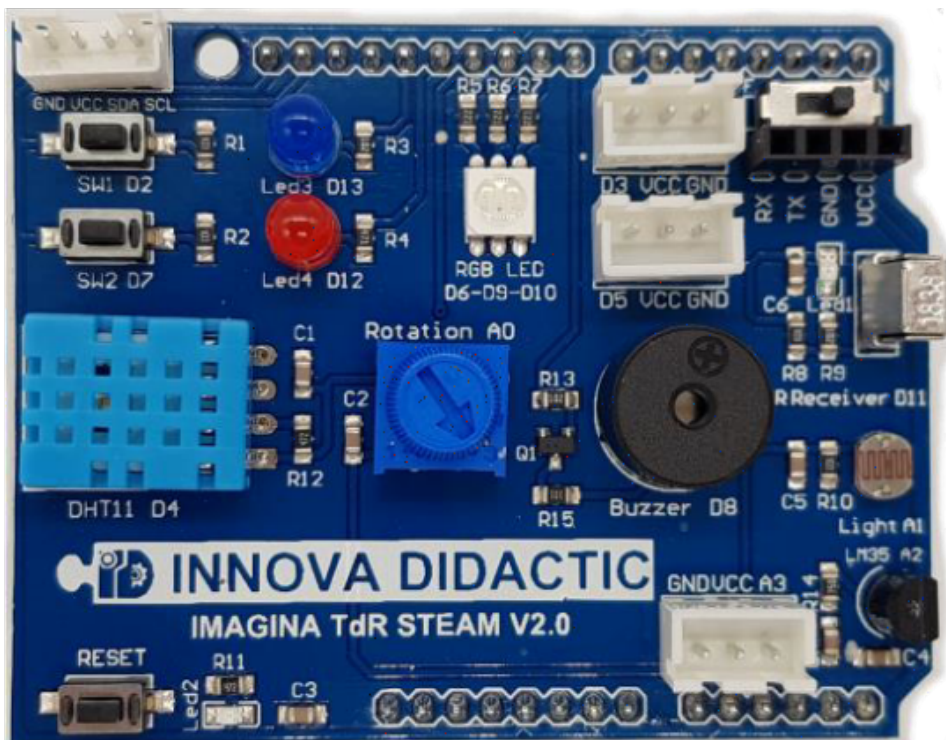


Instrucciones

- 1. Primeros pasos
- 2. Funcionamiento de un sistema de control programado
- 3. Componentes de la placa Imagina TDR STEAM
- 4. Placa de control Keystudio UNO (compatible con Arduino UNO)
- 5. Programación con ArduinoBlocks
- 6. Instalación de ArduinoBlocks

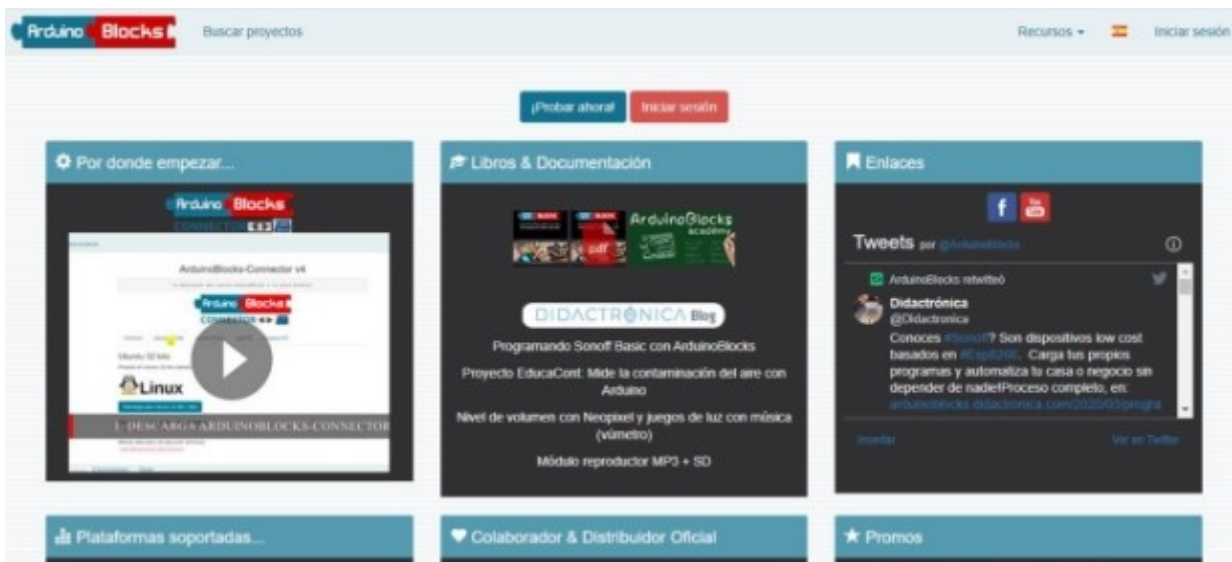
1. Primeros pasos

El presente manual pretende ser una herramienta base para iniciarse en el mundo de la programación, la electrónica y la robótica utilizando para ello la placa basada en Arduino: **Imaginla TDR STEAM** (sobre una placa **Keyestudio UNO**) y el entorno de programación **ArduinBlocks**. La placa Imagina TDR STEAM es una **Shield** (placa/escudo). Significa que es una placa que tiene que ir colocada sobre otra que contiene el sistema de control. En este caso, sobre una placa Keyestudio UNO. La placa Imagina TDR STEAM tiene numerosos sensores y actuadores que permitirán hacer infinidad de proyectos.



En este documento no pretende ser únicamente un manual para aprender programación. El manual presenta una serie de actividades guiadas y retos para aprender a programar de una manera entretenida y divertida mientras aprendemos conceptos relacionados con las S.T.E.A.M. (Science, Technology, Engineering, Arts, Math).

Para realizar la programación con la placa Imagina TDR STEAM utilizaremos un lenguaje de programación visual basado en bloques llamado **ArduinBlocks**. Hay quien podría pensar que un lenguaje de este estilo es muy básico y limitado, pero ya veremos a lo largo del manual la gran potencialidad y versatilidad que tiene este programa.



Con la placa Imagina TDR STEAM tenemos la mayoría de sensores y actuadores que necesitamos para poder introducirnos en el mundo de la programación de entornos físicos (Physical Computing). También tenemos conexiones de expansión para poner conectar más elementos externos.

Página del entorno de programación: ArduinoBlocks.

www.arduinoblocks.com

Página de información: Didactrónica.

www.didactronica.com

Página para comprar material: Innova Didactic S.L.

shop.innovadidactic.com

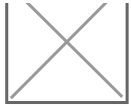
2. Funcionamiento de un sistema de control programado

Un sistema de control programado funciona de manera similar a la de un ser humano. Cuando nuestro cerebro recibe información de los sentidos; oído, olfato, gusto, vista y tacto; analiza esta información, la procesa y da órdenes a nuestros músculos para realizar movimientos, dar estímulos a las cuerdas vocales para emitir sonidos, etc. Los sentidos equivalen a entradas de información y la voz los músculos serían las salidas.



En el caso de un sistema de control programado, un microcontrolador hace la función de cerebro. Este componente electrónico unas entradas de información donde se conectan los sensores de luz (LDR), temperatura (NTC), sonido... y también tiene salidas, donde se conectan los motores, leds, zumbadores, etc.

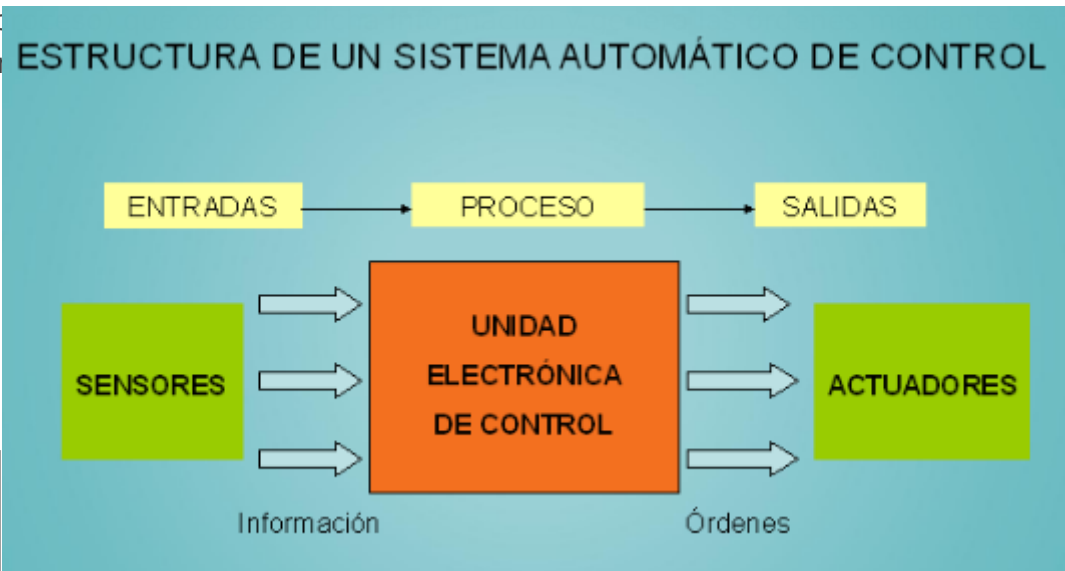




La diferencia principal es que, así como nuestro cerebro ha ido aprendido lo que tiene que hacer a lo largo de nuestra vida a base de estímulos, el sistema programado tiene su memoria vacía, es decir, no sabe lo que debe hacer. Entonces nosotros tenemos que decirle como tiene que actuar en función de las señales que recibe de los sensores. Estas órdenes se generan mediante el código de programa que introducimos en nuestro sistema de control.

En todo sistema de control automático, los sensores (entradas) convierten las magnitudes físicas, químicas, etc. en eléctricas, creando así la información. Esta información se envía a la unidad de control (proceso) que las eléctricas que serán

ESTRUCTURA DE UN SISTEMA AUTOMÁTICO DE CONTROL



3. Componentes de la placa Imagina TDR STEAM

La placa **Imagina TDR STEAM** es una placa didáctica desarrollada por el equipo **ROBOLOT TEAM** que presenta la gran ventaja de tener una gran cantidad de sensores, actuadores y conexiones de expansión incorporados directamente en ella. Únicamente hay que conectar esta placa (*shield*) a una placa Arduino UNO (en nuestro caso, una placa compatible llamada Keyestudio UNO) y ya está todo listo para empezar a programar.

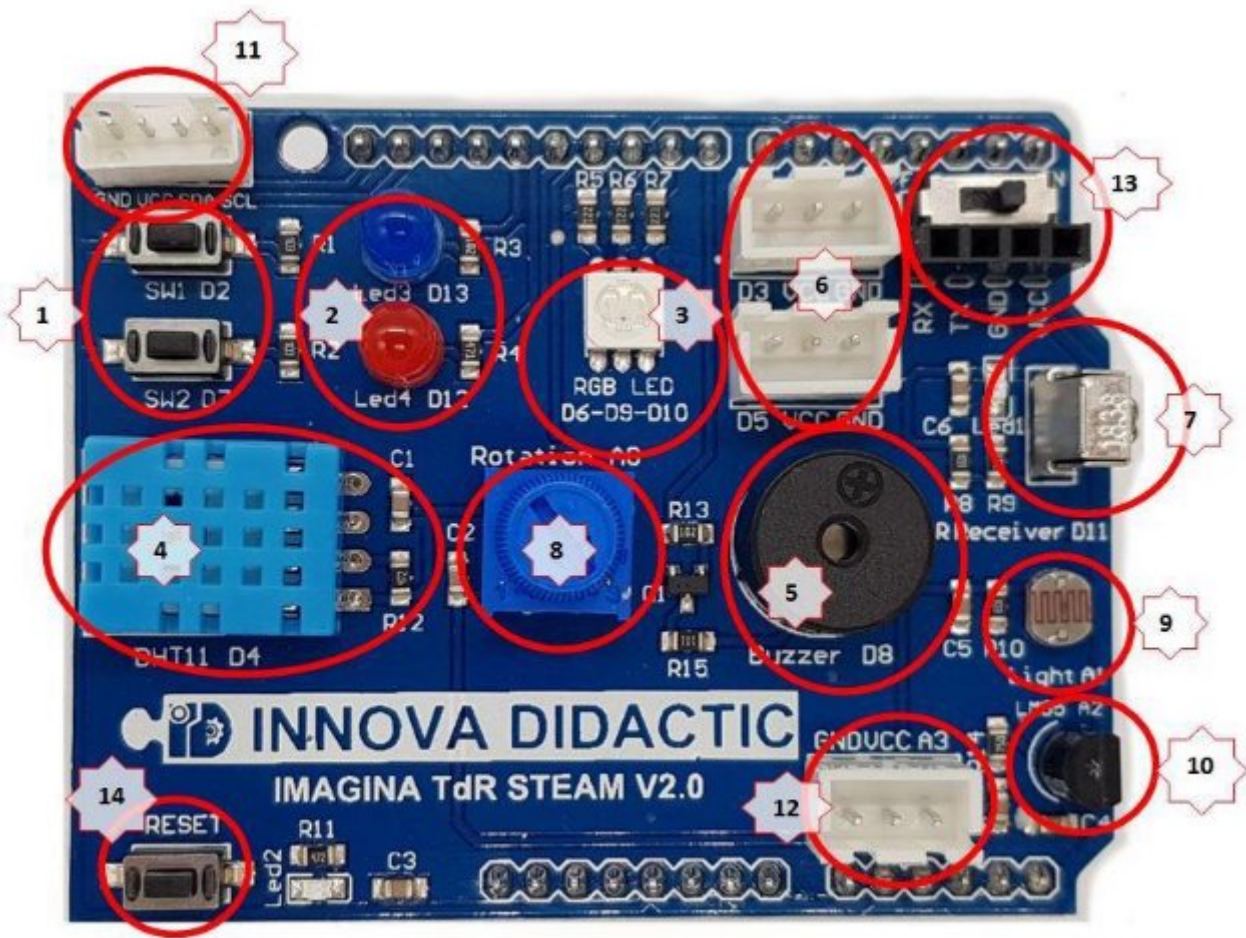
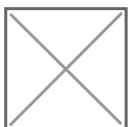
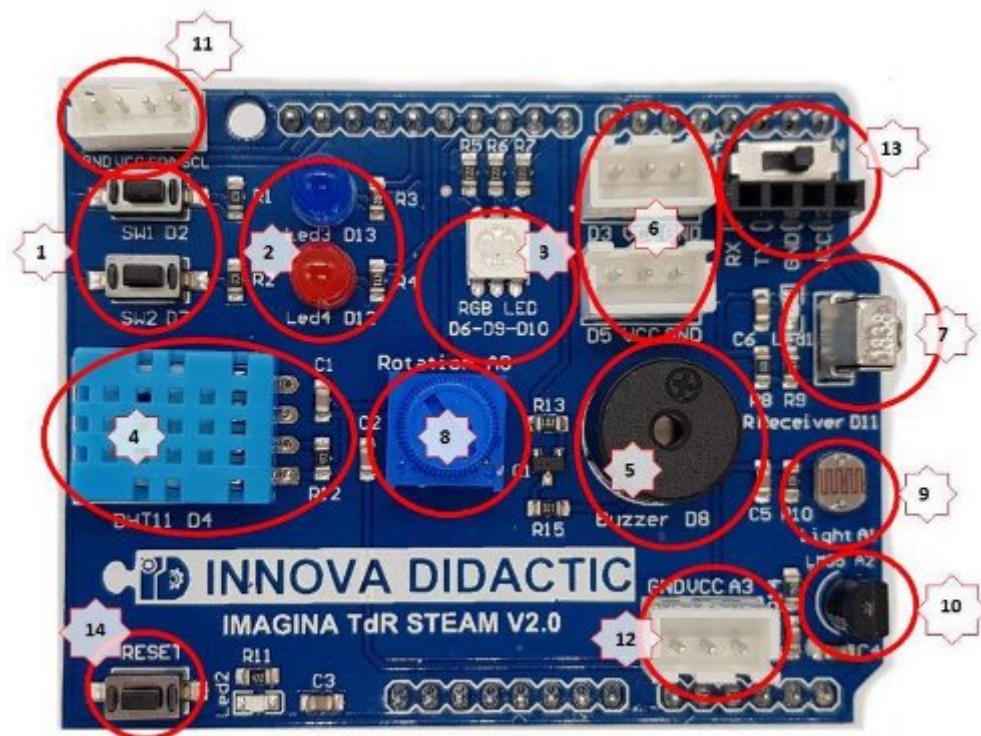


Tabla con la relación de elementos que hay en la placa Imagina TDR STEAM y sus conexiones:



	Sensor/ Actuador/ Módulo	Pin de conexión
1	Dos pulsadores (SW1, SW2)	D2 y D7

	Sensor/ Actuador/ Módulo	Pin de conexión
2	Dos leds (Azul Led3 y Rojo Led4)	D13 y D12
3	Led RGB	D6-D9-D10
4	Módulo DHT11 Sensor de Temperatura y Humedad	D4
5	Zumbador o Piezoeléctrico	D8
6	Dos puertos (Entradas/Salidas) digitales	D3 y D5
7	Módulo receptor de infrarrojos (IR)	D11
8	Módulo potenciómetro giratorio	A0
9	Sensor de luminosidad (LDR)	A1
10	Sensor de temperatura (LM35)	A2
11	Interface I2C compatible con sensores y módulos Keyestudio	SDA-A4 SCL-A5
12	Puerto Entrada Analógico	A3
13	Conexión de comunicaciones Bluetooth y Wifi (Switch On/Off)	Rx - Tx
14	Botón de reinicio.	-

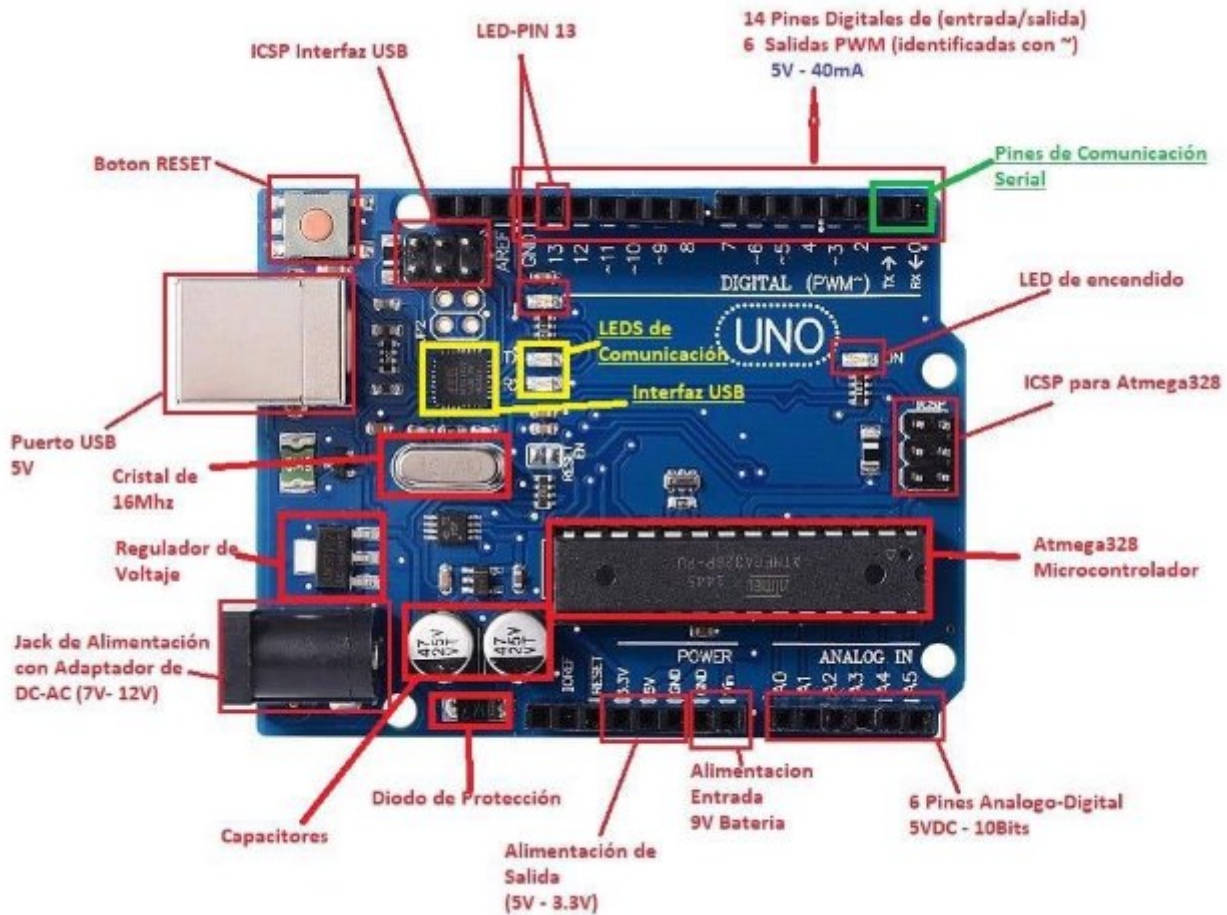


4. Placa de control

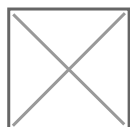
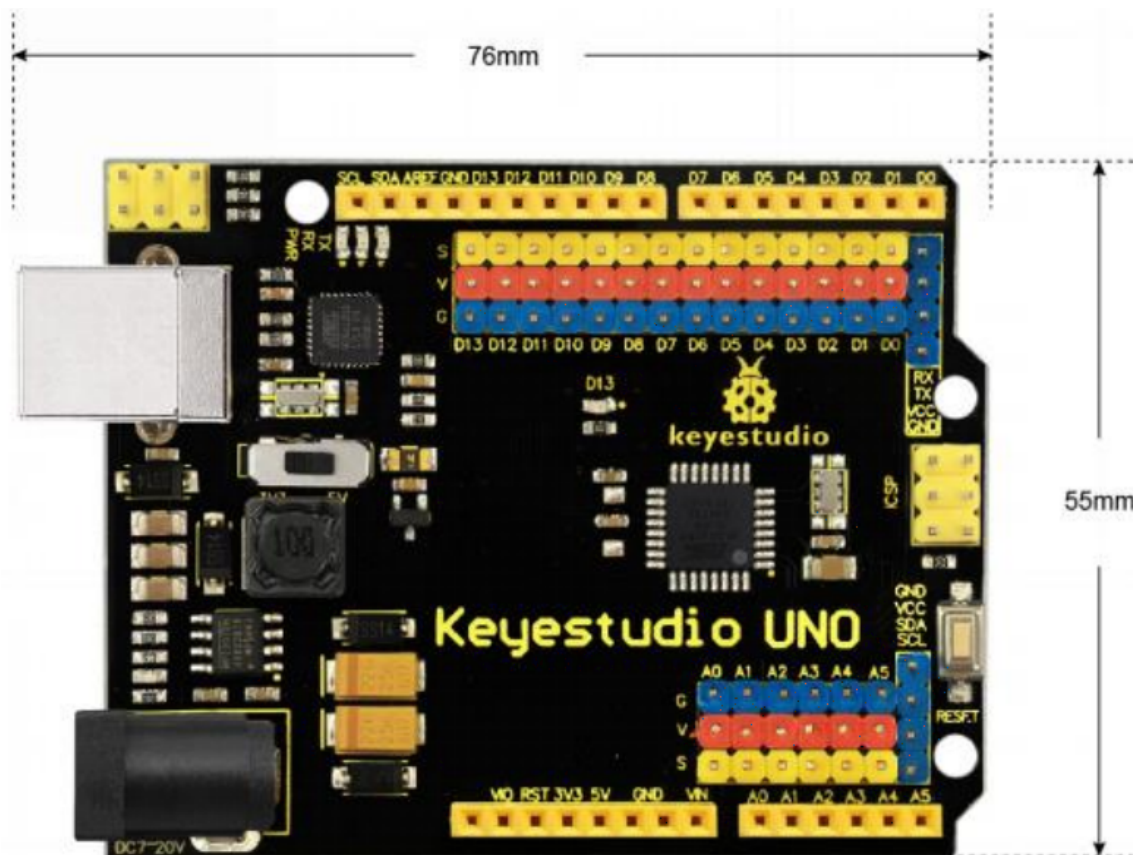
Keyestudio UNO (compatible con Arduino UNO)

La placa de control que gestiona la placa Imagina TDR STEAM está basada en una placa **Arduino UNO**. Arduino es una plataforma de prototipos electrónicos de código abierto (Open-Source) basada en hardware y software libre, flexible y fácil de usar, con una comunidad muy grande que genera muchísima información. Esta plataforma permite crear diferentes tipos de sistema para diferentes usos. Puede ser empleada por *artistas, diseñadores*, ingenieros, profesores, alumnos, etc., y en general, cualquier persona que esté interesada en crear objetos o entornos interactivos, prototipos, sistemas robóticos, etc.

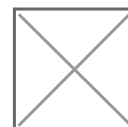




Al ser hardware libre existen multitud de fabricantes que han desarrollado versiones basadas en Arduino. Uno de esos fabricantes es **Keyestudio**.



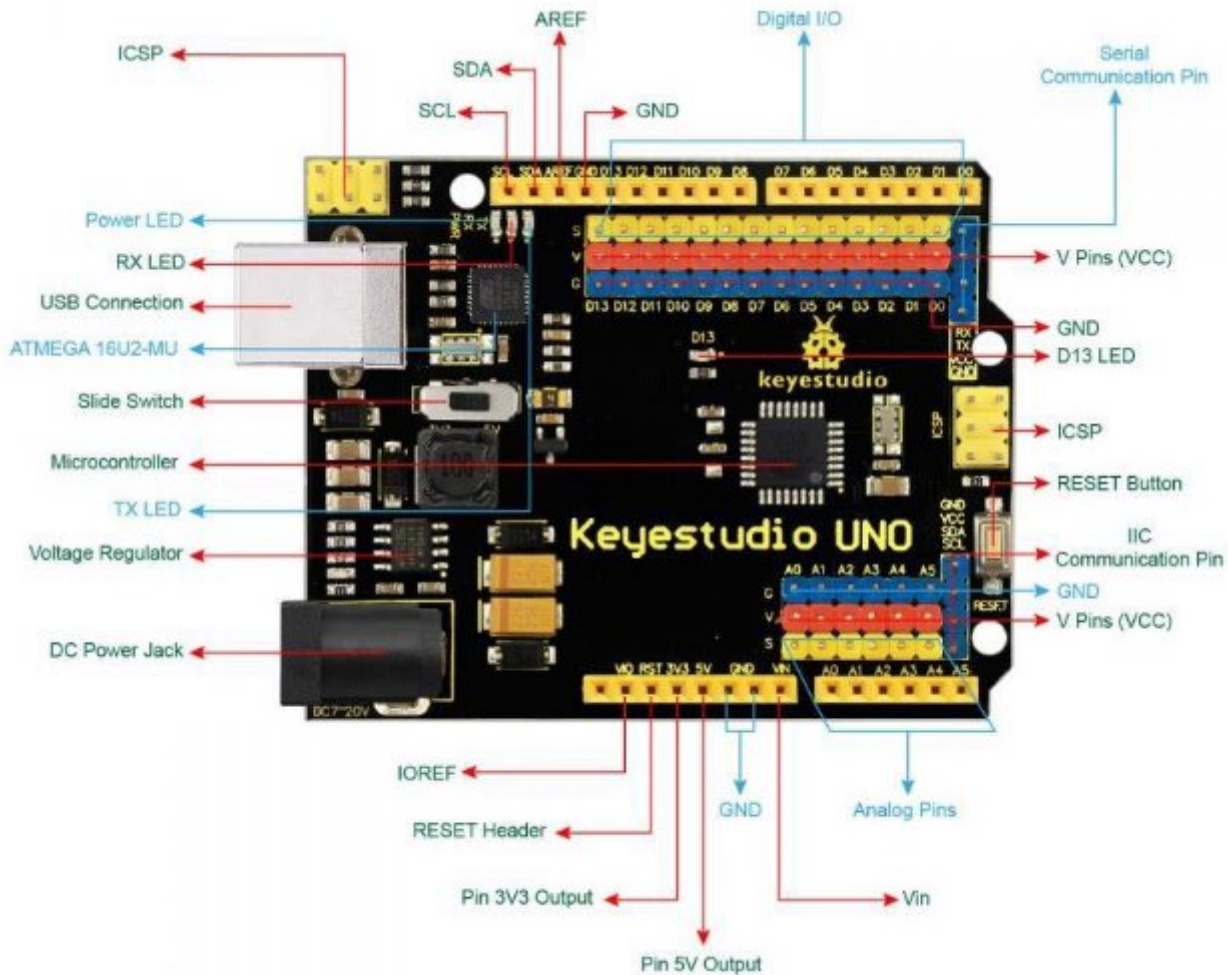
La placa **Keyestudio UNO** lleva un microcontrolador que está basado en el



ATmega328. Tiene 14 pines digitales de entrada / salida (de los cuales 6 se pueden usar como salidas PWM), 6 entradas analógicas, un cristal de cuarzo de 16 MHz, una conexión USB, un

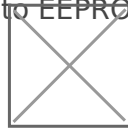
conector de alimentación, un conector ICSP y un botón de reinicio (reset).





Especificaciones:

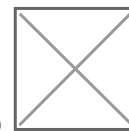
- Voltaje de funcionamiento: + 5V.
- Voltaje de entrada externo: + 7V ~ + 12V. (Límite: +6 V. <+ 20 V).
- Corriente de interfaz DCI / O: 20mA.
- Flash Memory: 32 KB (ATmega328) de los cuales 0,5 KB utilizados por el gestor de arranque.
- Capacidad de almacenamiento EEPROM: 1KB.




- Frecuencia del reloj: 16MHZ.
- Microcontrolador ATmega328P-PU
- Pines de E / S digital 14 (de los cuales 6 proporcionan salida PWM).
- Pines de E / S digitales PWM 6 (D3, D5, D6, D9, D10, D11).
- Pines de entrada analógica 6 (A0-A5).
- LED_BUILTIN D13.

Las conexiones de la placa TDR STEAM con la placa Keyestudio UNO son las siguientes:

Conexión	Descripción	Tipo
D0	Rx	Comunicación PC/Bluetooth/Wifi
D1	Tx	
D2	Pulsador SW1	Entrada digital
D3	Libre	E/S digital
D4	DHT11 (sensor de humedad y temperatura)	Entrada digital
D5	Libre	E/S digital
D6	RGB (rojo)	Salida digital
D7	Pulsador SW2	Entrada digital
D8	Zumbador	Salida digital
D9	RGB (verde)	Salida digital
D10	RGB (azul)	Salida digital
D11	Sensor infrarrojos	Entrada digital
D12	Led rojo	Salida digital
D13	Led azul	Salida digital
A0	Potenciómetro	Entrada analógica
A1	LDR (sensor de luz)	Entrada analógica
A2	LM35 (sensor de temperatura)	Entrada analógica
A3	Libre	Entrada analógica
A4	SDA	I2C
A5	SCL	



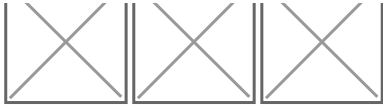
Para realizar la programación la podemos hacer mediante la IDE de Arduino  o mediante ArduinoBlocks. Como podemos ver son dos sistemas diferentes. En la IDE de Arduino la programación se realiza mediante instrucciones (derecha de la imagen) y en ArduinoBlocks se realiza mediante bloques (izquierda de la imagen). En la siguiente imagen se hace una comparación de código entre ArduinoBlocks y Arduino IDE.

The image shows a side-by-side comparison of Arduino programming in two environments: ArduinoBlocks (left) and Arduino IDE (right).

ArduinoBlocks (Left): The interface shows a workspace with a grid of blocks. A red box highlights the 'Inicializar' (Initialize) block. A blue box highlights a 'Bucle' (Loop) block containing several sub-blocks: two 'Led' blocks (Pin 13, Estado ON and OFF) and two 'Esperar' (Wait) blocks (500 milisegundos). A yellow box highlights the 'Escribir digital Pin 13' (Write digital Pin 13) block, which is set to 'ON'.

Arduino IDE (Right): The interface shows the C++ code for the same sketch. A red box highlights the 'void setup()' function. A blue box highlights the 'void loop()' function. A yellow box highlights the 'digitalWrite(13, HIGH);' line, which corresponds to the 'ON' state in the ArduinoBlocks block. Other lines of code include 'pinMode(13, OUTPUT);', 'delay(500);', and 'digitalWrite(13, LOW);'.

Arrows indicate the mapping between the visual blocks in ArduinoBlocks and the corresponding code lines in the Arduino IDE.

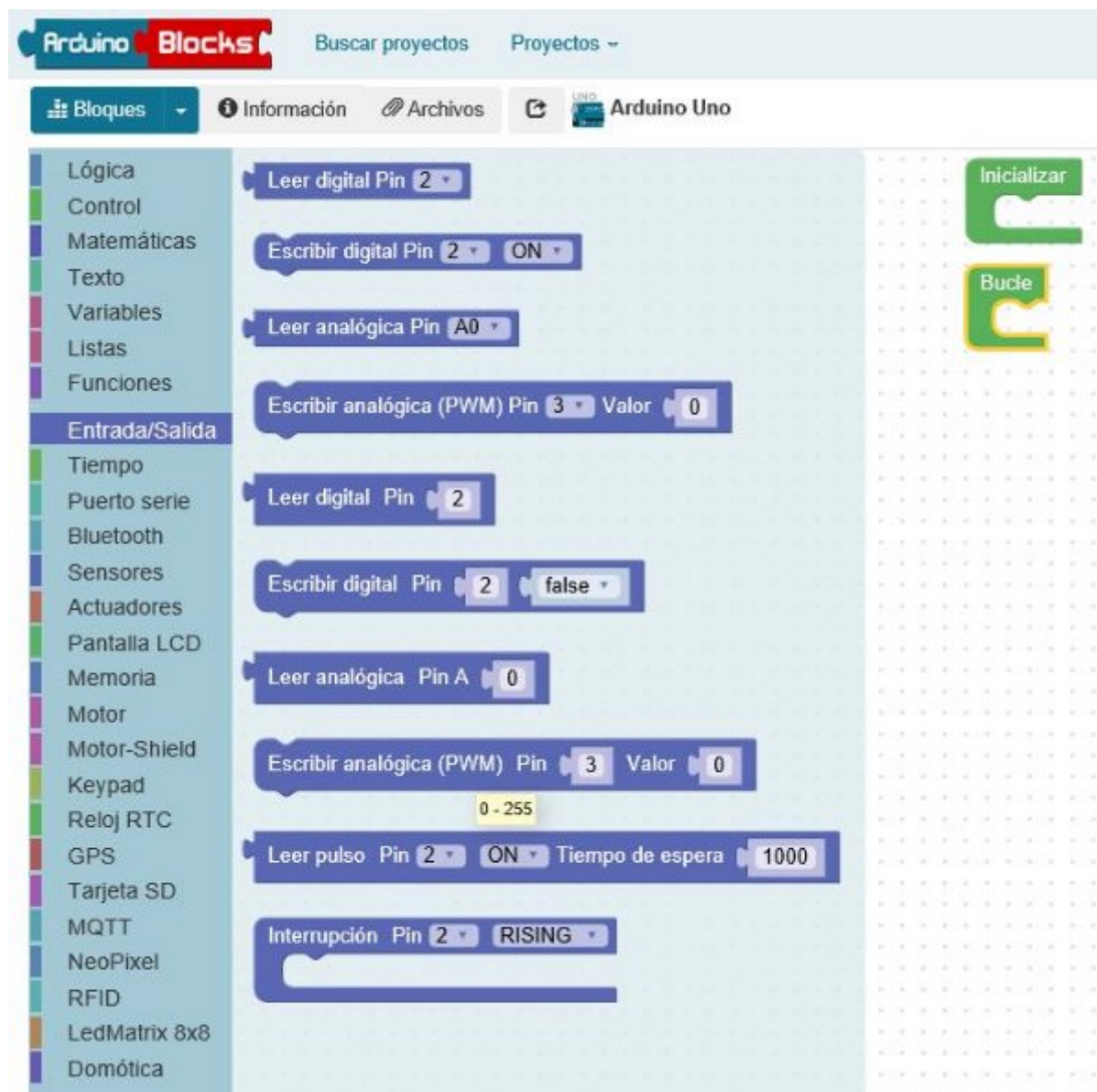


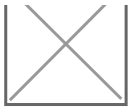
Utilizar ArduinoBlocks simplifica y hace más inteligible el código, lo que permite iniciarse en el mundo de la programación de un modo más amigable. ArduinoBlocks también permite programar de diversas formas, ya que tiene bloques específicos que realizan las mismas funciones pero que se pueden entender de forma más sencilla.

5. Programación con ArduinoBlocks

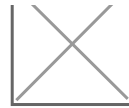
ArduinoBlocks es un lenguaje de programación gráfico por “Bloques” creado por el profesor Juanjo López. Está pensado para que niñas y niños aprendan a programar con placas Arduino a partir de unos 8-9 años.

Los distintos bloques sirven para leer y escribir las distintas entradas y salidas de la placa, así como programar funciones lógicas, de control, etc.





En esta guía usaremos ArduinoBlocks dedicado al uso de la placa Imagina TDR

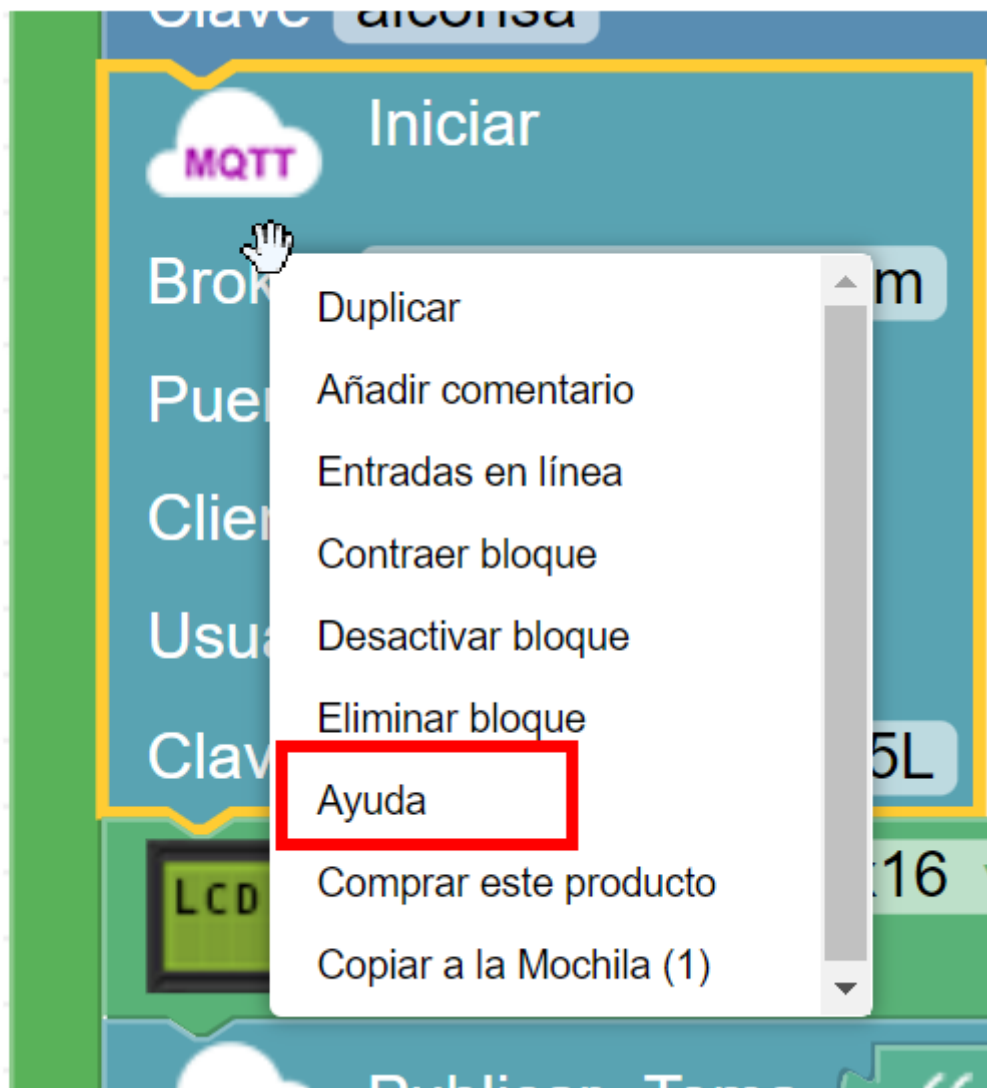


STEAM, con estos bloques podremos programar las entradas y salidas de nuestra placa para que realice las tareas que queramos.

Podemos programar ArduinoBlocks de diferentes maneras ya que tiene múltiples bloques. También permite exportar el código para la IDE de Arduino. Pero para este manual utilizaremos, principalmente, la programación que se muestra en medio (bloques específicos para Imagina TDR STEAM), ya que es más fácil de entender.

The screenshot displays the ArduinoBlocks web interface. At the top, there's a header with the 'ArduinoBlocks' logo, a search bar labeled 'Buscar proyectos', and a dropdown menu for 'Proyectos'. Below this is a navigation bar with 'Bloques', 'Información', 'Archivos', and a project-specific button 'Reto A01'. The main workspace is divided into a left sidebar with a category list and a central area with a sequence of blocks. The sidebar categories include Matemáticas, Texto, Variables, Listas, Funciones, Tiempo, Puerto serie, Bluetooth, Sensores, Actuadores, Motor, Pantalla LCD, Pantalla OLED, LedMatrix 8x8, Memoria, MQTT (IoT), Blynk (IoT), NeoPixel, and a highlighted 'TDR STEAM' section. The central workspace contains the following blocks in sequence: 1. 'Pulsador 1 pulsado' (Button 1 pressed) with a button icon. 2. 'Led Rojo Estado ON' (Red LED State ON) with a red LED icon. 3. 'Led RGB Color' (RGB LED Color) with a color wheel icon. 4. 'Led RGB R 0 G 0 B 0' (RGB LED R 0 G 0 B 0) with a color wheel icon and three input fields set to 0. 5. 'Potenciómetro %' (Potentiometer %) with a potentiometer icon. 6. 'Nivel de luz (LDR) %' (Light level (LDR) %) with an LDR sensor icon. The bottom of the interface has a footer with 'ArduinoBlocks - by Juanjo Lopez', 'Aviso legal / Política privacidad', and 'Tienda - Innovadidactic'.

Si queremos ayuda de cada bloque, podemos pulsar con el botón derecho en Ayuda



y entramos en el libro online y buscamos el bloque que queramos

FreeBook

[Free Book \(online & updated\).](#)

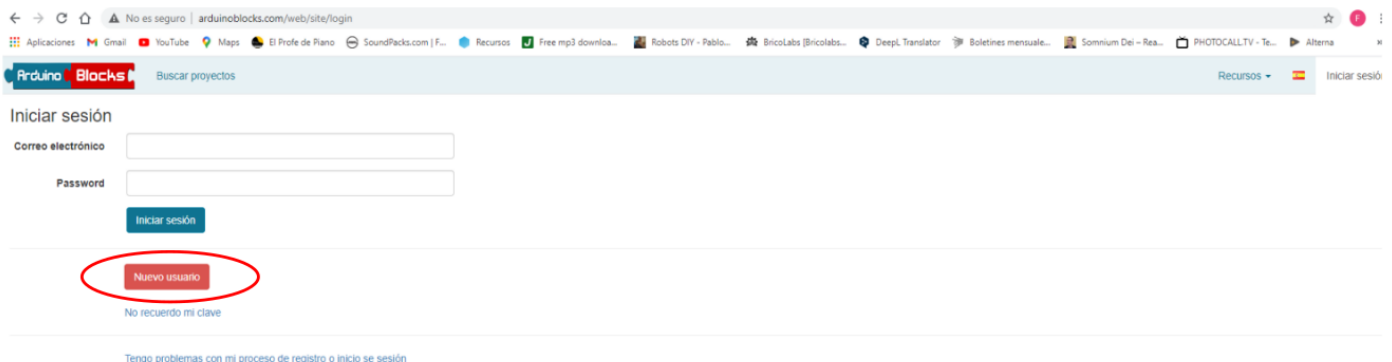
Libros & Documentación

6. Instalación de ArduinoBlocks

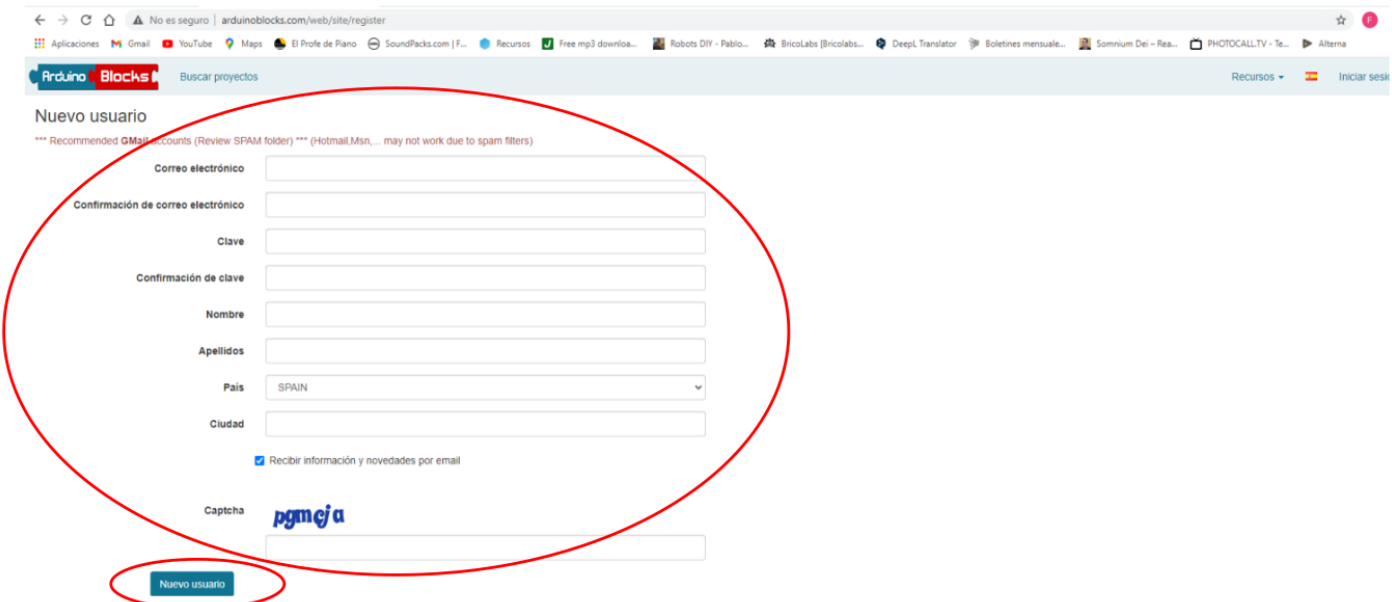
ArduinoBlocks trabaja on-line pero tenemos que instalar un pequeño programa que será el encargado de conectar nuestro programa con la placa Keyestudio UNO. Este programa basado en Python se llama *Connector*.

Primero deberemos crear una cuenta en ArduinoBlocks y después instalar el software **Connector**.

- Para crear una **cuenta** en **ArduinoBlocks**:



The screenshot shows the login page of ArduinoBlocks. At the top, there is a navigation bar with the ArduinoBlocks logo, a search bar, and links for 'Recursos' and 'Iniciar sesión'. Below the navigation bar, the 'Iniciar sesión' section contains two input fields for 'Correo electrónico' and 'Password', followed by an 'Iniciar sesión' button. A red circle highlights a 'Nuevo usuario' button located below the login fields. Below this button, there is a link 'No recuerdo mi clave' and a footer link 'Tengo problemas con mi proceso de registro o inicio de sesión'.



The screenshot shows the registration page of ArduinoBlocks. At the top, there is a navigation bar with the ArduinoBlocks logo, a search bar, and links for 'Recursos' and 'Iniciar sesión'. Below the navigation bar, the 'Nuevo usuario' section contains several input fields: 'Correo electrónico', 'Confirmación de correo electrónico', 'Clave', 'Confirmación de clave', 'Nombre', 'Apellidos', 'País' (a dropdown menu with 'SPAIN' selected), and 'Ciudad'. There is a checkbox for 'Recibir información y novedades por email' which is checked. Below these fields is a 'Captcha' section with a 'pymeja' logo and a text input field. A red circle highlights the entire registration form area. At the bottom, there is a 'Nuevo usuario' button, also highlighted with a red circle.

- Para instalar **Connector**:



← → ↻ 🔒 No es seguro | arduinoblocks.com/web/project/editor/333508#

Aplicaciones Gmail YouTube Maps El Profe de Piano SoundPacks.com | F... Recursos Free mp3 downloa... Robots DIY - Pablo... Bricolabs [Bricolabs... DeepL Translator Boletines mensuale... Somnium Dei - Rea... PHC

Arduino Blocks Buscar proyectos Proyectos

Bloques Información Proyecto A01: El LED [Robotol Arduinoblocks]

Lógica Control Matemáticas Texto Variables Listas Funciones Entrada/Salida Tiempo Puerto serie Bluetooth Sensores Actuadores

Inicializar Bucle

ArduinoBlocks-Connector v4

La aplicación que conecta ArduinoBlocks a tu placa Arduino!

Arduino Blocks CONNECTOR

Windows Ubuntu 32 bits Ubuntu 64 bits MacOS RaspberryPi

Windows

Probado en XP, 7, 10 (32/64 bits)

Windows

Descarga para Windows (.exe)

¡Desactiva el antivirus si la descarga falla!



```
ArduinoBlocks-Connector
```

```
ArduinoBlocks
connector
>> ArduinoBlocks-Connector v4
>> by Juanjo Lopez
>> www.arduinoblocks.com
>> Listening on port 9987
>> (Ctrl+C to finish)
>> Ready...
Checking libraries...
Libraries version: 28
```