

WAMPUM

User Manual

University of Rochester
Electrical & Computer Engineering
Senior Design 2015



Contents

[1. Project Description](#)

[2. Block Diagram](#)

[3. Wireless Node](#)

[3.1 PCB Without Attachments](#)

[3.2 PCB With Battery, Sensor, & XBee Module](#)

[3.3 Node Operation](#)

[4. Web Application](#)

[4.1. Overview](#)

[4.2 Display System](#)

[4.3 Interfacing](#)

[5. Appendix](#)

[5.1. Node Schematic](#)

[5.2. Node Bill of Materials](#)

[5.3. Code](#)

[5.4. Who is WAMPUM?](#)

[5.5. Acknowledgements](#)

1. Project Description

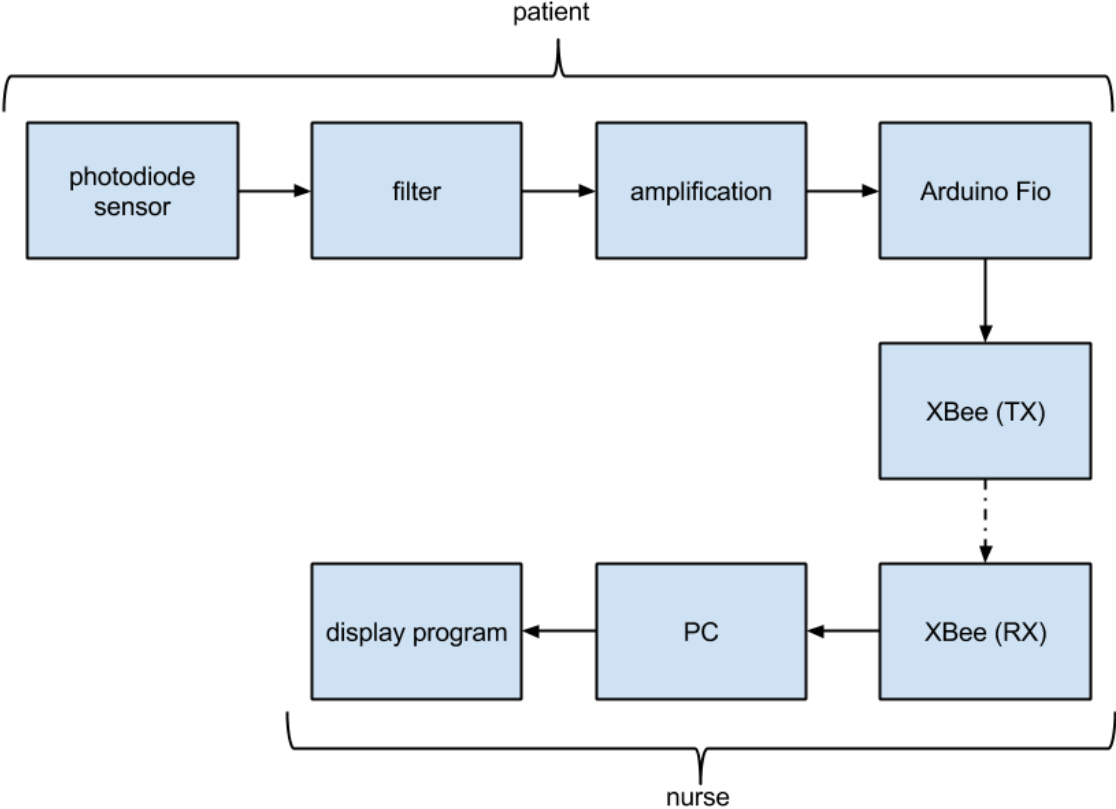
Our project is a system that allows for cheap heart rate monitoring in many patients simultaneously, by using a wireless network. There will be a base station, which consists of a high gain omnidirectional antennae and a laptop. The laptop will display the heart rate of the patients, and generate an alert if the heart rate shows very abnormal behaviour. Each patient will wear a wireless node on their wrist consisting of a small sensor placed on their finger, measuring their heart rate through photoplethysmography. The node will aggregate the pulse ratings, and periodically update the base station with the patient's status over a wireless network. In the event of an emergency, the node will emit an alarm locally in addition to notifying the base station.

This system is a low-cost support system for casualty monitoring after a disaster. In an earthquake or other emergency, sometimes hospitals cannot reach all patients because their equipment is not designed to be used outside of the hospital. Our system could be used to monitor unconscious patients in line to be seen by the medical staff, prompting staff to see them sooner if their condition gets increasingly dire.

An in depth video description can be found at:

<https://youtu.be/-uUUCI6ASRs>

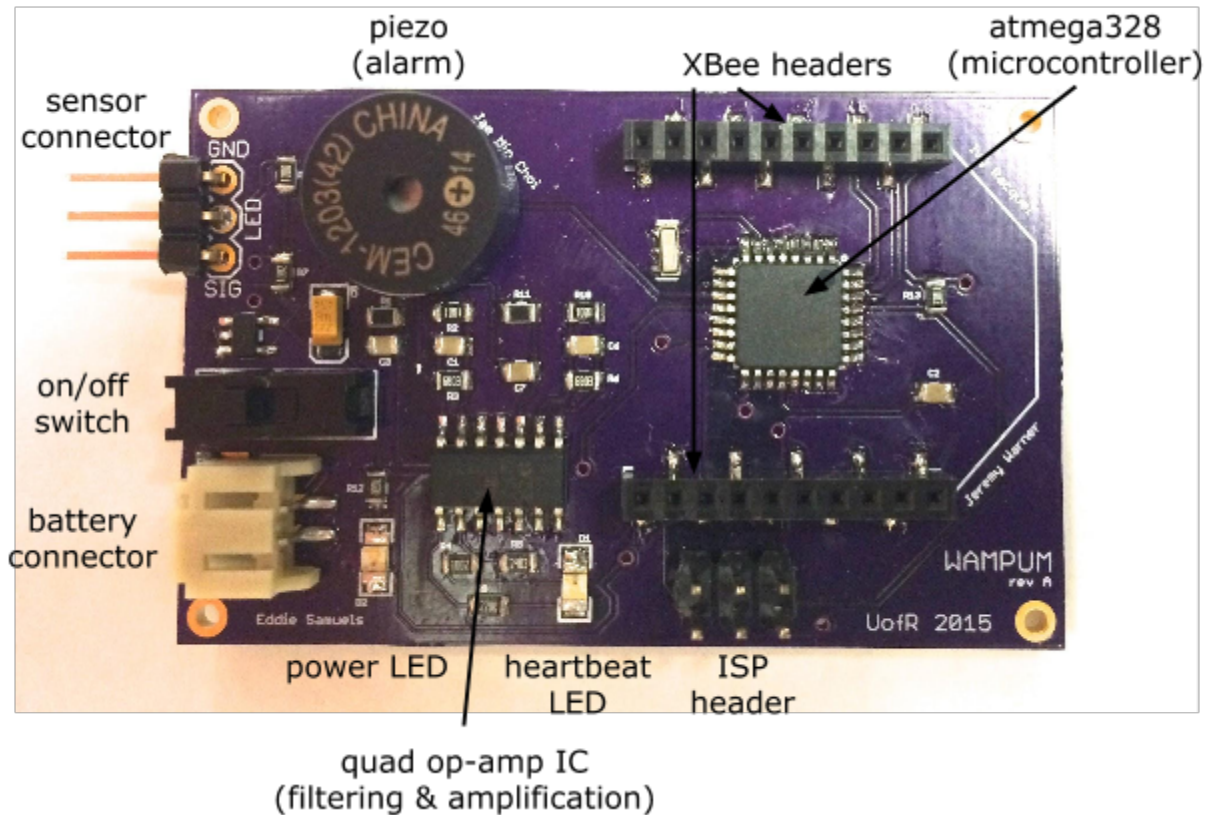
2. Block Diagram



3. Wireless Node

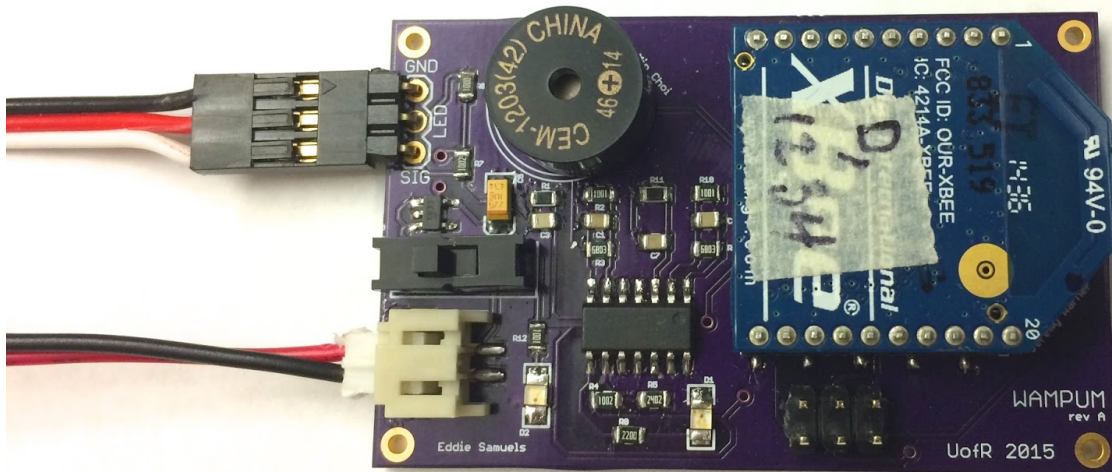
3.1 PCB Without Attachments

Below is an annotated diagram of the WAMPUM PCB.



3.2 PCB With Battery, Sensor, & XBee Module

The black wire of the phototransistor assembly should be connected to the pin labeled "GND" on the 3-pin connector on the top left of the PCB. The li-ion battery can only be inserted in one direction. The XBee module should be inserted with the trapezoidal part facing to the right.



3.3 Node Operation

The node acquires the user's pulse rate by shining an infrared LED on one side of the user's finger and then uses a phototransistor on the other side of the finger to detect the small fluctuations in light that shines through caused by blood rushing into the finger. The heartbeat LED should flash in unison with the users heartbeat. This is to ensure that the sensor is properly placed on the user's finger and can be adjusted if not. The number of pulses over a 10 second period are counted by the onboard microcontroller (using the `Arduino pulseIn` function) and multiplied by 6 to convert to beats per minute (BPM).

Each user's heart rate is wirelessly transmitted to a computer running a node js web application. This application displays the data for each patient in real time. More information about the web application can be viewed in Section 4.

If the user's heart rate goes below 30 or above 150 then the onboard piezo alarm will sound to alert any nearby medical staff. This alarm can be enabled or disabled through the web application.

4. Web Application

4.1. Overview

The web application is written in node.js and utilizes the express.js framework. It is a easy way to keep track of the status of many patients simultaneously, in addition to the onboard alarm that the nodes carry.

4.2. Display System

WAMPUM Base Monitor

Patient 0 - 0 BPM

Patient 1 - 72 BPM

Patient 2 - 12 BPM

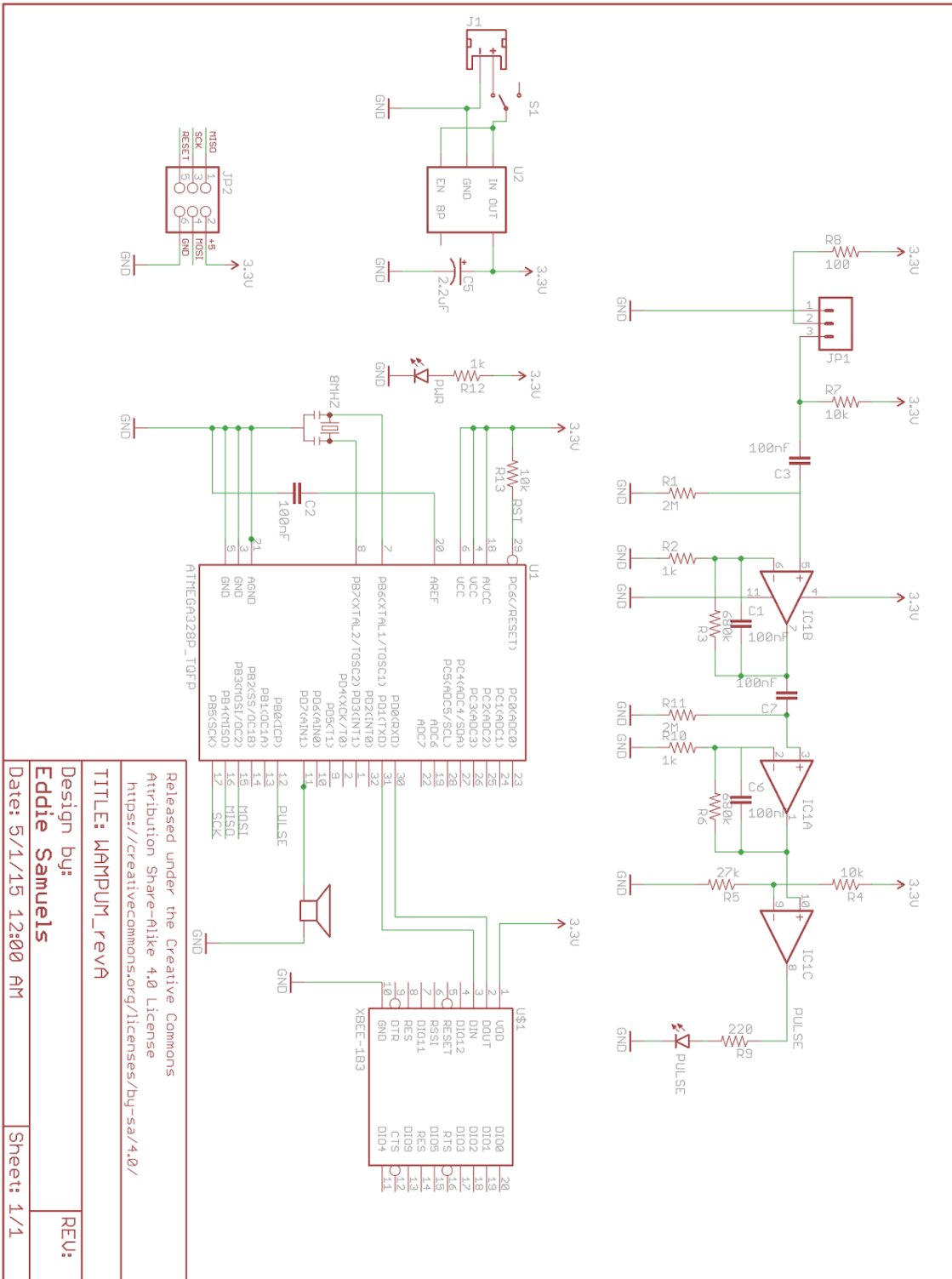
Patient 3 - 0 BPM

4.3. Interfacing

We use the serialport.js library to connect to Arduino, and pipe incoming patient heart rate data into a Postgres database, which is then used to update the web application in real time. This allows to examine the patients history as well their most recent update.

5. Appendix

5.1. Node Schematic



5.2. Node Bill of Materials

Part	Value	Digikey #
C1	100nF	399-7999-1-ND
C2	100nF	399-7999-1-ND
C3	100nF	399-7999-1-ND
C5	2.2uF	399-3690-1-ND
C6	100nF	399-7999-1-ND
C7	100nF	399-7999-1-ND
D1	PULSE	160-1457-1-ND
D2	PWR	160-1456-1-ND
IC1	MCP6234	MCP6234-E/SL-ND
J1	Battery connector	455-1693-1-ND
JP1	3 pin right angle	
JP2	AVR ISP Header	
R1	2M	311-2.00MCRCT-ND
R2	1k	311-1.00KCRCT-ND
R3	680k	311-680KCRCT-ND
R4	10k	311-10.0KCRCT-ND
R5	27k	311-27KCRCT-ND
R6	680k	311-680KCRCT-ND
R7	10k	311-10.0KCRCT-ND
R8	100	311-100CRCT-ND
R9	220	311-220CRCT-ND
R10	1k	311-1.00KCRCT-ND
R11	2M	311-2.00MCRCT-ND
R12	1k	311-1.00KCRCT-ND
R13	10k	311-10.0KCRCT-ND
S1	Switch	EG1903-ND
SP1	SPEAKER/AL11P	102-1153-ND
U\$1	XBEE Sockets	Sparkfun#: PRT-10030
U1	ATMEGA328P_TQFP	ATMEGA328P-AU-ND
U2	3.3V Reg	576-1259-1-ND
Y1	8MHZ	490-1195-1-ND

5.3. Code

Both the Arduino and web application code for the WAMPUM project can be viewed on our github repository:

<https://github.com/jeremywrnr/wampum>

5.4. Who is WAMPUM?

Team WAMPUM is composed of the following members:

- Ivy Awuor
- Jae Min Choi
- Eddie Samuels
- Jeremy Warner

5.5. Acknowledgements

Team WAMPUM would like to thank our advisors, Professor Wendi Heinzelman, Professor Derefinko, and Professor Motley for their guidance and assistance throughout the year. We would also like to thank the Electrical & Computer Engineering department for sponsoring the project and providing us a place to work.