

# MANUAL PARA ROBOT MÓVIL

## NEZHA V2

### LOG N8115M



C/ Andrés Obispo, 37 - 5º planta  
28043 Madrid  
Tlf: 91 759 59 10

[www.microlog.es](http://www.microlog.es)  
[pedidos@microlog.es](mailto:pedidos@microlog.es)



XUGA 2024  
CÓDIGO ARTIGO: 203059  
NOME: LOTE 12 KIT ELECTRÓNICA AVANZADA  
ANO ADXUDICACIÓN: 2024  
Nº EXPEDIENTE: ED-01/24-SU  
EMPRESA PROVEDORA: MICROLOG TECNOLOGÍA Y SISTEMAS, S.L.  
TELÉFONO DE ATENCIÓN AO USUARIO OU MANTEMENTO: 91 759 59 10



C/ Andrés Obispo, 37 - 5º planta  
28043 Madrid  
Tlf: 91 759 59 10

[www.microlog.es](http://www.microlog.es)  
[pedidos@microlog.es](mailto:pedidos@microlog.es)

## ÍNDICE

1. INTRODUCCIÓN
2. PRÁCTICAS
  - 2.1. SEMÁFORO
  - 2.2. SISTEMA DE RIEGO
  - 2.3. PUERTA AUTOMÁTICA
  - 2.4. LÁMPARA
  - 2.5. SECADOR AUTOMÁTICO
  - 2.6. VENTILADOR CON CONTROL DE TEMPERATURA
  - 2.7. FAROLA INTELIGENTE
  - 2.8. VENTILADOR DE VELOCIDAD AJUSTABLE
  - 2.9. CONTROL DE INVASIÓN
  - 2.10. COCHE CON VELOCIDAD AJUSTABLE
  - 2.11. COCHE SIGUE LÍNEAS
  - 2.12. COCHE EVITA OBSTÁCULOS

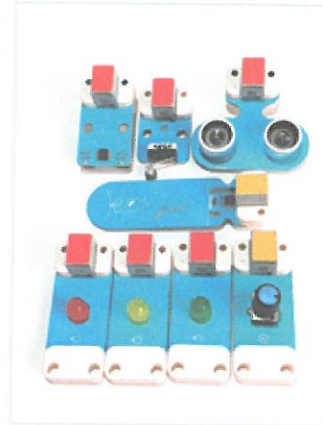
Más información en la web <https://microbit.microlog.es/nezha-v2>

## 1. INTRODUCCIÓN

El kit para inventores Nezha es un pack de piezas y sensores para Micro:bit con el que podrás crear numerosos proyectos.

Contiene los siguientes elementos:

- Led verde
- Led rojo
- Led amarillo
- Sensor de humedad
- Sensor de línea
- Sensor ultrasonido
- Sensor de contacto
- Potenciómetro
- 2 motores
- 1 servomotor
- Bloque de conexiones con batería
- Cables de conexión RJ
- Cable USB
- Piezas compatibles LEGO



Es compatible con piezas Lego y FischerTechnik.

Para programar los proyectos que construyas con Nezha, utilizaremos el software MakeCode con la extensión PlanetX.

**Este kit no requiere de ningún tipo de instalación.**

**Para cualquier duda con el material o la realización de las prácticas, puede ponerse en contacto con Microlog.**

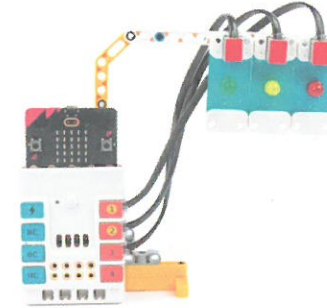
**Teléfono de contacto: 601 150 514**

**Correo electrónico: [direcciontecnica@microlog.es](mailto:direcciontecnica@microlog.es)**

## 2. PRÁCTICAS

### 2.1. SEMÁFORO

Realizamos un semáforo que enciende sus leds siguiendo el orden verde – amarillo – rojo.

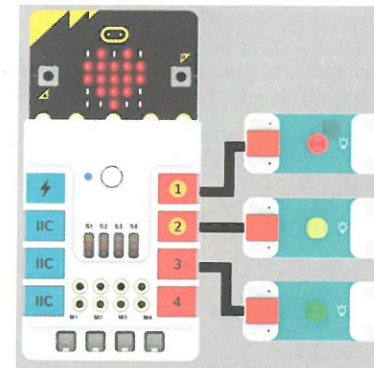


#### LISTA DE MATERIALES:

- Módulo de conexiones Nezha
- Placa Micro:bit
- Led Rojo
- Led Verde
- Led Amarillo
- 3 Cables RJ11
- Piezas Lego

#### CIRCUITO DE CONEXIONES:

Conecta los leds y la placa Micro:bit como se muestra en el siguiente esquema:



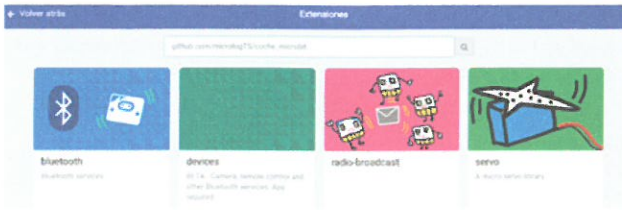
**PROGRAMACIÓN:**

Accede a la web <https://makecode.microbit.org/> y pulsa en crear nuevo proyecto. Asígnale un nombre al proyecto y pulsa en crear.

Una vez dentro de makecode, pulsa en el icono del engrane (arriba a la derecha).



En el menú que se ha desplegado, selecciona la opción "extensiones".



En el cuadro de búsqueda introduce PlanetX y pulsa la lupa. Encontrarás la siguiente extensión:



Pulsa sobre la extensión y automáticamente se añadirán nuevos bloques a tu proyecto que podrás utilizar para programar los sensores de Nezha. Realiza los mismos pasos para añadir la extensión Nezha y tendrás tu espacio listo para empezar a programar. En cada proyecto que realices con nezha, debes seguir estos pasos para poder utilizar los bloques de programación adecuados a este kit.

El programa de este proyecto debe realizar las siguientes acciones:

*En un bucle infinito:*

*Enciende el led verde*

*Realiza una pausa (para mantener el led encendido unos segundos)*

*Apaga el led verde*

*Enciende el led amarillo*

*Realiza una pausa (para mantener el led encendido unos segundos)*

*Apaga el led amarillo*

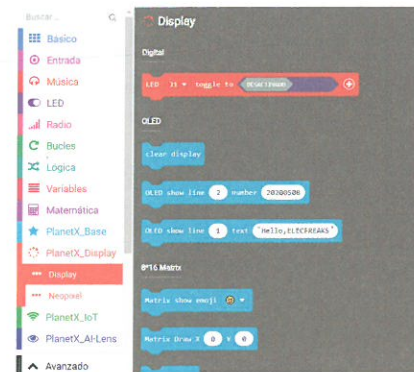
*Enciende el led rojo*

*Realiza una pausa (para mantener el led encendido unos segundos)*

*Apaga el led rojo*

A la vez que se ejecutan las órdenes dentro de un bucle infinito, se repite el ciclo continuamente.

Para localizar el bloque que enciende y apaga los leds, pulsa en PlanetX\_Display.

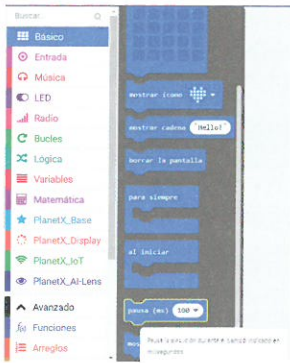


A continuación pulsa en Display y localiza el siguiente bloque:

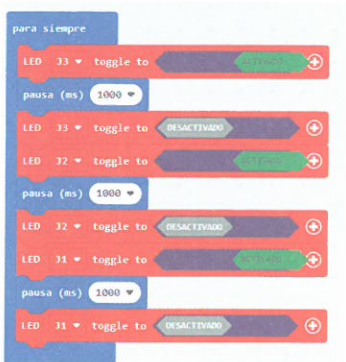


- J3 Activado / Desactivado – Enciende / Apaga el Led verde
- J2 Activado / Desactivado – Enciende / Apaga el Led amarillo
- J1 Activado / Desactivado – Enciende / Apaga el Led rojo

Si pulsas en básico, localizarás el bloque para realizar una pausa:



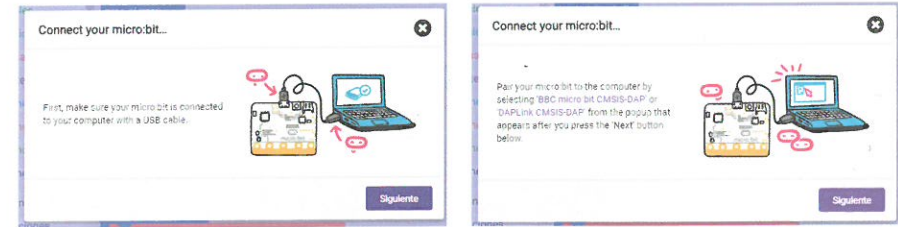
Arrastrando estos bloques dentro del bucle "para siempre" creamos el siguiente código:



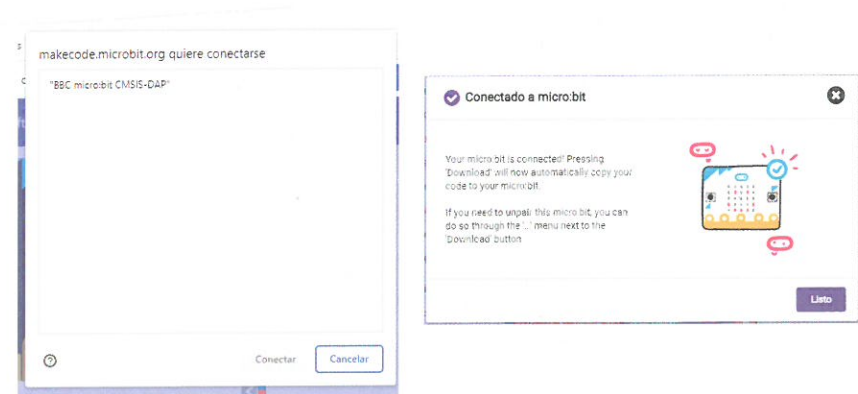
Para probar el programa, emparejamos la placa Micro:bit con el sistema. De esta forma los programas se cargarán en la placa con solo pulsar un botón. En la pantalla de makecode verás que abajo a la izquierda hay un botón morado llamado «descargar». Al lado tienes un icono con 3 puntos (...). Pulsa sobre los 3 puntos:



Aparece un pequeño menú, pulsa sobre «emparejar» o «pair device» y verás la siguiente pantalla.



Si no tenías conectada la placa al PC a través del cable USB, conéctala y pulsa en «emparejar» o «pair device». Se mostrará la siguiente pantalla:



Haz clic con el ratón sobre la placa microbit, y pulsa en conectar. A partir de ahora la placa quedará emparejada al ordenador durante toda la sesión de trabajo.

Para guardar el programa en la placa, pulsa el botón morado descargar, abajo a la izquierda. El programa se descargará directamente a la placa microbit y verás funcionar el semáforo.

## 2.2. SISTEMA DE RIEGO

Realizamos un sistema que nos indique si la tierra de una planta está húmeda o si necesita ser regada.

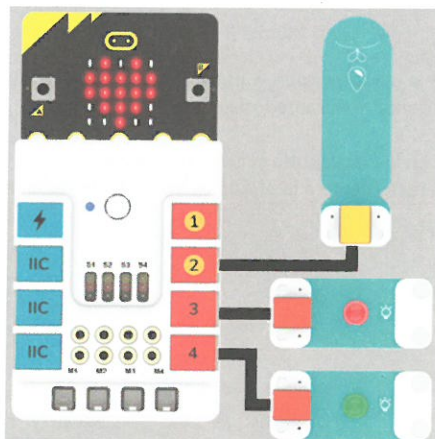


### LISTA DE MATERIALES:

- Módulo de conexiones Nezha
- Placa Micro:bit
- Led Rojo
- Led Verde
- Sensor de humedad
- 3 Cables RJ11
- Piezas Lego

### CIRCUITO DE CONEXIONES:

Conecta los leds, el sensor y la placa Micro:bit como se muestra en el siguiente esquema:



### PROGRAMACIÓN:

Crema un nuevo proyecto en makecode y agrega las extensiones planetX y Nezha. El programa de este proyecto debe realizar las siguientes acciones:

*En un bucle infinito:*

*Realiza la lectura del sensor de humedad.*

*Si el sensor indica un valor inferior a 30, enciende el led rojo por falta de humedad*

*Si el sensor indica un valor superior a 30, enciende el led verde.*

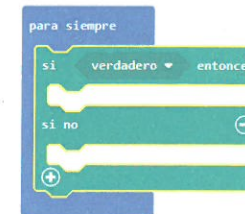
Al estar las órdenes dentro de un bucle infinito, se repite el ciclo continuamente.

El bloque para la lectura del sensor de humedad lo encontrarás en planetX\_base, dentro de los bloques de sensores (Soil moisture sensor).

Soil moisture sensor J1 value(0-100)

Para realizar la comparación de la lectura del sensor de humedad, pulsa en lógica y localiza el bloque "si...entonces...si no..."

Arrastra el bloque dentro de "para siempre".



Para crear la condición, pulsa sobre lógica y localiza el operador <. Arrastra el operador sobre "verdadero" en el bloque "si... entonces". A la izquierda del operador, introduce el bloque de lectura del sensor y a la derecha el valor de comparación (30). Selecciona dentro del bloque de lectura del sensor la conexión J2.

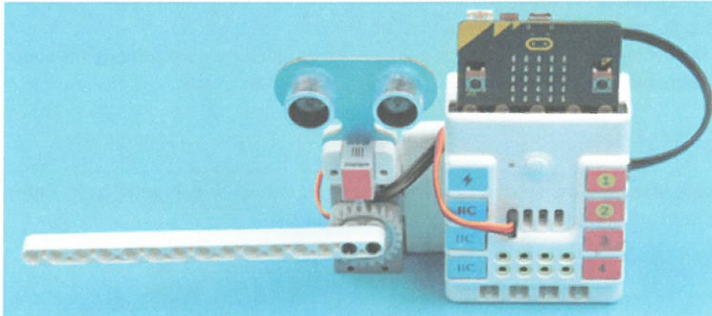
Si la condición es correcta, se enciende el led rojo y en caso contrario el led verde:



Pulsa en descargar y comprueba el resultado.

## 2.1 PUERTA AUTOMÁTICA

Realiza una barrera de acceso que se abrirá cuando detecte la presencia de un vehículo.

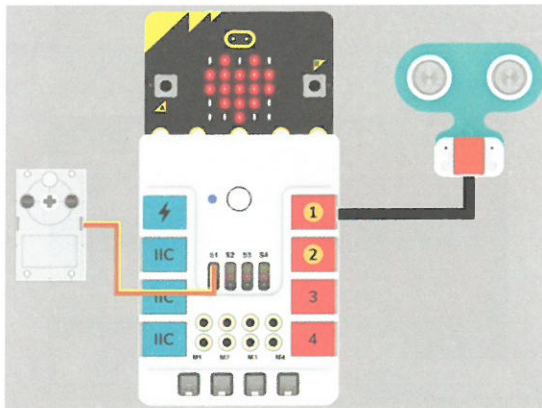


### LISTA DE MATERIALES:

- Módulo de conexiones Nezha
- Placa Micro:bit
- Servomotor
- Sensor ultrasonido
- 1 Cable RJ11
- Piezas Lego

### CIRCUITO DE CONEXIONES:

Conecta el sensor, el motor, y la placa Micro:bit como se muestra en el siguiente esquema:



### PROGRAMACIÓN:

Creas un nuevo proyecto en makecode y agrega las extensiones planetX y Nezha. El programa de este proyecto debe realizar las siguientes acciones:

*En un bucle infinito:*

*Realiza la lectura del sensor ultrasonido.*

*Si el sensor indica un valor inferior a 20cm, y superior a 3cm mueve el motor y abre la barrera.*

*Si el sensor indica un valor superior a 20cm o inferior a 3cm, mueve el motor y cierra la barrera.*

Al estar las órdenes dentro de un bucle infinito, se repite el ciclo continuamente.

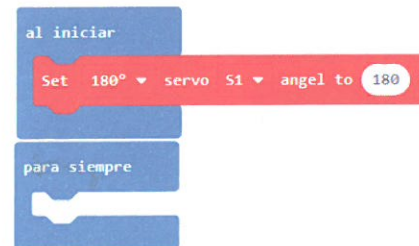
Al iniciar el programa, colocamos la barrera en posición de cerrado. Localizamos el bloque para mover el servomotor dentro de la sección Nezha.

Set 180° ▾ servo S1 ▾ angel to 0 °

Configuramos el bloque con los siguientes parámetros:

- Tipo de servomotor: 360°
- Conexión: pin S1
- Ángulo de posición: 180°

Arrastramos el bloque dentro de "al iniciar" para que sea la primera acción a realizar.



El bloque para la lectura del sensor ultrasonido lo encontrarás en planetX\_base, dentro de los bloques de sensores (Ultrasonic sensor)

Ultrasonic sensor J1 ▾ distance cm ▾

Para realizar la comparación de la lectura del sensor ultrasonido, pulsa en lógica y localiza el bloque “si...entonces...si no...”

```

si verdadero entonces
si no
    
```

Arrastra el bloque dentro de “para siempre”

```

al iniciar
Set 180° servo S1 angel to 180°

para siempre
si verdadero entonces
si no
    
```

Para detectar un coche a una distancia entre 3 y 20 cms, tenemos que unir dos comparaciones utilizando el operador “y”. En lógica y dentro de booleano localiza el operador “Y”. Arrástralo dentro de “verdadero”.

```

al iniciar
Set 180° servo S1 angel to 180°

para siempre
si y entonces
si no
    
```

Con el operador “Y” unimos la comparación distancia > 3 y distancia < 20. Si pulsas en lógica localizarás el operador <. Arrástralo dentro de los hexágonos del operador “y”. Si pulsas sobre el comparador, podrás elegir < o >.

```

al iniciar
Set 180° servo S1 angel to 180°

para siempre
si 0 > 3 y 0 < 20 entonces
si no
    
```

Completamos las comparaciones añadiendo la lectura del sensor ultrasonido y las distancias que queremos medir.

```

al iniciar
Set 180° servo S1 angel to 180°

para siempre
si Ultrasonic sensor J1 distance cm > 3 y Ultrasonic sensor J1 distance cm < 20 entonces
si no
    
```

Solo queda agregar el movimiento del motor para abrir y cerrar la barrera:

```

al iniciar
Set 180° servo S1 angel to 180°

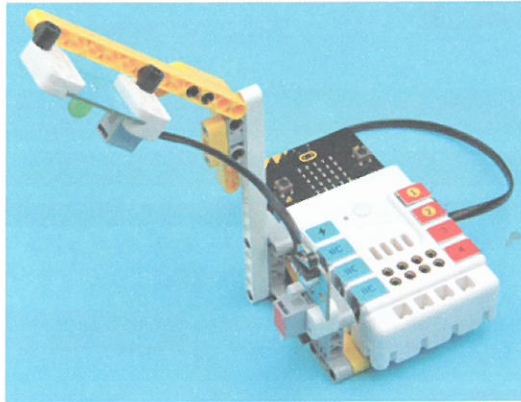
para siempre
si Ultrasonic sensor J1 distance cm > 3 y Ultrasonic sensor J1 distance cm < 20 entonces
Set 180° servo S1 angel to 90°
pausa (ms) 1000
si no
Set 180° servo S1 angel to 180°
    
```

Para probar la barrera, pulsa en descargar. Acerca un objeto a la barrera y comprueba que ésta se abre. Si alejas el objeto, ésta se cerrará.



### 2.4. LÁMPARA

Creamos una pequeña lámpara de mesa que podremos encender y apagar utilizando el sensor de contacto formado por un final de carrera.

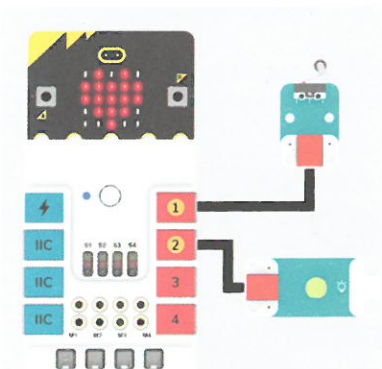


#### LISTA DE MATERIALES:

- Módulo de conexiones Nezha
- Placa Micro:bit
- LED
- Sensor de contacto
- 2 Cables RJ11
- Piezas Lego

#### CIRCUITO DE CONEXIONES:

Conecta el sensor, el motor, y la placa Micro:bit como se muestra en el siguiente esquema:



#### PROGRAMACIÓN:

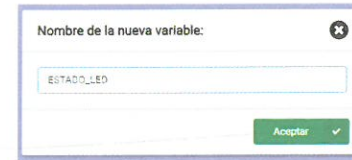
Crema un nuevo proyecto en makecode y agrega las extensiones planetX y Nezha.

Creamos una variable para controlar si el led está encendido o apagado.

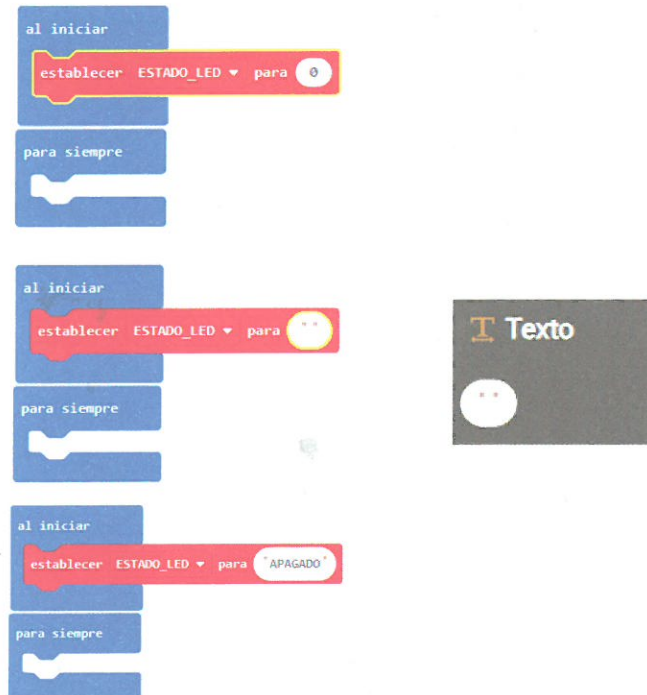
Cuando se presione el final de carrera

- Si la variable contiene el valor "encendido", procedemos a apagar el led y almacenar en la variable estado el valor "apagado"
- Si la variable contiene el valor "apagado", procedemos a encender el led y almacenar en la variable el valor "encendido"

Para crear la variable, pulsa en variables, y asígnale un nombre, por ejemplo "estado\_led".



En el bloque "al iniciar", establecemos la variable estado del "LED" como "apagado". Para poder asignar a la variable un valor no numérico, arrastramos dentro del bloque establecer, el bloque "" que localizarás dentro de Texto:



Para saber si se ha pulsado el final de carrera del sensor, utilizamos el bloque "Crash sensor" que localizarás en planetX\_Base (Sensor)



Utilízalo dentro de un condicional para comprobar si se ha presionado. Después, con otro condicional, comprueba si el led está encendido o apagado y realiza la acción correspondiente sobre el led:

```

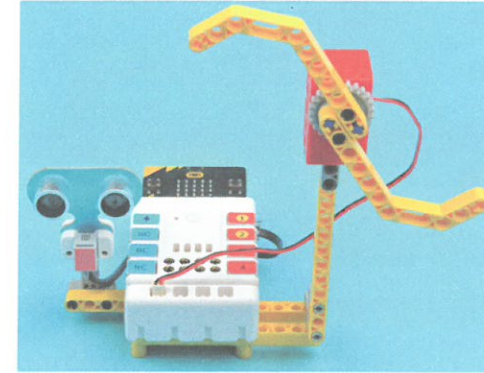
al iniciar
  establecer ESTADO_LED para "APAGADO"

para siempre
  si Crash Sensor J1 is pressed entonces
    si ESTADO_LED == "APAGADO" entonces
      establecer ESTADO_LED para "ENCENDIDO"
      LED J2 toggle to ON
    si no
      establecer ESTADO_LED para "APAGADO"
      LED J2 toggle to DESACTIVADO
  
```

Pulsa en descargar, y comprueba que con el sensor puedes encender y apagar la lámpara.

### 2.5. SECADOR AUTOMÁTICO

Realizaremos un ventilador que girará de forma automática.

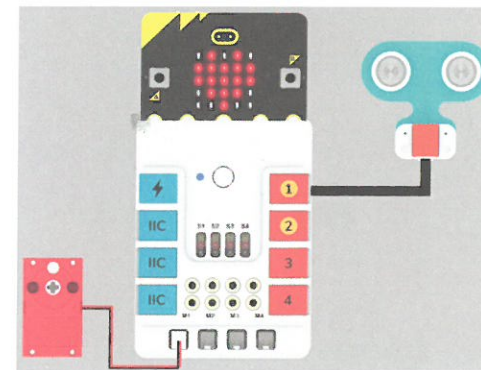


#### LISTA DE MATERIALES:

- Módulo de conexiones Nezha
- Placa Micro:bit
- Motor
- Sensor de ultrasonido
- 1 Cables RJ11
- Piezas Lego

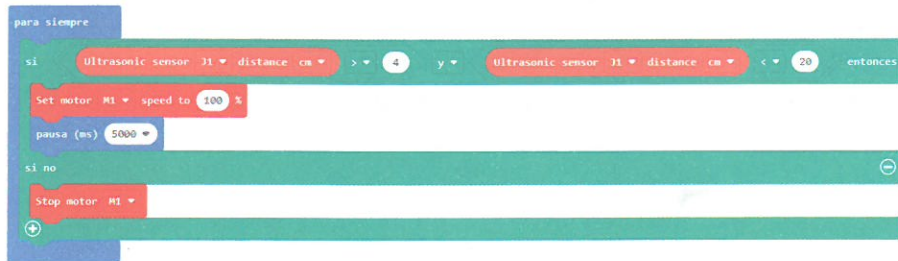
#### CIRCUITO DE CONEXIONES:

Conecta el sensor, el motor, y la placa Micro:bit como se muestra en el siguiente esquema:



**PROGRAMACIÓN:**

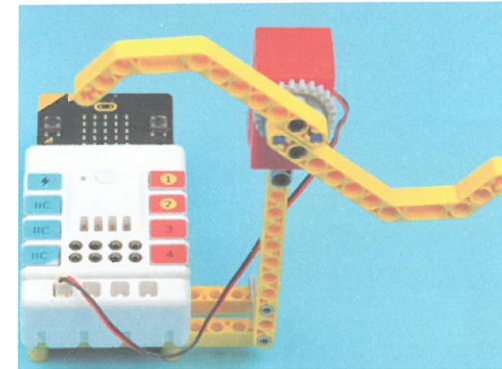
Crea un nuevo proyecto en makecode y agrega las extensiones planetX y Nezha. Cuando el sensor ultrasonido detecte un objeto a una distancia entre 4 y 20 cms, se activará el ventilador.



Pulsa en el botón descargar, y comprueba su funcionamiento.

**2.6. VENTILADOR CON CONTROL DE TEMPERATURA**

En este caso el ventilador se moverá dependiendo de la temperatura ambiente.

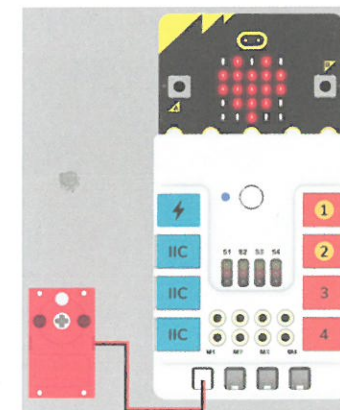
**LISTA DE MATERIALES:**

- Módulo de conexiones Nezha
- Placa Micro:bit
- Motor
- Piezas Lego

**CIRCUITO DE CONEXIONES:**

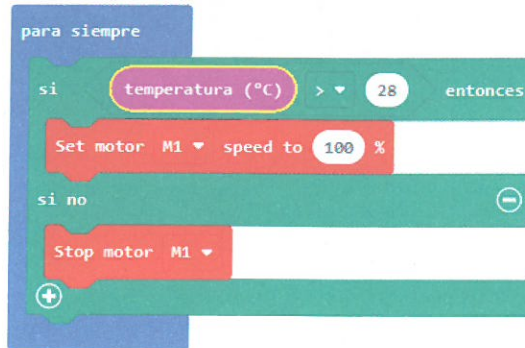
Conecta el motor, y la placa Micro:bit como se muestra en el siguiente esquema:

El sensor de temperatura se encuentra dentro de la propia placa microbit.



**PROGRAMACIÓN:**

Crea un nuevo proyecto en makecode y agrega las extensiones planetX y Nezha.  
En un bucle infinito, realizamos una lectura del termómetro de microbit. Si la temperatura es mayor de 28 grados, activamos el motor.

**2.7. FAROLA INTELIGENTE**

Realizaremos una lámpara que se enciende cuando detecta ausencia de luz.

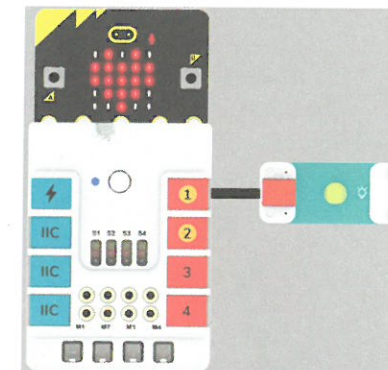
**LISTA DE MATERIALES:**

- Módulo de conexiones Nezha
- Placa Micro:bit
- Led
- 1 Cable RJ11
- Piezas Lego

**CIRCUITO DE CONEXIONES:**

Conecta el Led, y la placa Micro:bit como se muestra en el siguiente esquema:

El sensor de luz se encuentra en la propia placa microbit.



**PROGRAMACIÓN:**

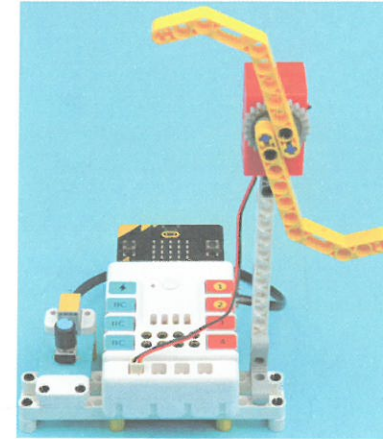
Crea un nuevo proyecto en makecode y agrega las extensiones planetX y Nezha.  
En un bucle infinito, introducimos un bloque condicional.  
Si la luz detectada es inferior a 10, enciende el led. Si no, apaga el led.



Pulsa en el botón descargar, y comprueba su funcionamiento.

**2.8. VENTILADOR DE VELOCIDAD AJUSTABLE**

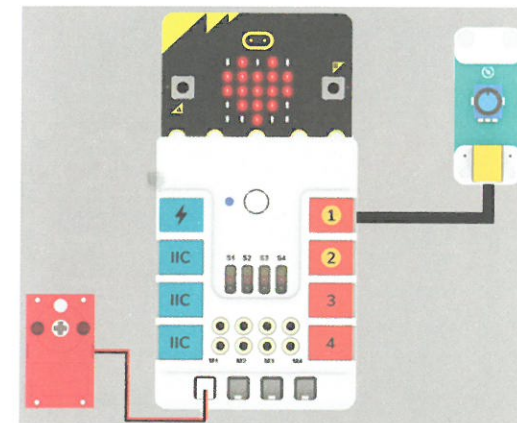
Realizamos un ventilador cuya velocidad de giro podremos controlar.

**LISTA DE MATERIALES:**

- Módulo de conexiones Nezha
- Placa Micro:bit
- Potenciómetro
- Motor
- 1 Cable RJ11
- Piezas Lego

**CIRCUITO DE CONEXIONES:**

Conecta el potenciómetro, y la placa Micro:bit como se muestra en el siguiente esquema:



**PROGRAMACIÓN:**

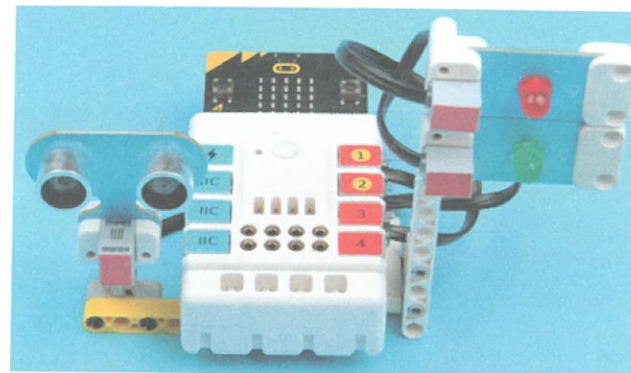
Crea un nuevo proyecto en makecode y agrega las extensiones planetX y Nezha.  
 En un bucle infinito, asignamos al motor una velocidad de movimiento en base al valor que proporciona el potenciómetro.



Pulsa en descargar y comprueba el resultado.

**2.9. CONTROL DE INVASIÓN**

Realizamos un dispositivo capaz de controlar el acceso

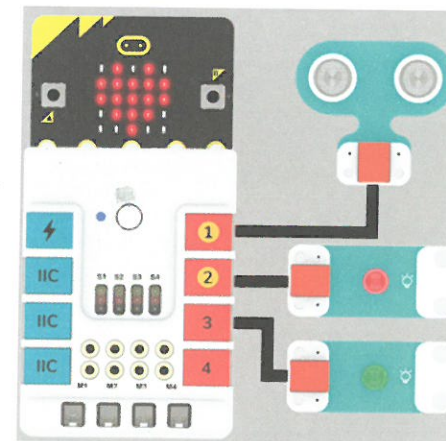


**LISTA DE MATERIALES:**

- Módulo de conexiones Nezha
- Placa Micro:bit
- Sensor ultrasonido
- 2 leds
- 3 Cable RJ11
- Piezas Lego

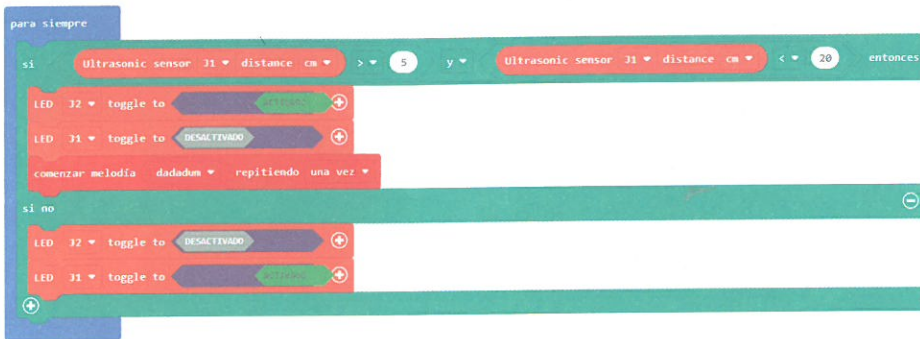
**CIRCUITO DE CONEXIONES:**

Conecta los leds, el sensor y la placa Micro:bit como se muestra en el siguiente esquema:



**PROGRAMACIÓN:**

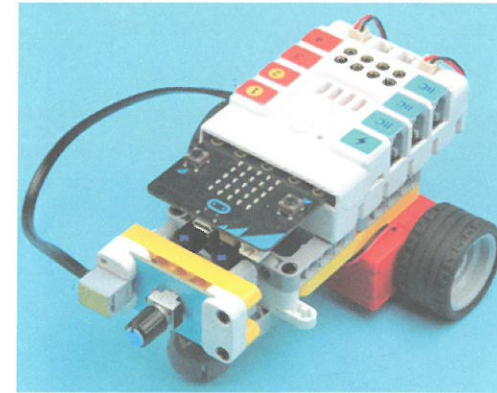
Creo un nuevo proyecto en makecode y agrega las extensiones planetX y Nezha.  
 En un bucle infinito, realizamos la lectura del sensor ultrasonido. Si detecta un objeto cercano se ilumina el led rojo y suena una melodía. Si no hay un objeto cercano, se ilumina el led verde.



Pulsa en descargar y comprueba el funcionamiento del proyecto.

**2.10. COCHE CON VELOCIDAD AJUSTABLE**

Realizamos un coche cuya velocidad podamos controlar con un potenciómetro

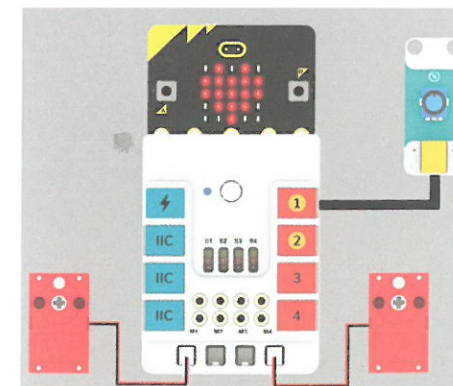


**LISTA DE MATERIALES:**

- Módulo de conexiones Nezha
- Placa Micro:bit
- Potenciómetro
- 2 Motores
- 1 Cable RJ11
- Piezas Lego

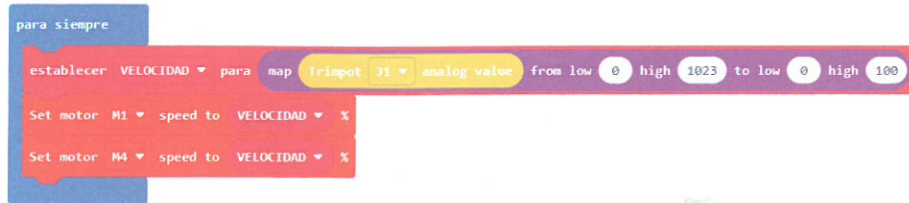
**CIRCUITO DE CONEXIONES:**

Conecta los motores, el potenciómetro, y la placa Micro:bit como se muestra en el siguiente esquema:



**PROGRAMACIÓN:**

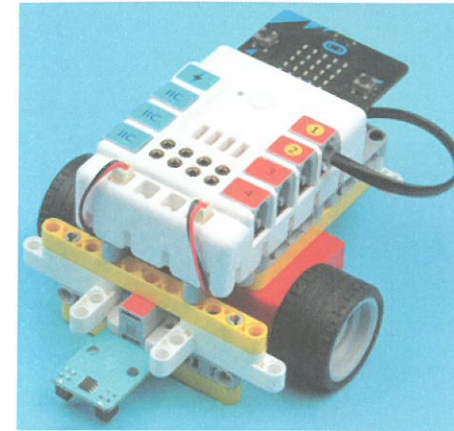
Crea un nuevo proyecto en makecode y agrega las extensiones planetX y Nezza.  
En un bucle infinito, realizamos la lectura del potenciómetro y asignamos la velocidad a los motores en función de este.



Pulsa en descargar y comprueba el funcionamiento del proyecto.

**2.11. COCHE SIGUE LÍNEAS**

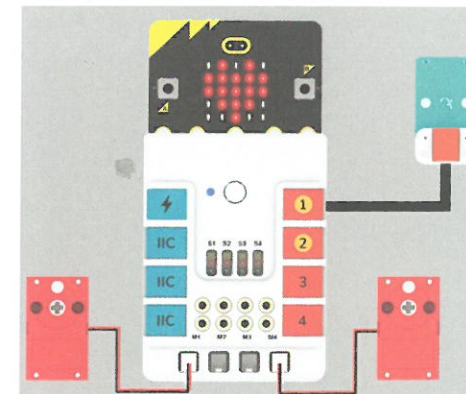
Realizamos un coche capaz de seguir el camino trazado con una línea negra.

**LISTA DE MATERIALES:**

- Módulo de conexiones Nezza
- Placa Micro:bit
- Sensor de línea
- 2 Motores
- 1 Cable RJ11
- Piezas Lego

**CIRCUITO DE CONEXIONES:**

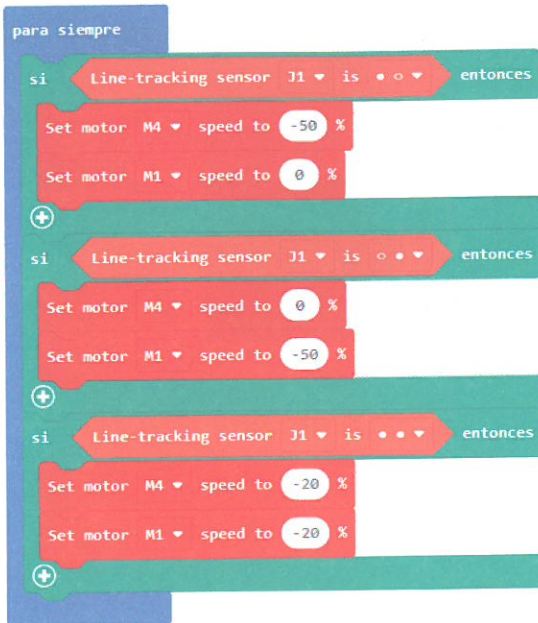
Conecta los motores, el sensor, y la placa Micro:bit como se muestra en el siguiente esquema:





## PROGRAMACIÓN:

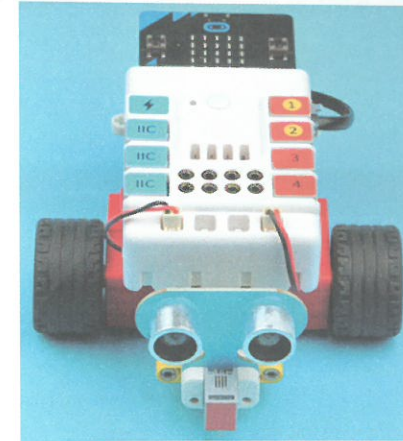
Crea un nuevo proyecto en makecode y agrega las extensiones planetX y Nezha.  
 En un bucle infinito, realizamos la lectura del sensor de línea.  
 Si el coche está sobre la línea negra, el coche avanza recto  
 Si el coche está fuera de la línea negra, realiza un giro



Pulsa en descargar y comprueba el funcionamiento del proyecto.

## 2.12. COCHE EVITA OBSTÁCULOS

Realizamos un coche capaz de no chocarse con ningún obstáculo.



## LISTA DE MATERIALES:

- Módulo de conexiones Nezha
- Placa Micro:bit
- Sensor ultrasonido
- 2 Motores
- 1 Cable RJ11
- Piezas Lego

## CIRCUITO DE CONEXIONES:

Conecta los motores, el sensor, y la placa Micro:bit como se muestra en el siguiente esquema:

