

<Bedienungsanleitung\_Basic Gripping\_TC2\_V1\_21(DE).docx>

Thema: <FB Basic Gripping>

Version: <1>

Status: <21>

### Historie

Bearbeiter	Änderungsgrund/Änderungsumfang	Version	Status	Freigabedatum
Armbruster	Grundversion	1	00	07.03.2018
Nock	Anzeige Parameteränderung mit Ausgangsbit Automatischer Reset der Richtungsmerker	1	21	01.04.2020

## Inhalt

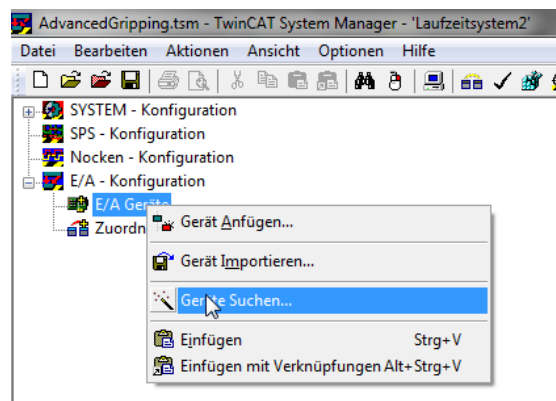
1	Vorwort.....	4
2	Erstellung einer System Konfiguration mit dem System Manager .....	4
3	PLC Control .....	6
4	Verwendung des Funktionsbausteins.....	8
5	Funktionen des Funktionsbausteins .....	9
5.1	Schrittkette zurücksetzen „cmd_b_StepReset“ (BOOL).....	9
5.2	Datenübertragung mit Handshake „cmd_b_DataTransfer“ (BOOL) .....	9
5.3	Werkstückrezeptur speichern „cmd_b_WritePDU“ (BOOL).....	9
5.4	Richtungsmerker zurücksetzen „cmd_b_ResetDirectionFlag“ (BOOL).....	9
5.5	Fahren auf BasePosition „cmd_b_MoveToBase“ (BOOL) .....	10
5.6	Fahren auf WorkPosition „cmd_b_MoveToWork“ (BOOL) .....	10
5.7	Begrenzung der Bewegungszeit „t_MotionTimeout“ (TIME) und „b_MotionError“ (BOOL).....	10
5.8	Datenübertragung wird benötigt „b_DataTransferRequired“ (BOOL).....	10
5.9	Fehler in der Datenübertragung „b_DataTransferError“ (BOOL).....	10
5.10	Befehl wird ausgeführt „b_StepBusy“ (BOOL).....	10
5.11	Bereit für Befehle „b_StepDone“ (BOOL) .....	10
5.12	Bit 6 des StatusWord „b_GripperPLCActive“ (BOOL) .....	10
5.13	Bit 8 des StatusWord „b_BasePosition“ (BOOL) .....	10
5.14	Bit 9 des StatusWord „b_TeachPosition“ (BOOL) .....	11
5.15	Bit 10 des StatusWord „b_WorkPosition“ (BOOL) .....	11
5.16	Bit 11 des StatusWord „b_UndefinedPosition“ (BOOL).....	11
5.17	Bit 12 des StatusWord „b_DataTransferOK“ (BOOL) .....	11
5.18	Bit 13 des StatusWord „b_ControlWord_100“ (BOOL).....	11
5.19	Bit 14 des StatusWord „b_ControlWord_200“ (BOOL).....	11
5.20	Bit 15 des StatusWord „b_Error“ (BOOL) und „n_Diagnose“ (UINT).....	11
5.21	n_ActualPosition (UINT) .....	11

### 1 Vorwort

Für die Nutzung des Beispielprogramms muss zuerst eine korrekte Hardwarekonfiguration erstellt werden. In diesem Beispiel wird eine Beckhoff CX9020 mit einem Beckhoff IO Link Master verwendet. Nach den Hardwareeinstellungen kann das Beispielprojekt implementiert werden. Dazu gehen Sie die nachfolgenden Schritte durch.

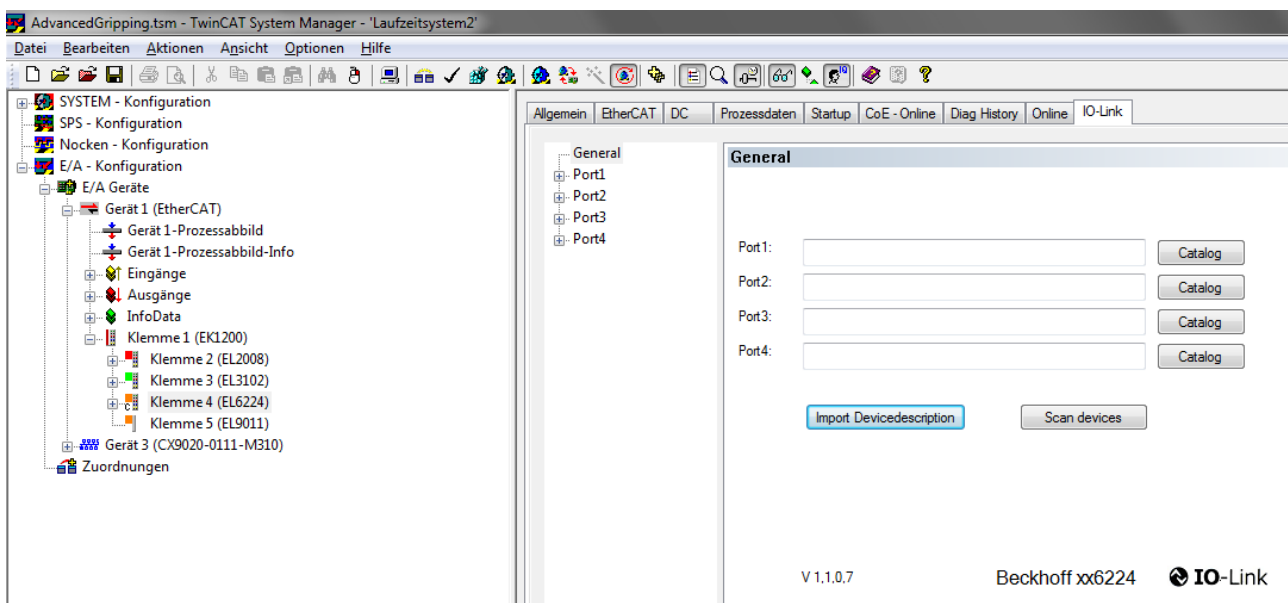
### 2 Erstellung einer System Konfiguration mit dem System Manager

Sobald die angeschlossene Steuerung in den KonfigMode versetzt wurde, kann nach Geräten gesucht werden.

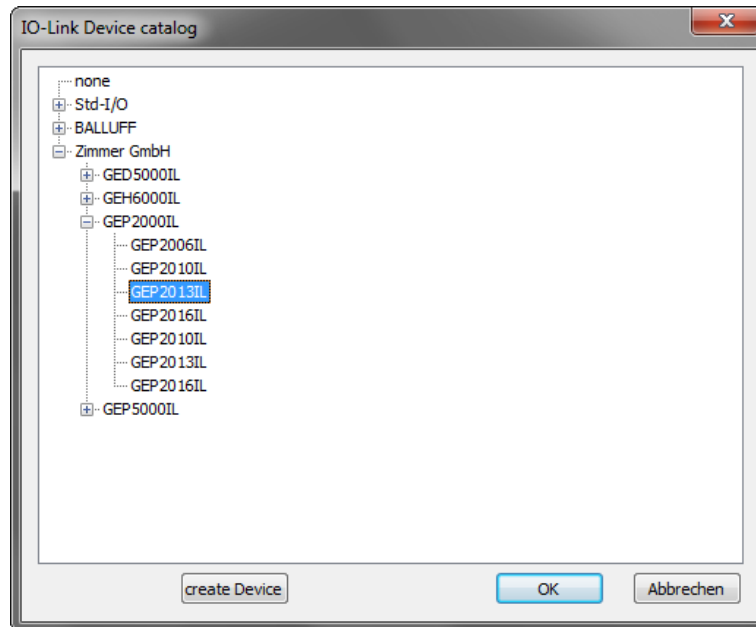


Hierbei sollten nun die angeschlossenen Geräte gefunden und aufgelistet werden. In unserem Beispiel befindet sich der IO Link Master (Klemme 4 / EL6224) auf dem ersten Modul unter der Klemme 1.

Der IO Link Master bietet die Funktion, den angeschlossenen Greifer zu identifizieren. Hierfür muss man in dem Reiter „IO Link“ im Menü des IO Link Masters auf den Button „Scan devices“ klicken. Anschließend öffnet sich ein weiteres Fenster, in welchem aufgelistet wird, an welchem Port, welcher Greifer angeschlossen wurde.

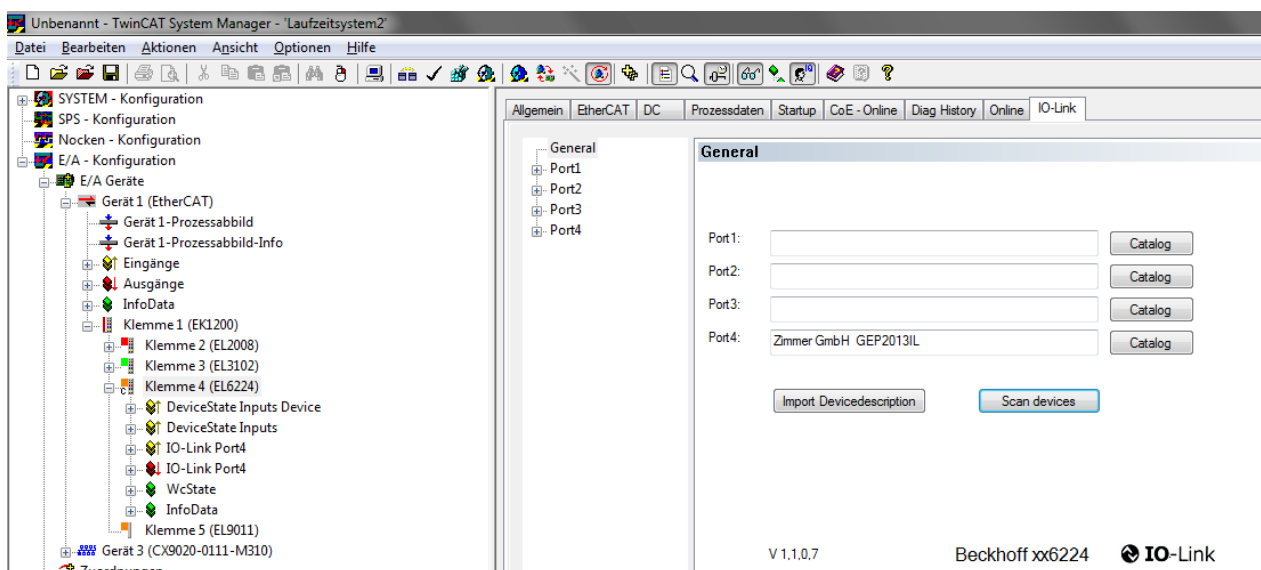


Falls nicht der richtige Greifer erkannt wurde, oder Sie die Systemkonfiguration ohne angeschlossenen Greifer durchführen wollen, kann der Greifer auch manuell eingefügt werden. Hierfür klicken Sie neben dem gewünschten Port auf den Button „Catalog“. Nun öffnet sich der „IO-Link Device catalog“, in welchem die verschiedenen Greifer aufgelistet werden.



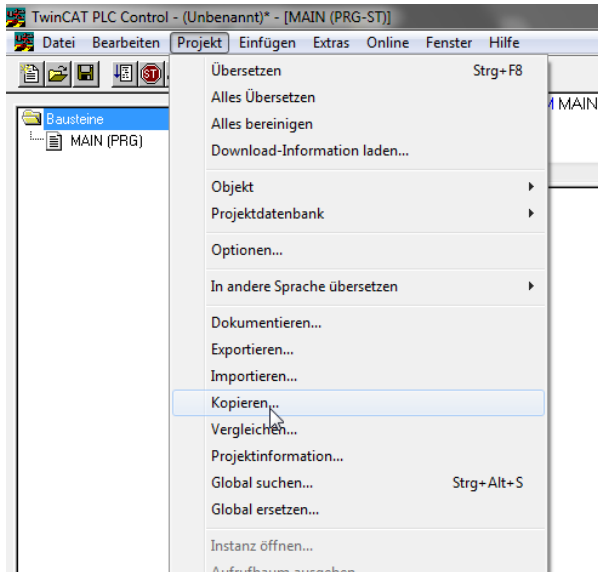
Ist der benötigte Greifer nicht in der Auflistung, kann dieser mit einem Klick auf den Button „Import Devicedescription“ eingefügt werden. Alle IODD Dateien für unsere Greifer können Sie auf unserer Homepage herunterladen.

Sobald der gewünschte Greifer an dem richtigen Port eingefügt wurde, wird der Greifertyp unter dem konfigurierten Port angezeigt.



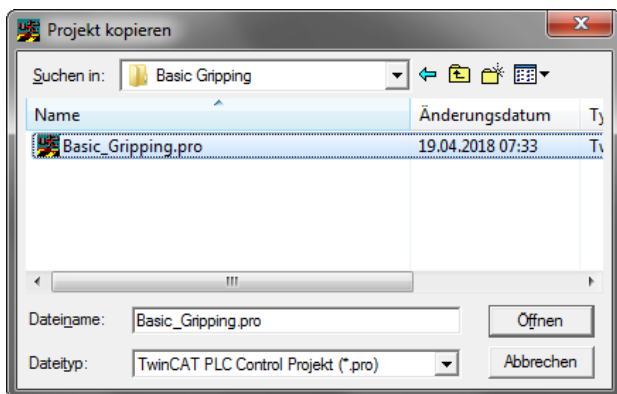
Im Parameterbaum können nun die Ein- bzw. Ausgänge des Greifer angezeigt und nach dem Einbinden einer SPS Konfiguration mit den Variablen des Programms verknüpft werden.

### 3 PLC Control

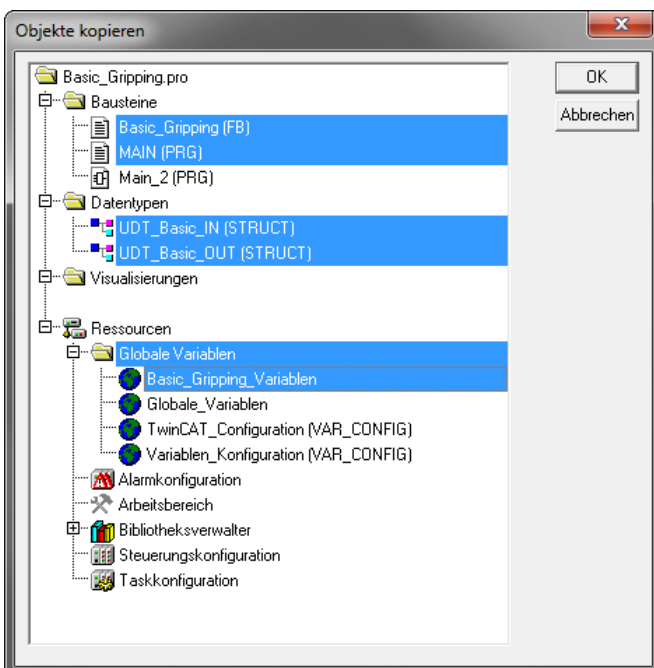


Um die Ansteuerung eines Greifers zu erleichtern, haben wir ein Beispielprojekt erstellt. In diesem Beispielprojekt wird gezeigt, wie ein Greifer angesteuert werden kann und wie die einzelnen Signale verarbeitet werden müssen.

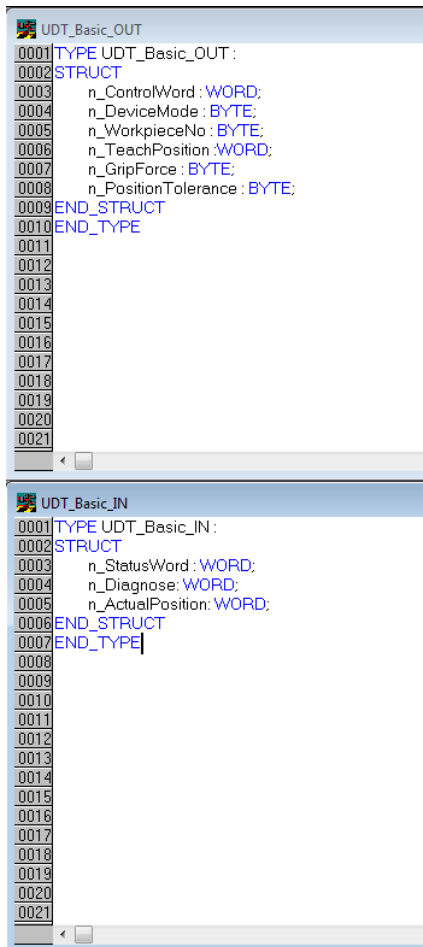
Um die gewünschten Bausteine in Ihrem Anwenderprogramm verwenden zu können, müssen Sie diese aus unserem Beispielprojekt kopieren. Hierfür müssen Sie in Ihrem Projekt unter dem Menü „Projekt“ den Befehl „Kopieren ...“ auswählen.



Im sich öffnenden Fenster müssen Sie nun unser Beispielprojekt auswählen.

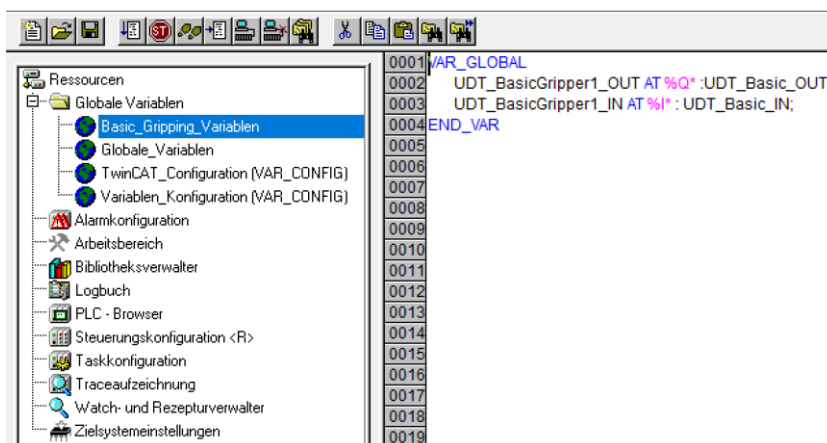


In dem Fenster „Objekte kopieren“ können die gewünschten Komponenten des Projektes kopiert werden. Der Funktionsbaustein „Basic\_Gripping“ beinhaltet den Programmablauf um den Greifer richtig anzusteuern. Damit der Baustein richtig arbeiten kann, werden die Datentypen „UDT\_Basic\_IN“ und „UDT\_Basic\_OUT“ und die Global Variablen Datei „Basic\_Gripping\_Variablen“ benötigt. Diese können Sie nun alle markieren und mit dem Button „OK“ in Ihr bestehendes Programm einfügen. Falls Sie ein neues, leeres Projekt erstellt haben, können Sie auch den Baustein „MAIN“ in Ihr Projekt kopieren. In diesem Baustein befindet sich der Programmaufruf für den Baustein „ZG\_FB\_Basic\_Gripping“.



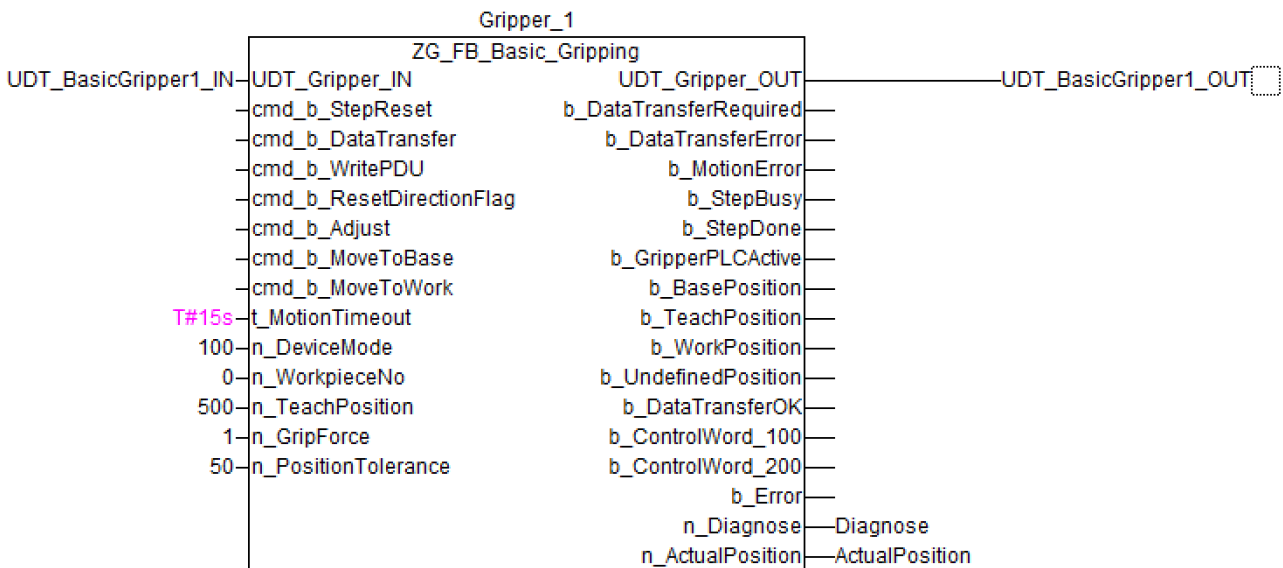
Die Definitionen der Datentypen „UDT\_Basic\_OUT“ und „UDT\_Basic\_Gripping\_IN“ sind ein genaues Abbild der Input- bzw. der Output-Variablen des Greifers, welchen Sie ansteuern möchten. Die Reihenfolge und die Datengröße müssen identisch zu der Vorgabe aus der Beschreibung des Greifers sein.

Falls sich diese ändern sollte, wären die Datenbereiche nichtmehr synchron und der Greifer könnte die Daten nicht richtig verarbeiten. Eine ordnungsgemäße Ansteuerung des Greifers wäre somit nicht mehr möglich.



In der angefügten Variablentabelle wird der Variable „UDT\_BasicGripper1\_OUT“ der Datentyp „UDT\_Basic\_OUT“ und der Variable „UDT\_BasicGripper1\_IN“ der Datentyp „UDT\_Basic\_IN“ zugewiesen. Diese beiden Variablen müssen später an den Baustein angefügt werden.

Für die einfachere Lesbarkeit des Bausteins wurde dieser in der Programmiersprache FUP eingefügt.



Variablen, welche mit „b\_“ deklariert sind, stellen binäre Signale dar.

Variablen, welche mit „cmd\_“ deklariert sind, stellen Befehlseingänge dar. Diese können z.B. mit einem Taster gesteuert werden.

Variablen, welche mit „n\_“ deklariert sind, sind Eingangs- bzw. Ausgangsbytes bzw. -wörter. Diese werden für die Übertragung der einzelnen Positionen und Funktionen benötigt.

## 4 Verwendung des Funktionsbausteins

An dem unter Punkt 3 eingefügten Funktionsbaustein sind nun mehrere Ein – bzw. Ausgänge zu beschalten. Der am Baustein befindliche Eingang „UDT\_Gripper\_IN“ muss nun mit der zugehörigen Variable aus Punkt 3 verbunden werden. Das gleiche muss auch mit der Ausgangsvariable „UDT\_Gripper\_OUT“ durchgeführt werden. Nun kann der Baustein die einzelnen Zustände und Positionen des Greifer auslesen und diese im Baustein verarbeiten. Des Weiteren kann durch die Ausgangsbeschaltung der Greifer parametrisiert werden.

Damit der Greifer gefahren werden kann, müssen diesem die verschiedenen Positionsdaten und Optionen (Fahrprofile) übertragen werden. Als Standardwerte können die in folgender Tabelle aufgeführten Werte verwendet werden. Diese sind beispielhaft und können projektspezifisch variieren. Sie können diese Parameter wie in diesem Beispiel als Konstanten am Baustein eintragen oder auch Variablen in entsprechender Länge verwenden, damit die Beschaltung flexibel ist. Bei Nichtbeschaltung sind die Variablen mit den Standardwerten vorinitialisiert.

n_DeviceMode	100 (1 bei GEP/GED5000IL)
n_WorkpieceNo	0
n_TeachPosition	500
n_GripForce	1
n_PositionTolerance	50



Die Variable `n_DeviceMode` entspricht dem Fahrprofil des Greifers. Diese Fahrprofile können der Montageanleitung des Greifers entnommen werden. In diesem Beispiel wurde der DeviceMode 100 (bei GEP2000IL oder GPP5000IL) bzw. 1 (bei GEP/GED5000IL) gewählt, welcher dem Fahrprofil „Universalbetrieb“ entspricht. Dieser kann als Standardwert verwendet werden.

Der fertige Baustein sollte nun obiger Abbildung entsprechen. Zu aller Letzt müssen die Einstellungen auf die PLC übertragen werden. Dazu gehen Sie die für Beckhoff notwendigen Schritte durch:

- Übersetzen
- Einloggen

## 5 Funktionen des Funktionsbausteins

Abhängig von der Eingangsbeschaltung des Funktionsbausteins werden die entsprechenden Funktionen ausgeführt. Weitere Informationen können Sie auch im Bausteinkopf finden.

### 5.1 Schrittkette zurücksetzen „cmd\_b\_StepReset“ (BOOL)

Die Eingangsvariable „cmd\_b\_StepReset“ setzt die Schrittkette innerhalb dieses Bausteins zurück. Dies geschieht unabhängig davon, in welchem Schritt sich der Baustein gerade befindet. Wenn der Baustein den Fehler „b\_DataTransferError“ oder „b\_MotionError“ ausgibt, kann er nur durch diesen Eingang quittiert werden.

### 5.2 Datenübertragung mit Handshake „cmd\_b\_DataTransfer“ (BOOL)

Nach jeder Änderung eines Prozessparameters (ausgenommen „ControlWord“) oder bei einem Kaltstart des Greifers müssen die Parameter mit einer Datenübertragung übernommen werden. Wenn die Ausgangsvariable „b\_DataTransferRequired“ „TRUE“ ist, arbeitet der Greifer noch nicht mit den aktuell eingestellten Parametern. In diesem Fall muss der Eingang „cmd\_b\_DataTransfer“ getriggert werden, damit die Prozessparameter übertragen werden. Die Variable „b\_DataTransferRequired“ wechselt anschließend auf „FALSE“. Dabei wird das „ControlWord“ auf Wert 1 gesetzt und auf das Bit 12 des „StatusWord“ gewartet. Das Bit 12 wird „TRUE“, sobald die Datenübertragung abgeschlossen ist. Anschließend wird das „ControlWord“ wieder auf 0 gesetzt und gewartet, bis das Bit 12 „FALSE“ wird. Diese Prozedur ist ein Handshake und sollte für die fehlerfreie Datenübertragung angewandt werden.

### 5.3 Werkstückrezeptur speichern „cmd\_b\_WritePDU“ (BOOL)

Wenn dieser Eingang auf „TRUE“ gesetzt wird, werden die aktuell gesetzten Prozessparameter am Bausteineingang in die aktuell eingestellte „WorkpieceNo“ gespeichert. Dabei wird das „ControlWord“ auf Wert 2 gesetzt und auf das Bit 12 des „StatusWord“ gewartet. Diese Prozedur kann bis zu 30 Sekunden dauern. Die Parameter werden stromausfallsicher im Greifer abgespeichert und können über die Angabe der „WorkpieceNo“ wieder angewählt werden. Es können bis zu 32 Rezepturen im Greifer abgespeichert werden.

### 5.4 Richtungsmerker zurücksetzen „cmd\_b\_ResetDirectionFlag“ (BOOL)

Wenn ein Greifer z.B. in Richtung „WorkPosition“ gefahren wird, wird im Greifer das Bit 14 des „StatusWord“ gesetzt. Dieses Signal bleibt bis zu einer Bewegung in die andere Richtung oder einem Kaltstart des Greifers bestehen. Wenn ein Greifer mehrmals nacheinander in die gleiche Richtung gefahren werden soll, dann muss dieses Bit zuerst zurückgesetzt werden. Dies kann durch den Eingang „cmd\_b\_ResetDirectionFlag“ durchgeführt werden. Dabei wird das „ControlWord“ auf den Wert 4 gesetzt und gewartet bis Bit 13 und Bit 14 des „StatusWord“ auf „FALSE“ wechseln. Danach kann eine erneute Bewegung in die gleiche Richtung erfolgen. Ab der Bausteinversion V1.21 wird diese Prozedur falls notwendig vor der Bewegung des Greifers automatisch durchgeführt.

#### 5.5 Fahren auf BasePosition „cmd\_b\_MoveToBase“ (BOOL)

Beim Setzen dieses Eingangssignals auf „TRUE“ bewegen sich die Greiferbacken mit dem eingestellten Fahrprofil und mit der eingestellten Greifkraft auf die „BasePosition“. Dabei wird das „ControlWord“ auf den Wert 256 gesetzt.

#### 5.6 Fahren auf WorkPosition „cmd\_b\_MoveToWork“ (BOOL)

Beim Setzen dieses Eingangssignals auf „TRUE“ bewegen sich die Greiferbacken mit dem eingestellten Fahrprofil und mit der eingestellten Greifkraft auf die „WorkPosition“. Dabei wird das „ControlWord“ auf den Wert 512 gesetzt.

#### 5.7 Begrenzung der Bewegungszeit „t\_MotionTimeout“ (TIME) und „b\_MotionError“ (BOOL)

Falls der Greifer eine Bewegung nicht ausführen kann oder das gewünschte Ziel nicht erreicht, würde die Schrittkette stehenbleiben und der Baustein wäre für weitere Befehle blockiert. Damit dies nicht passiert, kann die Zeit „t\_MotionTimeout“ am Baustein definiert werden, welche der Greifer maximal für seine Bewegung bis zur Position benötigen darf. Dies ist abhängig von den Eingangsparametern des Greifers und muss projektspezifisch angepasst werden. Wenn der Greifer innerhalb der eingestellten Zeit sein Ziel nicht erreicht, springt die Schrittkette in einen Fehlerschritt. Der Ausgang „b\_MotionError“ wird auf „TRUE“ gesetzt und kann nur durch Setzen des Eingangs „cmd\_b\_StepReset“ wieder zurückgesetzt werden.

#### 5.8 Datenübertragung wird benötigt „b\_DataTransferRequired“ (BOOL)

Die Variable "b\_DataTransferRequired" ist aktiv, wenn mindestens eine der Output-Variablen, die zum Greifer geschickt werden, geändert wurde. Solange diese Variable aktiv ist, hat der Greifer die geänderten Werte noch nicht übernommen. Zur Datenübernahme muss die Eingangsvariable "cmd\_b\_DataTransfer" getriggert werden. Die Variable "b\_DataTransferRequired" wechselt dann auf „FALSE“ und der Greifer verwendet die aktuell eingestellten Parameter.

#### 5.9 Fehler in der Datenübertragung „b\_DataTransferError“ (BOOL)

Der Ausgang „b\_DataTransferError“ wird auf „TRUE“ gesetzt, wenn die Datenübertragung („ControlWord“ = 1) nicht erfolgreich durchgeführt werden konnte und das Feedback des Greifers nicht innerhalb einer Sekunde gesendet wurde. Dies kann mehrere Gründe zur Ursache haben. Einen Fehlercode kann über den Ausgang „n\_Diagnose“ entnommen werden. Die einzelnen Fehlercodes sind in der Montageanleitung näher beschrieben. Durch Setzen des Eingangs „cmd\_b\_StepReset“ kann dieser Fehler quittiert werden.

#### 5.10 Befehl wird ausgeführt „b\_StepBusy“ (BOOL)

Wenn der Baustein einen Befehl bearbeitet und in einem Schritt steht, ist dieser Ausgang aktiv und signalisiert, dass er für weitere Befehle blockiert ist.

#### 5.11 Bereit für Befehle „b\_StepDone“ (BOOL)

Wenn der Baustein im Initialschritt steht und bereit für Befehle ist, ist dieser Ausgang auf „TRUE“.

#### 5.12 Bit 6 des StatusWord „b\_GripperPLCActive“ (BOOL)

Dieses Signal gibt die Betriebsbereitschaft der Steuerung im Greifer wieder. Bei einem Kaltstart oder Neustart nach Spannungsausfall, kann die Steuerung erst dann wieder Daten empfangen, wenn dieses Signal „TRUE“ ist.

#### 5.13 Bit 8 des StatusWord „b\_BasePosition“ (BOOL)

Sobald der Greifer seine eingestellte „BasePosition“ erreicht hat, wird dieses Signal aktiviert. Die Größe des Bereichs wird durch die „PositionTolerance“ definiert.

#### 5.14 Bit 9 des StatusWord „b\_TeachPosition“ (BOOL)

Sobald der Greifer seine eingestellte „TeachPosition“ erreicht hat, wird dieses Signal aktiviert. Die Größe des Bereichs wird durch die „PositionTolerance“ definiert.

#### 5.15 Bit 10 des StatusWord „b\_WorkPosition“ (BOOL)

Sobald der Greifer seine eingestellte „WorkPosition“ erreicht hat, wird dieses Signal aktiviert. Die Größe des Bereichs wird durch die „PositionTolerance“ definiert.

#### 5.16 Bit 11 des StatusWord „b\_UndefinedPosition“ (BOOL)

Wenn der Greifer stillsteht und weder auf „BasePosition“, „TeachPosition“ oder „WorkPosition“ steht, ist dieses Signal „TRUE“.

#### 5.17 Bit 12 des StatusWord „b\_DataTransferOK“ (BOOL)

Mit diesem Bit gibt der Greifer die Rückmeldung, dass eine Datenübertragung („ControlWord“ = 1) erfolgreich durchgeführt wurde. Daher wird es bei einer Handshakeprozedur verwendet.

#### 5.18 Bit 13 des StatusWord „b\_ControlWord\_100“ (BOOL)

Dieser Richtungsmerker wird „TRUE“, wenn der Greifer einen „MoveToBase“-Befehl erhalten hat. Der Greifer kann in diesem Zustand keinen weiteren „MoveToBase“-Befehl ausführen. Der Merker wird wieder auf „FALSE“ gesetzt, wenn der Greifer einen „MoveToWork“-Befehl erhält oder per „cmd\_b\_ResetDirectionFlag“ (s. 5.4) manuell ein Reset durchgeführt wird.

#### 5.19 Bit 14 des StatusWord „b\_ControlWord\_200“ (BOOL)

Dieser Richtungsmerker wird „TRUE“, wenn der Greifer einen „MoveToWork“-Befehl erhalten hat. Der Greifer kann in diesem Zustand keinen weiteren „MoveToWork“-Befehl ausführen. Der Merker wird wieder auf „FALSE“ gesetzt, wenn der Greifer einen „MoveToBase“-Befehl erhält oder per „cmd\_b\_ResetDirectionFlag“ (s. 5.4) manuell ein Reset durchgeführt wird.

#### 5.20 Bit 15 des StatusWord „b\_Error“ (BOOL) und „n\_Diagnose“ (UINT)

Wenn der Diagnosewert des Greifers nicht 0 ist, wird dieses Bit gesetzt. Der Fehlercode wird in dem Datenwort „n\_Diagnose“ ausgegeben. Die Beschreibungen zu den Fehlercodes können aus der Bedienungsanleitung entnommen werden.

#### 5.21 n\_ActualPosition (UINT)

In diesem Datenwort wird die aktuelle Position der Greiferbacken ausgegeben.