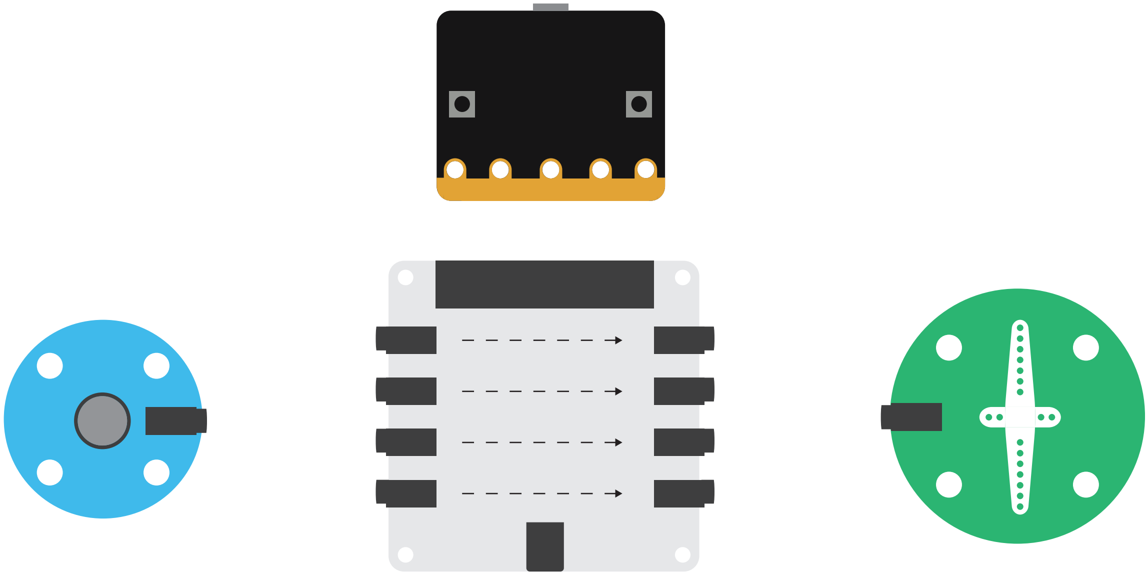
BYOR



Programming manual

2019 micro:bit

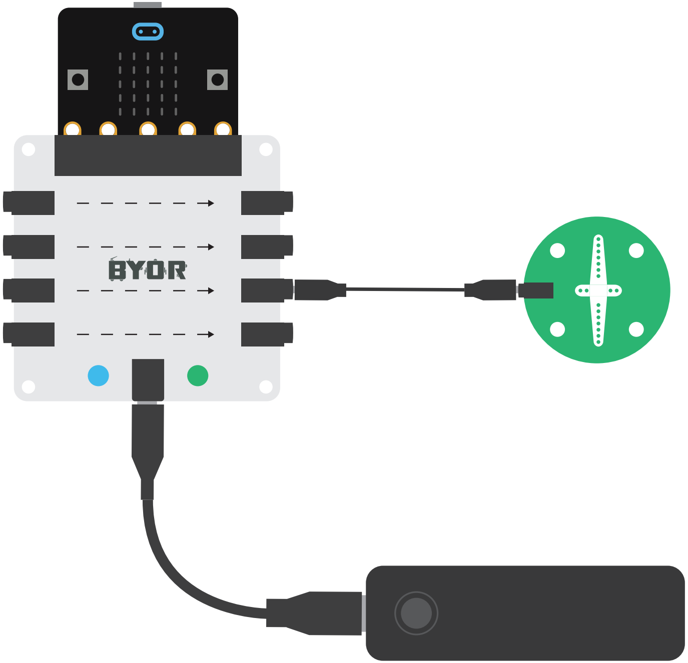
BYOR

Build Your Own Robot

Programming manual

BYOR & micro:bit

This manual describes how to program your micro:bit and make your BYOR parts respond to the programming on the micro:bit. Enabling you to program your BYOR creations in the most simple way.



The steps

1.
Basics

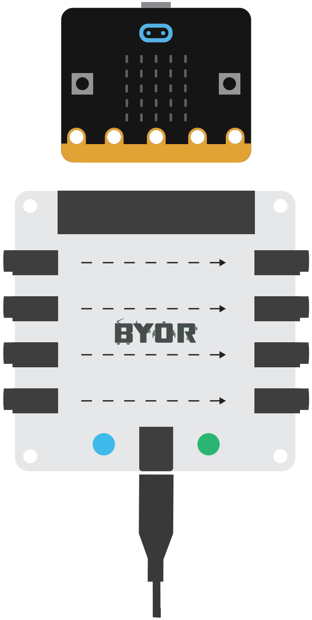
2.
Coding

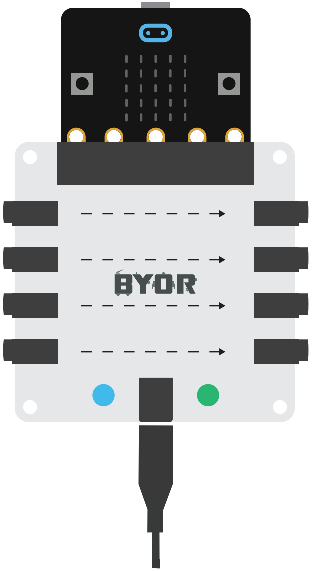
3.
Upload your code

4.
Good to know

Basics

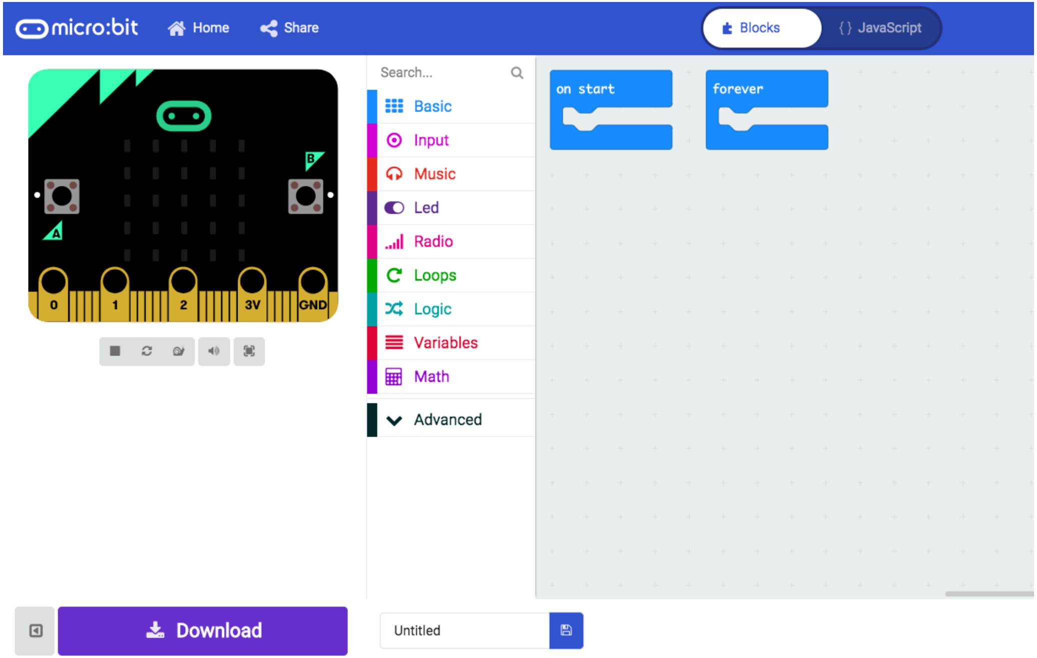
Note that understanding of the basic working of the BYOR-kit is required (see the general manual).
Power your Easyboard and slide the micro:bit into the designated slot. Note the power-LED is now pulsating.
Once the micro:bit is attached, the ports stop responding as described in the general manual.
As long as the micro:bit is connected, the ports will only respond to the code that is on the micro:bit.





Coding

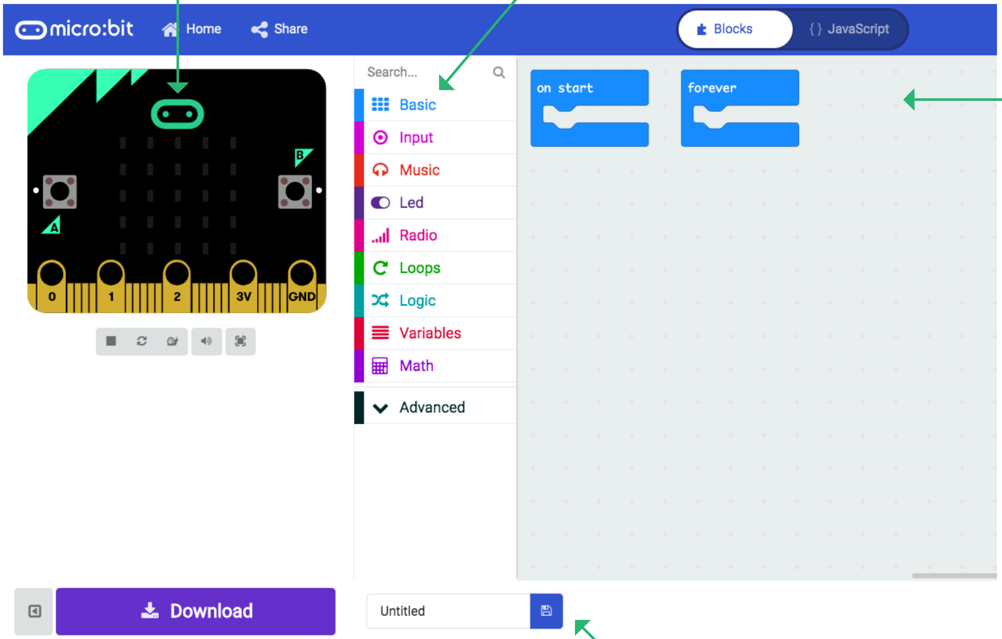
Code for the micro:bit kan be written in a number of different ways with different programs. The most basic way is the micro:bit editor. This is an online editor which is here: https://makecode.microbit.org/#editor



Programming Basics

A digital simulation of a micro:bit that will respond to your code

Different types of code blocks



The Stage
Shows your code in blocks
These blocks will be processed when running the code

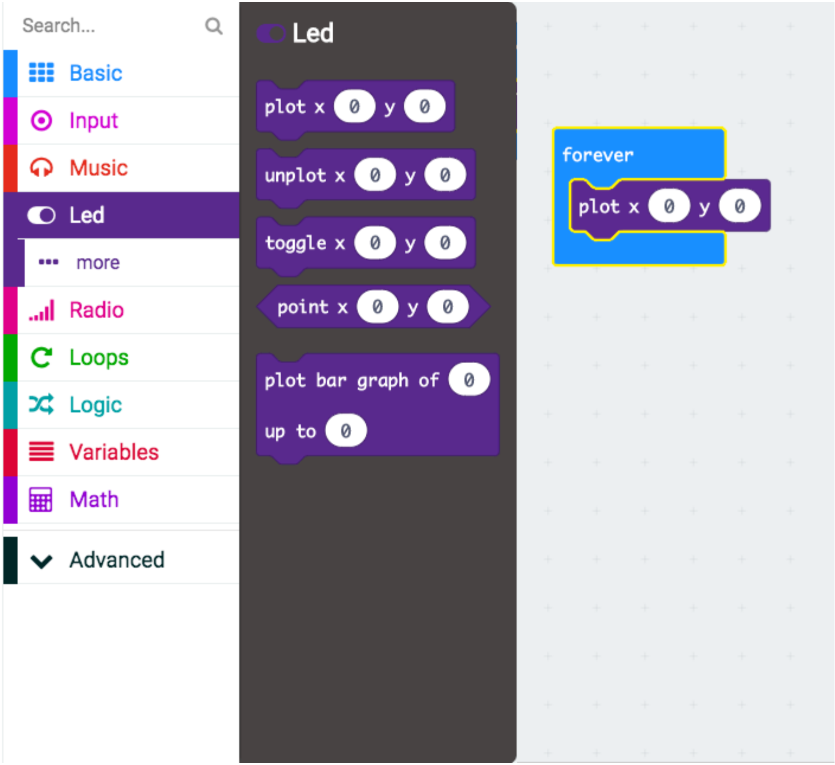
On Start will execute only once when your code starts running.

Forever will execute all the time in a loop when your code is running.

Button to download your code as a file or (when set up) directly to an attached micro:bit

Button to save your code as a file

You can click on one of the block types to open up the menu and drag and drop blocks of code to your 'On start' or 'Forever' function. The easiest function to start with is an LED function.



Click the LED button and drag and drop the Plot-block to your *Forever* function.

Note that the micro:bit on your screen will respond directly.

If you want to get this simple code on your physical micro:bit directly, skip ahead to the **Upload** chapter.

If you want to do more simple coding with the micro:bit without involving the BYOR-hardware yet, take a look at all the examples on the micro:bit website. https://makecode.microbit.org/

Blocks have a certain shape which determines where they can be applied. In some cases, default values are set in places where blocks can also be applied.

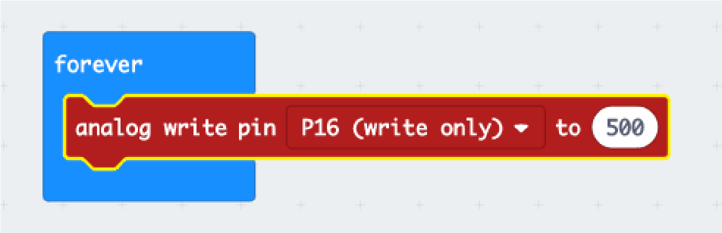
Control outputs

Use micro:bit to make your output-parts do stuff!

There are two ways to make your output parts do things. First is the Digital Write function. This function is digital so it can only turn your output all the way on or all the way off.

The second one is the Analog Write function. This function can turn your output part on in any range.

To use these functions, connect an output part to any of the output ports of the BYOR Easyboard.
Look on the back of the board to see what number the output port has (13, 14, 15 or 16).
Use the Analog or Digital Write functions in the Forever block and fill in the right port number.



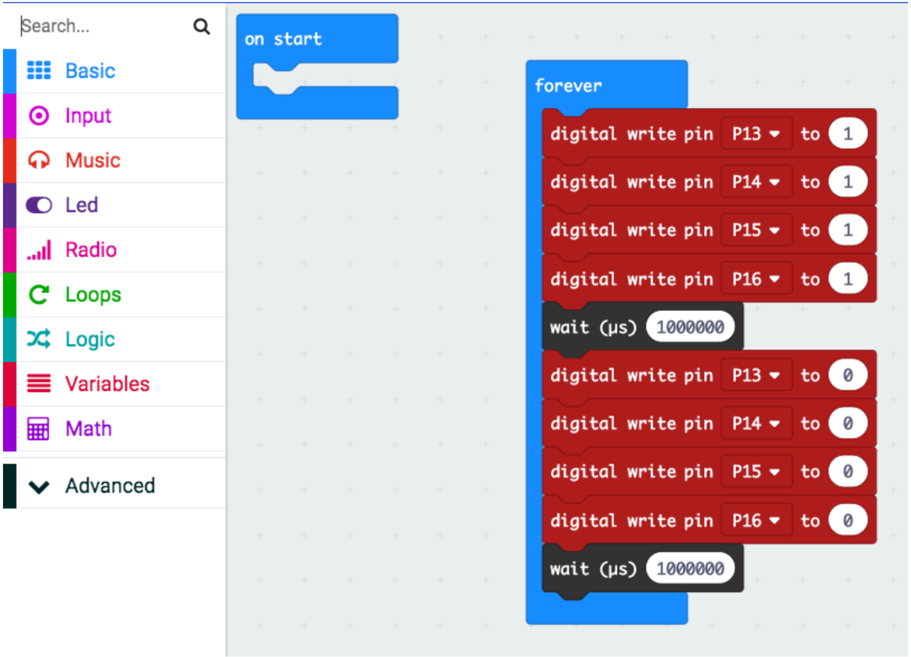
```
forever
    analog write pin P16 (write only) ▾ to 500
```

With an Analog Write function block you can decide the strength of the signal from 0 to 1024
With a Digital Write function block you can only decide on (1) or off (0)

Note: To use the analog write perfectly the pulse width must be set (see chapter: Good to Know)

Blink Code

Let's make a simple code that will make the BYOR servo motor go back and forth automatically! Don't worry, if you don't have a servo motor you can use any of the other BYOR outputs as well. Drag and drop the blocks so that they form the code which is shown in the image below. **Note** you can search for blocks using the search function.



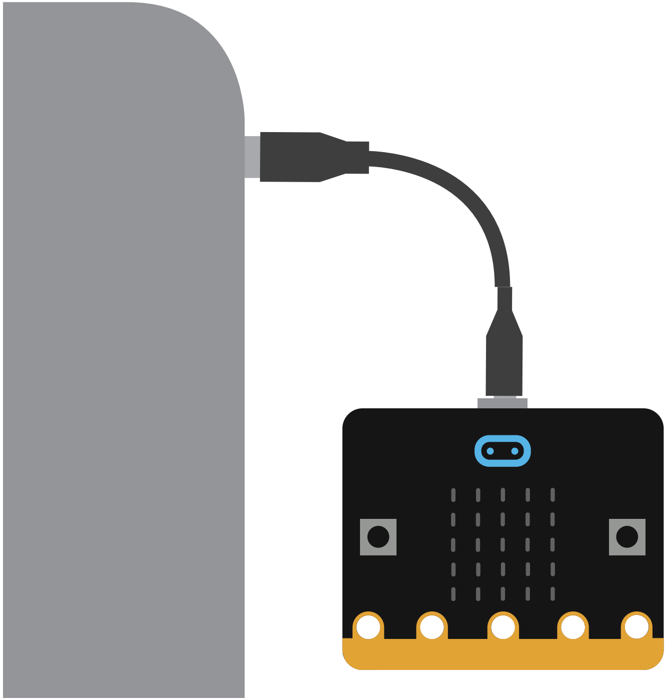
```
on start

forever
    digital write pin P13 to 1
    digital write pin P14 to 1
    digital write pin P15 to 1
    digital write pin P16 to 1
    wait (µs) 1000000
    digital write pin P13 to 0
    digital write pin P14 to 0
    digital write pin P15 to 0
    digital write pin P16 to 0
    wait (µs) 1000000
```

Parts of the blocks are adjustable

The numbers correspond to certain ports on the BYOR easyboard. **On the back of the easyboard you can see which port is which.** In this code, all the ports are actived and deactivated at the same time. So it makes no matter which output port you use to attach your output part.

Upload

Now to get your home made code on your micro:bit!
First: attach your micro:bit to the computer using the USB cable.



The micro:bit will now appear on your computer as a storage device (USB stick)

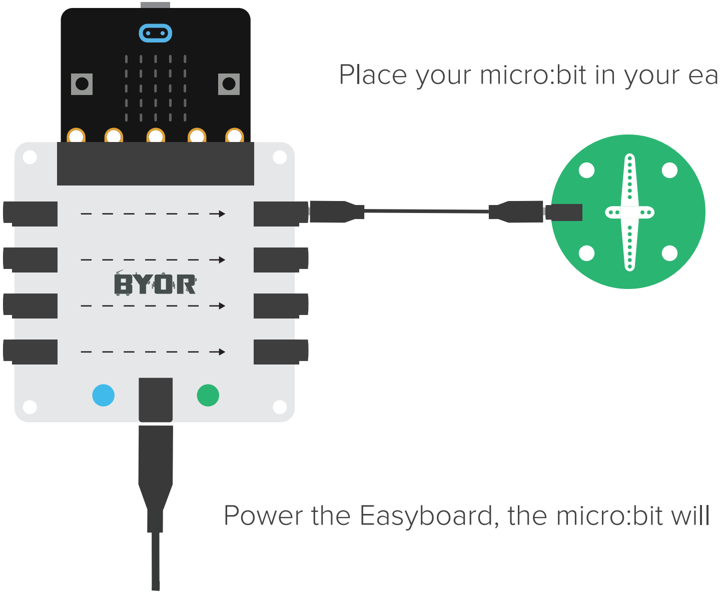
MICROBIT

See the quickstart on the micro:bit web page for a demo and tips.

https://microbit.org/guide/quick/

Once you are done making a code in the online editor (see Coding). You can download your code using the Download button. Save the file onto your micro:bit storage.
The light on the micro:bit will now flash until the code is completely on the chip.

Note You can also save the file on your computer and drag and drop or use copy/paste to get the file onto the micro:bit.



Place your micro:bit in your easyboard to make use of the ports of the board.

Attach the right parts to your easyboard

Power the Easyboard, the micro:bit will be powered too and your code will start to run.

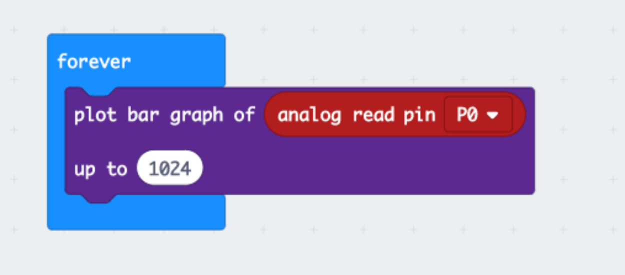
Read inputs

Use data from your sensors in your code!

There are two ways to read data from your input parts. First is the Digital Read function. This function is digital so it can only read your input all the way on or all the way off.

The second one is the Analog Read function. This function can read the entire range of your input part.

To use these functions, connect an input part to any of the input ports of the BYOR Easyboard. Look on the back of the board to see what number the input port has (0, 1, 2 or 3).
To see what is being read by the micro:bit we use the Plot Graph Bar function. Fill in the function as in the image below. Select the Analog Read pin number you want.



```
forever
  plot bar graph of analog read pin P0 ▾
  up to 1024
```

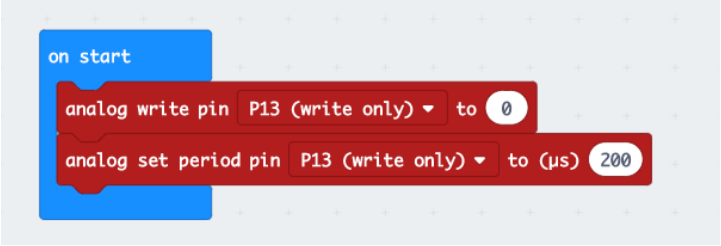
This code will show a bar graph of the input the micro:bit receives from the input part on port 0. Trigger the sensor to see how the micro:bit responds.

Good to know

Analog Write (PWM)

If you want a servo motor to go half way instead of all the way to its outer limits, or an LED to shine on half brightness, you will need to use the analog write function. In order for this function to work properly on the BYOR parts, the pulse width must be set to 200. See the image below to see how to get this to work.
Note Only 3 ports at a time can be set to do an analog write.

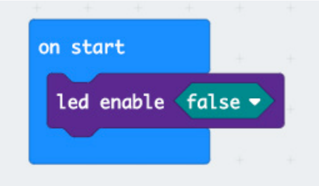
This piece of code will make sure Analog write functions that are called later in the Forever function will work properly with the BYOR parts. This piece of code works for port 13.



```
on start
    analog write pin P13 (write only) ▾ to 0
    analog set period pin P13 (write only) ▾ to (µs) 200
```

Input 3 & micro:bit LEDs

If you want to use analog input on port P3 (ports are shown on the bottom part of the Easyboard) the LED matrix on the Easyboard must be disabled to be able to use it properly for reading out sensor data. Disabling the LEDs on the micro:bit is shown in this code.



```
on start
    led enable false ▾
```

Block reference page

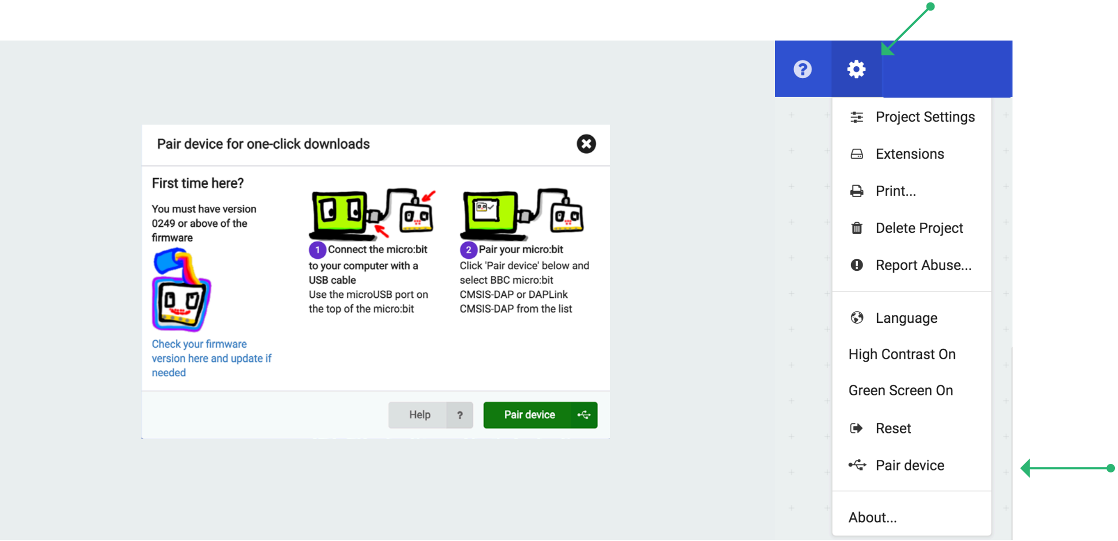
Find out what every block does on the reference page: https://makecode.microbit.org/reference/

Good to know

Pair your micro:bit koppelen

Uploading your code with one click of a button? It's possible! You do need to pair your micro:bit first.

Make sure the micro:bit is connected and click on the gear symbol and on *pair device*.
Follow the instructions on the pop-up to pair your micro:bit to the computer.



Pair device for one-click downloads

First time here?

You must have version 0249 or above of the firmware

Check your firmware version here and update if needed

1 Connect the micro:bit to your computer with a USB cable
Use the microUSB port on the top of the micro:bit

2 Pair your micro:bit
Click 'Pair device' below and select BBC micro:bit CMSIS-DAP or DAPLink CMSIS-DAP from the list

Wireless programming? There is an app for that! Search for micro:bit in the appstore or playstore.

More code!

Support page

Visit our support page for more example-codes.
And a video on how to make your code.

www.byor.nl/en/support

More on micro:bit

The micro:bit is a computer chip on its own with a lot of possibilities. Visit the micro:bit website to learn more!

microbit.org/

micro:bit example codes

Visit the micro:bit webpage for more tutorials on how to use the other cool stuff on the micro:bit!

makecode.microbit.org

BYOR

www.BYORcraft.com

@BYORcraft

@BYORcraft

@BYORcraft

BYORcraft