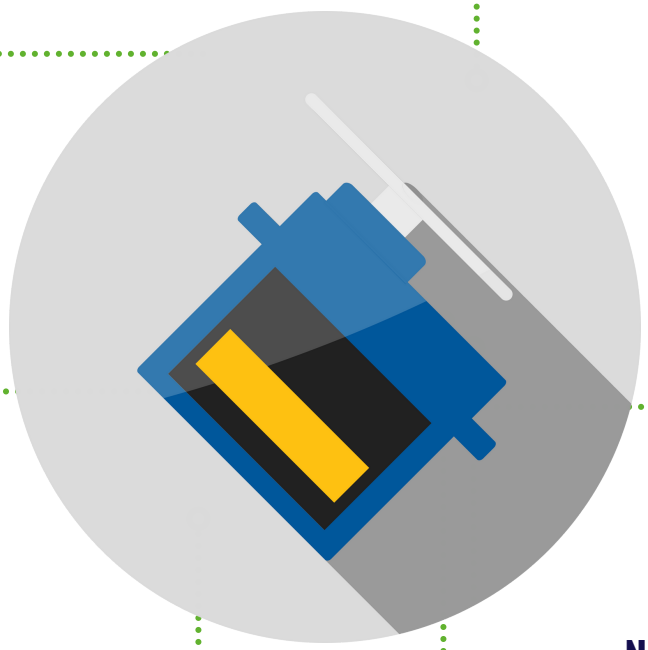
¡LOS SERVOS HACEN QUE LAS COSAS SE MUEVAN!

#R1AS13

¿Qué es?

El servo es un controlador para mantener la posición. Es adecuado para controlar un sistema con cambio de ángulo constante y puede mantener su estado.

Disponible en



Requisitos previos

- R1AS03 - Botones y pantalla LED

Duración

25 minutos

Nivel de dificultad

Intermedio

Material

- 1 placa de programación "***STM32 IoT Node Board***"
- Cable USB Micro-B
- 1 cable USB Micro-B
- 1 miniservo SG-90 (1,6 kg)
- Cables de puente

OBJETIVOS DE APRENDIZAJE

- Poner en movimiento un objeto

Esta hoja de actividades forma parte del proyecto Let's STEAM, financiado con el apoyo de la Comisión Europea a través del programa Erasmus + Asociación Estratégica. Su contenido refleja únicamente la opinión del autor, y la Comisión no se hace responsable del uso que pueda hacerse de la información contenida en ella. Esta obra está autorizada bajo la licencia Creative Commons Attribution-ShareAlike 4.0 International License.

Cofinanciado por el programa Erasmus+ de la Unión Europea

¡LOS SERVOS HACEN QUE LAS COSAS SE MUEVAN!

Un Servo es un motor con un conjunto de sistemas de control automático, que consta de un **motor de corriente continua** ordinario (motores eléctricos rotativos que convierten la energía eléctrica de corriente continua en energía mecánica), un reductor, un **potenciómetro** (divisor de tensión utilizado para medir el potencial eléctrico o la tensión) y un circuito de control. Puedes definir el ángulo de rotación del eje de salida mediante el envío de señales. Normalmente, un servo tiene un ángulo de rotación máximo (por ejemplo, 180 grados).
Fuentes: https://en.wikipedia.org/wiki/DC_motor, https://en.wikipedia.org/wiki/Potentiometer

El servosistema se puede controlar mediante un impulso, que puede cambiar su anchura. Utilizamos un cable de control para transmitir el impulso. El ciclo de una señal de referencia del servo es de 20ms y la anchura es de 1,5ms. La posición definida por la señal de referencia del servo es la posición media. Como el servo tiene un ángulo de rotación máximo, la definición de la posición media es a partir de esta posición donde el valor máximo y el valor mínimo son iguales.

HAZLO

1 Conecta el servo a la placa

Hay muchas formas de conectar un servo a tu placa. Puedes usar cualquier pin de salida analógica (pines PWM) para conectar el pin de control. En nuestro ejemplo, usaremos el pin **D2**. El servo se conectará de la siguiente manera:

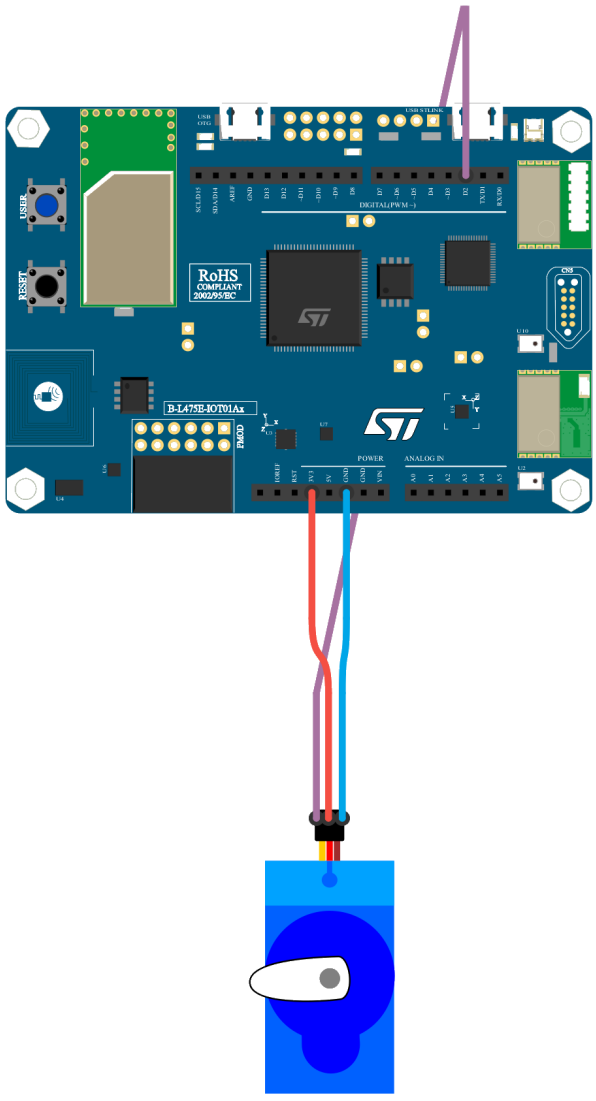
- Negro para **GND**
- Rojo para **V+ (3V3)**
- Naranja para **SIG (D2)**

2 Conecta la placa al ordenador

Con tu cable USB, conecta la placa a tu ordenador utilizando el conector **USB ST-LINK** (en la esquina derecha de la placa). Si todo va bien, deberías ver una nueva unidad en tu ordenador llamada **DIS_L4IOT**. Esta unidad se utiliza para programar la placa simplemente copiando un archivo binario.

3 Abra MakeCode y cree un nuevo proyecto en blanco

Ve al editor de **Let's STEAM MakeCode**. En la página de inicio, crea un nuevo proyecto haciendo clic en el botón "Nuevo proyecto". Dale un nombre a tu proyecto más expresivo que "Sin título" e inicia tu editor.
Fuente: makecode.lets-steam.eu



Conecta el servo a la placa

¡LOS SERVOS HACEN QUE LAS COSAS SE MUEVAN!

HAZLO

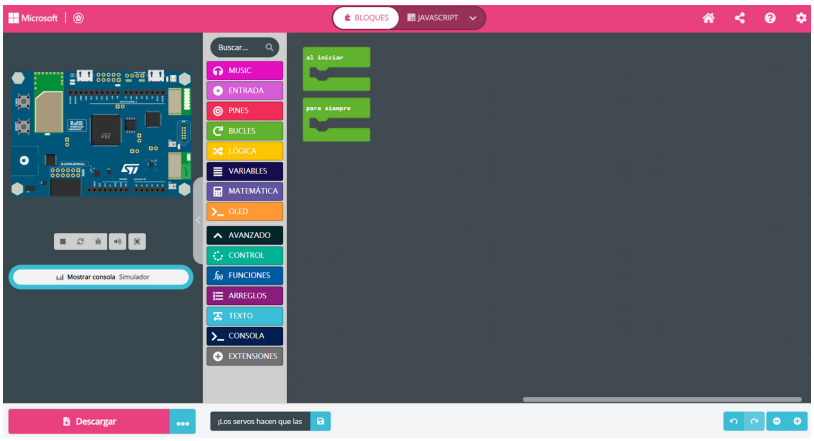
Después de crear su nuevo proyecto, obtendrá la pantalla predeterminada "lista para funcionar" que se muestra aquí.

4 Programa tu placa

Dentro del Editor de Javascript de MakeCode, copia/pega el código disponible en la Sección "**Prográmalo**" abajo.

Antes de probar este programa en la placa, puedes probarlo directamente dentro del simulador. Si cambias los valores 0 y 180, verás el resultado directamente.

Si no lo has hecho ya, da un nombre a tu proyecto y haz clic en el botón "**Descargar**". Copia el archivo binario en la unidad **DIS_L4IOT**, espera a que la placa termine de parpadear y tu servo comenzará a moverse.



Pantalla de "listo" por defecto

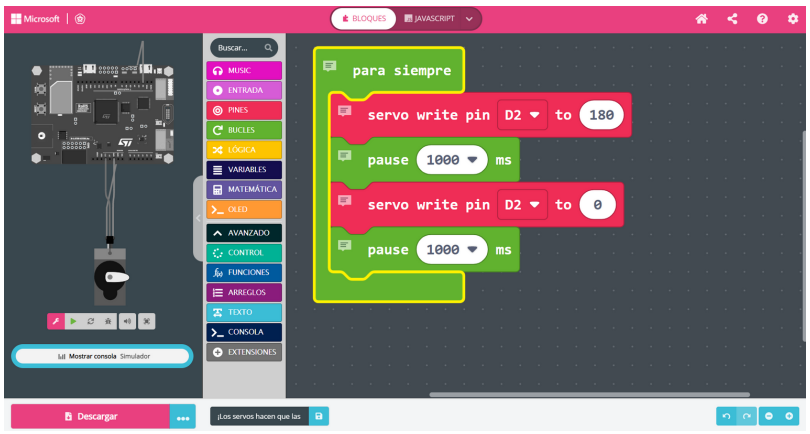
5 Ejecuta, modifica, juega

Tu programa se ejecutará automáticamente cada vez que lo guardes o reinicies tu placa (pulsa el botón etiquetado como RESET).

Si todo funciona bien, el servo comenzará a moverse.

Intenta entender el ejemplo y empieza a modificarlo cambiando el periodo entre los dos movimientos.

Editor de Javascript de Makecode



Su servo comienza a moverse

¡LOS SERVOS HACEN QUE LAS COSAS SE MUEVAN!

PROGRÁMALO

```
// va de 0 grados a 180 grados
forever(function () {
   // indicar al servo que vaya a la posición 180 grados
   pins.D2.servoWrite(180)
   // esperar a que el servo alcance la posición
   pause(1000)
   // indicar al servo que vaya a la posición 0 grados
   pins.D2.servoWrite(0)
   // esperar a que el servo alcance la posición
   pause(1000)
})
```

¿Cómo funciona?

Este ejemplo es bastante sencillo, ya que se trata del clásico "blinky" adaptado a un servo.

La instrucción principal es pins.D2.servoWrite(XXX). Esta instrucción pide al servo que gire en un ángulo de XXX grados (según tus necesidades específicas dependiendo del proyecto que estés desarrollando).

Para moverse entre dos posiciones, el servo tarda un tiempo, por lo que siempre hay que añadir un retardo antes de iniciar otro movimiento.

¡Este programa barre a diestro y siniestro para siempre!

En comparación con un motor de CC ordinario, un servo gira sólo dentro de un determinado rango de ángulos, mientras que un motor de CC ordinario gira en círculo.

Un servo no puede girar en círculo. Un motor ordinario de corriente continua no puede darnos información sobre el ángulo de rotación, pero un servo sí puede hacerlo. Por lo tanto, sus usos son diferentes.

Los motores de corriente continua ordinarios utilizan la rotación de todo el círculo como potencia, mientras que los servomotores utilizan un ángulo determinado de un objeto que controlan, como la articulación de un robot.

¡LOS SERVOS HACEN QUE LAS COSAS SE MUEVAN!

MEJÓRALO

Intenta **reducir al máximo el valor de la pausa** para eliminar cualquier parada de movimiento.

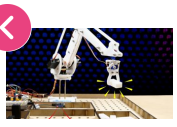
Añade instrucciones para **hacer una parada corta en la posición central**. Adapta el retardo de la pausa para asegurarte de que la parada sea muy corta.

Transforma este programa para **hacer un temporizador con un servo**. En cada paso, mueve el servo de 3 grados. Adapta el retardo para que cada paso dure aproximadamente 1s.

Inicia el **movimiento de barrido** sólo cuando se ha pulsado el botón USER.

¿QUIERES IR MÁS ALLÁ?

- **Servomotor** - Aprende más sobre el mecanismo y el funcionamiento del control del servomotor. *https://en.wikipedia.org/wiki/Servomotor*
- **Servomotores con micro:bit** - Todo sobre los botones y su uso en MakeCode con Shawn Hymel, creador de contenido técnico. *https://www.youtube.com/watch?v=okxooamdAP4&t=200s*, *https://shawnhymel.com*
- **Brazo robótico de clasificación de colores DIY** - Aprende a hacer su propio brazo robótico de clasificación de colores DIY utilizando sensores ultrasónicos e IR. *https://thestempedia.com/project/diy-color-sorting-robotic-arm/*



Fichas de actividades enlazadas

R1AS14 - Crear un temporizador de huevos