

RB-MOT02

Raspberry Pi Stepper Motor-Steuerung

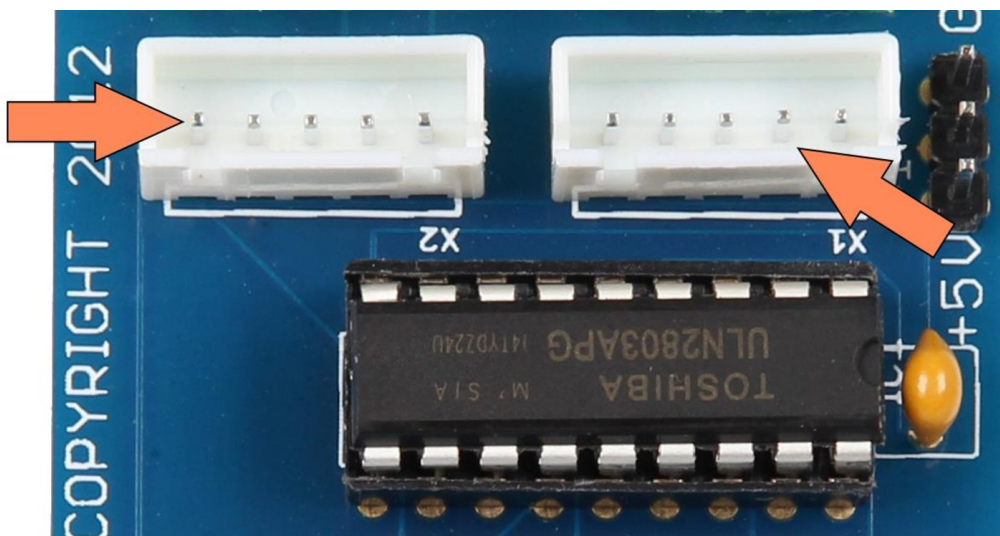
1. ALLGEMEINE INFORMATIONEN

Sehr geehrter Kunde,
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.

2. ZUSAMMENBAU UND EINRICHTUNG

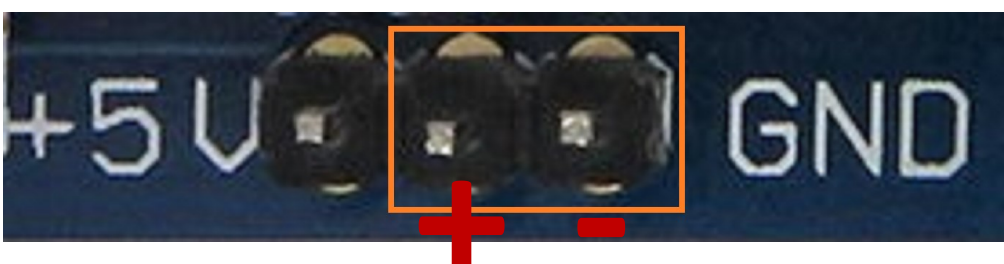
Schließen Sie die beiden Motoren an den entsprechenden Ausgangsbuchsen an.



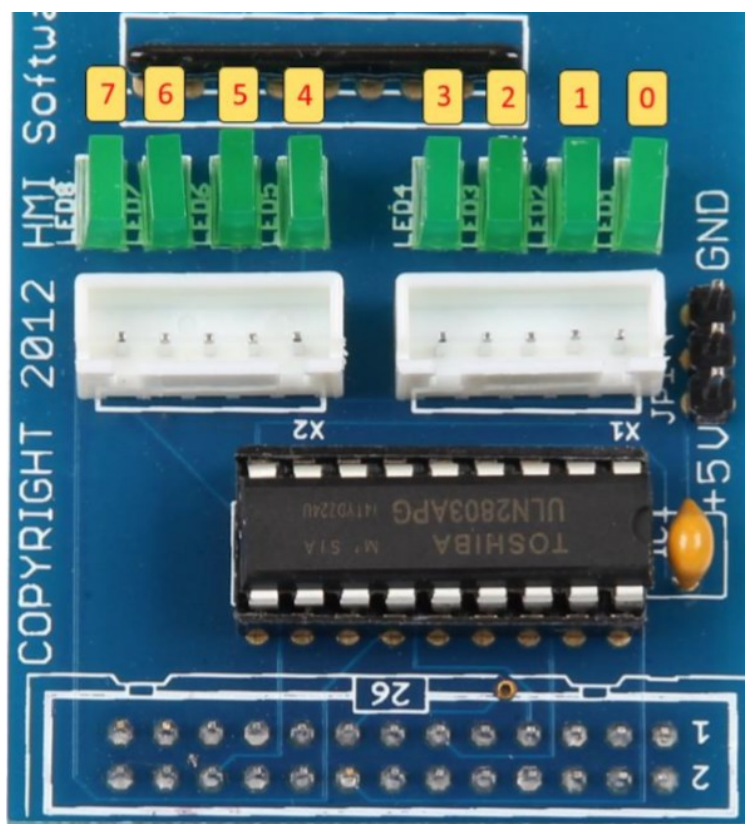
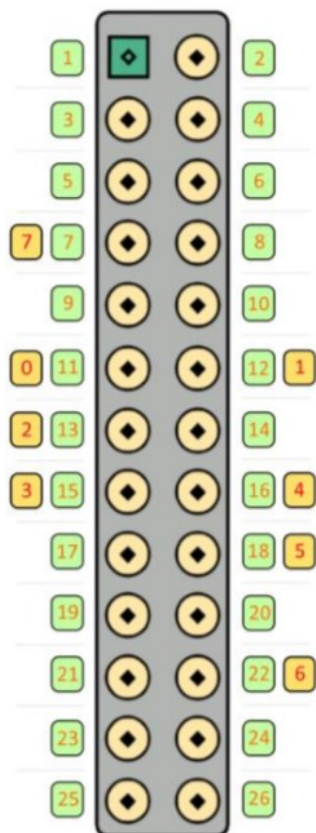
Mit der Stiftleiste können Sie Einfluss auf die Stromversorgung nehmen: Möchten Sie, dass die Motoren ihren Strom über den Raspberry Pi beziehen, dann verbinden Sie mit den beiliegenden "Jumper" die beiden linken Stifte, markiert auf folgendem Bild:



Möchten Sie jedoch eine externe Spannungsquelle einspeisen, dann können Sie diese an die beiden rechten Stifte, markiert auf folgendem Bild, installieren:



Schrittmotoren werden angesteuert, indem abwechselnd die einzelnen Schritte des Motors mit Strom versorgt werden. Die benötigten Pins für die Schritte der einzelnen Motoren entnehmen Sie der folgenden Anschlussübersicht:



3. CODEBEISPIEL

```
# -*- coding: utf-8 -*-
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)
coil_A_1_pin = 24 # pink
coil_A_2_pin = 4 # orange
coil_B_1_pin = 23 # blau
coil_B_2_pin = 25 # gelb
coil2_A_1_pin = 18 # pink
coil2_A_2_pin = 22 # orange
coil2_B_1_pin = 17 # blau
coil2_B_2_pin = 27 # gelb

StepCount = 8
Seq = list(range(0, StepCount))
Seq[0] = [0, 1, 0, 0]
Seq[1] = [0, 1, 0, 1]
Seq[2] = [0, 0, 0, 1]
Seq[3] = [1, 0, 0, 1]
Seq[4] = [1, 0, 0, 0]
Seq[5] = [1, 0, 1, 0]
Seq[6] = [0, 0, 1, 0]
Seq[7] = [0, 1, 1, 0]

GPIO.setup(coil_A_1_pin, GPIO.OUT)
GPIO.setup(coil_A_2_pin, GPIO.OUT)
GPIO.setup(coil_B_1_pin, GPIO.OUT)
GPIO.setup(coil_B_2_pin, GPIO.OUT)

GPIO.setup(coil2_A_1_pin, GPIO.OUT)
GPIO.setup(coil2_A_2_pin, GPIO.OUT)
GPIO.setup(coil2_B_1_pin, GPIO.OUT)
GPIO.setup(coil2_B_2_pin, GPIO.OUT)

def setStep(w1, w2, w3, w4):
    GPIO.output(coil_A_1_pin, w1)
    GPIO.output(coil2_A_1_pin, w1)
    GPIO.output(coil_A_2_pin, w2)
    GPIO.output(coil2_A_2_pin, w2)
    GPIO.output(coil_B_1_pin, w3)
    GPIO.output(coil2_B_1_pin, w3)
    GPIO.output(coil_B_2_pin, w4)
    GPIO.output(coil2_B_2_pin, w4)

def forward(delay, steps):
    for i in range(steps):
        for j in range(StepCount):
            setStep(Seq[j][0], Seq[j][1], Seq[j][2], Seq[j][3])
            time.sleep(delay)

def backwards(delay, steps):
    for i in range(steps):
        for j in reversed(range(StepCount)):
            setStep(Seq[j][0], Seq[j][1], Seq[j][2], Seq[j][3])
            time.sleep(delay)

# 20 langsame Schritte Vorwärts
delay = 20
steps = 20
forward(int(delay) / 1000.0, int(steps))
# 200 schnelle Schritte Rückwärts
delay = 1
steps = 200
backwards(int(delay) / 1000.0, int(steps))
```

4. CODEBEISPIEL P15

```
# -*- coding: utf-8 -*-
import gpiod
import time

coil_A_1_pin = 24 # pink
coil_A_2_pin = 4 # orange
coil_B_1_pin = 23 # blau
coil_B_2_pin = 25 # gelb
coil2_A_1_pin = 18 # pink
coil2_A_2_pin = 22 # orange
coil2_B_1_pin = 17 # blau
coil2_B_2_pin = 27 # gelb

chip = gpiod.Chip('gpiochip4')
coil_A_1 = chip.get_line(coil_A_1_pin)
coil_A_2 = chip.get_line(coil_A_2_pin)
coil_B_1 = chip.get_line(coil_B_1_pin)
coil_B_2 = chip.get_line(coil_B_2_pin)

coil2_A_1 = chip.get_line(coil2_A_1_pin)
coil2_A_2 = chip.get_line(coil2_A_2_pin)
coil2_B_1 = chip.get_line(coil2_B_1_pin)
coil2_B_2 = chip.get_line(coil2_B_2_pin)

coil_A_1.request(consumer="setstep", type=gpiod.LINE_REQ_DIR_OUT)
coil_A_2.request(consumer="setstep", type=gpiod.LINE_REQ_DIR_OUT)
coil_B_1.request(consumer="setstep", type=gpiod.LINE_REQ_DIR_OUT)
coil_B_2.request(consumer="setstep", type=gpiod.LINE_REQ_DIR_OUT)

coil2_A_1.request(consumer="setstep", type=gpiod.LINE_REQ_DIR_OUT)
coil2_A_2.request(consumer="setstep", type=gpiod.LINE_REQ_DIR_OUT)
coil2_B_1.request(consumer="setstep", type=gpiod.LINE_REQ_DIR_OUT)
coil2_B_2.request(consumer="setstep", type=gpiod.LINE_REQ_DIR_OUT)

StepCount = 8
Seq = list(range(0, StepCount))
Seq[0] = [0, 1, 0, 0]
Seq[1] = [0, 1, 0, 1]
Seq[2] = [0, 0, 0, 1]
Seq[3] = [1, 0, 0, 1]
Seq[4] = [1, 0, 0, 0]
Seq[5] = [1, 0, 1, 0]
Seq[6] = [0, 0, 1, 0]
Seq[7] = [0, 1, 1, 0]

def setStep(w1, w2, w3, w4):
    coil_A_1.set_value(w1)
    coil2_A_1.set_value(w1)
    coil_A_2.set_value(w2)
    coil2_A_2.set_value(w2)
    coil_B_1.set_value(w3)
    coil2_B_1.set_value(w3)
    coil_B_2.set_value(w4)
    coil2_B_2.set_value(w4)

def forward(delay, steps):
    for i in range(steps):
        for j in range(StepCount):
            setStep(Seq[j][0], Seq[j][1], Seq[j][2], Seq[j][3])
            time.sleep(delay)

def backwards(delay, steps):
    for i in range(steps):
        for j in reversed(range(StepCount)):
            setStep(Seq[j][0], Seq[j][1], Seq[j][2], Seq[j][3])
            time.sleep(delay)

# 20 langsame Schritte Vorwärts
delay = 20
steps = 100
forward(int(delay) / 1000.0, int(steps))
# 200 schnelle Schritte Rückwärts
delay = 1
steps = 200
backwards(int(delay) / 1000.0, int(steps))
```

5. 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.

6. 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.: 09:00 - 17:00 Uhr,
Fr.: 09:00 - 14:30 Uhr)

Für weitere Informationen besuchen Sie unsere Website:

www.joy-it.net