University of Moratuwa
Faculty of Engineering
Department of Electronic & Telecommunication Engineering

EN 1190 : Engineering Design Project

Group EN-13

Apnemeter (*Sleep Apnea Detector*)

Group Members	
Index No.	Name
200345N	Liyanaarachchi D.S.G.L.S
200352H	Liyanage P.H.S.
200356A	Lokugeegana D.L.
200358G	Luckshan G.W.C.M.

Contents

Abstract

A person can be diagnosed for the sleep apnea condition by undergoing a test called "polysomnography". But this test is quite costly. The minimum requirement needed to determine the apnea condition is measuring the average number of apnea instances occurred per hour. A cost effective device can be designed to calculate it, by only considering the heart rate, oxygen concentration level and the snoring pattern. The *MAX30102* and *MAX4466* modules are used to obtain the signals and they are analyzed to determine whether the user might have an apnea condition. The results obtains are displayed to the user via an mobile application in a user friendly manner.

1 Problem Description

1.1 Problem

Snoring while sleeping is a common condition that we see in the society. **Sleeping apnea** is a condition that occurs due to blockage of airways to the lungs and the body is deprived of oxygen during sleep.

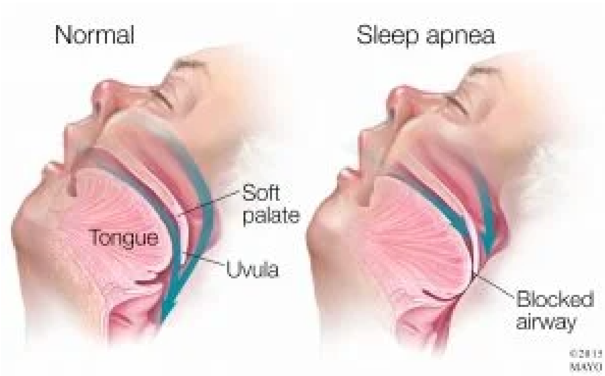


Figure 1: *Sleep Apnea*

Statistics show that nearly 25 million people in USA suffers from obstructive sleep apnea.(American Academy of sleeping Medicine)

Sleep apnea is a serious condition even though people usually neglect it. This would have serious effects ranging from depression & anxiety, hypertension, cardiac arrhythmia, strokes to even a sudden cardiac death if undiagnosed.(Journal of the American college of cardiology)

The current diagnostic method for this condition is sleep polysomnography which is a bulky device and requires to be admitted to a hospital or a place with a trained technician. The high cost of the device makes could not be used to personally collect data for a longer period of time.

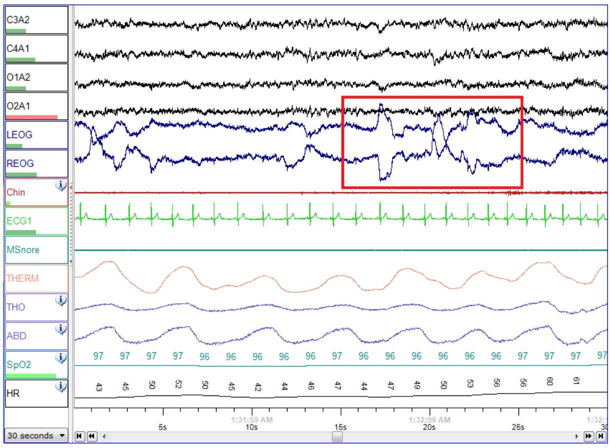


Figure 2: *Polysomnographic record of REM sleep.*

1.2 Arriving at the solution

The above mentioned inconveniences make people reluctant to be tested hospitalized, so there is a vacancy for a smaller, easy to use device at home that could be used as an initial test, and then decide whether the severity of the condition requires further testing.

When we conducted further research on the sleep polysomnography, we found out that it monitors many body functions, including brain activity (EEG), eye movements (EOG), muscle activity or skeletal muscle activation (EMG), and heart rhythm (ECG) as shown in the figure 2 above.

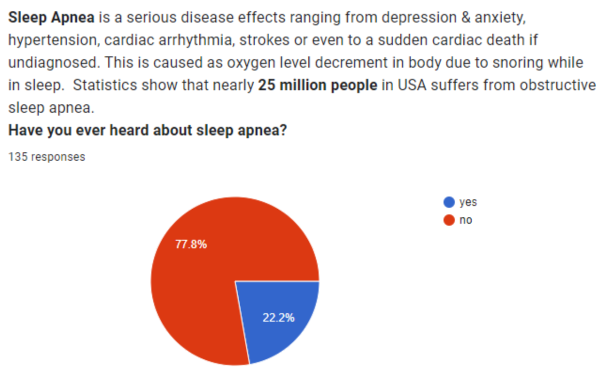
But, taking only the measurements of heart rate, SpO2 level(the percentage of oxygen in someone's blood) and sound level is enough for an initial detection of the condition according to the definition of *Sleap apnea*. If SpO2 level is less than a certain threshold value for more than 10 seconds, that time period can be considered as an apnea instance. If more than 5 of such instances occur within an hour, we can come to a conclusion that the person has a very high probability of having the condition hence he needs to get tested at a hospital and phisician could see three graphs via the clould dash board regarding SPO2 levels, heart rate and sound level which can analyse by him same as the way he analyse polysonography data.

Therefore, as a solution to the previously mentioned problem, we proposed to design a budget friendly, easy to use device that can be used at home, by anyone to conduct an initial test for the sleeping apnea condition(sleep apnea instance per hour). This product checks the minimum requirement that should be fulfilled by a patient to go for a final, more complicated testing at a hospital.Since this can be reused by multiple people at a household, it saves the time, money and effort of a healthy person that would go in vain if he was to get the test at a hospital without a pre-test.

1.3 Solution validation

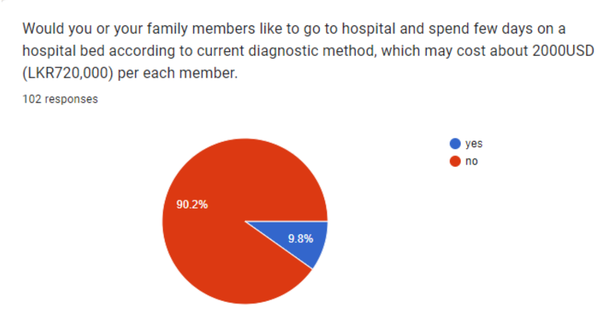
We have conducted a research whether people of Sri Lanka is aware about the diseases. Also whether this device is affordable for them to get an initial diagnosis. The conclusions from the results are as follows,

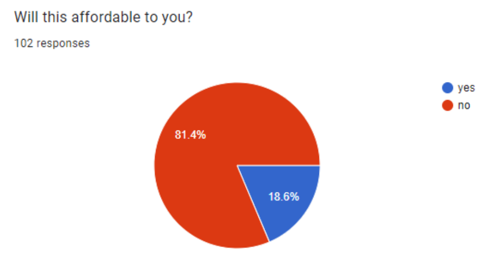
From the 135 sample we conducted the survey, 108 of them were not aware of Sleep Apnea. This shows that most of the people doesn't know about the disease.



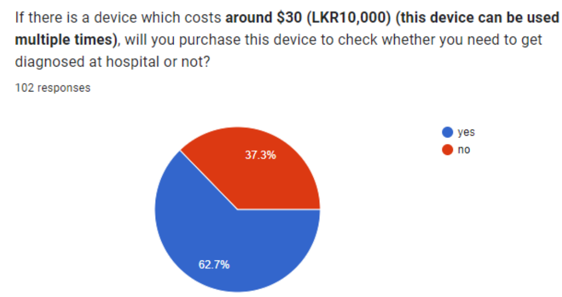
In the survey we conducted we gave a brief description about how much serious this disease is, then most of people in the sample are willing to get diagnosed. This shows us that if people were aware of the disease they might get diagnosed. For marketing purpose, we might be able to put some videos about sleep apnea to YouTube for people to get informed.

From the sample of people who are willing to get diagnosed (108), 81% of them cannot afford the current diagnosis method of Sleeping Polysomnography. Therefore, only 83 from the sample may consider whether or not to buy our product.





The device which we are making will cost around than LKR 10,000 (30) and 62.7% of the people who are not able to diagnose Sleep Polysomnography will but our product. This means our product will save about 64 more lives from a sample of 135. This means market for our product will be automatically created if we informed the people in a proper method.



2 Technical Feasibility

The function of our device is to detect the condition known as sleep apnea. The two main measurements needed to identify the minimum requirement is the oxygen concentration and heart rhythm.

- The **MAX4466 module** was selected to detect the sound level since it had the ability to detect sound through the mic and amplify it through the pre-amplifier.

- In order to obtain the reading of heart rate and oxygen concentration, **MAX30106** module was used.

Once the measurements have been obtained, the data needs to be analyzed to detect the disease condition. Hence a microcontroller is needed for this purpose. Since the device is connected to the patient when he/she goes to sleep we have to make sure that the power usage is low, size is minimum and that there are no complications with the pressure. Considering those requirements, we decided to use the **ATmega328p** micro-controller which is technically feasible.

This device takes measurements when the patient is sleeping. It means that the device should be capable of working throughout the night, taking measurements of heart rate and oxygen concentration while detecting the snoring sound as well. Therefore, power should be supplied continuously for 5-8 hours if the device was to function properly. For that, we are using a **3.7 V lithium polymer battery**. This voltage was sufficient since we have selected modules that consume least power. Even though our initial plan was to use TP4056 battery controller to control the power, due to the unavailability of resources in the country we decided to use the **TP5100** module.

Since the data is recorded during the patient's sleep, he/she cannot see the results at the same time. Therefore, there was no point in displaying the results through an LCD display. Hence, the recorded data must be stored somehow, to be reviewed later.

Our first idea was to use a SD card module for this purpose. Since SD card modules usually consume a lot of power and consumes a lot of memory capacity, using that seemed quite challenging. So we decided to send the data to a mobile app through a Bluetooth module.

3 Product Architecture

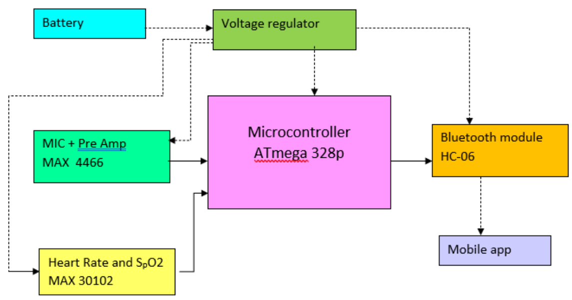


Figure 3: *Block Diagram*

3.1 Control PCB with ATmega328p Micro Controller

Atmel ATmega328p is a 8 bit Microcontroller with very low power consumption. A clock circuit of 16MHz is suggested to use with it. The microcontroller is equipped with 32KB of Flash memory, 1KB of EEPROM, 2KB of internal SPRAM. The PCB also comes with power management components as some modules run on different voltages.

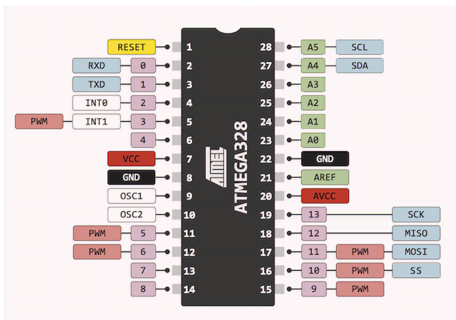


Figure 4: *ATmega328p*

3.2 MAX30102 Heart rate and pulse Oximeter Sensor

MAX30102 is a sensor that could be used to measure Heart Rate and Blood Oxygen Saturation. The module comes with two LEDs of IR and red, specialized optical elements, and filters. Heart rate is measured by the changing waveform of refracted IR light due to pulses. Oxygen saturation is measured by the relative intensities of refracted IR and Red lights as IR light is more absorbed by oxygenated blood red light is more absorbed by deoxygenated blood.

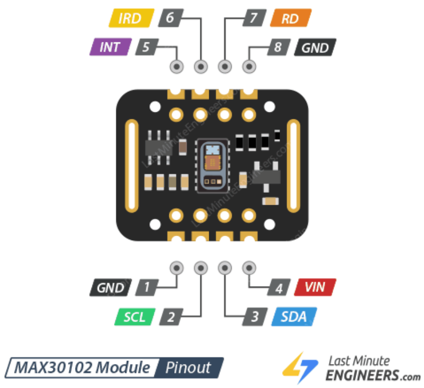


Figure 5: *MAX30102*

3.3 MAX4466 MIC & PreAmp Module

MAX4466 is a module with inbuilt microphone and max4466 op amp IC based pre amplifier. Despite the very low power consumption and the extremely small package size the device is capable of giving adjustable gain up to A_{VOL}=125dB max.

Figure 6: *MAX4466*

3.4 TP5100 Battery Controller

TP5100 is a double switch buck 8.4V, single cell 4. 2V lithium battery charge management chip. Its ultra-compact QFN16 package and simple external circuit, making TP5100 ideal for portable equipment so large current charging management applications. TP5100 built-in input overcurrent, undervoltage protection, over temperature protection, short circuit protection, battery temperature monitoring, reverse battery protection.

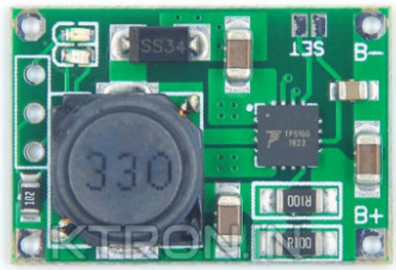


Figure 7: *TP5100*

3.5 HC-06 Bluetooth module

HC-06 is a Bluetooth module designed for establishing short range wireless data communication between two microcontrollers or systems. The module works on Bluetooth 2.0 communication protocol and it can only act as a slave device. This is cheapest method for wireless data transmission and more flexible compared to other methods and it even can transmit files at speed up to 2.1 $Mbps$.

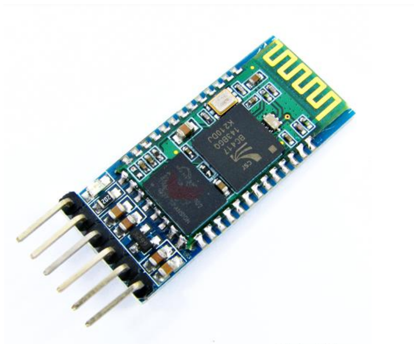


Figure 8: *HC-06*

3.6 Battery

Two 3.7V Lithium polymer batteries are to be used but thin Li-po battery is hard to find in the market and we are using 18650 batteries as alternative, if thin Li-po battery is used device height and weight will reduce in a significant amount.

4 Designs

4.1 PCB Design

PCB design was done using Altium software. Generally in order to minimize the power consumption we need to reduce the resistance of the connections. As a solution, width of the trace is increased. There is a trade off between the size of the PCB and the resistance of the traces. Therefore gap of 1 mm should be maintained. Although the ground needs a wider trace we have used a trace of 1mm because the estimated power consumption of our circuit is comparatively lower.

4.1.1 Schematic Design

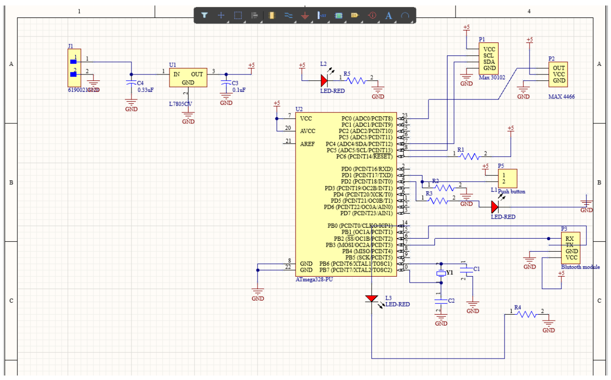


Figure 9: *Schematic*

The components that are needed to built the circuit is as follows

Componants	**Quantity**
Atmega328p Micro controller	1
MAX30102 Heart rate and pulse oximeter sensor	1
MAX4466 MIC and PreAmp	1
HC-06 Bluetooth module	1
7805 Voltage Regulator	1
18650 Single Battery holders	2
18650 Battery (3200 mAh)	2
TP5100 Battery charger	1
LEDs	2
0.33 μF electrolytic capacitors	2
22 pF ceramic capacitors	2
16 MHz Clock	1
1 K resistors	4
10 K resistors	1
ON/OFF switch	1
Push(ON) button	1

4.1.2 CAD Design (PCB)

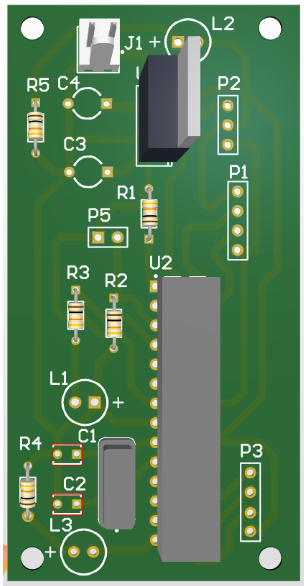


Figure 10: *Schematic*

4.2 Enclosure Design

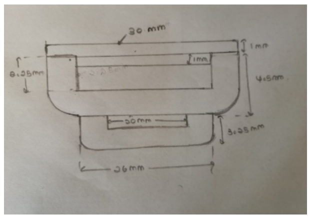
The enclosure is designed using the **Solidworks** software.
The enclosure has two parts:

- MAX30102 Module container(This is worn on the thumb to detect the SPO2 level and heart rate because the accuracy of measurements is relatively high when worn of the thumb)

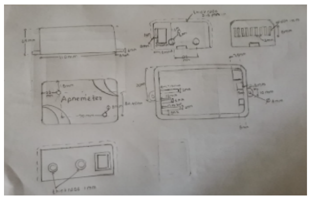
- PCB and the other parts container (This is worn around the bicep because the MAX4466 module must be located closer to the mouth.)

Within the enclosure the PCB designed using Altium is placed considering the relavent locations of the buttons, LEDs and other modules that needed to be connected to the PCB.

4.2.1 Finalized Sketches



(a) Enclosure 01

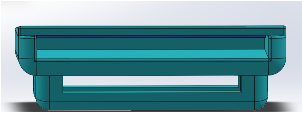


(b) Enclosure 02

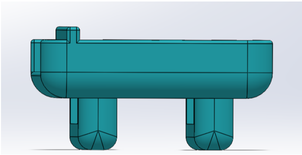
Figure 11: Finalized sketches

4.2.2 CAD Design

Part 01



(a) Right side view



(b) Left side view

Figure 12: Enclosure 01

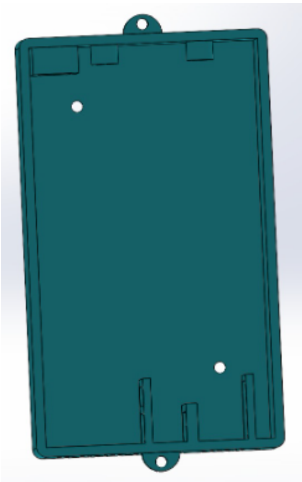


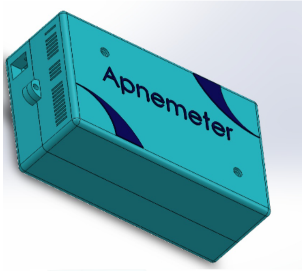
Figure 13: *Inside view of the top enclosure*

The final assembly which consists of the PCB and the two halves of the enclosure which encapsulates it is as follows

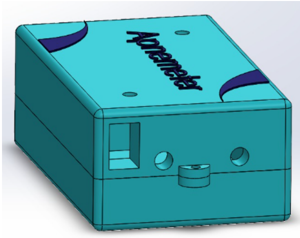
4.3 UI Design

Icons:

- Connect-select the bluetooth module of the apnemeter
- Start- start reading the data and send to the cloud
- Pasuse- pause reading data and sending data to the cloud



(a) Top view of the Enclosure



(b) Side view of the Enclosure

Figure 14: Enclosure Part 02

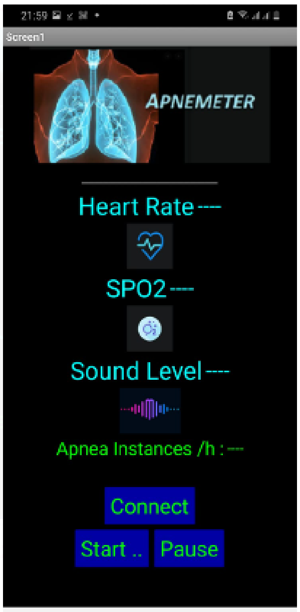


Figure 15: *Mobile App interface*

Figure 16: *Cloud Dashboard UI*

5 Methodology

5.1 Functionality

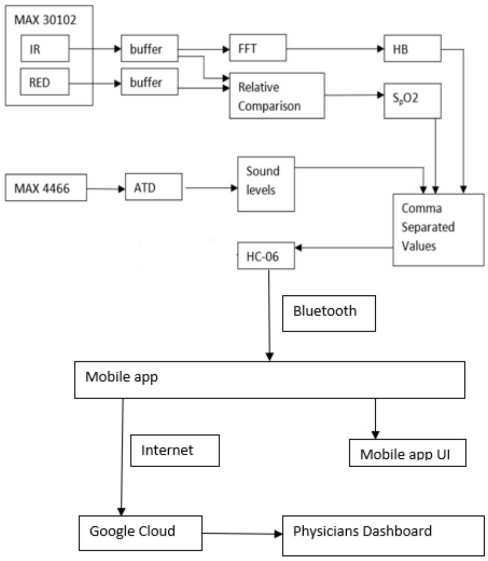


Figure 17: *Methodology*

The enclosure consists of 2 parts. The user must wear the oxygen detector on the finger and the other part on the upper arm. The MAX30102 module, which is worn on the finger gives 2 signals obtained from IR rays and RED light rays. Once the signals are obtained, the arduino code analyzes the two signals. It will send the IR ray through a buffer and then apples fast fourier transform to get the amplitude response. The maximum response is the heart rate. The IR and red signals are sent through a buffer and subjected to relative comparison to get the SPO2 level. The signal obtained from Max4466 module is sent through an analog to digital converter to get the sound level. Now, the 3 values obtained by the code are sent to the HC-06 module as CSV files. Then they are transferred to the mobile phone via bluetooth. All the data received by the mobile phone throughout 5-6 hours of sleeping time is further analyzed to calculate the average number of sleep apnea instances occurred per hour. The average value calculated is displayed to the user via the app, once he wakes up in the morning.

5.2 Breadboard Implementation with the Micro controller

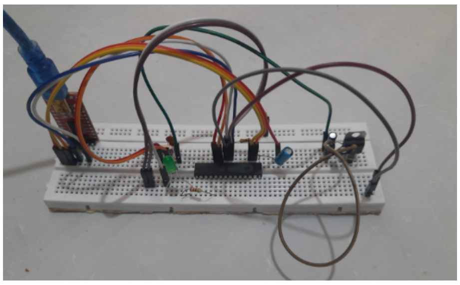


Figure 18: *Circuit*

The arduino code was uploaded to the ATmega328P micro-controller and mounted on the breadboard. Then the complete circuit was developed on the breadboard and the code was implemented to test the continuity of the circuit. Then each module was tested separately to check if they provide the correct outputs. The thumb was kept on the oximeter to check if it outputs heart rate and SpO2 level. Also, it was checked if the snoring sound level was detected by the MAX4466 module. This was checked via the user interface developed in the mobile app.

5.3 Results

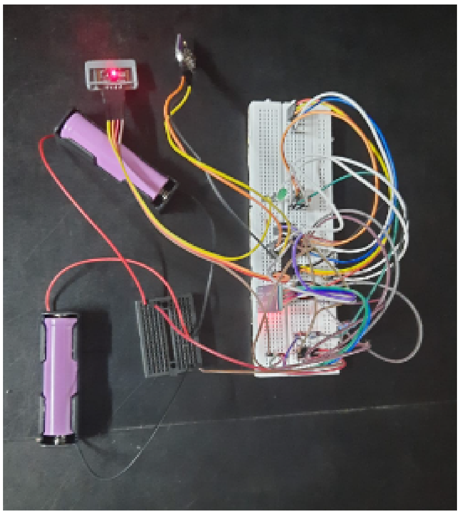


Figure 19: *Circuit for test*

When the device is switch on, power on button switches on. Then mobile app should be opened and pair with the Bluetooth module. Then Bluetooth module indicator bulb stops blinking. Then we have to keep the finger on the sensor and press start button on the app. Then the app will show the current heart rate and SpO2 level. Also it shows us the number of apnea instances. When we want to remove the finger we can press the pause button and remove the finger. App will show the last result.

6 Technical Specification

6.1 Physical Dimensions and Weigh

6.1.1 Dimensions

Physical dimensions are 11x6.5x4.5 in centimetres, due to the current electricity crisis in Sri Lanka certain thin Lipo batteries are hard to find on the market if we could find such a battery device height will reduce to 2.5cm

6.1.2 Weight

We estimate enclosure and belt will be about 100g, PCB and electronic components will be about 20g and batteries will be about 42 x 2 g. Then total mass will be about 204g. 3.7V thin Lipo batteries were hard to find due to the current situation, but if we use them weight will be reduced by 60g and final weight would be 144g.

6.2 Power Consumption

Componant	**Power consumption**
Atmega328p	70 mW
MAX30102	3 mW
MAX4466 MIC and PreAmp	1 mW
Bluetooth module	40 mW
Voltage Regulator	50 mW
other components	16 mW
Total Power	180 mW

Power is supplied to ATmega328p micro controller and the mic module MAX 4466 during the sleeping time continuously. Atmega328p micro controller with 16MHz clock circuit and 5V supply voltage typically draws 46 mW and maximum power consumption is 70 mW as the device is bit computer intensive higher value was selected for the safe margin.The Power consumption of resistors and capacitors are comparatively very law. Therefore the actual power consumption is less than 373.3mW.

7 Marketing and sales

7.1 Marketing Techniques

- The current diagnostic method for sleep apnea is sleep polysomnography. The cost of sleep polysomnography ranges for $1250-$6700. Average cost is around $2925. This cost is equal to the average monthly salary of an American citizen (it is about 5-month salary of a Sri Lankan employee). Our product which detects whether we need to diagnose sleep apnea, this is significantly cheaper than sleep polysomnography. Therefore, the price of the product is affordable to normal people also.

- Working people have a tight schedule and they are busy with their work. So, most of the people don't have enough time to admit to hospital and spend 4 or 5 days on a hospital bed. They can easily use our apnemeter and check whether they have affected or not at home.

- Sleeping polysomnography is a bulky device which is difficult to wear while sleeping but our device is simple and small which makes it convenient to the person.

We are going to go-ahead with above three marketing points majorly.

7.2 Product Packing

Product packaging is quite normal. Simple cardboard box is used and there is a user manual guide which is easily understandable. Since components are placed in separate from each other, repairing can be done unless there is a damage in PCB. If user is going to dispose the device, sensors or other electronic components which are working properly can be used to make other products as well as to make a new apnemeter. This would be a great help to reduce E-waste in the environment also.

8 Product Budget

we are expecting a profit of Rs. 1425.00

Equipment	**Quantity**	**Net Price (LKR)**
Main Technical Equipment (as on june 2022)		
Atmega328p	1	1200.00
MAX30102 Heart rate and pulse oximeter sensor	1	425.00
MAX4466 MIC and PreAmp	1	450.00
TP5100 Battery controller	1	300.00
Bluetooth module	1	700.00
Battery	2	2400.00
Other Equipment(caps, res, LEDs, wires)		100.00
PCB		500.00
Enclosure		2500.00
Cost		8575.00
Sales Price		10000.00

9 Problems and Solutions

The readings of heart rate and Sp02 are obtained by keeping the thumb on the MAX30106 module. If there is no proper contact between them, there won't be a reading or else, if a reading is obtained, it will be inaccurate. Then the equipment will work throughout the night but will fail to provide the correct result.

- Hence an indicator is needed to signal the right contact. Therefore, a **LED bulb** is used and it will light up when a proper connection is established.

The initial plan was to wear the mic module on the wrist, to make the distance between the 2 modules minimum, so that the wire connecting them is short as possible. If that was the case, then the patient has to keep his palm closer to the mouth, preferably on the chest to get a proper output signal. Since this is not practical, it was decided to wear the mic module on the bicep. Therefore, the length of the connecting wire was increased. But it was the better trade-off.

Required small lipo batteries are rare.

- had to use 18650 batteries, therefore size of the enclosure became little bit larger than we expected.

SRAM was not sufficient for the code

- The initial plan was to use an SD card module to store the recorded audio files. But the SRAM of the microcontroller was not enough to execute the code. Also it seemed useless to store the data in a SD card and then separately view it by connecting to a PC. Therefore we decided to use a bluetooth module to transfer the data to the mobile phone, and then run the next part of the code in the mobile phone, and display the result through the mobile app in a user friendly way. If a SD card was used, the user would have to connect it again and again to the PC to view the results multiple times. But the mobile app makes it more convenient for the user to review the results whenever he wants.

Mobile app made OLED display useless

- Since it is useless to display the outputs via an OLED display when the person is in sleep, it was removed. This reduced the size of the equipment as well.

MAX30102 sensor accuracy reduced when used at wrist

- Had to make another enclosure and move the sensor to the finger, had to find a way to make the sensor sticked to the finger

MAX4466 mic module input sound intensity is less

- Mic module was placed closer to the mouth by placing it in a separate enclosure.

Two 3.7V batteries had to be used to get the required voltage. But TP4456 was not capable of charging 2 batteries connected in series.

- TP5100 was used which can charge both batteries connected in series.

10 Future Developments

- Can minimize the size of the product by using smaller batteries
- Can increase the sensitivity by using high reliable modules instead of MAX30102 and MAX4466
- Since this is a medical device further testing and validating processes are required to proceed further **NMRA** (National Medicine Regulation Authority) approval should be required.

11 Acknowledgement

First and foremost, we would want to express our sincere gratitude to our lecturers for giving us the technical knowledge and design strategies we need. Additionally, we want to express our gratitude to the technical personnel for helping us test the product in the lab.

12 References

- https://aasm.org/rising-prevalence-of-sleep-apnea-in-u-s-threatens-public-health
- https://pubmed.ncbi.nlm.nih.gov/23770166/
- https://www.hopkinsmedicine.org/health/conditions-and-diseases/obstructive-sleep-apnea

13 Appendices

13.1 ATmega328p Data sheet Summary

Features

- High performance, low power AVR® 8-bit microcontroller
- Advanced RISC architecture
  - 131 powerful instructions – most single clock cycle execution
  - 32 × 8 general purpose working registers
  - Fully static operation
  - Up to 16MIPS throughput at 16MHz
  - On-chip 2-cycle multiplier
- High endurance non-volatile memory segments
  - 32K bytes of in-system self-programmable flash program memory
  - 1Kbytes EEPROM
  - 2Kbytes internal SRAM
  - Write/erase cycles: 10,000 flash/100,000 EEPROM
  - Optional boot code section with independent lock bits
    - In-system programming by on-chip boot program
    - True read-while-write operation
  - Programming lock for software security
- Peripheral features
  - Two 8-bit Timer/Counters with separate prescaler and compare mode
  - One 16-bit Timer/Counter with separate prescaler, compare mode, and capture mode
  - Real time counter with separate oscillator
  - Six PWM channels
  - 8-channel 10-bit ADC in TQFP and QFN/MLF package
    - Temperature measurement
  - Programmable serial USART
  - Master/slave SPI serial interface
  - Byte-oriented 2-wire serial interface (Phillips I^{2}C compatible)
  - Programmable watchdog timer with separate on-chip oscillator
  - On-chip analog comparator
  - Interrupt and wake-up on pin change
- Special microcontroller features
  - Power-on reset and programmable brown-out detection
  - Internal calibrated oscillator
  - External and internal interrupt sources
  - Six sleep modes: Idle, ADC noise reduction, power-save, power-down, standby, and extended standby

13.2 PCB

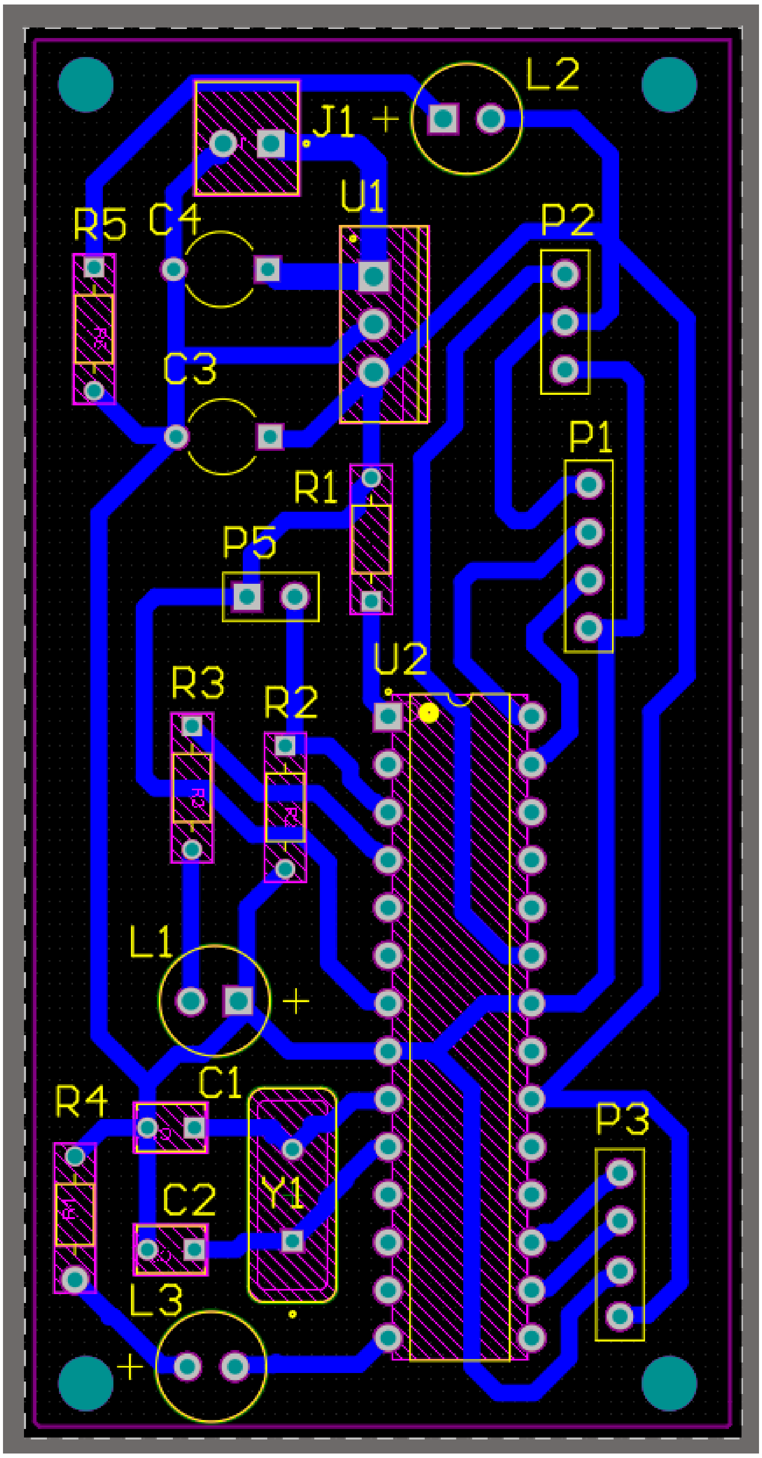


Figure 20: Layer Architecture

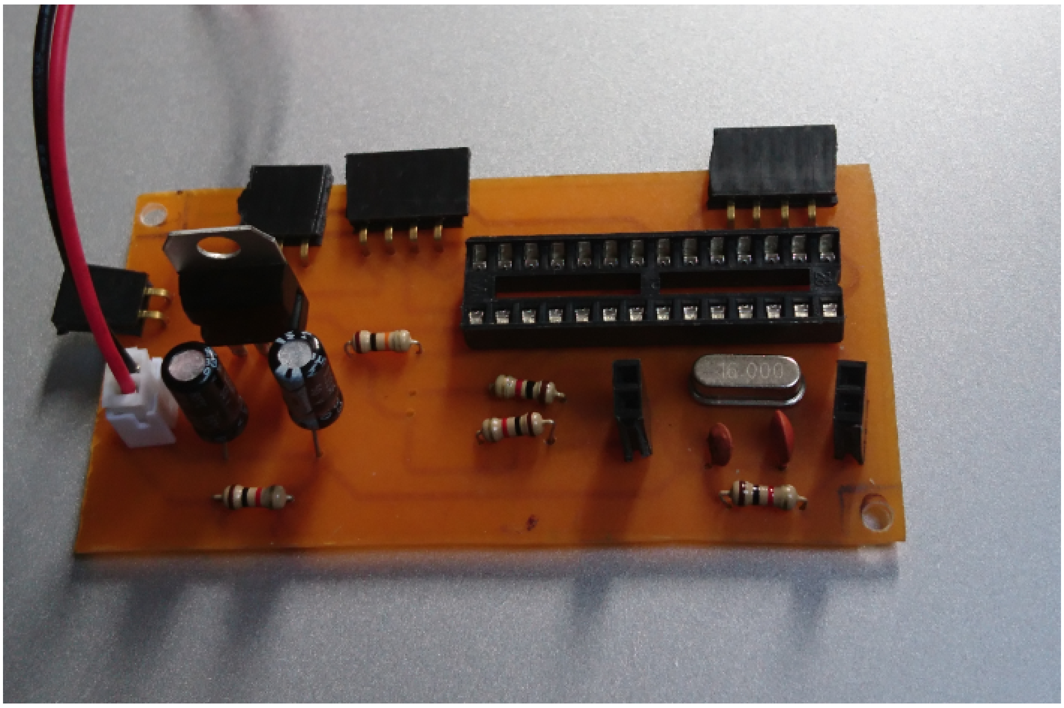


Figure 21: top view

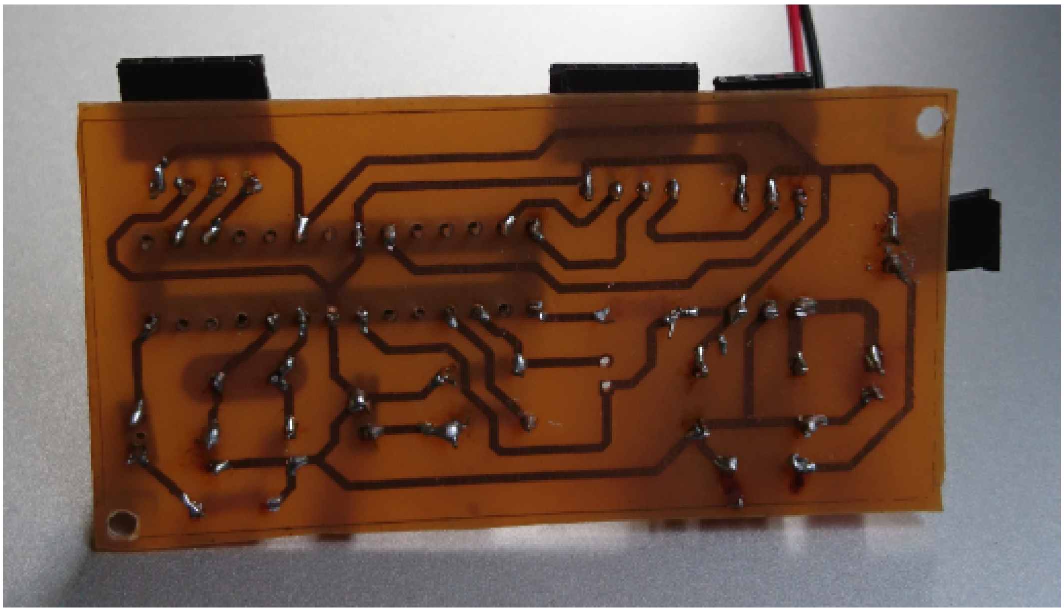


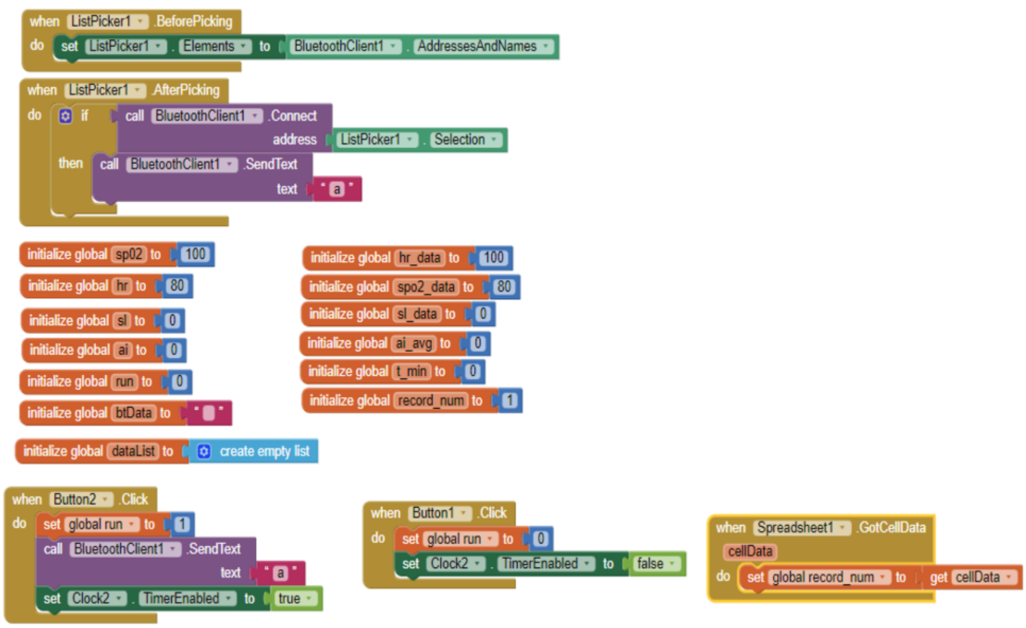
Figure 22: bottom view

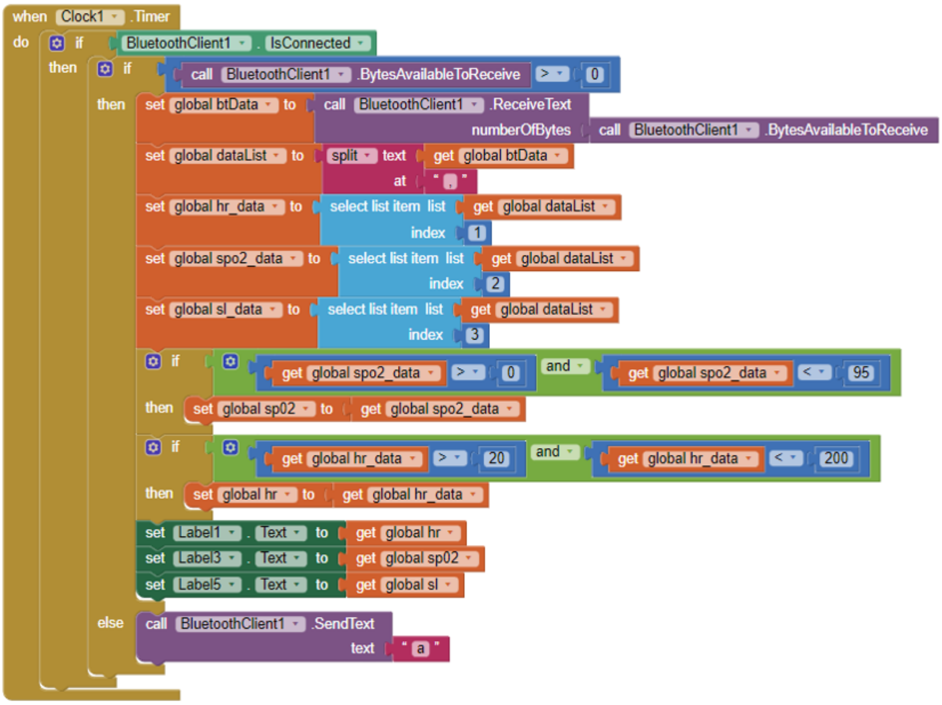
13.3 Code

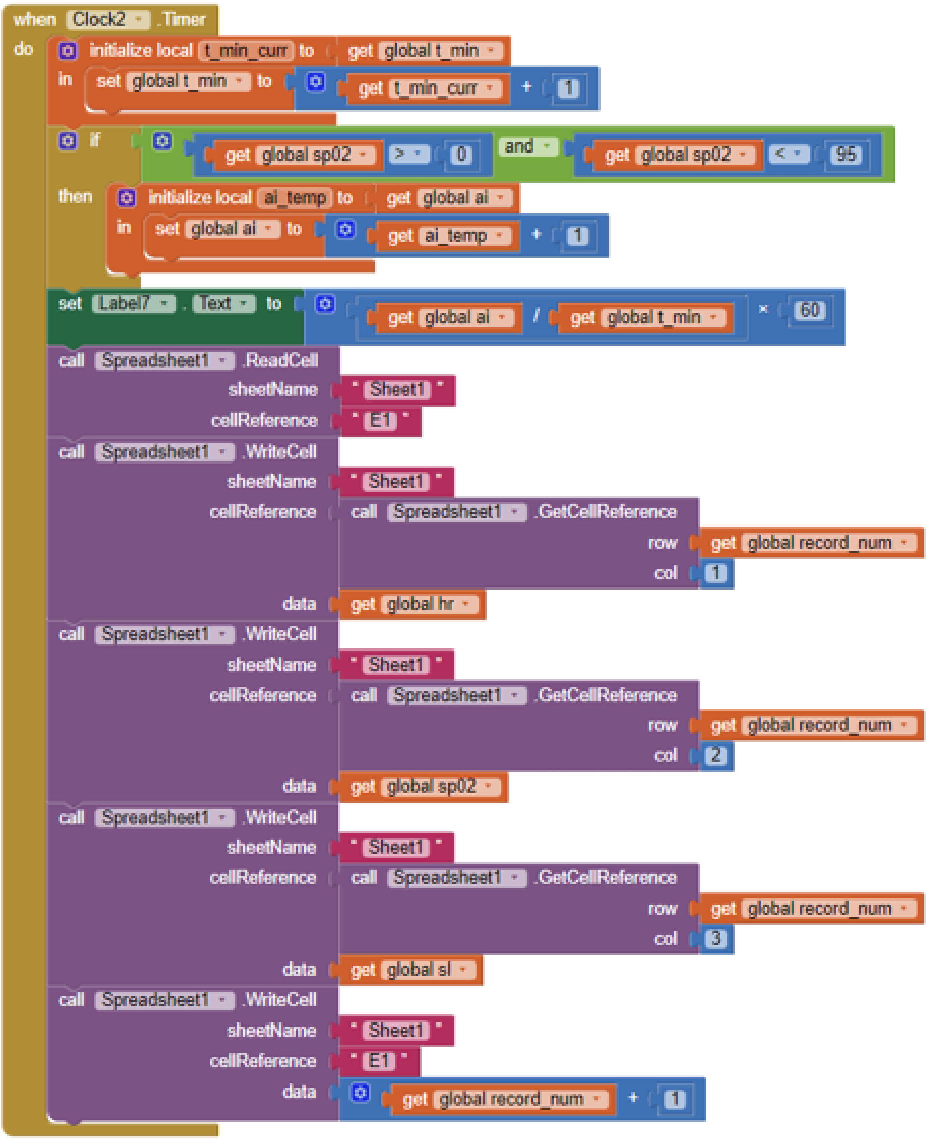
```
//------------------------------------------------------------------------------------------
// Apnemeter v3.5
// Group EN-13
//------------------------------------------------------------------------------------------

/*
*/
//------------------------------------------------------------------------------------------
// Import libraries
#include <Wire.h>
#include "MAX30105.h"
#include "spo2_algorithm.h"
#include <SoftwareSerial.h>

//------------------------------------------------------------------------------------------
SoftwareSerial mySerial(10,11); //RX,TX
MAX30105 particleSensor;
#define MAX_BRIGHTNESS 255
//------------------------------------------------------------------------------------------
//Arduino Uno doesn't have enough SRAM to store 100 samples of IR led data and red led data in 32-bit format
//To solve this problem, 16-bit MSB of the sampled data will be truncated. Samples become 16-bit data.
#if defined(__AVR_ATmega328P__) || defined(__AVR_ATmega168__)
uint16_t irBuffer[100]; //infrared LED sensor data
uint16_t redBuffer[100];  //red LED sensor data
#else
uint32_t irBuffer[100]; //infrared LED sensor data
uint32_t redBuffer[100];  //red LED sensor data
#endif
//------------------------------------------------------------------------------------------
// Define Variables
int32_t bufferLength; //data length
int32_t spo2; //SPO2 value
int8_t validSPO2; //indicator to show if the SPO2 calculation is valid
int32_t heartRate; //heart rate value
int8_t validHeartRate; //indicator to show if the heart rate calculation is valid


//------------------------------------------------------------------------------------------
// Initial Setup
void setup()
{


mySerial.begin(9600); // initialize hc-06 serial communication

pinMode(8, OUTPUT);

// Initialize MAX30102 sensor
if (!particleSensor.begin(Wire, I2C_SPEED_FAST)) //Use default I2C port, 400kHz speed
{
digitalWrite(8,HIGH);
  while (1);
}

while (mySerial.available() == 0) ; //wait until user presses a key
mySerial.read();
/------------------------------------------------------------------------------------------
/ Setup MAX30102 Sensor
byte ledBrightness = 60; //Options: 0=Off to 255=50mA
byte sampleAverage = 4; //Options: 1, 2, 4, 8, 16, 32
byte ledMode = 2; //Options: 1 = Red only, 2 = Red + IR, 3 = Red + IR + Green
byte sampleRate = 100; //Options: 50, 100, 200, 400, 800, 1000, 1600, 3200
int pulseWidth = 411; //Options: 69, 118, 215, 411
int adcRange = 4096; //Options: 2048, 4096, 8192, 16384

particleSensor.setup(ledBrightness, sampleAverage, ledMode, sampleRate, pulseWidth, adcRange); //Configure sensor with these settings


/------------------------------------------------------------------------------------------
/ Loop

oid loop()

bufferLength = 100; //buffer length of 100 stores 4 seconds of samples running at 25sps sample rate

for (byte i = 0 ; i < bufferLength ; i++)
{
  while (particleSensor.available() == false)
    particleSensor.check(); //Check the sensor for new data

  redBuffer[i] = particleSensor.getRed();
  irBuffer[i] = particleSensor.getIR();
  particleSensor.nextSample(); //move to the next sample
```

```
  int pulseWidth = 411; //Options: 69, 118, 215, 411
  int adcRange = 4096; //Options: 2048, 4096, 8192, 16384

  particleSensor.setup(ledBrightness, sampleAverage, ledMode, sampleRate, pulseWidth, adcRange); //Configure sensor with these settings
}

//----------------------------------------------------------------------------------------------------------------
// Loop

void loop()
{
  bufferLength = 100; //buffer length of 100 stores 4 seconds of samples running at 25sps sample rate

  for (byte i = 0 ; i < bufferLength ; i++)
  {
    while (particleSensor.available() == false)
      particleSensor.check(); //Check the sensor for new data

    redBuffer[i] = particleSensor.getRed();
    irBuffer[i] = particleSensor.getIR();
    particleSensor.nextSample(); //move to the next sample

  }

//----------------------------------------------------------------------------------------------------------------------------------
  //calculate heart rate and SpO2
  maxim_heart_rate_and_oxygen_saturation(irBuffer, bufferLength, redBuffer, &spo2, &validSPO2, &heartRate, &validHeartRate);
// take the sound level
  int soundLevel = analogRead(A0);

//-----------------------------------------------------------------------------------------------------------------------------------
// Serial communication using hc-06
  mySerial.print(heartRate, DEC);
  mySerial.print(F(","));
  mySerial.print(spo2, DEC);
  mySerial.print(F(","));
  mySerial.println(soundLevel, DEC);
//-----------------------------------------------------------------------------------------------------------------------------------
  delay(1000);

}
```

13.4 Android Application




when Clock1 .Timer
do if BluetoothClient1 . IsConnected
then if call BluetoothClient1 .BytesAvailableToReceive > 0
then set global btData to call BluetoothClient1 .ReceiveText
numberOfBytes call BluetoothClient1 .BytesAvailableToReceive
set global dataList to split text get global btData
at " | "
set global hr_data to select list item list get global dataList
index 1
set global spo2_data to select list item list get global dataList
index 2
set global sl_data to select list item list get global dataList
index 3
if get global spo2_data > 0 and get global spo2_data < 95
then set global sp02 to get global spo2_data
if get global hr_data > 20 and get global hr_data < 200
then set global hr to get global hr_data
set Label1 . Text to get global hr
set Label3 . Text to get global sp02
set Label5 . Text to get global sl
else call BluetoothClient1 .SendText
text " a "


when Clock2 .Timer
do initialize local t_min_curr to get global t_min
in set global t_min to get t_min_curr + 1
if get global sp02 > 0 and get global sp02 < 95
then initialize local ai_temp to get global ai
in set global ai to get ai_temp + 1
set Label7 . Text to get global ai / get global t_min × 60
call Spreadsheet1 .ReadCell
sheetName " Sheet1 "
cellReference " E1 "
call Spreadsheet1 .WriteCell
sheetName " Sheet1 "
cellReference call Spreadsheet1 .GetCellReference
row get global record_num
col 1
data get global hr
call Spreadsheet1 .WriteCell
sheetName " Sheet1 "
cellReference call Spreadsheet1 .GetCellReference
row get global record_num
col 2
data get global sp02
call Spreadsheet1 .WriteCell
sheetName " Sheet1 "
cellReference call Spreadsheet1 .GetCellReference
row get global record_num
col 3
data get global sl
call Spreadsheet1 .WriteCell
sheetName " Sheet1 "
cellReference " E1 "
data get global record_num + 1

13.5 Enclosure

Figure 23: *Enclosure*

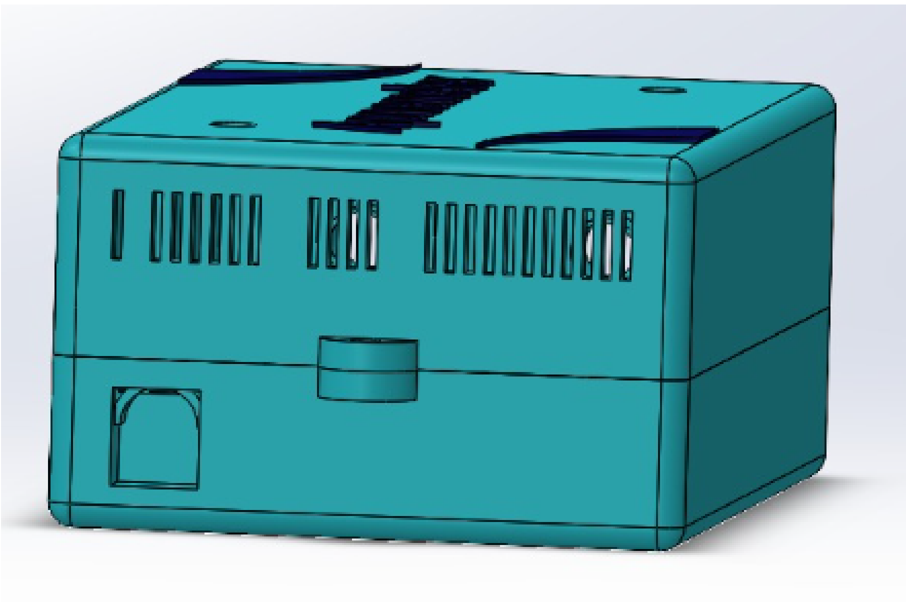


Figure 24: *Enclosure*