

# Introducción

- [Objetivos y contenidos](#)
- [¿Qué es Pico Bricks?](#)
- [Proyectos](#)
- [Libros](#)
- [Pensamiento computacional](#)
- [Lenguaje de programación](#)

# Objetivos y contenidos

## Objetivos

- Conocer las posibilidades didácticas del kit Picobricks Pico
- Ser capaz de realizar proyectos dando opción a programar en bloques o en código, dependiendo de la etapa educativa que imparte el docente.
- Sentar las bases para realizar proyectos STEAM con este kit

## Contenidos

**SOLO SE EXIGIRÁ EL DOMINIO DE UN LENGUAJE DE LOS CUATRO MOSTRADOS**

### **DOS LENGUAJES DE BLOQUES GRÁFICO (PicoBlocky y Microblocks) Y DOS LENGUAJES DE CÓDIGO (Python y Arduino IDE)**

- **Introducción**
  - ¿Qué es Pico Bricks?
  - Proyectos a realizar
    - PROYECTO BLINK
    - PROYECTO ACTION-REACTION
    - PROYECTO Autonomous Lighting
    - PROYECTO Thermometer
    - PROYECTO Graphic Monitor
    - PROYECTO Dominate the Rhythm
    - PROYECTO Show Your Reaction
    - PROYECTO My Timer
    - PROYECTO Alarm Clock
    - PROYECTO Know Your Color
- **PicoBlocky**
  - Interface
  - Conexión
  - Dos formas de ejecutar los programas
  - Proyectos
  - Algo diferente PROYECTO IR
- **Microblocks**
  - Instalación
  - Proyectos



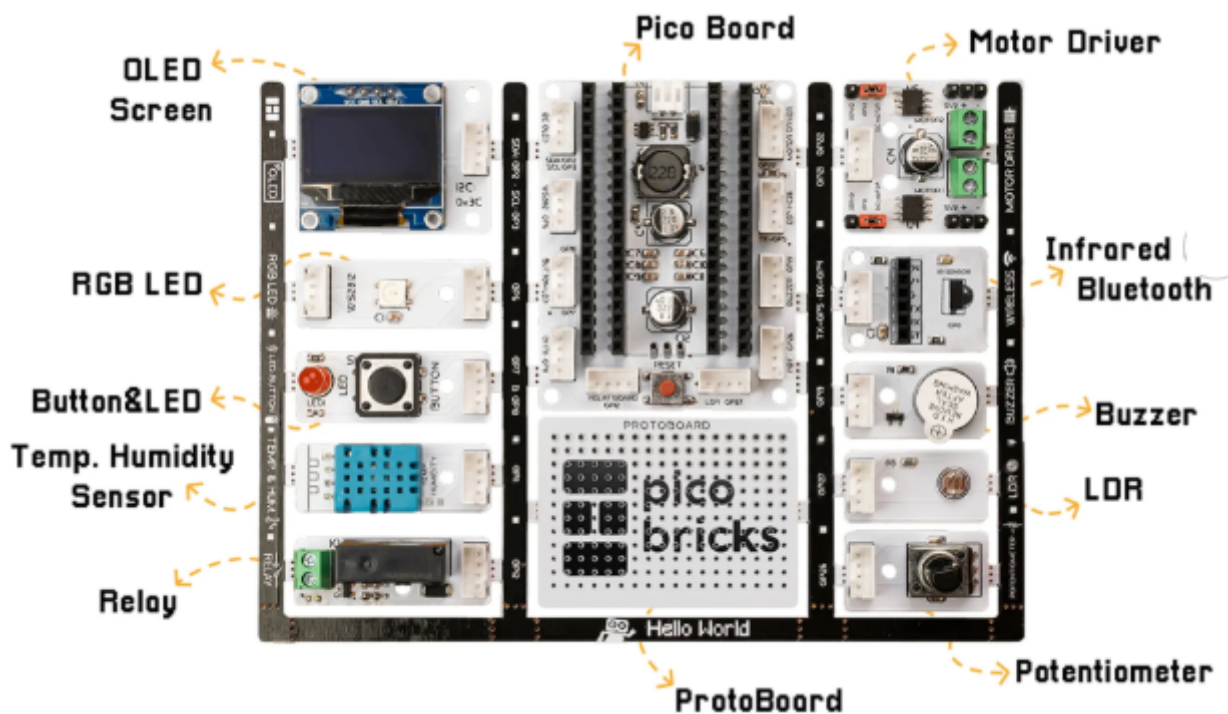
- Aldo diferente Data Graph
- **Micropython con Thonny**
  - Instalación
  - Algo diferente: Apagar y encender por Wifi
- **Arduino IDE**
  - Conexión con Arduino IDE
  - Proyectos
  - Proyectos con Wifi

# ¿Qué es Pico Bricks?

Pico Bricks pertenece a las placas electrónicas con sensores y actuadores básicos integrados, pero también preparado para poner externos y así poder hacer proyectos maker.

[2024-12-26 09\\_23\\_28-Pico Bricks IDE book - PDF-XChange Viewer.png](#)

## Hardware



- **Placa microcontroladora:** Raspberry pi Pico W.
- **Sensores o **entradas****
  - Botón
  - Sensor Temperatura y Humedad
  - Potenciómetro
  - Sensor IR con mando
  - LDR
- **Actuadores o **salidas****
  - OLED Screen
  - Led rojo
  - RGB Led
  - Relé
  - Driver motor

- o Buzzer

## ¿Qué es la Raspberry Pi Pico W?

Es una placa controladora con las siguientes características: ( adaptado de [Bricogeek](#) Licencia CC-BY )

- **Chip RP2040:** Un microcontrolador con un procesador ARM Cortex-M0 de doble núcleo funcionando a 133 MHz. Este es el corazón de la placa y le da su potencia.
- **Memoria:** 264 KB de RAM SRAM y 2 MB de memoria Flash. Aunque no es tanto como un Raspberry Pi tradicional, es más que suficiente para la mayoría de los proyectos de microcontroladores.
- **Conectividad:** 2.4GHz Wifi 802.11n
- **Pines:** 26 pines GPIO (General Purpose Input Output), de los cuales muchos son multifunción, incluyendo UART, SPI, I2C, PWM, y pines analógicos (ADC).
- **Alimentación:** Puede funcionar con una fuente de alimentación de entre 1.8V y 5.5V, lo que la hace versátil para distintas fuentes de energía, como baterías o conexiones USB.
- **Interfaz de programación:** Puedes programarla usando C/C , MicroPython, o el Arduino IDE, lo que la hace- extremadamente accesible para la comunidad de makers.

### P: ¿Qué diferencia tiene con respecto a otros microcontroladores?

R: Con respecto al de las placas Arduino UNO y similares (Atmegaxxx...) es más potente, permite la programación MicroPython, y sobre todo **tiene wifi incorporado**

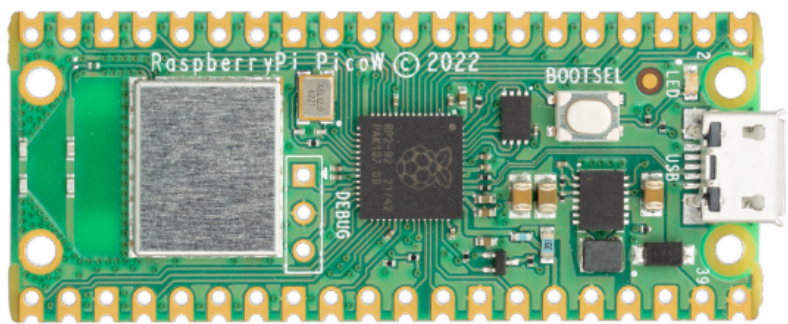
### P: ¿Igual que el ESP32?

R: No, el ESP32 tiene el Bluetooth y el Raspberry Pi Pico W **NO**.

### P: Entonces... ¿Cuál es mejor? ¿el ESP32 o Raspberry Pi Pico W?

R: Pues si tus proyectos no necesitan Bluetooth, ni sensores de tacto que tiene el ESP32, el Raspberry Pi Pico W es muy buena opción pues sólo cuesta 8€, además Raspberry tiene PIO (Programmable Input/Output) que permite visualizar en pantallas VGA [+info](#)

**Importante:** Tienes que localizar el botón **BOOTSEL** que lo hablaremos en este curso

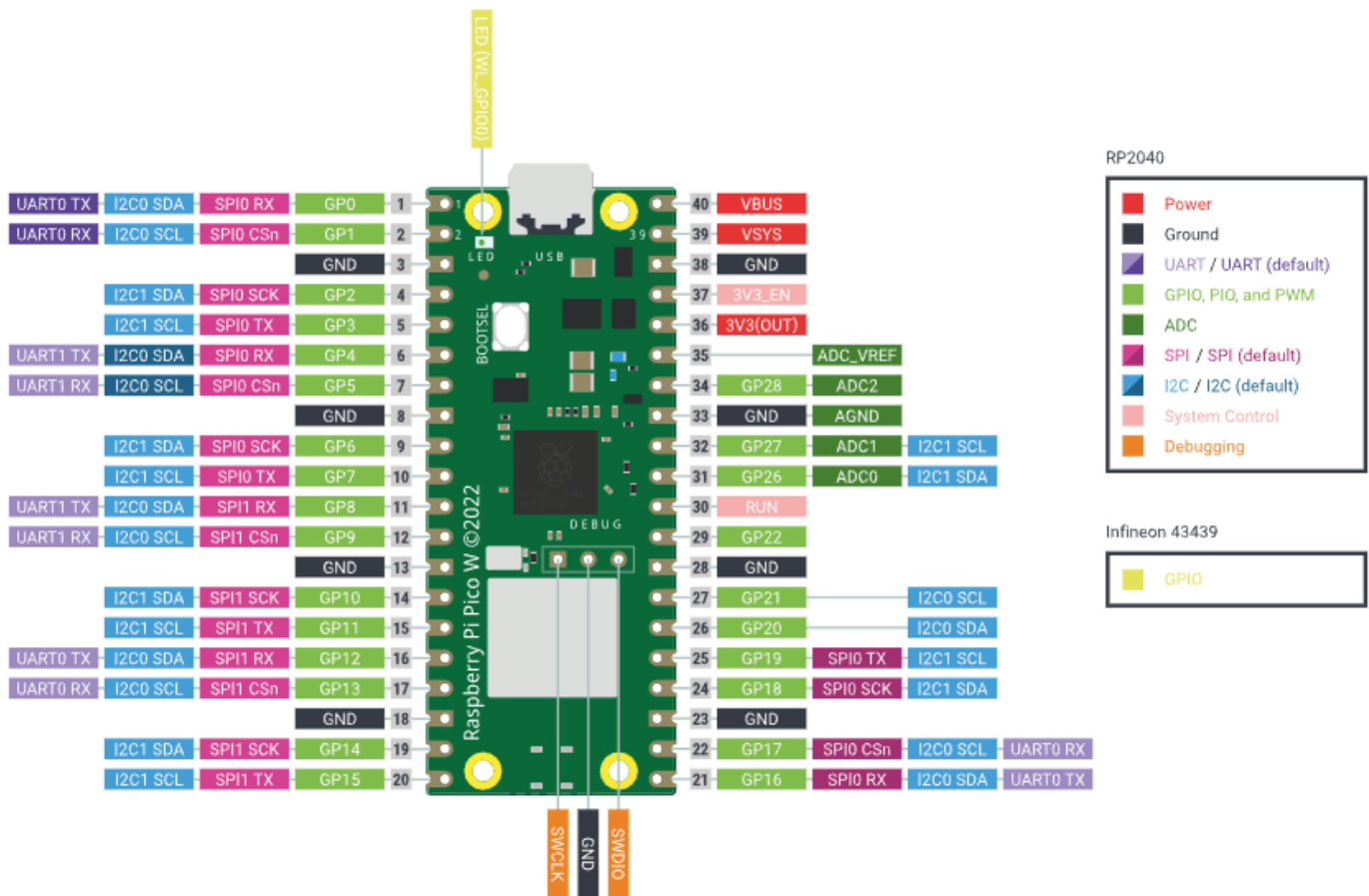


Fuente Datasheet Pico W

<https://datasheets.raspberrypi.com/picow/pico-w-datasheet.pdf>

## Pines GPIO

Intentamos que los proyectos que se muestran no necesites componentes externos, con los que hay en Picobricks sobra. Pero enseñamos algún proyecto que con un componente externo barato y fácilmente conseguible, que nos parece interesante. Para conectar estos componentes, es necesario que sepas los pines GP, la alimentación y la masa dónde están :



Fuente CC-BY <https://www.raspberrypi.com/documentation/microcontrollers/pico-series.html>

## Software

Puede programarse con multitud de plataformas de código o de bloques gráficos



bricksIDE

Thonny IDE

ARDUINO

MicroPython



MicroBlocks

circuit python

Fuente <https://picobricks.com/products/raspberry-pi-pico-w-kit>

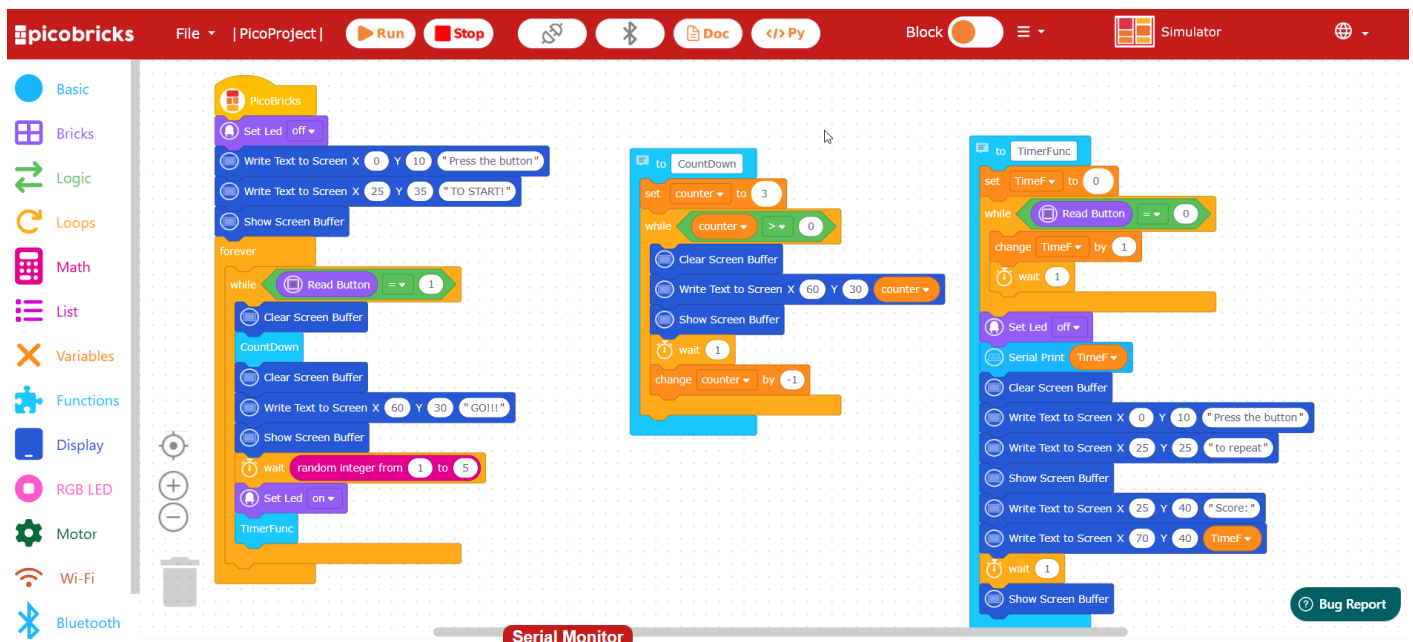
## ¿Qué software se va a utilizar en este curso?

Vamos a ver cuatro programas pertenecientes a dos formas de programar :

- **PROGRAMACIÓN POR BLOQUES** adecuado para primaria y primeros cursos de secundaria
  - **PICOBLOCKLY** perteneciente a los 4 programas de Pico Bricks IDE
  - **MICROBLOCKS** de software libre, multi tarjeta y popularmente extendido
- **PROGRAMACIÓN POR CÓDIGO** adecuado para secundaria
  - **THONNY IDE** con el lenguaje micropython
  - **ARDUINO IDE** con el lenguaje de código Arduino (C++)

## ¿Cuál es el recomendable?

Para programación con bloques: **PICOBLOCKLY** de **Pico Bricks IDE** que es un software propio y tiene unos tutoriales muy cómodos (en inglés), aunque si ya trabajas con **Microblocks**, sigue con él, pues es muy potente

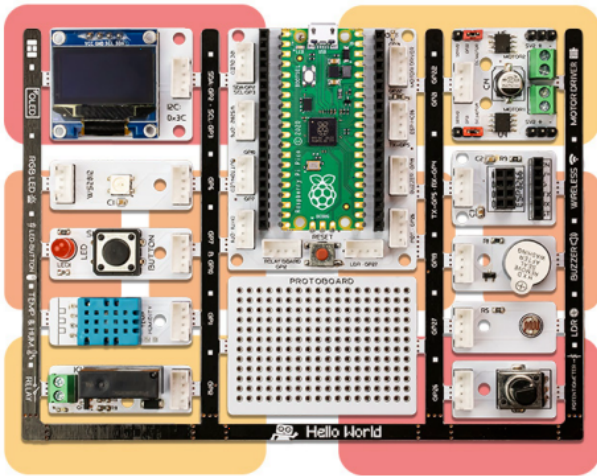


si quieres trabajar con código, el que quieras:

- Si estas acostumbrado al Python, **MicroPython con Thonny**,

- Si ya trabajas con **Arduino** puedes seguir trabajando con PicoBricks

## Opciones de compra



Fuente Pico Bricks IDE Book CC-BY-SA <https://picobricks.com/pages/idebook> ver créditos

Picobricks se venden en muchas tiendas y con muchas opciones y accesorios. Tienes que tener en cuenta que :

- ☐☐ **Lo más barato es ...**
  - Pack básico imprescindible:
    - [Ultra-lab 57€](#)
    - [Kubii 50€](#)
    - comprar **solo la placa** unos [35€](#)
    - Curiosamente en la web sale la foto sin el microcontrolador Raspberry Pi Pico W. En Catedu lo hemos comprado y han venido con el microcontrolador correctamente. ☐☐
    - No obstante comprar sólo el microcontrolador **Raspberry Pico W** aparte cuesta unos [8.60€](#) si buscas en otros sitios, lo puedes encontrar más barato pero ojo **que este soldado** los pines para poderlos insertar en el socket de la placa Picobricks
    - OJO No viene con cable. Vale un cable de datos de USB de móvil micro USB, [unos 2€ el cable](#)
    - Si le añades un sensor distancia HC-SR04 [2€](#) y un servo [2.20€](#) y mando IR, unos [2€](#) ya tienes un kit completísimo ☐☐
      - Y si le añades el kit de coche pues permite la conexión de dos motores [11€](#) ya es el novamás ♥♥♥♥♥
- ☐☐ **Lo más cómodo es ....**

- Comprar kits ya hechos ver <https://picobricks.com/collections>

En el caso de que te venga sin el microcontrolador Rasperry Pi Pico W, es muy fácil conectarlo :

[2025-01-24 11\\_56\\_45-EchidnaML.png](#)

Fuente [Microblocks Wiki](#)

# Proyectos

Vamos a ver los siguientes proyectos y vamos a ver que se pueden desarrollar de cuatro formas diferentes:

- **PROGRAMACIÓN POR BLOQUES** adecuado para primaria y primeros cursos de secundaria
  - **PICOBLOCKLY** perteneciente a los 4 programas de Pico Bricks IDE
  - **MICROBLOCKS** de software libre, multi tarjeta y popularmente extendido
- **PROGRAMACIÓN POR CÓDIGO** adecuado para secundaria
  - **THONNY IDE** con el lenguaje micropython
  - **ARDUINO IDE** con el lenguaje de código Arduino (C++)

Los proyectos son

- PROYECTO **BLINK**
  - <https://github.com/Robotistan/PicoBricks/tree/main/Software/Activities/Blink>
- PROYECTO **ACTION-REACTION**
  - [https://github.com/Robotistan/PicoBricks/tree/main/Software/Activities/Action-  
Reaction](https://github.com/Robotistan/PicoBricks/tree/main/Software/Activities/Action-<br/>Reaction)
- PROYECTO **Autonomous Lighting**
  - [https://github.com/Robotistan/PicoBricks/tree/main/Software/Activities/Autonomous%  
20Lighting](https://github.com/Robotistan/PicoBricks/tree/main/Software/Activities/Autonomous%<br/>20Lighting)
- PROYECTO **Thermometer**
  - <https://github.com/Robotistan/PicoBricks/tree/main/Software/Activities/Thermometer>
- PROYECTO **Graphic Monitor**
  - [https://github.com/Robotistan/PicoBricks/tree/main/Software/Activities/Graphic%20M  
onitor](https://github.com/Robotistan/PicoBricks/tree/main/Software/Activities/Graphic%20M<br/>onitor)
- PROYECTO **Dominate the Rhythm**
  - [https://github.com/Robotistan/PicoBricks/tree/main/Software/Activities/Dominate%20  
the%20Rhythm](https://github.com/Robotistan/PicoBricks/tree/main/Software/Activities/Dominate%20<br/>the%20Rhythm)
- PROYECTO **Show Your Reaction**
  - [https://github.com/Robotistan/PicoBricks/tree/main/Software/Activities/Show%20Your  
%20Reaction](https://github.com/Robotistan/PicoBricks/tree/main/Software/Activities/Show%20Your<br/>%20Reaction)
- PROYECTO **My Timer**

- <https://github.com/Robotistan/PicoBricks/tree/main/Software/Activities/My%20Timer>
- PROYECTO **Alarm Clock**
  - <https://github.com/Robotistan/PicoBricks/tree/main/Software/Activities/Alarm%20Clock>
- PROYECTO **Know Your Color**
  - <https://github.com/Robotistan/PicoBricks/tree/main/Software/Activities/Know%20Your%20Color>
- PROYECTO **Buzz Wire Game**
  - <https://github.com/Robotistan/PicoBricks/tree/main/Software/Activities/Buzz%20Wire%20Game>

Además de algún proyecto que creemos que completa la formación en este robot.

Hay más proyectos en <https://github.com/Robotistan/PicoBricks/tree/main/Software/Activities> pero implican usar actuadores y sensores externos pero interesantes y fáciles de conseguir y baratos, por ejemplo un servo, un sensor de distancia ultrasónico, etc..

*Mentirijilla: el proyecto [Buzz Wire Game](#) haría falta unos simples cables*

El mismo repositorio facilita los proyectos ordenándolos de **más fácil a más difícil dificultad**, marcamos los que vamos a dar en este curso

1. - [Blink](#)
2. - [Action-Reaction](#)
3. - [Autonomous Lighting](#)
4. - [Thermometer](#)
5. - [Graphic Monitor](#)
6. - [Dominate the Rhythm](#)
7. - [Show Your Reaction](#)
8. - [My Timer](#)
9. - [Alarm Clock](#)
10. - [Know Your Color](#)
11. - [Magic Lamp](#)
12. - [Smart Cooler](#)

13. [- Buzz Wire Game](#)
14. [- Dinosaur Game](#)
15. [- Night and Day](#)
16. [- Voice Controlled Robot Car](#)
17. [- Two Axis Robot Arm](#)
18. [- Smart House](#)
19. [- Piggy Bank](#)
20. [- RFID Smart Door](#)
21. [- Automatic Trash Bin](#)
22. [- Digital Ruler](#)
23. [-Air Piano](#)
24. [- Maze Solver Robot](#)
25. [- Smart Greenhouse](#)

# Libros

## Project Book

Un libro completo (en inglés) que lo puedes conseguir aquí

<https://picobricks.com/pages/projectbook> que pena que no tiene licencia CC

[https://drive.google.com/file/d/1PDqI\\_GYyxcz68JqmQAGOLs0YE6SXgPCm/preview](https://drive.google.com/file/d/1PDqI_GYyxcz68JqmQAGOLs0YE6SXgPCm/preview)

## IDEBOOK

Este libro esta especializado en realizar proyectos con el software PICOBLOCKS IDE lo puedes conseguir aquí <https://picobricks.com/pages/idebook> y sí que tiene licencia CC

[https://drive.google.com/file/d/1plad6bjn87FcgHb3cpd1vl-B\\_A25rnfF/preview](https://drive.google.com/file/d/1plad6bjn87FcgHb3cpd1vl-B_A25rnfF/preview)

## Teacher Book

Con 20 actividades STEM, <https://picobricks.com/pages/education>

No es descargable, pero puedes solicitarlo gratis en contacto <https://picobricks.com/pages/contact-information-picobricks>

# Pensamiento computacional

¿Dónde encaja Picobricks dentro de la oferta de equipos robóticos para la educación? Cómo puedes ver entra tanto en primaria como secundaria gracias a sus dos modos de programación:

- Programación en bloques (Primaria)
- Programación en código (Secundaria)

Unido al bajo precio y a las prestaciones que tiene con actuadores y sensores, es un buen producto con buena relación calidad/precio

Guía orientativa

[https://docs.google.com/presentation/d/e/2PACX-1vQHiZvv1cGHet7eXVy-QcECY4Lj0k0I7ntDi8MevRWHQX-9myA0bfR5IofMeuGZkWD0Hw-Ob-MGoco\\_/embed?start=trueloop=true&delayms=3000](https://docs.google.com/presentation/d/e/2PACX-1vQHiZvv1cGHet7eXVy-QcECY4Lj0k0I7ntDi8MevRWHQX-9myA0bfR5IofMeuGZkWD0Hw-Ob-MGoco_/embed?start=trueloop=true&delayms=3000)

Tenemos un **grupo Telegram Robótica Educativa en Aragón,**

<https://t.me/roboticaeducativaaragon>





# Lenguaje de programación

El Picobricks se puede programar

- Por bloques
- Por código

Guía de los lenguajes de programación para robots

[https://docs.google.com/presentation/d/e/2PACX-1vQb1Dv9wN9QK-F6V7yvwDoyzquqwWlGvlyVjr83Yk56kAoYD7bXlnYDm\\_tCQkeAgg/pubembed?start=falseloop=false&delayms=3000](https://docs.google.com/presentation/d/e/2PACX-1vQb1Dv9wN9QK-F6V7yvwDoyzquqwWlGvlyVjr83Yk56kAoYD7bXlnYDm_tCQkeAgg/pubembed?start=falseloop=false&delayms=3000)

Tenemos un **grupo Telegram Robótica Educativa en Aragón,**

<https://t.me/roboticaeducativaaragon>

