

SBC-MOTODRIVER3

Motor driver board for DC and stepper motors

TABLE OF CONTENTS

1. General information
3. Changing the I²C address
4. Pin assignment
5. Use with Raspberry Pi
 - 4.1 Connection
 - 4.2 Installation
 - 4.3 Use of stepper motors
5. Use with Raspberry Pi Pico
 - 5.1 Connection
 - 5.2 Installation
 - 5.3 Code example
 - 5.4 Use of stepper motors
6. Use with ARD-ONE-C (Arduino compatible)
 - 6.1 Connection
 - 6.2 Installation
 - 6.3 Code example
 - 6.4 Use of stepper motors
7. Use with micro:bit
 - 7.1 Connection
 - 7.2 Installation
 - 7.3 Code example
 - 7.4 Use of stepper motors
 - 7.5 Pairing the micro:bit
8. Additional information
9. Support

1. GENERAL INFORMATION

Dear Customer,
thank you for choosing our product. In the following, we will show you what you should pay attention to during commissioning and use.

Should you encounter any unexpected problems during use, please feel free to contact us.

With the MotoDriver3 you have the possibility to control and power up to four DC motors or two stepper motors. The connected motors are controlled via the PCA9634 chip, which provides an I2C interface.

2. CHANGING THE I2C ADDRESS

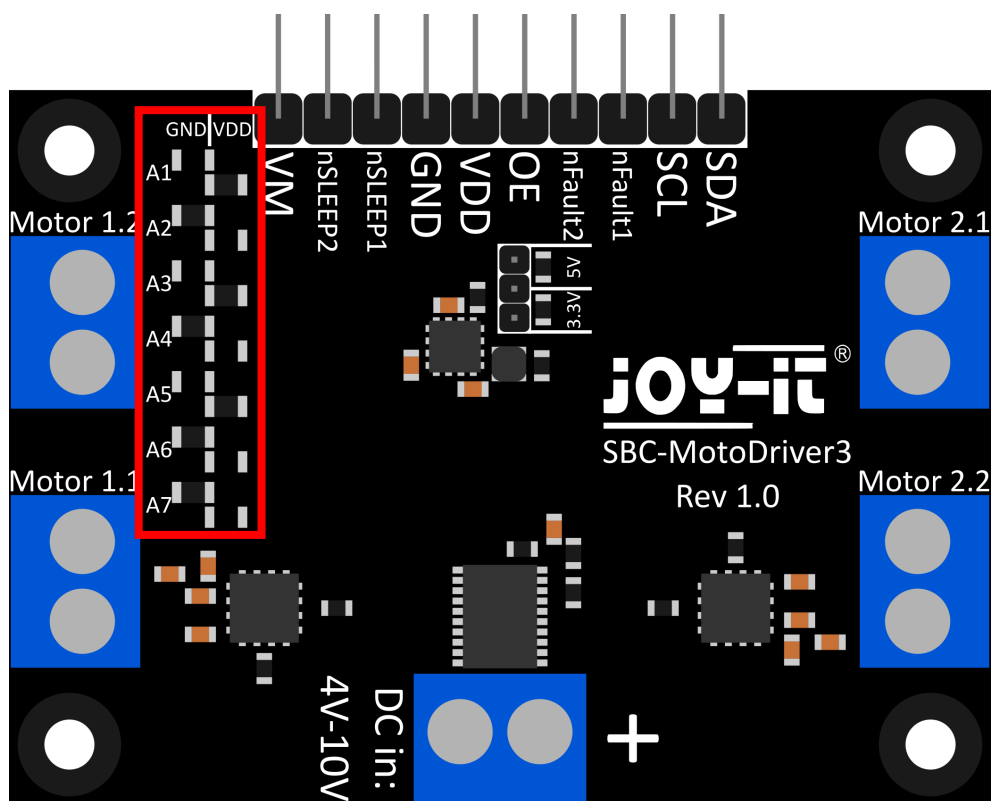
This motor driver board can be configured to 128 possible I2C addresses. These can be chosen freely, but in the further course the default address 0x15 is used.

In the following figure you can see which resistors can be changed to change the I2C address of the motor driver board. You can re-solder these resistors from A1 up to A7 between GND and VDD to get different I2C addresses.

Here the positions of the seven resistors can be translated to a 7-bit binary value. Here we take the standard address of the motor driver board **0x15** which can be translated to **001 0101b**.

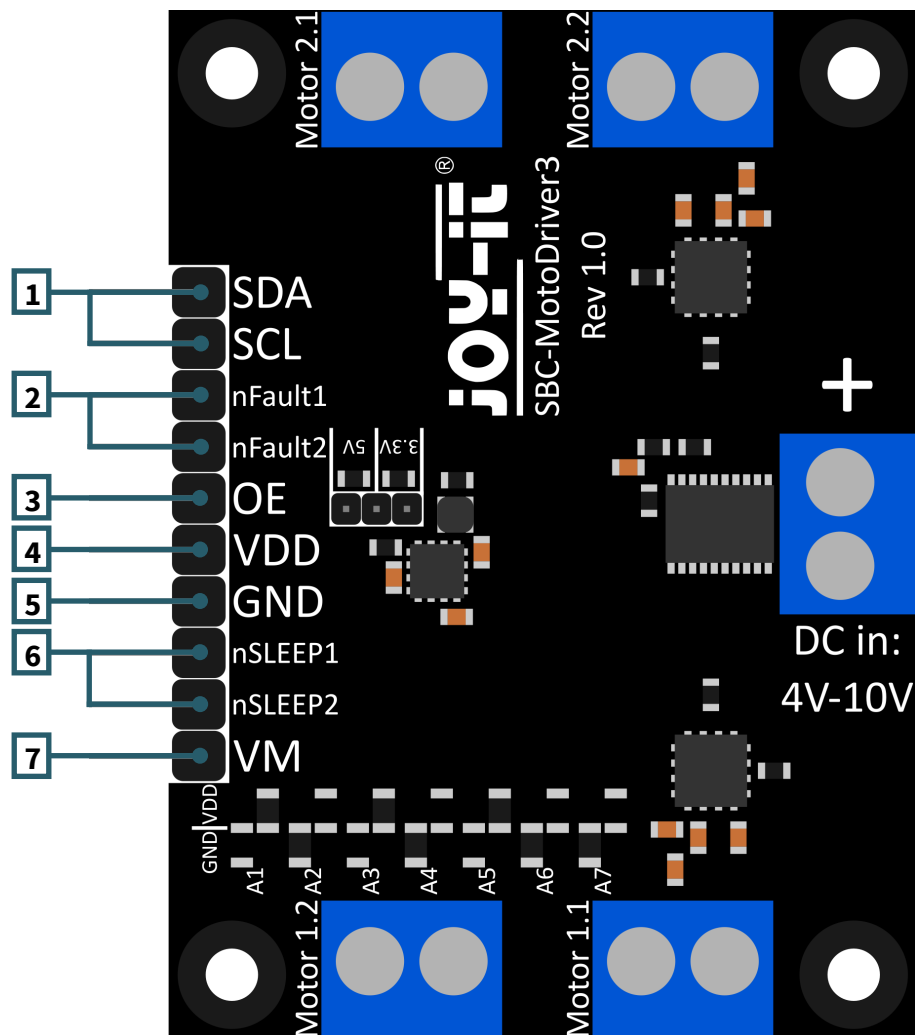
A1 = 1 / A2 = 0 / A3 = 1 / A4 = 0 / A5 = 1 / A6 = 0 / A7 = 0

It should be noted that one should never change the I2C address during operation as this can lead to unpredictable problems.



3. PIN ASSIGNMENT

In this section we briefly explain the functions of each pin.

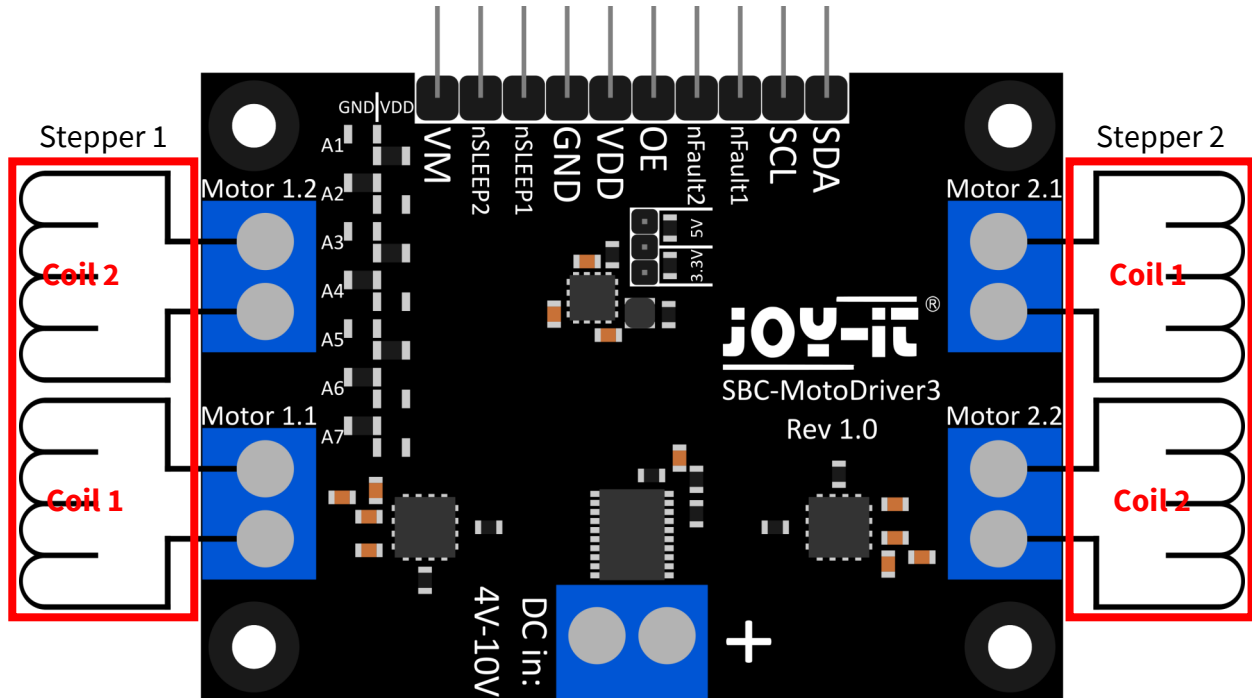


1. **SCL & SDA: I2C interface.** Pins for communication with the driver IC.
2. **nFault1 & nFault2: Status pins of the motor drivers.** Outputs a logical LOW signal if the motor drivers overheat or are loaded with too much current, for example.
3. **VDD: Voltage output of the voltage converter.** This pin outputs the voltage of the voltage converter integrated on the board. Here you can choose between 2 voltages via a jumper on the board. 5V and 3.3V are selectable here.
4. **GND: Ground connection.**
5. **nSLEEP1 & nSLEEP2: Enables the output of the motor drivers.** When these pins are pulled to a logic HIGH signal, the outputs of the motor drivers are enabled and when the pins are pulled to a logic LOW signal, the motor drivers go into low-power sleep mode.
6. **VM: Output of the motor voltage.** This pin outputs the voltage from DC in, which is used to drive the motors, as a usable voltage.

This motor driver board has 5 screw terminals.

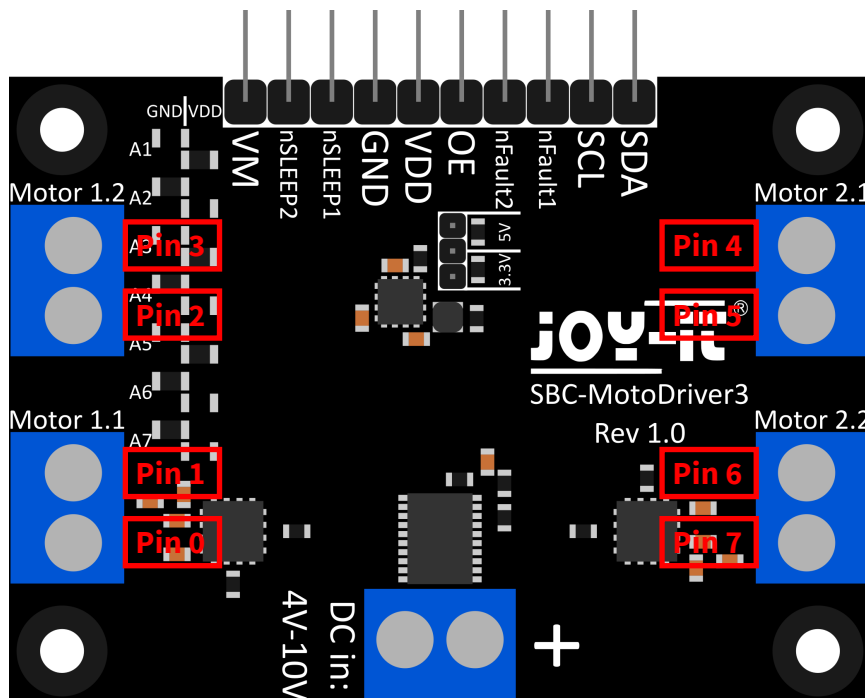
Four of these have the silkscreen "Motor 1.1 | Motor 1.2 | Motor 2.1 and Motor 2.2". These are the terminals for the motors. Either 2 stepper motors or 4 DC motors can be connected here. The last terminal is the voltage input for the motor drivers and the voltage transformer.

It is important that you do not exceed the maximum permissible voltage of the motor driver board and select the motors accordingly, otherwise the motor driver board could be damaged.



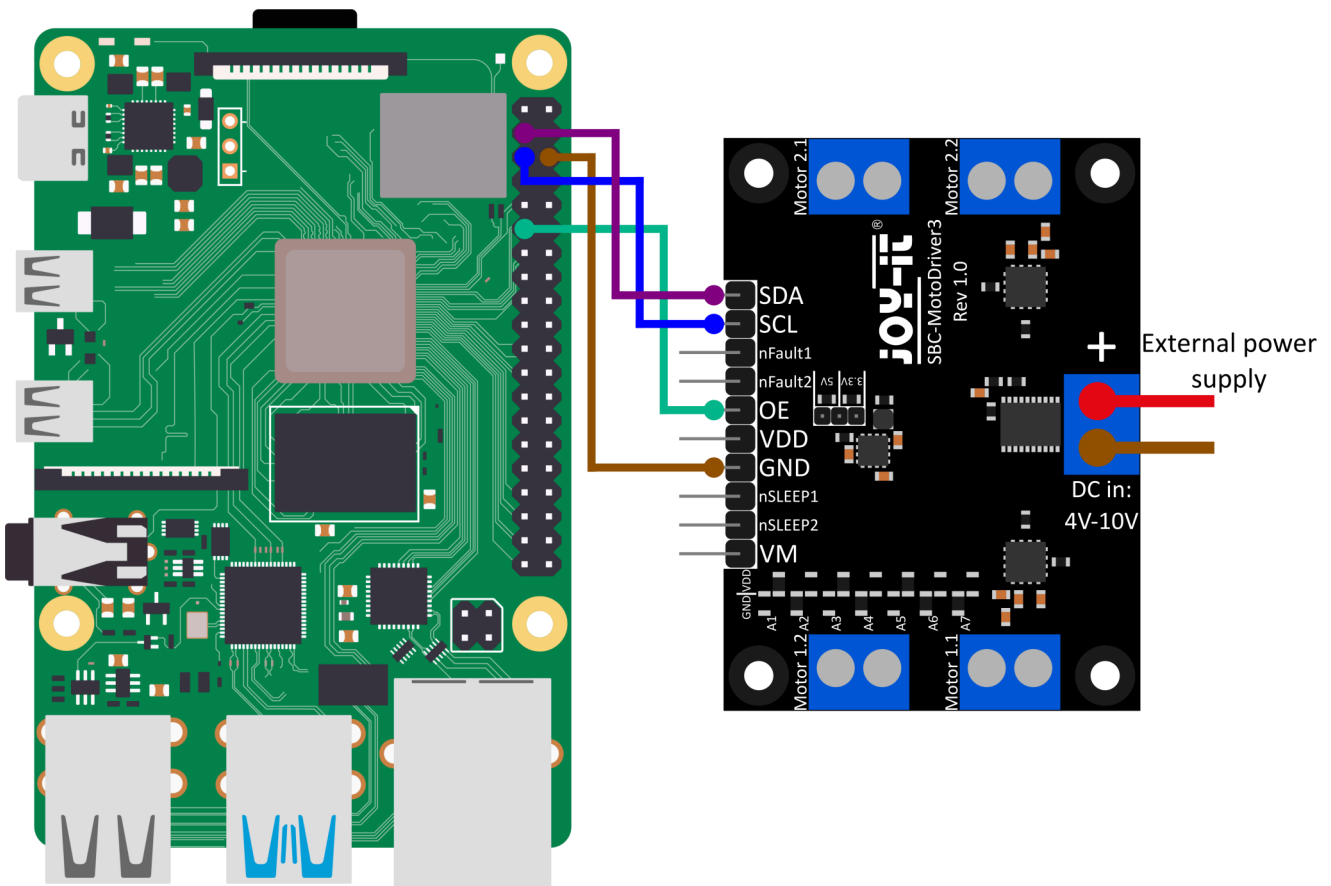
Connect the cables of the coils of your stepper motors to the motor driver board as shown above in order to be able to use them without any problems.

- The "Motor 1.1" terminal will later have pin 0 and pin 1 in the software.
- The "Motor 1.2" terminal will later have pin 2 and pin 3 in the software.
- The "Motor 2.1" terminal will later have pin 4 and pin 5 in the software.
- The "Motor 2.2" terminal will later have pin 6 and pin 7 in the software.



4. USE WITH THE RASPBERRY PI

4.1. Connection



Raspberry Pi	SBC-MotoDriver3	External
SDA (Pin 3)	SDA	-
SCL (Pin 5)	SCL	-
-	nFault1	-
-	nFault2	-
GPIO 17 (Pin 11)	OE	-
-	VDD	-
GND (Pin 6)	GND	-
-	nSLEEP1	-
-	nSLEEP2	-
-	VM/DC in	External voltage source is to be aligned according to the specifications of the motors

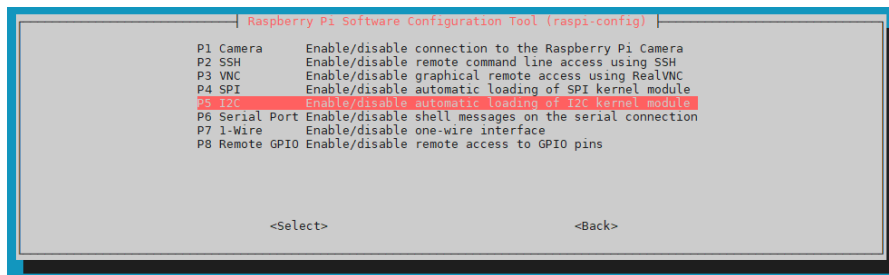
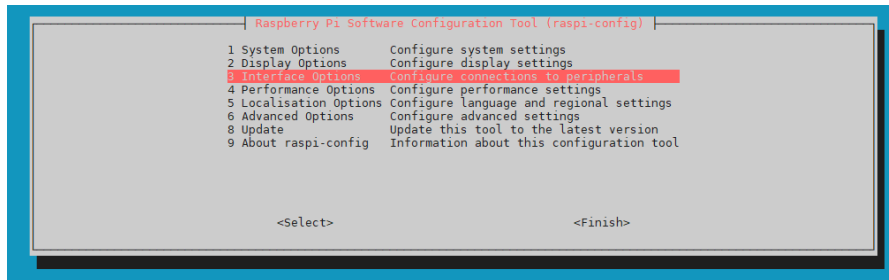
Please note that you cannot connect VDD to your Raspberry Pi because the voltage converter on the board can only supply 1 amp and the Raspberry Pi can need more than 1 amp under load. To avoid damaging the voltage converter ICs and thus the whole board, we advise you not to connect VDD to your Raspberry Pi.

4.2. Installation

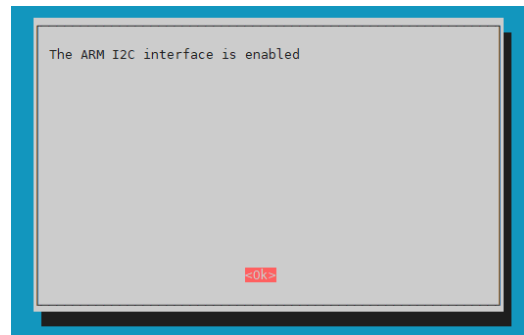
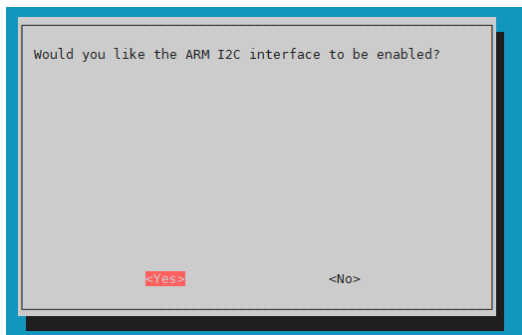
First you need to enable I2C on your Raspberry Pi. To open the configuration, enter the following command:

```
sudo raspi-config
```

There select **3 Interface Options** → **P5 I2C**.



There you get the question **Would you like the ARM I2C Interface to be enabled?**. Confirm this with **Yes**.



You have now successfully enabled I2C. The motor driver board is now accessible under the I2C address **0x15**, which is set by default for this board. The I2C address will be different if you have already configured it before you configured your Raspberry Pi.

Now to find your motor driver board, you need to download and install `i2c-tools` on your Raspberry Pi. You can do this with the following command.

```
sudo apt-get install i2c-tools
```

After downloading and installing `i2c-tools`, you can now start the program via

```
i2cdetect -y 1
```

To find your motor driver board. If you do this, you should see **0x15** and **0x70** as I2C addresses.

Please note that 0x70 is an internal I2C address of the PCA9634 and should not be used if it is not necessary. Because this address acts as hardware reset of the PCA9634.

Next you need to download the required additional libraries, for this you need to run the following commands.

```
sudo apt install -y python3-dev python3-pip
pip3 install smbus2
sudo apt install -y git
```

Next, you need to reboot your Raspberry Pi. To reboot your Raspberry Pi you just have to execute the following command.

```
sudo reboot now
```

For our code example we use the [SBC-MotoDriver3-RaspberryPi](#) library from [Joy-IT](#), which has been released under the [GNU-license](#) and already contains a code example. Use the following command to download this library from GitHub. Alternatively, you can download just the code sample [here](#).

```
git clone https://github.com/joy-it/SBC-MotoDriver3-RaspberryPi.git
```

Now that you have downloaded the library, you need to run the following command to use the example from the motor driver board library.

```
cd SBC-MotoDriver3-RaspberryPi
python3 SBC-MotoDriver3-Example.py
```

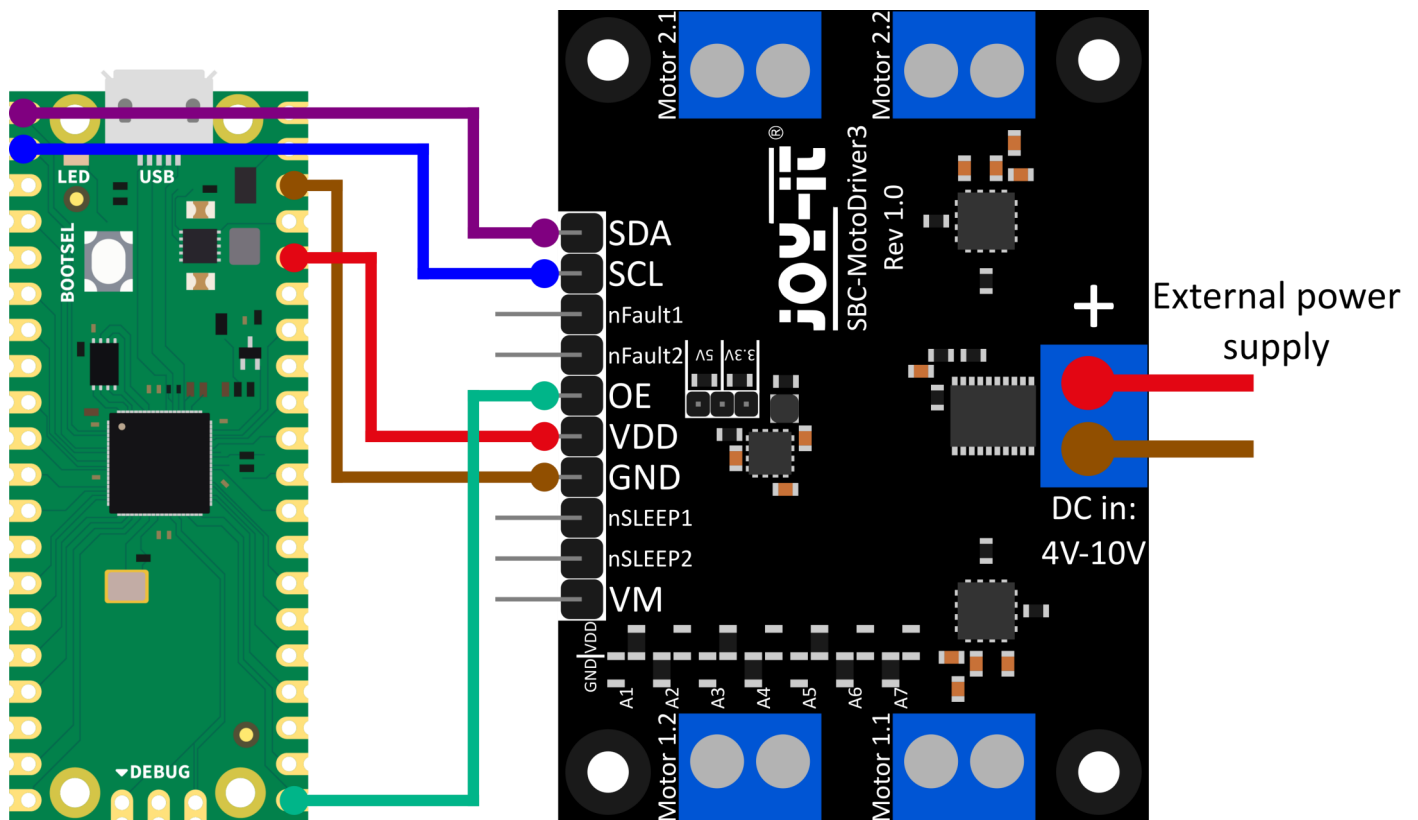
4.3 Use of stepper motors

Please note that only one stepper motor can be connected per side. In terms of hardware, it is important to use the appropriate stepper motors for the permissible voltage range. In addition, regardless of the platform on which you intend to use the product, make sure that you never connect a stepper motor and a DC motor to the same side at the same time. This can cause problems or even damage to the product.

In terms of software, make sure that you do not send the stepper motor function to the same motor driver with the DC motor functions, regardless of the platform on which you plan to use the product. Always keep these functions separate from each other.

5. USE WITH THE RASPBERRY PI PICO

5.1. Connection



Raspberry Pi Pico	SBC-MotoDriver3	External
Pin 0	SDA	-
Pin 1	SCL	-
-	nFault1	-
-	nFault2	-
Pin 16	OE	-
3V3	VDD	-
GND	GND	-
-	nSLEEP1	-
-	nSLEEP2	-
-	VM/DC in	External voltage source is to be aligned according to the specifications of the motors

Please note that when you connect VDD to your Raspberry Pi Pico, you need to set the output voltage of the voltage converter on the board to 3.3V with the jumper to be able to power your Pico via the motor driver board.

5.2. Installation

To use this library on your Raspberry Pi Pico, we recommend that you download the [Thonny Python IDE](#).

Once you have downloaded the Thonny IDE, you can now download the latest version of [Micropython](#) and flash it to your Pico.

After downloading the latest version of Micropython, you can now flash it to your Pico by holding down the BOOTSEL button when connecting your Pico to your PC. Afterwards you can release the BOOTSEL button. The Pico should now open in Explorer on your monitor. Now simply drag in the downloaded file and your Pico is flashed with the latest Micropython version.

5.3. Code example

Next, you can download the library from GitHub. We use the [SBC-MotoDriver3-RaspberryPiPico](#) library from [Joy-IT](#) for our code example, which is released under the [GNU-license](#).

After downloading the *.zip file, you need to unzip it and follow these steps to use the library without any problems.

1. Open the Thonny IDE, go to **"View"** and activate **"Files"** there
2. Create a new folder on your Pico which you name **"lib"**
3. In the **"Files"** window that is now open, open the folder in which the library from GitHub is located
4. Go to the **"lib"** folder on your Pico and right click on the **"SBC_MotoDriver3_Lib.py"** file in your **"Files"** window and download it to your **"lib"** folder on your Pico
5. Leave the **"lib"** folder on your Pico and now download the **"SBC-MotoDriver3-Example.py"** file in your **"Files"** window by right clicking on your Pico

If you have followed these steps, you can already use the code example.

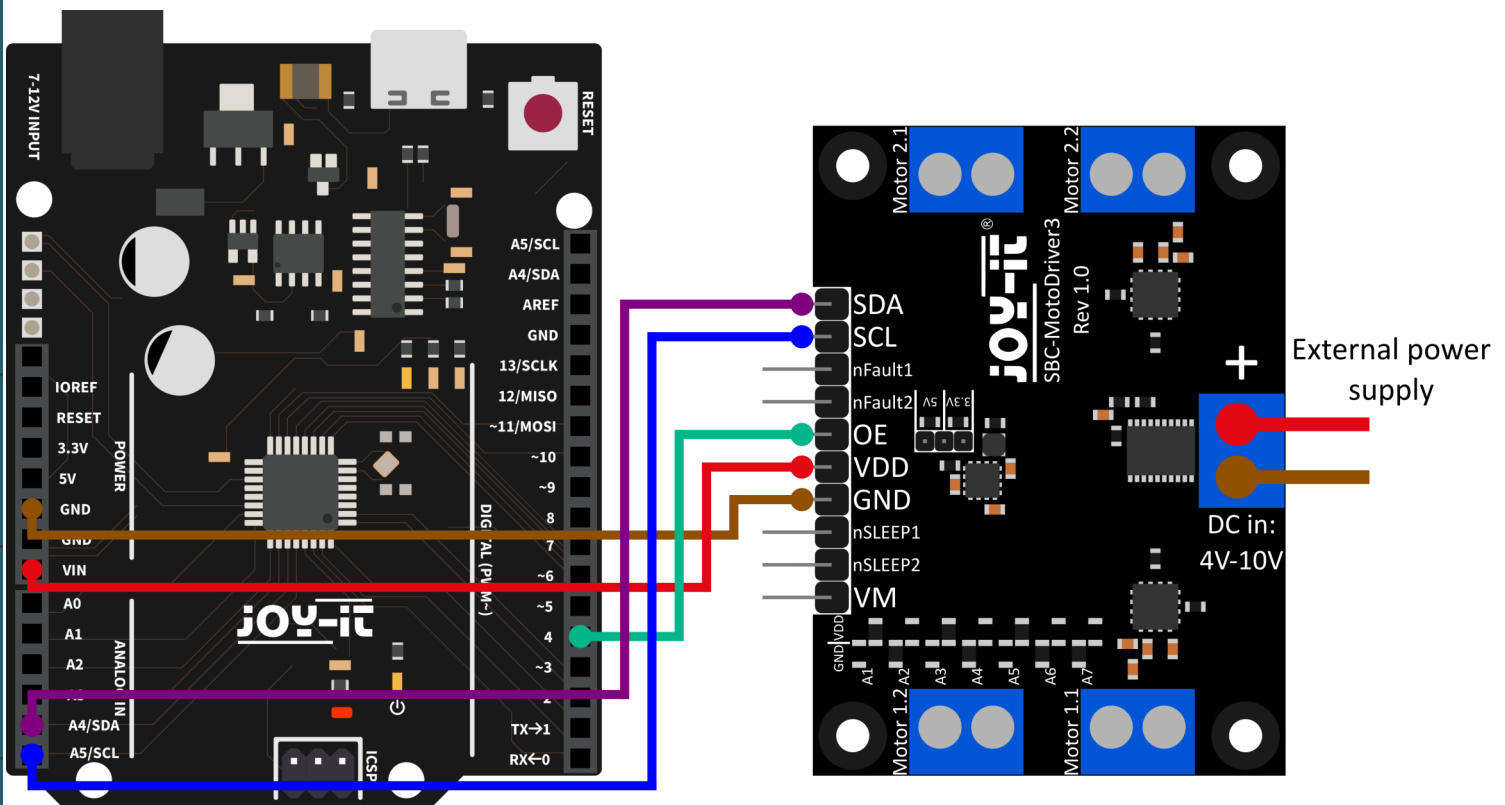
5.4 Use of stepper motors

Please note that only one stepper motor can be connected per side. In terms of hardware, it is important to use the appropriate stepper motors for the permissible voltage range. In addition, regardless of the platform on which you intend to use the product, make sure that you never connect a stepper motor and a DC motor to the same side at the same time. This can cause problems or even damage to the product.

In terms of software, make sure that you do not send the stepper motor function to the same motor driver with the DC motor functions, regardless of the platform on which you plan to use the product. Always keep these functions separate from each other.

6. USE WITH ARD-ONE-C (ARDUINO COMPATIBLE)

6.1. Connection



ARD-ONE-C	SBC-MotoDriver3	External
SDA (A4)	SDA	-
SCL (A5)	SCL	-
-	nFault1	-
-	nFault2	-
D4	OE	-
VIN	VDD	-
GND	GND	-
-	nSLEEP1	-
-	nSLEEP2	-
-	VM/DC in	External voltage source is to be aligned according to the specifications of the motors

Please note that if you connect VDD to your ARD-ONE-C, you have to set the output voltage of the voltage converter on the board with the jumper to 5V to be able to supply your ARD-ONE-C with voltage via the motor driver board.

6.2. Installation

For our code example we use the [SBC-MotoDriver3-Arduino](#) library from [Joy-IT](#), which is released under the [GNU-license](#). To use this library, you need to download it from GitHub. You can install this library by going to **Sketch → Include Library → Add .zip Library...** search for the library you just downloaded and select it.

6.3. Code example

To test your motor driver board with a code example, simply go to **File → Examples → SBC-MotoDriver3-Arduino** in your Arduino IDE and then to **SBC-MotoDriver3-Example...** Click Upload to upload the code sample to your Arduino. You can also download the code sample [here](#).

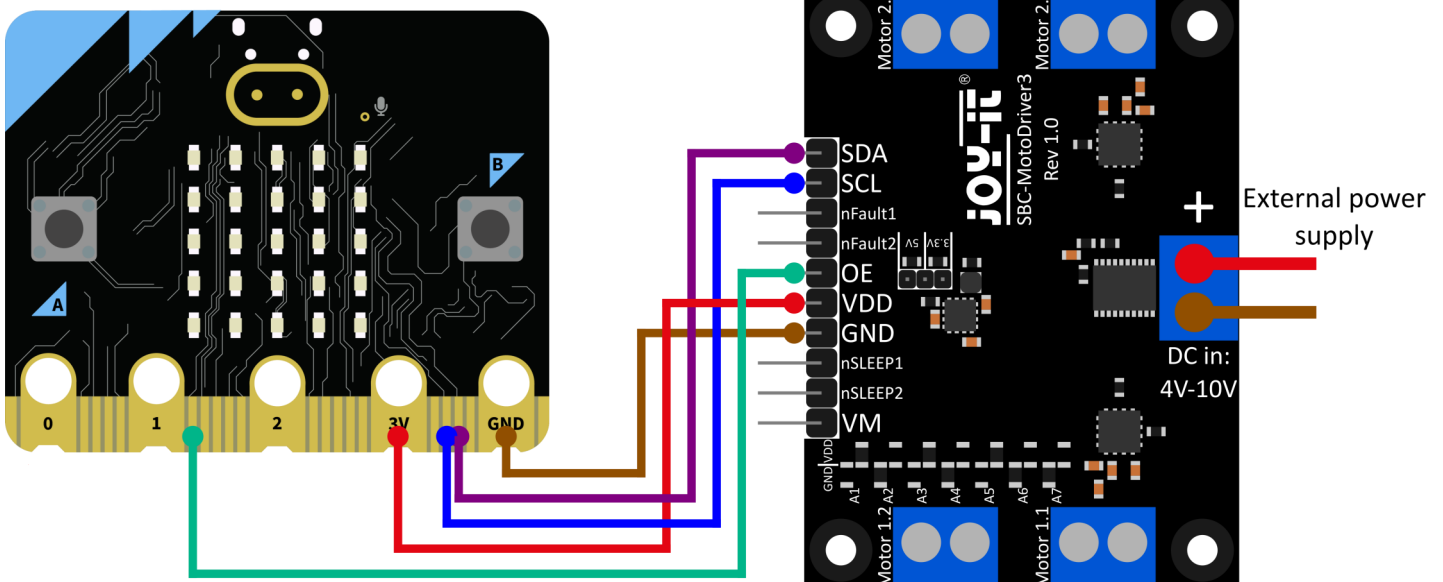
6.4 Use of stepper motors

Please note that only one stepper motor can be connected per side. In terms of hardware, it is important to use the appropriate stepper motors for the permissible voltage range. In addition, regardless of the platform on which you intend to use the product, make sure that you never connect a stepper motor and a DC motor to the same side at the same time. This can cause problems or even damage to the product.

In terms of software, make sure that you do not send the stepper motor function to the same motor driver with the DC motor functions, regardless of the platform on which you plan to use the product. Always keep these functions separate from each other.

7. USE WITH THE MICRO:BIT

7.1. Connection



micro:bit	SBC-MotoDriver3	External
SDA (Pin 20)	SDA	-
SCL (Pin 19)	SCL	-
-	nFault1	-
-	nFault2	-
Pin 8	OE	-
3V	VDD	-
GND	GND	-
-	nSLEEP1	-
-	nSLEEP2	-
-	VM/DC in	External voltage source is to be aligned according to the specifications of the motors

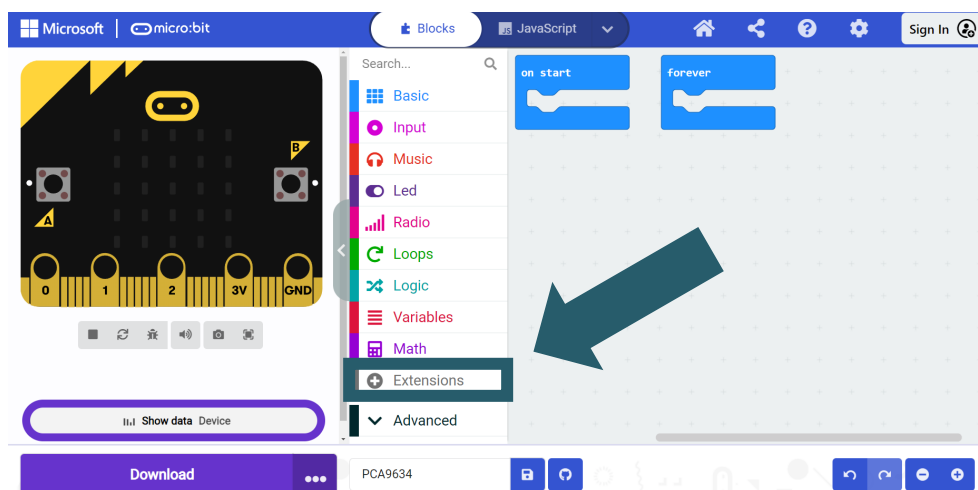
Please note that if you connect VDD to your micro:bit, you have to set the output voltage of the voltage converter on the board to 3.3V with the jumper to be able to supply your micro:bit with voltage via the motor driver board.

You may need to use a breakout board to connect cables to some pins of your Micro:Bit. In this example a breakout board is used. We recommend the [breakout board for micro:bit from Joy-IT](#).

7.2. Installation

For control we recommend the use of the [pxt-SBC-MotoDriver3](#) library, which has been published by [Joy-IT](#) under the [GNU-Lizenz](#).

You can add the library by clicking on **Extensions** on the [Makecode page](#)



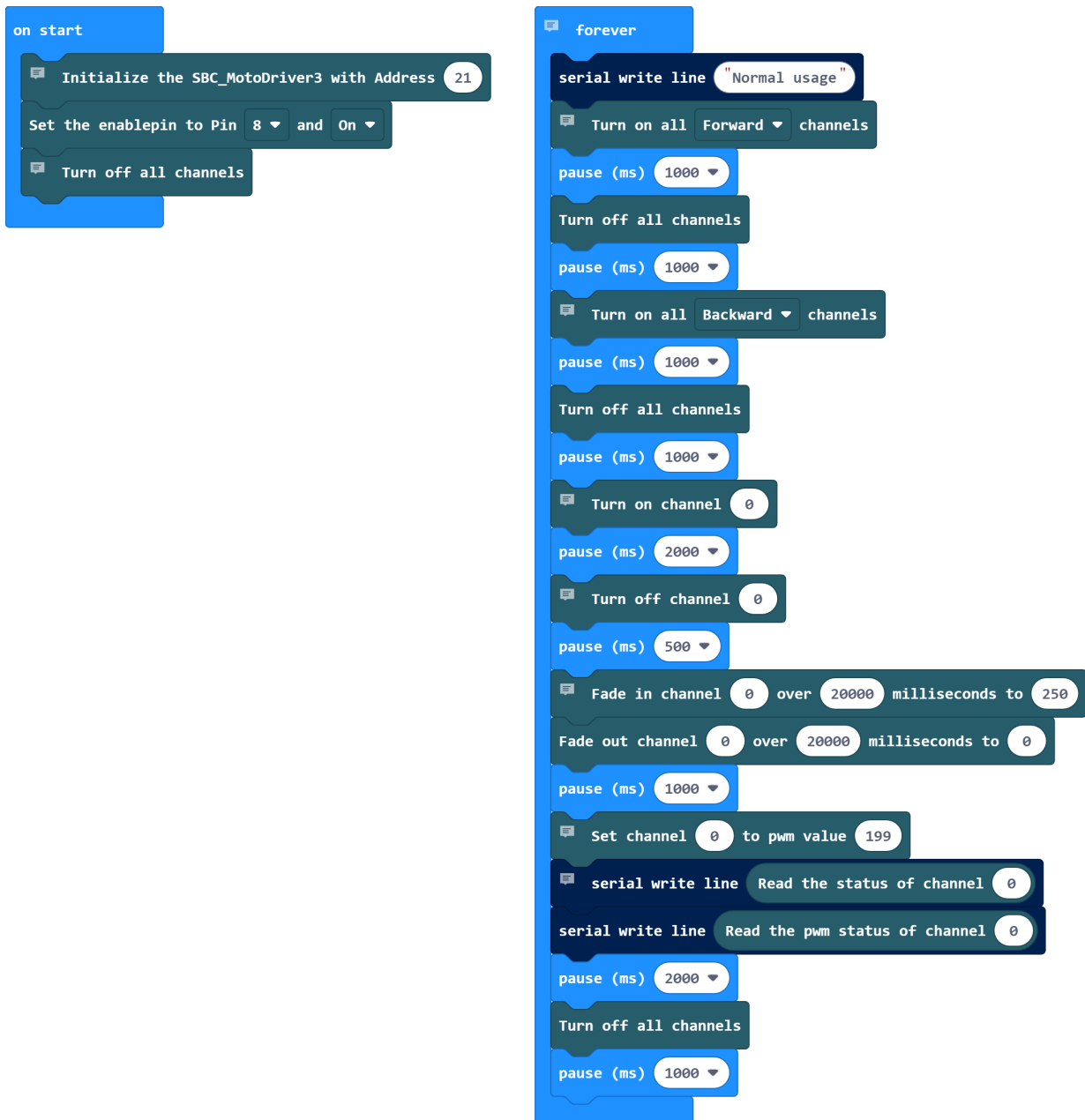
and then typing https://github.com/joy-it/pxt-SBC_MotoDriver3 in the search bar. After you have done that, all you have to do is click on the extension to add it automatically for your current project.

7.3. Code example

To test your Motor driver board, you can download a code sample from us [here](#) or simply rebuild the code yourself as shown below.

If you have decided to download our code sample, then **after unzipping the file** you can simply drag and drop the contents onto the Makecode web page and release to import the file and open it automatically.

Micro:bit code example Figure:



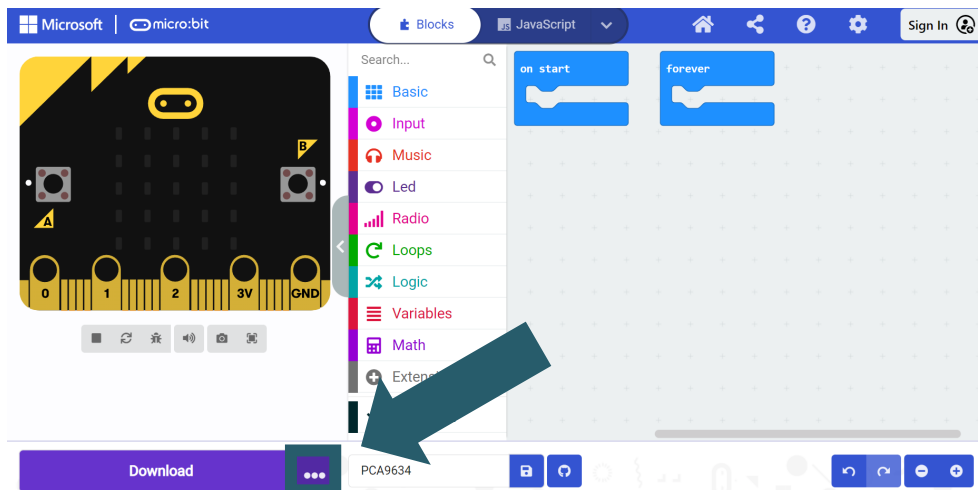
7.4 Use of stepper motors

Please note that only one stepper motor can be connected per side. In terms of hardware, it is important to use the appropriate stepper motors for the permissible voltage range. In addition, regardless of the platform on which you intend to use the product, make sure that you never connect a stepper motor and a DC motor to the same side at the same time. This can cause problems or even damage to the product.

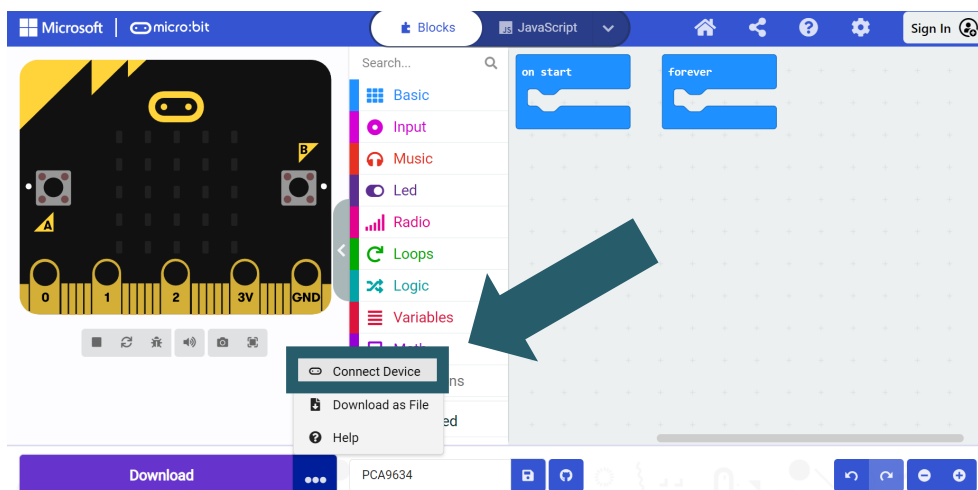
In terms of software, make sure that you do not send the stepper motor function to the same motor driver with the DC motor functions, regardless of the platform on which you plan to use the product. Always keep these functions separate from each other.

7.5. Pairing the Micro:Bit

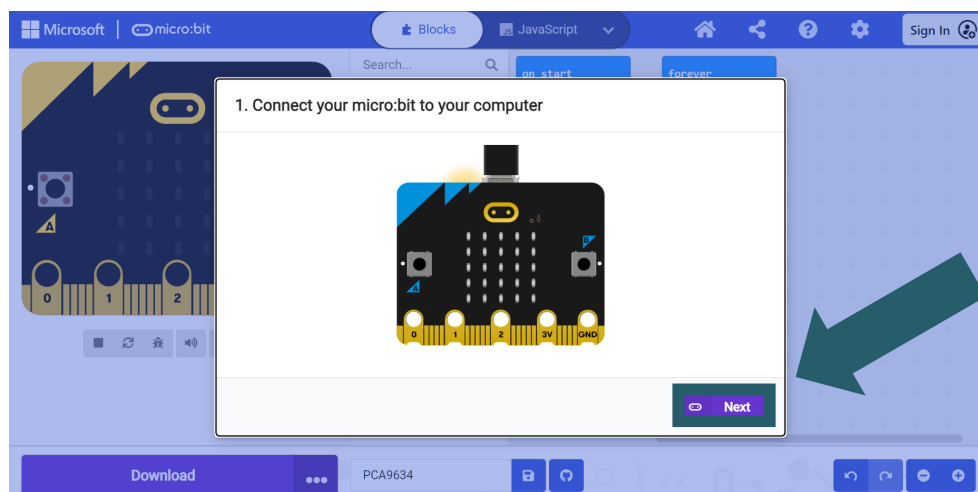
Click on the button located to the right of Upload/Download.



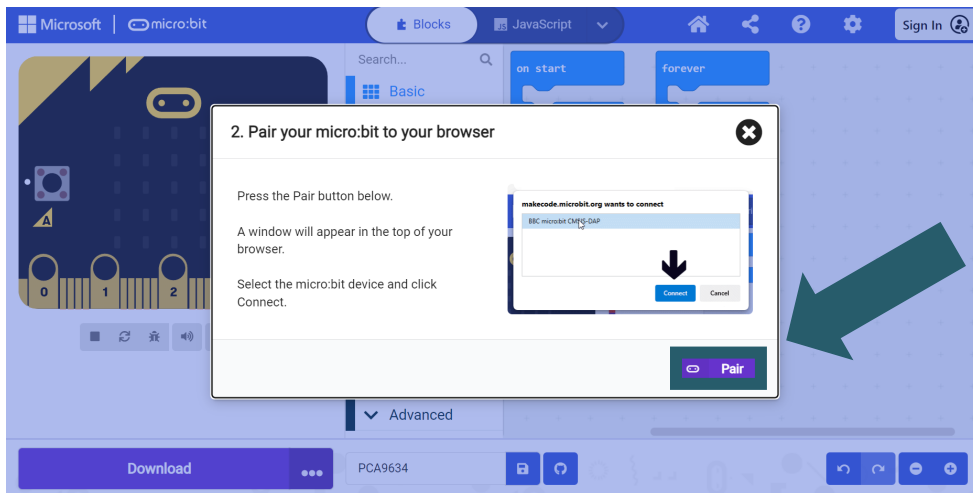
Click **Connect Device**.



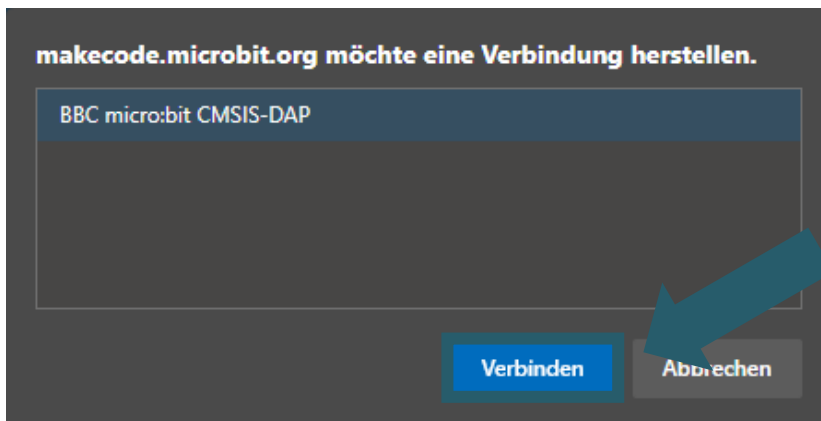
Click **Next**.



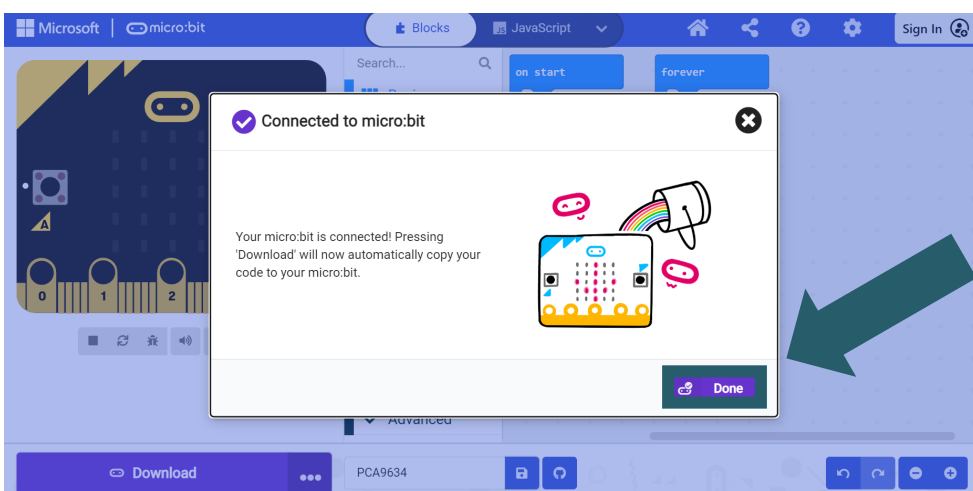
Click on **Pair**.



Select your Micro:Bit in the opened window and click Connect to be able to connect and use it.



The last thing you need to do is click on **Done** to be able to upload your code. You upload your code by clicking Download.



8. ADDITIONAL INFORMATION

Our information and take-back obligations according to the Electrical and Electronic Equipment Act (ElektroG)

Symbol on electrical and electronic equipment:



This crossed-out dustbin means that electrical and electronic appliances do not belong in the household waste. You must return the old appliances to a collection point.

Before handing over waste batteries and accumulators that are not enclosed by waste equipment must be separated from it.

Return options:

As an end user, you can return your old device (which essentially fulfills the same function as the new device purchased from us) free of charge for disposal when you purchase a new device.

Small appliances with no external dimensions greater than 25 cm can be disposed of in normal household quantities independently of the purchase of a new appliance.

Possibility of return at our company location during opening hours:

SIMAC Electronics GmbH, Pascalstr. 8, D-47506 Neukirchen-Vluyn, Germany

Possibility of return in your area:

We will send you a parcel stamp with which you can return the device to us free of charge. Please contact us by email at Service@joy-it.net or by telephone.

Information on packaging:

If you do not have suitable packaging material or do not wish to use your own, please contact us and we will send you suitable packaging.

9. SUPPORT

If there are still any issues pending or problems arising after your purchase, we will support you by e-mail, telephone and with our ticket support system.

Email: service@joy-it.net

Ticket system: <http://support.joy-it.net>

Telephone: +49 (0)2845 9360-50 (Mon. - Thu.: 10:00 - 17:00 o'clock,
Fri.: 10:00 - 14:30 Uhr)

For further information please visit our website:

www.joy-it.net