

SIEMENS

SIMOTION Interpreter




Listenhandbuch

Vorwort	1
Aufbau eines Kommandos	2
Kommandos zur Programmsteuerung des Interpreters	3
Kommandos für Motion Control	4
Kommandos für Weiterschaltbedingungen	5
Kommandos für Programmverzweigungen	6
Sonstige Kommandos	7
Kontakt	A

Rechtliche Hinweise

Warnhinweiskonzept

Dieses Handbuch enthält Hinweise, die Sie zu Ihrer persönlichen Sicherheit sowie zur Vermeidung von Sachschäden beachten müssen. Die Hinweise zu Ihrer persönlichen Sicherheit sind durch ein Warndreieck hervorgehoben, Hinweise zu alleinigen Sachschäden stehen ohne Warndreieck. Je nach Gefährdungsstufe werden die Warnhinweise in abnehmender Reihenfolge wie folgt dargestellt.

 GEFAHR
bedeutet, dass Tod oder schwere Körperverletzung eintreten wird , wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.
 WARNUNG
bedeutet, dass Tod oder schwere Körperverletzung eintreten kann , wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.
 VORSICHT
bedeutet, dass eine leichte Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.
ACHTUNG
bedeutet, dass Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.


Beim Auftreten mehrerer Gefährdungsstufen wird immer der Warnhinweis zur jeweils höchsten Stufe verwendet. Wenn in einem Warnhinweis mit dem Warndreieck vor Personenschäden gewarnt wird, dann kann im selben Warnhinweis zusätzlich eine Warnung vor Sachschäden angefügt sein.

Qualifiziertes Personal

Das zu dieser Dokumentation zugehörige Produkt/System darf nur von für die jeweilige Aufgabenstellung **qualifiziertem Personal** gehandhabt werden unter Beachtung der für die jeweilige Aufgabenstellung zugehörigen Dokumentation, insbesondere der darin enthaltenen Sicherheits- und Warnhinweise. Qualifiziertes Personal ist auf Grund seiner Ausbildung und Erfahrung befähigt, im Umgang mit diesen Produkten/Systemen Risiken zu erkennen und mögliche Gefährdungen zu vermeiden.

Bestimmungsgemäßer Gebrauch von Siemens-Produkten

Beachten Sie Folgendes:

 WARNUNG
Siemens-Produkte dürfen nur für die im Katalog und in der zugehörigen technischen Dokumentation vorgesehenen Einsatzfälle verwendet werden. Falls Fremdprodukte und -komponenten zum Einsatz kommen, müssen diese von Siemens empfohlen bzw. zugelassen sein. Der einwandfreie und sichere Betrieb der Produkte setzt sachgemäßen Transport, sachgemäße Lagerung, Aufstellung, Montage, Installation, Inbetriebnahme, Bedienung und Instandhaltung voraus. Die zulässigen Umgebungsbedingungen müssen eingehalten werden. Hinweise in den zugehörigen Dokumentationen müssen beachtet werden.

Marken

Alle mit dem Schutzrechtsvermerk ® gekennzeichneten Bezeichnungen sind eingetragene Marken der Siemens AG. Die übrigen Bezeichnungen in dieser Schrift können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen kann.

Haftungsausschluss

Wir haben den Inhalt der Druckschrift auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in dieser Druckschrift werden regelmäßig überprüft, notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten.

Inhaltsverzeichnis

1	Vorwort.....	5
1.1	Allgemeine Hinweise	5
1.2	Über dieses Dokument.....	7
2	Aufbau eines Kommandos	9
3	Kommandos zur Programmsteuerung des Interpreters	11
3.1	CMD_START_PROGRAM.....	11
3.2	CMD_STOP_PROGRAM.....	12
3.3	CMD_NEXT_STEP.....	13
4	Kommandos für Motion Control.....	15
4.1	CMD_POWER_ON.....	15
4.2	CMD_POWER_OFF	16
4.3	CMD_POS_ABS.....	17
4.4	CMD_POS_REL.....	18
4.5	CMD_POS_SUPERIMPOSED.....	19
4.6	CMD_MOVE	20
4.7	CMD_STOP	21
4.8	CMD_ESTOP.....	22
4.9	CMD_HOMING	23
4.10	CMD_RESET_AXIS	24
4.11	CMD_GEAR_IN.....	25
4.12	CMD_GEAR_OUT.....	26
4.13	CMD_CAM_IN_LEADING_OFFSET	27
4.14	CMD_CAM_IN_FOLLOWING_OFFSET.....	28
4.15	CMD_CAM_OUT.....	29
4.16	CMD_PHASING	30
4.17	CMD_SET_ACCEL_OVERRIDE.....	31
4.18	CMD_ENABLE_OUTPUT_CAM.....	32
4.19	CMD_DISABLE_OUTPUT_CAM.....	33
5	Kommandos für Weiterschaltbedingungen.....	35
5.1	CMD_WAIT_TIME.....	35
5.2	CMD_WAIT_FOR_CONDITION.....	36

5.3	CMD_WAIT_FOR_AXIS_STATE	37
5.4	CMD_WAIT_FOR_POS.....	39
6	Kommandos für Programmverzweigungen.....	41
6.1	CMD_JUMP	41
6.2	CMD_JUMP_CMP_CONDITION	42
6.3	CMD_JUMP_CMP_POS.....	43
6.4	CMD_JUMP_CMP_VARIABLE	44
6.5	CMD_JUMP_CMP_STATE.....	45
6.6	CMD_LOOP.....	46
6.7	CMD_END_LOOP.....	47
6.8	CMD_START_PARALLEL_SEQUENCE.....	48
6.9	CMD_STOP_SEQUENCE.....	49
7	Sonstige Kommandos	51
7.1	CMD_NONE.....	51
7.2	CMD_NOP	52
7.3	CMD_SET_CONDITION.....	53
7.4	CMD_SET_VARIABLE	54
7.5	CMD_INC_VARIABLE.....	55
7.6	CMD_CALC_VARIABLE.....	56
A	Kontakt.....	57
A.1	Ansprechpartner.....	57
A.2	Internetadressen	58
	Index.....	59

Vorwort

1.1 Allgemeine Hinweise

Hinweis

Die Standardapplikationen sind unverbindlich und erheben keinen Anspruch auf Vollständigkeit hinsichtlich Konfiguration und Ausstattung sowie jeglicher Eventualitäten. Die Standardapplikationen stellen keine kundenspezifischen Lösungen dar, sondern sollen lediglich Hilfestellung bieten bei typischen Aufgabenstellungen. Sie sind für den sachgemäßen Betrieb der beschriebenen Produkte selbst verantwortlich. Diese Standardapplikationen entheben Sie nicht der Verpflichtung zu sicherem Umgang bei Anwendung, Installation, Betrieb und Wartung. Durch Nutzung dieser Standardapplikationen erkennen Sie an, dass wir über die beschriebene Haftungsregelung hinaus nicht für etwaige Schäden haftbar gemacht werden können. Wir behalten uns das Recht vor, Änderungen an diesen Standardapplikationen jederzeit ohne Ankündigung durchzuführen. Bei Abweichungen zwischen den Vorschlägen in diesen Standardapplikationen und anderen Siemens Publikationen, wie z. B. Katalogen, hat der Inhalt der anderen Dokumentation Vorrang.

Gewährleistung, Haftung und Support

Im Falle kostenloser Überlassung der Applikation gilt folgendes:

Für die in diesem Dokument enthaltenen Informationen übernehmen wir keine Gewähr.

Andere und alle sonstigen Rechte und Ansprüche gegen die Siemens AG, gleich aus welchem Rechtsgrund, sind ausgeschlossen. Insbesondere Ansprüche gegen die Siemens AG auf Schadenersatz, insbesondere wegen Produktionsausfalls, Nutzungsausfalls, entgangenen Gewinns, direkter, indirekter oder Folgeschäden sind ausgeschlossen.

Dies gilt nicht, soweit zwingend gehaftet wird, z. B. nach dem Produkthaftungsgesetz, in Fällen des Vorsatzes, grober Fahrlässigkeit von Vorgesetzten und leitenden Angestellten der Siemens AG oder in Fällen des arglistigen Verschweigens von Mängeln.

Diese Haftungsbegrenzung findet auch Anwendung zu Gunsten von Subunternehmern, Zulieferern, Beauftragten, Vorgesetzten, leitenden Angestellten und Angestellten der Siemens AG.

Auf diese Vereinbarung findet für Kunden mit Firmensitz in Deutschland das deutsche Recht Anwendung für Kunden mit Firmensitz außerhalb Deutschlands findet das schweizerische Recht Anwendung. Die Anwendbarkeit des Übereinkommens der Vereinten Nationen über Verträge über den internationalen Warenkauf vom 11.04.1980 (CISG) ist ausgeschlossen.

Im Falle entgeltlicher Überlassung der Applikation gilt die für das jeweilige Geschäft zutreffende Alternative:

- Alternative 1: (Internes Geschäft)

Es gelten, sofern nicht unten etwas Abweichendes geregelt ist, die "Bedingungen für Lieferungen und Leistungen im Siemens-internen Geschäft" in der jeweils zum Zeitpunkt der Überlassung gültigen Fassung.

- Alternative 2: (Inlandsgeschäft der Siemens AG)

Es gelten, sofern nicht unten etwas Abweichendes geregelt ist, die "Allgemeine Bedingungen zur Überlassung von Software für Automatisierungs- und Antriebstechnik an Lizenznehmer mit Sitz in Deutschland" in der jeweils zum Zeitpunkt der Überlassung gültigen Fassung.

- Alternative 3: (Direktexportgeschäft der Siemens AG)

Es gelten, sofern nicht unten etwas Abweichendes geregelt ist, die "Allgemeine Bedingungen zur Überlassung von Softwareprodukten für Automation and Drives an Lizenznehmer mit Sitz außerhalb Deutschlands" in der jeweils zum Zeitpunkt der Überlassung gültigen Fassung.

Die Weitergabe oder Vervielfältigung dieser Applikationsbeispiele oder Auszüge daraus in unbearbeiteter Form sind nicht gestattet, soweit nicht ausdrücklich von Siemens Industry Sector gestattet.

Hinweis auf Exportkennzeichen

AL: N

ECCN: N

1.2 Über dieses Dokument

Ziel

In diesem Listenhandbuch werden alle Kommandos der Applikation **Interpreter** beschrieben. Die Beschreibung ist gegliedert nach den Aufgabenbereichen der Kommandos. Die Auswahl eines Kommandos erfolgt über einen Enumerator (Enum). Allen Kommandos ist das Präfix CMD vorangestellt. Der **Interpreter** ist eine Ergänzung zum Motion Control System SIMOTION. Details zum Engineeringssystem und Funktionen des SIMOTION SCOUT entnehmen Sie bitte der *SIMOTION SCOUT Standard-Dokumentation*.

Weitere Beschreibungen und Informationen zur Applikation **Interpreter** finden Sie im *Applikationshandbuch Interpreter*.

Das Handbuch ist eine Ergänzung zur SIMOTION Standard-Dokumentation.

Hinweis

Diese Dokumentation erhebt nicht den Anspruch, alle Gerätedetails oder Gerätevarianten zu erfassen oder jeden denkbaren Fall des Betriebs oder der Anwendung zu berücksichtigen.

Sollten Sie weitere Informationen benötigen oder sollten spezielle Probleme auftreten, die für das Anwendungsgebiet nicht ausführlich genug behandelt werden, wenden Sie sich bitte an die örtliche Siemens-Niederlassung.

Zielgruppe

Das vorliegende Dokument wendet sich an Programmierer.

Siemens Industry Online Support

Dieser Beitrag stammt aus dem Siemens Industry Online Support. Durch den folgenden Link gelangen Sie direkt zur Downloadseite dieses Dokuments:

<http://support.automation.siemens.com/WW/view/de/61182043>

Aufbau eines Kommandos

Der Aufbau aller Kommandos ist einheitlich. Diese Struktur ist unten stehend abgebildet.

Kommando (Enum)	TO-Index (DINT)	Parameter 1 bis 6 (REAL)	Mode (BYTE)	TimeOut (UINT)	Weiterschaltbedingung (Enum)
--------------------	--------------------	-----------------------------	----------------	-------------------	---------------------------------

Bild 2-1 Aufbau eines Kommandos

Ein Kommando enthält folgende Bestandteile:

- **Kommando:**
Das Kommando wird über ein Enumerator (Enum) vorgegeben. Eine Auflistung der Namen befindet sich in der Unit **dPublic**.
- **TO-Index:**
Technologieobjekte (TO) für den **Interpreter** werden mit eigenen Arrays übergeben. Die TO können dann über den entsprechenden Array-Index angesprochen werden. Über den TO-Index wird das TO, z. B. die Achse bzw. der Nocken angegeben, auf welches das Kommando angewendet werden soll. Bei manchen Kommandos kann ausgewählt werden, ob es für ein spezielles oder für alle TO (Vorgabe von -1) ausgeführt werden soll. Bei einigen Befehlen ist auch ein externer Geber erlaubt, z. B. als Leitachse für einen Gleichlauf. Dieser muss dann ebenfalls im Array der Achsen eingetragen werden und wird über den zugehörigen Array-Index angesprochen.
Da im Achs-Array unterschiedliche Achs-Typen vereinigt sind (Drehzahl-Achsen, Positionier-Achsen, Gleichlauf-Achsen, Bahnachsen, externe Geber) stellt der **Interpreter** in seiner Ausgangs-Schnittstelle ein Array mit den jeweils zugehörigen Typen zur Verfügung. Dieses Array kann z. B. von einem HMI ausgewertet werden.
- **Parameter 1 bis 6:**
Für ein Kommando können bis zu sechs Parameter zur weiteren Spezifizierung angegeben werden. Bei falscher Programmierung eines Parameters wird das entsprechende Programm mit einer Fehlermeldung abgebrochen. Es wird ein Eintrag in die Diagnose-Struktur des **Interpreters** geschrieben. Bei einigen Ausnahmefällen wird eine Default-Projektierung verwendet oder der entsprechende Befehl wird ignoriert. Dies ist bei den einzelnen Befehlen explizit beschrieben.
(siehe auch *Listenhandbuch Interpreter*)

- **Mode:**
Mit dem Parameter *mode* kann ausgewählt werden, ob die Parameter 1 bis 6 absolut oder indirekt über einen Zeiger auf ein REAL-Array (DurchgangsvARIABLE des **Interpreters**) vorgegeben werden oder ob die UserDefault-Werte von SIMOTION verwendet werden. Indirekte Vorgabe über eine Variable bedeutet, dass im entsprechenden Parameter nicht der geforderte Wert direkt steht, sondern der Index auf eine spezifische Variable im Variablenfeld *variables*, in der dann der tatsächliche Wert steht.
Dabei bedeutet: Für jeden Parameter wird ein Bit in der Variablen *mode* verwendet: Das Bit 0 ist für den Parameter 1 zuständig. Bei Wert TRUE wird eine indirekte Vorgabe verwendet.
Das Bit 7 wird für die Verwendung von UserDefault-Werten der Systemvariablen der TO verwendet. Dies wird jedoch derzeit nur für das Kommando CMD_ENABLE_OUTPUT_CAM verwendet. Ist Bit 7 gesetzt, dann werden die Bits 0 bis 5 ignoriert. Bit 6 hat keine Bedeutung und wird vom **Interpreter** ignoriert.
- **TimeOut:**
Für Motion Control-Kommandos und Warte-Kommandos mit Ausnahme von CMD_WAIT_TIME kann eine Zeit definiert werden, nach deren Ablauf das Kommando spätestens fertig bearbeitet sein muss. Bei Überschreiten der vorgegebenen Zeit wird die betreffende Bearbeitung mit Fehler abgebrochen. Sinnvoll ist diese Angabe aber nur, wenn mit der Weiterschaltung gewartet wird, bis das Kommando erfolgreich abgearbeitet wurde. Bei der Voreinstellung von 0 ms wird diese Überwachung abgeschaltet.
- **Weiterschaltbedingung:**
Bei allen Kommandos kann ausgewählt werden, ob das nächste Kommando sofort bearbeitet wird oder erst wenn das aktuelle Kommando erfolgreich ausgeführt und beendet wurde. Darüber hinaus können über diesen Parameter mehrere Kommandos geklammert werden d. h., sie werden im selben Interpreter-Zyklus ausgeführt. Ein Anwendungsbeispiel hierfür ist eine Berechnung von Stapelpositionen, die nicht von anderen Kommandos unterbrochen werden soll. Dies ist nur bis zu einem Maximalwert von Kommandos möglich, der über die Konstante LMCIPR_NUMBER_OF_COMMANDS_PER_CYCLE in der Unit **cPublic** vorgegeben wird, weil diese Funktionalität die Performancebelastung erhöht.
Es gibt somit drei verschiedene Weiterschaltbedingungen:
WHEN_COMMAND_DONE (WCD)
IMMEDIATELY (IM)
IMMEDEEDIATELY_AND_NEXT_COMMAND (IM_NC)
Bei Vorgabe eines falschen Werts wird WCD als Default-Projektierung verwendet und ein entsprechender Eintrag in der Diagnose-Struktur des **Interpreters** hinterlegt.

Hinweis

Bei manchen Kommandos sind bestimmte Werte für die Weiterschaltbedingung nicht sinnvoll. Diese können zwar programmiert werden, werden aber von der Applikation Interpreter ignoriert. Beispiel: Sprungbefehle mit der Programmierung Weiterschaltbedingung WCD. Sprünge werden prinzipiell in einem Interpretertakt abgearbeitet und wirken daher als wäre die Weiterschaltbedingung IM programmiert.

Kommandos zur Programmsteuerung des Interpreters

3

3.1 CMD_START_PROGRAM

Starte Programm

Dieses Kommando kann nur im Programmbetrieb oder Einzelschrittbetrieb ausgeführt werden. Bei Absetzen dieses Kommandos wird die Bearbeitung des angegebenen Verfahrssatz-Programms ab dem angegebenen Verfahrssatz begonnen. Ein bereits laufendes Programm wird dabei abgebrochen (einschließlich aller existierenden parallel laufenden Sequenzen). Laufende Kommandos, z. B. MOVE bleiben jedoch aktiv. Bei Einzelschritt wird dabei nur der 1. Verfahrssatz ausgeführt.

Enumerator: CMD_START_PROGRAM := 4010

Parameter	Wert	Beschreibung
i32IndexTO	-	
r32Parameter1	-	
r32Parameter2	Verfahrssatz-Programm	
r32Parameter3	Verfahrssatz	
r32Parameter4	-	
r32Parameter5	-	
r32Parameter6	-	
b8Mode	-	Angabe, welche Parameter indirekt vorgegeben werden
u16TimeOut	-	
eNextCommand	-	

3.2 CMD_STOP_PROGRAM

Stoppe Programm

Dieses Kommando kann nur im Programmbetrieb oder Einzelschrittbetrieb ausgeführt werden. Bei Absetzen dieses Kommandos wird die Bearbeitung des Verfahrstanz-Programms sowie aller etwaiger paralleler Sequenzen gestoppt.

Sollen mit dem Kommando **Stoppe Programm** bestimmte Aktionen durchgeführt werden, wie z. B. Achsen anhalten, Gleichlaufverbände oder Nocken deaktivieren usw., so muss dies in einem eigenen Programm parametrisiert und ausgeführt werden. Ein solches Programm kann in den Konfigurationsdaten des Interpreters projektiert werden.

Bei Absetzen des Stopp-Kommandos werden noch aktive Bewegungs-Kommandos immer bis zum Ende abgearbeitet.

Enumerator: CMD_STOP_PROGRAM := 4020

i32IndexTO	-
r32Parameter1	-
r32Parameter2	-
r32Parameter3	-
r32Parameter4	-
r32Parameter5	-
r32Parameter6	-
b8Mode	Angabe, welche Parameter indirekt vorgegeben werden.
u16TimeOut	-
eNextCommand	-

3.3 CMD_NEXT_STEP

Abarbeitung nächster Schritt

Dieses Kommando kann nur bei Einzelschrittbetrieb ausgeführt werden. Bei Absetzen wird der nächste Verfahrssatz (einschließlich aller parallelen Sequenzen) eines laufenden Programms, welches über CMD_START_PROGRAM gestartet wurde, ausgeführt. Eine etwaige Weberschaltbedingung wird eingehalten. Erst wenn diese Bedingung erfüllt ist, kann wiederum der nächste Schritt über CMD_NEXT_STEP bearbeitet werden. Solange die Ausgangsvariable des Interpreter *programState* noch den Wert SINGLESTEP_ACTIVE hat, sind noch nicht alle Kommandos beendet und CMD_NEXT_STEP wird ignoriert. Das Kommando wird ignoriert, wenn in einer Sequenz noch ein Kommando aktiv ist z. B. eine mit WCD programmierte Positionierung ist noch nicht abgeschlossen.

Enumerator : CMD_NEXT_STEP := 4040

Parameter	Wert	Beschreibung
i32IndexTO	-	
r32Parameter1	-	
r32Parameter2	-	
r32Parameter3	-	
r32Parameter4	-	
r32Parameter5	-	
r32Parameter6	-	
b8Mode	-	
u16TimeOut	-	
eNextCommand	-	

Kommandos für Motion Control

Generell gilt für alle Bewegungsbefehle:

Die Dynamikwerte können achsspezifisch am Parameter des **Interpreters** *axisConfigData* projiziert werden. Informationen zum Aufbau eines Kommandos finden Sie im Abschnitt Aufbau eines Kommandos (Seite 9).

Bei Kommandos zum Positionieren, zur Drehzahlvorgabe sowie Stoppen wird für die Beschleunigung und Verzögerung auch der interne Override berücksichtigt. Schreibbar ist der interne Override über das Kommando `CMD_SET_ACCEL_OVERRIDE`.

Wichtige Informationen zu Motion Control finden Sie in der SIMOTION Standard-Dokumentation im Handbuch *SIMOTION Motion Control TO Achse elektrisch/hydraulisch, Externer Geber* (Achse_Technologie_Funktionen.pdf).

4.1 CMD_POWER_ON

Achsfreigabe geben

Mit diesem Kommando können alle oder einzelne Achsen sowie externe Geber freigegeben werden.

Enumerator: `CMD_POWER_ON := 10`

Parameter	Wert	Beschreibung
<code>i32IndexTO</code>	-1 0..LMCIPR_NUMBER_OF_AXES -1:	Alle Achsen bzw. externe Geber Spezielle Achse bzw. Externer Geber, Index auf Array <i>axisConfigData.toAxis</i>
<code>r32Parameter1</code>	-	
<code>r32Parameter2</code>	-	
<code>r32Parameter3</code>	-	
<code>r32Parameter4</code>	-	
<code>r32Parameter5</code>	-	
<code>r32Parameter6</code>	-	
<code>u16TimeOut</code>	0..65535 [ms]	Überwachungszeit für Kommando
<code>eNextCommand</code>	WCD	Weiterschaltung, wenn Achse bzw. externer Geber freigegeben ist

4.2 CMD_POWER_OFF

Parameter	Wert	Beschreibung
	IM	Weiterschaltung im nächsten Takt
	IM_NC	Weiterschaltung sofort im gleichen Takt

4.2 CMD_POWER_OFF

Achsfreigabe zurücknehmen

Mit diesem Kommando kann für alle oder einzelne Achsen sowie externe Geber die Freigabe zurückgenommen werden.

Enumerator: CMD_POWER_OFF := 20

Parameter	Wert	Beschreibung
i32IndexTO	-1:	Alle Achsen bzw. externe Geber
	0..LMCIPR_NUMBER_OF_AXES -1:	Spezielle Achse bzw. externer Geber Index auf Array <i>axisConfigData.toAxis</i>
r32Parameter1	-	
r32Parameter2	-	
r32Parameter4	-	
r32Parameter5	-	
r32Parameter6	-	
u16TimeOut	0..65535 [ms]	Überwachungszeit für Kommando
eNextCommand	WCD	Weiterschaltung, wenn Achse nicht mehr in Regelung
	IM	Weiterschaltung im nächsten Takt
	IM_NC	Weiterschaltung sofort im gleichen Takt

4.3 CMD_POS_ABS

Absolutes Positionieren

Mit diesem Kommando kann eine einzelne Achse absolut positioniert werden.

Enumerator: CMD_POS_ABS := 30

Parameter	Wert	Beschreibung
i32IndexTO	0..LMCIPR_NUMBER_OF_AXES -1:	Spezielle Achse
r32Parameter1	Zielposition	
r32Parameter2	Geschwindigkeit	Nur positive Werte und -1.0 erlaubt -1.0: Verwendung der Vorbelegung im System
r32Parameter3	Richtung	
	0:	Positiv
	1:	Negativ
	2:	Kürzester Weg
	3:	Effektive, zuletzt programmierte Richtung
	9:	User default
r32Parameter4	-	
r32Parameter5	-	
r32Parameter6	-	
b8Mode		Angabe, welche Parameter indirekt vorgegeben werden
u16TimeOut	0..65535 [ms]	Überwachungszeit für Kommando
eNextCommand	WCD	Weiterschaltung, wenn Positionierung beendet
	IM	Weiterschaltung im nächsten Takt
	IM_NC	Weiterschaltung sofort im gleichen Takt

4.4 CMD_POS_REL

Relatives Positionieren

Mit diesem Kommando kann eine einzelne Achse relativ positioniert werden.

Enumerator: CMD_POS_REL := 40

Parameter	Wert	Beschreibung
i32IndexTO	0..LMCIPR_NUMBER_OF_AXES -1:	Spezielle Achse
r32Parameter1	Distanz	Vorzeichen definiert die Richtung
r32Parameter2	Geschwindigkeit	Nur positive Werte und -1.0 erlaubt -1.0: Verwendung der Vorbelegung im System
r32Parameter3	-	
r32Parameter4	-	
r32Parameter5	-	
r32Parameter6	-	
b8Mode		Angabe, welche Parameter indirekt vorgegeben werden
u16TimeOut	0..65535 [ms]	Überwachungszeit für Kommando
eNextCommand	WCD	Weiterschaltung, wenn Positionierung beendet
	IM	Weiterschaltung im nächsten Takt
	IM_NC	Weiterschaltung sofort im gleichen Takt

4.5 CMD_POS_SUPERIMPOSED

Überlagertes Positionieren

Mit diesem Kommando kann eine einzelne Achse überlagert positioniert werden.

Enumerator: CMD_POS_SUPERIMPOSED := 50

Parameter	Wert	Beschreibung
i32IndexTO	0..LMCIPR_NUMBER_OF_AXES -1:	Spezielle Achse
r32Parameter1	Distanz	Vorzeichen definiert Richtung
r32Parameter2	Geschwindigkeit	Nur positive Werte und -1.0 erlaubt -1.0: Verwendung der Vorbelegung im System
r32Parameter3	-	
r32Parameter4	-	
r32Parameter5	-	
r32Parameter6	-	
b8Mode		Angabe, welche Parameter indirekt vorgegeben werden
u16TimeOut	0..65535 [ms]	Überwachungszeit für Kommando
eNextCommand	WCD	Weiterschaltung, wenn Positionierung beendet
	IM	Weiterschaltung im nächsten Takt
	IM_NC	Weiterschaltung sofort im gleichen Takt

4.6 CMD_MOVE

Drehzahlvorgabe

Mit diesem Kommando kann für eine einzelne Achse eine Drehzahl vorgegeben werden. Ob die Achse drehzahl geregelt oder lagegeregelt verfahren werden soll, kann achsspezifisch projiziert werden (*axisConfigData*).

Enumerator: CMD_MOVE := 60

Parameter	Wert	Beschreibung
i32IndexTO	0..LMCIPR_NUMBER_OF_AXES -1:	Spezielle Achse
r32Parameter1	-	
r32Parameter2	Geschwindigkeit	Nur positive Werte und -1.0 erlaubt -1.0: Verwendung der Vorbelegung im System
r32Parameter3	Richtung: 0: 1: 2: 3: 9:	Positiv Negativ Kürzester Weg Effektive, zuletzt programmierte Richtung User default
r32Parameter4	-	
r32Parameter5	-	
r32Parameter6	-	
b8Mode		Angabe, welche Parameter indirekt vorgegeben werden
u16TimeOut	0..65535 [ms]	Überwachungszeit für Kommando
eNextCommand	WCD IM IM_NC	Weiterschalten, wenn Solldrehzahl erreicht Weiterschaltung im nächsten Takt Weiterschaltung sofort im gleichen Takt

4.7 CMD_STOP

Stopp der Achse

Mit diesem Kommando können alle oder eine einzelne Achse gestoppt werden.

Enumerator: CMD_STOP := 70

Parameter	Wert	Beschreibung
i32IndexTO	-1:	Alle Achsen
	0..LMCIPR_NUMBER_OF_AXES -1:	Spezielle Achse
r32Parameter1	Zu stoppende Bewegung:	
	0:	ALL (Default)
	1:	Basic
	2:	Superimposed
r32Parameter2	-	
r32Parameter3	-	
r32Parameter4	-	
r32Parameter5	-	
r32Parameter6	-	
b8Mode		Angabe, welche Parameter indirekt vorgegeben werden
u16TimeOut	0..65535 [ms]	Überwachungszeit für Kommando
eNextCommand	WCD	Weiterschaltung, wenn Achse steht
	IM	Weiterschaltung im nächsten Takt
	IM_NC	Weiterschaltung sofort im gleichen Takt

4.8 CMD_ESTOP

Notstopp der Achse

Mit diesem Kommando können alle oder eine einzelne Achse notgestoppt werden. Der Override von Beschleunigung und Verzögerung wird bei diesem Kommando nicht berücksichtigt (siehe CMD_SET_ACCEL_OVERRIDE (Seite 31)). Es wird mit den in der Konfigurationsstruktur angegebenen Dynamiken angehalten.

Hinweis

Bei Vorgabe EStop-Einstellungen = 2 und Angabe einer Stopp-Zeit von ≤ 0 wird die Default-Einstellung (Stopp mit maximaler Verzögerung) verwendet.

Enumerator: CMD_ESTOP := 80

Parameter	Wert	Beschreibung
i32IndexTO	-1:	Alle Achsen
	0..LMCIPR_NUMBER_OF_AXES -1:	Spezielle Achse
r32Parameter1	EStop-Modus:	
	0:	Notstopp (Default)
	1:	Stopp ohne Abbruch
	2:	Stopp mit Abbruch
r32Parameter2	EStop-Einstellungen:	
	0:	Nur relevant bei P1 = 0 Stopp mit maximaler Verzögerung (Default)
	1:	Stopp mit Sollwert 0
	2:	Stopp in definierter Zeit über Rampe
	3:	Stopp mit projektierten Dynamiken
r32Parameter3	Stopp-Zeit [s]	Nur relevant bei P2 = 2
r32Parameter4	-	
r32Parameter5	-	
r32Parameter6	-	
b8Mode		Angabe, welche Parameter indirekt vorgegeben werden
u16TimeOut	0..65535 [ms]	Überwachungszeit für Kommando
eNextCommand	WCD	Weiterschaltung, wenn Achse steht
	IM	Weiterschaltung im nächsten Takt
	IM_NC	Weiterschaltung sofort im gleichen Takt

4.9 CMD_HOMING

Referenzieren

Mit diesem Kommando kann eine einzelne Achse bzw. ein externer Geber referenziert werden.

Bei Absolutwertgebern muss kein Referenziermodus angegeben werden, da automatisch eine Absolutwertgeberjustage durchgeführt wird. Ähnliches gilt für virtuelle Achsen. Hier wird immer **Setzen der Referenzposition**, unabhängig von Parameter 2 durchgeführt.

Der Referenziermodus spielt somit nur in den folgenden Fällen eine Rolle:

- Reale Achsen und externer Geber mit Inkrementalgeber (alle Varianten möglich)
- Vorgabe von *redefinePosition* (dies ist für alle Typen von Achsen und externen Gebern möglich)

Hinweis

Der Referenziermodus 3 *redefinePosition* liefert nicht das Status-Signal *axisHomed* an der betroffenen Achse/dem betroffenen externen Geber.

Enumerator: CMD_HOMING := 90

Parameter	Wert	Beschreibung
i32IndexTO	0..LMCIPR_NUMBER_OF_AXES -1:	Spezielle Achse oder externer Geber
r32Parameter1	Referenzposition	
r32Parameter2	0:	Aktives Referenzieren
	1:	Passives Referenzieren
	2:	Setzen der Referenzposition
	3:	Setzen des Koordinatensystems eines TO
r32Parameter3	Nur relevant, wenn P2 = 3	Angabe des Zuweisungsorts für die neue Position (<i>EnumRedefineSpecification</i>)
	0:	Sollwert COMMAND_VALUE
	1:	Istwert ACTUAL_VALUE
	2:	Sollwert Hauptbewegung COMMAND_VALUE_BASIC_MOTION
	3:	Sollwert überlagerte Bewegung COMMAND_VALUE_SUPERIMPOSED_MOTION
r32Parameter4	Nur relevant, wenn P2 = 3	Angabe des Zuweisungsart für die neue Position (<i>EnumRedefineMode</i>)
	0:	Absolute Wertangabe ABSOLUTE

4.10 CMD_RESET_AXIS

Parameter	Wert	Beschreibung
	1:	Relative Wertangabe RELATIVE
r32Parameter5	-	
r32Parameter6	-	
b8Mode		Angabe, welche Parameter indirekt vorgegeben werden
u16TimeOut	0..65535 [ms]	Überwachungszeit für Kommando
eNextCommand	WCD	Achse ist referenziert bzw. auf Position gesetzt
	IM	Weiterschaltung im nächsten Takt
	IM_NC	Weiterschaltung sofort im gleichen Takt

4.10 CMD_RESET_AXIS

Reset von Achsen

Mit diesem Kommando können alle oder eine einzelne Achse bzw. externe Geber zurückgesetzt werden. An der Achse (und an etwaigen vorhandenen unterlagerten Objekten) anstehende TO-Alarme werden quittiert.

Enumerator: CMD_RESET_AXIS := 100

Parameter	Wert	Beschreibung
i32IndexTO	-1:	Alle Achsen bzw. externe Geber
	0..LMCIPR_NUMBER_OF_AXES -1:	Spezielle Achse bzw. externer Geber
r32Parameter1	-	
r32Parameter2	-	
r32Parameter3	-	
r32Parameter4	-	
r32Parameter5	-	
r32Parameter6	-	
b8Mode	-	
u16TimeOut	0..65535 [ms]	Überwachungszeit für Kommando
eNextCommand	WCD	Alarme sind quittiert
	IM	Weiterschaltung im nächsten Takt
	IM_NC	Weiterschaltung sofort im gleichen Takt

4.11 CMD_GEAR_IN

Getriebegleichlauf aktivieren

Mit diesem Kommando kann eine Gleichlaufachse auf eine Leitachse (Achse oder externer Geber) im Getriebegleichlauf synchronisiert werden.

Hinweis

Die Geschwindigkeit für den Synchronisiervorgang wird aus den Vorbelegungsdaten (siehe SCOUT: Gleichlaufobjekt, Vorbelegung) genommen. Alle weiteren Achs-Dynamikwerte werden aus den Konfigurationsdaten des **Interpreters** übernommen.

Soll ein gegensinniger Getriebegleichlauf realisiert werden, so ist das derzeit nur über entsprechende Projektierung unter den Vorbelegungen des Gleichlaufobjekts in SIMOTION SCOUT möglich.

Enumerator: CMD_GEAR_IN := 110

Parameter	Wert	Beschreibung
i32IndexTO	0..LMCIPR_NUMBER_OF_AXES - 1:	Spezielle Achse
r32Parameter1	Zähler	0 nicht erlaubt
r32Parameter2	Nenner	0 nicht erlaubt
r32Parameter3	Offset / Phasenversatz	
r32Parameter4	Mode	
	0:	Absolut
	1:	Relativ
r32Parameter5	Aufsynchronisieren	
	0:	USER_DEFAULT (Default)
	1:	IMMEDIATELY_BY_TIME_PROFILE
r32Parameter6		Angabe der Leitachse über Index im Achs-Array
b8Mode		Angabe, welche Parameter indirekt vorgegeben werden
u16TimeOut	0..65535 [ms]	Überwachungszeit für Kommando
eNextCommand	WCD	Weiterschalten, wenn synchron
	IM	Weiterschaltung im nächsten Takt
	IM_NC	Weiterschaltung sofort im gleichen Takt

4.12 CMD_GEAR_OUT

Getriebegleichlauf deaktivieren

Mit diesem Kommando kann eine im Getriebegleichlauf laufende Gleichlaufachse asynchronisiert werden. Die Dynamikwerte werden aus den UserDefault-Werten der Achse genommen. Es wird nicht geprüft, ob ein Getriebegleichlauf aktiv ist.

Enumerator: CMD_GEAR_OUT := 120

Parameter	Wert	Beschreibung
i32IndexTO	0..LMCIPR_NUMBER_OF_AXES -1:	Spezielle Achse
r32Parameter1	-	
r32Parameter2	-	
r32Parameter3	-	
r32Parameter4	-	
r32Parameter5	-	
r32Parameter6	-	
b8Mode	-	
u16TimeOut	0..65535 [ms]	Überwachungszeit für Kommando
eNextCommand	WCD	Weiterschalten, wenn Achse steht
	IM	Weiterschaltung im nächsten Takt
	IM_NC	Weiterschaltung sofort im gleichen Takt

4.13 CMD_CAM_IN_LEADING_OFFSET

Kurvenscheibengleichlauf mit Offset zur Leitachse aktivieren

Mit diesem Kommando kann eine Gleichlaufachse auf eine Leitachse (Achse oder externer Geber) im Kurvenscheibengleichlauf synchronisiert werden. Es kann ein Offset zur Position der Leitachse angegeben werden.

Hinweis

Die Dynamikwerte für das Aufsynchronisieren der Achsen werden aus den Vorbelegungsdaten der Achse entnommen.

Enumerator: CMD_CAM_IN_LEADING_OFFSET := 130

Parameter	Wert	Beschreibung
i32IndexTO	0..LMCIPR_NUMBER_OF_AXES - 1:	Spezielle Achse
r32Parameter1	Index auf Kurvenscheibe im Array für Kurvenscheiben	
r32Parameter2	Mode: 0: 1:	Zyklisch Einmalig
r32Parameter3	Leitachse Offset	
r32Parameter4	Mode: 0: 1: 2: 3:	Leitachse und Folgeachse absolut Leitachse und Folgeachse relativ Leitachse absolut und Folgeachse relativ Leitachse relativ und Folgeachse absolut
r32Parameter5	Aufsynchronisieren: 0: 1:	USER_DEFAULT (Default) IMMEDIATELY_BY_TIME_PROFILE
r32Parameter6		Angabe der Leitachse über Index des Achs-Array
b8Mode		Angabe, welche Parameter indirekt vorgegeben werden
u16TimeOut	0..65535 [ms]	Überwachungszeit für Kommando
eNextCommand	WCD IM IM_NC	Weiterschalten, sobald synchron Weiterschaltung im nächsten Takt Weiterschaltung sofort im gleichen Takt

4.14 CMD_CAM_IN_FOLLOWING_OFFSET

Kurvenscheibengleichlauf mit Offset der Folgeachse aktivieren

Mit diesem Kommando kann eine Gleichlaufachse auf einen Leitwert im Kurvenscheibengleichlauf synchronisiert werden. Es kann ein Offset zur Position der Folgeachse vorgegeben werden.

Das Verhalten entspricht prinzipiell dem Befehl für Kurvenscheibengleichlauf mit Offset Folgeachse zur Leitachse.

Enumerator: CMD_CAM_IN_FOLLOWING_OFFSET := 140

Parameter	Wert	Beschreibung
i32IndexTO	0..LMCIPR_NUMBER_OF_AXES -1:	Spezielle Achse
i32Parameter1	Index auf Kurvenscheibe im Array für Kurvenscheiben	
r32Parameter2	Mode:	
	0:	Zyklisch
	1:	Einmalig
r32Parameter3	Folgeachse Offset	
r32Parameter4	Mode:	
	0:	Leitachse und Folgeachse absolut
	1:	Leitachse und Folgeachse relativ
	2:	Leitachse absolut und Folgeachse relativ
	3:	Leitachse relativ und Folgeachse absolut
r32Parameter5	Aufsynchronisieren:	
	0:	USER_DEFAULT (Default)
	1:	IMMEDIATELY_BY_TIME_PROFILE
r32Parameter6		Angabe des Leitwert über Index des Achs-Array
b8Mode		Angabe, welche Parameter indirekt vorgegeben werden
u16TimeOut	0..65535 [ms]	Überwachungszeit für Kommando
eNextCommand	WCD	Weiterschalten, sobald synchron
	IM	Weiterschaltung im nächsten Takt
	IM_NC	Weiterschaltung sofort im gleichen Takt

4.15 CMD_CAM_OUT

Kurvenscheibengleichlauf deaktivieren

Mit diesem Kommando kann eine Gleichlaufachse im Kurvenscheibengleichlauf asynchronisiert werden. Die Dynamikwerte werden aus den User Default-Werten der Achse genommen. Es wird nicht geprüft, ob ein Kurvenscheibengleichlauf aktiv ist.

Enumerator: CMD_CAM_OUT := 150

Parameter	Wert	Beschreibung
i32IndexTO	0..LMCIPR_NUMBER_OF_AXES -1:	Spezielle Achse
r32Parameter2	-	
r32Parameter3	-	
r32Parameter4	-	
r32Parameter5	-	
r32Parameter6	-	
u16TimeOut	0..65535 [ms]	Überwachungszeit für Kommando
eNextCommand	WCD	Weiterschalten, wenn Achse nicht mehr synchron
	IM	Weiterschaltung im nächsten Takt
	IM_NC	Weiterschaltung sofort im gleichen Takt

4.16 CMD_PHASING

Phasenverschiebung einstellen

Mit diesem Kommando wird die Phase der Folgeachse gegenüber dem Leitwert im aktiven Gleichlauf verschoben. Das Kommando kann für Getriebegleichlauf und Kurvenscheibengleichlauf verwendet werden.

Enumerator: CMD_Phasing := 160

Parameter	Wert	Beschreibung
i32IndexTO	0..LMCIPR_NUMBER_OF_AXES -1:	Spezielle Achse
r32Parameter1	-	
r32Parameter2	Geschwindigkeit	-1.0: Verwendung der Vorbelegung aus dem System
r32Parameter3	Phasenverschiebung	
r32Parameter4	Art der Phasenverschiebung:	
	0:	Absolut
	1:	Relativ
r32Parameter5	-	
r32Parameter6	-	
b8Mode		Angabe, welche Parameter indirekt vorgegeben werden
u16TimeOut	0..65535 [ms]	Überwachungszeit für Kommando
eNextCommand	WCD	Weiterschalten, sobald Offset herausgefahren ist
	IM	Weiterschaltung im nächsten Takt
	IM_NC	Weiterschaltung sofort im gleichen Takt

4.17 CMD_SET_ACCEL_OVERRIDE

Setzen des Override für Beschleunigung und Verzögerung

Mit diesem Kommando kann ein Override für die Beschleunigung und Verzögerung von Achsen prozentual vorgegeben werden. Der 100 %-Wert für Beschleunigung und Verzögerung wird in der Konfigurationsstruktur für jede Achse angegeben. Der vorgegebene Override wird auf die Werte von 1 % bis 1000 % begrenzt. Das Kommando wird ignoriert, wenn (bei indirekter Programmierung) ein falscher Variablen-Index, außerhalb der Array-Grenzen, vorgegeben wurde.

Enumerator: CMD_SET_ACCEL_OVERRIDE := 170

Parameter	Wert	Beschreibung
i32IndexTO	-1: 0..LMCIPR_NUMBER_OF_AXES -1:	Alle Achsen Spezielle Achse
r32Parameter1	-	
r32Parameter2	Override für Beschleunigung und Verzögerung	Minimum = 1 % Maximum = 1000 %
r32Parameter3	-	
r32Parameter4	-	
r32Parameter5	-	
r32Parameter6	-	
b8Mode		Angabe, welche Parameter indirekt vorgegeben werden
u16TimeOut	-	
eNextCommand	WCD IM IM_NC	Ist in diesem Kommando nicht sinnvoll, da dieses Kommando immer in einem Takt abgearbeitet wird. Weiterschaltung im nächsten Takt Weiterschaltung sofort im gleichen Takt

4.18 CMD_ENABLE_OUTPUT_CAM

Aktivierung eines Nockens

Mit diesem Kommando kann ein Nocken aktiviert werden.

Enumerator: CMD_ENABLE_OUTPUT_CAM := 300

Parameter	Wert	Beschreibung
i32IndexTO	0..LMCIPR_NUMBER_OF_OUTPUT_CAMS -1:	Spezieller Nocken
r32Parameter1	Einschaltposition	
r32Parameter2	Einschaltdauer: Ausschaltposition	Je nach Nockentyp
r32Parameter3	Vorhaltezeit: Einschalten	Default: Wert aus den User Default-Daten
r32Parameter4	Vorhaltezeit, Ausschalten	Default: Wert aus den User Default-Daten
r32Parameter5	Wirkrichtung	
	0:	Positiv
	1:	Negativ
	2:	Beide Richtungen
	9:	User default
r32Parameter6	Invertierung des Ausgangs	
	0:	Nein
	1:	Ja
	2:	keine Änderung
b8Mode	Bit 7:	Ist Bit 7 gesetzt, so hat dieses Vorrang vor den Bits 0 bis 5. D. h. für den Fall, dass mehrere Bits gesetzt sind (z. B. Bit 7 und Bit 0) dann werden die Parameter 1 bis 5 aus den User Default-Daten verwendet, nicht aus indirekter Adressierung.
	Bit 0 bis 5:	Angabe, welche Parameter indirekt vorgegeben werden
u16TimeOut	-	
eNextCommand	WCD	Ist in diesem Kommando nicht sinnvoll, da dieses Kommando immer in einem Takt abgearbeitet wird.
	IM	Weiterschaltung im nächsten Takt
	IM_NC	Weiterschaltung sofort im gleichen Takt

4.19 CMD_DISABLE_OUTPUT_CAM

Deaktivierung eines Nockens

Mit diesem Kommando können ein oder alle Nocken deaktiviert werden.

Enumerator: CMD_DISABLE_OUTPUT_CAM := 310

Parameter	Wert	Beschreibung
i32IndexTO	-1:	Alle Nocken
	0..LMCIPR_NUMBER_OF_OUTPUT_CAMS -1:	Spezieller Nocken
r32Parameter1	-	
r32Parameter2	-	
r32Parameter3	-	
r32Parameter4	-	
r32Parameter5	-	
r32Parameter6	-	
b8Mode	-	
u16TimeOut	-	
eNextCommand	WCD	Ist in diesem Kommando nicht sinnvoll, da dieses Kommando immer in einem Takt abgearbeitet wird.
	IM	Weiterschaltung im nächsten Takt
	IM_NC	Weiterschaltung sofort im gleichen Takt

Kommandos für Weberschaltbedingungen

Die folgenden Kommandos sind im Servicemode nicht erlaubt.

5.1 CMD_WAIT_TIME

Warten einer Zeitdauer

Mit diesem Kommando wird auf den Ablauf der übergebenen Zeit gewartet. Wird als Zeitdauer 0 ms übergeben, wird für einen Zyklus gewartet.

Enumerator: CMD_WAIT_TIME:= 1010

Parameter	Wert	Beschreibung
i32IndexTO	-	
r32Parameter1	Wartezeit [ms]	Default: 0
r32Parameter2	-	
r32Parameter3	-	
r32Parameter4	-	
r32Parameter5	-	
r32Parameter6	-	
b8Mode		Angabe welche Parameter indirekt vorgegeben werden
u16TimeOut	-	
eNextCommand	-	

5.2 CMD_WAIT_FOR_CONDITION

Warten auf boolesche Bedingung

Mit diesem Kommando wird auf das Eintreten einer booleschen Bedingung gewartet. Die booleschen Bedingungen werden über die DurchgangsvARIABLEN *conditions* bereitgestellt.

Enumerator: CMD_WAIT_FOR_CONDITION:= 1030

Parameter	Wert	Beschreibung
i32IndexTO	-	
r32Parameter1	Index auf Bit im Array <i>conditions</i>	
r32Parameter2	-	
r32Parameter3	-	
r32Parameter4	1: 0:	Weitschaltung bei TRUE Weitschaltung bei FALSE
r32Parameter5	0: 1:	Bedingung bleibt erhalten Bedingung wird invertiert
r32Parameter6	-	
b8Mode		Angabe, welche Parameter indirekt vorgegeben werden
u16TimeOut	0..65535 [ms]	Überwachungszeit für Kommando
eNextCommand	WCD IM IM_NC	Weitschaltung sobald Bedingung erfüllt ist. Ist in diesem Kommando nicht sinnvoll Nur falls Bedingung beim Aufruf bereits erfüllt ist

5.3 CMD_WAIT_FOR_AXIS_STATE

Warten auf Achsstatus

Mit diesem Kommando wird gewartet, bis eine oder alle Achsen einen bestimmten Status besitzen. Dieses Kommando kann, soweit bezüglich des Status sinnvoll, auch für externe Geber verwendet werden. Ist der Status bereits aktiv, wird im gleichen Zyklus das nächste Kommando bearbeitet, wenn IM_NC projiziert und noch nicht das Maximum von Kommandos für einen Interpreter-Zyklus erreicht worden ist.

Hinweis

Der Achs-Status wird aus den Rückmeldungen des integrierten **Achs-FB** gelesen. Wird die angegebene Achse aus irgendeinem Grund außerhalb des **Interpreters** beeinflusst, so werden daraus resultierende Änderungen des Achs-Status im **Interpreter** nicht erkannt.

Hinweis

Die Rückmeldungen der Achsen sind so definiert, dass sie für nicht erreichbare Status TRUE liefern; z. B. eine Drehzahlachse hat immer den Zustand *axisHomed*.

Enumerator: CMD_WAIT_FOR_AXIS_STATE:= 1040

Parameter	Wert	Beschreibung
i32IndexTO	-1:	Alle Achsen
	0..LMCIPR_NUMBER_OF_AXES -1:	Spezielle Achse
r32Parameter1	Achsstatus:	
	0:	Achse freigegeben
	1:	Achse referenziert Referenzieren über <i>redefine position</i> liefert nicht den Achs-Status referenziert (<i>axisHomed</i>)
	2:	Achse synchron
	3:	Position im Positionsfenster
	4:	Geschwindigkeit erreicht
	5:	Achse im Stillstand
6:	Achse nicht freigegeben	
r32Parameter2	-	
r32Parameter3	-	
r32Parameter4	-	
r32Parameter5	-	
r32Parameter6	-	
b8Mode		Angabe, welche Parameter indirekt vorgegeben werden
u16TimeOut	0..65535 [ms]	Überwachungszeit für Kommando

Parameter	Wert	Beschreibung
eNextCommand	WCD	Weitschaltung, sobald Achsstatus erreicht ist.
	IM	Ist bei diesem Kommando nicht sinnvoll
	IM_NC	Nur falls die Bedingung beim Aufruf bereits erfüllt ist

5.4 CMD_WAIT_FOR_POS

Warten auf Erreichen einer Achsposition

Mit diesem Kommando wird gewartet, bis eine Achse eine bestimmte Position erreicht hat. Für die Auswertung des Positionsvergleichs kann eine Toleranz angegeben werden. Dies gilt nur für die Vergleichsart *gleich*. Kann die Toleranz nicht gelesen werden (indirekte Programmierung und die Variable ist außerhalb der Arraygrenzen), so wird der Vergleich mit Defaultwerten (Linearachse: 1 mm, Rundachse: 36°) durchgeführt.

Die Weiterschaltbedingung IM_NC hat nur dann eine Auswirkung, wenn bei Start dieses Kommandos die Bedingung bereits erfüllt ist. Ist dies nicht der Fall, wird immer gewartet, egal welche Weiterschaltbedingung projiziert worden ist.

Hinweis

Der Moduloübergang bei projizierten Modulo-Achsen wird berücksichtigt.

Enumerator: CMD_WAIT_FOR_POS:= 1050

Parameter	Wert	Beschreibung
i32IndexTO	0..LMCIPR_NUMBER_OF_AXES -1:	Spezielle Achse
r32Parameter1	Vergleichsposition	
r32Parameter2	-	
r32Parameter3	-	
r32Parameter4	0:	Vergleich auf <i>größer</i> Achspannung > Vergleichsposition
	2:	Vergleich auf <i>gleich</i> Achspannung = Vergleichsposition
	4:	Vergleich auf <i>kleiner</i> Achspannung < Vergleichsposition
r32Parameter5	Genauigkeitsfenster	Default: 1 mm bzw. 36° Maximaler Abstand Position und Vergleichsposition
r32Parameter6	-	
b8Mode		Angabe, welche Parameter indirekt vorgegeben werden
u16TimeOut	0..65535 [ms]	Überwachungszeit für Kommando
eNextCommand	WCD	Weiterschaltung sobald Vergleich erfüllt ist.
	IM	Ist in diesem Kommando nicht sinnvoll
	IM_NC	Nur falls Bedingung beim Aufruf bereits erfüllt ist

Siehe auch

CMD_JUMP_CMP_POS (Seite 43)

Kommandos für Programmverzweigungen

Die folgenden Kommandos sind im Servicemode nicht erlaubt.

Die Angaben für Verfahrprogramm und Verfahrersatz werden bei Bedarf (Vorgabe von Werten mit Nachkommastellen) gerundet.

Hinweis

Für alle Sprungbefehle gilt: Das nachfolgende Kommando wird noch im selben Interpreter-Takt ausgeführt.

6.1 CMD_JUMP

Unbedingter Sprung

Mit diesem Kommando wird unbedingt zum Kommando des angegebenen Verfahrersatzes und Verfahrersatz-Programms gesprungen.

Enumerator: CMD_JUMP := 3010

Parameter	Wert	Beschreibung
i32IndexTO	-	
r32Parameter1	-	
r32Parameter2	Verfahrersatz-Programm	
r32Parameter3	Verfahrersatz	
r32Parameter4	-	
r32Parameter5	-	
r32Parameter6	-	
b8Mode		Angabe, welche Parameter indirekt vorgegeben werden
u16TimeOut	-	
eNextCommand	-	

6.2 CMD_JUMP_CMP_CONDITION

Bedingter Sprung

Mit diesem Kommando wird bedingt zum Kommando des angegebenen Verfahrssatzes und Verfahrssatz-Programms gesprungen. Die Bedingung ist eine Variable vom Typ BOOL im Array *conditions*.

Enumerator: CMD_JUMP_CMP_CONDITION := 3020

Parameter	Wert	Beschreibung
i32IndexTO	-	
r32Parameter1	Index auf Bit im Array <i>conditions</i>	
r32Parameter2	Verfahrssatz-Programm	
r32Parameter3	Verfahrssatz	
r32Parameter4	1:	Sprung bei TRUE
	0:	Sprung bei FALSE
r32Parameter5	0:	Bedingung bleibt erhalten
	1:	Bedingung wird im Array <i>conditions</i> invertiert
r32Parameter6	-	
b8Mode		Angabe, welche Parameter indirekt vorgegeben werden
u16TimeOut	-	
eNextCommand	-	

6.3 CMD_JUMP_CMP_POS

Sprung bei Positionsvergleich

Mit diesem Kommando wird bedingt gesprungen, wenn der Vergleich der Achsposition mit einer vorgegebenen Position erfüllt ist. Über einen Parameter wird eingestellt wie verglichen wird. Die Toleranz wird nur bei einem Vergleich auf *gleich* verwendet.

Hinweis

Der Moduloübergang bei projektierten Modulo-Achsen wird berücksichtigt.

Enumerator: CMD_JUMP_CMP_POS := 3030

Parameter	Wert	Beschreibung
i32IndexTO	0..LMCIPR_NUMBER_OF_AXES -1:	Spezielle Achse
r32Parameter1	Position für Vergleich	
r32Parameter2	Verfahrstz-Programm	
r32Parameter3	Verfahrstz	
r32Parameter4	0:	Vergleich auf <i>größer</i> Achspannung > Vergleichspannung
	2:	Vergleich auf <i>gleich</i> Achspannung = Vergleichspannung
	4:	Vergleich auf <i>kleiner</i> Achspannung < Vergleichspannung
r32Parameter5	Genauigkeitsfenster	Default: 1 mm bzw. 36° Maximaler Abstand Position und Vergleichspannung
r32Parameter6	-	
b8Mode	-	Angabe, welche Parameter indirekt vorgegeben werden
u16TimeOut	-	
eNextCommand	-	

Siehe auch

CMD_WAIT_FOR_POS (Seite 39)

6.4 CMD_JUMP_CMP_VARIABLE

Sprung bei Wertevergleich

Mit diesem Kommando wird bedingt gesprungen, wenn der Vergleich des Werts im Durchgangsarray *variables* mit dem zu vergleichenden Wert im Parameter 5 erfüllt ist.

Enumerator: CMD_JUMP_CMP_VARIABLE := 3040

Parameter	Wert	Beschreibung
i32IndexTO	-	
r32Parameter1	Index auf REAL-Wert im Array <i>variables</i>	
r32Parameter2	Verfahrensatzprogramm	
r32Parameter3	Verfahrensatz	
r32Parameter4	0:	Vergleich auf <i>größer</i> Achsisposition > Vergleichsposition
	1:	Vergleich auf <i>größer gleich</i> Achsisposition >= Vergleichsposition
	2:	Vergleich auf <i>gleich</i> Achsisposition = Vergleichsposition
	3:	Vergleich auf <i>kleiner gleich</i> Achsisposition <= Vergleichsposition
	4:	Vergleich auf <i>kleiner</i> Achsisposition < Vergleichsposition
r32Parameter5	Vergleichswert	
r32Parameter6	-	
b8Mode		Angabe, welche Parameter indirekt vorgegeben werden
u16TimeOut	-	
eNextCommand	-	

6.5 CMD_JUMP_CMP_STATE

Sprung auf Grund von Zuständen

Mit diesem Kommando wird bedingt auf Grund eines Achszustands gesprungen. Dieses Kommando kann, soweit bezüglich des Status sinnvoll, auch für externe Geber verwendet werden.

Hinweis

Der Achs-Status wird aus den Rückmeldungen des integrierten **Achs-FB** gelesen. Wird die angegebene Achse aus irgendeinem Grund außerhalb des **Interpreters** beeinflusst, so werden daraus resultierende Änderungen des Achs-Status im **Interpreter** nicht erkannt.

Enumerator: CMD_JUMP_CMP_STATE:= 3050

Parameter	Wert	Beschreibung
i32IndexTO	0..LMCIPR_NUMBER_OF_AXES -1:	Spezielle Achse
r32Parameter1	Status:	
	0:	Achse freigegeben
	1:	Achse referenziert nicht für <i>redefinePosition</i>
	2:	Achse synchron
	3:	Position im Positionsfenster
	4:	Geschwindigkeit erreicht
	5:	Achse im Stillstand
	6:	Achse nicht freigegeben
	7:	Fehler der Klasse 4 steht an
r32Parameter2	Verfahrsatz-Programm	
r32Parameter3	Verfahrsatz	
r32Parameter4	-	
r32Parameter5	-	
r32Parameter6	-	
b8Mode		Angabe, welche Parameter indirekt vorgegeben werden
u16TimeOut	-	
eNextCommand	-	

6.6 CMD_LOOP

Schleife

Mit diesem Kommando wird zum Kommando des angegebenen Verfahrssatzes und Verfahrssatz-Programms gesprungen. Mit jedem Durchlauf wird der Wert der Anzahl der Schleifendurchläufe um 1 verringert. Bei einem Wert von 0 wird nicht gesprungen, sondern das dem LOOP-Befehl folgende Kommando ausgeführt. Geschachtelte Schleifen sind möglich. Ein Schleifendurchlauf wird definiert über Verfahrssatz /Verfahrssatz-Programm im Loop-Befehl (Anfang) und dem Kommando CMD_END_LOOP (Ende). Dazwischen sind alle Befehle erlaubt. Ist eine leere Schleife programmiert oder Parameter 1 hat den Wert 0, so wird auf den nächsten Verfahrssatz gewechselt.

In der Unit **cPublic** kann eingestellt werden, wie viele geschachtelte Schleifen maximal möglich sind (Konstante: LMCIPR_NUMBER_OF_LOOPS_PER_SEQUENCE).

Enumerator: CMD_LOOP := 3200

Parameter	Wert	Beschreibung
i32IndexTO	-	
r32Parameter1	Anzahl der Schleifendurchläufe	
r32Parameter2	Verfahrssatz-Programm	Für Schleifenbeginn
r32Parameter3	Verfahrssatz	Für Schleifenbeginn
r32Parameter4	-	
r32Parameter5	-	
r32Parameter6	-	
b8Mode		Angabe, welche Parameter indirekt vorgegeben werden
u16TimeOut	-	
eNextCommand	-	

6.7 CMD_END_LOOP

Ende Kennzeichnung für eine Schleife

Mit diesem Kommando wird das Schleifenende charakterisiert. Der Befehl wird ignoriert, wenn keine Schleife aktiv ist; dabei wird ein entsprechender Eintrag im Diagnosepuffer vorgenommen.

In der Unit **cPublic** kann eingestellt werden, wie viele geschachtelte Schleifen maximal möglich sind (Konstante: LMCIPR_NUMBER_OF_LOOPS_PER_SEQUENCE).

Enumerator: CMD_END_LOOP := 3210

Parameter	Wert	Beschreibung
i32IndexTO	-	
r32Parameter1	-	
r32Parameter2	-	
r32Parameter3	-	
r32Parameter4	-	
r32Parameter5	-	
r32Parameter6	-	
b8Mode	-	
u16TimeOut	-	
eNextCommand	-	

6.8 CMD_START_PARALLEL_SEQUENCE

Start einer parallelen Sequenz

Mit diesem Kommando wird eine neue parallele Sequenz gestartet. Für die maximale Anzahl Sequenzen ist in der Unit **cPublic** die Konstante LMCIPR_NUMBER_OF_PARALLEL_SEQUENCES angelegt.

Enumerator: CMD_START_PARALLEL_SEQUENCE := 3300

Parameter	Wert	Beschreibung
i32IndexTO	-	
r32Parameter1	-	
r32Parameter2	Verfahrsatz-Programm	
r32Parameter3	Verfahrsatz	
r32Parameter4	-	
r32Parameter5	-	
r32Parameter6	-	
b8Mode		Angabe, welche Parameter indirekt vorgegeben werden
u16TimeOut	-	
eNextCommand	-	

6.9 CMD_STOP_SEQUENCE

Stopp von parallelen Sequenzen

Mit diesem Kommando kann entweder eine spezifische Sequenz oder es können alle aktiven Sequenzen (außer einem Fehler-Programm) gestoppt werden. Dabei muss die Sequenz-Nummer vorgegeben werden. Der Wert -1 bedeutet: alle Sequenzen. Die Sequenz-Nummer kann nur durch Durchsicht der aktuellen Programm-Daten ermittelt werden, da nicht zwangsläufig vorhersehbar ist, in welcher Sequenz welches Programm läuft. Der **Interpreter** nimmt das Kommando CMD_START_PARALLEL_SEQUENCE immer für die erste freie Sequenz, die er findet.

Hinweis

Wird dieses Kommando nicht in einem Fehlerprogramm, sondern in einer Sequenz verwendet, so wird bei Vorgabe von -1 in Parameter 2 auch die Sequenz mit dem Stopp-Kommando selbst beendet.

Enumerator: CMD_STOP_SEQUENCE := 3310

Parameter	Wert	Beschreibung
i32IndexTO	-	
r32Parameter1	-	
r32Parameter2	Sequenznummer	Index im Array <i>actualProgramData</i> , die gestoppt werden soll -1.0: alle aktiven Sequenzen
r32Parameter3	-	
r32Parameter4	-	
r32Parameter5	-	
r32Parameter6	-	
b8Mode		Angabe, welche Parameter indirekt vorgegeben werden
u16TimeOut	-	
eNextCommand	-	

Sonstige Kommandos

7.1 CMD_NONE

Kein Kommando

Dieses Kommando signalisiert im Programmbetrieb, dass ein Verfahrstz-Programm beendet ist. Die Verfahrstz-Tabelle ist im Default-Zustand mit diesem Kommando belegt. Es ist keinerlei weitere Aktion damit verbunden.

Hinweis

Das Kommando wird im Servicemode ohne Fehlermeldung ignoriert.

Enumerator: CMD_NONE := 0

Parameter	Wert	Beschreibung
i32IndexTO	-	
r32Parameter1	-	
r32Parameter2	-	
r32Parameter3	-	
r32Parameter4	-	
r32Parameter5	-	
r32Parameter6	-	
b8Mode	-	
u16TimeOut	-	
eNextCommand	-	

7.2 CMD_NOP

Keine Operation

Mit diesem Kommando wird keinerlei Aktion verbunden; man kann es z. B. nutzen, um für Testzwecke bestimmte Programmzeilen vorübergehend auszublenden, ohne dass sich die Programmzeilen dadurch verschieben. Damit können auch alle Parameter erhalten bleiben; es muss nur das Kommando selbst überschrieben werden.

Hinweis

Das Kommando wird im Servicemode ohne Fehlermeldung ignoriert.

Enumerator: CMD_NOP := 1

Parameter	Wert	Beschreibung
i32IndexTO	-	
r32Parameter1	-	
r32Parameter2	-	
r32Parameter3	-	
r32Parameter4	-	
r32Parameter5	-	
r32Parameter6	-	
b8Mode	-	
u16TimeOut	-	
eNextCommand	-	

7.3 CMD_SET_CONDITION

Setzen eines booleschen Ausgangs

Mit diesem Kommando wird ein Bit im Array der DurchgangsvARIABLEN *conditions* gesetzt.

Enumerator: CMD_SET_CONDITION := 2010

Parameter	Wert	Beschreibung
i32IndexTO	-	
r32Parameter1	Index auf Bit im Array <i>conditions</i>	
r32Parameter2	1:	Auf TRUE setzen
	0:	Auf FALSE setzen
	2:	Invertieren
r32Parameter3	-	
r32Parameter4	-	
r32Parameter5	-	
r32Parameter6	-	
b8Mode		Angabe, welche Parameter indirekt vorgegeben werden
u16TimeOut	-	
eNextCommand	WCD	Ist in diesem Kommando nicht sinnvoll
	IM	Weiterschaltung im nächsten Takt
	IM_NC	Weiterschaltung sofort im gleichen Takt

7.4 CMD_SET_VARIABLE

Setzen eines REAL-Werts

Mit diesem Kommando wird ein REAL-Wert im Array der DurchgangsvARIABLEN *variables* gesetzt.

Enumerator: CMD_SET_VARIABLE := 2020

Parameter	Wert	Beschreibung
i32IndexTO	-	
r32Parameter1	Index auf REAL-Wert im Array <i>variables</i>	
r32Parameter2	Wert	
r32Parameter3	-	
r32Parameter4	-	
r32Parameter5	-	
r32Parameter6	-	
b8Mode		Angabe, welche Parameter indirekt vorgegeben werden
u16TimeOut	-	
eNextCommand	WCD	Ist in diesem Kommando nicht sinnvoll
	IM	Weiterschaltung im nächsten Takt
	IM_NC	Weiterschaltung sofort im gleichen Takt

7.5 CMD_INC_VARIABLE

Inkrementieren eines REAL-Werts

Mit diesem Kommando wird ein REAL-Wert im Array der Durchgangsvariablen *variables* um einen bestimmten Wert erhöht.

Enumerator: CMD_INC_VARIABLE := 2030

Parameter	Wert	Beschreibung
i32IndexTO	-	
r32Parameter1	Index auf REAL-Wert im Array <i>variables</i>	
r32Parameter2	Inkrement	
r32Parameter3	-	
r32Parameter4	-	
r32Parameter5	-	
r32Parameter6	-	
b8Mode		Angabe, welche Parameter indirekt vorgegeben werden
u16TimeOut	-	
eNextCommand	WCD	Ist in diesem Kommando nicht sinnvoll
	IM	Weiterschaltung im nächsten Takt
	IM_NC	Weiterschaltung sofort im gleichen Takt

7.6 CMD_CALC_VARIABLE

Berechnen einer Variablen nach einer Formel

Die Berechnungsformel ist nach dem Muster aufgebaut: $y = a_0 + a_1 \cdot x + a_2 \cdot x^2$

Mit x und y sind jeweils Variable gemeint, die über ihren Index im Array *variables* angesprochen werden.

Enumerator: CMD_CALC_VARIABLE := 2040

Parameter	Wert	Beschreibung
i32IndexTO	-	
r32Parameter1	Parameter a0	
r32Parameter2	Parameter a1	
r32Parameter3	Parameter a2	
r32Parameter4	-	
r32Parameter5	Index auf Eingangsgröße x	Im Array <i>variables</i>
r32Parameter6	Index auf Ausgangsgröße y	Im Array <i>variables</i>
b8Mode		Angabe, welche Parameter indirekt vorgegeben werden
u16TimeOut	-	
eNextCommand	WCD	Ist in diesem Kommando nicht sinnvoll
	IM	Weiterschaltung im nächsten Takt
	IM_NC	Weiterschaltung sofort im gleichen Takt

Kontakt

A.1 Ansprechpartner

Siemens AG
Industry Sector
I DT MC PMA APC
Frauenauracher Straße 80
D - 91056 Erlangen
Fax.: +49 9131 98 1297
mailto: tech.team.motioncontrol@siemens.com

A.2 Internetadressen

Weitere Informationen zu den verschiedenen Themen finden Sie auf folgenden Internetseiten.

Siehe auch

SIMOTION (www.siemens.com/simotion)

SINAMICS (www.siemens.com/sinamics)

MotionControl/Applikationszentrum (www.siemens.com/motioncontrol/apc)

Verpackung (www.siemens.com/packaging)

Index

C

CMD_CALC_VARIABLE, 56
CMD_CAM_IN_FOLLOWING_OFFSET, 28
CMD_CAM_IN_LEADING_OFFSET, 27
CMD_CAM_OUT, 29
CMD_DISABLE_OUTPUT_CAM, 33
CMD_ENABLE_OUTPUT_CAM, 32
CMD_END_LOOP, 47
CMD_ESTOP, 22
CMD_GEAR_IN, 25
CMD_GEAR_OUT, 26
CMD_HOMING, 23
CMD_INC_VARIABLE, 55
CMD_JUMP, 41
CMD_JUMP_CMP_CONDITION, 42
CMD_JUMP_CMP_POS, 43
CMD_JUMP_CMP_STATE, 45
CMD_JUMP_CMP_VARIABLE, 44
CMD_LOOP, 46
CMD_MOVE, 20
CMD_NEXT_STEP, 13
CMD_NONE, 51
CMD_NOP, 52
CMD_Phasing, 30
CMD_POS_ABS, 17
CMD_POS_REL, 18
CMD_POS_SUPERIMPOSED, 19
CMD_POWER_OFF, 16
CMD_POWER_ON, 15
CMD_RESET_AXIS, 24
CMD_SET_ACCEL_OVERRIDE, 31
CMD_SET_CONDITION, 53
CMD_SET_VARIABLE, 54
CMD_START_PARALLEL_SEQUENCE, 48
CMD_START_PROGRAM, 11
CMD_STOP, 21
CMD_STOP_PROGRAM, 12
CMD_STOP_SEQUENCE, 49
CMD_WAIT_FOR_AXIS_STATE, 37
CMD_WAIT_FOR_CONDITION, 36
CMD_WAIT_FOR_POS, 39
CMD_WAIT_TIME, 35

