

A line follower robot is shown from a top-down perspective. The robot has an orange plastic chassis with two black wheels. A teal semi-transparent rectangle is overlaid on the top half of the robot. Inside this rectangle, the text 'Code&Drive' is on the left, 'Project 5' is on the right, and 'LINE FOLLOWER' is centered in large white letters. A bundle of colorful wires (red, yellow, blue, black) is visible at the top of the robot.

Code&Drive

Project 5

LINE FOLLOWER

PROJECT N° 5: Line follower

Learn how to create a program so that the Code&Drive follows a line drawn on a flat surface. If the surface is white or light-colored, then the line must be black. If the surface is dark-colored or black, then the line must be white, so that the line follower can detect it.

DIFFICULTY LEVEL: Intermediate.

DURATION OF THE EXERCISE: 45 min.

MATERIALS:

- 6 AA batteries
- 1 USB - Micro USB cable

- Computer

The Code&Drive will have to be built according to the instructions manual.

CONNECTIONS:

For this project you will use two line-followers sensors. The line followers are calibrated by default, but the natural light can affect the sensors. Follow these steps to calibrate them again:

1. The line followers have a potentiometer situated on the top. It's a little blue box with a white circle.
2. Use a cross-head screwdriver to regulate the potentiometer, so that it recognizes the black line.
3. Place the Code&Drive on the line and put the line follower sensors over it.
4. Check if the red LED on the sensor turns on. If it doesn't, regulate the potentiometer until the red LED turns on.
5. Once the potentiometer has been regulated, move the Code&Drive to the white or light-colored surface and check if the red LED on the line follower sensor turns on.

The line follower sensors are correctly calibrated if when placing the Code&Drive on the black line the red LEDs on the sensors turn on, and when placing the Code&Drive on the white surface the red LEDs turn off.

The line follower sensors transmit a digital signal that we can see with the red LED integrated. If the sensors detect the black line, the red LEDs will be on. If the sensors detect the white or light-colored surface, the red LEDs will be off.

This way, when the red LED is on, it will send the value 1 to the control board. And when the LED red is off, it will send the value 0 to the control board.

Program the Code&Drive according to the table below, so that its movements correspond to the reading of the line follower sensors.

Left sensor	Right sensor	Code&Drive Direction
0	0	Forward
0	1	Left
1	0	Right
1	1	Back

PROGRAMMING CODE

You can do this project using the Arduino, Bitbloq and other visual programming software by blocks compatible. Below you will find the necessary code.

Arduino Code

1. Download and install the Arduino IDE program. It is available for Windows, Mac OS and Linux.
2. Open the Arduino program and copy the following program in it:

```
int MdirectionA = 4, velocityA = 5, MdirectionB = 7,
velocityB = 6; // SPEED AND DIRECTION OF MOTOR A AND
MOTOR B

int pinR = 2, pinL = 3, followerR, followerL; // LINE
FOLLOWER SENSOR PINS

void setup() {

    //BUILD&CODE 4IN1 PORTS CONFIGURATION

    pinMode ( MdirectionA, OUTPUT);
    pinMode ( velocityA, OUTPUT);
    pinMode ( MdirectionB, OUTPUT);
    pinMode ( velocityB, OUTPUT);
    pinMode (pinR, INPUT);
    pinMode (pinL, INPUT);

}

void loop() {

    // LINE FOLLOWER SENSOR READINGS
```

```

followerR = digitalRead (pinR);
followerL = digitalRead (pinL);

if ((followerR == 0)& (followerL == 0)) // FORWARD
{
    //MOTOR B
    analogWrite (velocityB, 150); // MOTOR B SPEED,
    FROM 0 (STOP) TO 255 (MAXIMUM SPEED)
    digitalWrite ( MdirectionB,LOW); // MOTOR B ROTATION
    DIRECTION, LOW (FORWARD), HIGH (BACKWARD)

    //MOTOR A

    analogWrite (velocityA, 150); // MOTOR A SPEED,
    FROM 0 (STOP) TO 255 (MAXIMUM SPEED)
    digitalWrite ( MdirectionA,HIGH); // MOTOR A ROTATION
    DIRECTION, HIGH (FORWARD), LOW (BACKWARD)

}

if ((followerR == 0)& (followerL == 1)) //RIGHT
{
    //MOTOR B
    analogWrite (velocityB, 0); // MOTOR B SPEED,
    FROM 0 (STOP) TO 255 (MAXIMUM SPEED)
    digitalWrite ( MdirectionB,LOW); // MOTOR B ROTATION
    DIRECTION, LOW (FORWARD), HIGH (BACKWARD)

    //MOTOR A

    analogWrite (velocityA, 150); // MOTOR A SPEED,
    FROM 0 (STOP) TO 255 (MAXIMUM SPEED)
    digitalWrite ( MdirectionA,HIGH); //MOTOR A
    ROTATION DIRECTION, LOW (FORWARD), HIGH (BACKWARD)

}

if ((followerR == 1)& (followerL == 0)) //LEFT
{
    //MOTOR B
    analogWrite (velocityB, 150); // MOTOR B SPEED,
    FROM 0 (STOP) TO 255 (MAXIMUM SPEED)
    digitalWrite ( MdirectionB,LOW); // MOTOR B

```

```

ROTATION DIRECTION, LOW (FORWARD), HIGH (BACKWARD)

    //MOTOR A
    analogWrite (velocityA, 0);    // MOTOR A SPEED,
    FROM 0 (STOP) TO 255 (MAXIMUM SPEED)
    digitalWrite ( MdirectionA,HIGH); //MOTOR A
    ROTATION DIRECTION, LOW (FORWARD), HIGH (BACKWARD)
}

if ((followerR == 1)& (followerL == 1)) //BACK
{
    //MOTOR B

    analogWrite (velocityB, 150); // MOTOR B SPEED,
    FROM 0 (STOP) TO 255 (MAXIMUM SPEED)
    digitalWrite ( MdirectionB,HIGH); // MOTOR B
    ROTATION DIRECTION, LOW (FORWARD), HIGH (BACKWARD)

    //MOTOR A

    analogWrite (velocityA, 150); // MOTOR A SPEED,
    FROM 0 (STOP) TO 255 (MAXIMUM SPEED)
    digitalWrite ( MdirectionA,LOW); //MOTOR A
    ROTATION DIRECTION, LOW (FORWARD), HIGH (BACKWARD)
}
}

```

Code for the visual programming software by blocks compatible

1. Download and install the program.
2. Open the software and copy the following code. Use the following image as a guide:

```

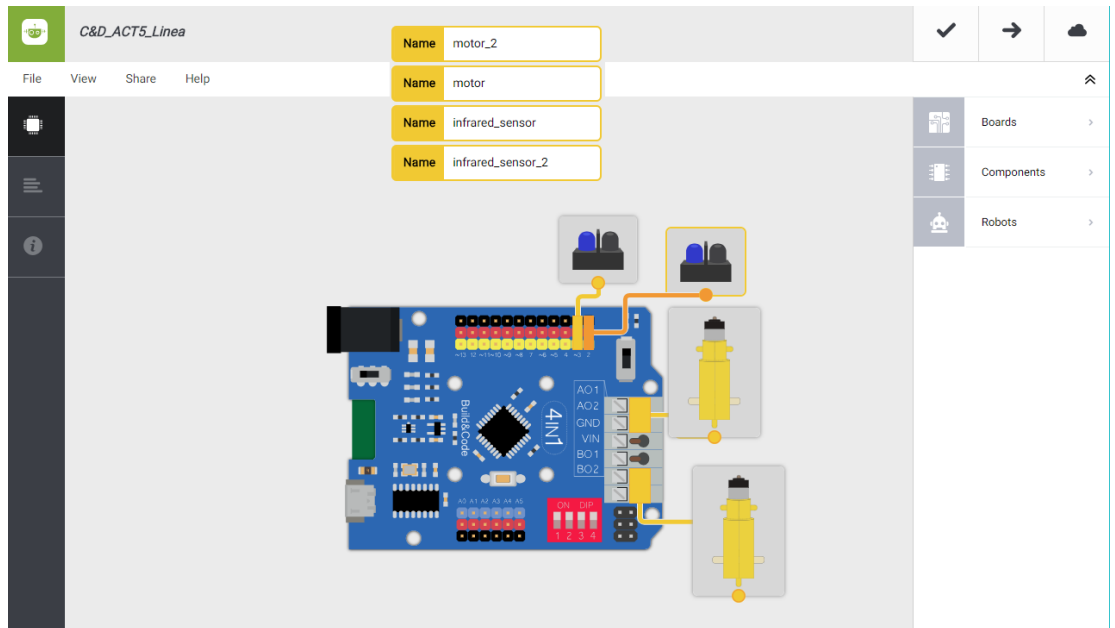
Arduino Program
forever
  if (read digital pin 2 = 0 and read digital pin 3 = 0) then FORWARD
    set pwm pin 6 output as 150 MOTOR-B
    set digital pin 7 output as LOW
    set pwm pin 5 output as 150 MOTOR-A
    set digital pin 4 output as HIGH
  if (read digital pin 2 = 1 and read digital pin 3 = 0) then LEFT
    set pwm pin 6 output as 150 MOTOR-B
    set digital pin 7 output as LOW
    set pwm pin 5 output as 0 MOTOR-A
    set digital pin 4 output as HIGH
  if (read digital pin 2 = 0 and read digital pin 3 = 1) then RIGHT
    set pwm pin 6 output as 0 MOTOR-B
    set digital pin 7 output as LOW
    set pwm pin 5 output as 150 MOTOR-A
    set digital pin 4 output as HIGH
  if (read digital pin 2 = 1 and read digital pin 3 = 1) then BACK
    set pwm pin 6 output as 150 MOTOR-B
    set digital pin 7 output as HIGH
    set pwm pin 5 output as 150 MOTOR-A
    set digital pin 4 output as LOW

```

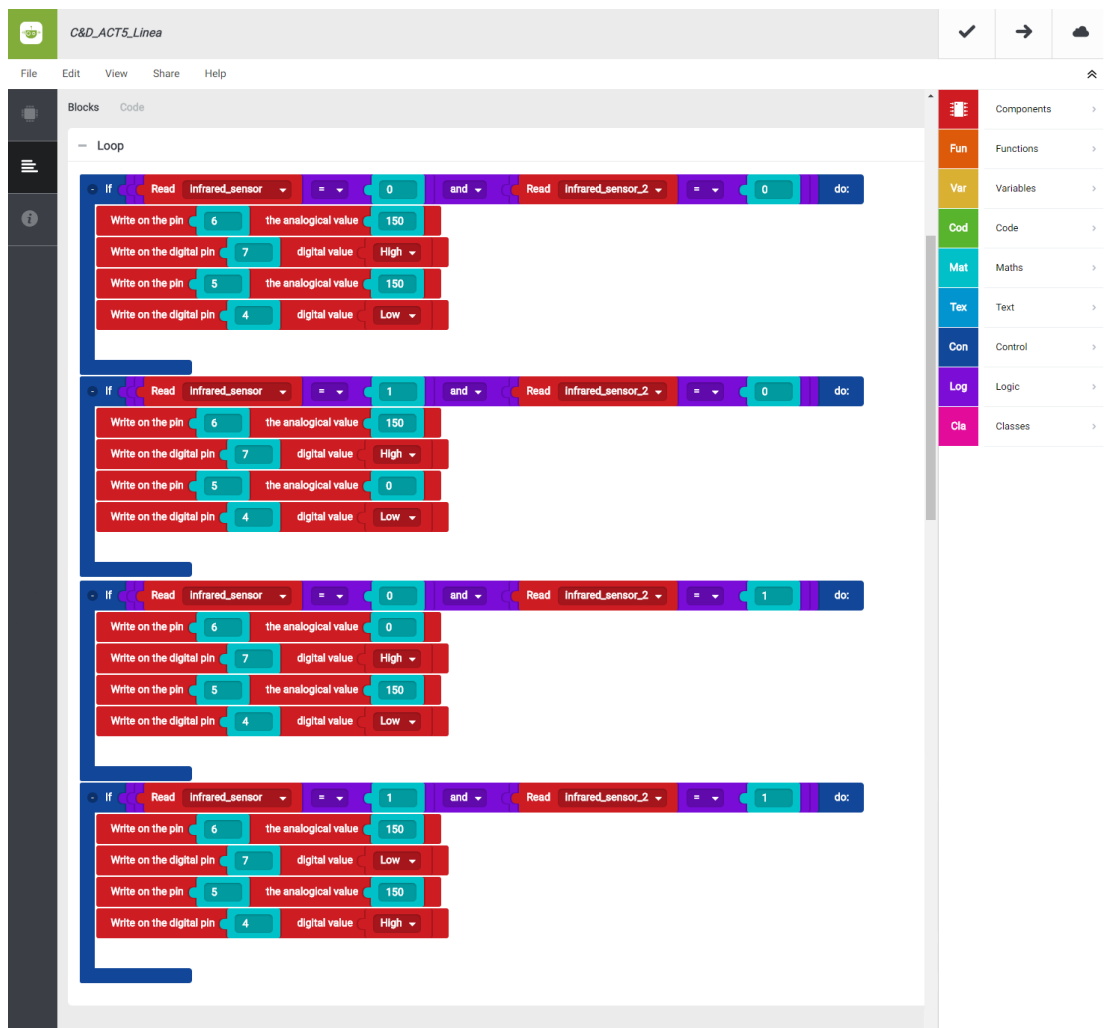
3. Configure and upload the code, following the indications on the Code&Drive First Steps guide.
4. Check that the BTL/USB switch on the Build&Code 4in1 board is set to USB, to upload the code correctly.

Bitblog code

1. Download Bitblog and install the Web2board app.
2. Open the software and copy the following code.
 - **Hardware**



▀ Software



3. Configure and upload the code, following the indications on the Code&Drive First Steps guide.
4. Check that the BTL/USB switch on the Build&Code 4in1 board is set to USB, to upload the code correctly.

RESULT OF THE EXERCISE

When putting the Code&Drive over a black line and white surface (or the other way around) the line follower will detect it and the car robot will move forward, following the line.