
Tabla de contenido

0 Inicio	1.1
----------	-----

1 Introducción

1.1 Pensamiento computacional	2.1
1.2 Material	2.2
1.2.1 NodeMCU	2.2.1
1.2.2. Shield motor	2.2.2
1.2.3 Chasis	2.2.3
1.2.4 DHT22	2.2.4
1.2.5 Hall	2.2.5
1.2.6 LDR	2.2.6
1.3 Conexiones	2.3

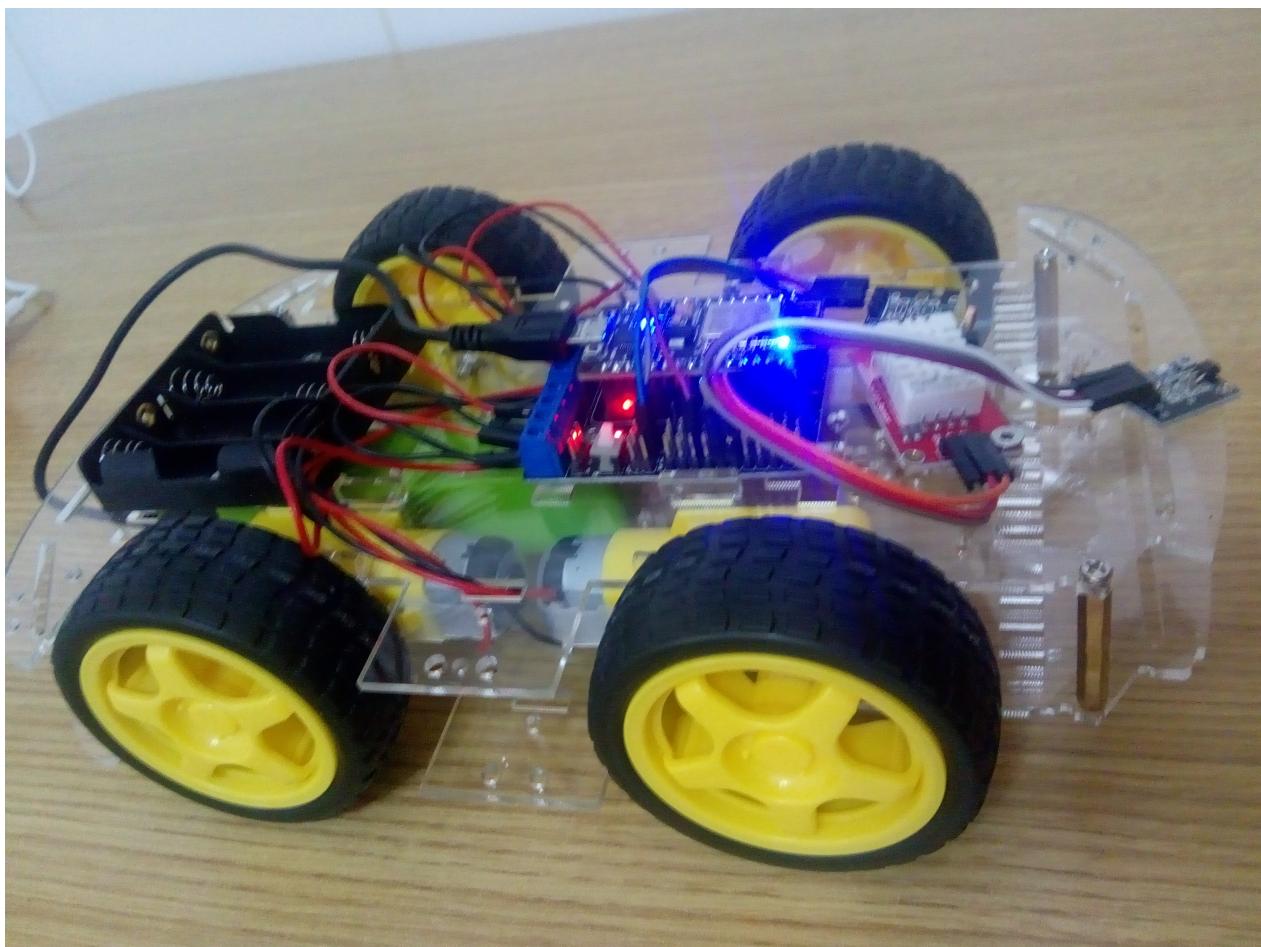
2 ARDUINOBLOCKS

2.1 Introducción	3.1
2.2 Cuentas alumnos	3.2
2.3 Driver	3.3
2.4 Empezando	3.4
2.5 Intermitente	3.5
2.6 Sensores	3.6
2.7 Motores	3.7

3 BLYNK

3.1 Qué es	4.1
3.2 Crear cuenta	4.2
3.3 Endender LED	4.3
3.3.1 Token	4.3.1
Créditos	5.1

ROVER MARCIANO CON NodeMCU ArduinoBlocks y BLYNK

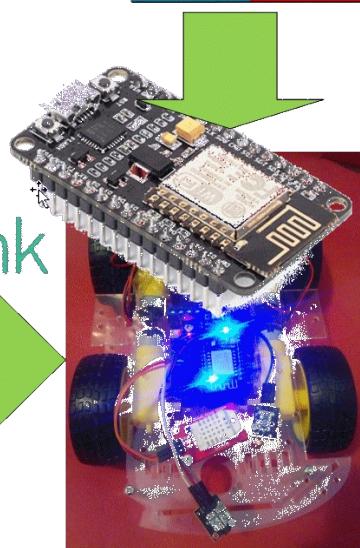


En este curso trataremos de manejar el rover propuesto utilizando las siguientes herramientas:

- Programación por bloques [ArduinoBlocks](#) para controlar la placa **NodeMCU**.
- Manejo de la [App Blink](#) para controlar el rover por wifi y con el móvil.
- AVANZADO
 - Creación de un **servidor local BLINK con Raspberry**

Conectaremos nuestro rover con una App que nos permitirá poner diferentes elementos de controlar

Arduino  Blocks



Pensamiento computacional

¿Dónde se encaja este robot? ¿se puede comparar este robot con otros robots de otros cursos que hacemos desde CATEDU?

Esta es la hoja de ruta que proponemos, no se tiene que tomar al pie de la letra, pero intenta ayudar al profesorado que tenga una visión global de tanta oferta:

Como se puede ver ROVER CON ARDUINO tiene la ventaja de tener un precio razonable, y dentro del rango de programación en bloques del Arduino en primaria.

RoboTICa

Oferta de formación en Pensamiento computacional del Centro Aragonés de Tecnologías para la Educación.



MATERIAL

El precio de este kit es aproximadamente 50€

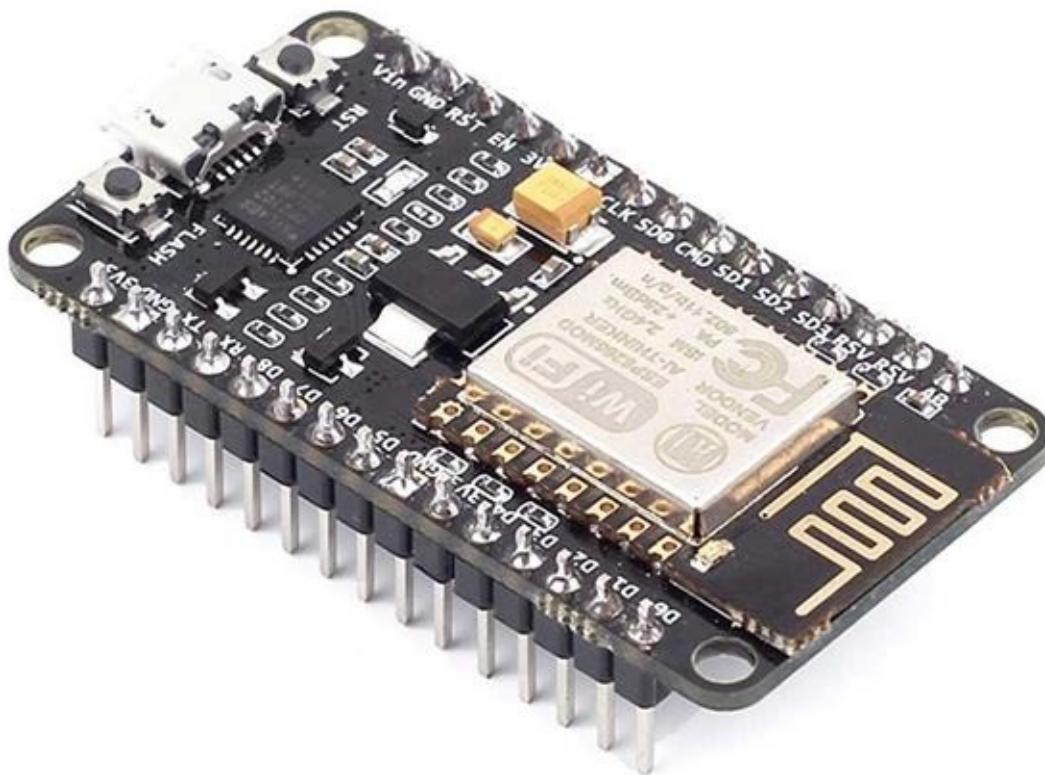
Lista de material : Hoja 1

Hoja 1

>

<

NodeMCU



Es una placa de desarrollo basado en el SoC (System on a chip) ESP8266 e incorpora la comunicación WiFi que tanto falta en el Arduino.

Desde que salió el ESP01 como primer módulo del ESP8266, se ha evolucionado mucho, mejorando la popularidad, potencialidad y bajando los precios. Actualmente (2020) se fabrica el ESP32.

Nosotros utilizamos en el Rover el NodeMCU basado en el ESP12E conocida como **NodeMCU** que apareció en el 2014 que se programaba con [Lua](#).

Con el paso del tiempo esta placa se programa en varios lenguajes, siendo la más interesante desde el punto pedagógico que se pueda programar en el didáctico **entorno Arduino**.

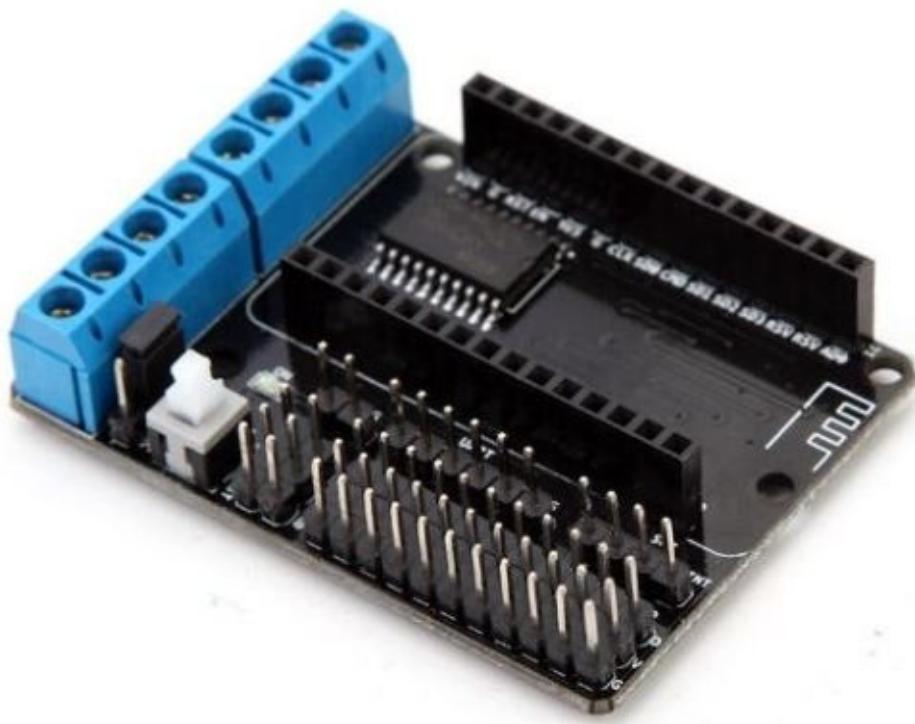
Existen disparidad en los fabricantes, en este curso se va a utilizar la generación segunda de 8 pines

Generación	Versión		Módulo	Anchura	Comentario
Primera	0.9		ESP12	10 pines	La original, no está a la venta
Segunda	1.0	V2	ESP12E	8 pines	Versión "oficial"
Tercera	1.0	V3	ESP12E	10 pines	Versión de Wemos/Lolin. Conversor CH340G

De [Luis Llamas](#) CC-BY-NC-SA

Para más información recomendamos la página de Luis Llamas (<https://www.luisllamas.es/esp8266-nodemcu/>)

Shield motor para NodeMCU



Esta shield permite conectar motores en los 4 primeros terminales azules con el NodeMCU que se coloca en el zócalo. Para saber cómo funcionan los motores con esta shield [ver aquí](#)

También tiene unos machos para las entradas y salidas digitales del NodeMCU y uno para la entrada analógica. Esto lo veremos en [conexiones](#).

chasis



Se elije este chasis por su bajo coste y con tracción 4x4 (en marte no hay carreteras asfaltadas).

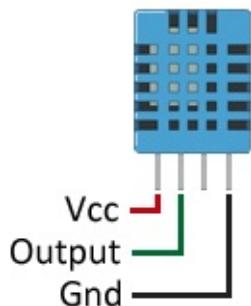
DHT22



Créditos: CC-BY-NC-SA [Luis Llamas](#)

Es un sensor que mide de forma sencilla tanto la temperatura como la humedad. Este sensor aparece en el [curso de Aularagón Arduino con código](#)

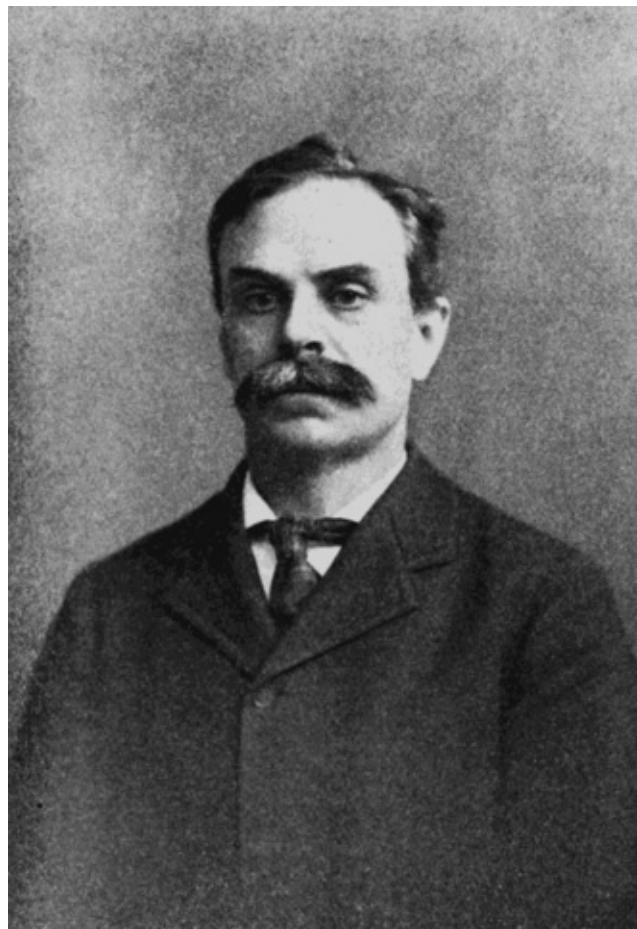
Su conexión es sencilla :



Créditos: CC-BY-NC-SA [Luis Llamas](#)

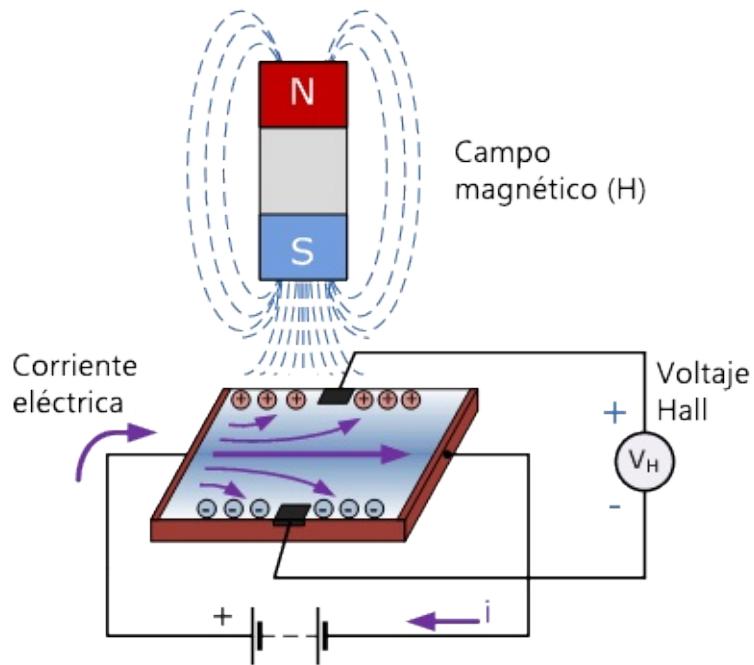
Para saber más recomendamos la página de [Luis Llamas](#)

Sensor efecto Hall



De Desconocido - Popular Science Monthly Volume 64, Dominio público

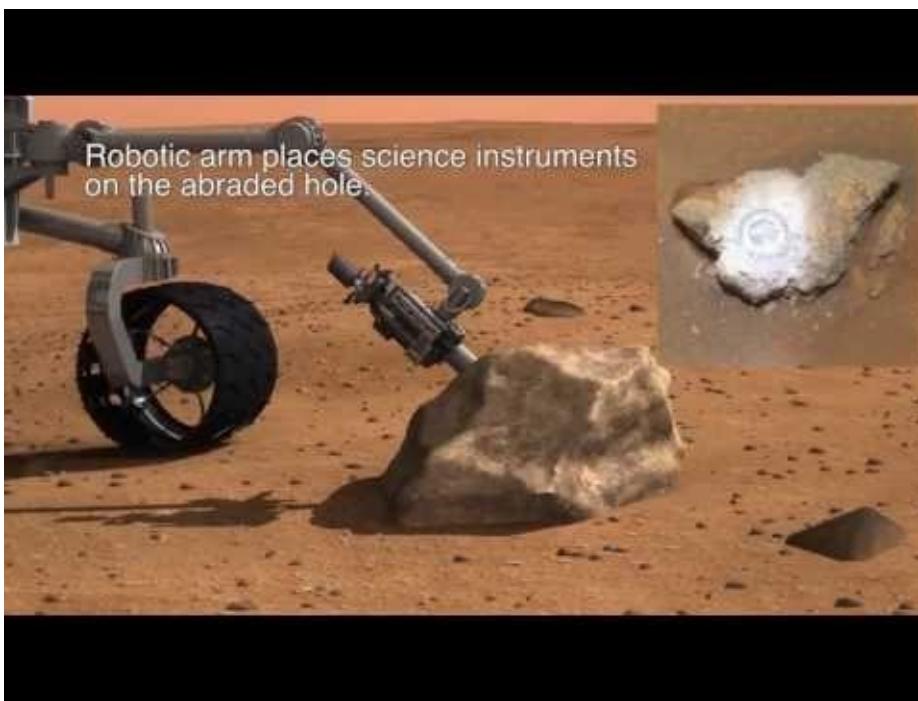
[Edwin Herbert Hall](#) descubrió en 1879 que en presencia de un campo magnético, un conductor que conduzca una corriente se le producía un campo eléctrico porque las cargas eléctricas se desviaban de su trayectoria principal, nuestro sensor simplemente mide ese campo eléctrico:



De [Luis Llamas CC-BY-NC](#)

El sensor tiene un led de color rojo que indica que hay una lectura de campo magnético.

Nuestro rover analizará si las rocas que hay cerca de él tienen ferritas midiendo su componente magnético. Una aproximación tosca de los rovers actuales pero sirve para acercar el mundo de la medición remota al alumnado :



[Video link](#)

Nuestro robot detectará si hay un imán cerca o no .

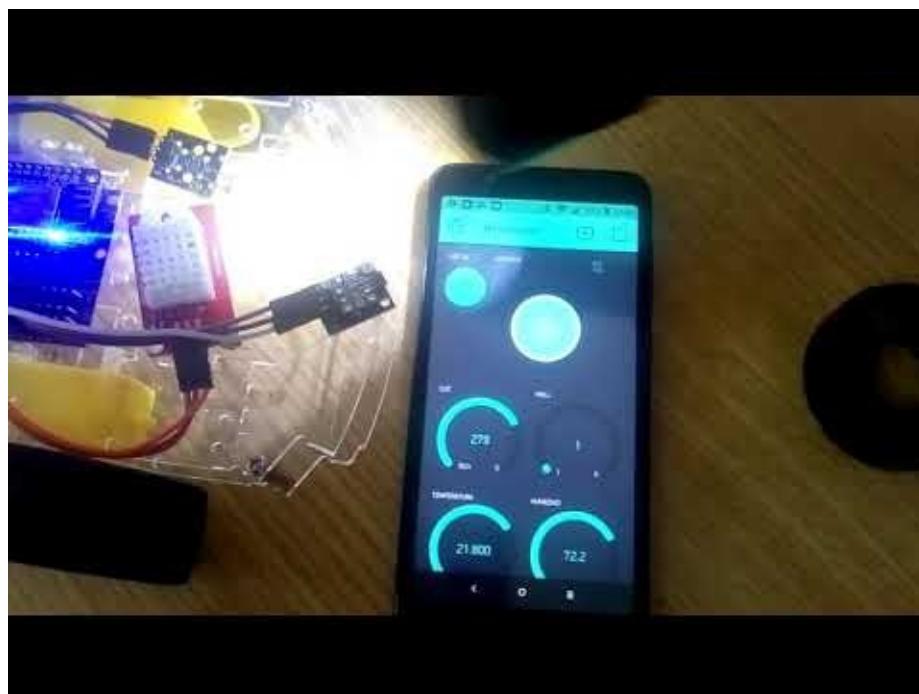


[Video link](#)

Sensor de luz LDR.



El LDR es una resistencia que varía con la luz. La conexión interna de este sensor analógico hace que **cuanto más luz menos valor nos proporciona** por lo tanto va al revés. Esto será importante para la configuración de nuestros programas, por ejemplo el medidor "Gauge" de la APP Blink de este vídeo va de 1023 a 0 para medir correctamente la cantidad de luz:

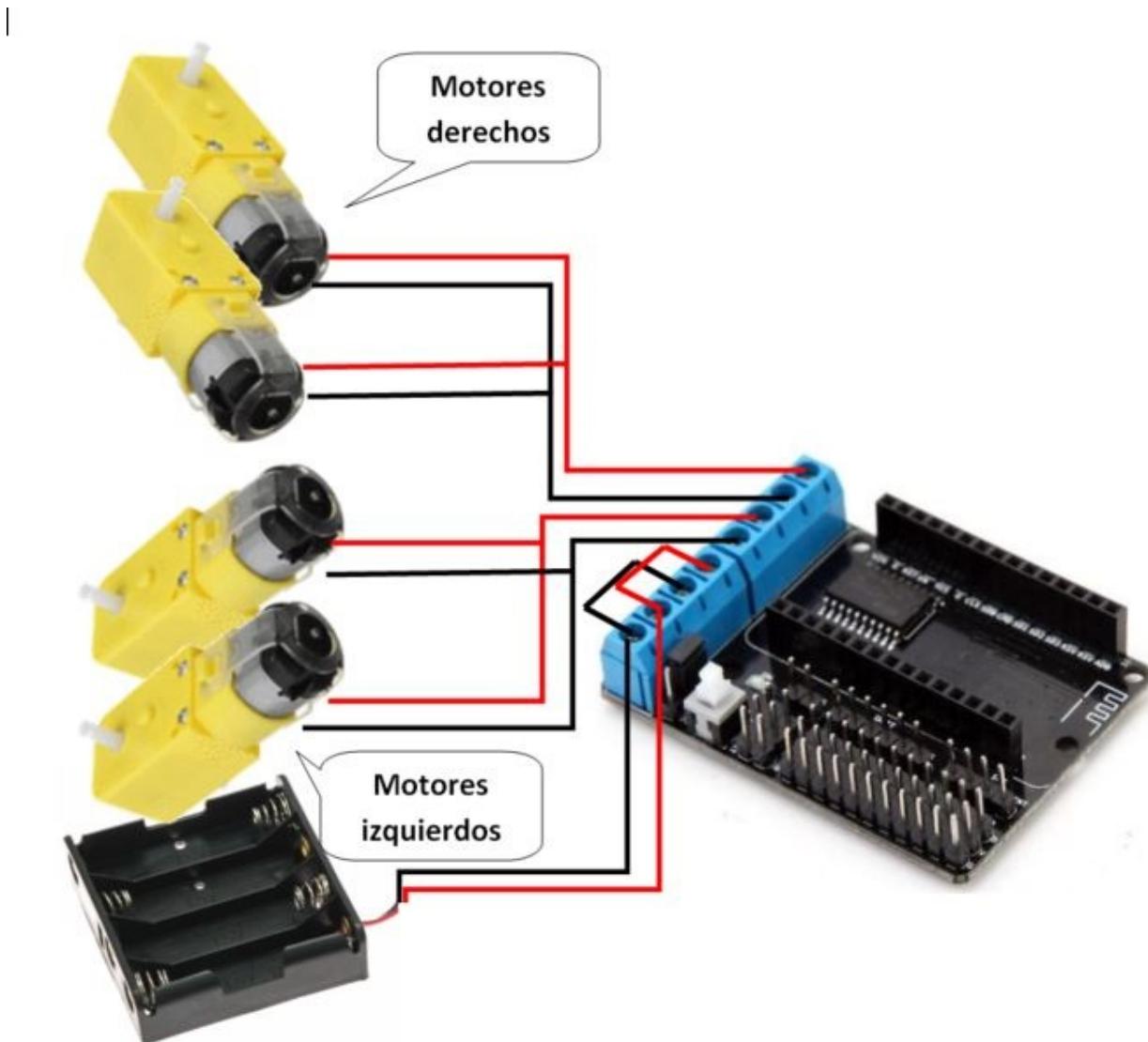


[Video link](#)

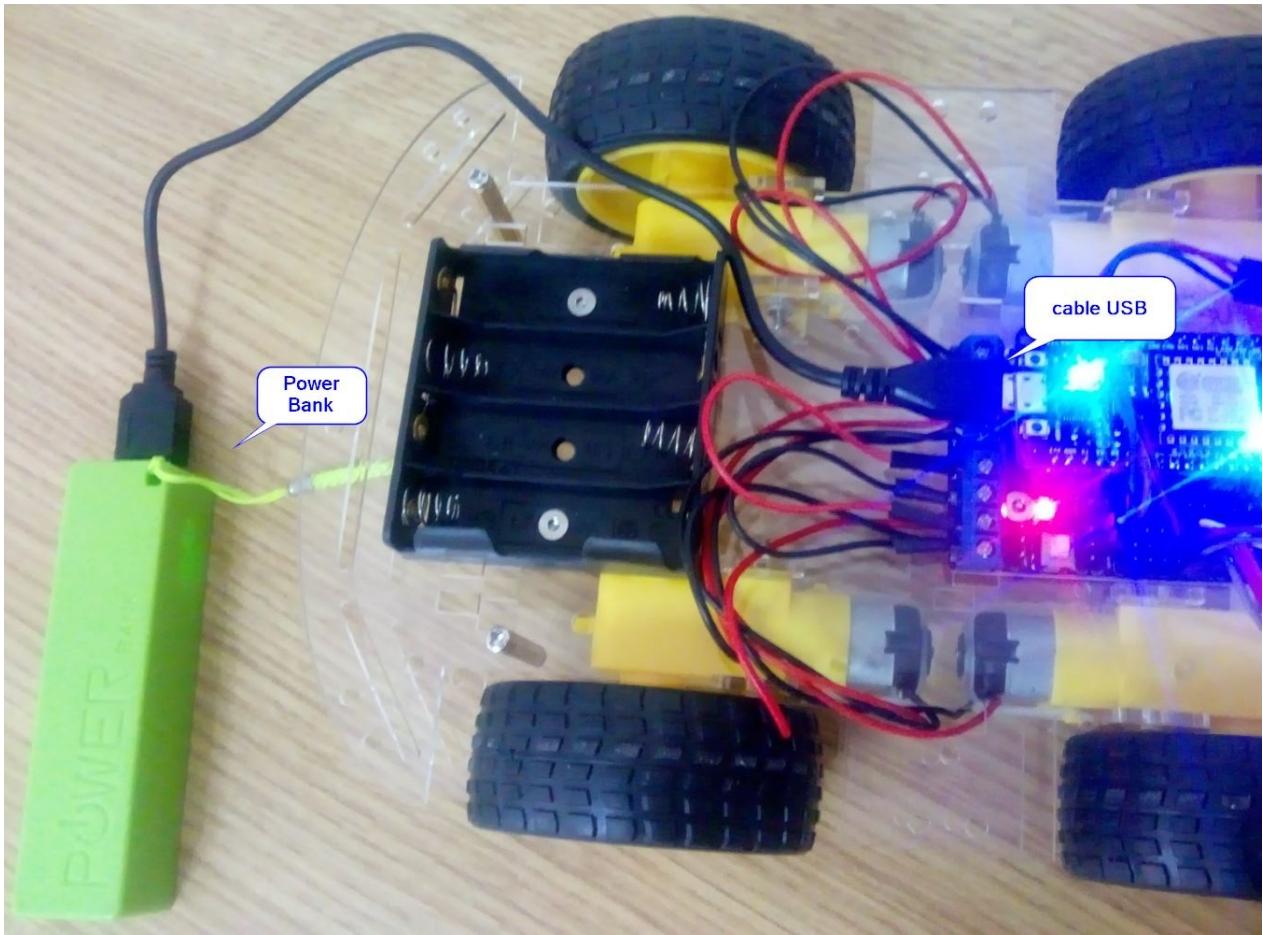
Conexiones.

Conexiones motores y batería

Este es el esquema de conexión de los motores y de la batería con el Shield :



Una alternativa a la conexión de la batería es utilizar un PowerBank y diréctamente al NodeMCU como podemos ver en la foto :



Otra posibilidad es utilizar baterías 18650 [ver](#) pero ojo con usar estas baterías [ver+](#)

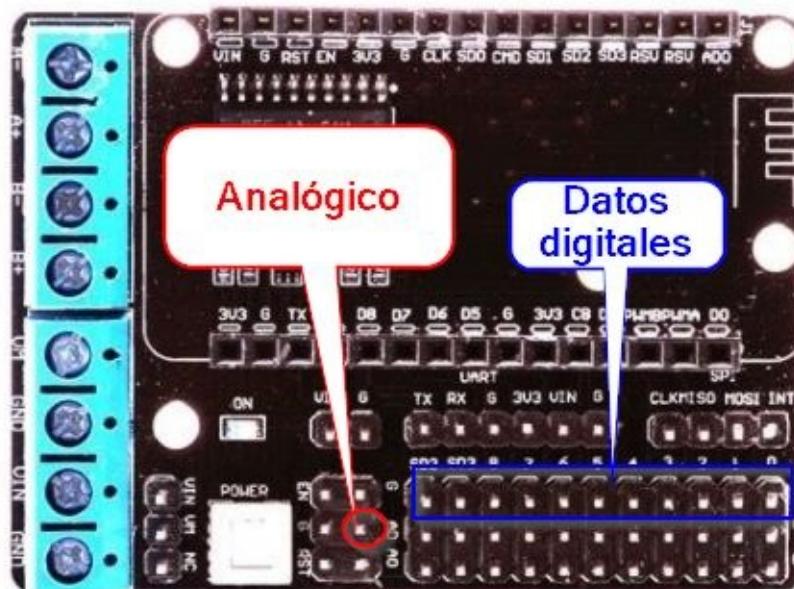
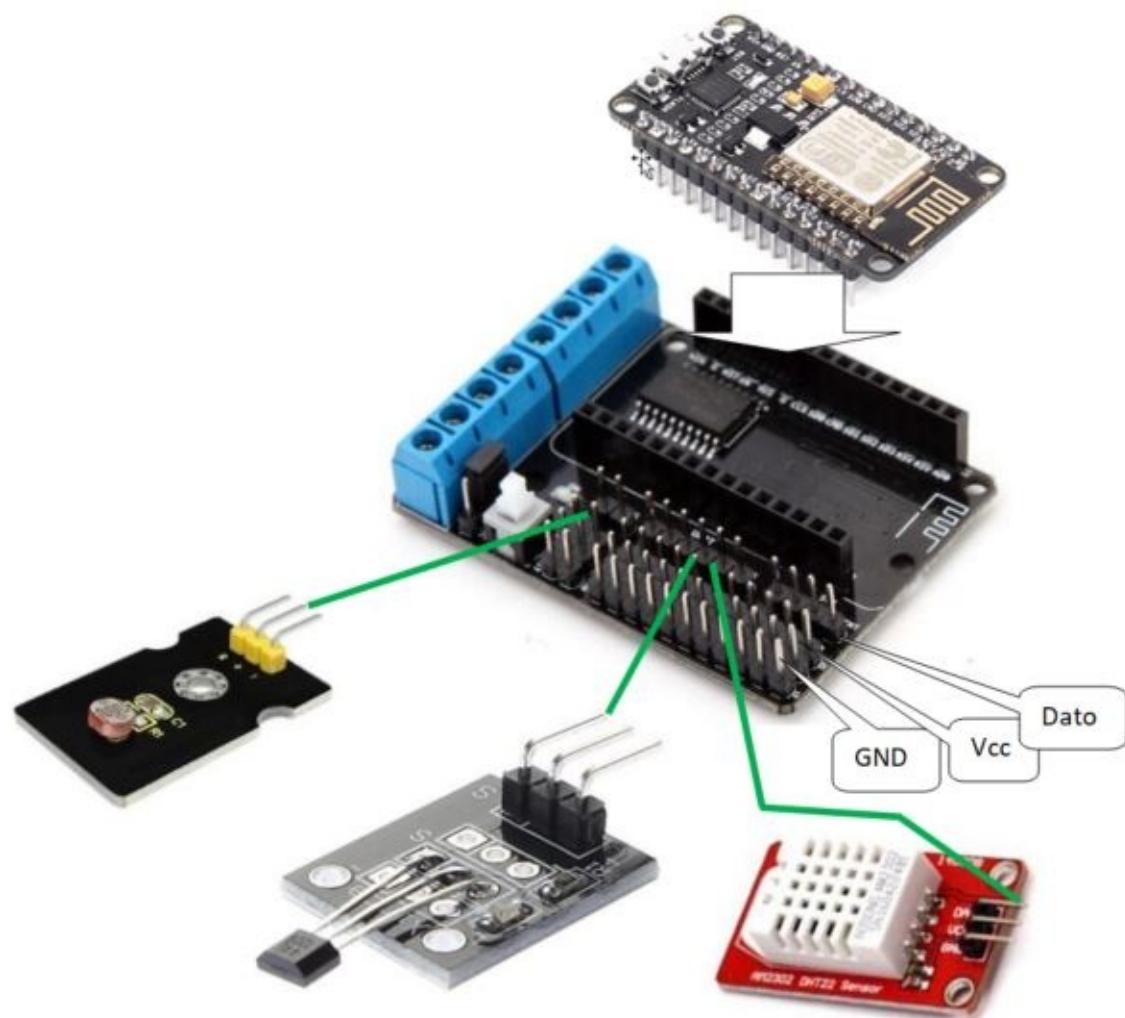
Conexiones NodeMCU sensores con la Shield.

El sensor de efecto Hall y el sensor DHT22 lo conectaremos a las entradas digitales, nosotros hemos elegido

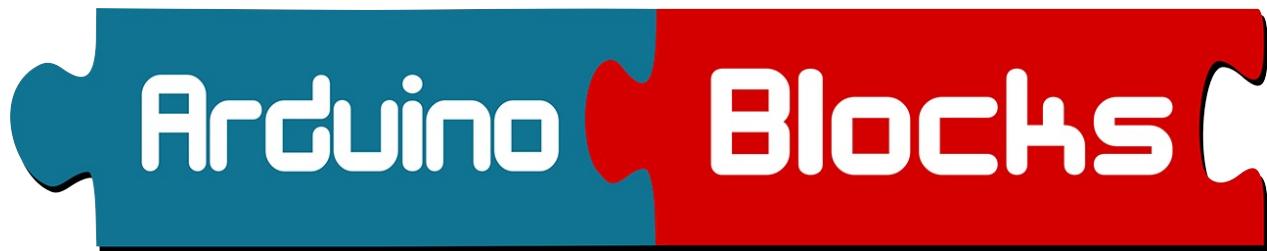
- DHT22 a D5
- Efecto Hall al D6

Es recomendable no usar D4 pues lo utilizaremos como Led

El sensor de luz LDR al ser analógico, lo conectaremos en la única entrada analógica que tiene esta shield



ARDUINOBLOCKS



Es un programa *online** creado por el profesor Juan José López Almendros:



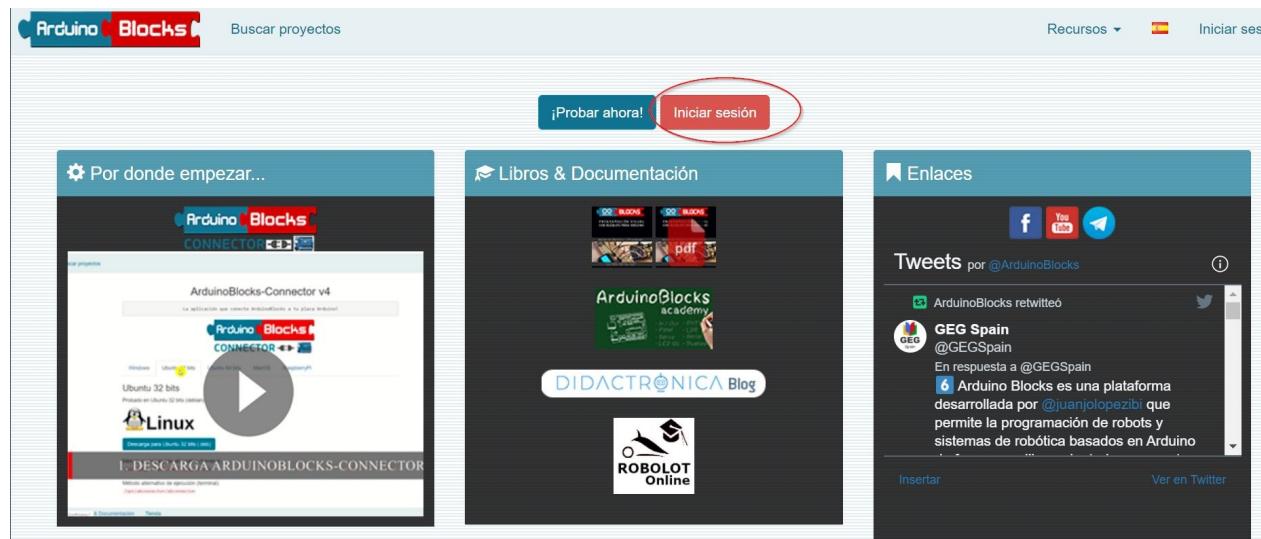
De [Juan José López Almendros](#) CC-BY-SA

Es una programación **gráfica** al estilo de Scratch y está pensado para utilizar a partir de niños y niñas de 8 años.

Crear cuenta

Entramos en <http://www.arduinoblocks.com/>

e iniciamos sesión



Y rellenamos el formulario

Correo electrónico	<input type="text"/>
Confirmación de correo electrónico	<input type="text"/>
Clave	<input type="text"/>
Confirmación de clave	<input type="text"/>
Nombre	<input type="text"/>
Apellidos	<input type="text"/>
País	SPAIN <input type="button" value="▼"/>
Ciudad	<input type="text"/>

Gestión de cuentas gestionadas para alumnos.

Tal y como dice el tutorial de Juanjo López :

Permite a un usuario registrado con email, crear y administrar nuevas cuentas de usuario dentro de una organización, centro educativo o institución.

Recomendamos seguir el tutorial de Juanjo López

https://github.com/arduinoblocks/documentacion/blob/master/usuarios_gestionados.pdf

Arduinoblocks conector.

Para poder usar la herramienta Arduinoblocks tenemos que ejecutar antes **Arduinoblocks conector**.



Lo instalamos y lo ejecutamos y aparece esta ventana :

The screenshot shows the 'ArduinoBlocks-Connector' application window. The title bar says 'ArduinoBlocks-Connector'. The main area is a terminal window displaying the following text:

```

>> ArduinoBlocks-Connector v4
>> by Juanjo Lopez
>> www.arduinoblocks.com
>> Listening on port 9987
>> (Ctrl+C to finish)
>> Ready...
Checking libraries...
Libraries version: 32

```

ATENCIÓN

No podemos cerrar la ventana mientras utilizamos Arduinoblocks, la minimizamos simplemente.

En caso contrario, Arduinoblocks no se puede comunicar con nuestro Arduino, NodeMCU, ...

Empezando un proyecto

Entramos en Proyectos y podemos ver nuestros proyectos creados como también empezar uno.

Cuando creas un proyecto, ten en cuenta que puedes hacer:

- Proyectos personales privados
- Proyectos personales públicos
- Proyectos para tu clase, donde lo compartes con tus alumnos y los puedes supervisar y corregir.

Lo primero que tenemos que elegir es para qué tipo de placa se hace el proyecto. En nuestro caso es para **NodeMCU**

Nuevo proyecto personal

Tipo de proyecto	
Nombre	Arduino Uno Arduino Nano / ATmega328 Arduino Nano / ATmega328 (new bootloader) Arduino Mega / 2560 Arduino Leonardo Arduino Pro Micro Imagina TDR STEAM 3dBot / Imagina-Arduino Keyestudio EasyPlug Keyestudio KeyBot Otto DIY / Nano Otto DIY / Uno NodeMCU / ESP8266
Descripción	WeMos D1 / ESP8266 ESP32 / ESP32-WROOM

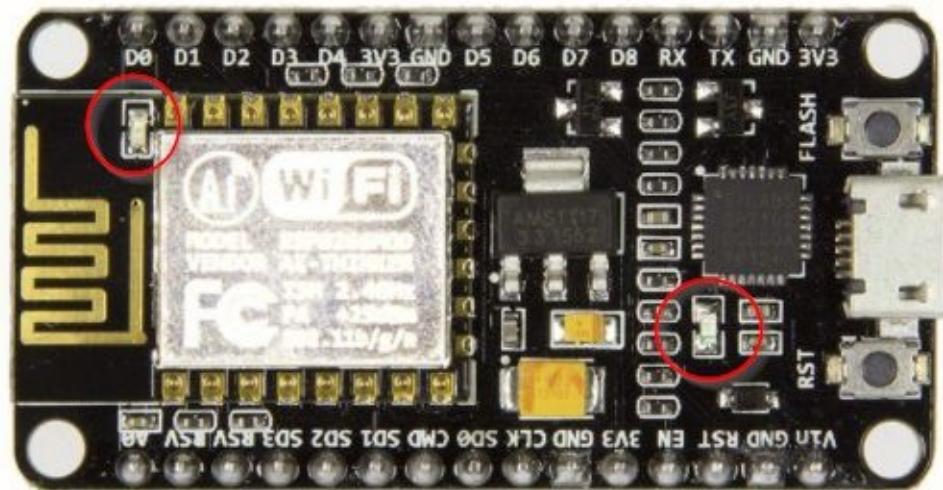
Luego el nombre y el resto de campos es optativo pero importante y buena costumbre rellenarlos, sobre todo si el proyecto lo compartimos:

- Descripción
- Componentes
- Comentarios

Intermitente en D4

Nuestro primer programa que vamos a hacer es un intermitente en D4

¿Por qué con D4 y no con otro? Porque D4 tiene un led integrado en la placa, también D0



ATENCIÓN

Estos leds van al revés, (configuración [pop-down](#)) luego **para encender el led, D4 tiene que estar a nivel bajo**, y al revés para apagarlo, D4 en nivel alto.

El programa es muy sencillo y lo completamos que se pueda visualizar por la salida puerto serie, por eso al iniciar ponemos el puerto serie a 9600 baudios.



Le damos a subir (teniendo el programa Arduinoblocks conector minimizado) y en "Cónsola" podemos ver el contenido del puerto serie.

Sensores

La ventaja de ARDUINOBLOCKS es que tiene ya incorporado una cantidad de bloques con sus librerías y funciones integradas por lo que el alumno no tiene que pelear con códigos y situaciones especiales, simplemente arrastrar el bloque y funcionar:



Probando el sensor LDR.

Vamos a probar este sensor que está conectado en una entrada analógica, para ello simplemente que escriba su valor en el puerto, y veremos cómo va cambiando.

Es recomendable para la lectura de los sensores, crear variables que almacenen los valores leidos, para luego utilizar estos valores:

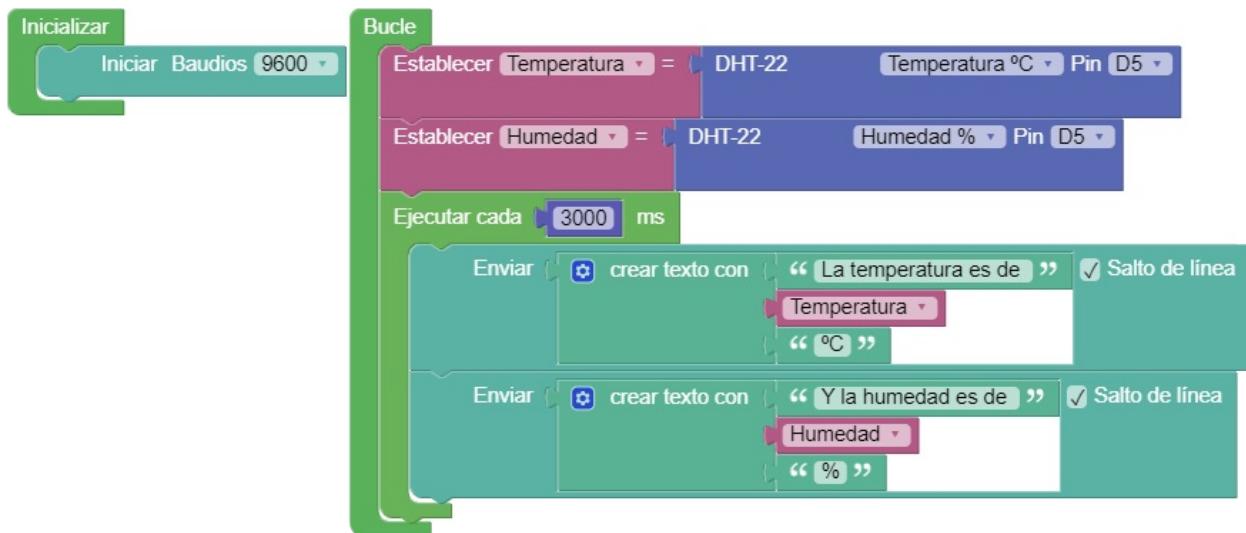


El siguiente código lee el sensor cada 2 seg. y lo vemos por el puerto serie. Podemos comprobar que **cuanto más luz, la lectura es menor**.



Probando el sensor DHT22.

Vamos a probar este sensor, que mide a la vez temperatura y humedad. Con el bloque correspondiente, nos ahorramos bastante código.



Probando el sensor Hall.

Este sensor tiene también su bloque :



Podemos construir un programa análogo al anterior y veremos que su salida es simplemente 0 o 1 depende si acercamos o no un imán.

MOTORES

Los motores **derechos** se gobiernan con la siguiente instrucción :



- Activando el pin D1 encendemos los motores
- El pin D3 en ON van hacia delante, si queremos que vayan hacia atrás, ponemos D3 en OFF
- El valor PWM del pin D1 es la potencia que transferimos al motor, puede ser desde 0 hasta 1023

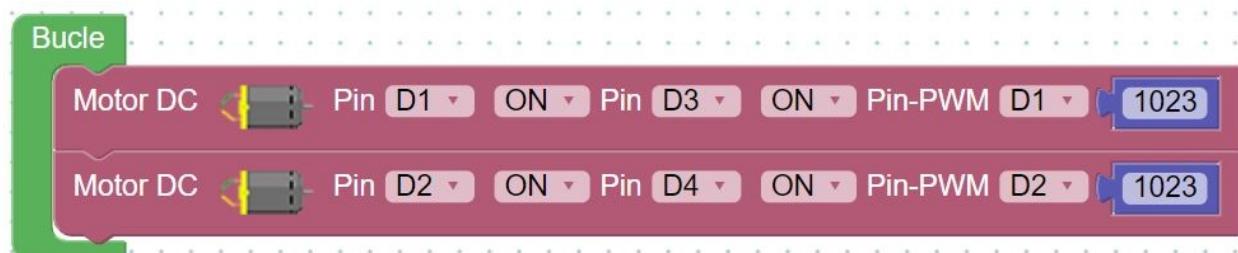
Recomendamos leer [esta página](#) sobre el significado de las salidas PWM.

Para los motores **izquierdos** se gobiernan con la siguiente instrucción :



- Activando el pin D2 encendemos los motores
- El pin D4 en ON van hacia delante, si queremos que vayan hacia atrás, ponemos D4 en OFF
- El valor PWM del pin D2 es la potencia que transferimos al motor, puede ser desde 0 hasta 1023.

Recomendamos ejecutar estas instrucciones y jugar con estos valores para ver si están bien conectados los motores :



3.1 Qué es

3.2 Crear cuenta

3.3.1 Token

Autores

Agradecimientos a :

CONSEJERÍA DE EDUCACIÓN Y FORMACIÓN PROFESIONAL DEL GOBIERNO DE CANTABRIA



Coordinación y montaje:

Cualquier observación o detección de error por favor aquí soporte.catedu.es

Los contenidos se distribuye bajo licencia Creative Commons tipo BY-NC-SA.



GOBIERNO DE ARAGÓN

Departamento de Educación,
Cultura y Deporte

