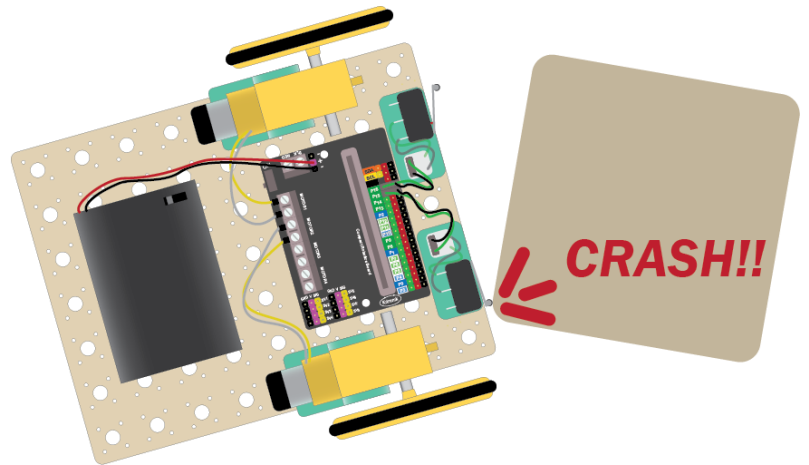


Add Crash Sensors to your Robot

Project 1.03

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.

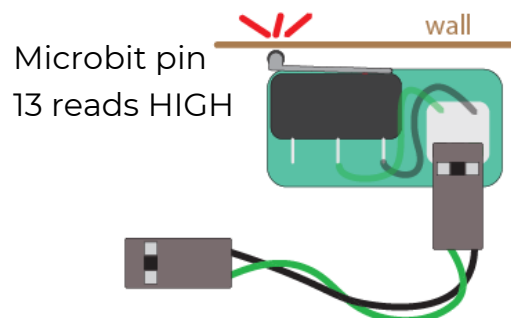
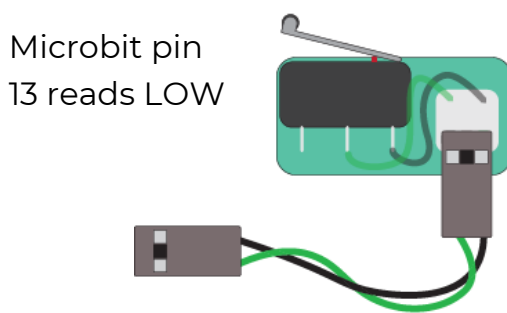


How it Works

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.



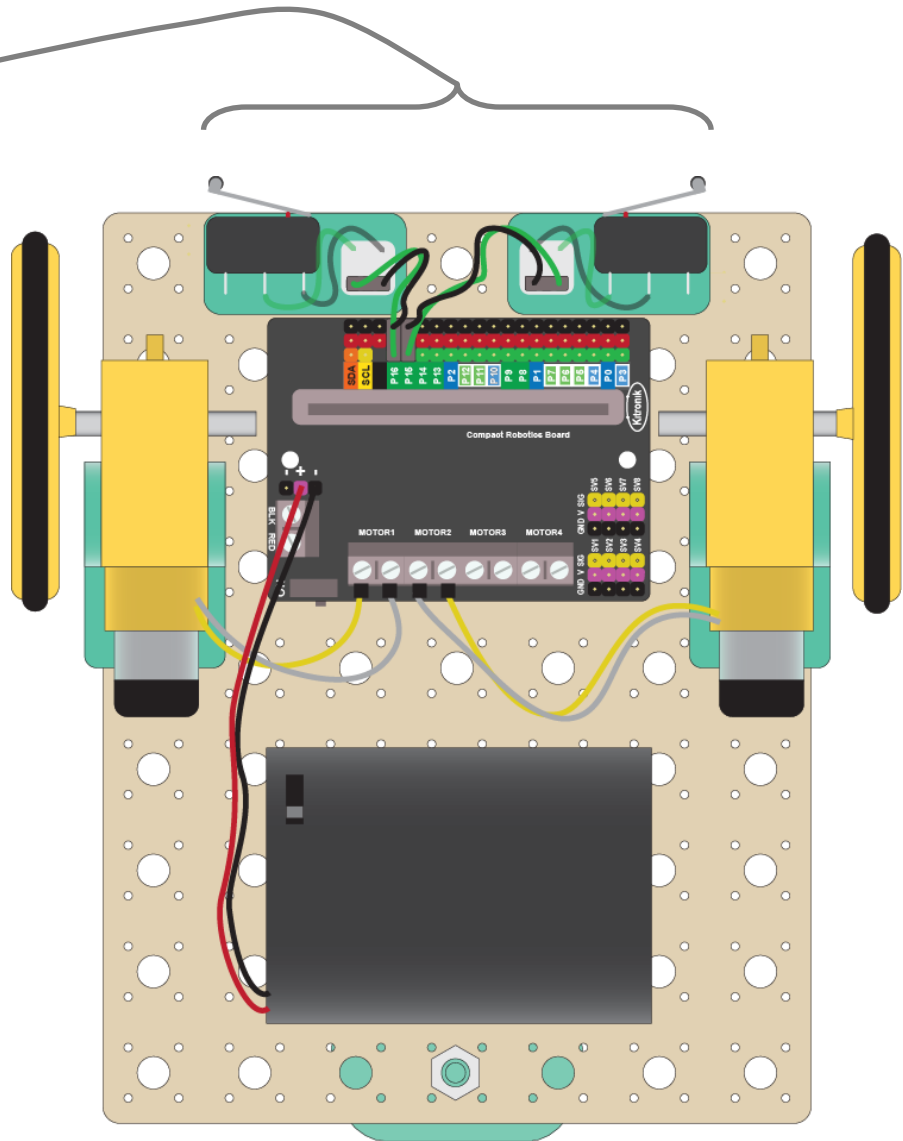
What to do

- 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.



2 Wire up the crash sensors as follows using GS cables

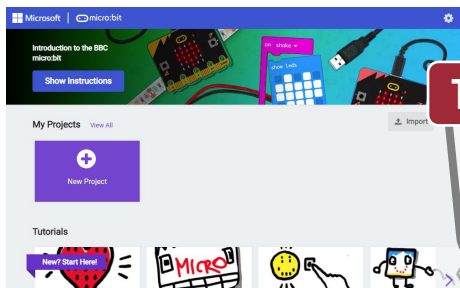
These connections on the Microbit are called **pins**

Component	Microbit Connections
Right crash sensor	P15
Left crash sensor	P16

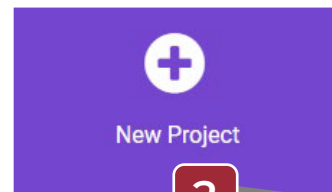
Code the Crash Sensors 1

Create a Project for the Robot

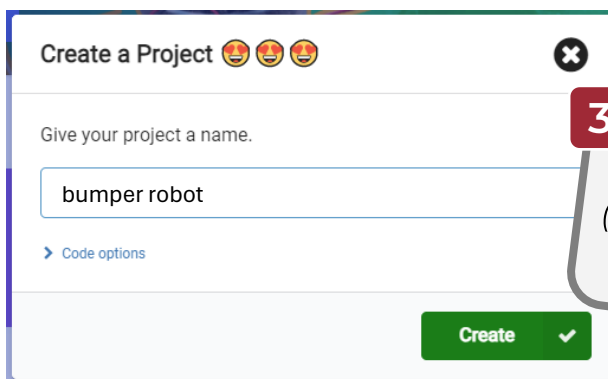
<https://makecode.microbit.org/>



1 Go to the Makecode website



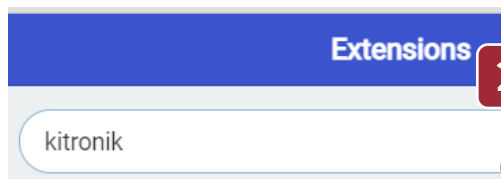
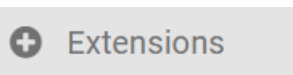
2 Click on New Project



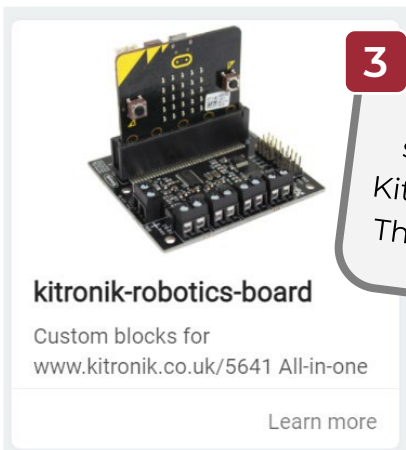
3 Give the project a name (whatever name you want!)

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



Code the Crash Sensors 2

Code a Random Robot

First we will code a robot that moves at random speeds. By turning each wheel at a random speed, the robot will keep changing direction.

1 Add this code

The **pick random** block chooses a speed between 0 and 100

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

Download

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

set pull pin P15 to up

set pull pin P16 to up

Code the Crash Sensors 3

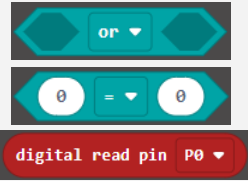
Responding to a Crash

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



```
forever
  if (digital read pin P15 = 0 or digital read pin P16 = 0) then
    play tone Low C for 2 beat until done
    turn off Motor 1
    turn off Motor 2
  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
```

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

Challenges

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 'forever' loop. It starts with an 'if' block that checks if digital pin P15 is equal to 0 OR if digital pin P16 is equal to 0. If true, it executes the following sequence: a 'play tone' block (Low C, 2 beats, until done), two 'Motor' blocks (Motor 1 and Motor 2, both set to 'Reverse' direction and '50' speed), a 'pause (ms)' block (500), another two 'Motor' blocks (Motor 1 and Motor 2, both set to 'Reverse' direction and 'pick random 0 to 100' speed), a 'pause (ms)' block (1000), and a 'stop all sounds' block. To the right of these blocks are three callout boxes: 'Reverse straight', 'Reverse randomly', and 'Forward randomly'. If the 'if' condition is false, it goes to the 'else' block, which sets Motor 1 and Motor 2 to 'Forward' direction and 'pick random 0 to 100' speed. To the right of these blocks is a callout box 'Forward randomly'. After the 'else' block, there is a 'pause (ms)' block (1000).

Solutions 2

Lights

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

```
forever
  if (digital read pin P15 = 0 or digital read pin P16 = 0) then
    play tone Low C for 2 beat until done
    digital write pin P13 to 1
    digital write pin P14 to 1
    pause (ms) 500
    digital write pin P13 to 0
    digital write pin P14 to 0
  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
```