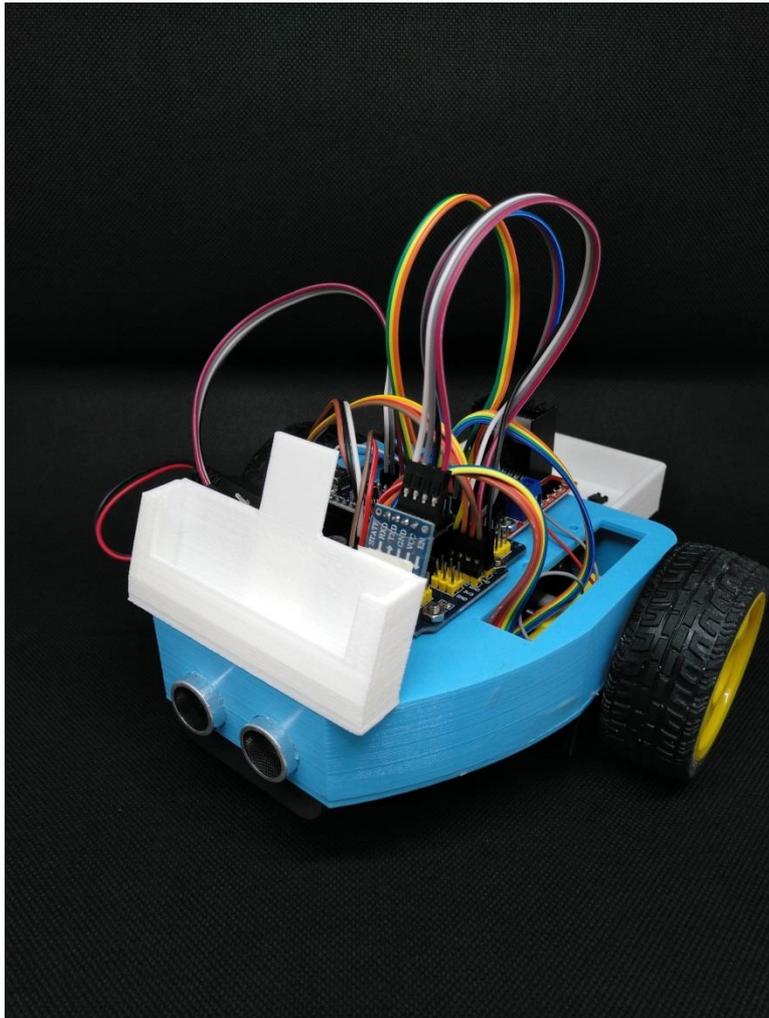


RETO ASCENTIC 2019

ASCENTICBOT



MANUAL DE MONTAJE Y PROGRAMACIÓN ASCENTICBOT

Práctica 2: MOTORES

Parte I: ARDUINOBLOCKS

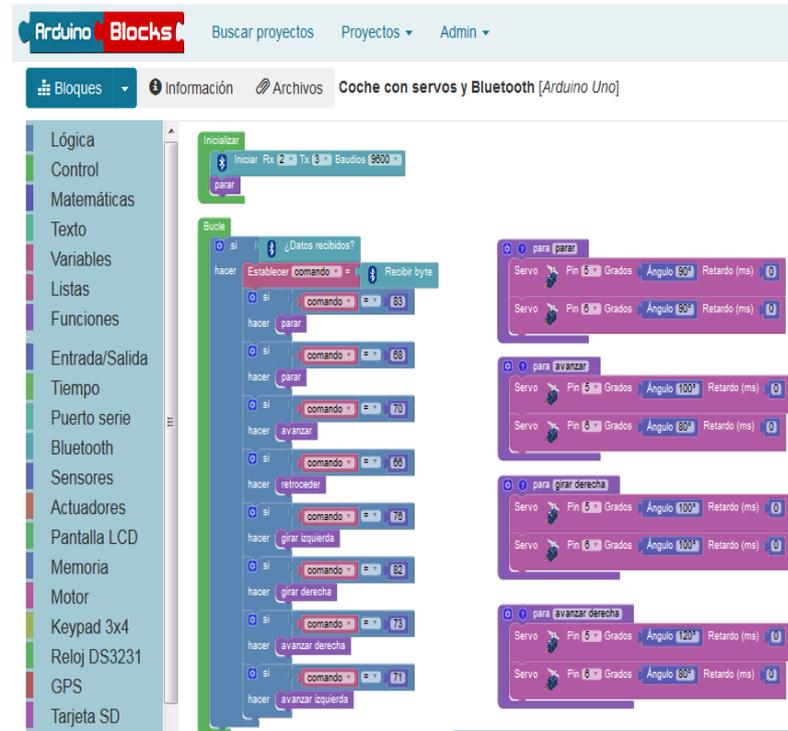
Arduinoblocks es un programa creado por el profesor Juanjo López. Gracias a su entorno gráfico facilita la programación de placas Arduino a todos los niveles. Esta herramienta permite programar a personas sin conocimientos previos de programación, pero su versatilidad y potencia es tan grande que expertos programadores también pueden utilizarlo.

ArduinoBlocks es una plataforma libre y online en la que sólo hay que registrarse.

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

En el siguiente enlace hay más información sobre **ArduinoBlocks** con tutoriales y prácticas.

[ARDUINOBLOCKS STEAM CANTABRIA](#)



Práctica 2: MOTORES

Parte I: ARDUINOBLOCKS

La placa que utiliza ASCENTIBOT es la **Arduino NANO** por lo que una vez inicializado un proyecto se debe seleccionar la placa **Arduino Nano/Atmega 328** en el **Tipo de proyecto**.



Nuevo proyecto

 Proyecto personal

[Iniciar un proyecto personal](#)

Empieza a trabajar en tu proyecto ahora mismo. Será totalmente privado para tí. Una vez finalizado, si quieres, lo puedes compartir con el resto del mundo!

 Nuevo proyecto personal

Tipo de proyecto

Arduino Nano / ATmega328

Nombre

Descripción

Normal 

IMPORTANTE!!! ANTES DE CARGAR EL PROGRAMA EN LA PLACA SE DEBE DESCONECTAR LA ALIMENTACIÓN DEL BLUETOOTH

Práctica 2: MOTORES

Parte II: Movimientos básicos de los motores:

Recordemos que los motores están conectados en los siguientes pines:

MOTOR A: EN A – 10, IN1 – 9, IN2 – 8

MOTOR B: EN B – 3, IN4 – 4, IN3 – 5

Ahora vamos a comprobar que es correcta la polaridad de los motores. Para ello vamos a realizar este sencillo programa.

The screenshot shows an Arduino IDE interface with the following components:

- Component Palette:** On the left, the 'Motor' category is selected, indicated by a red arrow. It contains three motor modules: L298N Iniciar, L298N Mover, and LM298P Iniciar.
- Program Structure:**
 - Inicializar (Initialize):** A purple block containing an 'L298N Iniciar' module. The pin configuration is: EN-A 10, IN1 9, IN2 8, IN3 5, IN4 4, EN-B 3. Red circles and arrows highlight these pin settings, with 'MOTOR A' and 'MOTOR B' labels above them.
 - Bucle (Loop):** A purple block containing two 'L298N Mover' modules.
 - The first module is for 'Motor A', set to 'Adelante' (Forward) with a 'Velocidad' (Velocity) of 255.
 - The second module is for 'Motor B', set to 'Adelante' (Forward) with a 'Velocidad' (Velocity) of 255.
- Annotations:** Red circles highlight the velocity values '255' in both 'L298N Mover' blocks. A red arrow points from these circles to the text 'PWM valores de 0 a 255' on the right.

Práctica 2: MOTORES

Parte II: Movimientos básicos de los motores:

Con este programa vamos a hacer que el robot avance durante 2 segundos, se pare otros dos segundos y retroceda dos segundos más.



```
graph TD
    subgraph Inicializar
        L298N_Iniciar[L298N Iniciar]
        L298N_Iniciar --> Bucle
    end
    subgraph Bucle
        L298N_Mover_A_Adelante[Motor A Adelante Velocidad 255]
        L298N_Mover_B_Adelante[Motor B Adelante Velocidad 255]
        Esperar_2000_1[Esperar 2000 milisegundos]
        L298N_Mover_A_Atras[Motor A Atras Velocidad 0]
        L298N_Mover_B_Atras[Motor B Atras Velocidad 0]
        Esperar_2000_2[Esperar 2000 milisegundos]
        L298N_Mover_A_Atras_2[Motor A Atras Velocidad 255]
        L298N_Mover_B_Atras_2[Motor B Atras Velocidad 255]
        Esperar_2000_3[Esperar 2000 milisegundos]
    end
    Bucle --> Bucle
```

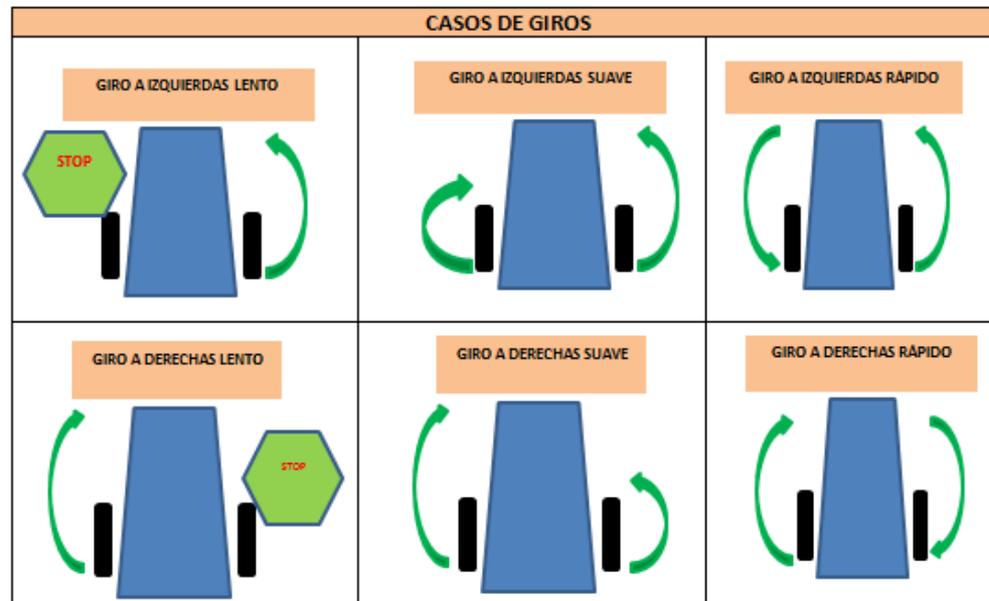
The code is written in the Arduino Blocks environment. It starts with an 'Inicializar' block containing an 'L298N Iniciar' block with the following settings: EN-A 10, IN1 9, IN2 8, IN3 5, IN4 4, and EN-B 3. This is followed by a 'Bucle' (loop) block. Inside the loop, the sequence of blocks is: 1. 'L298N Mover' for Motor A, direction 'Adelante', velocity 255. 2. 'L298N Mover' for Motor B, direction 'Adelante', velocity 255. 3. 'Esperar' block for 2000 milisegundos. 4. 'L298N Mover' for Motor A, direction 'Adelante', velocity 0. 5. 'L298N Mover' for Motor B, direction 'Adelante', velocity 0. 6. 'Esperar' block for 2000 milisegundos. 7. 'L298N Mover' for Motor A, direction 'Atrás', velocity 255. 8. 'L298N Mover' for Motor B, direction 'Atrás', velocity 255. 9. 'Esperar' block for 2000 milisegundos. The loop then repeats.

Práctica 2: MOTORES

Parte III: Giros:

Se pueden realizar 3 tipos de giros:

- A) Giro lento: en este giro el motor exterior tendrá movimiento y el motor interior quedará parado.
- B) Giro lento y amplio: para realizar este giro al motor exterior le damos un PWM de 255 y el motor interior un PWM por debajo de 100, pero ambos motores deben tener el mismo sentido de rotación.
- C) Giro rápido: tanto el motor exterior como el interior girarán con un PWM de 255, pero en sentidos contrarios.

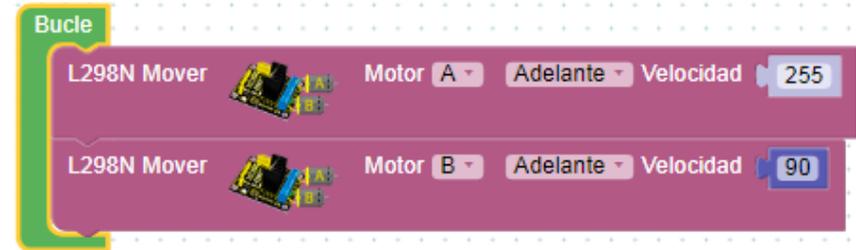


Práctica 2: MOTORES

Parte III: Giros:

Los programas de giros serían los siguientes:

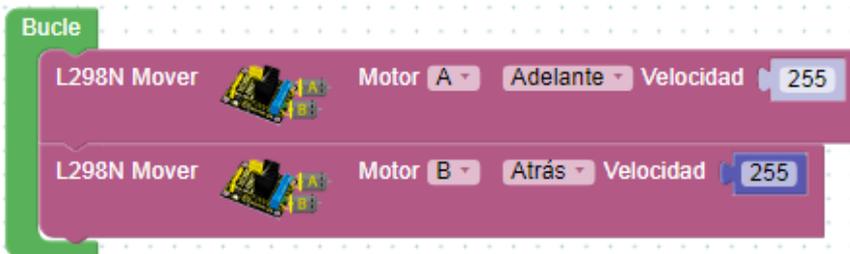
Ejemplo de giro lento y amplio a izquierdas



Ejemplo de giro lento a izquierdas



Ejemplo de giro rápido a izquierdas



Práctica 2: MOTORES

Parte IV: MiniRetos.



- .- **MiniReto I:** Avanza con el robot 50 cm, realiza un giro de 180° y avanza 50 cm. El robot debe volver al punto de partida.
- .- **MiniReto II:** Realiza los tres tipos de giros explicados anteriormente pero exactamente de 90° a izquierdas y derechas.
- .- **MiniReto III:** Utiliza el PWM para que el robot avance en línea recta variando la velocidad cada 2 seg. Utiliza 50, 100, 150, 200, 255 y 0.
- .- **MiniReto IV:** Haz que AscenticBOT realice un cuadrado de 50 cm de lado.