

# Internet Connected e-Healthcare System with Live Video Monitoring using LWIP Stack and SJF Priority Scheduling



Deepika .N, M. Anand, K. Sudhaman

**Abstract:** Continuous monitoring of patients in critical condition becomes essential in medical field. This paper aims at developing an innovative IoT based remote healthcare monitoring system by using Free RTOS with priority scheduling based on SJF (Shortest Job First). Proposed system offers vital medical information and live video of a patient who is present in rural area. It can make easy and quick accessing of a patient by medical professional from remote areas. Also helps for detailed analysis and taking correct remedies to effectively cure that patient. Various health parameters such as patient's body temperature, heart rate, body position along with live video are been monitored through internet using this system. In Free RTOS used in programming, SJF based priority scheduling is chosen because of reduced average waiting and turnaround time of CPU on performing multiple tasks associated with the camera and other sensors. The internet connectivity part of program is built using LWIP TCP/IP stack in focus of limited microcontroller resource.

**Keywords:** e-Healthcare, Free RTOS, Shortest Job First, Priority scheduling, Video Monitoring, LWIP stack.

## I. INTRODUCTION

Health care monitoring of patients is one of the time taking methods at hospitals. For example, it requires a person to take data of health parameters consistently, and have to prepare and maintain the manual record of those parameters, which may result in fault or delaying response by doctors during disease determination.

## II. RELATED WORKS

Wireless system for health care monitoring has become primary concern among research group and also in medical industry over the recent decade. Several inventive works have been published as paper. Several fundamental parameters can be monitored with health tracker. For example, health parameters like heart rate, Bp and respiration rate could be identified by using several health sensors [1]. The coordinated systems for monitoring the patient were combined with electronic records of the patient. The drawback of an electronic record of the patient requires enormous time, and this problem is overcome by employing the task of unique detection which permits the direct comparison [2]. When the fingertip is placed on the heartbeat sensor, it gets to the skin, and the sensing is denoted by sparkling of LED at one side of the heartbeat sensor. Most of the Pulse sensor can be based on lay on the fingertip of the human body.

Manuscript published on November 30, 2019.

\* Correspondence Author

**Deepika. N\***, Research Scholar, Department of Electronics and Communication Engineering, Dr. M. G. R Educational and Research Institute University, Chennai-95

**M. Anand**, Professor, Department of Electronics and Communication Engineering, Dr. M. G. R Educational and Research Institute University, Chennai-95

**K. Sudhaman**, Associate professor, Department of Electronics and Communication Engineering, Dr. M. G. R Educational and Research Institute University, Chennai-95

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an [open access](https://creativecommons.org/licenses/by-nc-nd/4.0/) article under the CC-BY-NC-ND license <http://creativecommons.org/licenses/by-nc-nd/4.0/>

The health information sends to the healthcare unit based on mobile by using GSM communication. Here the heart rate sensor is interfaced with Arduino micro controller [3]. Key information rates and the measure of the information collected in an assortment of smart health care utilize cases are discussed.

At last, they created fix sort wearable imperative observing device that is incorporated with an essential microcontroller, several health sensors and with Bluetooth module [4].

An observing system was designed with the help microcontroller ATMEGA8L [5] along with the several health sensors. It offers a constant non-invasive method for BP measurement. It was developed along with the alert unit which in turn employed for alerting the medical professional. The system accuracy is found in a certain range. If the patient’s health parameter such as Heart rate or body temperature goes beyond the threshold range, the alert system will be activated.

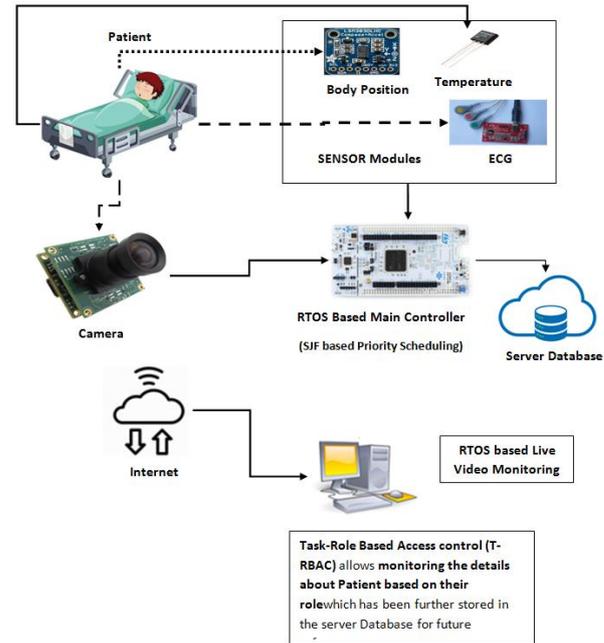
The scheduling policies of the CPU were discussed in [11]. Various issues were solved for finding the appropriate over the different systems. So, depending on the performance, the SJF algorithm is recommended for the problems associated with CPU scheduling for decreasing either average time for waiting or average turnaround time. And also, the FCFS algorithm is recommended for the difficulties associated with CPU scheduling for reducing either the average utilization of CPU.

The quick improvement of wireless technologies and the increases in communication transfer speed encourages the improvements of Android-based health care systems [6]. Remote and flexible patient observing systems not just expand the portability of patients and therapeutic staff additionally enhance the nature of health care [7]. In the improvements of remote patient monitoring frameworks, concentrated on patient’s details transmitting physiological signs utilizing the Zigbee innovation [8]. A few analysts have being used the individual personal digital assistant (PDA) to gain and send physiological symptoms from patient's space to the focal administration unit in the medicinal facility utilizing the remote advances using remote monitoring system [9-10]. ECG monitoring system is proposed using a new remote monitoring method which based on Internet of Things (IoT) in [12]. Here a wearable sensor collects patient’s ECG information and transmits to the cloud using WiFi internet. Doctors and patient’s relatives can use web application and mobile application for monitoring the status of patient according to their access restrictions.

**III. SYSTEM ARCHITECTURE AND WORKING PRINCIPLE**

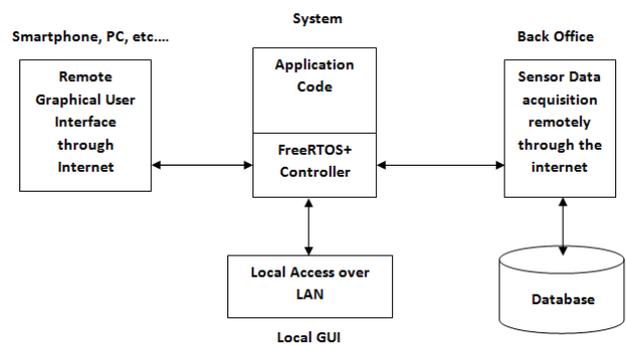
The primary aim of this paper is to develop the IoT based E-Health care system with live video monitoring by using RTOS with SJF based priority scheduling. This method is one of the pre-emptive based scheduling where it overcomes the drawbacks of the First Come First Serve (FCFS) based scheduling by reducing the average time for waiting for the processing of tasks and also average turnaround time of the functions to the CPU. This advanced system offers data of fundamental medical parameters along with the patient live

video monitoring with the less interrupt latency and with minimal thread switching latency for effective follow-up with the FreeRTOS. It will result in enhanced analysis and also the effective cure of that. Patient medical data are collected by using several medical sensors.



**Fig. 1 System Architecture**

The health parameters are collected from 3-channel ECG, Body temperature, and positions of the body which are further sent to the monitoring unit (Server unit) with the help of LwIP. The system uses an advanced camera module for live video monitoring of the patient. Figure 1 illustrates the whole process associated with the proposed method. AD8232 (ECG sensor) is used to observe the patient's regular heartbeat. By using this sensor, we can find whether the heart rate of the patient is normal or abnormal. MEMS accelerometer is used as body position sensor and it takes the three axis acceleration values of patient. LM35 is used for sensing body temperature.



**Fig. 2 Systematic representation of proposed system**

Figure 2 illustrates the systematic representation of a proposed system using FreeRTOS where the medical sensor data can be accessed either from local or remote.

Unique URL is given for every IoT device that wants to connect over the Internet and this type of technology permits authorized and very low bandwidth Peer-Peer connections. Data can also be accessed over a local network in case of no net connectivity. With the unique low power features, Free RTOS acts as a beneficial choice for IoT implementation.

**Sensor Signal Processing**

Sensor unit consists of 3 sensors such as an ECG sensor, body position and temperature sensor. In that, ECG and temperature sensor are analog output sensors and body position sensor is a digital output sensor. Digital output is measured using Inter Integrated Circuit (IIC or I2C) peripheral and analog outputs are processed using analog to digital converter peripheral available in STM32F429 microcontroller. This microcontroller uses Successive approximation type ADC which is popular and widely used. It takes less time for conversion irrespective of the magnitude of analog input voltage. Also the conversion time is maintained as constant.

SA-ADC consists of a Successive Approximation Register (SAR) to produce digital value equivalent to the analog input, Digital to Analog Converter (DAC) to produce analog voltage equivalent to the digital value received from SAR, sample and hold circuit to receive the analog input and a comparator to compare the analog input voltage and the voltage generated by DAC. The SAR generates an N bit digital value by changing one bit at a time, starts from MSB.

ECG signals are collected from patient using 3 electrodes at points such as right arm, left arm and right leg. The final output of the sensor is single analog voltage and it is given to one of ADC channel in microcontroller. This voltage is sampled at frequency of 150 Hz and sampled ECG values are packed as 300 bytes per packet and send to the internet client upon request. From the ECG data the heart rate of patient is calculated by counting the number of peaks in a particular time and it is converted to heart rate per minute.

Temperature sensor LM35 is used to measure the patient body temperature. It produces 10 mv output to 1 deg rise in temperature. So if temperature is 10 deg, the sensor output will be 100 mv. To get the temperature, the analog voltage measured using ADC is divided by 10.

The MEMS accelerometer LIS302DL is used to measure the body position of the patient. It measures acceleration parameters and microcontroller reads the values using I2C peripheral. The values are compared with thresholds to conclude the position of patient such as supine, right, left or sitting. Heart rate, body temperature and body position are packed as a separate packet and send to internet client upon request.

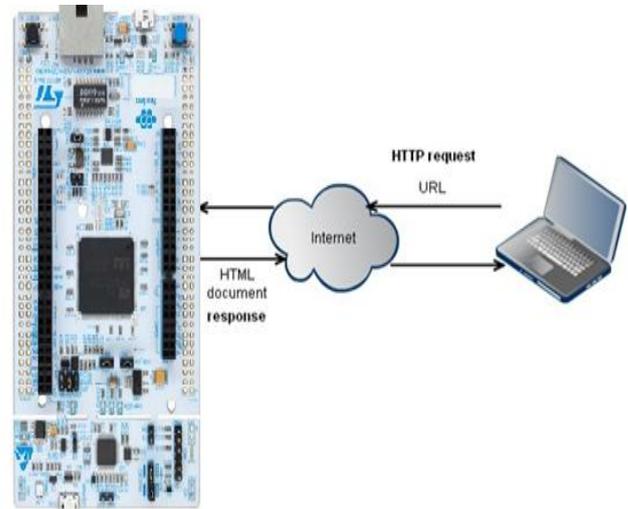
**Camera Image Acquisition using DCMI**

A project of this sort needs an extremely fit microcontroller with an extensive measure of RAM. Along these lines, STM32F429 from STMicroelectronics is picked as fundamental MCU, which is one of a powerful microcontroller right now accessible in the market. The camera is fixed with the system for streaming a live video feed. Whenever the request is made RTOS based microcontroller acquires the pictures which are of JPEG format from camera by using built-in peripheral (DCMI), and it starts to stream it in the web in MJPEG format and

plays at sufficient rate. The resolution of the picture is at 470 x 272. The STM32 controller has a huge RAM about 256KB, which is an absolute necessity for this kind of application.

**Embedded Web Server**

ARM Cortex-M4 based microcontroller is used in this project. It acts as a source for the creation of a web server by using the LwIP protocol. The information about the patient medical parameters along with the live video feed can be viewed by any device with internet connectivity. Due to access control mechanism, the person is authorized to access the monitoring server only if they login with correct username and password. Figure 3 illustrates the communication between client and server. The primary function of web server is to serve the pages, process request of client and for continuing TCP/IP connection until the session is closed by client. The language used for developing web page is HTML. The device uses Lightweight IP (LwIP) TCP/IP standard stack for its web interface.



**Fig. 3 Client and Server Communication**

**IV. SOFTWARE DESIGN**

**RTOS Architecture**

RTOS is a multitasking system employed for applications that are running continuously. The RTOS architecture is dependent on versatile quality of its association. RTOS is a self adaptable system to meet different need of arrangements for various applications. For smaller applications, RTOS usually contains kernel alone. For the embedded frameworks with more complexity, RTOS can be merge with multiple modules, which includes a core, convention stacks and various other sections associated with it. A working system of RTOS contains two parts: kernel space and user space. Kernel in RTOS is a smallest and essential segment of working framework. Its control incorporates devices and its memory supervision and additionally, it provides an interface to utilize the resources by programming the applications.



And also services like program protection management and multitasking get included based on OS architecture.

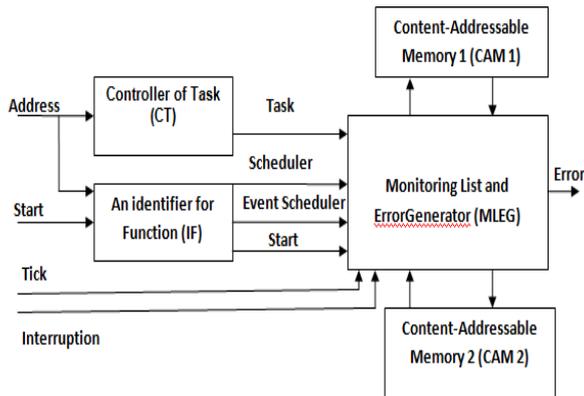


Fig. 4 Functional block diagram of RTOS

Abstraction layer between programming application and equipment is provided by RTOS kernel. To monitor signals like Tick, Start, Interrupt as well as addresses of RAM was accessed for the period of execution of application code; Free RTOS is connected to the system’s bus of microcontroller. Figure 5 shows functional block diagram of RTOS. The RTOS consists of five functional blocks.

The Controller of Task (CT) recognizes the task in execution depending on address where microcontroller was accessed during implementation of application. CT evaluates bus address with the associated task addresses during each clock cycle. If the address located is task related then equivalent task’s number is received by Task signal. To ensure the process of scheduling in order of executions, an analysis has been carried out by Identifier for Function (IF) on the functions implemented through a process of task scheduling. As a final point, an event which triggered the scheduling will be identified by IF based on that order. The Task in execution and signal for an event scheduler was received by Monitoring of List and Error Generator (MLEG) block. Depending on this data the MLEG categorize every task in two different ones, tasks at ready state and tasks at blocked state, each one is arranged with their priority and its state. Misbehaviour in scheduling will be identified by using MLEG block. Final two blocks are Content-Addressable Memory 1(CAM1), and Content-Addressable Memory2 (CAM2) keeps the lists created by the MLEG module. The tasks at ready state get saved in CAM1; on the other hand, the tasks at blocked state are stored in CAM2.

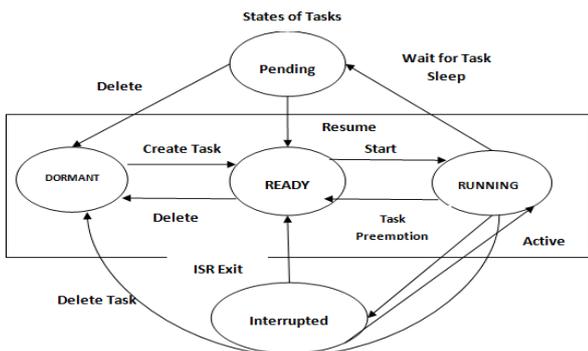


Fig. 5 Possible States Transition of Tasks

Management of Task

The application is divided into simple, successive, and schedulable units of a program called "Task" for achieving continuity in real-time application program. The task is a fundamental unit of implementation in real-time perspective, and it is managed by 3 time properties such as deadline, time of release, and time of execution. The timeframe is defined as a time limit where task must be completed. Time of release defined as a time where the function can be start executed and finally the time of execution defined as a time at which the task to get accomplished. Every task may be present in one of the four states, which includes ready, running, dormant and blocked as illustrated in Figure 5.

When an application program is under execution, specific tasks are changing continuously from one form of state to another. But in every case, only a single function can be in running mode during the execution. In the process control of CPU, is varying from one task to another in a specific controlling process. The task to be suspended or will be saved from that situation while background tasks which have to be executed will be retrieved. Context switching is a process of keeping a function to the background being in queue and resuming those tasks.

Pre-emptive Scheduling

For implementing pre-emption scheduling, the algorithm with support priority contains a list of every task at ready state. The tasks are getting sorted based on their preference. So, whenever a scheduling event occurs, the task at ready state with uppermost priority gets executed. Each task condition can keep traced by using scheduler, and it chooses the task among them which are in ready state to be performed, and CPU is allocated to one of the task determined by scheduler. It assists the CPU to maximize its utilization over a variety of functions in a program of multi-tasking, and it also offers reduction in time for waiting.

Figure 6 shows the model of pre-emptive scheduling. Pre-emptive scheduling based on priority needs a processor control which is given to an all-time highest priority task. In an incident which makes ready to run a job with top priority, the present task is suspended immediately, and processor control is offered to the task with higher priority.

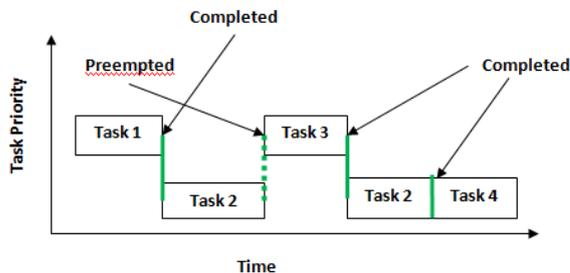


Fig. 6 Pre-emptive Scheduling

SJF based priority scheduling is a pre-emptive based scheduling algorithm where every thread or process which are having same priority is executed based on burst time.



This implies that procedures that are having least burst time will get performed first. This type of priority algorithm results in a reduction of average time for waiting and time for turnaround process. For calculating this, the following formulas are used:

$Turnaround\ Time = Completion\ time\ of\ Process - Time\ of\ Arrival.$

$Time\ of\ Waiting = Turnaround\ Time - Burst\ Time$

The following steps will be followed in this algorithm,

- The process is assigned to a ready queue.
- Process with high priority will obtain the CPU first when compared with a lower priority process.
- If similar priority exists between two processes, then SJF is employed to break the bind.
- The above steps will get repeated until the ready queue becomes empty.

### V. RESULTS AND DISCUSSION

Overall hardware implementation is shown in Figure 7. ECG sensor, body temperature sensor, 3 axis MEMS accelerometer, camera and a buzzer are connected with the microcontroller ports. The system is connected to internet modem through an Ethernet cable for accessing from anywhere. The system is powered from AC source through a 9V DC adapter. Free RTOS with pre-emptive scheduling is used in microcontroller programming. In RTOS, pre-emption is achieved by assigning the suitable level of priority to individual tasks.

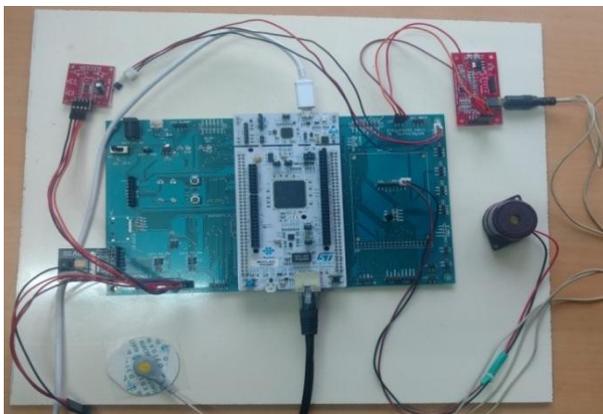


Fig. 7 Hardware Implementation

Priority to tasks can be assigned either by First Come First Serve (FCFS) basis or by Shortest Job First (SJF). Table 1 shows burst time for ten tasks along with their priorities assigned based on SJF. Gantt chart for the tasks in Table 1 is illustrated in Figure 8. Table 2 shows waiting time and turnaround time of tasks which are scheduled based on FCFS method. Similarly Table 3 shows waiting time and turnaround time of tasks which are scheduled based SJF method. Figure 9 shows graphical comparison between waiting time of FCFS and SJF based priority scheduling. From Table 2, Table 3 and graph we can see that the average waiting time and turnaround time are less in SJF based priority scheduling compared to FCFS method. Thus SJF based priority scheduling is used here in RTOS section of microcontroller programming. In our proposed priority

scheduling, if more than one task with a similar priority comes, then the task with low burst time gets executed first.

Table. 1 Burst time with Priorities for ten tasks

Tasks	Burst Time	Priorities
T1	160	2
T2	95	1
T3	175	3
T4	59	1
T5	180	4
T6	190	5
T7	200	6
T8	184	5
T9	195	6
T10	210	7

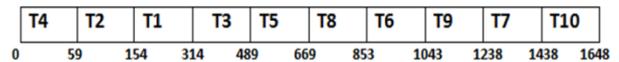


Fig. 8 Gantt chart for tasks mentioned in Table 1

Table. 2 Priority scheduling based on FCFS for 10 Tasks

Tasks	Burst time	Waiting Time(ms)	Turnaround time (ms)
T1	160	0	160
T2	95	160	255
T3	175	255	430
T4	59	430	489
T5	180	489	669
T6	190	669	859
T7	200	859	1059
T8	184	1059	1243
T9	195	1243	1438
T10	210	1438	1648
Average		660.2	825

Table. 3 Priority scheduling based on SJF for 10 Tasks

Tasks	Burst time	Priority	Waiting Time(ms)	Turnaround time (ms)
T4	59	1	0	59
T2	95	1	59	154
T1	160	2	154	314
T3	175	3	314	489
T5	180	4	489	669
T8	184	5	669	853
T6	190	5	853	1043
T9	195	6	1043	1238
T7	200	6	1238	1438
T10	210	7	1438	1648
Average			625.7	790.5

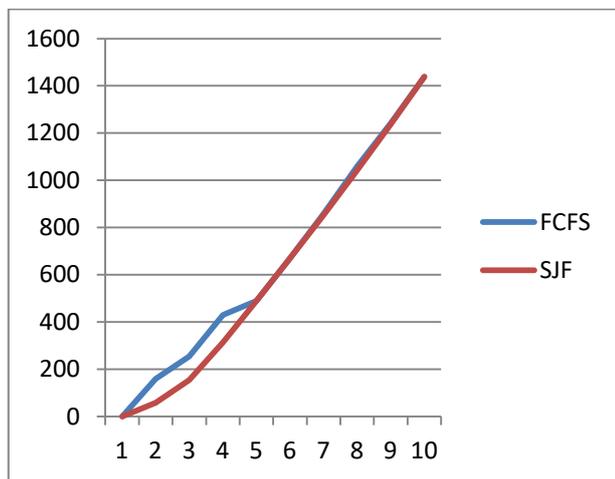


Fig. 9 waiting time of FCFS and SJF based priority scheduling

The entire microcontroller program is developed using a tool named Atollic True Studio IDE which is from ST Microelectronics. Any web browser can be used to see patient’s health information along with the live video feed. Data from embedded web server are sent as a packet to web client. There is no restriction on length of the data to be sent, because LWIP stack takes care of this. Web client receives as much as packets sending by server as shown in Figure 10.

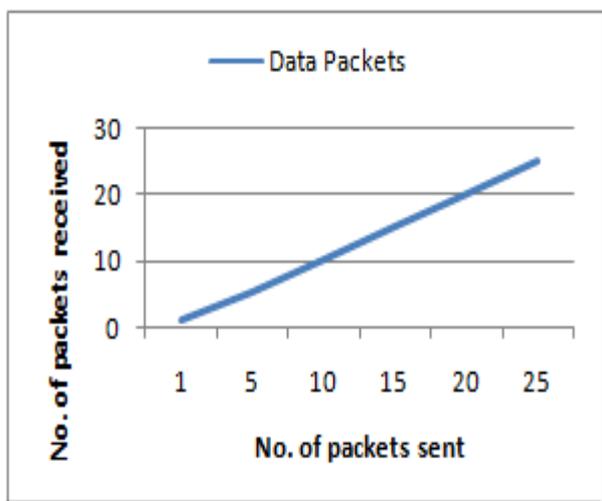


Fig. 10 Data packets sent & received

Authorized persons are identified from login information and are allowed to view the biological information. After login into the webpage, doctor can view the patient's health records as shown in Figure 11. The system also alerts the nearby caretakers in case of abnormal heart rate or body temperature by triggering and making sound using a buzzer. The web page can be changed to live streaming page as shown in Figure 12 to see the video capturing by camera which is near patient. The doctor’s can have full access over the information about the patient’s health parameters with the live video feed. It helps to monitor the physical activity of patient along with medical information. Doctors have right to view the information and give instruction for further improvement of patient’s health.

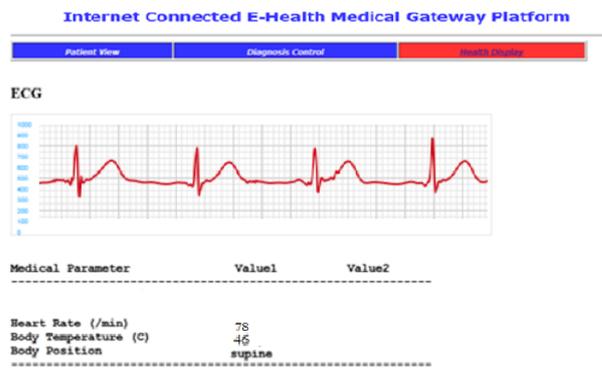


Fig. 11 Real Time Patient Health Status Monitoring



Fig. 12 Video Monitoring

VI. CONCLUSION

We developed a Free RTOS based Health monitoring system and live video monitoring using IoT techniques. The design of the health monitoring system was explained at first. Typical health sensing networks employ Lightweight IP (LwIP) which was introduced, and its relation with the Free RTOS based on the SJF based priority algorithm was explained. The whole IOT based health monitoring system was developed depending on the system architecture. By using a wearable monitoring node with a temperature sensor, body position sensor and ECG sensor with three electrodes, ECG signals and other sensor data are collected with acceptable precision. The high precision camera will show the useful live video feed on the dedicated webpage.

Analysis for the ten tasks is done by using the Free RTOS with SJF based priority scheduling average time taken for waiting of the tasks to get processed were reduced and also average turnaround time is received by the system to give the desired response to the functions get reduced. The obtained data were transmitted to the IoT cloud using LwIP protocol which helps widen coverage areas, and the data rates will be high. The collected data are stored in the database. We can access the data either from local or global using internet, where its implementation depends on the servers, i.e., the HTTP server, and storage server. Thus, the

proposed system makes the human's daily life easier and more comfortable.

## REFERENCE

1. Teaw, E., Hou, G., Gouzman, M., Tang, K.W., Kesluk, A., Kane, M. and Farrell, J., 2005. A wireless health monitoring system. In Information Acquisition, 2005 IEEE International Conference on (pp. 6-pp). IEEE.
2. Turcu, C.E. and Turcu, C.O., 2013. Internet of things as the key enabler for sustainable healthcare delivery. *Procedia-Social and Behavioral Sciences*, 73, pp.251-256.
3. Goyal, D., Bhaskar, J. and Singh, P., 2012, December. Designing the low-cost patient monitoring device (LCPMD) & Ubiquitous based Remote Health Monitoring and Health Management System using Tablet PC. In Parallel Distributed and Grid Computing (PDGC), 2012 2nd IEEE International Conference on (pp. 7-11). IEEE.
4. Chung, W.Y., Lee, Y.D. and Jung, S.J., 2008, August. A wireless sensor network compatible wearable u-healthcare monitoring system using integrated ECG, accelerometer and SpO<sub>2</sub>. In Engineering in Medicine and Biology Society, 2008. EMBS 2008. 30th Annual International Conference of the IEEE (pp. 1529-1532). IEEE.
5. Islam, M.M., Rafi, F.H.M., Ahmad, M., Mitul, A.F., Mansur, T.T. and Rashid, M.A., 2012, December. Microcontroller based health care monitoring system using a sensor network. In Electrical & Computer Engineering (ICECE), 2012 7th International Conference on (pp. 272-275). IEEE.
6. Wagner, M., Kuch, B., Cabrera, C., Enoksson, P. and Sieber, A., 2012, July. Android-based body area network for the evaluation of medical parameters. In Intelligent Solutions in Embedded Systems (WISES), 2012 Proceedings of the Tenth Workshop on (pp. 33-38). IEEE.
7. Pandian, P.S., Mohanavelu, K., Safeer, K.P., Kotresh, T.M., Shakunthala, D.T., Gopal, P. and Padaki, V.C., 2008. Smart Vest: Wearable multi-parameter remote physiological monitoring system. *Medical engineering & physics*, 30(4), pp.466-477.
8. Yan, Z. and Jiaying, Z., 2010, May. Design and implementation of Zigbee based wireless sensor network for remote SpO<sub>2</sub> monitor. In Future Computer and Communication (ICFCC), 2010 2nd International Conference on (Vol. 2, pp. V2-278). IEEE.
9. Yuan, B. and Herbert, J., 2011, March. Web-based real-time remote monitoring for pervasive healthcare. In Pervasive Computing and Communications Workshops (PERCOM Workshops), 2011 IEEE International Conference on (pp. 625-629). IEEE.
10. Sain, M., Lee, H. and Chung, W.Y., 2009, August. Private healthcare information system. In INC, IMS and IDC, 2009. NCM'09. Fifth International Joint Conference on (pp. 1540-1545). IEEE.
11. Jyotirmay Patel and A.K.Solanki, "CPU Scheduling: A Comparative Study", Proceedings of the 5th National Conference, ISSN 0973-7529, ISBN 978-93-80544-00-7, 2011.
12. Yang, Z., Zhou, Q., Lei, L., Zheng, K. and Xiang, W., 2016. An IoT-cloud based wearable ECG monitoring system for smart healthcare. *Journal of medical systems*, 40(12), p.286.