

## Micro:bit Robot Racing

You are going to make a **self-driving robot** and race it against others on a special robot race track.

Your robot will have to follow a black line on white paper. The first robot to complete the race wins.

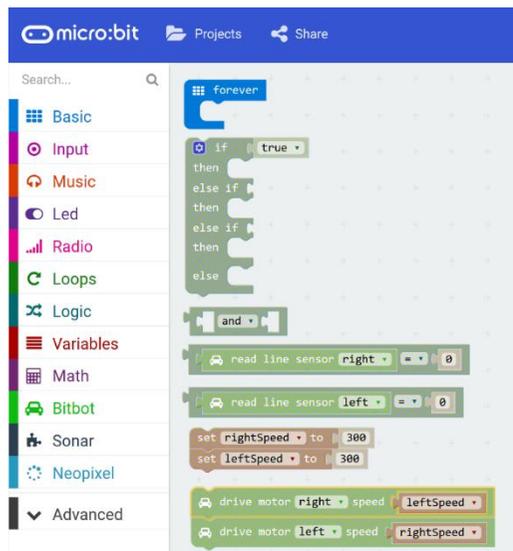
You will use the **Microsoft Block Editor** to create your program using **Blocks**. **Blocks** represent different things the **micro:bit** can do.

The **blocks** are organised into coloured groups. In the instruction, take a look at the colour of the block being selected - that will give you a clue which group on the left to select. Have an explore to find everything the **micro:bit** can do.

This project will explore the following features:

- **Inputs - Line Sensors**
- **Outputs - Motors**
- **Logic/Conditional**
- **Real Time Events**

The micro:bit has 2 line sensors. They can detect black or white. We will use the left and right line sensors to write a programme to steer the robot to follow the line.



## Task #1 – Testing Motors

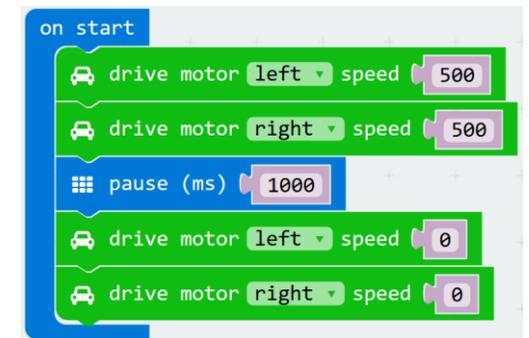
The motor speed is set by the “**drive motor left speed 0**” block. Set the “**speed**” parameter to different values to control the motor:

- **Stop: 0**
- **Forward: 1 to 1023**
- **Reverse: -1 to -1023**

In this task, you will make your **robo:bit** drive forward for 1 second, then stop. You will then experiment with different speeds and how to turn.

Remember – fast is not always best!

1. From “**Bitbot**” group, select “**drive motor left speed 0**” and add to “**on start**”. Change the speed value from “**0**” to “**500**”
2. Add another “**drive motor left speed 0**” – change “**left**” to “**right**”, and set speed to “**500**”
3. From “**Basic**” group, select “**pause (ms) 100**”. Change pause to **1000**.
4. Add more “**drive motor...**” blocks, and set “**left**” and “**right**” motors to stop “**0**”
5. Test your program – download to your **micro:bit**, connect to the **robo:bit**, and see what happens
6. Experiment with different **speed** values and different **pause** values
  1. How fast?
  2. How slow? What is the slowest useful speed?
  3. Backwards
    - *hint* – set “**speed**” to a number between **-1** and **-1024**
  4. Turn Left
    - *hint* – set “**right**” speed bigger than “**left**” speed
    - *hint* – change the “**pause**” time to smaller value like “**100**”
  5. Turn Right
    - *hint* – set “**left**” speed bigger than “**right**” speed



## Task #2 – Line Following Programme

The line sensors on the **robo:bit** can see when a sensor is over black or white.

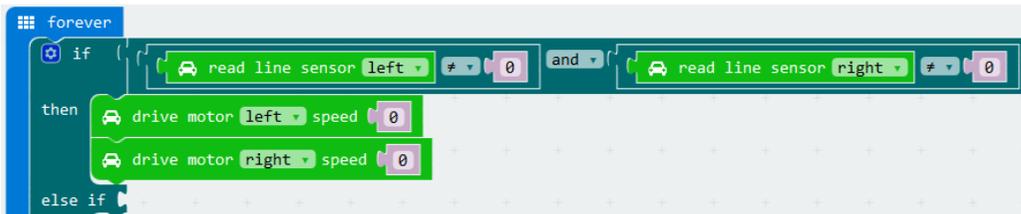
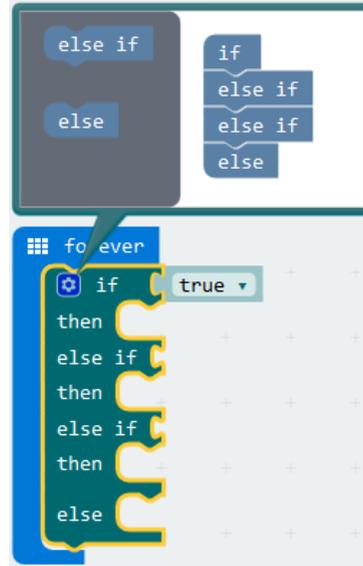
If the “left” line sensor is over black, then we need to turn left. Which wheel should we turn more? Left or right?

In this task, you will get your **robo:bit** to follow a black line – if it wanders too far off the centre of the line, it needs to turn to move back to the centre of it.

There are the 4 following cases:

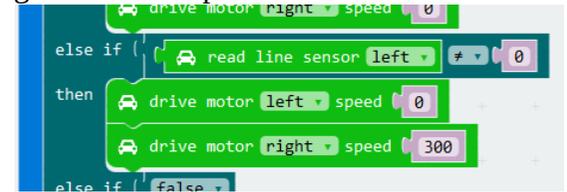
- **Both Sensors over black - stop**
- **Left Sensor over black – turn left**
- **Right Sensor over black – turn right**
- **Both Sensors over white – drive straight**

1. Remove the existing code.
2. Add “Basic” / “forever” block
3. **Conditional Outline**
  1. Add “Logic” / “if true then else” block to “forever”
  2. Press the gear button to the left of “if” and add 2 more “else if” subblocks
4. **Both Sensors over Black – Stop**
  1. Add “Logic” / “and” to first “if”
  2. Add “Logic” / “0 = 0” to both left and right of “and” – change “=” to “≠” (not equals)
  3. Add “Bitbot” / “read line sensor left” to the first “0” of both “0 ≠ 0”. Change one of the “left” to “right”
  4. Add 2 “drive motor left speed 0” to “then” – change one from “left” to “right”.



### 5. **Left Line Sensor over black – turn left:**

1. Add “Logic” / “0 = 0” block to 1<sup>st</sup> “else if”
2. Change “=” to “≠” (not equals)
3. Add “Bitbot” / “read line sensor left” to the first “0” of “0 ≠ 0”.
4. Add 2 “drive motor left speed 0” to “then” – change one from “left” to “right”, and change one motor speed to 300.



### 6. **Right Sensor over black – turn right:**

1. Repeat steps 1 through 5 from above, adding to the 2<sup>nd</sup> “else if”, and replace “left” with “right”. Change the correct motor speed to 300

### 7. **Both Sensors over white – drive straight:**

1. Add 2 “drive motor left speed 0” to “else” – change one from “left” to “right” and change both motor speeds to 300.
2. Test your program – download to your **micro:bit**, connect to the **robo:bit**, and see what happens



8. Now adjust the speed values to try to make your robot go faster
9. Can you enhance the program to make your robot go even faster?