



LABORATORIO DE DESARROLLO HARDWARE - MEMORIA DE ZPUINO



José Manuel Jarana Expósito

Objetivos.....	2
Diseño de la placa	2
Conclusiones	6

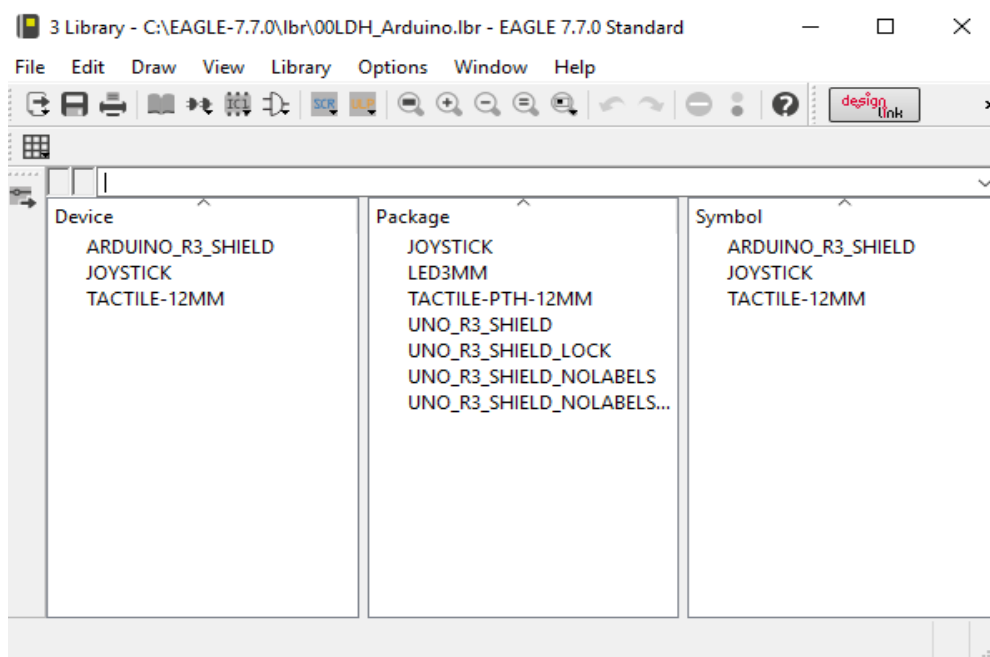
Objetivos

Nuestro objetivo de este proyecto es crear un shield de Arduino que sea un mando de juego con 4 botones y joystick con botón incorporado. Primero diseñaremos nuestro shield, haremos la placa con una maquina fresadora, soldaremos los componentes necesarios y por último programaremos.

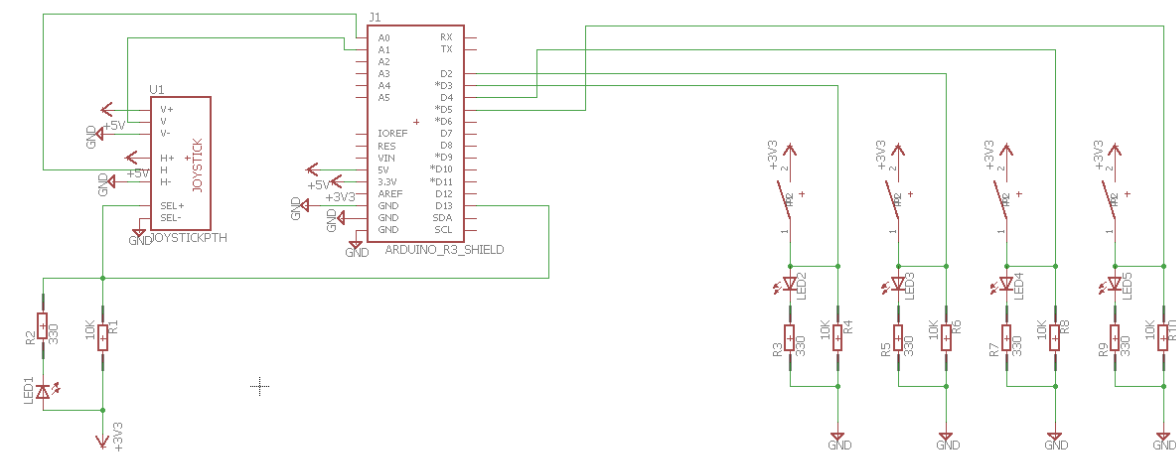
Diseño de la placa

Para empezar a diseñar la placa lo primero que hemos hecho es elegir el CAD en el que haremos el diseño. Nosotros hemos elegido Eagle, que aunque en principio sea un CAD de pago, tiene una versión gratuita que nos será más que útil. Aunque esta ha sido la he utilizado inicialmente y a partir de la que he generado los Gerbers para la creación de la placa, también utilice otro CAD que se está poniendo mucho de moda últimamente y que es totalmente libre, KiCad. He comprobado que también es una herramienta completísima y no tiene nada que envidiar a otros CAD de pago, puesto que tiene una comunidad muy grande por detrás.

Después de elegir el CAD en el que vamos a hacer es crear una librería con los componentes que vamos a utilizar en nuestro diseño de la placa. Esto no es algo necesario, puesto que se pueden coger los componentes de las librerías que necesitemos y punto, pero yo lo hice por buena praxis.



Tras crear nuestras librerías con estos componentes, creamos nuestro esquemático.

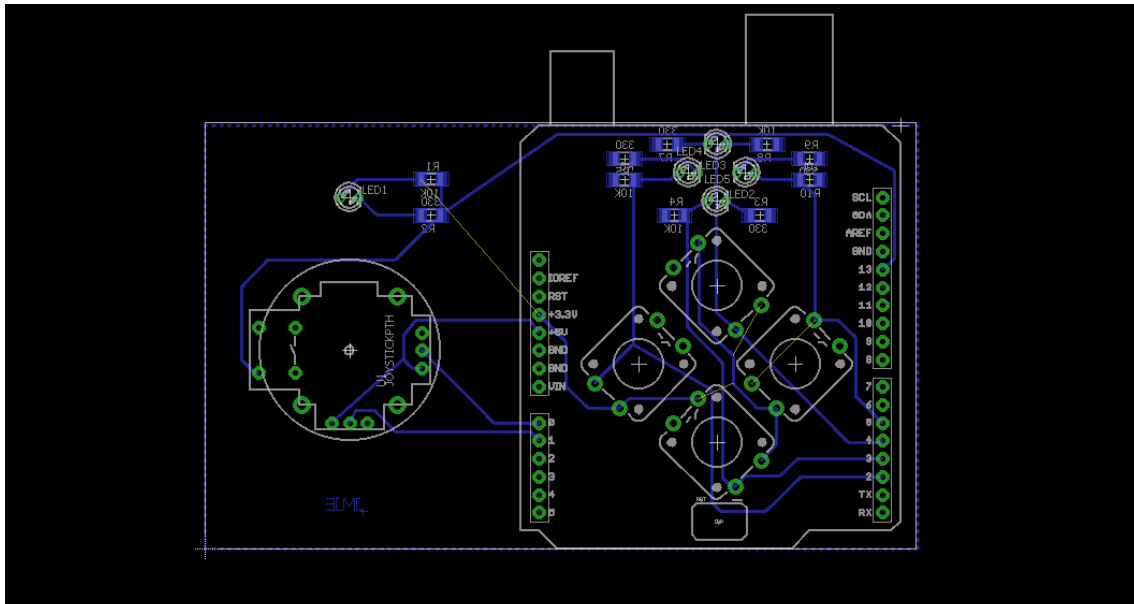


Una vez hecho el circuito básico, tenemos que proceder a plasmarlo en la placa, colocando los componentes en las dimensiones y las características que querremos de nuestra board. Las características con las que la hemos creado son:

- 10cm x 6cm de área, a una sola cara.
- Tamaño del cable es 0,4064 mm (0,016inch).
- Plano de tierra en el resto de la cara de soldadura conectando solo los GND.

Una vez diseñado toda la placa con las características descritas anteriormente, procedemos a colocar los componentes como queremos en el área diseñada. Cuando los pongamos, empezaremos a hacer el enrutado de los componentes hasta conectarlos todos. Podemos tener problemas a la hora de conectarlos, en ese caso quizás replantearnos la posición o el ángulo de rotación nos haga las cosas más fáciles.

Por último, como comprobación cuando hemos hecho todo el enrutado, miraremos la aplicación que tiene Eagle llamada DRC, para comprobar a ver si las conexiones hechas son correctas o si somos conscientes de los fallos que ponen para ignorarlos. En nuestro caso pudimos ignorar algunos puesto que había conexiones que no detectaba el programa ya que habíamos conectado pulsadores de forma tuvimos en cuenta que en ellos existía una conexión interna de los pines. Una vez comprobado los errores, crearemos los Gerbers siguiendo el tutorial indicado, así enviaremos todos los datos geométricos relativos a la placa para su fabricación. Así nos ha quedado la placa tras haber puesto todos los circuitos y demás



Tras crear los gerbers, para comprobar nuestra placa hemos utilizado la página de eurocircuits, para que nos haga un chequeo antes de fabricarla.

PCB Visualizer® v1.4-24-170119 ©2016 - Eurocircuits n.v.

Board name PCB LDH (B0930307) Data set: Customer data

DRC - DFM Information

Layer: Values

Plating	Values
Bottom copper	0.92
Not-connected soldermask-free pads - Potential fiducials	
Bottom copper	0
Copper free of soldermask	

Fault view

Plating - Bottom copper

Current issue

Plating index : 0.92

The plating index measures the uniformity of copper density on the board. A completely uniform board has an index of 1 which means that no plating problems are expected. Lower values show less uniformity, highlighted on the visual image by the red and blue areas. If the index falls to 0.4 or less, then special attention is required.

More information can be found [here](#).

Plating

Bottom plating index 0.92

Underplating Normal plating Overplating

Summary

Service	NAKED proto
Estimated shipment date	31-01-2017
Quantity	1 PCB
Board surface / Order surface	0.60 dm² / 0.60 dm²
Prices	
Single PCB	€ 27.00
Total boards	€ 27.00
Express transport	€ 3.68
VAT 0.00%	€ 0.00
Total gross	€ 30.68

[Save changes](#)

For more 'Advanced options', switch to our standard PCB calculator offering 'PCB proto' and 'STANDARD pool' service options.

[Advanced options](#)

[Alternatives](#) [Customized matrix](#)

PCB Visualizer® v1.4-24-170119 ©2016 - Eurocircuits n.v.

Board name PCB LDH (B0930307) Data set: Customer data

Customer data

The board build might not be correct

Imported 4 layers

NAKED proto

Delivery format	Single PCB	PCB quantity	1
Delivery term	7 working days	Number of layers	1
PCB width (X) (mm)	100.00	PCB height (Y) (mm)	60.00
Measured: 100.00 mm		Measured: 60.00 mm	
eC-registration compatible PCB	<input type="checkbox"/>		
Board name PCB LDH			

[Commercial details](#)

[Stencils](#)

[Material](#)

[Technology](#)

Board build

Top view

Total material thickness: 1.59 mm

Bird's Eye View

Summary

Service	NAKED proto
Estimated shipment date	31-01-2017
Quantity	1 PCB
Board surface / Order surface	0.60 dm² / 0.60 dm²
Prices	
Single PCB	€ 27.00
Total boards	€ 27.00
Express transport	€ 3.68
VAT 0.00%	€ 0.00
Total gross	€ 30.68

[Save changes](#)

For more 'Advanced options', switch to our standard PCB calculator offering 'PCB proto' and 'STANDARD pool' service options.

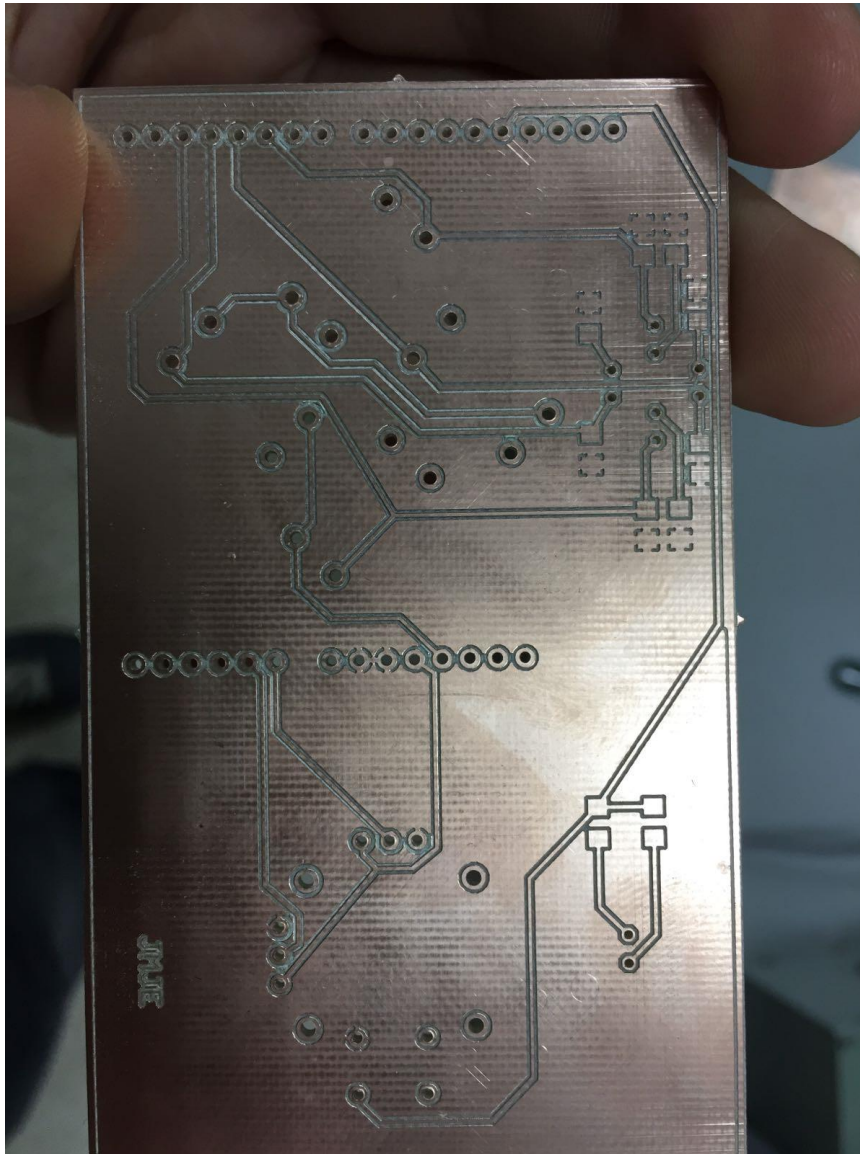
[Advanced options](#)

[Alternatives](#) [Customized matrix](#)

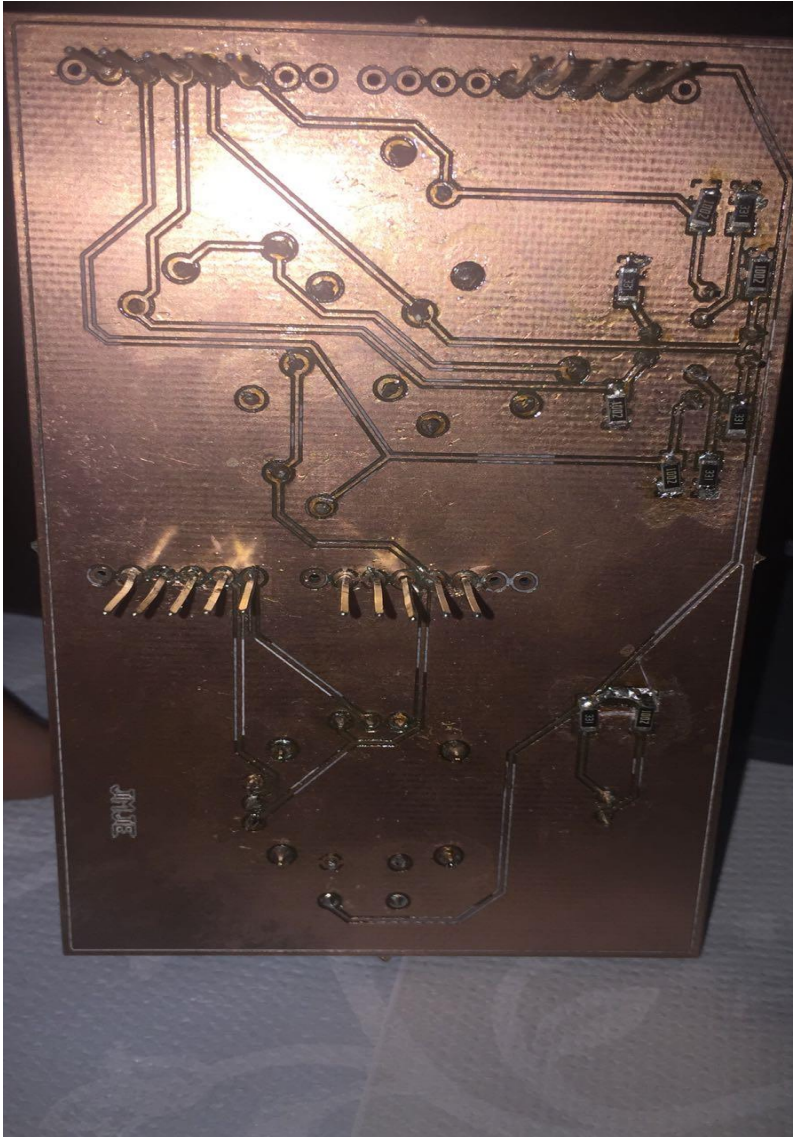
Nos ciertos warnings ya comentados antes, en la que nos dicen que algunas pistas no tienen conexión, o que algunos sitios están sin conectar, pero como dijimos antes, somos conscientes

de esto puesto que la conexión será interna dentro de los mismos pulsadores, así que no hay que preocuparse.

Una vez generados nuestros gerbers, mandamos a fabricar nuestra placa, dándonos como resultado esto:



Este esquemático como dijimos es el que hemos hecho en Eagle, sin embargo, también utilizamos KiCad para aprender a utilizar esta plataforma gratuita e hicimos otro esquemático. Para seguir el mismo método que anteriormente, lo que hicimos fue crearnos nuestra librería con los componentes que utilizamos. Podemos crearnos nuestros componentes al igual que en Eagle, sin embargo, los más específicos como el joystick, shield y pulsador lo conseguimos de los foros oficiales de KiCad, de gente que los ha creado o importado de otros CAD. El resto de los componentes simplemente los hemos buscado en la librería que trae ya de por sí KiCad. Una vez incorporado nuestros elementos, procederemos a hacer nuestro esquemático de nuevo, el mismo que anteriormente.



Software

Para utilizar nuestro shield en Arduino y que el PC lo detecte correctamente como joystick vamos a utilizar un programa que hizo alguien por internet y que compartió de manera libre, para que cada uno lo pudiera utilizar. El programa lo que hace básicamente es simular que nuestro mando sea un mando de PS3 al conectarlo por USB, y por tanto el resto de aplicaciones o juegos lo detecten directamente. Este programa se llama UnoJoy.

Para empezar vamos a descargar los elementos que necesitamos de esta página, en la que nos indica como haremos configuraremos todo en inglés. Ahora lo explicaré yo en Español.

<http://garagelab.com/profiles/blogs/tutorial-unojoy-utilize-seu-arduino-uno-como-um-joystick>

Una vez que nos descargamos el .zip que contiene el programa de UnoJoy, también descargaremos el sketch para subir a nuestra placa de Arduino. Debemos tener en cuenta

también que para que funcione correctamente tenemos que tener instalado el JRE Flip de atmel, ya que sin esto no se ejecutará correctamente el resto de ejecutables. Ahora vamos a configurar nuestro Arduino.

1. Subiremos el sketch en Arduino, haciendo las modificaciones oportunas respecto a en que pines tenemos conectado nosotros los botones y al tipo de conexión que nosotros tenemos hecha en ellos, siendo en nuestro caso Pull-Up. También nosotros cambiamos como teníamos configurando el joystick, donde el “eje x” era el “eje y” y viceversa, por tanto lo que hicimos fue intercambiar la configuración para que fuera de manera correctamente. El código que nosotros subimos en concreto fue este.

```
#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(4);
    controllerData.circleOn = digitalRead(5);
    controllerData.squareOn = digitalRead(2);
    controllerData.crossOn = digitalRead(3);
    //controllerData.selectOn = !digitalRead(2);
    //controllerData.startOn = !digitalRead(7);

    // 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);
```

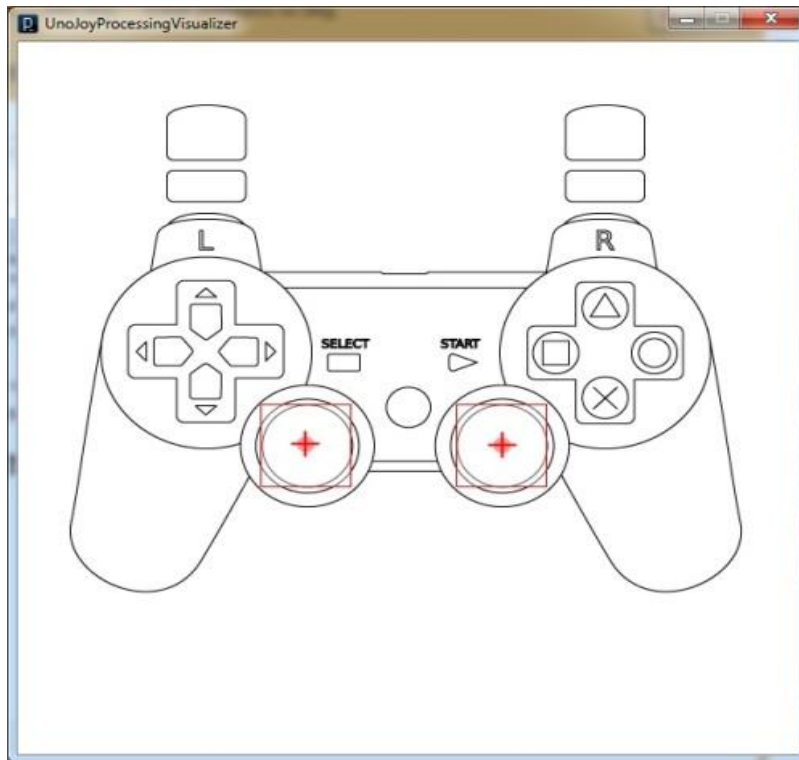
```

controllerData.leftStickY = map((analogRead(A0)), 0, 1023, 0, 255);

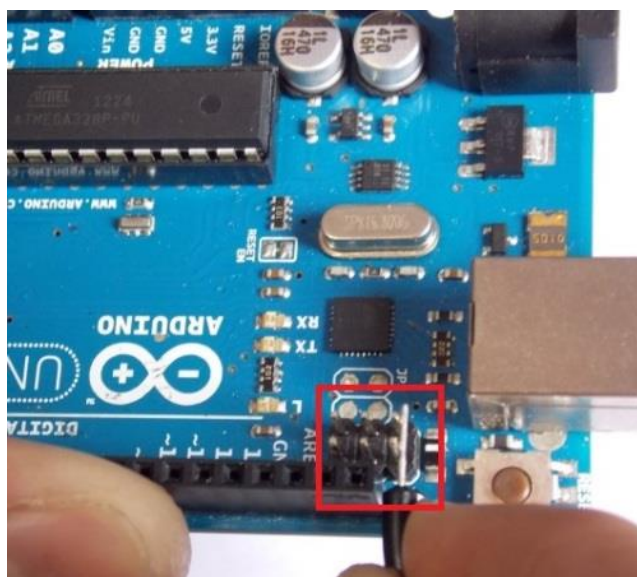
return controllerData;
}

```

2. Una vez hemos subido esto, vamos a comprobar que funciona correctamente pulsando los botones utilizando *UnoJoyProcessingVisualizer.exe*.



3. Tras comprobar esto, vamos a configurarlo ya para que el PC lo detecte como un joystick. Tenemos que provocar un cortocircuito entre estos jumpers de Arduino para que se active como microcontrolador ATmega16u2.



4. Ejecutaremos el archivo *UnoJoyDriverInstaller.bat* para instalar los drivers.
5. Por último, ejecutamos *TurnIntoAJoystick.bat*. Si todo sale correctamente nos saldrá una pantalla como está.

6. Ahora desconectamos y volvemos a conectar a nuestro Pc Arduino y ya se ejecutará como un mando con un joystick y cuatro botones. Se puede ver en el panel de administrador de dispositivos.

Aquí tenemos una demostración de su uso y correcto funcionamiento en un juego.

<https://youtu.be/VPkoM9piDo>

Conclusiones

En general es un proyecto y unas prácticas muy interesantes de hacer. Al principio puede que costara un poco empezar, sobre todo a la hora de diseñar y manejarse con los entornos, pero mirando algunos tutoriales uno se va haciendo idea de cómo tiene que hacerse las cosas. Además de ciertos trucos que se comparten con compañeros para hacerlo. Una vez hecho el diseño y fabricada nuestra PCB, de nuevo con la práctica vamos a soldar mejor nuestros componentes y haciéndolos de forma sencilla al final incluso.

En general es una práctica muy útil y didáctica, sobre todo teniendo en cuenta que estamos en la rama de Computadores y muchos de nosotros no habíamos tenido la posibilidad de trabajar con PCBs de este estilo y mucho menos crearla nosotros.

Yo en concreto cometí ciertos fallos que para una próxima vez estaré más pendiente de no cometerlos, como puede ser que, en el esquemático, la alimentación que iba al botón del joystick y el led la señalicé distinta al resto, por tanto, a la hora del diseño no me salió que la conexión debía hacerse, por tanto, no me funciona el botón ni el led por ese motivo.

Además, en el momento de la soldadura al usar la malla de de-soldadura para eliminar parte del estaño que eché de más en un pre-estrañeado, para que se calentará iba girando la punta

del soldador, y una parte de la punta de ese soldador no calentaba correctamente (nos dimos cuenta más tarde), por tanto, la malla dejó de calentarse y se pegó a la PCB. Intentando retirarla saltó una pista. Aun así, después pudimos arreglar el estruendo creando una nueva pista con un cúter y estaño, y no afecta ese error al funcionamiento de la placa.