



PROYECTO N° 6

Vehículo de emergencia

PROYECTO N° 6: Vehículo de emergencia

Aprende a crear un programa para que el Code&Drive simule un vehículo de emergencia. Por ejemplo: un coche de policía, una ambulancia o un coche de bomberos.

NIVEL DE DIFICULTAD: Intermedio.

DURACIÓN DEL EJERCICIO: 60 min.

MATERIALES:

- 1 Cable USB - micro USB

- Ordenador

CONEXIONES:

El kit Code&Drive deberá estar montado y conectado de acuerdo a las instrucciones indicadas en el manual.

CÓDIGO DE PROGRAMACIÓN

Puedes realizar esta actividad utilizando el *software* Arduino, así como un *software* de programación por bloques compatible. A continuación encontrarás el código de programación necesario.

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:

```
int MdirectionA = 4, velocityA = 5, MdirectionB = 7, velocityB = 6;
//VELOCIDAD Y DIRECCIÓN DE LOS MOTORES A Y B

int trigPin = 13; // PINS DEL SENSOR ULTRASONIDOS
int echoPin = 12;

float SSound = 0.0343; //VELOCIDAD DEL SONIDO EN cm/us

long length, distance ; // VARIABLES PARA CALCULAR LA DISTANCIA EN
cm
int LEDG = 9; // PIN LED VERDE
int LEDB = 10; // PIN LED AZUL

void setup() {

// CONFIGURACIÓN DE LOS PUERTOS DE LA BUILD&CODE 4IN1

pinMode ( MdirectionA, OUTPUT);
pinMode ( velocityA, OUTPUT);
pinMode ( MdirectionB, OUTPUT);
pinMode ( velocityB, OUTPUT);
pinMode(trigPin, OUTPUT);
pinMode(echoPin, INPUT);
pinMode(LEDG , OUTPUT);
pinMode(LEDB , OUTPUT);

}
```

```
void loop() {

// CALCULO DE LA DISTANCIA

digitalWrite(trigPin, LOW);           // ASEGURAR DE QUE EL TRIGGER
ESTÉ DESACTIVADO

delayMicroseconds(4);                 // PARA ASEGURARTE QUE EL
TRIGGER ESTÁ EN LOW

digitalWrite(trigPin, HIGH);          //ACTIVA EL PULSO

delayMicroseconds(14);                // ESPERA 10µs. EL PULSO SIGUE
ACTIVO ESTE TIEMPO

digitalWrite(trigPin, LOW);           // CORTA EL PULSO Y ESPERA AL
ECHO

length = 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* length / 2; // CÁLCULO DE LA DISTANCIA

if (( distance < 25) && ( distance > 1)) // SI LA DISTANCIA MEDIDA
ESTÁ ENTRE 25 Y 1 cm
{
// ATRAS; BACK
//MOTOR B
analogWrite (velocityB, 150); // VELOCIDAD EL MOTOR-B, ES DE 0
(STOP) A 255 (MAXIMA VELOCIDAD)
digitalWrite ( MdirectionB,HIGH); // SENTIDO DE GIRO MOTOR-B, LOW
(DELANTE) , HIGH (ATRÁS)
//MOTOR A
analogWrite (velocityA, 150); // VELOCIDAD EL MOTOR-A, ES DE 0
(STOP) A 255 (MAXIMA VELOCIDAD)
digitalWrite ( MdirectionA,LOW); // SENTIDO DE GIRO MOTOR-A, HIGH
(DELANTE) , LOW (ATRÁS)
delay (1000);

// GIRO A LA DERECHA; TURN RIGHT
//MOTOR B
analogWrite (velocityB, 150); // VELOCIDAD EL MOTOR-B, ES DE 0
```

```

(STOP) A 255 (MAXIMA VELOCIDAD)
digitalWrite ( MdirectionB,LOW); // SENTIDO DE GIRO MOTOR-B, LOW
(DELANTE) , HIGH (ATRÁS)
//MOTOR A
analogWrite (velocityA, 0); // VELOCIDAD EL MOTOR-A, ES DE 0
(STOP) A 255 (MAXIMA VELOCIDAD)
digitalWrite ( MdirectionA,HIGH); // SENTIDO DE GIRO MOTOR-A, HIGH
(DELANTE) , LOW (ATRÁS)
delay (1000);
}
else if ( distance > 25) // SI LA DISTANCIA ES MAYOR A 25 cm
{
//DELANTE; FORWARD
//MOTOR B
analogWrite (velocityB, 150); // VELOCIDAD EL MOTOR-B, ES DE 0
(STOP) A 255 (MAXIMA VELOCIDAD)
digitalWrite ( MdirectionB,LOW); // SENTIDO DE GIRO MOTOR-B, LOW
(DELANTE) , HIGH (ATRÁS)
//MOTOR A
analogWrite (velocityA, 150); // VELOCIDAD EL MOTOR-A, ES DE 0
(STOP) A 255 (MAXIMA VELOCIDAD)
digitalWrite ( MdirectionA,HIGH); // SENTIDO DE GIRO MOTOR-A, HIGH
(DELANTE) , LOW (ATRÁS)
}
// ACTIVAMOS LAS LUCES Y LA SIRENA DE EMERGENCIA
Emergency();

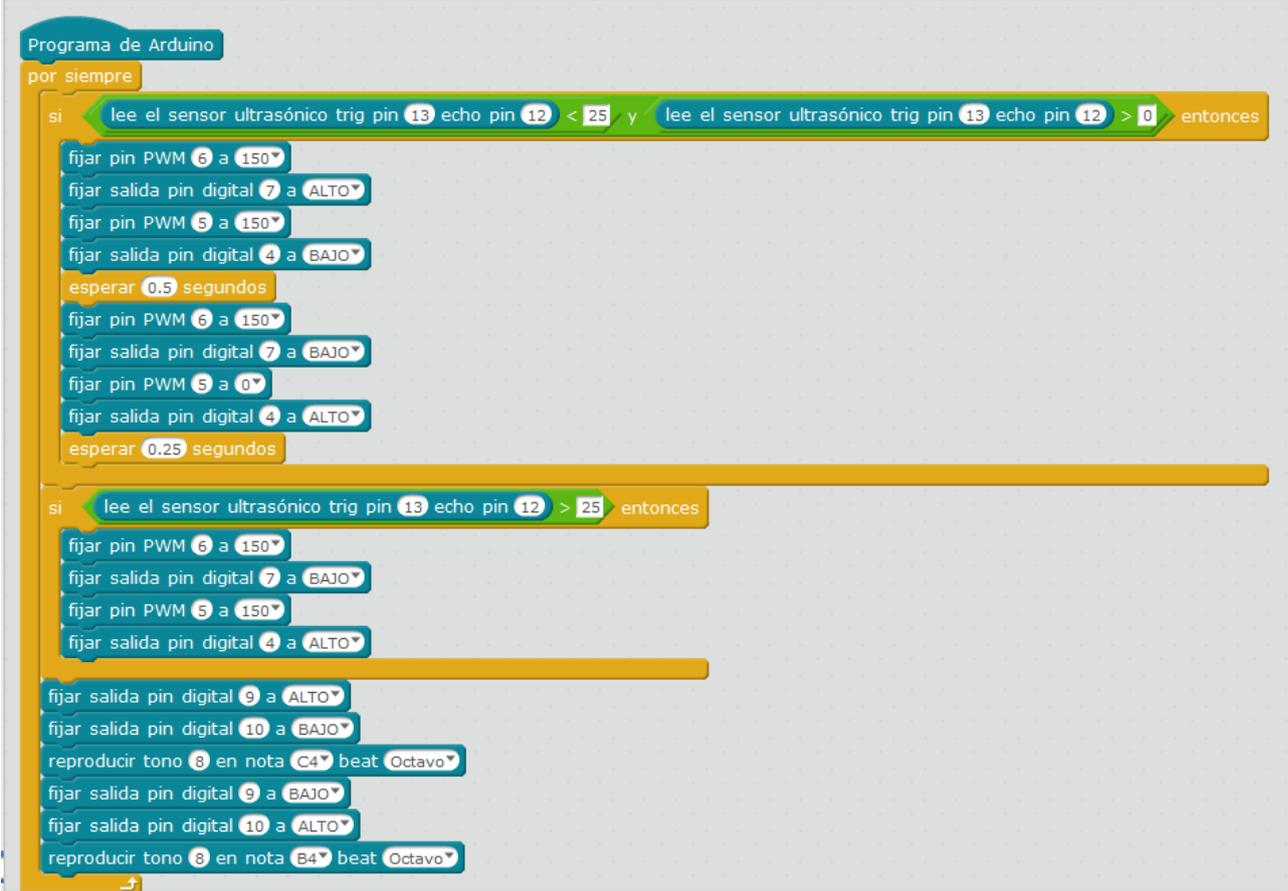
}
void Emergency ()
{
digitalWrite(LEDG, HIGH);// LUZ VERDE ON
digitalWrite(LEDB, LOW);// LUZ AZUL OFF
tone(8, 494, 500);// SONIDO BUZZER
delay(400);// PAUSA DE 400ms
digitalWrite(LEDG, LOW);// LUZ VERDE OFF
digitalWrite(LEDB, HIGH);// LUZ AZUL ON
tone(8, 440, 500);// SONIDO BUZZER
delay(400);// PAUSA DE 400ms
}

```

3. Configura y carga el código, siguiendo las instrucciones indicadas en la guía de Primeros Pasos del Code&Drive.
4. Revisa que el interruptor BLT/USB de la placa controladora Build&Code 4in1 esté en posición USB, para una correcta carga del código.

Código para 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:



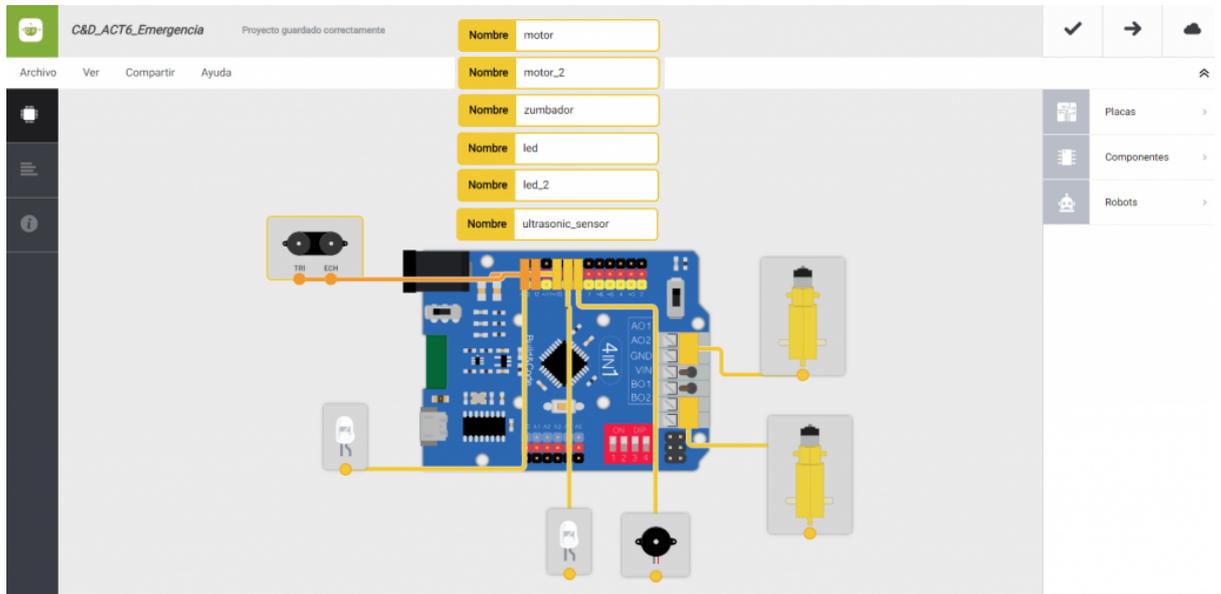
```

Programa de Arduino
por siempre
si lee el sensor ultrasónico trig pin 13 echo pin 12 < 25 y lee el sensor ultrasónico trig pin 13 echo pin 12 > 0 entonces
  fijar pin PWM 6 a 150
  fijar salida pin digital 7 a ALTO
  fijar pin PWM 5 a 150
  fijar salida pin digital 4 a BAJO
  esperar 0.5 segundos
  fijar pin PWM 6 a 150
  fijar salida pin digital 7 a BAJO
  fijar pin PWM 5 a 0
  fijar salida pin digital 4 a ALTO
  esperar 0.25 segundos
si lee el sensor ultrasónico trig pin 13 echo pin 12 > 25 entonces
  fijar pin PWM 6 a 150
  fijar salida pin digital 7 a BAJO
  fijar pin PWM 5 a 150
  fijar salida pin digital 4 a ALTO
  fijar salida pin digital 9 a ALTO
  fijar salida pin digital 10 a BAJO
  reproducir tono 8 en nota C4 beat Octavo
  fijar salida pin digital 9 a BAJO
  fijar salida pin digital 10 a ALTO
  reproducir tono 8 en nota B4 beat Octavo
  
```

3. Configura y carga el código, siguiendo las instrucciones indicadas en la guía de Primeros Pasos del Code&Drive.
4. Revisa que el interruptor BLT/USB de la placa controladora Build&Code 4in1 esté en posición USB, para una correcta carga del código.

Código Bitbloq

1. Accede [al software Bitbloq](#) y realiza el proceso de instalación de la aplicación Web2board.
2. Abre el programa Bitbloq y, una vez en él, copia el siguiente código:
 - o **Hardware**



○ **Software**

C&D_ACT6_Emergencia

Archivo Editar Ver Compartir Ayuda

Bloques Código

— Instrucciones iniciales (Setup)

- Escribir en el pin digital 6 el valor analógico 175
- Escribir en el pin digital 5 el valor analógico 175

— Bucle principal (Loop)

Si Leer ultrasonico_sensor < 25 ejecutar:

- Escribir en el pin digital 7 el valor digital Bajo
- Escribir en el pin digital 4 el valor digital Alto
- Esperar 1000 ms
- Escribir en el pin digital 7 el valor digital Alto
- Escribir en el pin digital 4 el valor digital Alto
- Esperar 1000 ms

de lo contrario, ejecutar:

- Apagar el LED led
- Apagar el LED led_2
- Escribir en el pin digital 7 el valor digital Alto
- Escribir en el pin digital 4 el valor digital Bajo
- Encender el LED led
- Apagar el LED led_2
- Sonar el zumbador zumbador con la nota Re durante 500 ms
- Apagar el LED led
- Encender el LED led_2
- Sonar el zumbador zumbador con la nota Sol# durante 500 ms

3. Configura y carga el código, siguiendo las instrucciones indicadas en la guía de Primeros Pasos del Code&Drive.
4. Revisa que el interruptor BLT/USB de la placa controladora Build&Code 4in1 está en posición USB, para una correcta carga del código.

RESULTADO DEL EJERCICIO

El Code&Drive avanzará con los LEDs y el zumbador encendidos, simulando una sirena de vehículo de emergencia. Cuando el sensor de ultrasonidos detecte un objeto a menos de 25 cm, se moverá hacia atrás, girará y seguirá por otro camino.

Puedes combinar el código de este proyecto con los proyectos 4 (Esquiva obstáculos) y 5 (Seguidor de línea).

También puedes personalizar el Code&Drive con la plantilla de furgoneta. Utiliza esta plantilla como base para tus propios diseños y conviértelo en una ambulancia, un coche de policía o un coche de bomberos.

