

Schnittstellenbeschreibung

Feldbus Schnittstelle, Allgemeine Beschreibung

Für Singlechannelsysteme: SMX100..400

Version 1.01

DE_SC_Schnittstellenbeschreibung_FeldbusAllgemein

Stand: 2021-03-11



SMX-Serie

Alfing Montagetechnik GmbH

Auguste-Kessler-Straße 20
73433 Aalen
Deutschland

Telefon: +49 (0) 7361 / 501 - 2701
Telefax: +49 (0) 7361 / 501 - 2709
E-Mail: info@amt.alfing.de
Web: amt.alfing.de

Bevollmächtigter zur Zusammenstellung der Technischen Unterlagen:

Wolfgang Mangold

Gruppenleiter
Softwareentwicklung Schraubtechnik (ME)

Revision index

Rev	Description of the change	Date	Creator	Released
V1.00	Create the document	08.10.2019	W. Mangold	
V1.01	Values of the examples adjusted	11.03.2021	W. Mangold	

AMT Alfing Montagetechnik GmbH • D-73433 Aalen

Alle Rechte, insbesondere das Recht der Vervielfältigung und Verbreitung sowie der Übersetzung, vorbehalten. Kein Teil dieser Anleitung darf in irgendeiner Form (Druck, Fotokopie, Mikrofilm oder ein anderes Verfahren) ohne schriftliche Genehmigung der Fa. AMT Alfing Montagetechnik GmbH reproduziert oder unter Verwendung elektronischer Systeme gespeichert, verarbeitet, vervielfältigt oder verbreitet werden.

Änderungen vorbehalten.

Inhalt

1.	Allgemein	4
1.1.	Einleitung	4
1.2.	Softwarevoraussetzung: Versions-Information.....	4
1.3.	Hardwarevoraussetzung: ProfiNet-Modul oder ProfiBus-Modul	4
1.4.	Aktivieren der ProfiNet/ProfiBus Kommunikation	4
1.5.	Konfiguration der Feldbus Karte (über Sycon.Net)	5
1.6.	Modulkonfiguration	5
1.7.	Konfiguration der Signale (im I/O-Mapper).....	5
2.	Schnittstellenbeschreibung.....	6
2.1.	Schnittstellenprotokoll.....	6
2.2.	Eingänge Schraubcontroller / Ausgänge SPS (TOOL-> SPS).....	6
2.3.	Ausgänge Schraubcontroller / Eingänge SPS (SPS->TOOL).....	7
2.4.	Schraubdaten (Ausgänge Schraubcontroller) (SPS->TOOL)	8
2.5.	Schraubdatenübertragung	9
2.6.	Internes Enable Signal	10
2.7.	Hinweise zu den Signalen	10
2.8.	Rücksetzen der Ausgänge	10
3.	Signallaufpläne	11
3.1.	Ablauf: OK Verschraubung	11
3.2.	Ablauf: NOK Verschraubung	12
3.3.	Ablauf: Signal Reset Outputs	13
3.4.	Ablauf: Signal StartReady	14
3.5.	Programmübernahme mit dem START-Signal.....	15
3.6.	Programmübernahme mit dem ENABLE-Signal	15
3.7.	Hinweise zur Programmübernahme	15
3.8.	Ablauf: Abbruch einer Verschraubung über Enable	16
3.8.1.	Hinweis zum Enable:	16
3.9.	Hardwarekonfiguration (über Sycon.Net)	17
3.10.	Modulkonfiguration über Sycon.Net	18
3.11.	GSD	19
4.	Mapping	20
4.1.	Mapping-Datei (mapping.map)	20
4.2.	Device-Datei (devices.map)	20
4.3.	Signals-Datei (signal.map)	21
5.	Signalbeschreibung ProfiNet	23
5.1.1.	Weitere Signale	30
5.2.	Signalbeschreibung ProfiBus	32

1. Allgemein

1.1. Einleitung

Im Folgenden wird die *Feldbus* Signalschnittstelle auf den *SMX100-400* Systemen beschrieben.

Beschrieben wird das Protokoll „Gesteuertes Schrauben mit binärer Schraubdatenübertragung und Redundanz“. Das Standard-Schnittstellenprotokoll wurde hier um die Redundanzwerte für Drehmoment und Winkel erweitert. Die Ergebnisdaten werden binär übertragen (bisher ASCII).

Die *Feldbus*-Signalschnittstelle ist komplett konfigurierbar und kann über sogenannte Mapping-Dateien auf Kundensystem angepasst werden.

Die meisten Signale der *Feldbus*-Signalschnittstelle können bei *ProfiNet* und *ProfiBus* verwendet werden.

1.2. Softwarevoraussetzung: Versions-Information

Die hier beschriebene Signalschnittstelle erfordert folgendes Softwarerelease oder höher:

Release V1.5.3 – A01

1.3. Hardwarevoraussetzung: ProfiNet-Modul oder ProfiBus-Modul

Für die *ProfiNet/ProfiBus*-Kommunikation muss die *SMX100* Hardware mit einem *Feldbus*-Modul ausgestattet sein. In dem vorgesehenen *Feldbus*-Slot auf der *SMX100* muss ein entsprechende *NetJACK* Modul der Firma *Hilscher* gesteckt sein.

1.4. Aktivieren der ProfiNet/ProfiBus Kommunikation

Um die *Feldbus*-Schnittstelle einsetzen zu können, muss diese im Webinterface unter *Konfiguration/Kommunikation/Feldbus* aktiviert werden.

Über den Parameter *Programmübernahme mit* kann festgelegt werden, mit welchem Signal die Programmübernahme erfolgt.

The screenshot shows a web interface for configuring the FELDBUS. It is divided into three main sections: MAPPING, PROFINET, and PROFIBUS.

- MAPPING:**
 - Aktuelles Mapping: Amt-FB-BinRedundancyResults
 - Aktuelles Mapping: Anzeigen (button)
 - Vorlage: Fieldbus | Amt-FB-BinRedund (dropdown menu)
- PROFINET:**
 - Profinet: OFF (toggle switch)
 - Programmübernahme mit: Freigabe (dropdown menu)
- PROFIBUS:**
 - Ansteuerung über Profibus aktivieren: OFF (toggle switch)

1.5. Konfiguration der Feldbus Karte (über Sycon.Net)

Als *ProfiNet/ProfiBus*-Modul wird eine *NetJACK-ProfiNet/ProfiBus* Karte von Hilscher eingesetzt. Die Konfiguration der I/O Module sowie ein Firmwareupdate, erfolgt über die Softwareapplikation *Sycon.NET* der Firma *Hilscher*. Die I/O-Konfiguration kann auch direkt über den SPS Master vorgenommen werden. Die Konfiguration der I/O-Module über *Sycon.NET* ist einem späteren Kapitel erläutert.

1.6. Modulkonfiguration

Für das beschriebene Schnittstellenprotokoll, muss die Karte mit folgenden I/O-Modulen konfiguriert werden:

Beschreibung aus Controllersicht	Anzahl Bytes
Anzahl Eingangsmodule	2 Byte
Anzahl Ausgangsmodule	24 Byte

Hinweis

- Byte 0 und Byte 1 wird für den Signalaustausch verwendet.
- Ab Byte 2 erfolgt bei den Ausgängen die Übertragung der Schraubdaten.
- Hier sind die I/O Module aus **Sicht des Controllers** angegeben

1.7. Konfiguration der Signale (im I/O-Mapper)

In der SMX-Steuerung ist ein sogenannter I/O-Mapper integriert. Dieser bietet die Möglichkeit die vordefinierten Feldbus-Signale auf entsprechende I/O-Bytes/Bits festzulegen. Die Feldbus-Signale können über Mapping-Dateien konfiguriert werden. Momentan muss die Konfiguration von der Firma AMT vorgenommen werden. Später können die Signale über die Softwareapplikation *I/O-Mapper-Configurator* in der Mapping-Datei verändert werden.

Die Signalnamen (Signalbezeichner) innerhalb der Datei sind in Englisch benannt und mit dem entsprechenden Signalnamen in der Mapping-Datei anzugeben.

Die Signalnamen beginnen mit der Kennung „*IOM_*“

Die konfigurierten Signale werden in der Datei „mapping.map“ definiert.

2. Schnittstellenbeschreibung

2.1. Schnittstellenprotokoll

Das Schnittstellenprotokoll „Gesteuertes Schrauben mit binärer Schraubdatenübertragung und Redundanzwerten“ muss in den Templates der Weboberfläche übertragen und ausgewählt werden.

2.2. Eingänge Schraubcontroller / Ausgänge SPS (TOOL-> SPS)

Byte	Bit	Signal	Signalname im I/O Mapper	Version
0	0	Freigabe (Spindel)	IOM_Enable	V1.5.3
	1	n.c.	-	
	2	Start (Taster)	IOM_Start	V1.5.0
	3	n.c. (not connected)	-	
	4	n.c.	-	
	5	n.c.	-	
	6	Reset Ausgänge	IOM_Reset_Outputs	V1.5.0
	7	Reset Servo Alarm	IOM_Reset_Servo_Alarm	V1.5.0
1	0	Programmanwahl binär 1	IOM_Program_Selection_Bin@[6](.1)	V1.5.0
	1	Programmanwahl binär 2	IOM_Program_Selection_Bin(.2)	V1.5.0
	2	Programmanwahl binär 4	IOM_Program_Selection_Bin(.3)	V1.5.0
	3	Programmanwahl binär 8	IOM_Program_Selection_Bin(.4)	V1.5.0
	4	Programmanwahl binär 16	IOM_Program_Selection_Bin(.5)	V1.5.0
	5	Programmanwahl binär 32	IOM_Program_Selection_Bin(.6)	V1.5.0
	6	n.c		
	7	n.c		

2.3. Ausgänge Schraubcontroller / Eingänge SPS (SPS->TOOL)

Byte	Bit	Typ	Signal	Signalname im I/O Mapper	Version
0	0	Bit	Betriebsbereit	IOM_Ready	V1.5.0
	1	Bit	Startbereit (Schraubbereit)	IOM_Start_Ready	V1.5.0
	2	Bit	Verschraubung beendet	IOM_Tightening_Finished	V1.5.0
	3	Bit	Verschraubung OK	IOM_SingleOK	V1.5.0
	4	Bit	Verschraubung NOK	IOM_SingleNOK	V1.5.0
	5	Bit	In Cycle	IOM_InCycle	V1.5.3
	6	Bit	TighteningRunning	IOM_TighteningRunning	V1.5.3
	7	Bit	Kein Servo Alarm	IOM_No_Servo_Alarm	V1.5.0
1	0	Bit	Echo Programmanwahl binär 1	IOM_Echo_Program_Sel_Bin@[6]	V1.5.0
	1	Bit	Echo Programmanwahl binär 2		V1.5.0
	2	Bit	Echo Programmanwahl binär 4		V1.5.0
	3	Bit	Echo Programmanwahl binär 8		V1.5.0
	4	Bit	Echo Programmanwahl binär 16		V1.5.0
	5	Bit	Echo Programmanwahl binär 32		V1.5.0
	6	Bit	Start Taster/Signal gedrückt	IOM_StartTriggerPressed	V1.5.3
	7	Bit	Linkslauf angewählt	IOM_Reverse_Switch_selected	V1.5.3

2.4. Schraubdaten (Ausgänge Schraubcontroller) (SPS->TOOL)

Ab Byte2 bei den Ausgängen werden die Schraubdaten in folgendem Format übertragen:

Byte	Bit	Typ	Signal	Signalname im I/O Mapper	Anz	Bsp.	Version
2	0..7	Byte	Program number, enabled	IOM_ProgramNumber_Enabled	1/1	01	V1.5.3
3	0..7	Byte	Program number, for result	IOM_ProgramNumber_Result	1/1	01	V1.5.3
4	0..7	Word LB	TorqueActNmX10	IOM_TorqueActNm_X10@[B2]	1/1	0x7D	V1.5.3
5	0..7	Word HB	TorqueActNmX10		2/1	0x00	V1.5.3
6	0..7	Word LB	TorqueMinNmX10	IOM_TorqueMinNm_X10@[B2]	1/1	0x70	V1.5.3
7	0..7	Word HB	TorqueMinNmX10		2/1	0x00	V1.5.3
8	0..7	Word LB	TorqueMaxNmX10	IOM_TorqueMaxNm_X10@[B2]	1/1	0x85	V1.5.3
9	0..7	Word HB	TorqueMaxNmX10		2/1	0x00	V1.5.3
10	0..7	Word LB	AngleActDeg	IOM_AngleActDeg@[B2]	1/1	0x69	V1.5.3
11	0..7	Word HB	AngleActDeg		2/1	0x01	V1.5.3
12	0..7	Word LB	AngleMinDeg	IOM_AngleMinDeg@[B2]	1/1	0x56	V1.5.3
13	0..7	Word HB	AngleMinDeg		2/1	0x01	V1.5.3
14	0..7	Word LB	AngleMaxDeg	IOM_AngleMaxDeg@[B2]	1/1	0x7A	V1.5.3
15	0..7	Word HB	AngleMaxDeg		2/1	0x01	V1.5.3
16	0..7	Word LB	CurrentActAmpX10	IOM_CurrentActAmp_X10[B2]	1/1	0x30	V1.5.3
17	0..7	Word HB	CurrentActAmpX10		2/1	0x00	V1.5.3
18	0..7	Word LB	Torque2ActNmX10	IOM_Torque2ActNm_X10@[B2]	1/1	0x7E	V1.5.3
19	0..7	Word HB	Torque2ActNmX10		2/1	0x00	V1.5.3
20	0..7	Word LB	Angle2ActDeg	IOM_Angle2ActDeg@[B2]	1/1	0x69	V1.5.3
21	0..7	Word HB	Angle2ActDeg		2/1	0x01	V1.5.3
22	0..7	BYTE	Reserved=0				
23	0..7	BYTE	Reserved=0				

Hinweis:

- Die Schraubwerte Torque, Torque2 und Strom sind mit 10 multipliziert und dadurch mit einer Nachkommastelle angegeben.

2.5. Schraubdatenübertragung

Die Schraubwerte/Schraubergebnisse werden in einem fixen Format als ASCII – Zeichenkette übertragen.

Signal	Typ	Beschreibung
ProgramNumber, Enabled, Result	Ausgang	Programmnummer (Big Endian) Die Programmnummer wird mit einem Byte angegeben. Dadurch sind Werte zwischen 0..255 möglich Freigabe von Programm 0 ist nicht erlaubt.
Torque Nm_X10 Torque2 Nm_X10 (Drehmoment in Nm)	Ausgang	Das Drehmoment wird in Nm als Word mit Lowbyte und Highbyte angegeben. Die Werte sind mit 10 multipliziert. Dadurch besitzt jeder Werte eine Nachkommastelle. LB: 0x7D HB: 0x00 -> 125 -> 12,5 Nm
Angle Deg Angle2 Deg (Drehwinkel in Grad)	Ausgang	Der Drehwinkel wird in Grad als Word mit Lowbyte und Highbyte ausgegeben LB: 0x69 HB: 01 -> 361 Grad
Current Amp (Strom in Ampere)	Ausgang	Der Strom wird in Ampere als Word mit Lowbyte und Highbyte ausgegeben. Die Werte sind mit 10 multipliziert. Dadurch besitzt jeder Werte eine Nachkommastelle. LB: 0x5C HB: 0x00 -> 92 -> 9,2 Ampere

Hinweis

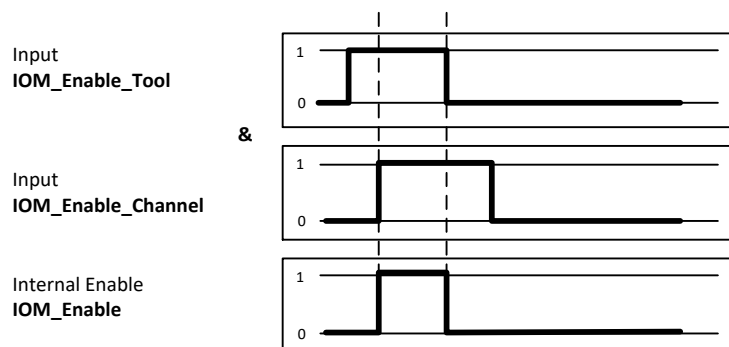
- Die Schraubwerte Torque und Current sind mit 10 multipliziert.
- Werte mit intern zwei Nachkommastellen, werden ab 0,05 aufgerundet, darunter abgerundet übertragen.

2.6. Internes Enable Signal

Zur Toolfreigabe können zwei unterschiedliche Übertragungen gewählt werden. Entweder wie oben konfiguriert nur ein *IOM_Enable* Signal oder wie hier dargestellt zwei Signale, welche intern verknüpft werden. Standardmäßig ist ein Signal *IOM_Enable* konfiguriert.

Die zwei Signale *IOM_EnableTool* und *IOM_EnableChannel* werden in der I/O-Mapping-Datei UND-verknüpft. Dadurch wird ein internes Signal *IOM_Enable* welches die Freigabe erzeugt. Zur Freigabe bzw. zum Schrauben müssen beide Signale anliegen.

Wenn nur ein Signal gewünscht ist, kann innerhalb der Mapping-Datei, auch nur das Signal *IOM_Enable* gemapped werden.



2.7. Hinweise zu den Signalen

- Die Programmübernahme erfolgt je nach Einstellung in der Konfiguration entweder mit dem Starttaster oder dem Enable Signal.
- Bei dem Signal *IOM_Echo_Program_Selection* wird nicht das verschraubte Programm, sondern das gespiegelte Programm ausgegeben.
- Über das Feld *ProgramNumberEnabled* wird das freigegebene Programm ausgegeben
- Vor einer erneuten Verschraubung sollte vom Master zum Rücksetzen der Schraubensignale ein Signal *IOM_Reset_Output* ausgelöst werden
- Ein Programmwechsel auf 0 bewirkt, dass *IOM_StartReady* auf 0 gesetzt wird.
- Gleichzeitiges Anlegen von Signalen wird in der Regel nicht akzeptiert. Signale werden immer sequentiell in der Reihenfolge der Mappinganordnung abgearbeitet.

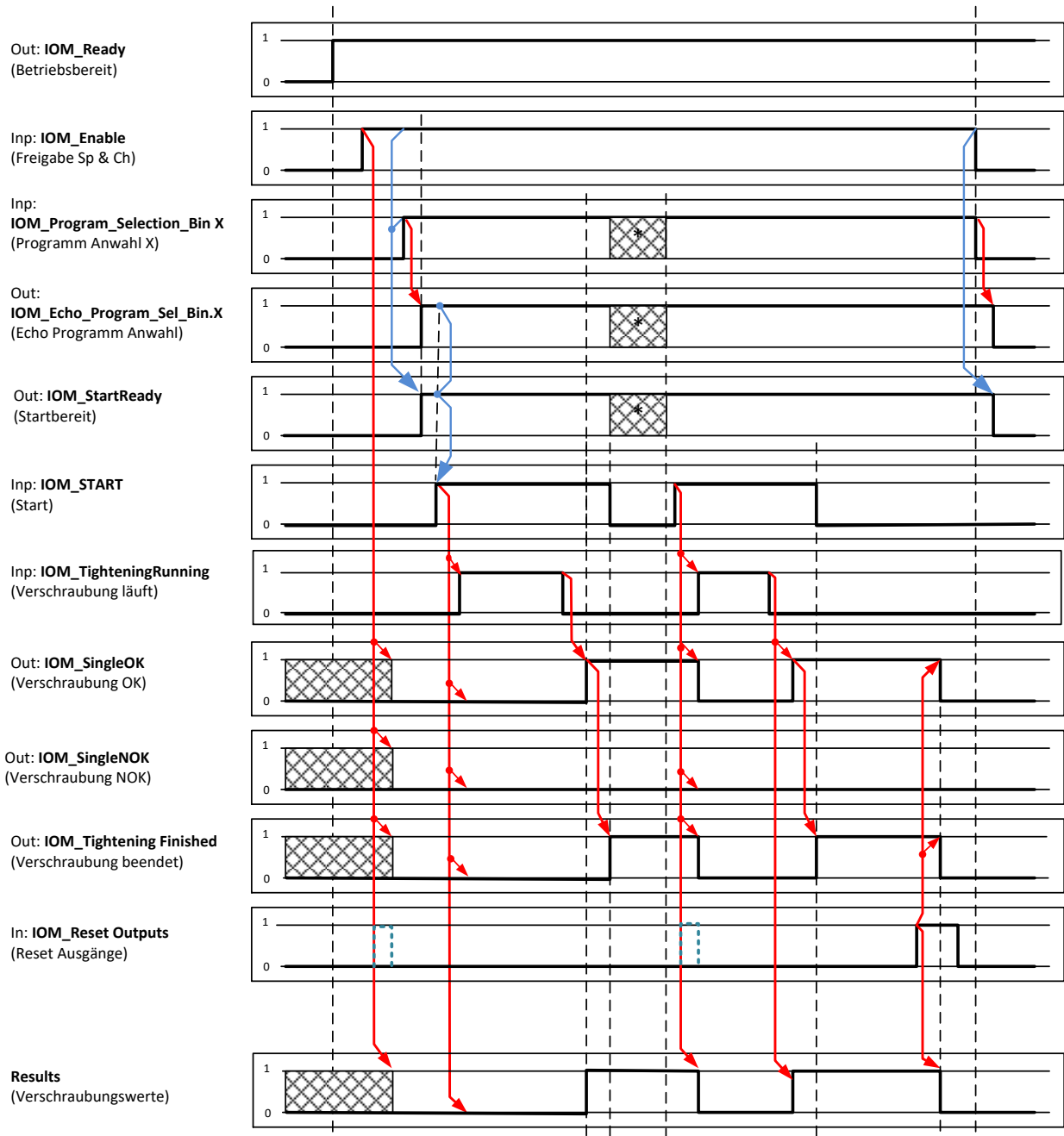
2.8. Rücksetzen der Ausgänge

Bei jedem *IOM_Enable* mit *IOM_Start* wird intern das Signal *IOM_Reset_Outputs* ausgelöst.

3. Signallaufpläne

3.1. Ablauf: OK Verschraubung

Vorraussetzung: Modus: „ProgramTakeOverWith = Start“

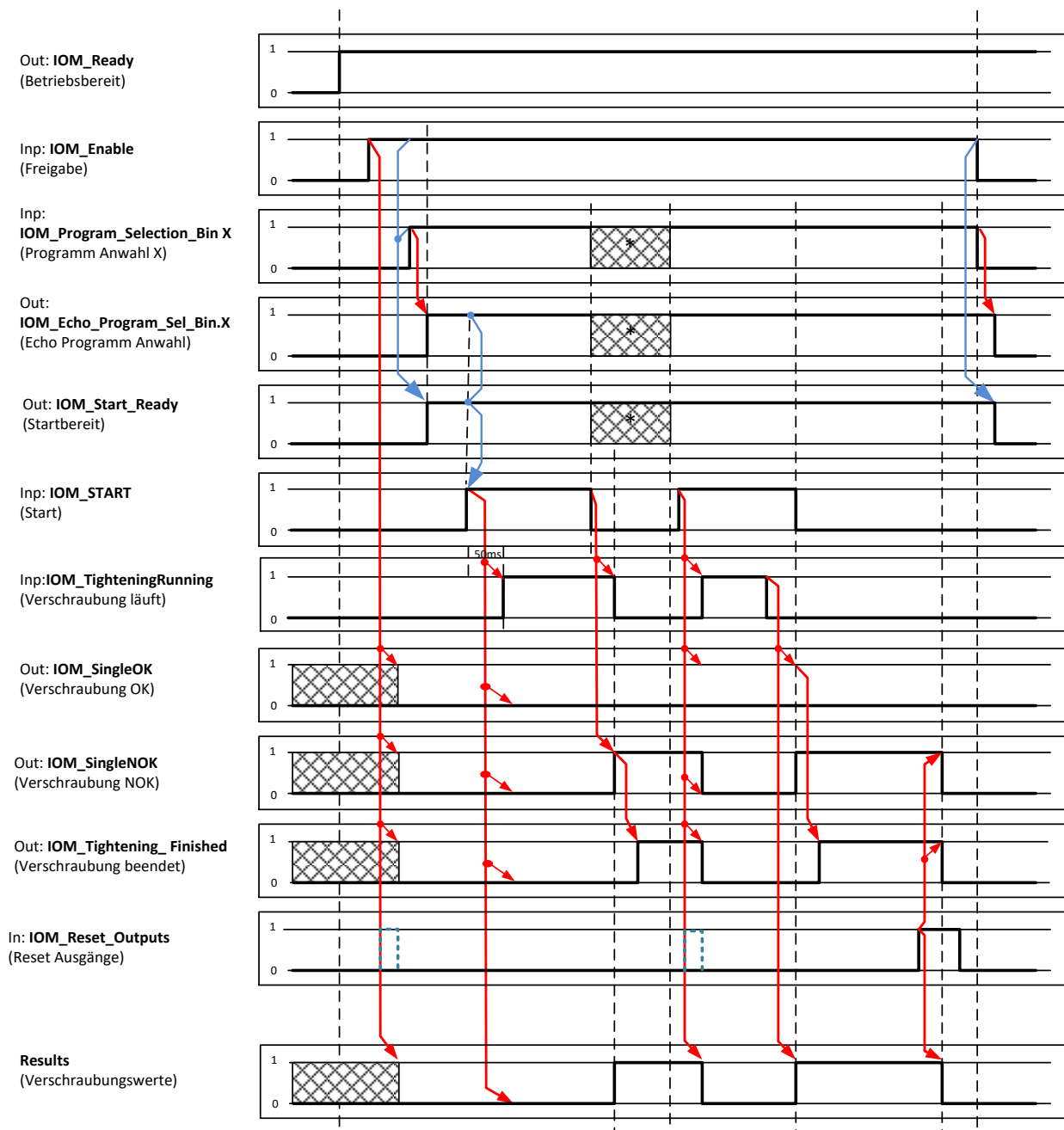


Hinweise

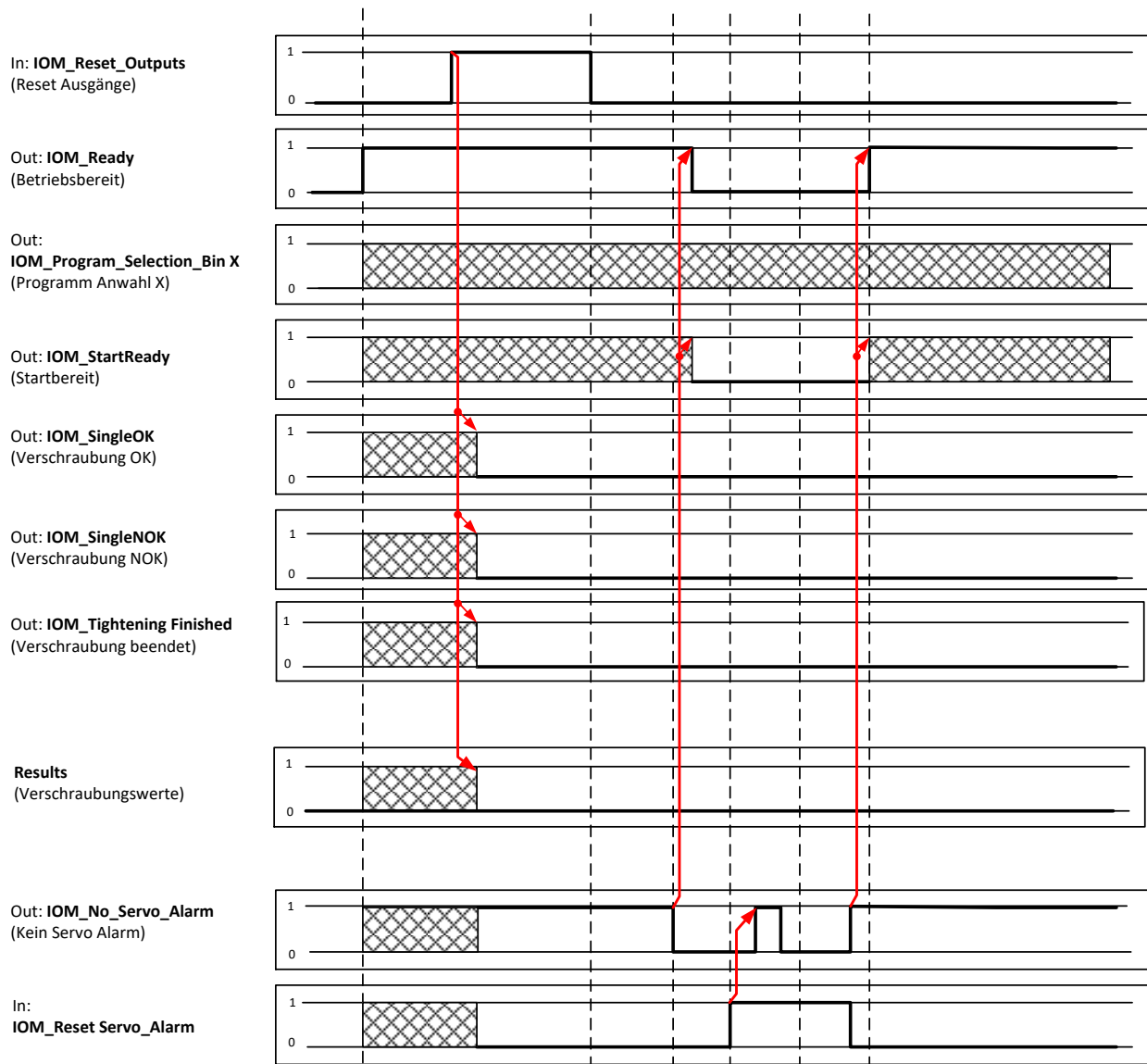
- Programmspiegelung muss beachtet werden. Das Programm muss vor dem Start gespiegelt anliegen
- Das Signal Enable setzt nicht in jedem Modus die Ausgangs-/Status-Signale zurück.
- Die Optione *ProgramTakeOver with Start* gilt momentan nur für *ProfiNet*, nicht für *Profibus*

3.2. Ablauf: NOK Verschraubung

Vorraussetzung: Modus: „ProgramTakeOverWith = Start“



3.3. Ablauf: Signal Reset Outputs

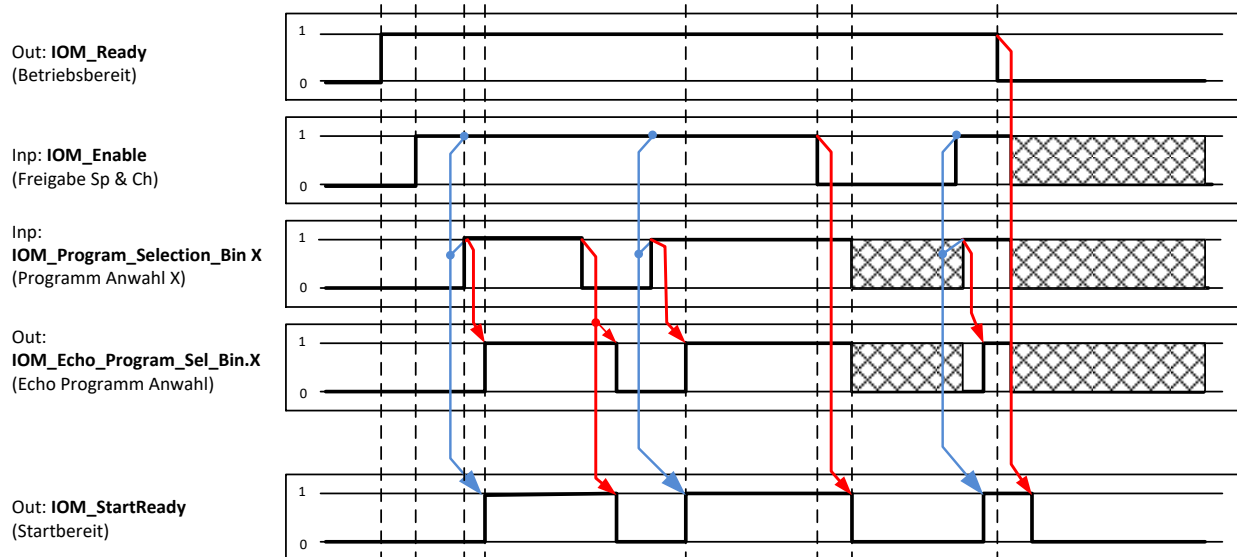


Hinweise:

- Ein anliegender Systemfehler wird über *IOM_Reset_Outputs* nicht zurückgesetzt
- *IOM_No_Servo_Alarm* wird durch das Signal *IOM_Reset_Servo_Alarm* gelöscht jedoch bei einem anliegenden Fehler sofort wieder gesetzt.

3.4. Ablauf: Signal StartReady

Vorraussetzung: „ProgramTakeOverWith = Start“

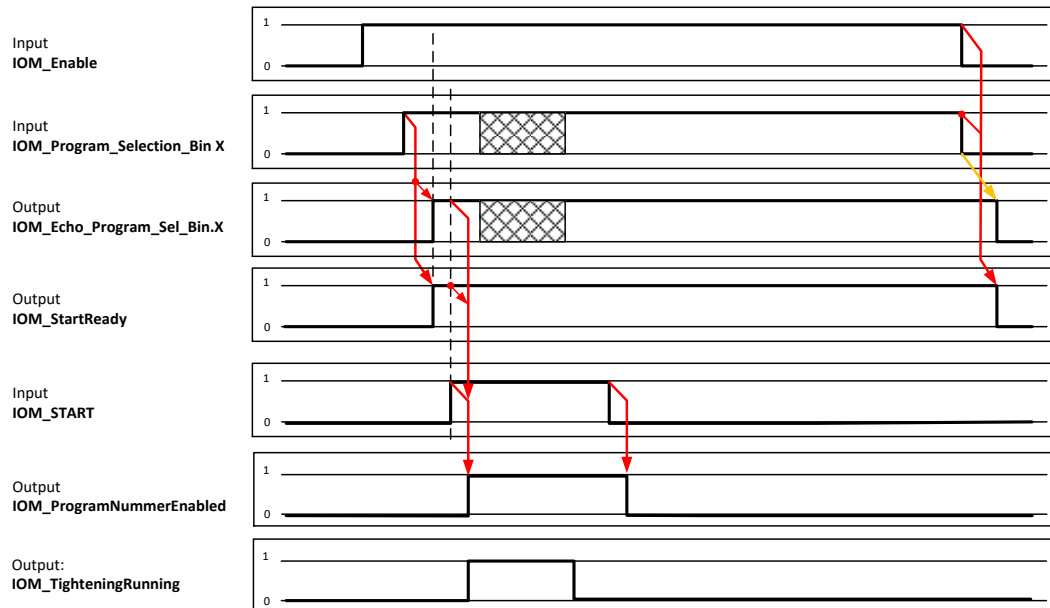


Hinweise

- Die Programmnummer wird auf PG!=0 geprüft
- Für das Signal *IOM_StartReady* (Startbereit) ist es notwendig, dass
 - a) kein Systemfehler anliegt (Ready = 1)
 - b) das Signal *IOM_Enable* anliegt
 - c) die Programmnummer ungleich Null ist (PG!=0).

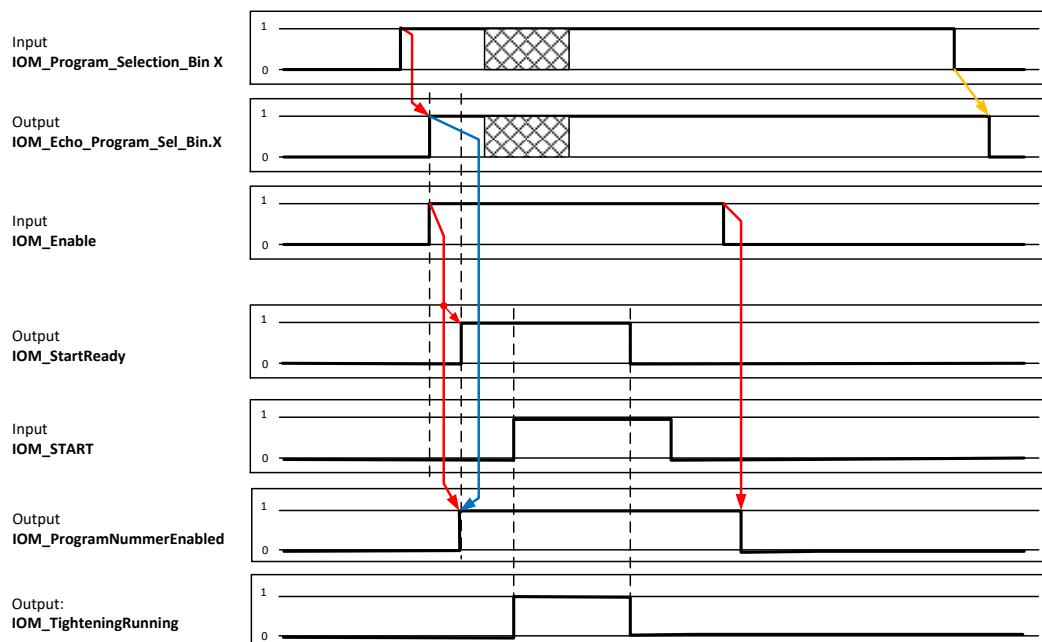
3.5. Programmübernahme mit dem START-Signal

Vorraussetzung: Modus: „ProgramTakeOverWith = Start“



3.6. Programmübernahme mit dem ENABLE-Signal

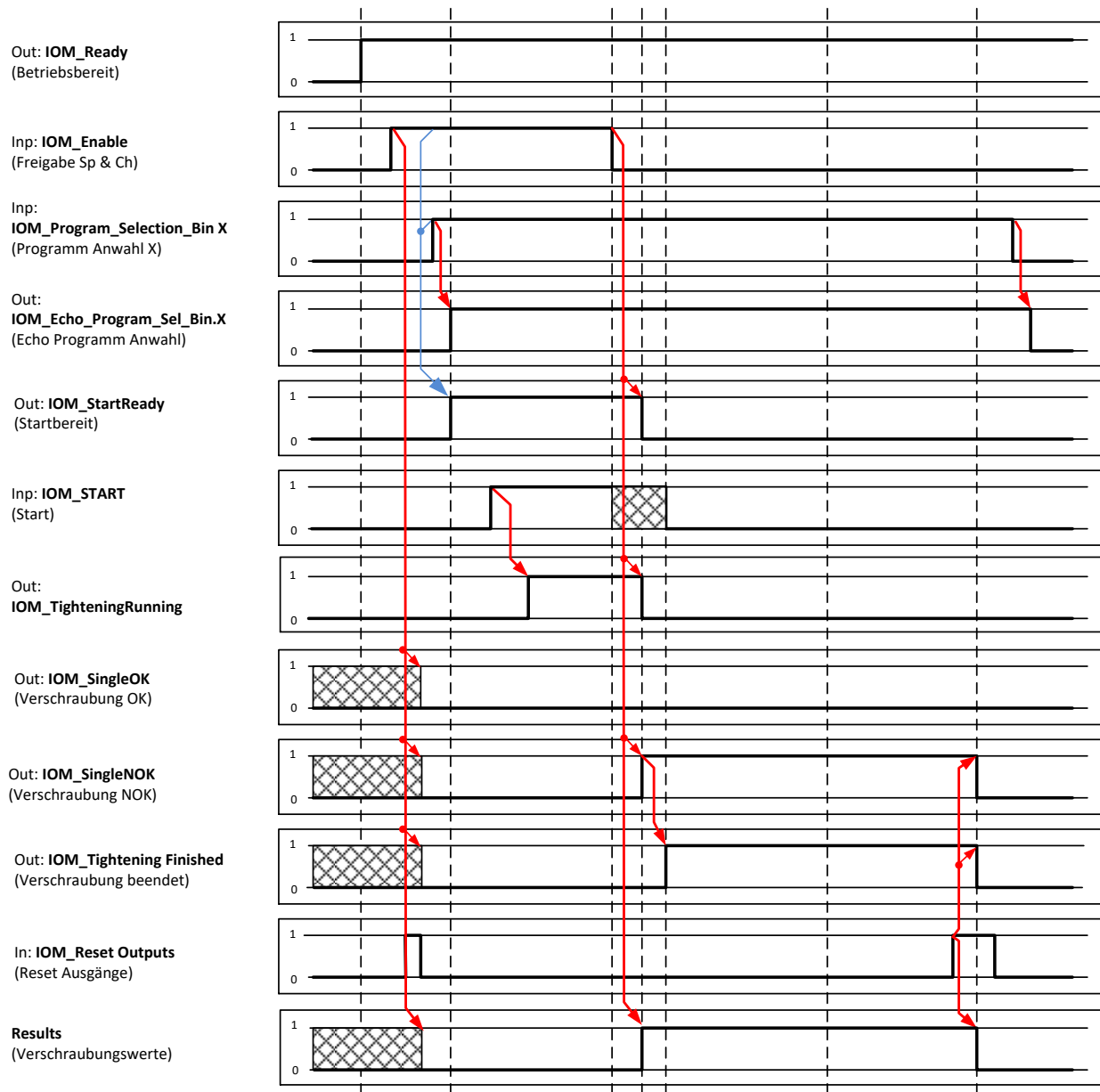
Vorraussetzung: Modus: „ProgramTakeOverWith = Enable“



3.7. Hinweise zur Programmübernahme

- **Wichtig:**
Die Programmspiegelung muss beachtet werden. Das Programm ist erst gültig zur Übernahme, wenn das Programm auf den gespiegelten Echo-Signalen anliegt.
- Wenn die Programmselektion geändert wird, kann sich das Echo Signal während der Verschraubung ändern.

3.8. Ablauf: Abbruch einer Verschraubung über Enable



3.8.1. Hinweis zum Enable:

- Achtung über Enable kann im Versions-Stand 1.5.3 und früher nur eine Verschraubung freigegeben werden.
- Nach einer Verschraubung muss ein erneutes Enable erteilt werden.

3.9. Hardwarekonfiguration (über Sycon.Net)

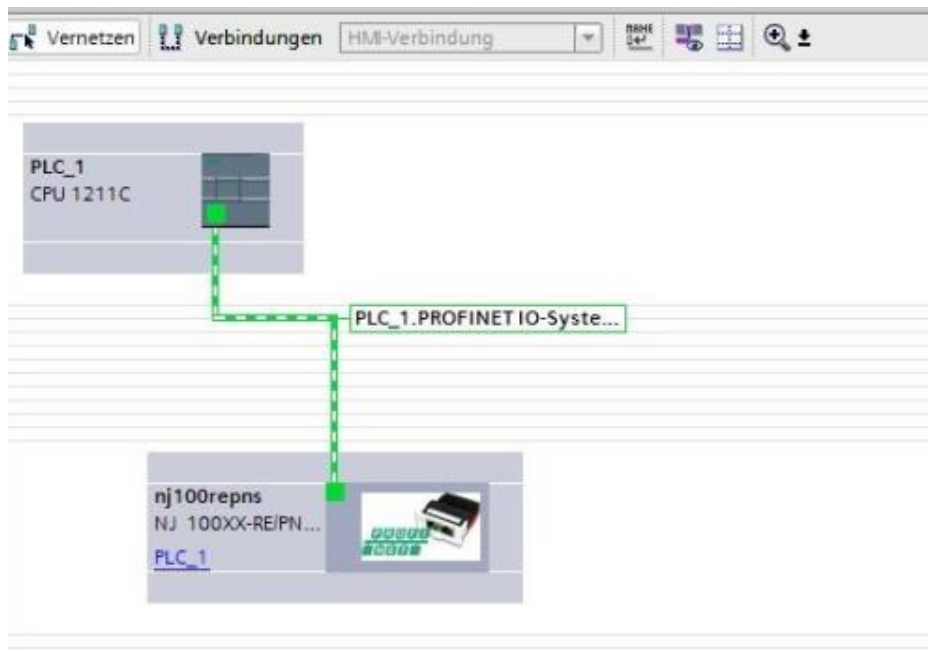
Info: Bei Sycon.Net handelt es sich um das Konfigurationstool der Firma *Hilscher*.
Im Folgenden sind die Konfigurationsseiten für die I/O-Modulkonfiguration der SW-Applikation dargestellt.

Für *ProfiNet* konfigurieren Sie ein NJ-Modul NJ_100x_RE_PNS.
Für *ProfiBus* konfigurieren Sie ein NJ-Modul: NJ_100xx-DP/DBS.

Im Beispiel wird nur ein ProfiNet dargestellt.



Komplettes Verbindungsschema mit einem SPS Master



3.10. Modulkonfiguration über Sycon.Net

Wichtig: Bei Sycon.NET muss die Modul-Konfiguration der Ein-/Ausgänge aus **Sicht des Masters (SPS)** erfolgen.

	Slot	Sub Slot	!	
+	0		⌘	NJ 100XX-RE/PNS V3.5.35 - V3.x [162x.100]
+	1			2 Byte Output
+	2			8 Bytes Input
+	3			8 Bytes Input
+	4			8 Bytes Input

netDevice - Configuration NJ_100XX-RE_PNS_V3.5.35_-_V3.x[NJ 100XX-RE/PNS V3.5.35 - V3.x]<nj100repns>

IO Device: NJ 100XX-RE/PNS V3.5.35 - V3.x Device ID: 0x0113
Vendor: Hilscher Gesellschaft für Systemautomation mbH Vendor ID: 0x011E

Navigation Area:

- Settings
 - Driver
 - netX Driver
 - Device Assignment
 - Firmware Download
 - Configuration
 - General
 - Modules
 - Signal Configuration
 - Address Table
 - Device Settings
 - Description
 - Device Info
 - Module Info
 - GSDML Viewer

Address Table

Display mode: Hexadecimal CSV Export

Inputs:

Module	Submodule	Type	Length	Address
2 Byte Output	2 Byte Output	IB	0x0002	0x0000

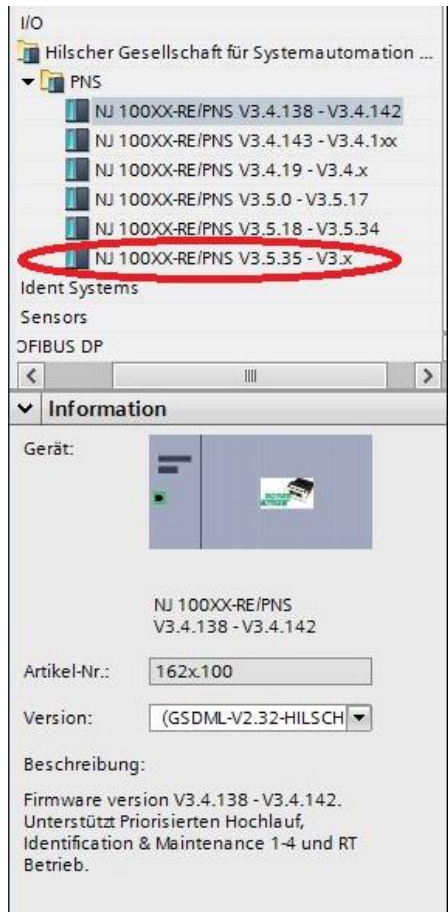
Outputs:

Module	Submodule	Type	Length	Address
8 Bytes Input	8 Bytes Input	QB	0x0008	0x0000
8 Bytes Input	8 Bytes Input	QB	0x0008	0x0008
8 Bytes Input	8 Bytes Input	QB	0x0008	0x0010

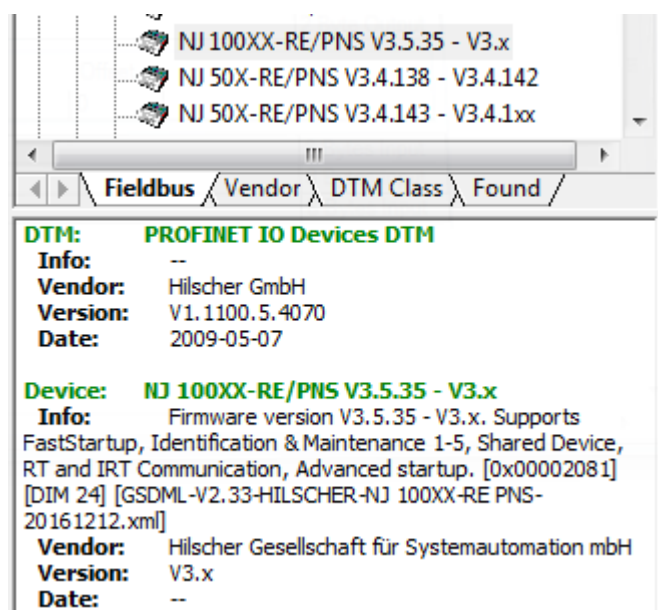
3.11. GSD

Bei Sycon.NET ist

- für ProfiNet folgendes GSD-Modul zu verwenden.



Sycon.Net:



4. Mapping

4.1. Mapping-Datei (mapping.map)

```
// Feldbus-Mapping
Feldbus@IN.B0.0 = IOM_Enable

Feldbus@IN.B0.2 = IOM_Start
Feldbus@IN.B0.6 = IOM_Reset_Outputs
Feldbus@IN.B0.7 = IOM_Reset_Servo_Alarm

Feldbus@IN.B1.0 = IOM_Program_Selection_Bin@[6]

Feldbus@OUT.B0.0 = IOM_Ready
Feldbus@OUT.B0.1 = IOM_Start_Ready
Feldbus@OUT.B0.2 = IOM_Tightening_Finished
Feldbus@OUT.B0.3 = IOM_SingleOK
Feldbus@OUT.B0.4 = IOM_SingleNOK
Feldbus@OUT.B0.5 = IOM_InCycle           // Threshold reached
Feldbus@OUT.B0.6 = IOM_TighteningRun     // TighteningRun: True if Tightening/Tool running.
                                           // Stopp on FSTA it is also true
Feldbus@OUT.B0.7 = IOM_No_Servo_Alarm

Feldbus@OUT.B1.0 = IOM_Echo_Program_Sel_Bin@[6]
Feldbus@OUT.B1.6 = IOM_StartTriggerPressed // Starttrigger is pressed
Feldbus@OUT.B1.7 = IOM_ReverseSwitchSelected

Feldbus@OUT.B2 = IOM_ProgramNumber_Enabled@[B1] // Enabled Program
Feldbus@OUT.B3 = IOM_ProgramNumber_Result@[B1]  // Program number that was tightened
Feldbus@OUT.B4 = IOM_TorqueActNm_X10@[B2]
Feldbus@OUT.B6 = IOM_TorqueMinNm_X10@[B2]

Feldbus@OUT.B8 = IOM_TorqueMaxNm_X10@[B2]

Feldbus@OUT.B10 = IOM_AngleActDeg@[B2]
Feldbus@OUT.B12 = IOM_AngleMinDeg@[B2]
Feldbus@OUT.B14 = IOM_AngleMaxDeg@[B2]
Feldbus@OUT.B16 = IOM_CurrentAmp_X10@[B2]
Feldbus@OUT.B18 = IOM_Torque2ActNm_X10@[B2]
Feldbus@OUT.B20 = IOM_Angle2ActDeg@[B2]
```

4.2. Device-Datei (devices.map)

```
[Feldbus]
Type = Feldbus
Inputs = B2
Outputs = B24
```

4.3. Signals-Datei (signal.map)

[IOM_Tightening_Finished]
Type = Bool

[IOM_Enable]
Type = Bool

[IOM_Ready]
Type = Bool

[IOM_Program_Selection_Bin]
Type = UInt

[IOM_Echo_Program_Sel_Bin]
Type = UInt

[IOM_Program_Number]
Type = String
Length = 2

[IOM_TorqueNm]
Type = String
Length = 6

[IOM_AngleDeg]
Type = String
Length = 5

[IOM_CurrentAmp]
Type = String
Length = 4

[IOM_SeparatorBlank]
Type = String
Length = 1

[IOM_Torque2Nm]
Type = String
Length = 6

[IOM_Angle2Deg]
Type = String
Length = 5

[IOM_InCycle]
Type = Bool

[IOM_ToolRun]
Type = Bool

[IOM_ToolStartTriggerPressed]
Type = Bool

[IOM_ToolReverseSelected]

Type = Bool

[IOM_ProgramNumber_Result]

Type = UInt

Length = 1

[IOM_ProgramNumber_Enabled]

Type = UInt

Length = 1

[IOM_TorqueActNm_X10]

Type = UInt

Length = 2

[IOM_TorqueMinNm_X10]

Type = UInt

Length = 2

[IOM_TorqueMaxNm_X10]

Type = UInt

Length = 2

[IOM_AngleActDeg]

Type = UInt

Length = 2

[IOM_AngleMinDeg]

Type = UInt

Length = 2

[IOM_AngleMaxDeg]

Type = UInt

Length = 2

[IOM_CurrentAmp_X10]

Type = UInt

Length = 2

[IOM_Torque2ActNm_X10]

Type = UInt

Length = 2

[IOM_Angle2ActDeg]

Type = UInt

Length = 2

5. Signalbeschreibung ProfiNet

Signal/ Signalname für Mapping	I/O-Typ	Typ	Beschreibung	Zusatzinfos
EnableTool IOM_Enable_Tool	Input	Bool 1 Bit	Das Signal <IOM_EnableTool> wird innerhalb der Mapping-Datei mit dem Signal <IOM_EnableChannel> UND verknüpft und dadurch das Signal <IOM_Enable> erzeugt. Beide Signale müssen anliegen um eine Freigabe auf dem Controller zu erzeugen.	0 = gesperrt 1 = freigegeben Das Signal ist nicht als Einzelsignal verwendbar
EnableChannel IOM_Enable_Channel	Input	Bool 1 Bit	Nach dem beide Signale gesetzt sind, wird der Schrauber freigegeben und ist danach schraubbereit. Je nach Konfiguration wird entweder mit dem Enable <IOM_Enable> oder mit dem Starttaster <IOM_Start> die Programmnummer übernommen.	
Enable (Program) IOM_Enable	Gener. Input-Signal	Bool 1 Bit	Neu: Das Signal <IOM_EnableTool> wird direkt auf Bit0 gemapped. In dieser Konfiguration wird es nicht über die unten genannten Signale gebildet. Das Signal <IOM_EnableTool> wird innerhalb der Mapping-Datei mit dem Signal <IOM_EnableChannel> UND verknüpft und dadurch das Signal <IOM_Enable> erzeugt. Beide Signale müssen anliegen um dieses Signal zu erzeugen und auszulösen. Je nach Konfiguration wird entweder mit dem Enable <IOM_Enable> oder mit dem Starttaster <IOM_Start> die Programmnummer übernommen. Programmnummer 0 ist nicht zulässig. Verknüpfung: <IOM_Enable_Tool> & <IOM_Enable_Channel> = IOM_Enable	0 = gesperrt 1 = freigegeben
Start (Tool) IOM_Start	Input	Bool 1 Bit	Der Verschraubungsvorgang / Verschraubungsprozess wird gestartet. Dieses Signal entspricht dem Starttaster des Werkzeugs Der Verschraubungsvorgang startet nur wenn vorher das Signal <IOM_StartReady> gesetzt war, eine Freigabe anliegt und kein Systemfehler vorhanden ist.	0 = Stopp 1 = Start
Ready IOM_Ready	Output	Bool 1 Bit	Betriebsbereit Das Schraubsystem meldet, dass es betriebsbereit ist und keine interne Funktionsstörung anliegt. Es kann eine Freigabe zum Schrauben erteilt werden.	0 = System Störung 1 = System Betriebsbereit

ProgramSelectionBinär @[n] IOM_Program_Selection_Bin @[n]	Input	n Bool n Bit	Programmanwahl binär Binärcodierte Anwahl eines Programms. ACHTUNG Wenn in den Einstellungen die Softwareoption "<i>Programmanwahl bei START</i>" gesetzt ist, muss das Programm einen Zyklus vor dem Signal <IOM_StartTool> gesetzt sein. Das Programm auf den n-Bits wird binärcodiert angelegt. Bsp: Bit 0 und Bit 1 gesetzt-> Programm 3 angewählt. [n]: stellt die Anzahl der Bits zur Verfügung ->[bits]	0 = nicht angewählt 1 = Bit angewählt Bit0 = binär 1 Bit1 = binär 2 Bit3 = binär 4 Bit4 = binär 8 ... usw.
StartReady IOM_Start_Ready	Output	Bool 1 Bit	Schraubbereit: Alle Voraussetzungen für eine Verschraubung sind erfüllt. Es wird gesetzt sobald eine Freigabe anliegt. Jetzt kann das Signal <IOM_Start> folgen. Die Programmnummer muss größer Null sein. WICHTIG: Das Signal <IOM_Start_Ready> wird automatisch auf 0 gesetzt sobald die Programm Nummer = 0 (trotz anliegender Freigabe) anliegt. Es wird nicht geprüft ob angewählte Programme auch vorhanden sind. Verknüpfung: <IOM_Enable> && (#ProgrammNr > 0) -> IOM_Start_Ready	0 = Keine Freigabe 1 = System bereit zum Verschrauben
TighteningFinished IOM_Tightening_Finished	Output	Bool 1 Bit	Zeigt an, dass eine Verschraubung fertig ist. (Der gestartete Verschraubungsprozess wurde durchgeführt) Die Verschraubungssignale OK, NOK, sowie die Werte sind erst gültig, wenn dieses Bit gesetzt ist. Das Bit wird zurückgenommen wenn • das Bit „Freigabe“ von 0 auf 1 wechselt • das Bit „Start“ von 0 auf 1 wechselt Wird gesetzt, wenn Ergebnisse zum Lesen verfügbar sind. Das Signal kann mit <IOM_Reset_Results> und <IOM_Reset_Outputs> zurückgesetzt werden.	0 = Verschraubung nicht fertig 1 = Verschraubung fertig

SingleOk IOM_SingleOK	Output Status	Bool 1 Bit	Schraubstatus: Einzel OK Das Schraubsystem meldet, dass der Schraubvorgang (einer Verschraubung) mit dem Ergebnis OK abgeschlossen wurde. Das Signal ist erst gültig, wenn das Signal <IOM_Tightening_finished> auf 1 gesetzt ist. ACHTUNG: Aus Sicherheitsgründen immer beide Signale OK und NOK auf Relevanz prüfen Gesetzt: - Wenn eine Verschraubung mit dem Status OK abgearbeitet wurde. Rückgesetzt: - Werkstück wird identifiziert - Beim Start des Schraubers - Bei Signal <IOM_Reset_Results> und bei <IOM_Reset_Output> - Bei Abbruch	0 = Kein Zustand 1 = Verschraubung OK
SingleNok IOM_SingleNOK	Output Status	Bool 1 Bit	Schraubstatus: Einzel NOK Das Schraubsystem meldet, dass der Schraubvorgang mit dem Ergebnis NOK abgeschlossen wurde. Das Signal ist erst gültig wenn das Signal <IOM_Tightening_finished> auf 1 gesetzt ist. ACHTUNG: Aus Sicherheitsgründen immer beide Signale OK und NOK auf Relevanz prüfen Gesetzt: - Wenn eine Verschraubung mit dem Status NOK abgearbeitet wurde. Rückgesetzt: - Werkstück wird identifiziert - Beim Start des Schraubers - Bei Signal <IOM_Reset_Results> und bei <IOM_Reset_Output>	0 = Kein Zustand 1 = Verschraubung NOK

EchoProgramSelectionBin @[n] IOM_Echo_Program_Sel_Bin @[6]	Output	n Bool n Bit	Rückgabe des aktuell anliegenden Programms an der I/O. Es handelt sich um ein reines ECHO, d.h. eine Spiegelung der anliegenden Programm-Signale. Es enthält KEINEN Snapshot des aktuell ausgeführten Programms. Programm – Nummer null ist nicht zulässig. [n]: stellt die Anzahl der Bits zur Verfügung	0 = nicht angewählt 1 = Bit angewählt Bit0 = binär 1 Bit1 = binär 2 Bit3 = binär 4 Bit4 = binär 8 ... usw.
IOM_InCycle	Output	Bit	Dieses Signal wird gesetzt wenn in einem Voran zug (Kennung: VS) das Schwellmoment erreicht wird. Es wird nicht gesetzt wenn das Schwellmoment= 0, d.h. deaktiviert ist.	0 = Schwellmoment nicht erreicht 1 = Schwellmoment erreicht
IOM_TighteningRun	Output	Bit	Wird gesetzt, wenn der Schrauber losdreht. Bei einem Findestart (FSTA – Starttasterloslassen erlaubt) bleibt es gesetzt, da der Schraubvorgang nach wie vor aktiv ist, der Schrauber in diesem Fall jedoch nicht dreht.	0 = Schraubvorg. Nicht aktiv 1= Schraubvorgang aktiv
IOM_ToolRun	Output	Bit	Diese Signal verhält sich ähnlich wie IOM_TighteningRun. Wird gesetzt, wenn der Schrauber dreht. Bei einem Findestart (FSTA – Starttasterloslassen erlaubt) wird es jedoch zurückgenommen. Das heißt, es ist nur gesetzt wenn das Werkzeug tatsächlich dreht.	0 = Schrauber steht 1 = Schrauber dreht

IOM_StartTriggerPressed	Output	Bit	Dieses Signal wird gesetzt wenn das interne Startsignal am Schrauber anliegt. Das Startsignal kann auch über die I/O-Karte oder eine Ethernetkommunikationsebene gesetzt werden, z.B. XML) Das Signal wird zurückgesetzt wenn der Start zurückgenommen wird, oder Schraubablauf beendet wurde.	0 = Startsignal liegt an 1 = Kein Startsignal
IOM_ReverseSwitchSelected	Output	Bit	Dieses Signal wird gesetzt, wenn am Werkzeug/Schrauber der Linkslauf angewählt (Schalter) wird. True wenn der Schalter angewählt wird. Es muss kein Enable vorhanden sein.	0 = Linkslauf angewählt 1 = kein Linkslauf
IOM_UntighteningSelected	Output	Bit	Dieses Signal wird gesetzt wenn das Lösen in Programm 0 aktiviert wurde. True wenn der Linkslauf aktiv ist. Nur wenn ein Enable vorhanden ist.	0 = kein Linkslauf aktiv 1 = Linkslauf aktiv

Schraubdaten

Signal Signalname für Mapping	I/O- Typ	Typ	Beschreibung	Zusatzinfos
ProgramNumber @[size] IOM_Program_Number@[B2]	Output Result	2 Char 2 Byte	Schraubergebnis: Verschraubte Programmnummer mit 2 Byte Die Programmnummer wird mit High- u. Lowbyte zweistellig ausgegeben Bsp: Programmnummer = 12 -> HB= 1 LB = 2 [size] Anzahl der Bytes (momentan max 2)	Format: HB LB (big endian)
ProgramNumberEnabled IOM_ProgramNumber_Enabled	Output Result	1 Byte	Wird gesetzt sobald über <i>IOM_Enable</i> das Programm angewählt wird. Es wird mit dem Senden eines Ergebnisses zurückgesetzt. Enthält die freigegebene Programmnummer.	ProgrammNr: Byte z.B 01
ProgramNumberResult IOM_ProgramNumber_Result	Output Result	1 Byte	Schraubergebnis: Verschraubte Programmnummer mit 1 Byte Wird gesetzt sobald ein Ergebnis versendet wird- Während der Verschraubung ist es 0. Wird bei <i>IOM_ResetOutput</i> zurückgesetzt.	ProgrammNr: Byte z.B 01
SeparatorBlank @[B1] IOM_SeparatorBlank@[B1]	Output	Char 1 Byte	Separator Der Separator enthält das Zeichen <BLANK> 0x20.	<BLANK> Hex: 0x20
TorqueNm @[B6] IOM_TorqueNm @[B6]	Output Result	6 Char 6 Byte	Schraubergebnis: Drehmoment in Nm Das Drehmoment wird im Format <xxxx.x> mit fix 6 Stellen und einem Punkt als Dezimalsparator ausgegeben	Format: xxxx.x Beispiel: 0012.5 -> 12,5 Nm
AngleDeg @[B5] IOM_AngleDeg @[B5]	Output Result	5 Char 5 Byte	Schraubergebnis: Drehwinkel in Grad Der Drehwinkel wird im Format <xxxxx> mit 5 Stellen ausgegeben.	Format: xxxxx Beispiel: 03600 -> 3600 Grad
CurrentAmp @[B4] IOM_CurrentAmp @[B4]	Output Result	4 Char 4 Byte	Schraubergebnis: Strom in Ampere Der Strom wird im Format <xx.x> mit 4 Stellen ausgegeben.	Format: xx.x Beispiel: 12.3 -> 12,3 Ampere

Torque2Nm @[B6] IOM_Torque2Nm @[B6]	Output Result	6 Char 6 Byte	Schraubergebnis: Drehmoment von Transducer 2 in Nm Das Drehmoment wird im Format <xxxx.x> mit fix 6 Stellen und einem Punkt als Dezimalsparator ausgegeben	Format: xxxx.x Beispiel: 0012.5 -> 12,5 Nm
Angle2Deg @[B5] IOM_Angle2Deg @[B5]	Output Result	5 Char 5 Byte	Schraubergebnis: Drehwinkel von Geber 2 in Grad Der Drehwinkel wird im Format <xxxxx> mit 5 Stellen ausgegeben.	Format: xxxxx Beispiel: 03600 -> 3600 Grad
TorqueActNm_X10 IOM_TorqueActNm_X10 @[B2]	Output Result	Word Lb/Hb	Schraubergebnis: Drehmoment in Nm Geber 1 Das Drehmoment wird mit 10 multipliziert und hier als Word abgelegt. Format: Lowbyte/Highbyte. Durch die Multiplikation ergibt sich eine Nachkommastelle.	LB: 0x7D HB: 0x00 -> 125 -> 12,5 Nm
TorqueMinNm_X10 IOM_TorqueMinNm_X10 @[B2]	Output Result	Word Lb/Hb	Schraubergebnis: minimales Drehmoment in Nm Das Drehmoment wird mit 10 multipliziert und hier als Word abgelegt. Format: Lowbyte/Highbyte. Durch die Multiplikation ergibt sich eine Nachkommastelle.	LB: 0x7D HB: 0x00 -> 125 -> 12,5 Nm
TorqueMaxNm_X10 IOM_TorqueMaxNm_X10 @[B2]	Output Result	Word Lb/Hb	Schraubergebnis: maximales Drehmoment in Nm Das Drehmoment wird mit 10 multipliziert und hier als Word abgelegt. Format: Lowbyte/Highbyte. Durch die Multiplikation ergibt sich eine Nachkommastelle.	LB: 0x7D HB: 0x00 -> 125 -> 12,5 Nm
AngleActDeg @[B1] IOM_AngleActDeg @[B1]	Output Result	Word Lb/Hb	Schraubergebnis: Drehwinkel in Grad, 1.Geber Der Drehwinkel wird als Word mit Lowbyte und Highbyte ausgegeben	LB: 0x69 HB: 01 -> 361 Grad
AngleMinDeg @[B1] IOM_AngleMinDeg @[B1]	Output Result	Word Lb/Hb	Schraubergebnis: minimaler Drehwinkel in Grad Der Drehwinkel wird als Word mit Lowbyte und Highbyte ausgegeben	LB: 0x69 HB: 01 -> 361 Grad
AngleActDeg @[B1] IOM_AngleMaxDeg @[B1]	Output Result	Word Lb/Hb	Schraubergebnis: maximaler Drehwinkel in Grad Der Drehwinkel wird als Word mit Lowbyte und Highbyte ausgegeben	LB: 0x69 HB: 01 -> 361 Grad
CurrentAmp_X10 @[B1] IOM_CurrentAmp_X10 @[B1]	Output Result	Word Lb/Hb	Schraubergebnis: Strom in Ampere Der Strom wird mit 10 multipliziert und als Word abgelegt. Format:	LB: 0x5C HB: 0x00 -> 92 -> 9,2 Ampere

			Lowbyte/Highbyte. Durch die Multiplikation ergibt sich eine Nachkommastelle.	
Torque2ActNm_X10 IOM_TorqueActNm_X10 @[B2]	Output Result	Word Lb/Hb	Schrauberergebnis: Drehmoment in Nm Geber 2 Das Drehmoment wird mit 10 multipliziert und hier als Word abgelegt. Format: Lowbyte/Highbyte. Durch die Multiplikation ergibt sich eine Nachkommastelle.	LB: 0x7D HB: 0x00 -> 125 -> 12,5 Nm
Angle2ActDeg @[B1] IOM_AngleActDeg @[B1]	Output Result	Word Lb/Hb	Schrauberergebnis: Drehwinkel in Grad, Geber 2 Der Drehwinkel wird als Word mit Lowbyte und Highbyte ausgegeben	LB: 0x69 HB: 01 -> 361 Grad

5.1.1. Weitere Signale

Signal / Signalname für Mapping	I/O-Typ	Typ	Beschreibung	Zusatzinfos															
SetOutput.[x] IOM_Output.x	Input	Bit	Setzt einen Ausgang x auf dem Regler 0 = Relais /Output off 1 = activate Relais/Output on Der Ausgang muss im I/O-Mapping des SpindleRunners als setzbarer Ausgang gemapped sein. Das mappbare I/O Signal des SR lautet: <OutExternCtrlId-Bx>. X = 1..4 Ausgänge	0 = Relais /Output off 1= activate Relais/Output on															
TorqueState @[2] IOM_TorqueState @[2]	Output Status	2 Bits	Schraubstatus: 2 Bits für den Drehmoment Ergebnisstatus Wird nach einer Verschraubung gesetzt. <table><tr><td>Bit1</td><td>Bit0</td><td>State</td></tr><tr><td>0</td><td>0</td><td>Init value</td></tr><tr><td>1</td><td>0</td><td>Torque too high</td></tr><tr><td>0</td><td>1</td><td>Torque OK</td></tr><tr><td>1</td><td>1</td><td>Torque too low</td></tr></table> Zurückgesetzt über das Signal <IOM_Reset_Results> (Achtung: Nicht bei <IOM_Reset_Output>)	Bit1	Bit0	State	0	0	Init value	1	0	Torque too high	0	1	Torque OK	1	1	Torque too low	Format: Bit0, Bit1 (LB, HB) (Little Endian)
Bit1	Bit0	State																	
0	0	Init value																	
1	0	Torque too high																	
0	1	Torque OK																	
1	1	Torque too low																	

AngleState @[2] IOM_AngleState @[2]	Output Status	2 Bits	Schraubstatus: 2 Bits für den Winkel-Ergebnisstatus Wird nach einer Verschraubung gesetzt. <table><tr><td>Bit1</td><td>Bit0</td><td>State</td></tr><tr><td>0</td><td>0</td><td>Init value</td></tr><tr><td>1</td><td>0</td><td>Angle too high</td></tr><tr><td>0</td><td>1</td><td>Angle OK</td></tr><tr><td>1</td><td>1</td><td>Angle too low</td></tr></table> Zurückgesetzt über das Signal <IOM_ResetResults> (Achtung: Nicht bei <IOM_ResetOutput>)	Bit1	Bit0	State	0	0	Init value	1	0	Angle too high	0	1	Angle OK	1	1	Angle too low	Format: Bit0, Bit1 (LB, HB) (Little Endian)
Bit1	Bit0	State																	
0	0	Init value																	
1	0	Angle too high																	
0	1	Angle OK																	
1	1	Angle too low																	

5.2. Signalbeschreibung ProfiBus

Folgende Signale können momentan **nur bei ProfiBus** verwendet werden:

Signal/ Signalname für Mapping	I/O-Typ	Typ	Beschreibung	Zusatzinfos
Disable_Tool IOM_Disable_Tool <i>Deshabilitar_Tool</i>	Input	Bool 1 Bit	Das Signal <i>IOM_DisableTool</i> aktiviert/deaktiviert das Werkzeug. Wenn die Signalfanke von High auf Low wechselt, wird ein internes Enable generiert und das Werkzeug mit dem anliegenden Programm freigegeben. Das Werkzeug ist schraubbereit. Es kann mit dem Starttaster gestartet werden. Liegt nach dem Booten ein Low an wird sofort ein internes Enable erzeugt.	0 = Enabled 1 = Disabled
ProgramSelectionHbLb @[B2] IOM_Program_Selection_HbLb @[B2] <i>Set_PSET</i>	Input	Word 16 Bit	16 Bits für die Programmanwahl über Hb/Lb – Byte (Big Endian) Word codierte Anwahl eines Programms. Das Programm wird mit dem Signal <i>#IOM_Enable</i> übernommen. Im „ <i>Permanent Enable Mode</i> “, wird das Programm durch einen Programmwechsel übernommen und das Werkzeug sofort freigegeben. Range: 01..99 Bsp: Byte 2 = 0x00 Byte3 = 0x05 -> Programm 5 angewählt.	Byte 2 = HB Byte 3 = LB
Reset_Results IOM_Reset_Results <i>Reset_Resultados</i>	Input	Bool Bit	Setzt die Statis: OK, NOK, Finished, TorqueState, AngleState sowie die Ergebniswerte für Torque und Angle auf 0 zurück. Folgende Signale werden bei IOM_Reset_Result = 1 zurückgesetzt: • IOM_TorqueState (<i>Par_Result</i> B0/B1) • IOM_AngleState (<i>Angelo_Result</i> B0/B1) • TorqueDecimal (<i>Par_Entero</i>) • TorqueDecimalDigit (<i>Par_Decimal</i>) • IOM_AngleDecimalValue (<i>Par_Angulo</i>) • IOM_Tightening_Finished (<i>Ciclo_completo</i>) • IOM_SingleOK/ IOM_Single NOK	0 = Nothing 1 = Reset
Selected Program/ IOM_Program_Selection_HbLb @[B2] <i>RunningPSET</i>	Output	Bool 16 Bit	Rückgabe des aktuell anliegenden Programms an der Controller-I/O. Spiegelt das angewählte Programm. Es handelt sich um ein reines ECHO, d.h. eine Spiegelung des anliegenden Programm-Signals. Es enthält KEINEN Snapshot des aktuell ausgeführten Programms. Programm-Nummer null ist nicht zulässig.	Byte2 = HB Byte3 = LB

ToolUnlocked IOM_ToolUnlocked <i>Habilitada</i>	Output	Bool Bit	Werkzeug freigegeben (enabled) Dieses Signal wird gesetzt, wenn das Signal <i>IOM_Disable_Tool</i> den Zustand Low einnimmt. Das Signal gibt die invertierte Spiegelung von <i>IOM_Disable_Tool</i> aus.	0 = Tool locked/ disabled 1 = Tool unlocked / enabled
AlifeNo IOM_AlifeNo <i>Contador Vida</i>	Input/ Output	Byte 8 Bit	Fortlaufende Live-Nummer / Lebenszeichen Die Alife-Nummer wird vom Master gesendet. Sie wird auf die Ausgangssignale vom Controller gespiegelt. Beim Controller erfolgt keine Auswertung dieser Nummer. Es handelt sich hierbei um einen Zähler. Die Alife-Nummer wird verwendet um zu überprüfen, ob der Feldbus "lebt". Wenn die gleiche Nummer vom Master empfangen wird, wird der Wert um eins erhöht.	Spiegelung auf Ausgänge
Tool model Number IOM_ModelType <i>Modelo_Atornillador</i>	Input/ Output	Byte 8 Bit	Controllermodell-Nummer Die Modellnummer wird vom Master gesendet. Die Modellnummer identifiziert den Controllertyp und enthält die Controllermodell-Nummer (z.B. 1=PF 2=MPRO 3= SMX). Dieser Ausgang hat keine interne Funktionalität. Da die Modelle nicht genauer spezifiziert sind, wird die Nummer die SMX-Ausgangssignal (PLC-Input Bereich) 7.0 ModelNumber gespiegelt	Spiegelung auf Ausgänge
Disconnect_Tool IOM_Disconnect_Tool	Input	Bool Bit	Sobald der Master (die SPS) das Werkzeug entfernen möchte, setzt sie das Eingang-Signal <i>IOM_Disconnect_Tool</i> . Wenn es beim SMX Controller möglich ist das Werkzeug zu entfernen, wird das Ausgangs-Signal <i>IOM_SafeToDisconTool</i> (Bit 0.6).auf 1 gesetzt.	
SafeToDisconTool IOM_SafeToDisconTool	Output	Bool Bit	Über das Signal <i>IOM_Disconnect_Tool</i> fordert der Master (SPS) einen Werkzeugwechsel an. Wenn es bei der SMX Steuerung möglich ist das Werkzeug zu entfernen, wird das Ausgangs-Signal <i>IOM_SafeToDisconTool</i> (Bit 0.6) auf High gesetzt. Da es bei einer SMX immer möglich ist, das Werkzeug zu entfernen wird das Signal <i>IOM_Disconnect_Tool</i> lediglich gespiegelt	0 = not ready 1= Safe to disconnect
GreenLight IOM_GreenLight <i>Luz_Verde</i>	Output	Bool Bit	Grüne LED Im Notmodus blinkt die grüne LED. (Output.1 wird ebenfalls angesteuert). Wenn der Notmodus nicht aktiv ist, wird dieses Signal auf Low gesetzt.	0 = green off 1 = green on

YellowLight IOM_YellowLight <i>Luz_Amarilla</i>	Output	Bool Bit	Gelbe LED Im Notmodus blinkt die gelbe LED. (Output.2 wird ebenfalls angesteuert). Wenn der Notmodus nicht aktiv ist, wird dieses Signal auf Low gesetzt.	0 = Yellow off 1 = Yellow on
--	--------	-------------	---	---------------------------------

Schraubdaten-Übertragung

Signal Signalname für Mapping	I/O-Typ	Typ	Beschreibung	Zusatzinfos
Torque decimal before point IOM_TorqueDecimalBeforePoint@[B2] <i>Par_Enter0</i>	Output Result	Word 2 Byte	Schraubergebnis: Drehmoment in Nm. Enthält die Vorkommastelle als 16 Bit Wert Das Drehmoment wird als Word ausgegeben: HB, LB (Big Endian)	Format: HB LB (Big Endian) 23.56 -> 0x00 0x17 (#23) -1.56 -> 0xFF 0xFF (#-1)
Torque decimal after point IOM_TorqueDecimalAfterPoint@[B2] <i>Par_Decimal</i>	Output Result	Word 2 Byte	Schraubergebnis: Drehmoment in Nm. Enthält die Nachkommastelle als 16 Bit Wert Das Drehmoment wird als Word ausgegeben: HB, LB (Big Endian)	Format: HB LB (big endian) 23.56 -> 0x38 (#56) -1.32 -> 0x20 (#32)
Angle decimal value IOM_AngleDecimalValue @[B2] <i>Par_Angulo</i>	Output Result	Word 2 Byte	Schraubergebnis: Drehwinkel in Grad Der Drehwinkel wird als Word (unsigned int) mit 16 Bit ausgegeben.	Format: HB LB (big endian) Beispiel: 361 Grad -> 0x01 0x69

Beispiel:

Dies ist die Übertragung von 2,1 Nm und 361 Grad

(Werte unten dezimal, Beispiel Winkel: LB-> 105 = 0x69 HB-> 1= 1 => 361 Grad)

		Eingänge	%IB5	ARRAY [0..7] OF BYTE	
Application.PLC_PRG.tsOutStates.iByte0State		Eingänge[0]	%IB5	BYTE	149
Application.PLC_PRG.tsOutStates.iByte1State		Eingänge[1]	%IB6	BYTE	1
Application.PLC_PRG.tsOutStates.byEnabledProgram		Eingänge[2]	%IB7	BYTE	0
Application.PLC_PRG.tsTightResult.iProgram		Eingänge[3]	%IB8	BYTE	1
Application.PLC_PRG.tsTightResult.iTorqueActX10.abVal[0]		Eingänge[4]	%IB9	BYTE	21
Application.PLC_PRG.tsTightResult.iTorqueActX10.abVal[1]		Eingänge[5]	%IB10	BYTE	0
Application.PLC_PRG.tsTightResult.iTorqueMinX10.abVal[0]		Eingänge[6]	%IB11	BYTE	0
Application.PLC_PRG.tsTightResult.iTorqueMinX10.abVal[1]		Eingänge[7]	%IB12	BYTE	0
		Inputs PS	%IB13	Enumeration of BYTE	GOOD

		Eingänge	%IB14	ARRAY [0..7] OF BYTE	
Application.PLC_PRG.tsTightResult.iTorqueMaxX10.abVal[0]		Eingänge[0]	%IB14	BYTE	17
Application.PLC_PRG.tsTightResult.iTorqueMaxX10.abVal[1]		Eingänge[1]	%IB15	BYTE	0
Application.PLC_PRG.tsTightResult.iAngleActual.abVal[0]		Eingänge[2]	%IB16	BYTE	105
Application.PLC_PRG.tsTightResult.iAngleActual.abVal[1]		Eingänge[3]	%IB17	BYTE	1
Application.PLC_PRG.tsTightResult.iAngleMin.abVal[0]		Eingänge[4]	%IB18	BYTE	86
Application.PLC_PRG.tsTightResult.iAngleMin.abVal[1]		Eingänge[5]	%IB19	BYTE	1
Application.PLC_PRG.tsTightResult.iAngleMax.abVal[0]		Eingänge[6]	%IB20	BYTE	122
Application.PLC_PRG.tsTightResult.iAngleMax.abVal[1]		Eingänge[7]	%IB21	BYTE	1



Alfing Montagetechnik GmbH

Auguste-Kessler-Straße 20
73433 Aalen
Deutschland

Telefon: +49 (0) 7361 / 501 - 2701
Telefax: +49 (0) 7361 / 501 - 2709
E-Mail: info@amt.alfing.de
Web: www.alfing.de

Service Hotline

Telefon: +49 (0) 7361 / 501 -2999
E-Mail: service@amt.alfing.de