APP International Class

Group G3

INTELLISENSE

Optimal Environment Quality

Table of content

1.Introduction

When people are exercising, they always want to know the environmental quality index of the outside world to know whether the environment is suitable for exercise. Therefore, we put sensors and microcontrollers at the entrances and exits of places where people often exercise, such as parks or forests, to detect air quality, such as temperature. So the purpose of this project is, by connecting different environmental sensors (sound, temperature and humidity, to the Tiva board, this detection system can be turned on by playing an audio with a certain sound intensity and a specific frequency by being close to the sensor , and record and analyze the stored data on the main board, display the sound intensity and quality and the overall air quality through the local playback screen, and athletes can use their smartphones to receive other specific temperature, humidity and other data.

The microphone is used to detect the intensity of ambient noise and specify the audio key.When the microphone receives the sound signal, it will perform a preliminary signal power check to save battery power; when the sound signal strength is greater than the minimum threshold value set, the band-pass filter is activated.

When the audio key successfully passes through two band-pass filters of specified frequency, the main board will process the data of each sensor received by tivaboard. tivaboard starts to analyze various environmental indexes to determine the quality of the environment and whether it is suitable for running sports. At the same time, the main board sends part of the data (sound intensity, quality, whether the environment is suitable for outdoor sports overall assessment) to Oled, and all parameters and analysis results Send to mobile phone via bluetooth.

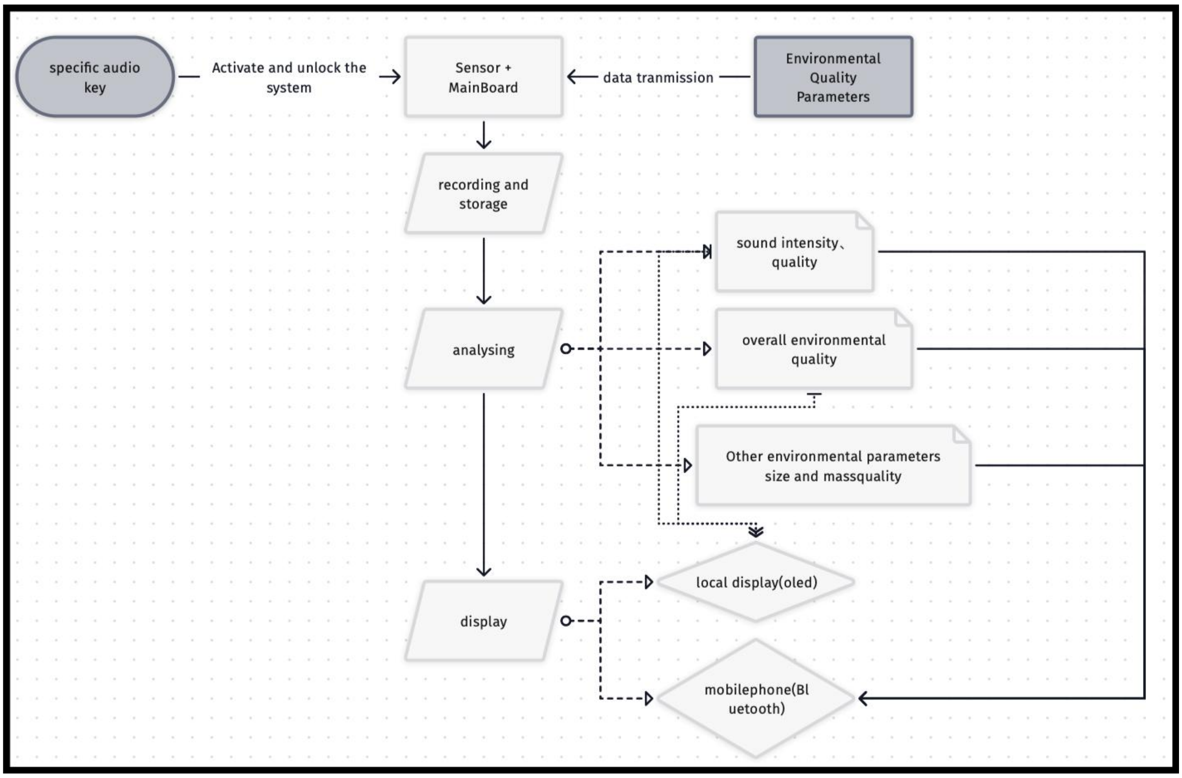


Figure1.1 Flow chart for environment dectection system

2.Project objective

To detect environmental indices by collecting relevant parameters such as:

Environmental sound quality

Temperature and Humidity

Set different thresholds to determine how good the environmental indices are represented.

The design mechanism of this solution is presented in Figure 2.1

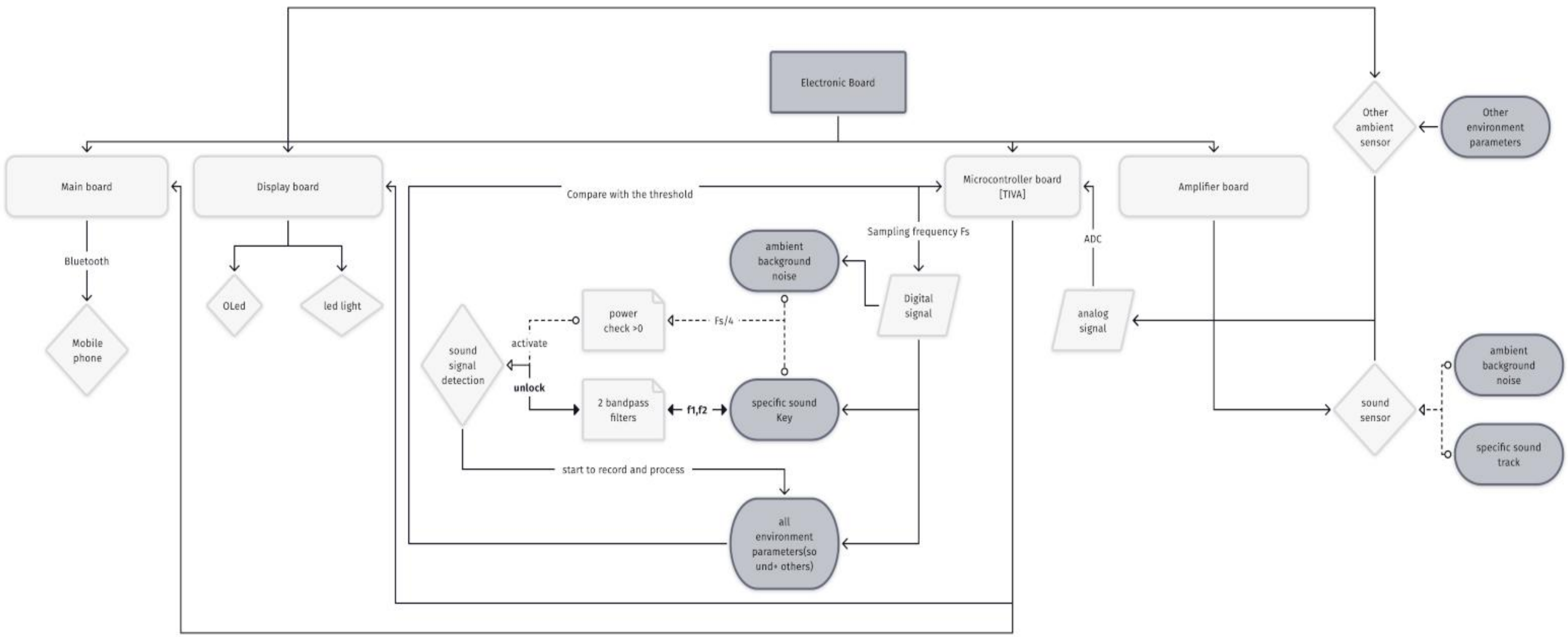


Figure 2.1 Flow Chart

3. Electronics implementation

3.1 Microcontroller and software introduction

Based on the project design, the electronics of the project were assembled and integrated around the microcontroller.

3.1.1 Tiva board:

The LaunchPad is an evaluation board (EK-TM4C123GXL) from Texas Instruments, which uses ARM Cortex-M4F-based microcontroller of Tiva C Series. It is a low-cost evaluation platform featuring 80-MHz Arm Cortex-M4F CPU, 256kB of flash, and 32kB of SRAM. It provides an integrated USB 2.0 support for USB Host/Device/OTG and two 12-bit ADC modules. The TM4C123GH6PM also includes a multitude of serial communication channels such as UART, SPI, I2C, and CAN with additional device features such as the hibernation and PWM modules. The EK-TM4C123GXL includes programmable user buttons and an RGB LED for custom applications. The stackable headers provide the interface to connect BoosterPack™ plug-in modules and make it easy and simple to expand the functionality of the TM4C123G LaunchPad through the Texas Instruments BoosterPack ecosystem.

Figure 3.1 ARM® Cortex®-M4F Based MCU TM4C123G Tiva Board

3.1.2 Energia Software

Energia IDE was used to program the microcontroller and in C language. It was also used to display the environmental parameters obtained by the sensors.

The energia is an open-source electronics prototyping platform aimed at bringing wiring and arduino framework to the Texas Instruments MSP430 based LaunchPad. It is an integrated development environment (IDE) supported by the Mac OS, windows and Linux. It has its foundation in the Processing IDE (Processing→Wiring→Arduino→Energia). Energia is also a portable framework/abstraction layer that can be used in other popular IDEs. The energia version used in this project is “energia-1.8.10E23-windows”.

3.2 Sensors

3.2.1 DHT 11 Temperature and Humidity Sensor

(1) introduction

The DHT 11 temperature and humidity sensor was used in this project. According to the datasheet, “the sensor features a temperature & humidity sensor complex with a calibrated digital signal output. it includes a resistive-type humidity measurement component and an NTC temperature measurement component, and connects to a high-performance 8-bit microcontroller, offering excellent quality, fast response, anti-interference ability and cost-effectiveness".

The sensor, shown in the Figure 3.2 below has four pins and was soldered (connected) to the appropriate terminal points on the pcb board (I.e 3.3v supply, ground, and digital pin 36 on the Tiva board processor).

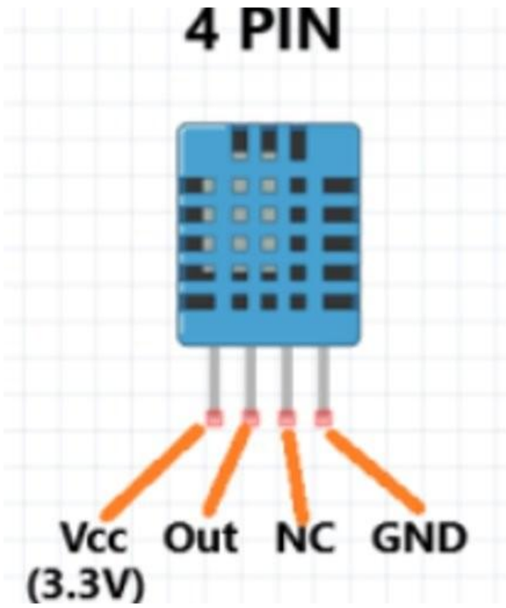


Figure 3.2. DHT 11 Temperature and Humidity Sensor

(2)Process for obtaining the data from DHT 11 Sensor

When Tiva microcontroller sends a start signal, the DHT11 sensor change state from the low-power-consumption mode to the running-mode, and waits for the microcontroller to complete the start signal. Once it is completed, the sensor sends a 40-bit response signal of data that includes the relative humidity and temperature information to the microcontroller. The data is den collected and read (see figure 3.4 for the output on the serial monitor). Once data is collected, DHT11 will change to the low-power-consumption mode until it receives another start signal from microcontroller.

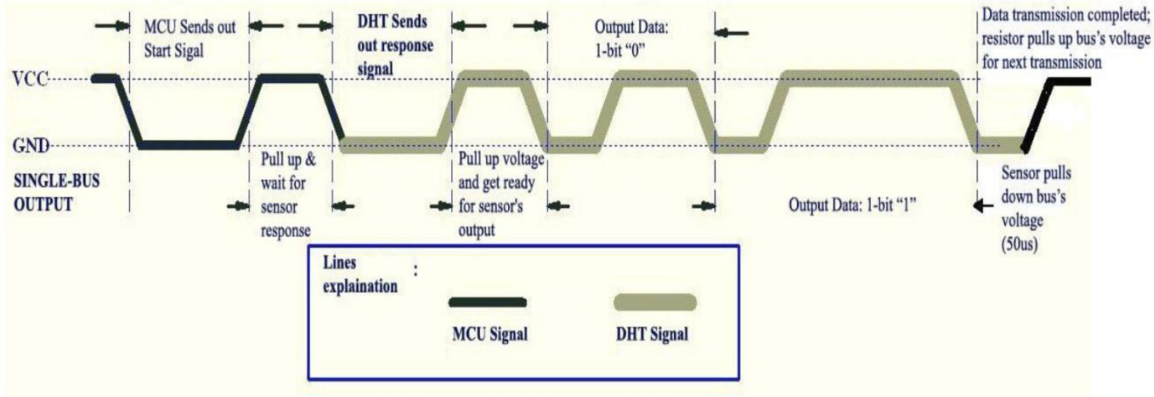


Figure 3.3 Data Communication Process of DHT 11 Sensor

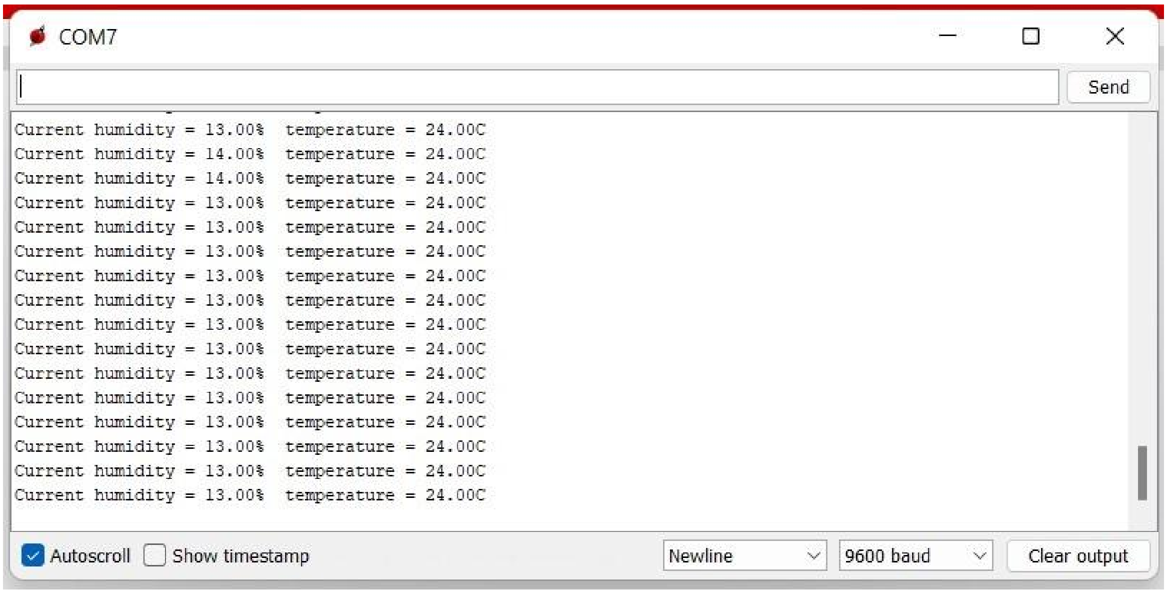


Figure 3.4. Humidity and Temperature Reading on the Serial Monitor

3.2.2 Sound sensor

(1) Introduction

The MAX4466 is an electret microphone amplifier, not an ultrasonic sensor. However, it can be used to amplify the output of an electret microphone, which can be used to measure distance in some applications using the ultrasound principle. Here are its specifications:

Amplifier:

Low-noise, high-gain, and unity-gain-stable op-amp

46 dB gain (typical)

Low input bias current: 500 pA

Low input offset voltage: 500 µV

Unity-gain stable

Power:

Operating voltage: 2.7V to 5.5V

Low quiescent current: 250 µA

Power-down mode: <1 µA

Other specifications :

SOT-23-5 package

Available in green (Pb-free) packaging options

Figure 3.5 MAX4466 sound Sensor

(2) implementation of sound sensor

We got the code from the internet and we got library of sound sensor and downloaded it . After that we attach library in the energia software. we specified pin number of sound sensor. Sound sensor is mainly used for keydetection part .

Operating frequency: the frequency of the ultrasonic waves used by the sensor.

Detection range: the maximum distance the sensor can detect an object.

Sensitivity: the ability of the sensor to detect objects accurately at various distances and in various conditions.

Resolution: the smallest distance change that the sensor can detect.

Beam angle: the angle of the ultrasonic waves emitted by the sensor.

Operating voltage: the voltage required to power the sensor.

Output signal: the type of output signal produced by the sensor (e.g. analog voltage, digital pulse).

Package type: the physical size and shape of the sensor.

Environmental compatibility: the sensor's ability to operate in various environmental conditions (e.g. temperature, humidity).

EMI/RFI Immunity: the sensor's ability to resist interference from electromagnetic and radio-frequency sources.

3.3 Data transfer and display

3.3.1ADC and Voltage value values via Putty

We interfaced the sound sensor to the Tiva microcontroller and programmed it to read the sound parameter of the environment. Although the sound sensor is an analog device which sends its reading to the microcontroller as analog signals, the 12-bits in-built Analog-to-Digital Converter (ADC) converts these analog signals and display them in 12 bits binary on the serial monitor which can also be viewed via Putty. These 12 bits binary values which ranges from 0-4095 are further interpolated to voltage values between a range of 0-3.3V which is also viewed in the serial monitor or via putty. The formular used to interpolate the 12-bit binary value to voltage is given by:

$$VCC * ADC/4095$$

Where VCC the maximum voltage of the microcontroller 3.3V and ADC is the reading of the sound sensor.

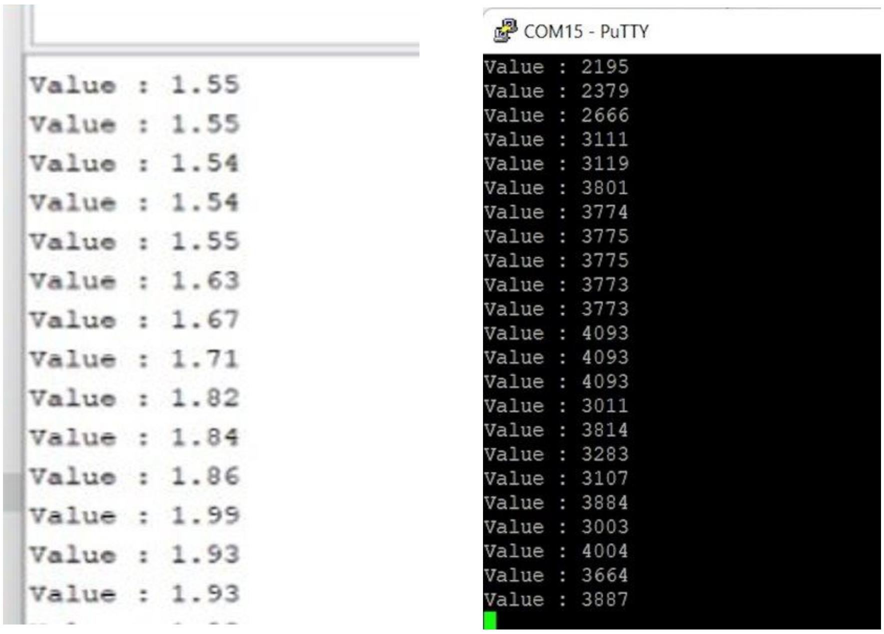


Figure 3.6 Images of the 12-bits binary display and interpolated display on putty

3.3.2 Display on PC and mobile via Bluetooth Connection

We activated the Bluetooth feature of the microcontroller by writing a program to this effect and tested the connectivity by connecting mobile phones and laptops to display the readings from the connected sensor.

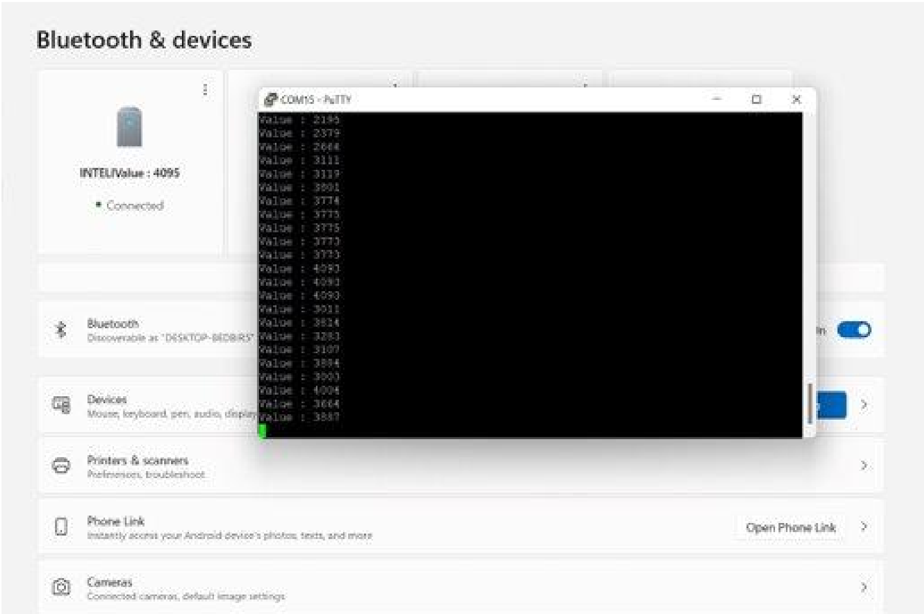


Figure 3.7 Image of the 12-bit binary display via Bluetooth

3.3.3 OLED DISPLAY

We uploaded the images of the school and Intellisense logo in a logo-to-hexadecimal converter website and converted the images into hexadecimal codes and input it into the code in Energia and displayed these logos in LCD.

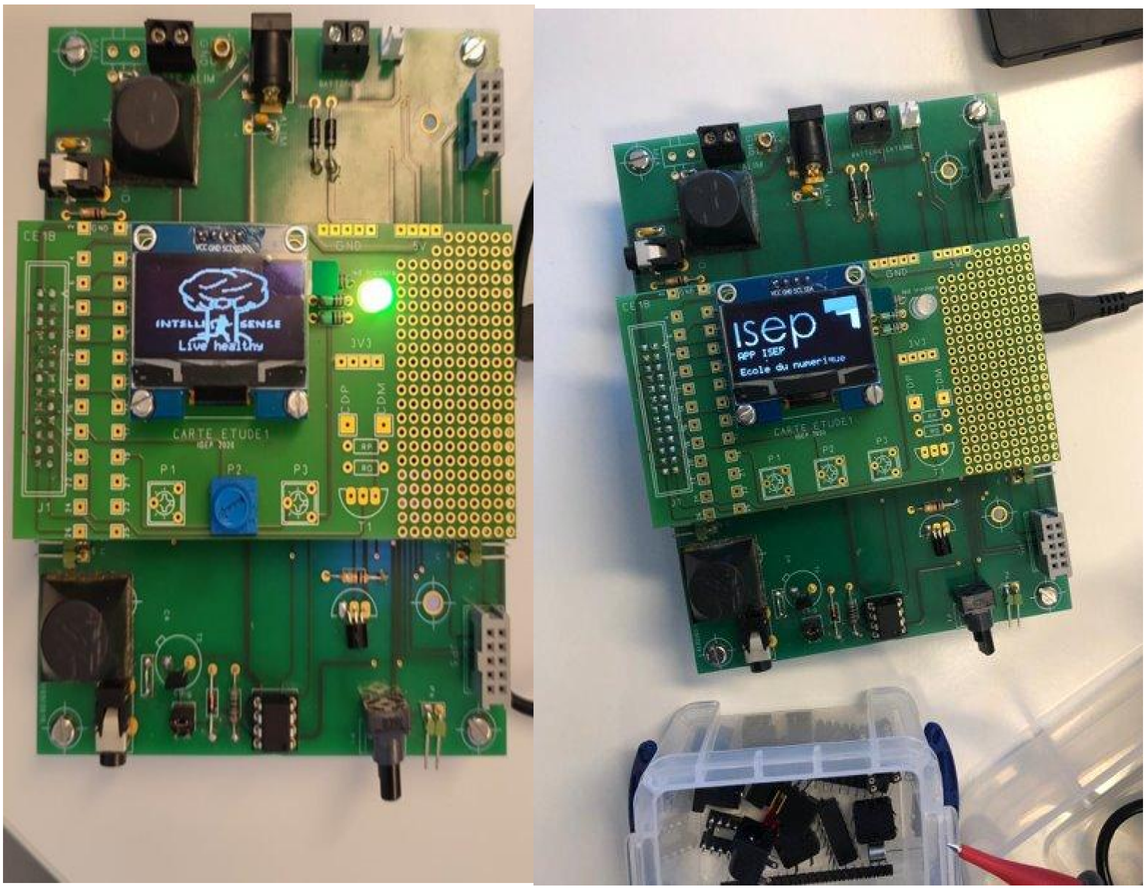


Figure 3.8 Images showing the display of the school and intellisense logos on LCD.

3.4 Battery

3.4.1 Energy Conservation

We tested current and voltage for our project, the current is 0.12 A and voltage 3.8 V, we calculated Power P = VI, which is 0.456 watt. To convert into mw, we multiplied by 1000, it gives 456 mw, Our project is expected to run 13 hours a day, we calculated mwh, 456 mw * 13 h = 5928 mwh, which is approximately 6000 mwh.

Therefore, energy consumed in 1 year = 6000 x 365 = 2,190,000mwh. This equivalent to 0.0009 Metric Tons of CO_2 = 1 pound of coal burned = 0.001 acres of US forest in 1 year.

To conserve battery, we recommend a solar solution to charge the battery. To convert mwh to mAh, we divided with voltage , which gives 1578 mAh, which is approximately 2000 mAh.

I = 0.12 A;

V = 3.8 V;

P = IV = 0.12 * 3.8 = 0.456 watt;

To convert mw,

P = 0.456 * 1000 = 456 mw;

To calulate 13 hours,

456 mw * 13 h = 5928 mwh , (approximately 6000 mwh)

To convert mwh to mAh,

6000 Mwh / 3.8 V = 1578 mAh (approximately 2000 mAh).

3.4.2Power consumption and battery autonomy

Based on the calculation of power, we decided to use 3.7V 2000 mAh battery for our device.

Battery Specification :
Battery Lithium-ION
EEMB 3.7V 2000mAh
Battery Lithium-ION
Rechargeable with connector

Storage & Application :
No leaks, excellent economic performance and long service life.

PCM protection function:
over charge, over discharge, over current, short circuit protection, over temperature protection.

Keep cells in 40-60% state of charge during long-term storage. Recommend charging the battery every 3 months after receiving the battery and keeping the voltage 3.7-4.0V.

4. Data Processing

4.1 System operation

(1)The device is triggered based on the power of an audio signal compared to the silence threshold.

(2)It starts to listen for the key by passing the signal through a filter. If the key is detected by the filter, then the device begins measuring process.

(3)It measures the sound quality in decibels

(4)It measures and outputs the temperature and humidity of the environment.

The figure 4.1 below shows the block diagram of the data treatment of the project.

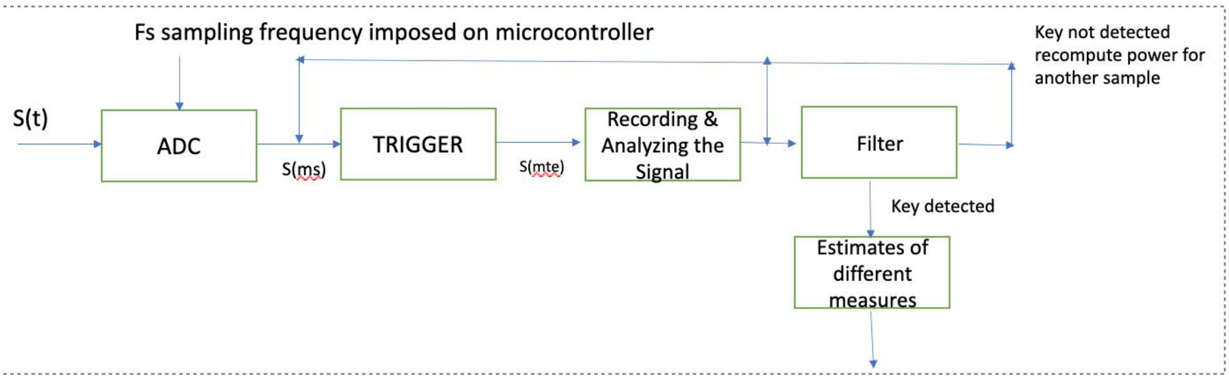


Figure 4.1 Data Treatment Block Diagram for the Project

4.2 Power comparison (Trigger)

An analogue sound signal S(t) is received by the analogue to digital converter (ADC). The ADC converts the signal to a discrete signal in time domain S(mts) with a sampling frequency (Fs) imposed on the signal. The signal is then passed to the trigger process shown in detail in the figure 4.2 below.

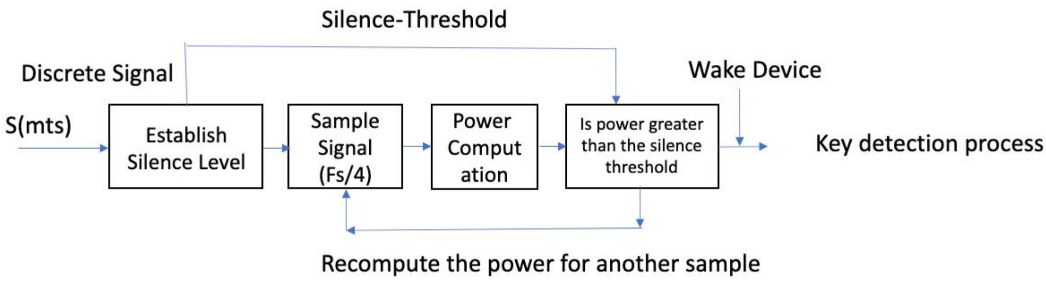


Figure 4.2 Trigger Process Block Diagram

A baseline is established by measuring and computing the power of the sample of a signal in a quiet environment to obtain a silence threshold. Consequently, if the power computed for a sample of the discrete signal passed through the trigger process is higher than the silence threshold, the device is turned on (wake up). Otherwise, the device remains asleep while computing power for another sample of the signal. See figure 4.1 for the formula for the computation of signal power.

Once the device is turned on from the trigger process explained above, the signal is passed to the filter process which involves the transformation of the signal into the frequency domain through the Fast Fourier Transform (FFT). Consequently, the power of the signal after passing through the filter is compared with power computed in the trigger process to identify the activation key (the desired frequency of the signal). The key is detected when the power of the signal after the filter process is greater than the power computed before the filter.

$$E = \sum_{n=0}^{N} |x[n]|^2$$

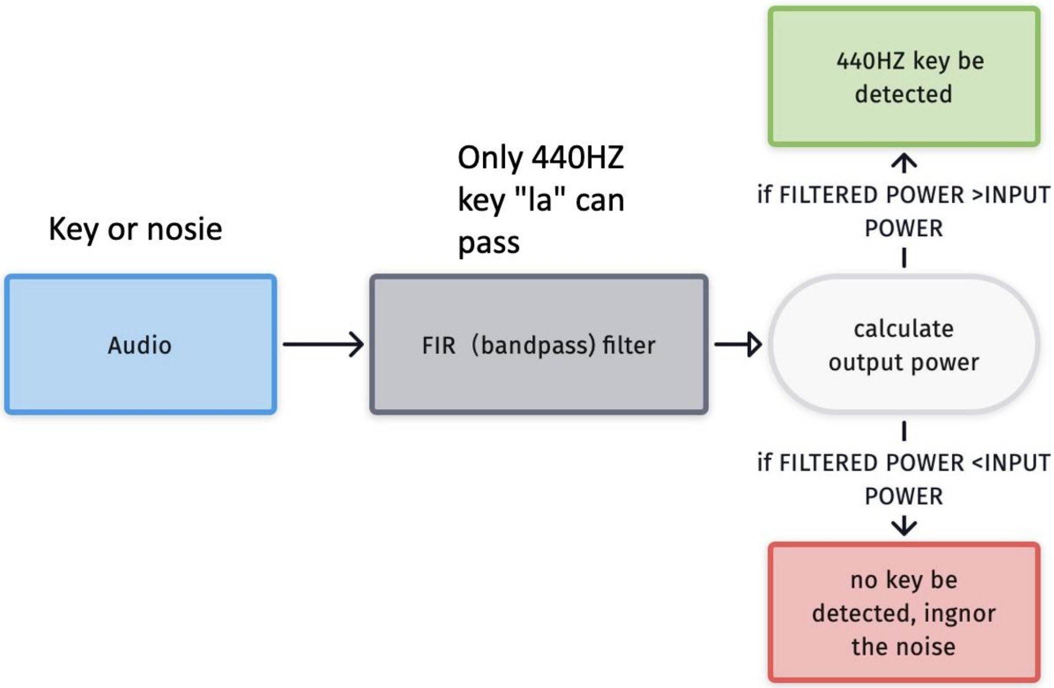
Figure 4.3. The Energy Formula of Discrete Time Signal

4.3FIR filter design and implement

4.3.1 Design idea

After the power comparison, the sound smaller than the threshold is excluded, and the sound larger than the threshold is filtered in the next step: through a bandpass filter.

Therefore, the design concept of this part is: only a sound signal of the specified frequency is used as the key, and a band-pass filter is designed according to the frequency of this signal. In this project, the frequency of the sound signal is 440HZ (the frequency of the mid-range "LA" in music), only this signal can pass through the filter, and the strength of the data signal is detected at the same time. If the strength of the output signal after filtering is greater than the strength of the input signal, it is considered 440Hz key signal is detected, if less than, it is considered that the input has only ambient



noise.

Figure 4.4 Flow chart for filter and key detection

4.3.2 Matlab simulation

In Matlab, we can easily build a bandpass filter through design tools, such as using fdatool-filterdesigner, only need to pay attention to the input filter parameters.

For example sampling frequency Fs: according to Nyquist frequency The sampling frequency must be at least twice the signal frequency. In this case, the signal frequency is 440Hz, so Fs at least=>880Hz. In this project, 4000Hz is selected as the sampling frequency to ensure the accuracy of the sample.

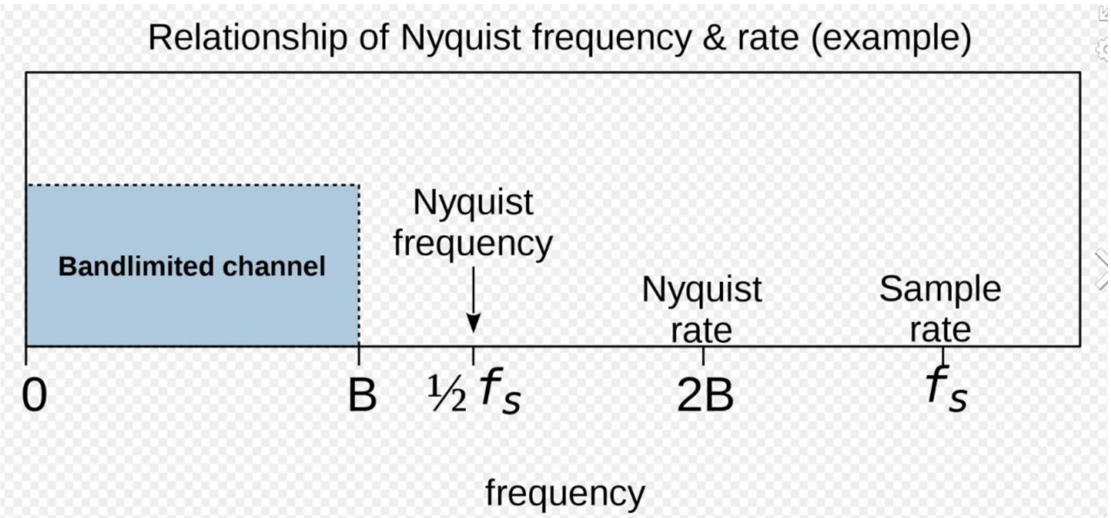


Figure 4.5 Relationship between Nyquist frequency Nyquist rate and signal bandwith

And design method: FIR, Because what we want to simulate is the input of the sound signal, the signal attenuation must exist, so we do not use the IIR model, but the limited FIR model

And response type: Bandpass; only allow 440hz signal pass filter.

And other parameters: like Fstop1, Fpass1, Fstop2,Fpass2...based on how precise filter we want , in this case we choose ±15HZ.

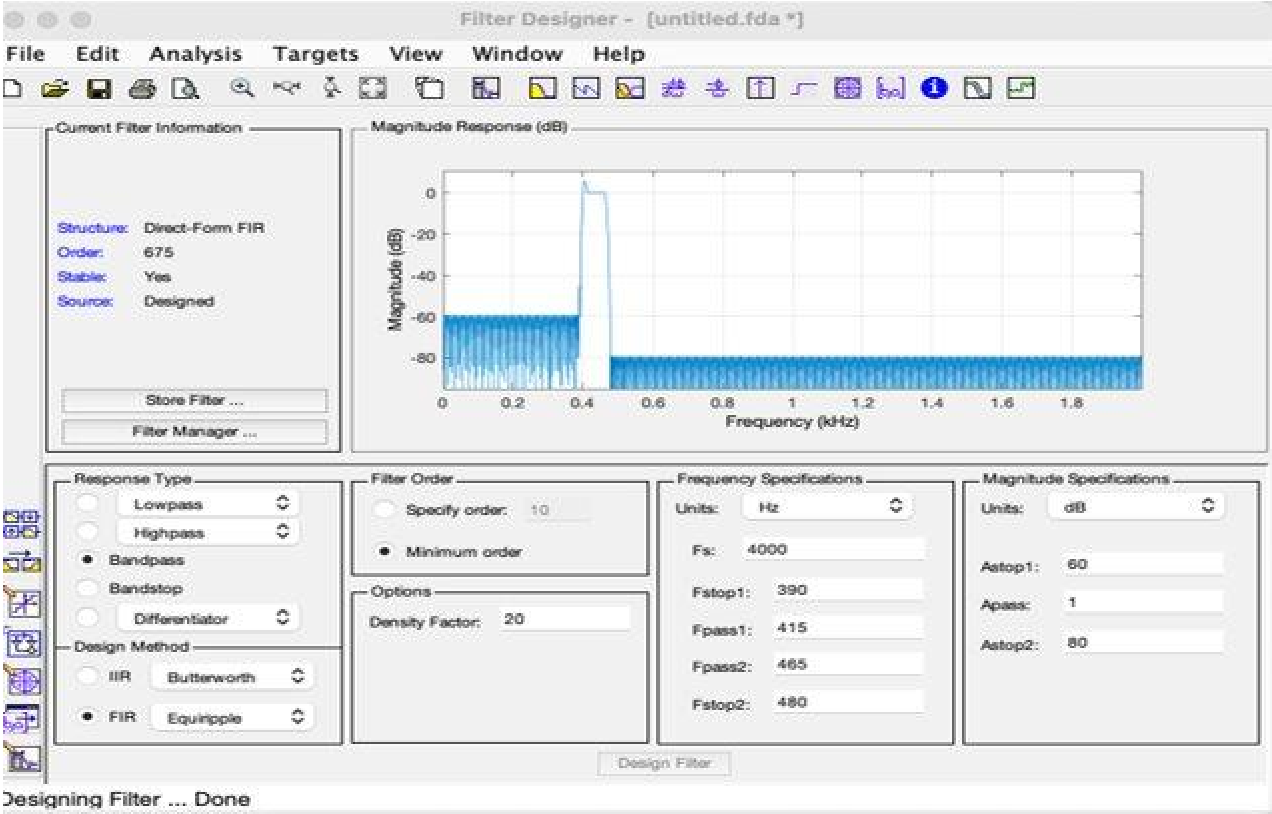


Figure 4.6 filter designer window and bandpass filter spectral response

We choose a signal with only one frequency as the input to test the function of bpf. Fast Fourier transformation FFT to transform the digital signal into frequency domain and then viewed the spectrum of the signals. It can be seen from figure4.7 that the input and output spectrum of the bpf signal are basically the same, which proves that the filter based on this signal allows the signal to pass.

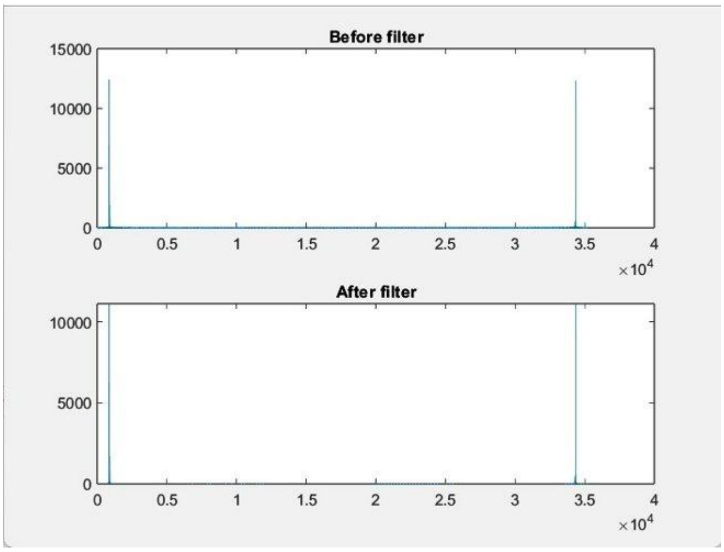


Figure 4.7 Filter input and output spectrum comparison

4.3.3 FIR filter implementation

In the frequency domain, the output of a bandpass filter can be expressed by multiplying the magnitude of the signal input frequency by the corresponding filter constant. But the microprocessor is a device that works in the time domain, so we need to convert the multiplication operation in the frequency domain to the convolution operation in the time domain

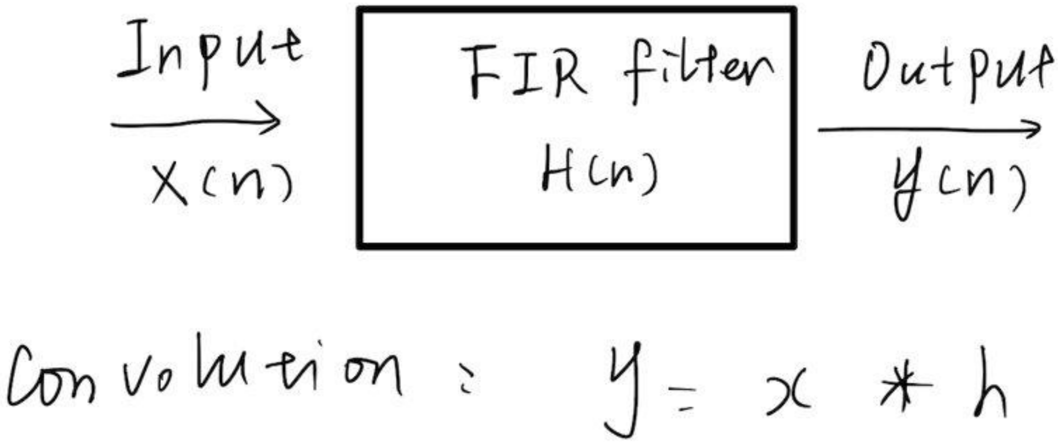


Figure 4.8 convolution-Time Domain Computation of Filter Output

At the same time, the input signals after sampling are all discrete digital signals, so the formula can be expanded one by one according to the sample order as follows:

$$y[n] = \sum_{j=0}^{m-1} h[j] \cdot x[n-j]$$

$$\Rightarrow y[n] = h(0) \cdot x(n) + h(1) \cdot x(n-1) + h(2) \cdot x(n-2) + h(3) \cdot x(n-3) + \ldots + h(m-1) \cdot x(n-m+1)$$

Figure 4.9 expand convolution formula

N: length of sample — NBSAMPLE 1000; which is the buffer to storage the sampling signal.
M: order of filter ---- number of coefficient; which we could generate from Matlab filterdesigner.

Then we need to program the formula of figure 4.9 into the energia platform in C language, and the implement of filter is completed.

4.3.4 key detection

Like we already say on the chapter 4.3.1.Evaluate if the key be detected by calculate the output power of the filtered signal. Key is detected when output power is more than input power.

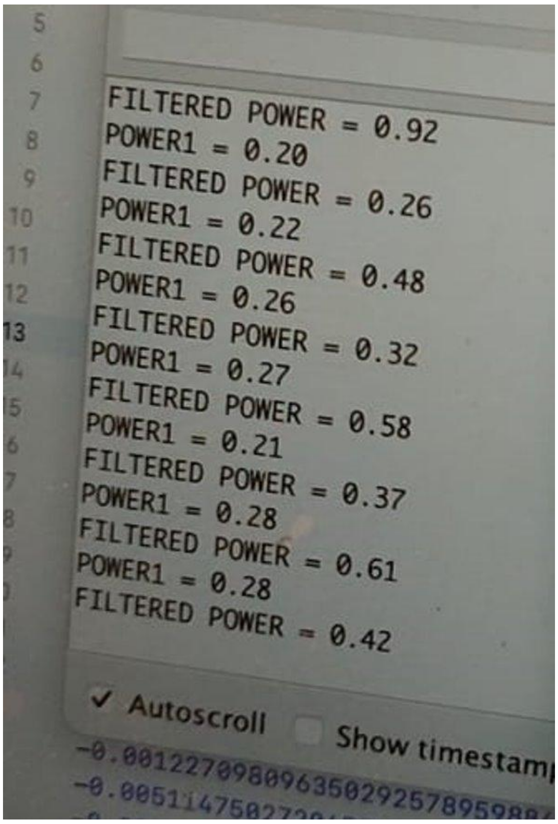


Figure 4.9 display the result of power calculation

When the key is activated, it starts to send temperature and humidity data to users mobile phone via Bluetooth. It also suggests the user to do sports or not in a specified time.

If the temperature is very low, it suggests user to run later. If the temperature is good, it tells the user, it’s a good time to run.

5. Future plans and ways to improve

5.1Future Planning

We can collaborate or sell the product to a fitness app development company. The data that we are gathering from user, is sent to fitness app development company. The data contains, how many users doing sports at a specified time, how many users running etc..

Based on this data, they may give special offer to the users who are using our device, advertise their products etc..

5.2Unfinished features and parts that can be improved

5.2.1 noise level detection

After the audio key pass the bandpass filter,

Sound power check will be performed again to confirm that there is a signal in the passband. The signal will be sent to the main board to display the level of sound and other parameters.

Here we use decibels dB to judge the noise level, through formula dB = 20*lg(A/B), A represents the actual environmental sound intensity, B is the sound intensity of the environment in a quiet state;

Sound level:

0-40 dB low; led green

40-80 dB middle; led blue

80～120dB noisy; led light red

Display the sound level on the Oled

5.2.2 other Signal quality analysis

Thresholds of the following environmental parameters:

Since temperature is a more important parameter to the pm2.5 in our final decision, we assigned numerical values of 3.5 and 2.5 respectively as constants in calculating the weighted average of each of the ranges of both parameters.

The final decision is represented as follows:

0<=final output<=50: unsuitable

51<=final output<=70: fairly suitable

71<=final output<=100: suitable

For the temperature (importance index: 3.5)

RANGE: -4°C - 30°C

-4 - -1: Bad Bad: 0=0%

0°C -6°C: Fair Fair: 2.5 =50%

7°C – 15°C: Ideal Ideal: 5=100%

16°C – 22°C: Fair

23°C – 30°C: Bad

PM2.5 (importance index: 2.5) (not installed)

Range: 0 – 150 Unhealthy: 0=0%

0 – 12: Good: 1=33.3%

12.1 – 35.5: Moderate Moderate: 3=60%

35.5 – 55.4: Unhealthy for Sensitive people Good: 5=100%

55.5 -150.4: Unhealthy

The final computation of the temperature and pm2.5 in percentage determines the decision as to whether it is conductive to jog or not.

5.2.3.Display the data

Use Oled display the following message:

“Sound level: __dB

Sound quality: low/middle/noisy

unsuitable/fairly suitable/ suitable for exercising!”;

Use led light display green/blue/red when the sound quality is low/middle/noisy;

Code on Tiva , and receive the environment data from mobile phone via Bluetooth:

Sound level: _dB， low/middle/noisy;

Temperature： __℃， Bad/Fair/Ideal；

Humidity: __

6. Conclusion

This project uses sensors to detect environmental parameters, and performs energy detection and filtering on sound signals through a microprocessor, and classifies the collected sound, temperature and humidity signals, threshold setting, display, transmission and other follow-up processing. Finally, the user can directly see the processed data through the oled display or receive the processed data through

Bluetooth, and provide the current environmental parameters for the user's exercise.