

gräbner-elektronik gmbh

Am Römerbrunnen 11a • 61118 Bad Vilbel

Tel.: 06101/523100 • Fax: 06101/523101

eMail: info@graebner-elektronik.de • Internet <http://www.graebner-elektronik.eu>

DIPSLAM[®]

Handbuch V20.10

DIP-Serial-Link-AntriebsMotor

Hinweis: Technische Daten auf der letzten Seite

Immer wieder gibt es Probleme bei der Programmierung der seriellen Kommunikation von Seiten der Anwender. Auf unserem Modul ist ein bestimmtes Protokoll implementiert, das eingehalten werden muss. Lesen Sie den Abschnitt *Zur Kommunikation über die serielle Schnittstelle* unbedingt durch, bevor Sie mit dem Programmieren beginnen. Das wird Ihnen eine Menge Mühe ersparen !

Beschreibung

Die Baugruppe DIPSLAM[®] ist ein miniaturisierter Positions- und Drehzahlregler. Kleine DC-Motore bis ca. 5W können direkt angeschlossen werden (integrierte Endstufe). Zum Betrieb größerer Motore steht die unser Modul DIPSLAR[®] zur Verfügung.

DIPSLAM[®] ist in einen 28poligen Dual-Inline-Gehäuse untergebracht und dazu gedacht in kundenseitige Applikationen übernommen zu werden. Daher sind alle Ein- und Ausgänge TTL-kompatibel und somit an den kundenseitigen Controller leicht anzuschließen.

Das Modul verfügt über:

- Eine serielle Schnittstelle mit TTL-Pegeln (19200Bd, 8Bit, No Parity, 1 Stoppbit) über die die gesamte Kommunikation abgewickelt wird.
- Zwei Endschaltereingänge
- Zwei Eingänge für inkrementale Winkelenkoder (Kanal A und B)
- Ausgang PWM (Drehzahl)
- Ausgang Drehrichtung
- Ausgänge für DC-Motor (Power-Ausgänge – nicht TTL !)
- Spannungsversorgung 5V-Logik (max. 150mA)
- Spannungsversorgung DC-Motor (12..30V)

Die Baugruppe DIPSLAM[®] ist dazu geeignet, bürstenbehaftete DC-Motore im Leistungsbereich bis ca. 7W zu steuern. Die elektrische Ansteuerung des Motors erfolgt getaktet, genauer durch PWM-Ansteuerung der Leistungsendstufe. Um die Drehzahl des Motors zu regeln wird dazu das Verhältnis von Ein- zu Auszeit variiert. Die PWM-Grundfrequenz beträgt ca. 20kHz. Bei Betrieb mit DC-Motor ist eine Strombegrenzung wirksam.

Kernstück der Baugruppe ist ein sehr schneller Microcontroller. Die Signalverarbeitung und -erzeugung geschieht rein digital. Auch der implementierte PID-Regler ist ein rein digitaler Regler. Ein geeigneter Algorithmus bildet den aus der Analogtechnik bekannten PID-Regler nach. Es sind alle Winkelenkoder verwendbar die Ausgangssignale mit TTL-Pegeln (High=5V, Low=0V) liefern.

An die serielle Schnittstelle kann direkt ein geeigneter Pegelwandler (z. B. MAX232) angeschlossen werden um RS232-kompatible Pegel ($\pm 12V$) zu erhalten.

gräbner-elektronik gmbh

Am Römerbrunnen 11a • 61118 Bad Vilbel

Tel.: 06101/523100 • Fax: 06101/523101

eMail: info@graebner-elektronik.de • Internet <http://www.graebner-elektronik.eu>

Zu den Endschaltern:

Die Eingänge der Endschalter sind auf Schmitt-Trigger-Eingänge mit internen Pull-Up's geführt. Die Pegel müssen TTL-konform sein.

Die Endschalter müssen die Eingänge auf Masse ziehen, wenn sie nicht betätigt sind. Das schützt vor Kabelbruch. Die zwei Endschalter müssen den richtigen Enden der Fahrstrecke zugeordnet werden: Endschalter1 am Ende der Strecke mit negativen Positionen, Endschalter2 am Ende der Strecke mit positiven Positionen. Überprüfen Sie das unbedingt, der Motor fährt sonst gegen den mechanischen Endanschlag und nur die Strombegrenzung schützt ihn. Noch ein Hinweis: Wenn der Motor einen richtig angebrachten Endschalter betätigt, wird sofort die Verzögerungsrampe eingeleitet. Das hat zur Folge, daß der Motor mit der voreingestellten Beschleunigung verzögert bis er zum Stillstand kommt. Das kann bedeuten, dass der Motor an den mech. Anschlag oder gegen den Endschalter fährt wenn die Beschleunigung/Verzögerung sehr klein ist. Es gibt nur eine Lösung für das Problem: Es sollten Endschalter mit einer langen Nachlaufstrecke benutzt werden und die Beschleunigungs-/Verzögerungsrampe sollte kleiner als diese Nachlaufstrecke sein.

Zur seriellen Schnittstelle:

Die Übertragung erfolgt mit 19200Bd, 8Bit Daten, 1 Stoppbit, No Parity. Diese Einstellungen lassen sich nicht verändern.

Wenn nicht anders bestellt, wird die Baugruppe so ausgeliefert, dass, wenn die serielle Schnittstelle angeschlossen ist, sich die Baugruppe nach Anlegen der Versorgungsspannung (5V) mit ihrer Kennung meldet. Danach kann der DIPSLAM[®] konfiguriert und der Motor verfahren werden.

Status

Der Status des DIPSLAM[®] lässt sich mit "SS" abfragen. Als Antwort erhält der angeschlossene Rechner einen dezimalen Zahlenwert in binärer Kodierung. Der Zahlenbereich beträgt 0 bis 255 und setzt sich wie folgt zusammen:

| Wert | Bezeichnung | Bedeutung |
|------|-------------|---|
| 1 | LIMIT1 | Endschalter 1 betätigt |
| 2 | LIMIT2 | Endschalter 2 betätigt |
| 4 | VMODE | Drehzahlregelmodus |
| 8 | PMODE | Positioniermodus |
| 16 | MOVE | Motor dreht (noch) |
| 32 | BREAK | Bewegung durch den Befehl "BR" abgebrochen |
| 64 | ESAKTIV | Endschalter High=aktiv, Low=inaktiv |
| 128 | COMERROR | Vorhergehendes Kommando war unverständlich/falsch |

Im Einzelnen:

VMODE

Dieses Flag wird gesetzt nachdem der Befehl "VM" empfangen wurde.

Weiterhin: Siehe PMODE !

Das Flag wird zurückgesetzt nachdem der Befehl "ST" empfangen wurde.

PMODE

Dieses Flag wird nur gesetzt, wenn der Befehl "PM" empfangen wurde.

Zusätzlich wird das Flag VMODE gesetzt wenn ein Positionierbefehl empfangen wurde. VMODE wird zurückgesetzt wenn das Ziel erreicht ist.

Das Flag PMODE wird zurückgesetzt, wenn der Befehl "ST" empfangen wurde

LIMIT1

Dieses Flag ist nur dann gesetzt, wenn der Endschalter 1 betätigt ist.

gräbner-elektronik gmbh

Am Römerbrunnen 11a • 61118 Bad Vilbel

Tel.: 06101/523100 • Fax: 06101/523101

eMail: info@graebner-elektronik.de • Internet <http://www.graebner-elektronik.eu>

LIMIT2

Dieses Flag ist nur dann gesetzt, wenn der Endschalter 2 betätigt ist.

MOVE

Dieses Flag wird gesetzt wenn im Positioniermodus ("PM") ein Verfahrbefehl ("MA..") empfangen wurde und der Motor noch nicht an der Zielposition angekommen ist. Das Flag wird zurückgesetzt wenn der Motor an der Zielposition angekommen ist oder ein Stoppbefehl ("ST") empfangen wurde.

BREAK

Zeigt den Empfang eines "BR"-Befehls an. Das Flag wird zurückgesetzt wenn die Motorgeschwindigkeit 0 ist. Dann wird auch das MOVE-Flag zurückgesetzt.

ESAKTIV

Dieses Flag zeigt an, ob die Endschalterabfrage aktiviert ist. Ist das Flag High (1) werden die Endschalter berücksichtigt, ist das Flag Low (0), werden die Endschalter nicht berücksichtigt (z.B. für Rotationen).

COMERROR

Dieses Flag wird gesetzt, wenn der vor "SS" übermittelte Befehl ungültig war. Das Flag wird nach der Übermittlung des Statusbytes automatisch zurückgesetzt.

Zum Kalibrieren der Achse

Die DIPSLAM's verfügen über keinerlei Kalibrierbefehle. Das stellt aber kein wirkliches Problem dar. Zum Kalibrieren sollte einer der Endschalter dienen. Er muss also eine hinreichende Wiederholgenauigkeit haben. Vorgehensweise: Die Beschleunigung wird auf einen relativ hohen Wert gesetzt, die Geschwindigkeit auf einen mittleren Wert. Danach wird ein Befehl zum Verfahren des Motors gesendet, der den Positionierbereich der Strecke sicher übersteigt. Ziel ist hier, dass der Motor in den Endschalter fährt. Während diese Bewegung stattfindet wird das Statusbyte alle Sekunde abgefragt, ob der betreffende Endschalter betätigt wurde. Wenn ja, wird die Geschwindigkeit auf einen geringen Wert gesetzt und ein Positionierbefehl gegeben, der den Motor aus dem Endschalter fährt. Währenddessen wird wieder das Statusbyte abgefragt (ca. alle 200ms). Ist der betreffende Endschalter nicht mehr betätigt, wird ein Break-Befehl ("BR") ausgelöst. Wenn der Motor gestoppt hat (siehe Status-Byte – MOVE-Flag) wird nun mit der eingestellten geringen Geschwindigkeit und einer höheren Beschleunigung abermals in den Endschalter hinein gefahren. Das Statusbyte wird nun zyklisch abgefragt ob das Flag MOVE zurückgesetzt ist. Danach wird der Befehl "ZP" gesendet um diese Position zur Homeposition (Null-Position) zu erklären. Der Grund für diese Vorgehensweise: Je schneller der Motor dreht um so mehr Positionen können vorübergehen, bis der DIPSLAM[®] erkennt, dass ein Endschalter betätigt ist. Da diese Verzögerung nicht berechenbar ist, liegt die Lösung des Problems darin, mit so geringer Geschwindigkeit zu fahren, dass keine Verluste auftreten können. Die Endschalter werden 2500mal pro Sekunde abgefragt. Also kann die Geschwindigkeit berechnet werden mit der der Motor maximal drehen darf damit keine Positionsverluste auftreten. Die Beschleunigung ist dabei von nicht so großer Bedeutung, ein größerer Wert garantiert dabei lediglich einen kleineren Versatz zu der Position an der der Endschalter ausgelöst wurde - der bleibt aber konstant wenn immer die gleiche Verzögerung benutzt wird. Ist das nicht gewünscht: Beschleunigung gleich Geschwindigkeit setzen. Dann kann aber nicht ausgeschlossen werden, dass der Motor überschwingt. Da hilft es, eine zeitlang (vielleicht 1 Sekunde) untätig zu warten bis der Befehl "ZP" gesendet wird, damit der Motor Einschwingen kann.

PID-Parameter

gräbner-elektronik gmbh

Am Römerbrunnen 11a • 61118 Bad Vilbel

Tel.: 06101/523100 • Fax: 06101/523101

eMail: info@graebner-elektronik.de • Internet <http://www.graebner-elektronik.eu>

Die Parameter P, I und D haben den gleichen Wertebereich: 0..255. Nur positive Zahlen sind erlaubt. Die Grundeinstellung ist:

P=40, I=40, D=80

Das ist richtig für einen unbelasteten Motor Typ Faulhaber 2224 mit 512er Encoder.

Die Steifigkeit des Motors wird mit P verändert.

Die Schwingneigung wird mit D unterdrückt.

I verschafft Positioniergenauigkeit am Zielort.

Parameter vorsichtig ändern, allzu große Werteänderungen sind meist nicht sinnvoll, besser ist es, sich langsam an das "Optimum" heranzutasten. Folgende Vorgehensweise kann empfohlen werden:

I-Anteil abschalten bzw. stark verringern ("KI0" oder "KI1"). Nun ist die Positioniergenauigkeit sehr schlecht, aber der D- und P-Anteil wird nicht mehr vom I-Anteil beeinflusst. D-Anteil und P-Anteil erhöhen, bis die gewünschte oder machbare Steifigkeit erreicht ist. Um dem Schwingen vorzubeugen ggf. D-Anteil variieren. Ist man mit P und D zufrieden, wird der I-Anteil solange erhöht bis auch die Positioniergenauigkeit stimmt.

Um Veränderungen des Regelverhaltens feststellen zu können muss zwischen den einzelnen Schritten der Parameterermittlung der Motor immer wieder verfahren werden, nur dann bemerkt man die vorgenommenen Veränderungen.

Dieser Vorgang ist nicht sonderlich schwierig, ich empfehle aber, sich einen sehr ruhigen Raum zu suchen (damit man den Motor hört), sich 30 Minuten Zeit zu nehmen. Danach hat man ein Gefühl für den Motor und die Regelung und kennt die Grenzen von beiden. Weitere Nachbesserungen gehen dann leicht von der Hand. Vorteilhaft ist auch, den Motor in die Hand zu nehmen. Dabei spürt man geringste Schwingungen, denn, egal was man will, der Motor muss ruhig und gleichmäßig laufen, das ist oberste Maxime. Übrigens gibt es Applikationen bei denen es besser ist, die Steifigkeit des Systems zu verringern. Das wirkt sich vorteilhaft auf die Geschwindigkeitsregelung aus, wenn mehr Gleichmäßigkeit als Drehzahlgenauigkeit gefragt ist. Im Positioniermodus bringt das nur dann etwas, wenn sich das System ein wenig federnd verhalten soll. Man sollte jetzt aber nicht auf die Idee kommen mit den Motoren und der Regelung eine Feder nachzubilden. Der Regelung ist das egal, aber der Motor macht es nicht lange, wenn die Bürsten immer über ein und dieselbe Stelle kratzen.

Endschalter

Nach dem Einschalten der Spannungsversorgung ist die Endschalterabfrage aktiviert, d.h.: Findet eine Bewegung statt und löst die Bewegung einen der Endschalter aus, wird die Bewegung abgebrochen.

Bit 6 des Statusbyte zeigt an, ob die Endschalterabfrage aktiviert (High) oder deaktiviert (Low) ist. Mittels LI0 kann die Endschalterabfrage deaktiviert und mit LI1 wieder aktiviert werden.

gräbner-elektronik gmbh

Am Römerbrunnen 11a • 61118 Bad Vilbel

Tel.: 06101/523100 • Fax: 06101/523101

eMail: info@graebner-elektronik.de • Internet <http://www.graebner-elektronik.eu>

Befehlssatz DIPSLAM[®]

| Kommando | Zahlenbereich | Bemerkung |
|--|-------------------|---|
| Spezielle Befehle im Positioniermodus | | |
| PM | | Einschalten des Positioniermodus |
| SV | 0..32767 | Festlegen der max. Drehzahl |
| MA | -8388608..8388607 | Zur absoluten Position positionieren |
| SW | | Warte bis Bewegung beendet (Siehe Allgemeines) |
| Spezielle Befehle im Drehzahlregelmodus | | |
| VM | | Einschalten des Drehzahlregelmodus |
| SV | -32768..32767 | Festlegen der max. Drehzahl und Drehrichtung |
| Allgemeine Befehle | | |
| SA | 0..32767 | Festlegen der Beschleunigung |
| ST | | Stop, der eingeschaltete Modus wird beendet |
| BR | | Break, die Bewegung wird abgebrochen(mit Rampe) |
| ZP | | Der interne Positionszähler wird auf 0 gesetzt |
| SS | | Auslesen des Statusbytes |
| RP | | Auslesen der momentanen Position |
| PE | | Auslesen des momentanen Schleppfehlers |
| KP | 0..255 | Setzen des P-Anteils der Regelung |
| KI | 0..255 | Setzen des I-Anteils der Regelung |
| KD | 0..255 | Setzen des D-Anteils der Regelung |
| LI0 | 0..1 | LI0=Endschalter abschalten (Endschalter unberücksichtigt) LI1= Endschalter einschalten (Endschalter berücksichtigt) |
| IL | 0..3 | Logikpegel der Endschalter invertieren (0=nicht invertieren, 1=Limit1 invertieren, 2=Limit2 invertieren, 3=beide Endschalter invertieren) |
| SB | 1 Byte | Auslesen des Statusbytes in binärer Form (Siehe Allgemeines) |
| RB | 3 Byte | Auslesen der momentanen Position in binärer Form (Siehe Allgemeines) |
| RV | -32768..32767 | Auslesen der eingestellten Geschwindigkeit |
| RA | 0..32767 | Auslesen der eingestellten Beschleunigung |
| QP | 0..255 | Auslesen des P-Anteils |
| QI | 0..255 | Auslesen des I-Anteils |
| QD | 0..255 | Auslesen des D-Anteils |
| ID | String | Auslesen der Versionsbezeichnung |
| | | |

Hinweise:

Jeder Befehl muß mit einem RETURN (ASCII 13) abgeschlossen werden.

gräbner-elektronik gmbh

Am Römerbrunnen 11a • 61118 Bad Vilbel

Tel.: 06101/523100 • Fax: 06101/523101

eMail: info@graebner-elektronik.de • Internet <http://www.graebner-elektronik.eu>

Pro Zeile darf nur ein Befehl übermittelt werden.

Die Werte für SV und SA müssen im Verhältnis $1 < SV^2/SA < 4194303$ stehen.

Manche Befehle sind nicht immer ausführbar:

SA ist im VMODE immer ausführbar, im PMODE jedoch nur, wenn der Motor sich nicht bewegt bzw. die letzte Bewegung beendet hat (Flag MOVE=0). Eine Fehlermeldung erfolgt über das Flag COMERROR.

MA wird im VMODE nicht ausgeführt, es erfolgt keine Fehlermeldung über COMERROR.

Berechnung der Drehzahl/Beschleunigung

Berechnung der Geschwindigkeit (Drehzahl):

Drehzahl[UpM] = SV[Counts] * 140,417 / Linien

SV[Counts] = Drehzahl[UpM] * Linien / 140,417

Berechnung der Beschleunigung:

Beschleunigung[UpM²] = SA[Counts] * 140,417 / Linien

SA[Counts] = Beschleunigung[UpM²] * Linien / 140,417

Wobei

...SV[Counts] wird mit "sv..." übergeben.

...SA[Counts] wird mit "sa..." übergeben.

...Linien die Angabe der Auflösung des Winkelenkoders ist.

Beachten Sie:

Die Werte für SV und SA müssen im Verhältnis $1 < SV^2/SA < 4194303$ stehen.

Allgemeine Hinweise

Mit der vorliegenden Version ist es möglich auch während eines Positioniervorgangs die Verstellgeschwindigkeit zu verändern. Dazu ist einfach die neue gewünschte Drehzahl mittels "V..." zu übergeben.

Hintergründe: Bei Aufruf von SV werden die Beschleunigungs-/Verzögerungsrampen neu kalkuliert und – falls der "Restweg" reicht - übernommen.

Unter bestimmten Umständen kann die geforderte Aktion nicht ausgeführt werden. Diese Umstände sind

- 1) Die neue gewünschte Zielposition erfordert einen Drehrichtungswechsel.
- 2) Der Motor ist nahe des ursprünglichen Ziels und die Verzögerungsrampe ist bereits eingeleitet.
- 3) Ein Erreichen der neuen Verstellgeschwindigkeit oder neuen Zielposition ist wegen des geringen "Restwegs" nicht mehr möglich.

In diesen Fällen wird "SV..." abgebrochen und es erfolgt eine Fehlermeldung über das COMERROR-Flag. Der Motor wird dann mit der ursprünglichen Verstellgeschwindigkeit auf die ursprüngliche Zielposition verfahren.

Zu 1):

Soll ein Richtungswechsel erzwungen werden empfiehlt sich zunächst einen "BR"-Befehl zu senden. "BR" beendet die laufende Bewegung mit sofortigem Einleiten der Verzögerungsrampe. Ist Motorstillstand erreicht, wird das MOVE-Flag automatisch zurückgesetzt. Das ist dann auch der Indikator, dass der Motor steht und es kann eine neue Verstellgeschwindigkeit und neue Zielposition gesendet werden.

gräbner-elektronik gmbh

Am Römerbrunnen 11a • 61118 Bad Vilbel

Tel.: 06101/523100 • Fax: 06101/523101

eMail: info@graebner-elektronik.de • Internet <http://www.graebner-elektronik.eu>

Auslesen des Statusbytes in binärer Form

Um die Datenübertragung zu beschleunigen wird das Statusbyte als einzelnes (binäres) Byte übertragen. Der Wertebereich liegt zwischen 0 und 255, da stets 8Bit übertragen werden. Dieser Befehl ähnelt "SS". Dort wird das Statusbyte zunächst intern in eine dezimale Zahl gewandelt und dieser Wert als ASCII-String übertragen. Diese Zeichenkette variiert in ihrer Länge je nachdem ob nur ein oder 3 Zeichen gesendet werden müssen. Im steuernden Rechner ist dieser Wert dann im Normalfall wieder in sein binäres Äquivalent umzurechnen. Das dauert. Wird "SB" benutzt entfällt dieser Zeitbedarf (und die Umrechnung). Mit "SB" stellt sich also das Statusbyte als einzelnes Zeichen dar. Die Bedeutung der einzelnen Bits bleibt gegenüber "SS" unverändert:

Bit 0 = LIMIT1

Bit 1 = LIMIT2 usw.

Auch diese Antwort wird – wie üblich – mit CR/LF abgeschlossen.

Auslesen der momentanen Position in binärer Form

Die Gründe für die Implementierung dieses Befehls sind die gleichen wie beim "SB"-Befehl. Mit "RB" werden stets 3 Byte übertragen. Das erste übertragene Byte ist das MSB, das Dritte also das LSB.

Auch diese Antwort wird – wie üblich – mit CR/LF abgeschlossen.

Der Befehl SW

Jeder an das Modul gesendete Befehl liefert einen Antwortstring. Dieser String (in diesem Fall ein Leerstring) wird auch von SW geliefert – aber erst, wenn die augenblicklich stattfindende Bewegung beendet ist.

Bewegungsabläufe zu automatisieren ist eigentlich recht einfach: Die einzelnen Befehle werden einfach in eine Textdatei geschrieben. Diese Datei wird sodann von einem Programm zeilenweise gelesen. Wenn sichergestellt ist, dass nur ein Befehl pro Zeile in die Textdatei geschrieben wurde ist alles recht einfach:

Zunächst wird das erste Zeichen gesendet und auf das Echo des Zeichens gewartet, dann kommt das nächste Zeichen usw. bis zum Zeilenende (CR). Auch das CR wird gesendet und auf das Echo gewartet. Nun muss das Programm alle Zeichen einsammeln, die vom Modul kommen bis einschließlich dem letzten Zeichen (einem LF). Danach kann die nächste Zeile des "Textes" gelesen und, wie dargestellt, gesendet werden.

Der Befehl SW verzögert nun die Antwort solange, bis das MOVE-Flag des Status-Registers zurückgesetzt wird – erst dann wird die Kombination CR/LF gesendet.

Das kann u.U. jede Menge Programmieraufwand ersparen.

Zur Kommunikation über die serielle Schnittstelle

Der DIPSLAM[®] besitzt keinen internen Puffer. Werden die Zeichen vom steuernden Rechner zu schnell gesendet, können Zeichen verloren gehen. Daher ist ein Echomodus implementiert.

Folgende Vorgehensweise ist einzuhalten:

Der Befehl mit seinem Parameter wird in einen String gewandelt.

Nun wird das erste Zeichen des Strings an die serielle Schnittstelle übergeben.

Danach hat der Rechner auf das Echo dieses Zeichens zu warten das der DIPSLAM[®] generiert.

Ist das Echo empfangen kann das nächste Zeichen gesendet werden.

Das letzte Zeichen sollte ein CR (\$0D, dezimal 13) sein. LF sollte nicht gesendet werden. Das führt zwar nicht zu Fehlern, wird aber ebenfalls geecho und kann als Endkennung der Antwort des DIPSLAM[®] interpretiert werden.

Nun hat der Rechner alle Zeichen, die DIPSLAM[®] nun senden wird in einen String zu packen.

Das vorletzte Zeichen des DIPSLAM[®] ist ein CR (\$0D, dezimal 13).

gräbner-elektronik gmbh

Am Römerbrunnen 11a • 61118 Bad Vilbel

Tel.: 06101/523100 • Fax: 06101/523101

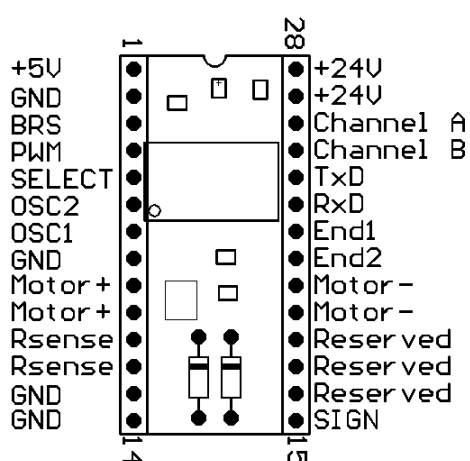
eMail: info@graebner-elektronik.de • Internet <http://www.graebner-elektronik.eu>

Das letzte Zeichen das DIPSLAM[®] sendet ist LF (\$0A, dezimal 10). Danach erfolgen keine weiteren Zeichen vom DIPSLAM[®] und der String kann ausgewertet werden. Was im String steht ist vom vorher gesendeten Befehl abhängig. "RP" liefert die momentane Position als ASCII-Zeichenkette, "MA" liefert "nichts" – also einen Leerstring.

Pinbelegung

Die Signale und ihre Bedeutung:

| Signal | Pin Nr. | Erläuterung |
|-----------|--------------|--|
| +5V | 1 | Spannungsversorgung 5V, max. 150mA |
| GND | 2, 8, 13, 14 | GND von +5V und +24V. Alle Pins sind auf Masse zu legen. |
| BRS | 3 | Reserviert |
| PWM | 4 | TTL-PWM-Ausgang für externe Leistungsstufen |
| SELECT | 5 | Reserviert |
| OSC2 | 6 | Darf nicht kontaktiert werden! |
| OSC1 | 7 | TTL-Clock-Eingang 50MHz |
| Motor+ | 9, 10 | Motoranschluß |
| Rsense | 11,12 | Dient zur Einstellung des Motorstroms |
| SIGN | 15 | TTL-Ausgang Drehrichtung für externe Leistungsstufen |
| Reserved | 16, 17, 18 | Darf nicht kontaktiert werden ! |
| Motor- | 19, 20 | Motoranschluß |
| End2 | 21 | TTL-Eingang für Endschalter 2 |
| End1 | 22 | TTL-Eingang für Endschalter 1 |
| RxD | 23 | TTL-Eingang serielle Schnittstelle |
| TxD | 24 | TTL-Ausgang serielle Schnittstelle |
| Channel B | 25 | TTL-Eingang Winkelenkoder Kanal B |
| Channel A | 26 | TTL-Eingang Winkelenkoder Kanal A |
| +24V | 27, 28 | Motorstromversorgung +12...30V= |



gräbner-elektronik gmbh

Am Römerbrunnen 11a • 61118 Bad Vilbel

Tel.: 06101/523100 • Fax: 06101/523101

eMail: info@graebner-elektronik.de • Internet <http://www.graebner-elektronik.eu>

Technische Daten DIPSLAM[®]

Datenübertragungsrate

19200Bd, 8Bit, 1 Stopbit, Keine Parität

Pegel auf den Datenübertragungsleitungen

Es werden TTL-Pegel erwartet

Spannungsversorgung Logik-Teil

5V \pm 5% Gleichspannung

Spannungsversorgung Motor

12..24V Gleichspannung

Stromaufnahme Logik-Teil

<100mA

Stromaufnahme Motor

Abhängig vom Motor, max. 400mA

Spannungsbereich Endschaltereingänge

TTL-Pegel

Der Endschalter gilt als betätigt, wenn der Eingang für mindestens 650 μ s auf logisch 1 (+5V) gelegt wird.

Strombegrenzung

Auslieferungszustand: ca. 270mA

Eingänge Winkelenkoder

TTL-Pegel, Innenwiderstand ca. 3000 Ohm

Es werden zwei, um 90° phasenverschobene Rechtecke erwartet (Quadraturdecoder)

Maximal zulässige Ausgangsfrequenz des Winkelenkoders

200kHz

Flankensteilheit der Enkodersignale

Kleiner 500ns