

WAGO I/O SYSTEM 750

Nutzung der Stepper_02.lib für die Module

750-670 und 750-671

Anwendungshinweis

A114900, Deutsch
Version 1.0.3

Copyright © 2006 by WAGO Kontakttechnik GmbH & Co. KG
Alle Rechte vorbehalten.

WAGO Kontakttechnik GmbH & Co. KG

Hansastraße 27
D-32423 Minden

Tel.: +49 (0) 571/8 87 – 0
Fax: +49 (0) 571/8 87 – 1 69
E-Mail: info@wago.com
Web: <http://www.wago.com>

Technischer Support

Tel.: +49 (0) 571/8 87 – 5 55
Fax: +49 (0) 571/8 87 – 85 55
E-Mail: support@wago.com

Es wurden alle erdenklichen Maßnahmen getroffen, um die Richtigkeit und Vollständigkeit der vorliegenden Dokumentation zu gewährleisten. Da sich Fehler, trotz aller Sorgfalt, nie vollständig vermeiden lassen, sind wir für Hinweise und Anregungen jederzeit dankbar.

Wir weisen darauf hin, dass die im Dokument verwendeten Soft- und Hardwarebezeichnungen und Markennamen der jeweiligen Firmen im Allgemeinen einem Warenzeichenschutz, Markenschutz oder patentrechtlichem Schutz unterliegen.

INHALTSVERZEICHNIS

1 Wichtige Erläuterungen	4
1.1 Rechtliche Grundlagen	4
1.1.1 Urheberschutz	4
1.1.2 Personalqualifikation	4
1.1.3 Bestimmungsgemäßer Gebrauch	4
1.2 Gültigkeitsbereich	5
1.3 Symbole	5
2 Allgemeines	6
2.1 Materialien	6
3 Verdrahtung	6
3.1 Modul 750-670 ohne Leistungsendstufe	6
3.2 Modul 750-671 mit Leistungsendstufe (24V / 1,0A)	7
4 Allgemeine Regeln für die Nutzung der Bibliotheksfunktionen	7
5 Konfiguration	8
5.1 Referenzieren	8
5.2 Rampen	9
5.2.1 Konstante Beschleunigung	9
5.2.2 Lineare Beschleunigung	10
5.2.3 \sin^2 Beschleunigung	11
5.3 Änderung der Bit Zuordnung	12
5.3.1 Beispiel 1	12
5.3.2 Beispiel 2	12
5.3.3 Nennstrom Modul 750-671 einstellen	13
5.3.3.1 Beispiel	13
5.3.4 Berechnung der Anzahl Steps pro Umdrehung für Modul 750-671 ..	13
5.3.5 Berechnung Geschwindigkeit und Beschleunigung	14
6 Positionierung	15
7 Fahrprogramm	16
8 Nocken Funktionalität	17
9 Beispiel	18
10 Visualisierungen aus der Stepper_02.lib	20

1 Wichtige Erläuterungen

Um dem Anwender eine schnelle Installation und Inbetriebnahme der beschriebenen Geräte zu gewährleisten, ist es notwendig, die nachfolgenden Hinweise und Erläuterungen sorgfältig zu lesen und zu beachten.

1.1 Rechtliche Grundlagen

1.1.1 Urheberschutz

Dieses Dokument, einschließlich aller darin befindlichen Abbildungen, ist urheberrechtlich geschützt. Jede Weiterverwendung dieses Dokumentes, die von den urheberrechtlichen Bestimmungen abweicht, ist nicht gestattet. Die Reproduktion, Übersetzung in andere Sprachen, sowie die elektronische und fototechnische Archivierung und Veränderung bedarf der schriftlichen Genehmigung der WAGO Kontakttechnik GmbH & Co. KG, Minden. Zuwiderhandlungen ziehen einen Schadenersatzanspruch nach sich.

Die WAGO Kontakttechnik GmbH & Co. KG behält sich Änderungen, die dem technischen Fortschritt dienen, vor.
Alle Rechte für den Fall der Patenterteilung oder des Gebrauchsmusterschutzes sind der WAGO Kontakttechnik GmbH & Co. KG vorbehalten. Fremdprodukte werden stets ohne Vermerk auf Patentrechte genannt. Die Existenz solcher Rechte ist daher nicht auszuschließen.

1.1.2 Personalqualifikation

Der in diesem Dokument beschriebene Produktgebrauch richtet sich ausschließlich an Fachkräfte mit einer Ausbildung in der SPS-Programmierung, Elektrofachkräfte oder von Elektrofachkräften unterwiesene Personen, die außerdem mit den geltenden Normen vertraut sind. Für Fehlhandlungen und Schäden, die an WAGO-Produkten und Fremdprodukten durch Missachtung der Informationen dieses Dokumentes entstehen, übernimmt die WAGO Kontakttechnik GmbH & Co. KG keine Haftung.

1.1.3 Bestimmungsgemäßer Gebrauch

Die Komponenten werden ab Werk für den jeweiligen Anwendungsfall mit einer festen Hard- und Softwarekonfiguration ausgeliefert. Änderungen sind nur im Rahmen der in dem Dokument aufgezeigten Möglichkeiten zulässig. Alle anderen Veränderungen an der Hard- oder Software, sowie der nicht bestimmungsgemäße Gebrauch der Komponenten, bewirken den Haftungsausschluss der WAGO Kontakttechnik GmbH & Co. KG.

Wünsche an eine abgewandelte bzw. neue Hard- oder Softwarekonfiguration richten Sie bitte an WAGO Kontakttechnik GmbH & Co. KG.

1.2 Gültigkeitsbereich

Dieser Anwendungshinweis basiert auf die genannte Hard- und Software der jeweiligen Hersteller sowie auf die zugehörige Dokumentation. Daher gilt dieser Anwendungshinweis nur für die beschriebene Installation.

Neue Hard- und Softwareversionen erfordern eventuell eine geänderte Handhabung.

Beachten Sie die ausführliche Beschreibung in den jeweiligen Handbüchern.

1.3 Symbole



Gefahr

Informationen unbedingt beachten, um Personen vor Schaden zu bewahren.



Achtung

Informationen unbedingt beachten, um am Gerät Schäden zu verhindern.



Beachten

Randbedingungen, die für einen fehlerfreien Betrieb unbedingt zu beachten sind.



ESD (Electrostatic Discharge)

Warnung vor Gefährdung der Komponenten durch elektrostatische Entladung. Vorsichtsmaßnahme bei Handhabung elektrostatisch entladungsgefährdeter Bauelemente beachten.



Hinweis

Routinen oder Ratschläge für den effizienten Geräteeinsatz und die Softwareoptimierung.



Weitere Informationen

Verweise auf zusätzliche Literatur, Handbücher, Datenblätter und INTERNET Seiten.

2.1 Materialien

Assigned hardware components	Producer	Type
Ethernet-Controller	WAGO	750-841
Stepper Module (Indexer)	WAGO	750-670 (SW02HW02)
Stepper Module	WAGO	750-671 (SW02HW02)
Stepper_02.lib	WAGO	Version 2.2.1
Stepper_ with integrated power amplifier	Sonceboz	
PowerMax II	Pacific Scientific	1.8°Stepper Motor

3 Verdrahtung

3.1 Modul 750-670 ohne Leistungsendstufe

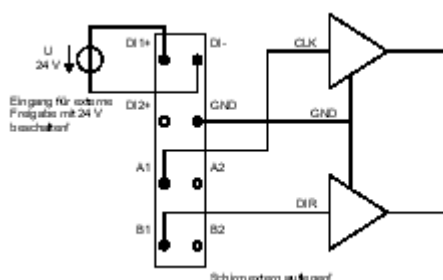
Eingang Ch1: Freigabe des Steppers(24V DC)

Eingang Ch2: Referenz Schalter(24V DC)

Ch-: Masse für Ch1 und Ch2

Ausgang A1: Frequenz(5V DC)

Ausgang B1: Richtung(5V DC)



3.2 Modul 750-671 mit Leistungsendstufe (24V / 1,0A)

Dieses Modul unterstützt 2 polige bipolare Schrittmotoren.

Eingang Ch1+: Freigabe des Steppers(24V DC)

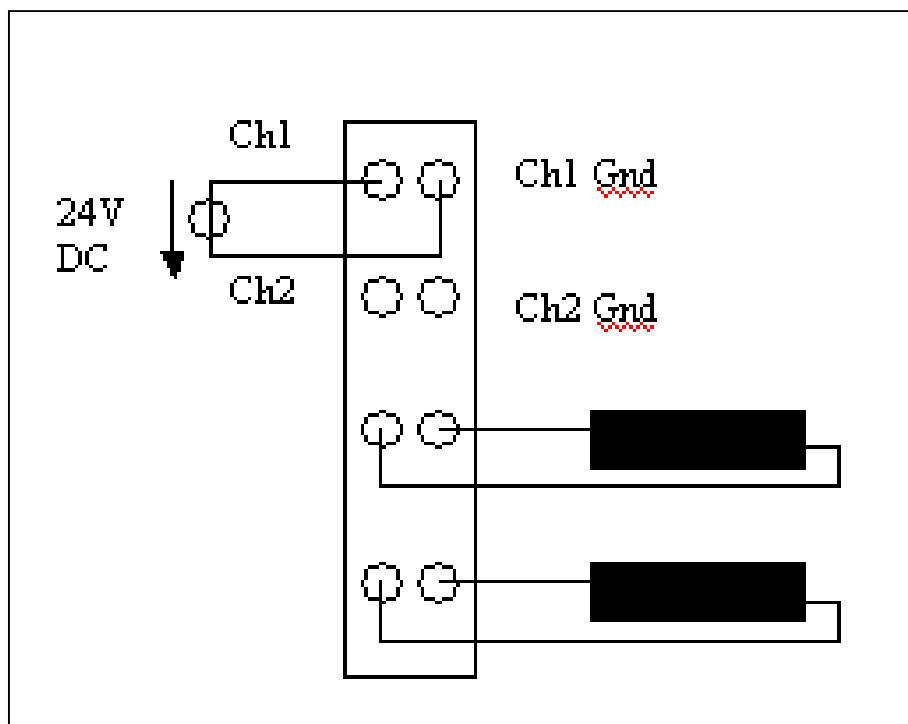
Ch1-: Masse Eingang 1

Eingang Ch2+: Referenz Schalter(24V DC)

Ch2-: Masse Eingang 2

Ausgang A1,A2: Spule A des Motors

Ausgang B1,B2: Spule B des Motors



4 Allgemeine Regeln für die Nutzung der Bibliotheksfunktionen

Die Bibliothek Stepper_02.lib stellt eine große Anzahl von Funktionsbausteinen zur Verfügung, mit denen die verschiedensten Anwendungen realisiert werden können. Diese Bausteine müssen durch das Anwendungsprogramm synchronisiert werden, da sie alle mit den gleichen Ein- und Ausgabedaten arbeiten.

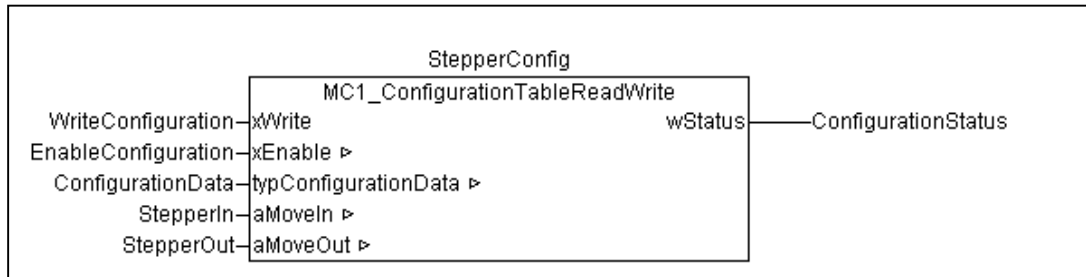


Zu jedem Zeitpunkt darf nur ein Funktionsbaustein aktiv sein.

Aufgrund der Vielzahl von Bausteinen in der Bibliothek Stepper_02 werden in diesem Anwendungshinweis die einzelnen Bausteine nicht im Detail erläutert. Dazu wird auf die Bibliotheksbeschreibung ML002600 verwiesen.

5 Konfiguration

Nutzung des Bausteins MC1_ConfigurationTableReadWrite



Mit Hilfe dieses Funktionsbausteins kann das Modul konfiguriert werden. Die Konfigurationswerte sind in einer Struktur MC1_typConfiguration zusammengefasst. Die Struktur ist in zwei Bereiche gegliedert. Die Parameter bis zur Adresse 124 können direkt beschrieben werden. Die Parameter von Adresse 128 bis 223 sind Zeiger. Diese Zeiger benötigen die Informationen des Bit I/O Driver Feldes, welches im Handbuch zu finden ist.



Es wird empfohlen diesen Baustein einmalig in der Initialisierungsphase zu benutzen, da nach dem Schreiben der Konfigurationswerte, die aktuelle Position auf Null zurückgesetzt wird.

Während der Bearbeitung dieses Bausteins darf die Variable xStop des Bausteins MC1_StepperBasicControl nicht gesetzt sein.

5.1 Referenzieren

Die Referenzfahrt wird durch die Parameter Reference_Offset, Reference_Mode, SetupAcceleration und SetupSpeed festgelegt. Für das Modul 750-671 sind die Default Werte SetupAcceleration und SetupSpeed aufgrund des 64-fachen Microsteppings anzupassen um eine sinnvolle Referenzfahrt zu ermöglichen.

Für den Parameter Reference_Mode gilt:

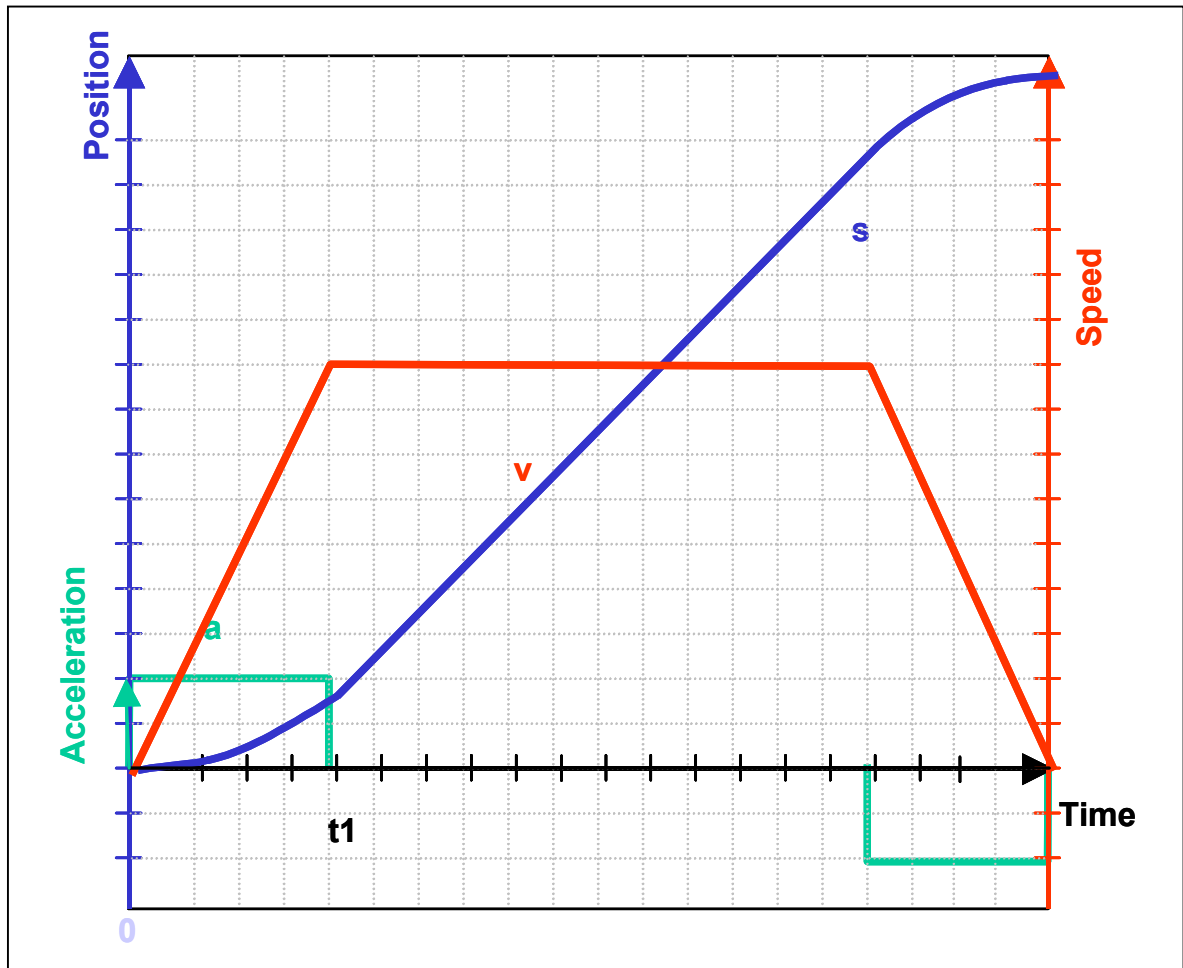
- 0:Nutzung eines Referenz Schalters, anfahren der negativen Seite des Schalters
- 1:Referenzieren auf Endschalter
- 2:Nutzung eines Referenz Schalters, anfahren der positiven Seite des Schalters
- 3:Referenzieren auf Endschalter

5.2 Rampen

5.2.1 Konstante Beschleunigung

Dieses Profil wird mittels des Parameters Acceleration_Modes eingestellt:

ConfigurationData.Acceleration_Modes:=0;



Für die folgende Berechnung wird angenommen, dass der Antrieb die Geschwindigkeit $v=v_{\max}$ erreichen soll. Die Beschleunigung wird über das Prozessabbild in Form von $\text{ACCELERATION}=32767$ vorgegeben:

t_1 berechnet sich in diesem Fall gemäß:

$$v_{\max} = 2\text{MHz}/\text{Freq_Div}$$

$$a_{\max} = 32767 \cdot \text{ACC_Fact}/\text{Freq_Div}$$

$$t_1 = v_{\max}/a_{\max} = 2\text{MHz}/(32767 \cdot \text{ACC_Fact})$$

ACC_Fact	80	800	8000
t1	760ms	76ms	7,6ms

5.2.2 Lineare Beschleunigung

Dieses Profil wird z.B. durch die folgende Einstellung erreicht:

ConfigurationData.Acceleration_Modes:=2#01010101

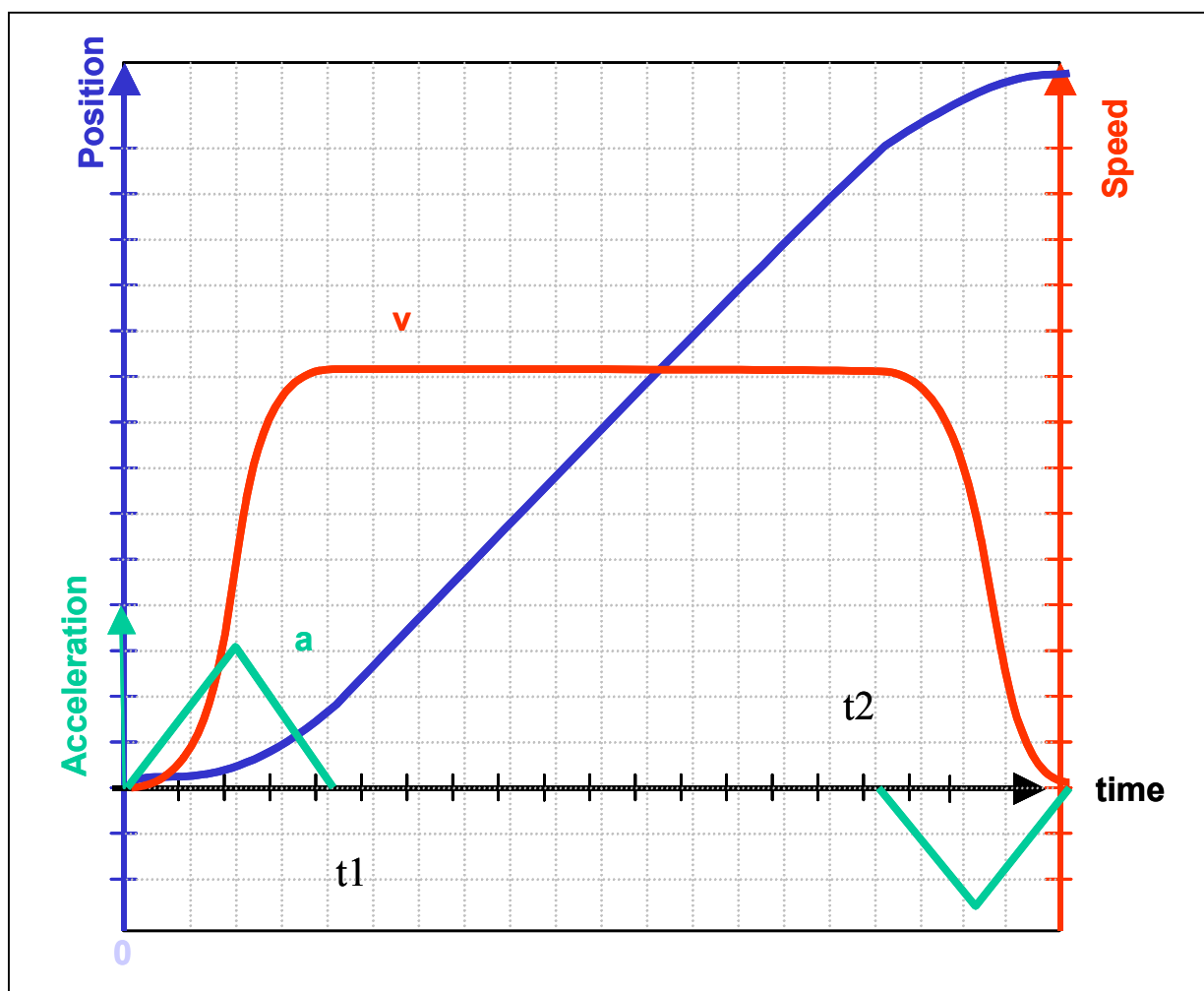
ConfigurationData.Acceleration_RampDown_Param:=2000

ConfigurationData.Acceleration_RampUp_Param:=2000

Die Parameter RampUp und RampDown definieren die Zeit für den Beschleunigungsanstieg bzw. die Verzögerung in ms:

t1=2 sec

t2=2 sec



5.2.3 \sin^2 Beschleunigung

Dieses Profil wird z.B. durch die folgende Einstellung erreicht:

ConfigurationData.Acceleration_Modes:= 2#01100110

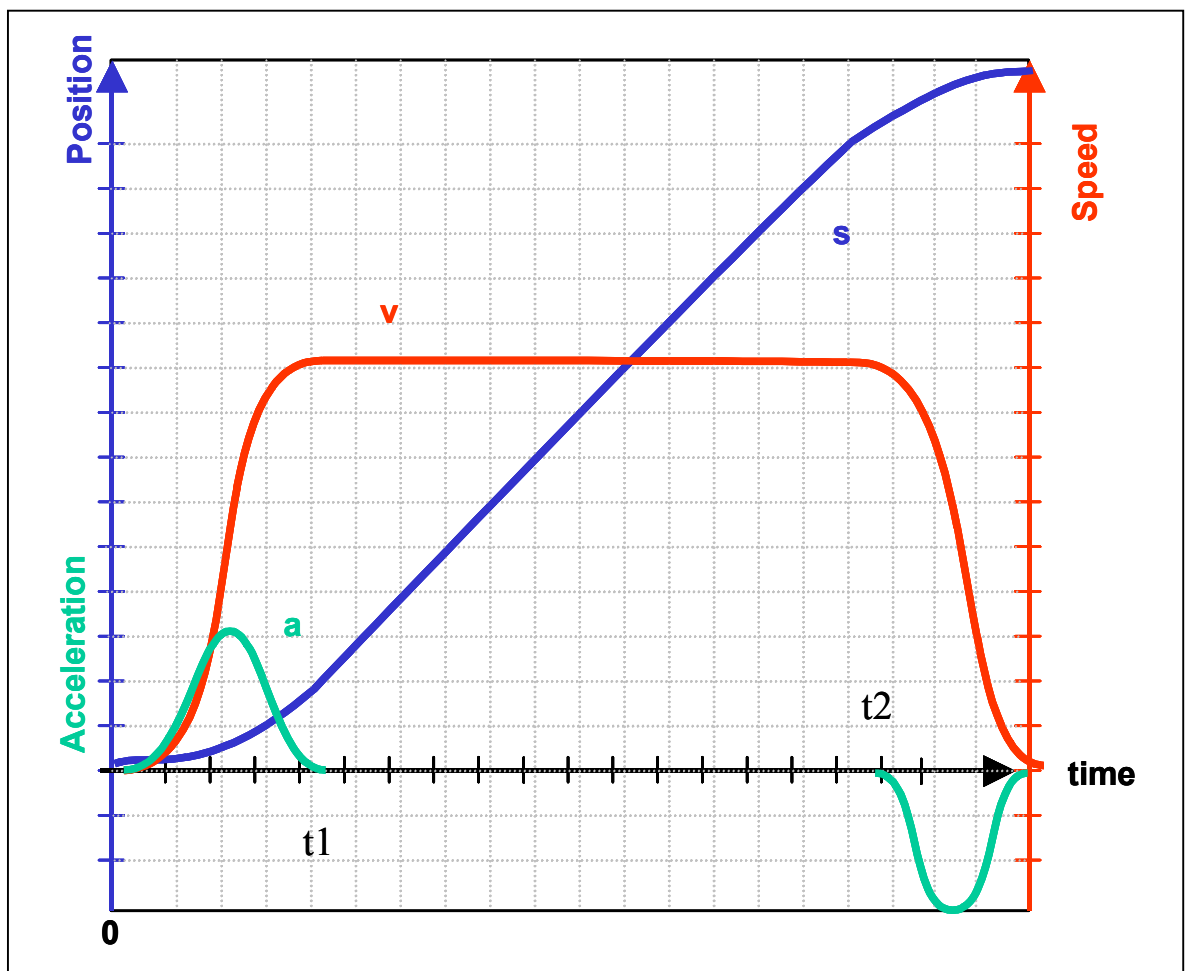
ConfigurationData.Acceleration_RampDown_Param:=2000

ConfigurationData.Acceleration_RampUp_Param:=2000

Die Parameter RampUp und RampDown definieren die Zeit für den Beschleunigungsanstieg bzw. die Verzögerung in ms:

t1=2 sec

t2=2 sec



5.3 Änderung der Bit Zuordnung

Das Stepper Modul erlaubt es, die Standardbelegung der digitalen Eingänge zu verändern. Die Standardbelegung sieht wie folgt aus:

Stop1_N	Eingang 1	(Ptr_Stop1_N:= 16#30)
Set_Reference:	Eingang 2	(Ptr_Set_Reference:= 16#31)
LimitSwitch_Pos	Control 3.4	(Ptr_LimitSwitch_Pos:= 16#54)
LimitSwitch_Neg	Control 3.5	(Ptr_LimitSwitch_Neg:= 16#55)

5.3.1 Beispiel 1

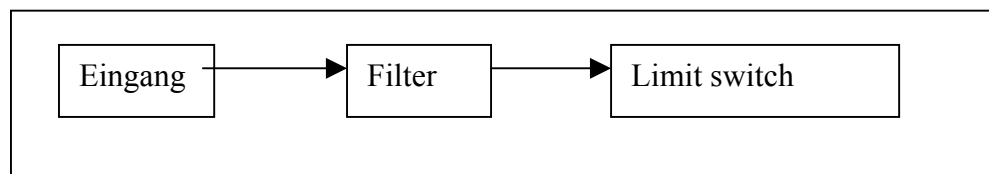
Der positive Endschanter soll auf den ersten digitalen Eingang gelegt werden:

LimitSwitch_Pos	Input 1	(Ptr_LimitSwitch_Pos:= 16#30)
Stop1_N	TRUE	(Ptr_Stop1_N:= 16#01)

5.3.2 Beispiel 2

Die beiden Eingänge sollen für den positiven und negativen Endschanter benutzt werden. Ferner sollen sie drahtbruchsicher arbeiten, d.h. nur wenn der Endschanter bedämpft wird, soll das Signal zu Null werden. Die Standard Logik des Eingangs muss invertiert werden.

Mittels der Filterfunktion lässt sich das Eingangssignal invertieren.



Ptr_FILT1:=0x30	(Eingang 1 auf den Filter 1 geben)
Ptr_FILT2:=0x31	(Eingang 2 auf den Filter 2 geben)
Ptr_C3_LimitSwitch_Pos:=0xA8	(Ausgang Filter 1 mit positivem Endschanter verbinden)
Ptr_C3_LimitSwitch_Neg:=0xA9	(Ausgang Filter 2 mit negativem Endschanter verbinden)
Filter1_Function :=1	(Filter 1 auf „Invertierung“ setzen)
Filter2_Function :=1	(Filter 2 auf „Invertierung“ setzen)
Ptr_Set_Reference:=0	(Referenz Schalter deaktivieren)
Ptr_Stop1_N:=1	(Stop1_N Funktionalität auf TRUE setzen)

5.3.3 Nennstrom Modul 750-671 einstellen

Für das Module 750-671 ist eine Anpassung des Motorstroms vorzunehmen. Der Nennstrom wird durch den Parameter:

ConfigurationData.Current

vorgegeben.

5.3.3.1 Beispiel

Für einen Motor Nennstrom von 0,5A gilt:

ConfigurationData.Current:=5

5.3.4 Berechnung der Anzahl Steps pro Umdrehung für Modul 750-671

Ausgehend von einem Motor mit 1,8° pro Schritt, berechnet sich die Anzahl der Steps pro Umdrehung, aufgrund des 64-fachen Micro-Steppings, gemäß:

$$\frac{\frac{360^\circ}{\text{Umdrehung}}}{\frac{1,8^\circ}{\text{Vollschritt}}} \times 64 \frac{\text{Schritte}}{\text{Vollschritte}} = 12800 \frac{\text{Schritte}}{\text{Umdrehung}}$$

5.3.5 Berechnung Geschwindigkeit und Beschleunigung

Beispiel:

Ein Motor soll mit einer Geschwindigkeit von 200rpm betrieben werden. Diese Geschwindigkeit soll in 0,5s erreicht werden.

Der Motor hat einen Schrittwinkel von 1.8°.

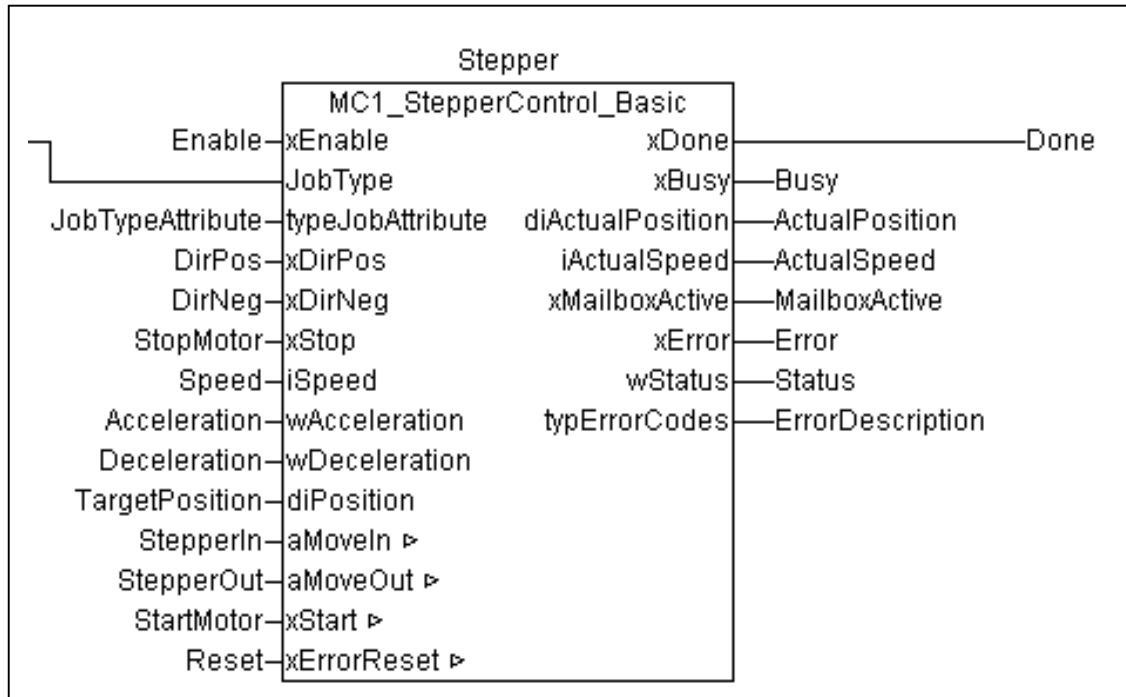
module		750-671	
module oscillator freq.	2000000	Hz	
speed quantisation range	25000	levels	
acc quantisation range	32767	levels	
microsteps/step	64	usteps	
motor			
deg/step	1,8	deg	
steps/revolution	200	steps	
microsteps/revolution	12800	usteps	
required			
max speed	200,0	rpm	
start-ramp time	0,500	sec	
result			
Freq_Div	47	min	max
Acc_fact	122	1	32767

Die Berechnung liefert für Freq_Div =47 und Acc_Fact=122.

Damit die vorgegeben Geschwindigkeit und Beschleunigung erreicht wird, ist am Baustein MC1_StepperControlBasic die Geschwindigkeit iSpeed=25000 und die Beschleunigung wAcceleration=32767 vorzugeben.

6 Positionierung

Einsatz des Funktionsbausteins MC1_StepperControl_Basic



Dieser Baustein ermöglicht verschiedene Bewegungsabläufe des Stepper Motors. Über den Eingang JobType wird zwischen Referenzfahrt (2), Absoluter Positionierung (0), Relativer Positionierung (1), Jogging (3), Geschwindigkeitssteuerung (4) und der Nutzung eines Fahrprogramms (6) unterschieden.

Im Falle eines Stops über den Eingang xStop wird die Schnellstoprampe gemäß acceleration_Stop_Fast benutzt.

Weitergehende Informationen zu diesem Baustein sind in der Bibliotheksbeschreibung zu finden.

Die folgende Tabelle listet auf, welche Werte in den verschiedenen Betriebsarten den Bewegungsablauf beeinflussen:

CFG: Wert wird aus der Konfigurationstabelle entnommen

	Homing	Jogging	MoveAbsolute	MoveRelative	VelocityControl
diPosition	CFG	CFG	TargetPosition	TargetPosition	Don't care
iSpeed	CFG	CFG	Speed	Speed	Speed
wAcceleration	CFG	CFG	Acceleration	Acceleration	Acceleration
wDeceleration	CFG	CFG	Deceleration	Deceleration	Deceleration

7 Fahrprogramm

Zur Nutzung der Fahrprogramm Funktionalität, ist als erstes ein Fahrprogramm zu laden. Der Baustein MC1_Drive_ProgramTable erledigt dieses.

Ein Fahrprogramm ist wie folgt aufgebaut:

Opcode	data_LSB	data	data_MSB	Beschreibung
16#25	16#20	16#4E	16#00	Setze Geschwindigkeit auf 20.000
16#22	16#B8	16#0B	16#00	Setze Beschleunigung auf 3.000
16#04	16#02	16#00	16#00	Fahre (Move_L) auf Position 2 der Positionstabelle
16#70	16#D0	16#07	16#00	Warte 2000 ms

Wenn ein Opcode wie Move_L benutzt wird, welcher sich auf eine Positionstabelle bezieht, muss eine entsprechende Tabelle vorhanden sein. Der Baustein MC1_PositionTableReadWrite ermöglicht das Laden einer Positionstabelle.

Eine Positionstabelle ist gemäß der folgenden Tabelle aufgebaut:

	LSB			MSB	Beschreibung
0	16#00	16#20	16#00	16#00	Position 0->8192
1	16#00	16#40	16#00	16#00	Position 1->16384
2	16#A0	16#86	16#01	16#00	Position 2->100.000
3	16#C0	16#D4	16#01	16#00	Position 3->120.000

Jede Tabellenzeile enthält einen 32 Bit Wert für die Position.

8 Nocken Funktionalität

Die Nocken Funktionalität basiert auf einer Nockentabelle. Mittels des Bausteins MC1_ConfigurationCamTable wird diese Tabelle geladen:

Nocken	LSB		MSB	Beschreibung
16#03	16#FF	16#FF	16#FF	Der erste Eintrag muss negative seien Nocken 1 an, Nocken 2 an, Nocken 3..8 aus Out=2#0000_0011
16#02	16#64	16#00	16#00	Position: 100 Nocken 1 aus, Nocken 2 an, Nocken 3..8 aus Out=2#0000_0010
16#01	16#C8	16#00	16#00	Position 200 Nocken 1 an, Nocken 2 aus, Nocken 3..8 aus Out=2#0000_0001
16#03	16#2C	16#01	16#00	Position 300 Nocken 1 an, Nocken 2 an, Nocken 3..8 aus Out=2#0000_0011

9 Beispiel

Das Beispielprojekt Stepper_02_Example_Visu_v2.pro nutzt die in der Bibliothek vorhandenen Visualisierungen. Über die Visualisierung Configuration_Stepper1 können wesentliche Parameter des Stepper Moduls verändert werden.

Stepper 1

Insert modul settings

Freq_div	200	Acceleration_Mode	0
Acc_Fact	80	Acceleration_RampUp_Param	300
		Acceleration_RampDown_Param	300
Setup_Speed	100		
Setup_Acceleration	10	Current	10
Reference_Mode	0	Current_RampUp	120
Reference_Offset	0	Current_RampDown	90
		Current_Drive	50
Acceleration_StopFast	1000	Current_Standstill	33
		Current_Stop	100
		RotaryAxisPeriode	0

Feedback

active ready error

Read or Write Stepper Configuration

Start "read/write"

Über die Schaltfläche "read/write" wird festgelegt, ob die Konfigurationswerte gelesen oder geschrieben werden sollen. Mit der Schaltfläche „Start“ wird der Vorgang ausgelöst.

Über den Erfolg des Vorgangs wird durch die Flächen „ready“ und „error“ informiert.

Die folgende Abbildung zeigt die Visualisierung mit der der Motor verfahren werden kann.

Dabei ist grundsätzlich folgendermaßen vorzugehen:

- 1) Den Motor freigeben (“enable” Schaltfläche muß grün seien)
- 2) Anwahl **einer** Betriebsart, z.B. MoveAbsolute
- 3) Starten des Bewegungsvorgangs durch einen geeigneten Button, z.B. “Start”

Im Bereich Stepper feedback wird die aktuelle Position sowie die aktuelle Geschwindigkeit ausgegeben.

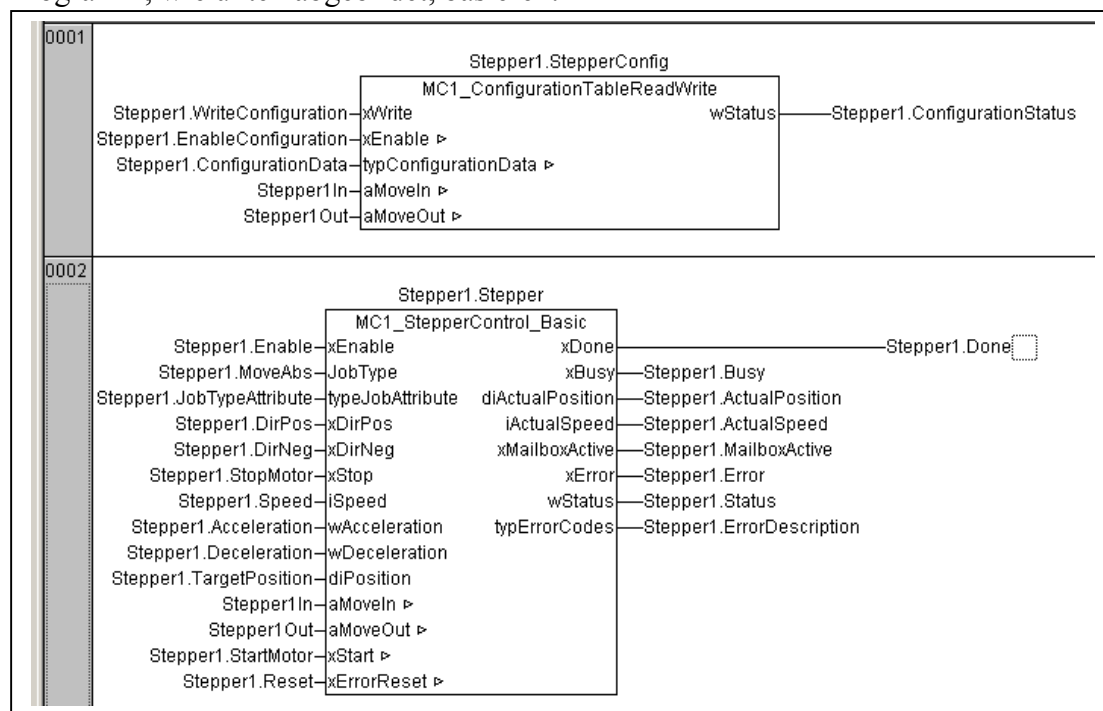
Wird ein Fehler angezeigt, kann dieser über die “Reset“ Schaltfläche zurückgesetzt werden. Der Reset wird nur durchgeführt, wenn keine Betriebsart angewählt ist.

<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;"> <p style="text-align: center; margin: 0;">Please enable Drive</p> <div style="background-color: #4a5a8a; color: white; text-align: center; padding: 10px; margin: 5px;"> Enabled </div> </div> <div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;"> <p style="text-align: center; margin: 0;">Stepper Feedback</p> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="border: 1px solid black; padding: 2px;">Actual Position</td> <td style="border: 1px solid black; text-align: center; width: 100px;">0</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">Actual Velocity</td> <td style="border: 1px solid black; text-align: center;">0</td> </tr> </table> </div> <div style="border: 1px solid black; padding: 5px;"> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="border: 1px solid black; padding: 2px;">Busy</td> <td style="border: 1px solid black; padding: 2px;">Done</td> <td style="border: 1px solid black; padding: 2px;">Mailbox Active</td> <td style="border: 1px solid black; padding: 2px;">Error</td> </tr> <tr> <td colspan="2" style="border: 1px solid black; padding: 2px;">Status</td> <td colspan="2" style="border: 1px solid black; text-align: center;">0</td> </tr> <tr> <td colspan="2" style="border: 1px solid black; padding: 2px;">Error Description</td> <td colspan="2" style="border: 1px solid black; text-align: center;">Drive_OK</td> </tr> </table> </div>	Actual Position	0	Actual Velocity	0	Busy	Done	Mailbox Active	Error	Status		0		Error Description		Drive_OK		<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;"> <p style="text-align: center; margin: 0;">Please activate one mode</p> <div style="background-color: #4a5a8a; color: white; text-align: center; padding: 10px; margin: 5px;"> MoveAbsolute MoveRelative Homing Jogging VelocityCtrl </div> </div> <div style="border: 1px solid black; padding: 5px;"> <p style="text-align: center; margin: 0;">Please insert positioning details</p> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="border: 1px solid black; padding: 2px;">Target Position</td> <td style="border: 1px solid black; text-align: center;">100000</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">Velocity</td> <td style="border: 1px solid black; text-align: center;">20000</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">Acceleration</td> <td style="border: 1px solid black; text-align: center;">3000</td> </tr> </table> </div>	Target Position	100000	Velocity	20000	Acceleration	3000
Actual Position	0																						
Actual Velocity	0																						
Busy	Done	Mailbox Active	Error																				
Status		0																					
Error Description		Drive_OK																					
Target Position	100000																						
Velocity	20000																						
Acceleration	3000																						
<div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p style="text-align: center; margin: 0;">Move Drive</p> <div style="background-color: #4a5a8a; color: white; text-align: center; padding: 10px; margin: 5px;"> Start Stop Reset </div> </div>																							

Aufgrund der Betriebsart MoveAbsolute sind keine Button fürs Joggen sichtbar.

10 Visualisierungen aus der Stepper_02.lib

In der Stepper Bibliothek sind zwei Visualisierungen enthalten die auf einem Programm, wie unten abgebildet, basieren:



Durch das Platzhalterkonzept ist es einfach möglich entsprechende Visualisierungen für einen zweiten oder dritten Motor zu erstellen. Jeder Motor benötigt in diesem Fall eine eigene Variable vom Typ **StepperVisu**, z.B. **Stepper1**, **Stepper2**, usw. . Nachdem die Visualisierung eingefügt wurde, ist der Platzhalter der entsprechenden Variable zuzuordnen.



WAGO Kontakttechnik GmbH & Co. KG
Postfach 2880 • D-32385 Minden
Hansastraße 27 • D-32423 Minden
Telefon: 05 71/8 87 – 0
Telefax: 05 71/8 87 – 1 69
E-Mail: info@wago.com

Internet: <http://www.wago.com>
