



1 EINLEITENDE BEMERKUNGEN REV. 0.5 (QLA).....	1
1.1 ÄNDERUNGEN IN VERSION V0.5.....	1
1.2 ÄNDERUNGEN UND BESTAND IN VERSION V0.3.....	2
2 EINIGE NOTWENDIGE AUSSAGEN ZUR HAFTUNG	3
3 EINSTELLUNGEN UND KONFIGURATION.....	3
4 EINBINDEN DES BAUSTEINS IN DAS ANWENDERPROGRAMM.....	5
4.1 ENTWURFSPRINZIPIEN.....	5
4.2 BAUSTEINE.....	5
4.3 EXPLIZITE INSTANZIIERUNG ODER INSTANZIIERUNG IM INSTANZ-DB DES AUFRUFENDEN FB'S.....	6
4.4 VARIABLENZUGRIFF DER INSTANZIIERUNGSARTEN, VARIABLENTAFELN.....	8
5 FB-INTERNES (STANDARD) ODER ANWENDERSEITIGES (EXTERNES) AUFRISCHEN DER KOMMUNIKATIONSDATEN.....	9
6 QLA-FB-PARAMETER.....	11
6.1 FORMATWANDLUNGEN.....	11
6.2 EINGÄNGE UND AUSGÄNGE AM QLA-FB.....	11
6.3 STATISCHE INSTANZDATEN DES QLA-FB'S FÜR DEN EXTERNEN ZUGRIFF.....	14
6.4 ÜBERLAPPUNGSSCHUTZ UND TARGETZEIT.....	17
6.5 PROFILPARAMETER DER BEWEGUNGSBEFEHLE IM EINZELNEN.....	17
6.6 LESEN UND SCHREIBEN VON SOFTWARE-ENDGRENZEN UND HOMING-OFFSET.....	19

1 Einleitende Bemerkungen Rev. 0.5 (QLA)

Die Neufassung FB_C3CtrlI20T11QLA des existierenden S7-FB's FB45 für die Steuerung des Parker -Positionierreglers C3I20T11 wurde für den Einsatz in QLA-Maschinen von Parker entwickelt. Die Neufassung wurde erstellt, weil Compax C3I20T11 nach Ersterstellung der S7-Treibersoftware (04/2004, Buyout an Parker) neue Funktionen erhalten hat.



Das Applikationsteam von Parker in Offenburg pflegt eine eigene S7/C3-Treibersoftware. Diese und die folgend beschriebene QLA-Fassung haben divergente Schnittstellen. Die QLA-Version wird vom Parker-Applikationsteam nicht unterstützt. Falls Sie die QLA-Version verwenden möchten, bitten wir um eine kurze schriftliche Mitteilung an support@inmotec.de.



Bei Bedarf sind entgeltliche Quellcodemodifikationen für Ihre Anwendung möglich, ebenso eine ergänzende Programmierung, Vor-Ort-Hilfestellung oder Inbetriebnahme Ihrer Anwendung (Step7, WinCC flexible) an der Maschine.

1.1 Änderungen in Version V0.5

Funktionen (QLA-Version V0.5)

- **NEU** ProfiNet-Unterstützung, Entwicklung mit Simatic Step7 V5.5+SP1
- **NEU** Möglichkeit zum Lesen der 3 Parameter „Positive SW-Endgrenze“, „Negative SW-Endgrenze“ und „Home-Offset“
- **NEU** Möglichkeit zum Schreiben der 3 Parameter „Positive SW-Endgrenze“, „Negative SW-Endgrenze“ und „Home-Offset“ mit einem anschließenden Flashen des C3's, so dass nach einem erneuten Einschalten des C3-Servos (24V ein), die Parameter aktiv sind



Fehlerbereinigung (QLA-Version V0.5)

- **Bugfix** S7-FB: PKW-Übertragungen wurden nicht korrekt mit einer Null-Übertragung auf dem PKE abgeschlossen, was dazu geführt hat, dass die Verzögerungsrampe bei Verwendung von ProfiNet nicht vom C3 übernommen wurde, bei Profibus ist das Verhalten allerdings korrekt gewesen
- **Bugfix** S7-FB: Das Ausgangsbit „bRefOK“ wurde nicht synchron mit dem Start der Referenzierung über das Eingangsbit „bStartHoming“ rückgesetzt, so dass man im Anwenderprogramm das Ausgangsbit „bRefOK“ ignorieren musste, bis es im Statuswort von C3 rückgesetzt war.

1.2 Änderungen und Bestand in Version V0.3

Funktionen (QLA-Version V0.3)

- Absolute Positionierung (MoveAbs),
- Relative Positionierung (MoveRel),
- **NEU** Relativ-additives Positionieren (MoveAdd),
- **NEU** Endlosdrehen (Velocity),
- **NEU** Elektronisches Getriebe (Gearing),
- **NEU** Markenpositionieren: Vorbereitender Befehl für „Weg nach Marke“ (RegMove),
- **NEU** Markenpositionieren: Suchbewegung Marke (RegSearch),
- Referenzieren,
- Stoppen (Verfahrssatz verwerfen) oder Unterbrechen (Halt, Verfahrssatz wieder aufnehmen), die Wirkung Stopp/Halt ist bewegungsartabhängig,
- Hand+/-,
- Achsmanagementfunktionen Bestromen, Entstromen mit Schnellstopprampe, Fehlerquittieren
- **NEU** Digitale Ausgänge (Standard) am C3 setzen/rücksetzen, digitale Eingänge (Standard) von C3 lesen,
- C3-PKW-Objekt lesen/schreiben,
- Statuswerte für aktuelle Position, Geschwindigkeit, Drehmoment,
- **NEU** Bewegungsartabhängige Aktivflags, Kommunikationsfehler usw.,
- Aktuelle C3-Fehlernummer,
- **NEU** Zeitüberwachungen von Referenzieren und Positionieren werden nicht mehr durchgeführt.

Handhabung (QLA-Version V0.3)

- **NEU** Adressvergabe im Hardwarekonfigurator S7 für Ein- und Ausgänge auf verschiedenen Adressen anstelle nur einer gemeinsamen Adresse, was zum Teil die Konfiguration vereinfacht (S7-400),
- **NEU** Neuer Typ PPO14 anstelle PPO13 für mehr Prozessdaten,
- Multiinstanzfähigkeit (direkte Instanz oder Integration in Instanz des aufrufenden Funktionsbausteins) ohne Verwendung globaler Ressourcen (Zeiten, Zähler, Merker),
- Internes (Standard, am internen DP-Master) oder externes Auffrischen (C3 am CP342-5) der Kommunikationsdatenbereiche,
- **NEU** Instanzinitialisierung über expliziten Eingang,
- **NEU** Betriebsartenabhängige Einzelbestätigung von Positionieraufträgen nach Handshake mit C3

Fehlerbereinigung (QLA-Version V0.3)

- **Bugfix** S7-FB: Teilweises Ignorieren des Erststarts nach Einschalten der SPS aufgrund remanenter Datenbits (Startbit im Steuerwort) beseitigt,
- **Bugfix** S7-FB: Zwang zum Anlegen des Stopp-Einganges vor Hand+/- beseitigt,
- **Bugfix** C3I20T11: Referenzsignal kommt zu früh, Workarround, Fehler in alten I20T11-AWL-Programmen



2 Einige notwendige Aussagen zur Haftung

Haftungsausschluss

Die inmotec Automation GmbH (im weiteren inmotec) übernimmt keine Garantie dafür, dass die vorliegenden Module für die Simatic-S7 unter allen Bedingungen einwandfrei funktionieren. Aus aktueller Sicht gibt es generell keine Software, die unter allen Umständen und Anforderungen fehlerfrei arbeitet. Der Hersteller lehnt darum jedwede Haftung für direkte und indirekte Schäden aller Art durch Einsatz der Softwaremodule ab, auch wenn die Module entsprechend den Beschreibungen des vorliegenden Handbuchs verwendet werden.

Produktbeobachtungspflicht

Inmotec versucht im Rahmen der Produktbeobachtungspflicht, Gefahren, die aus dem Einsatz unserer Softwaremodule entstehen, zu erkennen und zu beschreiben. Durch die Komplexität und unsere beschränkte Einsicht in die Anlagen der Endkunden, die auch durch den Einsatz von Produkten Dritter geprägt sind, ist es uns jedoch nicht möglich, alle Gefahren zu erkennen. Desgleichen sind in diesem Handbuch nicht alle Eigenschaften der Softwaremodule beschrieben.


Änderungsrecht

Inmotec nimmt das Recht in Anspruch, das Handbuch und die Softwaremodule jederzeit ohne Vorankündigung zu aktualisieren. Softwaremodule können ebenso ohne Mitteilung gesperrt werden, wenn Gefahren erkannt werden, die einen fehlerfreien Betrieb nicht zulassen. Es besteht generell keine Pflicht, auf Anforderung eines Kunden erkannte Fehler sofort zu beseitigen oder neue Funktionen bereitzustellen.

Copyright

Dieses Handbuch und die zugehörigen Softwaremodule enthalten Informationen, die geistiges Eigentum der Autoren sind. Der autorisierte Nutzer verpflichtet sich, diese Informationen exklusiv für den Betrieb der C3-Positionierregler und deren Integration zu benutzen. Die Vervielfältigung und Weitergabe der Dokumentation oder von Auszügen dieser an Dritte ist nur nach Genehmigung gestattet.

3 Einstellungen und Konfiguration

Die S7-seitige Konfiguration erfolgt mit dem S7-Hardwarekonfigurator. Auf der Tool-CD oder im Internet finden Sie die Parkerdateien für die Konfiguration: Gerätestammdatei *PAR_C320.GSD* und Icon-Datei *CPX3.DIB*. 

Pro C3 ist eine eigene Station anzulegen und mit einer Profibusadresse zu versorgen. Es ist zwingend der PPO-Typ 14 zu verwendet. Pro C3 könnten theoretisch 4 Adressen vergeben werden (zweimal PKW und zweimal PZD jeweils für den Ein- und Ausgangsbereich). Der QLA-FB erwartet aber für den Eingangs- und Ausgangsbereich jeweils zusammenhängende Bereiche (8 Byte PKW, 14 Byte PZD). Sie können also für den Ein- und Ausgangsbereich unterschiedliche Startadressen festlegen, aber für Eingangsbereich und Ausgangsbereich müssen jeweils PKW- und PZD-Bereich hintereinander liegen.



Bei der Parametrierung des QLA-FB's sind jeweils nur die ersten beiden Adressen (die PKW-Adressen) anzugeben – allerdings als Hex-Wert (z.B. W#16#100 entspricht 256).

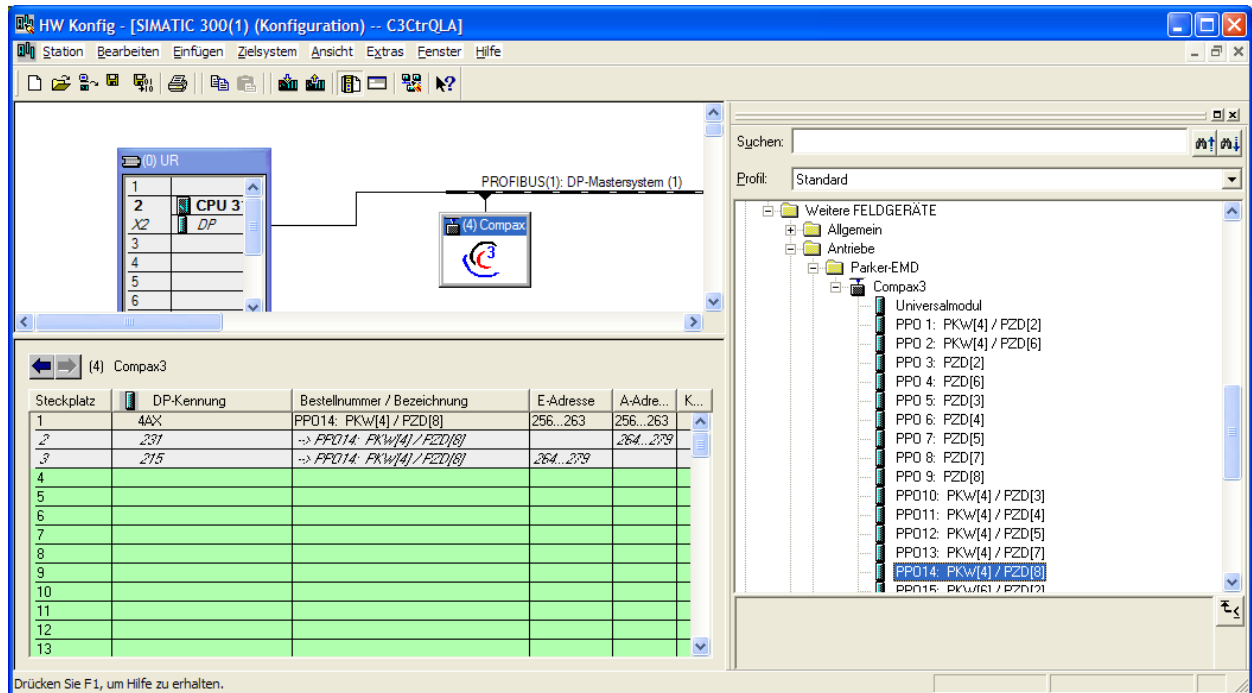


Abbildung 1 S7-Hardwarekonfiguration

Adressierungsbeispiele

E-Adresse PKW	A-Adresse PKW	E-Adresse PZD	A-Adresse PZD
256 .. 263	256 .. 263	264 .. 279	264 .. 279
256 .. 263	512 .. 519	264 .. 279	520 .. 535

Die Profibuskommunikation von C3 ist mit dem C3ServoManager zu projektieren. Ausschließlich die Betriebsart *Direktes Positionieren* wird unterstützt, da diese Betriebsart alle Funktionen abdeckt. Typ *PPO14* ergibt sich automatisch, wenn Sie folgende Einstellungen treffen:

Profibus-Betriebsart	Positionieren
Steuerwort 1	STW 1
1. Sollwert	Lagewollwert XSOLL_A [1100.6]
2. Sollwert	Soll-Verfahrdrehzahl D (32 Bit) [1100.7]
3. Sollwert	Soll-Beschleunigung B (32 Bit) [1111.3]
4. Sollwert	dig. Ausgänge (0 .. 3) [140.3]
Statuswort 1	ZSW 1
1. Istwert	Lageistwert X_IST_A [680.5]
2. Istwert	Drehzahlwert (32 Bit) [681.5]
3. Istwert	Momentenistwert [683.1]
4. Istwert	Aktueller Fehler [550.1]
5. Istwert	dig. Eingänge (0 .. 7) [120.3]



Bedarfsdatenkanal Parameterdatenkanal	PKW
--	-----

Die Einstellung der Fehlerreaktion bei Busausfall obliegt Ihnen. Die Baudrate wird automatisch detektiert, die PB-Adresse wird per Jumper auf der Gerätevorderseite am Busmodul eingestellt.

4 Einbinden des Bausteins in das Anwenderprogramm



Am Anfang sei darauf hingewiesen, dass man Instanz-DB's besser nicht kopiert. Überlassen Sie diese Arbeit dem AWL/FUP/KOP/SCL-Editor. Löschen Sie bei Unklarheiten (Zeitstempelkonflikt) einfach die Instanz-DB's, sie werden vom Editor bei Bedarf neu erzeugt. Kopieren Sie nicht den im QLA-Projekt vorhandenen Instanz-DB in Ihr Anwendungsprojekt. Der QLA-FB kann nicht auf S7-CPU's ohne -2DP laufen, da die benötigten SFC14/15 nicht auf diesen Systemen vorhanden sind. Auf der CPU müssen SFB3 und SFB4 (Timer) vorhanden sein. Der QLA-FB und die UDT's wurden mit S7-SCL erstellt. Das hat keine Auswirkungen auf Ihr Projekt, SCL muss in Ihrer S7-Umgebung nicht installiert sein. Die S7-Quelle ist auch nicht Bestandteil des QLA-Projektes, der QLA-FB ist gegen Einsicht geschützt.

4.1 Entwurfsprinzipien

- Die gesamte Funktionalität ist in *einem* FB (QLA-FB) integriert. Nicht benötigte Softwareteile im FB werden übersprungen, so dass kein Laufzeitnachteil entsteht.
- Binäre Ein- und Ausgänge werden im Ein-/Ausgabebereich (IN, OUT) übergeben, numerische Werte (z.B. Sollposition oder Istposition) jedoch im statischen Bereich (STAT). Das verhindert Netzwerke über mehr als eine Bildschirmseite (bessere Lesbarkeit), zum Teil entfällt auch aufwendiges Umkopieren von Variablen über die Formalparameter.
- Pro Achse ist eine Instanz des QLA-FB's zu erstellen. Bei einer expliziten Instanz ist pro Achse auch ein neuer Instanz-DB zu erzeugen. Bei einer Instanziierung im aufrufenden FB wird der Instanz-DB des aufrufenden FB's automatisch vergrößert.
- Der Bezug zwischen der Instanz eines QLA-FB's und einer „konkreten“ Achse wird über die beiden Adressen aus dem S7-Hardwarekonfigurator realisiert.
- Der C3-FB verwendet keine globalen Ressourcen wie Merker, T-Zeiten und Z-Zähler, so dass diese dem Anwenderprogramm zur vollen Verfügung stehen.

4.2 Bausteine



Folgende Bausteine wurden mit Step7 Professional V5.5 + SP1 erstellt bzw. gepflegt. Stand der C3-Software: C3-ServoManager C3Mgr 2011-R09-30 mit entsprechender Firmware.

Objektname	Symbolischer Name	Erstelsprache (Anzeige)	Beschreibung
FB45	FB_C3CtrlI20T11QLA	SCL	C3-Hantierungsbaustein, SCL-Quelle
FB13	FB_CallC3CtrlI20T11QLA	AWL	Instanziierung, Parametrierung und Aufruf FB45
DB13	IDB_CallC3CtrlI20T11QLA		Instanz-DB FB13
OB1		AWL	Instanziierung und Aufruf FB13
UDT40	ST_C3ControlBits	AWL	Interne Verwendung FB45, Steuerwort STW 1
UDT41	ST_C3StatusBits	AWL	Interne Verwendung FB45, Statuswort ZSW 1
UDT42	ST_C3PKWDint	AWL	Interne Verwendung FB45, PKW-Daten, sinnvoll auch für externes Auffrischen der Komm.-Daten



Objektname	Symbolischer Name	Erstelsprache (Anzeige)	Beschreibung
UDT43	ST_C3PKWInt	AWL	Interne Verwendung FB45, PKW-Daten, Union zu UDT42
UDT44	ST_C3PZDIn	AWL	Interne Verwendung FB45, PZD-Eingangsdaten, sinnvoll auch für externes Auffrischen der Komm.-Daten
UDT45	ST_C3PZDOut	AWL	Interne Verwendung FB45, PZD-Ausgangsdaten, sinnvoll auch für externes Auffrischen der Komm.-Daten

4.3 Explizite Instanziierung oder Instanziierung im Instanz-DB des aufrufenden FB's

Bei der expliziten Instanziierung wird der Instanz-DB für jede Achse einzeln erzeugt und benutzt.

```
CALL "FBC3CtrlI20T11QLA", "IDB_Achse_1"
```

Der Achsname verbirgt sich hier also im Instanz-DB. Der Zugriff auf STAT-Werte (für alle numerischen Parameter) erfolgt z.B. einfach mit

```
L      2300                               // Lade 2.3 mm
T      "IDB_Achse_1".iPosition            // in Zielposition
```

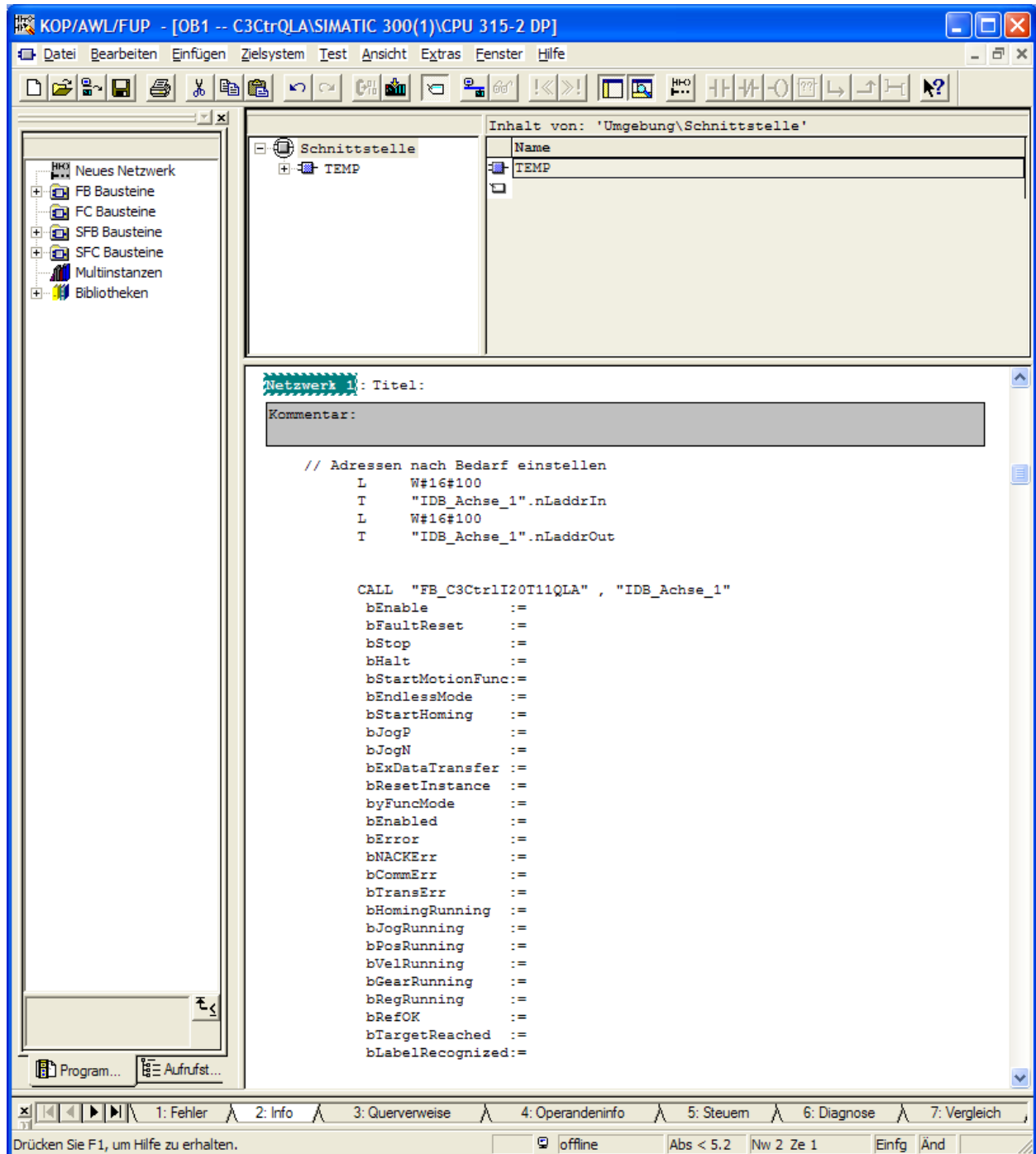


Abbildung 2 Explizite Instanziierung

Bei der Instanziierung im aufrufenden FB wird im Variablenkopf im Bereich der statischen Variablen (STAT) jede neue Achse angelegt. Dazu ist ein Objekt vom Typ FB_C3CtrlI20T11QLA anzulegen. Als Objektname können Sie sinnvolle Name wie z.B. X_Achse oder Achse_1 verwenden. Der Aufruf erfolgt in AWL dann einfach mit:

```
CALL #Achse_1
```

Der Zugriff auf STAT-Werte (für alle numerischen Parameter) erfolgt z.B. einfach mit

```
L      2300           // Lade 2.3 mm
```



T #Achse_1.iPosition // in Zielposition

Name	Datentyp	Adresse	Anfangswert	Ausschluss
Achse_1	FB_C3CtrlI20T11QLA	0.0		<input type="checkbox"/>

```

Netzwerk 1: Aufruf C3-Manager
// Adressen nach Bedarf einstellen
L   W#16#100
T   #Achse_1.nLaddrIn
L   W#16#100
T   #Achse_1.nLaddrOut

CALL #Achse_1
bEnable      :=
bFaultReset  :=
bStop        :=
bHalt        :=
bStartMotionFunc :=
bEndlessMode :=
bStartHoming :=
bJogP        :=
bJogN        :=
bExDataTransfer :=
bResetInstance :=
byFuncMode   :=
bEnabled     :=
bError       :=
bNACKErr     :=
bCommErr     :=
bTransErr   :=
bHomingRunning :=
bJogRunning  :=
bPosRunning  :=
bVelRunning  :=
bGearRunning :=
bRegRunning  :=
bRefOK       :=
bTargetReached :=
bLabelRecognized :=
  
```

Abbildung 3 Instanziierung im Instanz-DB des aufrufenden FB's

4.4 Variablenzugriff der Instanziierungsarten, Variablen tafeln

Die Variablen aus IN/OUT/IN_OUT werden automatisch angezeigt. Nicht jeder dieser Parameter muss mit einem Aktualwert oder einer Aktualvariablen versorgt werden, da alle nicht versorgten Formalparameter einen Standardwert annehmen (siehe Parameterbeschreibung, gilt auch für STAT-Variable).



Vorteil der Instanziierung im Instanz-DB des aufrufenden FB's ist die klare Lesbarkeit und die einfache Instanziierung. Falls aber neue Variablen in der Instanz des FB's eingefügt oder gelöscht werden, verschieben sich die Instanzdaten im Instanz-DB des aufrufenden FB's, d.h. alle Variablen tafeln müssen im folgend korrigiert werden, was man durchaus als Nachteil werten kann.

Vorteil der externen Instanziierung ist die feste Lokalisierung von Instanz-Daten, da sich die Variablen nach der Instanziierung nicht mehr verschieben können. VAT's können bei weiteren Achsen einfach kopiert werden (anschließend DB-Nummern einfach ändern). Nachteil ist



der Achsbezug über Instanz-DB's, was man recht einfach verschmerzen kann.

5 FB-internes (Standard) oder anwenderseitiges (externes) Auffrischen der Kommunikationsdaten



Falls Sie die C3 am internen DP-Master (*bExDataTransfer = False*) betreiben, dürfen Sie dieses Kapitel überspringen.

Falls die Kommunikation zu C3 über eine Kommunikationsbaugruppe (z.B. CP342-5) erfolgt, können die PB-Slavedaten nicht mehr stationsweise (also pro C3) im QLA-FB (über SFC14 DPRD_DAT und SFC15 DPWR_DAT) aufgefrischt werden. Alle Daten dieser Kommunikationsbaugruppe CP müssen über alle PB-Teilnehmer am Bus in einem gemeinsamen Aufruf außerhalb des QLA-FB's realisiert werden. Der Eingang *bExDataTransfer* ist in diesem Fall auf True zu setzen (verhindert das interne Auffrischen mit SFC14/SFC15). Im Folgenden wird gezeigt, wie die Kommunikationsdaten aufzufrischen sind. Über die bereitgestellten UDT's kann symbolisch vorgegangen werden.

Adresse	Name	Typ	Anfangswert	Kommentar
0.0		STRUCT		
+0.0	InAxis11PKW	"ST_C3PKWDint"		
+8.0	InAxis11PZD	"ST_C3PZDIn"		
+24.0	InAxis12PKW	"ST_C3PKWDint"		
+32.0	InAxis12PZD	"ST_C3PZDIn"		
+48.0	InAxis13PKW	"ST_C3PKWDint"		
+56.0	InAxis13PZD	"ST_C3PZDIn"		
=72.0		END_STRUCT		

Abbildung 4 DB8 (hier symbolisch „DATA DB FC2“) mit Kommunikationsdaten aus FC2 für 3 C3-Servos (hier Achsen 11, 12, 13), erstellt mit UDT42 und UDT44

Nach Aufrufen des FC2 (DP_RECV) muss jede einzelne QLA-FB-Instanz aus dem Empfangs-DB von extern aufgefrischt werden.

Bsp. Auffrischen der Eingangsvariablen der QLA-FB-Instanz aus dem DB8 (FC2), hier Achse 11

```
L   "DATA DB FC2".InAxis11PKW.nPKE
T   #Axis11.stC3PKWInDint.nPKE
L   "DATA DB FC2".InAxis11PKW.nIND
T   #Axis11.stC3PKWInDint.nIND
L   "DATA DB FC2".InAxis11PKW.nPlaceholder1
T   #Axis11.stC3PKWInDint.nPlaceholder1
L   "DATA DB FC2".InAxis11PKW.ipWE
T   #Axis11.stC3PKWInDint.ipWE
L   "DATA DB FC2".InAxis11PZD.nStatus
T   #Axis11.stC3PZDIn.nStatus
L   "DATA DB FC2".InAxis11PZD.iActPosition
T   #Axis11.stC3PZDIn.iActPosition
L   "DATA DB FC2".InAxis11PZD.iActVelocity
T   #Axis11.stC3PZDIn.iActVelocity
L   "DATA DB FC2".InAxis11PZD.iActTorque
T   #Axis11.stC3PZDIn.iActTorque
L   "DATA DB FC2".InAxis11PZD.nActError
```



```
T   #Axis11.stC3PZDIn.nActError
L   "DATA DB FC2".InAxis11PZD.nC3DigIn
T   #Axis11.stC3PZDIn.nC3DigIn
```

Adresse	Name	Typ	Anfangswert	Kommentar
0.0		STRUCT		
+0.0	OutAxis11PKW	"ST_C3PKWDint"		
+8.0	OutAxis11PZD	"ST_C3PZDout"		
+24.0	OutAxis12PKW	"ST_C3PKWDint"		
+32.0	OutAxis12PZD	"ST_C3PZDout"		
+48.0	OutAxis13PKW	"ST_C3PKWDint"		
+56.0	OutAxis13PZD	"ST_C3PZDout"		
=72.0		END_STRUCT		

Abbildung 5 DB9 (hier symbolisch „DATA DB FC1“) mit Kommunikationsdaten aus FC1 für 3 C3-Servos (hier Achsen 11, 12, 13), erstellt mit UDT42 und UDT45

Vor Aufrufen des FC1 (DP_SEND) muss der Sende-DB aus der QLA-FB-Instanz von extern aufgefrischt werden.

Bsp. Auffrischen des DB9 (FC1) aus den Ausgangsvariablen der QLA-FB-Instanz, hier Achse 11

```
L   #Axis11.stC3PKWOutDint.nPKE
T   "DATA DB FB1".OutAxis11PKW.nPKE
L   #Axis11.stC3PKWOutDint.nIND
T   "DATA DB FB1".OutAxis11PKW.nIND
L   #Axis11.stC3PKWOutDint.nPlaceholder1
T   "DATA DB FB1".OutAxis11PKW.nPlaceholder1
L   #Axis11.stC3PKWOutDint.ipWE
T   "DATA DB FB1".OutAxis11PKW.ipWE
L   #Axis11.stC3PZDOut.nControl
T   "DATA DB FB1".OutAxis11PZD.nControl
L   #Axis11.stC3PZDOut.iSetPosition
T   "DATA DB FB1".OutAxis11PZD.iSetPosition
L   #Axis11.stC3PZDOut.iSetVelocity
T   "DATA DB FB1".OutAxis11PZD.iSetVelocity
L   #Axis11.stC3PZDOut.iSetAccel
T   "DATA DB FB1".OutAxis11PZD.iSetAccel
L   #Axis11.stC3PZDOut.nC3DigOut
T   "DATA DB FB1".OutAxis11PZD.nC3DigOut
```



Man sollte bei externer Datenauffrischung (*bExDataTransfer = true*) insgesamt folgenden Ablauf programmieren:

1. Aufruf FC2 (für alle PB-Teilnehmer am Bus),
2. Auffrischen aller der QLA-Instanzstrukturen stC3PKWInDint und stC3PZDIn der QLA-FB-Instanz aus dem FC2-DB
3. Aufruf aller QLA-FB's
4. Auffrischen des FC1-DB aus den QLA-Instanzstrukturen stC3PKWOutDint und stC3PZDOut
5. Aufruf FC1

Die Schritte 2 bis 4 sind pro beteiligte Achse (und damit QLA-FB-Instanz) zu wiederholen.



6 QLA-FB-Parameter

6.1 Formatwandlungen



Formate werden vom QLA-FB intern aus Gründen der Laufzeitoptimierung nicht gewandelt, da der Entwickler des QLA-FB's nicht weiß, in welcher Form die Anwenderdaten (Sollwerte) vorliegen werden oder welche C3-Statuswerte vom Anwender wirklich benutzt werden. Wenn Sie die Sollwertdaten bereits im C4_3-Format abspeichern, entfallen „zeitintensive“ Formatwandlungen für die Sollwerte.

C4_3

Linearer Festpunkt Wert mit 3 dezimalen Nachkommastellen. 0 entspricht 0 und 0,001 entspricht 2^0 (0x0000 0001). Aufbau wie Datentyp Integer32, Wertigkeit der Bits um Faktor 1000 vermindert. Länge: 2 Worte.

E2_6

Linearer Festpunkt Wert mit sechs binären Nachkommastellen. 0 entspricht 0, 256 entspricht 2^{14} (0x4000). Länge 1 Wort.

Beispiel C4_3 nach REAL für die aktuelle Position

```
L      #Achse_1.iActPosition           // Instanziierung im aufrufenden FB
// L      "IDB_Achse_1".iActPosition // oder explizite Instanziierung
DTR                                         // Ganzz. (32 Bit) in Gleitpz. (32 Bit)
L      1.000000e+003                    // Faktor 1000 bei C4_3
/R                                           // beseitigen
T      MD      100                       // und abspeichern
```

Beispiel REAL nach C4_3 für eine Sollposition

```
L      1.245600e+001                    // neue Sollposition = 12.456
L      1.000000e+003                    // Faktor 1000 bei C4_3
*R                                           // einführen
RND                                         // runden einer Gleitpunktzahl zur Ganzz.
T      #Achse_1.iPosition               // Instanziierung im aufrufenden FB
// T      "IDB_Achse_1".iPosition      // oder explizite Instanziierung
```

Beispiel E2_6 nach REAL für den Prozentwert des aktuellen Motoriststroms (implizit Drehmoment bzw. Kraft)

```
L      #Achse_1.iActTorque              // Instanziierung im aufrufenden FB
// L      "IDB_Achse_1".iActTorque     // oder explizite Instanziierung
ITD                                         // Ganzz. 16 Bit in Ganzz. 32 Bit
DTR                                         // Ganzz. (32 Bit) in Gleitpz. (32 Bit)
L      6.400000e+001                    // Fakt. 64 b. E2_6 für Prozentdarstellung
/R                                           // beseitigen
T      MD      104                       // und abspeichern
```

6.2 Eingänge und Ausgänge am QLA-FB

Eingänge (Binäre Befehle und Betriebsart)

Parameter	Bereich	Daten- typ	Beschreibung	
bEnable	IN	BOOL	0→1	Bestromen



Parameter	Bereich	Datentyp	Beschreibung	
bFaultReset	IN	BOOL	0→1	Fehler am C3 rücksetzen
bStop	IN	BOOL	1	Stoppen mit „STOPP-Verzögerung“, Verfahrtsatz wird verworfen bei allen Bewegungsarten und Hand und MN-Suchen <i>C3Mgr-Konfiguration</i> Die STOP-Verzögerung [Einheit/s ²] wirkt bei Aktivieren des bStop-Einganges am C3Ctrl QLA-FB. Rück für STOP, HAND und MN-Fahrt [Einheit/s ³]
bHalt	IN	BOOL	1	Zwischenhalten bei Positionierungen MoveAbs, MoveRel, MoveAdd, RegMove, RegSearch mit Verfahrtsatz beibehalten, Verfahrtsatz verwerfen (wie Stoppen) bei Velocity, Gearing Wirkt <u>nicht</u> bei Referenzfahrt oder Hand+/- . Die zuletzt benutzte Verzögerung sollte wirken!
bStartMotionFunc	IN	BOOL	0→1	Starten der Bewegungsfunktion mit angewählten Profilwerten <i>Verwendung</i> MoveAbs, MoveRel, MoveAdd, Velocity, Gearing, RegSearch/RegMove (beide gemeinsam)
bEndlessMode	IN	BOOL	1	Rücksetzbetrieb (vor jedem Neustart wird die Position zurückgesetzt, nicht zu verwechseln mit der Rücksetzstrecke aus der C3Mgr-Konfiguration <i>Verwendung</i> MoveRel
bStartHoming	IN	BOOL	0→1	Starten der Referenzfahrt, Stoppen mit bStop-Eingang <i>C3Mgr-Konfiguration</i> <u>Maschinennull-Geschwindigkeit</u> [Einheit/s] <u>Maschinennull-Beschleunigung</u> [Einheit/s ²] Rück für STOP, HAND und MN-Fahrt [Einheit/s ³] <u>Maschinennull-Mode</u> Achtung: MN-Anfahren nach MN-Fahrt DEAKTIVIEREN, wenn das Verhalten zu CPXM/S analog sein soll.
bJogP, bJogN	IN	BOOL	0→1	Hand+/- auf SW-Endgrenze <i>C3Mgr-Konfiguration</i> <u>HAND-Geschwindigkeit</u> [Einheit/s] <u>HAND-Beschleunigung/-Verzögerung</u> [Einheit/s ²] Rück für STOP, HAND und MN-Fahrt [Einheit/s ³]
bExDataTransfer	IN	BOOL	1	Externer Datenaustausch über FC2/FC1
bReadLimAOff	IN	BOOL	0→1	Lesen der Parameter für „Positive SW-Endgrenze“, „Negative SW-Endgrenze“ und „Maschinennull-Offset“ Voraussetzung: Antrieb wird momentan nicht bewegt.
bWriteLimAOff	IN		0→1	Schreiben der Parameter für „Positive SW-Endgrenze“, „Negative SW-Endgrenze“ und „Maschinennull-Offset“, anschließendes automatisches Flashen der Antriebes Voraussetzung: Antrieb wird momentan nicht bewegt, besser noch, der Antrieb ist entstromt
bResetInstance	IN	BOOL	1	Statisches Rücksetzen aller QLA-FB-Instanzen und Schnellstopp analog zur fallenden Flanke von bEnable,



Parameter	Bereich	Daten-typ	Beschreibung	
				keine weitere Befehlsbearbeitung. Dieses Bit kann im Anlauf-OB für alle QLA-Instanzen gemeinsam gesetzt werden und nach dem erstmaligen Aufruf statisch zurückgesetzt werden. Damit werden nach einem SPS-Start einige für die Steuerung wichtige remanente Instanzwerte auf Standardwerte gesetzt.
byFuncMode	IN	BYTE	0 1 2 3 4 5	Bewegungsmodus <i>MoveAbs</i> , Anwahl absolute Positionierungen <i>MoveRel</i> , Anwahl relativer Positionierungen (Kettenmaß) <i>MoveAdd</i> , Anwahl einer rel. Pos. nach Ende der akt. Bew. <i>Velocity</i> , Endlosdrehen (Standard mit akt. Lageregler) <i>Gearing</i> , Elektronisches Getriebe Distanz nach Marke vorgeben und Markensuchfahrt über Maximaldistanz auslösen (eigentlich zwei Befehle <i>RegMove</i> und <i>RegSearch</i> , wird vom QLA-FB aber selbstständig organisiert), Suchfenster kann außerdem vorgegeben werden

Ausgänge (Binäre Statuswerte)

Parameter	Bereich	Daten-typ	Beschreibung	
bEnabled	OUT	BOOL	1	Status, C3-Endstufe ist bestromt
bError	OUT	BOOL	1	Status, C3 mit Fehler
bCommErr	OUT	BOOL	1	Status, Fehler bei der internen (Standard) Kommunikation, mindestens einer der insgesamt 4 Aufrufe von SFC14/SFC15 ist scheitert
bTransErr	OUT	BOOL	1	Status Fehler bei der PKW-Übertragung
bHomingRunning	OUT	BOOL	1	Status, Acknowledge, Referenzfahrt läuft
bJogRunning	OUT	BOOL	1	Status, Acknowledge, Hand+/- läuft
bPosRunning	OUT	BOOL	1	Status, Acknowledge, Bewegung (<i>MoveAbs</i> , <i>MoveRel</i> , <i>MoveAdd</i>) läuft
bVelRunning	OUT	BOOL	1	Status, Acknowledge, Bewegung (<i>Velocity</i>) läuft
bGearRunning	OUT	BOOL	1	Status, Acknowledge, Bewegung (<i>Gearing</i>) läuft
bRegRunning	OUT	BOOL	1	Status, Acknowledge, Bewegung (<i>RegSearch/RegMove</i>) läuft
bNackErr	OUT	BOOL	1	Status, Not Acknowledge, Positionierungen (alle Arten), Befehl nicht angenommen (kein Handshake von C3)
bRefOK	OUT	BOOL	1	Status, Bezug OK, Achtung bei Abs.-Gebern muss das Teachen über den PB erfolgt sein (liegt an C3-AWL-Programm)
bTargetReached	OUT	BOOL	1	Status, <i>MoveAbs</i> , <i>MoveRel</i> , <i>MoveAdd</i> : Position erreicht Status, <i>Velocity</i> : auf Drehzahl Status, <i>Gearing</i> : Folgedrehzahl/-position erreicht Status, <i>RegMove</i> , <i>RegSearch</i> : <i>RegMove</i> beendet (Marke gefunden) oder <i>RegSearch</i> beendet (Marke nicht gefunden)
bLabelRecognized	OUT	BOOL	1	Status, Marke erkannt bei, gilt nur für <i>RegMove/RegSearch</i>



Parameter	Bereich	Datentyp	Beschreibung	
bReadLimAOffDone	OUT	BOOL	1	Parameter für „Positive SW-Endgrenze“, „Negative SW-Endgrenze“ und „Maschinennull-Offset“ erfolgreich gelesen
bWriteLimAOffDone	OUT	BOOL	1	Parameter für „Positive SW-Endgrenze“, „Negative SW-Endgrenze“ und „Maschinennull-Offset“ erfolgreich geschrieben und geflasht

6.3 Statische Instanzdaten des QLA-FB's für den externen Zugriff

E/A-Konfiguration entsprechend S7-Hardwarekonfiguration

Parameter	Bereich	Datentyp	Beschreibung	
nLaddrIn	STAT	WORD	Standard W#16#100	C3-Eingangsadresse aus dem S7-Hardwarekonfigurator, HEX-Wert, Standard = W#16#100
nLaddrOut	STAT	WORD	Standard W#16#100	C3-Ausgangsadresse aus dem S7-Hardwarekonfigurator, HEX-Wert, Standard = W#16#100

Numerische Sollwerte C3

Parameter	Bereich	Datentyp	Beschreibung	
iPosition	STAT	DINT	Standard 0	Sollposition/Solldistanz im Format C4_3 [Nutzereinheiten * 1000], keine Wertebereichsprüfung <i>Verwendung</i> MoveAbs, MoveRel, MoveAdd, RegMove (Weg nach Marke)
iVelocity	STAT	DINT	Standard 1000	Profilgeschwindigkeit im Format C4_3 [Nutzereinheiten/s * 1000] <i>Verwendung</i> MoveAbs, MoveRel, MoveAdd, Velocity, RegMove (Geschwindigkeit nach Marke)
iAcceleration	STAT	DINT	Standard 10	Profilbeschleunigung im Format U32 [Nutzereinheiten/s ²] <i>Verwendung</i> MoveAbs, MoveRel, MoveAdd, Velocity, Gearing, RegMove <i>C3Mgr-Konfiguration</i> Beschleunigungsaufbau [Nutzereinheiten/s ³] über <u>Ruck für Positionierung</u>
iDeceleration	STAT	DINT	Standard 10	Profilverzögerung im Format U32 [Nutzereinheiten/s ²] <i>Verwendung</i> MoveAbs, MoveRel, MoveAdd, RegMove, RegSearch <i>C3Mgr-Konfiguration</i> Verzögerungsaufbau [Nutzereinheiten/s ³] über <u>Ruck für Positionierung</u>
iInPosWindowAbs	STAT	DINT	Standard 1000	In-Positionsfenster im Format C4_3 [Nutzereinheiten * 1000] <i>Verwendung</i> MoveAbs + bEndlessMode = false und keine Rücksetzstrecke bei der <i>C3Mgr-Konfiguration</i> definiert



Parameter	Bereich	Daten- typ	Beschreibung	
iNumerator	STAT	DINT	Standard 1000	Gearing-Zähler im Format C4_3 [* 1000] <i>Verwendung</i> Gearing
iDenominator	STAT	DINT	Standard 1000	Gearing-Nenner im Format C4_3 [* 1000] <i>Verwendung</i> Gearing
iRegSearchDistance	STAT	DINT	Standard 100000	Maximaler Suchweg bis Marke im Format C4_3 [Nutzereinheiten * 1000] <i>Verwendung</i> RegSearch
iRegSearchVelocity	STAT	DINT	Standard 1000	Suchgeschwindigkeit bis Marke im Format C4_3 [Nutzereinheiten/s * 1000] <i>Verwendung</i> RegSearch
iRegSearchAcceleration	STAT	DINT	Standard 10	Suchbeschleunigung im Format U32 Nutzereinheiten/s ² <i>Verwendung</i> RegSearch
iRegStartIgnore	STAT	DINT	Standard 0	Start Markenignorierfenster m Format C4_3 [Nutzereinheiten * 1000] <i>Verwendung</i> RegSearch
iRegStopIgnore	STAT	DINT	Standard 50000	Ende Markenignorierfenster m Format C4_3 [Nutzereinheiten * 1000] <i>Verwendung</i> RegSearch
diPosLimP	STAT	DINT	Standard 0	Pos. SW-Endgrenze m Format C4_3 [Nutzereinheiten * 1000] <i>Verwendung</i> Positive Software-Endgrenze
diPosLimN	STAT	DINT	Standard 0	Neg. SW-Endgrenze m Format C4_3 [Nutzereinheiten * 1000] <i>Verwendung</i> Negative Software-Endgrenze
diHomeOffset	STAT	DINT	Standard 0	Home-Offset m Format C4_3 [Nutzereinheiten * 1000] <i>Verwendung</i> Bei -1.0 * Home-Offset als aktuelle Position bleibt der Antrieb nach dem Homing stehen (MN-Fahrt), wenn die Checkbox „MN-Punkt anfahren nach MN-Fahrt“ im Servomanager deaktiviert wurde.

Numerische Statuswerte C3

Parameter	Bereich	Daten- typ	Beschreibung	
iActPosition	STAT	DINT	Standard 0	Istposition im Format C4_3 [Nutzereinheiten * 1000]
iActVelocity	STAT	DINT	Standard	Aktuelle Geschwindigkeit C4_3 [Nutzereinheiten/s * 1000]



Parameter	Bereich	Daten- typ	Beschreibung	
			0	
iActTorque	STAT	INT	Standard 0	Aktueller Motoriststrom E2_6, Prozentwert (+/- %)

Fehlerbeschreibung C3

Parameter	Bereich	Daten- typ	Beschreibung	
nActError	STAT	WORD	Standard W#16#1	Aktueller Fehler als HEX-Wert, W#16#0001 = kein Fehler

Standard-E/A-Zugriff C3

Parameter	Bereich	Daten- typ	Beschreibung	
nC3DigIn	STAT	WORD		Status der C3-Standardeingänge (E0 .. E7)
nC3DigOut	STAT	WORD		Setzwert der C3-Standardausgänge (A0 .. A3)

PKW-Kanal

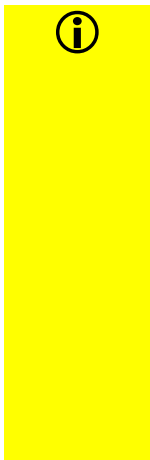
Parameter	Bereich	Daten- typ	Beschreibung	
nCmd	STAT	INT	0 1 2 3 0 -1	Übertragungsart der PKW-Parameter <i>Schreibend (Auftrag an C3)</i> Kein Befehl Lese Parameter Schreibe Parameter (WORD) Schreibe Parameter (DWORD) <i>Lesend (Antwort von C3, nCmd wird überschrieben)</i> Schreib- oder Lesebefehl korrekt ausgeführt Schreib- oder Lesebefehl mit Fehler
nPNUIndex	STAT	INT		PNU-Index aus C3-Online-Hilfe
nPNUSubindex	STAT	INT		PNU-Subindex aus C3-Online-Hilfe <i>DPV0/Profdriveprofil Version 2</i> Die bei der Übertragung nach DPV0 und Profdriveprofil Version 2 notwendige Erhöhung vom Subindex um 1 wird vom QLA-FB selbst erledigt!
iParameterValue	STAT	DINT		Parametersollwert (rechtsbündig) für Schreibbefehl oder Parameteristwert (rechtsbündig) nach Lesebefehl <i>Datenübernahme beim Lesen</i> Übernahme mit nCmd = 0.

6.4 Überlappungsschutz und Targetzeit

Überlappungsschutz und Targetzeit



Parameter	Bereich	Daten-typ	Beschreibung	
OverlappingTimeAck	STAT	TIME	Standard T#0.1s	<p>Verzögerungszeit (startet mit bStartMotionFunc), nach der das interne Bit bSetPointAck im C3-Statuswort abgefragt wird, bei erfolgreicher Befehlsübernahme werden bewegungsabhängig die Acknowledge-Anzeigen</p> <p><i>bPosRunning</i> (MoveAbs, MoveRel, MoveAdd) <i>bVelRunning</i> (Velocity) <i>bGearRunning</i> (Gearing) <i>bRegRunning</i> (RegSearch, RegMove)</p> <p>gesetzt. Bei abgelehntem Bewegungsbefehl wird das Bit <i>bNackErr</i> (MoveAbs, MoveRel, MoveRel, Velocity, Gearing, RegSearch, RegMove)</p>
OverlappingTimeTarget	STAT	TIME	Standard T#0.1s	<p>Verzögerungszeit (startet nach Ablauf der Zeit OverlappingTimeAck), verhindert, dass TargetReached zu früh ausgegeben wird (dieses Signal steht bei Starten der Bewegung immer noch im C3-Statuswort an, bedingt durch Buslaufzeit und dass sich die C3-Achse bei Bewegungstart noch im aktuellen Positionierfenster befindet)</p> <p>C3Mgr-Konfiguration Bei der Konfiguration kann ebenfalls eine <u>Positionierfensterzeit</u> angegeben werden. Diese Zeit läuft C3-intern ab und läuft los, wenn C3 in die Positionierzone eingelaufen ist. Die Zeit OverlappingTimeTarget läuft gegebenenfalls parallel zur Positionierfensterzeit.</p>



Die Bewegung wird normalerweise im ersten SPS-Zyklus nach Setzen des Eingangs *bStartMotionFunc* gestartet (falls keine vorherige PKW-Übertragung (z.B. für einen geänderten Verzögerungswert) erforderlich ist).

Jede neue Bewegung, auch falls eine laufende Bewegung „überschrieben“ wird, löst den Mechanismus erneut aus, z.B. wird bei einer bereits laufenden Positionierung (*bPosRunning* = true) für die Zeit *OverlappingTimeAck* *bPosRunning* = false gesetzt.

Falls die Summe = *Overlapping-TimeAck* + *Overlapping-TimeTarget* < der Positionierzeit oder Geschwindigkeitsänderung ist, kommt es durch die beiden Handshake-Zeiten zu keiner Verzögerung!

Beide Zeiten können bei Bedarf (sehr kurze Positionierungen) vom Anwender unter Beachtung von SPS-Zykluszeit und Buslaufzeit verändert (verkleinert) werden. Die minimale Größe beider Zeiten ist jeweils T#10ms.

6.5 Profilparameter der Bewegungsbefehle im Einzelnen

Punkt-zu-Punkt-Positionierungen mit MoveAbs (Modus 0), MoveRel (Modus 1), MoveAdd (Modus 2)

Parameter	Übertragung	Beschreibung
iPosition	Prozesskanal	Position oder Distanz
iVelocity	Prozesskanal	Geschwindigkeit
iAcceleration	Prozesskanal	Beschleunigung
iDeceleration	PKW-Kanal (änderungsgetriebene Übertragung)	Verzögerung
iInPosWindowAbs	Lokale Variable	In-Positionsfenster für absolute Positionierungen (Mo-



Parameter	Übertragung	Beschreibung
		veAbs) ohne Endlosbetrieb (bEndlessMode = false) und ohne Rücksetzstrecke (<i>C3Mgr-Konfiguration</i>). Zusätzliche Sicherheit durch doppelte Auswertung (Positionsvergleich und In-Positions-Meldung aus dem C3-Statuswort)



Der Endlosbetrieb sollte nur bei MoveRel verwendet werden.

Endlosdrehen mit Velocity (Modus 3)

Parameter	Übertragung	Beschreibung
iVelocity	Prozesskanal	Geschwindigkeit
iAcceleration	Prozesskanal	Beschleunigung und Verzögerung

Synchronfolgen mit Gearing (Modus 4)

Parameter	Übertragung	Beschreibung
iNumerator	Prozesskanal	Gearing-Zähler
iDenominator	Prozesskanal	Gearing-Nenner
iAcceleration	Prozesskanal	Max. Beschleunigung oder Verzögerung bei Änderung der Mastergeschwindigkeit oder Veränderung des Getriebefaktors



Bei Velocity und Gearing ist bei der *C3Mgr-Konfiguration* eine Rücksetzstrecke zu definieren, da es ansonsten an den Software-Endgrenzen zu einer Richtungsumkehr kommen kann.

Markenpositionieren mit RegSearch/RegMove (Modus 5)

Parameter	Übertragung	Beschreibung
iPosition	Prozesskanal	Weg nach Marke (RegMove)
iVelocity	Prozesskanal	Geschwindigkeit für Weg nach Marke (RegMove)
iAcceleration	Prozesskanal	Beschleunigung für Weg nach Marke (RegMove)
iDeceleration	PKW-Kanal (änderungsgetriebene Übertragung)	Verzögerung für Weg nach Marke (RegMove) und Markensuchen (RegSearch)
iRegSearchDistance	Prozesskanal	Maximale Suchdistanz (RegSearch)
iRegSearchVelocity	Prozesskanal	Suchgeschwindigkeit (RegSearch)
iRegSearchAcceleration	Prozesskanal	Beschleunigung für Markensuchen (RegSearch)
iRegStartIgnore	PKW-Kanal (änderungsgetriebene Übertragung)	Beginn Fenster Markenausblendung (RegSearch)
iRegStopIgnore	PKW-Kanal (änderungsgetriebene Übertragung)	Ende Fenster Markenausblendung (RegSearch)



Der QLA-FB enthält eine interne Schrittkette, die die Abfolge RegMove (Bewegung nach Marke) als erste Übertragung und RegSearch (Bewegung zum Markensuchen) als zweite



Übertragung automatisch abhandelt.

In der getesteten C3-Version (C3-Firmwarestand C3_2007R05-3) ist eine Realisierung des „Schnellen Starts“ aus dem Stillstand (COMPAX-M/S, quasi-verzögerungsfreier Start nach Aktivieren eines Eingangs) nicht möglich, da *iRegSearchVelocity* minimal mit 0.001 Einheiten/s vorgegeben werden kann. Der Antrieb „schleicht“ also z.B. mit 0.001 mm/s, bis die Marke erkannt wird oder die maximale Suchdistanz erreicht ist.

6.6 Lesen und Schreiben von Software-Endgrenzen und Homing-Offset

Mit den Befehlsbits bReadLimAOff und bWriteLimAOff können die entsprechenden Parameter im C3 gelesen und geschrieben (inklusive Flashen)

Lesen

- aktuelle Istwerte aus C3 stehen in diPosLimP, diPosLimN und diHomeOffset

Schreiben

- die neuen Sollwerte müssen vor dem Schreiben/Flashen in diPosLimP, diPosLimN und diHomeOffset stehen



Die Offsets der Variablen kann man am besten (und aktuellsten) mit dem DP-Parameter-Editor (Instanzdaten des FB's) ermitteln.

Es gibt nur einen PKW-Kanal, der beim Positionen zum Übertragen benutzt wird (z.B. für die Verzögerungsrampe und das Markenfenster). Daher muss der Antrieb beim Lesen stehen und beim Schreiben sollte der Antrieb nachh Möglichkeit zusätzlich entstromt sein.