

Digo todo esto porque he visto varios ejemplos de aficionados a despanzurrar artefactos y modificarlos completamente, o a veces solo para “recuperar” componentes, como si se tratara de nutrientes para futuros proyectos inciertos.

Es preferible que descansen en paz (los artefactos, claro está) completos, bien guardados si se tiene lugar, hasta que llegue por fin el día en que por ejemplo un niño pregunte por alguno de ellos.

¿Papá, qué es eso? - Un teléfono, hijo.  
Eso que ves es un teléfono y si colocás el dedo índice ahí vas a ver que podés girarlo y marcará un número. - ¿Y Funciona Papá, lo podemos usar?  
Si, claro...estem, creo que habría que pelar los cables porque no tenemos una ficha como esa casa, pero.. bueno mejor lo dejamos para otro día. Esta

conversación es absolutamente real. Y el aparato que funcionó en lo de mi bisabuela fabricado en Suecia hace muchísimo tiempo todavía espera a que algún día encuentre o conforme algo parecido a un conector que le sirva, porque realmente no queremos alterar su enchufe original.

Si estás en panne en medio del desierto del Sahara y se te aparece uno de estos personajes, seguramente te pida que por favor le dibujes un cordero. La solicitud no es como una orden judicial, pero es igual de irrevocable.

Papá yo me acuerdo que vi que vos tenías guardado en una caja un joystick ¿Puedo verlo?

Zas! Ya tiene 7 años (el niño, el joystick más de 20) y le interesa muchísimo todo lo que pueda llamarle la atención, que por cierto es bastante. Y además acaba de descubrir que existen simuladores de

vuelo muy realistas y que por más destreza que se tenga con el teclado...  
¿Puedo, puedo, puedo verlo?

¡MAYDAY!

Dentro de la caja efectivamente descansa el joystick, impecable, con su escasísimo manual de instrucciones, y su boleta de compra fechada en diciembre de 2000.

Ahá! **DB-15** En otras palabras, una ficha para conectarse al puerto de juegos de la placa de sonido de la vieja computadora, con 15 pines!!!

Seguro que **Arduino** nos dará una mano; veremos.

Segunda pregunta  
(Puesto que la primera fue ¿Puedo verlo?  
repetida varias veces)

¿Se puede conectar este dispositivo o similar a través de una placa Arduino y obtener algún resultado?

Respuesta: Si

El sentido de incluir esta pregunta y su respuesta aquí, es para quienes hayan visto que funciona, por ejemplo el módulo KY-023 de joystick en Arduino UNO, con el cual se pueden encender leds dispuestos en cruz o también actuar sobre servomotores, pero no se comportaría como un joystick conectado a una computadora según lo esperado.

Debo aclarar, además, que lo primero que hicimos fue ir a nuestra caja de 37 sensores para verificar si teníamos el famoso KY-023 para realizar los primeros ensayos antes de avanzar con el grande. Porque a esta altura ya había inferido que deberían existir bastantes líneas de código para leer y algunas bibliotecas

que descargar.

Pero no sabía que deberíamos adquirir algo nuevo para nuestra querida y modestamente provista colección de componentes.

Entonces Tercera pregunta

***¿Podría el joystick (con su enchufe de 15 pines) a través de la placa Arduino comunicarse con el programa de simulación de vuelo y utilizar todas sus características (botones, timón, acelerador)?***

Respuesta: Sí pero no pero sí  
Sí, pero NO si utilizas las placas habituales como UNO, Nano o Mini, basadas en el chip ATmega328 (\*).  
Y la respuesta es **SI** si eliges la placa **Pro Micro** equipada con el chip **ATmega32U4** (\*\*), y en este caso particular

encontramos y recomendamos la variante que en lugar del conector micro usb trae un **mini usb**, más manejable y resistente. Aunque la conseguimos por menos de **u \$s 5**, recuerda que los usuarios pueden tener menos de 8 años!

(\*) Hay una manera de hacer esto, consiste en reprogramar el chip traductor de protocolos usb ATmega16U2 -que es el otro o segundo integrado- en la placa UNO, pero excede este trabajo (también mis conocimientos). De paso digo, el chip ATmega32U4 es muy poderoso y hace de la placa Pro Micro una simpática obra maestra.

(\*\*) Arduino Leonardo y Micro también son placas de microcontrolador basadas en ATmega32U4, y por supuesto se comportarían en el mismo sentido.

### ***Ok, entonces ¿Qué necesitamos además de la Pro Micro?***

Soldar. No hay que ser un experto ni mucho menos, pero la placa Pro Micro (a menos que ya la tengamos) puede venir con los pines sin soldar. Y además es necesario soldar resistencias, varios

cables, y terminales de la ficha entre sí.

Una ficha DB-15 hembra de dos filas, con terminales para soldar.

Cuatro resistencias de  $100\text{K}\Omega$

Seis cables de los que se usan en los prototipados de Arduino llamados Dupont del tipo macho-macho de 20cm (que se cortarán por la mitad para conectar a la protoboard sus dos terminales y soldar los extremos opuestos al conector DB-15) preferentemente de distintos colores, porque una vez soldados y estando la ficha fijada en su lugar, es complicado reconocerlos y conectar al pin correspondiente.

Una mini protoboard de 170 puntos (suele venir con cinta adhesiva doble faz en su base).

Alicates, pelacables, artilugios de soldar,

lupa, buena luz, etc.

Cualquier pieza o armado de piezas que permita algo similar a lo que conseguimos con partes de un juego tipo Meccano, y cuyo peso final de 75 gramos alcanza para que se mantenga estable sobre la mesa aún si se mueven los cables.

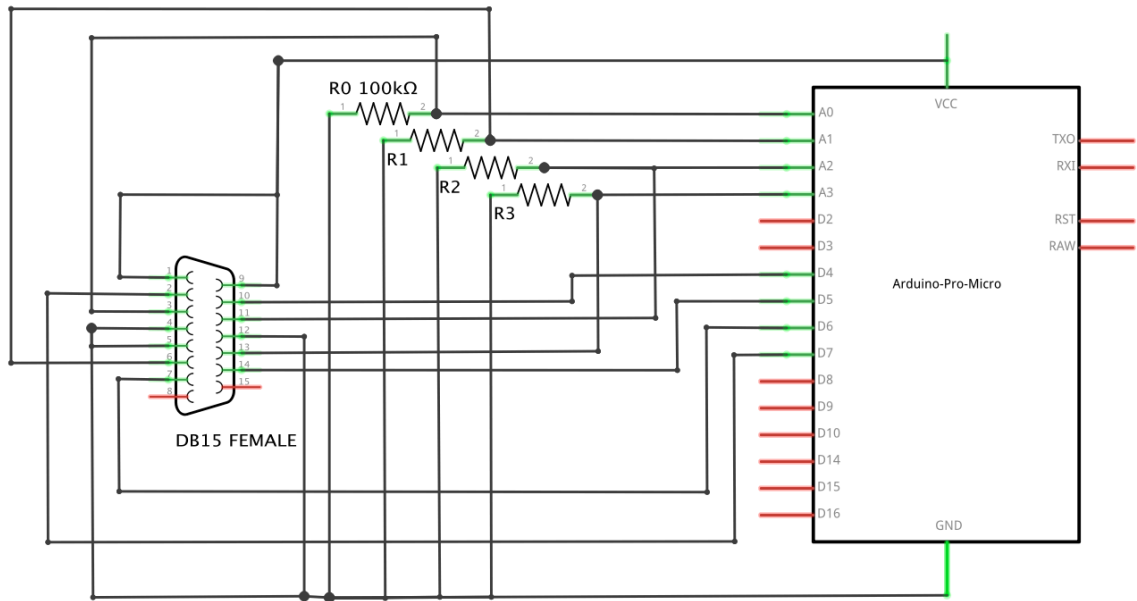
*Casi siempre hay más de una manera de hacer las cosas. Nosotros optamos por desarrollar un dispositivo desmontable. En cuestión de horas se nos puede ocurrir algo, como por ejemplo un avión RC, y necesitaríamos recuperar nuestra placa si fuera útil para el nuevo proyecto. Aunque mis hijos están encantados de "prestarme" sus juguetes, incluidos los Arduinos, para que podamos hacer nuevos experimentos, todos sabemos que deben volver a su estado original de inmediato al menor requerimiento. Saben exactamente cuál es y qué lugar ocupa cada pieza de Lego que fue usada tal vez como manecilla de un reloj, o como separador plástico entre placas y soportes. Aunque no se trataría de algún prodigio de memoria, sino de un implacable poder de fiscalización!*

*Por supuesto que si no se padece el acoso de un par de Sheriffs de Nottingham, o se busca algo de tamaño más reducido y permanente, soldar todo a una plaqueta experimental, sin que sea necesario tener el circuito impreso, tal vez sea la mejor solución. Entre las*

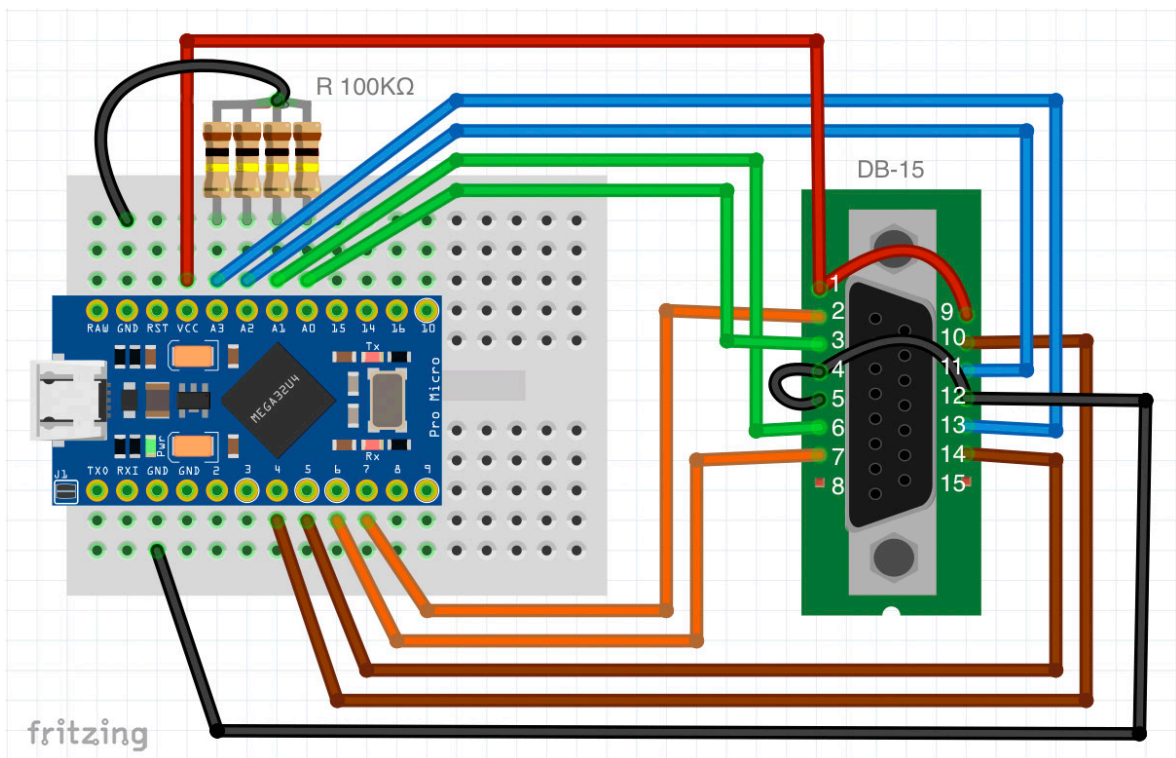
*páginas que he leído como investigación para este proyecto, encontré mención a distorsiones en las señales recibidas por la PC a causa de fallas de conexión a la protoboard, por lo que se recomienda soldar. Aclaro que por ahora en nuestras pruebas funciona todo perfecto tal cual lo hemos dispuesto.*

Este joystick consta de cuatro potenciómetros, dos que responden a los ejes X e Y , otro para acelerador y otro para timón, con valores de resistencia variable que podrían estar de entre cero y 100 k $\Omega$  aproximadamente. En cada uno de ellos, un extremo de potenciómetro está conectado al pin de entrada de 5 volts y el contacto del centro está conectado a la entrada analógica del joystick. El otro contacto del potenciómetro se deja al aire sin conectar. Esta disposición hace necesaria la descarga a tierra a través de cuatro resistencias de 100k $\Omega$ , una para cada entrada analógica desde los cuatro potenciómetros del joystick, como veremos en el circuito esquemático. Los pines digitales corresponden al manejo de

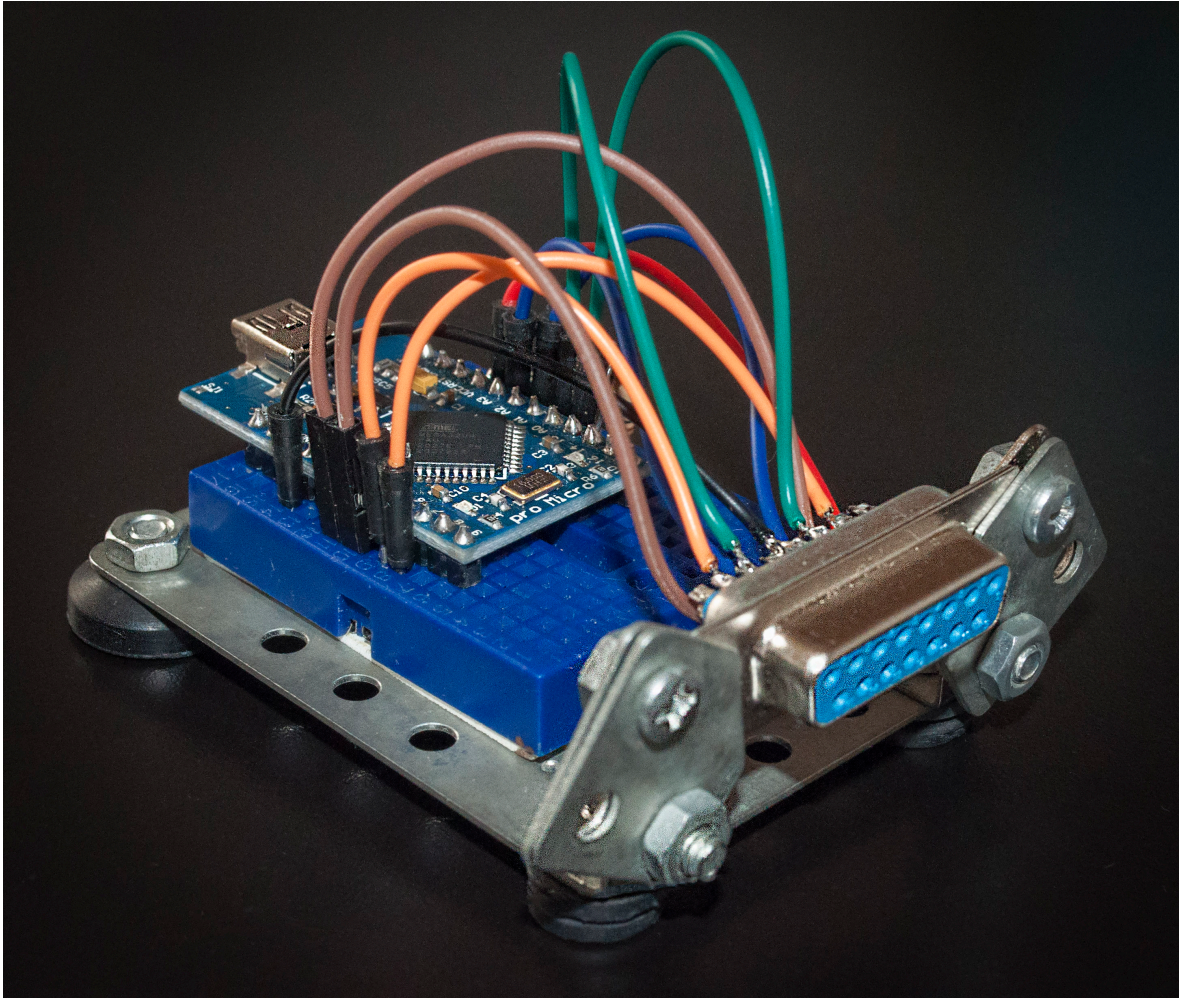
# los botones.

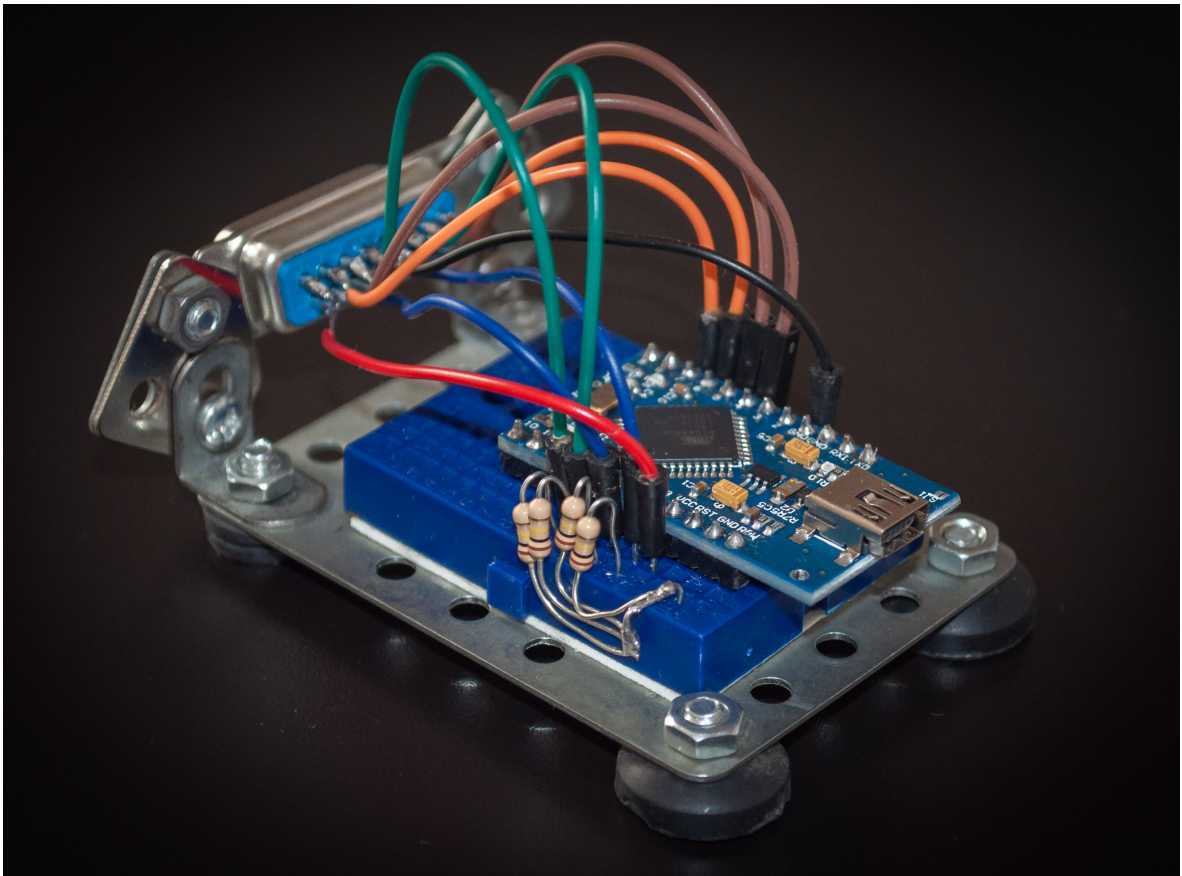
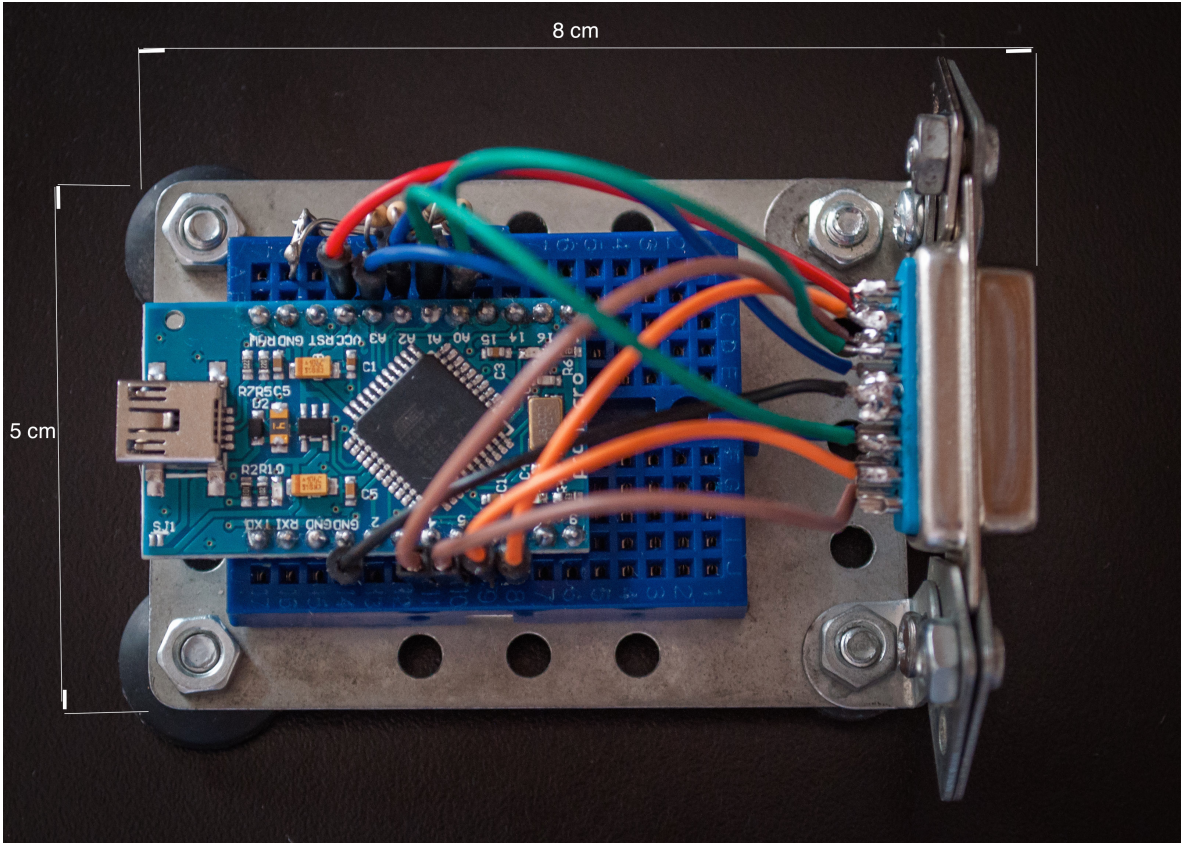


fritzing



fritzing





***Para utilizar la placa Pro Micro es necesario descargar e instalar un driver.***

Aquí el link para su descarga e instrucciones de instalación tanto para Linux , como para Mac o Windows:  
<https://learn.sparkfun.com/tutorials/pro-micro--fio-v3-hookup-guide#installing-windows>

***Y ahora el código para el IDE Arduino:***

```
// To make use of this Joystic Library, we  
must first download it at the address:
```

```
// https://github.com/MHeironimus/  
ArduinoJoystickLibrary
```

```
#include <Joystick.h>
```

```
#define BUTTON_1 4
```

```
#define BUTTON_2 5
#define BUTTON_3 6
#define BUTTON_4 7
```

```
#define DELAY 100
```

```
Joystick_ Joystick;
```

```
void setup() {
  pinMode(BUTTON_1, INPUT_PULLUP);
  pinMode(BUTTON_2, INPUT_PULLUP);
  pinMode(BUTTON_3, INPUT_PULLUP);
  pinMode(BUTTON_4, INPUT_PULLUP);
  Joystick.begin();
  Joystick.setYAxisRange(1023, 0); // You
  can reverse the order of these values to
  shift up or down
  Joystick.setXAxisRange(1023, 0);
  Joystick.setThrottleRange(1023, 0);
  Joystick.setRudderRange(1023, 0);
}
```

```
void loop() {
  Joystick.setXAxis(analogRead(A0));
```

```
Joystick.setYAxis(analogRead(A1));  
Joystick.setThrottle(analogRead(A2));  
Joystick.setRudder(analogRead(A3));
```

```
if (digitalRead(BUTTON_1) == LOW) {  
    Joystick.setButton(0, 1);  
}  
else{  
    Joystick.setButton(0, 0);  
}
```

```
if (digitalRead(BUTTON_2) == LOW) {  
    Joystick.setButton(1, 1);  
}  
else{  
    Joystick.setButton(1, 0);  
}
```

```
if (digitalRead(BUTTON_3) == LOW) {  
    Joystick.setButton(2, 1);  
}  
else{  
    Joystick.setButton(2, 0);  
}
```

```
if (digitalRead(BUTTON_4) == LOW) {  
  Joystick.setButton(3, 1);  
}  
else{  
  Joystick.setButton(3, 0);  
}  
  
delay(DELAY);  
}
```

-----  
-----

A partir de ahora puedes volar solo, es parte del juego.