

EUCHNER

Applikation



Beispielintegration EKS Data Service

Inhalt

1.	Zu diesem Dokument	4
1.1.	Version	4
1.2.	Gültigkeit	4
1.3.	Zielgruppe.....	4
1.4.	Ergänzende Dokumente	4
1.5.	Hinweis.....	4
2.	Verwendete Bauteile / Software	5
2.1.	EUCHNER Bauteile.....	5
2.2.	Andere Bauteile	5
2.3.	Software.....	5
3.	Funktionsbeschreibung	5
4.	Electronic-Key-Manager EKM.....	6
4.1.	EKM Server.....	6
4.2.	EKM Client	6
5.	Inbetriebnahme des EKS Data Service.....	8
5.1.	Installation des Dienstes.....	8
5.2.	Konfiguration.....	8
5.2.1.	Netzwerkkonfiguration	8
5.2.2.	Konfiguration des Dienstes	8
6.	Einbindung SPS-Bibliothek für S7-1200/1500 ins TIA Portal V14 SP1	10
6.1.	Dearchivieren der Bibliothek	10
6.2.	Integration des EKS Data Service in die SPS.....	11
6.2.1.	Bausteine ins Projekt kopieren	11
6.2.2.	Anpassen des Datentyps type_EKSDescription	12
6.2.3.	Anpassen des Datentyps type_EKSDatabase	13
6.2.4.	Hardware projektieren	13
6.2.5.	Anpassen des Datenbausteins DB_EKSData1	14
6.2.6.	Anpassen des Bausteins EKS_ReadData	15
6.2.7.	Aufruf der Funktion EKS_Communication_FC	15
6.2.8.	Aufruf des Funktionsbausteins FB_EKSDataService	16
6.2.9.	Beobachten der Daten im Datenbaustein DB_EKSData1	18
7.	Einbindung SPS-Bibliothek für S7-300/400 ins TIA Portal V14 SP1	19
7.1.	Dearchivieren der Bibliothek	19
7.2.	Integration des EKS Data Service in die SPS.....	19
7.2.1.	Bausteine ins Projekt kopieren	19
7.2.2.	Anpassen des Datentyps type_EKSDescription	20
7.2.3.	Anpassen des Datentyps type_EKSDatabase	21
7.2.4.	Erstellen des Datentyps type_EKSRead	21
7.2.5.	Hardware projektieren und Verbindungen anlegen	21
7.2.6.	Anpassen des Datenbausteins DB_EKSData1	25
7.2.7.	Aufruf des Funktionsbausteins FB_EKSReadKeyData	26
7.2.8.	Aufruf des Funktionsbausteins FB_EKSDataService	26
7.2.9.	Beobachten der Daten im Datenbaustein DB_EKSData1	28

8.	Einbindung SPS-Bibliothek für S7-300/400 in STEP7 Manager	29
8.1.	Dearchivieren der Bibliothek	29
8.2.	Integration des EKS Data Service in die SPS.....	29
8.2.1.	Bausteine ins Projekt kopieren	29
8.2.2.	Anpassen des Datentyps type_EKSDescription	30
8.2.3.	Anpassen des Datentyps type_EKSDatabase	30
8.2.4.	Hardware projektieren und Verbindungen anlegen	31
8.2.5.	Anpassen des Bausteins DB_EKSData1	36
8.2.6.	Erstellen eines EKSReadKeyData Bausteins	37
8.2.7.	Aufruf der Funktionsbausteine FB_EKSReadKeyData und FB_EKSDataService	38
9.	Wichtiger Hinweis – Bitte unbedingt sorgfältig beachten!	40

1. Zu diesem Dokument

1.1. Version

Version	Datum	Änderung/Erweiterung	Kapitel
01-10/19	22.10.2019	Erstellung	Alle

1.2. Gültigkeit

Dieses Dokument dient zur Unterstützung der Integration des EKS Data Service in Zusammenhang mit dem EKS mit PROFINET-Schnittstelle (ab Version lt. Tabelle) in das SIEMENS TIA-Portal ab Version V14 SP1.

Best. Nr.	Bezeichnung	Version
163316	EKS Data Service	V1.0.4
106305	EKS-A-IX-G01-ST02/03	V2.9.X
106306	EKS-A-IXA-G01-ST02/03/04	V2.9.X
122352	EKS-A-AIX-G18	V1.X.X
122353	EKS-A-AIXA-G18	V1.X.X

1.3. Zielgruppe

Konstrukteure, Anlagenplaner, sowie Inbetriebnahme- und Servicefachkräfte welche über Kenntnisse bei der Installation, Inbetriebnahme, Programmierung und Diagnose von speicherprogrammierbaren Steuerungen (SPS) und Bussystemen verfügen.

1.4. Ergänzende Dokumente

Die Gesamtdokumentation für diese Applikation besteht aus folgenden Dokumenten:

Dokumenttitel (Dokumentnummer)	Inhalt	
Handbuch (2528408)	EKS Data Service PC	
Handbuch (2528410)	EKS Data Service PLC	
Handbuch (2516210)	Electronic-Key-System Handbuch EKS und EKS FSA mit PROFINET IO-Schnittstelle	
Handbuch (093336)	Electronic-Key-Manager EKM	
Applikation (AP000238)	Einbindung EKS mit PROFINET Schnittstelle in das TIA Portal V13/V14/V15	
Ggf. beiliegende Datenblätter	Artikelspezifische Information zu Abweichungen oder Ergänzungen	

1.5. Hinweis

Diese Applikation basiert auf dem Handbuch des EKS mit PROFINET-Schnittstelle, der Handbücher EKS Data Service PC und PLC sowie der Applikation AP000238 "Einbindung EKS mit PROFINET Schnittstelle in das TIA Portal V13/V14/V15". Die technischen Details sowie weitere Informationen entnehmen Sie bitte den Handbüchern. Im weiteren Verlauf des Dokuments wird das EKS mit PROFINET-Schnittstelle kurz "EKS" und der EKS Data Service kurz "Dienst" genannt.

2. Verwendete Bauteile / Software

2.1. EUCHNER Bauteile

Beschreibung	Bestellnummer / Artikel
EKS mit PROFINET-Schnittstelle	106305 / EKS-A-IX-G01-ST02/03
EKS mit PROFINET-Schnittstelle FSA	106306 / EKS-A-IXA-G01-ST02/03/04



TIPP!

Weitere Informationen und Downloads zu den o.g. EUCHNER-Produkten finden Sie unter www.euchner.de. Geben Sie einfach die Bestellnummer in die Suche ein.

2.2. Andere Bauteile

Beschreibung	Bestellnummer / Artikel
SIMATIC S7-1516F-3 PN/DP	6ES7516-3FN00-0AB0
SIMATIC S7-315F-2 PN/DP	6ES7315-2FH13-0AB0
Kommunikationsprozessor CP 343-1	6GK7343-1EX30-0XE0
Desktop-PC	-

2.3. Software

Beschreibung	Version
Totally Integrated Automation Portal	Version V14 SP1 Update 7
STEP 7 Professional	Version V14 SP1 Update 7
STEP 7 Manager	STEP 7 V5.5
Electronic-Key-Manager Vollversion (Best. Nr. 093322)	Build 1.5.2.0
EKS Data Service (Best. Nr. 163316)	V 1.0.4

3. Funktionsbeschreibung

In EKS Anwendungen der SPS-Welt ist ein Datenabgleich bzw. ein Datenabruf aus einer zentralen Datenbank heute meist nicht etabliert. Das bedeutet, dass die Daten fast immer dezentral vom Schlüssel gelesen und dann individuell in der SPS weiterverarbeitet werden. Damit wird die Zugriffsinformation in diesem Fall ausschließlich über den Schlüssel transportiert. Hier besteht das Anliegen vieler EKS Betreiber einen Datenabgleich aus der SPS-Welt mit EKM Daten in der PC-Welt zu realisieren. Das Kernthema dabei ist der Wunsch, Schlüssel zentral zu sperren und ggf. weitere Daten abrufen zu können.

Der Export des Electronic-Key-Manager EKM Datenbankinhalts erfolgt in eine universell nutzbare Datei im CSV-Format in der PC-Umgebung. In dieser EKM CSV Exportdatei kann hinter der Seriennummer (KeyID) des Schlüssels ein Sperrkennzeichen ausgewertet werden, welches auf Wert ‚1‘ gesetzt ist, sobald der Schlüssel gesperrt ist.

Weitere Datenelemente, die der Schlüssel-Seriennummer zugeordnet sind, können ebenfalls abgeholt werden. Diese Datenelemente sind anwendungsspezifisch unterschiedlich.

Wird ein Schlüssel platziert, wird eine Anfrage vom EKS Data Service PLC an den EKS Data Service PC geschickt. Der EKS Data Service PC sucht anhand der KeyID den Eintrag in der EKM CSV Exportdatei und schickt die Daten an den EKS Data Service PLC zurück. Dort stehen nun die Daten dem Benutzer für die weitere Verarbeitung zu Verfügung. Weiter werden die angefragten Daten in einen Notlaufspeicher abgelegt. Sollte die Verbindung zum PC abbrechen, werden die Daten von zuvor platzierten Schlüsseln aus dem Speicher der Notlaufstrategie geladen. In dem Speicher der Notlaufstrategie stehen die letzten maximal 100 Schlüsseldaten (Größe des Notlaufspeichers kann angepasst werden).

Hinweis zur Notlaufstrategie: Sollte der Speicher der Notlaufstrategie voll sein, und ein neuer Schlüssel wird angefragt, so wird der am längsten nicht abgefragte Schlüssel ersetzt. Da die Einträge der Notlaufstrategie zyklisch aktualisiert werden, wird nicht zwingend der erste Schlüssel, der platziert wurde, ersetzt.

Es gibt 2 Notlaufstrategien. Bei Notlaufstrategie 1 kann der EKS Data Service PC die original EKM CSV Exportdatei nicht erreichen und verwendet die lokale Backupdatei der EKM CSV Exportdatei. Bei Notlaufstrategie 2 kann der EKS Data Service PC keine Daten zur Verfügung stellen, weil die Kommunikation gestört ist und es werden die Daten des internen SPS Datenbausteins verwendet.

4. Electronic-Key-Manager EKM

4.1. EKM Server

Starten Sie den EKM Server und öffnen Sie die Einstellungen. Unter dem Reiter *Export* aktivieren Sie die Checkbox *Start export after any change*. Anschließend legen Sie unter *Export-File* fest, wo und unter welchem Namen die csv-Datei abgelegt werden soll. Die csv-Datei muss für den Dienst später erreichbar sein.

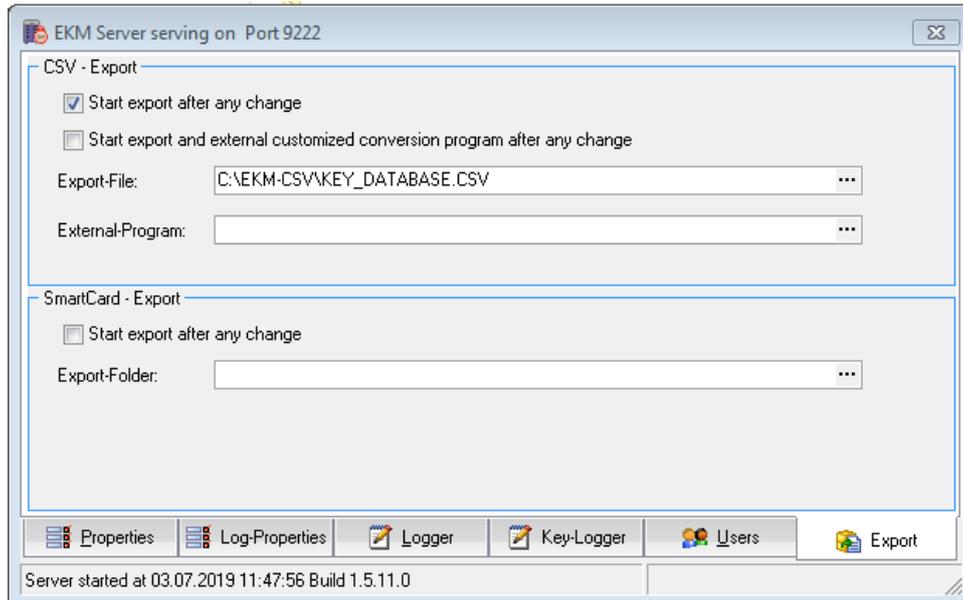


Bild 1: EKM Server Konfiguration

4.2. EKM Client

Voraussetzung um den EKS Data Service nutzen zu können ist eine vollständige EKM Anwendung. Dies beinhaltet eine Eingabemaske inklusive Datenbank. In der Applikation wird nachfolgendes Beispiel einer EKM Eingabemaske inklusive Datenbank verwendet.

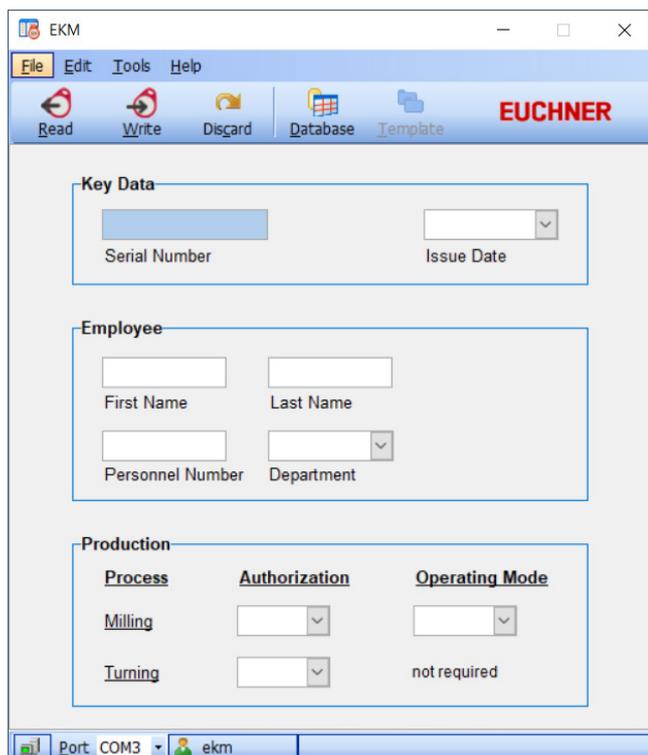


Bild 2: EKM Client (Beispiel)

Database Designer

Fiel..	OnKey	Fieldname	Type	StartByte	Length	BitNo	DisplayT...	Unique	Template
1	<input type="checkbox"/>	OM	Word (0 .. 65535)				Dez	<input type="checkbox"/>	<input type="checkbox"/>
2	<input type="checkbox"/>	Department	String		2			<input type="checkbox"/>	<input type="checkbox"/>
3	<input type="checkbox"/>	Last_Name	String		40			<input type="checkbox"/>	<input type="checkbox"/>
4	<input type="checkbox"/>	First_Name	String		40			<input type="checkbox"/>	<input type="checkbox"/>
5	<input type="checkbox"/>	Level_Milling	Byte (0 .. 255)				Dez	<input type="checkbox"/>	<input type="checkbox"/>
6	<input type="checkbox"/>	Level_Turning	Byte (0 .. 255)				Dez	<input type="checkbox"/>	<input type="checkbox"/>
7	<input type="checkbox"/>	Personnel_No	String		2			<input type="checkbox"/>	<input type="checkbox"/>
8	<input type="checkbox"/>	Issue_Date	Date					<input type="checkbox"/>	<input type="checkbox"/>

Bild 3: EKM Datenbank-Designer (Beispiel)

5. Inbetriebnahme des EKS Data Service

5.1. Installation des Dienstes

Die Vorgehensweise zur Installation des Dienstes entnehmen Sie bitte dem Handbuch EKS Data Service PC.

5.2. Konfiguration

5.2.1. Netzwerkkonfiguration

Der Server, auf dem der Dienst installiert wird, muss über eine Netzwerkverbindung mit dem Maschinennetzwerk verbunden sein.

Das Web-Interface des Dienstes zur Konfiguration wird standardmäßig über folgende Adresse auf dem Server aufgerufen:
<http://localhost:65080/EKSDataService>



HINWEIS!

TCP-Port 102: RFC1006-Verbindung der SIMATIC Steuerungen mit dem Dienst. Dieser Port kann nicht geändert werden und darf deshalb auch nicht durch andere Softwareprodukte reserviert sein. Siemens Software Produkte, wie z.B. TIA-Portal belegen standardmäßig mit einem Dienst den Port 102. Falls eine parallele Installation nicht verhindert werden kann, dann muss der SIMATIC Dienst S7DOS Help Service (s7oiehsx64.exe) beendet und deaktiviert werden.

5.2.2. Konfiguration des Dienstes

Um die Konfiguration des Dienstes vornehmen zu können, müssen Sie sich einloggen. Der Standardbenutzer besitzt folgende Login-Daten:

User: *admin*

Password: *admin*

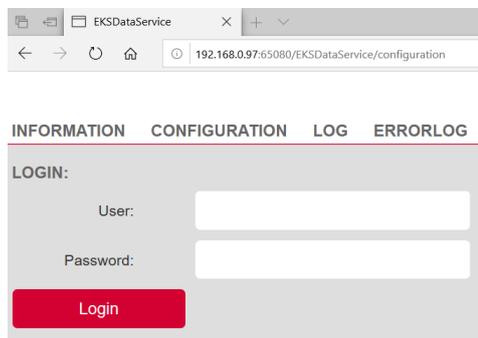


Bild 4: Login

Nachdem Login beginnt die einmalige Konfiguration des Dienstes. Im nachfolgenden Beispiel werden nicht alle Werte parametrierbar. Im Handbuch sind alle parametrierbaren Werte erklärt und können individuell eingestellt werden.

1. Geben Sie den Pfad und den Dateinamen an unter dem die csv-Datei zu finden ist. Klicken Sie anschließend auf *Read and check now...* . Sollte die csv-Datei gelesen werden können, wird Ihnen eine positive Rückmeldung über *File-Status* ausgegeben. Sollte eine negative Rückmeldung ausgegeben werden, kontrollieren Sie bitte die Pfadangabe und den Dateinamen und prüfen Sie, ob die Datei vom Server aus erreichbar ist.

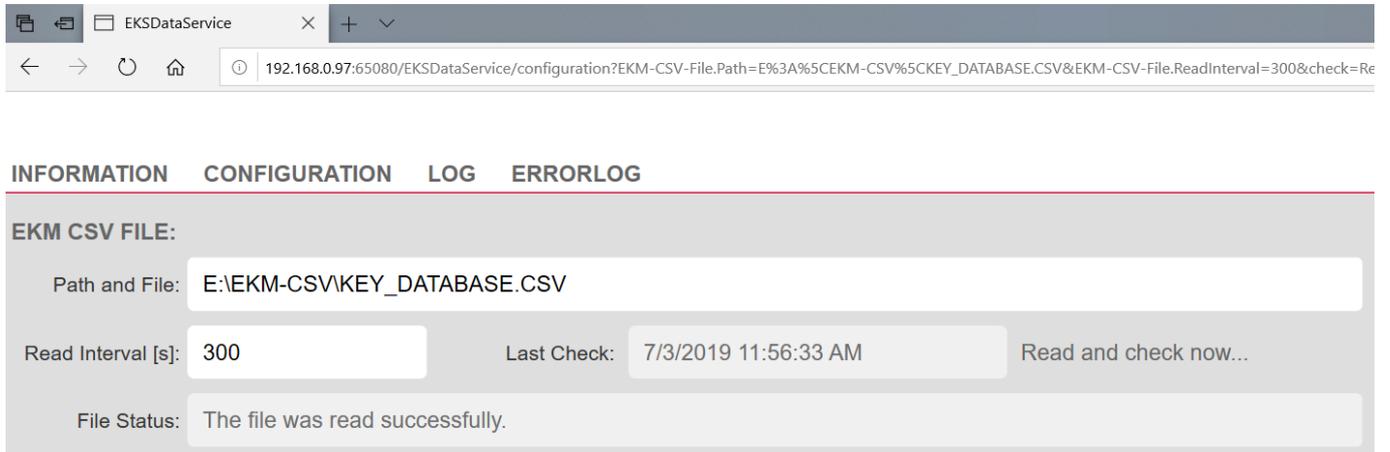


Bild 5: Pfadangabe csv-Datei und erstmaliges Einlesen

2. Nachdem die csv-Datei korrekt eingelesen wurde, werden automatisch unter *EKM PLC DATA TYPES* die Namen der Spalten aus der csv-Datei eingetragen. Sie haben die Möglichkeit anzugeben, welche Daten Sie zur SPS übertragen wollen. In diesem Beispiel werden alle Daten zur SPS übertragen.

Die Spalten *KEYID* und *LOCKED* sind Standardspalten, die jede EKM Datenbank enthält. Dort sind die Werte bereits vorgegeben. Der *PLC-Index* wird für jede Spalte um den Wert 1 erhöht. Anschließend werden die zugehörigen Datentypen (*PLC-Type*) eingetragen. Die Zuordnung der EKM-Datentypen mit den SPS-Datentypen sind im Handbuch EKS Data Service PC beschrieben.

EKM PLC DATA TYPES:										
Name:	KEYID	LOCKED	OM	Department	Last_Name	First_Name	Level_Milling	Level_Turning	Personnel_No	Issue_Date
PLC-Index:	1	2	3	4	5	6	7	8	9	10
PLC-Type:	String[16]	Bool	Word	String[2]	String[40]	String[40]	Byte	Byte	String[2]	Date

Bild 6: Zuordnung Datentypen

3. Speichern Sie die Konfiguration indem Sie auf *Apply* klicken.

6. Einbindung SPS-Bibliothek für S7-1200/1500 ins TIA Portal V14 SP1

Für die S7-1200/1500 benötigen Sie folgende Bibliotheken:

EKS_Data_Service_PLC_Library_TIA_1200_1500_20190227

Library_EKS_Data_Service-TIAV14SP1_PLC-1200_1500_YYYYMMDD

6.1. Dearchivieren der Bibliothek

1. Wechseln Sie zur Ansicht *Task Card* (Shortcut: *Strg+3*) und wählen Sie *Bibliotheken* aus.
2. Öffnen Sie mit einem Rechtsklick das Kontextmenü im Bereich *Globale Bibliotheken* und wählen Sie *Bibliothek dearchivieren...* Wählen Sie den Ordner mit der Bibliothek aus und dearchivieren Sie diesen in den gewünschten Zielordner. Beim Dearchivieren der Bibliothek mit TIA-Portal V15 oder höher erfolgt nach dem Dearchivieren eine Hochrüstung der Bibliothek, da diese mit TIA-Portal V14 SP1 erstellt wurde.

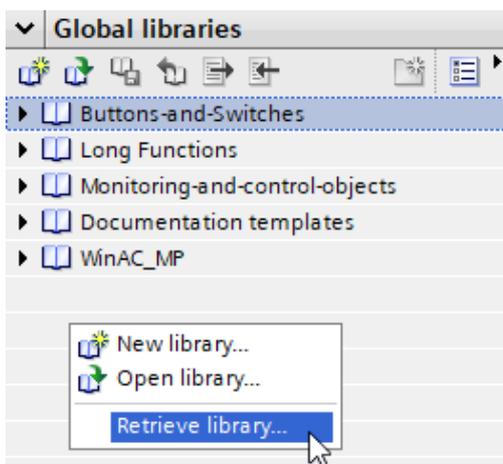


Bild 7: Bibliothek dearchivieren

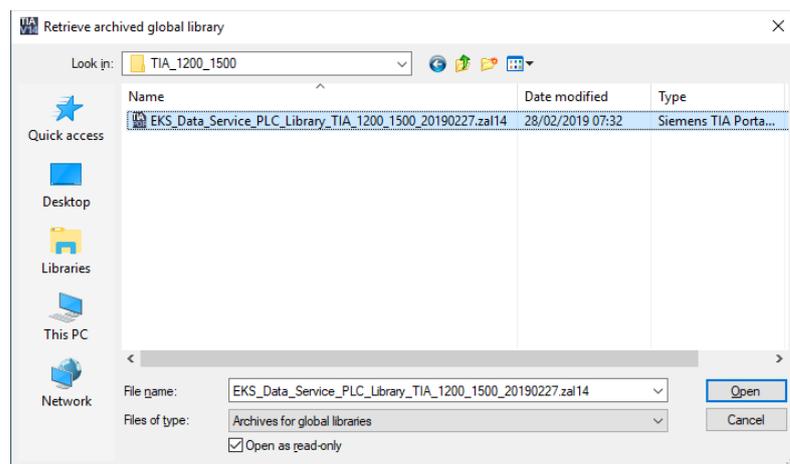


Bild 8: Bibliothek auswählen



TIPP!

Wenn die Bibliothek schon dearchiviert wurde, wählen Sie *Bibliothek öffnen...*, um die Bibliothek zum Projekt hinzuzufügen.

6.2. Integration des EKS Data Service in die SPS

6.2.1. Bausteine ins Projekt kopieren

1. Öffnen Sie die *Bibliothek* und kopieren Sie die Bausteine entsprechend Ihrer Steuerungsumgebung in die Ordner der *Projektnavigation*.

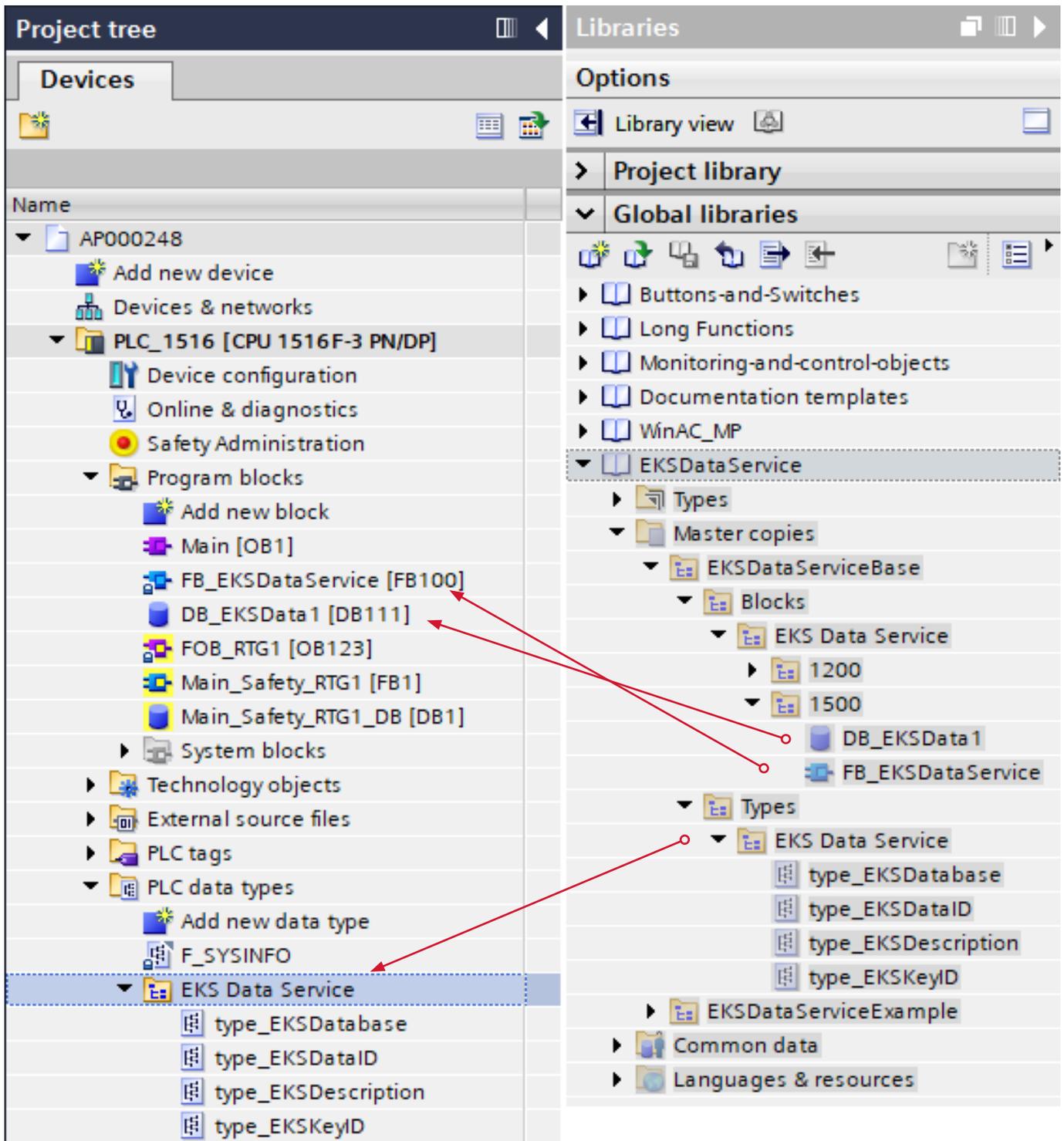


Bild 9: Bausteine ins Projekt kopieren

- Öffnen Sie die Bibliothek `Library_EKS_Data_Service-TIAV14SP1_PLC-1200_1500_YYYYMMDD` und kopieren Sie die Bausteine in die Ordner der *Projektnavigation*. Diese Bibliothek dient zum Auslesen der EKS Schlüssel. Es ist nicht zwingend notwendig diese Bibliothek zu verwenden.

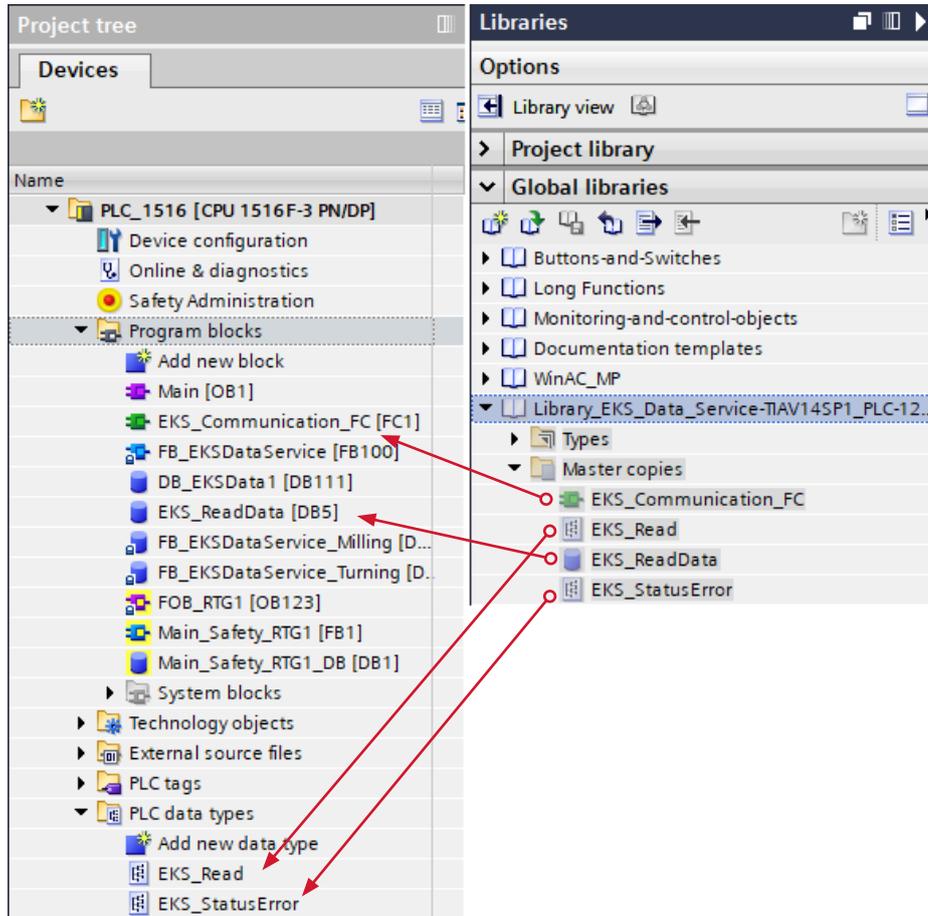


Bild 10: Bausteine ins Projekt kopieren

6.2.2. Anpassen des Datentyps `type_EKSDescription`

Öffnen Sie den Datentyp `type_EKSDescription`. Bilden Sie die im Dienst konfigurierten `EKM PLC DATA TYPES Name` und `PLC-Index` ab.

type_EKSDescription							
	Name	Data type	Default value	Accessible f...	Writa...	Visible in ...	Setpoint
1	KeyID	Int	1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2	LOCKED	Int	2	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3	OM	Int	3	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4	Department	Int	4	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5	Last_Name	Int	5	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6	First_Name	Int	6	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
7	Level_Milling	Int	7	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
8	Level_Turning	Int	8	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
9	Personnel_No	Int	9	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
10	Date	Int	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Bild 11: Anpassen des Datentyps `type_EKSDescription`

6.2.3. Anpassen des Datentyps `type_EKSDatabase`

Öffnen Sie den Datentyp `type_EKSDatabase`. Bilden Sie die im Dienst konfigurierten `EKM PLC DATA TYPES Name` und `PLC-Type` ab. Achten Sie dabei auf die Zuordnung der EKM Datentypen und SPS Datentypen, wie im Handbuch EKS Data Service PC beschrieben.

type_EKSDatabase							
	Name	Data type	Default value	Accessible f...	Writa...	Visible in ...	Setpoint
1	KeyID	String[16]	"	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2	LOCKED	Bool	false	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3	OM	Word	16#0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4	Department	String[2]	"	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5	Last_Name	String[40]	"	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6	First_Name	String[40]	"	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
7	Level_Milling	Byte	16#0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
8	Level_Turning	Byte	16#0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
9	Personnel_No	String[2]	"	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
10	Issue_Date	Date	D#1990-01-01	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Bild 12: Anpassen des Datentyps `type_EKSDatabase`

6.2.4. Hardware projektieren

In diesem Beispiel werden zwei EKS Geräte mit PROFINET Schnittstelle projiziert. In dieser Applikation werden alle Schlüsselinformationen aus der csv-Datei übertragen. Aus diesem Grund muss nur die Seriennummer vom EKS Schlüssel gelesen werden. Sie können dafür die kleinsten Submodule (Lesen/Schreiben) im EKS (*Gerätesicht*) projektieren (siehe Applikation AP000238). Somit ist es möglich Speicher in der Steuerung einzusparen. Für dieses Beispiel muss im Submodul Lesen der EKS, die Startadresse 116 parametrieren werden.

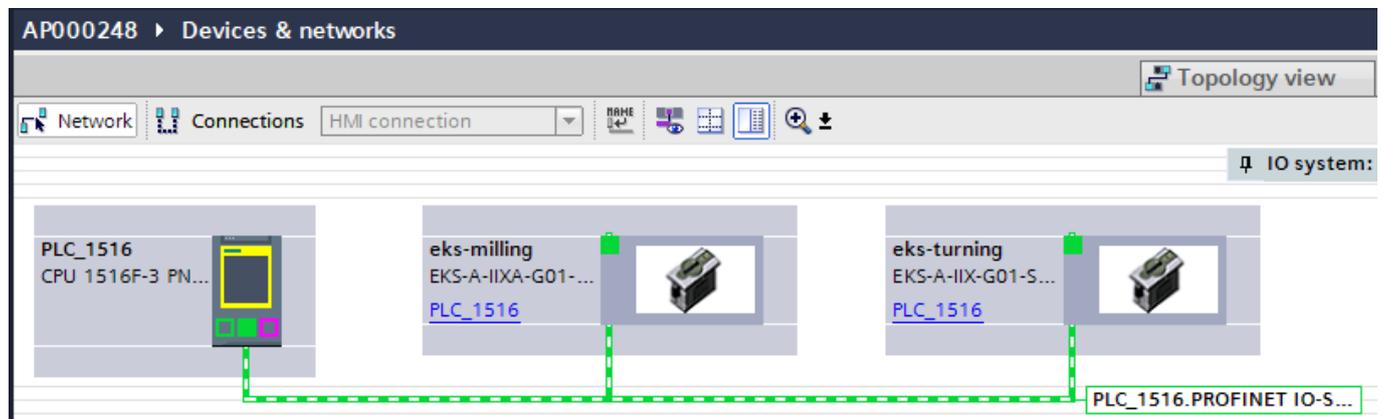


Bild 13: TIA Portal Netzsicht



HINWEIS!

Die Parametrierung der Submodule ist im Handbuch EKS und EKS FSA mit PROFINET IO-Schnittstelle beschrieben.

6.2.5. Anpassen des Datenbausteins DB_EKSData1

Im Datenbaustein DB_EKSData1 werden die notwendigen Daten für den FB_EKSDataService deklariert

Variablendeklaration DB_EKSData1

Variable	Datentyp	Beschreibung
Data_Key_Reader_Milling Data_Key_Reader_Turning	type_EKSDatabase	In dieser Variablen werden die Daten des aktuell im EKS platzierten Schlüssels abgelegt
Description	type_EKSDescription	In dieser Variablen wird die Struktur des Datentyp type_EKSDatabase dem FB bereitgestellt
ID	Array[0..100] of 'type_EKSDataID'	Die ID Variable wird benötigt, um zusätzliche Informationen wie den Zeitstempel für das Platzieren des Schlüssels oder den Zeitstempel der letzten Aktualisierung, die KeyID und die Checksumme des EKS Data Service zu speichern. Diese zusätzlichen Informationen werden für jeden platzierten Schlüssel eingetragen und im DB abgelegt. Dieser Datentyp darf nicht verändert werden und wird als Array in den DB abgelegt. Für jede Datenspeicherung (Data Array) benötigt man ein ID Array mit der gleichen Größe. Da die Daten auch nach Spannungsausfall/Neustart benötigt werden, ist es notwendig diese als remanent zu markieren.
Data	Array[0..100] of 'type_EKSDatabase'	Der gleiche Datentyp für Daten der platzierten Schlüssel, wird auch für die Speicherung der Daten für die Notlaufstrategie verwendet. Alle Daten der platzierten Schlüssel werden in das Data Array eingetragen. Die Daten werden erst überschrieben, wenn die Anzahl der platzierten Schlüssel größer ist, als die Anzahl der Array Einträge. Sollten die Daten überschrieben werden, dann wird immer der älteste Schlüsseleintrag überschrieben. Wird ein Schlüssel bei der Anfrage beim EKS Data Service PC nicht gelistet, dann wird der Schlüssel auch im Data Array gelöscht. Die Instanzen des FB_EKSDataService können alle auf die gleichen Daten (Description, ID, Data) im DB_EKSData1 zugreifen. Sollten die Daten auch nach einem Spannungsausfall zu Verfügung stehen, ist es notwendig diese als remanent zu markieren.
EKS_Data_Temp_Milling EKS_Data_Temp_Turning	type_EKSDatabase	Der FB_EKSDataService benötigt einen Zwischenspeicher außerhalb des Bausteins vom Typ type_EKSDatabase. Der Zwischenspeicher wird für Kopieraufgaben verwendet.
EKS_StatusError_Milling EKS_StatusError_Turning	EKS_StatusError	Für diese Applikation wurde ein Datentyp EKS_StatusError angelegt. Die darin enthaltenen Fehler und Daten sind im Handbuch EKS Data Service PLC beschrieben.

DB_EKSData1									
	Name	Data type	Start value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint	
1	Static			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
2	Data_Key_Reader_Milling	*type_EKSDatabase*		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
3	Data_Key_Reader_Turning	*type_EKSDatabase*		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
4	Description	*type_EKSDescription*		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
5	ID	Array[0..100] of *type_EKSDataID*		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
6	Data	Array[0..100] of *type_EKSDataba...		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
7	EKS_Data_Temp_Milling	*type_EKSDatabase*		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
8	EKS_Data_Temp_Turning	*type_EKSDatabase*		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
9	EKS_StatusError_Milling	*EKS_StatusError*		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
10	EKS_StatusError_Turning	*EKS_StatusError*		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

Bild 14: Deklaration DB_EKSData1

6.2.6. Anpassen des Bausteins *EKS_ReadData*

Passen Sie den Datenbaustein *EKS_ReadData* an. Für jedes EKS muss eine Variable vom Datentyp *EKS_Read* (gesamte Daten aus dem Eingangsbereich der Steuerung einlesen), *Word* (Fehleranzeige) und *type_EKSKeyID* (separieren der Schlüsselseriennummer) deklariert werden.

EKS_ReadData		
	Name	Data type
1	▼ Static	
2	▶ ReadData_Milling	"EKS_Read"
3	Error_Read_Milling	Word
4	▶ ReadData_Turning	"EKS_Read"
5	Error_Read_Turning	Word
6	▶ EKSKeyID_Milling	"type_EKSKeyID"
7	▶ EKSKeyID_Turning	"type_EKSKeyID"

Bild 15: Parametrieren des Datenbausteins *EKS_ReadData*

6.2.7. Aufruf der Funktion *EKS_Communication_FC*

Rufen Sie die Funktion *EKS_Communication_FC* in einem von Ihnen erstellten FB (hier: *FB_EKS_Euchner*) auf. Dieser Baustein muss für jedes EKS aufgerufen und parametrieren werden. Weisen Sie die Variablen den Ein- und Ausgängen zu. Am Eingang *EKS_PN_ReadModule* geben Sie die HW-Kennung vom Modul Lesen des entsprechenden EKS an. Die Daten aus dem Eingangsbereich werden in den Datenbaustein *EKS_ReadData* des jeweiligen EKS kopiert.

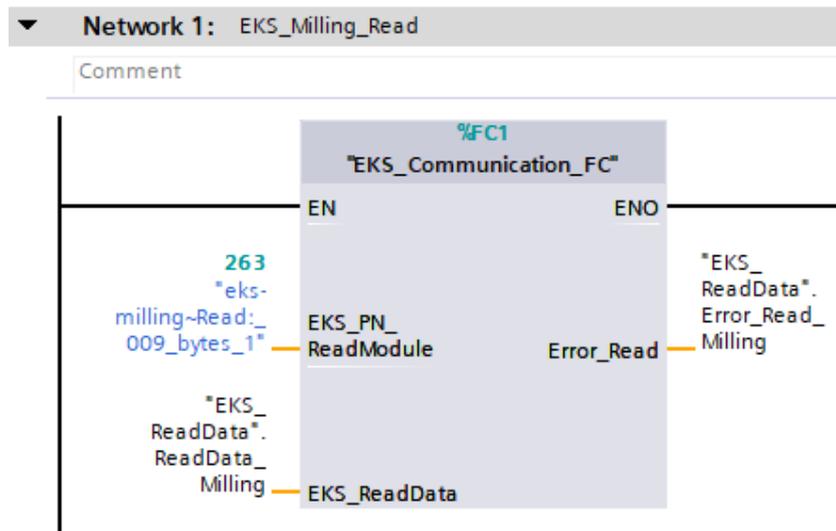


Bild 16: Aufruf des *EKS_Communication_FC* (Bsp.: Milling)

Anschließend muss die Schlüsselseriennummer noch von den EKS Empfangsdaten (*Receive_Data*) in die Seriennummer Variable (*type_EKSKeyID*) kopiert werden. Dies kann mit zwei Move Befehlen durchgeführt werden.

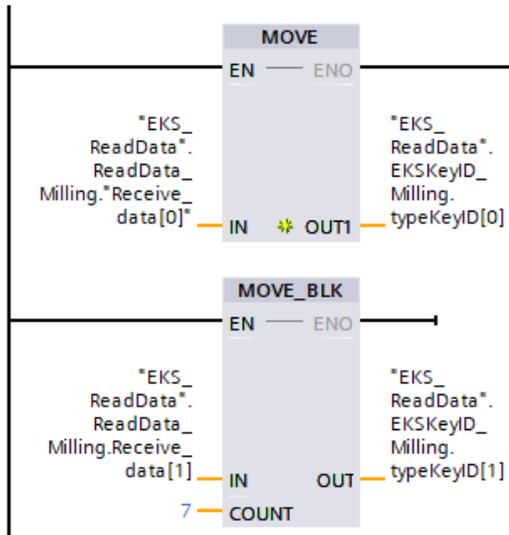


Bild 17: Seriennummer in die Variable *type_EKSKeyID* kopieren



HINWEIS!

Es ist zu beachten, dass aufgrund einer 16 Bit Grenzvorgabe seitens Siemens das erste Byte des Schlüsselspeichers innerhalb des DB (*EKS_ReadData*) NICHT im Array *ReceiveData* liegt, sondern als gesondertes Byte aufgelistet ist.

6.2.8. Aufruf des Funktionsbausteins *FB_EKSDataService*

Rufen Sie den Funktionsbaustein *FB_EKSDataService* im Hauptprogramm *Main (OB1)* auf. Dieser Baustein muss für jedes EKS aufgerufen und parametrisiert werden.



HINWEIS!

Die Eingänge und Ausgänge des *FB_EKSDataService* sind im Handbuch *EKS Data Service PLC* detailliert beschrieben.

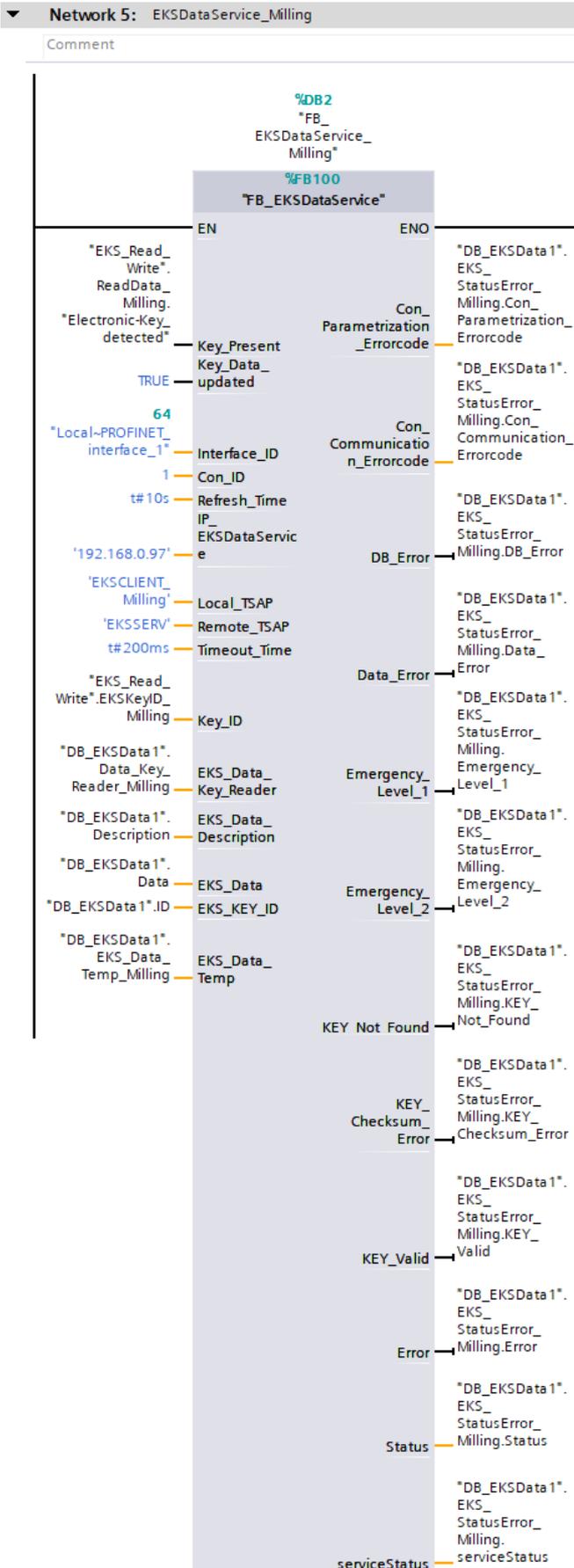


Bild 18: Aufruf eines `FB_EKSDataService` (Bsp.: `Milling`)

6.2.9. Beobachten der Daten im Datenbaustein *DB_EKSData1*

Nachdem alle Bausteine und UDTs parametrierung wurden, laden Sie Ihr Programm in die Steuerung. Im Dienst auf dem PC ist unter dem Reiter *LOG* zusehen, dass die Verbindung zur Steuerung aufgebaut wurde. Programmieren Sie im EKM ein paar Schlüssel und platzieren Sie diese an Ihre Anlage. Sobald Sie im TIA Portal den Baustein *DB_EKSData1* online betrachten, sehen Sie die abgeholten Daten aus der csv-Datei.

DB_EKSData1								
	Name	Data type	Start value	Monitor value	Retain	Accessible f...	Writa...	Visible in ...
1	Static				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2	Data_Key_Reader_Milling	*type_EKSDatabase*			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
3	KeyID	String[16]	"	'02877825FF001032'	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4	LOCKED	Bool	false	FALSE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5	OM	Word	16#0	16#FOOF	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6	Department	String[2]	"	'QS'	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
7	Last_Name	String[40]	"	'Doe'	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
8	First_Name	String[40]	"	'John'	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
9	Level_Milling	Byte	16#0	16#01	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
10	Level_Turning	Byte	16#0	16#03	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
11	Personnel_No	String[2]	"	'10'	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
12	Issue_Date	Date	D#1990-01-01	D#2019-07-04	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
13	Data_Key_Reader_Turning	*type_EKSDatabase*			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
14	Description	*type_EKSDescription*			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
15	ID	Array[0..100] of *type_EKSDataID*			<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
16	Data	Array[0..100] of *type_EKSDataBa...			<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
17	EKS_Data_Temp_Milling	*type_EKSDatabase*			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
18	EKS_Data_Temp_Turning	*type_EKSDatabase*			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
19	EKS_StatusError_Milling	*EKS_StatusError*			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
20	EKS_StatusError_Turning	*EKS_StatusError*			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Bild 19: Daten aus csv-Datei im *DB_EKSData1*

7. Einbindung SPS-Bibliothek für S7-300/400 ins TIA Portal V14 SP1

Für die S7-300/400 benötigen Sie folgende Bibliotheken:

EKS_Data_Service_PLC_Library_TIA_300_400_YYYYMMDD

Library_EKS_Data_Service-TIAV14SP1_PLC-300_400_YYYYMMDD

7.1. Dearchivieren der Bibliothek

Dearchivieren Sie die Bibliotheken wie in Abschnitt 6.1 beschrieben.

7.2. Integration des EKS Data Service in die SPS

7.2.1. Bausteine ins Projekt kopieren

- Öffnen Sie die *Bibliothek* EKS_Data_Service_PLC_Library_TIA_300_400_YYYYMMDD und kopieren Sie die Bausteine entsprechend Ihrer Steuerungsumgebung in die Ordner der *Projektnavigation*. In diesem Beispiel werden nicht alle Bausteine aus der Bibliothek verwendet.

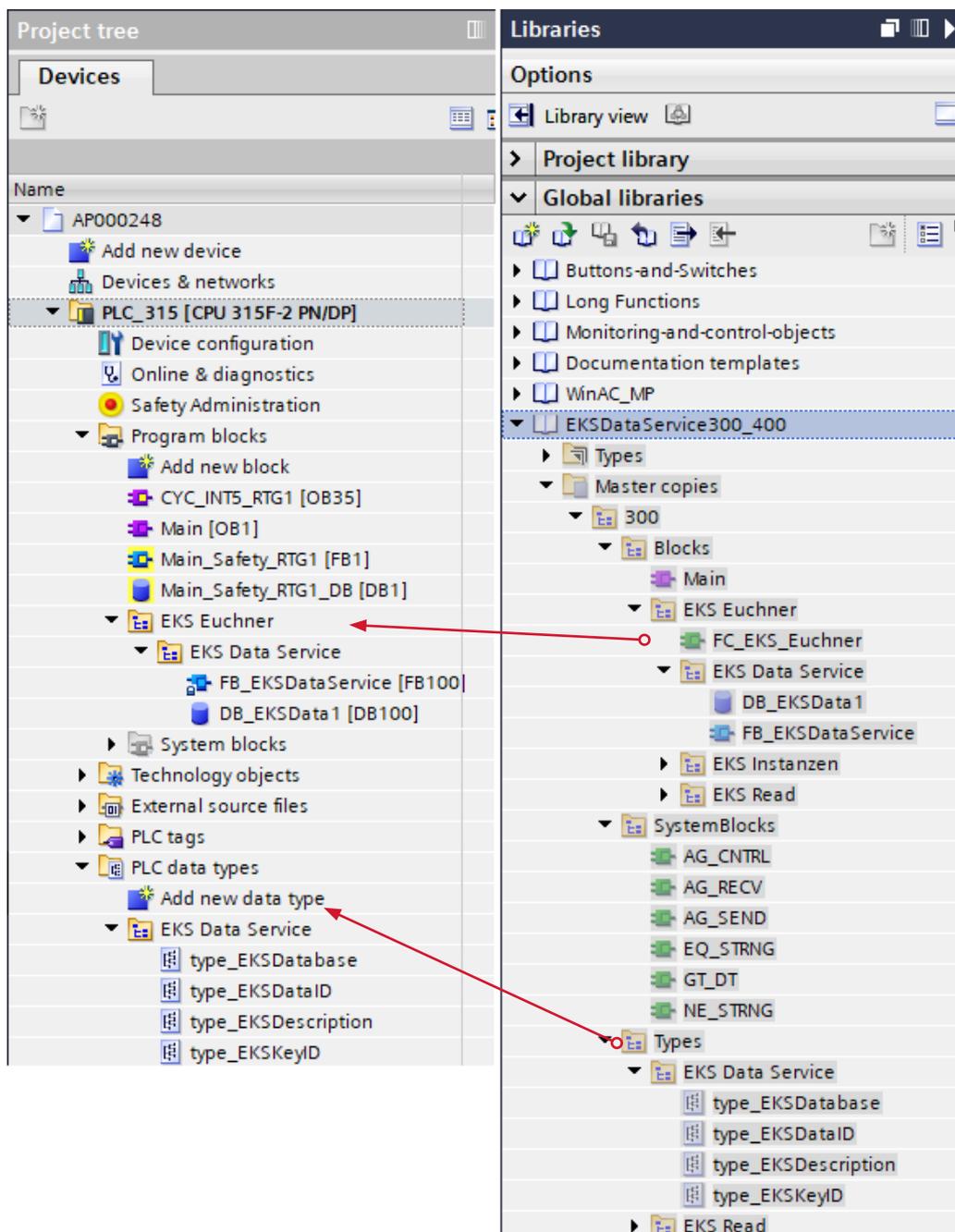


Bild 20: Bausteine ins Projekt kopieren

- Öffnen Sie die Bibliothek `Library_EKS_Data_Service-TIAV14SP1_PLC-300_400_YYYYMMDD` und kopieren Sie die Bausteine in die Ordner der *Projektnavigation*. Diese Bibliothek dient zum Auslesen der EKS Schlüssel. Es ist nicht zwingend notwendig diese Bibliothek zu verwenden.

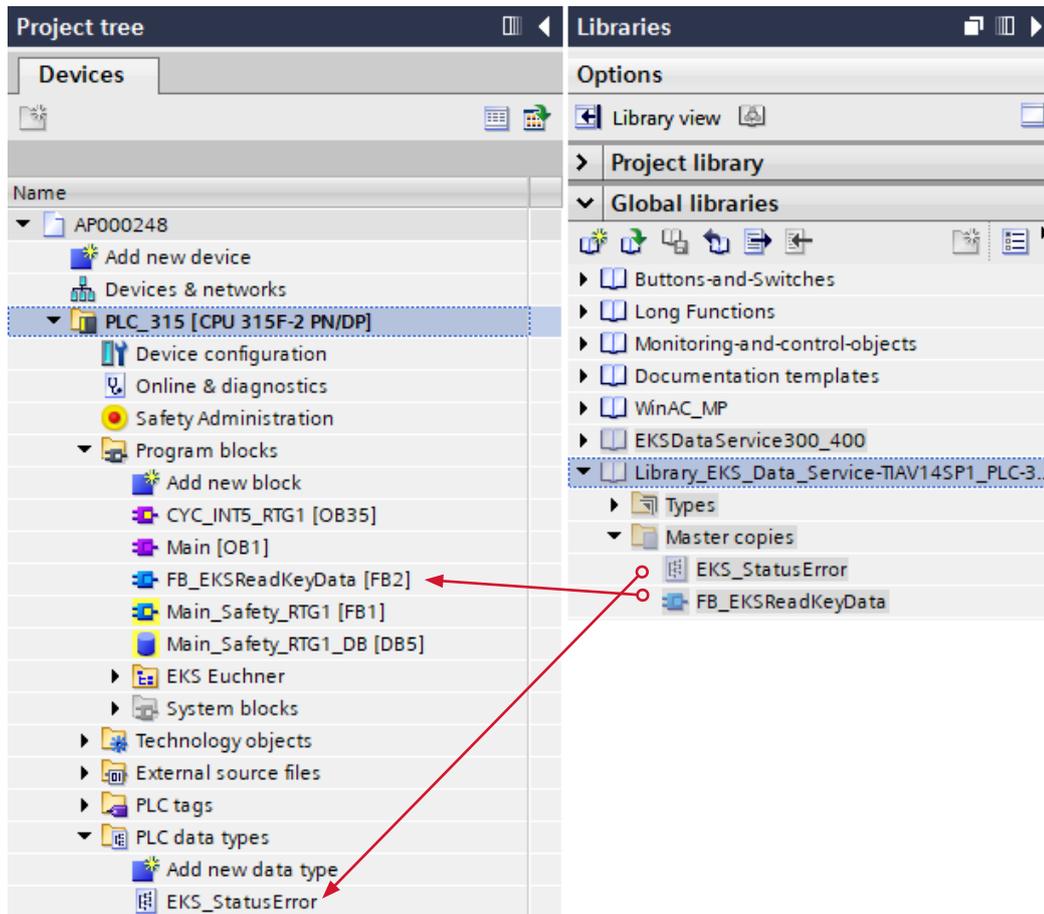


Bild 21: Bausteine ins Projekt kopieren

7.2.2. Anpassen des Datentyps `type_EKSDescription`

Öffnen Sie den Datentyp `type_EKSDescription`. Bilden Sie die im Dienst konfigurierten *EKM PLC DATA TYPES Name* und *PLC-Index* ab.

type_EKSDescription							
	Name	Data type	Default value	Accessible f...	Writa...	Visible in ...	Setpoint
1	KeyID	Int	1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2	LOCKED	Int	2	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3	OM	Int	3	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4	Department	Int	4	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5	Last_Name	Int	5	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6	First_Name	Int	6	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
7	Level_Milling	Int	7	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
8	Level_Turning	Int	8	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
9	Personnel_No	Int	9	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
10	Date	Int	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Bild 22: Anpassen des Datentyps `type_EKSDescription`

7.2.3. Anpassen des Datentyps `type_EKSDatabase`

Öffnen Sie den Datentyp `type_EKSDatabase`. Bilden Sie die im Dienst konfigurierten `EKM PLC DATA TYPES Name` und `PLC-Type` ab. Achten Sie dabei auf die Zuordnung der EKM Datentypen und SPS Datentypen, wie im Handbuch EKS Data Service PC beschrieben.

type_EKSDatabase							
	Name	Data type	Default value	Accessible f...	Writa...	Visible in ...	Setpoint
1	KeyID	String[16]	"	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2	LOCKED	Bool	false	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3	OM	Word	16#0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4	Department	String[2]	"	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5	Last_Name	String[40]	"	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6	First_Name	String[40]	"	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
7	Level_Milling	Byte	16#0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
8	Level_Turning	Byte	16#0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
9	Personnel_No	String[2]	"	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
10	Issue_Date	Date	D#1990-01-01	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Bild 23: Anpassen des Datentyps `type_EKSDatabase`

7.2.4. Erstellen des Datentyps `type_EKSRead`

Da in diesem Anwendungsbeispiel Daten ausschließlich aus einer Datenbank abgeholt werden, werden lediglich die Bits `Device_ready_for_operation` und `ElectronicKey_detected` sowie `EKSKeyID` vom Datentyp `type_EKSKeyID` benötigt.

type_EKSRead					
	Name	Data type	Default value	Visible in ...	Setpoint
1	Device_ready_for_operation	Bool	false	<input checked="" type="checkbox"/>	<input type="checkbox"/>
2	ElectronicKey_detected	Bool	false	<input checked="" type="checkbox"/>	<input type="checkbox"/>
3	EKSKeyID	"type_EKSKeyID"		<input checked="" type="checkbox"/>	<input type="checkbox"/>

Bild 24: Erstellen des Datentyps `type_EKSRead`

7.2.5. Hardware projektieren und Verbindungen anlegen

In diesem Beispiel werden zwei EKS Geräte mit PROFINET Schnittstelle projiziert. In dieser Applikation werden alle Schlüsselinformationen aus der csv-Datei übertragen. Aus diesem Grund muss nur die Seriennummer vom EKS Schlüssel gelesen werden. Sie können dafür die kleinsten Submodule (Lesen/Schreiben) im EKS (*Gerätesicht*) projektieren (siehe Applikation AP000238). Somit ist es möglich Speicher in der Steuerung einzusparen. Für dieses Beispiel muss im Submodul Lesen der EKS, die Startadresse 116 parametrieren werden.

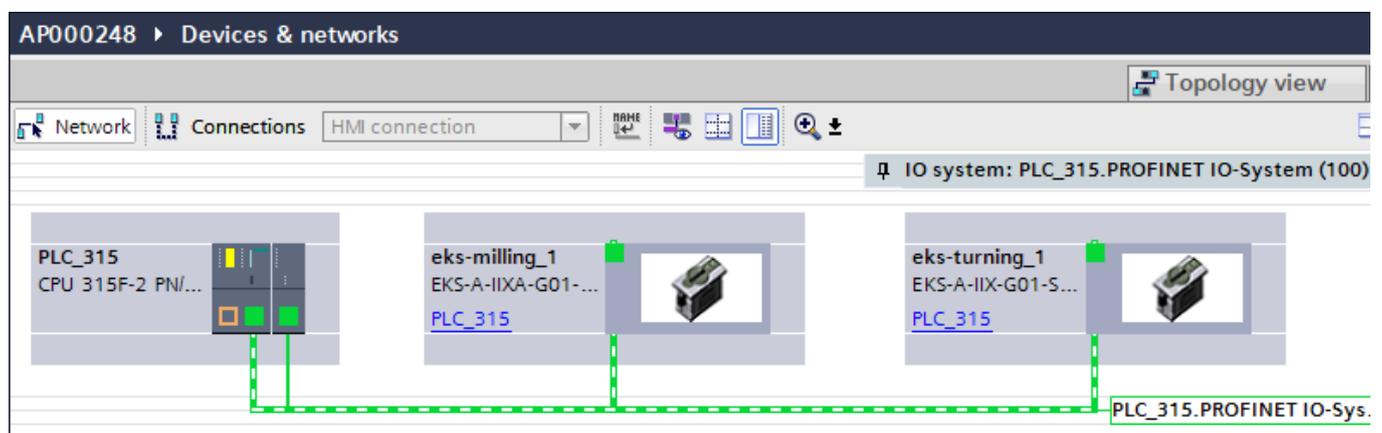


Bild 25: TIA Portal Netzsicht

Öffnen Sie die Eigenschaften des Kommunikationsprozessors und gehen Sie in die Ansicht *Ethernet-Adressen*. Verbinden Sie die *Subnetz*-Schnittstelle mit dem PROFINET-Netzwerks der SPS.

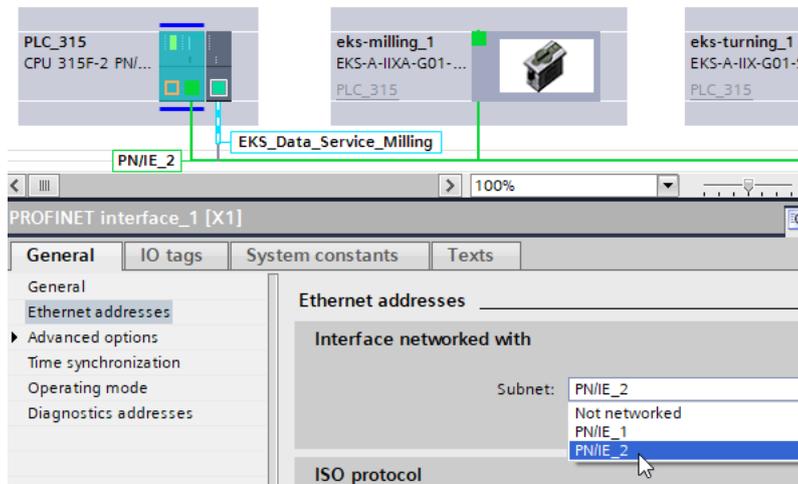


Bild 26: Schnittstelle mit Subnetz vernetzen

Für jede Instanz des Bausteins *FB_EKSDataService* (für jedes EKS) muss eine Verbindung zum EKS Data Service PC in der SPS angelegt werden. Um eine Verbindung anzulegen gehen Sie folgendermaßen vor:

1. Klicken Sie in der Netzansicht auf *Verbindungen* um sich die Verbindungen anzeigen zu lassen.

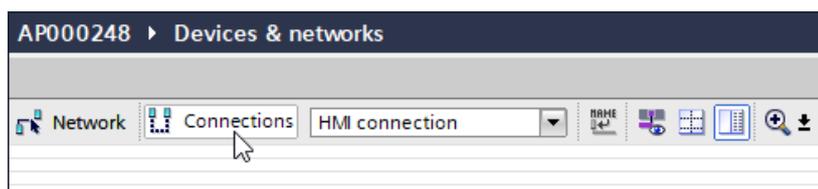


Bild 27: Verbindungen anzeigen

2. Fügen Sie eine neue Verbindung hinzu. Klicken Sie dafür mit der rechten Maustaste auf die Steuerung und wählen Sie *Neue Verbindung hinzufügen*.

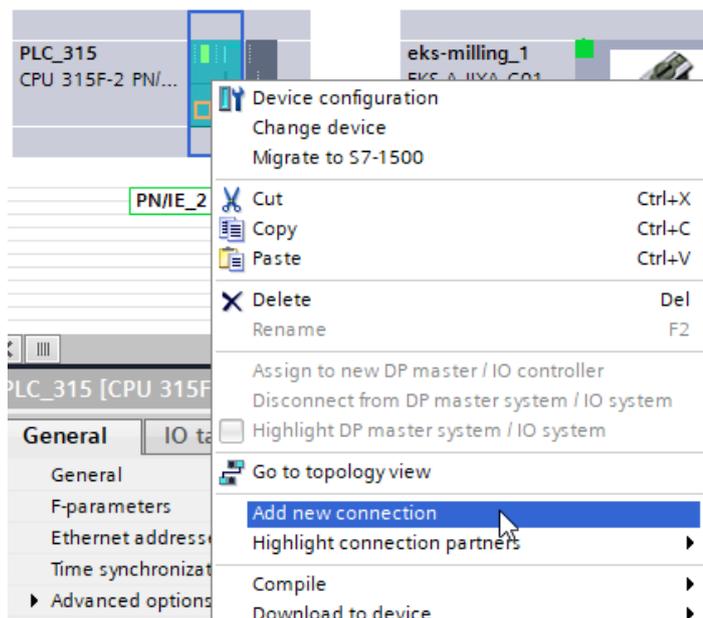


Bild 28: Neue Verbindung hinzufügen

3. Wählen Sie den Typ der Verbindung aus. Es handelt sich hierbei immer um eine *ISO-on-TCP* Verbindung. Klicken Sie auf *Aktive Verbindung aufbauen* und schließen Sie den Vorgang durch *Hinzufügen* ab.

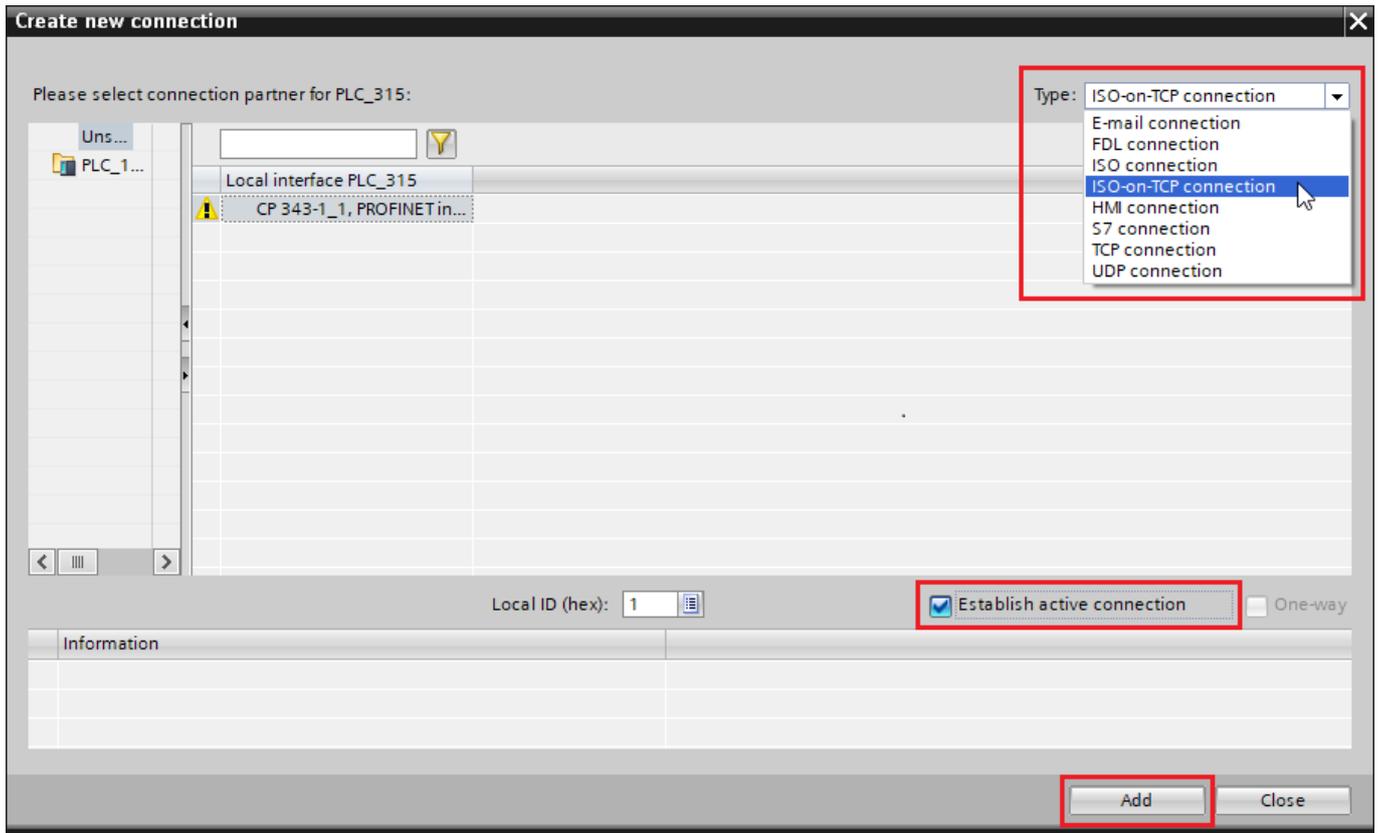


Bild 29: Verbindung anlegen

4. Gehen Sie in die allgemeinen Eigenschaften der angelegten Verbindung. Hier kann der Name der Verbindung zur besseren Identifizierung geändert werden.

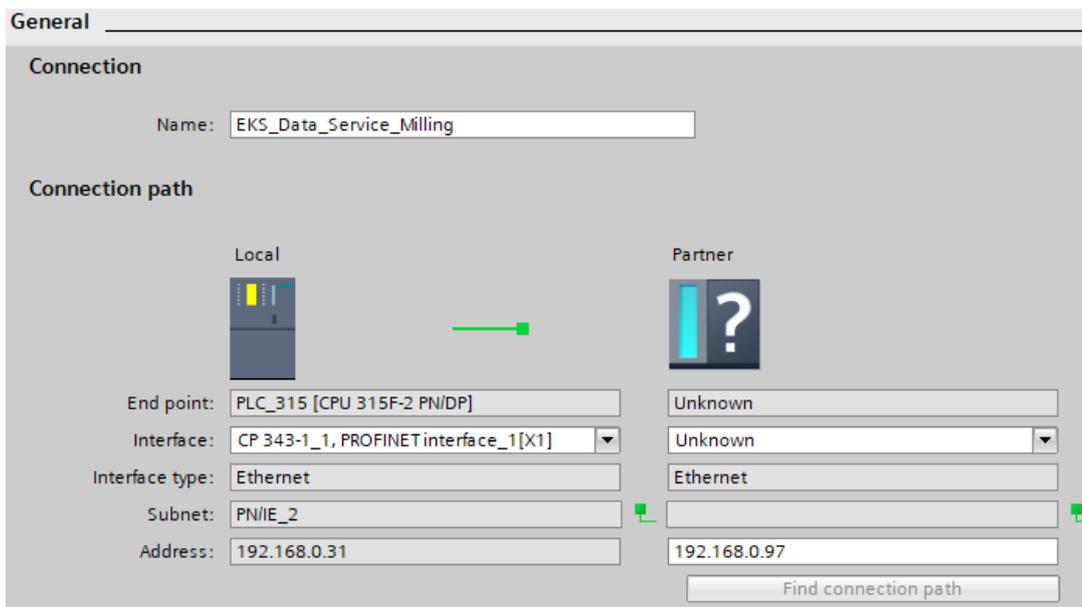


Bild 30: Eigenschaften Allgemein

- Gehen Sie jetzt in die Anzeige *Adressdetails*. Füllen Sie in der Spalte Partner die Zeile *IP (dec)* und *TSAP (ASCII)*. Die Partner IP-Adresse ist die des Servers, auf dem der EKS Data Service PC installiert ist. Die Partner *TSAP* muss bei jeder Verbindung *EKSSERV* heißen. Die lokale *TSAP* sollte bei mehreren Instanzen des *FB_EKSDataService* eindeutig sein. Dieser Name wird im EKS Data Service PC angezeigt.

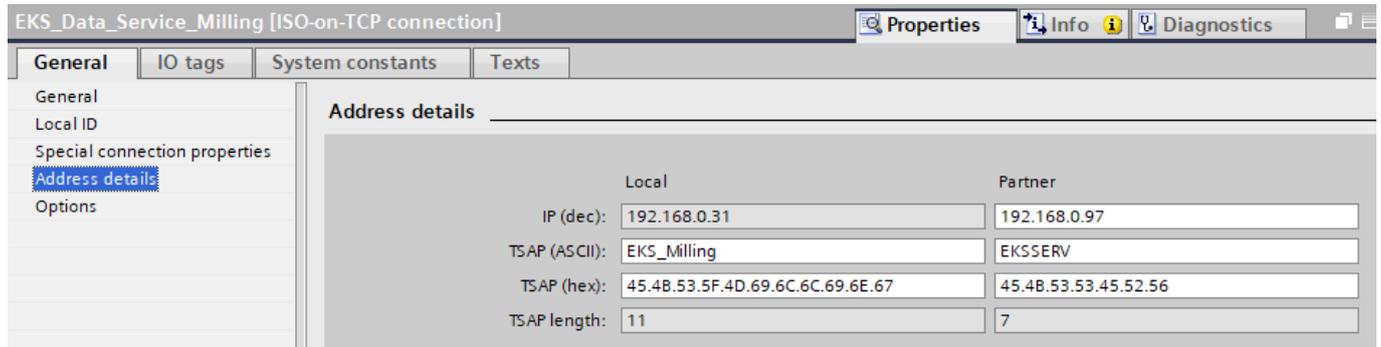


Bild 31: Einstellungen Adressdetails

- Sie benötigen später bei der Parametrierung des *FB_EKSDataService* die *Lokale ID (Con_ID)* sowie die Hardware-Adresse/ *LADDR (Interface_ID)* jeder Verbindung.

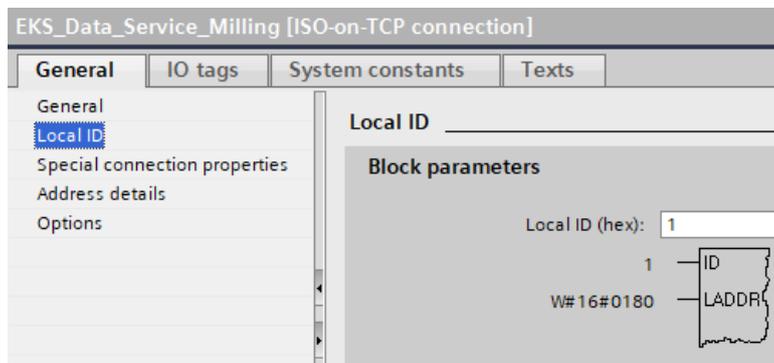


Bild 32: Einstellungen Lokale ID

Unser Beispiel beinhaltet zwei EKS. Es werden zwei Verbindungen angelegt und parametriert.

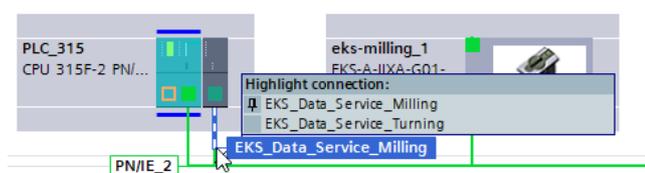


Bild 33: Parametrierte Verbindungen



HINWEIS!

Diese Schritte müssen für jede Instanz des Bausteins *FB_EKSDataService* wiederholt werden.

7.2.6. Anpassen des Datenbausteins DB_EKSData1

Variable	Datentyp	Beschreibung
Data_Key_Reader_Milling Data_Key_Reader_Turning	type_EKSDatabase	In dieser Variablen werden die Daten des aktuell im EKS platzierten Schlüssel abgelegt
Description	type_EKSDescription	In dieser Variablen wird die Struktur des Datentyp type_EKSDatabase dem FB bereitgestellt
ID	Array[0..100] of 'type_EKSDataID'	Die ID Variable wird benötigt, um zusätzliche Informationen wie den Zeitstempel für das Platzieren des Schlüssels oder den Zeitstempel der letzten Aktualisierung, die KeyID und die Checksumme des EKS Data Service zu speichern. Diese zusätzlichen Informationen werden für jeden platzierten Schlüssel eingetragen und im DB abgelegt. Dieser Datentyp darf nicht verändert werden und wird als Array in den DB abgelegt. Für jede Datenspeicherung (Data Array) benötigt man ein ID Array mit der gleichen Größe. Da die Daten auch nach Spannungsausfall/Neustart benötigt werden, ist es notwendig diese als remanent zu markieren.
Data	Array[0..100] of 'type_EKSDatabase'	Der gleiche Datentyp für Daten der platzierten Schlüssel, wird auch für die Speicherung der Daten für die Notlaufstrategie verwendet. Alle Daten der platzierten Schlüssel werden in das Data Array eingetragen. Die Daten werden erst überschrieben, wenn die Anzahl der platzierten Schlüssel größer ist, als die Anzahl der Array Einträge. Sollten die Daten überschrieben werden, dann wird immer der älteste Schlüsseleintrag überschrieben. Wird ein Schlüssel bei der Anfrage beim EKS Data Service PC nicht gelistet, dann wird der Schlüssel auch im Data Array gelöscht. Die Instanzen des FB_EKSDataService können alle auf die gleichen Daten (Description, ID, Data) im DB_EKSData1 zugreifen. Sollten die Daten auch nach einem Spannungsausfall zu Verfügung stehen, ist es notwendig diese als remanent zu markieren.
EKS_StatusError_Milling EKS_StatusError_Turning	EKS_StatusError	Für diese Applikation wurde ein Datentyp EKS_StatusError angelegt. Die darin enthaltenen Fehler und Daten sind im Handbuch EKS Data Service PLC beschrieben.

DB_EKSData1							
	Name	Data type	Offset	S..	Retain	Visible in ...	Setpoint
1	Static				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2	Data_Key_Reader_Milling	*type_EKSDatabase*	0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
3	ErrorStatus_Milling	*EKS_StatusError*	118.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
4	Data_Key_Reader_Turning	*type_EKSDatabase*	128.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
5	ErrorStatus_Turning	*EKS_StatusError*	246.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
6	Description	*type_EKSDescription*	256.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
7	ID	Array[0..20] of *type_EKSDataID*	276.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
8	Data	Array[0..20] of *type_EKSDatabase*	822.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Bild 34: Parametrieren des DB_EKSData1

7.2.7. Aufruf des Funktionsbausteins *FB_EKSReadKeyData*

Rufen Sie den Funktionsbaustein *FB_EKSReadKeyData* in einem zuvor von Ihnen erstellten FB (hier: *FB_EKS_Euchner*) auf. Dieser Baustein muss für jedes EKS aufgerufen und parametrieren werden. Um nicht unzählige Datenbausteine pro Instanz erstellen zu müssen, können Sie hierfür Multiinstanzen verwenden. Geben Sie an dem Eingang *StartAddressStatus* die Eingangsadresse des Statusbytes vom EKS an und an dem Eingang *StartAddressKeyID* die Startadresse an dem in Ihrer Anwendung die Schlüsselseriennummer beginnt. In unserem Beispiel beginnt die Schlüsselseriennummer im Eingangsbereich der Steuerung beim EKS *Milling* bei 1, da im Modul Lesen des EKS die Startadresse 116 parametrieren ist.

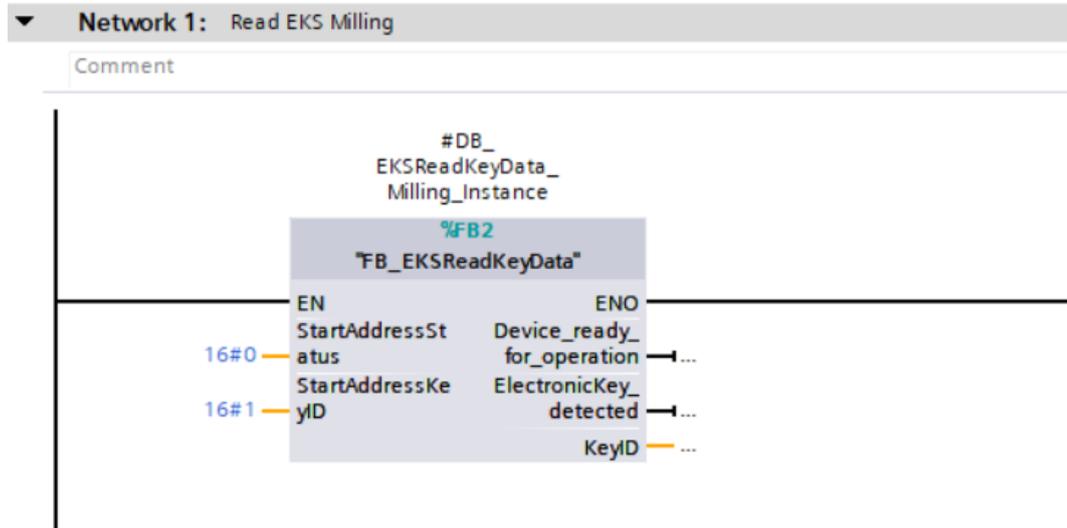


Bild 35: Aufruf des *FB_EKSReadKeyData*



HINWEIS!

Bitte beachten Sie, dass an den Eingängen des Bausteins hexadezimale Werte angegeben werden müssen.

7.2.8. Aufruf des Funktionsbausteins *FB_EKSDataService*

Rufen Sie den Funktionsbaustein *FB_EKSDataService* im gleichen FB (*FB_EKS_Euchner*) auf, wie zuvor den *FB_EKSReadKeyData*. Der Baustein *FB_EKSDataService* muss für jedes einzelne EKS aufgerufen und parametrieren werden. Anschließend rufen Sie den *FB_EKS_Euchner* im Hauptprogramm *Main (OB1)* auf.



HINWEIS!

Die Eingänge und Ausgänge des *FB_EKSDataService* sind im Handbuch EKS Data Service PLC detailliert beschrieben.

Network 3: EKS DataService Milling

Comment

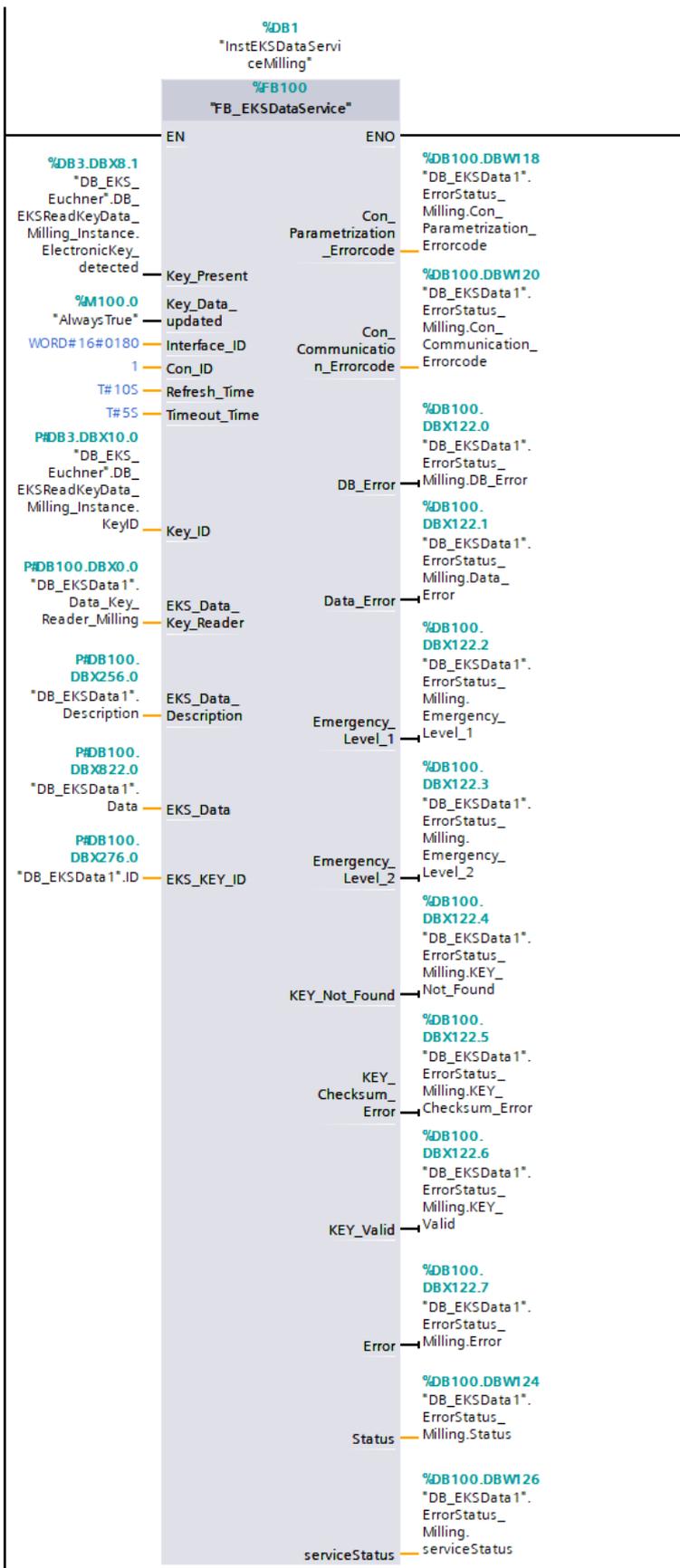


Bild 36: Aufruf eines `FB_EKSDataService` (Bsp.: Milling)

7.2.9. Beobachten der Daten im Datenbaustein *DB_EKSData1*

Nachdem alle Bausteine und UDTs parametrierung wurden, laden Sie Ihr Programm in die Steuerung. Im Dienst auf dem PC ist unter dem Reiter *LOG* zusehen, dass die Verbindung zur Steuerung aufgebaut wurde. Programmieren Sie im EKM ein paar Schlüssel und platzieren Sie diese an Ihre Anlage. Sobald Sie im TIA Portal den Baustein *DB_EKSData1* online betrachten, sehen Sie die abgeholten Daten aus der csv-Datei.

DB_EKSData1							
	Name	Data type	Offset	Start value	Monitor value	Retain	Visible in ...
1	Static					<input type="checkbox"/>	<input type="checkbox"/>
2	Data_Key_Reader_Milling	*type_EKSDatabase*	0.0			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
3	KeyID	String[16]	0.0	"	'02877825FF0010...	<input checked="" type="checkbox"/>	<input type="checkbox"/>
4	LOCKED	Bool	18.0	false	FALSE	<input checked="" type="checkbox"/>	<input type="checkbox"/>
5	OM	Word	20.0	16#0	16#FOOF	<input checked="" type="checkbox"/>	<input type="checkbox"/>
6	Department	String[2]	22.0	"	'QS'	<input checked="" type="checkbox"/>	<input type="checkbox"/>
7	Last_Name	String[40]	26.0	"	'Doe'	<input checked="" type="checkbox"/>	<input type="checkbox"/>
8	First_Name	String[40]	68.0	"	'John'	<input checked="" type="checkbox"/>	<input type="checkbox"/>
9	Level_Milling	Byte	110.0	16#0	16#01	<input checked="" type="checkbox"/>	<input type="checkbox"/>
10	Level_Turning	Byte	111.0	16#0	16#03	<input checked="" type="checkbox"/>	<input type="checkbox"/>
11	Personnel_No	String[2]	112.0	"	'10'	<input checked="" type="checkbox"/>	<input type="checkbox"/>
12	Issue_Date	Date	116.0	D#1990-01-01	D#2019-07-04	<input checked="" type="checkbox"/>	<input type="checkbox"/>
13	ErrorStatus_Milling	*EKS_StatusError*	118.0			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
14	Data_Key_Reader_Turning	*type_EKSDatabase*	128.0			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
15	ErrorStatus_Turning	*EKS_StatusError*	246.0			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
16	Description	*type_EKSDescription*	256.0			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
17	ID	Array[0..20] of *type_EKSDataID*	276.0			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
18	Data	Array[0..20] of *type_EKSDatabase*	822.0			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Bild 37: Daten aus csv-Datei im *DB_EKSData1*

8. Einbindung SPS-Bibliothek für S7-300/400 in STEP7 Manager

Für S7-300/400 verwenden Sie folgende Bibliothek:

EKS_Data_Service_PLC_Library_STEP_7_YYMMDD

8.1. Dearchivieren der Bibliothek

1. Klicken Sie in der Menüleiste auf *Datei* und wählen *Dearchivieren* aus.
2. Wählen Sie den Ordner mit der Bibliothek aus und dearchivieren Sie diesen in den gewünschten Zielordner.

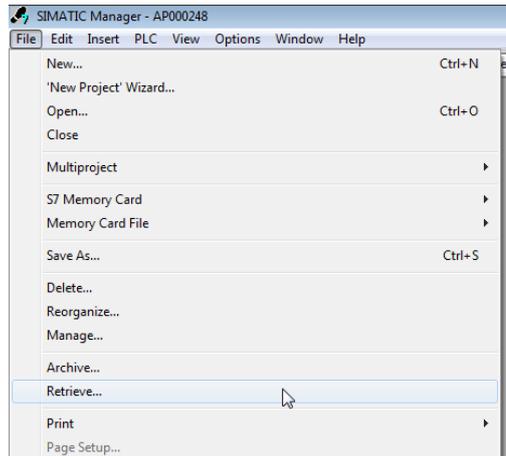


Bild 38: Bibliothek dearchivieren

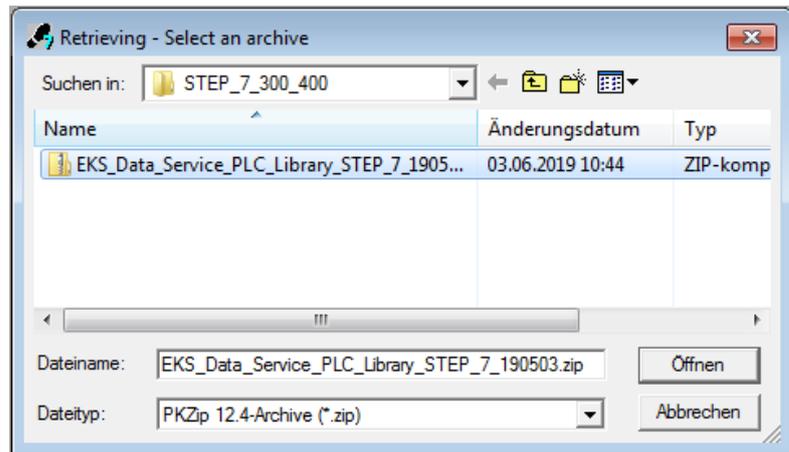


Bild 39: Bibliothek auswählen



TIPP!

Wenn die Bibliothek schon dearchiviert wurde, klicken Sie auf *Öffnen...* und wählen unter dem Reiter *Bibliotheken*, die Bibliothek aus, um Sie dem Projekt hinzuzufügen.

8.2. Integration des EKS Data Service in die SPS

8.2.1. Bausteine ins Projekt kopieren

Nachdem Sie die Bibliothek geöffnet haben, kopieren Sie, wie nachfolgend im Bild zusehen, die Bausteine entsprechend Ihrer Steuerungsumgebung in Ihr Projekt.

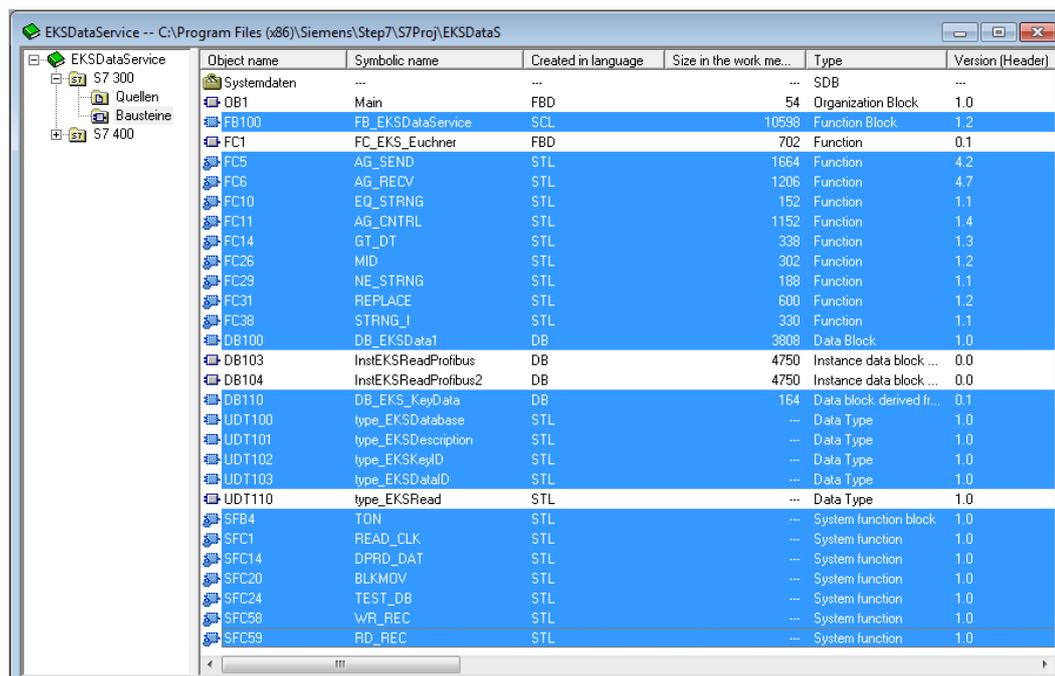


Bild 40: Bausteine ins Projekt kopieren

8.2.2. Anpassen des Datentyps *type_EKSDescription*

Öffnen Sie den Datentyp *type_EKSDescription*. Bilden Sie die im Dienst konfigurierten *EKM PLC DATA TYPES Name* und *PLC-Index* ab.

Address	Name	Type	Initial value
0.0		STRUCT	
+0.0	KeyID	INT	1
+2.0	LOCKED	INT	2
+4.0	OM	INT	3
+6.0	Department	INT	4
+8.0	Last_Name	INT	5
+10.0	First_Name	INT	6
+12.0	Level_Milling	INT	7
+14.0	Level_Turning	INT	8
+16.0	Personnel_No	INT	9
+18.0	Issue_Date	INT	10
=20.0		END_STRUCT	

Bild 41: Anpassen des Datentyps *type_EKSDescription*

8.2.3. Anpassen des Datentyps *type_EKSDatabase*

Öffnen Sie den Datentyp *type_EKSDatabase*. Bilden Sie die im Dienst konfigurierten *EKM PLC DATA TYPES Name* und *PLC-Type* ab. Achten Sie dabei auf die Zuordnung der EKM Datentypen und SPS Datentypen, wie im Handbuch EKS Data Service PC beschrieben.

Address	Name	Type	Initial value
0.0		STRUCT	
+0.0	KeyID	STRING[16]	''
+18.0	LOCKED	BOOL	FALSE
+20.0	OM	WORD	W#16#0
+22.0	Department	STRING[2]	''
+26.0	Last_Name	STRING[40]	''
+68.0	First_Name	STRING[40]	''
+110.0	Level_Milling	BYTE	B#16#0
+111.0	Level_Turning	BYTE	B#16#0
+112.0	Personnel_No	STRING[2]	''
+116.0	Issue_Date	DATE	D#1990-1-1
=118.0		END_STRUCT	

Bild 42: Anpassen des Datentyps *type_EKSDatabase*

8.2.4. Hardware projektieren und Verbindungen anlegen

In diesem Beispiel werden zwei EKS Geräte mit PROFINET Schnittstelle projektiert. In dieser Applikation werden alle Schlüsselinformationen aus der csv-Datei übertragen. Aus diesem Grund muss nur die Seriennummer vom EKS Schlüssel gelesen werden. Sie können dafür die kleinsten Submodule (Lesen/Schreiben) im EKS (*HW Config*) projektieren. Somit ist es möglich Speicher in der Steuerung einzusparen. Für dieses Beispiel muss im Submodul Lesen der EKS, die Startadresse 116 parametrieret werden.

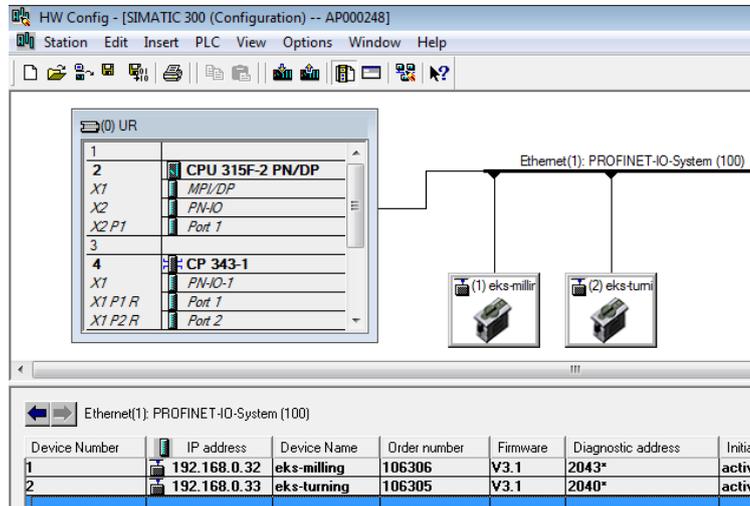


Bild 43: Ansicht HW Config

Öffnen Sie die *Objekteigenschaften* der PROFINET-Schnittstelle des Kommunikationsprozessors und öffnen Sie anschließend im Feld *Schnittstelle* die *Eigenschaften*. Vergeben Sie dem Kommunikationsprozessor eine IP-Adresse im gleichen Subnetz wie das bereits vorhandene Subnetz der SPS und wählen Sie das bereits vorhandene *Subnetz* des PROFINET-Netzwerks der SPS aus.

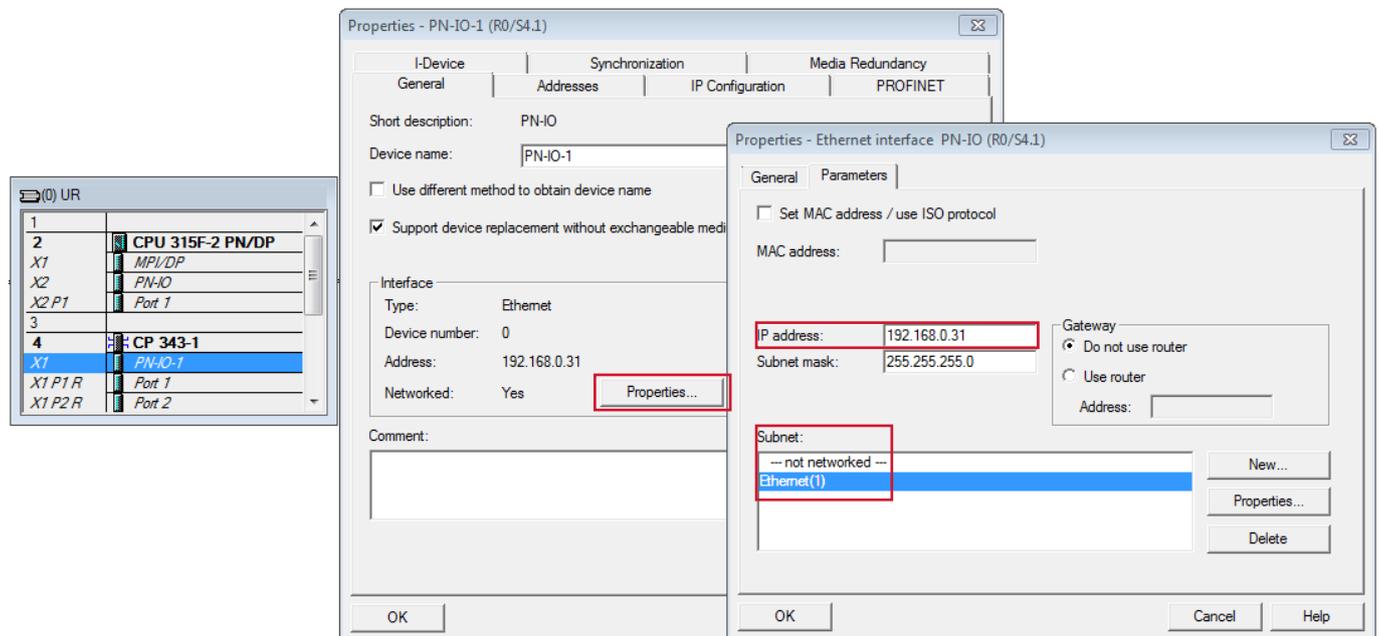


Bild 44: Schnittstelle mit Subnetz vernetzen

Für jede Instanz des Bausteins *FB_EKSDataService* (also für jedes EKS) muss eine Verbindung zum EKS Data Service PC in der SPS angelegt werden. Öffnen Sie NetPro in der Taskbar über das Icon 

1. Um eine Verbindung anlegen zu können, muss der EKS Data Service PC in der NetPro-Konfiguration etabliert werden. Fügen Sie rechts über den Baum eine *Andere Station* per Drag&Drop ins Netzwerk ein und öffnen Sie durch Doppelklicken die *Eigenschaften*. Unter dem Reiter *Allgemein* können Sie der Station einen anderen Namen geben, in unserem Beispiel *EKSDataService*. Anschließend öffnen Sie den Reiter *Schnittstellen*.

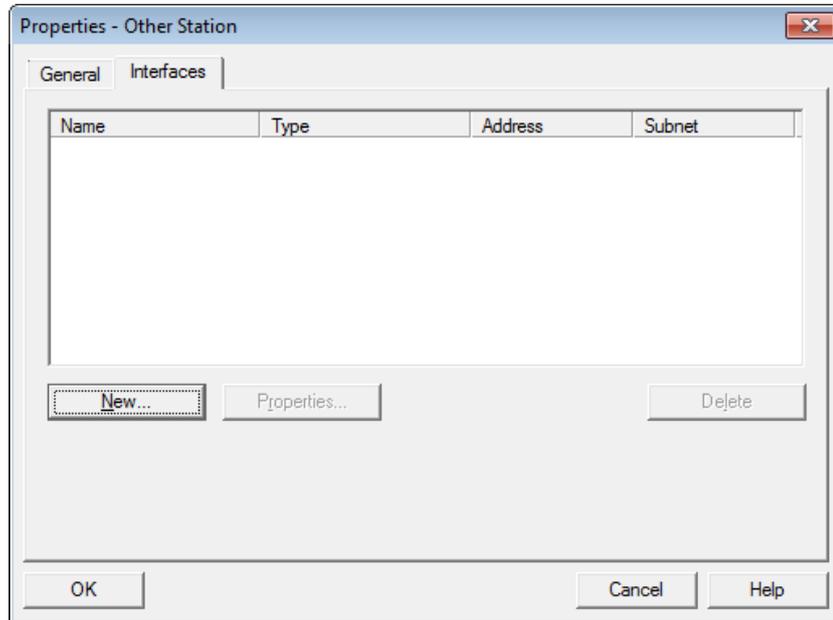


Bild 45: Schnittstelle anlegen von *Andere Station* (*EKSDataService*)

2. Klicken Sie auf *Neu...* und wählen Sie den Typ *Industrial Ethernet* aus.

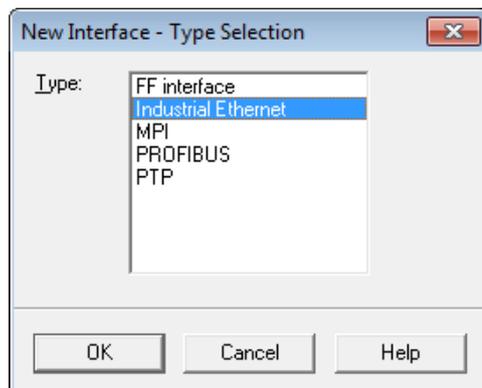


Bild 46: Typ der Verbindung auswählen

- Im nächsten Schritt werden die *Eigenschaften* der *Ethernet-Schnittstelle* unter dem Reiter *Parameter* automatisch geöffnet. Wichtig ist hierbei die Checkbox *MAC-Adresse einstellen / ISO Protokoll verwenden* zu deaktivieren. Anschließend wird die *IP-Adresse* und die *Subnetzmaske* des Servers eingetragen und das *Subnetz* der SPS ausgewählt (im Beispiel: Ethernet(1)). Bestätigen Sie die Eingabe durch Klicken auf *OK*.

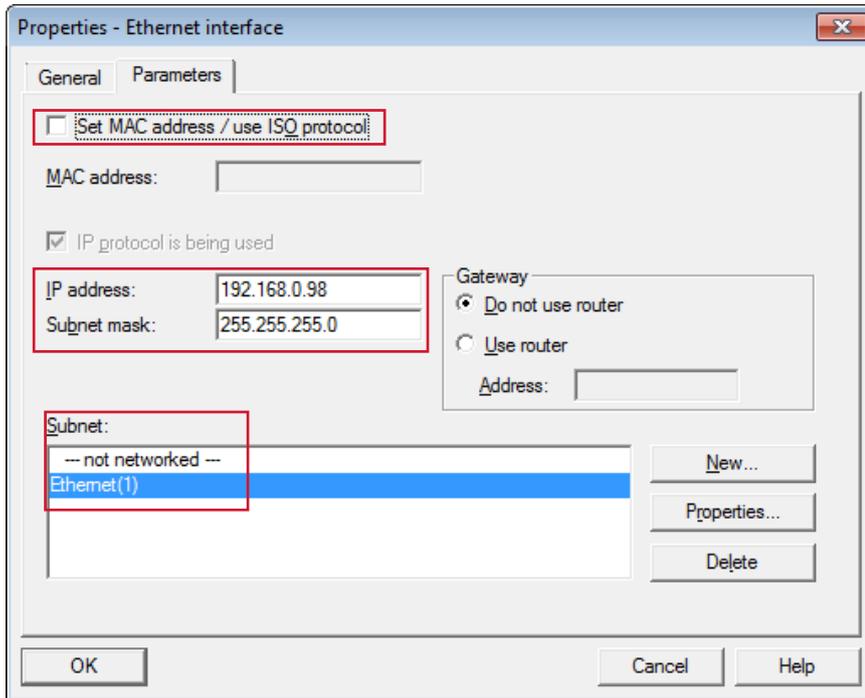


Bild 47: Eigenschaften - Ethernet Schnittstelle

- Jetzt können die Verbindungen angelegt werden. Wählen Sie dazu im NetPro-Konfigurator, wie im nachfolgenden Bild zu sehen, die CPU aus. Anschließend klicken Sie auf das Symbol in der Taskbar *Verbindung einfügen* 

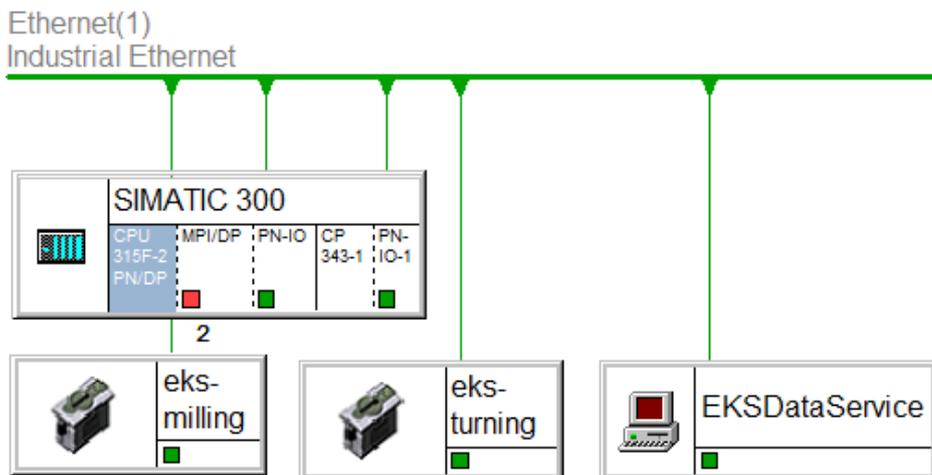


Bild 48: Auswahl der CPU um Verbindung anzulegen

5. Wählen Sie als *Verbindungspartner un spezifiziert* und den Verbindungstyp *ISO-on-TCP-Verbindung* aus.

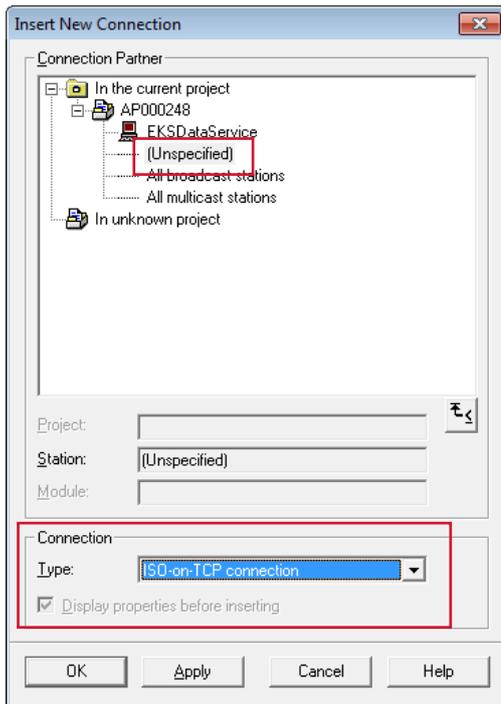


Bild 49: Neue Verbindung einfügen

6. Es wird automatisch das Eigenschaftsfenster der anzulegenden Verbindung geöffnet. Sie können unter dem Reiter Allgemein der Verbindung einen Namen geben. Wichtig ist das Setzen der Checkbox *Aktiver Verbindungsaufbau*.

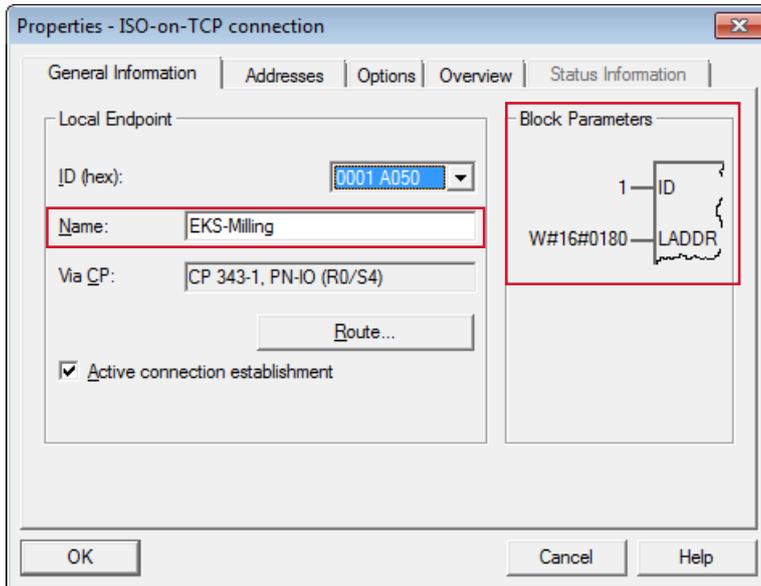


Bild 50: Allgemein und Bausteinparameter einer neuen Verbindung



HINWEIS!

Sie benötigen später bei der Parametrierung des *FB_EKSDataService* die *Lokale ID (Con_ID)* sowie die *Hardware-Adresse/LADDR (Interface_ID)* jeder Verbindung.

7. Öffnen Sie den Reiter *Adressen*. Hier muss die *TSAP* eingegeben werden. Geben Sie der lokalen *TSAP* einen beliebigen Namen und der Remote *TSAP* den Namen *EKSSERV* (fester Wert siehe Handbuch DataService PLC). Geben Sie die IP-Adresse des Servers an, auf dem der EKS Data Service PC installiert ist. Bestätigen Sie Ihre Eingaben durch Klicken auf *OK*.

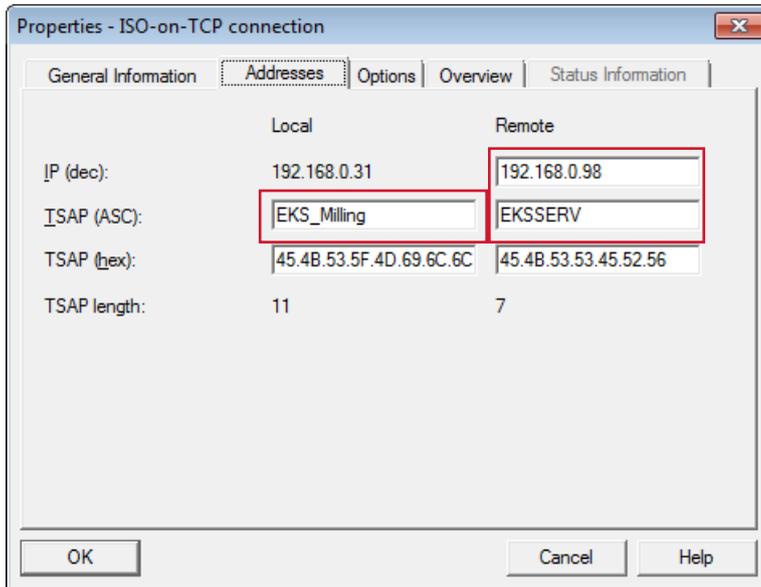


Bild 51: TSAP vergeben



HINWEIS!

Die Schritte 4 bis 7 müssen für die entsprechende Anzahl an EKS Instanzen wiederholt werden. Achten Sie darauf, dass jede Verbindung eindeutig ist (Name und *TSAP*).

Für diese Applikation sind zwei Verbindungen angelegt worden.

Local ID	Partner	Type	Active connection	Subnet	Local interface	Local address	Partner address
0001 A050	EKS-Milling	ISO-on-TCP connection	Yes	Ethernet(1) [E]	PN-IO-1	192.168.0.31	192.168.0.98
0002 A050	EKS-Turning	ISO-on-TCP connection	Yes	Ethernet(1) [E]	PN-IO-1	192.168.0.31	192.168.0.98

Bild 52: Angelegte Verbindungen

8.2.5. Anpassen des Bausteins DB_EKSData1

Variable	Datentyp	Beschreibung
Data_Key_Reader_Milling Data_Key_Reader_Turning	type_EKSDatabase	In dieser Variablen werden die Daten des aktuell im EKS platzierten Schlüssel abgelegt
Description	type_EKSDescription	In dieser Variablen wird die Struktur des Datentyp type_EKSDatabase dem FB bereitgestellt
ID	Array[0..100] of 'type_EKSDataID'	Die ID Variable wird benötigt, um zusätzliche Informationen wie den Zeitstempel für das Platzieren des Schlüssels oder den Zeitstempel der letzten Aktualisierung, die KeyID und die Checksumme des EKS Data Service zu speichern. Diese zusätzlichen Informationen werden für jeden platzierten Schlüssel eingetragen und im DB abgelegt. Dieser Datentyp darf nicht verändert werden und wird als Array in den DB abgelegt. Für jede Datenspeicherung (Data Array) benötigt man ein ID Array mit der gleichen Größe. Da die Daten auch nach Spannungsausfall/Neustart benötigt werden, ist es notwendig diese als remanent zu markieren.
Data	Array[0..100] of 'type_EKSDatabase'	Der gleiche Datentyp für Daten der platzierten Schlüssel, wird auch für die Speicherung der Daten für die Notlaufstrategie verwendet. Alle Daten der platzierten Schlüssel werden in das Data Array eingetragen. Die Daten werden erst überschrieben, wenn die Anzahl der platzierten Schlüssel größer ist, als die Anzahl der Array Einträge. Sollten die Daten überschrieben werden, dann wird immer der älteste Schlüsseleintrag überschrieben. Wird ein Schlüssel bei der Anfrage beim EKS Data Service PC nicht gelistet, dann wird der Schlüssel auch im Data Array gelöscht. Die Instanzen des FB_EKSDataService können alle auf die gleichen Daten (Description, ID, Data) im DB_EKSData1 zugreifen. Sollten die Daten auch nach einem Spannungsausfall zu Verfügung stehen, ist es notwendig diese als remanent zu markieren.
EKS_StatusError_Milling EKS_StatusError_Turning	EKS_StatusError	Für diese Applikation wurde ein Datentyp EKS_StatusError angelegt. Die darin enthaltenen Fehler und Daten sind im Handbuch EKS Data Service PLC beschrieben.

Address	Name	Type	Initial val	Comment
0.0		STRUCT		
+0.0	Data_Key_Reader_Milling	"type_EKSDatabase"		
+118.0	ErrorStatus_Milling	"EKS_StatusError"		
+128.0	Data_Key_Reader_Turning	"type_EKSDatabase"		
+246.0	ErrorStatus_Turning	"EKS_StatusError"		
+256.0	Description	"type_EKSDescription"		
+276.0	ID	ARRAY[0..20]		
*26.0		"type_EKSDataID"		
+822.0	Data	ARRAY[0..20]		
*118.0		"type_EKSDatabase"		
=3300.0		END_STRUCT		

Bild 53: Parametrieren des DB_EKSData1

8.2.6. Erstellen eines *EKSReadKeyData* Bausteins

Nachfolgend werden die für diese Applikation benötigten Werte des Schlüssels sowie Statusinformationen von jedem EKS abgeholt. Dazu wurde ein separater FB (*FB_EKSReadKeyData*) erstellt, der anschließend in einem weiteren FB (*FB_CallEKSDDataService*) (siehe Kapitel 8.2.7) aufgerufen wird.

Contents Of: 'Environment\Interface\STAT'

Name	Data Type	Address	Initial Value
KeyID_Milling	type_EKSKeyID	0.0	
KeyID_Turning	type_EKSKeyID	8.0	
Key_Detected_Milling	Bool	16.0	FALSE
Key_Detected_Turning	Bool	16.1	FALSE
DevReadyForOp_Milling	Bool	16.2	FALSE
DevReadyForOp_Turning	Bool	16.3	FALSE

FB2 : EKS Read Key Data

Comment:

```

Network : EKS_Key_Data_Milling
A      I      0.0
=      #DevReadyForOp_Milling      #DevReadyForOp_Milling
A      I      0.1
=      #Key_Detected_Milling      #Key_Detected_Milling

L      IB     1
T      #KeyID_Milling.typeKeyID[0]  #KeyID_Milling.typeKeyID[0]
L      IB     2
T      #KeyID_Milling.typeKeyID[1]  #KeyID_Milling.typeKeyID[1]
L      IB     3
T      #KeyID_Milling.typeKeyID[2]  #KeyID_Milling.typeKeyID[2]
L      IB     4
T      #KeyID_Milling.typeKeyID[3]  #KeyID_Milling.typeKeyID[3]
L      IB     5
T      #KeyID_Milling.typeKeyID[4]  #KeyID_Milling.typeKeyID[4]
L      IB     6
T      #KeyID_Milling.typeKeyID[5]  #KeyID_Milling.typeKeyID[5]
L      IB     7
T      #KeyID_Milling.typeKeyID[6]  #KeyID_Milling.typeKeyID[6]
L      IB     8
T      #KeyID_Milling.typeKeyID[7]  #KeyID_Milling.typeKeyID[7]
    
```

Bild 54: Daten vom EKS abholen



HINWEIS!

Für die Seriennummer des Schlüssels muss eine Variable des Datentyps *type_EKSKeyID* verwendet werden. Diese Variable wird verwendet um Daten am Server abzufragen.

8.2.7. Aufruf der Funktionsbausteine *FB_EKSReadKeyData* und *FB_EKSDataService*

Rufen Sie den Funktionsbaustein *FB_EKSReadKeyData* und *FB_EKSDataService* im FB (*FB_CallEKSDataService*) auf. Der Baustein *FB_EKSDataService* muss für jedes einzelne EKS aufgerufen und parametrierbar werden. Anschließend rufen Sie den *FB_EKS_Euchner* im Hauptprogramm *OB1 (CYCL_EXC)* auf.



HINWEIS!

Die Eingänge und Ausgänge des *FB_EKSDataService* sind im Handbuch EKS Data Service PLC detailliert beschrieben.

FB3 : Call EKS Data Service

Comment:

Network 1: Call FB 'EKSReadKeyData'

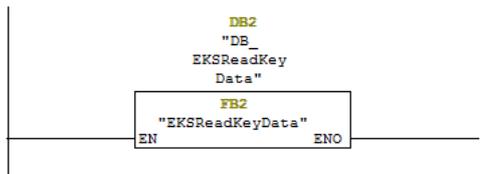


Bild 55: Aufruf *FB_EKSReadKeyData*

Network 2: Call FB 'EKSDDataService' Milling

DB4 "DB_ DataService_Milling"		FB100 "FB_EKSDDataService"	
EN			ENO
DB2.DEX16.0	"DB_EKSReadKeyData".Key_Detected_Milling	Key_Present	DB100.DSW118
M100.0	Key_Data_updated	Con_Parametri zation_ Error	DB100.DSW120
W#16#190	Interface_ID	Con_Communicat ion_ Error	"DB_EKSDat1".ErrorStatu s_Milling. Con_ Communicat io_Error
T#5S	Refresh_Time	Con_ Communicat io_Error	DB100.DEX122.0
T#10S	Timeout_Time		"DB_EKSDat1".ErrorStatu s_Milling. DB_Error
P#DB2.DBX0.0	"DB_EKSReadKeyData".KeyID_Milling	DB_Error	DB100.DEX122.1
P#DB100.DBX0.0	"DB_EKSDat1".Data_Key_Reader_Milling	Data_Error	"DB_EKSDat1".ErrorStatu s_Milling. Data_Error
P#DB100.DBX256.0	"DB_EKSDat1".Description	EKS_Data_Emergency_Level_1	DB100.DEX122.2
P#DB100.DBX22.0	"DB_EKSDat1".Data	EKS_Data_Emergency_Level_2	DB100.DEX122.3
P#DB100.DBX276.0	"DB_EKSDat1".ID	EKS_KEY_ID	DB100.DEX122.4
		KEY_Not_Found	"DB_EKSDat1".ErrorStatu s_Milling. Key_Not_Found
		KEY_Checksum_Error	DB100.DEX122.5
		KEY_Valid	"DB_EKSDat1".ErrorStatu s_Milling. KEY_Valid
		Error	DB100.DEX122.6
		Status	"DB_EKSDat1".ErrorStatu s_Milling. Error
		serviceStatus	DB100.DSW124
			"DB_EKSDat1".ErrorStatu s_Milling. Status
			DB100.DSW126
			"DB_EKSDat1".ErrorStatu s_Milling. serviceSt atus

Bild 56: Aufruf FB_EKSDDataService (Bsp.: Milling)

9. Wichtiger Hinweis – Bitte unbedingt sorgfältig beachten!

Dieses Dokument richtet sich an einen Konstrukteur, der die entsprechenden Kenntnisse in der Sicherheitstechnik hat und die Kenntnis der einschlägigen Normen besitzt, z.B. durch eine Ausbildung zum Sicherheitsingenieur. Nur mit entsprechender Qualifikation kann das vorgestellte Beispiel in eine vollständige Sicherheitskette integriert werden.

Das Beispiel stellt nur einen Ausschnitt aus einer vollständigen Sicherheitskette dar und erfüllt für sich allein genommen keine Sicherheitsfunktion. Zur Erfüllung einer Sicherheitsfunktion muss beispielsweise zusätzlich die Abschaltung der Energie der Gefährdungsstelle sowie auch die Software innerhalb der Sicherheitsauswertung betrachtet werden.

Die vorgestellten Applikationen stellen lediglich Beispiele zur Lösung bestimmter Sicherheitsaufgaben zur Absicherung von Schutztüren dar. Bedingt durch applikationsabhängige und individuelle Schutzziele innerhalb einer Maschine/Anlage können die Beispiele nicht erschöpfend sein.

Falls Fragen zu diesem Beispiel offen bleiben, wenden Sie sich bitte direkt an uns.

Nach der Maschinenrichtlinie 2006/42/EG ist der Konstrukteur einer Maschine bzw. Anlage verpflichtet, eine Risikobeurteilung durchzuführen und Maßnahmen zur Minderung des Risikos zu ergreifen. Er muss sich hierbei an die einschlägigen nationalen und internationalen Sicherheitsnormen halten. Normen stellen in der Regel den aktuellen Stand der Technik dar. Der Konstrukteur sollte sich daher laufend über Änderungen in den Normen informieren und seine Überlegungen darauf abstimmen, relevant für die funktionale Sicherheit sind u.a. die EN ISO 13849 und EN 62061. Diese Applikation ist immer nur als Unterstützung für die Überlegungen zu Sicherheitsmaßnahmen zu sehen.

Der Konstrukteur einer Maschine/Anlage ist verpflichtet die Sicherheitstechnik selbst zu beurteilen. Die Beispiele dürfen nicht zu einer Beurteilung herangezogen werden, da hier nur ein kleiner Ausschnitt einer vollständigen Sicherheitsfunktion sicherheitstechnisch betrachtet wurde.

Um die Applikationen der Sicherheitsschalter an Schutztüren richtig einsetzen zu können, ist es unerlässlich, dass die Normen EN ISO 13849-1, EN ISO 14119 und alle relevanten C-Normen für den jeweiligen Maschinentyp beachtet werden. Dieses Dokument ersetzt keinesfalls eine eigene Risikobeurteilung und kann auch nicht als Basis für eine Fehlerbeurteilung herangezogen werden.

Insbesondere bei einem Fehlerausschluss ist zu beachten, dass dieser nur vom Konstrukteur einer Maschine bzw. Anlage durchgeführt werden kann und dass hierzu eine Begründung notwendig ist. Ein genereller Fehlerausschluss ist nicht möglich. Nähere Auskünfte zum Fehlerausschluss gibt die EN ISO 13849-2.

Änderungen an Produkten oder innerhalb der Baugruppen von dritten Anbietern, die in diesem Beispiel verwendet werden, können dazu führen, dass die Funktion nicht mehr gewährleistet ist oder die sicherheitstechnische Beurteilung angepasst werden muss. In jedem Fall sind die Angaben in den Betriebsanleitungen sowohl seitens EUCHNER, als auch seitens der dritten Anbieter zugrunde zu legen, bevor diese Applikation in eine gesamte Sicherheitsfunktion integriert wird. Sollten hierbei Widersprüche zwischen Betriebsanleitungen und diesem Dokument auftreten, setzen Sie sich bitte mit uns direkt in Verbindung.

Verwendung von Marken- und Firmennamen

Alle aufgeführten Marken- und Firmennamen sind Eigentum des jeweiligen Herstellers. Deren Verwendung dient ausschließlich zur eindeutigen Identifikation kompatibler Peripheriegeräte und Betriebsumgebungen im Zusammenhang mit unseren Produkten.

Euchner GmbH + Co. KG
Kohlhammerstraße 16
70771 Leinfelden-Echterdingen
info@euchner.de
www.euchner.de

Ausgabe:
AP000248-01-10/19
Titel:
Applikation EKS
Beispiel Integration EKS Data Service

Copyright:
© EUCHNER GmbH + Co. KG, 10/2019

Technische Änderungen vorbehalten,
alle Angaben ohne Gewähr.