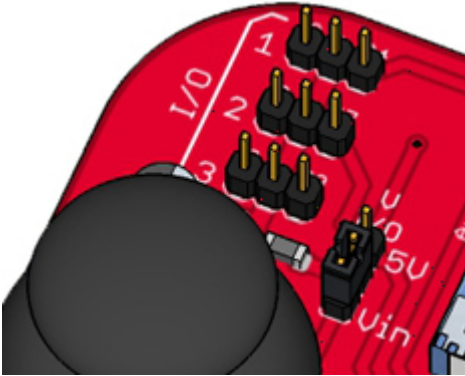


5. Extensiones

- [Extensiones de Echidna](#)
- [MONTAJE 13 Encender con el móvil \(muy difícil\)](#)
- [MONTAJE 14 Encendido sensible](#)
- [MONTAJE 15 Alarma láser](#)
- [5.4 MONTAJE 16 Semáforo distancia](#)
- [5.5 Servo](#)
- [5.6 MONTAJE 17 Tractor entrando al corral](#)

Extensiones de Echidna

Ahora vamos a utilizar las extensiones 3 digitales y una analógica:

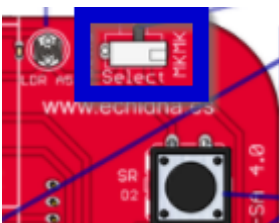


Para conexión de otros elementos como relés pero no pueden pasar de 300mA de lo contrario dañan al Arduino.

- **Entrada analógica A5**
- **Salidas/Entradas digitales D4, D7 y D8**

La alimentación de estas I/O se puede elegir entre la salida de 5V por el Arduino y Vin con la tensión que estemos alimentando al Arduino utilizando el jumper que ya vimos [en su momento](#).

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>

Ahora con las extensiones puedes aumentar más tus proyectos robóticos. Por ejemplo en este vídeo el Echidna se ha colocado en un chasis con unos motores y controlado por el Bluetooth del móvil. Este ejemplo del robot-coche no lo trataremos aquí pues es de un nivel superior, que se trata mejor en [Arduino con código](#)

<https://www.youtube.com/embed/CXOK8tyYp3g>

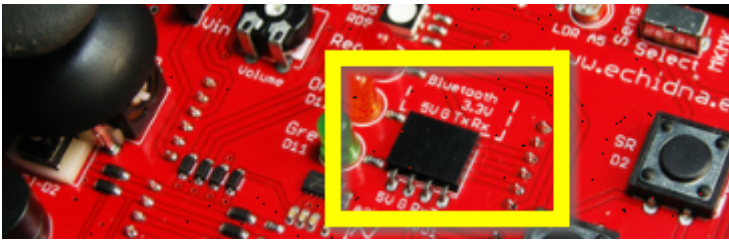
Pero sólo está el límite de tu imaginación para convertir al Echidna el algo más poderoso ...

MONTAJE 13 Encender con el móvil (muy difícil)

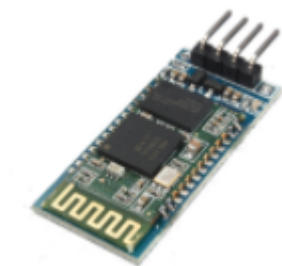
5.1 MONTAJE 13 ENCENDER CON EL MÓVIL Bluetooth

5.1.1 Módulo HC-06

Echidna tiene un conector preparado para conectar un módulo de Bluetooth



Nosotros utilizaremos un JY-MCU o [HC-06](#) muy común y barato. .



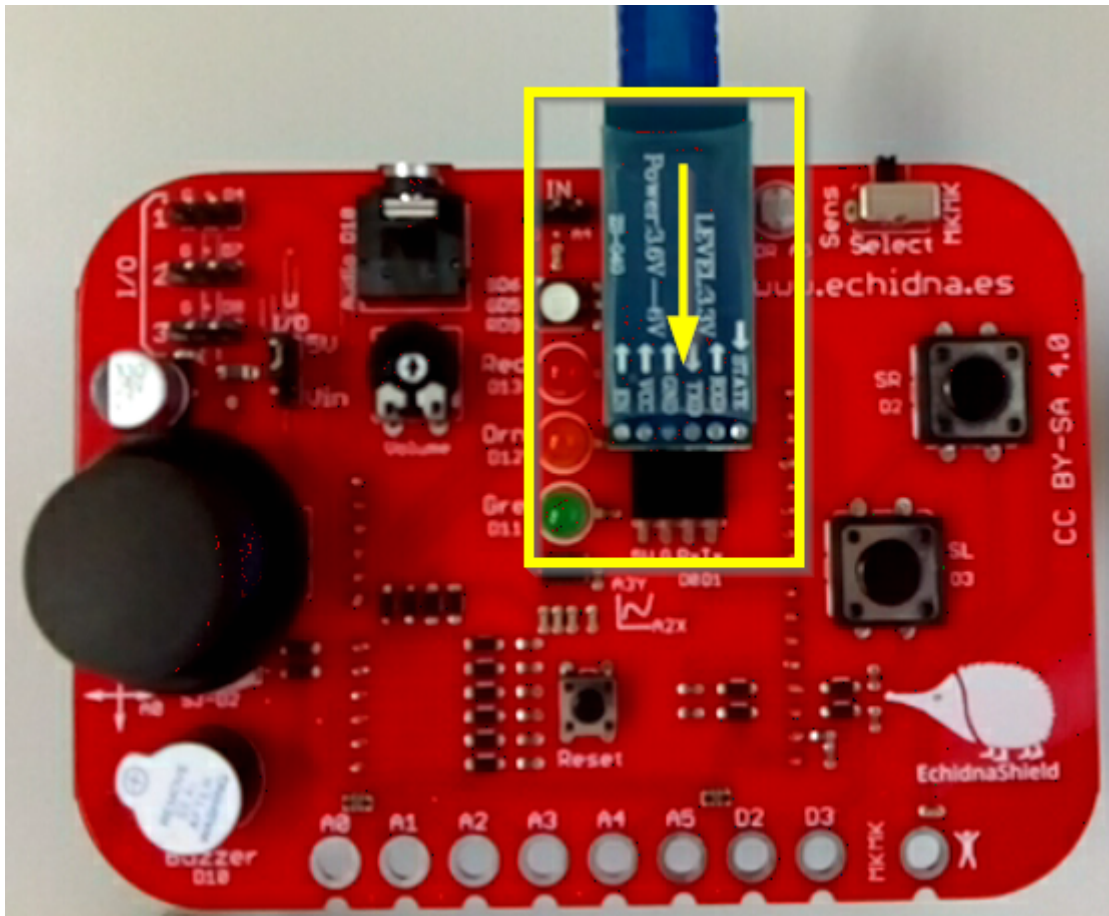
Te recomendamos estas páginas:

- [Teoría de Bluetooth](#)
- [Cómo se comunica con un Arduino](#)

Para conectar el HC-06 lo hacemos hacia abajo de modo que coincida los pines:

Pines del HC-06	Pines del Echidna	Pines del Arduino
-----------------	-------------------	-------------------

Vcc	5V	5V
GND	GND	GND
RX	TX	D1
TX	RX	D0



Nosotros vamos a utilizar la APP BlueControl:

- [ver cómo funciona](#)
- [ver cómo se vincula con el móvil](#)



5.1.2 Problema número 1: ocupamos el puerto serie

Si has leído [Cómo se comunica con un Arduino](#) habrás visto que ocupamos LOS MISMOS PINES D0 Y D1 QUE UTILIZA EL ARDUINO PARA COMUNICARSE POR EL PUERTO SERIE CON EL ORDENADOR.

Esto crea un problema: No se puede tener conectado el HC-06 mientras nos comunicamos el ordenador con el Arduino.

Solución: **Pues quítalo**, y luego cuando acabes de descargar el programa en el Arduino, **pues lo pones**.

Bah!! ¿sólo era eso? pues no, que nos ocupe el puerto serie nos fastidia: ¿puedes interaccionar con el Sprite? por ejemplo ¿puedes hacer que el oso panda de mBlock se mueva según el mando de BlueControl?...**NO**

5.1.3 Problema 2 la velocidad del puerto es elevada

Al programar con mBlock fija la velocidad del puerto serie a 115200 baudios, y nuestro HC-06 soporta 9600

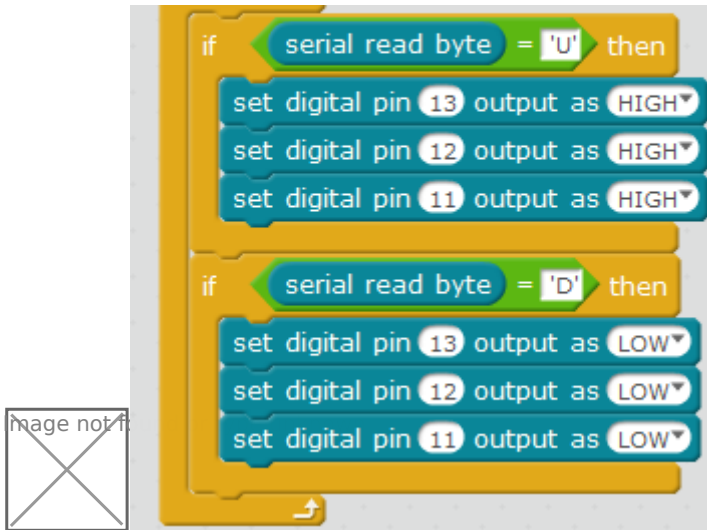
Solución: Bajarlo manualmente, **un rollo**, tenemos que salir de mBlock, editarlo en Arduino IDE esto se ve mejor en el ejemplo siguiente.

5.1.4 Reto: Encendido y apagado de LEDs con el móvil

Vamos a ejecutar este pequeño programa, que al apretar el botón de arriba se encienden los leds y al apretar el de abajo se apagan:

<https://www.youtube.com/embed/XFPGEuX7uTs>

Solución



Primero hay que vincular el móvil con la APP

Segundo hay que subir el programa solucionando los problemas anteriores

Mejor verlo con [esta presentación](#):

O sea, ya podemos jugar con el móvil y con nuestro Echidna !!!

Por ejemplo.. se podría hacer un coche teledirigido, el programa lo tienes en el repositorio:

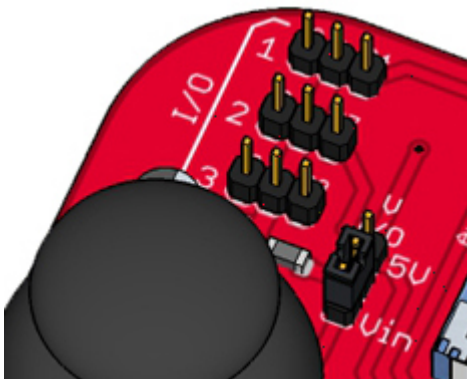
<https://github.com/JavierQuintana/Echidna> y el vídeo en el [muro](#)

MONTAJE 14 Encendido sensible

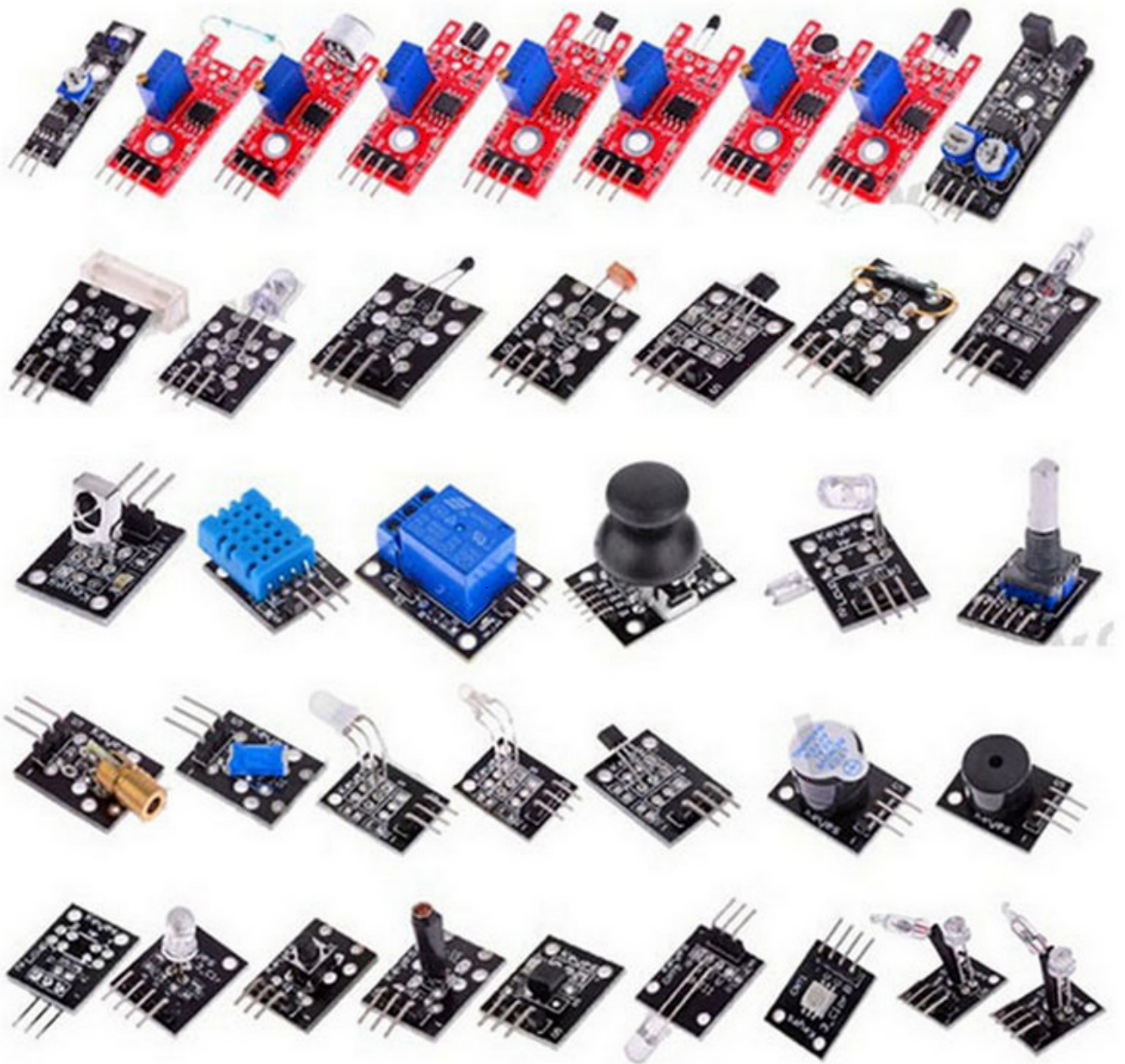
Sensor infrarojos

5.2.1 MONTAJE 14 Encendido Sensible

Vamos a provechar las conexiones digitales que tiene Echidna preparado para conectar módulos exteriores, si te fijas está preparado para conectar fácilmente estos módulos pues facilita la alimentación, GND y el pin digital



Hay muchos módulos para conectar, y todos tienen la misma configuración en sus pines : GND, + , I/O, donde I/O es el pin digital o analógico de entrada o salida, dependiendo del sensor, y como puedes ver en la figura, hay mucha variedad (busca en Internet sensores para Arduino)

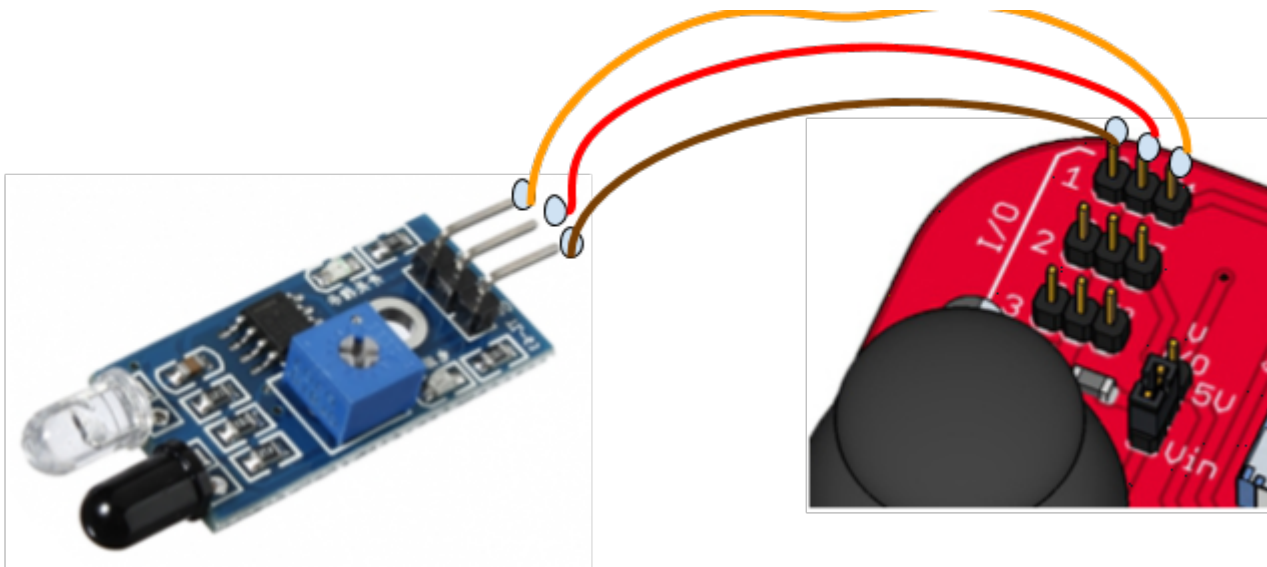


hay para a empezar con uno sencillo de entrada digital que es muy útil: el sensor Infrarrojos



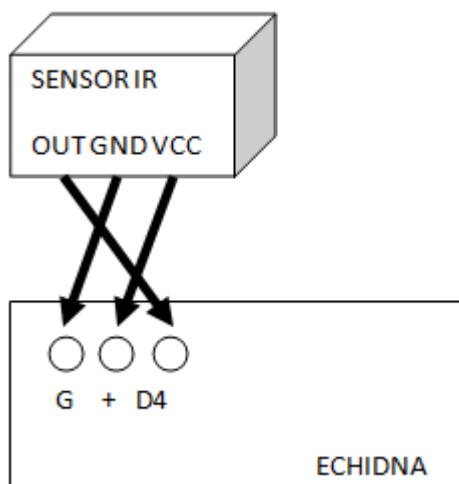
Para ver más información de este sensor te recomendamos [esta página](#).

En el Echidna arriba a la izquierda tienes 3 conectores digitales a elegir, nosotros elegimos el primero D4 luego el pin OUT del sensor se conecta al D4, el resto en el mismo orden



Nota: El potenciómetro es para ajustar la sensibilidad

OJO Hay sensores que tiene los pines en otro orden FIJATE de lo contrario el sensor se pondrá a arder



Nota: Hay sensores con 4 pines, que permiten alimentar el diodo IR de forma independiente, pero tienen un jumper que inutiliza el 4 para sólo utilizar los 3 pines.

Realizamos un pequeño programa muy fácil que detecta si hay un obstáculo o no:

En el Arduino



En el panda



Y el disfraz costume2 lo hemos tintado de rojo

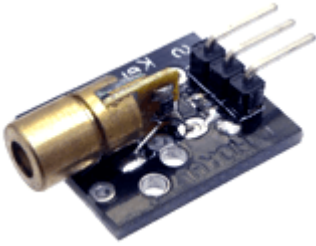
El programa lo tienes aquí <https://planet.mblock.cc/project/3232841>

El resultado es:

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

MONTAJE 15 Alarma láser

Hemos visto anteriormente una entrada digital sencilla con el sensor IR, ahora vamos a ver una salida digital muy simple, buscando siempre la motivación en nuestros proyectos: El diodo Láser.



Si quieres saber más de este componente, te recomendamos la página de [Luis Llamas](#). Si tienes que comprar uno, te recomendamos que no sea superior a 5mW, pues puede dañar permanentemente la retina del ojo [[+info](#)]. El modelo que te proponemos es de **1mW**, no obstante, **EVITA QUE EL LÁSER APUNTE A LOS OJOS** especialmente con niños.

RETO Te proponemos que realices un programa para desactivar una **ALARMA LASER**.

- La alarma está protegida por un haz laser (que enfoca a nuestro querido [LDR del Echidna](#)), si se corta el haz la alarma suena.
- El disparo va a ser simulada en el ordenador, no queremos que alertar a los vecinos, un sonido en el ordenador y en la pantalla una imagen en el ordenador.
- Para desactivar la alarma, pulsamos la tecla espacio, y nos pide introducir el código (tienes 10 segundos para darle más entusiasmo, sino se dispara), si has acertado, el laser se apaga y ya puedes entrar a tu lindo hogar.

Luego vamos a no ponernos nerviosos para desactivar la alarma y mantener la serenidad.

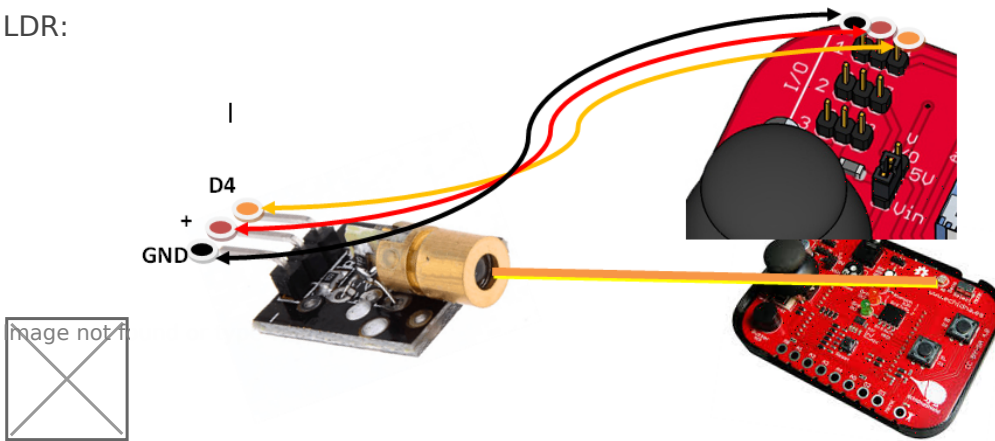
<https://giphy.com/embed/29SqSyXlyO6WI>

[via GIPHY](#)

Solución

Conexiones son sencillas, el laser al D4 por ejemplo y hay que tener maña para que apunte al

LDR:

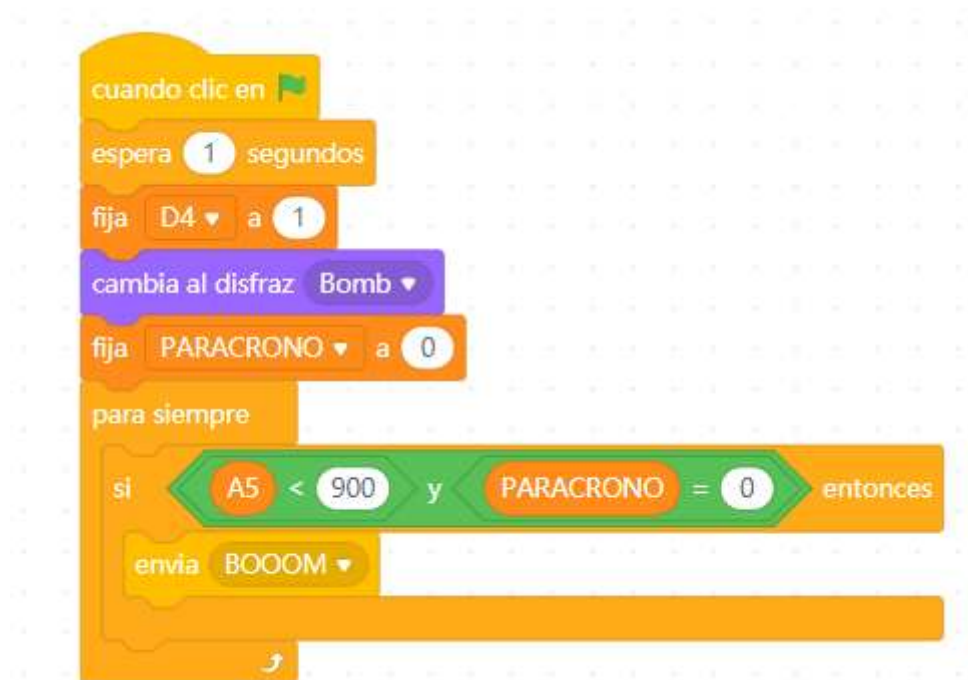


El script que controla si se corta el haz laser que se dispare es un condicional que lee el LDR en A5, mientras sea mayor que 900 es que tiene una intensidad de luz muy grande, es decir, le está enfocando el laser, la variable `_PARACRONO_` es en el caso de que se desactive la alarma, el laser se apagará luego que no se crea que se corta el haz. Si se corta el haz envía un mensaje `_Booom_` que lo lean los demás scripts. Se aprovecha este script para ENCENDER EL LASER pin digital 4 alto:

En el Arduino



En el objeto (que hemos seleccionado una bomba y le hemos puesto un disfraz explosión tal y como se ha explicado antes, utilizando el objeto sun)



El script que controla la desactivación de la alarma mediante código es el siguiente. Si se acierta con el código secreto 666, se envía el mensaje _Ufff_ al resto de scripts:



El script que desactiva la alarma es cuando reciba el mensaje _Ufff_ por lo tanto para el cronómetro y apaga el laser:



El script del tiempo cuenta de forma descendente, y si finaliza envía el mensaje Boom al resto de scripts, sólo se para si PARACRONO=1



Y por último la explosión



El resto: esconder y mostrar scripts y disfraces se omiten por simplicidad, el programa lo puedes descargar aquí : <https://libros.catedu.es/books/arduino-con-echidna-y-mblock-scratch/page/53-montaje-15-alarma-laser>

El resultado es

<https://www.youtube.com/embed/i282JU35m2k>

OTRO RETO

Si se corta el haz, hay 10 segundos para desactivarlo con código, sino, se dispara. No ponemos la solución ¡¡ponlo tú en [el muro](#) !!

Seguro que se te ocurren muchas ideas...

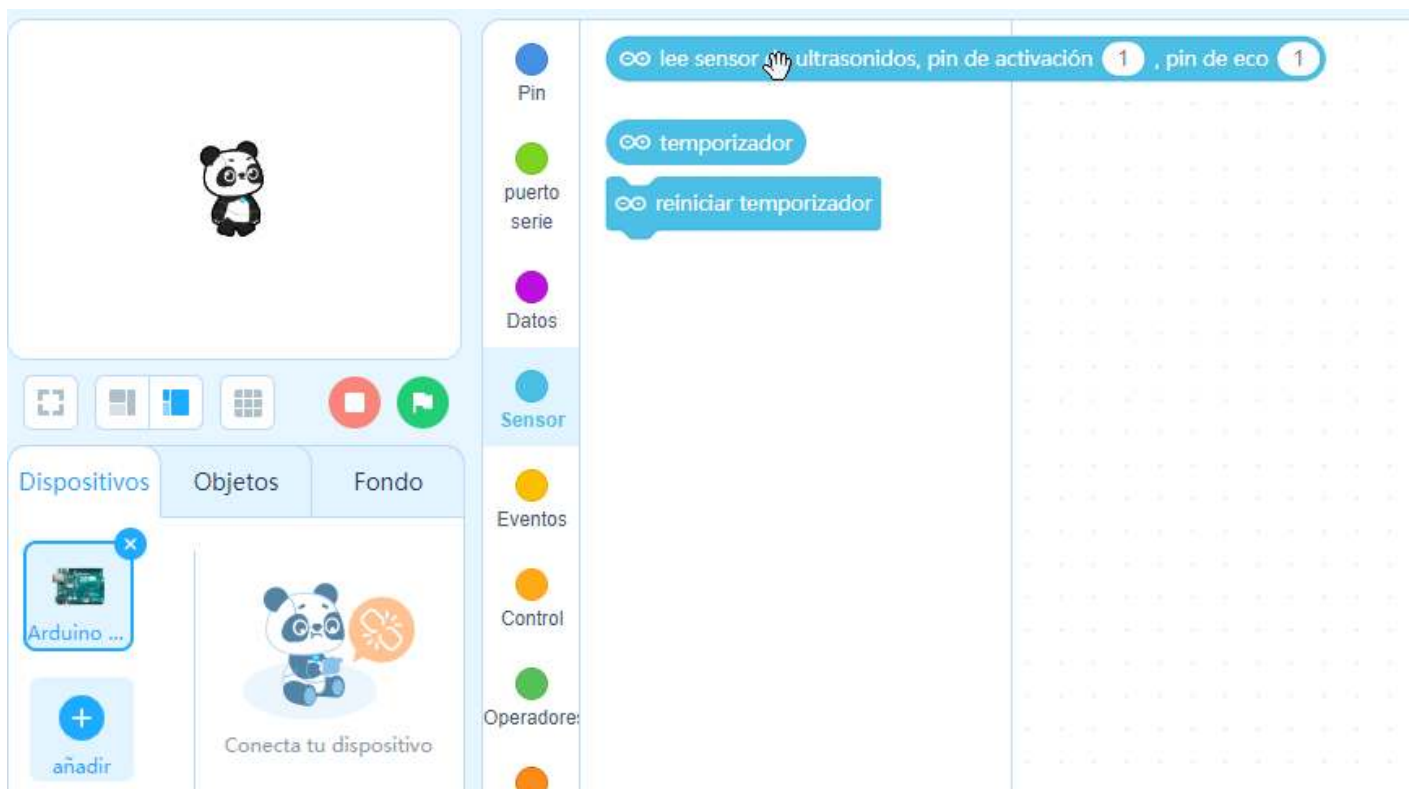


5.4 MONTAJE 16 Semáforo distancia

Queremos poner un ejemplo de un sensor que tenga 4 pines, barato y que puede darnos mucho juego pues nos da información de la distancia en la que se encuentra un objeto.



Funciona por eco entre la señal que se emite por Trg y la que se recibe por Echo y para su utilización requiere utilizar la fórmula de conversión de tiempo a espacio con la fórmula de la velocidad del sonido... tranqui !! no lo vamos a hacer, pues ya mBlock tiene una función especial para ello sin utilizar fórmulas y **nos da directamente la distancia en cm**, pero si quieres saber más sobre este sensor, te recomendamos la página de [Luis Llamas](#).



RETO LEER EL VALOR DEL SENSOR ULTRASONIDOS

Queremos que el objeto panda nos diga la distancia en cm

SOLUCIÓN

Hay un problema

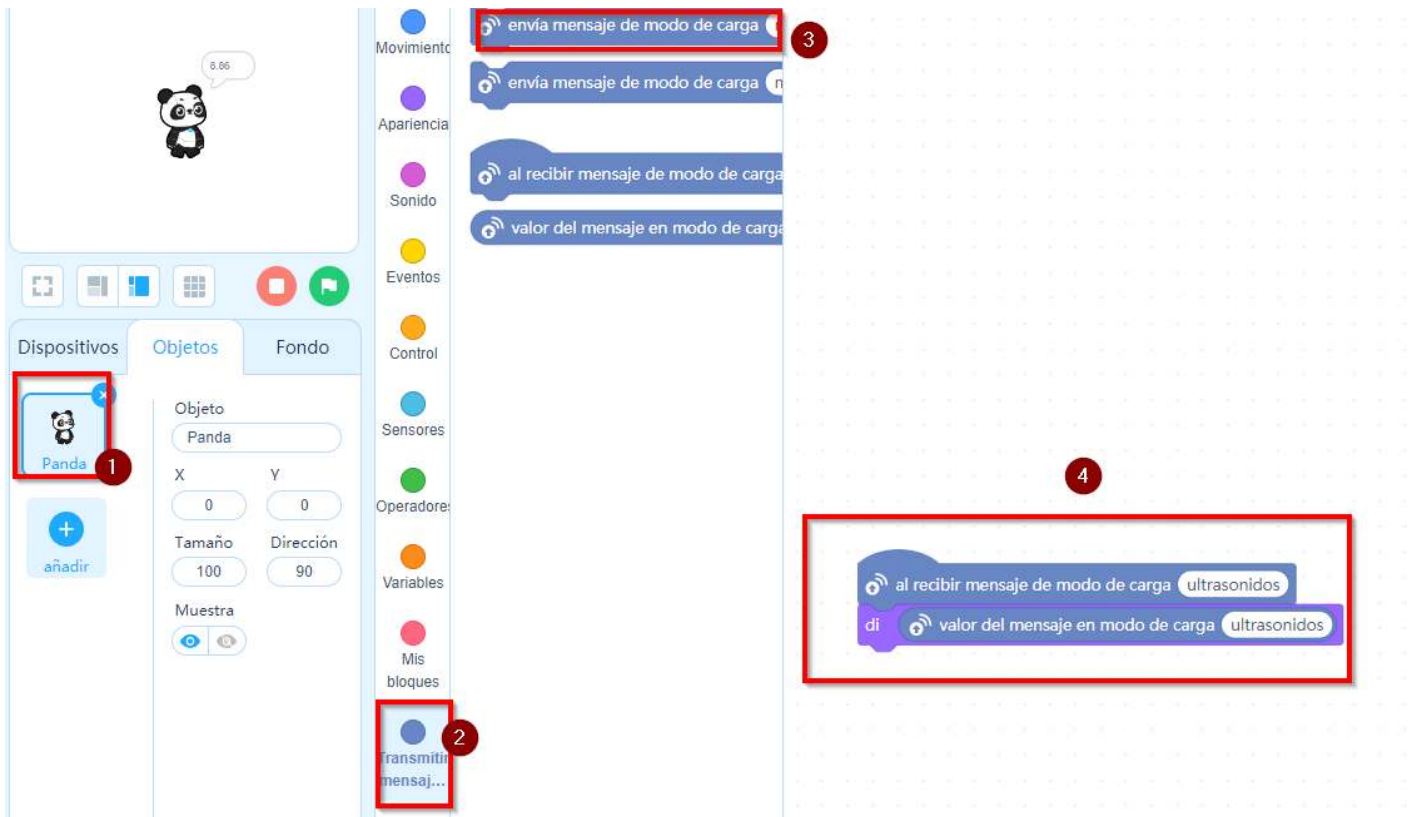
En mBlock3 se puede utilizar en vivo. En mBlock5 **no se puede UTILIZAR EN VIVO LA INSTRUCCION LEE SENSOR ULTRASONIDOS** no sabemos por qué. Sólo podemos cargar el programa en el Arduino

Gracias a la contestación [de este foro](#), podemos hacerlo instalando una extensión



Hay que instalarlo **en los dos, en el Arduino y en el objeto Panda**

En el objeto panda pondremos este código

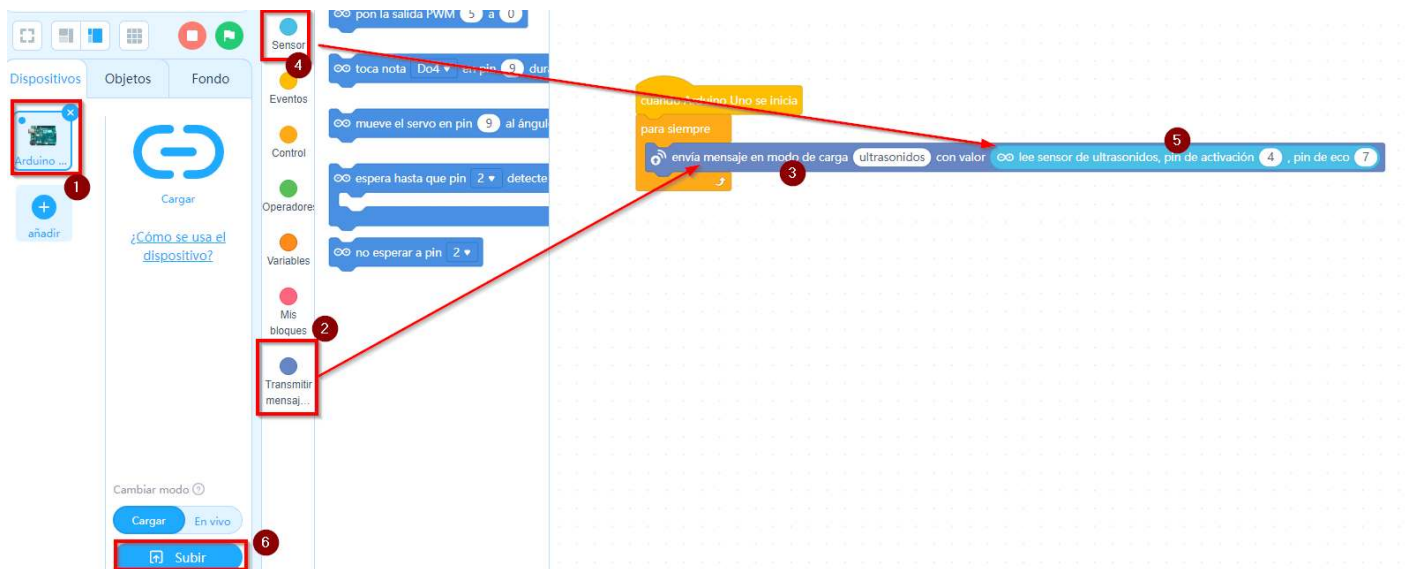


CONEXIOONES Como se necesitan 4 pines, y las extensiones tienen 3 utilizaremos alguno libre.

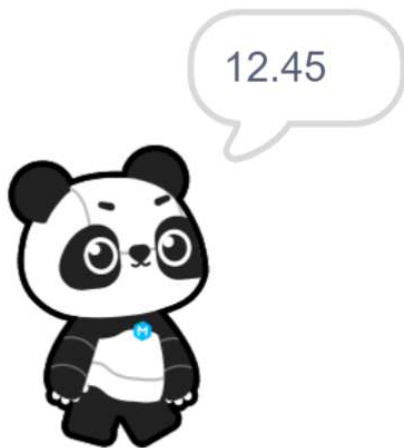
La conexión que vamos a realizar entre los pines I/O de Echidna y el sensor ultrasonidos HC-SR04 va a ser:

el D4 en Trig
 el D7 en Echo
 el '+' en Vcc
 el G en GND

y en el Arduino el siguiente programa, y pulsamos SUBIR



Resultado: El panda nos dice en cm el obstáculo que ponemos delante del sensor US:



El programa lo tienes aquí : <https://planet.mblock.cc/project/3233976>

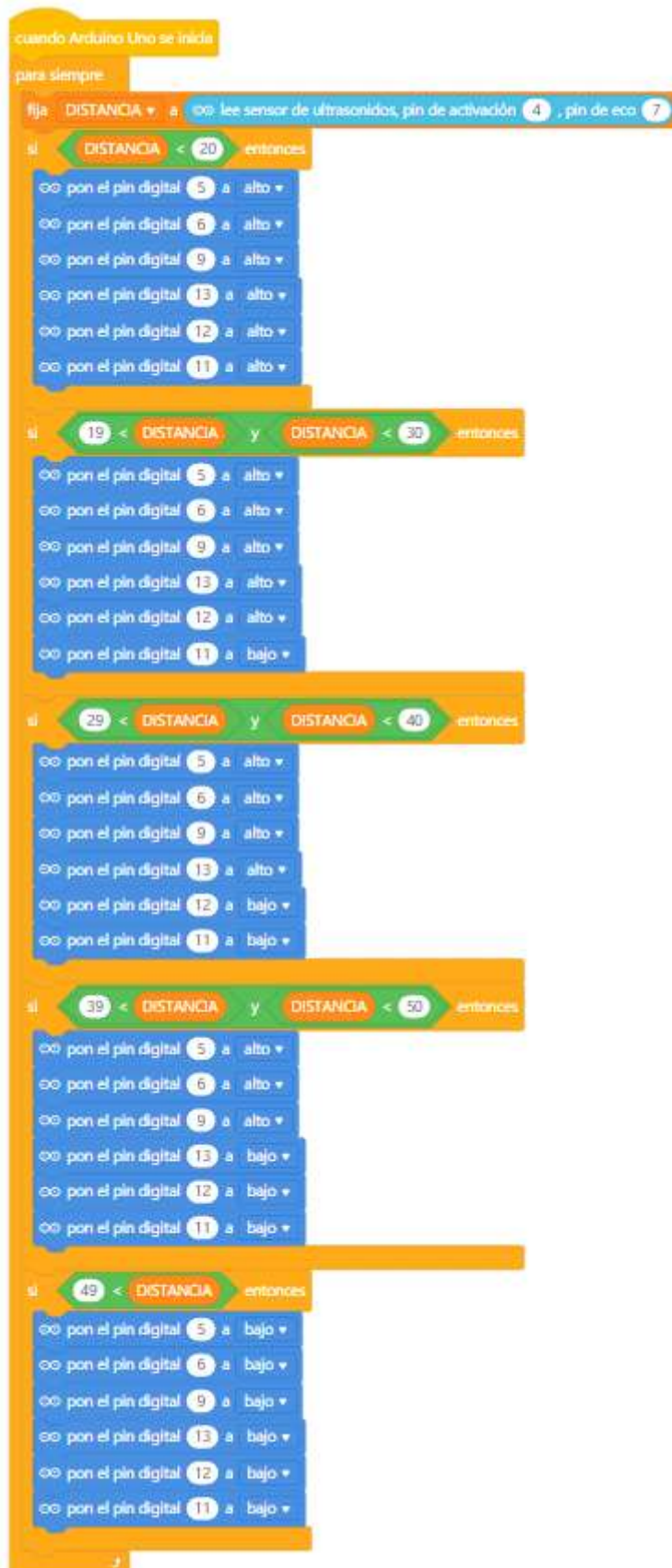
RETO RADAR LUMINOSO

Realizar un programa que a medida que se acerque un objeto, se enciendan más luces

Solución

El programa es simplemente recoger la distancia con la instrucción que hemos señalado antes. La conexión igual que antes y por lo tanto el programa es poner de límites 50cm, 40cm, 30cm, 20cm y 10cm para ir encendiendo luces.

El programa es largo



El programa lo tienes aquí <https://planet.mblock.cc/project/3233931>

El resultado es :

https://www.youtube.com/embed/7s1LDSDaA_A

Los siguientes retos, aunque las imágenes se ve que no están hechos con Echidna, da igual, es simplemente conectar el trig y echo en los pines D4 y D7 del Echidna y utilizar la instrucción de "lee el sensor ultrasónico trig pin 4 echo pin 7" (o utilizar otro orden o el D8 si te pones revelde y cambiar los números anteriores) **¿Te atreves a hacerlos todos ?**

<https://giphy.com/embed/aCrRttmzK1jKo>

[via GIPHY](#)

RETO HINCAR UNA PELOTA Pon de sprite una pelota y que se hinche a medida que acercas un objeto al ultrasonidos. [Solución](#)

RETO PIANO INVISIBLE

Que suene una nota según la distancia del objeto. [Solución](#)

RETO RADAR CON INTERMITENCIA DE UN LED

Cuanto más cerca está un objeto, más rápido un led se enciende y apaga. [Solución](#)

RETO SENSOR PARKING

Cuanto más cerca está un objeto, más rápido suena un pitido intermitente [Solución](#)

5.5 Servo

El servo es un motor que podemos controlar el ángulo de giro, hay diferentes clases, pero nos vamos a centrar en este que tiene un precio/calidad aceptable, **el MG90S**.



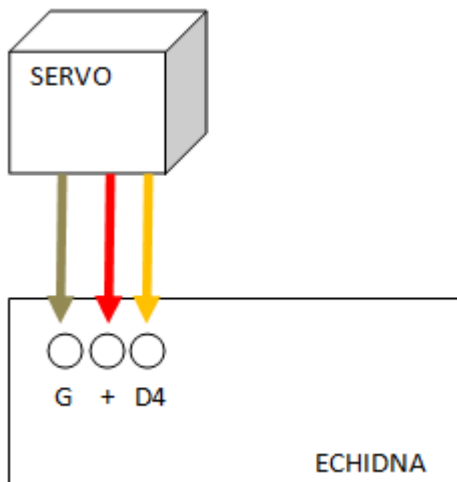
Este servo se controla con una salida digital, y se les indica el ángulo de giro, que puede ir desde 0º hasta 180º (no permite otro margen). Si elegimos otro más barato, puede tener deriva en los extremos ([ver vídeo con HD-144A](#))

Aprende más sobre servos en esta página de [Luis Llamas](#).

La instrucción para los servos en mBlock es muy fácil, simplemente le indicamos en qué pin digital está conectado y el ángulo que deseamos:



y ponlo bien, el marrón indica la masa:



Si intentamos trabajar con el servo con ángulos mayores de 180° no hace caso, va al valor 180°. Si ejecutas este programa lo verás:



<https://www.youtube.com/embed/xZI9bXtVzto>

Mira la diferencia con un **servo de rotación continua**, fíjate como:

- Los extremos 0º y 180º es a máxima velocidad, pero un sentido u otro.
- 90º es parado. - Un valor intermedio es menos velocidad (se ve el ejemplo 80º y 100º)
- Si tiene deriva, (cosa frecuente) hay un potenciómetro para ajustar.

<https://www.youtube.com/embed/Z-5SerXmRY0>

5.6 MONTAJE 17 Tractor entrando al corral

Te proponemos un reto utilizando dos elementos vistos en esta unidad, para dar un poco de rienda suelta a tu imaginación de la cantidad de proyectos que se pueden hacer.

Reto

Construir un proyecto donde la barrera (hecha de cartón y fijada al servo) está bajada y el semáforo en rojo. Si el sensor IR detecta un vehículo, tiene que abrir la barrera durante 5 segundos y el semáforo en verde. Antes de cerrar, se encenderá el semáforo en naranja para advertir que se va a cerrar la barrera.

<https://www.youtube.com/embed/5HEZ3kjr9hY>

Solución

Conectamos por ejemplo el servo al D4 y el sensor de IR al D7
Con mBlock 3



Con mBlock5 es igual



Lo tienes aquí <https://planet.mblock.cc/project/3234387>

Otro reto más difícil

Este ya es para los "sobresalientes":

Podemos hacer que la barrera se abra con Bluetooth, o que también baje si el sensor de Ultrasonidos en el otro lado detecta que el vehículo ya ha pasado. Aunque el vídeo está hecho con la Shield Edubásica perfectamente se puede hacer con Echidna, y en nuestro caso podemos sustituir un sensor ultrasonidos con un sensor IR.

https://www.youtube.com/embed/nlnxai_u360