

QuickPos® EASYSTART



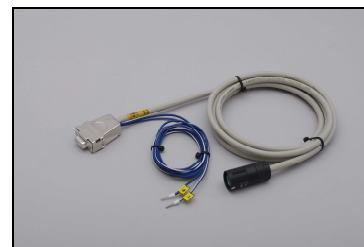
1. Relevante Produkte

Intelligente Miniatur Linearmotormodule QP01-2070-040-RS und QP01-2070-080-RS

2. Benötigtes Kabel

Zur Verbindung vom **QuickPos®** Modul mit der Spannungsversorgung und dem PC (RS232) wird ein Programmierkabel (unsere Artikel-Nr. 0291480) benötigt.

Anschluss am QuickPos® Winkeleinbaudose 12polig (m)	
Kontakt-Nr.	Signal
1	GND
2	+ 10 - 30 VDC
5	IN1
6	IN2
10	OUT1
11	TxD oder CAN_H
12	RxD oder CAN_L
Gehäuse	Schirm



QuickPos® Programmierkabel

3. Benötigte Software

Laden Sie bitte von www.faulhaber.com die aktuelle **Motion Manager** Software herunter und installieren Sie **SetupMoMan6_6_0.exe** oder eine höhere Version auf Ihrem PC!

Voraussetzung: PC mit MS Windows XP, W7 oder W10 und ein PC mit RS232 bzw. USB/RS232 Converter

4. Versorgungsspannung und Anschluss

Verbinden Sie das Programmierkabel mit dem **QuickPos®** Modul und schließen Sie ein 24 VDC Netzteil mit $\geq 2,5$ A Ausgangsstrom an.

Aderfarbe bl = + 24 VDC, Aderfarbe bl/ws = GND.

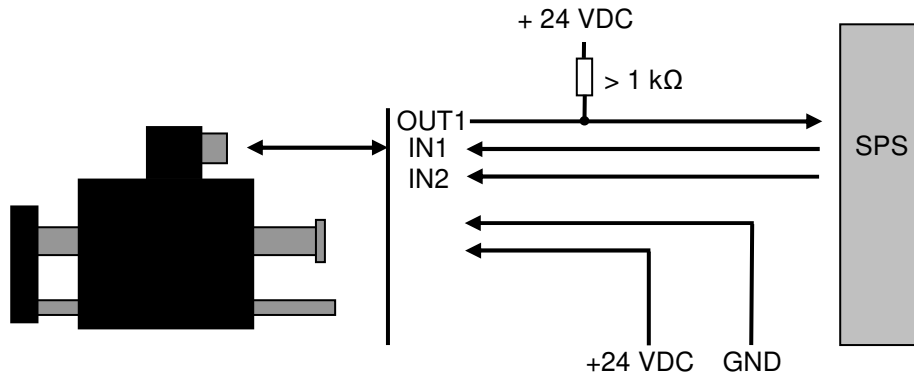
Auf richtige Polarität ist unbedingt zu achten!

5. Kommunikationsaufbau

Wir setzen in den **QuickPos®** Modulen den Faulhaber Motion Controller Typ **MCLM 3002 P RS** ein!

Die Bedienung der **Motion Manager** Software und der Kommunikationsaufbau über RS232 erfolgt gemäß Anleitung der Firma Faulhaber GmbH.

6. Beispielschaltplan für eine SPS-Anbindung



7. Parameter und Beispielprogramm

7.1 Parameter

QuickPos® Module werden von uns vorparametriert geliefert!

Der Parametersatz kann mit der **Motion Manager** Software ausgelesen und in eine Parameterdatei gespeichert werden!

7.2 Beispielprogramm '2 Eingänge - 4 Positionen'

QuickPos® Module werden von uns mit folgenden Beispielprogrammen vorprogrammiert ausgeliefert:

qp01_2070_040.mcl für das Modul QP01-2070-040

qp01_2070_080.mcl für das Modul QP01-2070-080

7.2.1 Beschreibung des Beispielprogrammes für QP01-2070-080

Nach Versorgungsspannung 'EIN'

- wird automatisch das Beispielprogramm gestartet
- werden die notwendigen Parameter vorbesetzt (Programmteil vor Label a3)
- geht der Antrieb in Lageregelung
- erfolgt eine Referenzfahrt auf mechanischen inneren Anschlag (Programmteil ab Label a3)
- erfolgt ein Verfahren zum Maß A gemäß Datenblatt (Programmteil bis Label a10)
- wird eine Programmschleife abgearbeitet in der die 2 Eingänge (ab Label a10) abgefragt werden
- wird abhängig von den Eingangszuständen der 2 Eingänge eine Positionierfahrt ausgeführt

Zuordnung der Eingangszustände zu den Positionen		
Eingangszustand	Aktion	Programmteil ab Label
IN1 = 0 u. IN2 = 0	Fahrt zur Position 1	a30
IN1 = 1 u. IN2 = 0	Fahrt zur Position 2	a40
IN1 = 0 u. IN2 = 1	Fahrt zur Position 3	a50
IN1 = 1 u. IN2 = 1	Fahrt zur Position 4	a60

Der Programmteil ab Label a10 fragt zyklisch die Eingänge IN1 und IN2 ab und startet entsprechend die 4 Positionierfahrten.

Label a10 fragt IN1 = H ab.

Label a30 fragt ab, ob IN1 = L. Falls nicht, verweist das Programm auf Label a40.

Andernfalls positioniert es den Antrieb zur Position 1 (0 Inkremente = 0 mm).

Label a20 fragt IN1 = H ab. Falls nicht, verweist es auf Label a30.

Label a40 fragt ab, ob IN2 = H ist. Ist dies der Fall, verweist es auf Label a60.

Ist dies nicht der Fall, positioniert es den Antrieb zur Position 2 (2.500 Inkremente = 20 mm).

Label a50 positioniert den Antrieb zur Position 3 (5.000 Inkremente = 40 mm).

Label a60 positioniert den Antrieb zur Position 4 (10.000 Inkremente = 80 mm).

Wichtige Hinweise:

Die Auslösung der Positionierfahrten durch die 2 Eingänge ist pegelgetriggert. D.h. solange der betreffende Eingangszustand ansteht, wird die entsprechende Position angefahren bzw. gehalten.
Die Eingänge IN1 und IN2 sind mit ca. 10 - 20 ms durch die Software entprellt.
Solange eine Positionierfahrt läuft, werden keine Eingänge abgefragt.
Die Zielpositionen sind in Inkrementen, mit der Einheit 1 Inkrement entsprechend 8 µm, anzugeben.
Die Geschwindigkeit ist in mm/s anzugeben (max. Wert 10.000).
Die Beschleunigung ist in mm/s² anzugeben (max. Wert 30.000).
Ausgang OUT1 ist ein NPN open collector Transistor (≤ 30 mA) mit Bezugspotential 0 V. Der Ausgang ist so programmiert, dass der Transistor durchgeschaltet ist, wenn der Antrieb referiert hat bzw. wenn der Antrieb die jeweilige Zielposition erreicht hat.

8. Beispielprogramm für QP01-2070-040

```
lpc 2370
lcc 790
ll 14500
ll -250
en
apl 1
ne 1
eri 1
digout
so
ac 0
dec 0
sp 0
ho 0
ci 40
a 3
so
seta 1000
ho
apl 0
ac 500
dec 500
sp 100
lr -14250
m
a 4
grc
jmpga 5
jmp 4
a 5
ho -1250
la 0
np
m
apl 1
co
a 10
delay 1
jph 20
delay 1
a 30
jph 40
jpt 50
so
ac 20000
dec 20000
sp 2000
la 0
np
m
co
jmp 10
```

a 20
jph 40
jmp 30
delay 1
a 40
jpt 60
so
ac 20000
dec 20000
sp 2000
la 1250
np
m
co
jmp 10
a 50
so
ac 20000
dec 20000
sp 2000
la 2500
np
m
co
jmp 10
a 60
so
ac 20000
dec 20000
sp 2000
la 5000
np
m
co
jmp 10

9. Beispielprogramm für QP01-2070-080

```
lpc 2370
lcc 790
ll 14500
ll -250
en
apl 1
ne 1
eri 1
digout
so
ac 0
dec 0
sp 0
ho 0
ci 40
a 3
so
seta 1000
ho
apl 0
ac 500
dec 500
sp 100
lr -14250
m
a 4
grc
jmpga 5
jmp 4
a 5
ho -1750
la 0
np
m
apl 1
co
a 10
delay 1
jph 20
delay 1
a 30
jph 40
jpt 50
so
ac 20000
dec 20000
sp 2000
la 0
np
m
co
jmp 10
```

a 20
jph 40
jmp 30
delay 1
a 40
jpt 60
so
ac 20000
dec 20000
sp 2000
la 2500
np
m
co
jmp 10
a 50
so
ac 20000
dec 20000
sp 2000
la 5000
np
m
co
jmp 10
a 60
so
ac 20000
dec 20000
sp 2000
la 10000
np
m
co
jmp 10