IJETST- Vol.||03||Issue||11||Pages 4742-4746||November||ISSN 2348-9480 2016

Open access Journal **International Journal of Emerging Trends in Science and Technology**
Impact Factor: 2.838 **DOI:** https://dx.doi.org/10.18535/ijetst/v3i11.03

A Framework for Medical Assistance using Internet of Things Architecture

Authors
Akurathi Hema Latha[#1], **Ammi Reddy Pulagam**[*2]
[#]PG Student, [*]Associate Professor, Department of Electronics and Communication Engineering,
Vasireddy Venkatadri Institute of Technology, Namburu, Guntur (dt.), A. P. India
Email: [1]*hema2692@ gmail.com*, [2]*pulagamammireddy@gmail.com*

ABSTRACT
Remote monitoring of patient is one of the important areas of research in the medical field. With the advancements in the field of sensors and semiconductors it is possible to monitor a patient and provide him medication. The network of sensors, actuators and other communication devices is called Internet of Things (IoT). This technology has revolutionized the health care industry by providing connected health technologies. With the devices connected, we can have a faster data transfer and easy access to the data. This paper proposes a framework for medical assistance using Internet of Things. This IoT based system collects vital parameters from the patient and stores this information in a server which can then be accessed by the doctor to provide medicines. Parameters such as temperature, humidity, heart rate, and blood pressure are gathered and posted into the cloud. ESP8266 Wi-Fi module has been used as a communication device. The results show that the system developed is working in par with the conventional monitoring systems.
Keywords-*ESP8266, Internet of Things, Network of Sensors, Remote Monitoring.*

INTRODUCTION

The Internet-of-Things is proving to be a technology that has great impact and potentiality to influence the standards of human living and affect many different fields such as safety, security, health and medicine. With the availability of accurate sensors and actuators, almost all parameters can be captured, processed and analyzed using advanced processors. By using Internet of Things technology, physical objects embedded with sensors, actuators are connected to networks through communication devices for data transportation.

Patient's quality of life can be improved using remote monitoring. Instead of moving the patient from one location to the other, the patient can stay at his own facility. Sensors and other devices attached to his body gather information about vital parameters such as temperature, humidity, blood pressure, heart beat rate etc. and this information is sent to the doctor and is aggregated. The doctor receives this information, analyses and finally gives proper medication to the patient and he can give timely response to the care takers of the patient depending upon the information. Fig. 1 shows the scenario used in remote monitoring of the patient.

EXISTING SYSTEMS

Chung and his group [1] proposed WSN-based mobile healthcare monitoring system with ECG and blood pressure measurement, where the mobile phone performs continuous data analysis and then transmits data over a wireless sensor network. In today's critical care environment, regardless of your age, disease or condition, it is very likely that you will receive the same type of patient monitoring. This traditional monitoring can include ECG leads that record cardiac rhythm and heart rate and SpO2 probes that capture

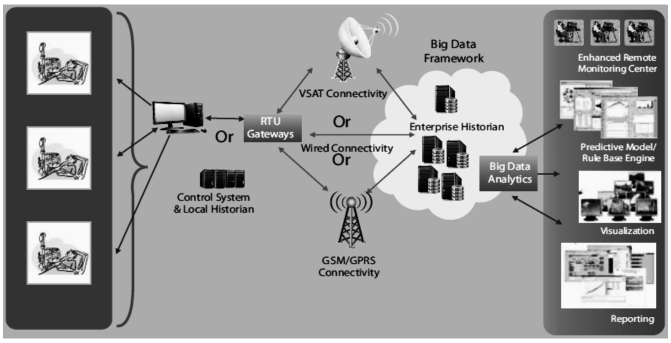

Fig. 1 Remote Monitoring of Patient

Blood, oxygen saturation levels along with other vital sign measurements [2,3]. This is mainly because these general parameters are the bone for the physician to know what to do in general. However, the fast change of health parameters is the big challenges, especially when the patient is outside home doing any usual activity and feels something wrong, and when arrives the hospital for testing, they find that everything is normal. In fact, it is not easy to detect all kinds of abnormal activity unless real-time monitoring, which can be done either by keeping the patient in the hospital for few days or more (which will lead to high costs). In such a case, a wireless real-time portable monitoring device can be used to help the physician and the medical centre to give proper medical treatment and procedures. It is so important to integrate low-power electronic devices, such as sensors and a microcontroller, with wireless communication technology to open new research trends in healthcare applications. Most of the e-health monitoring systems are offline units based on personal computers or smart phones used to send patient's health data to the health centres.

This paper presents an architectural framework for a real-time cloud-based health monitoring system. It uses Wi-Fi technology to connect to the internet. Sensors, both analog and digital, are used in the design. The proposed architecture of the system developed is shown in Fig. 2.

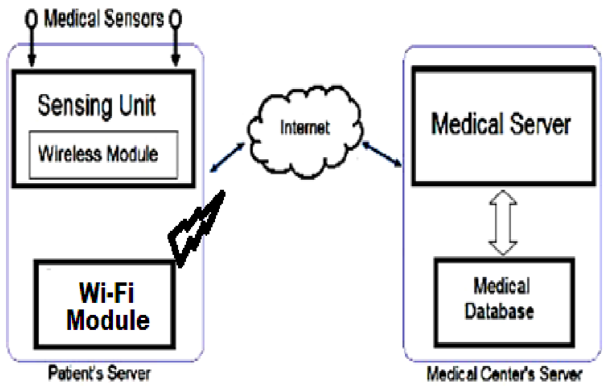

Fig. 2 Architecture of the proposed Medical System

PROPOSED SYSTEM

The proposed system layout is shown in Fig. 3. A single chip microcontroller We Mos D1 Mini is used as the embedded sensing unit. The sensors are then connected to the microcontroller. The microcontroller contains a built-in Wi-Fi module (ESP8266) which operates in station mode. Using this module, the system connects to the internet and posts the data into a cloud server. This data can be accessed by the authorized doctor, who can then analyze and prescribe the medicines. Developing an application for smart phone can make the data available on doctor's phone. The data can then be stored in the hospital server for future needs.

WEMOS D1 Mini

Fig. 4 shows the We Mos D1 Mini microcontroller [4]. This is an ESP8266-12 based WiFi enabled microprocessor unit on a Arduino-UNO footprint. That means the board looks and works (in most cases) like an UNO. Apparently several shields, sensors and output devices that are manufactured for the Arduino platform will work on the WeMos-D1R2 with the added advantage of built-in Wi-Fi. Though the Arduino UNO and the WeMos-D1R2 are similar, there are a few differences in their pin assignment.

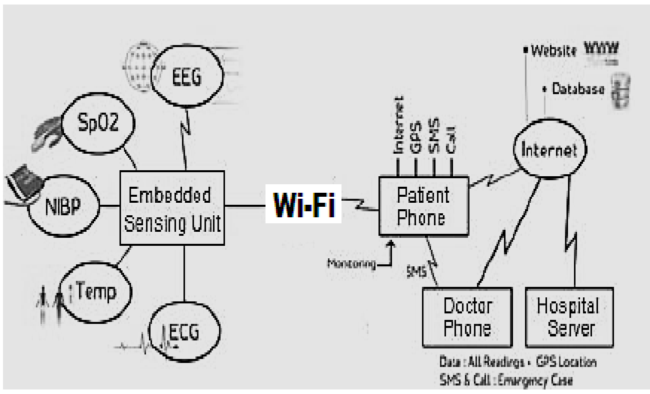

Figure 3 System Layout

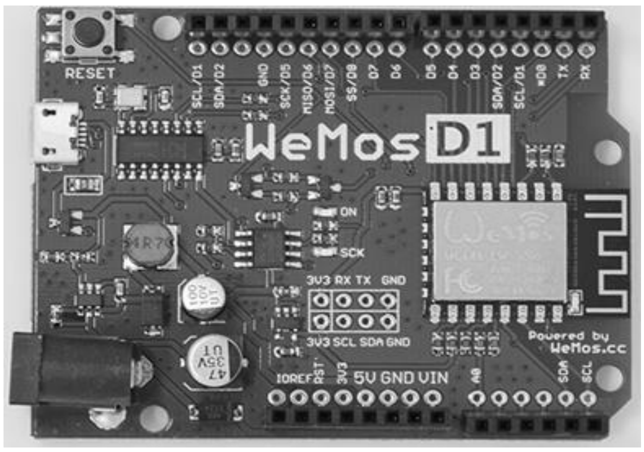

Fig. 4 We Mos D1 Microcontroller

ESP8266 Wi-Fi Module

The ESP8266 [5] is a low-cost Wi-Fi chip with full TCP/IP stack and MCU (Micro Controller Unit) capability produced by Shanghai-based Chinese manufacturer, Espressif Systems. It is a 32-bit RISC CPU Tensilica Xtensa LX106 running at 80 MHz. It has a 64 KB of instruction RAM, 96 KB of data RAM, External QSPI flash of 4 MB. It supports IEEE 802.11 b/g/n Wi-Fi and contains 16 GPIO pins along with support for SPI, I²C, I²S interfaces with DMA (sharing pins with GPIO)UART on dedicated pins, plus a transmit-only and UART can be enabled on GPIO2. It also has a 10-bit ADC built-in. Both the CPU and flash clock speeds can be doubled by over clocking on some devices. CPU can be run at 160 MHz and flash can be sped up from 40 MHz to 80 MHz.

EMG Sensor

Fig. 5 Shows EMG sensor from Advancer Technologies [6]. This sensor will measure the filtered and rectified electrical activity of a muscle; outputting 0-V_S Volts depending the amount of activity in the selected muscle, where V_S signifies the voltage of the power source. Power supply voltage: min. +-3.5V.

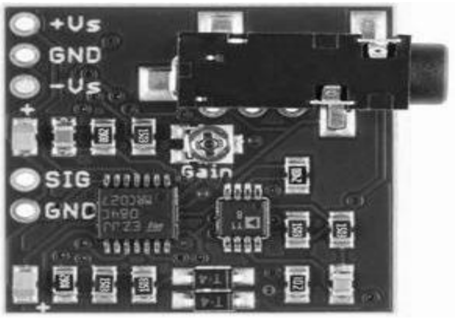

Fig. 5 EMG Sensor

ECG module AD8232

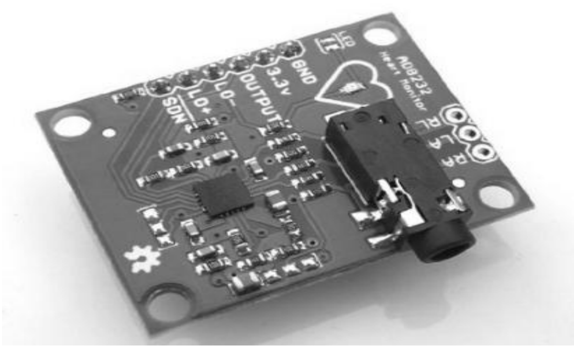

Fig. 6 ECG Sensor

Fig. 6 shows the ECG Sensor. The AD8232 [7] is a neat little chip used to measure the electrical activity of the heart. This electrical activity can be charted as an ECG or Electrocardiog-ram. Electrocardiography is used to help diagnose various heart conditions. Fig. 7 shows the ECG waveform. The QRS complex is a name for the combination of three of the graphical deflections seen on a typical electrocardiogram (EKG or ECG). It is usually the central and most visually obvious part of the tracing. It corresponds to the depolarization of the right and left ventricles of the human heart. In adults, it normally lasts 0.06–0.10 s; in children and during physical activity, it may be shorter. The Q, R, and S waves occur in rapid succession, do not all appear in all leads, and reflect a single event, and thus are usually considered together. A Q wave is any downward deflection after the P wave. An R wave follows as an upward deflection, and the S wave is any downward deflection afterthe R wave. The T wave follows the S wave, and in some cases an additional U wave follows the T wave.

Blood Pressure Sensor

The Honeywell ASDX 015PDAA5 pressure transducer has been used to measure the blood pressure. It offers ±2.0% accuracy and consumes 2mA typically. The pressure transducer is connected to a DC motor and a dump valve through an air pipe. The dump valve is directly connected to the controller, while the DC motor is connected to the controller through 2N3094 transistor. The program flow is shown Fig. 8.

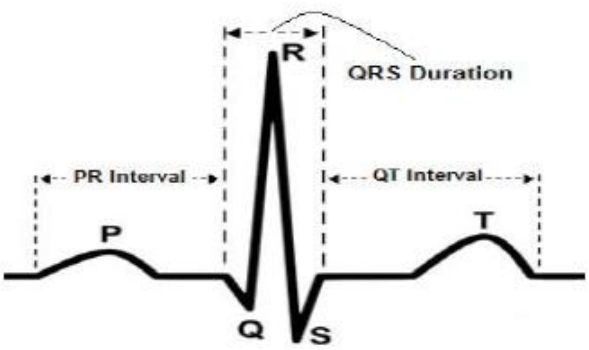

Fig. 7 Main Characteristics of ECG

IJETST- Vol.||03||Issue||11||Pages 4742-4746||November||ISSN 2348-9480 2016

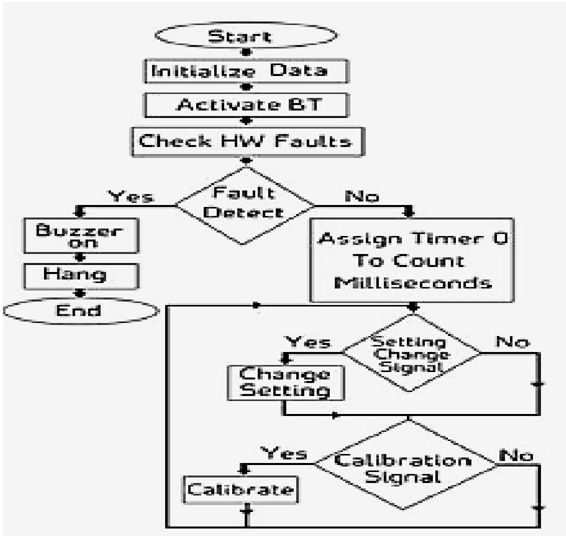

Fig. 8 Main Program Flow

As we can see in the flow chart, the initialization process will activate all the sensors on board and then will make the processor ready to take the data. Once data is collected from the sensors, the Wi-Fi module will configure itself to station mode and then connects to internet through any Access Point (AP). Then it establishes a stream to the cloud and starts streaming data.

RESULTS

The software is developed using Arduino IDE. The http://data.sparkfun.com [8] provides a free cloud service for data upto 50 MB. We have created a data stream to post data into the cloud. Public key and private keys are to be provided to store data in the server. In this way, security is provided for the data stream. The packet format is given below

http://data.sparkfun.com/input/[publicKey]?private_key=[privateKey]&blood_pressure=[value]&heart_beat_bpm=[value]&humidity=[value]&muscle_activity=[value]&temp_centigrade=[value]

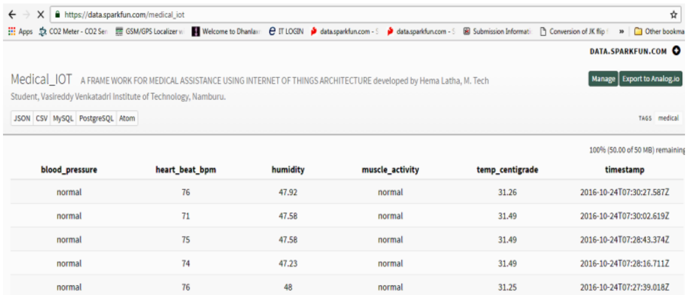

Fig. 9. Screenshot of the data posted at https://data.sparkfun.com/medical_iot

The data stored in the sparkfun server is shown in Fig. 9. This data can be accessed by the doctor from anywhere and can then provide the proper medication to the patient. We can hide the stream from the public domain by selecting hidden stream during the stream creation. In order to modify or delete data from this server we need private key. Since private key is not known to others, our data is secured.

CONCLUSION

We have designed a framework for medical system that monitors the critical parameters of the patient in real-time. The doctors can access this data from anywhere and can provide proper medication to the patients. The system is accurate and very easy to use. The sensors used are eco-friendly and will not cause any harmful effects to the patients wearing them. The system can be further extended by connecting medical stores thereby doctor's prescription and patient's address can sent to the stores and allowing medicines delivered at patient's premises.

REFERENCES

1. Pei-Cheng Hii, & Wan-Young Chung, *A Comprehensive Ubiquitous Healthcare Solution on an Android Mobile Device*, Sensors, Vol.11, No.7, 2011, pp: 6799-6815.
2. M. D. Sarmiento, P. Zhibo, M. F. Sanchez, C. Qiang, H. Tenhunen, & Z. Li-Rong, *Mobile wireless sensor system for tracking and environmental supervision*, IEEE International Symposium on Industrial Electronics (ISIE), 2010, pp: 470-477.
3. S. A. Haque, S. M. Aziz, & M. Rahman, *Review of Cyber-Physical System in Healthcare*, Intr. Journal of Distributed Sensor Networks, Vol. 2014, Article ID:217415, 20 pages.
4. Wemos D1 Rev 2 ESP8266 built-in, *www.wemos.cc/Products/d1_r2*
5. ESP8266 Wi-Fi Module, *https://en.wikipedia.org/wiki/ESP8266*
6. Advancer Technologies Inc., Muscle Sensor Kit V3. *www.advancer.com/musclesensorV3*

7. AD8232 ECG Sensor, Low Power and High Efficient *www.analog.com/en/products/application-specific /medical / ecg/ad8232.html*
8. Sparkfun Free Cloud for IoT Development *http://data.sparkfun.com*
9. *8051 Microcontroller* – Kenneth J Ayala
10. *The 8051 Microcontroller Applications and Interfac*ing – Mazidi & Mazidi