

# PROYECTO 09

## INGENIEROS DEL MOVIMIENTO

### Introducción

En este proyecto, titulado “Ingenieros del Movimiento”, el alumnado se adentrará en el mundo de la robótica móvil a través del uso de la placa Micro:bit y su placa de expansión para el control de motores. Desde el primer momento, el proyecto propone un enfoque práctico, en el que los estudiantes deberán montar su propio robot, programarlo y enfrentarse a retos que simulan situaciones reales. Esta propuesta permite trabajar de manera integrada contenidos del área de Tecnología, al tiempo que promueve el aprendizaje activo, la experimentación y la resolución de problemas.

A lo largo de las diferentes sesiones, los alumnos aprenderán a construir y controlar un robot móvil, a programar sus movimientos con precisión, a utilizar sensores para que el robot reaccione ante su entorno y a representar sus acciones mediante señales visuales en la matriz LED. Todo ello les permitirá comprender conceptos clave como la programación estructurada, el uso de bucles y condicionales, la calibración de distancias y velocidades, y la toma de decisiones automáticas en función de los datos recibidos.

Este proyecto se desarrolla en consonancia con el Decreto 65/2022 de la Comunidad de Madrid, que establece los aprendizajes esenciales del área de Tecnología en Educación Secundaria. Además de reforzar los contenidos curriculares, el proyecto potencia el desarrollo de las competencias clave del currículo: la competencia digital, el pensamiento computacional, el trabajo en equipo, la expresión visual de ideas técnicas y la capacidad de reflexionar y mejorar el propio trabajo.

### Objetivos

- Programar un robot móvil con Micro:bit para ejecutar movimientos precisos (avance, retroceso, giros).
- Analizar el efecto de la inercia en los movimientos y ajustar la programación para compensarlo.
- Utilizar sensores (luz, ultrasonidos, pulsadores...) para modificar el comportamiento del robot.
- Resolver retos de navegación en un entorno físico que simula condiciones reales.
- Comprender y aplicar estructuras algorítmicas como condicionales y bucles.
- Fomentar el trabajo colaborativo en la solución de problemas de ingeniería y diseño.

# Competencias

- Competencia digital: mediante el uso de la placa Micro:bit y su programación con MakeCode, favoreciendo el pensamiento computacional y la autonomía digital.
- Competencia matemática y en ciencia, tecnología e ingeniería (STEM): a través del uso de proporciones, cálculos de velocidad, tiempo y distancia, y la interpretación de datos de sensores.
- Competencia personal, social y de aprender a aprender: al trabajar en grupo, reflexionar sobre los errores y mejorar las soluciones de manera iterativa.
- Competencia ciudadana: al contextualizar los retos en situaciones reales de movilidad y seguridad.
- Competencia en conciencia y expresión culturales: a través de la representación gráfica de los estados del robot mediante símbolos e iconos en la matriz LED.

# Contenidos

- Programación y control. Programación estructurada por bloques. Uso de eventos, bucles, condicionales y variables.
- Tecnologías de la información. Uso y conexión de sensores. Lectura e interpretación de datos. Actuación en función de condiciones.
- Proyecto tecnológico. Diseño y ejecución de un proyecto de robótica móvil. Planificación, puesta en práctica, ajuste y exposición de resultados.

# Sesiones

## Sesión 1: Montaje del robot y primeras pruebas de avance

### Temporalización

45 minutos

### Tipo de actividad

Trabajo individual guiado y aprendizaje procedimental

### Descripción

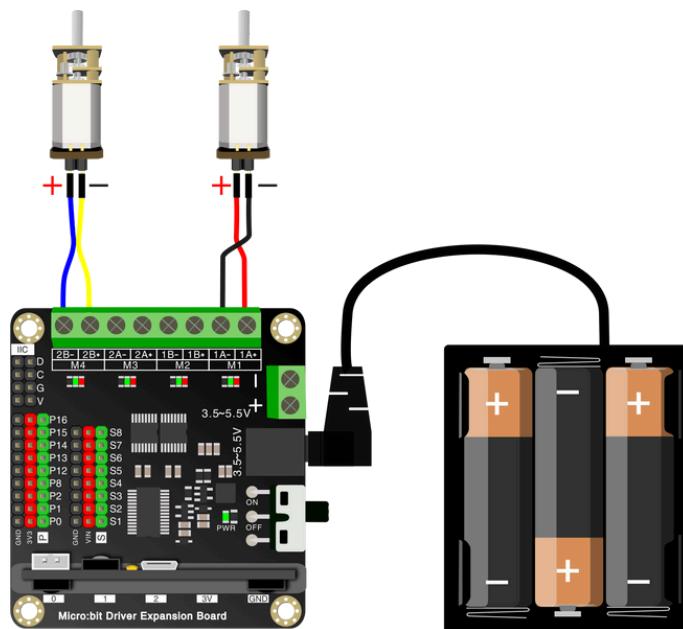
En esta primera sesión, el alumnado montará completamente el robot, utilizando una guía paso a paso. El montaje incluye la estructura del chasis, la fijación de los motores, la colocación de las ruedas, la instalación de la placa Micro:bit sobre la placa de expansión y la conexión de los elementos básicos.

Durante el proceso de montaje, se insistirá especialmente en que los motores deben conectarse tal como se muestra en la imagen de referencia incluida en la guía. Una conexión incorrecta puede hacer que el robot avance de forma errática o gire en sentido contrario al esperado. Por ello, se les pedirá que comprueben cuidadosamente la polaridad de los cables y su posición en la placa de expansión antes de continuar.

Una vez terminado el montaje, se instalará en MakeCode la extensión específica para controlar la placa de expansión. Finalmente, se cargará un programa básico que hace avanzar el robot durante unos segundos. Esta prueba servirá para comprobar que el montaje está correctamente realizado, que los motores funcionan como se espera, y que el robot responde a la programación inicial.

### Recursos

Micro:bit, placa de expansión, motores, ruedas, chasis, tornillería, cables, guía de montaje con imagen de conexión de motores, ordenador con MakeCode, cable USB, extensión <https://github.com/DFRobot/pxt-motor>



## Sesión 2: Programar una distancia concreta

---

### Temporalización

45 minutos

### Tipo de actividad

Exploración práctica y razonamiento matemático

### Descripción

En esta sesión el alumnado se centrará en programar desplazamientos con precisión. Partiendo de un programa que hace avanzar al robot durante un tiempo determinado, se planteará como reto que el robot recorra exactamente una distancia concreta (por ejemplo, 50 cm). Para ello se utilizará un campo de pruebas, que incluya una superficie milimetrada o referencias visuales que permitan medir con exactitud. El alumnado observará cuánto avanza su robot en un tiempo determinado y se les animará a no utilizar el método de prueba y error, sino aplicar razonamientos matemáticos. Por ejemplo, si con 1 segundo recorre 20 cm, ¿cuánto tiempo necesitará para 50 cm? Se fomentará el uso de reglas de tres y cálculos de proporcionalidad para ajustar el tiempo o la velocidad. Una vez conseguido el objetivo inicial, se propondrán otras distancias para comprobar si el razonamiento sigue siendo válido. Se anotarán los resultados y los posibles factores que hacen que el avance varíe (peso, fricción, nivel de batería...). Al final de la sesión se hará una puesta en común para valorar los métodos de cálculo y los ajustes realizados.

### Recursos

Micro:bit, placa de expansión, robot montado, ordenador con MakeCode, campo de pruebas con superficie milimetrada o referencias visuales, regla o cinta métrica, plantilla para registrar distancias y tiempos, extensión <https://github.com/DFRobot/pxt-motor>

## Sesión 3: Retos de desplazamiento: avances, retrocesos y control del movimiento

---

### Temporalización

45 minutos

### Tipo de actividad

Práctica técnica y resolución de retos por parejas

### Descripción

En esta sesión el alumnado pondrá en práctica lo aprendido sobre desplazamientos y lo aplicará a retos combinados de avance y retroceso. Partiendo de los programas anteriores, se ampliará la programación del robot para incluir movimientos de retroceso. Los retos propuestos consisten en recorrer distintas distancias hacia delante y hacia detrás, realizar paradas intermedias o ajustar la velocidad de desplazamiento.

Como parte del diseño del programa, el alumnado también utiliza la matriz LED para mostrar imágenes que indiquen el estado del robot. Por ejemplo, una flecha hacia adelante cuando avanza, una hacia atrás cuando retrocede, un cuadrado cuando se detiene o un ícono personalizado si cambia la velocidad. Esto les ayudará a depurar el programa y a ofrecer un feedback visual durante la ejecución.

Se insistirá en el uso de cálculos matemáticos y registros de observación, evitando el uso sistemático de prueba y error. Al final, cada grupo compartirá sus estrategias y se comentarán en común las soluciones más eficaces.

### Recursos

Micro:bit, placa de expansión, robot montado, ordenador con MakeCode, campo de pruebas con referencias visuales, reglas, plantilla de observación, extensión <https://github.com/DFRobot/pxt-motor>

## Sesión 4: Cómo girar: estrategias de rotación con dos motores

---

### Temporalización

45 minutos

### Tipo de actividad

Experimentación guiada y análisis de resultados

### Descripción

En esta sesión el alumnado aprenderá que existen tres formas principales de realizar un giro con un robot que utiliza dos motores.

Se explicarán con ejemplos prácticos las siguientes técnicas:

- Detener uno de los motores mientras el otro sigue en marcha.
- Hacer que un motor avance y el otro retroceda, generando un giro sobre el centro.
- Hacer que ambos motores avancen a distintas velocidades (esta última no se utilizará en esta sesión).

Antes de enfrentarse a los retos, se pedirá al alumnado que realice una fase de medición y análisis, programando ambos tipos de giro válidos durante exactamente 1 segundo y observando cuántos grados de giro realiza el robot en cada caso. Esta medición les permitirá calcular, mediante proporciones matemáticas, cuánto tiempo necesita su robot para girar 90°, 180°, etc., facilitando así la programación de los retos posteriores con mayor precisión.

A continuación, se propondrán varios retos de navegación, como recorrer un cuadrado, entrar en una zona delimitada o bordear un obstáculo. Para cada uno, deberán decidir qué técnica de giro utilizar, justificar su elección y ajustar los tiempos de rotación usando sus cálculos previos.

Además, se utilizará la matriz LED de la Micro:bit para representar los giros mediante imágenes (como flechas curvas o símbolos personalizados), ofreciendo una señal visual clara del comportamiento del robot. Al final de la sesión, los grupos compartirán sus resultados, explicarán cómo calcularon los tiempos de giro y analizarán qué técnica les dio mejores resultados en cada caso.

### Recursos

Micro:bit, placa de expansión, robot montado, ordenador con MakeCode, campo de pruebas con marcas visuales, regla, cronómetro, plantilla de observación, extensión

<https://github.com/DFRobot/pxt-motor>

## Sesión 5: Giros suaves y trayectorias curvas

---

### Temporalización

45 minutos

### Tipo de actividad

Investigación técnica con aplicación práctica

### Descripción

En esta sesión el alumnado descubrirá y pondrá en práctica el tercer tipo de giro que puede realizar un robot con dos motores: hacer que ambos motores avancen, pero a diferentes velocidades. Esta técnica permite realizar giros amplios y trayectorias curvas, como rodear un obstáculo o moverse en espiral.

Se explicará cómo configurar bloques en MakeCode para asignar diferentes niveles de potencia a cada motor, y se harán pruebas sobre el campo de pruebas para observar cómo varía el radio del giro según la diferencia de velocidad. A diferencia de los giros bruscos vistos en sesiones anteriores, en esta ocasión se trata de que el robot gire de forma suave y progresiva.

Una vez comprendido el funcionamiento, se propondrán varios retos de navegación, como:

- Recorrer un trazado en forma de U
- Completar un círculo o semicírculo
- Mantener una trayectoria curva constante durante varios segundos
- Cambiar de curva a línea recta y viceversa sin detenerse

En todos los casos, se animará al alumnado a usar cálculos matemáticos para ajustar las velocidades y medir la curvatura obtenida. También se utilizará la matriz LED para indicar cuándo el robot entra o sale de una curva, usando iconos como flechas curvas o imágenes personalizadas. Al final, se recogerán en grupo los aprendizajes clave sobre el control fino del movimiento y la utilidad de este tipo de giro frente a los anteriores.

### Recursos

Micro:bit, placa de expansión, robot montado, ordenador con MakeCode, campo de pruebas con trazados curvos o plantillas en papel, regla, cronómetro, extensión

<https://github.com/DFRobot/pxt-motor>

## Sesión 6: Retos finales: conducción autónoma y maniobras avanzadas

---

### Temporalización

45 minutos

### Tipo de actividad

Investigación técnica con aplicación práctica

### Descripción

En esta última sesión del proyecto, el alumnado se enfrentará a retos complejos de conducción autónoma que integran todo lo trabajado anteriormente: desplazamientos precisos hacia adelante y hacia atrás, paradas intermedias, uso de los tres tipos de giro, control de velocidad y representación gráfica en la matriz LED.

Los retos estarán inspirados en situaciones reales de conducción, y se desarrollarán sobre un campo de pruebas diseñado con diferentes zonas. Algunos de los desafíos propuestos serán:

- Realizar un aparcamiento en línea o en batería, ajustando retrocesos y giros.
- Simular un adelantamiento, comenzando en línea recta, haciendo una curva para sortear un obstáculo y volviendo al carril.
- Entrar, circular y salir de una rotonda, utilizando el tercer tipo de giro para describir un movimiento circular fluido.
- Navegar por un recorrido con múltiples paradas, como si se tratase de un autobús con distintas paradas señalizadas.

Los equipos deberán planificar el recorrido, programarlo, probarlo y ajustar los tiempos y velocidades en función de los resultados. Se fomentará el uso de la matriz LED para indicar el estado del robot (por ejemplo, señales de stop, dirección, intermitentes simulados, etc.). Al finalizar, cada grupo podrá compartir una demostración y explicar su solución al resto de la clase, destacando los elementos técnicos que han utilizado y las dificultades superadas.

### Recursos

Micro:bit, placa de expansión, robot montado, ordenador con MakeCode, campo de pruebas con zonas señalizadas (rotonda, carriles, líneas de parada), materiales para simular obstáculos, plantillas de planificación de trayecto, extensión <https://github.com/DFRobot/pxt-motor>

## Sesión 7: Día y noche: detección de luz y respuesta del robot

---

### Temporalización

45 minutos

### Tipo de actividad

Exploración de sensores y desarrollo de lógica condicional

### Descripción

En esta sesión el alumnado aprenderá a utilizar el sensor de luz integrado en la placa Micro:bit para que el robot responda a diferentes condiciones de iluminación. Se explicará cómo la Micro:bit puede medir la cantidad de luz ambiental y devolver un valor numérico entre 0 y 255. A partir de esa lectura, se construirá una estructura condicional en MakeCode que permita modificar el comportamiento del robot.

Como primer ejercicio, se programará al robot para que avance cuando haya luz (simulando el día) y se detenga cuando haya oscuridad (simulando la noche). Una vez comprendido el funcionamiento del sensor, se pasará a un reto más completo: los alumnos deberán hacer que el robot reduzca su velocidad al detectar poca luz y recupere la velocidad normal al volver a un entorno iluminado.

Para ello, el profesor tendrá preparado un túnel construido con cajas de zapatos u otros materiales, por el que el robot debe pasar. Al entrar, el nivel de luz disminuirá y el robot deberá adaptarse automáticamente. Además, se les animará a mostrar un ícono representativo en la matriz LED cuando el robot detecte que está en una zona oscura, como una luna o un símbolo de noche.

### Recursos

Micro:bit, placa de expansión, robot montado, ordenador con MakeCode, túnel preparado por el docente (con cajas de zapatos u otros materiales), campo de pruebas, regla, extensión <https://github.com/DFRobot/pxt-motor>

## Sesión 8: Seguridad y reacción: retos con el sensor de ultrasonidos

---

### Temporalización

45 minutos

### Tipo de actividad

Programación avanzada con toma de decisiones autónomas

### Descripción

En esta sesión el alumnado trabajará con el sensor de ultrasonidos, repasando su funcionamiento y explorando nuevas aplicaciones. Se comenzará recordando que este sensor permite medir la distancia entre el robot y un objeto cercano mediante la emisión y recepción de ondas sonoras, y que devuelve un valor en centímetros. Se explicará cómo conectarlo correctamente a la placa de expansión y cómo programarlo desde MakeCode.

A continuación, se propondrán varios retos prácticos que irán aumentando en complejidad:

- Detener el robot automáticamente al detectar un obstáculo a menos de una distancia definida (por ejemplo, 15 cm).
- Mostrar una imagen de aviso en la matriz LED cuando haya peligro de colisión.
- Mantener la distancia de seguridad respecto a otro objeto en movimiento.
- Simular una conducción con frenado progresivo.

### Recursos

Micro:bit, placa de expansión, robot montado, sensor de ultrasonidos, ordenador con MakeCode, obstáculos (cajas, objetos móviles), campo de pruebas, extensión <https://github.com/DFRobot/pxt-motor>

# Criterios de Evaluación

- Programación del Robot Móvil: El código es funcional, organizado y utiliza estructuras de control adecuadas.
- Uso de Sensores: El robot reacciona correctamente a estímulos del entorno, utilizando la lectura de sensores para tomar decisiones.
- Precisión y Optimización de Movimientos: El robot se desplaza y gira de forma controlada y ajustada a los objetivos del reto.
- Resolución de Retos en Campo de Pruebas: El robot supera obstáculos y cumple objetivos mediante soluciones eficaces.
- Trabajo en Equipo y Documentación del Proceso: Participación activa del alumnado, registro de observaciones, mejoras implementadas y exposición del proyecto final.