

3. Entradas de Echidna

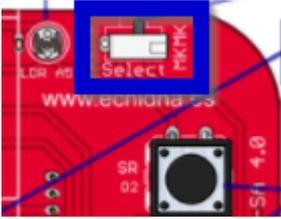
- 3 Entradas de Echidna
- 3.1 MONTAJE 4 Interruptor luz
- El LDR en A5
- 3.3 Joystick
- Acelerómetro

3 Entradas de Echidna

Aquí es donde vamos a sacar todo el jugo a esta Shield.

Nota: Si has hecho el anterior programa Grabar a Arduino, recuerda dejarlo preparado para que haga caso a mBlock: Conectar-Actualizar Firmware.

Nota: Acuerdate en toda esta sección de poner la Echidna en modo Sensor



Todos los programas de este curso se encuentran en este repositorio:

<https://github.com/JavierQuintana/Echidna>

3.1 MONTAJE 4 Interruptor luz

3.1 MONTAJE 4 INTERRUPTOR LUZ.

El anterior programa ya es un ejemplo de uso de estos botones digitales que están conectados a los pines digitales 2 y 3 de Arduino. Ten en cuenta que *sólo* pueden leer niveles lógicos.

RETO1

Al pulsar el botón D2 se enciende los 3 leds del Echidna y al soltarlo se apagan. El programa “tunealo” un poco, quitando el oso como disfraz y añadiendo un objeto que cambie según D2

Solución

Elegimos un objeto que cambie el disfraz, y le ponemos el código



Y al arduino:



El programa lo tienes aquí <https://planet.mblock.cc/project/3228733>

RETO2

Al pulsar D2 tiene que encenderse los leds, y **sólo se apagan** si se vuelve a pulsar D2. O sea, utilizar sólo un pulsador para encender y apagar los leds.

Solución



El objeto que cambia el disfraz es igual que en RETO1

El programa lo tienes aquí <https://planet.mblock.cc/project/3228741>

UNA CURIOSIDAD Fíjate como hay un retraso de 0.5 segundos para que "te de tiempo de apartar el dedo del pulsador" en caso contrario, volvería al otro estado. Si no sabes lo que

quiero decir, quita la instrucciones "Espera 0.5 segundos" y verás qué pasa.

UN POCO DE PARÉNTESIS TEÓRICO Fíjate en el enunciado del RETO1: “al pulsar el botón D2 se enciende y al soltarlo se apaga” ES UNA MÁQUINA LÓGICA pues el estado de la máquina sólo depende de las entradas (en este caso de un botón): Pulsar la entrada (botón D2) produce una salida concreta (encender leds).

CONTINUAMOS ...

Pero tal y como está redactado, el RETO2 tiene que memorizar el estado anterior, no es trivial el enunciado “Al pulsar D2 tiene que encenderse los leds, y sólo se apagarán si se vuelve a pulsar D2.” ES UNA MÁQUINA SECUENCIAL pues el estado de la máquina depende de las entradas y de lo que ha pasado antes. Pulsar la entrada (botón D2) NO produce una salida concreta (depende si estaba apagado o encendido anteriormente).

No pasa nada si no lo entiendes del todo, es teoría.

La programación se complica **necesitamos añadir una variable que recuerde lo que ha pasado antes** la vamos a llamar `_encendido_` que recordará si está encendido los leds o no:

Muchos de nuestros aparatos electrónicos se encienden y se apagan con el mismo botón, así que a partir de ahora aprecia que su funcionamiento no es trivial.

El LDR en A5

Primero de todo ¿Qué es un LDR? Vamos a la Wikipedia:



Fotorresistor

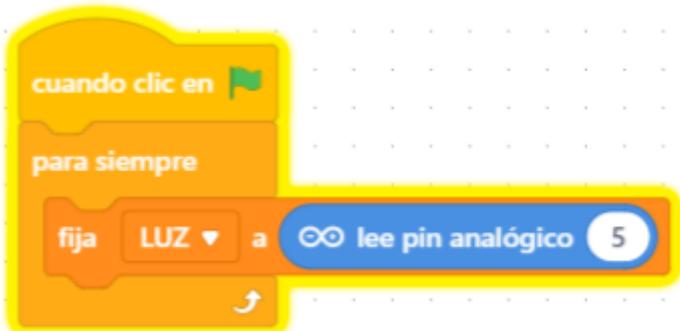
Una fotorresistencia es un componente electrónico cuya resistencia disminuye con el aumento de intensidad de luz incidente. Puede también ser llamado fotorresistor, fotoconductor, célula fotoeléctrica o resistor dependiente de la luz, cuyas siglas, LDR, se originan de su nombre en inglés light-dependent resistor. Su cuerpo está formado por una célula fotorreceptora y dos patillas.

[W More at Wikipedia \(ES\)](#)

Lo verás en el Echidna arriba un poco a la derecha, y está conectado a la entrada analógica del Arduino A5 y según <https://echidna.es/> los valores van desde 20 en ausencia de luz, hasta 1.000 con mucha luz.

Comprobar los límites

No os creáis al pie de la letra los límites oficiales del LDR! Los componentes electrónicos no son ideales, cada uno es particular, probar esos límites en vuestro Echidna, para ello vamos a utilizar nuestro mBlock y que nuestro simpático oso panda nos diga esos valores, con este sencillo programa:



DONDE **LUZ** ES UNA VARIABLE GLOBAL QUE LO LEEN TODOS LOS OBJETOS luego ahora el panda con este programa puede decirnos cuánta luz hay



El valor que leemos con máxima luz (utilizando una linterna) y el valor de máxima oscuridad (a tapar con el dedo, no te compliques) no llegan a los límites que se marcan oficiales en **Echidna** EN TU CASO PUEDEN SER OTROS !! pero parecidos.

El programa lo tienes aquí <https://planet.mblock.cc/project/3228782>

MONTAJE 5 Semáforo luminoso

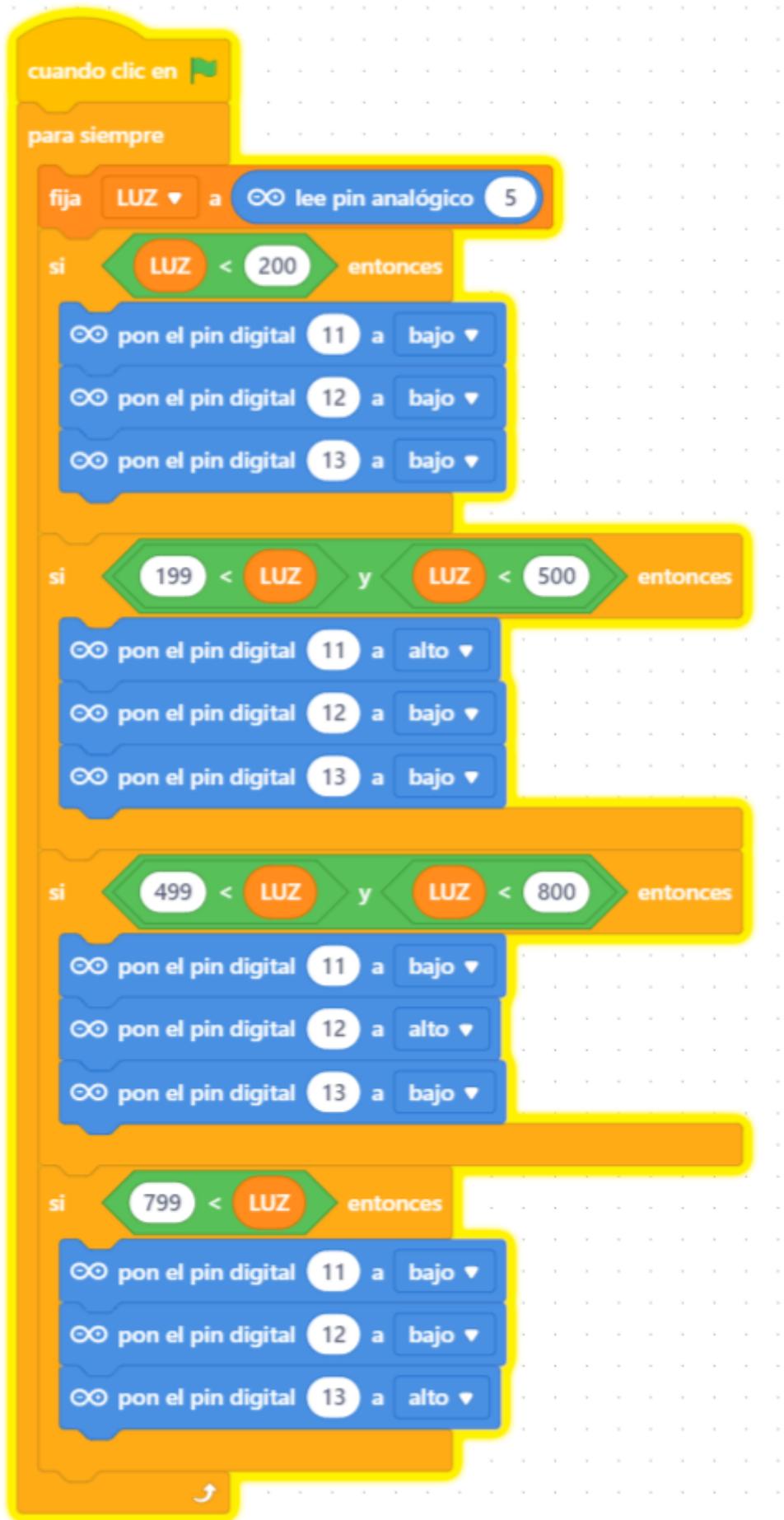
RETO Vamos a reutilizar esfuerzos: reutilizar el semáforo visto pero vamos a hacer que se iluminen los colores según la luz:

Semáforo	Luz
Todo apagado	Mucha oscuridad
Rojo	Oscuridad
Amarillo	Luz normal
Verde	Mucha luz

Solución

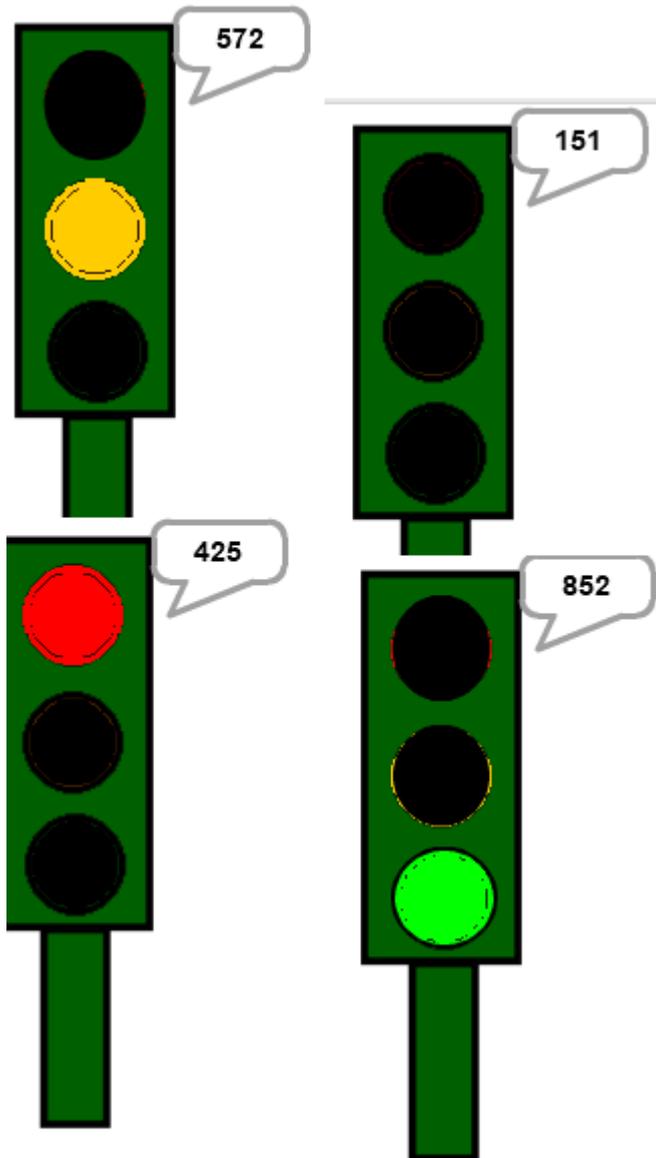
Según nuestros límites medidos en el anterior programa fijamos nuestras propias fronteras para cumplir el enunciado (puedes poner otros valores parecidos):

Semáforo	Luz	Límite inferior	Límite superior
Todo apagado	Mucha oscuridad	---	199
Rojo	Oscuridad	200	499
Amarillo	Luz normal	500	799
Verde	Mucha luz	800	---



El programa lo tienes aquí <https://planet.mblock.cc/project/3228793>

Reto: Añade un objeto que sea un semáforo, y que su disfraz cambie acorde a la luz y encima lo diga



El resultado lo puedes ver en [este vídeo](#):

https://www.youtube.com/embed/MX558VKV_pE

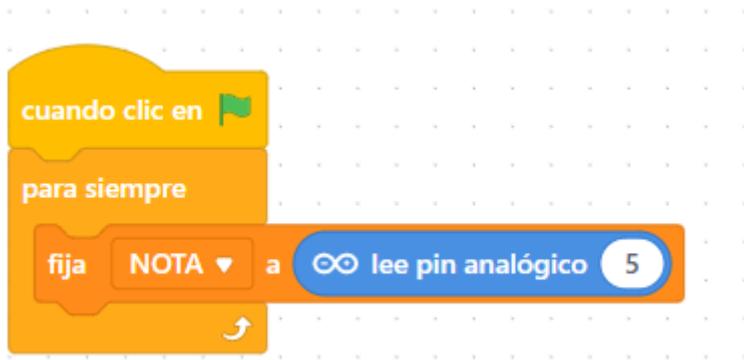
3.2.3 MONTAJE 6 Piano luminoso

Nuestro RETO es el siguiente:

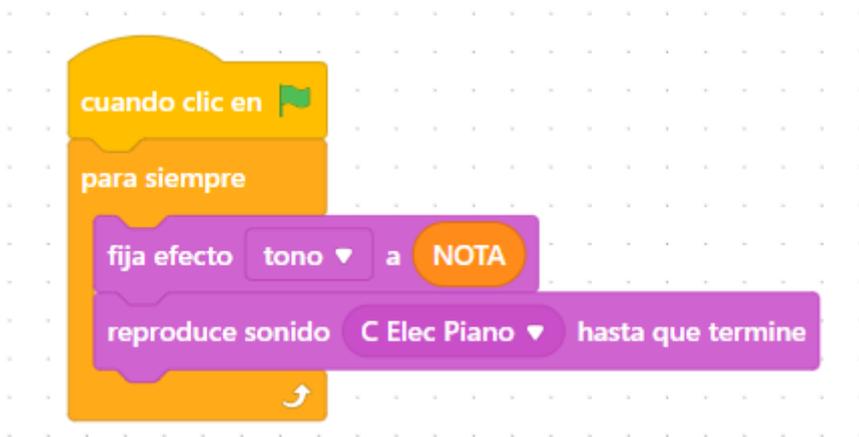
Realizar un programa que suene una nota diferente según la luz

Solución

En el Arduino el programa es



En el objeto, el panda es



Nota: se ha incorporado una música con un tono, el C Elec Piano

El programa lo tienes aquí

<https://planet.mblock.cc/project/projectshare/3228812>

Con mBlock versión 3 obsoleto

En esta versión no es tan sencillo ¿por qué? Porque hay cambios de escala: el LDR nuestro trabaja con valores distintos al de las notas, por lo tanto hay que hacer UN CAMBIO DE ESCALA, y esto necesita un apartado diferente, te recomendamos ver [3.2.3.1 Cambios de escala](#)

Solución

El programa es pues el siguiente



¿por qué lo hacemos con la opción de subir a Arduino? Porque la simulación va lenta (se oye tut-tut-tut) si lo subes al Arduino reproduce el tono correctamente. [VER COMO SUBIR AL ARDUINO](#).

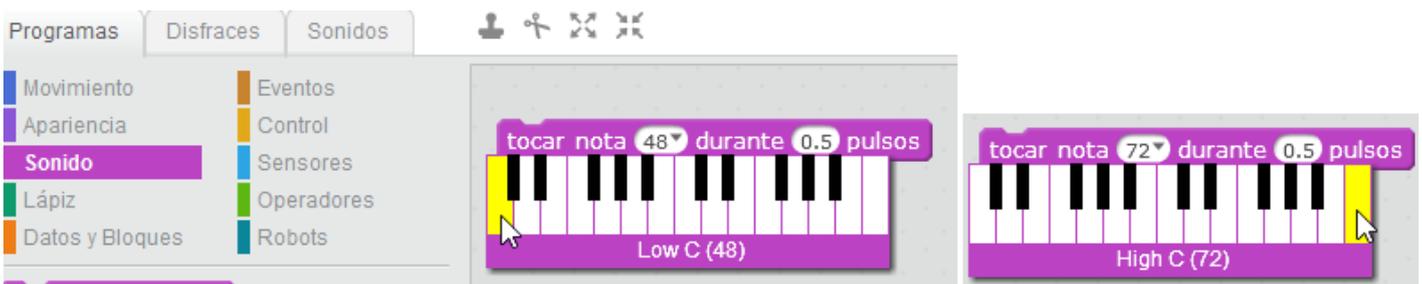
El resultado lo puedes ver en [este vídeo](#):

<https://www.youtube.com/embed/b7SSXn4q8WM>

3.2.3.1 Cambios de escala

Nos encontramos que:

- Los valores de entrada, es decir, el LDR trabaja con unos límites, que según hemos visto en [el apartado comprobar los límites](#), el nuestro va de 108 a 982, llamaremos a esta **variable de entrada X**
- Los valores de salida son los tonos, que mBlock trabaja con la norma americana (la europea es el típico Do-Re-Mi-Fa-Sol) y los americanos, son valores numéricos que van desde 48 hasta 72 (lo puedes comprobar con la instrucción “tocar nota” que está en Programas-Sonido). Llamaremos a la **variable salida de notas Y**:



Problema: ¿cómo convertimos X en Y?

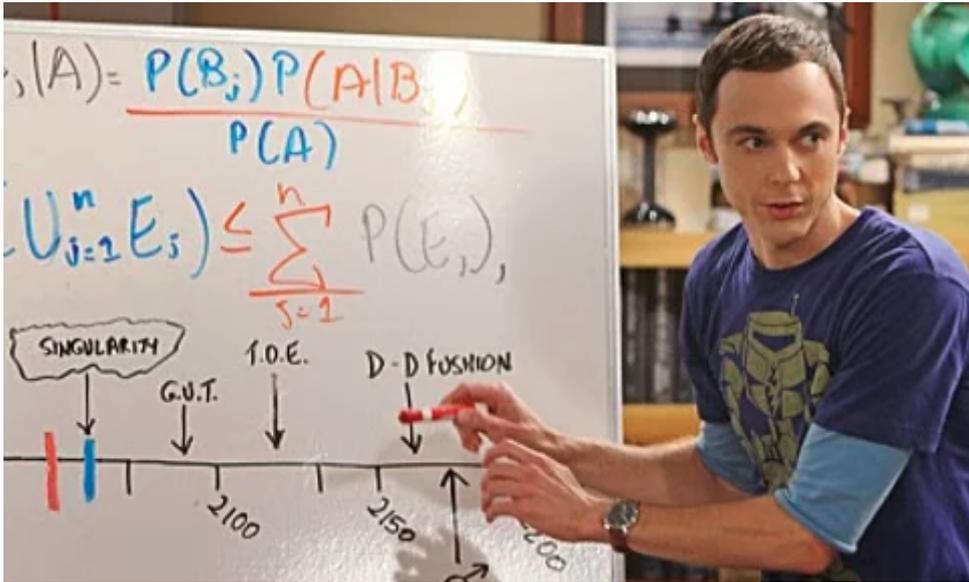
Matemáticamente es una recta con una pendiente m y una ordenada n :

$$Y = m \cdot X + n$$

Para calcular m y n tenemos que utilizar un sistema de ecuaciones dadas las condiciones límites de X e Y:

- Cuando $X = 108$ quiero que Y valga 48: $48 = m \cdot 108 + n$
- Cuando $X = 982$ quiero que Y valga 72: $72 = m \cdot 982 + n$

¡Ala! dos ecuaciones y dos incógnitas: calcula m y n



Buen ejercicio para los alumnos para que vean matemáticas aplicadas

No obstante, como esto lo repetiremos muchas veces los cambios de escala, hemos confeccionado [ESTA HOJA DE CÁLCULO](#) que te lo puedes descargar y facilita las cosas: Ponemos en las celdas amarillas los valores límites :

- $X_1 = 108$ $X_2 = 982$
- $Y_1 = 48$ $Y_2 = 72$

Y nos da los valores m y n automáticamente:

A	B	C
HOJA DE CALCULO DE CAMBIO DE ESCALAS		
$Y = mX + n$	CAMBIA LO AMARILLO	
Valores que tengo de x		
X1=	108	
X2=	982	
Valores que quiero de Y		
Y1=	48	
Y2=	72	
RESULTADOS		
m =	0,02745995	
n =	45,0343249	

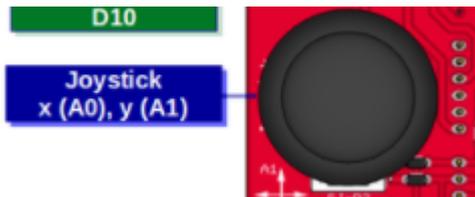
Luego la fórmula para el cálculo de la Y (las notas) en función de la luminosidad X es:

$$Y = 0.027 * X + 45$$

3.3 Joystick

El Joystick está conectado a los pines A0 y A1 y su valor varía desde 100 hasta 1024.

El pin A0 controla el eje X El pin A1 controla el eje Y



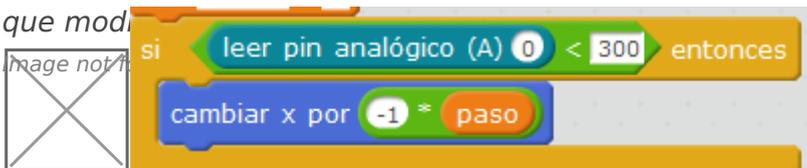
3.3.1 MONTAJE 7 Telesketch

RETO Vamos a realizar el mismo ejemplo que [este vídeo de Jorge Lobo](#) pero en vez de realizado en Snap4Arduino lo vamos a hacer en mBlock.

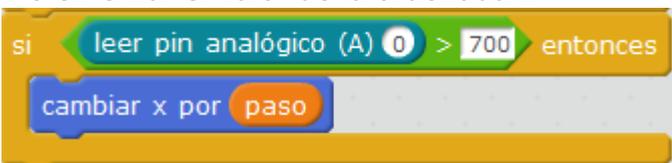
https://www.youtube.com/embed/Hx5DjQw7e_U

Simplemente vamos a comentar las condicionales principales:

Si A0 es menor que 300 es que estás inclinando el Joystick hacia la izquierda, luego el Sprite tiene que modificar incrementando su valor



Si A0 es mayor que 700 es seguro que estás inclinando el Joystick hacia la derecha luego hay que incrementar el valor de la ordenada X



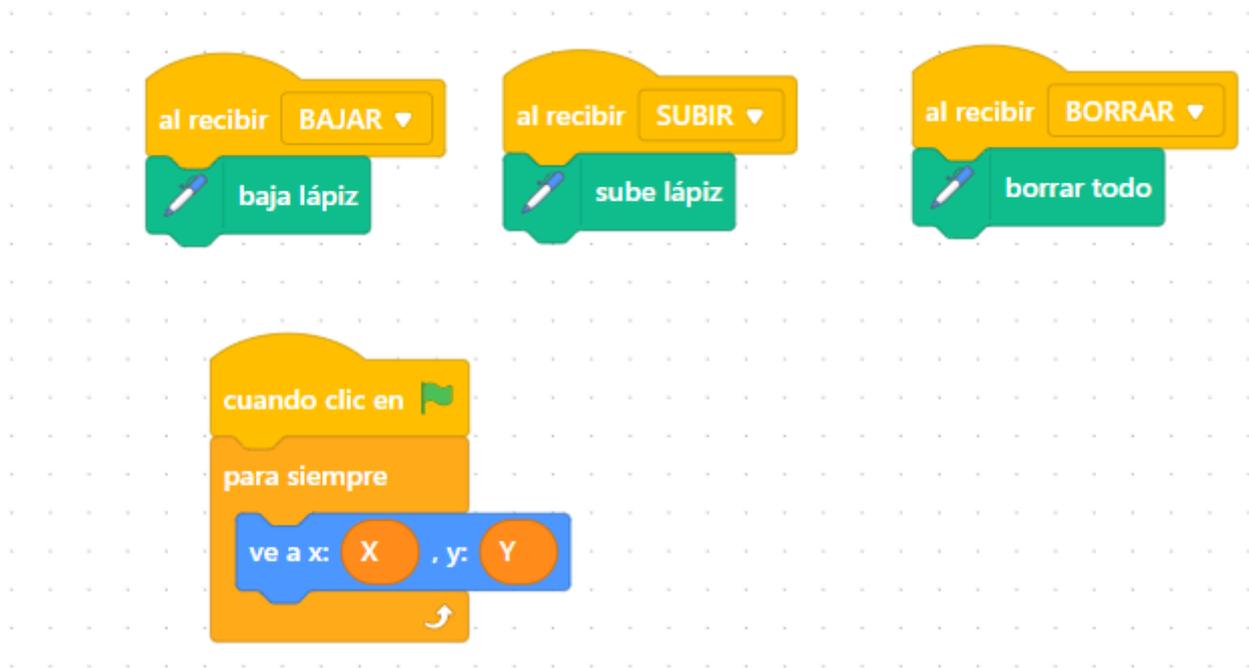
Para los valores del eje Y es igual, cambiando A0 por A1

SOLUCIÓN

Primero pondremo un objeto lápiz y añadiremos la extensión LAPIZ esto se entiende mejor con un vídeo

<https://www.youtube.com/embed/xft-6Nz3yzE>

Pero no hagas la programación de hacer el cuadrado. Sino este



Donde BAJAR, SUBIR Y BORRAR SON MENSAJES y X e Y son variables globales a todos los objetos.
En el arduino el programa será



El programa te lo puedes descargar aquí <https://planet.mblock.cc/project/3228849>

VA MUY LENTO El resultado lo puedes ver en [este vídeo](#):

<https://www.youtube.com/embed/jzyd5cPb2-Y>

Por curiosidad puedes ver el mismo programa en **Snap4Arduino** en este vídeo:

<https://www.youtube.com/embed/j1lsYq6X5-U>

Puedes ver que es mucho **más rápido** que mBlock.

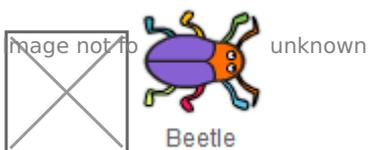
3.3.2 MONTAJE 8 Comebichos

No podemos dejar el Joystick sin hacer un videojuego !!



El siguiente **RETO** es: *Mover un Sprite "Bat" con el Joystick para atrapar un Beetle*

- Bat se tiene que mover con el joystick por toda la pantalla.
- Beetle se mueve horizontalmente desde el borde derecho hasta el borde izquierdo, y cuando llega al izquierdo, vuelve a aparecer en el derecho (y aleatoriamente desde cualquier altura)

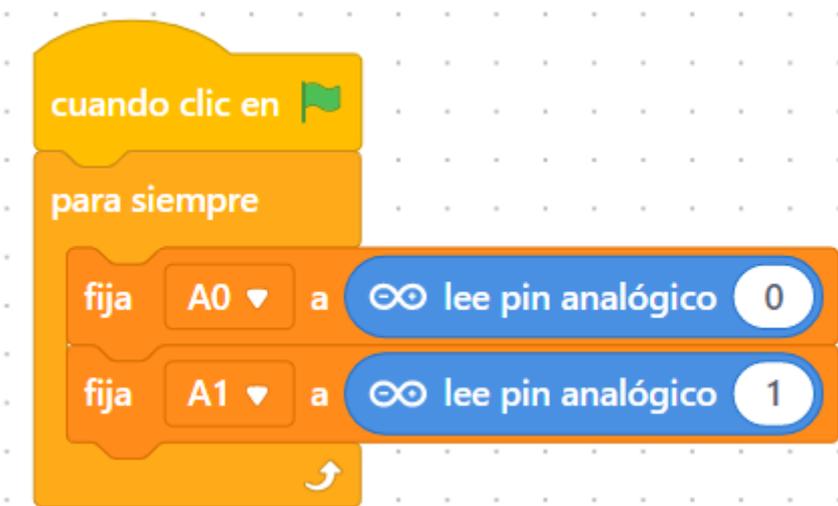


Solución

El bicho lo haremos mover bastante rápido y al azar en el eje Y para que lo tenga difícil Bat:



Y bat tiene que moverse con el joystick, que le pasaremos las variables A0 y A1. Por lo tanto el Arduino tiene este programa:



Mapeo

Hay que realizar un cambio de unidades, pues la variable A0 y A1 van desde 100 a 1024 y queremos que se traduzcan en las coordenadas -220+220 en el eje x y -150+150 en el eje y. La fórmula de conversión será

- $A0 = n + m \cdot \text{PinAnalogicoA0}$
- $A1 = n + m \cdot \text{PinAnalogicoA1}$ m, y n diferentes de los anteriores

Image not found or type unknown

Los valores de m y n lo hemos calculado con la [HOJA DE CÁLCULO](#) que comentamos anteriormente, los valores límites que hemos puesto son:

Para el eje X e Y :0-715

HOJA DE CALCULO DE CAMBIO DE ESC.

$x' = m x + n$	
Valores que tengo de x	
x1 =	100
x2 =	1024
Valores que quiero de x'	
x'1 =	-220
x'2 =	220

Para el ejemplo **HOJA DE CALCULO DE CAMBIO DE ESC.**

$x' = m x + n$	
Valores que tengo de x	
x1 =	100
x2 =	1024
Valores que quiero de x'	
x'1 =	-150
x'2 =	150
RESULTADOS	
m =	0,32467532
n =	-182,467532

Además hemos añadido la puntuación y el cambio de disfraz para que parezca que aletea:

Por lo tanto el programa del objeto murciélago es

The image shows two Scratch code blocks for a bat object. The first block is a 'when clicked' event block followed by a 'forever' loop. Inside the loop, there are two 'set x to' blocks with mathematical expressions: $0.47 * A0 - 267$ and $0.302 * A1 - 182$. Below these is a 'change costume' block and a 'repeat' block. The second block is a 'when clicked' event block followed by a 'set COMIDOS to 0' block, a 'forever' loop containing an 'if touching Beetle?' block with 'change COMIDOS by 1', 'play sound pop', and 'wait 0.5 seconds' blocks.

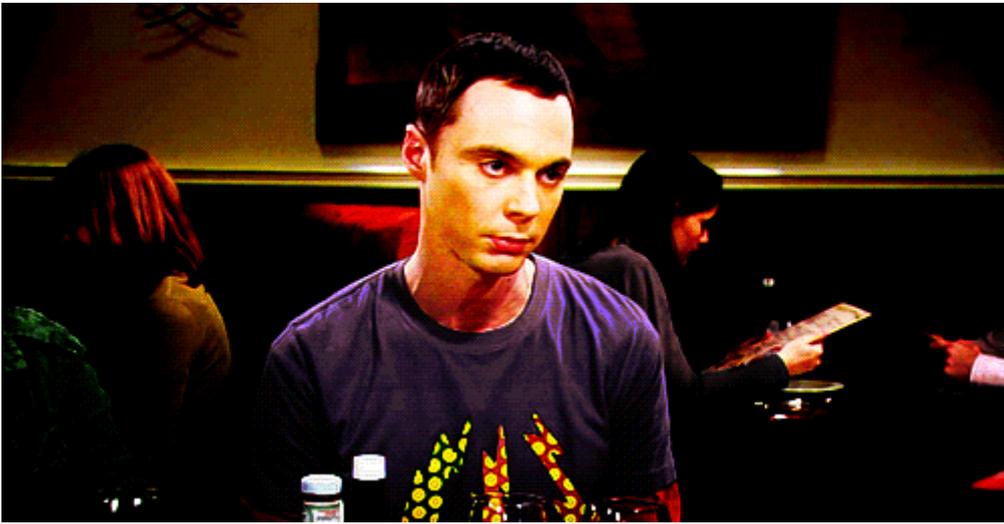
El programa completo lo puedes descargar aquí

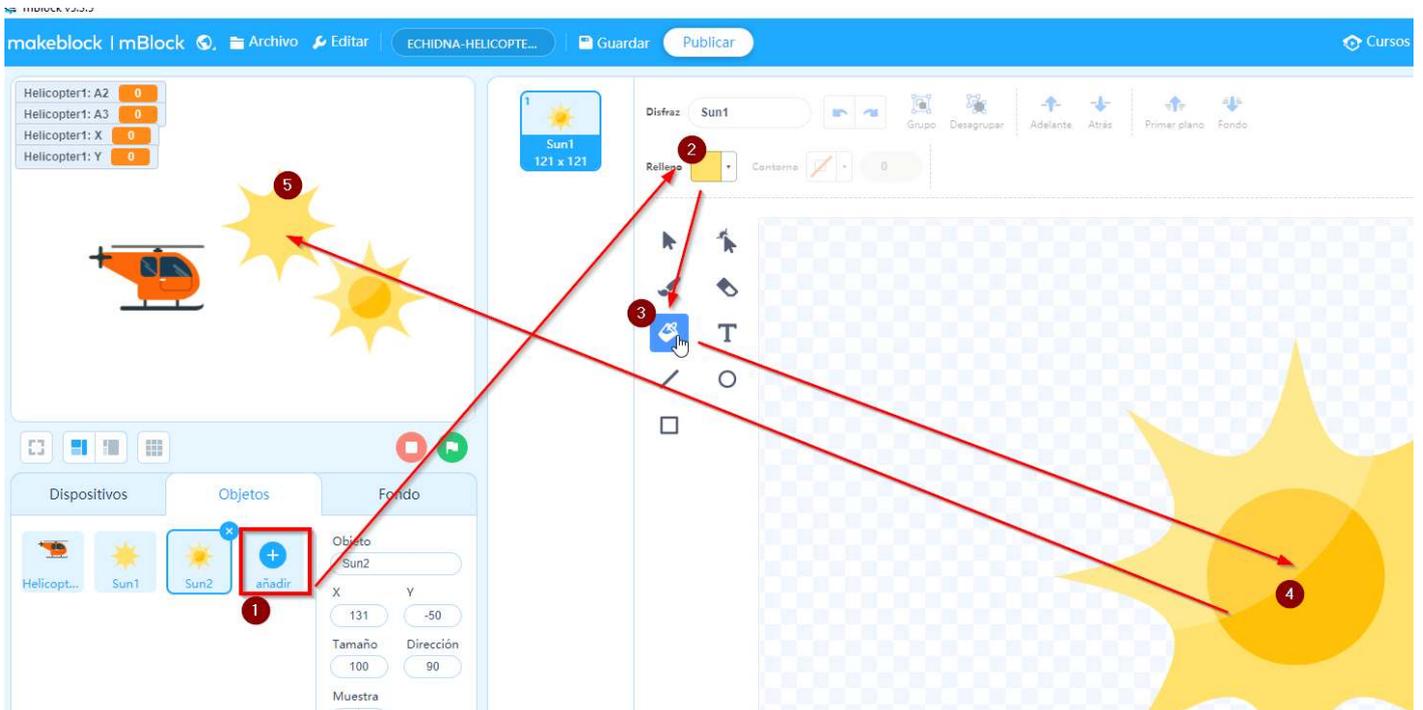
<https://planet.mblock.cc/project/projectshare/3230393>

El resultado es:

<https://www.youtube.com/embed/VERfepEkNv8>

Lo sé soy bastante malo !!





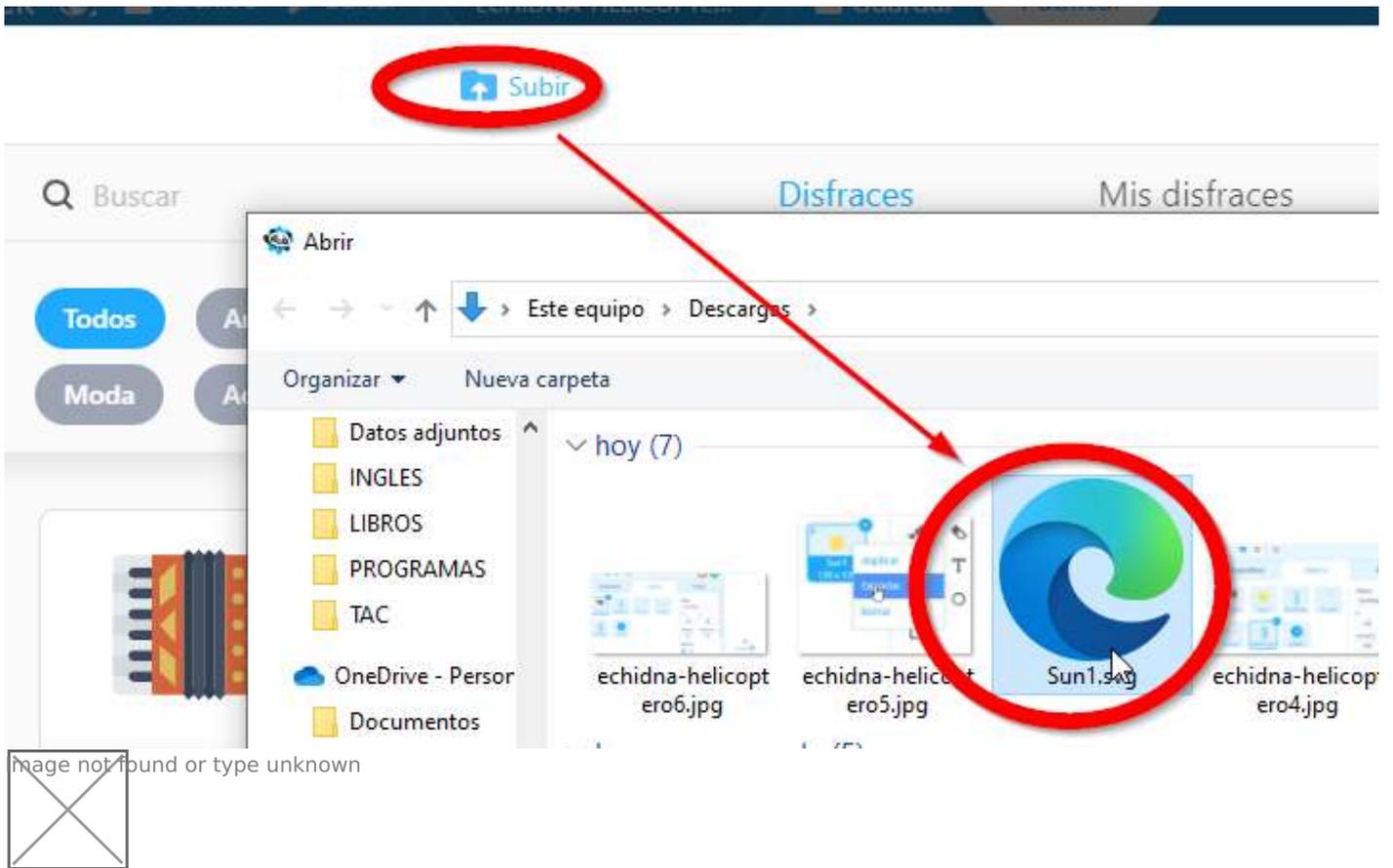
2.- Exporta el fichero imagen svg



3.- Añade al helicóptero el disfraz



4.- Eligiendo el fichero svg exportado en el paso 3



Vamos a hacer que el helicóptero que se mueva según el acelerómetro:

Posición X: La pantalla de mBlock tiene desde -240 hasta 240 y hemos visto que los márgenes de A2 del acelerómetro va desde 250 hasta 500 luego hacemos el cambio de escala con la [HOJA DE CÁLCULO](#)

HOJA DE CALCULO DE CAMBIO DE ESCALA		
$Y = mX + n$		CAMBIA LO AMARILLO
Valores que tengo de x		
X1=	250	
X2=	500	
Valores que quiero de Y		
Y1=	-240	
Y2=	240	
RESULTADOS		
m =	1,92	
n =	-720	

image not found

luego ya tenemos la fórmula para la posición X



E igualmente para la posición Y pero en este caso los márgenes de la pantalla van desde -180 a 180 luego:

HOJA DE CALCULO DE CAMBIO DE ESCAL	
$Y = m X + n$	CAMBIA LO AM
Valores que tengo de x	
X1=	250
X2=	500
Valores que quiero de Y	
Y1=	-180
Y2=	180
RESULTADOS	
m =	1,44
n =	-540

Testeando un poco hemos subido m a 1.9 pues no llegaba arriba del todo

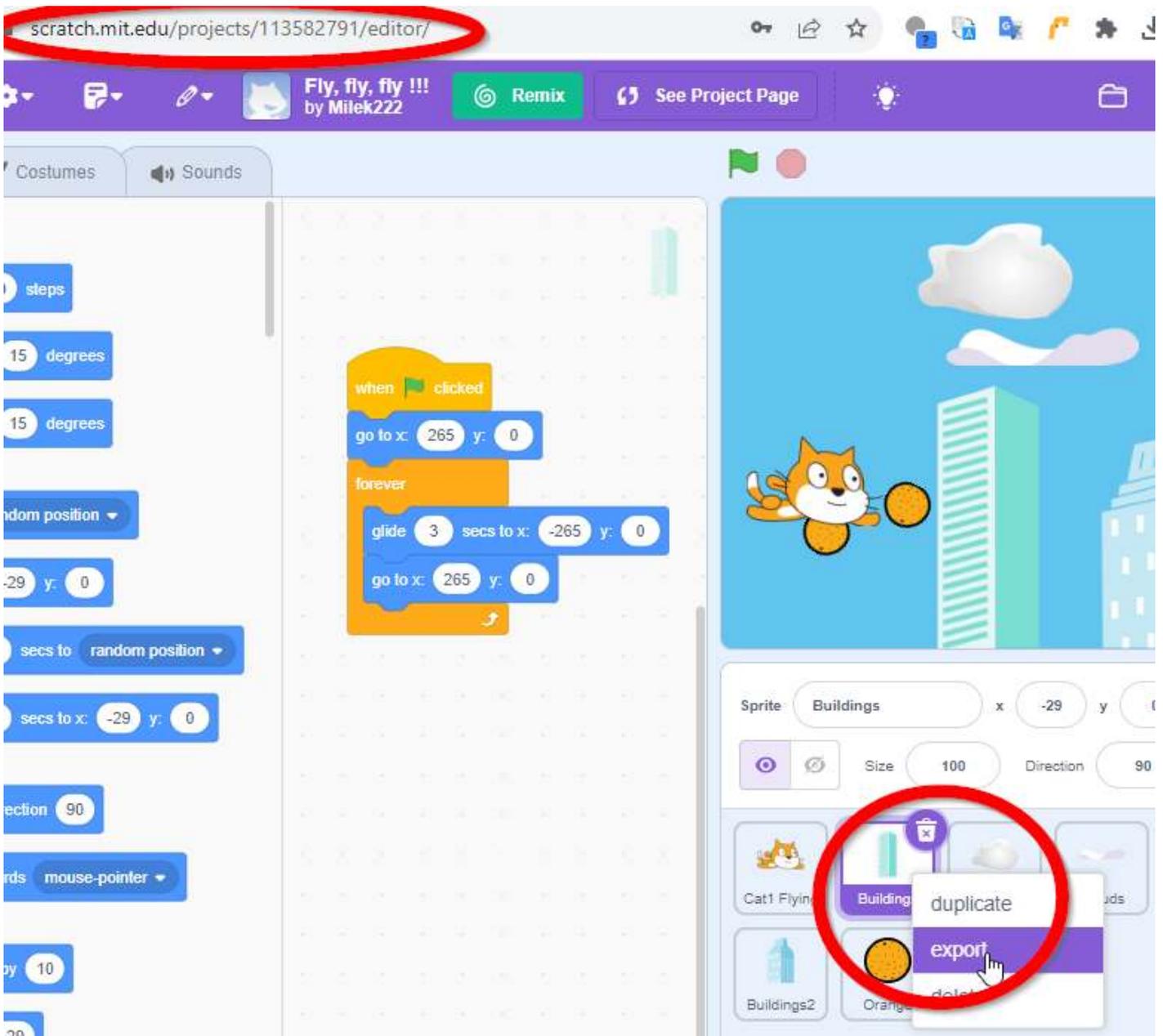


En el arduino le asignamos las coordenadas a las variables globales A2 y A3



Solución edificio, cómo se reutiliza un sprite

Reutilizamos un poco: Nos copiamos un Sprite, entramos en este proyecto FLY de scratch <https://scratch.mit.edu/projects/113582791/> entramos en su código y **exportamos** los edificios, nubes, etc...

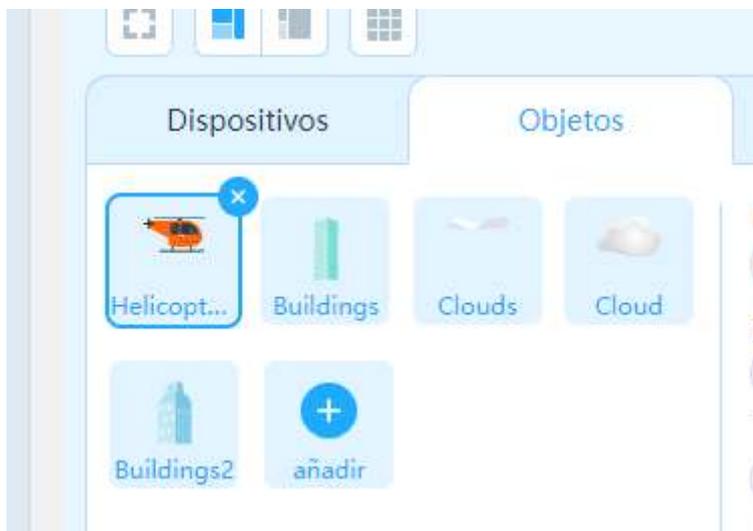


Vamos a mblock e **IMPORTAMOS** el fichero que hemos exportado



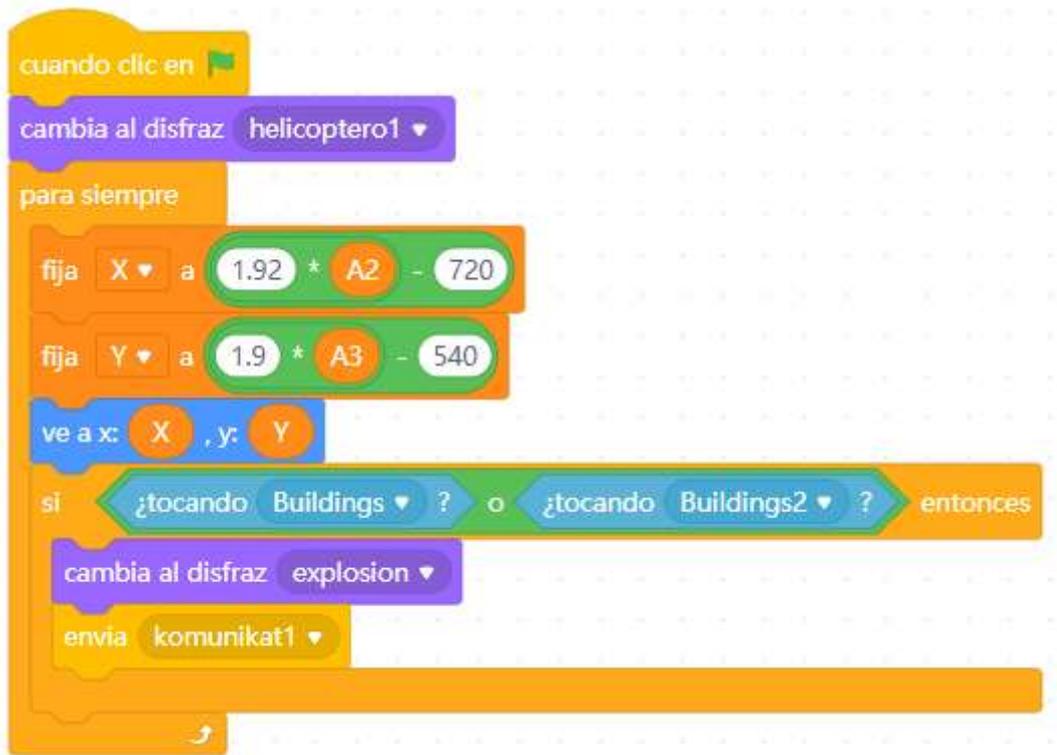
Nos importa con sus disfraces Y SU CÓDIGO

Nos queda pues esta colección de objetos



Solución a la muerte del helicóptero

Vamos a poner que si toca un edificio, “explota” (o sea, cambia el disfraz a explosión), además envía el mensaje “komunikat1” que vemos que el script del edificio lo necesita para esconderse una vez terminado el juego (debe de ser ruso el mensaje es del autor del objeto edificio) :



El programa lo tienes aquí <https://planet.mblock.cc/project/3230430>

El resultado se puede ver en este vídeo:

<https://www.youtube.com/embed/aMY6qiib-Bc>

Mejoras

Ya sé que colocando el helicóptero siempre arriba, siempre ganas!!

RETO: HAZ QUE SOLO SI TOCA BULTING1 MUERE, PERO SI TOCA A BULDING2 QUE GANE UN PUNTO de esta manera ya no se gana poniendo el helicóptero arriba del todo

<https://giphy.com/embed/f79OYWh5uwlFk>

via GIPHY

3.4.2 MONTAJE 10 MATA-ALIENS hackeando código

Esta vez no vamos a ser originales, vamos a ser un poco vagos y nos vamos a copiar el programa de otra persona. La [web de Scratch](#) tiene un repositorio de millones de proyectos, podemos seleccionar uno, y adaptarlo a nuestra simpática Echidna.



RETO Vamos a ver un ejemplo con este proyecto <https://scratch.mit.edu/projects/29744/>. El reto que te lanzamos es que funcione la nave espacial con nuestro acelerómetro de la Echidna.

Solución CON MBLOCK

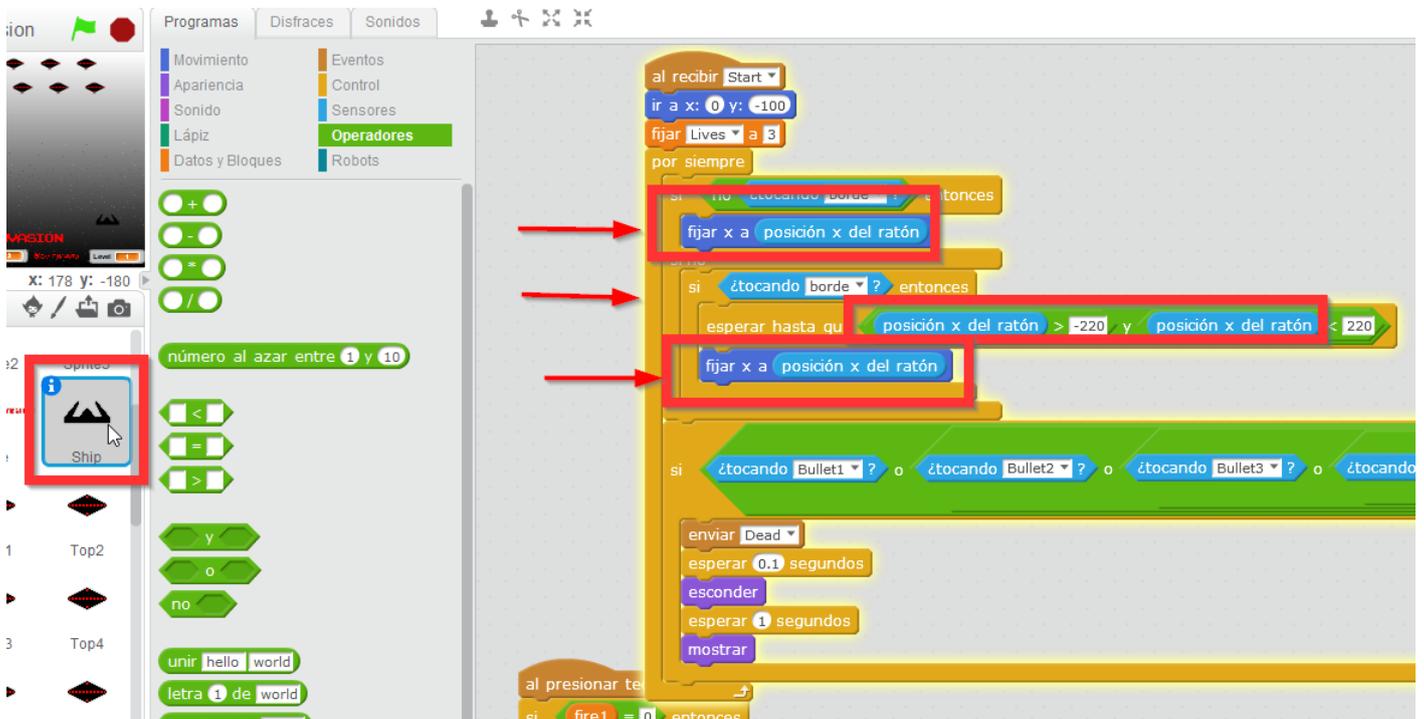
En [esta presentación](#) se muestra cómo se importa un proyecto a mBlock: (las capturas están con mBlock3, para mBlock5 simplemente no es abrir, sino **Importar desde el ordenador**

Curiosamente con mBlock3 sale algún fallo pero con mBlock5 no hay, lo importa perfectamente

https://docs.google.com/presentation/d/e/2PACX-1vSCDGNYTkUuXsIOtMJDzS3jjjvI8ZgYk9cy50vpgvICAUVZOZbLtTjynKjkZ_CHhq9nvJQzIefO9V2/embed?start=falseloop=falsedelayms=3000

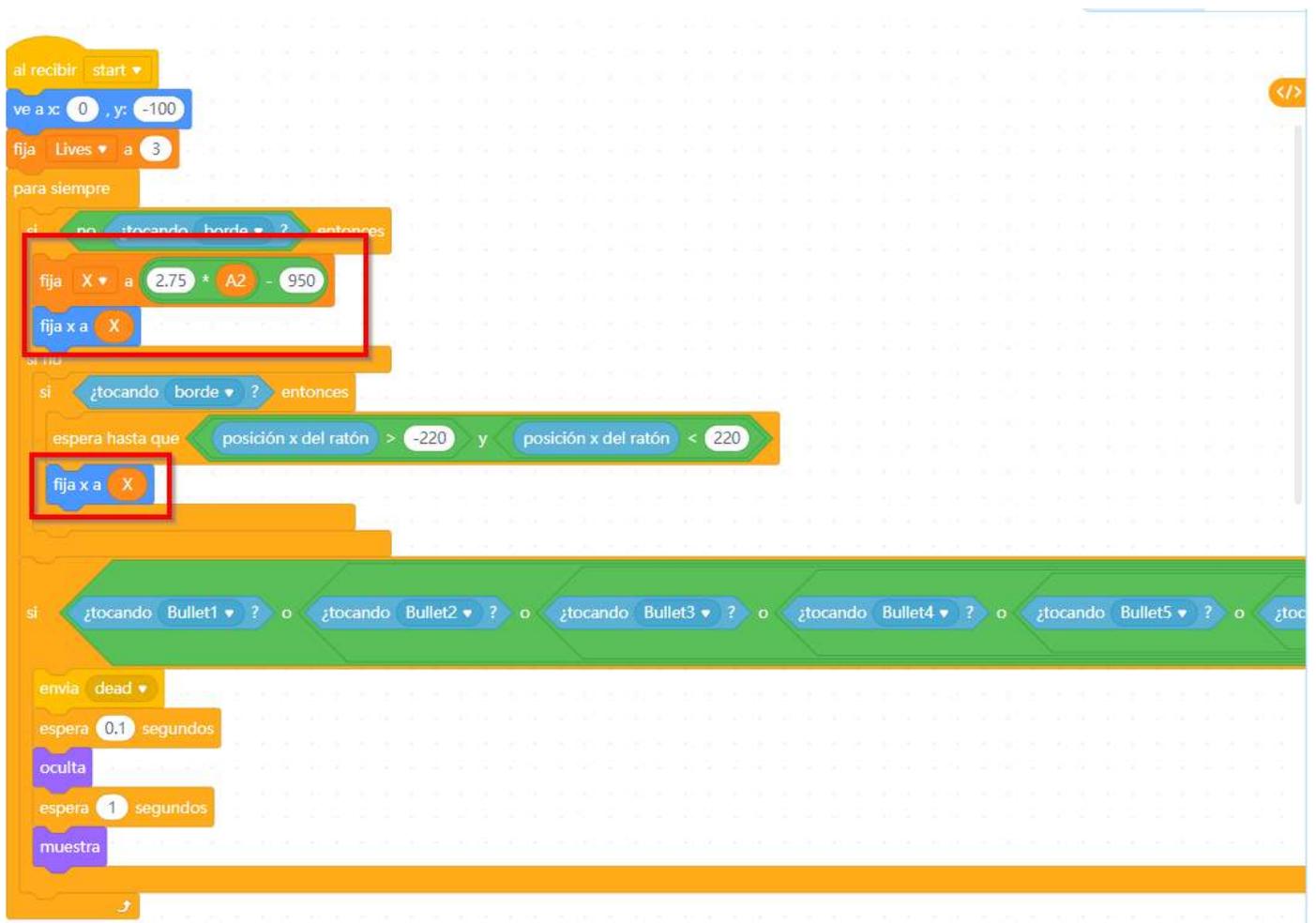
Hackear el código para que funcione con el acelerómetro

Una vez importando nos fijamos en el código que hay que modificar, está claro que lo que hay que tocar es el Sprite "Ship" que es el que queremos mover con el acelerómetro y modificar el "posicionX que se realiza con el ratón y cambiarlo por el acelerómetro:



Perooo como siempre: la posición del ratón es la misma que las coordenadas de la pantalla de mBloc y la x va desde -240 hasta 240 y nuestro acelerómetro de A2 va desde 250 hasta 500 luego hacemos el cambio de escala con la [HOJA DE CÁLCULO](#) anteriormente:

Y cambiamos el código:



En el arduino asignamos A2 (y también D2 luego lo comentamos)

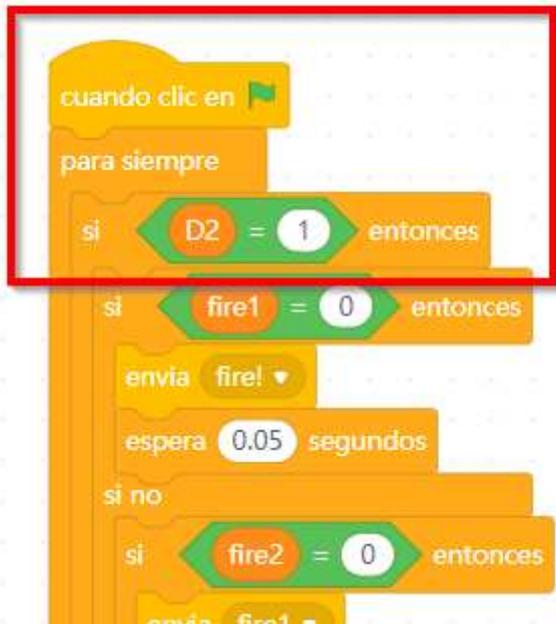


Un poco más, disparar con D2

También vamos a cambiar el disparador “espacio” :



por pin digital 2 para disparar con el botón D2 del Echidna:



El programa lo tienes aquí <https://planet.mblock.cc/project/projectshare/3232624>

El resultado en este vídeo:

<https://www.youtube.com/embed/Ps3jB89Klg8>

SOLUCIÓN CON SNAP4ARDUINO

Importar un proyecto de Scratch a Snap4Arduino

Para importar un proyecto desde Scratch es un poco más complicado, ver [esta presentación](#):

<https://docs.google.com/presentation/d/e/2PACX-1vQ7PutZ1rJdyUGdaQ6qgkoofiQu1-t0inaBN5JN0idYOlvIK84qa1THTBzNUx4SW3ksIXdjUdEs9NF/embed?start=falseloop=falsedelays=3000>

Cambios en el código en Snap4Arduino

Los cambios en el código son los mismos que los anteriores.



show
duplicate
delete
parent...
export...
connect to Arduino
disconnect Arduino

Recuerda que en [esta presentación](#) se enseña cómo configurar Snap4Arduino para que se comunique con nuestro Arduino y la Echidna.

El programa lo puedes abrir en Snap4Arduino en esta [URL](#)

El Sprite que tienes que conectar con Arduino es la nave esa con cuernos.

El resultado lo puedes ver en este vídeo

https://www.youtube.com/embed/ec4q_5bbBQQ