



PROYECTO N° 4: Puerta de parking automática

Aprende a construir y programar una puerta de parking con sistema de apertura y cierre automático utilizando los componentes del Maker Control Kit y el Maker Kit 3.

Abrirás la puerta presionando el pulsador. Luego, el sensor de distancia comprobará que el vehículo ya no se encuentra en la entrada y cerrará la puerta. Si el vehículo aún se encuentra en la entrada, la puerta se mantendrá abierta.

NIVEL DE DIFICULTAD: Principiante.

DURACIÓN DEL EJERCICIO: 60 min.

MATERIALES:

- 1 Botón
- 1 Sensor de distancia por ultrasonidos
- 1 Servomotor
- 1 Caja de cartón
- 1 Placa controladora Build&Code 4in1
- 1 Cable USB - Micro USB
- 1 Ordenador
- Material para la estructura de la puerta
- Adhesivo o cinta adhesiva

CONSTRUCCIÓN DE LA ESTRUCTURA:

Usarás una caja de cartón para construir la puerta y una lámina de cartón pluma para construir la pared que sujetará el botón, el sensor de distancia por ultrasonidos y la placa controladora Build&Code 4in1. [Descarga la guía rápida de montaje](#) y sigue los pasos indicados para montar la estructura.

Una vez montada la estructura, conecta todos los elementos electrónicos a la placa controladora Build&Code 4in1 según se indica en el apartado de conexiones que se encuentra más abajo.

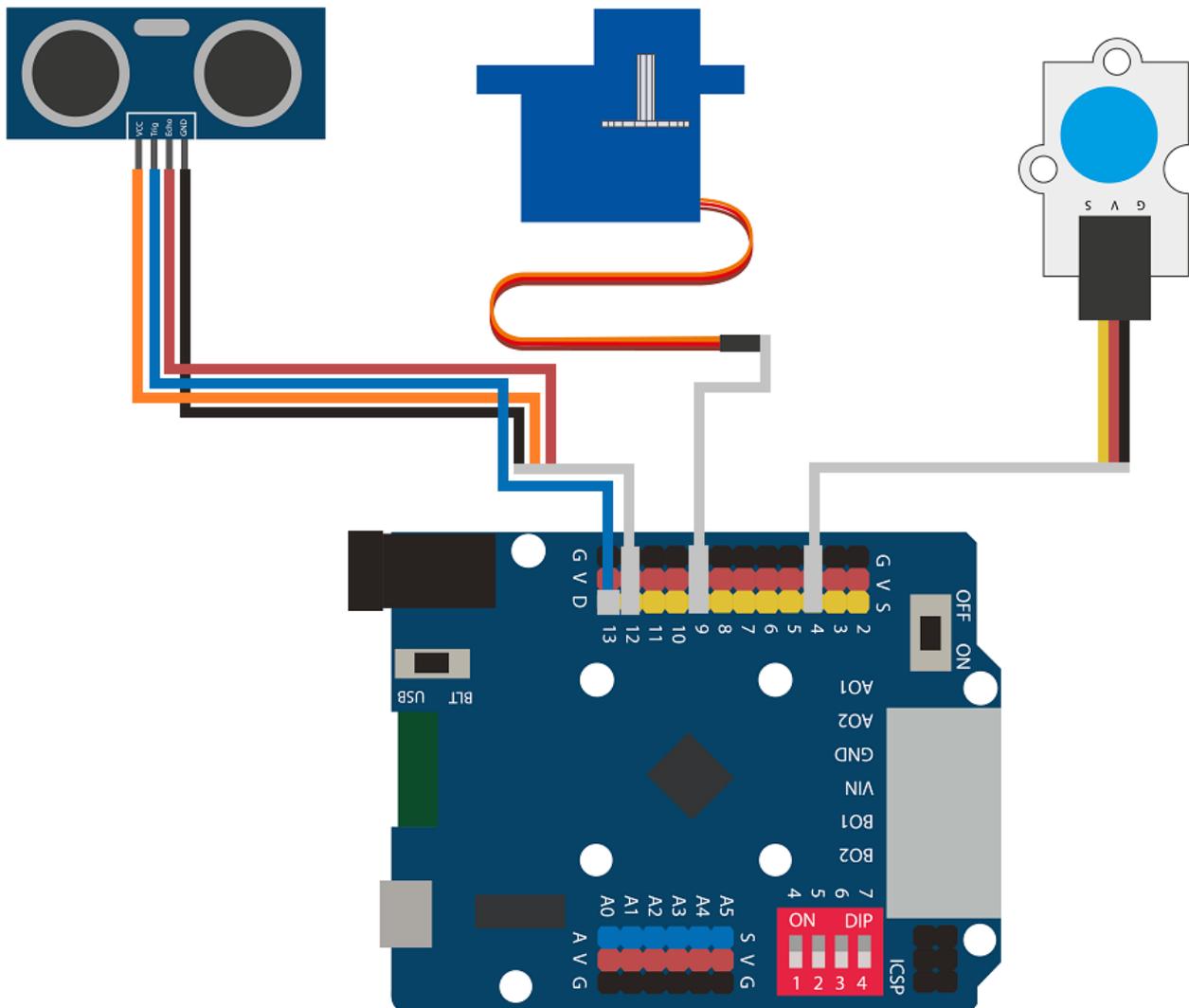
CONEXIONES:

1. Conecta el botón al puerto digital 4 de la placa controladora Build&Code 4in1.
2. Conecta el servomotor al puerto digital 9 de la placa controladora Build&Code 4in1.

Conecta el sensor de distancia por ultrasonidos a los puertos digitales 12 y 13 de la placa controladora Build&Code 4in1.

Puerto digital 12	Sensor de distancia por ultrasonidos
V	VCC
G	GND
S / D	Echo
Puerto digital 13	Sensor de distancia por ultrasonidos
S / D	Trig

Para guiarte, mira los colores de los cables y los colores de los terminales de la placa controladora Build&Code 4in1. Cada cable debe ir conectado a su color.



CÓDIGO DE PROGRAMACIÓN:

El programa consiste en lo siguiente:

1. Leer el estado en que se encuentra el botón. Si el botón es pulsado, la puerta se abrirá, quedará abierta durante 2 segundos y cambiará la variable de estado a 1.
2. Con la puerta abierta, se comprobará la lectura del sensor de distancia por ultrasonidos. Si la lectura es mayor de 20 cm, significa que no hay ningún objeto entrando al parking. Entonces, se esperará 2 segundos, se cambiará la variable de estado a 0 y se cerrará la puerta.
3. Si la lectura del sensor de ultrasonidos es menor a 20 cm, significa que hay un vehículo entrando al parking y se mantendrá la puerta abierta. Cuando el vehículo haya entrado y la lectura del sensor de distancia por ultrasonidos sea mayor a 20 cm, esperará 2 segundos,

cambiará la variable de estado a 0 y la puerta se cerrará.

Esta secuencia se irá repitiendo siempre que todo el sistema tenga alimentación eléctrica.

Puedes realizar esta actividad utilizando los *software* Arduino y Bitbloq, además de otros *software* de programación por bloques compatibles. A continuación encontrarás el código de programación necesario para cada *software*.

Código Arduino

1. [Descarga el software Arduino](#) y realiza el proceso de instalación.
2. Abre el programa Arduino y, una vez en él, copia el siguiente programa:

```
#include <Servo.h>
Servo motor1; // motor1 = Nombre del servomotor
int TrigPin = 13; // Puertos de conexión del sensor de
ultrasonidos
int EchoPin = 12;
float SSound = 0.0343; //VELOCIDAD DEL SONIDO EN cm/μs
long Lengh, Distance ; // VARIABLES PARA CALCULAR LA DISTANCIA EN
CM
int VButton = 0, PButton = 4; // VARIABLE DE LECTURA DEL BOTÓN;
Puerto digital 4 conectado botón
int State = 0; // Variable estado
void setup() {
  // put your setup code here, to run once:
  motor1.attach (9); // Servomotor conectado al puerto digital 9
  pinMode(TrigPin, OUTPUT); // Configuración puertos del sensor de
ultrasonidos
  pinMode(EchoPin, INPUT);
  pinMode (PButton, INPUT); // Configuración puerto botón como
entrada
}

void loop() {
  // put your main code here, to run repeatedly:
  VButton = digitalRead (PButton); // Leer el estado del botón
  if ((VButton == 1) && (State == 0)) // si botón = pulsado y State
= 0
  {
    motor1.write(15); // Servomotor = Abrir puerta
    delay (2000); // Espera 2 segundos
    State = 1; // State = 1;
  }
  DistanceCM(); // Activar función para medir la distancia del
```

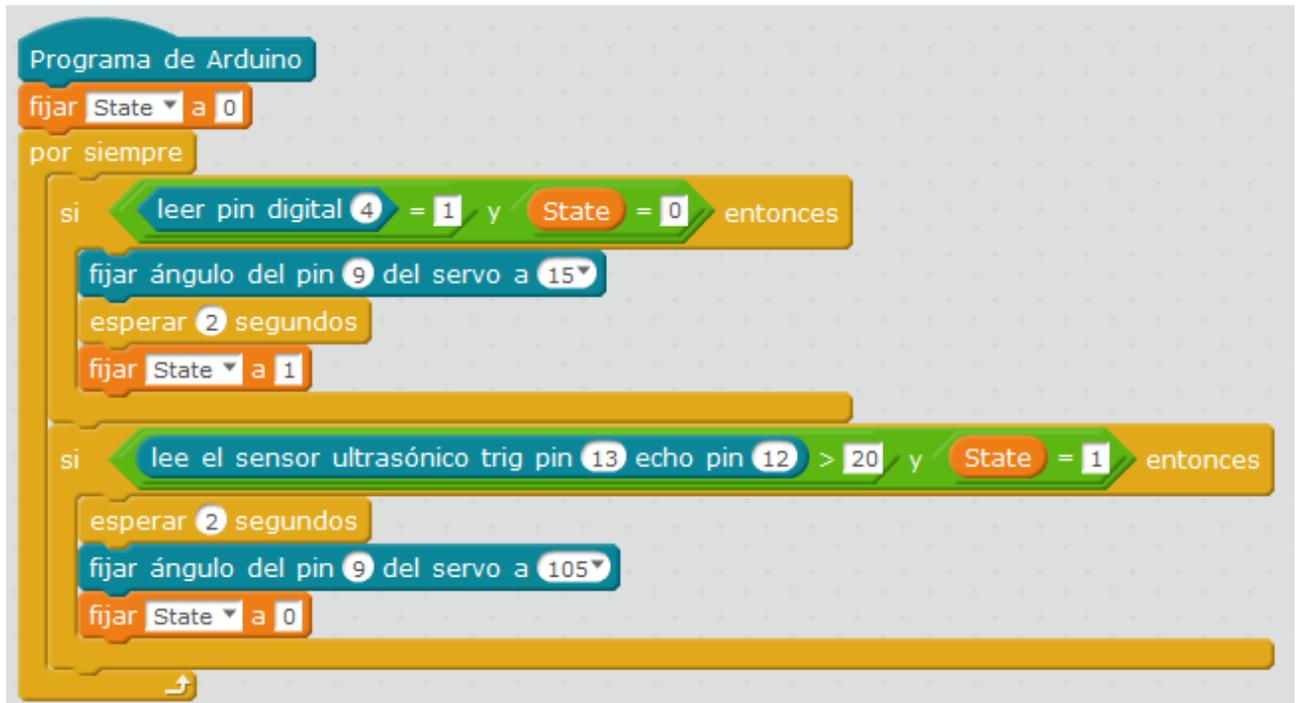
```
sensor de distancia por ultrasonidos
  if ((Distance > 20) && (State == 1)) // Si la distancia es mayor
a 20 cm y State = 1
  {
    delay (2000); // Esperar 2 segundos
    motor1.write(105); // Servomotor = Cerrar puerta
    State = 0; // State = 0
  }
}

void DistanceCM()// FUNCIÓN DE CÁLCULO DE DISTANCIA
{
  // CALCUL DE LA DISTANCIA EN CM
  digitalWrite(TrigPin, LOW);          // Nos aseguramos de que el
trigger está desactivado
  delayMicroseconds(4);                // Para asegurarnos de que el
trigger está LOW
  digitalWrite(TrigPin, HIGH);        // Activamos el pulso de
salida
  delayMicroseconds(14);               // Esperamos 10 µs. El pulso
sigue activo este tiempo
  digitalWrite(TrigPin, LOW);         // Cortamos el pulso y a
esperar el ECHO
  Lengh = pulseIn(EchoPin, HIGH) ; //pulseIn mide el tiempo que
pasa entre que el pin declarado (echoPin) cambia de estado bajo a
alto (de 0 a 1)
  Distance = SSound* Lengh / 2; // CÁLCULO DE LA DISTANCIA
}
```

3. Configura y carga el código, siguiendo las instrucciones indicadas en el [documento de Primeros Pasos de la placa Build&Code 4in1](#).

Código para el *software* de programación por bloques compatible

1. [Descarga el software](#) y realiza el proceso de instalación.
2. Abre el programa y, una vez en él copia el siguiente código:



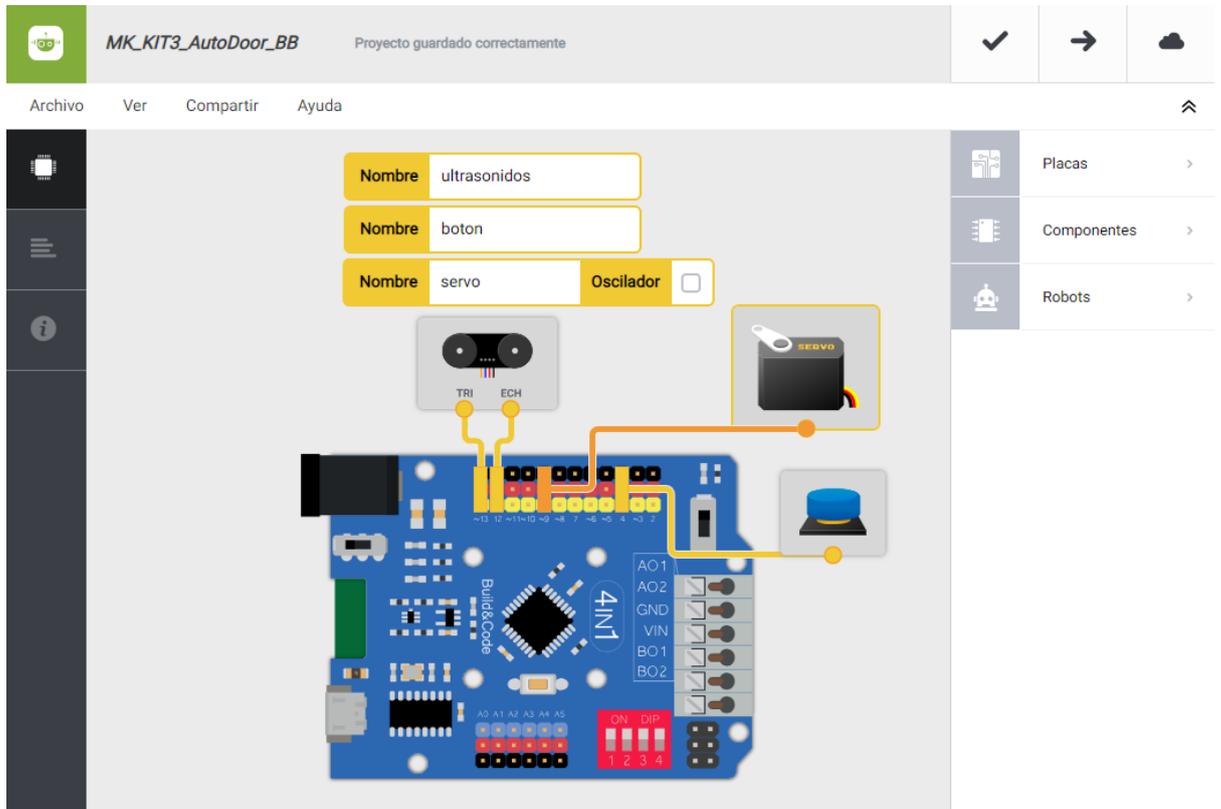
```

Programa de Arduino
fijar State a 0
por siempre
si leer pin digital 4 = 1 y State = 0 entonces
    fijar ángulo del pin 9 del servo a 15
    esperar 2 segundos
    fijar State a 1
si lee el sensor ultrasónico trig pin 13 echo pin 12 > 20 y State = 1 entonces
    esperar 2 segundos
    fijar ángulo del pin 9 del servo a 105
    fijar State a 0
  
```

3. Configura y carga el código, siguiendo las instrucciones indicadas en el [documento de Primeros Pasos de la placa Build&Code 4in1](#).

Código BitBloq

1. Accede al software [Bitbloq](#).
2. Abre el programa Bitbloq y, una vez en él, copia el siguiente código:
 - **Hardware**



o **Software**

Archivo Editar Ver Compartir Ayuda

Bloques Código

— Variables globales y funciones

Declarar variable State = 0

— Instrucciones iniciales (Setup)

Arrastra un bloque aquí para empezar tu programa

— Bucle principal (Loop)

Si Leer boton = 1 y Variable State = 0 ejecutar:

Mover servo a 15 grados

Esperar 2000 ms

Variable State = 1

Si Leer ultrasonidos > 20 y Variable State = 1 ejecutar:

Esperar 2000 ms

Mover servo a 105 grados

Variable State = 0

Componentes >

Fun Funciones >

Var Variables >

Cód Código >

Mat Matemáticas >

Tex Texto >

Con Control >

Lóg Lógica >

3. Configura y carga el código, siguiendo las instrucciones indicadas en el [documento de Primeros Pasos de la placa Build&Code 4in1](#).

RESULTADO DEL EJERCICIO:

Al pulsar el botón, la puerta de parking se abrirá. Luego de dos segundos, el sensor de ultrasonidos comprobará que el auto ya no se encuentre frente a la puerta. Si la lectura es mayor a 20 cm la puerta se cerrará y si es menor que 20 cm la puerta se mantendrá abierta hasta que haya pasado el vehículo. ¡Has creado tu puerta de parking automática!