

Lenze

Handbuch

IEC 61131-3

inside

***Global Drive
PLC Developer Studio***



Global Drive

Funktionsbibliothek

LenzeMotionControl1.lib

Diese Dokumentation ist gültig für die Funktionsbibliothek **LenzeMotionControl1.lib** ab Version 5.00.

Die Funktionsbibliothek **LenzeMotionControl1.lib** kann für folgende Lenze PLC-Geräte verwendet werden:

	Typenbezeichnung	ab Hardwarestand	ab Softwarestand
9300 Servo PLC	EVS93XX-xT	6A	6.0
ECSxA	ECSxAxxx	1C	7.0

Software-Systemvoraussetzungen:

Drive PLC Developer Studio Professional ab V2.2

Wichtiger Hinweis:

Die Software wird dem Benutzer in der vorliegenden Form zur Verfügung gestellt. Alle Risiken hinsichtlich der Qualität und der durch ihren Einsatz ermittelten Ergebnisse verbleiben beim Benutzer. Entsprechende Sicherheitsvorkehrungen gegen eventuelle Fehlbedienungen sind vom Benutzer vorzusehen.

Wir übernehmen keine Verantwortung für direkt oder indirekt entstandene Schäden, z. B. Gewinnverluste, Auftragsverluste oder geschäftliche Beeinträchtigungen jeglicher Art.

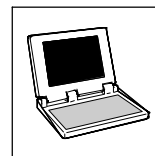
© 2008 Lenze Drive Systems GmbH

Ohne besondere schriftliche Genehmigung von Lenze Drive Systems GmbH darf kein Teil dieser Dokumentation vervielfältigt oder Dritten zugänglich gemacht werden.

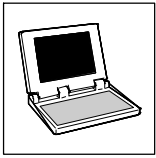
Wir haben alle Angaben in dieser Dokumentation mit größter Sorgfalt zusammengestellt und auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Trotzdem können wir Abweichungen nicht ganz ausschließen. Wir übernehmen keine juristische Verantwortung oder Haftung für Schäden, die dadurch eventuell entstehen. Notwendige Korrekturen werden wir in die nachfolgenden Auflagen einarbeiten.

Alle in dieser Dokumentation aufgeführten Markennamen sind Warenzeichen ihrer jeweiligen Besitzer.

Version 4.0 04/2008 - TD29

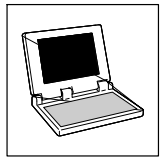


1 Vorwort und Allgemeines	1-1
1.1 Über dieses Handbuch	1-1
1.1.1 Konventionen in diesem Handbuch	1-1
1.1.2 Piktogramme in diesem Handbuch	1-1
1.1.3 Aufbau der Beschreibungen	1-2
1.1.4 Verwendete Begriffe	1-2
1.2 Versionskennung der Funktionsbibliotheken	1-3
2 Funktionsblöcke für Positionieraufgaben	2-1
2.1 L_MCAutoResetInterface - Prozesssteuerung	2-2
2.2 L_MCMachineData - Maschinenparameter	2-4
2.3 L_MCPositionCorrection - Vorschubkorrektur	2-10
2.4 L_MCProfileGenerator - Profilgenerator	2-15
2.5 L_MCResidualDistance - Restwegberechnung	2-28
2.6 L_MCSwitchCam - Wegschaltpunkte (Rundachse)	2-31
2.7 L_MCSwitchPoint - Wegschaltpunkte	2-35
2.8 L_MCTouchProbeCtrl - Touch-Probe Vorverarbeitung	2-38
3 Funktionsblöcke zur Umrechnung von Einheiten	3-1
3.1 L_MCGetPosition	3-2
3.2 L_MCGetVelocity	3-3
3.3 L_MCSetAcceleration	3-4
3.4 L_MCSetDeceleration	3-5
3.5 L_MCSetProfileConverter	3-6
3.6 L_MCSetProfileValues	3-7
3.7 L_MCSetPosition	3-9
3.8 L_MCSetVelocity	3-10
4 Strukturierte Datentypen	4-1
4.1 MC_AXIS_REF	4-1
4.2 MC_PROFILE_REAL_MEASURING	4-3
4.3 MC_PROFILE_INTERNAL_MEASURING	4-4
5 Index	5-1



Funktionsbibliothek LenzeMotionControl1.lib

Inhalt



1 Vorwort und Allgemeines

1.1 Über dieses Handbuch

Dieses Handbuch enthält Informationen zur Funktionsbibliothek **LenzeMotionControl1.lib** für das **Drive PLC Developer Studio**.

- In der Funktionsbibliothek **LenzeMotionControl1.lib** sind Funktionen zur Realisierung von Positionieraufgaben enthalten.
- In der Funktionsbibliothek **LenzeMotionControl2.lib** sind Funktionen zur Realisierung von Achsensynchronisierungsaufgaben enthalten.
- Die Funktionsbibliotheken sind Bestandteile des **Software Package - Positioner**.

1.1.1 Konventionen in diesem Handbuch

Dieses Handbuch verwendet folgende Konventionen zur Unterscheidung von verschiedenen Arten von Informationen:

Variablenbezeichner

sind im erklärenden Text in Kursivschrift dargestellt:

- "Mit *bReset...*"



Tipp!

Informationen zu den Konventionen, die für die Variablenbezeichner von Lenze Systembausteinen, Funktionsblöcken sowie Funktionen verwendet werden, um eine einheitliche und durchgängige Benennung zu gewährleisten und dadurch die Lesbarkeit von SPS-Programmen zu fördern, finden Sie im Anhang der DDS-Online-Dokumentation "Einführung in die IEC 61131-3 Programmierung".

Lenze Funktionen/Funktionsblöcke

erkennen Sie am Namen. Sie beginnen immer mit einem "L_":

- "Mit dem FB **L_MCPProfileGenerator...**"

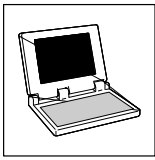
Programmlistings

sind in der Schriftart "Courier" aufgeführt, wobei Schlüsselwörter fett ausgezeichnet sind:

- "**IF** (ReturnValue < 0) **THEN...**"

1.1.2 Piktogramme in diesem Handbuch

	verwendete Piktogramme	Signalwörter	
Warnung vor Sachschäden		Stop!	Warnt vor möglichen Sachschäden . Mögliche Folgen bei Missachtung: Beschädigung des Antriebsreglers/Antriebssystems oder seiner Umgebung.
Sonstige Hinweise		Tipp! Hinweis!	Kennzeichnet einen Tipp bzw. Hinweis.



Funktionsbibliothek LenzeMotionControl1.lib

Vorwort und Allgemeines Über dieses Handbuch

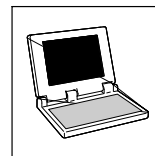
1.1.3 Aufbau der Beschreibungen

Die Beschreibungen der einzelnen Funktionen/Funktionsblöcke sowie Systembausteine in diesem Handbuch sind einheitlich nach folgendem Schema aufgebaut:

	Funktion	Funktionsblock (FB)/ Systembaustein (SB)
①	Überschrift mit Funktion und Bezeichner	
②	Deklaration der Funktion: • Datentyp des Rückgabewertes • Funktionsbezeichner • Liste der Übergabeparameter	-
③	Kurzbeschreibung mit den wichtigsten Eigenschaften	
④	Grafische Darstellung der Funktion mit den zugehörigen Variablen • Übergabeparameter • Rückgabewert	Grafische Darstellung des FB/SB mit den zugehörigen Variablen • Eingangsvariablen • Ausgangsvariablen
⑤	Tabelle mit Informationen zu den Übergabeparametern: • Bezeichner • Datentyp • Einstellmöglichkeiten • Info	Tabelle mit Informationen zu den Ein- und Ausgangsvariablen: • Bezeichner • Datentyp • Variablentyp • Einstellmöglichkeiten • Info
⑥	Tabelle mit Informationen zum Rückgabewert: • Datentyp des Rückgabewertes • Mögliche Rückgabewerte und ihre Bedeutung	-
⑦	Weitere Informationen (Anmerkungen, Tipps, Anwendungsbeispiele, etc.)	

1.1.4 Verwendete Begriffe

Begriff	Im folgenden Text verwendet für
Cxxx/y	Subcode y der Codestelle Cxxxx (z. B. C0470/3 = Subcode 3 der Codestelle C0470)
DDS	Drive PLC Developer Studio
FB	Funktionsblock
GDC	Global Drive Control (Parametrier-Programm von Lenze)
Parameter-Codestellen	Codestellen zum Einstellen der Funktionalität eines Funktionsblocks
PLC	9300 Servo PLC - Achsmodul ECSxA "Application"
SB	Systembaustein
[unit]	Platzhalter für die physikalische Einheit des verwendeten externen Maßsystems (z. B. "µm", "mm", "cm")



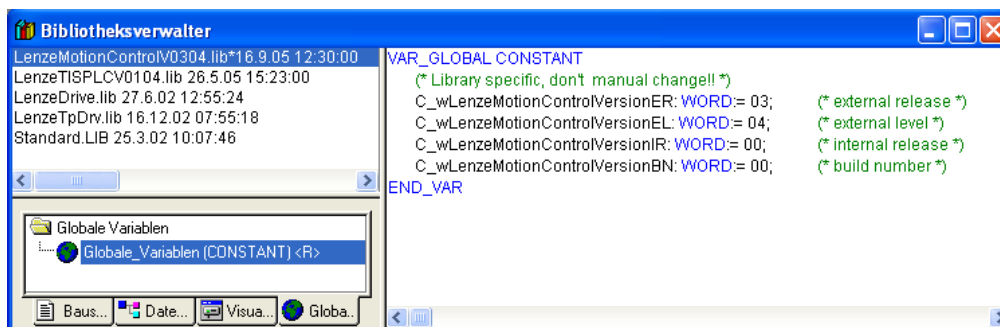
1.2 Versionskennung der Funktionsbibliotheken

Versionskennung **ab Version 3.04:**

Konstante	Bedeutung	Beispiel Wert
C_w[Funktionsbibliotheksname]VersionER	Externes Release	03
C_w[Funktionsbibliotheksname]VersionEL	Externes Level	04
C_w[Funktionsbibliotheksname]VersionIR	Internes Release	00
C_w[Funktionsbibliotheksname]VersionBN	Build-Nummer	00

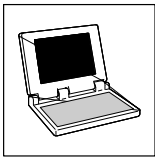
Version: 03 04 00 00

- Im Beispiel bedeutet der Wert "03040000" Version "3.04.00.00".



Im Offline-Modus können Sie sich den Wert dieser Konstanten über die **Bibliotheksverwaltung/Register Globale Variablen** anzeigen lassen:

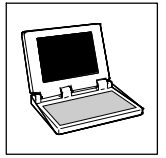
- Die Bibliotheksverwaltung öffnen Sie im DDS über den Menü-Befehl **Fenster→Bibliotheksverwaltung**.
- Wählen Sie die Registerkarte **Globale Variablen** und öffnen Sie den Ordner **Globale Variablen**.
- Wenn Sie die *Globale_Variablen...* markieren, erscheint die Version im Bibliotheksverwalter.



Funktionsbibliothek LenzeMotionControl1.lib

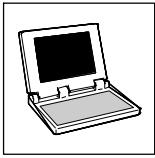
Vorwort und Allgemeines

Versionskennung der Funktionsbibliotheken



2 Funktionsblöcke für Positionieraufgaben

Dieses Kapitel enthält Informationen zu den Funktionsblöcken, die Sie für Positionieraufgaben verwenden können.



Funktionsbibliothek LenzeMotionControl1.lib

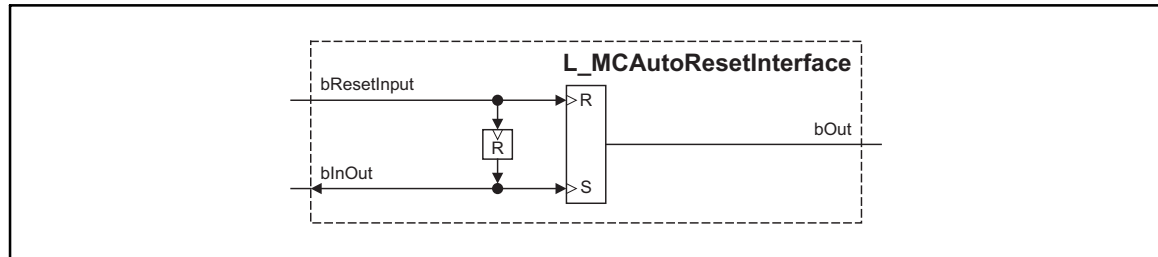
Funktionsblöcke für Positionieraufgaben

L_MCAutoResetInterface - Prozesssteuerung

2.1 L_MCAutoResetInterface - Prozesssteuerung

Oftmals ist es erforderlich, von einem Prozess (A) aus einen anderen Prozess (B) zu starten und dann dessen Abarbeitung abzuwarten. Mit diesem FB kann eine solche Prozesssteuerung in einfacher Form realisiert werden.

- Der Steuerausgang *bOut* zum Starten von Prozess (B) wird beim nächsten Aufruf des FB automatisch zurückgesetzt.



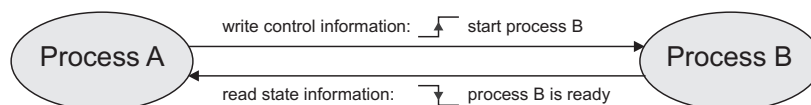
⇒ Eingänge (Variablentyp: VAR_INPUT)			
Bezeichner	Datentyp	Signaltyp	Info/Einstellmöglichkeiten
bResetInput	Bool	binary	Steuereingang zum Zurücksetzen von <i>bOut</i> und <i>bInOut</i> . FALSE ↗ TRUE <i>bOut</i> und <i>bInOut</i> werden auf FALSE zurückgesetzt.

⇒ Ein-/Ausgänge ⇒ (Variablentyp: VAR_IN_OUT)			
Bezeichner	Datentyp	Signaltyp	Info/Einstellmöglichkeiten
bInOut	Bool	-	Ein-/Ausgangsvariable, die sowohl als Steuereingang als auch als Statusausgang verwendet werden kann. Eingangsinformation: FALSE ↗ TRUE <i>bOut</i> wird bis zum nächsten FB-Aufruf auf TRUE gesetzt. Ausgangsinformation: TRUE ↘ FALSE <i>bOut</i> wurde über <i>bResetInput</i> auf FALSE zurückgesetzt.

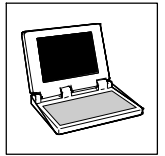
Ausgänge ⇒ (Variablentyp: VAR_OUTPUT)			
Bezeichner	Datentyp	Signaltyp	Info
bOut	Bool	-	Steuerausgang • Dieser Ausgang wird über eine FALSE-TRUE-Flanke an <i>bInOut</i> auf TRUE gesetzt und beim nächsten Aufruf des FB automatisch auf FALSE zurückgesetzt.

Verwendung einer Schnittstellen-Variable

Als Schnittstelle zwischen den beiden Prozessen dient eine Variable vom Typ VAR_IN_OUT, die sowohl Steuerinformationen (Prozess B starten) als auch Statusinformationen (Prozess B ist fertig) enthält:



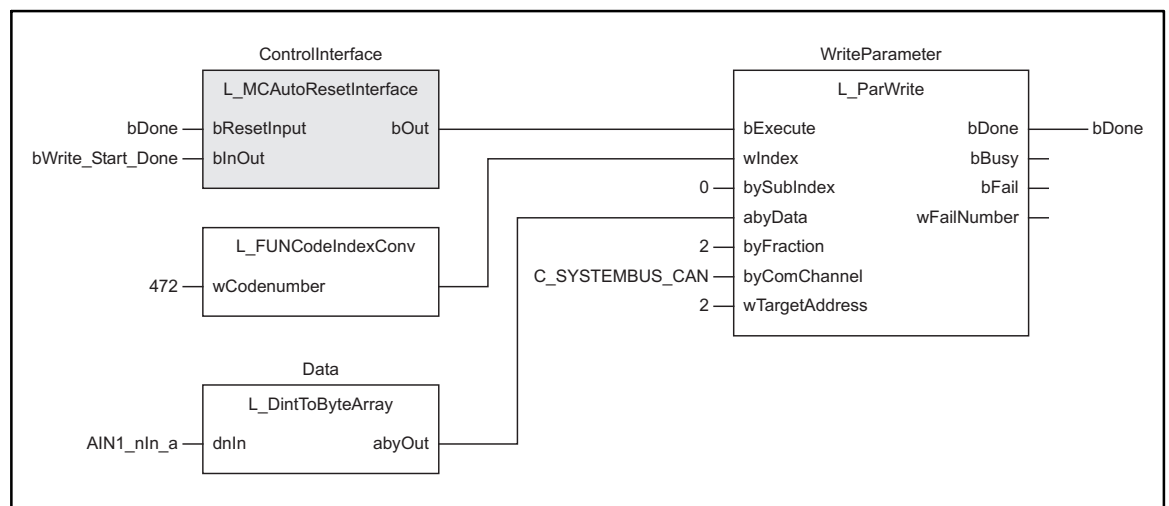
- Diese Schnittstellen-Variable ist mit *bInOut* zu verbinden.

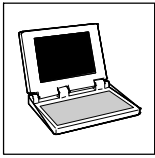


Typischer Ablauf

1. Prozess A setzt die mit *blnOut* verbundene Schnittstellen-Variable von FALSE auf TRUE. Hierdurch wird der Ausgang *bOut* von FALSE auf TRUE gesetzt, über den der Prozess B gestartet wird.
2. Mit dem nächsten Aufruf des FB's wird *bOut* automatisch wieder auf FALSE zurückgesetzt.
3. Prozess B meldet seine fertige Abarbeitung durch eine FALSE-TRUE-Flanke am Eingang *bResetInput*. Hierdurch wird die Schnittstellen-Variable von TRUE auf FALSE zurückgesetzt.
4. Prozess A wertet die Schnittstellen-Variable auf den Zustand FALSE aus und erkennt das Ende der Abarbeitung von Prozess B.

Verschaltungsbeispiel





Funktionsbibliothek LenzeMotionControl1.lib

Funktionsblöcke für Positionieraufgaben

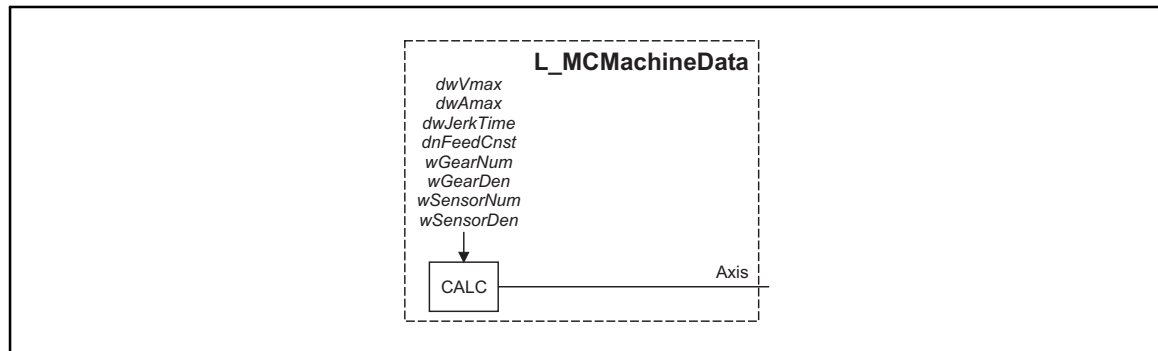
L_MCMachineData - Maschinenparameter

2.2

L_MCMachineData - Maschinenparameter

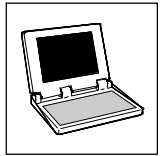
Dieser FB bereitet die Maschinenparameter, die Sie in den physikalischen Einheiten der Maschine vorgeben, für das interne Maßsystem des Profilersators auf.

- Die aufbereiteten Maschinenparameter legt der FB in einer Datenstruktur (MC_AXIS_REF (4-1)) ab.
- Über den Ausgang *Axis* können Sie diese Datenstruktur einer globalen Variable zuweisen, um über diese "Schnittstelle" die Maschinenparameter an andere MotionControl-FB's zu übergeben.



Retain-Variablen (Variablentyp: VAR CONSTANT RETAIN)				
Bezeichner	Datentyp	Signaltyp	Info/Einstellmöglichkeiten	
dwVmax	Double Word	-	Maximale Verfahrgeschwindigkeit in [unit/s] • 10000 = 1.0000 unit/s	
dwAmax	Double Word	-	Maximale Beschleunigung in [unit/s²] • 10000 = 1.0000 unit/s²	
dwJerkTime	Double Word	-	Ruckzeit • 10000 = 1.0000 s	
dnFeedCnst	Double Integer	-	Vorschub-Konstante [unit/r] • Vorschub der Maschine in Einheiten pro Umdrehung der Lastseite des Getriebes. • 10000 = 1.0000 [unit/r]	
wGearNum	Word	-	Getriebefaktor Zähler	
wGearDen	Word	-	Getriebefaktor Nenner	
wSensorNum	Word	-	Lagegeber Getriebefaktor Zähler	• Getriebefaktor zwischen Motor und Lagegeber. • Zähler/Nenner entspricht Motordrehzahl/Geberdrehzahl. • Geber auf Motorwelle: 1/1
wSensorDen	Word	-	Lagegeber Getriebefaktor Nenner	

Ausgänge ⇨ (Variablentyp: VAR_OUTPUT)			
Bezeichner	Datentyp	Signaltyp	Info
Axis	MC_AXIS_REF	-	Über diesen Ausgang können die Maschinenparameter anderen MotionControl-Funktionsblöcken zur Verfügung gestellt werden.



Parameter-Codestellen der Instanzen

Bezeichner	L_MCMachineData1	Einstellbereich			Lenze
dwVmax	C1240	0.0001	{0.0001 unit/s}	214000.0000	50.0000 unit/s
dwAmax	C1250	0.0001	{0.0001 unit/s ² }	214000.0000	100.0000 unit/s ²
dwJerkTime	C1256	0	{0.001 s}	10.000	0.000 s
dnFeedCnst	C1204	0.0001	{0.0001 unit/r}	214000.0000	1.0000 unit/r
wGearNum	C1202	1	{1}	65535	1
wGearDen	C1203	1	{1}	65535	1
wSensorNum	C1200	1	{1}	65535	1
wSensorDen	C1201	1	{1}	65535	1



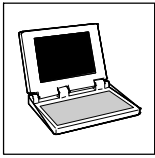
Achtung!

- Die Instanz des FB *L_MCMachineData* deklarieren Sie lokal in PLC_PRG.
- Wenn Sie nicht die Erstinstanz *L_MCMachineData1* verwenden, stellen Sie sicher, dass die Parameter der verwendeten Instanz > 0 sind, um bei Folgeberechnungen eine "Division durch Null" zu vermeiden!



Tipp!

Verbinden Sie den Ausgang *Axis* mit einer globalen Variable (z. B. *g_Axis*), um über diese "Schnittstelle" die Maschinenparameter anderen FB's für Positionieraufgaben zur Verfügung zu stellen.



Funktionsbibliothek LenzeMotionControl1.lib

Funktionsblöcke für Positionieraufgaben

L_MCMachineData - Maschinenparameter

Bestimmung und Eingabe der Maschinenparameter

Die Maschinenparameter werden über Parameter-Codestellen eingegeben, die für die Instanz des FB's im **Instanz-Parametermanager** anzulegen sind.



Tipp!

Wählen Sie für diesen FB den Instanznamen *L_MCMachineData1*, um die Lenze-Codestellen (Erst-Instanz-Codestellen) zu verwenden.

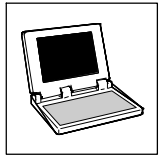
- Auch bei Verwendung der Erstinstanz sind die entsprechenden Codestellen noch im **Instanz-Parametermanager** anzulegen.

In der folgenden Tabelle sind die Maschinenparameter aufgeführt, deren Eingabe unbedingt erforderlich ist:

Maschinenparameter	Einheit	Codestelle*	Info
Getriebefaktor (Zähler)	-	C1202	Eingabe entsprechend den Angaben auf dem Typenschild des Getriebes.
Getriebefaktor (Nenner)	-	C1203	
Getriebefaktor Lagegeber (Zähler)	-	C1200	Eingabe Getriebefaktor zwischen Motor und Lagegeber.
Getriebefaktor Lagegeber (Nenner)	-	C1201	
Vorschubkonstante (V_k)	[unit/r]	C1204	Einzugeben ist die Anzahl in [unit] (z. B. [mm]), die bei einer Umdrehung der Getriebeabtriebsseite vorgeschoben werden soll.
Maximale Geschwindigkeit (v_{max})	[unit/s]	C1240	Diese Begrenzung gilt vorrangig für die gesamte Maschine. Achtung: Die Begrenzung darf max. so hoch eingestellt sein, dass sie mit der max. Motordrehzahl auch noch erreichbar ist: $v_{max} \leq n_{max} \cdot \frac{\text{Vorschubkonstante}}{60} \cdot \frac{\text{Getriebe (Nenner)}}{\text{Getriebe (Zähler)}}$ $C1240 \leq C0011 \cdot \frac{C1204}{60} \cdot \frac{C1203}{C1202}$
Maximale Beschleunigung (a_{max})	[unit/s ²]	C1250	
Ruckzeit	[s]	C1256	
Maximale Motordrehzahl (n_{max})	[rpm]	C0011	Diese Begrenzung gilt vorrangig für den Motor.

* Parameter-Codestelle der Erstinstanz L_MCMachineData1

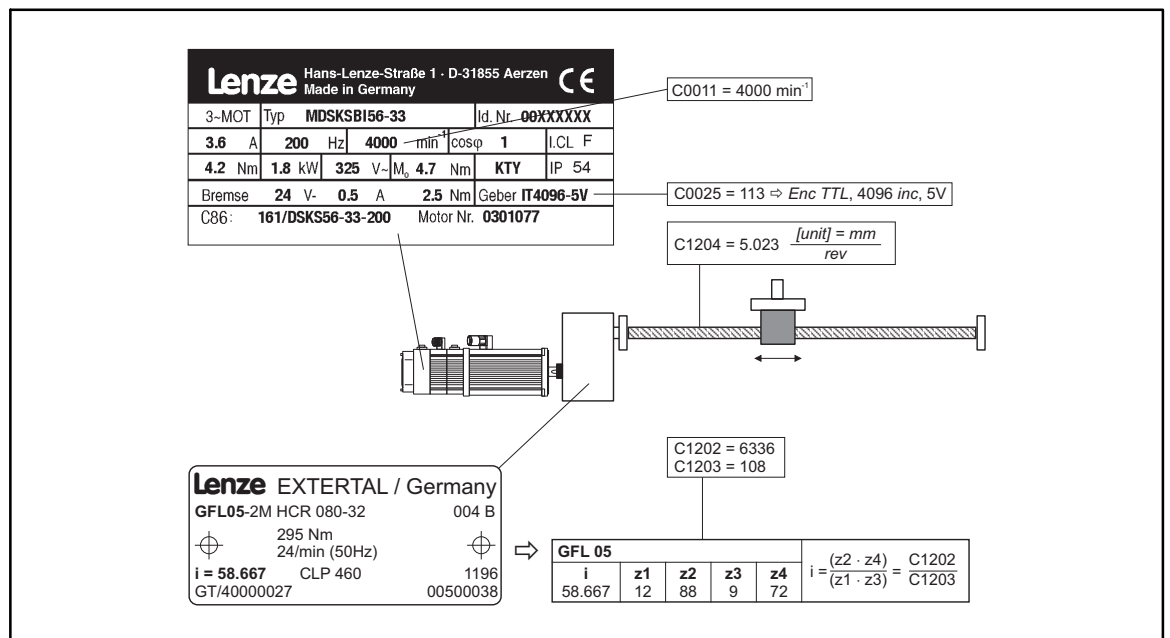
- Die physikalische Einheit wird über die Eingabe für [unit] definiert, z. B. [mm], [m] oder [Grad].



Beispiel

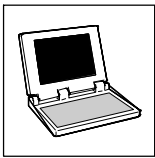
Zur Positionierung wird eine Spindel-Vorschubeinheit über ein Getriebe angetrieben.

- Als Rückführsystem wird nicht der standardmäßige Resolver, sondern ein Inkrementalgeber verwendet. Der Inkrementalgeber ist am Motor montiert und hat eine Strichzahl von 4096 Impulsen/Umdr.
- Das Getriebe hat eine Übersetzung von $i = 58.667$ ($n_{\text{Motor}}/n_{\text{Spindel}}$), die in der Abbildung aufgeführten Zähnezahlen (z1 ... z4) der Verzahnungsteile wurden den Technischen Daten zum Getriebe entnommen.
- Die Spindel hat eine Steigung von $h = 5.023$ mm.
- Die Eingaben sollen in [mm] erfolgen (1 mm = 1 unit).



Einstellungen für die Parameter der PLC (z. B 9300 Servo PLC) sowie der Erstinstanz (**L_MCMachineData1**) des FB **L_MCMachineData**:

Maschinenparameter	Codestelle	Eingabe	Info
Servo PLC: Auswahl Gebersystem	C0025	-113-	-113- ⇒ Inkrementalgeber, IT-4096-5V
ECSxA: Auswahl Encodersystem an X8	C0419		
Encoder- Konstante X8	C0420	4096	Konstante in [Inc]
Lage-Rückführsystem	C0490	-1-	-1- ⇒ Encoder-TTL
Drehzahl-Rückführsystem	C0495	-1-	-1- ⇒ Encoder-TTL
Getriebefaktor (Zähler)	C1202	6336	= z2 * z4 = 88 * 72; entspricht Drehzahl Motor
Getriebefaktor (Nenner)	C1203	108	= z1 * z3 = 12 * 9; entspricht Drehzahl Getriebeabtrieb
Vorschubkonstante	C1204	5.023	Getriebeabtrieb in [unit/rev] Die physikalische Einheit für [unit] in diesem Beispiel ist [mm] (Getriebeabtrieb = 5.023 mm/Umdrehung)
Lagegeber-Getriebefaktor (Zähler)	C1200	1	Zähler entspricht Motordrehzahl Faktor 1/1 = kein Getriebe zwischen Geber und Motor
Lagegeber-Getriebefaktor (Nenner)	C1201	1	Nenner entspricht Geberdrehzahl Faktor 1/1 = kein Getriebe zwischen Geber und Motor



Funktionsbibliothek LenzeMotionControl1.lib

Funktionsblöcke für Positionieraufgaben

L_MCMachineData - Maschinenparameter

Verwendung S-förmiger Rampen (S-Profile)

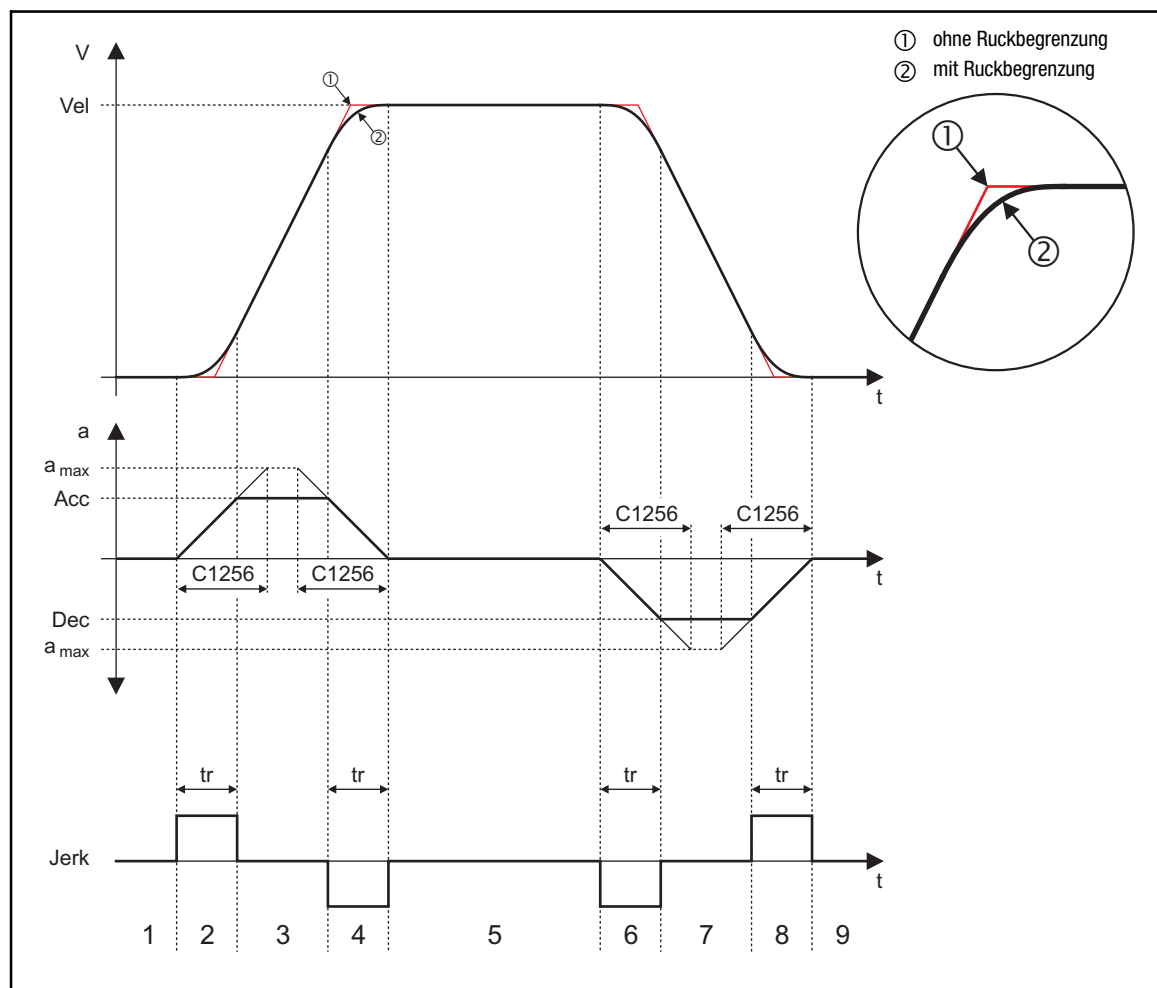
Durch die Verwendung S-förmiger Rampen (S-Profile) werden Beschleunigungs- und Bremsvorgänge zwecks Reduzierung des Rucks und somit Schonung der Antriebskomponenten sanft eingeleitet.

- Die vorgegebene Beschleunigung/Verzögerung (Acc/Dec) wird erst nach Ablauf der Ruck-Zeit (t_r) erreicht.



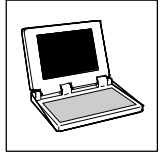
Tipp!

Der langsamere Beschleunigungsanstieg beim S-Profil führt prinzipbedingt dazu, dass sich die Positionierzeit im Vergleich zum zeitoptimalen L-Profil verlängert.



1. Stillstand
2. Beschleunigen mit eingestellter Ruckbegrenzung
3. Beschleunigen gemäß Beschleunigungsprofil (Acc)
4. Beschleunigung verringern (Ruckbegrenzung)
5. Verfahren mit Vel gemäß Geschwindigkeitsprofil

6. Verzögern mit eingestellter Ruckbegrenzung
7. Verzögern gemäß Verzögerungsprofil (Dec)
8. Verzögerung verringern (Ruckbegrenzung)
9. Stillstand (Lage-Ziel erreicht)



Hinweis!

Werden bei kleinen Beschleunigungszeiten unverhältnismäßig hohe S-Rampenzeiten (Ruckzeiten) eingestellt, so kann dies zu einer fehlerhaften Profilgenerierung führen.

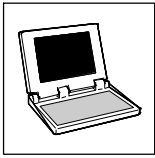
Beispiel: $v = 100 \text{ mm/s}$, $a = 1000 \text{ mm/s}^2$

$$\Rightarrow t_{\text{Beschleunigung}} = 0.1 \text{ s}$$

$$\Rightarrow T_{\text{Ruck}} = 1 \text{ s}$$

Stellen Sie daher nur plausible S-Rampenzeiten ein, diese sollten den halben Wert von $t_{\text{Beschleunigung}}$ nicht übersteigen.

In der **LenzeMotionControl1.lib** wird die Profilgenerierung mit dem Funktionsblock *L_MCProfileGenerator* realisiert. Über den Parameter *wSProfileMode* des Funktionsblocks *L_MCProfileGenerator* kann eine profiloptimierte S-Profilgenerierung eingestellt werden. Bei Einstellung *wSProfileMode = 1* gibt es bezüglich der Profilparameter keine Einschränkungen. Die profiloptimierte S-Profilgenerierung benötigt eine deutlich höhere Bearbeitungszeit. (S. 2-15)).



Funktionsbibliothek LenzeMotionControl1.lib

Funktionsblöcke für Positionieraufgaben

L_MCPositionCorrection - Vorschubkorrektur

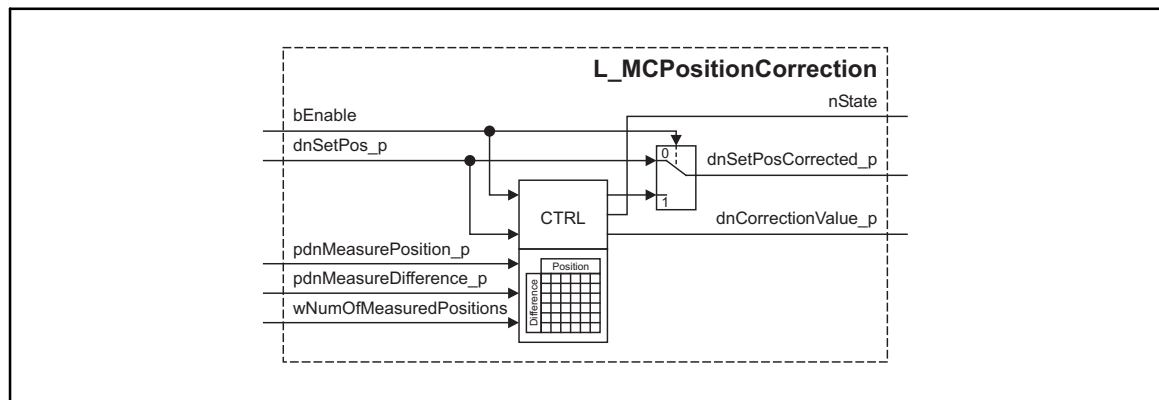
2.3

L_MCPositionCorrection - Vorschubkorrektur

Aufgrund von Linearitätsfehlern an Vorschubmechaniken (z. B. Spindeln) oder Messsystemen ist es im Werkzeugmaschinenbereich üblich, diese Fehler in den NC-Achssteuerungen durch entsprechende Korrekturwerte für den Lagesollwert zu kompensieren.

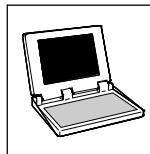
Dieser FB liefert einen korrigierten Positionssollwert anhand einer Messwerttabelle, in der zu verschiedenen Messpositionen die entsprechenden Positionsfehler aufgeführt sind.

- Diese Tabelle wird dem FB in Form zweier Datenfelder (Arrays) übergeben.
- Der korrigierte Positionssollwert kann z. B. einem Profilgenerator zur Weiterverarbeitung übergeben werden.



⇒ Eingänge (Variablentyp: VAR_INPUT)							
Bezeichner	Datentyp	Signaltyp	Info/Einstellmöglichkeiten				
bEnable	Bool	binary	Positionssollwertkorrektur aktivieren <table><tr><td>FALSE</td><td>Positionssollwertkorrektur deaktiviert, <i>dnSetPos_p</i> wird direkt am Ausgang <i>dnSetPosCorrected_p</i> ausgegeben.</td></tr><tr><td>TRUE</td><td>Positionssollwertkorrektur aktiviert.</td></tr></table>	FALSE	Positionssollwertkorrektur deaktiviert, <i>dnSetPos_p</i> wird direkt am Ausgang <i>dnSetPosCorrected_p</i> ausgegeben.	TRUE	Positionssollwertkorrektur aktiviert.
FALSE	Positionssollwertkorrektur deaktiviert, <i>dnSetPos_p</i> wird direkt am Ausgang <i>dnSetPosCorrected_p</i> ausgegeben.						
TRUE	Positionssollwertkorrektur aktiviert.						
dnSetPos_p	Double Integer	position	Sollwertposition in [inc]				
pdnMeasurePosition_p	Pointer to Double Integer	pointer	Zeiger auf den ersten Wert des Datenfeldes (Arrays) mit den Messpositionen				
pdnMeasure-Difference_p	Pointer to Double Integer	pointer	Zeiger auf den ersten Wert des Datenfeldes (Arrays) mit den Positionsfehlern				
wNumOfMeasured-Positions	Word	-	Anzahl der Messpositionen im Datenfeld (Array) • Als Anzahl sind nur 2ⁿ-Werte zulässig: 2, 4, 8, 16, 32, 64, 128, usw.				

Ausgänge ⇨ (Variablentyp: VAR_OUTPUT)									
Bezeichner	Datentyp	Signaltyp	Info						
nState	Integer	-	Status <table><tr><td>0</td><td>Ok</td></tr><tr><td>10</td><td>Anzahl <i>wNumOfMeasurePositions</i> ist kein 2ⁿ-Wert, die Korrekturrechnung wird nicht durchgeführt.</td></tr><tr><td>20</td><td>Korrektur nicht aktiv</td></tr></table>	0	Ok	10	Anzahl <i>wNumOfMeasurePositions</i> ist kein 2 ⁿ -Wert, die Korrekturrechnung wird nicht durchgeführt.	20	Korrektur nicht aktiv
0	Ok								
10	Anzahl <i>wNumOfMeasurePositions</i> ist kein 2 ⁿ -Wert, die Korrekturrechnung wird nicht durchgeführt.								
20	Korrektur nicht aktiv								
dnSetPosCorrected_p	Double Integer	position	Korrigierte Sollwertposition in [inc] Bei <i>bEnable</i> = FALSE ist <i>dnSetPosCorrected_p</i> = <i>dnSetPos_p</i>.						
dnCorrectionValue_p	Double Integer	position	Korrekturwert (interpolierter Positionskorrekturwert)						



Erstellen der Messwerttabelle

Üblicherweise werden hierzu verschiedene Messpunkte (φ_{ist}) an der Achse angefahren und mit einem Referenzmesssystem (z. B. Lasermessgerät) werden die jeweiligen Abweichungen ($\Delta\varphi$) von der vorgegebenen Sollposition (φ_{soll}) ermittelt:

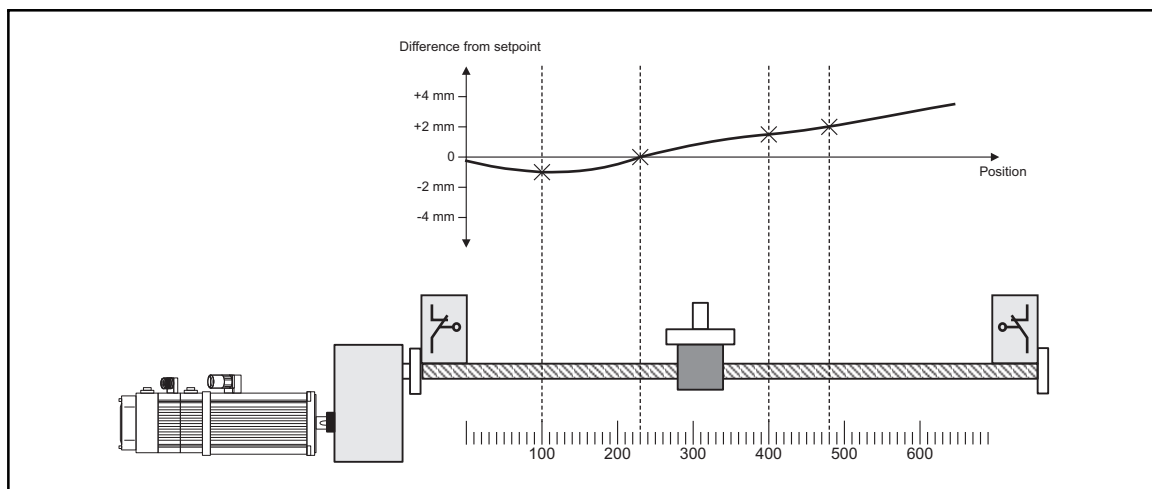


Abb. 2-1

Erfassen von Positionsfehlern an verschiedenen Messpunkten

Die ermittelten Werte können dann in eine Messwerttabelle eingetragen werden:

Messung Nr.	Angefahrte Sollposition φ_{soll}	gemessene Abweichung von der Sollposition $\Delta\varphi = \varphi_{ist} - \varphi_{soll}$
1	100 mm	-1.00 mm
2	230 mm	0.00 mm
3	400 mm	1.80 mm
4	480 mm	2.00 mm
...	...	...
n	___ mm	___ mm

Übergabe der Messwerttabelle an den FB

Im Programm ist diese Messwerttabelle in Form zweier Datenfelder (Arrays) anzulegen, einem Datenfeld mit den Messpositionen und einem Datenfeld mit den entsprechenden Positionsfehlern, wie im folgenden Beispiel gezeigt:

VAR

adnMeasurePos: **ARRAY**[0..15] **OF** DINT;

adnMeasureDif: **ARRAY**[0..15] **OF** DINT;

END_VAR

(* 1 rev = 65535 inc = 1 mm *)

adnMeasurePos[0] := 6553500; (* 100 mm *)

adnMeasureDif[0] := -65535; (* -1 mm *)

adnMeasurePos[1] := 15073050; (* 230 mm *)

adnMeasureDif[1] := 0; (* 0 mm *)

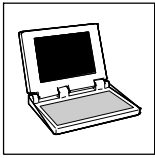
adnMeasurePos[2] := 26214000; (* 400 mm *)

adnMeasureDif[2] := 117963; (* 1.8 mm *)

adnMeasurePos[3] := 31456800; (* 480 mm *)

adnMeasureDif[3] := 1310702; (* 2 mm *)

...



Funktionsbibliothek LenzeMotionControl1.lib

Funktionsblöcke für Positionieraufgaben

L_MCPositionCorrection - Vorschubkorrektur

Dem FB ist dann am Eingang *pdnMeasurePosition_p* ein Zeiger auf den ersten Wert des Arrays mit den Messpositionen und am Eingang *pdnMeasureDifference_p* ein Zeiger auf den ersten Wert des Arrays mit den Positionsfehlern zu übergeben:

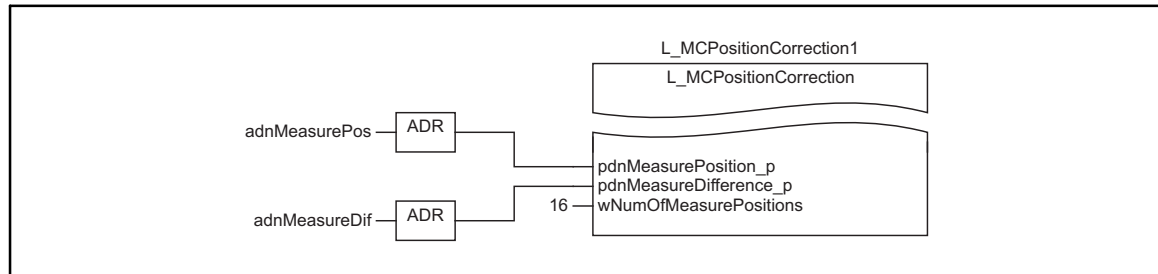


Abb. 2-2

Übergabe der Messwerttabelle an den FB



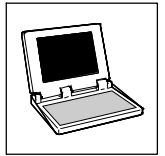
Hinweise

- Die Messpositionen müssen nicht äquidistant sein, auch muss nicht bei der Messposition "0" begonnen werden.
- Im Datenfeld müssen die Messpositionen in aufsteigender Reihenfolge eingetragen werden, gleiche Messpositionen sind nicht erlaubt.
- Als Datenfeldgröße sind nur 2^n -Werte zulässig, also z. B. 16, 32, 64, 128, 256, usw.



Tipp!

- Wenn Sie die Datenfelder mit User-Codestellen verknüpfen, können Sie die Messpunkte und Positionsfehler bequem über Codestellen/Subcodes einstellen und online im Betrieb auf die optimalen Ergebnisse anpassen.
- Die Rechenzeit des FB steigt linear mit der Anzahl der Messpunkte. Legen Sie daher die Datenfeldgröße über *wNumOfmeasurePositions* nicht unnötig groß an, um keine Rechenzeit zu verschwenden.



Berechnung der Korrekturwerte

Die FB berechnet einen Korrekturwert, der von der Sollposition subtrahiert wird. Liegt die Sollposition zwischen zwei Messwerten, so wird die Abweichung und der daraus resultierende Korrekturwert durch Interpolation ermittelt.

Die folgende Darstellung beschreibt das Interpolationsverfahren zur Ermittlung des korrigierten Positionswertes zwischen zwei Stützstellen $P_{(n-1)}$ und $P_{(n)}$, wobei gilt: $P_{(n)} > P_{(n-1)}$

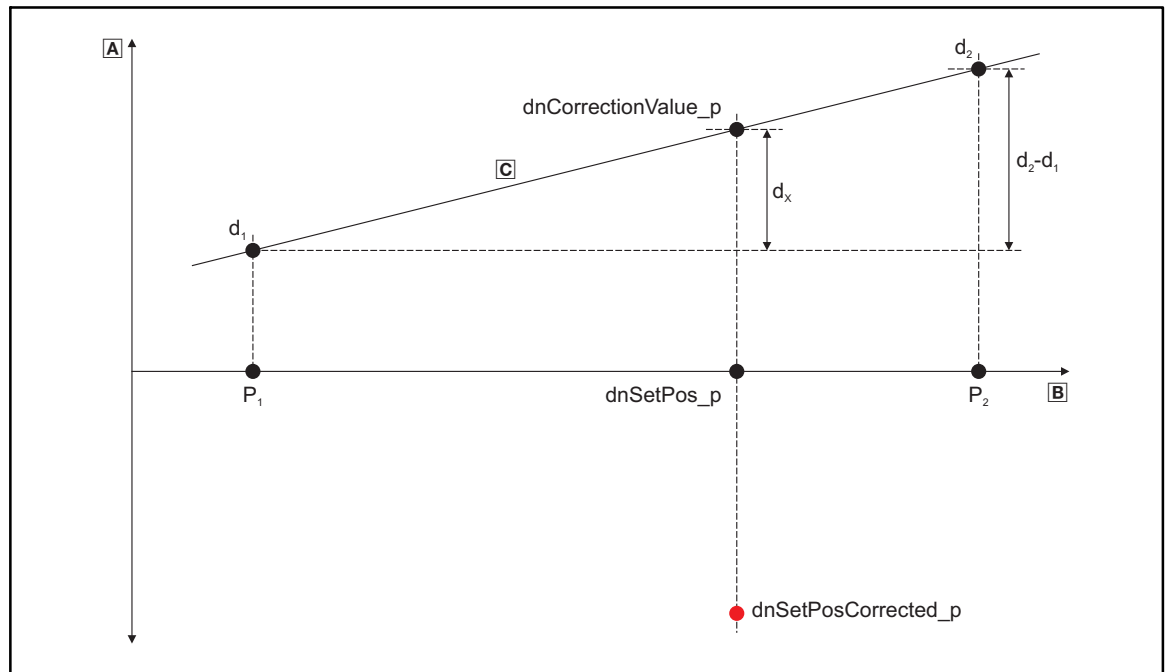


Abb. 2-3

Interpolationsverfahren

A Positionsfehler

B Messposition

C Fehlerinterpolationsgerade

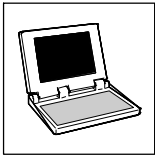
Geradensteigung (Fehlerinterpolationsgerade)	$= \frac{d_2 - d_1}{P_2 - P_1}$
d_x	$= \text{Geradensteigung} \cdot (\text{dnSetPos}_p - P_1)$ $= \frac{d_2 - d_1}{P_2 - P_1} \cdot (\text{dnSetPos}_p - P_1)$
Korrekturwert $\text{dnCorrectionValue}_p$	$= d_x + d_1$ $= \frac{d_2 - d_1}{P_2 - P_1} \cdot (\text{dnSetPos}_p - P_1) + d_1$
Korrigierter Positionswert $\text{dnSetPosCorrected}_p$	$= \text{dnSetPos}_p - \text{dnCorrectionValue}_p$ $= \text{dnSetPos}_p - \left(\frac{d_2 - d_1}{P_2 - P_1} \cdot (\text{dnSetPos}_p - P_1) + d_1 \right)$



Hinweis!

Die Berechnung erfolgt durch Multiplikation der Terme $(d_2 - d_1)$ und $(P_2 - P_1)$.

Das Ergebnis der Multiplikation $(d_2 - d_1) \times (P_2 - P_1)$ darf nicht größer $2^{31}-1$ oder kleiner -2^{31} sein, da ansonsten ein interner Überlauf stattfindet und das Interpolationsergebnis falsch ist!



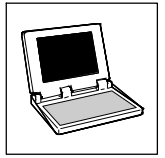
Funktionsbibliothek *LenzeMotionControl1.lib*

Funktionsblöcke für Positionieraufgaben

L_MCPositionCorrection - Vorschubkorrektur

Anzeige der Sollposition auf Bediengeräten

Für die Anzeige der Sollposition auf einem Bediengerät (z. B. Lenze H410) ist hierfür der Wert des Eingangs *dnSetPos_p* zu verwenden. Der Ausgang *dnSetPosCorrected_p* liefert je nach Status des FB den korrigierten Wert, der von der eigentlichen Sollposition abweichen kann.



2.4 L_MCPProfileGenerator - Profilgenerator

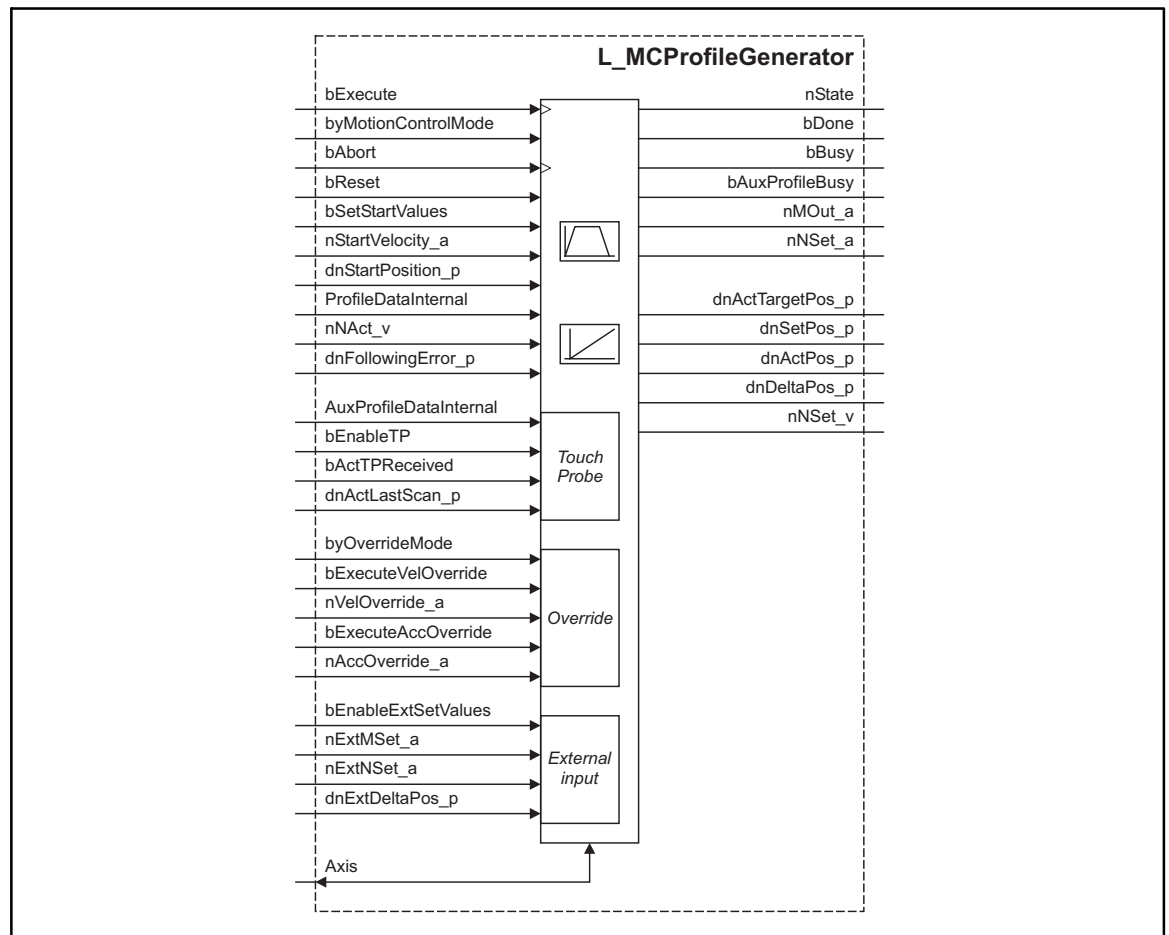
Mit diesem FB können Sie über ein lineares Geschwindigkeitsprofil einen bestimmten Wegpunkt anfahren.

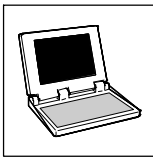


Hinweis!

Der FB **L_MCPProfileGenerator** muss auf Grund seiner internen Struktur in einer zeitäquidistanten Task aufgerufen werden (z. B. in einer 5 ms-Task).

Beachten Sie: Die zyklische Task PLC_PRG ist nicht zeitäquidistant!



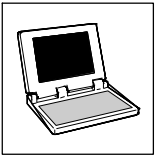


Funktionsbibliothek LenzeMotionControl1.lib

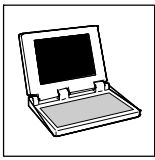
Funktionsblöcke für Positionieraufgaben

L_MCPProfileGenerator - Profilgenerator

⇒ Eingänge (Variablentyp: VAR_INPUT)																							
Bezeichner	Datentyp	Signaltyp	Info/Einstellmöglichkeiten																				
bExecute	Bool	binary	Positionierung ausführen Soll eine neue Positionierung gestartet werden, obwohl die neue Position gleich der Startposition ist, so wird die Positionierung nicht ausgeführt. <i>bDone</i> und <i>bBusy</i> werden kurzzeitig auf TRUE gesetzt. <table><tr><td>FALSE ➞ TRUE</td><td>Löst Positionierung oder Konstantfahrt aus.</td></tr><tr><td>FALSE ➞ TRUE (erneut)</td><td>Die aktuelle Positionierung wird abgebrochen und die neue Positionierung mit den neuen Positionsdaten gestartet.</td></tr></table>	FALSE ➞ TRUE	Löst Positionierung oder Konstantfahrt aus.	FALSE ➞ TRUE (erneut)	Die aktuelle Positionierung wird abgebrochen und die neue Positionierung mit den neuen Positionsdaten gestartet.																
FALSE ➞ TRUE	Löst Positionierung oder Konstantfahrt aus.																						
FALSE ➞ TRUE (erneut)	Die aktuelle Positionierung wird abgebrochen und die neue Positionierung mit den neuen Positionsdaten gestartet.																						
byMotionControlMode	Byte	-	Positioniermodus <table><tr><td>0</td><td>Absolute Positionierung</td></tr><tr><td>1</td><td>Relative Positionierung - Interner Wegintegrator wird vor dem Start zurückgesetzt, externe Wegintegratoren laufen normal weiter. - Anwendung: Rundtisch </td></tr><tr><td>2</td><td>Konstantfahrt endlos</td></tr><tr><td>3</td><td>Relative Positionierung - max. 2³⁰ - Wegintegratoren werden vor dem Start nicht zurückgesetzt. </td></tr></table> Positioniermodi mit Touch-Probe (Restwegpositionierung): <table><tr><td>10</td><td>Absolute Positionierung mit relativer Restwegpositionierung</td></tr><tr><td>11</td><td>Relative Positionierung mit relativer Restwegpositionierung - Interner Wegintegrator wird vor dem Start zurückgesetzt, externe Wegintegratoren laufen normal weiter. - Anwendung: Rundtisch </td></tr><tr><td>12</td><td>Konstantfahrt endlos mit relativer Restwegpositionierung</td></tr><tr><td>13</td><td>Relative Positionierung mit relativer Restwegpositionierung - max. 2³⁰ - Wegintegratoren werden vor dem Start nicht zurückgesetzt. </td></tr><tr><td>14</td><td>Absolute Positionierung mit absoluter Restwegpositionierung</td></tr><tr><td>15</td><td>Relative Positionierung mit absoluter Restwegpositionierung - Wegintegratoren werden vor dem Start nicht zurückgesetzt. </td></tr></table>	0	Absolute Positionierung	1	Relative Positionierung - Interner Wegintegrator wird vor dem Start zurückgesetzt, externe Wegintegratoren laufen normal weiter. - Anwendung: Rundtisch	2	Konstantfahrt endlos	3	Relative Positionierung - max. 2³⁰ - Wegintegratoren werden vor dem Start nicht zurückgesetzt.	10	Absolute Positionierung mit relativer Restwegpositionierung	11	Relative Positionierung mit relativer Restwegpositionierung - Interner Wegintegrator wird vor dem Start zurückgesetzt, externe Wegintegratoren laufen normal weiter. - Anwendung: Rundtisch	12	Konstantfahrt endlos mit relativer Restwegpositionierung	13	Relative Positionierung mit relativer Restwegpositionierung - max. 2³⁰ - Wegintegratoren werden vor dem Start nicht zurückgesetzt.	14	Absolute Positionierung mit absoluter Restwegpositionierung	15	Relative Positionierung mit absoluter Restwegpositionierung Wegintegratoren werden vor dem Start nicht zurückgesetzt.
0	Absolute Positionierung																						
1	Relative Positionierung - Interner Wegintegrator wird vor dem Start zurückgesetzt, externe Wegintegratoren laufen normal weiter. - Anwendung: Rundtisch																						
2	Konstantfahrt endlos																						
3	Relative Positionierung - max. 2³⁰ - Wegintegratoren werden vor dem Start nicht zurückgesetzt.																						
10	Absolute Positionierung mit relativer Restwegpositionierung																						
11	Relative Positionierung mit relativer Restwegpositionierung - Interner Wegintegrator wird vor dem Start zurückgesetzt, externe Wegintegratoren laufen normal weiter. - Anwendung: Rundtisch																						
12	Konstantfahrt endlos mit relativer Restwegpositionierung																						
13	Relative Positionierung mit relativer Restwegpositionierung - max. 2³⁰ - Wegintegratoren werden vor dem Start nicht zurückgesetzt.																						
14	Absolute Positionierung mit absoluter Restwegpositionierung																						
15	Relative Positionierung mit absoluter Restwegpositionierung Wegintegratoren werden vor dem Start nicht zurückgesetzt.																						
bAbort	Bool	binary	Positionierung abbrechen Dieser Eingang hat eine höhere Priorität als <i>bExecute</i>. <table><tr><td>FALSE ➞ TRUE</td><td>Löst Abbruch der Positionierung oder der Konstantfahrt aus, d. h. es wird über die Verzögerungsrampe des Profils auf Geschwindigkeit = 0 heruntergefahren.</td></tr></table>	FALSE ➞ TRUE	Löst Abbruch der Positionierung oder der Konstantfahrt aus, d. h. es wird über die Verzögerungsrampe des Profils auf Geschwindigkeit = 0 heruntergefahren.																		
FALSE ➞ TRUE	Löst Abbruch der Positionierung oder der Konstantfahrt aus, d. h. es wird über die Verzögerungsrampe des Profils auf Geschwindigkeit = 0 heruntergefahren.																						
bReset	Bool	binary	Positionierung zurücksetzen - Dieser Eingang hat eine höhere Priorität als <i>bExecute</i>, <i>bAbort</i> und <i>bSetStartValues</i>. <table><tr><td>TRUE</td><td> - Der Sollwinkelintegrator wird mit dem Istwinkelintegrator gleichgesetzt. - Die Positionierung wird gestoppt. - Die Ausgangsgeschwindigkeit <i>nNset_a</i> wird sofort auf "0" gesetzt. Achtung: Hierdurch entsteht ein Drehzahlsprung, der bei aktiver Motorregelung zu starken Momentenstößen führen kann! Empfehlung: Vor dem Rücksetzen der Positionierung Quick-stop (QSP) setzen. </td></tr></table>	TRUE	- Der Sollwinkelintegrator wird mit dem Istwinkelintegrator gleichgesetzt. - Die Positionierung wird gestoppt. - Die Ausgangsgeschwindigkeit <i>nNset_a</i> wird sofort auf "0" gesetzt. Achtung: Hierdurch entsteht ein Drehzahlsprung, der bei aktiver Motorregelung zu starken Momentenstößen führen kann! Empfehlung: Vor dem Rücksetzen der Positionierung Quick-stop (QSP) setzen.																		
TRUE	- Der Sollwinkelintegrator wird mit dem Istwinkelintegrator gleichgesetzt. - Die Positionierung wird gestoppt. - Die Ausgangsgeschwindigkeit <i>nNset_a</i> wird sofort auf "0" gesetzt. Achtung: Hierdurch entsteht ein Drehzahlsprung, der bei aktiver Motorregelung zu starken Momentenstößen führen kann! Empfehlung: Vor dem Rücksetzen der Positionierung Quick-stop (QSP) setzen.																						



⇒ Eingänge (Variablentyp: VAR_INPUT)			
Bezeichner	Datentyp	Signaltyp	Info/Einstellmöglichkeiten
bSetStartValues	Bool	binary	Startwerte für Positionierung laden
			FALSE/TRUE - Dieser Eingang hat eine höhere Priorität als <i>bExecute</i> und <i>bAbort</i>. - Der Profilgenerator wird mit der Startgeschwindigkeit <i>nStartVelocity_a</i> geladen. Achtung: Hierdurch entsteht ein Drehzahlsprung, der bei aktiver Motorregelung zu starken Momentenstößen führen kann! Empfehlung: Vor dem Laden der Startwerte Quickstop (QSP) setzen. - Eine gerade laufende Positionierung wird sofort abgebrochen. - Soll- und Istwinkelintegrator werden mit dem Wert <i>dnStartPosition_p</i> geladen.
			TRUE/FALSE Der am Eingang <i>dnFollowingError</i> anliegende Schleppfehler wird vom Istwertintegrator abgezogen
nStartVelocity_a	Integer	analog	Startgeschwindigkeit - Wert wird übernommen, wenn <i>bSetStartValues</i> = TRUE gesetzt wird. - Der Wert 16384 entspricht der Maximaldrehzahl aus C0011.
dnStartPosition_p	Double Integer	position	Startposition Wert wird übernommen, wenn <i>bSetStartValues</i> = TRUE gesetzt wird.
ProfileDataInternal	MC_PROFILE_INTERNAL_MEASURING	-	Positionierprofil Datenstruktur, deren Elemente das gesamte Profil beschreiben.
nNact_v	Integer	velocity	Istdrehzahl für den Istwinkelintegrator In der Regel ist dieser Eingang mit der Systemvariable <i>MCTRL_nNact_v</i> (Drehzahlwert des Rückführsystems) des SB MCTRL zu verbinden.
dnFollowingError_p	Double Integer	position	Schleppfehler Wert wird nach TRUE/FALSE-Flanke am Eingang <i>bSetStartValues</i> vom Istwert-Integrator abgezogen.
AuxProfileDataInternal	MC_PROFILE_INTERNAL_MEASURING	-	Zusätzliches Positionierprofil für die Restwegpositionierung Datenstruktur, deren Elemente das gesamte Profil beschreiben.
bEnableTP	Bool	binary	Touch-Probe Signalverarbeitung aktivieren TRUE Touch-Probe Signalverarbeitung ist aktiv.
bActTPReceived	Bool	binary	Eingang für Touch-Probe TRUE Touch-Probe Signal wurde erkannt.
dnActLastScan_p	Double Integer	position	Winkeldifferenz in [inc] zwischen Latchzeitpunkt und Startzeitpunkt der Task
byOverrideMode	Byte	-	Festlegung des 100 %-Bezugs für Override-Parameter
			0 16384 ≡ 100 % ≡ Vel_{max}/Acc_{max} aus den Achsparametern
			1 16384 ≡ 100 % ≡ Vel/Acc aus den Profilparametern
bExecuteVelOverride	Bool	binary	Geschwindigkeitsoverride aktivieren TRUE Geschwindigkeitsoverride ist aktiv.
nVelOverride_a	Integer	velocity	Geschwindigkeit für Override 16384 ≡ v_{max}
bExecuteAccOverride	Bool	binary	Beschleunigungsoverride aktivieren TRUE Beschleunigungsoverride ist aktiv.
nAccOverride_a	Integer	velocity	Beschleunigung für Override 16384 ≡ a_{max}
bEnableExtSetValues	Bool	binary	Umschalten auf externe Sollwerte TRUE Die externen Sollwerte werden 1:1 auf die entsprechenden Ausgänge durchgeschliffen: <i>nExtMSet_a</i> ⇒ <i>nMSet_a</i> <i>nExtNSet_a</i> ⇒ <i>nNSet_a</i> <i>dn_ExtDeltaPos_p</i> ⇒ <i>dnDeltaPos_p</i>
nExtMSet_a	Integer	analog	Externer Sollwert für Drehmomentenvorsteuerung
nExtNSet_a	Integer	analog	Externer Sollwert für Solldrehzahl
dnExtDeltaPos_p	Integer	position	Externer Sollwert für Lagedifferenz



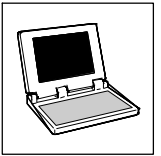
Funktionsbibliothek LenzeMotionControl1.lib

Funktionsblöcke für Positionieraufgaben

L_MCPProfileGenerator - Profilgenerator

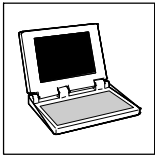
⇒ Ein-/Ausgänge ⇒ (Variablentyp: VAR_IN_OUT)			
Bezeichner	Datentyp	Signaltyp	Info/Einstellmöglichkeiten
Axis	AXIS_REF		Maschinenparameter • Datenstruktur, deren Elemente die Maschinenparameter im internen Maßsystem beinhalten. • Verbinden Sie diesen Eingang mit einer globalen Variable, die vom FB L_MCMachineData beschrieben wird.

Ausgänge ⇒ (Variablentyp: VAR_OUTPUT)			
Bezeichner	Datentyp	Signaltyp	Info
nState	Integer		Zeigt den Status der Positionierung an.
			Meldungen:
			0 Keine Aktion
			1 Positionierung abgeschlossen
			10 Beschleunigungsphase ist aktiv
			11 Konstantgeschwindigkeitsphase ist aktiv
			12 Verzögerungsphase ist aktiv
			13 Reversierphase ist aktiv
			100 Reset ist aktiv (<i>bReset</i>)
			101 Startwerte werden geladen
			102 Positionierung abgebrochen (<i>bAbort</i>)
			103 Externe Sollwertschnittstelle aktiviert
			200 Positionierung mit Touch-Probe (Restwegpositionierung) abgeschlossen, aber es wurde kein Touch-Probe-Signal erkannt.
			300 Überlauf des internen Wegintegrators
			Fehler:
			-200 Profil <i>ProfileDataInternal</i> ist ungültig: Parameter Beschleunigung = 0
			-201 Profil <i>ProfileDataInternal</i> ist ungültig: Parameter Verzögerung = 0
			-202 Profil <i>ProfileDataInternal</i> ist ungültig: Parameter Geschwindigkeit = 0
			-210 Profil <i>AuxProfileDataInternal</i> ist ungültig: Parameter Beschleunigung = 0
			-211 Profil <i>AuxProfileDataInternal</i> ist ungültig: Parameter Verzögerung = 0
			-212 Profil <i>AuxProfileDataInternal</i> ist ungültig: Parameter Geschwindigkeit = 0
			-300 Positionierung mit eingestellten Profilparametern ist nicht plausibel.
bDone	Bool	binary	Meldesignal "Positionierung wurde ausgeführt"
			TRUE Die Positionierung wurde ausgeführt.
			TRUE (Impuls) <i>bDone</i> wird beim Profilwechsel im Modus Overchange für einen Aufrufzyklus auf TRUE gesetzt.
bBusy	Bool	binary	Meldesignal "Positionierung ist aktiv"
			TRUE Die Positionierung ist aktiv.
			TRUE (Impuls) <i>bBusy</i> wird für einen Aufrufzyklus auf TRUE gesetzt, wenn beim Start einer absoluten Positionierung Ziel und Startposition gleich sind und daher die Positionierung nicht gestartet wurde.
bAuxProfileBusy	Bool	binary	Meldesignal "Restwegpositionierung ist aktiv"
			TRUE Die Restwegpositionierung ist aktiv.



Ausgänge ⇒ (Variablentyp: VAR_OUTPUT)			
Bezeichner	Datentyp	Signaltyp	Info
nMOut_a	Integer	analog	Ausgangssignal für Drehmomentenvorsteuerung
nNSet_a	Integer	analog	Solldrehzahl für den Drehzahlregler - Der Wert 16384 entspricht der Maximaldrehzahl aus C0011. - In der Regel ist dieser Ausgang mit der Systemvariable <i>MCTRL_nNSet_a</i> des SB MCTRL zu verbinden.
dnActTargetPos_p	Double Integer	position	Zielwinkelposition des aktuellen Profils 65536 entspricht einer Umdrehung auf Motorseite 0 bei externer Sollwertvorgabe (<i>bEnableExtSetValues</i> = TRUE) - Zeigt je nach Positioniermodus entweder relativen Verfahrweg oder absolute Zielposition an.
dnSetPos_p	Double Integer	position	Sollwinkelposition 65536 entspricht einer Umdrehung auf Motorseite - Der Sollintegrator kann überlaufen.
dnActPos_p	Double Integer	position	Istwinkelposition 65536 entspricht einer Umdrehung auf Motorseite - Der Istintegrator kann überlaufen.
dnDeltaPos_p	Double Integer	position	Lagedifferenzausgang direkt für den Lageregler. In der Regel ist dieser Ausgang mit der Systemvariable <i>MCTRL_dnPosSet_p</i> des SB MCTRL zu verbinden, wobei der Winkelregler aktiv geschaltet werden muss (<i>MCTRL_bPosOn_b</i> = TRUE).
nNSet_v	Integer	velocity	Drehzahlsollwert in Inkrementen pro Millisekunde 16384 entspricht 15000 rpm - Nur bei einem motorseitigen Gebersystem verwenden! - Bei Verwendung eines lastseitigen Gebersystems (Getriebefaktor ungleich 1) wird Drehzahlsollwert 0 ausgegeben

Parameter				
Bezeichner	Datentyp	Signaltyp	Info/Einstellmöglichkeiten	
wSProfileMode	Word		Modus S-Generierung	
			0	Laufzeitoptimiert (Werkseinstellung)
			1	S-Profiloptimiert erfordert deutlich höhere Bearbeitungszeit
wDecOverride	Word		Aktivierung Verzögerungsoverride	
			0	Verzögerungsoverride deaktiviert
			1	Die Overrideverzögerung (Wert am Eingang <i>nAccover-ride_a</i> bezogen auf die maximale Verzögerung) wird einmalig beim Profilstart übernommen.

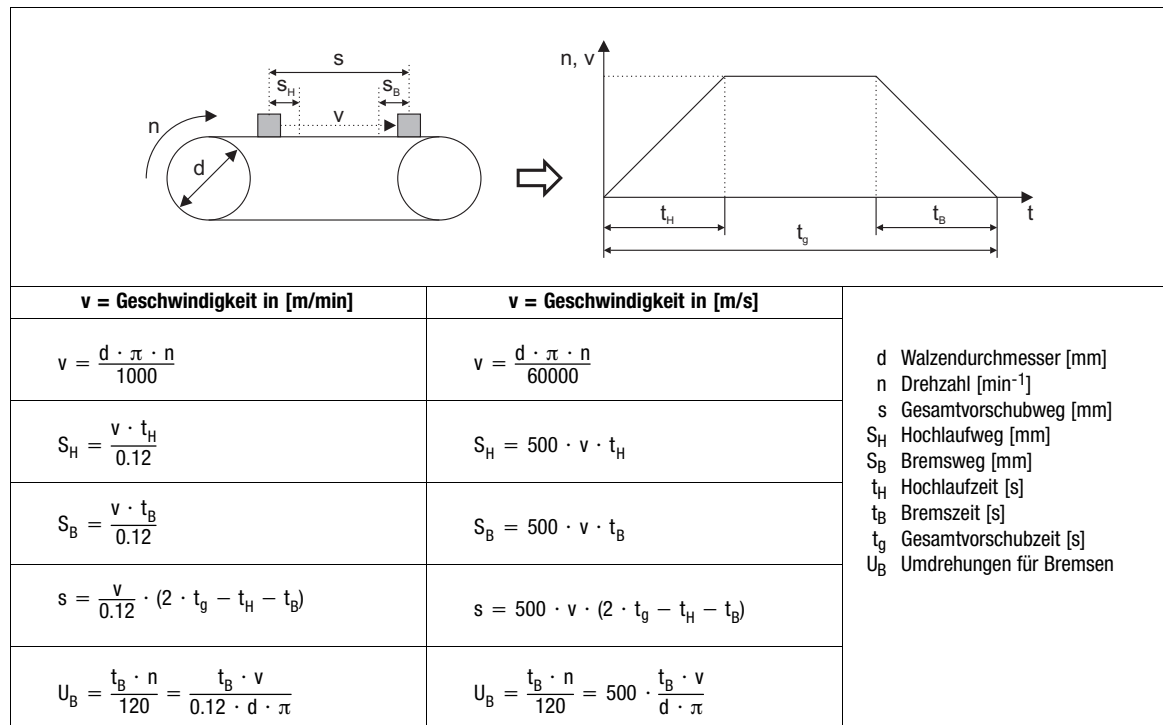


Funktionsbibliothek LenzeMotionControl1.lib

Funktionsblöcke für Positionieraufgaben

L_MCProfileGenerator - Profilgenerator

Formeln zum Positionieren



Übergabe der Maschinenparameter

Die Maschinendaten werden dem Profilgenerator über den Ein-/Ausgang *axis* übergeben.

- Verbinden Sie den Eingang *axis* mit einer globalen Variable, die vom FB **L_MCMachineData** beschrieben wird. (Kap. 2-4)

Vorgabe des Positionierprofils

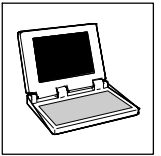
Das Positionierprofil wird dem Profilgenerator über den Eingang *ProfileDataInternal* übergeben.

- Der FB erwartet eine Datenstruktur (**MC_PROFILE_INTERNAL_MEASURING**), deren Elemente die Positionierdaten gemäß dem internen Maßsystem des Profilgenerators enthalten.



Tipp!

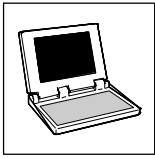
- Informationen zu den Funktionsblöcken, mit denen physikalische Einheiten des Realmaßsystems für das interne Maßsystem des Profilgenerators aufbereitet werden können, finden Sie im Kap. 3, "Funktionsblöcke zur Umrechnung von Einheiten". (Kap. 3-1)
- Informationen zu den verwendeten Datenstrukturen für die Maschinenparameter sowie Positionierprofilaten finden Sie im Kap. 4, "Strukturierte Datentypen". (Kap. 4-1)



Standard-Modi des Profilgenerators

Der Profilgenerator besitzt folgende vier Standard-Modi zum Positionieren:

Modus	Info	Hinweise
0 Absolute Positionierung	Dieser Modus setzt einen absoluten Nullpunkt voraus, der üblicherweise durch einen Referenziervorgang ermittelt und in den Lage-Istwertintegrator des Profilgenerators geladen wird. Alle Profile beziehen sich auf diesen Nullpunkt.	
1 Relative Positionierung	In diesem Modus wird keine absolute Position vorgegeben, sondern stattdessen eine Wegdistanz angefahren.	Die internen Wegintegratoren werden vor jeder Fahrt auf 0 gesetzt. • Eine darauffolgende absolute Positionierung ist nicht mehr möglich. Beispiel: Wird nach einer absoluten Positionierung im Modus 0 eine relative Positionierung im Modus 1 ausgeführt, so würde die Achse nach Zurückschalten in den Modus 0 die 0-Position verlieren. Die Achse verfährt dann mit Bezug auf die letzte Startposition im Modus 1.
2 Konstantfahrt	In diesem Modus wird keine Position angefahren, sondern eine Fahrt mit der Verfahrgeschwindigkeit durchgeführt, die über das Profil vorgegeben wird.	• Für das Beschleunigen und Abbremsen werden die Werte des Profils verwendet. • Die Fahrtrichtung ergibt sich aus dem Vorzeichen der Verfahrgeschwindigkeit. • Die Fahrt wird mit einer FALSE-TRUE-Flanke an <i>bExecute</i> gestartet. • Die Fahrt kann mit einer FALSE-TRUE-Flanke an <i>bAbort</i> abgebrochen werden.
3 Relative Positionierung (max. 2 ³⁰)	Wie Modus 1, "Relative Positionierung", wobei die internen Wegintegratoren nicht zurückgesetzt, sondern weiter inkrementiert werden.	Die Wegintegratoren werden auf 2 ³⁰ begrenzt.



Funktionsbibliothek LenzeMotionControl1.lib

Funktionsblöcke für Positionieraufgaben

L_MCProfileGenerator - Profilgenerator

Restwegpositionierung

Bei einer Restwegpositionierung wird wie auch bei einer Standardpositionierung zuerst das Standardprofil im entsprechenden Modus abgefahren.

Wird während des Verfahrens ein Touch-Probe erkannt, so erfolgt ein automatischer Wechsel auf das an *AuxProfileDataInternal* anliegende Profil, mit dessen Parametern der Profilgenerator daraufhin die Positionierung im entsprechenden Modus fortsetzt.

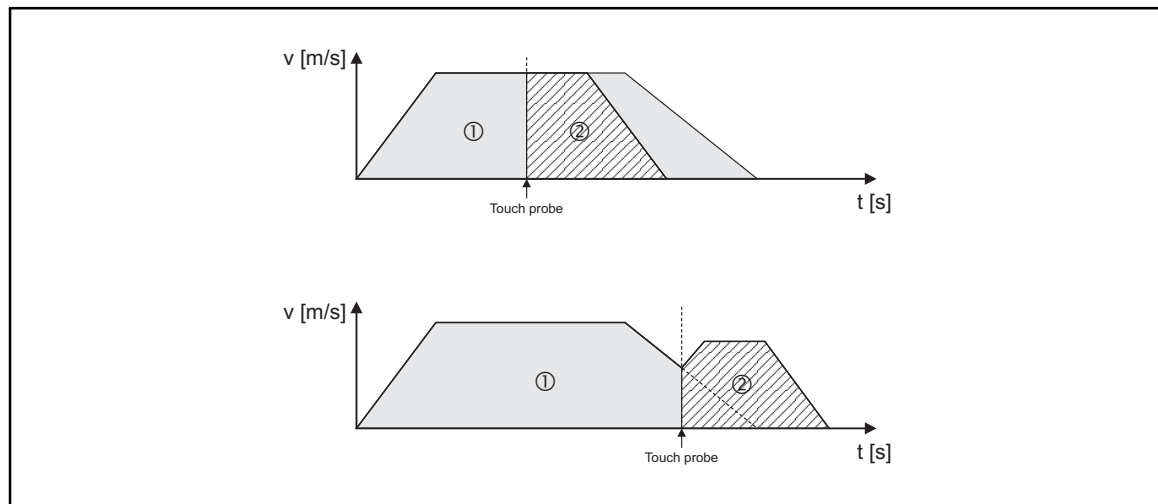
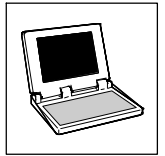


Abb. 2-4

Beispiele für eine Restwegpositionierung nach Erkennung eines Touch-Probe

- ① Standardprofil (*ProfileDataInternal*)
- ② Zusätzliches Profil für Restwegpositionierung (*AuxProfileDataInternal*)

- Eine aktive Restwegpositionierung wird durch den Zustand TRUE am Ausgang *bAuxProfileBusy* angezeigt.
- Während der Restwegpositionierung führt ein weiterer Touch-Probe am Eingang *bActTpReceived* zu einem Neustart, d. h. das zweite Profil für die Restwegpositionierung wird ausgehend von der aktuellen Position erneut abgefahren. Voraussetzung hierfür ist, dass der entsprechende Positioniermodus noch ausgewählt ist und das Profil für die Restwegpositionierung noch am FB anliegt.
- Über eine FALSE-TRUE-Flanke am Eingang *bExecute* kann eine aktive Restwegpositionierung abgebrochen und in einen anderen Positioniermodus gewechselt werden.



Modi des Profilgenerators für eine Restwegpositionierung

Der Profilgenerator besitzt folgende Modi für eine Restwegpositionierung mit Touch-Probe:

Modus	Info
10 Absolute Positionierung mit relativer Restwegpositionierung	Siehe auch Modus 0, "Absolute Positionierung". • Wird ein Touch-Probe erkannt, schaltet der Profilgenerator automatisch auf das zusätzliche Profil <i>AuxProfileDataInternal</i> um und führt eine relative Positionierung mit den Parametern dieses Profils durch.
11 Relative Positionierung mit relativer Restwegpositionierung	Siehe auch Modus 1, "Relative Positionierung". • Wird ein Touch-Probe erkannt, schaltet der Profilgenerator automatisch auf das zusätzliche Profil <i>AuxProfileDataInternal</i> um und führt eine relative Positionierung mit den Parametern dieses Profils durch.
12 Konstantfahrt mit relativer Restwegpositionierung	Siehe auch Modus 2, "Konstantfahrt". • Wird ein Touch-Probe erkannt, schaltet der Profilgenerator automatisch auf das zusätzliche Profil <i>AuxProfileDataInternal</i> um und führt eine relative Positionierung mit den Parametern dieses Profils durch.
13 Relative Positionierung mit relativer Restwegpositionierung	Siehe auch Modus 3, "Relative Positionierung". • Wird ein Touch-Probe erkannt, schaltet der Profilgenerator automatisch auf das zusätzliche Profil <i>AuxProfileDataInternal</i> um und führt eine relative Positionierung mit den Parametern dieses Profils durch. • Die internen Wegintegratoren werden in diesem Modus nicht zurückgesetzt, sondern weiter inkrementiert (interne Begrenzung auf 2^{30}).
14 Absolute Positionierung mit absoluter Restwegpositionierung	Siehe auch Modus 0, "Absolute Positionierung". • Wird ein Touch-Probe erkannt, schaltet der Profilgenerator automatisch auf das zusätzliche Profil <i>AuxProfileDataInternal</i> um und führt eine absolute Positionierung mit den Parametern dieses Profils durch.
15 Relative Positionierung mit absoluter Restwegpositionierung	Siehe auch Modus 3, "Relative Positionierung". • Wird ein Touch-Probe erkannt, schaltet der Profilgenerator automatisch auf das zusätzliche Profil <i>AuxProfileDataInternal</i> um und führt eine absolute Positionierung mit den Parametern dieses Profils durch. • Die internen Wegintegratoren werden in diesem Modus nicht zurückgesetzt, sondern weiter inkrementiert (interne Begrenzung auf 2^{30}).



Tipp!

Zur Vorverarbeitung von Touch-Probe-Signalen können Sie den FB **L_MCTouchProbeCtrl** verwenden. Dieser FB ermöglicht die Verwaltung von bis zu fünf Touch-Probe-Signalen sowie die Definition von Wegfenstern, um Touch-Probe-Signale in Abhängigkeit von der aktuellen Position auszublenzen.

Profiländerung während der Positionierung (Override)

Unter "Override" versteht man das Ändern von Profilparametern und deren Übernahme während des Positioniervorgangs.

Durch einen Override erhält das Verfahrprofil eine andere Geometrie, damit z. B. bei einer Geschwindigkeitsveränderung während der Positionierung dennoch exakt auf die definierte Zielposition positioniert wird:

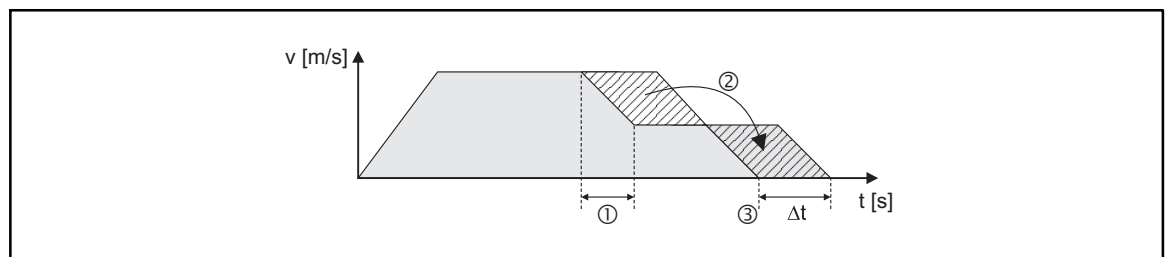
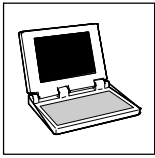


Abb. 2-5

Override (hier: Geschwindigkeitsoverride)

- ① Geschwindigkeit wird während des Positioniervorgangs verringert.
- ② Um die definierte Position zu erreichen, wird die fehlende Fläche automatisch dem Profil "angehängt".
- ③ Durch die Verringerung der Geschwindigkeit dauert der Positioniervorgang entsprechend länger (Δt).



Funktionsbibliothek *LenzeMotionControl1.lib*

Funktionsblöcke für Positionieraufgaben

L_MCProfileGenerator - Profilgenerator

Bezugsgröße für Override festlegen

Den Bezug für die Override-Werte legen Sie über *g_byOverrideMode* fest:

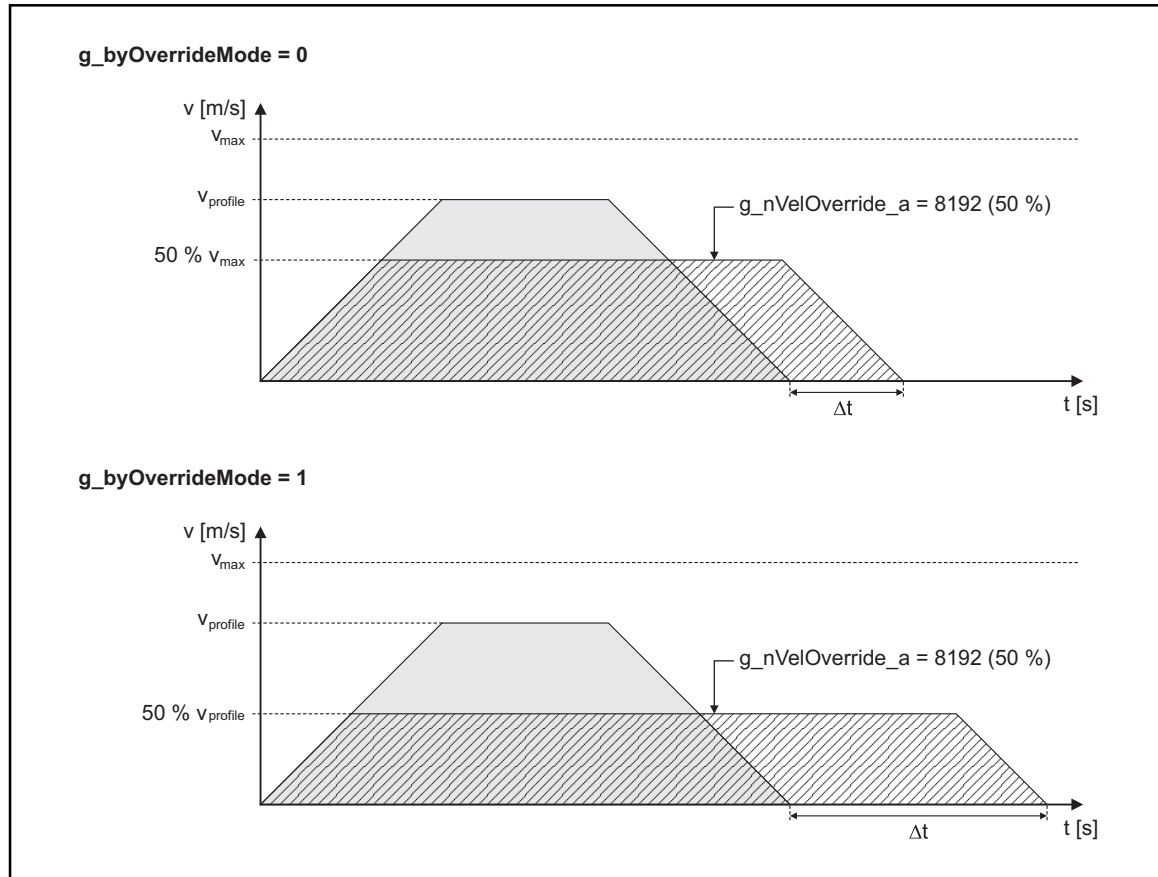


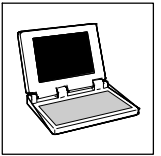
Abb. 2-6

Definition der Bezugsgröße für Override (hier: Geschwindigkeitsoverride)



Tipp!

Den Override-Modus "1" können Sie dazu verwenden, z. B. zwecks Einrichtbetrieb die Taktrate des Werkzeuges zu verringern, da in diesem Modus die Kontur der Profile erhalten bleibt und sich nur die Abarbeitungsdauer verlängert.



Override durchführen

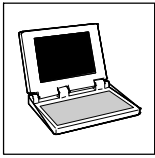
Der FB **L_MCProfileGenerator** ermöglicht die Veränderung der Geschwindigkeit ("Geschwindigkeitsoverride") sowie der Beschleunigung ("Beschleunigungsoverride").

- Der Geschwindigkeitsoverride wird durch Setzen von *bExecuteVelOverride* auf TRUE aktiviert.
- Der Beschleunigungsoverride wird durch Setzen von *bExecuteAccOverride* auf TRUE aktiviert.
- Die Änderung der Geschwindigkeit wird über den Eingang *nVelOverride_a* vorgegeben, die Änderung der Beschleunigung über den Eingang *nAccOverride_a*:

Bezeichner	Datentyp	Info/Einstellmöglichkeiten	
<i>nVelOverride_a</i>	Integer	1 ... 16384	Geschwindigkeit (bezogen auf v_{max}) • $16384 \equiv v_{max}$ • Eine Änderung der Geschwindigkeit ist nur in der Beschleunigungsphase und bei Konstantfahrt (bei Trapez) wirksam. • Wird während einer Konstantfahrt ein Geschwindigkeitsoverride durchgeführt, folgt eine Verzögerungs- bzw. Beschleunigungsphase bis zum Erreichen der neuen Geschwindigkeit. Die Verzögerung/Beschleunigung erfolgt mit den Parametern des aktuellen Profils. Ist zusätzlich ein Beschleunigungsoverride aktiviert, erfolgt die Beschleunigung mit dem geänderten Beschleunigungswert.
<i>nAccOverride_a</i>	Integer	1 ... 16384	Beschleunigung (bezogen auf a_{max}) • $16384 \equiv a_{max}$ • Eine Änderung der Beschleunigung ist nur in der Beschleunigungsphase wirksam.

Über den Parameter *wDecoverride* kann Verzögerungsoverride aktiviert werden. Beim aktivierten Verzögerungsoverride (*wDecoverride* = 1) wird die Overrideverzögerung (Wert am Eingang *nAccoverride_a* bezogen auf die maximale Verzögerung) einmalig beim Profilstart übernommen. Im weiteren Verlauf des Positioniervorgangs wird die Profilverzögerung nicht mehr verändert.

Parameter							
Bezeichner	Datentyp	Signaltyp	Info/Einstellmöglichkeiten				
wDecOverride	Word		Aktivierung Verzögerungsoverride <table><tr><td>0</td><td>Verzögerungsoverride deaktiviert</td></tr><tr><td>1</td><td>Die Profilverzögerung wird einmalig beim Profilstart auf Verzögerungsoverridewert (Wert vom Eingang <i>nAccoverride_a</i> bezogen auf die maximale Verzögerung) gesetzt.</td></tr></table> Festlegung des 100%-Bezugs für Verzögerungsoverride-Parameter: byOverrideMode = 0: 16384 ≡ 100 % ≡ Accmax aus den Achsparametern byOverrideMode = 1: 16384 ≡ 100 % ≡ Dec aus den Profilparametern	0	Verzögerungsoverride deaktiviert	1	Die Profilverzögerung wird einmalig beim Profilstart auf Verzögerungsoverridewert (Wert vom Eingang <i>nAccoverride_a</i> bezogen auf die maximale Verzögerung) gesetzt.
0	Verzögerungsoverride deaktiviert						
1	Die Profilverzögerung wird einmalig beim Profilstart auf Verzögerungsoverridewert (Wert vom Eingang <i>nAccoverride_a</i> bezogen auf die maximale Verzögerung) gesetzt.						



Funktionsbibliothek LenzeMotionControl1.lib

Funktionsblöcke für Positionieraufgaben

L_MCProfileGenerator - Profilgenerator



Tipp!

Ist ein Override aktiviert, wirken sich Änderungen an den Eingängen *nVelOverride_a* bzw. *nAccOverride_a* **unmittelbar** auf das Profil aus.

Nach der Aufhebung des Override durch Rücksetzen von *bExecuteVelOverride* auf FALSE bzw. *bExecuteAccOverride* auf FALSE wird das Profil mit dem aktuellen Geschwindigkeitswert zu Ende gefahren.

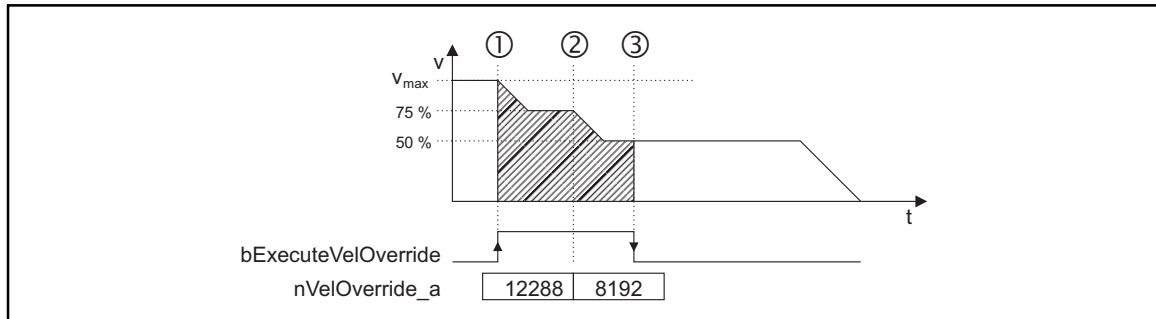
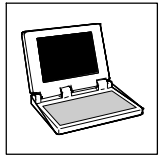


Abb. 2-7

Override (hier: Geschwindigkeitsoverride)

- ① Geschwindigkeitsoverride wird aktiviert (*bExecuteVelOverride* = TRUE);
Änderung auf 75 % von $v_{\max}$ (*nVelOverride_a* = 12288)
- ② Erneute Geschwindigkeitsänderung auf 50% von $v_{\max}$ (*nVelOverride_a* = 8192)
- ③ Geschwindigkeitsoverride wird aufgehoben (*bExecuteVelOverride* = FALSE)



Profilverkettung mit Overchange

Bei einer Profilverkettung mit Overchange wird von einem Verfahrprofil ① mit einer definierten Endgeschwindigkeit (ungleich "0") in ein anderes Verfahrprofil ② übergeleitet:

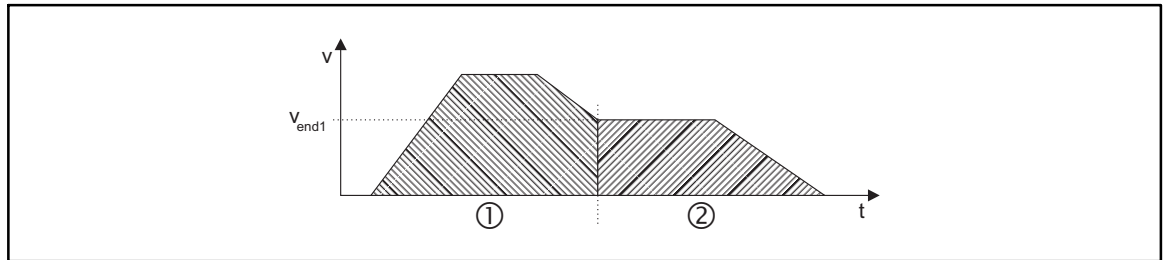
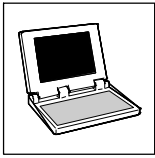


Abb. 2-8

Beispiel: Profilverkettung mit Overchange

- ① Erstes Verfahrprofil mit Endgeschwindigkeit v_{end1} ungleich "0".
- ② Zweites Verfahrprofil mit Startgeschwindigkeit = Endgeschwindigkeit des ersten Verfahrprofils.

- Die Endgeschwindigkeit v_{end} wird im Positionierprofil als v_{end} in [units] angegeben und beim Start des Positioniervorgangs vom Profilgenerator übernommen.
- Die Endgeschwindigkeit v_{end} wird auf den Wert der über die Profilparameter eingestellten Positioniergeschwindigkeit begrenzt.
- Ist der Positioniervorgang abgeschlossen und die vorgegebene Endgeschwindigkeit v_{end} erreicht, wird automatisch ein weiterer Positioniervorgang mit dem nun an *ProfileDataInternal* anliegenden Profil gestartet, ohne dass hierzu *bExecute* erneut gesetzt werden muss.
- Hat auch das Folgeprofil eine Endgeschwindigkeit ungleich "0", wird die Profilverkettung fortgesetzt, und zwar solange, bis ein Profil mit einer Endgeschwindigkeit gleich "0" abgefahren wird. Hiernach ist dann wieder ein Setzen von *bExecute* auf TRUE erforderlich, um einen weiteren Positioniervorgang zu starten.
- Liegt die Zielposition eines Profils in einem Bereich, der aufgrund der Endgeschwindigkeit des vorherigen Profils und der Verzögerung des aktuellen Profils nicht mehr erreicht werden kann, so kommt es zum Reversieren des Antriebs.
- Nach einer über Touch-Probe gestarteten Restwegpositionierung wird ebenfalls eine Profilverkettung mit dem an *ProfileDataInternal* anliegenden Profil durchgeführt, wenn die Endgeschwindigkeit der Restwegpositionierung (Profildaten an *AuxProfileDataInternal*) ungleich "0" ist.



Funktionsbibliothek LenzeMotionControl1.lib

Funktionsblöcke für Positionieraufgaben

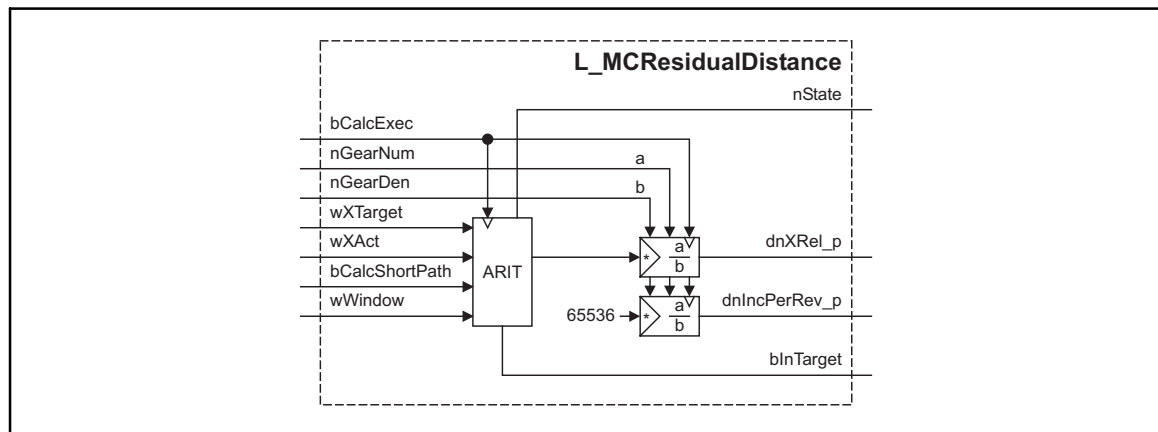
L_MCResidualDistance - Restwegberechnung

2.5

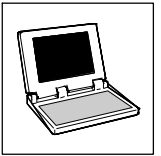
L_MCResidualDistance - Restwegberechnung

Dieser FB berechnet für Rundachsenanwendungen den zu fahrenden Restweg bis zu einer Zielposition.

- Die Berechnung des Restweges findet mit absoluten Werten statt, um eine optimale Auflösung für den Restweg zu erreichen. Hierfür sind die aktuelle Position sowie die Zielposition im Maschinenmaßsystem von 0 ... 65535 vorzugeben.
- Der Restweg wird mit dem Getriebefaktor bewertet als Distanz in Motorinkrementen ausgegeben.
- Zusätzlich wird die Anzahl der Inkremente ausgegeben, die einer Umdrehung im Zielmaßsystem entsprechen.



⇒ Eingänge (Variablentyp: VAR_INPUT)			
Bezeichner	Datentyp	Signaltyp	Info/Einstellmöglichkeiten
bCalcExec	Bool	binary	Berechnung starten
			FALSE FB deaktiviert (nState = 0)
			FALSE → TRUE Berechnung starten
			TRUE Prüfen, ob Ziel erreicht
nGearNum	Integer	-	Getriebefaktor Zähler
nGearDen	Integer	-	Getriebefaktor Nenner
wXTarget	Word	-	Positionsziel im Maschinenmaßsystem (0 ... 65535 [units])
wXAct	Word	-	Aktuelle Position im Maschinenmaßsystem (0 ... 65535 [units])
bCalcShortPath	Bool	binary	Berechnung kürzester Restweg
			TRUE Der FB berechnet immer den kürzesten Restweg, d. h. in diesem Modus sind auch negative Ergebnisse möglich.
wWindow	Word	-	Breite des Fensters für die Überprüfung "Zielposition erreicht"
			• Bedingt durch die symmetrische Anordnung des Fensters um die Zielposition werden ungerade Werte auf gerade Werte abgerundet.
			• Befindet sich die aktuelle Position innerhalb des Fensters, wird <i>bInTarget</i> auf TRUE gesetzt.



Ausgänge ⇨ (Variablentyp: VAR_OUTPUT)				
Bezeichner	Datentyp	Signaltyp	Info	
nState	Integer	-	Status	
			-10	Berechnung wurde nicht durchgeführt, da Getriebefaktoren- ner = 0 oder Getriebefaktorzähler = -32768 (<i>dnXRel_p</i> = 0; <i>dnIncPerRev_p</i> = 0; <i>blnTarget</i> = FALSE).
			0	Eingangsparameter ok, Berechnung wurde durchgeführt.
			10	FB ist nicht aktiv.
			20	Berechnung des kürzesten Restweges wurde durchgeführt.
dnXRel_p	Double Integer	position	Motorinkremente bis Positionsziel	
dnIncPerRev_p	Double Integer	position	Motorinkremente pro 1 Umdr. des Zielmaßsystems	
blnTarget	Bool	binary	Meldung "Zielposition erreicht"	
			TRUE	Die aktuelle Position befindet sich im über wWindow festgelegten Fenster.

Definition des Fensters für die Meldung "Zielposition erreicht"

Über *wWindow* können Sie ein Fenster festlegen, anhand dessen überprüft wird, ob die Zielposition erreicht ist.

- Befindet sich die aktuelle Position innerhalb dieses Fensters, so wird *blnTarget* auf TRUE gesetzt.
- Für die Überprüfung muss der Eingang *bCalcExec* nach der FALSE-TRUE-Flanke zum Starten der Berechnung auf TRUE gesetzt bleiben.
- Bedingt durch die symmetrische Anordnung des Fensters um die Zielposition werden ungerade Werte für *wWindow* auf gerade Werte abgerundet (Siehe ②).

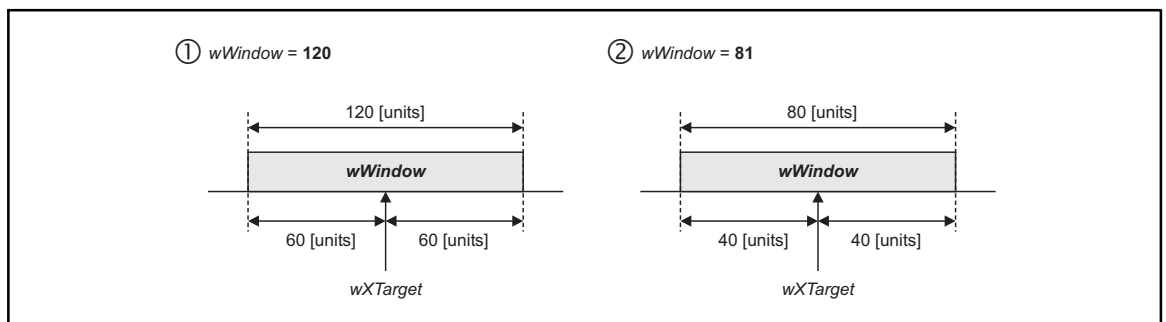


Abb. 2-9

Symmetrische Aufteilung von *wWindow* um die Zielposition *wXTarget*



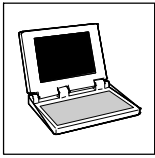
Hinweis!

Der Ausgang *blnTarget* arbeitet ohne Hysterese.

Berechnung des kürzesten Restweges

Ist *bCalcShortPath* auf TRUE gesetzt, wird immer der kürzeste Restweg berechnet.

- In diesem Fall kann die errechnete Distanz auch negative Werte annehmen.
- Beispiele hierzu finden Sie auf der folgenden Seite unter "Berechnung des Restweges".



Funktionsbibliothek LenzeMotionControl1.lib

Funktionsblöcke für Positionieraufgaben

L_MCResidualDistance - Restwegberechnung

Berechnung des Restweges

Eine FALSE-TRUE-Flanke an *bCalcExec* startet die Berechnung der Ausgangsgrößen wie folgt:

- Wenn $wXAct > wXTarget$:

<i>bCalcShortPath</i> = FALSE	$dnXRel_p = (65536 - wXAct + wXTarget) \cdot \frac{nGearNum}{nGearDen}$ $dnIncPerRev_p = 65536 \cdot \frac{nGearNum}{nGearDen}$
<i>bCalcShortPath</i> = TRUE	$dnXRel_p = - (wXAct - wXTarget) \cdot \frac{nGearNum}{nGearDen}$ $dnIncPerRev_p = 65536 \cdot \frac{nGearNum}{nGearDen}$

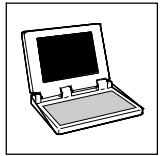
- Wenn $wXAct \leq wXTarget$:

<i>bCalcShortPath</i> = FALSE	$dnXRel_p = (wXTarget - wXAct) \cdot \frac{nGearNum}{nGearDen}$ $dnIncPerRev_p = 65536 \cdot \frac{nGearNum}{nGearDen}$
<i>bCalcShortPath</i> = TRUE	$dnXRel_p = - (65536 - wXTarget + wXAct) \cdot \frac{nGearNum}{nGearDen}$ $dnIncPerRev_p = 65536 \cdot \frac{nGearNum}{nGearDen}$



Hinweis!

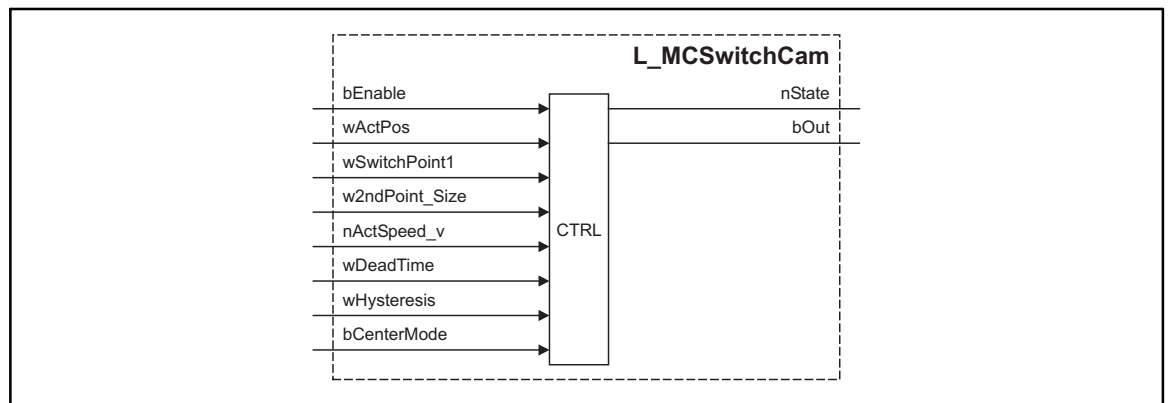
Dieser FB arbeitet ohne Restwertbehandlung.



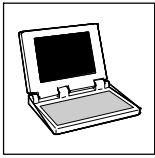
2.6 L_MCSwitchCam - Wegschaltpunkte (Rundachse)

Mit diesem FB können Sie einen Wegschaltpunkt (auch "Nocken" genannt) auf einer Rundachse realisieren, d. h. einen digitalen Schalter, dessen binärer Zustand (FALSE/TRUE) abhängig von der Istposition ist.

- Mit Hilfe eines Wegschaltpunktes können z. B. Peripheriegeräte wie Lackierdüsen oder Messer in Abhängigkeit der Werkzeugposition gestartet werden.
- Der FB unterstützt zudem die Kompensation von Verzögerungszeiten externer Schaltelemente (Totzeitkompensation).



⇒ Eingänge (Variablentyp: VAR_INPUT)			
Bezeichner	Datentyp	Signaltyp	Info/Einstellmöglichkeiten
bEnable	Bool	binary	Funktionsblock aktivieren TRUE Funktionsblock aktiviert.
wActPos	Word	position	Istposition in [unit] • Eingabe im Maschinenmaßsystem: 0 ... 65535 inc = 0° ... 360° der Rundachse
wSwitchPoint1	Word	position	Position des ersten Schaltpunktes in [unit] • Eingabe im Maschinenmaßsystem: 0 ... 65535 inc = 0° ... 360° der Rundachse
w2ndPoint_Size	Word	position	Bei <i>bCenterMode</i> = FALSE: Position des zweiten Schaltpunktes in [unit] Bei <i>bCenterMode</i> = TRUE: Größe des Schaltfensters in [unit] • Bedingt durch die symmetrische Anordnung des Fensters um den ersten Schalterpunkt werden ungerade Werte auf gerade Werte abgerundet.
nActSpeed_v	Integer	velocity	Istgeschwindigkeit in [inc/ms] • 16384 = 15000 inc/ms
wDeadTime	Word	-	Totzeit in [µs] für Totzeitkompensation • Vorgabe in 10 µs-Schritten • Die sich hieraus ergebende Schaltposition unterliegt keiner Plausibilitätsprüfung.
wHysteresis	Word	-	Schalthysterese in [unit] • Eingabe im Maschinenmaßsystem: 0 ... 16000 inc (Begrenzung auf 16000 inc)
bCenterMode	Bool	binary	Festlegung, wie die Vorgabe <i>w2ndPoint_Size_p</i> interpretiert wird: FALSE <i>w2ndPoint_Size</i> definiert den zweiten Schalterpunkt. TRUE <i>w2ndPoint_Size</i> definiert die Größe des Schaltfensters.



Funktionsbibliothek LenzeMotionControl1.lib

Funktionsblöcke für Positionieraufgaben

L_MGSwitchCam - Wegschaltpunkte (Rundachse)

Ausgänge ⇒ (Variablentyp: VAR_OUTPUT)			
Bezeichner	Datentyp	Signaltyp	Info
nState	Integer	binary	Status
			10 FB ist nicht aktiv
			0 OK
			-10 Die Angaben, aus denen sich die Schaltpunkte ergeben, sind nicht plausibel (<i>wSwitchPoint1</i> , <i>w2ndPoint_Size</i> , <i>wHysteresis</i>).
bOut	Bool	binary	Schaltausgang
			TRUE Die Istposition befindet sich innerhalb des definierten Schaltfensters.

Definition des Schaltbereichs

Je nach Einstellung von *bCenterMode* kann der Schaltbereich über eine Start- und Endposition oder über die Angabe von Schaltbereichmitte/Schaltbereichgröße auf der Rundachse definiert werden:

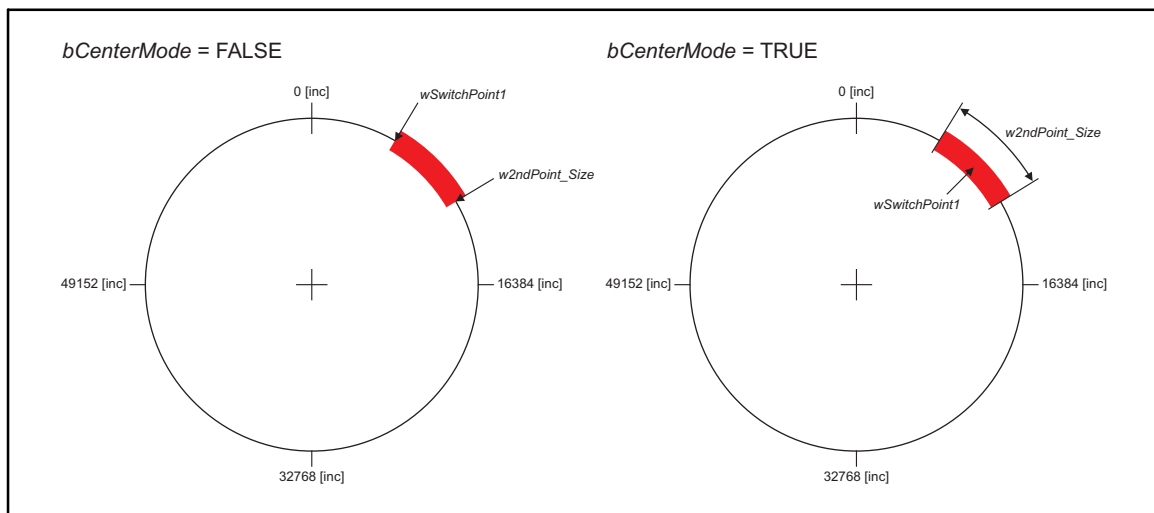
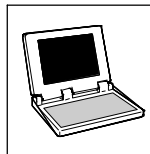


Abb. 2-10

Definition der Schaltbereiche



Totzeitkompensation

Die Verzögerungszeit externer Schaltelemente kann durch Vorgabe der entsprechenden Verzögerungszeit an *wDeadTime* kompensiert werden. Der FB berechnet aus dieser Vorgabe und der aktuellen Geschwindigkeit eine Vorhalteposition, d. h. der Schaltausgang *bOut* wird entsprechend früher gesetzt.

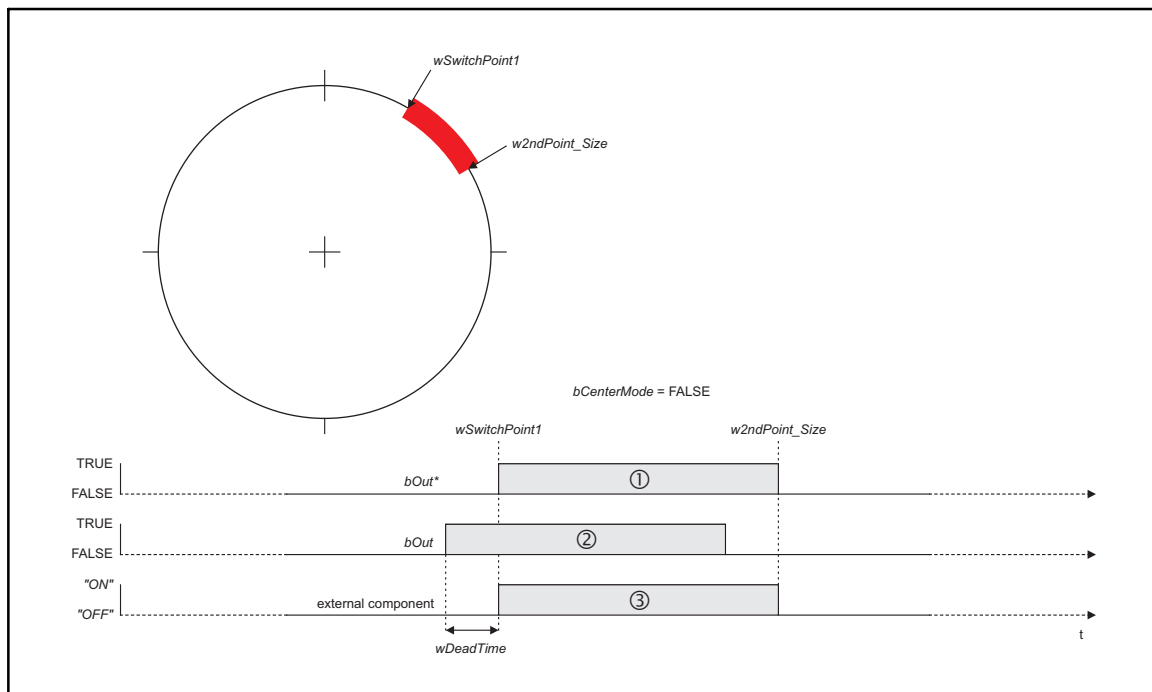


Abb. 2-11

Totzeitkompensation

- ① Geforderter Schaltbereich
- ② Schaltverhalten mit Totzeitkompensation
- ③ Schaltverhalten des externen Schaltelements (mit Totzeit)

Schalthysterese

Durch Vorgabe einer Hysterese am Eingang *wHysteresis* können Sie einen permanenten Zustandswechsel von *bOut* unterbinden, der durch Lageistwertänderungen aufgrund einer mechanischen Unruhe an der Achse auftreten kann.

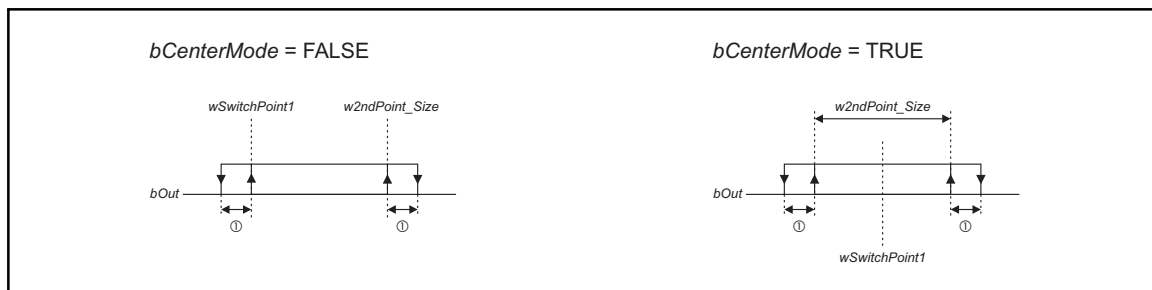
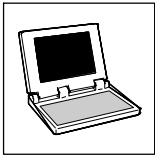


Abb. 2-12

Schalthysterese

- ① *wHysteresis*



Funktionsbibliothek LenzeMotionControl1.lib

Funktionsblöcke für Positionieraufgaben

L_MCSwitchCam - Wertschaltpunkte (Rundachse)

Realisierung mehrerer Nocken

Durch den Aufruf mehrerer Instanzen dieses Funktionsblocks können Sie dementsprechend viele Nocken auf einer Rundachse realisieren:

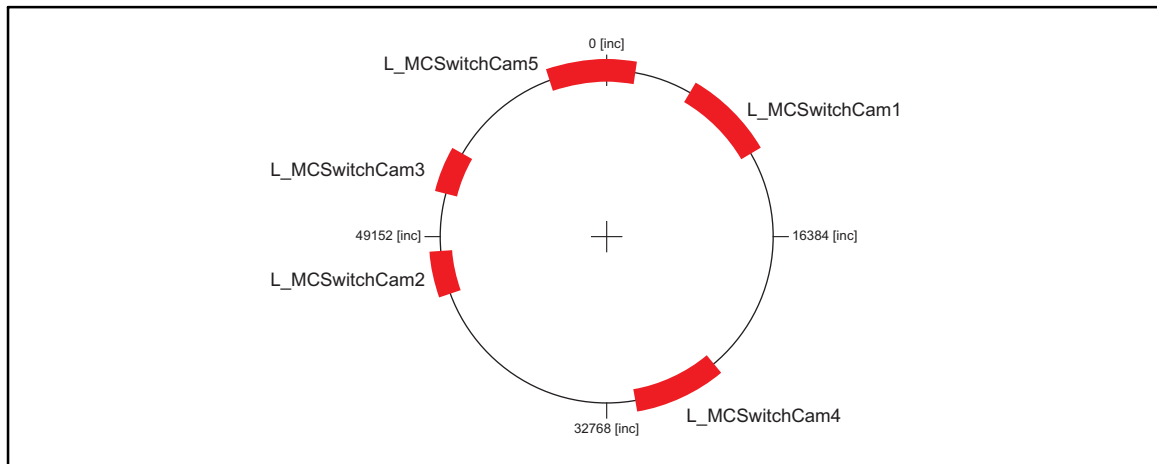
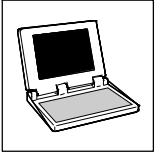


Abb. 2-13

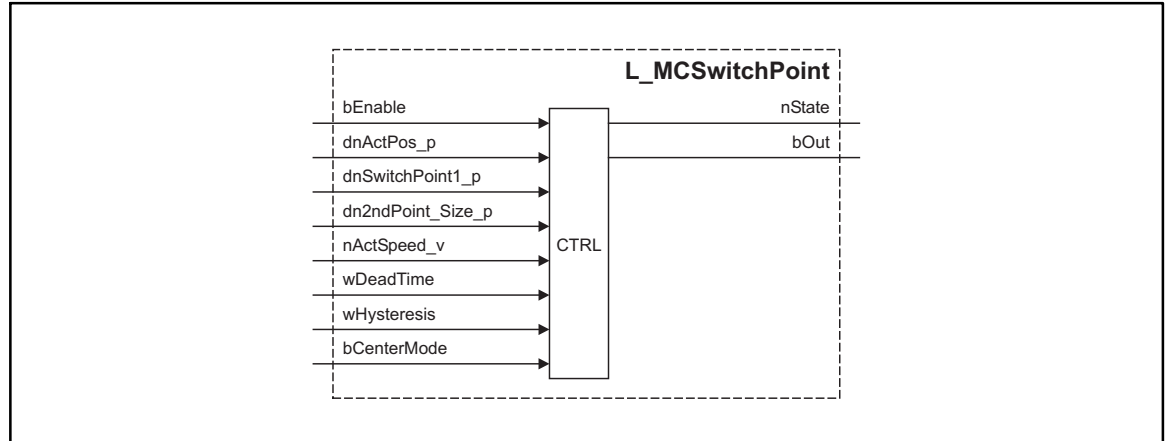
Realisierung mehrerer Nocken durch den Aufruf mehrerer Instanzen des FBs



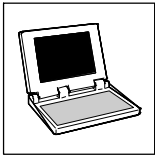
2.7 L_MCSwitchPoint - Wegschaltpunkte

Mit diesem FB können Sie einen Wegschaltpunkt realisieren, d. h. einen digitalen Schalter, dessen binärer Zustand (FALSE/TRUE) abhängig von der Istposition ist.

- Mit Hilfe eines Wegschaltpunktes können z. B. Peripheriegeräte wie Lackierdüsen oder Messer in Abhängigkeit der Werkzeugposition gestartet werden.
- Der FB unterstützt zudem die Kompensation von Verzögerungszeiten externer Schaltelemente (Totzeitkompensation).



⇒ Eingänge (Variablentyp: VAR_INPUT)			
Bezeichner	Datentyp	Signaltyp	Info/Einstellmöglichkeiten
bEnable	Bool	binary	Funktionsblock aktivieren TRUE Funktionsblock aktiviert.
dnActPos_p	Double Integer	position	Istposition in [inc]
dnSwitchPoint1_p	Double Integer	position	Position des ersten Schaltpunktes in [inc]
dn2ndPoint_Size_p	Double Integer	position	Bei <i>bCenterMode</i> = FALSE: Position des zweiten Schaltpunktes in [inc] • <i>dn2ndPoint_Size_p</i> muss größer <i>dnSwitchPoint1_p</i> sein. Bei <i>bCenterMode</i> = TRUE: Größe des Schaltfensters in [inc] • Bedingt durch die symmetrische Anordnung des Fensters um den ersten Schalterpunkt werden ungerade Werte auf gerade Werte abgerundet. • Nur positive Werte sind zulässig.
nActSpeed_v	Integer	velocity	Istgeschwindigkeit in [inc/ms] • 16384 ≙ 15000 rpm
wDeadTime	Word	-	Totzeit in [µs] für Totzeitkompensation • Vorgabe in 10 µS-Schritten • Die sich hieraus ergebende Schaltposition unterliegt keiner Plausibilitätsprüfung.
wHysteresis	Word	-	Schalthysteresis in [inc]
bCenterMode	Bool	binary	Festlegung, wie die Vorgabe <i>dn2ndPoint_Size_p</i> interpretiert wird: FALSE <i>dn2ndPoint_Size_p</i> definiert den zweiten Schalterpunkt. TRUE <i>dn2ndPoint_Size_p</i> definiert die Größe des Schaltfensters.



Funktionsbibliothek LenzeMotionControl1.lib

Funktionsblöcke für Positionieraufgaben

L_MGSwitchPoint - Wegschaltpunkte

Ausgänge ⇒ (Variablentyp: VAR_OUTPUT)			
Bezeichner	Datentyp	Signaltyp	Info
nState	Integer	binary	Status
			10 FB ist nicht aktiv
			0 OK
			-10 Die Angaben, aus denen sich die Schaltpunkte ergeben, sind nicht plausibel (dnSwitchPoint1_p, dn2ndPoint_Size_p, wHysteresis).
bOut	Bool	binary	Schaltausgang
			TRUE Die Istposition befindet sich innerhalb des definierten Schaltfensters.

Definition des Schaltbereichs

Je nach Einstellung von *bCenterMode* kann der Schaltbereich über eine Start- und Endposition oder über die Angabe von Schaltbereichmitte/Schaltbereichgröße definiert werden:

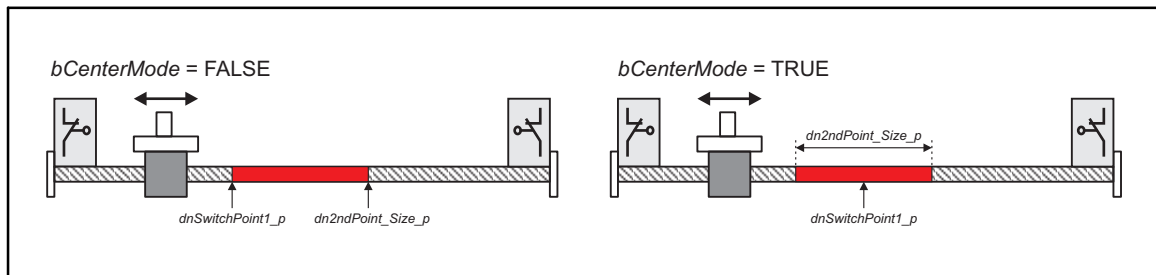


Abb. 2-14

Definition der Schaltbereiche

Totzeitkompensation

Die Verzögerungszeit externer Schaltelemente kann durch Vorgabe der entsprechenden Verzögerungszeit an *wDeadTime* kompensiert werden. Der FB berechnet aus dieser Vorgabe und der aktuellen Geschwindigkeit eine Vorhalteposition, d. h. der Schaltausgang *bOut* wird entsprechend früher gesetzt.

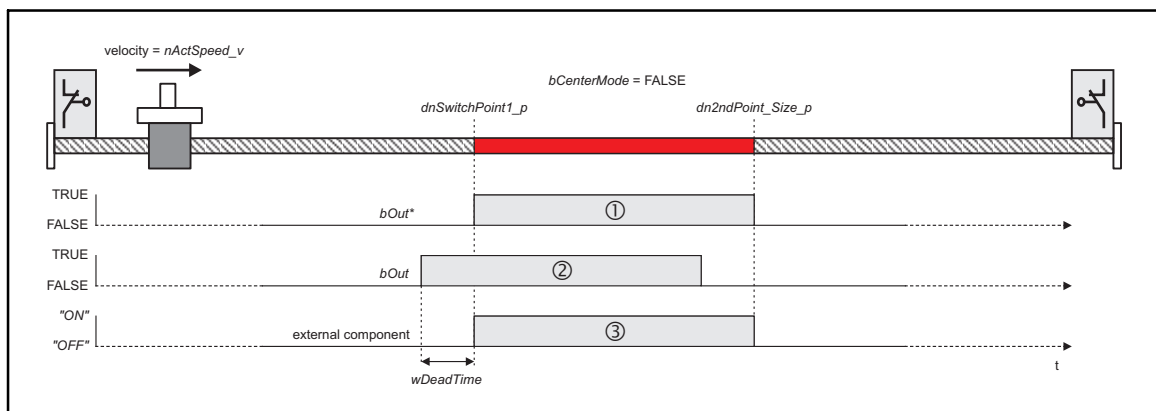
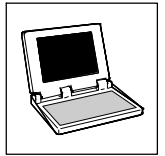


Abb. 2-15

Totzeitkompensation

- ① Schaltverhalten ohne Totzeitkompensation
- ② Schaltverhalten mit Totzeitkompensation
- ③ Schaltverhalten des externen Schaltelements (mit Totzeit)



Schalthysterese

Durch Vorgabe einer Hysterese am Eingang *wHysteresis* können Sie einen permanenten Zustandswechsel von *bOut* unterbinden, der durch Lageistwertänderungen aufgrund einer mechanischen Unruhe an der Achse auftreten kann.

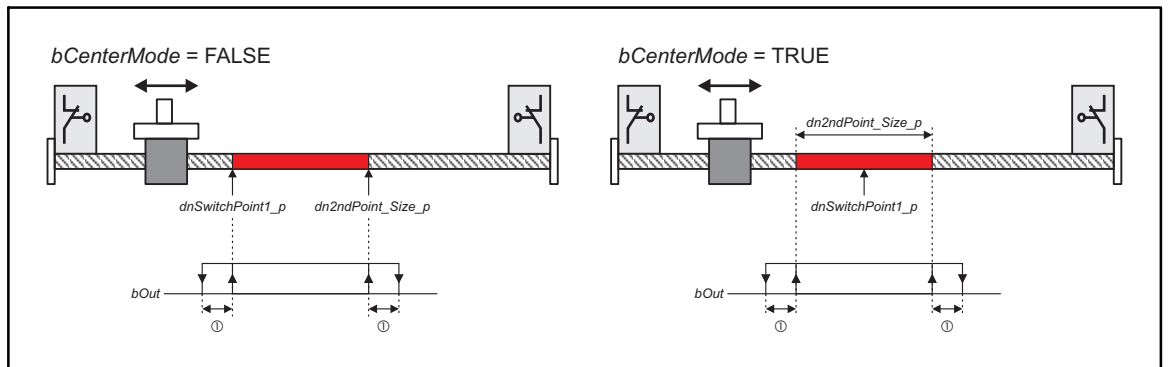


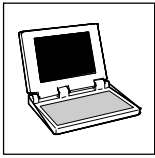
Abb. 2-16

Schalthysterese
① *wHysteresis*



Tipp!

Beachten Sie, dass durch eine über *wDeadTime* eingestellte Verzögerungszeit die Schaltpunkte inklusive der Hysterese verschoben werden.



Funktionsbibliothek LenzeMotionControl1.lib

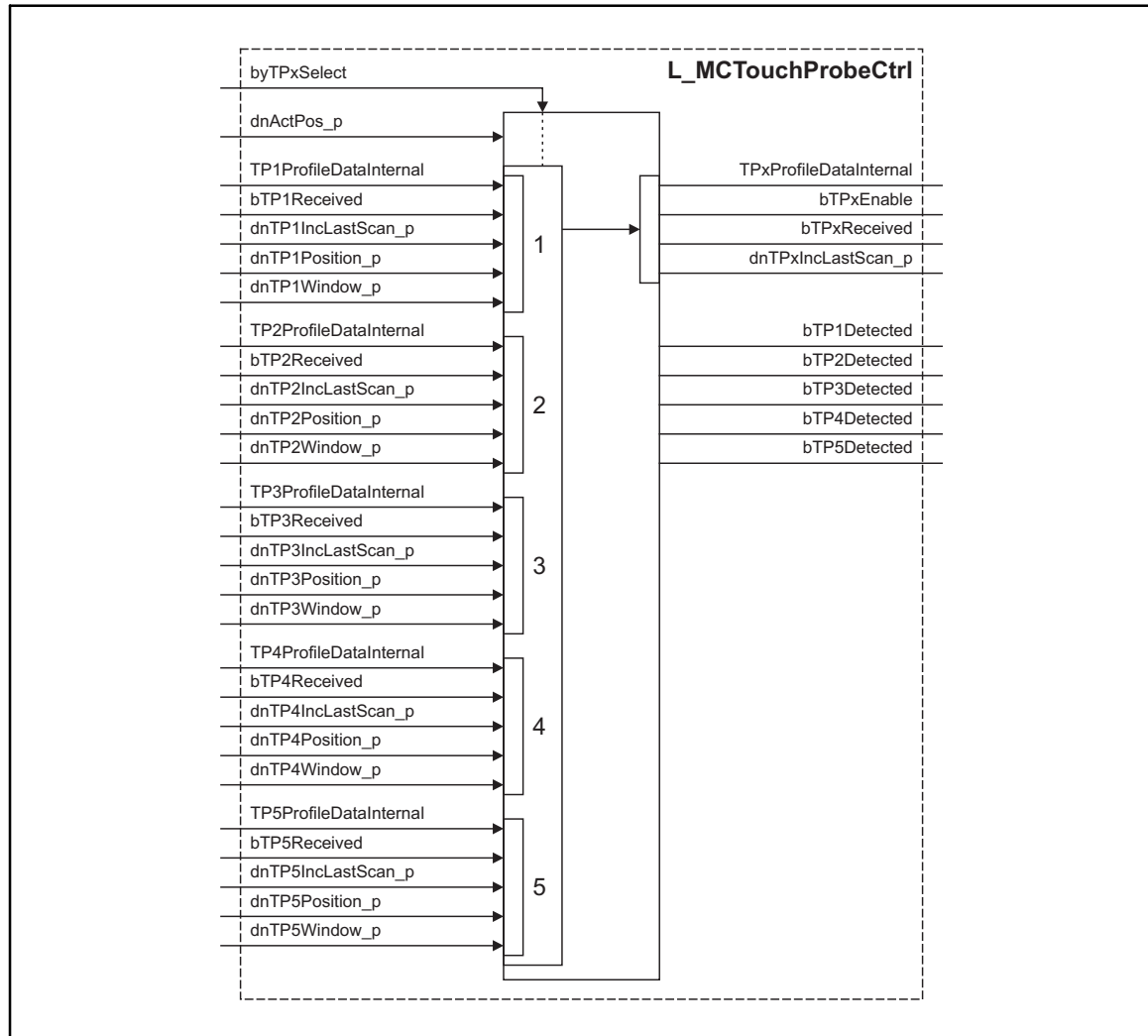
Funktionsblöcke für Positionieraufgaben

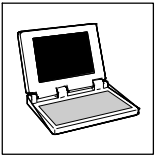
L_MCTouchProbeCtrl - Touch-Probe Vorverarbeitung

2.8

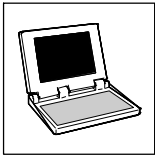
L_MCTouchProbeCtrl - Touch-Probe Vorverarbeitung

Mit diesem FB können Sie eine Umschaltung zwischen fünf verschiedenen Touch-Probe-Signalen realisieren. Zusätzlich lässt sich für jeden Touch-Probe ein Fenster für eine positionsabhängige Freigabe definieren.





⇒ Eingänge (Variablentyp: VAR_INPUT)			
Bezeichner	Datentyp	Signaltyp	Info/Einstellmöglichkeiten
byTPxSelect	Byte	binary	Bitcodierte Aktivierung der Touch-Probe-Kanäle
			0 Alle Ausgänge sind auf “0” gesetzt und alle Touch-Probe-Kanäle sind deaktiviert.
			Bitcodierung:
			Bit 0 0: Touch-Probe-Kanal 1 deaktiviert 1: Touch-Probe-Kanal 1 aktiviert
			Bit 1 0: Touch-Probe-Kanal 2 deaktiviert 1: Touch-Probe-Kanal 2 aktiviert
			Bit 2 0: Touch-Probe-Kanal 3 deaktiviert 1: Touch-Probe-Kanal 3 aktiviert
			Bit 3 0: Touch-Probe-Kanal 4 deaktiviert 1: Touch-Probe-Kanal 4 aktiviert
			Bit 4 0: Touch-Probe-Kanal 5 deaktiviert 1: Touch-Probe-Kanal 5 aktiviert
			Bit 5 reserviert
			Bit 6 reserviert
Bit 7 reserviert			
dnActPos_p	Double Integer	position	Positionsiswert in [inc]
Bei den folgenden Bezeichnern steht das x als Platzhalter für die Nummer 1 ... 5 des entsprechenden Touch-Probe.			
TPxProfileDataInternal	MC_PROFILE_INTERNAL_MEASURING	-	Positionierprofil für die Restwegpositionierung - Datenstruktur, deren Elemente das gesamte Profil beschreiben. - Die entsprechende Datenstruktur x wird zum Ausgang <i>TPxProfileDataInternal</i> durchgeschaltet, wenn sich der Positionsiswert im zugehörigen Touch-Probe-Freigabefenster x befindet und das entsprechende Touch-Probe-Signal erkannt wurde.
bTPxReceived	Bool	binary	Eingang für Touch-Probe - Typischerweise wird hier ein TP-Signal des Betriebssystems angeschlossen. - Der entsprechende Eingang x wird zum Ausgang <i>bTPxReceived</i> durchgeschaltet, wenn sich der Positionsiswert im zugehörigen Touch-Probe-Freigabefenster x befindet.
			TRUE Touch-Probe Signal wurde erkannt.
dnTPxIncLastScan_p	Double Integer	position	Winkeldifferenz in [inc] zwischen Latchzeitpunkt und Startzeitpunkt der Task Der entsprechende Eingang x wird zum Ausgang <i>dnTPxIncLastScan_p</i> durchgeschaltet, wenn sich der Positionsiswert im zugehörigen Touch-Probe-Freigabefenster x befindet und das entsprechende Touch-Probe-Signal erkannt wurde.
dnTPxPosition_p	Double Integer	position	Position des Touch-Probe-Sensors in [inc] bezogen auf die Referenzposition. Der Positionswert inklusive der Fenstergröße darf den Zahlenbereich Double Integer nicht überschreiten.
dnTPxWindow_p	Double Integer	position	Größe des Touch-Probe-Freigabefensters in [inc] vor und nach der Position des Touch-Probe-Sensors. Wird die Fenstergröße auf 0 [inc] gesetzt, so wird der Touch-Probe-Kanal unabhängig von der Istposition verarbeitet. Achtung: Die einzelnen Fenster dürfen sich nicht überlappen, da ansonsten eine einwandfreie Zuordnung der Touch-Probe-Kanäle nicht möglich ist.



Funktionsbibliothek LenzeMotionControl1.lib

Funktionsblöcke für Positionieraufgaben

L_MCTouchProbeCtrl - Touch-Probe Vorverarbeitung

Ausgänge ⇨ (Variablentyp: VAR_OUTPUT)						
Bezeichner	Datentyp	Signaltyp	Info			
TPxProfileDataInternal	MC_PROFILE_INTERNAL_MEASURING	-	Abgabe des Positionierprofils für die Restwegpositionierung - Datenstruktur, deren Elemente das gesamte Profil beschreiben. - Auf diesen Ausgang wird die entsprechende Profilstruktur des Touch-Probe-Kanals durchgeschaltet, der - über <i>byTPxSelect</i> aktiviert ist und - in dessen Freigabefenster sich der Istwert befindet.			
bTPxEnable	Bool	binary	Meldesignal "Touch-Probe Signalverarbeitung aktivieren" <table><tr><td>TRUE</td><td>Die Istposition befindet sich im Freigabefenster eines über <i>byTPxSelect</i> freigeschalteten Touch-Probe-Kanals.</td></tr></table>		TRUE	Die Istposition befindet sich im Freigabefenster eines über <i>byTPxSelect</i> freigeschalteten Touch-Probe-Kanals.
TRUE	Die Istposition befindet sich im Freigabefenster eines über <i>byTPxSelect</i> freigeschalteten Touch-Probe-Kanals.					
bTPxReceived	Bool	binary	Meldesignal "Touch-Probe erkannt" - Auf diesen Ausgang wird das entsprechende Eingangssignal des Touch-Probe-Kanals durchgeschaltet, der - über <i>byTPxSelect</i> aktiviert ist und - in dessen Freigabefenster sich der Istwert befindet. <table><tr><td>TRUE</td><td> Touch-Probe-Signal wurde erkannt. - Das Meldesignal wird nach einem Taskzyklus auf FALSE zurückgesetzt. </td></tr></table>		TRUE	Touch-Probe-Signal wurde erkannt. Das Meldesignal wird nach einem Taskzyklus auf FALSE zurückgesetzt.
TRUE	Touch-Probe-Signal wurde erkannt. Das Meldesignal wird nach einem Taskzyklus auf FALSE zurückgesetzt.					
dnTPxIncLastScan_p	Double Integer	position	Abgabe der Winkeldifferenz in [inc] zwischen Latchzeitpunkt und Startzeitpunkt der Task - Auf diesen Ausgang wird das entsprechende Eingangssignal des Touch-Probe-Kanals durchgeschaltet, der - über <i>byTPxSelect</i> aktiviert ist und - in dessen Freigabefenster sich der Istwert befindet.			
Meldesignale "Touch-Probe erkannt":						
bTP1Detected	Bool	binary	TRUE	Touch-Probe 1 wurde erkannt.		
bTP2Detected				Touch-Probe 2 wurde erkannt.		
bTP3Detected				Touch-Probe 3 wurde erkannt.		
bTP4Detected				Touch-Probe 4 wurde erkannt.		
bTP5Detected				Touch-Probe 5 wurde erkannt.		

Aktivierung/Deaktivierung der Touch-Probe-Kanäle

Die Aktivierung bzw. Deaktivierung der Touch-Probe-Kanäle erfolgt bitcodiert über den Eingang *byTPxSelect*:

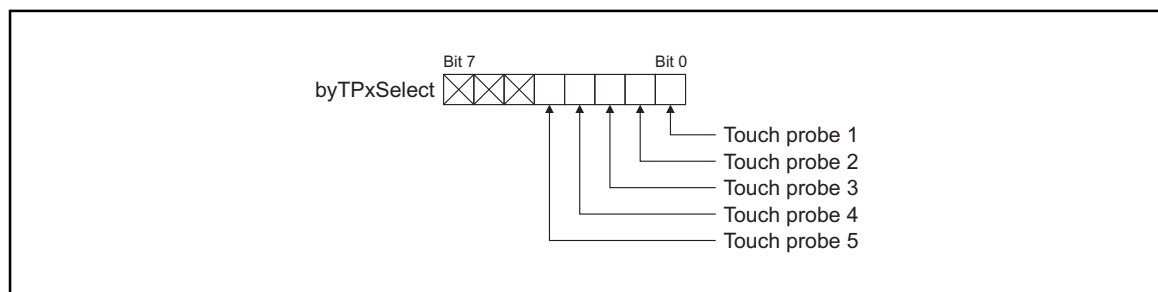
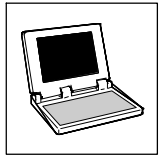


Abb. 2-17

Bitcodierte Aktivierung/Deaktivierung

- Hat *byTPxSelect* den Wert "0", so sind alle Ausgänge auf NULL gesetzt und alle Touch-Probe-Kanäle deaktiviert.



Positionsabhängige Freigabe eines Touch-Probe

Der FB unterstützt die positionsabhängige Freigabe eines Touch-Probe durch die Vorgabe eines sogenannten Freigabefensters. Dadurch kann die ungewollte Auslösung eines Touch-Probe ausgeschlossen werden, wenn sich das Werkzeug an einer Position außerhalb dieses Fensters befindet:

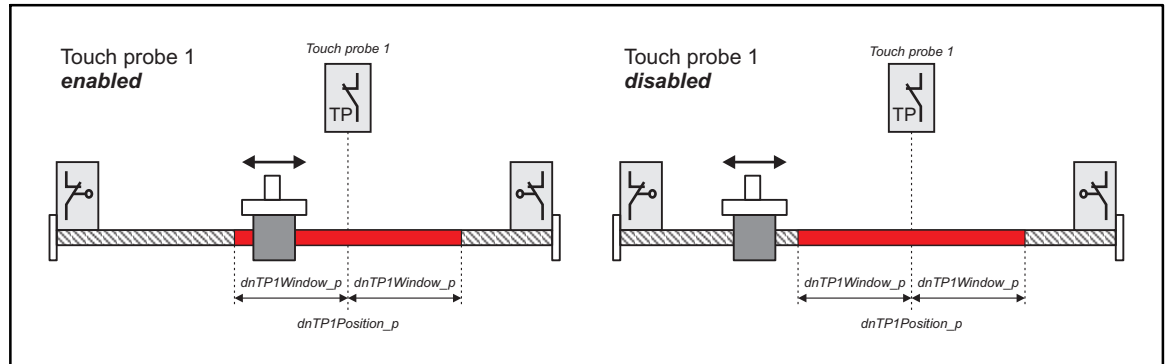


Abb. 2-18

Beispiel: Positionsabhängige Freigabe für Touch-Probe 1

Für jeden Touch-Probe-Kanal kann ein eigenes Freigabefenster definiert werden:

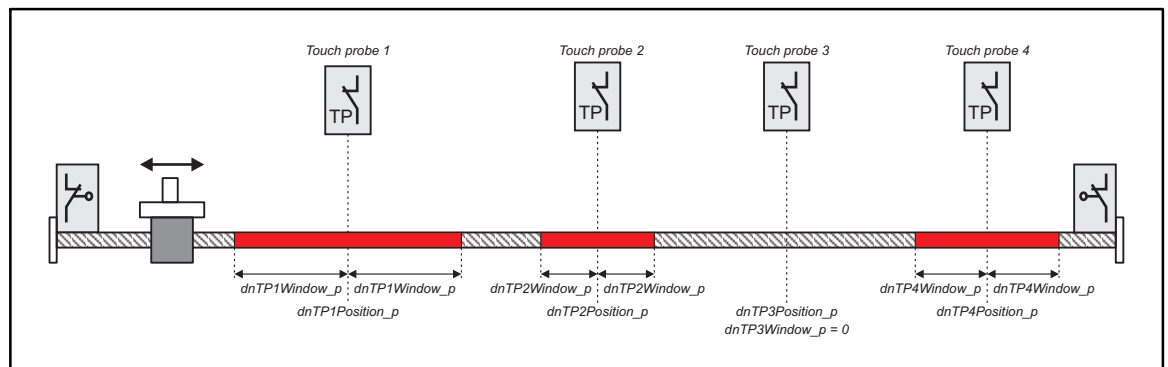


Abb. 2-19

Beispiel: Definition der Freigabefenster für Touch-Probe 1 ... Touch-Probe 4

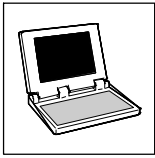
- Die Position des Touch-Probe-Sensors bezogen auf die Referenzposition geben Sie über *dnTPxPosition_p* vor.
- Die Größe des Fensters vor und nach der Position des Touch-Probe-Sensors geben Sie über *dnTPxWindow_p* vor.



Hinweise!

- Die einzelnen Fenster dürfen sich nicht überlappen, da ansonsten eine einwandfreie Zuordnung der Touch-Probe-Kanäle nicht möglich ist.
- Wird die Fenstergröße auf 0 [inc] gesetzt, so wird der Touch-Probe-Kanal unabhängig von der Istposition verarbeitet. (Siehe Abb. 2-19, Touch-Probe 3)
- Der Positionswert inklusive der Fenstergröße darf den Zahlenbereich "Double Integer" nicht überschreiten.

- *bTPxEnable* wird nur dann TRUE, wenn ein Touch-Probe-Kanal über *byTPxSelect* aktiviert wurde und sich die Istposition in dessen definierten Freigabefenster befindet, oder zumindest eine Fenstergröße auf 0 [inc] gesetzt ist.



Funktionsbibliothek LenzMotionControl1.lib

Funktionsblöcke für Positionieraufgaben

L_MCTouchProbeCtrl - Touch-Probe Vorverarbeitung

Ausgabe der Touch-Probe-Eingangssignale

Befindet sich die Istposition innerhalb des Freigabefensters eines aktivierten Touch-Probe-Kanals und das entsprechende Touch-Probe-Signal wurde erkannt, so werden die Eingänge des Kanals auf die entsprechenden Ausgänge durchgeschaltet.

Beispiel:

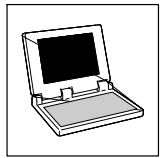
Touch-Probe-Kanal 2 ist aktiviert (*byTPxSelect* - Bit1 = "1"), die Istposition (*dnActPos_p*) befindet sich innerhalb des Freigabefensters von Touch-Probe-Kanal 2 (*dnTP2Window_p*) und an *bTP2Received* liegt ein TRUE-Signal an (TP erkannt):

- Eingang *TP2ProfileDataInternal* → Ausgang *TPxProfileDataInternal*
- Eingang *bTP2Received* → Ausgang *bTPxReceived*
- Eingang *dnTP2IncActLastScan_p* → Ausgang *dnTPxIncActLastScan_p*

Meldeausgänge für erkannte Touch-Probe

Die Ausgänge *bTP1Detected* ... *bTP5Detected* werden auf TRUE gesetzt, wenn der entsprechende Touch-Probe innerhalb des Freigabefensters bzw. bei Fenstergröße = 0 [inc] unabhängig von der Istposition erkannt wurde.

- Voraussetzung hierfür ist, dass der entsprechende Touch-Probe-Kanal über *byTPxSelect* ausgewählt ist.
- Rückgesetzt werden die Meldeausgänge durch Setzen von *byTPxSelect* auf "0".



3 Funktionsblöcke zur Umrechnung von Einheiten

Dieses Kapitel enthält Informationen zu den Funktionsblöcken, die Sie für die Umrechnung von Einheiten verwenden können:

Funktionsblock	Umrechnung	von $\Rightarrow$ nach*	Eingang	Ausgang	Info
L_MCGetPosition	Position	intern $\Rightarrow$ extern	inc	unit	3-2
L_MCGetVelocity	Geschwindigkeit	intern $\Rightarrow$ extern	inc/ms	unit/s	3-3
L_MCSetAcceleration	Beschleunigung	extern $\Rightarrow$ intern	unit/s ²	16384/A _{max}	3-4
L_MCSetDeceleration	Abbremsung	extern $\Rightarrow$ intern	unit/s ²	16384/A _{max}	3-5
L_MCSetProfileConverter	Alle Profilwerte • Die Eingangs- und Ausgangswerte liegen in Form einer Struktur vor.	extern $\Rightarrow$ intern	Kombination: unit, unit/s, unit/s ²	Kombination: inc, 16384/V _{max} 16384/A _{max}	3-6
L_MCSetProfileValues	Alle Profilwerte • Die Eingangswerte können einzeln angegeben werden, die Ausgangswerte liegen in Form einer Struktur vor.	extern $\Rightarrow$ intern	Kombination: unit, unit/s, unit/s ²	Kombination: inc, 16384/V _{max} 16384/A _{max}	3-7
L_MCSetPosition	Position	extern $\Rightarrow$ intern	unit	inc	3-9
L_MCSetVelocity	Verfahrgeschwindigkeit	extern $\Rightarrow$ intern	unit/s	16384/V _{max}	3-10

* intern = interne Einheit, extern = physikalische Einheit



Tipp!

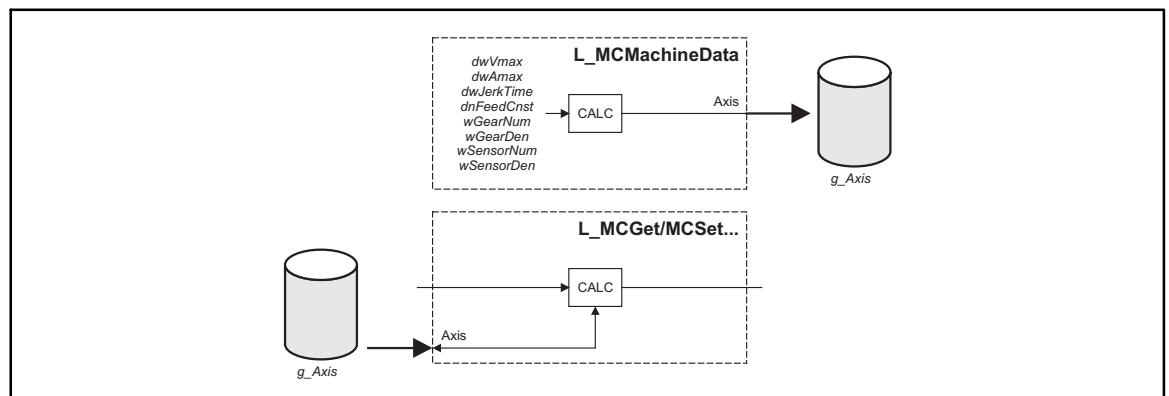
Sie müssen jeden Eingangswert in der Fixkommadarstellung beschreiben.

- Das bedeutet, dass die letzten vier Zahlenwerte Nachkommastellen sind.
- Beispiel: Für den Eingangswert "1.0000" müssen Sie den Zahlenwert "10000" eintragen.

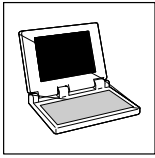
Übergabe der Maschinenparameter

Zusätzlich zu dem umzurechnenden Wert benötigen die in diesem Kapitel beschriebenen Funktionsblöcke am Eingang *Axis* die Maschinenparameter.

- Verbinden Sie dazu den Eingang *Axis* mit einer globalen Variable (z. B. *g_Axis*), die vom FB **L_MCMachineData** beschrieben wird:



- *g_Axis* ist eine Datenstruktur (Datentyp: MC_AXIS_REF), deren Elemente die Maschinenparameter im internen Maßsystem beinhalten. Siehe auch Kap. 4, "Strukturierte Datentypen". (4-1)



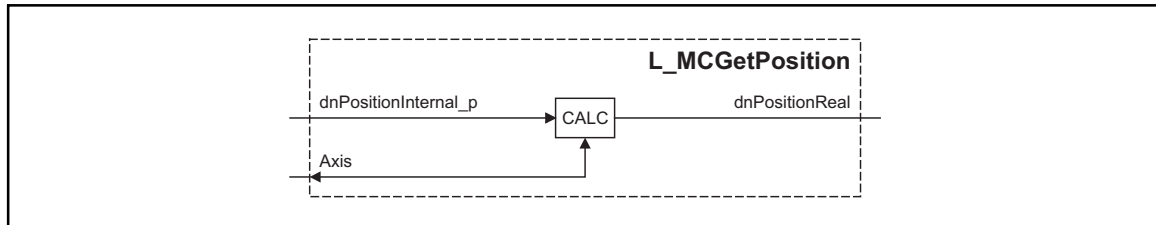
Funktionsbibliothek LenzeMotionControl1.lib

Funktionsblöcke zur Umrechnung von Einheiten

L_MCGetPosition

3.1 L_MCGetPosition

Dieser FB rechnet die interne Einheit der Istposition in die physikalische Einheit um.



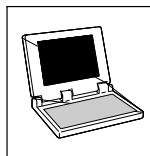
⇒ Eingänge (Variablentyp: VAR_INPUT)			
Bezeichner	Datentyp	Signaltyp	Info
dnPositionInternal_p	Double Integer	position	Position in [inc] 65536 = 1 Motorumdrehung

Ausgänge ⇒ (Variablentyp: VAR_OUTPUT)			
Bezeichner	Datentyp	Signaltyp	Info
dnPositionReal	Double Integer	-	Position in [unit] 10000 = 1.0000 unit - Größter darstellbarer Wert: 214748.3647 unit

⇒ Ein-/Ausgänge ⇒ (Variablentyp: VAR_IN_OUT)			
Bezeichner	Datentyp	Signaltyp	Info
Axis	MC_AXIS_REF	-	Maschinenparameter - Datenstruktur, deren Elemente die Maschinenparameter im internen Maßsystem beinhalten. - Verbinden Sie diesen Eingang mit einer globalen Variable, die vom FB L_MCMachineData beschrieben wird.

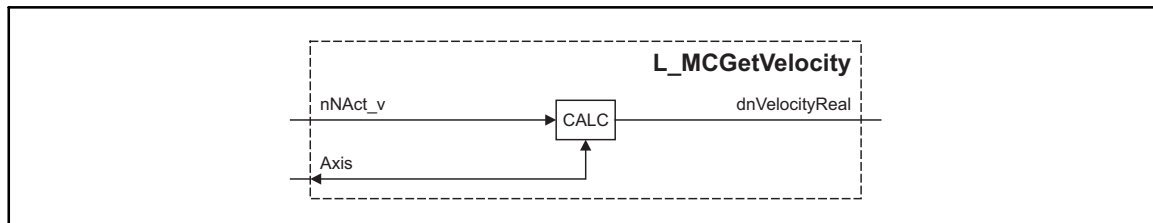
Formel

$$dnPositionReal \text{ [unit]} = dnPositionInternal_p \text{ [inc]} \cdot \frac{10000 \cdot \text{Vorschubkonstante [unit/Umdr.]} \cdot \text{Getriebefaktornenner} \cdot \text{Sensorfaktorzähler}}{65536 \text{ [inc/Umdr.]} \cdot \text{Getriebefaktorzähler} \cdot \text{Sensorfaktornenner}}$$



3.2 L_MCGetVelocity

Dieser FB rechnet die interne Einheit der Istgeschwindigkeit in die physikalische Einheit um.



⇒ Eingänge (Variablentyp: VAR_INPUT)			
Bezeichner	Datentyp	Signaltyp	Info
nNAct_v	Integer	velocity	Geschwindigkeit in [inc/ms] • 16384 [inc/ms] = 15000 Umdr./min

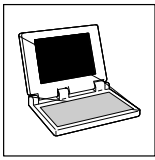
Ausgänge ⇒ (Variablentyp: VAR_OUTPUT)			
Bezeichner	Datentyp	Signaltyp	Info
dnVelocityReal	Double Integer	-	Geschwindigkeit in [unit/s] • 10000 = 1.0000 unit/s • Größter darstellbarer Wert: 214748.3647 unit/s

⇒ Ein-/Ausgänge ⇒ (Variablentyp: VAR_IN_OUT)			
Bezeichner	Datentyp	Signaltyp	Info
Axis	MC_AXIS_REF	-	Maschinenparameter • Datenstruktur, deren Elemente die Maschinenparameter im internen Maßsystem beinhalten. • Verbinden Sie diesen Eingang mit einer globalen Variable, die vom FB L_MCMachineData beschrieben wird.

Formel

$$dnVelocityReal \text{ [unit/s]} = nNAct_v \text{ [inc/ms]} \cdot 1000 \text{ [ms/s]} \cdot X$$

$$\text{mit } X = \frac{10000 \cdot \text{Vorschubkonstante [unit/Umdr.]} \cdot \text{Getriebefaktornenner} \cdot \text{Sensorfaktorzähler}}{65536 \text{ [inc/Umdr.]} \cdot \text{Getriebefaktorzähler} \cdot \text{Sensorfaktornenner}}$$



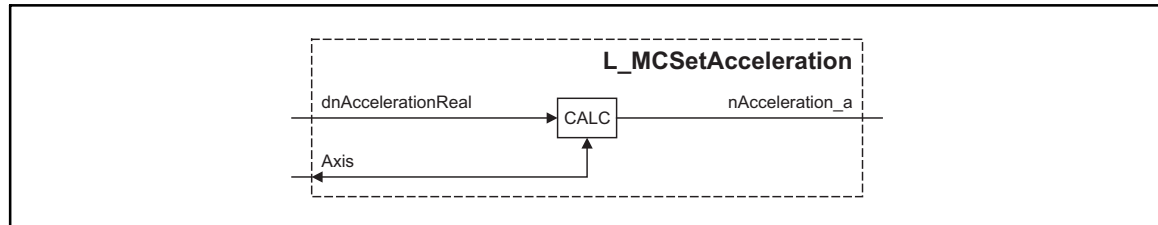
Funktionsbibliothek LenzeMotionControl1.lib

Funktionsblöcke zur Umrechnung von Einheiten

L_MCSetAcceleration

3.3 L_MCSetAcceleration

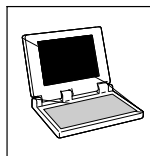
Dieser FB rechnet die physikalische Einheit der Beschleunigung in die interne Einheit um.



⇒ Eingänge (Variablentyp: VAR_INPUT)			
Bezeichner	Datentyp	Signaltyp	Info
dnAccelerationReal	Double Integer	-	Beschleunigung in [unit/s ²] • 10000 = 1.0000 unit/s ²

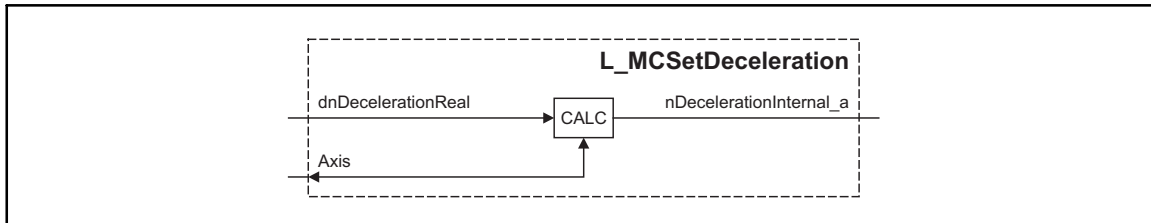
Ausgänge ⇒ (Variablentyp: VAR_OUTPUT)			
Bezeichner	Datentyp	Signaltyp	Info
nAcceleration_a	Integer	-	Beschleunigung • Der Wert stellt den relativen Bezug zur Maximalbeschleunigung dar, die über den FB L_MCMachineData definiert wurde (16384/A _{max}). • Der Wert wird auf ±16384 begrenzt. • Der Wert ist "0", wenn auch die Bezugsgröße "0" ist. (Axis.dwMaxAcceleration vom L_MCMachineData)

⇒ Ein-/Ausgänge ⇒ (Variablentyp: VAR_IN_OUT)			
Bezeichner	Datentyp	Signaltyp	Info
Axis	MC_AXIS_REF	-	Maschinenparameter • Datenstruktur, deren Elemente die Maschinenparameter im internen Maßsystem beinhalten. • Verbinden Sie diesen Eingang mit einer globalen Variable, die vom FB L_MCMachineData beschrieben wird.



3.4 L_MCSetDeceleration

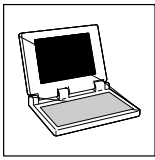
Dieser FB rechnet die physikalische Einheit der Abbremsung in die interne Einheit um.



⇒ Eingänge (Variablentyp: VAR_INPUT)			
Bezeichner	Datentyp	Signaltyp	Info
dnDecelerationReal	Double Integer	-	Abbremsung in [unit/s ²] • 10000 = 1.0000 unit/s ²

Ausgänge ⇒ (Variablentyp: VAR_OUTPUT)			
Bezeichner	Datentyp	Signaltyp	Info
nDecelerationInternal_a	Integer	-	Abbremsung • Der Wert stellt den relativen Bezug zur Maximalbeschleunigung dar, die über den FB L_MCMachineData definiert wurde (16384/A _{max}). • Der Wert wird auf ±16384 begrenzt. • Der Wert ist "0", wenn auch die Bezugsgröße "0" ist. (Axis.dwMaxAcceleration vom L_MCMachineData)

⇒ Ein-/Ausgänge ⇒ (Variablentyp: VAR_IN_OUT)			
Bezeichner	Datentyp	Signaltyp	Info
Axis	MC_AXIS_REF	-	Maschinenparameter • Datenstruktur, deren Elemente die Maschinenparameter im internen Maßsystem beinhalten. • Verbinden Sie diesen Eingang mit einer globalen Variable, die vom FB L_MCMachineData beschrieben wird.



Funktionsbibliothek LenzeMotionControl1.lib

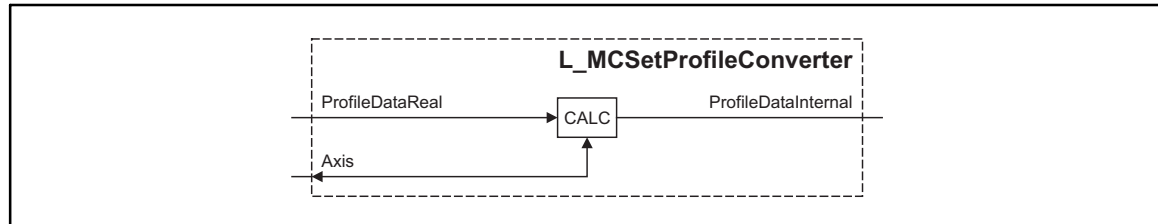
Funktionsblöcke zur Umrechnung von Einheiten

L_MCSetProfileConverter

3.5 L_MCSetProfileConverter

Dieser FB rechnet die physikalischen Einheiten eines kompletten Profils in interne Einheiten um.

- Während dem FB **L_MCSetProfileValues** die Eingangswerte einzeln übergeben werden müssen, können diesem FB die Eingangswerte als Struktur übergeben werden.



⇒ Eingänge (Variablentyp: VAR_INPUT)			
Bezeichner	Datentyp	Signaltyp	Info
ProfileDataReal	MC_PROFILE_REAL_MEASURING	-	Positionierprofil - Datenstruktur, deren Elemente das gesamte Profil beschreiben. - Die Werte liegen in den physikalischen Einheiten vor.

Weitere Informationen zum Datentyp MC-PROFILE_REAL_MEASURING entnehmen Sie dem Kapitel "Strukturierte Datentypen" (□ 4-3).

Ausgänge ⇒ (Variablentyp: VAR_OUTPUT)			
Bezeichner	Datentyp	Signaltyp	Info
ProfileDataInternal	MC_PROFILE_INTERNAL_MEASURING	-	Positionierprofil - Datenstruktur, deren Elemente das gesamte Profil beschreiben. - Die Werte liegen im internen Maßsystem vor. - Der Werte der Elemente <i>nTraversingVelocity</i>, <i>nAcceleration</i> und <i>nDeceleration</i> werden auf ± 16384 begrenzt. - Ist die Geschwindigkeit (<i>dwVmax</i>) oder die Beschleunigung (<i>dwAmax</i>) = "0", so haben auch die Elemente <i>nTraversingVelocity</i>, <i>nAcceleration</i> und <i>nDeceleration</i> den Wert "0".

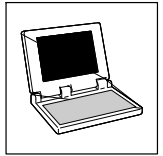
Weitere Informationen zum Datentyp MC-PROFILE_INTERNAL_MEASURING entnehmen Sie dem Kapitel "Strukturierte Datentypen" (□ 4-4).

⇒ Ein-/Ausgänge ⇒ (Variablentyp: VAR_IN_OUT)			
Bezeichner	Datentyp	Signaltyp	Info
Axis	MC_AXIS_REF	-	Maschinenparameter - Datenstruktur, deren Elemente die Maschinenparameter im internen Maßsystem beinhalten. - Verbinden Sie diesen Eingang mit einer globalen Variable, die vom FB L_MCMachineData beschrieben wird.



Tip!

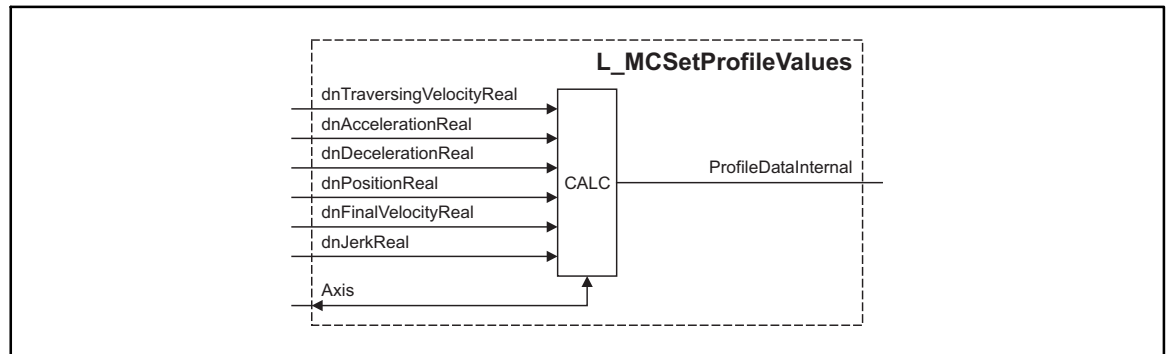
- Die Umrechnung für die Position erfolgt mit Restwertbehandlung. Ein Restwert wird bei der nächsten Rechnung berücksichtigt. Für die korrekte Restwertbehandlung muss der FB vor jedem neuen Profilstart einmalig aufgerufen werden.
- Dieser FB bildet eine Zusammenfassung der FB's **L_MCSetVelocity**, **L_MCSetAcceleration**, **L_MCSetDeceleration** und **L_MCSetPosition**.



3.6 L_MCSetProfileValues

Dieser FB rechnet die physikalischen Einheiten eines kompletten Profils in interne Einheiten um.

- Während dem FB **L_MCSetProfileConverter** die Eingangswerte als Struktur übergeben werden müssen, können diesem FB die Eingangswerte einzeln übergeben werden.

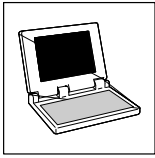


⇒ Eingänge (Variablentyp: VAR_INPUT)			
Bezeichner	Datentyp	Signaltyp	Info
dnTraversingVelocityReal	Double Integer	-	Verfahrgeschwindigkeit in [unit/s] • 10000 = 1.0000 unit/s
dnAccelerationReal	Double Integer	-	Beschleunigung in [unit/s²] • 10000 = 1.0000 unit/s²
dnDecelerationReal	Double Integer	-	Abbremsung in [unit/s²] • 10000 = 1.0000 unit/s²
dnPositionReal	Double Integer	-	Position in [unit] • 10000 = 1.0000 unit
dnFinalVelocityReal	Double Integer	-	Endgeschwindigkeit in [unit/s] • 10000 = 1.0000 unit/s
dnJerkReal	Double Integer	-	Reserviert

Ausgänge ⇒ (Variablentyp: VAR_OUTPUT)			
Bezeichner	Datentyp	Signaltyp	Info
ProfileDataInternal	MC_PROFILE_INTERNAL_MEASURING	-	Positionierprofil • Datenstruktur, deren Elemente das gesamte Profil beschreiben. • Die Werte liegen im internen Maßsystem vor. • Der Werte der Elemente <i>nTraversingVelocity</i> , <i>nAcceleration</i> und <i>nDeceleration</i> werden auf ±16384 begrenzt. • Ist die Geschwindigkeit (dwVmax) oder die Beschleunigung (dwAmax) = "0", so haben auch die Elemente <i>nTraversingVelocity</i> , <i>nAcceleration</i> und <i>nDeceleration</i> den Wert "0".

Weitere Informationen zum Datentyp MC_PROFILE_INTERNAL_MEASURING entnehmen Sie dem Kapitel "Strukturierte Datentypen" (4-4).

⇒ Ein-/Ausgänge ⇒ (Variablentyp: VAR_IN_OUT)			
Bezeichner	Datentyp	Signaltyp	Info
Axis	MC_AXIS_REF	-	Maschinenparameter • Datenstruktur, deren Elemente die Maschinenparameter im internen Maßsystem beinhalten. • Verbinden Sie diesen Eingang mit einer globalen Variable, die vom FB L_MCMachineData beschrieben wird.



Funktionsbibliothek LenzeMotionControl1.lib

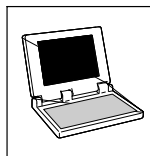
Funktionsblöcke zur Umrechnung von Einheiten

L_MCSetProfileValues



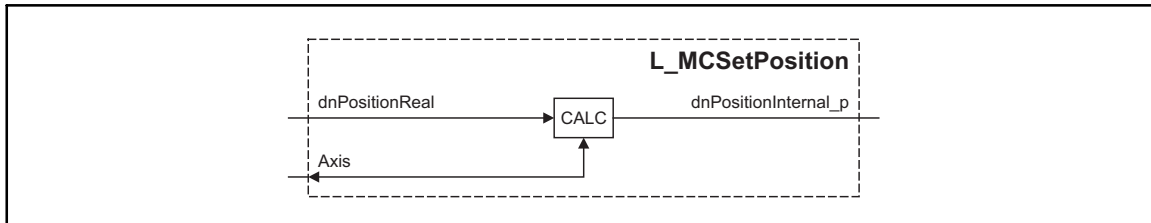
Tipp!

- Die Umrechnung für die Position erfolgt mit Restwertbehandlung. Ein Restwert wird bei der nächsten Rechnung berücksichtigt. Für die korrekte Restwertbehandlung muss der FB vor jedem neuen Profilstart einmalig aufgerufen werden.
- Dieser FB bildet eine Zusammenfassung der FB's **L_MCSetVelocity**, **L_MCSetAcceleration**, **L_MCSetDeceleration** und **L_MCSetPosition**.



3.7 L_MCSetPosition

Dieser FB rechnet die physikalische Einheit der Zielposition in die interne Einheit um.



⇒ Eingänge (Variablentyp: VAR_INPUT)			
Bezeichner	Datentyp	Signaltyp	Info
dnPositionReal	Double Integer	position	Position in [unit] • 10000 = 1.0000 unit

Ausgänge ⇒ (Variablentyp: VAR_OUTPUT)			
Bezeichner	Datentyp	Signaltyp	Info
dnPositionInternal_p	Double Integer	position	Position in [inc] • 65536 = 1 Motorumdrehung

⇒ Ein-/Ausgänge ⇒ (Variablentyp: VAR_IN_OUT)			
Bezeichner	Datentyp	Signaltyp	Info
Axis	MC_AXIS_REF	-	Maschinenparameter • Datenstruktur, deren Elemente die Maschinenparameter im internen Maßsystem beinhalten. • Verbinden Sie diesen Eingang mit einer globalen Variable, die vom FB L_MCMachineData beschrieben wird.

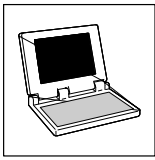
Formel

$$dnPositionInternal_p \text{ [inc]} = \frac{dnPositionReal \text{ [unit]}}{10000} \cdot \frac{65536 \text{ [inc/Umdr.]} \cdot \text{Getriebefaktor} \cdot \text{Sensorfaktornenner}}{\text{Vorschubkonstante [unit/Umdr.]} \cdot \text{Getriebefaktornenner} \cdot \text{Sensorfaktor}} \cdot \text{Zähler}$$



Tipp!

Die Umrechnung für die Position erfolgt mit Restwertbehandlung. Ein Restwert wird bei der nächsten Rechnung berücksichtigt. Für die korrekte Restwertbehandlung muss der FB vor jedem neuen Profilstart einmalig aufgerufen werden.



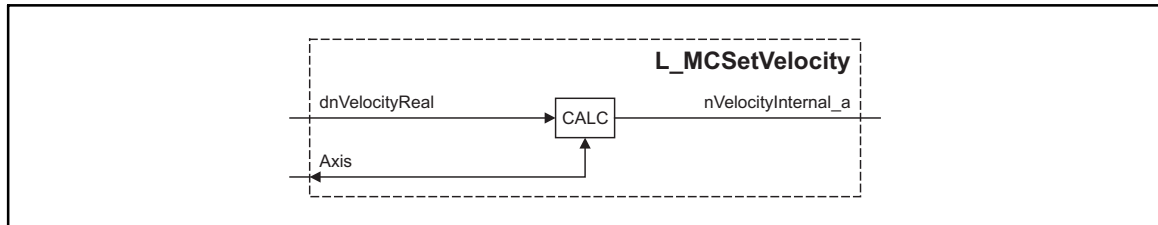
Funktionsbibliothek LenzeMotionControl1.lib

Funktionsblöcke zur Umrechnung von Einheiten

L_MCSetVelocity

3.8 L_MCSetVelocity

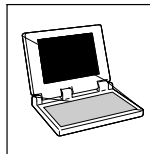
Dieser FB rechnet die physikalische Einheit der Verfahrensgeschwindigkeit in die interne Einheit um.



⇒ Eingänge (Variablentyp: VAR_INPUT)			
Bezeichner	Datentyp	Signaltyp	Info
dnVelocityReal	Double Integer	-	Verfahrensgeschwindigkeit in [unit/s] • 10000 = 1.0000 unit/s

Ausgänge ⇒ (Variablentyp: VAR_OUTPUT)			
Bezeichner	Datentyp	Signaltyp	Info
nVelocityInternal_a	Integer	-	Verfahrensgeschwindigkeit • Der Wert stellt den relativen Bezug zur Maximalgeschwindigkeit dar, die über den FB L_MCMachineData definiert wurde ($16384/V_{\max}$). • Der Wert wird auf ± 16384 begrenzt. • Der Wert ist "0", wenn auch die Bezugsgröße "0" ist. (<i>Axis.dwMaxVelocity</i> vom L_MCMachineData)

⇒ Ein-/Ausgänge ⇒ (Variablentyp: VAR_IN_OUT)			
Bezeichner	Datentyp	Signaltyp	Info
Axis	MC_AXIS_REF	-	Maschinenparameter • Datenstruktur, deren Elemente die Maschinenparameter im internen Maßsystem beinhalten. • Verbinden Sie diesen Eingang mit einer globalen Variable, die vom FB L_MCMachineData beschrieben wird.



4 Strukturierte Datentypen

Dieses Kapitel enthält Informationen zu den strukturierten Datentypen, die in der Funktionsbibliothek "LenzeMotionControl1.Lib" definiert sind und für die Maschinenparameter sowie Positionierprofil-
daten verwendet werden.

4.1 MC_AXIS_REF

Die Datenstruktur MC_AXIS_REF wird für die Übermittlung der Maschinenparameter an weitere Mo-
tionControl-Funktionsblöcke verwendet.

Die folgende Tabelle beschreibt die in der Datenstruktur MC_AXIS_REF enthaltenen Elemente:

Element-Bezeichner	Datentyp	Info
dwMaxVelocity	Double Word	Kopie der max. Verfahrgeschwindigkeit in [unit/s] • 10000 = 1.0000 unit/s
dwMaxAcceleration	Double Word	Kopie der maximalen Beschleunigung in [unit/s ²] • 10000 = 1.0000 unit/s ²
dnFeedConstant	Double Integer	Kopie der Vorschubkonstante • Vorschub der Maschine in Einheiten pro Umdrehung der Lastseite des Getriebes. • 10000 = 1.0000 unit/s
wRatioNumerator	Word	Kopie Getriebefaktor Zähler
wRatioDenominator	Word	Kopie Getriebefaktor Nenner
wSensorRatioNumerator	Word	Kopie Lagegeber Getriebefaktor Zähler
wSensorRatioDenominator	Word	Kopie Lagegeber Getriebefaktor Nenner
dnIntMaxVelocity	Double Integer	Interne maximale Geschwindigkeit für den Profilgenerator • ⇒ Formel 1
dnIntMaxAcceleration	Double Integer	Interne maximale Beschleunigung für den Profilgenerator • ⇒ Formel 2
nVnorm	Integer	Interner Wert, um die Sollwertdrehzahl für den auf C0011 normierten Drehzahlregler zu erhalten (Formel 4) • nVnorm ⇒ Formel 3 • normierte Drehzahl ⇒ Formel 4
dnIntMaxJerk	Double Integer	Reserviert

Formel 1

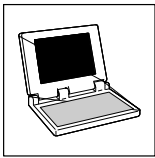
$$dnIntMaxVelocity = \frac{dwMaxVelocity}{10000} \cdot \frac{16384}{1000} \cdot PosResolution$$

Formel 2

$$dnIntMaxAcceleration = \frac{dwMaxAcceleration}{10000} \cdot \frac{16384}{1000000} \cdot PosResolution$$

Zwischenberechnung für Formel 1 und 2

$$PosResolution = 65535 \cdot \frac{wRatioNumerator}{wRatioDenominator} \cdot \frac{wSensorRatioDenominator}{wSensorRatioNumerator} \cdot \frac{10000}{dnFeedConst}$$



Funktionsbibliothek LenzeMotionControl1.lib

Strukturierte Datentypen

MC_AXIS_REF

Formel 3

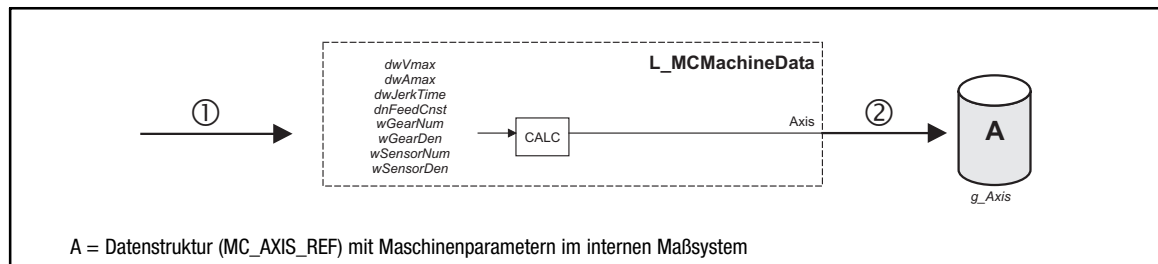
$$n_{\text{Vnorm}} = n_{\text{max}} \cdot \frac{65536}{60000} \cdot \frac{w_{\text{SensorRatioDenominator}}}{w_{\text{SensorRatioNumerator}}}$$

$$n_{\text{max}} = C0011$$

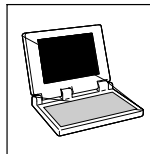
Formel 4

$$\text{Normierte Drehzahl für Drehzahlregler} = \frac{\text{aktuelle Verfahrgeschwindigkeit}}{n_{\text{Vnorm}}}$$

Typische Anwendung



- Die Maschinenparameter werden in den physikalischen Einheiten der Maschine vorgegeben. ①
- Die Umrechnung der Maschinenparameter in das interne Maßsystem des Profilgenerators erfolgt über den FB **L_MCMachineData**. (S. 2-4)
- Die umgerechneten Daten stehen am Ausgang *Axis* in Form der Datenstruktur MC_AXIS_REF zur Verfügung und werden zur Weitergabe an andere MotionControl-FB's üblicherweise einer globalen Variable (hier: *g_Axis*) zugewiesen. ②



4.2 MC_PROFILE_REAL_MEASURING

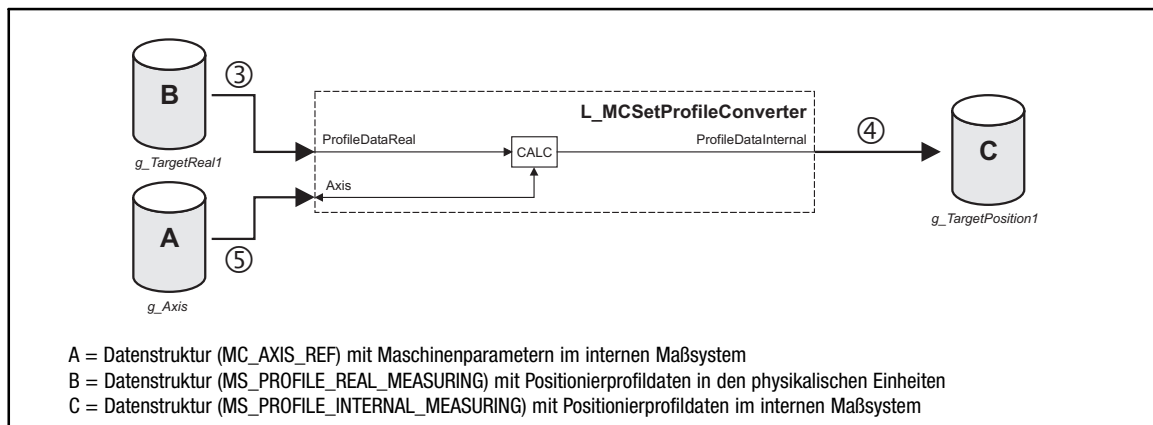
Die Datenstruktur MC_PROFILE_REAL_MEASURING beschreibt ein Positionierprofil in physikalischen Einheiten (Realmaßsystem).

Die folgende Tabelle beschreibt die in der Datenstruktur MC_PROFILE_REAL_MEASURING enthaltenen Elemente:

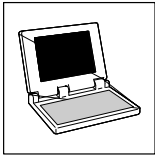
Element-Bezeichner	Datentyp	Info
dnTraversingVelocity	Double Integer	Verfahrgeschwindigkeit in [unit/s] • 10000 = 1.0000 unit/s
dnAcceleration	Double Integer	Beschleunigung in [unit/s ²] • 10000 = 1.0000 unit/s ²
dnDeceleration	Double Integer	Abbremsung in [unit/s ²] • 10000 = 1.0000 unit/s ²
dnPosition	Double Integer	Position in [unit] • 10000 = 1.0000 unit
dnFinalVelocity	Double Integer	Endgeschwindigkeit in [unit/s] • 10000 = 1.0000 unit/s
dnJerk	Double Integer	Reserviert

Typische Anwendung

Auch die Positionierprofile werden in den physikalischen Einheiten der Maschine vorgeben und mit den entsprechenden FB's der **LenzeMotionControl1.lib** in das interne Maßsystem umgerechnet.



- Dem FB **L_MCSetProfileConverter** können Sie über seinen Eingang *ProfileDataReal* ein komplettes Positionierprofil in physikalischen Einheiten in Form der Datenstruktur MC_PROFILE_REAL_MEASURING übergeben, um es in das interne Maßsystem zu konvertieren. ③ (3-6)
- Die umgerechneten Daten stehen am Ausgang *ProfileDataInternal* in Form der Datenstruktur MC_PROFILE_INTERNAL_MEASURING zur Verfügung und werden zur Weitergabe an andere MotionControl-FB's üblicherweise einer globalen Variable (hier: *g_TargetPosition1*) zugewiesen. ④
- Zur Umrechnung benötigt dieser FB über den Eingang *Axis* zusätzlich die Maschinenparameter im internen Maßsystem. ⑤



Funktionsbibliothek LenzeMotionControl1.lib

Strukturierte Datentypen

MC_PROFILE_INTERNAL_MEASURING

4.3 MC_PROFILE_INTERNAL_MEASURING

Die Datenstruktur MC_PROFILE_INTERNAL_MEASURING beschreibt ein Positionierprofil in internen Einheiten im internen Maßsystem, wie es die Profilgeneratoren benötigen.

Die folgende Tabelle beschreibt die in der Datenstruktur MC_PROFILE_INTERNAL_MEASURING enthaltenen Elemente:

Element-Bezeichner	Datentyp	Info
nTraversingVelocity	Integer	Verfahrgeschwindigkeit Wert stellt den relativen Bezug zur Maximalgeschwindigkeit dar, die über den FB L_MCMachineData definiert wurde ($16384/V_{max}$).
nAcceleration	Integer	Beschleunigung Wert stellt den relativen Bezug zur Maximalbeschleunigung dar, die über den FB L_MCMachineData definiert wurde ($16384/A_{max}$).
nDeceleration	Integer	Abbremsung Wert stellt den relativen Bezug zur Maximalbeschleunigung dar, die über den FB L_MCMachineData definiert wurde ($16384/A_{max}$).
dnPosition	Double Integer	Position in [inc] $65536 = 1$ Motorumdrehung ⇒ Formel 1
nFinalVelocity	Integer	Endgeschwindigkeit Wert stellt den relativen Bezug zur Maximalgeschwindigkeit dar, die über den FB L_MCMachineData definiert wurde ($16384/V_{max}$).
nJerk	Integer	Reserviert

Formel 1

$$dnPosition [inc] = \frac{dnPositionReal [unit]}{10000} \cdot \frac{65536 [inc/Umdr.]}{dnFeedConstant [unit/Umdr.]} \cdot \frac{wRatioNumerator}{wRatioDenominator} \cdot \frac{wSensorRatioDenominator}{wSensorRatioNumerator}$$

Typische Anwendung

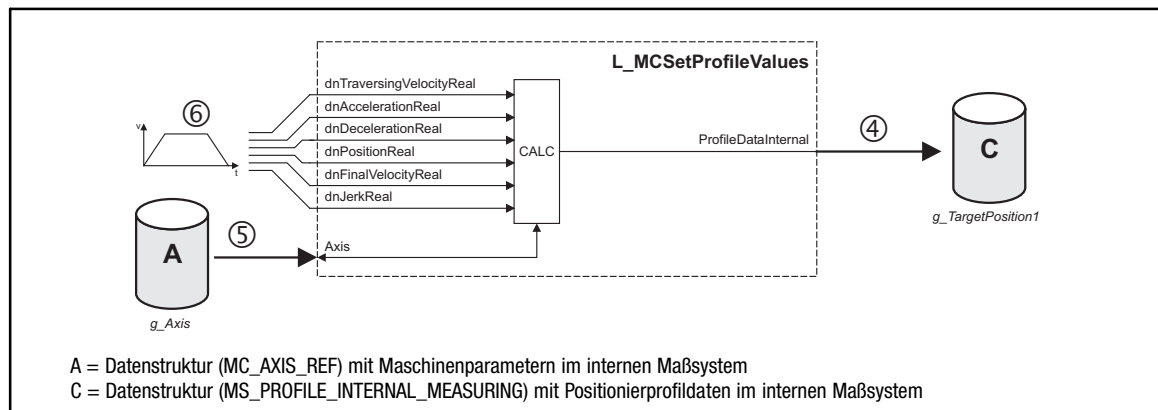
Die Positionierprofildaten werden in physikalischen Einheiten eingegeben und mit Hilfe des FB **L_MCSetProfileConverter** oder des FB **L_MCSetProfileValues** in das interne Maßsystem konvertiert.

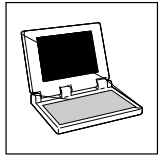


Tipp!

Im Gegensatz zum FB **L_MCSetProfileConverter**, der die Positionierprofildaten in den physikalischen Einheiten in Form der Datenstruktur MC_PROFILE_REAL_MEASURING erwartet, können Sie dem FB **L_MCSetProfileValues** die Daten als einzelne Werte übergeben.

- Typische Anwendung mit dem FB **L_MCSetProfileConverter** siehe Kap. 4.2. (4-3)





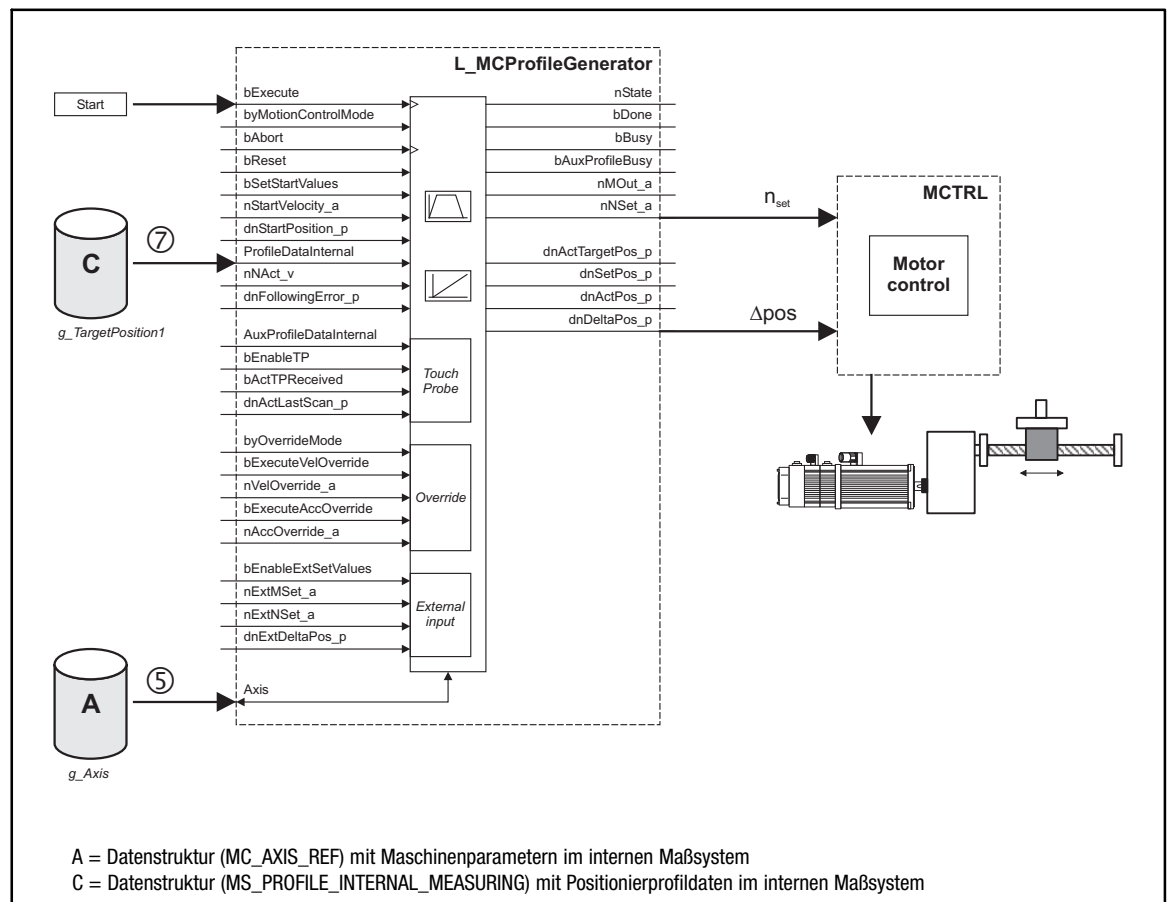
- Die umgerechneten Daten stehen am Ausgang *ProfileDataInternal* in Form der Datenstruktur *MC_PROFILE_INTERNAL_MEASURING* zur Verfügung und werden zur Weitergabe an andere MotionControl-FB's üblicherweise einer globalen Variable (hier: *g_TargetPosition1*) zugewiesen. ④
- Ebenso benötigt dieser FB über den Eingang *Axis* zusätzlich die Maschinenparameter im internen Maßsystem. ⑤



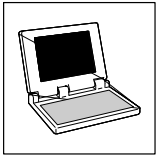
Tipp!

Informationen zu den Funktionsblöcken, mit denen physikalische Einheiten des Realmaßsystems für das interne Maßsystem des Profilgenerators aufbereitet werden können, finden Sie im Kap. 3, "Funktionsblöcke zur Umrechnung von Einheiten". (📖 3-1)

Übergabe der Daten an den Profilgenerator (hier: **L_MCPProfileGenerator**):



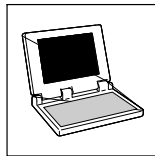
- Liegen die Positionierprofile im internen Maßsystem vor, können diese über die globale Variable (hier: *g_TargetPosition1*) dem Profilgenerator zur Weiterverarbeitung übergeben werden. ④
- Auch der Profilgenerator benötigt über den Eingang *Axis* zusätzlich die Maschinenparameter im internen Maßsystem. ⑤



Funktionsbibliothek LenzeMotionControl1.lib

Strukturierte Datentypen

MC_PROFILE_INTERNAL_MEASURING



5 Index

A

Anzeige der Sollposition auf Bediengeräten, 2-14

B

Begriffsdefinitionen, 1-2

E

Eingabe der Maschinenparameter, 2-6

F

Formeln zum Positionieren, 2-20

I

Interpolationsverfahren für Korrekturwertberechnung, 2-13

K

Korrekturwertberechnung, 2-13

L

L_MCAutoResetInterface, 2-2

L_MCGetPosition, 3-2

L_MCGetVelocity, 3-3

L_MCMachineData, 2-4

L_MCPositionCorrection, 2-10

L_MCProfileGenerator, 2-15

L_MCResidualDistance, 2-28

L_MCSetAcceleration, 3-4

L_MCSetDeceleration, 3-5

L_MCSetPosition, 3-9

L_MCSetProfileConverter, 3-6

L_MCSetProfileValues, 3-7

L_MCSetVelocity, 3-10

L_MCSwitchCam, 2-31

L_MCSwitchPoint, 2-35

L_MCTouchProbeCtrl, 2-38

M

Maschinenparameter (L_MCMachineData), 2-4

Beispiel, 2-7

Eingabe, 2-6

S-Profile, 2-8

Übergabe an andere FB, 3-1

Messwerttabelle, 2-11

O

Overchange, 2-27

Override, 2-23

P

Profilgenerator (L_MCProfileGenerator), 2-15

Formeln zum Positionieren, 2-20

Modi des Profilgenerators, 2-21

Modi für Restwegpositionierung, 2-23

Override, 2-23

Positionierprofil vorgeben, 2-20

Profiländerung während der Positionierung, 2-23

Profilverkettung mit Overchange, 2-27

Restwegpositionierung, 2-22

Übergabe der Maschinenparameter, 2-20

Profilverkettung mit Overchange, 2-27

R

Restwegberechnung, 2-28

Berechnung des kürzesten Restweges, 2-29

Definition des Fensters für die Meldung "Zielposition erreicht", 2-29

Formeln für die Berechnung, 2-30

Restwegpositionierung, 2-22

S

S-förmige Rampen (S-Profile), 2-8

Sicherheitshinweise, Gestaltung

Sonstige Hinweise, 1-1

Warnung vor Sachschäden, 1-1

T

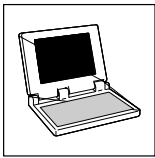
Touch-Probe Vorverarbeitung, 2-38

Aktivierung/Deaktivierung der Touch-Probe-Kanäle, 2-40

Ausgabe der Touch-Probe-Eingangssignale, 2-42

Meldeausgänge für erkannte Touch-Probe, 2-42

Positionsabhängige Freigabe eines Touch-Probe, 2-41



Funktionsbibliothek LenzeMotionControl1.lib

Index

U

unit, 1-2

V

Vorschubkorrektur, 2-10

W

Wegschaltpunkte (L_MCSwitchPoint), 2-35

Definition des Schaltbereichs, 2-32 , 2-36

Schalthysterese, 2-33 , 2-37

Totzeitkompensation, 2-33 , 2-36

Wegschaltpunkte Rundachse (L_MCSwitchCam), 2-31