2nd Buffalo Day *for* 5G and Wireless Internet of Things

Coronavirus: Location, Oxygen-saturation, Verification-of-temperature, and Individual-pulse-rate Detection Device (CLOVIDD)

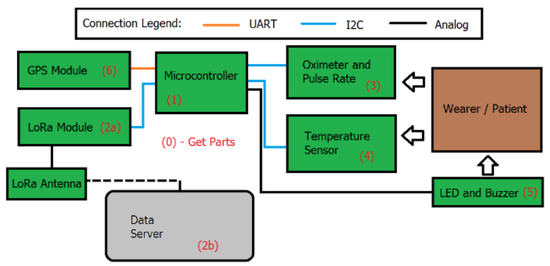
Maitreyee Joshi, Steven Starowicz

Introduction

Given speculation that coronavirus may cause a sudden decrease in patient oxygen saturation levels[1,2], we explore the creation of a device that observes the skin temperature, heart rate, and blood-oxygen levels. Data is collected and processed locally, with anonymized alert status indicators sent to a remote server using the LoRa protocol.

Device Construction

The proposed device includes a combination heart-rate and pulse oximeter (MAX30102), an infrared thermometer (MLX90614), a microcontroller (Arduino MEGA2560), a custom-built university-provided LoRa transceiver, and a GPS module (GT-U7). The component diagram below details the initial conceptual layout of the device.



Data collection for the device is based upon the use of I^2C and UART communication channels, with the microcontroller firmware collecting periodic data from the MAX30102, MLX90614, and GT-U7 components. Simple processing algorithms trigger an alert when the values are outside of a preset range. If an alert is triggered, the device will provide the type of alert and GPS data for the location to the LoRa gateway.

Project Schedule

The initial proposal was to complete the design over the course of eight (8) week, beginning with a week of research and selection of prototype parts. Specific allocations for each week are:

1. Evaluate Arduino, setup protocol functions (I^2C, UART)
2. LoRa transceiver accepts data from microprocessor
3. Creation of online portal/database
4. Interface MAX30102 sensor with microprocessor
5. Interface MLX90614 sensor with microprocessor
6. Interface buzzer and LED with microprocessor
7. Interface GPS module with microprocessor

Deliverables

Deliverables are enumerated in two separate categories:

Basic Connectivity

- Read O_2 data from MAX30102 sensor
- Read pulse data from MAX30102 sensor
- Read temperature data from MLX90614 sensor
- Read data from the GT-U7 GPS module
- Activate buzzer and LED alert hardware on command
- Send static data to the LoRa gateway for transmission

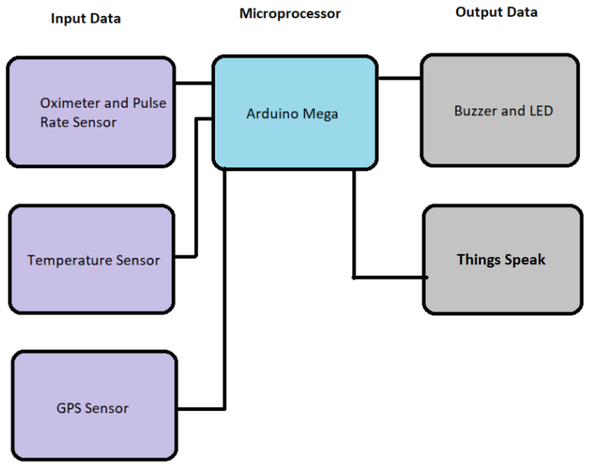
Verification of Data

- Test O_2 data with human subject to determine accuracy
- Test pulse data with human subject for accuracy
- Determine offset for temperature data using human finger or earlobe
- Activate buzzer and LED when sensor data outside of set range
- Parse data from the GT-U7 GPS module for transmission
- Send sensor data to the LoRa gateway for transmission

Results

The assembled device provided readings from all sensors and the GPS module to the serial debugging output port when individually connected, however, only the temperature sensor provides accurate values when combined. Alas, there is only limited direct correlation between temperatures of the finger and that of internal organs. The O_2 and pulse rate values read by the microprocessor vary wildly, possibly due to sensitivity in the processing algorithm. A brief review of online maker and hobbyist forums[3,4] provides a suggestion that the Maxim algorithm has difficulty dealing with ambient light returns during data collection. The GPS module provides NMEA formatted data sentences, but the loop function does not always catch them properly.

Additionally, the microprocessor selected for integration with the sensors and GPS module has proven difficult to interface with the LoRa transceiver. Despite repeated attempts to communicate, the only microprocessor successfully linked with the LoRa transceiver is the Texas Instruments LaunchPad 2560 XL. As a result, our final component connections are shown below, without the LoRa transceiver. This setup is the only way we could test data transmission and capture on the Things Speak server to verify functionality of our device.



Conclusion

The key takeaways from our development cycle are the need to ensure (1) viable sensor data for any IoT device, (2) proper integration of multi-channel data intake to the microprocessor, and (3) packaging the data stream for transmission via the wireless transceiver. Even with our results, we believe that such a device could provide valuable feedback in situations where the patient does not experience any difficulty breathing or obvious distress.

Future Work

Based on the setbacks described in the results section, we propose the following direction for additional development:

- **GPS Sentence Retrieval:** Use of the I^2C protocol may be possible on a u-blox manufactured chip (not the GT-U7 chip we used), removing the problem of missing UART data and having no workable GPS fix.
- **Heart Rate and O_2:** Creation of a custom data processing function, which uses auto-correlation to detect the periodic peaks in the pulse oximeter output. Additional creation of a smoothing algorithm to remove false peaks that misrepresent the pulse rate.
- **LoRa Transceiver Interface:** Use of the SPI protocol to shuttle alert data from the Arduino (slave) to the LaunchPad (master) could allow proper LoRa transmissions using a known good microprocessor. Creation of a semi-custom serial handler to interface the LoRa transceiver to the Arduino microprocessor.
- **Temperature Sensor:** Relocate the device to be a headband or similar, to allow forehead temperature checks.

References

1. https://www.health.state.mn.us/diseases/coronavirus/hcp/pulseoximetry.pdf
2. https://www.bbc.com/news/uk-wales-53136289
3. https://github.com/sparkfun/SparkFun_MAX3010x_Sensor_Library/issues/17
4. https://www.instructables.com/Pulse-Oximeter-With-Much-Improved-Precision/

University at Buffalo The State University of New York

Electrical Engineering
School of Engineering and Applied Science
buffalo.edu