

EKS-Profinet an Siemens S7-300 – EKS Schlüssel einlesen



Inhalt

Verwendete Bauteile / Module	2
EUCHNER	2
Andere.....	2
Funktionsbeschreibung	2
Allgemein	2
Beispiel einer Schlüsselstruktur	2
Einstellung der EKS Schlüsselaufnahme.....	3
Profinet Adressierung.....	3
Schreibschutzeinstellung.....	3
Projektierung in der Steuerung	4
Hardware	4
Programmierung in der Steuerung	6
Globale Datenbausteine.....	6
AWL Programm zum Abruf des Schlüsselinhalts.....	7
Wichtiger Hinweis – Bitte unbedingt sorgfältig beachten!.....	10

Verwendete Bauteile / Module

EUCHNER

Beschreibung	Best.-Nr. / Artikelbezeichnung
EKS Profinet	106305 / EKS-A-IX-G01-ST02/03
EKS Schlüssel	077859 / EKS-A-K1RDWT32-EU 084735 / EKS-A-K1BKWT32-EU 091045 / EKS-A-K1BLWT32-EU 094839 / EKS-A-K1GNWT32-EU 094840 / EKS-A-K1YEWT32-EU

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.

Andere

Beschreibung	Artikel
S7-300, CPU 315F-2 PN/DP	6ES7315-2FJ14-0AB0

Funktionsbeschreibung

Allgemein

Das EKS wird über das Profinet an einer Siemens S7-300 SPS angeschlossen. Es sollen alle Daten entsprechend der folgenden Datenstruktur ausgelesen werden.

Beispiel einer Schlüsselstruktur

Die Daten auf dem Schlüssel sind wie folgt strukturiert:

Bytenr.	Beschreibung	Typ	Länge	Erläuterung
103 – 104	KEYCRC	CRC	2 Byte	Prüfsumme über einen bestimmten Teil des Schlüssels als Kopierschutz. Nähere Erläuterungen zur CRC siehe EKM Handbuch.
105 – 112	Verfallsdatum	Date	8 Byte	Verfallsdatum des Schlüssels.
113 – 114	Berechtigungsstufe	Word	2 Byte	Autorisierungsstufe für Zugriff auf die Maschine.
115	Abteilung	Byte	1 Byte	Nummer, die eine begrenzte Menge an Maschinen oder Anlagen beschreibt.
116 – 123	KeyID	KeyID	8 Byte	Die KeyID ist eine von EUCHNER fest programmierte Nummer auf dem Schlüssel. Diese Nummer ist bei jedem Schlüssel unterschiedlich. Diese Nummer kann zur Werkeridentifizierung herangezogen werden.

Die Struktur entspricht dem Applikationsbeispiel AP000169-2...

Einstellung der EKS Schlüsselaufnahme

Profinet Adressierung

Das Gerät soll durch die Siemens Konfigurationssoftware Simatic Manager die Adresse erhalten. Dementsprechend werden am DIP Schalter S2 alle Schalter auf OFF gestellt.

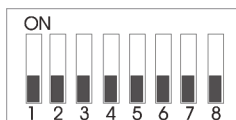


Bild 1

Das Gerät soll durch die Siemens Konfigurationssoftware Simatic Manager den DCP Namen erhalten. Dementsprechend werden am DIP Schalter S3 alle Schalter auf OFF gestellt.



Bild 2

Schreibschutzeinstellung

Das Gerät wird nur zum Lesen konfiguriert. Dementsprechend wird der DIP Schalter S1.1 auf ON gestellt.

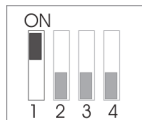


Bild 3

Projektierung in der Steuerung

Hardware

Für die Projektierung wird der Simatic Manager Version 5.5+SP1 eingesetzt. Um das EKS am Profinet zu parametrieren, ziehen Sie das Objekt „EKS-A-IX-G01-ST02/03“ auf das Profinet. Der Adressbereich kann auf 256 bis 383 eingestellt bleiben.

Wenn ein neuer Schlüssel gesteckt wird, werden die Daten ab Byte 0 immer automatisch gelesen. Da in diesem Beispiel die Nutzdaten nicht am Anfang des Schlüssels sondern am Ende stehen, werden die eigentlichen Nutzdaten eingeschränkt, jedoch muss trotz dessen immer ein 128 Byte großer Bereich im Eingang zur Verfügung gestellt werden.

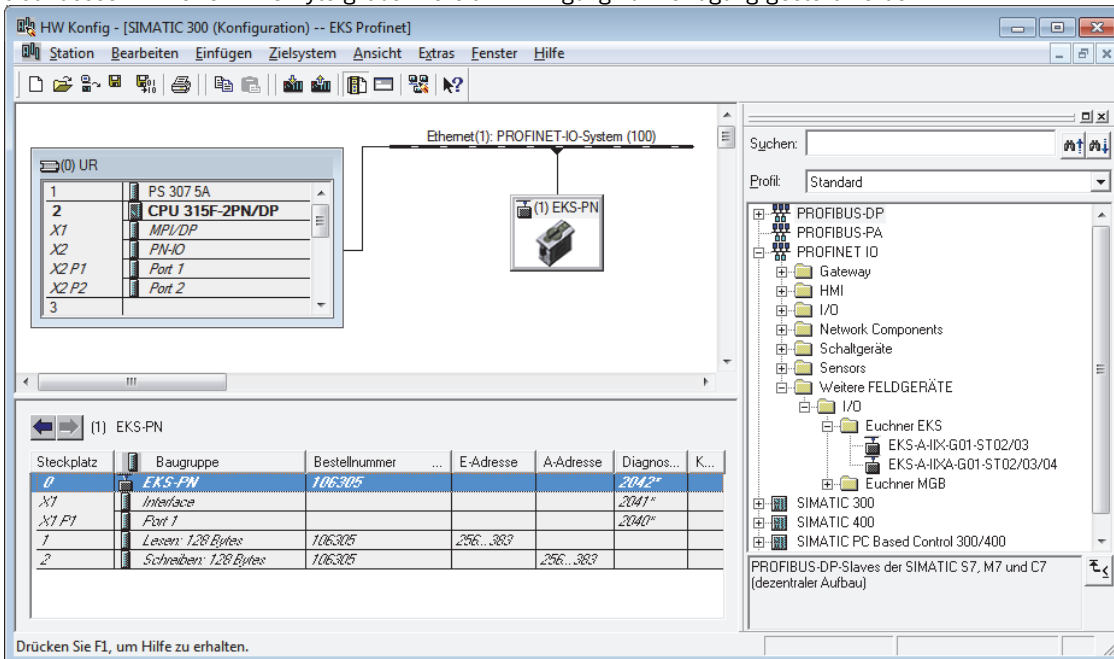


Bild 4

In den Eigenschaften des EKS Profinet stellen Sie den DCP-Namen sowie die Gerätenummer ein. In diesem Beispiel wird der Defaultname EKS-PN sowie die Gerätenummer 1 verwendet.

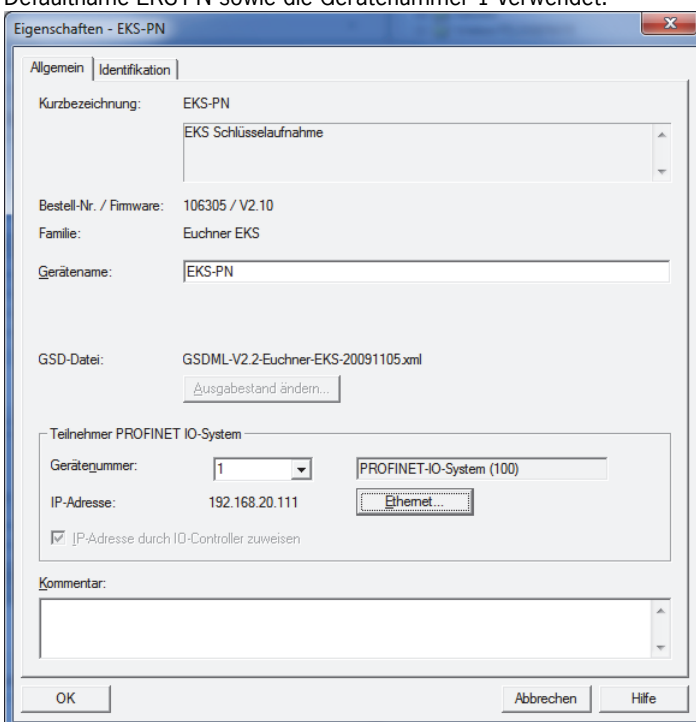


Bild 5

Im Slot „Interface“ des EKS Profinet muss in den Objekteigenschaften unter dem Reiter „IO-Zyklus“ eine Zykluszeit von mehr als 128 ms eingestellt werden.

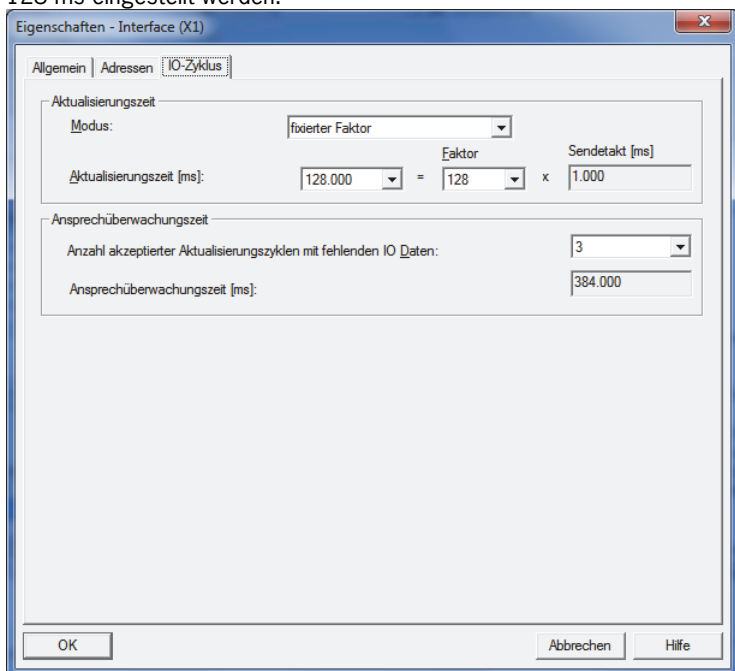


Bild 6

Auf die Alarmeinstellungen soll in diesem Beispiel nicht eingegangen werden.

Im Slot „Lesen: 128 Bytes,“ wird unter den Objekteigenschaften im Reiter Parameter eingestellt, dass die Nutzdaten ab Start-Adresse 103 mit einer Länge von 21 Byte abgeholt werden sollen. Die 21 Bytes setzen sich aus den Nutzdaten mit der Länge 13 Bytes sowie der KeyID mit der Länge 8 Bytes zusammen, werden also gemeinsam abgeholt und im Eingangsbereich ab dem Datenbyte 1 abgelegt. Im Datenbyte 0 des Eingangsbereichs liegt das Statuswort des EKS Profinet.

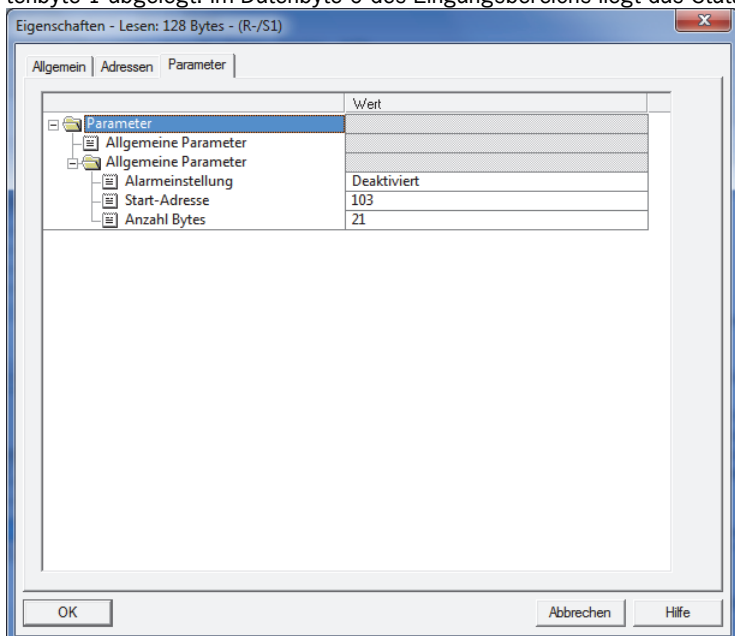


Bild 7

Programmierung in der Steuerung

Globale Datenbausteine

Es wird ein Datenbaustein angelegt, in dem die Empfangsdaten für das EKS abgelegt werden.

Im Datenbaustein für das Lesen sind die Daten strukturiert angelegt, wobei alle Daten mit mehr als einem Byte als Einzelbytes angelegt sind, um das geradzahlige Alignment in der Steuerung zu umgehen. Der Datenbaustein muss dieselbe Länge wie der Eingangsbereich des EKS haben, da andernfalls die Systemfunktion zum Lesen nicht funktioniert.

DB1, ReadBufferEKS

Adresse	Name	Typ	Anfangswert	Kommentar
0.0		STRUCT		
+0.0	ReadEKSStatus	BYTE	B#16#0	Statusbyte vom EKS
+1.0	ReadKeyCount	BYTE	B#16#0	Zähler für gesteckte Schlüssel
+2.0	ReadStartAddress	BYTE	B#16#0	Erstes gelesenes Byte
+3.0	ReadNumberBytes	BYTE	B#16#0	Anzahl der gelesenen Bytes
+4.0	ReadCRC_00	BYTE	B#16#0	CRC Byte 0
+5.0	ReadCRC_01	BYTE	B#16#0	CRC Byte 1
+6.0	ReadDate_00	BYTE	B#16#0	Datum Byte 0
+7.0	ReadDate_01	BYTE	B#16#0	Datum Byte 1
+8.0	ReadDate_02	BYTE	B#16#0	Datum Byte 2
+9.0	ReadDate_03	BYTE	B#16#0	Datum Byte 3
+10.0	ReadDate_04	BYTE	B#16#0	Datum Byte 4
+11.0	ReadDate_05	BYTE	B#16#0	Datum Byte 5
+12.0	ReadDate_06	BYTE	B#16#0	Datum Byte 6
+13.0	ReadDate_07	BYTE	B#16#0	Datum Byte 7
+14.0	ReadAuthorization_00	BYTE	B#16#0	Berechtigungsstufe Byte 0
+15.0	ReadAuthorization_01	BYTE	B#16#0	Berechtigungsstufe Byte 1
+16.0	ReadDepartment	BYTE	B#16#0	Abteilungsnummer
+17.0	ReadKeyID_00	BYTE	B#16#0	KeyID Byte 0
+18.0	ReadKeyID_01	BYTE	B#16#0	KeyID Byte 1
+19.0	ReadKeyID_02	BYTE	B#16#0	KeyID Byte 2
+20.0	ReadKeyID_03	BYTE	B#16#0	KeyID Byte 3
+21.0	ReadKeyID_04	BYTE	B#16#0	KeyID Byte 4
+22.0	ReadKeyID_05	BYTE	B#16#0	KeyID Byte 5
+23.0	ReadKeyID_06	BYTE	B#16#0	KeyID Byte 6
+24.0	ReadKeyID_07	BYTE	B#16#0	KeyID Byte 7
+26.0	Buffer	ARRAY[0..5]		Empfangspuffer auf 32 Byte füllen
+1.0		BYTE		
=32.0		END_STRUCT		

Bild 8

DB10, Instanzenbaustein für FB1

Da der Funktionsbaustein FB1 mit statischen Variablen arbeitet, muss ein DB als Instanzenbaustein verwendet werden. Im Beispiel wird dazu DB10 angelegt.

AWL Programm zum Abruf des Schlüsselinhalts

Das Leseprogramm ist in diesem Beispiel im FB1 programmiert. Das Programm liest nur dann, wenn ein Schlüssel gesteckt ist und neue Daten bereit stehen. Ein bereits gelesener Schlüssel wird kein zweites Mal eingelesen. Es werden die Daten ab Byte 103 (KeyCRC) inklusive der KeyID gelesen und im Datenbaustein DB1 ab Byte 1 zur weiteren Bearbeitung zur Verfügung gestellt. Insgesamt werden 21 Bytes Nutzdaten vom EKS Schlüssel abgeholt.

Im Byte 0 DB1 wird das Statusbyte vom EKS gespeichert.

Beschreibung der Schnittstelle

Eingabedaten

Keine.

Ausgabedaten

Fehlermeldung, neuer Schlüssel und Status des DP-Slaves.

Ein- Ausgabedaten

Keine.

Statische Daten

Der Merker KeyRead wird statisch angelegt. Dieser Merker kennzeichnet, ob ein Schlüssel bereits einmal vollständig gelesen wurde. Es werden nur dann Daten abgeholt, wenn der Merker nicht gesetzt ist. Der Merker wird immer dann zurückgesetzt, wenn sich kein Schlüssel mehr im EKS befindet

Temporäre Daten

Keine

Name	Datentyp	Adresse	Anfangswert	Kommentar
IN		0.0		
OUT		0.0		
Error	Bool	0.0	FALSE	Fehlermeldung
NewKey	Bool	0.1	FALSE	Neuer Schlüssel eingelesen
DPStatus	Word	2.0	W#16#0	Status des DP Slaves
IN_OUT		0.0		
STAT		0.0		
KeyRead	Bool	4.0	FALSE	Kennung, ob Schlüssel gelesen wurde
TEMP		0.0		

Bild 9

Veränderte Register

A1, A2, SW

Unveränderte Register

AR1, AR2, DBR1, DBR2

Verwendete Systemfunktionen

SFC14, DPRD_DAT – DP-Normslaves/PROFINET IO-Devices lesen

Globale Daten

Der Datenbaustein DB1 wird mit einer Mindestgröße von 128 Byte vorausgesetzt.

Der Inhalt des Datenbausteins DB1 wird vollständig überschrieben.

Symboltabelle

	Status	Symbol /	Adresse	Datentyp	Kommentar
1		Calculate CRC	FB 2	FB 2	
2		COMPLETE RESTART	OB 100	OB 100	Complete Restart
3		Data FB1	DB 10	FB 1	
4		Data FB2	DB 11	FB 2	
5		DIS_AIRT	SFC 41	SFC 41	Delay the Higher Priority Interrupts and Asynchronous Errors
6		DPRD_DAT	SFC 14	SFC 14	Read Consistent Data of a Standard DP Slave
7		DPWR_DAT	SFC 15	SFC 15	Write Consistent Data to a Standard DP Slave
8		EKSIn	E 256.1	BOOL	Kennung, ob ein Schlüssel gesteckt ist
9		EKSMemIn	EB 256	BYTE	Erstes Byte des EKS Eingangsspeichers
10		EKSMemOut	AB 256	BYTE	Erstes Byte des EKS Ausgangsspeichers
11		EN_AIRT	SFC 42	SFC 42	Enable Higher Priority Interrupts and Asynchronous Errors
12		F_CTRL_1	FB 273	FB 273	
13		F_CTRL_2	FB 274	FB 274	F_: Test Block an Programm Run Control
14		F_DIAG_N	FB 275	FB 275	F_: Diagnosticbuffer Message with non CPU-Stop
15		F_GLOBDB	DB 545	DB 545	F_: F_Global_Data Block
16		F_IO_CGP	FB 272	FB 272	F_: Driver Block In-Output with Channel Granular Passivation
17		Globaler Speicher	DB 3	DB 3	
18		VO_FLT1	OB 82	OB 82	VO Point Fault 1
19		Main Program	OB 1	OB 1	
20		PROG_ERR	OB 121	OB 121	Programming Error
21		RDSYSST	SFC 51	SFC 51	Read a System Status List or Partial List
22		Read EKS	FB 1	FB 1	
23		ReadBufferEKS	DB 1	DB 1	
24		STP	SFC 46	SFC 46	Change the CPU to STOP
25		VAT_1	VAT 1		
26		VAT_2	VAT 2		
27		WriteBufferEKS	DB 2	DB 2	

Bild 10

AWL Programm im FB1- ReadEKS

```
//Abholen der Daten vom EKS Schlüssel

// Prüfen, ob ein Schlüssel gesteckt wurde und nur in diesem Fall Daten abholen
U   "EKSIn"           // Prüfe, ob ein Schlüssel gesteckt ist
SPB MKEY             // Wenn gesteckt, prüfen, ob bereits Daten abgeholt wurden
R   #KeyRead         // Kennzeichnen, das nun ein neuer Schlüssel kopmmen kann
SPA MRET

// Schlüssel ist gesteckt
// Prüfen, ob es sich um einen neuen Schlüssel handelt
#KEY: U   #KeyRead     // Wenn KeyRead gesetzt ist, wurde dieser Schlüssel bereits gelesen
        SPBN MRD       // Nur mit neuem Schlüssel auch Daten holen

// Kein neuer Schlüssel gesteckt, kein Fehler
MRET: R   #Error      // Rückmeldung, kein Fehler
        R   #NewKey   // Rückmeldung, kein Schlüssel
        BE
```

Bild 11a

```
// Lesen der Daten vom EKS in den DB1
MRD: CALL "DPRD_DAT" // Aufruf SFC 14 DPRD_DAT
        LADDR :=#W#16#100 // Adresse des EKS Speicherbereichs
        RET_VAL:=MW1 // Rückmeldung
        RECORD :=#P#DB1.DBX0.0 BYTE 128 // Anfangsadresse des DB,in dem empfangen werden soll, Länge muss 32 sein

// Prüfe, ob ein Fehler aufgetreten ist
L   MW 1
L   0 // Nur Rückgabewert 0 ist ok
==I
SPBN MERR // wenn im Merkerwort 1 ein Wert <> 0 zurück gegeben wurde: Fehler

// Schlüssel vollständig gelesen, die Daten stehen jetzt im DB1
S   #KeyRead // Merken, dass mit diesem Zählerstand vollständig gelesen wurde
S   #NewKey // Zurück melden, dass ein neuer Schlüssel komplett gelesen wurde
R   #Error // Keine Fehler
BE
```

Bild11b


```
// Fehlerbearbeitung
MERR: L   MW   1           // DP Status als Rückmeldung Im Fehlerfall
      T   #DPStatus
      S   #Error           // Rückgabewert = 1, Fehler aufgetreten
      R   #NewKey
      BE
```

Bild 11c

Aufruf von FB1

```
//Abholen der Daten vom EKS Schlüssel
CALL "Read EKS" , "Data FB1"
      Error   :=M0.0           // Rückgabewert für Fehler
      NewKey  :=M0.1           // Rückgabewert, ob neuer Schlüssel
      DPStatus:=#Status       // Status des DP-Slaves

      U   M   0.0           // Prüfe ob Fehler aufgetreten ist
      SPB MERR           // Wenn Werte = 1 springe zur Fehlerroutine
```

Bild 12

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 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 Risikoanalyse 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.