

# TRAVELING CONTINUO



- 1. INTRODUCCIÓN.**
- 2. OBJETIVOS Y HIPOTESIS.**
  - 2.1. Objetivos teóricos.**
  - 2.2. Objetivos prácticos.**
- 3. MARCO TEORICO**
  - 3.1. Slider**
    - 3.1.1. ¿Qué es?**
    - 3.1.2. ¿Para qué sirve? Campos de utilización.**
    - 3.1.3. Sus componentes.**
  - 3.2. Arduino**
    - 3.2.1. Introducción a Arduino.**
    - 3.2.2. Orígenes.**
    - 3.2.3. ¿Para qué sirve? Campos de utilización.**
    - 3.2.4. Sus componentes.**
    - 3.2.5. “Háztelo tú mismo” Impresoras 3D**
  - 3.3. Software. Programación.**
- 4. MARCO PRÁCTICO**
  - 4.1. Construcción del Slider.**
    - 4.1.1. Proceso de realización.**
    - 4.1.2. “Háztelo tú mismo” Impresoras 3D**
    - 4.1.3. Electrónica.**
    - 4.1.4. Software.**
  - 4.2. Montaje y edición de vídeo.**
- 5. RESULTADOS Y CONCLUSIONES**
  - 5.1. Estructura.**
  - 5.2. Electrónica.**
  - 5.3. Software.**
  - 5.4. Montaje y edición de vídeo.**
- 6. CONCLUSIÓN DEL TRABAJO.**
- 7. AGRADECIMIENTOS.**
- 8. BIBLIOGRAFÍA.**
- 9. OTRAS FUENTES.**

*"El verdadero progreso es el que  
pone la tecnología al alcance de todos."*

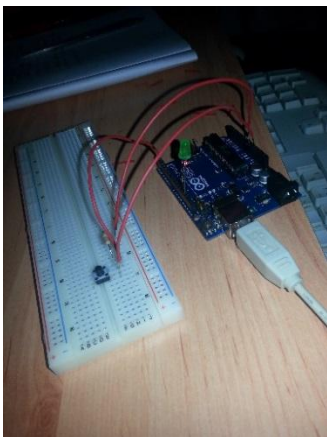
Henry Ford

## 1. INTRODUCCIÓN

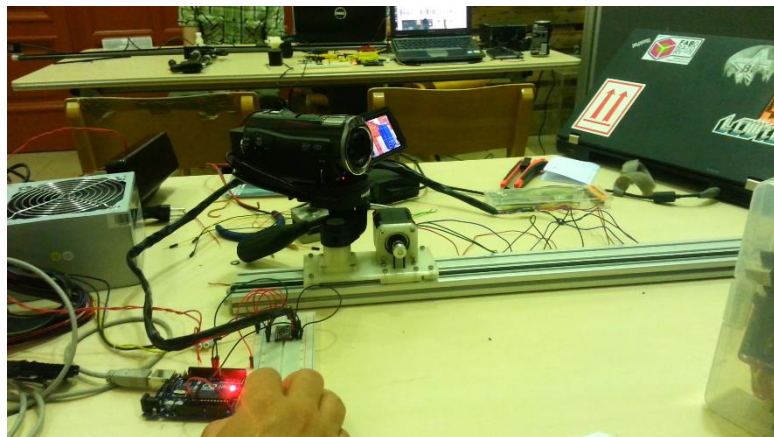
He acabado decidiéndome por este proyecto ya que buscaba algo que me llamara la atención y que estuviera dentro de la innovación, es decir, relacionado con la tecnología. Desde bien pequeño he tenido un gran interés sobre las cámaras fotográficas y sobre la edición de vídeo. Mi tío me introdujo en este grandioso y diverso mundo de la tecnología, él me enseñó y me explicó muchas cosas sobre el entorno que contiene la imagen, el vídeo y la fotografía.

Un día me trajo una placa base pequeña, llamada Arduino. Me estuvo enseñando qué era, y para que servía, hicimos una pequeña demostración de encender y apagar un LED, algo sencillo pero que me captó plenamente la atención y en mí se despertó un interés de saber más sobre esta curiosa placa. Más adelante me compré una e hice muchísimas pruebas con LED's en una protoboard, con botones, cableado, sensores... me resultó fascinante.

Llegó el día de ir más lejos que todo esto de encendido y apagado de un LED, decidí apuntarme a un cursillo sobre el Slider, con gente más mayor que yo, de hecho era el más joven. Era un tema del que apenas tenía idea de que trataba, me dijeron que era algo experimental. La primera clase me resultó muy familiar, me daba cuenta de que entendía las cosas y que aparecía materia explicada en clase. El tema relacionaba desde un simple diseño hasta las mates aplicadas, pasando por la electrónica, mecánica y software. Más adelante a medida que avanzaban las clases iba conociendo a gente y cada vez aprendía más, nos transmitíamos conocimiento entre todos nosotros, fue una experiencia única. A medida que evolucionaba el trabajo y daba sus frutos el gran esfuerzo y las horas empleadas en el proyecto me di cuenta de que era algo fascinante, esto tenía que ser mi trabajo de investigación.



Prueba de LED con pulsador.



Probando el Slider en el curso.

## **2. OBJETIVOS Y HIPOTESIS.**

### **2.1. Objetivos teóricos.**

- Aprender el funcionamiento de los motores bipolares.
- Buscar información sobre Slider.
- Aprender a programar.
- Clasificar redacción y redactar memoria

### **2.2. Objetivos prácticos.**

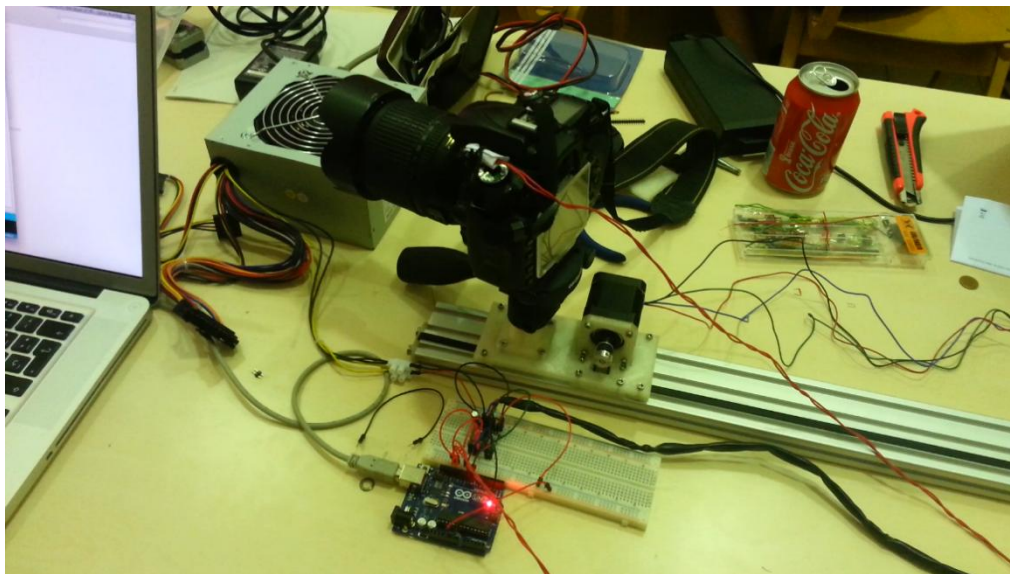
- Construir un Slider.
- Conseguir que funcione dándole órdenes a través de una placa Arduino.
- Programar la placa Arduino.
- Montar y añadir efectos de vídeo y fotografía.
- Interactuar con el mundo físico/virtual
- Asistir a un curso de 50 horas para desarrollar el proyecto Slider.
- Conseguir el efecto óptico deseado con la cámara.
- Aprender a soldar.

### 3. MARCO TEÓRICO

#### 3.1. SLIDER

##### 3.1.1. ¿Qué es?

El Slider es un rail motorizado por un motor paso a paso denominado Stepper. Este motor es un actuador electromagnético rotatorio que convierte mecánicamente entradas de pulsos digitales a movimiento rotatorio incremental. Stepper mueve una estructura deslizante friccionada por una correa dentada. Esta estructura soporta una cámara fotográfica y el motor paso a paso. Gracias a la fricción del motor con la correa se genera un movimiento lineal con una velocidad variable en el tiempo. Este movimiento se consigue gracias a un driver que interpreta las órdenes de un microcontrolador (Arduino) programado por *código fuente*, dando lugar al movimiento rotatorio del motor paso a paso.



Haciendo pruebas con Slider.

*Código fuente:* El código fuente de un programa informático es un conjunto de líneas de texto que son las instrucciones que debe seguir la computadora para ejecutar dicho programa.

### 3.1.2. ¿Para qué sirve? Campos de utilización

Este rail motorizado está dentro del mundo de la imagen y su referente es el vídeo y la fotografía.

En el vídeo sirve para controlar el movimiento y repetirlo varias veces de la misma manera, con las mismas aceleraciones y desaceleraciones, paradas, arranques y posiciones en el espacio para generar efectos de colorización o incrustación de objetos 3D o imágenes 2D en *croma key*.



Ejemplo Croma key.



Ejemplo Croma key.

*Croma key*: es una técnica audiovisual utilizada ampliamente tanto en cine y televisión como en fotografía, que consiste en extraer un color de la imagen y reemplazar el área que ocupaba ese color por otra imagen.

Con relación a la fotografía es hacer disparos a diferentes distancias entre lapsos de tiempo generando una ilusión de movimiento (*Time lapses*).



Ejemplo time lapse.



Ejemplo time lapse.

*Time lapse*: Es una técnica fotográfica muy popular para mostrar diferentes sucesos que por lo general suceden a velocidades muy lentas e imperceptibles al ojo humano.

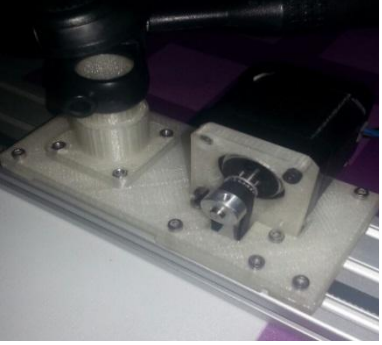
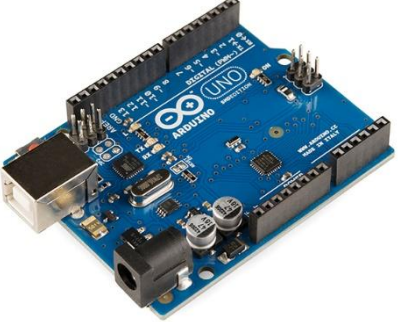

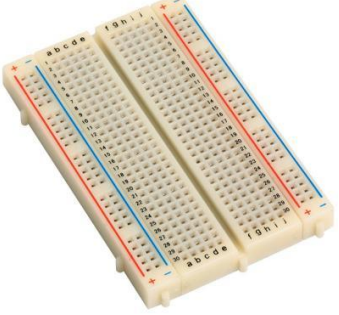



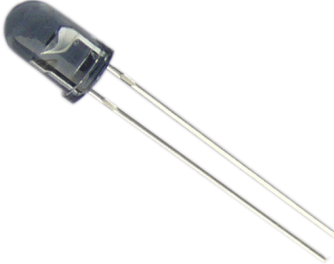


### 3.1.3. Componentes

Slider está compuesto por:

Nombre	Función	Imagen
<b>Rail de aluminio,</b> diseñado a ordenador y cortado por una impresora láser.	Guía.	
<b>Correa dentada de plástico.</b>	Crear fricción con el motor.	
<b>Motor bipolar paso a paso. (Stepper)</b>	Rotar.	
<b>Rótula de trípode.</b>	Sujetar la cámara.	



<p><b>Estructura base,</b> diseñada a ordenador y imprimida por una impresora 3D.</p>	<p>Sujetar cámara, rótula y motor Stepper</p>	
<p><b>Placa Arduino</b></p>	<p>Enviar órdenes al microdriver.</p>	
<p><b>A4988</b></p>	<p>Controlador con el que podremos mover el motor paso a paso desde nuestro Arduino.</p>	
<p><b>Protoboard</b></p>	<p>Placa experimental de circuitos.</p>	

<p><b>Cámara réflex.</b> <b>(Nikon D5100)</b></p>	<p>Disparador con control remoto. Hacer fotografías.</p>	
<p><b>Led infrarrojo.</b></p>	<p>Emisor de señales vía infrarrojos hacia la cámara.</p>	
<p><b>Fuente de alimentación.</b></p>	<p>Transmite energía eléctrica.</p>	
<p><b>Ordenador.</b></p>	<p>Programar y alimentar a Arduino.</p>	

## 3.2. ARDUINO



### 3.2.1. Introducción

Arduino es una plataforma de hardware libre (open-hardware), que conecta el mundo físico con el mundo virtual, o el mundo analógico con el digital.

Consta de una sencilla placa con entradas y salidas, analógicas y digitales, en un entorno de desarrollo que instalándolo y conectando la placa Arduino al ordenador la podemos programar ya que está basada en el lenguaje de programación Processing.

### 3.2.2. Orígenes

El nombre del proyecto proviene del nombre de Bar di Re Arduino, traducido al español Bar del Rey Arduino, donde Massiano, uno de sus fundadores pasaba el tiempo.

Se inicia en 2005 como un proyecto para estudiantes en el instituto IVREA, en Italia, cuando estos no podían costear las placas que se utilizaban en aquel entonces. Dichas placas eran placas Basic que costaban más de 100 dólares cada una. A la hora de hacer un proyecto que necesitaba 3 o más placas, fácilmente les salía a un precio elevado. Con esto en mente nació Arduino, placas de código abierto a precio asequible. A parte uno de los muchos motivos pero más importante de su creación, fue la necesidad de pasar del mundo virtual al mundo físico mediante placas de programación sencilla.

### 3.2.3. ¿Para qué sirve? Campos de utilización

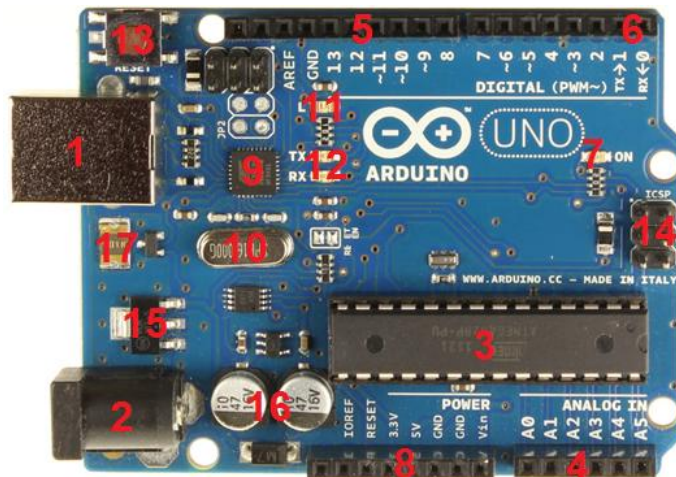
Arduino viene con una *memoria flash* a la que tú le cargas un programa. El programa se lo cargas por USB y podemos realizar múltiples funciones. Por ejemplo puedes programar tu placa para encender una luz, activar un ventilador o incluso ir tan lejos como para crear tu propio robot. Con un poco de creatividad y conocimientos verdaderamente mínimos de Arduino puedes crear prácticamente cualquier cosa.

Arduino lo podemos encontrar desde un simple accionamiento de una bombilla hasta el sistema de una casa domótica. Un tema muy interesante que es aplicado con Arduino es la utilización de impresoras 3D. Arduino es utilizado en aplicaciones electrónicas.

**Memoria flash:** *Permite la lectura y escritura de múltiples posiciones de memoria en la misma operación. Se trata de la tecnología empleada en los dispositivos denominados memoria USB.*

### 3.2.4. Componentes

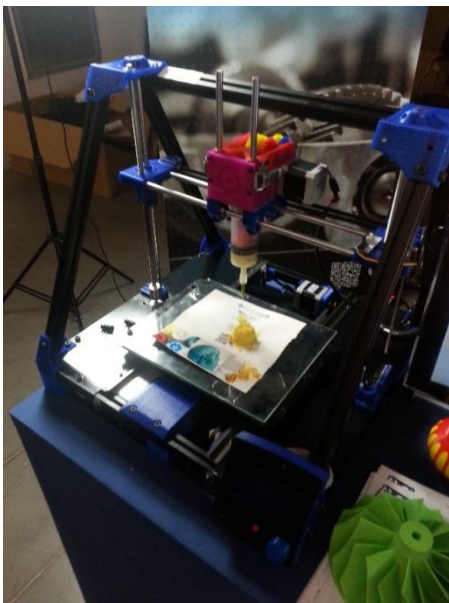
- 1- Conexión USB.
- 2- Power (Entrada de la fuente de alimentación).
- 3- ATmega328 (Cerebro principal de la placa).
- 4- Entradas analógicas.
- 5- Conexiones digitales (Entrada/Salida).
- 6- Pines TX/RX.
- 7- Led de encendido y apagado.
- 8- Barra de energía (Voltios).
- 9- Chip comunicador con el ordenador.
- 10- Chip controlador de la frecuencia del microcontrolador.
- 11- Led de pruebas.
- 12- Led TX/RX (Indicador comunicador con el ordenador).
- 13- Botón reset de código de programación. (Borrado de memoria).
- 14- ICSP.
- 15- Regulador de voltaje.
- 16- Condensadores.
- 17- Fusible.



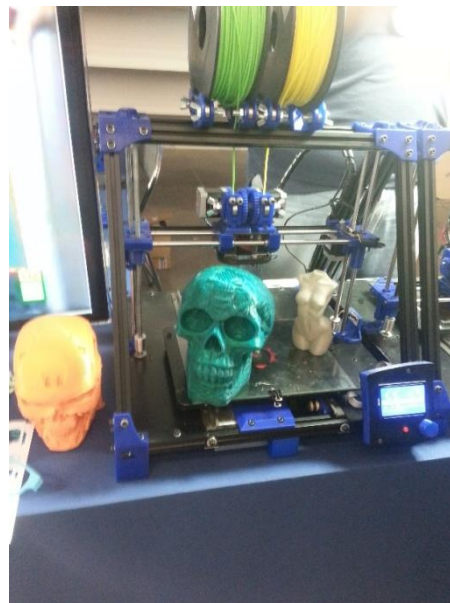
Placa Arduino.

### 3.2.5. “Háztelo tú mismo” Impresoras 3D.

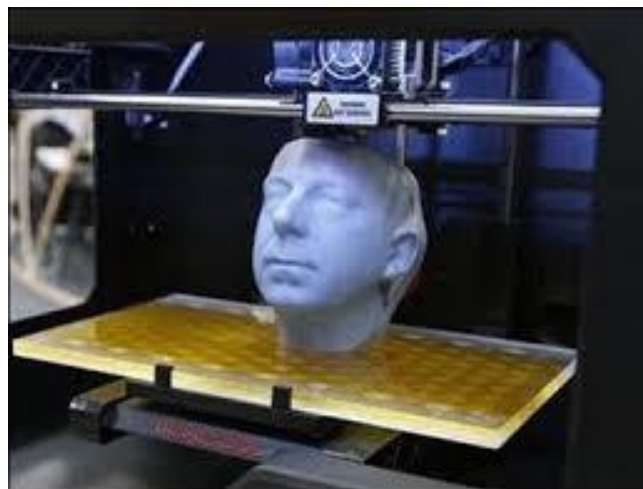
La placa Arduino ha sido una revolución para el mundo tecnológico. Simplemente que con esta simple placa puedes lograr hacer grandes cosas. Es algo histórico pensar y decir, necesito “x”, y poder diseñarlo por tus propios medios sin depender de nadie, solo de mí mismo. Podemos construir y programar para hacerlo funcionar como queramos. Un ejemplo claro de esto son las impresoras 3D, programadas por Arduino, capaces de construir cualquier cosa que le ordenes, incluso materiales para crear otra impresora 3D.



Impresora 3D en funcionamiento.



Impresora 3D en funcionamiento.



Impresora 3D en funcionamiento.

### **3.3. SOFTWARE. PROGRAMACIÓN.**

#### **3.3.1. Introducción**

«Software libre» es el software que respeta la libertad de los usuarios y la comunidad. En grandes líneas, significa que los usuarios tienen la libertad para ejecutar, copiar, distribuir, estudiar, modificar y mejorar el software. Es decir, el «software libre» es una cuestión de libertad, no de precio. Es decir es una «libertad de expresión».

Se basa en:

- La libertad de ejecutar el programa, para cualquier propósito.
  
- La libertad de estudiar cómo trabaja el programa, y cambiarlo para que haga lo que usted quiera. El acceso al código fuente es una condición necesaria para ello.
  
- La libertad de redistribuir copias para que pueda ayudar al prójimo.
  
- La libertad de mejorar el programa y publicar sus mejoras, y versiones modificadas en general, para que se beneficie toda la comunidad. El acceso al código fuente es una condición necesaria.

Algunas ordenes de programación:

#### **Estructura de un Sketch:**

- void setup()** Inicio de un código. Se utiliza para iniciar las variables y las librerías.
  
- void loop()** Genera un bucle repitiendo el código hasta que se lo indicas.
  
- Delimitadores: {} entre llaves
  
- ; punto y coma
  
- //, /\* comentarios. (No afectan al código).
  
- Operadores aritméticos: +, -, \*, /, %
  
- Asignación: =
  
- Operadores de comparación: ==, !=, <, >, <=, >=

## Estructuras de control

-Condicionales: **if, if...else, switch case**

-Bucles: **for, while, do... while**

-Bifurcaciones y saltos: **break, continue, return, goto**

## Tiempo

-**unsigned long millis** () Milisegundos.

-**unsigned long micros** () Microsegundos en un segundo.

-**delay** (ms) Pausa en milisegundos.

-**delayMicroseconds** (microsegundos)

## Constantes

-**HIGH/LOW**: representan los niveles alto y bajo de las señales de entrada y salida. Los niveles altos son aquellos de 3 voltios o más.

-**INPUT/OUTPUT**: entrada o salida.

-**false (falso)**: Señal que representa al cero lógico. A diferencia de las señales **HIGH/LOW**, su nombre se escribe en letra minúscula.

-**true (verdadero)**: Señal cuya definición es más amplia que la de false. Cualquier número entero diferente de cero es "verdadero", según el álgebra de Boole, como en el caso de -200, -1 o 1. Si es cero, es "**false**".

## Funciones básicas

### E/S digital

-**pinMode** (pin, modo)

-**digitalWrite** (pin, valor)

-**int digitalRead** (pin)



## E/S analógica

-**analogReference** (tipo)

-**int analogRead** (pin)

-**analogWrite** (pin, valor)

## E/S avanzada

**shiftOut** (dataPin, clockPin, bitOrder, valor)

**unsigned long pulseIn**(pin, valor)

Ejemplo de encendido y apagado de un LED:

```
// Pin 13 has an LED connected on most Arduino boards.  
(Conectado en el pin 13 de la placa de Arduino)  
// give it a name:  
int led = 13;  
  
// the setup routine runs once when you press reset:  
(Inicio despues de cada arranque o reinicio de la placa Arduino)  
void setup() {  
    // initialize the digital pin as an output. (Salida)  
    pinMode(led, OUTPUT);  
}  
  
// the loop routine runs over and over again forever:  
(Repetición infinita. Una vez terminada va de vuelta al principio).  
void loop() {  
    digitalWrite(led, HIGH); // turn the LED on (HIGH is the  
voltage level)(Encendido)  
    delay(1000); // wait for a second  
    digitalWrite(led, LOW); // turn the LED off by making the  
voltage LOW (Apagado)  
    delay(1000); // wait for a second  
}
```

## 4. MARCO PRÁCTICO

### 4.1. CONSTRUCCIÓN DEL SLIDER

#### 4.1.1. Proceso de realización.

Este proceso es la construcción del esqueleto.

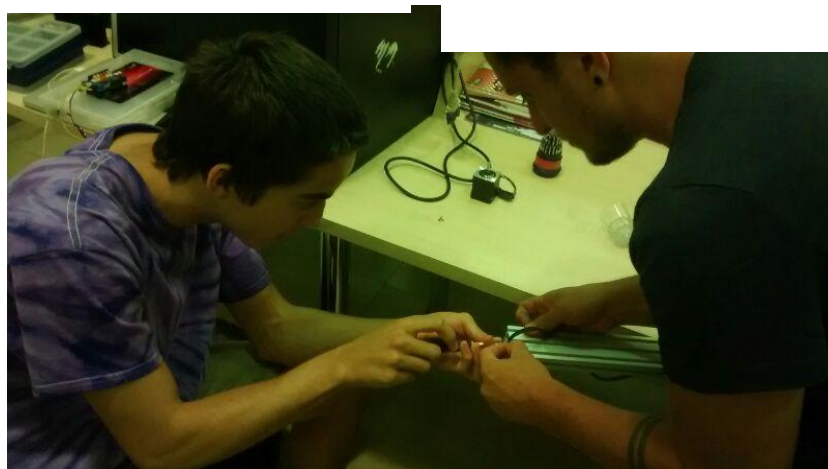
Para la construcción de un Slider en la parte mecánica se necesita una barra de aluminio, que su función será de carril, es decir, será la guía de la cámara y el motor. Una correa dentada, cuatro piezas de plástico diseñadas y construidas por medio de una impresora 3D, que irán colocadas dos a cada extremo del carril, serán atornilladas y su función será sujetar dicha correa. A parte, necesitaremos un motor stepper y la base donde irá colocada también la cámara. Esta base incorpora una pieza diseñada por mí, para poder adaptar la rótula y así poder estabilizar y mantener la cámara quieta sin ningún riesgo de movimiento ni de caída.



Montando Slider. Colocando la correa dentada.



Atornillando el motor a la estructura de plástico.



Atornillando la correa a los extremos del carril.

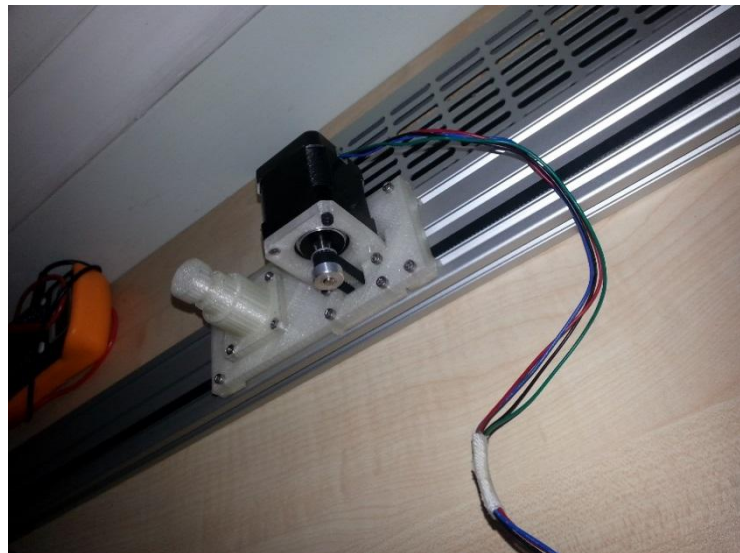
#### 4.1.2. “Háztelo tú mismo” Impresoras 3D

En mi caso necesitaba una estructura que encajara en el rail y fuera capaz de deslizarse con una mínima fricción y que pudiera sujetar una rótula y un motor. Sólo tuve que diseñarla tomando las medidas exactas con unos milímetros de más y de menos dependiendo de la zona para que encajara bien y no fuera duro, es decir, dejando un poco de juego para mejorar su rendimiento y evitar que el motor sufriera daños teniendo que mover algo atascado. Una vez finalizado el diseño en el ordenador la imprimí enviando órdenes a mi Arduino el cual ordenaba al micro driver que movimientos debían hacer los motores Stepper.

Es algo fascinante, rápido pero que lleva su tiempo diseñar, cuyos resultados son increíbles. He logrado pasar de algo virtual al mundo físico.



Probando el Slider en casa.



Estructura blanca de plástico imprimida por una impresora 3D.



Mi professor con la impresora que imprimí la estructura.

### 4.1.3. Electrónica.

Este proceso es la construcción de los órganos, es decir, los componentes del Slider.

A la hora de introducirse en la electrónica corremos una serie de riesgos, ya que hay que ir con mucho cuidado de no estropear los componentes ni de hacernos daño, hay que trabajar con seguridad y con medidas. Hay que tener en cuenta que las piezas o componentes no son baratos.

Para lograr que nuestro Slider funcione debemos tener los siguientes materiales y herramientas.

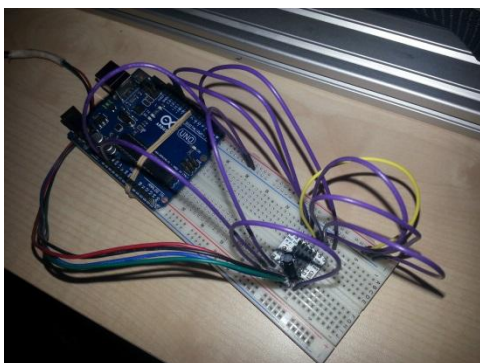
Una fuente de alimentación, cables con cabeza de plástico en ambos extremos, una protoboard, un driver A4988, un Arduino con su cable USB, un soldador, soporte para soldar, estaño, un voltímetro, un ordenador y un esquema para montar nuestro circuito.

Para lograr que la fuente de alimentación funcionase sin estar en la placa base del ordenador tuvo que **puentear** el cable **GND** (Ground).

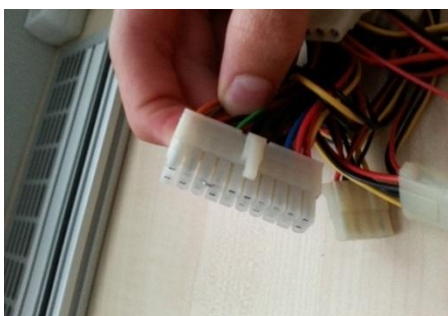
**Puentear:** Unir dos cables mediante un conductor para que estén en contacto.

**Ground:** Denominado tierra. Es un referencial negativo cuyo valor potencial es 0.

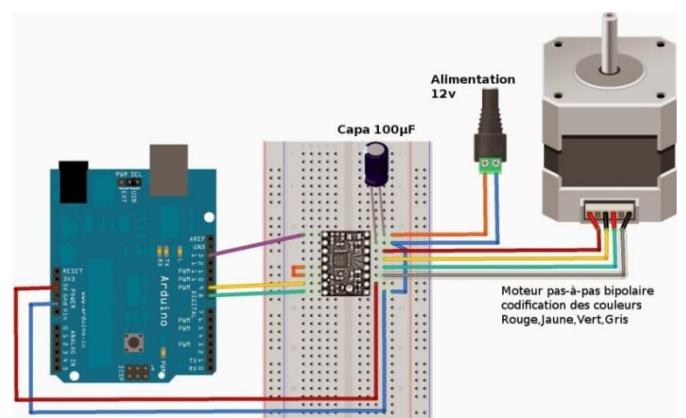
Este proceso es el más delicado ya que un mínimo error puede ser el fracaso de tu proyecto... pero esto no quiere decir que no lo puedas volver a intentar.



Circuito utilizado para el Slider.



Cable GND puentado.



Esquema circuito A4988.

#### 4.1.2. Software.

Este es el Software que utilicé para hacer funcionar mi Slider con el objetivo de hacer fotografías con una distancia diferente y con control remoto para luego a la hora de montar el video diese el efecto de movimiento.

Explicación del Software:

**#include <NikonIrControl.h>** : Librería para utilizar el control remoto.

**#include <AccelStepper.h>** : Librería para poder utilizar el motor Stepper.

**Int CameraIrPin = 13;** Asignas la colocación del Led infrarrojo para el control remoto en la placa Arduino. En este caso asignas la colocación en el Pin 13 de la placa.

**Int potos = 50;** Número de fotos que quieres realizar.

**Int distance = 800;** Distancia asignada al desplazamiento de la estructura, es decir, la distancia entre foto y foto.

**Int shuterTime = 200;** Velocidad de transferencia para efectuar el disparo con la cámara.

**AccelStepper stepper (1,9,8) ;** Pines asignados al motor Stepper.

**Void setup ()** : Para iniciar librerías y efectuar el arranque, en este caso del motor. Accionamiento de lo que hay entre { } a continuación.

{ **Stepper.setMaxSpeed(500);** Velocidad máxima asignada.

**Stepper.setAcceleration (50);** Aceleración asignada.

**Stepper.move (distance);** Pasos que debe hacer el motor para efectuar la distancia asignada anteriormente en los intervalos.

**Serial.begin (9600);** 9600 bits por segundo es el valor predeterminado para la placa Arduino.

**PinMode (CameraIrPin, OUTPUT);** } Salida del emisor.

**Void loop ()** : Repetición de lo que hay entre { } a continuación.

**If (stepper.distanceToGo () ==0 ) {** : Si el stepper recorre la distancia asignada anteriormente pasará a realizarse lo que hay entre { } a continuación.

**CameraSnap (CameraIrPin) ;** Preparación para efectuar el disparo de cámara. Preparación.

**Stepper.move (distance) ;** Avisa cuando el Stepper ha finalizado la distancia asignada en el intervalo y así poder efectuar el disparo.

**Delay (shuterTime) ;** Tiempo de disparo asignado en el intervalo.

**Serial.println (potos) ; }** Número de fotos asignadas en los intervalos.

**If (photos == 0) {** : Empieza con 0 fotos.

**Potos = 50+1;** Efectuar 51 disparos.

**Stepper.moveTo (0) ; }** : Cuando el stepper llega a 0 se efectúa el disparo.

**Stepper.run ();** Una vez se efectúa el Void loop el stepper vuelve al Void setup hasta que se hacen las 50 fotos.





traveling\_continuo \$

```
#include <nikonIrControl.h>
#include <AccelStepper.h>

int CameraIrPin = 13;           // LED connected to digital pin 13
int photos = 50;
int distance = 800;
int shuterTime = 200;

// Define a stepper and the pins it will use
//AccelStepper stepper (1,3,2); // Defaults to AccelStepper::FULL4WIRE (4 pins) on 2, 3, 4, 5
AccelStepper stepper (1,9,8); // Defaults to AccelStepper::FULL4WIRE (4 pins) on 2, 3, 4, 5

void setup()
{
  // Change these to suit your stepper if you want
  stepper.setMaxSpeed(500);
  stepper.setAcceleration(50);
  stepper.move(distance);

  Serial.begin(9600);

  pinMode(CameraIrPin, OUTPUT);

  //stepper.setCurrentPosition (0);
}

void loop()
{
  if (stepper.distanceToGo() ==0 ) {

    cameraSnap(CameraIrPin);

    photos --;
    stepper.move(distance);
    delay(shuterTime);
    Serial.println(photos);

  }
  if (photos == 0 ) {
    photos = 50+1;
    stepper.moveTo(0);
  }

  stepper.run();
}
```

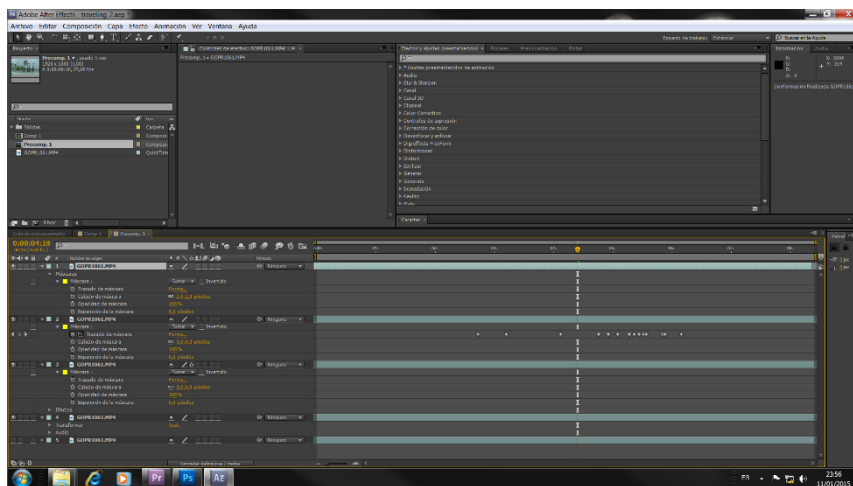


## 4.2. Montaje y edición de vídeo.

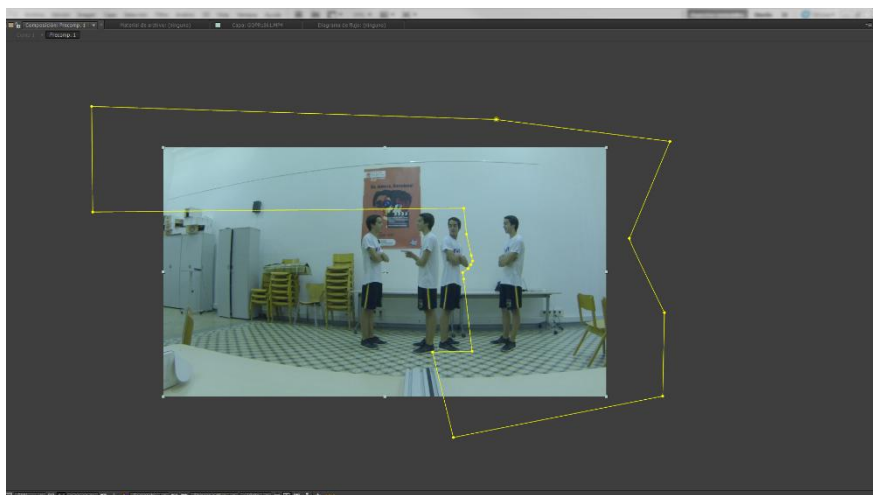
Esta es la parte con la que estoy más familiarizado. Me gusta y estoy bastante cómodo a la hora de realizar el montaje. He utilizado el Adobe After Effects FX 5.5, no es la primera vez que lo utilizo.

Para realizar este efecto de duplicación tuve que grabarme cuatro veces, es decir este montaje está formado por cuatro vídeos. Cada vídeo es el recorrido entero del Slider que me graba de una forma distinta en cada traveling (viaje), así formando cuatro vídeos en la misma posición de cámara, mismo desplazamiento de cámara pero con un personaje cambiante, en este caso yo.

Si juntamos los cuatro vídeos y recortamos tres de estos vídeos el personaje creando máscaras y las pegamos en el cuarto vídeo y hacemos transparente el espacio de los personajes recortados quedarán encima del cuarto vídeo creando una ilusión de cuatro objetos iguales haciendo movimientos distintos en el mismo espacio.



Montando vídeo con el After Effects. Unión de los cuatro vídeos.



Pegando la máscara. (Recuadro amarillo).

## **5. RESULTADOS Y CONCLUSIONES**

### **5.1. Estructura.**

A primera vista el hecho de montar el raíl no parecía complicado, pero cuando me puse a construirlo me di cuenta de que no era tan fácil. A la hora de atornillar tuve que emplear mucha fuerza ya que las piezas de plástico que tenían que ir colocadas en los extremos del raíl estaban hechas por impresoras 3D lo cual quiere decir que no eran exactas y debía hacerle el carril de rosca. Finalmente logré atornillar los cuatro tornillos y colocar la correa. En esta fase no tuve problemas, solo dificultad en realizarla.

### **5.2. Electrónica.**

Esta fase al contrario de la construcción de la estructura pensaba que iba a ser muy difícil, pero me equivocaba, es muy sencillo pero debes ir con mucho cuidado. Principalmente busqué un esquema supervisado por el profesor para que me diera el visto bueno y empezar a experimentar. Una vez encontrado me dispuse a montar pieza por pieza, cable por cable en una protoboard. Conecté Arduino al ordenador, a Arduino lo conecté en la protoboard donde estaba el microcontrolador A4988, el motor Stepper lo conecté en dicha placa experimental junto con la fuente de alimentación. Todo colocado siguiendo los pasos que me marcaba el esquema del circuito buscado.

Pero me faltaba una cosa, la corriente. Aprendí a puentear una fuente de alimentación para que funcionara sin necesidad de estar dentro de la placa base del ordenador.

Cuando estuve sin supervisión de nadie fue cuando más experimenté. Fundí 3 Drivers a causa de la fuente de alimentación que tenía un voltaje muy elevado, la cambié y todo funcionaba correctamente, excepto el funcionamiento del motor, al ser un motor paso a paso, tenía el error que se saltaba algunos pasos y no lograba dar la vuelta del todo provocando que no se moviera la estructura de plástico. También hacía un ruido extraño y no tenía la suficiente fuerza para desplazar la cámara ya que pesaba bastante.

La parte de la electrónica, el circuito, estaba correctamente. Revisé el circuito que estuviera todo bien colocado, lo desmonté y lo volví a montar pero el funcionamiento del motor seguía igual. Hasta que me puse a investigar dentro del Software.

Esta fase fue la más divertida, pero más frustrante de todas a causa de los problemas citados anteriormente.

### 5.3. Software.

Desde un principio pensaba que el software era la parte más difícil de todas porque su contenido es muy extenso y apenas tenía conocimiento sobre él ya que nunca había programado nada.

A medida que mi compañero programaba me iba explicando para que servía cada palabra que escribía, poco a poco iba entendiendo las funciones de cada cosa. A los pocos días de introducirme en el Software cogí mi ordenador y me dispuse a programar por mi propia mano, sin ayuda de nadie, solo con ayuda de lo que me explicó mi compañero Alejandro.

Una vez terminé el código tuve varios problemas ya que el software no se ejecutaba correctamente. Investigué en foros de internet y acabé logrando que funcionara, era un error de librerías. Estas librerías son necesarias ya que contienen el código fuente, en este caso el de Stepper y el controlador de Nikon, necesarias para poder ejecutar la acción, es un referente.

Cuando el curso terminó y me llevé el proyecto a casa, lo puse en marcha y el motor no funcionaba bien. Revisé que todo estuviera correcto y me puse a investigar dentro del código, el Software. Estuve mucho rato revisándolo, y todo parecía estar bien, hasta que me puse a probar cambiando los intervalos, el delay, la aceleración y la velocidad máxima. El problema persistía en la aceleración y la velocidad máxima, al ponerle un miembro mayor logré que el motor funcionara correctamente sin ningún ruido extraño o temblor. Pero aquí no acababa todo.

A la hora de poner la cámara el motor no pudo ejercer su función debido al peso, no tenía suficiente fuerza para ejecutar todos los pasos. Pensaba que era un error del motor, pero me equivocaba, el error era el driver, estaba estropeado. Entonces compré uno más eficaz, pero más complicado de montar en la protoboard. El resultado fue increíble, el motor tenía mucha fuerza con este driver llamado A4988. Entonces puse la cámara y ya no hacía ruidos extraños, se desplazaba correctamente y no se saltaba los pasos.

## 5.4. Montaje y edición de vídeo.

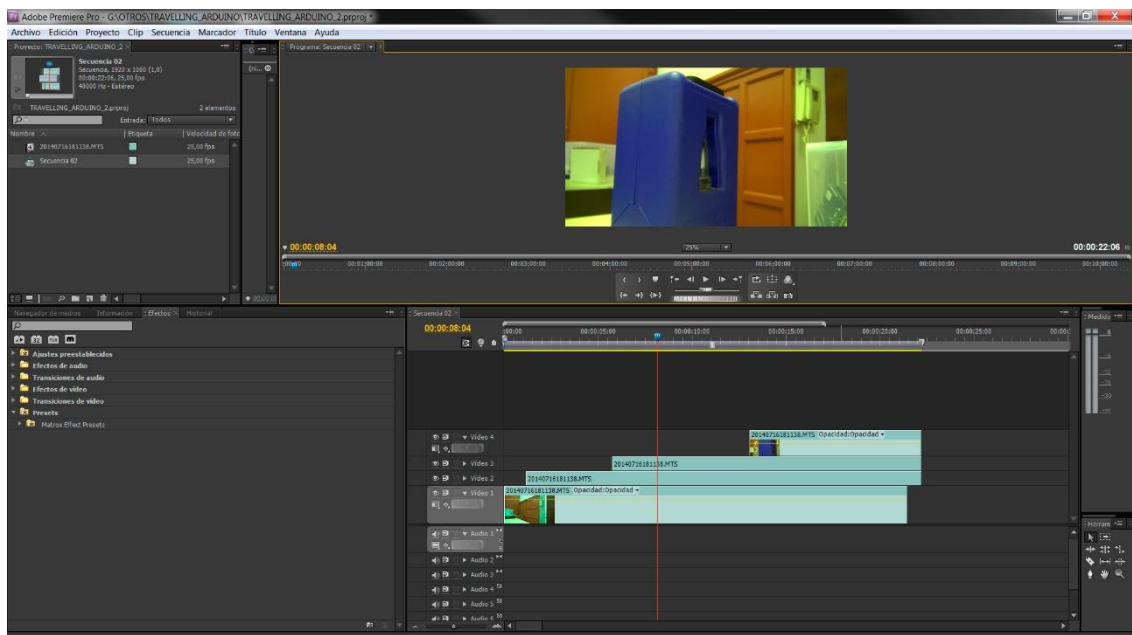
El primer proyecto que hice con Slider fue bastante sencillo. Consistía en la aparición de objetos mientras la cámara se desplazaba hacia a estos objetos. Para lograr la aparición de estos objetos tuve que grabar tantas veces como objetos quería que saliesen. En este caso quería que apareciesen tres objetos, por lo tanto grabé cuatro travelings. El primer traveling no había ningún objeto, el segundo viaje colocaba el primer objeto, en el tercer viaje colocaba el segundo objeto y en el cuarto viaje el tercer objeto.

Los cuatro vídeos los pasé al ordenador y los envié al Adobe Premiere, un programa de edición de video, lo único que debía hacer era cortar los videos y escalonarlos ya que la posición de la cámara no cambiaba ni tampoco variaba la velocidad. Pero me surgieron unos problemas.

El primer problema fue la luz, entre el primer viaje y segundo hay un cambio de luz debido al fluorescente situado encima de la sala y a la colocación de la cámara respecto a él.

El segundo problema es que en el primer vídeo aparece un chico y al cortarse el video y enlazarse con el siguiente desaparece dicho hombre ya que en el segundo vídeo no salía.

Son fallos de principiante, pero que para la próxima vez tendré en cuenta. No volví a hacer más proyectos, pero tengo unos cuantos en mente que entregaré más adelante.



Montaje de video con el Adobe Premiere. Aparición de objetos mientras la cámara se desplaza.

## **6. CONCLUSIÓN DEL TRABAJO.**

A lo largo del trabajo he tenido altos y bajos momentos, en los que o he estado muy motivado o con ganas de dejarlo debido a la desmotivación. Pero aun así nunca me he dado por vencido y he aprendido a no rendirme y a que el esfuerzo siempre acaba dando sus frutos.

El Slider empecé a desarrollarlo en verano y la verdad que fue bastante desmotivador el hecho de trabajar en vacaciones, también fueron unos meses de estudio para las recuperaciones de Septiembre y era todo muy estresante, pero al final conseguí adaptarme y convertirlo en una rutina.

El proyecto la verdad que me captó mucho la atención y creo que es una de las cosas más interesantes y entretenidas que he hecho hasta el día de hoy.

Gracias a este trabajo he logrado realizar todos mis objetivos, he aprendido el funcionamiento de los motores bipolares, he logrado entender en profundidad que es el Slider, he aprendido a programar, a clasificar mi redacción y a redactar mi memoria del trabajo, he construido mi propio Slider, y he logrado que funcione dándole órdenes a través de mi placa Arduino, he montado y añadido efectos de vídeo y fotografía, he interactuado con el mundo físico/virtual, he asistido a un curso de 50 horas para desarrollar el proyecto Slider, también he logrado conseguir el efecto óptico deseado con la cámara y he aprendido a soldar.

En conclusión he aprendido muchísimo sobre programación, cámaras, electrónica, mecánica y a saber resolver problemas que iban surgiendo por mis propios medios.

Podría decir que ahora lo veo todo desde otro punto de vista, desde un punto de vista más ocurrente. Es decir que todo funciona por un motivo y por un propósito.

## **6. AGRADECIMIENTOS.**

Principalmente doy las gracias a mi Tío que me ha ayudado muchísimo y me ha inspirado a lo largo del proyecto. También doy las gracias a mi tutora del trabajo Maite Díaz-Gonzalez, profesora de Tecnología, por haberme facilitado las cosas en el trabajo escrito y por haberme motivado a no parar y a no rendirme. Finalmente agradezco la ayuda de Raúl, mi profesor del curso de Slider, por haberme ayudado en las complicaciones de software, doy las gracias a mi familia por haber estado ahí colaborando y ayudándome cuando lo necesitaba. También agradezco la información obtenida por los compañeros que tuve en el curso de Slider, en especial a mi compañero Lucas, que me enseñó gran parte de la programación, a construir circuitos, a puentear y me enseñó la utilidad de las librerías.

## **7. BIBLIOGRAFÍA.**

-<http://arduino.cc/> (Durante todo el trabajo).

-<http://es.wikipedia.org/> (Durante todo el trabajo).

-<http://www.vonroth.com/Arduino/NikonIrControl/> (Durante el curso. Mes de Julio)

-[http://www.airspayce.com/mikem/arduino/AccelStepper/AccelStepper\\_8h\\_source.html](http://www.airspayce.com/mikem/arduino/AccelStepper/AccelStepper_8h_source.html)  
(Durante el curso. Mes de Julio)

-BANZI, Massimo. Introducción a Arduino, Anaya Multimedia. (Durante el mes de Junio)

## **9. OTRAS FUENTES**

-He asistido a la feria de Arduino en Barcelona el mes de julio, llamada FABLAB.

-He realizado un curso de 50 horas durante el mes de julio.

-He estado yendo a un taller para experimentar y aprender más sobre mi proyecto y otros como por ejemplo sintetizadores, impresoras 3D, sprays con Stepper y Arduinos emitiendo sonido.

