Vol.54, pp. 2787-2805, 2016.
2. Sneha N. Malokar 1, Samadhan D. Mali2, "Patient Monitoring System Based on Internet of Things using Raspberry pi: Key features, application and open issues", Computer Communication, Vol.54, pp., 2016.
3. Gennar Tartarisco, " Patient Health Monitoring System using IOT device https://www.elprocus.com/wpcontent/uploads/2017/07/heartbeat-sensor.jpg
4. Giovanni Baldus, "Design of IOT Based Smart Health Monitoring and Alert System" 2016 Center for TeleInFrastuktur, Aalborg University, Denmark, P.P
5. Pioggia, "Personal Health System architecture for stress monitoring and support to clinical decisions", Computer Communications Vol.35, pp.1296-1305, 2017.
6. Franca Delmastro, "Pervasive communications in healthcare", Computer Communications Vol.35, pp.1284-1295,2017.
7. ArunaDevi.S et al. "SMART HUMAN HEALTH MONITORING SYSTEM BY USING IoT/ International Journal of Computer Science & Engineering Technology (IJCSET) ISSN,2017
8. Manikandan Venkatesan Gomathy. "Development of an Oxygen Saturation Monitoring System by Embedded Electronics". University of Arkansas, Fayetteville,2017
9. Gayathri.R.Hepsiba.D, "Development of a Pulse Oximeter using MSP430 Microcontroller", International Journal of Engineering and Technology (IJET), Vol.5 No.3, Jun-Jul 2013.

10. Joseph Bailey, Michael Fecteau and Noah L. Pendleton. "Wireless pulse oximeter". Worcester Polytechnic Institute, 2008.

11. Zhang, Feng. "Pulse Oximeter Design Using Microchip's Analog Devices and dsPIC Digital Signal Controllers (DSCs)". Microchip Technology Inc, 2015.

12. Rekha Chandra, R., K. P. Safeer, and P. Srividya. "Design and Development of Miniaturized Pulse Oximeter for Continuous SpO2 and HR Monitoring with Wireless Technology." International Journal of New Technology and Research (IJNTR), 2015: 11-15.

13. Saraswathi S.; Aravindhan P.; Boovesh L.; Sureshkumar V.. "IOT Based Future Technologies and Developments in Agriculture". International Research Journal on Advanced Science Hub, 2, 8, 2020, 131-135. doi: 10.47392/irjash.2020.107

14. Ravindra Parab; Pawan Pandey. "Design Smart Solar Inverter using e-waste with IOT control". International Research Journal on Advanced Science Hub, 3, 1, 2021, 5-12. doi: 10.47392/irjash.2021.002

15. Mr. Dinesh; Mr. Meiyarasan; Mr. Prabu. "Health Monitoring System Using IOT Sensor". International Research Journal on Advanced Science Hub, 3, 4, 2021, 83-89. doi: 10.47392/irjash.2021.119

16. Jahan, Esrat, Tilottoma Barua, and Umme Salma. "An Overview on Heart Rate Monitoring and Pulse Oximeter System." International. 2014.

17. Babikir, Sharief Fadul, and Reem Abedalmoniam Ismail. "Oxygen Level Measurement Techniques: Pulse Oximetry." Journal of Science and Technology, 2015.

18. Tamayo, Michael, Andrew Westover, and Ying Sun. "Microcontroller based pulse oximeter for undergraduate capstone design." Bioengineering Conference, Proceedings of the 2010 IEEE 36th Annual Northeast. IEEE, 2010.