

Technologiemodul



Basic Motion

Referenzhandbuch

DE

1	Über diese Dokumentation	3
1.1	Dokumenthistorie	5
1.2	Verwendete Konventionen	6
1.3	Gestaltung der Sicherheitshinweise	7
2	Sicherheitshinweise	8
3	Übersicht der Funktionen des Technologiemoduls	10
3.1	Wichtige Hinweise zum Betrieb des Technologiemoduls	11
4	Funktionsbaustein L_TB2P_BasicMotionBase	12
4.1	Eingänge	13
4.2	Ausgänge	14
4.3	L_TB2P_scCtrlBasicMotion	15
4.4	L_TB2P_scStatusBasicMotion	17
4.5	Parameter	19
5	State Machine	20
6	Beschreibung der Basisfunktionen	21
6.1	Wichtige Hinweise zum Betrieb des Technologiemoduls	21
6.2	Handfahren (Jogging)	22
6.3	Referenzfahrt (Homing)	24
6.4	QSP Application	27
6.5	Halt	28
6.6	Absolute Positionierung	29
6.7	Relative Positionierung	31
6.8	Endlosfahrt mit definierter Geschwindigkeit	34
6.9	Abbruch und Beendigung einer Positionierung	37
6.10	Override für Geschwindigkeit	40
	Index	42
	Ihre Meinung ist uns wichtig	43

1 Über diese Dokumentation

Diese Dokumentation ...

- enthält ausführliche Informationen zu den Funktionalitäten des Technologiemoduls "Basic Motion"
- ordnet sich in die Handbuchsammlung "Controller-based Automation" ein. Diese besteht aus folgenden Dokumentationen:

Dokumentationstyp	Thema
Produktkatalog	Controller-based Automation (Systemübersicht, Beispieltopologien) Lenze-Controller (Produktinformationen, Technische Daten)
Systemhandbücher	Visualisierung (Systemübersicht/Beispieltopologien)
Kommunikationshandbücher Online-Hilfen	Bussysteme • Controller-based Automation EtherCAT® • Controller-based Automation CANopen® • Controller-based Automation PROFIBUS® • Controller-based Automation PROFINET®
Referenzhandbücher Online-Hilfen	Lenze-Controller: • Controller 3200 C • Controller c300 • Controller p300 • Controller p500
Software-Handbücher Online-Hilfen	Lenze Engineering Tools: • »PLC Designer« (Programmierung) • »Engineer« (Parametrierung, Konfigurierung, Diagnose) • »VisiWinNET® Smart« (Visualisierung) • »Backup & Restore« (Datensicherung, Wiederherstellung, Aktualisierung)

Weitere Technische Dokumentationen zu Lenze-Produkten

Weitere Informationen zu Lenze-Produkten, die in Verbindung mit der Controller-based Automation verwendbar sind, finden Sie in folgenden Dokumentationen:

Planung / Projektierung / Technische Daten	
☐	Produktkataloge • Controller-based Automation • Controller • Inverter Drives/Servo Drives
Montage und Verdrahtung	
☐	Montageanleitungen • Controller • Kommunikationskarten (MC-xxx) • I/O-System 1000 (EPM-Sxxx) • Inverter Drives/Servo Drives • Kommunikationsmodule
☐	Gerätehandbücher • Inverter Drives/Servo Drives
Parametrierung / Konfigurierung / Inbetriebnahme	
☐	Online-Hilfe / Referenzhandbücher • Controller • Inverter Drives/Servo Drives • I/O-System 1000 (EPM-Sxxx)
☐	Online-Hilfe / Kommunikationshandbücher • Bussysteme • Kommunikationsmodule
Beispielapplikationen und Vorlagen	
☐	Online-Hilfe / Software- und Referenzhandbücher • Application Sample i700 • Application Samples 8400/9400 • FAST Application Template Lenze/PackML • FAST Technologiemodule

Symbole:

-  Gedruckte Dokumentation
- ☐ PDF-Datei / Online-Hilfe im Lenze Engineering Tool



Tipp!

Aktuelle Dokumentationen und Software-Updates zu Lenze-Produkten finden Sie im Download-Bereich unter:

www.lenze.com

Zielgruppe

Diese Dokumentation richtet sich an alle Personen, die ein Lenze-Automatisierungssystem auf Basis der Lenze FAST Application Software programmieren und in Betrieb nehmen.

1 Über diese Dokumentation

1.1 Dokumenthistorie

1.1 Dokumenthistorie

Version			Beschreibung
4.0	04/2021	TD06	Fehlerkorrekturen & Ergänzungen
3.0	04/2020	TD17	Fehlerkorrekturen & Ergänzungen
2.0	11/2019	TD06	Fehlerkorrekturen & Ergänzungen
1.0	04/2019	TD06	Erstausgabe

1.2 Verwendete Konventionen

Diese Dokumentation verwendet folgende Konventionen zur Unterscheidung verschiedener Arten von Information:

Informationsart	Auszeichnung	Beispiele/Hinweise
Zahlenschreibweise		
Dezimaltrennzeichen	Punkt	Es wird generell der Dezimalpunkt verwendet. Zum Beispiel: 1234.56
Hexadezimalzahl	0x	Für Hexadezimalzahlen wird der Präfix "0x" verwendet. Beispiel: 0x60F4
Textauszeichnung		
Programmname	» «	»PLC Designer« ...
Variablenbezeichner	<i>kursiv</i>	Durch Setzen von <i>bEnable</i> auf TRUE ...
Funktionsbausteine	fett	Der Funktionsbaustein L_MC1P_AxisBasicControl ...
Funktionsbibliotheken		Die Funktionsbibliothek L_TT1P_TechnologyModules ...
Quellcode	Schriftart "Courier new"	... dwNumerator := 1; dwDenominator := 1; ...
Hyperlink	<u>unterstrichen</u>	Optisch hervorgehobener Verweis auf ein anderes Thema. Wird in dieser Online-Dokumentation per Mausklick aktiviert.
Symbole		
Seitenverweis	 6	Optisch hervorgehobener Verweis auf eine andere Seite. Wird in dieser Online-Dokumentation per Mausklick aktiviert.
Schrittweise Anleitung		Schrittweise Anleitungen sind durch ein Piktogramm gekennzeichnet.

Variablenbezeichner

Die von Lenze verwendeten Konventionen, die für die Variablenbezeichner von Lenze Systembausteinen, Funktionsbausteinen sowie Funktionen verwendet werden, basieren auf der sogenannten "Ungarischen Notation", wodurch anhand des Bezeichners sofort auf die wichtigsten Eigenschaften (z. B. den Datentyp) der entsprechenden Variable geschlossen werden kann, z. B. *xAxisEnabled*.

1.3

Gestaltung der Sicherheitshinweise

⚠ GEFAHR!

Kennzeichnet eine außergewöhnlich große Gefahrensituation. Wird dieser Hinweis nicht beachtet, kommt es zu schweren irreversiblen Verletzungen oder zum Tod.

⚠ WARNUNG!

Kennzeichnet eine außergewöhnlich große Gefahrensituation. Wird dieser Hinweis nicht beachtet, kann es zu schweren irreversiblen oder tödlichen Verletzungen kommen.

⚠ VORSICHT!

Kennzeichnet eine Gefahrensituation. Wird dieser Hinweis nicht beachtet, kann es zu leichten oder mittleren Verletzungen kommen.

HINWEIS

Kennzeichnet Sachgefahren. Wird dieser Hinweis nicht beachtet, kann es zu Sachschäden kommen.

Anwendungshinweise

Piktogramm	Signalwort	Bedeutung
	Tipp!	Nützlicher Tipp zur einfachen Bedienung
		Verweis auf andere Dokumentation

2 Sicherheitshinweise

Beachten Sie die Sicherheitshinweise in dieser Dokumentation, wenn Sie ein Automatisierungssystem oder eine Anlage mit einem Lenze-Controller in Betrieb nehmen möchten.



Die Gerätedokumentation enthält Sicherheitshinweise, die Sie beachten müssen!

Lesen Sie die mitgelieferten und zugehörigen Dokumentationen der jeweiligen Komponenten des Automatisierungssystems sorgfältig durch, bevor Sie mit der Inbetriebnahme des Controllers und der angeschlossenen Geräte beginnen.

⚠ GEFAHR!

Hohe elektrische Spannung

Personenschäden durch gefährliche elektrische Spannung

Mögliche Folgen

Tod oder schwere Verletzungen

Schutzmaßnahmen

Die Spannungsversorgung ausschalten, bevor Arbeiten an den Komponenten des Automatisierungssystems durchgeführt werden.

Nach dem Ausschalten der Spannungsversorgung spannungsführende Geräteteile und Leistungsanschlüsse nicht sofort berühren, weil Kondensatoren aufgeladen sein können.

Die entsprechenden Hinweisschilder auf dem Gerät beachten.

⚠ GEFAHR!

Personenschäden

Verletzungsgefahr besteht durch ...

- nicht vorhersehbare Motorbewegungen (z. B. ungewollte Drehrichtung, zu hohe Geschwindigkeit oder ruckhafter Lauf);
- unzulässige Betriebszustände bei der Parametrierung, während eine Online-Verbindung zum Gerät besteht.

Mögliche Folgen

Tod oder schwere Verletzungen

Schutzmaßnahmen

- Anlagen mit eingebauten Umrichtern ggf. mit zusätzlichen Überwachungs- und Schutzeinrichtungen nach den jeweils gültigen Sicherheitsbestimmungen ausrüsten (z. B. Gesetz über technische Arbeitsmittel, Unfallverhütungsvorschriften).
- Während der Inbetriebnahme einen ausreichenden Sicherheitsabstand zum Motor oder den vom Motor angetriebenen Maschinenteilen einhalten.

HINWEIS

Beschädigung oder Zerstörung von Maschinenteilen

Beschädigung oder Zerstörung von Maschinenteilen besteht durch ...

- Kurzschluss oder statische Entladungen (ESD);
- nicht vorhersehbare Motorbewegungen (z. B. ungewollte Drehrichtung, zu hohe Geschwindigkeit oder ruckhafter Lauf);
- unzulässige Betriebszustände bei der Parametrierung, während eine Online-Verbindung zum Gerät besteht.

Schutzmaßnahmen

- Vor allen Arbeiten an den Komponenten des Automatisierungssystems immer die Spannungsversorgung ausschalten.
 - Elektronische Bauelemente und Kontakte nur berühren, wenn zuvor ESD-Maßnahmen getroffen wurden.
 - Anlagen mit eingebauten Umrichtern ggf. mit zusätzlichen Überwachungs- und Schutzeinrichtungen nach den jeweils gültigen Sicherheitsbestimmungen ausrüsten (z. B. Gesetz über technische Arbeitsmittel, Unfallverhütungsvorschriften).
-

3 Übersicht der Funktionen des Technologiemoduls

3 Übersicht der Funktionen des Technologiemoduls

Funktionalität	Beschreibung
Handfahren (Logging) (📖 22)	Achse im Handbetrieb verfahren
Referenzfahrt (Homing) (📖 24)	Achse referenzieren
OSP Application (📖 27)	Bewegung über eine Applikationsschnellhalt-Verzögerungsrampe (AXIS_REF) stoppen
Halt (📖 28)	Bewegung über eine Halt-Verzögerungsrampe stoppen
Absolute Positionierung (📖 29)	Achse auf eine eindeutige absolute Position fahren
Relative Positionierung (📖 31)	Achse auf eine zur aktuellen Position relativen Zielposition fahren
Endlosfahrt mit definierter Geschwindigkeit (📖 34)	Endlosfahrt durchführen
Abbruch und Beendigung einer Positionierung (📖 37)	Positionierung beenden
Override für Geschwindigkeit (📖 40)	Overrides für Geschwindigkeit vorgeben

Desweiteren kann das Technologiemodul auch zur Zustands- und Fehleranzeige sowie zum Zurücksetzen eines Fehlers verwendet werden.

3.1 Wichtige Hinweise zum Betrieb des Technologiemoduls

Anlauf der Achsen

Der Anlauf der Achsen nach gesperrtem Regler hängt von der beauftragten Bewegungsart ab.

Unterschieden wird zwischen der Bewegungsart mit pegelaktivierter Funktion und der Bewegungsart mit flankenaktivierter Funktion.

- Die Bewegungsart mit pegelaktivierter Funktion wird bei Reglerfreigabe automatisch aktiv, solange keine höherpriorie Funktion aktiv ist. Zu dieser Bewegungsart gehören
 - [Handfahren \(Jogging\)](#) (📖 39)
 - [QSP Application](#) (📖 45)
- Die Bewegungsart mit flankenaktivierter Funktion muss nach Reglerfreigabe mit einer FALSE→TRUE-Flanke aktiviert werden. Zu dieser Bewegungsart gehört
 - [Referenzfahrt \(Homing\)](#) (📖 43)



Beispiel zu einer Bewegungsart mit pegelaktivierter Funktion: Handfahren (Jogging)

1. Im gesperrten Achszustand wird der Eingang $xJogPos = TRUE$ gesetzt.
 - Eingang $xEnableOperation = FALSE$ (Achse ist gesperrt.)
==> Zustand "STANDBY" (Ausgang $xOperationEnabled = FALSE$)
 - Eingang $xJogPos = TRUE$ (Handfahren soll ausgeführt werden.)
2. Achse freigeben.
 - Eingang $xEnableOperation = TRUE$
==> Zustand "JOGGING" ($xOperationEnabled = TRUE$)



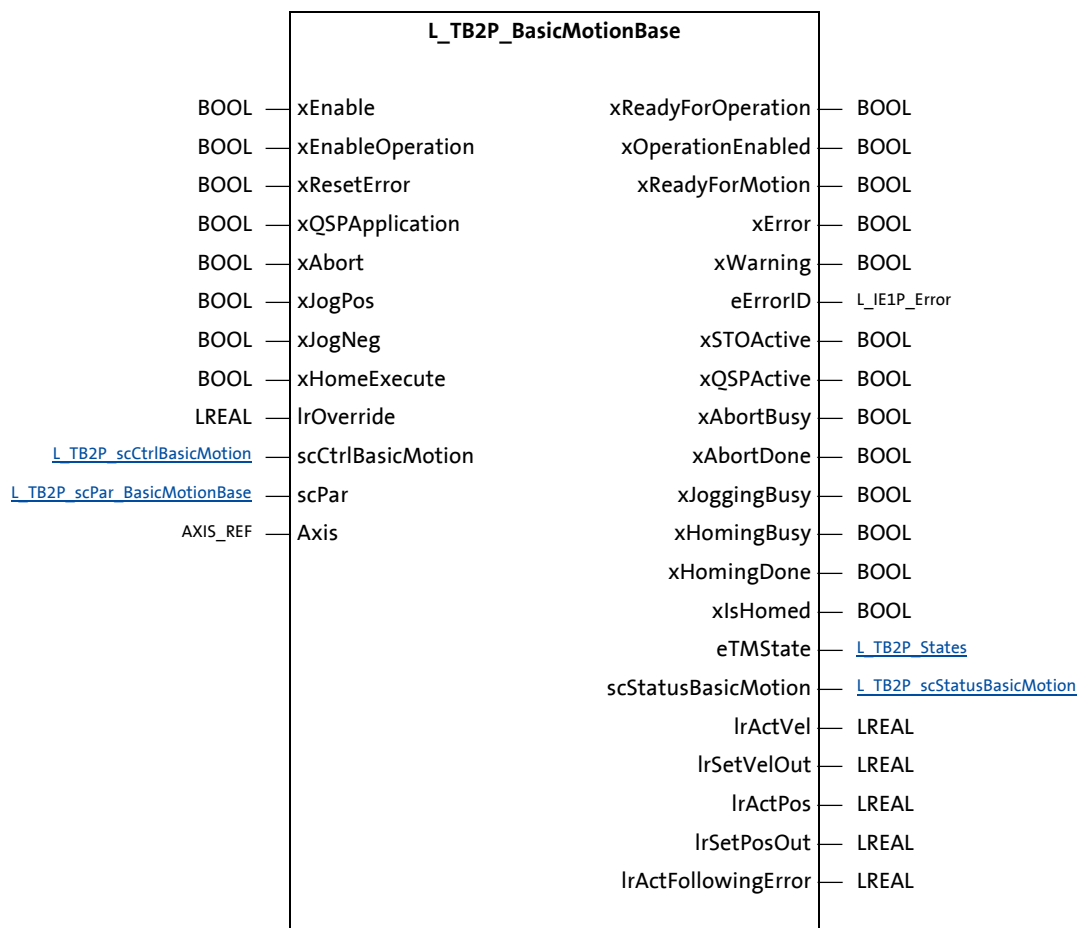
Beispiel zu einer Bewegungsart mit flankenaktivierter Funktion: Referenzfahrt (Homing)

1. Im gesperrten Achszustand wird der Eingang $xHomeExecute = TRUE$ gesetzt.
 - Eingang $xEnableOperation = FALSE$ (Achse ist gesperrt.)
==> Zustand "STANDBY" (Ausgang $xOperationEnabled = FALSE$)
 - Eingang $xHomeExecute = TRUE$ (Referenzierung soll ausgeführt werden.)
2. Achse freigeben.
 - Eingang $xEnableOperation = TRUE$
==> Zustand "STANDBY" (Ausgang $xOperationEnabled = TRUE$)
3. Referenzierung ausführen.
 - Eingang $xHomeExecute = FALSE \rightarrow TRUE$
==> Zustand "HOMING"

4

Funktionsbaustein L_TB2P_BasicMotionBase

Die Abbildung zeigt die Zugehörigkeit der Ein- und Ausgänge des Funktionsbausteins.



4.1

Eingänge

Bezeichner	Datentyp	Beschreibung	
xEnable	BOOL		Ausführung des Funktionsbausteins
		TRUE	Der Funktionsbaustein wird ausgeführt.
		FALSE	Der Funktionsbaustein wird nicht ausgeführt.
xEnableOperation	BOOL	TRUE	Die Achse wird betriebsbereit geschaltet. Die Reglerfreigabe wird erteilt.
		FALSE	
xResetError	BOOL		Der Eingang ist flankengesteuert und wertet die steigende Flanke aus.
		FALSE	Fehler der Achse oder der Software zurücksetzen.
		TRUE	
xQSPApplication	BOOL		Der Eingang ist pegelgesteuert und wertet den High-Pegel aus.
		TRUE	Aktive Bewegung wird abgebrochen und Achse wird über die QSP Application (□ 27) Stopprampe in den Stillstand geführt. - Ein Wechsel in den Zustand "STOP" erfolgt. - Das Technologiemodul bleibt im Zustand "STOP", solange <i>xQSPApplication</i> = TRUE gesetzt ist.
xAbort	BOOL	TRUE	Aktive Bewegung wird abgebrochen und Achse wird über die Profilverzögerung (<i>scPar.IrDec</i>) in den Stillstand geführt. Ein Wechsel in den Zustand "STOP" erfolgt. Die Verzögerungsrampe kann durch eine Beauftragung einer neuen Bewegung (z. B. <i>xMoveVelExecute</i>) abgebrochen werden. ▶ Abbruch und Beendigung einer Positionierung (□ 37)
xJogPos	BOOL		Der Eingang ist pegelgesteuert und wertet den High-Pegel aus. ▶ Handfahren (Jogging) (□ 22)
		TRUE	Achse in positive Richtung fahren (Handfahren). Ist <i>xJogNeg</i> auch TRUE, wird über den Ausgang <i>eErrorID</i> die Fehlermeldung "InvalidInputJogPosJogNeg" ausgegeben und die Achse stoppt.
xJogNeg	BOOL		Der Eingang ist pegelgesteuert und wertet den High-Pegel aus. ▶ Handfahren (Jogging) (□ 22)
		TRUE	Achse in negative Richtung fahren (Handfahren). Ist <i>xJogPos</i> auch TRUE, wird über den Ausgang <i>eErrorID</i> die Fehlermeldung "InvalidInputJogPosJogNeg" ausgegeben und die Achse stoppt.
xHomeExecute	BOOL		Der Eingang ist flankengesteuert und wertet die steigende Flanke aus. ▶ Referenzfahrt (Homing) (□ 24)
		FALSE	Referenzierung starten.
		TRUE	Der Abbruch der Funktion erfolgt über den Eingang <i>xHalt</i> .
IrOverride	LREAL		Mit dem Faktor <i>IrOverride</i> wird die vorgegebene Sollgeschwindigkeit multipliziert (z. B. ein Override-Wert von 0.5 erzeugt die Hälfte der Sollgeschwindigkeit) - Wertebereich: 0 ... 1.0 - Initialwert: 1.0 Hinweis: Die Übernahme des Wertes ist unabhängig vom Status in <i>xContinuousUpdate</i> .
scCtrlBasicMotion			Struktureingang zur Ansteuerung von Basic-Motion-Funktionen L_TB2P_scCtrlBasicMotion
scPar			Die Parameterstruktur enthält die Parameter des Technologiemoduls. L_TB2P_scPar_BasicMotionBase
Axis	AXIS_REF		Referenz auf die Achse

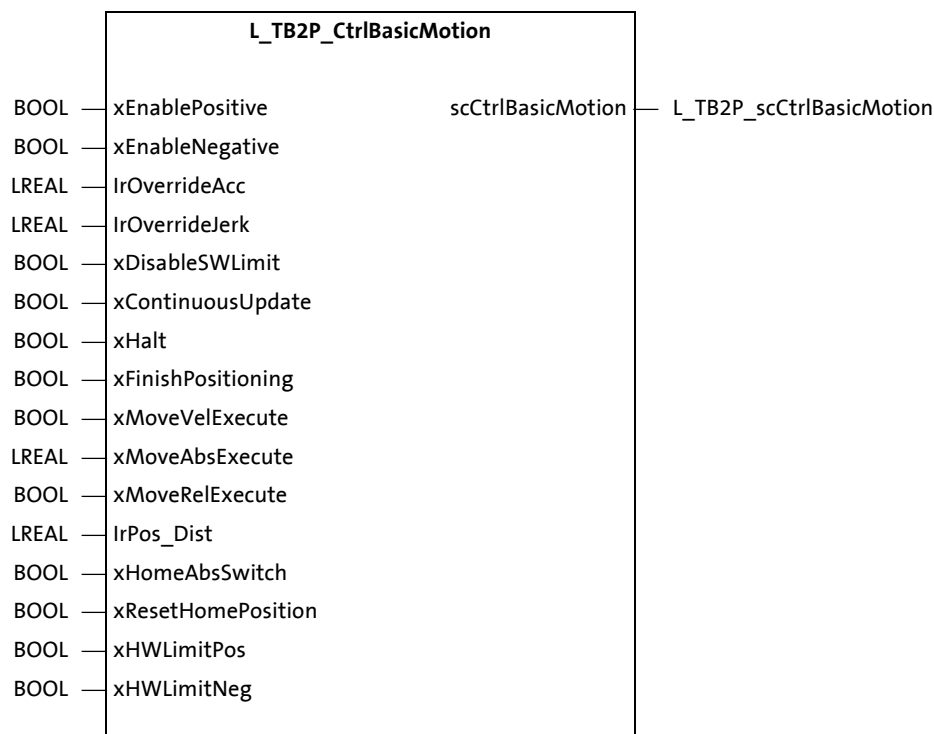
4.2

Ausgänge

Bezeichner	Datentyp	Beschreibung
xReadyForOperation	BOOL	Die Achse ist fehlerfrei. Die Zwischenkreisspannung ist vorhanden. "Safe Torque Off" (STO) ist nicht aktiv.
xOperationEnabled	BOOL	Die Achse ist freigegeben. Die Bremse ist gelüftet.
xReadyForMotion	BOOL	Die Achse ist freigegeben und fehlerfrei. Die Bremse ist gelüftet.
xError	BOOL	TRUE Im Technologiemodul liegt ein Fehler vor.
xWarning	BOOL	TRUE Im Technologiemodul liegt eine Warnung vor.
eErrorID	L_IE1P_Error	ID der Fehler- oder Warnungsmeldung, wenn <i>xError</i> = TRUE oder <i>xWarning</i> = TRUE ist. Referenzhandbuch "FAST & Technologiemodule": Hier finden Sie Informationen zu Fehler- oder Warnungsmeldungen.
xSTOActive	BOOL	TRUE Ein "Safe Torque Off" (STO) wurde ausgelöst und ist aktiv. Die Achse ist drehmomentfrei und der Regler ist gesperrt.
xQSPActive	BOOL	TRUE Der Eingang <i>xQSPApplication</i> wurde auf TRUE gesetzt. Das Technologiemodul befindet sich im Zustand "STOP", die Achse wird in den Stillstand geführt. Der Ausgang wird zurückgesetzt, sobald sich die Achse im Stillstand befindet.
xAbortBusy	BOOL	TRUE Die aktivierte Funktion wird durch den Eingang <i>xAbsort</i> abgebrochen.
xAbsortDone	BOOL	TRUE Die aktivierte Funktion wurde durch den Eingang <i>xAbsort</i> abgebrochen.
xJoggingBusy	BOOL	TRUE Die Funktion Handfahren ist aktiv. Das Technologiemodul befindet sich im Zustand "JOGGING".
xHomingBusy	BOOL	TRUE Die Referenzfahrt wird ausgeführt. Das Technologiemodul befindet sich im Zustand "HOMING".
xHomingDone	BOOL	TRUE Die Referenzfahrt wurde erfolgreich abgeschlossen.
xIsHomed	BOOL	TRUE Die Achse ist referenziert ("Referenz bekannt"). <i>xIsHomed</i> darf erst ausgewertet werden, wenn der Status <i>xAxisAvailable</i> gemeldet wird.
eTMState	L_TB2P_States	Aktueller Zustand des Technologiemoduls
scStatusBasicMotion	L_TB2P_scStatusBasicMotion	Struktur der Zustandsdaten des Funktionsbausteins
IrActVel	LREAL	Aktuelle Geschwindigkeit • Einheit: units/s
IrSetVelOut	LREAL	Sollgeschwindigkeit • Einheit: units/s
IrActPos	LREAL	Aktuelle Position • Einheit: units
IrSetPosOut	LREAL	Sollposition • Einheit: units
IrActFollowingError	LREAL	Aktueller Schleppabstand • Einheit: units

4.3

L_TB2P_scCtrlBasicMotion



Bezeichner	Datentyp	Beschreibung	
xEnablePositive	BOOL	TRUE	Verfahren in positive Richtung freigeben. • Initialwert: TRUE
xEnableNegative	BOOL	TRUE	Verfahren in negative Richtung freigeben. • Initialwert: TRUE
IrOverrideAcc	LREAL	Faktor für den globalen Override der Beschleunigung • Wertebereich: 0.1 ... 1.0 • Initialwert: 1.0 (100 %)	
IrOverrideJerk	LREAL	Faktor für den globalen Override des Rucks • Wertebereich: 0.1 ... 1.0 • Initialwert: 1.0 (100 %)	
xDisableSWLimit	BOOL	TRUE	Software-Endlagen deaktivieren. Dieser Eingang übersteuert die in der Achse (Subindex 50) parametrisierte Einstellung und deaktiviert die Software-Endlagenüberwachung.
		FALSE	Die Einstellung der Achse ist wirksam.
xContinuousUpdate	BOOL	TRUE	Profilparameter werden zyklisch auf Änderung überprüft und übernommen. • Initialwert: TRUE
xHalt	BOOL	TRUE	Die aktive Bewegung wird abgebrochen und die Achse wird über die Haltrampe in den Stillstand geführt. • Ein Wechsel in den Zustand "STOP" erfolgt. Die Verzögerungsrampe kann durch die Beauftragung einer neuen Bewegung (z. B. <i>xMoveVelExecute</i>) abgebrochen werden. ▶ Halt (□ 28)

Bezeichner	Datentyp	Beschreibung	
xFinishPositioning	BOOL	FALSE TRUE	Eine unterbrochene Positionierung wird fortgesetzt. • Eine zuvor über <i>xAbort</i> unterbrochene Positionierung wird zu Ende geführt. • Bereits zurückgelegte Distanzen einer relativen Positionierung werden berücksichtigt. ► Abbruch und Beendigung einer Positionierung (📖 37)
xMoveVelExecute	BOOL	FALSE TRUE	Endlosfahrt mit definierter Geschwindigkeit (MoveVelocity) starten. ► Endlosfahrt mit definierter Geschwindigkeit (📖 34)
xMoveAbsExecute	BOOL	FALSE TRUE	Absolute Positionierung (MoveAbsolute) starten. ► Absolute Positionierung (📖 29)
xMoveRelExecute	BOOL	FALSE TRUE	Relative Positionierung (MoveRelative) starten. ► Relative Positionierung (📖 31)
lrPos_Dist	LREAL		Absolute Positionierung Absolute Zielposition bezogen auf die Nullposition der Achse. • Einheit: units • Initialwert: 0
			Relative Positionierung Zu verfahrenende Wegstrecke bezogen auf die Sollposition zum Startzeitpunkt des Kommandos. • Einheit: units • Initialwert: 0
xHomeAbsSwitch	BOOL	TRUE	Anschluss für Referenzschalter Bei Referenzfahrmodi mit Referenzschalter verbinden Sie diesen Eingang mit dem Digitalsignal, das den Zustand des Referenzschalters wiedergibt.
xResetHomePosition	BOOL	TRUE	Statusinformation "Referenzposition bekannt" (Ausgang <i>xisHomed</i>) zurücksetzen.
xHWLimitPos	BOOL		Positiver Hardware-Endschalter Verbinden Sie diesen Eingang mit dem entsprechenden Digitaleingang, an dem der Endschalter angeschlossen ist.
		TRUE	Der positive Hardware-Endschalter wurde erreicht oder angefahren. • Der Strukturausgang <i>xHwLimitSwitchPos</i> wird ebenfalls auf TRUE gesetzt. • Die Achse wird mit der Verzögerung des QSP Application (📖 27) in den Stillstand geführt. • Es wird in den Zustand "ERROR" gewechselt mit der Fehlermeldung "HWLimitPos" (<i>eErrorID</i> = 20500).
xHWLimitNeg	BOOL		Negativer Hardware-Endschalter Verbinden Sie diesen Eingang mit dem entsprechenden Digitaleingang, an dem der Endschalter angeschlossen ist.
		TRUE	Der negative Hardware-Endschalter wurde erreicht oder angefahren. • Der Strukturausgang <i>xHwLimitSwitchNeg</i> wird ebenfalls auf TRUE gesetzt. • Die Achse wird mit der Verzögerung des QSP Application (📖 27) in den Stillstand geführt. • Es wird in den Zustand "ERROR" gewechselt mit der Fehlermeldung "HWLimitNeg" (<i>eErrorID</i> = 20501).

4.4

L_TB2P_scStatusBasicMotion



Bezeichner	Datentyp	Beschreibung	
xFunBusy	BOOL	TRUE	Abarbeitung der aktivierten Funktion läuft. Die Zwischenkreisspannung ist vorhanden.
xFunDone	BOOL	TRUE	Abarbeitung der aktivierten Funktion erfolgreich beendet.
xFunCommandAborted	BOOL	TRUE	Die aktivierte Funktion wurde durch einen anderen Funktionsbaustein abgebrochen.
xHaltBusy	BOOL	TRUE	Die Haltfunktion wird ausgeführt
xHaltDone	BOOL	TRUE	Die Haltfunktion wurde erfolgreich abgeschlossen.
eControlMode L_MC1P_ControlMode		Aktueller Regelungsmodus der Achse	
xHwLimitSwitchNeg	BOOL	TRUE	Negativer Hardware-Endschalter überschritten.
xHwLimitSwitchPos	BOOL	TRUE	Positiver Hardware-Endschalter überschritten.
xSwLimitEnabled	BOOL	TRUE	Überwachung der Software-Endlagen ist aktiviert.

Bezeichner	Datentyp	Beschreibung	
xSwLimitSwitchNeg	BOOL	TRUE	Negative Software-Endlage überschritten.
xSwLimitSwitchPos	BOOL	TRUE	Positive Software-Endlage überschritten.
eAxisType	L_MC1P_DriveType	Verbundener Achstyp	
eAxisState	L_MC1P_AXIS_STATE	Aktueller PLCopen-Zustand der Achse	
xAxisWarning	BOOL	TRUE	In den Achsdaten oder einem angeschlossenen Funktionsbaustein liegt eine Warnung vor.
xAxisError	BOOL	TRUE	Der Ausgang ist TRUE bei: • Fehler in den Achsdaten • Fehler eines angeschlossenen Funktionsbausteins • Fehlermeldungen des Funktionsbausteins am Ausgang <i>eErrorID</i>
xDriveWarning	BOOL	TRUE	Im Inverter liegt eine Warnung vor.
xDriveError	BOOL	TRUE	Im Inverter liegt ein Fehler vor.
dwDriveErrorID	DWORD	TRUE	Wenn <i>xDriveError</i> = TRUE oder <i>xDriveWarning</i> = TRUE: Fehlernummer zum aufgetretenen Fehler im Inverter.
xConstantVelocity	BOOL	TRUE	Die vorgegebene Geschwindigkeit wurde das erste Mal erreicht.
xAccelerating	BOOL	TRUE	Die Achse wird beschleunigt. (Betragsmäßige Erhöhung der Geschwindigkeit)
xDecelerating	BOOL	TRUE	Die Achse wird verzögert oder abgebremst. (Betragsmäßige Verringerung der Geschwindigkeit)
xDirectionPositive	BOOL	TRUE	Die Achse verfährt in positive Richtung.
xDirectionNegative	BOOL	TRUE	Die Achse verfährt in negative Richtung.
xInPosition	BOOL	TRUE	Die Zielposition wurde erreicht.
xAxisAvailable	BOOL	TRUE	Die Achse kann in der Applikation verwendet werden, d. h. die Initialisierung wurde fehlerfrei beendet. Gültige Istwerte sind verfügbar und die Kommunikation zu dem unterlagerten Antrieb ist vorhanden.
xStandstillSetVelocity	BOOL	TRUE	Die Sollgeschwindigkeit der Achse ist Null.
xStandstillActVelocity	BOOL	TRUE	Die Istgeschwindigkeit liegt innerhalb des definierten Fensters (<i>Window standstill</i> , Subindex 132)

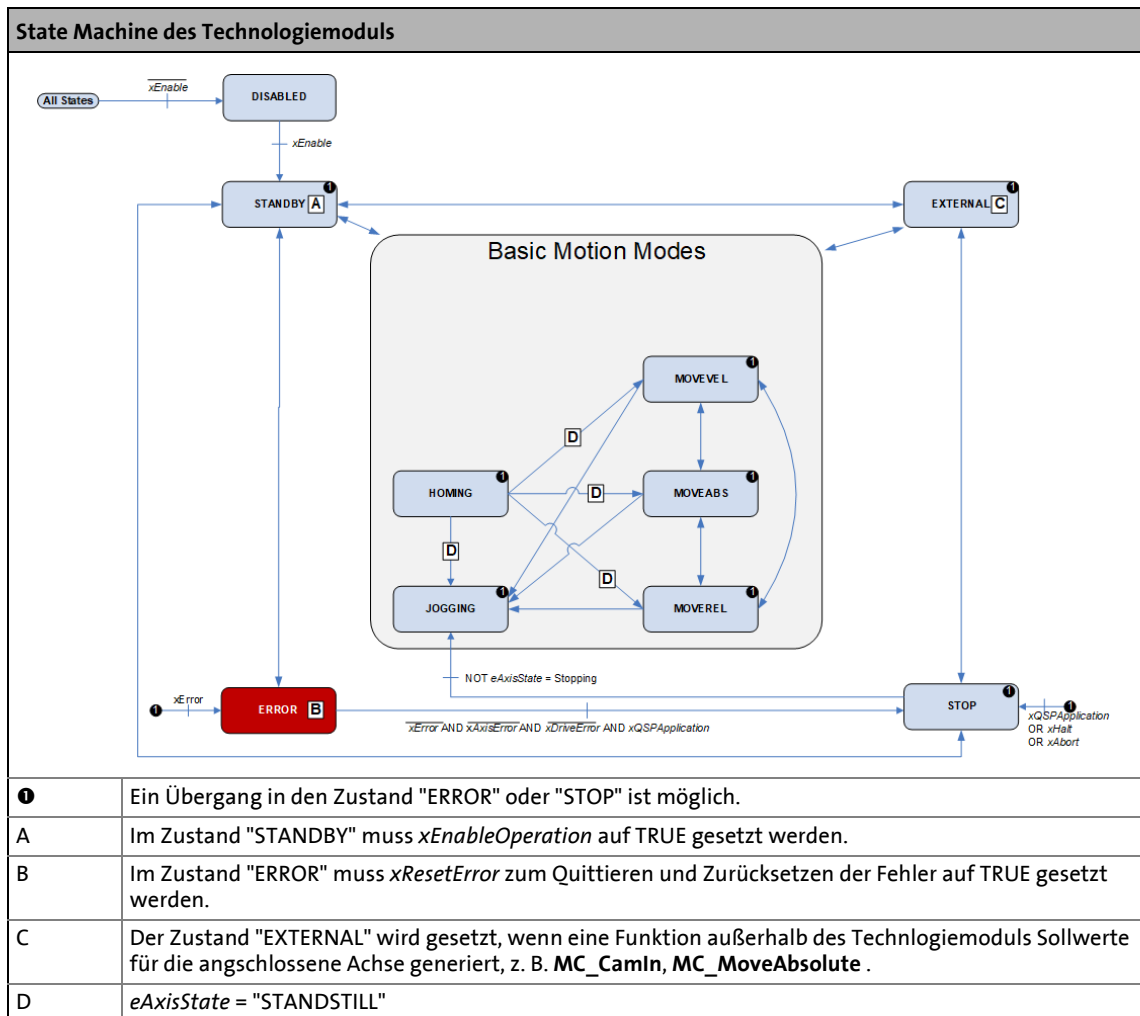
4.5

Parameter

L_TB2P_scPar_BasicMotionBase

Bezeichner	Datentyp	Beschreibung
IrVel	LREAL	Geschwindigkeit für Bewegungen (MoveAbs, MoveRel, MoveVel), die über die Struktur scCtrlBasicMotion beauftragt werden. - Einheit: units/s - Initialwert: 3600
IrAcc	LREAL	Beschleunigung für Bewegungen (MoveAbs, MoveRel, MoveVel), die über die Struktur scCtrlBasicMotion beauftragt werden. Vorgabe, mit welcher Geschwindigkeitsänderung maximal beschleunigt werden soll. - Einheit: units/s² - Initialwert: 36000
IrDec	LREAL	Verzögerung für Bewegungen (MoveAbs, MoveRel, MoveVel), die über die Struktur scCtrlBasicMotion beauftragt werden. Vorgabe, mit welcher Geschwindigkeitsänderung maximal bis zum Stillstand verzögert werden soll. - Einheit: units/s² - Initialwert: 100
IrJerk	LREAL	Ruck für Bewegungen (MoveAbs, MoveRel, MoveVel), die über die Struktur scCtrlBasicMotion beauftragt werden. - Einheit: units/s³ - Initialwert: 0
eDirection	MC_DIRECTION	Verfahrrichtung für Bewegungen (MoveAbs, MoveVel), die über die Struktur scCtrlBasicMotion beauftragt werden: <i>mcCurrentDirection</i>: Aktuelle Richtung beibehalten (nur bei Modulo-Achsen) <i>mcPositiveDirection</i>: Positive Richtung <i>mcNegativeDirection</i>: Negative Richtung <i>mcShortestWay</i>: Kürzester Weg (nur bei Modulo-Achsen) - Initialwert: <i>mcPositiveDirection</i>
scHomeExtTP	MC_TRIGGER_REF	Übergabe eines externen Touch-Probe-Ereignisses: - Nur relevant bei der Touch-Probe-Konfiguration "Externer Geber". - Zur Beschreibung der Struktur MC_TRIGGER_REF siehe Funktionsbaustein MC_TouchProbe.

5 State Machine



6 Beschreibung der Basisfunktionen

Übersicht der Basisfunktionen

- ▶ [Handfahren \(Jogging\)](#) (📖 22)
- ▶ [Referenzfahrt \(Homing\)](#) (📖 24)
- ▶ [QSP Application](#) (📖 27)
- ▶ [Halt](#) (📖 28)
- ▶ [Absolute Positionierung](#) (📖 29)
- ▶ [Relative Positionierung](#) (📖 31)
- ▶ [Endlosfahrt mit definierter Geschwindigkeit](#) (📖 34)
- ▶ [Abbruch und Beendigung einer Positionierung](#) (📖 37)
- ▶ [Override für Geschwindigkeit](#) (📖 40)

6.1 Wichtige Hinweise zum Betrieb des Technologiemoduls

Kontrollierter Anlauf der Achsen

Bewegungsbefehle, die im gesperrten Achszustand ($xAxisEnabled = FALSE$) gesetzt werden, müssen nach der Freigabe ($xRegulatorOn = TRUE$) erneut durch eine $FALSE \nearrow TRUE$ -Flanke aktiviert werden.

So wird verhindert, dass der Antrieb nach der Reglerfreigabe unkontrolliert anläuft.



Beispiel [Handfahren \(Jogging\)](#) (📖 22):

1. Im gesperrten Achszustand ($xOperationEnabled = FALSE$) wird $xJogPos = TRUE$ gesetzt.
 - $xEnableOperation = FALSE$ (Achse ist gesperrt.)
==> Zustand "READY" ($xOperationEnabled = FALSE$)
 - $xJogPos = TRUE$ (Handfahren soll ausgeführt werden.)
2. Achse freigeben.
 - $xEnableOperation = TRUE$
==> Zustand "READY" ($xOperationEnabled = TRUE$)
3. Handfahren ausführen.
 - $xJogPos = FALSE \nearrow TRUE$
==> Zustand "JOGPOS"

6.2 Handfahren (Jogging)

Voraussetzung

- Die Achse ist freigegeben (*xReadyForMotion* = TRUE).

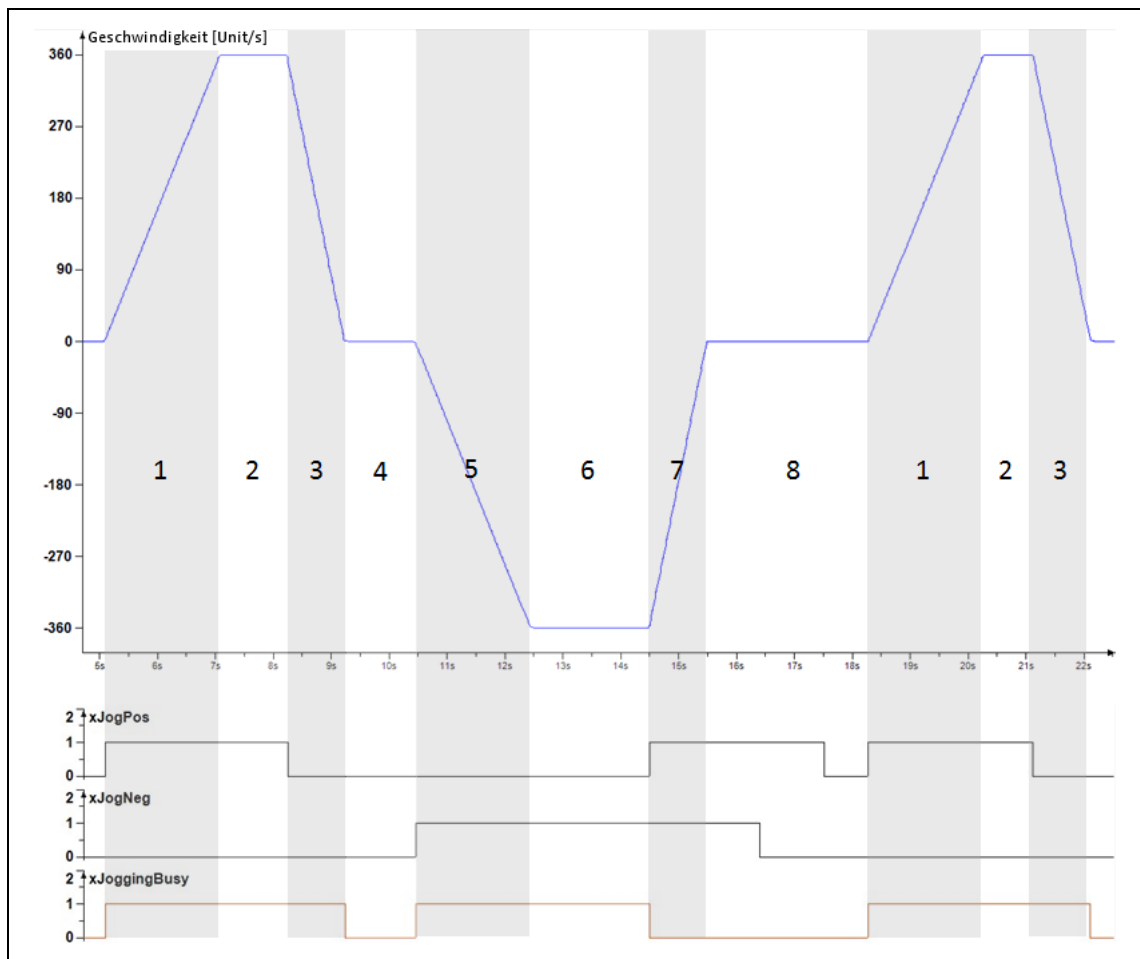
Ausführung

Mit dem Eingang *xJogPos* = TRUE wird die Achse in positive Richtung und mit dem Eingang *xJogNeg* = TRUE in negative Richtung gefahren. Die Achse wird solange gefahren, wie *xJogPos* = TRUE oder *xJogNeg* = TRUE gesetzt bleibt.

Der laufende Fahrbefehl kann nicht durch den anderen Jog-Befehl abgelöst werden. Wenn *xJogPos* = TRUE und *xJogNegPos* = TRUE sind, bremst die Achse in den Stillstand und der Statusausgang *eTMState* wechselt in den Zustand "ERROR".

Erst wenn beide Eingänge zurückgesetzt wurden, wechselt das Technologiemodul wieder zurück in den Zustand "STANDBY".

Fahrtdiagramm



Nr	xJogPos	xJogNeg	xJogging Busy	eTMState	INFO
1	FALSE → TRUE	FALSE	FALSE → TRUE	STANDBY → JOGGING	Die Achse beschleunigt mit 180 units/s ² auf die positive Handfahrgeschwindigkeit 360 units/s.
2	TRUE	FALSE	TRUE	JOGGING	Die Achse verfhrt konstant mit 360 units/s.
3	FALSE → TRUE	FALSE	TRUE	JOGGING	Die Achse wird mit 360 units/s ² in den Stillstand gefhrt.
4	FALSE	FALSE	TRUE → FALSE	JOGGING → STANDBY	Die Achse hat den Stillstand erreicht.
5	FALSE	FALSE → TRUE	FALSE → TRUE	STANDBY → JOGGING	Die Achse beschleunigt mit 180 units/s ² auf die negative Handfahrgeschwindigkeit -360 units/s.
6	FALSE	TRUE	TRUE	JOGGING	Die Achse verfhrt konstant mit -360 units/s.
7	TRUE	TRUE	TRUE → FALSE	ERROR	Die Achse wird mit 360 units/s ² in den Stillstand gefhrt.
8	FALSE	FALSE	FALSE	ERROR → STANDBY	Die Achse steht still und wechselt in den Zustand "STANDBY", wenn beide Eingnge xJogPos und xJogNeg den Wert FALSE haben.

Einzustellende Parameter

Die Parameter fr das Handfahren befinden sich in der AXIS_REF in Subindex 181 ... 184.

Die Parameterwerte knnen whrend des Betriebs verndert werden. Sie werden bei erneutem Setzen der Eingnge *xJogPos* = TRUE oder *xJogNeg* = TRUE bernommen und sofort wirksam.

Die Handfahrparameter aus dem SPS-Programm knnen auch ber den Baustein **L_MC1P_SetBasicParameter** geschrieben werden.

6.3 Referenzfahrt (Homing)

Voraussetzungen

- Des Technologiemodul befindet sich im Zustand "STANDBY".
- Die Achse ist freigegeben (*xReadyForMotion* = TRUE).

Ausführung

Mit einer steigenden Flanke (FALSE→TRUE) am Eingang *xHomeExecute* wird die Referenzfahrt gestartet (Ausgang *xHomingBusy* = TRUE). Die Achse fährt solange, bis die Referenzposition erreicht ist.

Nach Erreichen des Referenzschalters wird *xisHomed* = TRUE.

Bei erfolgreicher Beendigung der Referenzfahrt wechselt *xHomingDone* = TRUE.

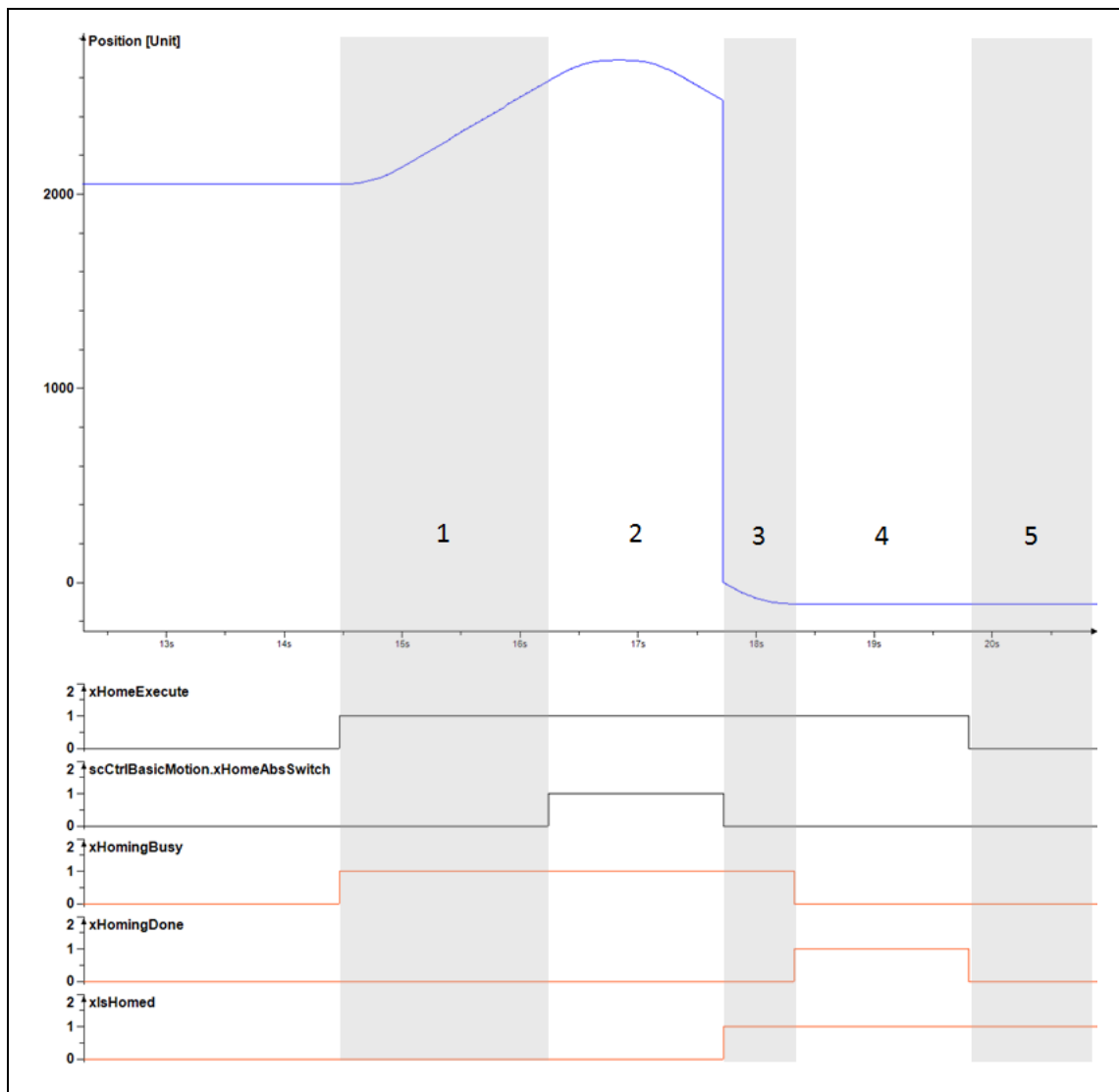
Wird *xHomeExecute* = FALSE gesetzt, dann wird in den Zustand "STANDBY" gewechselt und *xHomingDone* = FALSE gesetzt.

Abbruch der Referenzfahrt

Über den Eingang *xHalt* wird die Funktion abgebrochen.

Die Referenzfahrt wird nicht unterbrochen, wenn vorzeitig *xHomeExecute* = FALSE gesetzt wird.

Fahrtdiagramm



[6-1] Referenzfahrt im Modul CwRpCcwRn mit der Referenzposition "0"

Nr	xHome Execute	xHome AbsSwitch	xHoming Busy	xHoming Done	xIsHomed	eTMState	INFO
1	FALSE → TRUE	FALSE	FALSE → TRUE	FALSE	FALSE	STANDBY → HOMING	Die Achse startet die Referenzfahrt in positive Richtung.
2	TRUE	TRUE → FALSE	TRUE	FALSE	FALSE	HOMING	Die Achse wechselt die Drehrichtung aufgrund der positiven Flanke von xHomeAbsSwitch.
3	TRUE	FALSE → TRUE	TRUE	FALSE	FALSE → TRUE	HOMING	Die negative Flanke von xHomeAbsSwitch setzt die Position der Achse auf die Referenzposition 0 units und die Achse wird in den Stillstand geführt.
4	TRUE	FALSE	TRUE → FALSE	FALSE → TRUE	TRUE	HOMING	Sobald die Achse den Stillstand erreicht hat, wird xHomingBusy = FALSE und xHomingDone = TRUE gesetzt. xHomingDone ist TRUE, solange xHomeExecute = TRUE gesetzt ist, aber immer mindestens für einen Taktzyklus.
5	TRUE → FALSE	FALSE	FALSE	TRUE → FALSE	TRUE	HOMING → STANDBY	Die Achse ist betriebsbereit und kann absolut positioniert werden.

Einzustellende Parameter

Die Parameter für die Referenzfahrt befinden sich in der AXIS_REF in Subindex 70 ... 83.

Die Handfahrparameter aus dem SPS-Programm können auch über den Baustein **L_MC1P_SetBasicParameter** geschrieben werden.

6.4 QSP Application

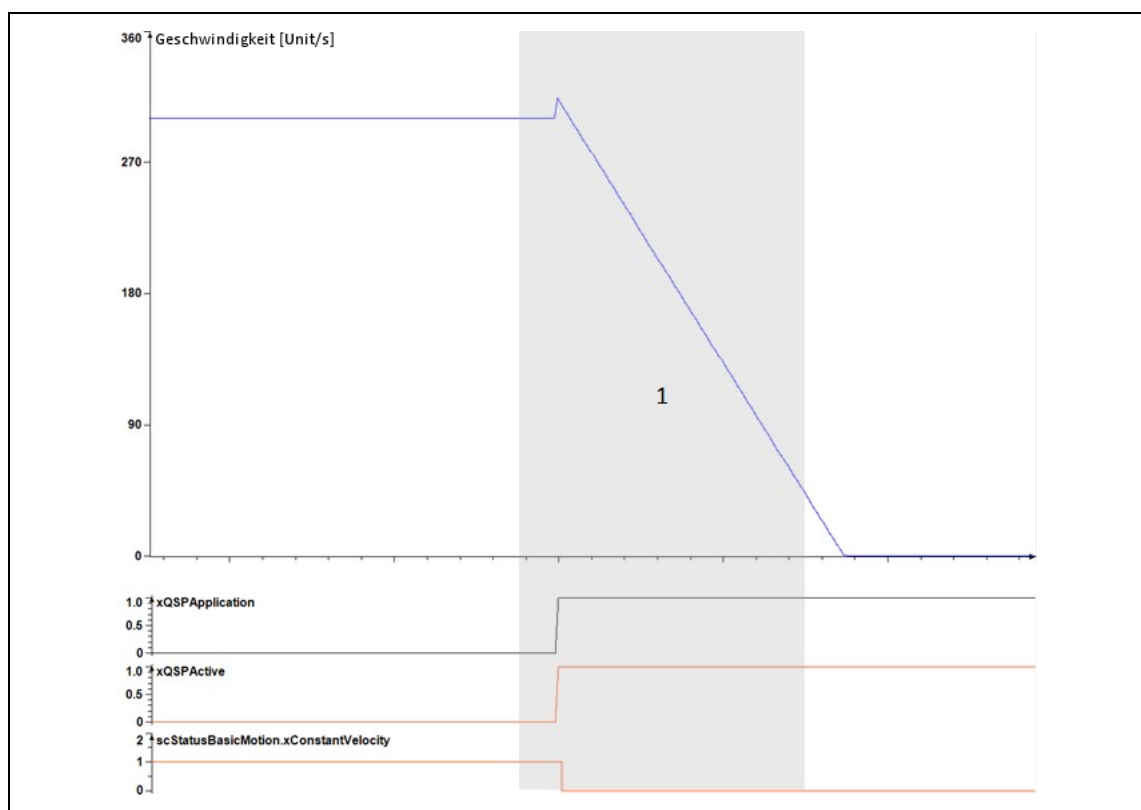
Voraussetzung

- Die Achse ist freigegeben ($xReadyForMotion = TRUE$).

Ausführung

Mit einer steigenden Flanke ($FALSE \rightarrow TRUE$) am Eingang $xQSPApplication$ wird die aktuelle Bewegung der Achse unterbrochen und aufgrund ihres letzten aktuellen Geschwindigkeitswertes an der QSP-Application-Verzögerungsrampe in den Stillstand geführt. Der Parameter für die QSP-Application-Verzögerungsrampe wird in der $AXIS_REF$ eingestellt.

Fahrtdiagramm



Nr	xQSP Application	xQSPActive	xConstant Velocity	eTMState	INFO
1	FALSE $\rightarrow$ TRUE	FALSE $\rightarrow$ TRUE	TRUE $\rightarrow$ FALSE	MOVEVEL $\rightarrow$ STOP	Der Geschwindigkeitssollwert der Achse wird auf die letzte aktuelle Geschwindigkeit gesetzt und über die QSP-Application-Verzögerungsrampe in den Stillstand geführt.

Einzustellende Parameter

Die Parameter für den "QSP Application" befinden sich in der $AXIS_REF$ in Subindex 48 und 49.

Die Handfahrparameter aus dem SPS-Programm können auch über den Baustein **L_MC1P_SetBasicParameter** geschrieben werden.

6.5 Halt

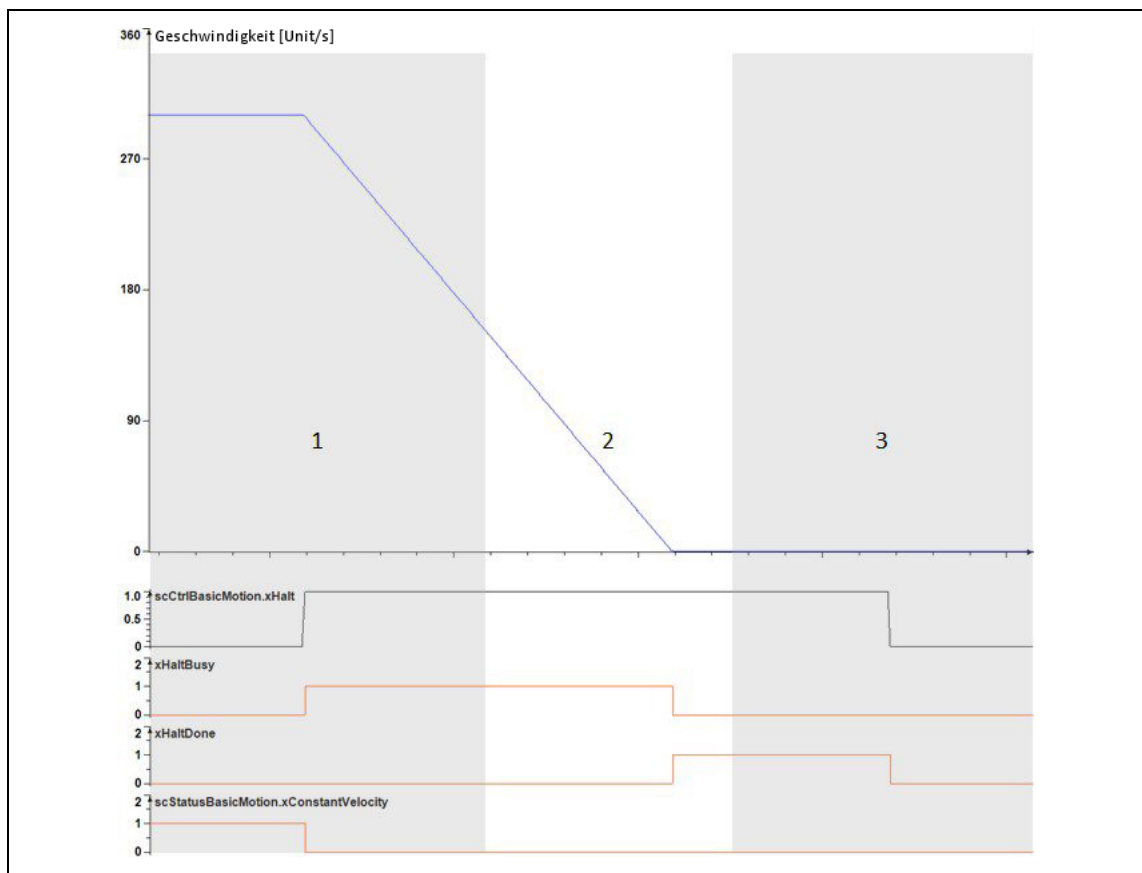
Voraussetzung

- Die Achse ist freigegeben ($xReadyForMotion = TRUE$).

Ausführung

Mit einer steigenden Flanke ($FALSE \rightarrow TRUE$) am Eingang $xHalt$ wird die aktuelle Bewegung der Achse unterbrochen und mit der Halt-Verzögerungsrampe in den Stillstand geführt.

Fahrdiagramm

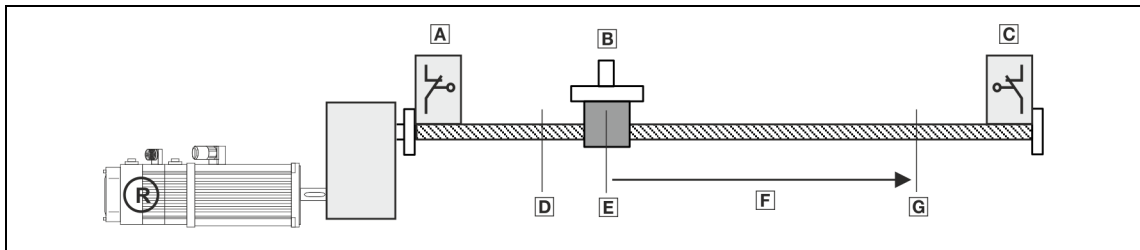


Nr	xHalt	xHaltBusy	xHaltDone	xConstantVelocity	eTMState	INFO
1	$FALSE \rightarrow TRUE$	$FALSE \rightarrow TRUE$	FALSE	$TRUE \rightarrow FALSE$	$MOVEVEL \rightarrow STOP$	Die Achse verzögert mit der Halt-Verzögerungsrampe: $xHaltBusy = TRUE$ $xConstantVelocity = FALSE$.
2	TRUE	$TRUE \rightarrow FALSE$	$FALSE \rightarrow TRUE$	FALSE	STOP	Die Achse wird in den Stillstand geführt: $xHaltBusy = FALSE$ $xHaltDone = TRUE$
3	$TRUE \rightarrow FALSE$	$FALSE \rightarrow TRUE$	$TRUE \rightarrow FALSE$	FALSE	$STOP \rightarrow STANDBY$	$xHalt = FALSE$ $xHaltDone = TRUE \rightarrow FALSE$.

Einzustellende Parameter

Die Parameter für den Halt befinden sich in der AXIS_REF in Subindex 186 und 187.

6.6 Absolute Positionierung



[6-2] Beispiel: Absolute Positionierung einer linearen Achse

Pos.	Beschreibung
A	Negativer Hardware-Endschalter
B	Last (z. B. Schlitten)
C	Positiver Hardware-Endschalter
D	Nullposition des Maßsystems
E	Aktuelle Istposition
F	Zu fahrende Strecke
G	Zielposition (Eingang <i>IrPos_Dist</i>)

Bei der absoluten Positionierung wird die Zielposition im Eingang *IrPos_Dist* (G) als eindeutige absolute Position, bezogen auf die Nullposition des Maßsystems (D), vorgegeben. Die zu fahrende Wegstrecke (F) ergibt sich aus der Distanz zwischen der aktuellen Istposition (E) und der absoluten Zielposition (G).

Voraussetzungen

- Die Achse ist freigegeben (*xReadyForMotion* = TRUE).
- Die Achse ist referenziert (*xIsHomed* = TRUE).

Ausführung

Mit einer steigenden Flanke (FALSE $\rightarrow$ TRUE) am Eingang *xMoveAbsExecute* wird eine Fahrt auf die absolute Zielposition im Eingang *IrPos_Dist* gestartet, gemäß der Parameter für das Bewegungsprofil (siehe unten). Das Technologiemodul wechselt in den Zustand "MOVEABS".

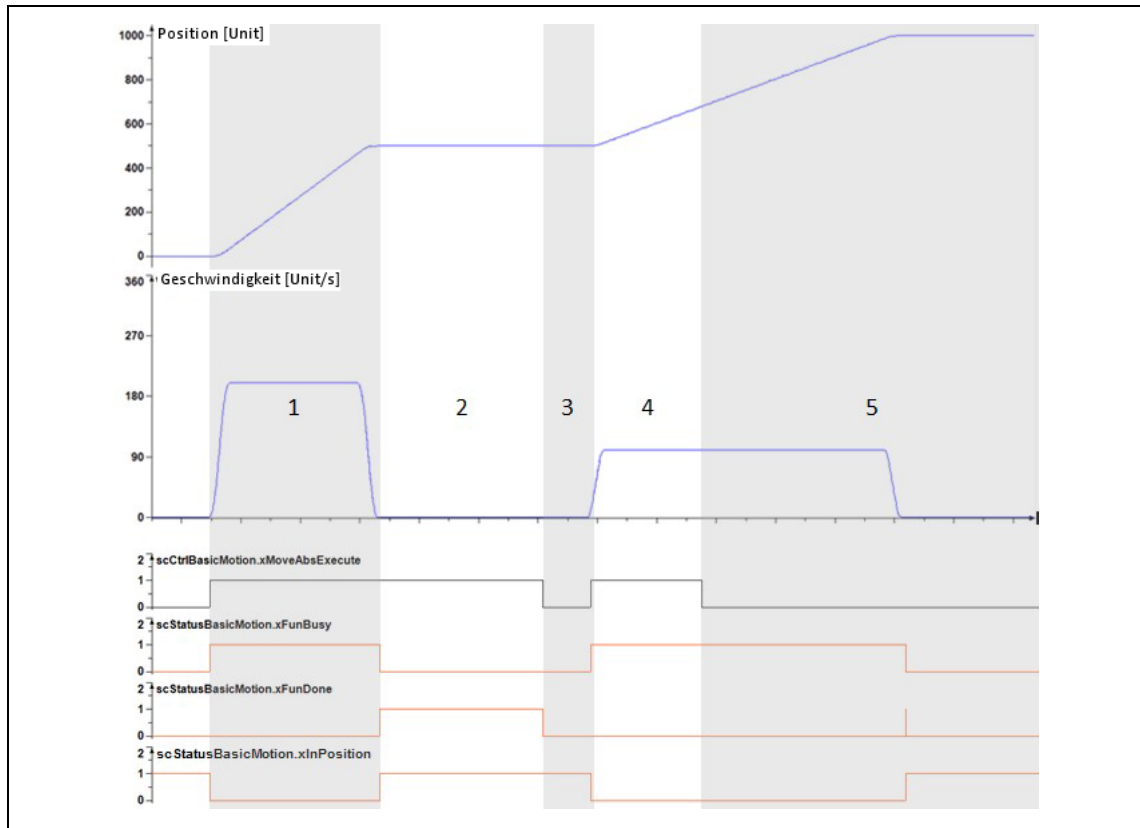
Zum Erreichen der Zielposition wird die Achse über den Parameter *IrDec* in den Stillstand geführt. Die Achse kommt genau in der Zielposition zum Stillstand.

Nach dem Erreichen der Zielposition wird zurück in den Zustand "STANDBY" gewechselt (*xMoveAbsExecute* = FALSE).

Während der Fahrt kann die absolute Zielposition im Eingang *IrPos_Dist* geändert werden. Hierbei ist ein erneutes Auslösen des Execute-Eingangs für die Positionierung erforderlich.

Eine Positionierung kann unterbrochen werden über *xAbort*, *xHalt* oder *xQSPApplication*. Über *xFinishPositioning* kann die Positionierung dann wieder fortgesetzt und beendet werden.

Fahrtdiagramm



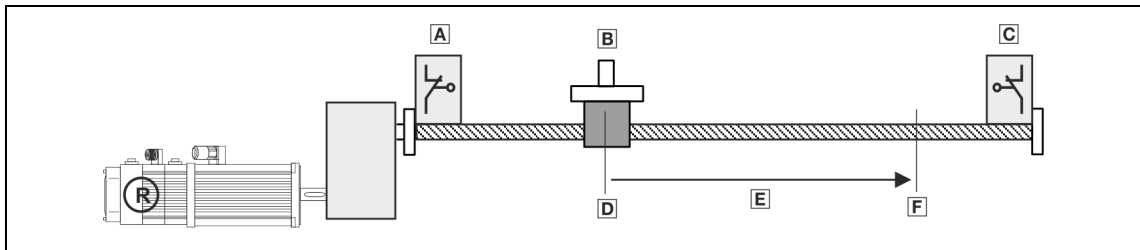
Nr	xMoveAbsExecute	xFunBusy	xFunDone	xInPosition	eTMState	INFO
1	FALSE → TRUE	FALSE → TRUE	FALSE	TRUE → FALSE	STANDBY → MOVEABS	Die Achse startet die Positionierfahrt auf die absolute Zielposition 500 units.
2	TRUE	TRUE → FALSE	FALSE → TRUE	FALSE → TRUE	MOVEABS → STANDBY	Nach Erreichen der Zielposition wird <i>xInPosition</i> = TRUE gesetzt.
3	TRUE → FALSE	FALSE	TRUE → FALSE	TRUE	STANDBY	Sobald <i>xMoveAbsExecute</i> = FALSE gesetzt wird, wechselt das Statussignal <i>xFunDone</i> von TRUE → FALSE.
4	FALSE → TRUE	FALSE → TRUE	FALSE	TRUE → FALSE	STANDBY → MOVEABS	Die Achse startet die Positionierfahrt auf die absolute Zielposition 1000 units.
5	TRUE → FALSE	TRUE → FALSE	FALSE → TRUE → FALSE	FALSE → TRUE	MOVEABS → STANDBY	Nach Erreichen der Zielposition ist <i>xFunDone</i> = TRUE für genau einen Takt, weil während der Positionierung <i>xMoveAbsExecute</i> = FALSE angefordert wurde.

Einstellende Parameter

- *IrVel*, Initialwert: 3600 [units/s]
- *IrAcc*, Initialwert: 36000 [units/s²]
- *IrDcc*, Initialwert: 36000 [units/s²]
- *IrJerk*, Initialwert: 0 [units/s³]

Weitere Parameter für die absolute Positionierung befinden sich in der AXIS_REF in Subindex 130 ... 137.

6.7 Relative Positionierung



[6-3] Beispiel: Relative Positionierung einer linearen Achse

Pos.	Beschreibung
A	Negativer Hardware-Endschalter
B	Last (z. B. Schlitten)
C	Positiver Hardware-Endschalter
D	Aktuelle Istposition
E	Zu fahrende Strecke (= Eingang <i>IrPos_Dist</i>)
F	Zielposition (Eingang aktuelle Istposition + <i>IrPos_Dist</i>)

Bei der relativen Positionierung entspricht die vorgegebene Zielposition im Eingang *IrPos_Dist* der zu fahrenden Wegstrecke ($F = E$). Der Antrieb fährt also genau die vorgegebene Distanz ab der aktuellen Istposition (D).

Voraussetzung

- Die Achse ist freigegeben (*xReadyForMotion* = TRUE).

Ausführung

Mit einer steigenden Flanke ($FALSE \rightarrow TRUE$) am Eingang *xMoveRelExecute* wird eine Fahrt auf die relative Zielposition (Eingang *IrPos_Dist*) gestartet, gemäß der Parameter für das Bewegungsprofil (siehe unten). Das Technologiemodul wechselt in den Zustand "MOVEREL":

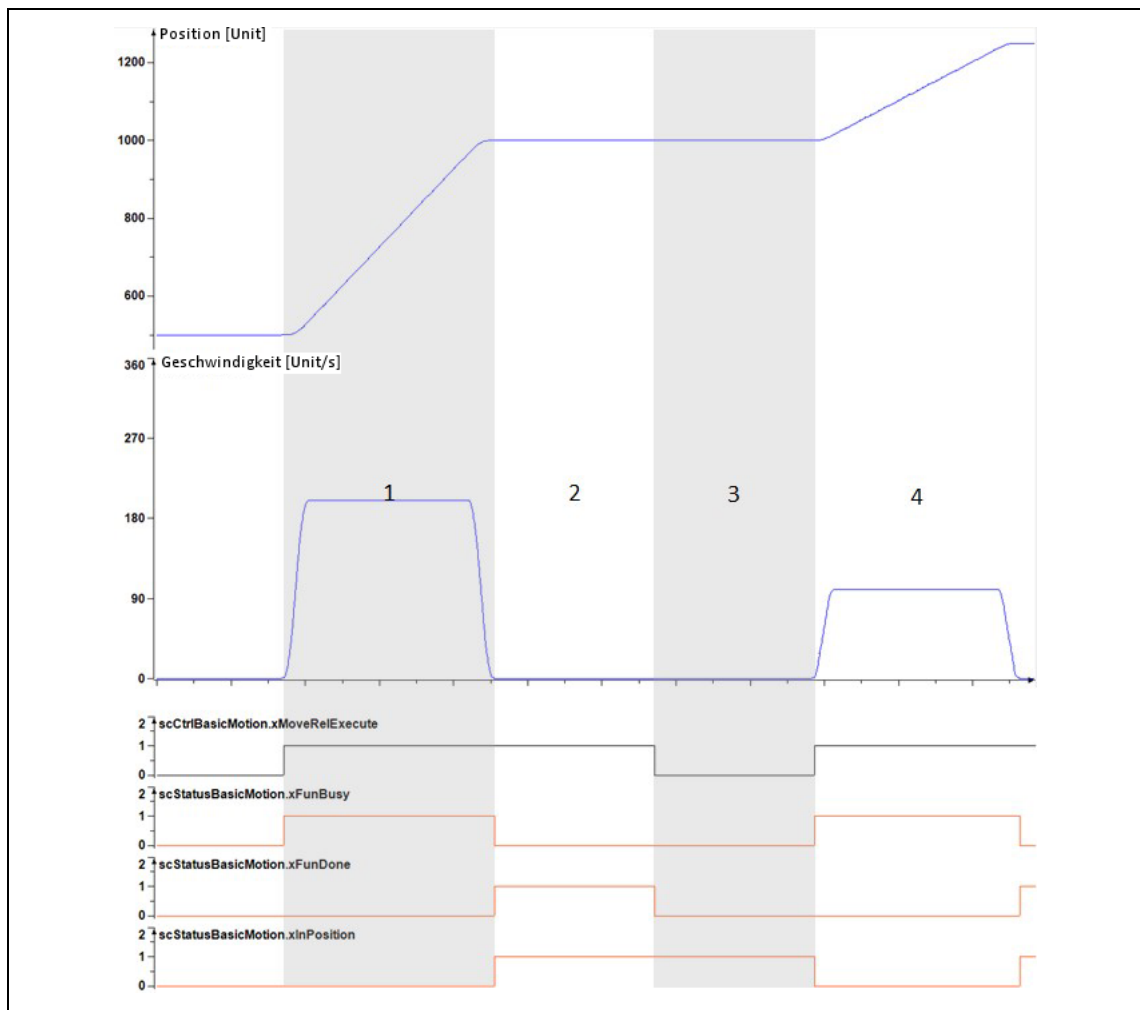
Zum Erreichen der Zielposition wird die Achse über den Parameter *IrDec* in den Stillstand geführt. Die Achse kommt genau in der Zielposition zum Stillstand.

Nach dem Erreichen der Zielposition wird zurück in den Zustand "STANDBY" gewechselt (*xMoveRelExecute* = FALSE).

Während der Fahrt kann die absolute Zielposition im Eingang *IrPos_Dist* geändert werden. Hierbei ist ein erneutes Auslösen des Execute-Eingangs für die Positionierung erforderlich.

Eine Positionierung kann unterbrochen werden über *xAbort*, *xHalt* oder *xQSPApplication*. Über *xFinishPositioning* kann die Positionierung dann wieder fortgesetzt und beendet werden.

Fahrtdiagramm



Nr	xMoveRelExecute	xFunBusy	xFunDone	xInPosition	eTMState	INFO
1	FALSE ↗ TRUE	FALSE ↗ TRUE	FALSE	FALSE	STANDBY → MOVEREL	Die Achse startet die Positionierfahrt auf die relative Zielposition 500 units.
2	TRUE	TRUE ↘ FALSE	FALSE ↗ TRUE	FALSE ↗ TRUE	MOVEREL → STANDBY	Sobald die relative Zielposition erreicht ist, wird <code>xInPosition</code> = TRUE und <code>xFunDone</code> = TRUE gesetzt.
3	FALSE ↗ TRUE	FALSE -	TRUE ↘ FALSE	TRUE	STANDBY	Sobald <code>xMoveRelExecute</code> = FALSE gesetzt wird, wechselt <code>xFunDone</code> von TRUE ↘ FALSE.
4	FALSE ↗ TRUE	FALSE ↗ TRUE ↘ FALSE	TRUE ↘ FALSE ↗ TRUE	TRUE ↘ FALSE ↗ TRUE	STANDBY → MOVEREL → STANDBY	Die Achse startet die Positionierfahrt auf die relative Zielposition 250 units. Nach Erreichen der relativen Zielposition wird <code>xInPosition</code> = TRUE und <code>xFunDone</code> = TRUE gesetzt.

Einzustellende Parameter

- *IrVel*, Initialwert: 3600 [units/s]
- *IrAcc*, Initialwert: 36000 [units/s²]
- *IrDcc*, Initialwert: 36000 [units/s²]
- *IrJerk*, Initialwert: 0 [units/s³]

Weitere Parameter für die relative Positionierung befinden sich in der AXIS_REF in Subindex 130 ... 137.

6.8 Endlosfahrt mit definierter Geschwindigkeit

Voraussetzung

- Die Achse ist freigegeben (*xReadyForMotion* = TRUE).

Ausführung

Mit einer steigenden Flanke (FALSE→TRUE) am Eingang *xMoveVelExecute* wird die Endlosfahrt gestartet, gemäß der Parameter für das Bewegungsprofil (siehe unten). Das Technologiemodul wechselt in den Zustand "MOVEVEL".

Die Achse fährt solange bis eine Software-Endlage erreicht wurde oder bis ein Hardware-Endschalter angefahren wurde:

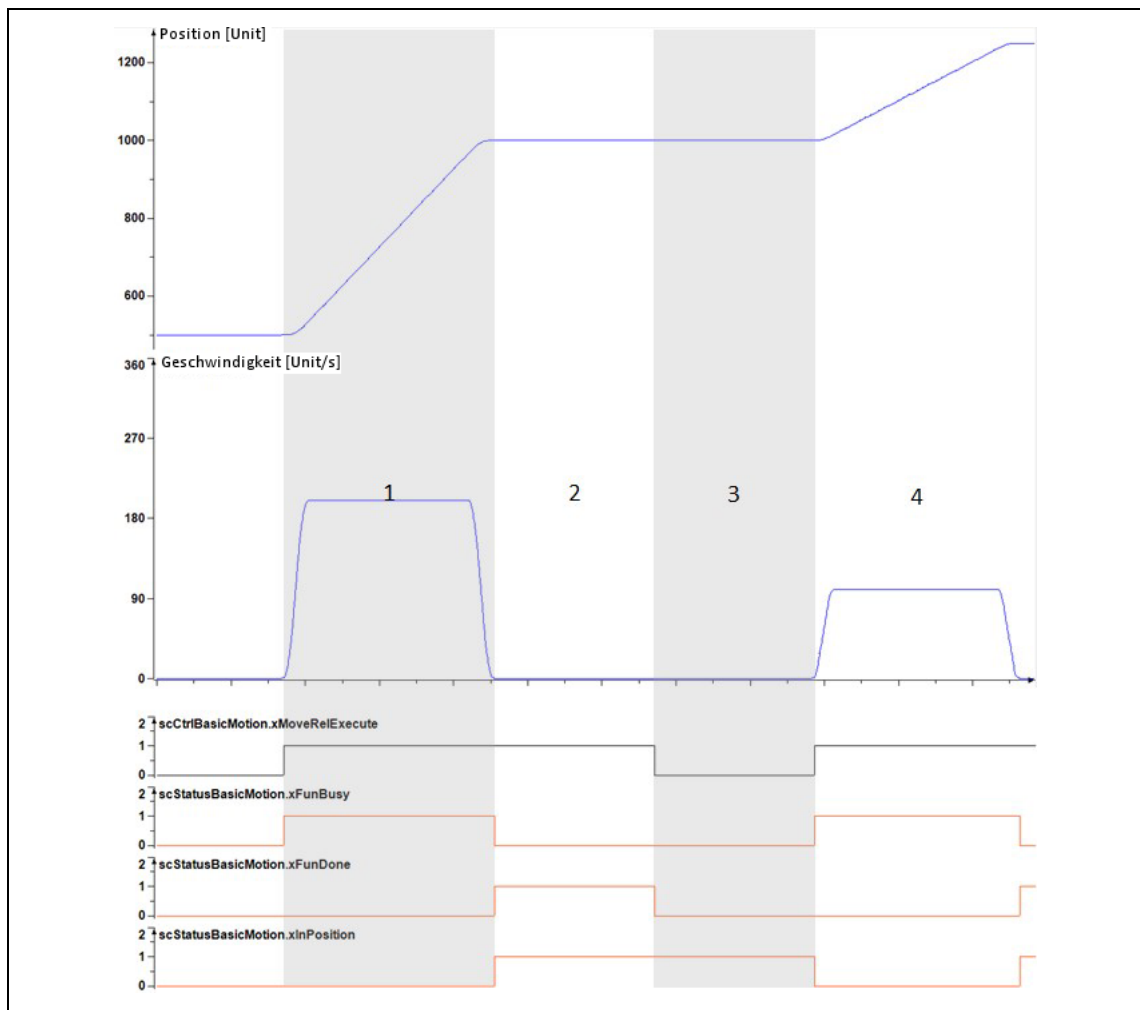
- Eingang *xDisableSWLimit* = FALSE
- Ausgang *xSwLimitEnabled* = TRUE
- Ausgang *xSwLimitSwitchPos/Neg* = TRUE
- Ausgang *xHwLimitSwitchPos/Neg* = TRUE

Bei Erreichen einer Software-Endlage wird die Achse über den Parameter *IrDec* in den Stillstand geführt.

Bei Erreichen eines Hardware-Endschalters wird die Achse über die QSP-Application-Verzögerungsrampe in den Stillstand geführt.

Danach wird zurück in den Zustand "STANDBY" gewechselt.

Fahrtdiagramm



Nr	xMoveVel Execute	xAbort	xFunBusy	xFunDone	xConstant Velocity	eTMState	INFO
1	FALSE ↗ TRUE ↘ FALSE	FALSE	FALSE ↗ TRUE	FALSE	FALSE	STANDBY → MOVEVEL	Die Achse beschleunigt auf die Zielgeschwindigkeit von 200 units/s.
2	FALSE	FALSE	TRUE	FALSE ↗ TRUE	FALSE ↗ TRUE	MOVEVEL	Nach Erreichen der Zielgeschwindigkeit, wechseln <i>xFunDone</i> von FALSE ↗ TRUE und <i>xConstantVelocity</i> von FALSE ↗ TRUE.
3	FALSE	FALSE ↗ TRUE	TRUE ↘ FALSE	TRUE ↘ FALSE ↗ TRUE	TRUE ↘ FALSE	MOVEVEL → STANDBY	Die Geschwindigkeitsfahrt wird durch <i>xAbsort</i> = TRUE beendet und mit der Profilverzögerung in den Stillstand geführt. Nach Erreichen des Stillstands wird <i>xFunBusy</i> = FALSE gesetzt.

Einzustellende Parameter

- *IrVel*, Initialwert: 3600 [units/s]
- *IrAcc*, Initialwert: 36000 [units/s²]
- *IrDcc*, Initialwert: 36000 [units/s²]
- *IrJerk*, Initialwert: 0 [units/s³]

Weitere Parameter für die Endlosfahrt mit definierter Geschwindigkeit befinden sich in der AXIS_REF in Subindex 50 ... 53 und Subindex 104 ... 105.

6.9 Abbruch und Beendigung einer Positionierung

Voraussetzung

- Die Achse ist freigegeben (*xReadyForMotion* = TRUE).

Ausführung

Eine aktive Positionierung wird mit einer steigenden Flanke (FALSE $\nearrow$ TRUE) am Eingang *xAbort* abgebrochen.

Die Achse fährt mit der Verzögerungsrampe *scPar.IrDec* in den Stillstand.

- Ausnahme: TM **SpeedControl** und TM **TablePositioning**

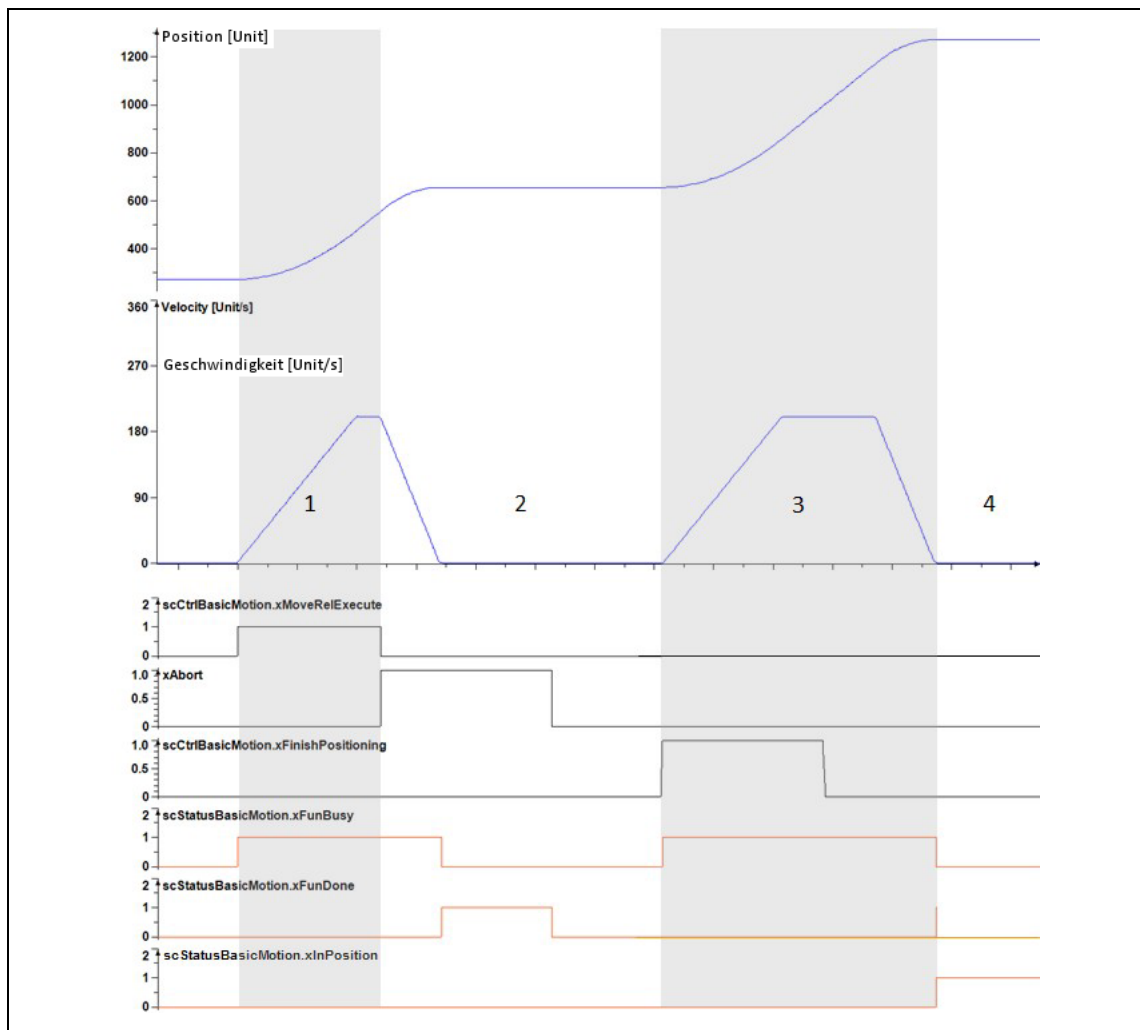
Beide Technologiemodule stellen mehrere Profildatensätze zur Verfügung von denen einer aktiv sein kann.

Wenn einer dieser Profildatensätze aktiv ist, fährt die Achse mit der Verzögerungsrampe dieses aktiven Profildatensatzes in den Stillstand.

Wenn kein Profildatensatz aktiv ist, wird die standardmäßige Verzögerungsrampe *scPar.IrDec* verwendet.

Die Positionierung wird beendet mit einer steigenden Flanke (FALSE $\nearrow$ TRUE) am Eingang *xFinishPositioning*.

Fahrdiagramm



6.9 Abbruch und Beendigung einer Positionierung

Nr	xMoveRel Execute	xAbort	xFinish Positioning	xInPosition	eTMState	INFO
1	FALSE ↗ TRUE	FALSE	FALSE	FALSE	STANDBY → MOVEREL	Die Achse startet die Positionierfahrt mit der relativen Zielposition 1000 units.
2	TRUE ↘ FALSE	TRUE ↘ FALSE	FALSE	FALSE	MOVEREL → STOP → STANDBY	Die Positionierfahrt wird mit <i>xAbsort</i> = TRUE abgebrochen. Die Achse wird mit der Profilverzögerung in den Stillstand geführt.
3	FALSE	FALSE ↗ TRUE	FALSE ↗ TRUE ↘ FALSE	FALSE	STANDBY → MOVEREL	Mit <i>xFinishPositioning</i> = TRUE wird die Positionierung fortgesetzt.
4	FALSE	FALSE	FALSE	FALSE ↗ TRUE	MOVEREL → STANDBY	Sobald die relative Zielposition erreicht ist, werden <i>xInPosition</i> und <i>xFunDone</i> auf TRUE gesetzt.
	FALSE ↗ TRUE	FALSE ↗ TRUE ↘ FALSE	TRUE ↘ FALSE ↗ TRUE	TRUE ↘ FALSE ↗ TRUE	STANDBY → MOVEREL → STANDBY	Die Achse startet die Positionierfahrt mit der relativen Zielposition 250 units. Nach Erreichen der relativen Zielposition wird <i>sxInPosition</i> = TRUE und <i>xFunDone</i> = TRUE gesetzt.

6.10 Override für Geschwindigkeit

Voraussetzung

- Die Achse ist freigegeben (*xReadyForMotion* = TRUE).

Ausführung

Um Overrides für die aktiven Profildaten einstellen zu können, stehen am Technologiemodul die folgenden Eingänge zur Verfügung:

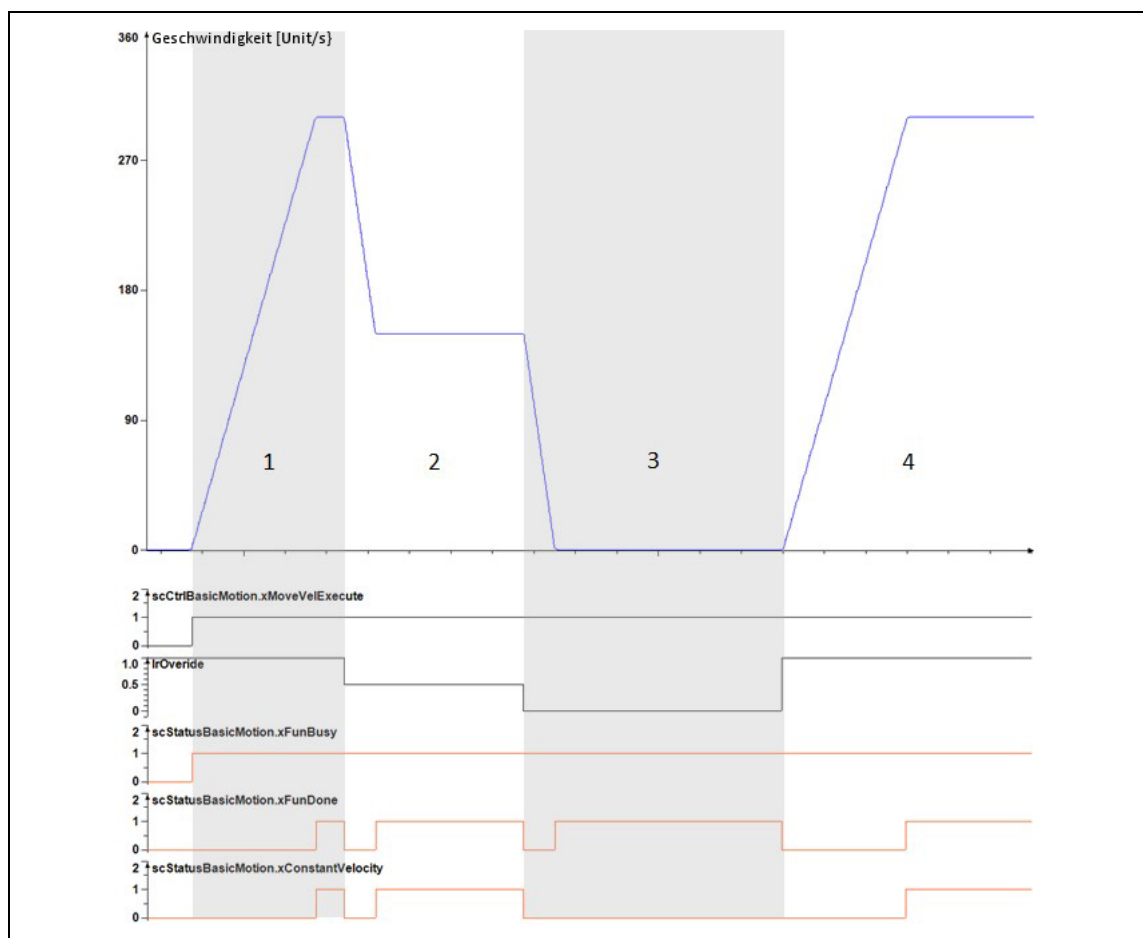
- *IrOverride*, Initialwert: 1.0
- *scCtrlBasicMotion.IrOverrideAcc*, Initialwert: 1.0 (100 %)
- *scCtrlBasicMotion.IrOverrideJerk*, Initialwert: 1.0 (100 %)

Für profilgenerierende Funktionen können separate Overrides für Geschwindigkeit, Beschleunigung, Verzögerung und Ruck eingestellt werden.

Die eingestellten Werte wirken als Faktor und werden mit dem jeweils aktiven Wert multipliziert.

Beispiel

In diesem Beispiel wird die Verwendung des Eingangs *IrOverride* gezeigt.



6.10 Override für Geschwindigkeit

Nr	xMoveVel Execute	IrOverride	xFunBusy	xFunDone	xConstant Velocity	eTMState	INFO
1	FALSE ↗ TRUE	1	FALSE ↗ TRUE	FALSE ↗ TRUE	FALSE ↗ TRUE	STANDBY → MOVEVEL	Die Achse beschleunigt auf die Zielgeschwindigkeit von 300 units/s.
2	TRUE	0.5	TRUE	TRUE ↘ FALSE ↗ TRUE	TRUE ↘ FALSE ↗ TRUE	MOVEVEL	Der Override-Wert wird auf 0.5 gesetzt und die Achse verzögert mit der Profilverzögerung auf 150 units/s
3	TRUE	0	TRUE	TRUE ↘ FALSE ↗ TRUE	TRUE ↘ FALSE	MOVEVEL	Der Override-Wert wird auf 0 gesetzt und die Achse wird mit der Profilverzögerung auf 0 units/s in den Stillstand geführt.
4	TRUE	1	TRUE	TRUE ↘ FALSE ↗ TRUE	FALSE ↗ TRUE	MOVEVEL	Der Override-Wert wird auf 1 gesetzt und die Achse beschleunigt mit der Profilbeschleunigung auf 300 units/s.

A

Abbruch einer Positionierung [37](#)
Absolute Positionierung [29](#)
Anlauf der Achsen [11](#), [21](#)
Anwendungshinweise [7](#)
Ausgänge [14](#)

B

Basisfunktionen der Technologiemodule [21](#)
Beendigung einer Positionierung [37](#)

D

Dokumenthistorie [5](#)

E

Eingänge [13](#)
E-Mail an Lenze [43](#)
Endlosfahrt mit definierter Geschwindigkeit [34](#)

F

Feedback an Lenze [43](#)
Funktionsbaustein L_TB2P_BasicMotionBase [12](#)

G

Geschwindigkeits-Override [40](#)
Gestaltung der Sicherheitshinweise [7](#)

H

Halt [28](#)
Handfahren (Jogging) [22](#)
Hinweise zum Betrieb des Technologiemoduls [11](#), [21](#)

K

Kontrollierter Anlauf der Achsen [21](#)

L

L_TB2P_BasicMotionBase [12](#)
L_TB2P_scCtrlBasicMotion [15](#)
L_TB2P_scPar_BasicMotionBase [19](#)
L_TB2P_scStatusBasicMotion [17](#)

O

Override für Geschwindigkeit [40](#)

P

Parameterstruktur L_TB2P_scPar_BasicMotionBase [19](#)

Q

QSP Application [27](#)

R

Referenzfahrt (Homing) [24](#)
Relative Positionierung [31](#)

S

Sicherheitshinweise [8](#)
State Machine [20](#)

V

Variablenbezeichner [6](#)
Verwendete Konventionen [6](#)

Z

Zielgruppe [4](#)
Zustände [20](#)



Ihre Meinung ist uns wichtig

Wir erstellen diese Anleitung nach bestem Wissen mit dem Ziel, Sie bestmöglich beim Umgang mit unserem Produkt zu unterstützen.

Vielleicht ist uns das nicht überall gelungen. Wenn Sie das feststellen sollten, senden Sie uns Ihre Anregungen und Ihre Kritik in einer kurzen E-Mail an:

feedback-docu@lenze.com

Vielen Dank für Ihre Unterstützung.

Ihr Lenze-Dokumentationsteam

