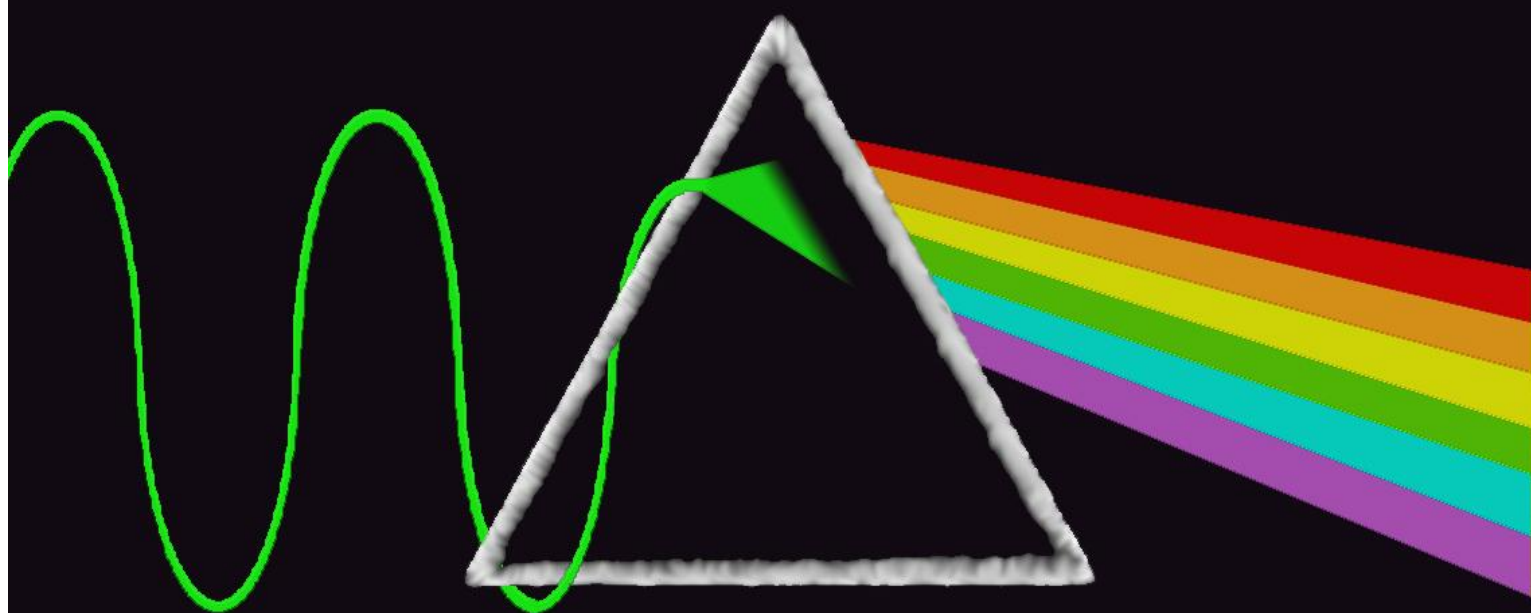


**Maristes Valldemia 2014**

**Tutor: Albert Dotras**

# Veient la música

**Representació visual de la música, amb llums i moviment  
mitjançant la robòtica**



**Marc Sala Hidalgo**

**2n BTL A**



Abans de començar el treball, m'agradaria donar les gràcies a totes les persones que m'han ajudat i donat suport.

Agraeixo especialment l'ajuda i el suport rebut pel meu pare, Lluís Sala, i la meva cosina, Arantxa Entonado, especialment a la part final d'aquest treball així com tot l'acompanyament durant la creació d'aquest. També m'agradaria donar un agraïment especial al doctor Andreu Veà tant pels seus comentaris encoratjadors com pels contactes proporcionats; a l'Eloi Vilalta i l'Albert Dotras, tant pels coneixements transmesos, com per l'interès i acompanyament durant el treball.

Crec oportú reconèixer i agrair el temps dedicat en ajudar-me en el meu treball a les persones entrevistades i visitades: David Cuartielles, Nacho Sánchez, Neil Harbisson, a l'equip del doctor Miquel Badia, David Bassetti, Miquel Vidal i Albert Mercadé.

Moltes gràcies a tots.



“Los mayores desafíos engendran los mejores resultados”

Guy Kawasaki



# ÍNDEX DE CONTINGUTS

---

<b>1. INTRODUCCIÓ</b>	<b>1</b>
<b>2. PRESENTACIÓ</b>	<b>8</b>
<b>3. TEORIA</b>	<b>9</b>
<b>3.1. PARÀMETRES DEL SO</b>	<b>9</b>
3.1.1. Intensitat	11
3.1.2. Ritme	13
3.1.3. Altura	14
<b>3.2. LED</b>	<b>15</b>
<b>3.3. ARDUINO</b>	<b>16</b>
<b>3.4. EMMASCARAMENT VISUAL TEMPORAL</b>	<b>21</b>
<b>3.5. PROJECTES AFINS</b>	<b>24</b>
3.4.1. EYEBORG	24
3.4.2. SPLYCE APP	27
<b>4. PRACTICA</b>	<b>31</b>
<b>4.1. COMPOSICIÓ DEL ROBOT</b>	<b>31</b>
4.1.1. Plaques	31
4.1.2. Estructura	36
4.1.3. Il·luminació	40
4.1.4. Font d'energia	44
4.1.5. Font de so	47



<b>4.2. PROGRAMES D'ANÀLISI</b>	<b>50</b>
4.2.1. Intensitat	50
4.2.2. Ritme	53
4.2.3. Altura	56
<b>4.3. ACOBLAMENT DE PROGRAMES</b>	<b>62</b>
<b>4.4. MUNTATGE I EVOLUCIÓ DEL ROBOT</b>	<b>64</b>
<b>5. CONCUSIONS</b>	<b>69</b>
<b>6. BIBLIOGRAFIA</b>	<b>72</b>
<b>7. ANNEXOS</b>	<b>74</b>
<b>7.1. ANNEX 1 (Programes finals)</b>	<b>74</b>
7.1.1. W_RHYTHM	74
7.1.2. W_INTENSITY	75
7.1.3. W_PITCH	76
<b>7.2. ANNEX 2 (Versions anteriors)</b>	<b>78</b>
7.2.1. W_LECTOR	78
7.2.2. W_VUMETRE	78
7.2.3. W_LED	79
7.2.4. W_FADE	79
7.2.5. W_ARRAY	80



# 1. INTRODUCCIÓ

---

Des de ben petits ens han ensenyat que la música s'escolta, les pintures es miren i la llum es veu però és possible representar la música visualment? Potser amb llum? Potser amb pintura? Amb moviment? Jo crec que sí i aquest és l'objectiu del meu treball: construir un robot que representi una cançó mitjançant la combinació de llum i moviment.

Durant les vacances de Nadal del 2012, vaig fer un curs, al Tecnocampus de Mataró, d'iniciació a Arduino que és una plataforma d'electrònica que s'utilitza per a la creació de prototips interactius. Abans d'aquest curs, ja havia programat abans però no un robot ni amb aquesta plataforma. El curs em va descobrir un nou sistema que m'agradava i relacionava l'electrònica amb la programació: dues tecnologies que m'agradaven per separat així que ajuntar-les va ser la combinació perfecta!

En aquella època també estava buscant un tema per al treball de recerca i sabia que volia fer quelcom tecnològic on em sentís còmode i em pogués esplaïar però no sabia exactament què fer.

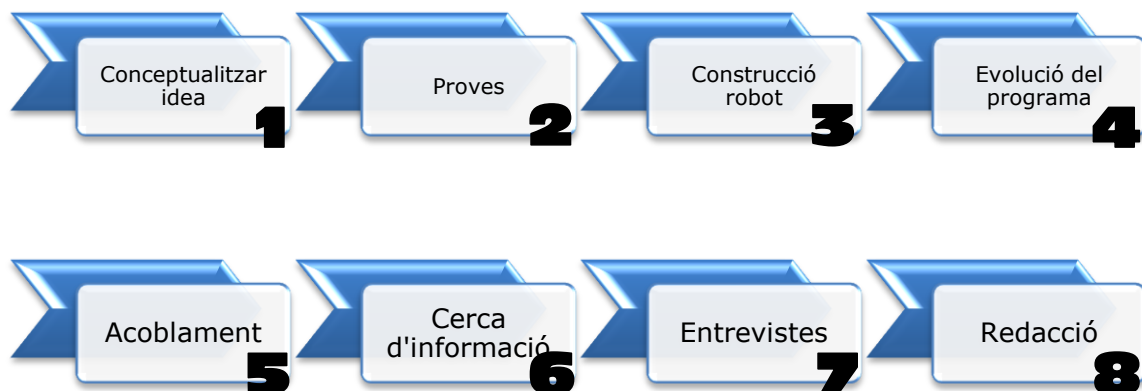
La idea inicial va sorgir juntament amb el meu pare, mentre parlàvem sobre el curs que estava fent, va relacionar la plataforma Arduino amb una altra gran hobby, i possible pla futur d'estudis que tinc: la música. Parlant sobre com relacionar aquests dos àmbits van



començar a sorgir idees fins a arribar a una possibilitat que ha anat evolucionant fins al robot actual.

La idea del treball complia tots els meus objectius: relacionava mons que m'agradessin, no em cansaria de fer-ho perquè m'interessava el tema, era principalment pràctic on podia aportar bastants coneixements propis i estava relacionat amb qualsevol dels possibles àmbits dels meus estudis futurs.

L'esquema de treball plantejat va ser aquest:



- 1. Idea + Conceptualització:** Tria del tema i Naixement del robot com idea.
- 2. Proves tecnològiques:** Primeres preses de contacte amb els materials a utilitzar i proves de funcionament de petites parts.
- 3. Construcció robot:** Acoblament de les peces prototips i de mica en mica les definitives.



- 4. Creació i Evolució dels programes:** desenvolupament de les tres aplicacions que el robot executaria; W\_Rhythm , W\_Intensity i W\_Pitch.
- 5. Projecte d'acoblament dels programes:** Intent frustrat d'acoblar els tres programes en un sol de sol.
- 6. Cerca d'informació:** Internet, biblioteca, preguntant a experts, converses amb tutor.
- 7. Entrevistes:** Preparació, programació i realització de les entrevistes amb els experts, utilitzant el mitjà idoni en cada moment.
- 8. Redacció del treball:** Resumir, extreure i escriure el document del treball.

### Organització del treball

Una part teòrica en la qual explico:

- Paràmetres del so i de la llum: relació de tres aspectes analitzats: el volum (mesurat en dB<sup>1</sup>), l'altura (mesurada en Hz<sup>2</sup>) i el ritme (mesurat en BPM<sup>3</sup>) i combinar-ho amb tres característiques del moviment i la llum: la intensitat de la llum d'una tira de díodes led, el color d'aquests i el moviment de tot el conjunt del robot.

---

<sup>1</sup> Db: Decibels.

<sup>2</sup> Hz: Hertz.

<sup>3</sup> Bpm: "beats per minute", batecs per minut.





- Explicació dels paràmetres bàsics de la música i el so, per tal d'entendre el funcionament de l'autòmat i com interpreto cadascun d'aquests paràmetres indispensables per fer qualsevol tipus de música.
- El sistema Arduino: Què és? Perquè té tant èxit? Història sobre la creació de la plataforma i explicació del software i hardware utilitzats al robot. Molta informació d'aquest apartat és extreta de l'entrevista amb el cofundador de la plataforma: David Cuartielles.
- L' "emascarament visual temporal" : Que és un fenomen òptic que quan vaig començar a fer el treball sabia com fer-lo funcionar però no sabia perquè passava ni com es deia. He cregut important posar-lo dins del treball, ja que és molt utilitzat pel robot alhora de la representació. Per obtenir informació d'aquest fenomen òptic em vaig posar-me en contacte amb un oftalmòleg molt conegut de Mataró, el Dr. Badia i ell em va posar en contacte amb altres persones del seu equip que em van ajudar.
- Finalment, un parell de projectes afins al meu. Vaig estar buscant per Internet altres projectes que hagués fet algú i que s'assemblessin al meu però al contrari del que jo pensava, no hi havia pràcticament res que s'hi assemblés. El primer que vaig trobar va ser un robot que feia un dibuix amb un bolígraf a través del ritme de la música i vaig contactar per e-mail amb el seu creador, l'holandès Tom Fejer (amb el que havia de parlar



en anglès). Després d'intercanviar alguns correus i demanar informació sobre el que havia fet em va dir que no em podia dir res i va deixar de contestar els missatges.

Més tard vaig incorporar al treball l'aplicació Slyce App; m'ho va recomanar el Dr. Andreu Veà perquè creia que em podia interessar, ja que feia moltes coses iguals que la idea del meu robot. Aquesta vegada sí que vaig poder parlar amb un dels seus creadors, Nacho Sánchez, amb el qual concertarem una entrevista realment fructífera.

I per últim l' "Eyeborg", un projecte amb molt de ressò i el més semblant al meu i que ja el coneixia abans. El creador és el Neil Harbisson i he aconseguit contactar amb ell via e-mail i m'ha explicat coses sobre el seu projecte.

I una segona part pràctica on s'expliquen detalladament els programes que he creat per l'anàlisi dels paràmetres del so, així com el muntatge i la construcció física del robot. L'explicació dels programes està conformada per una petita explicació del que fa i de com ha anat evolucionant pas a pas fins a arribar al programa final.

Per a explicar la construcció del robot he anat fent un breu resum del què fa cada element, les modificacions que li he fet perquè s'adeqüi a les necessitats del projecte i finalment explico com l'he muntat i la solució als problemes d'espai. Tot aquest bloc de pràctica està complementat amb fotografies i versions anteriors dels programes per tal de poder-ho entendre tot amb facilitat.



Com aquest treball és bàsicament pràctic el primer que vaig fer va ser la part pràctica, hi vaig dedicar moltes hores, però va ser divertit perquè realment ho passava molt bé. Quan ja tenia el robot gairebé acabat, em vaig posar a fer entrevistes tot ensenyant-lo a les persones amb les quals "em trobava". Vaig decidir fer-ho d'aquesta manera perquè així aquestes persones em podrien donar la seva opinió sobre el robot i el podria anar millorant.

Les "trobades" amb aquestes persones han estat una experiència interessant i peculiar, per exemple: amb el Neil Harbisson vaig contactar per e-mail però no va respondre ell personalment sinó que ho va fer la seva "communications manager". Amb el cofundador d'Arduino, el David Cuartielles em vaig entrevistar a través d'una videoconferència (perquè ell viu a Malmö, Suècia) el dia de Sant Esteve, gairebé al mig del dinar amb la família perquè ja havia quedat amb ell dues vegades i no s'havia pogut connectar a l'últim moment. També he tingut una entrevista personal, al centre de co-working MOB de Barcelona, amb el britànic David Bassetti (només parla anglès) i a l'empresa InqBarna, amb Nacho Sánchez.

Estic content perquè crec que he aconseguit transmetre la idea del meu robot i fer que l'entenguessin. El que m'he trobat són persones molt enteses i reconegudes en alguns d'aquests mons i a totes els ha causat entusiasme veure algunes de les representacions del robot.

Quan ja tenia totes les entrevistes i el robot acabats, vaig començar a plasmar-ho tot per escrit i em va suposar un gran avantatge haver



anat guardant un gran volum de dades i de programes organitzats curosament.

La major part dels contactes amb els quals he intercanviat coneixements i he ensenyat l'autòmat, m'hi he posat en contacte gràcies a la intermediació del meu pare, Lluís Sala, i del Doctor Andreu Veà. L'ajuda, suport i contactes proporcionats per ells han sigut clau per al treball.

Per explicar el funcionament del robot i el tema del treball a les diferents persones interessades en el treball, a robot l'hi vaig posar el nom de "White" ja que era massa tediós haver de dir sempre "el robot del treball de recerca". Crec que és un nom curt, concís i fàcil de recordar. "White" vol dir blanc en anglès, tot i que el robot és negre, el nom prové del grup de música "White Stripes", un dels meus grups preferits i que he utilitzat molt per aquest treball perquè tenen ritmes de bateria marcats on és fàcil poder extreure'n la base.

Com el robot ja tenia nom, vaig aprofitar per incorporar-hi aquest nom als programes, tots ells comencen amb la lletra "W" de White, i segueixen amb una paraula o dues que descriuen breument què tracta el programa, com per exemple "W\_Rhythm".

Us convido a gaudir d'aquest treball tal com jo ho he fet desenvolupant-lo i m'agradaria que us poguéssiu fer una idea de com és el robot, com és la representació que fa i les seves possibles aplicacions perquè, com va dir David Cuartielles a la videoconferència: "La idea debe ser más grande que el prototipo".



## 2. PRESENTACIÓ

---

Com s'ha explicat anteriorment, l'objectiu d'aquest treball és aconseguir fer un autòmat capaç de representar visualment la música, gràcies a la utilització de dues característiques de la llum (el color i la intensitat) i el moviment de les rodes del robot.

Aquest robot, i el treball en general, pretén ser una eina per ser utilitzada en altres àmbits que s'escapen del meu objectiu com a treball tecnològic.



## 3. TEORIA

### 3.1. PARÀMETRES DEL SO

Aquest apartat té el nom de "paràmetres del so" ja que són els paràmetres analitzats per l'autòmat i rebuts a través de la font de so del sistema robòtic, tot i que són una mescla de les qualitats del so i dels paràmetres de la música.

Cal destacar la diferència entre so i música; segons l'institut d'estudis catalans, el so, és una impressió produïda en l'òrgan de l'oïda per les vibracions elàstiques d'un cos que es propaguen en tots els medis materials en forma d'ones, mentre que la música és una combinació de sons d'acord amb les lleis de la melodia, harmonia i ritme.

El so és una ona mecànica longitudinal que es propaga a través medis elàstics (aire, aigua...), per entendre les qualitats del so i els paràmetres de la música és essencial entendre els següents conceptes sobre les ones sonores:

- Freqüència: És el nombre d'ones que fa el so en un segon, es mesura en hertz (Hz).
- Període: És el temps que tarda el so a fer una oscil·lació, mesurat en segons (s).
- Amplitud: L'amplitud d'una ona indica el nivell de potència amb què es produeix les oscil·lacions. Això és captat en forma de volum o intensitat sonora, és mesurat en Bels (B) tot i que és més comú utilitzar el seu múltiple decibel (dB).



Com es pot apreciar, la música i el so es regeixen per una sèrie de lleis o paràmetres diferents entre ells, tot i que estan molt relacionats entre si. Les qualitats del so són:

- Altura: També anomenat to, és la qualitat del so que permet distingir entre les diferents freqüències de l'espectre d'àudio. Està relacionat amb la gravetat o agudesesa del so, i com té relació directa amb la freqüència del so també es mesura en Hertz (Hz).
- Duració: Persistència de l'ona de so en el temps. És mesurat en segons (s).
- Intensitat: Relacionat amb l'amplitud de l'ona de so, és la potència acústica transferida per l'ona de so. Es mesura en Bels (B).

Els paràmetres de la música, ja esmentats abans, són:

- Melodia: Combinació successiva de sons.
- Harmonia: Combinació simultània de sons.
- Ritme: Velocitat a la qual es reproduïx la música, determinada pel temps que hi ha entre cada so que compon una melodia.

Per fer la representació visual d'una cançó concreta no podia utilitzar tots aquests paràmetres, així que vaig analitzar i representar les que crec que són més significatives dins d'una cançó.



### 3.1.1. Intensitat

---

La intensitat o volum d'un so és la característica que marca si l'amplitud d'ona és més gran o més petita, això ho percebem en un so sentint-lo més fort o més fluix.

La sensació que soni amb una determinada intensitat ve determinada, com he dit, per l'amplitud de l'ona sonora, com més gran és aquesta amplitud, més energia desplaça i, alhora de rebre-ho al timpà, aquest vibra més i fa una sensació de més sonoritat.

La intensitat sonora es mesura en Bels, tot i que aquesta és una simplificació de la mesura real  $W/m^2$ . Com que el so és una ona, conté energia, la mesura d'aquesta ve determinada per la quantitat d'energia de la font emissora ( $W$ ) dividit entre la superfície receptora ( $m^2$ ) en el temps d'un segon.

En transformar aquesta energia sonora en un corrent elèctric, el voltatge d'aquest varia depenent de la intensitat del so. Aplicat en un altaveu, fa el mateix efecte que en el timpà de l'oïda amb les ones sonores transmeses per l'aire: les variacions del corrent elèctric fan vibrar en major o menor mesura la membrana de l'altaveu, creant així més o menys desplaçament d'energia a l'aire.

L'oïda humana pot captar intensitats sonores compreses entre els 0dB i els 140dB, per sota d'aquestes és imperceptible per les qualitats del timpà (llindar d'audició) i per sobre crea una sensació de dolor, i fins i tot pot causar el trencament del timpà (llindar de dolor).





La intensitat sonora de les melodies analitzades són captades pel robot de la següent manera: com la font de so transmet les dades de la música en forma d'electricitat, l'autòmat detecta el voltatge que li arriba pel connector perquè, com ja hem dit, el voltatge varia depenent de la intensitat sonora.

Fruit de tot això s'obté un valor que després és utilitzat per representar-ho visualment.



### 3.1.2. Ritme

---

El ritme d'una cançó és la característica que marca la velocitat que es reproduïx o s'interpreta aquesta. Cada cançó té un ritme propi que és determinat per l'autor de la cançó al principi d'aquesta.

El ritme acostuma a ser constant durant tota la cançó, tot i que hi ha alguna excepció. És mesura en "beats per minute" (BPM), que vol dir les pulsacions que fan els instruments en la cançó a cada minut. Una cançó està dividida en compassos, depenent de les característiques d'aquest compàs incorpora un determinat nombre de pulsacions (els "beats").

El robot és capaç de representar aquest ritme, tot i que no és capaç de captar-lo per si mateix, ja que el programa utilitzat seria massa complex pel maquinari del qual dispo. Incloent el valor del ritme (en "BPM") al programa de representació d'aquest, el robot fa una sèrie de càlculs per tal d'adaptar aquest valor a les seves necessitats per la representació.



### 3.1.3. Altura

---

L'altura d'un so és la característica que marca si la longitud d'ona és més o menys extensa, això es percep auditivament amb l'agudesesa o gravetat d'un so.

Aquest paràmetre està relacionat amb la freqüència, mesurada en Hertz(Hz), defineix el nombre d'oscil·lacions d'ona acústica per segon.

Com explica l'efecte Fourier, els sons estan formats per la consecució de freqüències. És a dir, qualsevol so que hi ha a la natura, vida diària o reproduït per dispositius no té una sola freqüència, sinó que n'hi ha moltes barrejades per tal de crear un so, per simple que sigui.

Per limitacions anatòmiques, l'oïda humana no pot captar menys de 20Hz ni més de 20.000Hz (aproximadament). Per sota d'aquesta freqüència de 20Hz el so és massa agut per sentir-lo i se l'anomena infrasò, per el contrari, un so superior a 20.000Hz s'anomena ultrasò, ja que tampoc es pot captar. Aquestes freqüències imperceptibles per nosaltres, altres animals si que les poden captar i, fins i tot els humans, les utilitzem en diferents àmbits sense sentir-les.

Aquesta característica de les ones sonores no és captada en essència pel robot, per impossibilitats de càlcul del sistema utilitzat. Però es representa utilitzant les notes que toca un dels instruments de la cançó analitzada, ja que una nota musical no deixa de ser un conjunt de freqüències diferents.



## 3.2. LED

---

Un díode led ("Light Emitting Diode") és un dispositiu electrònic capaç de produir llum quan es polaritza correctament amb corrent elèctric.

Com tot díode, la seva característica principal és que només deixa passar el corrent en un sentit, però en aquest cas ho fa alhora que emet llum.

Hi ha leds de molts colors, des d'infraroig fins ultraviolat, passant per tot l'espectre visible, i el color d'aquest depèn del material que contenen a l'interior.

El led, com qualsevol díode té dues potes: ànode i càtode, que és per on entra i per on surt l'electricitat respectivament, provocant una davallada de tensió de 0,7 volts.

El voltatge d'alimentació d'un led pot anar des de 1,5V fins a 5V i la intensitat oscil·la entre 10 i 20 mA.

Un led RGB, és un tipus de led que pot representar tota la gamma de colors del espectre visible gràcies a la combinació dels colors primaris de la llum: blau, vermell i verd. Tot i que és un sol encapsulat, per dins hi porta tres leds d'aquests colors primaris.

El breu temps de reacció del led, sumat al baix consum i la seva versatilitat d'ús en general, m'ha facilitat molt la representació visual.



### 3.3. ARDUINO

---

Arduino és una plataforma electrònica que proporciona tant un programari com un maquinari ideat per ells, amb un llenguatge de programació propi. Tot el software i hardware és obert al públic és a dir, que pots trobar fàcilment l'esquema per a fer les plaques o el codi font del seu programa a Internet. Això és el més destacat i que està fent que Arduino aconsegueixi

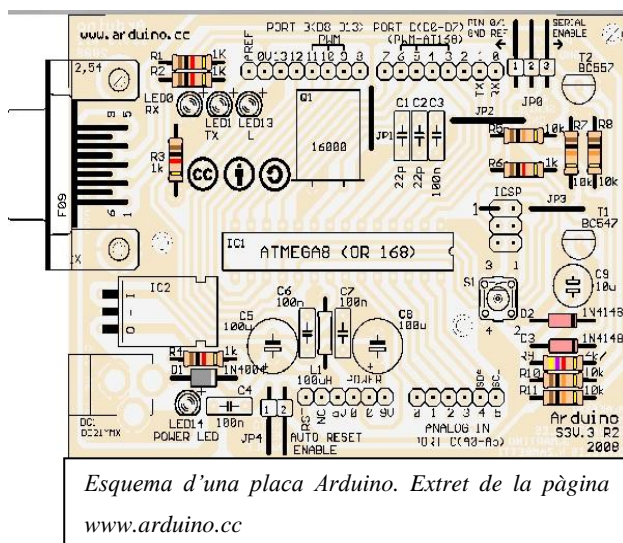


un impacte cada cop més gran en la programació i desenvolupament de prototips, ja que permet que molta gent pugui crear els projectes de manera senzilla i assequible.

La creació d'aquesta plataforma va ser al 2005 a l'institut de disseny IVREA (Itàlia), necessitaven crear una eina que poguessin utilitzar tots els alumnes, ja que les plaques usades fins al moment eren massa cares (uns 100 dòlars) i provocaven queixes. Així va ser com Massimo Banzi i David Cuartielles, dos professors de l'institut, van començar a buscar una solució per a aquest problema: cadascun d'ells va crear un sistema diferent i es van posar d'acord per a decidir quin dels dos sistemes era més convenient. Finalment van decidir que s'utilitzaria el sistema de Massimo, ja que era compatible amb molts sistemes operatius i proporcionava una major llibertat. Un cop decidit el sistema, David Cuartielles va revisar aquest hardware i va arreglar un parell de problemes que tenia.



En aquest procés s’hi va anar afegint gent, Hernando Barragán (estudiant de l’institut IVREA) va començar a fer el llenguatge de programació (basat el Wiring) i Casey Reas amb l’ajuda d’Hernando van crear l’entorn de desenvolupament Arduino (basat en Processing). Més tard s’hi va incorporar David Mellis, que va ajudar a escriure el software fent compatible la placa Arduino amb programes escrits en altres llenguatges de programació. També s’hi va incorporar Tom Igoe, fent de supervisor de tot el projecte i Gianluca Martino, qui va fer un parell de canvis a la placa Arduino per tal de poder-la manufacturar i començar a produir-les.



Esquema d’una placa Arduino. Extret de la pàgina [www.arduino.cc](http://www.arduino.cc)

Arduino, com ja he dit, és una plataforma oberta al públic de hardware i software lliure al mateix temps. Que el software sigui obert, “open-software”, vol dir que s’ha creat un programari (en aquest cas el programa Arduino amb el que es programen les plaques) del

qual desvetllen el codi font gratuïtament. El hardware també és lliure, “open-hardware”, això vol dir que els mapes i circuits amb els qual es fan les plaques estan disponibles a Internet de forma gratuïta. Tot i que les plaques es poden comprar, a la mateixa pàgina web d’Arduino s’hi pot trobar les instruccions de com fer-ne una tu mateix.



Les dues raons per la que la plataforma és oberta, segons va explicar David Cuartielles a l'entrevista són:

- Per ideals, les persones que van fundar Arduino i que hi formen part, defensen la llibertat d'informació i veuen positiu el fet de compartir aquestes dades amb els usuaris dels seus sistemes.
- La facilitat de trobar la informació ho fa més accessible i més barat per a tothom, a més, a la mateixa plataforma interessa aquesta facilitat, ja que la gent pot fer les seves modificacions a les plaques base i això enriqueix el sistema.



*Equip fundador d'Arduino. Extret de la pàgina <http://solaria-arduino-esp.blogspot.com.es>*



Amb el software d'Arduino és amb el que es programen els microcontroladors de les plaques base. Aquest microcontrolador és un xip que, gràcies al que s'ha programat i se li guarda a la memòria, controla els diferents components de la placa per tal que aquesta executi les instruccions de l'esmentat programa.

La plataforma Arduino també es caracteritza per ser molt versàtil: es pot programar en més de 30 llenguatges de programació diferents. Tot i això el més comú és el llenguatge C o C++.

El programa té un entorn de desenvolupament basat en "Processing" que és simultàniament llenguatge de programació i entorn de desenvolupament. A més, aquest programa, està basat en "Wiring" que és un marc de programació per a microcontroladors. Tant el "Wiring" com el "Processing" són dos sistemes de programació open-source, igual que Arduino.

El hardware Arduino és el conjunt de plaques que formen la part electrònica del robot i aquest maquinari està classificat en dos tipus:

- "Boards": Són plaques electròniques programables i que es connecten a un ordinador per fer-ho, hi ha més de 20 models i cadascun presenta unes característiques diferents depenent del seu objectiu. Aquestes plaques se li acoblen altres elements o plaques per tal d'ampliar les seves funcions a través d'uns ports anomenats pin.
- "Shields": Són plaques electròniques d'expansió que fan alguna funció específica dins d'un sistema. Segons la pàgina oficial de la plataforma hi ha 7 models, que són els que han creat ells,





però hi ha multitud de "shields" diferents creats per empreses o particulars i que serveixen per a les "boards" Arduino, els preus de les mateixes són realment assequibles i de molt fàcil localització i compra per Internet, a part d'estar força ben documentades.

Igual que la majoria de plataformes de programació, l'entorn d'Arduino pot ser ampliat mitjançant "libraries" (biblioteques). Aquestes biblioteques, són necessàries per programar determinats "shields".



Una biblioteca és un arxiu que proporciona unes funcions que no existeixen en el programari inicial, en el cas d'Arduino una biblioteca serveix per simplificar l'ús de determinats "shields" creant noves funcions que interactuen d'una determinada forma.

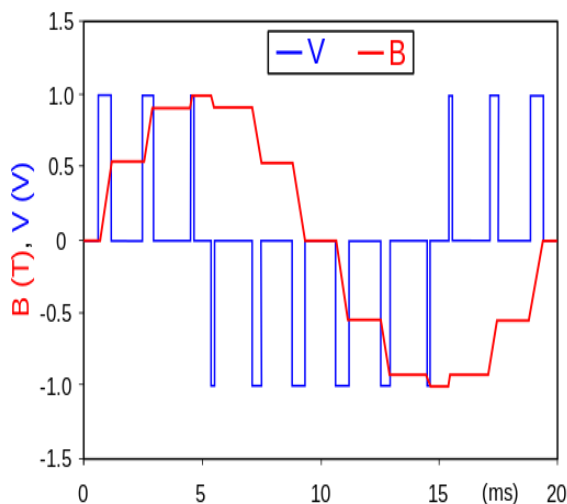
Més endavant veurem un exemple d'utilització d'una "library" per simplificar el funcionament d'una placa d'expansió.

El mateix programari d'Arduino té algunes llibreries de sèrie per a algunes plaques molt generals, però hi ha d'altres desenvolupades per aficionats i professionals del món de la robòtica i la programació, llibreries que habitualment posen a disposició de qui les vulgui utilitzar de forma gratuïta, seguint la filosofia de l'open-source.



## 3.4. EMMASCARAMENT VISUAL TEMPORAL

L'emascarament visual temporal és un fenomen òptic fisiològic molt utilitzat per aquest treball. L'òptica fisiològica és la ciència que estudia les característiques físiques, biològiques i ambientals de l'ull humà i la influència d'aquestes en la visió.



Representació de PWM. Extret de [www.wikipedia.org](http://www.wikipedia.org)

Aquest fenomen és utilitzat en el treball perquè no és possible controlar la intensitat a la qual brilla la tira de llums led augmentant o disminuint el voltatge subministrat, per tant el que es fa, són parpellejos molt propers en el temps per tal que la vista no capti el temps d'apagada entre aquests cops de llum. Fent això s'aconsegueix

l'emascarament visual temporal, de manera que sembla que els leds brillin amb més o menys intensitat depenent de la durada de les pauses entre cada encesa dels llums.

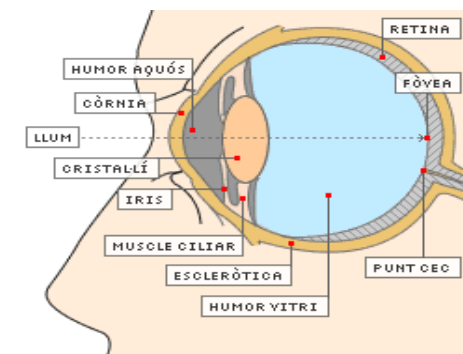
Tot sistema visual necessita un temps de neuroadaptació a la llum, depenent de la persona i les condicions exteriors, aquests períodes poden variar, però en termes generals són:



- 30ms<sup>4</sup>: Temps màxim al qual la vista no capta les pauses entre les enceses de llums, i les emmascara.
- 100ms: De 30ms a 100ms, l'ull pot captar els intervals on la llum està apagada, però no veu la brillantor real d'aquesta, sembla que se'n vegi menys.
- 300ms: A partir d'aquest temps entre pauses, la llum es percep com seria "normal" és a dir, es veu la brillantor màxima de la llum i se'n capten les pauses.

Aquest temps d'adaptació neurològica ve influenciat per l'obertura i el tancament de la pupil·la (oberta o tancada mitjançant el múscul ciliar) i la sensibilitat de les cèl·lules fotoreceptores (cons i bastons).

La pupil·la és un orifici situat a la part exterior de l'ull per on hi entra la llum per tal que la retina la pugui captar i convertir-la en estímuls elèctrics dirigits al cervell. Per tal que la quantitat de llum sigui la necessària, la pupil·la conté al seu voltant un múscul que l'obre o tanca depenent de la llum que entra per aquesta: el múscul ciliar, com l'excitació del múscul ciliar és un procés de transmissió d'impulsos elèctrics enviats des del cervell, això triga uns mil·lisegons que fan que no sigui immediat.



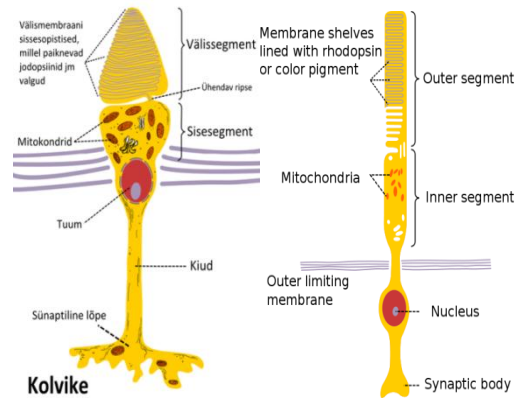
Ull humà. Extret de [www.dtic.upf.edu](http://www.dtic.upf.edu)

<sup>4</sup> Ms: mil·lisegons, mil·lèsima part d'un segon.



Un fotorceptor té la funció de transformar la llum en impulsos elèctrics. Aquesta funció la fan els cons i els bastons, situats a la retina, que transformen les imatges arribades en forma de llum en estímuls elèctrics que s'envien al cervell perquè interpreti la informació.

La funció dels cons és aportar informació dels colors (RGB) i la visió espacial, la dels bastons és aportar informació sobre la brillantor de la llum i proporcionen visió quan hi ha poca il·luminació.



Con i bastó. Extret de [www.wikipedia.org](http://www.wikipedia.org)



## 3.5. PROJECTES AFINS

### 3.4.1. EYEBORG

Neil Harbisson, nascut a Londres a l'any 1982, de pare Irlandès i mare catalana (va créixer a Mataró), és un artista visual i activista.



Fotografia de Neil Harbisson amb l'Eyeborg. Extreta de [www.facebook.com](http://www.facebook.com)

D'ençà que va néixer pateix una infermetat anomenada acromatòpsia, és una malaltia

que afecta els cons i bastons blaus de la retina i que no permet veure els colors, només és capaç de veure en una escala de grisos.



Passaport amb l' Eyeborg incorporat. Extret de [www.plus.google.com](http://www.plus.google.com)

Aquesta discapacitat, juntament amb la seva afició a la música, el va portar a desenvolupar amb 21 anys (conjuntament amb Adam Montandon) un projecte d'aparell que relacionés la freqüència després pels colors en freqüències sonores que pogués sentir, van crear un aparell anomenat Eyeborg el qual porta incorporat al seu cap sent així una extensió del seu propi cos.



Després de molta insistència i recomanacions de metges, Neil ha estat reconegut com el primer ciborg del món (persona amb una extensió electrònica del seu organisme), com es pot veure a la fotografia del seu passaport, ell hi surt amb unes de les primeres versions de l'Eyeborg acoblat al seu cap com una extensió de la seva persona.

Durant els seus estudis a Anglaterra va conèixer a Adam Montandon, llicenciat en cibernètica a la universitat de Plymouth. Els dos universitaris van treballar conjuntament en la creació de l' Eyeborg. Aquest aparell, que en Neil porta instal·lat al cap, consisteix en un sensor de llum penjat d'una antena. Aquesta antena rodeja el cap per la part superior fins a arribar a la part posterior, on es connecta a un petit ordinador que, a través de la cavitat òssia, envia senyals sonors a l'oïda interna. Els senyals sonors que emet són freqüències que, depenent del color que capti (els colors tenen una freqüència lluminosa) el sensor que porta a la part frontal del cap, emet una freqüència sonora relacionada amb la freqüència lluminosa del



*Fotografia de l'Eyeborg. Extreta de [www.facebook.com](http://www.facebook.com)*



d'art creades a partir del Eyeborg amb el qual dibuixa, per exemple, retrats de personalitats importants depenent dels colors de la seva cara, o plasma en un quadre (mitjançant les freqüències despreses) composicions musicals. El xip incorporat i que controla tot el muntatge té incorporat un connector de tipus USB per tal de poder-lo carregar (amb 3 hores de càrrega, funciona gairebé 4 dies) i també per a poder fer actualitzacions del software controlador, aquest programari s'ha anat millorant fins al punt que ara pot percebre colors que ni tan sols la vista humana pot veure (infraroig i ultraviolat propers a la gamma visible).

L'Eyeborg i el meu treball tenen molta relació, ja que aquest s'encarrega de convertir senyal lluminós (visual) en senyal sonor i el meu projecte tracta de la conversió d'una música (sonor) en llum i moviments (visual). Respecte als mètodes utilitzats per a fer aquesta conversió no hi ha cap relació, ja que el funcionament del Eyeborg se centra a detectar la freqüència de la llum és a dir, el seu color; mentre que el meu robot es centra en paràmetres menys objectius com el ritme i l'altura.





### 3.4.2. SPLYCE APP

Splyce és una aplicació per a telèfons mòbils i tauletes que utilitzen el sistema operatiu IOS és a dir, que estan dissenyades per a funcionar en dispositius de



*Fabricant de l'Aplicació. Extret de [www.inqbarna.com](http://www.inqbarna.com)*

la marca Apple, té com a objectiu ser un reproductor de música però amb alguns detalls nous fins al moment en aquest tipus d'aplicació. Va ser llençada el 2 de juliol del 2013 i actualment té vora 600.000 descàrregues i ha guanyat el premi a la millor aplicació de l'any a AppStore (tenda d'aplicacions dels dispositius Apple). Ha estat dissenyada i desenvolupada per l'empresa InqBarna.

L'empresa InqBarna va ser creada al 2009 per Nacho Sánchez i es dedica al desenvolupament d'aplicacions per a mòbils, on és un referent del sector, principalment les creen pel sistema operatiu IOS, tot i que també treballen amb Android i Windows mobile. Aquesta



*Aplicació.  
Extret de [www.inqbarna.com](http://www.inqbarna.com)*

empresa ja ha creat 25 aplicacions entre les quals hi ha de pròpies i d'encarregades per tercers.

Principalment, Splyce, és un reproductor de música que fa sonar les cançons que es tenen guardades a la memòria del dispositiu o es poden crear llistes de





reproducció de diferents fitxers, però incorpora algunes funcions que no tenen la resta de reproductors i fan que sigui diferent.

Abans de començar a sonar la cançó o llista de reproducció, l'aplicació fa un rastreig de les cançons que sonaran per buscar-hi els paràmetres del ritme, intensitat i altura en cadascuna d'elles, això permet que l'aplicació pugui tenir algunes funcionalitats extremes respecte els altres reproductors, algunes d'aquestes funcions són: mescla automàtica de cançons, efectes sonors, efectes visuals i canvis d'interfície, equalitzador, ordenació de la llista de reproducció depenent del ritme de les cançons, diferents representacions visuals del ritme...

La funció que més destacada d'aquesta aplicació és la mescla de dues cançons automàticament quan s'està acabant la primera i comença la següent, aquesta funció és igual que la d'un DJ de discoteca, fusiona dues cançons de manera consecutiva ajustant el ritme d'aquestes perquè encaixin. El programa presenta 4 tipus de transicions de música diferents:

- Parada de la primera cançó amb el soroll d'un vinil i reproducció instantània de la segona.
- Disminució del volum de la primera mentre augmenta el de la segona linealment.
- Bucle del final de la primera cançó i reproducció de la segona ("echo freeze").
- Disminució de les freqüències greus de la primera cançó i acoblament amb la segona, la qual té silenciats les freqüències agudes.



L'origen d'aquesta aplicació prové d'una anterior (també de InqBarna) que era una taula de mescles de cançons virtual (anomenada "deej"), aquesta aplicació agradava molt a la gent, doncs era força divertit fer mescles a l'estil d'un DJ professional però, com diu Nacho Sánchez a l'entrevista, "la gent es divertia 10 minuts i després no sabien que fer perquè no ho sabien utilitzar bé". Tenint en compte això, van crear aquesta nova aplicació que, amb diferents transicions fetes de sèrie, pot fer les fusions o mescles de música automàticament, i es poden ajustar només alguns paràmetres com el tipus de transició, l'efecte o el temps del pas entre cançons.



*Reconeixement millor app 2.013  
Extret de la pròpia web*

La principal similitud d'aquesta aplicació amb el meu treball està relacionada amb els ritmes de la música. Tenint en compte aquesta semblança entre els projectes, vaig aprofitar la trobada per preguntar quin sistema utilitzaven per analitzar el ritme d'una cançó perquè jo dubtava de com fer-ho: pensava que es podia calcular a la mateixa vegada que sonava la cançó.

Parlant amb en Nacho Sánchez em va explicar que el ritme d'una cançó no es pot detectar a l'instant que sona, ja que és un procés impossible actualment, el que fan ells és analitzar-lo abans que comenci la cançó i mitjançant un motor d'anàlisi molt complex (reciclat del programa "deej"), aquest motor era massa complex per



a la placa d'Arduino així que em va aconsellar que no analitzés el ritme, sinó que l'agafes d'una font externa. Aquest consell em va simplificar molt la feina, ja que era un repte realment difícil o fins i tot impossible en el qual vaig estar invertint-hi molt de temps sense resultats. Això no vol dir que en un futur es busqui la manera de fer-ho en temps real degut a les seves clares avantatges. Si fos així, es podria implementar perfectament en el robot, doncs els seus sistemes de càlcul i la rapidesa de programació ho permetria.



*Nacho Sánchez, fundador de InqBarna amb el "White" i jo. Font pròpia*



## 4. PRACTICA

### 4.1. COMPOSICIÓ DEL ROBOT

#### 4.1.1. Plaques

Juntament amb el programari, el maquinari és la part principal del robot. Està format per plaques, que són elements amb un o més senyals d'entrada i un o més senyals de sortida, que depenent dels estímuls que li arriben per l'entrada cada placa desenvolupa una funció diferent depenent del seu objectiu a l'autòmat.

En aquest robot hi ha un total de 6 plaques:

- 3 Plaques base "Arduino Uno"
- 1 Placa "shield" de motors
- 1 Placa "shield" de relé
- 1 Placa de control de llums (de fabricació pròpia)

#### Arduino Uno

La placa Arduino Uno és una placa electrònica que fa la funció de placa base en aquest robot. Dins d'Arduino és una de les plaques més bàsiques i comuns que hi ha, ja que és assequible i versàtil.



Placa base Arduino. Extret de la propia web



Presenta les següents característiques:

- Microprocessador ATmega328 (de la companyia Atmel).
- 16MHz de velocitat de rellotge (processos per segon).
- 32Kb de memòria flash <sup>5</sup> (0,5Kb són utilitzats pel gestor d'arrencada).
- 2Kb de memòria SRAM<sup>6</sup>.
- 1Kb de memòria EEPROM<sup>7</sup>.
- 5V de tensió de funcionament.
- Voltatge d'entrada compres entre 7V i 12V.
- 14 pins digitals (entrada o sortida), 6 dels quals proporcionen PWM.
- 6 pins analògics (només d'entrada).
- Sortida de 5V i 3.3V.

Totes les plaques base Arduino (excepte Arduino Esplora) presenten una sèrie de pins, que són uns forats per connectar-hi els diferents elements per tal que la placa interactuï amb altres dispositius, poden ser d'entrada i de sortida o només d'entrada.

Alguns d'aquests pins són GND "ground", que fan de massa on es connecta el negatiu de tots els circuits que es desenvolupin a partir d'un senyal que entra o surt pels altres pins.

Els pins que s'encarreguen de donar o rebre informació (en forma de voltatge) només poden ser de dos tipus:

---

<sup>5</sup> Memòria flash: Memòria on s'emmagatzema el programa carregat.

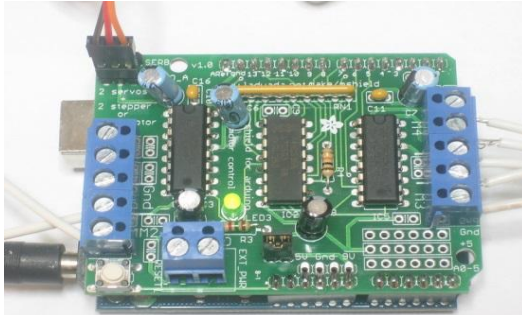
<sup>6</sup> Memòria SRAM: Memòria on es guarden les variables dels programes.

<sup>7</sup> Memòria EEPROM: Memòria de només lectura amb les instruccions bàsiques d'arrencada de la placa.



- Analògics: Amb valors compresos entre el 0 i el 1024, llegeixen el voltatge (fins a un màxim de 5V) que els arriba d'una font externa. Amb un altre pin anomenat ARef "analogic referent" es pot augmentar el voltatge màxim al qual estarà exposat l'entrada.
- Digitals: Poden ser tant d'entrada com de sortida (la sortida subministra 5V), qualsevol dels dos casos ha de ser en binari (hi ha corrent o no n'hi ha, cert o fals).

Tot i que els pins digitals no poden donar valors analògics, si es treballa amb llum, com és el meu cas, es pot enganyar a l'ull humà i fer que sembli una sortida analògica. Aquests són els ports PWM "pulse-width modulation" (modulació per ample d'impuls), aquestes sortides són capaces de donar valors vertader i fals (voltatge o no voltatge) a una velocitat molt elevada, mitjançant la funció "WriteAnalog" se li pot donar un valor entre 0 i 255; depenent d'aquest valor el senyal de no voltatge (no fa llum) té una durada més extensa o menys (com més alt el nombre, menys durada té); per tant si el nombre és més elevat, fa les pauses més llargues i els cops de llums tenen més separació en el temps (tot i que no s'aprecia aquesta pausa a simple vista). Fent aquest sistema, es produeix l'emascament visual temporal, que com s'ha explicat anteriorment si aquestes pauses són llargues, sembla que la llum tingui menys brillantor que quan la pausa és més curta.



"shield" de motors. Extret de [www.adafruit.com](http://www.adafruit.com)

### "Shield" de motors

Com he explicat, un "shield" és una placa d'expansió especialitzada en una funció concreta, en el cas d'aquesta és especialitzada en el control de motors (que representen el ritme fent un moviment de ziga-

zaga). El model de placa que utilitzo ha sigut creat per Ladyada i distribuïda per Adafruit Industries, un dels principals distribuïdors Arduino. Tot i que el programa Arduino té algunes biblioteques pròpies, per utilitzar aquesta placa és necessari instal·lar una especial que no hi és, s'anomena "AFMotor" i es pot descarregar gratuïtament al web del fabricant, aquesta llibreria activa noves funcions per fer més fàcil la programació amb la placa.

Les característiques d'aquest "shield" són les següents:

- Connexió per 3 servomotors.
- Connexió per 4 motors DC o 2 motors pas a pas.
- 2 chips L293D que ofereixen 0.6A a cada pont.
- Botó de reinici de la placa base.
- Alimentació exterior.

L'última característica va ser molt important pel funcionament de l'autòmat, ja que els motors necessitaven un voltatge més elevat del que la placa base li podia subministrar, el que vaig fer va ser posar-hi una alimentació externa especial per als motors, sense aquesta alimentació tenien molt poca potència i el robot no arribava a moure's per si sol.





“Shield” Relé. Extret de [pt.aliexpress.com](http://pt.aliexpress.com)

### “Shield” de relé

Un relé és un element electromagnètic que separa dos circuits amb voltatges diferents (un de feble i un de més elevat), l'estimulació del relé mitjançant el circuit de corrent feble tanca el segon circuit de voltatge més elevat per tal que hi pugui circular el corrent. La funció d'aquest “shield” és que mitjançant un sol ordre (HIGH o LOW) es pugui excitar el relé per tancar un altre circuit, generalment un relé serveix per tancar un voltatge elevat mitjançant un altre circuit però en el meu cas l'utilitzo com un botó que tanca el circuit d'encesa/pausa de la música de la font de so.

Aquest “shield” conté un relé que funciona a 5V, també un led que indica si li arriba corrent i un altre que indica l'estat del relé.

S'utilitzen 3 dels 5 pins que té:

- Entrada de corrent a 5V.
- Entrada de senyal per fer el canvi d'estat.
- Negatiu (GND) comú tant pel senyal d'entrada com pel corrent.





### 4.1.2. Estructura

---

El xassís del robot és una simple placa de metacrilat de color negre amb diferents forats que, juntament amb els motors, pneumàtics, llantes i la roda guia, van ser comprats per Internet, ja que no són elements molt específics i, especialment els motors, són els més estàndard que hi ha. Els altres dos elements que formen l'estructura general del robot són la coberta superior i els pilars que la suporten, tot i que més endavant ho explicaré, aquestes parts són reciclades i adaptades per mi. He utilitzat cargols, volanderes i femelles, així com brides per a la subjecció dels elements. En algun cas he posat goma calenta per a evitar vibracions i millorar la subjecció.

El xassís té multitud de forats de diferents formes i mides, a més d'algun afegit per mi per a poder subjectar-hi elements que no encaixaven. Aquests forats tenen dues funcions, la més bàsica que és subjectar els pilars i les torretes que aguanten els diferents elements suportats a la base així com les brides que he utilitzat per aguantar-ne d'altres i la segona funció que fan és poder passar cables des de la part inferior del robot a la superior sense que hagin que anar per fora i que quedi antiestètic o fins i tot molestar les rodes amb fregaments i entrebancs.

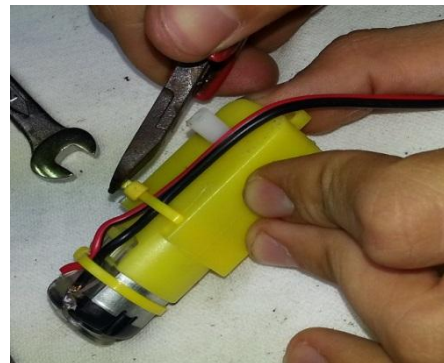
Aquesta base, de color negre i cantonades arrodonides, presenta unes mides de 159mm de llarg, 95mm d'ample i 3mm de gruix.

El robot disposa de 2 motors i cadascun d'aquests motors està compost per les següents peces: motor, transmissions reductores,



dues rodes amb la llanta i finalment, una roda guia o també anomenada "roda boja". Aquest model de motor és molt utilitzat per aficionats a la robòtica i als projectes d'Arduino.

- Motors: poden funcionar de 3V fins a 12V amb una intensitat de 95mA. I assoleixen una velocitat de 150rpm a 6V.
- Transmissions Reductores: es caracteritzen per tenir una relació de transmissió d'1:42.



*Motor. Font pròpia*



*Rodes i llantes. Font pròpia*

- Rodes i llantes: són de 64mm de diàmetre, 27mm d'ample i presenten un dibuix al pneumàtic molt marcat que afavoreix l'adherència a terra i això és positiu perquè així el robot no rellisca quan utilitza els motors i per tant no es desvia del seu camí.



- Roda Guia: Aquesta peça està formada per una base metàl·lica subjecta al xassís de metacrilat i hi té una roda acoblada que pot girar lliurement endavant, enrere i 360°.

La funció d'aquesta roda guia és tenir un punt de suport del robot amb el terra però sense que afecti la mobilitat del robot, ja que cobreix tots els possibles girs de l'autòmat.



*Roda guia. Font pròpia*

Per tal de poder fixar aquesta peça a la placa de metacrilat només es van fer dues modificacions: una va ser els cargols, que van ser substituïts per uns de més petits i curts per una qüestió estètica.

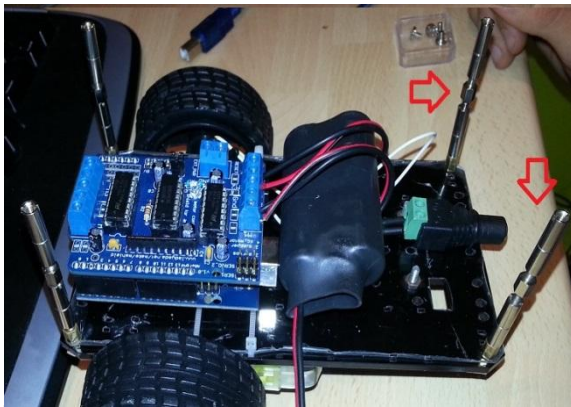


*Fixat de Roda guia. Font pròpia*

L'altra modificació va ser afegir unes volanderes "Glober" entre la volandera normal i la femella; aquest element incorporat serveix per a absorbir les vibracions creades pel moviment del robot i d'aquesta manera no es transmeten a les femelles, que es podrien anar afluint per les tremolors.



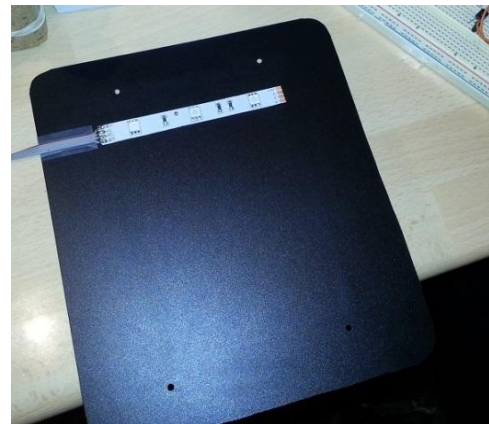
Finalment cal parlar dels pilars. Aquests elements (dos d'ells són de



*Pilars. Font pròpia*

52,5mm i els altres dos de 48mm) són quatre columnes de torretes muntades una sobre l'altre, degut que el robot té una inclinació cap a la part del darrere, s'hi han col·locat pilastres de dues mides diferents per a intentar contrarestar aquest lleuger desnivell.

L'últim element estructural del robot és la coberta superior: està feta a mà i cobreix completament l'interior del robot; d'aquesta manera observant-lo verticalment no es pot veure cap peça de les que hi ha a l'interior.



*Coberta. Font pròpia*

La coberta té també una fina finestra amb un doble recobriment de plàstic translúcid a sota dels quals hi va el sistema d'il·luminació. L'objectiu d'aquesta obertura és que es puguin veure els llums des de sobre però sense que aquesta sigui directa, ja que el plàstic translúcid dissipa lleugerament la llum. La coberta té unes mides de 19cm de llarg, 17cm d'ample i 0,7mm de gruix amb uns orificis practicats a les mides exactes perquè coincideixin amb els pilars subjectats al xassís.



### 4.1.3. Il·luminació

---

La llum que projecta el robot per tal de representar els paràmetres de l'altura i el volum de la música és una de les parts principals, si més no, la més vistosa i atractiva.

La il·luminació que fa el robot es pot desglossar en dos paràmetres que utilitzo per variar-la: la intensitat de la llum i el color d'aquesta. Com la resta de paràmetres, també està controlada per una placa base Arduino Uno.

Com he explicat abans la placa base no disposa de sortides de tipus analògic, per tant no puc variar el voltatge de sortida que li arriba als leds. Això significa que no puc canviar la lluminositat del led de manera convencional, és a dir, baixant el voltatge per fer que brilli més o pujant-lo perquè s'il·lumini més. Per a crear la sensació que la llum després pot ser més lluminosa o menys utilitzo la sortida digital, i fent parpellejar cada led a una velocitat molt ràpida puc aconseguir un emmascarament visual temporal, que com he explicat en un punt anteriorment, quan es veu el led parpellejar molt ràpid, la vista el capta com si estigués totalment il·luminat; si les pauses entre encesa i encesa són més gran, però segueixen estant a una velocitat imperceptible per a la vista humana, dona la sensació que té menys lluminositat que quan els pipellejos són més seguits. Aquest emmascarament visual temporal és utilitzat per als dos paràmetres esmentats de la llum.



El circuit d'il·luminació necessita un voltatge de 12V, però la placa base només en pot subministrar 5V. La primera solució que hi vaig pensar va ser posar-hi un relé que separés el circuit de 5V i el de 12V i pogués activar-lo des del robot, però em va semblar que com era un element mecànic, no podria fer els contactes a la velocitat necessària així que finalment vaig optar per un optoacobrador. Un optoacobrador que és un element electrònic amb la mateixa funció que un relé, separa dos circuits elèctrics de diferents voltatges de manera que activant o desactivant un dels circuits es pot activar o desactivar l'altre, l'avantatge que em proporciona aquest element respecte al relé és la velocitat, ja



*Placa resistències leds. Font pròpia*

que mentre el relé separa els dos circuits per una bobina que crea un camp magnètic i ajunta o separa dos borns, l'optoacobrador és un petit encapsulat amb un led infraroig que emet llum quan li arriba corrent i un fototransistor connectat al segon circuit capta aquesta llum infraroja i tanca el segon circuit, el led utilitzat és infraroig perquè aquesta llum és més ràpida de transmetre que una de l'espectre visible.

Pel que fa al color emès pel led, no s'ha emprat cap element de separació de circuits, ja que no és necessari un voltatge superior a 5V per a anular el voltatge provinent de la pila de 12V, que alimenta al sistema d'il·luminació.

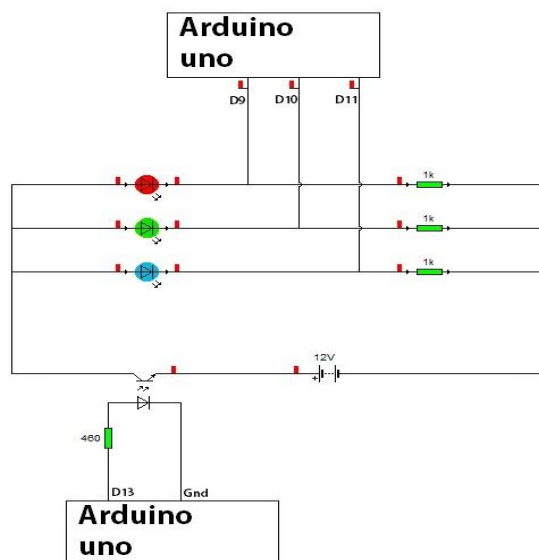




L'emissor de llum emprat, després de diferents proves sobre el funcionament del canvi de color dels leds automatitzat amb la base, és una tira de leds de 10,20cm (és el mateix sistema que es pot utilitzar per a posar llums decoratives d'ús domèstic. Aquesta tira té leds RGB i al sistema original un comandament a distància per controlar la intensitat i el color. Vaig investigar i provar com funcionava el canvi de color per poder-lo incorporar al "White" sense necessitat d'un comandament, i que fos automatitzat amb la base.

La tira té quatre contactes, un pel positiu de 12V i tres pels negatius corresponents als colors verd, vermell i blau de cada led. Cada díode led RGB conté tres petits díodes a l'interior, un vermell, un verd i un blau. Per controlar el color s'ha de fer combinacions connectant o desconnectant el negatiu de cadascun d'aquests tres leds de colors.

No obstant això, utilitzant aquest sistema de connexió o desconnexió amb tres negatius, només és possible obtenir sis colors. Per tant, donat que l'objectiu era crear més colors i escales d'un color a un altre oposat, ha estat necessari fer proves amb el sistema de connexió dels diferents leds.



*Circuit controlador de la il·luminació. Esquema de font pròpia.*

Tal com es pot observar a la imatge del circuit controlador de



la il·luminació (esquema de font pròpia), els leds estan alimentats per una pila de 12V que mantindria encesos sempre els tres colors però si al negatiu de cadascun dels colors es connecta la placa base amb una sortida diferent, es pot variar el voltatge que retorna cada negatiu aplicant un voltatge de 0V o 5V des de la placa per tal de canviar l'estat de cada led.

Finalment, després del díode de led i de l'entrada de la placa base, ha estat necessari col·locar una resistència per a cada branca del circuit de 1 K $\Omega$  per tal d'aconseguir que l'alimentació de la pila no creï un curtcircuit amb la placa i faci malbé la sortida digital.





#### 4.1.4. Font d'energia

---

La font d'energia és la part del robot encarregada de subministrar el corrent elèctric necessari a les diferents parts i components d'aquest; l'autòmat necessita corrent per a les plaques, les rodes, el sistema d'il·luminació (que n'utilitza dues) i la font de so, però aquest últim ja en porta una integrada de sèrie.

Per si soles, les plaques del robot, necessiten un voltatge comprés entre els 7 i els 12V, i els motors entre 3 i 12 volts amb un mínim de 95 mA.

Per tal que sigui el més independent possible, he optat perquè vagi alimentat per bateries recarregables, ja que així no depèn de cap cable; totes aquestes bateries, excepte la del sistema d'il·luminació, són recarregables per no haver que desmuntar-ho tot per accedir a canviar les piles.

Com ja he dit, la font de so té una bateria recarregable pròpia de 3.7V i 1020 mAh, la qual no he modificat ni utilitzat per a cap altre element.

La il·luminació va alimentada per una pila de 12V i 55 mAh, l'elecció d'aquesta va ser perquè la tira de leds és sobrant d'un sistema més gran que té un transformador propi i funciona a 12V. Aquesta pila no té una gran capacitat ni potència, però és suficient per alimentar al sistema d'il·luminació i a més, la seva reduïda mida suposa un gran avantatge alhora de fixar-ho a la base.



Respecte a les bateries que subministren corrent a les plaques i els motors, vaig desestimar des d'un primer moment l'opció de posar una bateria recarregable de 12V comuna, ja que és massa gran per al xassís i desequilibraria el robot pel seu pes.

La primera solució que vaig incorporar per alimentar les plaques i els motors va ser una pila de 9V, però no tenia suficient intensitat per alimentar a tot el que era necessari.

Mentre no aconseguia cap altra font d'alimentació que s'ajustés als requeriments, feia les provatures amb un transformador de 9V connectat al corrent altern de casa; aquest subministrava el voltatge necessari i, com prové d'una línia de corrent d'ús domèstic, no hi ha cap tipus de problema en la intensitat elèctrica.

Però la falta de mobilitat em suposava un gran problema, ja que el robot només es podia moure en un radi igual de llarg que el cable de corrent i l'havia d'anar seguint subjectant-lo per tal que les rodes no s'entrebanquessin amb el cable.

La tercera i darrera solució per la qual vaig optar va ser un parell de bateries recarregables de 3.7V connectades en sèrie, aquesta idea me la va donar un aficionat a la robòtica que em deia que ells sempre utilitzen aquest sistema.



Per no haver que posar un porta piles al robot, ja que ocuparia molt d'espai, vaig fer jo la connexió de les bateries, soldant-les consecutivament i amb un connector de tipus DC jack als extrems.

L'elecció d'aquest tipus de connector va ser degut que és molt estàndard i comú per a corrents continus de baix voltatge.

Finalment vaig unir les dues piles amb un macarró termoretràctil per tal d'evitar curtcircuits amb altres elements, evitar que es dessoldi i poder tenir-ho tot compactat en un mateix element.

Com el connector és molt comú per a carregadors i bateries, puc aconseguir fàcilment un transformador de corrent de 9V per carregar les bateries i no tenir que desmuntar el paquet si es gasten.

És important destacar i tenir en compte que el funcionament del robot es fonamenta en el sistema Arduino i aquest està pensat per que es nodreixi fonamentalment d'un alimentador connectat al corrent domèstic. Per tant, el consum elèctric del robot és molt elevat degut que les plaques segueixen consumint encara que no hi hagi cap programa o aplicació en ús.

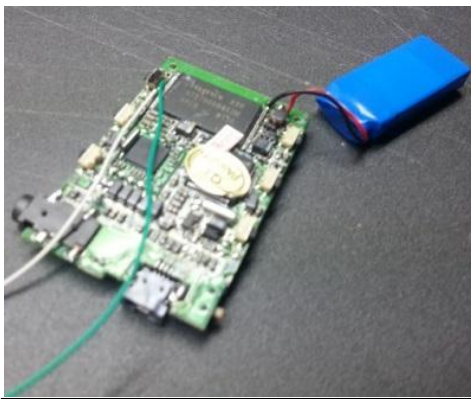
Aquesta qüestió, va estar comentada personalment a l'entrevista amb David Cuartielles, creador de la plataforma Arduino. Ell mateix afirma que en una de les pròximes actualitzacions que estan previstes, volen millorar aquest problema, ja que Arduino s'està implementant molt en sistemes que no van endollats.



#### 4.1.5. Font de so

---

La font de so és un dispositiu que envia el senyal musical al robot per tal que pugui analitzar i representar els paràmetres de volum, ritme i altura.



*Font de so. Font pròpia*

En aquest cas concret, la font de so utilitzada és un reproductor de música digital que proporciona la música necessària perquè el robot es mogui (connexió directa del reproductor amb la placa base) i emeti senyals lluminosos que poden variar segons el tipus de música que es reproduïx.

A més, també emet la música a l'exterior mitjançant un altaveu que porta incorporat.

La font de so és un element transcendental per entendre el funcionament, moviment i representació de la música que fa el robot.

La primera font de so que va tenir el robot va ser l'ordinador, al que li vaig connectar un doblador de senyal de tipus mini jack per tal de poder sentir la música per els altaveus i tenir un cable al qual connectar el robot per a fer les provatures.

Aquest sistema va ser molt còmode al principi, ja que des de l'ordinador podia carregar els programes alhora que encenia o



apagava la música, però en el moment que el robot necessitava moure's ja vaig hi haver que canviar de sistema.

Quan vaig començar a utilitzar motors i, en conseqüència, començar a moure el robot. Vaig haver de canviar de font de so per una de mòbil.

Juntament amb això, sorgia un nou problema que no vaig saber solucionar fins al final, aquest problema era la sincronització entre els programes i la música, ja que com ho encenia manualment, no podia sincronitzar-ho perfecte.

La segona font de so que vaig utilitzar va ser el telèfon mòbil, el qual connectava amb un cable dins del robot i el subjectava a la base, com he dit, em sorgia el problema de la sincronització, així que de moment el que feia era engegar manualment la música i el programa tan simultàniament com m'era possible.

Quan ja tenia tots els programes fets, em vaig centrar en buscar una solució a la sincronització.

En un primer moment vaig pensar que una de les plaques base pogués detectar el primer so de la cançó i així engegar tots els programes alhora, però seguia sense ser del tot sincronitzat, i hi havia un petit desfasament dels programes respecte al so.

Com aquesta primera versió no va ser del tot bona vaig seguir investigant fins a trobar una segona i definitiva via per solucionar-ho,



vaig substituir el telèfon mòbil per un reproductor de música portàtil, el qual vaig desmuntar i vaig fer-hi sortir dos cables molt prims connectats als dos pols del botó d'encesa i pausa de la música.

Quan un botó és polsat el que fa és comunicar aquests dos pols permetent que hi passi l'electricitat i així enviant un senyal perquè la música comenci o es pari.

Amb els dos cables que sortien dels pols podia engegar la música ajuntant els cables sense necessitat de pressionar el botó, un cop fer això només vaig haver de connectar els cables a un relé controlat per la placa base i programar un petit inici encarregat de donar la sortida a les dues plaques restants i la música;

Un relé és un dispositiu que permet tancar o obrir un circuit elèctric per mitjà d'un altre circuit, en el meu cas el primer circuit seria el dels cables que envien el senyal tancant el circuit d'encesa de la cançó i el circuit de control estaria connectat a un dels pins de la placa.



## 4.2. PROGRAMES D'ANÀLISI

### 4.2.1. Intensitat

Aquest programa s'encarrega d'interpretar el volum al qual està sonant la música i, d'acord amb això, variar la brillantor que presenta el conjunt de leds representants del robot, aquest és l'únic programa que analitza i representa la música que està sonant al mateix instant.

Com ja he mencionat en diferents ocasions en aquest treball, el voltatge de sortida de la placa base Arduino no és variable, per tant no és possible variar la lluminositat del led mitjançant aquest sistema. Així que faig les sortides PWM per crear un emmascarament visual temporal.

La funció principal d'aquest programa és la mateixa que la d'un vúmetre, que és un dispositiu sovint incorporat en equips d'edició i mescla d'àudio, el qual indica el volum del senyal acústic al qual està connectat.



Vúmetre digital (Esquerra) i vúmetre analògic (dreta).  
Imatge de font pròpia

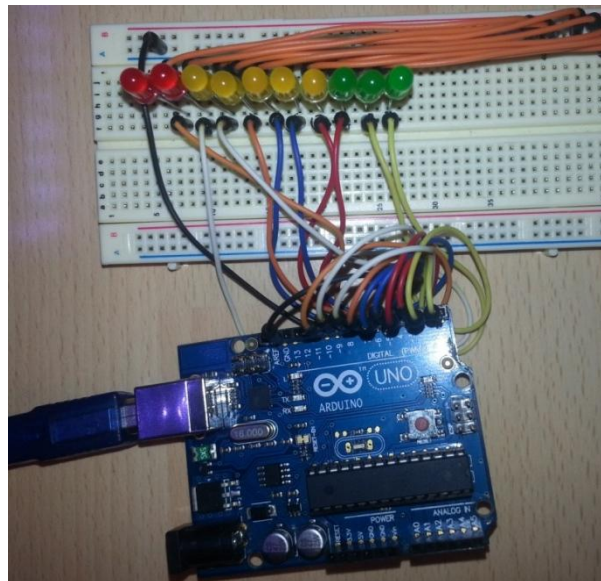
Un vúmetre digital, és a dir, els que representen amb una sèrie de llums i no amb una agulla, s'il·luminen totes les llums des de la més inferior fins a la superior



variant l'altura d'aquesta depenent si sona amb més o menys intensitat (com més alt arriba la llum, més volum té).

Abans de fer aquest programa d'interpretació del volum d'un so vaig fer un vúmetre.

Primer vaig connectar un micròfon a la placa base; un micròfon està compost per un imant que mitjançant la membrana a la qual està subjectat capta les vibracions emeses per una font de so, aquest imant vibra



*Vúmetre digital fet per mi, i controlat per Arduino.  
Imatge de font pròpia*

lleugerament amb les oscil·lacions creant un petit voltatge que envia a través del cable.

Sabent això vaig utilitzar un programa molt bàsic anomenat "W\_lector" (el podeu trobar a l'annex 2), que llegeix el voltatge que li arriba per un dels pins analògics i els mostra en un monitor en forma de valor numèric (en una escala de 0 a 1024), un cop obtingut aquest valor el vaig dividir entre 100 per a poder obtenir un valor màxim de 10.

Connectant deu leds a deu sortides digitals diferents, encenia des de el led número 1 fins al valor que obtingués del programa, de manera que com més fort era el so entrant, donava un nombre més alt i per





tant, s'encenia la fila de leds més amunt. Aquest programa es pot trobar a l'annex 2 com a "W\_vumetre".

Un cop aconseguit el programa anterior vaig avançar donant-li una forma més propera a l'objectiu d'aquest programa dins del robot, ja que aquest no incorpora cap vúmetre convencional.

Com l'objectiu final d'aquest programa és controlar la intensitat d'una tira de leds, vaig optar per començar a variar la intensitat d'un sol led dependent del volum de la cançó.

Un cop aconseguit el circuit amb l'optoacoblador esmentat anteriorment, a aquest últim programa no li va caldre cap reforma, ja que funciona igual per a un sol led que per a tota la tira amb el sistema d'enllaç dels dos circuits.

A l'Anex 1 s'hi pot trobar el programa o aplicació final, que utilitza el robot per funcionar. En aquest cas, pot representar qualsevol cançó perquè agafa els valors directament.



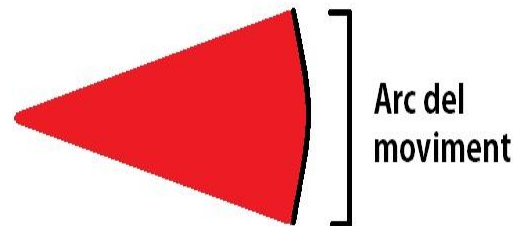
## 4.2.2. Ritme

---

El ritme d'una cançó és la característica que marca la seva velocitat.

Al robot, aquest paràmetre està relacionat amb el seu moviment.

L'autòmat gira circularment,



*Moviment teòric del robot. Imatge de Font*

utilitzant només un motor, fins a cada "beat" de la música, en el qual canvia el motor de gir i per tant la direcció del moviment.

Això crea un moviment de ziga-zaga que, depenent del ritme de la música; si el ritme és ràpid, els batecs per minut són més ràpids i la direcció canvia més ràpidament, com sempre porta la mateixa velocitat al motor que gira, l'arc del moviment serà més curt.

Hi ha dos aspectes que dificulten molt la detecció del ritme d'una cançó: el primer és que una cançó pot no portar el mateix ritme tota l'estona, és possible que variï durant la cançó; i el segon aspecte és que en una cançó hi toquen molts instruments i, tot i que majoritàriament un en porta el ritme més bàsic, la barreja de tots els sons fa molt difícil l'aïllament d'aquest únic ritme.

En un primer intent de fer aquest programa vaig buscar el ritme de la cançó per Internet. Per interpretar les cançons, els músics han de saber el ritme que porta i això està determinat pels batecs que fa la



cançó per minuts (BPM) és a dir, per la duració de les notes que sonen una respecte a les altres.

Fent el càlcul:  $\left(\frac{1}{\frac{\text{BPM}}{60}}\right) \times 1000$ , aconseguia saber el temps (en mil·lèsimes de segon) que passen entre "beat" i "beat", aquesta mesura l'he anomenat mil·lisegons per beat (mspb).

En un primer moment no vaig utilitzar les rodes, la primera versió va ser donant un cop de llum amb un led a cada "beat" de la música, el temps que passa entre cada encesa de la llum m'era proporcionat pel nombre anterior (podeu trobar aquest programa a l'annex 2, "W\_led").

Aquest sistema no m'acabava de convèncer, ja que havia de buscar el ritme per Internet cada vegada que canviés de cançó.

Primer vaig trobar un programa que detectava el ritme, però no era gens efectiu; no vaig entendre com ho feia, però els valors que donava no es corresponien al que em marcava les pàgines web (molt més fiables).

Com no ho aconseguia vaig demanar consell a l'Eloi Vilalta, estudiant d'enginyeria electrònica industrial i enginyeria mecànica al Tecnocampus Mataró i aficionat a la robòtica.

Ell em va proposar de fer una taula de valors analitzant els primer 20 segons de la cançó per detectar el seu ritme mitjançant els canvis



provocats al volum pels instruments, després de molt provar, tampoc va ser útil, ja que el robot estava molta estona quiet abans de començar la cançó i els valors que em sortien a la taula del ritme no eren regulars ni fiables.

Més tard vaig tenir l'entrevista amb Nacho Sánchez, programador d'aplicacions de música per a mòbils, em va explicar que el ritme no es pot detectar al moment com jo pensava, ni tan sols ells podien amb les aplicacions per a mòbils (molt més potents que la placa base Arduino Uno), primer agafen la música, l'analitzen i després fan servir les dades, però al moment no és possible. Amb un potent motor creat per ells, aconsegueixen captar el ritme de la cançó en poc més de dos segons abans de començar a sonar.

Sabent això, vaig tornar al sistema inicial de buscar el ritme per Internet. Vaig seguir buscant maneres més fàcils de fer-ho i el màxim que he aconseguit arribar és posar-ho manualment però amb l'ajuda d'una pàgina web que t'ho diu exacte i d'una manera clara, aquesta pàgina és [www.jog.fm](http://www.jog.fm).

El programa final, es pot trobar íntegrament al annex1, amb l'explicació de què fa cada procés.

A l'Anex 1 s'hi pot trobar el programa o aplicació final, que utilitza el robot per funcionar. En aquest cas, com no es un valor que es pugui analitzar al moment (juntament amb l'altura), forma part de la cançó "Fake tales of an Francisco" del grup "Artic Monkeys".



### 4.2.3. Altura

---

L'altura d'una cançó és el paràmetre que defineix l'agudeses o gravetat del so d'aquesta.

Al robot, aquest paràmetre es representa amb el color de les llums de la part superior, canviant de color depenent de l'altura de la cançó. Fent colors blavosos per als tons aguts i colors més càlids per als sons més greus.

L'assignació de colors càlids pels greus greu i blavosos pels aguts, l'he feta sense cap motiu especial, simplement és una manera com qualsevol altre de representar-ho.

Per entendre el següent apartat s'ha de saber alguns conceptes bàsics de teoria musical:

- Una partitura és una transcripció escrita de les notes que ha de tocar un instrument, aquestes notes estan ordenades de més agut a més greu formant octaves (com més a baix més agut és).
- Una octava és una sèrie de sons que comprenen les 8 notes d'una escala música (do, re, [...] si, do)
- Dues octaves musicals no formen 16 notes, ja que la primera d'aquestes acaba amb el mateix "do" que comença la següent, per tant dues octaves són 15 notes.

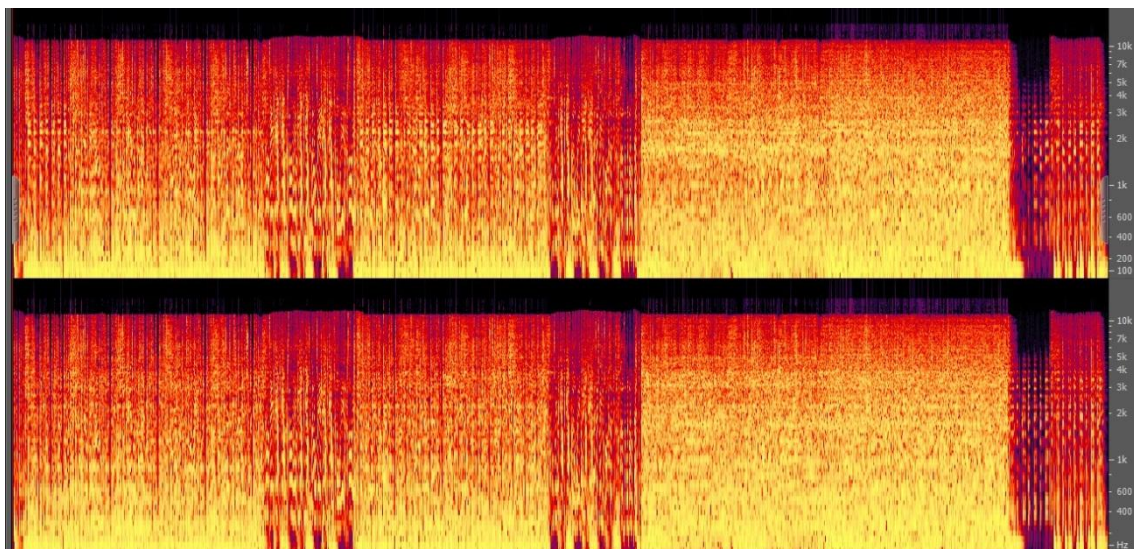


Per fer el programa d'anàlisi i representació de l'altura d'una cançó vaig fer servir diferents mètodes, ja que no ho aconseguia.

Primer vaig utilitzar un oscil·loscopi per llegir l'altura d'un so, ja que si hi connectes un cable amb una freqüència, pots veure la longitud d'ona d'aquesta i determinar si és més greu o més agut depenent de la llargada d'aquesta.

Però en fer la prova vaig adonar-me que com en una cançó hi ha molts instruments i cada un d'ells fa moltes freqüències diferents era molt difícil determinar-ne una freqüència predominant.

La segona alternativa en que vaig pensar va ser en un espectrograma de la cançó, en aquest s'hi mostren totes les freqüències que sonen a la cançó i relaciona mitjançant el color si una freqüència sona més o



*Espectrograma tret d'un analitzador de musica. Font pròpia*



menys que una altra, així podia saber quina d'elles predominava i definir un "to general" de la cançó.

Però em sorgien dos problemes:

- El maquinari d'Arduino no era suficientment potent com per fer un espectrograma, i si utilitzava un petit ordinador connectat amb la placa, no tenia coneixements per transmetre les dades d'una a l'altre.
- Necessitava fer un programa, no aplicable al Arduino sinó a l'ordinador extern, que analitzés el color de l'espectrograma i ho transformés en valors. Això és massa difícil pels meus coneixements de programació.

Com no aconseguia fer cap altre sistema de representació de l'altura d'una cançó vaig optar per l'única solució que se m'acudia i pogués sense fer servir coneixements avançats de programació.

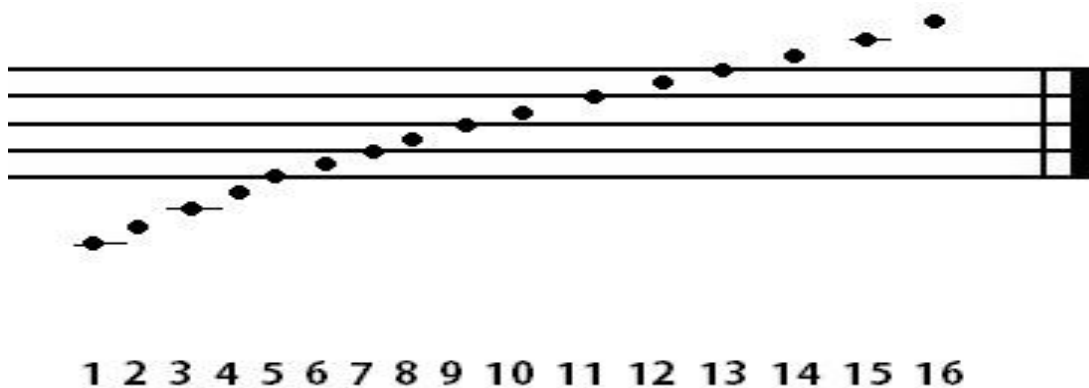
Aquest sistema consistia a agafar la partitura d'un dels instruments de la cançó, comparar nota màxima i la nota mínima que tocava l'instrument, fer una escala de nombres (de l'1 al 15) que correspon a dues octaves musicals, i un cop feta aquesta assignació, donar-li un valor a cada nota que hi havia a la partitura de l'instrument en qüestió depenent de la posició que es trobava la nota al pentagrama (i per tant si era més agut o més greu).

A totes les cançons analitzades he agafat la guitarra com a instrument a representar, ja que és l'instrument principal de les melodies usades.



Com he explicat anteriorment, un led RGB està format per tres leds de colors verd, vermell i blau al seu interior, i la combinació d'aquests tres colors (augmentant o disminuint la lluminositat de cadascun) fa que es puguin veure totes les freqüències de la llum de l'espectre visual.

Però com també he dit, no es pot baixar el voltatge d'un led perquè brilli menys, així que faig servir un cop més les sortides PWM per crear l'emascarament visual temporal.



*Assignació de valors de dues octaves musicals . Font pròpia*

Després de fer proves i entendre com funcionen les combinacions de colors amb els tres colors del led, vaig fer un programa que, mitjançant la pujada i baixada dels dos colors (ver i vermell) fes tots els colors necessaris per al treball, des del més càlid fins al més fred que podia fer la tira de leds.

Aquest programa es pot trobar al annex2 "W\_fade" i, com s'hi pot apreciar allà, fer aquests colors amb les llums no és només anar baixant el nombre d'un led i pujant l'altre, per tal que pugui fer una





gamma més amplia de colors, s'arriben a fer fins a 1020 colors diferents.

Com per transcriure una cançó de la partitura als valors només utilitzava dues octaves, 1020 colors eren masses, ho vaig reduir a 16 colors.

Per fer els 16 colors no podia anar baixant els valors un per un, ja que necessitava saber els nombres exactes de cada posició, per això vaig fer dos taules (una pel led vermell i l'altre pel verd) amb 16 posicions cadascuna, de manera que al programa hi posis un valor de l'1 al 16 i depenent de quin nombre hi hagi a la posició d'aquest valor hi representi aquesta "intensitat lluminosa" de cadascun dels dos colors.

Tot i que dues octaves són 15 notes, el fet de reduir-ho a 16 colors va ser per dos motius:

- El primer és per facilitats de càlcul, ja que vaig dividir 16 entre dos, per obtenir totes les intensitats d'un sol color, i aquest nombre dividint el valor màxim de 255, de manera que si vas sumant aquest resultat consecutivament, arribes a aquest valor de màxim temps entre parpellejos del led.
- Tot i que una escala musical té 15 notes, el segon motiu de posar-hi 16 valors va ser perquè així un dels colors és el blanc, que l'utilitza per quan no sona l'instrument principal però si els altres (per seguir representant la intensitat general de la cançó). Per a crear aquest últim valor de (0 al verd i 0 al



vermell) vaig eliminar el color més càlid, ja que era molt semblant al seu consecutiu.

Combinant els valors de les dues taules (una pel led verd i l'altre pel vermell), faria tot l'espectre de colors utilitzats. Definint l'altura del so amb un nombre que correspon a la posició a la taula.

Un cop aconseguides les dues taules va ser qüestió de transcriure la cançó que volgués representar en forma de nombres, i escriure-la en una taula de valors del programa, fent que aquesta taula anés executant cada nombre consecutivament i amb el seu ordre. Podeu trobar les taules a l'annex 2 "W\_array"

A l'Anex 1 s'hi pot trobar el programa o aplicació final, que utilitza el robot per funcionar. En aquest cas, com no es un valor que es pugui analitzar al moment (juntament amb el ritme), forma part de la cançó "Fake tales of an Francisco" del grup "Artic Monkeys".



## 4.3. ACOBLAMENT DE PROGRAMES

---

L'acoblament dels programes és el procés de fusió dels tres softwares que formen el programari total del robot, per tal que siguin executats tots al mateix temps.

- W\_rhythm (representa el ritme de la cançó)
- W\_intensity (representa el volum de la musica)
- W\_pitch) (representa l'altura)

Tot i que aquest procés és el pas final, no s'ha fet a l'acabament del treball, ja que un cop acoblat han seguit havent-hi modificacions en aquests programes per tal de millorar-los.

A les primeres versions del robot, aquest estava pensat per portar només una placa base que fes els tres programes simultàniament però, alhora de fer l'acoblament em vaig adonar que no es podia amb una sola placa "Arduino Uno" ja que aquesta no és multi funció.

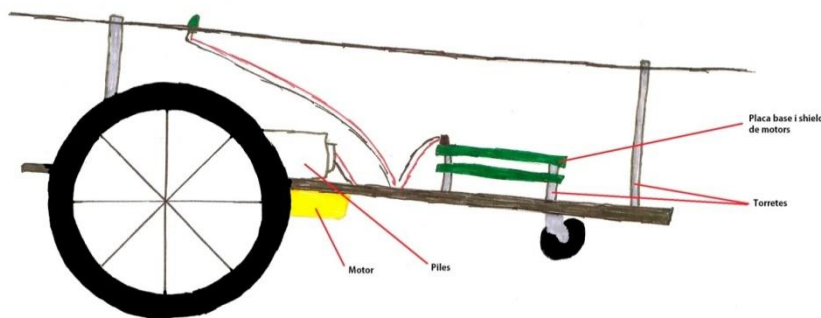
Vaig provar diferents sistemes per tal que, aprofitant la velocitat de 16MHz del rellotge, fes consecutivament un procés de cada programa i que amb aquesta velocitat tan elevada no fos perceptible per l'ull que realment no és un procés simultani, però cap de les possibles solucions que havia ideat va ser d'èxit.

Estava buscant una solució a aquest problema mentre vaig fer l'entrevista a David Cuartielles, a qui li vaig preguntar una possible solució. Em va respondre dient que hi ha una placa que es diu "Intel

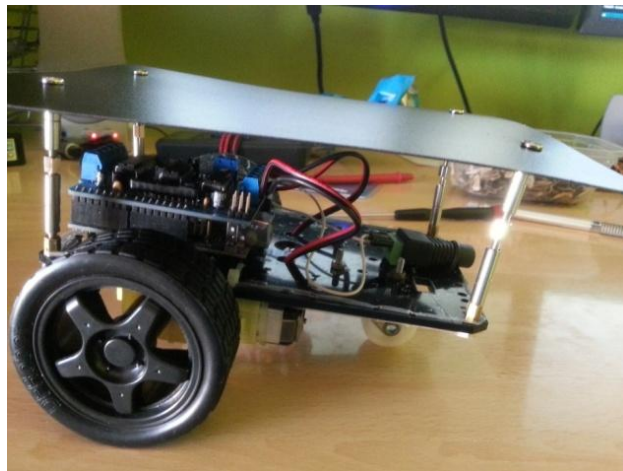


Galileo" que permet executar diferents processos d'Arduino paral·lelament, però que el més aconsellable és posar 3 plaques independents.

Amb aquesta informació vaig fer el que és la versió definitiva de l'acoblament dels programes, utilitzant tres plaques iguals, designo una (concretament, la que s'encarrega d'analitzar el so) com a iniciadora de les altres dues plaques i també com iniciadora de la font de so. Mitjançant un interruptor connectat a aquesta placa iniciadora de tots els processos, activo o desactivo tota la representació i la música.



*Primer esbós i versió provisional del robot. Font pròpia*





## 4.4. MUNTATGE I EVOLUCIÓ DEL ROBOT

---

Des de la idea inicial del treball, la forma i funcions del robot han estat en constant canvi, adaptant-se als problemes que sorgien i a la facilitat de comprensió de la representació portada a terme.

Els canvis més radicals en la forma del robot es van patir en el procés ideació d'aquest i no es van arribar a construir. Totes tenen alguns trets comuns que fins i tot en la versió definitiva no han canviat. Les primeres versions, que no es van arribar a crear, eren:

1. Una màquina estàtica amb una obertura a l'interior, al mig de la qual hi haguessin uns aerògrafs de color blau, magenta i groc (colors primaris de la pintura). Un paper llarg hauria anat passant per l'obertura empès per una sèrie de rodes situades a l'obertura (quelcom semblant a un sismògraf). La dificultat de crear una estructura tan gran va declinar el projecte.
2. Un robot mòbil amb un conjunt de tres aerògrafs a la part posterior, que anés movent-se fent moviments en forma de ziga-zaga i l'amplada del rastre de pintura marcaria la intensitat sonora. L'autòmat s'hauria anat movent per sobre d'un paper de grans dimensions. Va ser rebutjat per la dificultat que suposa controlar els aerògrafs mecànicament.

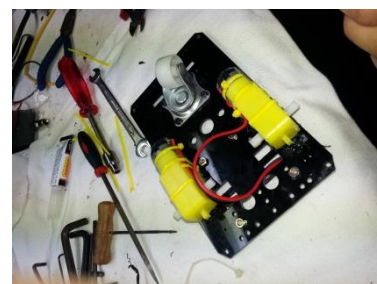


3. En el mateix model de robot anterior, substitució dels aerògrafs per retoladors dels tres colors bàsics de la pintura, mitjançant motors s'anirien pujant o baixant els retoladors de manera que com més baixés, més força faria i per tant un rastre més ample. Va ser rebutjat per la complexitat de controlar els motors a la mesura que volgués i per la possible desviació del robot alhora de prémer els retoladors contra el full.
4. Substitució dels retoladors per llums a la part superior suposant la creació d'una estructura que les subjectés i evités reflexos en els components interns.

Finalment, el robot ha sigut creat amb l'última de les modificacions tot i que més tard, com ja he explicat en cada apartat, han sorgit alguns canvis extres de determinades parts del robot.

Tots els elements del robot estan subjectats al xassís al qual se li ha anat acoblant les diferents parts a mida que es donaven per definitives. La consecució va ser la següent:

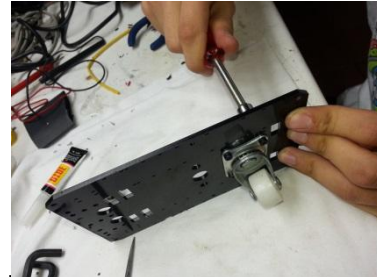
- Motors: Els motors van ser fixats a la base amb goma calenta per tal que no es pogués moure ni rotar, a més estan reforçats amb dues brides per evitar que moviments violents desenganxin els motors del xassís. Les llantes amb les rodes van encaixades a l'eix del motor.



*Fixament de motors. Font pròpia*



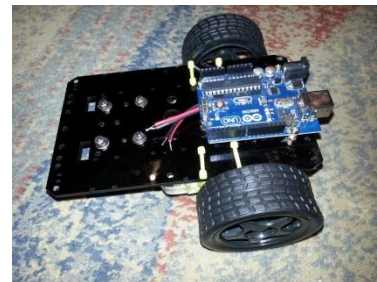
- Roda boja: Va ser fixada a la base mitjançant uns cargols, té quatre punts de subjecció que encaixen amb els forats que ja venien practicats. S'hi van incorporar unes volanderes comunes i unes "glober" per evitar que es descargoli amb les vibracions, a sobre d'aquestes sobre unes femelles estàndard per fixar tot el conjunt.



Collat roda boja. Font pròpia

- Bateria: La bateria va ser fixada també amb goma calenta i recoberta amb dues brides per donar més subjecció. El paquet obstaculitza el gir de 360 graus de la roda guia, però no afecta el moviment del robot, ja que la roda guia només necessita fer la mitja volta perquè el robot funciona en un sol sentit.

- Cables: Tant els cables dels motors com els de la bateria (ambdós elements a la part inferior del robot) han sigut fixats amb goma calenta i passant pel lateral d'altres elements (especialment els motors) per tal que no es puguin enganxar amb algun obstacle del terra i es trenquin o dessoldin. Mitjançant dos forats del xassís els cables passen d'una banda a l'altre, ja que s'han de fixar a les diferents plaques per ser utilitzats.



Cables dels motors. Font pròpia





- **Plaques:** La primera placa base està fixada a la base mitjançant quatre torretes de suport de plaques, a la base hi vaig haver que practicar dos forats més, ja que no encaixaven en els que venien pròpiament a la base.



*Collat de la placa base. Font pròpia*

Les altres dues plaques "Arduino Uno" estan fixades a la primera, com són el mateix model els forats són al mateix lloc i encaixen.

La placa de motors està fixada a la placa base que la controla per mitjà dels pins de comunicació, no cal que tingui més subjecció, ja que hi ha molts pins i no és possible moure-la.



*Unió de "shield" de motors i placa base. Font pròpia*

Finalment, la placa controladora de llums (de creació pròpia) i la placa de relé són adherides amb goma calenta al lateral de la placa base sense molestar el moviment de les rodes. El mateix circuit de control de llums incorpora la seva petita font d'energia, fixada a la part posterior del robot.

- **Pilars:** Els pilars de subjecció de la coberta, tot i que no és fixada fins al final, ja es deixen collats a la base. Aquests pilars són una conjunt de torretes de plaques unides entre si a dues altures diferents.

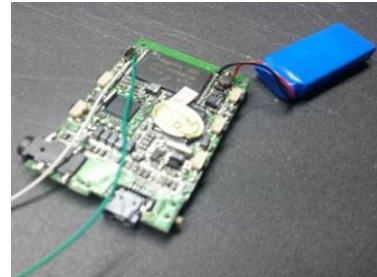


*Pilars. Font pròpia*





- Font de so: Igual que les bateries i els motors, el reproductor de música també és fixat amb brides al xassís però, a diferència d'aquests, no incorpora goma calenta entre l'aparell i la base, ja que no afecta en el funcionament si té un lleuger moviment.



*Font de so. Font pròpia*



## 5. CONCLUSIONS

---

La primera vegada que vaig pensar en la idea de representar música visualment amb un robot em va semblar difícil de desenvolupar i fins i tot, potser difícil de transmetre i fer-la entendre a altres persones.

Però a mida que anava desenvolupant el treball i els avenços que feia amb el robot, em feien veure que el robot podia arribar-se a utilitzar com una eina d'educació musical, ja que amb petites accions, podia fer entendre a gent que no sabia res sobre les característiques de la música alguns d'aquests aspectes que diferencien una cançó d'una altra.

Com va dir David Cuartielles, el cofundador d'Arduino a la videoconferència que vàrem fer: "La idea debe ser más grande que el prototipo". Aquesta frase em va inspirar molt i em va animar a trobar noves utilitats per al robot si altres persones l'utilitzessin per a diferents fins. Altres possibles utilitats que podria tenir el meu treball, fora de l'àmbit tecnològic, podrien ser: una eina de formació bàsica per a gent no entesa en la música, un mitjà per a la investigació sobre les sensacions provocades per la representació visual d'una cançó o fins i tot com una nova forma d'art, tal com fa Neil Harbisson amb el seu "Eyeborg".

Gràcies a les tasques de recerca que he fet per a desenvolupar aquest treball, he tingut l'oportunitat no només de conèixer persones reconegudes professionalment en el seu camp de treball sinó que



també m'han donat l'oportunitat de poder intercanviar opinions, de consultar dubtes i fins i tot, alguna vegada he pogut trobar la solució a algun problema gràcies als seus consells.

Algunes d'aquestes persones han estat: el Doctor Andreu Veà, reconegut escriptor i historiador de la història d'Internet que em va donar suport i va proporcionar-me alguns dels contactes més importants; David Bassetti, creador amb Arduino de la impressora 3D més gran del món; Miguel Vidal, líder de Java a Sun Microsistems i dedicat a la telemetria de la Formula 1 i que es va posar en contacte personalment amb mi per obtenir informació sobre el robot; David Cuartielles, cocreador de la plataforma Arduino i que gràcies a una videoconferència que vaig tenir amb ell, em va donar l'oportunitat de preguntar-li alguns dels meus dubtes amb el robot i em va proposar possibles solucions; Nacho Sánchez, emprenedor d'èxit que dirigeix l'empresa "InqBarna" i que es dedica al desenvolupament d'aplicacions de música per a mòbils i gràcies al qual vaig poder resoldre els dubtes que se'm plantejaven sobre els paràmetres de la música; Doctor Miquel Badia, reconegut oftalmòleg de Mataró que, juntament amb el seu equip, em van ajudar a obtenir informació relacionada amb l'emascament visual temporal.

Gràcies a la cerca i anàlisi de la informació que obtenia i posteriorment, a l'anàlisi i la investigació que personalment feia, he tingut l'oportunitat d'adquirir coneixements sobre paràmetres del so i teoria musical. Coneixements que poden ser molt útils per al futur perquè una de les opcions professionals que em plantejo estarà vinculat a les tecnologies del so.



Tot i això, he de reconèixer que en moments puntuals he pensat que m'havia ficat en un projecte massa gran. Aleshores en aquells moments, parava i reflexionava, anava per parts i assolint els problemes en petites dosis aconseguia solucionar el repte sencer; però finalment considero que he aconseguit el que tenia al cap el moment que vaig plantejar la idea inicial. He dedicat molt de temps i forces però en general m'ha proporcionat hores d'oci treballant en mons que m'agraden: l'electrònica i la música.

M'agradaria tornar a donar les gràcies a totes les persones que m'han ajudat i que m'han donat suport; veure que hi ha persones que s'interessen pel que estàs fent, que t'animen en el projecte escollit i en la feina que estàs fent és molt gratificant. Juntament amb el suport, veure que era capaç de sortir-me'n dels problemes que s'anaven presentant, m'han fet defensar la meva idea fins al final. La cita "Los mayores desafíos engendran los mejores resultados" és un bon reflex de què ha significat aquest treball per a mi.

Finalment concloc aquest treball amb la satisfacció d'haver assolit l'objectiu de crear un robot que pogués representar la música visualment i amb l'esperança que això sigui útil per a altres investigacions.



## 6. BIBLIOGRAFIA

---

La bibliografia que he utilitzat per a desenvolupar el treball i que recomano a qualsevol persona interessada en el tema és la següent:

### Llibres:

- CUENCA, Ignasi; GÓMEZ, Eduard. *Manual Técnico de Sonido*, 7ª ed. Madrid: Ediciones Paraninfo, 2011.
- *Equipos de Sonido*. Madrid: Ed. McGraw-Hill, 2006.
- McCORMICK, Tim; RUMSEY, Francis. *Sonido y grabación*. Barcelona: Omega, 2008.
- PARKER, Steve. *Sonido: proyectos y experimentos con música y ondas sonoras*. Barcelona: Parramon, 2008.
- POHLMANN, KEN. *Principios de audio digital*. Madrid: McGraw-Hill, cop. 2002.
- TÀPIAS, Montserrat [et al.]. *Òptica i optometria*. Barcelona: Servei de Llengües i Terminologia de la UPC, cop. 2002.



Webs:

- [www.arduino.cc](http://www.arduino.cc)
- [www.ca.wikipedia.org](http://www.ca.wikipedia.org)
- [www.cit.iec.cat](http://www.cit.iec.cat)
- [www.dezeen.com/2013/11/20/interview-with-human-cyborg-neil-harbisson](http://www.dezeen.com/2013/11/20/interview-with-human-cyborg-neil-harbisson)
- [www.diccionaris.cat](http://www.diccionaris.cat)
- [www.ebay.com](http://www.ebay.com)
- [www.eprints.ucm.es](http://www.eprints.ucm.es)
- [www.inqbarna.com](http://www.inqbarna.com)
- [www.ladyada.net](http://www.ladyada.net)
- [www.learn.adafruit.com](http://www.learn.adafruit.com)
- [www.marlonj.com](http://www.marlonj.com)
- [www.musicouniversal.blogspot.com.es](http://www.musicouniversal.blogspot.com.es)
- [www.ojocientifico.com](http://www.ojocientifico.com)
- [www.opensource.com](http://www.opensource.com)
- [www.processing.org](http://www.processing.org)
- [www.softonic.com](http://www.softonic.com)
- [www.songsterr.com](http://www.songsterr.com)
- [www.stuff.tv/brit-neil-harbisson-officially-world-s-first-cyborg/news](http://www.stuff.tv/brit-neil-harbisson-officially-world-s-first-cyborg/news)
- [www.tuxguitar.softonic.com](http://www.tuxguitar.softonic.com)
- [www.vimeo.com](http://www.vimeo.com)
- [www.wikipedia.org](http://www.wikipedia.org)
- [www.wiring.org.co](http://www.wiring.org.co)



## 7. ANNEXOS

---

### 7.1. ANNEX 1 (Programes finals)

---

#### 7.1.1. W\_RHYTHM

---

```
#include <AFMotor.h>

AF_DCMotor motorE(3);
AF_DCMotor motorD(4);
float mspb;
float bpm=127;
int start=2;
int vrun;

void setup(){
  mspb=((1/(bpm/60))*1000);
  pinMode(start, INPUT);

  motorE.setSpeed(120);
  motorE.run(RELEASE);

  motorD.setSpeed(130);
  motorD.run(RELEASE);
}

void loop(){
  vrun=digitalRead(start);
  if (vrun==1){
    motorE.run(RELEASE);
    motorD.run(FORWARD);
    delay(mspb);

    motorE.run(RELEASE);
    motorD.run(RELEASE);

    motorD.run(RELEASE);
    motorE.run(FORWARD);
    delay(mspb);

    motorE.run(RELEASE);
    motorD.run(RELEASE);
  }
}
```



## 7.1.2. W\_INTENSITY

---

```
int channel = A0;
int led = 9;
int music = 2;
int start = 4;
int button = 7;
int value, ans, buttonstat, vrun;

void setup() {
  pinMode(led, OUTPUT);
  pinMode(music, OUTPUT);
  pinMode(start, INPUT);
  pinMode(button, INPUT);
  pinMode(channel, INPUT);
}

void loop() {
  buttonstat=digitalRead(button);
  if(buttonstat=1){
    if (vrun==1){
      digitalWrite(start,HIGH);
      vrun=0;
    }

    value=analogRead(channel);
    value=value/5;
    value=value*17;

    delay(1);

    if(value==0){value=1;};
    analogWrite(led, value);
  }
  else {
    digitalWrite(start,LOW);
    vrun=1;
  }
}
```





### 7.1.3. W\_PITCH

---

```
int red[16]={
    0, 0, 0, 0, 0, 0, 0, 0, 0, 63, 96, 127, 159, 191, 223, 255,
};
```

```
int green[16]={
    0, 223, 191, 159, 127, 96, 63, 0, 0, 0, 0, 0, 0, 0, 0, 0,
};
```

```
int ft[300]={
    3,5,3,5,6,5,
    3,5,3,5,6,5,
    3,5,3,5,6,5,
    3,5,6,7,9,6,5,
```

```
    3,5,3,5,6,5,
    3,5,3,5,6,5,
    3,5,3,5,6,5,
    3,5,6,7,9,6,5,
```

```
    3,5,3,5,6,5,
    3,5,3,5,6,5,
    3,5,3,5,6,5,
    3,5,6,7,9,6,5,
```

```
    3,5,3,5,6,5,
    3,5,3,5,6,5,
    3,5,3,5,6,5,
    3,5,6,7,9,6,5,
```

```
    3,5,3,5,6,5,
    3,5,3,5,6,5,
    3,5,3,5,6,5,
    3,5,6,7,9,6,5,
    5,1,1,1,
```

```
    1,1,1,1, 1,1,1,1,
    1,1,1,1, 1,1,1,1,
    1,1,1,1, 1,1,1,1,
    1,1,1,1,
```

```
    3,5,3,5,6,5,
    3,5,3,5,6,5,
    3,5,3,5,6,5,
    3,5,6,7,9,6,5,
```



3,5,3,5,6,5,  
 3,5,3,5,6,5,  
 3,5,3,5,6,5,  
 3,5,6,7,9,6,5,

3,5,3,5,6,5,  
 3,5,3,5,6,5,  
 3,5,3,5,6,5,  
 3,5,6,7,9,6,5,

3,5,3,5,6,5,  
 3,5,3,5,6,5,  
 3,5,3,5,6,5,  
 3,5,6,7,9,6,5,  
 5,1,1,1,

1,1,1,1, 1,1,1,1,  
 1,1,1,1, 1,1,1,1,  
 1,1,1,1, 1,1,1,1,  
 1,1,1,1,  
 };

```
int x,g,r;
int led1=10;
int led2=11;
float mspb;
float bpm=127;
int start=2;
int vrun;
```

```
void setup(){
    mspb=((1/(bpm/60))*1000);
    pinMode(start, INPUT);
```

```
    pinMode(led1,OUTPUT);
    pinMode(led2,OUTPUT);
}
```

```
void loop(){
    vrun=digitalRead(start);
    if (vrun==1){
        g=green[ft[x]];
        r=red[ft[x]];
        analogWrite(led1,r);
        analogWrite(led2,g);
        if(x<146){x++;}else{x=0;};
        delay(mspb);
    }
}
```



## 7.2. ANNEX 2 (Versions anteriors)

### 7.2.1. W\_LECTOR

```

lector_valors_analogic
int x; //assignació (en integer) de la variable on anirà el valor de l'analògic

void setup(){
  Serial.begin(9600); //iniciació del monitor sèrie, on es mostraran els valors
}

void loop(){
  x=analogRead(A0); //es guarden temporalment els valors obtinguts a l'analògic en la variable
  Serial.println(x); //es mostra el valor de la variable al monitor sèrie
  delay(1); //s'hi posa una espera molt petita per a poder llegir amb claredat els valors
}

```

### 7.2.2. W\_VUMETRE

```

vumetre_Arduino
int led[10]=(1,2,3,4,5,6,7,8,9,10); //es declaren els 10 leds utilitzats a les 10 sortides
int valor,x; //es declaren dos variables, una pels valors i l'altre com a comptador

void setup(){
  for (x=0;x<10;x++){
    pinMode(led[x],OUTPUT); //definim tots els pins dels leds com a sortides
  }
  Serial.begin(9600); //s'inicia el monitor sèrie
}

void loop(){
  valor=analogRead(A0); //llegim el valor de l'entrada analògica i el guardem a la variable
  Serial.println(valor); //mostrem aquest valor en el monitor sèrie
  valor=valor/100; //el dividim entre 100 per tenir-ho en una escala de 0 a 10

  if(valor==0){ //si el valor és 0, no sona música, per tant:
    for (x=0;x<10;x++){
      digitalWrite(led[x],LOW); //s'apaguen tots els leds
    }
  }
  else{ //en cas que no sigui 0, sona música, llavors:
    for (x=0;x<valor;x++){
      digitalWrite(led[x],HIGH); //s'encenen tots els leds des del 0 fins al valor de la variable
    }
    for (x=x;x<10;x++){
      digitalWrite(led[x],LOW); //la resta de leds s'apaguen
    }
  }
}
}

```



### 7.2.3. W\_LED

```

ritme_led
int led=13; //es declara el led de sortida
float bpm=128; //es declara el bpm i s'hi indica la velocitat
float mspb; //es declaren els mspb, la variable float és per guardar decimals

void setup() {
  mspb= ((1/(bpm/60))*1000); //es calculen els mspb a partir del bpm
  pinMode(led,OUTPUT); //definim el led com una sortida
}

void loop(){
  digitalWrite(led,HIGH); //s'encen el led
  delay(1); //es manté un mil·lisegon encés, suficient per veure'l sense afectar al ritme
  digitalWrite(led, LOW); //s'apaga el led
  delay(mspb); //es retarda l'encesca del led fins al següent batec
}

```

### 7.2.4. W\_FADE

```

W_fade
int led1=10; //es designa el led verd com led1
int led2=11; //es designa el led vermell com led2
int r,g; //es declaren les variables on s'hi guardarà el valor de cada color

void setup() {
  pinMode(led1,OUTPUT); //es designa el led1 com una sortida de senyal
  pinMode(led2,OUTPUT); //es designa el led2 com una sortida de senyal
}

void loop() {
  while (g>0) { //el led vermell es manté i el verd augmenta la "brillantor"
    r=0;
    analogWrite(led1,g);
    analogWrite(led2,r);
    g--;
    delay(10);
  }

  while (r<255) { //el led verd es manté i el vermell disminueix la "brillantor"
    g=0;
    analogWrite(led1,g);
    analogWrite(led2,r);
    r++;
    delay(10);
  }

  while (g<255) { //el led vermell es manté i el verd disminueix la "brillantor"
    r=255;
    analogWrite(led1,g);
    analogWrite(led2,r);
    g++;
    delay(10);
  }

  while (r>0) { //el led vermell es manté i el verd augmenta la "brillantor"
    g=255;
    analogWrite(led1,g);
    analogWrite(led2,r);
    r--;
    delay(10);
  }
}

```



## 7.2.5. W\_ARRAY

```

W_array
int red[16]={                                //taula de valors del led vermell
    0,  0,  0,  0,  0,  0,  0,  0,
    0,  63,  96, 127, 159, 191, 223, 255,
};

int green[16]={                              //taula de valors del led verd
    0, 223, 191, 159, 127, 96, 63, 0,
    0,  0,  0,  0,  0,  0,  0,  0,
};

int p;                                       //variable per guardar la posició
int led1=10;                                //designació del led vermell al pin 10
int led2=11;                                //designació del led verd al pin 11

void setup(){
    pinMode(led1,OUTPUT);                   //declaració del led vermell com a sortida
    pinMode(led2,OUTPUT);                   //declaració del led verd com a sortida
}

void loop(){
    analogWrite(led1,red[p]);                //determinació del temps de "no-llum" respecte la posició
    analogWrite(led2,green[p]);              //determinació del temps de "no-llum" respecte la posició
    if(p<15){p++;}else{p=0;};               //canvi de posició abansçant linealment
    delay(500);                              //pausa per poder visualitzar els colors
}

```



