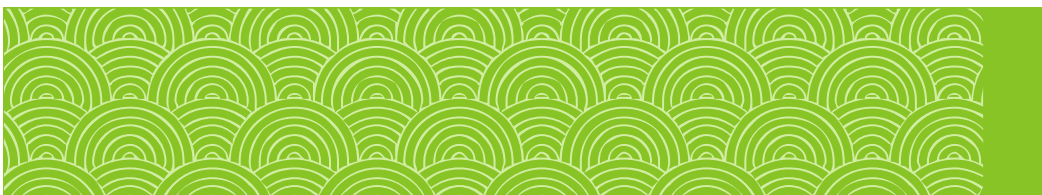
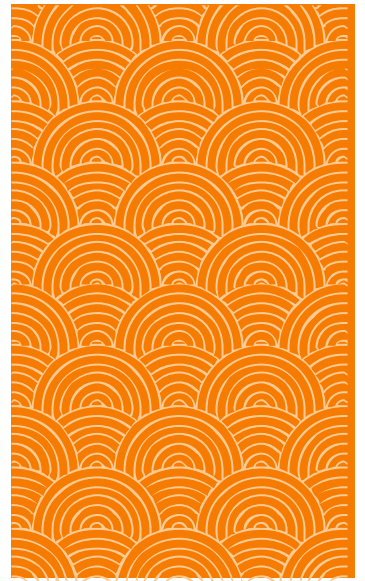


MEMORIA DISEÑO PCB JOYSTICK PARA ARDUINO

27/ENERO/2017

DANIEL CASTRO CHINCHO :



INDICE

1.1 OBJETIVOS-----	→ 3
1.2 DISEÑO ESQUEMATICOS-----	→ 4
1.3 DISEÑO BOARD-----	→ 6
1.4 FASE FABRICACIÓN-----	→ 7
1.5 FASE ENSAMBLAJE-----	→ 9
1.6 FASE TEST-----	→ 11

1.1 Objetivos:

El objetivo principal de este bloque es conocer el proceso de diseño, fabricación, ensamblaje y testeo de una placa de circuito impreso (PCB). De forma teórica y práctica se estudiarán las distintas fases que comprende este proceso y los elementos que forman parte de él.

Se desarrollará una placa de expansión con los componentes necesarios para cumplir la función de joystick a lo largo de las sesiones de laboratorio.

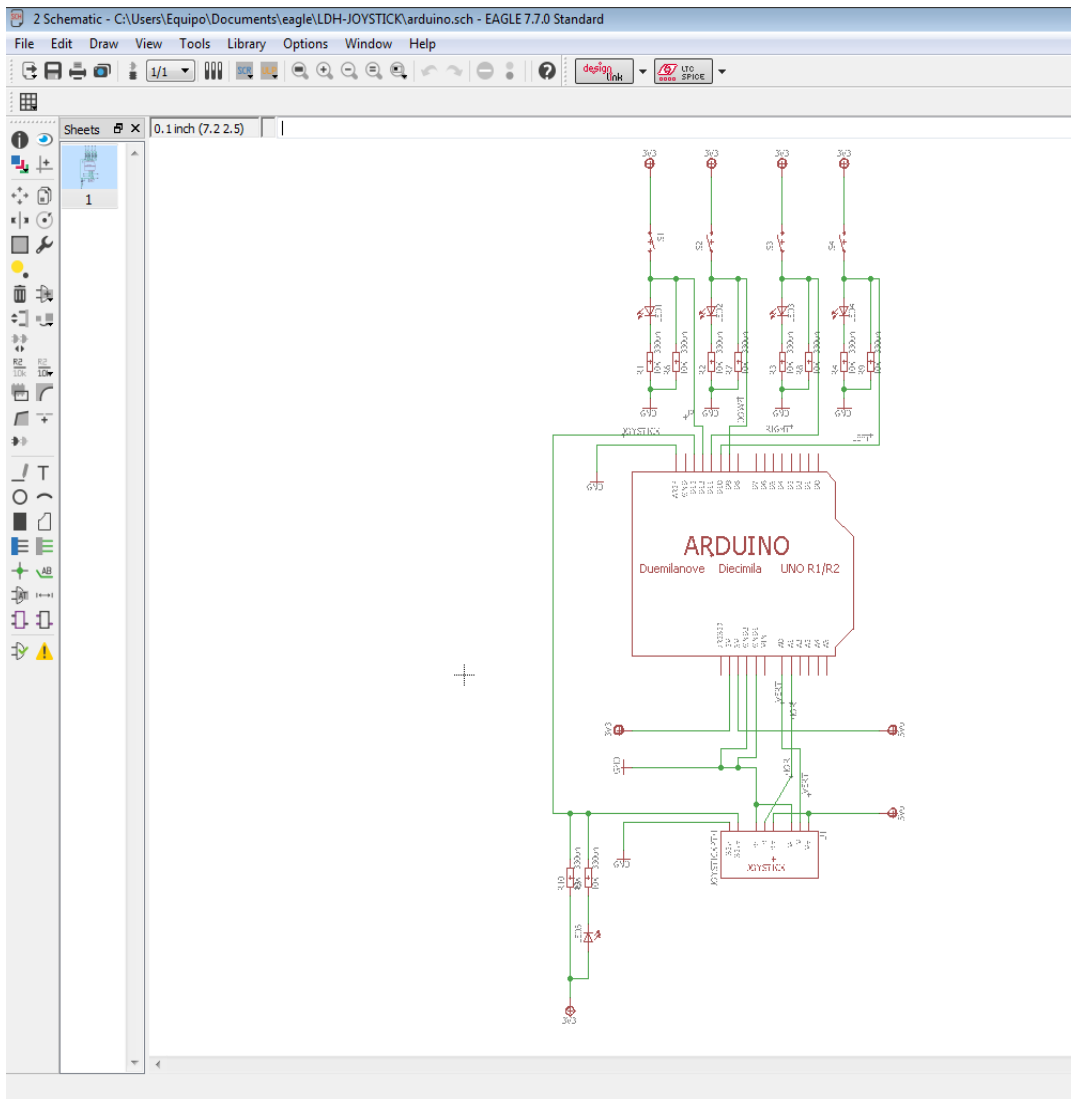
Para ello, se diseñarán 2 tipos de PCB, uno compatible con Arduino y otro compatible con ZPuino (Papilio). Una vez diseñados ambos diseños, se llevará a cabo la fabricación de una de ellas, en este caso, para Arduino.

Para que esto sea posible, los elementos con los que debemos familiarizarnos son:

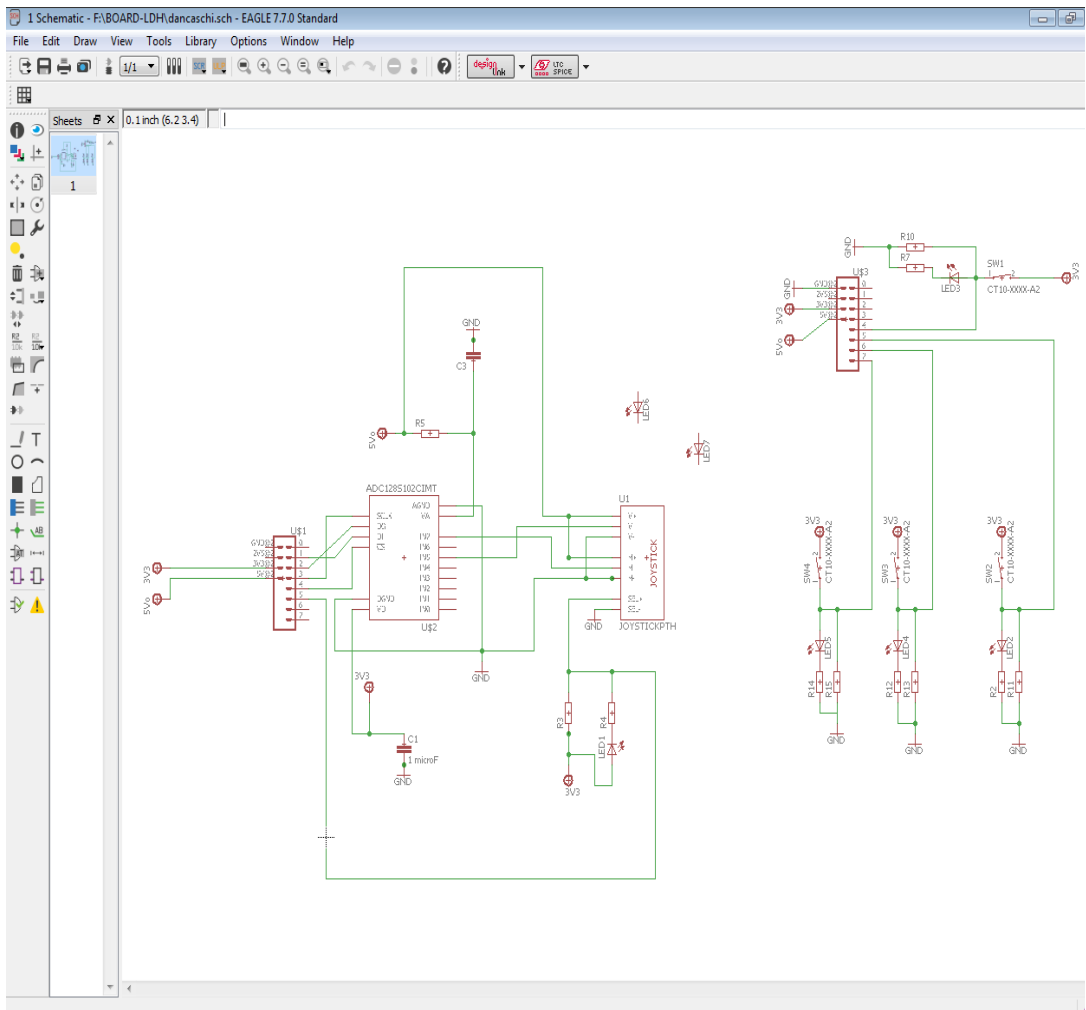
- EAGLE (Herramienta CAD elegido para el diseño de la PCB).**
- Componentes Through-Hole y SMD.**
- Flux.**
- Soldador.**
- Aleación de soldadura (Estaño-Plomo).**
- Malla de desoldadura.**
- Polímetro (para comprobar que las interconexiones soldadas son correctas).**

1.2 Diseño Esquemáticos:

Esquemático para implementación solo arduino



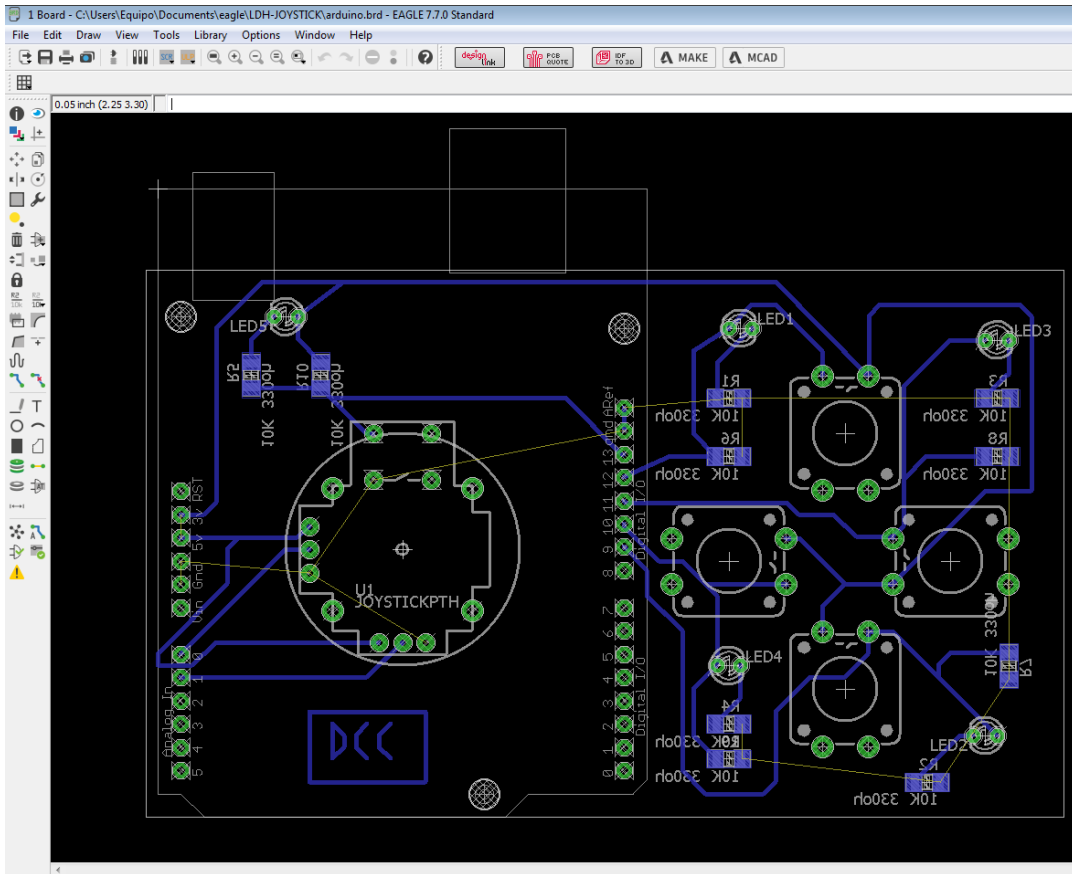
Esquemático implementación para Papilio



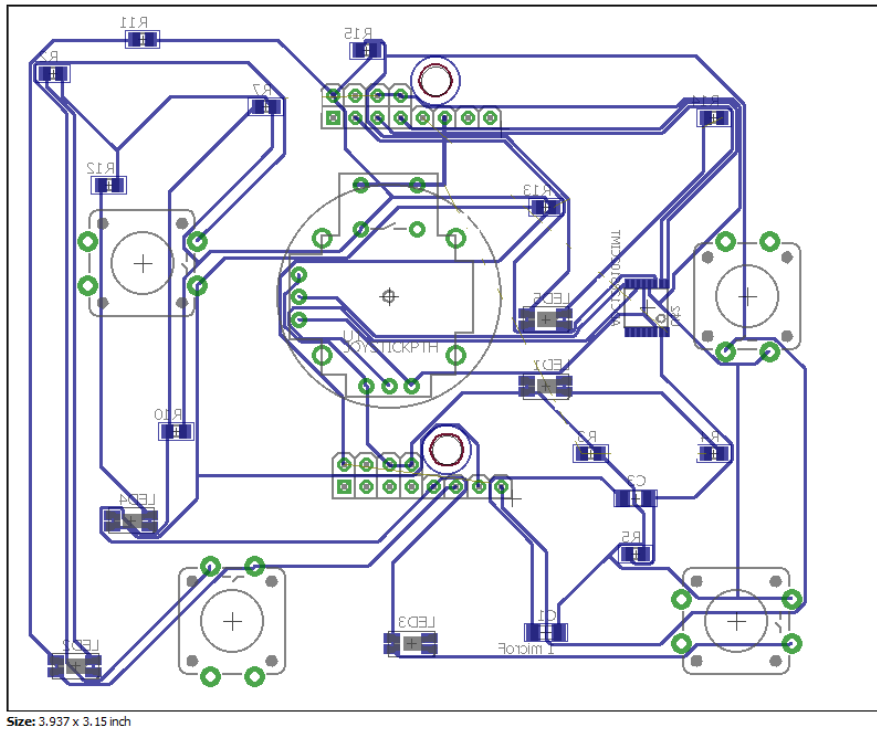
1.3 Diseño PCB:

Una vez diseñados los esquemáticos, se procederá a elaborar el diseño final de como quedará la board.

Board Arduino:



Board Papilio:



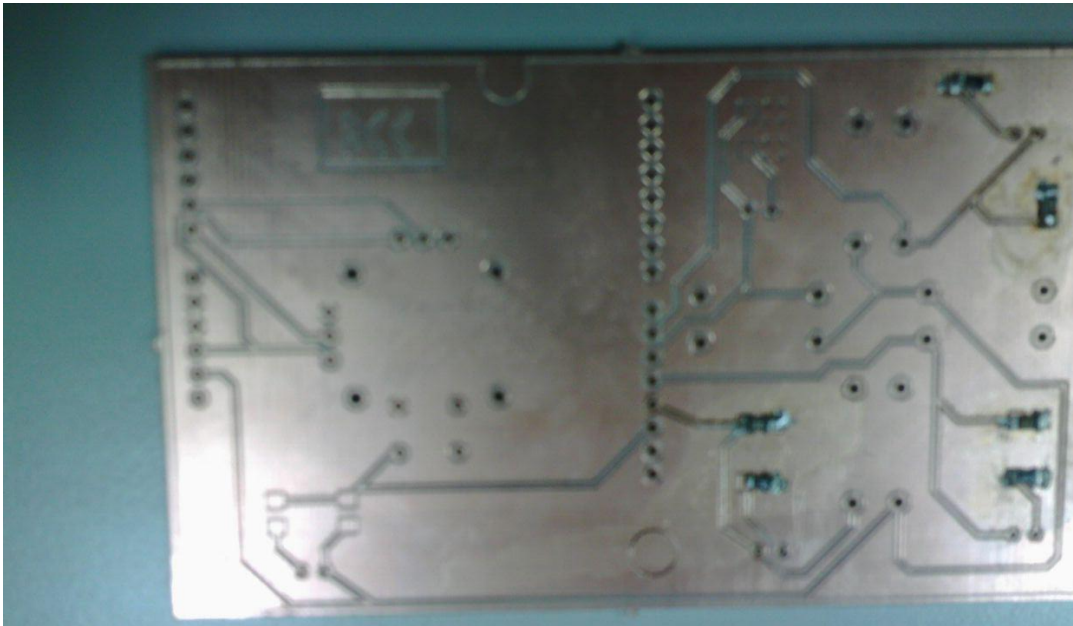
1.4 Fase de fabricación:

Una vez ya diseñadas las boards, se procederá al chequeo de errores de la placa pasándole el DRC. Se corrigen los posibles errores y después se crearán los gerbers, que son los archivos necesarios para la fabricación de la placa, ya que contienen toda la información respecto a ella.

Los archivos que necesitamos generar son:

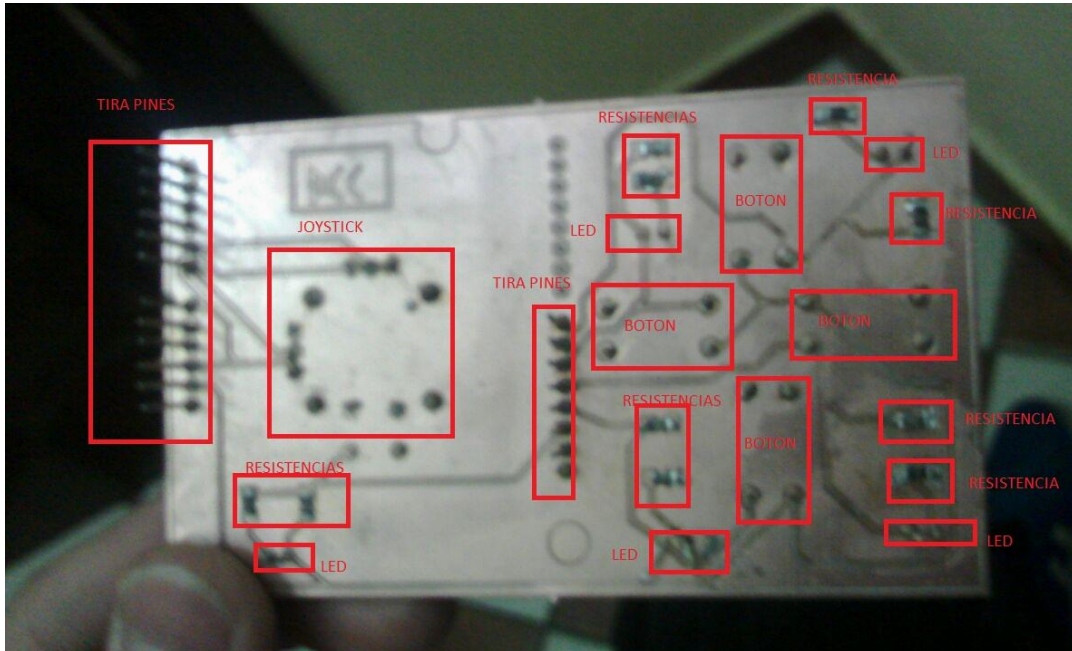
Capa de componentes	*.cmp
Capa de board	*.brd
Capa de diámetro de taladros	*.dri
Capa de plano de taladros	*.drd
Capa de soldadura	*.sol

Con estos archivos , la Protomat S62, nos fabricará la placa, guiándose por las medidas, y dimensiones proporcionadas en los gerbers. El resultado es el siguiente:

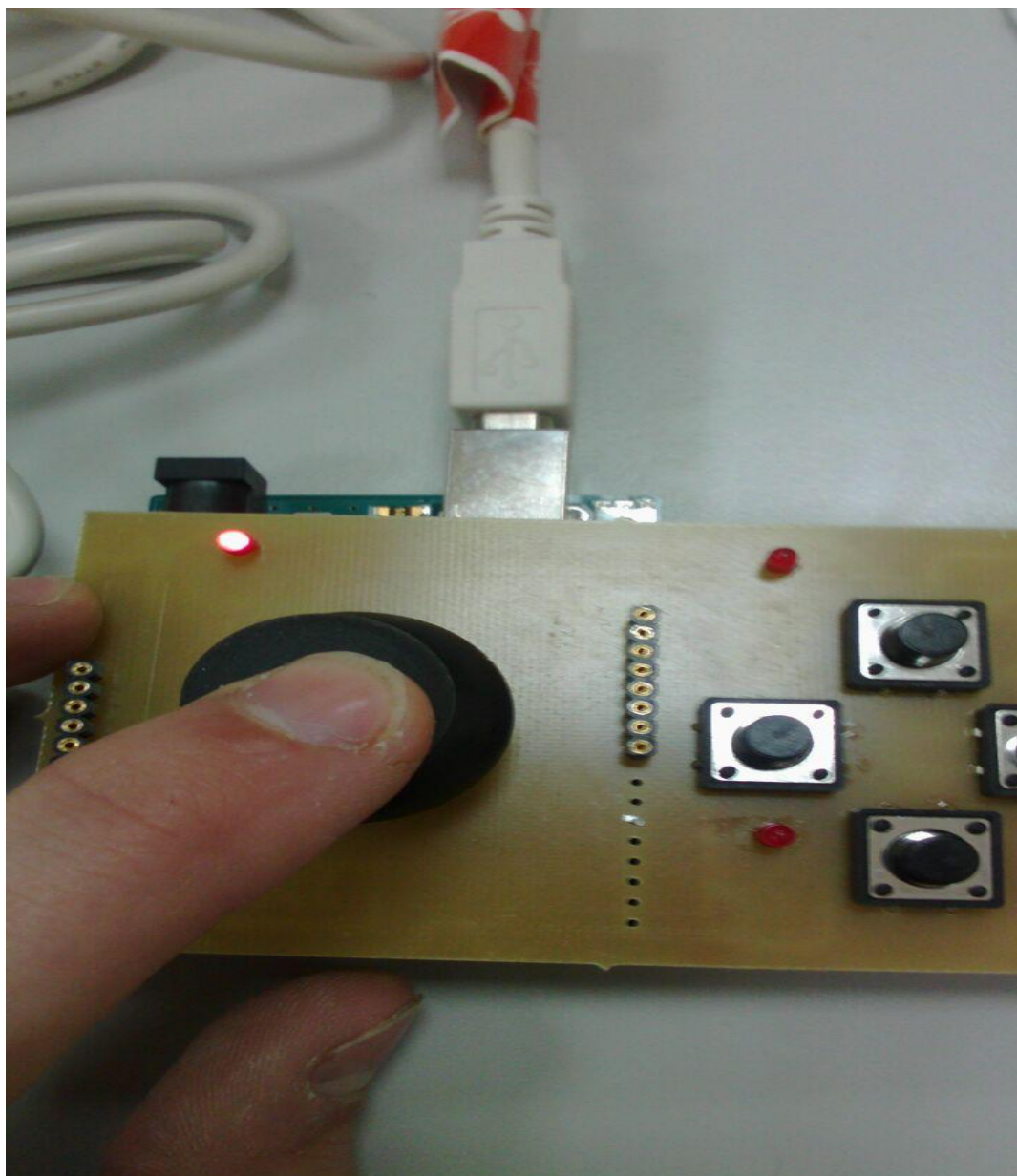


1.5 Fase de ensamblaje:

Comenzamos ensamblando las resistencias tipo SMD. Una vez terminados de colocar todas las resistencias, procedemos a soldar los componentes through-hole (leds, botones, pines y por ultimo el joystick). El resultado final es el siguiente:



El resultado final es el siguiente:



1.6 Fase de test:

-Timbrado: Con el polímetro se procederá a verificar que cada una de las soldaduras esta conectada a la pista que se debe y que no existan contactos indeseados entre tierra o con otras conexiones.

-Alimentación: Trabajaremos con 3.3Vcc y 5Vcc.

La parte digital (pulsadores) deberán variar entre 0 y 3.3Vcc, y que se enciendan sus correspondientes leds.

Nota: El valor del joystick está invertido respecto a los botones.

La parte analógica trabajara con 5 Vcc. El valor del joystick en reposo será de 2.5Vcc y variará entre 0 y 5Vcc a medida que se mueva, tanto para el ejeX como para el ejeY.

-Prueba código arduino: Una vez comprobados los 2 pasos anteriores , se procederá a probar el mando sobre la arduino, donde cargaremos un pequeño programa para verificar el correcto funcionamiento del joystick y de los botones.

El código es el siguiente:

```

int ejex;
int ejeY;
int botup;
int botleft;
int botright;
int botdown;
int botjoystick;

void setup(){
  pinMode(A0,INPUT);
  pinMode(A1,INPUT);
  pinMode(9,INPUT);
  pinMode(10,INPUT);
  pinMode(11,INPUT);
  pinMode(12,INPUT);
  pinMode(13,INPUT);
  Serial.begin(9600);
}

void loop(){
  ejex=analogRead(A0);
  ejeY=analogRead(A1);
  botdown=digitalRead(9);
  botleft=digitalRead(10);
  botright=digitalRead(11);
  botup=digitalRead(12);
  botjoystick=digitalRead(13);
  Serial.print("Posicion X= ");
  Serial.print(ejex);
  Serial.print("Posicion Y= ");
  Serial.print(ejeY);
  Serial.print("Boton down = ");
  Serial.print(botdown);
  Serial.print("Boton left = ");
  Serial.print(botleft);
  Serial.print("Boton right = ");
  Serial.print(botright);
  Serial.print("Boton up = ");
  Serial.print(botup);
  Serial.print("Boton joystick = ");
  Serial.print(botjoystick);
  delay(500);
}

```

Tras añadir dicho código a la placa arduino, se testean los valores de los botones y del joystick para su verificar su funcionamiento correcto.

Aplicaciones Lugares

/dev/ttyACM0

Enviar

Posicion Y= 459
Boton down = 1
Boton left = 0
Boton right = 0
Boton up = 0
Boton joystick = 1
Posicion X= 498
Posicion Y= 459
Boton down = 1
Boton left = 0
Boton right = 0
Boton up = 0
Boton joystick = 1
Posicion X= 498
Posicion Y= 458
Boton down = 0
Boton left = 0
Boton right = 0
Boton up = 0
Boton joystick = 1
Posicion X= 498
Posicion Y= 458
Boton down = 1
Boton left = 0
Boton right = 0
Boton up = 1
Boton joystick = 1
Posicion X= 499
Posicion Y= 459
Boton down = 1
Boton left = 0
Boton right = 1
Boton up = 0
Boton joystick = 1
Posicion X= 499
Posicion Y= 458
Boton down = 1
Boton left = 0
Boton right = 0
Boton up = 0
Boton joystick = 1

☒ Desplazamiento automático

No hay fin de línea

9600 baud

br4v0s@br4v0s: ~

sketch_jan26a | Ardui...

arduino.txt (~/Escri...

/dev/ttyACM0 (como ...

Aplicaciones Lugares

/dev/ttyACM0

Enviar

Posicion Y= 458
Boton down = 1
Boton left = 0
Boton right = 0
Boton up = 1
Boton joystick = 1
Posicion X= 499
Posicion Y= 459
Boton down = 1
Boton left = 0
Boton right = 1
Boton up = 0
Boton joystick = 1
Posicion X= 499
Posicion Y= 458
Boton down = 1
Boton left = 0
Boton right = 0
Boton up = 0
Boton joystick = 1
Posicion X= 499
Posicion Y= 458
Boton down = 1
Boton left = 0
Boton right = 0
Boton up = 0
Boton joystick = 1
Posicion X= 1022
Posicion Y= 775
Boton down = 1
Boton left = 0
Boton right = 0
Boton up = 0
Boton joystick = 1
Posicion X= 0
Posicion Y= 191
Boton down = 1
Boton left = 0
Boton right = 0
Boton up = 0
Boton joystick = 1

☒ Desplazamiento automático

No hay fin de línea

9600 baud

br4v0s@br4v0s: ~

sketch_jan26a | Ardui...

arduino.txt (~/Escri...

/dev/ttyACM0 (como ...