ISSN 2063-5346

NODE MCU BASED ROOM AMBIENT ALONG WITH PATIENT MONITORING SYSTEM

Mritunjay Bistu[1], Shashwata Mukherjee[1], Subhrajit Mitra[1], Tiya Dey Malakar[1] , Arpita Ghosh[1*]

Article History: Received: 01.02.2023 Revised: 07.03.2023 Accepted: 10.04.2023

Abstract

Health plays a major role in our daily life. Monitoring the health of a patient is very crucial to provide proper treatment at the proper time .The Real-time health monitoring is developed for initial detection of life threatening diseases through advance sensing and communication technology to save lives of patient through advanced treatment .The system gives body temperature, heart-rate and blood oxygen(SpO2). Moreover the system also provides the room ambient parameters such as room temperature, humidity to adjust them according to the patient's health requirement. The NodeMCU development board on ESP32 is interfaced with the sensors for taking the different data such as temperature, humidity and body vitals. The controller ESP32 is utilized for the wireless data transmission part with the inbuilt Wi-Fi module android Blynk app. The patient's record is stored over a period of time stored using an Android app. The results from the proposed prototype can be employed to various clinical investigations

Keywords— IoT, Patient monitoring system, NodeMCU dev board, room ambient, Blynk App,COVID-19

[1]Dept. of Electronics and Communication Engineering,RCC Institute of Information Technology,Kolkata,India.

Email: mritunjaybistu09@gmail.com, shashwatamukherjee@gmail.com, mailtosubhra99@gmail.com

, tiya_biet@yahoo.co.in, *arpita161@gmail.com

DOI:10.31838/ecb/2023.12.s1-B.382

1.0 Introduction

The healthcare technology has become our priority mainly during the recent pandemic situation due to the outbreak of COVID. The Internet of Things (IoT) is playing a significant role in the present revolutionized form of the healthcare system. Monitoring the health status of a COVID patient is a hard task because of the communicable nature of the coronavirus. Mostly, the elderly COVID patients need to be monitored periodically. So far many works are reported on the smart health monitoring system . So far several works are reported in the field of automatic health monitoring[1]-[4] ,some of the systems also included micro controller based systems[5-6], some of them also included IoT[7-9]. Another factor which also plays an important role while maintain certain health condition of the patient is the room ambient[10]. Many home automation systems[11-12] are separately available for the room temperature or humidity monitoring. Khan et.al.[13]in 2019 described a smart health monitoring system with IoT which used Arduino UNO board a ,Bluetooth module and LCD display. The wireless data transfer was implemented using the Bluetooth module. Additionally they used buck converter for hardware implementation. The mobile app part was implemented using the Massachusetts Institute of Technology(MIT) App Inventor. In 2020: Taiwo and Ezugwu in their work[14] demonstrated a smart healthcare-monitoring system for the COVID situation. It was mainly implemented using the concept of home automation. But so far no system is reported which include both the patient monitoring features and the room ambient monitoring option remotely using a compact system. Most of the available systems are using Arduino[15-17] along with separate Bluetooth or Wi-Fi module for transferring the data to the cloud.

The proposed system is developed to accommodate both the requirements in an automatic way .The basic microcontroller used here is ESP32. The ESP32 is a microcontroller developed by Espressif Systems. ESP32 has gain a lot of popularity as it can be used for several applications[18]. This device uses Node MCU ESP32 development board(ESP32-DevkitC)[19] and an android Blynk app[20] for tracking the health of patient. Hence, associated health parameters for instance body temperature, BPM(heart rate per minute), SpO2 (blood oxygen levels) as well as room ambient (temperature and humidity) can be observed from any device such as Laptop, PC, Smartphone, Smart TV etc. To measure BPM and SpO2, here the MAX30100 (pulse oximeter sensor) is used. Similarly, for measuring body temperature, DS18B20 (temperature sensor) is used. Meanwhile, if the patient is isolated in a room at his home or at hospital room, room temperature and humidity level can be also monitored. The room ambient can be maintained at a certain temperature and humidity level so that the patient does not feel uncomfortable. For room ambient monitoring DHT11 Temperature & Humidity sensor is used. Here mainly MAX30100, DHT11, and DS18B20 are interfaced with NodeMCU ESP32.The parameters measured and displayed are the Blood Oxygen(SpO2), BPM, Body Temperature, room temperature, and humidity online on the Blynk Application.

The second section of the paper gives overview of all the components required for implementing the entire hardware part. The software requirements are also included in the same section. The third section deals with the entire proposed work idea along with the flow chart, circuit diagram, code uploading steps. The result discussion is included in the fourth section . The future scope and conclusion is included in the last or fifth section.

2.0 HARDWARE AND SOFTWARE IMPLEMENTATION

The proposed system monitors the patient health remotely along with the room ambient.

2.1 THE HARDWARE IMPLEMENTATION OF THE SYSTEM REQUIRES THE FOLLOWING COMPONENTS:

A. Node MCU (ESP32) development board, Pulse Oximeter Sensor (MAX30100 Module),Body Temperature Sensor (DS18B20) , Room Temperature and Humidity Sensor (DHT11),Connecting Wires (Jumper Wire), Breadboard.

i.MAX30100 Pulse Oximeter Sensor: It is an integrated heart rate monitor and pulse oximetry sensor. It accommodates 2 LEDs, optimized optics, a photo detector and low-noise analog signal processing for the detection of pulse oximetry and heart-rate signals. The power supply requirement for MAX30100 is from 1.8V and 3.3V. The software power down is possible with negligible standby current and permitting the power supply to remain connected at all times. The Shutdown Current is ultra-low (0.7µA) and the data output capability is very fast. Fig.1 shows the pulse oximeter and heart rate sensor MAX30100 with the pin configurations. The device has two LEDs(one emits light ,another emitting infrared light). Only the infrared light is needed for determination of pulse rate. The oxygen level in the blood is measured using the red light and infrared light. The oxygenated blood increases when the heart pumps blood ,as a result more blood flows. The volume of oxygenated blood reduces during the period when heart relaxes. The identification of the time difference between the oxygenated blood volume increase and decrease, the pulse rate can be determined. The oxygenated blood passes more red light by absorbing more infrared light . On the other hand the deoxygenated blood passes more infrared light by absorbing the red light . The function of MAX30100 is to read both light source absorption levels and store them in a buffer that can be read via I2C.

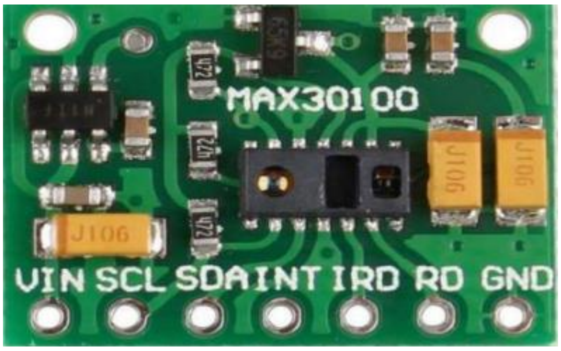


Fig.1.Pulse Oximeter and heart rate sensor MAX30100

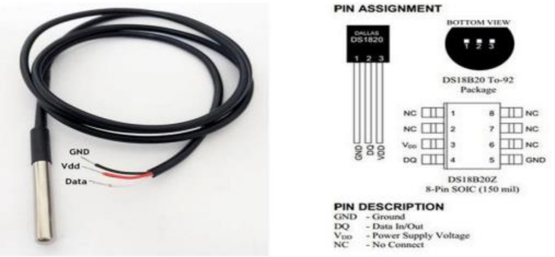


Fig.2. Body Temperature Sensor(DS18B20)

ii.Body Temperature Sensor (DS18B20): The DS18B20 temperature sensor works as thermometer ranges between 9-12 bit Celsius temperature measurements and also has an additional application alarm convertible user programmable, upper and activate points. The DS18B20 connects around 1 wire bus that interpreter's one data line (earthen) in order to contact the main microprocessor. Moreover this also passes through the power directly from the data line by excluding the requirements of external power supply. DS18B20 has an exclusive serial code of 64-bit, that allows varies DS18B20 to operate simultaneously. Hence it is easy to monitor multiple thermistors dispersed over a huge area to practice one microprocessor. Applications that has a profit from which a parameter counts in advantage which consist of HVAC environmental controls , temperature control systems , apparatus , and process controlling and monitoring systems. Fig.2 shows the pin configuration of DS18B20 body temperature sensor.

iii. Room Temperature and Humidity Sensor (DHT11): DHT11 is a low-cost digital sensor for sensing temperature and humidity. The interfacing of this sensor is easy with ESP32 for measuring humidity and temperature on instantaneous basis. The room temperature sensor(DHT11) with pin configurations is shown in Fig 3.

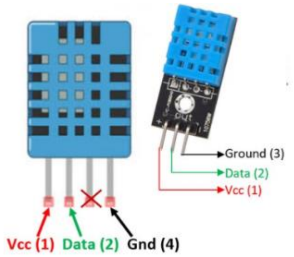


Fig.3. Room Temperature Sensor and Humidity Sensor(DHT11)

iv. NodeMCU ESP32 Dev Board: The pin configuration of the NodeMCU ESP32 development board is shown in Fig.4. This board supports inbuilt WiFi (WiFi Protocol IEEE 802.11 b/g/n),Bluetooth (4.2)connectivity and keys. The dev board comes with the features of Onboard ESP32-S module, CH340, USB to UART converter, USB port for power input, UART debugging (firmware programming),

extension headers(2x19pin),2x keys for using them as reset or user-defined. The processor used in this dev board is ESP32. It comes with 32Mbit built-in-Flash , onboard PCB antenna, supports the peripheral interface through UART/I2C/I2S /ADC/DAC/PWM/ GPIO /SDIO, supports WiFi modes Station, SoftAP , SoftAP+Station. The frequency range is near about 2.4GHz . The power supply requirement for the board is 5V whereas the logic level is 3.3V. The overall dimension of the board is 48.26mm x 25.4mm

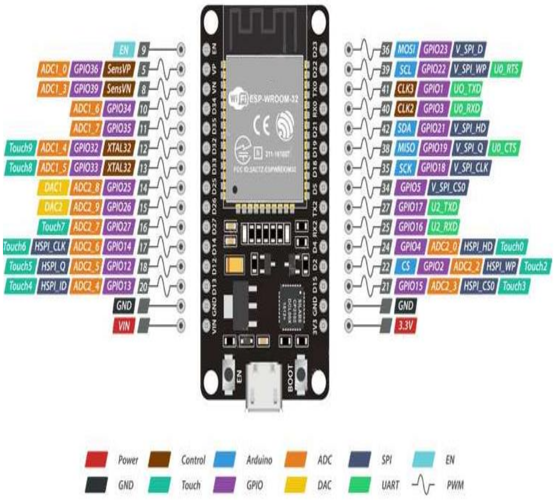


Fig.4.ESP32 Dev board pin diagram

2.2 THE SOFTWARE IMPLEMENTATION IS DONE USING ARDUINO IDE AND BLYNK APP:

i. Arduino IDE(Integrated Development Environment)- It is an open source application where coding, compiling, and uploading a file in an Arduino device is possible. The Arduino modules having default in built features can be adapted by this software. It runs on the java and is available for operating systems such as MAC, Linux, Windows . It supports a many Arduino modules such as Arduino Uno, Mega, Leonardo, Micro etc. Every module contains an onboard which is in built by default. The languages C and C++ along with special code structuring rules are supported by this IDE.

ii. Android Blynk App: The Blynk is an open source Android app which is designed and developed in order to control the hardware via internet of things (IoT). It can display ,accumulate and visualize the sensor data digitally. It can also do other functions such as:

Blynk App –this app gives opportunity for creating amazing interfaces for a project using multiple widgets which is an in-built app.

Blynk Server – It acts as an interface between the smart phone and hardware which is responsible for the communication. It can be also used as Blynk cloud or compile private cloud. It's an open source that can control any number of devices.

Blynk Libraries- supports all the standard hardware platforms

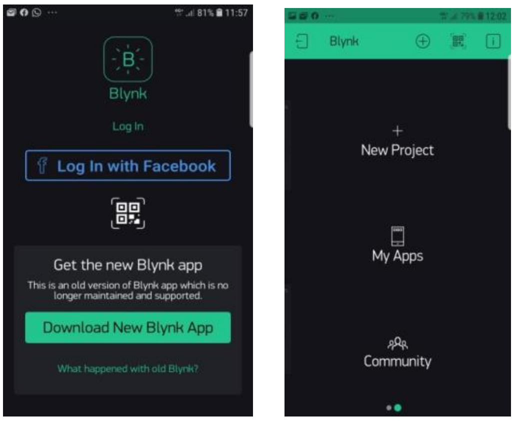

Blynk
Log In
Log In with Facebook
Get the new Blynk app
This is an old version of Blynk app which is no longer maintained and supported.
Download New Blynk App
What happened with old Blynk?
Blynk
New Project
My Apps
Community

Fig .5. Blynk app (a) log in page, (b)project creation page

3. PROPOSED SYSTEM:

The flow Chart of the patient monitoring system along with the room ambient is shown in the flow chart below (Fig .6). The sensors (MAX30100, DHT11, DS12B20)for measuring the patient vitals are connected with the patient and the Blynk App through the ESP32 microcontroller.

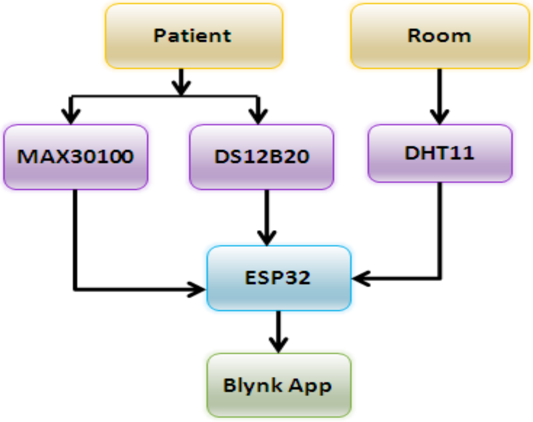


Fig.6. Flow Chart of the Proposed Room Ambient cum patient vital monitoring System

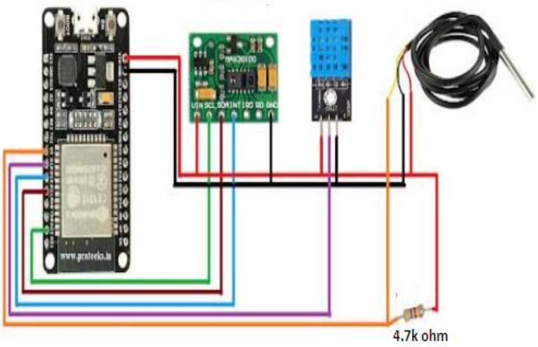


Fig.7. Circuit Diagram

The circuit diagram for the entire hardware part is shown in the Fig.7 which includes all the interfacing of MAX30100, DHT11,DS18B20 sensor with NodeMCU ESP32. The circuit diagram & connection is given figure 7. All the sensor can work at 3.3V VCC. GND to GND connection is made for all. MAX30100 is an I2C Sensor, so its SDA & SCL pin is connected to GPIO21 & GPIO22. The INT pin of the sensor is connected to GPIO19 of ESP32. The DHT11 output pin is connected to ESP32 pin GPIO18. Similarly, the output pin of DS18B20 is connected to GPIO5 of ESP32. A 4.7K pull-up resistor is connected between output pin & VCC pin of DS18B20. The entire real life circuit implementation is shown in the Fig 8 .

The code uploading steps in ESP32 :

- ESP32 is connected to the PC via USB cable .The circuit connection is done as per the above diagram. Here GPIO2 has been used for the purpose.
- The code is copied in the Arduino IDE window by providing proper WiFi credentials.
- In the next step ESP32 Dev module is selected by clicking on Tools--> Board--> ESP32 Dev module.
- The port is then selected to which your ESP32 is connected.
- Finally the code is uploaded.
- To start ESP ,the serial monitor is opened and the reset button of ESP32 is pressed.

- In Blynk app the project created earlier is selected and PLAY is tapped. Through this step live data can be seen in the phone.

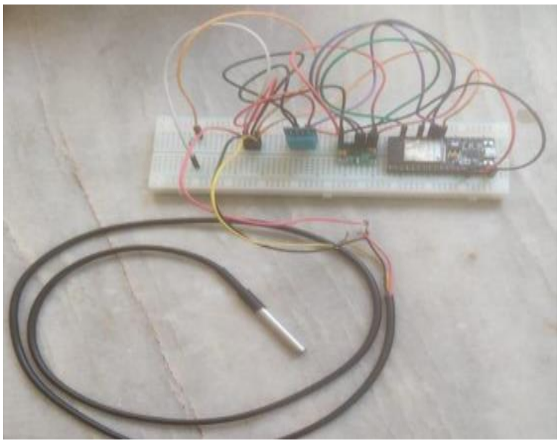


Fig.8.Circuit implementation

4. RESULT AND DISCUSSION:

The results of the work are shown in two ways one through the Blynk app shown in Fig 9 and another one using the serial monitor in Fig.10. Fig.9 shows the results in the gauge for the parameters such as room temperature, humidity, body temperature, SpO2 and BPM(heartrate). The use of such an app makes the proposed work more user friendly due to the visual interpretation of the parameters using gauge. Fig.10 shows the same results using the serial monitor available.

After the code is uploaded, the result is visible in the serial monitor for the BPM , SpO2,Body Temperature, Room temperature and humidity values after the NodeMCU is connected to the WiFi. On the Android Blynk App, the BPM,SpO2,Body Temperature , Room Temperature and Humidity value is uploaded after a second and the change in gauge and display parameters can be seen then.

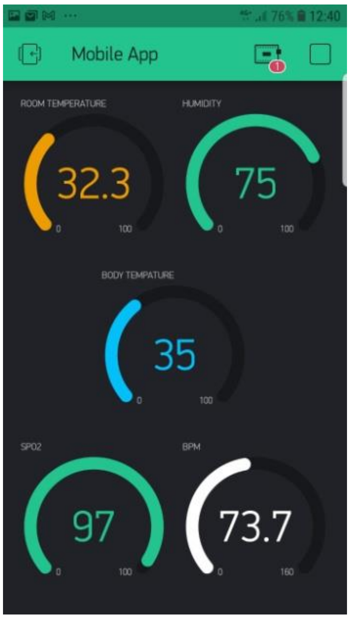
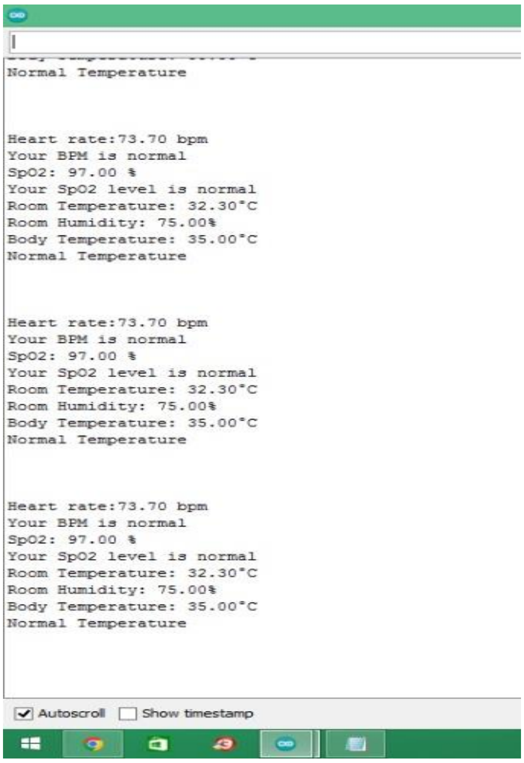


Fig.9 Blynk App results



```
Normal Temperature



Heart rate:73.70 bpm
Your BPM is normal
SpO2: 97.00 %
Your SpO2 level is normal
Room Temperature: 32.30°C
Room Humidity: 75.00%
Body Temperature: 35.00°C
Normal Temperature



Heart rate:73.70 bpm
Your BPM is normal
SpO2: 97.00 %
Your SpO2 level is normal
Room Temperature: 32.30°C
Room Humidity: 75.00%
Body Temperature: 35.00°C
Normal Temperature



Heart rate:73.70 bpm
Your BPM is normal
SpO2: 97.00 %
Your SpO2 level is normal
Room Temperature: 32.30°C
Room Humidity: 75.00%
Body Temperature: 35.00°C
Normal Temperature
```

Fig.10.Serial Monitor Results

5. CONCLUSION:

The proposed system provides a compact system which will provide the patient health parameters along with the room ambient parameters. Those parameters can be accessed remotely and accordingly the measures can be taken. Instead of using ARDUINO board along with the separate Wi-Fi module a single Node MCU ESP32 Development board serves the purpose. Different sensors are interfaced for patient health monitoring such as temperature(DS18B20), heart rate, SpO2(MAX30100). The room temperature and humidity data are collected by the ESP32 and sent to the cloud by using sensor DHT11. Finally all data can be visualized using the Blynk app using several gauges. This will be helpful for the healthcare industry as it can help both the patients and hospitals to function efficiently. Also, by incorporating some additional features the, this device could be used for emergency alerts to notify the nearby hospital in case of a rise in the heart rate/decrease in SpO2 of the patient. When there is a lack of hospital beds or if there is any patient with a highly contagious disease this device could be used to monitor the patient’s health remotely by the doctors.

References

1. Fu, Hailing & Sharif Khodaei, Zahra & Aliabadi, M.H.. (2018). An Event-Triggered Energy-Efficient Wireless Structural Health Monitoring System for Impact Detection in Composite Airframes. PP. 1-1. 10.1109/JIOT.2018.2867722.
2. Chang, Hung-Fu & Lin, Tzu-Kang. (2018). Real-time Structural Health Monitoring System Using Internet of Things and Cloud Computing.
3. Araujo, Alvaro & García-Palacios, Jaime & Blesa, Javier & Tirado-Andrés, Francisco & Romero, Elena & Samartín, Avelino & Nieto-Taladriz, Octavio. (2012). Wireless Measurement System for Structural Health Monitoring With High Time-Synchronization Accuracy. IEEE T. Instrumentation and Measurement. 61. 801-810. 10.1109/TIM.2011.2170889.
4. Basko, Artem & Ponomarova, Olena & Prokopchuk, Yurii. (2021). Review of Technologies for Automatic Health Monitoring of Structures and Buildings. International Journal of Prognostics and Health Management. 12. 10.36001/ijphm.2021.v12i2.3015.
5. Islam, Md & Rafi, Fida & Ahmad, Mohiuddin & Mitul, Abu & Tunku Mansur, Tunku Muhammad Nizar & Rashid, Mohd. (2012). Microcontroller based health care monitoring system using sensor network. 2012 7th International Conference on Electrical and Computer Engineering, ICECE 2012. 272-275. 10.1109/ICECE.2012.6471538.
6. Islam, Md & Rafi, Fida & Ahmad, Mohiuddin & Mitul, Abu & Tunku Mansur, Tunku Muhammad Nizar & Rashid, Mohd. (2012). Microcontroller based health care monitoring system using sensor network. 2012 7th International Conference on Electrical and Computer Engineering, ICECE 2012. 272-275. 10.1109/ICECE.2012.6471538.
7. Sarnaik, Samiksha & Kshirsagar, Sakshi & Jori, Sadanand. (2022). IoT Based Health Monitoring System.
8. Rao, Dr. (2022). IoT Based Health Monitoring System. International Journal of Engineering and Advanced Technology. 11. 138-143. 10.35940/ijeat.E3616.0611522.
9. Rajagopal, Sivakumar & Mehta, Sujal. (2022). An IoT-Based Health Monitoring System. ECS Transactions. 107. 13491-13504. 10.1149/10701.13491ecst.
10. Kamweru, Paul & Robinson, Owino & Mutava, Mutinda. (2020). Monitoring Temperature and Humidity using Arduino Nano and Module-DHT11 Sensor with Real Time DS3231 Data Logger and LCD Display. 9. 416-422.

11. Singh, Alka & Ponde, Poonam. (2022). Home Automation: IoT. 10.1007/978-3-030-92905-3_32.

12. Gouda, Dr & Begum, Syeda & M, Muskaan & Madhushree, & Bai, Jaya. (2022). Home Automation using Blynk. International Journal of Advanced Research in Science, Communication and Technology. 355-356. 10.48175/IJARSCT-5352.

13. Khan, Mohammad. (2021). IoT-Based Smart Health Monitoring System for COVID-19 Patients. Computational and Mathematical Methods in Medicine. 2021. 1-11. 10.1155/2021/8591036.

14. Taiwo, Olutosin & Ezugwu, Absalom. (2020). Smart healthcare support for remote patient monitoring during covid-19 quarantine. Informatics in Medicine Unlocked. 20. 10.1016/j.imu.2020.100428.

15. Mukherjee, S., Dhar, M., Ghosh, A.: Accelerometer based wireless gesture controlled robot for medical assistance using Arduino Lilypad. Int. J. Eng. Technol. Sci. Res. 5(3), 155–161. ISSN: 2394–3386(2018)

16. Datta, S., et. al.: ARDUINO based Water Level Detector for Visually Impaired people, Computational Science and Engineering, pp. 137–140. CRC Press, Taylor & Francis (2016). ISBN: 978-1-138-02983-5

17. Mukherjee, S., Ghosh, A., Sarkar, S.K.: Arduino based wireless heart-rate monitoring system with automatic SOS message and/or call facility using SIM900A GSM module. In: International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN), pp. 1–5 (2019). doi:https://doi.org/10.1109/ViTECoN. 2019.8899504.

18. Kumar, S., Sharma, K., Raj, G., Datta, D., Ghosh, A. (2022). Arduino and ESP32-CAM-Based Automatic Touchless Attendance System. In: Sikdar, B., Prasad Maity, S., Samanta, J., Roy, A. (eds) Proceedings of the 3rd International Conference on Communication, Devices and Computing. Lecture Notes in Electrical Engineering, vol 851. Springer, Singapore. https://doi.org/10.1007/978-981-16-9154-6_14

19. https://esphome.io/devices/nodemcu_esp32.html

20. https://blynk.io