

EKS an Siemens S7-300 – CRC prüfen



Inhalt

Verwendete Bauteile / Module	2
EUCHNER	2
Andere.....	2
Funktionsbeschreibung	2
Allgemein	2
Beispiel einer Schlüsselstruktur	2
CRC Festlegung.....	2
Programmierung in der Steuerung	3
Globaler Datenbaustein	3
AWL Programm zur Berechnung der CRC	4
Wichtiger Hinweis – Bitte unbedingt sorgfältig beachten!	7

Verwendete Bauteile / Module

EUCHNER

Beschreibung	Best.-Nr. / Artikelbezeichnung
EKS Profibus	084800 / EKS-A-IDX-G01-ST09/03
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 den Profibus an einer Siemens S7-300 SPS angeschlossen. Es soll die CRC über den Schlüsselinhalt berechnet werden. Dieses Programm basiert darauf, dass die Daten vom Schlüssel bereits in einem Datenbaustein eingelesen wurden. Ein Beispiel zum Lesen der Daten finden Sie in AP000169-3... für ein Profibus EKS sowie in AP000169-4... für ein Profinet EKS. Beachten Sie, dass der Datenbaustein DB1 bei den beiden Versionen unterschiedlich aufgebaut ist.

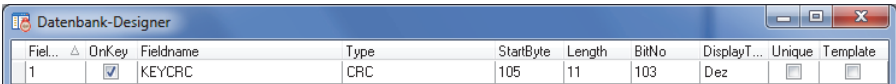
Beispiel einer Schlüsselstruktur

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.

CRC Festlegung

Die KeyCRC ist im EKM wie folgt projektiert. Die Berechnung ist vom Byte 105 ausgehend und 11 Byte lang. Die CRC steht ab Byte 103 auf dem Schlüssel.



Fiel...	OnKey	Fieldname	Type	StartByte	Length	BitNo	DisplayT...	Unique	Template
1	☒	KEYCRC	CRC	105	11	103	Dez	☐	☐

Bild 1

Programmierung in der Steuerung

Globaler Datenbaustein

Es wird ein globaler Datenbaustein angelegt, in dem bereits der Inhalt des Schlüssel stehen muss, wenn die Prüfung der CRC aufgerufen wird.

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.

DB1, ReadBufferEKS

Der in Bild 2 abgebildete Datenbaustein passt zum Beispiel AP000169-3... , in dem das EKS mit Profibus verwendet wird.

Mit einem Profinet EKS ist der Datenbaustein DB1 etwas anders aufgebaut. Die Bytes 1 bis 3 werden bei Profinet nicht genutzt (ReadKeyCount, ReadStartAddress, ReadNumberBytes). Die entsprechenden Zeilen werden im DB1 für das EKS Profinet weggelassen.

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 2

DB11, Instanzenbaustein für FB2

Damit der Funktionsbaustein FB2 um statische Variable ergänzt werden kann, wird ein DB als Instanzenbaustein verwendet. Im Beispiel wird dazu DB11 angelegt. Beispielsweise kann die weitere Auswertung des Schlüssels direkt nach positiver CRC Überprüfung programmiert werden.

AWL Programm zur Berechnung der CRC

Das Berechnungsprogramm ist in diesem Beispiel im FB2 programmiert. Da die Anzahl der Daten, über die die CRC berechnet werden soll, ungerade ist, muss die Prüfsumme in 3 Schritten berechnet werden. Im ersten Schritt werden alle Worte gelesen und berechnet, die komplett in den Nutzdaten stehen. Im zweiten Schritt wird dann das letzte Byte der Nutzdaten gelesen und mit einer 0 ergänzt. Im letzten Schritt wird dann die KeyID zur CRC dazu gerechnet. Das Lesen der Daten erfolgt aufgrund der Anordnung im Little Endian Format byteweise. Das erste Byte wird jeweils um 8 Bit nach links geschoben und dann das zweite Byte gelesen.

Tipp: Wenn die zu berechnende CRC über einen geradzahligen Nutzdatenbereich berechnet werden soll, kann die gesamte Berechnung in nur einem einzigen Schritt erfolgen, sofern die Daten aufeinanderfolgend liegen.

Beschreibung der Schnittstelle

Eingabedaten

Keine.

Ausgabedaten

Meldung ob CRC korrekt ist.

Ein- Ausgabedaten

Keine.

Statische Daten

Keine.

Temporäre Daten

Zähler für die Schleifen in Schritt 1 und 3 und berechneter Wert der CRC.

Name	Datentyp	Adresse	Anfangswert	Kommentar
IN		0.0		
OUT		0.0		
CRC_ok	Bool	0.0	FALSE	CRC ist identisch
IN_OUT		0.0		
STAT		0.0		
TEMP		0.0		
TCounter	Int	0.0		Zähler
CRC	Word	2.0		Berechnungswert der CRC

Bild 3

Veränderte Register

A1, A2, SW, AR1, DBR1

Unveränderte Register

AR2, DBR2

Verwendete Systemfunktionen

Keine.

Globale Daten

Der Datenbaustein DB1 muss die Daten des bereits gelesenen EKS Schlüssels enthalten. Der Inhalt des Datenbausteins DB1 wird nicht geändert.

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		EKSInCount	EB 257	BYTE	Zählerstand des EKS
10		EKSMemIn	EB 256	BYTE	Erstes Byte des EKS Eingangsspeichers
11		EKSMemOut	AB 256	BYTE	Erstes Byte des EKS Ausgangsspeichers
12		EN_AIRT	SFC 42	SFC 42	Enable Higher Priority Interrupts and Asynchronous Errors
13		F_CTRL_1	FB 273	FB 273	
14		F_CTRL_2	FB 274	FB 274	F.: Test Block an Programm Run Control
15		F_DIAG_N	FB 275	FB 275	F.: Diagnosticbuffer Message with non CPU-Stop
16		F_GLOBDB	DB 545	DB 545	F.: F_Global_Data Block
17		F_IO_CGP	FB 272	FB 272	F.: Driver Block In-Output with Channel Granular Passivation
18		Globaler Speicher	DB 3	DB 3	
19		VO_FLT1	OB 82	OB 82	VO Point Fault 1
20		Main Program	OB 1	OB 1	
21		PROG_ERR	OB 121	OB 121	Programming Error
22		RDSYST	SFC 51	SFC 51	Read a System Status List or Partial List
23		Read EKS	FB 1	FB 1	
24		ReadBufferEKS	DB 1	DB 1	
25		STP	SFC 46	SFC 46	Change the CPU to STOP
26		VAT_1	VAT 1		
27		VAT_2	VAT 2		
28		WriteBufferEKS	DB 2	DB 2	

Bild 4

AWL Programm im FB2 – Calculate CRC

Das abgebildete Programm Bild 5a passt zum Beispiel AP000169-3... , in dem das EKS mit Profibus verwendet wird.

Da mit einem EKS Profinet der Datenbaustein DB1 etwas anders aufgebaut ist, muss statt Byte 6 das Byte 3 als Start genutzt werden. Beim Profinet EKS fehlen 3 Statusbytes, die das EKS Profibus sendet, deshalb wird der Datenbereich um 3 Bytes nach vorn geschoben.

Die entsprechende Programmzeile lautet dann:

LAR1 P#DBX3.0

```
// Die CRC wird passend für die vorgegebene Schlüsselstruktur berechnet
// Es wird in 3 Schritten berechnet, die je nach Anordnung der Daten nicht immer notwendig sind
// Nur wenn eine ungerade Anzahl an utzdaten ohne KeyId gelsen wird, muss in 3 Schritten gerechnet werden
// Im ersten Schritt werden alle Worte, bis auf das letzte Byte berechnet
// Im zweiten Schritt wird das einzelne Byte, verlängert um eine eingeschobene 0 berechnet
// Im dritten Schritt wird die KeyID dazu berechnt

// Berechnung des ersten Teils über die Nutzdaten auf dem Schlüssel
L      5                      // Anzahl der Worte im ersten Schritt
T      #TCounter              // Zähler für Aufsummierung initiieren
L      0                      // CRC Berechnungswert initiieren
T      #CRC
AUF     "ReadBufferEKS"       // Daten aus dem DB1 lesen
LAR1    P#DBX 6.0             // Beginn ist ab dem Word hinter der CRC
LOP1:  L      B [AR1,P#1.0]    // Nächstes Wort laden (Offset 1)
        SLW     8
        L      B [AR1,P#0.0]
        +I
        L      #CRC           // zur bisherigen CRC addieren
        +I
        T      #CRC           // neu berechnete CRC
        L      P#2.0           // Zeiger um 2 Byte erhöhen
        +AR1                  // Adressregister
        L      #TCounter       // Zähler um 1 erniedrigen
        DEC     1
        T      #TCounter
        L      0              // Ende der Schleife, wenn Wert 0 erreicht
        ==I
        SPBN  LOP1
```

Bild 5a

```
// Spezielle Berechnung, da eine ungerade Anzahl an Bytes auf dem Schlüssel in der CRC steckt
L   B [AR1,P#0.0]           // Nächstes Wort laden, da eine 0 eingeschoben wird, ist es nur das Byte
L   #CRC                    // zur bisherigen CRC addieren
+I
T   #CRC                    // neu berechnete CRC
L   P#1.0                   // Zeiger um 1 Byte erhöhen
+AR1                          // Adressregister
```

Bild 5b

```
// Berechnen über die KeyID
L   4                        // Anzahl der Worte, nur KeyID
T   #TCounter               // Zähler für Aufsummierung initiieren

LOP2: L   B [AR1,P#1.0]      // Nächstes Wort laden
      SLW 8
      L   B [AR1,P#0.0]
      +I
      L   #CRC              // zur bisherigen CRC addieren
      +I
      T   #CRC              // neu berechnete CRC
      L   P#2.0             // Zeiger um 2 Byte erhöhen
      +AR1                  // Adressregister
      L   #TCounter         // Zähler um 1 erniedrigen
      DEC 1
      T   #TCounter
      L   0                 // Ende der Schleife, wenn Wert 0 erreicht
      ==I
      SPBN LOP2
```

Bild 5c

Das abgebildete Programm Bild 5d passt zum Beispiel AP000169-3... , in dem das EKS mit Profibus verwendet wird. Da mit einem EKS Profinet der Datenbaustein DB1 etwas anders aufgebaut ist, muss statt des Byte 4 das Byte 1 als Start genutzt werden. Beim Profinet EKS fehlen 3 Statusbytes, die das EKS Profibus sendet, deshalb wird der Datenbereich um 3 Bytes nach vorn geschoben.

Die entsprechende Programmzeile lautet unverändert, wenn die Symboltabelle aufgebaut ist.

```
// Vergleiche gegen die CRC, die auf dem Schlüssel steht
L   "ReadBufferEKS".ReadCRC_01 // ab Byte 4 steht die CRC
SLW 8
L   "ReadBufferEKS".ReadCRC_00 //
+I
L   #CRC                      // Vergleiche berechnete CRC mit der vom Schlüssel
==I
=   #CRC_ok                   // Rückgabewert setzen
BE
```

Bild 5d

Aufruf von FB2

Die Berechnung wird nur aufgerufen, wenn ein gültiger neuer Schlüsselinhalt zur Verfügung gestellt wurde, bspw. von einem EKS Leseprogramm wie in Applikation AP000169-3... Dies ist gekennzeichnet durch einen gesetzten Merker M0.1.

```
// Berechne die Schlüssel CRC
U   M   0.1                  // Prüfe ob neuer Schlüssel
SPBN M001
CALL "Calculate CRC" , "Data FB2"
CRC_ok:=M0.0

M001: BE                    // Ende Lesen EKS
```

Bild 6

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.