IoT Based Real-Time Remote Patient Monitoring System

Conference Paper · February 2020

DOI: 10.1109/CSPA48992.2020.9068699

2020 16th IEEE International Colloquium on Signal Processing & its Applications (CSPA 2020), 28-29 Feb. 2020, Langkawi, Malaysia

IoT Based Real-Time Remote Patient Monitoring System

Hoe Tung Yew
Universiti Malaysia Sabah
Kota Kinabalu, Sabah
htyew@ums.edu.my

Ming Fung Ng
Universiti Malaysia Sabah
Kota Kinabalu, Sabah
mingfung93@gmail.com

Soh Zhi Ping
Universiti Malaysia Sabah
Kota Kinabalu, Sabah
sohzhiping95@hotmail.com

Seng Kheau Chung
Universiti Malaysia Sabah
Kota Kinabalu, Sabah
kheau@ums.edu.my

Ali Chekima
Universiti Malaysia Sabah
Kota Kinabalu, Sabah
chekima@ums.edu.my

Jamal A. Dargham
Universiti Malaysia Sabah
Kota Kinabalu, Sabah
jamalad@ums.edu.my

***Abstract*—Healthcare technology is one of the most popular studies nowadays. With the development of healthcare technology, the lifespan of people has successfully extended. However, people in the rural area are still having a hard time to obtain professional healthcare services due to the barrier of distance and lack of doctors. A remote patient monitoring system is one of the best solutions to overcome this issue. This paper proposes an Internet of Things (IoT) based real-time remote patient monitoring system that is able to guarantee the integrity of the real-time electrocardiogram (ECG). Message Queuing Telemetry Transport (MQTT) protocol is used for transmitting the real-time ECG from the proposed system to the webserver. The doctor can access the webserver via smartphone or computer to monitor the real-time or previously recorded ECG data. The proposed system has been tested in both Local Area Network and Wide Area Network environments. The results show that the proposed system has no package loss and packet error in both Local Area Network and Wide Area Network.**

Keywords—: electrocardiogram, real-time monitoring, MQTT protocol, remote patient monitoring system.

I. INTRODUCTION

The ratio of doctors to the population in Malaysia is improving from year to year. However, it is still lacking doctor in rural areas. The report from the Ministry of Rural Development Malaysia showed that 24.5% of the population are staying in rural areas. The majority of them are from East Malaysia and the East Coast of Peninsular Malaysia [1]. According to the statistic from the Ministry of Health Malaysia, the ratio of doctor to the population in west coast of Peninsular Malaysia, east coast of Peninsular Malaysia and East Malaysia is 1:482, 1:730 and 1:786, respectively [1]. Obviously, the number of doctors in East Malaysia and the east coast of Peninsular is much lower compared to the west coast of Peninsular Malaysia.

A remote patient monitoring system is one of the solutions to overcome the barriers of health service. It provides professional health care services to the rural population through information and communication technologies (ICT). Patel et al. stated that the continuation of health and wellness monitoring can help in the treatment of the patient [2]. Remote patient monitoring system allows the doctor to observe the patient's health condition remotely at any time and any place [3]. The prevention can be done during the early detection of the disorder.

This research is supported by Ministry of Education Malaysia and Universiti Malaysia Sabah, Geran Penyelidikan UMS (SBK0364-2017).

This paper proposes an IoT based real-time electrocardiogram (ECG) monitoring system that is able to support both real-time and store-and-forward modes. The store-and-forward mode has no data integrity issue and it also does not require high network quality. However, the real-time mode is challenging because transmission delay and packet loss during real-time transmission will affect the data integrity, especially for real-time ECG signal transmission. Inaccurate health data might lead to misdiagnosis.

II. RELATED WORKS

A WLAN based remote health monitoring system with end to end flow had been presented in [4]. The system collects the patient vital sign such as heart rate, blood pressure, pulse oximetry and body temperature by using medical sensors. Real-time ECG signal was not included in this work. Arduino Yun is used as the main controller to process and analyze the data. The data is then uploaded to the cloud using the Application Programming Interface (API) through WLAN. Doctors can log in to the online server for checking the patient medical data and send comments to the patient through the server. If the device lost the WLAN connection, the data will be temporarily stored in the memory card. The system will synchronize the data with the server once the connection with the server is recovered.

Lee et al. proposed a smartphone based remote patient monitoring system [5]. In this system, a smartphone application (app) is implemented. Patient can have a video conference with the doctor by using a smartphone to discuss the treatment process. This system is collaborating with Skype's AES-style encryption to secure the patient's video data. This system does not support real-time mode. Patient is required to upload the collected vital sign to the server manually.

Roy et al. presented an RF based remote patient monitoring system which enables the user to use medical care information through web and mobile application platforms [6]. The system is built up with sensor nodes, coordinator node, web and database server and graphical user interface (GUI). The sensor node is used for collecting the data. The acquired data is uploaded to the central server. Users can observe the data and analyse result through the GUI. Due to the delay in transmission and slow server response, this system is unable to support real-time mode.

An IoT based health care system for doctors remotely monitoring patient's health conditions via the internet had been presented by Ghosh et al. [7]. E-Health Sensor Shield Kit was used in this module for collecting the biometric information and then transmitting to the online server through wireless technology. The data collected by the sensor was temporarily stored in the microcontroller and uploaded to the online database after some time. The user is required to log in with unique ID for accessing the data. The total time required by the system from collecting health data, transmitting data to the webserver and visualizing the data on the webpage is around 15 seconds.

Mustapha et al. proposed a patient remote monitoring system using Web Real-Time Communication (WebRTC) and Edge Cloud [8]. Edge Cloud is not only the storage but it is also used as a communication platform between the patients and remote patient monitoring system. The system has two modes: push–mode and pull-mode. In push-mode remote monitoring, the system sends alert to the user if abnormal health data is detected. While pull-mode remote monitoring is for the user to review back the previously collected data. WebRTC supports multiple ways of communication between the cloud server and the main system. The collected data and video will be analyzed by the analytic engine of the system. This system is mainly focusing on optimizing and transmitting video data instead of vital sign data.

Pap et al. implemented an IoT based e-health system by using a Raspberry Pi [9]. Node.js server-side application was used for recording the data in a specific chart and providing a web interface for the user to study the chart. The e-health system is equipped with blood pressure, pulse oximeter, airflow, galvanic skin and temperature sensors. This system allows the user to select the data recording mode either a live session or a recording session. In the live session mode , the data will only be temporarily stored. For the recording session, the data will be permanently stored.

III. METHODOLOGY

The proposed patient monitoring system consists of four modules: a sensor, a main controller, a communication module and a webserver as shown in Figure 1. An ECG sensor AD8232 is used to collect the ECG data from the cardiac patient. It measures and amplifies the ECG signal based on the electrical activity of the heart muscle. In the proposed system, Arduino ESP 32 which equipped with WiFi is used as the main controller. The collected real-time ECG data will be processed by the main controller and then transmitted to the MQTT broker via WiFi module. The ECG data is published to the webserver through Mosquito Message Queuing Telemetry Transport (MQTT) broker which is hosted by the Raspberry Pi 3.

An MQTT-DB relay is applied in this work. It subscribes the same Topic with the webserver in the MQTT broker, buffering the ECG JSON data and then storing the ECG data in the database. This allows the proposed system to display the real-time ECG signal on the webserver and stores it in a database simultaneously. Besides webserver, the doctor also can view the real-time ECG signal via Android mobile application (App). The App can display both heartbeat rate (BPM) and a real-time ECG chart. The BPM is extracted from the received real-time ECG data, it updates every 3 seconds. Doctors or nurses can access the database server to view the recorded history ECG data.

This work uses MQTT network protocol instead of Hypertext Transfer Protocol (HTTP) because MQTT requires low bandwidth, low power consumption and suitable for IoT development [10]. The ECG signals data are packed into a JSON Format string before sending it to the webserver and database. This greatly improves the efficiency of data converting at the client side because both webserver and database have a built-in function that can process the JSON format data.

The webserver was implemented by using JavaScript programming language. The web sockets are used to display a real-time ECG signal based on the ECG JSON data that is transmitted from the proposed patient monitoring system. The webserver also performs package loss and jitter delay calculation. The equation for calculating the packet loss is given as follow:

$$P_L = N_S - N_R \tag{1}$$

where P_L is package loss, N_S is the number of packages sent by the main controller and N_R is the number of packages received by the webserver. Equation (2) is used for calculating the jitter delay.

$$J_D = T_R - T_S \tag{2}$$

where J_D is jitter delay, T_S is the time interval between the packages sent by the main controller and T_R is the time interval between the package received by the webserver. The average jitter delay A_{JD} can be computed by using equation (3).

$$A_{JD} = \frac{\sum J_D}{N_P} \tag{3}$$

where S_{JD} is the total jitter delay and N_P is the total number of packages.

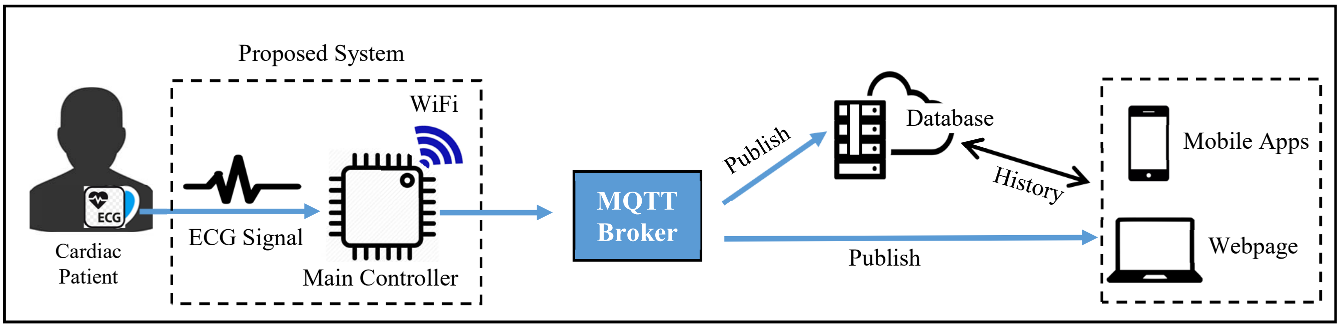


Fig. 1. Architecture of the proposed real-time patient monitoring system

IV. Experiment Setup

In order to test the performance of the proposed system in different environments, two experiments had been set up in real networking environments. The first experiment is set up in a private network or local area network (LAN). The proposed system, database system and computers as shown in Figure 1, are connected to the same access point. Assuming the patients in the hospital are equipped with the proposed system. The collected ECG data are stored in the in-house server. Doctors monitor their patients' real-time or previously recorded ECG data from their office. The second experiment is set up in the public network or Wide Area Network (WAN), as shown in Figure 2. Assuming the proposed system is connected to the home WiFi. The collected real-time ECG data is transmitted to the cloud server via internet. Doctors access the cloud server for observing the patient real-time ECG through their own mobile cellular network.

The performance of the proposed system is measured based on the number of packet loss and average jitter delay. The real-time ECG signal was observed at Arduino Serial Monitor and webserver. The ECG signal displayed on the Arduino Serial Monitor is the real-time ECG signal collected by the AD8232 sensor. This ECG signal will be published by the MQTT broker to store in the database system and also display on the webserver.

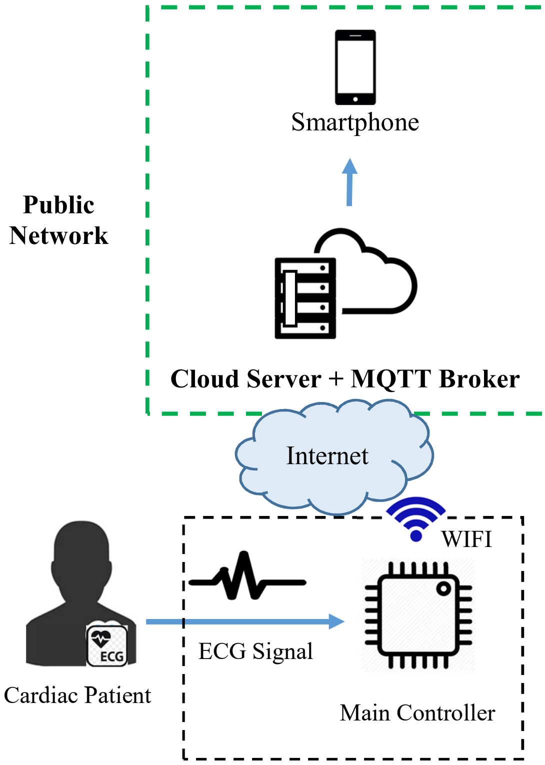


Fig. 2. Experiment setup in the public network

V. Results and Discussion

The experiment results show that there are no packet loss and packet errors in both private and public networks as shown in Table I and II, respectively. This is because MQTT protocol has a checksum function. It will resend the packet if subscribers failed to receive the packet. However, this leads to delay in sending the next packet.

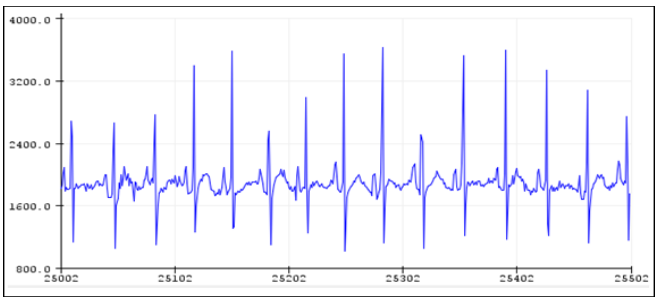
To further verify there is no packet error, we compared the total value of ECG data of the selected range of 5000 packets between sender (main controller) and subscriber (database system or webserver). The results show that both the sender and subscriber have an identical sum of ECG values, as shown in Table I and II. In term of jilter delay, the performance in public network is about ten times greater than private network. This is because the public network has greater transmission distance compared to the private network. The delay of 50.08 ms in public network is acceptable because the maximum tolerance can be up to 1 second [11]. Figure 3 and Figure 4 show the ECG signal that displayed at Arduino Serial Monitor and webserver in private network and public network, respectively.

TABLE I. Performance of the Proposed System In Private Network

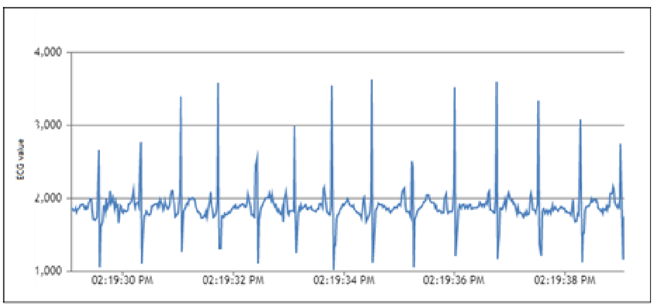
Module	Number of ECG packages	Sum of ECG values	Average Jitter Delay, A_{JD} (ms)
ECG packets sent by main controller	5000	9511560	5.714
ECG packets received by Webserver	5000	9511560	

TABLE II. Performance of the Proposed System In Public Network

Module	Number of ECG packages	Sum of values	Average Jitter Delay, A_{JD} (ms)
ECG packets sent by main controller	5000	9477828	50.0832
ECG packets received by Webserver/database	5000	9477828	

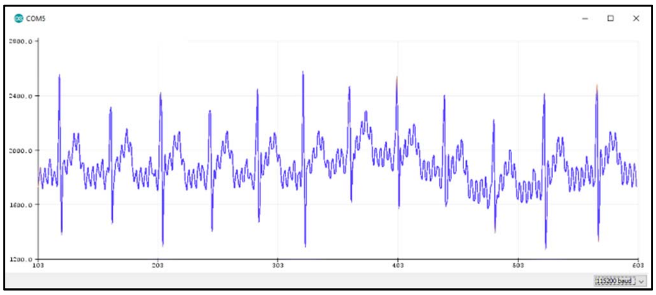


(a)



(b)

Fig. 3. ECG signal displayed on the (a) Arduino serial monitor and (b) webserver in private network



(a)

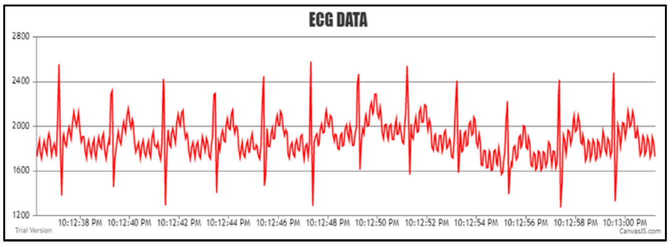


Fig. 4. Section of ECG signal displayed on the (a) Arduino serial monitor and (b) webserver in public network

Apart from the webserver, Android mobile application (App) is developed for the doctor to view the real-time heartbeat rate (RPM) and ECG signal through the smartphone. Figure 5 and Figure 6 show the interface of real-time BPM and ECG signal on the Android mobile App.

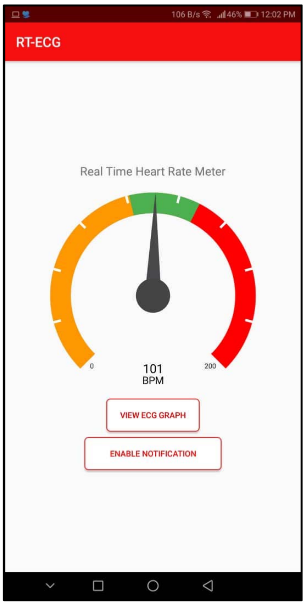


Fig. 5. Real Time heart rate meter in Android App

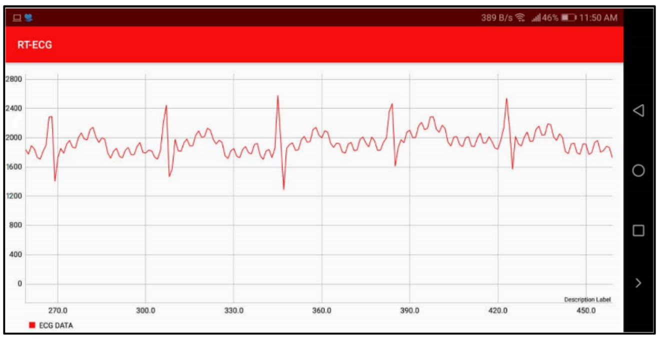


Fig. 6. Section of Real-Time ECG signal displayed in Android App

VI. Conclusion

The developed real-time patient monitoring system has been tested in both private and public networks. The real-time ECG signal was displayed on the web and storing in the cloud server successfully. The experiment results show that there are no package loss and package errors during transmission of ECG packages in both private and public networks. The proposed system can be applied in the hospital. The doctors or nurses can observe the ECG signal of any patients at any time any place through the computer or smartphone, without going to the wards. This system is able to reduce the patients' traveling time and cost especially for those patients who are from suburban or rural areas. Doctors can monitor the ECG of the chronic cardiac patient in distance via the internet. The proposed system has the potential of improving the quality of healthcare services all over the nation.

As future work, ECG self-interpretation algorithm can be implemented into the system so that the system can detect the abnormal ECG signal and generate an alert. This system also can be expanded by adding more e-health sensors to collect various health parameters. Research on reducing the jitter delay and eliminating the noise signal are also required to improve the performance of the proposed system.

References

[1] Ministry of Health Malaysia, "Health indicators," 2018.

[2] S. Patel, H. Park, P. Bonato, L. Chan, and M. Rodgers, "A review of wearable sensors and systems with application in rehabilitation," *J. Neuroeng. Rehabil.*, vol. 9, no. 1, p. 21, 2012.

[3] H. T. Yew, Y. Aditya, H. Satrial, E. Supriyanto, and Y. W. Hau, "Telecardiology system for fourth generation heterogeneous wireless networks," *ARPN J. Eng. Appl. Sci.*, vol. 10, no. 2, 2015.

[4] N. B. Krishnan, S. S. S. Sai, and S. B. Mohanthy, "Real Time Internet Application with distributed flow environment for medical IoT," *Proc. 2015 Int. Conf. Green Comput. Internet Things, ICGCIoT 2015*, pp. 832–837, 2016.

[5] E. Lee, Y. Wang, R. Davis, and B. Egan, *Designing a low-cost adaptable and personalized remote patient monitoring system*. 2017.

[6] S. Roy, A. Rahman, M. Helal, M. S. Kaiser, and Z. I. Chowdhury, "Low cost RF based online patient monitoring using web and mobile applications," *2016 5th Int. Conf. Informatics, Electron. Vision, ICIEV 2016*, pp. 869–874, 2016.

[7] A. M. Ghosh, D. Halder, and S. K. A. Hossain, "Remote health monitoring system through IoT," *2016 5th Int. Conf. Informatics, Electron. Vision, ICIEV 2016*, pp. 921–926, 2016.

[8] H. Moustafa, E. M. Schooler, G. Shen, and S. Kamath, "Remote monitoring and medical devices control in eHealth," in *2016 IEEE 12th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, 2016, pp. 1–8.

[9] I. A. Pap, S. Oniga, I. Orha, and A. Alexan, "IoT-based eHealth data acquisition system," *2018 IEEE Int. Conf. Autom. Qual. Testing, Robot. AQTR 2018 - THETA 21st Ed. Proc.*, pp. 1–5, 2018.

[10] T. Yokotani and Y. Sasaki, "Transfer protocols of tiny data blocks in IoT and their performance evaluation," *2016 IEEE 3rd World Forum Internet Things, WF-IoT 2016*, pp. 54–57, 2017.

[11] H. T. Yew, E. Supriyanto, M. H. Satria, and Y. W. Hau, "Adaptive network selection mechanism for telecardiology system in developing countries," in *3rd IEEE EMBS International Conference on Biomedical and Health Informatics, BHI 2016*, 2016.