Patient Monitoring System

Oguz Alp Saglam
Department of Computer Science
Binghamton University
New York, USA
osaglam1@binghamton.edu

ABSTRACT

The biggest problem, in these days, is COVID-19. All human beings must focus on this threat, setting aside their self expediencies, such as wars for oil or natural gas so on. People must achieve 3 main goals during the COVID-19 pandemic; stay at least 6 ft away from others, wear a medical face mask and washing hands. Until a vaccine invention that kills and takes out the COVID virus out of the human body, this is the only way to protect ourselves. It should not be hard to implement.

With the same approach of 3 rules of these days should be same for the doctors, nurses or any other health workers. They are also human beings. The only difference between them and us is they dedicated their lives to save or heal the citizens. To make things easier and limit their movements in the hospital or any other place, Patient Monitoring System first prototype is proposed.

Doctors or nurses can monitor their patients from outside of the patient's room. There are various sensors and a camera to monitor. The system is also allows to the computer, that stays with the patient, to add many actuators as possible. In this project, a ventilation machine is added. Eventually; there is no limit for sensors, cameras and the actuators for various patient. The system is working with either a personal computer(PC) or an embedded system on module(SoM), as long as SoM has USB ports, cards such as Raspberry Pi. The only difference between the PC and SoM would be the energy consumption.

KEYWORDS

Covid, IoT health solutions, Patient Monitoring, ECG, SPO2, BPM, Ventilation

1 Introduction

The "Internet of Things" concept is very popular almost for a decade. Yet there are some security issues with IoT[1]. Common use cases are; e-commerce, e-health, e-home, etc. Since the concept is still valnerable for various malicious attacks internally or externally. So this project related to human health concept to reduce the vulnerability in human manner. The motivation is behind this project is; "Why would a hacker wants to kill or hurt another person in a hospital?".

Beside of the security issues of IoT, there is a bigger problem that threatens the human beings in these days. It is called COVID-19 virus. COVID-19 virus is started at Wuhan, China. It spread all over the world and already killed almost 1,5 billion people[2]. There is no medicine for the disease yet, there are vaccine invention movements around the world that they are in progress.

In this work, our target society does not consists of every person who living in the planet. Target society is health workers around the world, including doctors, nurses and other labourers on the medical health community.

Project helps those labourers to limit their movement in the hospital and preventing them to go to patient room for checking the patient purposes.

Addition of infinite number of devices to Patient Monitoring System allow health people to monitor their patients remotely, and they might visit their patient only for intervention and injection of medicine.

2 Design

2.1 Software

2.1.1 Linux Based Operating System

The PMS is structured on Linux based Operating System. Most common usage examples are handled in Ubuntu, a Linux based Operating System.

The PMS can adapt all kinds of computers; it can be a personal computer(PC) or System on Modules(SoM), as long as they have USB ports on it, such as Raspberry Pi or Jetson Nano.

2.1.2 Robotic Operation System

Robotic Operating System (ROS)[4] is an embedded operating system that can also be called a communication system between the components that are wired to any computer. There were some issues during the progress of PMS. One of them is the Serial Communication between microcontroller and the computer. Without ROS, the sensor data couldn't be used for multi purpose. ROS solved the problem which save the data in the computer once and, send or read the data to webserver or a graphical user interface(GUI).

2.1.3 Node-Red

Node-Red[5] is a web interface application that can read data from sensors, microcontrollers or even ROS[6].

Node-Red is used in this project in order to handle devices that doesn't support Linux Operating System directly; such as Windows OS/ MacOS machines or smartphones. In order to get

the data from various devices for analysis, Node-Red was one of the best options on for the PMS.

2.1.4 OpenCV

OpenCV is used for computer vision applications[7]. PMS does not consist of any image processing or deep learning examples, but it is used to transfer the image that taken from the camera to health issuers to monitor the patient remotely. The scenario consists of both to watch patient and monitor the levels of medicine that injected to the patient.

There is no use case for image processing for PMS, but it is open to develop image processing and deep learning for infinite applications from just one camera.

In general cases, OpenCV or any other applications takes one camera per image processing script. With the help of ROS, the raw camera image being pushed to the ROS internal sockets. Developers can derive infinite image source from one camera[8].

2.2 Hardware

In the first prototype of the Patient Monitoring System(PMS) includes 2 Arduino UNO, ECG module, SPO2 and Heartbeat module, temperature sensor, a role-playing ventilation machine and a camera for the equipment.

Figure 1: First prototype of PMS.

2.2.1 AD8232 ECG Module

- What is ECG?

An ECG is a digital recording of the electrical signals in the heart. The extended version of ECG is electrocardiogram. The ECG is used to determine heart rate, heart rhythm and other information regarding the heart's condition. Mostly used to diagnose heart anomalies[9].

Figure 2: ECG waveform

Analysis of the waveform of an ECG; first upward tracing is the P wave, it indicates atrial contraction. The QRS complex begins with Q, followed by the peak, R, and S wave downwards. The QRS complex indicated ventricular depolarization and contraction. Lastly, T wave indicates ventricular re-polarization.

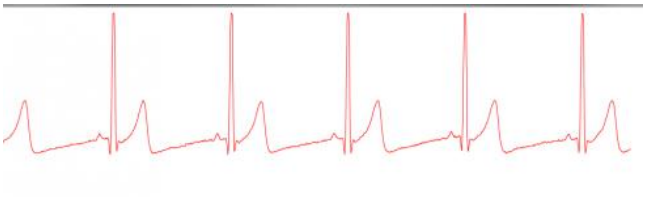

Figure 3: When the module separately run from the Arduino.

-Connection and Wiring

AD8232 has 9 pinouts. As the instructions[3] said, 5 of the pinout used; 3.3V, GND, LO-, LO+ and OUTPUT pins respectively in Arduino as well, to receive data.

2.2.2 MAX30102 SPO2 & Heartrate Module

- How does the module works?

The sensor has built-in leds that send small beams of light to the finger. Measuring the changes in light absorption in oxygenated or deoxygenated blood[10].

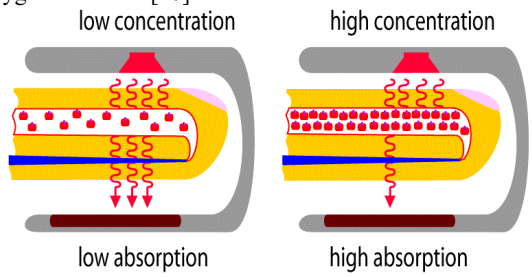

Figure 4: Workflow of an oximeter

-Connection and Wiring

The sensor has I2C communication and uses timer interrupt by the code. The developers that wish to use interrupt pin, the sensor has a pinout for that as well. Basically; 3.3V, GND, I2C pins (SDA and SCL) are plugged in and ready to use.

2.2.3 LM35 Temperature Sensor

LM35 is an analog temperature sensor. It can be used in the range of -55 to +150 degrees of Celcius. Generally, it is used to measure the temperature of the environment or a room temperature. In this scenario, it is used the measure patient's body temperature.

2.2.4 5V Fan Ventilator

Another role-player of the first prototype, uses another Arduino UNO with a motor driver L298N. Basically, it is a 5V personal computer fan that controlled with an extra Arduino. It moves for 5 seconds when it triggered by the user after 5 seconds, it will move in reverse for 5 seconds as well. The reason is the usage of an extra Arduino is to keep receiving sensor data from the other Arduino and run the fake ventilator at the same time[12].

2.2.5 Camera

The camera module will work as a surveillance/IP camera to monitor the levels of injected medicine and the patient.

Additionally, open for developing image processing or deep learning.

3 Implementation

The sensors are connected to one Arduino that publishes the sensor data via Rosserial package over Serial Port, which is USB ports. The ventilation machine is connected to motor driver L298N with PWM pins in order to set speed to ventilator. The motor driver connected to second Arduino and become a subscriber node over Rosserial, reverse implementation of the first Arduino.

The receiving and sending the data to Web is handled over ROS Bridge Server package[15]. The package allows the communication between the computer and browser. With ROS Bridge, Node Red and ROS can interface over a Web Browser.

The remote computers, that has Ubuntu, ROS, OpenCV and some other dependencies, can connect the PMS. Simply by adding "export ROS_MASTER_URI=http://PATIENT_IP:11311" the line to bash file ".bashrc". Then the user allowed to see the topics and nodes that ROS has[16]. This remote connection allows the Health people to view clearer see of graphs and allowing them to watch camera payload.

Camera is also publishing the camera payload over ROS via CV Bridge. CV Bridge allows doing manipulation, such as image processing and deep learning, on the payload infinitely. The camera payload can be monitored in only Linux based Operating System which has ROS and OpenCV installed[8].

Plotting the sensor data; handled in two methods. Firstly, the data can be watched from ROS Qt Graphical User Interface called "rqt"[17].
Second, the sensor data can be reachable from non-Linux machines and smartphones over Web Browser with Node Red.

Ventilation machine can be triggered with a simple switch on Node Red based GUI on the Web Browser from any device. Also, can be triggered from Linux shell terminal with commands.

ACKNOWLEDGMENTS

Patient Monitoring System, will work based on the dependencies that we've covered above. It can be a personal computer or a simple Raspberry Pi. The pros and cons must be considered when making choice, which device should be used? Personal Computer will consume more energy but works more fluent than a system on modules(SoM) unless we develop an Operating System just for this project anyway. On the other hand, a Raspberry Pi with version higher than Model 3B, the Rpi has power over internet specification, it can be booted by ethernet cable. There are many external devices that appending on the Rpi, less energy may led to data/packet loss. Eventually, it will cost less energy but, may slow down the data flow as it runs for a long time.

The issue for the first prototype; is the frequencies of the sensors. SPO2 module must work way faster than the others, and ECG module should be running at lower frequency. The issue can be solved by adding a new microcontroller, an Arduino.

REFERENCES

[1] M. M. Hossain, M. Fotouhi and R. Hasan, "Towards an Analysis of Security Issues, Challenges, and Open Problems in the Internet of Things," 2015 IEEE World Congress on Services, New York, NY, 2015, pp. 21-28, doi: 10.1109/SERVICES.2015.12.
[2] https://www.worldometers.info/coronavirus/
[3] https://how2electronics.com/ecg-monitoring-with-ad8232-ecg-sensor-arduino/
[4] About ROS: http://wiki.ros.org/
[5] About Node Red: https://nodered.org/
[6] Node Red Interfacing ROS Bridge: https://flows.nodered.org/node/node-red-contrib-ros
[7] What is OpenCV: https://docs.opencv.org/master/
[8] CV Bridge: http://wiki.ros.org/cv_bridge
[9] How ECG modules works: https://how2electronics.com/ecg-monitoring-with-ad8232-ecg-sensor-arduino/
[10] How SPO2 module works: https://how2electronics.com/blood-oxygen-heart-rate-monitor-max30100-arduino/
[11] Temperature Sensor: https://how2electronics.com/blood-oxygen-heart-rate-monitor-max30100-arduino/
[12] L298N Motor Driver: https://components101.com/modules/l293n-motor-driver-module
[14] ROS Serial: http://wiki.ros.org/rosserial
[15] ROS Bridge: http://wiki.ros.org/rosbridge_suite
[16] ROS topics on multi computers: https://razbotics.wordpress.com/2018/01/23/ros-distributed-systems/
[17]ROS RQt interface: http://wiki.ros.org/rqt_plot