LIXIE UNIVERSAL TUNER

Technical Documentation
Version Final

Project Manager :
Jérémy Lesauvage (jeremy.lesauvage@abo.fi)

Project Members:
Thibault Cormery (Thibault.cormery@abo.fi)
Andreas Salminen (andreas.salminen@abo.fi)
Marine Poidevin (marine.poidevin@abo.fi)
Oscar Schauman (cschauma@abo.fi)
Benjamin Heinonen (benjamin.heinonen@abo.fi)

GitHub: https://github.com/JeremyLesauvage/LUT-Lixie-Universal-Tuner

Revision history

Date	**Version**	**Description**	**Author**
02/10/2019	1		all the team
07/11/2019	2		Jérémy Lesauvage
07/02/2020	3		Jérémy Lesauvage
10/03/2020	Final		Jérémy Lesauvage

1. Product Requirements

The purpose of building this product is to create a vintage looking and universal tuner, re-using the Nixie principle. To achieve that we need to create a specific hardware, that will contain the battery, the Arduino board, and all the other electronics as well as the Lixie display.

The user of this product may be anyone who plays any kind of music instruments, the tuner will be designed to be intuitive and easy to use. Furthermore, the display will be easy to read due to vibrant colours.

Unlike the others regular tuner, ours will have a rechargeable battery and will be useable by any brass, woodwind, strings, and even percussion instruments. The user will be able to select the mode he needs for his instrument on the device: C, Bb, Eb, F.

This is **not** an **exhaustive list:**

C	• Accordion, Flute, Piccolo, Glockenspiel, Oboe Piano, Violin, Xylophone, Baritone, Euphonium, Tuba, Sousaphone, Trombone, Cello
Bb	• Clarinet, Bass Clarinet, Soprano and Tenor Saxophones, Trumpet, Cornet, Bugle/Flugelhorn
Eb	• Alto and Baritone Saxophone, Alto Clarinet
F	• French horn, English horn

Requirement Id	*REQ-1.001*
Requirement description	*All electronics and display should form one unique device.*
Requirement priority	*High*
Input	
Output	
Actors using the requirement	*Customer, software developers, hardware builders*

Requirement Id	*REQ-1.002*
Requirement description	*The information such as the note played, the frequency, played, the mode and the reference frequency are displayed on the LCD screen.*
Requirement priority	*high*
Input	
Output	*LCD screen*
Actors using the requirement	*software developers, hardware builders*

Requirement Id	*REQ-1.003*
Requirement description	*The accuracy is displayed via the Lixie part.*
Requirement priority	*high*
Input	
Output	*Lixie display*
Actors using the requirement	*software developers, hardware builders*

Requirement Id	*REQ-1.004*
Requirement description	*The user can switch between C, Bb, Eb, F modes.*
Requirement priority	*high*
Input	*Push button*
Output	*the tuner changes to the selected mode and the LCD update the info*
Actors using the requirement	*software developers, hardware builders*

Requirement Id	*REQ-1.005*
Requirement description	*The user can switch between different reference frequencies*
Requirement priority	*Medium*
Input	*Push button*
Output	*the tuner changes to the selected reference frequencies value and the LCD update the info*
Actors using the requirement	*software developers, hardware builders*

Requirement Id	*REQ-1.006*
Requirement description	*The user will be able to change the colours of the Lixie display.*
Requirement priority	*Low*
Input	*Colour picker via Android app*
Output	*The RGB LED light up with the corresponding colour*
Actors using the requirement	*software developers*

Requirement Id	*REQ-1.007*
Requirement description	*The user will be able to use the device while charging it.*
Requirement priority	*Low*
Input	
Output	
Actors using the requirement	*hardware builders*

Requirement Id	*REQ-1.008*
Requirement description	*The system has a Bluetooth module to transfer and receive data.*
Requirement priority	*Low*
Input	
Output	
Actors using the requirement	*hardware builders*

2. Chromatic mode and transposed mode

The chromatic mode (C mode) is the standard mode, mainly used in other tuner and let you tune piano, guitar, bass, violin, etc. The Bb mode let the user tune some band and orchestra instruments such as clarinet nad trumpet, Eb mode let the user tune saxophone and finaly the F mode used for French and English horn.

How does it work:
The chromatic mode is the one that corelate notes with frequencies.
When an instrument in Bb play a C we hear in reality a frequency coressponding to a Bb in chromatic mode.
When an instrument in Eb play a C we hear in reality a frequency coressponding to a Eb in chromatic mode.
When an instrument in F play a C we hear in reality a frequency coressponding to a F in chromatic mode.

Meaning that if not in chromatic mode, the tuner need to offset the value by:

- +2 semitone if in Bb mode.
- -3 semitone if in Eb mode.
- -5 semitone if in F mode.

C mode	**Bb mode**	**Eb mode**	**F mode**
A	B	F#	E
A#	C	G	F
B +2	C#	G#	F#
C	D	A	G
C#	D#	A#	G#
D	E -3	B	A
D#	F	C	A#
E	F#	C#	B
F	G -5	D	C
F#	G#	D#	C#
G	A	E	D
G#	A#	F	D#

3. Reference frequency

The frequency chart is built with only one frequency, called the reference frequency. The standard value is A=440Hz then A one octave higher is A=440x2=880 but between two octave there is 12 notes, so we need to subdivide this interval in 12 to do that we use power of two.

440*2^(-1/12)=415,30 440*2^(1/12)=466,16 440*2^(2/12)=493,88

Note\octave	0	1	2	3	4	5
C	32,7031957	65,4063913	130,812783	261,625565	523,251131	1046,50226
C#	34,6478289	69,2956577	138,591315	277,182631	554,365262	1108,73052
D	36,708096	73,416192	146,832384	293,664768	587,329536	1174,65907
D#	38,890873	77,7817459	155,563492	311,126984	622,253967	1244,50793
E	41,2034446	82,4068892	164,813778	329,627557	659,255114	1318,51023
F	43,6535289	87,3070579	174,614116	349,228231	698,456463	1396,91293
F#	46,2493028	92,4986057	184,997211	369,994423	739,988845	1479,97769
G	48,9994295	97,998859	195,997718	391,995436	783,990872	1567,98174
G#	51,9130872	103,826174	207,652349	415,304698	830,609395	1661,21879
A	55	110	220	440	880	1760
A#	58,2704702	116,54094	233,081881	466,163762	932,327523	1864,65505
B	61,7354127	123,470825	246,941651	493,883301	987,766603	1975,53321

But some musician would like to tune their instrument to A=432Hz for example wich give warmer and deeper tune due to the general lower frequency. The A=432Hz freq chart is then the following.

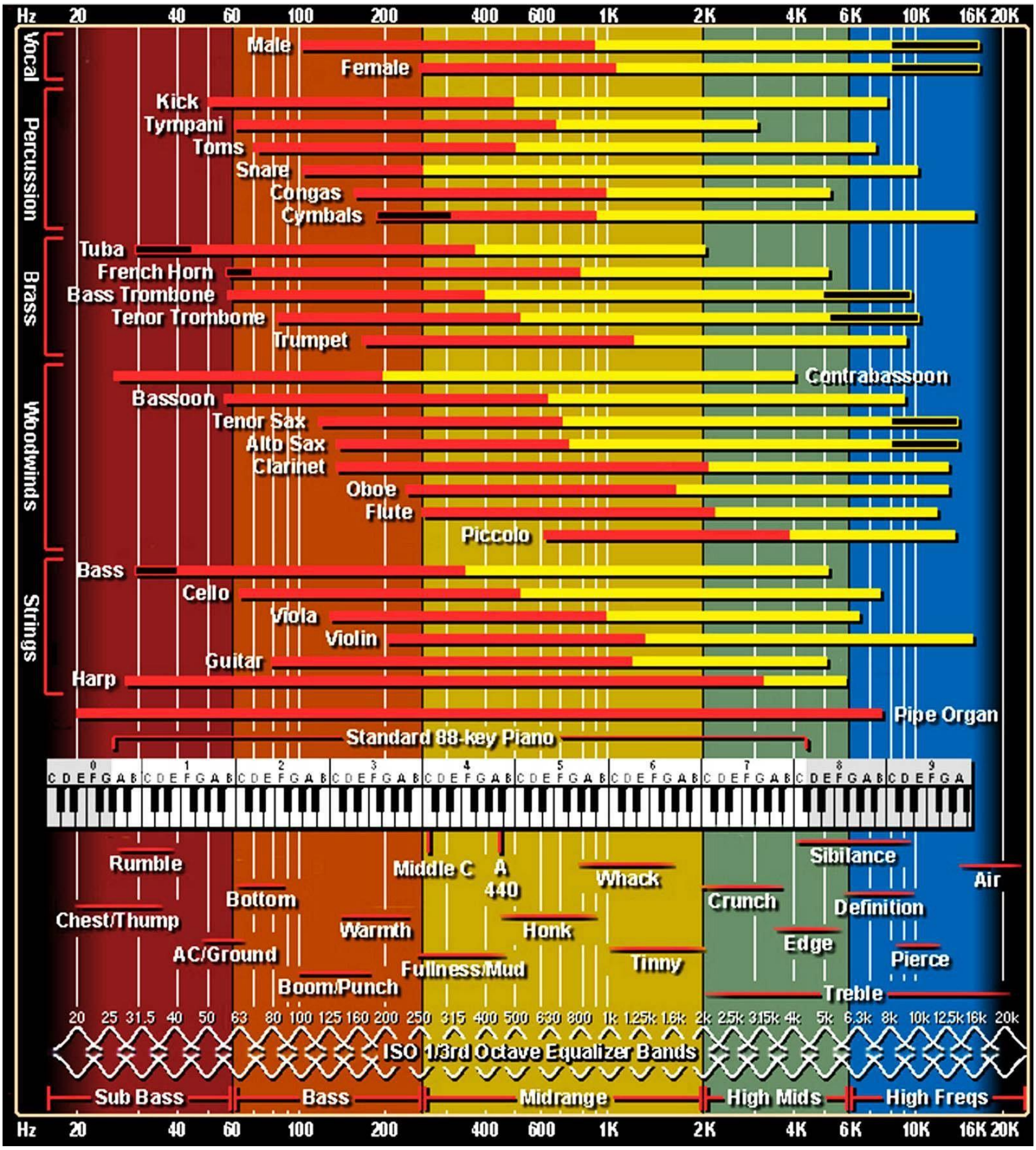
Note\octave	0	1	2	3	4	5
C	32,1085921	64,2171842	128,434368	256,868737	513,737474	1027,47495
C#	34,0178683	68,0357367	136,071473	272,142947	544,285894	1088,57179
D	36,0406761	72,0813521	144,162704	288,325409	576,650817	1153,30163
D#	38,1837662	76,3675324	152,735065	305,470129	610,940259	1221,88052
E	40,4542911	80,9085822	161,817164	323,634329	647,268657	1294,53731
F	42,8598284	85,7196568	171,439314	342,878627	685,757254	1371,51451
F#	45,4084064	90,8168128	181,633626	363,267251	726,534503	1453,06901
G	48,1085308	96,2170616	192,434123	384,868246	769,736492	1539,47298
G#	50,9692129	101,938426	203,876852	407,753703	815,507406	1631,01481
A	54	108	216	432	864	1728
A#	57,2110071	114,422014	228,844028	457,688057	915,376114	1830,75223
B	60,6129506	121,225901	242,451802	484,903605	969,80721	1939,61442

Our tuner will have the possibility to select between the 432, 434, 436, 438, 440, 442 and 444 Hz reference frequency.

4. Range of our tuner

Microphone range: 20Hz – 20KHz

Our tuner range is 20Hz to 2000Hz. The tuner has some difficulties with bass frequency due to the noise and the microphone-based solution, this problem could be fixed for electrical instrument with an input jack that will bypass the microphone. Otherwise in all the midrange our tuner is accurate and this is why our tuner is universal, since all the instrument cross this midrange.

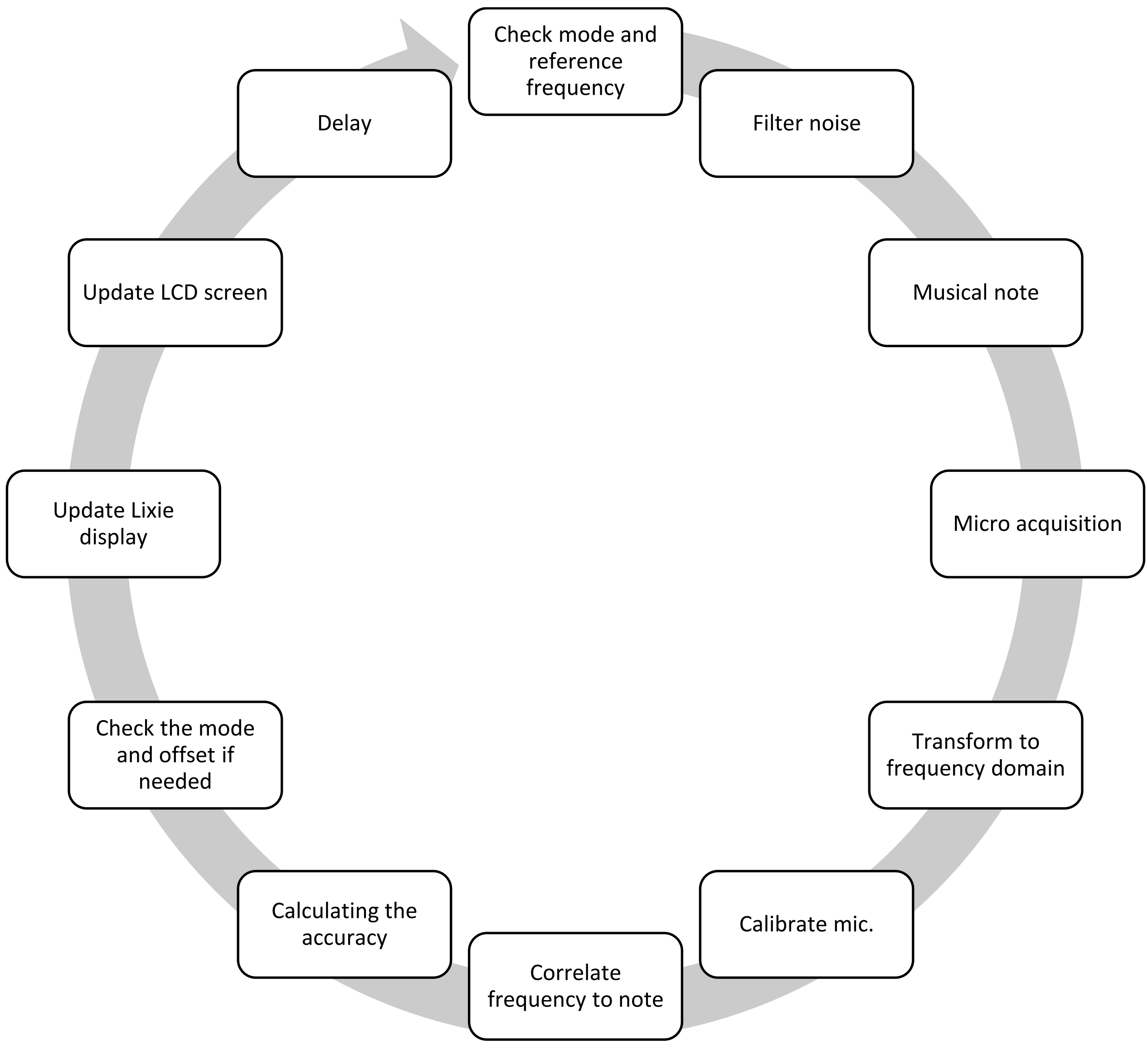


5. System architecture

The Arduino UNO board work by using, first, a setup (declaration of the variables, library). As well as a loop function, when all the instruction in this loop are executed it come back to the beginning.

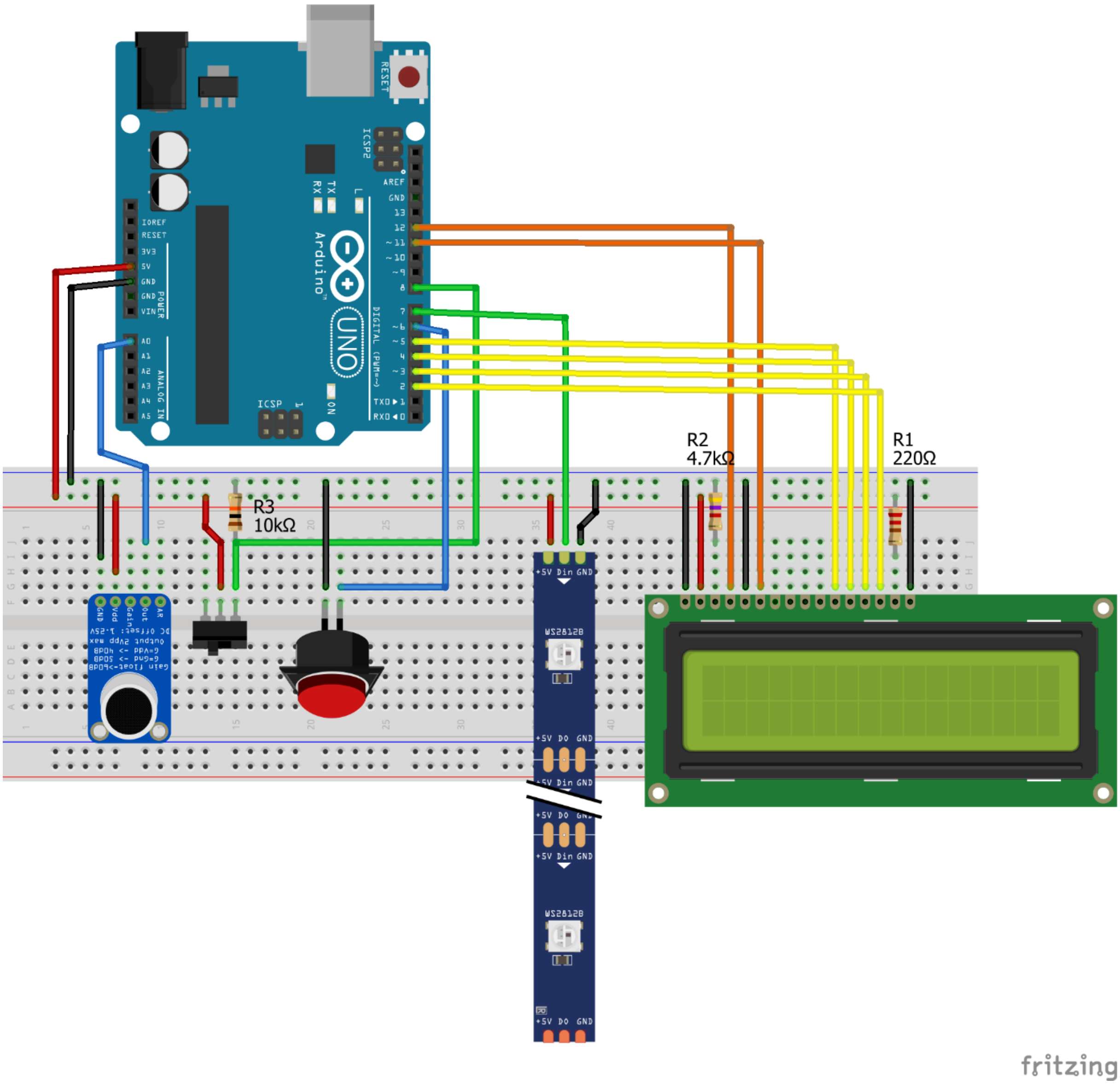
Inside this loop we want to:

- Check mode and reference frequency
- Filter noise
- Acquire the music note through our micro.
- Transform this time domain data into the frequency domain, perform FFT
- Calibrate microphone.
- Correlate the output frequency to the right note.
- Calculating the accuracy of the note
- Check in which mode we are, if not in chromatic mode, offset the value if needed.
- Light up the corresponding Lixie display and update the LCD screen to let the user know which note he his playing and with which accuracy.
- Add a delay.



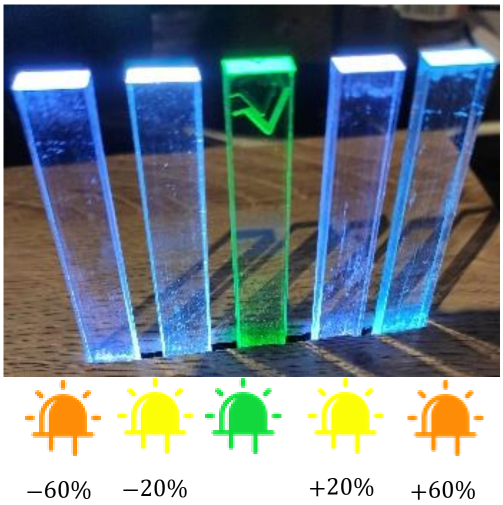
6. Wiring schematics

- The board is an Arduino Uno rev3 with his proto shield.
- The microphone is connected to the A0 pin of the Arduino. The arduinoFFT.h need to be included.
- The toggle switch let the user change between mode and reference frequency selection and is connected to the D8 pin of the Arduino.
- The push button let the user select the mode or the reference frequency and is connected to the D6 pin of the Arduino.
- The RGB WS2812 LED strip is connected to the D7 pin of the Arduino and is used to display the accuracy via the Lixie. The FastLED.h need to be included.
- The LCD screen need the LiquidCrystal.h to work.



7. Design

7.1 Lixie display



The note is in tune

The design was made under Adobe illustrator. The Helsinki based company Studio Laser Cut has cut and engrave a 5mm thick transparent acrylic for us.

How it works:
By putting LED on the bottom of an acrylic sheet, the light beams are reflected inside the acrylic sheet and stand out on the edges as well as on the engraving.

Dimension: 10cm by 1cm, 5mm thick
Used for displaying the accuracy (1 spare)

7.2 LCD screen

The LCD will display the note played, the frequency, as well as the mode and the reference frequency the user has selected.

7.3 Logo

LUT is the acronym for: Lixie Universal Tuner.

The Bold dark line design of the logo looks like the old Nixie tubes which our project re-uses the idea of.

The thin green line in the background represents electrical wires to remind that it is a hardware project with electronics inside.

7.4 Box

The box is built of cardboard because it is an easy material to work with and inexpensive. To hide the cardboard, we will use a fake wood adhesive.