

# **Somien els robots en tocar el piano?**

Construcció d'una mà robòtica pianista

***«Podria parèixer que hem arribat al límit del que és possible aconseguir amb la tecnologia informàtica, però s'ha de tenir compte amb aquestes declaracions, ja que tendeixen a sonar estúpides en cinc anys.» - John von Neumann***

# Índex

1.Objectius.....	4
2.El robot. Introducció al funcionament.....	5
3.Arduino.....	6
3.1.ARDUINO, una mica d'història.....	6
3.2.Funcionament d'Arduino.....	7
3.3.Arduino Mega 2560.....	8
3.4.Parts de l'Arduino.....	9
4.Programació.....	10
5.Els components electrònics.....	12
5.1.Els servomotors.....	12
5.1.1.Funcionament.....	13
5.1.2.Servomotors utilitzats.....	13
5.2.El motor pas a pas.....	14
5.2.1.Tipus de motor pas a pas.....	14
5.2.2.Funcionament.....	15
5.2.3.Motor pas a pas utilitzat.....	15
5.3.Les <i>shields</i> .....	16
6.Componentes addicionals.....	17
6.1.L'alimentació.....	17
7.Circuit elèctric.....	18
7.1.Motor pas a pas.....	18
7.2.Servomotors.....	19
7.3.Leds.....	20
8.Estructura.....	21
8.1.Versions de l'estructura.....	21
8.1.1.Primera idea conceptual.....	22
8.1.2.Primer disseny físic.....	24
8.1.3.Segon i final disseny físic.....	26
9.Materials per a l'estructura.....	28
10.Muntatge.....	29
11.Pressupost.....	31
12.Programa.....	32
13.Possibles aplicacions del robot.....	38
14.Conclusions.....	39
15.Bibliografia i Webgrafia.....	40
15.1.Webgrafia.....	40
15.2.Bibliografia.....	41

# 1.Objectius

-L'objectiu d'aquest treball és, principalment, aconseguir construir manualment una mà robòtica capaç de tocar el piano.

Òbviament amb els meus coneixements tècnics el resultat final pot ser limitat. Una altra variable que limitarà el procés són els articles, eines i diferents productes que posseeixo des del començament. La meua intenció és construir el robot amb un conjunt de materials assequibles i a l'abast de qualsevol estudiant de batxillerat i utilitzar les eines proporcionades al taller de tecnologia de l'institut. Tot això farà que el valor econòmic del projecte sigui baix i, per tant, la inversió en la seva fabricació també ho sigui.

La idea principal és que el robot tingui la capacitat de tocar peces musicals senzilles (només una mà, sigui la dreta o l'esquerra). Per les limitacions comentades anteriorment em concentraré en el fet que el robot sigui capaç de tocar només les tecles blanques del piano i deixar de banda les negres (bemolls i sostinguts).

## 2.El robot. Introducció al funcionament.

El robot funciona gràcies a una placa programable *ARDUINO*. Aquesta placa envia les accions que ha de fer el robot a cinc servomotors<sup>1</sup>. Els servomotors estan en una base que es mou d'una punta a l'altra del teclat gràcies a un motor pas a pas<sup>2</sup> i tenen una estructura en forma de dit



cada un. Aquests dits són accionats gràcies al programa introduït a la placa *ARDUINO*. El motor pas a pas mou l'estructura al lloc desitjat i els dits s'accionen per tocar les diferents tecles del piano. En aquest cas el robot ha estat adaptat a un teclat de dues octaves, per tant només té la capacitat de tocar les tecles pertanyents a dues octaves<sup>3</sup>.

Es pot observar que a simple vista la idea és bastant simple. Aquesta era la intenció principal, aconseguir bons resultats amb la màxima simplicitat possible i utilitzant el mínim nombre de components electrònics possibles.

---

1 Els servomotors són petits motors, s'explicarà el seu funcionament més endavant.

2 Un altre tipus de motors, s'explicarà més detalladament més endavant.

3 Una octava és un conjunt de vuit notes. Aquesta octava és un interval en el qual la freqüència de la primera nota i la última mantenen una relació de 2:1, una és el doble que l'altra.

## 3.Arduino

### 3.1.ARDUINO, una mica d'història

El robot funciona principalment gràcies a l'*ARDUINO*, el qual és pot considerar el cervell. «Arduino» és una companyia de hardware original d'Itàlia coneguda principalment per la venda de plaques microcontroladores<sup>4</sup> de baix cost. Aquestes són utilitzades i recomanades per a principiants del món de l'electrònica gràcies a la seva senzillesa i facilitat d'ús. Una de les seves característiques principals és que tots els productes d'Arduino es basen en un hardware i un software lliure, és a dir, totes les seves especificacions, característiques i informació són d'accés públic per a tothom.

*ARDUINO* neix l'any 2005 com una necessitat d'aprenentatge a l'institut IVREA per part de Massimo Banzi. En aquests anys, adquirir una placa amb microcontroladors suposava una inversió bastant alta que molts estudiants no podien permetre's. La intenció de Massimo Banzi no era només fabricar una placa de microcontroladors barata i amb un bon suport, sinó que també volia ajudar econòmicament a



l'institut IVREA que en aquells moments estava a punt de fer fallida. La primera placa, que era molt bàsica, va ser fabricada al mateix institut.

Anys més tard David Mellis, Hernando Barragán, David Cuartielles i Tom Igoe es van unir a Massimo Banzi per continuar treballant en aquest projecte. Tots ells, provinents de diferents llocs del món són els fundadors de l' empresa. Junts van desenvolupar i millorar la primera placa de la qual es van fer dues-centes còpies.

---

4 Un microcontrolador és un circuit integrat (dispositiu electrònic que conté diferents components) que pot ser programat.

Amb aquesta primera tirada dirigida a estudiants de l'institut IVREA, es van posicionar en el món de plaques amb microcontroladors.

Al començament aquestes plaques tenien una baixa potència i encara no tenien una sortida usb, que es va implantar posteriorment i que permet connectar l'ARDUINO directament a l'ordinador. Una de les primeres característiques per què es va apostar va ser per una àmplia compatibilitat, des de un bon començament les plaques ARDUINO són funcionals en els tres sistemes operatius principals: Windows, Mac OS i Linux. Apostant per una alta compatibilitat, un sistema lliure, un baix cost i una facilitat d'ús no vista anteriorment en plaques similars es van fer un bon lloc en la indústria fins a convertir els seus productes en els més importants en l'actualitat.

El disseny de les plaques Arduino està inspirat en les plaques Wiring, creades per Hernando Barragán (també estudiant de l'institut IVREA) l'any 2003.

Actualment hi ha una gran gamma de diferents plaques que s'adapten a diferents necessitats i a diferents tipus d'usuaris, però no va ser fins el 2012 que es van presentar i introduir nous models.

### **3.2.Funcionament d'Arduino**

Com s'ha esmentat anteriorment un Arduino és una placa de hardware (concretament una placa PCB<sup>5</sup>) lliure que conté un microcontrolador i diferents «pins», als quals s'hi connectaran diferents sensors i actuadors. Depenent del model de la placa varia el nombre de «pins», el model del microcontrolador integrat i la memòria de què disposa.

---

5 De l'anglès «printed circuit board». Aquestes plaques són superfícies fabricades amb un material no conductor sobre les quals s'apeguen pistes de material conductor. Les pistes conductores connecten diferents components que es solden a les plaques. El principal avantatge de les PBC és que són circuits molt compactes i estables, per contra, una cop fabricades són difícils de modificar.

El funcionament de la placa es divideix en tres fases principals. La primera és la recollida de dades, és a dir, es recopilen les dades enviades pels diferents sensors que hi ha connectats (si n'hi ha). A continuació la placa interpretarà aquestes dades i finalment seran enviades als perifèrics de sortida, que poden ser altres plaques que tornaran a rebre la informació i interpretar-la o simples actuadors.

### 3.3.Arduino Mega 2560

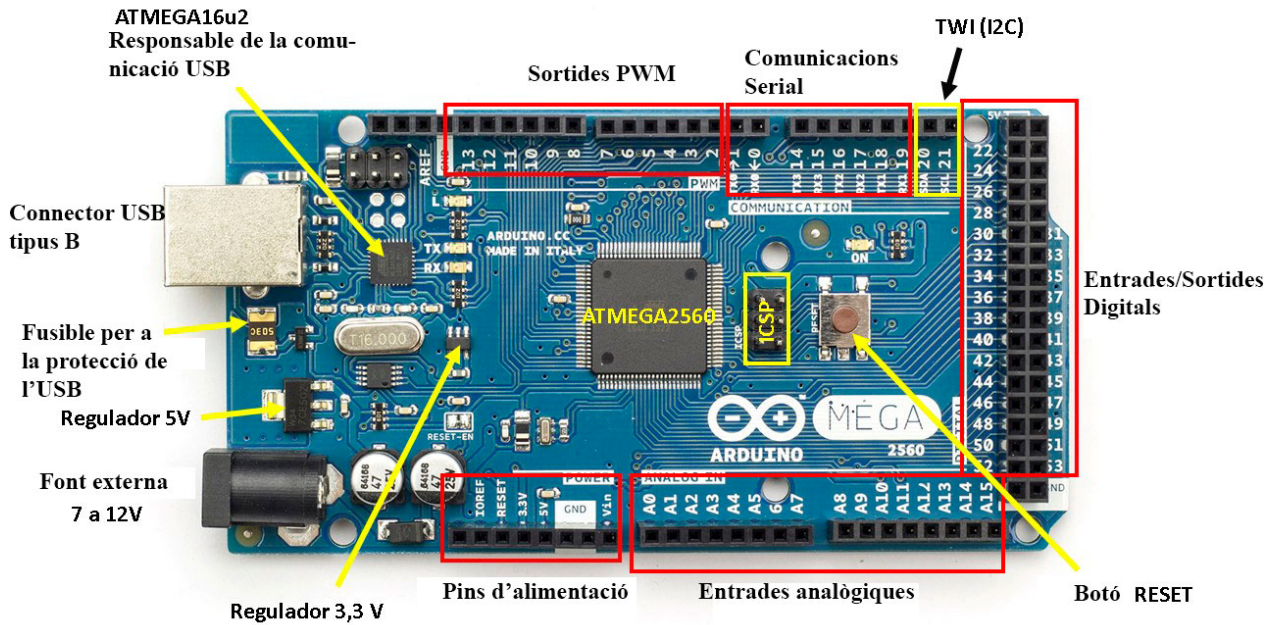
Arduino Mega és una de les moltes variants oficials que hi ha de la placa Arduino i la que he utilitzat per aquest projecte. Aquesta destaca per ser una de les plaques més potents. També es diferencia de les altres gràcies al seu gran nombre de «pins».

A continuació es pot observar una taula amb les seves característiques generals:

<b>Microcontrolador</b>	ATmega2560
<b>Voltatge operatiu</b>	5V
<b>Voltatge d'entrada</b>	7-12V
<b>Voltatge d'entrada (límits)</b>	6-20V
<b>Pins digitals</b>	54 (15 PWM)
<b>Pins analògics</b>	16
<b>Corrent màxima per cada pin</b>	40 mA
<b>Memòria Flash</b>	256 Kb
<b>SRAM</b>	8 Kb
<b>EEPROM</b>	4Kb
<b>Velocitat de rellotge</b>	16 MHz



### 3.4.Parts de l'Arduino



## 4.Programació

Per tal que el nostre projecte amb Arduino funcioni és necessari donar unes ordres a la placa, això ho aconseguim fent un programa. Aquest programa consta bàsicament de les ordres i condicions que nosaltres desitgem. El problema és que aquest programa que nosaltres hem escrit anteriorment, la placa no l'entén, aquí és on entra la participació del compilador. Aquest s'encarrega de transformar el programa a llenguatge màquina, és a dir, en binari.

El llenguatge de programació que utilitza Arduino és C i C++. Aquests llenguatges són de nivell mig i són els segons més utilitzat en el món de la programació, els podem trobar tant en videojocs com en editors de text.

C va ser creat per Dennis Ritchie a principis de la dècada dels 70 com una evolució del llenguatge B. A principis dels anys 80 es van començar a afegir noves característiques a aquest llenguatge fins que l'any 1985 es va consolidar com a C++. La gran importància i fama d'aquests dos llenguatges ve donada per la seva potència, lleugeresa i flexibilitat ja que posseeixen un ampli ecosistema de llibreries que li afegeixen funcionalitats.



Anteriorment s'ha mencionat que C i C++ són llenguatges de nivell mitjà, però què vol dir això? En contra del que pot semblar el qualificatiu no es refereix a la dificultat dels llenguatges ja que aquesta és més aviat alta. Definim un llenguatge de nivell alt com aquell que s'assembla molt a la nostra forma de raonar aïllant al programador dels detalls tècnics. Això fa que aquests tipus de llenguatges siguin poc eficients. Per altra banda, els llenguatges de baix nivell controlen directament la màquina física,

podent així obtenir una eficiència màxima. Per tant, amb un llenguatge d'alt nivell és senzill, uniforme i portable.

Un dels principals problemes dels llenguatges d'alt nivell és que el programa s'ha de traduir a llenguatge màquina. Això s'aconsegueix tal com s'ha dit abans mitjançant el compilador. Aquest, que és un programa, accepta com a dades d'entrada un programa d'alt nivell i genera un programa en llenguatge màquina.

Però C i C++ no són de nivell alt ni baix, sinó mitjà, de manera que es beneficien dels avantatges dels dos tipus: són potents, eficients, proporcionen un control total del que passa a l'interior de l'ordinador i permeten una àmplia llibertat d'organització del treball.

Com a conseqüència són difícils d'aprendre i sense una gran experiència i disciplina és difícil mantenir el control del programa.

Com que Arduino és una plataforma molt estesa en el món de l'educació s'han creat alternatives a l'hora de programar. Algunes de les més esteses i utilitzades són les que consisteixen en una programació en blocs. Aquestes ofereixen una alternativa més senzilla i fàcil d'aprendre. Un dels millors exemples de plataformes de programació en blocs és Bitbloq («<http://bitbloq.bq.com/>»). Aquesta plataforma està feta i controlada per la marca espanyola Bq, que també ofereix plaques similars a Arduino.

Per al meu projecte he optat per una programació directa amb l'IDE<sup>6</sup> d'Arduino ja que m'ofereix una total llibertat a l'hora de desenvolupar els projectes.

---

<sup>6</sup> Un IDE («Integrated development environment») és una eina informàtica per al desenvolupament de programes de manera còmoda i ràpida. Es podria dir que un IDE és un editor de text dissenyat per a programar.

## 5.Els components electrònics

Per al desenvolupament i la construcció he utilitzat diferents components a banda de la placa d'Arduino. Els dits s'accionen gràcies a cinc servomotors, l'estructura dels dits es mou amb un motor pas a pas i el funcionament d'aquest components s'ha millorat amb dos *shields*.

### 5.1.Els servomotors

Els servomotors, també anomenats servos, són servomecanismes similars als motors de corrent continu que tenen la capacitat de situar-se en qualsevol posició dins del seu rang d'operació. Un dels principals avantatges d'aquests motors és que poden arribar a ser molt assequibles i tenen una mida bastant reduïda, cosa que els converteix en perfectes candidats per a projectes de robòtica.



Un servomotor està principalment format per un motor de corrent continu, el circuit de control, els engranatges i tres cables de connexió. Hi ha un cable vermell (alimentació a 5V), un de negre (GND), i un que pot variar de color (encarregat de rebre el senyal de control). També hi trobem un potenciòmetre que ens indica en quina posició es troba el motor.

Normalment els servomotors tenen uns 180 graus d'operació, tot i que hi ha alguns models que arriben fins als 360. En el robot jo he utilitzat servos d'uns 120 graus operacionals, ja que són els que més s'adapten a les meves necessitats.

### 5.1.1. Funcionament

Els servomotors funcionen per PWM<sup>7</sup>. Aquesta tècnica s'utilitza per controlar la direcció i posició el motor. Depenent de la durada del senyal que rebí el circuit de control, el motor es mourà en un sentit o un altre i determinarà la seva posició.

### 5.1.2. Servomotors utilitzats

Al principi vaig intentar fer proves amb servos que tenien els engranatges de plàstic, concretament el model *SG90 Tower Pro*. En les primeres proves ja em vaig adonar que hauria d'optar per servos diferents, ja que aquest no m'oferia les característiques que necessitava. Amb un parell d'1,2 Kg/cm era incapaç d'accionar la tecla del piano i es quedava encallat.

Després de les diferents proves amb aquest petit servomotor vaig optar pel model *MG996r* de la mateixa marca. Aquest, tot i ser més gran, m'oferia un parell d'uns 11 Kg/cm. A més a més el circuit d'engranatges d'aquest model és metàl·lic, cosa que el fa més resistent. Característiques del servomotor *MG996r Tower Pro*:

**Voltatge d'operació:** de 4,8V fins a 7,2 V

**Velocitat d'operació:** 0,17 s/60° (4,8V) i 0,14 s/60° (6v)

**Parell:** 9,4 Kg/cm (4,8V) i 11 Kg/cm (6V)

**Angle de rotació:** 120° aproximadament

**Pes:** 55 g

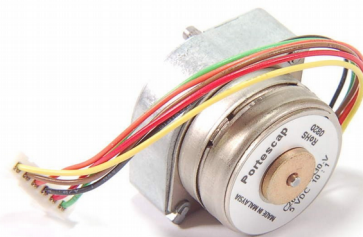
**Dimensions:** 40,7x19,7x42,9 mm

---

<sup>7</sup> De l'anglès *pulse-width modulation*. La PWM d'un senyal és una tècnica que modifica el cicle de treball d'un senyal periòdic.

## 5.2.El motor pas a pas

Tal com s'ha dit anteriorment l'estructura dels cinc servos es mou gràcies a un motor pas a pas. Aquest està instal·lat en unes guies d'una vella màquina d'escriure elèctrica (el motor pas a pas utilitzat també formava part de la màquina d'escriure).



Un motor pas a pas està format per un seguit de bobines. Per tal que giri és necessari que les bobines s'alimentin d'una manera determinada. Aquests motors tenen la característica que no giren contínuament, sinó amb petits passos tal com indica el seu nom. Una altra característica d'aquests motors és que quan no giren però segueixen rebent corrent són capaços de mantenir-se quietos gràcies al gran parell que ofereixen les bobines.

### 5.2.1.Tipus de motor pas a pas

**Unipolars:** tenen quatre bobines internes (en realitat són dues, cada una de les quals està dividida per una connexió central comuna per on rep l'alimentació) rodejant l'eix central. Del motor surten sis cables: quatre corresponen als extrems de les bobines i els altres dos corresponen a les connexions centrals. Pot ser que algun motor unipolar només tingui cinc cables. La millor manera de saber quin cable és quin és consultar la taula informativa de cada motor tot i que hi ha un mètode manual (més lent i complicat).

**Bipolars:** tenen dues bobines internes independents (que no estan dividides per cap connexió central) rodejant l'eix central . Del motor surten quatre cables corresponents als extrems de cada bobina. Aquests motors tenen més parell que els unipolars, concretament al voltant d'un 30% més, però també són més complicats d'utilitzar.

## **5.2.2.Funcionament**

El control d'un motor pas a pas es realitza aplicant un voltatge concret a les bobines amb un determinat patró repetitiu. Depenent de si reben tensió o no les bobines van creant camps magnètics que atreuen o repel·leixen els imants de l'eix de manera que aquest gira. D'aquesta manera, depenent del patró de corrent podem controlar el moviment del motor de manera detallada.

## **5.2.3.Motor pas a pas utilitzat**

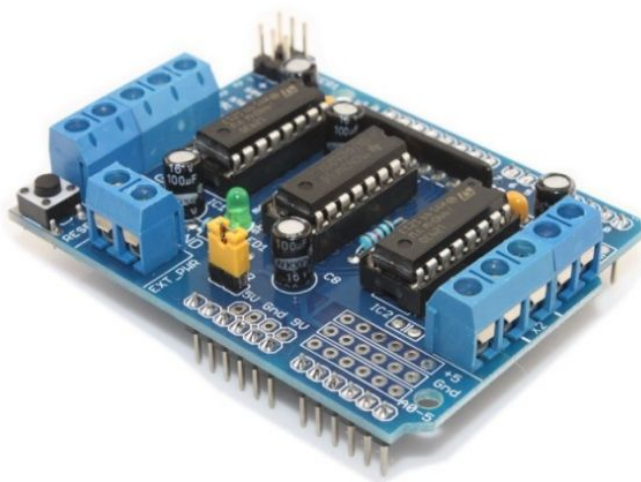
El motor que he utilitzat per controlar el moviment de l'estructura és un motor unipolar que vaig poder extreure d'una màquina d'escriure elèctrica que ja no funcionava. L'eix en el qual es movia el motor també el vaig aprofitar, ja que s'adaptava perfectament a l'espai del que disposava sobre el piano.

Com que el motor és reutilitzat i també una mica vell no dispo del nom de la marca ni el model.

### 5.3.Les *shields*

Per a millorar el rendiment del motor pas a pas he utilitzat una *shield*. Les *shields* són circuits modulars que s'afegeixen i es connecten a la placa Arduino per a donar-li més funcionalitats i ampliar les capacitats i possibilitats d'un projecte. Aquestes plaques es connecten a la placa controladora pels «pins» digitals o analògics.

Per al motor pas a pas he utilitzat la placa *L293D*. Gràcies a aquesta placa he pogut aplicar un voltatge més gran al motor de manera que he aconseguit una velocitat més gran i òptima per al projecte. Aquesta placa està formada per dos circuits integrats *L293D* proporcionant capacitat per al control de dos motors. Aquesta placa permet el control de la velocitat i direcció de gir dels dos motors, a part de també donar suport a dos servos. La *shield* es connecta directament a l'Arduino posant-la a sobre dels pins que corresponen al model de l'Arduino UNO.





## 6. Components addicionals

### 6.1. L'alimentació

Per alimentar el robot he utilitzar diferent vies per aconseguir els millors resultats i el millor rendiment possible. Per tant, he optat per alimentar les diferents parts (Arduino, servomotors i motor pas a pas) de manera independent.

La placa controladora funciona gràcies a una pila de 9V.

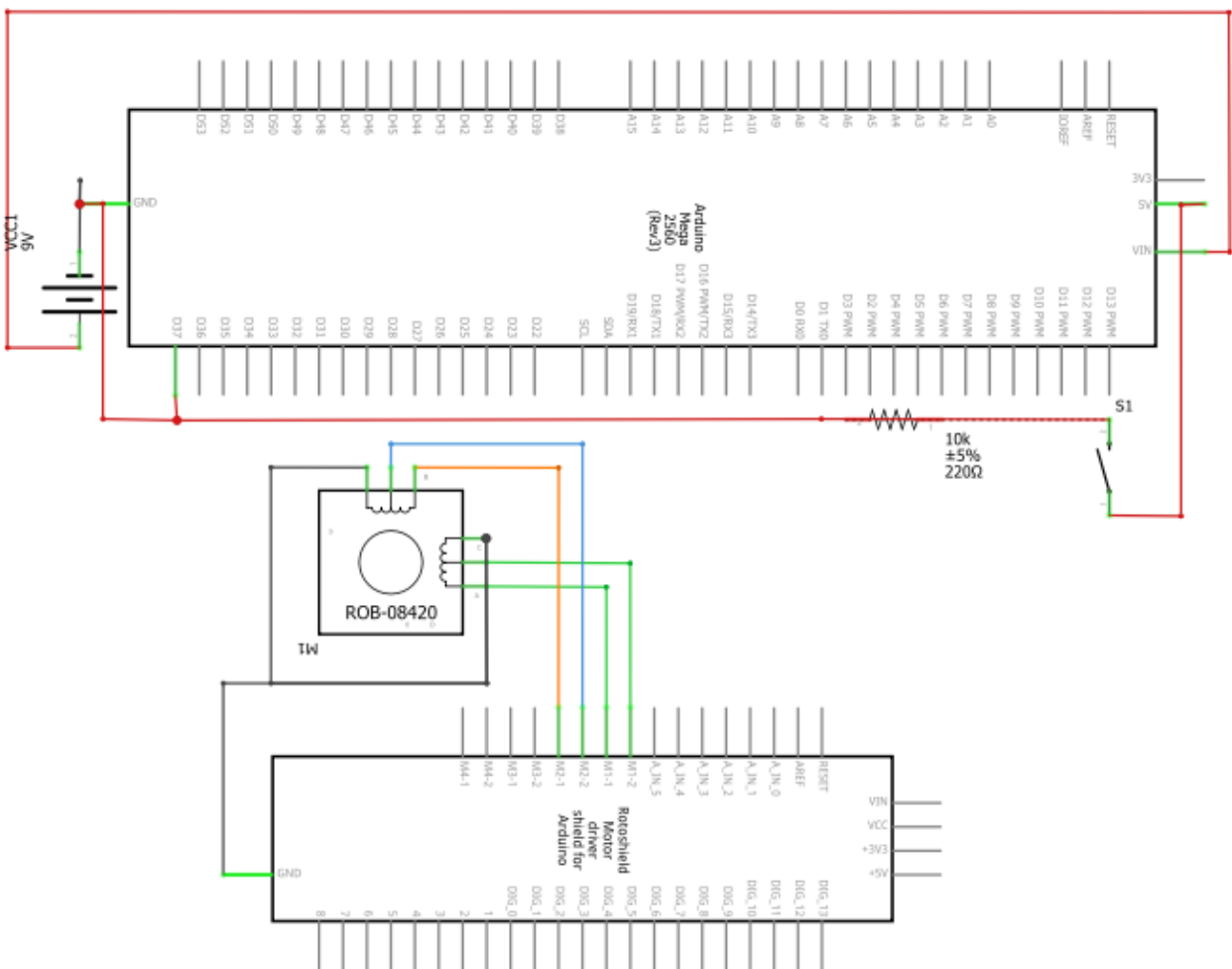
El motor pas a pas s'alimenta directament al corrent elèctric. Per aconseguir això he utilitzat un carregador que proporciona 2,1 A d'intensitat amb 19V.

Finalment una bateria de tipus LiPo alimenta els servos. Aquesta ofereix 7,4V i 2200 mAh.

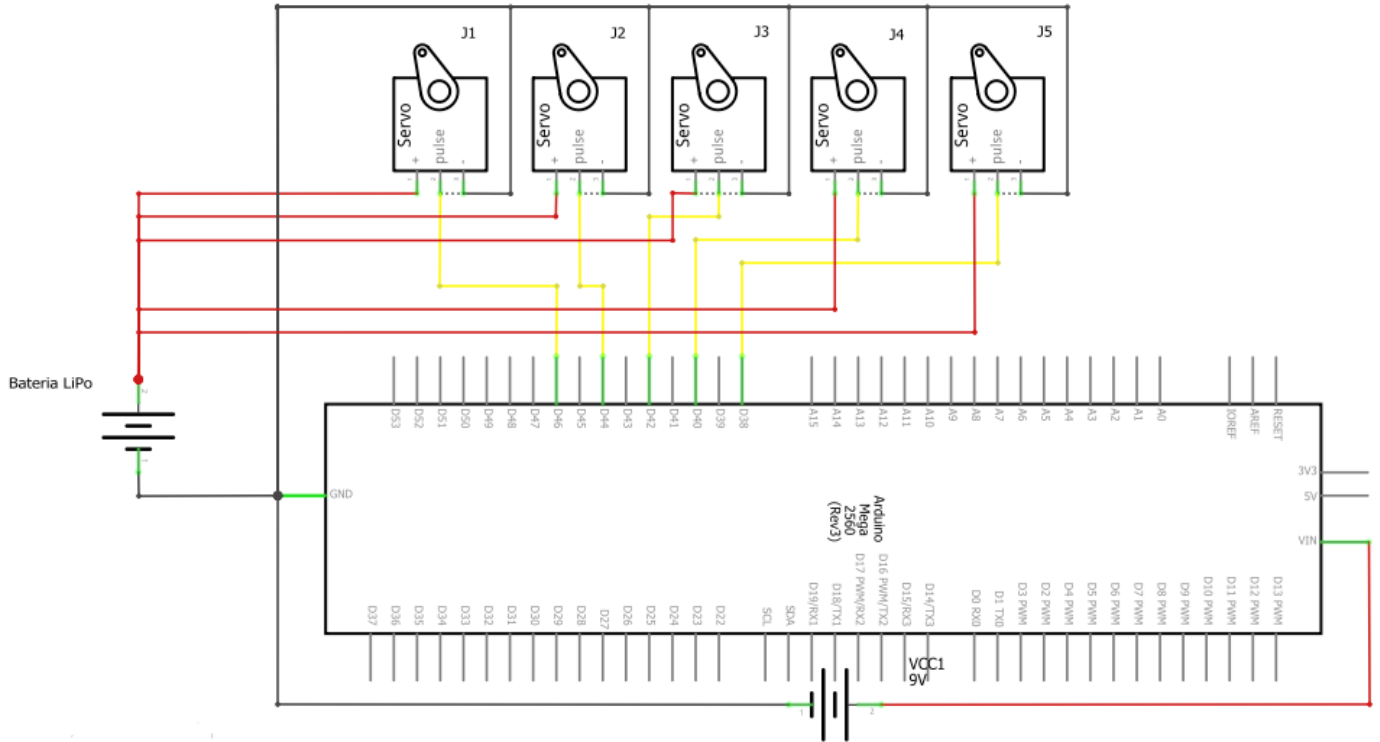
# 7.Circuit elèctric

## 7.1.Motor pas a pas

En aquest circuit també hi ha implementat un «switch". Aquest està implantat en la banda dreta de la mà. S'utilitza per saber quan la mà arriba al final del recorregut i d'aquesta manera poder començar el programa des de qualsevol posició.

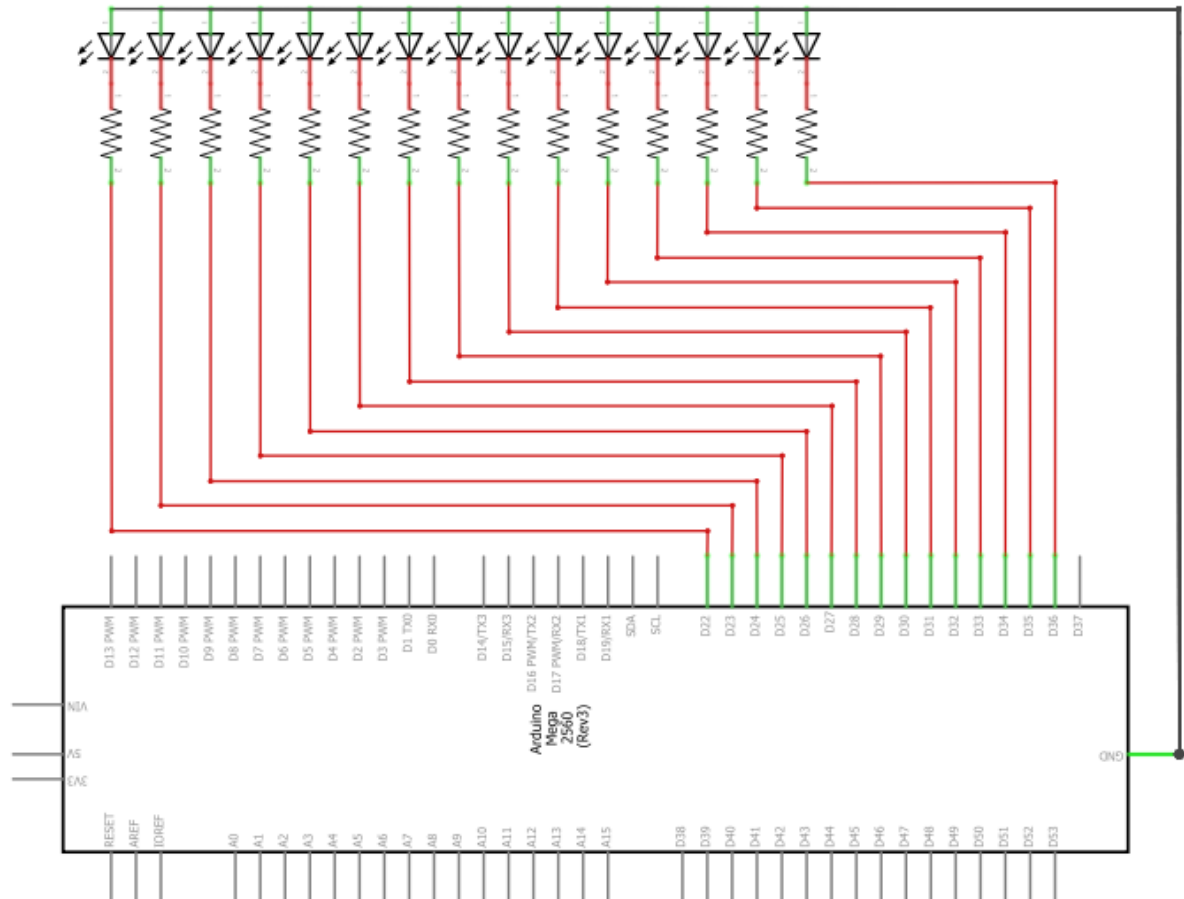


## 7.2.Servomotors



## 7.3.Leds

Aquest circuit és un complement al robot principal. Aquesta tira de leds permet senyalitzar les notes que toca el robot.



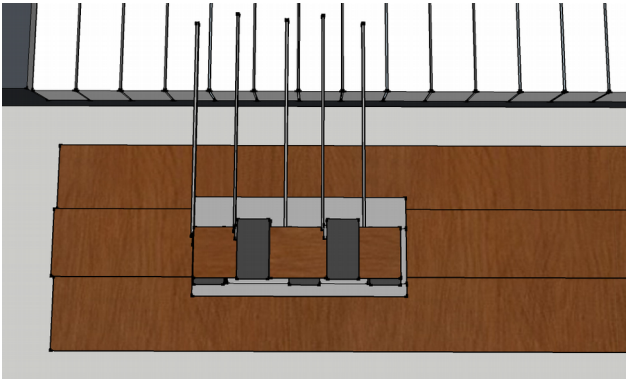
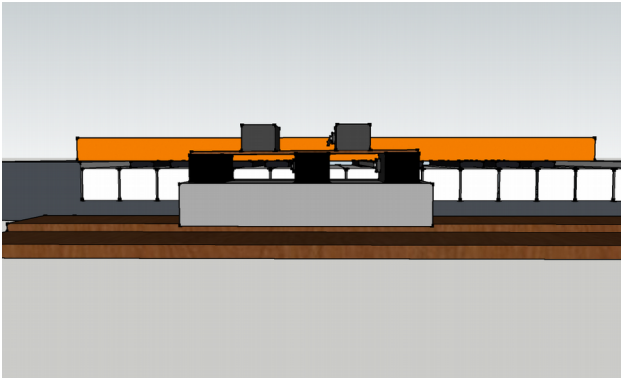
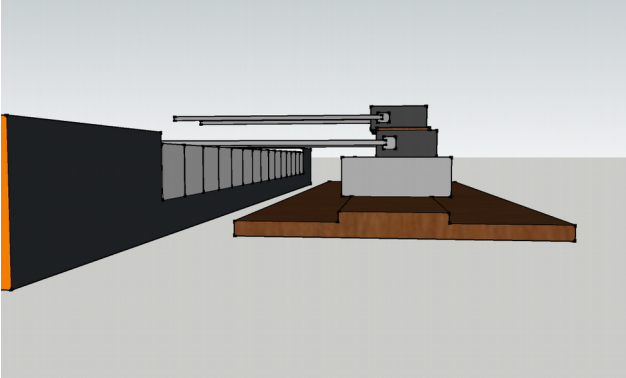
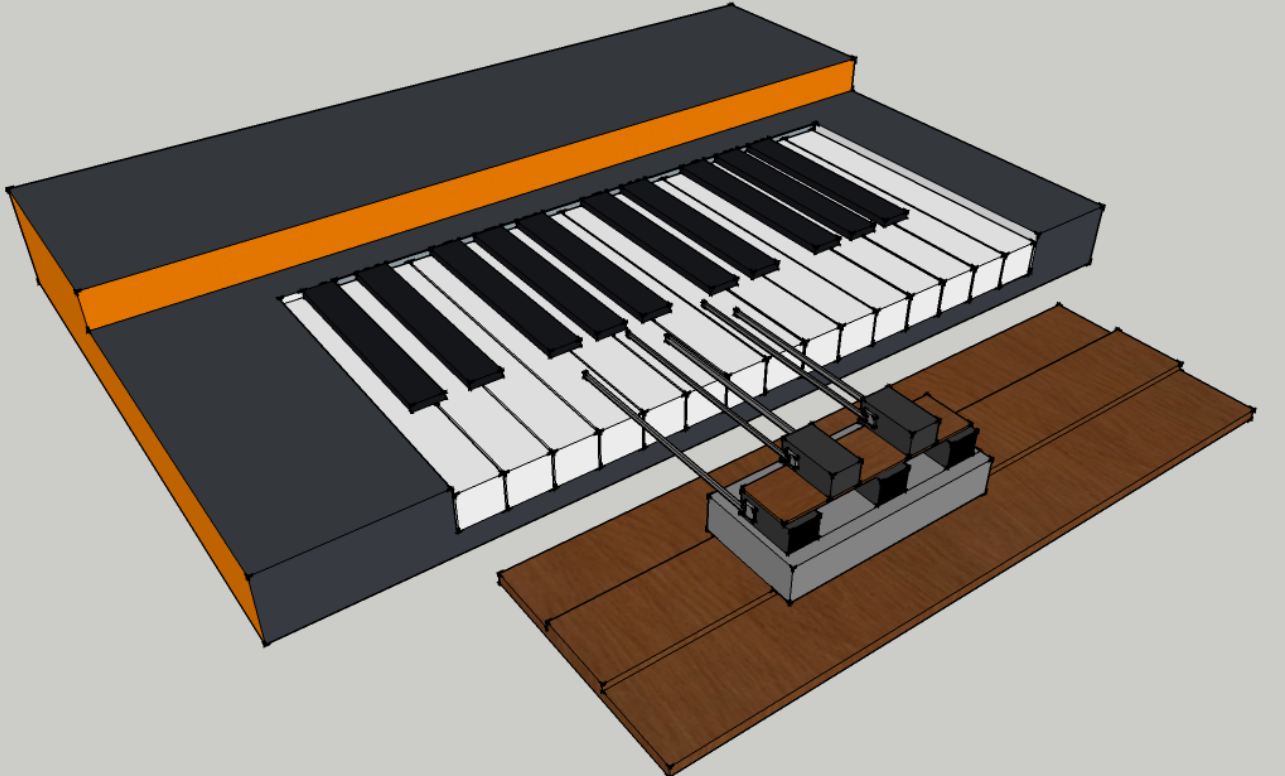
# 8.Estructura

## 8.1.Versions de l'estructura

Durant el desenvolupament del robot m'he trobat amb diferents impediments i dificultats, però sense cap dubte la dificultat més gran ha estat trobar una disposició de tots els components adequada, funcional i còmoda. Per això el treball final gairebé no té res a veure amb la primera idea que jo vaig tenir.

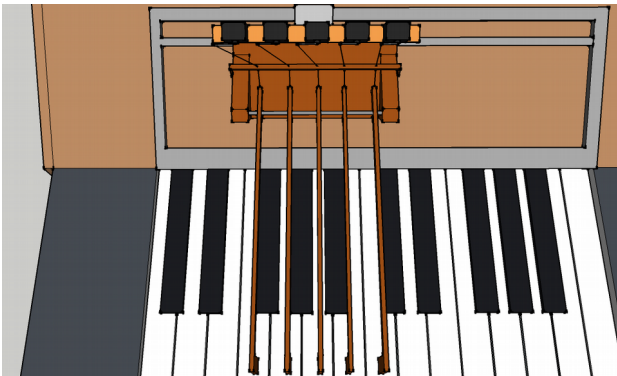
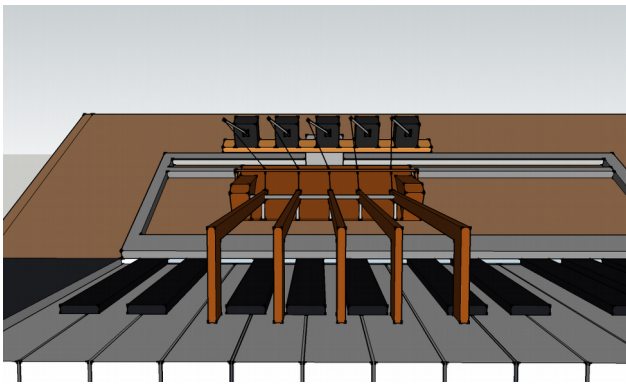
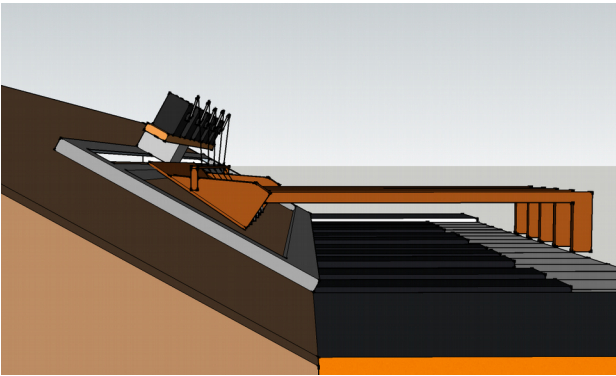
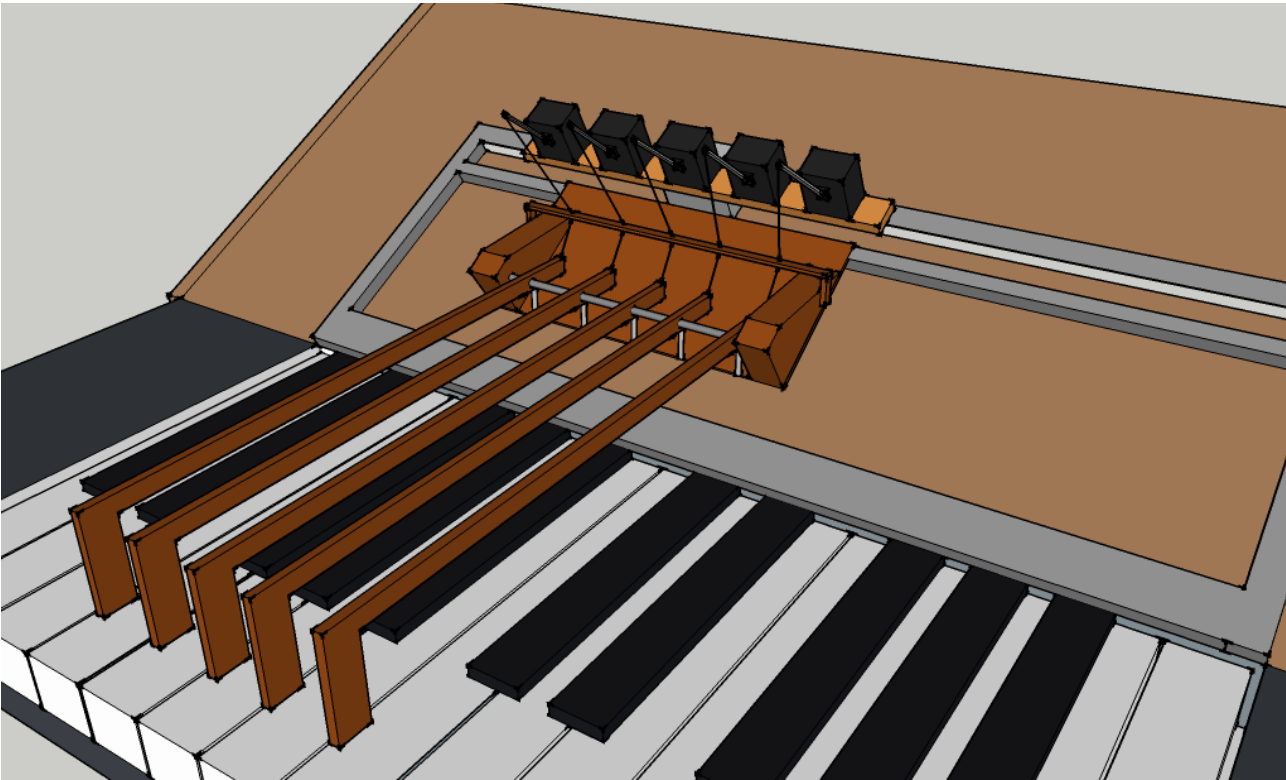
A continuació mostraré els diferents dissenys de l'estructura que he provat, des de la primera idea que vaig tenir fins al treball final i funcional. El canvi de disseny ha estat per raons diferents en cada cas. Alguns no eren molt còmodes o ocupaven molt espai, altres no funcionaven de la manera desitjada.

8.1.1. Primera idea conceptual



Aquesta va ser la primera idea que vaig tenir. A partir d'aquí vaig pensar altres models millorats i més funcionals. Aquest ja no el vaig portar a la pràctica per diverses raons. La primera i la més important: amb aquest disseny era impossible que l'usuari utilitzés l'altra mà per tocar el piano, ja que l'estructura queda al davant del piano i molestaria el pianista. Per altra banda volia optar per una versió més agradable a la vista, més compacta i amb una millor estètica. Després de pensar-hi molt vaig decidir optar per alguna versió diferent abans de començar a construir el robot.

8.1.2.Primer disseny físic



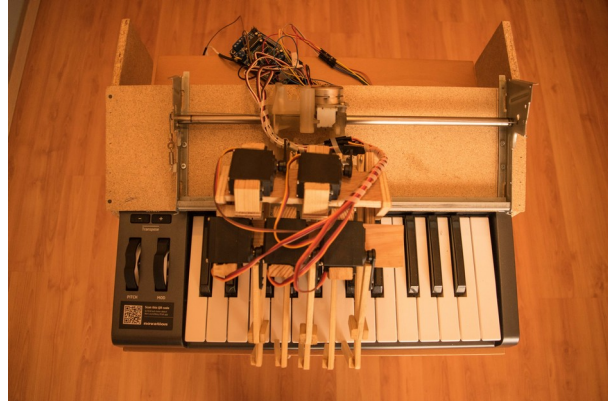
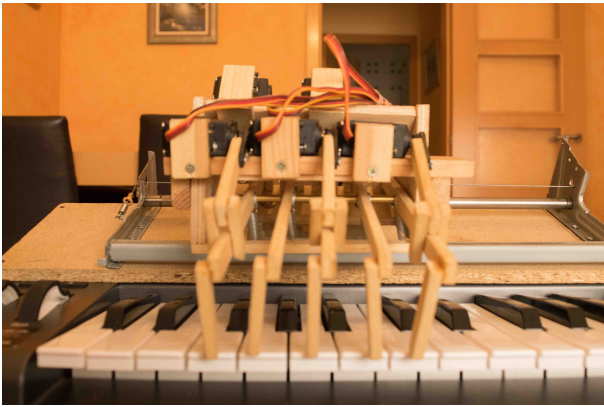
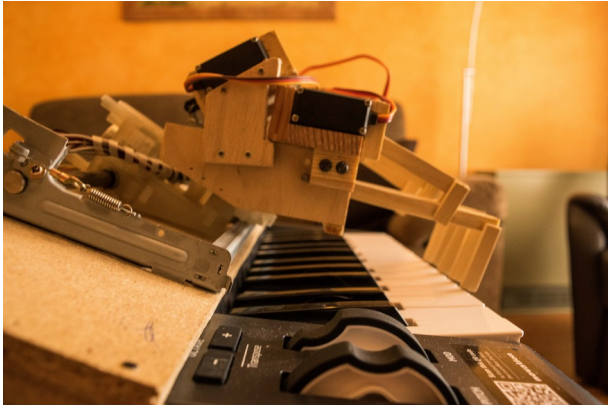
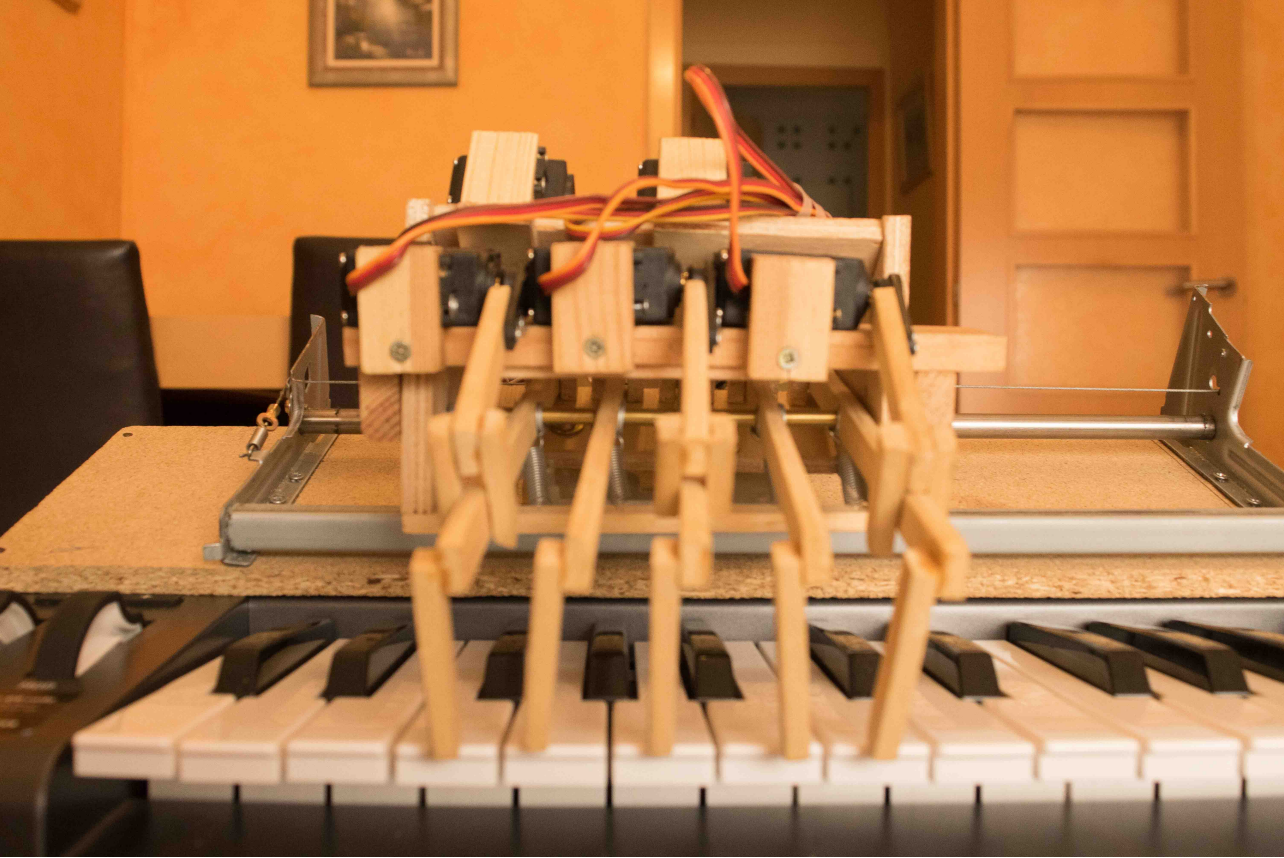


Aquesta va ser la primera versió que vaig portar a la pràctica. Vaig començar amb una estructura de fusta en la qual el robot es mantindria estable i on es podria moure. Aquest pla inclinat va ser dissenyat a mida per al teclat de què dispo, de manera que el robot es únicament funcional amb aquest. Sobre del pla inclinat vaig instal·lar una guia, sobre la qual el robot es podria moure. Aquesta formava part d'una màquina d'escriure, igual que el motor pas a pas. A partir d'aquí vaig començar a pensar la millor manera de disposar els servos. Vaig afegir a la guia dues plataformes, una per als servos i l'altra per als dits de fusta.

Els servomotors els vaig col·locar verticalment, ja que d'aquesta manera m'estalviava molt espai. El problema de posar-los verticalment és que el moviment rotacional d'aquests és perpendicular al que han de fer els dits, per això vaig connectar un fil a la punta de cada servo i als dits. Els fils primer passaven per una petita fusta que servia de guia, de manera que els cables quedaven rectes respecte als dits.

A primera vista semblava que aquesta versió podia ser la definitiva, però a l'hora de la veritat em vaig adonar que no era gens funcional, ja que els servos no tenien prou recorregut per estirar el fil, i per tant els dits no premien del tot les tecles del piano i les notes sonaven massa fluix. Per aquesta raó vaig decidir utilitzar un disseny que fos una barreja entre la primera i la segona versió.

8.1.3.Segon i final disseny físic



Aquesta estructura és l'actual. Si ens referim a la disposició dels servomotors he utilitzat un disseny similar al primer que vaig pensar, però amb la principal diferència que el segon i el quart dit són accionats pel darrere. També cal destacar que el segon pis està inclinat, de manera que s'aprofita més el recorregut dels servomotors i també ofereix una estètica més agradable a la vista. A sota de cada dit de fusta hi ha una molla, de manera que en retornar a la posició de repòs, el moviment és el mínim brusca possible. Tota aquesta estructura es mou sobre l'eix que vaig utilitzar en la versió anterior.

El primer, el tercer i el cinquè dit són accionats per una fusta que està enganxada a la pota dels servomotors. Per altra banda, la resta de dits són accionats directament per les potes dels servos amb molles.

Tot i ser una peça bastant gran, el conjunt de servos i dits queda molt compacte i el més important és que funciona bastant bé. El problema de falta de recorregut dels servomotors que tenia la versió anterior ha desaparegut i la disposició del robot ja no interfereix en les accions del pianista i no el pot molestar.

## 9.Materials per a l'estructura

Per a la construcció de l'estructura he utilitzat els següents materials:

2 Llistons de fusta	0,5x1,5x100 cm
Rectangle de fusta	1,5x4x8 cm
Plafó de fusta	1,5x50x50 cm
2 Llistons de fusta	1x4x15,5 cm
Guia màquina d'escriure	15x37 cm
Tub de ferro cargolat	Ø0,5x8,5 cm
Tub metàl·lic	Ø0,6x2cm
7 molles	
35 cargols	
Cola instantània	

# 10.Muntatge

Per muntar tot el robot he utilitzat les següents eines: serra, trepant, tornavís, soldador, pistola de claus i tenalles.

Passos per muntar el robot:

- 1) Tallar dues peces de fusta per fer el pla inclinat i unir-les amb ajuda de la pistola de claus.
- 2) Fer quatre forats a la guia de la màquina d'escriure i amb cargols unir-la al pla inclinat.
- 3) Tallar una fusta rectangular (base que uneix la guia amb la resta de components) i cargolar-la a la guia.
- 4) Tallar dues peces de fusta (costats de la base), fer un forat a cada peça amb el trepant i clavar-les a la peça anterior amb la pistola de claus.
- 5) Tallar cinc peces del llistó de fusta prim (dits). Fer un forat a cada dit amb el trepant.
- 6) Introduir el tub de ferro cargolat als costats de la base.
- 7) Introduir els dits al tub cargolat amb un tub de metall entre cada dit.
- 8) Enganxar un llistó de fusta sota la base. Unir les molles entre aquest llistó i els dits.
- 9) Tallar llistons de fusta per fer les dues plataformes del servos i cargolar-les als costats de la base.
- 10) Cargolar petites peces de fusta a les plataformes i a continuació cargolar els servos a aquestes peces.
- 11) Cargolar els dits secundaris del tres servos del davant i unir els servos del darrere als dits amb molles.
- 12) Connectar els cables dels servos i el motor pas a pas a la placa *Arduino*.

# 11.Pressupost

<b>Producte</b>	<b>Preu</b>
Arduino Mega 2560	16,95€
5 Servos	16,8€
Motor pas a pas	0€ (reciclat)
Bateria LiPo	14€
Bateria 9V	1,5€
Cables	2€
Cargols	1€
Tubs metàl·lics	0€ (reciclat)
Shield servos	1,2€
Shield motor pas a pas	3€
Fustes	0€ (reciclares)
Carregadors per l'alimentació	0€ (reciclat)
<b>Total</b>	<b>56,45€</b>

## 12.Programa

```
#include <AFMotor.h> // s'introdueixen les llibreries (motor pas a pas i servos)
#include <Servo.h>

//pin on es connecta el switch.
#define SWITCH 37

//pins dels servos:
#define PIN_DIT1 38
#define PIN_DIT2 40
#define PIN_DIT3 42
#define PIN_DIT4 44
#define PIN_DIT5 46

//posició normal dels servos
#define POS_NORMAL_DIT1 10
#define POS_NORMAL_DIT2 30
#define POS_NORMAL_DIT3 10
#define POS_NORMAL_DIT4 55
#define POS_NORMAL_DIT5 24

//posició dels servos al tocar una nota
#define POS_PITJAT_DIT1 19
#define POS_PITJAT_DIT2 57
#define POS_PITJAT_DIT3 20
#define POS_PITJAT_DIT4 90
#define POS_PITJAT_DIT5 35
```

```

/*****OBJECTES*****/

AF_Stepper motor(54, 2);
Servo dit1;
Servo dit2;
Servo dit3;
Servo dit4;
Servo dit5;

//es donen valors inicials a les variables.

int posicioDits[5]= {1,2,3,4,5};
int passosXtecla=54;
int tecla=0;
int tempsTecla=0;
int tempsEspera=0;

//cançó d'exemple
/*
The Ants Go Marching
[nota, tempsNota, relay]
*/
int notes = 104; //número de notes
int canso[104][3] = {
9, 275, 350, //el primer valor és la nota del piano que es toca
9, 400, 100, //els segon valor és la duració de cada nota (milisegons)
10, 275, 100, //el tercer valor és el temps entre cada nota (milisegons)
11, 400, 100,
10, 275, 100,
11 400 100
void setup() {
Serial.begin(9600);
pinMode(SWITCH, INPUT_PULLUP); //es connecta el switch al seu pin

// Pins dels leds, del 22 al 36
for(int i=22; i<37; i++){
pinMode(i, OUTPUT);
}

//es connecten els servos als seus pins corresponents
dit1.attach(PIN_DIT1);
dit2.attach(PIN_DIT2);
dit3.attach(PIN_DIT3);
dit4.attach(PIN_DIT4);
dit5.attach(PIN_DIT5);
motor.release();
//es posicionen els servos en la seva posició inicial
dit1.write(POS_NORMAL_DIT1);
dit2.write(POS_NORMAL_DIT2);
dit3.write(POS_NORMAL_DIT3);
dit4.write(POS_NORMAL_DIT4);
dit5.write(POS_NORMAL_DIT5);
delay(1000);
posicioInicial();
delay(1000);
tocaCanso();
}

```



```

//el motor pas a pas es mou cap a l'esquerra, independentment de la seva posició.
void posicioInicial() {
    motor.setSpeed(150);
    for(int i=0; i<600;i++){
        if(digitalRead(SWITCH) == LOW){ //si no s'activa el switch el motor es segueix movent
            motor.step(1, FORWARD, DOUBLE);
        }else{
//quan l'estructura entra en contacte amb el switch (el primer dit està en la posició de la primera nota)
//el motor pas a pas s'atura.
            break;
        }
    }
//incrementa els valors de l'array dels dits segons el valor passat.
void incrementaPosicioDits(int increment){
    for(int i=0;i<5;i++){
        posicioDits[i]= posicioDits[i]+increment;
    }
}

//decrementa els valors de l'array dels dits segons el valor passat.
void decreixPosicioDits(int decrement){
    for(int i=0;i<5;i++){
        posicioDits[i]= posicioDits[i]-decrement;
    }
}

//funcio que mira si amb els dits puc tocar la tecla donada.
//Retorna el numero de dit que la pot tocar i -1 si no.
int pucTocarTecla(int tecla){

    for(int i=0;i<5;i++){
        if (posicioDits[i]==tecla){
            return i;
        }
    }
    return -1;
}

```

```

//incrementa els valors de l'array dels dits segons el valor passat.
void incrementaPosicioDits(int increment){
    for(int i=0;i<5;i++){
        posicioDits[i]= posicioDits[i]+increment;
    }
}

//decrementa els valors de l'array dels dits segons el valor passat.
void decreixPosicioDits(int decrement){
    for(int i=0;i<5;i++){
        posicioDits[i]= posicioDits[i]-decrement;
    }
}

//funcio que mira si amb els dits puc tocar la tecla donada.
//Retorna el numero de dit que la pot tocar i -1 si no.
int pucTocarTecla(int tecla){

    for(int i=0;i<5;i++){
        if (posicioDits[i]==tecla){
            return i;
        }
    }
    return -1;
}

```

```

void tocaCanso() {

    int dit = -1;
    //es recuperen les dades de la cançó fila per fila
    for(int i=0;i<notes;i++){
        tecla= canso[i][0];
        tempsTecla= canso[i][1];
        tempsEspera= canso[i][2];

        //s'encen el led que correspon a la nota que es toca
        digitalWrite((21+tecla), HIGH);

        dit = pucTocarTecla(tecla);

        tocaTecla(dit, tecla, tempsTecla);

        //s'apaga el led de la tecla tocada
        digitalWrite((21+tecla), LOW);
        //Temps espera entre tecles
        delay(tempsEspera);
    }
}

//Funcio que li passo com a parmetres el valor q te el dit, la tecla a tocar
//i el temps que ha d'estar pitjada.
//Si un dels dits de la ma correspon amb la tecla, la toca el temps indicat.
//Si cap dit pot, posiciona la ma al lloc correcte i toca la tecla.
void tocaTecla(int dit, int tecla, int tempsTecla){
    switch (dit) {
        case -1:

            if(posicioDits[4]< tecla){
                //la tecla la tinc a la dreta
                posicionaMa((tecla-posicioDits[4]), 1);
                incrementaPosicioDits((tecla-posicioDits[4]));
                for(int i=0;i<5;i++){
                }
                Serial.println();
            }else{
                //la tecla la tinc a l'esquerra
                posicionaMa((posicioDits[0]-tecla), 0);
                decreixPosicioDits((posicioDits[0]-tecla));
                for(int i=0;i<5;i++){
                }
                Serial.println();
            }
    }
}

```

```

    dit = pucTocarTecla(tecla);
    //RECUSIVITAT
    tocaTecla(dit, tecla, tempsTecla);
    break;

    //es toquen les notes, depenent de a quin servo li correspon fer-ho

    //dit 1
case 0:
    dit1.write(POS_PITJAT_DIT1);
    delay(tempsTecla);
    dit1.write(POS_NORMAL_DIT1);
    break;

    //dit 2
case 1:
    dit2.write(POS_PITJAT_DIT2);
    delay(tempsTecla);
    dit2.write(POS_NORMAL_DIT2);
    break;

    //dit 3
case 2:
    dit3.write(POS_PITJAT_DIT3);
    delay(tempsTecla);
    dit3.write(POS_NORMAL_DIT3);
    break;

    //dit 4
case 3:
    dit4.write(POS_PITJAT_DIT4);
    delay(tempsTecla);
    dit4.write(POS_NORMAL_DIT4);
    break;

    //dit 5
case 4:
    dit5.write(POS_PITJAT_DIT5);
    delay(tempsTecla);
    dit5.write(POS_NORMAL_DIT5);
    break;
}
}

void loop() {

}

```

## 13.Possibles aplicacions del robot

Tot i ser un robot molt simple i amb unes capacitats limitades, el robot pot adquirir més aplicacions a part de les d'entreteniment. De fet aquesta ja era una premissa cercada des de l'inici, que el robot pogués respondre a una necessitat real, ja que d'alguna manera, aquest hauria de ser l'objectiu final de qualsevol projecte.

Per tant, l'aplicació més important i destacada és que podria ser un suport per a persones amb alguna discapacitat o limitació de moviment en alguna de les dues mans. Gràcies al robot la persona pot tocar la part de la cançó que pertoca a la mà que ell necessiti, d'aquesta manera s'aconsegueix tocar una cançó de manera física només amb una mà. Això també es podria aconseguir amb programes d'ordinador que fan sonar les tecles d'una mà determinada, però d'aquesta manera es perdria que es pogués veure com es toca la cançó directament al teclat.

A part d'aquesta aplicació més clínica, també es pot optar per aplicacions educatives, de manera que amb l'ajuda d'una tira de leds es podria aprendre a tocar cançons sense necessitat de tenir un ordinador obert i haver de mirar constantment a la pantalla. Els leds d'aquesta tira s'encendrien quan seria necessari tocar una tecla.

## 14. Conclusions

Després de totes les dificultats i havent arribat a la versió final del robot, he d'admetre que el resultat ha estat bastant similar al que vaig imaginar des d'un bon principi. Potser amb un motor pas a pas més potent s'aconseguiria una velocitat de moviment molt més alta, però l'adquisició d'aquest trencaria amb el principi de fer un robot amb la majoria de components reciclats i el més barats possibles.

Al final el robot és capaç de tocar cançons d'un nivell baix i mitjà, de manera que els resultants són bastant adients als esperats.

El més satisfactori ha estat l'aprenentatge paral·lel al disseny i a la construcció del robot. No només he adquirit alguns coneixements de programació i d'electrònica, sinó que també he adquirit coneixements bàsics de piano i disseny amb SketchUp.

Per concloure si hagués de millorar el robot optaria per un millor motor pas a pas i fer una estructura adaptable a qualsevol piano.

# 15. Bibliografía i Webgrafía

## 15.1. Webgrafía

Arduino:

<https://www.arduino.cc/>

<https://es.wikipedia.org/wiki/Arduino>

<https://en.wikipedia.org/wiki/Arduino>

<https://vimeo.com/18390711> (Documental)

<http://www.prometec.net/indice-tutoriales/>

[Consulta: 11 Octubre, 27 Octubre]

Motor pas a pas:

[https://es.wikipedia.org/wiki/Motor\\_paso\\_a\\_paso](https://es.wikipedia.org/wiki/Motor_paso_a_paso)

<http://www.prometec.net/motor-28byj-48/>

<http://diymakers.es/mover-motores-paso-paso-con-arduino/>

[http://robots-argentina.com.ar/MotorPP\\_basico.htm](http://robots-argentina.com.ar/MotorPP_basico.htm)

<http://www.monografias.com/trabajos17/motor-paso-a-paso/motor-paso-a-paso.shtml>

[Consulta: 5 Novembre]

Servomotors:

<https://es.wikipedia.org/wiki/Servomotor>

<https://ca.wikipedia.org/wiki/Servomotor>

[http://platea.pntic.mec.es/vgonzale/cyr\\_0204/ctrl\\_rob/robotica/sistema/motores\\_servo.htm](http://platea.pntic.mec.es/vgonzale/cyr_0204/ctrl_rob/robotica/sistema/motores_servo.htm)

<http://www.areatecnologia.com/electricidad/servomotor.html>

[Consulta: 5 Novembre]

C i C++:

<https://ca.wikipedia.org/wiki/C%2B%2B>

<https://es.wikipedia.org/wiki/C%2B%2B>

<http://www.monografias.com/trabajos4/lenguajec/lenguajec.shtml>

[http://www.aprenderaprogramar.com/index.php?option=com\\_content&view=category&id=82&Itemid=210](http://www.aprenderaprogramar.com/index.php?option=com_content&view=category&id=82&Itemid=210)

[http://decsai.ugr.es/~jfv/ed1/c/cdrom/cap1/f\\_cap12.htm](http://decsai.ugr.es/~jfv/ed1/c/cdrom/cap1/f_cap12.htm)

[Consulta: 3 Deseembre]

## **15.2.Bibliografia**

TORRENTE ARTERO, Óscar. ARDUINO, *Curso práctico de formación*. México, Alfaomega Grupo editor, 2013.

[Consulta: durant tot el projecte]



# Vídeo del robot en funcionament

En el següent enllaç es pot trobar un vídeo del robot en funcionament, tocant quatre petites peces musicals.

<https://www.youtube.com/watch?v=fmTlPdhvGRI>