Emergency Vitals Monitor

Team 11 - Brandon Noble, Songtao He, Zihong Zeng
ECE445 Project Proposal - Spring 2020
TA - Madison Hedlund

1. Introduction

1.1 Objective

Problem: Administering first aid in disaster situations is an extremely stressful task, that is prone to error if not done by a very well trained professional. Under circumstances with multiple injuries (gunshot, heat/fire related, earthquake, etc.), with limited bystanders available, treatment for these injuries are typically left for emergency response personnel. According to an article by H. K. Bakke et. al. that looked into the role of the bystander in trauma response, only 35% of those bystanders who assisted the injured had first aid training. Furthermore, it is estimated that 6-20% of trauma victims who die prior to making it to the hospital could have been saved if bystanders had acted to assist them.

Solution: A uniquely colored automated blood pressure cuff with extending rod for taking temperature, which doubles as the alignment for the pressure cuff's microphone. This simplifies the use as much as possible so that only a few people can attach them to as large a group as necessary in as little time needed. After activation, the blood pressure cuff will automatically take readings of blood pressure, heart rate, and temperature every thirty seconds. The blood pressure and heart rate can be used to determine an individual's 'shock index', which would set up a triage system automatically. This data is collated on a device for the user so as to see the ranking of each cuff in use in terms of who needs attention first, and gives first aid advice on how to treat the conditions detected. This improves the likelihood of an individual performing effective first aid, as well as ensuring that bystanders are able to perform first aid on those who need it most if first responders are limited.

1.2 Background

There is a similar product named QardioArm[1] that supports blood pressure monitoring for multiple users. This product requires to be paired with other mobile devices before displaying its measurements and it’s mostly used in daily life instead of in emergency situations that it lacks the ability to rank the priority of different patients. Our project ranks the severity of patients depending on MSI (modified shock index)[2] and improves first aid capabilities in untrained bystanders by providing methods to assist injured individuals.

1.3 Physical Design

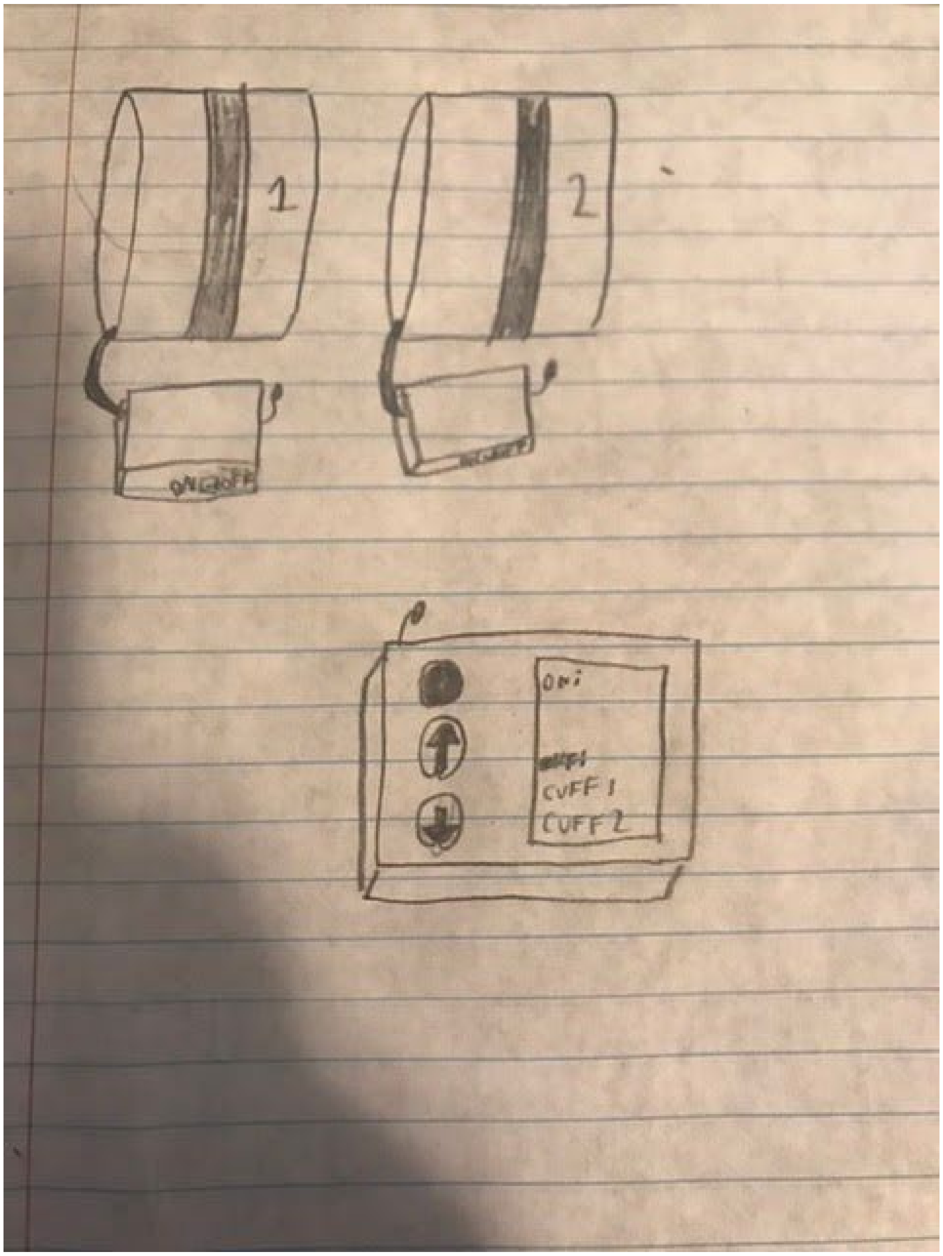

Figure 1. Physical Design High-Level

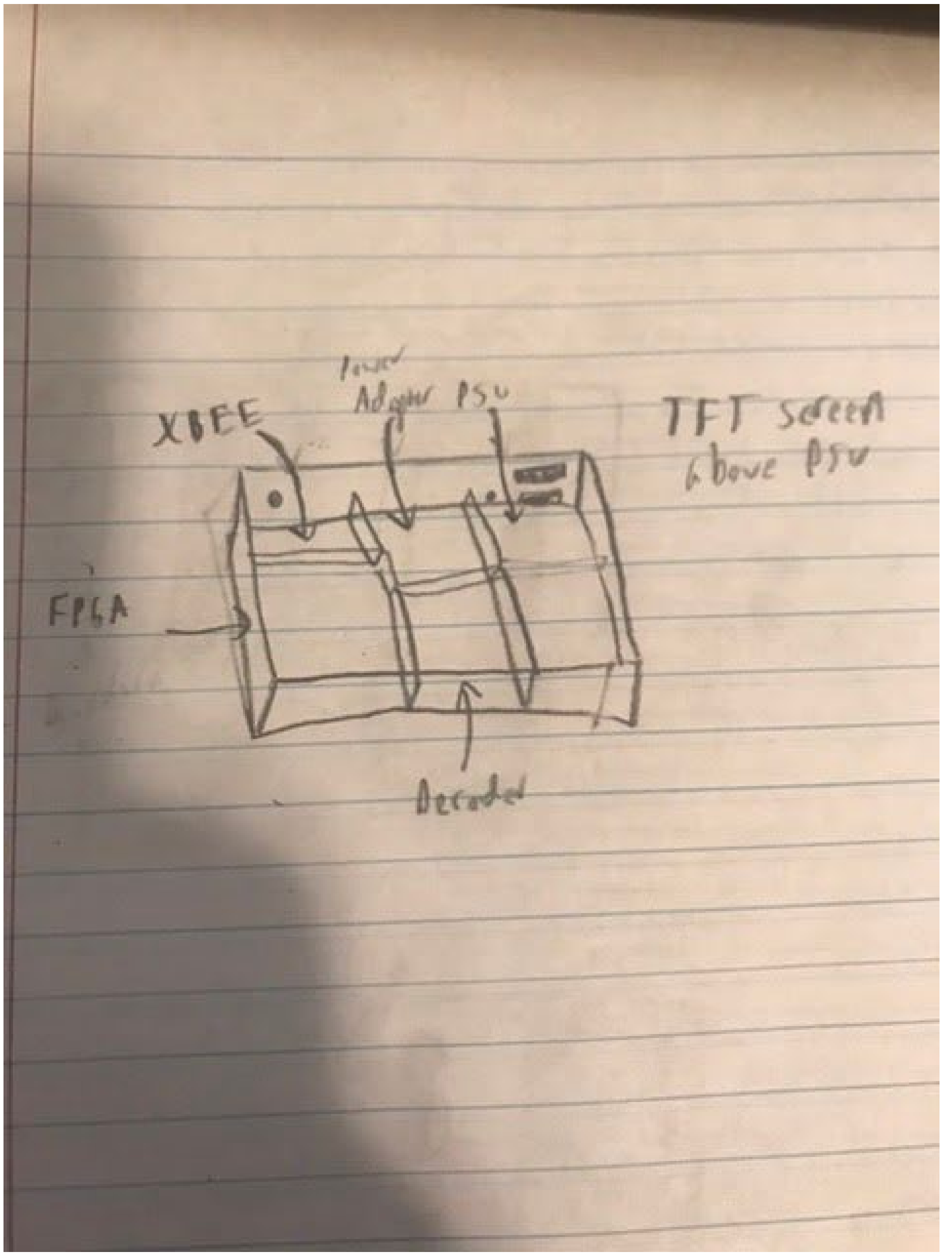

Figure 2. Display Box Layout

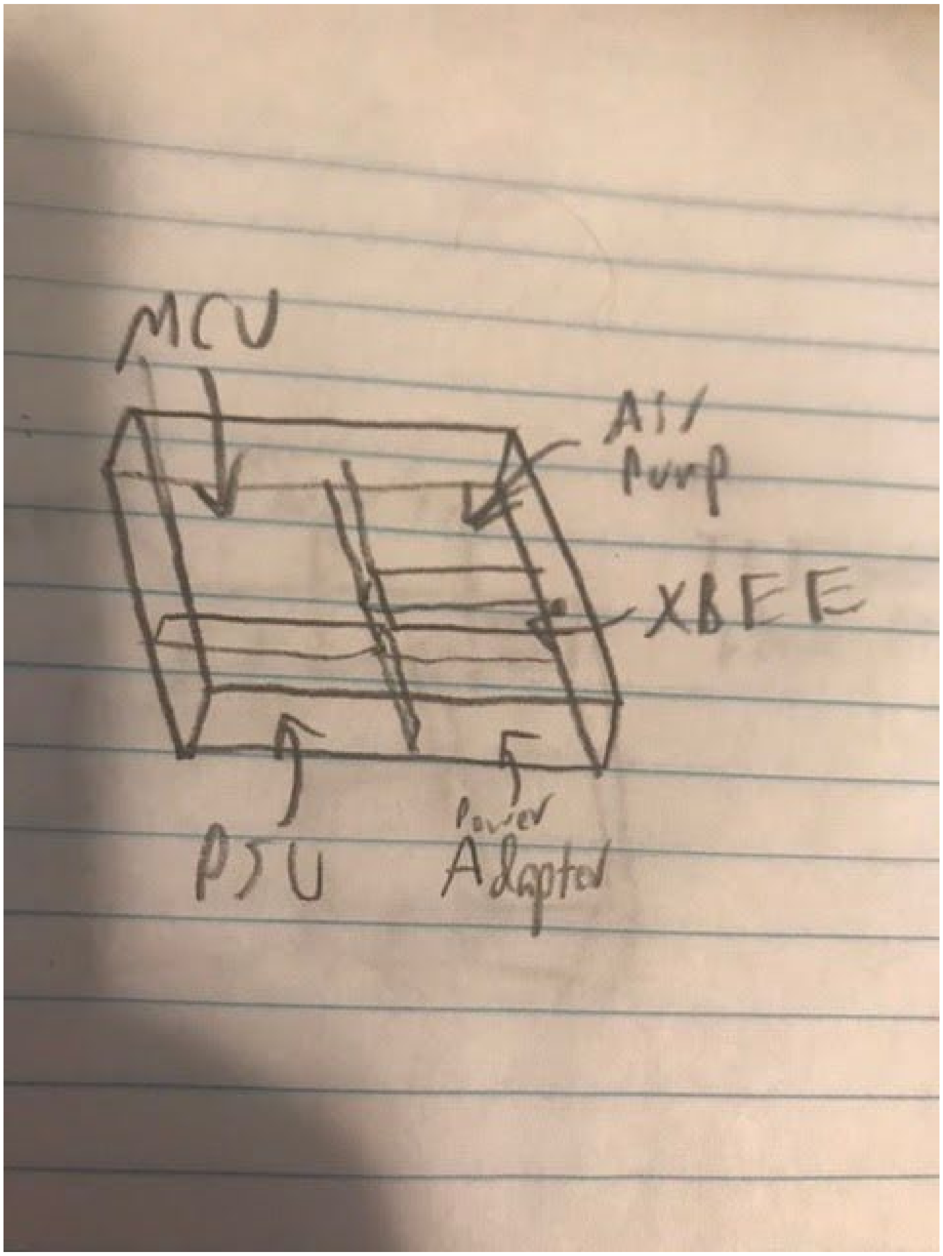

Figure 3. Measurement Unit Box Layout

1.4 High-Level Requirements

- Automatic blood pressure cuffs must operate every one to two minutes, sending vital readings to display device for triage ranking
- Display Device must be able to handle more than one cuff being used at a time
- Display Device doubles as a guide for first aid, with topics prioritized based off of the vital readings of the active blood pressure cuffs.

2. Design

2.1 Block Diagram

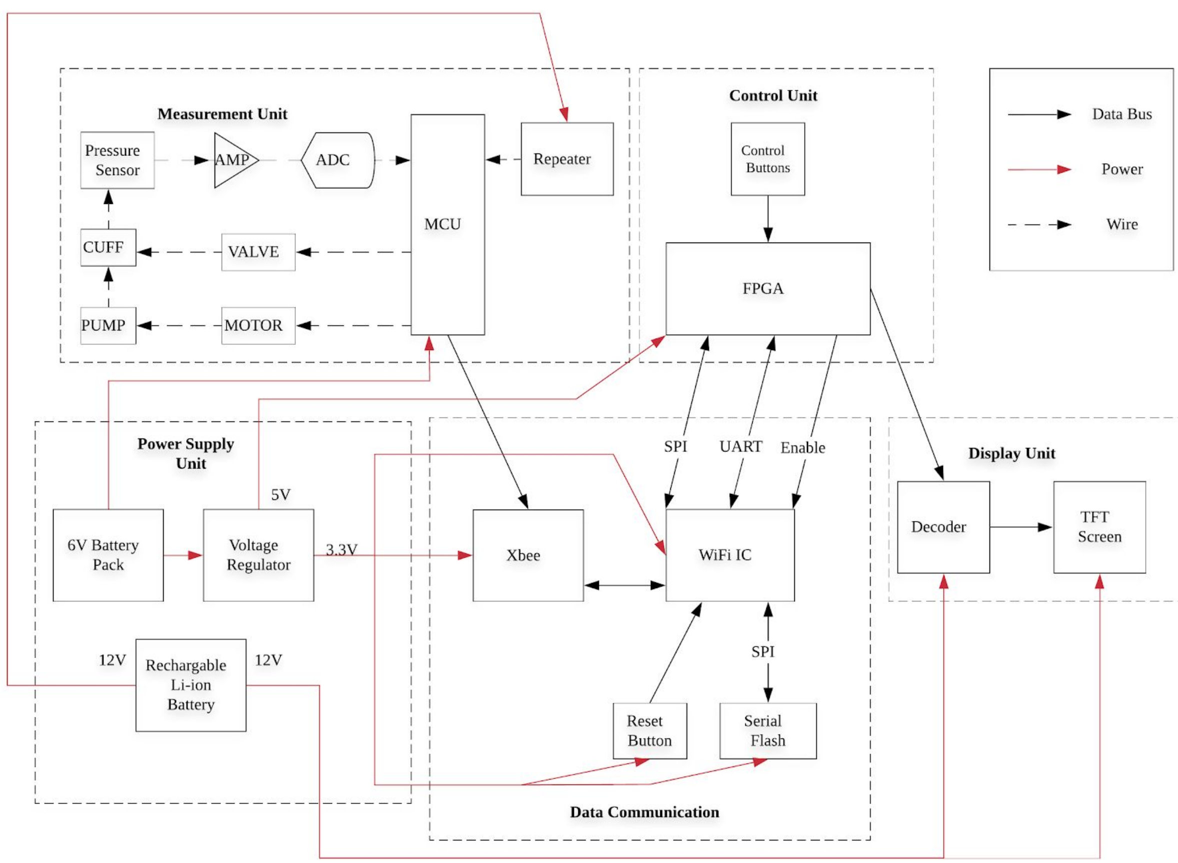

Figure 4. Block Diagram

2.2 Functional Overview and Block Requirements

2.2.1 Power Unit

Using the integrated power supply from our sphygmomanometer, we'll power our cuff and data communication device. Our display device and processing unit will either use a rechargeable lithium-ion battery, or a disposable battery pack.

Requirement: Power supply of 6V +/- 1V is able to power blood pressure cuff for at least 30 minutes.

Voltage Regulator

Requirement	Verification
1. Provides 3.3V +/- 5% and 5V +/- 5% from a 6V- 12V source. 2. Can operate at 0-300mA 3. Maintains thermal stability below 125 degree Celsius.	1-2. A. Using the circuit schematic in figure 3., we connect 6V battery supply to the input and regulate 3.3V and 5V. B. Measure the output voltageVREG using an oscilloscope, ensuring that the voltage stays within 5% of 3.3V and 5V. 3. A. Use an IR thermometer to ensure that the temperature of ICs stays below 125 degree celsius during above steps.

2.2.2 Measurement Unit

A measurement unit reads the vitals of the individual it's attached to. It measures blood pressure, heart rate, and temperature, sending this data to the processing unit through the wifi module (ESP-32).

Blood Pressure Cuff

A sphygmomanometer, contains an air pump and pressure sensor to determine the Diastolic and Systolic blood pressures, as well as the heart rate through the impulses of pressure that come through.

Requirement	Verification

1. The cuff must be able to read Diastolic blood pressures as low as 60 mmHg consistently with readings taken every 1-2 minutes. 2. Cuff must be able to operate in at least a minimum of 32 degrees Fahrenheit weather.	1. A. Measure blood pressure every 3.5 minutes, at recommended delay between readings. Record average Diastolic/Systolic blood pressures. Take blood pressure readings every 1 minute. If readings deviate, compensate at FPGA calculations. 2. B. Measure blood pressure every 3.5 minutes, at room temperature. Record average Diastolic/Systolic blood pressures. Take blood pressure readings in a freezing environment. If readings deviate, installation of an ambient temperature sensor may be necessary to compensate pressure readings at FPGA calculations.

2.2.3.1 Data Communication

For our initial prototype, we'll be using a Wi-Fi module and XBee to send and receive data from our cuff to the control unit. The power supply is shared between the blood pressure cuff and the cuff's transmitter.

XBee

One XBee RF module receives data from the cuff/measurement unit. It then transmits cuff readings through a data bus to ESP32 Wi-Fi module. We will use XBee along with ESP-32 to set up a mesh network to allow data communication between modules.

Requirement	Verification
1. Must be connected to a 5V/3.3V power supply. 2. Xbee can communicate to ESP32	1. A. Use the constant-current circuit in the Fig.3 connecting the output of the voltage regulator to "VDD" in the image, and draw 300mA B. Measure the output voltage using an oscilloscope, ensuring that the output voltage stays within 5% of 3.3V. 2. A. Configure Xbee using XCTU software B. Connect Xbee to the network setup by

	ESP32 C. Test if Xbee can send message to or recieve message from ESP32

ESP-32 Wi-Fi Module

ESP-32 can perform as a standalone system to interface with other systems to provide Wi-Fi (Bluetooth) through SPI/SDIO[3]. We will use ESP-32 Wi-Fi module to set up a mesh network along with XBee to enable devices to talk to each other. ESP-32 will receive cuff readings collected by XBee and periodically send readings to the control unit.

Requirement	Verification
1. ESP-32 must be able to communicate over IEEE 802.11b/g/n at 4.5Mbps with a 50Ω nominal RF connecton. 2. It must be able to communicate over both SPI and UART.	1. A. Assemble WiFi IC on PCB as specified in the datasheet as the basic application schematic. B. Note default WiFi network on a mobile device 2. A. Connect to the ESP32's UART port with an FT232 UART bridge, as per the FT232 reference diagram, and a computer. This can be done on a through-hole breadboard with an ESP-32 module. B. Program 4.5Mbit HTML page (large photo) to SPI flash (program memory). C. Connect to default network with a mobile device, navigate to webpage from step 4 D. Time loading process, ensure that it is less than one second

Serial flash

The serial flash is connected to ESP-32 by SPI, and it stores the program used by ESP-32 WiFi module. The memory size is 1MB and is subject to change as program size varies.

Requirement	Verification
1. Operates consistently at 80 MHz (depends on real purchased product) 2. Size must be >= 1MB to store program	1. A. Operates the serial flash at 80MHz, and WiFi module is able to read a simple program (send strings) B. Receive correct strings as expected 2. A. Store 1MB memory of program/real-time data B. Read expected data from serial monitor in arduino IDE

Wi-Fi Reset button

The button is connected to ESP-32's reset pin. It only resets network connection and does not affect any other embedded program/unit.

Requirement	Verification
Must be a pressable button with Wi-Fi RESET label	Press the button and ensure Wi-Fi connection is refreshed by looking at the Wi-Fi signal on LED screen

2.2.3.2 Data Transmission Testing

Arduino Uno

Arduino Uno uploads programs to Wi-Fi/XBee Module and serves as a processing unit for data bus. It is connected to a power supply with a breadboard to enable WiFi/XBee logic mappings. Unit testing can be done by connecting each individual device to the Arduino and checking the serial monitor for output values.

Requirement	Verification
Operation voltage is 5V and input voltage is recommended to be 7-12V.	A. Use the constant-current circuit in the Fig.3 connecting the output of the voltage regulator to "VDD" in the image, and draw 300mA B. Measure the output voltage using an

	oscilloscope, ensuring that the output voltage stays within 5% of 5V.

2.2.4 Control Unit

MCU

The MCU built into the blood pressure cuff. A black box in terms of operation. Outputs blood pressure and heart rate into a serial read/write IC. Extending the connection to a parallel read/write IC allows for a connection to the XBee.

Requirement	Verification
Limit voltage to 6V +/- 1V.	Measure the output voltage using an oscilloscope, ensuring that the output voltage stays within 5% of 6V.

FPGA and TFT Display

7.00” 40 pin 800x480 TFT Display screen connected to a button/dial for user input, and a TFP401 decoder to read HDMI data from our HDMI FPGA. FPGA reads data from our WiFi module, calculates the MSI, then displays the ‘triage priority’ based off of the MSI, as well as their heart rate and blood pressure. Pressing the side button toggles between medical advice and the triage priority list.

Requirement	Verification
1. FPGA must read digital input pins for each new data point collected, collecting information such as cuff sending data, blood pressure, and heart rate, and calculate the MSI/change in MSI since last read. 2. Device has a button and dial to swap display to first aid recommendations based off of vitals reading of currently selected blood pressure cuff.	1. A. Read data in a proper range of value and can inspect a 3-digit data/triage priority on screen. B. MSI is properly calculated, as confirmed by hand. Rate of change of MSI is stored to factor into their triage ranking. 2. A. After pressing a dial, the person with triage priority should be displayed correctly on a ranking tab.

Button/dial
A button/dial on device to swap between readings and triage ranking.

Requirement	Verification
A pressable button/dial with label.	Press button and check if reading on screen changes as expected.

2.3 Server

The server design will consist of a Raspberry Pi to set up a webpage demonstrating real time data (both for testing and demo purposes). However, it is not in the scope of this class, but we consider it as a necessity to communicate with the outside world (rescue team, etc.).

2.4 Case

Figure 1, 2, and 3 are drafts of the cases for respective devices.

2.5 Protocol

We use UDP protocol to transmit data between communication modules (ESP32, XBee). Since we have verification mechanisms within our design, it is suitable to use UDP as error checking and correction are not necessary. UDP effectively reduces the complexity of routing protocol.

2.6 Tolerances

- K24C04 (MCU Serial Input/Output Data Storage)
    - Vcc
        - Max: 5.5V
        - Min: 1.8V
    - Serial Input Low
        - Max: Vcc x 0.3
        - Min: -0.3V
    - Serial Input High
        - Max: Vcc + 0.3V
        - Min: Vcc x 0.7
- CD4014B (MCU Parallel Input/Output Data Storage)
    - Vcc
        - Max: 20V

        - Min: -5V
    - Input Low
        - Max: 1.5V @ 5V Vcc
        - Min: 0V
    - Input High
        - Max: 5V @ 5V Vcc
        - Min: 3.5V @ 5V Vcc
- LM555 (MCU Activation Timer)
    - Vcc
        - Max: 16V
        - Min: 4.5V
    - Output Low
        - Max: 0.2V @ 4.5V
        - Min: 0.08V @ 4.5V
    - Output High:
        - Max: 3.3V @ 4.5V
        - Min: 2.75V @ 4.5V

2.7 Schematics

POWER SUPPLY

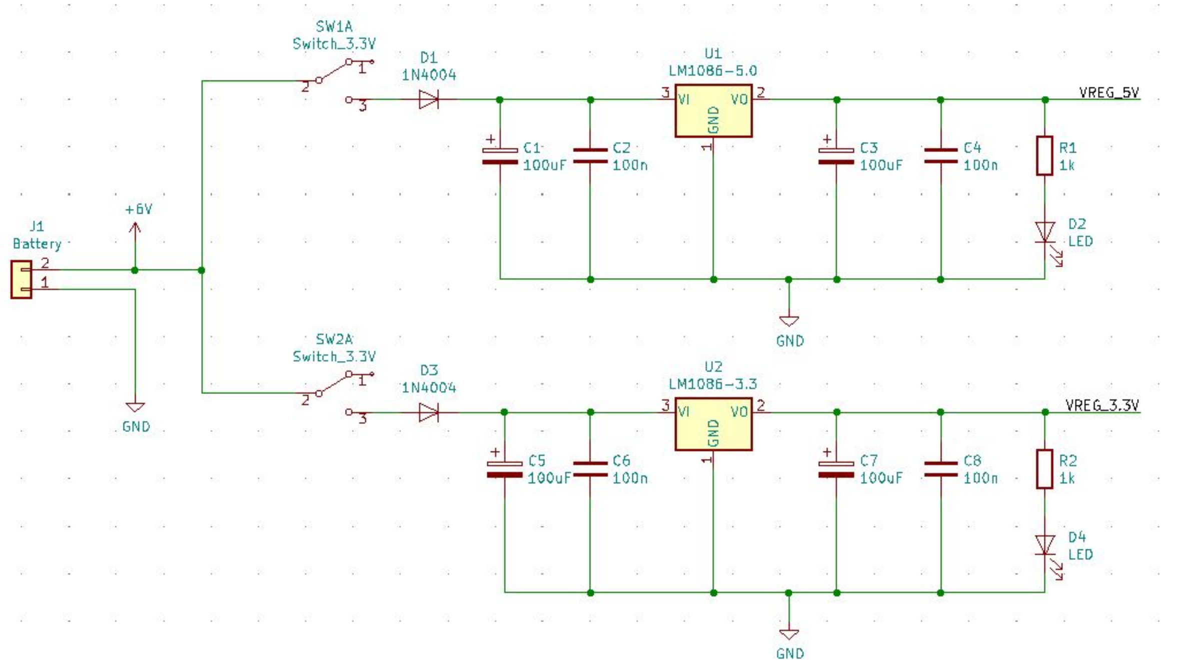

Figure 5. Voltage Regulator Schematic[4], [5]

DATA COMMUNICATION

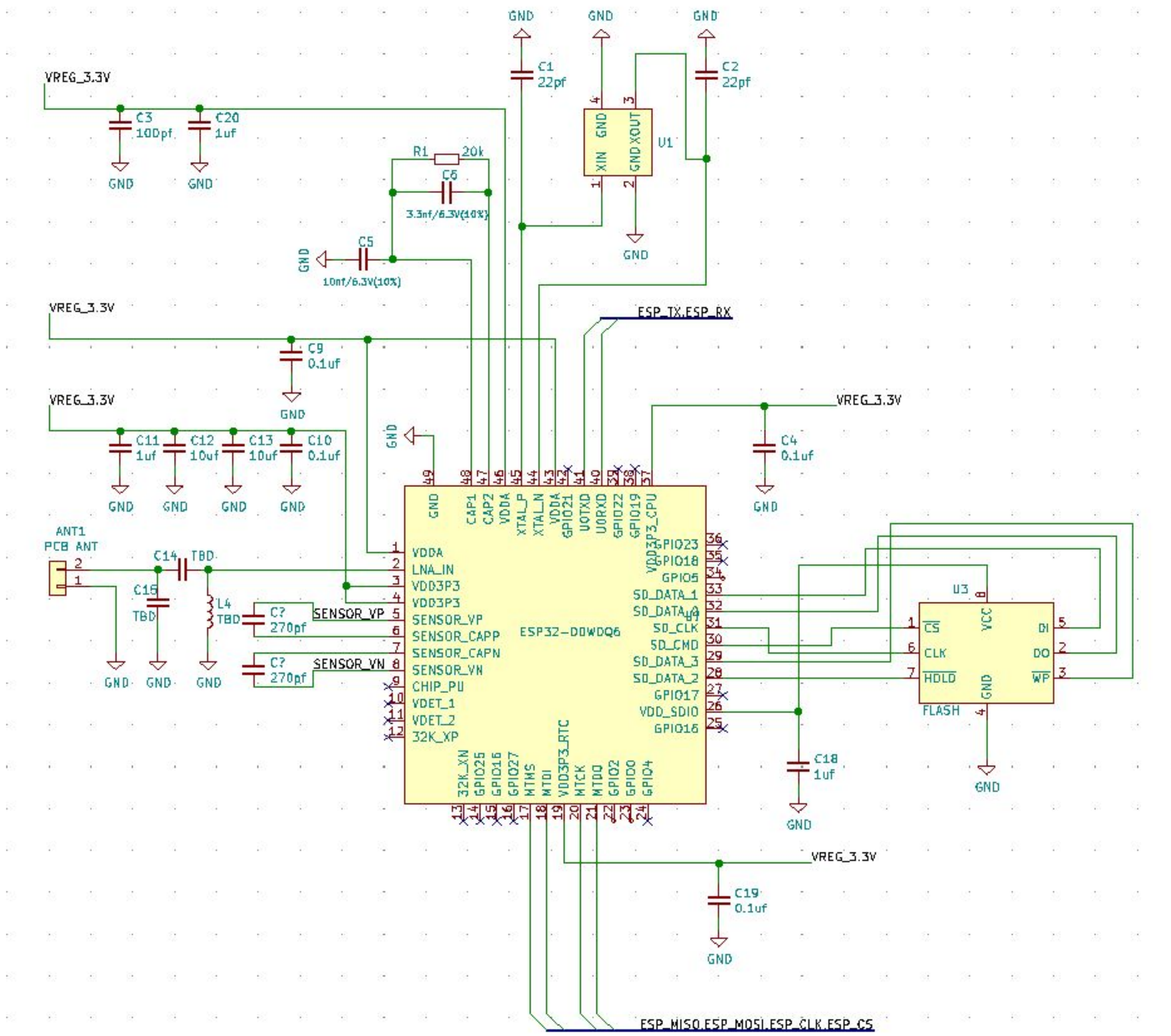

Figure 6. ESP-32 Schematic with SPI [6]

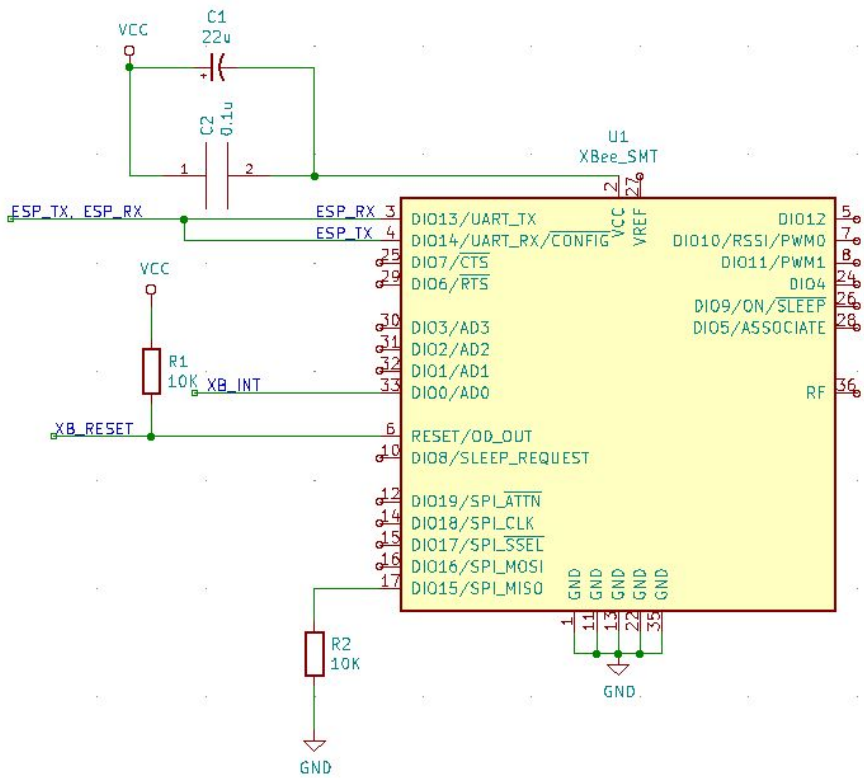

Figure 7. XBee (XB_INT is the serial input from MCU)

CONTROL UNIT

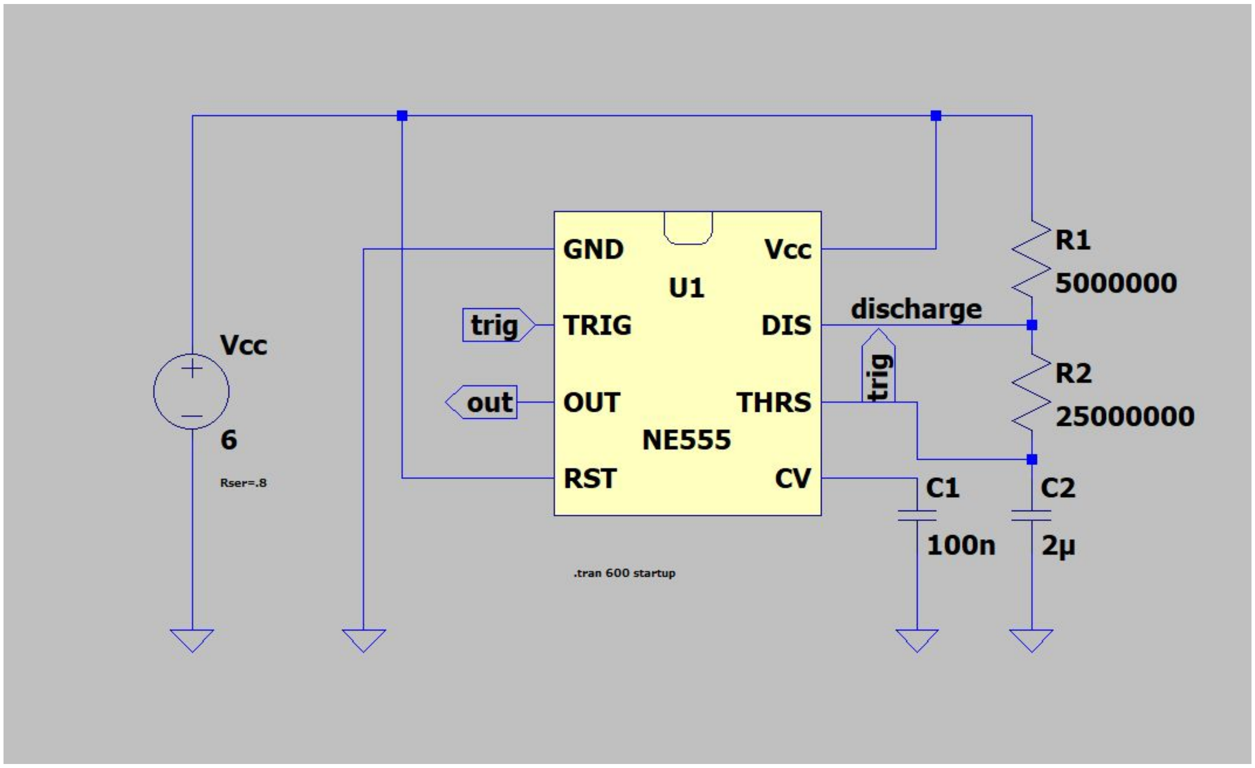

Figure 8. Repeater Circuit Diagram (Blood Pressure Cuff Activator)

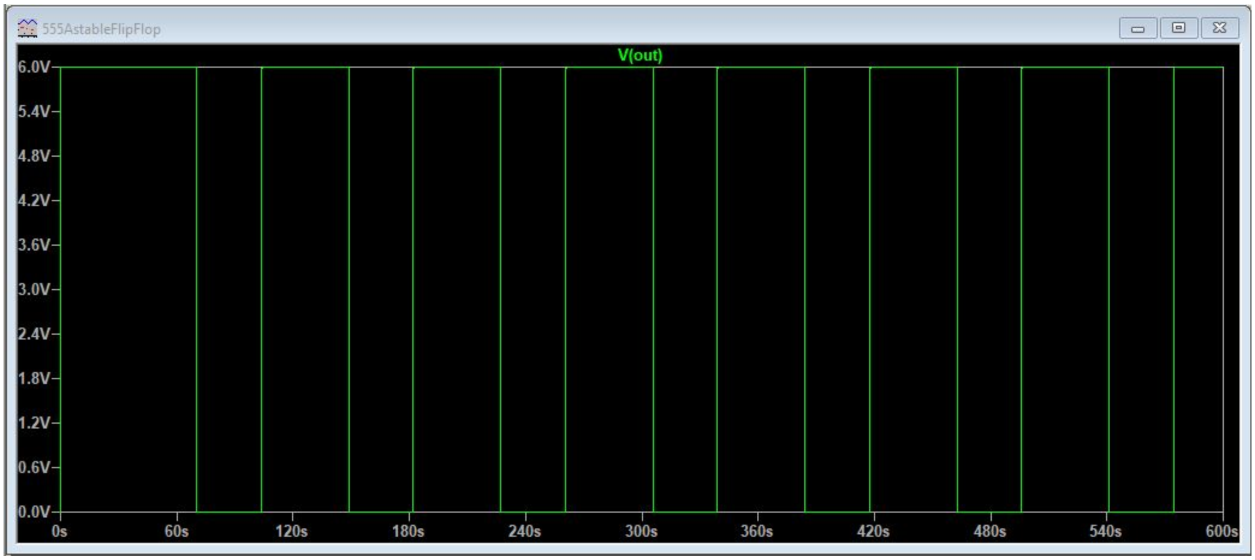

Figure 9. Repeater Circuit Simulation (Blood Pressure Cuff Activator)

Repeater Circuit Calculations:

$f = 1.38/(R_1 + 2R_2)C_2$ Hz, from NE555 data sheet

$R_1 = 5M\Omega$

$R_2 = 25M\Omega$

$C_2 = 2\mu F$

$f = 0.012077\ Hz$

$P = 1/f = 82.8\ seconds$

$Duty\ Cycle = (R_1 + R_2)/(R_1 + 2R_2)$

$Duty\ Cycle = 86\%$

2.8 Risk Analysis

Of our blocks, the piece that will pose the greatest problem will be the Data Communication block. Being able to handle a multitude of blood pressure cuffs requires either a set of unique receivers, or for means to 'mute' each cuff selectively so that only one cuff transmits data at a time, as one of our goals is to handle at least 6 cuffs, and ideally go past 6 into 16 or more simultaneous cuffs. Furthermore, if the wireless communication is unreliable, then data transferal may result in misconstrued readings, which could either manifest as extremely abnormal blood pressure, and thus inaccurately rank them in the triage list, or prevent it from communicating at all - preventing the individual from being considered in the triage list.

3. Cost and Schedule

3.1 Cost Analysis

Fix cost is estimated to be $40/hr, 10hr/week for three people.

3*$40/hr*10 hrs/wk*16 wks*2.5 = $48000

Part	Cost (Prototype)
ESP-32 (Expressif)	$15.0
XBee*2	$44.0
Cuff*2	$40.0
PCB	$5.0
TFT Screen	$37.50

HDMI/DVI to TFT Decoder	$24.95
Pluto-IIx XC3S200 HDMI	$59.95
Resistors, Voltage Regulator ICs, capacitors, LEDs	$20
Total	$241.4

Since we build 2 measurement boxes and 1 display box, this yields to a total cost of $48246.4.

3.2 Schedule

Week	Brandon	Songtao	Zihong
2/24/20	Design control system for blood pressure monitor	Begin on version1 wifi-module regulator schematics	Begin on version1 voltage regulator schematics
3/2/20	Design and test serial-to-parallel read for sensor/RF module connection	Set up ESP-32 Wi-Fi module and test its functionality with serial monitor in Arduino IDE	Test functionality of voltage regulators
3/9/20	Program FPGA for HDMI output. Test with TFT Decoder.	Set up XBee and a mesh network along with ESP-32, ensure communication between modules	Setup ESP-32Wifi network and test communication between ESP-32 and Xbee
3/16/20	Version 1 PCB Design	Version 1 PCB Design	Version 1 PCB Design
3/23/20	Attempt FPGA replacement with PCB/Modify FPGA as needed, based off of results from 3/16/20	Attempt FPGA replacement with PCB/Modify FPGA as needed, based off of results from 3/16/20	Work on data communication between ESP32 and FPGA/Attempt FPGA replacement with PCB
3/30/20	Version 2 PCB Design	Version 2 PCB Design	Version 2 PCB Design
4/6/20	Power tests	Assemble modules to create final product	Continue to work on data communication between ESP32 and FPGA
4/13/20	Power sources/boxes modifications	Collect samples to create a data chart	Bugfix on problems occurred during

			transmission between Xbee, ESP32, FPGA
4/20/20	Demo day	Demo day	Demo day
4/27/20	Demo day	Demo day	Demo day

4. Ethics and Safety

Any and all electrical components/wires on the blood pressure cuff must be fully contained inside a box/shell, as the testing will not emulate the hectic nature of the wake of a disaster, and any free wires will pose a safety concern, even if insulated. While assembling the device, the maximum voltage provided will be 6V, and the power supply will only be connected during testing to limit the probability of burning out the batteries on our skin. Working in the electronics labs carries its own challenge. We will strictly follow the guidelines in the laboratory safety training to work in pairs allowing for one individual to provide emergency response if something goes wrong. During the unit testing, we would also use an IR thermometer to ensure that the temperature of the ICs stays within safety range. We are also using an oscilloscope to monitor the output voltage from the voltage regulator to avoid burning other chips in our circuit caused by potential voltage spikes. Our testing and debugging techniques follow the IEEE code of ethics, "to avoid injuring others, their property, reputation, or employment by false or malicious action"[7]. Before using the blood pressure cuff, consulting a medical professional about the safety of an automated blood pressure cuff, as there have been reports of side-effects to their use. The delay between each reading will be done based off of this medical advice. By alternating which partner tests the blood pressure cuff on themselves, we'll further limit the chances of damaging our arm through repeated use.

Furthermore, in accordance with FDA, commercial use of this product will require adherence to SP10 regulations, involving clinical trials, advertised variability, as well as graphical analysis of the data for the user. Not using any sterile components, none of them need to be listed. The product is not legally allowed to be used as a medical device until SP10 regulations have cleared it as a blood pressure cuff.

5. Citations

- Useful IoT Resources from CS 498 IoT, Professor Matthew Caesar
  R. Santos. “Installing the ESP32 Board in Arduino IDE (Windows, Mac OS X, Linux).” Random Nerd Tutorials. https://randomnerdtutorials.com/installing-the-esp32-board-in-arduino-ide-windows-instructions/ (accessed Feb. 12, 2020).

  SM. “Getting Started with Arduino UNO.” Arduino. https://www.arduino.cc/en/Guide/ArduinoUno (accessed Feb. 12, 2020).

  Digi. *XBee-PRO 900HP DigiMesh Kit Radio Frequency (RF) Module, User Guide*. Feb. 12, 2020. [Online]. Available:https://www.digi.com/resources/documentation/digidocs/pdfs/90001496.pdf

  J. Rhysider. “XBee S2 Quick Reference Guide/Cheat Sheet and Video Tutorials to Getting Started.” TunnelsUp.com. https://www.tunnelsup.com/xbee-guide/ (accessed Feb. 12, 2020)

- Useful readings on measuring blood pressure (including non-invasive approach in development)
  [1] “QARDIOARM Smart Blood Pressure Monitor.” QARDIO. https://www.getqardio.com/qardioarm-blood-pressure-monitor-iphone-android/ (accessed Feb. 13, 2020)

  J. Beckerman. “Checking Your Blood Pressure.” WebMD. https://www.webmd.com/hypertension-high-blood-pressure/monitoring-blood-pressure#1 (accessed Feb. 12, 2020)

  [2] Y. Liu, J. Liu, Z. Fang, G. Shan, J. Xu, Z. Qi, *et al*., “Modified shock index and mortality rate of emergency patients.” Accessed: Feb. 12, 2020. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4129788/

  J. Nesselroad, V. Flacco, D. Phillips and J. Kruse. “Accuracy of automated finger blood pressure devices.” Accessed: Feb. 12, 2020. [Online]. Available: https://www.ncbi.nlm.nih.gov/pubmed/8900551

  B. Imholz, W. Wieling, G. Montrans and K. Wesseling. “Fifteen years experience with finger arterial pressure monitoring: assessment of the technology.” in *Cardiovascular Research*, June 1998. [Online]. Available: https://academic.oup.com/cardiovascres/

article/38/3/605/326615

- Other References
  [3] *ESP32 Overview*. Espressif. Accessed: Feb. 27, 2020. [Online]. Available: https://www.espressif.com/en/products/hardware/esp32/overview

  [4] Stish. "DIY Power Supply for 5V and 3.3V." Instructables Circuits. https://www.instructables.com/id/DIY-Power-Supply-for-5V-and-33V/ (Accessed: Feb. 22, 2020).

  [5] *1.5-A Low Dropout Positive Regulators datasheet*, LM1086 , Rev, J. Texas Instruments, April 2015. Accessed: Feb. 25,2020. [Online]. Available: http://www.ti.com/lit/ds/symlink/lm1086.pdf

  [6] *ESP32-WROOM-32 datasheet*, Espressif Systems, 2019. Accessed: Feb. 21, 2020. [Online]. Available: https://www.espressif.com/sites/default/files/documentation/Esp32-wroom-32_datasheet_en.pdf

  [7] ieee.org. "IEEE Code of Ethics", 2020. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: Feb. 12, 2020].

  H.K Bakke, T. Steinvik, S.I. Eidissen, M. Gilbert, and T. Wisborg. "Bystander first aid in trauma - prevalence and quality: a prospective observational study." in *Acta Anaesthesiol Scand.,* October 2015. Accessed: Feb. 13, 2020. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4744764/