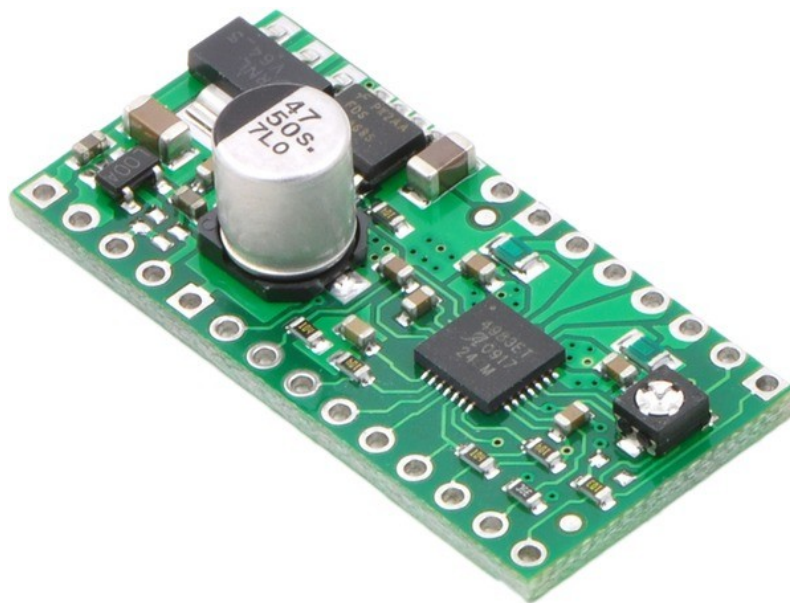


Anleitung - Stand vom 12.4.2011

Schrittmotortreiber A4983POW

mit 3,3V und 5V Ausgang



Der Schrittmotortreiber A4983POW eignet sich zum komfortablen und besonders einfachem Ansteuern von Schrittmotoren (bis 35V/2A) per Controllerboard. Lediglich 2 Leitungen zu einem Controllerboard (Richtung und Takt) genügen um einen Schrittmotor anzusteuern. Neben den üblichen Voll- und Halbschritt, beherrscht dieses Modul auch Mikroschritte (1/4 Schritte, 1/8 Schritte und 1/16 Schritte). Ein Standardschrittmotor mit 200 Schritten pro Umdrehung, kann dadurch also eine Genauigkeit von 3200 Schritten pro Umdrehung erreichen. Dadurch lassen sich präzise Aufgaben bei ruhigerem Lauf erzielen.

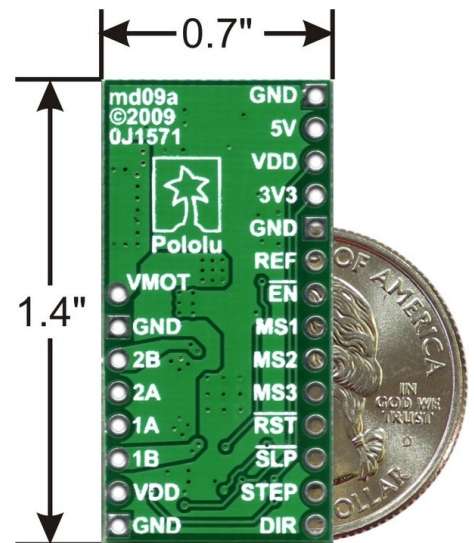
Der maximale Strom für den Schrittmotor wird über ein Poti (Regler) auf dem Modul fest eingestellt. Dadurch können nahezu alle bipolaren Schrittmotoren bis zu einer Stromstärke von 2A ansteuern. Die Nennspannung der Motoren ist in diesem Fall nahezu unerheblich, da sich diese durch die Stromregelung automatisch ergibt. Es ist nur wichtig, dass die Nennspannung der Schrittmotoren unter der Motorspannung liegt, je deutlicher das der Fall ist, desto mehr Kraftreserven können im höheren Drehzahlbereich erreicht werden. Nähere Infos zu Schrittmotoren findet man auf der Seite:
<http://www.rn-wissen.de/index.php/Schrittmotoren>

Dieser Schrittmotortreiber eignet sich somit für alle Schrittmotoren die im

Shop von Robotikhardware.de angeboten werden!

Technische Daten:

- Motorspannung 8 bis 35 Volt
- Phasenstrom/Motorstrom 2A (bei geeigneter Kühlung); ca. 1A ohne Kühlkörper
- maximaler Motorstrom über Poti einfach einstellbar
- Intelligente Stromregelung (chopping control)
- Einfaches Schritt und Richtung Interface, zwei Leitungen genügen
- Keine externe Logikspannung notwendig
- Chip Allegro's A4983 DMOS Microstepping Driver (Datenblatt auf DVD)
- Vollschritt, Halbschritt, Mikroschritte (1/4 Schritte, 1/8 Schritte und 1/16 Schritte)
- automatische Übertemperatur-Abschaltung, Unterspannungs-Lockout und Crossover-Schutz
- 5V Ausgang - z.B. zum versorgen eines Mikrocontrollers
- 3,3V Ausgang - z.B. zum versorgen eines Mikrocontrollers
- Maße 35x17mm



Wichtiger Hinweis

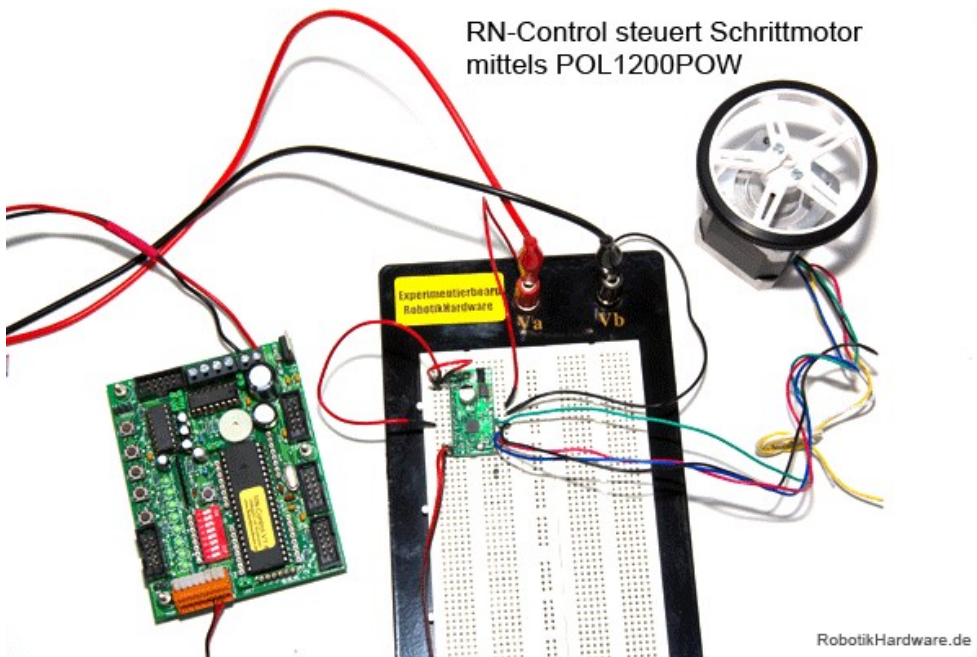
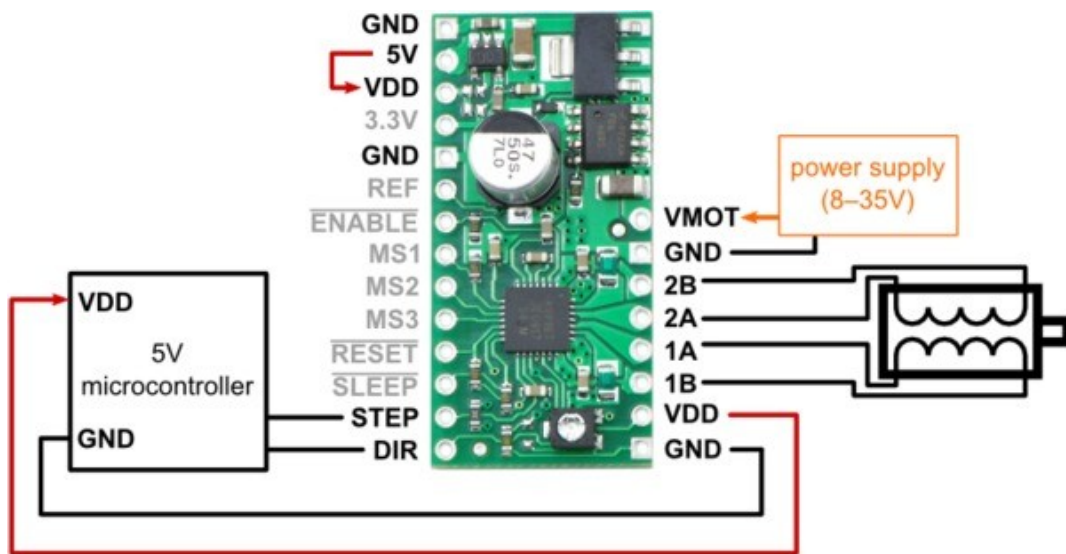
Achtung, während das Board mit Spannung versorgt wird darf keine Leitung zum Motor unterbrochen werden - dies kann das Modul ansonsten zerstören.

Bei höheren Stromstärken kann das Modul sehr heiß werden bevor der eingebaute Übertemperaturschutz anspricht. Achten Sie drauf das keine hitzeempfindlichen oder brennbaren Teile in der Nähe des Moduls sind, diese könnten sich entzünden. Lassen sie das Board nicht unbeaufsichtigt.

Ansteuerung

Die Verdrahtung und Ansteuerung dieses Schrittmotormoduls ist besonders einfach. Neben dem Motor (4 Drähte) muss lediglich die Motorspannung (zwischen 8 und 35V) angelegt werden. Für Logikspannung reicht eine einfache Brücke zwischen 5V und VDD (oder zwischen 3,3V und VDD falls man ein 3,3V Mikrocontroller zur Ansteuerung nutzt).

Ansonsten müssen nur der STEP und DIR Eingang mit zwei Ausgangsports eines Microcontrollers verbunden werden. Im unteren Bild sieht man wie die Verschaltung bei dem beliebten Board RN-Control erfolgen könnte.



Pinnbelegung

GND

Minuspol/Masse

5V

Stabilisierte Ausgangsspannung von 5V für eigene Verwendung

3,3V

Stabilisierte Ausgangsspannung von 3,3V für eigene Verwendung

VDD

Logikspannungseingang (3,3V oder 5V). Kann einfach mit 5V oder 3,3V Pin gebrückt werden.

REF

Hier kann die Referenzspannung gemessen werden, welche per Poti eingestellt werden kann. Die Referenzspannung bestimmt den maximalen Motorstrom. Hier gilt die Formel: $I_{\text{tripMax}} = V_{\text{Ref}} / (8 \times R_s)$. Da R_s immer 0,05 Ohm ist, also: $I_{\text{tripMax}} = V_{\text{Ref}} / (0,4)$. Beispiel: Gemessene 0,4 Volt ergäbe also einen maximalen Strom von 1 A. Gemessene 0,1 Volt ergäbe 250 mA usw.

Nähere Informationen zur Berechnung finden Sie im Datenblatt zum Schrittmotor IC (auf der Robotikhardware DVD).

Tipp: Steuern sie Schrittmotoren nicht mit zu hohen Strom an, wenn dies nicht wirklich notwendig ist. Oft reicht die halbe Stromstärke für viele Aufgaben aus, das spart Energie, reduziert die Hitzeentwicklung und schont den Motor!

ENABLE

Dieser Logikeingang kann auf Low oder High gesetzt werden. Dadurch kann der Motor ein oder ausgeschaltet werden, also die Motorspannung abgeschaltet werden.

MS1 / MS2 / MS3

Diese drei Logikeingänge bestimmen die Schrittweite. Wenn diese unbeschaltet bleiben, dann arbeitet das Board im Vollschritt-Betrieb.. Das bedeutet bei einem herkömmlichen Schrittmotor braucht dieser dann 200 Schritte (Impulse) bis er eine Umdrehung erreicht hat. Für viele Aufgaben sind aber kleinere Schritte günstiger weil er der Motor so präziser gesteuert werden kann und auch von den Vibrationen und Laufgeräuschen ruhiger arbeitet. Unsere Empfehlung wäre 1/4 Schritt Modus. Dazu muss lediglich MS2 mit High (+5V/3,3V) verbunden werden, danach sind 800 Schritte für eine Umdrehung notwendig.

MS1	MS2	MS3	Schrittweite
Low	Low	Low	Vollschritt
High	Low	Low	Halber Schritt
Low	High	Low	1/4 Schritt
High	High	Low	1/8 Schritt
High	High	High	1/16 Schritt

RESET

Bringt den Schrittmotor in einen definierten Ausgangszustand. Nähere Infos dazu im Datenblatt des Chips (auf Robotikhardware DVD).

SLEEP

Dieser Logikeingang kann auf Low oder High gesetzt werden. Dadurch läßt sich das Modul in einen Stromsparmmodus versetzen wenn der Motor gerade nicht gebraucht wird. Nähere Infos dazu im Datenblatt des Chips (auf Robotikhardware DVD).

STEP

Dieser Logikeingang kann auf Low oder High gesetzt werden. Jeder Impuls (steigende Flanke) bewegt den Motor genau um einen Schritt. Es ist sinnvoll diesen Eingang mit einem Controllerboard zu steuern. Ein Controllerboard kann Schritimpulse beispielsweise mit einem Timer erzeugen. Zählt das Controllerboard die Impulse mit, so weiß dieses genau wie weit sich die Welle gedreht hat. Dadurch sind ganz präzise Steuerungen möglich.

DIR

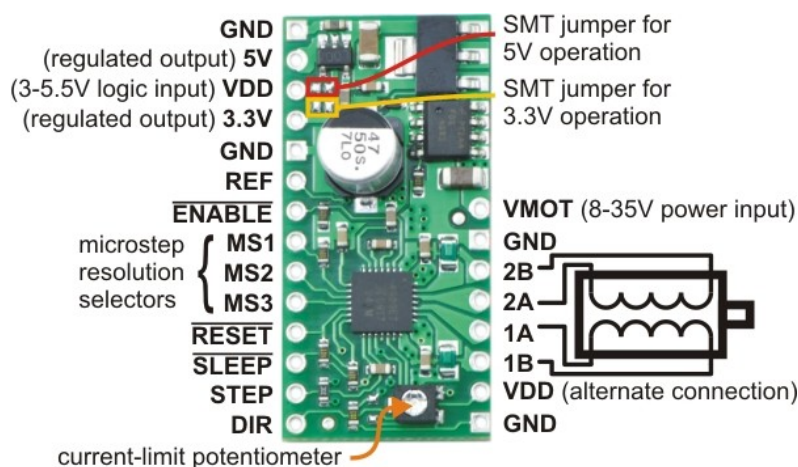
Dieser Logikeingang kann auf Low oder High gesetzt werden. Dadurch wird die Drehrichtung des Schrittmotors bestimmt. Es ist sinnvoll diesen Eingang mit einem Controllerboard zu steuern.

VMOT

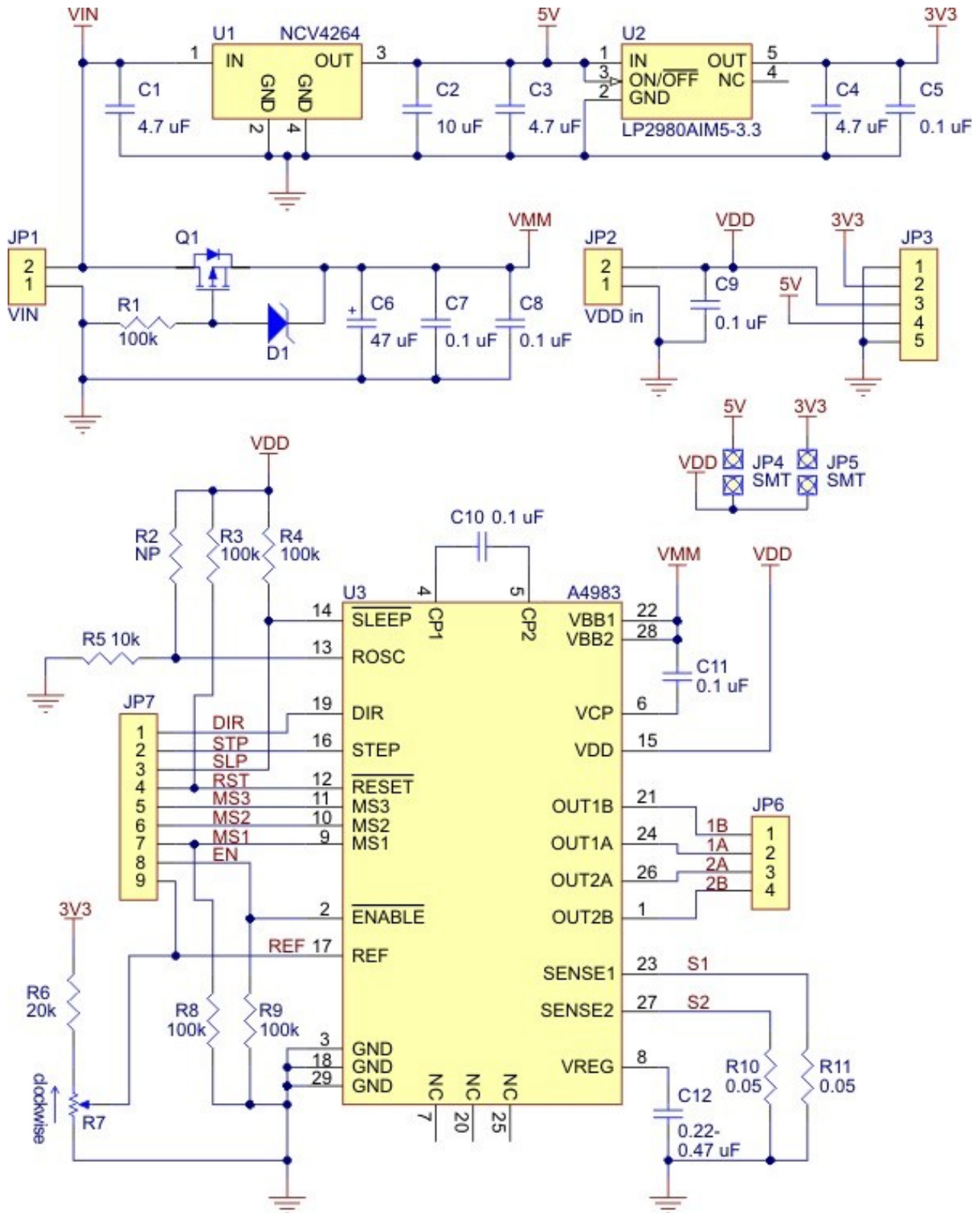
An diesen Pin wird die Motorspannung angelegt. Die Motorspannung kann zwischen 8 und 35 Volt liegen. Sie sollte deutlich höher sein als die Nennspannung des angeschlossenen Schrittmotors, mindestens 2V höher. je höher die Spannung, desto mehr Kraftreserven besitzt der Motor im höheren Drehzahlbereich.

2B / 2A / 1A / 1B

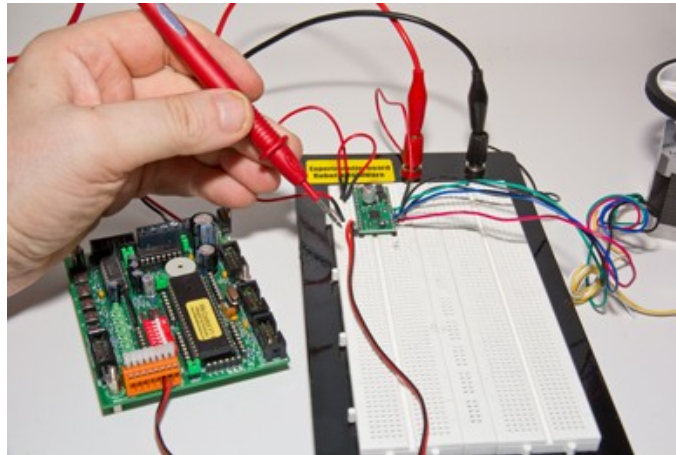
An diese vier Pins werden die beiden Wicklungen eines bipolaren Schrittmotors angeschlossen. Bei Schrittmotoren mit 6 Kabeln ist wird gewöhnlich eine Mittelanzapfung einer Wicklung nach außen geführt, diese wird nicht benötigt und kann/muss weggelassen werden



Schaltplan



Ein Bascom Beispielprogramm für RN-Control



```
'#####  
'Schrittmotoransteuerung mit RN-Control und  
'Ansteuerungsboard A4983POW  
'A4983POW ist ein Schrittmotortreiber  
'der Microschritte unterstützt  
'Nähere Infos bei robotikhardware.de  
'A4983POW_beispiel.bas  
'  
'Achtung:  
'Diese Demo ist mit Bascom Compiler 1.12 getestet  
'  
'A4983POW wird wie folgt angeschlossen  
'Enable auf PA0 / Richtung auf PA1 / Takt auf PA2  
'Enable muss nicht unbedingt angeschlossen werden!  
'  
'Aufgabe:  
'Dieses Testprogramm zeigt wie man RN-Stepp297  
'über Takt- und Richtungsleitung mit hoher  
'Taktfrequenz ansteuert. Nicht alle Schrittmotoren  
'machen diesen Geschwindigkeit mit!  
  
'Den verschiedenen Tasten sind bestimmte Funktionen zugeordnet  
'Taste 1: Schrittmotor genau mit 500 Schritten pro Sekunde drehen  
'Taste 2: Schrittmotor genau mit 1000 Schritten pro Sekunde drehen  
'Taste 3: Schrittmotor genau mit 1500 Schritten pro Sekunde drehen  
'Taste 4: Schrittmotor genau mit 2000 Schritten pro Sekunde drehen  
'Taste 5: Schrittmotor genau mit 3000 Schritten pro Sekunde drehen  
'  
'Autor: Frank  
'Weitere Beispiele und Beschreibung der Hardware  
'in der Anleitung zu A4983POW  
'Anleitung findet man unter http://www.robotikhardware.de  
'Weitere Beispiele sind im Roboternetz.de gerne willkommen!  
'#####  
  
$programmer = 12          'MCS USB (Zeile weglassen wenn anderer Programmer)  
  
'----- RN-Control übliche -----  
Declare Function Tastenabfrage() As Byte
```

```
$regfile = "m32def.dat"
$framesize = 32
$swstack = 32
$hwstack = 64

$crystal = 16000000           'Quarzfrequenz
$baud = 9600

Config Adc = Single , Prescaler = Auto   'Für Tastenabfrage und Spannungsmessung
Config Pina.7 = Input                    'Für Tastenabfrage
Porta.7 = 1                              'Pullup Widerstand ein
Dim Taste As Byte
Dim Ton As Integer

Start Adc
Sound Portd.7 , 400 , 450               'BEEP
Sound Portd.7 , 400 , 250               'BEEP
Sound Portd.7 , 400 , 450               'BEEP
Print
Print "***** RN-CONTROL V1.4 *****"
Print "Demoprogramm um Zusatzboard A4983POW zu testen/demonstrieren"
Print
' ----- Ende RN-Control übliche -----

Config Pina.0 = Output
Stepper_enable Alias Porta.0

Config Pina.1 = Output
Stepper_richtung Alias Porta.1

Config Pina.2 = Output
Stepper_takt Alias Porta.2

Dim I As Integer
Dim Timervorgabe As Byte

Stepper_enable = 0           ' Erst mal aus
On Timer0 Timer_irq
Enable Timer0

Stepper_richtung = 0

Do
  Taste = Tastenabfrage()
  If Taste <> 0 Then

    Select Case Taste

      Case 1           'Schrittmotor genau mit 500 Schritten pro Sekunde drehen
        Stepper_enable = 1
        Config Timer0 = Timer , Prescale = 256
        Timervorgabe = 131
        Enable Interrupts

      Case 2           'Schrittmotor genau mit 1000 Schritten pro Sekunde drehen
        Stepper_enable = 1
        Config Timer0 = Timer , Prescale = 64
        Timervorgabe = 6
        Enable Interrupts

      Case 3           'Schrittmotor genau mit 1500 Schritten pro Sekunde drehen
        Stepper_enable = 1
        Config Timer0 = Timer , Prescale = 64
        Timervorgabe = 89
        Enable Interrupts
```



```
Case 4          'Schrittmotor genau mit 2000 Schritten pro Sekunde drehen
  Stepper_enable = 1
  Config_Timer0 = Timer , Prescale = 64
  Timervorgabe = 131
  Enable Interrupts

Case 5          ''Schrittmotor genau mit 3000 Schritten pro Sekunde drehen
  Stepper_enable = 1
  Config_Timer0 = Timer , Prescale = 64
  Timervorgabe = 173
  Enable Interrupts

End Select
Sound Portd.7 , 400 , 500          'BEEP
End If

Waitms 100
Loop
End

Function Tastenabfrage() As Byte
Local Ws As Word

  Tastenabfrage = 0
  Ton = 600
  Ws = Getadc(7)
  Print "ws= " ; Ws
  If Ws < 1010 Then
    Select Case Ws
      Case 400 To 455
        Tastenabfrage = 1
        Ton = 550
      Case 335 To 380
        Tastenabfrage = 2
        Ton = 500
      Case 250 To 305
        Tastenabfrage = 3
        Ton = 450
      Case 180 To 220
        Tastenabfrage = 4
        Ton = 400
      Case 100 To 130
        Tastenabfrage = 5
        Ton = 350
    End Select
    Sound Portd.7 , 400 , Ton          'BEEP
  End If
End Function

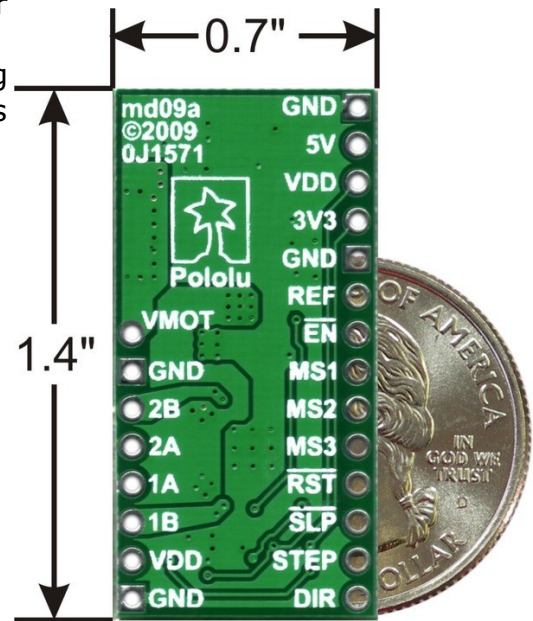
'Dies ist der Programmteil, der in dem von ihnen gewählten
'Intervall aufgerufen wird
Timer_irq:
  Timer0 = Timervorgabe
  Stepper_takt = 0 : Waitus 1 : Stepper_takt = 1
  Return
```

Englische Originalanleitung vom Hersteller

Overview

This product is a carrier board or breakout board for Allegro's A4983 DMOS Microstepping Driver with Translator; we therefore recommend careful reading of the A4983 datasheet (368k pdf) before using this product. This stepper motor driver lets you control one bipolar stepper motor at up to 2 A output current per coil (see the Power Dissipation Considerations section below for more information). Here are some of the driver's key features:

- Simple step and direction control interface
- Five different step resolutions: full-step, half-step, quarter-step, eighth-step, and sixteenth-step
- Adjustable current control lets you set the maximum current output with a potentiometer, which lets you use voltages above your stepper motor's rated voltage to achieve higher step rates
- Intelligent chopping control that automatically selects the correct current decay mode (fast decay or slow decay)
- Over-temperature thermal shutdown, under-voltage lockout, and crossover-current protection

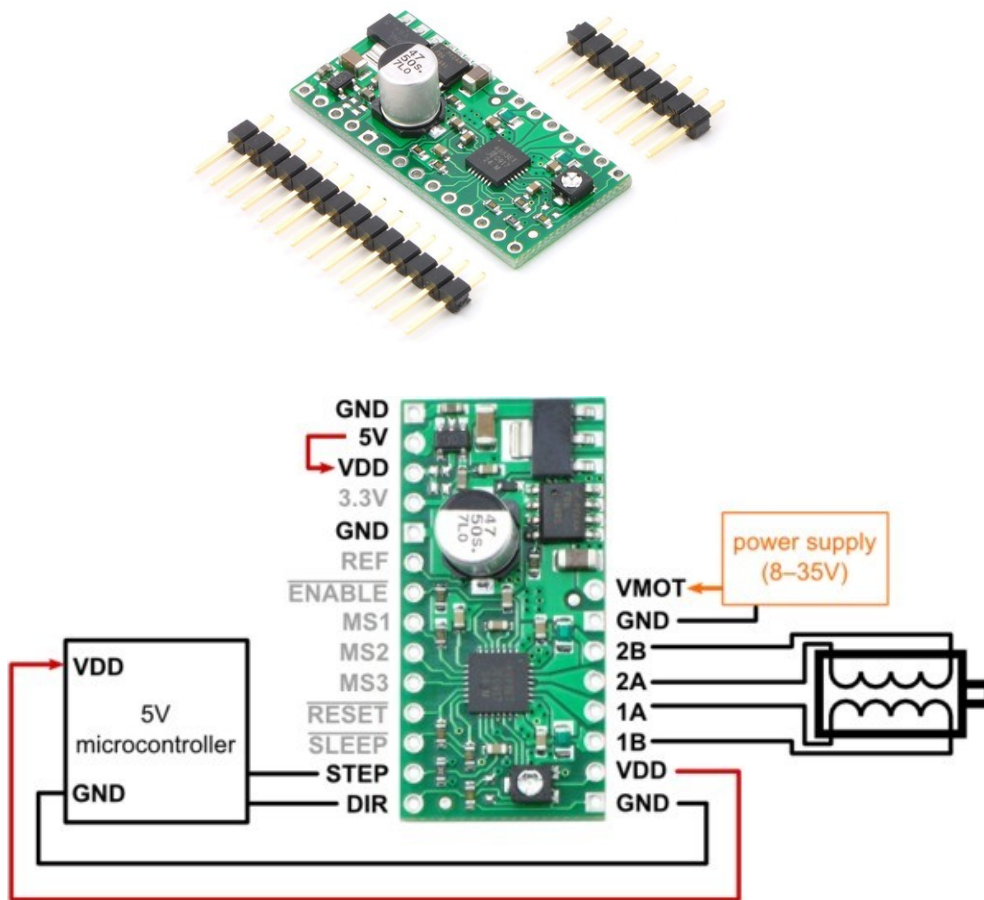


This carrier has reverse power protection on the main power input and built-in 5 V and 3.3 V voltage regulators that eliminate the need for separate logic and motor supplies and let you control the driver with microcontrollers powered at 5 V or 3.3 V. We also sell a smaller version of the A4983 carrier without voltage regulators.

Like nearly all our other carrier boards, this product ships with all surface-mount components—including the A4983 driver IC—installed as shown in the product picture.

Included Hardware

The A4983 stepper motor driver carrier with voltage regulators comes with one 1×14-pin breakaway 0.1" male header and one 1×8-pin breakaway 0.1" male header. The headers can be soldered in for use with solderless breadboards or 0.1" female connectors. You can also solder your motor leads and other connections directly to the board.



Power connections

The driver requires a logic supply voltage (3 – 5.5 V) to be connected across the VDD and GND pins and a motor supply voltage of (8 – 35 V) to be connected across VMOT and GND. The logic voltage can be supplied by jumpering the output of the 5 V or 3 V voltage regulator outputs to VDD. There are also surface-mount pads that allow VDD selection to be made by making a solder bridge across the appropriate pads. The power supply should be capable of delivering the expected currents (peaks up to 4 A for the motor supply).

Motor connections

Four, six, and eight-wire stepper motors can be driven by the A4983 if they are properly connected; a FAQ answer explains the proper wirings in detail.

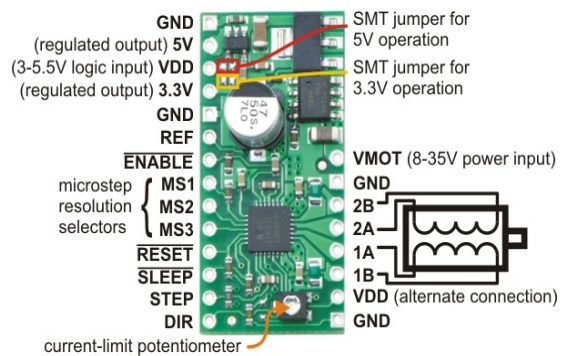
Warning: Connecting or disconnecting a stepper motor while the driver is powered can destroy the driver. (More generally, rewiring anything while it is powered is asking for trouble.)

Step (and microstep) size

Stepper motors typically have a step size specification (e.g. 1.8° or 200 steps per revolution), which applies to full steps. A microstepping driver such as the A4983 allows higher resolutions by allowing intermediate step locations, which are achieved by energizing the coils with intermediate current levels. For instance, driving a motor

in quarter-step mode will give the 200-step-per-revolution motor 800 microsteps per revolution by using four different current levels.

The resolution (step size) selector inputs (MS1, MS2, MS3) enable selection from the five step resolutions according to the table below. MS1, MS2 and MS3 all have internal 100kΩ pull-down resistors. For the microstep modes to function correctly, the current limit must be set low enough (see below) so that current limiting gets engaged. Otherwise, the intermediate current levels will not be correctly maintained, and the motor will effectively operate in a full-step mode.



MS1	MS2	MS3	Stepp
Low	Low	Low	Full
High	Low	Low	Half
Low	High	Low	1/4 Schritt
High	High	Low	1/8 Schritt
High	High	High	1/16 Schritt

Control inputs

Each pulse to the STEP input corresponds to one microstep of the stepper motor in the direction selected by the DIR pin. The chip has three different inputs for controlling its many power states: RST, SLP, and EN. For details about these power states, see the datasheet.

Current limiting

To achieve high step rates, the motor supply is typically much higher than would be permissible without active current limiting. For instance, a typical stepper motor might have a maximum current rating of 1 A with a 5Ω coil resistance, which would indicate a maximum motor supply of 5 V. Using such a motor with 12 V would allow higher step rates, but the current must actively be limited to under 1 A to prevent damage to the motor.

The A4983 supports such active current limiting, and the trimmer potentiometer on the board can be used to set the current limit. One way to set the current limit is to put the driver into full-step mode and to measure the coil current without clocking the STEP input. The measured current will be 0.7 times the current limit (since both coils are always on and limited to 70% in full-step mode). Please note that the current limit is dependent on the Vdd voltage.

Another way to set the current limit is to measure the voltage on the REF pin and to calculate the resulting current limit (the current sense resistors are 0.05Ω). See the A4983 datasheet for more information.

Power dissipation considerations

The A4983 driver IC has a maximum current rating of 2 A per coil, but the actual current you can deliver depends on how well you can keep the IC cool. The carrier's printed circuit board is designed to draw heat out of the IC, but to supply more than approximately 1 A per coil, a heat sink or other cooling method is required.

This product can get hot enough to burn you long before the chip overheats. Take care when handling this product and other components connected to it.

Please note that measuring the current draw at the power supply does not necessarily provide an accurate measure of the coil current. Since the input voltage to the driver can be significantly higher than the coil voltage, the measured current on the power supply can be quite a bit lower than the coil current (the driver and coil basically act like a switching step-down power supply). Also, if the supply voltage is very high compared to what the motor needs to achieve the set current, the duty cycle will be very low, which also leads to significant differences between average and RMS currents.

Anhang: Beispielsweise passende Motoren

Stepper Pol1206A 4V / 1,2A

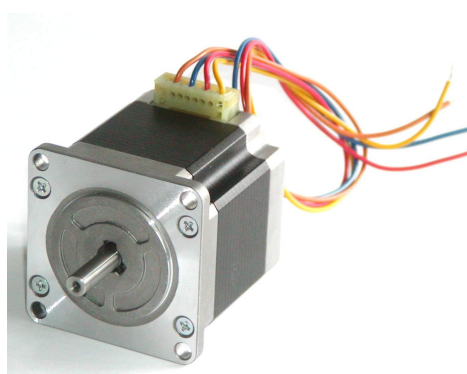


Stepper 0,7A 1,8 Grad



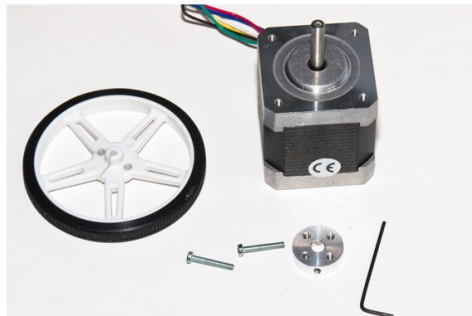
Stepper 1,6V/2,0A 1,8 Grad [Stepp5010]

(nur mit Kühlung des Moduls oder mit herabgesetzter Stromstärke)

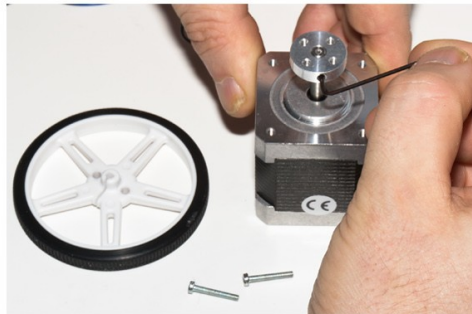


Anhang: Montage von Rädern

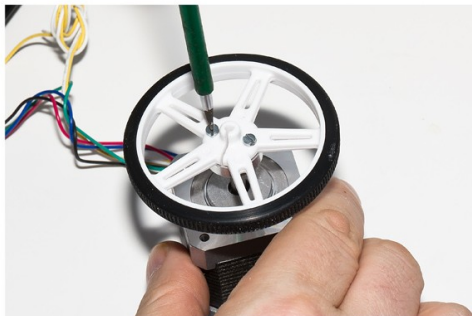
Radbefestigung mit Aluminum Befestigungsnahe



70mm Rad mit weicher Gummibereifung für beste Bodenhaftung



Montage auf 5mm Wellen mit Madenschraube



Rad wird mit 2.5mm Schrauben angeschraubt



Fertig! Natürlich nicht nur für Schrittmotoren geeignet sondern auch alle andere Motoren.

Rad/Nabe/Motor Bezug: robotikhardware.de

Bezugsquelle für dieses Modul: <http://robotikhardware.de/>

Modul-Hersteller: Pololu/USA

Aktuelle Infos/Ergänzungen:

Pololu/USA - <http://www.pololu.com/catalog/product/1202>

Aktuelle deutsche Doku: <http://robotikhardware.de/>

Übersetzung/ Deutsche Doku: <http://robotikhardware.de/>

Hilfreiches Technik-Forum: <http://www.roboternetz.de/community/>

Tipps zu Schrittmotoren: <http://www.rn-wissen.de/index.php/Schrittmotoren>

Hinweise zur beschränkten Haftung

Das Modul ist nicht für Geräte geeignet die direkt oder indirekt medizinischen, gesundheitlichen oder anderen Zwecken, bei denen ein Ausfall / Fehler zu Schäden an Personen oder Sachwerten führen würde. Soll das Modul für einen solchen Fall eingesetzt werden, so ist der Kunde für den Test und alle Zulassungen selbst verantwortlich. Robotikhardware.de übernimmt dafür keinerlei Haftung. Die Haftung beschränkt sich in allen Fällen auf den Austausch des fehlerhaften Moduls. Module die durch fehlerhaften Anschluß / Bedienung beschädigt wurden, können nicht ersetzt werden. Das Modul ist nicht für den kommerziellen Einsatz gedacht, sondern vornehmlich für Ausbildung, Entwicklung, Experimente und Modellbau. Eine andere Anwendung erfolgt auf eigenes Risiko. Die von uns angegebenen Daten sind weitestgehend Herstellerangaben für die wir keine Gewähr übernehmen können.

Haftung, EMV-Konformität

Alle Teile der Schaltung wurden sorgfältig geprüft und getestet. Trotzdem übernimmt der Entwickler/Vertrieber keine Garantie dafür daß alles einwandfrei funktioniert. Insbesondere wird keine Haftung für Schäden, die durch Nachbau, Inbetriebnahme etc. der hier vorgestellten Schaltungen entstehen. Derjenige, der den Bausatz zusammenbaut oder das Modul in eigene Schaltungen/Gehäuse integriert, gilt als Hersteller und ist damit selbst für die Einhaltung der geltenden Sicherheits- und EMV-Vorschriften verantwortlich.

Wenn nicht anders angegeben handelt es sich generell bei allen Bausätzen, Modulen und Boards um "nicht CE-geprüfte" Komponenten und sind konzipiert für den Einbau in Geräte oder Gehäuse. Bei der Anwendung müssen die CE-Normen eingehalten werden. Hierfür ist der Käufer verantwortlich.

Für Schäden die durch fehlerhaften Aufbau entstanden sind, direkt oder indirekt, ist die Haftung generell ausgeschlossen. Schadensersatzansprüche, gleich aus welchem Rechtsgrund, sind ausgeschlossen, soweit nicht vorsätzliches oder grob fahrlässiges Handeln vorliegt. Sofern wir haften, umfaßt unsere Haftung nicht solche Schäden, die nicht typischerweise erwartet werden konnten. Haftung und Schadenersatzansprüche sind auf den Auftragswert / Bauteilwert beschränkt. Bei der Lieferung von Fremdprodukten als auch Software gelten über diese Bedingungen hinaus die besonderen Lizenz- oder sonstigen Bedingungen des Herstellers.