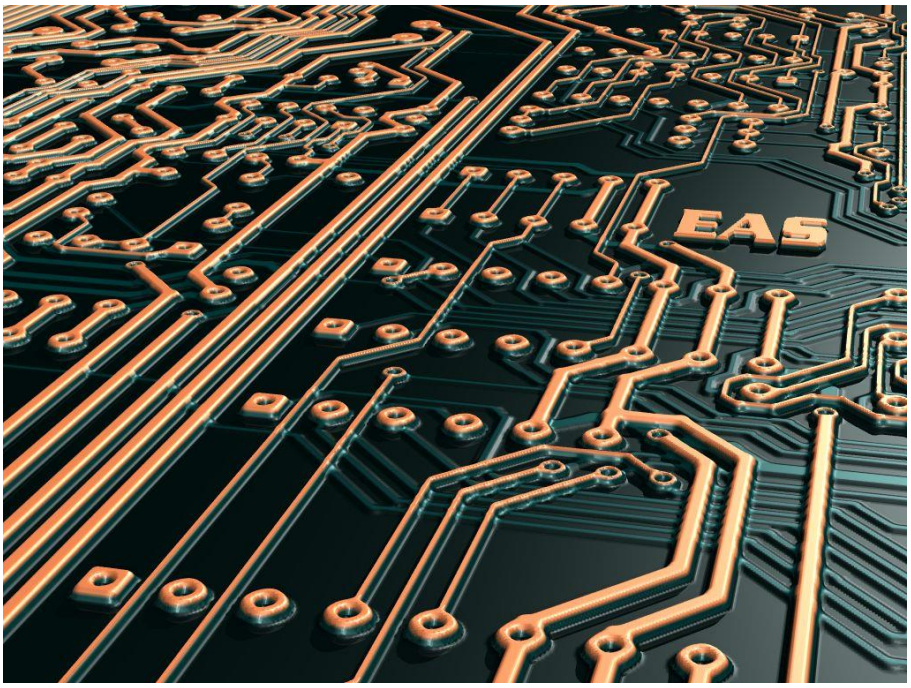


LABORATORIO DE DESARROLLO DE HARDWARE "DISEÑO DE PCB"

MEMORIA

2016 / 2017

Rafael arroyo alemán
Ingeniería Informática de Computadores

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de Ingeniería Informática

Contenido

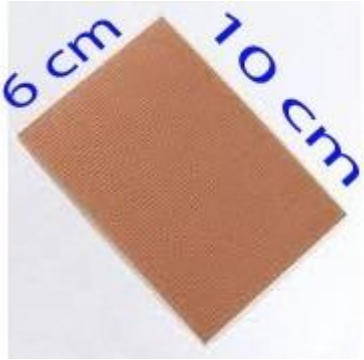
Introducción 2
Descripción de la actividad 5
Descripción del sistema 5
Partes de nuestro sistema 6
Conexión entre el joystick y Arduino 7
Componentes utilizados 8
Elaboración de huellas 10
Instalación Eagle 10
Creando la librería 10
Pulsador 10
Pines Arduino 12
Nuevo Proyecto 14
Creación del esquemático 15
Creación de la Board 17
Generación de los gerbers 19
Revisión de la placa 20
Segundo Diseño de PCB en KIKAD 24
Esquemático 24
Obtención y valoración del listado de los componentes 29
Archivo de enlace de redes y enrutado 32
Gerbers 35
Programación de la placa Arduino como dispositivo HID para mando de juegos y conexión a móvil Android 37
Jugar en móvil 41
Eurocircuits, proceso de encargo a fabricación 43

Introducción

Durante el desarrollo de esta práctica procederemos a diseñar una Placa de circuito impreso “PCB”.

La PCB, es una superficie la cual está constituida por caminos, pistas o buses de material conductor laminadas sobre una base no conductora, el circuito impreso se emplea para conectar eléctricamente a través de pistas conductoras y sostener mecánicamente, por medio de la base, los componentes electrónicos.

Las pistas normalmente suelen ser de cobre, denominado un buen conductor, y la base se suele fabricar en materiales como fibra de vidrio, Pertinax, Cerámica, plástico, teflón o polímeros como la baquelita, o placas flexibles dependiendo el tipo de uso para el que se vaya a aplicar.



Aquí vemos un ejemplo del tipo de placa que emplearemos para el desarrollo de la práctica, de una base de Baquelita.

Una vez que hemos elegido la base de nuestro proyecto se procederá a hacer el estudio del sistema.

El sistema a desarrollar será la de una Shield para Arduino. La Shield es una placa de expansión que se colocará sobre los pines de nuestra Arduino la cual cuenta con una funcionalidad ya preparada para hacer pruebas sobre ella.

El diseño electrónico de dicha placa se ha realizado sobre el Programa para desarrollo **MultiSim** en el cual se pueden realizar estudios eléctricos con **Simulación.**

Una vez que tenemos realizado nuestro sistema procederemos a realizar el diseño de nuestra PCB sobre un programa informático CAD para el desarrollo de circuitos impresos el cual nos permitirá plantar nuestro **esquemático y** el conexionado eléctrico de nuestros componentes a través de pistas trazadas en el cobre.

Bien nuestro programa se llama **EAGLE** la cual pese a ser una herramienta de pago,

dispone también de licencias de uso gratuitas, pero con **restricciones "Limite de uso de capas …"** pero para el caso nos valdrá puesto que nuestro diseño se realiza a una cara.

También se aplicará el desarrollo de dicha Shield en otro programa CAD, este programa se llama **KICAD.**

Que a diferencia de **Eagle** es una herramienta **OpenSource** sin costo alguno, mantenida por la comunidad y que dispone de casi las mismas funcionalidades que otras herramientas de pago.

Bien una vez trazado el esquemático y nuestro layout "traza de pistas sobre la PCB" se generarán los archivos gerbers que nos interesan para nuestra placa "La cual se realizara como **NAKED PROTO**" y el archivo de taladrado, tras esto se introducirán dichos archivos en nuestra máquina de prototipado **Protomat S62.**

Una vez que hayamos fabricado la placa y hallamos ensamblado nuestros componentes se realizará la programación del nuestra Arduino, en concreto el **chip 16U2** que nos hace la conversión de USB-Serial para que nos reconozca como

dispositivo HID como mando de juegos, para nuestro caso utilizaremos el proyecto **UnoJoy.**

En dicha memoria también se incorpora el proceso que hay que realizar para realizar la fabricación de nuestro diseño PCB en una empresa dedicada a la fabricación de placas de circuito impreso.

En nuestro caso optamos por **Eurocircuits:**

la cual se caracteriza por precios competitivos y la capacidad de chequear online nuestro diseño mediante un .Zip con nuestros archivos para así poder darnos el precio de nuestro pedido, así como posibles infracciones de diseño que se puedan dar a la hora de la fabricación.

Descripción de la actividad

Descripción del sistema

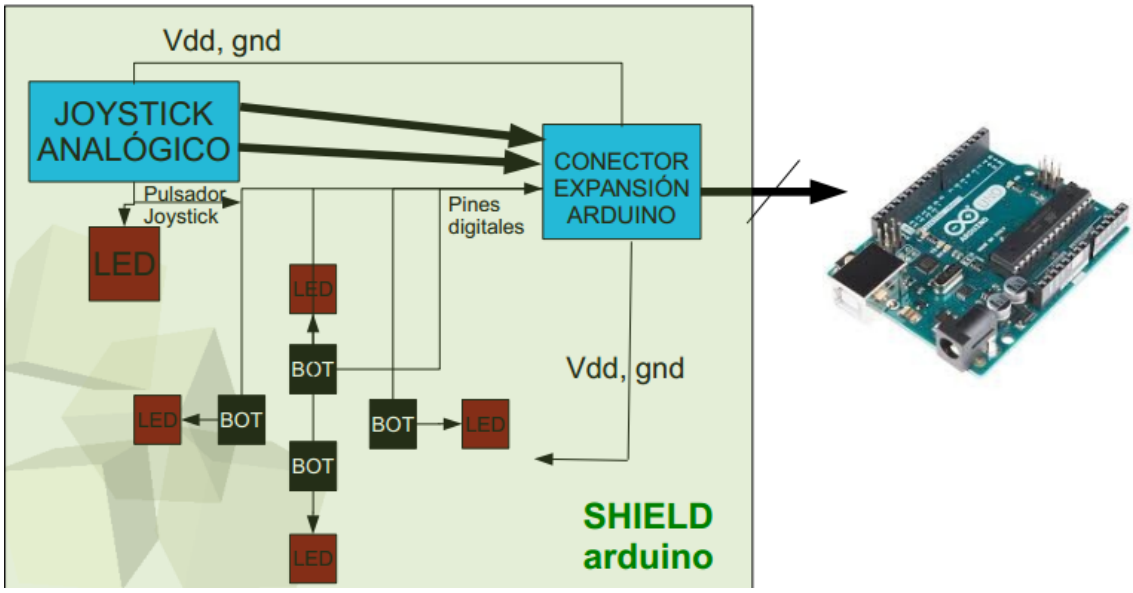
En nuestro desarrollo de la práctica, realmente se comenzaría haciendo el diseño del sistema electrónico y su emulación para así comprobar que se cumple con lo que se quiere alcanzar:

En nuestro caso se nos dota del circuito a realizar ya desarrollado:

En la figura de abajo se muestra la especificación de nuestro sistema electrónico en el cual se puede observar un **joystick analógico**, junto con un led que se iluminará cuando pulsemos el botón central, que ira directamente conectado a las entradas analógicas.

Y 4 botones con sus respectivos leds los cuales se iluminarán cada vez que se pulsen cualquiera de ellos, estos irán conectados a los pines digitales.

La Shield estará conectada con una alimentación de 3.3V para los leds y 5V para el joystick



Partes de nuestro sistema

Tendremos:

- **Joystick Analógico**: Se encuentra alimentado a 5V. Nos da señal analógica de los potenciómetros en los ejes X e Y, estos irán conectados a las entradas analógicas de nuestra Arduino.
  También dispone de un interruptor para su pulsación central el cual si es pulsado nos dará toma a GND, ira conectado a un pin digital de Arduino.

- **Pulsadores:** Dispondremos de 4 pulsadores que irán conectados también a pines digitales.

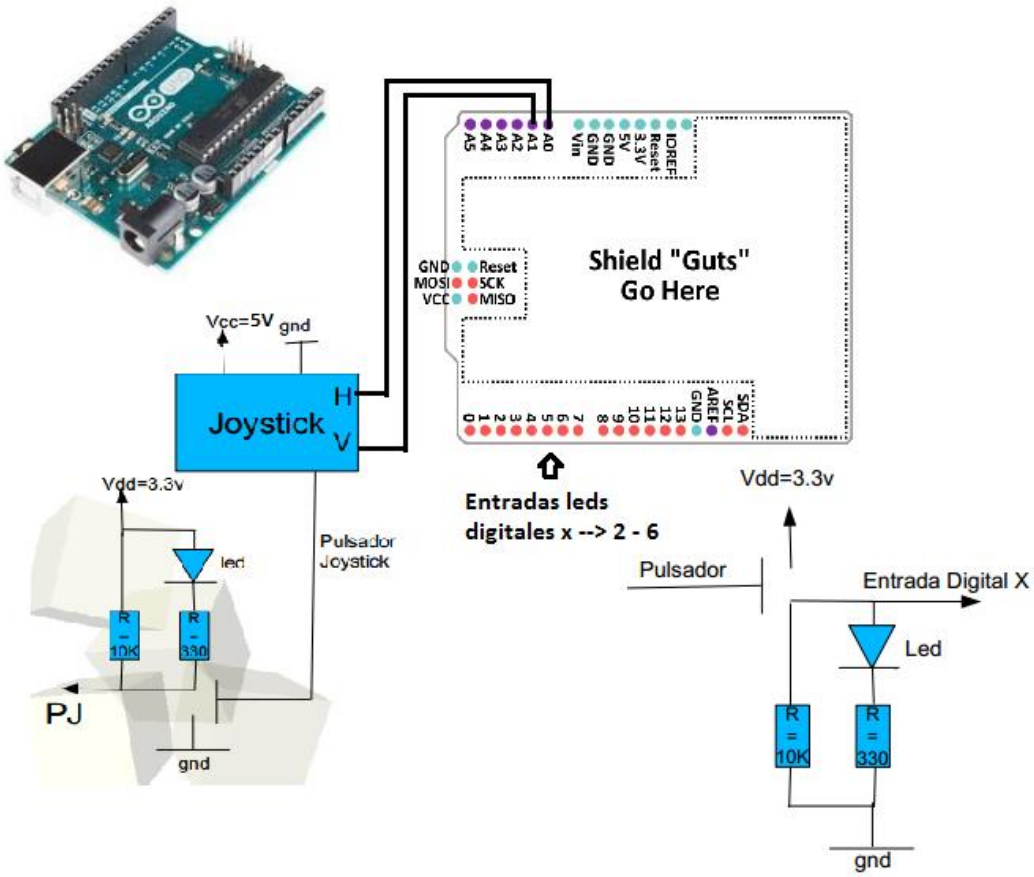
- **Leds:** Dispondremos de 5 Leds, de los cuales irán conectados 4 a los pulsadores con sus respectivas resistencias instaladas en disposición Pull-Down y un último Led instalado para mostrar cuando se pulsa el botón central del joystick en disposición Pull-Down también.

- **Pines de conexión:** Son las encargadas de la conexión que se realizará entre la Shield y Arduino.
  5 pines para señales digitales, 2 pines para entradas analógicas y 3 pines para alimentaciones y tierra.

Conexión entre el joystick y Arduino

En esta imagen tenemos lo que va a ser un esquema de conexionado de nuestra Shield con Arduino, básicamente lo que vemos es las salidas del joystick conectadas a las entradas analógicas A0 y A1 para recoger los datos de las posiciones del cursor en su eje X e Y junto con la conexión del Led del botón central el cual se encenderá cuando se pulse registrando un 0 en la entrada digital 6 en nuestro caso.

Y por otro lado el esquema de los botones con su respectivo Led instalados con el mismo esquema eléctrico, pero dando lugar a que cuando se pulse el botón se encienda el led registrándose un 1 en la entrada digital.



Componentes utilizados

■ **Joystick:** Thumb Joystick Sparkfun

Su precio rondaría los 3,70€ la unidad

Se dispone en la librería de Eagle **Sparkfun-Electromechanical** la huella del componente

■ **Pulsadores**: Momentary Pushbutton Switch - 12mm
Square

Precio unos 0.61€ la unidad

También se dispone en la librería de Eagle la huella del componente en Sparkfun.lbr aunque en nuestro la hemos hecho nosotros.

■ **Resistencias de 10K y 330oh**: SMD 1206

Se compran por carretes o lote. Un lote de 25 unidades saldría más o menos por 0.33€

Existen huellas adecuadas en las librerías de EAGLE. Importante coger el tamaño adecuado “1206, 0805 …”.

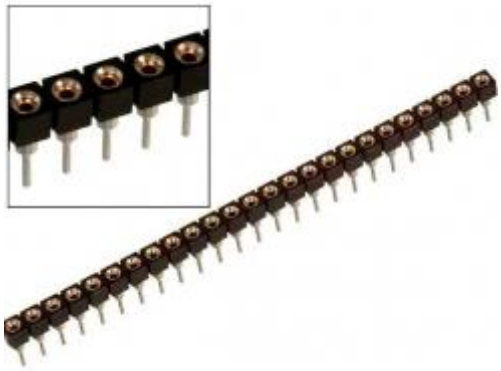
■ Tiras de pines:

- Conector estándar: Se compran por tiras unos 1.48€
  → 4x1, 5x1

Distancia entre pines: 0,1inch.

- Diámetro Pad: 0.066 – 0.07 inch
- Diámetro agujero (drill): 0.03543 inch (0,9mm)

También se ha realizado la huella manualmente.



■ Leds: Leds TH. De 3mm → A unos 0.30€ la unidad

- Distancia entre pines = 2,54 mm (0,1inch)
- Diámetro Pad=1,88mm(0.074inch)
- Diámetro Drill= 0,8mm (0.032 inch)
- Existen componentes en la librería LED de Eagle

Elaboración de huellas

Como se ha mencionado antes, se han realizado las huellas de las tiras de pines y de los pulsadores.

Primeramente, se hizo el componente del pulsador para ello se tuvo que instalar el programa de Eagle.

Instalación Eagle

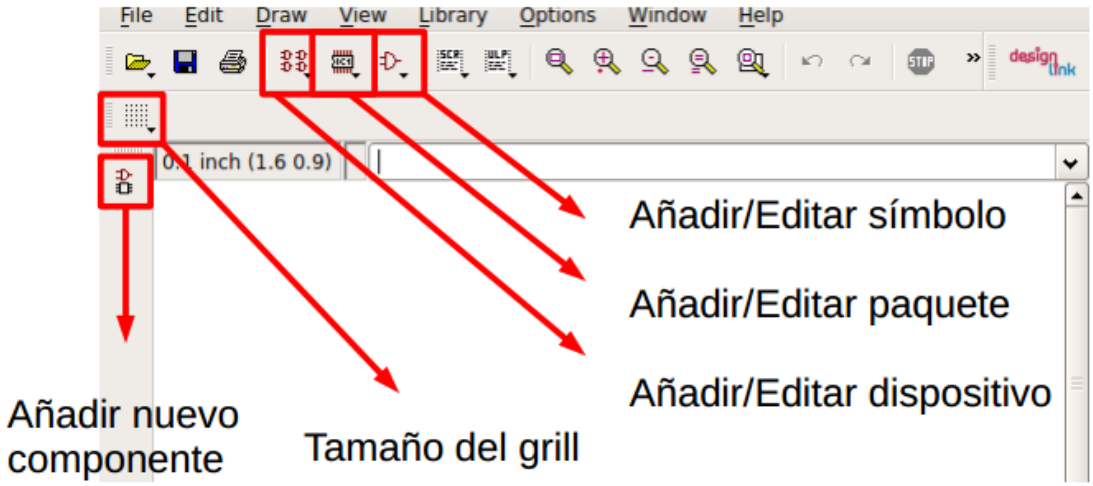
Desde la página de http://cadsoft.io/ se realiza la descarga del programa y cuando vayamos a realizar la instalación lo ejecutaremos con una licencia restringida de Eagle Express.

Creando la librería

Una vez que ejecutamos el programa

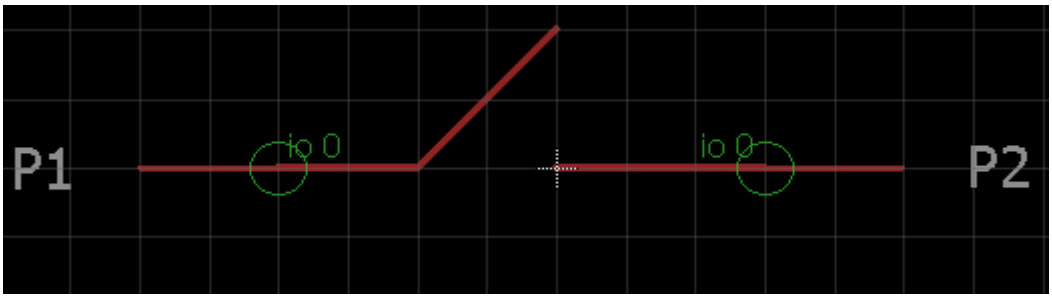
EN File>New>Library abriremos el editor de nueva librería

- Una vez allí nos encontraremos esta pantalla



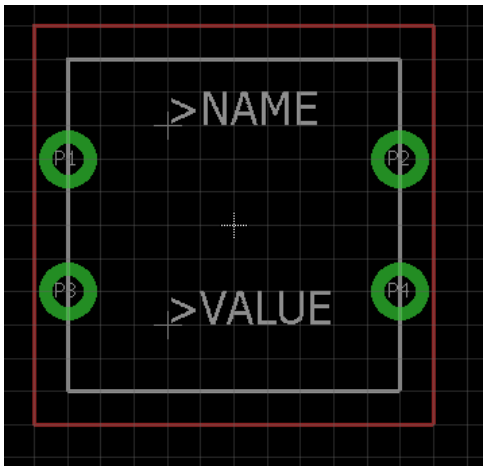
Pulsador

- Guardaremos nuestra librería y le damos el nombre que sea y la activamos dándole al **círculo verde** para que se encienda.
- Para ello se hizo el diseño primeramente del **símbolo**.



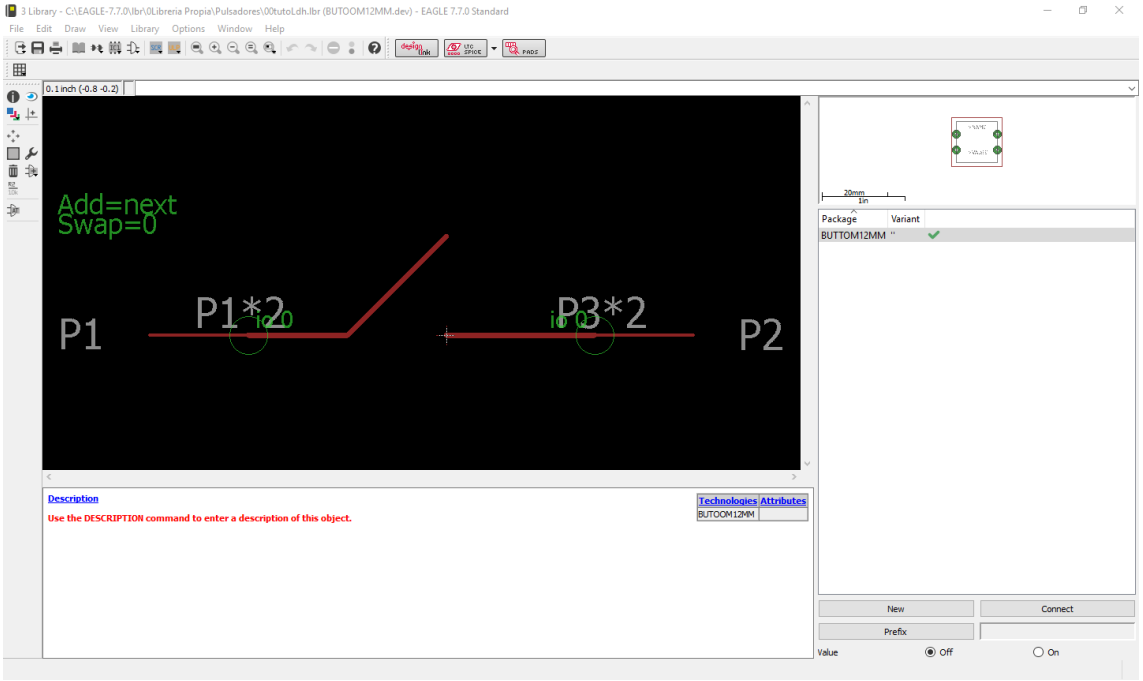
En ella se puede observar P1 y P2 como las entradas y/o salidas del pulsador para conectar en el esquemático y las líneas rojas que serán representativas en el esquemático.

Y la **huella**:



Donde vemos los Pads representados en verde junto con una línea gris que representa lo que ocupa el componente y la línea roja keep-out como parámetro para mantener distancia con otros componentes, los >Name y >Value serán usados para e el esquemático dar valores y nombres representativos a éste.

El **dispositivo**:
Por ultimo hablar del dispositivo que es la unión del símbolo con la huella:

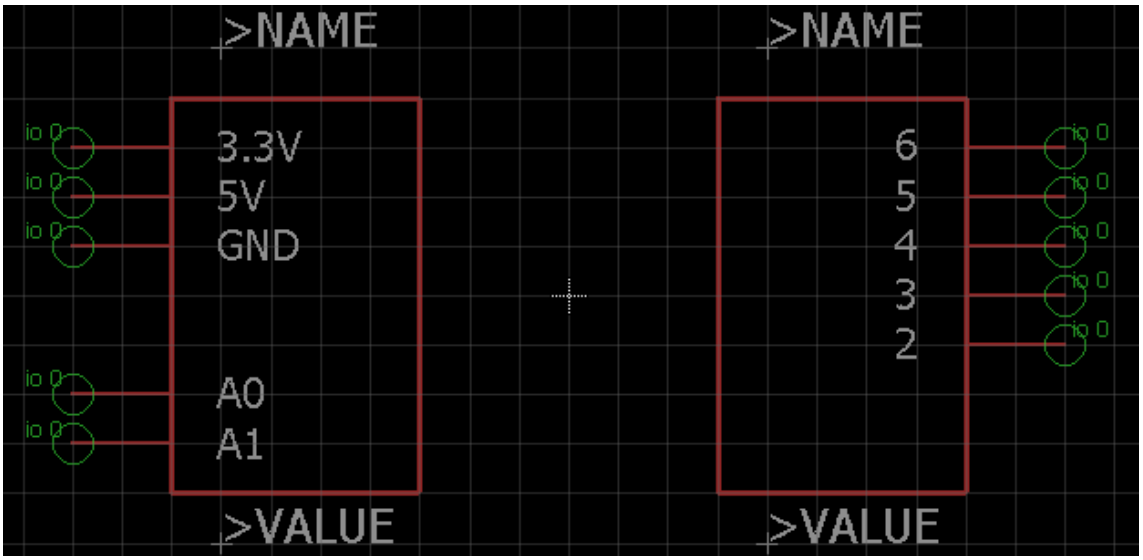


Aquí se realiza el conexionado de la huella y del símbolo para adjudicar que entrada y/o salida se relaciona con que Pad de la huella.

Pines Arduino

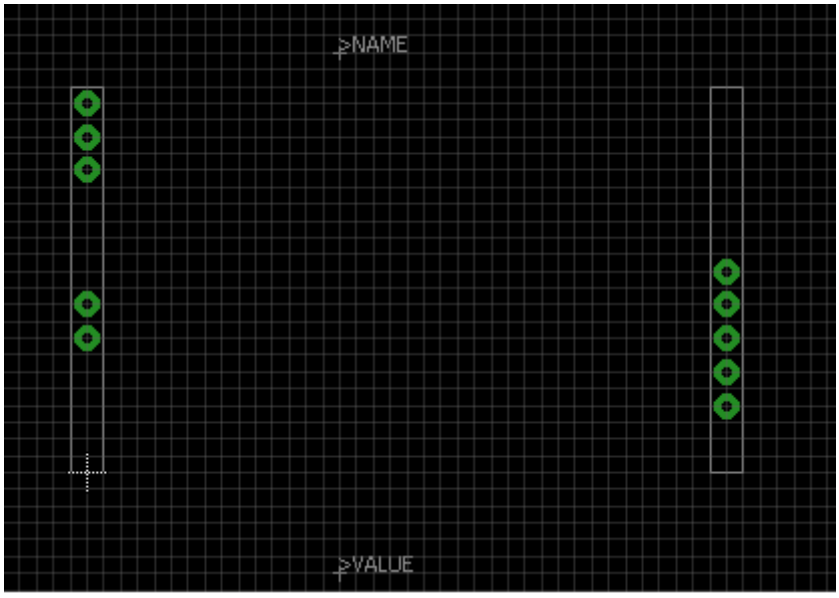
También se realizó el componente de los pines de conexión de Arduino con la Shield.

Se realizó el **símbolo**.



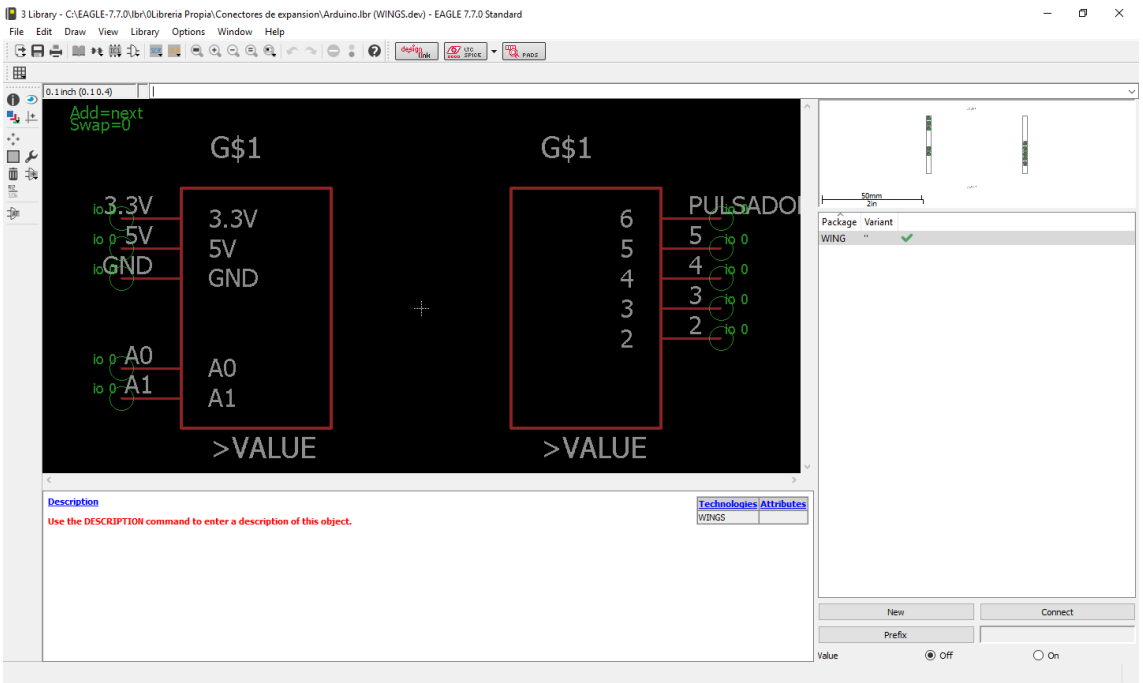
Aquí se puede observar las **entradas analógicas** A0 y A1 de Arduino los **pines de alimentación** y los **pines digitales** de Arduino 2 – 6.

La **huella**:



Se pueden apreciar los **Pads** en verde y la separación de 4,8 cm de los pines de izquierda a derecha.

Por último, **el device**, la asociación de las entradas/salidas del símbolo con la huella correspondiente.

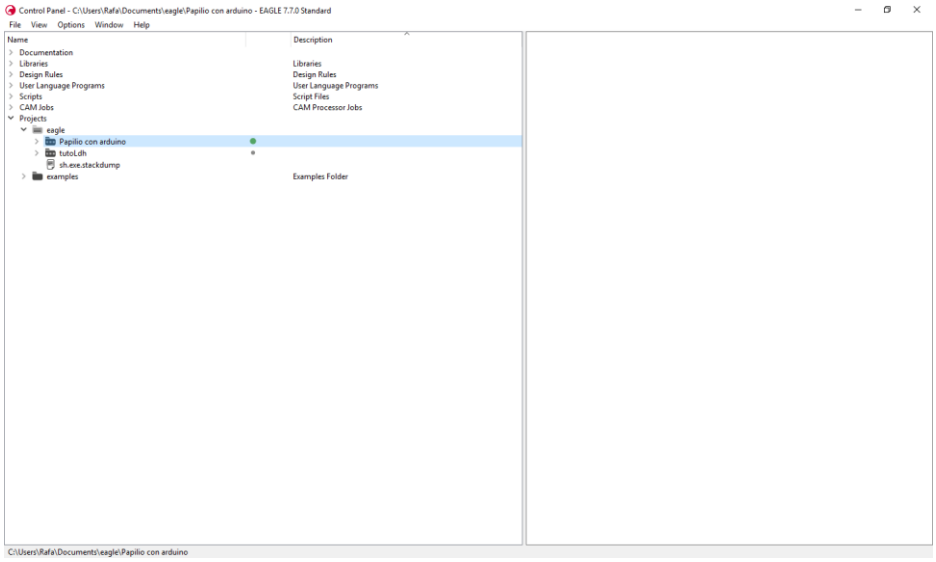


Nuevo Proyecto

En panel de control seleccionaremos la pestañita de projects y seleccionaremos la carpeta donde crearemos el proyecto.

Seleccionada la carpeta, botón de la derecha del ratón y New project. Cambiaremos el nombre y tendremos creado el proyecto, inicialmente vacío.

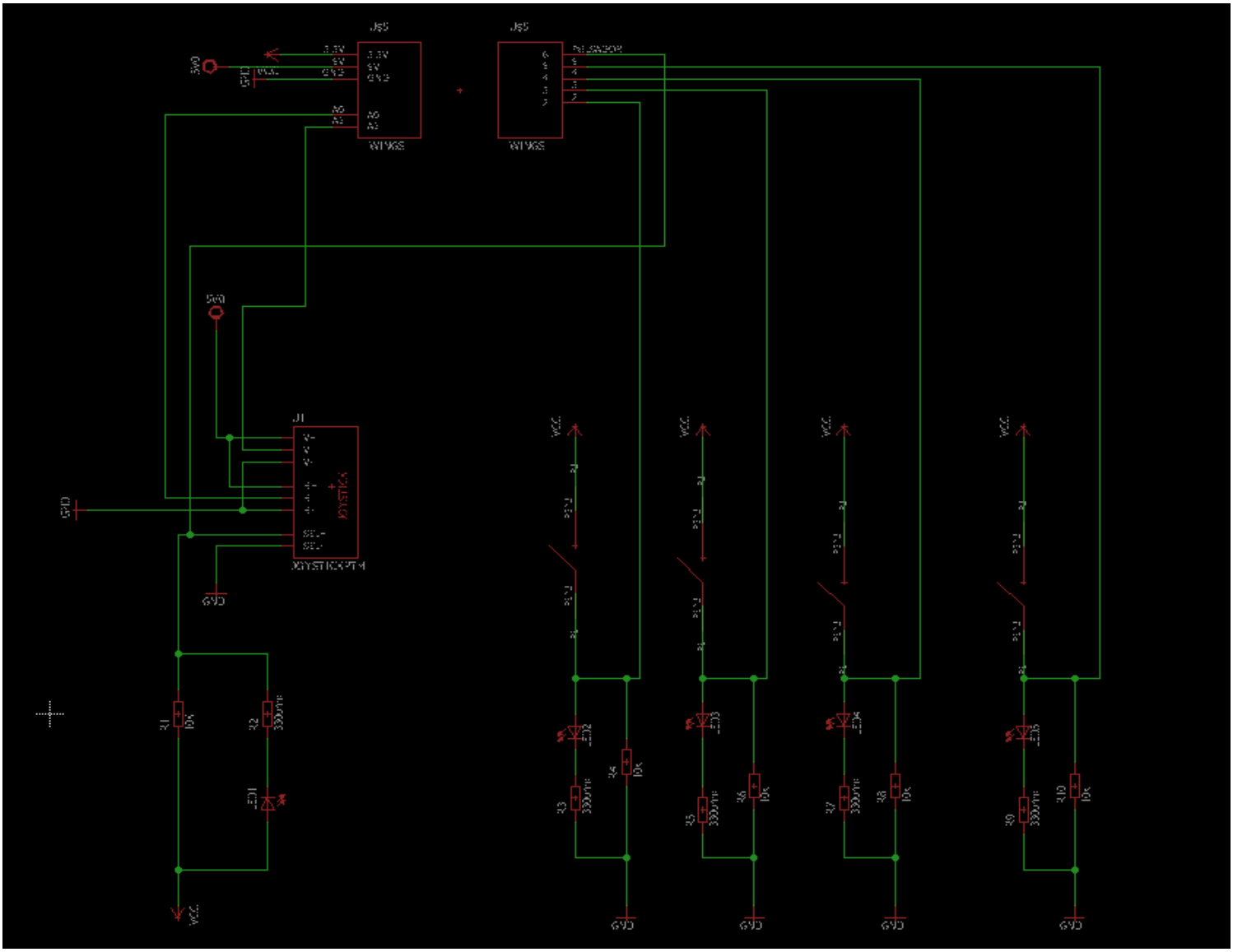
- Modificaremos la carpeta de proyectos:
    - En la pestaña Options → Directories y modificar/añadir nueva carpeta
- Sobre el proyecto nuevo ya podremos crear un nuevo esquemático, layout o incluso una librería.



Creación del esquemático

Ya una vez llegados a esta parte tendremos nuestras librerías echas y nuestros componentes preparados para poder realizar el esquema eléctrico de nuestro proyecto.

1. Nos iremos en nuestro proyecto creado y nos iremos a File >New > Shematic y guardaremos nuestro esquemático.



En la parte izquierda inferior de la imagen se puede observar el conexionado del led al pulsador del Joystick mediante dos resistencias **Pull-Down** una de 330ohm directamente conectada al led y otra resistencia de 1k en paralelo a la del led,

Cuando se pulse se registrará un 0 en el pin digital Arduino.

Por otra parte, el joystick a su vez se observa cómo se encuentra conectado a tierra y los dos potenciómetros del componente las cuales disponen de 3 terminales cada uno.

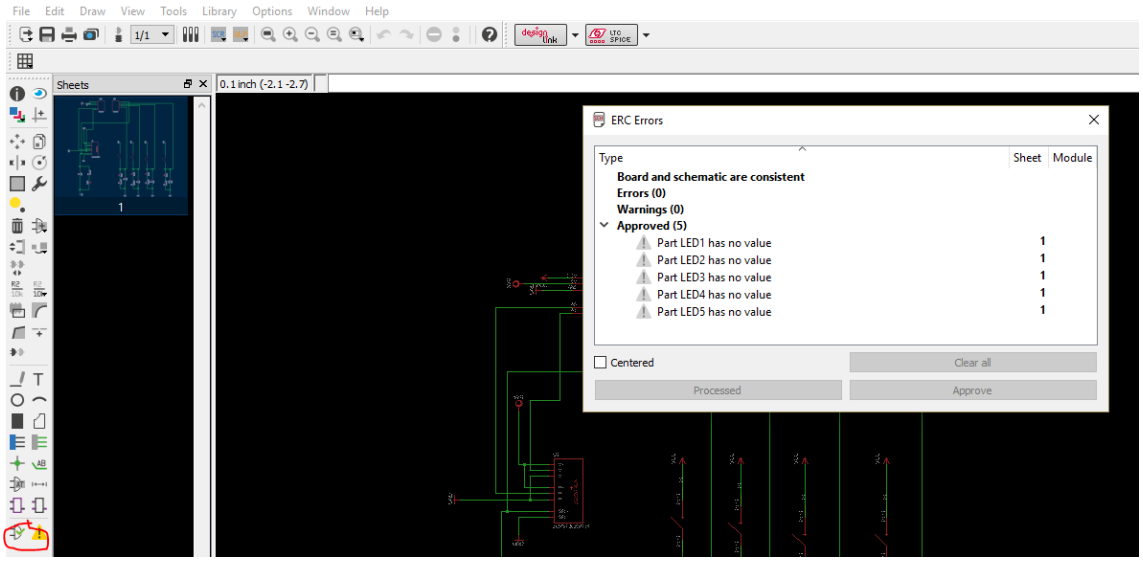
- 1 extremo de ellos que a tierra
- El otro extremo va a 5 voltios
- Y el del medio nos da la diferencia de voltaje según los desplacemos que va conectado a la entrada analógica de Arduino.

En la parte derecha de la imagen tenemos los pulsadores los cuales tienen el mismo sistema eléctrico montado que el led del Joystick, pero con la diferencia de que van conectados a los interruptores los cuales cuando se cierran hacen que llegue a los pines digitales conectados a Arduino un valor lógico 1.

Y por último en la parte superior del esquema se observan los pines de conexión a la placa los cuales en la parte izquierda se encuentran los pines analógicos y las tomas de corriente y tierra.

Y en la parte derecha los pines digitales de los cuatro botones y el botón central del joystick.

Una vez realizado el esquema se pasó el test **ERC** el cual nos dio que no se presentaban problemas en el conexionado.



Lo único que nos aparecieron fueron advertencias en los valores led que no le teníamos asignado nada pero que para lo nuestro no nos interfiere para nada en el funcionamiento.

Creación de la Board

Una vez terminado con el proceso del esquemático podemos empezar a rutar las pistas, para ello la ventana de esquemático la dejaremos abierta ya que, si tenemos que realizar cualquier tipo de modificación sobre el esquemático la haremos en el momento, cuyos cambios se verán reflejados directamente en la pantalla de Layout.

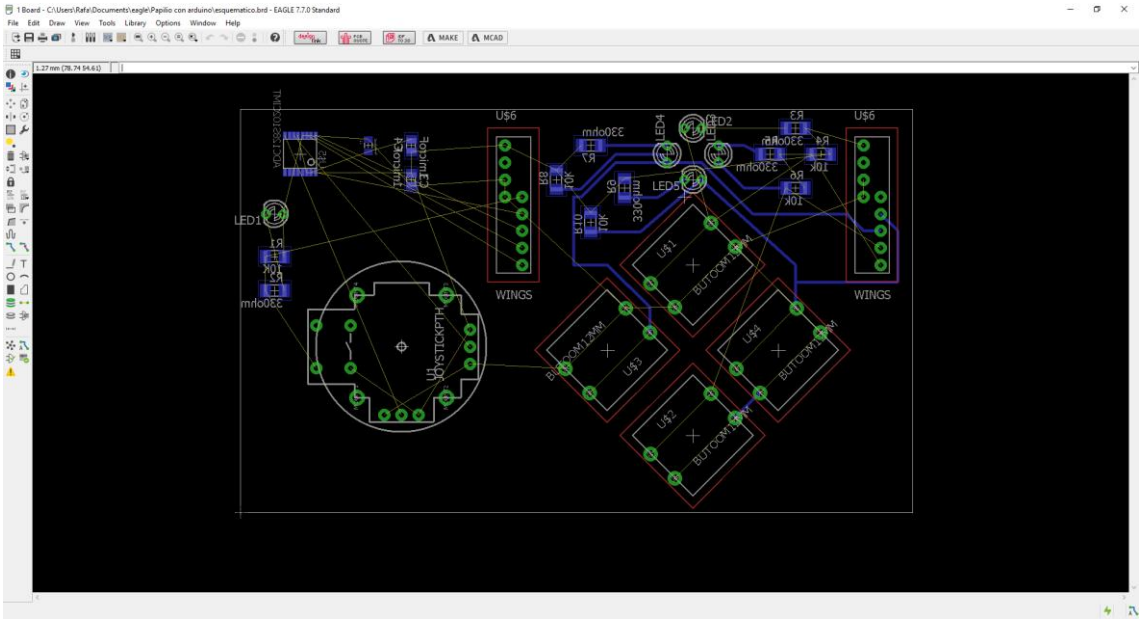
Las características de nuestra Board son:

- Un área de 10cm x 6cm
- Diseño a una sola cara
- Tamaño de la pista de 0.016 inch

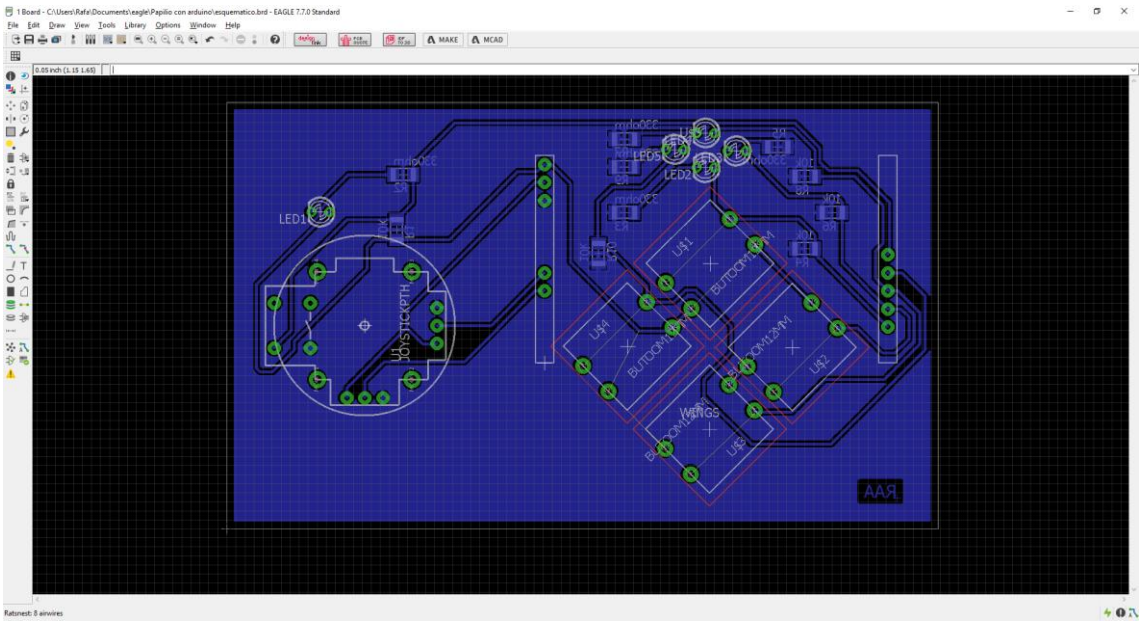
Cuando abramos la Board nos aparecerán los componentes del esquemático en una esquina todos arrinconados y conectados entre sí por los **Rats** que son las dependencias de conexiones entre los diferentes componentes.

Para debemos coger uno a uno los componentes y colocarlos a poder ser de la forma más adecuadamente posible, es decir, de manera que intentemos reducir al máximo el número de cruces entre los Rats y de manera modular colocados para intentar facilitar lo más posible el rutado.

En esta imagen se enseña los componentes colocados y empezándose el ruteado, mencionar que aquí se ha realizado de forma manual, ya que se intentó usar el AutoRoute, pero se quedó en el 90%, por lo que se decidió hacerlo a mano.



Una vez realizado el rutado completo, se trazó el plano de masa para todas aquellas GND que quedaban sin conectar y el grabado de las iniciales en la esquina.



Algunos comentarios respecto a su elaboración:

- Se debe tener cuidado con el isolate que es la distancia entre la pista y el plano de tierra.
- Dejar bien definido el tamaño de la pista
- Trazar las pistas sin dejar ángulos de 45º
- Tener cuidado con las pistas cercanas a los límites de la base, intentar dejar un margen adecuado.
- Dejar suficiente espacio de una pista a otra
- En este caso se hizo el trazado de las pistas sobre el plano de tierra directamente para evitar dejar pines GND aislados
- Emplear de vez en cuando el ratness para actualizar los Rats a su camino más corto.
- Asegurarse de que los componentes están colocados en la capa adecuada.
- Comprobar el DRC metiendo en él los datos de la empresa en la que se va a mandar a fabricar y cerciorarse de que cumple los requisitos de fabricación.

Generación de los gerbers

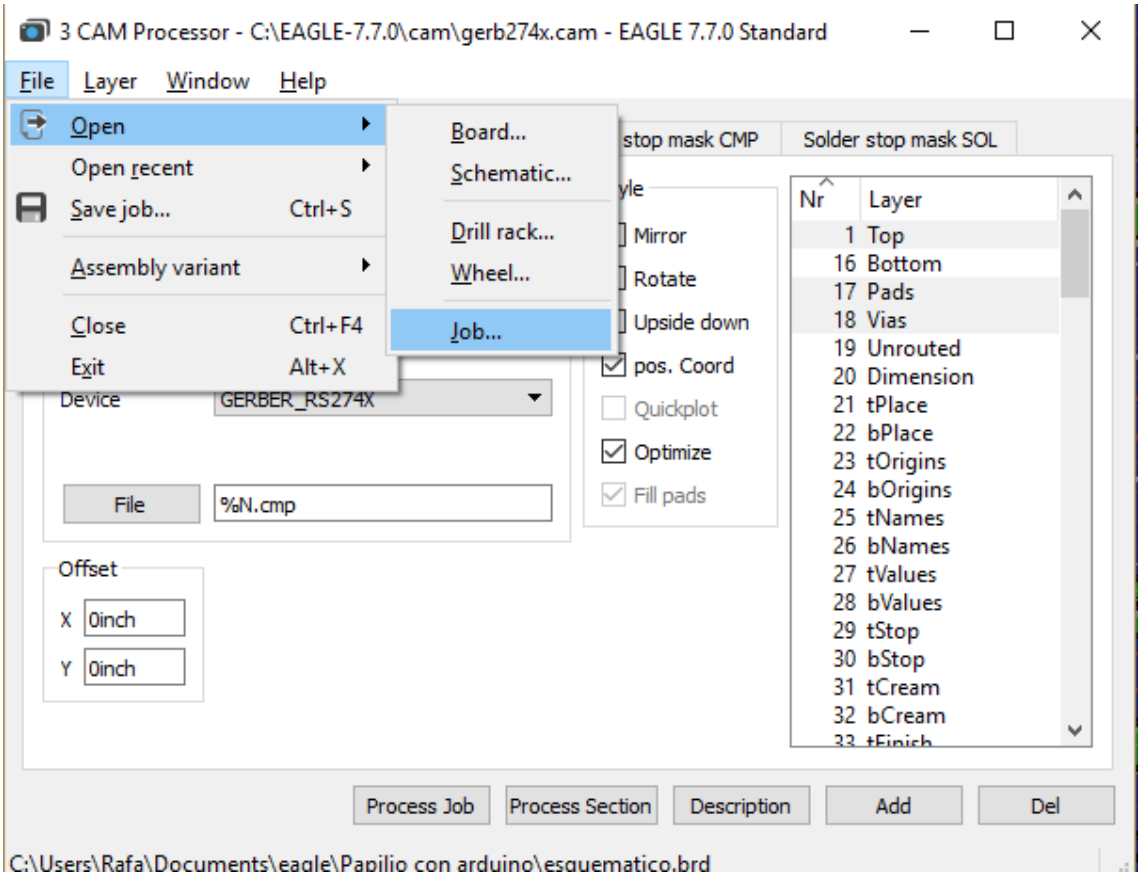
Ya por ultimo solo nos queda generar los archivos necesarios para mandar al fabricante.

Para ello:

Ejecutamos el **camProcessor** y aplicamos el archivo LPKFCAMFiles.zip de la página:

http://medesign.seas.upenn.edu/index.php/Guides/GeneratingGerberFilesForS62

Ya en este menú ejecutamos el Job.



Primero procesamos el Job del excellon.cam que son los taladros y los agujeros

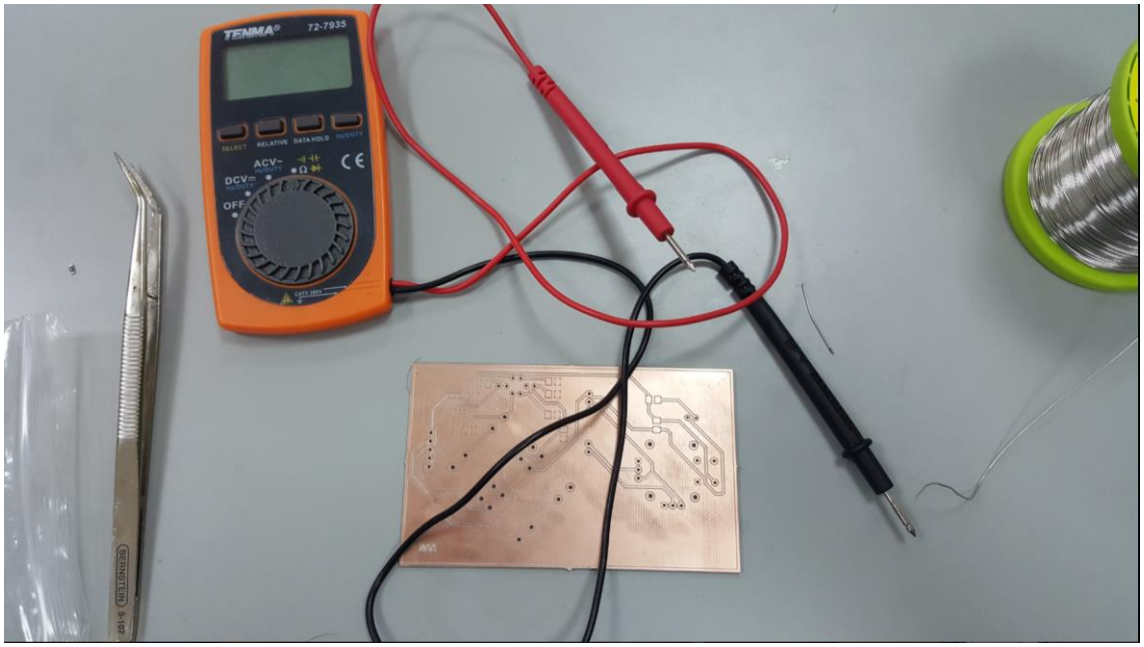
Segundo procesamos el Job del gerberforLPKF.cam y terminamos de generarlo todo.

Aquí tendremos el resultado de los archivos que necesitábamos para mandar a fabricar a la maquina nuestra protomats62.

RSW-LPKF.bor	5/14/2013 1:54 PM	BOR File	4 KB
RSW-LPKF.brd	5/14/2013 1:53 PM	BRD File	155 KB
RSW-LPKF.cmp	5/14/2013 1:54 PM	CMP File	40 KB
RSW-LPKF.crm	5/14/2013 1:54 PM	CRM File	3 KB
RSW-LPKF.drd	5/14/2013 1:53 PM	DRD File	2 KB
RSW-LPKF.dri	5/14/2013 1:53 PM	DRI File	1 KB
RSW-LPKF.gpi	5/14/2013 1:54 PM	GPI File	2 KB
RSW-LPKF.plc	5/14/2013 1:54 PM	PLC File	32 KB
RSW-LPKF.sol	5/14/2013 1:54 PM	SOL File	53 KB
RSW-LPKF.stc	5/14/2013 1:54 PM	STC File	8 KB
RSW-LPKF.sts	5/14/2013 1:54 PM	STS File	4 KB
RSW-LPKF.vsc	5/14/2013 1:54 PM	VSC File	1 KB

Revisión de la placa

Una vez que hemos recibido nuestra placa, lo primero que se debe hacer es comprobar pista a pista.



Para ello con la ayuda de un polímetro y la función de detectar continuidad, con el cable negro situado en el plano de masa comprobamos pista a pista si existe algún

cortocircuito, así como posibles cortos entre pistas o la posible mala fabricación de una pista completa.

Esto es muy importante puesto que, si comprobamos que todas las pistas están trazadas adecuadamente, nos puede librar de un quebradero de cabeza que sería buscar por qué ahora no me funciona esto o aquello.

Pues una vez todo dispuesto se procede a realizar las soldaduras, para ello se va a requerir un soldador que en nuestro caso será una estación de soldadura electrónica la cual se va a encargar de mantener la misma temperatura para poder realizar bien el proceso de fundir el estaño y calentar la superficie a soldar.

El estaño que hemos empleado libre de plomo, ya que su uso no se permite y es una aleación de estaño en su mayor parte, cobre y plata, a esto habría que añadir un **apunte y es que la temperatura de fusión seria de +217 ºC** en la práctica se hubo que aumentar la temperatura a 400 ºC para que la soldadura se efectuara correctamente.

Y para realizar la limpieza de la superficie y dejarla libre de óxido, así como para que el estaño se reparta lo más equitativamente posible a lo largo de la superficie que queramos soldar se emplea el flux.

En nuestro caso lo hemos empleado en forma de tubo para que nos sea más fácil su aplicación.

Hemos de tener en cuenta que una vez se haya realizado el proceso de soldadura se debe limpiar los restos de flux ya que este podría deteriorar el cobre de la placa.

A la hora de realizar el **ensamblaje** hubo que tenerse en cuenta por peligro de que los Leds se fundieran debido a que tienen **Polaridad** :

¡¡¡ NOTA IMPORTANTE POLARIDAD !!!

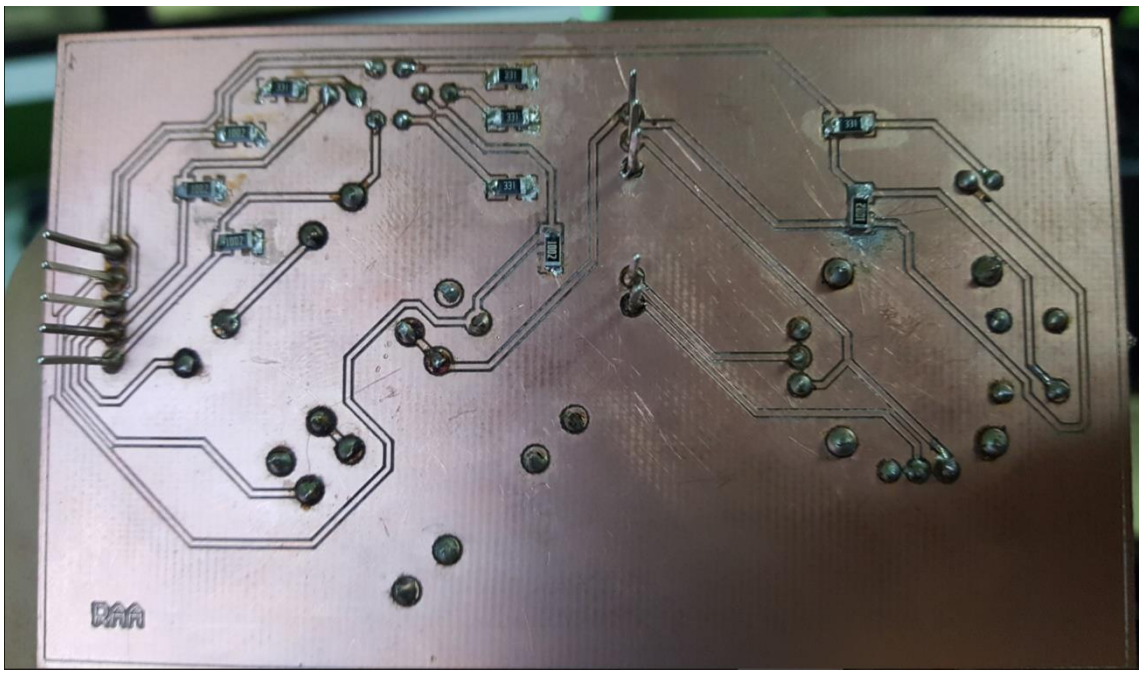
Led: **Pata larga ANODO +**

Pata Corta CÁTODO -

NOTA:

- Cada vez que se realizó una soldadura se fue comprobando si esta se había realizado correctamente y que no se habían producido cortocircuitos.

En la siguiente foto podemos ver el resultado de la placa en su cara bottom en la que se puede apreciar todos los componentes soldados, resistencias y pines, así como el trazado de las pistas que unen los Pads de los componentes según el esquema trazado en al Board.



Por ultimo mostrar la placa en su cara Top en donde se encuentran los componentes Leds, pulsadores, pines y Joystick.

Segundo Diseño de PCB en KIKAD

En este apartado se hace el mismo diseño de Arduino pero utilizando otro programa para la edición de PCB, en este caso se ha usado **KIKAD** veremos cómo se hace un esquemático , como se pasa el test ERC eléctrico , como se crea un símbolo , la asociación entre símbolo y huella que es distinta a la de Eagle(en Eagle el símbolo y la huella quedan casados, por decirlo de alguna manera, en el mismo device para siempre), en el caso de KiCad cada símbolo puede ser reutilizado por cualquier huella, también obtendremos un listado de los componentes y por último veremos cómo se crea el archivo de enlace de redes la cual se usa para crear la Board**.**

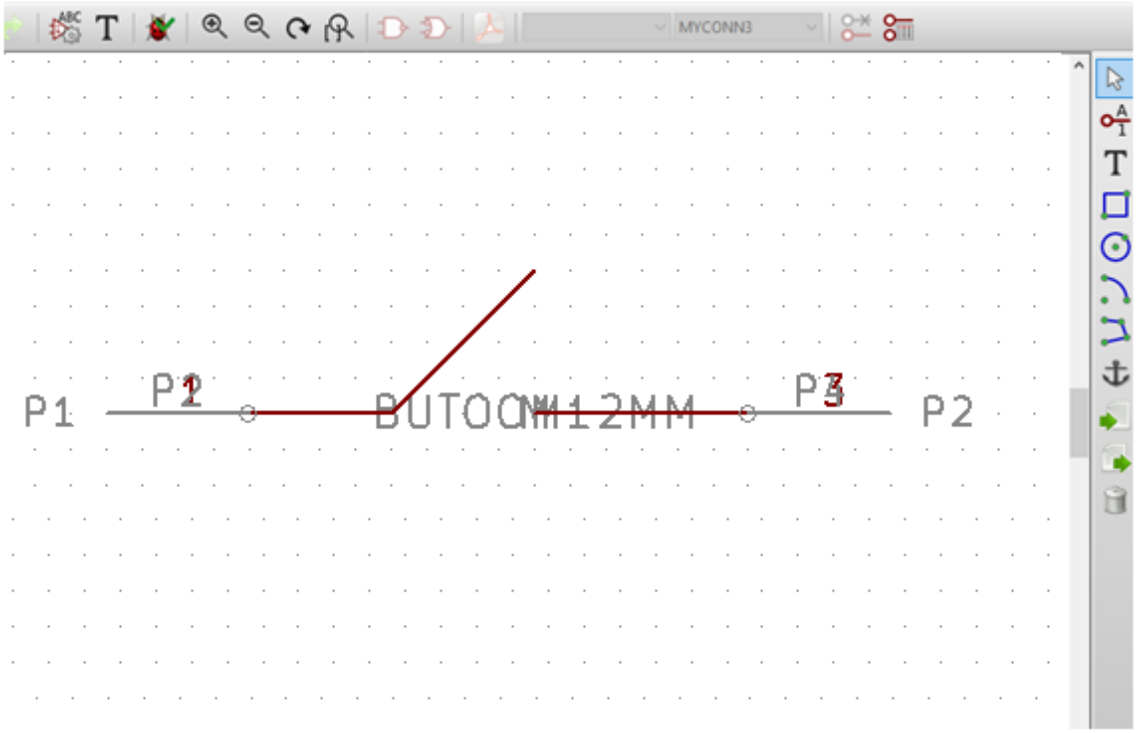
Esquemático

En primer lugar, se ha empezado haciendo un diseño de un esquemático (. sch) que será el circuito que vayamos a implementar en la PCB con sus respectivas conexiones de los componentes entre si y alimentaciones.

Para ello se tuvo que crear un símbolo para el **Interruptor** y otro para los **pines con Arduino**.

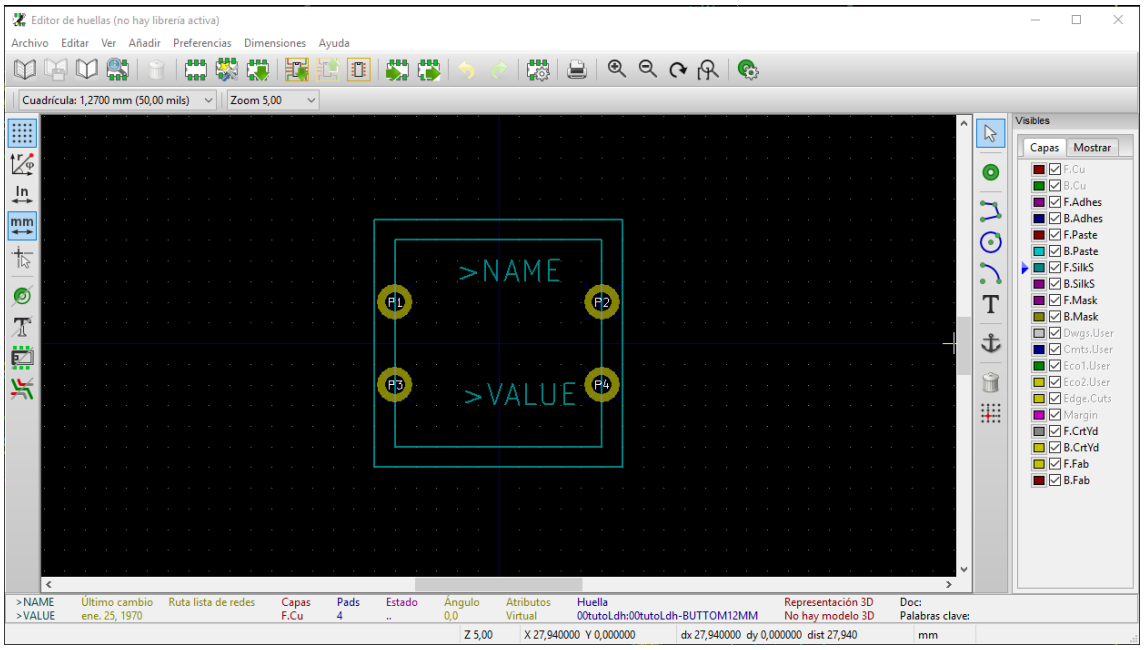
Interruptor

Símbolo



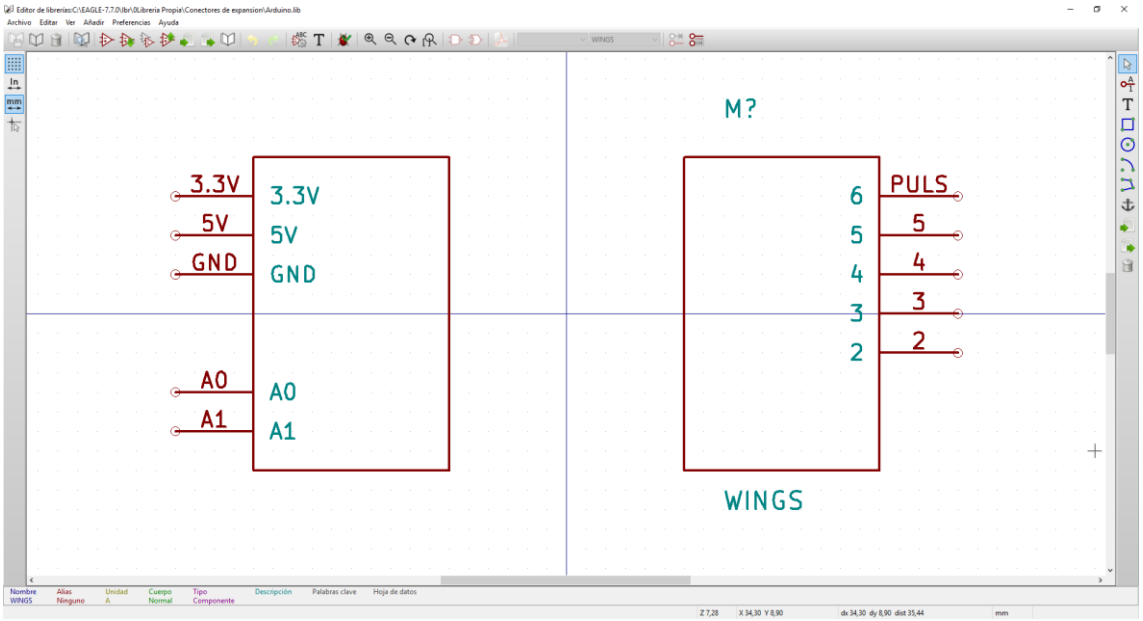
Siendo éste el interruptor Sparkfun de 12mm con sus dos E/S.

Y su huella:



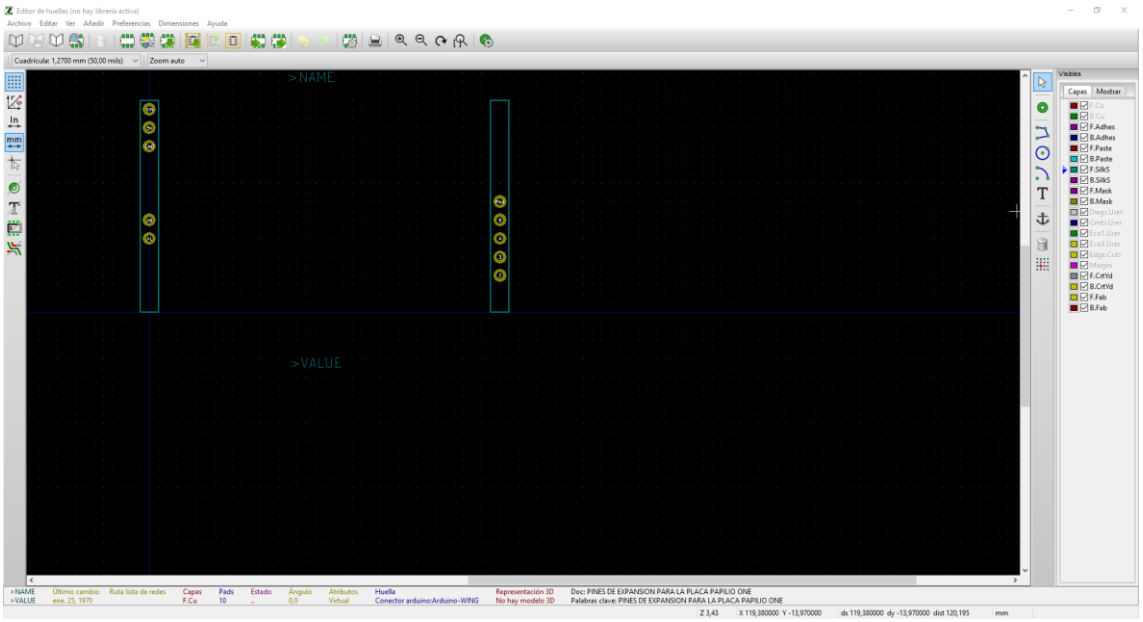
Pines con Arduino

Símbolo

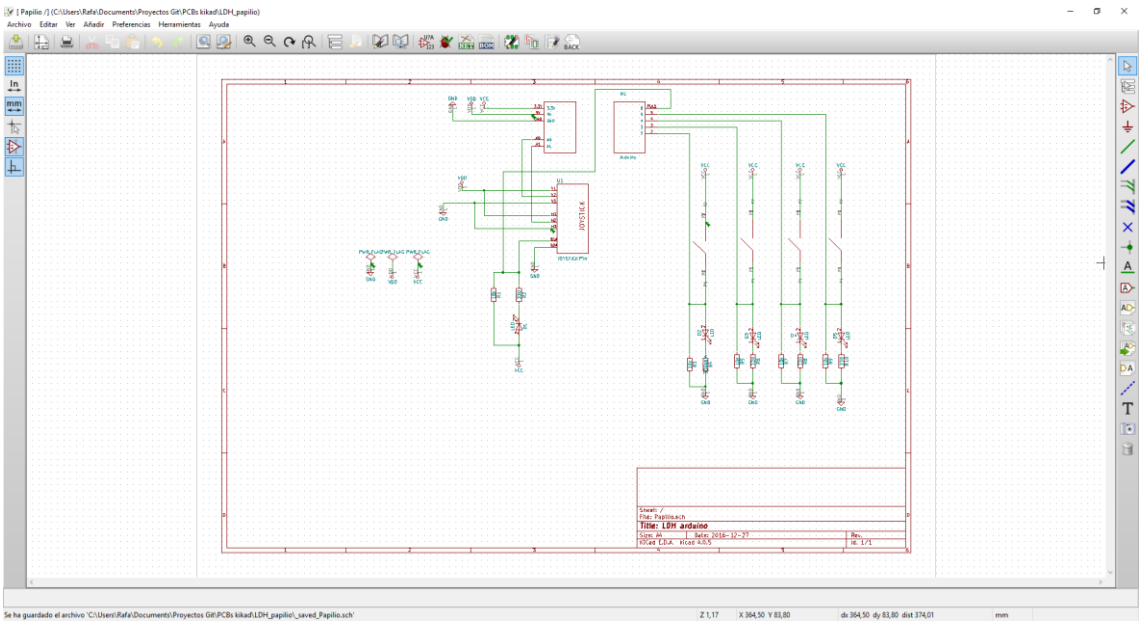


Aquí se pueden observar a la izquierda los pines analógicos y las alimentaciones y a la derecha los pines digitales.

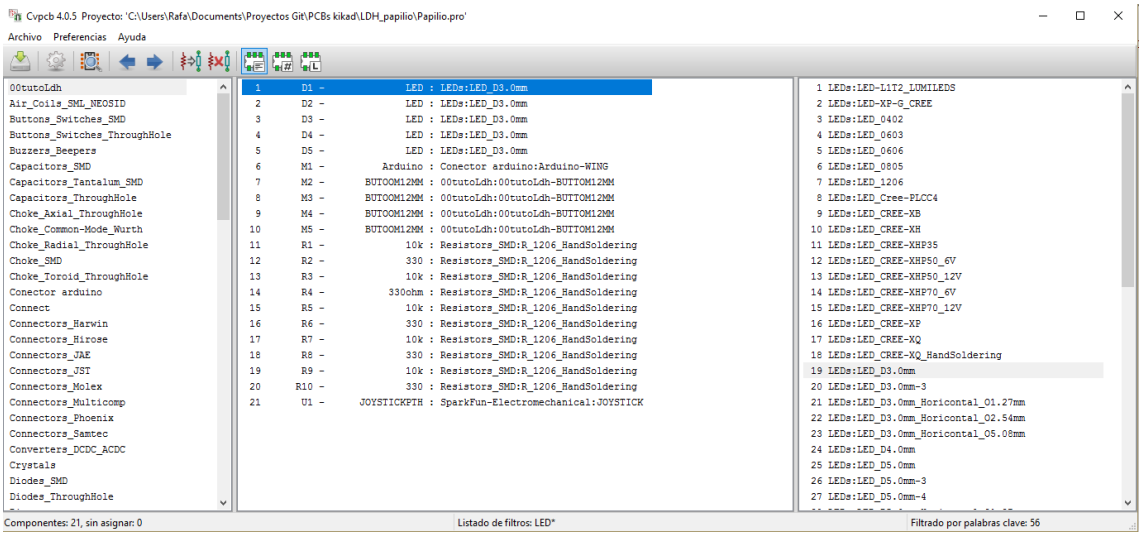
Y su huella:



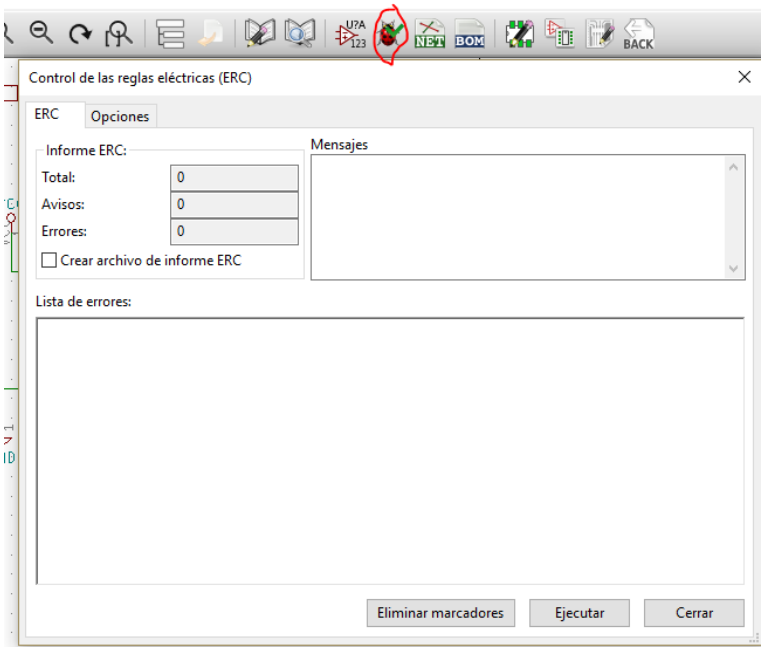
Y ya podemos comenzar con el dibujado del esquemático de nuestro circuito puesto que los demás componentes ya los podemos encontrar en las librerías de KiCad.



Tras el dibujado del circuito se asocian los símbolos con las huellas con CVPCB:



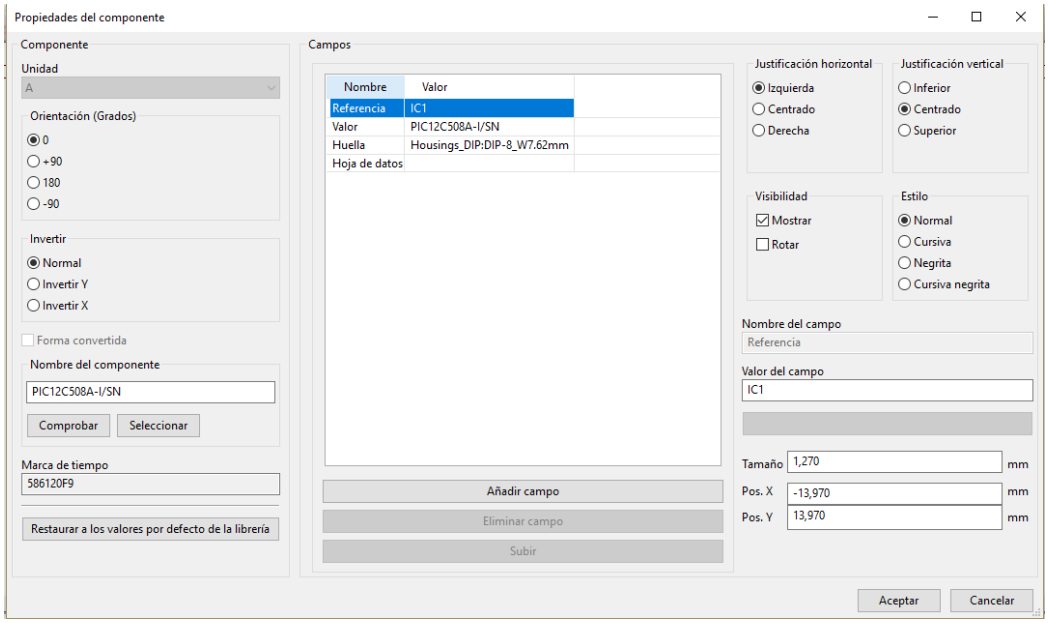
Tras dicha asignación se ha hecho el testeo de las reglas eléctricas:



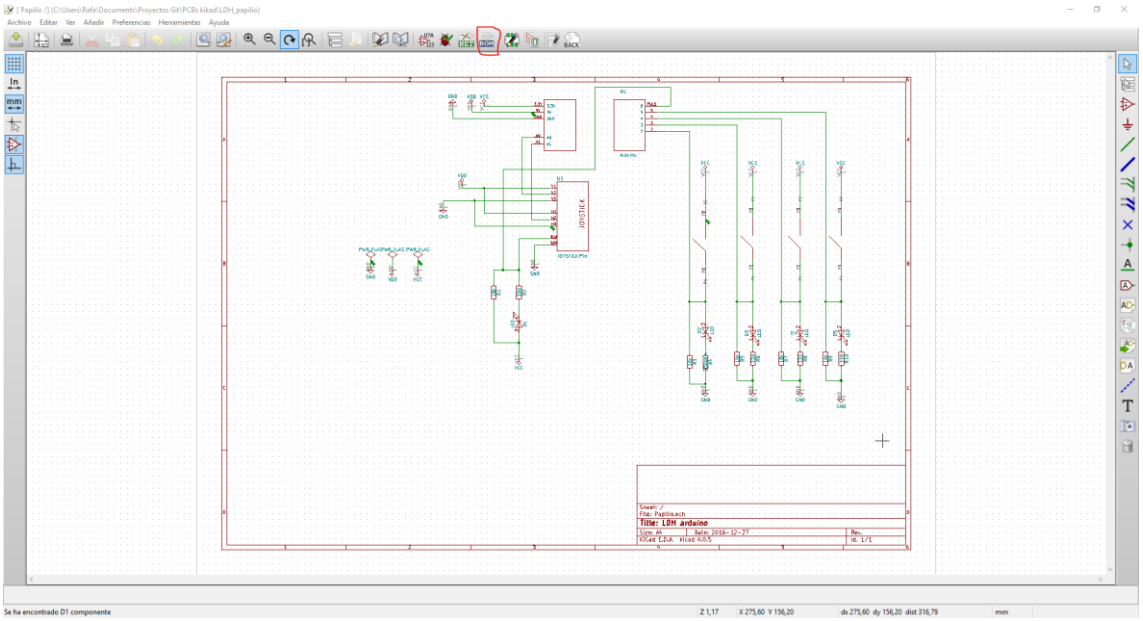
Y se ha generado un archivo de informe, que en este caso no nos dio errores.

Obtención y valoración del listado de los componentes

Para la obtención del listado de componentes le damos BOM (generar lista de materiales) el cual nos va a coger del esquema los componentes que vamos a necesitar en el proceso de montaje y nos genera automáticamente una lista de éstos, indicándonos la cantidad, tipo, huella o cualquier otro dato que nosotros queramos disponer editando la información de los componentes en el esquemático que podremos enviar a nuestro proveedor.

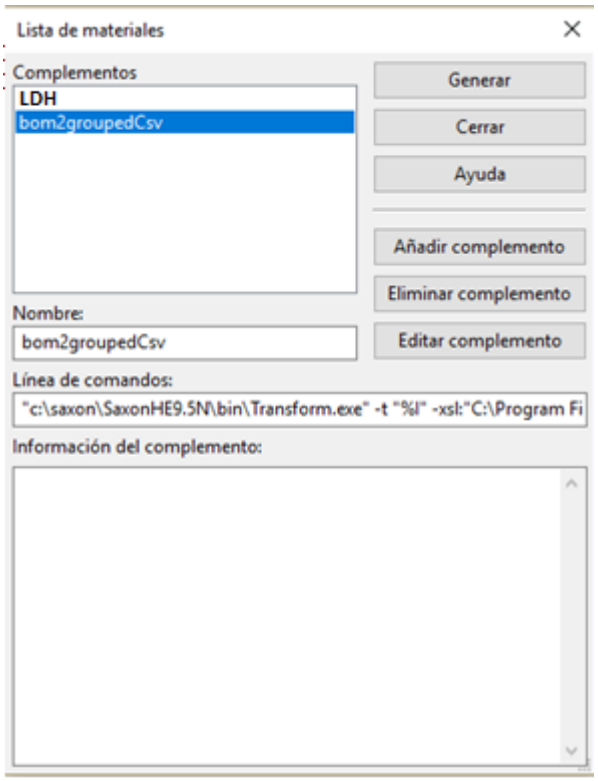


Aquí se muestra donde debemos darle para entrar en el menú de generación de lista de componentes.



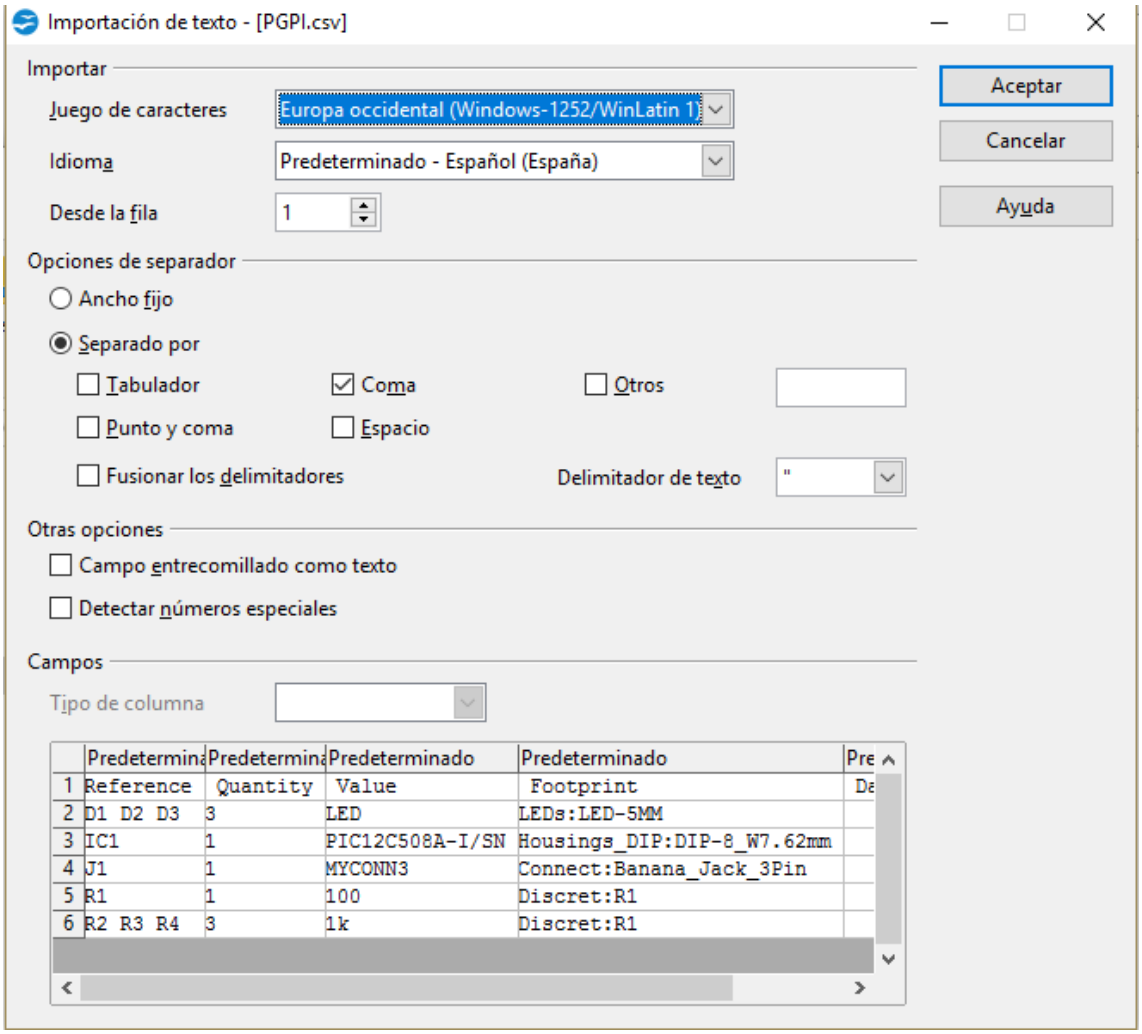
Una vez dentro tenemos que añadir un plugin, que en este caso hemos tenido que utilizar la información de este enlace debido a fallos.

https://forum.kicad.info/t/bom-generation-in-new-kicad-windows/228



Una vez terminado, se cambia la ruta del plugin, en nuestro caso hemos usado **bom2groupedCsv** y le damos a generar.

Una vez generado se ha abierto el archivo .csv con **libre Office** para poder crear el archivo separado por comas:

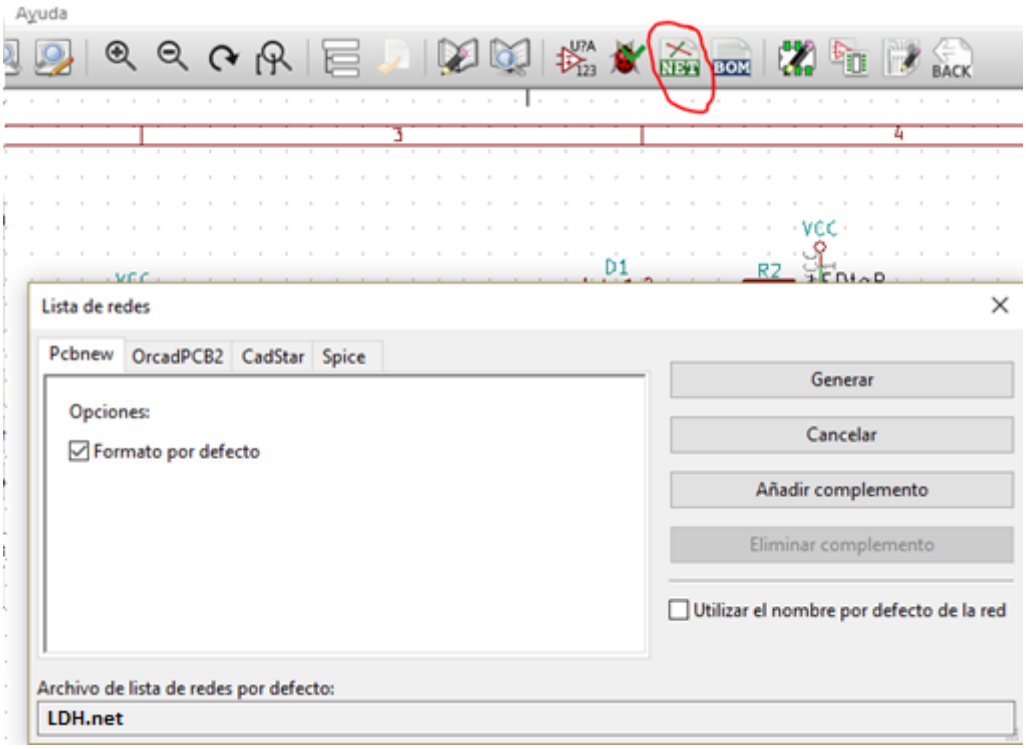


Y se nos crea el documento que hemos generado automáticamente en el proceso anterior:

	A	Cuadro de nombre	C	D	E
1	Reference	Quantity	Value	Footprint	Datasheet
2	D1 D2 D3	3	LED	LEDs:LED-5MM	
3	IC1	1	PIC12C508A-I/SN	Housings_DIP:DIP-8_W7.62mm	
4	J1	1	MYCONN3	Connect:Banana_Jack_3Pin	
5	R1	1	100	Discret:R1	
6	R2 R3 R4	3	1k	Discret:R1	
7					

Archivo de enlace de redes y enrutado

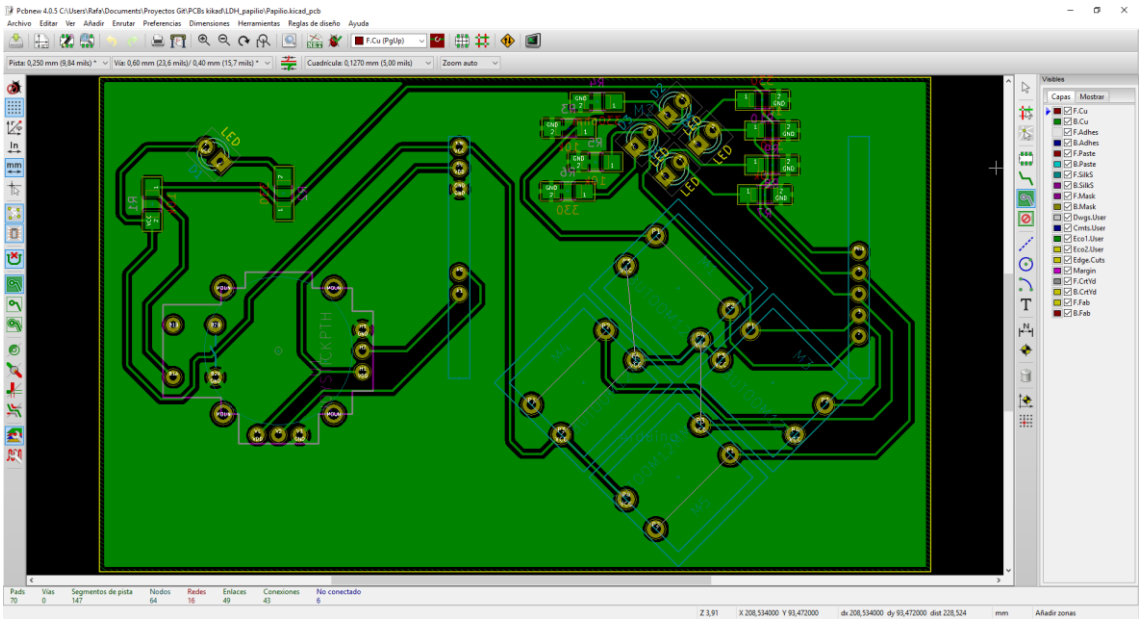
Ahora se procede al enrutado del circuito, para ello se genera el archivo de enlace de redes para poder realizar las conexiones en la herramienta pcbnew.



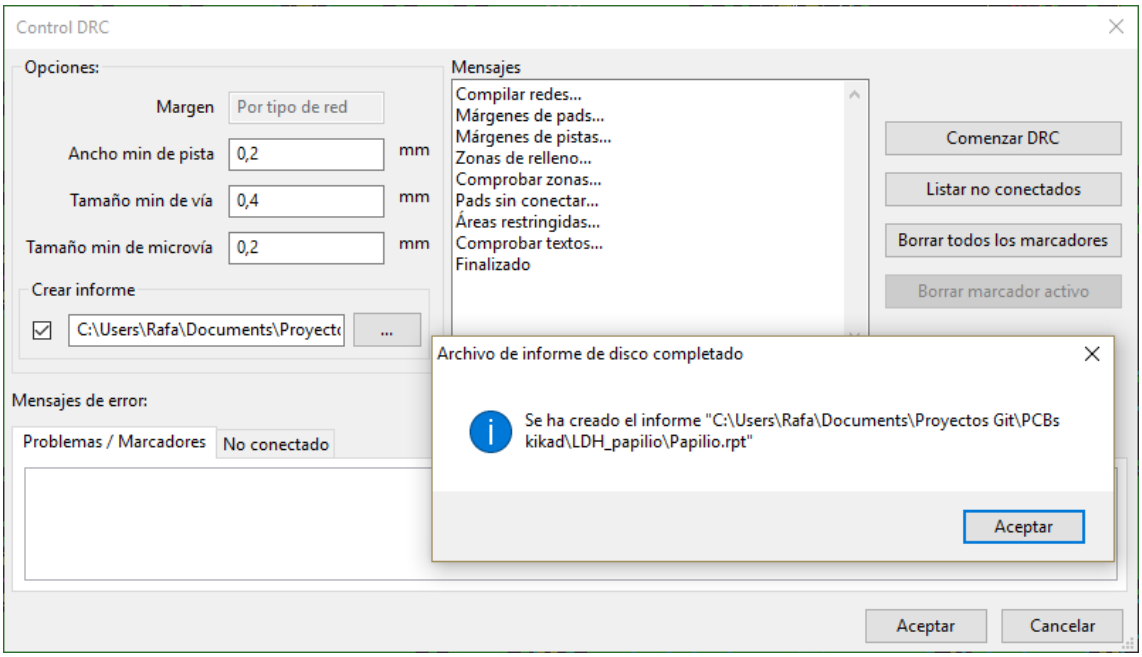
Ya dentro de ésta, se realiza la carga de dicha lista y nos aparecen todos los componentes unidos entre sí por los ratsnest que nos indica las conexiones que deben existir.

Se colocan los componentes sobre la que va a ser nuestra PCB, los conectamos con pistas en la capa Bottom de manera adecuada, y en la cara Top se colocaran los componentes.

Resultado:

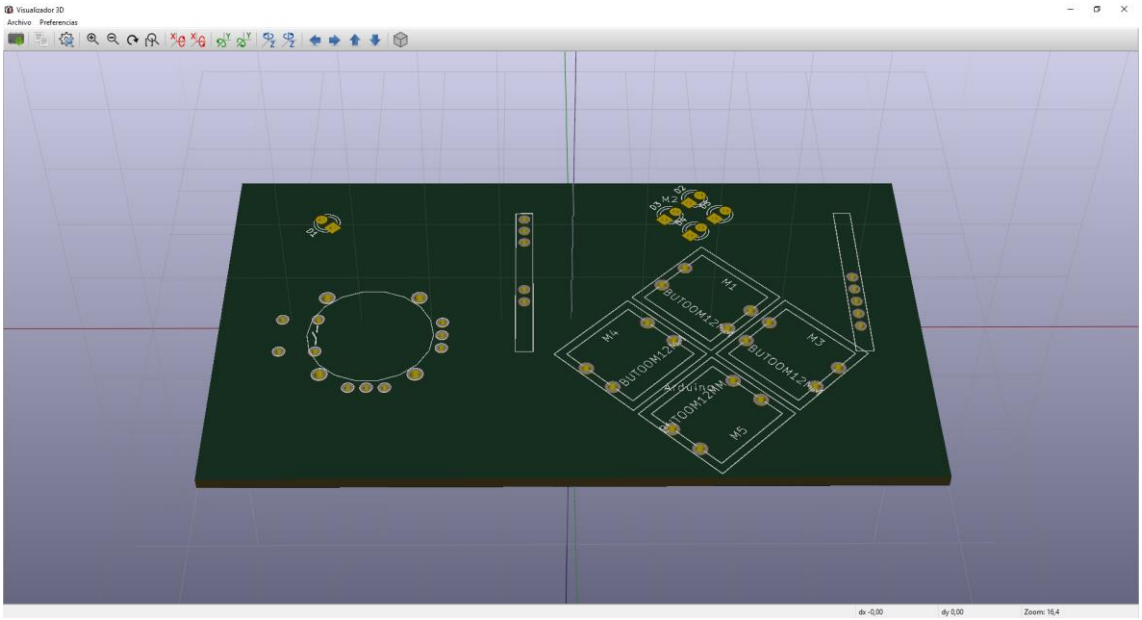


Tras el enrutado se realiza un control DRC para comprobar la correcta realización respetando las reglas de diseño.

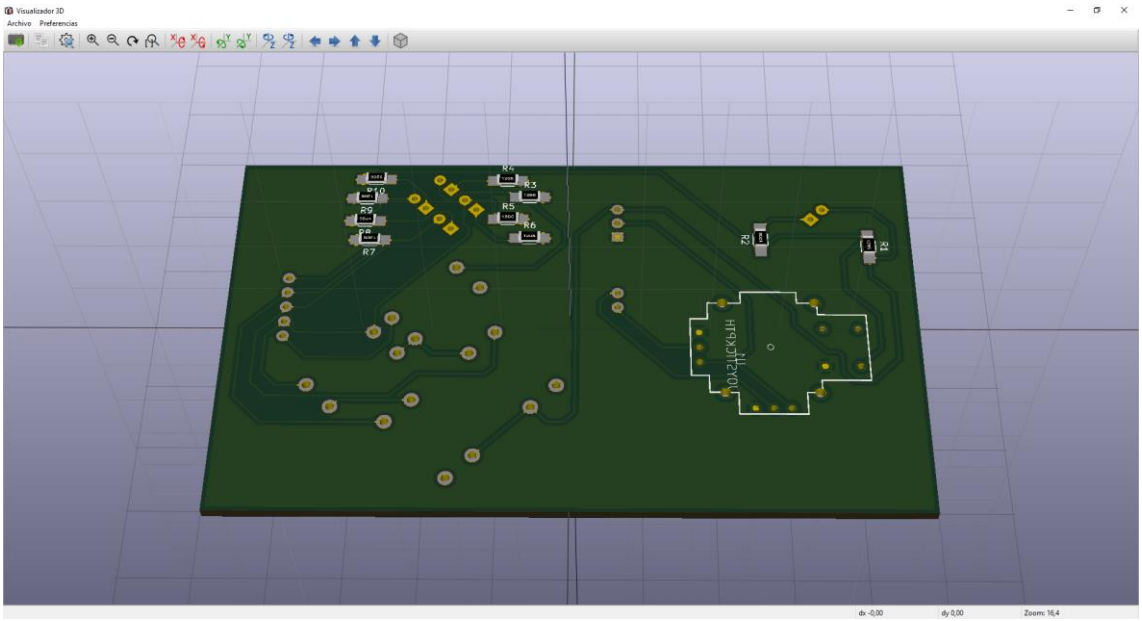


Diseño 3d de la placa:

Cara Superior

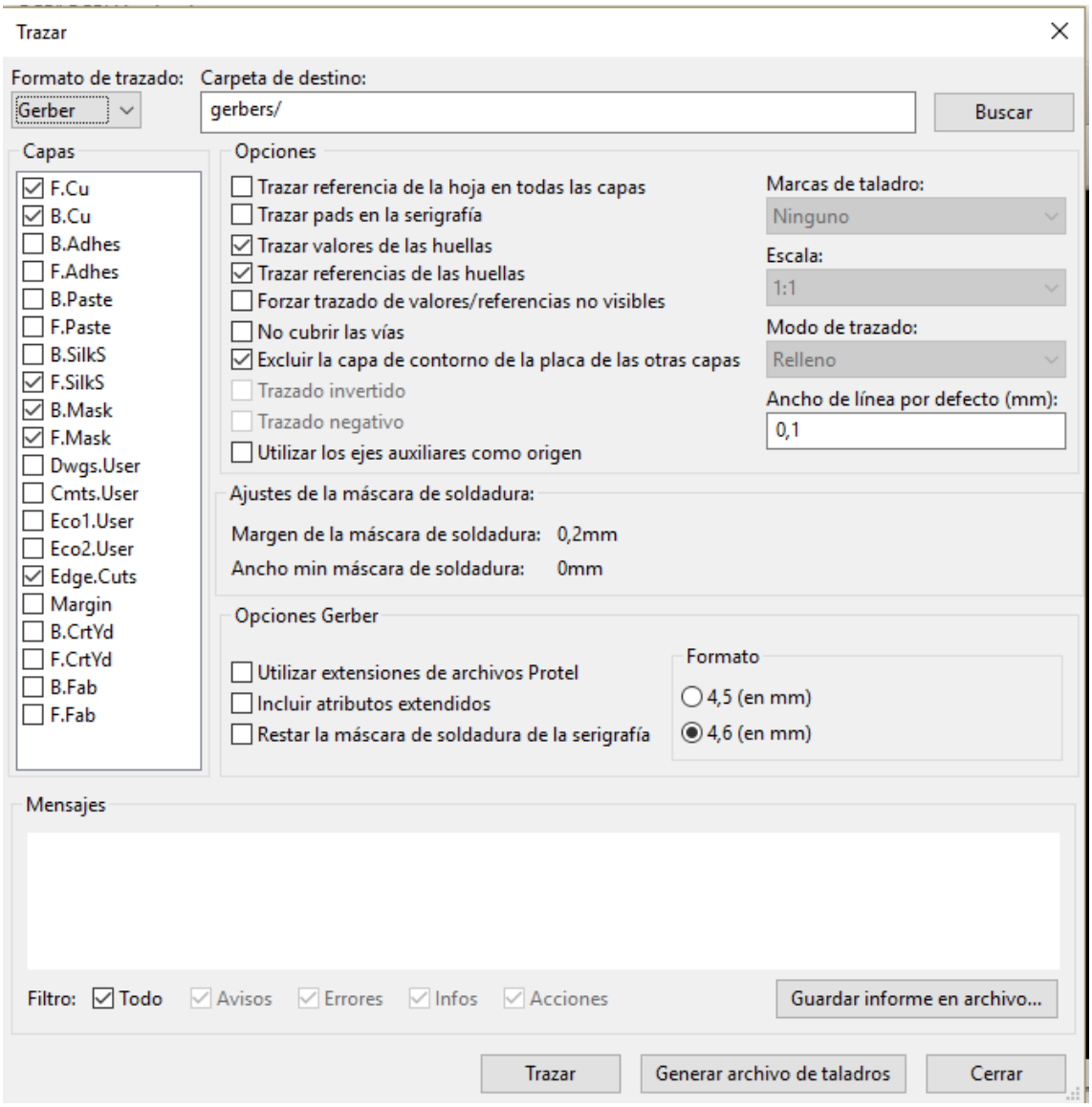


Vista Cara inferior



Gerbers

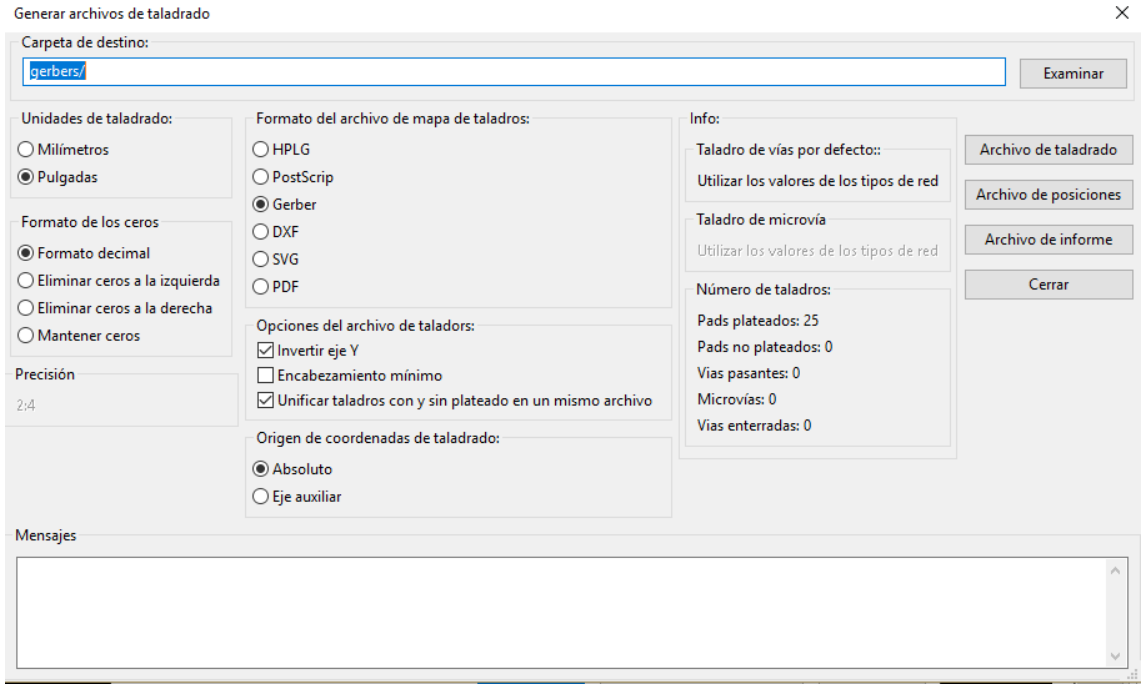
Una vez realizado procedemos a generar los gerbers y el archivo de taladros.



Marcamos en formato de trazado Gerbers y seleccionamos:

- Capa de cobre de la capa Top y Bottom
- Capa de serigrafía de componentes de la cara Top
- Capa de máscara de soldadura de ambas capas
- Así como el archivo de corte de la placa edge.cuts.

Y por último el archivo de taladrado



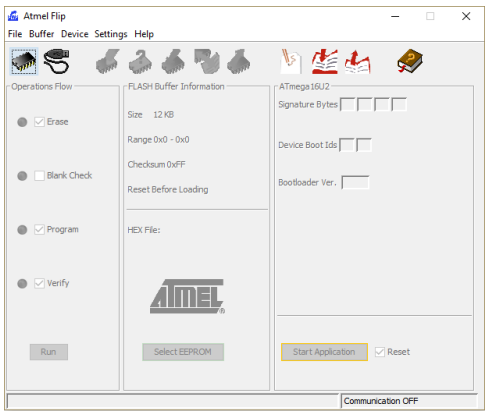
Y ya tenemos todo listo para poder enviar a fabricar nuestra placa a cualquier fabricante.

Programación de la placa Arduino como dispositivo HID para mando de juegos y conexión a móvil Android

En esta parte de la memoria se hará la programación de nuestra Arduino de manera que al conectarlo a un pc o a una consola este lo reconozca como si de un mando se tratase.

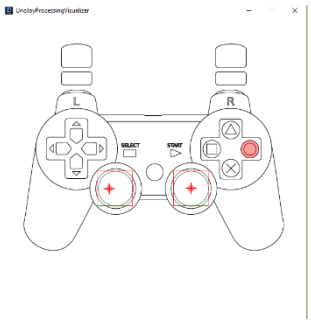
Para ello Necesitaremos tener instalado el Flip de Atmel.

Este programa nos permitirá flashear el chip que se encarga de hacer la conversión del USB-Serial para que se pueda realizar la comunicación como dispositivo HID.



Bien antes de hacer la programación de este chip tenemos que programar nuestra Arduino para que reconozca los botones y el Joystick de la manera que deseemos tras esta configuración comprobaremos mediante una herramienta proporcionada por el proyecto UnoJoy si los botones que hemos configurado están a nuestro gusto, el **UnoJoyProcessingVisualizer** aquí podremos simular un mando de la ps2 y comprobar si los botones están correctamente configurados o no , y si por ejemplo el joystick hace el movimiento e manera adecuada o de la forma que deseemos:

- Por ejemplo, en esta imagen se puede apreciar de que se está pulsando el botón de abajo del todo y que el joystick no se está tocando por eso se mantiene centrado



- En esta otra imagen se está desplazando el joystick y pulsando el botón de la derecha.

En este código se puede apreciar como constantemente se están realizando lecturas de las posiciones de los botones y joystick.

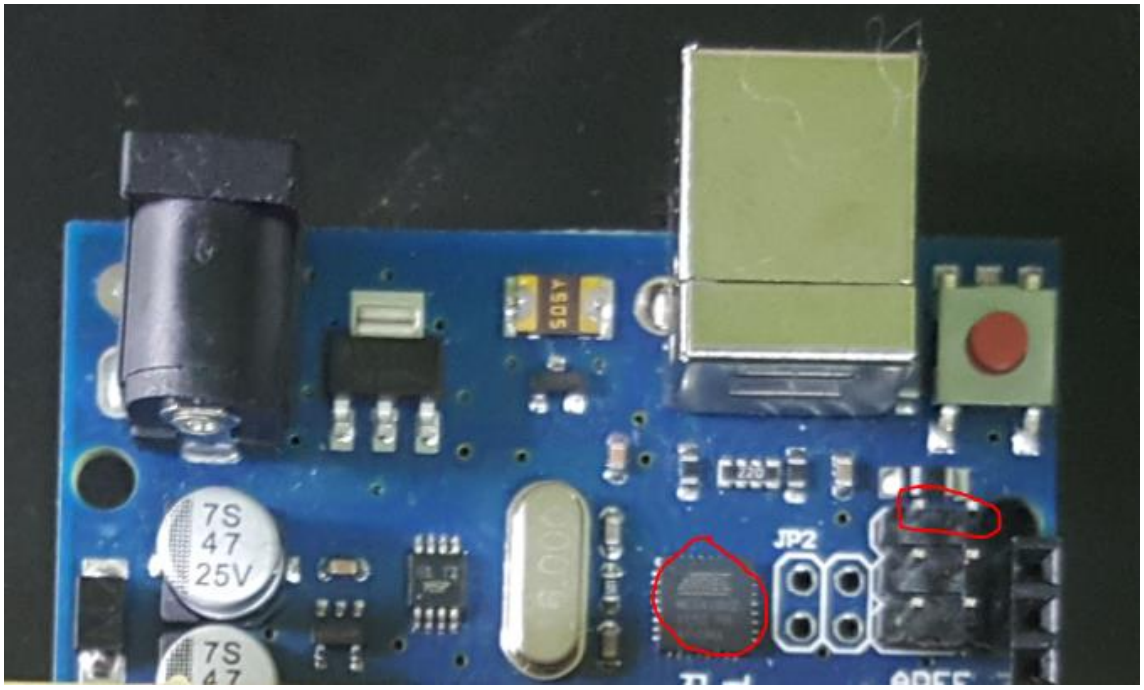
Botones cuando son pulsados se registra un 1.

Botón del joystick cuando se pulsa se registra un 0.

JoystickShield_Unojoy | UnoJoy.h

```
#include "UnoJoy.h"

void setup(){
  setupPins(); // Subrotina que configura os pinos do Arduino
  setupUnoJoy();// Inicializa as funções do UnoJoy
}

void loop(){
  // Sempre fica atualizando o dado a ser enviado ao computador
  dataForController_t controllerData = getControllerData();
  setControllerData(controllerData);
}

void setupPins(void){
  // Configura os pinos digitais 2~12
  // como entrada e com pull-up ativado
  for (int i = 2; i <= 12; i++){
    pinMode(i, INPUT);
    digitalWrite(i, HIGH);
  }
}

dataForController_t getControllerData(void){

  //  Configura local onde dados do controle são armazenados
  //  Serve para limpar o buffer onde os dados do controle são armazenados
  dataForController_t controllerData = getBlankDataForController();


  //  Associa os pinos digitais com o botão do UnoJoy
  //  A "!" inverte a leitura
  controllerData.triangleOn = digitalRead(2);
  controllerData.circleOn = digitalRead(3);
  controllerData.squareOn = digitalRead(5);
  controllerData.crossOn = digitalRead(4);
  controllerData.selectOn = !digitalRead(7);
  controllerData.startOn = !digitalRead(6);
  //  Configura o joystick Analógico
  //  A leitura "analogRead(pin)" retorna um valor de 10 bits(0~1023),
  //  Nós usamos um o "map" para deixar na faixa de 8 bits (0~255)
  //  Aproveitamos o map para inverte a leitura no eixo Y
  controllerData.leftStickX = map((analogRead(A1)),0,1023,255,0);
  controllerData.leftStickY = map((analogRead(A0)),0,1023,0,255);

  return controllerData;
}
```

Una vez realizado los pasos anteriores, ya podemos reprogramar el chip de conversión USB-Serial para que nos reconozca como HID para mando de juegos, para ello tenemos que hacer un jumper entre los dos pines marcados en la imagen conectado al PC.



Una vez hecho esto saltará un sonido en el PC indicando que se ha introducido un nuevo dispositivo y empezara la instalación de éste, es el chip del que estábamos hablando, que ahora está en modo DFU (Device Firmware Update) los podremos observar en el administrador de dispositivos del PC y tendremos que esperar a que termine el proceso de instalación del dispositivo.

Entonces ya podremos ejecutar el archivo **UnoJoyDriverInstaller.bat** que básicamente lo que hace es ejecutar un comando del Flip para que flashee el archivo de configuración.

Este será el resultado que nos debe de dar si el proceso se ha realizado satisfactoriamente.

Entonces desenchufamos nuestra Arduino y la volvemos a enchufar, y nos aparecerá nuestra placa como mando de juegos, en el apartado de Dispositivos.

Si lo queremos es revertir el proceso se deben de seguir los mismos pasos anteriormente citados, solo que en lugar de ejecutar **UnoJoyDriverInstaller.bat** se debe ejecutar **TurnIntoAJoystick.bat** y tendremos de nuevo nuestra Arduino como el dispositivo de USB-Serial.

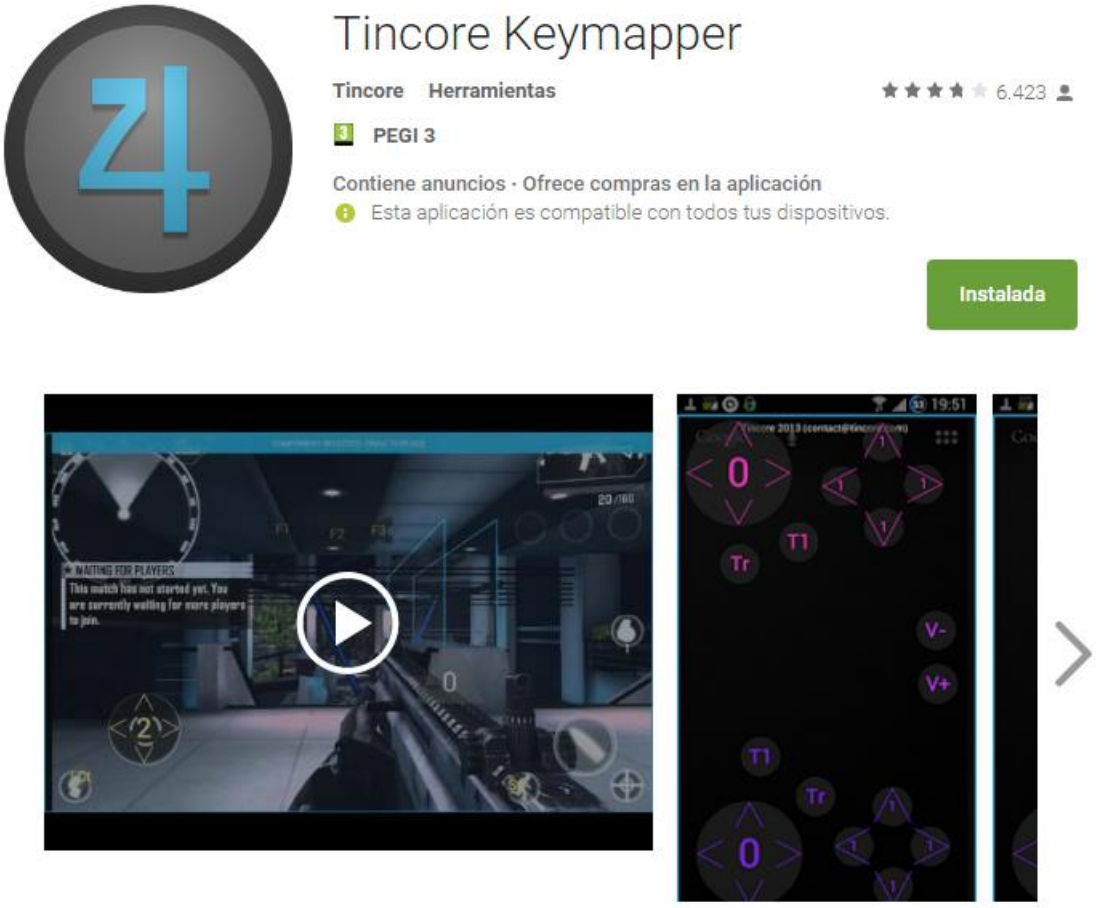
Nota:

- Cuando se conecte a un Pc se debe de calibrar el mando para que nos detecte bien el joystick de la manera deseada y adjudicar los botones y comprobar su buen funcionamiento.

Jugar en móvil

Para poder jugar con nuestro mando en un dispositivo Android nos encontramos con el problema de que los juegos de hoy están optimizados para jugar tocando la pantalla, para ello se empleó una aplicación que se encarga de mapear los botones que se pulsan en el mando.

Nuestra placa sera conectada al móvil **via OTG**.



De esta manera podemos jugar a cualquier juego, independientemente del sistema de juego que tenga o de la distribución de botones en la pantalla puesto que podremos configurar en la aplicación perfiles para los distintos juegos en los cuales se guardaran las distribuciones de los botones para poder realizar una rápida configuración a la hora de jugar.

Se adjunta el video que se siguió para aprender a configurar la aplicación:

https://www.youtube.com/watch?v=bh6izAAfyFc

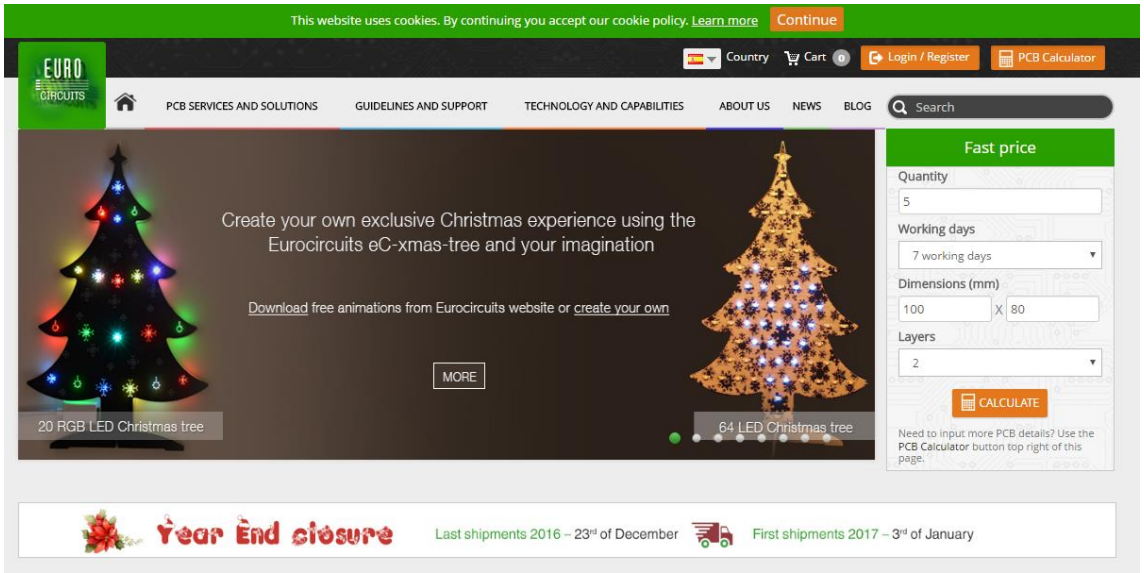
Nota:

- Es posible que la aplicación no funcione de manera adecuada si no está rooteado.

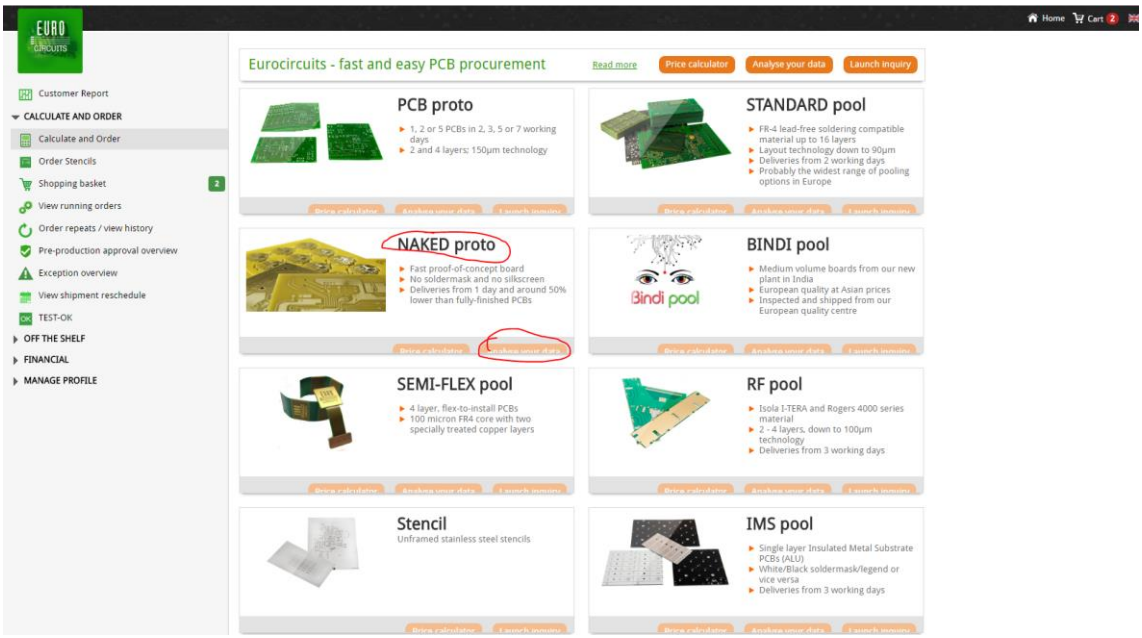
Eurocircuits, proceso de encargo a fabricación

En este caso se ha utilizado Eurocircuits el cual nos va a permitir subir los gerbers generados en nuestro proyecto, los cuales habremos empaquetado en un .zip previamente.

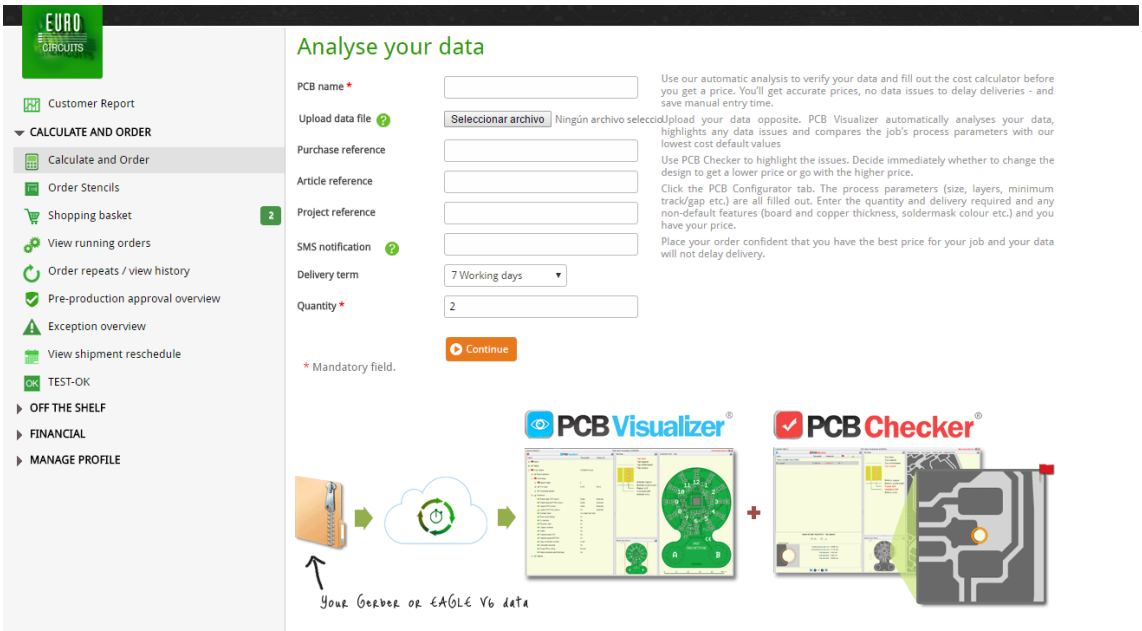
Aquí se muestra una imagen de la página principal



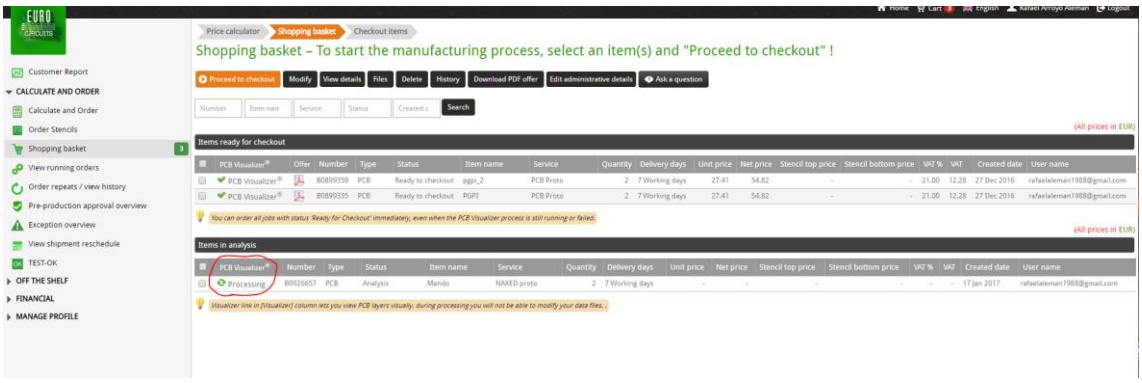
Para ello hacemos la simulación del cálculo de precio y orden de pedido, buscamos el campo de analizar los datos y elegimos cuál de los distintos tipos de placas disponibles que podemos mandar a fabricar nos va a interesar.



Y aquí subimos nuestros archivos gerbers que habíamos generado, pero ya todos empaquetados en formato .zip dándole clic al botón de Upload data file rellenamos cualquiera de los campos que nos pueda interesar y le damos a continuar.

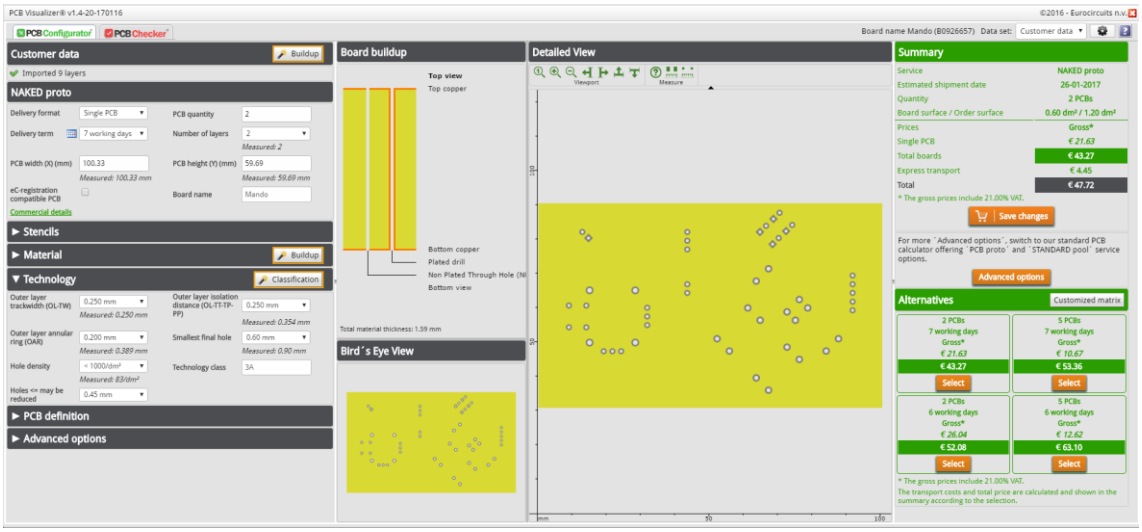


Y tenemos que esperar a que termine el proceso de análisis de nuestros archivos el cual demora un tiempo.

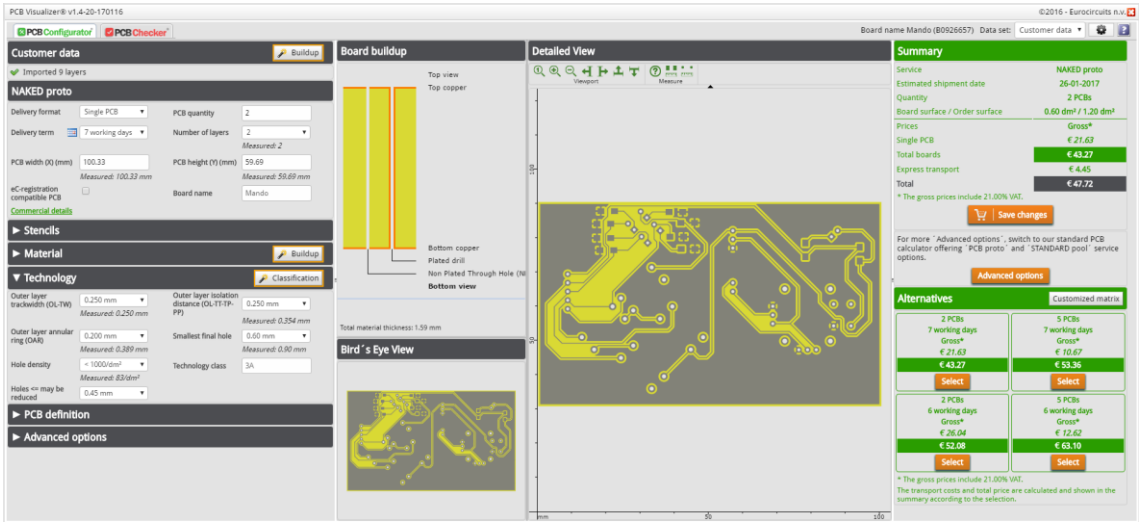


Una vez terminado recibiremos un PDF con la oferta que nos hacen en su fabricación a nuestro correo electrónico y podremos acceder a la herramienta PCB Visualizer en la que podremos ver un resultado en imagen de cómo quedaría nuestra PCB.

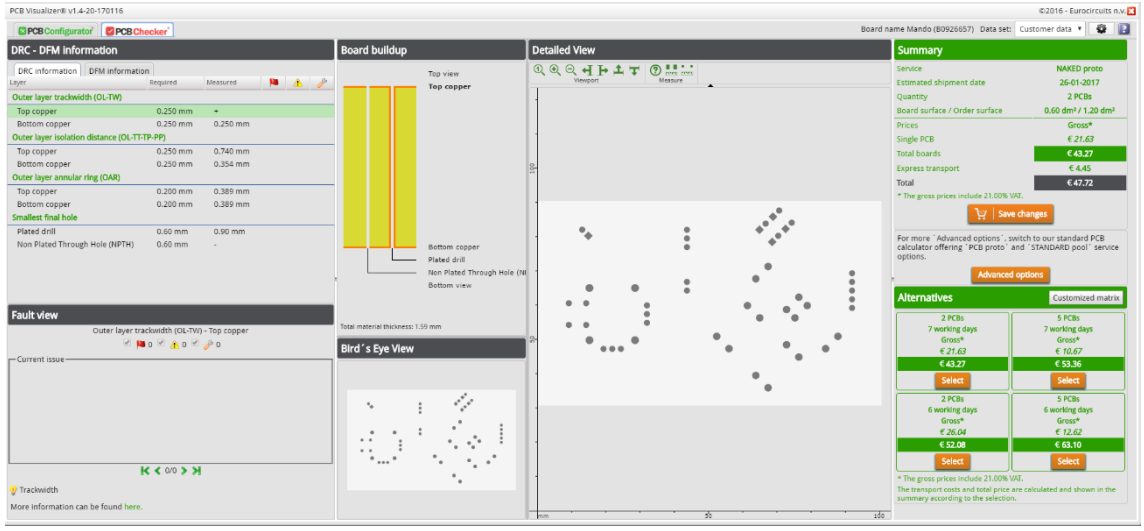
Aquí vemos las características de nuestra placa, y la imagen del resultado con los agujeros en su cara top.



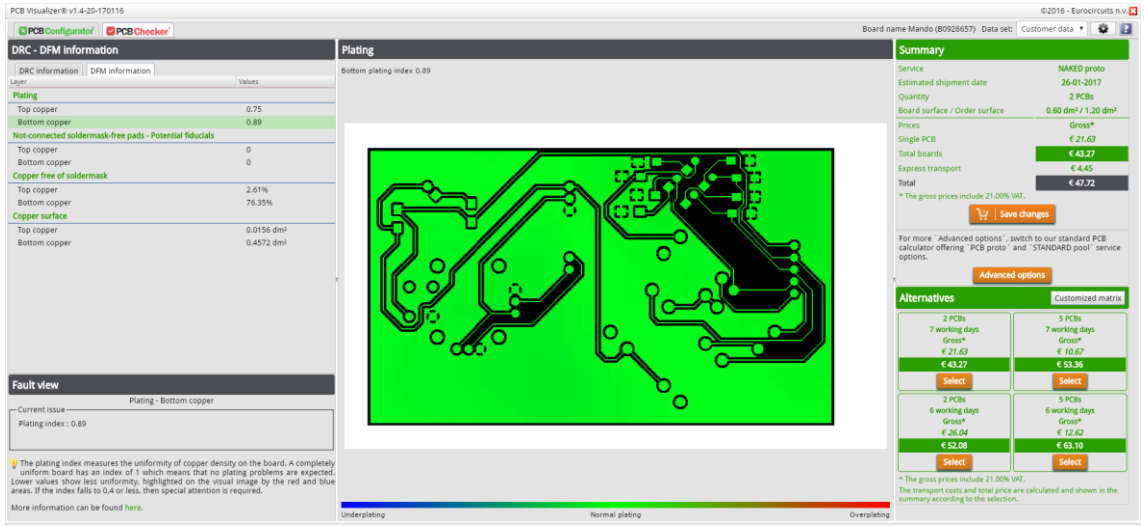
Y la imagen del resultado de las pistas de cobre en su cara Bottom, donde podemos observar tamaño de la placa, numero de PCBs, el isolate, los Pads, agujeros …



Y el PCB Checker que nos permite comprobar si cumplimos las reglas de diseño de las limitaciones que nos permite este fabricante y nos informa de cualquier posible cambio que debiéramos hacer en el diseño del layout.



Más información de nuestra placa, aquí se muestra por ejemplo la cantidad de cobre y otros datos que nos pueden ser interesantes:



Y por último el PDF con la oferta, aunque se adjunta con los archivos de la memoria:

https://1drv.ms/b/s!Ao2_30mhw3bigYJMmHaN_-6_WP585g