

Schnittstellenbeschreibung

Profinet Schnittstelle

Für Singlechannelsysteme: SMX100..400

Version 1.00

DE_SC_Schnittstellenbeschreibung_ProfiNet

Stand: 2019-05-14



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	02.05.2019	W. Mangold	14.05.2019

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	4
1.4.	Aktivieren der ProfiNet Kommunikation.....	4
1.5.	Konfiguration der ProfiNet Karte (über <i>Sycon.Net</i>)	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 <i>Sycon.Net</i>)	17
3.10.	Modulkonfiguration über <i>Sycon.Net</i>	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.....	22
5.1.1.	Weitere konfigurierbare Signale	26

1. Allgemein

1.1. Einleitung

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

Beschrieben wird das Protokoll „Gesteuertes Schrauben mit Schraubdatenübertragung“. Dieses Schnittstellenprotokoll ist standardmäßig auf den Einkanalssystemen eingestellt.

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

1.2. Softwarevoraussetzung: Versions-Information

Die hier beschriebene Signalschnittstelle erfordert folgendes Softwarerelease oder höher:

Release V1.5.0

1.3. Hardwarevoraussetzung: ProfiNet-Modul

Für die *ProfiNet*-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 Kommunikation

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

Über den Parameter *Program takeover with* oder *Programmübername* kann festgelegt werden, mit welchem Signal die Programmübernahme erfolgt.



PROFINET	
Profinet	<input type="checkbox"/> OFF
Program takeover with	Start ▼

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

Als *ProfiNet*-Modul wird eine *NetJACK*-ProfiNet 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 *ProfiNet*-Signale auf entsprechende I/O-Bytes/Bits festzulegen. Die *ProfiNet*-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 Schraubdatenübertragung“ ist voreingestellt.

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

Byte	Bit	Signal	Signalname im I/O Mapper	Version
0	0	Freigabe Spindel	IOM_Enable_Tool	V1.5.0
	1	Freigabe Kanal	IOM_Enable_Channel	V1.5.0
	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		n.c.	-	
	6		n.c.	-	V1.5.0
	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](.1)	V1.5.0
	1	Bit	Echo Programmanwahl binär 2	IOM_Echo_Program_Sel_Bin(.2)	V1.5.0
	2	Bit	Echo Programmanwahl binär 4	IOM_Echo_Program_Sel_Bin(.3)	V1.5.0
	3	Bit	Echo Programmanwahl binär 8	IOM_Echo_Program_Sel_Bin(.4)	V1.5.0
	4	Bit	Echo Programmanwahl binär 16	IOM_Echo_Program_Sel_Bin(.5)	V1.5.0
	5	Bit	Echo Programmanwahl binär 32	IOM_Echo_Program_Sel_Bin(.6)	V1.5.0
	6		n.c		
	7		n.c		

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	char	Program number HB	IOM_Program_Number[2]	1/2	0	V1.5.0
3	0..7	char	Program number LB	IOM_Program_Number[2]	2/2	1	V1.5.0
4	0..7	char	Separator Blank	IOM_SeparatorBlank	1/1	0x20	V1.5.0
5	0..7	char	Torque Nm 1/6	IOM_TorqueNm @[B6]	1/6	0	V1.5.0
6	0..7	char	Torque Nm 2/6	IOM_TorqueNm @[B6]	2/6	0	V1.5.0
7	0..7	char	Torque Nm 3/6	IOM_TorqueNm @[B6]	3/6	1	V1.5.0
8	0..7	char	Torque Nm 4/6	IOM_TorqueNm @[B6]	4/6	2	V1.5.0
9	0..7	char	Torque Nm 5/6 (decimal point)	IOM_TorqueNm @[B6]	5/6	.	V1.5.0
10	0..7	char	Torque Nm 6/6 (decimal place)	IOM_TorqueNm @[B6]	6/6	5	V1.5.0
11	0..7	char	Separator Blank	IOM_SeparatorBlank	1/1	0x20	V1.5.0
12	0..7	char	Angle Deg 1/5	IOM_AngleDeg @[B5]	1/5	0	V1.5.0
13	0..7	char	Angle Deg 2/5	IOM_AngleDeg @[B5]	2/5	3	V1.5.0
14	0..7	char	Angle Deg 3/5	IOM_AngleDeg @[B5]	3/5	6	V1.5.0
15	0..7	char	Angle Deg 4/5	IOM_AngleDeg @[B5]	4/5	0	V1.5.0
16	0..7	char	Angle Deg 5/5	IOM_AngleDeg @[B5]	5/5	0	V1.5.0
17	0..7	char	Separator Blank	IOM_SeparatorBlank	1/1	0x20	V1.5.0
18	0..7	char	Current Amp 1/4	IOM_CurrentAmp @[B4]	1/4	1	V1.5.0
19	0..7	char	Current Amp 2/4	IOM_CurrentAmp @[B4]	2/4	2	V1.5.0
20	0..7	char	Current Amp 3/4 (decimal point)	IOM_CurrentAmp @[B4]	3/4	.	V1.5.0
21	0..7	char	Current Amp 4/4 (decimal place)	IOM_CurrentAmp @[B4]	4/4	3	V1.5.0
22	0..7	char	Separator / reserve	IOM_SeparatorBlank	1/1		V1.5.0
23	0..7	char	Separator / reserve	IOM_SeparatorBlank	1/1		V1.5.0

Hinweis:

- Die Schraubwerte Torque und Angle sind mit einer Nachkommastelle angegeben.
- Werte mit zwei Nachkommastellen, werden ab 0,05 aufgerundet, darunter abgerundet übertragen.

2.5. Schraubdatenübertragung

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

Signal	Typ	Beschreibung
ProgramNumber (Programm-Nummer) HB = HighByte LB = LowByte	Ausgang	Programmnummer (Big Endian) Die Programmnummer wird mit High- und Lowbyte zweistellig ausgegeben (Zuerst HB dann LB) 12 -> HB= 1 LB = 2
Torque Nm (Drehmoment in Nm)	Ausgang	Das Drehmoment wird in Nm im folgenden Format mit 6 Stellen ausgegeben. Als Dezimaltrennzeichen wird ein Punkt verwendet. 0012.5 -> 12,5 Nm
Angle Deg (Drehwinkel in Grad)	Ausgang	Der Drehwinkel wird in Grad im folgenden Format mit 5 Stellen ausgegeben 03600 -> 3600 Grad
Current Amp (Strom in Ampere)	Ausgang	Der Strom wird im folgenden Format in Ampere mit 4 Stellen ausgegeben. Als Dezimaltrennzeichen wird ein Punkt verwendet. 12.3 -> 12,3 Ampere

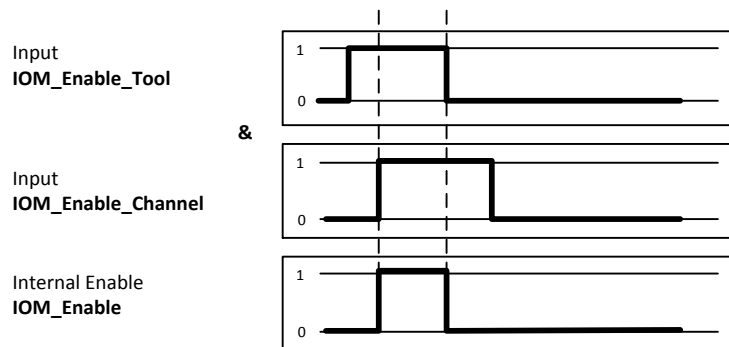
Hinweis

- Als Dezimaltrennzeichen von Float-Werten wird ein Punkt verwendet.
- Die Schraubwerte Torque und Angle sind mit einer Nachkommastelle angegeben.
- Werte mit zwei Nachkommastellen, werden ab 0,05 aufgerundet, darunter abgerundet übertragen.

2.6. Internes Enable Signal

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.
- Vor einer erneuten Verschraubung sollte vom Master zum Rücksetzen der Schraubsignale 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.

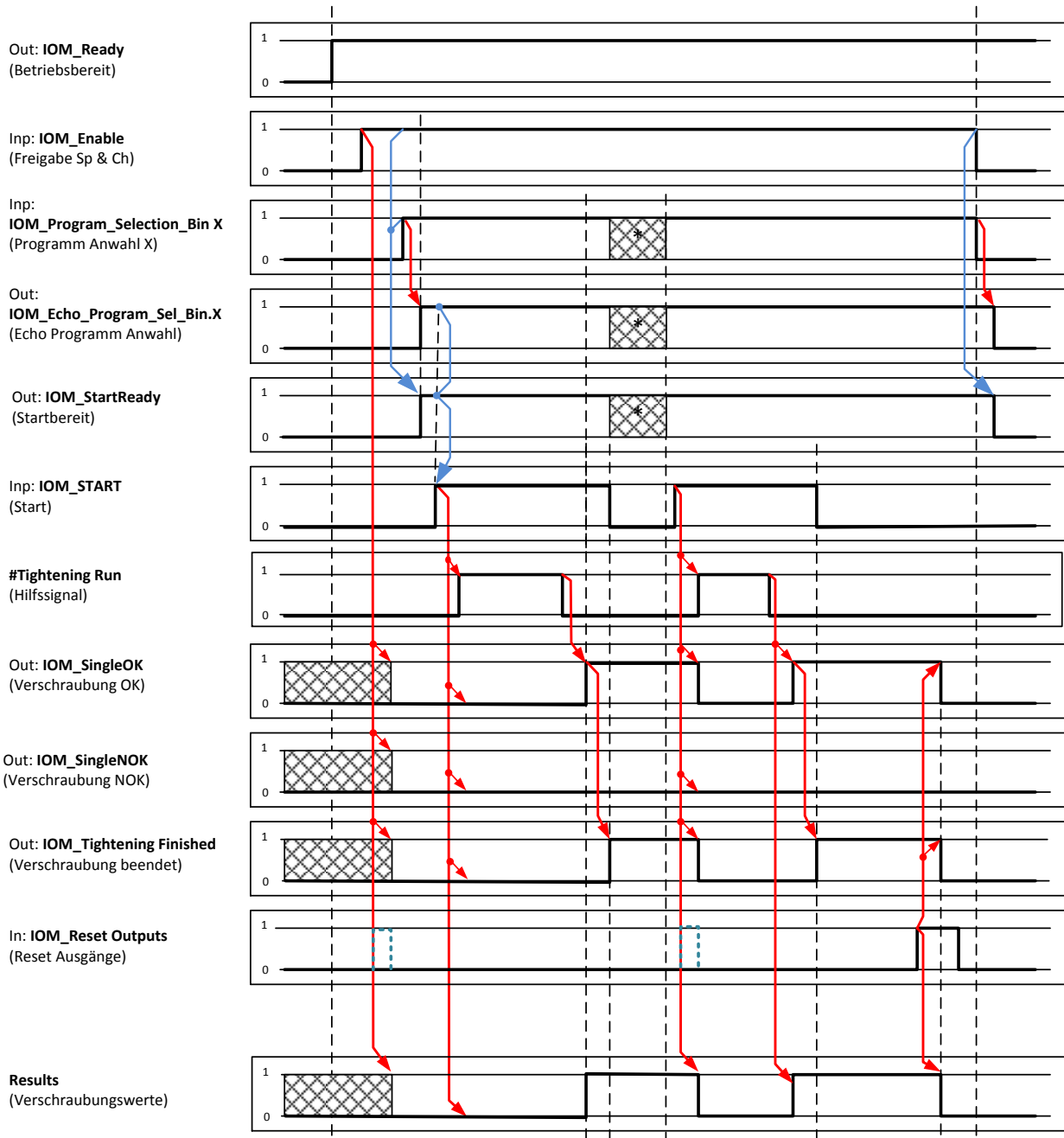
2.8. Rücksetzen der Ausgänge

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

3. Signallaufpläne

3.1. Ablauf: OK Verschraubung

Voraussetzung: Modus: „ProgramTakeOverWith = Start“

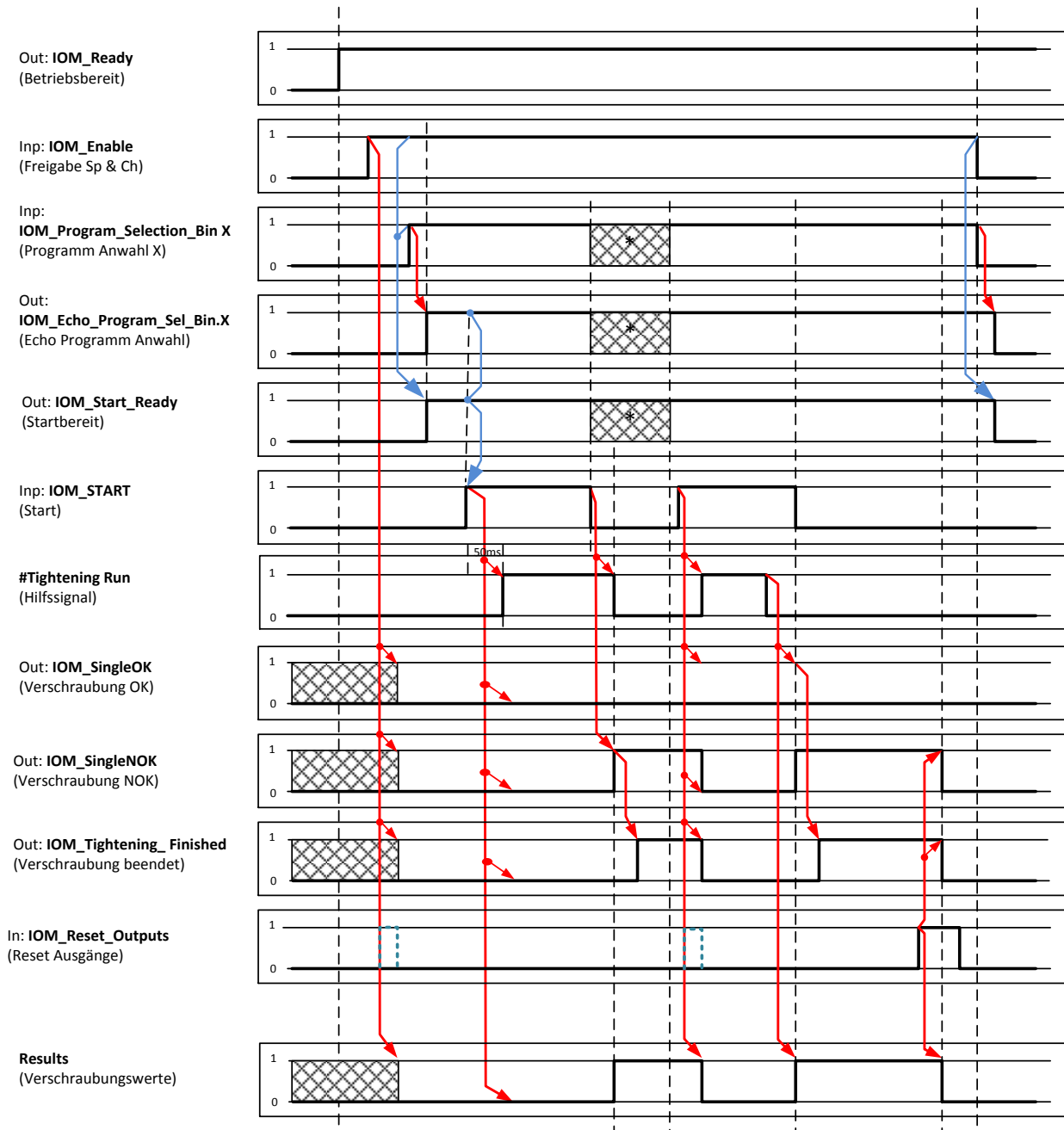


Hinweise

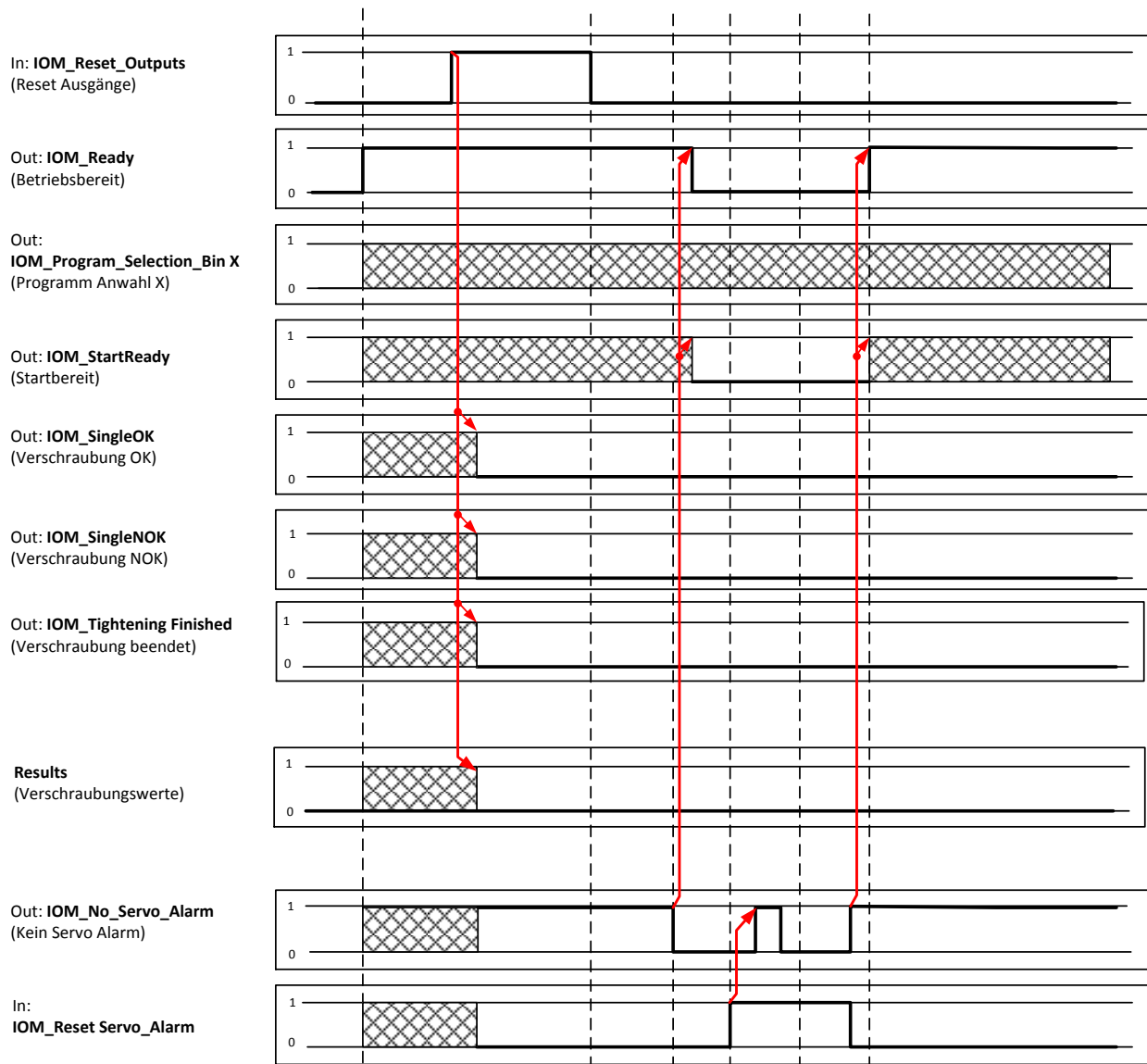
- Programmspiegelung muss beachtet werden. Das Programm muss vor dem Start gespiegelt anliegen
- Das Signal Enable setzt nicht in jedem Modus die Ausgangs-/Status-Signale zurück.

3.2. Ablauf: NOK Verschraubung

Vorraussetzung: Modus: „ProgramTakeOverWith = Start“



3.3. Ablauf: Signal Reset Outputs

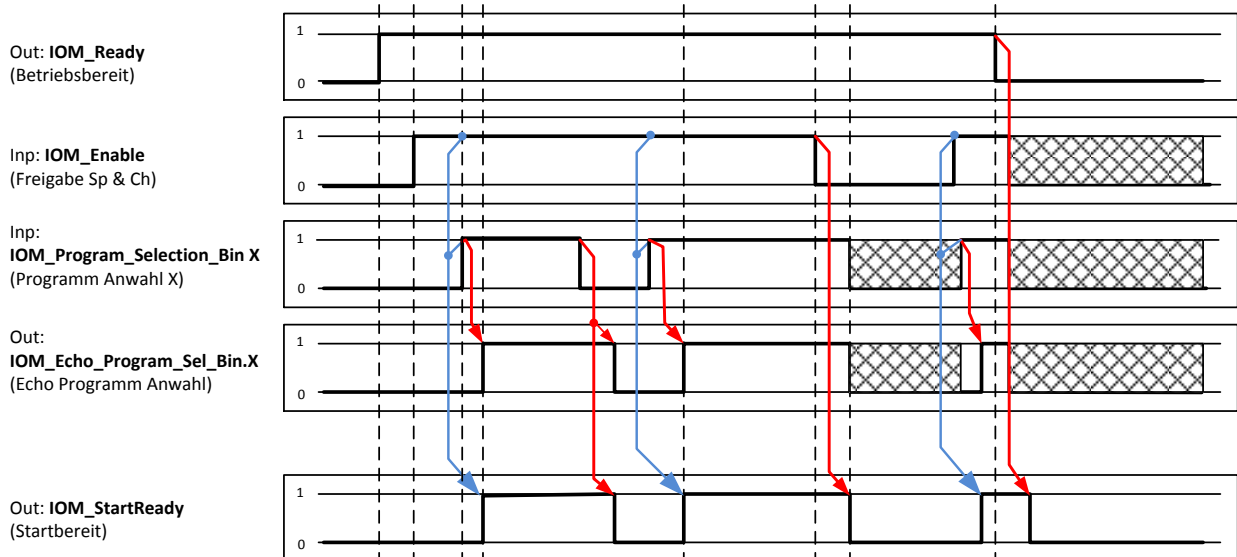


Hinweise:

- 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“

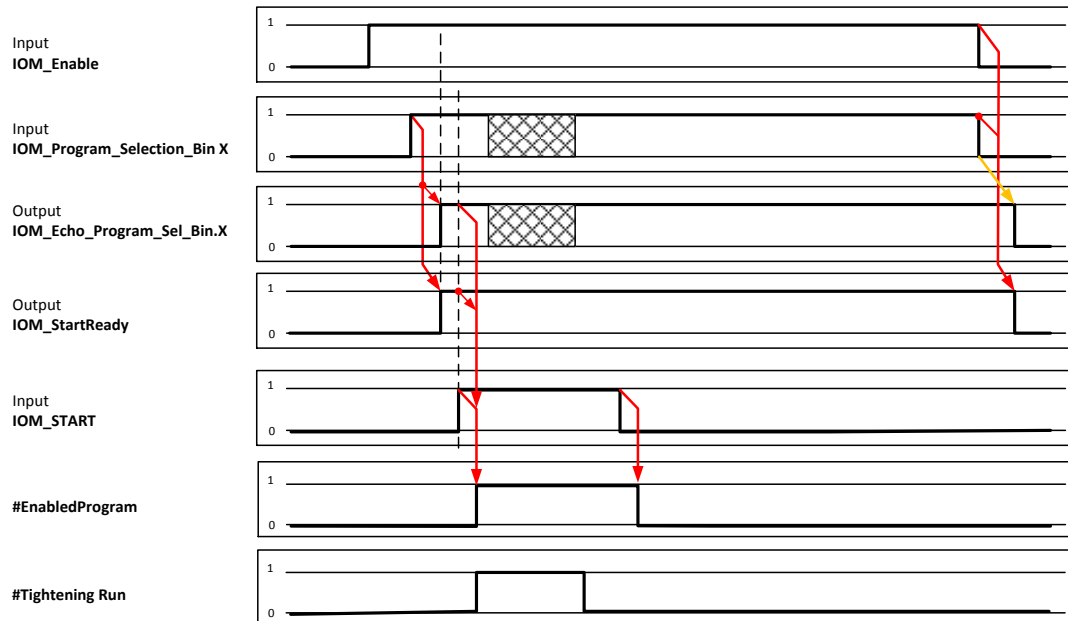


Hinweise

- 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).

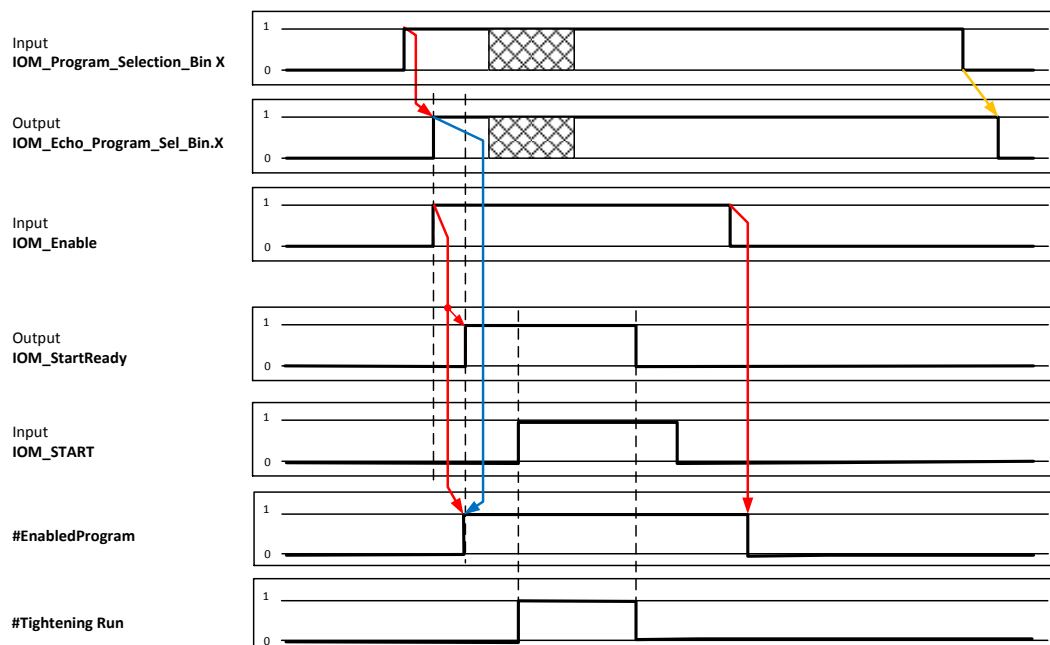
3.5. Programmübernahme mit dem START-Signal

Vorraussetzung: Modus: „ProgramTakeOverWith = Start“



3.6. Programmübernahme mit dem ENABLE-Signal

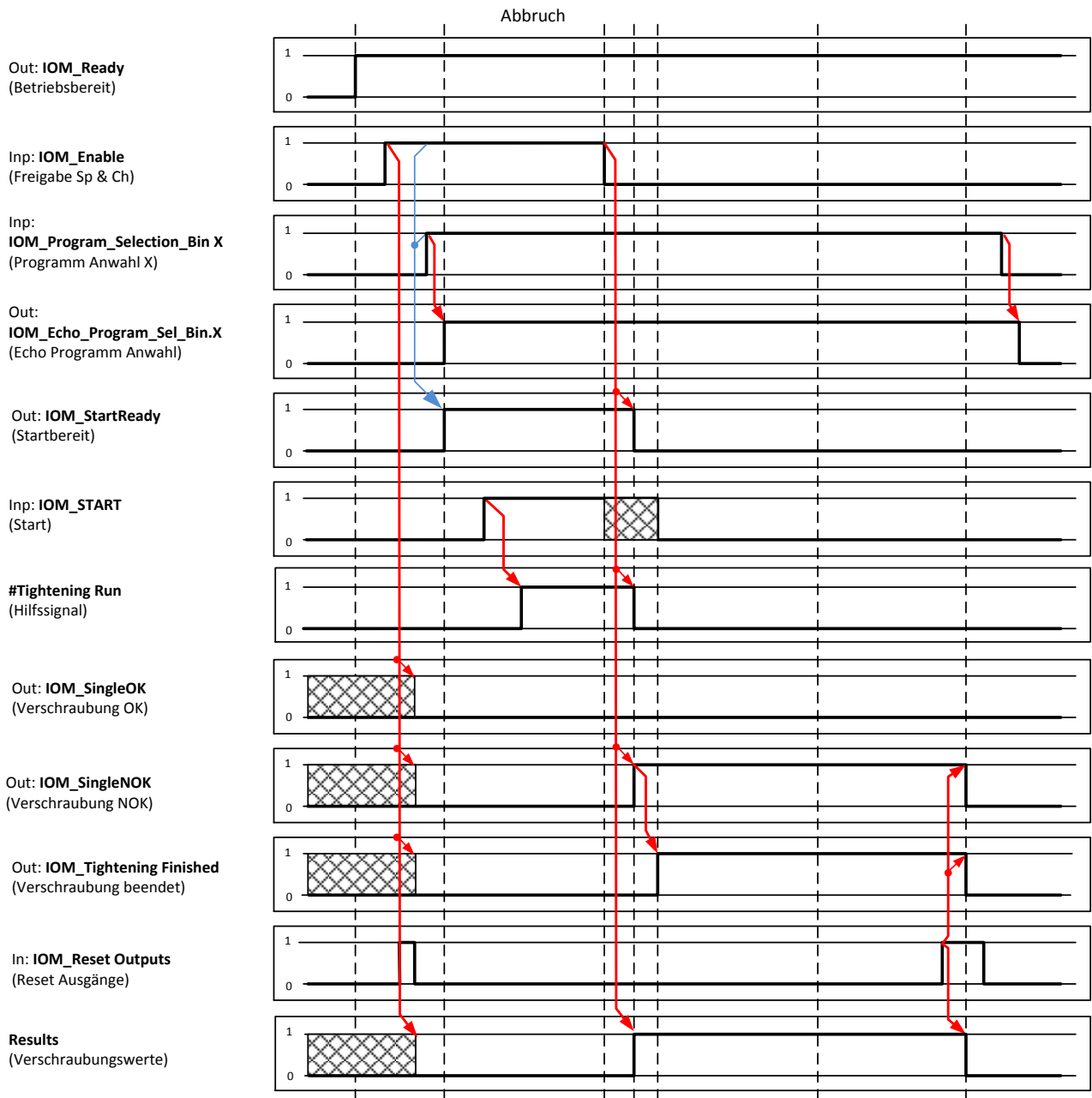
Vorraussetzung: Modus: „ProgramTakeOverWith = Enable“



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



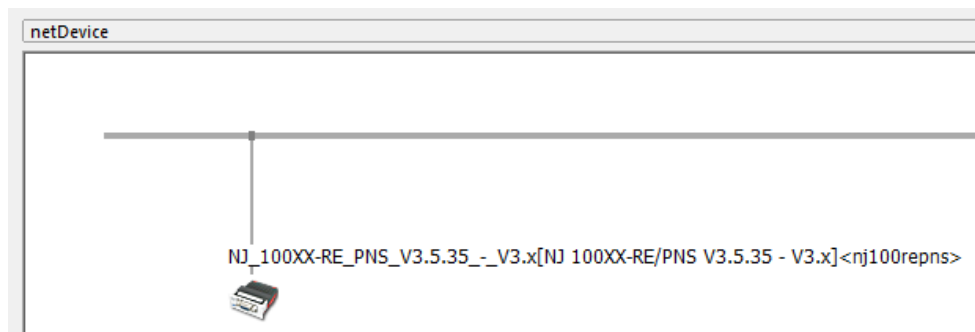
3.8.1. Hinweis zum Enable:

- Achtung über Enable kann im Versions-Stand 1.5.0 und früher nur eine Verschraubung freigegeben werden.
- Nach einer Verschraubung muss ein erneutes Enable erteilt werden.

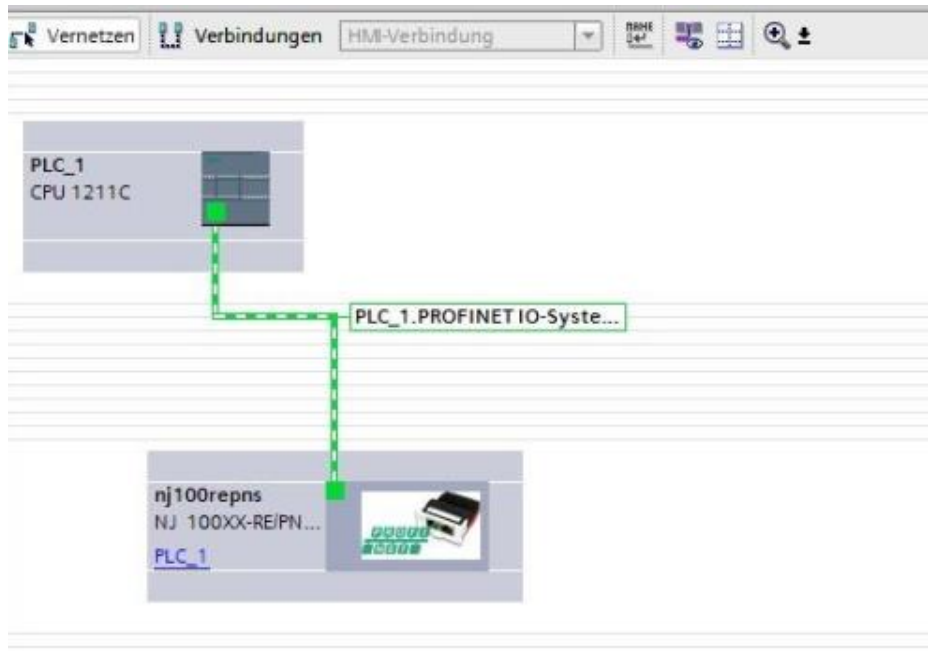
3.9. Hardwarekonfiguration (über Sycon.Net)

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.

Konfigurieren Sie ein NJ-Modul NJ_100x_RE_PNS.



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]<nj100reps>

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

Address Table

Display mode: Hexadecimal CSV Export

Inputs:

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

Outputs:

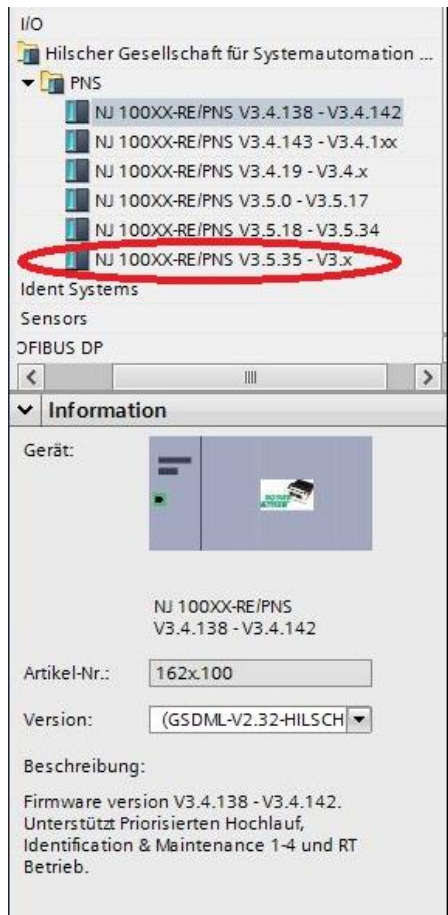
Module	Submodule	Type	Length	Address
8 Bytes Input	8 Bytes Input	QB	0x0008	0x0000
8 Bytes Input	8 Bytes Input	QB	0x0008	0x0008
8 Bytes Input	8 Bytes Input	QB	0x0008	0x0010

Aus Sicht vom Master

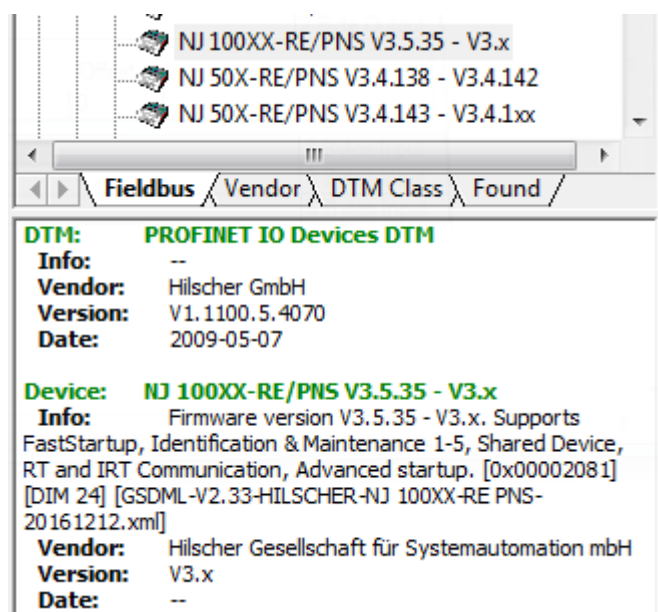
Signal Configuration								
Slot	Name	Module Type						
Slot 1 / Subslot 1	2 Byte Ausgang / 2 Byte Ausgang	2 Byte Ausgang						
<table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Offset</th> </tr> </thead> <tbody> <tr> <td>Ausgänge</td> <td>2 byte array</td> <td>0</td> </tr> </tbody> </table>			Name	Type	Offset	Ausgänge	2 byte array	0
Name	Type	Offset						
Ausgänge	2 byte array	0						
Slot 2 / Subslot 1	8 Byte Eingang / 8 Byte Eingang	8 Byte Eingang						
Slot 3 / Subslot 1	8 Byte Eingang / 8 Byte Eingang	8 Byte Eingang						
Slot 4 / Subslot 1	8 Byte Eingang / 8 Byte Eingang	8 Byte Eingang						

3.11. GSD

Bei Sycon.NET ist folgendes GSD-Modul zu verwenden.



Sycon.Net:



4. Mapping

4.1. Mapping-Datei (mapping.map)

```
// ProfiNet-Mapping
Profinet@IN.B0.0 = IOM_Enable_Tool
Profinet@IN.B0.1 = IOM_Enable_Channel
IOM_Enable_Tool & IOM_Enable_Channel = IOM_Enable

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

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

Profinet@OUT.B0.0 = IOM_Ready
Profinet@OUT.B0.1 = IOM_Start_Ready
Profinet@OUT.B0.2 = IOM_Tightening_Finished
Profinet@OUT.B0.3 = IOM_SingleOK
Profinet@OUT.B0.4 = IOM_SingleNOK

Profinet@OUT.B0.7 = IOM_No_Servo_Alarm

Profinet@OUT.B1.0 = IOM_Echo_Program_Sel_Bin@[6]

Profinet@OUT.B2 = IOM_Program_Number@[B2]
Profinet@OUT.B4 = IOM_SeparatorBlank@[B1]
Profinet@OUT.B5 = IOM_TorqueNm@[B6]
Profinet@OUT.B11 = IOM_SeparatorBlank@[B1]
Profinet@OUT.B12 = IOM_AngleDeg@[B5]
Profinet@OUT.B17 = IOM_SeparatorBlank@[B1]
Profinet@OUT.B18 = IOM_CurrentAmp@[B4]
Profinet@OUT.B22 = IOM_SeparatorBlank@[B1]
Profinet@OUT.B23 = IOM_SeparatorBlank@[B1]
```

4.2. Device-Datei (devices.map)

```
[Profinet]
Type = Profinet
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 = 6

[IOM_CurrentAmp]
Type = String
Length = 4

[IOM_SeparatorBlank]
Type = String
Length = 1

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
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.	Das Signal ist nicht als Einzelsignal verwendbar
Enable (Program) IOM_Enable	Gener. Input- Signal	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 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	<p>Programmanwahl binär</p> <p>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.</p> <p>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]</p>	<p>0 = nicht angewählt 1 = Bit angewählt</p> <p>Bit0 = binär 1 Bit1 = binär 2 Bit3 = binär 4 Bit4 = binär 8 ... usw.</p>
StartReady IOM_Start_Ready	Output	Bool 1 Bit	<p>Schraubbereit: Alle Voraussetzungen für eine Verschraubung sind erfüllt.</p> <p>Es wird gesetzt sobald eine Freigabe anliegt. Jetzt kann das Signal <IOM_Start> folgen. Die Programmnummer muss größer Null sein.</p> <p>WICHTIG: Das Signal <IOM_Start_Ready> wird automatisch auf 0 gesetzt sobald die Programm Nummer = 0 (trotz anliegender Freigabe) anliegt.</p> <p>Es wird nicht geprüft ob angewählte Programme auch vorhanden sind. Verknüpfung: <IOM_Enable> && (#ProgrammNr > 0) -> IOM_Start_Ready</p>	<p>0 = Keine Freigabe 1 = System bereit zum Verschrauben</p>
TighteningFinished IOM_Tightening_Finished	Output	Bool 1 Bit	<p>Zeigt an, dass eine Verschraubung fertig ist. (Der gestartete Verschraubungsprozess wurde durchgeführt)</p> <p>Die Verschraubungssignale OK, NOK, sowie die Werte sind erst gültig, wenn dieses Bit gesetzt ist.</p> <p>Das Bit wird zurückgenommen wenn</p> <ul style="list-style-type: none"> • das Bit „Freigabe“ von 0 auf 1 wechselt • das Bit „Start“ von 0 auf 1 wechselt <p>Wird gesetzt, wenn Ergebnisse zum Lesen verfügbar sind. Das Signal kann mit <IOM_Reset_Results> und <IOM_Reset_Outputs> zurückgesetzt werden.</p>	<p>0 = Verschraubung nicht fertig 1 = Verschraubung fertig</p>

<p>SingleOk IOM_SingleOK</p>	<p>Output Status</p>	<p>Bool 1 Bit</p>	<p>Schraubstatus: Einzel OK</p> <p>Das Schraubsystem meldet, dass der Schraubvorgang (einer Verschraubung) mit dem Ergebnis OK abgeschlossen wurde.</p> <p>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</p> <p>Gesetzt: - Wenn eine Verschraubung mit dem Status OK abgearbeitet wurde.</p> <p>Rückgesetzt: - Werkstück wird identifiziert - Beim Start des Schraubers - Bei Signal <IOM_Reset_Results> und bei <IOM_Reset_Output> - Bei Abbruch</p>	<p>0 = Kein Zustand 1 = Verschraubung OK</p>
<p>SingleNok IOM_SingleNOK</p>	<p>Output Status</p>	<p>Bool 1 Bit</p>	<p>Schraubstatus: Einzel NOK</p> <p>Das Schraubsystem meldet, dass der Schraubvorgang mit dem Ergebnis NOK abgeschlossen wurde.</p> <p>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</p> <p>Gesetzt: - Wenn eine Verschraubung mit dem Status NOK abgearbeitet wurde.</p> <p>Rückgesetzt: - Werkstück wird identifiziert - Beim Start des Schraubers - Bei Signal <IOM_Reset_Results> und bei <IOM_Reset_Output></p>	<p>0 = Kein Zustand 1 = Verschraubung NOK</p>

EchoProgramSelectionBin @[n] IOM_Echo_Program_Sel_Bin @[6]	Output	n Bool n Bit	<p>Rückgabe des aktuell anliegenden Programms an der I/O.</p> <p>Es handelt sich um ein reines ECHO, d.h. eine Spiegelung der anliegenden Programm-Signale. Es enthält KEINEN Snapshot des aktuell ausgeführten Programms.</p> <p>Programm – Nummer null ist nicht zulässig.</p> <p>[n]: stellt die Anzahl der Bits zur Verfügung</p>	<p>0 = nicht angewählt 1 = Bit angewählt</p> <p>Bit0 = binär 1 Bit1 = binär 2 Bit3 = binär 4 Bit4 = binär 8 ... usw.</p>
--	--------	-----------------	---	--

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)
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

Hinweis

Als Dezimalseparator/trennzeichen von Float-Werten wird ein Punkt verwendet.

5.1.1. Weitere konfigurierbare Signale

Diese Signale stehen in Version 1.5.0 zwar zur Verfügung, können aber über *IOM_Reset_Output* nicht zurückgesetzt werden.

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 border="0"> <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 border="0"> <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																	



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