

3. Entradas de Echidna

- [Otra vez acuérdate](#)
- [MONTAJE 4 ENCENDIDOAPAGADO](#)
- [MONTAJE 4-ENCENDIDOAPAGADO-AVANZADO](#)
- [MONTAJE 5 LDR](#)
- [MONTAJE 5 SEMAFORO SONIDO](#)
- [MONTAJE 6 Piano luminoso](#)
- [MONTAJE 7 Telesketch](#)
- [MONTAJE 7 BIS MATA ALIENS con JOYSTICK](#)
- [MONTAJE 7 BIS MATA-ALIENS con acelerómetro](#)

Otra vez acuérdate

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: Acuérdate 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>

MONTAJE 4

ENCENDIDO APAGADO

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.

MONTAJE4-ENCENDIDO APAGADO

Al pulsar el botón D2 se enciende los 3 leds del Echidna y al soltarlo se apagan. El programa también tiene que tener un efecto en un sprite

Solución con EchidnaScratch

Cambiamos el sprite del gato por otro más acorde y cambiamos el disfraz, no lo desarrollamos aquí pues vimos cómo se hace con el MONTAJE1

[2024-12-03 13_31_41-EchidnaML.png](#)

El código :

[2024-12-03 13_38_56-EchidnaML.png](#)

https://www.youtube.com/embed/9pn_a5v6PA4

El fichero lo tienes en el Github : <https://github.com/JavierQuintana/Echidna>

Solución con mBlock

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

[echidna-d2-encendido-apagado1.png](#)

Y al arduino:

[echidna-d2-encendido-apagado2.png](#)

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



MONTAJE 4- ENCENDIDO APAGADO- AVANZADO

MONTAJE4-ENCENDIDO APAGADO-AVANZADO

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 EchidnaShield


En este caso creamos una variable **ENCENDIDO** que registra si están los leds y el sprite encendidos o no

El código es

[2024-12-03 13_56_27-EchidnaML.png](#)

Para evitar que el programa sea engorroso, se ha creado los bloques ENCENDER y APAGAR, pero perfectamente se puede meter estas instrucciones dentro del algoritmo principal y pasar de bloques.

FIJATE  que la variable ENCENDIDO se cambia a 1 o a 0 para que refleje el estado en el que ha quedado el sistema

FIJATE  que hay una espera de 0.5segundos para que en ese tiempo **no lea** si no lo pones, esta todo el rato cambiando los pocos milisegundos que pulsas el botón

Una manera de quitar ese efecto es poner un "Esperar que NO este pulsado" mira:
[2025-11-28 14_38_16-Manual EchidnaBlack y EchidnaML - PDF-XChange Viewer.png](#)

Manual EchidnaBlack² y EchidnaML www.echidna.es CC-BY-SA

[2024-12-03 13_58_18-EchidnaML.png](#)

[2024-12-03 13_58_49-EchidnaML.png](#)

<https://www.youtube.com/embed/ZH-nsYdAazY>

El programa lo tienes en : <https://github.com/JavierQuintana/Echidna>

Solución con mBlock

echidna-d2-encendido-apagado3.png

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 apagaran 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.

MONTAJE 5 LDR

¿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.



[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

2024-12-03 19_55_23-EchidnaML.png

¿Qué vamos a hacer?

Jugar un poco con vuestras condiciones de luz, y definir unos valores de oscuridad, poca luz, media luz y mucha luz, en mi caso mis valores son

- Oscuridad menor 200
- Poca luz entre 200 y 500

- Media luz entre 500 y 800
- Mucha luz mayor 800

Con estos valores vamos a realizar un programa que

- Oscuridad: Ningún led encendido
- Poca luz Luz roja
- Media luz Luz naranja
- Mucha luz Luz verde

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

Según nuestros valores experimentales, (puedes poner otros según las condiciones de luz de tu aula)

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

Solución con EchidnaScratch

2024-12-05 09_43_50-Presentación1 - PowerPoint.png

<https://www.youtube.com/embed/5JJY39A2Fm8>

Lo puedes encontrar en en este repositorio: <https://github.com/JavierQuintana/Echidna>

Reto

Haz un sprite tipo semáforo que también se enciendan las luces igual que el Echidna real, tal y como hicimos en MONTAJE1 SEMAFORO y encima que diga los valores

[echidna-medirluz4.png](#)

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

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

Solución con mBlock

Con mBlock al no tener instrucciones específicas para Echidna, para leer en directo los valores del LDR, hay que utilizar nuestro mBlock y que nuestro simpático oso panda nos diga esos valores, con este sencillo programa en el decide dispositivo **Arduino**:[echidna-medirluz1.png](#)

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

[echidna-medirluz3.png](#)[echidna-medirluz2.png](#)

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>

Programa mBlock

[echidna-medirluz6.png](#)

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



MONTAJE 5 SEMAFORO SONIDO

No vamos a dejar sin poder experimentar este sensor que mide la intensidad del sonido y como está conectado a la salida analógica A7 sus valores van desde el 0 al 1024

Vamos a hacer un semáforo que mida la intensidad del sonido, verde si es poco, y sube hasta rojo cuando es elevado

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

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

El programa en EchidnaScratch es :

[2024-12-08 13_37_26-EchidnaML.png](#)

Hemos simulado el sonido soplando en el micro :

<https://www.youtube.com/embed/bi17ruyNxSg>

¿Te gustaría este semáforo pero que se ejecute **de manera independiente** (sin necesidad de ordenador), para usarlo en el aula, comedor...? Pues claro, cargando el programa, pero no con Echidna Scratch sino con código, ver en el punto 8 semaforo sonido

¿Te atreves a ...?

Realizar un programa que haga lo mismo pero con la temperatura (los valores límites de encendido y apagado de las luces dependen de la temperatura de trabajo)

MONTAJE 6 Piano luminoso

Enunciado

Realizar un programa que suene una nota diferente según la luz

Solución en EchidnaScratch

En echidna no podemos dar las notas numéricas (en mBlock sí) tenemos que utilizar la notación midi a b c d e f g luego simplemente según un nivel de luz que toque una de esas notas

[2024-12-05 12_29_44-Presentación1 - PowerPoint.png](#)

<https://www.youtube.com/embed/O59dMDvtELQ>

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

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

Solución con mBlock5

En el Arduino el programa es

[ECHIDNA-PIANOLUZ1.png](#)

En el objeto, el panda es

[ECHIDNA-PIANOLUZ2.png](#)

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 ATENCIÓN ESTA VERSIÓN ES OBSOLETA NO RECOMENDADO sólo para PCs muy viejos

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](#).

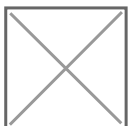
El resultado lo puedes ver en [este vídeo](#):

<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: $72 = m \cdot 982 + n$

¡A! 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:



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

$$Y = 0.027 \cdot X + 45$$

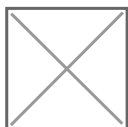


MONTAJE 7 Telesketch

¿Qué es el Joystick?

Son dos potenciómetros, uno en el eje X y otro en el eje Y además de un pulsador digital cuando se pulsa hacia dentro.

El potenciómetro X está conectado en los pines A0 del Arduino y el del eje Y al A1 por lo tanto si has leído los capítulos de ARDUINO **sus valores varía desde 100 hasta 1024**. El pulsador está conectado al D10.



LAS COSAS CLARAS: IGUAL DA ESTA INSTRUCCIÓN

[2025-04-30 19_54_37-EchidnaML.png](#)

QUE ESTA: (poniendo A0)

[2025-11-22 23_41_08-EchidnaML.png](#)

Preferimos la primera pues EchidnaML se adapta a la versión de la placa es decir, que si en el futuro el joystick x en una versión futura es por ejemplo A5 la instrucción primera se adapta y lo envía a la A5 siendo transparente al alumado, mientras que en la segunda tienes que saber que ha cambiado a A5

Reto Telesketch

Si no sabes lo que es esto, no has tenido infancia

TELSKET.jpg

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

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



Solución con EchidnaScratch

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

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

Simplemente vamos a considerar estas condiciones:

- Si el joystick en el **eje X** es menor que **300** es que estás inclinando el Joystick hacia la izquierda, luego el Sprite tiene que modificar su variable **decrementando** su valor, un valor fijo que lo definiremos como PASO
- Si el joystick en el **eje X** es mayor que **700** es seguro que estás inclinando el Joystick hacia la derecha luego hay que **incrementar** el valor de la ordenada X el valor predefinido PASO
- Idem para el **eje Y**
- **PASO** es una variable que de momento lo vamos a poner como 3, si es menor, va más preciso pero más lento, si es mayor es más rápido pero el dibujo es menos preciso.
- Los botones SR y SL nos servirán para subir y bajar el lápiz y borrar

¿Lápiz ? ¿Qué lápiz?

Lápiz es una extensión para poder dibujar. Para instalar una extensión pulsa aquí

[2024-12-05 15_49_35-EchidnaML.png](#)

E instalamos esta extensión

[2024-12-05 15_50_03-EchidnaML.png](#)

Y se instalan nuevas instrucciones para dibujar

[2024-12-05 15_51_36-EchidnaML.png](#)

Código con EchidnaScratch

Recomendamos cambiar el sprite del gato por otro más apropiado, por ejemplo el lápiz y bajarlo a un tamaño 15



2024-12-05 15_54_22-EchidnaML.png

El programa :

- Inicia las variables que hemos explicado
- En el bucle, las condicionales hacen que se incremente o decremente las variables X e Y la cantidad definida PASO
- El botón SR
 - si se pulsa, pinta
 - si se suelta no pinta
- El botón SL borra todo
- Mover el sprite según las coordenadas X e Y

2025-04-30 20_06_40-EchidnaML.png

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

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

https://www.youtube.com/embed/a27fEoP_A2w

Solución con mBlock

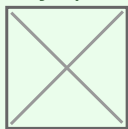
En mBlock no hay instrucciones especiales para Echidna, hay que hablar pues de A0 para el eje X y A1 para el eje Y por lo tanto las nuevas condiciones son:

Si A0 es menor que 300 es que estás inclinando el Joystick hacia la izquierda, luego el Sprite tiene que modificar su variable x hacia la derecha, es decir decrementando su valor



image-1648727378013.png

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

Programa en mBlock

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

[echidna-teleschetch.png](#)

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

[echidna-teleschetch2.png](#)

El programa te lo puedes descargar aqui <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.

MONTAJE 7 BIS MATA ALIENS con JOYSTICK

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.



Vamos a copiar un proyecto ALIENS INVASION nave espacial con nuestro JOYSTICK de la Echidna.

1.- IMPORTANDO PROYECTO EXTERNO SCRATCH A EchidnaScratch o mBlock

Entramos en SCRATCH <https://scratch.mit.edu/> y buscamos ALIEN INVASION, concretamente este : <https://scratch.mit.edu/projects/29744/>

2025-11-22 23_13_30-EchidnaML.png

Entramos en **SEE INSIDE**

2025-07-01 20_39_34-Aliens invasion on Scratch.png

Lo grabamos en nuestro ordenador

2025-07-01 20_40_59-Aliens invasion on Scratch.png

Nos vamos al programa ECHIDNA ML y cargamos el programa descargado

2025-07-01 20_42_13-EchidnaML.png

2.- Hackear el código para que funcione la nave con el joystick



Vamos a fijarnos en el código de la nave, si nos fijamos se mueve según la posición X del ratón, **vamos a cambiar ese código**

[2025-11-22 23_19_05-EchidnaML.png](#)

Creamos dos variables :

- Una que se llame **X** que tendrá la posición X de la nave, la inicializamos a 0 al principio y así se coloca la nave en medio al empezar la partida
- Otra que se llame **PASO** que fija el paso o incremento cuando se mueva, cuanto más alto, más rápido se mueve, pero menos preciso, a golpes, cuanto más bajo, más lento pero más preciso, lo pondremos a 2

Y cambiamos el código de mover la nave :

[2025-11-22 23_27_17-EchidnaML.png](#)

- Si el joystick en el sentido X **baja** de 200 (puede ser un número entre 0-512 pues 512 en teoría es en medio) pues que X se **decremente** un PASO
- Si el joystick en el sentido X **sube** de 600 (puede ser un número entre 512-1021 pues 1024 es el máximo) pues que X se **incremente** un PASO

3.- QUE DISPARE APRETANDO EL BOTÓN SL

Si nos fijamos, dispara si pulsamos la tecla **espacio**, lo quitamos y ponemos todo ese código dentro del código nuevo que se ejecuta si apretamos el botón SL

[2025-11-22 23_31_09-EchidnaML.png](#)

Queda pues

[2025-11-22 23_33_12-EchidnaML.png](#)

El resultado

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

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

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



MONTAJE 7 BIS MATA-ALIENS con acelerómetro

¿Se puede hacer el anterior ejercicio con el acelerómetro? Por supuesto que sí, la única diferencia es que el Joystick va de 0 a 1024 pero el acelerómetro tiene otros valores como puedes ver en esta página <https://echidna.es/hardware/componentes/acelerometro-black/>

	VALOR MÍNIMO	VALOR REPOSO	VALOR MÁXIMO
JOYSTICK	0	512	1.024
ACELERÓMETRO	285	359	428

más o menos, pues cada placa es diferente

Puedes probar tus valores con este sencillo programa

[2025-11-23 08_50_15-EchidnaML.png](#)

Luego definimos los valores 300 y 400 arbitrarios pero menores y mayores que 359 pero sin llegar a los extremos 285 y 428 para que nuestro sprite nave se mueva :

[2025-11-23 08_53_18-EchidnaML.png](#)