

## B - Rock Field Navigation 2

Using some switch sensors and clever coding, let's make your robot safely navigate the Martian rock fields!



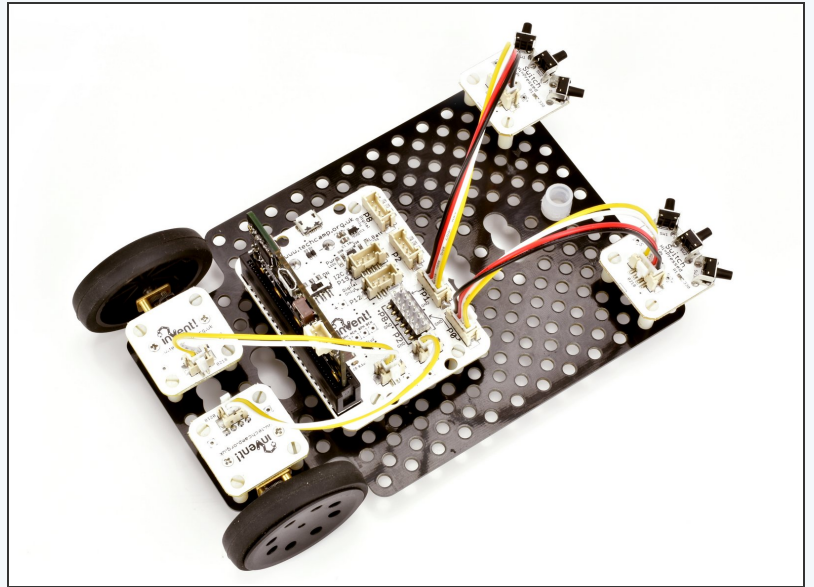
# INTRODUCTION

Using some switch sensors and clever coding, let's make your robot safely navigate the Martian rock fields!

## Step 1

### Two Switches

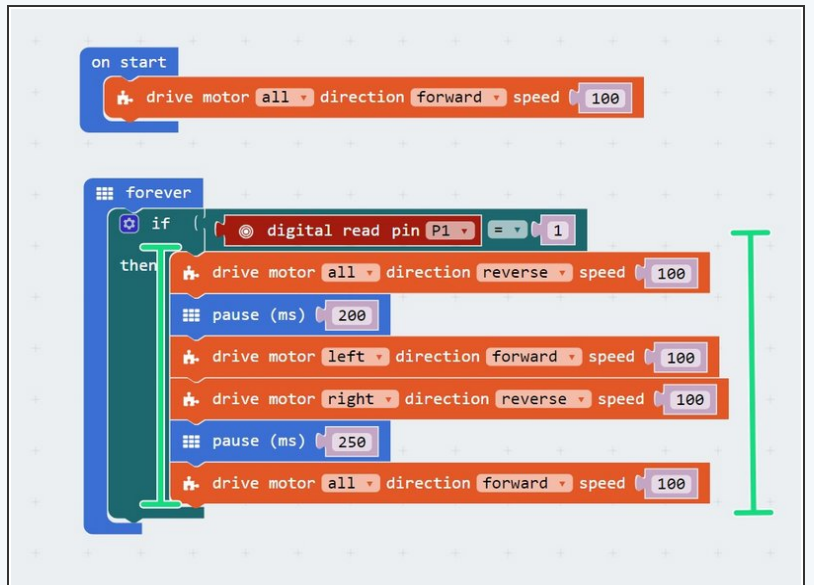
- Hopefully your robot now makes it across the rock field fairly well - but only if the obstacle is **directly in front**.
- You might find it gets stuck with obstacles on the left and right - but we have **two** switch sensors, so we can **upgrade** the robot to fix that!
- Put your **other** switch module onto your robot, and **move the first one** so it looks like the picture.
- They should be plugged into **P0** and **P1**.



## Step 2

### Left Switch

- Let's test the left sensor.
- When the left switch is pressed, the robot should **reverse** and **turn right** to avoid the obstacle.
- Try building the program in the picture and **test your program!**
- Don't forget, you might need to **adjust** how far the robot **turns** and **reverses** for it to work well.



### Step 3

## Right Switch

- Now let's test the **right switch**.
- **Change the program** so it checks the **right sensor**, and then turns **left** to avoid the obstacle!

```
on start
  drive motor all direction forward speed 100

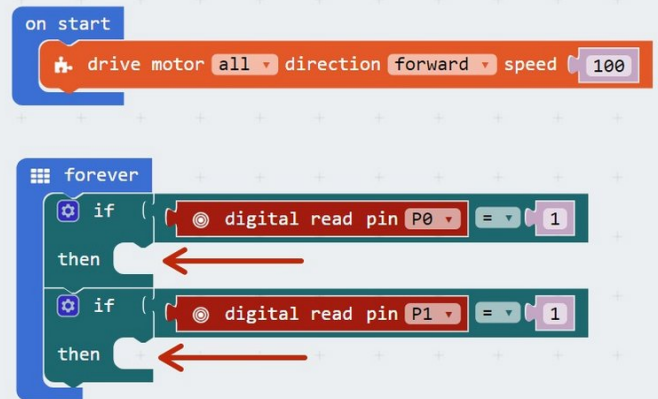
forever
  if (digital read pin P0 = 1)
  then
    drive motor all direction reverse speed 100
    pause (ms) 200
    drive motor left direction reverse speed 100
    drive motor right direction forward speed 100
    pause (ms) 250
    drive motor all direction forward speed 100
```

The code is written in a Scratch-like block-based language. It starts with an 'on start' block containing a 'drive motor all direction forward speed 100' block. Below this is a 'forever' loop. Inside the loop is an 'if' block with the condition 'digital read pin P0 = 1'. If this condition is true, the 'then' block contains a sequence of actions: 'drive motor all direction reverse speed 100', a 'pause (ms) 200' block, 'drive motor left direction reverse speed 100', 'drive motor right direction forward speed 100', a 'pause (ms) 250' block, and finally 'drive motor all direction forward speed 100'.

## Step 4

### Two Sensor Evasion

# Challenge!



- **Change** your one switch code from the last challenge so **both** switches are being checked.
- Make sure that when the **left switch** is hit, the robot reverses and then turns to the **right**, (away from the obstacle) and when the **right switch** is hit it reverses and then turns to the **left**.
- **Test** your program properly in the rock field, and try to adjust the wait times so it works as well as possible!
- There's some example code in the picture if you need a hint - but it is **missing a lot of blocks** you will need to fill in!



## Step 5

### Feedback and Safety

#### Buzzer

- Let's add some more feedback to the robot so we know what its doing.
- **Add the LED and buzzer modules** to your robot.
- Add blocks to your program so that:
  - When driving forwards, the **green LED** is on.
  - When an obstacle is hit and the robot is reversing and turning, the **green LED** is off, and the **red LED** is on.
  - When the robot is reversing, the **buzzer** should **beep** to warn people to get out of the way!
  - If you can, try to make the buzzer **beep several times** as the robot is reversing instead of being on all the time.

**Extension**  
**Challenge!**

