

EUCHNER

Applikation



Einbindung EKS mit TCP/IP Schnittstelle in OMRON Sysmac Studio®

DE

Inhalt

1.	Zu diesem Dokument	3
1.1.	Version	3
1.2.	Gültigkeit	3
1.3.	Zielgruppe.....	3
1.4.	Ergänzende Dokumente	3
1.5.	Hinweis.....	3
2.	Verwendete Bauteile / Module	4
2.1.	EUCHNER	4
2.2.	Andere.....	4
2.3.	Software	4
3.	Funktionsbeschreibung	4
4.	Importieren der Bibliothek	5
5.	Einbinden der Baustein-Bibliothek	7
5.1.	Variablentabelle der Bibliothek	7
5.2.	Einfügen der Baustein-Bibliothek.....	7
6.	Schlüsseldaten lesen und schreiben	9
6.1.	Programm an die SPS übertragen.....	9
6.2.	Inhalt des Schlüsselspeichers mittels einer Beobachtungstabelle lesen	9
6.3.	Inhalt des Schlüsselspeichers mittels einer Beobachtungstabelle schreiben	11
7.	Wichtiger Hinweis – Bitte unbedingt sorgfältig beachten!	14

1. Zu diesem Dokument

1.1. Version

Version	Datum	Änderung/Erweiterung	Kapitel
01-08/20	21.08.2020	Erstellung	Alle

1.2. Gültigkeit



Dieses Dokument dient zur Einbindung und Projektierung des EKS mit TCP/IP-Schnittstelle in OMRON Sysmac Studio®.

1.3. Zielgruppe

Konstrukteure und Anlagenplaner für Sicherheitseinrichtungen an Maschinen, sowie Inbetriebnahme- und Servicefachkräfte, die über spezielle Kenntnisse im Umgang mit Sicherheitsbauteilen sowie 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 (2100420)	Electronic-Key-System Handbuch Schlüsselaufnahme EKS und EKS FSA mit Ethernet TCP/IP Schnittstelle	
Ggf. beiliegende Datenblätter	Artikelspezifische Information zu Abweichungen oder Ergänzungen	

1.5. Hinweis

Diese Applikation basiert auf dem Handbuch des EKS mit TCP/IP-Schnittstelle. Die technischen Details sowie weitere Informationen entnehmen Sie bitte dem Handbuch. Im weiteren Verlauf des Dokuments wird das EKS mit TCP/IP-Schnittstelle kurz *EKS* genannt.

2. Verwendete Bauteile / Module

2.1. EUCHNER

Beschreibung	Bestellnummer / Artikel
EKS mit TCP/IP-Schnittstelle	100401 / EKS-A-EX-G01-ST02/03
	099265 / 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

Beschreibung	Bestellnummer / Artikel
NX102-1120 CPU UNIT	NX102-1120

2.3. Software

Beschreibung	Version
OMRON Sysmac Studio®	Version Lite Edition V1.27

3. Funktionsbeschreibung

Bei den EKS TCP/IP Geräten handelt es sich um ein Schreib-/Lesesystem mit Elektronik für die induktive bidirektionale Schnittstelle zum Transponder und Schnittstellenelektronik.

Die System-Anbindung erfolgt über die integrierte TCP/IP-Schnittstelle, welche als RJ45-Buchse ausgeführt ist. Zur TCP/IP-Anbindung wird ggf. ein separater Switch benötigt. Das EKS besitzt keinen integrierten Switch.

Der aktuelle Zustand der Schlüsselaufnahme wird über eine 3-farbige LED angezeigt.


Der Schlüssel wird für den Betrieb an der Schlüsselaufnahme platziert. Die Stromversorgung für den Transponder und die Daten werden kontaktlos zwischen Schlüsselaufnahme und Schlüssel übertragen.

Die Datenübertragung zwischen Steuerung und EKS wird mittels einer Bibliothek realisiert. Die Bibliothek handelt den Aufbau der Kommunikation zwischen Steuerung und EKS sowie das Senden und Empfangen der TCP/IP Kommunikationstelegramme.

Die Bibliothek kann auf www.euchner.de im Bereich *Downloads/Software/Beispieldateien* und *Bibliotheken/EKS* heruntergeladen werden.

4. Importieren der Bibliothek

1. Öffnen Sie den Reiter *Project* und wählen Sie aus dem Kontextmenü *Library* → *Show References* aus.

 EKS_Ethernet_TCP_IP - NX102 - Sysmac Studio

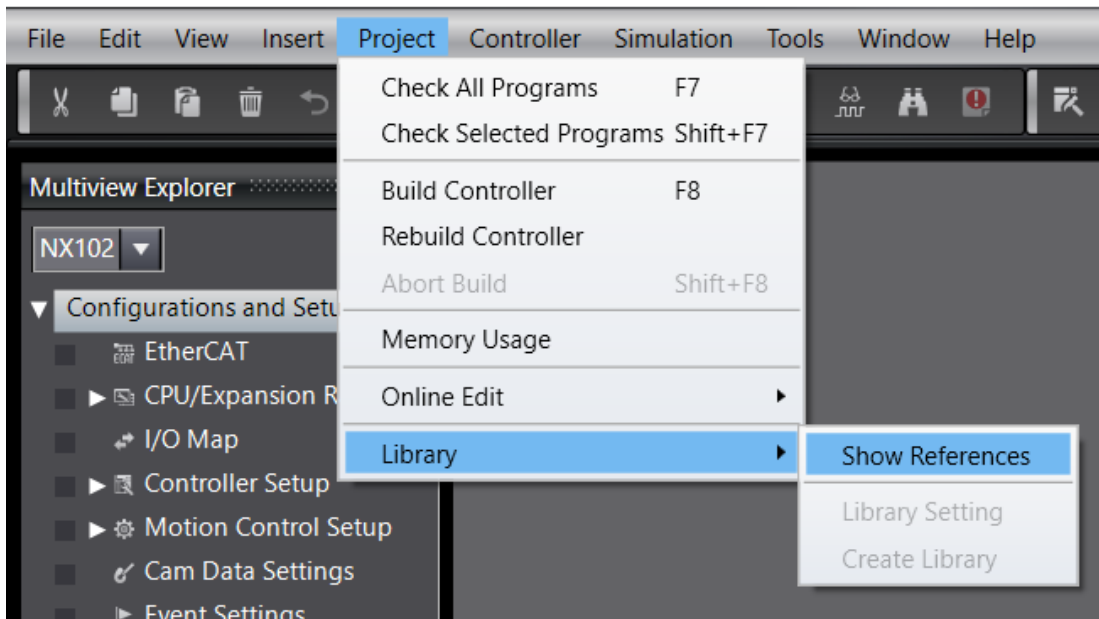


Bild 1: Kontextmenü Bibliothek

2. Klicken Sie auf das + Symbol um eine neue Bibliothek hinzuzufügen.

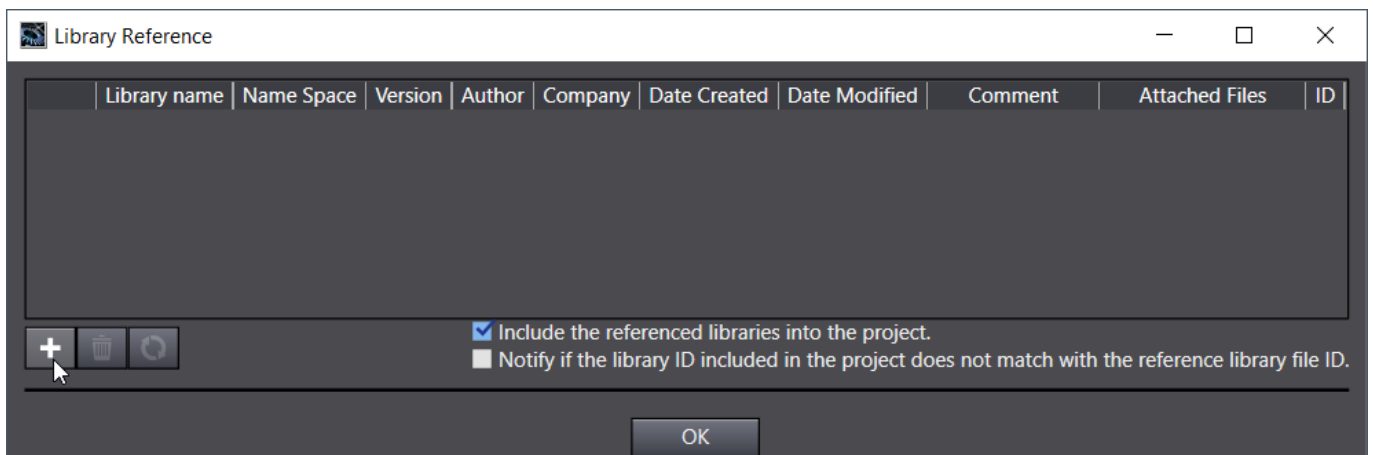


Bild 2: Hinzufügen der Bibliothek

3. Wählen Sie die Bibliothek aus und klicken Sie auf *Öffnen*.

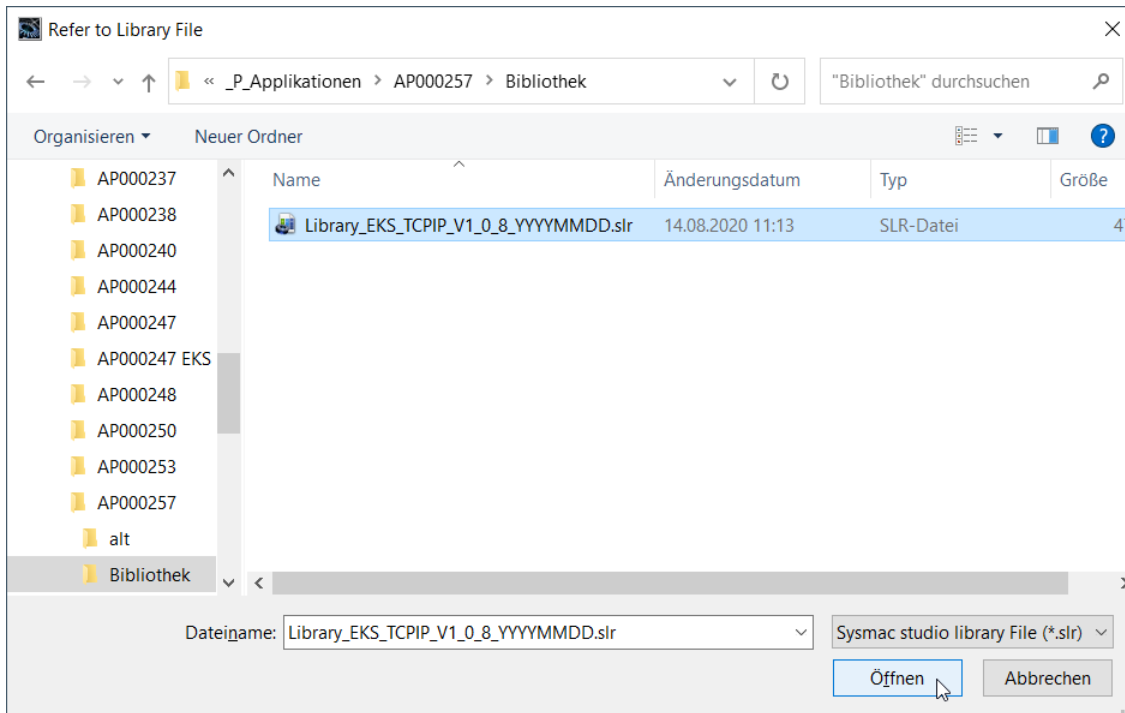


Bild 3: Auswählen der Bibliothek

4. Schließen Sie den Import der Bibliothek durch das Klicken auf *OK* ab.

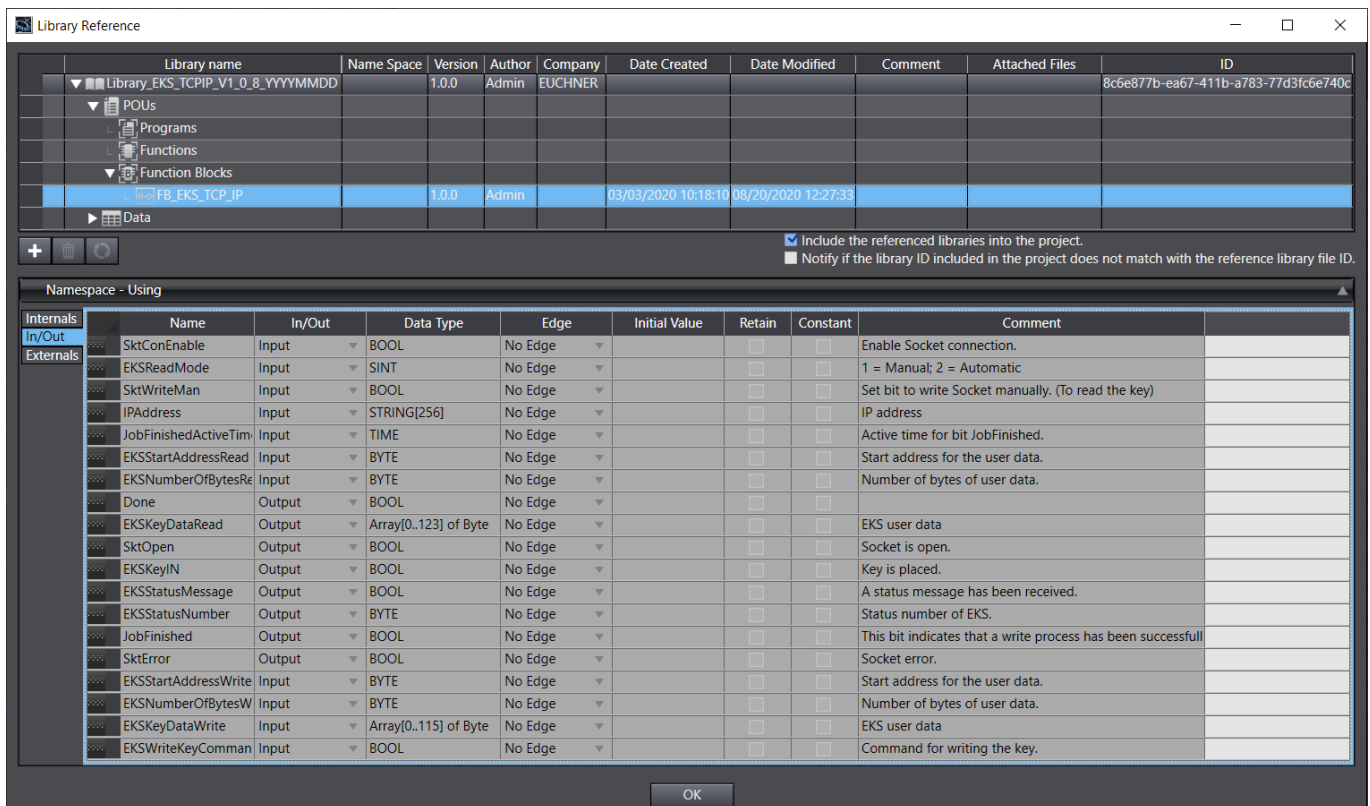


Bild 4: Import abschließen

5. Einbinden der Baustein-Bibliothek

5.1. Variablentabelle der Bibliothek

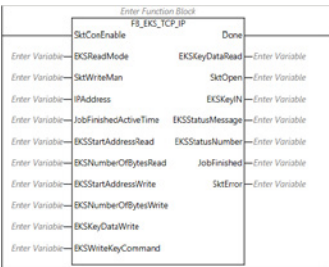
	Variable	Verwendung	Datentyp	Beschreibung
	FB_EKS_TCP_IP	-	FB_EKS_TCP_IP	Instanz für den FB
	SktConEnable	Input	BOOL	Aktiviere TCP/IP Socketverbindung
	EKSReadMode	Input	SINT	Modus der Abfrage 1= manuell; 2= automatisch
	SktWriteMan	Input	BOOL	Trigger zum Lesen der Daten im EKSReadMode = 1
	IPAddress	Input	STRING[256]	EKS IP-Adresse
	JobFinishedActiveTime	Input	TIME	Zeitwert wie lange das Bit JobFinished nach dem Schreibvorgang aktiv bleiben soll
	EKSStartAddressRead	Input	BYTE	Startadresse der anzufragenden Schlüsseldata
	EKSNumberOfBytesRead	Input	BYTE	Anzahl der anzufragenden Schlüsseldata
	EKSStartAddressWrite	Input	BYTE	Startadresse der zuschreibenden Schlüsseldata
	EKSNumberOfBytesWrite	Input	BYTE	Anzahl der zuschreibenden Schlüsseldata
	EKSKeyDataWrite	Input	Array[0..115] of BYTE	Zuschreibende Daten
	EKSWriteKeyCommand	Input	BOOL	Befehl zum Schlüsselpeicher beschreiben
	Done	Output	BOOL	-
	EKSKeyDataRead	Output	Array[0..123] of BYTE	Antwort der Nutzdaten des EKS Schlüssels
	SktOpen	Output	BOOL	Socketverbindung ist geöffnet
	EKSKeyIN	Output	BOOL	EKS Schlüssel in der Schlüsselaufnahme platziert
	EKSStatusMessage	Output	BOOL	Eine EKS Statusmeldung wurde empfangen
	EKSStatusNumber	Output	BYTE	EKS Status
	JobFinished	Output	BOOL	Schreibvorgang abgeschlossen
	SktError	Output	BOOL	Fehler Socketverbindung

Tabelle 1: Variablentabelle Bibliothek

5.2. Einfügen der Baustein-Bibliothek

1. Öffnen Sie ein Programm (z.B. *Section0*) und ziehen Sie aus der *Toolbox* mit Drag & Drop den Baustein in einen neuen Rung.

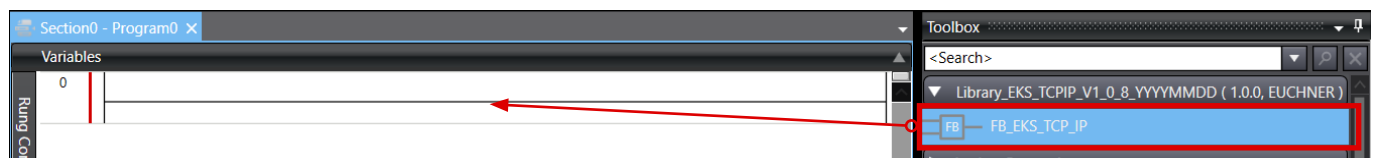


Bild 5: Hinzufügen des Baustein im Hauptprogramm

- Es wird automatisch eine Instanz des Bausteins erzeugt. Schreiben Sie den **Namen** der Instanz *FB_EKS_TCP_IP* in das Feld für die Variable. In diesem Beispiel geben Sie *EKS_Milling* ein und bestätigen mit der Eingabetaste. In der Variablentabelle (*Section0*) wird die Variable automatisch mit dem dazugehörigen Variablentyp angelegt.
- Erzeugen Sie nun, wie in Schritt 2, für jeden Ein- und Ausgang die entsprechenden Variablen.
- Fügen Sie dem Rung eine Eingangsvariable (z.B. *EKS_Milling_SktConEnable*) als Kontakt (*Examine On*) hinzu.

Variable	Wert
EKSReadMode	1 = manueller Modus; Die Schlüsseldata werden durch Triggern der Variable <i>SktWriteMan</i> empfangen 2 = Automatischer Modus; Das Empfangen der Schlüsseldata wird durch Auslesen des Key-Status <i>KeyIN</i> (siehe Handbuch) getriggert.
IPAddress	IPv4-Adresse, eingeschlossen in Hochkomma z.B. '192.168.0.222'
JobFinishedActiveTime	Variabel (Zeitwert (z.B. T#500ms) → Zeitwert wie lange das Bit <i>JobFinished</i> nach dem Schreibvorgang aktiv bleiben soll
EKSStartAddressRead	Variabel (Wert von 0 bis 116) → Startadresse der verwendeten Nutzdaten
EKSNumberOfBytesRead	Variabel (Wert von 1 bis 124) → Anzahl der zu lesenden Bytes
EKSStartAddressWrite	Variabel (Wert von 0 bis 112) → Startadresse der zu schreibenden Nutzdaten
EKSNumberOfBytesWrite	Variabel (Wert von 4 bis 116) → Anzahl der zu schreibenden Bytes

Tabelle 2: Eingangsvariablen

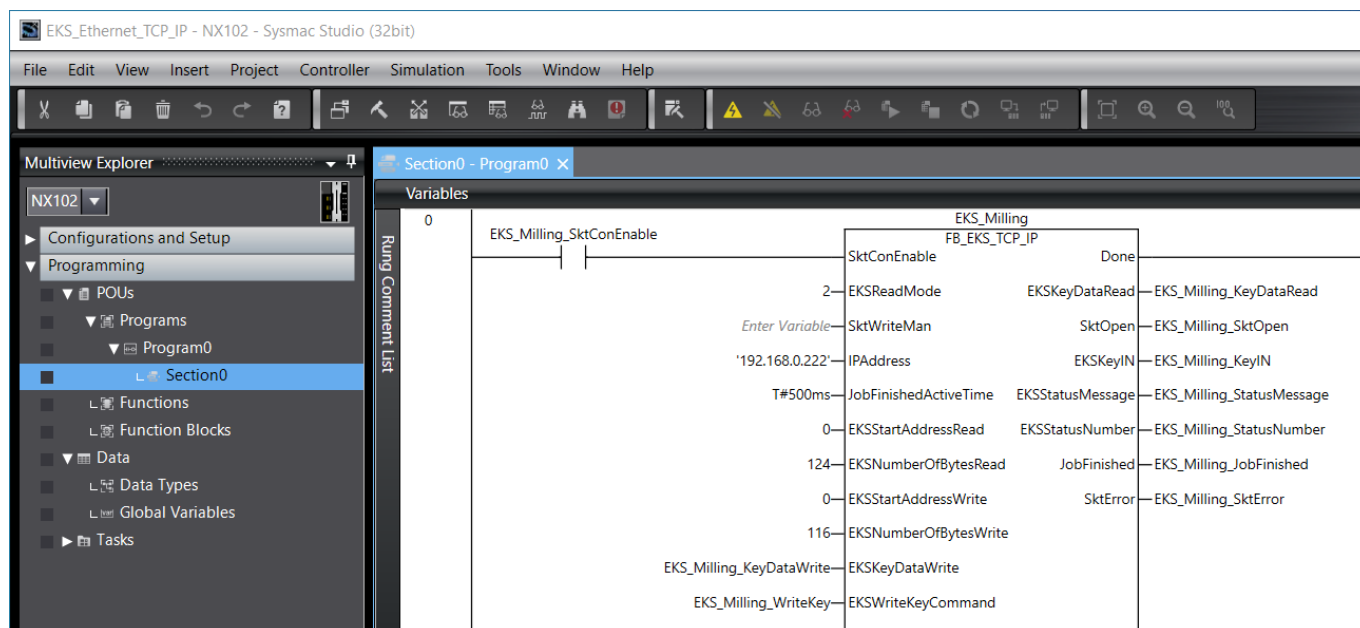


Bild 6: Instanzbaustein deklarieren



WICHTIG

Beim Schreib-/Lese-Schlüssel mit frei programmierbaren 116 Bytes ist der Speicher in 4-Byte-Blöcken organisiert. Dies bedeutet, die Start-Adresse muss beim Schreiben im Bereich Byte Nr. 0 bis Byte Nr. 112, immer in 4-Byte-Schritten, angegeben werden (Byte Nr. 0, 4, 8 ... 112). Außerdem muss immer in einem Vielfachen von 4-Bytes großen Blöcken geschrieben werden (4, 8, 12 ... 116 Bytes)! Beim Lesen kann allerdings wiederum byteweise auf den Speicher zugegriffen werden, ohne die oben genannte Einschränkung beim Schreiben. Der Schreib-/Lese-Schlüssel hat zusätzlich eine einmalige 8-Byte große Seriennummer, die bei der Schlüssel-Produktion absolut unzerstörbar in den Speicher geschrieben wird. Die Seriennummer kann daher nicht geändert werden. Diese Seriennummer dient zur sicheren Unterscheidung eines jeden einzelnen Schlüssels. Für eine sichere Unterscheidung ist es erforderlich alle 8 Bytes komplett auszuwerten. Die Seriennummer schließt sich an den frei programmierbaren Speicher an. Die Seriennummer kann unter Eingabe der Start-Adresse Byte Nr. 116 und Anzahl Bytes 8 ausgelesen werden.





TIPP

Die Vergabe der IP-Adresse der Schlüsselaufnahme EKS wird mittels des Web-Interfaces durchgeführt. Die Beschreibung hierzu finden Sie im Handbuch, Kapitel 7.2

6. Schlüsseldaten lesen und schreiben

6.1. Programm an die SPS übertragen

Übertragen Sie das Programm indem Sie online gehen . Anschließend öffnen Sie den Reiter *Controller* → *Transfer...* → *To Controller...* .

 EKS_Ethernet_TCP_IP - NX102 - Sysmac Studio

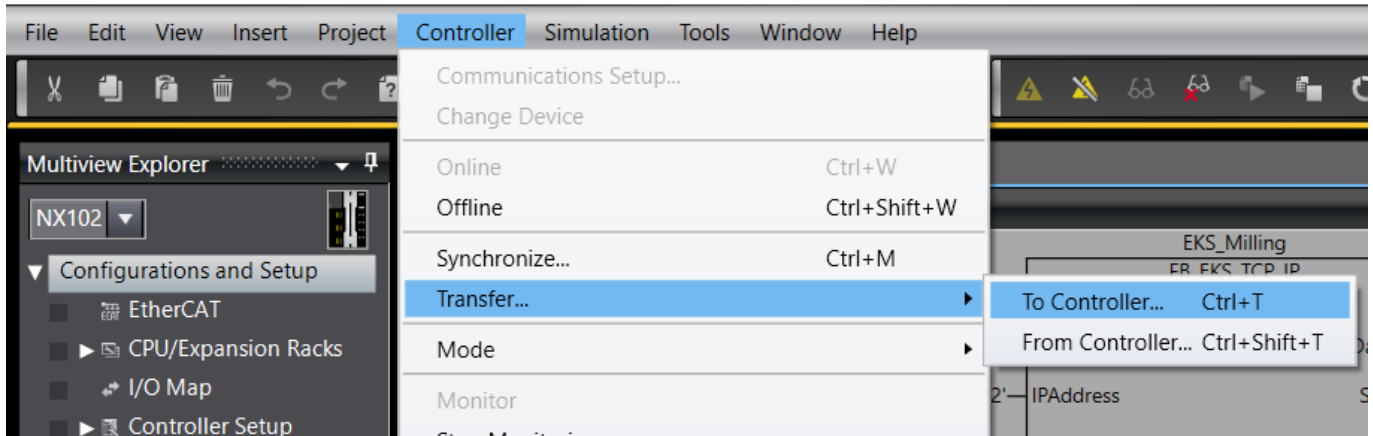


Bild 7: Programm an SPS übertragen

6.2. Inhalt des Schlüsselspeichers mittels einer Beobachtungstabelle lesen

Durch Setzen des Bits *EKS_Milling_SktConEnable* wird die Verbindung zum EKS aufgebaut und *SktOpen* wird *True*. Sobald ein Schlüssel in der Schlüsselaufnahme platziert wird, wechselt *EKSKeyIN* zu *True*.

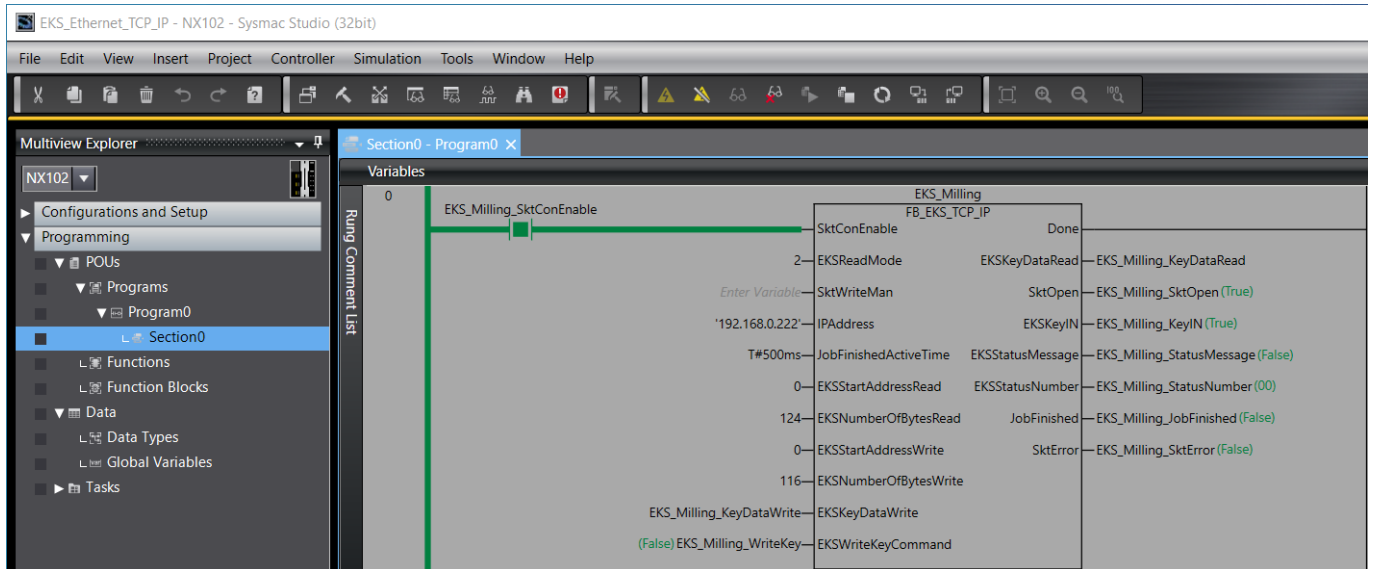


Bild 8: Instanzbaustein online

Durch die Parametrierung am Eingang *EKSReadMode* mit dem Wert 2, werden die Schlüsseldaten nach dem Platzieren des Schlüssels automatisch ausgelesen. Dies kann in einer Beobachtungstabelle beobachtet werden.

Device name	Name	Online value	Modify	Co
NX102	Program0.EKS_Milling_KeyDataRead[0..123]			
	EKS_Milling_KeyDataRead[0]	E (16#45)		
	EKS_Milling_KeyDataRead[1]	U (16#55)		
	EKS_Milling_KeyDataRead[2]	C (16#43)		
	EKS_Milling_KeyDataRead[3]	H (16#48)		
	EKS_Milling_KeyDataRead[4]	N (16#4E)		
	EKS_Milling_KeyDataRead[5]	E (16#45)		
	EKS_Milling_KeyDataRead[6]	R (16#52)		
	EKS_Milling_KeyDataRead[7]	(16#20)		
	EKS_Milling_KeyDataRead[8]	G (16#47)		
	EKS_Milling_KeyDataRead[9]	m (16#6D)		
	EKS_Milling_KeyDataRead[10]	b (16#62)		
	EKS_Milling_KeyDataRead[11]	H (16#48)		
	EKS_Milling_KeyDataRead[12]	+ (16#2B)		
	EKS_Milling_KeyDataRead[13]	C (16#43)		
	EKS_Milling_KeyDataRead[14]	o (16#6F)		
	EKS_Milling_KeyDataRead[15]	. (16#2E)		
	EKS_Milling_KeyDataRead[16]	K (16#4B)		
	EKS_Milling_KeyDataRead[17]	G (16#47)		
	EKS_Milling_KeyDataRead[18]	. (16#00)		
	EKS_Milling_KeyDataRead[19]	00		
	EKS_Milling_KeyDataRead[20]	00		

Bild 9: Beobachtungstabelle Schlüsseldaten Lesen

6.3. Inhalt des Schlüsselspeichers mittels einer Beobachtungstabelle schreiben

Eine weitere Beobachtungstabelle wurde so vorbereitet, dass damit auch Daten auf den Schlüssel geschrieben werden können. Hierzu wurden Werte, wie im nachfolgenden Screenshot zu sehen, vorbereitet. Die Eingabe des Wertes in die Spalte *Modify* muss mit der Eingabetaste bestätigt werden.

Device name	Name	Online value	Modify
NX102	Program0.EKS_Milling_KeyDataWrite[0..115]		
	EKS_Milling_KeyDataWrite[0]	E (16#45)	E
	EKS_Milling_KeyDataWrite[1]	K (16#4B)	K
	EKS_Milling_KeyDataWrite[2]	S (16#53)	S
	EKS_Milling_KeyDataWrite[3]	(16#20)	
	EKS_Milling_KeyDataWrite[4]	E (16#45)	E
	EKS_Milling_KeyDataWrite[5]	T (16#54)	T
	EKS_Milling_KeyDataWrite[6]	H (16#48)	H
	EKS_Milling_KeyDataWrite[7]	E (16#45)	E
	EKS_Milling_KeyDataWrite[8]	R (16#52)	R
	EKS_Milling_KeyDataWrite[9]	N (16#4E)	N
	EKS_Milling_KeyDataWrite[10]	E (16#45)	E
	EKS_Milling_KeyDataWrite[11]	T (16#54)	T
	EKS_Milling_KeyDataWrite[12]	(16#20)	
	EKS_Milling_KeyDataWrite[13]	T (16#54)	T
	EKS_Milling_KeyDataWrite[14]	C (16#43)	C
	EKS_Milling_KeyDataWrite[15]	P (16#50)	P
	EKS_Milling_KeyDataWrite[16]	/ (16#2F)	/
	EKS_Milling_KeyDataWrite[17]	I (16#49)	I
	EKS_Milling_KeyDataWrite[18]	P (16#50)	P
	EKS_Milling_KeyDataWrite[19]	00	

Bild 10: Beobachtungstabelle Schlüsseldaten *Schreiben*

Durch das Ausführen des Schreibbefehls *WriteKeyCommand* werden die modifizierten Daten in den Speicher des Schlüssels geschrieben. Dieses Bit muss anschließend wieder zurückgesetzt werden. Zum Ausführen genügt eine Flanke.

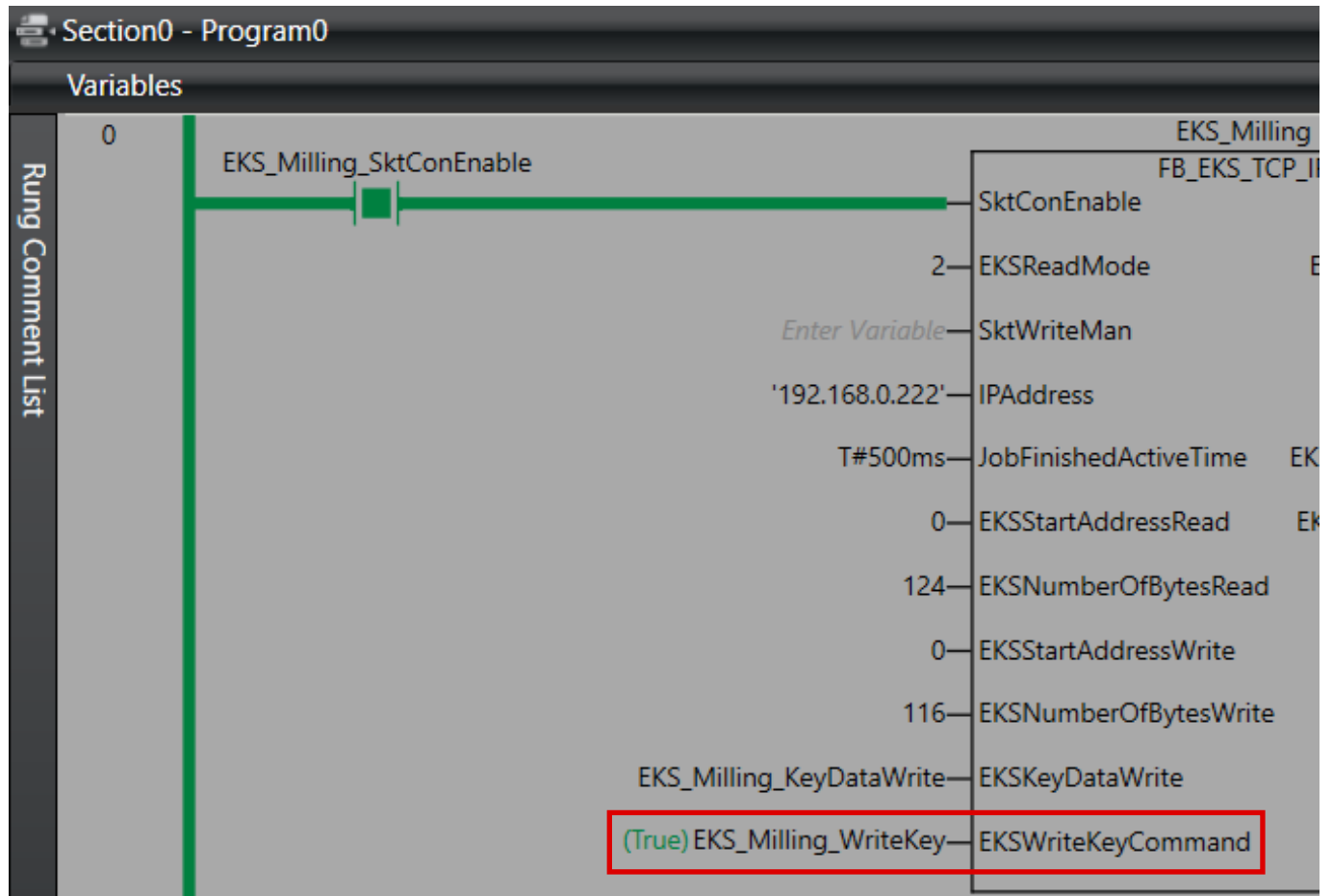


Bild 11: Kommando zum Schlüssel beschreiben

Device name	Name	Online value	Modify	Co
NX102	Program0.EKS_Milling_KeyDataRead[0..123]			
	EKS_Milling_KeyDataRead[0]	E (16#45)		
	EKS_Milling_KeyDataRead[1]	K (16#4B)		
	EKS_Milling_KeyDataRead[2]	S (16#53)		
	EKS_Milling_KeyDataRead[3]	(16#20)		
	EKS_Milling_KeyDataRead[4]	E (16#45)		
	EKS_Milling_KeyDataRead[5]	T (16#54)		
	EKS_Milling_KeyDataRead[6]	H (16#48)		
	EKS_Milling_KeyDataRead[7]	E (16#45)		
	EKS_Milling_KeyDataRead[8]	R (16#52)		
	EKS_Milling_KeyDataRead[9]	N (16#4E)		
	EKS_Milling_KeyDataRead[10]	E (16#45)		
	EKS_Milling_KeyDataRead[11]	T (16#54)		
	EKS_Milling_KeyDataRead[12]	(16#20)		
	EKS_Milling_KeyDataRead[13]	T (16#54)		
	EKS_Milling_KeyDataRead[14]	C (16#43)		
	EKS_Milling_KeyDataRead[15]	P (16#50)		
	EKS_Milling_KeyDataRead[16]	/ (16#2F)		
	EKS_Milling_KeyDataRead[17]	I (16#49)		
	EKS_Milling_KeyDataRead[18]	P (16#50)		
	EKS_Milling_KeyDataRead[19]	00		
	EKS_Milling_KeyDataRead[20]	00		

Bild 12: Beobachtungstabelle Schlüsseldaten Lesen aktualisiert

7. 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
Deutschland
info@euchner.de
www.euchner.de

Ausgabe:
AP000257-01-08/20
Titel:
Applikation EKS
Einbindung EKS mit TCP/IP Schnittstelle in OMRON Sysmac
Studio®

Copyright:
© EUCHNER GmbH + Co. KG, 08/2020

Technische Änderungen vorbehalten,
alle Angaben ohne Gewähr.