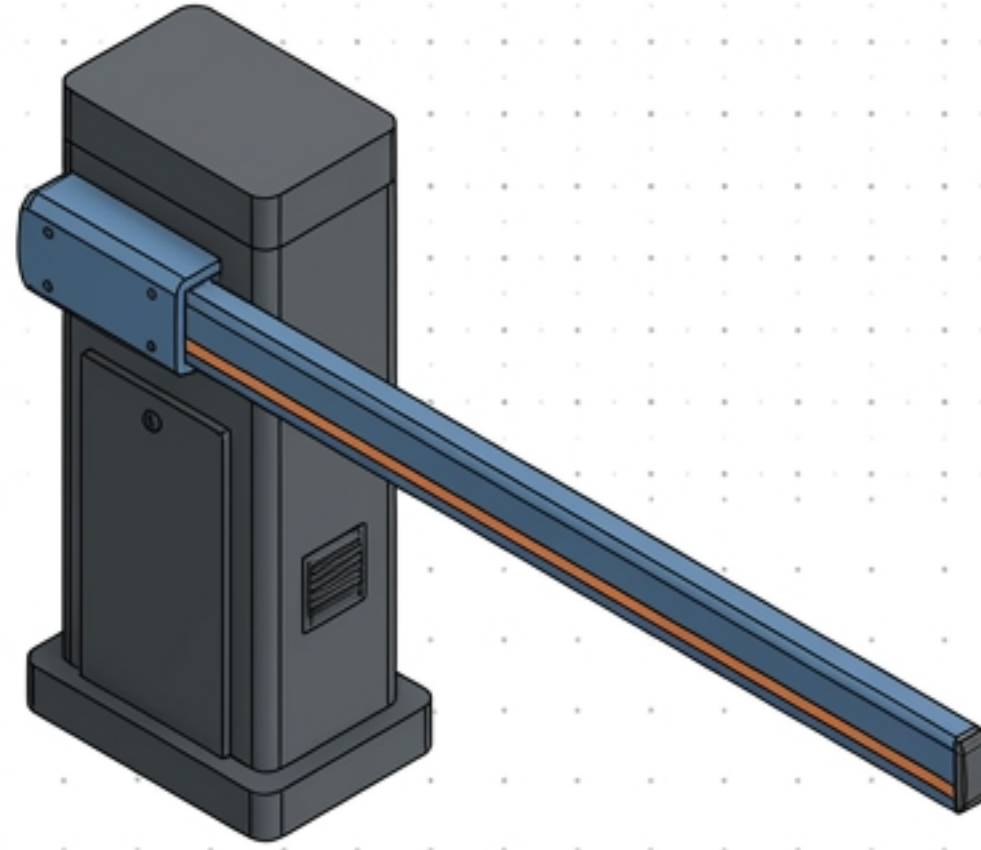
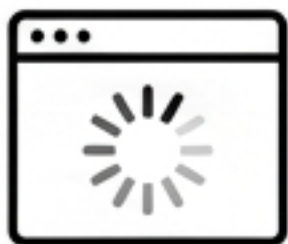




Proyecto Integrador: Sistema de Parqueadero Inteligente

Diseño, Simulación y Programación con micro:bit



Filas Virtuales

Control de tráfico web para no colapsar servidores durante alta demanda.



Turnos de Atención

Gestión del tiempo de espera y capacidad de personal en centros de servicio.



Celdas de Parqueo

Monitoreo de cupos por piso para evitar la congestión vehicular.

El denominador común: La gestión eficiente de la capacidad a través de sistemas automatizados.

El Reto Técnico

Construir una unidad de control centralizada usando micro:bit para un parqueadero automatizado.

09

Espacios Iniciales



Control de Acceso
(Entrada/Salida)



Contador en
tiempo real



Prevención de
desbordamientos



El Entorno de Pruebas: ¿Por qué Tinkercad?



Cero Riesgos Físicos

Experimentación segura sin peligro de cortocircuitos o daños a componentes reales.



Inventario Ilimitado

Acceso a un catálogo completo de sensores, actuadores y placas sin restricciones de presupuesto.



Iteración Rápida

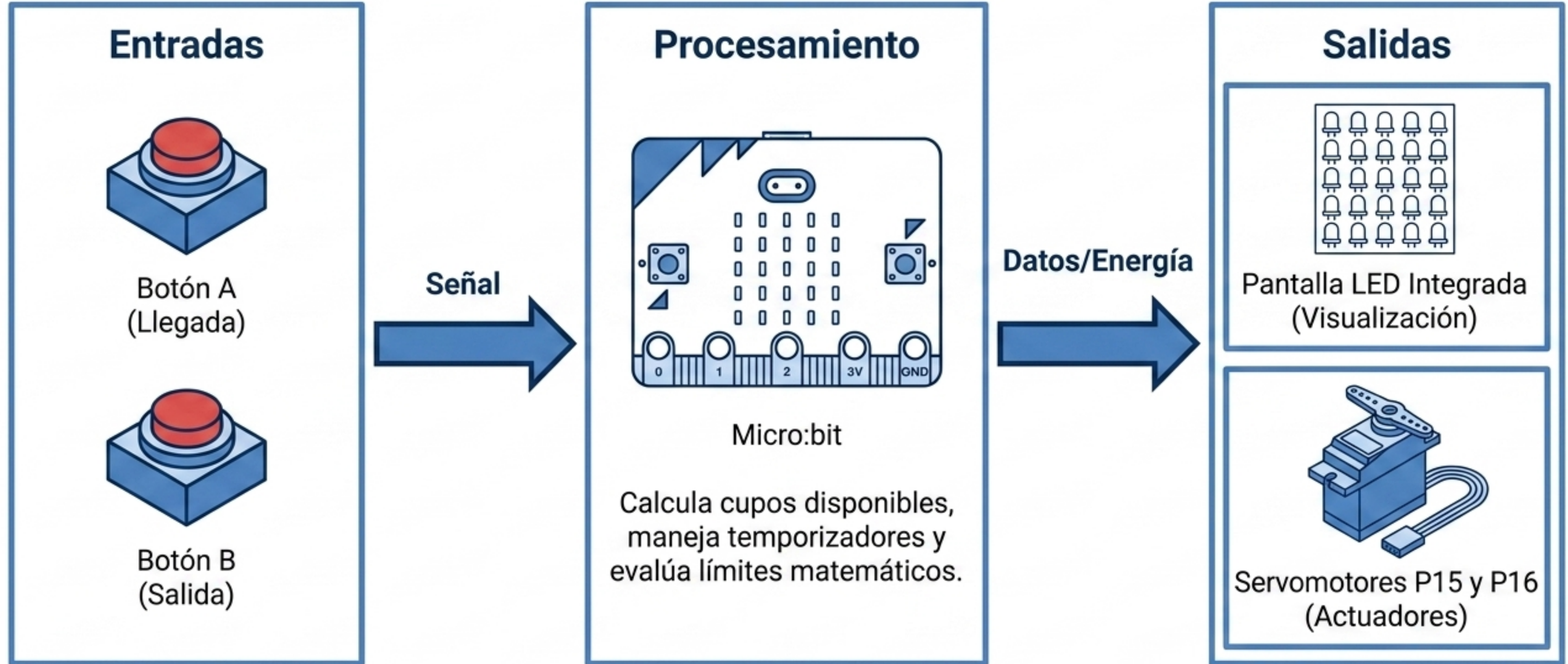
Capacidad de modificar conexiones de hardware y probar código en segundos.



Transición Fluida

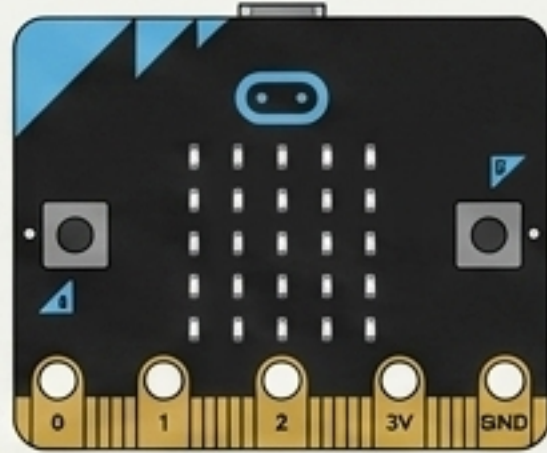
Programación híbrida que permite pasar de bloques visuales a código Python instantáneamente.

Arquitectura del Sistema



Inventario de Hardware

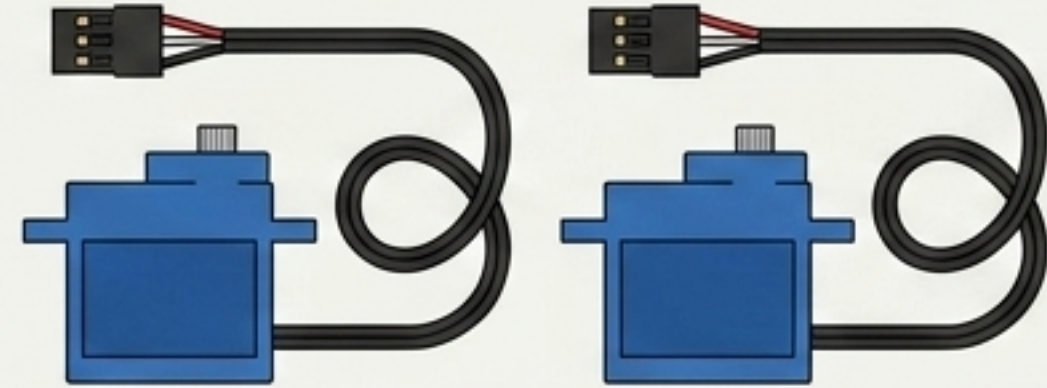
01



Micro:bit

El cerebro del sistema (Unidad de control lógico).

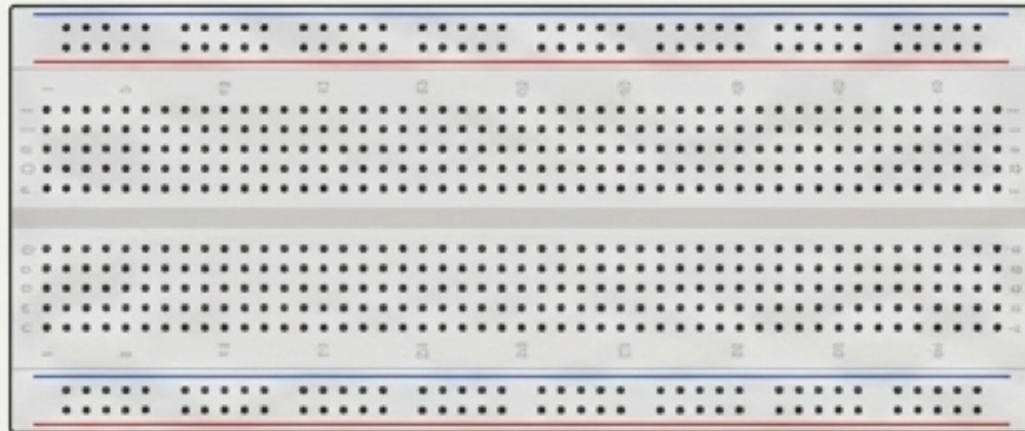
02



2x Micro-servomotores

Los actuadores físicos (Simulan las barreras mecánicas).

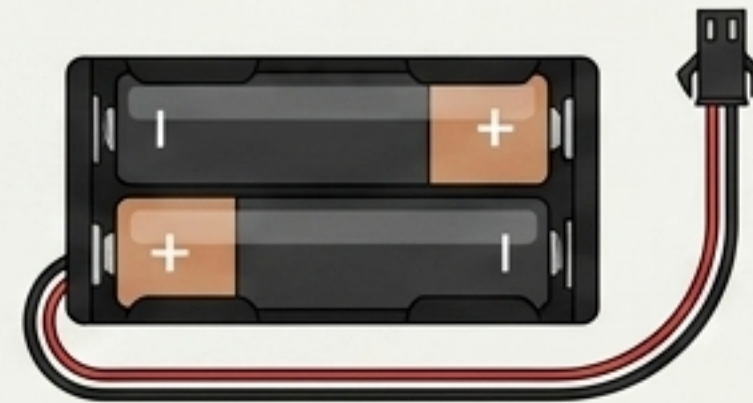
03



Placa de Pruebas

El puente de conexión y distribución eléctrica.

04



2x Baterías AA (1.5V)

El músculo (Fuente de energía independiente para los servos).

Mapeo de Conexiones

Conexión Tierra (GND)

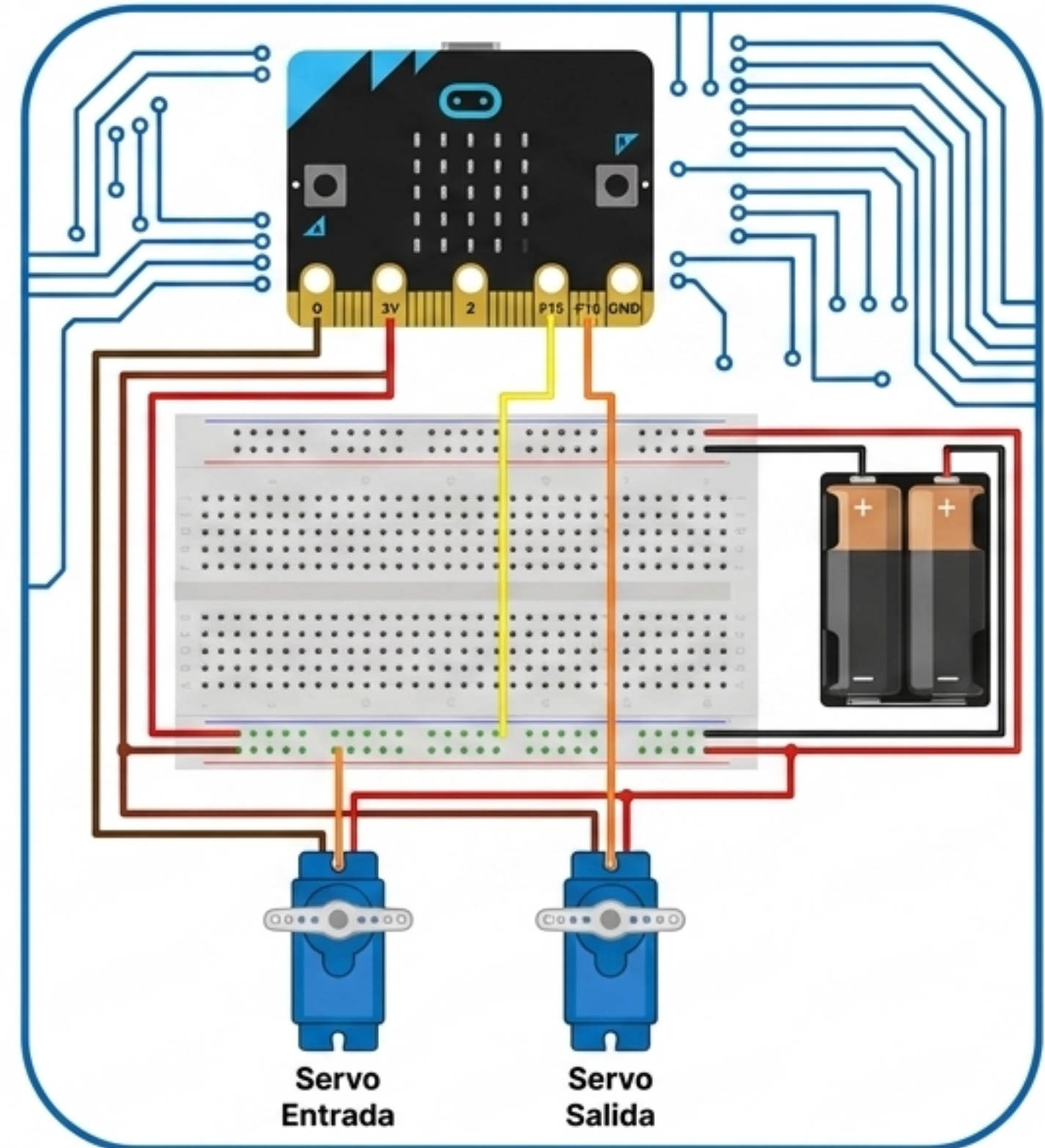
Conecta el polo negativo de las baterías y los servos al pin GND de la micro:bit.

Conexión Potencia (3V)

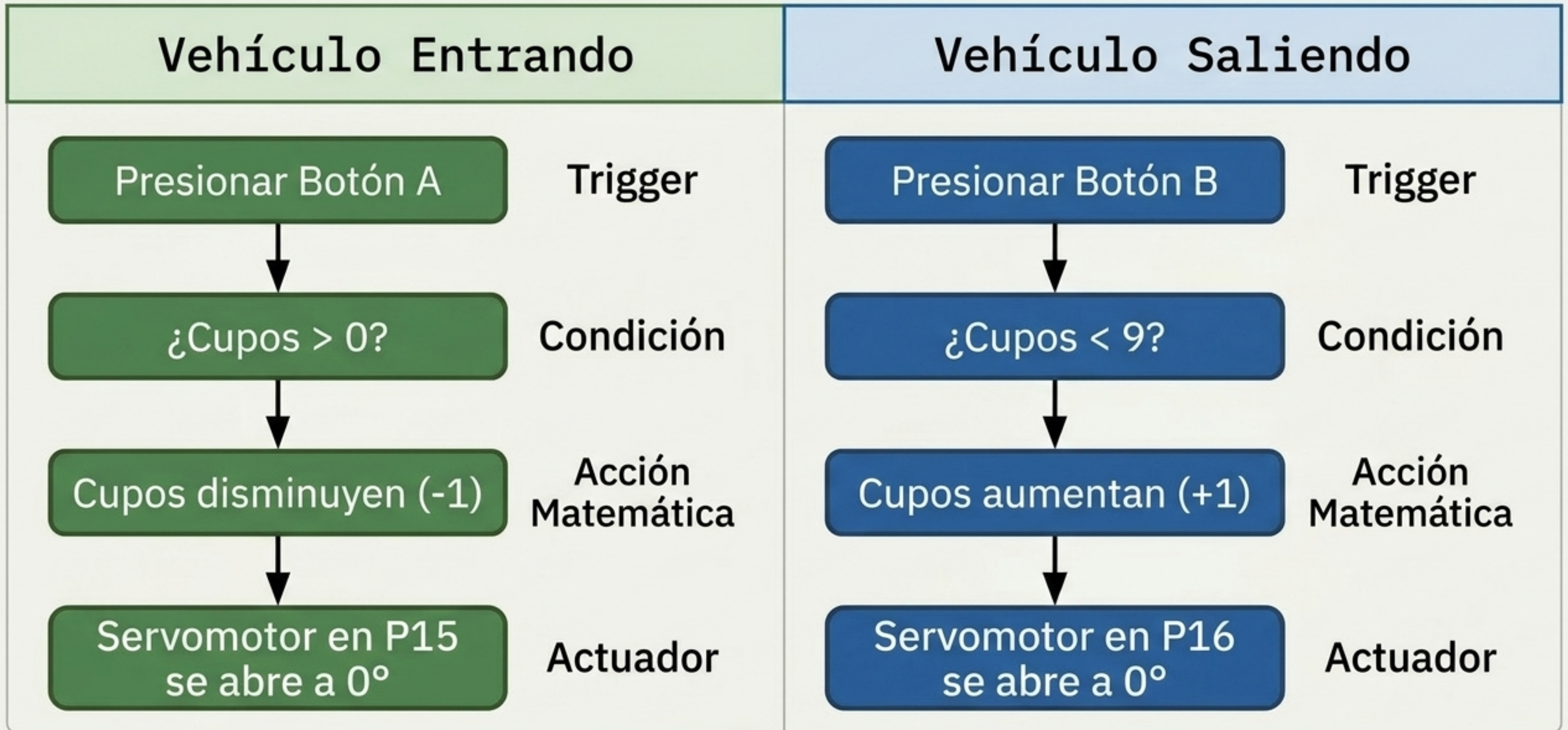
Conecta el pin de 3V de la micro:bit y el polo positivo de las baterías.

Conexión Señal (Datos)

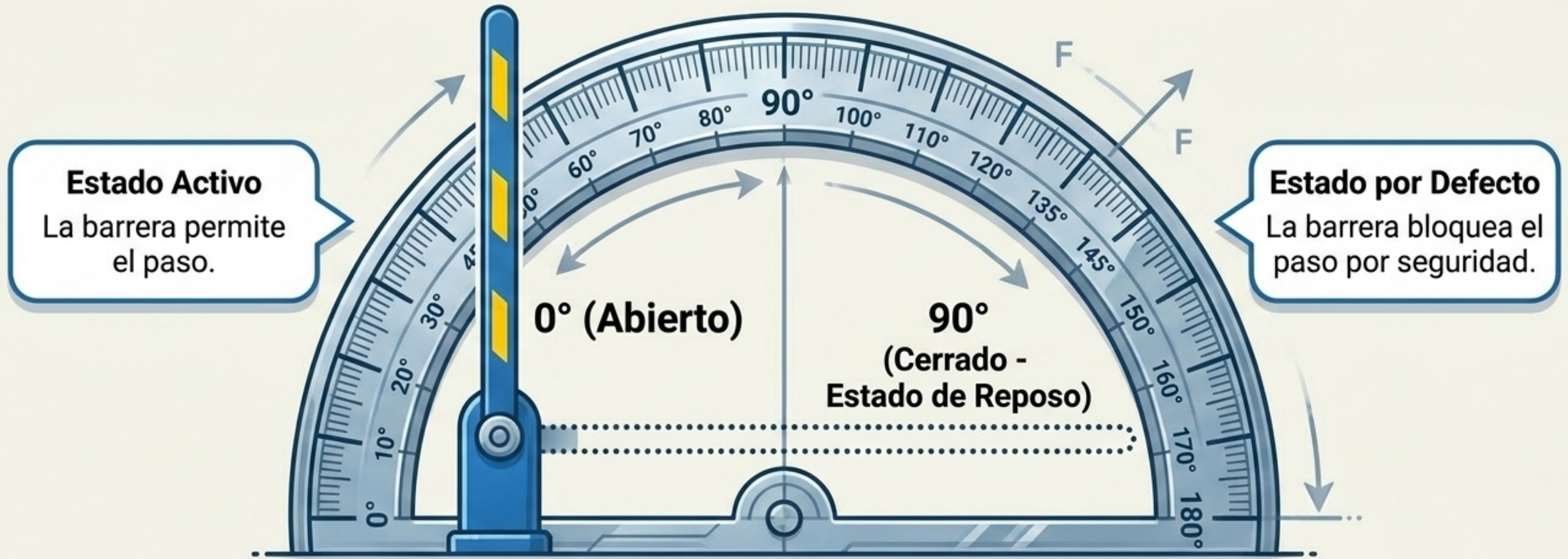
El servo de entrada al Pin P15.
El servo de salida al Pin P16.



Lógica de Control: Entrada vs. Salida



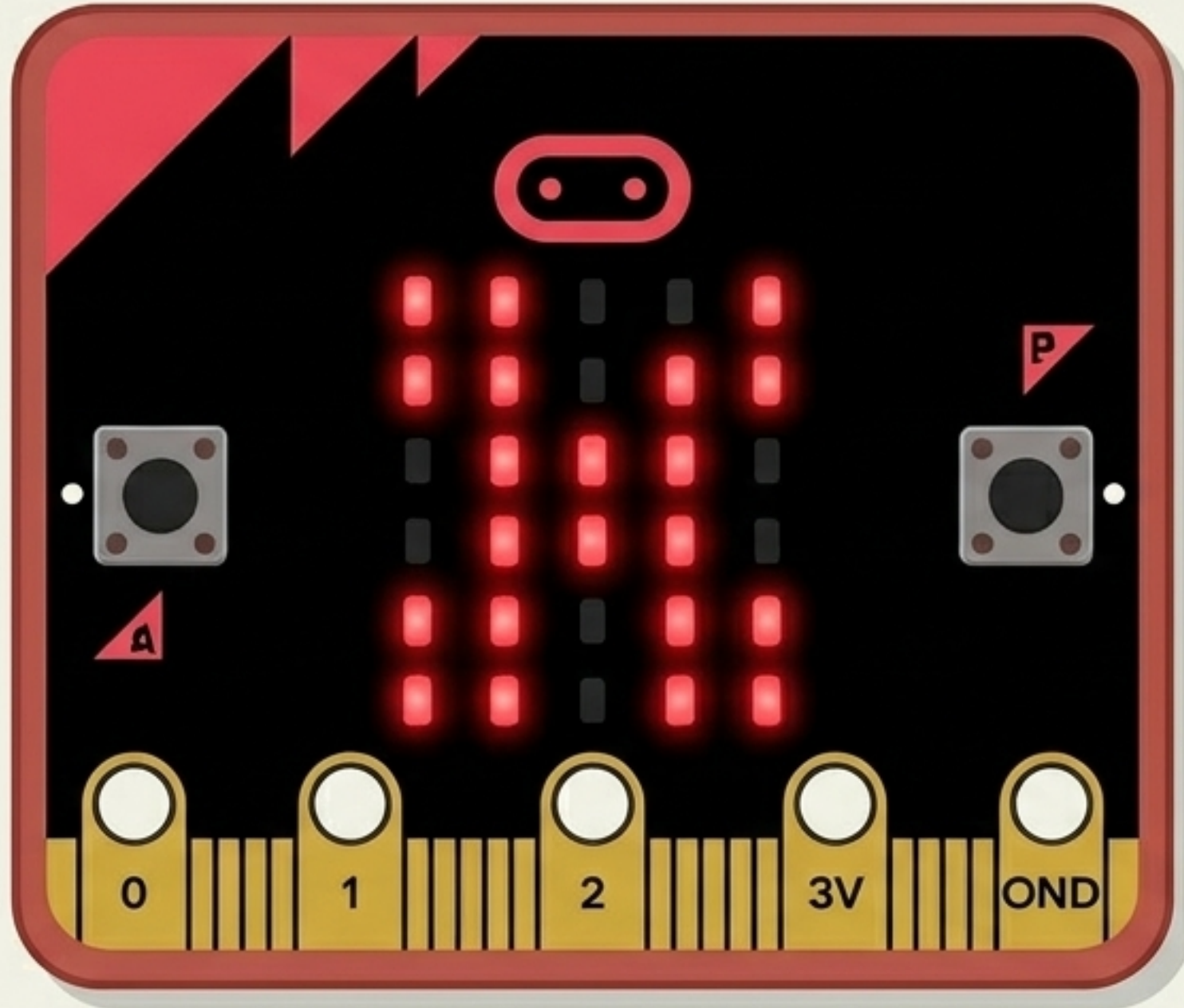
Física de las Barreras (Control de Servos)



Proceso de Retorno: El sistema espera exactamente 5 segundos (5000 ms) antes de enviar la instrucción de volver a 90°.

`basic.pause(5000)`

Casos Límite y Manejo de Errores

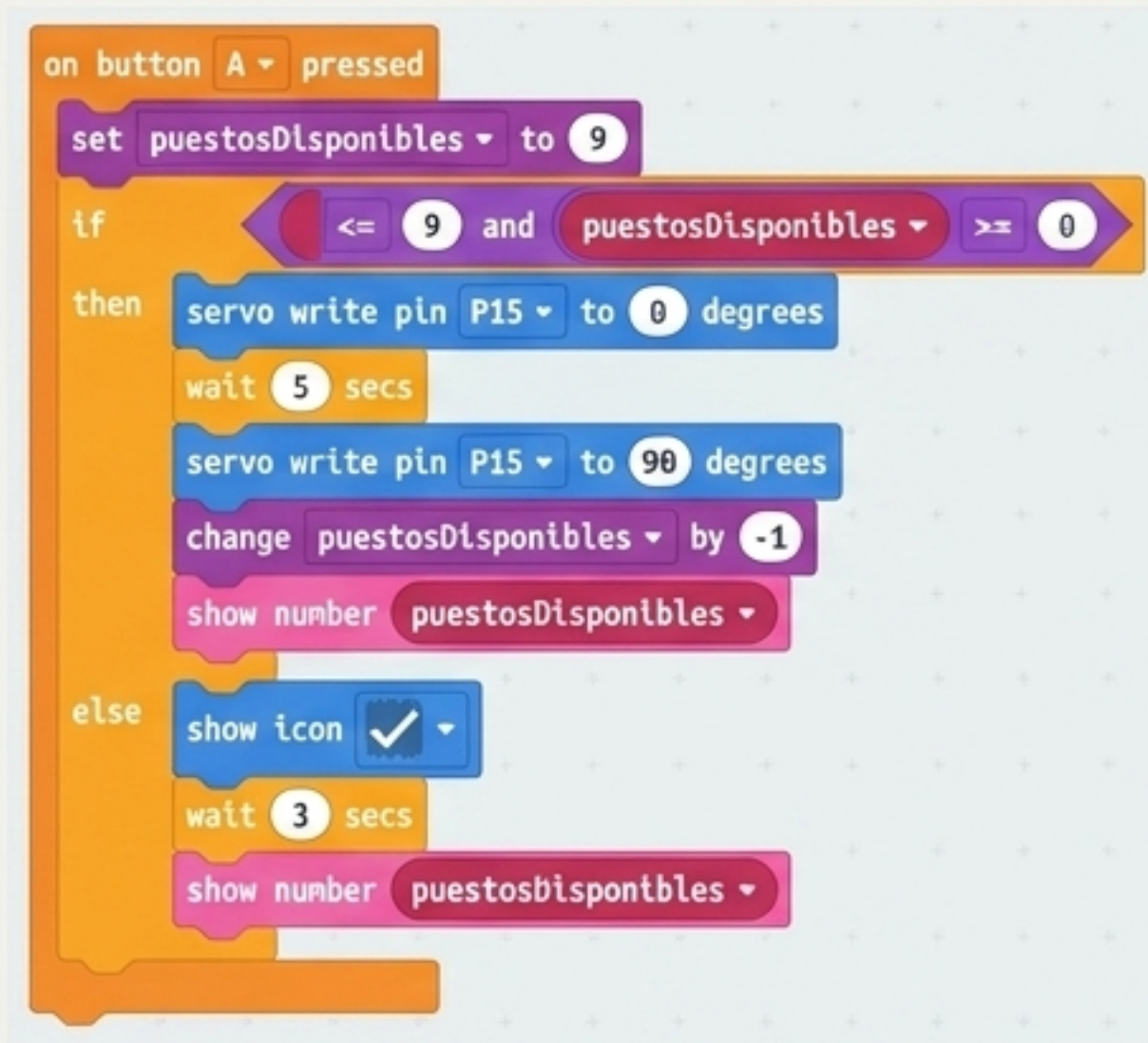


El Problema

¿Qué ocurre si un auto intenta entrar a un parqueadero lleno (0 cupos) o salir de uno vacío (9 cupos)?

Solución de Excepción

- ➡ **Bloqueo Mecánico:** Se detiene la orden de apertura a los pines P15 y P16.
- ➡ **Alerta Visual:** Despliega el ícono `IconNames.No` ('X') en la matriz LED durante 3 segundos.
- ➡ **Restauración:** Limpia la alerta y vuelve a mostrar el número de cupos actuales.



Bloques Visuales

Lógica gráfica intuitiva basada en encajes. Ideal para estructurar el diagrama de flujo mental de los eventos.

```
def on_button_pressed_a():  
    global puestosDisponibles  
    if puestosDisponibles <= 9 and puestosDisponibles > 0:  
        pins.servo_write_pin(AnalogPin.P15, 0)  
        basic.pause(5000)  
        pins.servo_write_pin(AnalogPin.P15, 90)  
        puestosDisponibles += -1  
        basic.show_number(puestosDisponibles)  
    else:  
        basic.show_icon(IconNames.No)  
        basic.pause(3000)  
        basic.show_number(puestosDisponibles)  
  
input.on_button_pressed(Button.A, on_button_pressed_a)
```

Código Python

Sintaxis profesional basada en texto. Permite el control preciso de variables y llamadas a funciones.

El Código: Inicialización del Sistema

IDE

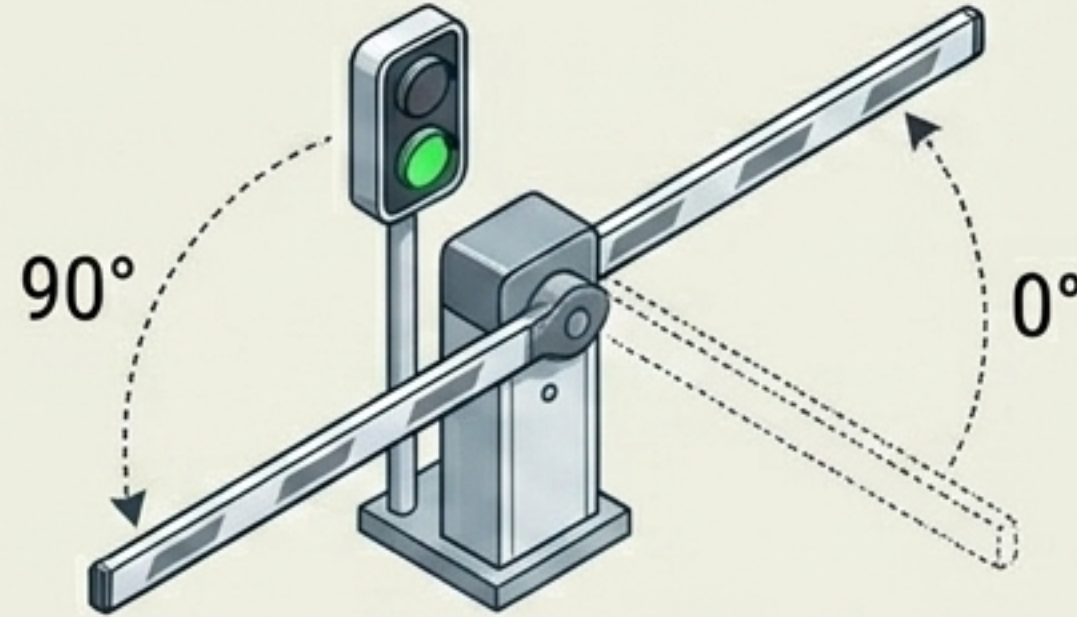
Declaración de la variable global (El estado inicial de capacidad del parqueadero).

```
puestosDisponibles = 9
basic.show_number(puestosDisponibles)
pins.servo_write_pin(AnalogPin.P15, 90)
pins.servo_write_pin(AnalogPin.P16, 90)
```

Reflejar el inventario inicial de cupos en la pantalla LED de la micro:bit.

Posicionar ambas barreras mecánicas en estado cerrado (90°) por seguridad desde el inicio.

Reto Avanzado: Señalización Visual



Barrera Abajo (90°)		LED Rojo ENCENDIDO / LED Verde APAGADO.
Barrera Arriba (0°)		LED Verde ENCENDIDO / LED Rojo APAGADO.

Concepto: Integrar 4 LEDs adicionales al circuito (un par Rojo/Verde por cada barrera) mejora la experiencia y seguridad del usuario.

Análisis de Riesgos: Del Simulador a la Realidad

Falla Técnica	Acción en el Circuito	Efecto Físico Real
 Cortocircuito	Conectar tierra y alimentación directa al mismo polo.	Sobrecalentamiento, daño permanente de la placa micro:bit.
 Sobrecarga	Agregar 4 baterías en lugar de 2.	Superar el voltaje tolerado, quemando los servomotores irreversiblemente.
 Drenaje de Corriente	Conectar los servomotores directamente a la micro:bit sin energía externa.	La placa se reinicia o se apaga por incapacidad de suplir la demanda de potencia mecánica.

Síntesis: Verificación del Sistema

- [X] Manipulación de Variables:** Lógica matemática implementada correctamente en Python (puestosDisponibles).
- [X] Simulación Segura:** Circuito ensamblado en el simulador sin riesgos eléctricos o daños de hardware.
- [X] Configuración de Pines:** Puertos virtuales (P15 y P16) mapeados y calibrados a los ángulos correctos.
- [X] Manejo de Excepciones:** Prevención exitosa programada para evitar cupos negativos o sobrecupo.

El plano está completo. El sistema de gestión de capacidad está listo para ser implementado en el entorno físico.