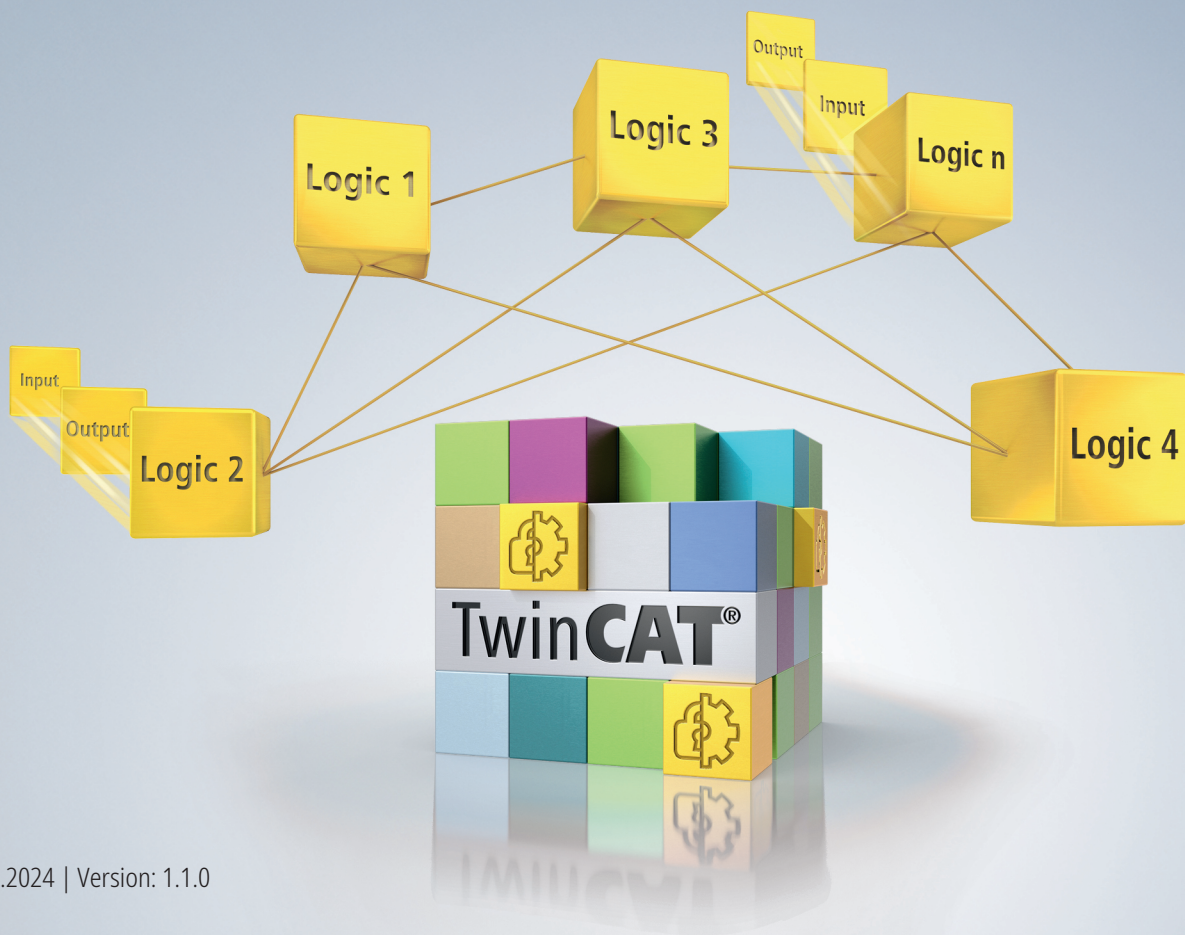


Originaldokumentation | DE

Import und Export von XML-Dateien

TwinCAT 3 Safety Editor



Inhaltsverzeichnis

1	Dokumentationshinweise	5
1.1	Disclaimer.....	5
1.1.1	Marken	5
1.1.2	Patente.....	5
1.1.3	Haftungsbeschränkungen	6
1.1.4	Copyright.....	6
1.2	Ausgabestände der Dokumentation.....	7
1.3	Referenzen.....	7
1.4	Personalqualifikation	8
1.5	Sicherheit und Einweisung.....	9
1.5.1	Symbolerklärung	9
1.6	Support und Service.....	10
1.7	Hinweise zur Informationssicherheit	11
2	Abkürzungsverzeichnis	12
3	Zu Ihrer Sicherheit	13
3.1	Sorgfaltspflicht.....	13
3.2	Allgemeine Sicherheitshinweise.....	13
4	Schnittstellenbeschreibung	14
5	Systembeschreibung	15
5.1	Allgemeines.....	15
5.2	Systemvoraussetzungen.....	15
5.3	Systemgrenzen	15
5.4	Bestimmungsgemäße Verwendung.....	16
6	Funktionsweise	17
6.1	Manuell.....	17
6.1.1	Export.....	17
6.1.2	Import.....	18
6.2	Automation Interface	19
6.2.1	Export.....	19
6.2.2	Import.....	20
7	XSD-Datei.....	21
7.1	TwinCATExport	21
7.1.1	TwinCATExport Attribut.....	21
7.1.2	BaseConfiguration.....	22
7.1.3	Multi Settings.....	70
7.2	ID-Regeln	75
7.3	FBTypes.....	77
7.4	PortIds.....	78
7.4.1	FBAnd (SubType 0x22).....	78
7.4.2	FBOr (SubType 0x23).....	79
7.4.3	FBDecouple (SubType 0x24).....	79
7.4.4	FBOpmode (SubType 0x25).....	79
7.4.5	FBRs (SubType 0x29).....	80

7.4.6	FBSr (SubType 0x2a)	80
7.4.7	FBTof (SubType 0x2b).....	80
7.4.8	FBTon (SubType 0x2c).....	80
7.4.9	FBTon2 (SubType 0x1132).....	80
7.4.10	FBEdm (SubType 0x27).....	81
7.4.11	FBEstop (SubType 0x20).....	81
7.4.12	FBMon (SubType 0x21).....	81
7.4.13	FBTwohand (SubType 0x26)	82
7.4.14	FBMuting (SubType 0x28)	82
7.4.15	FBConnShutdown (SubType 0x2d)	82
7.4.16	FBXor (SubType 0x2e)	82
7.4.17	FBAdd (SubType 0x41).....	83
7.4.18	FBSub (SubType 0x42).....	83
7.4.19	FBMul (SubType 0x43)	83
7.4.20	FBDiv (SubType 0x44).....	84
7.4.21	FBCounter (SubType 0x45)	84
7.4.22	FBLimit (SubType 0x46)	84
7.4.23	FBCompare (SubType 0x40)	84
7.4.24	FBScale (SubType 0x47).....	84
7.4.25	FBCamMonitor (SubType 0x48)	85
7.4.26	FBLoadSensing (SubType 0x4a).....	85
7.4.27	FBSpeed (SubType 0x49).....	85
7.4.28	FBViolationCnt (SubType 0x4b).....	86
7.4.29	FBSLI (SubType 0x4c).....	86
7.4.30	FBSLI2 (SubType 0x1db2).....	86
7.4.31	FBEnvelope (SubType 0x4d).....	86
7.4.32	FBSLP (SubType 0x4e).....	86
7.4.33	FBSBT (SubType 0x4f).....	87
7.4.34	FBAdvPosMon (SubType 0x50).....	87
7.5	ConnectionType	88
7.5.1	ConnectionType in TE9000.....	88
7.6	LogicType.....	89
7.6.1	LogicType in TE9000	90
8	Kompatibles Safety-Projekt generieren	91
8.1	Beispiel EStop-Funktion.....	91
8.1.1	BaseConfiguration.....	91
8.2	XSD-Validierung.....	101
9	Automation Interface testen.....	102
9.1	Manueller Export.....	102
9.2	Manueller Import	103
10	FMEDA.....	104

1 Dokumentationshinweise

1.1 Disclaimer

Beckhoff Produkte werden fortlaufend weiterentwickelt. Wir behalten uns vor, die Betriebsanleitung jederzeit und ohne Ankündigung zu überarbeiten. Aus den Angaben, Abbildungen und Beschreibungen in dieser Betriebsanleitung können keine Ansprüche auf Änderung bereits gelieferter Produkte geltend gemacht werden.

Wir definieren in dieser Betriebsanleitung alle zulässigen Anwendungsfälle, deren Eigenschaften und Betriebsbedingungen wir zusichern können. Die von uns definierten Anwendungsfälle sind vollumfänglich geprüft und zertifiziert. Darüberhinausgehende Anwendungsfälle, die nicht in dieser Betriebsanleitung beschrieben werden, bedürfen eine Prüfung der Firma Beckhoff Automation GmbH & Co. KG.

1.1.1 Marken

Beckhoff®, TwinCAT®, TwinCAT/BSD®, TC/BSD®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, Safety over EtherCAT®, TwinSAFE®, XFC®, XTS® und XPlanar® sind eingetragene und lizenzierte Marken der Beckhoff Automation GmbH.

Die Verwendung anderer Marken oder Kennzeichen durch Dritte kann zu einer Verletzung von Rechten der Inhaber der entsprechenden Bezeichnungen führen.

1.1.2 Patente

Die EtherCAT-Technologie ist patentrechtlich durch folgende Anmeldungen und Patente mit den entsprechenden Anmeldungen und Eintragungen in verschiedenen anderen Ländern geschützt:

- EP1590927
- EP1789857
- EP1456722
- EP2137893
- DE102015105702

EtherCAT® 

EtherCAT® ist eine eingetragene Marke und patentierte Technologie, lizenziert durch die Beckhoff Automation GmbH.

Safety over
EtherCAT® 

Safety over EtherCAT® ist eine eingetragene Marke und patentierte Technologie, lizenziert durch die Beckhoff Automation GmbH.

1.1.3 Haftungsbeschränkungen

Die gesamten Komponenten des beschriebenen Produkts werden je nach Anwendungsbestimmungen in bestimmter Konfiguration von Hardware und Software ausgeliefert. Umbauten und Änderungen der Konfiguration von Hardware oder Software, die über die dokumentierten Möglichkeiten hinausgehen, sind verboten und führen zum Haftungsausschluss der Beckhoff Automation GmbH & Co. KG.

Folgendes wird aus der Haftung ausgeschlossen:

- Nichtbeachtung dieser Betriebsanleitung
- Nicht-bestimmungsgemäße Verwendung
- Einsatz nicht ausgebildeten Fachpersonals
- Erlöschen der Zertifizierungen
- Verwendung nicht zugelassener Ersatzteile

1.1.4 Copyright

© Beckhoff Automation GmbH & Co. KG, Deutschland.

Weitergabe sowie Vervielfältigung dieses Dokuments, Verwertung und Mitteilung seines Inhalts sind verboten, soweit nicht ausdrücklich gestattet.

Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte für den Fall der Patent-, Gebrauchsmuster- oder Geschmacksmustereintragung vorbehalten.

1.2 Ausgabestände der Dokumentation

Ausgabe	Kommentar
1.1.0	<ul style="list-style-type: none"> • Korrekturen • FBAdvPosMon hinzugefügt
1.0.0	<ul style="list-style-type: none"> • Erste freigegebene Version
0.0.1	<ul style="list-style-type: none"> • Vorläufig (nur intern)

Aktualität

Prüfen Sie, ob Sie die aktuelle und gültige Version des vorliegenden Dokumentes verwenden. Auf der Beckhoff Homepage finden Sie unter <http://www.beckhoff.com/twinsafe> die jeweils aktuelle Version zum Download. Im Zweifelsfall wenden Sie sich an den technischen Support (siehe Beckhoff Services).

Dokumentenursprung

Diese Dokumentation ist das Originalhandbuch und ist in deutscher Sprache verfasst. Alle weiteren Sprachen werden von dem deutschen Original abgeleitet.

Produkteigenschaften

Gültig sind immer die Produkteigenschaften, die in dem aktuellen Handbuch angegeben sind. Weitere Informationen, die auf den Produktseiten der Beckhoff Homepage, in E-Mails oder sonstigen Publikationen angegeben werden, sind nicht maßgeblich.

1.3 Referenzen

Nr.	Ausgabe	Titel / Beschreibung
[1]	1.4 oder neuer	<p>Handbuch Automation Interface TwinCAT 3</p> <p>Das Dokument beschreibt die Grundlagen, das Vorgehen und die Methoden des Automation Interface in TwinCAT 3.</p>
[2]	1.0 oder neuer	<p>Beckhoff ELx9xx ESI-Datei</p> <p>Das Dokument enthält eine XML-Beschreibung über EtherCAT Slaves der Gruppe ELx9xx.</p>
[3]	1.4.1 oder neuer	<p>Betriebsanleitung zu EL6910 TwinSAFE-Logic-Modul</p> <p>Das Dokument enthält eine Beschreibung der Logic-Funktionen der EL6910 und somit auch der AMI8911 und deren Programmierung</p>
[4]	3.1.0 oder neuer	<p>Dokumentation TwinSAFE-Logic-FB</p> <p>Das Dokument beschreibt die sicherheitstechnischen Funktionsbausteine, die in der EL6910 und somit auch der AMI8911-Optionskarte zur Verfügung stehen und die sicherheitstechnische Applikation bilden.</p>
[5]	1.8.0 oder neuer	<p>TwinSAFE Applikationshandbuch</p> <p>Das Applikationshandbuch gibt dem Anwender Beispiele für die Berechnung von Ausfallgrenzwerten für Sicherheitsfunktionen entsprechend der Normen DIN EN ISO 13849-1 und EN 62061 bzw. EN 61508:2010, wie sie typischerweise an Maschinen Verwendung finden.</p>
[6]	2006/42/EG	<p>Richtlinie 2006/42/EG des Europäischen Parlaments und des Rates vom 17. Mai 2006 über Maschinen und zur Änderung der Richtlinie 95/16-7EG (Neufassung) vom 29.06.2006</p> <p>Diese Richtlinie, auch Maschinenrichtlinie genannt, definiert Anforderungen an das Inverkehrbringen von Maschinen und maschinenähnlichen Komponenten, wie Sicherheitsbauteile.</p>

1.4 Personalqualifikation

Diese Betriebsanleitung wendet sich ausschließlich an ausgebildetes Fachpersonal der Steuerungstechnik und Automatisierung mit den dazugehörigen Kenntnissen.

Das ausgebildete Fachpersonal muss sicherstellen, dass die Anwendungen und der Einsatz des beschriebenen Produkts alle Sicherheitsanforderungen erfüllen. Dazu zählen sämtliche anwendbare und gültige Gesetze, Vorschriften, Bestimmungen und Normen.

Ausgebildetes Fachpersonal

Ausgebildetes Fachpersonal verfügt über umfangreiche fachliche Kenntnisse aus Studium, Lehre oder Fachausbildung. Verständnis für Steuerungstechnik und Automatisierung ist vorhanden. Ausgebildetes Fachpersonal kann:

- Eigenständig Gefahrenquellen erkennen, vermeiden und beseitigen
- Relevante Normen und Richtlinien anwenden
- Vorgaben aus den Unfallverhütungsvorschriften umsetzen
- Das Arbeitsumfeld beurteilen, vorbereiten und einrichten
- Arbeiten selbständig beurteilen, optimieren und ausführen

1.5 Sicherheit und Einweisung

Lesen Sie die Inhalte, welche sich auf die von Ihnen durchzuführenden Tätigkeiten mit dem Produkt beziehen. Lesen Sie immer das Kapitel Zu Ihrer Sicherheit in der Betriebsanleitung.

Beachten Sie die Warnhinweise in den Kapiteln, sodass Sie bestimmungsgemäß und sicher mit dem Produkt umgehen und arbeiten.

1.5.1 Symbolerklärung

Für eine übersichtliche Gestaltung werden verschiedene Symbole verwendet:

1. Die Nummerierung zeigt eine Handlungsanweisung, die Sie ausführen sollen.
 - Der Punkt zeigt eine Aufzählung.
- [...] Die eckigen Klammern zeigen Querverweise auf andere Textstellen in dem Dokument.
- [1] Die Zahl in eckigen Klammern zeigt die Nummerierung eines referenzierten Dokuments.

1.5.1.1 Signalwörter

Im Folgenden werden die Signalwörter eingeordnet, die in der Dokumentation verwendet werden.

Warnung vor Personenschäden

GEFAHR

Es besteht eine Gefährdung mit hohem Risikograd, die den Tod oder eine schwere Verletzung zur Folge hat.

WARNUNG

Es besteht eine Gefährdung mit mittlerem Risikograd, die den Tod oder eine schwere Verletzung zur Folge haben kann.

VORSICHT

Es besteht eine Gefährdung mit geringem Risikograd, die eine mittelschwere oder leichte Verletzung zur Folge haben kann.

Warnung vor Umwelt- oder Sachschäden

HINWEIS

Hinweise

Es besteht eine mögliche Schädigung für Umwelt, Geräte oder Daten.

Information zum Umgang mit dem Produkt



Diese Information beinhaltet z. B.:
Handlungsempfehlungen, Hilfestellungen oder weiterführende Informationen zum Produkt.

1.6 Support und Service

Beckhoff und seine weltweiten Partnerfirmen bieten einen umfassenden Support und Service, der eine schnelle und kompetente Unterstützung bei allen Fragen zu Beckhoff Produkten und Systemlösungen zur Verfügung stellt.

Downloadfinder

Unser Downloadfinder beinhaltet alle Dateien, die wir Ihnen zum Herunterladen anbieten. Sie finden dort Applikationsberichte, technische Dokumentationen, technische Zeichnungen, Konfigurationsdateien und vieles mehr.

Die Downloads sind in verschiedenen Formaten erhältlich.

Beckhoff Niederlassungen und Vertretungen

Wenden Sie sich bitte an Ihre Beckhoff Niederlassung oder Ihre Vertretung für den lokalen Support und Service zu Beckhoff Produkten!

Die Adressen der weltweiten Beckhoff Niederlassungen und Vertretungen entnehmen Sie bitte unserer Internetseite: www.beckhoff.com

Dort finden Sie auch weitere Dokumentationen zu Beckhoff Komponenten.

Beckhoff Support

Der Support bietet Ihnen einen umfangreichen technischen Support, der Sie nicht nur bei dem Einsatz einzelner Beckhoff Produkte, sondern auch bei weiteren umfassenden Dienstleistungen unterstützt:

- Support
- Planung, Programmierung und Inbetriebnahme komplexer Automatisierungssysteme
- umfangreiches Schulungsprogramm für Beckhoff Systemkomponenten

Hotline: +49 5246 963-157

E-Mail: support@beckhoff.com

Beckhoff Service

Das Beckhoff Service-Center unterstützt Sie rund um den After-Sales-Service:

- Vor-Ort-Service
- Reparaturservice
- Ersatzteilservice
- Hotline-Service

Hotline: +49 5246 963-460

E-Mail: service@beckhoff.com

Beckhoff Unternehmenszentrale

Beckhoff Automation GmbH & Co. KG

Hülshorstweg 20
33415 Verl
Deutschland

Telefon: +49 5246 963-0

E-Mail: info@beckhoff.com

Internet: www.beckhoff.com

1.7 Hinweise zur Informationssicherheit

Die Produkte der Beckhoff Automation GmbH & Co. KG (Beckhoff) sind, sofern sie online zu erreichen sind, mit Security-Funktionen ausgestattet, die den sicheren Betrieb von Anlagen, Systemen, Maschinen und Netzwerken unterstützen. Trotz der Security-Funktionen sind die Erstellung, Implementierung und ständige Aktualisierung eines ganzheitlichen Security-Konzepts für den Betrieb notwendig, um die jeweilige Anlage, das System, die Maschine und die Netzwerke gegen Cyber-Bedrohungen zu schützen. Die von Beckhoff verkauften Produkte bilden dabei nur einen Teil des gesamtheitlichen Security-Konzepts. Der Kunde ist dafür verantwortlich, dass unbefugte Zugriffe durch Dritte auf seine Anlagen, Systeme, Maschinen und Netzwerke verhindert werden. Letztere sollten nur mit dem Unternehmensnetzwerk oder dem Internet verbunden werden, wenn entsprechende Schutzmaßnahmen eingerichtet wurden.

Zusätzlich sollten die Empfehlungen von Beckhoff zu entsprechenden Schutzmaßnahmen beachtet werden. Weiterführende Informationen über Informationssicherheit und Industrial Security finden Sie in unserem <https://www.beckhoff.de/secguide>.

Die Produkte und Lösungen von Beckhoff werden ständig weiterentwickelt. Dies betrifft auch die Security-Funktionen. Aufgrund der stetigen Weiterentwicklung empfiehlt Beckhoff ausdrücklich, die Produkte ständig auf dem aktuellen Stand zu halten und nach Bereitstellung von Updates diese auf die Produkte aufzuspielen. Die Verwendung veralteter oder nicht mehr unterstützter Produktversionen kann das Risiko von Cyber-Bedrohungen erhöhen.

Um stets über Hinweise zur Informationssicherheit zu Produkten von Beckhoff informiert zu sein, abonnieren Sie den RSS Feed unter <https://www.beckhoff.de/secinfo>.

2 Abkürzungsverzeichnis

Anbei finden Sie eine Auflistung der verwendeten Abkürzungen und ihre Bedeutung:

Abkürzung	Bedeutung
ESI	EtherCAT Slave Information
FB	Funktionsbaustein
FB-Gruppe	Funktionsbaustein-Gruppe
FSoE	FailSafe over EtherCAT
TwinSAFE SC	TwinSAFE Single Channel
User FB	Benutzer-Funktionsbaustein
User FB Libs	Bibliotheken der Benutzer-Funktionsbausteine
XSD	XML Schema Definition

3 Zu Ihrer Sicherheit

3.1 Sorgfaltspflicht

Der Betreiber muss alle die in dieser Betriebsanleitung genannten Anforderungen und Hinweise einhalten, um seiner Sorgfaltspflicht nachzukommen. Dazu zählt insbesondere, dass Sie

- die gesamte Dokumentation der TwinSAFE-Komponente lesen
- die in dem Kapitel [Haftungsbeschränkung](#) [▶ 6] definierten Bestimmungen einhalten.
- die TwinSAFE-Komponente nur in einem einwandfreien und funktionstüchtigen Zustand betreiben.
- die Betriebsanleitung in einem lesbaren Zustand und vollständig am Einsatzort der TwinSAFE-Komponente zur Verfügung stellen.

3.2 Allgemeine Sicherheitshinweise

In Maschinen nach der Maschinenrichtlinie verwenden

Setzen Sie die TwinSAFE-Komponente nur in Maschinen gemäß der Maschinenrichtlinie ein. So gewährleisten Sie einen sicheren Betrieb.

Weitere Informationen entnehmen Sie Dokument [6] unter [Referenzen](#) [▶ 7].

Bestimmungsgemäß verwenden

Eine Verwendung der TwinSAFE-Komponenten, die über die beschriebene bestimmungsgemäße Verwendung hinausgeht, ist nicht zulässig.

Sicherheitsfunktionen prüfen

Nehmen Sie bei jeder Änderung Ihres Safety-Projekts eine erneute Abnahme der Sicherheitsfunktionen durch. Dazu zählt auch das Auslesen und Prüfen der aktuell auf der TwinSAFE-Sicherheitsteuerung aktiven Parameter und Customizing-Einstellungen.

Workflow spezifizieren

Spezifizieren Sie für das Laden und Anpassen eines Safety-Projekts einen Workflow. So stellen Sie sicher, dass das für die Anwendung korrekte Safety-Projekt auf den TwinSAFE-Logik-Komponente aktiviert wird.

4 Schnittstellenbeschreibung

Dieses Dokument beschreibt das Datenformat der Import/Export-Schnittstelle des TwinCAT 3 Safety Editors.

Wenn Sie diese Beschreibung zur Erzeugung einer Import-Datei benutzen, berücksichtigen Sie die Anforderungen der einschlägigen Normen und Standards für die Entwicklung von sicherheitsrelevanter Software beziehungsweise Software-Tools. Extern (außerhalb des TwinCAT 3 Safety Editors) erzeugte Dateien müssen dem Datei-Schema entsprechen. Prüfen Sie die Datei auf Korrektheit und Fehlerfreiheit.

Greifen Sie die dazu geeignete Maßnahmen anhand der Norm IEC 61508. Nach dieser Norm handelt es sich bei dieser Schnittstelle um ein T3-Tool.

Außerdem werden wesentliche Schutzmaßnahmen des TwinCAT 3 Safety Editors bei dem Import einer externen erzeugten XML-Applikationsdatei außer Kraft gesetzt. Halten Sie daher die folgenden Kriterien ein:

- Eindeutigkeit der FB und Adressen
- Korrektheit der Verbindungen, Links und Datentypen
- Korrektheit der Parameter und Projektgröße

Das bedeutet insbesondere:

- Korrekte sichere Parameter der Verbindungen, Gruppen und FBs
- Korrekte, eindeutige und gewollte Abarbeitungsreihenfolgen von Verbindungen, Gruppen und FBs
- Größe und Aufteilung (Position, Länge, Datentyp) der Verbindungstelegramme in Einzelsignalen. Dies gilt für beide Verbindungsrichtungen.
- Im jeweiligen Kontext eindeutige Namen von Gruppen, Netzwerken, FBs, Variablen, etc.
- Verwendete Logik(typ)-Features and -grenzen
- Kein mehrfaches Beschreiben von Eingängen (FB- oder Gruppen-Eingänge, Connection- oder Standard Signal-Ausgängen)
- Korrekte User FBs
- Korrekte Referenzen auf User FB Libs
- Korrekte und eindeutige gewählte Logik und gewählter Logiktyp
- Korrekte Zuordnung von Backup/Restore-Einstellungen zu Verbindungen
- Korrekte MultiSetting-Zuordnungen zu FBs, Verbindungen und System-Manager-Knoten sowie korrekte Parametereinstellungen
- Korrekte Angabe des Prozessabbilds
- Korrektes Mapping von Logik, Verbindungen und Standard-Signalen zu System-Manager-Knoten

Die Korrektheit kann durch den TwinCAT 3 Safety Editor nicht überprüft werden.

Diese Schnittstelle wird im Rahmen der Anpassung an die Anforderungen der Cybersecurity zukünftig erweitert. Aufgrund der Rechtslage ist ein Erhalt der Schnittstelle in der jetzigen Form vorrausichtlich nicht möglich.

Erkundigen Sie sich bei Ihrem Beckhoff-Ansprechpartner bezüglich der anstehenden Modifikationen und der Roadmap.

5 Systembeschreibung

5.1 Allgemeines

In TwinCAT 3 haben Sie zwei Möglichkeiten, um ein Safety-Projekt im XML-Format zu exportieren oder importieren.

- Das TwinCAT 2 XML-Format dient dem Austausch von Safety-Projekten zwischen den Entwicklungsumgebungen TwinCAT 2 und TwinCAT 3.
- Das TwinCAT 3 XML-Format ermöglicht Ihnen die Verfügbarkeit Ihres Safety-Projekts außerhalb TwinCAT 3 im XML-Format.
Dies ermöglicht Ihnen beispielsweise das Abspeichern des Projekts in XML-Format für zum Beispiel Source Control. Außerdem bietet das XML-Schema dieser XML-Datei die Möglichkeit ein Projekt zu erzeugen.

Diese Dokumentation beschränkt sich ausschließlich auf den Export und Import im TwinCAT 3 XML Format.

5.2 Systemvoraussetzungen

Für den XML-Export oder -Import benötigen Sie die folgenden Versionen:

TwinCAT	TE9000	XML	XSD
3.1 4020.0 - 3.1 4022.0	Kein separates Setup, in TwinCAT integriert	0.24	Nicht verfügbar
3.1 4022.2 - 3.1 4024.7		0.27	
3.1 4024.10 oder neuer	1.0.0 oder neuer	0.30	
	1.4.6	0.31	0.31

Als Zielsystem eignen sich alle in TwinCAT 3 verwendbaren TwinSAFE-Komponenten mit Ausnahme TwinCAT Safety PLC.

Die aktuelle TwinCAT 3 Software finden Sie auf der Beckhoff Website unter [TwinCAT 3 download / eXtended Automation Engineering \(XAE\)](#).

Den aktuellen TwinCAT 3 Safety Editor finden Sie auf der Beckhoff Website unter [TwinCAT 3 Safety Editor](#).

5.3 Systemgrenzen

XML-Export

Für den XML-Export benötigen Sie ein valides Safety-Projekt. Nicht-valide Projekte können Sie nicht exportieren.

XML-Import

In älteren TwinCAT 3-Versionen sind gegebenenfalls noch nicht alle Funktionen verfügbar.

Falls Sie eine XML-Datei importieren, die mit einer neueren Version erstellt wurde als die TwinCAT-Version, in die Sie importieren möchten, kann es sein, dass einige Daten oder auch das gesamte Projekt nicht mit importiert werden.

Um sicherzustellen, dass alle Daten beim Import beibehalten werden, vermeiden Sie den Import in eine ältere TwinCAT-Version.

5.4 Bestimmungsgemäße Verwendung

Betreiben Sie die TwinSAFE-Komponente ausschließlich für die vorgesehenen und in dieser Dokumentation definierten Tätigkeiten unter Berücksichtigung der vorgeschriebenen Werte.

Durch die Funktionen ist der Einsatz der Beckhoff-TwinSAFE-Komponenten im Bereich der Maschinensicherheit erlaubt. Das angestrebte Einsatzgebiet der TwinSAFE-Komponenten sind Sicherheitsfunktionen an Maschinen und die damit unmittelbar zusammenhängenden Aufgaben in der industriellen Automatisierung.

TwinSAFE-Komponenten sind daher nur für Anwendungen mit einem definierten „Fail-Safe-Zustand“ zugelassen. Dieser sichere Zustand ist immer der energielose Zustand. Dafür ist eine Fehlersicherheit entsprechend den zugrunde gelegten Normen erforderlich.

WARNUNG

Nicht-bestimmungsgemäße Verwendung

Jeder Gebrauch, der die zulässigen niedergeschriebenen Werte überschreitet oder andere Festlegungen aus dieser Betriebsanleitung oder anderen Dokumenten der Gesamtdokumentation nicht beachtet, gilt als nicht-bestimmungsgemäß und ist somit verboten.

Dies gilt insbesondere für die durch die Beckhoff Automation definierten Anwendungsfälle, die vollumfänglich geprüft und zertifiziert sind und deren Eigenschaften und Betriebsbedingungen zugesichert werden können. Darüberhinausgehende Anwendungsfälle sind nicht-bestimmungsgemäß und bedürfen der Prüfung der Beckhoff Automation.

Eine nicht-bestimmungsgemäße Verwendung hat den Verlust der Sicherheit sowie das Erlöschen der Zertifizierungen und der Zulassung zur Folge.

6 Funktionsweise

Sie haben 2 verschiedene Möglichkeiten, um Ihr Safety-Projekt zu exportieren oder importieren, die Sie den folgenden Kapiteln entnehmen.

6.1 Manuell

Dieses Kapitel beinhaltet eine Beschreibung des manuellen Vorgehens, welches in der Entwicklungsumgebung TwinCAT 3 über das Kontextmenü des Safety-Projekts erfolgt.

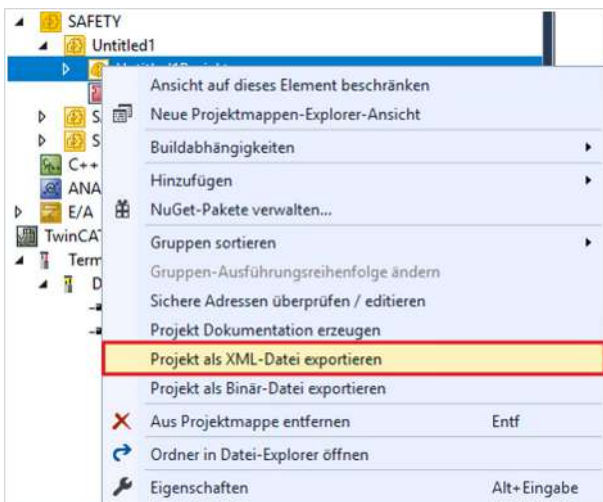
6.1.1 Export

i Export-Voraussetzung erfüllen

- valides Safety-Projekt

⇒ Die Validität Ihres Safety-Projekts prüfen Sie über die Funktionsflächen „Safety-Projekt verifizieren“ oder „Safety-Projekt / I/O Konfiguration verifizieren“ in der Menüleiste.

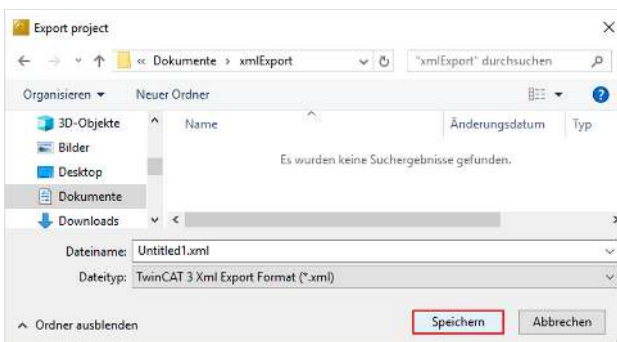
Gehen Sie wie folgt vor, um ein Safety-Projekt zu exportieren:



1. Rechtsklick auf Safety-Projekt
2. „Projekt als XML-Datei exportieren“ anklicken



3. Speicherort auswählen
4. In der Zeile „Dateityp“ das Format „TwinCAT 3 Xml Export Format (*.xml)“ auswählen



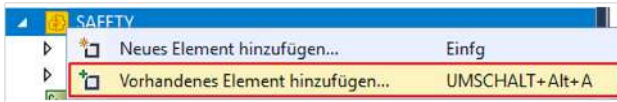
5. Speicherort mit „Speichern“ bestätigen

6.1.2 Import

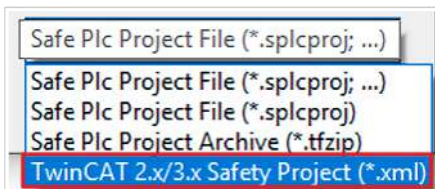
● Import-Voraussetzungen erfüllen

i

- Gültige Daten in Ihrer XML-Datei
- XML-Datei valide zur XSD-Datei

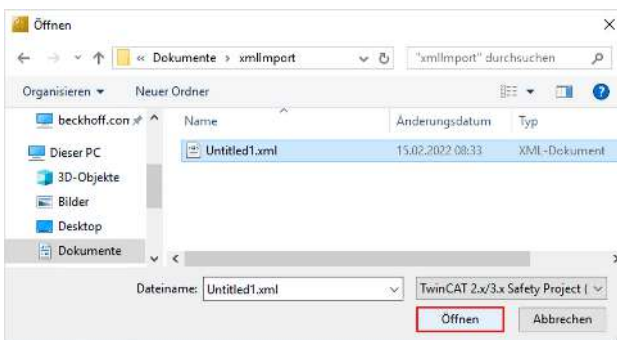


1. Rechtsklick auf Safety-Applikation
2. „Vorhandenes Element hinzufügen...“ anklicken



Ihr Explorer öffnet sich und sie können eine Datei auswählen.

3. Rechts in der Suchleiste „TwinCAT 2.x/3.x Safety Project (*.xml)“ anklicken, um nach XML-Dateien zu suchen



4. Speicherort auswählen
5. Datei auswählen
6. Mit „Öffnen“ bestätigen

6.2 Automation Interface

Mit dem TwinCAT Automation Interface haben Sie die Möglichkeit TwinCAT-XAE-Konfigurationen per Programmiercodes oder Skriptcodes automatisch zu erzeugen und zu bearbeiten.

Das Automation Interface unterstützt den XML-Export und XML-Import ab TwinCAT 3-Version 3.1.4020.0 oder neuer.

Für weitere Informationen zum Automation Interface sehen Sie in das Dokument [1] unter [Referenzen](#) [► 7].

Die manuelle Testung des Automation Interface finden Sie in Kapitel [Automation Interface testen](#) [► 102].

6.2.1 Export

Dieses Kapitel zeigt beispielhaft, wie eine programmatische Anwendung in C#-Code aussehen kann.

Die manuelle Testung des Automation Interface finden Sie in Kapitel [Manueller Export](#) [► 102].

● Methoden des Automation Interface

i Beim Export mit dem Automation Interface verwenden Sie die folgenden Methoden:

- LookupTreeItem
- ProduceXml
- ConsumeXml

⇒ Weitere Informationen zu diesen Methoden finden Sie im [Beckhoff Information System](#).

● Usings

i Für dieses Beispiel benötigen Sie die folgenden Usings:

- System.Collections
- System.Xml
- System.Xml.XPath
- TCatSysManagerLib

Gehen Sie wie folgt vor, um mit dem Automation Interface ein Safety-Projekt zu exportieren:

```
/// <summary>
/// Export all Safety Projects in TC3 XML format.
/// </summary>
/// <param name="systemManager">System Manager</param>
private void Tc3SafetyProjectXmlExport(ITcSysManager systemManager)
{
```

1. Hauptelement des Safety-Baums im TwinCAT 3-Projekt nachschlagen

```
ITcSmTreeItem treeItem = systemManager.LookupTreeItem("TISC");
```

2. Enumerator des Hauptelements des Safety-Baums abrufen, um die Safety-Projekte zu durchlaufen

```
IEnumerator enumerator = treeItem.GetEnumerator();
while (enumerator.MoveNext())
{
```

3. Safety-Projekt abrufen

```
ITcSmTreeItem safetyProjectParentNode = enumerator.Current as ITcSmTreeItem;
ITcSmTreeItem safetyProjectNode = safetyProjectParentNode.LookupChild(safetyProjectParent
Node.Name + " Project");
if (safetyProjectNode != null)
{
```

4. XML auf dem Safety-Projekt-Baumelement exportieren

```
string producedXml = safetyProjectNode.ProduceXml();
XmlDocument xmlDoc = new XmlDocument();
xmlDoc.LoadXml(producedXml);
```

5. Zu den baumrelevanten Parametern „Active“, „File“ und „Format“ navigieren

6. Werte ändern

```

XPathNavigator navigator = xmlDoc.CreateNavigator();
XPathNavigator navigator_Export = navigator.SelectSingleNode(@"TreeItem/SafetyProjectDef/
Commands/Export");
if (navigator_Export == null) continue;
XPathNavigator navigator_Export_Active = navigator_Export.SelectSingleNode("Active");
if (navigator_Export_Active == null) continue;
XPathNavigator navigator_Export_File = navigator_Export.SelectSingleNode("File");
if (navigator_Export_File == null) continue;
XPathNavigator navigator_Export_Format = navigator_Export.SelectSingleNode("Format");
if (navigator_Export_Active.SetValue(XmlConvert.ToString(true));
navigator_Export_File.SetValue(string.Format(@"C:\{0}.xml", safetyProjectNode.Name));
navigator_Export_Format.SetValue("TwinCAT 3 Xml Export Format");

```

7. Die geänderte XML-Datei importieren, um die exportierte XML-Datei des Safety-Projekts zu erzeugen

```

safetyProjectNode.ConsumeXml(xmlDoc.InnerXml);
}
}
}

```

6.2.2 Import

Dieses Kapitel zeigt beispielhaft, wie eine programmatische Anwendung in C#-Code aussehen kann.

Die manuelle Testung des Automation Interface finden Sie in Kapitel [Manueller Import](#) [► 103].

● Methoden des Automation Interface



Beim Export mit dem Automation Interface verwenden Sie die folgenden Methoden:

- LookupTreeItem
- CreateChild

⇒ Weitere Informationen zu diesen Methoden finden Sie im [Beckhoff Information System](#).

● Using



Für dieses Beispiel benötigen Sie das folgende Using:

- TCatSysManagerLib

Gehen Sie wie folgt vor, um ein Safety-Projekt zu importieren:

1. Hauptelement des Safety-Baums in TwinCAT 3-Projekt nachschlagen

```
ITcSmTreeItem safety = systemManager.LookupTreeItem("TISC");
```

2. Safety-Projekt importieren

```
ITcSmTreeItem newProject = safety.CreateChild("NameOfProject", 0, null,
pathToProjectOrTfzipFile);
```

Auf „CreateChild“ folgen die Informationen:

"Projektname", SubType, null, Pfaddatei zum Projekt

Sie können 3 verschiedene Werte als SubType verwenden:

- 0: Projekt zum Lösungsverzeichnis kopieren
- 1: Projekt zum Lösungsverzeichnis verschieben
- 2: Original-Projektspeicherort verwenden (falls verwendet, verwenden Sie "" als Projektnamensparameter)

Sie können zwischen 3 verschiedenen Import-Formaten als Quellvorlage für Ihr Safety-Projekt wählen:

- *.splcproj-Datei
- *.tfzip-Safety-Projekt-Archivdatei
- *.xml-Datei

7 XSD-Datei

Eine XML Schema Definition (XSD) definiert die Struktur von XML-Dateien.

Die XSD beschreibt unter anderem:

- Datentypen für Elemente und Attribute
- XML-Schema-Instanzen und Instanz-Gruppen
- Wertebereiche
- Reihenfolgen
- Mindestangaben und optionale Angaben

Bei selbsterstellten XML-Dateien mit externen Tools können Sie mit der XSD-Datei sicherstellen, dass Ihre XML-Datei für TE9000 valide ist. Dies gewährleistet einen korrekten Import.

Die XSD-Datei gilt für die XML-Version 0.31 oder neuer. Eine Validierung der XML-Datei gegen die XSD ist ebenfalls erst ab einer XML-Version 0.31 möglich.

Die folgenden Kapitel beschreiben die einzelnen Elemente und Attribute der XSD-Datei.

7.1 TwinCATExport



Abb. 1: TwinCATExport

Element	TwinCATExport
Inhalt	komplex
Beschreibung	Wurzelement der exportierten XML-Datei

7.1.1 TwinCATExport Attribut

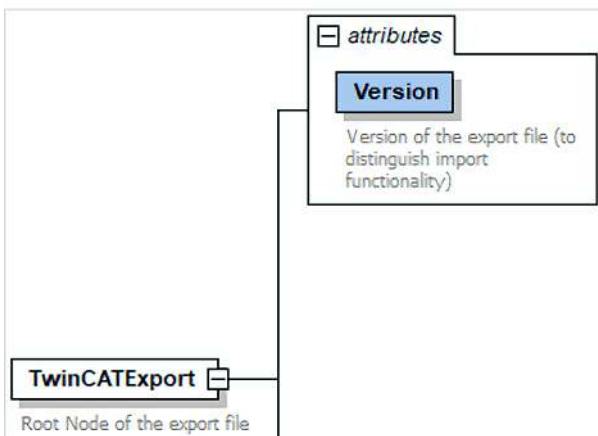


Abb. 2: TwinCATExport Attribut

Attribut	Version
Datentyp	xs:string
Einschränkung	Auf verwendete TE9000-Version festgelegt Siehe Kapitel Systemvoraussetzungen [► 15].
Verwendung	erforderlich
Beschreibung	Version der Exportdatei

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<TwinCATExport Version="0.31">
</TwinCATExport>
```

7.1.2 BaseConfiguration

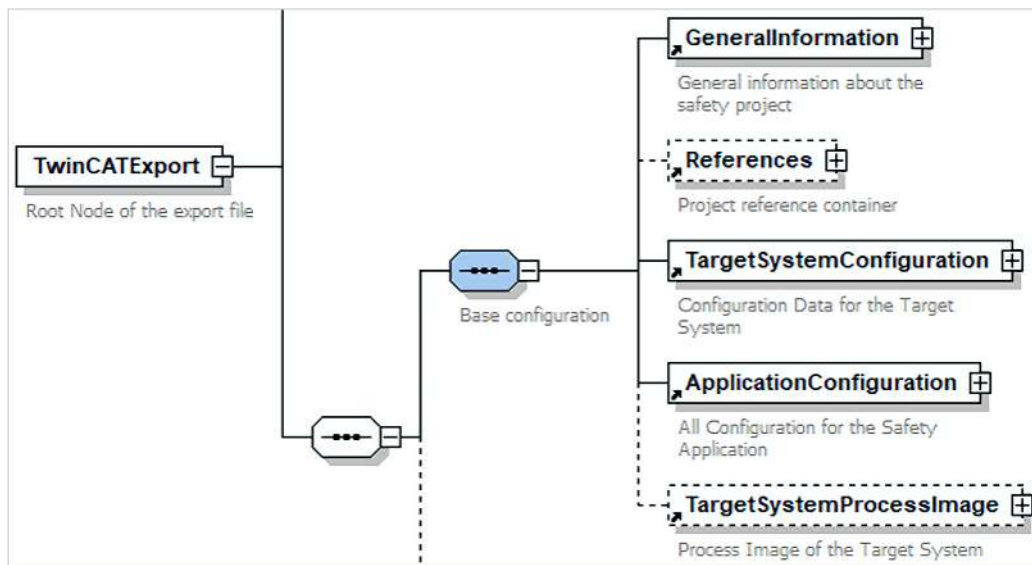


Abb. 3: Base configuration

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Elemente	GeneralInformation	References	TargetSystem-Configuration	Application-Configuration	TargetSystem-ProcessImage
minOcc	1	0	1	1	0
maxOcc	1	1	1	1	1
Inhalt	komplex	komplex	komplex	komplex	komplex
Beschreibung	Allgemeine Informationen über das Safety-Projekt Siehe Kapitel GeneralInformation [23].	Referenzen des Safety-Projekts Siehe Kapitel References [24].	Konfigurationsinformationen für das Zielsystem Siehe Kapitel TargetSystemConfiguration [25].	Gesamte Konfiguration für die Sicherheitsapplikation Siehe Kapitel ApplicationConfiguration [31].	Prozessabbild des Zielsystems Siehe Kapitel TargetSystemProcessImage [67].

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<TwinCATExport Version="0.31">
  <GeneralInformation>
  </GeneralInformation>
  <References>
  </References>
  <TargetSystemConfiguration>
  </TargetSystemConfiguration>
  <ApplicationConfiguration>
  </ApplicationConfiguration>
  <TargetSystemProcessImage>
  </TargetSystemProcessImage>
</TwinCATExport>
```


7.1.2.1 GeneralInformation

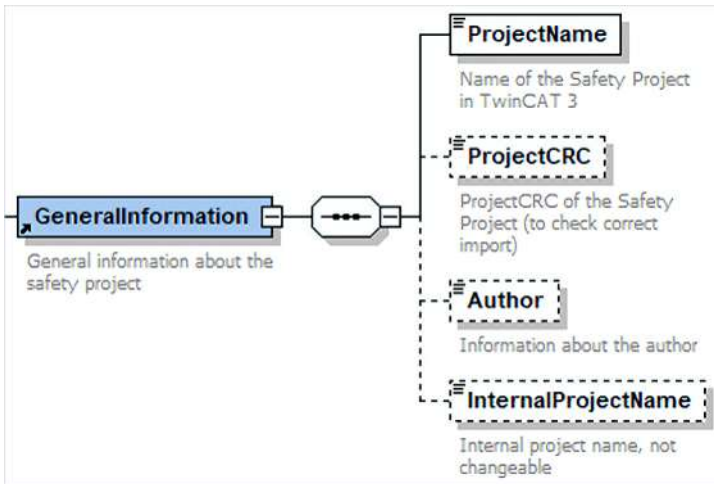


Abb. 4: GeneralInformation

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Elemente	ProjectName	ProjectCRC	Author	InternalProjectName
minOcc	1	0	0	0
maxOcc	1	1	1	1
Datentyp	xs:string	HexDecUInt32Value	xs:string	xs:string
Inhalt	einfach	einfach	einfach	einfach
Beschreibung	Name des Safety-Projekts im TE9000-Projekt	CRC des Safety-Projekts	Informationen über den Autor, der bei der Projekterzeugung angegeben wurde	Name des Safety-Projekts, der bei der Projekterzeugung festgelegt wurde

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<GeneralInformation>
  <ProjectName>Example</ProjectName>
  <ProjectCRC>0xFB2D</ProjectCRC>
  <Author>Person1</Author>
  <InternalProjectName>Example</InternalProjectName>
</GeneralInformation>
```

7.1.2.2 References

Hier können Sie aktuell die User-FB-Bibliotheken referenzieren.

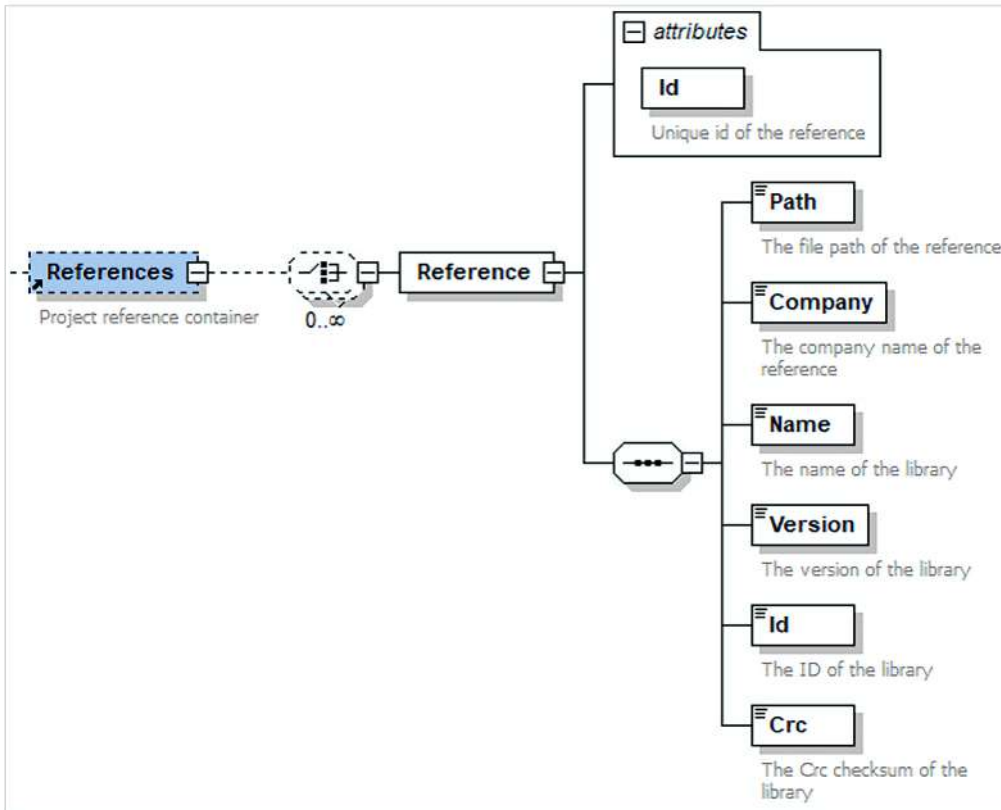


Abb. 5: References

Element	Reference
minOcc	1
maxOcc	1
Inhalt	komplex
Beschreibung	Projekt-Referenz

Attribut	Id
Datentyp	xs:ID
Verwendung	erforderlich
Beschreibung	Eindeutige Identifikationsnummer der Referenz Siehe Kapitel ID-Regeln [▶ 75].

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Elemente	Path	Company	Name	Version	Id	Crc
minOcc	1	1	1	1	1	1
maxOcc	1	1	1	1	1	1
Datentyp	xs:string	xs:string	xs:string	xs:string	xs:string	xs:unsignedInt
Inhalt	einfach	einfach	einfach	einfach	einfach	einfach
Beschreibung	Dateipfad der Referenz	Unternehmensname der Referenz	Name der Bibliothek	Version der Bibliothek	Identifikationsnummer der Bibliothek	CRC-Prüfsumme der Bibliothek

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<References>
  <Reference Id="r1">
    <Path>C:\TwinCAT\3.1\Components\Safety\Managed Libraries\Beckhoff\TestLib\1.0\TestLib.ufbllib</
Path>
    <Company>Beckhoff</Company>
    <Name>TestLib</Name>
    <Version>1.0</Version>
    <Id>f7e6a540-3480-4c19-b0b9-a75d5d5100e9</Id>
    <Crc>802636370</Crc>
  </Reference>
</References>
```

7.1.2.3 TargetSystemConfiguration

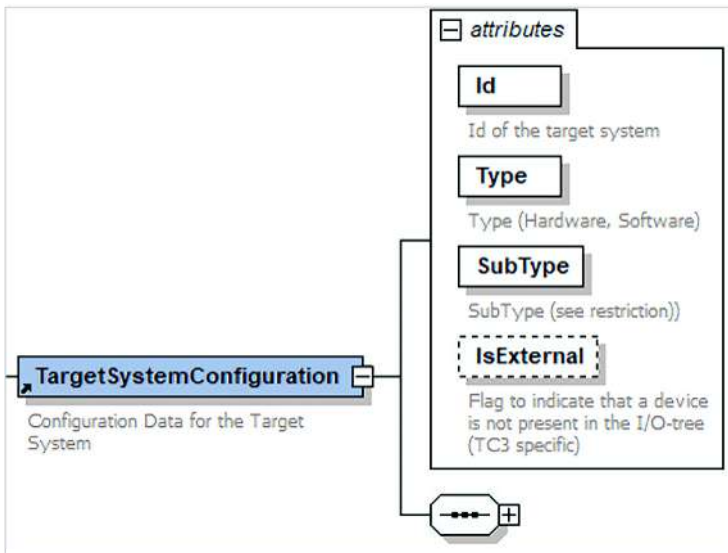


Abb. 6: TargetSystemConfiguration Attribute

Attribute	Id	Type	SubType	IsExternal
Datentyp	xs:ID	xs:string	xs:string	xs:boolean
Verwendung	erforderlich	erforderlich	erforderlich	erforderlich
Einschränkung	/	Hardware, Software	AMI891x, AMP891x, AX891x, EJ1914, EJ1918, EJ1957, EJ2914, EJ2918, EJ6910, EK1960, EL1918, EL2911, EL2912, EL6900, EL6910, EL6930, EP1918, EP1957, EP2918	/
Beschreibung	Identifikationsnummer des Zielsystems innerhalb der XML-Datei Die Default-ID ist „t1“. siehe Kapitel ID-Regeln [▶ 75].	Typ des Zielsystems	Sub-Typ des Zielsystems	Kennzeichnet, dass ein Gerät nicht im I/O-Baum verfügbar ist

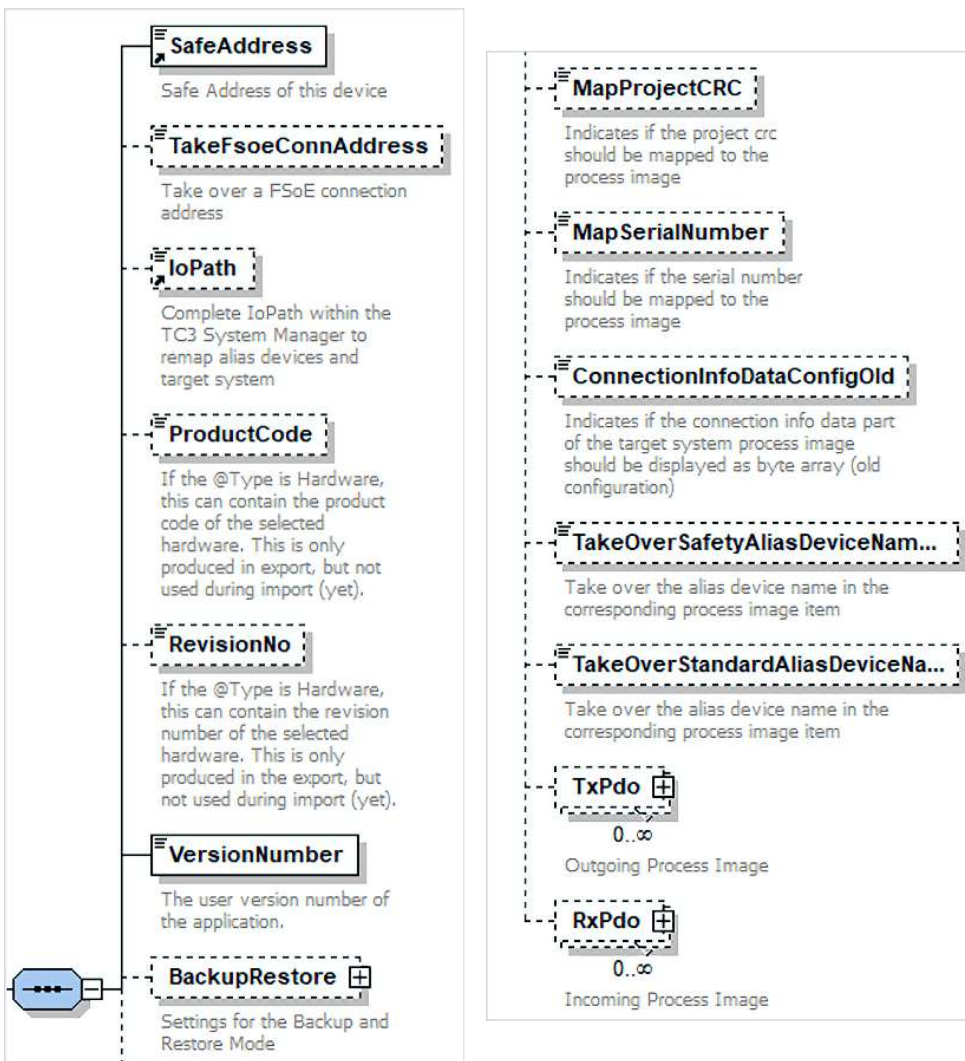


Abb. 7: TargetSystemConfiguration Elemente

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Elemente	SafeAddress	TakeFsoeConnAddress	IoPath	ProductCode	RevisionNo
minOcc	1	0	0	0	0
maxOcc	1	1	1	1	1
Datentyp	xs:int	xs:boolean	xs:string	xs:int	xs:int
Inhalt	einfach	einfach	einfach	einfach	einfach
Beschreibung	Sichere Adresse des Geräts Der Wertebereich der sicheren Adresse ist abhängig vom verwendeten Gerät.	Eine FSoE-Adresse einer Verbindung übernehmen Aktuell nicht in Verwendung.	Gesamter IO-Pfad im TwinCAT 3 System Manger, um Alias-Geräte und Zielsysteme neu zuzuordnen	Zeigt den Produktcode der verwendeten Hardware Wird ausschließlich im Export aber nicht im Import verwendet.	Zeigt die Revisionsnummer der verwendeten Hardware Wird ausschließlich im Export aber nicht im Import verwendet.

Elemente	VersionNumber	BackupRestore	MapProjectCRC	MapSerialNumber	ConnectioninfoDataConfigOld
minOcc	1	0	0	0	0
maxOcc	1	1	1	1	1

Elemente	VersionNumber	BackupRestore	MapProjectCRC	MapSerialNum- ber	Connectioninfo- DataConfigOld
Datentyp	xs:unsignedShort		xs:boolean	xs:boolean	xs:boolean
Inhalt	einfach	komplex	einfach	einfach	einfach
Beschreibung	Nummer der anwenderspezifischen Version	Einstellungen für die „Backup & Restore“-Funktion Siehe Kapitel BackupRestore [▶ 28].	Gibt an, ob die Projekt-CRC im Prozessabbild angelegt werden soll	Gibt an, ob die Seriennummer im Prozessabbild angelegt werden soll	Gibt an, ob der Teil der Verbindungsinfo- daten im Prozessabbild des Zielsystems als Byte-Anordnung angezeigt werden soll

Elemente	TakeOverSafetyAliasDeviceNamesIn- ProcessImage	TakeOverStandardAliasDeviceName- sInProcessImage	TxPdo	RxPdo
minOcc	0	0	0	0
maxOcc	1	1	unbegrenzt	unbegrenzt
Datentyp	xs:boolean	xs:boolean	ProcessImage	ProcessImage
Inhalt	einfach	einfach	komplex	komplex
Beschreibung	Den Alias-Gerätenamen im Prozessabbild des Zielsystems übernehmen	Den Alias-Gerätenamen im Prozessabbild des Zielsystems übernehmen	Ausgangsprozessabbild Siehe Kapitel TxPdo [▶ 29]. Aktuell nicht in Verwendung.	Eingangsprozessabbild Siehe Kapitel RxPdo [▶ 31]. Aktuell nicht in Verwendung.

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<TargetSystemConfiguration Id="t1" Type="Hardware" SubType="AX891x" IsExternal="false">
  <SafeAddress>1</SafeAddress>
  <TakeFsoeConnAddress>>false</TakeFsoeConnAddress>
  <IoPath>TIID^Device 2 (EtherCAT)^Term 30 (AX8620-0000-0104)^Drive 33 (AX8206-0210-0104)</IoPath>
  <ProductCode>537813010</ProductCode>
  <RevisionNo>6815954</RevisionNo>
  <VersionNumber>1</VersionNumber>
  <BackupRestore Activated="false" RestoreUserAdministration="false"
NumberOfDevicesWithMatchingCRC="0"/>
  <MapProjectCRC>>false</MapProjectCRC>
  <MapSerialNumber>>false</MapSerialNumber>
  <ConnectionInfoDataConfigOld>>false</ConnectionInfoDataConfigOld>
  <TakeOverSafetyAliasDeviceNamesInProcessImage>>false</TakeOverSafetyAliasDeviceNamesInProcessImage>
  <TakeOverStandardAliasDeviceNamesInProcessImage>>false</
TakeOverStandardAliasDeviceNamesInProcessImage>
</TargetSystemConfiguration>
```

7.1.2.3.1 BackupRestore

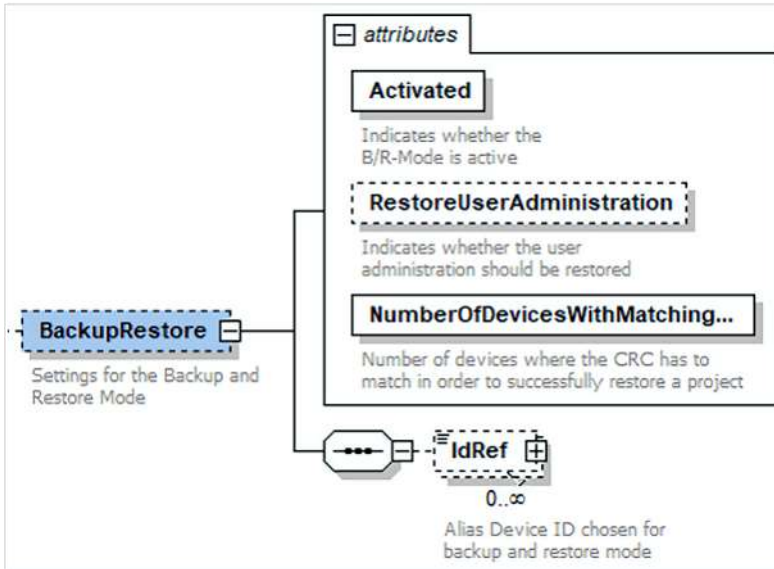


Abb. 8: TargetSystemConfiguration - BackupRestore

Attribute	Activated	RestoreUserAdministration	NumberOfDevicesWithMatchingCRC
Datentyp	xs:boolean	xs:boolean	xs:int
Verwendung	erforderlich	optional	erforderlich
Beschreibung	Gibt an, ob die Backup&Restore-Funktion aktiviert ist	Gibt an, ob die Benutzerverwaltung wiederhergestellt werden soll	Anzahl der Geräte, bei denen die CRC übereinstimmen muss, damit ein Projekt erfolgreich wiederhergestellt werden kann

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Element	IdRef
minOcc	0
maxOcc	unbegrenzt
Datentyp	xs:IDREF
Inhalt	komplex
Beschreibung	Die Identifikationsnummer des Alias-Geräts, welche für die Backup&Restore-Funktion ausgewählt wurde Siehe ID-Attribut in Kapitel SafetyAliasDevice [▶ 57].

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<BackupRestore Activated="true" RestoreUserAdministration="false"
NumberOfDevicesWithMatchingCRC="2">
  <IdRef StoreProjectCrcInSlave="true" StoreSlaveProjectCrcInMaster="true"
StoreMasterProjectCrcInSlave="false" ReadProjectCrcFromMaster="false">g1_a3</IdRef>
  <IdRef StoreProjectCrcInSlave="true" StoreSlaveProjectCrcInMaster="false"
StoreMasterProjectCrcInSlave="false" ReadProjectCrcFromMaster="false">g1_a6</IdRef>
</BackupRestore>
```


7.1.2.3.1.1 IdRef



Abb. 9: BackupRestore - IdRef

Attribute	StoreProjectCrcInSlave	StoreSlaveProjectCrcInMaster	StoreMasterProjectCrcInSlave	ReadProjectCrcFromMaster
Datentyp	xs:boolean	xs:boolean	xs:boolean	xs:boolean
Verwendung	erforderlich	erforderlich	erforderlich	erforderlich
Beschreibung	Gibt an, ob die Project-CRC im Slave gespeichert werden soll	Gibt an, ob die Project-CRC des Slaves im Master gespeichert werden soll	Gibt an, ob die Project-CRC des Masters im Slave gespeichert werden soll	Gibt an, ob die Project-CRC vom Master gelesen werden soll

7.1.2.3.2 TxPdo

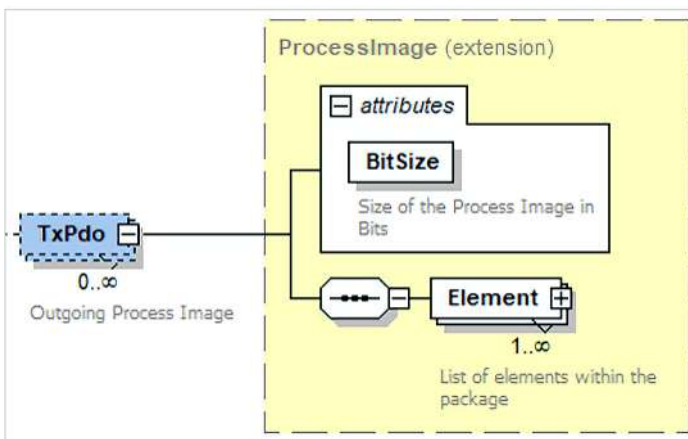


Abb. 10: TargetSystemConfiguration - TxPdo

Attribut	BitSize
Datentyp	xs:int
Verwendung	erforderlich
Beschreibung	Größe des Prozessabbilds in Bits

Element	Element
minOcc	1
maxOcc	unbegrenzt
Inhalt	komplex
Beschreibung	Liste der Elemente

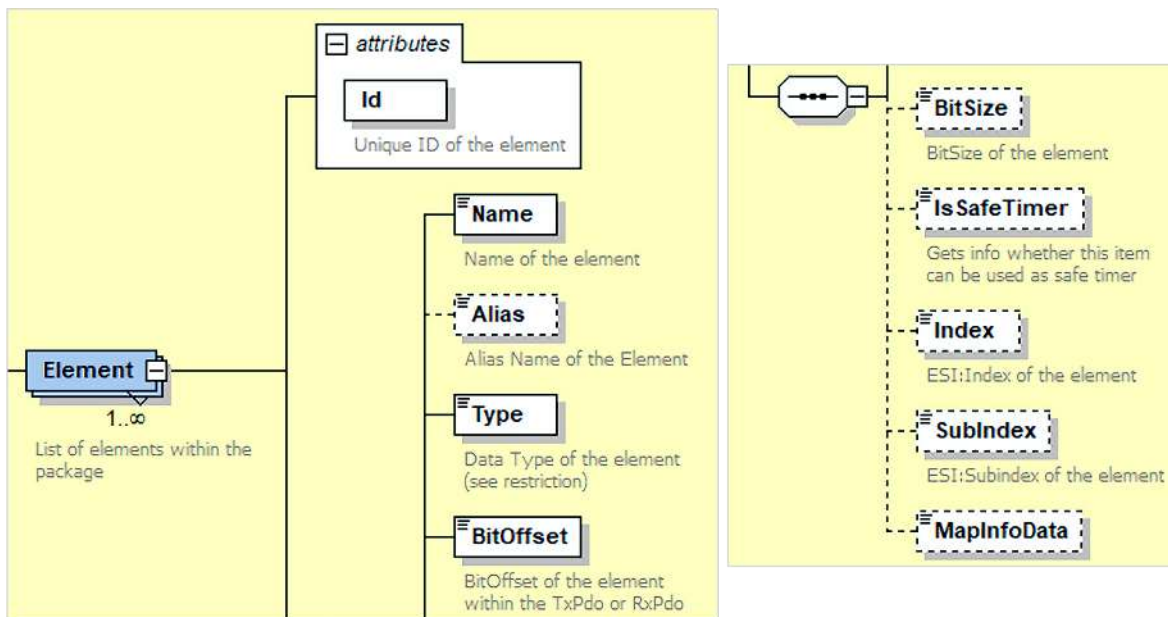


Abb. 11: TxPdo – Element

Attribut	Id
Datentyp	xs:ID
Verwendung	erforderlich
Beschreibung	Eindeutige Identifikationsnummer des Elements Die Identifikationsnummer ist abhängig vom Kontext, siehe Kapitel ID-Regeln [75].

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Elemente	Name	Alias	Type	BitOffset	BitSize
minOcc	1	0	1	1	0
maxOcc	1	1	1	1	1
Datentyp	xs:string	xs:string	xs:string	xs:int	xs:int
Inhalt	einfach	einfach	einfach	einfach	einfach
Einschränkung	/	/	BIT, BYTE, USINT, UINT, INT, UDINT, DINT	/	/
Beschreibung	Name des Elements	Alias-Name des Elements	Datentyp des Elements	Bit-Offset des Elements im Prozessabbild	Bit-Größe des Elements

Elemente	IsSafeTimer	Index	SubIndex	MapInfoData
minOcc	0	0	0	0
maxOcc	1	1	1	1
Datentyp	xs:boolean	/	/	xs:boolean
Inhalt	einfach	/	/	einfach
Beschreibung	Gibt an, ob das Element als sicherer Timer verwendet werden kann	Index aus ESI-Gerätebeschreibung	Sub-Index aus ESI-Gerätebeschreibung	Als Informationsdaten in I/O-Prozessabbild übernehmen

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<TxPdo BitSize="4096">
  <Element Id="g1_a2_lo1">
    <Name>ChA_STO_1</Name>
    <Type>BIT</Type>
    <BitOffset>0</BitOffset>
```

```

<BitSize>1</BitSize>
<IsSafeTimer>>false</IsSafeTimer>
<Index>49281</Index>
<SubIndex>1</SubIndex>
<MapInfoData>>false</MapInfoData>
</Element>
<Element Id="g1_a2_lo2">
  <Name>ChA_STO_2</Name>
  <Type>BIT</Type>
  <BitOffset>1</BitOffset>
  <BitSize>1</BitSize>
  <IsSafeTimer>>false</IsSafeTimer>
  <Index>49281</Index>
  <SubIndex>2</SubIndex>
  <MapInfoData>>false</MapInfoData>
</Element>
...
</TxPdo>

```

7.1.2.3 RxPdo

Das Eingangsprozessabbild „RxPdo“ entspricht in seinen Attributen und Elemente dem Ausgangsprozessabbild „TxPdo“. Für weitere Informationen sehen Sie daher in das Kapitel [TxPdo](#) [► 29].

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```

<RxPdo BitSize="48">
  <Element Id="g1_a3_i1">
    <Name>Safe Data Byte 0[0]</Name>
    <Type>BIT</Type>
    <BitOffset>8</BitOffset>
    <BitSize>1</BitSize>
    <IsSafeTimer>>false</IsSafeTimer>
  </Element>
  <Element Id="g1_a3_i2">
    <Name>Safe Data Byte 0[1]</Name>
    <Type>BIT</Type>
    <BitOffset>9</BitOffset>
    <BitSize>1</BitSize>
    <IsSafeTimer>>false</IsSafeTimer>
  </Element>
  ...
</RxPdo>

```

7.1.2.4 ApplicationConfiguration

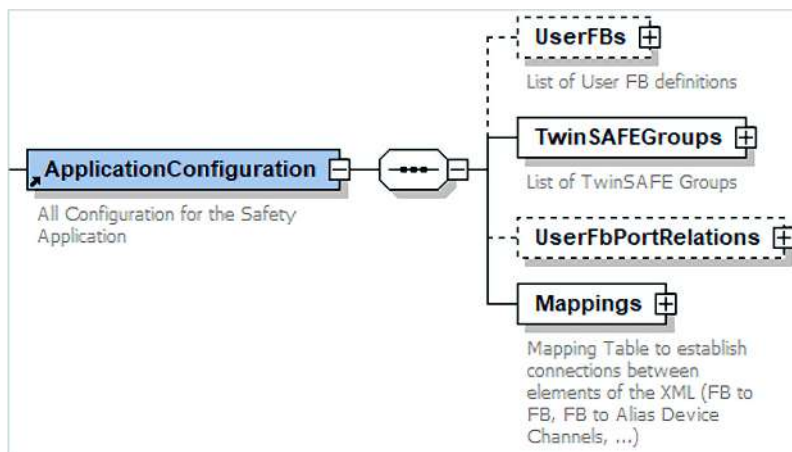


Abb. 12: ApplicationConfiguration

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Elemente	UserFBs	TwinSAFEGroups	UserFbPortRelations	Mappings
minOcc	0	1	0	1
maxOcc	1	1	1	1

Elemente	UserFBs	TwinSAFEGroups	UserFbPortRelations	Mappings
Inhalt	komplex	komplex	komplex	komplex
Beschreibung	Liste der User-FB-Definitionen Siehe Kapitel UserFBs [▶ 32].	Liste der TwinSAFE-Gruppen Siehe Kapitel TwinSAFEGroups [▶ 51].	Beziehung zwischen den User-FB-Port-Identifikationsnummern und den FB-Port-Identifikationsnummern Siehe Kapitel UserFbPortRelations [▶ 65].	Mapping-Tabelle zur Herstellung von Verbindungen zwischen den Elementen der XML-Datei (zwischen verschiedenen FBs, zwischen einem FB und dem Kanal eines Alias-Geräts) Siehe Kapitel Mappings [▶ 66].

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<ApplicationConfiguration>
  <UserFBs>
  </UserFBs>
  <TwinSAFEGroups>
  </TwinSAFEGroups>
  <UserFbPortRelations>
  </UserFbPortRelations>
  <Mappings>
  </Mappings>
</ApplicationConfiguration>
```

7.1.2.4.1 UserFBs

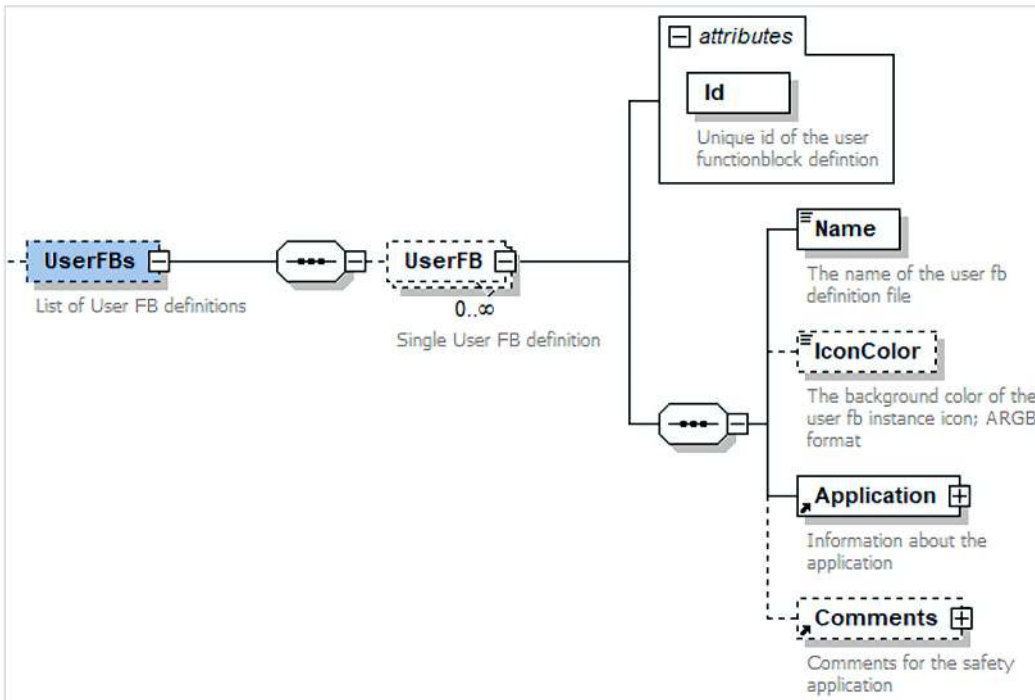


Abb. 13: ApplicationConfiguration - UserFBs

Element	UserFB
minOcc	0
maxOcc	unbegrenzt
Inhalt	komplex
Beschreibung	Definition eines Einzelbenutzer-Funktionsbausteins

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Attribut	Id
Datentyp	xs:ID
Verwendung	erforderlich
Beschreibung	Eindeutige Identifikationsnummer der User-FB-Definition Siehe Kapitel ID-Regeln [► 75].

Elemente	Name	IconColor	Application	Comments
minOcc	1	0	1	0
maxOcc	1	1	1	1
Inhalt	einfach	einfach	komplex	komplex
Beschreibung	Dateiname der User-FB-Definition	Hintergrundfarbe des Symbols der User-FB-Instanz Format: ARGB	Informationen über die Applikation	Kommentare zur Sicherheitsapplikation

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<UserFBs>
  <UserFB Id="u1">
    <Name>UserFB1</Name>
    <IconColor>-990068</IconColor>
    <Application>
    </Application>
    <Comments>
    </Comments>
  </UserFB>
</UserFBs>
```

7.1.2.4.1.1 Application

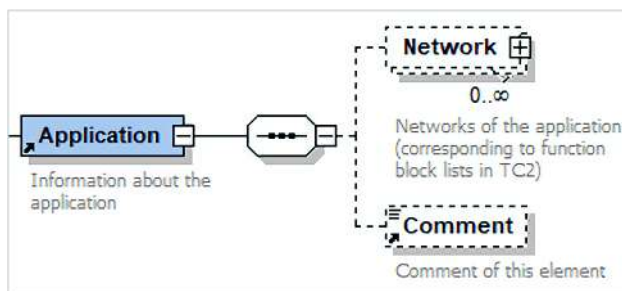


Abb. 14: Application

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Elemente	Network	Comment
minOcc	0	0
maxOcc	unbegrenzt	1
Datentyp	/	xs:string
Inhalt	komplex	einfach
Beschreibung	Netzwerke der Anwendung (entsprechend den FB-Listen in TwinCAT 3) Siehe Kapitel Network [► 34].	Kommentar zu diesem Element

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<Application>
  <Network Id="u1_n1" OrderId="1">
  </Network>
  <Comment>ScalingOffset20</Comment>
</Application>
```

7.1.2.4.1.1.1 Network

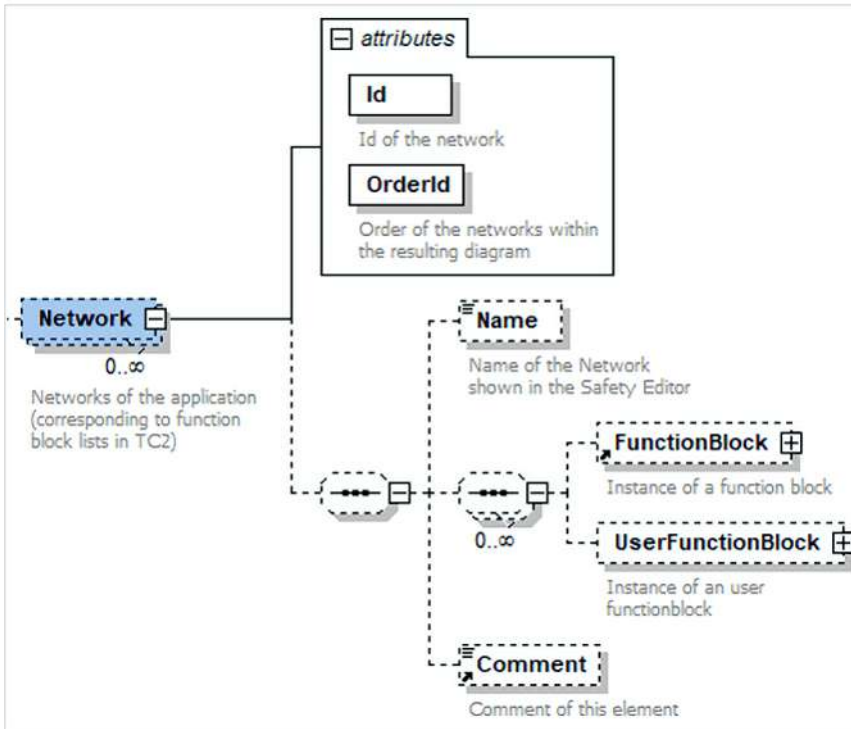


Abb. 15: Network

Attribute	Id	OrderId
Datentyp	xs:ID	xs:int
Verwendung	erforderlich	erforderlich
Beschreibung	Netzwerk-Identifikationsnummer Siehe Kapitel ID-Regeln [75].	Anordnung der Netzwerke im Diagramm der Safety-Applikation

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Elemente	Name	FunctionBlock	UserFunctionBlock	Comment
minOcc	0	0	0	0
maxOcc	1	1	1	1
Datentyp	xs:string	/	/	xs:string
Inhalt	einfach	komplex	komplex	einfach
Beschreibung	Netzwerkname, der im Safety Editor angezeigt wird	Instanz eines FBs Siehe Kapitel FunctionBlock [35].	Instanz eines User FBs Siehe Kapitel UserFunctionBlock [43].	Kommentar zu diesem Element

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<Network Id="g1_n1" OrderId="1">
  <Name>Network1</Name>
  <FunctionBlock Id="g1_n1_f1" OrderId="1">
  </FunctionBlock>
  <UserFunctionBlock Id="g1_n1_u2" MinOrderId="2" MaxOrderId="4">
  </UserFunctionBlock>
  <Comment>Test Network1</Comment>
</Network>
```

7.1.2.4.1.1.1 FunctionBlock

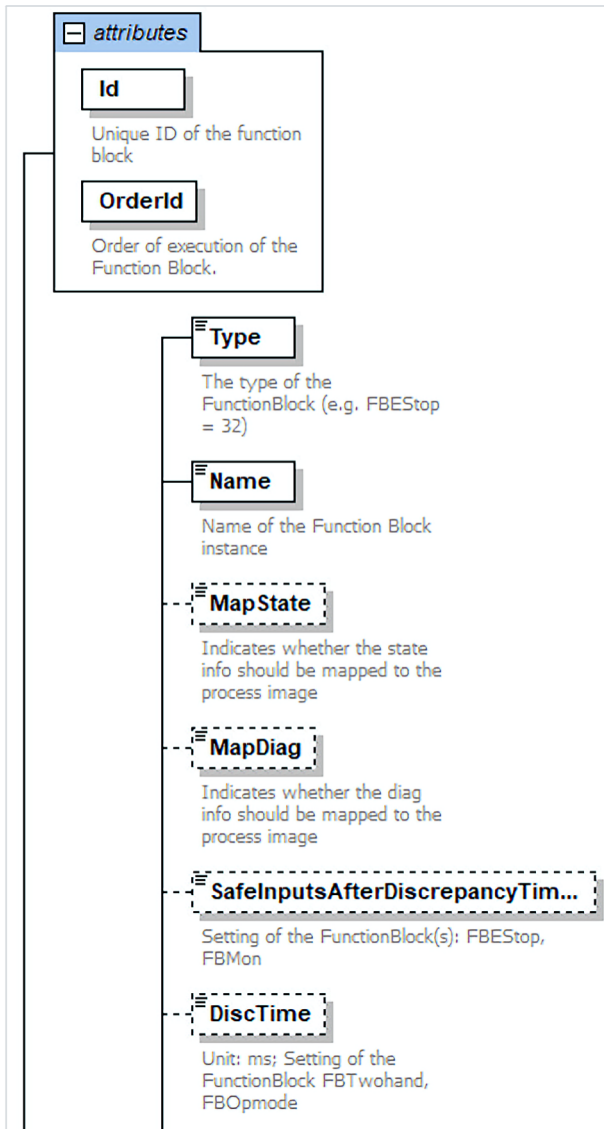


Abb. 16: FBFunctionBlock (Ausschnitt)

Für weitere Informationen über FBs sehen Sie in Dokument [4] in Kapitel [Referenzen](#) [7].

Attribute	Id	OrderId
Datentyp	xs:ID	xs:unsignedInt
Verwendung	erforderlich	erforderlich
Beschreibung	Identifikationsnummer des FBs Siehe Kapitel ID-Regeln [75].	Ausführungsreihenfolge der FBs

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Elemente	Type	Name	MapState	MapDiag	SafeInputsAfterDiscrepancy-TimeError
minOcc	1	1	0	0	0
maxOcc	1	1	1	1	1
Datentyp	xs:unsignedInt	xs: string	xs:boolean	xs:boolean	xs:boolean
Inhalt	einfach	einfach	einfach	einfach	einfach

Elemente	Type	Name	MapState	MapDiag	SafeInputsAfterDiscrepancy-TimeError
Beschreibung	FB-Typ Siehe Kapitel FBTypes [▶ 77].	Name der FB-Instanz	Gibt an, ob die Status-Informationen im Prozessabbild des Zielsystems übernommen werden soll	Gibt an, ob die Diag-Information im Prozessabbild des Zielsystems übernommen werden soll	Einstellung der FBs FBEstop und FBMon

Elemente	DiscTime	OutputDelayTime	ManualRestartActive	ManualFunctionTestActive	SwitchOffTime
minOcc	0	0	0	0	0
maxOcc	1	1	1	1	1
Datentyp	xs:int	xs:int	xs:boolean	xs:boolean	xs:int
Inhalt	einfach	einfach	einfach	einfach	einfach
Beschreibung	Einstellung der FBs FBTwohand und FBOpMode; Einheit: ms	Einstellung der FBs FBEstop, FBMon und FBMuting; Einheit: ms	Einstellung der FBs FBMon und FBOpMode	Einstellung des FBs FBMon	Einstellung des FBs FBEEdm; Einheit: ms

Elemente	SwitchOnTime	SequentialInputsActive	FilterTime	MaxMutingTime	BackwardsMode
minOcc	0	0	0	0	0
maxOcc	1	1	1	1	1
Datentyp	xs:int	xs:boolean	xs:int	xs:int	xs:boolean
Einfach	einfach	einfach	einfach	einfach	einfach
Beschreibung	Einstellung des FBs FBEEdm; Einheit: ms	Einstellung des FBs FBMuting	Einstellung des FBs FBMuting; Einheit: ms	Einstellung des FBs FBMuting; Einheit: ms	Einstellung des FBs FBMuting

Elemente	TofTime	TonTime	ConnId	CompareMode	ToleranceTime
minOcc	0	0	0	0	0
maxOcc	1	1	1	1	1
Datentyp	xs:float	xs:float	xs:int	xs:int	xs:unsignedInt
Inhalt	einfach	einfach	einfach	einfach	einfach
Einschränkung	/	/	/	0 = 1oo2 1 = 2oo3 2 = 3oo5	/
Beschreibung	Einstellung des FBs FBEEdm; Einheit: ms	Einstellung des FBs FBMuting; Einheit: ms	Einstellung des FBs FBMuting; Einheit: ms	Einstellung des FBs FBMuting; Einheit: ms	Einstellung des FBs FBMuting

Elemente	AllowedDeviation	DivisionRounding	PresetValue	CounterLimit	MinValue
minOcc	0	0	0	0	0
maxOcc	1	1	1	1	1
Datentyp	xs:unsignedInt	xs:integer	xs:long	xs:long	xs:long
Inhalt	einfach	einfach	einfach	einfach	einfach
Beschreibung	Einstellung des FBs FBCompare	Einstellung des FBs FBDiv und FBScale	Einstellung des FBs FBCounter	Einstellung des FBs FBCounter und FBViolationCnt	Einstellung des FBs FBLimit

Elemente	MaxValue	Multiplication-First	ScalingFactor-Mul	ScalingFactor-Div	ScalingOffset
minOcc	0	0	0	0	0
maxOcc	1	1	1	1	1
Datentyp	xs:long	xs:boolean	xs:unsigned-Short	xs:unsigned-Short	xs:int
Inhalt	einfach	einfach	einfach	einfach	einfach
Beschreibung	Einstellung des FBs FBLimit	Einstellung des FBs FBScaling	Einstellung des FBs FBScaling	Einstellung des FBs FBScaling	Einstellung des FBs FBScaling

Elemente	MoveDetection-Time	PressStartDelayTime	MinPosition-Change	AllowedPositionJitter	ExcentricMode
minOcc	0	0	0	0	0
maxOcc	1	1	1	1	1
Datentyp	xs:unsigned-Short	xs:unsigned-Short	xs:unsignedInt	xs:unsigned-Short	xs:boolean
Inhalt	einfach	einfach	einfach	einfach	einfach
Beschreibung	Einstellung des FBs FBCamMonitor	Einstellung des FBs FBCamMonitor	Einstellung des FBs FBCamMonitor	Einstellung des FBs FBCamMonitor	Einstellung des FBs FBCamMonitor; True = ExcentricMode False = PendulumMode

Elemente	StopDetection-Time	MaxPosition	TDC1LowerLimit	TDC1UpperLimit	TDC2LowerLimit
minOcc	0	0	0	0	0
maxOcc	1	1	1	1	1
Datentyp	xs:unsigned-Short	xs:unsignedInt	xs:long	xs:long	xs:long
Inhalt	einfach	einfach	einfach	einfach	einfach
Beschreibung	Einstellung des FBs FBCamMonitor	Einstellung des FBs FBCamMonitor	Einstellung des FBs FBCamMonitor	Einstellung des FBs FBCamMonitor	Einstellung des FBs FBCamMonitor

Elemente	TDC2UpperLimit	BDCLowerLimit	BDCUpperLimit	OverrunMax	Overflow
minOcc	0	0	0	0	0
maxOcc	1	1	1	1	1
Datentyp	xs:long	xs:long	xs:long	xs:long	xs:boolean
Inhalt	einfach	einfach	einfach	einfach	einfach
Beschreibung	Einstellung des FBs FBCamMonitor	Einstellung des FBs FBCamMonitor	Einstellung des FBs FBCamMonitor	Einstellung des FBs FBCamMonitor	Einstellung des FBs FBSpeed

Elemente	TimeInterval	Outside	LoadSensing-Table	IncValue	DecValue
minOcc	0	0	0	0	0
maxOcc	1	1	1	1	1
Datentyp	xs:unsigned-Short	xs:boolean	/	xs:unsigned-Short	xs:unsigned-Short
Inhalt	einfach	einfach	komplex	einfach	einfach

Elemente	TimeInterval	Outside	LoadSensing-Table	IncValue	DecValue
Beschreibung	Einstellung des FBs FBSpeed	Einstellung des FBs FBLoadSensing	Einstellung des FBs FBLoadSensing Siehe Kapitel LoadSensingTable 39].	Einstellung des FBs FBViolationCnt	Einstellung des FBs FBViolationCnt

Elemente	LimitPos	LimitNeg	TargetValue	Offset	MaxTime
minOcc	0	0	0	0	0
maxOcc	1	1	1	1	1
Datentyp	xs:unsignedInt	xs:unsignedInt	xs:unsignedInt	xs:unsigned-Short	xs:unsigned-Short
Inhalt	einfach	einfach	einfach	einfach	einfach
Beschreibung	Einstellung des FBs FBSLI	Einstellung des FBs FBSLI	Einstellung des FBs FBEnvelope	Einstellung des FBs FBEnvelope	Einstellung des FBs FBEnvelope

Elemente	TimeAfterInTarget	OSSDCheckOptional	Watchdog	StartsWithRemainingTime	OverflowAllowed
minOcc	0	0	0	0	0
maxOcc	1	1	1	1	1
Datentyp	xs:unsigned-Short	xs:boolean	xs:unsigned-Short	xs:boolean	xs:boolean
Inhalt	einfach	einfach	einfach	einfach	einfach
Beschreibung	Einstellung des FBs FBEnvelope	Einstellung des FBs FBmuting	Einstellung des FBs FBScaling	Einstellung des FBs FBton2	Einstellung des FBs FBSLI2

Elemente	ExceededSupport	MultiturnMode	SingleturnPosLL	SingleturnPosUL	MultiturnPosLL
minOcc	0	0	0	0	0
maxOcc	1	1	1	1	1
Datentyp	xs:boolean	xs:boolean	xs:unsignedInt	xs:unsignedInt	xs:int
Inhalt	einfach	einfach	einfach	einfach	einfach
Beschreibung	Einstellung des FBs FBSLI2	Einstellung des FBs FBSLP	Einstellung des FBs FBSLP	Einstellung des FBs FBSLP	Einstellung des FBs FBSLP

Elemente	MultiturnPosUL	ScalingFactorNumerator	ScalingFactorDenominator	ScalingUnit	EncoderMask
minOcc	0	0	0	0	0
maxOcc	1	1	1	1	1
Datentyp	xs:int	xs:double	xs:double	xs:string	xs:unsignedLong
Inhalt	einfach	einfach	einfach	einfach	einfach
Beschreibung	Einstellung des FBs FBSLP	Einstellung des FBs FBSLP	Einstellung des FBs FBSLP	Einstellung des FBs FBSLP	Einstellung des FBs FBSLP

Elemente	EncoderSubMask	BrakeTestInterval	MinCurrentIq	MaxPositionDeviation	MinDuration
minOcc	0	0	0	0	0
maxOcc	1	1	1	1	1
Datentyp	xs:unsignedLong	xs:unsignedInt	xs:int	xs:unsignedInt	xs:unsigned-Short
Inhalt	einfach	einfach	einfach	einfach	einfach
Beschreibung	Einstellung des FBs FBSLP	Einstellung des FBs FBsbt	Einstellung des FBs FBsbt	Einstellung des FBs FBsbt	Einstellung des FBs FBsbt

Elemente	MaxDuration	Feedback	Outputs	Inports	Comment
minOcc	0	0	1	1	0
maxOcc	1	1	1	1	1
Datentyp	xs:unsignedShort	xs:unsigned short	FBPorts	FBPorts	xs:string
Inhalt	einfach	einfach	komplex	komplex	einfach
Beschreibung	Einstellung des FBs FBSBT	Einstellung des Funktionsblocks FBAdvPosMon: ChA Primary Feedback = 0; ChA Secondary Feedback = 1; ChB Primary Feedback = 2; ChB Secondary Feedback = 3	Ausgangsprozessabbild Siehe Kapitel Outputs [► 40].	Eingangsprozessabbild Siehe Kapitel Inports [► 42].	Kommentar zu diesem Element

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<FunctionBlock Id="u1_n1_f1" OrderId="1">
  <Type>71</Type>
  <Name>FBScaling1</Name>
  <MapState>>false</MapState>
  <MapDiag>>false</MapDiag>
  <DivisionRounding>2</DivisionRounding>
  <MultiplicationFirst>>true</MultiplicationFirst>
  <ScalingFactorMul>1</ScalingFactorMul>
  <ScalingFactorDiv>1</ScalingFactorDiv>
  <ScalingOffset>20</ScalingOffset>
  <Watchdog>0</Watchdog>
  <Outputs>
  </Outputs>
  <Inports>
  </Inports>
</FunctionBlock>
```

7.1.2.4.1.1.1.1 LoadSensingTable

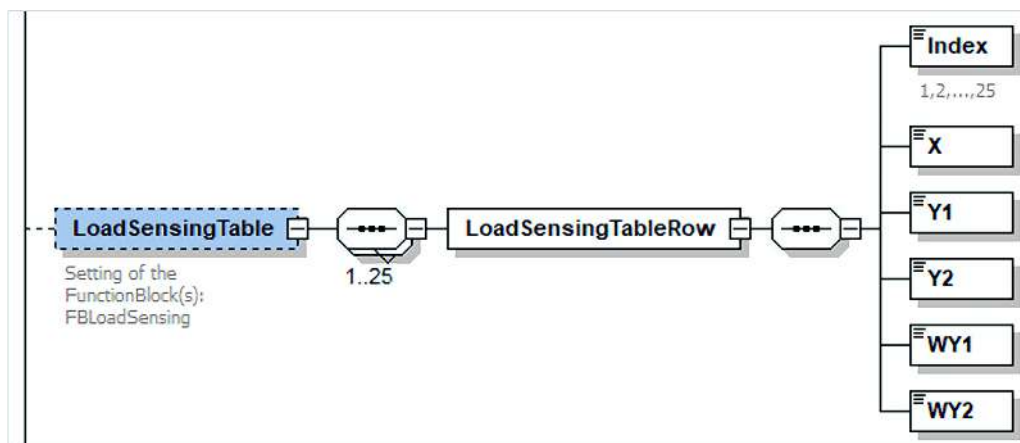


Abb. 17: LoadSensingTable

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Element	LoadSensingTableRow
minOcc	1
maxOcc	1
Datentyp	/
Inhalt	komplex
Beschreibung	Einstellungen des FBs FBLoadSensing - Tabellenzeile

Elemente	Index	X	Y1
minOcc	1	1	1
maxOcc	1	1	1
Datentyp	xs:int	xs:long	xs:long
Inhalt	einfach	einfach	einfach

Elemente	Y2	WY1	WY2
minOcc	1	1	1
maxOcc	1	1	1
Datentyp	xs:long	xs:long	xs:long
Inhalt	einfach	einfach	einfach

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```

<LoadSensingTable>
  <LoadSensingTableRow>
    <Index>1</Index>
    <X>0</X>
    <Y1>4</Y1>
    <Y2>14</Y2>
    <WY1>5</WY1>
    <WY2>13</WY2>
  </LoadSensingTableRow>
  <LoadSensingTableRow>
    <Index>2</Index>
    <X>1</X>
    <Y1>5</Y1>
    <Y2>15</Y2>
    <WY1>6</WY1>
    <WY2>14</WY2>
  </LoadSensingTableRow>
  ...
  <LoadSensingTableRow>
    <Index>25</Index>
    <X>0</X>
    <Y1>0</Y1>
    <Y2>0</Y2>
    <WY1>0</WY1>
    <WY2>0</WY2>
  </LoadSensingTableRow>
</LoadSensingTable>
    
```

7.1.2.4.1.1.1.2 Outputs

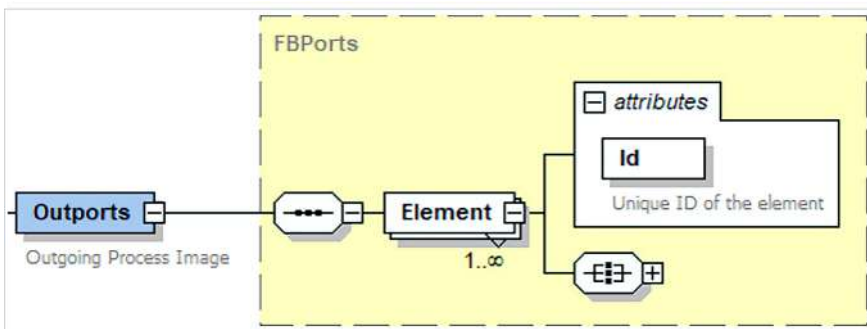


Abb. 18: Outputs

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Element	Element
minOcc	1
maxOcc	unbegrenzt
Inhalt	komplex
Beschreibung	Element im Ausgangsprozessabbild

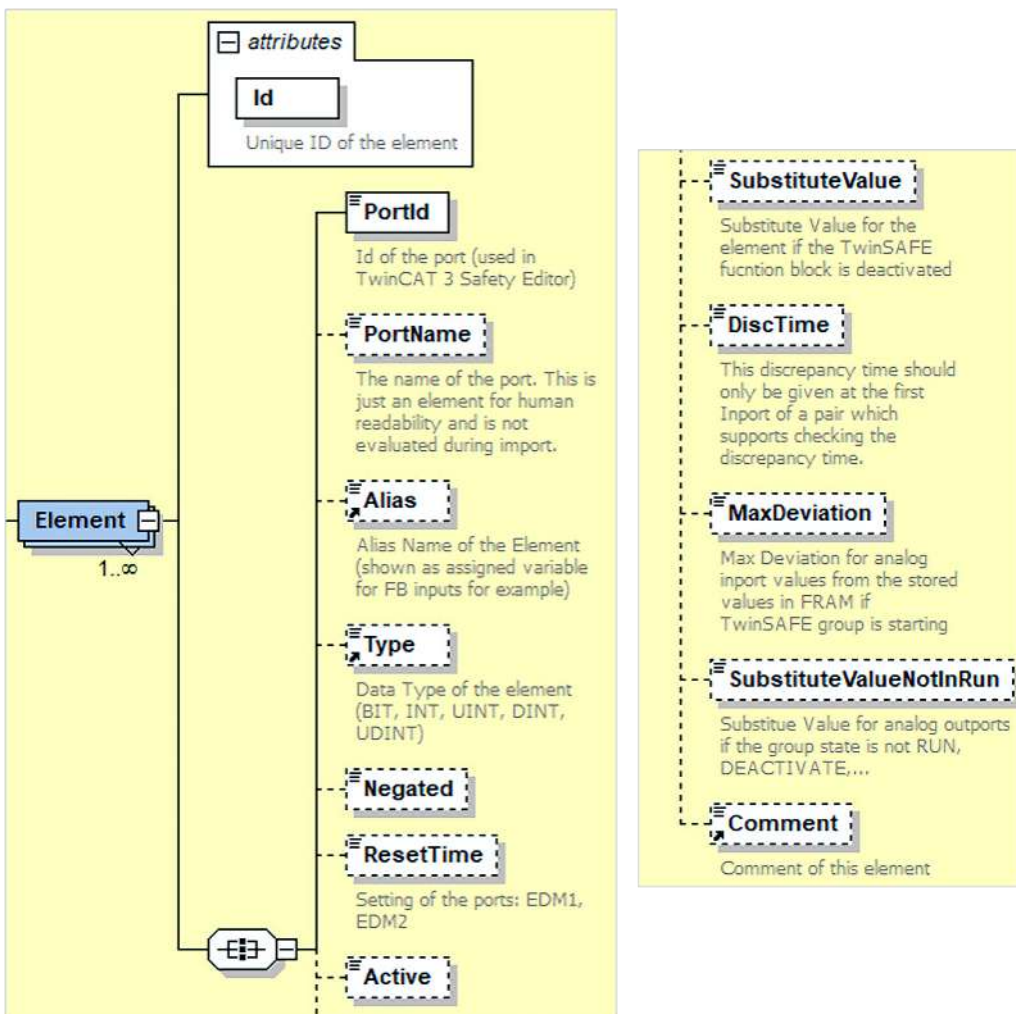


Abb. 19: Output-Element

Attribut	Id
Datentyp	xs:ID
Verwendung	erforderlich
Beschreibung	Identifikationsnummer eines Ausgangs-Elements Siehe Kapitel ID-Regeln [▶ 75].

Elemente	PortId	PortName	Alias	Type
minOcc	1	0	0	0
maxOcc	1	1	1	1
Datentyp	HexDecUInt32Value	xs:string	xs:string	xs:string
Inhalt	einfach	einfach	einfach	einfach
Einschränkung	/	/	/	BIT, BYTE, USINT, UINT, INT, UDINT, DINT
Beschreibung	Identifikationsnummer des Ports (verwendet im TwinCAT 3 Safety Editor) Siehe Kapitel PortIds [▶ 78].	Port-Name Dies wird beim Import nicht ausgewertet, sondern dient nur der Lesbarkeit	Alias-Name des Elements (zum Beispiel angezeigt als zugewiesene Variable für FB-Eingänge)	Datentyp des Elements

Elemente	Negated	ResetTime	Active	SubstituteValue
minOcc	0	0	0	0

Elemente	Negated	ResetTime	Active	SubstituteValue
maxOcc	1	1	1	1
Datentyp	xs:boolean	xs:unsignedShort	xs:boolean	/
Inhalt	einfach	einfach	einfach	einfach
Beschreibung		Port-Einstellungen: EDM1, EDM2		Ersatzwert für den Fall, dass die TwinSAFE-Gruppe deaktiviert ist

Elemente	DiscTime	MaxDeviation	SubstituteValueNo- tInRun	Comment
minOcc	0	0	0	0
maxOcc	1	1	1	1
Datentyp	xs:int	xs:unsignedShort	xs:long	xs:string
Inhalt	einfach	einfach	einfach	einfach
Beschreibung	Diese Diskrepanz-Zeit sollte nur am ersten Eingang eines Paares angegeben werden, das die Überprüfung der Diskrepanz-Zeit unterstützt	maximale Abweichung für analoge Eingangswerte von den im permanenten Speicher gespeicherten Werten, wenn die TwinSAFE-Gruppe startet	Ersatzwerte für analoge Ausgänge, wenn der Gruppenstatus nicht RUN, DEACTIVATE,... ist	Kommentar zu diesem Element

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<Outports>
  <Element Id="u1_n1_f1_o1">
    <PortId>33554503</PortId>
    <PortName>Error</PortName>
    <Type>BIT</Type>
    <SubstituteValue>0</SubstituteValue>
    <SubstituteValueNotInRun>0</SubstituteValueNotInRun>
  </Element>
  <Element Id="u1_n1_f1_o2">
    <PortId>33620039</PortId>
    <PortName>StuckAtError</PortName>
    <Type>BIT</Type>
    <SubstituteValue>0</SubstituteValue>
    <SubstituteValueNotInRun>0</SubstituteValueNotInRun>
  </Element>
  <Element Id="u1_n1_f1_o3">
    <PortId>34603079</PortId>
    <PortName>AnalogOut</PortName>
    <Alias>scaleOut</Alias>
    <Type>UINT</Type>
    <SubstituteValue>10</SubstituteValue>
    <SubstituteValueNotInRun>15</SubstituteValueNotInRun>
  </Element>
</Outports>
```

7.1.2.4.1.1.1.3 Inports

Das Eingangsprozessabbild „Inports“ entspricht in seinen Attributen und Elemente dem Ausgangsprozessabbild „Outports“. Für weitere Informationen sehen Sie daher in das Kapitel [Outports](#) [► 40].

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<Inports>
  <Element Id="u1_n1_f1_i1">
    <PortId>17825863</PortId>
    <PortName>AnalogIn</PortName>
    <Alias>scaleIn</Alias>
    <Type>UINT</Type>
```

```
<Negated>>false</Negated>
<Active>>true</Active>
<MaxDeviation>0</MaxDeviation>
</Element>
</Imports>
```

7.1.2.4.1.1.2 UserFunctionBlock

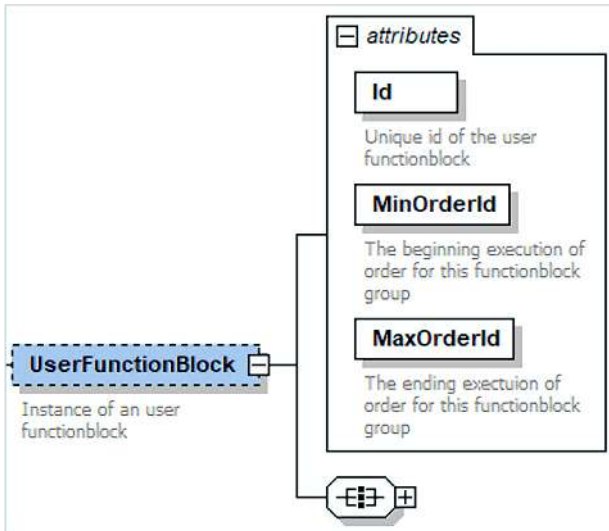


Abb. 20: UserFunctionBlock Attribute

Attribute	Id	MinOrderId	MaxOrderId
Datentyp	xs:ID	xs:unsignedInt	xs:unsignedInt
Verwendung	erforderlich	erforderlich	erforderlich
Beschreibung	Identifikationsnummer des FBs Siehe Kapitel ID-Regeln [▶ 75].	Minimale Order of Execution	Maximale Order of Execution

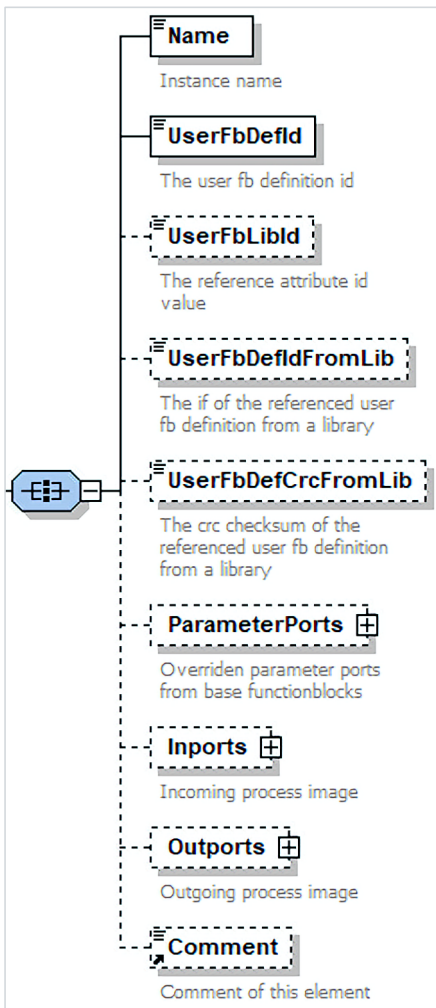


Abb. 21: UserFunctionBlock Elemente

Elemente	Name	UserFbDefId	UserFbLibId	UserFbDefIdFromLib	UserFbDefCrcFromLib
minOcc	1	1	0	0	0
maxOcc	1	1	1	1	1
Datentyp	xs:string	xs:string	xs:string	xs:string	xs:unsignedInt
Inhalt	einfach	einfach	einfach	einfach	einfach
Beschreibung	Name der Instanz	Identifikationsnummer des User FBs Wird gebildet aus Reference.Id + "_" UserFB.Id (Beispiel: r1_u1) Siehe Reference.Id-Attribut in Kapitel References [▶ 24] und UserFB.Id-Attribut in Kapitel UserFbs [▶ 32] .	Identifikationsnummer des Referenzattributs Siehe Reference.Id-Attribut in Kapitel References [▶ 24] .	Identifikationsnummer der referenzierten User-FB-Definition aus einer Bibliothek	CRC-Prüfsumme der referenzierten User-FB-Definition aus einer Bibliothek

Elemente	ParameterPorts	Inports	Outports	Comment
minOcc	0	0	0	0
maxOcc	1	1	1	1

Elemente	ParameterPorts	Inports	Outports	Comment
Datentyp	/	/	/	xs:string
Inhalt	komplex	komplex	komplex	einfach
Beschreibung	Überschriebene Parameter-Ports vom Basis-FB Siehe Kapitel <u>ParameterPorts</u> [▶ 46].	Eingangs- prozessabbild Siehe Kapitel <u>Inports</u> [▶ 48].	Ausgangs- prozessabbild Siehe Kapitel <u>Outports</u> [▶ 49].	Kommentar zu diesem Element

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<UserFunctionBlock Id="g1_n1_u2" MinOrderId="2" MaxOrderId="4">
  <Name>UserFB</Name>
  <UserFbDefId>r1_u1</UserFbDefId>
  <UserFbLibId>r1</UserFbLibId>
  <UserFbDefIdFromLib>88e7f0fe-8ac3-4eab-b02e-cb7df03efbdb</UserFbDefIdFromLib>
  <UserFbDefCrcFromLib>3494585285</UserFbDefCrcFromLib>
  <ParameterPorts>
  </ParameterPorts>
  <Inports>
  </Inports>
  <Outports>
  </Outports>
  <Comment>User FB Test Ton And</Comment>
</UserFunctionBlock>
```

7.1.2.4.1.1.2.1 ParameterPorts

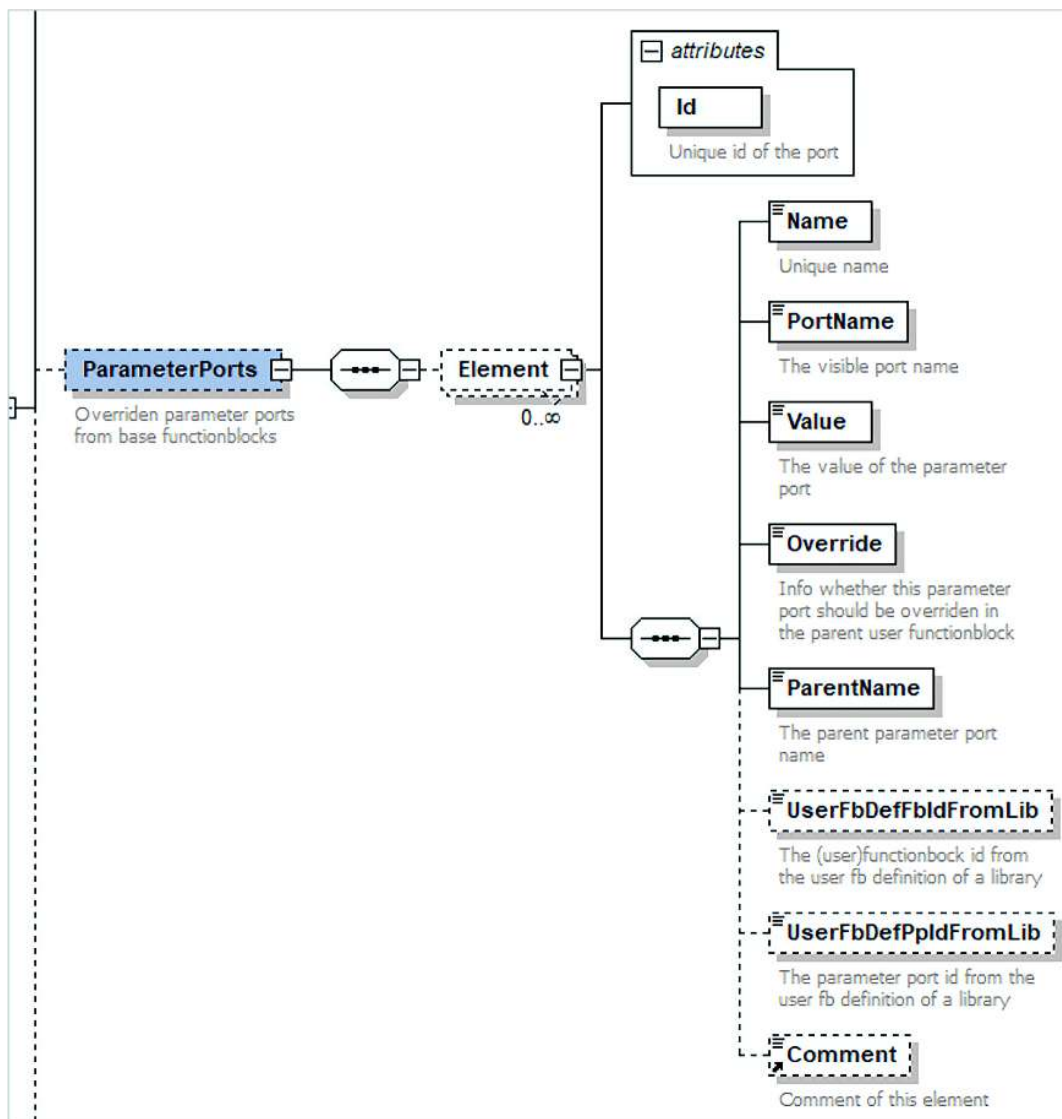


Abb. 22: ParameterPorts

Attribute	Id
Datentyp	xs:ID
Verwendung	erforderlich
Beschreibung	Identifikationsnummer des Ports Siehe Kapitel ID-Regeln [► 75].

Elemente	Name	PortName	Value	Override
minOcc	1	1	1	1
maxOcc	1	1	1	1
Datentyp	xs:string	xs:string	xs:string	xs:boolean
Inhalt	einfach	einfach	einfach	einfach
Beschreibung	Eindeutiger Name	Sichtbarer Name des Ports	Wert des Parameter-Ports	Info, ob dieser Parameter-Port im übergeordneten User FB überschrieben werden soll

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Elemente	ParentName	UserFbDefFbId-FromLib	UserFbDefPpId-FromLib	Comment
minOcc	1	0	0	0
maxOcc	1	1	1	1
Datentyp	xs:string	xs:string	xs:string	xs:string
Inhalt	einfach	einfach	einfach	einfach
Beschreibung	Name des übergeordneten Parameter-Ports	die (User-) FB-Identifikationsnummer aus der User-FB-Definition einer Bibliothek	die Parameter-Port-Identifikationsnummer aus der User-FB-Definition einer Bibliothek	Kommentar zu diesem Element

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<ParameterPorts>
  <Element Id="g1_n1_u2_p1">
    <Name>Network1_FBton1_Delay Time (s)</Name>
    <PortName>FBton1_Delay Time (s)</PortName>
    <Value>2</Value>
    <Override>true</Override>
    <ParentName>FBton1_Delay Time (s)</ParentName>
    <UserFbDefFbIdFromLib>6d61e205-2f4c-4988-8f9d-d0a517134733</UserFbDefFbIdFromLib>
    <UserFbDefPpIdFromLib>aaffefce-ed0a-45ce-aa16-bdee52f4cd58</UserFbDefPpIdFromLib>
  </Element>
  <Element Id="g1_n1_u2_p2">
    <Name>Network1_FBton11_Delay Time (s)</Name>
    <PortName>FBton2_Delay Time (s)</PortName>
    <Value>5</Value>
    <Override>true</Override>
    <ParentName>FBton2_Delay Time (s)</ParentName>
    <UserFbDefFbIdFromLib>68bc8331-0910-4cb1-946f-b4e78cee5a22</UserFbDefFbIdFromLib>
    <UserFbDefPpIdFromLib>97c4e7f3-e650-4da6-90bb-60b7fb549555</UserFbDefPpIdFromLib>
  </Element>
</ParameterPorts>
```

7.1.2.4.1.1.2.2 Inports

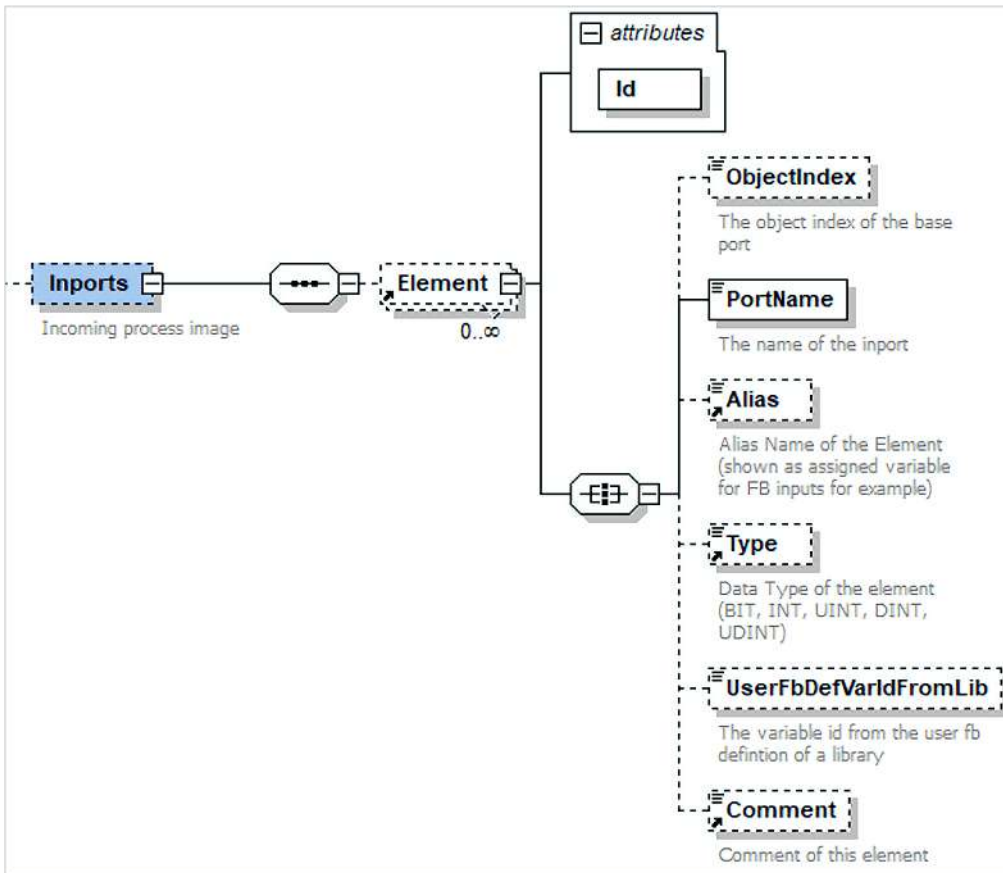


Abb. 23: UserFunctionBlock Inports

Element	Element
minOcc	0
maxOcc	unbegrenzt
Datentyp	/
Inhalt	komplex
Beschreibung	Element im Eingangs-Prozessabbild

Attribut	Id
Datentyp	xs:ID
Verwendung	erforderlich
Beschreibung	Identifikationsnummer eines Elements Siehe Kapitel ID-Regeln [▶ 75].

Elemente	ObjectIndex	PortName	Alias
minOcc	0	1	0
maxOcc	1	1	1
Datentyp	xs:int	/	xs:string
Inhalt	einfach	einfach	einfach
Beschreibung	Objektindex des Basis-Ports	Name des Eingangs	Alias-Name des Elements (zum Beispiel als zugewiesene Variable für FB-Eingänge angezeigt)

Elemente	Type	UserFbDefVarIdFromLib	Comment
minOcc	0	0	0

Elemente	Type	UserFbDefVarIdFromLib	Comment
maxOcc	1	1	1
Datentyp	xs:string	xs:string	xs:string
Inhalt	einfach	einfach	einfach
Einschränkung	BIT, BYTE, USINT UINT, INT, UDINT, DINT	/	/
Beschreibung	Datentyp des Elements	Variablen-Identifikationsnummer einer User-FB-Definition aus der Bibliothek	Kommentar zu diesem Element

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<Inports>
  <Element Id="g1_n1_u2_i1">
    <UserFbDefVarIdFromLib>2110a823-9d12-4d01-8b3e-8877bbf961ff</UserFbDefVarIdFromLib>
    <PortName>FBTon1_TonIn</PortName>
    <Type>BIT</Type>
  </Element>
  <Element Id="g1_n1_u2_i2">
    <UserFbDefVarIdFromLib>4ae048c3-839a-435f-b80d-5583e891b1e4</UserFbDefVarIdFromLib>
    <PortName>FBTon2_TonIn</PortName>
    <Type>BIT</Type>
  </Element>
</Inports>
```

7.1.2.4.1.1.2.3 Outputs

Das Ausgangsprozessabbild „Outputs“ entspricht in seinen Attributen und Elemente dem Eingangsprozessabbild „Inports“. Für weitere Informationen sehen Sie daher in das Kapitel [Inports \[► 48\]](#).

Als XML-Code kann dies wie folgt aussehen:

```
<Outputs>
  <Element Id="g1_n1_u2_o1">
    <ObjectIndex>1</ObjectIndex>
    <UserFbDefVarIdFromLib>8a6f4e1f-eca4-4f83-8bc9-f263dd3e7856</UserFbDefVarIdFromLib>
    <PortName>FBTon1_TonOut</PortName>
    <Alias>out1</Alias>
    <Type>BIT</Type>
  </Element>
  <Element Id="g1_n1_u2_o2">
    <ObjectIndex>1</ObjectIndex>
    <UserFbDefVarIdFromLib>5d5de48d-8a8d-4cd3-8bd6-6b262d808729</UserFbDefVarIdFromLib>
    <PortName>FBAnd1_AndOut</PortName>
    <Alias>out2</Alias>
    <Type>BIT</Type>
  </Element>
</Outputs>
```

7.1.2.4.1.2 Comments

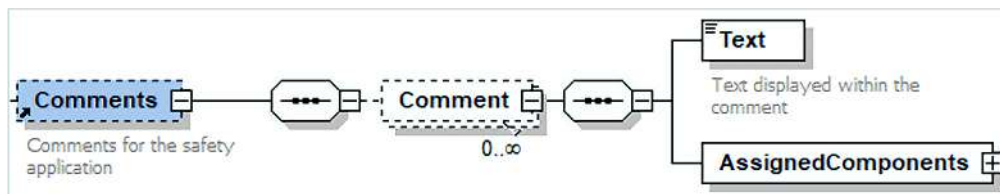


Abb. 24: UserFBs comments

Element	Comment
minOcc	0
maxOcc	unbegrenzt
Inhalt	komplex
Beschreibung	Kommentar

Elemente	Text	AssignedComponents
minOcc	1	1
maxOcc	1	1
Datentyp	xs:string	/
Inhalt	einfach	komplex
Beschreibung	Im Kommentar dargestellter Text	Zugewiesene Komponenten Siehe Kapitel AssignedComponents [► 50].

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<Comments>
  <Comment>
    <Text>TestComment1</Text>
    <AssignedComponents>
      <Component IdRef="u1_n1_f1" />
    </AssignedComponents>
  </Comment>
  <Comment>
    <Text>TestComment2</Text>
    <AssignedComponents>
      <Component IdRef="u1" />
    </AssignedComponents>
  </Comment>
</Comments>
```

7.1.2.4.1.2.1 AssignedComponents

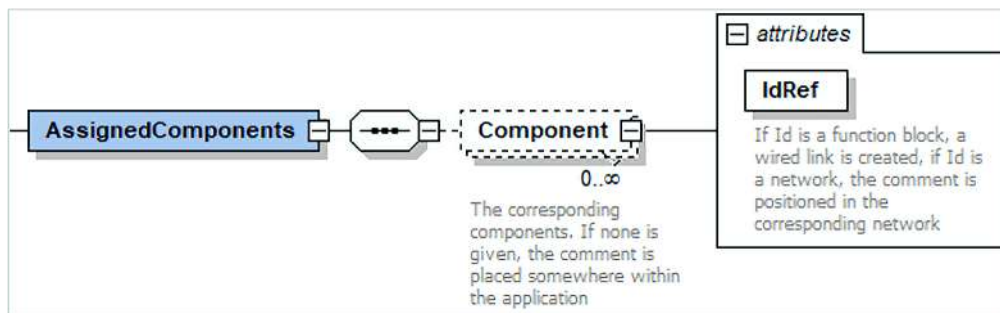


Abb. 25: UserFBs comments

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Element	Component
minOcc	0
maxOcc	unbegrenzt
Inhalt	komplex
Beschreibung	Dazugehörige Komponente

Attribut	IdRef
Datentyp	xs:IDREF
Verwendung	erforderlich
Beschreibung	Falls die Identifikationsnummer ein FB ist, wird eine verdrahtete Verbindung erstellt. Siehe FunctionBlock.Id-Attribut oder UserFunctionBlock.Id-Attribut in Kapitel FunctionBlock [► 35] oder UserFunctionBlock [► 43]. Falls die Identifikationsnummer ein Netzwerk ist, wird der Kommentar im entsprechenden Netzwerk positioniert. Siehe Network.Id-Attribut in Kapitel Network [► 34].

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<AssignedComponents>
  <Component IdRef="u1_n1_f1"/>
</AssignedComponents>
```

7.1.2.4.2 TwinSAFEGroups

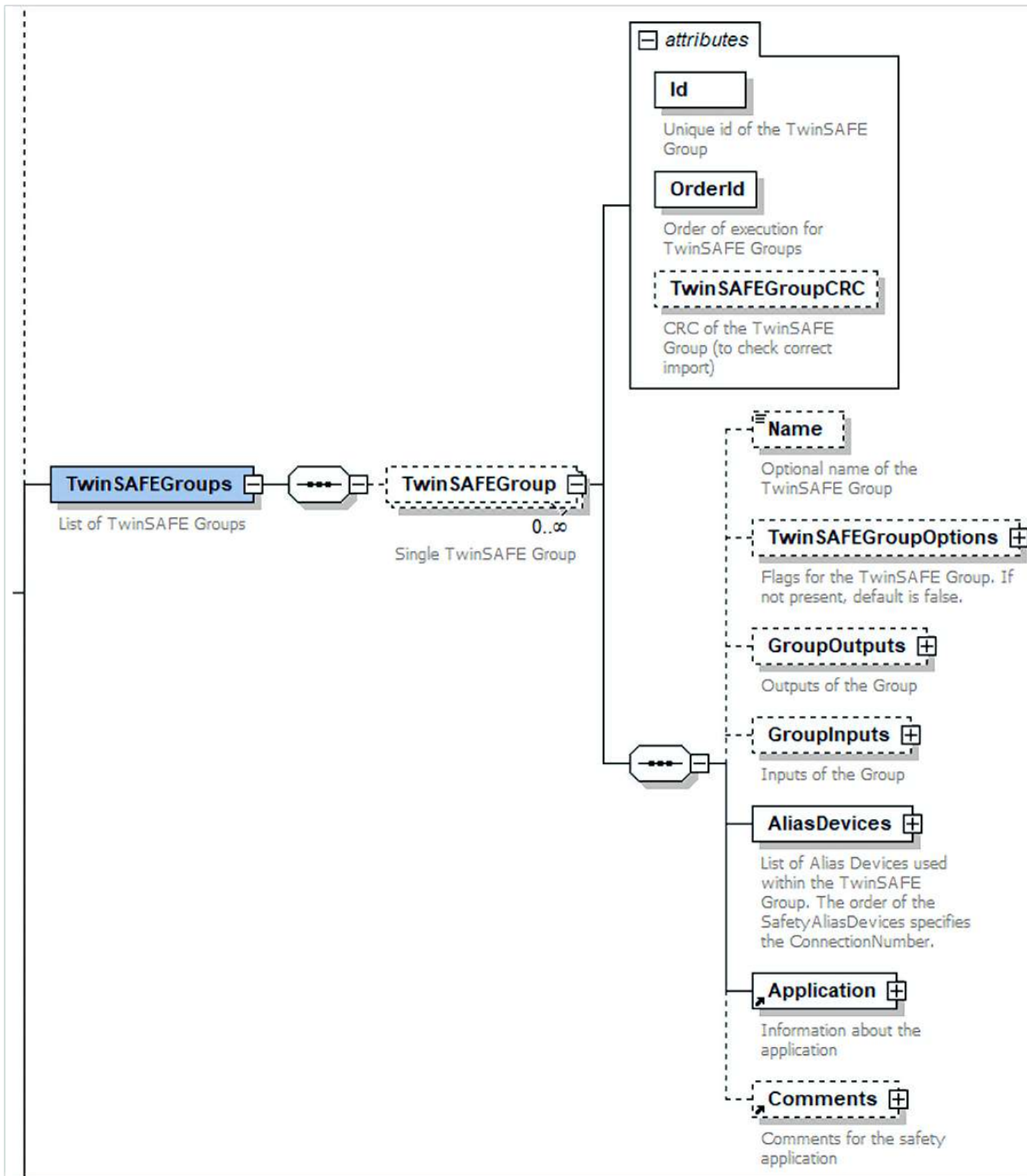


Abb. 26: TwinSAFEGroups

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Element	TwinSAFEGroup
minOcc	0
maxOcc	unbegrenzt
Inhalt	komplex
Beschreibung	Einzelne TwinSAFE-Gruppe

Attribute	Id	OrderId	TwinSAFEGroupCRC
Datentyp	xs:ID	xs:int	xs:int
Verwendung	erforderlich	erforderlich	optional

Attribute	Id	OrderId	TwinSAFEGroupCRC
Beschreibung	Eindeutige Identifikationsnummer der TwinSAFE-Gruppe Siehe Kapitel ID-Regeln [75].	Ausführungsreihenfolge der TwinSAFE-Gruppe	CRC der TwinSAFE-Gruppe Aktuell nicht in Verwendung.

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Elemente	Name	TwinSAFEGroupOptions	GroupOutputs	GroupInputs
minOcc	0	0	0	0
maxOcc	1	1	1	1
Datentyp	xs:string	TwinSAFEGroupOptions	GroupPorts	GroupPorts
Inhalt	einfach	komplex	komplex	komplex
Beschreibung	Name der TwinSAFE-Gruppe	Kennzeichnung der TwinSAFE-Gruppe Siehe Kapitel TwinSAFEGroupOptions [53].	Ausgänge der TwinSAFE-Gruppe Siehe Kapitel GroupOutputs [54].	Eingänge der TwinSAFE-Gruppe Siehe Kapitel GroupInputs [56].

Elemente	AliasDevices	Application	Comments
minOcc	1	1	0
maxOcc	1	1	1
Inhalt	komplex	komplex	komplex
Beschreibung	Liste der Alias-Geräte, die in der TwinSAFE-Gruppe verwendet werden Siehe Kapitel AliasDevices [56].	Informationen über die Applikation Siehe Kapitel Application [65].	Kommentare zur Sicherheitsapplikation Siehe Kapitel Comments [65].

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<TwinSAFEGroups>
  <TwinSAFEGroup Id="g1" OrderId="0">
    <Name>TwinSafeGroup1</Name>
    <TwinSAFEGroupOptions>
    </TwinSAFEGroupOptions>
    <GroupOutputs>
    </GroupOutputs>
    <GroupInputs>
    </GroupInputs>
    <AliasDevices>
    </AliasDevices>
    <Application>
    </Application>
  </TwinSAFEGroup>
  <TwinSAFEGroup Id="g2" OrderId="1">
    <Name>TwinSafeGroup2</Name>
    <TwinSAFEGroupOptions>
    </TwinSAFEGroupOptions>
    <GroupOutputs>
    </GroupOutputs>
    <GroupInputs>
    </GroupInputs>
    <AliasDevices>
    </AliasDevices>
    <Application>
    </Application>
  </TwinSAFEGroup>
</TwinSAFEGroups>
```


7.1.2.4.2.1 TwinSAFEGroupOptions

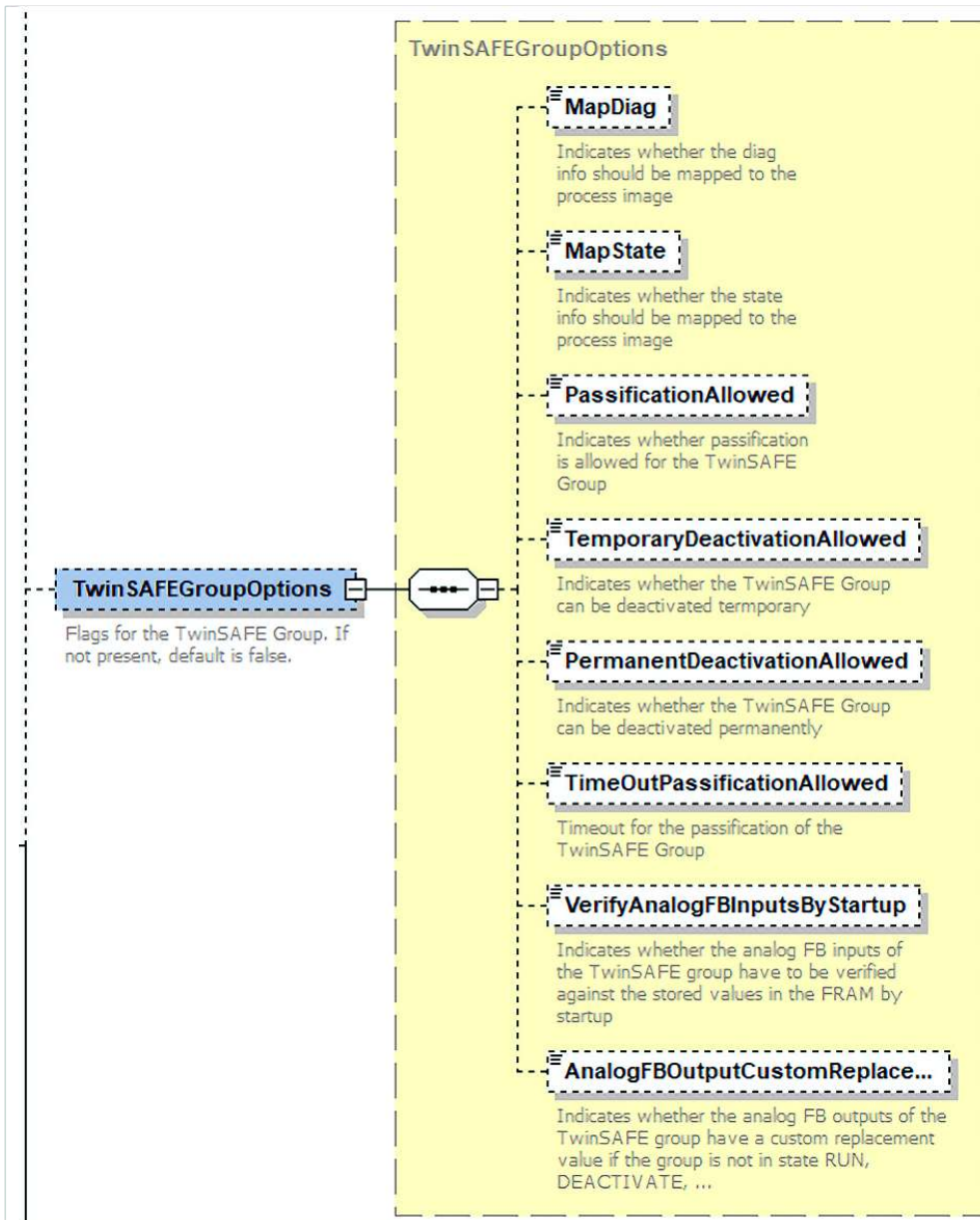


Abb. 27: TwinSAFEGroup Options

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Elemente	MapDiag	MapState	PassificationAllowed	TemporaryDeactivationAllowed
minOcc	0	0	0	0
maxOcc	1	1	1	1
Datentyp	xs:boolean	xs:boolean	xs:boolean	xs:boolean
Inhalt	einfach	einfach	einfach	einfach
Beschreibung	Gibt an, ob die Diag-Information im Prozessabbild angelegt werden soll	Gibt an, ob die Status-Informationen im Prozessabbild angelegt werden soll	Gibt an, ob eine Passivierung für die TwinSAFE-Gruppe erlaubt ist	Gibt an, ob eine vorübergehende Deaktivierung für die TwinSAFE-Gruppe erlaubt ist

Elemente	PermanentDeactivationAllowed	TimeOutPassificationAllowed	VerifyAnalogFBInputsByStartup	AnalogFBOutputCustomReplacementValues
minOcc	0	0	0	0
maxOcc	1	1	1	1
Datentyp	xs:boolean	xs:unsignedShort	xs:boolean	xs:boolean
Inhalt	einfach	einfach	einfach	einfach
Beschreibung	Gibt an, ob eine dauerhafte Deaktivierung für die TwinSAFE-Gruppe erlaubt ist	Timeout für die Passivierung der TwinSAFE-Gruppe	Gibt an, ob die analogen FB-Eingänge der TwinSAFE-Gruppe beim Start mit den gespeicherten Werten im permanenten Speicher abgeglichen werden müssen	Gibt an, ob die analogen FB-Ausgänge der Twinsafe-Gruppe einen benutzerdefinierten Ersatzwert haben, wenn sich die Gruppe nicht im Zustand RUN, DEACTIVATE ... befindet

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```

<TwinSAFEGroupOptions>
  <MapDiag>false</MapDiag>
  <MapState>false</MapState>
  <PassificationAllowed>false</PassificationAllowed>
  <TemporaryDeactivationAllowed>false</TemporaryDeactivationAllowed>
  <PermanentDeactivationAllowed>false</PermanentDeactivationAllowed>
  <TimeOutPassificationAllowed>10000</TimeOutPassificationAllowed>
  <VerifyAnalogFBInputsByStartup>false</VerifyAnalogFBInputsByStartup>
  <AnalogFBOutputCustomReplacementValues>false</AnalogFBOutputCustomReplacementValues>
</TwinSAFEGroupOptions>
    
```

7.1.2.4.2 GroupOutputs

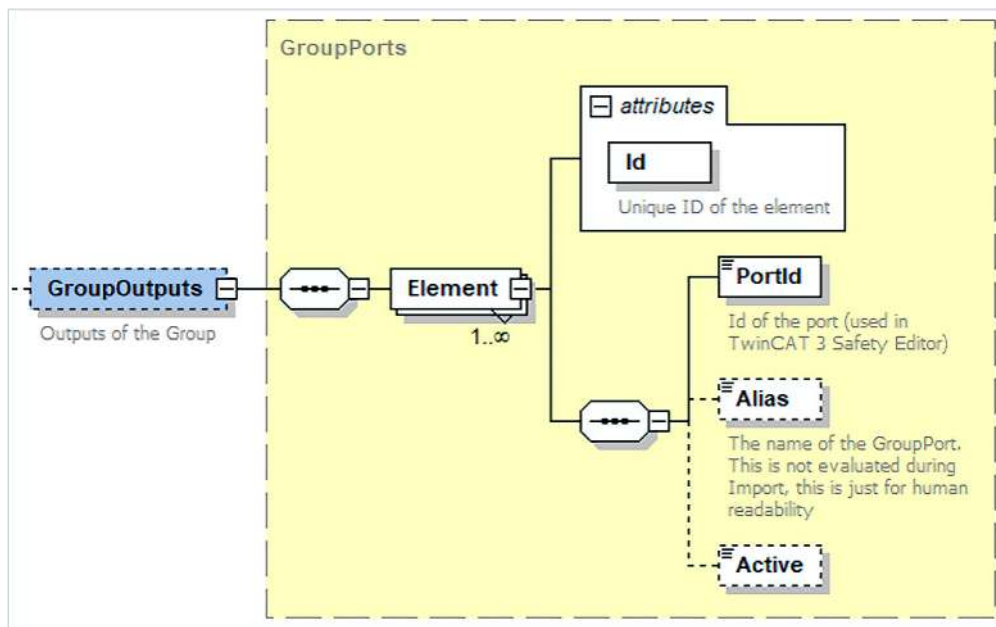


Abb. 28: GroupOutputs

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Element	Element
minOcc	1
maxOcc	unbegrenzt
Inhalt	komplex

Element	Element
Beschreibung	Element eines Gruppenausgangs
Attribute	Id
Datentyp	xs:ID
Verwendung	erforderlich
Beschreibung	Eindeutige Identifikationsnummer des Elements Siehe Kapitel ID-Regeln [► 75].

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Elemente	PortId	Alias	Active
minOcc	1	0	0
maxOcc	1	1	1
Datentyp	HexDecUInt32Value	xs:string	xs:boolean
Inhalt	einfach	einfach	einfach
Beschreibung	Identifikationsnummer des Ports (verwendet im TwinCAT 3 Safety Editor) Siehe Kapitel PortIds [► 78].	Name des Gruppen-Ports Siehe Kapitel PortIds [► 78]. Dies wird beim Import nicht ausgewertet, sondern dient nur der Lesbarkeit.	Zeigt an, ob der Port genutzt wird Dies wird beim Import nicht ausgewertet, sondern dient nur der Lesbarkeit.

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<GroupOutputs>
  <Element Id="g1_o1">
    <PortId>33619968</PortId>
    <Alias>FbErr</Alias>
    <Active>true</Active>
  </Element>
  <Element Id="g1_o2">
    <PortId>33685504</PortId>
    <Alias>ComErr</Alias>
    <Active>true</Active>
  </Element>
  <Element Id="g1_o3">
    <PortId>33751040</PortId>
    <Alias>OutErr</Alias>
    <Active>true</Active>
  </Element>
  <Element Id="g1_o4">
    <PortId>33816576</PortId>
    <Alias>OtherErr</Alias>
    <Active>true</Active>
  </Element>
  <Element Id="g1_o5">
    <PortId>33882112</PortId>
    <Alias>ComStartup</Alias>
    <Active>true</Active>
  </Element>
  <Element Id="g1_o6">
    <PortId>33947648</PortId>
    <Alias>FbDeactive</Alias>
    <Active>true</Active>
  </Element>
  <Element Id="g1_o7">
    <PortId>34013184</PortId>
    <Alias>FbRun</Alias>
    <Active>true</Active>
  </Element>
  <Element Id="g1_o8">
    <PortId>34078720</PortId>
    <Alias>InRun</Alias>
    <Active>true</Active>
  </Element>
</GroupOutputs>
```

7.1.2.4.2.3 GroupInputs

Das Element der Gruppeneingänge „GroupInputs“ entspricht in seinen Attributen und Elemente dem Element der Gruppenausgänge „GroupOutputs“. Für weitere Informationen sehen Sie daher in das Kapitel [GroupOutputs](#) [► 54].

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<GroupInputs>
  <Element Id="g1_i1">
    <PortId>16842752</PortId>
    <Alias>RunStop</Alias>
    <Active>true</Active>
  </Element>
  <Element Id="g1_i2">
    <PortId>16908288</PortId>
    <Alias>ErrAck</Alias>
    <Active>true</Active>
  </Element>
  <Element Id="g1_i3">
    <PortId>16973824</PortId>
    <Alias>ModuleFault</Alias>
    <Active>true</Active>
  </Element>
</GroupInputs>
```

7.1.2.4.2.4 AliasDevices

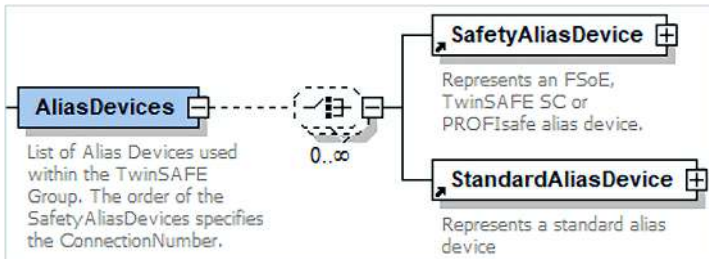


Abb. 29: AliasDevices

Elemente	SafetyAliasDevices	StandardAliasDevices
minOcc	1	1
maxOcc	1	1
Inhalt	komplex	komplex
Beschreibung	Steht für ein FSoE-, TwinSAFE-SC- oder PROFIsafe-Alias-Gerät Siehe Kapitel SafetyAliasDevice [► 57].	Steht für ein Standard-Alias-Gerät Siehe Kapitel StandardAliasDevice [► 63].

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<AliasDevices>
  <StandardAliasDevice Id="g1_a1" OrderId="1">
    </StandardAliasDevice>
  <SafetyAliasDevice Id="g1_a2">
    </SafetyAliasDevice>
  <SafetyAliasDevice Id="g1_a3" IsExternal="true">
    </SafetyAliasDevice>
  <StandardAliasDevice Id="g1_a4" OrderId="2">
    </StandardAliasDevice>
  <StandardAliasDevice Id="g1_a5" OrderId="3">
    </StandardAliasDevice>
  <SafetyAliasDevice Id="g1_a6" IsExternal="false">
    </SafetyAliasDevice>
</AliasDevices>
```

7.1.2.4.2.4.1 SafetyAliasDevice



Abb. 30: SafetyAliasDevices Attribute

Attribute	Id	IsExternal	SafeConnectionCRC
Datentyp	xs:ID	xs:boolean	xs:int
Verwendung	erforderlich	optional	optional
Beschreibung	Identifikationsnummer des Alias-Geräts Siehe Kapitel ID-Regeln [▶ 75].	Kennzeichen, dass ein Gerät nicht im I/O-Baum vorhanden ist (TwinCAT 3 spezifisch)	CRC der sicheren Verbindung Aktuell nicht in Verwendung.

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

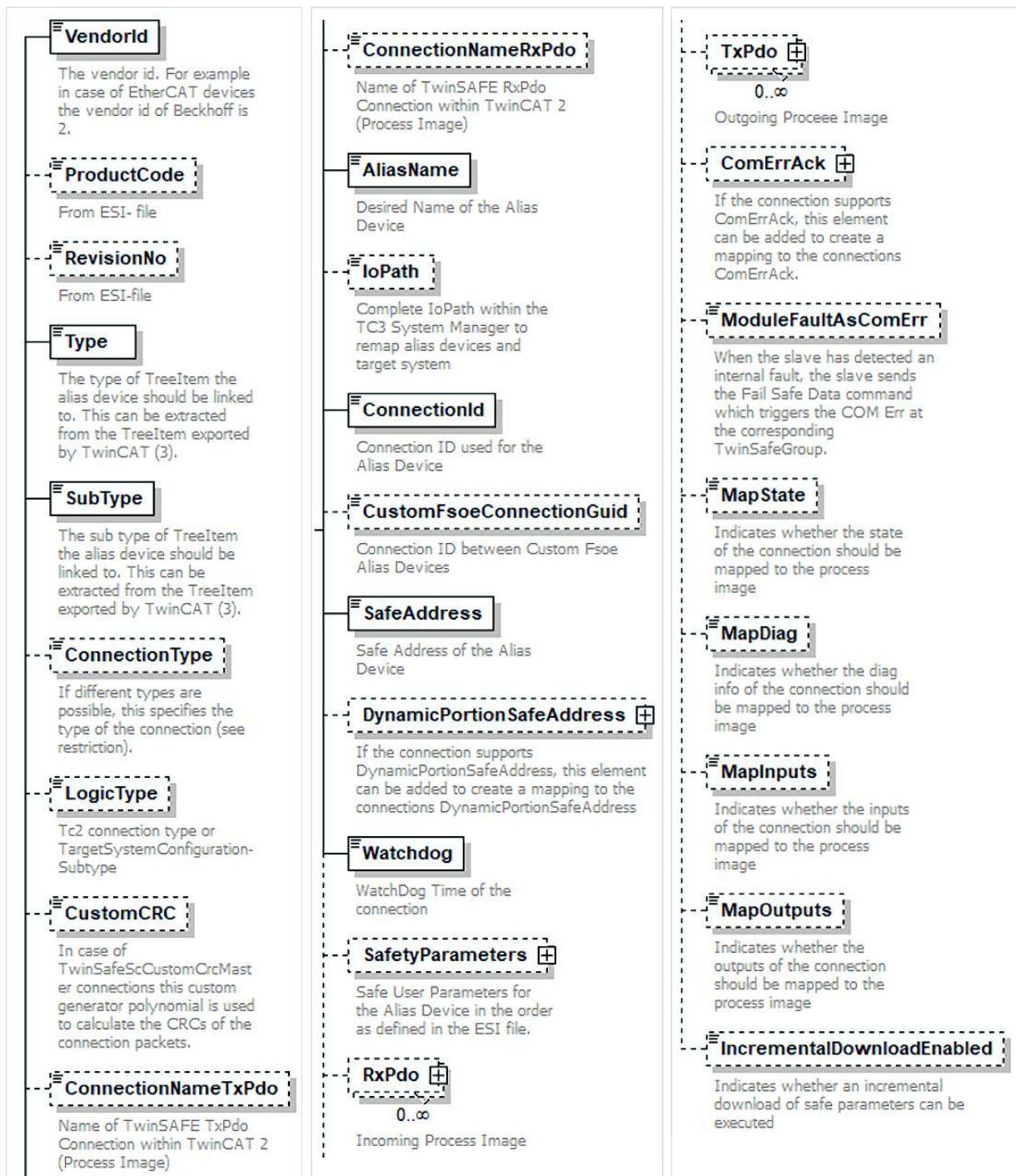


Abb. 31: SafetyAliasDevices Elemente

Elemente	VendorId	ProductCode	RevisionNo	Type
minOcc	1	0	0	1
maxOcc	1	1	1	1
Datentyp	xs:int	xs:int	xs:int	xs:int
Inhalt	einfach	einfach	einfach	einfach
Beschreibung	Hersteller-Identifikationsnummer	Produkt-Code Wird im TwinSAFE-Editor nicht ausgewertet	Revisionsnummer Wird im TwinSAFE-Editor nicht ausgewertet	Gerätetyp

Elemente	SubType	ConnectionType	LogicType
minOcc	1	0	0

Elemente	SubType	ConnectionType	LogicType
maxOcc	1	1	1
Datentyp	xs:int	xs:string	xs:string
Inhalt	einfach	einfach	einfach
Einschränkung	/	FSoEMaster, FSoESlave, TwinSafeScCrc1master, TwinSafeScCrc2master, TwinSafeScCrc3master, TwinSafeScCrc4master, TwinSafeScCrc5master, TwinSafeScCrc6master, TwinSafeScCrc7master, TwinSafeScCrc8master, TwinSafeScCustomCrcMaster, PROFIsafeMaster, PROFIsafeSlave, Local	Mögliche Einstellungen: <ul style="list-style-type: none"> • None • KL6904 • EL6900EL6930 • EL6910EJ6910 • Alle in TargetSystemConfiguration /SubType genannten Zielsysteme ausgenommen EL6900, EL6930, EL6910 und EJ6910
Beschreibung	Subtyp des Geräts	Spezifischer Verbindungstyp, falls verschiedene Verbindungstypen verfügbar sind Siehe Kapitel ConnectionType [► 88].	TwinCAT 3-Verbindungstyp oder -Subtyp der TargetSystemConfiguration Siehe Kapitel LogicType [► 89].

Elemente	CustomCRC	ConnectionNameTxPdo	ConnectionNameRxPdo	AliasName
minOcc	0	0	0	1
maxOcc	1	1	1	1
Datentyp	xs:unsignedInt	xs:string	xs:string	xs:string
Inhalt	einfach	einfach	einfach	einfach
Beschreibung	Im Falle von TwinSafeScCustomCrcMaster-Verbindungen wird dieses benutzerdefinierte Generator-Polynom zur Berechnung der CRCs der Verbindungspakete verwendet	Name der TwinSAFE-TxPdo-Verbindung in TwinCAT 2 (Prozessabbild) Aktuell nicht in Verwendung.	Name der TwinSAFE-RxPdo-Verbindung in TwinCAT 2 (Prozessabbild) Aktuell nicht in Verwendung.	Gewünschter Name des Alias-Geräts

Elemente	IoPath	ConnectionId	CustomFsoe-ConnectionGuid	SafeAddress
minOcc	0	1	0	1
maxOcc	1	1	1	1
Datentyp	xs:string	xs:positiveInteger	xs:string	xs:unsignedShort
Inhalt	einfach	einfach	einfach	einfach
Beschreibung	Gesamter I/O-Pfad innerhalb des TwinCAT 3 System Manager um Alias-Geräte und Zielsysteme neu zuzuordnen	Identifikationsnummer der Verbindung, die für das Alias-Gerät verwendet wird	Identifikationsnummer der Verbindung zwischen benutzerdefinierten FSoE-Alias-Geräten	Sichere Adresse des Alias-Geräts

Elemente	DynamicPortionSafeAddress	Watchdog	SafetyParameters	RxPdo
minOcc	0	1	0	0
maxOcc	1	1	1	unbegrenzt
Datentyp	/	xs:unsignedShort	/	ProcessImage
Inhalt	komplex	einfach	komplex	komplex
Beschreibung	Falls ein SafetyAliasDevice DynamicPortionSafeAddress unterstützt, kann mit diesem Element ein Mapping erzeugt werden Siehe Kapitel DynamicPortionSafeAddress [► 61].	Watchdog-Zeit der Verbindung	Sichere Benutzer-Parameter für die Alias-Geräte Siehe Kapitel SafetyParameters [► 61].	Eingangsprozessabbild Siehe Kapitel RxPdo [► 62].

Elemente	TxPdo	ComErrAck	ModuleFaultAsComErr	MapState
minOcc	0	0	0	0
maxOcc	unbegrenzt	1	1	1
Datentyp	ProcessImage	/	xs:boolean	xs:boolean
Inhalt	komplex	komplex	einfach	einfach
Beschreibung	Ausgangsprozessabbild Siehe Kapitel TxPdo [► 62].	Falls die Verbindung ComErrAck unterstützt, kann dieses Element hinzugefügt werden, um eine Zuordnung zu den Verbindungen ComErrAck zu erstellen Siehe Kapitel ComErrAck [► 62].	Wenn der Slave einen internen Fehler erkannt hat, sendet der Slave das Fail-Safe-Data-Kommando, das den COM Err an der entsprechenden TwinSAFE-Gruppe auslöst	Gibt an, ob die Status-Informationen der Verbindung im Prozessabbild angelegt werden sollen

Elemente	MapDiag	MapInputs	MapOutputs	IncrementalDownloadEnabled
minOcc	0	0	0	0
maxOcc	1	1	1	1
Datentyp	xs:boolean	xs:boolean	xs:boolean	xs:boolean
Inhalt	einfach	einfach	einfach	einfach
Beschreibung	Gibt an, ob die Diag-Informationen der Verbindung im Prozessabbild angelegt werden sollen	Gibt an, ob die Verbindungseingänge im Prozessabbild angelegt werden sollen Auswahl der SafetyAliasDevices über TxPdo und RxPdo	Gibt an, ob die Verbindungsausgänge im Prozessabbild angelegt werden sollen	Gibt an, ob ein inkrementeller Download der Safety-Parameter ausführbar ist

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<SafetyAliasDevice Id="g1_a6" IsExternal="false">
  <VendorId>2</VendorId>
  <Type>60</Type>
  <SubType>17827710</SubType>
  <ConnectionType>FSoEMaster</ConnectionType>
  <LogicType>None</LogicType>
```



```

<AliasName>EL1918, 8 digital inputs (FW 2)_1</AliasName>
<ConnectionId>9</ConnectionId>
<SafeAddress>23</SafeAddress>
<DynamicPortionSafeAddress Id="g1_a6_d" />
<Watchdog>100</Watchdog>
<SafetyParameters>
</SafetyParameters>
<RxPdo>
</RxPdo>
<TxPdo>
</TxPdo>
<ComErrAck Id="g1_a6_c" />
<ModuleFaultAsComErr>>false</ModuleFaultAsComErr>
<MapState>>false</MapState>
<MapDiag>>false</MapDiag>
<MapInputs>>false</MapInputs>
<MapOutputs>>false</MapOutputs>
<IncrementalDownloadEnabled>>false</IncrementalDownloadEnabled>
</SafetyAliasDevice>
    
```

7.1.2.4.2.4.1.1 DynamicPortionSafeAddress

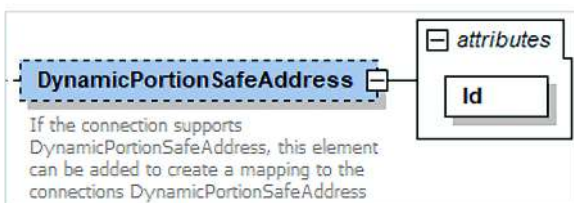


Abb. 32: DynamicPortSafeAddress

Attribute	Id
Datentyp	xs:ID
Verwendung	erforderlich
Beschreibung	Identifikationsnummer der sicheren Adresse Siehe Kapitel ID-Regeln [► 75].

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```

<DynamicPortionSafeAddress Id="g1_a6_d" />
    
```

7.1.2.4.2.4.1.2 SafetyParameters

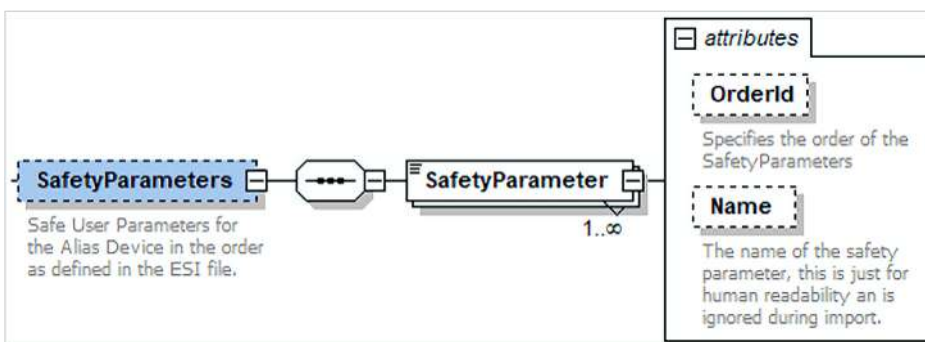


Abb. 33: SafetyParameters

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Element	SafetyParameter
minOcc	1
maxOcc	unbegrenzt
Datentyp	xs:string
Inhalt	komplex

Element	SafetyParameter	
Beschreibung	Safety-Parameter-Wert	
Attribute	OrderId	Name
Datentyp	xs:int	xs:string
Verwendung	optional	optional
Beschreibung	Legt die Reihenfolge der Safety-Parameter fest	Name des Safety-Parameters Dies wird beim Import nicht ausgewertet, sondern dient nur der Lesbarkeit

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<SafetyParameters>
  <SafetyParameter OrderId="1" Name="1018:01 Identity:Vendor ID">2</SafetyParameter>
  <SafetyParameter OrderId="2" Name="F050:01 Detected modules:SubIndex 001">17827710</
SafetyParameter>
  <SafetyParameter OrderId="3" Name="8000:01 FSIN Module 1 Settings Common:ModuloDiagTestPulse">0</
SafetyParameter>
  <SafetyParameter OrderId="4" Name="8000:02 FSIN Module 1 Settings
Common:MultiplierDiagTestPulse">1</SafetyParameter>
  <SafetyParameter OrderId="5" Name="8000:04 FSIN Module 1 Settings Common:Diag TestPulse
active">1</SafetyParameter>
  <SafetyParameter OrderId="6" Name="8000:05 FSIN Module 1 Settings Common:Module Fault Link
active">1</SafetyParameter>
  <SafetyParameter OrderId="7" Name="8001:01 FSIN Module 1 Settings Channel:InputFilterTime">10</
SafetyParameter>
  <SafetyParameter OrderId="8" Name="8001:02 FSIN Module 1 Settings
Channel:DiagTestPulseFilterTime">2</SafetyParameter>
  ...
</SafetyParameters>
```

7.1.2.4.2.4.1.3 RxPdo

Das Eingangsprozessabbild „RxPdo“ der Safety-Alias-Geräte entspricht in seinen Attributen und Elemente dem Ausgangsprozessabbild „TxPdo“ des Zielsystems. Für weitere Informationen sehen Sie daher in das Kapitel [TxPdo](#) [► 29].

7.1.2.4.2.4.1.4 TxPdo

Das Ausgangsprozessabbild „TxPdo“ der Safety-Alias-Geräte entspricht in seinen Attributen und Elemente dem Ausgangsprozessabbild „TxPdo“ des Zielsystems. Für weitere Informationen sehen Sie daher in das Kapitel [TxPdo](#) [► 29].

7.1.2.4.2.4.1.5 ComErrAck

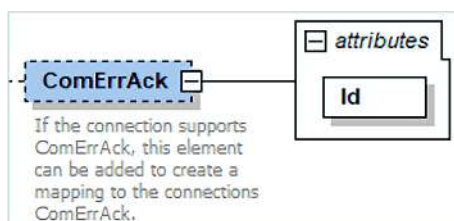


Abb. 34: ComErrAck

Attribute	Id
Datentyp	xs:ID
Verwendung	erforderlich
Beschreibung	Identifikationsnummer Siehe Kapitel ID-Regeln [► 75].

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<ComErrAck Id="g1_a6_c" />
```

7.1.2.4.2.4.2 StandardAliasDevice

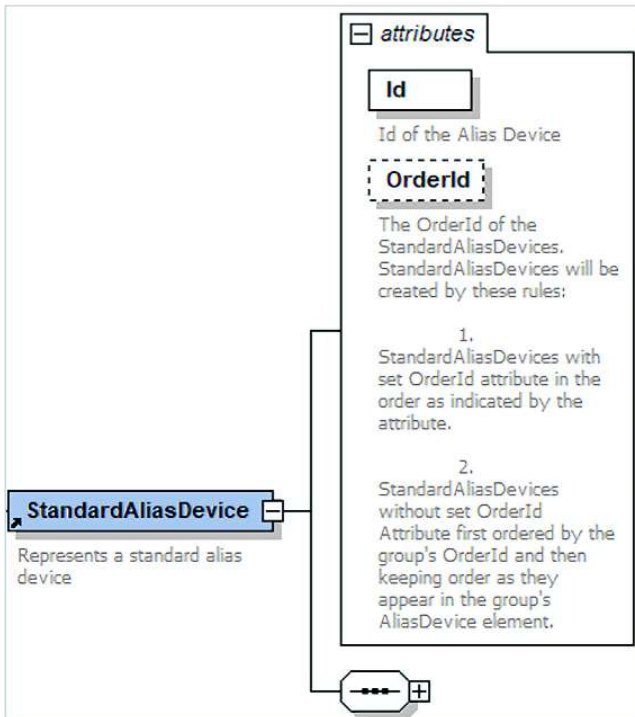


Abb. 35: StandardAliasDevice Attribute

Attribute	Id	OrderId
Datentyp	xs:ID	xs:unsignedInt
Verwendung	erforderlich	optional
Beschreibung	Identifikationsnummer des Alias-Geräts Siehe Kapitel ID-Regeln [► 75].	Reihenfolge-Identifikationsnummer des Standard-Alias-Geräts Standard-Alias-Geräte werden nach den folgenden Regeln erstellt: <ol style="list-style-type: none"> Standard-Alias-Geräte mit gesetztem OrderId-Attribut in der durch das Attribut angegebenen Reihenfolge Standard-Alias-Geräte ohne gesetztem OrderId-Attribut zunächst in der Reihenfolge der OrderId der Gruppe und dann in der Reihenfolge, in der sie im AliasDevice-Element der Gruppe erscheinen

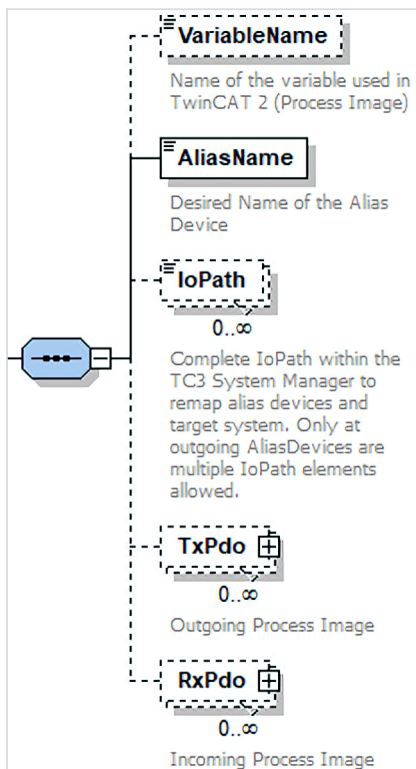


Abb. 36: StandardAliasDevice Elemente

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Elemente	VariableName	AliasName	IoPath	TxPdo	RxPdo
minOcc	0	1	0	0	0
maxOcc	1	1	unbegrenzt	unbegrenzt	unbegrenzt
Datentyp	xs:string	xs:string	xs:string	ProcessImage	ProcessImage
Inhalt	einfach	einfach	einfach	komplex	komplex
Beschreibung	Variablen-Name wie er in TwinCAT 2 verwendet wird (Prozessabbild) Aktuell nicht in Verwendung.	Gewünschter Name des Alias-Geräts	Gesamter I/O-Pfad innerhalb des TwinCAT 3 System Manager um Alias-Geräte und Zielsysteme neu zuordnen	Ausgangsprozessabbild Siehe Kapitel TxPdo [► 64].	Eingangsprozessabbild Siehe Kapitel RxPdo [► 65].

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<StandardAliasDevice Id="g1_a1" OrderId="1">
  <AliasName>ErrorAcknowledgement</AliasName>
  <IoPath>TIRT^Task 2^Outputs^ErrAck</IoPath>
  <RxPdo BitSize="1">
    <Element Id="g1_a1_i1">
      <Name>In</Name>
      <Type>BIT</Type>
      <BitOffset>0</BitOffset>
      <BitSize>1</BitSize>
      <IsSafeTimer>>false</IsSafeTimer>
    </Element>
  </RxPdo>
</StandardAliasDevice>
```

7.1.2.4.2.4.2.1 TxPdo

Das Ausgangsprozessabbild „TxPdo“ der Standard-Alias-Geräte entspricht in seinen Attributen und Elemente dem Ausgangsprozessabbild „TxPdo“ des Zielsystems. Für weitere Informationen sehen Sie daher in das Kapitel [TxPdo](#) [► 29].

7.1.2.4.2.4.2 RxPdo

Das Eingangsprozessabbild „RxPdo“ der Standard-Alias-Geräte entspricht in seinen Attributen und Elemente dem Ausgangsprozessabbild „TxPdo“ des Zielsystems. Für weitere Informationen sehen Sie daher in das Kapitel [TxPdo](#) [▶ 29].

7.1.2.4.2.5 Application

Die Applikation der TwinSAFE-Gruppen entspricht in ihren Attributen und Elemente der Applikation der UserFBs. Für weitere Informationen sehen Sie daher in das Kapitel [Application](#) [▶ 33].

7.1.2.4.2.6 Comments

Die Kommentare zu den TwinSAFE-Gruppen entspricht in ihren Attributen und Elemente der Kommentaren zu den UserFBs. Für weitere Informationen sehen Sie daher in das Kapitel [Comments](#) [▶ 49].

7.1.2.4.3 UserFbPortRelations

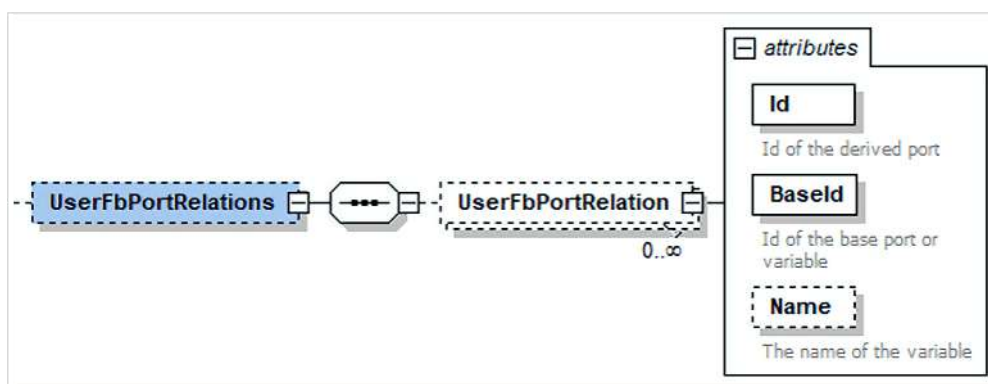


Abb. 37: UserFBPortRelations

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Elemente	UserFbPortRelation
minOcc	0
maxOcc	unbegrenzt
Inhalt	komplex
Beschreibung	Beziehung zwischen den User FB Ports

Attribute	Id	BaseId	Name
Datentyp	xs:string	xs:string	xs:string
Verwendung	erforderlich	erforderlich	optional
Beschreibung	Identifikationsnummer des abgeleiteten Ports Siehe Kapitel ID-Regeln [▶ 75].	Identifikationsnummer des Basis-Ports oder der Variable	Name der Variable

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<UserFbPortRelations>
  <UserFbPortRelation Id="g1_n1_u2_p1" BaseId="r1_u1_n1_f1_p1" />
  <UserFbPortRelation Id="g1_n1_u2_p2" BaseId="r1_u1_n1_f3_p1" />
  <UserFbPortRelation Id="g1_n1_u2_i1" BaseId="r1_u1_n1_f1_i1" Name="FBTon1_TonIn" />
  <UserFbPortRelation Id="g1_n1_u2_i2" BaseId="r1_u1_n1_f3_i1" Name="FBTon2_TonIn" />
  <UserFbPortRelation Id="g1_n1_u2_o1" BaseId="r1_u1_n1_f1_o1" Name="FBTon1_TonOut" />
  <UserFbPortRelation Id="g1_n1_u2_o2" BaseId="r1_u1_n1_f2_o1" Name="FBAnd1_AndOut" />
  <UserFbPortRelation Id="g1_n2_u2_p1" BaseId="u1_n1_f1_p1" />
  <UserFbPortRelation Id="g1_n2_u2_p2" BaseId="u1_n1_f1_p2" />
  <UserFbPortRelation Id="g1_n2_u2_p3" BaseId="u1_n1_f1_p3" />
</UserFbPortRelations>
```

```
<UserFbPortRelation Id="g1_n2_u2_i1" BaseId="u1_n1_f1_i1" Name="scaleIn" />
<UserFbPortRelation Id="g1_n2_u2_o1" BaseId="u1_n1_f1_o3" Name="scaleOut" />
</UserFbPortRelations>
```

7.1.2.4.4 Mappings

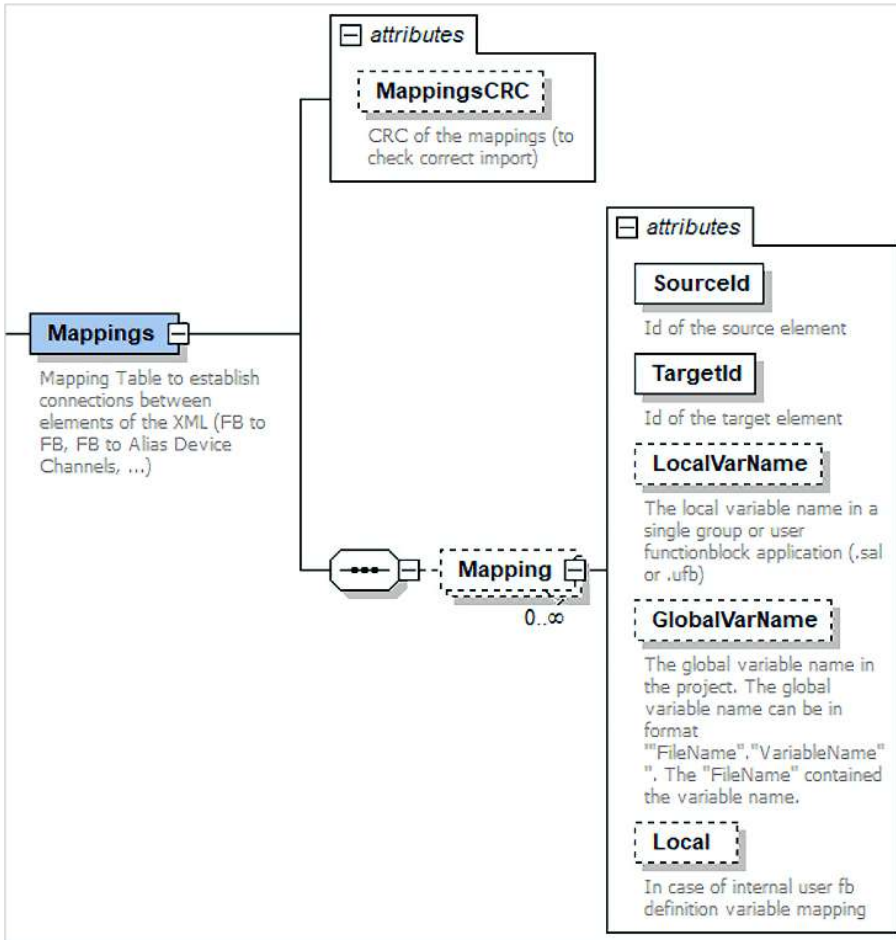


Abb. 38: Mappings

Attribut	MappingsCRC
Datentyp	xs:int
Verwendung	optional
Beschreibung	CRC der Verknüpfung Aktuell nicht in Verwendung.

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Element	Mapping
minOcc	0
maxOcc	unbegrenzt
Inhalt	komplex
Beschreibung	Verknüpfung

Attribute	SourceId	TargetId	LocalVarName	GlobalVarName	Local
Datentyp	xs:IDREF	xs:IDREF	xs:string	xs:string	xs:boolean
Verwendung	erforderlich	erforderlich	optional	optional	optional
Beschreibung	Identifikationsnummer des Quellelements	Identifikationsnummer des Zielelements	Name der Variablen innerhalb einer Gruppe oder	Name der globalen Variablen in einem Projekt	Interne Verknüpfung, bei interner Nutzung der User FBs

Attribute	SourceId	TargetId	LocalVarName	GlobalVarName	Local
			einer User-FB-Definition (*.sal oder *.ufb)	Mögliche Formate: • FileName • VariableName	

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<Mappings>
  <Mapping TargetId="g1_i2" SourceId="g1_a1_i1" GlobalVarName="GVL1.ErrAck"/>
  <Mapping TargetId="g2_i2" SourceId="g1_a1_i1" GlobalVarName="GVL1.ErrAck"/>
  <Mapping TargetId="g1_i1" SourceId="g1_a4_i1" GlobalVarName="GVL1.Run"/>
  <Mapping TargetId="g2_i1" SourceId="g1_a4_i1" GlobalVarName="GVL1.Run"/>
  <Mapping TargetId="g2_n1_f1_i2" SourceId="g1_n2_f1_o1" GlobalVarName="GVL1.SafeOutF1"/>
  <Mapping TargetId="g1_n1_f1_i1" SourceId="g1_a3_i1" LocalVarName="in"/>
  <Mapping TargetId="g1_n2_u2_i1" SourceId="g1_a2_li20" LocalVarName="scaleIn"/>
  <Mapping TargetId="g1_a5_o1" SourceId="g1_n2_u2_o1" LocalVarName="scaleOut"/>
  <Mapping TargetId="g1_n2_f1_i1" SourceId="g1_n1_u2_o1" LocalVarName="out1"/>
  <Mapping TargetId="g1_a2_lo1" SourceId="g1_n1_u2_o2" LocalVarName="out2"/>
  <Mapping TargetId="g1_a2_lo2" SourceId="g1_n1_u2_o2" LocalVarName="out2"/>
  <Mapping TargetId="g1_a2_lo3" SourceId="g1_n1_u2_o2" LocalVarName="out2"/>
  <Mapping TargetId="g1_a2_lo4" SourceId="g1_n1_u2_o2" LocalVarName="out2"/>
  <Mapping TargetId="g1_a2_lo6" SourceId="g1_n1_u2_o2" LocalVarName="out2"/>
  <Mapping TargetId="g1_a3_o1" SourceId="g1_a6_i1" LocalVarName="Variable1"/>
  <Mapping TargetId="g1_n1_u2_i1" SourceId="g1_n1_f1_o1"/>
  <Mapping TargetId="g1_n1_u2_i2" SourceId="g1_n1_f1_o1"/>
</Mappings>
```

7.1.2.5 TargetSystemProcessImage

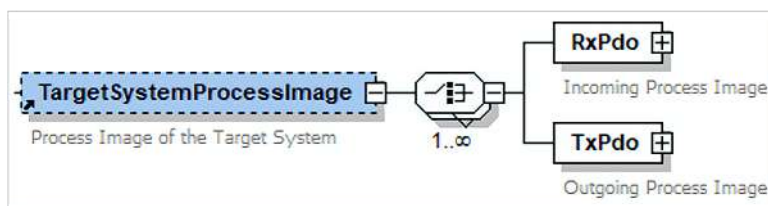


Abb. 39: TargetSystemProcessImage

Elemente	RxPdo	TxPdo
minOcc	1	1
maxOcc	1	1
Inhalt	komplex	komplex
Beschreibung	Eingangsprozessabbild Siehe Kapitel RxPdo [► 68].	Ausgangsprozessabbild Siehe Kapitel TxPdo [► 69].

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<TargetSystemProcessImage>
  <TxPdo BitSize="48" OrderId="1" Name="FSoE Master Connection 1 Master Message">
  </TxPdo>
  <TxPdo BitSize="48" OrderId="2" Name="FSoE Master Connection 2 Master Message">
  </TxPdo>
  ...
  <RxPdo BitSize="48" OrderId="9" Name="FSoE Master Connection 1 Slave Message">
  </RxPdo>
  <RxPdo BitSize="56" OrderId="10" Name="FSoE Master Connection 2 Slave Message">
  </RxPdo>
  ...
</TargetSystemProcessImage>
```


7.1.2.5.1 RxPdo

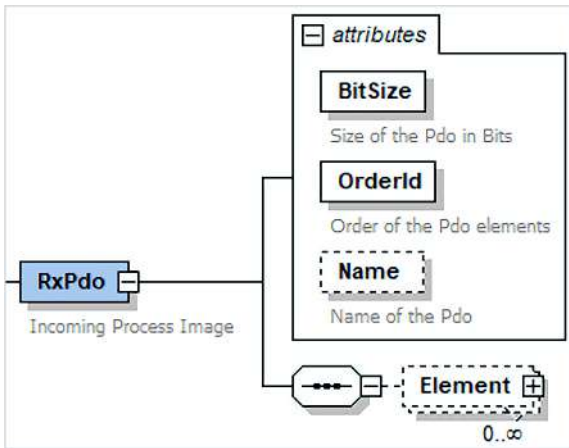


Abb. 40: RxPdo Attribute

Attribute	BitSize	OrderId	Name
Datentyp	xs:int	xs:int	xs:string
Verwendung	erforderlich	erforderlich	optional
Beschreibung	Größe des Prozessabbildes in Bits	Reihenfolge der Prozessabbild-Elemente	Name des Prozessabbildes

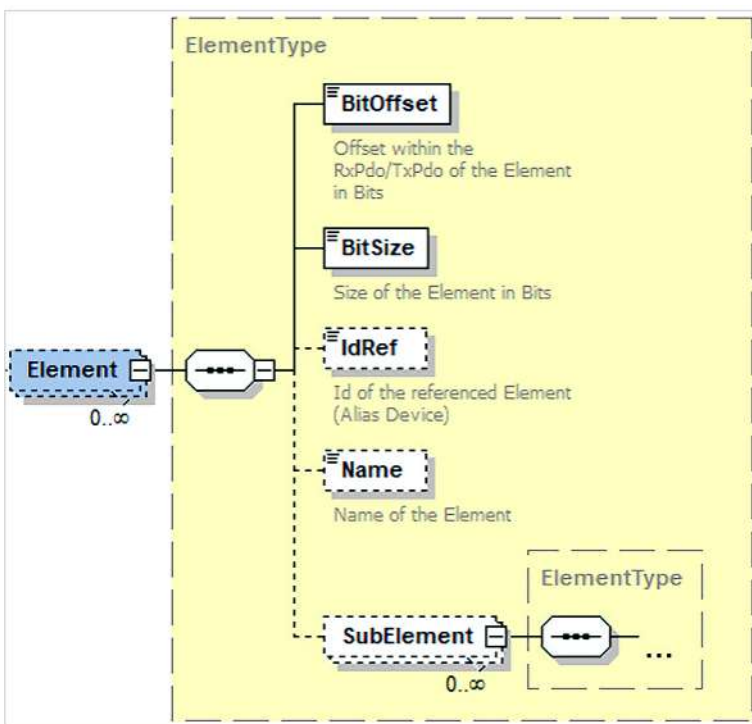


Abb. 41: RxPdo Element

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Element	Element
minOcc	0
maxOcc	unbegrenzt
Datentyp	ElementType
Inhalt	komplex
Beschreibung	Prozessabbild-Element

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Elemente	BitOffset	BitSize	IdRef	Name	SubElement
minOcc	1	1	0	0	0
maxOcc	1	1	1	1	unbegrenzt
Datentyp	xs:int	xs:int	xs:string	xs:string	ElementType
Inhalt	einfach	einfach	einfach	einfach	komplex
Beschreibung	Element-Offset im Prozessabbild in Bits	Element-Größe in Bits	Identifikationsnummer des referenzierten Elements (Alias-Gerät)	Name des Elements	Sub-Element

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<RxPdo BitSize="48" OrderId="9" Name="FSoE Master Connection 1 Slave Message">
  <Element>
    <BitOffset>0</BitOffset>
    <BitSize>48</BitSize>
    <IdRef>g1_a3</IdRef>
    <Name>Message_3</Name>
    <SubElement>
      <BitOffset>0</BitOffset>
      <BitSize>8</BitSize>
      <Name>FSoE_CMD</Name>
    </SubElement>
    ...
  </Element>
</RxPdo>
```

7.1.2.5.2 TxPdo

Das Ausgangsprozessabbild „TxPdo“ entspricht in seinen Attributen und Elemente dem Eingangsprozessabbild „RxPdo“. Für weitere Informationen sehen Sie daher in das Kapitel [RxPdo](#) [► 68].

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<TxPdo BitSize="48" OrderId="1" Name="FSoE Master Connection 1 Master Message">
  <Element>
    <BitOffset>0</BitOffset>
    <BitSize>48</BitSize>
    <IdRef>g1_a3</IdRef>
    <Name>Message_3</Name>
    <SubElement>
      <BitOffset>0</BitOffset>
      <BitSize>8</BitSize>
      <Name>FSoE_CMD</Name>
    </SubElement>
    ...
  </Element>
</TxPdo>
```

7.1.3 Multi Settings

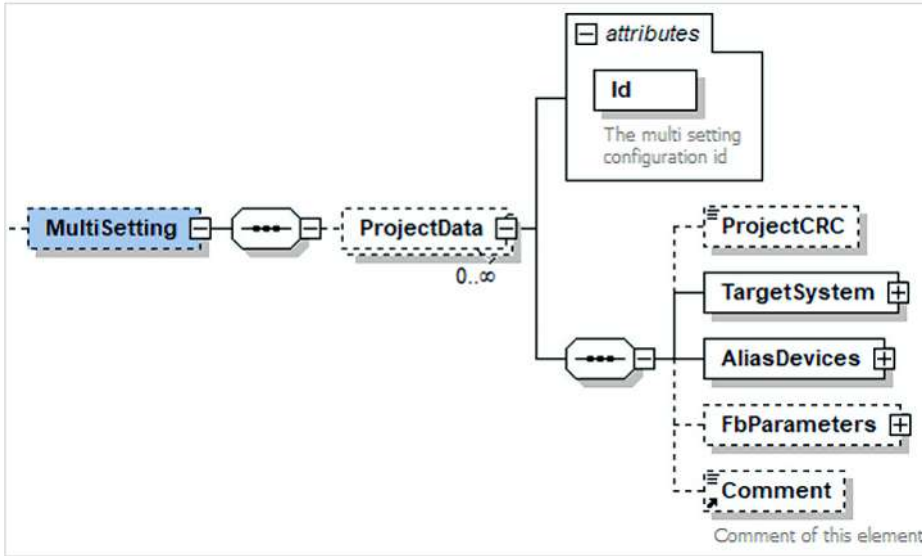


Abb. 42: Multisettings

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Elemente	ProjectData
minOcc	0
maxOcc	unbegrenzt
Inhalt	komplex
Beschreibung	Projektdaten

Attribute	Id
Datentyp	xs:ID
Verwendung	erforderlich
Beschreibung	Identifikationsnummer der Multi-Settings Siehe Kapitel ID-Regeln [▶ 75].

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Elemente	ProjectCRC	TargetSystem	AiasDevices	FbParameters	Comment
minOcc	0	1	1	0	0
maxOcc	1	1	1	1	1
Datentyp	HexDecUInt32Value	/	/	/	xs:string
Inhalt	einfach	komplex	komplex	komplex	einfach
Beschreibung	CRC des Projekts	Zielsystem Siehe Kapitel TargetSystem [▶ 71].	Alias-Gerät Siehe Kapitel AliasDevices [▶ 72].	FB-Parameter Siehe Kapitel FbParameters [▶ 74].	Kommentar zu diesem Element

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<MultiSetting>
  <ProjectData Id="ms_1">
    <ProjectCRC>0xCC9E</ProjectCRC>
    <TargetSystem BaseId="t1">
    </TargetSystem>
    <AliasDevices>
    </AliasDevices>
  </ProjectData>
</MultiSetting>
```

```
<FbParameters>
</FbParameters>
<Comment>
</Comment>
</ProjectData>
</MultiSetting>
```

7.1.3.1 TargetSystem

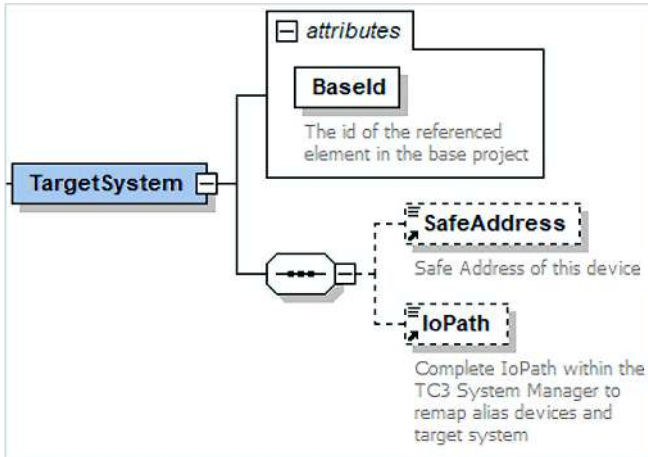


Abb. 43: TargetSystem

Attribute	BaseId
Datentyp	xs:IDREF
Verwendung	erforderlich
Beschreibung	Identifikationsnummer des Referenz-Elements im Basisprojekt Siehe TargetSystemConfiguration.Id-Attribut in Kapitel TargetSystemConfiguration [► 25].

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Elemente	SafeAddress	IoPath
minOcc	0	0
maxOcc	1	1
Datentyp	xs:int	xs:string
Inhalt	einfach	einfach
Beschreibung	Sichere Adresse des Geräts	Gesamter I/O-Pfad innerhalb des TwinCAT 3 System Manager um Alias-Geräte und Zielsysteme neu zuordnen

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<TargetSystem BaseId="t1">
  <SafeAddress>1</SafeAddress>
  <IoPath>TIID^Device 2 (EtherCAT)^Term 30 (AX8620-0000-0104)^Drive 33 (AX8206-0210-0104)</IoPath>
</TargetSystem>
```

7.1.3.2 AliasDevices

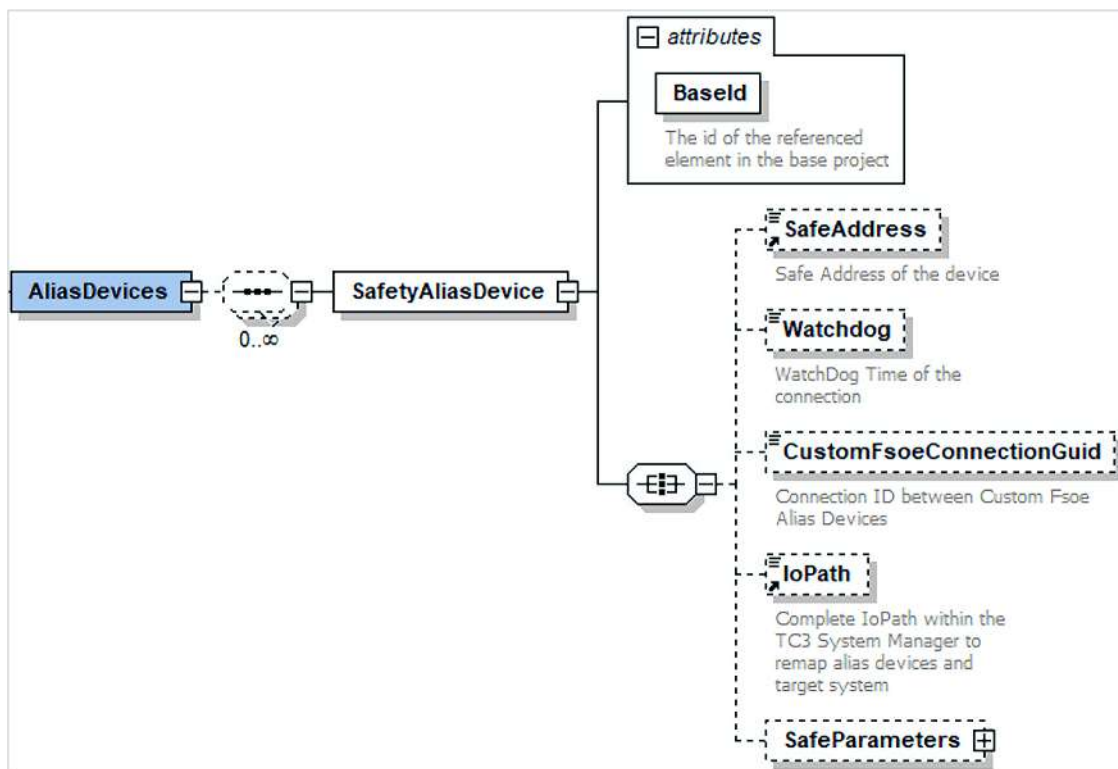


Abb. 44: AliasDevices

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Element	SafetyAliasDevice
minOcc	1
maxOcc	1
Inhalt	komplex
Beschreibung	Sicheres Alias-Gerät

Attribut	BaseId
Datentyp	xs:IDREF
Verwendung	erforderlich
Beschreibung	Identifikationsnummer des Referenz-Elements im Basisprojekt Siehe SafetyAliasDevice.Id-Attribut in Kapitel SafetyAliasDevice [► 57].

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Elemente	SafeAddress	Watchdog	CustomFsoe-ConnectionGuid	IoPath	SafeParameters
minOcc	0	0	0	0	0
maxOcc	1	1	1	1	1
Datentyp	xs:int	xs:unsignedShort	xs:string	xs:string	/
Inhalt	einfach	einfach	einfach	einfach	komplex
Beschreibung	Sichere Adresse des Geräts	Watchdog-Zeit der Verbindung	Identifikationsnummer der Verbindung zwischen benutzerdefinierten FSoE-Alias-Geräten	Gesamter I/O-Pfad innerhalb des TwinCAT 3 System Manager um Alias-Geräte und Zielsysteme neu zuordnen	Sichere Parameter des Geräts Siehe Kapitel SafeParameters [► 73].

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<AliasDevices>
  <SafetyAliasDevice BaseId="g1_a3">
    <SafeAddress>123</SafeAddress>
    <Watchdog>100</Watchdog>
    <CustomFsoeConnectionGuid>44ec90a0-1c84-4e8a-8a28-20811c35db74</CustomFsoeConnectionGuid>
  </SafetyAliasDevice>
  <SafetyAliasDevice BaseId="g1_a6">
    <SafeAddress>23</SafeAddress>
    <Watchdog>100</Watchdog>
    <IoPath>TIID^Device 2 (EtherCAT)^Term 15 (EK1101)^Term 1 (EL1918)^Module 1 (FSOE)</IoPath>
    <SafeParameters>
    </SafeParameters>
  </SafetyAliasDevice>
</AliasDevices>
```

7.1.3.2.1 SafeParameters

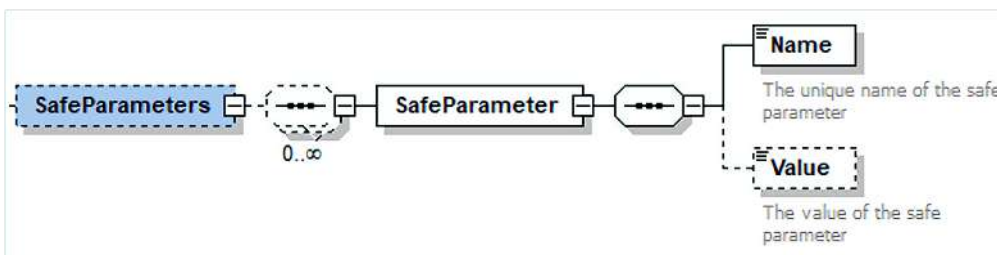


Abb. 45: SafeParameters

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Element	Safe Parameter
minOcc	1
maxOcc	1
Inhalt	komplex
Beschreibung	Sicherer Parameter des Geräts

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Elemente	Name	Value
minOcc	1	0
maxOcc	1	1
Datentyp	xs:string	xs:string
Inhalt	einfach	einfach
Beschreibung	Eindeutiger Name des sicheren Parameters	Wert des sicheren Parameters

Als XML-Code sieht dies zum Beispiel wie folgt aus:

```
<SafeParameters>
  <SafeParameter>
    <Name>32768_1_8_16</Name>
    <Value>0</Value>
  </SafeParameter>
</SafeParameters>
```

7.1.3.3 FbParameters

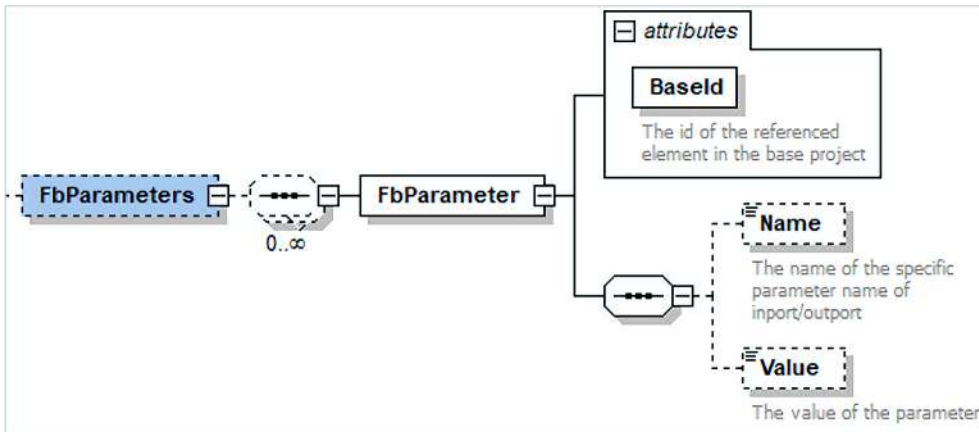


Abb. 46: FbParameters

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Elemente	FbParameter
minOcc	1
maxOcc	1
Inhalt	komplex
Beschreibung	Funktionsbaustein-Parameter

Attribute	Baselid
Datentyp	xs:string
Verwendung	erforderlich
Beschreibung	Identifikationsnummer des Referenz-Elements im Basisprojekt Siehe <ul style="list-style-type: none"> • FunctionBlock.Id-Attribut in Kapitel FunktionBlock [► 35], • FunctionBlock.Outputs.Element.Id-Attribut in Kapitel Outports [► 40], • FunctionBlock.Inports.Element.Id-Attribut in Kapitel Inports [► 42], • UserFunctionBlock.ParameterPorts.Element.Id-Attribut in Kapitel ParameterPorts [► 46].

Halten Sie für eine valide XML-Datei die Reihenfolge der Elemente zwingend ein.

Elemente	Name	Value
minOcc	0	0
maxOcc	1	1
Datentyp	xs:string	xs:string
Inhalt	einfach	einfach
Beschreibung	Name des spezifischen Parameters am Inport oder Output	Wert des Parameters

7.2 ID-Regeln



ID-Benennung

Die aufgelisteten ID-Regeln sind eine Empfehlung. Sie können die IDs frei benennen. Achten Sie dabei auf Eindeutigkeit.

Der TwinSAFE Editor verwendet folgende Konvention für die Erzeugung von Werten für das Attribut „Id“:

ID	Format	Link
Eindeutige Identifikationsnummer der Referenz	r1, r2, ...	Siehe Kapitel References [▶ 24].
Identifikationsnummer des Zielsystems	t1, t2, ...	Siehe Kapitel TargetSystemConfiguration [▶ 25].
Identifikationsnummer eines Elements im Ausgangsprozessabbild des Zielsystems	TargetSystemConfiguration.Id + "_o1", TargetSystemConfiguration.Id + "_o2", ... Beispiel: t1_o1	Siehe Kapitel TxPdo [▶ 29].
Identifikationsnummer eines Elements im Eingangsprozessabbild des Zielsystems	TargetSystemConfiguration.Id + "_i1", TargetSystemConfiguration.Id + "_i2", ... Beispiel: t1_i1	Siehe Kapitel RxPdo [▶ 31].
Identifikationsnummer einer User-FB-Definition	u1, u2, ...	Siehe Kapitel UserFbs [▶ 32].
Identifikationsnummer eines Netzwerks	Innerhalb einer Benutzer FB Definition: UserFB.Id + "_n1", UserFB.Id + "_n2", ... Beispiel: u1_n1 Innerhalb einer Gruppe: TwinSAFEGroup.Id + "_n1", TwinSAFEGroup.Id + "_n2", ... Beispiel: g1_n1	Siehe Kapitel Network [▶ 34].
Identifikationsnummer eines FBs	Network.Id + "_f1", Network.Id + "_f2", ... Beispiele: g1_n1_f1, u1_n1_f1	Siehe Kapitel FunctionBlock [▶ 35].
Identifikationsnummer eines Elements im Ausgangsprozessabbild eines FBs	FunctionBlock.Id + "_o1", FunctionBlock.Id + "_o2", ... Beispiele: g1_n1_f1_o1, u1_n1_f1_o1	Siehe Kapitel Outports [▶ 40].
Identifikationsnummer eines Elements im Eingangsprozessabbild eines FBs	FunctionBlock.Id + "_i1", FunctionBlock.Id + "_i2", ... Beispiele: g1_n1_f1_i1, u1_n1_f1_i1	Siehe Kapitel Inports [▶ 42].
Identifikationsnummer eines User FBs	Network.Id + "_u1", Network.Id + "_u2", ... Beispiele: g1_n1_u1, u1_n1_u1	Siehe Kapitel UserFunctionBlock [▶ 43].
Identifikationsnummer eines Parameter-Ports	UserFB.Id + "_p1", UserFB.Id + "_p2", ... Beispiele: g1_n1_u1_p1, u1_n1_u1_p1	Siehe Kapitel ParameterPorts [▶ 46].
Identifikationsnummer eines Elements im Eingangsprozessabbild eines User FBs	UserFB.Id + "_i1", UserFB.Id + "_i2", ... Beispiele: g1_n1_u1_i1, u1_n1_u1_i1	Siehe Kapitel Inports [▶ 48].

ID	Format	Link
Identifikationsnummer eines Elements im Ausgangsprozessabbild eines User FBs	UserFB.Id + "_o1", UserFB.Id + "_o2", ... Beispiele: g1_n1_u1_o1, u1_n1_u1_o1	Siehe Kapitel Outports [► 49].
Identifikationsnummer einer TwinSAFE-Gruppe	g1, g2, ...	Siehe Kapitel TwinSAFEGroups [► 51].
Identifikationsnummer eines Elements im Ausgangsprozessabbild einer TwinSAFE-Gruppe	TwinSAFEGroup.Id + "_o1", TwinSAFEGroup.Id + "_o2", ... Beispiel: g1_o1	Siehe Kapitel GroupOutputs [► 54].
Identifikationsnummer eines Elements im Eingangsprozessabbild einer TwinSAFE-Gruppe	TwinSAFEGroup.Id + "_i1", TwinSAFEGroup.Id + "_i2", ... Beispiel: g1_i1	Siehe Kapitel GroupInputs [► 56].
Identifikationsnummer eines Safety-Alias-Geräts	TwinSAFEGroup.Id + "_a1", TwinSAFEGroup.Id + "_a2", ... Beispiel: g1_a1	Siehe Kapitel SafetyAliasDevice [► 57].
Identifikationsnummer von DynamicPortionSafeAddress	SafetyAliasDevice.Id + "_d" Beispiel: g1_a1_d	Siehe Kapitel DynamicPortionSafeAddress [► 61]. Siehe Kapitel SafetyAliasDevice [► 57].
Identifikationsnummer eines Elements im Eingangsprozessabbild eines Safety-Alias-Geräts	SafetyAliasDevice.Id + "_i1", SafetyAliasDevice.Id + "_i2", ... Bei ConnectionType = Local: SafetyAliasDevice.Id + "_li1", SafetyAliasDevice.Id + "_li2", ... Beispiele: g1_a1_i1, g1_a1_li1	Siehe Kapitel RxPdo [► 62].
Identifikationsnummer eines Elements im Ausgangsprozessabbild eines Safety-Alias-Geräts	SafetyAliasDevice.Id + "_o1", SafetyAliasDevice.Id + "_o2", ... Bei ConnectionType = Local: SafetyAliasDevice.Id + "_lo1", SafetyAliasDevice.Id + "_lo2", ... Beispiele: g1_a1_o1, g1_a1_lo1	Siehe Kapitel TxPdo [► 62].
Identifikationsnummer von ComErrAck	SafetyAliasDevice.Id + "_c" Beispiel: g1_a1_c	Siehe Kapitel ComErrAck [► 62]. Siehe Kapitel SafetyAliasDevice [► 57].
Identifikationsnummer eines Standard-Alias-Geräts	TwinSAFEGroup.Id + "_a1", TwinSAFEGroup.Id + "_a2", ... Beispiel: g1_a1	Siehe Kapitel StandardAliasDevice [► 63].
Identifikationsnummer eines Elements im Ausgangsprozessabbild eines Standard-Alias-Geräts	StandardAliasDevice.Id + "_o1", StandardAliasDevice.Id + "_o2", ... Beispiel: g1_a1_o1	Siehe Kapitel TxPdo [► 64].
Identifikationsnummer eines Elements im Eingangsprozessabbild eines Standard-Alias-Geräts	StandardAliasDevice.Id + "_i1", StandardAliasDevice.Id + "_i2", ... Beispiel: g1_a1_i1	Siehe Kapitel RxPdo [► 65].
Identifikationsnummer eines abgeleiteten Ports	Inports.Element.Id oder Outports.Element.Id oder ParameterPorts.Element.Id Beispiele: u2_n1_u2_i1, g1_n1_u3_o3, g1_n2_u6_p1	Siehe Kapitel UserFbPortRelations [► 65].

ID	Format	Link
Identifikationsnummer der Multi Settings	ms_1, ms_2, ...	Siehe Kapitel Multi Settings [▶ 70].

7.3 FBTypes

Dieser Anhang enthält Types für die eindeutige Identifizierung der Funktionsbausteine. Siehe Kapitel [FunctionBlock](#) [▶ 35].

Name	Type
FBAAnd	34
FBOOr	35
FBDDecouple	36
FBOpmode	37
FBRs	41
FBSr	42
FBTof	43
FBTon	44
FBTon2	4402
FBEdm	39
FBEstop	32
FBMon	33
FBTwohand	38
FBMuting	40
FBConnShutdown	45
FBXor	46
FBAAdd	65
FBSub	66
FBMul	67
FBDiv	68
FBCounter	69
FBLimit	70
FBCompare	64
FBScaling	71
FBCamMonitor	72
FBLoadSensing	74
FBSpeed	73
FBViolationCnt	75
FBSLI	76
FBSLI2	7602
FBEnvelope	77
FBSLP	78
FBSBT	79
FBAdvPosMon	80

7.4 PortIds

In diesem Kapitel finden Sie die zu den Ports gehörigen IDs zur eindeutigen Identifizierung

- einer TwinSAFE-Gruppe (siehe Kapitel [TwinSAFEGroups](#) [► 51]),
- eines Funktionsbausteins (siehe Kapitel [UserFbs](#) [► 32]),
- eines Gruppen-Ports (siehe Kapitel [GroupOutputs](#) [► 54]).

Die Port-ID für einen FB-Port setzt sich zusammen aus

- (A) einem Identifizierer, der angibt, ob es sich um einen Eingang (0x01) oder Ausgang (0x02) handelt und
- (B) einer Port Kennung.

Diese werden kombiniert durch (in Pseudocode):

Gruppen Port ID = (A << 24 | B << 16)

Alias	PortId	
	Hexadezimal	Dezimal
Inports		
RunStop	0x01010000	16842752
ErrAck	0x01020000	16908288
ModuleFault	0x01030000	16973824
Outports		
FbErr	0x02010000	33619968
ComErr	0x02020000	33685504
OutErr	0x02030000	33751040
OtherErr	0x02040000	33816576
ComStartup	0x02050000	33882112
FbDeactive	0x02060000	33947648
FbRun	0x02070000	34013184
InRun	0x02080000	34078720

Die Port-ID für einen FB-Port setzt sich zusammen aus

- (A) einem Identifizierer, der angibt, ob es sich um einen Eingang (0x01) oder Ausgang (0x02) handelt,
- (B) einer Port Kennung und
- (C) der SubType-ID des Funktionsbausteins.

Diese werden kombiniert durch (in Pseudocode):

Port ID = (A << 24 | B << 16 | C)

7.4.1 FBAnd (SubType 0x22)

PortName	PortId	
	Hexadezimal	Dezimal
AndIn1	0x01080022	17301538
AndIn2	0x01090022	17367074
AndIn3	0x010a0022	17432610
AndIn4	0x010b0022	17498146
AndIn5	0x010c0022	17563682
AndIn6	0x010d0022	17629218
AndIn7	0x010e0022	17694754
AndIn8	0x010f0022	17760290

PortName	PortId	
	Hexadezimal	Dezimal
AndOut	0x02010022	33620002

7.4.2 FBOr (SubType 0x23)

PortName	PortId	
	Hexadezimal	Dezimal
OrIn1	0x01080023	17301539
OrIn2	0x01090023	17367075
OrIn3	0x010a0023	17432611
OrIn4	0x010b0023	17498147
OrIn5	0x010c0023	17563683
OrIn6	0x010d0023	17629219
OrIn7	0x010e0023	17694755
OrIn8	0x010f0023	17760291
OrOut	0x02010023	33620003

7.4.3 FBDecouple (SubType 0x24)

PortName	PortId	
	Hexadezimal	Dezimal
Decln1	0x01080024	17301540
Decln2	0x01090024	17367076
Decln3	0x010a0024	17432612
Decln4	0x010b0024	17498148
Decln5	0x010c0024	17563684
Decln6	0x010d0024	17629220
Decln7	0x010e0024	17694756
Decln8	0x010f0024	17760292
DecOut1	0x02080024	34078756
DecOut2	0x02090024	34144292
DecOut3	0x020a0024	34209828
DecOut4	0x020b0024	34275364
DecOut5	0x020c0024	34340900
DecOut6	0x020d0024	34406436
DecOut7	0x020e0024	34471972
DecOut8	0x020f0024	34537508

7.4.4 FBOpmode (SubType 0x25)

PortName	PortId	
	Hexadezimal	Dezimal
Restart	0x01000025	16777253
OpIn1	0x01080025	17301541
OpIn2	0x01090025	17367077
OpIn3	0x010a0025	17432613
OpIn4	0x010b0025	17498149
OpIn5	0x010c0025	17563685
OpIn6	0x010d0025	17629221

PortName	PortId	
	Hexadezimal	Dezimal
OpIn7	0x010e0025	17694757
OpIn8	0x010f0025	17760293
Error	0x02000025	33554469
OpOut1	0x02080025	34078757
OpOut2	0x02090025	34144293
OpOut3	0x020a0025	34209829
OpOut4	0x020b0025	34275365
OpOut5	0x020c0025	34340901
OpOut6	0x020d0025	34406437
OpOut7	0x020e0025	34471973
OpOut8	0x020f0025	34537509

7.4.5 FBRs (SubType 0x29)

PortName	PortId	
	Hexadezimal	Dezimal
Reset	0x01080029	17301545
Set	0x01090029	17367081
RsOut	0x02000029	33554473

7.4.6 FBSr (SubType 0x2a)

PortName	PortId	
	Hexadezimal	Dezimal
Set	0x0109002a	17367082
Reset	0x0108002a	17301546
SrOut	0x0200002a	33554474

7.4.7 FBTof (SubType 0x2b)

PortName	PortId	
	Hexadezimal	Dezimal
ToIn	0x0108002b	17301547
ToOut	0x0201002b	33620011

7.4.8 FBTon (SubType 0x2c)

PortName	PortId	
	Hexadezimal	Dezimal
TonIn	0x0108002c	17301548
TonOut	0x0201002c	33620012

7.4.9 FBTon2 (SubType 0x1132)

PortName	PortId	
	Hexadezimal	Dezimal
TonIn	0x01081132	17305906
Enable	0x01091132	17371442

PortName	PortId	
	Hexadezimal	Dezimal
TonOut	0x02011132	33624370

7.4.10 FBEdm (SubType 0x27)

PortName	PortId	
	Hexadezimal	Dezimal
Mon1	0x01080027	17301543
Mon2	0x01090027	17367079
Error	0x02000027	33554471

7.4.11 FBEstop (SubType 0x20)

PortName	PortId	
	Hexadezimal	Dezimal
Restart	0x01000020	16777248
EDM1	0x01010020	16842784
EDM2	0x01020020	16908320
EStopIn1	0x01080020	17301536
EStopIn2	0x01090020	17367072
EStopIn3	0x010a0020	17432608
EStopIn4	0x010b0020	17498144
EStopIn5	0x010c0020	17563680
EStopIn6	0x010d0020	17629216
EStopIn7	0x010e0020	17694752
EStopIn8	0x010f0020	17760288
Error	0x02000020	33554464
EStopOut	0x02010020	33620000
EStopDelOut	0x02020020	33685536

7.4.12 FBMon (SubType 0x21)

PortName	PortId	
	Hexadezimal	Dezimal
Restart	0x01000021	16777249
EDM1	0x01010021	16842785
EDM2	0x01020021	16908321
MonIn1	0x01080021	17301537
MonIn2	0x01090021	17367073
MonIn3	0x010a0021	17432609
MonIn4	0x010b0021	17498145
Secure1	0x010c0021	17563681
Secure2	0x010d0021	17629217
Error	0x02000021	33554465
MonOut	0x02010021	33620001
MonDelOut	0x02020021	33685537

7.4.13 FBTwohand (SubType 0x26)

PortName	PortId	
	Hexadezimal	Dezimal
TwoHand1	0x01080026	17301542
TwoHand2	0x01090026	17367078
TwoHand3	0x010a0026	17432614
TwoHand4	0x010b0026	17498150
Error	0x02000026	33554470
TwoHandOut	0x02010026	33620006

7.4.14 FBMuting (SubType 0x28)

PortName	PortId	
	Hexadezimal	Dezimal
Enable	0x01000028	16777256
EDM1	0x01010028	16842792
EDM2	0x01020028	16908328
Muting1	0x01080028	17301544
Muting2	0x01090028	17367080
OSSDIn1	0x010a0028	17432616
OSSDIn2	0x010b0028	17498152
Muting3	0x010c0028	17563688
Muting4	0x010d0028	17629224
Error	0x02000028	33554472
MutingActive	0x02010028	33620008
MuteOut	0x02020028	33685544
MuteDelOut	0x02030028	33751080

7.4.15 FBConnShutdown (SubType 0x2d)

PortName	PortId	
	Hexadezimal	Dezimal
Deactivate1	0x0108002d	17301549
Deactivate2	0x0109002d	17367085
Error	0x0200002d	33554477
Deactivated	0x0201002d	33620013

7.4.16 FBXor (SubType 0x2e)

PortName	PortId	
	Hexadezimal	Dezimal
Xor1In1	0x0100002e	16777262
Xor2In1	0x0108002e	17301550
Xor1In2	0x0101002e	16842798
Xor2In2	0x0109002e	17367086
Xor1In3	0x0102002e	16908334
Xor2In3	0x010a002e	17432622
Xor1In4	0x0103002e	16973870
Xor2In4	0x010b002e	17498158

PortName	PortId	
	Hexadezimal	Dezimal
Xor1In5	0x0104002e	17039406
Xor2In5	0x010c002e	17563694
Xor1In6	0x0105002e	17104942
Xor2In6	0x010d002e	17629230
Xor1In7	0x0106002e	17170478
Xor2In7	0x010e002e	17694766
Xor1In8	0x0107002e	17236014
Xor2In8	0x010f002e	17760302
XorOut1	0x0208002e	34078766
XorOut2	0x0209002e	34144302
XorOut3	0x020a002e	34209838
XorOut4	0x020b002e	34275374
XorOut5	0x020c002e	34340910
XorOut6	0x020d002e	34406446
XorOut7	0x020e002e	34471982
XorOut8	0x020f002e	34537518

7.4.17 FBAdd (SubType 0x41)

PortName	PortId	
	Hexadezimal	Dezimal
AnalogIn1	0x01100041	17825857
AnalogIn2	0x01110041	17891393
Error	0x02000041	33554497
AnalogOut	0x02100041	34603073

7.4.18 FBSub (SubType 0x42)

PortName	PortId	
	Hexadezimal	Dezimal
AnalogIn1	0x01100042	17825858
AnalogIn2	0x01110042	17891394
Error	0x02000042	33554498
AnalogOut	0x02100042	34603074

7.4.19 FBMul (SubType 0x43)

PortName	PortId	
	Hexadezimal	Dezimal
AnalogIn1	0x01100043	17825859
AnalogIn2	0x01110043	17891395
Error	0x02000043	33554499
AnalogOut	0x02100043	34603075

7.4.20 FBDiv (SubType 0x44)

PortName	PortId	
	Hexadezimal	Dezimal
AnalogIn1	0x01100044	17825860
AnalogIn2	0x01110044	17891396
Error	0x02000044	33554500
AnalogOut	0x02100044	34603076

7.4.21 FBCounter (SubType 0x45)

PortName	PortId	
	Hexadezimal	Dezimal
Reset	0x01000045	16777285
CountUp	0x01010045	16842821
CountDown	0x01020045	16908357
Error	0x02000045	33554501
CounterOut	0x02010045	33620037
CounterZero	0x02020045	33685573
ActValue	0x02100045	34603077

7.4.22 FBLimit (SubType 0x46)

PortName	PortId	
	Hexadezimal	Dezimal
AnalogIn	0x01100046	17825862
MinValue	0x01110046	17891398
MaxValue	0x01120046	17956934
Error	0x02000046	33554502
InLimit	0x02010046	33620038
BelowMin	0x02020046	33685574
AboveMax	0x02030046	33751110

7.4.23 FBCompare (SubType 0x40)

PortName	PortId	
	Hexadezimal	Dezimal
CompIn1	0x01100040	17825856
CompIn2	0x01110040	17891392
CompIn3	0x01120040	17956928
CompIn4	0x01130040	18022464
CompIn5	0x01140040	18088000
Error	0x02000040	33554496
CompOut	0x02100040	34603072

7.4.24 FBScale (SubType 0x47)

PortName	PortId	
	Hexadezimal	Dezimal
AnalogIn	0x01100047	17825863

PortName	PortId	
	Hexadezimal	Dezimal
Error	0x02000047	33554503
AnalogOut	0x02100047	34603079
StuckAtError	0x02010047	33620039

7.4.25 FBCamMonitor (SubType 0x48)

PortName	PortId	
	Hexadezimal	Dezimal
Automatic	0x01000048	16777288
BackwardsMove	0x01010048	16842824
Reset	0x01020048	16908360
PressStarted	0x01030048	16973896
OverrunCAM	0x01040048	17039432
UpwardCAM	0x01050048	17104968
SettingMode	0x01060048	17170504
Position	0x01100048	17825864
TDC1-Lower Limit	0x01110048	17891400
TDC1-Upper Limit	0x01120048	17956936
TDC2-Lower Limit	0x01130048	18022472
TDC2-Upper Limit	0x01140048	18088008
BDC-Lower Limit	0x01150048	18153544
BDC-Upper Limit	0x01160048	18219080
OverrunMax	0x01170048	18284616
Error	0x02000048	02000048
CAMMonOk	0x02010048	33620040
UpwardsMove	0x02020048	33685576
TDC	0x02030048	33751112
BDC	0x02040048	33816648
OverrunTDC	0x02100048	34603080
OverrunStop	0x02110048	34668616

7.4.26 FBLoadSensing (SubType 0x4a)

PortName	PortId	
	Hexadezimal	Dezimal
Inactive	0x0106004a	17170506
AnalogInX	0x0110004a	17825866
AnalogInY	0x0111004a	17891402
Valid	0x0201004a	33620042
Warning	0x0202004a	33685578

7.4.27 FBSpeed (SubType 0x49)

PortName	PortId	
	Hexadezimal	Dezimal
Position	0x01100049	17825865
Error	0x02000049	33554505
Speed	0x02100049	34603081

7.4.28 FBViolationCnt (SubType 0x4b)

PortName	PortId	
	Hexadezimal	Dezimal
Input	0x0100004b	16777291
Enable	0x0101004b	16842827
InputOK	0x0201004b	33620043
ActViolationCnt	0x0210004b	34603083

7.4.29 FBSLI (SubType 0x4c)

PortName	PortId	
	Hexadezimal	Dezimal
SLI	0x0100004c	16777292
Position	0x0110004c	17825868
Error	0x0200004c	33554508
SLIActive	0x0201004c	33620044
PositionDiff	0x0210004c	34603084

7.4.30 FBSLI2 (SubType 0x1db2)

PortName	PortId	
	Hexadezimal	Dezimal
SLI	0x01001db2	16784818
Position	0x01101db2	17833394
Error	0x02001db2	33562034
SLIActive	0x02011db2	33627570
BelowMin	0x02021db2	33693106
AboveMax	0x02031db2	33758642
PositionDiff	0x02101db2	34610610

7.4.31 FBEnvelope (SubType 0x4d)

PortName	PortId	
	Hexadezimal	Dezimal
SafeFunction	0x0100004d	16777293
InValue	0x0110004d	17825869
Error	0x0200004d	33554509
SafeFunctionOut	0x0201004d	33620045

7.4.32 FBSLP (SubType 0x4e)

PortName	PortId	
	Hexadezimal	Dezimal
SafePosValid	0x0100004e	16777294
SingleturnPos	0x0110004e	17825870
MultiturnPos	0x0111004e	17891406
SingleturnPosLL	0x0112004e	17956942
MultiturnPosLL	0x0113004e	18022478
SingleturnPosUL	0x0114004e	18088014

PortName	PortId	
	Hexadezimal	Dezimal
MultiturnPosUL	0x0115004e	18153550
Error	0x0200004e	33554510
PositionInLimit	0x0202004e	33685582
BelowMin	0x0203004e	33751118
AboveMax	0x0204004e	33816654

7.4.33 FBSBT (SubType 0x4f)

PortName	PortId	
	Hexadezimal	Dezimal
SBT	0x0100004f	16777295
SBCIn	0x0101004f	16842831
SingleturnPos	0x0110004f	17825871
ActCurrentIq	0x0111004f	17891407
MinCurrentIq	0x0112004f	17956943
TestError	0x0201004f	33620047
BrakeValid	0x0202004f	33685583
RemainingTime	0x0210004f	34603087

7.4.34 FBAdvPosMon (SubType 0x50)

PortName	PortId	
	Hexadezimal	Dezimal
Restart	0x01000050	16777296
Enable	0x01080050	17301584
Error	0x02000050	33554512
Active	0x02010050	33620048
Wait	0x02020050	33685584
OverRange	0x02030050	33751120

7.5 ConnectionType

Der ConnectionType enthält Informationen über das sichere Kommunikationsprotokoll des SafetyAliasDevices und die Rolle des Kommunikationspartners in diesem Projekt.

ConnectionType	Kommunikationsprotokoll	Rolle des Kommunikationspartners
FSoEMaster	FSoE Siehe Dokument [5] in Kapitel Referenzen [▶ 7].	Master
FSoESlave	FSoE	Slave
TwinSafeScCrc1master	TwinSAFE SC mit dem Generatorpolynom 0x7B0F Siehe Dokument [5] in Kapitel Referenzen [▶ 7].	Master
TwinSafeScCrc2master	TwinSAFE SC mit dem Generatorpolynom 0x571F	Master
TwinSafeScCrc3master	TwinSAFE SC mit dem Generatorpolynom 0x1F95	Master
TwinSafeScCrc4master	TwinSAFE SC mit dem Generatorpolynom 0x53F1	Master
TwinSafeScCrc5master	TwinSAFE SC mit dem Generatorpolynom 0xF1D5	Master
TwinSafeScCrc6master	TwinSAFE SC mit dem Generatorpolynom 0x663B	Master
TwinSafeScCrc7master	TwinSAFE SC mit dem Generatorpolynom 0xB8CD	Master
TwinSafeScCrc8master	TwinSAFE SC mit dem Generatorpolynom 0xE1BD	Master
TwinSafeScCustomCrcMaster	TwinSAFE SC mit einem selbstgewählten Generatorpolynom Siehe Element „CustomCRC“ in Kapitel SafetyAliasDevice [▶ 57].	Master
PROFIsafeMaster	PROFIsafe Siehe Dokument [5] in Kapitel Referenzen [▶ 7].	Master
PROFIsafeSlave	PROFIsafe	Slave
Local	Mit dieser Einstellung wird mitgeteilt, dass dieses SafetyAliasDevice teil der TargetSystemConfiguration ist. Siehe Kapitel TargetSystemConfiguration [▶ 25].	

7.5.1 ConnectionType in TE9000

Den ConnectionType finden Sie im TwinCAT Safety Editor TE9000 über den „Connection“ und den „Linking“-Reiter eines SafetyAliasDevices.

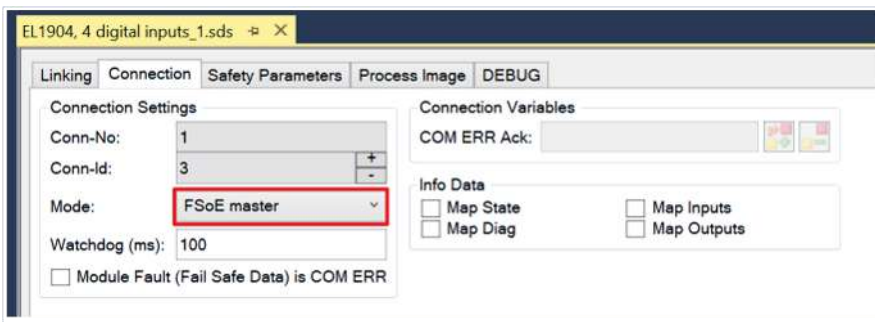


Abb. 47: ConnectionType EL1904

In der Abbildung sehen Sie, dass das SafetyAliasDevice EL1904 als „FSoE master“ definiert ist.

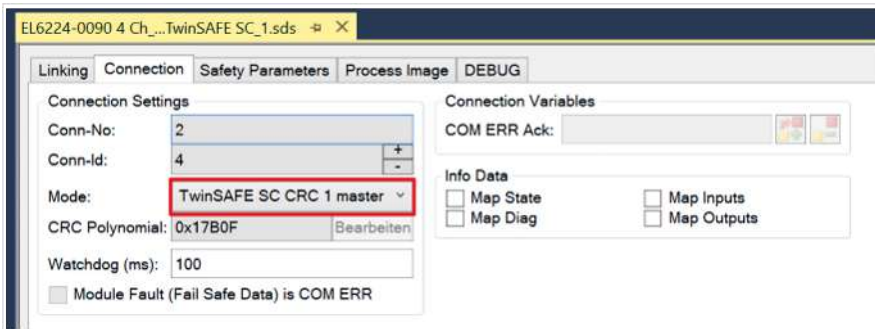


Abb. 48: ConnectionType TwinSAFE SC

Für TwinSAFE SC ist als Modus „TwinSAFE SC CRC 1 master“ ausgewählt.

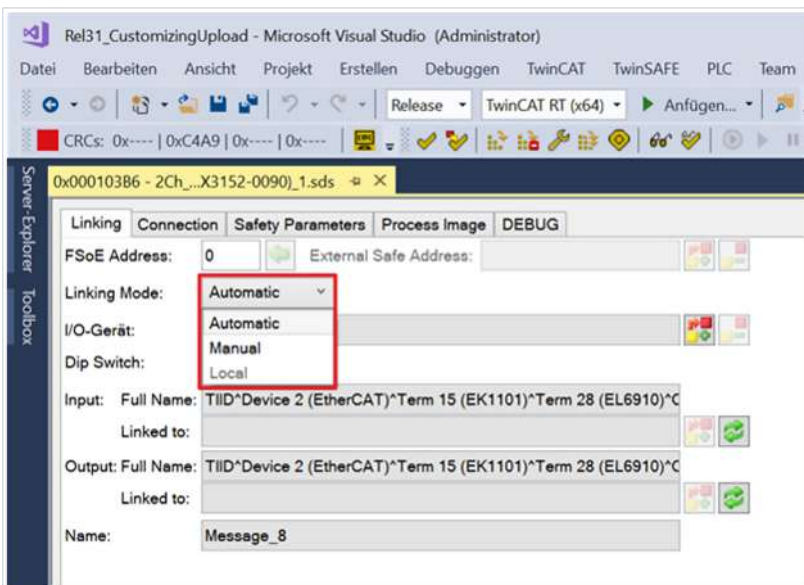


Abb. 49: Linking Mode

Über den „Linking“-Reiter finden Sie die Einstellungen zum Linking Mode.

7.6 LogicType

In TwinCAT 3 haben Sie bei den CustomFSoEConnections die Möglichkeit den Typ einzustellen. Der Typ entspricht dem LogicType. Für alle anderen SafetyAliasDevices existiert diese Möglichkeit nicht.

Mit dieser Einstellung wird der Kommunikationspartner bekannt gegeben. Sie können folgende Werte einstellen:

- None

- KL6904
- EL6900EL6930 (falls der Kommunikationspartner eine EL6900 oder EL6930 ist)
- EL6910EJ6910 (falls der Kommunikationspartner eine EL6910 oder EJ6910 ist)
- Alle in TargetSystemConfiguration/SubType genannten Zielsysteme ausgenommen EL6900, EL6930, EL6910 und EJ6910

Abhängig von der Auswahl werden zum einen zusätzliche Backup & Restore-Funktionen freigeschaltet oder deaktiviert. Weitere Informationen entnehmen Sie dem Kapitel [BackupRestore](#) [► 28].

Falls beim SafetyAliasDevice als ConnectionType „FsoEMaster“ eingestellt und der Kommunikationspartner ein Gerät aus der möglichen Geräteliste ist, dann wird automatisch die SafeParameters-Konfiguration für diese Kommunikation erstellt. Für weitere Informationen über ConnectionType und SafetyParameters sehen Sie in die Kapitel [SafetyAliasDevice](#) [► 57] und [SafetyParameters](#) [► 61].

Für die Zielsysteme EL6900 und EL6930 wird aus Kompatibilitätsgründen zu TwinCAT 2 der LogicType mit in die Projekt CRC eingerechnet.

7.6.1 LogicType in TE9000

Das SafetyAliasDevice „Custom FSoE Connection“ fügen Sie über den „Add“-Dialog hinzu:

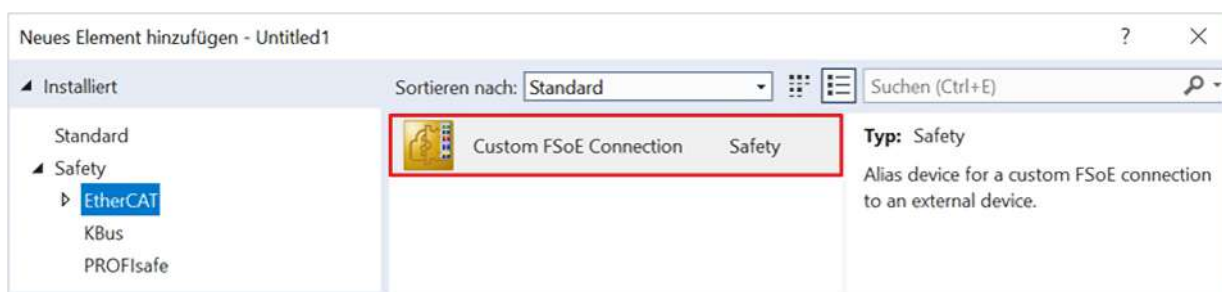


Abb. 50: Element hinzufügen

Die Einstellung zum LogicType finden Sie über den „Connection“-Reiter des SafetyAliasDevices:

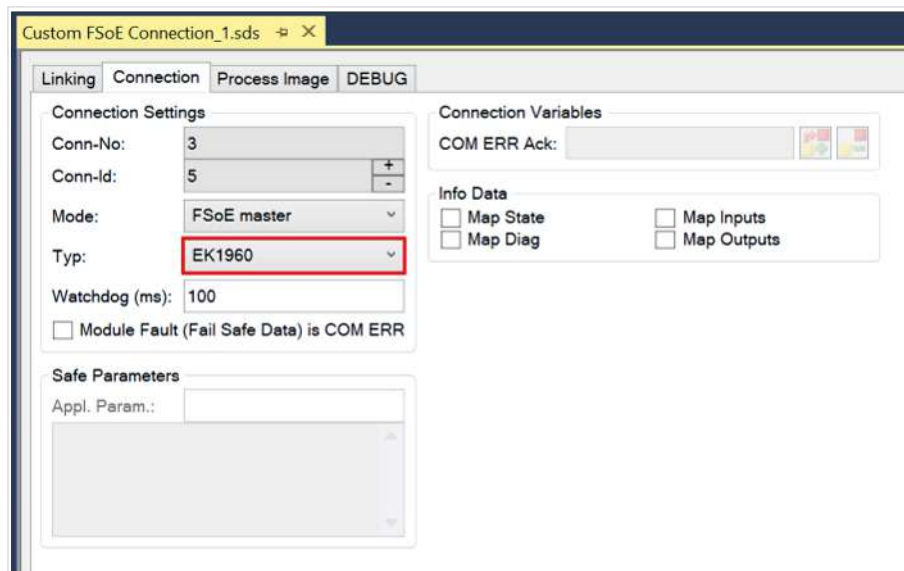


Abb. 51: LogicType

8 Kompatibles Safety-Projekt generieren

Dieses Kapitel beschreibt beispielhaft die Erstellung eines TwinCAT-3-kompatiblen Safety-Projekts außerhalb der Entwicklungsumgebung TwinCAT 3.

Dafür können Sie die Informationen aus den vorherigen Kapiteln nutzen. In der Regel benötigen Sie nicht alle Elemente aus der XSD-Datei.

Information

Das Element `TargetSystemProcessImage` müssen Sie nicht erstellen. Der TwinSAFE Editor erstellt das Prozessabbild für das Basisprojekt automatisch nach dem Import der XML-Datei.

Aufgrund der Komplexität der Bereiche `UserFunctionBlock`, `UserFbPortRelations` und `MultiSettings`, wird seitens Beckhoff eine automatische Erstellung eines Safety-Projekts mit diesen Elementen nicht empfohlen und ist gegebenenfalls nicht möglich.

8.1 Beispiel EStop-Funktion

Dieses Kapitel beschreibt die Erstellung Safety-Projekts anhand des Beispiels „2.3 ESTOP-Funktion Variante 3 (Kategorie 4, PL e)“ aus Dokument [5] in Kapitel [Referenzen](#) [7].

Erstellen Sie zunächst eine XML-Datei mit dem Namen „`SafetyProject_Estop_Function.xml`“. Das weitere Vorgehen richtet sich anhand der Struktur der XSD-Datei. Dies entnehmen Sie diesem und den nachfolgenden Kapiteln.

TwinCATExport

Für dieses Element benötigen Sie das [TwinCATExport Attribut](#) [21] „Version“. Die Version ist abhängig von der verwendeten TwinCAT-Version und der TE9000-Version. Sehen Sie für weitere Informationen in die Kapitel [Systemvoraussetzungen](#) [15] und [Systemgrenzen](#) [15].

In diesem Beispiel legen Sie die Version auf 0.31 fest.

```
<TwinCATExport Version="0.31">
</TwinCATExport>
```

8.1.1 BaseConfiguration

Die Basiskonfiguration beinhaltet die vollständige Beschreibung des Safety-Projekts.

GeneralInformation

Hier vergeben Sie den Projekt-Namen. Alle weiteren Angaben sind optional.

Da Sie das Projekt neu erstellen, ist die Projekt-CRC noch unbekannt und kann daher noch nicht angegeben werden.

```
<GeneralInformation>
  <ProjectName>EstopFunction</ProjectName>
  <Author>Person</Author>
  <InternalProjectName>EstopFunction</InternalProjectName>
</GeneralInformation>
```

References

Für dieses Beispiel wird keine Referenz verwendet.

TargetSystemConfiguration

In diesem Beispiel wird als Zielsystem die TwinSAFE Logic EL6900 verwendet. Definieren Sie dafür die Attribute „`Id`“, „`Type`“ und „`SubType`“ sowie die Elemente „`SafeAddress`“ und „`VersionNumber`“.

Alle weiteren Elemente und Attribute sind optional. Die Id, den Type und den SubType entnehmen Sie den vorhergehenden Kapiteln [ID-Regeln \[► 75\]](#) und [TargetSystemConfiguration \[► 25\]](#). Für SafeAddress und VersionNumber wählen Sie zunächst beliebig gültige Zahlenwerte.

```
<TargetSystemConfiguration Id="t1" Type="Hardware" SubType="EL6900">
  <SafeAddress>1</SafeAddress>
  <VersionNumber>1</VersionNumber>
</TargetSystemConfiguration>
```

ApplicationConfiguration

Die die Konfiguration der Applikation umfangreicher ist, wird dieses Element im nächsten Kapitel näher beschrieben.

8.1.1.1 ApplicationConfiguration

Hier definieren Sie die wesentliche Logik.

```
<ApplicationConfiguration>
</ApplicationConfiguration>
```

UserFbs

Für dieses Beispiel werden keine User FBs verwendet.

TwinSAFEGroups

Jedes Safety-Projekt benötigt immer mindestens eine TwinSAFE-Gruppe. Für dieses Beispiel mit zwei FBs ist eine TwinSAFE-Gruppe ausreichend.

Die die Konfiguration der TwinSAFE-Gruppe umfangreicher ist, wird dieses Element im nächsten Kapitel näher beschrieben.

8.1.1.1.1 TwinSAFEGroup

Den Wert für das Attribut „Id“ entnehmen Sie dem Kapitel [ID-Regeln \[► 75\]](#). Die OrderId beginnt für gewöhnlich bei „0“ und erhöht sich für jede TwinSAFE-Gruppe um den Wert „1“.

```
<TwinSAFEGroups>
  <TwinSAFEGroup Id="g1" OrderId="0">
  </TwinSAFEGroup>
</TwinSAFEGroups>
```

8.1.1.1.1.1 GroupInputs

In diesem Beispiel wird für das Zielsystem EL6900 die Verwendung des Gruppeneingangsports ErrAck und des Eingangs RunStop vorgegeben.

Aus den Kapitel [ID-Regeln \[► 75\]](#) und [PortIds \[► 78\]](#) entnehmen Sie die Werte für das Attribut „Id“ sowie die Elemente „PortId“ und „Alias“. Das Element „Alias“ ist optional.

```
<GroupInputs>
  <Element Id="g1_i1">
    <PortId>16842752</PortId>
    <Alias>RunStop</Alias>
  </Element>
  <Element Id="g1_i2">
    <PortId>16908288</PortId>
    <Alias>ErrAck</Alias>
  </Element>
</GroupInputs>
```

8.1.1.1.1.2 AliasDevices

In diesem Beispiel verwenden Sie die TwinSAFE-I/O-Geräte EL1904 und EL2904. Diese Geräte werden im TwinSAFE Editor als SafetyAliasDevices beschrieben. Alle FSoE-, TwinSAFE-SC- oder PROFIsafe-Geräte interpretiert TwinCAT 3 als SafetyAliasDevices.

Außerdem benötigen Sie 4 StandardAliasDevices.

- 1 x StandardAliasDevice für den Gruppeneingangsport ErrAck
- 1 x StandardAliasDevice für den Gruppeneingang RunStop
- 2 x StandardAliasDevice für die Verknüpfung mit dem 1. und 2. Eingang einer EL1004

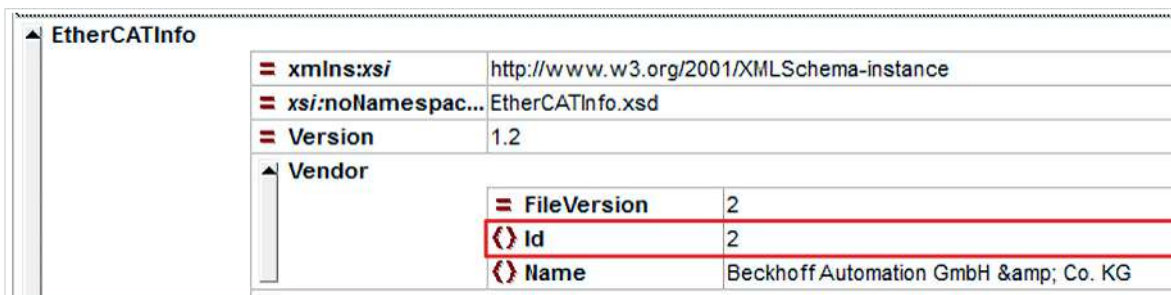
Die Werte für die Id-Attribute setzen Sie anhand der [ID-Regeln \[▶ 75\]](#).

```
<AliasDevices>
  <SafetyAliasDevice Id="g1_a1">
  </SafetyAliasDevice>
  <SafetyAliasDevice Id="g1_a2">
  </SafetyAliasDevice>
  <StandardAliasDevice Id="g1_a3">
  </StandardAliasDevice>
  <StandardAliasDevice Id="g1_a4">
  </StandardAliasDevice>
  <StandardAliasDevice Id="g1_a5">
  </StandardAliasDevice>
  <StandardAliasDevice Id="g1_a6">
  </StandardAliasDevice>
</AliasDevices>
```

8.1.1.1.2.1 SafetyAliasDevice Id=„g1_a1“

Dieses SafetyAliasDevice wird durch die EL1904 beschrieben. Da die EL1904 ein EtherCAT-Gerät ist, können Sie die VendorId aus der ESI-Datei entnehmen. Sehen Sie dazu in Dokument [2] unter [Referenzen \[▶ 7\]](#).

VendorId (EtherCATInfo.Vendor.Id):



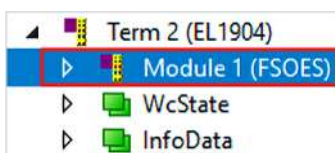
Für Beckhoff-EtherCAT-Geräte ist die VendorId immer 2.

Type und SubType

Den Type und den SubType lesen Sie aus dem TwinCAT System Manager aus, falls diese noch nicht bekannt sind. Sie können zwischen 2 Vorgehensweisen wählen.

Erste Vorgehensweise: Modul-Informationen exportieren:

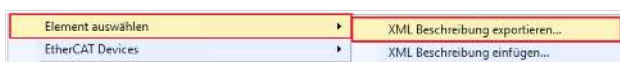
Gehen Sie dazu wie folgt vor:



1. In TwinCAT 3 das Safety-Modul der EL1904 auswählen



2. In der Menüleiste „TwinCAT“ auswählen



3. Über die Schaltfläche „Element auswählen“ „XML Beschreibung exportieren...“ anklicken

4. Aus der XML-Datei den Type aus dem ItemType und den SubType aus dem ItemSubType herauslesen

```
<TreeItem>
  <ItemName>Module 1 (FSOES)</ItemName>
  <PathName>TIID^Device 2 (EtherCAT)^Term 2 (EL1904)^Module 1 (SFOES)</PathName>
  <ItemType>60</ItemType>
  <ItemId>50921473</ItemId>
  <ObjectId>#x03090001</ObjectId>
  <ItemSubType>190</ItemSubType>
  <ChildCount>0</ChildCount>
  <Disabled>false</Disabled>
  <TreeImageId>118</TreeImageId>
```

Zweite Vorgehensweise: Safety-Projekt exportieren

Als weitere Möglichkeit können Sie Type und SubType herausfinden, indem Sie ein TwinCAT-3-Safety-Projekt mit dem entsprechenden SafetyAliasDevice erstellen und als TwinCAT 3 Xml Format exportieren.

Aus der exportierten XML-Datei entnehmen Sie Type und SubType.

Elemente definieren

Hier definieren Sie die folgenden Elemente:

- AliasName
 - Dateinamen des AliasDevice EL1904
- ConnectionId
 - Identifizierung der EL1904 als Kommunikationsteilnehmer
- SafeAddress
 - In diesem Beispiel wird für das SafetyAliasDevice die Adresse „2“ gewählt, da die Adresse „1“ bereits für die EL6900 vergeben ist.
- Watchdog
 - In diesem Beispiel wird der typische Wert „100 [ms]“ gewählt.

```
<SafetyAliasDevice Id="g1_al">
  <VendorId>2</VendorId>
  <Type>60</Type>
  <SubType>190</SubType>
  <AliasName>EL1904</AliasName>
  <ConnectionId>1</ConnectionId>
  <SafeAddress>2</SafeAddress>
  <Watchdog>100</Watchdog>
</SafetyAliasDevice >
```

8.1.1.1.2.1.1 SafetyParameter

Der Not-Halt-Taster ist mit dem 3. und 4. Eingang der EL1904 verdrahtet. Außerdem muss die Testung der Eingänge aktiviert werden, um den Diagnosedeckungsgrad zu erhöhen. Diese Einstellung nehmen Sie in den SafetyParametern vor.

Öffnen Sie dazu die ESI-Datei (Beckhoff Elx9xx.xml), in der das Modul der EL1904 definiert ist, und navigieren Sie bis zu dem Element „SafetyParaMapping“.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<EtherCATInfo xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="EtherCATInfo.xsd" Version="1.2">
  <Vendor FileVersion="2">
    <Descriptions>
      <Groups>
      <Devices>
      <Modules>
        <Module Crc32="#xc1f11b2b">
          <Type ModuleIdent="190" ModuleClass="190">FSOES</Type>
          <Name>EL1904 FSoE Message</Name>
          <RxPdo Fixed="true" Sm="2">
          <TxPdo Fixed="true" Sm="3">
          <SafetyParaMapping Fixed="true">
          <Profile>
        </Module>
```

Listen Sie die Parameter in der Reihenfolge auf, wie sie in der ESI-Datei aufgelistet werden. Ausgenommen davon sind die ersten drei Standard-Parameter und Parameter ohne Namen, die Sie nicht weiter beachten müssen.

Index	SubIndex	BitLen	Name	DataType
1 #x9001	7	16	LengthOfCommParameter	UINT
2 #x9001	4	16	WatchdogTime	UINT
3 #x9001	8	16	LengthOfAppParameter	UINT
4 #x8000	1	8	Operating Mode	USINT
5 #x8001	1	1	Sensor test Channel 1 active	BOOL
6 #x8001	2	1	Sensor test Channel 2 active	BOOL
7 #x8001	3	1	Sensor test Channel 3 active	BOOL
8 #x8001	4	1	Sensor test Channel 4 active	BOOL
9 #x0000	0	4		
10 #x8002	1	2	Logic of Channel 1 and 2	BIT2
11 #x8002	3	2	Logic of Channel 3 and 4	BIT2
12 #x0000	0	12		
13 #x10E0	1	16	Store Code	UINT
14 #x10E0	2	16	Project CRC	UINT

Falls Sie sich an die Reihenfolge der ESI-Datei halten, können Sie das Attribut „OrderId“ weglassen. Zur besseren Zuordnung der Parameter zu den Werten, werden in diesem Beispiel die Parameter-Namen übernommen. Diese werden beim Import nicht ausgewertet.

```
<SafetyParameters>
  <SafetyParameter Name="Operating Mode">0</SafetyParameter>
  <SafetyParameter Name="Sensor test Channel 1 active">0</SafetyParameter>
  <SafetyParameter Name="Sensor test Channel 2 active">0</SafetyParameter>
  <SafetyParameter Name="Sensor test Channel 3 active">1</SafetyParameter>
  <SafetyParameter Name="Sensor test Channel 4 active">1</SafetyParameter>
  <SafetyParameter Name="Logic of Channel 1 and 2">0</SafetyParameter>
  <SafetyParameter Name="Logic of Channel 3 and 4">0</SafetyParameter>
  <SafetyParameter Name="Store Code">0</SafetyParameter>
  <SafetyParameter Name="Project CRC">0</SafetyParameter>
</SafetyParameters>
```

8.1.1.1.2.1.2 RxPdo

Die TwinSAFE-Klemme EL1904 hat 4 sichere digitale Eingänge. Diese müssen Sie in den Bereich „RxPdo“ eingearbeiten.

Wenden Sie sich zunächst dem 1. Eingang zu. Den Wert für das Id-Attribut entnehmen Sie dem Kapitel [ID-Regeln](#) [► 75]. Außerdem müssen Sie zwingend die Elemente „Name“, „Type“ und „BitOffset“ definieren, sowie den „Index“ und „SubIndex“, da es sich bei der EL1904 um ein ESI-Gerät handelt.

Öffnen Sie die ESI-Datei (Beckhoff ELx9xx.xml) und navigieren Sie bis zu dem Element „TxPdo“.

Index	SubIndex	BitLen	Name	DataType
1 #x6000	1	8	FSoE Slave CMD	USINT
2 #x6001	1	1	InputChannel1	BOOL
3 #x6001	2	1	InputChannel2	BOOL
4 #x6001	3	1	InputChannel3	BOOL
5 #x6001	4	1	InputChannel4	BOOL
6 #x0000	0	4		
7 #x6000	3	16	FSoE Slave CRC_0	UINT
8 #x6000	2	16	FSoE Slave ConnID	UINT

Hier finden Sie den Namen, sowie den Index und Subindex und Sie können aus der BitLen den BitOffset berechnen. Den Type wählen sie nach der Betrachtung der BitLen des DataTypes aus der ESI-Datei und der Liste an möglichen Types in der XSD-Datei aus. Für die möglichen Types sehen Sie in das Kapitel [PortIds](#) [► 78].

Für den ersten Eingang sieht die XML-Beschreibung dann wie folgt aus:

```
<Element Id="g1_a1_i1">
  <Name>InputChannel1</Name>
  <Type>BIT</Type>
  <BitOffset>8</BitOffset>
  <Index>24577</Index>
  <SubIndex>1</SubIndex>
</Element>
```

Definieren Sie nach demselben Prinzip die weiteren Eingänge der EL1904.

```
<Element Id="g1_a1_i2">
  <Name>InputChannel2</Name>
  <Type>BIT</Type>
  <BitOffset>9</BitOffset>
  <Index>24577</Index>
  <SubIndex>2</SubIndex>
</Element>
<Element Id="g1_a1_i3">
  <Name>InputChannel3</Name>
  <Type>BIT</Type>
  <BitOffset>10</BitOffset>
  <Index>24577</Index>
  <SubIndex>3</SubIndex>
</Element>
<Element Id="g1_a1_i4">
  <Name>InputChannel4</Name>
  <Type>BIT</Type>
  <BitOffset>11</BitOffset>
  <Index>24577</Index>
  <SubIndex>4</SubIndex>
</Element>
```

Abschließend erfolgt die Angabe der BitSize für das Element „RxPdo“. Summieren Sie dafür alle BitLen auf.

```
<RxPdo BitSize="48">
  <Element Id="g1_a1_i1">
  <Element Id="g1_a1_i2">
  <Element Id="g1_a1_i3">
  <Element Id="g1_a1_i4">
</RxPdo>
```

8.1.1.1.2.2 *SafetyAliasDevice Id=„g1_a2“*

Dieses SafetyAliasDevice wird durch die EL2904 beschrieben. Da die EL2904 ein EtherCAT-Gerät ist, gehen Sie analog zur EL1904 (g1_a1) vor.

In diesem Beispiel müssen Sie die Strommessung und die Testung der Ausgänge aktivieren. Diese Einstellungen nehmen Sie in dem Element „SafetyParameters“ vor.

Für dieses StandardAliasDevice sieht die Konfiguration wie folgt aus:

```
<SafetyAliasDevice Id="g1_a2">
  <VendorId>2</VendorId>
  <Type>60</Type>
  <SubType>290</SubType>
  <AliasName>EL2904</AliasName>
  <ConnectionId>2</ConnectionId>
  <SafeAddress>3</SafeAddress>
  <Watchdog>100</Watchdog>
  <SafetyParameters>
    <SafetyParameter Name="Standard outputs active">0</SafetyParameter>
    <SafetyParameter Name="Current measurement active">1</SafetyParameter>
    <SafetyParameter Name="Testing of outputs active">1</SafetyParameter>
    <SafetyParameter Name="Error acknowledge active">0</SafetyParameter>
    <SafetyParameter Name="Store Code">0</SafetyParameter>
    <SafetyParameter Name="Project CRC">0</SafetyParameter>
  </SafetyParameters>
  <TxPdo BitSize="48">
    <Element Id="g1_a2_o1"> <Name>OutputChannel1</Name>
      <Type>BIT</Type>
```

```

    <BitOffset>8</BitOffset>
    <Index>28673</Index>
    <SubIndex>1</SubIndex>
  </Element>
  <Element Id="g1_a2_o2">
    <Name>OutputChannel2</Name>
    <Type>BIT</Type>
    <BitOffset>9</BitOffset>
    <Index>28673</Index>
    <SubIndex>2</SubIndex>
  </Element>
  <Element Id="g1_a2_o3">
    <Name>OutputChannel3</Name>
    <Type>BIT</Type>
    <BitOffset>10</BitOffset>
    <Index>28673</Index>
    <SubIndex>3</SubIndex>
  </Element>
  <Element Id="g1_a2_o4">
    <Name>OutputChannel4</Name>
    <Type>BIT</Type>
    <BitOffset>11</BitOffset>
    <Index>28673</Index>
    <SubIndex>4</SubIndex>
  </Element>
</TxPdo>
</SafetyAliasDevice>

```

8.1.1.1.2.3 *StandardAliasDevice Id=„g1_a3“*

Dieses StandardAliasDevice verknüpfen Sie mit dem Gruppen-Eingang „ErrAck“. Dazu benötigen Sie den AliasName. Der AliasName repräsentiert nach dem Importieren den Dateinamen des AliasDevices.

Passen Sie außerdem das Prozessabbild an. Aktuell sind alle StandardAliasDevices so ausgelegt, dass es nur ein Pdo-Element geben kann. Da Sie in diesem Beispiel einen booleschen Eingang benötigen, definieren Sie diesen StandardAliasDevice ähnlich zu dem SafetyAliasDevice.

Für dieses StandardAliasDevice sieht die Konfiguration wie folgt aus:

```

<StandardAliasDevice Id="g1_a3">
  <AliasName>ErrAck</AliasName>
  <RxPdo BitSize="1">
    <Element Id="g1_a3_i1">
      <Name>In</Name>
      <Type>BIT</Type>
      <BitOffset>0</BitOffset>
    </Element>
  </RxPdo>
</StandardAliasDevice>

```

8.1.1.1.2.4 *StandardAliasDevice Id=„g1_a4“*

Dieses StandardAliasDevice verknüpfen Sie mit dem Gruppen-Eingang „RunStop“. Gehen Sie dabei vor wie bei dem StandardAliasDevice „g1_a3“.

```

<StandardAliasDevice Id="g1_a4">
  <AliasName>Run</AliasName>
  <RxPdo BitSize="1">
    <Element Id="g1_a4_i1">
      <Name>In</Name>
      <Type>BIT</Type>
      <BitOffset>0</BitOffset>
    </Element>
  </RxPdo>
</StandardAliasDevice>

```

8.1.1.1.2.5 *StandardAliasDevice Id=„g1_a5“*

Dieses StandardAliasDevice verknüpfen Sie mit dem 1. Eingang der EL1004. Gehen Sie dabei vor wie bei dem StandardAliasDevice „g1_a3“. Dieses Signal verknüpfen Sie mit dem Eingang „Restart“ des Funktionsbausteins „FBEstop“.

```
<StandardAliasDevice Id="g1_a5">
  <AliasName>Restart</AliasName>
  <RxPdo BitSize="1">
    <Element Id="g1_a5_i1">
      <Name>In</Name>
      <Type>BIT</Type>
      <BitOffset>0</BitOffset>
    </Element>
  </RxPdo>
</StandardAliasDevice>
```

8.1.1.1.2.6 *StandardAliasDevice Id=„g1_a6“*

Dieses `StandardAliasDevice` verknüpfen Sie mit dem 2. Eingang der EL1004. Dieser Eingang ist mit dem Rückführsignal der Schütze K1 und K2 verdrahtet. Gehen Sie dabei vor wie bei dem `StandardAliasDevice „g1_a3“`.

```
<StandardAliasDevice Id="g1_a6">
  <AliasName>Feedback</AliasName>
  <RxPdo BitSize="1">
    <Element Id="g1_a6_i1">
      <Name>In</Name>
      <Type>BIT</Type>
      <BitOffset>0</BitOffset>
    </Element>
  </RxPdo>
</StandardAliasDevice>
```

8.1.1.1.1.3 Application

In diesem Element konfigurieren Sie die FBs. Dafür definieren Sie zunächst das Netzwerk, da Sie FBs nur innerhalb von Netzwerken konfigurieren können. Da in dieser Applikation nur 2 FBs benötigt werden, ist ein Netzwerk ausreichend.

Die Konfiguration der Id entnehmen Sie dem Kapitel [ID-Regeln](#) [► 75]. Sie benötigen die zwei FBs „FBEstop“ und „FBEdm“. Die Beschreibung der FBs finden Sie der in der FB-Dokumentation. Dies ist Dokument [4] in Kapitel [Referenzen](#) [► 7].

```
<Application>
  <Network Id="g1_n1" OrderId="1">
    <FunctionBlock Id="g1_n1_f1" OrderId="1">
    </FunctionBlock>
    <FunctionBlock Id="g1_n1_f2" OrderId="2">
    </FunctionBlock>
  </Network>
</Application>
```

8.1.1.1.1.3.1 *FBEstop*

Geben Sie für einen Funktionsbaustein den Type, Instanz-Namen sowie die Eingänge und Ausgänge an. Für jeden FB müssen Sie mindestens ein Eingang und ein Ausgang konfigurieren. Den FB-Type entnehmen Sie dem Kapitel [FBTypes](#) [► 77]. Falls Sie weitere Parameter benötigen, deren Werte von den Default-Werten abweichen, geben Sie diese Parameter hier ebenfalls an. In diesem Beispiel verwenden Sie ausschließlich die Default-Werte des FBs.

```
<FunctionBlock Id="g1_n1_f1" OrderId="1">
  <Type>32</Type>
  <Name>FBEstop1</Name>
</FunctionBlock>
```

8.1.1.1.1.3.1.1 OutPorts

Der `FBEstop` hat 3 Ausgänge. Geben Sie für jeden dieser Ausgänge die Id und die PortId an. Die Id entnehmen Sie dem Kapitel [ID-Regeln](#) [► 75]. Die PortId entnehmen Sie dem Kapitel [PortIds](#) [► 78]. Für eine bessere Lesbarkeit können Sie den PortName mit angeben. Die Konfiguration der Ports nutzen Sie auch für das Element Mappings.

```
<Outports>
  <Element Id="g1_n1_f1_o1">
    <PortId>33554464</PortId>
```

```

    <PortName>Error</PortName>
  </Element>
  <Element Id="g1_n1_f1_o2">
    <PortId>33620000</PortId>
    <PortName>EStopOut</PortName>
  </Element>
  <Element Id="g1_n1_f1_o3">
    <PortId>33685536</PortId>
    <PortName>EStopDelOut</PortName>
  </Element>
</Outports>

```

8.1.1.1.3.1.2 InPorts

Der FB EStop hat 11 Eingänge. Konfigurieren Sie diese nach demselben Prinzip wie die OutPorts. Um den Diagnosedeckungsgrad zu erhöhen, stellen Sie eine Diskrepanz-Zeit für die Eingänge „EStopIn1“ und „EstopIn2“ ein. Für die InPorts, für die Sie später ein Mapping konfigurieren, setzen Sie das Element „Active“.

```

<Inports>
  <Element Id="g1_n1_f1_i1">
    <PortId>16777248</PortId>
    <PortName>Restart</PortName>
    <Active>true</Active>
  </Element>
  <Element Id="g1_n1_f1_i10">
    <PortId>16842784</PortId>
    <PortName>EDM1</PortName>
    <Active>true</Active>
  </Element>
  <Element Id="g1_n1_f1_i11">
    <PortId>16908320</PortId>
    <PortName>EDM2</PortName>
  </Element>
  <Element Id="g1_n1_f1_i2">
    <PortId>17301536</PortId>
    <PortName>EStopIn1</PortName>
    <DiscTime>200</DiscTime>
    <Active>true</Active> </Element>
  <Element Id="g1_n1_f1_i3">
    <PortId>17367072</PortId>
    <PortName>EStopIn2</PortName>
    <Active>true</Active>
  </Element>
  <Element Id="g1_n1_f1_i4">
    <PortId>17432608</PortId>
    <PortName>EStopIn3</PortName>
  </Element>
  <Element Id="g1_n1_f1_i5">
    <PortId>17498144</PortId>
    <PortName>EStopIn4</PortName>
  </Element>
  <Element Id="g1_n1_f1_i6">
    <PortId>17563680</PortId>
    <PortName>EStopIn5</PortName>
  </Element>
  <Element Id="g1_n1_f1_i7">
    <PortId>17629216</PortId>
    <PortName>EStopIn6</PortName>
  </Element>
  <Element Id="g1_n1_f1_i8">
    <PortId>17694752</PortId>
    <PortName>EStopIn7</PortName>
  </Element>
  <Element Id="g1_n1_f1_i9">
    <PortId>17760288</PortId>
    <PortName>EStopIn8</PortName>
  </Element>
</Inports>

```

8.1.1.1.3.2 FBEdm

Gehen Sie für diesen FB ähnlich vor. Hierbei verwenden Sie die Parameter „SwitchOffTime“ und „SwitchOnTime“. Sehen Sie dazu in das Kapitel [FunctionBlock \[► 35\]](#).


```
<FunctionBlock Id="g1_n1_f2" OrderId="2">
  <Type>39</Type>
  <Name>FBEEdml</Name>
  <SwitchOffTime>1500</SwitchOffTime>
  <SwitchOnTime>1000</SwitchOnTime>
  <Outports>
    <Element Id="g1_n1_f2_o1">
      <PortId>33554471</PortId>
      <PortName>Error</PortName>
    </Element>
  </Outports>
  <Inports>
    <Element Id="g1_n1_f2_i1">
      <PortId>17301543</PortId>
      <PortName>Mon1</PortName>
      <Active>true</Active>
    </Element>
    <Element Id="g1_n1_f2_i2">
      <PortId>17367079</PortId>
      <PortName>Mon2</PortName>
      <Active>true</Active>
    </Element>
  </Inports>
</FunctionBlock>
```

8.1.1.1.2 Mappings

In diesem Element nehmen Sie die Verknüpfungen zwischen den Eingängen und Ausgängen der Gruppen, FBs und Alias Devices vor. Dazu geben Sie die SourceId und die TargetId an. Das sind Ids, die Sie zuvor definiert haben. Falls es sich um eine Verknüpfung mit einem Port einer TwinSAFE-Gruppe oder eines FBs handelt, geben Sie den Variablennamen mit an.

```
<Mappings>
</Mappings>
```

Im Element [GroupInputs](#) [► 92] haben Sie bereits die für dieses Beispiel relevanten Gruppeneingänge „Run“ (g1_i2) und „ErrAck“ (g1_i1) definiert. Diese verknüpfen Sie mit den Eingängen des [StandardAliasDevice](#) [Id=„g1_a3“](#) [► 97] und [StandardAliasDevice](#) [Id=„g1_a4“](#) [► 97].

```
<Mapping TargetId="g1_i1" SourceId="g1_a3_i1" LocalVarName="GroupPort_RunStop"/>
<Mapping TargetId="g1_i2" SourceId="g1_a4_i1" LocalVarName="GroupPort_ErrAck"/>
```

Den Eingang „Restart“ von [FBEstop](#) [► 98] verknüpfen Sie mit dem Eingang des [StandardAliasDevice](#) [Id=„g1_a5“](#) [► 97].

```
<Mapping TargetId="g1_n1_f1_i1" SourceId="g1_a5_i1" LocalVarName="Restart"/>
```

Den Eingang „EStopIn1“ von [FBEstop](#) [► 98] verknüpfen Sie mit dem 3. Eingang des [SafetyAliasDevice](#) [Id=„g1_a1“](#) [► 93]. Den Eingang „EstopIn2“ vom [FBEstop](#) [► 98] verknüpfen Sie mit dem 4. Eingang des [SafetyAliasDevice](#) [Id=„g1_a1“](#) [► 93].

```
<Mapping TargetId="g1_n1_f1_i2" SourceId="g1_a1_i3" LocalVarName="S1_1"/>
<Mapping TargetId="g1_n1_f1_i3" SourceId="g1_a1_i4" LocalVarName="S1_2"/>
```

Das Rückführsignal legen Sie auf den Eingang „EDM1“ des [FBEstop](#) [► 98] und den Eingang „Mon2“ des [FBEEdm](#) [► 99]. Verknüpfen Sie dieses Signal [StandardAliasDevice](#) [Id=„g1_a6“](#) [► 98].

```
<Mapping TargetId="g1_n1_f1_i10" SourceId="g1_a6_i1" LocalVarName="Feedback"/>
<Mapping TargetId="g1_n1_f2_i2" SourceId="g1_a6_i1" LocalVarName="Feedback"/>
```

Den Ausgang „EStopOut“ des [FBEstop](#) [► 98] verknüpfen Sie mit dem 1. Eingang des [SafetyAliasDevice](#) [Id=„g1_a2“](#) [► 96].

```
<Mapping TargetId="g1_a2_o1" SourceId="g1_n1_f1_o2" LocalVarName="Output"/>
```

Zwischen dem Ausgang „EStopOut“ des [FBEstop](#) [► 98] und dem Eingang „Mon1“ des [FBEEdm](#) [► 99] definieren Sie eine Linienverknüpfung.

```
<Mapping TargetId="g1_n1_f2_i1" SourceId="g1_n1_f1_o2"/>
```


8.2 XSD-Validierung

Nachdem Sie die XML-Datei erstellt haben, validieren Sie diese XML mit der XSD-Datei mit derselben Versionsnummer.

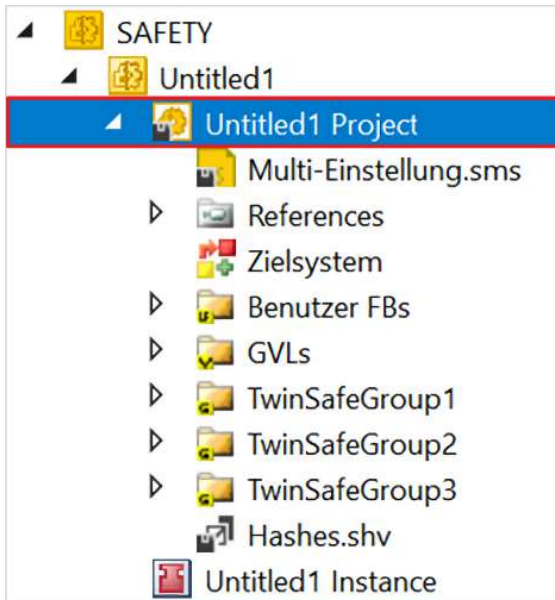
Nach der erfolgreichen Validierung können Sie den Import durchführen.

9 Automation Interface testen

9.1 Manueller Export

Sie haben die Möglichkeit die Export-Funktion des Automation Interface manuell zu testen.

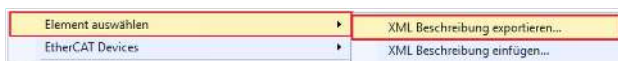
Gehen Sie dazu wie folgt vor:



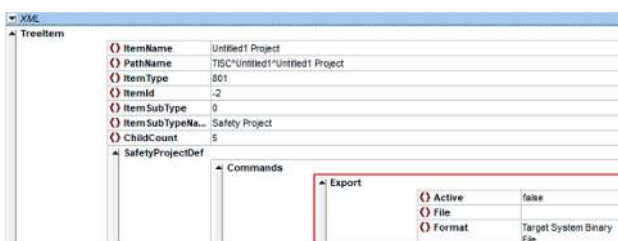
1. Safety-Projekt auswählen



2. In der Menüleiste „TwinCAT“ anklicken



3. Über die Schaltfläche „Element auswählen“ „XML Beschreibung exportieren...“ anklicken



4. XML-Datei öffnen

Unter dem Element „SafetyProjectDef“ finden Sie alle Kommandos, die der TwinSAFE-Editor für das Automation Interface zur Verfügung stellt. Für die Export-Funktion benötigen Sie das Kommando „Export“.



Um die Export-Funktion zu aktivieren, müssen Sie als nächstes 3 Werte setzen:

5. Element „Active“ auf „true“ setzen

6. Im Element „File“ einen Dateinamen eingeben

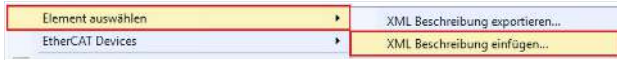
7. Im Element „Format“ eins der folgenden Formate auswählen:



• TwinCAT 3 Xml Export Format

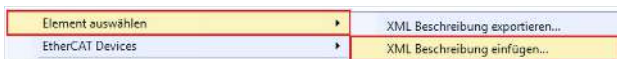
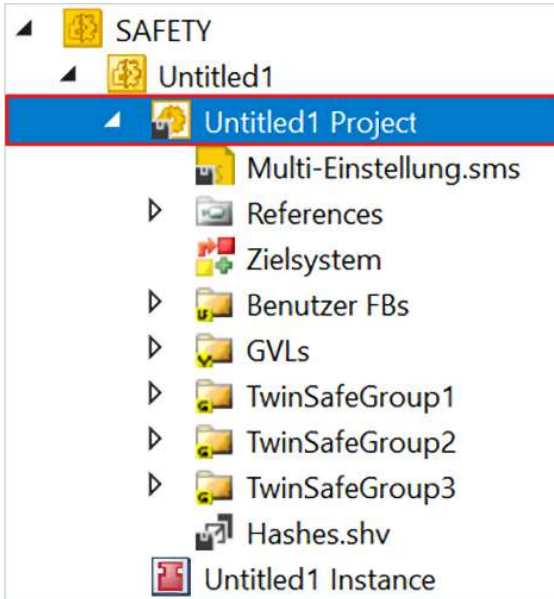
• TwinCAT 2 Xml Export Format

• Target System Binary File



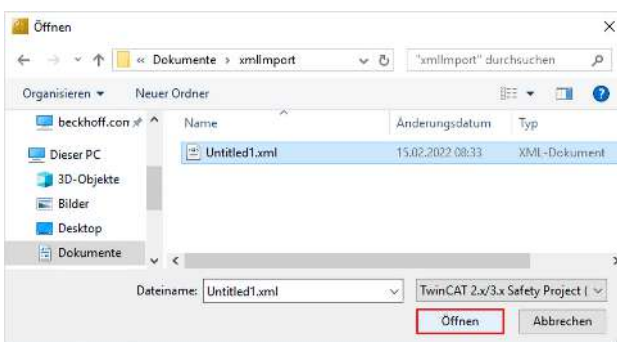
8. Änderungen in der Datei abspeichern
9. Safety-Projekt auswählen
10. In der Menüleiste „TwinCAT“ anklicken
11. Über die Schaltfläche „Element auswählen“, „XML Beschreibung einfügen...“ anklicken

9.2 Manueller Import



1. Safety-Projekt auswählen

2. In der Menüleiste „TwinCAT“ anklicken
3. Über die Schaltfläche „Element auswählen“, „XML Beschreibung einfügen...“ anklicken



4. Speicherort auswählen
5. Modifizierte XML-Datei auswählen
6. Mit „Öffnen“ bestätigen


Die Modifizierung der XML-Datei entnehmen Sie dem Kapitel [Manueller Export](#) [102].


10 FMEDA

Die folgende Tabelle enthält die FMEDA für den XML-Export und XML-Import. Die Spalte *Fehlermodus* beinhaltet die Fehlerbeschreibung, Spalte *Effekt* beschreibt die Auswirkung und Spalte *Diagnose* beschreibt, wie der Fehler erkannt oder nicht erkannt wird.

⚠ VORSICHT				
FMEDA				
Die letzte Spalte <i>Erforderliche Maßnahmen des Anwenders</i> der folgenden Tabelle gibt an, ob der Anwender Maßnahmen ergreifen muss, um den unter <i>Fehlermodus</i> beschriebenen Fehler in einer sicheren Art und Weise zu beheben. Diese Maßnahmen müssen vom Anwender in Form von zum Beispiel Prozessbeschreibungen oder Software-Spezifikationen definiert und realisiert werden. Für den Nachweis der Richtigkeit und Wirksamkeit dieser Maßnahmen ist der Anwender allein verantwortlich.				

Die in diesem Kapitel erwähnte XSD-Validierung erfolgt nicht automatisch über den TwinCAT 3 Safety Editor. Für eine Validierung ist der Anwender selbstständig verantwortlich.

FMEDA ID	Fehlermodus	Effekt	Diagnose	Erforderliche Maßnahmen des Anwenders
				
1	Das TwinCAT 3-Safety-Projekt enthält Fehler.	Das Safety-Projekt kann nicht exportiert werden, weder manuell noch über das Automation Interface. In TwinCAT 3 ist die Schaltfläche „Export XML“ deaktiviert.	Über die Safety-Projekt-Verifikation im manuellen Betrieb von TwinCAT 3 können die Fehler angezeigt werden.	Beseitigen Sie die angegebenen Fehler aus der Safety-Projekt-Verifikation.
2	Das Safety-Projekt in XML-Format ist bezüglich der XSD nicht valide.	Solange die Daten in sich korrekt sind, wird das Projekt importiert. Andernfalls wird das Projekt abgelehnt.	Im manuellen Betrieb von TwinCAT 3 werden Import-Fehler in einem Dialog angezeigt. Bei Verwendung des Automation Interfaces werden Fehler nicht angezeigt.	Stellen Sie vor dem Xml-Import sicher, dass das zu importierende Safety-Projekt gegen die entsprechende XSD valide ist. Beseitigen Sie die angegebenen Fehler im Dialog.
3	Die XSD-Validierung des XML-Safety-Projekts ist erfolgreich, aber es sind zum Beispiel Tippfehler bei den PortIds, Types oder anderen Elementen enthalten.	Beim Importieren können Elemente nicht mehr richtig zugeordnet werden. Der Import wird abgebrochen.	Im manuellen Betrieb von TwinCAT 3 werden Import-Fehler in einem Dialog angezeigt. Bei Verwendung des Automation Interfaces werden Fehler nicht angezeigt.	Beseitigen Sie die angegebenen Fehler im Dialog.
4	Die XSD-Validierung des XML-Safety-Projekts ist erfolgreich, aber es fehlen zum Beispiel Verknüpfungen oder es sind sichere Adressen doppelt vergeben.	Das Projekt wird importiert, aber es ist in TE9000 invalide.	Über die Safety-Projekt-Verifikation im manuellen Betrieb von TwinCAT 3 können die Fehler angezeigt werden.	Beseitigen Sie die angegebenen Fehler aus der Safety-Projekt-Verifikation oder gegebenenfalls die Fehler in der automatischen Erzeugung der XML.
5	Die XSD-Validierung des XML-Safety-Projekts ist erfolgreich, aber es werden in den	Das Safety-Projekt wird importiert. Das Projekt ist valide falls es keine anderen Fehler gibt,	Im manuellen TwinCAT-Betrieb wird in einem Dialog angezeigt, warum	Beseitigen Sie die angegebenen Fehler aus dem TwinCAT-Dialog oder

FMEDA ID	Fehlermodus	Effekt	Diagnose	Erforderliche Maßnahmen des Anwenders
	Elementen „IoPath“ Pfade eingetragen, die bereits im TwinCAT-Projekt verwendet werden. Zum Beispiel kann so eine Eingangsvariable mit mehreren Ausgangsvariablen verlinkt werden.	aber die TwinCAT-Konfiguration kann nicht aktiviert werden.	das TwinCAT-Projekt nicht aktiviert werden kann.	 gegebenenfalls die Fehler in der automatischen Erzeugung der XML.
6	Die XSD-Validierung des XML-Safety-Projekts ist erfolgreich, aber es wird in der TargetSystemConfiguration ein IoPath eingetragen, der bereits für ein bestehendes Safety-Projekt in einem TwinCAT-Projekt verwendet wird, in das die XML importiert werden soll.	Das Safety-Projekt wird importiert. Das Projekt ist valide (falls es keine anderen Fehler gibt), aber das Prozessabbild des bereits vorher existierenden Projekts wird durch das importierte Safety-Projekt überschrieben.	Dieser Fall lässt sich durch Tests aufdecken.	Wählen Sie einen anderen TargetSystemConfiguration – IoPath, oder überprüfen Sie vor dem Import den IoPath, oder passen Sie die automatische Erzeugung der XML an.

Mit dem XML-Export und -Import wird keine Aktion auf einem sicherheitskritischen Gerät ausgeführt! Es ist eine reine Offline-Funktion.

Abbildungsverzeichnis

Abb. 1	TwinCATExport	21
Abb. 2	TwinCATExport Attribut	21
Abb. 3	Base configuration	22
Abb. 4	GeneralInformation	23
Abb. 5	References	24
Abb. 6	TargetSystemConfiguration Attribute	25
Abb. 7	TargetSystemConfiguration Elemente	26
Abb. 8	TargetSystemConfiguration - BackupRestore	28
Abb. 9	BackupRestore - IdRef	29
Abb. 10	TargetSystemConfiguration - TxPdo	29
Abb. 11	TxPdo – Element	30
Abb. 12	ApplicationConfiguration	31
Abb. 13	ApplicationConfiguration - UserFBs	32
Abb. 14	Application	33
Abb. 15	Network	34
Abb. 16	FBFunctionBlock (Ausschnitt)	35
Abb. 17	LoadSensingTable	39
Abb. 18	Outports	40
Abb. 19	Outport-Element	41
Abb. 20	UserFunctionBlock Attribute	43
Abb. 21	UserFunctionBlock Elemente	44
Abb. 22	ParameterPorts	46
Abb. 23	UserFunctionBlock Inports	48
Abb. 24	UserFBs comments	49
Abb. 25	UserFBs comments	50
Abb. 26	TwinSAFEGroups	51
Abb. 27	TwinSAFEGroup Options	53
Abb. 28	GroupOutputs	54
Abb. 29	AliasDevices	56
Abb. 30	SafetyAliasDevices Attribute	57
Abb. 31	SafetyAliasDevices Elemente	58
Abb. 32	DynamicPortSafeAddress	61
Abb. 33	SafetyParameters	61
Abb. 34	ComErrAck	62
Abb. 35	StandardAliasDevice Attribute	63
Abb. 36	StandardAliasDevice Elemente	64
Abb. 37	UserFBPortRelations	65
Abb. 38	Mappings	66
Abb. 39	TargetSystemProcessImage	67
Abb. 40	RxPdo Attribute	68
Abb. 41	RxPdo Element	68
Abb. 42	Multisettings	70
Abb. 43	TargetSystem	71
Abb. 44	AliasDevices	72

Abb. 45	SafeParameters	73
Abb. 46	FbParameters	74
Abb. 47	ConnectionType EL1904.....	89
Abb. 48	ConnectionType TwinSAFE SC	89
Abb. 49	Linking Mode.....	89
Abb. 50	Element hinzufügen	90
Abb. 51	LogicType.....	90

Mehr Informationen:
www.beckhoff.de/TE9000

Beckhoff Automation GmbH & Co. KG
Hülshorstweg 20
33415 Verl
Deutschland
Telefon: +49 5246 9630
info@beckhoff.com
www.beckhoff.com

