

Add Crash Sensors to your Robot

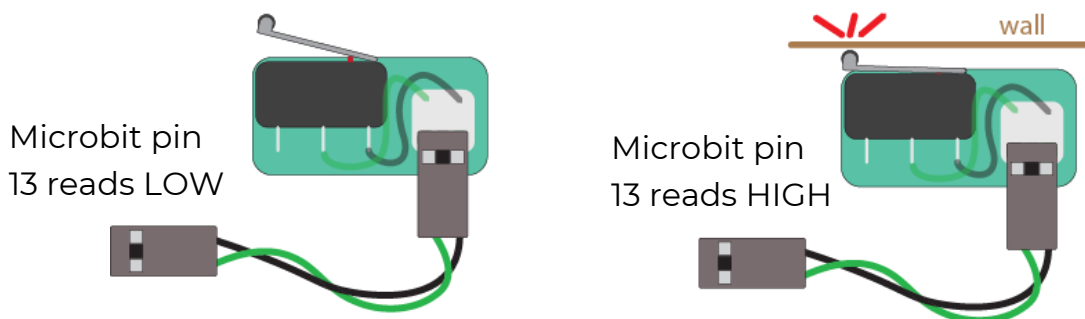
In this workshop you will add some crash sensors to your robot. These will detect when the robot has crashed into a wall or another robot. You can use these to take evasive action, such as turning around or making a noise.

How it Works

The crash sensors are a type of switch called a microswitch. We will use a digital input to detect when the switch is hit, which closes the switch.

Each switch will be connected to a pin on the Microbit. When the switch is open, the pin will read HIGH, corresponding to a value of 1. When the switch is closed, the pin will be set LOW, corresponding to a value of 0. We can use code to read to 0 and 1 values and respond accordingly.

The crash sensors need to be connected to the Microbit using GS cables, which have 2 wires. G is ground, which is the black wire. S is signal, which is the green wire and connects to the pin on the Microbit.

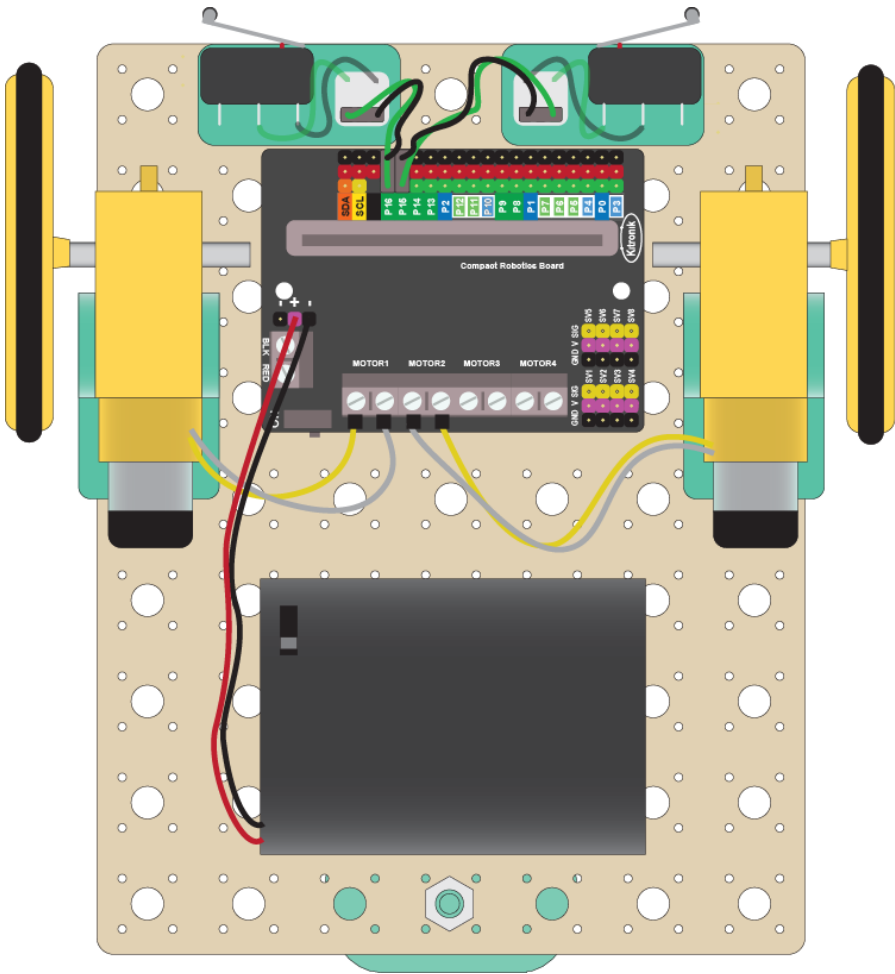


What to do

- If you haven't already done so, build and code the robot by referring to the previous worksheets
- Then follow this worksheet to add two crash sensors and make your robot stop when it crashes
- Finally, attempt the challenges to make your robot respond to a crash in different ways

Add the Crash Sensors

Connect Two Crash Sensors



1 Add two crash sensors to your robot.

These connections on the Microbit are called **pins**

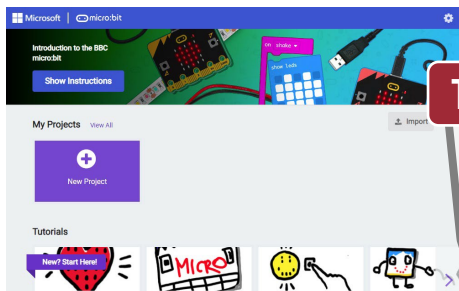
2 Wire up the crash sensors as follows using GS cables

Component	Microbit
Right crash sensor	P15
Left crash sensor	P16

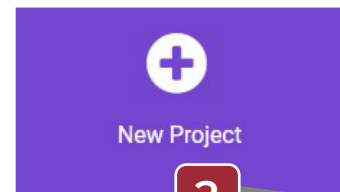
Coding the Robot

Create a Project for the Robot

<https://makecode.microbit.org/>



1 Go to the Makecode website

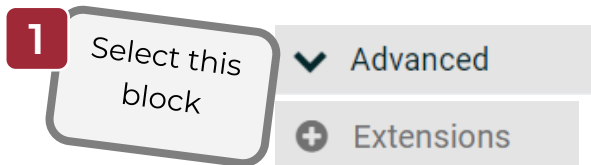


2 Click on New Project

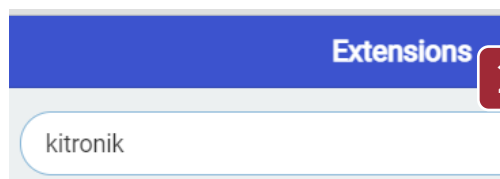
A 'Create a Project' dialog box with a title bar containing three smiley face emojis and a close button. The main text says 'Give your project a name.' Below this is a text input field containing 'bumper robot'. There's a link for 'Code options' and a green 'Create' button with a checkmark.

3 Give the project a name (whatever name you want!)

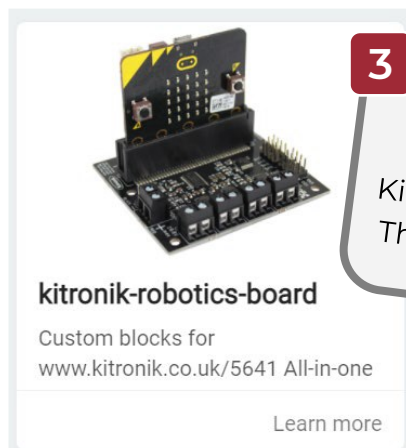
Add the Motor Driver Extension



1 Select this block

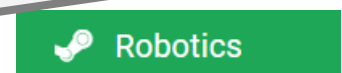


2 Search for Kitronik



3 Make sure you select the right Kitronik extension. There are a few of

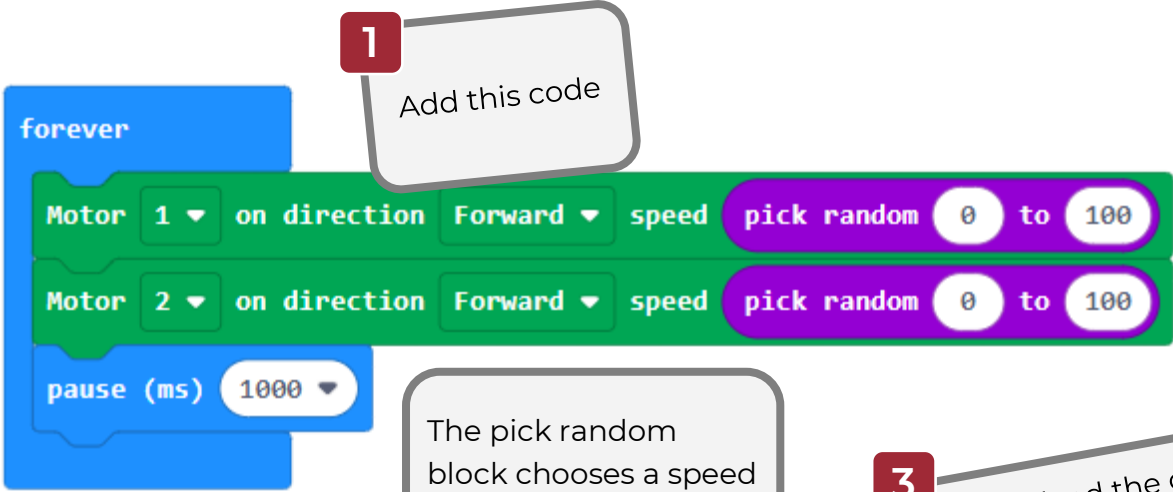
4 If all goes well you will see this new



Coding the Robot

Code a Random Robot

First we will code a robot that moves at random speeds.



1 Add this code

forever

Motor 1 on direction Forward speed pick random 0 to 100

Motor 2 on direction Forward speed pick random 0 to 100

pause (ms) 1000

The pick random block chooses a speed between 0 and 100

3 Download the code to the Microbit, place your robot on the floor and watch it move around

Download

The image shows a Scratch code editor snippet. A blue 'forever' loop block contains two green 'Motor' blocks and a blue 'pause (ms)' block. The first 'Motor' block is set to Motor 1, direction Forward, and speed 'pick random 0 to 100'. The second 'Motor' block is set to Motor 2, direction Forward, and speed 'pick random 0 to 100'. The 'pause' block is set to 1000 ms. A callout box with a red '1' says 'Add this code'. Another callout box with a red '3' says 'Download the code to the Microbit, place your robot on the floor and watch it move around'. A 'Download' button is at the bottom right.

Set up the Crash Sensors

Now we will start to code the crash sensors. First we need to tell the Microbit that the pins should be set to HIGH when the switches are not pressed.



1 Add this code

on start

set pull pin P15 to up

set pull pin P16 to up

The image shows a Scratch code editor snippet. A blue 'on start' block contains two red 'set pull pin' blocks. The first 'set pull pin' block is set to P15 and 'up'. The second 'set pull pin' block is set to P16 and 'up'. A callout box with a red '1' says 'Add this code'.

Coding the Robot

Responding to a Crash

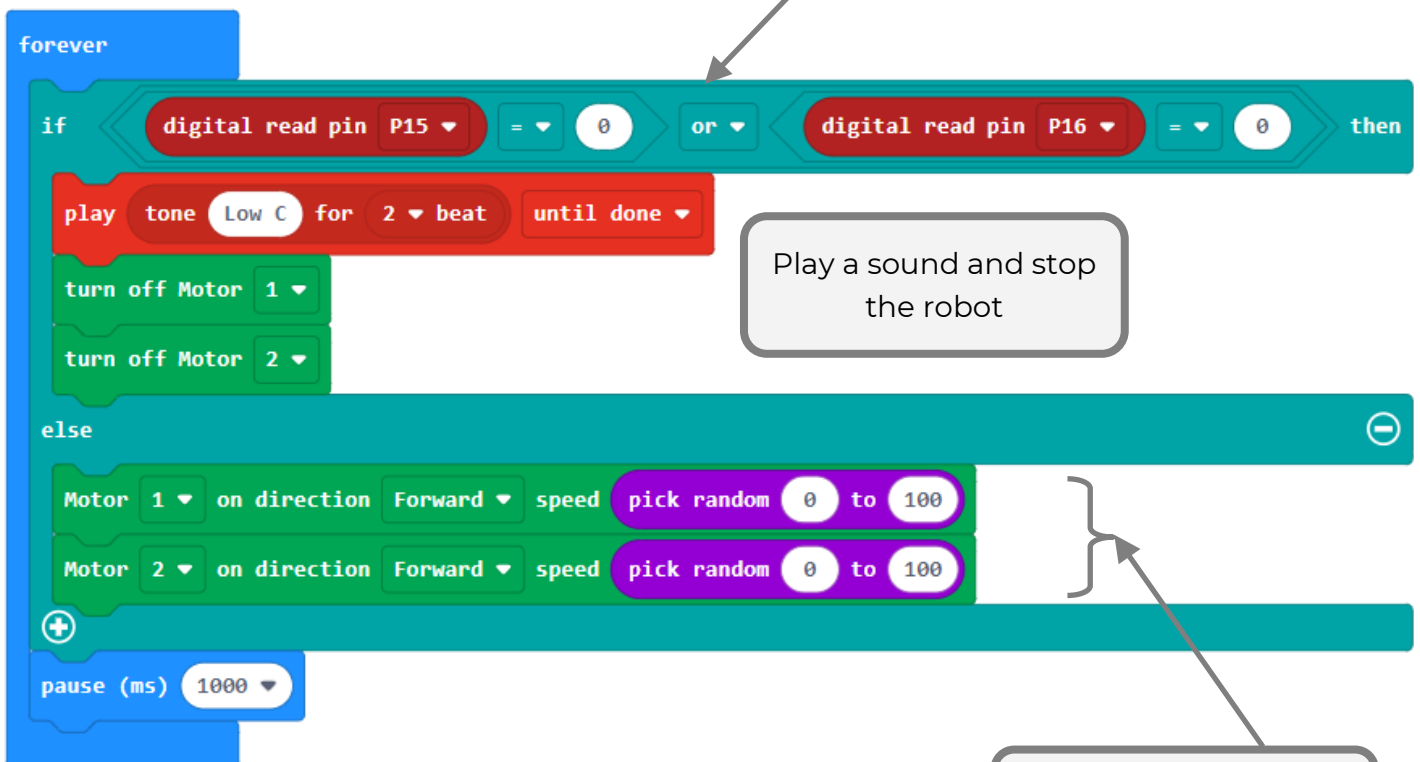
Now we will get the robot to stop and play a sound when the robot hits something.

1

Change the forever block so it looks like this

Detect if either crash sensor is hit. We check if pin 15 is LOW or pin 16 is LOW. So only one sensor needs to be hit for a crash to be detected. This code block is a little tricky! There are several parts:

First add an OR block
then add the = blocks
and then add the pin blocks



Play a sound and stop the robot

Move forwards randomly

3

Download the code to the Microbit, place your robot on the floor and check that it stops when it hits something

Download

Your challenge!

Try these challenges:

- Instead of just stopping, can you get your robot to turn around when it hits a wall?
- Get the robot to flash its lights when it crashes (if you haven't added lights go back to the worksheet **Adding Lights to your Robot**).

Solutions 1

Turning Around

This is one way to get your robot to turn around when it hits something. You may come up with a different way.

The code is a Scratch script for a robot's behavior. It starts with a 'forever' loop. Inside the loop, there is an 'if' condition that checks if digital pin P15 is equal to 0 or if digital pin P16 is equal to 0. If either condition is true, the 'then' block executes. This block contains a 'play tone' block set to 'Low C' for 2 beats until done. This is followed by two 'Motor' blocks: Motor 1 and Motor 2, both set to 'Reverse' direction and '50' speed. A 'pause (ms)' block for 500 ms follows. Then, two more 'Motor' blocks for Motor 1 and Motor 2 are set to 'Reverse' direction and 'pick random 0 to 100' speed. A 'pause (ms)' block for 1000 ms follows, and then a 'stop all sounds' block. If the 'if' condition is false, the 'else' block executes. This block contains two 'Motor' blocks for Motor 1 and Motor 2, both set to 'Forward' direction and 'pick random 0 to 100' speed. After the 'else' block, there is a 'pause (ms)' block for 1000 ms. The 'forever' loop is indicated by a '+' sign at the bottom of the loop block.

```
forever
  if (digital read pin P15 = 0 or digital read pin P16 = 0) then
    play tone Low C for 2 beat until done
    Motor 1 on direction Reverse speed 50
    Motor 2 on direction Reverse speed 50
    pause (ms) 500
    Motor 1 on direction Reverse speed pick random 0 to 100
    Motor 2 on direction Reverse speed pick random 0 to 100
    pause (ms) 1000
    stop all sounds
  else
    Motor 1 on direction Forward speed pick random 0 to 100
    Motor 2 on direction Forward speed pick random 0 to 100
  pause (ms) 1000
```

Reverse straight

Reverse randomly

Forward randomly

Solutions 2

Lights

This code will flash this lights while the robot is crashed.

