INDEX

KEYWORDS

Pitch: Auditory perceptual property that allows the ordering of sounds on a frequency-related scale.

Orff instruments: Orff considered the percussive rhythm as a natural basic form of human expression. Music played on Orff instruments is often very simple and easy to play even for first time musicians.

Some of this instruments are miniature xylophones, marimbas, glockenspiels, and metallophones; all of which have removable bars, resonating columns to project the sound, and are easily transported and stored. Orff teachers also use different sized drums, recorders, and non- pitched percussion instruments to round out the songs that are sung and played.

Interference: Physic phenomenon in which two waves superpose to form a resultant wave of greater or lower amplitude. Interference usually refers to the interaction of waves that are correlated or coherent with each other, either because they come from the same source or because they have the same or nearly the same frequency. Interference effects can be observed with all types of waves.

Staff/Stave: In standard Western musical notation, the staff, or stave, is a set of five horizontal lines and four spaces that each represents a different musical pitch—or, in the case of a percussion staff, different percussion instruments. Appropriate music symbols, depending upon the intended effect, are placed on the staff according to their corresponding pitch or function

Fast Fourier Transform (FFT): decomposes a sequence of values into components of different frequencies.

Square wave: kind of non-sinusoidal waveform, most typically encountered in electronics and signal processing. An ideal square wave alternates regularly and instantaneously between two levels.

Jaime Ruiz Serra
INS Moianès

Boot loader: Central processor units can only execute program code found in read-only memory (ROM) or random access memory (RAM). Modern operating systems, application code, and data are stored on nonvolatile data storage devices (hard drives, CDs, DVDs, SD cards or USB flash drives).When a computer is first powered on, it usually does not have an operating system in ROM or RAM. The computer must execute a relatively small program stored in ROM, along with the bare minimum of data needed to access the nonvolatile devices from which the operating system programs and data may be loaded into RAM.

The small program that starts this sequence is known as a *bootstrap loader*, *bootstrap* or *boot loader*. This small program's only job is to load other data and programs which are then executed from RAM.

Wiring: Open source electronics prototyping platform composed of a programming language, an integrated development environment (IDE), single-board microcontroller documentation related with designers and artists in mind and a community where experts, intermediate and beginners from around the world share ideas, knowledge and their collective experience.

Processing: Open source programming language and integrated development environment (IDE) built for the electronic arts and visual design communities with the purpose of teaching the basics of computer programming in a visual context, and to serve as the foundation for electronic sketchbooks. One of the stated aims of Processing is to act as a tool to get non-programmers started with programming, through the instant gratification of visual feedback. The language builds on the Java programming language, but uses a simplified syntax and graphics programming model.

Piezoelectricity: Charge that accumulates in certain solid materials (notably crystals, certain ceramics, and biological matter such as bone, DNA and various proteins) in response to applied mechanical stress. The word piezoelectricity means electricity resulting from pressure. It has many useful applications such as the production and detection of sound, generation of high voltages, electronic frequency generation, microbalances, and ultrafine focusing of optical assemblies. It is also the basis of a number of scientific instrumental techniques with atomic resolution, and everyday uses such as acting as the ignition source for cigarette lighters and push-start propane barbecues.

Joule effect: Physical law expressing the relationship between the generated heat in a conductor and current flow, resistance, and time.

Ohm's law: Ohm's law states that the current through a conductor between two points is directly proportional to the potential difference across the two points. More specifically, Ohm's law states that the R in this relation is constant, independent of the current. In physics, the term Ohm's law is also used to refer to various generalizations of the law originally formulated by Ohm.

PCB: A printed circuit board (PCB) is used to mechanically support and electrically connect electronic components using conductive pathways or tracks etched from copper sheets laminated onto a non-conductive substrate.

INTRODUCTION

The aim of this project was to build a highly robust electronic drum kit. The main objective during the design process was to make it simple, cheap and versatile, respecting the basic qualities of a standard drum kit.

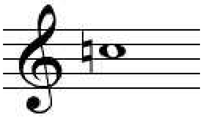
I wanted to do an interactive music-related project. That remind me about a little experiment we did in a summer science program, where we used an *Arduino Duemilanove* and a pressure sensor to make rhythm graphics and statistics. Furthermore, my younger brother had just acquired an electronic drum kit and that gave me the opportunity to see how they work – and actually realize how simple their hit sensor behavior was.

Once I had decided my objective for the project I followed the technologic process. My intention to build an electronic drum kit was the first point of the process. Second point was explore any ideas and designs to carry out with the project. Then, as technologic process says, I planned my work and finally built the drum kit. This was the hardest point and that gave me lots of ideas of possible applications of the drum hardware and components for future work.

To make the research project more complete I increased the scope of the areas I was going to focus in, so that I would not focus just in the essential to do the work – the branch of electronics. That's why this work is done in English, and that's why it has a part of music, science and design in addition to the electronics part and the main part of the project, where the creation process is explained.

So before you keep on reading I apologize for my basic, and many times wrong, written English.

To completely understand – and enjoy – this paper I recommend to have a look to the keywords in the previous pages.



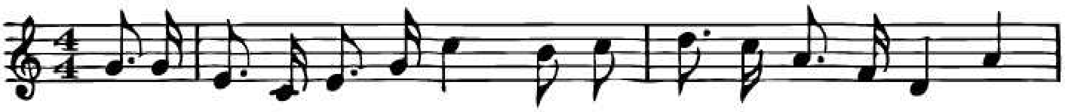
1. THEORETICAL PART

1.1. MUSIC

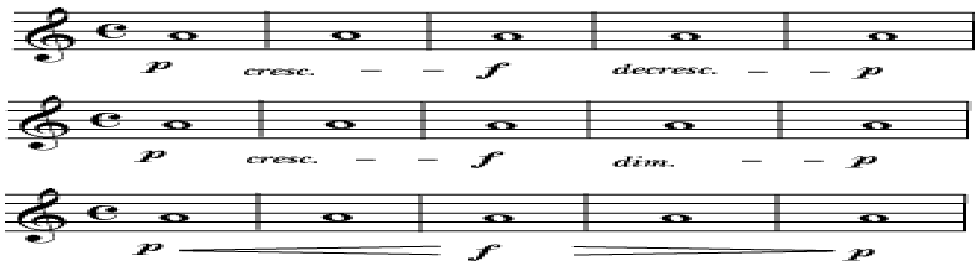
The concept of *music* comes from the Greek word *musiké* (*μουσική*, which contains the concept *muse*), by which the Ancient Greeks understood, first, the arts of the muses (poetry, music, and dance as a block), and then *the art of the sounds* concretely. In the history of music, it has always been related to language and dance, but a self-contained musical phenomenon was developed with *instrumental music*.

Music can be defined as an art expressed by sound and silence which is concerned by combining different elements such as *pitch* –which governs melody and harmony–, *rhythm* –and its associated concepts tempo, meter, and articulation–, *dynamics*, and *sonic qualities* –timbre and texture–.

In a music score, the higher the note is placed, the higher is the pitch of this note.

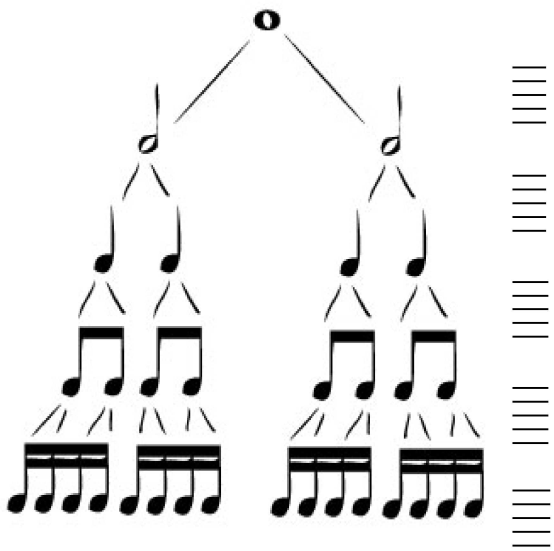


Two notes in the same space or line can have different pitches when one of them is altered (*sharp* notes are more acute than natural ones and *flat* notes are more grave than natural ones). When a note in the same bar of an altered one has to be natural, we put the natural symbol before it to show that it's not altered.



Dynamics are what tell the musician what kind of expressivity has to give to the fragment of the score he’s reading (referring to volume, velocity and character). They are usually annotated above or below the staff.

Referring to the rhythm, there’s a code for note durations, where every note type has the duration of a portion of a *whole note*.



musical instrument family closely

1.1.1. Percussion

The word "percussion" comes from Latin terms: *percussio* (“to beat”, “strike" in the musical sense), and *percussus* (noun meaning "a beating").

In a musical context the term "percussion instruments" is used to describe the instrumental family formed by any object which produces a sound when it is hit with a stick or when it is shaken, rubbed, scraped, or otherwise acted upon in a way that sets the object into vibration.

We are not wrong if we think that percussion instruments were probably the first musical appliances used by human because of their simple operation way. All along history, percussion instruments have been modified and have evolved in different ways. Nowadays, we can classify them by different criteria.

By methods of sound production

Idiophone: (From Greek *ἰδιαφωνη*, “own sound”). The sound is produced by the vibration of the entire instrument’s body.

Membranophone: (From Greek *μεμβραφωνη*, “membrane sound”). The sound is produced when the membrane or head is struck.

<u>Chordophone</u>: (From Greek *χορδο*, “string”, and *φωνη,* “sound”). The sound is produced by hitting one or more strings. Most of them are considered *string instruments* rather than *percussion instruments*.

<u>Aerophone</u>: (From Greek *ἀεροφωνη*, “air sound”). The sound is produced by an air column vibrating inside an object. Many people consider them *wind instruments*.

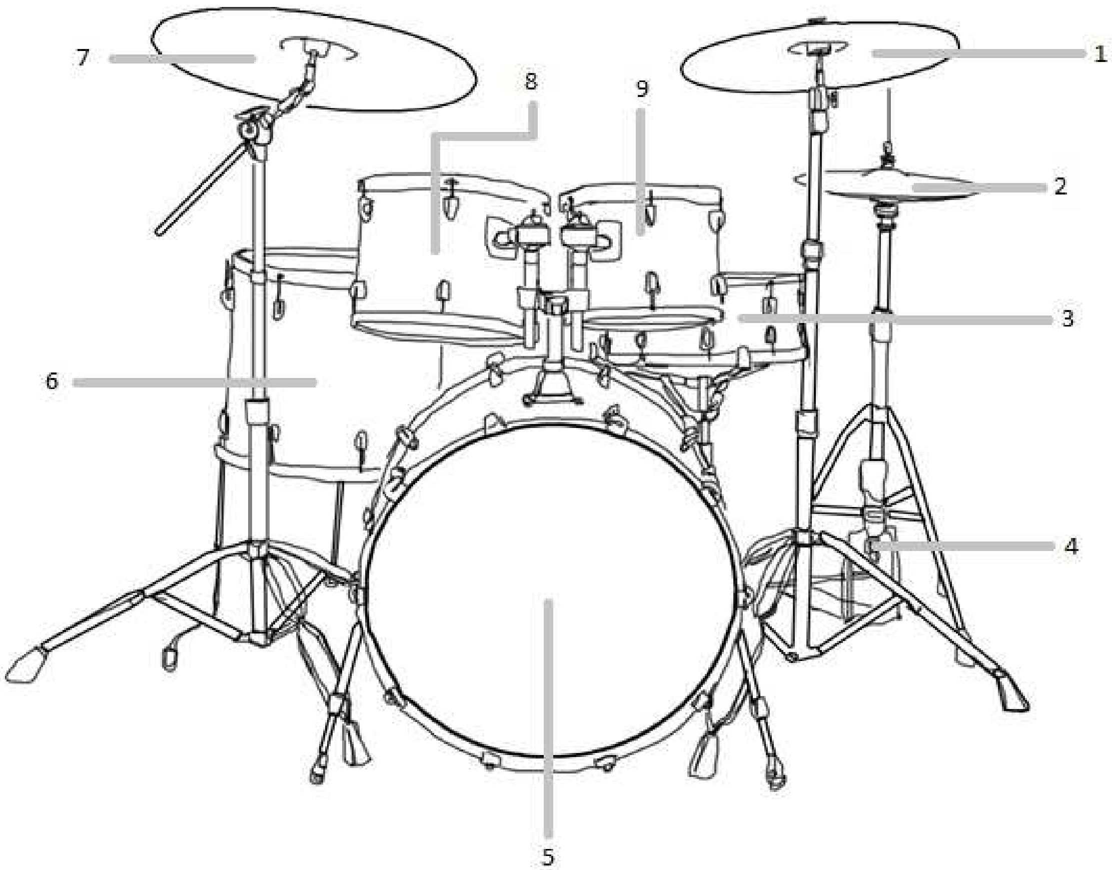
By the sound they produce

<u>Definite pitch</u>: The sound emitted by these instruments has a measurable pitch.

<u>Indefinite pitch</u>: The sound of the instrument contains complex frequencies that make the pitch immeasurable.

1.1.2. Drum Kit

Drum kits are sets of drums, cymbals and sometimes other percussion instruments arranged to be played by just one person. Each component of the drum kit is hit by the percussionist with a stick, a brush or a ramrod, apart from the bass drum, which is hit by the drummer with a pedal usually placed on the right foot position; and the hi hat opening, which is controlled by another pedal placed in the other foot of the player.



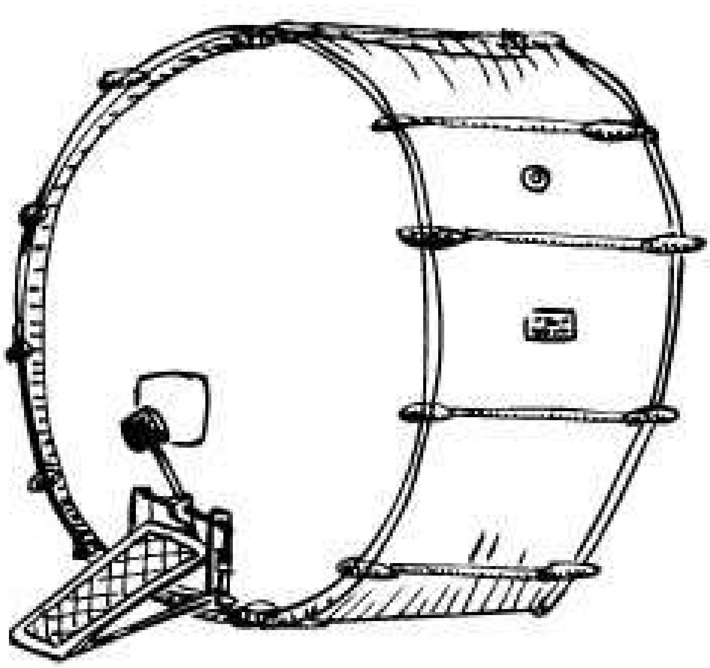
Drum kits as we know them nowadays were born in the 20th century. The main instruments of the drum kits were used in marching bands and military bands, but they were played one by musician. The birth of the drum kits is directly related to *Jazz*, where, before 1910, there were 3 o 4 percussionists in each band (one for the snare drum, one for the bass drum and one or two for the other percussion instruments like cymbals or cowbells).

The inventions of the snare stand and especially of the bass drum pedal (marketed in 1910 by Ludwig) allowed playing more instruments simultaneously and thus reduce the number of percussionists to one per band.

When the *Ragtime* and the need of musicians for the dance halls arrived, the assembly of the drum kits as they are known today finished taking shape and so they could be played by just one drummer.

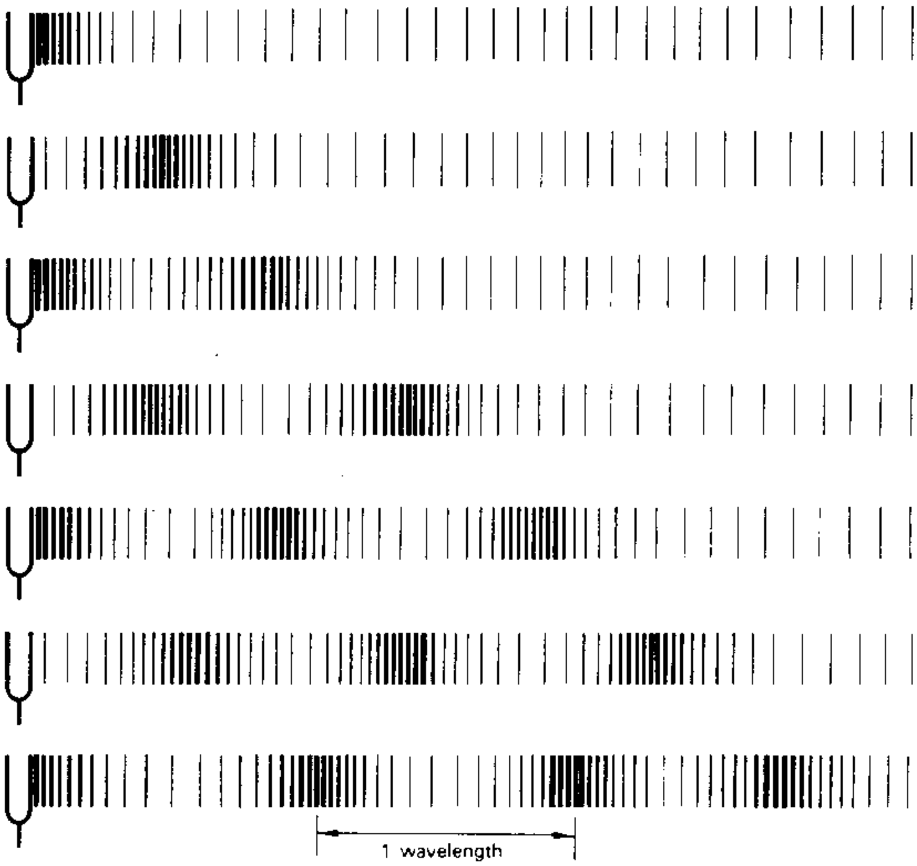
The number of components a drum has and their placement depends on the style of music it is going to play and the preferences of the percussionist. You can see the basic components of a drum kit in the previous page.

If you want to see some examples of drum sets for different music styles check out the *A annex*.



Bass drum pedal allows the drummer hit the bass drum with his/her foot.

1.2. SOUND (science)

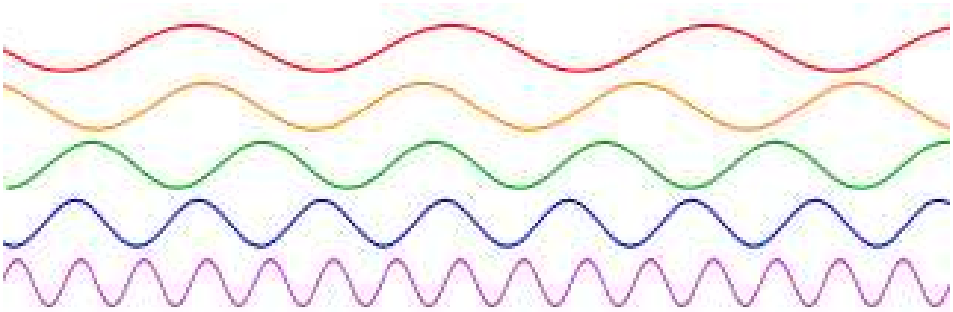


According to physics, sound is a variation of pressure transmitted through a solid, liquid or gas medium as a mechanical wave that can be perceived. During propagation, waves can be reflected, refracted, or attenuated by the medium. Density, temperature or pressure of the medium can affect the sound transmission. Movement in the medium can change transmission results as well.

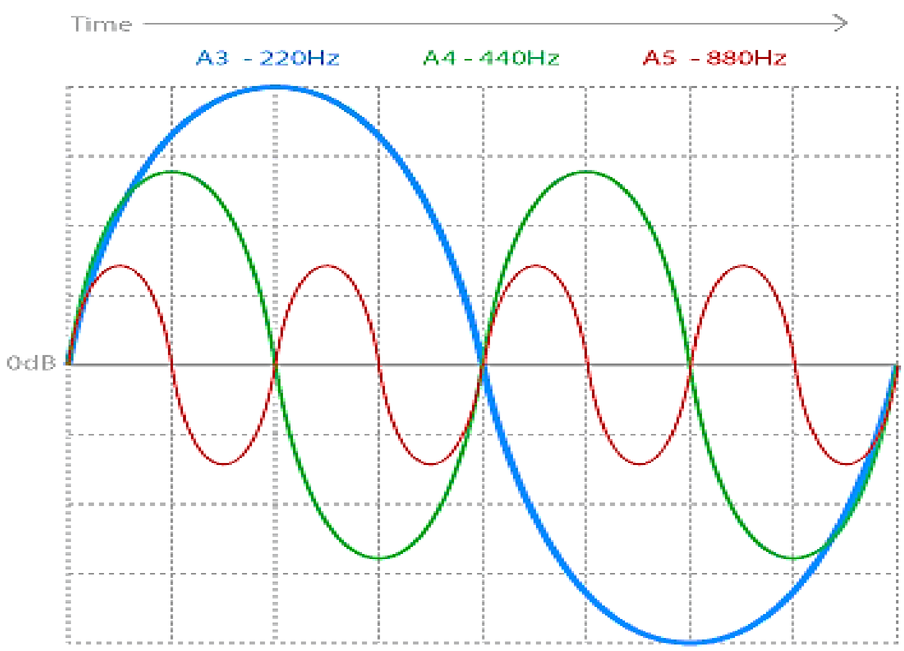
Sound waves travel through the medium by the movement of its molecules, which separate and come together in the medium transmitting sound waves' energy.

Jaime Ruiz Serra
INS Moianès

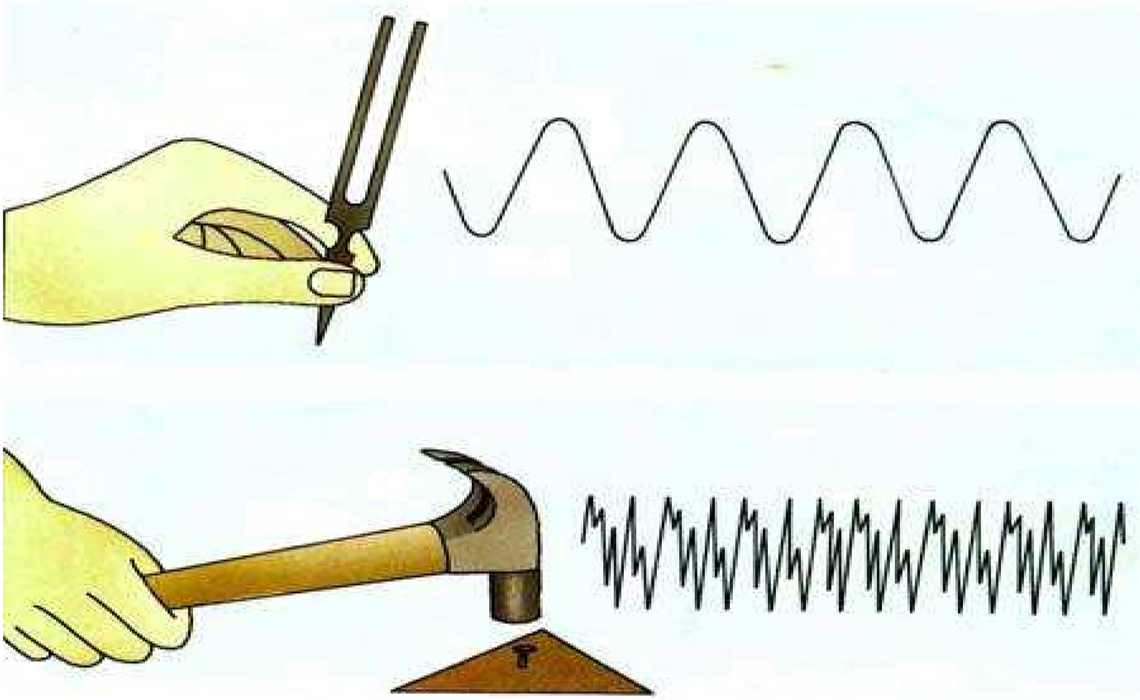
Sound waves are defined by really different factors. The most important qualities of sound waves are:



Frequency: Frequency is the number of occurrences of a repeating event per unit time. It is also referred to as temporal frequency. The period is the duration of one cycle in a repeating event, so the period is the reciprocal of the frequency. Frequency is measured in Hertz (Hz). Human hearing can perceive sounds between 20 Hz and 20 kHz (20.000 Hz). Sounds from 0 to 20 Hz of frequency are known as *Infrasound*, and those with a frequency higher than 20 kHz are known as *Ultrasound*.

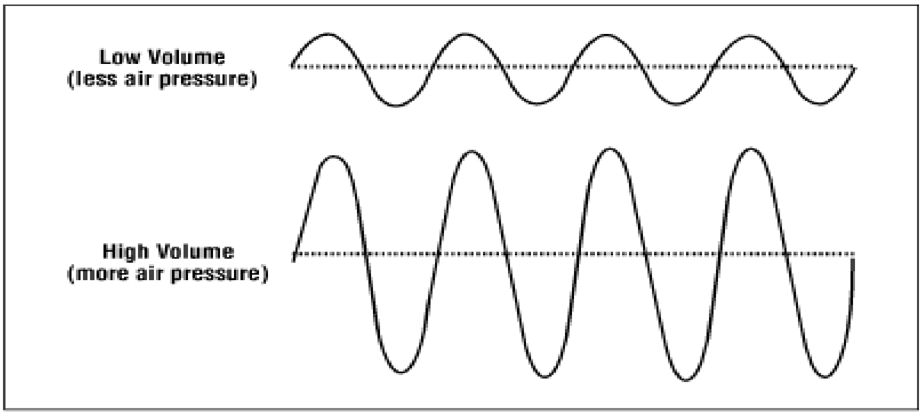


Frequency is actually the pitch of a sound. Notes in music have a related pitch (440Hz is A4) and notes in different octaves have twice the frequency of the same note in a lower octave, so 220Hz is A3, 110Hz is A2, 880Hz is A5, and so on.



Sound waves have harmonics. Harmonics are component frequencies of a sound wave. The pitch of a sound wave is the fundamental frequency *f*, and the harmonics giving tone characteristics to the sound are *2f, 3f, 4f,* etc.

Noises are complex sounds with different frequencies and without any harmonic relationship.



Loudness: The loudness of a sound depends on its wave's amplitude. Amplitude in sound waves is the magnitude local pressure deviation from the ambient atmospheric pressure caused by a sound wave. It can be measured using a microphone. The SI unit for sound pressure *p* is the Pascal (Pa). The greater is the amplitude, the greater is the impact of the air molecules on human's ear and so it's the sound.

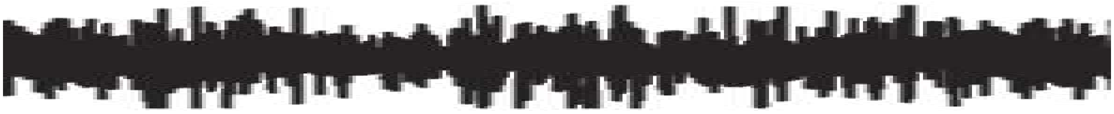
Jaime Ruiz Serra
INS Moianès

Sound pressure level (SPL) or sound level is a logarithmic measure of the effective sound pressure of a sound relative to a reference value. It is measured in decibels (dB) above a standard reference level. The commonly used 0 dB reference sound pressure in air is 20 µPa (at 1 kHz).

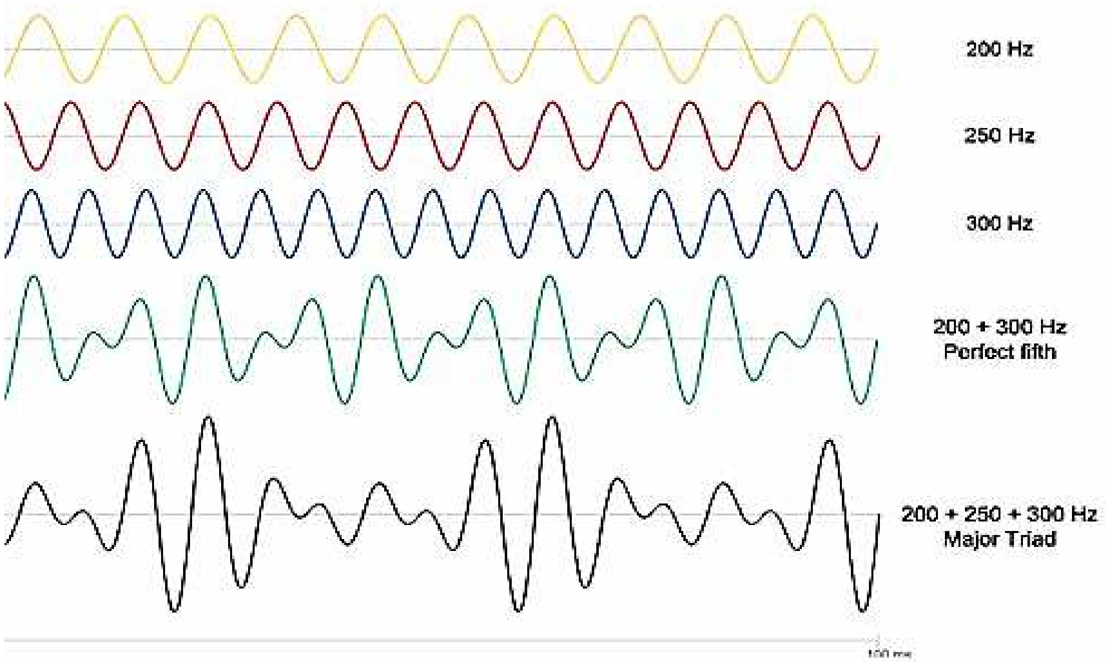
LOUDNESS LEVELS	
Hearing threshold	**0 dB**
Breathing	10 dB
Whisper	20 dB
Quiet library	30 dB
Refrigerator noise	40 dB
Office	50 dB
Conversation	60 dB
Street traffic	70 dB
Noisy restaurant	75 dB
Carpet sweeper	80 dB
Long exposure may cause hearing loss	**90 dB**
Subway	90 dB
Mp3 player at half volume	95 dB
Motorcycle	105 dB
Bass drum	110 dB
Baby crying	115 dB
Pain threshold	**120 dB**
Disco	120 dB
Cymbal crash	120 dB
Rock concert	130 dB
Plane takeoff	140 dB
Chest wall vibrates	**150 dB**
Hearing Tissue breaking	**180 dB**
Space shuttle takeoff	215 dB

Some facts about SPL due to it is a logarithmic measure:

- At double distance of the sound focus, we would perceive just 6 dB less than in the initial point.
- Two sounds with a loudness of 100 dB each would generate 103 dB (not 200).



Speed: Speed (velocity) of sound is the distance traveled by a sound wave in a unit of time. Speed of sound in air is 340 m/s and can change depending on its temperature. The density of the medium affects radically the speed of sound transmission. The closer are the molecules in a substance, the more speed the sound acquires, and vice versa.

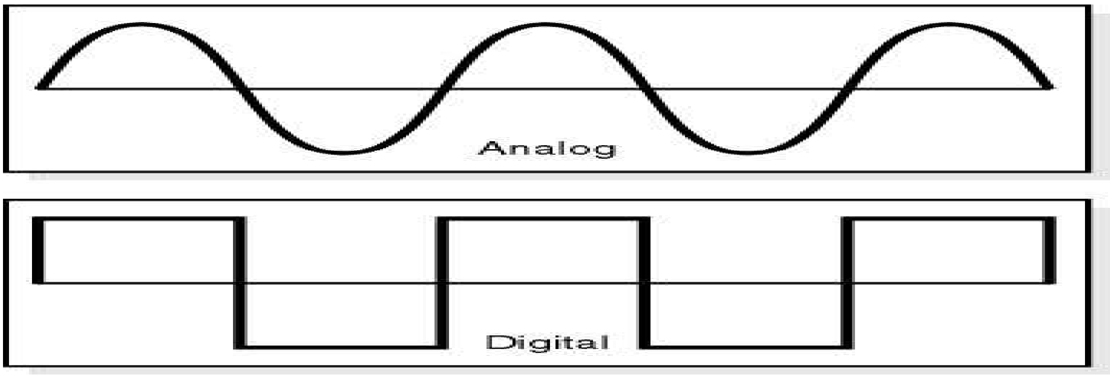


Timbre: The same note (in frequency and loudness) played in two different instruments would sound quite different. That's because of the harmonic loudness proportion in each sound wave. If a FFT is made to two different sound waves (A and B in next page's example) with the same pitch to have a spectral view of that wave, this phenomenon can be seen.

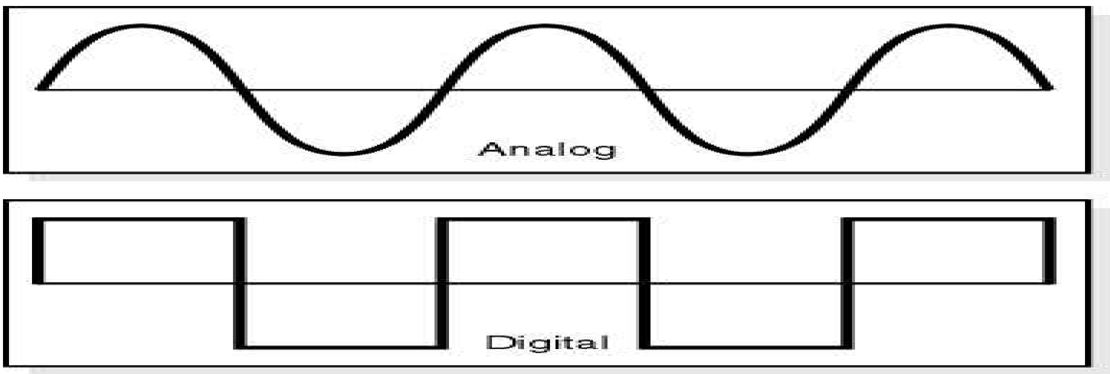
Despite being two sounds with the same frequency (pitch), their harmonic spectrums are different due to their different timbre.

1.3. ELECTRONICS

Electronics is the branch of technology or science that works with electrical circuits which contain many electrical components such as resistors, capacitors and diodes among others. There are two types of circuits:



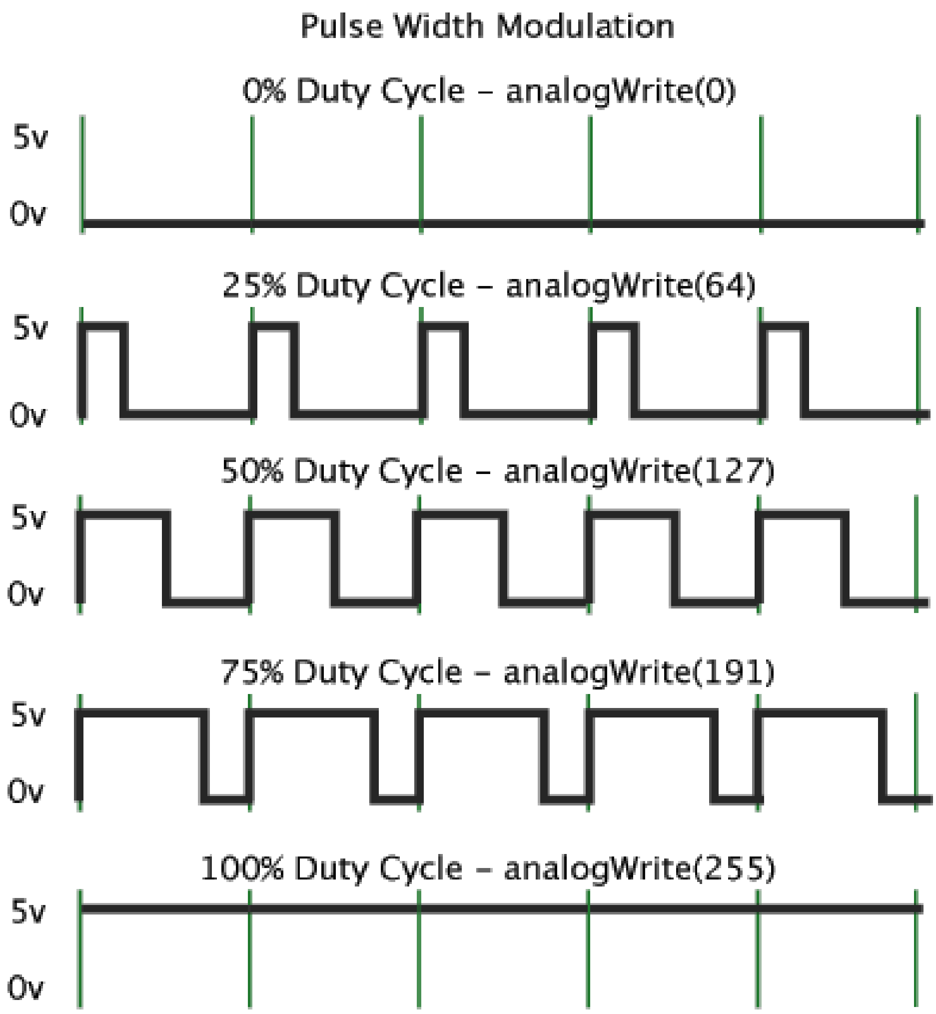
Analog circuits: (From Greek *ανάλογος (analogos)*, “proportional”). The signal is continuously variable and can include infinite values. It has to represent information of other physical forms, such as sound, temperature, pressure or light. It is represented as a proportional voltage, current, frequency or charge.



Digital circuits: The signal is represented by discrete bands of analog levels. All levels within a range equivalent to the same signal logic state. Small changes in the analog signal levels (interferences or manufacturing tolerance) don’t affect the signal state. In most cases there are two logic states (1/0; which can be interpreted as true/false, yes/no, open/close, on/off or others) represented by two voltage ranges (0 ≈ 0 volts; 1 ≈ V_{max} volts) with a gap between them to avoid interference errors.

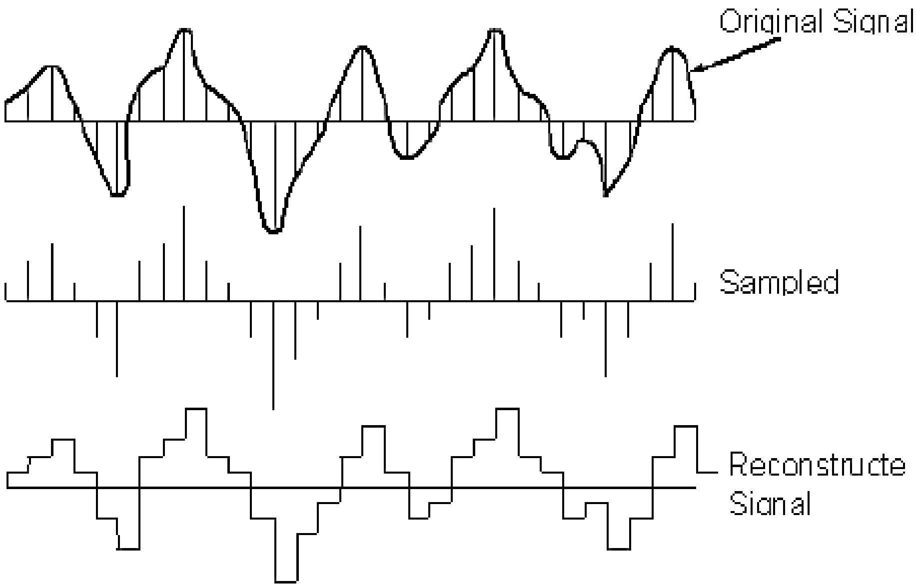
- Pulse-width modulation: (*PWM*) Is a technique to simulate analog outputs in a digital circuit. The digital control generates a square wave –a signal which commutes between *on* and *off* logic states. This *on/off* pattern can simulate voltages from 0 to V_{max} volts just by changing the time proportion between *on* and *off*. We call *pulse-width* the duration of the time that the signal is *on*. To change the analog value we change this *pulse-width*. If we control this pattern fast enough we can *modulate* the analog output signal.

In the picture below we can appreciate the operation of PWM. Green lines represent a regular period. In that particular case the frequency is about 500Hz, so the period is 2 milliseconds approx. The function analogWrite( ) must have a value between 0 and 255, where 255 is the 100% of the period on and 0 is the whole period off. Vmax = 5V.

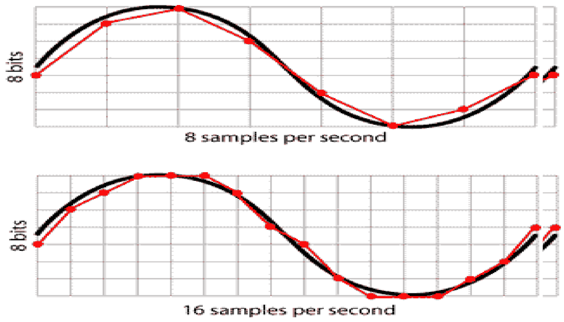


1.3.1. Digital audio

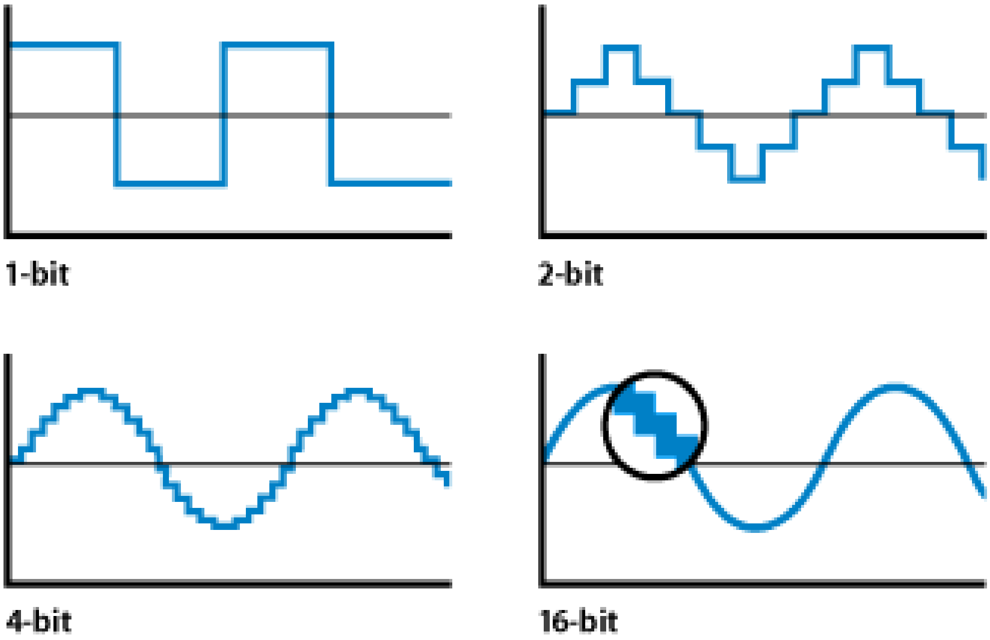
Computer equipment cannot work with analog data. That's a problem when working with sound due to it is a wave. To solve this problem analog wave's data has to be converted into digital data (ADC or *Analog-Digital Conversion)*.



When a microphone captures sounds they are transformed into current variations in an analog circuit. This voltage is periodically measured and the obtained values are rounded to a finite value group. Each digitized sample of audio is assigned a value that corresponds to the amplitude of the analog wave. That way we obtain concrete values of the wave for every portion of time. Then, these values are registered in a memory support.



Sampling rate (or sampling frequency) is the number of measurements of an audio signal per unit of time. The more sampling rate, the higher is the fidelity to the original analog wave's amplitude.



Bit depth is the accuracy used when rounding values in ADC. The more bits, the higher is the fidelity to the original analog wave's temporal resolution.

To reproduce a concrete frequency, sampling rate has to be at least twice this frequency. CD sampling rate is 44.1 kHz, so the highest frequency that can be reproduced in a CD is 22.05 kHz. It is enough for human hearing, that can hear sound with frequencies under 20 kHz.

1.3.2. MIDI

MIDI means *Musical Instrument Digital Interface*. It's a standard industrial protocol which allows computers, synthesizers, samplers and some other musical electronic devices to share information between them and generate sounds.

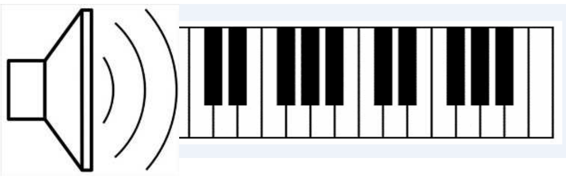
In the 70's there were a few digital controlled systems, but any of them was standardized, and that made it difficult to communicate devices with different systems. MIDI appeared in 1983, when its normative, which defined things like the format of the connectors, was written. Then several companies started producing MIDI-compatible devices. In the early 90's appeared PCs, sound cards and some other technological advances that helped redefining the MIDI protocol.

MIDI information contains different data types that define particular notes, volumes, channel numbers or device values. This simple data transmission method makes it easier to work with musical information between devices. MIDI works at a sampling rate of 31250 bits per second.

We've got to notice that MIDI information is not a kind of audio signal. It just transmits event data and device messages, which can be interpreted arbitrary depending on the receiver's presets. On balance, MIDI can be defined as a music score that contains the instructions in number values (from 0 to 127) that say when a sound has got to be generated and his features. A synthesizer is what converts this information in sounds.

MIDI devices just can transmit information from a *master* device (which sends activation messages) to a *slave* device (who interprets this information). They can be classified in three groups:

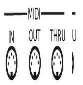
- Controllers: They produce MIDI messages from the musical interpretation of the user. MIDI controllers are usually mechanic equipment which adopts musical instrument's shape – such as keyboards, drum kits or guitars.
- Synthesizers: Sound generating units. They receive MIDI messages and convert them into sound signals. Samplers and sequencers are examples of synthesizers.



Referring to software, MIDI instruments can be monophonic/polyphonic; single-pitched/multi-pitched and mono-timbral/poly-timbral.

Monophonic instruments are those which can produce one sound at a time (E.g. trumpet), while polyphonic instruments are those which can produce more than one sound at a time (chords) (E.g. guitar or piano).

Single-pitched instruments can produce sounds of a concrete pitch (E.g. snare drum), while multi-pitched instruments can produce sound with a range of pitches (E.g. violin).

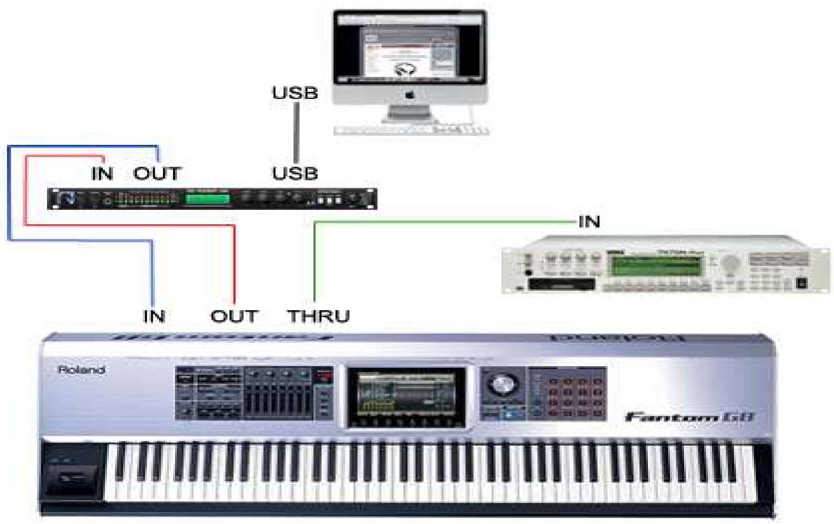


-timbral instruments can produce the sound of just an instrument (E.g. ; while poly-timbral instruments can produce the sound of different instruments (E.g. drum kits).

MIDI devices can have three different connectors.

<u>MIDI IN</u>: From where the device receives data from other devices.

<u>MIDI THRU</u>: The data arriving through the MIDI IN port is sent through this port without any changes. The main utility for this port is to chain different midi devices, but the more devices chained, the less is the quality of the data sent.

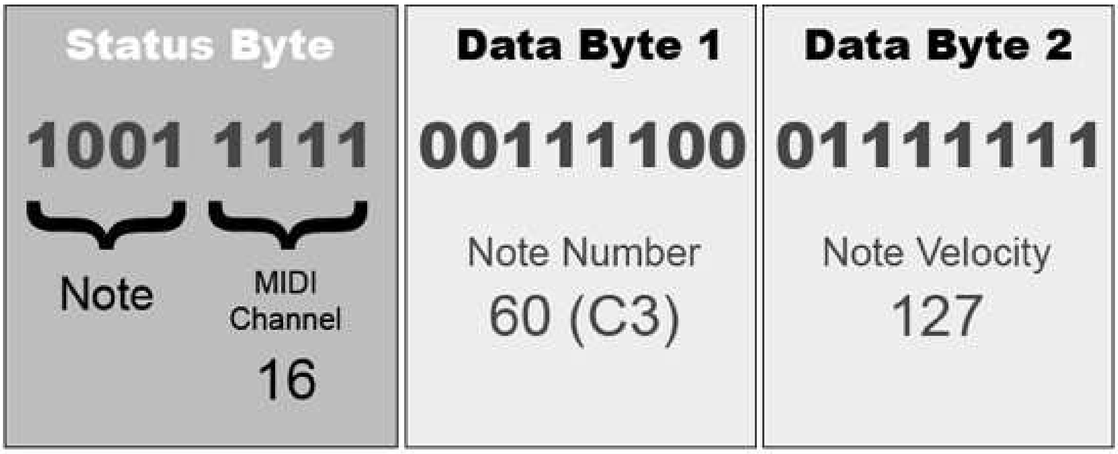


<u>MIDI OUT</u>: The generated data in a device is sent through this port.

1.3.3. MIDI messages

MIDI messages are formed by MIDI bytes. MIDI bytes are formed by 10 bits – unlike standard bytes, which are formed by 8 bits.

First bit is start bit, which is always 0; and last bit is stop bit, which is always 1, so that MIDI devices can count sent and received bytes. The remaining eight are the MIDI message containers. MIDI bytes can be *status bytes* or *data bytes*.
Status bytes values are equal or higher than 128 (01xxx xxxx1) in binary. They are used to transmit instructions, such as *note on, note off* or *pitch* bend.
Data bytes values are between 0 and 127 (00xxx xxxx1) in binary. Their mission is to transmit information like *tone*, *velocity* (volume), etc.



When we press a key in a MIDI keyboard a MIDI message is sent. This message starts with a status byte (*noteOn*) reporting that the note is activated and which channel is being used. This byte is followed by a data byte containing the pitch or name of the note and his velocity, and finally the note is 'released' with another status byte (*noteOff*) or simply setting the velocity to 0.

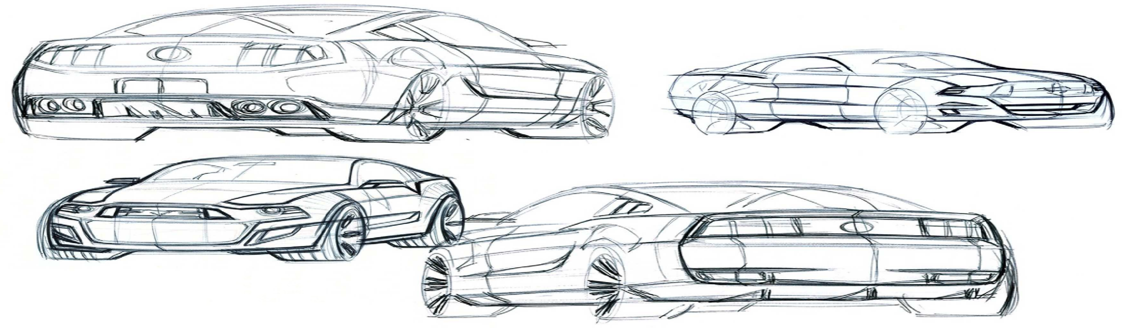
In this project we have focused in MIDI's channel for percussion instruments (channel 10) to work with the different drum kit sounds. Each of the 128 different possible note numbers is interpreted as a separate, different instrument, and the percussion sound's pitch is not related to the note number. The different instruments in channel 10 are follows:

35 Bass Drum 2	**59** Ride Cymbal 2
36 Bass Drum 1	**60** High Bongo
37 Side Stick/Rimshot	**61** Low Bongo
38 Snare Drum 1	**62** Mute High Conga
39 Hand Clap	**63** Open High Conga
40 Snare Drum 2	**64** Low Conga
41 Low Tom 2	**65** High Timbale
42 Closed Hi-hat	**66** Low Timbale
43 Low Tom 1	**67** High Agogô
44 Pedal Hi-hat	**68** Low Agogô
45 Mid Tom 2	**69** Cabasa

46 Open Hi-hat	**70** Maracas
47 Mid Tom 1	**71** Short Whistle
48 High Tom 2	**72** Long Whistle
49 Crash Cymbal 1	**73** Short Güiro
50 High Tom 1	**74** Long Güiro
51 Ride Cymbal 1	**75** Claves
52 Chinese Cymbal	**76** High Wood Block
53 Ride Bell	**77** Low Wood Block
54 Tambourine	**78** Mute Cuíca
55 Splash Cymbal	**79** Open Cuíca
56 Cowbell	**80** Mute Triangle
57 Crash Cymbal 2	**81** Open Triangle
58 Vibra Slap	

1.4. DESIGN

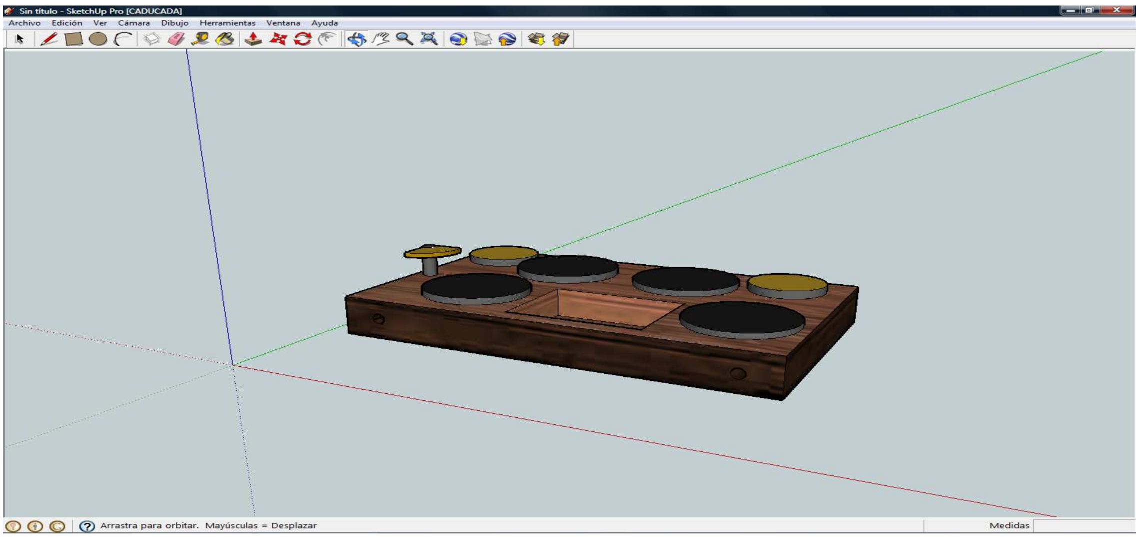
Design (from Italian *disegno*, “drawing”) is the mental configuration process — *pre-figuration* — which looks for a solution in any field in the context of arts, engineering, architecture, packaging and many other creative disciplines.



His action refers to the investigation and development process whereby an object or a communication tool is created or modified. Moreover, it has functional and esthetic considerations which are determined by multi-field factors.

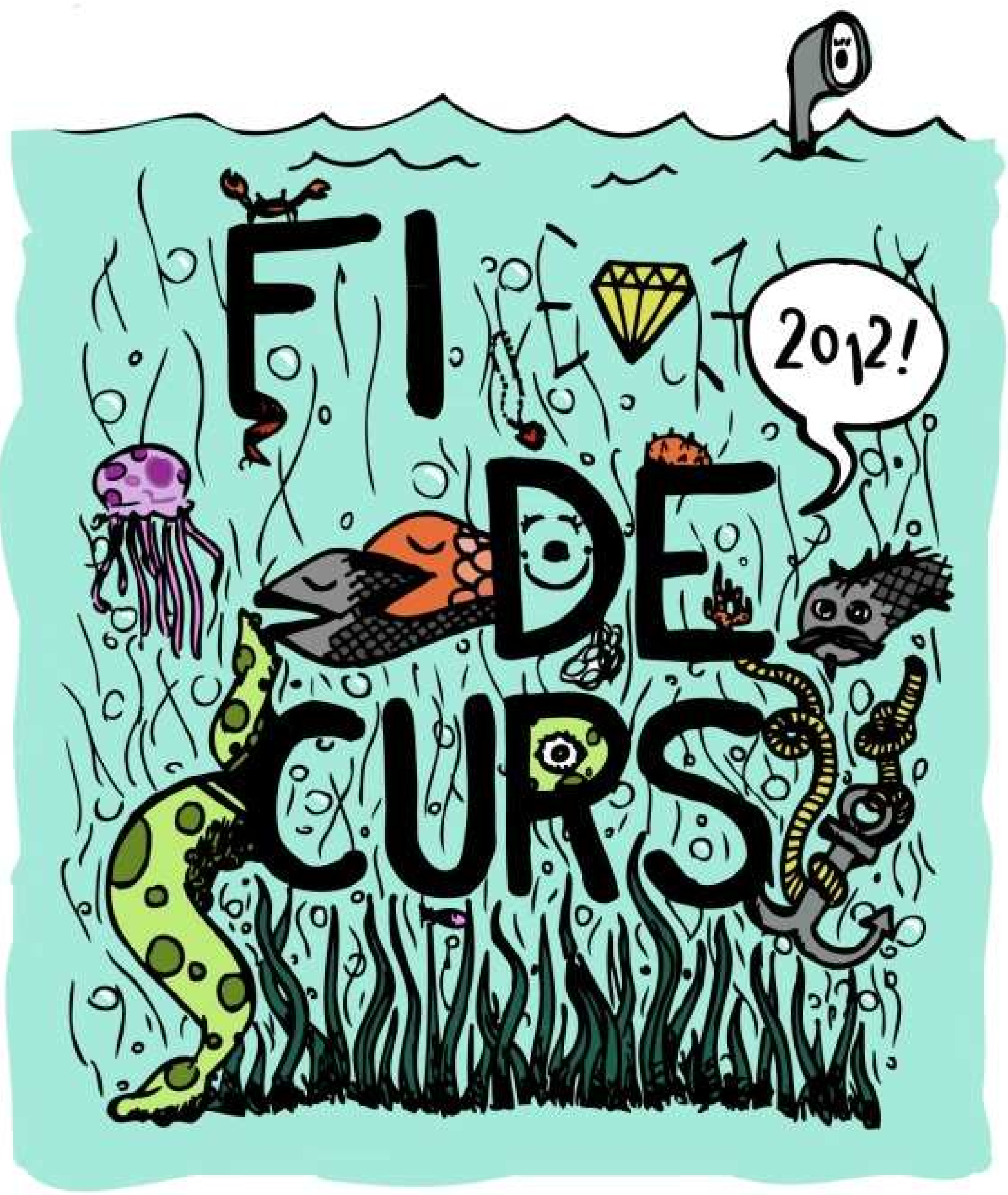
Design has several categories; each one specialized in different disciplines. In this project, we have focused on *product* and *graphic* design.

1.4.1. Product design



Product design is a discipline of design where products are created, developed or improved with the objective of making them functional and visually attractive. His objective is to satisfy human’s necessities by adapting not just objects’ shape but also their applications, functions and structural and esthetic qualities, looking for a final improved product.

1.4.2. Graphic design



Graphic design is de branch of design that creates visual messages in any support. It is used as a tool to graphically express ideas, facts and values. There are different categories in graphic design, such as advertising design, editorial design, corporative image design, web design, package design, typographic design and multimedia design, amongst others.

2. PRACTICAL PART

The main part of this project is the practical part, where we explain the process of design and creation of the electronic drum kit. This part is divided in three subparts where the different steps of the process are shown.

2.1. Hardware

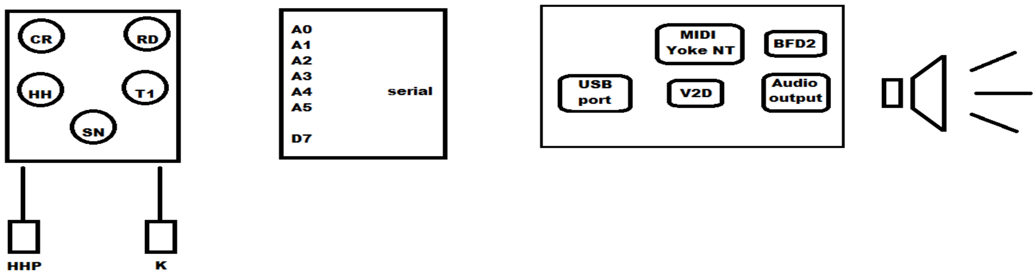
Hardware is the tangible part of the drum. It ranges from microcontroller boards to any physical element involved. The different parts of the hardware are going to be explained in this part, including their properties and design.

2.1.1. Working method

In the experiment we did in the science program we used a pressure sensor as hit sensor. These sensors are variable resistors that offer more resistance when they are under pressure. To work that way, a little current has to pass constantly through the resistor all the time. *Arduino Duemilanove* microcontroller board received data from the pressure sensor and sent it to a PC.
For the drum kit I preferred using piezoelectric transducers as hit sensors because they work pretty the other way round and that simplifies the labor.

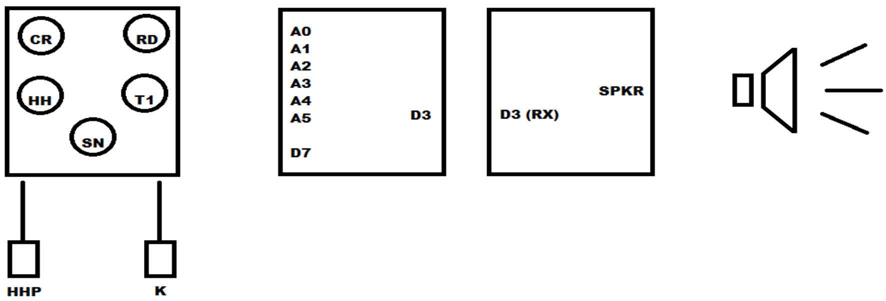
First I wanted Arduino (UNO) to synthetize the drum sounds from an algorithm I would store in its chip's memory, but then I realized that it just can store program codes. In addition, it just can output square waves with different frequencies, so I needed another support which, based on Arduino's instructions, could reproduce 'real' drum kit sounds. That way, Arduino was going to work as a MIDI controller; so I needed a support to work as MIDI synthesizer.

I searched for a shield that could do this task but I did not find anything. That's why I decided to use the PC. To make this work, MIDI messages had to be sent through the serial port of the Arduino. A virtual MIDI port had to be used because the used PC has no MIDI port. MIDI data was received through the USB COM port of the PC and sent to the virtual MIDI port with the Serial-MIDI



converter. Virtual MIDI port output was connected to the input of the virtual drum kit synthesizer software, which was going to reproduce the drum sounds.

But that didn't work either. Virtual MIDI drum machine software received MIDI data but did not reproduce any sound. After some tries I decided to look for any other method to make my project work properly. At that time I found *Sparkfun*'s *Musical Instrument Shield*, which was exactly the shield I was looking for.



I thought I had the work done when I found this shield, but there is a problem with the communication protocol between the Arduino and the shield, so for the time being the drum kit sounds just like a little keyboard square wave synth.

2.1.2. Design

The main objective during the design process was to make it simple, cheap and versatile, respecting the basic qualities of a standard drum kit.

At first, I wanted to build a completely standalone drum kit just using an Arduino *UNO* microcontroller, some piezoelectric sensors and a speaker with his amp. Initial design's number of pads was too big for the Arduino's number of analog inputs, so I reduced the number of pads to 6 (*A0* to *A5* pins).

I dismiss the idea of putting an amp and a speaker to the project because it could work perfectly with external speakers, so I reserved this option for future work.

At this point, my design seemed to fit the Arduino's characteristics, because I thought I could reproduce analog sounds stored in Arduino's chip memory. I started with some basic programs to learn how Arduino's inputs and outputs work.

Then I knew that Arduino was not able to perform as I thought and I had to include another support to carry out with my idea, and I finally opted for Musical Instrument Shield.

When I did the first sketch of my idea the drum 'brain' was placed inside a big box which also worked as a support for the pads. I discarded this option because the vibration produced by the hits in the pads could produce damage to brain's connections.

Another reason for which I discarded this idea was that I wanted to make an independent drum brain that gave option to choose the format and setting of the pads, so I put it in a relatively small closed box that allows plug and unplug the needed stuff for the drum kit performance.

To plug the cables and resistors between the Arduino and the pads and pedals I designed a small PCB but I decided to use a breadboard to simplify the work and allow plugging and unplugging pads easily.

2.1.3. Boards

Arduino UNO

Arduino Uno is a microcontroller board based on the ATmega328. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with an AC-to-DC adapter or battery to get started.
The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform.

Specifications:

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recom.)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328) (0.5 KB used by bootloader)
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Clock Speed	16 MHz

Input/Output:

Each of the 14 digital pins on the Uno can be used as an input or output, using pinMode(), digitalWrite(), and digitalRead() functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- Serial: 0 (RX) and 1 (TX): Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- External Interrupts: 2 and 3: These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the attachInterrupt() function for details.
- PWM: 3, 5, 6, 9, 10, and 11: Provide 8-bit PWM output with the analogWrite() function.
- SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK): These pins support SPI communication using the SPI library.
- LED: 13: There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and the analogReference() function.

There are a couple of other pins on the board:

- AREF: Reference voltage for the analog inputs. Used with analogReference().
- Reset: Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

Communication:

The Arduino Uno has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The '16U2 firmware uses the standard USB COM drivers, and no external driver is needed. Arduino's software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer.

A Software Serial library allows for serial communication on any of the Uno's digital pins.

Programming:

Arduino Uno can be programmed with the Arduino software (free download at Arduino's web page). The microcontroller on the board is programmed using the *Arduino programming language* (based on *Wiring*) and the Arduino development environment (based on *Processing*).

Schematic/Reference design:

B Annex

Musical Instrument Shield

Musical Instrument Shield is a microcontroller board which permits adding MIDI sound to Arduino projects. This board is built around the VS1053 MP3 and MIDI codec IC, wired in MIDI mode. To make it sound a speaker or headphones have to be connected to the 3.175 mm stereo jack on the shied and the proper serial commands had to be passed to the board.

The VS1053 contains two large tone banks including various piano, woodwinds, brass, synth, SFX and percussion sounds. The shield is also capable of playing several tones simultaneously (with a maximum of 31 sounds).

Input/Output:

Musical Instrument Shield has few pins.

- Serial: 2 (RX) and 3 (TX): Used to receive (RX) and transmit (TX) TTL serial data from Arduino.
- Reset: 4: Used to reset the board by software.
- Reset: RST: As MIS blocks Arduino’s built in reset button it has this reset button which can be tied to Arduino’s reset pin to reset the board.
- 1/8” Jack RCA breakout: To plug external mono speakers to the board.
- Mono speaker output: To attach a Jack breakout or directly a speaker.

Communication:

MIS only communicates with Arduino. It receives data through the RX pin and sends data through the TX pin. Depending on the information received, MIS will output different sounds through the speaker output.

Schematic/Reference design:

C Annex

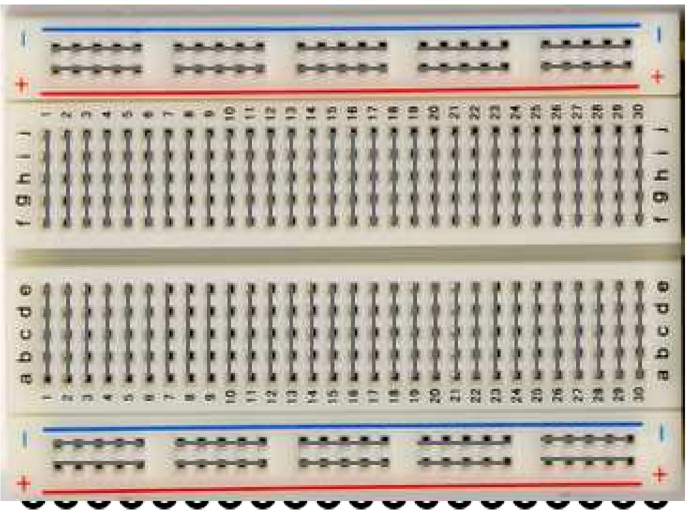
Breadboard

A breadboard (protoboard) is a construction base for prototyping of electronics. The term is commonly used to refer to solderless breadboard (plugboard).

It is reusable because does not require soldering. This makes it easy to use for creating temporary prototypes and experimenting with circuit design.

It consists of a perforated block of plastic with numerous conductor clips under the perforations. The clips are often called tie points or contact points. The number of tie points is often given in the specification of the breadboard.

The spacing between the clips is typically 2.54 mm. Where ICs are not used, discrete components and connecting wires may use any of the holes.



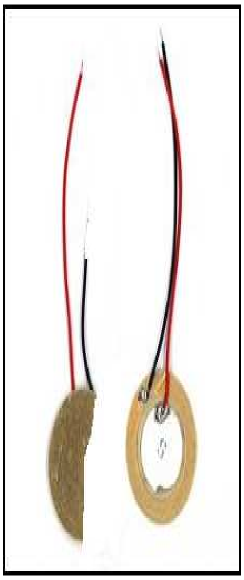
I chose this base for connections because it allows plugging and unplugging easily the pad cables and that way they can be replaced with no need of a solder.

You can see connections between pins of a medium size breadboard in the picture above.

Electronic components

lectric transducer

oelectric transducer is a two terminal device that uses piezoelectricity to measure a force by converting it into an electrical charge. They have very high DC output impedance and can be modeled as a proportional voltage source. The voltage V at the source is directly proportional to the applied force.



Based on piezoelectric technology various physical quantities can be measured; the most common are pressure and acceleration. For pressure sensors, a thin membrane and a massive base are used, ensuring that an applied pressure specifically loads the elements in one direction, to transfer the force to the elements. These kind of sensors can also be used to harvest otherwise wasted energy from mechanical vibrations by using piezoelectric materials to convert mechanical strain into usable electrical energy.

Resistor

A linear resistor is a linear, passive two-terminal electrical component designed to introduce a determinate electrical resistance between two points in a circuit. The current through a resistor is in direct proportion to the voltage across the resistor's terminals.

In irons, heaters, toasters and this kind of appliances use resistors to produce heat taking advantage of the Joule effect.

Variable resistors are called potentiometers or rheostats.

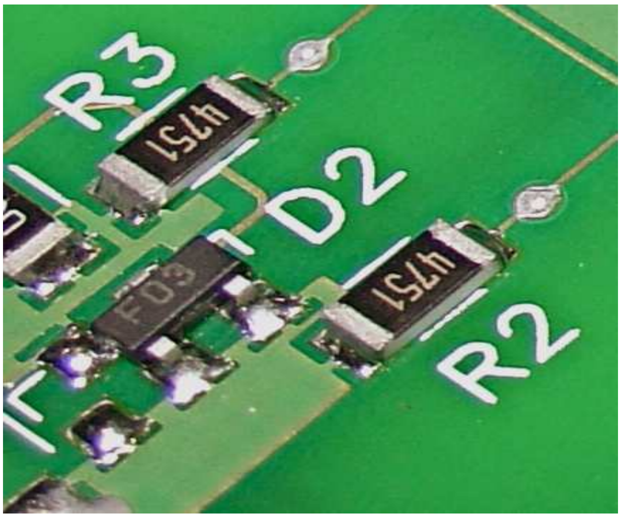
Resistors are made of carbon and other resistive elements to diminish the current that passes through a resistor. Maximum current in a resistor comes conditioned by the maximum power it can dissipate. The bigger is the resistor, the higher is its maximum dissipation. Resistor's most common power values are 0,25 W, 0,5 W and 1 W.

To distinguish a resistor three values are needed: electrical resistance, maximum dissipation and precision (tolerance). These values are indicated on resistor's body and depending on its type are going to be expressed differently.

In axial resistors, values are indicated with 4-6 color lines. To read these values tolerance line (usually silver or golden) has to be placed on the right. Having the resistor in this position we can know its value by reading the lines from left to right. First and second lines (and third in resistor with 5-6 lines) indicate a number from 0 to 9. First line is the first number of the value. Second line is second number. Third is multiplier line, which multiplies the number formed by the previous lines by 10^x.

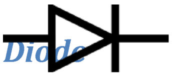
Band 1	Band 2	(Band 3) not common	Multiplier	Tolerance
-	-	-	$x10^{-2}$	±10%
-	-	-	$x10^{-1}$	±5%
0	0	0	$x10^{0}$	±1%
1	1	1	$x10^{1}$	±2%
2	2	2	$x10^{2}$	-
3	3	3	$x10^{3}$	-
4	4	4	$x10^{4}$	±4%
5	5	5	$x10^{5}$	±0.5%
6	6	6	$x10^{6}$	±0.25%
7	7	7	$x10^{7}$	±0.1%
8	8	8	-	±0.05%
9	9	9	-	-

Values in SMD (Surface Mounted Device) resistors are expressed with numbers placed in resistors body. The code is similar to that used in axial resistors but in this case, a three digit code is used. First digits represent first two meaningful



digits of the resistor’s value. Last digit is multiplier and expresses the number of zeros of the number multiplying the first digits.

In the picture above, R2 and R3 values are 4750 Ω.

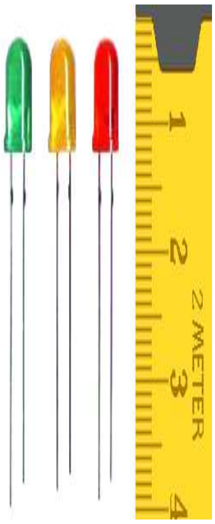


Diode

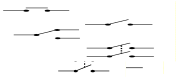
A diode is a type of two-terminal electronic component with nonlinear resistance and conductance (i.e., a nonlinear current–voltage characteristic), distinguishing it from components such as two-terminal linear resistors which obey Ohm's law. A semiconductor diode, the most common type today, is a crystalline piece of semiconductor material connected to two electrical terminals.

The most common function of a diode is to allow an electric current to pass in one direction (called the diode's forward direction), while blocking current in the opposite direction (the reverse direction).

Semiconductor diodes do not begin conducting electricity until a certain threshold voltage is present in the forward direction (a state in which the diode is said to be forward-biased). The voltage drop across a forward-biased diode varies only a little with the current, and is a function of temperature; this effect can be used as a temperature sensor or voltage reference.



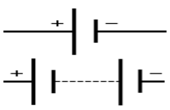
Semiconductor diodes' nonlinear current–voltage characteristic can be tailored by varying the semiconductor materials and introducing impurities into (doping) the materials. These are exploited in special purpose diodes that perform many different functions. For example, diodes are used to regulate voltage (Zener diodes), to protect circuits from high voltage surges (avalanche diodes), to electronically tune radio and TV receivers (varactor diodes), to generate radio frequency oscillations (tunnel diodes, Gunn diodes), and to produce light (*Light Emitting Diodes* aka LED).



Switch

A switch is an electrical component that can break an electrical circuit, interrupting the current or diverting it from one conductor to another.

Switches usually appear as a manually operated electromechanical device with one or more sets of electrical contacts, which are connected to external circuits. Each set of contacts can be in one of two states: either "closed" meaning the contacts are touching and electricity can flow between them, or "open", meaning the contacts are separated and the switch is non-conducting. The mechanism actuating the transition between these two states (open or closed) can be either a "toggle" (flip switch for continuous "on" or "off") or "momentary" (push-for "on" or push-for "off") type.



Battery

An electrical battery is one or more electrochemical cells that convert stored chemical energy into electrical energy. Since the invention of the first battery in

Jaime Ruiz Serra
INS Moianès

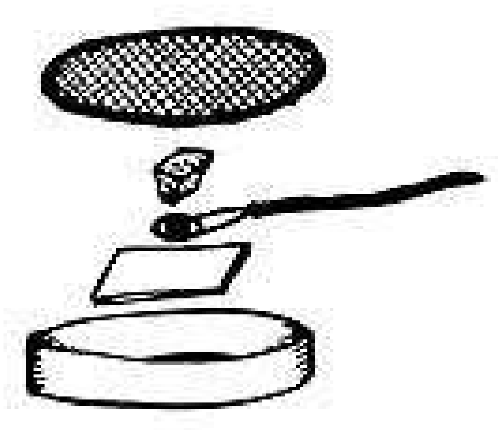
1800 by Alessandro Volta batteries have become a common power source for many household and industrial applications.

There are two types of batteries: primary batteries (disposable batteries), which are designed to be used once and discarded, and secondary batteries (rechargeable batteries), which are designed to be recharged and used multiple times. Batteries come in many sizes; from miniature cells used to power watches to battery banks the size of rooms that provide standby power for telephone exchanges and computer data centers.

2.1.5. Pads

The pads are the physical part of the electronic drum that receives the hits. It's important for their good implementation to have minimum rebound when hit with the sticks and a good absorption of the hits' vibrations.

For my drum kit, I was looking for not really big pads but cheap, comfortable, robust, and easily replaceable and moveable. The drum kit brain (i.e. controller, placed inside of the black box) can fit any kind of pad as long as they work with piezoelectric transducers. This allows the owner of the drum to assemble a full practice drum kit with real drum kit dimensions as well as a desktop drum kit or a portable drum kit.



For the prototype, I decided to use the desktop format. I designed different pads with ideas from other electronic drum kits I had seen. In the first one, the hit surface was a mosquito net. The piezo was paced inside a PVC tube and under a piece of foam in contact with the net. I thought it would work nicely with real drum kit dimensions pads, but I discarded it because it was difficult to tense the net and it would be fragile. Furthermore, it would be pretty expensive to build and hard to replace the piezo in case of damage.

Then I thought about my brother's electronic drum kit. Its pads were quite simple and I quickly thought a way to build similar pads easily. These pads had a thin rubber cover in the upper part covered with a tense plastic membrane. A small aluminum plate was placed in the center of the pad under the rubber cover. This plate communicated vibrations to the piezo. To avoid undesired vibrations the pad was settled over a wide piece of foam. All this components were inside of a wood case. This was my idea to simplify the pad:

I eventually decided to use CDs to harvest the vibrations of the hits and communicate them to the piezo. CDs are perfect for this task for both its size and its material characteristics. A 5 mm width rubber sheet was glued on the CD surface to give rebound to the pad. In cymbal pads foam was replaced by rubber blocks which would let the piezo vibrate after the hit.

Piezo sensor was glued to the CDs with a reusable putty-like pressure-sensitive adhesive (similar to Blu-Tack®). Pads with foam use Velcro® to hook the foam to the CD allowing future access to the piezo in case of damage or intention to change the drum kit format.

Pads can be hooked to a wood plank or simply disposed on a table. Their piezos are soldered to a resistor and each resistor is plugged to the breadboard so they can be easily replaced or removed.

Left pad is for cymbals and right pad is for drums.

For the portable version of the drum kit, piezos can be glued directly to a piece of rubber sheet of the same diameter of them for each and hooked to a piece of cardboard.
A funny version of the drum kit is available. Just attach the piezos with the sticky paste to any kind of stuff that can be hit and play a good grove with, for example, your mouse as a hi-hat, your desktop lamp as cymbal and your pencil case as snare drum; just let your imagination flow.

2.1.6. Pedals

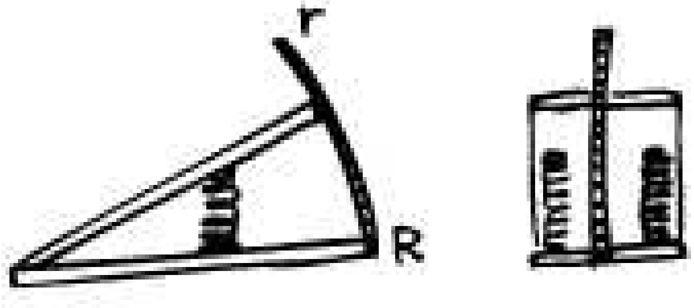
Pedals are designed to be robust, cheap, simple and functional. Pedal case is the same for both bass drum and hi-hat pedals. They are made of 1mm width steel plate with a piece of copper tube and a bolt to support the spring. Sensor is placed between the two plates of the pedal in both cases.
Pedals can be removed from the controller box even more easily than the pads. They are plugged to the front of the box with 6.35 mm jack.

Bass drum pedal

Bass drum pedal (or *kick pedal*) works pretty much as the pads. A piezo is placed inside the pedal on a piece of foam. A block of rubber is glued to the upper plate of the pedal on the inner surface. When the pedal is pressed, the rubber block hits the piezo. Thus, the current we were looking for is generated.

Hi-hat pedal

The initial idea for the hi-hat pedal was to use variable resistors to define how closed it was. To make it work that way I thought of three options:



- Coil potentiometer: The idea was based in a coil potentiometer placed vertically from the inner surface of the base of the pedal. Thereby, the more close to the base would be the upper part of the pedal, the more resistance to the current would offer the resistor, and that way the opening of the hi-hat cymbals could be defined.

- Pressure sensor: This idea used a pressure sensor placed in the hi-hat pedal in a similar way to the piezo in the bass drum pedal. In this case, the sensor would be on a hard surface and under a piece of soft rubber. With this configuration, the closer was the upper part of the pedal to the base; the more pressure would support the sensor.

- Photoresistor: In this design a light dependent resistor (LDR) was placed facing a LED in the inner base of the pedal. In the inner part of the upper plate of the

pedal would be a small plate placed vertically. This plate, when approaching to the base of the pedal, would be located between the LED and the LDR. That way, the closer to the base, the less light the LDR would receive and so the resistance offered to the current would increase.

Those ideas were great. The one I liked the most was the one using a pressure sensor for its low cost and hardiness. I had to discard those ideas because they would use analog inputs of the Arduino and as you know they were all already in use. I finally based the pedal in a simple microswitch. That way the hi-hat opening status could just be open or closed, but was enough for my intentions.

2.1.7. Power

The Arduino can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

- Vin: The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- 5V: The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.
- 3V3: A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- GND: Ground pins.

Musical Instrument shield is powered through the 5V output of the Arduino and its power characteristics were designed following Arduino's specifications, so it should work whenever it is properly connected to the Arduino.

2.2. Firmware

Firmware is a term often used to denote the fixed, usually rather small, programs and/or data structures that internally control various electronic devices. It is typically involved with very basic low-level operations without which a device would be completely non-functional.

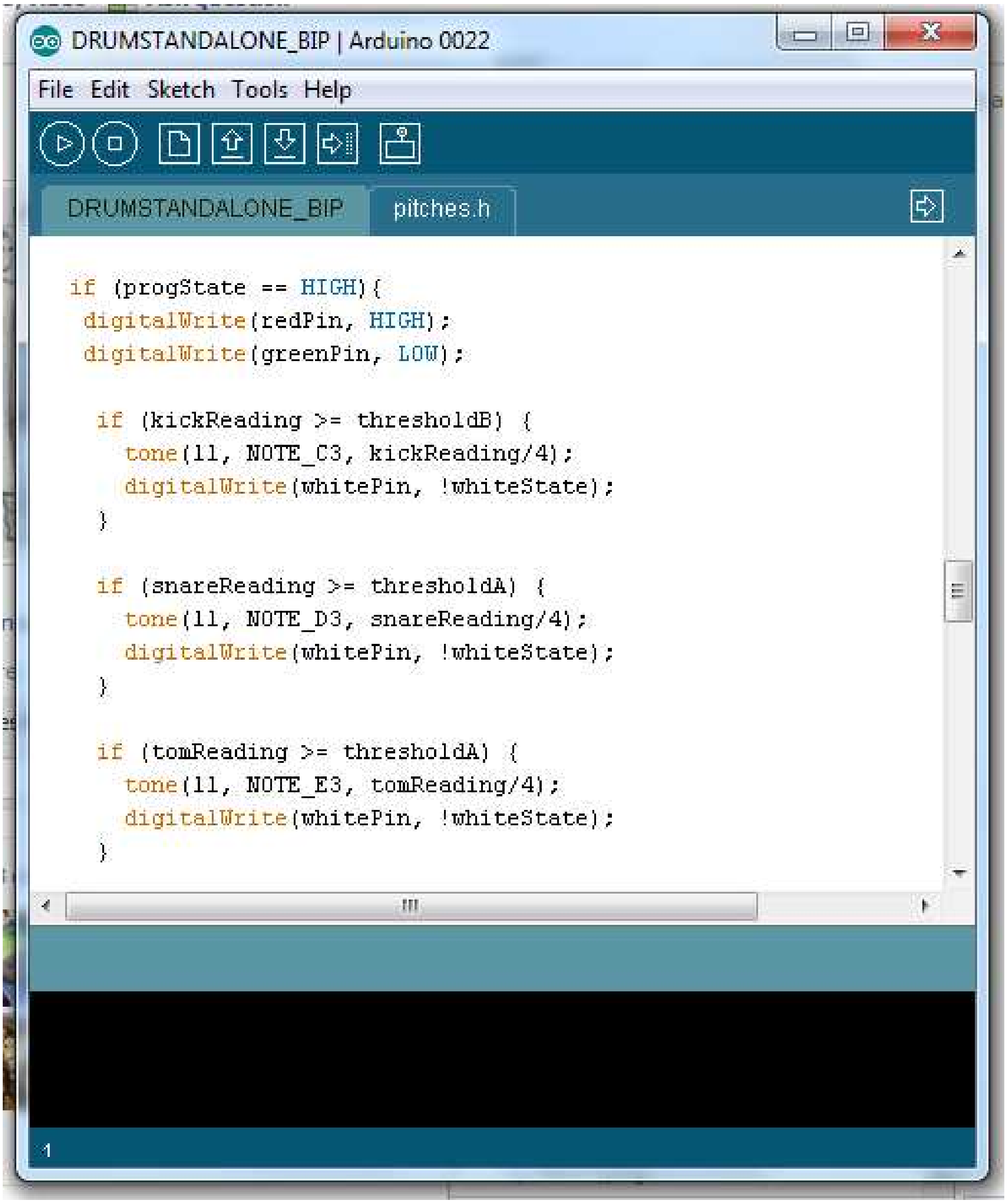
There are no strict boundaries between firmware and software, as both are quite loose descriptive terms. I defined as firmware the program codes stored in Arduino's chip which make different applications of the drum kit work.

This program codes are written in *Wiring* in a *Processing* environment and are loops of orders executed by the Arduino, which sends outputs (in our case, it should emit MIDI data through de digital pin 3) depending on the input data received (in our case, analog data from the piezos).

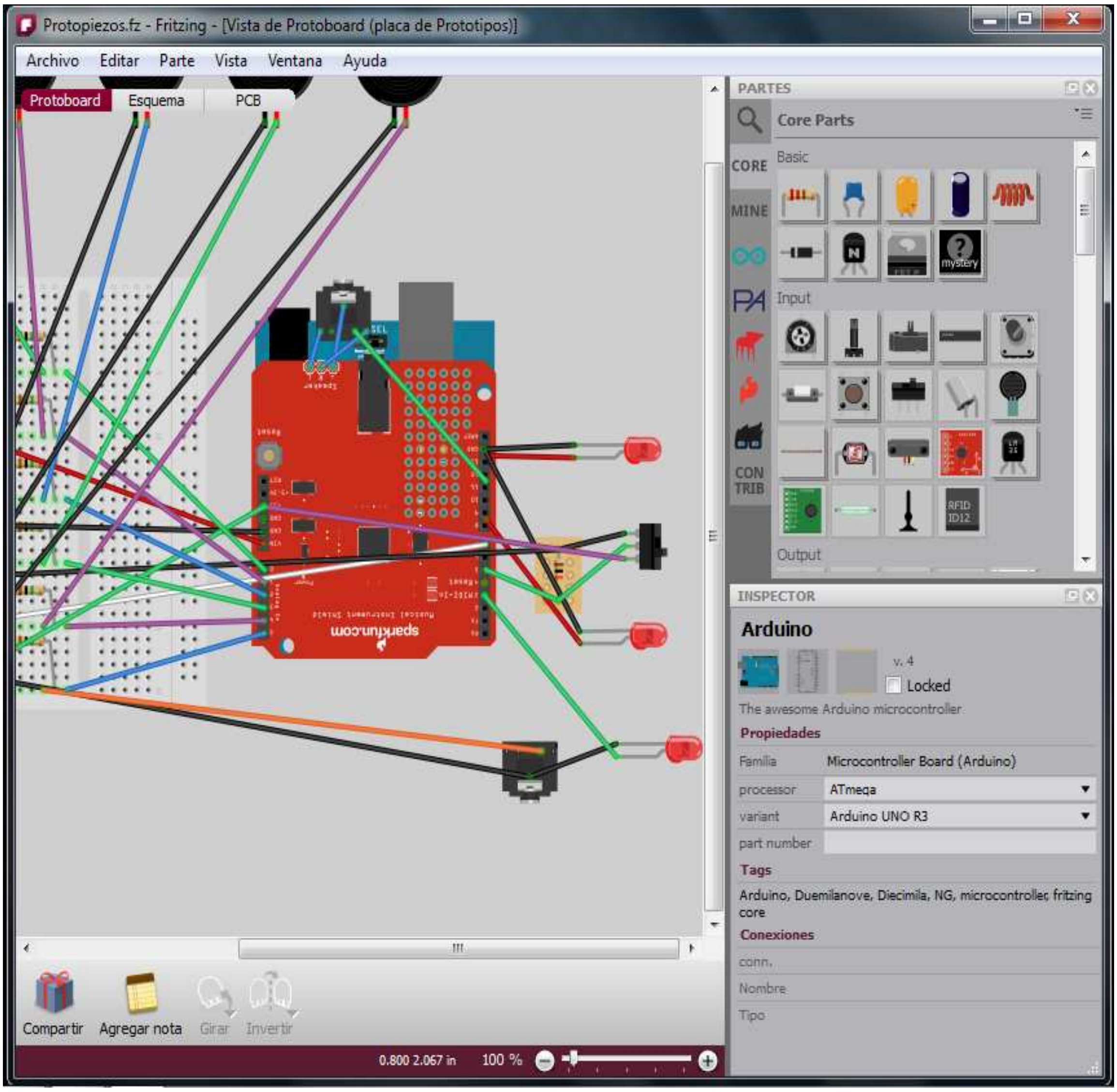
The program codes are stored in Arduino's chip.
There are some examples of the different program codes used in the *E Annex*.

2.3. Software

Software is the collection of computer programs and related data that provides the instructions for telling a computer what to do and how to do it. In other words, software is a conceptual entity which is a set of computer programs, procedures, and associated documentation concerned with the operation of a data processing system.
In this project different software has been used:



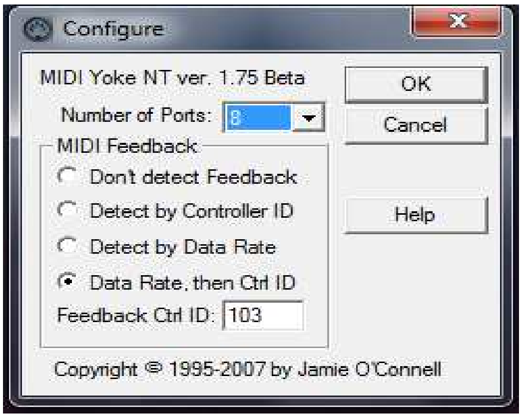
- <u>Programming environtment</u>: The most important software in this project. This software is used to write the program codes (Arduino's firmware) and set them to the Arduino. This software is essential to make the project work. It can be downloaded in Arduino's web page for free.



- Frintzing®: This program specialized in Arduino is used to design the electronic circuits of the project. It very practical because of its visual environment and easy handling. Have a look to the *E Annex* to know how they look like.

When I tried to work with the Arduino connected to the PC to generate sounds three software programs had to be used simultaneously:

- Serial to MIDI V2D: This program allows us to use a USB or Serial port as a MIDI port so the PC can receive MIDI data without having a MIDI port. MIDI data received through the USB port is send to a virtual MIDI port.



- MIDI Yoke NT: This software is a virtual MIDI port. It receives and sends MIDI data from or to other software. In our project, MIDI Yoke received data from V2D and sent it to BFD2.

- BFD2 (demo version): This software receives MIDI data from V2D's virtual port and converts it to drum sounds.

CONCLUSIONS

I think it was a very good idea to make this project in English. I have learnt lots of specific fields' vocabulary and ways to express my ideas that I had never used before.

About the different fields I have researched information I can't say I have learnt too much but the fact of making this research helped me clearing my ideas and understanding better how the things work in those fields.

The fact that the drum kit did not sound as expected taught me that in a design process things could not work and the designer has to adapt his work to the circumstances. In spite of this fact, I think I have achieved my objectives referring to the hardware. My drum kit has a fine looking and robust case, as well as portable and functional. Pads have a good response to the hits and pedals work pretty well.

In a near future I would like to make the drum kit sound as I thought at the beginning of this project anyway.

One of the things I like the most of this project is piezoelectricity phenomenon. I would really like to find a way to get energy from a device using this principle.

Summarizing, this research project has been a great experience which has helped me to have an idea of how the things work in a product design process. I have noticed that the world of design is very interesting and I decided to study something related to this field.

REFERENCES

VARIOUS AUTHORS; *Wikipedia, the free encyclopedia* [online] <en.wikipedia.org/wiki/Main_Page>

VARIOUS AUTHORS; *Arduino Team webpage.* [online] <arduino.cc/en>

VINCENT, James; *Lesson 49: Properties of sound.* [online] <studyphysics.ca/newnotes/20/unit03_mechanicalwaves/chp141516_waves/lesson49.htm>

ALMIRALL, Oriol; *La música: Història de la bateria.* [online] <gaudeixamblamusica.blogspot.com/2008/11/historia-de-la-bateria.html>

VARIOUS AUTHORS; *Musical Instrument Shield – SparkFun.* [online] <sparkfun.com/products/10529>

VARIOUS AUTHORS; *Arduino shield list.* [online] <shieldlist.org/sparkfun/music>

EVANS, Brian W; *Arduino programming notebook.* [online] <lulu.com/items/volume_63/1108000/1108699/4/print/ARDUINO_NOTEBOOKv6.pdf>

ANNEXES

(A- Examples of drum sets)

(B- Arduino UNO - schematic)

(C- Musical Instrument Shield - schematic)

(D- Designs, schematics, programs, applications)

(E- Materials and assembly)