

SBC-MOTODRIVER3

Motortreiberboard für DC und Stepper Motoren

1. Allgemeine Informationen
2. Änderung der I²C Adresse
3. Pinbelegung
4. Verwendung mit dem Raspberry Pi
 - 4.1 Anschluss
 - 4.2 Installation
 - 4.3 Verwendung von Stepper Motoren
5. Verwendung mit Raspberry Pi Pico
 - 5.1 Anschluss
 - 5.2 Installation
 - 5.3 Codebeispiel
 - 5.4 Verwendung von Stepper Motoren
6. Verwendung mit ARD-ONE-C (Arduino Kompatibel)
 - 6.1 Anschluss
 - 6.2 Installation
 - 6.3 Codebeispiel
 - 6.4 Verwendung von Stepper Motoren
7. Verwendung mit micro:bit
 - 7.1 Anschluss
 - 7.2 Installation
 - 7.3 Codebeispiel
 - 7.4 Verwendung von Stepper Motoren
 - 7.5 Kopplung des micro:bit
8. Sonstige Informationen
9. Support

1. ALLGEMEINE INFORMATIONEN

Sehr geehrte*r Kunde *in,
vielen Dank, dass Sie sich für unser Produkt entschieden haben. Im Folgenden zeigen wir Ihnen, was bei der Inbetriebnahme und der Verwendung zu beachten ist.

Sollten Sie während der Verwendung unerwartet auf Probleme stoßen, so können Sie uns selbstverständlich gerne kontaktieren.

Mit dem MotoDriver3 haben Sie die Möglichkeit, bis zu vier Gleichstrommotoren oder zwei Schrittmotoren anzusteuern und zu betreiben. Die angeschlossenen Motoren werden über den PCA9634-Chip angesteuert, der über eine I2C-Schnittstelle verfügt.

2. ÄNDERUNG DER I2C-ADRESSE

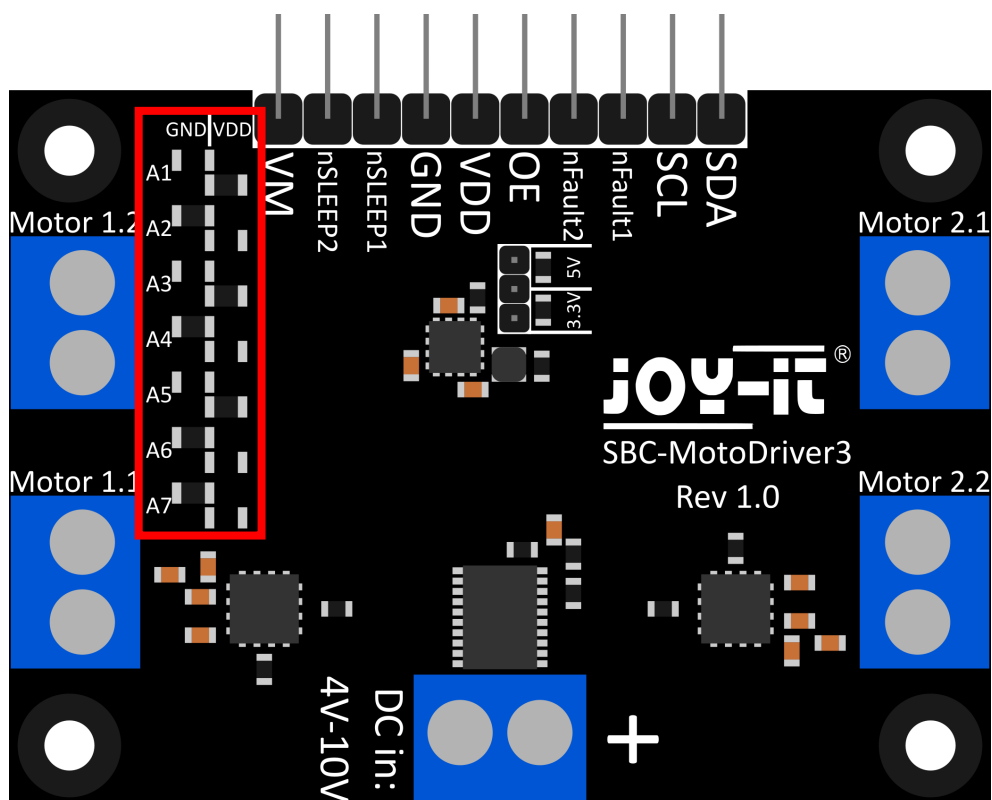
Diese Motortreiberkarte kann auf 128 mögliche I2C-Adressen konfiguriert werden. Diese können frei gewählt werden, aber im weiteren Verlauf wird die Standardadresse 0x15 verwendet.

In der folgenden Abbildung sehen Sie, welche Widerstände verändert werden können, um die I2C-Adresse der Motortreiberkarte zu ändern. Man kann diese Widerstände von A1 bis A7 zwischen GND und VDD umlöten, um verschiedene I2C-Adressen zu erhalten.

Hierbei lassen sich die Positionen der sieben Widerstände auf einen 7-Bit binären Wert übersetzen. Hierbei nehmen wir einmal die Standardadresse des Motortreiberboards **0x15**, welche sich zu **001 0101b** übersetzen lässt:

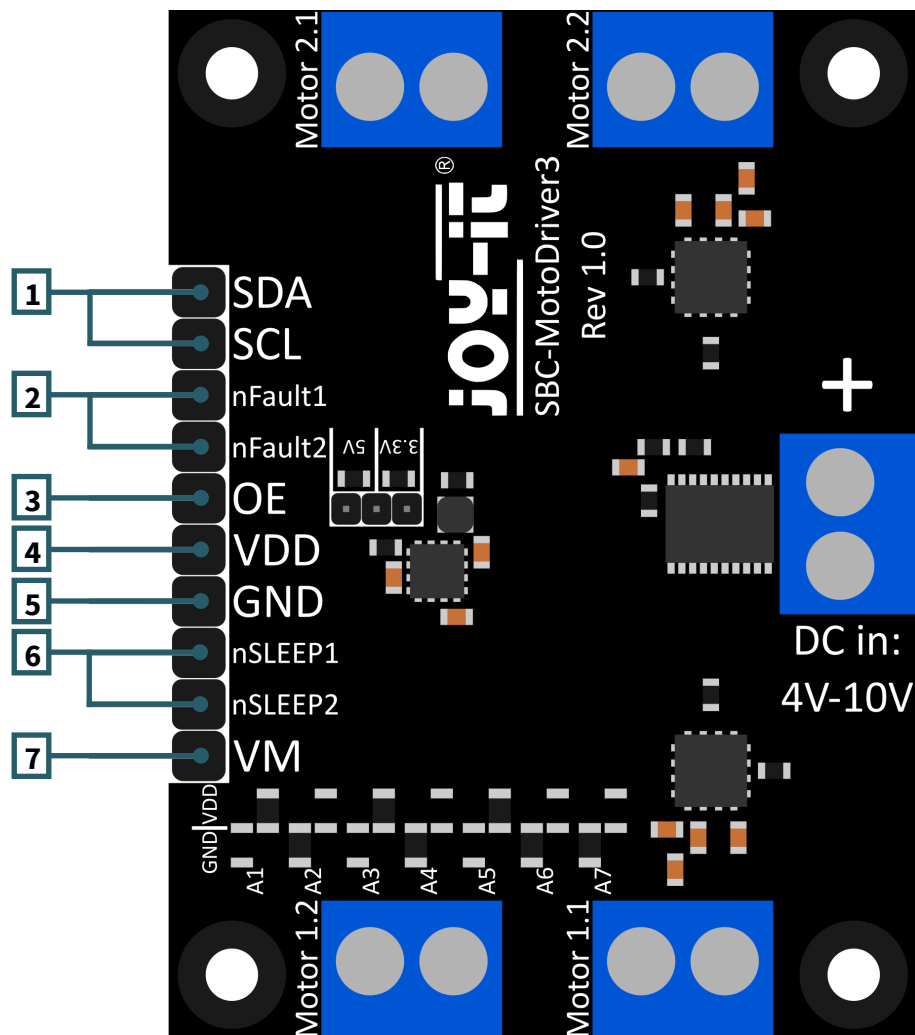
$$A1 = 1 / A2 = 0 / A3 = 1 / A4 = 0 / A5 = 1 / A6 = 0 / A7 = 0$$

Es ist zu beachten, dass man die I2C-Adresse niemals während des Betriebs ändern sollte, da dies zu unvorhersehbaren Problemen führen kann.



3. PINBELEGUNG

In diesem Abschnitt erklären wir Ihnen kurz die Funktionen der einzelnen Pins.

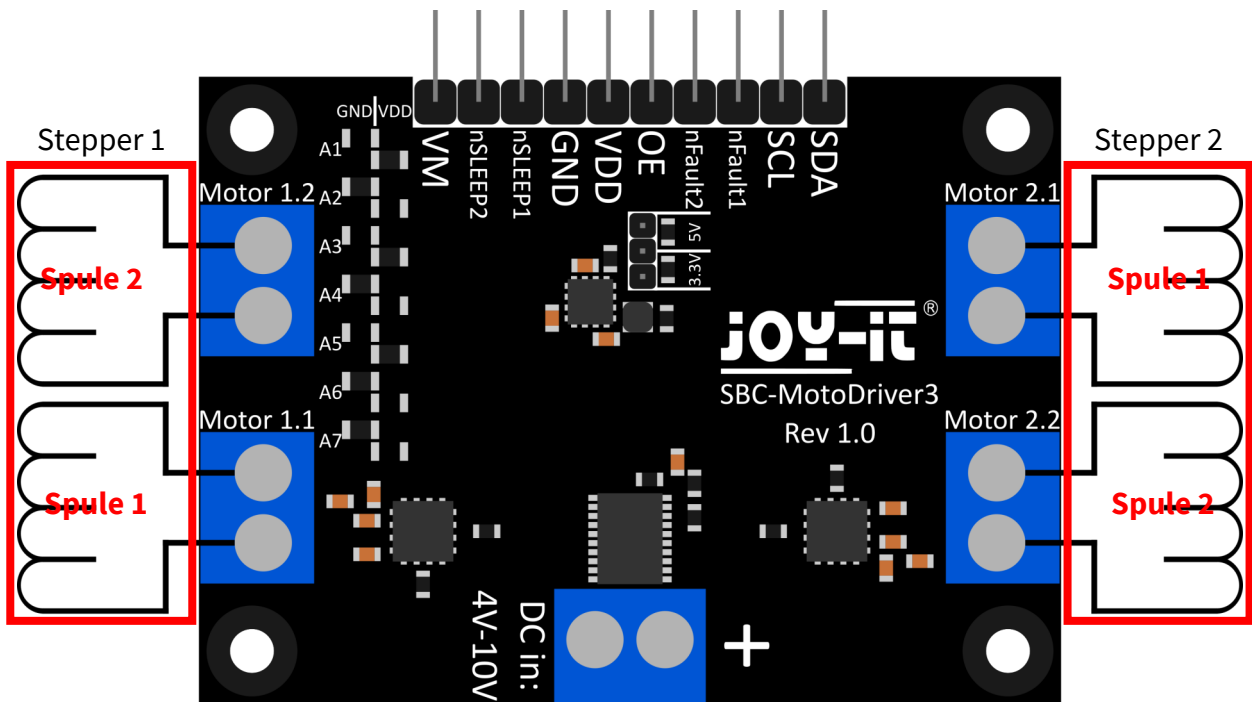


1. **SCL & SDA: I2C Schnittstelle.** Pins zur Kommunikation mit dem Treiber IC.
2. **nFault1 & nFault2:** Status Pins der Motortreiber. Gibt ein logisches LOW Signal aus, wenn die Motortreiber z. B. überhitzen oder mit zu viel Strom belastet werden.
3. **OE: Schaltet den Output des Treiber ICs frei.** Wenn der Pin auf ein logisches LOW Signal gezogen wird, wird der Output des Treiber ICs freigegeben.
4. **VDD: Spannungsausgang der Spannungswandlung.** Dieser Pin gibt die Spannung des auf dem Board integrierten Spannungswandlers aus. Hier kann man zwischen 2 Spannungen per Jumper auf dem Board wählen. 5V und 3,3V sind hier wählbar.
5. **GND: Ground Verbindung.**
6. **nSLEEP1 & nSLEEP2: Schaltet die Ausgabe der Motortreiber frei.** Wenn diese Pins auf ein logisches HIGH Signal gezogen werden, werden die Outputs der Motortreiber freigegeben und wenn die Pins auf ein logisches LOW Signal gezogen werden, gehen die Motortreiber in den low-power sleep Modus.
7. **VM: Ausgabe der Motorspannung.** Dieser Pin gibt die Spannung von DC in, welche für das Betreiben der Motoren genutzt wird, als nutzbare Spannung wieder aus.

Dieses Motortreiberboard hat 5 Schraubterminals.

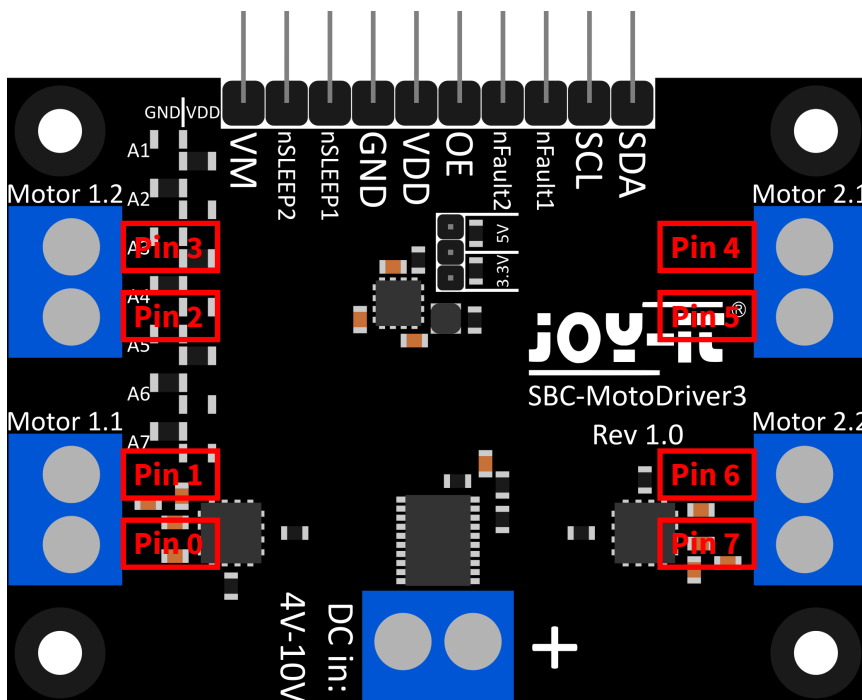
Davon haben 4 den Silkscreen "Motor 1.1 | Motor 1.2 | Motor 2.1 und Motor 2.2". Dies sind die Terminals für die Motoren. Hier können entweder 2 Schrittmotoren oder 4 Gleichstrommotoren angeschlossen werden. Das letzte Terminal ist der Spannungseingang für die Motortreiber und den Spannungswandler.

Wichtig ist, dass Sie die maximal zulässige Spannung der Motortreiberplatine nicht überschreiten und die Motoren entsprechend auswählen, da sonst die Motortreiberplatine beschädigt werden könnte.



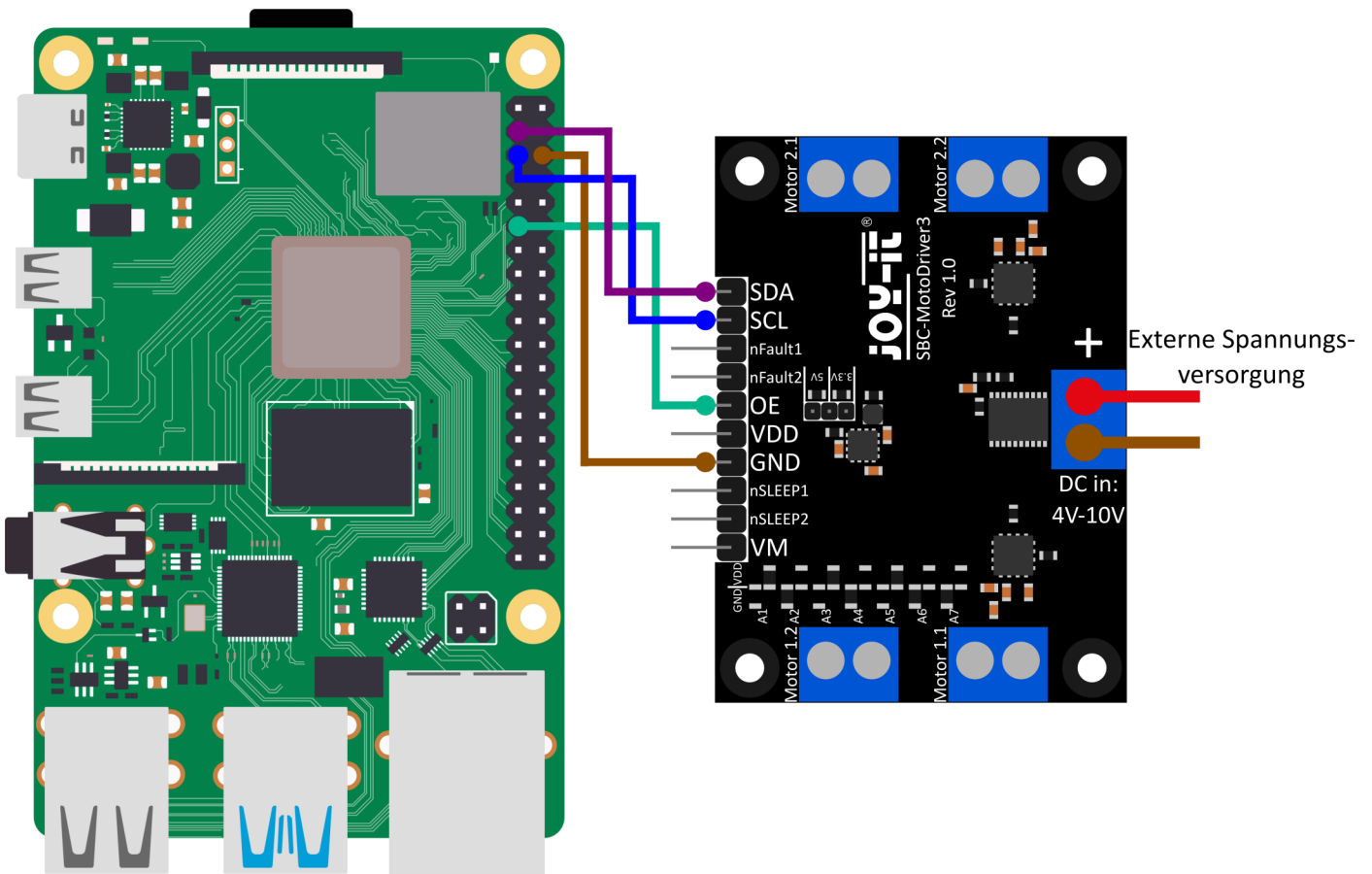
Verbinden Sie die Kabel der Spulen Ihrer Stepper Motoren so wie oben gezeigt mit dem Motortreiberboard um diese ohne Probleme benutzen zu können.

Das Terminal "Motor 1.1" hat in der Software später Pin 0 und Pin 1.
Das Terminal "Motor 1.2" hat in der Software später Pin 2 und Pin 3.
Das Terminal "Motor 2.1" hat in der Software später Pin 4 und Pin 5.
Das Terminal "Motor 2.2" hat in der Software später Pin 6 und Pin 7.



4. VERWENDUNG MIT DEM RASPBERRY PI

4.1. Anschluss



Raspberry Pi	SBC-MotoDriver3	Extern
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	Externe Spannungsquelle ist nach den Spezifikationen der Motoren auszurichten

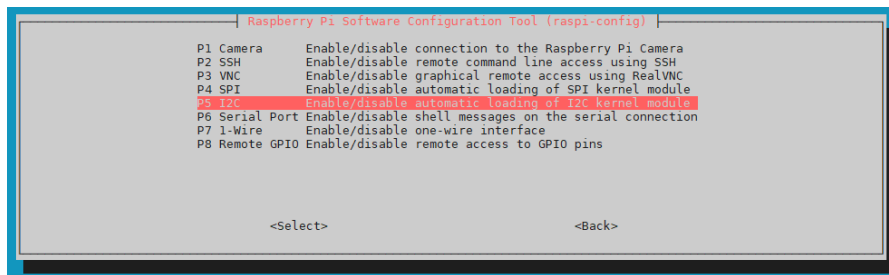
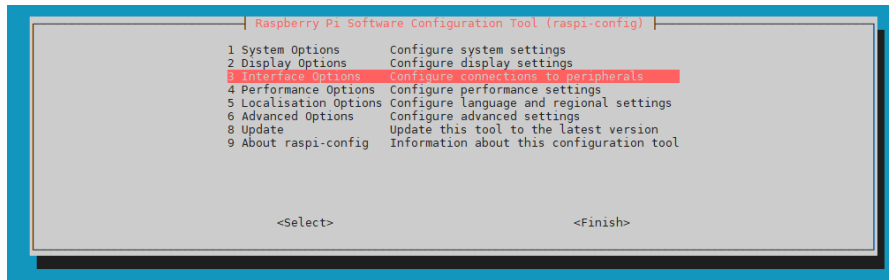
Bitte beachten Sie, dass Sie VDD nicht an Ihren Raspberry Pi anschließen können, da der Spannungswandler auf dem Board nur 1 Ampere liefern kann und der Raspberry Pi unter Last mehr als 1 Ampere benötigen kann. Um eine Beschädigung der Spannungswandler ICs und damit des gesamten Boards zu vermeiden, raten wir Ihnen, VDD nicht mit Ihrem Raspberry Pi zu verbinden.

4.2. Installation

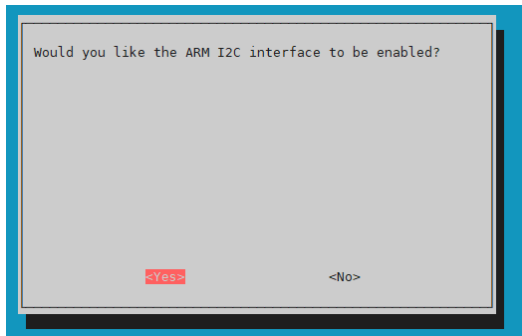
Zunächst müssen Sie I2C auf Ihrem Raspberry Pi aktivieren. Um die Konfiguration zu öffnen, geben Sie den folgenden Befehl ein:

```
sudo raspi-config
```

Wählen Sie dort **3 Interface Options** → **P5 I2C** aus.



Dort bekommen Sie die Frage gestellt **Would you like the ARM I2C Interface to be enabled?**. Diese bestätigen Sie mit **Yes**.



"Die Aktivierung von I2C war erfolgreich. Das Motortreiberboard ist nun unter der standardmäßigen I2C-Adresse **0x15** erreichbar. Beachten Sie, dass die I2C-Adresse unterschiedlich sein kann, falls Sie diese bereits vor der Konfiguration Ihres Raspberry Pi angepasst haben."

Um jetzt Ihr Motortreiberboard zu finden, müssen Sie **i2c-tools** auf Ihrem Raspberry Pi herunterladen und installieren. Dies können Sie mit dem folgenden Befehl machen.

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

Nach dem Herunterladen und Installieren von i2c-tools können Sie nun über

```
i2cdetect -y 1
```

Ihr Motortreiberboard finden. Wenn Sie dies tun, sollten Sie **0x15** und **0x70** als I2C Adressen angezeigt bekommen.

Hierbei ist zu beachten das 0x70 eine interne I2C Adresse des PCA9634 ist und nicht angesteuert werden sollte wenn es nicht nötig ist. Da diese Adresse als Hardware Reset des PCA9634 fungiert.

Als nächstes müssen Sie die benötigten Zusatzbibliotheken herunterladen, dazu müssen Sie die folgenden Befehle ausführen.

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

Als nächstes müssen Sie einen Neustart Ihres Raspberry Pis vornehmen. Zum neu starten Ihres Raspberry Pis müssen Sie nur den folgenden Befehl ausführen.

```
sudo reboot now
```

Wir verwenden für unser Codebeispiel die [SBC-MotoDriver3-RaspberryPi](#) Bibliothek von [Joy-IT](#), welche unter der [GNU-Lizenz](#) veröffentlicht wurde und bereits ein Codebeispiel enthält. Mit dem folgenden Befehl laden Sie sich diese Bibliothek von GitHub herunter. Alternativ können Sie auch nur das Codebeispiel [hier](#) herunterladen.

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

Nachdem Sie die Bibliothek heruntergeladen haben, müssen Sie nun den folgenden Befehl ausführen, um das Beispiel aus der Bibliothek des Motortreiberboards benutzen zu können.

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

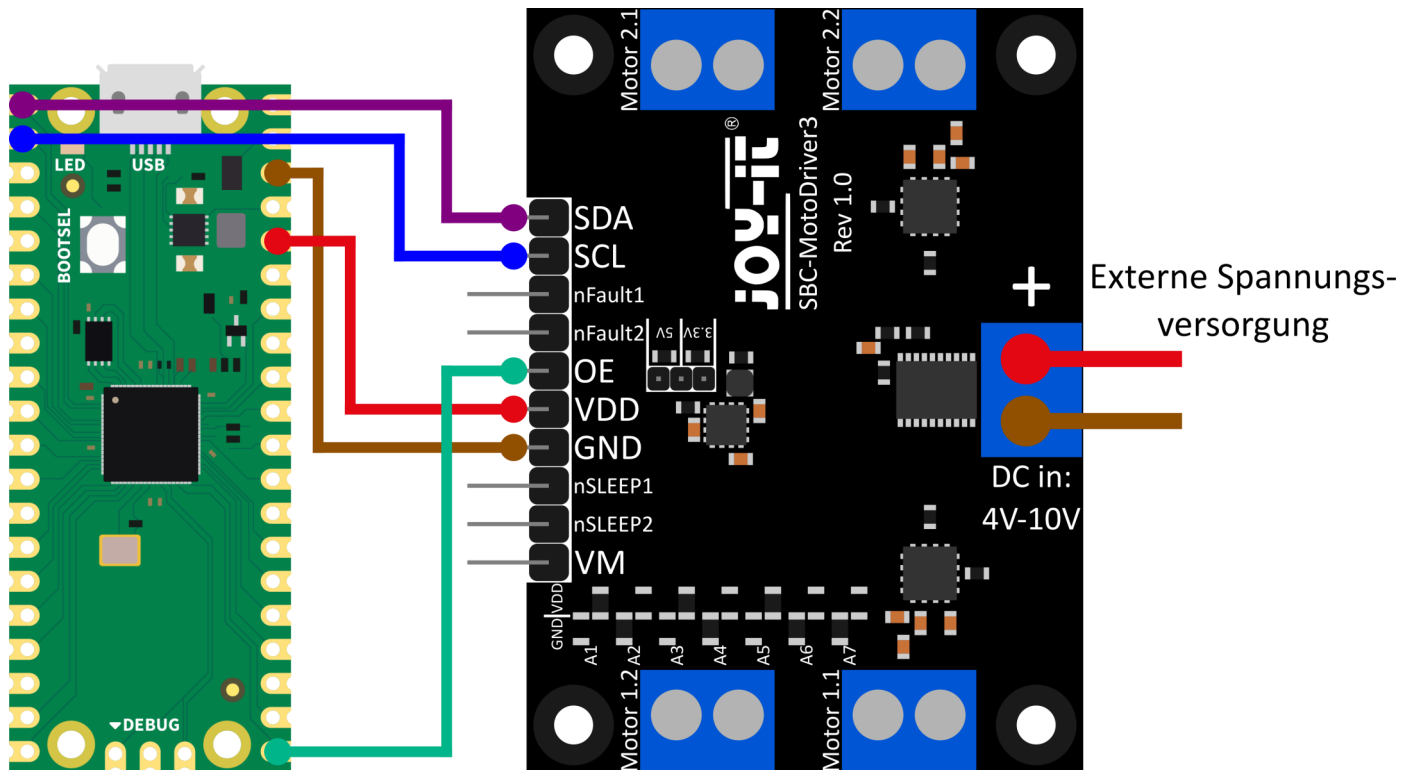
4.3 Verwendung von Stepper Motoren

Bitte beachten Sie, dass pro Seite immer nur ein Schrittmotor angeschlossen werden kann. In Bezug auf die Hardware ist es wichtig, die passenden Schrittmotoren für den zulässigen Spannungsbereich zu verwenden. Darüber hinaus sollten Sie sicherstellen, dass Sie, unabhängig von der Plattform, auf der Sie das Produkt verwenden möchten, niemals gleichzeitig einen Schrittmotor und einen Gleichstrommotor an dieselbe Seite anschließen. Dies kann zu Problemen oder sogar zu Beschädigungen des Produkts führen.

In Bezug auf die Software sollten Sie sicherstellen, dass Sie die Schrittmotorfunktion nicht mit den Funktionen für Gleichstrommotoren an den gleichen Motortreiber senden, unabhängig von der Plattform, auf der Sie das Produkt verwenden möchten. Halten Sie diese Funktionen stets getrennt voneinander.

5. VERWENDUNG MIT DEM RASPBERRY PI PICO

5.1. Anschluss



Raspberry Pi Pico	SBC-MotoDriver3	Extern
Pin 0	SDA	-
Pin 1	SCL	-
-	nFault1	-
-	nFault2	-
Pin 16	OE	-
3V3	VDD	-
GND	GND	-
-	nSLEEP1	-
-	nSLEEP2	-
-	VM/DC in	Externe Spannungsquelle ist nach den Spezifikationen der Motoren auszurichten

Bitte beachten Sie, dass wenn Sie VDD an Ihren Raspberry Pi Pico anschließen, Sie die Ausgangsspannung des Spannungswandlers auf dem Board mit dem Jumper auf 3,3V einstellen müssen, um Ihren Pico über das Motortreiberboard mit Spannung versorgen zu können.

5.2. Installation

Zur Benutzung dieser Bibliothek auf Ihrem Raspberry Pi Pico empfehlen wir Ihnen sich die [Thonny Python IDE](#) herunterzuladen.

Wenn Sie die Thonny IDE heruntergeladen haben, können Sie nun die neueste Version von [Micropython](#) herunterladen und auf Ihren Pico flashen.

Nachdem Sie die neueste Version von Micropython heruntergeladen haben, können Sie diese jetzt auf Ihren Pico flashen, indem Sie den BOOTSEL Knopf gedrückt halten, wenn Sie ihren Pico mit Ihrem PC verbinden. Anschließend können Sie den BOOTSEL Knopf loslassen. Auf Ihrem Monitor sollte sich jetzt der Pico im Explorer geöffnet haben. Dort ziehen Sie jetzt einfach die so eben heruntergeladene Datei herein und schon ist Ihr Pico geflasht mit der neuesten Micropython Version.

5.3. Codebeispiel

Als Nächstes können Sie die Bibliothek von GitHub herunterladen. Wir verwenden für unser Codebeispiel die [SBC-MotoDriver3-RaspberryPiPico](#) Bibliothek von [Joy-IT](#), welche unter der [GNU-Lizenz](#) veröffentlicht wurde.

Nach dem Download der *.zip Datei müssen Sie diese entpacken und folgende Schritte befolgen, um die Bibliothek ohne Probleme benutzen zu können.

1. Öffnen Sie die Thonny IDE, gehen Sie zu **"View"** und aktivieren Sie dort **"Files"**
2. Erstellen Sie einen neuen Ordner auf Ihrem Pico welchen Sie **"lib"** nennen
3. Öffnen Sie in dem nun offenen **"Files"** Fenster den Ordner in dem sich die entpackte Bibliothek von GitHub befindet
4. Gehen Sie in den **"lib"** Ordner auf Ihrem Pico und laden Sie per Rechtsklick auf die **"SBC_MotoDriver3_Lib.py"** Datei in Ihrem **"Files"** Fenster diese in Ihren **"lib"** Ordner auf Ihren Pico herunter
5. Verlassen Sie den **"lib"** Ordner auf Ihrem Pico und laden Sie nun die **"SBC-MotoDriver3-Example.py"** Datei in Ihrem **"Files"** Fenster per Rechtsklick auf Ihren Pico herunter

Wenn Sie diese Schritte befolgt haben, können Sie das Codebeispiel auch schon direkt benutzen.

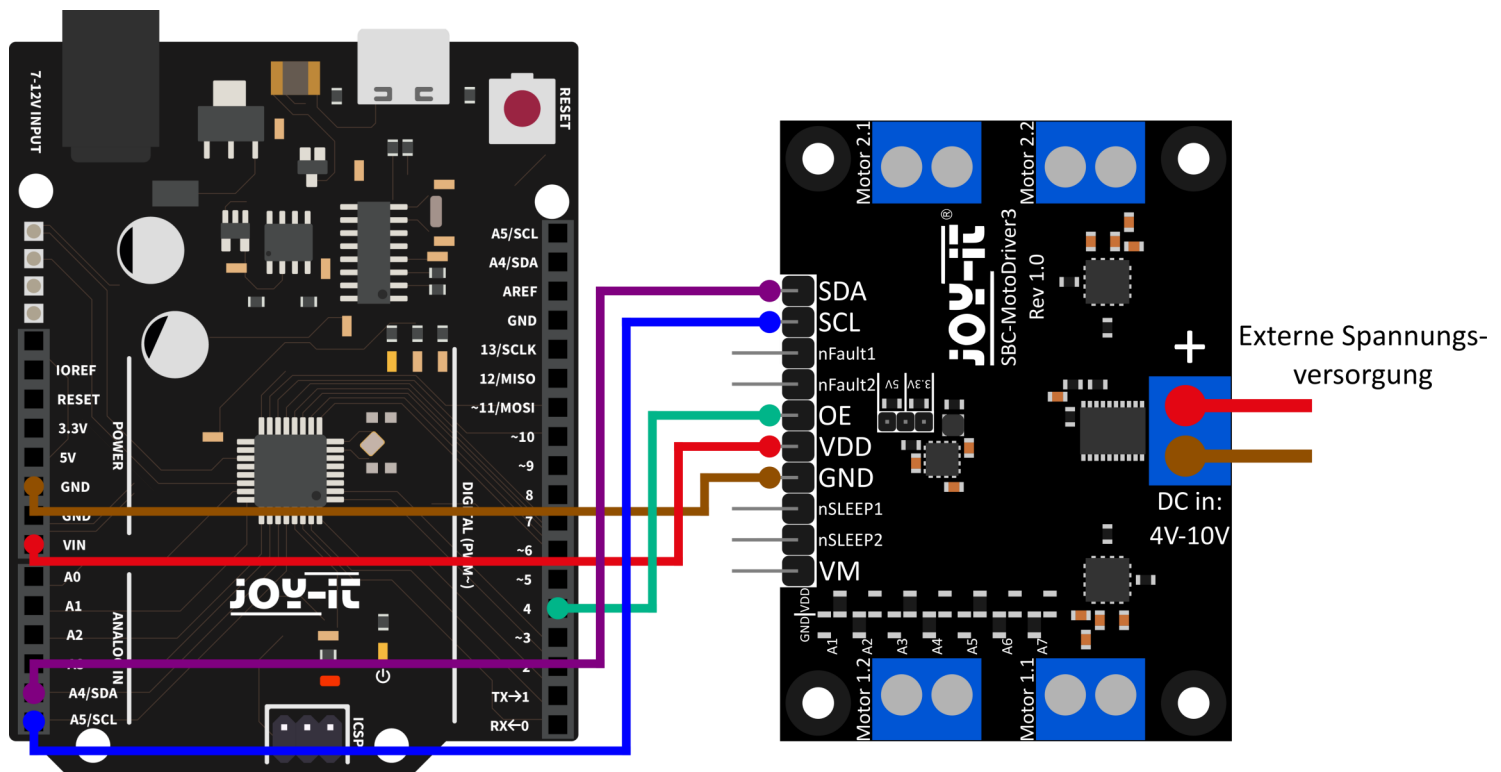
5.4 Verwendung von Stepper Motoren

Bitte beachten Sie, dass pro Seite immer nur ein Schrittmotor angeschlossen werden kann. In Bezug auf die Hardware ist es wichtig, die passenden Schrittmotoren für den zulässigen Spannungsbereich zu verwenden. Darüber hinaus sollten Sie sicherstellen, dass Sie, unabhängig von der Plattform, auf der Sie das Produkt verwenden möchten, niemals gleichzeitig einen Schrittmotor und einen Gleichstrommotor an dieselbe Seite anschließen. Dies kann zu Problemen oder sogar zu Beschädigungen des Produkts führen.

In Bezug auf die Software sollten Sie sicherstellen, dass Sie die Schrittmotorfunktion nicht mit den Funktionen für Gleichstrommotoren an den gleichen Motortreiber senden, unabhängig von der Plattform, auf der Sie das Produkt verwenden möchten. Halten Sie diese Funktionen stets getrennt voneinander.

6. VERWENDUNG MIT ARD-ONE-C (ARDUINO KOMPATIBEL)

6.1. Anschluss



ARD-ONE-C	SBC-MotoDriver3	Extern
SDA (A4)	SDA	-
SCL (A5)	SCL	-
-	nFault1	-
-	nFault2	-
D4	OE	-
VIN	VDD	-
GND	GND	-
-	nSLEEP1	-
-	nSLEEP2	-
-	VM/DC in	Externe Spannungsquelle ist nach den Spezifikationen der Motoren auszurichten

Bitte beachten Sie, dass wenn Sie VDD an Ihren ARD-ONE-C anschließen, Sie die Ausgangsspannung des Spannungswandlers auf dem Board mit dem Jumper auf 5V einstellen müssen, um Ihren ARD-ONE-C über das Motortreiberboard mit Spannung versorgen zu können.

6.2. Installation

Für unser Codebeispiel verwenden wir die [SBC-MotoDriver3-Arduino](#) Bibliothek von [Joy-IT](#), welche unter der [GNU-Lizenz](#) veröffentlicht wurde. Um diese Bibliothek benutzen zu können, müssen Sie diese von GitHub herunterladen. Sie können diese Bibliothek installieren, indem Sie unter **Sketch** → **Bibliothek einbinden** → **.zip-Bibliothek hinzufügen...** nach der so eben heruntergeladenen Bibliothek suchen und diese auswählen.

6.3. Codebeispiel

Zum Testen Ihres Motortreiberboards mit einem Codebeispiel gehen Sie in Ihrer Arduino IDE einfach auf **Datei** → **Beispiele** → **SBC-MotoDriver3-Arduino** und dann auf **SBC-MotoDriver3-Example...** Klicken Sie auf **Hochladen**, um das Codebeispiel auf Ihren Arduino zu laden. Sie können das Codebeispiel auch [hier](#) herunterladen.

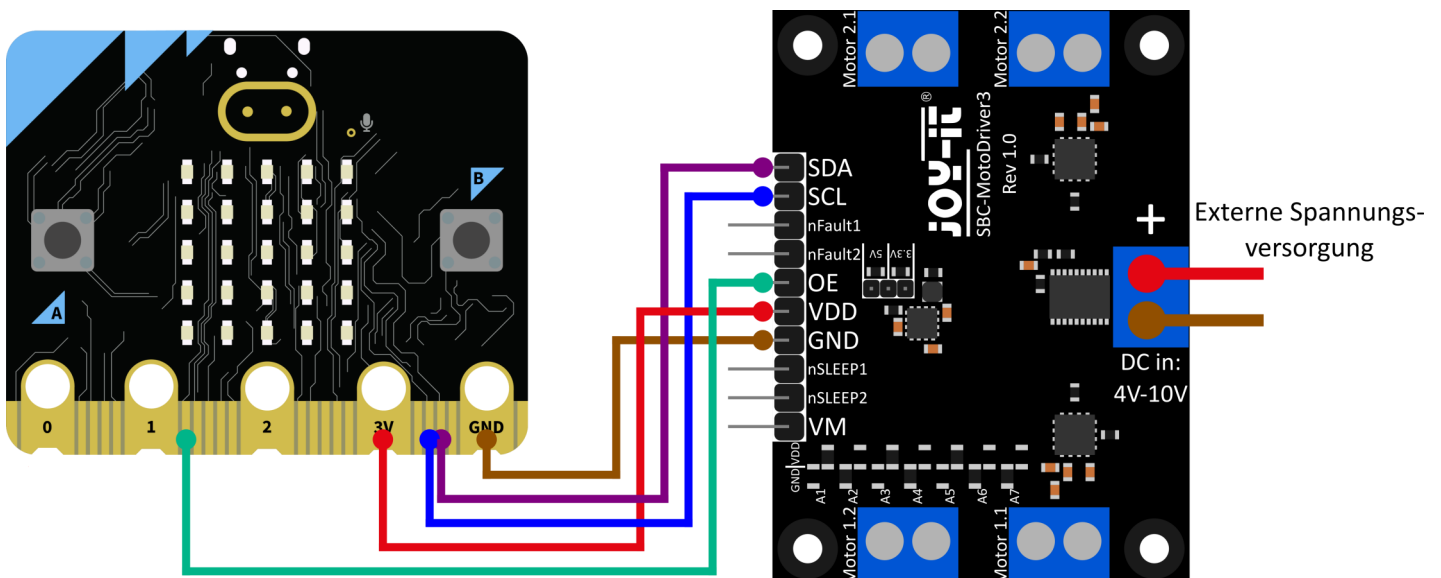
6.4 Verwendung von Stepper Motoren

Bitte beachten Sie, dass pro Seite immer nur ein Schrittmotor angeschlossen werden kann. In Bezug auf die Hardware ist es wichtig, die passenden Schrittmotoren für den zulässigen Spannungsbereich zu verwenden. Darüber hinaus sollten Sie sicherstellen, dass Sie, unabhängig von der Plattform, auf der Sie das Produkt verwenden möchten, niemals gleichzeitig einen Schrittmotor und einen Gleichstrommotor an dieselbe Seite anschließen. Dies kann zu Problemen oder sogar zu Beschädigungen des Produkts führen.

In Bezug auf die Software sollten Sie sicherstellen, dass Sie die Schrittmotorfunktion nicht mit den Funktionen für Gleichstrommotoren an den gleichen Motortreiber senden, unabhängig von der Plattform, auf der Sie das Produkt verwenden möchten. Halten Sie diese Funktionen stets getrennt voneinander.

7. VERWENDUNG MIT DEM MICRO:BIT

7.1. Anschluss



micro:bit	SBC-MotoDriver3	Extern
SDA (Pin 20)	SDA	-
SCL (Pin 19)	SCL	-
-	nFault1	-
-	nFault2	-
Pin 8	OE	-
3V	VDD	-
GND	GND	-
-	nSLEEP1	-
-	nSLEEP2	-
-	VM/DC in	Externe Spannungsquelle ist nach den Spezifikationen der Motoren anzurichten

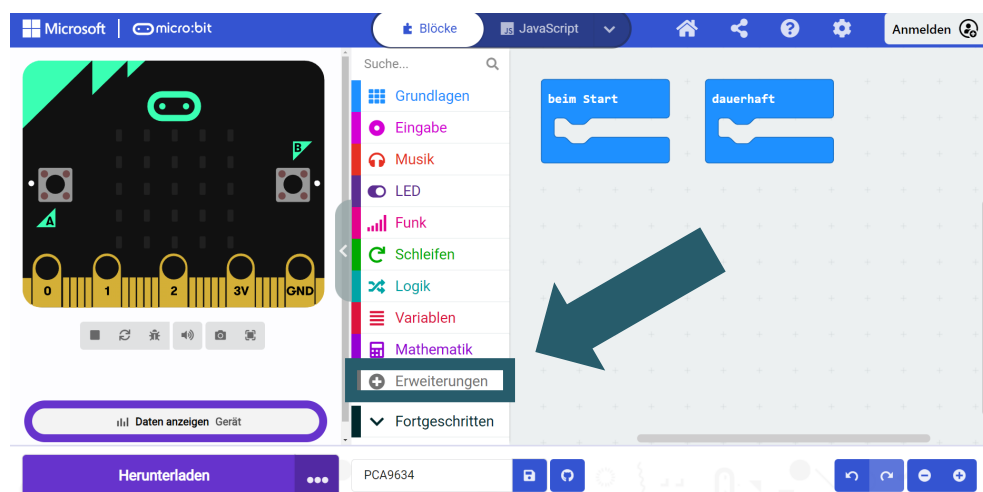
Bitte beachten Sie, dass wenn Sie VDD an Ihren micro:bit anschließen, Sie die Ausgangsspannung des Spannungswandlers auf dem Board mit dem Jumper auf 3,3V einstellen müssen, um Ihren micro:bit über das Motortreiberboard mit Spannung versorgen zu können.

Möglicherweise müssen Sie ein Breakoutboard verwenden, um an manche Pins Ihres Micro:Bits Kabel anschließen zu können. Bei diesem Beispiel wird ein Breakoutboard benutzt. Wir empfehlen hierfür das [Breakoutboard für Micro:Bit von Joy-IT](#).

7.2. Installation

Zur Ansteuerung empfehlen wir die Verwendung der [pxt-SBC-MotoDriver3](#) Bibliothek, die von [Joy-IT](#) unter der [GNU-Lizenz](#) veröffentlicht wurde.

Die Bibliothek können Sie hinzufügen, indem Sie auf der [Makecode Seite](#) auf **Erweiterungen** klicken



und dort dann https://github.com/joy-it/pxt-SBC_MotoDriver3 in der Suchleiste eingeben. Nachdem Sie das getan haben, müssen Sie nur noch auf die Erweiterung klicken, um diese automatisch für Ihr aktuelles Projekt hinzuzufügen.

7.3. Codebeispiel

Zum Testen Ihres Motortreiberboards können Sie [hier](#) ein Codebeispiel von uns herunterladen oder wie unten gezeigt den Code einfach selber nachbauen.

Wenn Sie sich dazu entschieden haben unser Codebeispiel herunterzuladen, dann können Sie **nach dem Entpacken** der Datei den Inhalt einfach per Drag-and-Drop auf die Makecode Internetseite ziehen und loslassen, um die Datei zu importieren und automatisch zu öffnen.

Micro:bit Codebeispiel Abbildung:

```
beim Start
  Initialisieren des SBC_MotoDriver3 mit der Adresse 21
  Setzen Sie den enablepin auf Pin 8 und On
  Alle Kanäle ausschalten

dauerhaft
  seriell Zeile ausgeben "Normal usage"
  Schalten Sie alle Vorwärts Kanäle ein
  pausiere (ms) 1000
  Alle Kanäle ausschalten
  pausiere (ms) 1000
  Schalten Sie alle Rückwärts Kanäle ein
  pausiere (ms) 1000
  Alle Kanäle ausschalten
  pausiere (ms) 1000
  Kanal einschalten 0
  pausiere (ms) 2000
  Kanal ausschalten 0
  pausiere (ms) 500
  Kanal 0 ueber 20000 Millisekunden auf 250 einblenden
  Kanal 0 ueber 20000 Millisekunden auf 0 ausblenden
  pausiere (ms) 1000
  Kanal 0 auf PWM-Wert setzen 199
  seriell Zeile ausgeben Status des Kanals ablesen 0
  seriell Zeile ausgeben Lesen des Pwm-Status von Kanal 0
  pausiere (ms) 2000
  Alle Kanäle ausschalten
  pausiere (ms) 1000
```

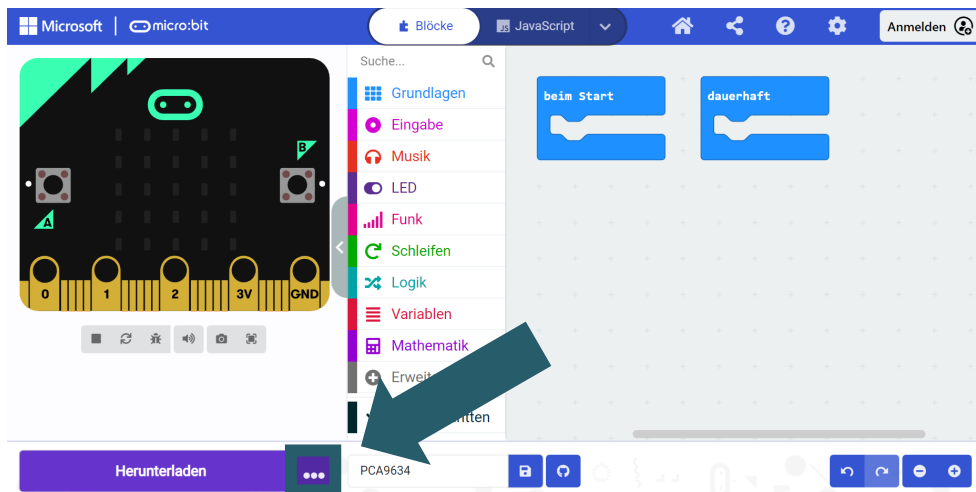
7.4 Verwendung von Stepper Motoren

Bitte beachten Sie, dass pro Seite immer nur ein Schrittmotor angeschlossen werden kann. In Bezug auf die Hardware ist es wichtig, die passenden Schrittmotoren für den zulässigen Spannungsbereich zu verwenden. Darüber hinaus sollten Sie sicherstellen, dass Sie, unabhängig von der Plattform, auf der Sie das Produkt verwenden möchten, niemals gleichzeitig einen Schrittmotor und einen Gleichstrommotor an dieselbe Seite anschließen. Dies kann zu Problemen oder sogar zu Beschädigungen des Produkts führen.

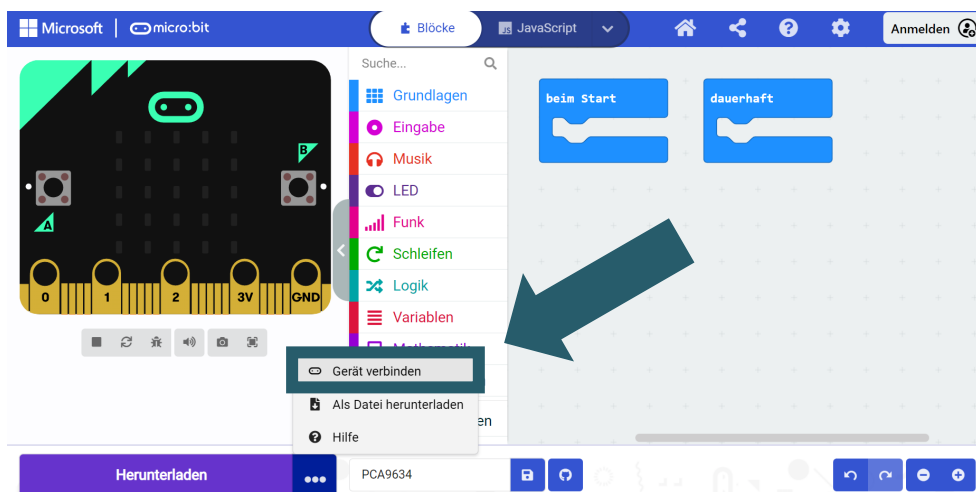
In Bezug auf die Software sollten Sie sicherstellen, dass Sie die Schrittmotorfunktion nicht mit den Funktionen für Gleichstrommotoren an den gleichen Motortreiber senden, unabhängig von der Plattform, auf der Sie das Produkt verwenden möchten. Halten Sie diese Funktionen stets getrennt voneinander.

7.5. Kopplung des Micro:Bit

Klicken Sie auf die Schaltfläche, welche sich rechts neben Hochladen/Herunterladen befindet.



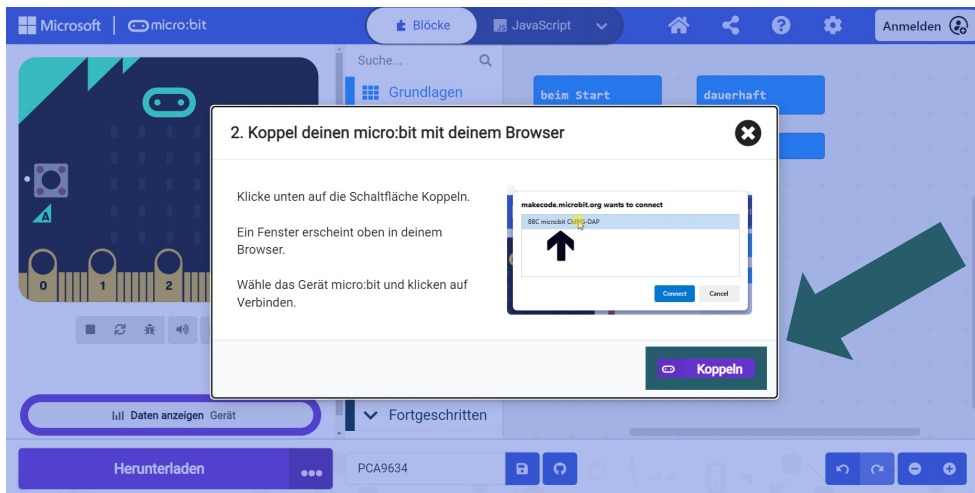
Klicken Sie auf **Gerät verbinden**.



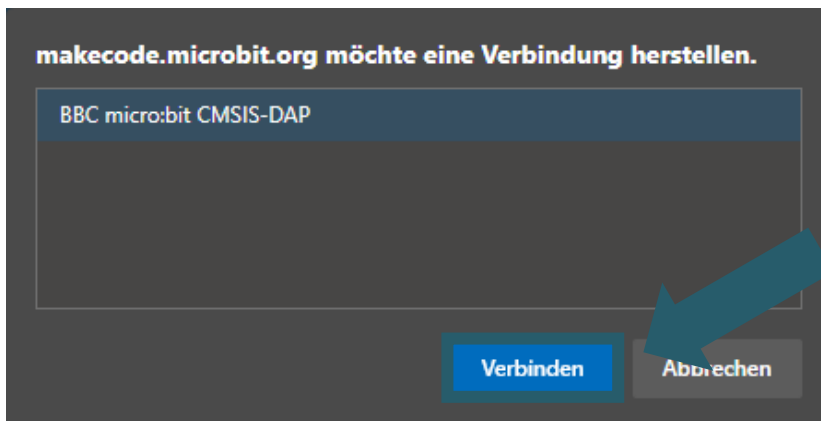
Klicken Sie auf **Weiter**.



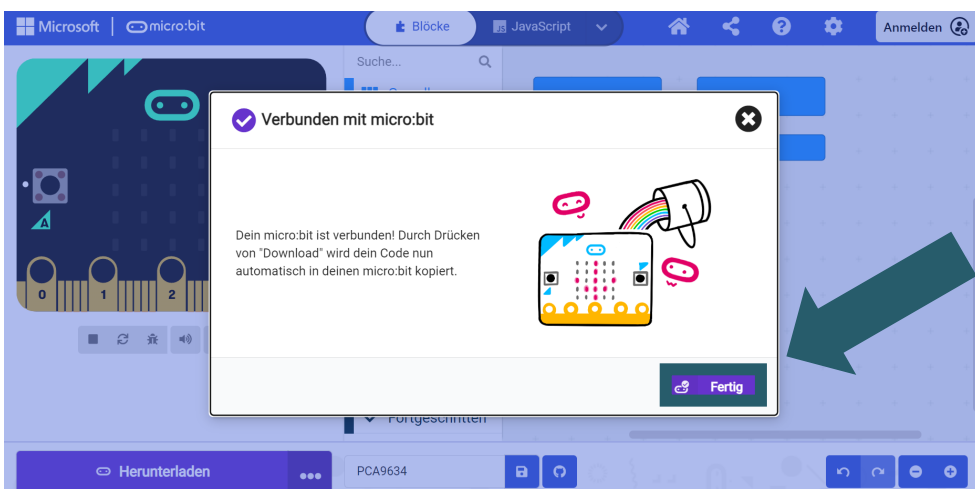
Klicken Sie auf **Koppeln**.



Wählen Sie Ihren Micro:Bit im geöffneten Fenster aus und klicken Sie auf Verbinden, um diesen zu verbinden und benutzen zu können.



Als Letztes müssen Sie nun noch auf **Fertig** klicken, um im Anschluss Ihren Code hochladen zu können. Sie laden ihren Code hoch, indem Sie auf Herunterladen klicken.



8. SONSTIGE INFORMATIONEN

Unsere Informations- und Rücknahmepflichten nach dem Elektroggesetz (ElektroG)



Symbol auf Elektro- und Elektronikgeräten:

Diese durchgestrichene Mülltonne bedeutet, dass Elektro- und Elektronikgeräte **nicht** in den Hausmüll gehören. Sie müssen die Altgeräte an einer Erfassungsstelle abgeben. Vor der Abgabe haben Sie Altbatterien und Altakkumulatoren, die nicht vom Altgerät umschlossen sind, von diesem zu trennen.

Rückgabemöglichkeiten:

Als Endnutzer können Sie beim Kauf eines neuen Gerätes, Ihr Altgerät (das im Wesentlichen die gleiche Funktion wie das bei uns erworbene neue erfüllt) kostenlos zur Entsorgung abgeben. Kleingeräte, bei denen keine äußere Abmessungen größer als 25 cm sind können unabhängig vom Kauf eines Neugerätes in haushaltsüblichen Mengen abgeben werden.

Möglichkeit Rückgabe an unserem Firmenstandort während der Öffnungszeiten:

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

Möglichkeit Rückgabe in Ihrer Nähe:

Wir senden Ihnen eine Paketmarke zu, mit der Sie das Gerät kostenlos an uns zurücksenden können. Hierzu wenden Sie sich bitte per E-Mail an Service@joy-it.net oder per Telefon an uns.

Informationen zur Verpackung:

Verpacken Sie Ihr Altgerät bitte transportsicher, sollten Sie kein geeignetes Verpackungsmaterial haben oder kein eigenes nutzen möchten kontaktieren Sie uns, wir lassen Ihnen dann eine geeignete Verpackung zukommen.

9. SUPPORT

Wir sind auch nach dem Kauf für Sie da. Sollten noch Fragen offen bleiben oder Probleme auftauchen, stehen wir Ihnen auch per E-Mail, Telefon und Ticket-Supportsystem zur Seite.

E-Mail: service@joy-it.net

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

Telefon: +49 (0)2845 9360 - 50 (Mo. - Do.: 10:00 - 17:00 Uhr,
Fr.: 10:00 - 14:30 Uhr)

Für weitere Informationen besuchen Sie unsere Website:

www.joy-it.net