Prototype 1 - Deliverable F

Nathan Gaudaur (300138966), Brandon Joseph Broderick (0300128727), Michel Pellerin (0300131059), Andrew Bui (300116223), Matthew Yakubu (300123797), Cian Brushett

March 2nd, 2020

Abstract

In this report, our group needed to investigate the riskiest assumptions that would be needed to be made in order to build our prototypes. After this, we had to come up with different prototypes that we thought would be necessary to build in order to test all the components of the device that we weren't sure about. Using these different prototypes will also help us give a better presentation to our client when we have to do that because we will have a much better idea about what works well and what doesn't with our design.

Our group got into a group to decide what the necessary assumptions and prototypes were for our design. When we met up, we decided that the necessary assumptions were that all the necessary components for the device could be held within a wrist-band, another assumption that we had to make was that we would not have any losses in power from our theoretical to experimental results, and lastly that we could program an app with all the components outlined by our third prototype.

Using these three assumptions we came up with three prototypes to make us feel more confident that the assumptions will not hurt the overall quality of the device, these prototypes were casing/formatting, where we came up with the dimensions of all the components for the device and saw that we could fit them within a wrist-band, the second prototype we designed was the calculations for the battery life of the device, and the third prototype was the interface and functionality designs for the app that we will be using to send notifications to the user of our apps emergency contacts. We also developed some code prototype that will send a notification to a user's emergency contact.

Introduction

This document will analyze and develop multiple assumptions and prototype designs. In order to come up with the necessary assumptions, we talked with each other and used the slides from lecture 11, we then used these assumptions and brainstormed to come up with a list of things that we were concerned about with the project, which we then turned into our prototypes.

Prototype Test plan

This report was created for the purpose of defining our prototype test plan and for documenting the results. Our group decided to make discrete prototypes with the goal of testing a specific and important part of the overall design. We have a total of 3 prototypes, as well as power calculations based on the requirements of each component.

The first prototype we designed was for the dimensions of the device. We analyzed each component that contributed a significant amount to the overall size of the device. We added all the components together and made a prototype that is roughly the same size as our final product. The purpose of this prototype is mainly for customer feedback, the customer can now communicate accurately with us regarding the size of the device.

Our second prototype is the app required to receive data from the Arduino via Bluetooth and send out a notification message based on the oxygen saturation level. This prototype was designed to analyze the app so we can ensure it will function properly. The app is a major part of the final product since it collects at the data and decides when to send a help message, we must have this functioning properly.

Our second prototype is the Arduino sketch required to measure the pulse oximeter and send a notification to the app we create. The purpose of creating this was for reducing the risk for later prototypes. The Arduino sketches can be complex, so we designed a prototype specifically to ensure the code is correct so our next prototype will be more accurate when we aggregate our prototypes.

The final aspect we decided to examine this device is the total power consumption. This was done so we can reduce the risk when building our second prototype. We need to ensure the battery we choose is capable of powering all of the components in our device for a significant amount of time, at least 7 hours.

1st prototype - Plan-casing/formating

Why are we doing this test?

The general objective of this test is for learning and understanding of the physical design for the casing that will house the electronic components of the device. The intent is to move from an analytically designed prototype towards a physical one. This prototype will need to incorporate the results from other design prototypes on battery size, coding and the app needed for the program interface.

Although this prototype testing is focused on the casing, it will take a comprehensive look at this aspect of our design. This comprehensive test is however at the pre-alpha stage. The main parameters that will be considered include size and functionality.

Test Objectives Description

The test objectives are to validate initial design concepts for the casing that will provide the external structure for our device.

A key objective is to evaluate the appeal of a housing design that provides the needed structure to ensure that the device runs effectively. This testing will identify what aspects of the design work well and which aspects do not. This information will allow for improvements in the next iteration of the design. The next round of testing can then be more focused.

*What **exactly** is being learned or communicated with the prototype?*

The prototype testing will help us learn which measurements for the casing are feasible. In particular, the size (length, width and weight) as well as the form of the casing for the device. If will also provide information on whether or not people will wear the device, how comfortable it will be and the likelihood that there will be uptake by the users (i.e. functionality).

What are the possible types of results?

Possible results include confirmation that the design concept works. There may also be partial results that indicate which components of the design are acceptable and which are not. For example, the design may confirm that the weight is acceptable however the device is too bulky. Worst case, none of the design concepts are feasible and the result is that we have to look at alternative options for our product (ie pocket device vs something worn on the wrist).

Feedback will provide improvement ideas that can be used for adjustments that will be included in the next prototype.

How will these results be used to make decisions or select concepts?

The results from the prototype testing will identify aspects of the design that work (are favourable) and other aspects that are not functional. The aspects that work will be retained. Aspects of the design that do not are the aspects of the casing that will need to be modified.

The main attributes to me tested are weight, size (length, width and amount of space the casing occupies), bulkiness (i.e. protrude from the wrist when worn) and comfort. We may possibly look at the heat generated.

The prototype testing will also consider the location of the device and whether or not it catches on clothing or hits against things when worn.

What are the criteria for test success or failure?

This is a pre-alpha testing that is intended to evaluate the casing parameters when housing the various components of the device (i.e. battery, program app and code).

Criteria for success is that the casing for the device is within a 30% margin for existing specifications for wrist-worn devices (i.e. apple watches, fitness and GPs watches). (These specs to be determined from market research of existing products). In particular, success will be determined as to whether or not the weight, length and width fall within this 30% margin. A qualitative evaluation will be given to the contour and bulkiness and how similar the shape is to existing wrist-worn devices.

The next round of testing will be with a sample of the targeted audience (blue-collar workers). Overall success would then be having results that indicate that 70% of users would wear the device and find it useful.

What is going on and **how** is it being done?

*Describe the prototype **type** (e.g. focused or comprehensive) and the reason for the selection of this type of prototype.*

Comprehensive testing is being undertaken of the casing design. This is because we are in the early stages of the design and this is the first jump from an analytical model to a more physical one. It is also the first time that the other components of the design (batteries, code, program app) are being integrated.

A comprehensive approach was chosen because it allows us to take an overview of all the different components and make adjustments as needed before becoming too detailed in specific areas of the design process. Ideally, this should allow us to identify critical flaws (i.e. whether or not a wrist-worn device feasible given the components needed to operate the device) and correct early in the design process.

Describe the testing process in enough detail to allow someone else to build and test the prototype instead of you.

Concept drawing with dimensions can be programmed into a 3D printer. This will provide information on the height, width and overall bulkiness of the design. The weight can be scaled based on the internal components needed for the other design areas (i.e. batteries). The 3D printed model can then have weights added to it.

This weighted model can then be strapped to the wrist of the design team to determine the functionality of the design parameters described above. Measurements can be taken and compared to similar types of devices. In addition, measurements can be taken as to how far the device protrudes outside of the wrist of the user.

Once the parameters of the initial design have been taken, the casing will need to be optimized. For example, a critical component is whether or not the device can be designed with a lighter weight.

*What information is being **measured**?*

The protocol testing will measure the physical design characteristics of the device. In particular the weight, length, height and depth, shape (i.e. contour on the wrist) and bulkiness.

*What is being observed and how is it being **recorded**?*

In addition to the measured parameters, information on the shape and bulkiness are also being recorded. This information will provide information on the position of the device on the user's wrist and look at whether or not it protrudes and if it catches on things like clothing. This information is important for assessing the overall fit of the device.

What materials are required and what is the approximate estimated cost?

A $100 budget has been provided for all four testing protocols. The casing is made of plastic that must be durable to house the following components:

- Arduino Nano
- 1 double AA battery case that holds 4 batteries
- 1 AA battery case that holds 2 batteries
- Arduino Bluetooth module (J-DEAL® HC-05 Wireless Bluetooth Host Serial Transceiver Module Slave and Master RS232 For Arduino)
- Pulse oximeter sensor (Lysignal MAX30100 Pulse Oximeter Heart Rate Sensor Module for Arduino for Wearable Health Fitness Assistant Devices Medical Monitoring Devices)
- Copper wire

This casing will be the cheapest of the 4 testing protocols. Most of the expenses will be for the plastic resin used in the 3D printer.

What work (e.g. test software or construction or modelling work or research) needs to be done?

Research needs to be done on the following areas:

- Fabrication process and how this can be done while ensuring materials remain lightweight.
- Average design specifications for other wrist devices to determine the weight and size ranges for the variety of products already on the market.
- Plastic materials that can be used to house the device with consideration to weight
- Use of a 3D printer for modelling

When is it happening?

*How long will the test take and what are the **dependencies** (i.e. what needs to happen before the testing can occur)?*

The casing design is dependent on the fabrication of the internal components (i.e. battery, coding, program app). The first prototype will need to be sufficient to house these other design parameters.

Once the initial design has been determined the next step is to optimize to minimize weight, size and bulkiness, while still ensuring maximum functionality. The ability to optimize also needs to be compared to the specifications of existing devices to determine if the results are within a 30% range of what has been deemed acceptable for existing devices

2nd prototype - Battery life

These are the 3 main aspects of our device that will draw from the battery. We must find a battery that is both suitable to power our device and also to fit inside of our device without having a big effect on the device's size.

We have two options when it comes to choosing a battery to power the module:

- A rechargeable battery with a good and efficient charging method.
- A disposable battery that is cheap and easy to change. This battery would preferably last a decently long time.

Power Consumption of the Main Devices:

- Arduino Nano:
    - Power consumption 19 mA

- Bluetooth Module:
    - Power consumption 30 mA

- Blood Oxygen Sensor:
    - Power consumption 20 mA

All together these 3 parts of our device have a power consumption of roughly 69 mA. Our goal is for our device to run for at least 16 hours before needing to be recharged. This allows the user to power the device for a full day without needing to recharge the device. With a draw of roughly 69 mA, the battery used in the device should have a battery capacity of at least 1100 mAh.

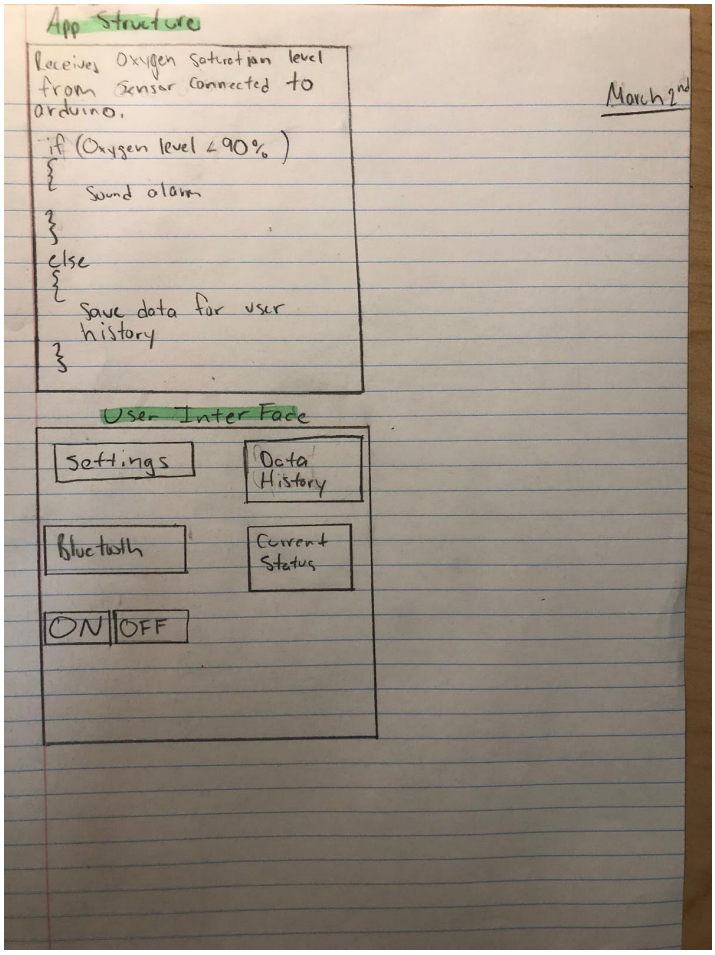
We need to find a battery that fits our device best. We need to find a balance between all the aspects of the battery.

- Size
- Cost
- Battery capacity
- Rechargeable or non-rechargeable
- Charging method or replacement method

3rd prototype - App Structure

The purpose of building this prototype for the app is to ensure we have all the necessary components to have it functioning properly once it is built. This prototype will be a focused prototype that will target the app structure and user interface design.

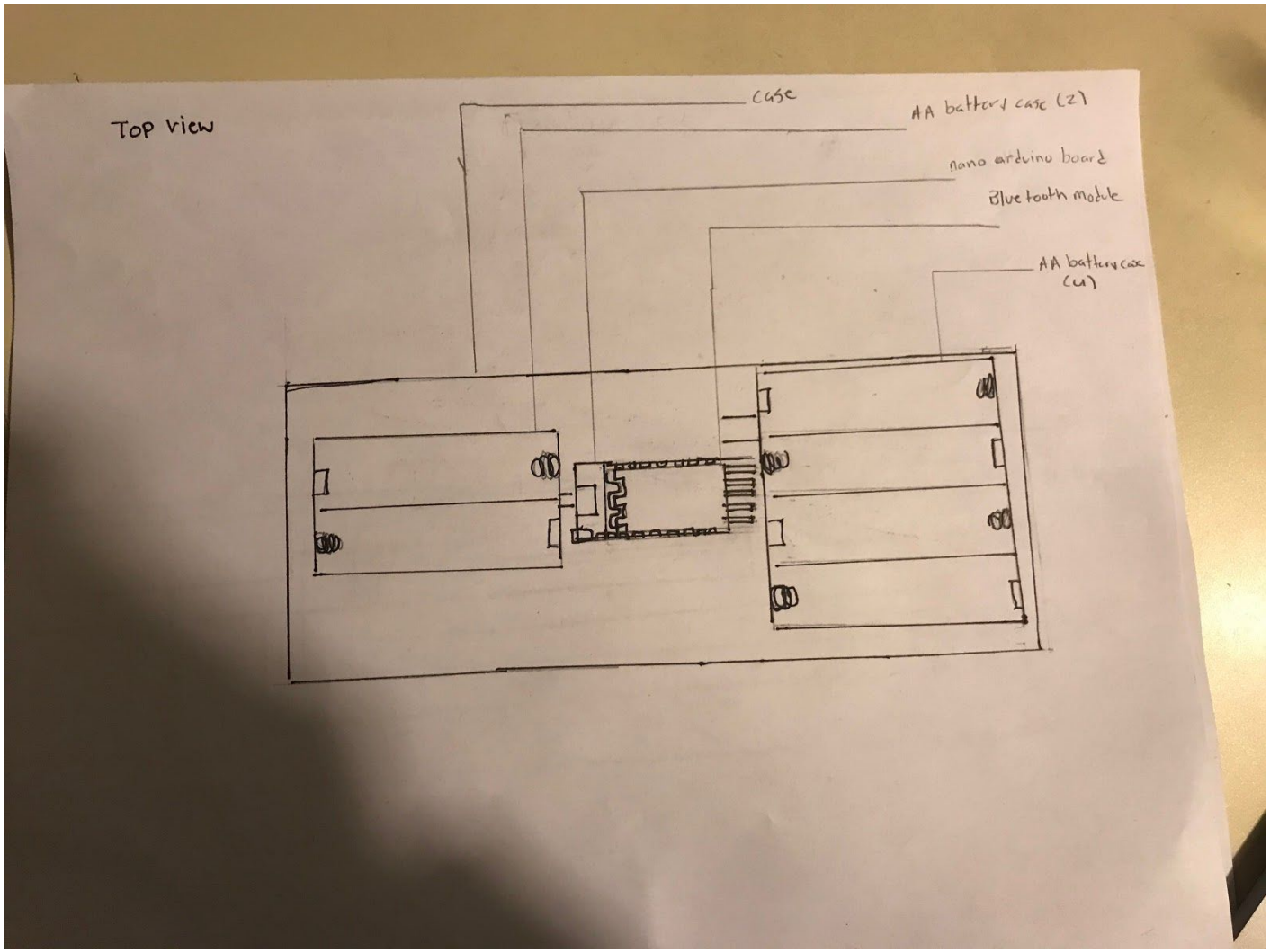
As the image illustrates below, the app will receive data from the Arduino and determine when to sound an alarm or send a notification for help. Based on research the oxygen saturation level should never be below 90%, so this will determine when someone is having an overdose. The Arduino will be programmed to send new data readings from the sensor every 30 seconds, all the sensor readings that are greater than 90% will be saved. The data will be saved so users will be able to look back throughout the course of a day and see if there are any suspicious signs of anything.



The user interface is the component of this app that the user will interact with. There will be a settings component for the user to change their preferences. There will be a button that the user presses to see the sensor readings history throughout the day. There will be a current status component that the user presses to see up to date readings. Then there will be a BlueTooth component so the user can see when a device is connected or not.

This prototype displays all of the components believed to be necessary for the app to function properly. The results show that if we can successfully build this app then the communication for help will work as we plan it to.

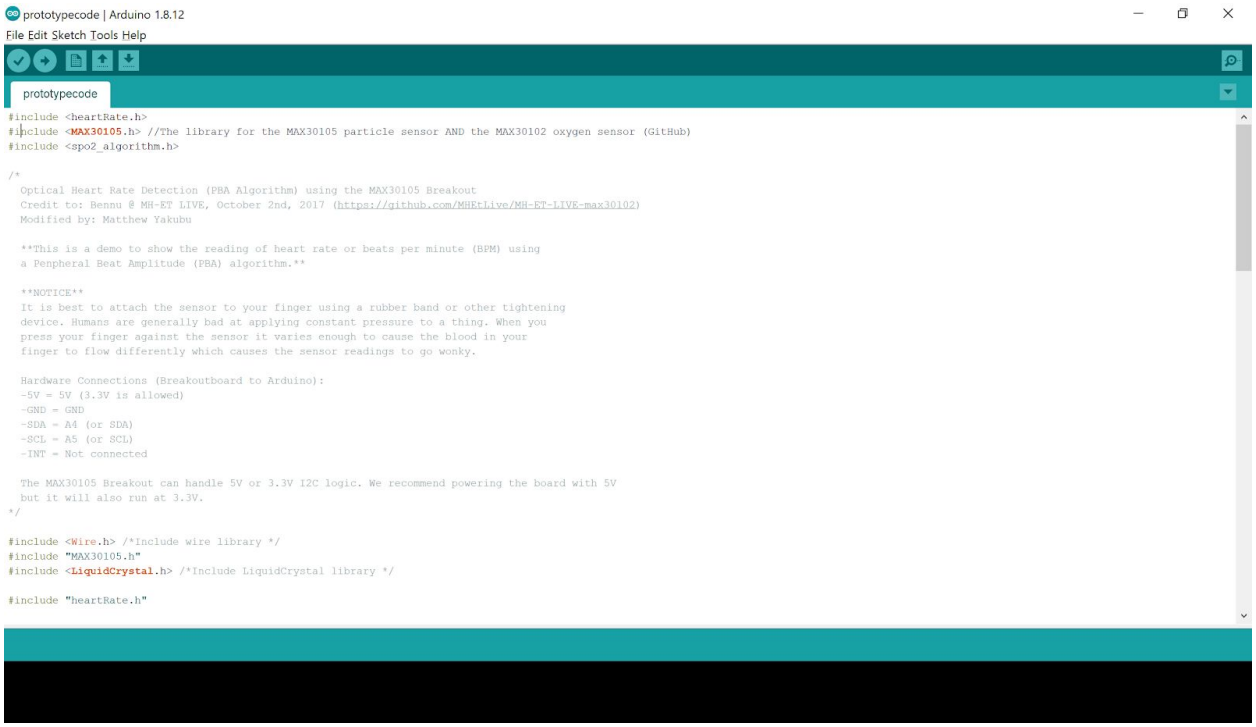
Design prototype1 :

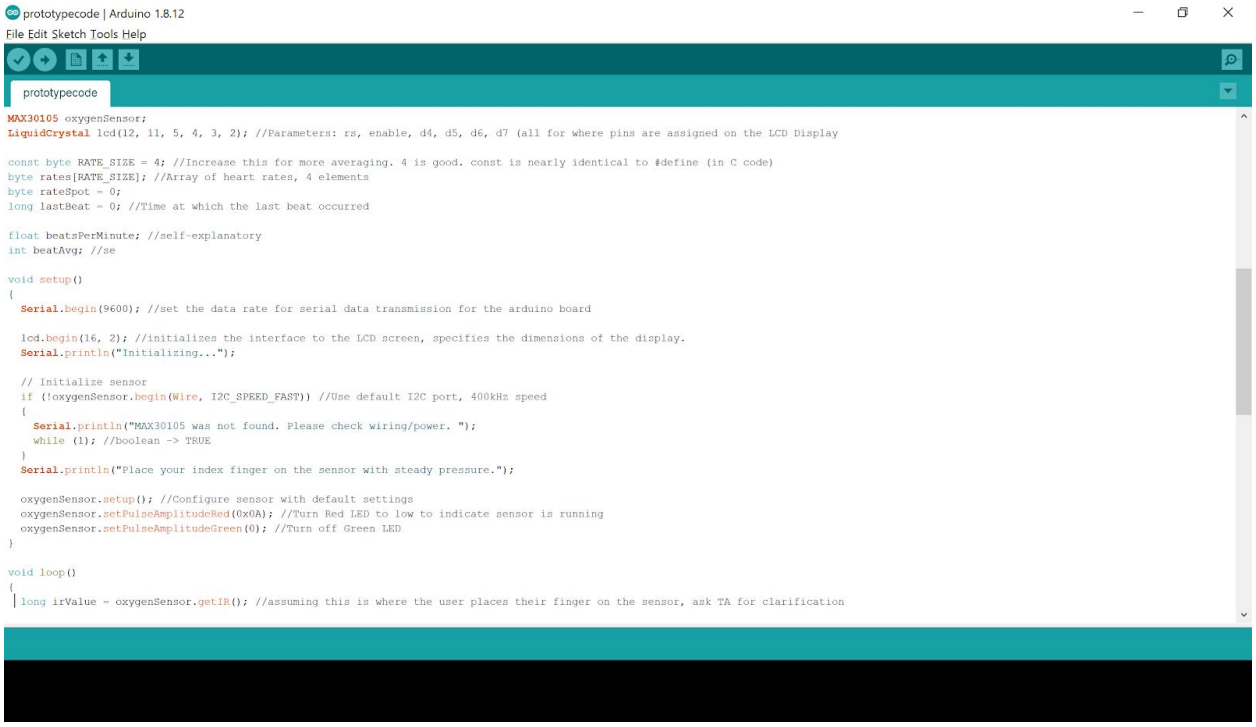

Top view
case
AA battery case (2)
nano arduino board
Blue tooth module
AA battery case (4)

Arduino Sensor Code Prototype

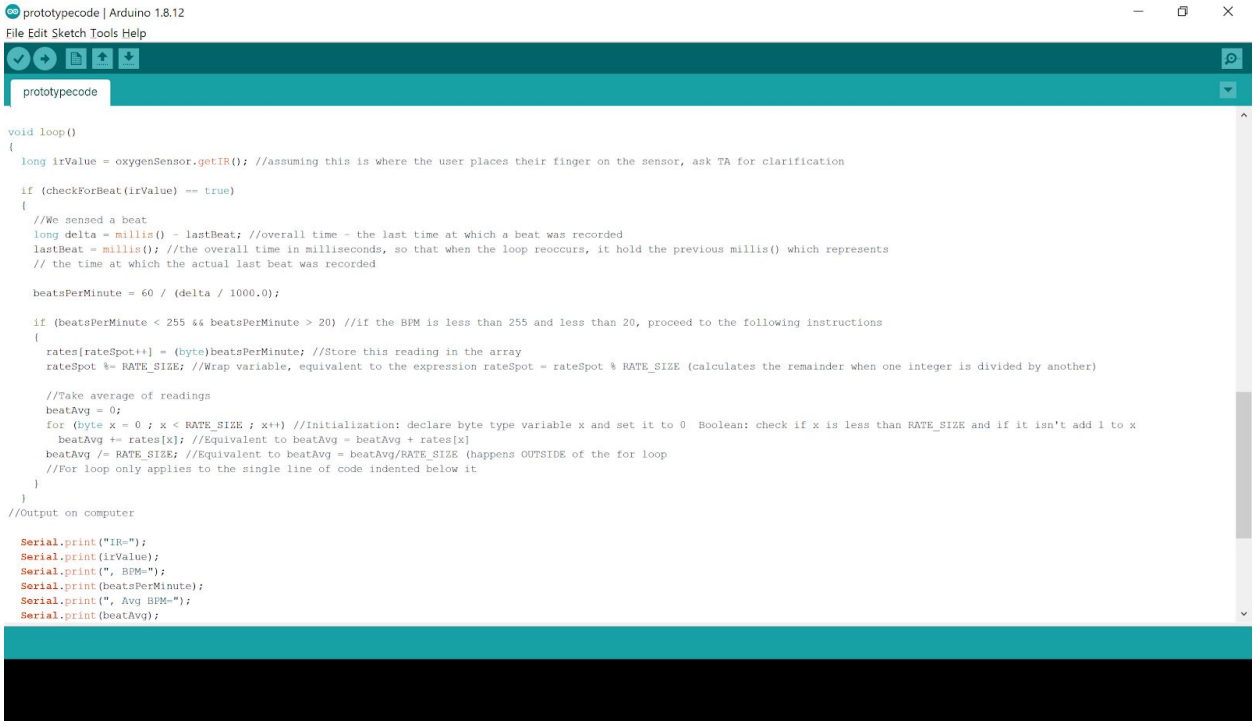
The purpose of this code is to show the reading of heart rate or beats per minute using a Penpheral Beat Amplitude (PBA) algorithm. This will allow us to see if there are any
This will be tested by running the compiled code on any computer device that can run Arduino confidently,

Here is the following code:

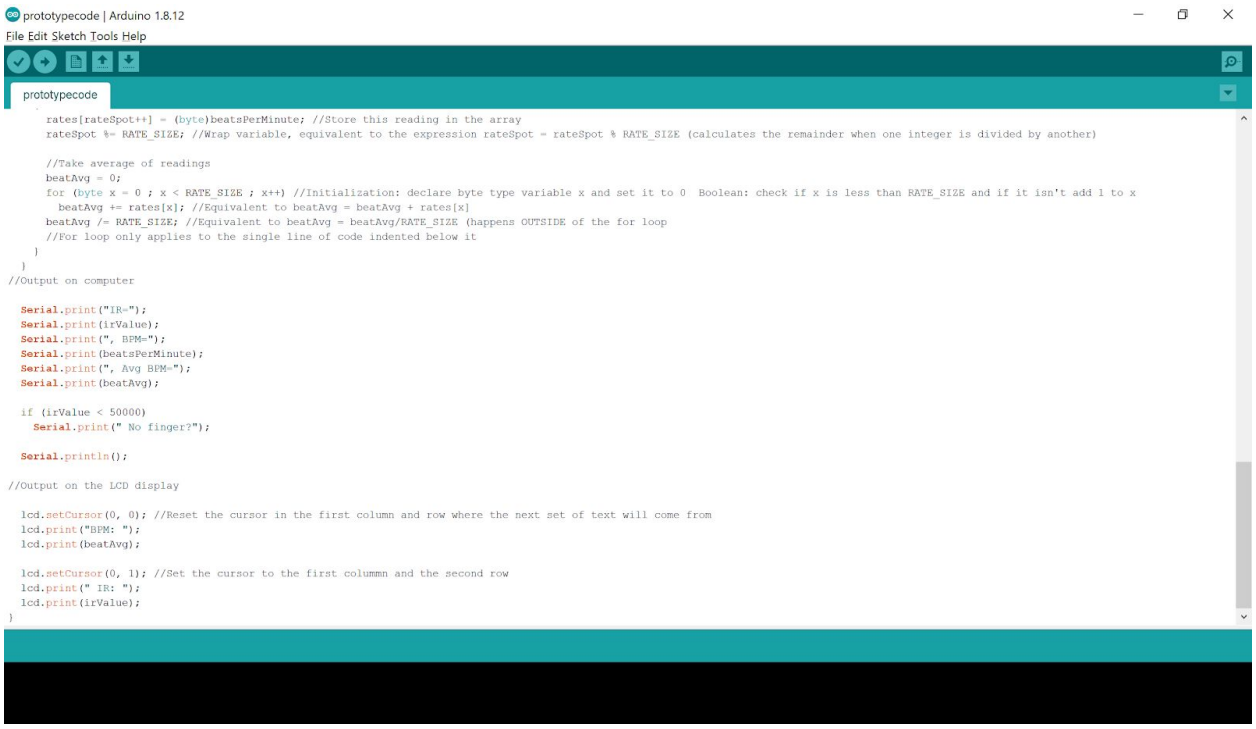





```
MAX30105 oxygenSensor;
LiquidCrystal lcd(12, 11, 5, 4, 3, 2); //Parameters: rs, enable, d4, d5, d6, d7 (all for where pins are assigned on the LCD Display

const byte RATE_SIZE = 4; //Increase this for more averaging. 4 is good. const is nearly identical to #define (in C code)
byte rates[RATE_SIZE]; //Array of heart rates, 4 elements
byte rateSpot = 0;
long lastBeat = 0; //Time at which the last beat occurred

float beatsPerMinute; //self-explanatory
int beatAvg; //se

void setup()
{
  Serial.begin(9600); //set the data rate for serial data transmission for the arduino board

  lcd.begin(16, 2); //initializes the interface to the LCD screen, specifies the dimensions of the display.
  Serial.println("Initializing...");

  // Initialize sensor
  if (!oxygenSensor.begin(Wire, I2C_SPEED_FAST)) //Use default I2C port, 400kHz speed
  {
    Serial.println("MAX30105 was not found. Please check wiring/power. ");
    while (1); //boolean -> TRUE
  }
  Serial.println("Place your index finger on the sensor with steady pressure.");

  oxygenSensor.setup(); //Configure sensor with default settings
  oxygenSensor.setPulseAmplitudeRed(0x0A); //Turn Red LED to low to indicate sensor is running
  oxygenSensor.setPulseAmplitudeGreen(0); //Turn off Green LED
}

void loop()
{
  long irValue = oxygenSensor.getIR(); //assuming this is where the user places their finger on the sensor, ask TA for clarification
```




```
void loop()
{
  long irValue = oxygenSensor.getIR(); //assuming this is where the user places their finger on the sensor, ask TA for clarification

  if (checkForBeat(irValue) == true)
  {
    //We sensed a beat
    long delta = millis() - lastBeat; //overall time - the last time at which a beat was recorded
    lastBeat = millis(); //the overall time in milliseconds, so that when the loop reoccurs, it hold the previous millis() which represents
    // the time at which the actual last beat was recorded

    beatsPerMinute = 60 / (delta / 1000.0);

    if (beatsPerMinute < 255 && beatsPerMinute > 20) //if the BPM is less than 255 and less than 20, proceed to the following instructions
    {
      rates[rateSpot++] = (byte)beatsPerMinute; //Store this reading in the array
      rateSpot %= RATE_SIZE; //Wrap variable, equivalent to the expression rateSpot = rateSpot % RATE_SIZE (calculates the remainder when one integer is divided by another)

      //Take average of readings
      beatAvg = 0;
      for (byte x = 0 ; x < RATE_SIZE ; x++) //Initialization: declare byte type variable x and set it to 0  Boolean: check if x is less than RATE_SIZE and if it isn't add 1 to x
        beatAvg += rates[x]; //Equivalent to beatAvg = beatAvg + rates[x]
      beatAvg /= RATE_SIZE; //Equivalent to beatAvg = beatAvg/RATE_SIZE (happens OUTSIDE of the for loop
      //For loop only applies to the single line of code indented below it
    }
  }
//Output on computer

  Serial.print("IR=");
  Serial.print(irValue);
  Serial.print(", BPM=");
  Serial.print(beatsPerMinute);
  Serial.print(", Avg BPM=");
  Serial.print(beatAvg);
```



The libraries required for this code include:

- Wire.h - the communications library
- MAX30105.h - the library used for the oxygen sensor
- LiquidCrystal.h - used for the display on the LCD

setup()

- For the setup, we initialized the data rate for serial data transmission for the arduino board,
    - as well as initializing the interface to the LCD screen and specifying the dimensions of the display.
- oxygenSensor.setup()
    - Configure the oxygen sensor with default settings
    - Turn the Red LED to low to indicate that the sensor is running and turn off the the green LED

loop()

- Call the function to get a sensor reading from the user and assign it to a variable (in this case the variable is **long** irValue)
- Check to see if there was a reading (boolean if statement to check if true or false)
    - If true
        - Overall time - The last time at which a beat was recorded

        - Convert to millis(), so that the loop reoccurs, it holds the previous millis() value which represents the time at which the actual last beat was recorded
- If the BPM reading (formula used in the code) is within 20-255:
    - Store the reading within an array - **rates []**
    - **beatAvg** - Find the average of readings
- for loop
    - Declare byte type variable **x** and make it equal to 0
    - Check if x is less than **RATE_SIZE** and if it isn't, add 1 to x

Output on computer

- Overall outputs a display on the LCD to show the IR and BPM readings

Conclusion

In conclusion, this report discussed our prototype test plan, gave physical examples, and the results from each prototype. Based on the content in this report our group is keeping on track with the deadlines and should be continuing to do so. We have all of the outlined details necessary to make a successful second prototype.