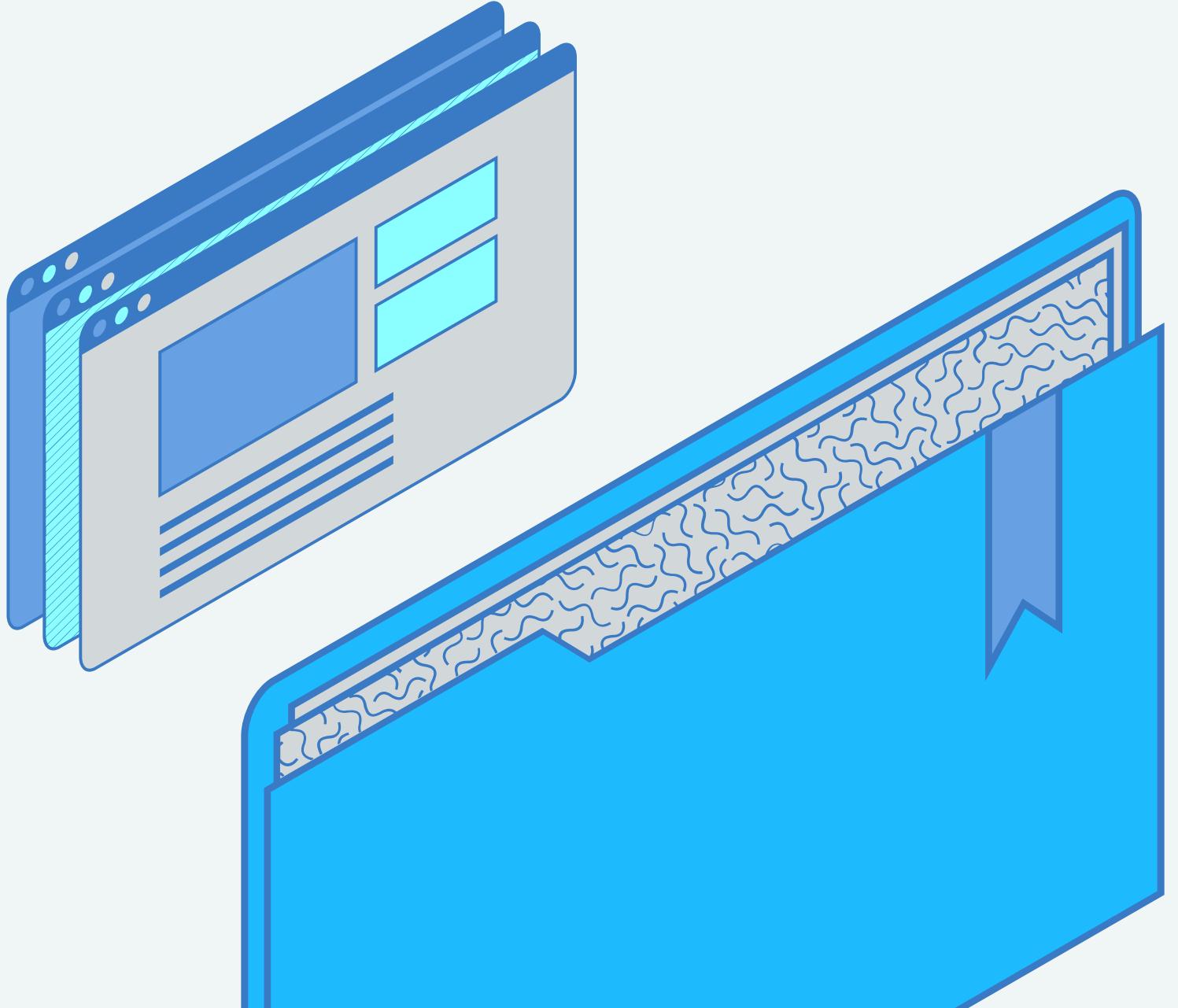


INTRODUCCIÓN A MICROCONTROLADORES

16 de Junio 2025

Antonia I. Morales Gutiérrez

íNDICE



01. ¿Qué es un microcontrolador?

02. Diferencias

03. Arquitectura básica

04. Tipos y/o familias

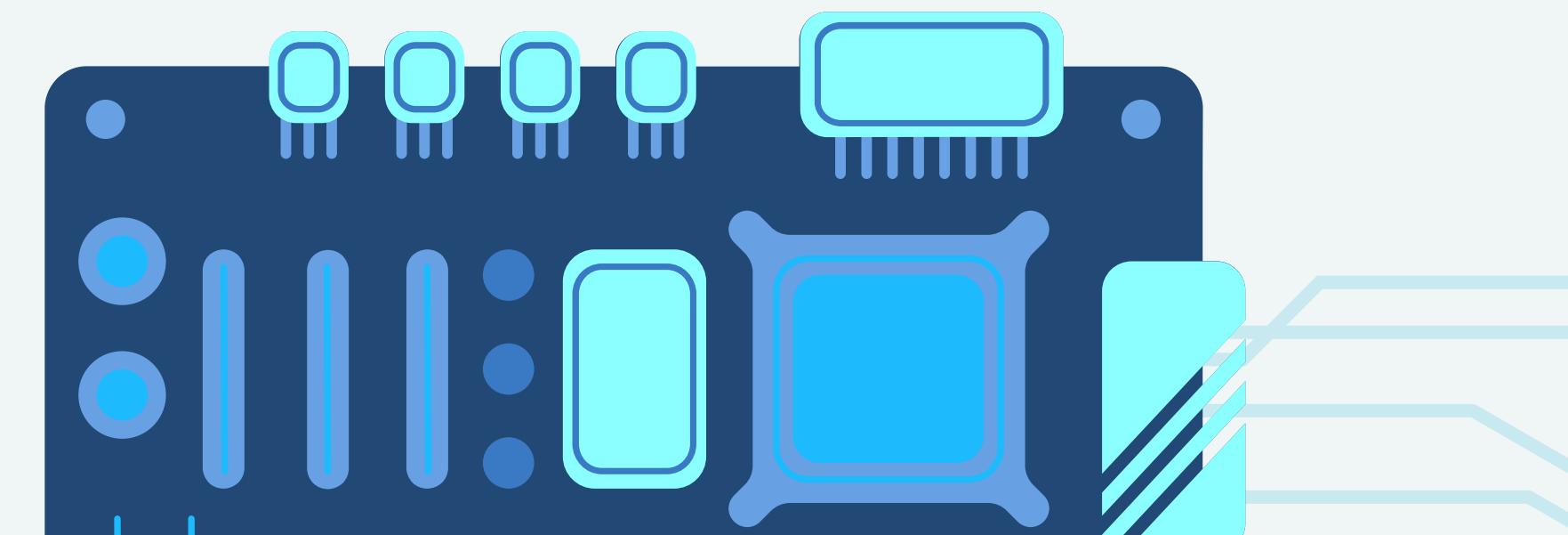
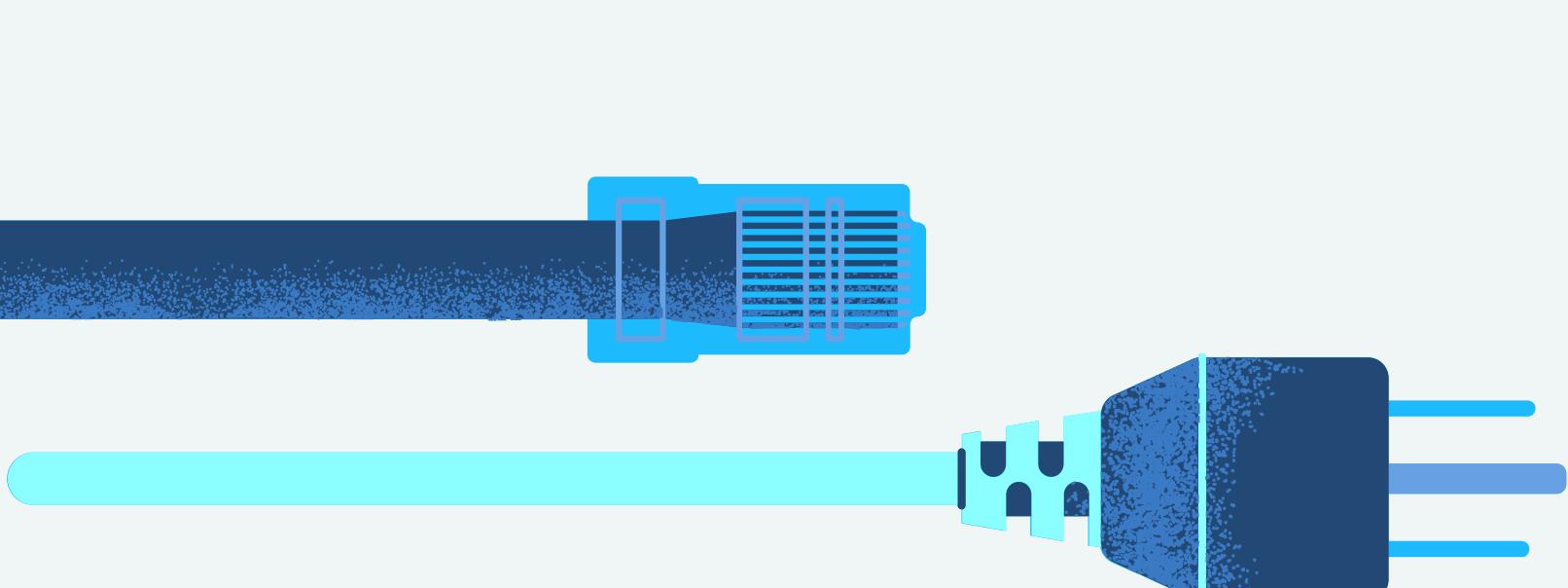
05. Buenas prácticas

06. Proyectos interesantes



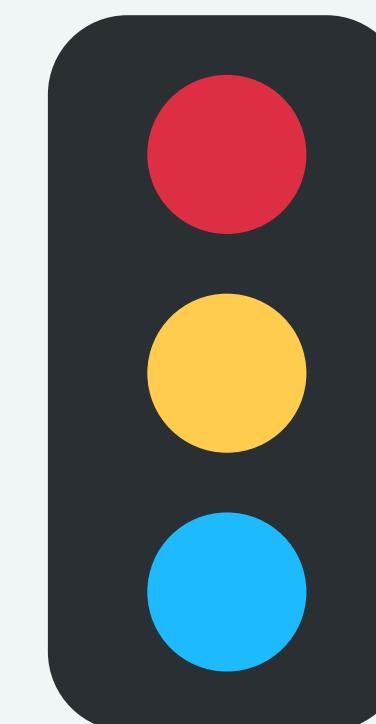
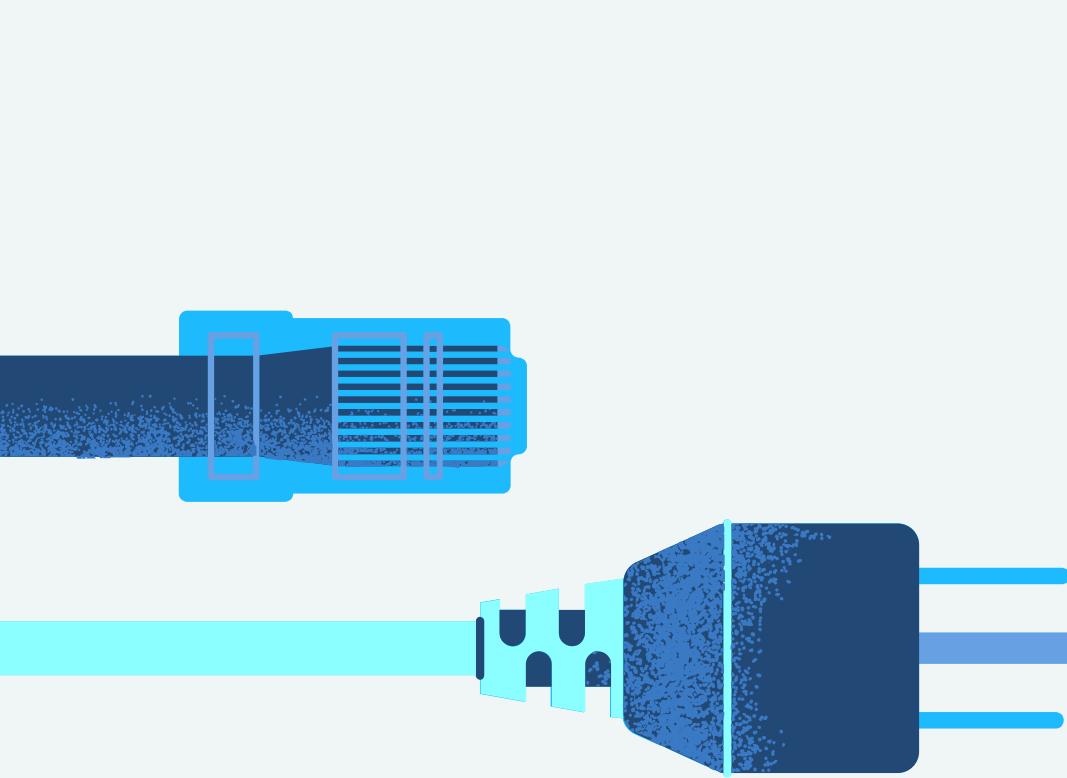
01. ¿QUÉ ES UN MICROCONTROLADOR?

Es un **circuito integrado programable** que contiene, en un solo chip, todos los componentes necesarios para ejecutar tareas de control automático en sistemas embebidos. En términos simples, actúa como un **pequeño computador autónomo** diseñado para realizar funciones específicas en dispositivos electrónicos.



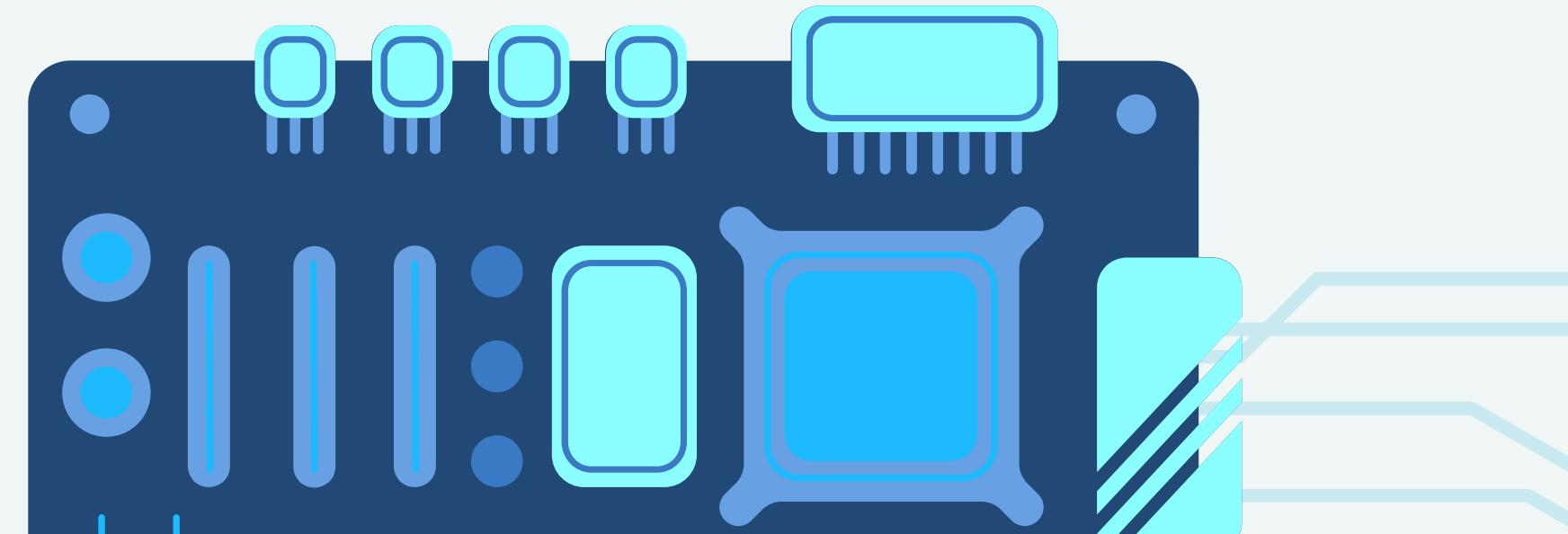
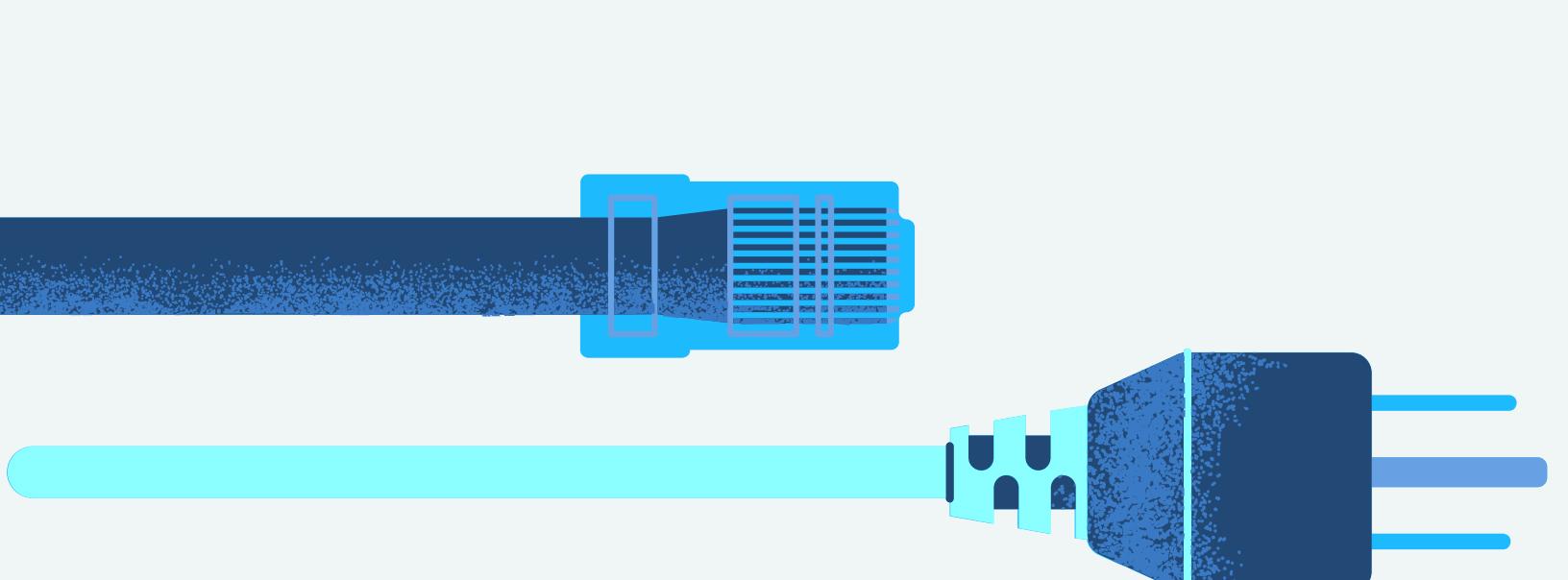
¿QUÉ ES UN SISTEMA EMBEBIDO?

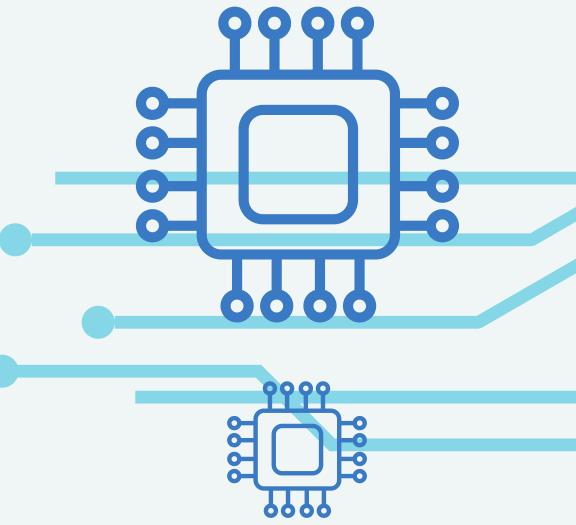
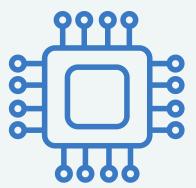
Es una combinación de hardware y software que opera de forma **autónoma**, generalmente con recursos limitados, y está embebido (integrado) dentro de un producto o sistema mayor. Son generalmente no reconfigurables por el usuario final



¿QUÉ ES UN MICROPROCESADOR?

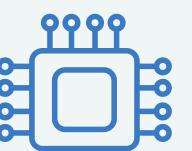
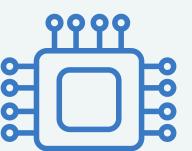
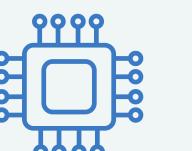
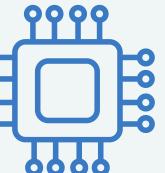
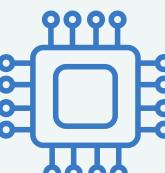
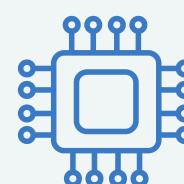
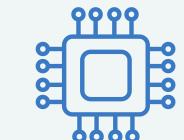
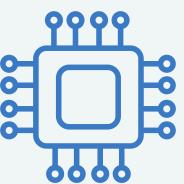
Es el núcleo de un sistema de cómputo de propósito general. Se trata de un chip electrónico que interpreta y ejecuta instrucciones, coordinando el funcionamiento de los demás componentes mediante buses de datos, direcciones y control. Es el cerebro de los sistemas computacionales de propósito general. A diferencia de un microcontrolador, **no tiene todo integrado**, pero es mucho más potente y versátil, ideal para ejecutar múltiples programas y gestionar grandes volúmenes de datos.



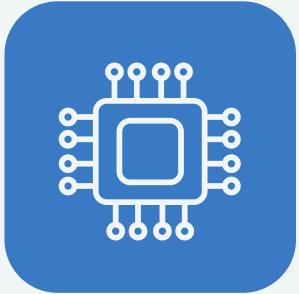


02. DIFERENCIAS

Característica	Microprocesador	Microcontrolador
Componentes integrados	Solo CPU	CPU + memoria + I/O + periféricos
Uso	Computación general	Sistemas embebidos / control dedicado
Complejidad del sistema	Alta, requiere circuitos adicionales	Baja, sistema autosuficiente
Coste	Más alto	Más económico
Velocidad y potencia	Alto	Moderada, suficiente para tareas específicas
Consumo de energía	Alto	Bajo



03. ARQUITECTURA BÁSICA



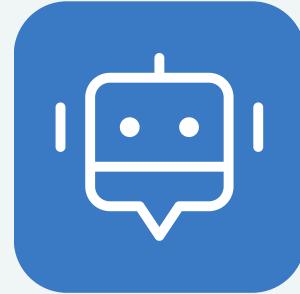
CPU (Unidad Central de Procesamiento)

- Es el "cerebro" del microcontrolador.
- Ejecuta instrucciones del programa almacenado.
- Realiza operaciones aritméticas, lógicas y de control.



Temporizadores y Contadores

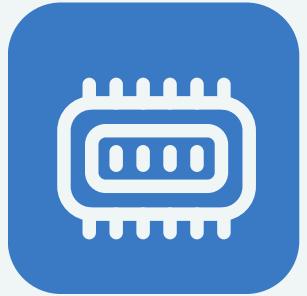
Permiten medir el tiempo, generar señales periódicas (PWM) y eventos precisos..



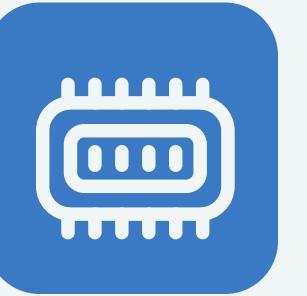
Periféricos de Entrada/Salida (I/O)

Los Puertos de entrada y salida son uno o más conectores donde el microcontrolador recibe información.

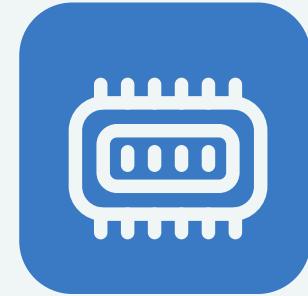
03. ARQUITECTURA BÁSICA



RAM



ROM/FLASH



EEPROM

La *Memoria de acceso aleatorio*, es el espacio donde se almacena la información recibida de manera temporal.

La Memoria de sólo lectura es un componente donde se almacenan las instrucciones básicas necesarias para que el microcontrolador funcione.

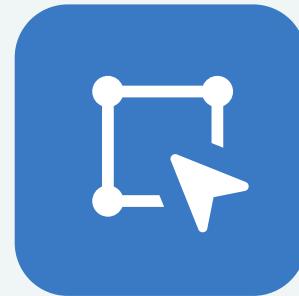
La ROM programable y borrable eléctricamente es el componente donde se almacenan las instrucciones programadas en el microcontrolador.

03. ARQUITECTURA BÁSICA



Conversores A/D y D/A
(ADC/DAC)

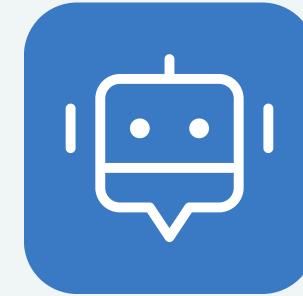
La *Memoria de acceso aleatorio*, es el espacio donde se almacena la información recibida de manera temporal.



Módulos de
Comunicación

Permiten que el microcontrolador se comunique con otros dispositivos:

- WiFi, Bluetooth, USB (como en ESP32 o STM3)

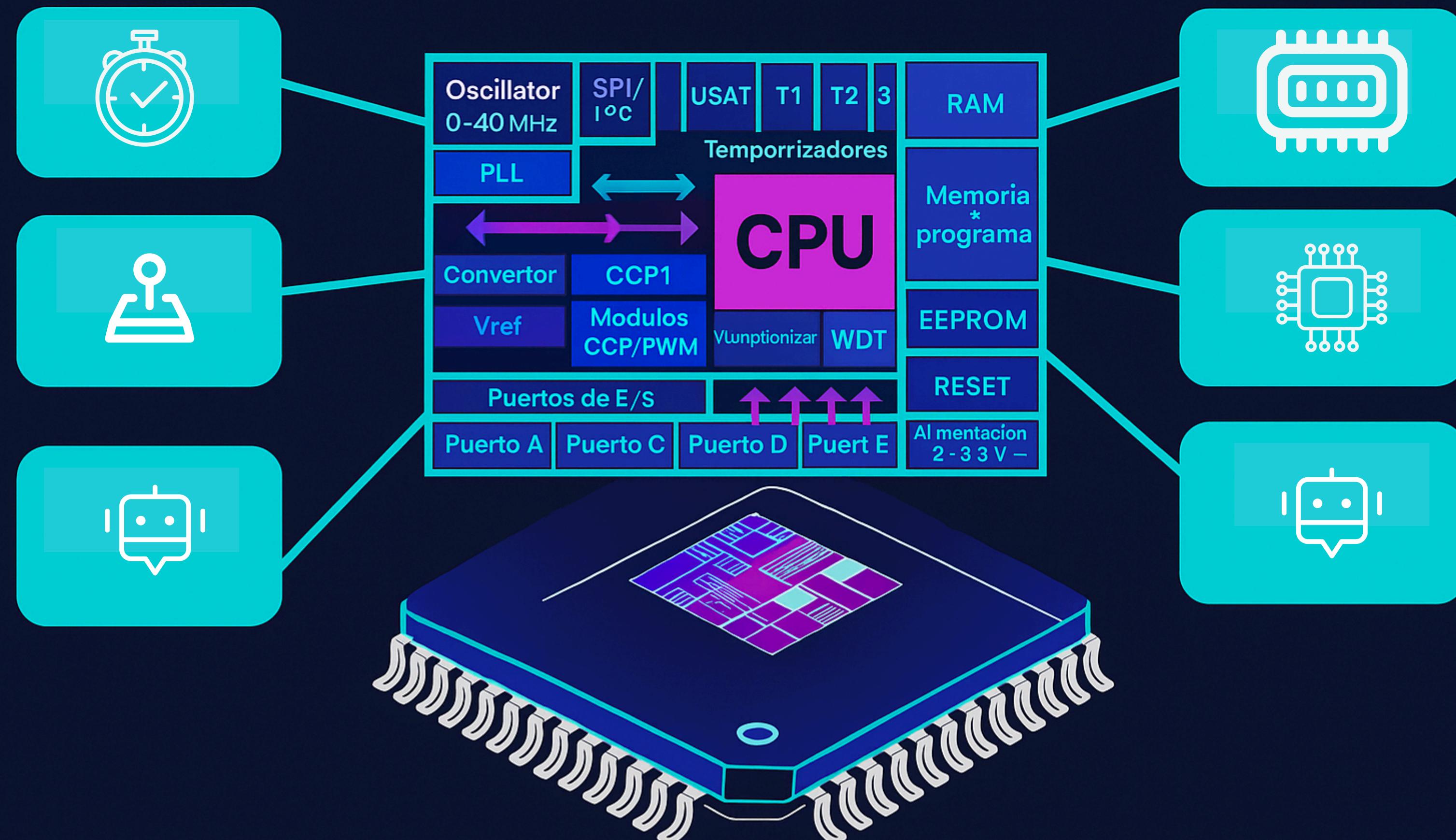


Sistema de
interrupciones

La ROM programable y borrable eléctricamente es el componente donde se almacenan las instrucciones programadas en el microcontrolador.

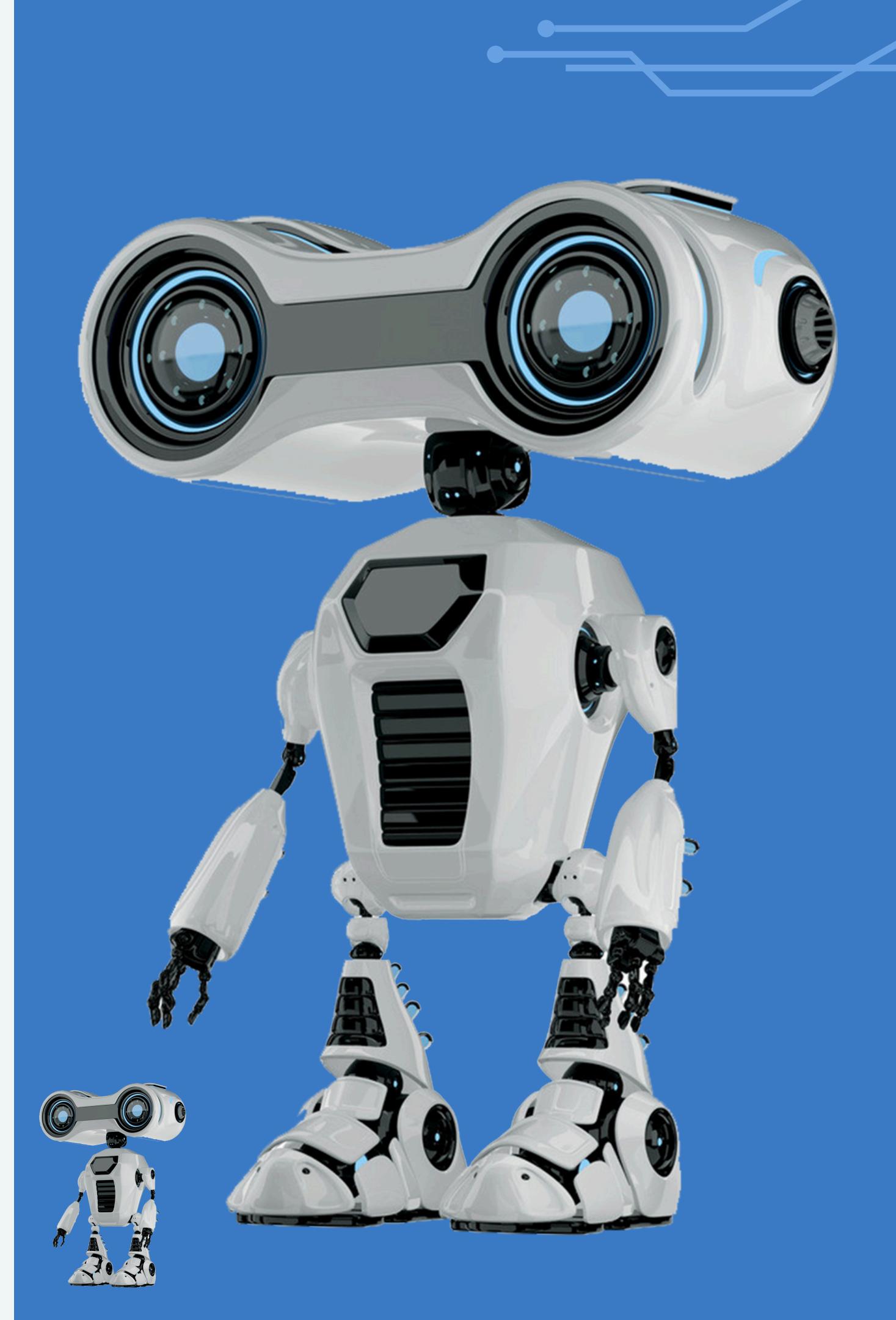
Microcontroladores

311



04. TIPOS Y/O FAMILIAS

En el mundo de la electrónica digital y la robótica, existen diversas familias de microcontroladores que se han desarrollado para atender distintos **tipos de aplicaciones**. Cada familia se caracteriza por sus capacidades de procesamiento, conectividad, facilidad de uso, consumo energético y entorno de programación

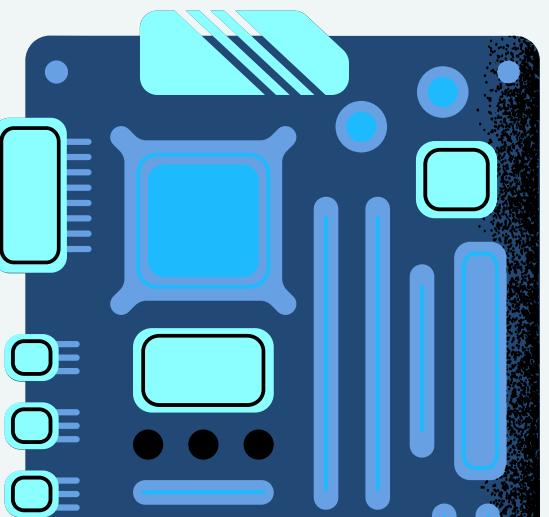
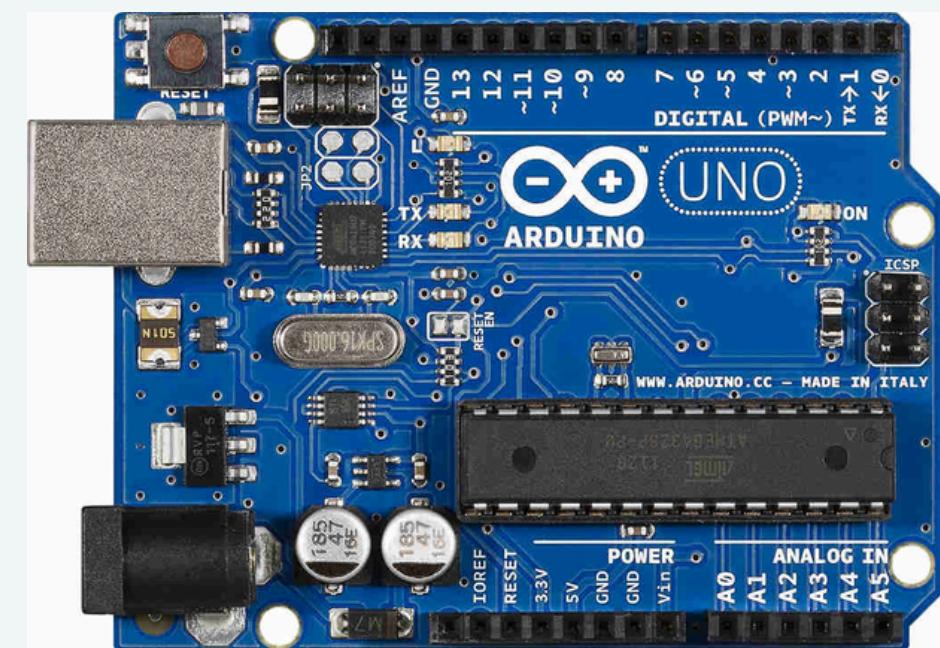


AVR (ARDUINO - ATMEL/MICROCHIP)

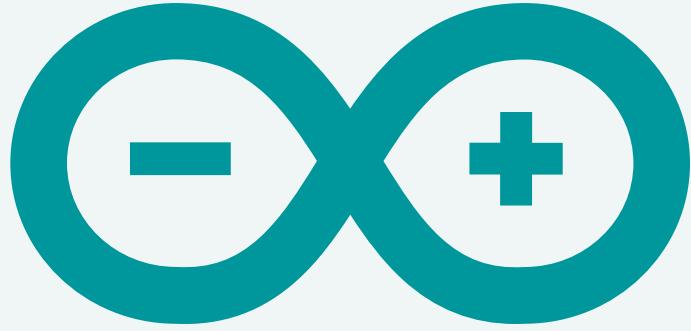


Familia de microcontroladores de 8 bits basada en arquitectura RISC (Reduced Instruction Set Computing).

- Utilizan una frecuencia de reloj común de 16 MHz, aunque puede variar según el modelo.
- Programación extremadamente accesible gracias al entorno Arduino IDE, que permite codificar en lenguaje C/C++ con funciones simplificadas.

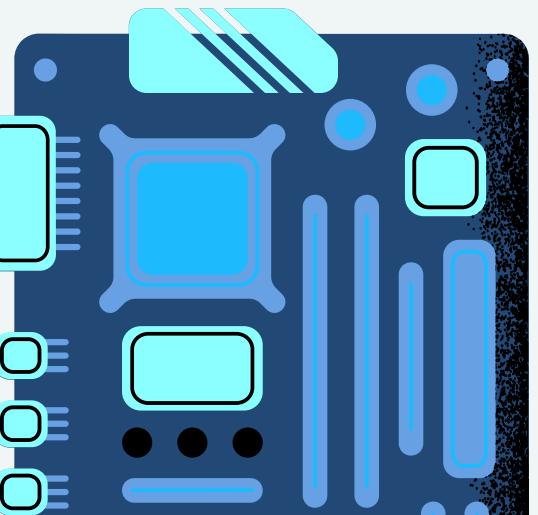
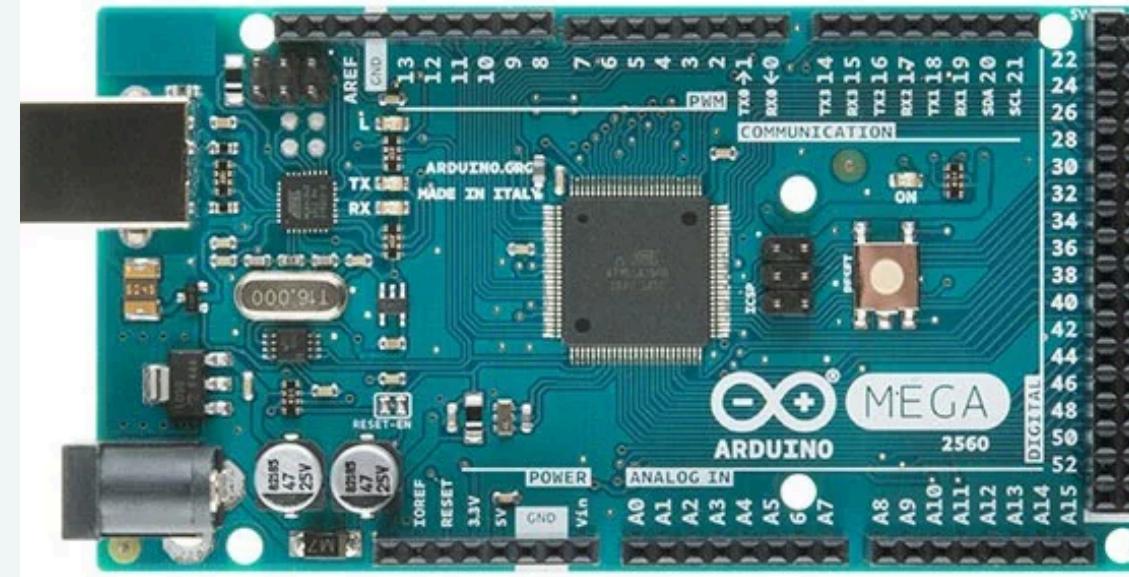


AVR (ARDUINO - ATMEL/MICROCHIP)



ARDUINO

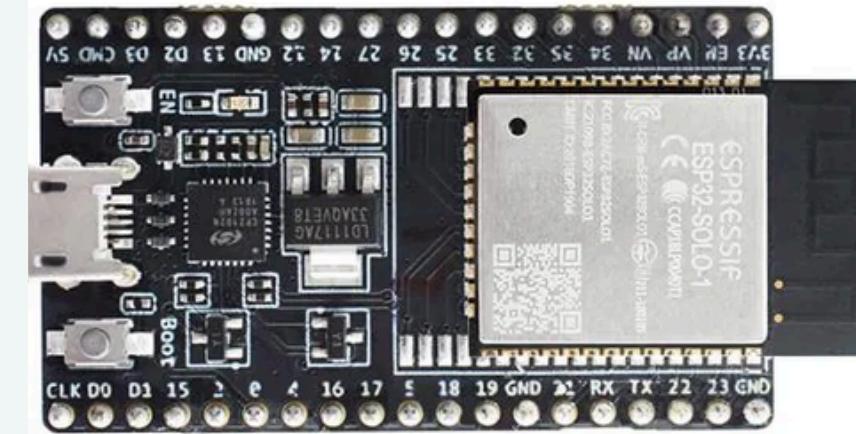
- Utilizados en educación, debido a su facilidad para configurar entradas/salidas digitales y analógicas.
- Incluyen convertidores ADC, módulos PWM, timers, y comunicación UART, SPI, I²C integrados.
- Disponen de memoria Flash (programa), SRAM (datos temporales) y EEPROM (datos permanentes)



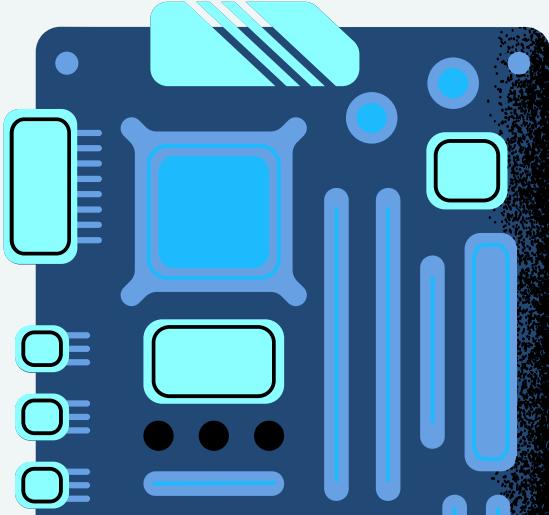
ESP32 (ESPRESSIF SYSTEMS)

Microcontrolador de 32 bits, basado en núcleo Xtensa LX6 o RISC-V (según modelo)

- Integra WiFi y Bluetooth BLE, lo que lo hace ideal para proyectos de Internet de las Cosas (IoT) sin módulos externos.
- Capaz de operar a frecuencias de hasta 240 MHz, con doble núcleo en muchos modelos.



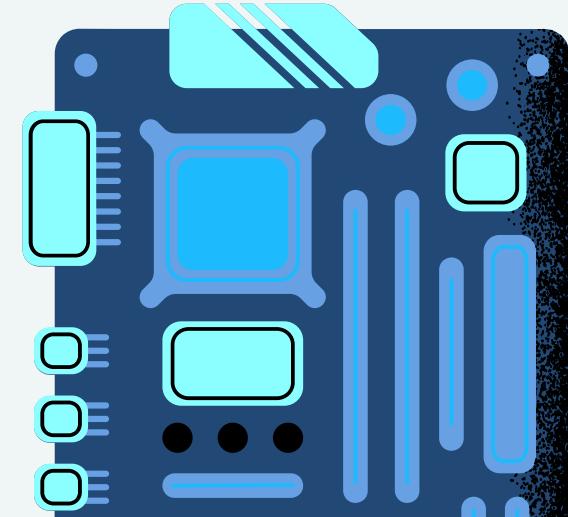
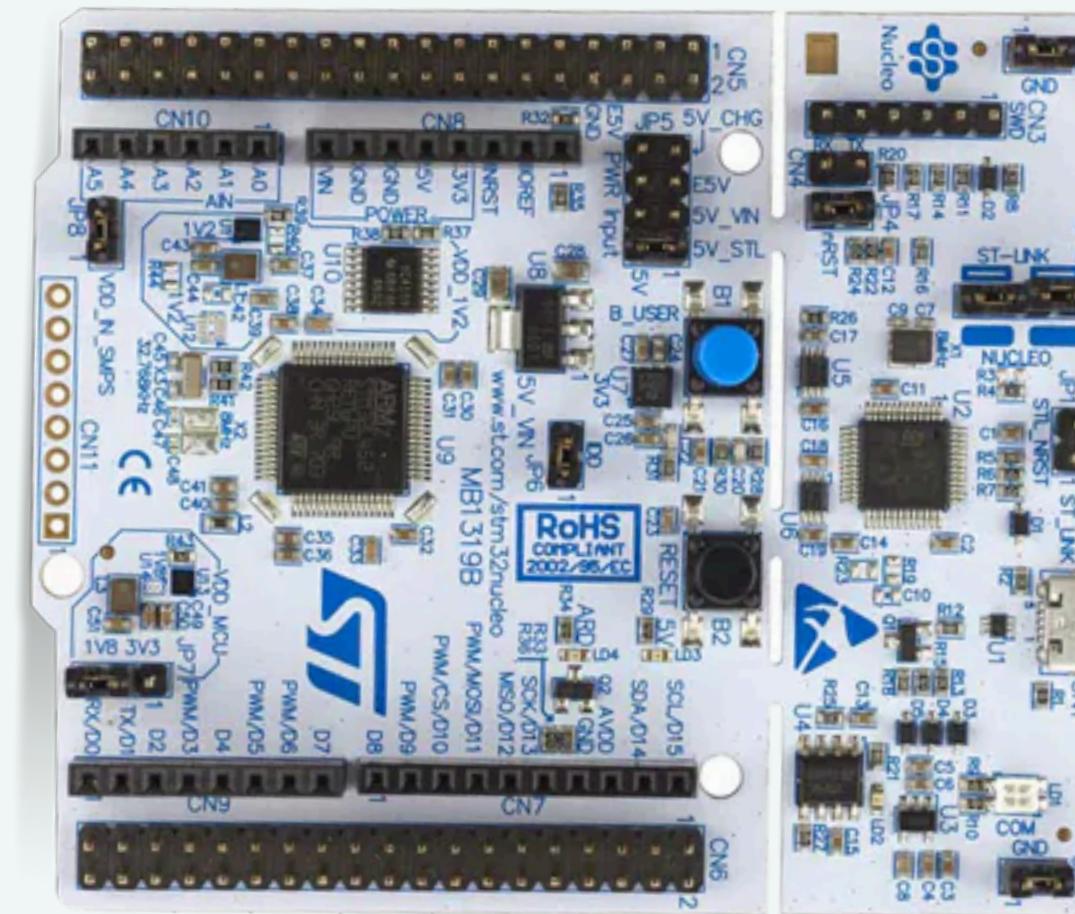
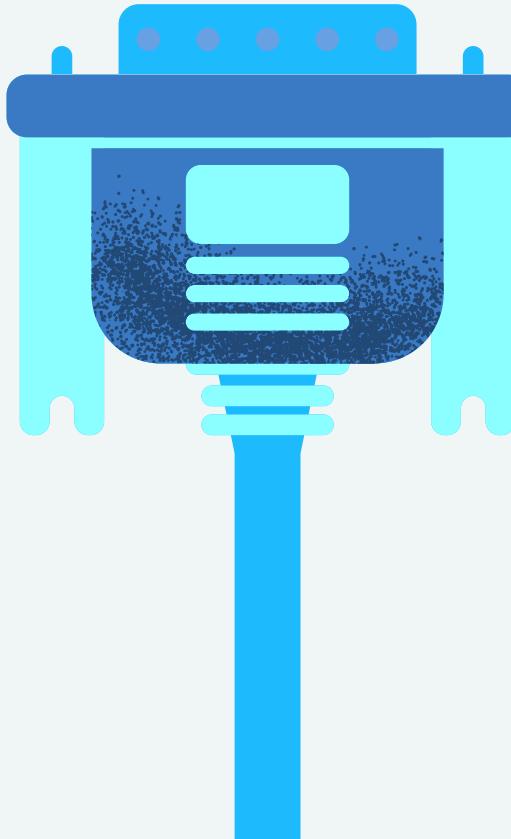
- Cuenta con gran cantidad de pines I/O y soporte para interrupciones, timers y deep sleep.
- Se puede programar en Arduino IDE, PlatformIO, MicroPython, ESP-IDF (entorno nativo).

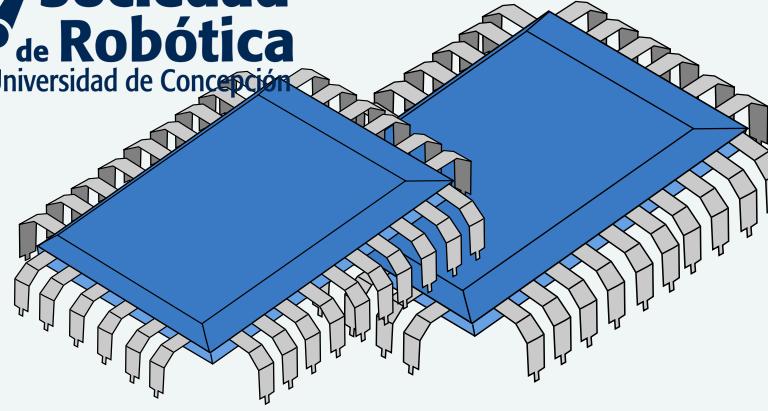
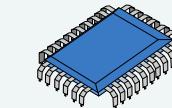


STM32

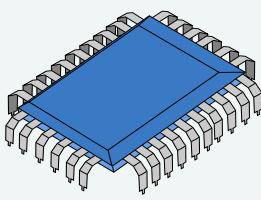
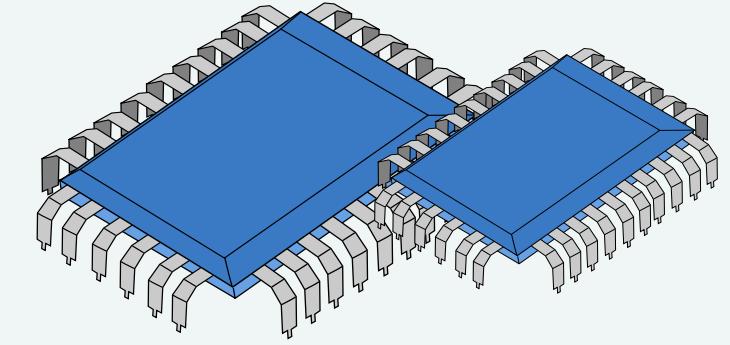
(STMICROÉLECTRONICS)

TM32 es una familia de microcontroladores de 32 bits basados en la arquitectura ARM Cortex-M. Se utilizan en proyectos que requieren alto rendimiento, eficiencia energética y control avanzado. Aunque su programación es más técnica, ofrecen gran flexibilidad y potencia para robótica, procesamiento de señales y dispositivos portátiles.

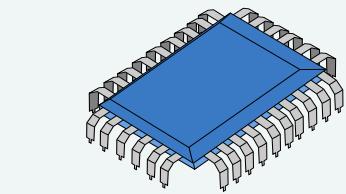
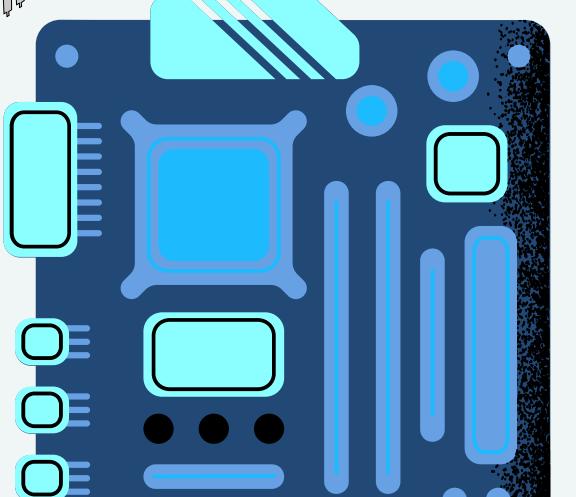
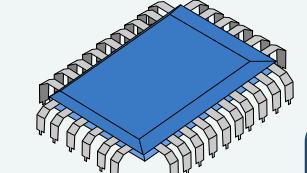
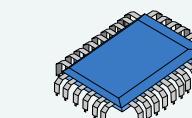
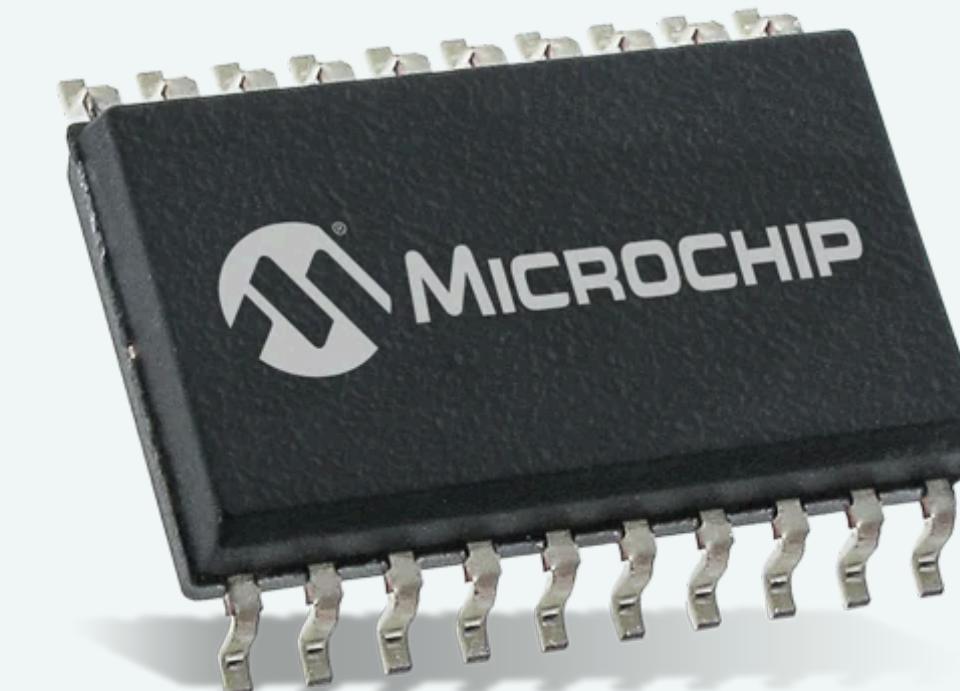
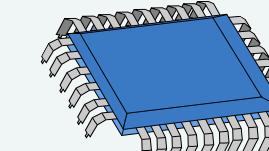
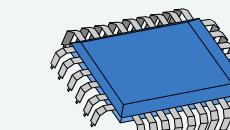
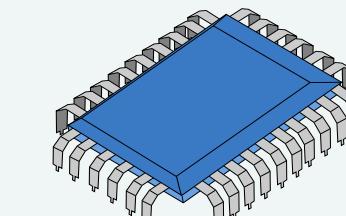
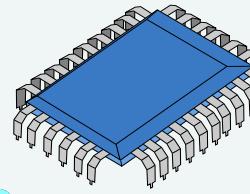
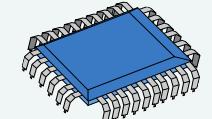
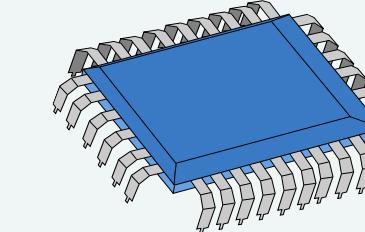
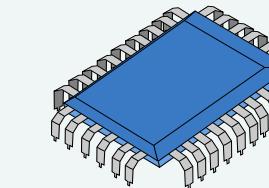
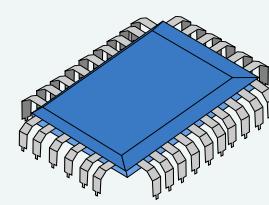
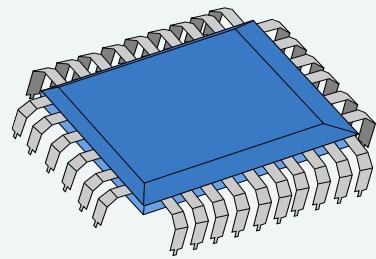


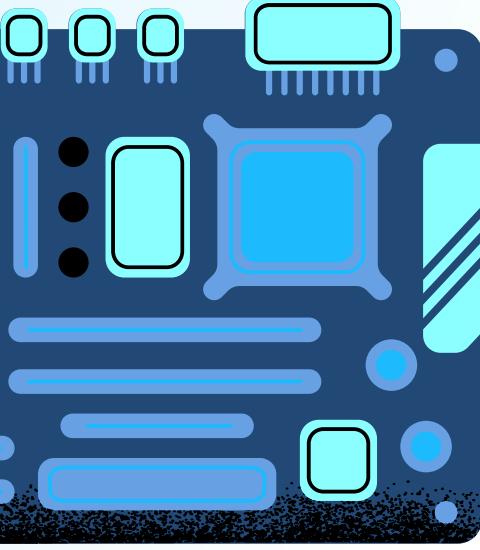


PIC (MICROCHIP)



Los microcontroladores PIC abarcan desde 8 hasta 32 bits y se destacan por su robustez y fiabilidad en entornos industriales. Se programan con el entorno MPLAB X IDE y ofrecen un control preciso de temporizadores y señales PWM. Son ideales para sistemas que requieren estabilidad a largo plazo y precisión en tareas específicas.





Comparación Visual

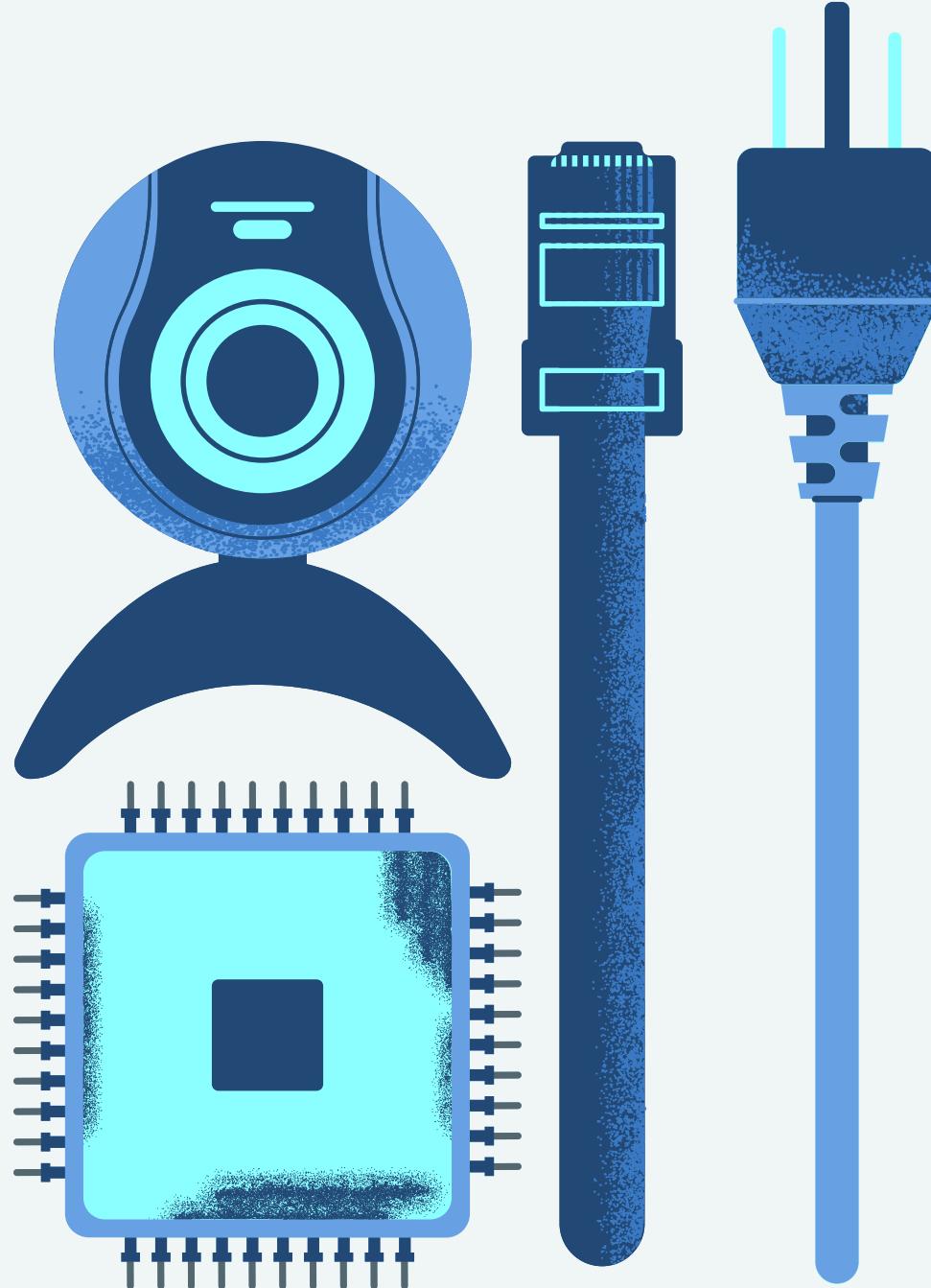
Familia	Bits	Conectividad	Facilidad	Potencia	Uso principal
AVR	8	No	★★★★☆	★☆	Educación, prototipos
ESP32	32	WiFi/BLE	★★★★	★★★★☆	IoT, domótica, robots
PIC	8- 32	Opcional	★★☆	★★☆	Automatización industrial
STM32	32	Opcional	★★	★★★★☆	Robótica avanzada



05. BUENAS PRACTICAS

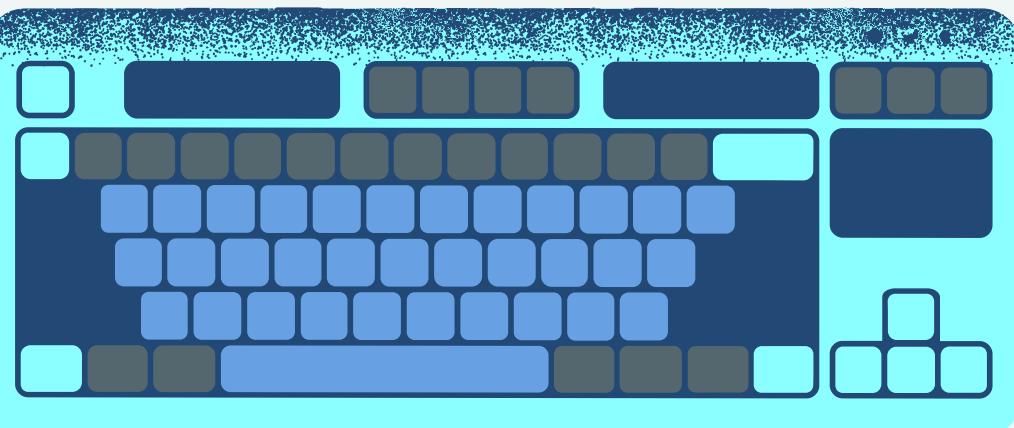
01 COMPRENSIÓN DE LAS ESPECIFICACIONES

Antes de programar o conectar cualquier componente, es crucial leer el datasheet del microcontrolador. Ahí encontrarás los voltajes de operación, capacidades de memoria, funciones especiales de pines y protocolos de comunicación que soporta. Esto evita errores como sobrecargas o conexiones incorrectas.



02 USO CORRECTO DE PINES DE ENTRADA/SALIDA

Cada pin debe definirse como entrada o salida de acuerdo al dispositivo conectado. Una buena configuración evita comportamientos erráticos y asegura lecturas fiables.



05. BUENAS PRACTICAS

03

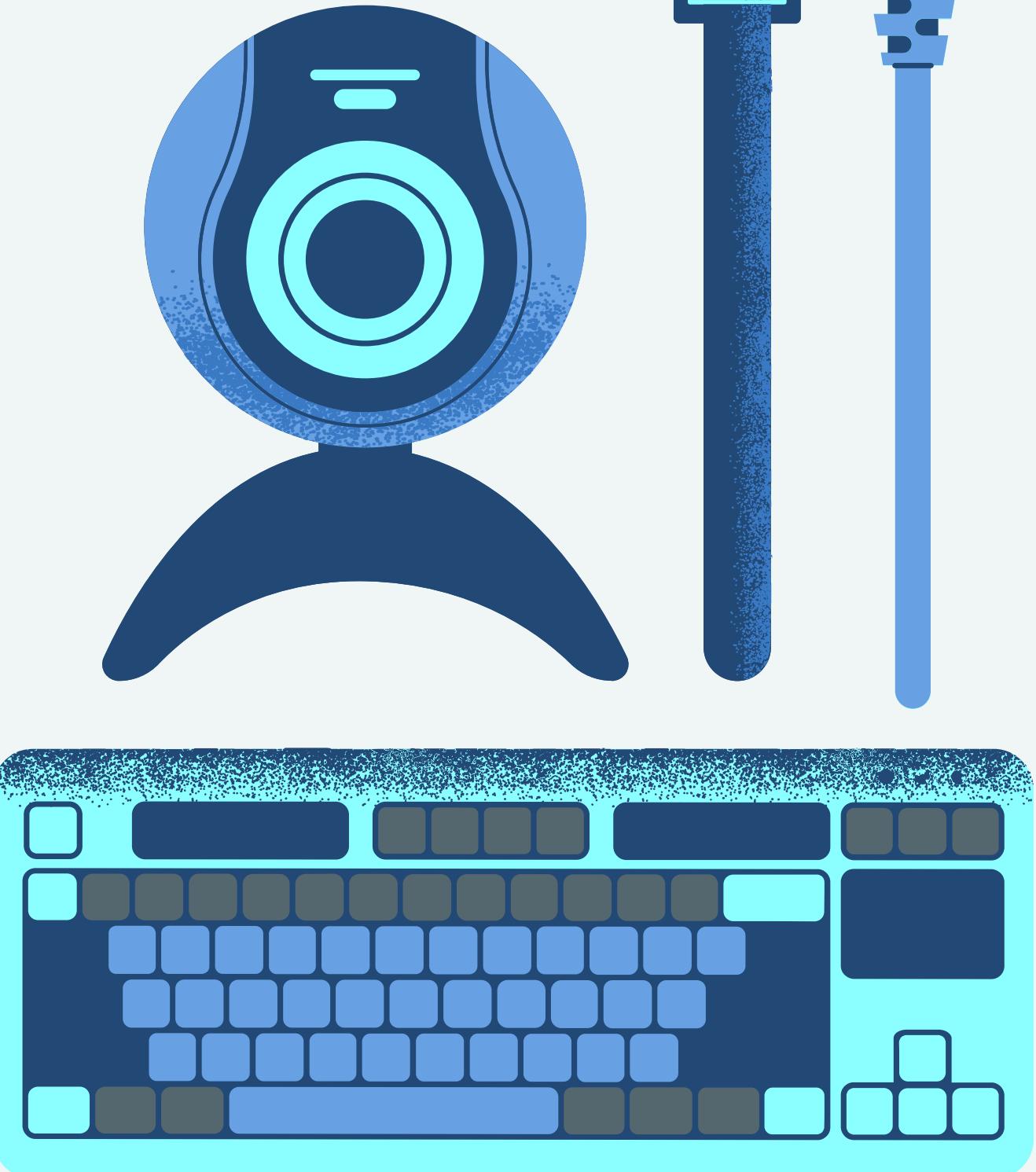
EVITAR SOBRECARGAR SALIDAS DIGITALES

Los pines del microcontrolador no están hechos para manejar corrientes grandes. Conectar directamente un motor o relé puede dañarlos. Siempre utiliza drivers, transistores o módulos intermedios que actúen como interfaz entre el microcontrolador y las cargas.

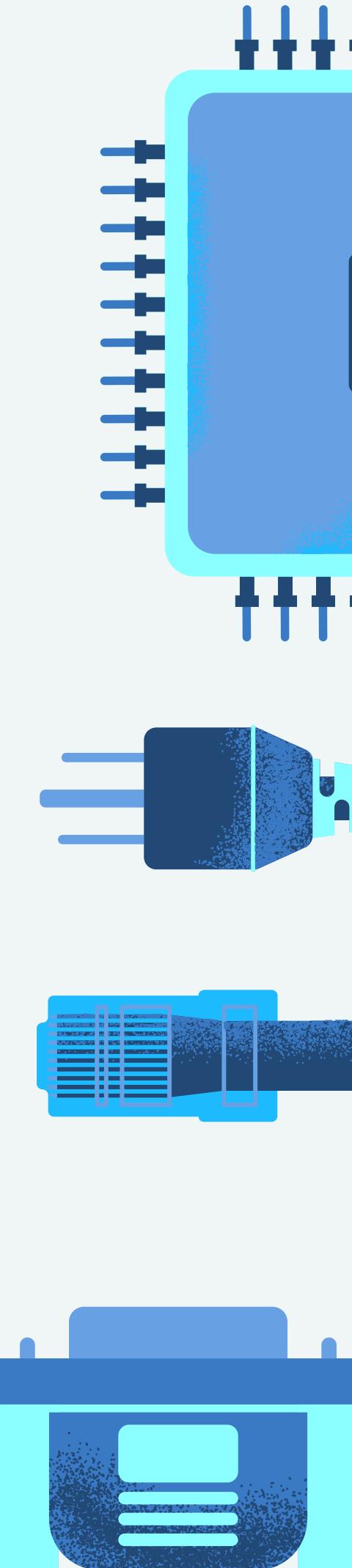
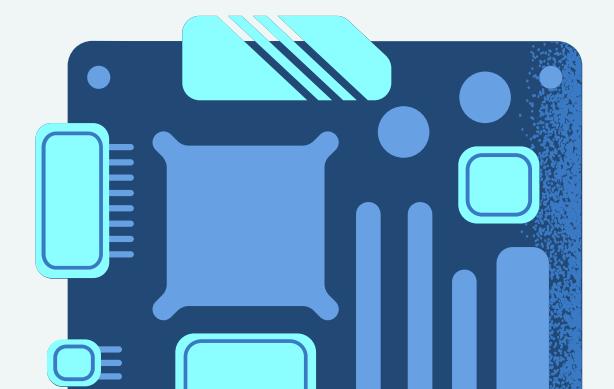
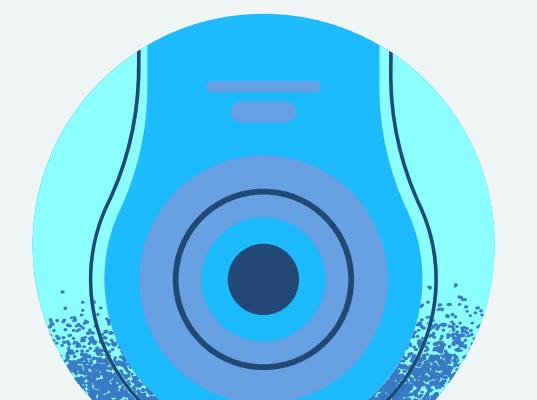
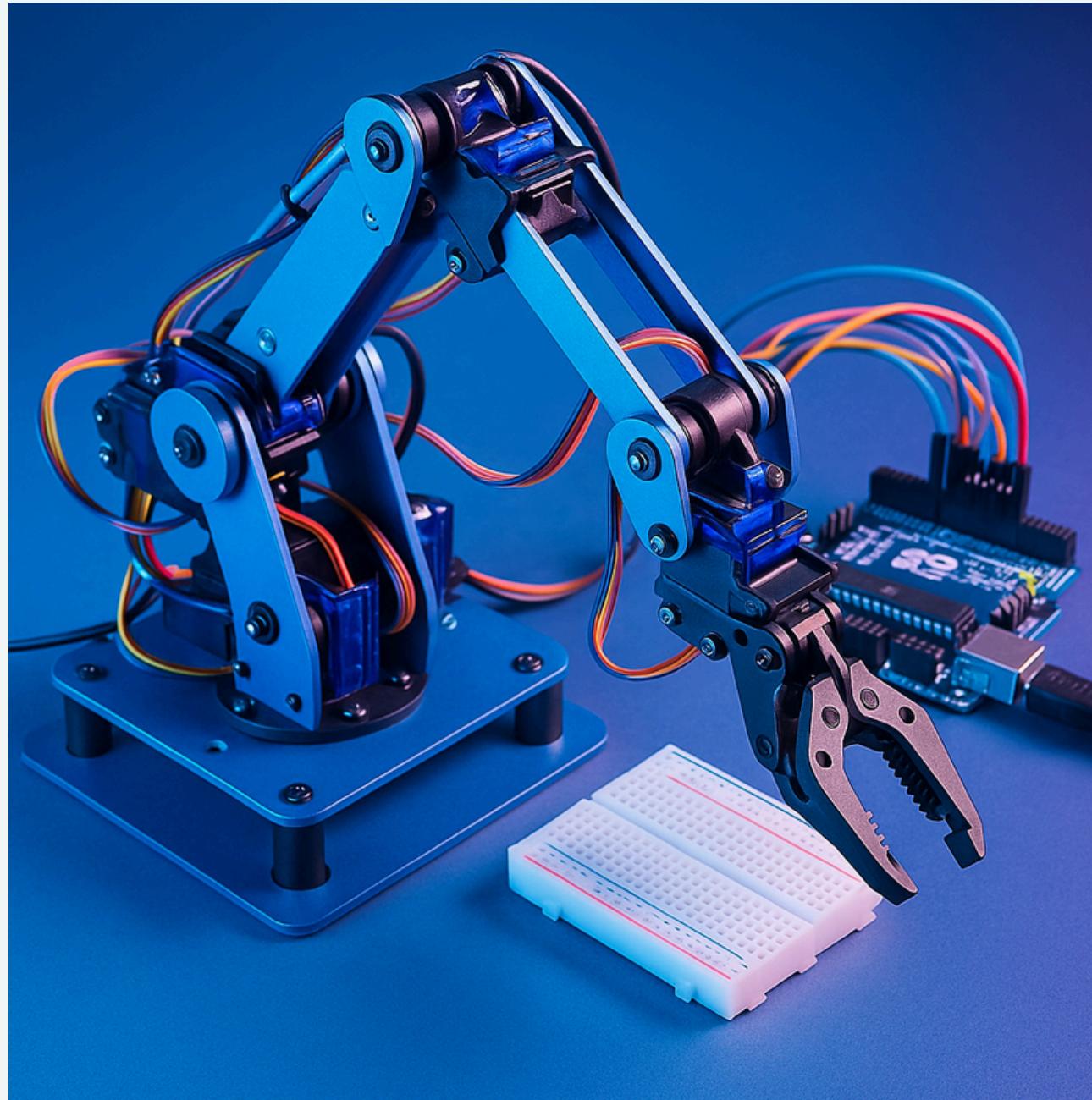
04

DOCUMENTAR Y COMENTAR EL CÓDIGO

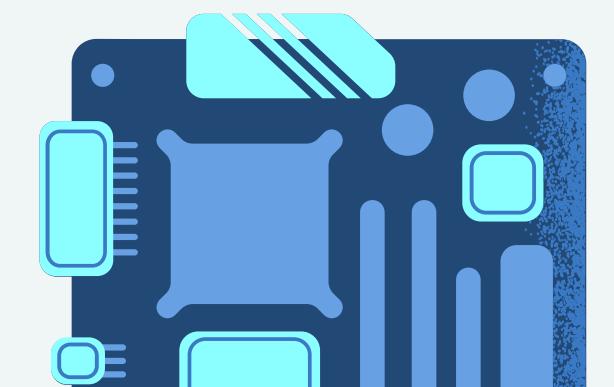
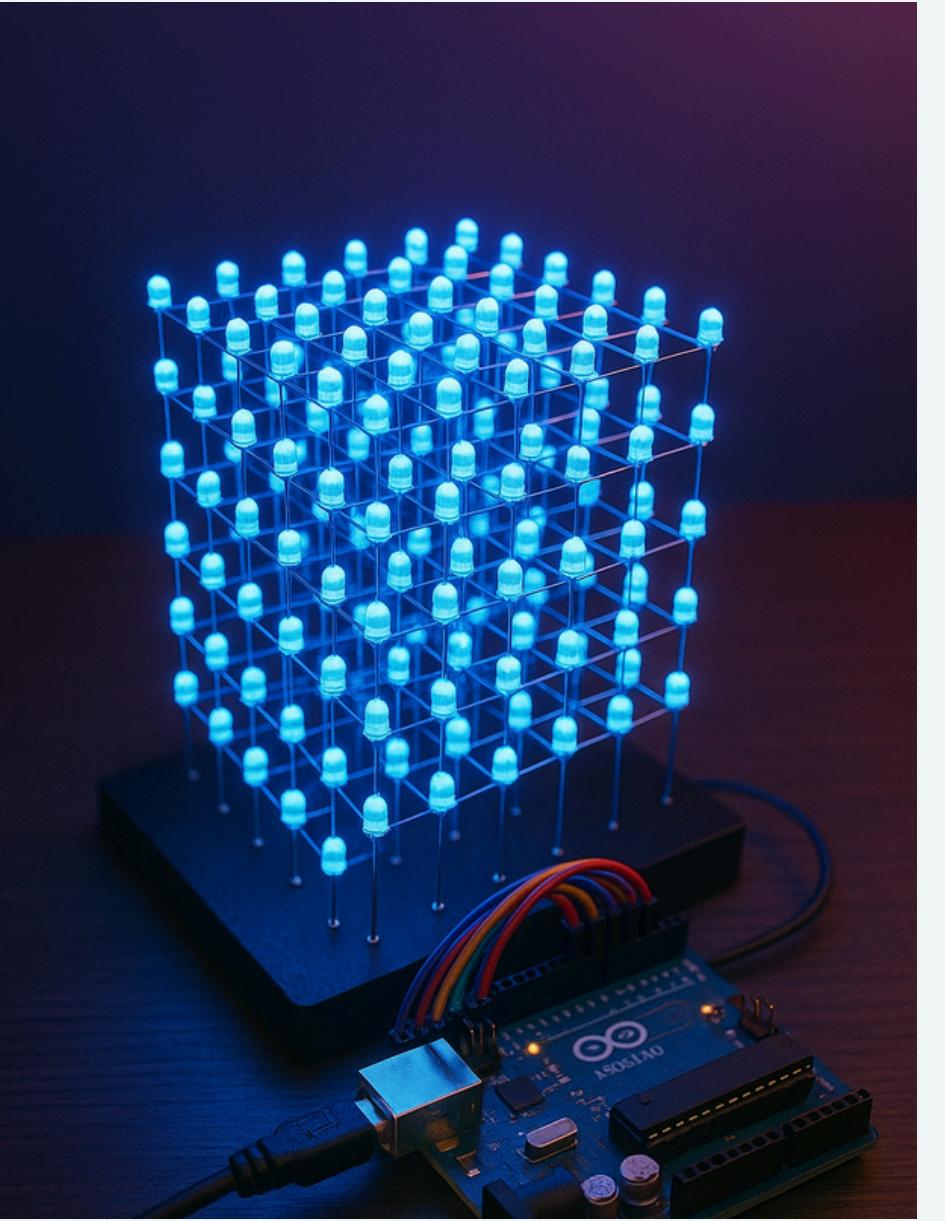
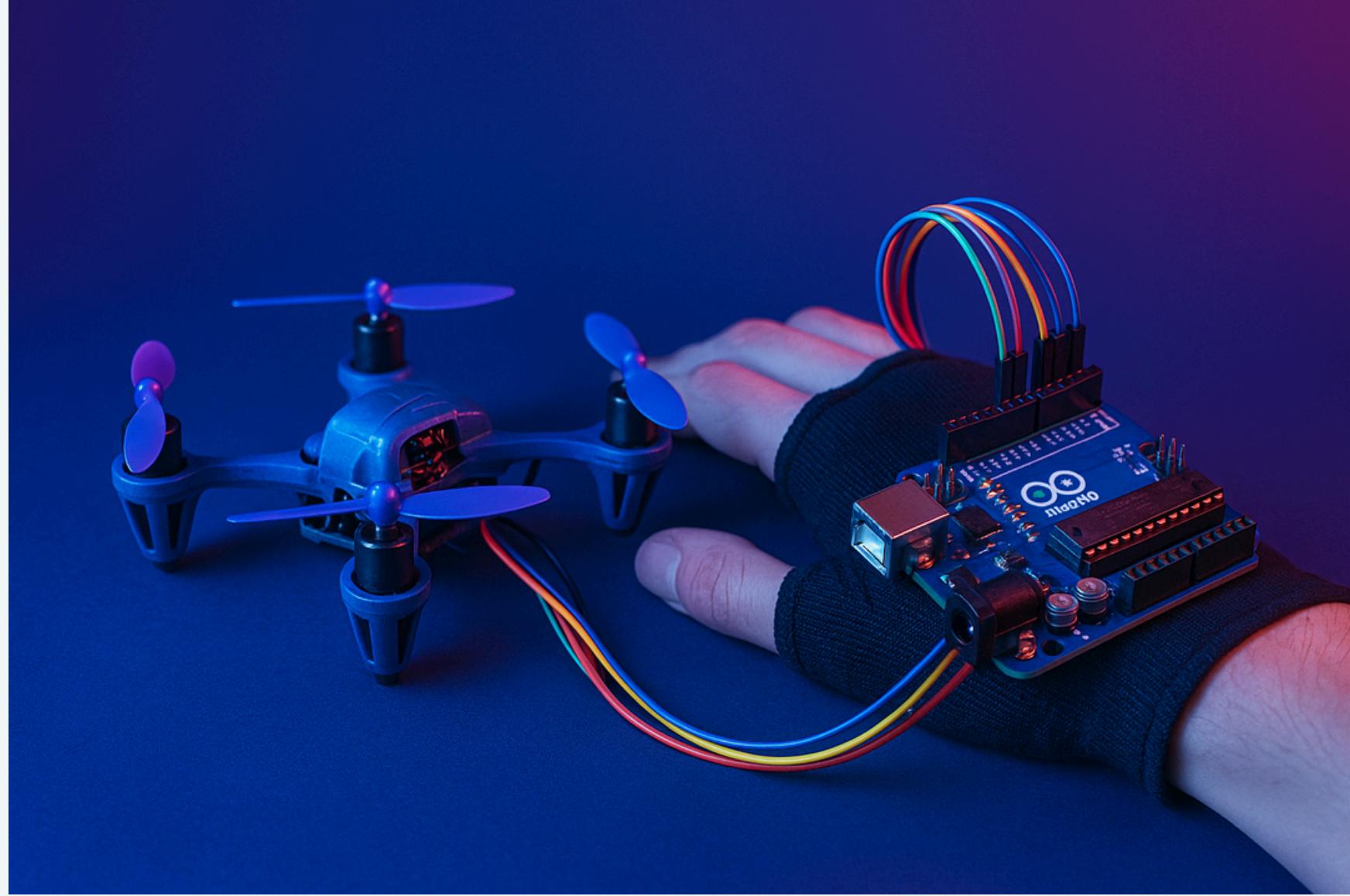
Un código claro, bien organizado y con comentarios facilita su comprensión, especialmente si trabajas en equipo o debes revisarlo tiempo después. Usar nombres de variables descriptivos y dividir el programa en funciones también mejora su mantenibilidad.



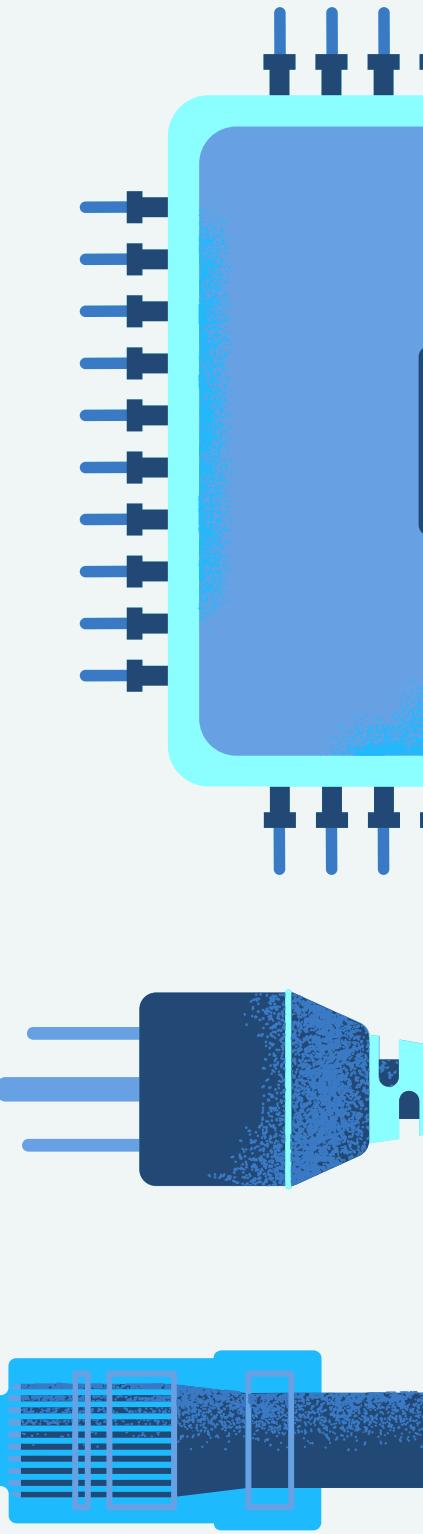
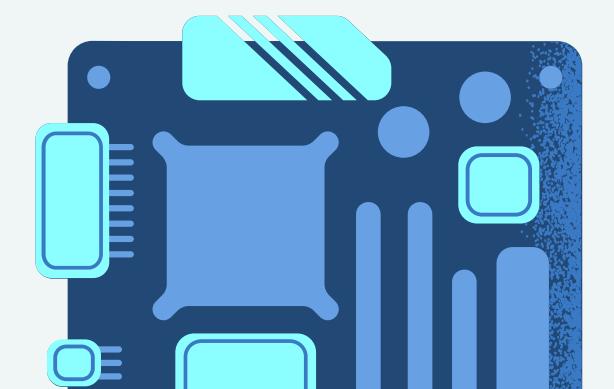
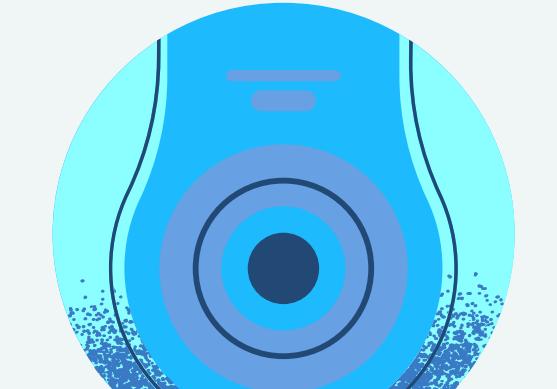
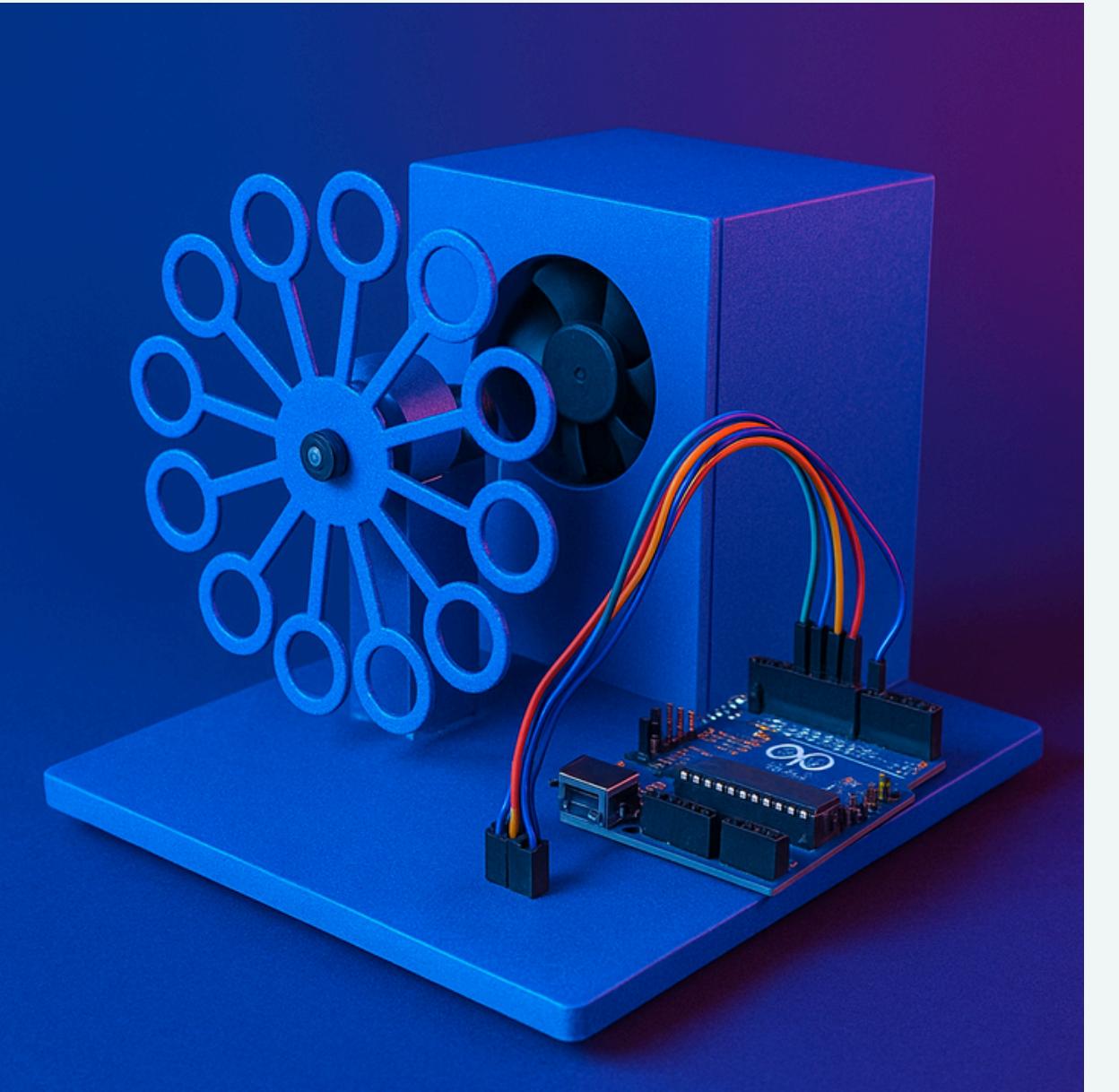
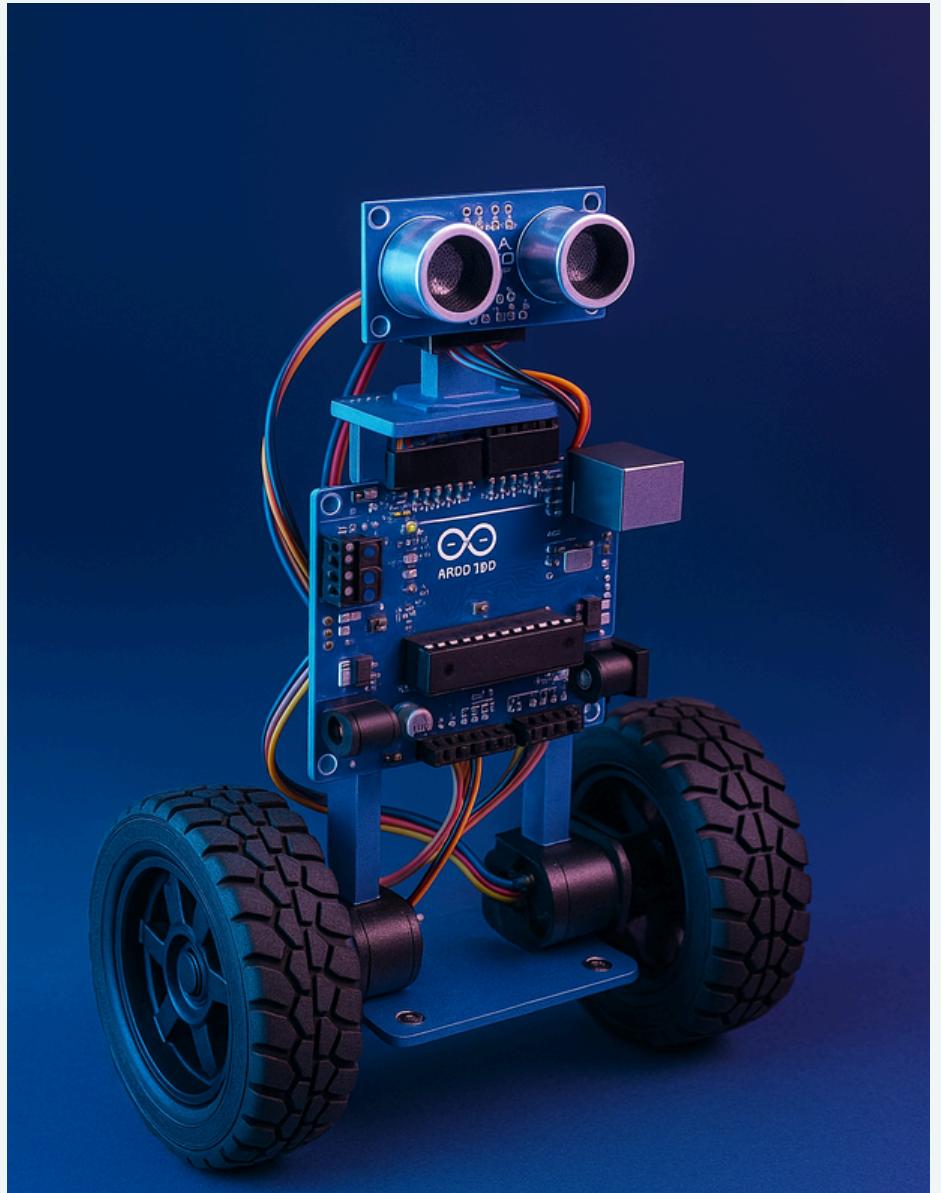
06. PROYECTOS



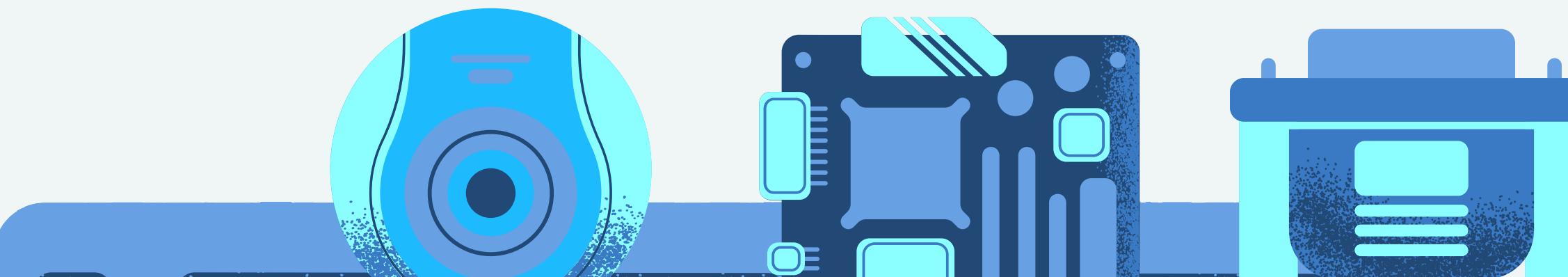
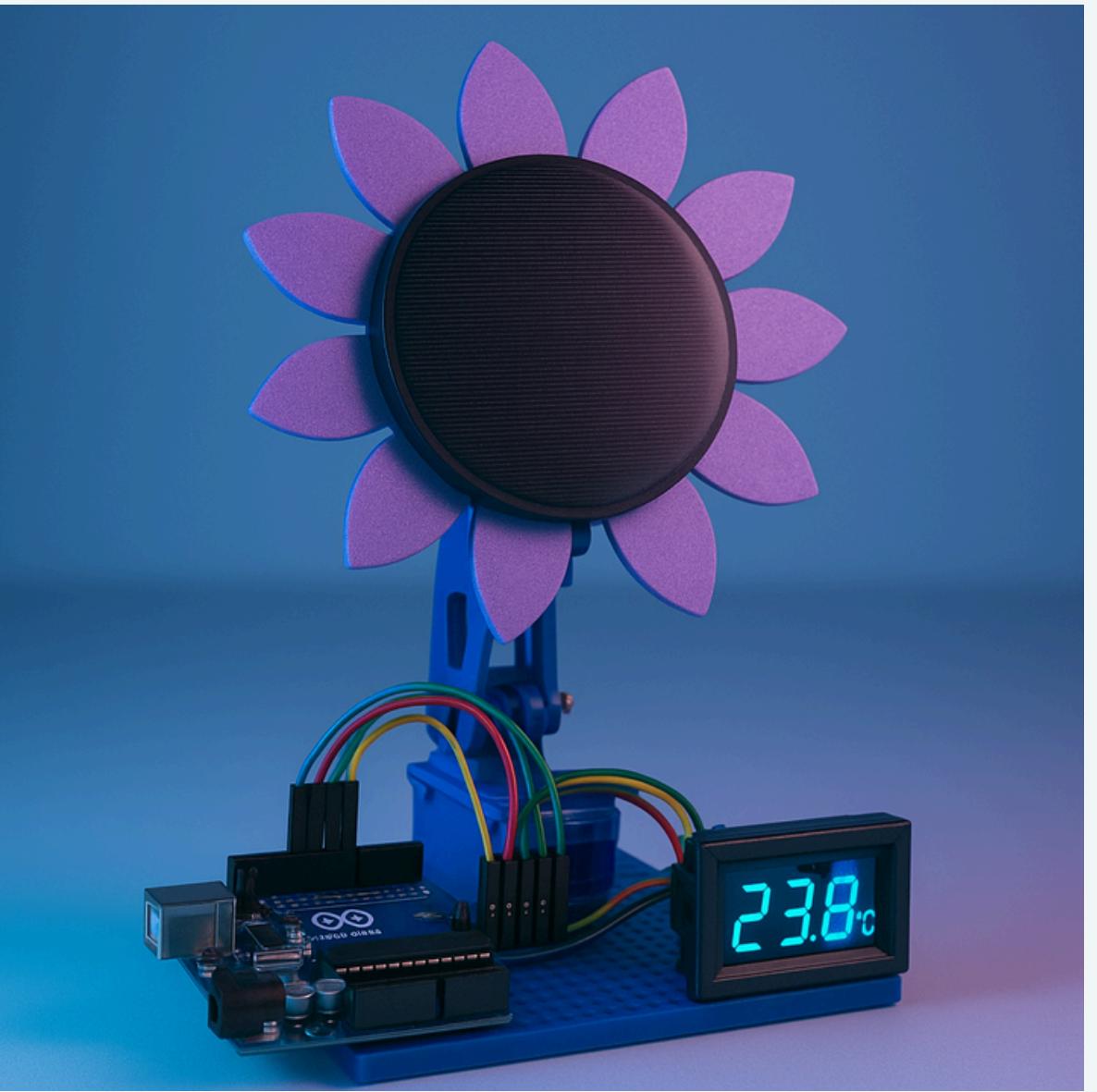
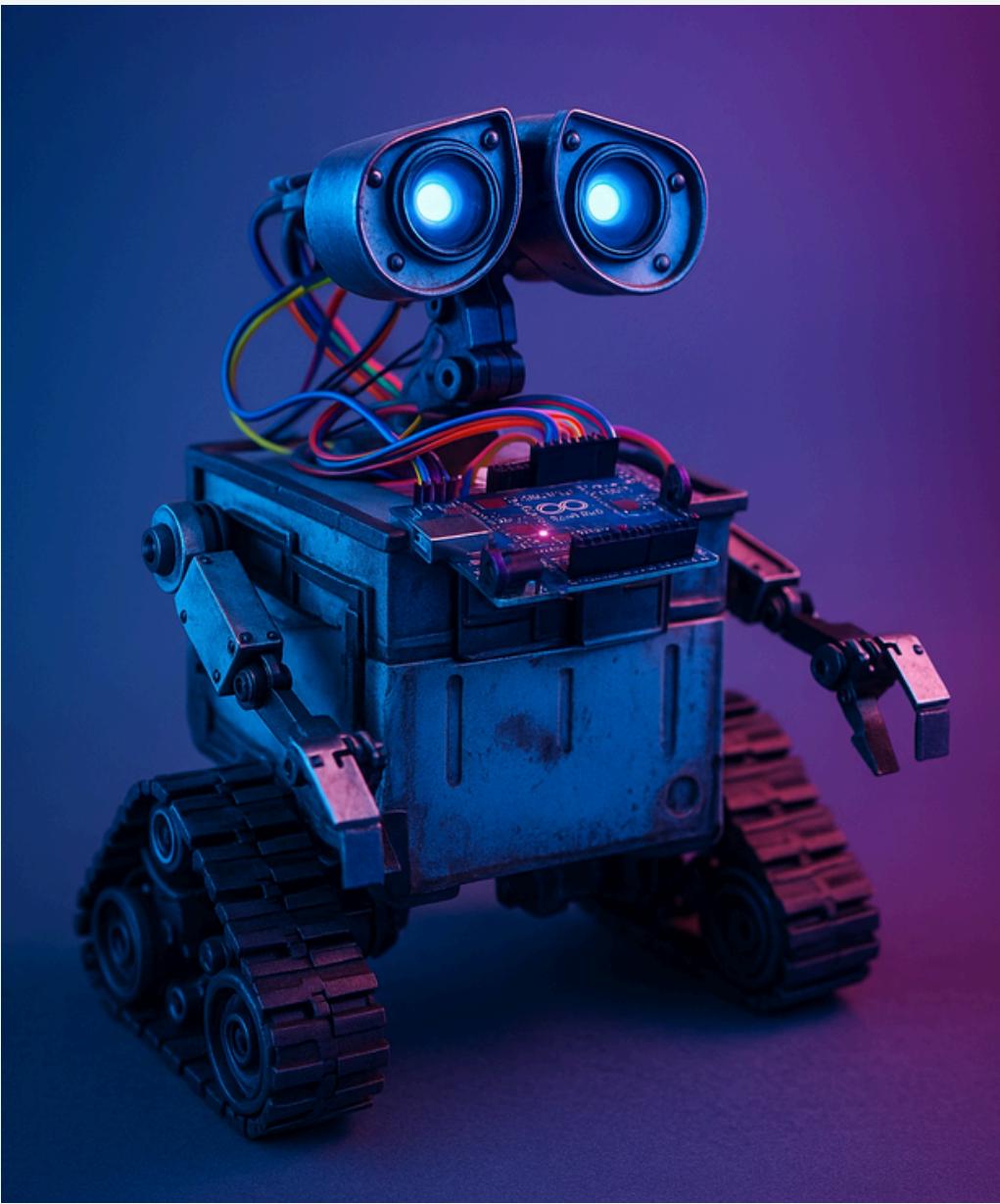
PROYECTOS



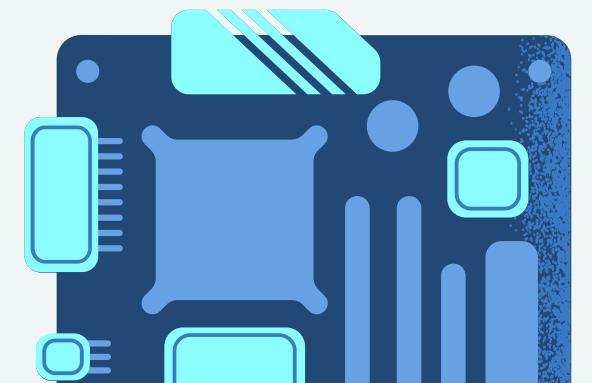
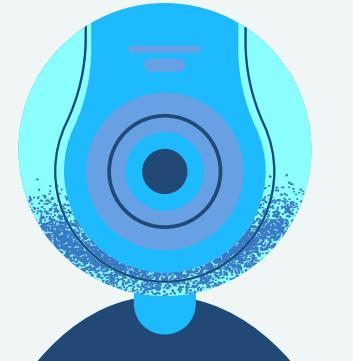
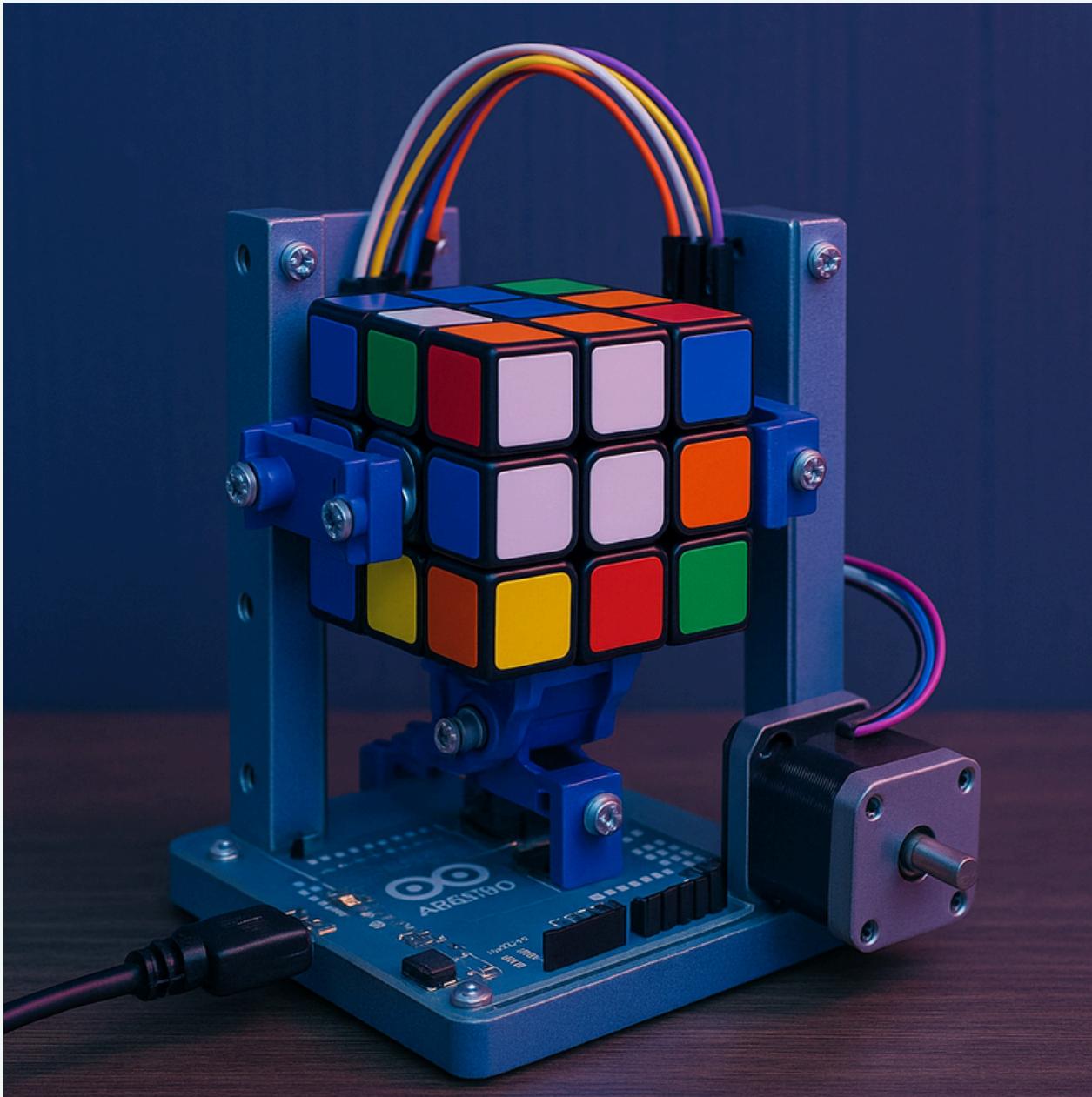
PROYECTOS



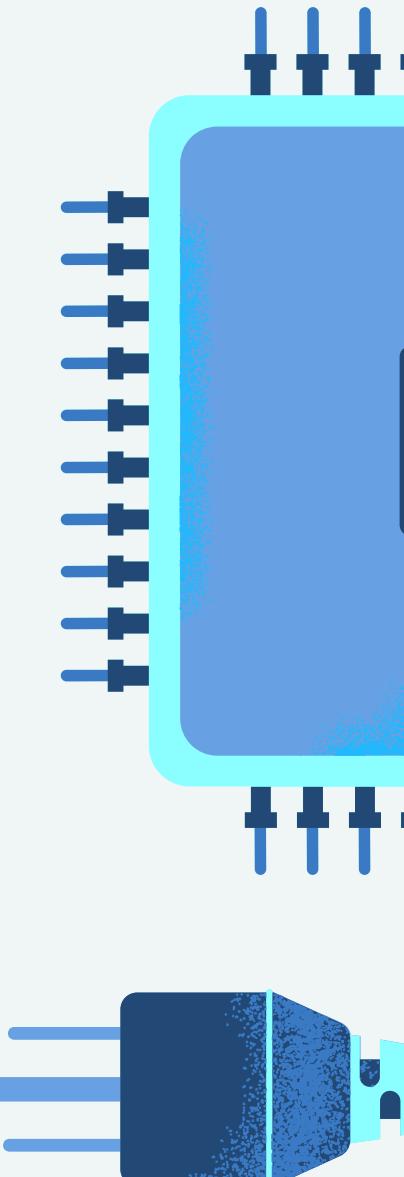
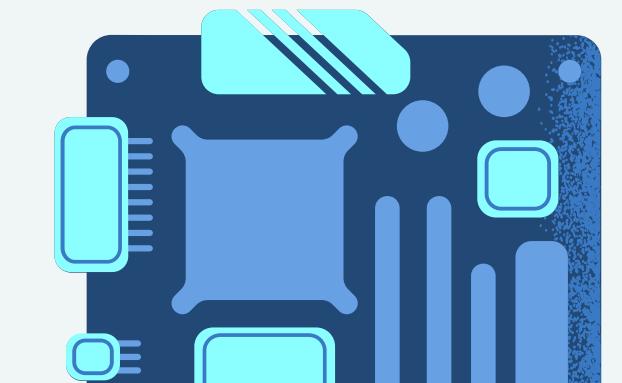
PROYECTOS



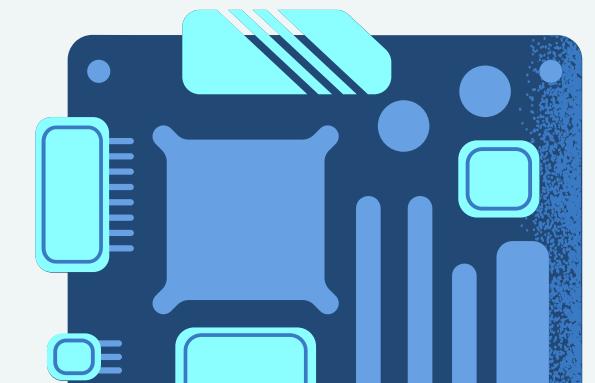
PROYECTOS



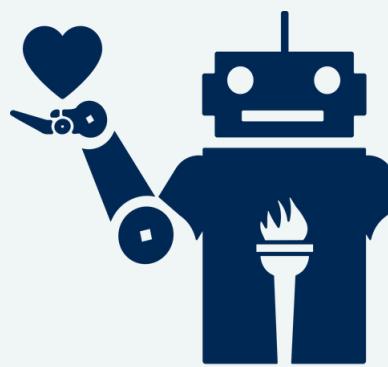
PROYECTOS



PROYECTOS



MUCHAS GRACIAS



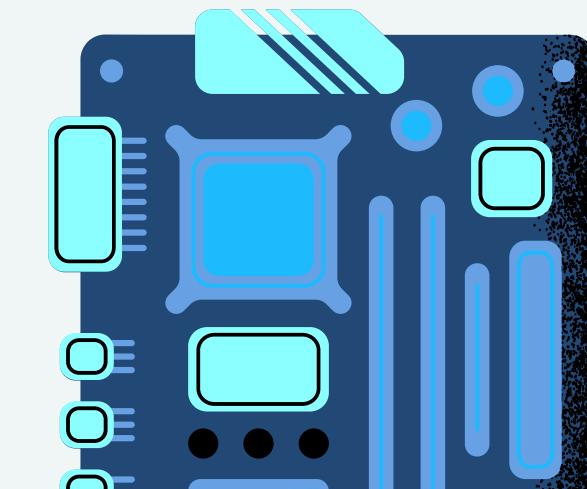
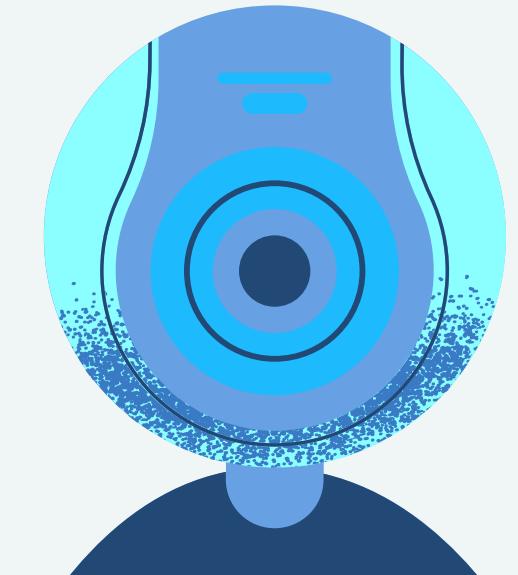
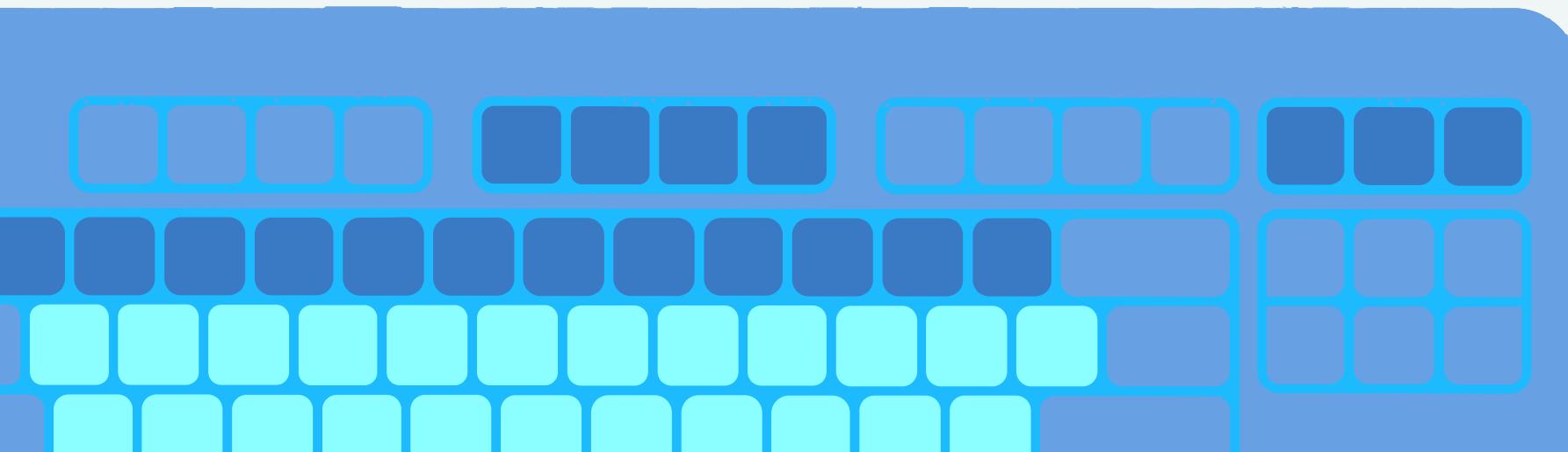
**Sociedad
de Robótica**
Universidad de Concepción



@RAS.UDEC



ras.udc@gmail.com



TÉRMINOS DE USO

Este material ha sido elaborado con fines educativos y gratuitos para el taller de introducción a los microcontroladores, organizado por la Sociedad de Robótica RAS IEEE UdeC.

Su uso está permitido únicamente para propósitos académicos, docentes o personales.

Queda prohibida su reproducción con fines comerciales sin autorización expresa.

Si deseas usar parte del contenido en otras instancias formativas, por favor solicita autorización al equipo organizador.

Puedes citar o reutilizar contenido siempre que se reconozca la autoría de forma clara.



Contacto: ras.udc@gmail.com