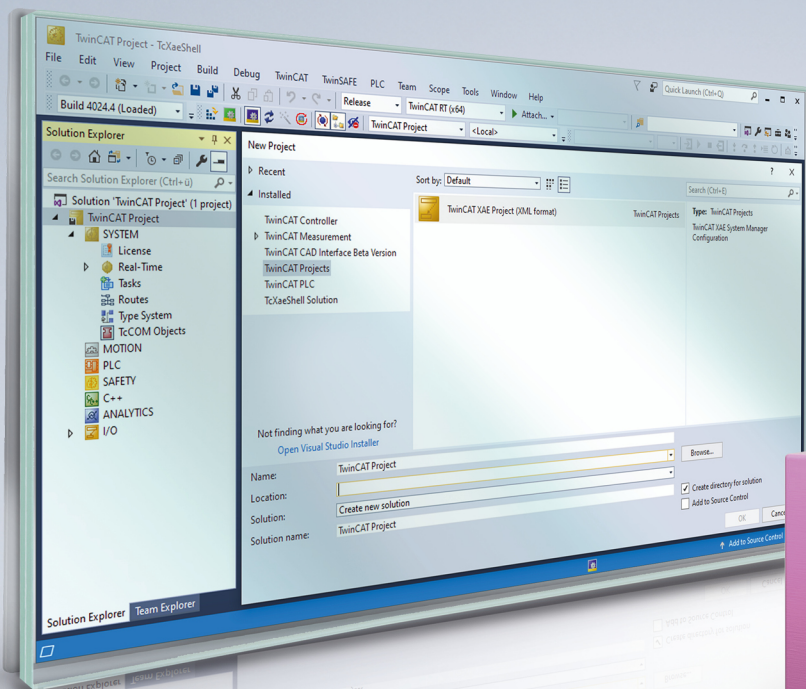


# BECKHOFF New Automation Technology

Handbuch | DE

# TF6500

TwinCAT 3 | IEC 60870-5-10x





# Inhaltsverzeichnis

<b>1</b>	<b>Vorwort</b>	<b>9</b>
1.1	Hinweise zur Dokumentation	9
1.2	Zu Ihrer Sicherheit	10
1.3	Hinweise zur Informationssicherheit	11
<b>2</b>	<b>Übersicht</b>	<b>12</b>
2.1	Produktbeschreibung	12
2.2	Kompatibilitätslisten	14
2.2.1	Kompatibilitätsliste nach IEC 60870-5-101 (master)	14
2.2.2	Kompatibilitätsliste nach IEC 60870-5-101 (slave)	32
2.2.3	Kompatibilitätsliste nach IEC 60870-5-102 (master)	49
2.2.4	Kompatibilitätsliste nach IEC 60870-5-103 (master)	57
2.2.5	Kompatibilitätsliste nach IEC 60870-5-104 (master)	68
2.2.6	Kompatibilitätsliste nach IEC 60870-5-104 (slave)	86
<b>3</b>	<b>Installation</b>	<b>104</b>
3.1	Systemvoraussetzungen	104
3.2	Installation	104
3.3	Installation TwinCAT/BSD	107
3.4	Installation Windows CE	107
3.5	Lizenzierung	109
3.6	Migration von TwinCAT 2	111
<b>4</b>	<b>Technische Einführung</b>	<b>114</b>
4.1	IEC 60870-5-101	114
4.1.1	IEC 60870-5-101 telegram structure	115
4.1.2	ASDU-Objektbeschreibung	119
4.2	IEC 60870-5-102	241
4.2.1	IEC 60870-5-102 telegram structure	242
4.2.2	ASDU-Objektbeschreibung	244
4.3	IEC 60870-5-103	295
4.3.1	IEC 60870-5-103 telegram structure	296
4.3.2	ASDU-Objektbeschreibung	298
4.4	IEC 60870-5-104	330
4.4.1	IEC 60870-5-104 telegram structure	331
4.4.2	ASDU-Objektbeschreibung	335
<b>5</b>	<b>SPS-Bibliotheken</b>	<b>385</b>
5.1	Funktionsbausteine	385
5.1.1	High Level	385
5.1.2	Low Level	399
5.1.3	FB_IEC870_PartyLineCtrl	412
5.1.4	FB_IEC870_SerialLineCtrl	417
5.1.5	FB_IEC870_DebugLogFifo	419
5.1.6	FB_IEC870_5_101ErrorFifo	420
5.2	Funktionen	421
5.2.1	ASDU101COT_TO_STRING	421

5.2.2	ASDU102COT_TO_STRING.....	422
5.2.3	ASDU103COT_TO_STRING.....	422
5.2.4	ASDU101TYPEID_TO_STRING.....	423
5.2.5	ASDU102TYPEID_TO_STRING.....	424
5.2.6	ASDU103TYPEID_TO_STRING.....	424
5.2.7	CP40Time2a_TO_SYSTEMTIME.....	425
5.2.8	CP56Time2b_TO_CP40Time2a.....	426
5.2.9	CP56Time2b_TO_SYSTEMTIME.....	426
5.2.10	SYSTEMTIME_TO_CP56Time2b.....	427
5.2.11	CP56Time2a_TO_SYSTEMTIME.....	428
5.2.12	SYSTEMTIME_TO_CP56Time2a.....	428
5.2.13	CP56Time2a_TO_CP32Time2a.....	429
5.2.14	CP56Time2a_TO_CP24Time2a.....	430
5.2.15	CP24IOA_TO_DWORD.....	430
5.2.16	DWORD_TO_CP24IOA.....	431
5.2.17	IEC101ASDU_TO_STRING.....	431
5.2.18	IEC102ASDU_TO_STRING.....	432
5.2.19	IEC103ASDU_TO_STRING.....	432
5.2.20	INT_TO_NORMF16.....	433
5.2.21	QOI_TO_COT.....	434
5.2.22	RQT_TO_COT.....	434
5.2.23	F_iecInitAOEntry.....	435
5.2.24	F_iecSetAOQuality.....	437
5.2.25	F_iecGetAOQuality.....	438
5.2.26	F_iecGetAOTimeTag.....	439
5.2.27	F_iecIncVTI.....	441
5.2.28	F_iecDecVTI.....	441
5.2.29	F_iecCopyBufferToStream.....	442
5.2.30	F_iecCopyBufferToStreamByOffset.....	443
5.2.31	F_iecCopyStreamToBuffer.....	444
5.2.32	F_iecCopyStreamToBufferByOffset.....	446
5.2.33	F_iecCopyStreamToStream.....	447
5.2.34	F_iecMoveStreamToBuffer.....	449
5.2.35	F_iecMoveStreamToStream.....	450
5.2.36	F_iecResetStream.....	451
5.2.37	F_iecCreateTableHnd.....	452
5.2.38	F_iecAddTableEntry.....	453
5.2.39	F_iecGetPosOfTableEntry.....	456
5.2.40	F_iecLookupTableEntry.....	458
5.2.41	F_iecRemoveTableEntry.....	460
5.2.42	F_iecCmpAddrOctets.....	461
5.2.43	F_iecGetSPI.....	462
5.2.44	F_iecGetDPI.....	463
5.2.45	F_iecGetSCS.....	464
5.2.46	F_iecGetDCS.....	464
5.2.47	F_iecSetSPI.....	465

5.2.48	F_iecSetDPI.....	466
5.2.49	F_iecSetSCS.....	467
5.2.50	F_iecSetDCS.....	468
5.2.51	F_iecChangeLinkLayerMode.....	468
5.2.52	F_iecApdu101ToAsduLen.....	471
5.2.53	F_iecApdu102ToAsduLen.....	471
5.2.54	F_iecApdu103ToAsduLen.....	472
5.2.55	F_iecApdu104ToAsduLen.....	473
5.2.56	IEC870_5_102_DEFAULT_ASDFMTPARAMS.....	473
5.2.57	IEC870_5_104_DEFAULT_ASDFMTPARAMS.....	474
5.2.58	Veraltet.....	475
5.3	Datentypen.....	480
5.3.1	ST_IEC870_5_101ExSystemInterface.....	480
5.3.2	ST_IEC870_5_101SystemInterface.....	480
5.3.3	ST_IEC870_5_101SystemServices.....	481
5.3.4	ST_IEC870_5_101TBuffer.....	481
5.3.5	ST_IEC870_5_101AODBEntry.....	482
5.3.6	ST_IEC870_5_101AOGen.....	483
5.3.7	ST_IEC870_5_101AOCfg.....	483
5.3.8	ST_IEC870_5_101DataUnit_Ident.....	484
5.3.9	ST_IEC870_5_101AOInfoObj.....	485
5.3.10	ST_IEC870_5_101Stream.....	485
5.3.11	ST_IEC870_5_101SystemParams.....	487
5.3.12	ST_IEC870_5_101DeviceInterface.....	489
5.3.13	ST_IEC870_5_101AsduFmtParams.....	490
5.3.14	ST_IEC870_5_101ErrorFifoEntry.....	490
5.3.15	ST_IEC870_5_101AcquisitionParams.....	491
5.3.16	ST_IEC870_5_101TestPollParams.....	493
5.3.17	ST_IEC870_5_101ClockPollParams.....	494
5.3.18	ST_IEC870_5_101GenroPollParams.....	494
5.3.19	ST_IEC870_5_101CoroPollParams.....	495
5.3.20	ST_IEC870_5_101GenCmdPollParams.....	496
5.3.21	ST_IEC870_5_101DelayPollParams.....	497
5.3.22	ST_IEC870_5_101HashTableKey.....	497
5.3.23	E_IEC870_5_101AcquisitionMode.....	498
5.3.24	E_IEC870_5_101TcTypeID.....	500
5.3.25	E_IEC870_5_102TypeID.....	503
5.3.26	E_IEC870_5_103MTypeID.....	505
5.3.27	E_IEC870_5_103CTTypeID.....	505
5.3.28	E_IEC870_5_101IOMappingType.....	506
5.3.29	E_IEC870_5_101AsduAddrSize.....	506
5.3.30	E_IEC870_5_101COTSize.....	507
5.3.31	E_IEC870_5_101LinkAddrSize.....	507
5.3.32	E_IEC870_5_101ObjAddrSize.....	508
5.3.33	E_IEC870_5_101ErrorSourceID.....	508
5.3.34	E_IEC870_5_101ClassType.....	509

5.3.35	E_IEC870_5_101COTType .....	509
5.3.36	E_IEC870_5_102COTType .....	512
5.3.37	E_IEC870_5_103MCOT .....	513
5.3.38	E_IEC870_5_103CCOT.....	514
5.3.39	E_IEC870_5_101FifoDbgFlags .....	514
5.3.40	E_IEC870_5_101AODBType.....	515
5.3.41	E_IEC870_5_101InitSeqStep .....	515
5.3.42	E_IEC870_5_101FBufferState.....	520
5.3.43	E_IEC870_5_101SCS .....	520
5.3.44	E_IEC870_5_101DCS .....	520
5.3.45	E_IEC870_5_101COI .....	521
5.3.46	E_IEC870_5_101QOI .....	521
5.3.47	E_IEC870_5_101QL.....	522
5.3.48	E_IEC870_5_101FRZ.....	522
5.3.49	E_IEC870_5_101RQT .....	523
5.3.50	E_IEC870_5_101QRP .....	523
5.3.51	E_IEC870_5_101QU .....	524
5.3.52	E_IEC870_5_101ES .....	524
5.3.53	E_IEC870_5_101KPA.....	525
5.3.54	E_IEC870_5_101QPA .....	525
5.3.55	E_IEC870_5_101RCS .....	526
5.3.56	E_IEC870_5_101SPI.....	526
5.3.57	E_IEC870_5_101DPI.....	527
5.3.58	E_IEC870_5_101FrameType .....	527
5.3.59	E_IEC870_5_101LinkMode .....	527
5.3.60	E_IEC870_5_101SerialLinkState.....	528
5.3.61	E_IEC870_5_101PartyLineMode.....	528
5.3.62	E_IEC870_5_101LinkReset.....	529
5.3.63	E_IEC870_DEVICE_TYPE .....	529
5.3.64	E_IEC870_5_104DataTransferState .....	530
5.3.65	T_HSERIALCTRL .....	530
5.3.66	T_HAODBTable .....	531
5.3.67	T_CP56Time2a.....	531
5.3.68	T_CP56Time2b.....	532
5.3.69	T_CP40Time2a .....	533
5.3.70	T_CP32Time2a.....	533
5.3.71	T_CP24Time2a.....	534
5.3.72	T_CP16Time2a.....	534
5.3.73	T_CP24IOA.....	535
5.3.74	T_IEC870_5_101COTBits .....	535
5.3.75	ST_IEC870_5_101FBufferCfg .....	536
5.3.76	ST_IEC870_5_101FBufferStatus.....	537
5.3.77	ST_IEC870_5_101ProtocolParams.....	538
5.3.78	ST_IEC870_5_102TBuffer.....	544
5.3.79	ST_IEC870_5_102AOGen.....	545
5.3.80	ST_IEC870_5_102AOInfoObj.....	545

5.3.81	ST_IEC870_5_102DataUnit_Ident.....	546
5.3.82	ST_IEC870_5_103TBuffer.....	547
5.3.83	ST_IEC870_5_103AOGen.....	547
5.3.84	ST_IEC870_5_103AOInfoObj.....	548
5.3.85	ST_IEC870_5_103DataUnit_Ident.....	549
5.3.86	ST_IEC870_5_104ProtocolParams.....	549
5.3.87	ST_IEC870_5_104ExSystemInterface.....	552
5.3.88	ST_IEC870_5_104ServerConnection.....	552
5.3.89	ST_IEC870_5_104GrpStatus.....	553
5.3.90	ST_IEC870_5_104SystemInterface.....	553
5.4	Globale Konstanten.....	554
5.4.1	Group Konfigurationsflags.....	554
5.4.2	Qualitätsflags.....	556
5.4.3	IEC 60870-5-10x Fehlercodes.....	557
5.4.4	Parameterliste.....	559
5.4.5	Bibliotheksversion.....	560
<b>6</b>	<b>Beispiele.....</b>	<b>561</b>
6.1	Konfiguration der seriellen Schnittstellen.....	561
6.2	IEC 60870-5-101.....	564
6.2.1	Zentralstation.....	564
6.2.2	Unterstation.....	592
6.3	IEC 60870-5-102.....	623
6.3.1	Low-Level Interface.....	623
6.4	IEC 60870-5-103.....	627
6.4.1	Low-Level Interface.....	627
6.5	IEC 60870-5-104.....	631
6.5.1	Zentralstation.....	631
6.5.2	Unterstation.....	655
<b>7</b>	<b>Anhang.....</b>	<b>685</b>
7.1	IEC 60870-5-101.....	685
7.1.1	TwinCAT IEC 60870-5-101 Fehlercodes.....	685
7.1.2	Fehlersuche/Diagnose.....	690
7.1.3	Debugmeldungen in TwinCAT->Error List.....	691
7.2	IEC 60870-5-102.....	694
7.2.1	TwinCAT IEC 60870-5-102 Fehlercodes.....	694
7.2.2	Fehlersuche/Diagnose.....	699
7.2.3	Debugmeldungen in TwinCAT->Error List.....	699
7.3	IEC 60870-5-103.....	701
7.3.1	TwinCAT IEC 60870-5-103 Fehlercodes.....	701
7.3.2	Fehlersuche/Diagnose.....	706
7.3.3	Debugmeldungen in TwinCAT->Error List.....	706
7.4	IEC 60870-5-104.....	708
7.4.1	TwinCAT IEC 60870-5-104 Fehlercodes.....	708
7.4.2	Fehlersuche/Diagnose.....	777
7.4.3	Debugmeldungen in TwinCAT->Error List.....	778

---

7.4.4	Firewall Einstellungen -104 Protokoll .....	781
7.5	Glossar .....	781
7.6	FAQ - Häufig gestellte Fragen und Antworten .....	782



# 1 Vorwort

## 1.1 Hinweise zur Dokumentation

Diese Beschreibung wendet sich ausschließlich an ausgebildetes Fachpersonal der Steuerungs- und Automatisierungstechnik, das mit den geltenden nationalen Normen vertraut ist.

Zur Installation und Inbetriebnahme der Komponenten ist die Beachtung der Dokumentation und der nachfolgenden Hinweise und Erklärungen unbedingt notwendig.

Das Fachpersonal ist verpflichtet, stets die aktuell gültige Dokumentation zu verwenden.

Das Fachpersonal hat sicherzustellen, dass die Anwendung bzw. der Einsatz der beschriebenen Produkte alle Sicherheitsanforderungen, einschließlich sämtlicher anwendbaren Gesetze, Vorschriften, Bestimmungen und Normen erfüllt.

### Disclaimer

Diese Dokumentation wurde sorgfältig erstellt. Die beschriebenen Produkte werden jedoch ständig weiterentwickelt.

Wir behalten uns das Recht vor, die Dokumentation jederzeit und ohne Ankündigung zu überarbeiten und zu ändern.

Aus den Angaben, Abbildungen und Beschreibungen in dieser Dokumentation können keine Ansprüche auf Änderung bereits gelieferter Produkte geltend gemacht werden.

### 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 in dieser Dokumentation enthaltenen Marken oder Kennzeichen durch Dritte kann zu einer Verletzung von Rechten der Inhaber der entsprechenden Bezeichnungen führen.

### Patente

Die EtherCAT-Technologie ist patentrechtlich geschützt, insbesondere durch folgende Anmeldungen und Patente:

EP1590927, EP1789857, EP1456722, EP2137893, DE102015105702

mit den entsprechenden Anmeldungen und Eintragungen in verschiedenen anderen Ländern.



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

### Copyright

© Beckhoff Automation GmbH & Co. KG, Deutschland.

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

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

## 1.2 Zu Ihrer Sicherheit

### Sicherheitsbestimmungen

Lesen Sie die folgenden Erklärungen zu Ihrer Sicherheit.  
Beachten und befolgen Sie stets produktspezifische Sicherheitshinweise, die Sie gegebenenfalls an den entsprechenden Stellen in diesem Dokument vorfinden.

### Haftungsausschluss

Die gesamten Komponenten werden je nach Anwendungsbestimmungen in bestimmten Hard- und Software-Konfigurationen ausgeliefert. Änderungen der Hard- oder Software-Konfiguration, die über die dokumentierten Möglichkeiten hinausgehen, sind unzulässig und bewirken den Haftungsausschluss der Beckhoff Automation GmbH & Co. KG.

### Qualifikation des Personals

Diese Beschreibung wendet sich ausschließlich an ausgebildetes Fachpersonal der Steuerungs-, Automatisierungs- und Antriebstechnik, das mit den geltenden Normen vertraut ist.

### Signalwörter

Im Folgenden werden die Signalwörter eingeordnet, die in der Dokumentation verwendet werden. Um Personen- und Sachschäden zu vermeiden, lesen und befolgen Sie die Sicherheits- und Warnhinweise.

### Warnungen 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**

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.

## 1.3 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 Übersicht

### 2.1 Produktbeschreibung

In der Fernwirktechnik müssen Geräte verschiedener Hersteller miteinander kommunizieren. Die IEC 60870-5-Familie definiert auf der Basis der fünf Grundnormen IEC 60870-5-1 (Telegrammformate), -2 (Übertragungsprozeduren/Verbindungsschicht), -3 (Strukturen/Anwendungsdaten), -4 (Informationselemente) und -5 (grundlegende Anwendungsfunktionen) die anwendungsbezogenen Normen IEC 60870-5-101, -102, -103 und 104. Durch die international genormten Fernwirkprotokolle IEC 60870-5-101/-102/-103 kann eine serielle bzw. über das Fernwirkprotokoll IEC 60870-5-104 eine TCP/IP-basierte Datenübertragung realisiert werden.

Die Hersteller sind nicht verpflichtet die komplette Norm in den Geräten zu implementieren. Aus diesem Grund kann es zu Inkompatibilitäten zwischen den Geräten bei der Inbetriebnahme kommen. Um dem vorzubeugen bieten Hersteller zu jedem Gerät eine passende Kompatibilitätsliste an. In dieser Liste sind die implementierten Funktionen aufgelistet bzw. markiert. Mit der Hilfe der Kompatibilitätsliste können die benötigten Funktionalitäten zweier Geräte im Vorfeld verglichen werden. Bitte beachten Sie auch die Kompatibilitätslisten zu den TwinCAT IEC 60870-5-10x-Bibliotheken.

Mit den SPS-Funktionen und -Funktionsbausteinen können Zentralstationen und/oder Unterstationen (Master/Slave) nach der IEC60870-5-101/102/103/104 Norm in der TwinCAT SPS realisiert werden. Die SPS-Bibliotheken verfügen teilweise über zwei Software-Schnittstellen ("Low level"- und "High level"-Schnittstelle). Die Endapplikation setzt auf einer dieser Schnittstellen auf. High Level eignet sich für schnelle Applikationen mit geringeren Anforderungen. Low Level bietet dagegen viele Freiheitsgrade und eignet sich hervorragend für komplexe Anwendungen. Welche Schnittstelle zum Einsatz kommt hängt damit von der Endapplikation ab.

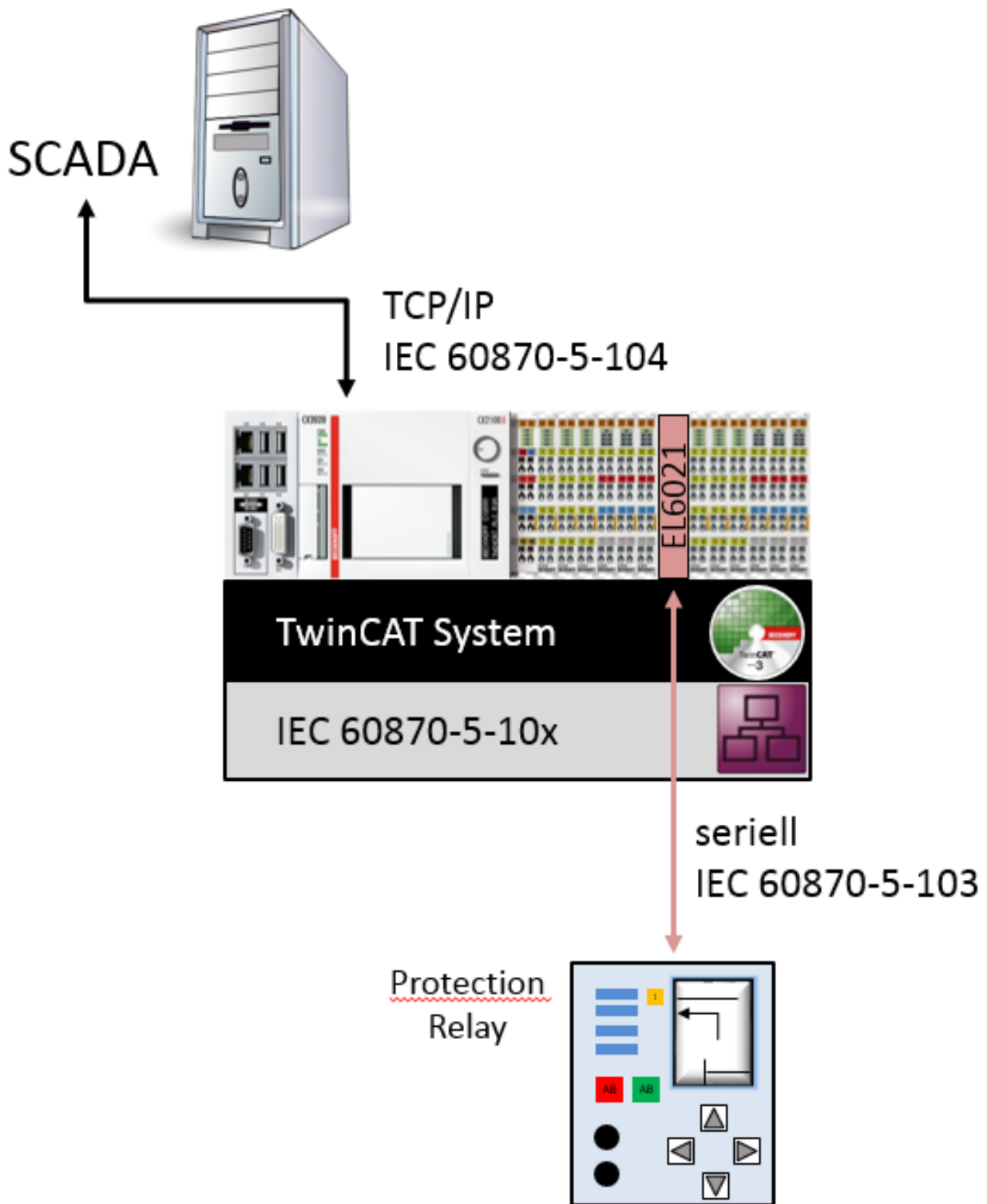
#### Produktkomponenten

Die Function TF6500 | IEC 60870-5-10x besteht aus den folgenden Komponenten, welche automatisch beim Setup installiert werden:

- **Funktionen** : TF6310 und TF6340.
- **SPS-Bibliothek**: Tc2\_IEC60870\_5\_10x (Tc2\_Tcplp, Tc2\_SerialCom, installiert mit TF6310 und TF6340).
- **Hintergrundprogramm**: TwinCAT TCP/IP Connection Server (für Kommunikation, wird mit TF6310 installiert).

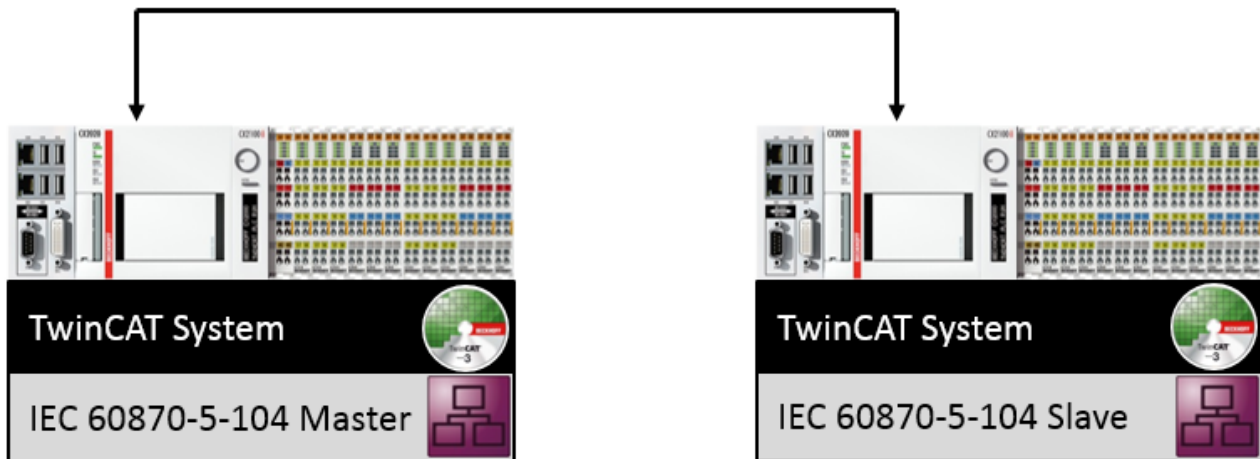
#### Funktionsprinzip und Anwendungsszenarien:

IEC 60870-5-104 Slave mit unterlagertem -103 Master für die Anbindung von Schutzgeräten.



Eine Steuerung als IEC 60870-5-104 Master und eine Steuerung als angebundener -104 Slave

TCP/IP → IEC 60870-5-104

**Nächste Schritte:**

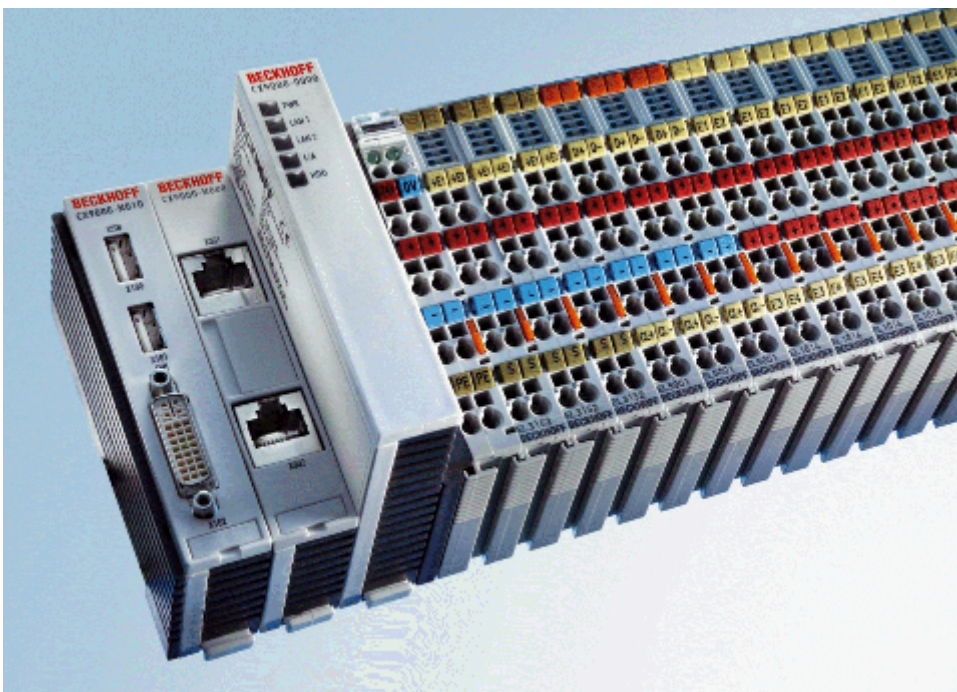
Auswahl der Schnittstelle High oder Low Level:

- [Technische Einführung IEC 60870-5-101](#) [▶ 114]
- [Technische Einführung IEC 60870-5-102](#) [▶ 241]
- [Technische Einführung IEC 60870-5-103](#) [▶ 295]
- [Technische Einführung IEC 60870-5-104](#) [▶ 330]
- [Beispiele](#) [▶ 561] für die verschiedenen Protokolle.

## 2.2 Kompatibilitätslisten

### 2.2.1 Kompatibilitätsliste nach IEC 60870-5-101 (master)

für TwinCAT SPS Bibliothek: IEC 60870-5-101 Zentralstation (master).



Version: 1.2  
Datum: 07.03.2014

Hier können Sie die Kompatibilitätsliste öffnen/abspeichern (gezipptes PDF): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/984583435.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/984583435.zip).

**Inhaltsverzeichnis:**

- [Allgemeine Informationen \[▶ 15\]](#)
  - [Projektinformationen \[▶ 16\]](#)
- [Kompatibilität \[▶ 16\]](#)
  - [System oder Gerät \[▶ 16\]](#)
  - [Netzkonfiguration \[▶ 16\]](#)
  - [Physikalische Schicht \[▶ 16\]](#)
  - [Verbindungsschicht \[▶ 16\]](#)
  - [Anwendungsschicht \[▶ 17\]](#)
  - Auswahl genormter ASDU's

Prozessinformation in Überwachungsrichtung

Prozessinformation in Steuerungsrichtung

Systeminformation in Überwachungsrichtung

Systeminformation in Steuerungsrichtung

Parameter in Steuerungsrichtung

Dateiübermittlung

- [Grundlegende Anwendungsfunktionen \[▶ 27\]](#)
  - Stationsinitialisierung
  - Zyklische Datenübertragung
  - Abrufprozedur
  - Spontane Übertragung
  - Doppelübertragung
  - Stationsabfrage
  - Uhrzeitsynchronisation
  - Befehlsübertragung
  - Zählerabfrage
  - Laden eines Parameters
  - Parameter für Aktivierung
  - Prüfprozedur
  - Dateiübermittlung
  - Hintergrundabfrage
  - Telegrammlaufzeiterfassung
- [Anhang \[▶ 30\]](#)
  - [Support und Service \[▶ 30\]](#)

**Allgemeine Informationen**

In dieser Kompatibilitätsliste finden Sie die unterstützten Funktionen innerhalb der IEC 870-5-101.

Die Kompatibilität der einzelnen Komponenten muss von allen Beteiligten für das jeweilige Projekt vereinbart werden.

**Projektinformationen**

Projekt-Nr.:

---

Projekt:

---

Zuständig:

---

Datum:

---

**Kompatibilität**

Die ausgewählten Parameter sollten in den weißen Kontrollfeldern wie folgt markiert werden:

- Funktion oder ASDU wird nicht benutzt
- Funktion oder ASDU wird wie genormt benutzt (Vorzugswert)
- Funktion oder ASDU wird im Umkehrmodus benutzt
- Funktion oder ASDU wird im Regel- und Umkehrmodus benutzt

**System oder Gerät**

- Systemfestlegung
- Festlegungen für die Zentralstation (MASTER)
- Festlegungen für die Unterstation

**Netzkonfiguration**

- |                                     |   |                                     |   |
|-------------------------------------|---|-------------------------------------|---|
| <input checked="" type="checkbox"/> | End-End-Konfiguration[en: Point-to-point]                   | <input checked="" type="checkbox"/> | Linienkonfiguration[en: Multipoint-partyline] |
| <input checked="" type="checkbox"/> | Mehrfach-End-End-Konfiguration[en: Multiple point-to-point] | <input type="checkbox"/>            | Sternkonfiguration[en: Multipoint-star]       |

**Physikalische Schicht***Übertragungsgeschwindigkeit:*

- |                                     |            |                                     |             |                          |             |
|-------------------------------------|------------|-------------------------------------|-------------|--------------------------|-------------|
| <input type="checkbox"/>            | 100 bit/s  | <input checked="" type="checkbox"/> | 2400 bit/s  | <input type="checkbox"/> | 56000 bit/s |
| <input type="checkbox"/>            | 200 bit/s  | <input checked="" type="checkbox"/> | 4800 bit/s  | <input type="checkbox"/> | 64000 bit/s |
| <input checked="" type="checkbox"/> | 300 bit/s  | <input checked="" type="checkbox"/> | 9600 bit/s  |                          |             |
| <input checked="" type="checkbox"/> | 600 bit/s  | <input checked="" type="checkbox"/> | 19200 bit/s |                          |             |
| <input checked="" type="checkbox"/> | 1200 bit/s | <input type="checkbox"/>            | 38400 bit/s |                          |             |

**Verbindungsschicht**

Nach dieser Anwendungsbezogenen Norm werden ausschließlich Telegrammformat FT 1.2, Einzelzeichen 1 und das feste Zeitüberwachungsintervall benutzt.

*Adressfeld der Verbindungsschicht:*



- Nicht vorhanden (nur symmetrische Übertragung)
- Ein Oktett
- Zwei Oktette
- Strukturiert
- Unstrukturiert

Übertragungsprozedur der Verbindungsschicht:

- Symmetrische Übertragung
- Unsymmetrische Übertragung

Telegrammlänge:

Maximale Länge L 255

Wird unsymmetrisch übertragen, werden die folgenden ASDUs als Anwenderdaten mit den angegebenen Übertragungsursachen mit der Datenklasse 2 (niedrige Priorität) zurückgesendet.

- Die genormte Zuweisung von ASDU zur Datenklasse 2 wird wie folgt angewendet:

Typkennung

Übertragungsursache

- Eine spezielle Zuweisung von ASDU zur Datenklasse 2 wird wie folgt angewendet:

Typkennung

Übertragungsursache



Als Antwort auf eine Anforderung nach Daten der Klasse 2 darf eine Unterstation Daten der Datenklasse 1 übertragen, wenn keine Daten der Datenklasse 2 vorhanden sind (konfigurierbar).

Anwendungsschicht

Übertragungsbetriebsart für Anwendungsdaten

Nach dieser Anwendungsbezogenen Norm wird ausschließlich Betriebsart 1 (niedrigstwertiges Oktett zuerst) nach 4.10 von IEC 60870-5-4 benutzt.

Gemeinsame Adresse der ASDU

- Ein Oktett  Zwei Oktette

Adresse des Informationsobjekts

- Ein Oktett  Strukturiert
- Zwei Oktette  Unstrukturiert
- Drei Oktette

**Übertragungsursache**



Ein Oktett



Zwei Oktette (mit  
Herkunftsadresse).

Mit 0 vorbesetzt, falls  
Herkunftsadresse nicht vorhanden.

**Auswahl genormter ASDU**

*Tab. 1: Prozessinformation in Überwachungsrichtung*

<input checked="" type="checkbox"/>	<1> := Einzelmeldung	M_SP_NA_1
<input checked="" type="checkbox"/>	<2> := Einzelmeldung mit Zeitmarke	M_SP_TA_1
<input checked="" type="checkbox"/>	<3> := Doppelmeldung	M_DP_NA_1
<input checked="" type="checkbox"/>	<4> := Doppelmeldung mit Zeitmarke	M_DP_TA_1
<input checked="" type="checkbox"/>	<5> := Stufenstellungsmeldung	M_ST_NA_1
<input checked="" type="checkbox"/>	<6> := Stufenstellungsmeldung mit Zeitmarke	M_ST_TA_1
<input checked="" type="checkbox"/>	<7> := Bitmuster von 32 bit	M_BO_NA_1
<input checked="" type="checkbox"/>	<8> := Bitmuster von 32 bit mit Zeitmarke	M_BO_TA_1
<input checked="" type="checkbox"/>	<9> := Messwert, normierter Wert	M_ME_NA_1
<input checked="" type="checkbox"/>	<10> := Messwert, normierter Wert mit Zeitmarke	M_ME_TA_1
<input checked="" type="checkbox"/>	<11> := Messwert, skaliertes Wert	M_ME_NB_1
<input checked="" type="checkbox"/>	<12> := Messwert, skaliertes Wert mit Zeitmarke	M_ME_TB_1
<input checked="" type="checkbox"/>	<13> := Messwert, verkürzte Gleitkommazahl	M_ME_NC_1
<input checked="" type="checkbox"/>	<14> := Messwert, verkürzte Gleitkommazahl mit Zeitmarke	M_ME_TC_1
<input checked="" type="checkbox"/>	<15> := Zählwerte	M_IT_NA_1
<input checked="" type="checkbox"/>	<16> := Zählwerte mit Zeitmarke	M_IT_TA_1
<input checked="" type="checkbox"/>	<17> := Schutzereignis mit Zeitmarke	M_EP_TA_1
<input checked="" type="checkbox"/>	<18> := Geblockte Anregungen des Schutzes mit Zeitmarke	M_EP_TB_1
<input checked="" type="checkbox"/>	<19> := Geblockte Auslösungen des Schutzes mit Zeitmarke	M_EP_TC_1
<input checked="" type="checkbox"/>	<20> := Geblockte Einzelmeldungen mit Zustandsanzeige	M_PS_NA_1
<input checked="" type="checkbox"/>	<21> := Messwert, normierter Wert ohne Qualitätskennung	M_ME_ND_1
<input checked="" type="checkbox"/>	<30> := Einzelmeldung mit Zeitmarke CP56Time2a	M_SP_TB_1
<input checked="" type="checkbox"/>	<31> := Doppelmeldung mit Zeitmarke CP56Time2a	M_DP_TB_1
<input checked="" type="checkbox"/>	<32> := Stufenstellungsmeldung mit Zeitmarke CP56Time2a	M_ST_TB_1
<input checked="" type="checkbox"/>	<33> := Bitmuster von 32 bit mit Zeitmarke CP56Time2a	M_BO_TB_1
<input checked="" type="checkbox"/>	<34> := Messwert, normierter Wert mit Zeitmarke CP56Time2a	M_ME_TD_1
<input checked="" type="checkbox"/>	<35> := Messwert, skaliertes Wert mit Zeitmarke CP56Time2a	M_ME_TE_1
<input checked="" type="checkbox"/>	<36> := Messwert, verkürzte Gleitkommazahl mit Zeitmarke CP56Time2a	M_ME_TF_1
<input checked="" type="checkbox"/>	<37> := Zählwerte mit Zeitmarke CP56Time2a	M_IT_TB_1
<input checked="" type="checkbox"/>	<38> := Schutzereignis mit Zeitmarke CP56Time2a	M_EP_TD_1
<input checked="" type="checkbox"/>	<39> := Geblockte Anregungen des Schutzes mit Zeitmarke CP56Time2a	M_EP_TE_1

<input checked="" type="checkbox"/>	<40> := Geblockte Auslösungen des Schutzes mit Zeitmarke CP56Time2a	M_EP_TF_1
-------------------------------------	---	-----------

*Tab. 2: Prozessinformation in Steuerungsrichtung*

<input checked="" type="checkbox"/>	<45> := Einzelbefehl	C_SC_NA_1
<input checked="" type="checkbox"/>	<46> := Doppelbefehl	C_DC_NA_1
<input type="checkbox"/>	<47> := Stufenstellbefehl	C_RC_NA_1
<input checked="" type="checkbox"/>	<48> := Sollwert-Stellbefehl, normierter Wert	C_SE_NA_1
<input checked="" type="checkbox"/>	<49> := Sollwert-Stellbefehl, skaliertes Wert	C_SE_NB_1
<input checked="" type="checkbox"/>	<50> := Sollwert-Stellbefehl, verkürzte Gleitkommazahl	C_SE_NC_1
<input checked="" type="checkbox"/>	<51> := Bitmuster von 32 bit	C_BO_NA_1
<input checked="" type="checkbox"/>	<58> := Einzelbefehl mit Zeitmarke CP56Time2a	C_SC_TA_1
<input checked="" type="checkbox"/>	<59> := Doppelbefehl mit Zeitmarke CP56Time2a	C_DC_TA_1
<input type="checkbox"/>	<60> := Stufenstellbefehl mit Zeitmarke CP56Time2a	C_RC_TA_1
<input checked="" type="checkbox"/>	<61> := Sollwert-Stellbefehl, normierter Wert mit Zeitmarke CP56Time2a	C_SE_TA_1
<input checked="" type="checkbox"/>	<62> := Sollwert-Stellbefehl, skaliertes Wert mit Zeitmarke CP56Time2a	C_SE_TB_1
<input checked="" type="checkbox"/>	<63> := Sollwert-Stellbefehl, verkürzte Gleitkommazahl mit Zeitmarke CP56Time2a	C_SE_TC_1
<input checked="" type="checkbox"/>	<64> := Bitmuster von 32 bit mit Zeitmarke CP56Time2a	C_BO_TA_1

*Tab. 3: Systeminformation in Überwachungsrichtung*

<input checked="" type="checkbox"/>	<70> := Initialisierungsende	M_EI_NA_1
-------------------------------------	------------------------------	-----------

*Tab. 4: Systeminformation in Steuerungsrichtung*

<input checked="" type="checkbox"/>	<100>:= (Stations-)Abfragebefehl	C_IC_NA_1
<input checked="" type="checkbox"/>	<101>:= Zählerabfragebefehl	C_CI_NA_1
<input type="checkbox"/>	<102>:= Abfragebefehl (Lesebefehl)	C_RD_NA_1
<input checked="" type="checkbox"/>	<103>:= Uhrzeit-Synchronisationsbefehl	C_CS_NA_1
<input checked="" type="checkbox"/>	<104>:= Prüfbefehl	C_TS_NA_1
<input type="checkbox"/>	<105>:= Prozess-Rücksetzbefehl	C_RP_NA_1
<input type="checkbox"/>	<106>:= Befehl zur Telegrammlaufzeit-Erfassung	C_CD_NA_1
<input type="checkbox"/>	<107>:=Prüfbefehl mit Zeitstempel CP56Time2a	C_TS_TA_1

Tab. 5: Parameter in Steuerungsrichtung

<input type="checkbox"/>	<110>:= Parameter für Messwerte, normierter Wert	P_ME_NA_1
<input type="checkbox"/>	<111>:= Parameter für Messwerte, skaliertes Wert	P_ME_NB_1
<input type="checkbox"/>	<112>:= Parameter für Messwerte, verkürzte Gleitkommazahl	P_ME_NC_1
<input type="checkbox"/>	<113>:= Parameter für Aktivierung	P_AC_NA_1

Tab. 6: Dateiübermittlung

<input type="checkbox"/>	<120>:= Datei bereit	F_FR_NA_1
<input type="checkbox"/>	<121>:= Abschnitt bereit	F_SR_NA_1
<input type="checkbox"/>	<122>:= Abfrage Dateiverzeichnis, -auswahl, -abfrage, Abschnittsabfrage	F_SC_NA_1
<input type="checkbox"/>	<123>:= Letzter Abschnitt, letztes Segment	F_LS_NA_1
<input type="checkbox"/>	<124>:= Dateibestätigung, Abschnittsbestätigung	F_AF_NA_1
<input type="checkbox"/>	<125>:= Segment	F_SG_NA_1
<input type="checkbox"/>	<126>:= Dateiverzeichnis[leer oder <b>X</b> , nur in Überwachungsrichtung verfügbar (regulär)]	F_DR_TA_1

### Zuweisungen der Übertragungsursachen zu den Typkennungen (stationsbezogene Parameter)

Markierung der Kombination Typkennung / Übertragungsursache mit:

Schattiert	Funktion oder ASDU wird nicht benötigt
Leer	Funktion oder ASDU wird nicht angewendet
<input checked="" type="checkbox"/>	Nur in der Normrichtung angewendet
<input checked="" type="checkbox"/>	Nur in der Umkehrrichtung angewendet
<input checked="" type="checkbox"/>	In beiden Richtungen angewendet

Tab. 7: Übertragungsursachen

Nummer	Übertragungsursache
<0>	Not used
<1>	Periodic, cyclic
<2>	Background scan
<3>	Spontaneous
<4>	Initialized
<5>	Request or requested
<6>	Activation (ACT)
<7>	Activation confirmation (ACTCON)
<8>	Deactivation (DEACT)
<9>	Deactivation confirmation (DEACTCON)
<10>	Activation termination (ACTTERM)
<11>	Return information caused by a remote command
<12>	Return information caused by a local command
<13>	File transfer
<20..36>	General interrogation
<37..41>	Counter interrogation
<44>	Unknown type identification
<45>	Unknown cause of transmission
<46>	Unknown address of ASDU
<47>	Unknown information object address

Typkennung		Übertragungsursache																		
		<1>	<2>	<3>	<4>	<5>	<6>	<7>	<8>	<9>	<10>	<11>	<12>	<13>	<20..36>	<37..41>	<44>	<45>	<46>	<47>
		per/cyc	back	spon	init	req	act	actcon	deact	deactcon	actterm	retrem	retloc	file	inro	reqco	unknowntype	unknowncot	unknowncaa	unknownlOA
<1>	M_S P_N A_1		X	X								X			X					
<2>	M_S P_T A_1			X								X								
<3>	M_D P_N A_1		X	X								X			X					
<4>	M_D P_T A_1			X								X								
<5>	M_S T_N A_1		X	X								X			X					
<6>	M_S T_TA _1			X																
<7>	M_B O_N A_1		X	X											X					
<8>	M_B O_T A_1			X																
<9>	M_M E_N A_1	X	X	X											X					
<10>	M_M E_T A_1			X																
<11>	M_M E_N B_1	X	X	X											X					
<12>	M_M E_T B_1			X																
<13>	M_M E_N C_1	X	X	X											X					
<14>	M_M E_T C_1			X																
<15>	M_IT _NA _1			X												X				
<16>	M_IT _TA _1			X												X				
<17>	M_E P_T A_1			X																
<18>	M_E P_T B_1			X																



Typkennung	Übertragungsursache																
<19> M_E P_T C_1			X														
<20> M_P S_N A_1		X	X						X			X					
<21> M_M E_N D_1	X	X	X									X					
<30> M_S P_T B_1			X						X								
<31> M_D P_T B_1			X						X								
<32> M_S T_TB _1			X						X								
<33> M_B O_T B_1			X														
<34> M_M E_T D_1			X														
<35> M_M E_T E_1			X														
<36> M_M E_TF _1			X														
<37> M_IT _TB _1			X										X				
<38> M_E P_T D_1			X														
<39> M_E P_T E_1			X														
<40> M_E P_TF _1			X														
<45> C_S C_N A_1						X	X	X	X	X				X	X	X	X
<46> C_D C_N A_1						X	X	X	X	X				X	X	X	X
<47> C_R C_N A_1																	
<48> C_S E_N A_1						X	X	X	X	X				X	X	X	X
<49> C_S E_N B_1						X	X	X	X	X				X	X	X	X
<50> C_S E_N C_1						X	X	X	X	X				X	X	X	X
<51> C_B O_N A_1						X	X	X	X	X				X	X	X	X
<58> C_S C_T A_1						X	X	X	X	X				X	X	X	X
<59> C_D C_T A_1						X	X	X	X	X				X	X	X	X

Typkennung	Übertragungsursache																	
<60> C_R C_T A_1																		
<61> C_S E_T A_1						X	X	X	X	X					X	X	X	X
<62> C_S E_T B_1						X	X	X	X	X					X	X	X	X
<63> C_S E_T C_1						X	X	X	X	X					X	X	X	X
<64> C_B O_T A_1						X	X	X	X	X					X	X	X	X
<70> M_EI _NA _1				X														
<100> C_IC _NA _1						X	X	X	X	X					X	X	X	X
<101> C_CI _NA _1						X	X			X					X	X	X	X
<102> C_R D_N A_1																		
<103> C_C S_N A_1			X			X	X								X	X	X	X
<104> C_T S_N A_1						X	X								X	X	X	X
<105> C_R P_N A_1																		
<106> C_C D_N A_1																		
<107> C_T S_T A_1																		
<110> P_M E_N A_1															X	X	X	X
<111> P_M E_N B_1															X	X	X	X
<112> P_M E_N C_1															X	X	X	X
<113> P_A C_N A_1															X	X	X	X
<120> F_F R_N A_1															X	X	X	X
<121> F_S R_N A_1															X	X	X	X
<122> F_S C_N A_1															X	X	X	X
<123> F_LS _NA _1															X	X	X	X
<124> F_AF _NA _1															X	X	X	X

Typkennung		Übertragungsursache																		
<125 >	F_S G_N A_1																X	X	X	X
<126 >	F_D R_T A_1																			

**Grundlegende Anwendungsfunktionen**

**Stationsinitialisierung**

Remote initialization

*Tab. 8: Zyklische Datenübertragung*

Zyklische Datenübertragung

*Tab. 9: Abrufprozedur*

Abrufprozedur

*Tab. 10: Spontane Übertragung*

Spontane Übertragung

**Doppelübertragung von Informationsobjekten mit der Übertragungsursache spontan**

Die folgenden Typkennungen dürfen nacheinander in Folge eines einzigen Zustandswechsels eines Informationsobjekts, übertragen werden. Die einzelnen Adressen der Informationsobjekte, die für die Doppelübertragung vorgesehen sind, werden in einer projektbezogenen Liste festgelegt.

- Einzelmeldung M\_SP\_NA\_1, M\_SP\_TA\_1, M\_SP\_TB\_1 und M\_SP\_NA\_1
- Doppelmeldung M\_DP\_NA\_1, M\_DP\_TA\_1 und M\_DP\_TB\_1
- Stufenstellungsmeldung M\_ST\_NA\_1, M\_ST\_TA\_1 und M\_ST\_TB\_1
- Bitmuster von 32 bit M\_BO\_NA\_1, M\_BO\_TA\_1 und M\_BO\_TB\_1
- Messwert, normierter Wert M\_ME\_NA\_1, M\_ME\_TA\_1, M\_ME\_ND\_1 und M\_ME\_TD\_1
- Messwert, skaliertes Wert M\_ME\_NB\_1, M\_ME\_TB\_1 und M\_ME\_TE\_1
- Messwert, verkürzte Gleitkommazahl M\_ME\_NC\_1, M\_ME\_TC\_1 und M\_ME\_TF\_1

Tab. 11: Stationsabfrage

<input checked="" type="checkbox"/>	Global
<input checked="" type="checkbox"/>	Gruppe 1
<input checked="" type="checkbox"/>	Gruppe 2
<input checked="" type="checkbox"/>	Gruppe 3
<input checked="" type="checkbox"/>	Gruppe 4
<input checked="" type="checkbox"/>	Gruppe 5
<input checked="" type="checkbox"/>	Gruppe 6
<input checked="" type="checkbox"/>	Gruppe 7
<input checked="" type="checkbox"/>	Gruppe 8
<input checked="" type="checkbox"/>	Gruppe 9
<input checked="" type="checkbox"/>	Gruppe 10
<input checked="" type="checkbox"/>	Gruppe 11
<input checked="" type="checkbox"/>	Gruppe 12
<input checked="" type="checkbox"/>	Gruppe 13
<input checked="" type="checkbox"/>	Gruppe 14
<input checked="" type="checkbox"/>	Gruppe 15
<input checked="" type="checkbox"/>	Gruppe 16

Die Zuweisung der Adressen der Informationsobjekte zu jeder einzelnen Gruppe muss in einer getrennten Tabelle festgelegt werden.

Tab. 12: Uhrzeitsynchronisation

<input checked="" type="checkbox"/>	Uhrzeitsynchronisation
-------------------------------------	------------------------

*Tab. 13: Befehlsübertragung*

<input checked="" type="checkbox"/>	Direkte Befehlsübertragung
<input checked="" type="checkbox"/>	Direkte Sollwert-Befehlsübertragung
<input type="checkbox"/>	Befehl .Anwahl und Ausführung
<input type="checkbox"/>	Sollwertbefehl .Anwahl und Ausführung
<input checked="" type="checkbox"/>	C_SE ACTTERM angewendet
<input checked="" type="checkbox"/>	Keine zusätzliche Festlegung
<input type="checkbox"/>	Kurze Befehlsausführungsdauer(Ausführungsdauer durch einen Systemparameter in Unterstation bestimmt)
<input type="checkbox"/>	Lange Befehlsausführungsdauer(Ausführungsdauer durch einen Systemparameter in Unterstation bestimmt)
<input type="checkbox"/>	Dauerbefehl
<input type="checkbox"/>	Überwachung der maximalen Verzögerung von Befehlen und Sollwertbefehlen in Befehlsrichtung
<input type="checkbox"/>	Maximal zulässige Verzögerung von Befehlen und Sollwertbefehlen

*Tab. 14: Übertragung von Zählwerten*

<input checked="" type="checkbox"/>	Zählerabfrage
<input checked="" type="checkbox"/>	Zähler umspeichern ohne Rücksetzen
<input checked="" type="checkbox"/>	Zähler umspeichern mit Rücksetzen
<input checked="" type="checkbox"/>	Zähler rücksetzen
<input checked="" type="checkbox"/>	Allgemeine Zählerabfrage
<input checked="" type="checkbox"/>	Zählerabfrage Gruppe 1
<input checked="" type="checkbox"/>	Zählerabfrage Gruppe 2
<input checked="" type="checkbox"/>	Zählerabfrage Gruppe 3
<input checked="" type="checkbox"/>	Zählerabfrage Gruppe 4
<input checked="" type="checkbox"/>	Modus A: Örtliches Umspeichern mit spontaner Übertragung
<input checked="" type="checkbox"/>	Modus B: Örtliches Umspeichern mit Zählerabfrage
<input type="checkbox"/>	Modus C: Umspeichern und Übertragen durch Zähler-Abfrage bei Umspeichern und Übertragen durch Zählerabfragebefehle
<input checked="" type="checkbox"/>	Modus D: Umspeichern durch Zähler-Abfragebefehl, umgespeicherte Werte werden spontan übertragen

*Tab. 15: Laden eines Parameters*

<input type="checkbox"/>	Schwellenwert
<input type="checkbox"/>	Glättungsfaktor
<input type="checkbox"/>	Unterer Grenzwert für Messwertübertragung
<input type="checkbox"/>	Oberer Grenzwert für Messwertübertragung

**Tab. 16: Parameter für Aktivierung**

Act/deact der zyklischen oder periodischen Übertragung des adressierten Objekts

**Tab. 17: Prüfprozedur**

Prüfprozedur

**Dateiübermittlung**

Dateiübermittlung in Überwachungsrichtung:

- Transparente Datei
- Übermittlung von Ereignisfolgen
- Übermittlung von Störfalldaten aus Schutzeinrichtungen
- Übermittlung von Folgen aufgezeichneter Analogwerte

Dateiübermittlung in Steuerungsrichtung

Transparente Datei

**Tab. 18: Hintergrundabfrage**

Hintergrundabfrage

**Tab. 19: Telegrammlaufzeit-Erfassung**

Telegrammlaufzeit-Erfassung

**Anhang****2.2.1.1 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](http://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**

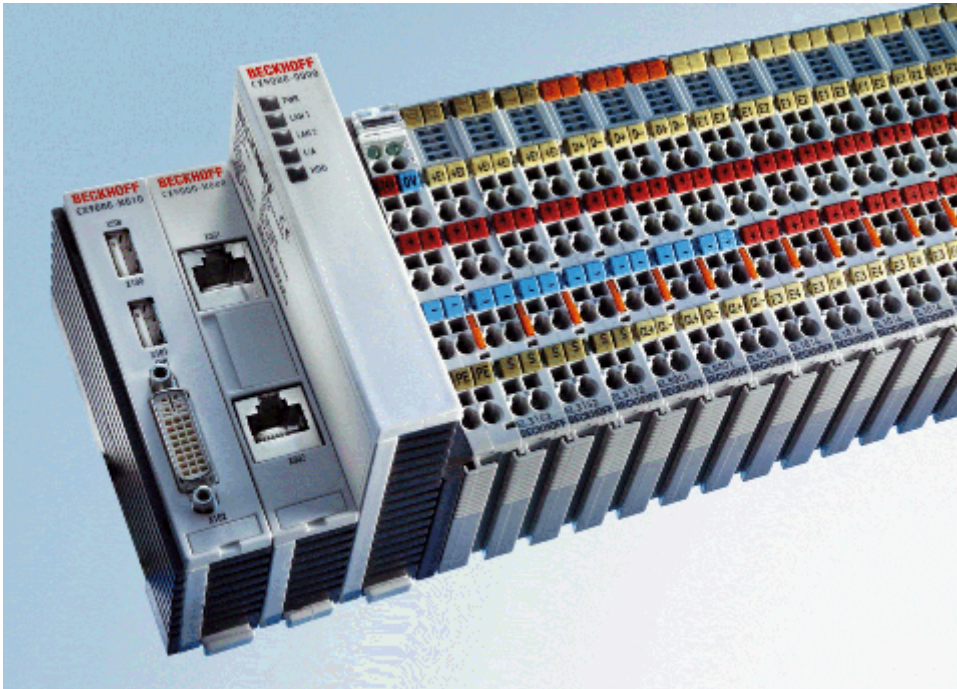
Beckhoff Automation GmbH & Co. KG

Hülshorstweg 20  
33415 Verl  
Deutschland

Telefon: +49 5246 963-0  
E-Mail: info@beckhoff.com  
Internet: [www.beckhoff.com](http://www.beckhoff.com)

## 2.2.2 Kompatibilitätsliste nach IEC 60870-5-101 (slave)

Für die TwinCAT SPS Bibliothek: IEC 60870-5-101 Unterstation (slave).



Version: 1.3

Datum: 07.03.2014

Hier können Sie die Kompatibilitätsliste öffnen/abspeichern (gezipptes PDF): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/984586379.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/984586379.zip)

### Inhaltsverzeichnis:

- [Allgemeine Informationen \[▶ 33\]](#)
  - [Projektinformationen \[▶ 33\]](#)
- [Kompatibilität \[▶ 33\]](#)
  - [System oder Gerät \[▶ 33\]](#)
  - [Netzkonfiguration \[▶ 34\]](#)
  - [Physikalische Schicht \[▶ 34\]](#)
  - [Verbindungsschicht \[▶ 34\]](#)
  - [Anwendungsschicht \[▶ 35\]](#)
  - [Auswahl genormter ASDUs \[▶ 36\]](#)

Prozessinformation in Überwachungsrichtung

Prozessinformation in Steuerungsrichtung

Systeminformation in Überwachungsrichtung

Systeminformation in Steuerungsrichtung

Parameter in Steuerungsrichtung

Dateiübermittlung

- [Grundlegende Anwendungsfunktionen \[▶ 44\]](#)
  - Stationsinitialisierung



- Zyklische Datenübertragung
- Abrufprozedur
- Spontane Übertragung
- Doppelübertragung
- Stationsabfrage
- Uhrzeitsynchronisation
- Befehlsübertragung
- Zählerabfrage
- Laden eines Parameters
- Parameter für Aktivierung
- Prüfprozedur
- Dateiübermittlung
- Hintergrundabfrage
- Telegrammlaufzeiterfassung
- [Anhang \[▶ 47\]](#)
  - [Support und Service \[▶ 47\]](#)

**Allgemeine Informationen**

In dieser Kompatibilitätsliste finden Sie die unterstützten Funktionen innerhalb der IEC 870-5-101.

Die Kompatibilität der einzelnen Komponenten muss von allen Beteiligten für das jeweilige Projekt vereinbart werden.

**Projektinformationen**

Projekt-Nr.:

---

Projekt:

---

Zuständig:

---

Datum:

---

**Kompatibilität**

Die ausgewählten Parameter sollten in den weißen Kontrollfeldern wie folgt markiert werden:

Funktion oder ASDU wird nicht benutzt

Funktion oder ASDU wird wie genormt benutzt (Vorzugswert)

Funktion oder ASDU wird im Umkehrmodus benutzt

Funktion oder ASDU wird im Regel- und Umkehrmodus benutzt

**System oder Gerät**

Systemfestlegung

Festlegungen für die Zentralstation

Festlegungen für die Unterstation (SLAVE)

**Netzkonfiguration**

<input checked="" type="checkbox"/>	End-End-Konfiguration[en: Point-to-point]	<input checked="" type="checkbox"/>	Linienkonfiguration[en: Multipoint-partyline]
<input checked="" type="checkbox"/>	Mehrfach-End-End-Konfiguration[en: Multiple point-to-point]	<input type="checkbox"/>	Sternkonfiguration[en: Multipoint-star]

**Physikalische Schicht**

*Übertragungsgeschwindigkeit:*

<input type="checkbox"/>	100 bit/s	<input checked="" type="checkbox"/>	2400 bit/s	<input type="checkbox"/>	56000 bit/s
<input type="checkbox"/>	200 bit/s	<input checked="" type="checkbox"/>	4800 bit/s	<input type="checkbox"/>	64000 bit/s
<input checked="" type="checkbox"/>	300 bit/s	<input checked="" type="checkbox"/>	9600 bit/s		
<input checked="" type="checkbox"/>	600 bit/s	<input checked="" type="checkbox"/>	19200 bit/s		
<input checked="" type="checkbox"/>	1200 bit/s	<input type="checkbox"/>	38400 bit/s		

**Verbindungsschicht**

Nach dieser Anwendungsbezogenen Norm werden ausschließlich Telegrammformat FT 1.2, Einzelzeichen 1 und das feste Zeitüberwachungsintervall benutzt.

*Adressfeld der Verbindungsschicht:*

<input checked="" type="checkbox"/>	Nicht vorhanden (nur symmetrische Übertragung)
<input checked="" type="checkbox"/>	Ein Oktett
<input checked="" type="checkbox"/>	Zwei Oktette
<input checked="" type="checkbox"/>	Strukturiert
<input checked="" type="checkbox"/>	Unstrukturiert

*Übertragungsprozedur der Verbindungsschicht:*

<input checked="" type="checkbox"/>	Symmetrische Übertragung
<input checked="" type="checkbox"/>	Unsymmetrische Übertragung

*Telegrammlänge:*

Maximale Länge L            255

Wird unsymmetrisch übertragen, werden die folgenden ASDUs als Anwenderdaten mit den angegebenen Übertragungsursachen mit der Datenklasse 2 (niedrige Priorität) zurückgesendet.

<input checked="" type="checkbox"/>	Die genormte Zuweisung von ASDU zur Datenklasse 2 wird wie folgt angewendet:						
	<table border="0"> <tr> <td style="text-align: center;"><b>Typkennung</b></td> <td style="text-align: center;"><b>Übertragungsursache</b></td> </tr> <tr> <td style="text-align: center;">9, 11, 13, 21</td> <td style="text-align: center;">&lt;1&gt;</td> </tr> <tr> <td style="text-align: center;">1, 3, 5, 7, 9, 11, 13, 20, 21</td> <td style="text-align: center;">&lt;2&gt;</td> </tr> </table>	<b>Typkennung</b>	<b>Übertragungsursache</b>	9, 11, 13, 21	<1>	1, 3, 5, 7, 9, 11, 13, 20, 21	<2>
<b>Typkennung</b>	<b>Übertragungsursache</b>						
9, 11, 13, 21	<1>						
1, 3, 5, 7, 9, 11, 13, 20, 21	<2>						
<input type="checkbox"/>	Eine spezielle Zuweisung von ASDU zur Datenklasse 2 wird wie folgt angewendet:						

**Typkennung**

**Übertragungsursache**



Als Antwort auf eine Anforderung nach Daten der Klasse 2 darf eine Unterstation Daten der Datenklasse 1 übertragen, wenn keine Daten der Datenklasse 2 vorhanden sind (konfigurierbar).

**Anwendungsschicht**

**Übertragungsbetriebsart für Anwendungsdaten**

Nach dieser Anwendungsbezogenen Norm wird ausschließlich Betriebsart 1 (niedrigstwertiges Oktett zuerst) nach 4.10 von IEC 60870-5-4 benutzt.

**Gemeinsame Adresse der ASDU**

<input checked="" type="checkbox"/>	Ein Oktett	<input checked="" type="checkbox"/>	Zwei Oktette
-------------------------------------	------------	-------------------------------------	--------------

**Adresse des Informationsobjekts**

<input checked="" type="checkbox"/>	Ein Oktett	<input checked="" type="checkbox"/>	Strukturiert
<input checked="" type="checkbox"/>	Zwei Oktette	<input checked="" type="checkbox"/>	Unstrukturiert
<input checked="" type="checkbox"/>	Drei Oktette		

**Übertragungsursache**

<input checked="" type="checkbox"/>	Ein Oktett	<input checked="" type="checkbox"/>	Zwei Oktette(mit Herkunftsadresse). Mit 0 vorbesetzt, falls Herkunftsadresse nicht vorhanden.
-------------------------------------	------------	-------------------------------------	--

**Auswahl genormter ASDU**

*Tab. 20: Prozessinformation in Überwachungsrichtung*

<input checked="" type="checkbox"/>	<1> := Einzelmeldung	M_SP_NA_1
<input checked="" type="checkbox"/>	<2> := Einzelmeldung mit Zeitmarke	M_SP_TA_1
<input checked="" type="checkbox"/>	<3> := Doppelmeldung	M_DP_NA_1
<input checked="" type="checkbox"/>	<4> := Doppelmeldung mit Zeitmarke	M_DP_TA_1
<input checked="" type="checkbox"/>	<5> := Stufenstellungsmeldung	M_ST_NA_1
<input checked="" type="checkbox"/>	<6> := Stufenstellungsmeldung mit Zeitmarke	M_ST_TA_1
<input checked="" type="checkbox"/>	<7> := Bitmuster von 32 bit	M_BO_NA_1
<input checked="" type="checkbox"/>	<8> := Bitmuster von 32 bit mit Zeitmarke	M_BO_TA_1
<input checked="" type="checkbox"/>	<9> := Messwert, normierter Wert	M_ME_NA_1
<input checked="" type="checkbox"/>	<10> := Messwert, normierter Wert mit Zeitmarke	M_ME_TA_1
<input checked="" type="checkbox"/>	<11> := Messwert, skaliertes Wert	M_ME_NB_1
<input checked="" type="checkbox"/>	<12> := Messwert, skaliertes Wert mit Zeitmarke	M_ME_TB_1
<input checked="" type="checkbox"/>	<13> := Messwert, verkürzte Gleitkommazahl	M_ME_NC_1
<input checked="" type="checkbox"/>	<14> := Messwert, verkürzte Gleitkommazahl mit Zeitmarke	M_ME_TC_1
<input checked="" type="checkbox"/>	<15> := Zählwerte	M_IT_NA_1
<input checked="" type="checkbox"/>	<16> := Zählwerte mit Zeitmarke	M_IT_TA_1
<input checked="" type="checkbox"/>	<17> := Schutzereignis mit Zeitmarke	M_EP_TA_1
<input checked="" type="checkbox"/>	<18> := Geblockte Anregungen des Schutzes mit Zeitmarke	M_EP_TB_1
<input checked="" type="checkbox"/>	<19> := Geblockte Auslösungen des Schutzes mit Zeitmarke	M_EP_TC_1
<input checked="" type="checkbox"/>	<20> := Geblockte Einzelmeldungen mit Zustandsanzeige	M_PS_NA_1
<input checked="" type="checkbox"/>	<21> := Messwert, normierter Wert ohne Qualitätskennung	M_ME_ND_1
<input checked="" type="checkbox"/>	<30> := Einzelmeldung mit Zeitmarke CP56Time2a	M_SP_TB_1
<input checked="" type="checkbox"/>	<31> := Doppelmeldung mit Zeitmarke CP56Time2a	M_DP_TB_1
<input checked="" type="checkbox"/>	<32> := Stufenstellungsmeldung mit Zeitmarke CP56Time2a	M_ST_TB_1
<input checked="" type="checkbox"/>	<33> := Bitmuster von 32 bit mit Zeitmarke CP56Time2a	M_BO_TB_1
<input checked="" type="checkbox"/>	<34> := Messwert, normierter Wert mit Zeitmarke CP56Time2a	M_ME_TD_1
<input checked="" type="checkbox"/>	<35> := Messwert, skaliertes Wert mit Zeitmarke CP56Time2a	M_ME_TE_1
<input checked="" type="checkbox"/>	<36> := Messwert, verkürzte Gleitkommazahl mit Zeitmarke CP56Time2a	M_ME_TF_1
<input checked="" type="checkbox"/>	<37> := Zählwerte mit Zeitmarke CP56Time2a	M_IT_TB_1
<input checked="" type="checkbox"/>	<38> := Schutzereignis mit Zeitmarke CP56Time2a	M_EP_TD_1
<input checked="" type="checkbox"/>	<39> := Geblockte Anregungen des Schutzes mit Zeitmarke CP56Time2a	M_EP_TE_1

<input checked="" type="checkbox"/>	<40> := Geblockte Auslösungen des Schutzes mit Zeitmarke CP56Time2a	M_EP_TF_1
-------------------------------------	---	-----------

Tab. 21: Prozessinformation in Steuerungsrichtung

<input checked="" type="checkbox"/>	<45> := Einzelbefehl	C_SC_NA_1
<input checked="" type="checkbox"/>	<46> := Doppelbefehl	C_DC_NA_1
<input checked="" type="checkbox"/>	<47> := Stufenstellbefehl	C_RC_NA_1
<input checked="" type="checkbox"/>	<48> := Sollwert-Stellbefehl, normierter Wert	C_SE_NA_1
<input checked="" type="checkbox"/>	<49> := Sollwert-Stellbefehl, skaliertes Wert	C_SE_NB_1
<input checked="" type="checkbox"/>	<50> := Sollwert-Stellbefehl, verkürzte Gleitkommazahl	C_SE_NC_1
<input checked="" type="checkbox"/>	<51> := Bitmuster von 32 bit	C_BO_NA_1
<input checked="" type="checkbox"/>	<58> := Einzelbefehl mit Zeitmarke CP56Time2a	C_SC_TA_1
<input checked="" type="checkbox"/>	<59> := Doppelbefehl mit Zeitmarke CP56Time2a	C_DC_TA_1
<input checked="" type="checkbox"/>	<60> := Stufenstellbefehl mit Zeitmarke CP56Time2a	C_RC_TA_1
<input checked="" type="checkbox"/>	<61> := Sollwert-Stellbefehl, normierter Wert mit Zeitmarke CP56Time2a	C_SE_TA_1
<input checked="" type="checkbox"/>	<62> := Sollwert-Stellbefehl, skaliertes Wert mit Zeitmarke CP56Time2a	C_SE_TB_1
<input checked="" type="checkbox"/>	<63> := Sollwert-Stellbefehl, verkürzte Gleitkommazahl mit Zeitmarke CP56Time2a	C_SE_TC_1
<input checked="" type="checkbox"/>	<64> := Bitmuster von 32 bit mit Zeitmarke CP56Time2a	C_BO_TA_1

Tab. 22: Systeminformation in Überwachungsrichtung

<input checked="" type="checkbox"/>	<70> := Initialisierungsende	M_EI_NA_1
-------------------------------------	------------------------------	-----------

Tab. 23: Systeminformation in Steuerungsrichtung

<input checked="" type="checkbox"/>	<100>:= (Stations-)Abfragebefehl	C_IC_NA_1
<input checked="" type="checkbox"/>	<101>:= Zählerabfragebefehl	C_CI_NA_1
<input checked="" type="checkbox"/>	<102>:= Abfragebefehl (Lesebefehl)	C_RD_NA_1
<input checked="" type="checkbox"/>	<103>:= Uhrzeit-Synchronisationsbefehl	C_CS_NA_1
<input checked="" type="checkbox"/>	<104>:= Prüfbefehl	C_TS_NA_1
<input checked="" type="checkbox"/>	<105>:= Prozess-Rücksetzbefehl	C_RP_NA_1
<input type="checkbox"/>	<106>:= Befehl zur Telegrammlaufzeit-Erfassung	C_CD_NA_1
<input checked="" type="checkbox"/>	<107>:= Test command with time tagCP56Time2a	C_TS_TA_1

*Tab. 24: Parameter in Steuerungsrichtung*

<input type="checkbox"/>	<110>:= Parameter für Messwerte, normierter Wert	P_ME_NA_1
<input type="checkbox"/>	<111>:= Parameter für Messwerte, skaliertes Wert	P_ME_NB_1
<input type="checkbox"/>	<112>:= Parameter für Messwerte, verkürzte Gleitkommazahl	P_ME_NC_1
<input type="checkbox"/>	<113>:= Parameter für Aktivierung	P_AC_NA_1

*Tab. 25: Dateiübermittlung*

<input type="checkbox"/>	<120>:= Datei bereit	F_FR_NA_1
<input type="checkbox"/>	<121>:= Abschnitt bereit	F_SR_NA_1
<input type="checkbox"/>	<122>:= Abfrage Dateiverzeichnis, -auswahl, -abfrage, Abschnittsabfrage	F_SC_NA_1
<input type="checkbox"/>	<123>:= Letzter Abschnitt, letztes Segment	F_LS_NA_1
<input type="checkbox"/>	<124>:= Dateibestätigung, Abschnittsbestätigung	F_AF_NA_1
<input type="checkbox"/>	<125>:= Segment	F_SG_NA_1
<input type="checkbox"/>	<126>:= Dateiverzeichnis[leer oder <b>X</b> , nur in Überwachungsrichtung verfügbar (regulär)]	F_DR_TA_1

**Zuweisungen der Übertragungsursachen zu den Typkennungen(stationsbezogene Parameter)**

Markierung der Kombination Typkennung / Übertragungsursache mit:

Schattiert	Funktion oder ASDU wird nicht benötigt
Leer	Funktion oder ASDU wird nicht angewendet
<input checked="" type="checkbox"/>	Nur in der Normrichtung angewendet
<input checked="" type="checkbox"/>	Nur in der Umkehrrichtung angewendet
<input checked="" type="checkbox"/>	In beiden Richtungen angewendet

Tab. 26: Übertragungsursachen

Nummer	Übertragungsursache
<0>	Not used
<1>	Periodic, cyclic
<2>	Background scan
<3>	Spontaneous
<4>	Initialized
<5>	Request or requested
<6>	Activation (ACT)
<7>	Activation confirmation (ACTCON)
<8>	Deactivation (DEACT)
<9>	Deactivation confirmation (DEACTCON)
<10>	Activation termination (ACTTERM)
<11>	Return information caused by a remote command
<12>	Return information caused by a local command
<13>	File transfer
<20..36>	General interrogation
<37..41>	Counter interrogation
<44>	Unknown type identification
<45>	Unknown cause of transmission
<46>	Unknown address of ASDU
<47>	Unknown information object address



Typkennung		Übertragungsursache																		
		<1>	<2>	<3>	<4>	<5>	<6>	<7>	<8>	<9>	<10>	<11>	<12>	<13>	<20..36>	<37..41>	<44>	<45>	<46>	<47>
		per/cyc	back	sponnt	init	req	act	actcon	deact	deactcon	actterm	retrem	retloc	file	inro	reqco	unknownntype	unknownCOT	unknownCAA	unknownIOA
<1>	M_S P_N A_1		X	X		X						X			X					
<2>	M_S P_T A_1			X		X						X								
<3>	M_D P_N A_1		X	X		X						X			X					
<4>	M_D P_T A_1			X		X						X								
<5>	M_S T_N A_1		X	X		X						X			X					
<6>	M_S T_T A_1			X		X														
<7>	M_B O_N A_1		X	X		X									X					
<8>	M_B O_T A_1			X		X														
<9>	M_M E_N A_1	X	X	X		X									X					
<10>	M_M E_T A_1			X		X														
<11>	M_M E_N B_1	X	X	X		X									X					
<12>	M_M E_T B_1			X		X														
<13>	M_M E_N C_1	X	X	X		X									X					
<14>	M_M E_T C_1			X		X														
<15>	M_IT _NA _1			X												X				
<16>	M_IT _TA _1			X												X				
<17>	M_E P_T A_1			X																
<18>	M_E P_T B_1			X																

Typkennung		Übertragungsursache																	
<19>	M_E P_T C_1			X															
<20>	M_P S_N A_1		X	X		X					X			X					
<21>	M_M E_N D_1	X	X	X		X								X					
<30>	M_S P_T B_1			X		X					X								
<31>	M_D P_T B_1			X		X					X								
<32>	M_S T_T B_1			X		X					X								
<33>	M_B O_T B_1			X		X													
<34>	M_M E_T D_1			X		X													
<35>	M_M E_T E_1			X		X													
<36>	M_M E_T F_1			X															
<37>	M_IT _TB_ _1			X											X				
<38>	M_E P_T D_1			X															
<39>	M_E P_T E_1			X															
<40>	M_E P_T F_1			X															
<45>	C_S C_N A_1						X	X	X	X	X					X	X	X	X
<46>	C_D C_N A_1						X	X	X	X	X					X	X	X	X
<47>	C_R C_N A_1						X	X	X	X	X					X	X	X	X
<48>	C_S E_N A_1						X	X	X	X	X					X	X	X	X
<49>	C_S E_N B_1						X	X	X	X	X					X	X	X	X
<50>	C_S E_N C_1						X	X	X	X	X					X	X	X	X
<51>	C_B O_N A_1						X	X	X	X	X					X	X	X	X
<58>	C_S C_T A_1						X	X	X	X	X					X	X	X	X
<59>	C_D C_T A_1						X	X	X	X	X					X	X	X	X

Typkennung		Übertragungsursache																	
<60>	C_R C_T A_1						X	X	X	X	X					X	X	X	X
<61>	C_S E_T A_1						X	X	X	X	X					X	X	X	X
<62>	C_S E_T B_1						X	X	X	X	X					X	X	X	X
<63>	C_S E_T C_1						X	X	X	X	X					X	X	X	X
<64>	C_B O_T A_1						X	X	X	X	X					X	X	X	X
<70>	M_EI _NA _1				X														
<100>	C_IC _NA _1						X	X	X	X	X					X	X	X	X
<101>	C_CI _NA _1						X	X			X					X	X	X	X
<102>	C_R D_N A_1				X											X	X	X	X
<103>	C_C S_N A_1		X				X	X								X	X	X	X
<104>	C_T S_N A_1						X	X								X	X	X	X
<105>	C_R P_N A_1						X	X								X	X	X	X
<106>	C_C D_N A_1															X	X	X	X
<107>	C_T S_T A_1						X	X								X	X	X	X
<110>	P_M E_N A_1															X	X	X	X
<111>	P_M E_N B_1															X	X	X	X
<112>	P_M E_N C_1															X	X	X	X
<113>	P_A C_N A_1															X	X	X	X
<120>	F_F R_N A_1															X	X	X	X
<121>	F_S R_N A_1															X	X	X	X
<122>	F_S C_N A_1															X	X	X	X
<123>	F_LS _NA _1															X	X	X	X
<124>	F_A F_N A_1															X	X	X	X

Typkennung		Übertragungsursache																		
<125>	F_S G_N A_1																X	X	X	X
<126>	F_D R_T A_1																			

**Grundlegende Anwendungsfunktionen**

**Stationsinitialisierung**

Remote initialization

*Tab. 27: Zyklische Datenübertragung*

Zyklische Datenübertragung

*Tab. 28: Abrufprozedur*

Abrufprozedur

*Tab. 29: Spontane Übertragung*

Spontane Übertragung

**Doppelübertragung von Informationsobjekten mit der Übertragungsursache spontan**

Die folgenden Typkennungen dürfen nacheinander in Folge eines einzigen Zustandswechsels eines Informationsobjekts, übertragen werden. Die einzelnen Adressen der Informationsobjekte, die für die Doppelübertragung vorgesehen sind, werden in einer projektbezogenen Liste festgelegt.

- Einzelmeldung M\_SP\_NA\_1, M\_SP\_TA\_1, M\_SP\_TB\_1 und M\_SP\_NA\_1
- Doppelmeldung M\_DP\_NA\_1, M\_DP\_TA\_1 und M\_DP\_TB\_1
- Stufenstellungsmeldung M\_ST\_NA\_1, M\_ST\_TA\_1 und M\_ST\_TB\_1
- Bitmuster von 32 bit M\_BO\_NA\_1, M\_BO\_TA\_1 und M\_BO\_TB\_1
- Messwert, normierter Wert M\_ME\_NA\_1, M\_ME\_TA\_1, M\_ME\_ND\_1 und M\_ME\_TD\_1
- Messwert, skaliertes Wert M\_ME\_NB\_1, M\_ME\_TB\_1 und M\_ME\_TE\_1
- Messwert, verkürzte Gleitkommazahl M\_ME\_NC\_1, M\_ME\_TC\_1 und M\_ME\_TF\_1

Tab. 30: Stationsabfrage

<input checked="" type="checkbox"/>	Global
<input checked="" type="checkbox"/>	Gruppe 1
<input checked="" type="checkbox"/>	Gruppe 2
<input checked="" type="checkbox"/>	Gruppe 3
<input checked="" type="checkbox"/>	Gruppe 4
<input checked="" type="checkbox"/>	Gruppe 5
<input checked="" type="checkbox"/>	Gruppe 6
<input checked="" type="checkbox"/>	Gruppe 7
<input checked="" type="checkbox"/>	Gruppe 8
<input checked="" type="checkbox"/>	Gruppe 9
<input checked="" type="checkbox"/>	Gruppe 10
<input checked="" type="checkbox"/>	Gruppe 11
<input checked="" type="checkbox"/>	Gruppe 12
<input checked="" type="checkbox"/>	Gruppe 13
<input checked="" type="checkbox"/>	Gruppe 14
<input checked="" type="checkbox"/>	Gruppe 15
<input checked="" type="checkbox"/>	Gruppe 16

Die Zuweisung der Adressen der Informationsobjekte zu jeder einzelnen Gruppe muss in einer getrennten Tabelle festgelegt werden.

Tab. 31: Uhrzeitsynchronisation

<input checked="" type="checkbox"/>	Uhrzeitsynchronisation
-------------------------------------	------------------------

Tab. 32: Befehlsübertragung

<input checked="" type="checkbox"/>	Direkte Befehlsübertragung
<input checked="" type="checkbox"/>	Direkte Sollwert-Befehlsübertragung
<input checked="" type="checkbox"/>	Befehl .Anwahl und Ausführung
<input checked="" type="checkbox"/>	Sollwertbefehl .Anwahl und Ausführung
<input checked="" type="checkbox"/>	C_SE ACTTERM angewendet
<input checked="" type="checkbox"/>	Keine zusätzliche Festlegung
<input type="checkbox"/>	Kurze Befehlsausführungsdauer(Ausführungsdauer durch einen Systemparameter in Unterstation bestimmt)
<input type="checkbox"/>	Lange Befehlsausführungsdauer(Ausführungsdauer durch einen Systemparameter in Unterstation bestimmt)
<input type="checkbox"/>	Dauerbefehl
<input type="checkbox"/>	Überwachung der maximalen Verzögerung von Befehlen und Sollwertbefehlen in Befehlsrichtung
<input type="checkbox"/>	Maximal zulässige Verzögerung von Befehlen und Sollwertbefehlen

Tab. 33: Übertragung von Zählwerten

<input checked="" type="checkbox"/>	Zählerabfrage
<input checked="" type="checkbox"/>	Zähler umspeichern ohne Rücksetzen
<input checked="" type="checkbox"/>	Zähler umspeichern mit Rücksetzen
<input checked="" type="checkbox"/>	Zähler rücksetzen
<input checked="" type="checkbox"/>	Allgemeine Zählerabfrage
<input checked="" type="checkbox"/>	Zählerabfrage Gruppe 1
<input checked="" type="checkbox"/>	Zählerabfrage Gruppe 2
<input checked="" type="checkbox"/>	Zählerabfrage Gruppe 3
<input checked="" type="checkbox"/>	Zählerabfrage Gruppe 4
<input checked="" type="checkbox"/>	Modus A: Örtliches Umspeichern mit spontaner Übertragung
<input checked="" type="checkbox"/>	Modus B: Örtliches Umspeichern mit Zählerabfrage
<input checked="" type="checkbox"/>	Modus C: Umspeichern und Übertragen durch Zähler-Abfrage bei Umspeichern und Übertragen durch Zählerabfragebefehle
<input checked="" type="checkbox"/>	Modus D: Umspeichern durch Zähler-Abfragebefehl, umgespeicherte Werte werden spontan übertragen

Tab. 34: Laden eines Parameters

<input type="checkbox"/>	Schwellenwert
<input type="checkbox"/>	Glättungsfaktor
<input type="checkbox"/>	Unterer Grenzwert für Messwertübertragung
<input type="checkbox"/>	Oberer Grenzwert für Messwertübertragung

Tab. 35: Parameter für Aktivierung

Act/deact der zyklischen oder periodischen Übertragung des adressierten Objekts

Tab. 36: Prüfprozedur

Prüfprozedur

### Dateiübermittlung

Dateiübermittlung in Überwachungsrichtung:

- Transparente Datei
- Übermittlung von Ereignisfolgen
- Übermittlung von Störfalldaten aus Schutzeinrichtungen
- Übermittlung von Folgen aufgezeichneter Analogwerte

Dateiübermittlung in Steuerungsrichtung

Transparente Datei

Tab. 37: Hintergrundabfrage

Hintergrundabfrage

Tab. 38: Telegrammlaufzeit-Erfassung

Telegrammlaufzeit-Erfassung

## Anhang

### 2.2.2.1 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](http://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

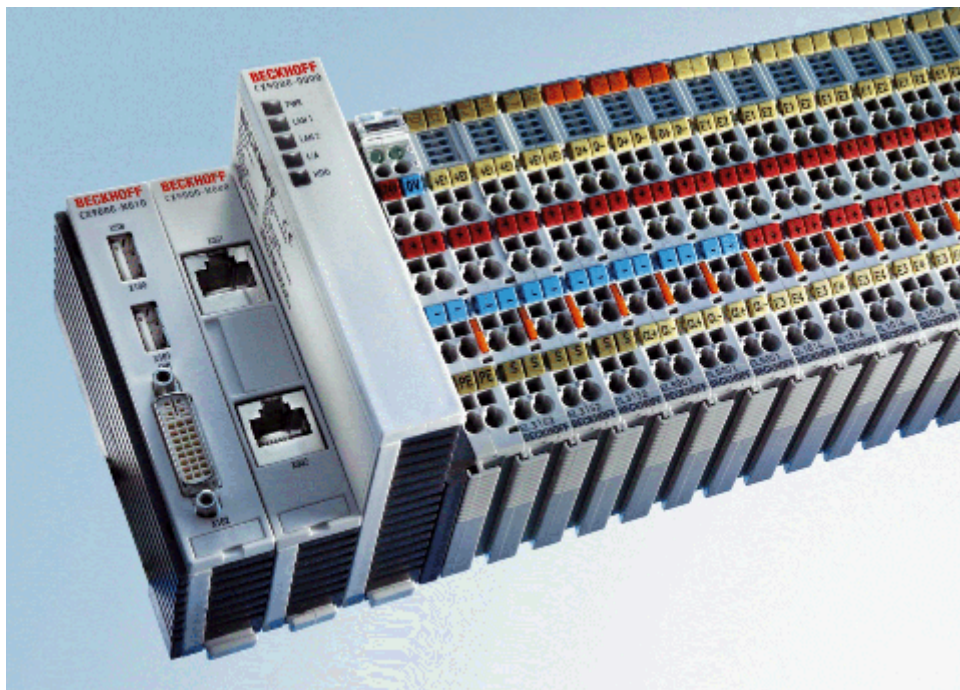
Hülshorstweg 20  
33415 Verl  
Deutschland

Telefon: +49 5246 963-0  
E-Mail: info@beckhoff.com  
Internet: [www.beckhoff.com](http://www.beckhoff.com)



## 2.2.3 Kompatibilitätsliste nach IEC 60870-5-102 (master)

für TwinCAT SPS Bibliothek: IEC 60870-5-102 Zentralstation (master).



Version: 1.2

Datum: 07.03.2014

Hier können Sie die Kompatibilitätsliste öffnen/abspeichern (gezipptes PDF): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/984589323.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/984589323.zip)

### Inhaltsverzeichnis:

- [Allgemeine Informationen \[▶ 50\]](#)
  - [Projektinformationen \[▶ 50\]](#)
- [Kompatibilität \[▶ 50\]](#)
  - [System oder Gerät \[▶ 50\]](#)
  - [Netzkonfiguration \[▶ 50\]](#)
  - [Physikalische Schicht \[▶ 50\]](#)
  - [Verbindungsschicht \[▶ 51\]](#)
  - [Anwendungsschicht \[▶ 51\]](#)
  - [Auswahl genormter ASDUs \[▶ 53\]](#)

Prozessinformation in Überwachungsrichtung (slave->master)

Systeminformation in Überwachungsrichtung (slave->master)

Systeminformation in Steuerungsrichtung (master->slave)

- [Grundlegende Anwendungsfunktionen \[▶ 54\]](#)
  - [Übertragung von Zählwerten \[▶ 54\]](#)
  - [Abruf von Listen \[▶ 55\]](#)
  - [Übertragung von Einzelmeldungen \[▶ 55\]](#)
- [Anhang \[▶ 55\]](#)

- [Support und Service \[► 55\]](#)

## Allgemeine Informationen

In dieser Kompatibilitätsliste finden Sie die unterstützten Funktionen innerhalb der IEC 870-5-102.

Die Kompatibilität der einzelnen Komponenten muss von allen Beteiligten für das jeweilige Projekt vereinbart werden.

## Projektinformationen

Projekt-Nr.:

---

Projekt:

---

Zuständig:

---

Datum:

---

## Kompatibilität

Aus den vorstehend festgelegten Parametersätzen und Alternativen sind Untermengen auszuwählen, um bestimmte Systeme für die Übertragung von Zählerständen zu erstellen.

Bestimmte Parameter wie die Anzahl der Oktette in der GEMEINSAMEN ADRESSE DER ASDU, schließen sich gegenseitig aus. Das bedeutet, dass nur ein Wert des festgelegten Parameters je System zulässig ist.

Andere Parameter, wie der aufgelistete Satz unterschiedlicher Information in Steuerungs- und Überwachungsrichtung, erlauben die Festlegung des Gesamtumfangs oder von Untermengen, die für die gegebene Anwendung geeignet sind.

Die ausgewählten Parameter sollten in den weißen Kontrollfeldern wie folgt markiert werden:

Funktion oder ASDU wird nicht benutzt

Funktion oder ASDU wird wie genormt benutzt (Vorzugswert)

## System oder Gerät

Systemfestlegung

Festlegungen für die Zentralstation

Festlegungen für die Unterstation

## Netzkonfiguration

End-End-Konfiguration[en:  
Point-to-point]

Linienkonfiguration[en:  
Multipoint-partyline]

Mehrfach-End-End-  
Konfiguration[en: Multiple  
point-to-point]

Sternkonfiguration[en:  
Multipoint-star]

## Physikalische Schicht

*Übertragungsgeschwindigkeit:*

<input type="checkbox"/>	100 bit/s	<input checked="" type="checkbox"/>	2400 bit/s	<input type="checkbox"/>	56000 bit/s
<input type="checkbox"/>	200 bit/s	<input checked="" type="checkbox"/>	4800 bit/s	<input type="checkbox"/>	64000 bit/s
<input checked="" type="checkbox"/>	300 bit/s	<input checked="" type="checkbox"/>	9600 bit/s		
<input checked="" type="checkbox"/>	600 bit/s	<input checked="" type="checkbox"/>	19200 bit/s		
<input checked="" type="checkbox"/>	1200 bit/s	<input type="checkbox"/>	38400 bit/s		

**Verbindungsschicht**

Nach dieser anwendungsbezogenen Norm werden ausschließlich Telegrammformat FT 1.2, Einzelzeichen 1 und das feste Zeitüberwachungsintervall benutzt.

*Adressfeld der Verbindungsschicht:*

<input checked="" type="checkbox"/>	Nicht vorhanden (nur symmetrische Übertragung)
<input checked="" type="checkbox"/>	Ein Oktett
<input checked="" type="checkbox"/>	Zwei Oktette
<input checked="" type="checkbox"/>	Strukturiert
<input checked="" type="checkbox"/>	Unstrukturiert

*Übertragungsprozedur der Verbindungsschicht:*

<input type="checkbox"/>	Symmetrische Übertragung
<input checked="" type="checkbox"/>	Unsymmetrische Übertragung

*Telegrammlänge:*

Maximale Länge L: 255

*Dienstfunktionen der Verbindungsschicht:*

<input type="checkbox"/>	Rücksetzen des Anwendungsprozesses nicht vorhanden
--------------------------	--

**Anwendungsschicht**

**Übertragungsbetriebsart für Anwendungsdaten**

Nach dieser Anwendungsbezogenen Norm wird ausschließlich Betriebsart 1 (niedrigstwertiges Oktett zuerst) nach 4.10 von IEC 60870-5-4 benutzt.

**Herstellerkodierung**

(Herstellerbezogener Parameter)

Systemspezifische Kodierungen für verschiedene Hersteller

Nummer	Hersteller
1	
2	
...	
255	

**Adresse der Zähler-DEE (gemeinsame ASDU adresse)**

(Systembezogener Parameter)

Ein Oktett

Zwei Oktette

**Signatur**

(Systembezogener Parameter)

Signatur

Keine Signatur

**Listenadresse**

(Systembezogener Parameter)

Grundwerte

Periode 1

Periodenzeit:

Periode 2

Periodenzeit:

Periode 3

Periodenzeit:

Tageswerte

Monatswerte

Jahreswerte

Periode 1

Periode 1

Periode 1

Periode 2

Periode 2

Periode 2

Periode 3

Periode 3

Periode 3

Zählerstände vom Beginn der Abrechnungsperiode

Monatliche Abrechnungsperiode

Jährliche Abrechnungsperiode

Vierteljährliche Abrechnungsperiode (alle Drei Monate)

Älteste Einzelmeldung

Gesamtliste mit Einzelmeldungen

Teilliste 1 mit Einzelmeldungen

Teilliste 2 mit Einzelmeldungen

Teilliste 3 mit Einzelmeldungen

Teilliste 4 mit Einzelmeldungen

**Tarifinformation**

(Systembezogener Parameter)

Tarifinformation

Keine Tarifinformation

**Auswahl genormter ASDU**

*Tab. 39: Prozessinformation in Überwachungsrichtung*

<input checked="" type="checkbox"/>	<1> := Einzelmeldung mit Zeitmarke	M_SP_TA_2
<input checked="" type="checkbox"/>	<2> := Abrechnungszählerstände, je vier Oktette	M_IT_TA_2
<input checked="" type="checkbox"/>	<3> := Abrechnungszählerstände, je drei Oktette	M_IT_TB_2
<input checked="" type="checkbox"/>	<4> := Abrechnungszählerstände, je zwei Oktette	M_IT_TC_2
<input checked="" type="checkbox"/>	<5> := Periodisch rückgesetzte Abrechnungszählerstände, je vier Oktette	M_IT_TD_2
<input checked="" type="checkbox"/>	<6> := Periodisch rückgesetzte Abrechnungszählerstände, je drei Oktette	M_IT_TE_2
<input checked="" type="checkbox"/>	<7> := Periodisch rückgesetzte Abrechnungszählerstände, je zwei Oktette	M_IT_TF_2
<input checked="" type="checkbox"/>	<8> := Betriebszählerstände, je vier Oktette	M_IT_TG_2
<input checked="" type="checkbox"/>	<9> := Betriebszählerstände, je drei Oktette	M_IT_TH_2
<input checked="" type="checkbox"/>	<10> := Betriebszählerstände, je zwei Oktette	M_IT_TI_2
<input checked="" type="checkbox"/>	<11> := Periodisch rückgesetzte Betriebszählerstände, je vier Oktette	M_IT_TK_2
<input checked="" type="checkbox"/>	<12> := Periodisch rückgesetzte Betriebszählerstände, je drei Oktette	M_IT_TL_2
<input checked="" type="checkbox"/>	<13> := Periodisch rückgesetzte Betriebszählerstände, je zwei Oktette	M_IT_TM_2

*Tab. 40: Systeminformation in Überwachungsrichtung*

<input checked="" type="checkbox"/>	<70> := Initialisierungsende	M_EI_NA_2
<input checked="" type="checkbox"/>	<71> := Hersteller- und Produktspezifikation der Zähler-DEE	P_MP_NA_2
<input checked="" type="checkbox"/>	<72> := Aktuelle Systemzeit der Zähler-DEE	M_TI_TA_2

Tab. 41: Systeminformation in Steuerungsrichtung

<input checked="" type="checkbox"/>	<100>:= Abruf der Hersteller- und Produktspezifikation	C_RD_NA_2
<input checked="" type="checkbox"/>	<101>:= Abruf einer Liste von Einzelmeldungen mit Zeitmarke	C_SP_NA_2
<input checked="" type="checkbox"/>	<102>:= Abruf einer Liste von Einzelmeldungen mit Zeitmarke eines ausgewählten Zeitbereichs	C_SP_NB_2
<input checked="" type="checkbox"/>	<103>:= Abruf der aktuellen Systemzeit der Zähler-DEE	C_TI_NA_2
<input checked="" type="checkbox"/>	<104>:= Abruf der Abrechnungszählerstände der ältesten Messperiode	C_CI_NA_2
<input checked="" type="checkbox"/>	<105>:= Abruf der Abrechnungszählerstände der ältesten Messperiode und eines ausgewählten Adressbereichs	C_CI_NB_2
<input checked="" type="checkbox"/>	<106>:= Abruf der Abrechnungszählerstände einer bestimmten vergangenen Messperiode	C_CI_NC_2
<input checked="" type="checkbox"/>	<107>:= Abruf der Abrechnungszählerstände einer bestimmten vergangenen Messperiode und eines ausgewählten Adressbereichs	C_CI_ND_2
<input checked="" type="checkbox"/>	<108>:= Abruf periodisch rückgesetzter Abrechnungszählerstände der ältesten Messperiode	C_CI_NE_2
<input checked="" type="checkbox"/>	<109>:= Abruf periodisch rückgesetzter Abrechnungszählerstände der ältesten Messperiode und eines ausgewählten Adressbereichs	C_CI_NF_2
<input checked="" type="checkbox"/>	<110>:= Abruf periodisch rückgesetzter Abrechnungszählerstände einer bestimmten vergangenen Messperiode	C_CI_NG_2
<input checked="" type="checkbox"/>	<111>:= Abruf periodisch rückgesetzter Abrechnungszählerstände bestimmten vergangenen Messperiode und eines ausgewählten Adressbereichs	C_CI_NH_2
<input checked="" type="checkbox"/>	<112>:= Abruf der Betriebszählerstände der ältesten Messperiode	C_CI_NI_2
<input checked="" type="checkbox"/>	<113>:= Abruf der Betriebszählerstände der ältesten Messperiode und eines ausgewählten Adressbereichs	C_CI_NK_2
<input checked="" type="checkbox"/>	<114>:= Abruf der Betriebszählerstände einer bestimmten vergangenen Messperiode	C_CI_NL_2
<input checked="" type="checkbox"/>	<115>:= Abruf der Betriebszählerstände einer bestimmten vergangenen Messperiode und eines ausgewählten Adressbereichs	C_CI_NM_2
<input checked="" type="checkbox"/>	<116>:= Abruf periodisch rückgesetzter Betriebszählerstände der ältesten Messperiode	C_CI_NN_2
<input checked="" type="checkbox"/>	<117>:= Abruf periodisch rückgesetzter Betriebszählerstände der ältesten Messperiode und eines ausgewählten Adressbereichs	C_CI_NO_2
<input checked="" type="checkbox"/>	<118>:= Abruf periodisch rückgesetzter Betriebszählerstände einer bestimmten vergangenen Messperiode	C_CI_NP_2
<input checked="" type="checkbox"/>	<119>:= Abruf periodisch rückgesetzter Betriebszählerstände einer bestimmten vergangenen Messperiode und eines ausgewählten Adressbereichs	C_CI_NQ_2
<input checked="" type="checkbox"/>	<120>:= Abruf der Abrechnungszählerstände eines ausgewählten Zeitbereichs und eines ausgewählten Adressbereichs	C_CI_NR_2
<input checked="" type="checkbox"/>	<121>:= Abruf periodisch rückgesetzter Abrechnungszählerstände eines ausgewählten Zeitbereichs und eines ausgewählten Adressbereichs	C_CI_NS_2
<input checked="" type="checkbox"/>	<122>:= Abruf der Betriebszählerstände eines ausgewählten Zeitbereichs und eines ausgewählten Adressbereichs	C_CI_NT_2
<input checked="" type="checkbox"/>	<123>:= Abruf periodisch rückgesetzter Betriebszählerstände eine ausgewählten Zeitbereichs und eines ausgewählten Adressbereichs	C_CI_NU_2

## Grundlegende Anwendungsfunktionen

### Übertragung von Zählwerten

(Stationsbezogener Parameter)

Anzahl der Abrechnungszählerstände:

Anzahl der periodisch rückgesetzten Abrechnungszählerstände:

Anzahl der Betriebszählerstände:

Anzahl der periodisch rückgesetzten Betriebszählerstände:

Zählerstände werden nur durch Abruffunktion übertragen (einschließlich der jüngsten Periode)

**Abruf von Listen**

(Stationsbezogener Parameter)

Listenumfang:

- |             |                          |                             |
|-------------|--------------------------|-----------------------------|
| Gesamtliste | <input type="checkbox"/> | Anzahl der Einzelmeldungen: |
| Teilliste 1 | <input type="checkbox"/> | Anzahl der Einzelmeldungen: |
| Teilliste 2 | <input type="checkbox"/> | Anzahl der Einzelmeldungen: |
| Teilliste 3 | <input type="checkbox"/> | Anzahl der Einzelmeldungen: |
| Teilliste 4 | <input type="checkbox"/> | Anzahl der Einzelmeldungen: |

Keine Liste, Einzelmeldungen werden spontan übertragen

**Übertragung von Einzelmeldungen**

(Stationsbezogener Parameter)

- Übertragung örtlich quittierter Einzelmeldungen
- Keine Übertragung örtlich quittierter Einzelmeldungen

**Anhang**

**2.2.3.1 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](http://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](mailto: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](mailto:service@beckhoff.com)

### **Beckhoff Unternehmenszentrale**

Beckhoff Automation GmbH & Co. KG

Hülshorstweg 20  
33415 Verl  
Deutschland

Telefon: +49 5246 963-0

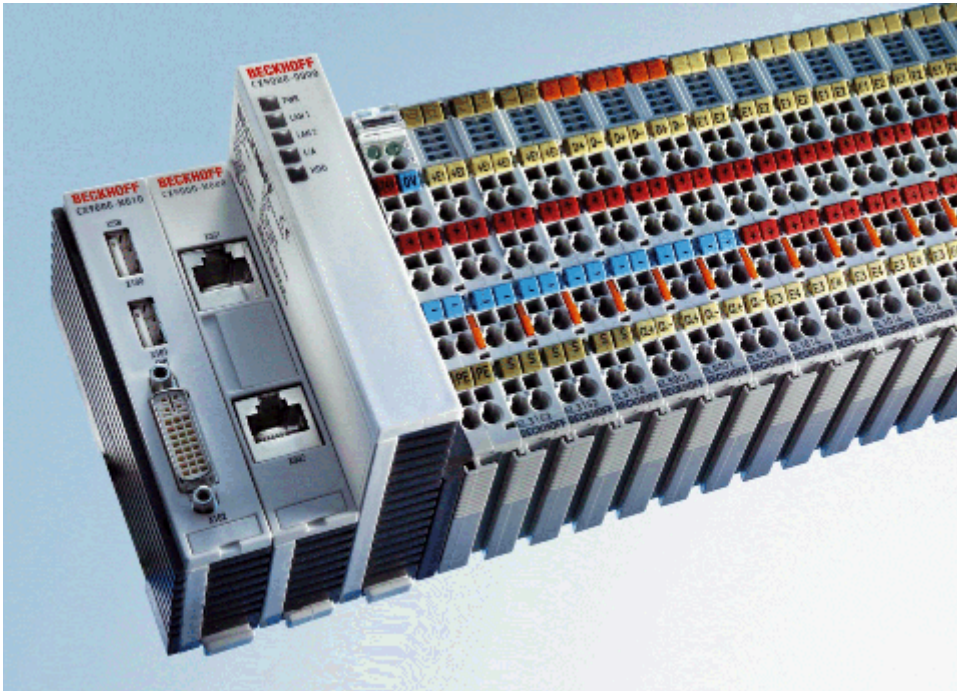
E-Mail: [info@beckhoff.com](mailto:info@beckhoff.com)

Internet: [www.beckhoff.com](http://www.beckhoff.com)



## 2.2.4 Kompatibilitätsliste nach IEC 60870-5-103 (master)

für TwinCAT SPS Bibliothek: IEC 60870-5-103 Zentralstation (master).



Version: 1.2

Datum: 07.03.2014

Hier können Sie die Kompatibilitätsliste öffnen/abspeichern (gezipptes PDF): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/984592267.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/984592267.zip)

### Inhaltsverzeichnis:

- [Allgemeine Informationen \[► 58\]](#)
  - [Projektinformationen \[► 58\]](#)
- [Kompatibilität \[► 58\]](#)
  - [System oder Gerät \[► 58\]](#)
  - [Physikalische Schicht \[► 58\]](#)

Elektrische Schnittstelle

Optische Schnittstelle

Übertragungsgeschwindigkeit

- Verbindungsschicht
- Anwendungsschicht

Übertragungsmodus für Anwendungsdaten

Gemeinsame Adresse der ASDU

Auswahl genormter ASDU in Überwachungsrichtung

Auswahl genormter ASDU in Steuerungsrichtung

Auswahl von Norm-Informationsnummern in Überwachungsrichtung

Auswahl von Norm-Informationsnummern in Steuerungsrichtung

[Grundlegende Anwendungsfunktionen \[► 65\]](#)

## Verschiedenes

- [Anhang \[▶ 66\]](#)
  - [Support und Service \[▶ 66\]](#)

**Allgemeine Informationen**

In dieser Kompatibilitätsliste finden Sie die unterstützten Funktionen innerhalb der IEC 870-5-103. Die Kompatibilität der einzelnen Komponenten muss von allen Beteiligten für das jeweilige Projekt vereinbart werden.

**Projektinformationen**

Projekt-Nr.:

---

Projekt:

---

Zuständig:

---

Datum:

---

**Kompatibilität**

Die ausgewählten Parameter sollten in den weißen Kontrollfeldern wie folgt markiert werden:

Funktion oder ASDU wird nicht benutzt

Funktion oder ASDU wird wie genormt benutzt

**System oder Gerät**

Systemfestlegung

Festlegungen für die Zentralstation

Festlegungen für die Unterstation

**Physikalische Schicht**

EIA RS-485

Anzahl Lasten ..... für eine Schutzeinrichtung



EIA RS-485 legt die Lasten [en: unit loads] derart fest, dass an einer Linie 32 von ihnen betrieben werden können.

Einzelinformationen siehe von EIA RS 485, Abschnitt 3

- Glasfaser
- Kunststofffaser
- F-SMA-Stecker
- BFOC/2,5-Stecker
- 9600 bit/s
- 19200 bit/s

**Verbindungsschicht**

Für die Verbindungsschicht bestehen keine Auswahlmöglichkeiten.

**Anwendungsschicht**

Nach dieser anwendungsbezogenen Norm wird ausschließlich Betriebsart 1 (niedrigstwertiges Oktett zuerst) nach 4.10 von IEC 60870-5-4 benutzt.

<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	Identisch mit der Stationsadresse		Nicht identisch mit der Stationsadresse
<input checked="" type="checkbox"/>	Eine GEMEINSAME ADRESSE der ASDU	<input checked="" type="checkbox"/>	
<input type="checkbox"/>	Mehr als eine GEMEINSAME ADRESSE der ASDU		
	<0> := wird nicht benutzt		-
<input checked="" type="checkbox"/>	<1> := Meldung mit Zeitmarke		M_TTM_TA_3
<input checked="" type="checkbox"/>	<2> := Meldung mit Relativzeit		M_TMR_TA_3
<input checked="" type="checkbox"/>	<3> := Messwerte I		M_MEI_NA_3
<input checked="" type="checkbox"/>	<4> := Echtzeitmesswerte mit Relativzeit		M_TME_TA_3
<input checked="" type="checkbox"/>	<5> := Identifikationsmeldung		M_IRC_NA_3
<input checked="" type="checkbox"/>	<6> := Zeitsynchronisierung		M_SYN_TA_3
<input checked="" type="checkbox"/>	<8> := Generalabfrage-Ende		M_TGI_NA_3
<input checked="" type="checkbox"/>	<9> := Messwerte II		M_MEII_NA_3
<input checked="" type="checkbox"/>	<10> := Generische Daten		M_GD_XA_3
<input checked="" type="checkbox"/>	<11> := Generische Identifikation		M_GI_XA_3
<input checked="" type="checkbox"/>	<23> := Störfallübersicht		M_LRD_TA_3
<input checked="" type="checkbox"/>	<26> := Bereit zur Übertragung von Stördaten		M_RTD_TA_3
<input checked="" type="checkbox"/>	<27> := Bereit zur Übertragung eines Kanals		M_RTC_NA_3
<input checked="" type="checkbox"/>	<28> := Bereit zur Übertragung von Marken		M_RTT_NA_3
<input checked="" type="checkbox"/>	<29> := Übertragung von Marken		M_TOT_NA_3
<input checked="" type="checkbox"/>	<30> := Übertragung von Störwerten		M_TOV_NA_3
<input checked="" type="checkbox"/>	<31> := Ende der Übertragung		M_EOT_NA_3

<input checked="" type="checkbox"/>	<0> := wird nicht benutzt	-
<input checked="" type="checkbox"/>	<6> := Zeitsynchronisierung	C_SYN_TA_3
<input checked="" type="checkbox"/>	<7> := Generalabfragebefehl	C_IGI_NA_3
<input checked="" type="checkbox"/>	<10> := Generische Daten	C_GD_NA_3
<input checked="" type="checkbox"/>	<20> := Allgemeiner Befehl	C_GRC_NA_3
<input checked="" type="checkbox"/>	<21> := Generischer Befehl	C_GC_NA_3
<input checked="" type="checkbox"/>	<24> := Auftrag zur Übertragung von Stördaten	C_ODT_NA_3
<input checked="" type="checkbox"/>	<25> := Quittung für Stördatenübertragung	C_ADT_NA_3

**Auswahl von Norm-Informationsnummern in Überwachungsrichtung**

*Systemfunktionen in Überwachungsrichtung*

	<b>INF</b>	<b>Beschreibung</b>
<input type="checkbox"/>	<0>	Ende der Generalabfrage
<input type="checkbox"/>	<1>	Zeitsynchronisierung
<input type="checkbox"/>	<2>	Rücksetzen FCB
<input type="checkbox"/>	<3>	Rücksetzen KE
<input type="checkbox"/>	<4>	Anlauf / Wiederanlauf
<input type="checkbox"/>	<5>	Erstanlauf

*Zustandsmeldungen in Überwachungsrichtung*

	<b>INF</b>	<b>Beschreibung</b>
<input type="checkbox"/>	<16>	Autom. Wiedereinschaltung aktiv
<input type="checkbox"/>	<17>	Schutzsignalübertragung aktiv
<input type="checkbox"/>	<18>	Schutz aktiv
<input type="checkbox"/>	<19>	Rückstellung der Anzeigen
<input type="checkbox"/>	<20>	Sperre der Überwachungsrichtung
<input type="checkbox"/>	<21>	Testbetrieb
<input type="checkbox"/>	<22>	Parametrierung vor Ort
<input type="checkbox"/>	<23>	Kennlinie 1
<input type="checkbox"/>	<24>	Kennlinie 2
<input type="checkbox"/>	<25>	Kennlinie 3
<input type="checkbox"/>	<26>	Kennlinie 4
<input type="checkbox"/>	<27>	Eingang 1
<input type="checkbox"/>	<28>	Eingang 2
<input type="checkbox"/>	<29>	Eingang 3
<input type="checkbox"/>	<30>	Eingang 4

*Überwachungsmeldungen in Überwachungsrichtung*

	<b>INF</b>	<b>Beschreibung</b>
<input type="checkbox"/>	<32>	Messwertüberwachung I
<input type="checkbox"/>	<33>	Messwertüberwachung V
<input type="checkbox"/>	<35>	Drehfeldüberwachung
<input type="checkbox"/>	<36>	Auslösekreisüberwachung
<input type="checkbox"/>	<37>	UMZ-Notbetrieb
<input type="checkbox"/>	<38>	VT Automatenfall
<input type="checkbox"/>	<39>	Schutzsignalübertragung gestört
<input type="checkbox"/>	<46>	Warnsammelmeldung
<input type="checkbox"/>	<47>	Störungssammelmeldung

*Erdschlussmeldungen in Überwachungsrichtung*

	<b>INF</b>	<b>Beschreibung</b>
<input type="checkbox"/>	<48>	Erdschluss L <sub>1</sub>
<input type="checkbox"/>	<49>	Erdschluss L <sub>2</sub>
<input type="checkbox"/>	<50>	Erdschluss L <sub>3</sub>
<input type="checkbox"/>	<51>	Erdschluss vorwärts, d.h. Leitung
<input type="checkbox"/>	<52>	Erdschluss rückwärts, d.h. Sammelschiene

*Störfallmeldungen in Überwachungsrichtung*

	<b>INF</b>	<b>Beschreibung</b>
<input type="checkbox"/>	<64>	Anregung L <sub>1</sub>
<input type="checkbox"/>	<65>	Anregung L <sub>2</sub>
<input type="checkbox"/>	<66>	Anregung L <sub>3</sub>
<input type="checkbox"/>	<67>	Anregung N
<input type="checkbox"/>	<68>	General-Auslösung
<input type="checkbox"/>	<69>	Auslösung L <sub>1</sub>
<input type="checkbox"/>	<70>	Auslösung L <sub>2</sub>
<input type="checkbox"/>	<71>	Auslösung L <sub>3</sub>
<input type="checkbox"/>	<72>	Auslösung UMZ-Notbetrieb
<input type="checkbox"/>	<73>	Fehlerort X in Ohm
<input type="checkbox"/>	<74>	Fehler vorwärts / Leitung
<input type="checkbox"/>	<75>	Fehler rückwärts / Sammelschiene
<input type="checkbox"/>	<76>	Schutzsignalübertragung Signal gesendet
<input type="checkbox"/>	<77>	Schutzsignalübertragung Signal empfangen
<input type="checkbox"/>	<78>	Stufe 1
<input type="checkbox"/>	<79>	Stufe 2
<input type="checkbox"/>	<80>	Stufe 3
<input type="checkbox"/>	<81>	Stufe 4
<input type="checkbox"/>	<82>	Stufe 5
<input type="checkbox"/>	<83>	Stufe 6
<input type="checkbox"/>	<84>	Generalanregung
<input type="checkbox"/>	<85>	Schalterversager
<input type="checkbox"/>	<86>	Auslösung Messsystem L <sub>1</sub>
<input type="checkbox"/>	<87>	Auslösung Messsystem L <sub>2</sub>
<input type="checkbox"/>	<88>	Auslösung Messsystem L <sub>3</sub>
<input type="checkbox"/>	<89>	Auslösung Messsystem E
<input type="checkbox"/>	<90>	Auslösung I>
<input type="checkbox"/>	<91>	Auslösung I>>
<input type="checkbox"/>	<92>	Auslösung IN>
<input type="checkbox"/>	<93>	Auslösung IN>>

*AWE-Meldungen in Überwachungsrichtung*

	<b>INF</b>	<b>Beschreibung</b>
<input type="checkbox"/>	<128>	LS EIN durch AWE
<input type="checkbox"/>	<129>	LS EIN durch Langzeit AWE
<input type="checkbox"/>	<130>	AWE gesperrt

**Betriebsmesswerte in Überwachungsrichtung**

	<b>INF</b>	<b>Beschreibung</b>
<input type="checkbox"/>	<144>	Betriebsmesswert I
<input type="checkbox"/>	<145>	Betriebsmesswerte I, U
<input type="checkbox"/>	<146>	Betriebsmesswerte I, U, P, Q
<input type="checkbox"/>	<147>	Betriebsmesswerte I <sub>N</sub> , U <sub>EN</sub>
<input type="checkbox"/>	<148>	Betriebsmesswerte I <sub>L1,2,3</sub> , U <sub>L1,2,3</sub> , P, Q, f

**Generische Funktionen in Überwachungsrichtung**

	<b>INF</b>	<b>Beschreibung</b>
<input type="checkbox"/>	<240>	Bezeichnung aller festgelegten Gruppen lesen
<input type="checkbox"/>	<241>	Werte oder Attribute aller Einträge einer Gruppe lesen
<input type="checkbox"/>	<242>	Verzeichnis eines einzelnen Eintrags lesen
<input type="checkbox"/>	<243>	Werte oder Attribute eines einzelnen Eintrags lesen
<input type="checkbox"/>	<244>	Ende Generalabfrage Generische Daten
<input type="checkbox"/>	<249>	Eintrag schreiben (mit Bestätigung)
<input type="checkbox"/>	<250>	Eintrag schreiben (mit Ausführung)
<input type="checkbox"/>	<251>	Eintrag schreiben (mit Abbruch)

**Auswahl von Norm-Informationsnummern in Steuerungsrichtung****Systemfunktionen in Steuerungsrichtung**

	<b>INF</b>	<b>Beschreibung</b>
<input type="checkbox"/>	<0>	Generalabfrage-Abstoß
<input type="checkbox"/>	<1>	Zeitsynchronisierung

**Allgemeine Befehle in Steuerungsrichtung**



	<b>INF</b>	<b>Beschreibung</b>
<input type="checkbox"/>	<16>	Wiedereinschaltung EIN/AUS
<input type="checkbox"/>	<17>	Schutzsignalübertragung EIN/AUS
<input type="checkbox"/>	<18>	Schutz EIN/AUS
<input type="checkbox"/>	<19>	Rückstellung der Anzeigen
<input type="checkbox"/>	<23>	Aktivieren der Kennlinie 1
<input type="checkbox"/>	<24>	Aktivieren der Kennlinie 2
<input type="checkbox"/>	<25>	Aktivieren der Kennlinie 3
<input type="checkbox"/>	<26>	Aktivieren der Kennlinie 4

**Generische Funktionen in Steuerungsrichtung**

	<b>INF</b>	<b>Beschreibung</b>
<input type="checkbox"/>	<240>	Bezeichnung aller festgelegten Gruppen lesen
<input type="checkbox"/>	<241>	Werte oder Attribute aller Einträge einer Gruppe lesen
<input type="checkbox"/>	<243>	Verzeichnis eines einzelnen Eintrags lesen
<input type="checkbox"/>	<244>	Werte oder Attribute eines einzelnen Eintrags lesen
<input type="checkbox"/>	<245>	Ende Generalabfrage Generische Daten
<input type="checkbox"/>	<248>	Eintrag schreiben
<input type="checkbox"/>	<249>	Eintrag schreiben (mit Bestätigung)
<input type="checkbox"/>	<250>	Eintrag schreiben (mit Ausführung)
<input type="checkbox"/>	<251>	Eintrag schreiben (mit Abbruch)

**Grundlegende Anwendungsfunktionen**

<input type="checkbox"/>	Testbetrieb
<input type="checkbox"/>	Sperre der Überwachungsrichtung
<input type="checkbox"/>	Stördaten
<input type="checkbox"/>	Generische Dienste
<input type="checkbox"/>	Private Daten

**Verschiedenes**

Messwerte werden sowohl mit ASDU <3> als auch mit ASDU <9> übertragen. Nach 7.2.6.8 kann MVAL maximal entweder 1,2 oder 2,4 Mal Nennwert betragen. In den ASDU <3> und ASDU <9> dürfen keine unterschiedlichen Bezugsfaktoren benutzt werden, d.h., für jeden Messwert gibt es nur eine Auswahl.

Messwert	Max. MVAL	
	Nennwert x ( 1 oder 2 )	Nennwert x ( 2 oder 4 )
Strom L <sub>1</sub>	<input type="checkbox"/>	<input type="checkbox"/>
Strom L <sub>2</sub>	<input type="checkbox"/>	<input type="checkbox"/>
Strom L <sub>3</sub>	<input type="checkbox"/>	<input type="checkbox"/>
Spannung L <sub>1-E</sub>	<input type="checkbox"/>	<input type="checkbox"/>
Spannung L <sub>2-E</sub>	<input type="checkbox"/>	<input type="checkbox"/>
Spannung L <sub>3-E</sub>	<input type="checkbox"/>	<input type="checkbox"/>
Wirkleistung Q	<input type="checkbox"/>	<input type="checkbox"/>
Frequenz f	<input type="checkbox"/>	<input type="checkbox"/>
Spannung zwischen L <sub>1</sub> und L <sub>2</sub>	<input type="checkbox"/>	<input type="checkbox"/>

## Anhang

### 2.2.4.1 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](http://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](mailto: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](mailto:service@beckhoff.com)

**Beckhoff Unternehmenszentrale**

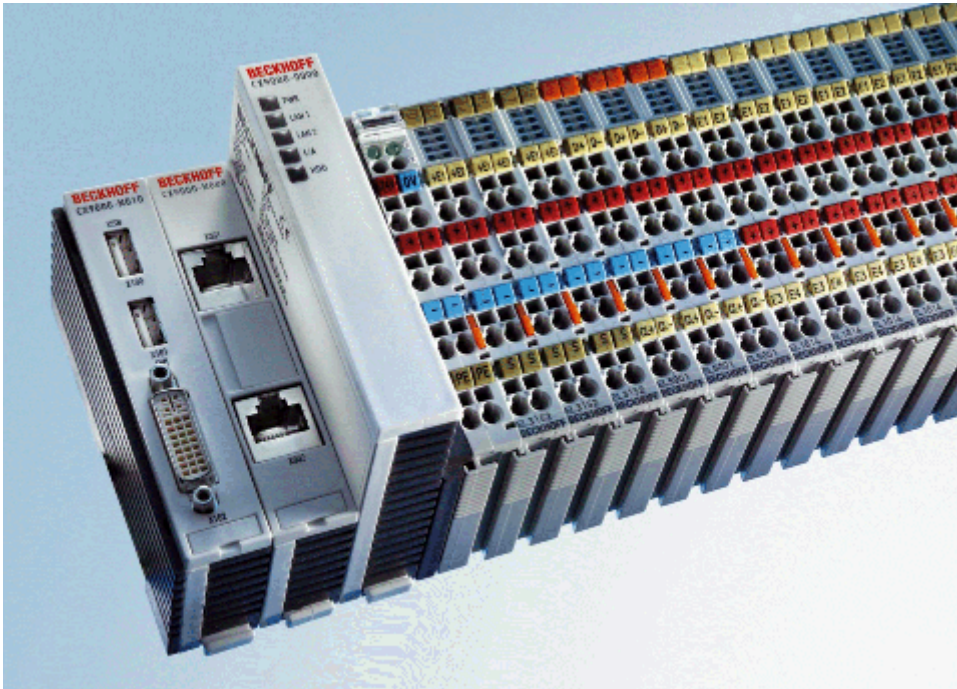
Beckhoff Automation GmbH & Co. KG

Hülshorstweg 20  
33415 Verl  
Deutschland

Telefon: +49 5246 963-0  
E-Mail: [info@beckhoff.com](mailto:info@beckhoff.com)  
Internet: [www.beckhoff.com](http://www.beckhoff.com)

## 2.2.5 Kompatibilitätsliste nach IEC 60870-5-104 (master)

für TwinCAT SPS Bibliothek: IEC 60870-5-104 Zentralstation (master)



Version: 1.4

Datum: 07.03.2014

Hier können Sie die Kompatibilitätsliste öffnen/abspeichern (gezipptes PDF): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/984595211.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/984595211.zip)

### Inhaltsverzeichnis:

- [Allgemeine Informationen \[▶ 69\]](#)
  - [Projektinformationen \[▶ 69\]](#)
- [Kompatibilität \[▶ 69\]](#)
  - [System oder Gerät \[▶ 70\]](#)
  - [Netzkonfiguration \[▶ 70\]](#)
  - [Physikalische Schicht \[▶ 70\]](#)
  - [Verbindungsschicht \[▶ 70\]](#)
  - [Anwendungsschicht \[▶ 70\]](#)
  - [Auswahl genormter ASDU's \[▶ 72\]](#)

Prozessinformation in Überwachungsrichtung

Prozessinformation in Steuerungsrichtung

Systeminformation in Überwachungsrichtung

Systeminformation in Steuerungsrichtung

Parameter in Steuerungsrichtung

Dateiübermittlung

- [Grundlegende Anwendungsfunktionen \[▶ 80\]](#)
  - Stationsinitialisierung

- Zyklische Datenübertragung
- Abrufprozedur
- Spontane Übertragung
- Doppelübertragung
- Stationsabfrage
- Uhrzeitsynchronisation
- Befehlsübertragung
- Zählerabfrage
- Laden eines Parameters
- Parameter für Aktivierung
- Prüfprozedur
- Dateiübermittlung
- Hintergrundabfrage
- Telegrammlaufzeiterfassung
- Festlegungen für Zeitüberwachung [▶ 83]
- k- und w-Parameter
- Portnummer [▶ 84]
- RFC-2200-Sammlung [▶ 84]
- Anhang [▶ 84]
  - Support und Service [▶ 84]

**Allgemeine Informationen**

In dieser Kompatibilitätsliste finden Sie die unterstützten Funktionen innerhalb der IEC 870-5-104.



Dieses Kontrollfeld bedeutet, dass die Funktion bei der TCP/IP-Kopplung über IEC 870-5-104 nicht verwendet wird.

Die Kompatibilität der einzelnen Komponenten muss von allen Beteiligten für das jeweilige Projekt vereinbart werden.

**Projektinformationen**

Projekt-Nr.:

---

Projekt:

---

Zuständig:

---

Datum:

---

**Kompatibilität**

Die ausgewählten Parameter sollten in den weißen Kontrollfeldern wie folgt markiert werden:



Funktion oder ASDU wird nicht benutzt



Funktion oder ASDU wird wie genormt benutzt (Vorzugswert)



Funktion oder ASDU wird im Umkehrmodus benutzt



Funktion oder ASDU wird im Regel- und Umkehrmodus benutzt

**System oder Gerät**

- Systemfestlegung
- Festlegungen für die Zentralstation (MASTER)
- Festlegungen für die Unterstation

**Netzkonfiguration**

- |                          |   |                          |   |
|--------------------------|---|--------------------------|---|
| <input type="checkbox"/> | End-End-Konfiguration[en: Point-to-point]                       | <input type="checkbox"/> | Linienkonfiguration<br>[en: Multipoint-partyline] |
| <input type="checkbox"/> | Mehrfach-End-End-Konfiguration<br>[en: Multiple point-to-point] | <input type="checkbox"/> | Sternkonfiguration<br>[en: Multipoint-star]       |

**Physikalische Schicht***Übertragungsgeschwindigkeit:*

- |                          |            |                          |             |                          |             |
|--------------------------|------------|--------------------------|-------------|--------------------------|-------------|
| <input type="checkbox"/> | 100 bit/s  | <input type="checkbox"/> | 2400 bit/s  | <input type="checkbox"/> | 56000 bit/s |
| <input type="checkbox"/> | 200 bit/s  | <input type="checkbox"/> | 4800 bit/s  | <input type="checkbox"/> | 64000 bit/s |
| <input type="checkbox"/> | 300 bit/s  | <input type="checkbox"/> | 9600 bit/s  |                          |             |
| <input type="checkbox"/> | 600 bit/s  | <input type="checkbox"/> | 19200 bit/s |                          |             |
| <input type="checkbox"/> | 1200 bit/s | <input type="checkbox"/> | 38400 bit/s |                          |             |

**Verbindungsschicht**

Nach dieser Anwendungsbezogenen Norm werden ausschließlich Telegrammformat FT 1.2, Einzelzeichen 1 und das feste Zeitüberwachungsintervall benutzt.

*Adressfeld der Verbindungsschicht:*

- Nicht vorhanden (nur symmetrische Übertragung)
- Ein Oktett
- Zwei Oktette
- Strukturiert
- Unstrukturiert

*Übertragungsverfahren der Verbindungsschicht:*

- Symmetrische Übertragung
- Unsymmetrische Übertragung

*Telegrammlänge:*

- Maximale Länge L

**Anwendungsschicht****Übertragungsbetriebsart für Anwendungsdaten**

Nach dieser Anwendungsbezogenen Norm wird ausschließlich Betriebsart 1 (niedrigstwertiges Oktett zuerst) nach 4.10 von IEC 60870-5-4 benutzt.

**Gemeinsame Adresse der ASDU**

<input type="checkbox"/>	Ein Oktett	<input checked="" type="checkbox"/>	Zwei Oktette
--------------------------	------------	-------------------------------------	--------------

**Adresse des Informationsobjekts**

<input type="checkbox"/>	Ein Oktett	<input checked="" type="checkbox"/>	Strukturiert
<input type="checkbox"/>	Zwei Oktette	<input checked="" type="checkbox"/>	Unstrukturiert
<input checked="" type="checkbox"/>	Drei Oktette		

**Übertragungsursache**

<input type="checkbox"/>	Ein Oktett	<input checked="" type="checkbox"/>	Zwei Oktette(mit Herkunftsadresse). Mit 0 vorbesetzt, falls Herkunftsadresse nicht vorhanden.
--------------------------	------------	-------------------------------------	--

**Länge der APDU**

(systembezogener Parameter, die maximale Länge der APDU je System ist festzulegen)

Die maximale Länge der APDU beträgt 253 (falls nicht anders festgelegt). Je System darf die maximale Länge

reduziert werden.

253            Maximale Länge der APDU je System

**Auswahl genormter ASDU**

*Tab. 42: Prozessinformation in Überwachungsrichtung*



<input checked="" type="checkbox"/>	<1> := Einzelmeldung	M_SP_NA_1
<input type="checkbox"/>	<2> := Einzelmeldung mit Zeitmarke	M_SP_TA_1
<input checked="" type="checkbox"/>	<3> := Doppelmeldung	M_DP_NA_1
<input type="checkbox"/>	<4> := Doppelmeldung mit Zeitmarke	M_DP_TA_1
<input checked="" type="checkbox"/>	<5> := Stufenstellungsmeldung	M_ST_NA_1
<input type="checkbox"/>	<6> := Stufenstellungsmeldung mit Zeitmarke	M_ST_TA_1
<input checked="" type="checkbox"/>	<7> := Bitmuster von 32 bit	M_BO_NA_1
<input type="checkbox"/>	<8> := Bitmuster von 32 bit mit Zeitmarke	M_BO_TA_1
<input checked="" type="checkbox"/>	<9> := Messwert, normierter Wert	M_ME_NA_1
<input type="checkbox"/>	<10> := Messwert, normierter Wert mit Zeitmarke	M_ME_TA_1
<input checked="" type="checkbox"/>	<11> := Messwert, skaliertes Wert	M_ME_NB_1
<input type="checkbox"/>	<12> := Messwert, skaliertes Wert mit Zeitmarke	M_ME_TB_1
<input checked="" type="checkbox"/>	<13> := Messwert, verkürzte Gleitkommazahl	M_ME_NC_1
<input type="checkbox"/>	<14> := Messwert, verkürzte Gleitkommazahl mit Zeitmarke	M_ME_TC_1
<input checked="" type="checkbox"/>	<15> := Zählwerte	M_IT_NA_1
<input type="checkbox"/>	<16> := Zählwerte mit Zeitmarke	M_IT_TA_1
<input type="checkbox"/>	<17> := Schutzereignis mit Zeitmarke	M_EP_TA_1
<input type="checkbox"/>	<18> := Geblockte Anregungen des Schutzes mit Zeitmarke	M_EP_TB_1
<input type="checkbox"/>	<19> := Geblockte Auslösungen des Schutzes mit Zeitmarke	M_EP_TC_1
<input checked="" type="checkbox"/>	<20> := Geblockte Einzelmeldungen mit Zustandsanzeige	M_PS_NA_1
<input checked="" type="checkbox"/>	<21> := Messwert, normierter Wert ohne Qualitätskennung	M_ME_ND_1
<input checked="" type="checkbox"/>	<30> := Einzelmeldung mit Zeitmarke CP56Time2a	M_SP_TB_1
<input checked="" type="checkbox"/>	<31> := Doppelmeldung mit Zeitmarke CP56Time2a	M_DP_TB_1
<input checked="" type="checkbox"/>	<32> := Stufenstellungsmeldung mit Zeitmarke CP56Time2a	M_ST_TB_1
<input checked="" type="checkbox"/>	<33> := Bitmuster von 32 bit mit Zeitmarke CP56Time2a	M_BO_TB_1
<input checked="" type="checkbox"/>	<34> := Messwert, normierter Wert mit Zeitmarke CP56Time2a	M_ME_TD_1
<input checked="" type="checkbox"/>	<35> := Messwert, skaliertes Wert mit Zeitmarke CP56Time2a	M_ME_TE_1
<input checked="" type="checkbox"/>	<36> := Messwert, verkürzte Gleitkommazahl mit Zeitmarke CP56Time2a	M_ME_TF_1
<input checked="" type="checkbox"/>	<37> := Zählwerte mit Zeitmarke CP56Time2a	M_IT_TB_1
<input checked="" type="checkbox"/>	<38> := Schutzereignis mit Zeitmarke CP56Time2a	M_EP_TD_1
<input checked="" type="checkbox"/>	<39> := Geblockte Anregungen des Schutzes mit Zeitmarke CP56Time2a	M_EP_TE_1

<input checked="" type="checkbox"/>	<40> := Geblockte Auslösungen des Schutzes mit Zeitmarke CP56Time2a	M_EP_TF_1
-------------------------------------	---	-----------

Tab. 43: Prozessinformation in Steuerungsrichtung

<input checked="" type="checkbox"/>	<45> := Einzelbefehl	C_SC_NA_1
<input checked="" type="checkbox"/>	<46> := Doppelbefehl	C_DC_NA_1
<input type="checkbox"/>	<47> := Stufenstellbefehl	C_RC_NA_1
<input checked="" type="checkbox"/>	<48> := Sollwert-Stellbefehl, normierter Wert	C_SE_NA_1
<input checked="" type="checkbox"/>	<49> := Sollwert-Stellbefehl, skaliertes Wert	C_SE_NB_1
<input checked="" type="checkbox"/>	<50> := Sollwert-Stellbefehl, verkürzte Gleitkommazahl	C_SE_NC_1
<input checked="" type="checkbox"/>	<51> := Bitmuster von 32 bit	C_BO_NA_1
<input checked="" type="checkbox"/>	<58> := Einzelbefehl mit Zeitmarke CP56Time2a	C_SC_TA_1
<input checked="" type="checkbox"/>	<59> := Doppelbefehl mit Zeitmarke CP56Time2a	C_DC_TA_1
<input type="checkbox"/>	<60> := Stufenstellbefehl mit Zeitmarke CP56Time2a	C_RC_TA_1
<input checked="" type="checkbox"/>	<61> := Sollwert-Stellbefehl, normierter Wert mit Zeitmarke CP56Time2a	C_SE_TA_1
<input checked="" type="checkbox"/>	<62> := Sollwert-Stellbefehl, skaliertes Wert mit Zeitmarke CP56Time2a	C_SE_TB_1
<input checked="" type="checkbox"/>	<63> := Sollwert-Stellbefehl, verkürzte Gleitkommazahl mit Zeitmarke CP56Time2a	C_SE_TC_1
<input checked="" type="checkbox"/>	<64> := Bitmuster von 32 bit mit Zeitmarke CP56Time2a	C_BO_TA_1

Tab. 44: Systeminformation in Überwachungsrichtung

<input checked="" type="checkbox"/>	<70> := Initialisierungsende	M_EI_NA_1
-------------------------------------	------------------------------	-----------

Tab. 45: Systeminformation in Steuerungsrichtung

<input checked="" type="checkbox"/>	<100>:= (Stations-)Abfragebefehl	C_IC_NA_1
<input checked="" type="checkbox"/>	<101>:= Zählerabfragebefehl	C_CI_NA_1
<input type="checkbox"/>	<102>:= Abfragebefehl (Lesebefehl)	C_RD_NA_1
<input checked="" type="checkbox"/>	<103>:= Uhrzeit-Synchronisationsbefehl	C_CS_NA_1
<input type="checkbox"/>	<104>:= Prüfbefehl	C_TS_NA_1
<input type="checkbox"/>	<105>:= Prozess-Rücksetzbefehl	C_RP_NA_1
<input type="checkbox"/>	<106>:= Befehl zur Telegrammlaufzeit-Erfassung	C_CD_NA_1
<input checked="" type="checkbox"/>	<107>:= Prüfbefehl mit Zeitmarke CP56Time2a	C_TS_TA_1

*Tab. 46: Parameter in Steuerungsrichtung*

<input type="checkbox"/>	<110>:= Parameter für Messwerte, normierter Wert	P_ME_NA_1
<input type="checkbox"/>	<111>:= Parameter für Messwerte, skaliertes Wert	P_ME_NB_1
<input type="checkbox"/>	<112>:= Parameter für Messwerte, verkürzte Gleitkommazahl	P_ME_NC_1
<input type="checkbox"/>	<113>:= Parameter für Aktivierung	P_AC_NA_1

*Tab. 47: Dateiübermittlung*

<input type="checkbox"/>	<120>:= Datei bereit	F_FR_NA_1
<input type="checkbox"/>	<121>:= Abschnitt bereit	F_SR_NA_1
<input type="checkbox"/>	<122>:= Abfrage Dateiverzeichnis, -auswahl, -abfrage, Abschnittsabfrage	F_SC_NA_1
<input type="checkbox"/>	<123>:= Letzter Abschnitt, letztes Segment	F_LS_NA_1
<input type="checkbox"/>	<124>:= Dateibestätigung, Abschnittsbestätigung	F_AF_NA_1
<input type="checkbox"/>	<125>:= Segment	F_SG_NA_1
<input type="checkbox"/>	<126>:= Dateiverzeichnis[leer oder <b>X</b> , nur in Überwachungsrichtung verfügbar (regulär)]	F_DR_TA_1

**Zuweisungen der Übertragungsursachen zu den Typkennungen(stationsbezogene Parameter)**

Markierung der Kombination Typkennung / Übertragungsursache mit:

Schattiert	Funktion oder ASDU wird nicht benötigt
Leer	Funktion oder ASDU wird nicht angewendet
<input checked="" type="checkbox"/>	Nur in der Normrichtung angewendet
<input checked="" type="checkbox"/>	Nur in der Umkehrrichtung angewendet
<input checked="" type="checkbox"/>	In beiden Richtungen angewendet

Tab. 48: Übertragungsursachen

Nummer	Übertragungsursache
<0>	Not used
<1>	Periodic, cyclic
<2>	Background scan
<3>	Spontaneous
<4>	Initialized
<5>	Request or requested
<6>	Activation (ACT)
<7>	Activation confirmation (ACTCON)
<8>	Deactivation (DEACT)
<9>	Deactivation confirmation (DEACTCON)
<10>	Activation termination (ACTTERM)
<11>	Return information caused by a remote command
<12>	Return information caused by a local command
<13>	File transfer
<20..36>	General interrogation
<37..41>	Counter interrogation
<44>	Unknown type identification
<45>	Unknown cause of transmission
<46>	Unknown address of ASDU
<47>	Unknown information object address

Typkennung		Übertragungsursache																		
		<1>	<2>	<3>	<4>	<5>	<6>	<7>	<8>	<9>	<10>	<11>	<12>	<13>	<20..36>	<37..41>	<44>	<45>	<46>	<47>
		per/cyc	back	spon	init	req	act	actcon	deact	deactcon	actterm	retrem	retloc	file	inro	reqco	unknowntype	unknowncot	unknowncaa	unknownlOA
<1>	M_S P_N A_1		X	X								X	X		X					
<2>	M_S P_T A_1																			
<3>	M_D P_N A_1		X	X								X	X		X					
<4>	M_D P_T A_1																			
<5>	M_S T_N A_1		X	X								X	X		X					
<6>	M_S T_TA _1																			
<7>	M_B O_N A_1		X	X											X					
<8>	M_B O_T A_1																			
<9>	M_M E_N A_1	X	X	X											X					
<10>	M_M E_T A_1																			
<11>	M_M E_N B_1	X	X	X											X					
<12>	M_M E_T B_1																			
<13>	M_M E_N C_1	X	X	X											X					
<14>	M_M E_T C_1																			
<15>	M_IT _NA _1			X												X				
<16>	M_IT _TA _1																			
<17>	M_E P_T A_1																			
<18>	M_E P_T B_1																			

Typkennung	Übertragungsursache																
<19> M_E P_T C_1																	
<20> M_P S_N A_1		X	X						X	X		X					
<21> M_M E_N D_1	X	X	X									X					
<30> M_S P_T B_1			X						X	X							
<31> M_D P_T B_1			X						X	X							
<32> M_S T_TB _1			X						X	X							
<33> M_B O_T B_1			X														
<34> M_M E_T D_1			X														
<35> M_M E_T E_1			X														
<36> M_M E_TF _1			X														
<37> M_IT _TB _1			X										X				
<38> M_E P_T D_1			X														
<39> M_E P_T E_1			X														
<40> M_E P_TF _1			X														
<45> C_S C_N A_1						X	X	X	X	X				X	X	X	X
<46> C_D C_N A_1						X	X	X	X	X				X	X	X	X
<47> C_R C_N A_1																	
<48> C_S E_N A_1						X	X	X	X	X				X	X	X	X
<49> C_S E_N B_1						X	X	X	X	X				X	X	X	X
<50> C_S E_N C_1						X	X	X	X	X				X	X	X	X
<51> C_B O_N A_1						X	X	X	X	X				X	X	X	X
<58> C_S C_T A_1						X	X	X	X	X				X	X	X	X
<59> C_D C_T A_1						X	X	X	X	X				X	X	X	X

Typkennung	Übertragungsursache																	
<60>	C_R C_T A_1																	
<61>	C_S E_T A_1					X	X	X	X	X					X	X	X	X
<62>	C_S E_T B_1					X	X	X	X	X					X	X	X	X
<63>	C_S E_T C_1					X	X	X	X	X					X	X	X	X
<64>	C_B O_T A_1					X	X	X	X	X					X	X	X	X
<70>	M_EI _NA _1				X													
<100>	C_IC _NA _1					X	X	X	X	X					X	X	X	X
<101>	C_CI _NA _1					X	X			X					X	X	X	X
<102>	C_R D_N A_1																	
<103>	C_C S_N A_1			X		X	X								X	X	X	X
<104>	C_T S_N A_1																	
<105>	C_R P_N A_1																	
<106>	C_C D_N A_1																	
<107>	C_T S_T A_1					X	X								X	X	X	X
<110>	P_M E_N A_1														X	X	X	X
<111>	P_M E_N B_1														X	X	X	X
<112>	P_M E_N C_1														X	X	X	X
<113>	P_A C_N A_1														X	X	X	X
<120>	F_F R_N A_1														X	X	X	X
<121>	F_S R_N A_1														X	X	X	X
<122>	F_S C_N A_1														X	X	X	X
<123>	F_LS _NA _1														X	X	X	X
<124>	F_AF _NA _1														X	X	X	X

Typkennung		Übertragungsursache																			
<125 >	F_S G_N A_1																	X	X	X	X
<126 >	F_D R_T A_1																				

**Grundlegende Anwendungsfunktionen**

**Stationsinitialisierung**

Remote initialization

*Tab. 49: Zyklische Datenübertragung*

Zyklische Datenübertragung

*Tab. 50: Abrufprozedur*

Abrufprozedur

*Tab. 51: Spontane Übertragung*

Spontane Übertragung

**Doppelübertragung von Informationsobjekten mit der Übertragungsursache spontan**

Die folgenden Typkennungen dürfen nacheinander in Folge eines einzigen Zustandswechsels eines Informationsobjekts, übertragen werden. Die einzelnen Adressen der Informationsobjekte, die für die Doppelübertragung vorgesehen sind, werden in einer projektbezogenen Liste festgelegt.

- Einzelmeldung M\_SP\_NA\_1, M\_SP\_TA\_1, M\_SP\_TB\_1 und M\_SP\_NA\_1
- Doppelmeldung M\_DP\_NA\_1, M\_DP\_TA\_1 und M\_DP\_TB\_1
- Stufenstellungsmeldung M\_ST\_NA\_1, M\_ST\_TA\_1 und M\_ST\_TB\_1
- Bitmuster von 32 bit M\_BO\_NA\_1, M\_BO\_TA\_1 und M\_BO\_TB\_1
- Messwert, normierter Wert M\_ME\_NA\_1, M\_ME\_TA\_1, M\_ME\_ND\_1 und M\_ME\_TD\_1
- Messwert, skaliertes Wert M\_ME\_NB\_1, M\_ME\_TB\_1 und M\_ME\_TE\_1
- Messwert, verkürzte Gleitkommazahl M\_ME\_NC\_1, M\_ME\_TC\_1 und M\_ME\_TF\_1



*Tab. 52: Stationsabfrage*

<input checked="" type="checkbox"/>	Global
<input checked="" type="checkbox"/>	Gruppe 1
<input checked="" type="checkbox"/>	Gruppe 2
<input checked="" type="checkbox"/>	Gruppe 3
<input checked="" type="checkbox"/>	Gruppe 4
<input checked="" type="checkbox"/>	Gruppe 5
<input checked="" type="checkbox"/>	Gruppe 6
<input checked="" type="checkbox"/>	Gruppe 7
<input checked="" type="checkbox"/>	Gruppe 8
<input checked="" type="checkbox"/>	Gruppe 9
<input checked="" type="checkbox"/>	Gruppe 10
<input checked="" type="checkbox"/>	Gruppe 11
<input checked="" type="checkbox"/>	Gruppe 12
<input checked="" type="checkbox"/>	Gruppe 13
<input checked="" type="checkbox"/>	Gruppe 14
<input checked="" type="checkbox"/>	Gruppe 15
<input checked="" type="checkbox"/>	Gruppe 16

Die Zuweisung der Adressen der Informationsobjekte zu jeder einzelnen Gruppe muss in einer getrennten Tabelle festgelegt werden.

*Tab. 53: Uhrzeitsynchronisation*

<input checked="" type="checkbox"/>	Uhrzeitsynchronisation
-------------------------------------	------------------------

Tab. 54: Befehlsübertragung

<input checked="" type="checkbox"/>	Direkte Befehlsübertragung
<input checked="" type="checkbox"/>	Direkte Sollwert-Befehlsübertragung
<input type="checkbox"/>	Befehl .Anwahl und Ausführung
<input type="checkbox"/>	Sollwertbefehl .Anwahl und Ausführung
<input checked="" type="checkbox"/>	C_SE ACTTERM angewendet
<input checked="" type="checkbox"/>	Keine zusätzliche Festlegung
<input type="checkbox"/>	Kurze Befehlsausführungsdauer(Ausführungsdauer durch einen Systemparameter in Unterstation bestimmt)
<input type="checkbox"/>	Lange Befehlsausführungsdauer(Ausführungsdauer durch einen Systemparameter in Unterstation bestimmt)
<input type="checkbox"/>	Dauerbefehl
<input type="checkbox"/>	Überwachung der maximalen Verzögerung von Befehlen und Sollwertbefehlen in Befehlsrichtung
<input type="checkbox"/>	Maximal zulässige Verzögerung von Befehlen und Sollwertbefehlen

Tab. 55: Übertragung von Zählwerten

<input checked="" type="checkbox"/>	Zählerabfrage
<input checked="" type="checkbox"/>	Zähler umspeichern ohne Rücksetzen
<input checked="" type="checkbox"/>	Zähler umspeichern mit Rücksetzen
<input checked="" type="checkbox"/>	Zähler rücksetzen
<input checked="" type="checkbox"/>	Allgemeine Zählerabfrage
<input checked="" type="checkbox"/>	Zählerabfrage Gruppe 1
<input checked="" type="checkbox"/>	Zählerabfrage Gruppe 2
<input checked="" type="checkbox"/>	Zählerabfrage Gruppe 3
<input checked="" type="checkbox"/>	Zählerabfrage Gruppe 4
<input checked="" type="checkbox"/>	Modus A: Örtliches Umspeichern mit spontaner Übertragung
<input checked="" type="checkbox"/>	Modus B: Örtliches Umspeichern mit Zählerabfrage
<input type="checkbox"/>	Modus C: Umspeichern und Übertragen durch Zähler-Abfrage bei Umspeichern und Übertragen durch Zählerabfragebefehle
<input checked="" type="checkbox"/>	Modus D: Umspeichern durch Zähler-Abfragebefehl, umgespeicherte Werte werden spontan übertragen

Tab. 56: Laden eines Parameters

<input type="checkbox"/>	Schwellenwert
<input type="checkbox"/>	Glättungsfaktor
<input type="checkbox"/>	Unterer Grenzwert für Messwertübertragung
<input type="checkbox"/>	Oberer Grenzwert für Messwertübertragung

Tab. 57: Parameter für Aktivierung

Act/deact der zyklischen oder periodischen Übertragung des adressierten Objekts

Tab. 58: Prüfprozedur

Prüfprozedur

**Dateiübermittlung**

Dateiübermittlung in Überwachungsrichtung:

- Transparente Datei
- Übermittlung von Ereignisfolgen
- Übermittlung von Störfalldaten aus Schutzeinrichtungen
- Übermittlung von Folgen aufgezeichneter Analogwerte

Dateiübermittlung in Steuerungsrichtung

Transparente Datei

Tab. 59: Hintergrundabfrage

Hintergrundabfrage

Tab. 60: Telegrammlaufzeit-Erfassung

Telegrammlaufzeit-Erfassung

**Festlegungen für Zeitüberwachungen**

Parameter	Falls kein anderer Wert festgelegt	Bemerkungen	Ausgewählter Wert
t0	30s	Zeitüberwachung für die Verbindungsherstellung	
t1	15s	Zeitüberwachung für die Verbindungsherstellung	
t2	10s	Zeitüberwachung für Quittierungen, falls keine Datentelegramme übertragen werden $t_2 < t_1$	
t3	20s	Zeitüberwachung für gesendete Testtelegramme im Falle langer Ruhezustände	

Maximalbereich aller Zeitüberwachungswerte: 1 bis 255 s, Genauigkeit 1 s.

**Maximale Anzahl k der unquittierten APDU im I Format und späteste APDU-Quittierung (w) dek**

Parameter	Falls kein anderer Wert festgelegt	Bemerkungen	Ausgewählter Wert
k	12 APDU	Maximale Differenz Anzahl der Empfangsfolgen zur Anzahl der Sendefolgen	
w	8 APDU	Späteste Quittierung nach Empfang von wAPDU im I-Format	

Maximaler Wertebereich k: 1 bis 32767 ( $2^{15} - 1$ )APDU, Genauigkeit 1APDU.

Maximaler Wertebereich w: 1 bis 32767APDU, Genauigkeit 1APDU (w sollte Zweidrittel von k nicht überschreiten).

**Portnummer**

Parameter	Wert	Bemerkungen
Portnummer	2404	in allen Fällen

**RFC-2200-Sammlung**

RFC 2200 ist ein offizieller Internet-Standard, der den Stand der Normung im Internet angewandeter Protokolle beschreibt, wie sie durch das Internet Architecture Board (IAB) festgelegt sind. Es bietet ein breites Spektrum aktueller, im Internet angewandeter Standards. Die geeignete Auswahl in der vorliegenden Norm festgelegter Dokumente aus RFC 2200 für vorgegebene Projekte ist durch den Anwender dieser Norm auszuwählen.

- Ethernet 802.3
- Serielle Schnittstelle X.21
- Andere Auswahl aus RFC 2200
- Liste der anzuwendenden RFC-2200-Dokumente
- 1.
  - 2.
  - 3.
  - 4.
  - 5.
  - 6.
  7. usw.

**Anhang****2.2.5.1 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](http://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**

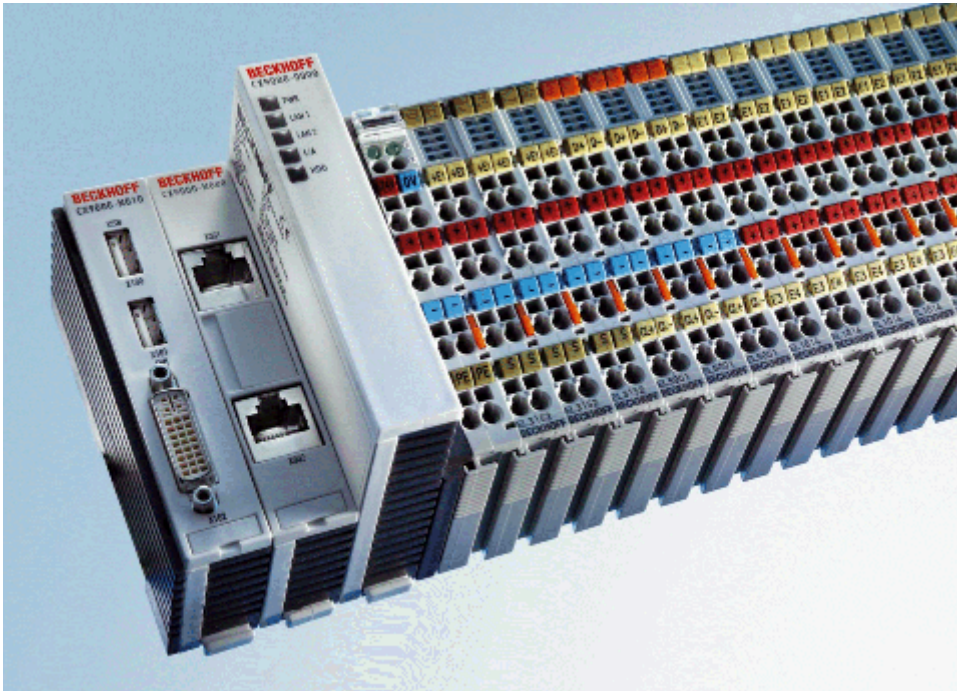
Beckhoff Automation GmbH & Co. KG

Hülshorstweg 20  
33415 Verl  
Deutschland

Telefon: +49 5246 963-0  
E-Mail: info@beckhoff.com  
Internet: [www.beckhoff.com](http://www.beckhoff.com)

## 2.2.6 Kompatibilitätsliste nach IEC 60870-5-104 (slave)

für TwinCAT SPS Bibliothek: IEC 60870-5-104 Unterstation (slave).



Version: 1.5

Datum: 07.03.2014

Hier können Sie die Kompatibilitätsliste öffnen/abspeichern (gezipptes PDF): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/984598155.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/984598155.zip)

### Inhaltsverzeichnis:

- [Allgemeine Informationen \[▶ 87\]](#)
  - [Projektinformationen \[▶ 87\]](#)
- [Kompatibilität \[▶ 87\]](#)
  - [System oder Gerät \[▶ 88\]](#)
  - [Netzkonfiguration \[▶ 88\]](#)
  - [Physikalische Schicht \[▶ 88\]](#)
  - [Verbindungsschicht \[▶ 88\]](#)
  - [Anwendungsschicht \[▶ 88\]](#)
  - [Auswahl genormter ASDUs \[▶ 90\]](#)

Prozessinformation in Überwachungsrichtung

Prozessinformation in Steuerungsrichtung

Systeminformation in Überwachungsrichtung

Systeminformation in Steuerungsrichtung

Parameter in Steuerungsrichtung

Dateiübermittlung

- [Grundlegende Anwendungsfunktionen \[▶ 98\]](#)
  - [Stationsinitialisierung \[▶ 98\]](#)

- Zyklische Datenübertragung
- Abrufprozedur
- Spontane Übertragung
- Doppelübertragung
- Stationsabfrage
- Uhrzeitsynchronisation
- Befehlsübertragung
- Zählerabfrage
- Laden eines Parameters
- Parameter für Aktivierung
- Prüfprozedur
- Dateiübermittlung [▶ 101]
- Hintergrundabfrage
- Telegrammlaufzeiterfassung
- Festlegungen für Zeitüberwachung [▶ 101]
- k- und w-Parameter
- Portnummer [▶ 87]
- RFC-2200-Sammlung [▶ 102]
- Anhang [▶ 102]
  - Support und Service [▶ 102]

**Allgemeine Informationen**

In dieser Kompatibilitätsliste finden Sie die unterstützten Funktionen innerhalb der IEC 870-5-104.



Dieses Kontrollfeld bedeutet, dass die Funktion bei der TCP/IP-Kopplung über IEC 870-5-104 nicht verwendet wird.

Die Kompatibilität der einzelnen Komponenten muss von allen Beteiligten für das jeweilige Projekt vereinbart werden.

**Projektinformationen**

Projekt-Nr.:  
 Projekt:  
 Zuständig:  
 Datum:

**Kompatibilität**

Die ausgewählten Parameter sollten in den weißen Kontrollfeldern wie folgt markiert werden:



Funktion oder ASDU wird nicht benutzt



Funktion oder ASDU wird wie genormt benutzt (Vorzugswert)



Funktion oder ASDU wird im Umkehrmodus benutzt



Funktion oder ASDU wird im Regel- und Umkehrmodus benutzt

**System oder Gerät**

- Systemfestlegung
- Festlegungen für die Zentralstation
- Festlegungen für die Unterstation (SLAVE)

**Netzkonfiguration**

- |                          |   |                          |   |
|--------------------------|---|--------------------------|---|
| <input type="checkbox"/> | End-End-Konfiguration[en: Point-to-point]                       | <input type="checkbox"/> | Linienkonfiguration<br>[en: Multipoint-partyline] |
| <input type="checkbox"/> | Mehrfach-End-End-Konfiguration<br>[en: Multiple point-to-point] | <input type="checkbox"/> | Sternkonfiguration<br>[en: Multipoint-star]       |

**Physikalische Schicht***Übertragungsgeschwindigkeit:*

- |                          |            |                          |             |                          |             |
|--------------------------|------------|--------------------------|-------------|--------------------------|-------------|
| <input type="checkbox"/> | 100 bit/s  | <input type="checkbox"/> | 2400 bit/s  | <input type="checkbox"/> | 56000 bit/s |
| <input type="checkbox"/> | 200 bit/s  | <input type="checkbox"/> | 4800 bit/s  | <input type="checkbox"/> | 64000 bit/s |
| <input type="checkbox"/> | 300 bit/s  | <input type="checkbox"/> | 9600 bit/s  |                          |             |
| <input type="checkbox"/> | 600 bit/s  | <input type="checkbox"/> | 19200 bit/s |                          |             |
| <input type="checkbox"/> | 1200 bit/s | <input type="checkbox"/> | 38400 bit/s |                          |             |

**Verbindungsschicht**

Nach dieser Anwendungsbezogenen Norm werden ausschließlich Telegrammformat FT 1.2, Einzelzeichen 1 und das feste Zeitüberwachungsintervall benutzt.

*Adressfeld der Verbindungsschicht:*

- Nicht vorhanden (nur symmetrische Übertragung)
- Ein Oktett
- Zwei Oktette
- Strukturiert
- Unstrukturiert

*Übertragungsverfahren der Verbindungsschicht:*

- Symmetrische Übertragung
- Unsymmetrische Übertragung

*Telegrammlänge:*

- Maximale Länge L

**Anwendungsschicht****Übertragungsbetriebsart für Anwendungsdaten**



Nach dieser Anwendungsbezogenen Norm wird ausschließlich Betriebsart 1 (niedrigstwertiges Oktett zuerst) nach 4.10 von IEC 60870-5-4 benutzt.

**Gemeinsame Adresse der ASDU**

<input type="checkbox"/>	Ein Oktett	<input checked="" type="checkbox"/>	Zwei Oktette
--------------------------	------------	-------------------------------------	--------------

**Adresse des Informationsobjekts**

<input type="checkbox"/>	Ein Oktett	<input checked="" type="checkbox"/>	Strukturiert
<input type="checkbox"/>	Zwei Oktette	<input checked="" type="checkbox"/>	Unstrukturiert
<input checked="" type="checkbox"/>	Drei Oktette		

**Übertragungsursache**

<input type="checkbox"/>	Ein Oktett	<input checked="" type="checkbox"/>	Zwei Oktette (mit Herkunftsadresse). Mit 0 vorbesetzt, falls Herkunftsadresse nicht vorhanden.
--------------------------	------------	-------------------------------------	---

**Länge der APDU**

(systembezogener Parameter, die maximale Länge der APDU je System ist festzulegen)  
Die maximale Länge der APDU beträgt 253 (falls nicht anders festgelegt). Je System darf die maximale Länge reduziert werden.

253            Maximale Länge der APDU je System

**Auswahl genormter ASDU**

*Tab. 61: Prozessinformation in Überwachungsrichtung*

<input checked="" type="checkbox"/>	<1> := Einzelmeldung	M_SP_NA_1
<input type="checkbox"/>	<2> := Einzelmeldung mit Zeitmarke	M_SP_TA_1
<input checked="" type="checkbox"/>	<3> := Doppelmeldung	M_DP_NA_1
<input type="checkbox"/>	<4> := Doppelmeldung mit Zeitmarke	M_DP_TA_1
<input checked="" type="checkbox"/>	<5> := Stufenstellungsmeldung	M_ST_NA_1
<input type="checkbox"/>	<6> := Stufenstellungsmeldung mit Zeitmarke	M_ST_TA_1
<input checked="" type="checkbox"/>	<7> := Bitmuster von 32 bit	M_BO_NA_1
<input type="checkbox"/>	<8> := Bitmuster von 32 bit mit Zeitmarke	M_BO_TA_1
<input checked="" type="checkbox"/>	<9> := Messwert, normierter Wert	M_ME_NA_1
<input type="checkbox"/>	<10> := Messwert, normierter Wert mit Zeitmarke	M_ME_TA_1
<input checked="" type="checkbox"/>	<11> := Messwert, skaliertes Wert	M_ME_NB_1
<input type="checkbox"/>	<12> := Messwert, skaliertes Wert mit Zeitmarke	M_ME_TB_1
<input checked="" type="checkbox"/>	<13> := Messwert, verkürzte Gleitkommazahl	M_ME_NC_1
<input type="checkbox"/>	<14> := Messwert, verkürzte Gleitkommazahl mit Zeitmarke	M_ME_TC_1
<input checked="" type="checkbox"/>	<15> := Zählwerte	M_IT_NA_1
<input type="checkbox"/>	<16> := Zählwerte mit Zeitmarke	M_IT_TA_1
<input type="checkbox"/>	<17> := Schutzereignis mit Zeitmarke	M_EP_TA_1
<input type="checkbox"/>	<18> := Geblockte Anregungen des Schutzes mit Zeitmarke	M_EP_TB_1
<input type="checkbox"/>	<19> := Geblockte Auslösungen des Schutzes mit Zeitmarke	M_EP_TC_1
<input checked="" type="checkbox"/>	<20> := Geblockte Einzelmeldungen mit Zustandsanzeige	M_PS_NA_1
<input checked="" type="checkbox"/>	<21> := Messwert, normierter Wert ohne Qualitätskennung	M_ME_ND_1
<input checked="" type="checkbox"/>	<30> := Einzelmeldung mit Zeitmarke CP56Time2a	M_SP_TB_1
<input checked="" type="checkbox"/>	<31> := Doppelmeldung mit Zeitmarke CP56Time2a	M_DP_TB_1
<input checked="" type="checkbox"/>	<32> := Stufenstellungsmeldung mit Zeitmarke CP56Time2a	M_ST_TB_1
<input checked="" type="checkbox"/>	<33> := Bitmuster von 32 bit mit Zeitmarke CP56Time2a	M_BO_TB_1
<input checked="" type="checkbox"/>	<34> := Messwert, normierter Wert mit Zeitmarke CP56Time2a	M_ME_TD_1
<input checked="" type="checkbox"/>	<35> := Messwert, skaliertes Wert mit Zeitmarke CP56Time2a	M_ME_TE_1
<input checked="" type="checkbox"/>	<36> := Messwert, verkürzte Gleitkommazahl mit Zeitmarke CP56Time2a	M_ME_TF_1
<input checked="" type="checkbox"/>	<37> := Zählwerte mit Zeitmarke CP56Time2a	M_IT_TB_1
<input checked="" type="checkbox"/>	<38> := Schutzereignis mit Zeitmarke CP56Time2a	M_EP_TD_1
<input checked="" type="checkbox"/>	<39> := Geblockte Anregungen des Schutzes mit Zeitmarke CP56Time2a	M_EP_TE_1

<input checked="" type="checkbox"/>	<40> := Geblockte Auslösungen des Schutzes mit Zeitmarke CP56Time2a	M_EP_TF_1
-------------------------------------	---	-----------

Tab. 62: Prozessinformation in Steuerungsrichtung

<input checked="" type="checkbox"/>	<45> := Einzelbefehl	C_SC_NA_1
<input checked="" type="checkbox"/>	<46> := Doppelbefehl	C_DC_NA_1
<input checked="" type="checkbox"/>	<47> := Stufenstellbefehl	C_RC_NA_1
<input checked="" type="checkbox"/>	<48> := Sollwert-Stellbefehl, normierter Wert	C_SE_NA_1
<input checked="" type="checkbox"/>	<49> := Sollwert-Stellbefehl, skaliertes Wert	C_SE_NB_1
<input checked="" type="checkbox"/>	<50> := Sollwert-Stellbefehl, verkürzte Gleitkommazahl	C_SE_NC_1
<input checked="" type="checkbox"/>	<51> := Bitmuster von 32 bit	C_BO_NA_1
<input checked="" type="checkbox"/>	<58> := Einzelbefehl mit Zeitmarke CP56Time2a	C_SC_TA_1
<input checked="" type="checkbox"/>	<59> := Doppelbefehl mit Zeitmarke CP56Time2a	C_DC_TA_1
<input checked="" type="checkbox"/>	<60> := Stufenstellbefehl mit Zeitmarke CP56Time2a	C_RC_TA_1
<input checked="" type="checkbox"/>	<61> := Sollwert-Stellbefehl, normierter Wert mit Zeitmarke CP56Time2a	C_SE_TA_1
<input checked="" type="checkbox"/>	<62> := Sollwert-Stellbefehl, skaliertes Wert mit Zeitmarke CP56Time2a	C_SE_TB_1
<input checked="" type="checkbox"/>	<63> := Sollwert-Stellbefehl, verkürzte Gleitkommazahl mit Zeitmarke CP56Time2a	C_SE_TC_1
<input checked="" type="checkbox"/>	<64> := Bitmuster von 32 bit mit Zeitmarke CP56Time2a	C_BO_TA_1

Tab. 63: Systeminformation in Überwachungsrichtung

<input checked="" type="checkbox"/>	<70> := Initialisierungsende	M_EI_NA_1
-------------------------------------	------------------------------	-----------

Tab. 64: Systeminformation in Steuerungsrichtung

<input checked="" type="checkbox"/>	<100>:= (Stations-)Abfragebefehl	C_IC_NA_1
<input checked="" type="checkbox"/>	<101>:= Zählerabfragebefehl	C_CI_NA_1
<input checked="" type="checkbox"/>	<102>:= Abfragebefehl (Lesebefehl)	C_RD_NA_1
<input checked="" type="checkbox"/>	<103>:= Uhrzeit-Synchronisationsbefehl	C_CS_NA_1
<input type="checkbox"/>	<104>:= Prüfbefehl	C_TS_NA_1
<input checked="" type="checkbox"/>	<105>:= Prozess-Rücksetzbefehl	C_RP_NA_1
<input type="checkbox"/>	<106>:= Befehl zur Telegrammlaufzeit-Erfassung	C_CD_NA_1
<input checked="" type="checkbox"/>	<107>:= Prüfbefehl mit Zeitmarke CP56Time2a	C_TS_TA_1

*Tab. 65: Parameter in Steuerungsrichtung*

<input type="checkbox"/>	<110>:= Parameter für Messwerte, normierter Wert	P_ME_NA_1
<input type="checkbox"/>	<111>:= Parameter für Messwerte, skaliertes Wert	P_ME_NB_1
<input type="checkbox"/>	<112>:= Parameter für Messwerte, verkürzte Gleitkommazahl	P_ME_NC_1
<input type="checkbox"/>	<113>:= Parameter für Aktivierung	P_AC_NA_1

*Tab. 66: Dateiübermittlung*

<input type="checkbox"/>	<120>:= Datei bereit	F_FR_NA_1
<input type="checkbox"/>	<121>:= Abschnitt bereit	F_SR_NA_1
<input type="checkbox"/>	<122>:= Abfrage Dateiverzeichnis, -auswahl, -abfrage, Abschnittsabfrage	F_SC_NA_1
<input type="checkbox"/>	<123>:= Letzter Abschnitt, letztes Segment	F_LS_NA_1
<input type="checkbox"/>	<124>:= Dateibestätigung, Abschnittsbestätigung	F_AF_NA_1
<input type="checkbox"/>	<125>:= Segment	F_SG_NA_1
<input type="checkbox"/>	<126>:= Dateiverzeichnis[leer oder <b>X</b> , nur in Überwachungsrichtung verfügbar (regulär)]	F_DR_TA_1

**Zuweisungen der Übertragungsursachen zu den Typkennungen(stationsbezogene Parameter)**

Markierung der Kombination Typkennung / Übertragungsursache mit:

Schattiert	Funktion oder ASDU wird nicht benötigt
Leer	Funktion oder ASDU wird nicht angewendet
<input checked="" type="checkbox"/>	Nur in der Normrichtung angewendet
<input checked="" type="checkbox"/>	Nur in der Umkehrrichtung angewendet
<input checked="" type="checkbox"/>	In beiden Richtungen angewendet

Tab. 67: Übertragungsursachen

Nummer	Übertragungsursache
<0>	Not used
<1>	Periodic, cyclic
<2>	Background scan
<3>	Spontaneous
<4>	Initialized
<5>	Request or requested
<6>	Activation (ACT)
<7>	Activation confirmation (ACTCON)
<8>	Deactivation (DEACT)
<9>	Deactivation confirmation (DEACTCON)
<10>	Activation termination (ACTTERM)
<11>	Return information caused by a remote command
<12>	Return information caused by a local command
<13>	File transfer
<20..36>	General interrogation
<37..41>	Counter interrogation
<44>	Unknown type identification
<45>	Unknown cause of transmission
<46>	Unknown address of ASDU
<47>	Unknown information object address

Typkennung		Übertragungsursache																		
		<1>	<2>	<3>	<4>	<5>	<6>	<7>	<8>	<9>	<10>	<11>	<12>	<13>	<20..36>	<37..41>	<44>	<45>	<46>	<47>
		per/cyc	back	spon	init	req	act	actcon	deact	deactcon	actterm	retrem	retloc	file	inro	reqco	unknown type	unknown COT	unknown CAA	unknown IOA
<1>	M_S P_N A_1		X	X		X						X			X					
<2>	M_S P_T A_1																			
<3>	M_D P_N A_1		X	X		X					X				X					
<4>	M_D P_T A_1																			
<5>	M_S T_N A_1		X	X		X					X				X					
<6>	M_S T_TA _1																			
<7>	M_B O_N A_1		X	X		X									X					
<8>	M_B O_T A_1																			
<9>	M_M E_N A_1	X	X	X		X									X					
<10>	M_M E_T A_1																			
<11>	M_M E_N B_1	X	X	X		X									X					
<12>	M_M E_T B_1																			
<13>	M_M E_N C_1	X	X	X		X									X					
<14>	M_M E_T C_1																			
<15>	M_IT _NA _1			X												X				
<16>	M_IT _TA _1																			
<17>	M_E P_T A_1																			
<18>	M_E P_T B_1																			

Typkennung	Übertragungsursache																
<19> M_E P_T C_1																	
<20> M_P S_N A_1		X	X		X				X			X					
<21> M_M E_N D_1	X	X	X		X							X					
<30> M_S P_T B_1			X		X				X								
<31> M_D P_T B_1			X		X				X								
<32> M_S T_TB _1			X		X				X								
<33> M_B O_T B_1			X		X												
<34> M_M E_T D_1			X		X												
<35> M_M E_T E_1			X		X												
<36> M_M E_TF _1			X														
<37> M_IT _TB _1			X										X				
<38> M_E P_T D_1			X														
<39> M_E P_T E_1			X														
<40> M_E P_TF _1			X														
<45> C_S C_N A_1						X	X	X	X	X				X	X	X	X
<46> C_D C_N A_1						X	X	X	X	X				X	X	X	X
<47> C_R C_N A_1						X	X	X	X	X				X	X	X	X
<48> C_S E_N A_1						X	X	X	X	X				X	X	X	X
<49> C_S E_N B_1						X	X	X	X	X				X	X	X	X
<50> C_S E_N C_1						X	X	X	X	X				X	X	X	X
<51> C_B O_N A_1						X	X	X	X	X				X	X	X	X
<58> C_S C_T A_1						X	X	X	X	X				X	X	X	X
<59> C_D C_T A_1						X	X	X	X	X				X	X	X	X



Typkennung		Übertragungsursache																		
<60>	C_R C_T A_1						X	X	X	X	X						X	X	X	X
<61>	C_S E_T A_1						X	X	X	X	X						X	X	X	X
<62>	C_S E_T B_1						X	X	X	X	X						X	X	X	X
<63>	C_S E_T C_1						X	X	X	X	X						X	X	X	X
<64>	C_B O_T A_1						X	X	X	X	X						X	X	X	X
<70>	M_EI _NA _1				X															
<100>	C_IC _NA _1						X	X	X	X	X						X	X	X	X
<101>	C_CI _NA _1						X	X			X						X	X	X	X
<102>	C_R D_N A_1				X												X	X	X	X
<103>	C_C S_N A_1			X			X	X									X	X	X	X
<104>	C_T S_N A_1																			
<105>	C_R P_N A_1						X	X									X	X	X	X
<106>	C_C D_N A_1																			
<107>	C_T S_T A_1						X	X									X	X	X	X
<110>	P_M E_N A_1																X	X	X	X
<111>	P_M E_N B_1																X	X	X	X
<112>	P_M E_N C_1																X	X	X	X
<113>	P_A C_N A_1																X	X	X	X
<120>	F_F R_N A_1																X	X	X	X
<121>	F_S R_N A_1																X	X	X	X
<122>	F_S C_N A_1																X	X	X	X
<123>	F_LS _NA _1																X	X	X	X
<124>	F_AF _NA _1																X	X	X	X

Typkennung		Übertragungsursache																		
<125 >	F_S G_N A_1																X	X	X	X
<126 >	F_D R_T A_1																			

**Grundlegende Anwendungsfunktionen**

**Stationsinitialisierung**

Remote initialization

*Tab. 68: Zyklische Datenübertragung*

Zyklische Datenübertragung

*Tab. 69: Abrufprozedur*

Abrufprozedur

*Tab. 70: Spontane Übertragung*

Spontane Übertragung

**Doppelübertragung von Informationsobjekten mit der Übertragungsursache spontan**

Die folgenden Typkennungen dürfen nacheinander in Folge eines einzigen Zustandswechsels eines Informationsobjekts, übertragen werden. Die einzelnen Adressen der Informationsobjekte, die für die Doppelübertragung vorgesehen sind, werden in einer projektbezogenen Liste festgelegt.

- Einzelmeldung M\_SP\_NA\_1, M\_SP\_TA\_1, M\_SP\_TB\_1 und M\_SP\_NA\_1
- Doppelmeldung M\_DP\_NA\_1, M\_DP\_TA\_1 und M\_DP\_TB\_1
- Stufenstellungsmeldung M\_ST\_NA\_1, M\_ST\_TA\_1 und M\_ST\_TB\_1
- Bitmuster von 32 bit M\_BO\_NA\_1, M\_BO\_TA\_1 und M\_BO\_TB\_1
- Messwert, normierter Wert M\_ME\_NA\_1, M\_ME\_TA\_1, M\_ME\_ND\_1 und M\_ME\_TD\_1
- Messwert, skaliertes Wert M\_ME\_NB\_1, M\_ME\_TB\_1 und M\_ME\_TE\_1
- Messwert, verkürzte Gleitkommazahl M\_ME\_NC\_1, M\_ME\_TC\_1 und M\_ME\_TF\_1

*Tab. 71: Stationsabfrage*

<input checked="" type="checkbox"/>	Global
<input checked="" type="checkbox"/>	Gruppe 1
<input checked="" type="checkbox"/>	Gruppe 2
<input checked="" type="checkbox"/>	Gruppe 3
<input checked="" type="checkbox"/>	Gruppe 4
<input checked="" type="checkbox"/>	Gruppe 5
<input checked="" type="checkbox"/>	Gruppe 6
<input checked="" type="checkbox"/>	Gruppe 7
<input checked="" type="checkbox"/>	Gruppe 8
<input checked="" type="checkbox"/>	Gruppe 9
<input checked="" type="checkbox"/>	Gruppe 10
<input checked="" type="checkbox"/>	Gruppe 11
<input checked="" type="checkbox"/>	Gruppe 12
<input checked="" type="checkbox"/>	Gruppe 13
<input checked="" type="checkbox"/>	Gruppe 14
<input checked="" type="checkbox"/>	Gruppe 15
<input checked="" type="checkbox"/>	Gruppe 16

Die Zuweisung der Adressen der Informationsobjekte zu jeder einzelnen Gruppe muss in einer getrennten Tabelle festgelegt werden.

*Tab. 72: Uhrzeitsynchronisation*

<input checked="" type="checkbox"/>	Uhrzeitsynchronisation
-------------------------------------	------------------------

Tab. 73: Befehlsübertragung

<input checked="" type="checkbox"/>	Direkte Befehlsübertragung
<input checked="" type="checkbox"/>	Direkte Sollwert-Befehlsübertragung
<input checked="" type="checkbox"/>	Befehl .Anwahl und Ausführung
<input checked="" type="checkbox"/>	Sollwertbefehl .Anwahl und Ausführung
<input checked="" type="checkbox"/>	C_SE ACTTERM angewendet
<input checked="" type="checkbox"/>	Keine zusätzliche Festlegung
<input type="checkbox"/>	Kurze Befehlsausführungsdauer(Ausführungsdauer durch einen Systemparameter in Unterstation bestimmt)
<input type="checkbox"/>	Lange Befehlsausführungsdauer(Ausführungsdauer durch einen Systemparameter in Unterstation bestimmt)
<input type="checkbox"/>	Dauerbefehl
<input type="checkbox"/>	Überwachung der maximalen Verzögerung von Befehlen und Sollwertbefehlen in Befehlsrichtung
<input type="checkbox"/>	Maximal zulässige Verzögerung von Befehlen und Sollwertbefehlen

Tab. 74: Übertragung von Zählwerten

<input checked="" type="checkbox"/>	Zählerabfrage
<input checked="" type="checkbox"/>	Zähler umspeichern ohne Rücksetzen
<input checked="" type="checkbox"/>	Zähler umspeichern mit Rücksetzen
<input checked="" type="checkbox"/>	Zähler rücksetzen
<input checked="" type="checkbox"/>	Allgemeine Zählerabfrage
<input checked="" type="checkbox"/>	Zählerabfrage Gruppe 1
<input checked="" type="checkbox"/>	Zählerabfrage Gruppe 2
<input checked="" type="checkbox"/>	Zählerabfrage Gruppe 3
<input checked="" type="checkbox"/>	Zählerabfrage Gruppe 4
<input checked="" type="checkbox"/>	Modus A: Örtliches Umspeichern mit spontaner Übertragung
<input checked="" type="checkbox"/>	Modus B: Örtliches Umspeichern mit Zählerabfrage
<input checked="" type="checkbox"/>	Modus C: Umspeichern und Übertragen durch Zähler-Abfrage bei Umspeichern und Übertragen durch Zählerabfragebefehle
<input checked="" type="checkbox"/>	Modus D: Umspeichern durch Zähler-Abfragebefehl, umgespeicherte Werte werden spontan übertragen

Tab. 75: Laden eines Parameters

<input type="checkbox"/>	Schwellenwert
<input type="checkbox"/>	Glättungsfaktor
<input type="checkbox"/>	Unterer Grenzwert für Messwertübertragung
<input type="checkbox"/>	Oberer Grenzwert für Messwertübertragung

Tab. 76: Parameter für Aktivierung

Act/deact der zyklischen oder periodischen Übertragung des adressierten Objekts

Tab. 77: Prüfprozedur

Prüfprozedur

**Dateiübermittlung**

Dateiübermittlung in Überwachungsrichtung:

- Transparente Datei
- Übermittlung von Ereignisfolgen
- Übermittlung von Störfalldaten aus Schutzeinrichtungen
- Übermittlung von Folgen aufgezeichneter Analogwerte

Dateiübermittlung in Steuerungsrichtung

Transparente Datei

Tab. 78: Hintergrundabfrage

Hintergrundabfrage

Tab. 79: Telegrammlaufzeit-Erfassung

Telegrammlaufzeit-Erfassung

**Festlegungen für Zeitüberwachungen**

Parameter	Falls kein anderer Wert festgelegt	Bemerkungen	Ausgewählter Wert
t0	30s	Zeitüberwachung für die Verbindungsherstellung	
t1	15s	Zeitüberwachung für die Verbindungsherstellung	
t2	10s	Zeitüberwachung für Quittierungen, falls keine Datentelegramme übertragen werden $t_2 < t_1$	
t3	20s	Zeitüberwachung für gesendete Testtelegramme im Falle langer Ruhezustände	

Maximalbereich aller Zeitüberwachungswerte: 1 bis 255 s, Genauigkeit 1 s.

**Maximale Anzahl k der unquittierten APDU im I Format und späteste APDU-Quittierung (w) dek**

Parameter	Falls kein anderer Wert festgelegt	Bemerkungen	Ausgewählter Wert
k	12 APDU	Maximale Differenz Anzahl der Empfangsfolgen zur Anzahl der Sendefolgen	
w	8 APDU	Späteste Quittierung nach Empfang von w APDU im I-Format	

Maximaler Wertebereich k: 1 bis 32767 (215 -1)APDU, Genauigkeit 1APDU.

Maximaler Wertebereich w: 1 bis 32767APDU, Genauigkeit 1APDU (w sollte Zweidrittel von k nicht überschreiten).

**Portnummer**

Parameter	Wert	Bemerkungen
Portnummer	2404	in allen Fällen

**RFC-2200-Sammlung**

RFC 2200 ist ein offizieller Internet-Standard, der den Stand der Normung im Internet angewendeter Protokolle beschreibt, wie sie durch das Internet Architecture Board (IAB) festgelegt sind. Es bietet ein breites Spektrum aktueller, im Internet angewendeter Standards. Die geeignete Auswahl in der vorliegenden Norm festgelegter Dokumente aus RFC 2200 für vorgegebene Projekte ist durch den Anwender dieser Norm auszuwählen.

- Ethernet 802.3
- Serielle Schnittstelle X.21
- Andere Auswahl aus RFC 2200
- Liste der anzuwendenden RFC-2200-Dokumente
- 1.
  - 2.
  - 3.
  - 4.
  - 5.
  - 6.
  7. usw.

**Anhang****2.2.6.1 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](http://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](http://www.beckhoff.com)

## 3 Installation

### 3.1 Systemvoraussetzungen

Im folgenden Artikel werden die Mindestanforderungen an die Entwicklungsumgebung- und/oder Laufzeitsystem beschrieben.

#### Entwicklungsumgebung

Eine reine Entwicklungsumgebung beschreibt einen Computer, auf dem SPS-Programme entwickelt, aber nicht ausgeführt werden. Auf einem Entwicklungsrechner muss folgendes installiert sein:

- TwinCAT 3.1.4018.0 XAE (Engineering) oder höher
- TwinCAT 3 Function TF6500
- Bitte beachten: Für die Entwicklungsumgebung kann (auch mehrfach) eine 7-Tage Testlizenz genutzt werden, siehe Lizenzierung

#### Laufzeitumgebung

Eine Laufzeitumgebung beschreibt einen Computer, auf dem SPS-Programme ausgeführt werden. Auf einem Laufzeitcomputer muss folgendes installiert sein:

- TwinCAT 3.1.4018.0 XAR (Runtime) oder höher
- TwinCAT 3 Function TF6500
- Lizenzen für TC1200 PLC und TF6500
- Bitte beachten: Für Testzwecke kann eine 7-Tage Testlizenz genutzt werden, siehe Lizenzierung

#### Entwickler- und Laufzeit-Version auf einem Computer

Sollen auf einem Computer Laufzeit- und Entwicklungsumgebung laufen (z.B. um ein SPS-Programm zu testen, bevor es auf den Ziel-Computer geladen wird), müssen folgende Anforderungen erfüllt sein:

- TwinCAT 3.1.4018.0 XAE (Engineering) oder höher
- TwinCAT 3 Function TF6500
- Lizenzen für TC1200 PLC und TF6500
- Bitte beachten: Für Testzwecke kann eine 7-Tage Testlizenz genutzt werden, siehe Lizenzierung

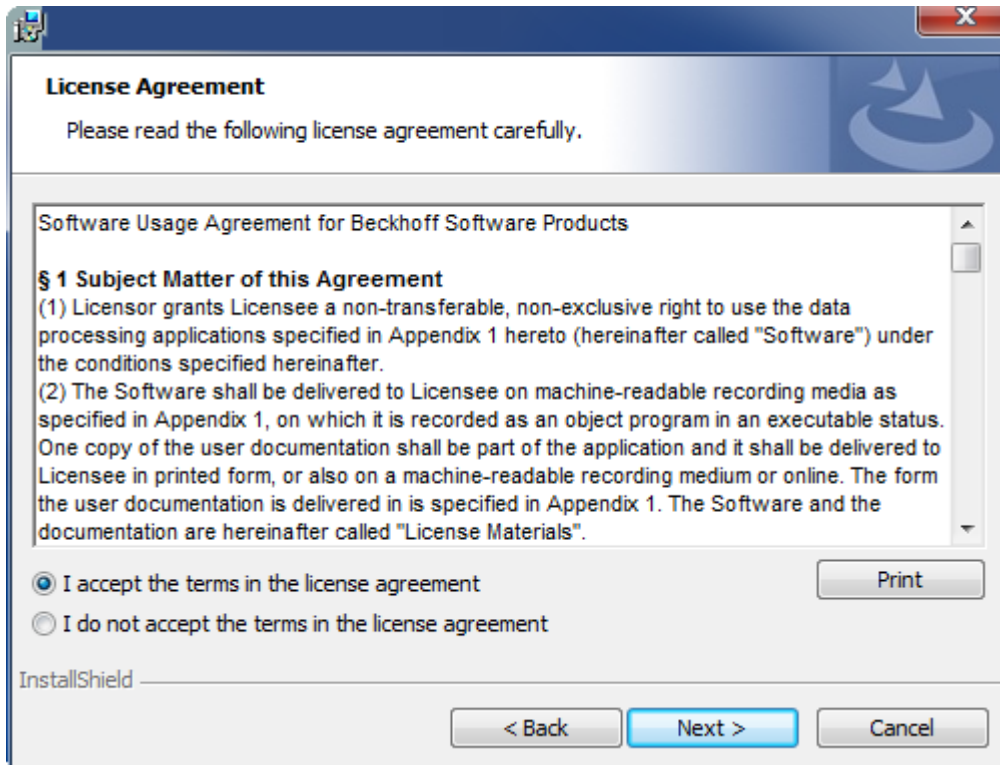
### 3.2 Installation

Nachfolgend wird beschrieben, wie die TwinCAT 3 Function für Windows-basierte Betriebssysteme installiert wird.

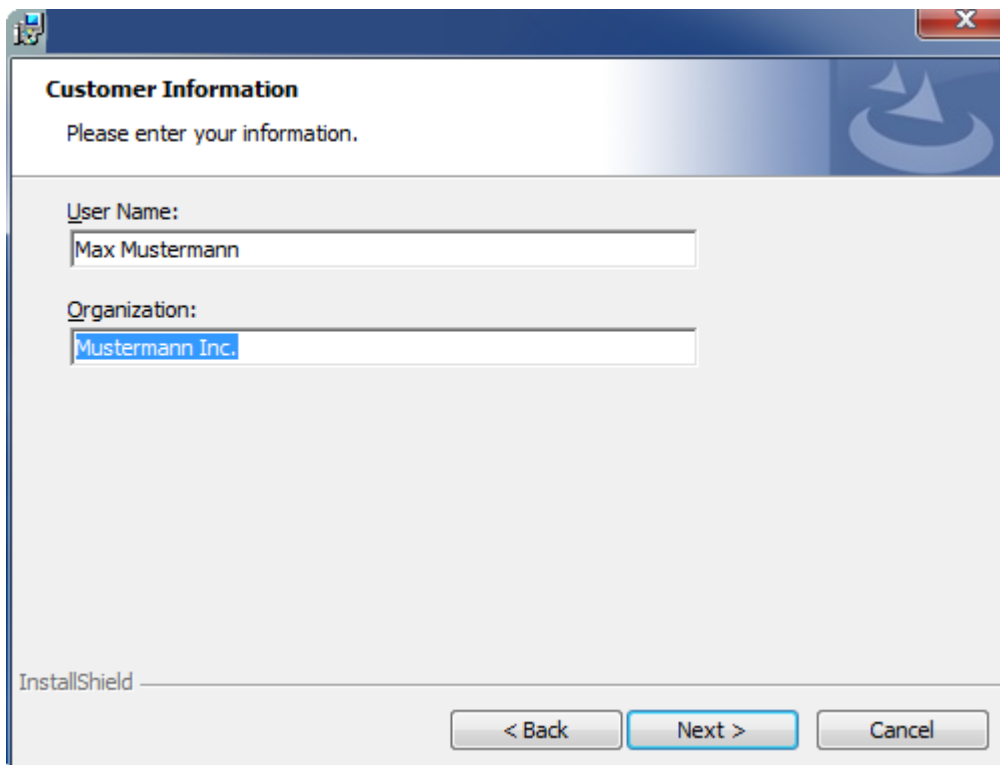
- ✓ Die Setup-Datei der TwinCAT 3 Function wurde von der Beckhoff-Homepage heruntergeladen.
1. Führen Sie die Setup-Datei als Administrator aus. Wählen Sie dazu im Kontextmenü der Datei den Befehl **Als Administrator ausführen**.
    - ⇒ Der Installationsdialog öffnet sich.



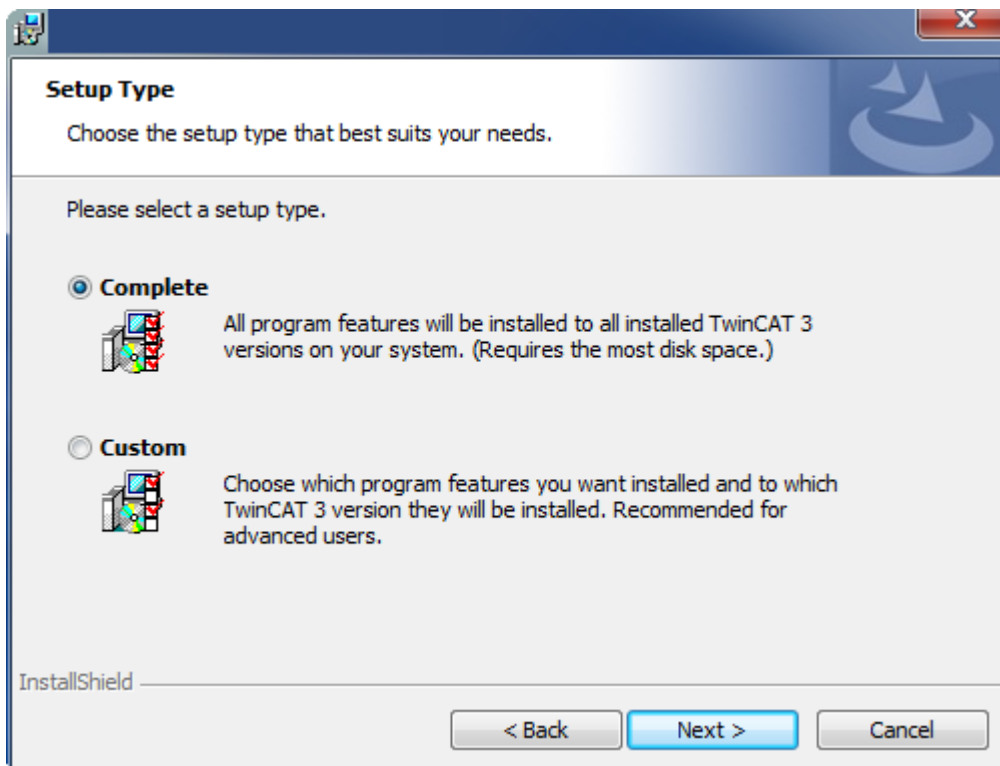
2. Akzeptieren Sie die Endbenutzerbedingungen und klicken Sie auf **Next**.



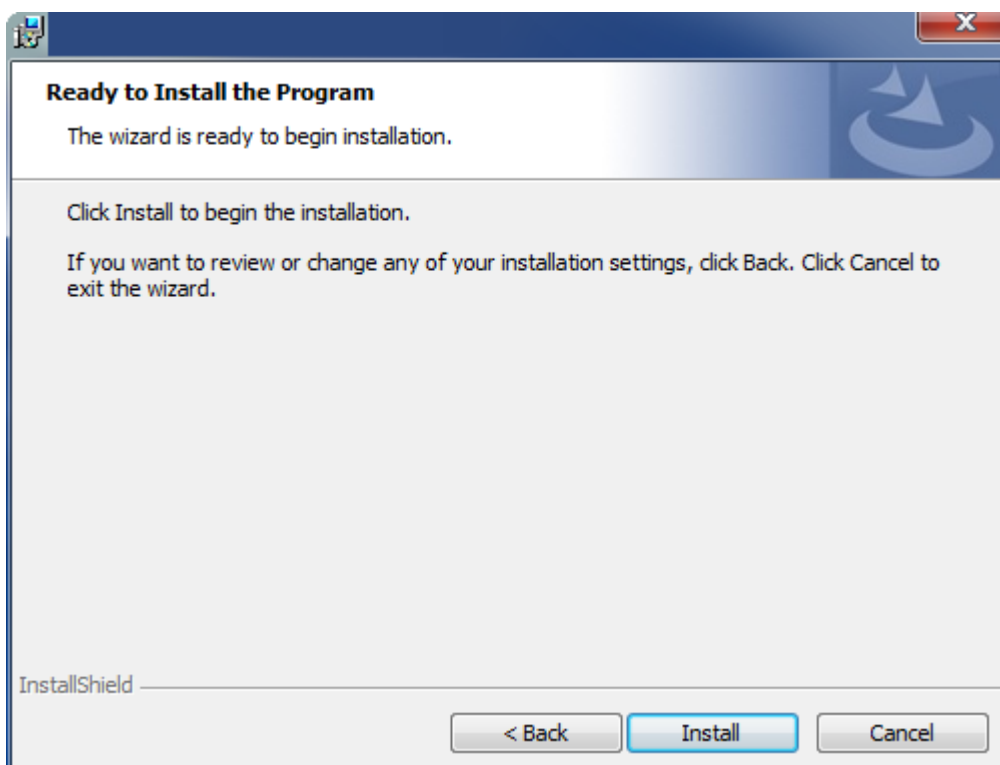
3. Geben Sie Ihre Benutzerdaten ein.



4. Wenn Sie die TwinCAT 3 Function vollständig installieren möchten, wählen Sie **Complete** als Installationstyp. Wenn Sie die Komponenten der TwinCAT 3 Function separat installieren möchten, wählen Sie **Custom**.

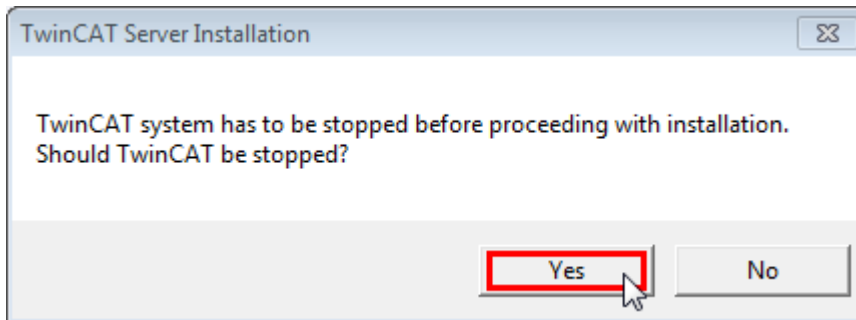


5. Wählen Sie **Next** und anschließend **Install**, um die Installation zu beginnen.

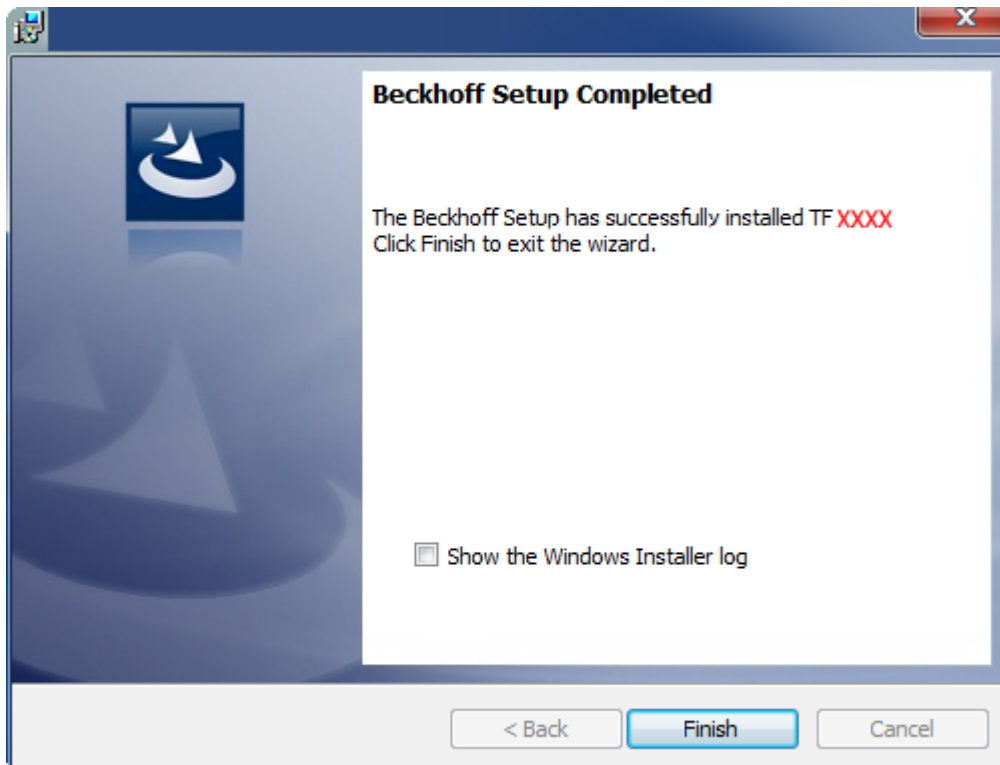


- ⇒ Ein Dialog weist Sie darauf hin, dass das TwinCAT-System für die weitere Installation gestoppt werden muss.

6. Bestätigen Sie den Dialog mit **Yes**.



7. Wählen Sie **Finish**, um das Setup zu beenden.



⇒ Die TwinCAT 3 Function wurde erfolgreich installiert.

### 3.3 Installation TwinCAT/BSD

Die Function TF6500 TwinCAT 3 IEC 60870-5-10x steht nicht direkt als Package für TwinCAT/BSD zur Verfügung. Relevant für den Betrieb unter TwinCAT/BSD ist das Package des TwinCAT 3 TCP/IP Servers. Dieses steht im TwinCAT/BSD Package-Repository bereit. Unter dem Package-Namen „TF6310-TCP-IP“ kann es über den Befehl installiert werden.

```
doas pkg install TF6310-TCP-IP
```

Weitere Informationen über den [Package Server](#) finden Sie im Bereich Embedded-PC im TwinCAT/BSD-Handbuch.

Nach einem Neustart des Systems oder Restart von TwinCAT, wird der TwinCAT 3 TCP/IP Server ebenfalls gestartet und ist entsprechend über das Fernwirkprotokoll nutzbar.

### 3.4 Installation Windows CE

Nachfolgend wird beschrieben, wie eine TwinCAT 3 Function (TFxxx) auf einem Beckhoff Embedded-PC mit Windows CE installiert wird.

1. [Download der Setup-Datei und Installation](#) [► 108]

2. [CAB-Datei auf das Windows-CE-Gerät übertragen \[► 108\]](#)
3. [CAB-Datei auf dem Windows-CE-Gerät ausführen \[► 108\]](#)

Wenn bereits eine ältere TFxxx-Version auf dem Windows-CE-Gerät installiert ist, kann diese aktualisiert werden:

- [Upgrade der Software \[► 109\]](#)

### Download der Setup-Datei und Installation

Die CAB-Installationsdatei für Windows CE ist Teil des TFxxx-Setups. Dieses wird Ihnen auf der Beckhoff-Homepage [www.beckhoff.com](http://www.beckhoff.com) zur Verfügung gestellt und enthält automatisch alle Versionen für Windows XP, Windows 7 und Windows CE (x86 und ARM).

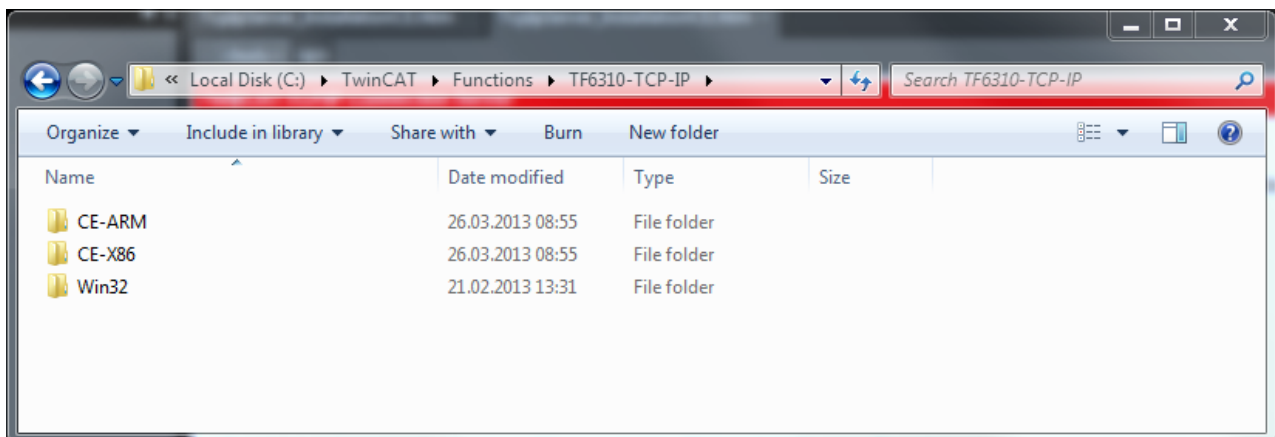
Laden Sie das TFxxx-Setup herunter und installieren Sie die TwinCAT 3 Function wie es im Abschnitt [Installation \[► 104\]](#) beschrieben wird.

Nach der Installation enthält der Installationsordner drei Verzeichnisse (pro Hardware-Plattform ein Verzeichnis):

- **CE-ARM:** ARM-basierte Embedded-PC, die unter Windows CE laufen, z. B. CX8090, CX9020
- **CE-X86:** X86-basierte Embedded-PC, die unter Windows CE laufen, z. B. CX50xx, CX20x0
- **Win32:** Embedded-PC, die unter Windows XP, Windows 7 oder Windows Embedded Standard laufen

Die Verzeichnisse CE-ARM und CE-X86 enthalten die CAB-Dateien der TwinCAT 3 Function für Windows CE in Bezug auf die jeweilige Hardware-Plattform des Windows-CE-Gerätes.

Beispiel: Installationsordner „TF6310“



### CAB-Datei auf das Windows-CE-Gerät übertragen

Übertragen Sie die entsprechende CAB-Datei auf das Windows-CE-Gerät.

Für die Übertragung der ausführbaren Datei stehen Ihnen verschiedene Möglichkeiten zur Verfügung:

- über Netzwerkfreigaben
- über den integrierten FTP-Server
- über ActiveSync
- über CF/SD-Karten

Weitere Informationen finden Sie im Beckhoff Information System in der Dokumentation „Betriebssysteme“ (Embedded-PC > Betriebssysteme > [CE](#)).

### CAB-Datei auf dem Windows-CE-Gerät ausführen

Nachdem Sie die CAB-Datei auf das Windows-CE-Gerät übertragen haben, führen Sie die Datei dort mit einem Doppelklick aus. Bestätigen Sie den Installationsdialog mit **OK**. Starten Sie das Windows-CE-Gerät anschließend neu.

Nach dem Neustart des Gerätes werden die Dateien der TwinCAT 3 Function (TFxxxx) automatisch im Hintergrund geladen und sind verfügbar.

Die Software wird in dem folgenden Verzeichnis auf dem Windows-CE-Gerät installiert:  
`Hard Disk\TwinCAT\Functions\TFxxxx`

### Upgrade der Software

Wenn auf dem Windows-CE-Gerät bereits eine ältere Version der TwinCAT 3 Function installiert ist, führen Sie die folgenden Schritte auf dem Windows-CE-Gerät durch, um ein Upgrade auf eine neue Version durchzuführen:

1. Öffnen Sie den CE Explorer, indem Sie auf **Start > Run** klicken und „Explorer“ eingeben.
  2. Navigieren Sie nach `Hard Disk\TwinCAT\Functions\TFxxx\xxxx`.
  3. Benennen Sie die Datei `Tc*.exe` in `Tc*.old` um.
  4. Starten Sie das Windows-CE-Gerät neu.
  5. Übertragen Sie die neue CAB-Datei auf das Windows-CE-Gerät.
  6. Führen Sie die CAB-Datei auf dem Windows-CE-Gerät aus und installieren Sie die neue Version.
  7. Löschen Sie die Datei `Tc*.old`.
  8. Starten Sie das Windows-CE-Gerät neu.
- ⇒ Nach dem Neustart ist die neue Version aktiv.

## 3.5 Lizenzierung

Die TwinCAT 3 Function ist als Vollversion oder als 7-Tage-Testversion freischaltbar. Beide Lizenztypen sind über die TwinCAT-3-Entwicklungsumgebung (XAE) aktivierbar.

### Lizenzierung der Vollversion einer TwinCAT 3 Function

Die Beschreibung der Lizenzierung einer Vollversion finden Sie im Beckhoff Information System in der Dokumentation „[TwinCAT-3-Lizenzierung](#)“.

### Lizenzierung der 7-Tage-Testversion einer TwinCAT 3 Function

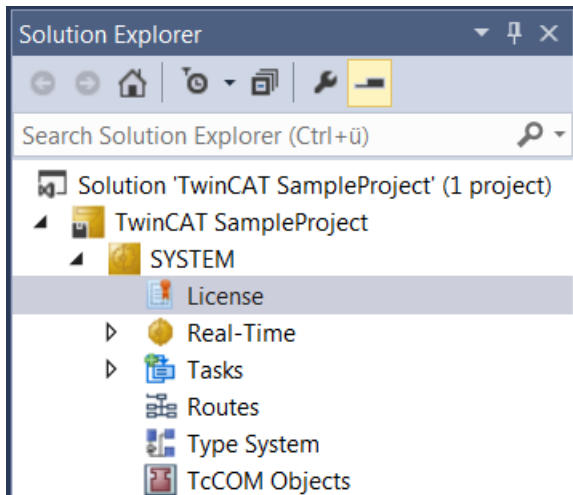


Eine 7-Tage-Testversion kann nicht für einen [TwinCAT-3-Lizenz-Dongle](#) freigeschaltet werden.

---

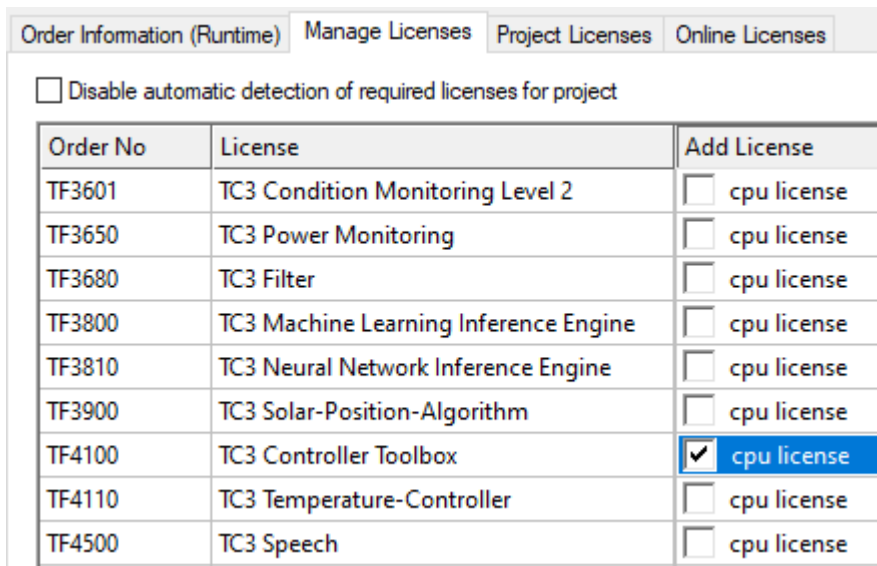
1. Starten Sie die TwinCAT-3-Entwicklungsumgebung (XAE).
2. Öffnen Sie ein bestehendes TwinCAT-3-Projekt oder legen Sie ein neues Projekt an.
3. Wenn Sie die Lizenz für ein Remote-Gerät aktivieren wollen, stellen Sie das gewünschte Zielsystem ein. Wählen Sie dazu in der Symbolleiste in der Drop-down-Liste **Choose Target System** das Zielsystem aus.
  - ⇒ Die Lizenzierungseinstellungen beziehen sich immer auf das eingestellte Zielsystem. Mit der Aktivierung des Projekts auf dem Zielsystem werden automatisch auch die zugehörigen TwinCAT-3-Lizenzen auf dieses System kopiert.

4. Klicken Sie im **Solution Explorer** im Teilbaum **SYSTEM** doppelt auf **License**.



⇒ Der TwinCAT-3-Lizenzmanager öffnet sich.

5. Öffnen Sie die Registerkarte **Manage Licenses**. Aktivieren Sie in der Spalte **Add License** das Auswahlkästchen für die Lizenz, die Sie Ihrem Projekt hinzufügen möchten (z. B. „TF4100 TC3 Controller Toolbox“).



6. Öffnen Sie die Registerkarte **Order Information (Runtime)**.

⇒ In der tabellarischen Übersicht der Lizenzen wird die zuvor ausgewählte Lizenz mit dem Status „missing“ angezeigt.

7. Klicken Sie auf **7 Days Trial License...**, um die 7-Tage-Testlizenz zu aktivieren.

The screenshot shows a software interface with several sections:

- Order Information (Runtime)**: Includes tabs for 'Manage Licenses', 'Project Licenses', and 'Online Licenses'. Below are fields for 'License Device' (set to 'Target (Hardware Id)'), 'System Id' (2DB25408-B4CD-81DF-5488-6A3D9B49EF19), and 'Platform' (other (91)).
- License Request**: Includes a 'Provider' dropdown set to 'Beckhoff Automation', 'License Id', 'Customer Id', and a 'Comment' field. A 'Generate File...' button is present.
- License Activation**: Contains two buttons: '7 Days Trial License...' (highlighted with a red box) and 'License Response File...'.

⇒ Es öffnet sich ein Dialog, der Sie auffordert, den im Dialog angezeigten Sicherheitscode einzugeben.

The dialog box is titled 'Enter Security Code' and contains the following elements:

- Text: 'Please type the following 5 characters:'
- Code display box: 'Kg8T4'
- Input field: A two-character input box with a red border, currently empty.
- Buttons: 'OK' (highlighted with a red box) and 'Cancel'.

8. Geben Sie den Code genauso ein, wie er angezeigt wird, und bestätigen Sie ihn.

9. Bestätigen Sie den nachfolgenden Dialog, der Sie auf die erfolgreiche Aktivierung hinweist.

⇒ In der tabellarischen Übersicht der Lizenzen gibt der Lizenzstatus nun das Ablaufdatum der Lizenz an.

10. Starten Sie das TwinCAT-System neu.

⇒ Die 7-Tage-Testversion ist freigeschaltet.

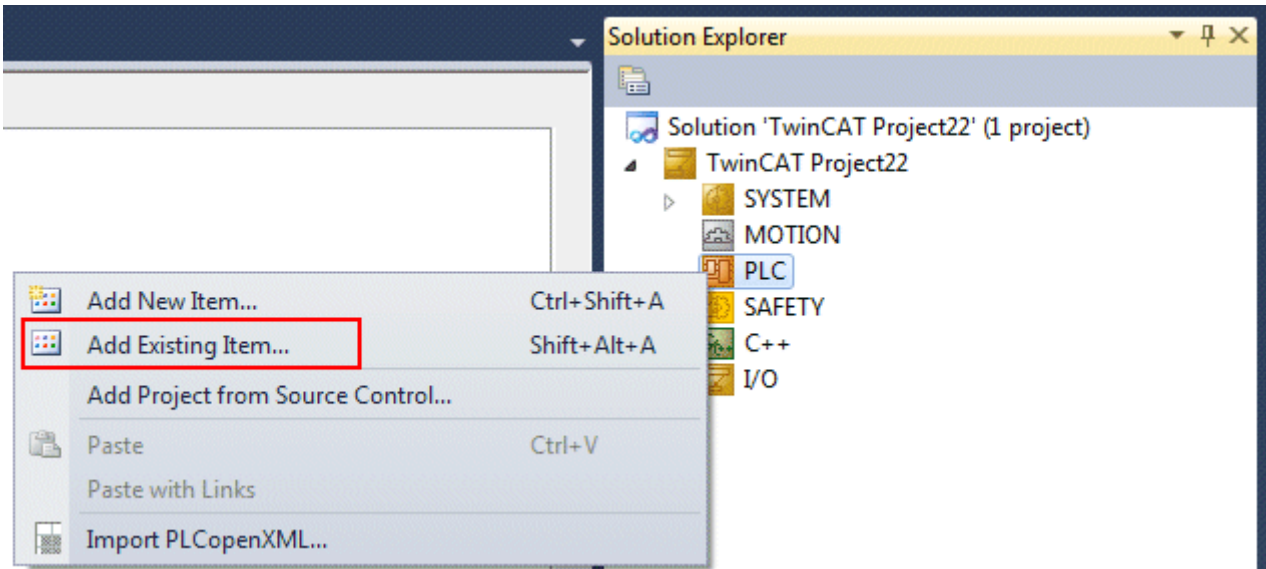
### 3.6 Migration von TwinCAT 2

Wenn Sie ein existierendes TwinCAT 2 PLC Projekt migrieren wollen, welches die IEC 60870-5-10x (Supplement) Bibliotheken der SPS nutzt, muss durch einige manuelle Schritte sichergestellt werden, dass der TwinCAT 3 SPS Konverter die Projektdatei aus TwinCAT2 (\*.pro) verarbeiten kann.

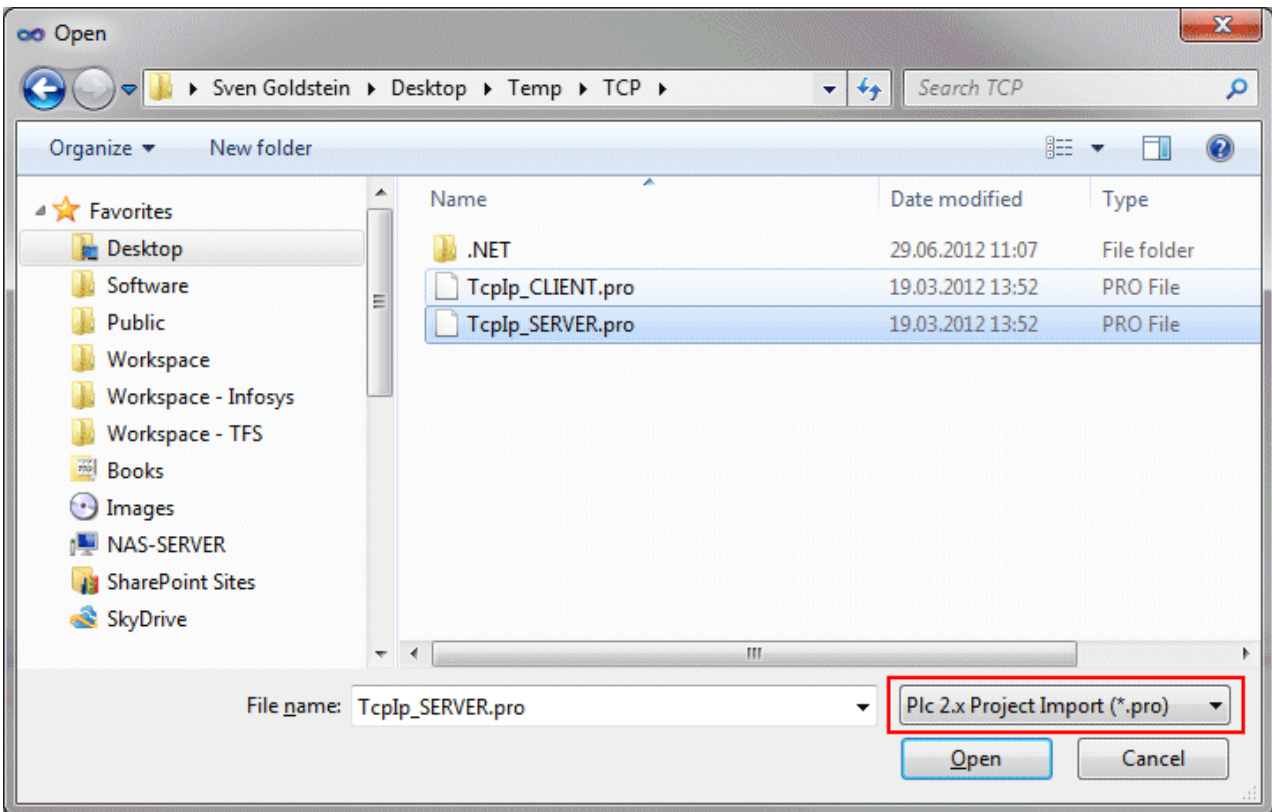
Die TwinCAT 2 Bibliotheken werden mit der Supplement-Installation geliefert.

Normalerweise werden diese Dateien im Verzeichnis "C:\TwinCAT\PlcLib\" installiert. Abhängig von der in ihrem SPS-Projekt genutzten Bibliothek müssen Sie die entsprechende Datei nach "C:\TwinCAT\3.1\Components\Plc\Converter\Lib\" kopieren und folgende Schritte ausführen:

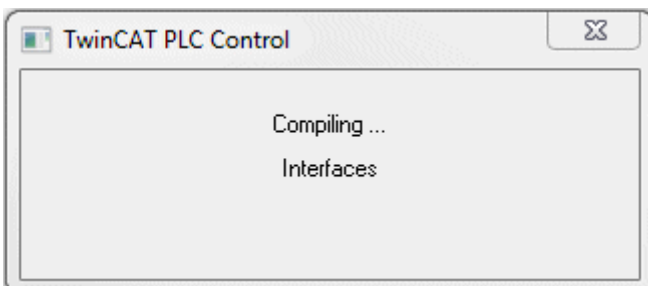
- Öffnen Sie TwinCAT XAE
- Erzeugen Sie eine neue TwinCAT 3 Solution
- Klicken Sie mit der rechten Maustaste auf den Knoten "PLC" und wählen Sie *Vorhandenes Element hinzufügen* aus



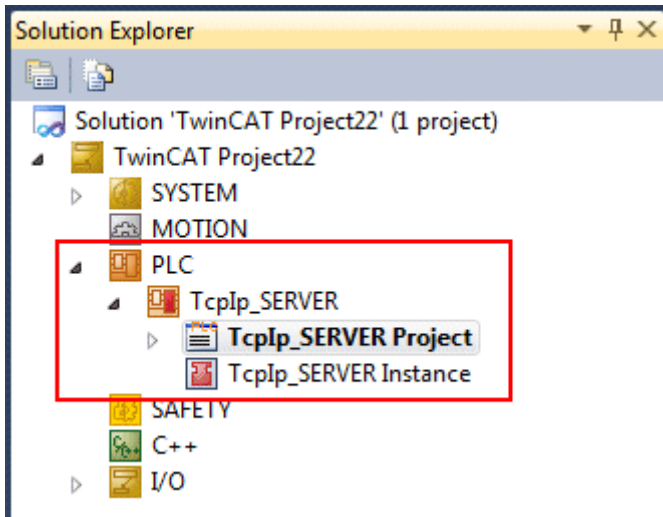
- Unter "Datei öffnen" wählen sie *Plc 2.x Projektimport (\*.pro)* als Dateityp aus, gehen zum Verzeichnis mit Ihrem TwinCAT 2 SPS Projekt, wählen die entsprechende .pro-Datei aus und klicken auf *Öffnen*



- Nach dem Klick auf *Öffnen*, startet TwinCAT 3 die Konvertierung und zeigt anschliessend das konvertierte SPS-Projekt im *SPS-Knoten* an.







## 4 Technische Einführung

### 4.1 IEC 60870-5-101

TF6500 implementiert zwei Schnittstellen über die einzelne IEC 60870-5-101 ASDUs (Dienstdateneinheiten der Anwendungsschicht) über die seriellen Schnittstellen des PCs und die Beckhoff KL6xxx-/EL6xxx-Klemmen verschickt und empfangen werden können.

#### "High level"-Schnittstelle:

Bei dieser Schnittstelle handelt es sich um eine sogenannte "Ein-Baustein-Lösung". Alle Funktionalitäten sind in einem SPS-Baustein gekapselt. Der Baustein implementiert die wichtigsten Dienste und Funktionen. Diese Implementierung ist für über 90% der Anwendungen ausreichend.

**Pro:** Sehr kleiner SPS-Programmieraufwand um eine laufende Applikation zu erhalten; Alle Dienste wie Generalabfrage, Zählerabfrage, Uhrzeitsynchronisation, Befehlsausführung, spontane Datenübertragung usw sind bereits in dem Baustein implementiert und werden automatisch ausgeführt; Das Mapping der IEC->SPS Prozessdaten und das der Datenpunkte wird über Funktionsaufrufe konfiguriert; Der SPS-Programmierer muss nicht sehr gut mit der Protokollnorm vertraut sein;

**Contra:** Die SPS-Applikation hat nur einen geringen Einfluss auf die Protokollausführung; Kein Einfluss auf die Ausführung der Dienste, diese werden intern automatisch ausgeführt; Zeitstempel werden von dem Baustein automatisch generiert und können nicht verändert (von extern übergeben) werden; Es ist z.B. nur die direkte Befehlsausführung möglich; Schlechtere Performance bei vielen Datenpunkten.

Diese Schnittstelle empfiehlt sich wenn Sie:

- Mit der Protokollnorm nicht vertraut sind;
- Eine einfache Applikation mit einer handvoll Datenpunkten implementieren möchten (<1000);
- Keine grossen Performace-Anforderungen an die Applikation stellen;
- Keine besondere Befehlsausführung wie Select/Execute oder Daten + Zeistempel von externen Geräten versenden möchten;
- Keine Funktionalitäten benötigen die laut Kompatibilitätsliste nicht unterstützt werden;

#### "Low level"-Schnittstelle:

Diese Schnittstelle setzt bei dem Protokollstack tiefer an und ermöglicht das Versenden und/oder Empfangen einzelner Frames (ASDU's). Innerhalb der Protokollstruktur liegt diese Schnittstelle oberhalb der Verbindungsschicht und implementiert bereits die nötigen Prozeduren und Übertragungstelegrammformate. Anwendungsfunktionen wie z.B. Abruf der einzelnen Zählerstände oder der Produktidentifikation sind in der Schnittstelle nicht implementiert, der Anwender kann aber mit Hilfe der Schnittstelle diese Anwendungsfunktionen selber implementieren.

**Pro:** Sehr flexibel; Alle Eigenschaften im ASDU-Frame können verändert werden (z.B. ein eigener Zeitstempel, Select/Execute oder eine besondere Befehlsausführung usw.); Weil nur die benötigten Dienste implementiert werden kann eine hohe Performance erreicht werden; hohe Performance bei vielen Datenpunkten;

**Contra:** Größerer Programmieraufwand; Alle Dienste wie Generalabfrage, Zählerabfrage, Uhrzeitsynchronisation, Befehlsausführung, spontane Datenübertragung usw. müssen von dem SPS-Programmierer selbst implementiert (ausprogrammiert) werden; Der SPS-Programmierer muss mit der Protokollnorm vertraut sein.

Diese Schnittstelle empfiehlt sich wenn Sie:

- Mit der Protokollnorm vertraut sind;
- Eine Protokollkonverter-Applikation implementieren;
- In der Applikation die verfügbaren Norm-Funktionalitäten fast vollständig implementieren;
- Besondere Funktionalitäten verwenden, wie z.B. das Weiterleiten der Zeitstempel von einem Modbus-Gerät oder die Kontrolle über die Befehlsausführung erlangen;
- Funktionalitäten benötigen, die laut Kompatibilitätsliste nicht unterstützt werden;

- Viele Datenpunkte haben (>1000) und eine hohe Performance benötigen;

Innerhalb der Protokollstruktur liegt diese Schnittstelle oberhalb der Verbindungsschicht und implementiert bereits die nötigen Prozeduren und Übertragungstelegrammformate. Anwendungsfunktionen wie z.B. Generalabfrage, Uhrzeitsynchronisation oder Zählerabfrage sind in der Schnittstelle nicht implementiert, der Anwender kann aber mit Hilfe der Schnittstelle diese Anwendungsfunktionen selber implementieren.

**Protokollstruktur des Endsystems:**

Auswahl von Anwendungsfunktionen aus IEC 60870-5-5	Anwenderprozess
Auswahl von ASDU (Dienstdateneinheiten der Anwendungsschicht) aus IEC 60870-5-3, IEC 60870-5-4 und IEC 60870-5-101	Anwendungsschicht (7)
N/A	Darstellungsschicht (6)
	Sitzungsschicht (5)
	Transportschicht (4)
	Vermittlungsschicht (3)
unbalanced / balanced IEC 60870-5-2 IEC 60870-5-1 (FT 1.2)	Verbindungsschicht (2)
EIA RS485, RS232 (V.24), Fibre Optics	Physikalische Schicht (1)



Die Schichten 3 bis 6 werden nicht benutzt.

**4.1.1 IEC 60870-5-101 telegram structure**

- FT 1.2 frame format is used;
- Frames with fixed and variable block length and single control characters are used;
- Hamming distance = 4;

**Frame with variable length**

This frame type is used to transmit user data between controlling and controlled station

byte\bit	7	6	5	4	3	2	1	0					
0	Start byte 1 (0x68)								Header		LPCI		LPDU
1	Block length												
2	Block length (copy)												
3	Start byte 2 (0x68)												
4	RES	PRM	FCB / ACD	FCV / DFC	Function code			Control field					
5	Link address fields (0, 1 or 2 octets)												
	<u>Type identification [▶ 500]</u>								DATA UNIT IDENTIFIER		ASDU		
	SQ	Number of objects											
	T	P/N	<u>Cause of transmission (COT) [▶ 509]</u>										
	Originator address (ORG, 0 or 1 octets)												
	ASDU address fields (1 or 2 octets)												
	Information object address fields (IOA)							Info-object					
	(1,2 or 3 octets)												
	Object information												
n-1	Checksum								Tail				
n	Stop byte (0x16)												

**Frame with fixed length**

This frame type is used for link layer services. Sometimes used as an confirmation.

byte\bit	7	6	5	4	3	2	1	0				
0	Start byte 1 (0x10)										LPCI	
1	RES	PRM	FCB / ACD	FCV / DFC	Function code			Control field				
2	Link address (0, 1 or 2 octets)											
n-1	Checksum											
n	Stop byte (0x16)											

**Single character data**

Single characters are used to confirm link layer and user data transmissions.

- 0xE5 (positive acknowledge);
- 0xA2 (negative acknowledge);

**Simple samples and explanations**

101substation configuration: Link address = 2 octets, COT = 1 octet (ORG address not used), ASDU address = 2 octets, IOA address = 2 octets

**Sample 1**

10 49 0C 00 55 16

LPDU bytes	Explanation
10	Start byte: frame with fixed length
49	Control field: PRM-bit set (frame from primary station), function code = 9 (link status)
0C 00	Link address (2 octets) = 12 dec.
55	Checksum
16	Stop byte

**Sample 2**

10 0B 0C 00 17 16

LPDU bytes	Explanation
10	Start byte: frame with fixed length
0B	Control field: PRM-bit not set (frame from secondary station), function code = 11 (status of link or access demand)
0C 00	Link address (2 octets) = 12 dec.
17	Checksum
16	Stop byte

**Sample 3**

68 0B 0B 68 08 0C 00 65 01 0A 0C 00 00 00 05 95 16

LPDU bytes	Explanation
68	Start byte 1: frame with variable length
0B 0B	Block length and block length copy
68	Start byte 2
08	Control field: PRM-bit not set (frame from secondary station), function code = 8 (user data)
0C 00	Link address (2 octets) = 12 dec.
65	Type identification: C_CI_NA_1 (counter interrogation)
01	Number of objects = 1
0A	Cause of transmission = 10 (activation confirmation)
0C 00	Common ASDU address (2 octets) = 12 dec.
00 00	Object address (2 octets) = 0
05	Counter interrogation request qualifier = 5 (general counter interrogation)
95	Checksum
16	Stop byte

**Sample 4**

68 0F 0F 68 08 0C 00 0F 01 03 0C 00 81 30 DA 16 00 00 07 DB 16

LPDU bytes	Explanation
68	Start byte 1: frame with variable length
0F 0F	Block length and block length copy
68	Start byte 2
08	Control field: PRM-bit not set (frame from secondary station), function code = 8 (user data)
0C 00	Link address (2 octets) = 12 dec.
0F	Type identification: M_IT_NA_1 (integrated total)
01	Number of objects = 1
03	Cause of transmission = 3 (spontaneous)
0C 00	Common ASDU address (2 octets) = 12 dec.
81 30	Object address (2 octets)
DA 16 00 00	BCR (binary counter value)
07	Quality descriptor = 7 (sequence)
DB	Checksum
16	Stop byte

#### Sample 4

68 2B 2B 68 08 0C 00 0B 07 03 0C 00 10 30 BE 09 00 11 30 90 09 00 0E 30 75 00 00 28 30 25 09 00 29 30 75 00 00 0F 30 0F 0A 00 2E 30 AE 05 00 85 16

LPDU bytes	Explanation
68	Start byte 1: frame with variable length
2B 2B	Block length and block length copy
68	Start byte 2
08	Control field: PRM-bit not set (frame from secondary station), function code = 8 (user data)
0C 00	Link address (2 octets) = 12 dec.
0B	Type identification: M_ME_NB_1(measured value, scaled value)
07	Number of objects = 7
03	Cause of transmission = 3 (spontaneous)
0C 00	Common ASDU address (2 octets) = 12 dec.
10 30	Object address (2 octets) of <b>first</b> information object
BE 09 00	Scaled value + QDS (quality descriptor) of <b>first</b> information object
11 30	Object address (2 octets) of <b>second</b> information object
90 09 00	Scaled value + QDS (quality descriptor) of <b>second</b> information object
0E 30	Object address (2 octets) of <b>third</b> information object
75 00 00	Scaled value + QDS (quality descriptor) of <b>third</b> information object
28 30 25 09 00 29 30 75 00 00 0F 30 0F 0A 00 2E 30 AE 05 00	Object address + Scaled value + QDS (quality descriptor) of information object four to seven
85	Checksum
16	Stop byte

## **4.1.2 ASDU-Objektbeschreibung**

### **4.1.2.1 Standard IEC 60870-5-101 Datentypen**

Type	Dez	Hex	Beschreibung
ASDU_TYPEUNDEF	0	0x00	Wird nicht verwendet
<a href="#">M SP NA 1 [▶ 122]</a>	1	0x01	Einzelmeldung
<a href="#">M SP TA 1 [▶ 124]</a>	2	0x02	Einzelmeldung mit einem Zeitstempel
<a href="#">M DP NA 1 [▶ 128]</a>	3	0x03	Doppelmeldung
<a href="#">M DP TA 1 [▶ 130]</a>	4	0x04	Doppelmeldung mit einem Zeitstempel
<a href="#">M ST NA 1 [▶ 134]</a>	5	0x05	Stufenstellungsmeldung
<a href="#">M ST TA 1 [▶ 136]</a>	6	0x06	Stufenstellungsmeldung mit einem Zeitstempel
<a href="#">M BO NA 1 [▶ 140]</a>	7	0x07	Bitmuster von 32 bit
<a href="#">M BO TA 1 [▶ 142]</a>	8	0x08	Bitmuster von 32 bit mit einem Zeitstempel
<a href="#">M ME NA 1 [▶ 146]</a>	9	0x09	Messwert, normierter Wert
<a href="#">M ME TA 1 [▶ 148]</a>	10	0x0A	Messwert, normierter Wert mit einem Zeitstempel
<a href="#">M ME NB 1 [▶ 154]</a>	11	0x0B	Messwert, skaliertes Wert
<a href="#">M ME TB 1 [▶ 156]</a>	12	0x0C	Messwert, skaliertes Wert mit einem Zeitstempel
<a href="#">M ME NC 1 [▶ 160]</a>	13	0x0D	Messwert, Gleitkommazahl mit einfacher Genauigkeit
<a href="#">M ME TC 1 [▶ 162]</a>	14	0x0E	Messwert, Gleitkommazahl mit einfacher Genauigkeit und einem Zeitstempel
<a href="#">M IT NA 1 [▶ 166]</a>	15	0x0F	Zählwerte
<a href="#">M IT TA 1 [▶ 168]</a>	16	0x10	Zählwerte mit einem Zeitstempel
<a href="#">M EP TA 1 [▶ 214]</a>	17	0x11	Schutzereignis mit einem Zeitstempel
<a href="#">M EP TB 1 [▶ 216]</a>	18	0x12	Gepackte Anregungen des Schutzes mit einem Zeitstempel
<a href="#">M EP TC 1 [▶ 219]</a>	19	0x13	Gepackte Auslösungen des Schutzes mit einem Zeitstempel
<a href="#">M PS NA 1 [▶ 230]</a>	20	0x14	Gepackte Einzelmeldungen mit Zustandsanzeige
<a href="#">M ME ND 1 [▶ 152]</a>	21	0x15	Messwert, normierter Wert ohne Qualitätskennung
ASDU_TYPE_22..29	22..29	0x16..0x1D	Reserviert (Standardbereich)
<a href="#">M SP TB 1 [▶ 126]</a>	30	0x1E	Einzelmeldung mit einem Zeitstempel CP56Time2a
<a href="#">M DP TB 1 [▶ 132]</a>	31	0x1F	Doppelmeldung mit einem Zeitstempel CP56Time2a
<a href="#">M ST TB 1 [▶ 138]</a>	32	0x20	Stufenstellungsmeldung mit einem Zeitstempel CP56Time2a
<a href="#">M BO TB 1 [▶ 144]</a>	33	0x21	Bitmuster von 32 bit mit einem Zeitstempel CP56Time2a
<a href="#">M ME TD 1 [▶ 150]</a>	34	0x22	Messwert, normierter Wert mit einem Zeitstempel CP56Time2a
<a href="#">M ME TE 1 [▶ 158]</a>	35	0x23	Messwert, skaliertes Wert mit einem Zeitstempel CP56Time2a
<a href="#">M ME TF 1 [▶ 164]</a>	36	0x24	Messwert, Gleitkommazahl mit einfacher Genauigkeit und einem Zeitstempel CP56Time2a
<a href="#">M IT TB 1 [▶ 170]</a>	37	0x25	Zählwerte mit einem Zeitstempel CP56Time2a
<a href="#">M EP TD 1 [▶ 222]</a>	38	0x26	Schutzereignis mit einem Zeitstempel CP56Time2a
<a href="#">M EP TE 1 [▶ 224]</a>	39	0x27	Gepackte Anregungen des Schutzes mit einem Zeitstempel CP56Time2a
<a href="#">M EP TF 1 [▶ 227]</a>	40	0x28	Gepackte Auslösungen des Schutzes mit einem Zeitstempel CP56Time2a
ASDU_TYPE_41..44	41..44	0x29..0x2C	Reserviert (Standardbereich)
<a href="#">C SC NA 1 [▶ 172]</a>	45	0x2D	Einzelbefehl
<a href="#">C DC NA 1 [▶ 176]</a>	46	0x2E	Doppelbefehl



Type	Dez	Hex	Beschreibung
<a href="#">C_RC_NA_1 [▶ 180]</a>	47	0x2F	Stufenstellbefehl
<a href="#">C_SE_NA_1 [▶ 184]</a>	48	0x30	Sollwert-Stellbefehl, normierter Wert
<a href="#">C_SE_NB_1 [▶ 188]</a>	49	0x31	Sollwert-Stellbefehl, skaliertes Wert
<a href="#">C_SE_NC_1 [▶ 192]</a>	50	0x32	Sollwert-Stellbefehl, Gleitkommazahl mit einfacher Genauigkeit
<a href="#">C_BO_NA_1 [▶ 196]</a>	51	0x33	Bitmusterbefehl von 32 bit
ASDU_TYPE_52..57	52..57	0x34..0x39	Reserviert (Standardbereich)
<a href="#">C_SC_TA_1 [▶ 174]</a>	58	0x3A	Einzelbefehl mit einem Zeitstempel CP56Time2a
<a href="#">C_DC_TA_1 [▶ 178]</a>	59	0x3B	Doppelbefehl mit einem Zeitstempel CP56Time2a
<a href="#">C_RC_TA_1 [▶ 182]</a>	60	0x3C	Stufenstellbefehl mit einem Zeitstempel CP56Time2a
<a href="#">C_SE_TA_1 [▶ 186]</a>	61	0x3D	Sollwert-Stellbefehl, normierter Wert mit einem Zeitstempel CP56Time2a
<a href="#">C_SE_TB_1 [▶ 190]</a>	62	0x3E	Sollwert-Stellbefehl, skaliertes Wert mit einem Zeitstempel CP56Time2a
<a href="#">C_SE_TC_1 [▶ 194]</a>	63	0x3F	Sollwert-Stellbefehl, Gleitkommazahl mit einfacher Genauigkeit und einem Zeitstempel CP56Time2a
<a href="#">C_BO_TA_1 [▶ 198]</a>	64	0x40	Bitmusterbefehl von 32 bit mit einem Zeitstempel CP56Time2a
ASDU_TYPE_65..69	65..69	0x41..0x45	Reserviert (Standardbereich)
<a href="#">M_EI_NA_1 [▶ 202]</a>	70	0x46	Initialisierungsende
ASDU_TYPE_71..99	71..99	0x47..0x63	Reserviert (Standardbereich)
<a href="#">C_IC_NA_1 [▶ 206]</a>	100	0x64	(General-, Stations-) Abfragebefehl
<a href="#">C_CI_NA_1 [▶ 208]</a>	101	0x65	Zähler-Abfragebefehl
<a href="#">C_RD_NA_1 [▶ 212]</a>	102	0x66	Abfragebefehl
<a href="#">C_CS_NA_1 [▶ 204]</a>	103	0x67	Uhrzeit-Synchronisationsbefehl
<a href="#">C_TS_NA_1 [▶ 200]</a>	104	0x68	Prüfbefehl
<a href="#">C_RP_NA_1 [▶ 210]</a>	105	0x69	Prozess-Rücksetzbefehl
<a href="#">C_CD_NA_1</a>	106	0x6A	Befehl zur Telegrammlaufzeit-Erfassung
<a href="#">C_TS_TA_1</a>	107	0x6B	Prüfbefehl mit einem Zeitstempel CP56Time2a
ASDU_TYPE_108..109	108..109	0x6C..0x6D	Reserviert (Standardbereich)
<a href="#">P_ME_NA_1 [▶ 232]</a>	110	0x6E	Parameter für Messwerte, normierter Wert
<a href="#">P_ME_NB_1 [▶ 234]</a>	111	0x6F	Parameter für Messwerte, skaliertes Wert
<a href="#">P_ME_NC_1 [▶ 236]</a>	112	0x70	Parameter für Messwerte, Gleitkommazahl mit einfacher Genauigkeit
<a href="#">P_AC_NA_1 [▶ 238]</a>	113	0x71	Parameter für Aktivierung
ASDU_TYPE_114..119	114..119	0x72..0x77	Reserviert (Standardbereich)
<a href="#">F_FR_NA_1</a>	120	0x78	File ready
<a href="#">F_SR_NA_1</a>	121	0x79	Section ready
<a href="#">F_SC_NA_1</a>	122	0x7A	Call directory, select file, call file, call section
<a href="#">F_LS_NA_1</a>	123	0x7B	Last section, last segment
<a href="#">F_FA_NA_1</a>	124	0x7C	ACK file, ACK section
<a href="#">F_SG_NA_1</a>	125	0x7D	Segment
<a href="#">F_DR_TA_1</a>	126	0x7E	Verzeichnis
ASDU_TYPE_127..255	127..255	0x7F..0xFF	Reserviert (benutzerdefinierter Bereich)

## **4.1.2.2          Single-point information**

### **4.1.2.2.1          M\_SP\_NA\_1**

Single-point information without time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[► 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[► 484]</a>
	---	eType	= 0x01 (1)	= M_SP_NA_1	<a href="#">Type identification</a> <a href="#">[► 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[► 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[► 509]</a>

---	- info													INFOR MATIO N OBJEC T [▶ 485]
	---	objAdd r												Inform ation object addres s
	---	stream												Inform ation elemen t/ object data [▶ 485]
		---	length	= 1										
		---	data		7	6	5	4	3	2	1	0		
		---	data[0]	=	IV [▶ 329]	NT [▶ 329]	SB [▶ 329]	BL [▶ 329]	0	0	0	SPI [▶ 526]		SIQ = Single- point informa tion with quality descri ptor
		---	data[1..	.IEC87 0_MAX_ AS DU_D ATA_ BYTE] =										Reser ved

**4.1.2.2.2 M\_SP\_TA\_1**

Single-point information with CP24Time2a time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType	= 0x02 (2)	= M_SP_TA_1	<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[▶ 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[▶ 509]</a>

---	- info												INFOR MATIO N OBJEC T [▶ 485]
	---	objAdd r											Inform ation object addres s
	---	stream											Inform ation elemen t/ object data [▶ 485]
		---	length = 4										
		---	data		7	6	5	4	3	2	1	0	
		---	data[0] =	IV [▶ 329]	NT [▶ 329]	SB [▶ 329]	BL [▶ 329]	0	0	0	SPI [▶ 526]		SIQ = Single- point informa tion with quality descri ptor
		---	data[1. .3] =	CP24Time2a [▶ 534]									Three octets binary time tag
		---	data[4. .IEC87 0_MA X_AS DU_D ATA_ BYTE] =										Reser ved

**4.1.2.2.3 M\_SP\_TB\_1**

Single-point information with CP56Time2a time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType	= 0x1E (30)	= M_SP_TB_1	<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[▶ 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[▶ 509]</a>

---	- info												INFOR MATIO N OBJEC T [▶ 485]
	---	objAdd r											Inform ation object addres s
	---	stream											Inform ation elemen t/ object data [▶ 485]
	---	length	= 8										
	---	data		7	6	5	4	3	2	1	0		
	---	data[0]	=	IV [▶ 329]	NT [▶ 329]	SB [▶ 329]	BL [▶ 329]	0	0	0	SPI [▶ 526]		SIQ = Single- point informa tion with quality descri ptor
	---	data[1. .7]	=	CP56Time2a [▶ 531]									Seven octets binary time tag
	---	data[8. .IEC87 0_MA X_AS DU_D ATA_ BYTE]	=										Reser ved

### 4.1.2.3 Double-point information

#### 4.1.2.3.1 M\_DP\_NA\_1

Double-point information without time tag.



- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType	= 0x03 (3)	= M_DP_NA_1	<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[▶ 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[▶ 509]</a>

---	- info												INFOR MATIO N OBJEC T [▶ 485]
	---	objAdd r											Inform ation object addres s
	---	stream											Inform ation elemen t/ object data [▶ 485]
		---	length	= 1									
		---	data		7	6	5	4	3	2	1	0	
		---	data[0]	=	IV [▶ 329]	NT [▶ 329]	SB [▶ 329]	BL [▶ 329]	0	0	DPI [▶ 527]		DIQ = Doubl e-point inform ation with quality descri ptor
		---	data[1..	.IEC87 0_MA X_AS DU_D ATA_ BYTE] =									Reser ved

**4.1.2.3.2 M\_DP\_TA\_1**

Double-point information with CP24Time2a time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType	= 0x04 (4)	= M_DP_TA_1	<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[▶ 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[▶ 509]</a>

---	- info												INFOR MATIO N OBJEC T [▶ 485]
	---	objAdd r											Inform ation object addres s
	---	stream											Inform ation elemen t/ object data [▶ 485]
		---	length	= 4									
		---	data		7	6	5	4	3	2	1	0	
		---	data[0]	=	IV [▶ 329]	NT [▶ 329]	SB [▶ 329]	BL [▶ 329]	0	0	DPI [▶ 527]		DIQ = Doubl e-point informa tion with quality descri ptor
		---	data[1. .3]	=	CP24Time2a [▶ 534]								Three octet binary time tag
		---	data[4. .IEC87 0_MA X_AS DU_D ATA_ BYTE]	=									Reser ved

**4.1.2.3.3 M\_DP\_TB\_1**

Double-point information with CP56Time2a time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType	= 0x1F (31)	= M_DP_TB_1	<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[▶ 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[▶ 509]</a>

---	- info												INFOR MATIO N OBJEC T [▶ 485]
	---	objAdd r											Inform ation object addres s
	---	stream											Inform ation elemen t/ object data [▶ 485]
	---	length	= 8										
	---	data		7	6	5	4	3	2	1	0		
	---	data[0]	=	IV [▶ 329]	NT [▶ 329]	SB [▶ 329]	BL [▶ 329]	0	0	DPI [▶ 527]			DIQ = Doubl e-point informa tion with quality descri ptor
	---	data[1. .7]	=	CP56Time2a [▶ 531]									Seven octets binary time tag
	---	data[8. .IEC87 0_MA X_AS DU_D ATA_ BYTE]	=										Reser ved

**4.1.2.4 Step position information**

**4.1.2.4.1 M\_ST\_NA\_1**

Step position information without time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType	= 0x05 (5)	= M_ST_NA_1	<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[▶ 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[▶ 509]</a>

---	- info																<u>INFORMATION OBJECT</u> <u>IT</u> <u>[▶ 485]</u>
	---	objAddr															Information object address
	---	stream															Information element/object data <u>[▶ 485]</u>
	---	length	= 2														
	---	data		7	6	5	4	3	2	1	0						
	---	data[0]	=	VTI <u>[▶ 240]</u>											Value with transient state indication		
	---	data[1]	=	<u>IV</u> <u>[▶ 329]</u>	<u>NT</u> <u>[▶ 329]</u>	<u>SB</u> <u>[▶ 329]</u>	<u>BL</u> <u>[▶ 329]</u>	0	0	0	<u>OV</u> <u>[▶ 329]</u>						QDS = Quality descriptor
	---	data[2]	.IEC87 O_MAX DU_DATA_BYTE] =												Reserved		

#### 4.1.2.4.2 M\_ST\_TA\_1

Step position information with CP24Time2a time tag.



- obj					<a href="#">ASDU object</a> <a href="#">[► 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[► 484]</a>
	---	eType	= 0x06 (6)	= M_ST_TA_1	<a href="#">Type identification</a> <a href="#">[► 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[► 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[► 509]</a>

---	- info												INFOR MATIO N OBJEC T [▶ 485]
	---	objAdd r											Inform ation objec t addres s
	---	stream											Inform ation elemen t/ objec t data [▶ 485]
	---	length	= 5										
	---	data		7	6	5	4	3	2	1	0		
	---	data[0]	=	VTI [▶ 240]									Value with transie nt state indicati on
	---	data[1]	=	<u>IV</u> [▶ 329]	<u>NT</u> [▶ 329]	<u>SB</u> [▶ 329]	<u>BL</u> [▶ 329]	0	0	0	<u>OV</u> [▶ 329]		QDS = Quality descri ptor
	---	data[2. .4]	=	CP24Time2a [▶ 534]									Three octets binary time tag
	---	data[5. .IEC87 0_MA X_AS DU_D ATA_ BYTE]	=										Reser ved

**4.1.2.4.3 M\_ST\_TB\_1**

Step position information with CP56Time2a time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType	= 0x20 (32)	= M_ST_TB_1	<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[▶ 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[▶ 509]</a>

---	- info												INFOR MATIO N OBJEC T [▶ 485]
	---	objAdd r											Inform ation objec t addres s
	---	stream											Inform ation elemen t/ objec t data [▶ 485]
	---	length	= 9										
	---	data		7	6	5	4	3	2	1	0		
	---	data[0]	=	VTI [▶ 240]									Value with transie nt state indicati on
	---	data[1]	=	<u>IV</u> [▶ 329]	<u>NT</u> [▶ 329]	<u>SB</u> [▶ 329]	<u>BL</u> [▶ 329]	0	0	0	<u>OV</u> [▶ 329]		QDS = Quality descri ptor
	---	data[2. .8]	=	CP56Time2a [▶ 531]									Seven octets binary time tag
	---	data[9. .IEC87 0_MA X_AS DU_D ATA_ BYTE]	=										Reser ved

**4.1.2.5 Bitstring of 32 bits**

**4.1.2.5.1 M\_BO\_NA\_1**

Bitstring of 32 bits without time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[► 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[► 484]</a>
	---	eType	= 0x07 (7)	= M_BO_NA_1	<a href="#">Type identification</a> <a href="#">[► 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[► 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[► 509]</a>

---	- info														INFOR MATIO N OBJEC T [► 485]
	---	objAdd r													Inform ation object addres s
	---	stream													Inform ation elemen t/ object data [► 485]
		---	length = 5												
		---	data		7	6	5	4	3	2	1	0			
		---	data[0. .3] =	BSI [► 241]										Binary state inform ation	
		---	data[4] =	<u>IV</u> [► 329]	<u>NT</u> [► 329]	<u>SB</u> [► 329]	<u>BL</u> [► 329]	0	0	0	<u>OV</u> [► 329]	<u>QDS</u> [► 329]	= Quality descri ptor		
		---	data[5. .IEC87 0_MA X_AS DU_D ATA_ BYTE] =											Reser ved	

**4.1.2.5.2 M\_BO\_TA\_1**

Bitstring of 32 bits with CP24Time2a time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType	= 0x08 (8)	= M_BO_TA_1	<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[▶ 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[▶ 509]</a>

---	- info													INFORMATION OBJECT ID [▶ 485]
	---	objAddr												Information object address
	---	stream												Information element/object data [▶ 485]
		---	length	= 8										
		---	data		7	6	5	4	3	2	1	0		
		---	data[0..3]	=	BSI [▶ 241]								Binary state information	
		---	data[4]	=	IV [▶ 329]	NT [▶ 329]	SB [▶ 329]	BL [▶ 329]	0	0	0	OV [▶ 329]	QDS [▶ 329]	= Quality descriptor
		---	data[5..7]	=	CP24Time2a [▶ 534]								Three octets binary time tag	
		---	data[8..IEC870_MAX_AS_DUDA_BYTE]	=									Reserved	

**4.1.2.5.3 M\_BO\_TB\_1**

Bitstring of 32 bits with CP56Time2a time tag.



- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType	= 0x21 (33)	= M_BO_TB_1	<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[▶ 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[▶ 509]</a>

---	- info																<u>INFORMATION OBJECT I</u> [▶ 485]	
	---	objAddr															Information object address	
	---	stream															Information element/object data [▶ 485]	
		---	length	= 12														
		---	data		7	6	5	4	3	2	1	0						
		---	data[0..3] =		BSI [▶ 241]													Binary state information
		---	data[4] =		<u>IV</u> [▶ 329]	<u>NT</u> [▶ 329]	<u>SB</u> [▶ 329]	<u>BL</u> [▶ 329]	0	0	0	<u>OV</u> [▶ 329]					<u>QDS</u> [▶ 329] = Quality descriptor	
		---	data[5..11] =		CP56Time2a [▶ 531]													Seven octets binary time tag
		---	data[12..IEC870_MAX_AX_SDU_DATA_BYTE] =														Reserved	

**4.1.2.6 Measured value, normalized value**

**4.1.2.6.1 M\_ME\_NA\_1**

Measured value, normalized value without time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[► 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[► 484]</a>
	---	eType	= 0x09 (9)	= M_ME_NA_1	<a href="#">Type identification</a> <a href="#">[► 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[► 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[► 509]</a>

---	- info												INFOR MATIO N OBJEC T [▶ 485]
	---	objAdd r											Inform ation object addres s
	---	stream											Inform ation elemen t/ object data [▶ 485]
	---	length	= 3										
	---	data		7	6	5	4	3	2	1	0		
	---	data[0. .1] =		NVA [▶ 241]									Norma lized value
	---	data[2] =		<u>IV</u> [▶ 329]	<u>NT</u> [▶ 329]	<u>SB</u> [▶ 329]	<u>BL</u> [▶ 329]	0	0	0	<u>OV</u> [▶ 329]		QDS = Quality descri ptor
	---	data[3. .IEC87 0_MA X_AS DU_D ATA_ BYTE] =											Reser ved

#### 4.1.2.6.2 M\_ME\_TA\_1

Measured value, normalized value with CP24Time2a time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType	= 0x0A (10)	= M_ME_TA_1	<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[▶ 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[▶ 509]</a>

---	- info																<a href="#">INFORMATION OBJECT I</a> <a href="#">[▶ 485]</a>
	---	objAddr															Information object addresses
	---	stream															Information element/object data <a href="#">[▶ 485]</a>
		---	length	= 6													
		---	data		7	6	5	4	3	2	1	0					
		---	data[0..1]	=	<a href="#">NVA [▶ 241]</a>									Normalized value			
		---	data[2]	=	<a href="#">IV [▶ 329]</a>	<a href="#">NT [▶ 329]</a>	<a href="#">SB [▶ 329]</a>	<a href="#">BL [▶ 329]</a>	0	0	0	<a href="#">OV [▶ 329]</a>		QDS = Quality descriptor			
		---	data[3..5]	=	<a href="#">CP24Time2a [▶ 534]</a>									Three octets binary time tag			
		---	data[6..IEC870_MAX_AS DU_DATA_BYTE]	=										Reserved			

#### 4.1.2.6.3 M\_ME\_TD\_1

Measured value, normalized value with CP56Time2a time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[► 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[► 484]</a>
	---	eType	= 0x22 (34)	= M_ME_TD_1	<a href="#">Type identification</a> <a href="#">[► 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[► 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[► 509]</a>

---	- info																		INFORMATION OBJECT <a href="#">IT</a> <a href="#">[▶ 485]</a>
	---	objAddr																	Information object address
	---	stream																	Information element/object data <a href="#">[▶ 485]</a>
		---	length	= 10															
		---	data			7	6	5	4	3	2	1	0						
		---	data[0..1]	=	<a href="#">NVA</a> <a href="#">[▶ 241]</a>											Normalized value;			
		---	data[2]	=	<a href="#">IV</a> <a href="#">[▶ 329]</a>	<a href="#">NT</a> <a href="#">[▶ 329]</a>	<a href="#">SB</a> <a href="#">[▶ 329]</a>	<a href="#">BL</a> <a href="#">[▶ 329]</a>	0	0	0	<a href="#">OV</a> <a href="#">[▶ 329]</a>							QDS = Quality descriptor
		---	data[3..9]	=	<a href="#">CP56Time2a</a> <a href="#">[▶ 531]</a>											Seven octets binary time tag			
		---	data[10..IEC870_MAX_AX_A_SDU_DATA_BYT_E]	=												Reserved			

**4.1.2.6.4 M\_ME\_ND\_1**

Measured value, normalized value without quality descriptor.



- obj					<a href="#">ASDU object</a> <a href="#">[► 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[► 484]</a>
	---	eType	= 0x15 (21)	= M_ME_ND_1	<a href="#">Type identification</a> <a href="#">[► 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[► 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[► 509]</a>

---	- info																		<u>INFORMATION OBJECT</u> I [▶ 485]	
	---	objAddr																	Information object addresses	
	---	stream																	Information element/object data [▶ 485]	
		---	length	= 2																
		---	data		7	6	5	4	3	2	1	0								
			---	data[0..1] =	<u>NVA [▶ 241]</u>															Normalized value
			---	data[2..IEC870_MAX_ADDRESS_BYTE] =															Reserved	

#### 4.1.2.7 Measured value, scaled value

##### 4.1.2.7.1 M\_ME\_NB\_1

Measured value, scaled value without time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType = 0x0B (11)	= M_ME_NB_1		<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ	= FALSE		Sequence of information objects
	---	nObj	= 1		Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[▶ 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[▶ 509]</a>

---	- info												INFOR MATIO N OBJEC T [▶ 485]
	---	objAdd r											Inform ation object address
	---	stream											Inform ation elemen t/ object data [▶ 485]
		---	length	= 3									
		---	data		7	6	5	4	3	2	1	0	
		---	data[0. .1] =	SVA [▶ 241]									Scaled value
		---	data[2] =	<u>IV</u> [▶ 329]	<u>NT</u> [▶ 329]	<u>SB</u> [▶ 329]	<u>BL</u> [▶ 329]	0	0	0	<u>OV</u> [▶ 329]		QDS = Quality descri ptor
		---	data[3. .IEC87 0_MA X_AS DU_D ATA_ BYTE] =										Reser ved

**4.1.2.7.2 M\_ME\_TB\_1**

Measured value, scaled value with CP24Time2a time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType	= 0x0C (12)	= M_ME_TB_1	<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[▶ 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[▶ 509]</a>

---	- info												INFOR MATIO N OBJEC T [▶ 485]
	---	objAdd r											Inform ation object address
	---	stream											Inform ation elemen t/ object data [▶ 485]
		---	length	= 6									
		---	data		7	6	5	4	3	2	1	0	
		---	data[0. .1]	=	SVA [▶ 241]								Scaled value
		---	data[2]	=	<u>IV</u> [▶ 329]	<u>NT</u> [▶ 329]	<u>SB</u> [▶ 329]	<u>BL</u> [▶ 329]	0	0	0	<u>OV</u> [▶ 329]	QDS = Quality descri ptor
		---	data[3. .5]	=	CP24Time2a [▶ 534]								Three octets binary time tag
		---	data[6. .IEC87 0_MA X_AS DU_D ATA_ BYTE]	=									Reser ved

**4.1.2.7.3 M\_ME\_TE\_1**

Measured value, scaled value with CP56Time2a time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType	= 0x23 (35)	= M_ME_TE_1	<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[▶ 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[▶ 509]</a>

---	- info												<a href="#">INFORMATION OBJECT I</a> <a href="#">[▶ 485]</a>
	---	objAddr											Information object addresses
	---	stream											Information element/object data <a href="#">[▶ 485]</a>
		---	length	= 10									
		---	data		7	6	5	4	3	2	1	0	
		---	data[0..1] =	<a href="#">SVA [▶ 241]</a>									Scaled value
		---	data[2] =	<a href="#">IV [▶ 329]</a>	<a href="#">NT [▶ 329]</a>	<a href="#">SB [▶ 329]</a>	<a href="#">BL [▶ 329]</a>	0	0	0	<a href="#">OV [▶ 329]</a>		QDS = Quality descriptor
		---	data[3..9] =	<a href="#">CP56Time2a [▶ 531]</a>									Seven octets binary time tag
		---	data[10..IEC870_MAX_AX_SDU_DATA_BYTE] =										Reserved

**4.1.2.8 Measured value, short floating point value**

**4.1.2.8.1 M\_ME\_NC\_1**

Measured value, short floating point value without time tag.



- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType	= 0x0D (13)	= M_ME_NC_1	<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[▶ 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[▶ 509]</a>

---	- info													INFORMATION OBJECT [▶ 485]
	---	objAddress												Information object address
	---	stream												Information element/object data [▶ 485]
	---	length	= 5											
	---	data		7	6	5	4	3	2	1	0			
	---	data[0..3]	=	R32 [▶ 241]									Short floating point value	
	---	data[4]	=	IV [▶ 329]	NT [▶ 329]	SB [▶ 329]	BL [▶ 329]	0	0	0	OV [▶ 329]		QDS = Quality descriptor	
	---	data[5..IEC870_MAX_DATA_BYTE]	=										Reserved	

**4.1.2.8.2 M\_ME\_TC\_1**

Measured value, short floating point value with CP24Time2a time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType	= 0x0E (14)	= M_ME_TC_1	<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[▶ 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[▶ 509]</a>

---	- info												INFORMATION OBJECT [▶ 485]
	---	objAddr											Information object address
	---	stream											Information element/object data [▶ 485]
	---	length	= 8										
	---	data		7	6	5	4	3	2	1	0		
	---	data[0..3]	=	R32 [▶ 241]									Short floating point value
	---	data[4]	=	IV [▶ 329]	NT [▶ 329]	SB [▶ 329]	BL [▶ 329]	0	0	0	OV [▶ 329]		QDS = Quality descriptor
	---	data[5..7]	=	CP24Time2a [▶ 534]									Three octets binary time tag
	---	data[8..IEC870_MAX_DATA_BYTE]	=										Reserved

**4.1.2.8.3 M\_ME\_TF\_1**

Measured value, short floating point value with CP56Time2a time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[► 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[► 484]</a>
	---	eType	= 0x24 (36)	= M_ME_TF_1	<a href="#">Type identification</a> <a href="#">[► 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[► 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[► 509]</a>

---	- info															<u>INFOR MATIO N OBJEC T [▶ 485]</u>
	---	objAdd r														Inform ation object addres s
	---	stream														<u>Inform ation elemen t/ object data [▶ 485]</u>
	---	length	= 12													
	---	data		7	6	5	4	3	2	1	0					
	---	data[0. .3] =		<u>R32 [▶ 241]</u>											<u>Short floatin g point value</u>	
	---	data[4] =		<u>IV</u> [▶ 329]	<u>NT</u> [▶ 329]	<u>SB</u> [▶ 329]	<u>BL</u> [▶ 329]	0	0	0	<u>OV</u> [▶ 329]					<u>QDS = Quality descri ptor</u>
	---	data[5. .11] =		<u>CP56Time2a [▶ 531]</u>											<u>Seven octets binary time tag</u>	
	---	data[1 2..IEC 870_M AX_A SDU_ DATA_ BYT E] =													<u>Reser ved</u>	

### 4.1.2.9 Integrated totals

#### 4.1.2.9.1 M\_IT\_NA\_1

Integrated total without time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[► 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[► 484]</a>
	---	eType	= 0x0F (15)	= M_IT_NA_1	<a href="#">Type identification</a> <a href="#">[► 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[► 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[► 509]</a>

---	- info													<u>INFORMATION OBJECT ID</u> [▶ 485]
---		objAddr												Information object address
---		stream												<u>Information element/object data</u> [▶ 485]
		---	length	= 5										
		---	data		7	6	5	4	3	2	1	0		
		---	data[0..3]	=	<u>BCR</u> [▶ 241]									Binary counter reading
		---	data[4]	=	<u>IV</u> [▶ 329]	<u>CA</u> [▶ 330]	<u>CY</u> [▶ 330]	Sequence						Quality descriptor
		---	data[5..IEC870_MAX_AS DU_DATA_BYTE]	=										Reserved

**4.1.2.9.2 M\_IT\_TA\_1**

Integrated total with CP24Time2a time tag.



- obj					<a href="#">ASDU object</a> <a href="#">[► 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[► 484]</a>
	---	eType	= 0x10 (16)	= M_IT_TA_1	<a href="#">Type identification</a> <a href="#">[► 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[► 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[► 509]</a>

---	- info																INFOR MATIO N OBJEC T [▶ 485]	
	---	objAdd r															Inform ation object addres s	
	---	stream															Inform ation elemen t/ object data [▶ 485]	
		---	length	= 8														
		---	data			7	6	5	4	3	2	1	0					
		---	data[0. .3]	=													BCR [▶ 241]	Binary counte r readin g
		---	data[4] =		<u>IV</u> [▶ 329]	<u>CA</u> [▶ 330]	<u>CY</u> [▶ 330]										Sequence	Quality descri ptor
		---	data[5. .7]	=													CP24Time2a [▶ 534]	Three octets binary time tag
		---	data[8. .IEC87 0_MA X_AS DU_D ATA_ BYTE] =															Reser ved

**4.1.2.9.3 M\_IT\_TB\_1**

Integrated total with CP56Time2a time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType	= 0x25 (37)	= M_IT_TB_1	<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[▶ 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[▶ 509]</a>

---	- info												INFORMATION OBJECT ID [▶ 485]	
	---	objAddress											Information object address	
	---	stream											Information element/object data [▶ 485]	
		---	length	= 12										
		---	data			7	6	5	4	3	2	1	0	
		---	data[0..3]	=		BCR [▶ 241]							Binary counter reading	
		---	data[4]	=		IV [▶ 329]	CA [▶ 330]	CY [▶ 330]	Sequence					Quality descriptor
		---	data[5..11]	=		CP56Time2a [▶ 531]							Seven octets binary time tag	
		---	data[12..IEC870_MAX_ASDU_DATA_BYT_E]	=									Reserved	

**4.1.2.10 Single command**

**4.1.2.10.1 C\_SC\_NA\_1**

Single command without time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[► 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[► 484]</a>
	---	eType = 0x2D (45)	= C_SC_NA_1		<a href="#">Type identification</a> <a href="#">[► 500]</a>
	---	bSQ	= FALSE		Sequence of information objects
	---	nObj	= 1		Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[► 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[► 509]</a>

---	- info												INFORMATION OBJECT [▶ 485]
	---	objAddr											Information object address
	---	stream											Information element/object data [▶ 485]
		---	length	= 1									
		---	data		7	6	5	4	3	2	1	0	
		---	data[0] =	S/E [▶ 241]	QU [▶ 524]						0	SCS [▶ 520]	SCO = Single command
		---	data[1..IEC870_MAX_DATA_BYTE] =										Reserved

**4.1.2.10.2 C\_SC\_TA\_1**

Single command with CP56Time2a time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reser ved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType = 0x3A (58)	= C_SC_TA_1		<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ	= FALSE		Seque nce of inform ation object s
	---	nObj	= 1		Numb er of object s
	---	bT			Test
	---	bPN			Positiv e/ negati ve confir mation / activati on
	---	eCOT			<a href="#">Cause of transm ission</a> <a href="#">[▶ 509]</a>
	---	nORG			Origin ator address s
	---	asduA ddr			Comm on address s of asdu
	---	eClass			<a href="#">Fifo priorit y class</a> <a href="#">[▶ 509]</a>

---	- info												<a href="#">INFORMATION OBJECT I</a> <a href="#">[▶ 485]</a>	
	---	objAddr											Information object address	
	---	stream											Information element/ object data <a href="#">[▶ 485]</a>	
		---	length	= 8										
		---	data			7	6	5	4	3	2	1	0	
		---	data[0] =	<a href="#">S/E</a> <a href="#">[▶ 241]</a>				<a href="#">QU</a> <a href="#">[▶ 524]</a>				0	<a href="#">SCS</a> <a href="#">[▶ 520]</a>	<a href="#">SCO</a> = Single command
		---	data[1..7] =					<a href="#">CP56Time2a</a> <a href="#">[▶ 531]</a>						Seven octets binary time tag
		---	data[8..IEC870_MAX_DU_DATA_BYTE] =											Reserved

**4.1.2.11 Double command**

**4.1.2.11.1 C\_DC\_NA\_1**

Double command without time tag.



- obj					<a href="#">ASDU object</a> <a href="#">[► 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[► 484]</a>
	---	eType	= 0x2E (46)	= C_DC_NA_1	<a href="#">Type identification</a> <a href="#">[► 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[► 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[► 509]</a>

---	- info												INFOR MATIO N OBJEC T [▶ 485]
	---	objAdd r											Inform ation objec t addres s
	---	stream											Inform ation elemen t/ objec t data [▶ 485]
		---	length = 1										
		---	data		7	6	5	4	3	2	1	0	
		---	data[0] =	S/E [▶ 241]	QU [▶ 524]						DCS [▶ 520]		DCO = Doubl e comm and
		---	data[1. .IEC87 0_MA X_AS DU_D ATA_ BYTE] =										Reser ved

**4.1.2.11.2 C\_DC\_TA\_1**

Double command with CP56Time2a time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType = 0x3B (59)	= C_DC_TA_1		<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ	= FALSE		Sequence of information objects
	---	nObj	= 1		Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[▶ 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[▶ 509]</a>

---	- info													INFOR MATIO N OBJEC T [▶ 485]
	---	objAdd r												Inform ation object addres s
	---	stream												Inform ation elemen t/ object data [▶ 485]
	---	length	= 8											
	---	data		7	6	5	4	3	2	1	0			
	---	data[0]	=	S/E [▶ 241]	QU [▶ 524]				DCS [▶ 520]				DCO = Doubl e comm and	
	---	data[1. .7]	=	CP56Time2a [▶ 531]										Seven octets binary time tag
	---	data[8. .IEC87 0_MA X_AS DU_D ATA_ BYTE]	=											Reser ved

**4.1.2.12 Regulating step command**

**4.1.2.12.1 C\_RC\_NA\_1**

Regulating step command without time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[► 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[► 484]</a>
	---	eType	= 0x2F (47)	= C_RC_NA_1	<a href="#">Type identification</a> <a href="#">[► 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[► 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[► 509]</a>

---	- info												INFOR MATIO N OBJEC T [▶ 485]
	---	objAdd r											Inform ation object addres s
	---	stream											Inform ation elemen t/ object data [▶ 485]
		---	length	= 1									
		---	data		7	6	5	4	3	2	1	0	
		---	data[0] =	S/E [▶ 241]	QU [▶ 524]						RCS [▶ 526]		RCO = Regul ating step comm and
		---	data[1. .IEC87 0_MA X_AS DU_D ATA_ BYTE] =										Reser ved

**4.1.2.12.2 C\_RC\_TA\_1**

Regulating step command with CP56Time2a time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType	= 0x3C (60)	= C_RC_TA_1	<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[▶ 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[▶ 509]</a>

---	- info												INFORMATION OBJECT <a href="#">[► 485]</a>
	---	objAddr											Information object address
	---	stream											Information element/object data <a href="#">[► 485]</a>
	---	length	= 8										
	---	data		7	6	5	4	3	2	1	0		
	---	data[0] =	S/E <a href="#">[► 241]</a>	QU <a href="#">[► 524]</a>					RCS <a href="#">[► 526]</a>			RCO = Regulating step command	
	---	data[1..7] =		CP56Time2a <a href="#">[► 531]</a>								Seven octets binary time tag	
	---	data[8..IEC870_MAX_DU_DATA_BYTE] =										Reserved	

**4.1.2.13 Set-point command, normalized value**

**4.1.2.13.1 C\_SE\_NA\_1**

Set-point command, normalized value without time tag.



- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType	= 0x30 (48)	= C_SE_NA_1	<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[▶ 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[▶ 509]</a>



- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType = 0x3D (61)	= C_SE_TA_1		<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ	= FALSE		Sequence of information objects
	---	nObj	= 1		Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[▶ 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[▶ 509]</a>



- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType	= 0x31 (49)	= C_SE_NB_1	<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[▶ 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[▶ 509]</a>



- obj					<a href="#">ASDU object</a> <a href="#">[► 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[► 484]</a>
	---	eType = 0x3E (62)	= C_SE_TB_1		<a href="#">Type identification</a> <a href="#">[► 500]</a>
	---	bSQ	= FALSE		Sequence of information objects
	---	nObj	= 1		Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[► 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[► 509]</a>





- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType	= 0x32 (50)	= C_SE_NC_1	<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[▶ 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[▶ 509]</a>

---	- info												<u>INFORMATION OBJECT ID</u> [▶ 485]	
	---	objAddr											Information object address	
	---	stream											<u>Information element/object data</u> [▶ 485]	
		---	length	= 5										
		---	data			7	6	5	4	3	2	1	0	
		---	data[0..3] =			<u>R32 [▶ 241]</u>							Short floating point value	
		---	data[4] =	<u>S/E [▶ 241]</u>		<u>QL [▶ 522]</u>							QOS = Qualifier of set-point command	
		---	data[5..IEC870_MAX_AXIS_DATA_BYTE] =											Reserved

**4.1.2.15.2 C\_SE\_TC\_1**

Set-point command, short floating point value with CP56Time2a time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType	= 0x3F (63)	= C_SE_TC_1	<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[▶ 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[▶ 509]</a>

---	- info																		INFORMATION OBJECT I <a href="#">[► 485]</a>			
	---	objAddr																	Information object address			
	---	stream																	Information element/object data <a href="#">[► 485]</a>			
		---	length	= 12																		
		---	data			7	6	5	4	3	2	1	0									
		---	data[0..3]	=		<a href="#">R32 [► 241]</a>														Short floating point value		
		---	data[4]	=	<a href="#">S/E [► 241]</a>	<a href="#">QL [► 522]</a>															QOS = Qualifier of command	
		---	data[5..11]	=		<a href="#">CP56Time2a [► 531]</a>																Seven octets binary time tag
		---	data[12..IEC870_MAX_AX_SDU_DATA_BYT_E]	=																	Reserved	

**4.1.2.16 Bitstring command**

**4.1.2.16.1 C\_BO\_NA\_1**

Bitstring of 32 bits without time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType	= 0x33 (51)	= C_BO_NA_1	<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[▶ 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[▶ 509]</a>

---	- info												INFORMATION OBJECT [▶ 485]
	---	objAddr											Information object address
	---	stream											Information element/object data [▶ 485]
		---	length = 4										
		---	data		7	6	5	4	3	2	1	0	
		---	data[0..3] =	BSI [▶ 241]									Binary state information
		---	data[4..IEC870_MAX_DDU_DATA_BYTE] =										Reserved

### 4.1.2.16.2 C\_BO\_TA\_1

Bitstring of 32 bits with CP56Time2a time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType	= 0x40 (64)	= C_BO_TA_1	<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[▶ 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[▶ 509]</a>

---	- info																				<u>INFORMATION OBJECT</u> <u>IT</u> <a href="#">[► 485]</a>	
	---	objAddr																			Information object address	
	---	stream																			Information element/object data <a href="#">[► 485]</a>	
		---	length	= 11																		
		---	data			7	6	5	4	3	2	1	0									
			---	data[0..3] =																	<u>BSI</u> <a href="#">[► 241]</a>	Binary state information
			---	data[4..10] =																	<u>CP56Time2a</u> <a href="#">[► 531]</a>	Seven octets binary time tag
			---	data[11..IEC 870_M AX_A SDU_DATA_BYTE] =																		Reserved

**4.1.2.17 Test command**

**4.1.2.17.1 C\_TS\_NA\_1**

Test command without time tag.



- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reser ved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType	= 0x68 (104)	= C_TS_NA_1	<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ		= FALSE	Seque nce of inform ation object s
	---	nObj		= 1	Numb er of object s
	---	bT			Test
	---	bPN			Positiv e/ negati ve confir mation / activati on
	---	eCOT			<a href="#">Cause of transm ission</a> <a href="#">[▶ 509]</a>
	---	nORG			Origin ator address s
	---	asduA ddr			Comm on address s of asdu
	---	eClass			<a href="#">Fifo priorit y class</a> <a href="#">[▶ 509]</a>

---	- info													INFORMATION OBJECT I [▶ 485]
	---	objAddr	= 0											Information object addresses
	---	stream												Information element/object data [▶ 485]
		---	length = 2											
		---	data		7	6	5	4	3	2	1	0		
		---	data[0] = 0xAA	1	0	1	0	1	0	1	0			FBP [▶ 241] =
		---	data[1] = 0x55	0	1	0	1	0	1	0	1			Fixed test pattern
		---	data[2..IEC870_MAX_AS DU_DATA_BYTE] =											Reserved

### 4.1.2.18 System information in monitor direction

#### 4.1.2.18.1 M\_EI\_NA\_1

End of initialization.

- obj					<a href="#">ASDU object</a> <a href="#">[► 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[► 484]</a>
	---	eType	= 0x46 (70)	= M_EI_NA_1	<a href="#">Type identification</a> <a href="#">[► 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[► 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[► 509]</a>

---	- info																INFOR MATIO N OBJEC T [► 485]		
	---	objAdd r	= 0														Inform ation objec t addres s		
	---	stream															Inform ation elemen t/ objec t data [► 485]		
		---	length = 1																
		---	data		7	6	5	4	3	2	1	0							
		---	data[0] =	LPC [► 240]	COL [► 521]														Cause of initializ ation
		---	data[1. .IEC87 0_MA X_AS DU_D ATA_ BYTE] =															Reser ved	

### 4.1.2.19 System information in control direction

#### 4.1.2.19.1 C\_CS\_NA\_1

Clock synchronisation command

- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType	= 0x67 (103)	= C_CS_NA_1	<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[▶ 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[▶ 509]</a>

---	- info																					<u>INFORMATION OBJECT I</u> <a href="#">[▶ 485]</a>		
	---	objAddr	= 0																			Information object address		
	---	stream																				<u>Information element/object data</u> <a href="#">[▶ 485]</a>		
		---	length = 7																					
		---	data		7	6	5	4	3	2	1	0												
		---	data[0..6] =	<a href="#">CP56Time2a [▶ 531]</a>																			Seven octets binary time tag	
		---	data[7..IEC870_MAX_AXIS_DATA_BYTE] =																					Reserved

#### 4.1.2.19.2 C\_IC\_NA\_1

Interrogation command.

- obj					<a href="#">ASDU object</a> <a href="#">[► 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[► 484]</a>
	---	eType	= 0x64 (100)	= C_IC_NA_1	<a href="#">Type identification</a> <a href="#">[► 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[► 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[► 509]</a>

---	- info																					INFOR MATIO N OBJEC T [▶ 485]
	---	objAdd r	= 0																			Inform ation object addres s
	---	stream																				Inform ation elemen t/ object data [▶ 485]
		---	length = 1																			
		---	data		7	6	5	4	3	2	1	0										
			---	data[0] =																		Qualifi er of interro gation
			---	data[1. .IEC87 0_MA X_AS DU_D ATA_ BYTE] =																		Reser ved

**4.1.2.19.3 C\_CI\_NA\_1**

Counter interrogation command.



- obj					<a href="#">ASDU object</a> <a href="#">[► 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[► 484]</a>
	---	eType	= 0x65 (101)	= C_CI_NA_1	<a href="#">Type identification</a> <a href="#">[► 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[► 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[► 509]</a>

---	- info												INFOR MATIO N OBJEC T [► 485]	
	---	objAdd r		= 0									Inform ation object address	
	---	stream											Inform ation eleme nt/ object data [► 485]	
	---	length		= 1										
	---	data			7	6	5	4	3	2	1	0		
	---	data[0] =		FRZ [► 522]									RQT [► 523]	QCC = Qualifi er of counte r interro gation
	---	data[1. .IEC87 0_MA X_AS DU_D ATA_ BYTE] =											Reser ved	

#### 4.1.2.19.4 C\_RP\_NA\_1

Reset process command.

- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType	= 0x69 (105)	= C_RP_NA_1	<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[▶ 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[▶ 509]</a>

---	- info																		<u>INFORMATION OBJECT I</u> <a href="#">[▶ 485]</a>
	---	objAddr	= 0																Information object address
	---	stream																	Information element/object data <a href="#">[▶ 485]</a>
		---	length = 1																
		---	data		7	6	5	4	3	2	1	0							
			---	data[0] =															Qualifier of reset process
			---	data[1. .IEC87 0_MAX_AS DU_DATA_BYTE] =															Reserved

**4.1.2.19.5 C\_RD\_NA\_1**

Read command.

- obj					<a href="#">ASDU object</a> <a href="#">[► 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[► 484]</a>
	---	eType	= 0x66 (102)	= C_RD_NA_1	<a href="#">Type identification</a> <a href="#">[► 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[► 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[► 509]</a>

---	- info																<u>INFORMATION OBJECT</u> <u>IT</u> <a href="#">▶ 485</a>
	---	objAddress															Information object address
	---	stream															<u>Information element/object data</u> <a href="#">▶ 485</a>
		---	length	= 0													
		---	data			7	6	5	4	3	2	1	0				
			---	data[0..IEC870_MAX_AS DU_DATA_BYTE]													Reserved

#### 4.1.2.20 Protection equipment information

##### 4.1.2.20.1 M\_EP\_TA\_1

Event of protection equipment with CP24Time2a time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType	= 0x11 (17)	= M_EP_TA_1	<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[▶ 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[▶ 509]</a>

---	- info												INFORMATION OBJECT ID [▶ 485]
	---	objAddr											Information object address
	---	stream											Information element/object data [▶ 485]
		---	length = 6										
		---	data		7	6	5	4	3	2	1	0	
		---	data[0] =	IV [▶ 329]	NT [▶ 329]	SB [▶ 329]	BL [▶ 329]	EI [▶ 330]	0		ES [▶ 524]		SEP = Single event of protection equipment
		---	data[1..2] =	CP16Time2a [▶ 534]									Elapsed time, two octets binary time
		---	data[3..5] =	CP24Time2a [▶ 534]									Three octets binary time tag
		---	data[6..IEC870_MAX_AS DU_DATA_BYTE] =										Reserved

**4.1.2.20.2 M\_EP\_TB\_1**

Packed start events of protection equipment with CP24Time2a time tag.



- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reser ved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType	= 0x12 (18)	= M_EP_TB_1	<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ		= FALSE	Seque nce of inform ation object s
	---	nObj		= 1	Numb er of object s
	---	bT			Test
	---	bPN			Positiv e/ negati ve confir mation / activati on
	---	eCOT			<a href="#">Cause of transm ission</a> <a href="#">[▶ 509]</a>
	---	nORG			Origin ator address s
	---	asduA ddr			Comm on address s of asdu
	---	eClass			<a href="#">Fifo priorit y class</a> <a href="#">[▶ 509]</a>

---	- info												INFORMATION OBJECT [▶ 485]
	---	objAddr											Information object address
	---	stream											Information element/object data [▶ 485]
		---	length = 7										
		---	data		7	6	5	4	3	2	1	0	
		---	data[0] =	0	0	SRD	SIE	SL3	SL2	SL1	GS		SEP = Start events of protection equipment
		---	data[1] =	<u>IV</u> [▶ 329]	<u>NT</u> [▶ 329]	<u>SB</u> [▶ 329]	<u>BL</u> [▶ 329]	<u>EI</u> [▶ 330]	0	0	0		QDP = Quality descriptor for events of protection equipment
		---	data[2..3] =	<u>CP16Time2a</u> [▶ 534]									Relay duration time, two octets binary time
		---	data[4..6] =	<u>CP24Time2a</u> [▶ 534]									Three octets binary time tag

			---	data[7. .IEC87 0_MA X_AS DU_D ATA_ BYTE] =		Reser ved
--	--	--	-----	---	--	--------------

**4.1.2.20.3 M\_EP\_TC\_1**

Packed output circuit information of protection equipment with CP24Time2a time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[► 483]</a>
---	+ head				Reser ved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[► 484]</a>
	---	eType	= 0x13 (19)	= M_EP_TC_1	<a href="#">Type identification</a> <a href="#">[► 500]</a>
	---	bSQ		= FALSE	Seque nce of inform ation object s
	---	nObj		= 1	Numb er of object s
	---	bT			Test
	---	bPN			Positiv e/ negati ve confir mation / activati on
	---	eCOT			<a href="#">Cause of transm ission</a> <a href="#">[► 509]</a>
	---	nORG			Origin ator address s
	---	asduA ddr			Comm on address s of asdu
	---	eClass			<a href="#">Fifo priorit y class</a> <a href="#">[► 509]</a>

---	- info												INFORMATION OBJECT [▶ 485]
	---	objAddr											Information object address
	---	stream											Information element/object data [▶ 485]
		---	length = 7										
		---	data		7	6	5	4	3	2	1	0	
		---	data[0] =	0	0	0	0	CL3	CL2	CL1	GC		OCI = Output circuit information of protection equipment
		---	data[1] =	<u>IV</u> [▶ 329]	<u>NT</u> [▶ 329]	<u>SB</u> [▶ 329]	<u>BL</u> [▶ 329]	<u>EI</u> [▶ 330]	0	0	0		QDP = Quality descriptor for events of protection equipment
		---	data[2..3] =	<u>CP16Time2a</u> [▶ 534]									Relay operating time, two octets binary time
		---	data[4..6] =	<u>CP24Time2a</u> [▶ 534]									Three octets binary time tag

			---	data[7. .IEC87 0_MA X_AS DU_D ATA_ BYTE] =		Reser ved
--	--	--	-----	---	--	--------------

**4.1.2.20.4 M\_EP\_TD\_1**

Event of protection equipment with CP56Time2a time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[► 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[► 484]</a>
	---	eType	= 0x26 (38)	= M_EP_TD_1	<a href="#">Type identification</a> <a href="#">[► 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[► 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[► 509]</a>

---	- info												INFOR MATIO N OBJEC T [▶ 485]
	---	objAdd r											Inform ation object addres s
	---	stream											Inform ation elemen t/ object data [▶ 485]
	---	length	= 10										
	---	data		7	6	5	4	3	2	1	0		
	---	data[0]	=	<u>IV</u> [▶ 329]	<u>NT</u> [▶ 329]	<u>SB</u> [▶ 329]	<u>BL</u> [▶ 329]	<u>EI</u> [▶ 330]	0	<u>ES</u> [▶ 524]			SEP = Single event of protect ion equip ment
	---	data[1..2]	=	<u>CP16Time2a</u> [▶ 534]									Elapse d time, two octets binary time
	---	data[3..9]	=	<u>CP56Time2a</u> [▶ 531]									Sevent octets binary time tag
	---	data[10..IEC 870_M AX_A SDU_ DATA _BYT E]	=										Reser ved

**4.1.2.20.5 M\_EP\_TE\_1**

Packed start events of protection equipment with CP56Time2a time tag.



- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reser ved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType	= 0x27 (39)	= M_EP_TE_1	<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ		= FALSE	Seque nce of inform ation object s
	---	nObj		= 1	Numb er of object s
	---	bT			Test
	---	bPN			Positiv e/ negati ve confir mation / activati on
	---	eCOT			<a href="#">Cause of transm ission</a> <a href="#">[▶ 509]</a>
	---	nORG			Origin ator address s
	---	asduA ddr			Comm on address s of asdu
	---	eClass			<a href="#">Fifo priorit y class</a> <a href="#">[▶ 509]</a>

---	- info												INFORMATION OBJECT [▶ 485]
	---	objAddr											Information object address
	---	stream											Information element/object data [▶ 485]
		---	length = 11										
		---	data		7	6	5	4	3	2	1	0	
		---	data[0] =	0	0	SRD	SIE	SL3	SL2	SL1	GS		SEP = Start events of protection equipment
		---	data[1] =	<u>IV</u> [▶ 329]	<u>NT</u> [▶ 329]	<u>SB</u> [▶ 329]	<u>BL</u> [▶ 329]	<u>EI</u> [▶ 330]	0	0	0		QDP = Quality descriptor for events of protection equipment
		---	data[2..3] =	<u>CP16Time2a</u> [▶ 534]									Relay duration time, two octets binary time
		---	data[4..10] =	<u>CP56Time2a</u> [▶ 531]									Seven octets binary time tag

			---	data[1 1..IEC 870_M AX_A SDU_ DATA_ BYT E] =		Reser ved
--	--	--	-----	---	--	--------------

**4.1.2.20.6 M\_EP\_TF\_1**

Packed output circuit information of protection equipment with CP56Time2a time tag.

- obj					<a href="#">ASDU object</a> <a href="#">[► 483]</a>
---	+ head				Reser ved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[► 484]</a>
	---	eType	= 0x28 (40)	= M_EP_TF_1	<a href="#">Type identification</a> <a href="#">[► 500]</a>
	---	bSQ		= FALSE	Seque nce of inform ation object s
	---	nObj		= 1	Numb er of object s
	---	bT			Test
	---	bPN			Positiv e/ negati ve confir mation / activati on
	---	eCOT			<a href="#">Cause of transm ission</a> <a href="#">[► 509]</a>
	---	nORG			Origin ator address s
	---	asduA ddr			Comm on address s of asdu
	---	eClass			<a href="#">Fifo priorit y class</a> <a href="#">[► 509]</a>

---	- info													INFORMATION OBJECT [▶ 485]
	---	objAddr												Information object address
	---	stream												Information element/object data [▶ 485]
		---	length = 11											
		---	data		7	6	5	4	3	2	1	0		
		---	data[0] =	0	0	0	0	CL3	CL2	CL1	GC			OCI = Output circuit information of protection equipment
		---	data[1] =	<u>IV</u> [▶ 329]	<u>NT</u> [▶ 329]	<u>SB</u> [▶ 329]	<u>BL</u> [▶ 329]	<u>EI</u> [▶ 330]	0	0	0			QDP = Quality descriptor for events of protection equipment
		---	data[2..3] =	<u>CP16Time2a</u> [▶ 534]										Relay operating time, two octets binary time
		---	data[4..10] =	<u>CP56Time2a</u> [▶ 531]										Seven octets binary time tag

			---	data[1 1..IEC 870_M AX_A SDU_ DATA_ BYT E] =		Reser ved
--	--	--	-----	---	--	--------------

**4.1.2.20.7 M\_PS\_NA\_1**

Packed single point information with status change detection.

- obj					<a href="#">ASDU object</a> <a href="#">[► 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[► 484]</a>
	---	eType	= 0x14 (20)	= M_PS_NA_1	<a href="#">Type identification</a> <a href="#">[► 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[► 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[► 509]</a>

---	- info																	INFORMATION OBJECT [▶ 485]	
	---	objAddr																Information object addresses	
	---	stream																Information element/object data [▶ 485]	
		---	length	= 5															
		---	data			7	6	5	4	3	2	1	0						
		---	data[0..3] =			SCD													Status and status change detection (32 bit)
		---	data[4] =		IV [▶ 329]	NT [▶ 329]	SB [▶ 329]	BL [▶ 329]									OV [▶ 329]	QDS: Quality descriptor	
		---	data[5..IEC870_MAX_AS_DUDATA_BYTE] =															Reserved	

### 4.1.2.21 Parameter loading/activation

#### 4.1.2.21.1 P\_ME\_NA\_1

Parameter of measured value, normalized value.



- obj					<a href="#">ASDU object</a> <a href="#">[► 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[► 484]</a>
	---	eType	= 0x6E (110)	= P_ME_NA_1	<a href="#">Type identification</a> <a href="#">[► 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[► 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[► 509]</a>

---	- info															<u>INFORMATION OBJECT I</u> <a href="#">[▶ 485]</a>	
	---	objAddr														Information object addresses	
	---	stream														<u>Information element/object data</u> <a href="#">[▶ 485]</a>	
	---	length	= 3														
	---	data		7	6	5	4	3	2	1	0						
	---	data[0..1]	=	<u>NVA</u> <a href="#">[▶ 241]</a>												<u>Normalized value</u>	
	---	data[2]	=	<u>LPC</u> <a href="#">[▶ 240]</a>	POP	<u>KPA</u> <a href="#">[▶ 525]</a>											<u>QPM: Qualifier of parameter of measured value</u>
	---	data[3..IEC870_MAX_AS DU_DATA_BYTE]	=													<u>Reserved</u>	

**4.1.2.21.2 P\_ME\_NB\_1**

Parameter of measured value, scaled value.

- obj					<a href="#">ASDU object</a> <a href="#">[► 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[► 484]</a>
	---	eType	= 0x6F (111)	= P_ME_NB_1	<a href="#">Type identification</a> <a href="#">[► 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[► 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[► 509]</a>

---	- info												<u>INFORMATION OBJECT I</u> <a href="#">[► 485]</a>
	---	objAddr											Information object addresses
	---	stream											<u>Information element/object data</u> <a href="#">[► 485]</a>
	---	length	= 3										
	---	data			7	6	5	4	3	2	1	0	
	---	data[0..1]	=	<u>SVA</u> <a href="#">[► 241]</a>									Scaled value
	---	data[2]	=	<u>LPC</u> <a href="#">[► 240]</a>	POP	<u>KPA</u> <a href="#">[► 525]</a>							QPM: Qualifier of parameter of measured value
	---	data[3..IEC870_MAX_AXIS_DATA_BYTE]	=										Reserved

**4.1.2.21.3 P\_ME\_NC\_1**

Parameter of measured value, short floating point value.

- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType	= 0x70 (112)	= P_ME_NC_1	<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[▶ 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[▶ 509]</a>

---	- info																	INFOR MATIO N OBJEC T [▶ 485]	
	---	objAdd r																Inform ation objec t addres s	
	---	stream																Inform ation elemen t/ objec t data [▶ 485]	
		---	length	= 5															
		---	data			7	6	5	4	3	2	1	0						
		---	data[0. .3] =			R32 [▶ 241]													Short floatin g point value
		---	data[4] =		LPC [▶ 240]	POP		KPA [▶ 525]											QPM: Qualifi er of param eter of measu red value
		---	data[5. .IEC87 0_MA X_AS DU_D ATA_ BYTE] =																Reser ved

**4.1.2.21.4 P\_AC\_NA\_1**

Parameter activation.

- obj					<a href="#">ASDU object</a> <a href="#">[▶ 483]</a>
---	+ head				Reserved
---	- ident				<a href="#">DATA UNIT IDENTIFIER</a> <a href="#">[▶ 484]</a>
	---	eType	= 0x71 (113)	= P_AC_NA_1	<a href="#">Type identification</a> <a href="#">[▶ 500]</a>
	---	bSQ		= FALSE	Sequence of information objects
	---	nObj		= 1	Number of objects
	---	bT			Test
	---	bPN			Positive/negative confirmation / activation
	---	eCOT			<a href="#">Cause of transmission</a> <a href="#">[▶ 509]</a>
	---	nORG			Originator address
	---	asduAddr			Common address of asdu
	---	eClass			<a href="#">Fifo priority class</a> <a href="#">[▶ 509]</a>

---	- info																	<u>INFORMATION OBJECT I</u> <a href="#">[► 485]</a>
	---	objAddr		= 0														Information object address
	---	stream																Information element/object data <a href="#">[► 485]</a>
		---	length	= 1														
		---	data			7	6	5	4	3	2	1	0					
			---	data[0] =														Qualifier of parameter activation
			---	data[1..IEC870_MAX_AS DU_DATA_BYTE] =														Reserved

**4.1.2.22 Information elements**

**4.1.2.22.1 LPC**

Local parameter change flag.

- <0> = No change;
- <1> = Changed;

**4.1.2.22.2 VTI**

Value with transient state indication (8 bits).

Transient state (bit 7):

- <0> = equipment is not in transient state;
- <1> = equipment is in transient state;

Value (bits 0..6) = <-64..63>;



**4.1.2.22.3 BSI**

Bitstring of 32 bits.

**4.1.2.22.4 NVA**

Normalized value.

**4.1.2.22.5 SVA**

Scaled value.

**4.1.2.22.6 R32**

Short floating point value.

**4.1.2.22.7 BCR**

Binary counter reading.

**4.1.2.22.8 S/E**

Select/execute state.

- <0> = Ausführen (execute);
- <1> = Anwählen (select);

**4.1.2.22.9 FBP**

Fixed test pattern.

**4.1.2.22.10 TSC**

Test command counter.

## 4.2 IEC 60870-5-102

TF6500 implementiert eine Schnittstelle über die einzelne IEC 60870-5-102 ASDUs (Dienstdateneinheiten der Anwendungsschicht) über die seriellen Schnittstellen des PCs und die Beckhoff KL6xxx-/EL6xxx-Klemmen verschickt und empfangen werden können.

Die Endapplikation (Implementierung einer Zentralstation) wird auf der Software-Schnittstelle der SPS-Bibliothek aufgesetzt. Anders als bei den 101/104-TwinCAT Supplement Produkten steht dem SPS-Programmierer bei der IEC60870-5-102-Protokollimplementierung nur die sogenannte "Low level"-Schnittstelle zur Verfügung. Der Grund: Die meisten Unterstationen (Zählwerteinrichtungen) besitzen viele Herstellerspezifische oder Gerätespezifische Parameterdaten oder Informationen. Die "Low level"-Schnittstelle ermöglicht einen uneingeschränkten Zugriff auf all diese Daten. Im Folgenden werden die Eigenschaften dieser Schnittstelle kurz beschrieben.

### "Low level"-Schnittstelle:

Diese Schnittstelle setzt bei dem Protokollstack tiefer an und ermöglicht das Versenden und/oder Empfangen einzelner Frames (ASDU's). Innerhalb der Protokollstruktur liegt diese Schnittstelle oberhalb der Verbindungsschicht und implementiert bereits die nötigen Prozeduren und Übertragungstelegrammformate. Anwendungsfunktionen wie z.B. Abruf der einzelnen Zählerstände oder der Produktidentifikation sind in der Schnittstelle nicht implementiert, der Anwender kann aber mit Hilfe der Schnittstelle diese Anwendungsfunktionen selber implementieren.

**Pro:** Sehr flexibel; Alle Eigenschaften im ASDU-Frame können manipuliert werden (z.B. information object address, record address usw); Weil nur die benötigten Dienste (abhängig vom Gerät) implementiert werden kann eine hohe Performance erreicht werden; Hohe Performance bei vielen Daten bzw. Zählwerten;

**Contra:** Größerer Programmieraufwand; Alle Dienste wie Generalabfrage, Zählerabfrage, Uhrzeitsynchronisation, Befehlsausführung usw. müssen von dem SPS-Programmierer selbst implementiert (ausprogrammiert) werden; Der SPS-Programmierer muss mit der Protokollnorm vertraut sein.

Protokollstruktur des Endsystems:

#### Protokollstruktur des Endsystems:

Auswahl von Anwendungsfunktionen aus IEC 60870-5-5	Anwenderprozess
Auswahl von ASDU (Dienstdateneinheiten der Anwendungsschicht) aus IEC 60870-5-3, IEC 60870-5-4 und IEC 60870-5-102	Anwendungsschicht (7)
N/A	Darstellungsschicht (6)
	Sitzungsschicht (5)
	Transportschicht (4)
	Vermittlungsschicht (3)
unbalanced IEC 60870-5-2 IEC 60870-5-1 (FT 1.2)	Verbindungsschicht (2)
EIA RS485, RS232 (V.24), Fibre Optics	Physikalische Schicht (1)



Die Schichten 3 bis 6 werden nicht benutzt.

### 4.2.1 IEC 60870-5-102 telegram structure

- FT 1.2 frame format is used;
- Frames with fixed and variable block length and single control characters are used;
- Hamming distance = 4;

#### Frame with variable length

This frame type is used to transmit user data between controlling and controlled station

byte/bit	7	6	5	4	3	2	1	0						
0	Start byte 1 (0x68)								Header		LPCI	LPDU		
1	Block length													
2	Block length (copy)													
3	Start byte 2 (0x68)													
4	RES	PRM	FCB / ACD	FCV / DFC	Function code			Control field						
5	Link address fields (0, 1 or 2 octets)													
	<u>Type identification</u> [▶ 503]								DATA UNIT IDENTIFIER		ASDU			
	SQ	Number of object												
	T	P/N	Cause of transmission [▶ 512]											
	ASDU address fields (1 or 2 octets)													
	Record address													
	Information object address							Info-object						
	Information elements													
n-1	Checksum								Tail				LPCI	
n	Stop byte (0x16)													

**Frame with fixed length**

This frame type is used for link layer services. Sometimes used as an confirmation.

byte/bit	7	6	5	4	3	2	1	0		
0	Start byte 1 (0x10)									
1	RES	PRM	FCB / ACD	FCV / DFC	Function code			Control field		
2	Link address (0, 1 or 2 octets)									
n-1	Checksum									
n	Stop byte (0x16)									

**Single character data**

Single characters are used to confirm link layer and user data transmissions.

- 0xE5 (positive acknowledge);

## **4.2.2 ASDU-Objektbeschreibung**

### **4.2.2.1 Standard IEC 60870-5-102 Datentypen**

Type	Dez	Hex	Beschreibung
ASDU_TYPEUNDEF_2	0	0x00	Wird nicht verwendet
<a href="#">M SP TA 2  ▶ 247 </a>	1	0x01	Einzelmeldung mit Zeitstempel
<a href="#">M IT TA 2  ▶ 248 </a>	2	0x02	Abrechnungszählerstände, je 4 Oktette
<a href="#">M IT TB 2  ▶ 250 </a>	3	0x03	Abrechnungszählerstände, je 3 Oktette
<a href="#">M IT TC 2  ▶ 252 </a>	4	0x04	Abrechnungszählerstände, je 2 Oktette
<a href="#">M IT TD 2  ▶ 254 </a>	5	0x05	Periodisch rückgesetzte Abrechnungszählerstände, je 4 Oktette
<a href="#">M IT TE 2  ▶ 256 </a>	6	0x06	Periodisch rückgesetzte Abrechnungszählerstände, je 3 Oktette
<a href="#">M IT TF 2  ▶ 258 </a>	7	0x07	Periodisch rückgesetzte Abrechnungszählerstände, je 2 Oktette
<a href="#">M IT TG 2  ▶ 260 </a>	8	0x08	Betriebszählerstände, je 4 Oktette
<a href="#">M IT TH 2  ▶ 262 </a>	9	0x09	Betriebszählerstände, je 3 Oktette
<a href="#">M IT TI 2  ▶ 264 </a>	10	0x0A	Betriebszählerstände, je 2 Oktette
<a href="#">M IT TK 2  ▶ 265 </a>	11	0x0B	Periodisch rückgesetzte Betriebszählerstände, je 4 Oktette
<a href="#">M IT TL 2  ▶ 266 </a>	12	0x0C	Periodisch rückgesetzte Betriebszählerstände, je 3 Oktette
<a href="#">M IT TM 2  ▶ 267 </a>	13	0x0D	Periodisch rückgesetzte Betriebszählerstände, je 2 Oktette
	14..69	0x0E..0x45	Reserviert (Standardbereich)
<a href="#">M EI NA 2  ▶ 268 </a>	70	0x46	Initialisierungsende
<a href="#">P MP NA 2  ▶ 268 </a>	71	0x47	Hersteller- und Produktspezifikation der Zähler-DEE
<a href="#">M TI TA 2  ▶ 269 </a>	72	0x48	Aktuelle Systemzeit der Zähler-DEE
	73..99	0x49..0x63	Reserviert (Standardbereich)
<a href="#">C RD NA 2  ▶ 270 </a>	100	0x64	Abruf der Hersteller und Produktspezifikation
<a href="#">C SP NA 2  ▶ 271 </a>	101	0x65	Abruf einer Liste von Einzelmeldungen mit Zeitstempel
<a href="#">C SP NB 2  ▶ 272 </a>	102	0x66	Abruf einer Liste von Einzelmeldungen mit Zeitstempel eines ausgewählten Zeitbereichs
<a href="#">C TI NA 2  ▶ 273 </a>	103	0x67	Abruf der aktuellen Systemzeit der Zähler-DEE
<a href="#">C CI NA 2  ▶ 274 </a>	104	0x68	Abruf der Abrechnungszählerstände der ältesten Messperiode
<a href="#">C CI NB 2  ▶ 275 </a>	105	0x69	Abruf der Abrechnungszählerstände der ältesten Messperiode und eines ausgewählten Adressbereichs
<a href="#">C CI NC 2  ▶ 276 </a>	106	0x6A	Abruf der Abrechnungszählerstände einer bestimmten vergangenen Messperiode
<a href="#">C CI ND 2  ▶ 277 </a>	107	0x6B	Abruf der Abrechnungszählerstände einer bestimmten vergangenen Messperiode und eines ausgewählten Adressbereichs
<a href="#">C CI NE 2  ▶ 278 </a>	108	0x6C	Abruf periodisch rückgesetzter Abrechnungszählerstände der ältesten Messperiode
<a href="#">C CI NF 2  ▶ 279 </a>	109	0x6D	Abruf periodisch rückgesetzter Abrechnungszählerstände der ältesten Messperiode und eines ausgewählten Adressbereichs
<a href="#">C CI NG 2  ▶ 280 </a>	110	0x6E	Abruf periodisch rückgesetzter Abrechnungszählerstände einer bestimmten vergangenen Messperiode

Type	Dez	Hex	Beschreibung
<a href="#">C_CI_NH_2</a> [ <a href="#">▶ 281</a> ]	111	0x6F	Abruf periodisch rückgesetzter Abrechnungszählerstände einer bestimmten vergangenen Messperiode und eines ausgewählten Adressbereichs
<a href="#">C_CI_NI_2</a> [ <a href="#">▶ 282</a> ]	112	0x70	Abruf der Betriebszählerstände der ältesten Messperiode
<a href="#">C_CI_NK_2</a> [ <a href="#">▶ 283</a> ]	113	0x71	Abruf der Betriebszählerstände der ältesten Messperiode und eines ausgewählten Adressbereichs
<a href="#">C_CI_NL_2</a> [ <a href="#">▶ 284</a> ]	114	0x72	Abruf der Betriebszählerstände einer bestimmten vergangenen Messperiode
<a href="#">C_CI_NM_2</a> [ <a href="#">▶ 285</a> ]	115	0x73	Abruf der Betriebszählerstände einer bestimmten vergangenen Messperiode und eines ausgewählten Adressbereichs
<a href="#">C_CI_NN_2</a> [ <a href="#">▶ 286</a> ]	116	0x74	Abruf periodisch rückgesetzter Betriebszählerstände der ältesten Messperiode
<a href="#">C_CI_NO_2</a> [ <a href="#">▶ 287</a> ]	117	0x75	Abruf periodisch rückgesetzter Betriebszählerstände der ältesten Messperiode und eines ausgewählten Adressbereichs
<a href="#">C_CI_NP_2</a> [ <a href="#">▶ 288</a> ]	118	0x76	Abruf periodisch rückgesetzter Betriebszählerstände einer bestimmten vergangenen Messperiode
<a href="#">C_CI_NQ_2</a> [ <a href="#">▶ 289</a> ]	119	0x77	Abruf periodisch rückgesetzter Betriebszählerstände einer bestimmten vergangenen Messperiode und eines ausgewählten Adressbereichs
<a href="#">C_CI_NR_2</a> [ <a href="#">▶ 290</a> ]	120	0x78	Abruf der Abrechnungszählerstände eines ausgewählten Zeitbereichs und eines ausgewählten Adressbereichs
<a href="#">C_CI_NS_2</a> [ <a href="#">▶ 291</a> ]	121	0x79	Abruf periodisch rückgesetzter Abrechnungszählerstände eines ausgewählten Zeitbereichs und eines ausgewählten Adressbereichs
<a href="#">C_CI_NT_2</a> [ <a href="#">▶ 292</a> ]	122	0x7A	Abruf der Betriebszählerstände eines ausgewählten Zeitbereichs und eines ausgewählten Adressbereichs
<a href="#">C_CI_NU_2</a> [ <a href="#">▶ 293</a> ]	123	0x7B	Abruf periodisch rückgesetzter Betriebszählerstände eines ausgewählten Zeitbereichs und eines ausgewählten Adressbereichs
	124..127	0x7C..0x7F	Reserviert (Standardbereich)
<a href="#">M_DS_TA_2</a>	128	0x80	-
<a href="#">P_ME_NA_2</a>	129	0x81	Parameters of the measuring point
<a href="#">M_DS_TB_2</a>	130	0x82	-
<a href="#">M_CH_TA_2</a>	131	0x83	-
<a href="#">C_PK_2</a>	132	0x84	Load private key
<a href="#">C_TA_VC_2</a>	133	0x85	Read tariff information ( current values )
<a href="#">C_TA_VM_2</a>	134	0x86	Read tariff information ( stored values )
<a href="#">M_TA_VC_2</a>	135	0x87	Tariff information ( current values )
<a href="#">M_TA_VM_2</a>	136	0x88	Tariff information ( stored values )
<a href="#">C_TA_CP_2</a>	137	0x89	Close accounting period
<a href="#">M_IB_TG_2</a>	139	0x8B	Block of operational integrated totals ( absolute values )
<a href="#">M_IB_TK_2</a>	140	0x8C	Block of periodical reset operational integrated totals ( increment values )
<a href="#">C_RM_NA_2</a>	141	0x8D	Read configuration data of the meter device
<a href="#">M_RM_NA_2</a>	142	0x8E	Configuration of the meter device
<a href="#">C_MR_NA_2</a>	143	0x8F	Change configuration data of the meter device
<a href="#">C_PC_NA_2</a>	144	0x90	-

Type	Dez	Hex	Beschreibung
M_PC_NA_2	145	0x91	-
C_MC_NA_2	146	0x92	-
C_DF_NA_2	147	0x93	-
M_DF_NA_2	148	0x94	-
C_MF_NA_2	149	0x95	-
	150..179	0x96..0xB3	Reserviert
C_DS_TA_2	180	0xB4	-
C_CS_TA_2	181	0xB5	Change date and time ( Time synchronization )
C_PI_NA_2	182	0xB6	Read parameters of the measuring point
C_AC_NA_2	183	0xB7	Start session and send access key
C_DS_TB_2	184	0xB8	-
C_CH_TA_2	185	0xB9	-
C_MH_TA_2	186	0xBA	-
C_FS_NA_2	187	0xBB	Finish session
C_MP_NA_2	188	0xBC	-
C_CB_NT_2	189	0xBD	Read a block of operational integrated totals of a time period and a selected address
C_CB_UN_2	190	0xBE	Read a block of periodical reset operational integrated totals of a time period and a selected address
	191..255	0xBF..0xFF	Reserviert

#### 4.2.2.2 Prozessinformation in Überwachungsrichtung

##### 4.2.2.2.1 M\_SP\_TA\_2

Single-point information with time tag.

- obj					ASDU object [▶ 545]								
---	+ head				Reserved								
---	- ident				DATA UNIT IDENTIFIER [▶ 546]								
	---	eType	= 0x1 (1)	= M_SP_TA_2	Type identification [▶ 503]								
	---	bSQ		= FALSE	Sequence of information objects								
	---	nObj		= i	Number of objects								
	---	bT			Test								
	---	bPN			Positive/negative confirmation/activation								
	---	eCOT			Cause of transmission [▶ 512]								
	---	asduAddr			Common address of asdu								
	---	eClasses			Fifo priority class [▶ 509]								
---	- info				INFORMATION OBJECT [▶ 545]								
	---	rcdAddr			Record address								
	---	stream			Information element/object data [▶ 485]								
	---	length	= i * 9										
	---	data			7	6	5	4	3	2	1	0	
	---	data[0]	= SPA [▶ 294]		Single-point address							Information object 1	
	---	data[1]	= SPQ [▶ 294]		SPI [▶ 526]				SPQ = Single-point qualifier SPI = Single-point information				
	---	data[2..8]	= CP56Time2b [▶ 532]		Seven octets binary time tag								
	---	data[9..length - 1]	=									Information object 2..i	
	---	data[length.IEC870_MAX_ASDU_DATA_BYTES]	=		Reserved								

**4.2.2.2 M\_IT\_TA\_2**

Accounting integrated totals, 4 octets each



- obj					ASDU object [▶ 545]									
---	+ head				Reserved									
---	- ident				DATA UNIT IDENTIFIER [▶ 546]									
	---	eType	= 0x2 (2)	= M_IT_TA_2	Type identification [▶ 503]									
	---	bSQ		= FALSE	Sequence of information objects									
	---	nObj		= i	Number of objects									
	---	bT			Test									
	---	bPN			Positive/negative confirmation/activation									
	---	eCOT			Cause of transmission [▶ 512]									
	---	asduAddr			Common address of asdu									
	---	eClasses			Fifo priority class [▶ 509]									
---	- info				INFORMATION OBJECT [▶ 545]									
	---	rcdAddr			Record address									
	---	stream			Information element/object data [▶ 485]									
	---	length	= i * ( 6 + [ 1 (signature) ] ) + 5											
	---	data			7	6	5	4	3	2	1	0		
	---	data[0]	=	IOA	Information object address								Information object 1	
	---	data[1..4]	=	CR4	Counter reading, 4 octets									
	---	data[5]	=	IV [▶ 329] 1	CA [▶ 330] 1	CY [▶ 330] 1	S e q u e n c e	Quality descriptor						
	---	data[6]	=	Signature	Signature (optional)									
	---	data[7..length - 6]	=										Information object 2..i	
	---	data[length - 5..length - 1]	=	CP40Time2a [▶ 533]	Five octets binary time tag									
	---	data[length..IEC870_MAX_ASDU_DATA_BYTE]	=		Reserved									

#### **4.2.2.2.3 M\_IT\_TB\_2**

Accounting integrated totals, 3 octets each.

- obj																			ASDU object [▶ 545]
---	+ head																		Reserved
---	- ident																		DATA UNIT IDENTIFIER [▶ 546]
	---	eType	= 0x3 (3)	= M_IT_TB_2															Type identification [▶ 503]
	---	bSQ		= FALSE															Sequence of information objects
	---	nObj		= i															Number of objects
	---	bT																	Test
	---	bPN																	Positive/negative confirmation/activation
	---	eCOT																	Cause of transmission [▶ 512]
	---	asduAddr																	Common address of asdu
	---	eClasses																	Fifo priority class [▶ 509]
---	- info																		INFORMATION OBJECT [▶ 545]
	---	rcdAddr																	Record address
	---	stream																	Information element/object data [▶ 485]
	---	length	= i * ( 5 + [ 1 (signature) ] ) + 5																
	---	data																	
	---	data[0]	=	IOA															Information object address
	---	data[1..3]	=	CR3															Counter reading, 3 octets
	---	data[4]	=	IV [▶ 329]	CA [▶ 330]	CY [▶ 330]	S e q u e n c e												Quality descriptor
	---	data[5]	=	Signature															Signature (optional)
	---	data[6..length - 6]	=																Information object 2..i
	---	data[length - 5..length - 1]	=	CP40Time2a [▶ 533]															Five octets binary time tag
	---	data[length..IEC870_MAX_ASDU_DATA_BYTE]	=																Reserved

#### **4.2.2.2.4 M\_IT\_TC\_2**

Accounting integrated totals, 2 octets each.

- obj																			ASDU object [▶ 545]	
---	+ head																		Reserved	
---	- ident																		DATA UNIT IDENTIFIER [▶ 546]	
	---	eType	= 0x4 (4)	= M_IT_TC_2															Type identification [▶ 503]	
	---	bSQ		= FALSE															Sequence of information objects	
	---	nObj		= i															Number of objects	
	---	bT																	Test	
	---	bPN																	Positive/negative confirmation/activation	
	---	eCOT																	Cause of transmission [▶ 512]	
	---	asduAddr																	Common address of asdu	
	---	eClass																	Fifo priority class [▶ 509]	
---	- info																		INFORMATION OBJECT [▶ 545]	
	---	rcdAddr																	Record address	
	---	stream																	Information element/object data [▶ 485]	
	---	length	= i * ( 4 + [ 1 (signature) ] ) + 5																	
	---	data																		
	---	data[0]	= IOA																Information object address	
	---	data[1..2]	= CR2																Counter reading, 2 octets	
	---	data[3]	=	IV [▶ 329]	CA [▶ 330]	CY [▶ 330]	S												Quality descriptor	
	---	data[4]	=	Signature																Signature (optional)
	---	data[5..length - 6]	=																Information object 2..i	
	---	data[length - 5..length - 1]	=	CP40Time2a [▶ 533]																Five octets binary time tag
	---	data[length..IEC870_MAX_ASDU_DATA_BYTE]	=																Reserved	

#### **4.2.2.2.5 M\_IT\_TD\_2**

Periodical reset accounting integrated totals, 4 octets each.

- obj					ASDU object [▶ 545]									
---	+ head				Reserved									
---	- ident				DATA UNIT IDENTIFIER [▶ 546]									
	---	eType	= 0x5 (5)	= M_IT_TD_2	Type identification [▶ 503]									
	---	bSQ		= FALSE	Sequence of information objects									
	---	nObj		= i	Number of objects									
	---	bT			Test									
	---	bPN			Positive/negative confirmation/activation									
	---	eCOT			Cause of transmission [▶ 512]									
	---	asduAddr			Common address of asdu									
	---	eClasses			Fifo priority class [▶ 509]									
---	- info				INFORMATION OBJECT [▶ 545]									
	---	rcdAddr			Record address									
	---	stream			Information element/object data [▶ 485]									
	---	length	= i * ( 6 + [ 1 (signature) ] ) + 5											
	---	data			7	6	5	4	3	2	1	0		
	---	data[0]	=	IOA	Information object address								Information object 1	
	---	data[1..4]	=	CR4	Counter reading, 4 octets									
	---	data[5]	=	IV [▶ 329] 1	CA [▶ 330] 1	CY [▶ 330] 1	S e q u e n c e	Quality descriptor						
	---	data[6]	=	Signature	Signature (optional)									
	---	data[7..length - 6]	=										Information object 2..i	
	---	data[length - 5..length - 1]	=	CP40Time2a [▶ 533]	Five octets binary time tag									
	---	data[length..IEC870_MAX_ASDU_DATA_BYTE]	=		Reserved									

#### **4.2.2.2.6 M\_IT\_TE\_2**

Periodical reset accounting integrated totals, 3 octets each.



- obj																			ASDU object [▶ 545]
---	+ head																		Reserved
---	- ident																		DATA UNIT IDENTIFIER [▶ 546]
	---	eType	= 0x6 (6)	= M_IT_TE_2															Type identification [▶ 503]
	---	bSQ		= FALSE															Sequence of information objects
	---	nObj		= i															Number of objects
	---	bT																	Test
	---	bPN																	Positive/negative confirmation/activation
	---	eCOT																	Cause of transmission [▶ 512]
	---	asduAddr																	Common address of asdu
	---	eClass																	Fifo priority class [▶ 509]
---	- info																		INFORMATION OBJECT [▶ 545]
	---	rcdAddr																	Record address
	---	stream																	Information element/object data [▶ 485]
	---	length	= i * ( 5 + [ 1 (signature) ] ) + 5																
	---	data				7	6	5	4	3	2	1	0						
	---	data[0]	= IOA																Information object address
	---	data[1..3]	= CR3																Counter reading, 3 octets
	---	data[4]	=	IV [▶ 329]	CA [▶ 330]	CY [▶ 330]	S	e	q	u	e	n	c	e					Quality descriptor
	---	data[5]	=	Signature															Signature (optional)
	---	data[6..length - 6]	=																Information object 2..i
	---	data[length - 5..length - 1]	=	CP40Time2a [▶ 533]															Five octets binary time tag
	---	data[length..IEC870_MAX_ASDU_DATA_BYTE]	=																Reserved

#### **4.2.2.2.7 M\_IT\_TF\_2**

Periodical reset accounting integrated totals, 2 octets each.

- obj																ASDU object [▶ 545]	
---	+ head															Reserved	
---	- ident															DATA UNIT IDENTIFIER [▶ 546]	
	---	eType	= 0x7 (7)	= M_IT_TF_2												Type identification [▶ 503]	
	---	bSQ		= FALSE												Sequence of information objects	
	---	nObj		= i												Number of objects	
	---	bT														Test	
	---	bPN														Positive/negative confirmation/activation	
	---	eCOT														Cause of transmission [▶ 512]	
	---	asduAddr														Common address of asdu	
	---	eClass														Fifo priority class [▶ 509]	
---	- info															INFORMATION OBJECT [▶ 545]	
	---	rcdAddr														Record address	
	---	stream														Information element/object data [▶ 485]	
	---	length	= i * ( 4 + [ 1 (signature) ] ) + 5														
	---	data															
	---	data[0]	= IOA													Information object address	
	---	data[1..2]	= CR2													Counter reading, 2 octets	
	---	data[3]	=	IV [▶ 329]	CA [▶ 330]	CY [▶ 330]	S	sequence								Quality descriptor	
	---	data[4]	=	Signature													Signature (optional)
	---	data[5..length - 6]	=													Information object 2..i	
	---	data[length - 5..length - 1]	=	CP40Time2a [▶ 533]													Five octets binary time tag
	---	data[length..IEC870_MAX_ASDU_DATA_BYTE]	=													Reserved	

#### **4.2.2.2.8 M\_IT\_TG\_2**

Operational integrated totals, 4 octets each.

- obj																			ASDU object [▶ 545]
---	+ head																		Reserved
---	- ident																		DATA UNIT IDENTIFIER [▶ 546]
	---	eType	= 0x8 (8)	= M_IT_TG_2															Type identification [▶ 503]
	---	bSQ		= FALSE															Sequence of information objects
	---	nObj		= i															Number of objects
	---	bT																	Test
	---	bPN																	Positive/negative confirmation/activation
	---	eCOT																	Cause of transmission [▶ 512]
	---	asduAddr																	Common address of asdu
	---	eClass																	Fifo priority class [▶ 509]
---	- info																		INFORMATION OBJECT [▶ 545]
	---	rcdAddr																	Record address
	---	stream																	Information element/object data [▶ 485]
		---	length	= i * ( 6 + [ 1 (signature) ] ) + 5															
		---	data			7	6	5	4	3	2	1	0						
		---	data[0]	= IOA															Information object address
		---	data[1..4]	= CR4															Counter reading, 4 octets
		---	data[5]	=	IV [▶ 329]	CA [▶ 330]	CY [▶ 330]	Sequence											Quality descriptor
		---	data[6]	= Signature															Signature (optional)
		---	data[7..length - 6]	=															Information object 2..i
		---	data[length - 5..length - 1]	=	CP40Time2a [▶ 533]														Five octets binary time tag
		---	data[length..IEC870_MAX_ASDU_DATA_BYTE]	=															Reserved

#### **4.2.2.2.9 M\_IT\_TH\_2**

Operational integrated totals, 3 octets each.

- obj																ASDU object [▶ 545]	
---	+ head															Reserved	
---	- ident															DATA UNIT IDENTIFIER [▶ 546]	
	---	eType	= 0x9 (9)	= M_IT_TH_2												Type identification [▶ 503]	
	---	bSQ		= FALSE												Sequence of information objects	
	---	nObj		= i												Number of objects	
	---	bT														Test	
	---	bPN														Positive/negative confirmation/activation	
	---	eCOT														Cause of transmission [▶ 512]	
	---	asduAddr														Common address of asdu	
	---	eClass														Fifo priority class [▶ 509]	
---	- info															INFORMATION OBJECT [▶ 545]	
	---	rcdAddr														Record address	
	---	stream														Information element/object data [▶ 485]	
	---	length	= i * ( 5 + [ 1 (signature) ] ) + 5														
	---	data															
	---	data[0]	= IOA													Information object address	
	---	data[1..3]	= CR3													Counter reading, 3 octets	
	---	data[4]	=	IV [▶ 329] 1	CA [▶ 330] 1	CY [▶ 330] 1	Sequence									Quality descriptor	
	---	data[5]	=	Signature													Signature (optional)
	---	data[6..length - 6]	=													Information object 2..i	
	---	data[length - 5..length - 1]	=	CP40Time2a [▶ 533]													Five octets binary time tag
	---	data[length..IEC870_MAX_ASDU_DATA_BYTE]	=													Reserved	

**4.2.2.2.10 M\_IT\_TI\_2**

Operational integrated totals, 2 octets each.

- obj													ASDU object [▶ 545]
---	+ head												Reserved
---	- ident												DATA UNIT IDENTIFIER [▶ 546]
	---	eType	= 0xA (10)	= M_IT_TI_2									Type identification [▶ 503]
	---	bSQ		= FALSE									Sequence of information objects
	---	nObj		= i									Number of objects
	---	bT											Test
	---	bPN											Positive/negative confirmation/activation
	---	eCOT											Cause of transmission [▶ 512]
	---	asduAddr											Common address of asdu
	---	eClasses											Fifo priority class [▶ 509]
---	- info												INFORMATION OBJECT [▶ 545]
	---	rcdAddr											Record address
	---	stream											Information element/object data [▶ 485]
	---	length	= i * ( 4 + [ 1 (signature) ] ) + 5										
	---	data		7	6	5	4	3	2	1	0		
	---	data[0]	= IOA					Information object address				Information object 1	
	---	data[1..2]	= CR2					Counter reading, 2 octets					
	---	data[3]	=	IV [▶ 329] 1	CA [▶ 330] 1	CY [▶ 330] 1	Se qu en ce	Quality descriptor					
	---	data[4]	= Signature					Signature (optional)					
	---	data[5..length - 6]	=									Information object 2..i	
	---	data[length - 5..length - 1]	=	CP40Time2a [▶ 533]				Five octets binary time tag					
	---	data[length..IEC870_MAX_ASDU_DATA_BYTE]	=					Reserved					



**4.2.2.2.11 M\_IT\_TK\_2**

Periodical reset operational integrated totals, 4 octets each.

- obj													ASDU object [▶ 545]
---	+ head												Reserved
---	- ident												DATA UNIT IDENTIFIER [▶ 546]
	---	eType	= 0xB (11)	= M_IT_TK_2									Type identification [▶ 503]
	---	bSQ		= FALSE									Sequence of information objects
	---	nObj		= i									Number of objects
	---	bT											Test
	---	bPN											Positive/negative confirmation/activation
	---	eCOT											Cause of transmission [▶ 512]
	---	asduAddr											Common address of asdu
	---	eClasses											Fifo priority class [▶ 509]
---	- info												INFORMATION OBJECT [▶ 545]
	---	rcdAddr											Record address
	---	stream											Information element/object data [▶ 485]
	---	length	= i * ( 6 + [ 1 (signature) ] ) + 5										
	---	data		7	6	5	4	3	2	1	0		
		---	data[0]	IOA				Information object address				Information object 1	
		---	data[1..4]	CR4				Counter reading, 4 octets					
		---	data[5]	IV [▶ 329] 1	CA [▶ 330] 1	CY [▶ 330] 1	Se qu en ce	Quality descriptor					
		---	data[6]	Signature				Signature (optional)					
		---	data[7..length - 6]									Information object 2..i	
		---	data[length - 5..length - 1]	CP40Time2a [▶ 533]				Five octets binary time tag					
		---	data[length..IEC870_MAX_ASDU_DATA_BYTES]					Reserved					

**4.2.2.2.12 M\_IT\_TL\_2**

Periodical reset operational integrated totals, 3 octets each.

- obj															ASDU object [▶ 545]															
---	+ head														Reserved															
---	- ident														DATA UNIT IDENTIFIER [▶ 546]															
	---	eType	= 0xC (12)	= M_IT_TL_2											Type identification [▶ 503]															
	---	bSQ		= FALSE											Sequence of information objects															
	---	nObj		= i											Number of objects															
	---	bT													Test															
	---	bPN													Positive/negative confirmation/activation															
	---	eCOT													Cause of transmission [▶ 512]															
	---	asduAddr													Common address of asdu															
	---	eClasses													Fifo priority class [▶ 509]															
---	- info														INFORMATION OBJECT [▶ 545]															
	---	rcdAddr													Record address															
	---	stream													Information element/object data [▶ 485]															
	---	length	= i * ( 5 + [ 1 (signature) ] ) + 5																											
	---	data		7	6	5	4	3	2	1	0																			
	---	data[0]	= IOA	Information object address										Information object 1																
	---	data[1..3]	= CR3	Counter reading, 3 octets																										
	---	data[4]	= <table border="1" style="display: inline-table; vertical-align: middle;"><tr><td>IV</td><td>CA</td><td>CY</td><td>Se</td></tr><tr><td>[▶ 329]</td><td>[▶ 330]</td><td>[▶ 330]</td><td>qu</td></tr><tr><td>1</td><td>1</td><td>1</td><td>en</td></tr><tr><td></td><td></td><td></td><td>ce</td></tr></table>	IV	CA	CY	Se	[▶ 329]	[▶ 330]	[▶ 330]	qu	1	1		1	en				ce	Quality descriptor									
IV	CA	CY	Se																											
[▶ 329]	[▶ 330]	[▶ 330]	qu																											
1	1	1	en																											
			ce																											
	---	data[5]	= Signature	Signature (optional)																										
	---	data[6..length - 6]	=											Information object 2..i																
	---	data[length - 5..length - 1]	=	CP40Time2a [▶ 533]										Five octets binary time tag																
	---	data[length..IEC870_MAX_ASDU_DATA_BYTE]	=											Reserved																

4.2.2.2.13 M\_IT\_TM\_2

Periodical reset operational integrated totals, 2 octets each.

- obj																ASDU object [▶ 545]										
---	+ head															Reserved										
---	- ident															DATA UNIT IDENTIFIER [▶ 546]										
	---	eType	= 0xD (13)	= M_IT_TM_2												Type identification [▶ 503]										
	---	bSQ		= FALSE												Sequence of information objects										
	---	nObj		= i												Number of objects										
	---	bT														Test										
	---	bPN														Positive/negative confirmation/activation										
	---	eCOT														Cause of transmission [▶ 512]										
	---	asduAddr														Common address of asdu										
	---	eClasses														Fifo priority class [▶ 509]										
---	- info															INFORMATION OBJECT [▶ 545]										
	---	rcdAddr														Record address										
	---	stream														Information element/object data [▶ 485]										
	---	length	= i * ( 4 + [ 1 (signature) ] ) + 5																							
	---	data			7	6	5	4		3	2	1	0													
		---	data[0]	= IOA									Information object address	Information object 1												
		---	data[1..2]	= CR2									Counter reading, 2 octets													
		---	data[3]	= <table border="1" style="display: inline-table; vertical-align: middle;"><tr><td>IV</td><td>CA</td><td>CY</td><td>Sequence</td></tr><tr><td>[▶ 329]</td><td>[▶ 330]</td><td>[▶ 330]</td><td></td></tr><tr><td>1</td><td>1</td><td>1</td><td></td></tr></table>	IV	CA	CY	Sequence	[▶ 329]	[▶ 330]	[▶ 330]		1		1	1										Quality descriptor
IV	CA	CY	Sequence																							
[▶ 329]	[▶ 330]	[▶ 330]																								
1	1	1																								
		---	data[4]	= Signature									Signature (optional)													
		---	data[5..length - 6]	=										Information object 2..i												
		---	data[length - 5..length - 1]	= CP40Time2a [▶ 533]									Five octets binary time tag													
		---	data[length..IEC870_MAX_ASDU_DATA_BYTES]	=									Reserved													

### 4.2.2.3 Systeminformation in Überwachungsrichtung

#### 4.2.2.3.1 M\_EI\_NA\_2

End of initialization.

- obj															ASDU object [▶ 545]
---	+ head														Reserved
---	- ident														DATA UNIT IDENTIFIER [▶ 546]
	---	eType	= 0x46 (70)	= M_EI_NA_2											Type identification [▶ 503]
	---	bSQ		= FALSE											Sequence of information objects
	---	nObj		= 1											Number of objects
	---	bT													Test
	---	bPN													Positive/negative confirmation/activation
	---	eCOT													Cause of transmission [▶ 512]
	---	asduA addr													Common address of asdu
	---	eClass													Fifo priority class [▶ 509]
---	- info														INFORMATION OBJECT [▶ 545]
	---	rcdAd dr		= 0											Record address
	---	stream													Information element/object data [▶ 485]
		---	length	= 2											
		---	data			7	6	5	4	3	2	1	0		
		---	data[0]	= 0											Address of information object
		---	data[1]	=	LPC [▶ 240]		COI [▶ 521]								COI = Cause of initialization
		---	data[2. .IEC87 0_MA X_AS DU_D ATA_ BYTE]	=											Reserved

#### 4.2.2.3.2 P\_MP\_NA\_2

Manufacturer and product specification of integrated total DTE.

- obj					ASDU object [► 545]								
---	+ head				Reserved								
---	- ident				DATA UNIT IDENTIFIER [► 546]								
	---	eType	= 0x47 (71)	= P_MP_NA_2	Type identification [► 503]								
	---	bSQ		= FALSE	Sequence of information objects								
	---	nObj		= 1	Number of objects								
	---	bT			Test								
	---	bPN			Positive/negative confirmation/activation								
	---	eCOT			Cause of transmission [► 512]								
	---	asduAddr			Common address of asdu								
	---	eClass			Fifo priority class [► 509]								
---	- info				INFORMATION OBJECT [► 545]								
	---	rcdAddr		= 0	Record address								
	---	stream			Information element/object data [► 485]								
	---	length	= 6										
	---	data			7	6	5	4	3	2	1	0	
	---	data[0]	=	YEAR	MONT		Date of standard						
	---	data[1]	=		Manufacturer code								
	---	data[2..5]	=		Product code								
	---	data[6..IEC870_MAX_AS_DDU_DATA_BYTE]	=		Reserved								

**4.2.2.3.3 M\_TI\_TA\_2**

Current system time of integrated total DTE.

- obj															ASDU object [▶ 545]
---	+ head														Reserved
---	- ident														DATA UNIT IDENTIFIER [▶ 546]
	---	eType	= 0x48 (72)	= M_TI_TA_2											Type identification [▶ 503]
	---	bSQ		= FALSE											Sequence of information objects
	---	nObj		= 1											Number of objects
	---	bT													Test
	---	bPN													Positive/negative confirmation/activation
	---	eCOT													Cause of transmission [▶ 512]
	---	asduA addr													Common address of asdu
	---	eClass													Fifo priority class [▶ 509]
---	- info														INFORMATION OBJECT [▶ 545]
	---	rcdAd dr		= 0											Record address
	---	stream													Information element/object data [▶ 485]
	---	length	= 7												
	---	data			7	6	5	4	3	2	1	0			
	---	data[0. .6]	=		CP56Time2b [▶ 532]									Seven octets binary time tag	
	---	data[7. .IEC87 0_MA X_AS DU_D ATA_ BYTE]	=												Reserved

#### 4.2.2.4 Systeminformation in Steuerungsrichtung

##### 4.2.2.4.1 C\_RD\_NA\_2

Read manufacturer and product specification.

- obj					ASDU object [▶ 545]							
---	+ head				Reserved							
---	- ident				DATA UNIT IDENTIFIER [▶ 546]							
	---	eType	= 0x64 (100)	= C_RD_NA_2	Type identification [▶ 503]							
	---	bSQ		= FALSE	Sequence of information objects							
	---	nObj		= 0	Number of objects							
	---	bT			Test							
	---	bPN			Positive/negative confirmation/activation							
	---	eCOT			Cause of transmission [▶ 512]							
	---	asduA ddr			Common address of asdu							
	---	eClass			Fifo priority class [▶ 509]							
---	- info				INFORMATION OBJECT [▶ 545]							
	---	rcdAd dr		= 0	Record address							
	---	stream			Information element/object data [▶ 485]							
	---	length		= 0								
	---	data			7	6	5	4	3	2	1	0
	---	data[0. .IEC87 0_MA X_AS DU_D ATA_ BYTE] =			Reserved							

**4.2.2.4.2 C\_SP\_NA\_2**

Read record of single-point information with time tag.

- obj															ASDU object [▶ 545]
---	+ head														Reserved
---	- ident														DATA UNIT IDENTIFIER [▶ 546]
	---	eType	= 0x65 (101)	= C_SP_NA_2											Type identification [▶ 503]
	---	bSQ		= FALSE											Sequence of information objects
	---	nObj		= 0											Number of objects
	---	bT													Test
	---	bPN													Positive/negative confirmation/activation
	---	eCOT													Cause of transmission [▶ 512]
	---	asduA ddr													Common address of asdu
	---	eClass													Fifo priority class [▶ 509]
---	- info														INFORMATION OBJECT [▶ 545]
	---	rcdAd dr													Record address
	---	stream													Information element/object data [▶ 485]
		---	length	= 0											
		---	data			7	6	5	4	3	2	1	0		
		---	data[0. .IEC87 0_MA X_AS DU_D ATA_ BYTE] =												Reserved

#### 4.2.2.4.3 C\_SP\_NB\_2

Read record of single-point information with time tag of selected time range.



- obj											ASDU object [▶ 545]	
---	+ head										Reserved	
---	- ident										DATA UNIT IDENTIFIER [▶ 546]	
	---	eType	= 0x66 (102)	= C_SP_NB_2							Type identification [▶ 503]	
	---	bSQ		= FALSE							Sequence of information objects	
	---	nObj		= 1							Number of objects	
	---	bT									Test	
	---	bPN									Positive/negative confirmation/activation	
	---	eCOT									Cause of transmission [▶ 512]	
	---	asduA addr									Common address of asdu	
	---	eClass									Fifo priority class [▶ 509]	
---	- info										INFORMATION OBJECT [▶ 545]	
	---	rcdAd dr									Record address	
	---	stream									Information element/object data [▶ 485]	
	---	length	= 10									
	---	data			7	6	5	4	3	2	1	0
		---	data[0. .4] =		CP40Time2a [▶ 533]					Five octets binary time tag (from)		
		---	data[5. .9] =		CP40Time2a [▶ 533]					Five octets binary time tag (until)		
		---	data[1 0..IEC 870_M AX_A SDU_ DATA _BYT E] =							Reserved		

**4.2.2.4.4 C\_TI\_NA\_2**

Read current system time of integrated total data terminal equipment.

- obj															ASDU object [▶ 545]
---	+ head														Reserved
---	- ident														DATA UNIT IDENTIFIER [▶ 546]
	---	eType	= 0x67 (103)	= C_TI_NA_2											Type identification [▶ 503]
	---	bSQ		= FALSE											Sequence of information objects
	---	nObj		= 0											Number of objects
	---	bT													Test
	---	bPN													Positive/negative confirmation/activation
	---	eCOT													Cause of transmission [▶ 512]
	---	asduA ddr													Common address of asdu
	---	eClass													Fifo priority class [▶ 509]
---	- info														INFORMATION OBJECT [▶ 545]
	---	rcdAd dr		= 0											Record address
	---	stream													Information element/object data [▶ 485]
	---	length		= 0											
	---	data				7	6	5	4	3	2	1	0		
	---	data[0. .IEC87 0_MA X_AS DU_D ATA_ BYTE] =													Reserved

#### 4.2.2.4.5 C\_CI\_NA\_2

Read accounting integrated totals of the oldest integration period.

- obj															ASDU object [▶ 545]
---	+ head														Reserved
---	- ident														DATA UNIT IDENTIFIER [▶ 546]
	---	eType	= 0x68 (104)	= C_CI_NA_2											Type identification [▶ 503]
	---	bSQ		= FALSE											Sequence of information objects
	---	nObj		= 0											Number of objects
	---	bT													Test
	---	bPN													Positive/negative confirmation/activation
	---	eCOT													Cause of transmission [▶ 512]
	---	asduA ddr													Common address of asdu
	---	eClass													Fifo priority class [▶ 509]
---	- info														INFORMATION OBJECT [▶ 545]
	---	rcdAd dr													Record address of accounting period
	---	stream													Information element/object data [▶ 485]
		---	length	= 0											
		---	data			7	6	5	4	3	2	1	0		
		---	data[0. .IEC87 0_MA X_AS DU_D ATA_ BYTE] =												Reserved

**4.2.2.4.6 C\_CI\_NB\_2**

Read accounting integrated totals of the oldest integration period and of selected range of addresses.

- obj											ASDU object [▶ 545]	
---	+ head										Reserved	
---	- ident										DATA UNIT IDENTIFIER [▶ 546]	
	---	eType	= 0x69 (105)	= C_CI_NB_2							Type identification [▶ 503]	
	---	bSQ		= FALSE							Sequence of information objects	
	---	nObj		= 1							Number of objects	
	---	bT									Test	
	---	bPN									Positive/negative confirmation/activation	
	---	eCOT									Cause of transmission [▶ 512]	
	---	asduA addr									Common address of asdu	
	---	eClass									Fifo priority class [▶ 509]	
---	- info										INFORMATION OBJECT [▶ 545]	
	---	rcdAd dr									Record address of accounting period	
	---	stream									Information element/object data [▶ 485]	
	---	length	= 2									
	---	data			7	6	5	4	3	2	1	0
		---	data[0] =									Integrated total address (from)
		---	data[1] =									Integrated total address (to)
		---	data[2. .IEC87 0_MA X_AS DU_D ATA_ BYTE] =									Reserved

**4.2.2.4.7 C\_CI\_NC\_2**

Read accounting integrated totals of a specific past integration period.

- obj											ASDU object [▶ 545]	
---	+ head										Reserved	
---	- ident										DATA UNIT IDENTIFIER [▶ 546]	
	---	eType	= 0x6A (106)	= C_CI_NC_2							Type identification [▶ 503]	
	---	bSQ		= FALSE							Sequence of information objects	
	---	nObj		= 1							Number of objects	
	---	bT									Test	
	---	bPN									Positive/negative confirmation/activation	
	---	eCOT									Cause of transmission [▶ 512]	
	---	asduAddr									Common address of asdu	
	---	eClass									Fifo priority class [▶ 509]	
---	- info										INFORMATION OBJECT [▶ 545]	
	---	rcdAddr									Record address of accounting period	
	---	stream									Information element/object data [▶ 485]	
	---	length	= 5									
	---	data			7	6	5	4	3	2	1	0
	---	data[0..4]	=		CP40Time2a [▶ 533]					Five octets binary time tag		
	---	data[5..IEC870_MAX_ASDU_DATA_BYTE]	=									Reserved

**4.2.2.4.8 C\_CI\_ND\_2**

Read accounting integrated totals of a specific past integration period and of selected range of addresses.

- obj											ASDU object [▶ 545]	
---	+ head										Reserved	
---	- ident										DATA UNIT IDENTIFIER [▶ 546]	
	---	eType	= 0x6B (107)	= C_CI_ND_2							Type identification [▶ 503]	
	---	bSQ		= FALSE							Sequence of information objects	
	---	nObj		= 1							Number of objects	
	---	bT									Test	
	---	bPN									Positive/negative confirmation/activation	
	---	eCOT									Cause of transmission [▶ 512]	
	---	asduA addr									Common address of asdu	
	---	eClass									Fifo priority class [▶ 509]	
---	- info										INFORMATION OBJECT [▶ 545]	
	---	rcdAd dr									Record address of accounting period	
	---	stream									Information element/object data [▶ 485]	
	---	length	= 7									
	---	data			7	6	5	4	3	2	1	0
	---	data[0]	=									Integrated total address (from)
	---	data[1]	=									Integrated total address (to)
	---	data[2..6]	=	CP40Time2a [▶ 533]								Five octets binary time tag
	---	data[7..IEC870_MAX_AS DU_DATA_BYTE]	=									Reserved

**4.2.2.4.9 C\_CI\_NE\_2**

Read periodically reset accounting integrated totals of the oldest integration period.

- obj															ASDU object [▶ 545]
---	+ head														Reserved
---	- ident														DATA UNIT IDENTIFIER [▶ 546]
	---	eType	= 0x6C (108)	= C_CI_NE_2											Type identification [▶ 503]
	---	bSQ		= FALSE											Sequence of information objects
	---	nObj		= 0											Number of objects
	---	bT													Test
	---	bPN													Positive/negative confirmation/activation
	---	eCOT													Cause of transmission [▶ 512]
	---	asduAddr													Common address of asdu
	---	eClass													Fifo priority class [▶ 509]
---	- info														INFORMATION OBJECT [▶ 545]
	---	rcdAddr													Record address of accounting period
	---	stream													Information element/object data [▶ 485]
	---	length	= 0												
	---	data			7	6	5	4	3	2	1	0			
	---	data[0..IEC870_MAX_AS DU_DATA_BYTE]	=												Reserved

**4.2.2.4.10 C\_CI\_NF\_2**

Read periodically reset accounting integrated totals of the oldest integration period and of selected range of addresses.

- obj											ASDU object [▶ 545]		
---	+ head										Reserved		
---	- ident										DATA UNIT IDENTIFIER [▶ 546]		
	---	eType	= 0x6D (109)	= C_CI_NF_2							Type identification [▶ 503]		
	---	bSQ		= FALSE							Sequence of information objects		
	---	nObj		= 1							Number of objects		
	---	bT									Test		
	---	bPN									Positive/negative confirmation/activation		
	---	eCOT									Cause of transmission [▶ 512]		
	---	asduA addr									Common address of asdu		
	---	eClass									Fifo priority class [▶ 509]		
---	- info										INFORMATION OBJECT [▶ 545]		
	---	rcdAd dr									Record address of accounting period		
	---	stream									Information element/object data [▶ 485]		
	---	length	= 2										
	---	data			7	6	5	4	3	2	1	0	
		---	data[0]	=									Integrated total address (from)
		---	data[1]	=									Integrated total address (to)
		---	data[2..	.IEC87 0_MA X_AS DU_D ATA_ BYTE]	=								Reserved

**4.2.2.4.11 C\_CI\_NG\_2**

Read periodically reset accounting integrated totals of a specific past integration period.



- obj											ASDU object [▶ 545]	
---	+ head										Reserved	
---	- ident										DATA UNIT IDENTIFIER [▶ 546]	
	---	eType	= 0x6E (110)	= C_CI_NG_2							Type identification [▶ 503]	
	---	bSQ		= FALSE							Sequence of information objects	
	---	nObj		= 1							Number of objects	
	---	bT									Test	
	---	bPN									Positive/negative confirmation/activation	
	---	eCOT									Cause of transmission [▶ 512]	
	---	asduAddr									Common address of asdu	
	---	eClass									Fifo priority class [▶ 509]	
---	- info										INFORMATION OBJECT [▶ 545]	
	---	rcdAddr									Record address of accounting period	
	---	stream									Information element/object data [▶ 485]	
	---	length	= 5									
	---	data			7	6	5	4	3	2	1	0
	---	data[0..4]	=		CP40Time2a [▶ 533]					Five octets binary time tag		
	---	data[5..7]	=	.IEC870_MAX_ASDU_DATA_BYTE]						Reserved		

**4.2.2.4.12 C\_CI\_NH\_2**

Read periodically reset accounting integrated totals of a specific past integration period and of selected range of addresses.

- obj											ASDU object [▶ 545]	
---	+ head										Reserved	
---	- ident										DATA UNIT IDENTIFIER [▶ 546]	
	---	eType	= 0x6F (111)	= C_CI_NH_2							Type identification [▶ 503]	
	---	bSQ		= FALSE							Sequence of information objects	
	---	nObj		= 1							Number of objects	
	---	bT									Test	
	---	bPN									Positive/negative confirmation/activation	
	---	eCOT									Cause of transmission [▶ 512]	
	---	asduA ddr									Common address of asdu	
	---	eClass									Fifo priority class [▶ 509]	
---	- info										INFORMATION OBJECT [▶ 545]	
	---	rcdAd dr									Record address of accounting period	
	---	stream									Information element/object data [▶ 485]	
	---	length	= 7									
	---	data			7	6	5	4	3	2	1	0
	---	data[0]	=									
	---	data[1]	=									
	---	data[2. .6]	=	CP40Time2a [▶ 533]								
	---	data[7. .IEC87 0_MA X_AS DU_D ATA_ BYTE]	=									
												Reserved

**4.2.2.4.13 C\_CI\_NI\_2**

Read operational integrated totals of the oldest integration period.

- obj															ASDU object [▶ 545]
---	+ head														Reserved
---	- ident														DATA UNIT IDENTIFIER [▶ 546]
	---	eType	= 0x70 (112)	= C_CI_NI_2											Type identification [▶ 503]
	---	bSQ		= FALSE											Sequence of information objects
	---	nObj		= 0											Number of objects
	---	bT													Test
	---	bPN													Positive/negative confirmation/activation
	---	eCOT													Cause of transmission [▶ 512]
	---	asduA ddr													Common address of asdu
	---	eClass													Fifo priority class [▶ 509]
---	- info														INFORMATION OBJECT [▶ 545]
	---	rcdAd dr													Record address of accounting period
	---	stream													Information element/object data [▶ 485]
		---	length	= 0											
		---	data			7	6	5	4	3	2	1	0		
		---	data[0. .IEC87 0_MA X_AS DU_D ATA_ BYTE] =												Reserved

**4.2.2.4.14 C\_CI\_NK\_2**

Read operational integrated totals of the oldest integration period and of selected range of addresses.

- obj											ASDU object [▶ 545]	
---	+ head										Reserved	
---	- ident										DATA UNIT IDENTIFIER [▶ 546]	
	---	eType	= 0x71 (113)	= C_CI_NK_2							Type identification [▶ 503]	
	---	bSQ		= FALSE							Sequence of information objects	
	---	nObj		= 1							Number of objects	
	---	bT									Test	
	---	bPN									Positive/negative confirmation/activation	
	---	eCOT									Cause of transmission [▶ 512]	
	---	asduA ddr									Common address of asdu	
	---	eClass									Fifo priority class [▶ 509]	
---	- info										INFORMATION OBJECT [▶ 545]	
	---	rcdAd dr									Record address of accounting period	
	---	stream									Information element/object data [▶ 485]	
	---	length	= 2									
	---	data			7	6	5	4	3	2	1	0
		---	data[0] =									Integrated total address (from)
		---	data[1] =									Integrated total address (to)
		---	data[2. .IEC87 0_MA X_AS DU_D ATA_ BYTE] =									Reserved

#### 4.2.2.4.15 C\_CI\_NL\_2

Read operational integrated totals of a specific past integration period.

- obj											ASDU object [▶ 545]	
---	+ head										Reserved	
---	- ident										DATA UNIT IDENTIFIER [▶ 546]	
	---	eType	= 0x72 (114)	= C_CI_NL_2							Type identification [▶ 503]	
	---	bSQ		= FALSE							Sequence of information objects	
	---	nObj		= 1							Number of objects	
	---	bT									Test	
	---	bPN									Positive/negative confirmation/activation	
	---	eCOT									Cause of transmission [▶ 512]	
	---	asduA ddr									Common address of asdu	
	---	eClass									Fifo priority class [▶ 509]	
---	- info										INFORMATION OBJECT [▶ 545]	
	---	rcdAd dr									Record address of accounting period	
	---	stream									Information element/object data [▶ 485]	
	---	length	= 5									
	---	data			7	6	5	4	3	2	1	0
		---	data[0. .4] =		CP40Time2a [▶ 533]					Five octets binary time tag		
		---	data[5. .IEC87 0_MA X_AS DU_D ATA_ BYTE] =							Reserved		

**4.2.2.4.16 C\_CI\_NM\_2**

Read operational integrated totals of a specific past integration period and of selected range of addresses.

- obj											ASDU object [▶ 545]	
---	+ head										Reserved	
---	- ident										DATA UNIT IDENTIFIER [▶ 546]	
	---	eType	= 0x73 (115)	= C_CI_NN_2							Type identification [▶ 503]	
	---	bSQ		= FALSE							Sequence of information objects	
	---	nObj		= 1							Number of objects	
	---	bT									Test	
	---	bPN									Positive/negative confirmation/activation	
	---	eCOT									Cause of transmission [▶ 512]	
	---	asduA ddr									Common address of asdu	
	---	eClass									Fifo priority class [▶ 509]	
---	- info										INFORMATION OBJECT [▶ 545]	
	---	rcdAd dr									Record address of accounting period	
	---	stream									Information element/object data [▶ 485]	
	---	length	= 7									
	---	data			7	6	5	4	3	2	1	0
	---	data[0]	=									
	---	data[1]	=									
	---	data[2. .6]	=	CP40Time2a [▶ 533]								
	---	data[7. .IEC87 0_MA X_AS DU_D ATA_ BYTE]	=									
												Reserved

**4.2.2.4.17 C\_CI\_NN\_2**

Read periodically reset operational integrated totals of the oldest integration period.

- obj															ASDU object [▶ 545]
---	+ head														Reserved
---	- ident														DATA UNIT IDENTIFIER [▶ 546]
	---	eType	= 0x74 (116)	= C_CI_NN_2											Type identification [▶ 503]
	---	bSQ		= FALSE											Sequence of information objects
	---	nObj		= 0											Number of objects
	---	bT													Test
	---	bPN													Positive/negative confirmation/activation
	---	eCOT													Cause of transmission [▶ 512]
	---	asduA ddr													Common address of asdu
	---	eClass													Fifo priority class [▶ 509]
---	- info														INFORMATION OBJECT [▶ 545]
	---	rcdAd dr													Record address of accounting period
	---	stream													Information element/object data [▶ 485]
		---	length	= 0											
		---	data			7	6	5	4	3	2	1	0		
		---	data[0. .IEC87 0_MA X_AS DU_D ATA_ BYTE] =												Reserved

**4.2.2.4.18 C\_CI\_NO\_2**

Read periodically reset operational integrated totals of the oldest integration period and of selected range of addresses.

- obj											ASDU object [▶ 545]	
---	+ head										Reserved	
---	- ident										DATA UNIT IDENTIFIER [▶ 546]	
	---	eType	= 0x75 (117)	= C_CI_NO_2							Type identification [▶ 503]	
	---	bSQ		= FALSE							Sequence of information objects	
	---	nObj		= 1							Number of objects	
	---	bT									Test	
	---	bPN									Positive/negative confirmation/activation	
	---	eCOT									Cause of transmission [▶ 512]	
	---	asduA ddr									Common address of asdu	
	---	eClass									Fifo priority class [▶ 509]	
---	- info										INFORMATION OBJECT [▶ 545]	
	---	rcdAd dr									Record address of accounting period	
	---	stream									Information element/object data [▶ 485]	
	---	length	= 2									
	---	data			7	6	5	4	3	2	1	0
		---	data[0] =									Integrated total address (from)
		---	data[1] =									Integrated total address (to)
		---	data[2. .IEC87 0_MA X_AS DU_D ATA_ BYTE] =									Reserved

**4.2.2.4.19 C\_CI\_NP\_2**

Read periodically reset operational integrated totals of a specific past integration period.



- obj											ASDU object [▶ 545]	
---	+ head										Reserved	
---	- ident										DATA UNIT IDENTIFIER [▶ 546]	
	---	eType	= 0x76 (118)	= C_CI_NP_2							Type identification [▶ 503]	
	---	bSQ		= FALSE							Sequence of information objects	
	---	nObj		= 1							Number of objects	
	---	bT									Test	
	---	bPN									Positive/negative confirmation/activation	
	---	eCOT									Cause of transmission [▶ 512]	
	---	asduA addr									Common address of asdu	
	---	eClass									Fifo priority class [▶ 509]	
---	- info										INFORMATION OBJECT [▶ 545]	
	---	rcdAd dr									Record address of accounting period	
	---	stream									Information element/object data [▶ 485]	
	---	length	= 5									
	---	data			7	6	5	4	3	2	1	0
		---	data[0. .4] =		CP40Time2a [▶ 533]					Five octets binary time tag		
		---	data[5. .IEC87 0_MA X_AS DU_D ATA_ BYTE] =							Reserved		

**4.2.2.4.20 C\_CI\_NQ\_2**

Read periodically reset operational integrated totals of a specific past integration period and of selected range of addresses.

- obj												ASDU object [▶ 545]
---	+ head											Reserved
---	- ident											DATA UNIT IDENTIFIER [▶ 546]
	---	eType	= 0x77 (119)	= C_CI_NQ_2								Type identification [▶ 503]
	---	bSQ		= FALSE								Sequence of information objects
	---	nObj		= 1								Number of objects
	---	bT										Test
	---	bPN										Positive/negative confirmation/activation
	---	eCOT										Cause of transmission [▶ 512]
	---	asduA addr										Common address of asdu
	---	eClass										Fifo priority class [▶ 509]
---	- info											INFORMATION OBJECT [▶ 545]
	---	rcdAd dr										Record address of accounting period
	---	stream										Information element/object data [▶ 485]
	---	length	= 7									
	---	data			7	6	5	4	3	2	1	0
	---	data[0]	=									Integrated total address (from)
	---	data[1]	=									Integrated total address (to)
	---	data[2. .6]	=	CP40Time2a [▶ 533]								Five octets binary time tag
	---	data[7. .IEC87 0_MA X_AS DU_D ATA_ BYTE]	=									Reserved

**4.2.2.4.21 C\_CI\_NR\_2**

Read accounting integrated totals of selected time and of selected range of addresses.

- obj																				ASDU object [▶ 545]
---	+ head																			Reserved
---	- ident																			DATA UNIT IDENTIFIER [▶ 546]
	---	eType	= 0x78 (120)	= C_CI_NR_2																Type identification [▶ 503]
	---	bSQ		= FALSE																Sequence of information objects
	---	nObj		= 1																Number of objects
	---	bT																		Test
	---	bPN																		Positive/negative confirmation/activation
	---	eCOT																		Cause of transmission [▶ 512]
	---	asduA addr																		Common address of asdu
	---	eClass																		Fifo priority class [▶ 509]
---	- info																			INFORMATION OBJECT [▶ 545]
	---	rcdAd dr																		Record address of accounting period
	---	stream																		Information element/object data [▶ 485]
	---	length	= 12																	
	---	data				7	6	5	4	3	2	1	0							
	---	data[0] =																		Integrated total address (from)
	---	data[1] =																		Integrated total address (to)
	---	data[2. .6] =																		Five octets binary time tag (from)
	---	data[7. .11] =																		Five octets binary time tag (to)
	---	data[1 2..IEC 870_M AX_A SDU_ DATA_ _BYT _E] =																		Reserved

**4.2.2.4.22 C\_CI\_NS\_2**

Read periodically reset accounting integrated totals of selected time and of selected range of addresses.

- obj																			ASDU object [► 545]
---	+ head																		Reserved
---	- ident																		DATA UNIT IDENTIFIER [► 546]
	---	eType	= 0x79 (121)	= C_CI_NS_2															Type identification [► 503]
	---	bSQ		= FALSE															Sequence of information objects
	---	nObj		= 1															Number of objects
	---	bT																	Test
	---	bPN																	Positive/negative confirmation/activation
	---	eCOT																	Cause of transmission [► 512]
	---	asduA addr																	Common address of asdu
	---	eClass																	Fifo priority class [► 509]
---	- info																		INFORMATION OBJECT [► 545]
	---	rcdAd dr																	Record address of accounting period
	---	stream																	Information element/object data [► 485]
	---	length	= 12																
	---	data				7	6	5	4	3	2	1	0						
	---	data[0] =																	Integrated total address (from)
	---	data[1] =																	Integrated total address (to)
	---	data[2. .6] =																	Five octets binary time tag (from)
	---	data[7. .11] =																	Five octets binary time tag (to)
	---	data[1 2..IEC 870_M AX_A SDU_ DATA_ _BYT _E] =																	Reserved

**4.2.2.4.23 C\_CI\_NT\_2**

Read operational integrated totals of selected time and of selected range of addresses.

- obj																					ASDU object [► 545]
---	+ head																				Reserved
---	- ident																				DATA UNIT IDENTIFIER [► 546]
	---	eType	= 0x7A (122)	= C_CI_NT_2																	Type identification [► 503]
	---	bSQ		= FALSE																	Sequence of information objects
	---	nObj		= 1																	Number of objects
	---	bT																			Test
	---	bPN																			Positive/negative confirmation/activation
	---	eCOT																			Cause of transmission [► 512]
	---	asduAddr																			Common address of asdu
	---	eClass																			Fifo priority class [► 509]
---	- info																				INFORMATION OBJECT [► 545]
	---	rcdAddr																			Record address of accounting period
	---	stream																			Information element/object data [► 485]
	---	length	= 12																		
	---	data				7	6	5	4	3	2	1	0								
	---	data[0]	=																		Integrated total address (from)
	---	data[1]	=																		Integrated total address (to)
	---	data[2..6]	=																		Five octets binary time tag (from)
	---	data[7..11]	=																		Five octets binary time tag (to)
	---	data[12..IEC870_MAX_ASDU_DATA_BYTE]	=																		Reserved

**4.2.2.4.24 C\_CI\_NU\_2**

Read periodically reset operational integrated totals of selected time and of selected range of addresses.

- obj																				ASDU object [▶ 545]
---	+ head																			Reserved
---	- ident																			DATA UNIT IDENTIFIER [▶ 546]
	---	eType	= 0x7B (123)	= C_CI_NU_2																Type identification [▶ 503]
	---	bSQ		= FALSE																Sequence of information objects
	---	nObj		= 1																Number of objects
	---	bT																		Test
	---	bPN																		Positive/negative confirmation/activation
	---	eCOT																		Cause of transmission [▶ 512]
	---	asduA ddr																		Common address of asdu
	---	eClass																		Fifo priority class [▶ 509]
---	- info																			INFORMATION OBJECT [▶ 545]
	---	rcdAd dr																		Record address of accounting period
	---	stream																		Information element/object data [▶ 485]
	---	length	= 12																	
	---	data				7	6	5	4	3	2	1	0							
	---	data[0]	=																	Integrated total address (from)
	---	data[1]	=																	Integrated total address (to)
	---	data[2..6]	=																	Five octets binary time tag (from)
	---	data[7..11]	=																	Five octets binary time tag (to)
	---	data[12..15]	=																	Reserved

### 4.2.2.5 Informationselemente

#### 4.2.2.5.1 SPA

Single-point address: <0..255>

- <0..127> compatible range
- <128..255> private range

#### 4.2.2.5.2 SPQ

Single-point qualifier: <0..127>

- <0> not specified
- <1..127> vendor specific

### 4.2.2.5.3 Record address

Die Listenadressen sind wie folgt festgelegt: <0..255>

- <0> Default, wenn kein anderer Wert festgelegt ist
- <1> Listenadresse für Zählerstände vom Beginn der Abrechnungsperiode
- <2..10> Reserviert für weitere kompatible Festlegungen
- <11> Listenadresse für Zählerstände der Messperiode 1
- <12> Listenadresse für Zählerstände der Messperiode 2
- <13> Listenadresse für Zählerstände der Messperiode 3
- <14..20> Reserviert für weitere kompatible Festlegungen
- <21> Listenadresse für Zählerstände (Tageswerte) der Messperiode 1
- <22> Listenadresse für Zählerstände (Tageswerte) der Messperiode 2
- <23> Listenadresse für Zählerstände (Tageswerte) der Messperiode 3
- <24..30> Reserviert für weitere kompatible Festlegungen
- <31> Listenadresse für Zählerstände (Monatswerte) der Messperiode 1
- <32> Listenadresse für Zählerstände (Monatswerte) der Messperiode 2
- <33> Listenadresse für Zählerstände (Monatswerte) der Messperiode 3
- <34..40> Reserviert für weitere kompatible Festlegungen
- <41> Listenadresse für Zählerstände (Jahreswerte) der Messperiode 1
- <42> Listenadresse für Zählerstände (Jahreswerte) der Messperiode 2
- <43> Listenadresse für Zählerstände (Jahreswerte) der Messperiode 3
- <44..49> Reserviert für weitere kompatible Festlegungen
- <50> Älteste Einzelmeldung
- <51> Vollständige Liste mit Einzelmeldungen
- <52> Teilliste 1 mit Einzelmeldungen
- <53> Teilliste 2 mit Einzelmeldungen
- <54> Teilliste 3 mit Einzelmeldungen
- <55> Teilliste 4 mit Einzelmeldungen
- <56..127> Reserviert für weitere kompatible Festlegungen
- <128..255> Für besondere Anwendungen (privater Bereich)

Die Größe einer Teilliste ist ein Systemparameter.

## 4.3 IEC 60870-5-103

TF6500 implementiert eine Schnittstelle über die einzelne IEC 60870-5-103 ASDUs (Dienstdateneinheiten der Anwendungsschicht) über die seriellen Schnittstellen des PCs und die Beckhoff KL6xxx-/EL6xxx-Klemmen verschickt und empfangen werden können.

Die Endapplikation (Implementierung einer Zentralstation) wird auf der Software-Schnittstelle der SPS-Bibliothek aufgesetzt. Anders als bei den 101/104-TwinCAT Supplement Produkten steht dem SPS-Programmierer bei der IEC60870-5-103-Protokollimplementierung nur die sogenannte "Low level"-Schnittstelle zur Verfügung. Der Grund: Die meisten Unterstationen (Schutzgeräte) besitzen viele Herstellerspezifische oder Gerätespezifische Parameterdaten oder Informationen. Die "Low level"-Schnittstelle ermöglicht einen uneingeschränkten Zugriff auf all diese Daten. Im Folgenden werden die Eigenschaften dieser Schnittstelle kurz beschrieben.

**"Low level"-Schnittstelle:**

Diese Schnittstelle setzt bei dem Protokollstack tiefer an und ermöglicht das Versenden und/oder Empfangen einzelner Frames (ASDU's). Innerhalb der Protokollstruktur liegt diese Schnittstelle oberhalb der Verbindungsschicht und implementiert bereits die nötigen Prozeduren und Übertragungstelegrammformate. Anwendungsfunktionen wie z.B. Generalabfrage sind in der Schnittstelle nicht implementiert, der Anwender kann aber mit Hilfe der Schnittstelle diese Anwendungsfunktionen selber implementieren.

**Pro:** Sehr flexibel; Alle Eigenschaften im ASDU-Frame können manipuliert werden (z.B. Funktionscodes, Informationsnummern usw.); Weil nur die benötigten Dienste (abhängig vom Gerät) implementiert werden kann eine hohe Performance erreicht werden; Hohe Performance bei vielen Datenpunkten oder einem hohem Datenaufkommen;

**Contra:** Grösserer Programmieraufwand; Alle Dienste wie Generalabfrage, Zählerabfrage, Uhrzeitsynchronisation, Befehlsausführung usw. müssen von dem SPS-Programmierer selbst implementiert (ausprogrammiert) werden; Der SPS-Programmierer muss mit der Protokollnorm vertraut sein.

Protokollstruktur des Endsystems:

**Protokollstruktur des Endsystems:**

Auswahl von Anwendungsfunktionen aus IEC 60870-5-5	Anwenderprozess
Auswahl von ASDU (Dienstdateneinheiten der Anwendungsschicht) aus IEC 60870-5-3, IEC 60870-5-4 und IEC 60870-5-103	Anwendungsschicht (7)
N/A	Darstellungsschicht (6)
	Sitzungsschicht (5)
	Transportschicht (4)
	Vermittlungsschicht (3)
unbalanced IEC 60870-5-2 IEC 60870-5-1 (FT 1.2)	Verbindungsschicht (2)
EIA RS485, RS232 (V.24), Fibre Optics	Physikalische Schicht (1)



Die Schichten 3 bis 6 werden nicht benutzt.

**4.3.1 IEC 60870-5-103 telegram structure**

- FT 1.2 frame format is used;
- Frames with fixed and variable block length and single control characters are used;
- Hamming distance = 4;

**Frame with variable length**

This frame type is used to transmit user data between controlling and controlled station



byte\bit	7	6	5	4	3	2	1	0					
0	Start byte 1 (0x68)								Header		LPCI		LPDU
1	Block length												
2	Block length (copy)												
3	Start byte 2 (0x68)												
4	RES	PRM	FCB / ACD	FCV / DFC	Function code				Control field				
5	Link address												
6	Type identification ( <u>monitoring-direction</u> [▶ 505], <u>control-direction</u> [▶ 505])								DATA UNIT IDENTIFIER		ASDU		
7	SQ	Number of object											
8	Cause of transmission ( <u>monitoring-direction</u> [▶ 513], <u>control-direction</u> [▶ 514])												
9	ASDU address												
10	Function type [▶ 325]												
11	Information number [▶ 325]								Info-object				
	Information elements												
n-1	Checksum								Tail		LPCI		
n	Stop byte (0x16)												

**Frame with fixed length**

This frame type is used for link layer services. Sometimes used as an confirmation.

byte\bit	7	6	5	4	3	2	1	0	
0	Start byte 1 (0x10)								
1	RES	PRM	FCB / ACD	FCV / DFC	Function code				Control field
2	Link address								
3	Checksum								
4	Stop byte (0x16)								

**Single character data**

Single characters are used to confirm link layer and user data transmissions.

- 0xE5 (positive acknowledge);

## 4.3.2 ASDU-Objektbeschreibung

### 4.3.2.1 Standard IEC 60870-5-103 Datentypen

In Überwachungsrichtung	In Steuerungsrichtung	Dez	Hex	Beschreibung
M_TYPEUNDEF_3	C_TYPEUNDEF_3	0	0x00	Wird nicht verwendet
M_TTM_TA_3 [▶ 298]	-	1	0x01	Meldung mit Zeitstempel
M_TMR_TA_3 [▶ 299]	-	2	0x02	Meldung mit Zeitstempel und Relativzeit
M_MEI_NA_3 [▶ 300]	-	3	0x03	Messwerte I
M_TME_TA_3 [▶ 302]	-	4	0x04	Echtzeitmesswerte mit Relativzeit
M_IRC_NA_3 [▶ 311]	-	5	0x05	Identifikationsmeldung
M_SYN_TA_3 [▶ 313]	C_SYN_TA_3 [▶ 318]	6	0x06	Zeitsynchronisierung
-	C_IGI_NA_3 [▶ 318]	7	0x07	Generalabfrage-Initialisierung
M_TGI_NA_3 [▶ 312]	-	8	0x08	Generalabfrage-Ende
M_MEII_NA_3 [▶ 302]	-	9	0x09	Messwerte II
M_GD_XA_3 [▶ 316]	C_GD_NA_3 [▶ 319]	10	0x0A	Generische Daten
M_GI_XA_3 [▶ 314]	-	11	0x0B	Generische Identifikation
-	-	12..19	0x0C..0x13	Reserviert (Standardbereich)
-	C_GRC_NA_3 [▶ 321]	20	0x14	Allgemeiner Befehl
-	C_GC_NA_3 [▶ 321]	21	0x15	Generischer Befehl
-	-	22	0x16	Reserviert (Standardbereich)
M_LRD_TA_3 [▶ 304]	-	23	0x17	Störfallübersicht
-	C_ODT_NA_3 [▶ 322]	24	0x18	Auftrag zur Übertragung der Stördaten
-	C_ADT_NA_3 [▶ 323]	25	0x19	Quittung für Stördatenübertragung
M_RTD_TA_3 [▶ 305]	-	26	0x1A	Bereit zur Übertragung von Stördaten
M_RTC_NA_3 [▶ 306]	-	27	0x1B	Bereit zur Übertragung eines Kanals
M_RTT_NA_3 [▶ 307]	-	28	0x1C	Bereit zur Übertragung von Marken
M_TOT_NA_3 [▶ 308]	-	29	0x1D	Übertragung von Marken
M_TOV_NA_3 [▶ 309]	-	30	0x1E	Übertragung von Störwerten
M_EOT_NA_3 [▶ 310]	-	31	0x1F	Ende der Übertragung
-	-	32..255	0x1F..0xFF	Reserviert (benutzerdefinierter Bereich)

### 4.3.2.2 Prozessinformation in Überwachungsrichtung

#### 4.3.2.2.1 M\_TTM\_TA\_3

Time-tagged message.

- obj					ASDU object [▶ 547]								
---	+ head				Reserved								
---	- ident				DATA UNIT IDENTIFIER [▶ 549]								
	---	eType	= 0x1 (1)	= M_TTM_TA_3	Type identification [▶ 505]								
	---	bSQ		= TRUE	Sequence of information objects								
	---	nObj		= 1	Number of objects								
	---	eCOT			Cause of transmission [▶ 513]								
	---	asduAddr			Common address of asdu								
	---	eClasses			Fifo priority class [▶ 509]								
---	- info				INFORMATION OBJECT [▶ 548]								
	---	fc			Function code/type [▶ 325]								
	---	n			Information number [▶ 325]								
	---	stream			Information element/object data [▶ 485]								
		---	length	= 6									
		---	data		7	6	5	4	3	2	1	0	
		---	data[0]	=	0	0	0	0	0	0	DPI	1	DPI = Double-point information [▶ 527]
		---	data[1..4]	=	CP32Time2a [▶ 533]				Four octets binary time tag				
		---	data[5]	=	SIN [▶ 328]				Supplementary information				
		---	data[6..IEC870_MAX_ASDU_DATA_BYTE]	=	Reserved								

**4.3.2.2.2 M\_TMR\_TA\_3**

Time-tagged message with relative time.

- obj					ASDU object [▶ 547]									
---	+ head				Reserved									
---	- ident				DATA UNIT IDENTIFIER [▶ 549]									
	---	eType	= 0x2 (2)	= M_TMR_TA_3	Type identification [▶ 505]									
	---	bSQ		= TRUE	Sequence of information objects									
	---	nObj		= 1	Number of objects									
	---	eCOT			Cause of transmission [▶ 513]									
	---	asduAddr			Common address of asdu									
	---	eClasses			Fifo priority class [▶ 509]									
---	- info				INFORMATION OBJECT [▶ 548]									
	---	fc			Function code/type [▶ 325]									
	---	n			Information number [▶ 325]									
	---	stream			Information element/object data [▶ 485]									
		---	length	= 10										
		---	data		7	6	5	4	3	2	1	0		
		---	data[0]	=	0	0	0	0	0	0	DPI	0	DPI = Double-point information [▶ 527]	
											1			
		---	data[1..2]	=	RET [▶ 328]		Relative time							
		---	data[3..4]	=	FAN [▶ 327]		Fault number							
		---	data[5..8]	=	CP32Time2a [▶ 533]		Four octets binary time tag							
		---	data[9]	=	SIN [▶ 328]		Supplementary information							
		---	data[10..IEC 870_MAX_ASDU_DATA_BYTE]	=	Reserved									

### 4.3.2.2.3 M\_MEI\_NA\_3

Measurands I.

- obj					ASDU object [▶ 547]								
---	+ head				Reserved								
---	- ident				DATA UNIT IDENTIFIER [▶ 549]								
	---	eType	= 0x3 (3)	= M_MEI_NA_3	Type identification [▶ 505]								
	---	bSQ		= FALSE	Sequence of information objects								
	---	nObj		= i	Number of objects								
	---	eCOT			Cause of transmission [▶ 513]								
	---	asduAddr			Common address of asdu								
	---	eClasses			Fifo priority class [▶ 509]								
---	- info				INFORMATION OBJECT [▶ 548]								
	---	fc			Function code/type [▶ 325]								
	---	n			Information number [▶ 325]								
	---	stream			Information element/object data [▶ 485]								
	---	length	= 8 * i										
	---	data			7	6	5	4	3	2	1	0	
	---	data[0]	=							RES [▶ 330] 1	ER [▶ 330] 1	OV [▶ 329] 1	MEA = Measurand with quality descriptor
	---	data[1]	=	MVAL [▶ 327]	Current L2								
	---	data[2]	=							RES [▶ 330] 1	ER [▶ 330] 1	O V [▶ 329] 1	MEA = Measurand with quality descriptor
	---	data[3]	=	MVAL [▶ 327]	Voltage L1-L2								
	---	data[4]	=							RES [▶ 330] 1	ER [▶ 330] 1	O V [▶ 329] 1	MEA = Measurand with quality descriptor
	---	data[5]	=	MVAL [▶ 327]	Active power P								
	---	data[6]	=							RES [▶ 330] 1	ER [▶ 330] 1	O V [▶ 329] 1	MEA = Measurand with quality descriptor
	---	data[7]	=	MVAL [▶ 327]	Reactive power Q								

			---	data[8 ..IEC8 70_M AX_A SDU_ DATA _BYT E] =	Reserved
--	--	--	-----	--	----------

#### 4.3.2.2.4 M\_TME\_TA\_3

Time-tagged measurands with relative time

- obj					ASDU object [▶ 547]
---	+ head				Reserved
---	- ident				DATA UNIT IDENTIFIER [▶ 549]
	---	eType = 0x4 (4)	= M_TME_TA_3		Type identification [▶ 505]
	---	bSQ	= TRUE		Sequence of information objects
	---	nObj	= 1		Number of objects
	---	eCOT			Cause of transmission [▶ 513]
	---	asduAddress			Common address of asdu
	---	eClasses			Fifo priority class [▶ 509]
---	- info				INFORMATION OBJECT [▶ 548]
	---	fc			Function code/type [▶ 325]
	---	n			Information number [▶ 325]
	---	stream			Information element/object data [▶ 485]
		---	length = 12		
		---	data	7   6   5   4   3   2   1   0	
		---	data[0..3] =	R32 [▶ 241]	Short circuit location
		---	data[4..5] =	RET [▶ 328]	Relative time
		---	data[6..7] =	FAN [▶ 327]	Fault number
		---	data[8..11] =	CP32Time2a [▶ 533]	Four octets binary time tag
		---	data[12..IEC870_MAX_ASDU_DATA_BYTE] =		Reserved

#### 4.3.2.2.5 M\_MEII\_NA\_3

Measurands II.

- obj					ASDU object [▶ 547]									
---	+ head				Reserved									
---	- ident				DATA UNIT IDENTIFIER [▶ 549]									
	---	eType	= 0x9 (9)	= M_MEII_NA_3	Type identification [▶ 505]									
	---	bSQ		= FALSE	Sequence of information objects									
	---	nObj		= i	Number of objects									
	---	eCOT			Cause of transmission [▶ 513]									
	---	asduA addr			Common address of asdu									
	---	eClas s			Fifo priority class [▶ 509]									
---	- info				INFORMATION OBJECT [▶ 548]									
	---	fc			Function code/type [▶ 325]									
	---	n			Information number [▶ 325]									
	---	strea m			Information element/object data [▶ 485]									
	---	length	= 18 * i											
	---	data			7	6	5	4	3	2	1	0		
	---	data[0]	=							RES [▶ 330] 1	ER [▶ 330] 1	OV [▶ 329] 1	MEA = Measuran d with quality descriptor	
	---	data[1]	=	MVAL [▶ 327]	Current L1									
	---	data[2]	=							RES [▶ 330] 1	ER [▶ 330] 1	OV [▶ 329] 1	MEA = Measuran d with quality descriptor	
	---	data[3]	=	MVAL [▶ 327]	Current L2									
	---	data[4]	=							RES [▶ 330] 1	ER [▶ 330] 1	OV [▶ 329] 1	MEA = Measuran d with quality descriptor	
	---	data[5]	=	MVAL [▶ 327]	Current L3									
	---	data[6]	=							RES [▶ 330] 1	ER [▶ 330] 1	OV [▶ 329] 1	MEA = Measuran d with quality descriptor	
	---	data[7]	=	MVAL [▶ 327]	Voltage L1-E									
	---	data[8]	=							RES [▶ 330] 1	ER [▶ 330] 1	OV [▶ 329] 1	MEA = Measuran d with quality descriptor	
	---	data[9]	=	MVAL [▶ 327]	Voltage L2-E									

			---	data[10] =		<u>RES</u> [▶ 330] 1	<u>ER</u> [▶ 330] 1	<u>OV</u> [▶ 329] 1	MEA = Measurand with quality descriptor
			---	data[11] =	<u>MVAL</u> [▶ 327]	Voltage L3-E			
			---	data[12] =		<u>RES</u> [▶ 330] 1	<u>ER</u> [▶ 330] 1	<u>OV</u> [▶ 329] 1	MEA = Measurand with quality descriptor
			---	data[13] =	<u>MVAL</u> [▶ 327]	Active power P			
			---	data[14] =		<u>RES</u> [▶ 330] 1	<u>ER</u> [▶ 330] 1	<u>OV</u> [▶ 329] 1	MEA = Measurand with quality descriptor
			---	data[15] =	<u>MVAL</u> [▶ 327]	Reactive power Q			
			---	data[16] =		<u>RES</u> [▶ 330] 1	<u>ER</u> [▶ 330] 1	<u>OV</u> [▶ 329] 1	MEA = Measurand with quality descriptor
			---	data[17] =	<u>MVAL</u> [▶ 327]	Frequency f			
			---	data[18..IEC870_MAX_ASDU_DATA_BYTE] =		Reserved			

**4.3.2.2.6 M\_LRD\_TA\_3**

List of recorded disturbances.



- obj					ASDU object [▶ 547]										
---	+	head			Reserved										
---	-	ident			DATA UNIT IDENTIFIER [▶ 549]										
	---	eType	=	= M_LRD_TA_3	Type identification [▶ 505]										
	---	bSQ		= FALSE	Sequence of information objects										
	---	nObj		= i	Number of objects										
	---	eCOT			Cause of transmission [▶ 513]										
	---	asdu Addr			Common address of asdu										
	---	eClasses			Fifo priority class [▶ 509]										
---	-	info			INFORMATION OBJECT [▶ 548]										
	---	fc			Function code/type [▶ 325]										
	---	n		= 0	not used										
	---	stream			Information element/object data [▶ 485]										
	---	length	=	i * 10											
	---	data			7	6	5	4	3	2	1	0			
		---	data[0..1]	=	FAN [▶ 327]					Fault number					Dataset 1
		---	data[2]	=	RES			OEV	TEST	TM	TP	SOF = Status of fault			
		---	data[3..9]	=	CP56Time2a [▶ 531]					Seven octets binary time tag					
		---	data[10..IEC870_MAX_ASDU_DATA_BYTE]	=											Dataset 2..i

**4.3.2.2.7 M\_RTD\_TA\_3**

Ready for transmission of disturbance data.

- obj					ASDU object [▶ 547]										
---	+ head				Reserved										
---	- ident				DATA UNIT IDENTIFIER [▶ 549]										
	---	eType	0x1A (26)	= M_RTD_TA_3	Type identification [▶ 505]										
	---	bSQ		= TRUE	Sequence of information objects										
	---	nObj		= 1	Number of objects										
	---	eCOT			Cause of transmission [▶ 513]										
	---	asduAddr			Common address of asdu										
	---	eClasses			Fifo priority class [▶ 509]										
---	- info				INFORMATION OBJECT [▶ 548]										
	---	fc			Function code/type [▶ 325]										
	---	n		= 0	not used										
	---	stream			Information element/object data [▶ 485]										
		---	length	= 15											
		---	data		7	6	5	4	3	2	1	0			
		---	data[0]	= 0	not used										
		---	data[1]	=	TOV [▶ 329]							Type of disturbance values			
		---	data[2..3]	=	FAN [▶ 327]							Fault number			
		---	data[4..5]	=	NOF [▶ 328]							Number of grid faults			
		---	data[6]	=	NOC [▶ 328]							Number of channels			
		---	data[7..8]	=	NOE [▶ 328]							Number of information elements of a channel			
		---	data[9..10]	=	INT [▶ 327]							Interval			
		---	data[11..14]	=	CP32Time2a [▶ 533]							Four octets binary time tag			
		---	data[15..IEC 870 MAX_ASDU_DATA_BYTE]	=	Reserved										

#### 4.3.2.2.8 M\_RTC\_NA\_3

Ready for transmission of channel.

- obj					ASDU object [▶ 547]								
---	+ head				Reserved								
---	- ident				DATA UNIT IDENTIFIER [▶ 549]								
	---	eType = 0x1B (27)	= M_RTC_NA_3		Type identification [▶ 505]								
	---	bSQ	= TRUE		Sequence of information objects								
	---	nObj	= 1		Number of objects								
	---	eCOT			Cause of transmission [▶ 513]								
	---	asduAddr			Common address of asdu								
	---	eClasses			Fifo priority class [▶ 509]								
---	- info				INFORMATION OBJECT [▶ 548]								
	---	fc			Function code/type [▶ 325]								
	---	n	= 0		not used								
	---	stream			Information element/object data [▶ 485]								
		---	length = 17										
		---	data		7	6	5	4	3	2	1	0	
		---	data[0] =	0	not used								
		---	data[1] =	TOV [▶ 329]	Type of disturbance value								
		---	data[2..3] =	FAN [▶ 327]	Fault number								
		---	data[4] =	ACC [▶ 325]	Actual channel								
		---	data[5..8] =	R32 [▶ 241]	RPV = Rated primary value								
		---	data[9..12] =	R32 [▶ 241]	RSV = Rated secondary value								
		---	data[13..16] =	R32 [▶ 241]	RFA = Reference factor								
		---	data[17..IEC 870_MAX_ASDU_DATA_BYTE] =		Reserved								

**4.3.2.2.9 M\_RTT\_NA\_3**

Ready for transmission of tags.

- obj					ASDU object [▶ 547]								
---	+ head				Reserved								
---	- ident				DATA UNIT IDENTIFIER [▶ 549]								
	---	eType	0x1C (28)	= M_RTT_NA_3	Type identification [▶ 505]								
	---	bSQ		= TRUE	Sequence of information objects								
	---	nObj		= 1	Number of objects								
	---	eCOT			Cause of transmission [▶ 513]								
	---	asduAddr			Common address of asdu								
	---	eClasses			Fifo priority class [▶ 509]								
---	- info				INFORMATION OBJECT [▶ 548]								
	---	fc			Function code/type [▶ 325]								
	---	n		= 0	not used								
	---	stream			Information element/object data [▶ 485]								
		---	length	= 4									
		---	data		7	6	5	4	3	2	1	0	
		---	data[0]	= 0	not used								
		---	data[1]	= 0	not used								
		---	data[2..3]	= FAN [▶ 327]	Fault number								
		---	data[4..IEC870_MAX_AX_ASDU_DATA_BYTE]	=	Reserved								

**4.3.2.2.10 M\_TOT\_NA\_3**

Transmission of tags.

- obj					ASDU object [▶ 547]							
---	+ head				Reserved							
---	- ident				DATA UNIT IDENTIFIER [▶ 549]							
	---	eType	= 0x1D (29)	= M_TOT_NA_3	Type identification [▶ 505]							
	---	bSQ		= TRUE	Sequence of information objects							
	---	nObj		= 1	Number of objects							
	---	eCOT			Cause of transmission [▶ 513]							
	---	asdu Addr			Common address of asdu							
	---	eClasses			Fifo priority class [▶ 509]							
---	- info				INFORMATION OBJECT [▶ 548]							
	---	fc			Function code/type [▶ 325]							
	---	n		= 0	not used							
	---	stream			Information element/object data [▶ 485]							
	---	length	= 5 + (i * 3)									
	---	data			7	6	5	4	3	2	1	0
		---	data[0..1] =	FAN [▶ 327]	Fault number							
		---	data[2] =	NOT [▶ 328]	Number of tags							
		---	data[3..4] =	TAP [▶ 329]	Tag position							
		---	data[5] =		Function code/type							Tag 1
		---	data[6] =		Information number							
		---	data[7] =	0					DPI [▶ 527]	Double point information		
		---	data[8..IE C870_MAX_ASDU_DATA_BYTE] =									Tag 2..i

**4.3.2.2.11 M\_TOV\_NA\_3**

Transmission of disturbance values.

- obj					ASDU object [▶ 547]								
---	+ head				Reserved								
---	- ident				DATA UNIT IDENTIFIER [▶ 549]								
	---	eType = 0x1E (30)	= M_TOV_NA_3		Type identification [▶ 505]								
	---	bSQ	= TRUE		Sequence of information objects								
	---	nObj	= 1		Number of objects								
	---	eCOT			Cause of transmission [▶ 513]								
	---	asduAddr			Common address of asdu								
	---	eClasses			Fifo priority class [▶ 509]								
---	- info				INFORMATION OBJECT [▶ 548]								
	---	fc			Function code/type [▶ 325]								
	---	n	= 0		not used								
	---	stream			Information element/object data [▶ 485]								
		---	length = 8 + (i * 2)										
		---	data		7	6	5	4	3	2	1	0	
		---	data[0] =	0	not used								
		---	data[1] =	TOV [▶ 329]	Type of disturbance values								
		---	data[2..3] =	FAN [▶ 327]	Fault number								
		---	data[4] =	ACC [▶ 325]	Actual channel								
		---	data[5] =	NDV [▶ 328]	Number of relevant disturbance values per ASDU								
		---	data[6..7] =	NFE [▶ 328]	Number of the ASDU's first information element								
		---	data[8..9] =	SDV [▶ 328]	Single disturbance value 1								
		---	data[10..11] =	SDV [▶ 328]	Single disturbance value 2								
		---	data[12..IEC 870_MAX_ASDU_DATA_BY TE] =	SDV [▶ 328]	Single disturbance value 3..i								

**4.3.2.2.12 M\_EOT\_NA\_3**

End of transmission.

- obj					ASDU object [▶ 547]								
---	+ head				Reserved								
---	- ident				DATA UNIT IDENTIFIER [▶ 549]								
	---	eType	= 0x1F (31)	= M_EOT_NA_3	Type identification [▶ 505]								
	---	bSQ		= TRUE	Sequence of information objects								
	---	nObj		= 1	Number of objects								
	---	eCOT			Cause of transmission [▶ 513]								
	---	asduAddr			Common address of asdu								
	---	eClasses			Fifo priority class [▶ 509]								
---	- info				INFORMATION OBJECT [▶ 548]								
	---	fc			Function code/type [▶ 325]								
	---	n		= 0	not used								
	---	stream			Information element/object data [▶ 485]								
		---	length	= 5									
		---	data		7	6	5	4	3	2	1	0	
		---	data[0]	=	TOO [▶ 329]		Type of order						
		---	data[1]	=	TOV [▶ 329]		Type of disturbance values						
		---	data[2..3]	=	FAN [▶ 327]		Fault number						
		---	data[4]	=	ACC [▶ 325]		Actual channel						
		---	data[5..IEC870_MAX_ASDU_DATA_BYTE]	=	Reserved								

### 4.3.2.3 Systeminformation in Überwachungsrichtung

#### 4.3.2.3.1 M\_IRC\_NA\_3

Identification.

- obj					ASDU object [▶ 547]									
---	+ head				Reserved									
---	- ident				DATA UNIT IDENTIFIER [▶ 549]									
	---	eType	= 0x5 (5)	= M_IRC_NA_3	Type identification [▶ 505]									
	---	bSQ		= TRUE	Sequence of information objects									
	---	nObj		= 1	Number of objects									
	---	eCOT			Cause of transmission [▶ 513]									
	---	asduA ddr			Common address of asdu									
	---	eClas s			Fifo priority class [▶ 509]									
---	- info				INFORMATION OBJECT [▶ 548]									
	---	fc			Function code/type [▶ 325]									
	---	n			Information number [▶ 325]									
	---	strea m			Information element/object data [▶ 485]									
		---	length	= 13										
		---	data		7	6	5	4	3	2	1	0		
		---	data[0 ]	=	COL [▶ 326]						Compatibility level			
		---	data[1 ..8]	=	ASC [▶ 326]						ASCII characters			
		---	data[9 ..12]	= 0x20 if not used	free						Internal vendor identification			
		---	data[1 3..IEC 870_ MAX_ ASDU _DAT A_BY TE]	=							Reserved			

### 4.3.2.3.2 M\_TGI\_NA\_3

Termination of general interrogation.



- obj					ASDU object [▶ 547]								
---	+ head				Reserved								
---	- ident				DATA UNIT IDENTIFIER [▶ 549]								
	---	eType	= 0x8 (8)	= M_TGI_NA_3	Type identification [▶ 505]								
	---	bSQ		= TRUE	Sequence of information objects								
	---	nObj		= 1	Number of objects								
	---	eCOT			Cause of transmission [▶ 513]								
	---	asduAddr			Common address of asdu								
	---	eClasses			Fifo priority class [▶ 509]								
---	- info				INFORMATION OBJECT [▶ 548]								
	---	fc		= GLB	Function code/type [▶ 325]								
	---	n			Information number [▶ 325]								
	---	stream			Information element/object data [▶ 485]								
		---	length	= 1									
		---	data		7	6	5	4	3	2	1	0	
		---	data[0]	=	SCN [▶ 328]			Scan number					
		---	data[1..IEC870_MAX_ASDU_DATA_BYTES]	=	Reserved								

**4.3.2.3.3 M\_SYN\_TA\_3**

Time synchronisation.

- obj					ASDU object [▶ 547]								
---	+ head				Reserved								
---	- ident				DATA UNIT IDENTIFIER [▶ 549]								
	---	eType	= 0x6 (6)	= M_SYN_TA_3	Type identification [▶ 505]								
	---	bSQ		= TRUE	Sequence of information objects								
	---	nObj		= 1	Number of objects								
	---	eCOT			Cause of transmission [▶ 513]								
	---	asduAddr			Common address of asdu								
	---	eClasses			Fifo priority class [▶ 509]								
---	- info				INFORMATION OBJECT [▶ 548]								
	---	fc		= GLB	Function code/type [▶ 325]								
	---	n			Information number [▶ 325]								
	---	stream			Information element/object data [▶ 485]								
		---	length	= 7									
		---	data		7	6	5	4	3	2	1	0	
		---	data[0..6]	= CP56Time2a [▶ 531]	Seven octets binary time format								
		---	data[7..IEC870_MAX_ASDU_DATA_BYTE]	=	Reserved								

**4.3.2.3.4 M\_GI\_XA\_3**

Generic identification.

- obj					ASDU object [▶ 547]										
---	+	head			Reserved										
---	-	ident			DATA UNIT IDENTIFIER [▶ 549]										
	---	eType	= 0xB	= M_GI_XA_3	Type identification [▶ 505]										
	---	bSQ		= TRUE	Sequence of information objects										
	---	nObj		= 1	Number of objects										
	---	eCOT			Cause of transmission [▶ 513]										
	---	asdu Addr			Common address of asdu										
	---	eClasses			Fifo priority class [▶ 509]										
---	-	info			INFORMATION OBJECT [▶ 548]										
	---	fc		GEN	Function code/type [▶ 325]										
	---	n			Information number [▶ 325]										
	---	stream			Information element/object data [▶ 485]										
	---	length	= variable												
	---	data			7	6	5	4	3	2	1	0			
	---	data[0]	=	RII [▶ 328]	Return information identifier										
	---	data[1]	=	GROUP [▶ 327]	GIN = Generic identification number										
	---	data[2]	=	ENTRY [▶ 327]											
	---	data[3]	=	CONT [▶ 326]	COU [▶ 326]	N [▶ 326]	NDE = Number of descriptive elements								
				1	1	1									
	---	data[4]	=	KOD [▶ 327]	Kind of description										Element 1
	---	data[5]	=	DATATYPE [▶ 326]	GDD = Generic data description										
	---	data[6]	=	DATASIZE [▶ 326]											
	---	data[7]	=	CONT [▶ 326]	NUMBER [▶ 329]										
				1											
	---	data[8..8+ (DATASIZE*NUMBER)]	=	GID [▶ 329]	Generic identification data										

			---	data[. .IEC8 70_M AX_A SDU_ DATA_ BYT E] =			Elem ent 2..i
--	--	--	-----	---	--	--	---------------------

**4.3.2.3.5 M\_GD\_XA\_3**

Generic data.

- obj					ASDU object [▶ 547]									
---	+	head			Reserved									
---	-	ident			DATA UNIT IDENTIFIER [▶ 549]									
	---	eType	= 0xA	= M_GD_XA_3	Type identification [▶ 505]									
	---	bSQ		= TRUE	Sequence of information objects									
	---	nObj		= 1	Number of objects									
	---	eCOT			Cause of transmission [▶ 513]									
	---	asdu Addr			Common address of asdu									
	---	eClasses			Fifo priority class [▶ 509]									
---	-	info			INFORMATION OBJECT [▶ 548]									
	---	fc		GEN	Function code/type [▶ 325]									
	---	n			Information number [▶ 325]									
	---	stream			Information element/object data [▶ 485]									
	---	length		= variable										
	---	data			7	6	5	4	3	2	1	0		
	---	data[0]	=		RII [▶ 328]			Return information identifier						
	---	data[1]	=		CONT	COU	N	NGD = Number of generic data sets						
					[▶ 326]	NT	O							
					1	[▶ 326]	[▶ 328]							
					1	1	1							
	---	data[2]	=		GROUP [▶ 327]			GIN = Generic identification number					Dataset 1	
	---	data[3]	=		ENTRY [▶ 327]									
	---	data[4]	=		KOD [▶ 327]			Kind of description						
	---	data[5]	=		DATATYPE [▶ 326]			GDD = Generic data description						
	---	data[6]	=		DATASIZE [▶ 326]									
	---	data[7]	=		CONT	NUMBER								
					[▶ 326]	[▶ 329]								
					1									
	---	data[8..8+ (DATASIZE*NUMBER)]	=		GID [▶ 329]			Generic identification data						

			---	data[.. IEC8 70_M AX_A SDU_ DATA_ BYT E] =															Dataset 2..i
--	--	--	-----	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--------------

### 4.3.2.4 Systeminformation in Steuerungsrichtung

#### 4.3.2.4.1 C\_SYN\_TA\_3

Time synchronisation.

- obj					ASDU object [▶ 547]																	
---	+	head			Reserved																	
---	-	ident			DATA UNIT IDENTIFIER [▶ 549]																	
	---	eType	= 0x6 (6)	= C_SYN_TA_3	Type identification [▶ 505]																	
	---	bSQ		= TRUE	Sequence of information objects																	
	---	nObj		= 1	Number of objects																	
	---	eCOT			Cause of transmission [▶ 514]																	
	---	asduAddress			Common address of asdu																	
	---	eClasses			Fifo priority class [▶ 509]																	
---	-	info			INFORMATION OBJECT [▶ 548]																	
	---	fc		:= GLB	Function code/type [▶ 325]																	
	---	n			Information number [▶ 325]																	
	---	stream			Information element/object data [▶ 485]																	
		---	length	= 7																		
		---	data			7	6	5	4	3	2	1	0									
		---	data[0..6]	=	CP56Time2a [▶ 531]											Sevent octets binary time tag						
		---	data[7..IEC870_MAX_ASDU_DATA_BYTE]	=	Reserved																	

#### 4.3.2.4.2 C\_IGI\_NA\_3

General interrogation.

- obj						ASDU object [▶ 547]								
---	+ head					Reserved								
---	- ident					DATA UNIT IDENTIFIER [▶ 549]								
	---	eType	= 0x7 (7)	= C_IGI_NA_3		Type identification [▶ 505]								
	---	bSQ		= TRUE		Sequence of information objects								
	---	nObj		= 1		Number of objects								
	---	eCOT				Cause of transmission [▶ 514]								
	---	asduAddr				Common address of asdu								
	---	eClasses				Fifo priority class [▶ 509]								
---	- info					INFORMATION OBJECT [▶ 548]								
	---	fc		:= GLB		Function code/type [▶ 325]								
	---	n				Information number [▶ 325]								
	---	stream				Information element/object data [▶ 485]								
		---	length	= 1										
		---	data			7	6	5	4	3	2	1	0	
		---	data[0]	=		SCN [▶ 328]							Scan number	
		---	data[1..IEC870_MAX_ASDU_DATA_BYTES]	=									Reserved	

**4.3.2.4.3 C\_GD\_NA\_3**

Generic data.

- obj																		ASDU object [▶ 547]
---	+	head																Reserved
---	-	ident																DATA UNIT IDENTIFIER [▶ 549]
	---	eType	=	C_GD_NA_3														Type identification [▶ 505]
	---	bSQ		= FALSE														Sequence of information objects
	---	nObj		= 1														Number of objects
	---	eCOT																Cause of transmission [▶ 514]
	---	asdu Addr																Common address of asdu
	---	eClasses																Fifo priority class [▶ 509]
---	-	info																INFORMATION OBJECT [▶ 548]
	---	fc		:= GEN														Function code/type [▶ 325]
	---	n																Information number [▶ 325]
	---	stream																Information element/object data [▶ 485]
	---	length	=	variable														
	---	data				7	6	5	4	3	2	1	0					
	---	data[0]	=	RII [▶ 328]														Return information identifier
	---	data[1]	=	CONT [▶ 326]	COU [▶ 326]	N [▶ 326]												NGD = Number of generic data sets
				1	1	1												
	---	data[2]	=	GROUP [▶ 327]														Dataset 1
	---	data[3]	=	ENTRY [▶ 327]														
	---	data[4]	=	KOD [▶ 327]														
	---	data[5]	=	DATATYPE [▶ 326]														
	---	data[6]	=	DATASIZE [▶ 326]														
	---	data[7]	=	CONT [▶ 326]	NUMBER [▶ 329]													
				1														
	---	data[8]	=	GID [▶ 329]														Generic identification data
	---	data[9..IE_C870_MAX_ASDU_DATA_BYTE]	=															Dataset 2..i



**4.3.2.4.4 C\_GRC\_NA\_3**

General command.

- obj					ASDU object [▶ 547]								
---	+ head				Reserved								
---	- ident				DATA UNIT IDENTIFIER [▶ 549]								
	---	eType = 0x14 (20)	= C_GRC_NA_3		Type identification [▶ 505]								
	---	bSQ	= TRUE		Sequence of information objects								
	---	nObj	= 1		Number of objects								
	---	eCOT			Cause of transmission [▶ 514]								
	---	asduAddr			Common address of asdu								
	---	eClasses			Fifo priority class [▶ 509]								
---	- info				INFORMATION OBJECT [▶ 548]								
	---	fc	:= GLB		Function code/type [▶ 325]								
	---	n			Information number [▶ 325]								
	---	stream			Information element/object data [▶ 485]								
	---	length	= 2										
	---	data			7	6	5	4	3	2	1	0	
	---	data[0]	=								DC		DCO = Double command
											1		[▶ 520]
	---	data[1]	=	RII [▶ 328]									Return information identifier
	---	data[2..IEC870_MAX_ASDU_DATA_BYTE]	=										Reserved

**4.3.2.4.5 C\_GC\_NA\_3**

Generic command.

- obj					ASDU object [▶ 547]							
---	+	head			Reserved							
---	-	ident			DATA UNIT IDENTIFIER [▶ 549]							
	---	eType	=	= C_GC_NA_3	Type identification [▶ 505]							
			0x15 (21)									
	---	bSQ		= TRUE	Sequence of information objects							
	---	nObj		= 1	Number of objects							
	---	eCOT			Cause of transmission [▶ 514]							
	---	asdu Addr			Common address of asdu							
	---	eClas s			Fifo priority class [▶ 509]							
---	-	info			INFORMATION OBJECT [▶ 548]							
	---	fc		:= GEN	Function code/type [▶ 325]							
	---	n			Information number [▶ 325]							
	---	strea m			Information element/object data [▶ 485]							
	---	lengt h	=	$2 + (i * 3)$								
	---	data			7	6	5	4	3	2	1	0
	---	data[0] =		<u>RII</u> [▶ 328]	Return information identification							
	---	data[1] =		<u>NOG</u> [▶ 328]	Number of generic data sets							
	---	data[2] =		<u>GROUP</u> [▶ 327]	GIN = Generic identification number							Daset 1
	---	data[3] =		<u>ENTRY</u> [▶ 327]								
	---	data[4] =		<u>KOD</u> [▶ 327]	Kind of description							
	---	data[5..IE C870 _MAX _ASD U_DA TA_B YTE] =										Daset 2..i

#### 4.3.2.4.6 C\_ODT\_NA\_3

Order of disturbance data transmission.

- obj					ASDU object [▶ 547]							
---	+ head				Reserved							
---	- ident				DATA UNIT IDENTIFIER [▶ 549]							
	---	eType	= 0x18 (24)	= C_ODT_NA_3	Type identification [▶ 505]							
	---	bSQ		= TRUE	Sequence of information objects							
	---	nObj		= 1	Number of objects							
	---	eCOT			Cause of transmission [▶ 514]							
	---	asduAddr			Common address of asdu							
	---	eClasses			Fifo priority class [▶ 509]							
---	- info				INFORMATION OBJECT [▶ 548]							
	---	fc			Function code/type [▶ 325]							
	---	n		= 0	not used							
	---	stream			Information element/object data [▶ 485]							
		---	length	= 5								
		---	data		7	6	5	4	3	2	1	0
		---	data[0]	= TOO [▶ 329]	Type of order							
		---	data[1]	= TOV [▶ 329]	Type of disturbance value							
		---	data[2..3]	= FAN [▶ 327]	Fault number							
		---	data[4]	= ACC [▶ 325]	Actual channel							
		---	data[5..IEC870_MAX_ASDU_DATA_BYTE]	=								

**4.3.2.4.7 C\_ADT\_NA\_3**

Acknowledgement for disturbance data transmission.

- obj					ASDU object [▶ 547]							
---	+ head				Reserved							
---	- ident				DATA UNIT IDENTIFIER [▶ 549]							
	---	eType	= 0x19 (25)	= C_ADT_NA_3	Type identification [▶ 505]							
	---	bSQ		= TRUE	Sequence of information objects							
	---	nObj		= 1	Number of objects							
	---	eCOT			Cause of transmission [▶ 514]							
	---	asduAddr			Common address of asdu							
	---	eClasses			Fifo priority class [▶ 509]							
---	- info				INFORMATION OBJECT [▶ 548]							
	---	fc			Function code/type [▶ 325]							
	---	n		= 0	not used							
	---	stream			Information element/object data [▶ 485]							
		---	length	= 5								
		---	data		7	6	5	4	3	2	1	0
		---	data[0]	= TOO [▶ 329]	Type of order							
		---	data[1]	= TOV [▶ 329]	Type of disturbance value							
		---	data[2..3]	= FAN [▶ 327]	Fault number							
		---	data[4]	= ACC [▶ 325]	Actual channel							
		---	data[5..IEC870_MAX_ASDU_DATA_BYTE]	=								

**4.3.2.5 Funktionstyp (code)**

Typ	Beschreibung
0..127	Reserviert (privater Bereich)
128	Distanzschutz
129	Nicht benutzt (kompatibler Bereich)
130..143	Reserviert (privater Bereich)
144..145	Nicht benutzt (kompatibler Bereich)
146..159	Reserviert (privater Bereich)
160	Überstromzeitschutz
161	Nicht benutzt (kompatibler Bereich)
162..175	Reserviert (privater Bereich)
176	Transformator-Differentialschutz
177	Nicht benutzt (kompatibler Bereich)
178..191	Reserviert (privater Bereich)
192	Leitungsdifferentialschutz
193	Nicht benutzt (kompatibler Bereich)
194..207	Reserviert (privater Bereich)
208	Nicht benutzt (kompatibler Bereich)
209	Nicht benutzt (kompatibler Bereich)
210..223	Reserviert (privater Bereich)
224	Nicht benutzt (kompatibler Bereich)
225	Nicht benutzt (kompatibler Bereich)
226..239	Reserviert (privater Bereich)
240	Nicht benutzt (kompatibler Bereich)
241	Nicht benutzt (kompatibler Bereich)
242..253	Reserviert (privater Bereich)
254	Generische Funktion (GEN)
255	Globale Funktion (GLB)

**4.3.2.6 Informationsnummer**

Nummer	Überwachungsrichtung	Steuerungsrichtung
0..15	Systemfunktionen	Systemfunktionen
16..31	Zustand	Allgemeine Befehle
32..47	Überwachung	Nicht benutzt
48..63	Erdschlüsse	Nicht benutzt
64..127	Kurzschlüsse	Nicht benutzt
128..143	Automatische Wiedereinschaltung	Nicht benutzt
144..159	Betriebsmesswerte	Nicht benutzt
160..239	Nicht benutzt	Nicht benutzt
240..255	Generische Funktionen	Generische Funktionen

**4.3.2.7 Informationselemente**

**4.3.2.7.1 ACC**

Actual channel: <0..255>

**4.3.2.7.2 ASC**

ASCII 8 bit code: <0..255>

**4.3.2.7.3 COL**

Compatibility level: <0..255>

**4.3.2.7.4 CONT**

Continuous flag

- <0> no following ASDU with the same RII;
- <1> following ASDU has the same RII;

**4.3.2.7.5 COUNT**

One bit counter for ASDU with equal RII: <0..1>

**4.3.2.7.6 DATASIZE**

Data size: <1..255>

**4.3.2.7.7 DATATYPE**

Data type: <0..255>

- <0> no data
- <1> OS8ASCII
- <2> PACKEDBITSTRING
- <3> UI
- <4> I
- <5> UF
- <6> F
- <7> R32
- <8> R64
- <9> Double poing information
- <10> Single poing information
- <11>
- <12>
- <13>
- <14>
- <15>
- <16>
- <17>
- <18>
- <19>
- <20>
- <21>
- <22>
- <23> data struct
- <24> index

- <25..255> reserved

#### 4.3.2.7.8 ENTRY

Entry identification: <0..255>

#### 4.3.2.7.9 FAN

Fault number: <0..65535>

#### 4.3.2.7.10 GROUP

Group identification: <0..255>

#### 4.3.2.7.11 INT

Interval: <1..65535> [ $\mu$ s]

#### 4.3.2.7.12 KOD

Kind of description: <0..255>

- <0> no KOD specified
- <1> actual value
- <2> default value
- <3> range (min, max, step size)
- <4> reserved
- <5> precision
- <6> factor
- <7> % reference
- <8> enumeration
- <9> dimension
- <10> description
- <11> reserved
- <12> password entry
- <13> is read only
- <14> is write only
- <15> reserved
- <16> reserved
- <17> reserved
- <18> reserved
- <19> corresponding function type and information number
- <20> corresponding event
- <21> enumerated text array
- <22> enumerated value array
- <23> related entries
- <24..255> reserved

#### 4.3.2.7.13 MVAL

Measured value: <-1..+1.-2E-12>

**4.3.2.7.14 NDV**

Number of relevant disturbance values per ASDU: <1..255>

- <1..25> used
- <26..255> not used

**4.3.2.7.15 NFE**

Number of the ASDU's first information element: <0..65535>

**4.3.2.7.16 NO**

Number of generic data sets: <0..63>

**4.3.2.7.17 NOC**

Number of channels: <0..255>

**4.3.2.7.18 NOE**

Number of information elements of a channel: <0..65535>

**4.3.2.7.19 NOF**

Number of grid faluts: <0..65535>

**4.3.2.7.20 NOG**

Number of generic identifications: <0..255>

**4.3.2.7.21 NOT**

Number of tags: <1..255>

**4.3.2.7.22 RET**

Relative time: <0..65535>

**4.3.2.7.23 RII**

Return information identifier.

USINT <0..255>

**4.3.2.7.24 SCN**

Scan number: <0..255>

**4.3.2.7.25 SDV**

Single disturbance value: <-1..+1-2E-15>

**4.3.2.7.26 SIN**

Supplementary information: <0..255>



**4.3.2.7.27 TAP**

Tag position: <0..65535>

**4.3.2.7.28 TOO**

Type of order: <1..255>

**4.3.2.7.29 TOV**

Type of disturbance value: <0..255>

**4.3.2.7.30 GID**

Generic identification data.

**4.3.2.7.31 NUMBER**

Number.

**4.3.2.7.32 Other information elements****Quality descriptor****QDS****OV**

Overflow quality flag:

- <0> = no overflow;
- <1> = overflow;

**BL**

Blocked quality flag:

- <0> = not blocked;
- <1> = blocked;

**SB**

Substituted quality flag:

- <0> = not substituted;
- <1> = substituted;

**NT**

Topical quality flag:

- <0> = topical;
- <1> = not topical;

**IV**

Invalid quality flag:

- <0> = valid;
- <1> = invalid;

**CY**

Carry flag:

- <0> = no carry;
- <1> = carry;

**CA**

Adjusted flag:

- <0> = Counter was not adjusted;
- <1> = Counter was adjusted;

**EI**

Elapsed flag:

- <0> = Elapsed time valid;
- <1> = Elapsed time not valid;

**RES****ER****OTEV****TEST****TM****TP**

## 4.4 IEC 60870-5-104

TF6500 implementiert zwei Schnittstellen über die einzelne IEC 60870-5-104 ASDUs (Dienstdateneinheiten der Anwendungsschicht) verschickt und empfangen werden können.

### "High level"-Schnittstelle:

Bei dieser Schnittstelle handelt es sich um eine sogenannte "Ein-Baustein-Lösung". Alle Funktionalitäten sind in einem SPS-Baustein gekapselt. Der Baustein implementiert die wichtigsten Dienste und Funktionen. Diese Implementierung ist für über 90% der Anwendungen ausreichend.

**Pro:** Sehr kleiner SPS-Programmieraufwand um eine laufende Applikation zu erhalten; Alle Dienste wie Generalabfrage, Zählerabfrage, Uhrzeitsynchronisation, Befehlsausführung, spontane Datenübertragung usw. sind bereits in dem Baustein implementiert und werden automatisch ausgeführt; Das Mapping der IEC->SPS Prozessdaten und das der Datenpunkte wird über Funktionsaufrufe konfiguriert; Der SPS-Programmierer muss nicht sehr gut mit der Protokollnorm vertraut sein;

**Contra:** Die SPS-Applikation hat nur einen geringen Einfluss auf die Protokollausführung; Kein Einfluss auf die Ausführung der Dienste, diese werden intern automatisch ausgeführt; Zeitstempel werden von dem Baustein automatisch generiert und können nicht verändert (von extern übergeben) werden; Es ist z.B. nur die direkte Befehlsausführung möglich; Schlechtere Performance bei vielen Datenpunkten.

Diese Schnittstelle empfiehlt sich wenn Sie:

- Mit der Protokollnorm nicht vertraut sind;
- Eine einfache Applikation mit einer handvoll Datenpunkten implementieren möchten (<1000);
- Keine großen Performace-Anforderungen an die Applikation stellen;

- Keine besondere Befehlsausführung wie Select/Execute oder Daten + Zeitstempel von externen Geräten versenden möchten;
- Keine Funktionalitäten benötigen die laut Kompatibilitätsliste nicht unterstützt werden;

**"Low level"-Schnittstelle:**

Diese Schnittstelle setzt bei dem Protokollstack tiefer an und ermöglicht das Versenden und/oder Empfangen einzelner Frames (ASDU's). Innerhalb der Protokollstruktur liegt diese Schnittstelle oberhalb der Transportschicht (4) und implementiert bereits die APCI-Funktionen (Protokollsteuerinformationen der Anwendungsschicht, siehe untere Tabelle). Anwendungsfunktionen wie z.B. Generalabfrage und Zählerabfrage sind in der Schnittstelle nicht implementiert, der Anwender kann aber mit Hilfe der Schnittstelle diese Anwendungsfunktionen selber implementieren.

**Pro:** Sehr flexibel; Alle Eigenschaften im ASDU-Frame können manipuliert werden (z.B. ein eigener Zeitstempel, Select/Execute oder eine besondere Befehlsausführung usw.); Weil nur die benötigten Dienste implementiert werden kann eine hohe Performance erreicht werden; Hohe Performance bei vielen Datenpunkten;

**Contra:** Größerer Programmieraufwand; Alle Dienste wie Generalabfrage, Zählerabfrage, Uhrzeitsynchronisation, Befehlsausführung, spontane Datenübertragung usw. müssen von dem SPS-Programmierer selbst implementiert (ausprogrammiert) werden; Der SPS-Programmierer muss mit der Protokollnorm vertraut sein.

Diese Schnittstelle empfiehlt sich wenn Sie:

- Mit der Protokollnorm vertraut sind;
- Eine Protokollkonverter-Applikation implementieren möchten;
- In der Applikation die verfügbaren Norm-Funktionalitäten fast vollständig implementieren müssen;
- Besondere Funktionalitäten wie z.B. das weiterleiten der Zeitstempel von einem Modbusgerät oder die Kontrolle über die Befehlsausführung haben möchten;
- Funktionalitäten benötigen die laut Kompatibilitätsliste nicht unterstützt werden;
- Viele Datenpunkte haben (>1000) und eine hohe Performance benötigen;

Protokollstruktur des Endsystems:

**Protokollstruktur des Endsystems:**

Auswahl von Anwendungsfunktionen aus IEC 60870-5-5 nach IEC 60870-5-101	Anwenderprozess
Auswahl von ASDU (Dienstdateneinheiten der Anwendungsschicht) aus IEC 60870-5-101 und IEC 60870-5-104	Anwendungsschicht (7)
APCI (Protokollsteuerinformation der Anwendungsschicht) Transportschnittstelle (Anwender zur TCP-Schnittstelle)	
Auswahl aus der TCP/IP-Protokollsammlung (RFC 2200)	Transportschicht (4)
	Vermittlungsschicht (3)
	Verbindungsschicht (2)
	Physikalische Schicht (1)



Die Schichten 5 und 6 werden nicht benutzt.

**4.4.1 IEC 60870-5-104 telegram structure**

APCI = Application Protocol Control Information

ASDU = Application Service Data Unit

APDU = Application Protocol Data Unit

**Telegram format with variable length**

This frame type is used to transmit user data between controlling and controlled station

byte\bit	7	6	5	4	3	2	1	0				
0	Start byte (0x68)								APCI		APDU	
1	Length of the APDU (max. 253)											
2	Control field 1											
3	Control field 2											
4	Control field 3											
5	Control field 4								ASDU			
6	Type identification [ <a href="#">▶ 500</a> ]											
7	SQ	Number of objects										
8	T	P/N	Cause of transmission (COT) [ <a href="#">▶ 509</a> ]									
9	Originator address (ORG)											
10	ASDU address fields											
11	(2 octets)											
12	Information object address fields (IOA)											
13	(3 octets)											
14												
15	Object information											
...												
...												
...												
n-1												
n												

**Telegram format with fixed length**

byte/bit	7	6	5	4	3	2	1	0		
0	Start byte (0x68)								APCI	
1	4 (Length of the APDU)									
2	Control field 1									
3	Control field 2									
4	Control field 3									
5	Control field 4									

**Control field formats**

Two types of control field formats: I-Format, S-Format are used to perform numbered information transfer.

The third: U-Format control field is used to perform unnumbered link layer control functions.

**I-Format**

byte\bit	7	6	5	4	3	2	1	0
0	Send sequence number N(S) LSB							0
1	Send sequence number N(S) MSB							
2	Receive sequence number N(R) LSB							0
3	Receive sequence number N(R) MSB							

**S-Format**

byte\bit	7	6	5	4	3	2	1	0
0	0						0	1
1	0							
2	Receive sequence number N(R) LSB							0
3	Receive sequence number N(R) MSB							

**U-Format**

byte\bit	7	6	5	4	3	2	1	0
0	TESTFR		STOPDT		STARTDT		1	1
1	0							
2	0							
3	0							

**Simple samples and explanations**

104 substation configuration: COT = 2 octets (includes originator address), ASDU address = 2 octets, IOA address = 3 octets

**Sample 1**

68 0E 4E 14 7C 00 65 01 0A 00 0C 00 00 00 00 05

LPDU bytes	Explanation
68	Start byte
0E	Length of the APDU = 14
4E	Send sequence number N(S) LSB, bit 0 = 0 => I-Format
14	Send sequence number N(S) MSB
7C	Receive sequence number N(R) LSB
00	Receive sequence number N(R) MSB
65	Type identification: C_CI_NA_1 (counter interrogation command)
01	Number of objects = 1
0A	Cause of transmission = 10 (activation termination)
00	Originator address = 0
0C 00	Common ASDU address (2 octets) = 12 dec.
00 00 00	Object address (3 octets)
05	Counter interrogation request qualifier = 5 (general counter interrogation)

**Sample 2**

68 34 5A 14 7C 00 0B 07 03 00 0C 00 10 30 00 BE 09 00 11 30 00 90 09 00 0E 30 00 75 00 00 28 30 00 25 09 00 29 30 00 75 00 00 0F 30 00 0F 0A 00 2E 30 00 AE 05 00

LPDU bytes	Explanation
68	Start byte
34	Length of the APDU = 52
5A	Send sequence number N(S) LSB, bit 0 = 0 => I-Format
14	Send sequence number N(S) MSB
7C	Receive sequence number N(R) LSB
00	Receive sequence number N(R) MSB
0B	Type identification: M_ME_NB_1(measured value, scaled value)
07	Number of objects = 7
03	Cause of transmission = 3 (spontaneous)
00	Originator address = 0
0C 00	Common ASDU address (2 octets) = 12 dec.
10 30 00	Object address (3 octets) of <b>first</b> information object
BE 09 00	Scaled value + QDS (quality descriptor) of <b>first</b> information object
11 30 00	Object address (3 octets) of <b>second</b> information object
90 09 00	Scaled value + QDS (quality descriptor) of <b>second</b> information object
0E 30 00	Object address (3 octets) of <b>third</b> information object
75 00 00	Scaled value + QDS (quality descriptor) of <b>third</b> information object
28 30 00 25 09 00 29 30 00 75 00 00 0F 30 00 0F 0A 00 2E 30 00 AE 05 00	Object address + Scaled value + QDS (quality descriptor) of information object four to seven

**Sample 3**

68 04 01 00 7E 14

LPDU bytes	Explanation
68	Start byte
04	Length of the APDU = 4
01	bits 2..7 reserved, bit 0 = 1 and bit 1 = 0 => S-Format
00	reserved
7E	Receive sequence number N(R) LSB
14	Receive sequence number N(R) MSB

## **4.4.2 ASDU-Objektbeschreibung**

### **4.4.2.1 Standard IEC 60870-5-104 Datentypen**

Type	Dez	Hex	Beschreibung
ASDU_TYPEUNDEF	0	0x00	Wird nicht verwendet
<a href="#">M_SP_NA_1 [► 122]</a>	1	0x01	Einzelmeldung
<a href="#">M_SP_TA_1</a>	2	0x02	Einzelmeldung mit einem Zeitstempel
<a href="#">M_DP_NA_1 [► 128]</a>	3	0x03	Doppelmeldung
<a href="#">M_DP_TA_1</a>	4	0x04	Doppelmeldung mit einem Zeitstempel
<a href="#">M_ST_NA_1 [► 134]</a>	5	0x05	Stufenstellungsmeldung
<a href="#">M_ST_TA_1</a>	6	0x06	Stufenstellungsmeldung mit einem Zeitstempel
<a href="#">M_BO_NA_1 [► 140]</a>	7	0x07	Bitmuster von 32 bit
<a href="#">M_BO_TA_1</a>	8	0x08	Bitmuster von 32 bit mit einem Zeitstempel
<a href="#">M_ME_NA_1 [► 146]</a>	9	0x09	Messwert, normierter Wert
<a href="#">M_ME_TA_1</a>	10	0x0A	Messwert, normierter Wert mit einem Zeitstempel
<a href="#">M_ME_NB_1 [► 154]</a>	11	0x0B	Messwert, skaliertes Wert
<a href="#">M_ME_TB_1</a>	12	0x0C	Messwert, skaliertes Wert mit einem Zeitstempel
<a href="#">M_ME_NC_1 [► 160]</a>	13	0x0D	Messwert, Gleitkommazahl mit einfacher Genauigkeit
<a href="#">M_ME_TC_1</a>	14	0x0E	Messwert, Gleitkommazahl mit einfacher Genauigkeit und einem Zeitstempel
<a href="#">M_IT_NA_1 [► 166]</a>	15	0x0F	Zählwerte
<a href="#">M_IT_TA_1</a>	16	0x10	Zählwerte mit einem Zeitstempel
<a href="#">M_EP_TA_1</a>	17	0x11	Schutzereignis mit einem Zeitstempel
<a href="#">M_EP_TB_1</a>	18	0x12	Gepackte Anregungen des Schützes mit einem Zeitstempel
<a href="#">M_EP_TC_1</a>	19	0x13	Gepackte Auslösungen des Schützes mit einem Zeitstempel
<a href="#">M_PS_NA_1 [► 230]</a>	20	0x14	Gepackte Einzelmeldungen mit Zustandsanzeige
<a href="#">M_ME_ND_1 [► 152]</a>	21	0x15	Messwert, normierter Wert ohne Qualitätskennung
ASDU_TYPE_22..29	22..29	0x16..0x1D	Reserviert (Standardbereich)
<a href="#">M_SP_TB_1 [► 126]</a>	30	0x1E	Einzelmeldung mit einem Zeitstempel CP56Time2a
<a href="#">M_DP_TB_1 [► 132]</a>	31	0x1F	Doppelmeldung mit einem Zeitstempel CP56Time2a
<a href="#">M_ST_TB_1 [► 138]</a>	32	0x20	Stufenstellungsmeldung mit einem Zeitstempel CP56Time2a
<a href="#">M_BO_TB_1 [► 144]</a>	33	0x21	Bitmuster von 32 bit mit einem Zeitstempel CP56Time2a
<a href="#">M_ME_TD_1 [► 150]</a>	34	0x22	Messwert, normierter Wert mit einem Zeitstempel CP56Time2a
<a href="#">M_ME_TE_1 [► 158]</a>	35	0x23	Messwert, skaliertes Wert mit einem Zeitstempel CP56Time2a
<a href="#">M_ME_TF_1 [► 164]</a>	36	0x24	Messwert, Gleitkommazahl mit einfacher Genauigkeit und einem Zeitstempel CP56Time2a
<a href="#">M_IT_TB_1 [► 170]</a>	37	0x25	Zählwerte mit einem Zeitstempel CP56Time2a
<a href="#">M_EP_TD_1 [► 222]</a>	38	0x26	Schutzereignis mit einem Zeitstempel CP56Time2a



Type	Dez	Hex	Beschreibung
<a href="#">M EP TE 1 [► 224]</a>	39	0x27	Gepackte Anregungen des Schutzes mit einem Zeitstempel CP56Time2a
<a href="#">M EP TF 1 [► 227]</a>	40	0x28	Gepackte Auslösungen des Schutzes mit einem Zeitstempel CP56Time2a
ASDU_TYPE_41..44	41..44	0x29..0x2C	Reserviert (Standardbereich)
<a href="#">C SC NA 1 [► 172]</a>	45	0x2D	Einzelbefehl
<a href="#">C DC NA 1 [► 176]</a>	46	0x2E	Doppelbefehl
<a href="#">C RC NA 1 [► 180]</a>	47	0x2F	Stufenstellbefehl
<a href="#">C SE NA 1 [► 184]</a>	48	0x30	Sollwert-Stellbefehl, normierter Wert
<a href="#">C SE NB 1 [► 188]</a>	49	0x31	Sollwert-Stellbefehl, skaliertes Wert
<a href="#">C SE NC 1 [► 192]</a>	50	0x32	Sollwert-Stellbefehl, Gleitkommazahl mit einfacher Genauigkeit
<a href="#">C BO NA 1 [► 196]</a>	51	0x33	Bitmusterbefehl von 32 bit
ASDU_TYPE_52..57	52..57	0x34..0x39	Reserviert (Standardbereich)
<a href="#">C SC TA 1 [► 174]</a>	58	0x3A	Einzelbefehl mit einem Zeitstempel CP56Time2a
<a href="#">C DC TA 1 [► 178]</a>	59	0x3B	Doppelbefehl mit einem Zeitstempel CP56Time2a
<a href="#">C RC TA 1 [► 182]</a>	60	0x3C	Stufenstellbefehl mit einem Zeitstempel CP56Time2a
<a href="#">C SE TA 1 [► 186]</a>	61	0x3D	Sollwert-Stellbefehl, normierter Wert mit einem Zeitstempel CP56Time2a
<a href="#">C SE TB 1 [► 190]</a>	62	0x3E	Sollwert-Stellbefehl, skaliertes Wert mit einem Zeitstempel CP56Time2a
<a href="#">C SE TC 1 [► 194]</a>	63	0x3F	Sollwert-Stellbefehl, Gleitkommazahl mit einfacher Genauigkeit und einem Zeitstempel CP56Time2a
<a href="#">C BO TA 1 [► 198]</a>	64	0x40	Bitmusterbefehl von 32 bit mit einem Zeitstempel CP56Time2a
ASDU_TYPE_65..69	65..69	0x41..0x45	Reserviert (Standardbereich)
<a href="#">M EI NA 1 [► 202]</a>	70	0x46	Initialisierungsende
ASDU_TYPE_71..99	71..99	0x47..0x63	Reserviert (Standardbereich)
<a href="#">C IC NA 1 [► 206]</a>	100	0x64	(General-, Stations-) Abfragebefehl
<a href="#">C CI NA 1 [► 208]</a>	101	0x65	Zähler-Abfragebefehl
<a href="#">C RD NA 1 [► 212]</a>	102	0x66	Abfragebefehl
<a href="#">C CS NA 1 [► 204]</a>	103	0x67	Uhrzeit-Synchronisierungsbefehl
<a href="#">C TS NA 1</a>	104	0x68	Prüfbefehl
<a href="#">C RP NA 1 [► 210]</a>	105	0x69	Prozess-Rücksetzbefehl
<a href="#">C CD NA 1</a>	106	0x6A	Befehl zur Telegrammlaufzeit-Erfassung
<a href="#">C TS TA 1 [► 368]</a>	107	0x6B	Prüfbefehl mit einem Zeitstempel CP56Time2a
ASDU_TYPE_108..109	108..109	0x6C..0x6D	Reserviert (Standardbereich)
<a href="#">P ME NA 1 [► 232]</a>	110	0x6E	Parameter für Messwerte, normierter Wert
<a href="#">P ME NB 1 [► 234]</a>	111	0x6F	Parameter für Messwerte, skaliertes Wert
<a href="#">P ME NC 1 [► 236]</a>	112	0x70	Parameter für messwerte, Gleitkommazahl mit einfacher Genauigkeit
<a href="#">P AC NA 1 [► 238]</a>	113	0x71	Parameter für Aktivierung
ASDU_TYPE_114..119	114..119	0x72..0x77	Reserviert (Standardbereich)
<a href="#">F FR NA 1</a>	120	0x78	File ready

Type	Dez	Hex	Beschreibung
F_SR_NA_1	121	0x79	Section ready
F_SC_NA_1	122	0x7A	Call directory, select file, call file, call section
F_LS_NA_1	123	0x7B	Last section, last segment
F_FA_NA_1	124	0x7C	ACK file, ACK section
F_SG_NA_1	125	0x7D	Segment
F_DR_TA_1	126	0x7E	Verzeichniss
ASDU_TYPE_127..255	127..255	0x7F..0xFF	Reserviert (benutzerdefinierter Bereich)

### 4.4.2.2 Single-point information

#### 4.4.2.2.1 M\_SP\_NA\_1

Single-point information without time tag.

- obj					ASDU object [▶ 483]								
---	+ head				Reserved								
---	- ident				DATA UNIT IDENTIFIER [▶ 484]								
	---	eType	= 0x01 (1)	= M_SP_NA_1	Type identification [▶ 500]								
	---	bSQ		= FALSE	Sequence of information objects								
	---	nObj		= 1	Number of objects								
	---	bT			Test								
	---	bPN			Positive/negative confirmation/activation								
	---	eCOT			Cause of transmission [▶ 509]								
	---	nORG			Originator address								
	---	asduAddr			Common address of asdu								
	---	eClass			Fifo priority class [▶ 509]								
---	- info				INFORMATION OBJECT [▶ 485]								
	---	objAddr			Information object address								
	---	stream			Information element/object data [▶ 485]								
		---	length	= 1									
		---	data		7	6	5	4	3	2	1	0	
		---	data[0]	=	IV [▶ 329]	NT [▶ 329]	SB [▶ 329]	BL [▶ 329]	0	0	0	SPI [▶ 526]	SIQ = Single-point information with quality descriptor
		---	data[1..IEC870_MAX_ASDU_DATA_BYTE]	=	Reserved								

**4.4.2.2 M\_SP\_TB\_1**

Single-point information with CP56Time2a time tag.

- obj					ASDU object [▶ 483]								
---	+ head				Reserved								
---	- ident				DATA UNIT IDENTIFIER [▶ 484]								
	---	eType	= 0x1E (30)	= M_SP_TB_1	Type identification [▶ 500]								
	---	bSQ		= FALSE	Sequence of information objects								
	---	nObj		= 1	Number of objects								
	---	bT			Test								
	---	bPN			Positive/negative confirmation/activation								
	---	eCOT			Cause of transmission [▶ 509]								
	---	nORG			Originator address								
	---	asduAddr			Common address of asdu								
	---	eClass			Fifo priority class [▶ 509]								
---	- info				INFORMATION OBJECT [▶ 485]								
	---	objAddr			Information object address								
	---	stream			Information element/object data [▶ 485]								
		---	length	= 8									
		---	data		7	6	5	4	3	2	1	0	
		---	data[0]	=	IV [▶ 329]	NT [▶ 329]	SB [▶ 329]	BL [▶ 329]	0	0	0	SPI [▶ 526]	SIQ = Single-point information with quality descriptor
		---	data[1..7]	=	CP56Time2a [▶ 531]								
		---	data[8..IEC870_MAX_ASDU_DATA_BYTE]	=	Reserved								

**4.4.2.3 Double-point information**

**4.4.2.3.1 M\_DP\_NA\_1**

Double-point information without time tag.

- obj					ASDU object [▶ 483]								
---	+ head				Reserved								
---	- ident				DATA UNIT IDENTIFIER [▶ 484]								
	---	eType	= 0x03 (3)	= M_DP_NA_1	Type identification [▶ 500]								
	---	bSQ		= FALSE	Sequence of information objects								
	---	nObj		= 1	Number of objects								
	---	bT			Test								
	---	bPN			Positive/negative confirmation/activation								
	---	eCOT			Cause of transmission [▶ 509]								
	---	nORG			Originator address								
	---	asduA ddr			Common address of asdu								
	---	eClass			Fifo priority class [▶ 509]								
---	- info				INFORMATION OBJECT [▶ 485]								
	---	objAdd r			Information object address								
	---	stream			Information element/object data [▶ 485]								
		---	length	= 1									
		---	data		7	6	5	4	3	2	1	0	
		---	data[0]	=	<u>IV</u> [▶ 329]	<u>NT</u> [▶ 329]	<u>SB</u> [▶ 329]	<u>BL</u> [▶ 329]	0	0	<u>DPI</u> [▶ 527] 1	DIQ = Double-point information with quality descriptor	
		---	data[1..	.IEC87 0_MAX_AS DU_DATA_BYTE]	Reserved								

#### 4.4.2.3.2 M\_DP\_TB\_1

Double-point information with CP56Time2a time tag.

- obj					ASDU object [▶ 483]								
---	+ head				Reserved								
---	- ident				DATA UNIT IDENTIFIER [▶ 484]								
	---	eType	= 0x1F (31)	= M_DP_TB_1	Type identification [▶ 500]								
	---	bSQ		= FALSE	Sequence of information objects								
	---	nObj		= 1	Number of objects								
	---	bT			Test								
	---	bPN			Positive/negative confirmation/activation								
	---	eCOT			Cause of transmission [▶ 509]								
	---	nORG			Originator address								
	---	asduAddr			Common address of asdu								
	---	eClass			Fifo priority class [▶ 509]								
---	- info				INFORMATION OBJECT [▶ 485]								
	---	objAddr			Information object address								
	---	stream			Information element/object data [▶ 485]								
		---	length	= 8									
		---	data		7	6	5	4	3	2	1	0	
		---	data[0]	=	IV [▶ 329]	NT [▶ 329]	SB [▶ 329]	BL [▶ 329]	0	0	DPI [▶ 527]	1	DIQ = Double-point information with quality descriptor
		---	data[1..7]	=	CP56Time2a [▶ 531]								
		---	data[8..IEC870_MAX_AS_DUTA_BYTE]	=	Reserved								

**4.4.2.4 Step position information**

**4.4.2.4.1 M\_ST\_NA\_1**

Step position information without time tag.

- obj					ASDU object [▶ 483]								
---	+ head				Reserved								
---	- ident				DATA UNIT IDENTIFIER [▶ 484]								
	---	eType	= 0x05 (5)	= M_ST_NA_1	Type identification [▶ 500]								
	---	bSQ		= FALSE	Sequence of information objects								
	---	nObj		= 1	Number of objects								
	---	bT			Test								
	---	bPN			Positive/negative confirmation/activation								
	---	eCOT			Cause of transmission [▶ 509]								
	---	nORG			Originator address								
	---	asduA addr			Common address of asdu								
	---	eClass			Fifo priority class [▶ 509]								
---	- info				INFORMATION OBJECT [▶ 485]								
	---	objAdd r			Information object address								
	---	stream			Information element/object data [▶ 485]								
		---	length	= 2									
		---	data		7	6	5	4	3	2	1	0	
		---	data[0]	=	VTI [▶ 240]								Value with transient state indication
		---	data[1]	=	IV [▶ 329]	NT [▶ 329]	SB [▶ 329]	BL [▶ 329]	0	0	0	OV [▶ 329] 1	QDS = Quality descript or
		---	data[2.. .IEC87 0_MA X_AS DU_D ATA_ BYTE]	=	Reserved								

**4.4.2.4.2 M\_ST\_TB\_1**

Step position information with CP56Time2a time tag.

- obj															ASDU object [▶ 483]
---	+ head														Reserved
---	- ident														DATA UNIT IDENTIFIER [▶ 484]
	---	eType	= 0x20 (32)	= M_ST_TB_1											Type identification [▶ 500]
	---	bSQ		= FALSE											Sequence of information objects
	---	nObj		= 1											Number of objects
	---	bT													Test
	---	bPN													Positive/negative confirmation/activation
	---	eCOT													Cause of transmission [▶ 509]
	---	nORG													Originator address
	---	asduA ddr													Common address of asdu
	---	eClass													Fifo priority class [▶ 509]
---	- info														INFORMATION OBJECT [▶ 485]
	---	objAdd r													Information object address
	---	stream													Information element/object data [▶ 485]
		---	length	= 9											
		---	data			7	6	5	4	3	2	1	0		
		---	data[0] =			VTI [▶ 240]		Value with transient state indication							
		---	data[1] =			IV [▶ 329]	NT [▶ 329]	SB [▶ 329]	BL [▶ 329]	0	0	0	OV [▶ 329]	QDS = Quality descriptor	
		---	data[2. .8] =			CP56Time2a [▶ 531]		Seven octets binary time tag							
		---	data[9. .IEC87 0_MA X_AS DU_D ATA_ BYTE] =			Reserved									

**4.4.2.5 Bitstring of 32 bits**

**4.4.2.5.1 M\_BO\_NA\_1**

Bitstring of 32 bits without time tag.

- obj					ASDU object [▶ 483]								
---	+ head				Reserved								
---	- ident				DATA UNIT IDENTIFIER [▶ 484]								
	---	eType	= 0x07 (7)	= M_BO_NA_1	Type identification [▶ 500]								
	---	bSQ		= FALSE	Sequence of information objects								
	---	nObj		= 1	Number of objects								
	---	bT			Test								
	---	bPN			Positive/negative confirmation/activation								
	---	eCOT			Cause of transmission [▶ 509]								
	---	nORG			Originator address								
	---	asduA ddr			Common address of asdu								
	---	eClass			Fifo priority class [▶ 509]								
---	- info				INFORMATION OBJECT [▶ 485]								
	---	objAdd r			Information object address								
	---	stream			Information element/object data [▶ 485]								
		---	length	= 5									
		---	data		7	6	5	4	3	2	1	0	
		---	data[0. .3] =		BSI [▶ 241]								
		---	data[4] =		<u>IV</u> [▶ 329]	<u>NT</u> [▶ 329]	<u>SB</u> [▶ 329]	<u>BL</u> [▶ 329]	0	0	0	<u>OV</u> [▶ 329]	<u>QDS</u> [▶ 329] = Quality descriptor
		---	data[5. .IEC87 0_MA X_AS DU_D ATA_ BYTE] =		Reserved								

#### 4.4.2.5.2 M\_BO\_TB\_1

Bitstring of 32 bits with CP56Time2a time tag.



- obj					ASDU object [▶ 483]								
---	+ head				Reserved								
---	- ident				DATA UNIT IDENTIFIER [▶ 484]								
	---	eType	= 0x21 (33)	= M_BO_TB_1	Type identification [▶ 500]								
	---	bSQ		= FALSE	Sequence of information objects								
	---	nObj		= 1	Number of objects								
	---	bT			Test								
	---	bPN			Positive/negative confirmation/activation								
	---	eCOT			Cause of transmission [▶ 509]								
	---	nORG			Originator address								
	---	asduAddr			Common address of asdu								
	---	eClass			Fifo priority class [▶ 509]								
---	- info				INFORMATION OBJECT [▶ 485]								
	---	objAddr			Information object address								
	---	stream			Information element/object data [▶ 485]								
		---	length	= 12									
		---	data		7	6	5	4	3	2	1	0	
		---	data[0..3]	=	BSI [▶ 241]								
		---	data[4]	=	IV [▶ 329]	NT [▶ 329]	SB [▶ 329]	BL [▶ 329]	0	0	0	OV [▶ 329]	QDS [▶ 329] = Quality descriptor
		---	data[5..11]	=	CP56Time2a [▶ 531]								
		---	data[12..IEC870_M_AX_A_SDU_DATA_BYT_E]	=	Reserved								

**4.4.2.6 Measured value, normalized value**

**4.4.2.6.1 M\_ME\_NA\_1**

Measured value, normalized value without time tag.

- obj					ASDU object [▶ 483]								
---	+ head				Reserved								
---	- ident				DATA UNIT IDENTIFIER [▶ 484]								
	---	eType	= 0x09 (9)	= M_ME_NA_1	Type identification [▶ 500]								
	---	bSQ		= FALSE	Sequence of information objects								
	---	nObj		= 1	Number of objects								
	---	bT			Test								
	---	bPN			Positive/negative confirmation/activation								
	---	eCOT			Cause of transmission [▶ 509]								
	---	nORG			Originator address								
	---	asduA ddr			Common address of asdu								
	---	eClass			Fifo priority class [▶ 509]								
---	- info				INFORMATION OBJECT [▶ 485]								
	---	objAdd r			Information object address								
	---	stream			Information element/object data [▶ 485]								
		---	length	= 3									
		---	data		7	6	5	4	3	2	1	0	
		---	data[0. .1] =		NVA [▶ 241]								Normalized value
		---	data[2] =		<u>IV</u> [▶ 329]	<u>NT</u> [▶ 329]	<u>SB</u> [▶ 329]	<u>BL</u> [▶ 329]	0	0	0	<u>OV</u> [▶ 329] 1	QDS = Quality descript or
		---	data[3. .IEC87 0_MA X_AS DU_D ATA_ BYTE] =		Reserved								

**4.4.2.6.2 M\_ME\_TD\_1**

Measured value, normalized value with CP56Time2a time tag.

- obj													ASDU object [▶ 483]
---	+ head												Reserved
---	- ident												DATA UNIT IDENTIFIER [▶ 484]
	---	eType	= 0x22 (34)	= M_ME_TD_1									Type identification [▶ 500]
	---	bSQ		= FALSE									Sequence of information objects
	---	nObj		= 1									Number of objects
	---	bT											Test
	---	bPN											Positive/negative confirmation/activation
	---	eCOT											Cause of transmission [▶ 509]
	---	nORG											Originator address
	---	asduAddr											Common address of asdu
	---	eClass											Fifo priority class [▶ 509]
---	- info												INFORMATION OBJECT [▶ 485]
	---	objAddr											Information object address
	---	stream											Information element/object data [▶ 485]
		---	length	= 10									
		---	data			7	6	5	4	3	2	1	0
		---	data[0..1]	=	NVA [▶ 241]		Normalized value;						
		---	data[2]	=	IV [▶ 329]	NT [▶ 329]	SB [▶ 329]	BL [▶ 329]	0	0	0	OV [▶ 329]	QDS = Quality descriptor
		---	data[3..9]	=	CP56Time2a [▶ 531]		Seven octets binary time tag						
		---	data[10..IEC870_MAX_ASDU_DATA_BYTES]	=	Reserved								

**4.4.2.6.3 M\_ME\_ND\_1**

Measured value, normalized value without quality descriptor.

- obj															ASDU object [▶ 483]
---	+ head														Reserved
---	- ident														DATA UNIT IDENTIFIER [▶ 484]
	---	eType	= 0x15 (21)	= M_ME_ND_1											Type identification [▶ 500]
	---	bSQ		= FALSE											Sequence of information objects
	---	nObj		= 1											Number of objects
	---	bT													Test
	---	bPN													Positive/negative confirmation/activation
	---	eCOT													Cause of transmission [▶ 509]
	---	nORG													Originator address
	---	asduA ddr													Common address of asdu
	---	eClass													Fifo priority class [▶ 509]
---	- info														INFORMATION OBJECT [▶ 485]
	---	objAdd r													Information object address
	---	stream													Information element/object data [▶ 485]
		---	length	= 2											
		---	data			7	6	5	4	3	2	1	0		
		---	data[0. .1]	=											Normalized value NVA [▶ 241]
		---	data[2. .IEC87 0_MA X_AS DU_D ATA_ BYTE]	=											Reserved

**4.4.2.7 Measured value, scaled value**

**4.4.2.7.1 M\_ME\_NB\_1**

Measured value, scaled value without time tag.

- obj					ASDU object [▶ 483]							
---	+ head				Reserved							
---	- ident				DATA UNIT IDENTIFIER [▶ 484]							
	---	eType = 0x0B (11)	= M_ME_NB_1		Type identification [▶ 500]							
	---	bSQ	= FALSE		Sequence of information objects							
	---	nObj	= 1		Number of objects							
	---	bT			Test							
	---	bPN			Positive/negative confirmation/activation							
	---	eCOT			Cause of transmission [▶ 509]							
	---	nORG			Originator address							
	---	asduAddr			Common address of asdu							
	---	eClass			Fifo priority class [▶ 509]							
---	- info				INFORMATION OBJECT [▶ 485]							
	---	objAddr			Information object address							
	---	stream			Information element/object data [▶ 485]							
	---	length	= 3									
	---	data		7	6	5	4	3	2	1	0	
	---	data[0..1] =		SVA [▶ 241]		Scaled value						
	---	data[2] =		IV [▶ 329]	NT [▶ 329]	SB [▶ 329]	BL [▶ 329]	0	0	0	OV [▶ 329] 1	QDS = Quality descriptor
	---	data[3..IEC870_MAX_ASDU_DATA_BYTE] =		Reserved								

**4.4.2.7.2 M\_ME\_TE\_1**

Measured value, scaled value with CP56Time2a time tag.

- obj													ASDU object [▶ 483]	
---	+ head												Reserved	
---	- ident												DATA UNIT IDENTIFIER [▶ 484]	
	---	eType	= 0x23 (35)	= M_ME_TE_1									Type identification [▶ 500]	
	---	bSQ		= FALSE									Sequence of information objects	
	---	nObj		= 1									Number of objects	
	---	bT											Test	
	---	bPN											Positive/negative confirmation/activation	
	---	eCOT											Cause of transmission [▶ 509]	
	---	nORG											Originator address	
	---	asduA ddr											Common address of asdu	
	---	eClass											Fifo priority class [▶ 509]	
---	- info												INFORMATION OBJECT [▶ 485]	
	---	objAdd r											Information object address	
	---	stream											Information element/object data [▶ 485]	
		---	length	= 10										
		---	data			7	6	5	4	3	2	1	0	
		---	data[0. .1] =			SVA [▶ 241]		Scaled value						
		---	data[2] =			IV [▶ 329]	NT [▶ 329]	SB [▶ 329]	BL [▶ 329]	0	0	0	OV [▶ 329] 1	QDS = Quality descriptor
		---	data[3. .9] =			CP56Time2a [▶ 531]		Seven octets binary time tag						
		---	data[1 0..IEC 870_M AX_A SDU_ DATA_ _BYT _E] =			Reserved								

#### 4.4.2.8 Measured value, short floating point value

##### 4.4.2.8.1 M\_ME\_NC\_1

Measured value, short floating point value without time tag.

- obj					ASDU object [▶ 483]								
---	+ head				Reserved								
---	- ident				DATA UNIT IDENTIFIER [▶ 484]								
	---	eType = 0x0D (13)	= M_ME_NC_1		Type identification [▶ 500]								
	---	bSQ	= FALSE		Sequence of information objects								
	---	nObj	= 1		Number of objects								
	---	bT			Test								
	---	bPN			Positive/negative confirmation/activation								
	---	eCOT			Cause of transmission [▶ 509]								
	---	nORG			Originator address								
	---	asduAddr			Common address of asdu								
	---	eClass			Fifo priority class [▶ 509]								
---	- info				INFORMATION OBJECT [▶ 485]								
	---	objAddr			Information object address								
	---	stream			Information element/object data [▶ 485]								
	---	length	= 5										
	---	data			7	6	5	4	3	2	1	0	
	---	data[0..3] =			R32 [▶ 241] Short floating point value								
	---	data[4] =			IV [▶ 329]	NT [▶ 329]	SB [▶ 329]	BL [▶ 329]	0	0	0	OV [▶ 329]	QDS = Quality descriptor
	---	data[5..IEC870_MAX_ASDU_DATA_BYTE] =			Reserved								

4.4.2.8.2 M\_ME\_TF\_1

Measured value, short floating point value with CP56Time2a time tag.

- obj					ASDU object [▶ 483]								
---	+ head				Reserved								
---	- ident				DATA UNIT IDENTIFIER [▶ 484]								
	---	eType	= 0x24 (36)	= M_ME_TF_1	Type identification [▶ 500]								
	---	bSQ		= FALSE	Sequence of information objects								
	---	nObj		= 1	Number of objects								
	---	bT			Test								
	---	bPN			Positive/negative confirmation/activation								
	---	eCOT			Cause of transmission [▶ 509]								
	---	nORG			Originator address								
	---	asduAddr			Common address of asdu								
	---	eClass			Fifo priority class [▶ 509]								
---	- info				INFORMATION OBJECT [▶ 485]								
	---	objAddr			Information object address								
	---	stream			Information element/object data [▶ 485]								
		---	length	= 12									
		---	data		7	6	5	4	3	2	1	0	
		---	data[0..3]	=	R32 [▶ 241]				Short floating point value				
		---	data[4]	=	IV [▶ 329]	NT [▶ 329]	SB [▶ 329]	BL [▶ 329]	0	0	0	OV [▶ 329]	QDS = Quality descriptor
											1		
		---	data[5..11]	=	CP56Time2a [▶ 531]				Seven octets binary time tag				
		---	data[12..IEC870_MAX_ASDU_DATA_BYTES]	=	Reserved								

#### 4.4.2.9 Integrated totals

##### 4.4.2.9.1 M\_IT\_NA\_1

Integrated total without time tag.



- obj					ASDU object [▶ 483]								
---	+ head				Reserved								
---	- ident				DATA UNIT IDENTIFIER [▶ 484]								
	---	eType	= 0x0F (15)	= M_IT_NA_1	Type identification [▶ 500]								
	---	bSQ		= FALSE	Sequence of information objects								
	---	nObj		= 1	Number of objects								
	---	bT			Test								
	---	bPN			Positive/negative confirmation/activation								
	---	eCOT			Cause of transmission [▶ 509]								
	---	nORG			Originator address								
	---	asduA ddr			Common address of asdu								
	---	eClass			Fifo priority class [▶ 509]								
---	- info				INFORMATION OBJECT [▶ 485]								
	---	objAdd r			Information object address								
	---	stream			Information element/object data [▶ 485]								
		---	length	= 5									
		---	data		7	6	5	4	3	2	1	0	
		---	data[0. .3]	=	BCR [▶ 241]			Binary counter reading					
		---	data[4]	=	IV [▶ 329]	CA [▶ 330]	CY [▶ 330]	1	Sequence		Quality descriptor		
		---	data[5. .IEC87 0_MA X_AS DU_D ATA_ BYTE]	=	Reserved								

**4.4.2.9.2 M\_IT\_TB\_1**

Integrated total with CP56Time2a time tag.

- obj																ASDU object [▶ 483]
---	+ head															Reserved
---	- ident															DATA UNIT IDENTIFIER [▶ 484]
	---	eType	= 0x25 (37)	= M_IT_TB_1												Type identification [▶ 500]
	---	bSQ		= FALSE												Sequence of information objects
	---	nObj		= 1												Number of objects
	---	bT														Test
	---	bPN														Positive/negative confirmation/activation
	---	eCOT														Cause of transmission [▶ 509]
	---	nORG														Originator address
	---	asduA ddr														Common address of asdu
	---	eClass														Fifo priority class [▶ 509]
---	- info															INFORMATION OBJECT [▶ 485]
	---	objAdd r														Information object address
	---	stream														Information element/object data [▶ 485]
		---	length	= 12												
		---	data			7	6	5	4	3	2	1	0			
		---	data[0. .3] =			BCR [▶ 241]		Binary counter reading								
		---	data[4] =			IV [▶ 329]	CA [▶ 330]	CY [▶ 330]	Sequenc e	Quality descriptor						
		---	data[5. .11] =			CP56Time2a [▶ 531]		Seven octets binary time tag								
		---	data[1 2..IEC 870_M AX_A SDU_ DATA_ _BYT _E] =			Reserved										

#### 4.4.2.10 Single command

##### 4.4.2.10.1 C\_SC\_NA\_1

Single command without time tag.

- obj					ASDU object [▶ 483]								
---	+ head				Reserved								
---	- ident				DATA UNIT IDENTIFIER [▶ 484]								
	---	eType	= 0x2D (45)	= C_SC_NA_1	Type identification [▶ 500]								
	---	bSQ		= FALSE	Sequence of information objects								
	---	nObj		= 1	Number of objects								
	---	bT			Test								
	---	bPN			Positive/negative confirmation/activation								
	---	eCOT			Cause of transmission [▶ 509]								
	---	nORG			Originator address								
	---	asduAddr			Common address of asdu								
	---	eClass			Fifo priority class [▶ 509]								
---	- info				INFORMATION OBJECT [▶ 485]								
	---	objAddr			Information object address								
	---	stream			Information element/object data [▶ 485]								
		---	length	= 1									
		---	data		7	6	5	4	3	2	1	0	
		---	data[0]	= S/E [▶ 241]	QU [▶ 524]						0	SC [▶ 520]	SCO = Single command
		---	data[1..IEC870_MAX_ASDU_DATA_BYTE]	=	Reserved								

4.4.2.10.2 C\_SC\_TA\_1

Single command with CP56Time2a time tag.

- obj													ASDU object [▶ 483]
---	+ head												Reserved
---	- ident												DATA UNIT IDENTIFIER [▶ 484]
	---	eType	= 0x3A (58)	= C_SC_TA_1									Type identification [▶ 500]
	---	bSQ		= FALSE									Sequence of information objects
	---	nObj		= 1									Number of objects
	---	bT											Test
	---	bPN											Positive/negative confirmation/activation
	---	eCOT											Cause of transmission [▶ 509]
	---	nORG											Originator address
	---	asduAddr											Common address of asdu
	---	eClass											Fifo priority class [▶ 509]
---	- info												INFORMATION OBJECT [▶ 485]
	---	objAddr											Information object address
	---	stream											Information element/object data [▶ 485]
		---	length	= 8									
		---	data		7	6	5	4	3	2	1	0	
		---	data[0]	= S/E [▶ 241]								0	SC S [▶ 520] SCO = Single command
			---	data[1..7]	= CP56Time2a [▶ 531]								Seven octets binary time tag
			---	data[8..IEC870_MAX_AS_DDU_DATA_BYTE]	=								Reserved

**4.4.2.11 Double command**

**4.4.2.11.1 C\_DC\_NA\_1**

Double command without time tag.

- obj					ASDU object [▶ 483]							
---	+ head				Reserved							
---	- ident				DATA UNIT IDENTIFIER [▶ 484]							
	---	eType	= 0x2E (46)	= C_DC_NA_1	Type identification [▶ 500]							
	---	bSQ		= FALSE	Sequence of information objects							
	---	nObj		= 1	Number of objects							
	---	bT			Test							
	---	bPN			Positive/negative confirmation/activation							
	---	eCOT			Cause of transmission [▶ 509]							
	---	nORG			Originator address							
	---	asduAddr			Common address of asdu							
	---	eClass			Fifo priority class [▶ 509]							
---	- info				INFORMATION OBJECT [▶ 485]							
	---	objAddr			Information object address							
	---	stream			Information element/object data [▶ 485]							
	---	length	= 1									
	---	data			7	6	5	4	3	2	1	0
	---	data[0]	=	S/E [▶ 241]	QU [▶ 524]						DC S [▶ 520]	DCO = Double command
	---	data[1..IEC870_MAX_AS_DDU_DATA_BYTE]	=		Reserved							

**4.4.2.11.2 C\_DC\_TA\_1**

Double command with CP56Time2a time tag.

- obj					ASDU object [▶ 483]									
---	+ head				Reserved									
---	- ident				DATA UNIT IDENTIFIER [▶ 484]									
	---	eType	= 0x3B (59)	= C_DC_TA_1	Type identification [▶ 500]									
	---	bSQ		= FALSE	Sequence of information objects									
	---	nObj		= 1	Number of objects									
	---	bT			Test									
	---	bPN			Positive/negative confirmation/activation									
	---	eCOT			Cause of transmission [▶ 509]									
	---	nORG			Originator address									
	---	asduAddr			Common address of asdu									
	---	eClass			Fifo priority class [▶ 509]									
---	- info				INFORMATION OBJECT [▶ 485]									
	---	objAddr			Information object address									
	---	stream			Information element/object data [▶ 485]									
		---	length	= 8										
		---	data		7	6	5	4	3	2	1	0		
		---	data[0]	= S/E [▶ 241]	QU [▶ 524]						DCS [▶ 520]	DCO = Double command		
											1			
		---	data[1..7]	= CP56Time2a [▶ 531]	Seven octets binary time tag									
		---	data[8..IEC870_MAX_ASDU_DATA_BYTE]	=	Reserved									

#### 4.4.2.12 Regulating step command

##### 4.4.2.12.1 C\_RC\_NA\_1

Regulating step command without time tag.

- obj					ASDU object [▶ 483]								
---	+ head				Reserved								
---	- ident				DATA UNIT IDENTIFIER [▶ 484]								
	---	eType	= 0x2F (47)	= C_RC_NA_1	Type identification [▶ 500]								
	---	bSQ		= FALSE	Sequence of information objects								
	---	nObj		= 1	Number of objects								
	---	bT			Test								
	---	bPN			Positive/negative confirmation/activation								
	---	eCOT			Cause of transmission [▶ 509]								
	---	nORG			Originator address								
	---	asduAddr			Common address of asdu								
	---	eClass			Fifo priority class [▶ 509]								
---	- info				INFORMATION OBJECT [▶ 485]								
	---	objAddr			Information object address								
	---	stream			Information element/object data [▶ 485]								
		---	length	= 1									
		---	data		7	6	5	4	3	2	1	0	
		---	data[0]	=	S/E [▶ 241]	QU [▶ 524]					RC	RCO = Regulating step command	
											S		[▶ 526]
											1		
		---	data[1..IEC870_MAX_ASDU_DATA_BYTE]	=	Reserved								

**4.4.2.12.2 C\_RC\_TA\_1**

Regulating step command with CP56Time2a time tag.

- obj					ASDU object [▶ 483]									
---	+ head				Reserved									
---	- ident				DATA UNIT IDENTIFIER [▶ 484]									
	---	eType	= 0x3C (60)	= C_RC_TA_1	Type identification [▶ 500]									
	---	bSQ		= FALSE	Sequence of information objects									
	---	nObj		= 1	Number of objects									
	---	bT			Test									
	---	bPN			Positive/negative confirmation/activation									
	---	eCOT			Cause of transmission [▶ 509]									
	---	nORG			Originator address									
	---	asduAddr			Common address of asdu									
	---	eClass			Fifo priority class [▶ 509]									
---	- info				INFORMATION OBJECT [▶ 485]									
	---	objAddr			Information object address									
	---	stream			Information element/object data [▶ 485]									
		---	length	= 8										
		---	data		7	6	5	4	3	2	1	0		
		---	data[0]	= S/E [▶ 241]	QU [▶ 524]						RCS [▶ 526]	RCO = Regulating step command		
		---	data[1..7]	= CP56Time2a [▶ 531]	Seven octets binary time tag									
		---	data[8..IEC870_MAX_ASDU_DATA_BYTE]	=	Reserved									

**4.4.2.13 Set-point command, normalized value**

**4.4.2.13.1 C\_SE\_NA\_1**

Set-point command, normalized value without time tag.



- obj					ASDU object [▶ 483]							
---	+ head				Reserved							
---	- ident				DATA UNIT IDENTIFIER [▶ 484]							
	---	eType	= 0x30 (48)	= C_SE_NA_1	Type identification [▶ 500]							
	---	bSQ		= FALSE	Sequence of information objects							
	---	nObj		= 1	Number of objects							
	---	bT			Test							
	---	bPN			Positive/negative confirmation/activation							
	---	eCOT			Cause of transmission [▶ 509]							
	---	nORG			Originator address							
	---	asduA ddr			Common address of asdu							
	---	eClass			Fifo priority class [▶ 509]							
---	- info				INFORMATION OBJECT [▶ 485]							
	---	objAdd r			Information object address							
	---	stream			Information element/object data [▶ 485]							
		---	length	= 3								
		---	data		7	6	5	4	3	2	1	0
		---	data[0. .1]	=	NVA [▶ 241]		Normalized value					
		---	data[2]	=	S/E [▶ 241]	QL [▶ 522]	QOS = Qualifier of set-point command					
		---	data[3. .IEC87 0_MA X_AS DU_D ATA_ BYTE]	=	Reserved							

**4.4.2.13.2 C\_SE\_TA\_1**

Set-point command, normalized value with CP56Time2a time tag.

- obj													ASDU object [▶ 483]
---	+ head												Reserved
---	- ident												DATA UNIT IDENTIFIER [▶ 484]
	---	eType	= 0x3D (61)	= C_SE_TA_1									Type identification [▶ 500]
	---	bSQ		= FALSE									Sequence of information objects
	---	nObj		= 1									Number of objects
	---	bT											Test
	---	bPN											Positive/negative confirmation/activation
	---	eCOT											Cause of transmission [▶ 509]
	---	nORG											Originator address
	---	asduAddr											Common address of asdu
	---	eClass											Fifo priority class [▶ 509]
---	- info												INFORMATION OBJECT [▶ 485]
	---	objAddr											Information object address
	---	stream											Information element/object data [▶ 485]
		---	length	= 10									
		---	data			7	6	5	4	3	2	1	0
		---	data[0..1]	=	NVA [▶ 241]								Normalized value
		---	data[2]	=	S/E [▶ 241]	QL [▶ 522]							QOS = Qualifier of command
		---	data[3..9]	=	CP56Time2a [▶ 531]								Seven octets binary time tag
		---	data[10..IEC 870_M AX_A SDU_DATA_BYT E]	=									Reserved

**4.4.2.14 Set-point command, scaled value**

**4.4.2.14.1 C\_SE\_NB\_1**

Set-point command, scaled value without time tag.

- obj					ASDU object [▶ 483]							
---	+ head				Reserved							
---	- ident				DATA UNIT IDENTIFIER [▶ 484]							
	---	eType	= 0x31 (49)	= C_SE_NB_1	Type identification [▶ 500]							
	---	bSQ		= FALSE	Sequence of information objects							
	---	nObj		= 1	Number of objects							
	---	bT			Test							
	---	bPN			Positive/negative confirmation/activation							
	---	eCOT			Cause of transmission [▶ 509]							
	---	nORG			Originator address							
	---	asduAddr			Common address of asdu							
	---	eClass			Fifo priority class [▶ 509]							
---	- info				INFORMATION OBJECT [▶ 485]							
	---	objAddr			Information object address							
	---	stream			Information element/object data [▶ 485]							
		---	length	= 3								
		---	data		7	6	5	4	3	2	1	0
		---	data[0..1]	=	SVA [▶ 241]		Scaled value					
		---	data[2]	=	S/E [▶ 241]	QL [▶ 522]	QOS = Qualifier of set-point command					
		---	data[3..IEC870_MAX_ASUDU_DATA_BYTE]	=	Reserved							

**4.4.2.14.2 C\_SE\_TB\_1**

Set-point command, scaled value with CP56Time2a time tag.

- obj															ASDU object [▶ 483]
---	+ head														Reserved
---	- ident														DATA UNIT IDENTIFIER [▶ 484]
	---	eType	= 0x3E (62)	= C_SE_TB_1											Type identification [▶ 500]
	---	bSQ		= FALSE											Sequence of information objects
	---	nObj		= 1											Number of objects
	---	bT													Test
	---	bPN													Positive/negative confirmation/activation
	---	eCOT													Cause of transmission [▶ 509]
	---	nORG													Originator address
	---	asduAddr													Common address of asdu
	---	eClass													Fifo priority class [▶ 509]
---	- info														INFORMATION OBJECT [▶ 485]
	---	objAddr													Information object address
	---	stream													Information element/object data [▶ 485]
		---	length	= 10											
		---	data			7	6	5	4	3	2	1	0		
		---	data[0..1]	=	SVA [▶ 241]										Scaled value
		---	data[2]	=	S/E [▶ 241]	QL [▶ 522]									QOS = Qualifier of command
		---	data[3..9]	=	CP56Time2a [▶ 531]										Seven octets binary time tag
		---	data[10..IEC 870_M AX_A SDU_DATA_BYT E]	=											Reserved

**4.4.2.15 Set-point command, short floating value**

**4.4.2.15.1 C\_SE\_NC\_1**

Set-point command, short floating point value without time tag.

- obj																			ASDU object [▶ 483]
---	+ head																		Reserved
---	- ident																		DATA UNIT IDENTIFIER [▶ 484]
	---	eType	= 0x32 (50)	= C_SE_NC_1															Type identification [▶ 500]
	---	bSQ		= FALSE															Sequence of information objects
	---	nObj		= 1															Number of objects
	---	bT																	Test
	---	bPN																	Positive/negative confirmation/activation
	---	eCOT																	Cause of transmission [▶ 509]
	---	nORG																	Originator address
	---	asduA ddr																	Common address of asdu
	---	eClass																	Fifo priority class [▶ 509]
---	- info																		INFORMATION OBJECT [▶ 485]
	---	objAdd r																	Information object address
	---	stream																	Information element/object data [▶ 485]
		---	length	= 5															
		---	data			7	6	5	4	3	2	1	0						
		---	data[0. .3]	=	R32 [▶ 241]														Short floating point value
		---	data[4]	=	S/E [▶ 241]	Q_L [▶ 522]													QOS = Qualifier of set-point command
		---	data[5. .IEC87 0_MA X_AS DU_D ATA_ BYTE]	=															Reserved

**4.4.2.15.2 C\_SE\_TC\_1**

Set-point command, short floating point value with CP56Time2a time tag.

- obj															ASDU object [▶ 483]
---	+ head														Reserved
---	- ident														DATA UNIT IDENTIFIER [▶ 484]
	---	eType	= 0x3F (63)	= C_SE_TC_1											Type identification [▶ 500]
	---	bSQ		= FALSE											Sequence of information objects
	---	nObj		= 1											Number of objects
	---	bT													Test
	---	bPN													Positive/negative confirmation/activation
	---	eCOT													Cause of transmission [▶ 509]
	---	nORG													Originator address
	---	asduA ddr													Common address of asdu
	---	eClass													Fifo priority class [▶ 509]
---	- info														INFORMATION OBJECT [▶ 485]
	---	objAdd r													Information object address
	---	stream													Information element/object data [▶ 485]
		---	length	= 12											
		---	data			7	6	5	4	3	2	1	0		
		---	data[0. .3] =		R32 [▶ 241]										Short floating point value
		---	data[4] =		S/E [▶ 241]	QL [▶ 522]									QOS = Qualifier of command
		---	data[5. .11] =		CP56Time2a [▶ 531]										Seven octets binary time tag
		---	data[1 2..IEC 870_M AX_A SDU_ DATA _BYT E] =												Reserved

#### 4.4.2.16 Bitstring command

##### 4.4.2.16.1 C\_BO\_NA\_1

Bitstring of 32 bits without time tag.

- obj					ASDU object [▶ 483]							
---	+ head				Reserved							
---	- ident				DATA UNIT IDENTIFIER [▶ 484]							
	---	eType	= 0x33 (51)	= C_BO_NA_1	Type identification [▶ 500]							
	---	bSQ		= FALSE	Sequence of information objects							
	---	nObj		= 1	Number of objects							
	---	bT			Test							
	---	bPN			Positive/negative confirmation/activation							
	---	eCOT			Cause of transmission [▶ 509]							
	---	nORG			Originator address							
	---	asduAddr			Common address of asdu							
	---	eClass			Fifo priority class [▶ 509]							
---	- info				INFORMATION OBJECT [▶ 485]							
	---	objAddr			Information object address							
	---	stream			Information element/object data [▶ 485]							
		---	length	= 4								
		---	data		7	6	5	4	3	2	1	0
		---	data[0..3]	=	BSI [▶ 241]		Binary state information					
		---	data[4..IEC870_MAX_AS_DDU_DATA_BYTE]	=	Reserved							

**4.4.2.16.2 C\_BO\_TA\_1**

Bitstring of 32 bits with CP56Time2a time tag.

- obj						ASDU object [▶ 483]								
---	+ head					Reserved								
---	- ident					DATA UNIT IDENTIFIER [▶ 484]								
	---	eType	= 0x40 (64)	= C_BO_TA_1		Type identification [▶ 500]								
	---	bSQ		= FALSE		Sequence of information objects								
	---	nObj		= 1		Number of objects								
	---	bT				Test								
	---	bPN				Positive/negative confirmation/activation								
	---	eCOT				Cause of transmission [▶ 509]								
	---	nORG				Originator address								
	---	asduA ddr				Common address of asdu								
	---	eClass				Fifo priority class [▶ 509]								
---	- info					INFORMATION OBJECT [▶ 485]								
	---	objAdd r				Information object address								
	---	stream				Information element/object data [▶ 485]								
		---	length	= 11										
		---	data			7	6	5	4	3	2	1	0	
		---	data[0. .3]	=	BSI [▶ 241]	Binary state information								
		---	data[4. .10]	=	CP56Time2a [▶ 531]	Seven octets binary time tag								
		---	data[1 1..IEC 870_M AX_A SDU_ DATA_ BYT E]	=		Reserved								

#### 4.4.2.17 Test command

##### 4.4.2.17.1 C\_TS\_TA\_1

Test command with CP56Time2a time tag.



- obj					ASDU object [▶ 483]							
---	+ head				Reserved							
---	- ident				DATA UNIT IDENTIFIER [▶ 484]							
	---	eType	= C_TS_TA_1		Type identification [▶ 500]							
	---	bSQ	= FALSE		Sequence of information objects							
	---	nObj	= 1		Number of objects							
	---	bT			Test							
	---	bPN			Positive/negative confirmation/activation							
	---	eCOT			Cause of transmission [▶ 509]							
	---	nORG			Originator address							
	---	asduA ddr			Common address of asdu							
	---	eClass			Fifo priority class [▶ 509]							
---	- info				INFORMATION OBJECT [▶ 485]							
	---	objAdd r			Information object address							
	---	stream			Information element/object data [▶ 485]							
		---	length = 9									
		---	data		7	6	5	4	3	2	1	0
		---	data[0. .1] =	TSC [▶ 241]	Test counter							
		---	data[2. .8] =	CP56Time2a [▶ 531]	Seven octets binary time tag							
		---	data[9. .IEC87 0_MA X_AS DU_D ATA_ BYTE] =		Reserved							

**4.4.2.18 System information in monitor direction**

**4.4.2.18.1 M\_EI\_NA\_1**

End of initialization.

- obj					ASDU object [▶ 483]							
---	+ head				Reserved							
---	- ident				DATA UNIT IDENTIFIER [▶ 484]							
	---	eType	= 0x46 (70)	= M_EI_NA_1	Type identification [▶ 500]							
	---	bSQ		= FALSE	Sequence of information objects							
	---	nObj		= 1	Number of objects							
	---	bT			Test							
	---	bPN			Positive/negative confirmation/activation							
	---	eCOT			Cause of transmission [▶ 509]							
	---	nORG			Originator address							
	---	asduA ddr			Common address of asdu							
	---	eClass			Fifo priority class [▶ 509]							
---	- info				INFORMATION OBJECT [▶ 485]							
	---	objAdd r		= 0	Information object address							
	---	stream			Information element/object data [▶ 485]							
		---	length	= 1								
		---	data		7	6	5	4	3	2	1	0
		---	data[0]	=	LPC [▶ 240]	COI [▶ 521]	Cause of initialization					
		---	data[1. .IEC87 0_MA X_AS DU_D ATA_ BYTE]	=	Reserved							

#### 4.4.2.19 System information in control direction

##### 4.4.2.19.1 C\_CS\_NA\_1

Clock synchronisation command

- obj						ASDU object [▶ 483]								
---	+ head					Reserved								
---	- ident					DATA UNIT IDENTIFIER [▶ 484]								
	---	eType	= 0x67 (103)	= C_CS_NA_1		Type identification [▶ 500]								
	---	bSQ		= FALSE		Sequence of information objects								
	---	nObj		= 1		Number of objects								
	---	bT				Test								
	---	bPN				Positive/negative confirmation/activation								
	---	eCOT				Cause of transmission [▶ 509]								
	---	nORG				Originator address								
	---	asduA ddr				Common address of asdu								
	---	eClass				Fifo priority class [▶ 509]								
---	- info					INFORMATION OBJECT [▶ 485]								
	---	objAdd r		= 0		Information object address								
	---	stream				Information element/object data [▶ 485]								
		---	length	= 7										
		---	data			7	6	5	4	3	2	1	0	
		---	data[0. .6] =		CP56Time2a [▶ 531]	Seven octets binary time tag								
		---	data[7. .IEC87 0_MA X_AS DU_D ATA_ BYTE] =			Reserved								

**4.4.2.19.2 C\_IC\_NA\_1**

Interrogation command.

- obj					ASDU object [▶ 483]								
---	+ head				Reserved								
---	- ident				DATA UNIT IDENTIFIER [▶ 484]								
	---	eType	= 0x64 (100)	= C_IC_NA_1	Type identification [▶ 500]								
	---	bSQ		= FALSE	Sequence of information objects								
	---	nObj		= 1	Number of objects								
	---	bT			Test								
	---	bPN			Positive/negative confirmation/activation								
	---	eCOT			Cause of transmission [▶ 509]								
	---	nORG			Originator address								
	---	asduAddr			Common address of asdu								
	---	eClass			Fifo priority class [▶ 509]								
---	- info				INFORMATION OBJECT [▶ 485]								
	---	objAddr		= 0	Information object address								
	---	stream			Information element/object data [▶ 485]								
		---	length	= 1									
		---	data		7	6	5	4	3	2	1	0	
		---	data[0]	=	QOI [▶ 521]								
		---	data[1..IEC870_MAX_ASDU_DATA_BYTE]	=	Reserved								

**4.4.2.19.3 C\_CI\_NA\_1**

Counter interrogation command.

- obj						ASDU object [▶ 483]
---	+ head					Reserved
---	- ident					DATA UNIT IDENTIFIER [▶ 484]
	---	eType	= 0x65 (101)	= C_CI_NA_1		Type identification [▶ 500]
	---	bSQ		= FALSE		Sequence of information objects
	---	nObj		= 1		Number of objects
	---	bT				Test
	---	bPN				Positive/negative confirmation/activation
	---	eCOT				Cause of transmission [▶ 509]
	---	nORG				Originator address
	---	asduAddr				Common address of asdu
	---	eClass				Fifo priority class [▶ 509]
---	- info					INFORMATION OBJECT [▶ 485]
	---	objAddr		= 0		Information object address
	---	stream				Information element/object data [▶ 485]
		---	length	= 1		
		---	data		7 6 5 4 3 2 1 0	
		---	data[0]	=	FRZ [▶ 522]	RQCC = Qualifier of counter interrogation
		---	data[1..IEC870_MAX_AS_DUDATA_BYTE]	=		Reserved

**4.4.2.19.4 C\_RP\_NA\_1**

Reset process command.

- obj					ASDU object [▶ 483]								
---	+ head				Reserved								
---	- ident				DATA UNIT IDENTIFIER [▶ 484]								
	---	eType	= 0x69 (105)	= C_RP_NA_1	Type identification [▶ 500]								
	---	bSQ		= FALSE	Sequence of information objects								
	---	nObj		= 1	Number of objects								
	---	bT			Test								
	---	bPN			Positive/negative confirmation/activation								
	---	eCOT			Cause of transmission [▶ 509]								
	---	nORG			Originator address								
	---	asduA ddr			Common address of asdu								
	---	eClass			Fifo priority class [▶ 509]								
---	- info				INFORMATION OBJECT [▶ 485]								
	---	objAdd r		= 0	Information object address								
	---	stream			Information element/object data [▶ 485]								
		---	length	= 1									
		---	data		7	6	5	4	3	2	1	0	
		---	data[0]	=	QRP [▶ 523]								Qualifier of reset process
		---	data[1..										Reserved
			.IEC87										
			0_MA										
			X_AS										
			DU_D										
			ATA_										
			BYTE]										
			=										

**4.4.2.19.5 C\_RD\_NA\_1**

Read command.

- obj					ASDU object [▶ 483]							
---	+ head				Reserved							
---	- ident				DATA UNIT IDENTIFIER [▶ 484]							
	---	eType	= 0x66 (102)	= C_RD_NA_1	Type identification [▶ 500]							
	---	bSQ		= FALSE	Sequence of information objects							
	---	nObj		= 1	Number of objects							
	---	bT			Test							
	---	bPN			Positive/negative confirmation/activation							
	---	eCOT			Cause of transmission [▶ 509]							
	---	nORG			Originator address							
	---	asduA ddr			Common address of asdu							
	---	eClass			Fifo priority class [▶ 509]							
---	- info				INFORMATION OBJECT [▶ 485]							
	---	objAdd r			Information object address							
	---	stream			Information element/object data [▶ 485]							
		---	length	= 0								
		---	data		7	6	5	4	3	2	1	0
		---	data[0. .IEC87 0_MA X_AS DU_D ATA_ BYTE] =		Reserved							

**4.4.2.20 Protection equipment information**

**4.4.2.20.1 M\_EP\_TD\_1**

Event of protection equipment with CP56Time2a time tag.

- obj					ASDU object [▶ 483]								
---	+ head				Reserved								
---	- ident				DATA UNIT IDENTIFIER [▶ 484]								
	---	eType	= 0x26 (38)	= M_EP_TD_1	Type identification [▶ 500]								
	---	bSQ		= FALSE	Sequence of information objects								
	---	nObj		= 1	Number of objects								
	---	bT			Test								
	---	bPN			Positive/negative confirmation/activation								
	---	eCOT			Cause of transmission [▶ 509]								
	---	nORG			Originator address								
	---	asduAddr			Common address of asdu								
	---	eClass			Fifo priority class [▶ 509]								
---	- info				INFORMATION OBJECT [▶ 485]								
	---	objAddr			Information object address								
	---	stream			Information element/object data [▶ 485]								
		---	length	= 10									
		---	data		7	6	5	4	3	2	1	0	
		---	data[0]	=	<u>IV</u> [▶ 329]	<u>NT</u> [▶ 329]	<u>SB</u> [▶ 329]	<u>BL</u> [▶ 329]	<u>EI</u> [▶ 330]	0	<u>ES</u> [▶ 524]	SEP = Single event of protection equipment	
		---	data[1..2]	=	<u>CP16Time2a</u> [▶ 534]		Elapsed time, two octets binary time						
		---	data[3..9]	=	<u>CP56Time2a</u> [▶ 531]		Sevent octets binary time tag						
		---	data[10..IEC 870_M AX_A SDU_DATA_BYT E]	=	Reserved								

#### 4.4.2.20.2 M\_EP\_TE\_1

Packed start events of protection equipment with CP56Time2a time tag.



- obj					ASDU object [▶ 483]								
---	+ head				Reserved								
---	- ident				DATA UNIT IDENTIFIER [▶ 484]								
	---	eType	= 0x27 (39)	= M_EP_TE_1	Type identification [▶ 500]								
	---	bSQ		= FALSE	Sequence of information objects								
	---	nObj		= 1	Number of objects								
	---	bT			Test								
	---	bPN			Positive/negative confirmation/activation								
	---	eCOT			Cause of transmission [▶ 509]								
	---	nORG			Originator address								
	---	asduAddr			Common address of asdu								
	---	eClass			Fifo priority class [▶ 509]								
---	- info				INFORMATION OBJECT [▶ 485]								
	---	objAddr			Information object address								
	---	stream			Information element/object data [▶ 485]								
		---	length	= 11									
		---	data		7	6	5	4	3	2	1	0	
		---	data[0]	=	0	0	SRD	SIE	SL3	SL2	SL1	GS	SEP = Start events of protection equipment
		---	data[1]	=	<u>IV</u> [▶ 329]	<u>NT</u> [▶ 329]	<u>SB</u> [▶ 329]	<u>BL</u> [▶ 329]	<u>EI</u> [▶ 330]	0	0	0	QDP = Quality descriptor for events of protection equipment
		---	data[2..3]	=	CP16Time2a [▶ 534]								Relay duration time, two octets binary time
		---	data[4..10]	=	CP56Time2a [▶ 531]								Seven octets binary time tag
		---	data[11..16]	=	IEC 870_M AX_A SDU_DATA_BYTE								Reserved

**4.4.2.20.3 M\_EP\_TF\_1**

Packed output circuit information of protection equipment with CP56Time2a time tag.

- obj					ASDU object [▶ 483]								
---	+ head				Reserved								
---	- ident				DATA UNIT IDENTIFIER [▶ 484]								
	---	eType	= 0x28 (40)	= M_EP_TF_1	Type identification [▶ 500]								
	---	bSQ		= FALSE	Sequence of information objects								
	---	nObj		= 1	Number of objects								
	---	bT			Test								
	---	bPN			Positive/negative confirmation/activation								
	---	eCOT			Cause of transmission [▶ 509]								
	---	nORG			Originator address								
	---	asduAddr			Common address of asdu								
	---	eClass			Fifo priority class [▶ 509]								
---	- info				INFORMATION OBJECT [▶ 485]								
	---	objAddr			Information object address								
	---	stream			Information element/object data [▶ 485]								
		---	length	= 11									
		---	data		7	6	5	4	3	2	1	0	
		---	data[0]	=	0	0	0	0	CL3	CL2	CL1	G C	OCI = Output circuit information of protection equipment
		---	data[1]	=	<u>IV</u> [▶ 329]	<u>NT</u> [▶ 329]	<u>SB</u> [▶ 329]	<u>BL</u> [▶ 329]	<u>EI</u> [▶ 330]	0	0	0	QDP = Quality descriptor for events of protection equipment
		---	data[2..3]	=	CP16Time2a [▶ 534]			Relay operating time, two octets binary time					
		---	data[4..10]	=	CP56Time2a [▶ 531]			Seven octets binary time tag					
		---	data[11..IEC870_MAX_ASDU_DATA_BYTES]	=	Reserved								

#### 4.4.2.20.4 M\_PS\_NA\_1

Packed single point information with status change detection.

- obj													ASDU object [▶ 483]	
---	+ head												Reserved	
---	- ident												DATA UNIT IDENTIFIER [▶ 484]	
	---	eType	= 0x14 (20)	= M_PS_NA_1									Type identification [▶ 500]	
	---	bSQ		= FALSE									Sequence of information objects	
	---	nObj		= 1									Number of objects	
	---	bT											Test	
	---	bPN											Positive/negative confirmation/activation	
	---	eCOT											Cause of transmission [▶ 509]	
	---	nORG											Originator address	
	---	asduA ddr											Common address of asdu	
	---	eClass											Fifo priority class [▶ 509]	
---	- info												INFORMATION OBJECT [▶ 485]	
	---	objAdd r											Information object address	
	---	stream											Information element/object data [▶ 485]	
		---	length	= 5										
		---	data			7	6	5	4	3	2	1	0	
		---	data[0. .3]	=	SCD				Status and status change detection (32 bit)					
		---	data[4]	=	<u>IV</u> [▶ 329]	<u>NT</u> [▶ 329]	<u>SB</u> [▶ 329]	<u>BL</u> [▶ 329]				<u>OV</u> [▶ 329]	QDS: Quality descriptor	
		---	data[5. .IEC87 0_MA X_AS DU_D ATA_ BYTE]	=	Reserved									

4.4.2.21 Parameter loading/activation

4.4.2.21.1 P\_ME\_NA\_1

Parameter of measured value, normalized value.

- obj					ASDU object [▶ 483]							
---	+ head				Reserved							
---	- ident				DATA UNIT IDENTIFIER [▶ 484]							
	---	eType	= 0x6E (110)	= P_ME_NA_1	Type identification [▶ 500]							
	---	bSQ		= FALSE	Sequence of information objects							
	---	nObj		= 1	Number of objects							
	---	bT			Test							
	---	bPN			Positive/negative confirmation/activation							
	---	eCOT			Cause of transmission [▶ 509]							
	---	nORG			Originator address							
	---	asduAddr			Common address of asdu							
	---	eClass			Fifo priority class [▶ 509]							
---	- info				INFORMATION OBJECT [▶ 485]							
	---	objAddr			Information object address							
	---	stream			Information element/object data [▶ 485]							
	---	length	= 3									
	---	data			7	6	5	4	3	2	1	0
	---	data[0..1]	=	NVA [▶ 241]	Normalized value							
	---	data[2]	=	LPC [▶ 240]	POP	QPM: Qualifier of parameter of measured value						
	---	data[3..IEC870_MAX_ASDU_DATA_BYTE]	=		Reserved							

**4.4.2.21.2 P\_ME\_NB\_1**

Parameter of measured value, scaled value.

- obj															ASDU object [▶ 483]
---	+ head														Reserved
---	- ident														DATA UNIT IDENTIFIER [▶ 484]
	---	eType	= 0x6F (111)	= P_ME_NB_1											Type identification [▶ 500]
	---	bSQ		= FALSE											Sequence of information objects
	---	nObj		= 1											Number of objects
	---	bT													Test
	---	bPN													Positive/negative confirmation/activation
	---	eCOT													Cause of transmission [▶ 509]
	---	nORG													Originator address
	---	asduAddr													Common address of asdu
	---	eClass													Fifo priority class [▶ 509]
---	- info														INFORMATION OBJECT [▶ 485]
	---	objAddr													Information object address
	---	stream													Information element/object data [▶ 485]
		---	length	= 3											
		---	data			7	6	5	4	3	2	1	0		
		---	data[0..1]	=	SVA [▶ 241]										Scaled value
		---	data[2]	=	LPC [▶ 240]	POP									QPM: Qualifier of parameter of measured value
															525
		---	data[3..IEC870_MAX_DATA_BYTE]	=											Reserved

**4.4.2.21.3 P\_ME\_NC\_1**

Parameter of measured value, short floating point value.

- obj														ASDU object [▶ 483]
---	+ head													Reserved
---	- ident													DATA UNIT IDENTIFIER [▶ 484]
	---	eType	= 0x70 (112)	= P_ME_NC_1										Type identification [▶ 500]
	---	bSQ		= FALSE										Sequence of information objects
	---	nObj		= 1										Number of objects
	---	bT												Test
	---	bPN												Positive/negative confirmation/activation
	---	eCOT												Cause of transmission [▶ 509]
	---	nORG												Originator address
	---	asduA ddr												Common address of asdu
	---	eClass												Fifo priority class [▶ 509]
---	- info													INFORMATION OBJECT [▶ 485]
	---	objAdd r												Information object address
	---	stream												Information element/object data [▶ 485]
		---	length	= 5										
		---	data			7	6	5	4	3	2	1	0	
		---	data[0. .3]	=	R32 [▶ 241]									Short floating point value
		---	data[4]	=	LPC [▶ 240]	POP								QPM: Qualifier of parameter of measured value
		---	data[5. .IEC87 0_MA X_AS DU_D ATA_ BYTE]	=										Reserved

**4.4.2.21.4 P\_AC\_NA\_1**

Parameter activation.

- obj																				ASDU object [▶ 483]
---	+ head																			Reserved
---	- ident																			DATA UNIT IDENTIFIER [▶ 484]
	---	eType	= 0x71 (113)	= P_AC_NA_1																Type identification [▶ 500]
	---	bSQ		= FALSE																Sequence of information objects
	---	nObj		= 1																Number of objects
	---	bT																		Test
	---	bPN																		Positive/negative confirmation/activation
	---	eCOT																		Cause of transmission [▶ 509]
	---	nORG																		Originator address
	---	asduA addr																		Common address of asdu
	---	eClass																		Fifo priority class [▶ 509]
---	- info																			INFORMATION OBJECT [▶ 485]
	---	objAdd r		= 0																Information object address
	---	stream																		Information element/object data [▶ 485]
		---	length	= 1																
		---	data			7	6	5	4	3	2	1	0							
		---	data[0]	=																QPA [▶ 525]
		---	data[1..																	Reserved
			.IEC87																	
			0_MA																	
			X_AS																	
			DU_D																	
			ATA_																	
			BYTE]																	
			=																	

**4.4.2.22 Informationselemente**

**4.4.2.22.1 LPC**

Local parameter change flag.

- <0> = No change;
- <1> = Changed;

**4.4.2.22.2 VTI**

Value with transient state indication (8 bits).

Transient state (bit 7):

- <0> = equipment is not in transient state;
- <1> = equipment is in transient state;

Value (bits 0..6) = <-64..63>;

**4.4.2.22.3 BSI**

Bitstring of 32 bits.

**4.4.2.22.4 NVA**

Normalized value.

**4.4.2.22.5 SVA**

Scaled value.

**4.4.2.22.6 R32**

Short floating point value.

**4.4.2.22.7 BCR**

Binary counter reading.

**4.4.2.22.8 S/E**

Select/execute state.

- <0> = Ausführen (execute);
- <1> = Anwählen (select);

**4.4.2.22.9 FBP**

Fixed test pattern.

**4.4.2.22.10 TSC**

Test command counter.



## 5 SPS-Bibliotheken

Mit den SPS-Funktionen und Funktionsbausteinen können Zentralstationen und/oder Unterstationen (Master/Slaves) nach der IEC60870-5-101/102/103/104 Norm in der TwinCAT SPS realisiert werden. Die SPS-Bibliothek verfügt über zwei Software-Schnittstellen. Die Endapplikation setzt auf einer dieser Schnittstellen auf. Die Wahl der Schnittstelle hängt von den Anforderungen an die Endapplikation ab.

### 5.1 Funktionsbausteine

#### 5.1.1 High Level

##### 5.1.1.1 FB\_IEC870\_5\_101Master

FB_IEC870_5_101Master	
hSerial	T_HSERIALCTRL
hTable	T_HAODBTable
protPara	ST_IEC870_5_101ProtocolParams
sysPara	ST_IEC870_5_101SystemParams
acqPara	ST_IEC870_5_101AcquisitionParams
pAOEntries	POINTER TO ARRAY[0..IEC870_ANYSIZE_ARRAY] OF ST_IEC870_5_101AODBEntry
cbAOEntries	UDINT
pInputs	PVOID
cbInputs	UDINT
pOutputs	PVOID
cbOutputs	UDINT
pMemory	PVOID
cbMemory	UDINT
pData	PVOID
cbData	UDINT
bEnable	BOOL

Mit einer Instanz des Funktionsbausteins FB\_IEC870\_5\_101Master kann in der TwinCAT SPS eine IEC 60870-5-101 Zentralstation (Master, High-Level) implementiert werden. Pro Instanz des Funktionsbausteins wird eine Verbindung zum Slave aufgebaut. Im Normalfall wird der Datenaustausch automatisch gestartet, nachdem die Verbindung hergestellt wurde. Standardmäßig ist der Funktionsbaustein auch so konfiguriert.

#### VAR\_IN\_OUT

```
VAR_IN_OUT
  hSerial : T_HSERIALCTRL;
  hTable  : T_HAODBTable;
END_VAR
```

**hSerial:** Verbindungs-Handle [► 530] zum FB\_IEC870\_SerialLineCtrl [► 417]-Funktionsbaustein. Über diese Variable werden mit dem FB\_IEC870\_SerialLineCtrl-Funktionsbaustein die zu sendenden und empfangenen Daten ausgetauscht.

**hTable:** Applikationsobjekt-Datenbankhandle [► 531] (Hash-Tabellenhandle). Das Tabellenhandle muss vor der Benutzung einmalig mit der Funktion F\_iecCreateTableHnd [► 452] initialisiert werden.

#### VAR\_INPUT

```
VAR_INPUT
  protPara      : ST_IEC870_5_101ProtocolParams := ( eType := eIEC870_101_MASTER );
  sysPara       : ST_IEC870_5_101SystemParams := ( bEndOfInit := FALSE );
  acqPara       : ST_IEC870_5_101AcquisitionParams;
  pAOEntries    : POINTER TO ARRAY[0..IEC870_ANYSIZE_ARRAY] OF ST_IEC870_5_101AODBEntry := 0;
  cbAOEntries   : UDINT := 0;
  pInputs       : PVOID := 0;
  cbInputs      : UDINT := 0;
  pOutputs      : PVOID := 0;
  cbOutputs     : UDINT := 0;
  pMemory       : PVOID := 0;
  cbMemory      : UDINT := 0;
  pData         : PVOID := 0;
  cbData        : UDINT := 0;
  bEnable       : BOOL := TRUE;
END_VAR
```

**protPara:** IEC 60870-5-101-Protocolparameter [► 538].

**sysPara:** [Systemparameter](#) [► 487].

**acqPara:** [Parameter für die zyklische Datenerfassung](#) [► 491].

**pAOEntries:** [Adresse der ersten Applikationsobjekt-Datenbankvariablen](#) [► 482].

**cbAOEntries:** Bytegröße der Applikationsobjekt-Datenbankvariablen.

**pInputs:** Adresse des SPS-Prozessdatenbereichs der Eingänge.

**cbInputs:** Bytegröße des SPS-Prozessdatenbereichs der Eingänge.

**pOutputs:** Adresse des SPS-Prozessdatenbereichs der Ausgänge.

**cbOutputs:** Bytegröße des SPS-Prozessdatenbereichs der Ausgänge.

**pMamory:** Adresse des SPS-Prozessdatenbereichs der Merker.

**cbMamory:** Bytegröße des SPS-Prozessdatenbereichs der Merker.

**pData:** Adresse des SPS-Datenbereichs.

**cbData:** Bytegröße des SPS-Datenbereichs.

**bEnable:** Aktiviert/Deaktiviert den Funktionsbaustein (Kommunikation und Verbindungen).

Die Adressen können mit dem ADR- und die Bytegrößen mit dem SIZEOF-Operator ermittelt werden.

## VAR\_OUTPUT

```
VAR_OUTPUT
  system : ST_IEC870_5_101ExSystemInterface;
  eState : E_IEC870_5_101SerialLinkState := eSERIALLINK_DISCONNECTED;
END_VAR
```

**system:** [System-Interface](#) [► 480]. Diese Variable dient anderen IEC-Applikationsfunktionen als Kommunikationsschnittstelle zum IEC-Gerät (hier: Zentralstation).

- Membervariable *system.device* wird z.B. von der [F\\_iecSetAOQuality](#) [► 437]-Funktion als VAR\_IN\_OUT-Parameter erwartet.
- Membervariable *system.device.errors* ist ein Gerätefehler-Fifo. Die registrierten Fehler können von der SPS-Applikation ausgelesen und ausgewertet werden.

**eState:** [Status der Verbindung](#) [► 528] zum Slave.

### Beispiele (High-Level Interface):

- [IEC 60870-5-101 minimale Zentralstation \(master\)](#) [► 565];
- [IEC 60870-5-101 komplexe Zentralstation \(master\)](#) [► 566];

### Weitere Beispiele:

Im folgenden Beispiel wird der Gerätefehler-Fifo zyklisch ausgelesen und die registrierten Fehler ins TwinCAT XAE->"Error List"-Fenster geschrieben.

```
...
fbClient : FB_IEC870_5_101Master; (* IEC 60870-5-101 control station function block instance *)
...
...
REPEAT
  fbClient.system.device.errors.RemoveError();
  IF fbClient.system.device.errors.bOk THEN
    ADSLOGSTR( ADSLOG_MSGTYPE_ERROR OR ADSLOG_MSGTYPE_LOG, 'IEC 60870-5-101 master communication
error: 0x%s',
              DWORD_TO_HEXSTR( fbClient.system.device.errors.getError.nErrId, 8, FALSE) );
  END_IF
UNTIL NOT fbClient.system.device.errors.bOk
END_REPEAT
...
```

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

5.1.1.2 FB\_IEC870\_5\_101Slave

```

FB_IEC870_5_101Slave
-----
hSerial T_SERIALCTRL
protPara ST_IEC870_5_101ProtocolParams
sysPara ST_IEC870_5_101SystemParams
pAOEntries POINTER TO ARRAY[0..IEC870_ANYSIZE_ARRAY] OF ST_IEC870_5_101AODBEntry
cbAOEntries UDINT
pInputs PVOID
cbInputs UDINT
pOutputs PVOID
cbOutputs UDINT
pMemory PVOID
cbMemory UDINT
pData PVOID
cbData UDINT
bEnable BOOL
    
```

Mit einer Instanz des Funktionsbausteins FB\_IEC870\_5\_101Slave kann in der TwinCAT SPS eine IEC 60870-5-101 Unterstation (Slave, High-Level) implementiert werden. Im Normalfall wird der Datenaustausch automatisch gestartet, nachdem die Verbindung hergestellt wurde. Standardmäßig ist der Funktionsbaustein auch so konfiguriert.

VAR\_IN\_OUT

```

VAR_IN_OUT
  hSerial      : T_SERIALCTRL;
END_VAR
    
```

**hSerial:** Verbindungs-Handle [▶ 530] zum FB\_IEC870\_SerialLineCtrl [▶ 417]-Funktionsbaustein. Über diese Variable werden mit dem FB\_IEC870\_SerialLineCtrl-Funktionsbaustein die zu sendenden und empfangenen Daten ausgetauscht.

VAR\_INPUT

```

VAR_INPUT
  protPara      : ST_IEC870_5_101ProtocolParams;
  sysPara       : ST_IEC870_5_101SystemParams;
  pAOEntries    : POINTER TO ARRAY[0..IEC870_ANYSIZE_ARRAY] OF ST_IEC870_5_101AODBEntry := 0;
  cbAOEntries   : UDINT := 0;
  pInputs       : PVOID := 0;
  cbInputs      : UDINT := 0;
  pOutputs      : PVOID := 0;
  cbOutputs     : UDINT := 0;
  pMemory       : PVOID := 0;
  cbMemory      : UDINT := 0;
  pData         : PVOID := 0;
  cbData        : UDINT := 0;
  bEnable       : BOOL := TRUE;
END_VAR
    
```

**protPara:** IEC 60870-5-101-Protokollparameter [▶ 538].

**sysPara:** Systemparameter [▶ 487].

**pAOEntries:** Adresse der ersten Applikationsobjekt-Datenbankvariablen [▶ 482].

**cbAOEntries:** Bytegröße der Applikationsobjekt-Datenbankvariablen.

**plnputs:** Adresse des SPS-Prozessdatenbereichs der Eingänge.

**cbInputs:** Bytegröße des SPS-Prozessdatenbereichs der Eingänge.

**pOutputs:** Adresse des SPS-Prozessdatenbereichs der Ausgänge.

**cbOutputs:** Bytegröße des SPS-Prozessdatenbereichs der Ausgänge.

**pMemory:** Adresse des SPS-Prozessdatenbereichs der Merker.

**cbMemory:** Bytegröße des SPS-Prozessdatenbereichs der Merker.

**pData:** Adresse des SPS-Datenbereichs.

**cbData:** Bytegröße des SPS-Datenbereichs.

**bEnable:** Aktiviert/Deaktiviert den Funktionsbaustein (Kommunikation und Verbindung).

Die Adressen können mit dem ADR- und die Bytegrößen mit dem SIZEOF-Operator ermittelt werden.

## VAR\_OUTPUT

```
VAR_OUTPUT
  system      : ST_IEC870_5_101SystemInterface;
  eState      : E_IEC870_5_101SerialLinkState := eSERIALLINK_DISCONNECTED;
END_VAR
```

**system:** [System-Interface \[► 480\]](#). Diese Variable dient anderen SPS Funktionen oder Funktionsbausteinen als Kommunikationsschnittstelle zum IEC-Gerät (hier: Unterstation).

- Membervariable *system.device* wird z.B. von der [F\\_iecSetAOQuality \[► 437\]](#)-Funktion als VAR\_IN\_OUT-Parameter erwartet.
- Membervariable *system.device.errors* ist ein Gerätefehler-Fifo. Die registrierten Fehler können von der SPS-Applikation ausgelesen und ausgewertet werden.

**eState:** [Status der Verbindung \[► 528\]](#) zum Master.

### Beispiele (High-Level Interface):

- [IEC 60870-5-101 minimale Unterstation \(slave\) \[► 592\]](#);
- [IEC 60870-5-101 komplexe Unterstation \(slave\) \[► 594\]](#);

### Weitere Beispiele:

```
PROGRAM test
VAR
  slavelAODB      : ARRAY[1..50] OF ST_IEC870_5_101AODBEntry;

  inputs AT%IB0   : ARRAY[0..999] OF BYTE;
  outputs AT%QB0  : ARRAY[0..999] OF BYTE;
  memory AT%MB0   : ARRAY[0..999] OF BYTE;
  data            : ARRAY[0..999] OF BYTE;

  server1         : FB_IEC870_5_101Slave;

  bEnable         : BOOL;
  eState          : E_IEC870_5_101SerialLinkState;

  bError         : BOOL;
  iecError        : ST_IEC870_5_101ErrorFifoEntry;
END_VAR

server1.protPara.linkAddr := 220;
server1.protPara.eLinkMode := eIEC870_LinkMode_Unbalanced;
server1.protPara.elinkAddrSize := eIEC870_LinkAddr_TwoOctets;

server1.sysPara.asduFmt.eAsduAddrSize := eIEC870_AsduAddr_TwoOctets;
server1.sysPara.asduFmt.eObjAddrSize := eIEC870_ObjAddr_ThreeOctets;
server1.sysPara.asduFmt.eCOTSize := eIEC870_COT_TwoOctets;
server1.sysPara.asduAddr := 7;
server1.sysPara.bUsePCTime := TRUE;
server1.sysPara.bSyncTime := TRUE;
server1.sysPara.bSyncPCTime := FALSE;
server1.sysPara.bEndOfInit := TRUE;
server1.sysPara.bPerCyclic := FALSE;
server1.sysPara.tPerCyclicBase := T#5s;
server1.sysPara.bBackScan := FALSE;
server1.sysPara.tBackScanCycle := T#30s;
server1.sysPara.bPerFRZ := TRUE;
```

```
server1.sysPara.tPerFRZCycle := T#15s;
server1.sysPara.dbgMode := IEC870_DEBUGMODE_LINKLAYER;
(* OR IEC870_DEBUGMODE_DEVSTATE OR IEC870_DEBUGMODE_ASDU;*)
server1.sysPara.bTimeCOT3 := FALSE;

server1( pInputs := ADR( inputs ),
        cbInputs := SIZEOF( inputs ),
        pOutputs := ADR( outputs ),
        cbOutputs := SIZEOF( outputs ),
        pMemory := ADR( memory ),
        cbMemory := SIZEOF( memory ),
        pData := ADR( data ),
        cbData := SIZEOF( data ),
        pAOEntries := ADR( slavelAODB ),
        cbAOEntries := SIZEOF( slavelAODB ),
        hSerial := P_SerialComm_HighSpeed.hSerial,
        bEnable := bEnable,
        eState=>eState );
```

Im folgenden Beispiel wird der Gerätefehler-Fifo zyklisch ausgelesen und die registrierten Fehler ins TwinCAT XAE->"Error Log"-Fenster geschrieben.

```
REPEAT
    server1.system.device.errors.RemoveError( getError=>iecError, bOk=>bError );
    IF bError THEN
        ADSLOGSTR( ADSLOG_MSGTYPE_ERROR OR ADSLOG_MSGTYPE_LOG, 'IEC 60870-5-101 slave error: 0x%s',
            DWORD_TO_HEXSTR( iecError.nErrId, 8, FALSE) );
    END_IF
UNTIL NOT bError
END_REPEAT
```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.1.1.3 FB\_IEC870\_5\_104Master**

```
FB_IEC870_5_104Master
-----
hTable T_HAODBTable
sSrvNetID T_AmsNetID
protPara ST_IEC870_5_104ProtocolParams
sysPara ST_IEC870_5_101SystemParams
acqPara ST_IEC870_5_101AcquisitionParams
pAOEntries POINTER TO ARRAY[0...IEC870_ANYSIZE_ARRAY] OF ST_IEC870_5_101AODBEntry
cbAOEntries UDINT
pInputs PVOID
cbInputs UDINT
pOutputs PVOID
cbOutputs UDINT
pMemory PVOID
cbMemory UDINT
pData PVOID
cbData UDINT
bEnable BOOL
ST_IEC870_5_104ExSystemInterface system
E_SocketConnectionState eConnState
E_IEC870_5_104DataTransferState eDTState
```

Mit einer Instanz des Funktionsbausteins FB\_IEC870\_5\_104Master kann in der TwinCAT SPS eine IEC60870-5-104 Zentralstation (Master, High-Level) implementiert werden. Pro Instanz des Funktionsbausteins wird eine Verbindung zum Slave aufgebaut.

Der Funktionsbaustein besitzt folgende Aktionen;

- **STARTDT** (startet den Datenaustausch);
- **STOPDT** (stoppt den Datenaustausch);

Im Normalfall wird der Datenaustausch automatisch gestartet, nachdem die Verbindung hergestellt wurde. Standardmäßig ist der Funktionsbaustein auch so konfiguriert. Bei Bedarf kann der Datenaustausch durch einen Aufruf der Aktionen gestoppt, bzw. gestartet werden.

**VAR\_IN\_OUT**

```
VAR_IN_OUT
  hTable : T_HAODBTable;
END_VAR
```

**hTable:** Applikationsobjekt-Datenbankhandle [[▶ 531](#)] (Hash-Tabellenhandle). Das Tabellenhandle muss vor der Benutzung einmalig mit der Funktion [F\\_iecCreateTableHnd](#) [[▶ 452](#)] initialisiert werden.

**VAR\_INPUT**

```
VAR_INPUT
  sSrvNetID   : T_AmsNetID := '';
  protPara    : ST_IEC870_5_104ProtocolParams := ( bControlDT := TRUE, bDTControlled := FALSE,
                                                    sRemoteHost := '', nRemotePort := 2404 );
  sysPara     : ST_IEC870_5_101SystemParams := ( bEndOfInit := FALSE, asduAddr := 11, tSyncTimeut
:= T#0s );
  acqPara     : ST_IEC870_5_101AcquisitionParams;
  pAOEntries  : POINTER TO ARRAY[0..IEC870_ANYSIZE_ARRAY] OF ST_IEC870_5_101AODBEntry := 0;
  cbAOEntries : UDINT := 0;
  pInputs     : PVOID := 0;
  cbInputs    : UDINT := 0;
  pOutputs    : PVOID := 0;
  cbOutputs   : UDINT := 0;
  pMemory     : PVOID := 0;
  cbMemory    : UDINT := 0;
  pData       : PVOID := 0;
  cbData      : UDINT := 0;
  bEnable     : BOOL := TRUE;
END_VAR
```

**sSrvNetID:** String mit der Netzwerkadresse des TwinCAT TCP/IP Connection Servers. Für den lokalen Rechner (default) kann auch ein Leerstring angegeben werden.

**protPara:** [IEC 60870-5-104-Protokolparameter](#) [[▶ 549](#)].

**sysPara:** [Systemparameter](#) [[▶ 487](#)].

**acqPara:** [Parameter für die zyklische Datenerfassung](#) [[▶ 491](#)].

**pAOEntries:** Adresse der ersten [Applikationsobjekt-Datenbankvariablen](#) [[▶ 482](#)].

**cbAOEntries:** Bytegröße der Applikationsobjekt-Datenbankvariablen.

**pInputs:** Adresse des SPS-Prozessdatenbereichs der Eingänge.

**cbInputs:** Bytegröße des SPS-Prozessdatenbereichs der Eingänge.

**pOutputs:** Adresse des SPS-Prozessdatenbereichs der Ausgänge.

**cbOutputs:** Bytegröße des SPS-Prozessdatenbereichs der Ausgänge.

**pMamory:** Adresse des SPS-Prozessdatenbereichs der Merker.

**cbMamory:** Bytegröße des SPS-Prozessdatenbereichs der Merker.

**pData:** Adresse des SPS-Datenbereichs.

**cbData:** Bytegröße des SPS-Datenbereichs.

**bEnable:** Aktiviert/Deaktiviert den Funktionsbaustein (Kommunikation und Verbindungen).

Die Adressen können mit dem ADR- und die Bytegrößen mit dem SIZEOF-Operator ermittelt werden.

**VAR\_OUTPUT**

```
VAR_OUTPUT
  system      : ST_IEC870_5_104ExSystemInterface;
  eConnState  : E_SocketConnectionState := eSOCKET_DISCONNECTED;
  eDTState    : E_IEC870_5_104DataTransferState := eIEC870_STOPDPT;
END_VAR
```

**system:** [System-Interface](#) [[▶ 552](#)]. Diese Variable dient anderen IEC-Applikationsfunktionen als Kommunikationsschnittstelle zum IEC-Gerät (hier: Zentralstation).

- Membervariable *system.device* wird z.B. von der *F\_iecSetAOQuality* [▶ 437]-Funktion als VAR\_IN\_OUT-Parameter erwartet.
- Membervariable *system.device.errors* ist ein Gerätefehler-Fifo. Die registrierten Fehler können von der SPS-Applikation ausgelesen und ausgewertet werden.

**eConnState:** Status der TCP/IP-Verbindung zum Slave.

**eDTState:** Status des IEC 60870-5-104-Datenaustauschs [▶ 530] (STARTDT, STOPDT)

**Beispiele (High-Level interface):**

- IEC 60870-5-104 minimale Zentralstation (master) [▶ 632];
- IEC 60870-5-104 komplexe Zentralstation (master); [▶ 647]

**Weitere Beispiele:**

Im folgenden Beispiel wird der Gerätefehler-Fifo zyklisch ausgelesen und die registrierten Fehler ins TwinCAT XAE->"Error List"-Fenster geschrieben.

```

...
fbClient : FB_IEC870_5_104Master; (* IEC 60870-5-104 master function
block instance *)
...
...
REPEAT
    fbClient.system.device.errors.RemoveError( );
    IF fbClient.system.device.errors.bOk THEN
        ADSLOGSTR( ADSLOG_MSGTYPE_ERROR OR ADSLOG_MSGTYPE_LOG,
'IEC60870-5-104 master error: 0x%s',
                DWORD_TO_HEXSTR(
fbClient.system.device.errors.getError.nErrId, 8, FALSE) );
        END_IF
UNTIL NOT fbClient.system.device.errors.bOk
END_REPEAT
...

```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.1.1.4 FB\_IEC870\_5\_104Slave

FB_IEC870_5_104Slave	
hServer	T_HSERVER
protPara	ST_IEC870_5_104ProtocolParams
sysPara	ST_IEC870_5_101SystemParams
pAOEntries	POINTER TO ARRAY[0..IEC870_ANYSIZE_ARRAY] OF ST_IEC870_5_101AODBEntry
cbAOEntries	UDINT
pInputs	PVOID
cbInputs	UDINT
pOutputs	PVOID
cbOutputs	UDINT
pMemory	PVOID
cbMemory	UDINT
pData	PVOID
cbData	UDINT
bEnable	BOOL

Mit einer Instanz des Funktionsbausteins FB\_IEC870\_5\_104Slave kann in der TwinCAT SPS eine IEC60870-5-104 Unterstation (Slave, High-Level) implementiert werden. Eine Instanz des Funktionsbausteins kann nur eine Verbindung zu einem Master aufbauen.



Wenn Sie redundante Systeme mit zwei oder mehr Verbindungen zum Master realisieren wollen, benutzen Sie bitte den [FB\\_IEC870\\_5\\_104SlaveGrp](#) [► 395]-Funktionsbaustein.

#### VAR\_IN\_OUT

```
VAR_IN_OUT
    hServer      : T_HSERVER;
END_VAR
```

**hServer:** TCP/IP Server-Handle. Die internen Parameter der Server-Handle-Variable müssen vorher mit der Funktion F\_CreateServerHnd initialisiert werden.

#### VAR\_INPUT

```
VAR_INPUT
    protPara      : ST_IEC870_5_104ProtocolParams;
    sysPara       : ST_IEC870_5_101SystemParams;
    pAOEntries    : POINTER TO ARRAY[0..IEC870_ANYSIZE_ARRAY] OF ST_IEC870_5_101AODBEntry := 0;
    cbAOEntries   : UDINT := 0;
    pInputs       : PVOID := 0;
    cbInputs      : UDINT := 0;
    pOutputs      : PVOID := 0;
    cbOutputs     : UDINT := 0;
    pMemory       : PVOID := 0;
    cbMemory      : UDINT := 0;
    pData         : PVOID := 0;
    cbData        : UDINT := 0;
    bEnable       : BOOL := TRUE;
END_VAR
```

**protPara:** [IEC 60870-5-104-Protokolparameter](#) [► 549].

**sysPara:** [Systemparameter](#) [► 487].

**pAOEntries:** Adresse der ersten Applikationsobjekt-Datenbankvariablen [► 482].

**cbAOEntries:** Bytegröße der Applikationsobjekt-Datenbankvariablen.

**pInputs:** Adresse des SPS-Prozessdatenbereichs der Eingänge.

**cbInputs:** Bytegröße des SPS-Prozessdatenbereichs der Eingänge.

**pOutputs:** Adresse des SPS-Prozessdatenbereichs der Ausgänge.

**cbOutputs:** Bytegröße des SPS-Prozessdatenbereichs der Ausgänge.

**pMemory:** Adresse des SPS-Prozessdatenbereichs der Merker.

**cbMemory:** Bytegröße des SPS-Prozessdatenbereichs der Merker.

**pData:** Adresse des SPS-Datenbereichs.

**cbData:** Bytegröße des SPS-Datenbereichs.



**bEnable:** Aktiviert/Deaktiviert den Funktionsbaustein (Kommunikation und Verbindung).

Die Adressen können mit dem ADR- und die Bytegrößen mit dem SIZEOF-Operator ermittelt werden.

## VAR\_OUTPUT

```
VAR_OUTPUT
  system      : ST_IEC870_5_104SystemInterface;
  eConnState  : E_SocketConnectionState      := eSOCKET_DISCONNECTED;
  eDTState    : E_IEC870_5_104DataTransferState := eIEC870_STOPDT;
END_VAR
```

**system:** [System-Interface \[► 553\]](#). Diese Variable dient anderen IEC-Applikationsfunktionen als Kommunikationsschnittstelle zum IEC-Gerät (hier: Unterstation).

- Membervariable *system.device* wird z.B. von der [F\\_iecSetAOQuality \[► 437\]](#)-Funktion als VAR\_IN\_OUT-Parameter erwartet.
- Membervariable *system.device.errors* ist ein Gerätefehler-Fifo. Die registrierten Fehler können von der SPS-Applikation ausgelesen und ausgewertet werden.

**eConnState:** Status der TCP/IP-Verbindung zum Master.

**eDTState:** [Status des IEC 60870-5-104-Datenaustauschs \[► 530\]](#) (STARTDT, STOPDT)

### Beispiele (High-Level Interface):

- [IEC 60870-5-104 minimale Unterstation \(slave\) \[► 656\]](#);
- [IEC 60870-5-104 komplexe Unterstation \(slave\) \[► 674\]](#);

### Weitere Beispiele:

Aufruf in FUP:

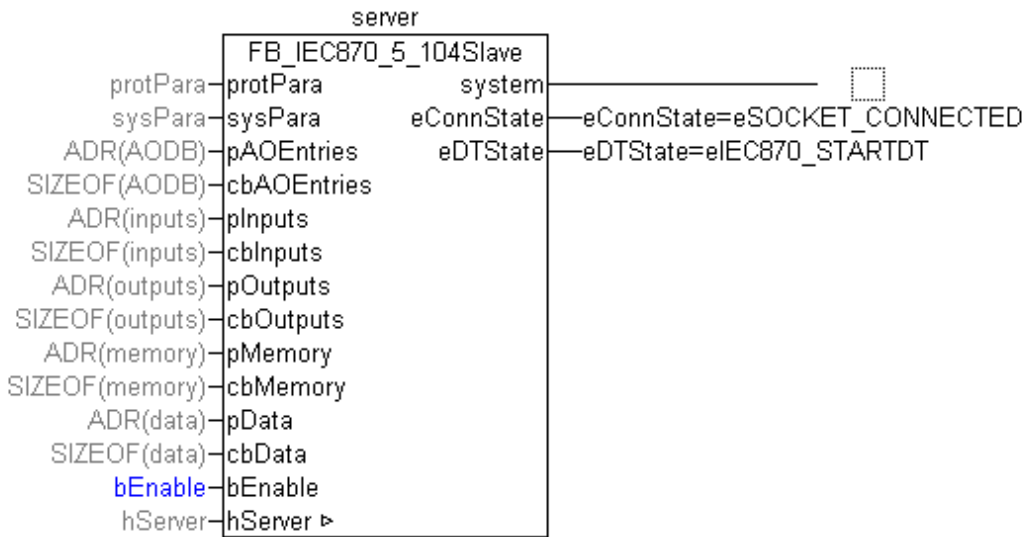
```
PROGRAM test
VAR
  AODB          : ARRAY[0..49] OF ST_IEC870_5_101AODBEntry;
  inputs AT%IB0 : ARRAY[0..999] OF BYTE;
  outputs AT%QB0 : ARRAY[0..999] OF BYTE;
  memory AT%MB0 : ARRAY[0..999] OF BYTE;
  data          : ARRAY[0..999] OF BYTE;

  hServer      : T_HSERVER;
  server       : FB_IEC870_5_104Slave;

  bEnable      : BOOL := TRUE;
  protPara     : ST_IEC870_5_104ProtocolParams;
  sysPara      : ST_IEC870_5_101SystemParams := ( asduAddr := 7 );
  eConnState   : E_SocketConnectionState;
  eDTState     : E_IEC870_5_104DataTransferState;
  bError       : BOOL;
  iecError     : ST_IEC870_5_101ErrorFifoEntry;
END_VAR
```

```

LISTEN_MODE_CLOSEALL OR CONNECT_MODE_ENABLEDBG
F_CreateServerHnd
"sSrvNetID
'172.16.6.195' sLocalHost
2404 nLocalPort
nMode
bEnable bEnable
hServer hServer ▶
    
```



Im folgenden Beispiel wird der Gerätefehler-Fifo zyklisch ausgelesen und die registrierten Fehler ins TwinCAT XAE->"Error List"-Fenster geschrieben.

```

REPEAT
  server.system.device.errors.RemoveError( getError=>iecError, bOk=>bError );
  IF bError THEN
    ADSLOGSTR( ADSLOG_MSGTYPE_ERROR OR ADSLOG_MSGTYPE_LOG, 'IEC 60870-5-104 slave error: 0x%s',
      DWORD_TO_HEXSTR( iecError.nErrId, 8, FALSE) );
  END_IF
UNTIL NOT bError
END_REPEAT
    
```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.1.1.5 FB\_IEC870\_5\_104SlaveGrp

```

FB_IEC870_5_104SlaveGrp
hServer T_HSERVER
protPara ST_IEC870_5_104ProtocolParams
sysPara ST_IEC870_5_101SystemParams
pAOEntries POINTER TO ARRAY[0..IEC870_ANYSIZE_ARRAY] OF ST_IEC870_5_101AODBEntry
cbAOEntries UDINT
pInputs PVOID
cbInputs UDINT
pOutputs PVOID
cbOutputs UDINT
pMemory PVOID
cbMemory UDINT
pData PVOID
cbData UDINT
pConnections POINTER TO ARRAY[0..IEC870_ANYSIZE_ARRAY] OF ST_IEC870_5_104ServerConnection
cbConnections UDINT
bEnable BOOL
    
```

Mit einer Instanz des Funktionsbausteins FB\_IEC870\_5\_104SlaveGrp kann in der TwinCAT SPS eine IEC 60870-5-104 Unterstation (Slave, High-Level) implementiert werden. Mit einer Instanz können einfache Systeme mit einer Verbindung oder Systeme mit mehreren Verbindungen zum Master realisiert werden (redundantes System).

Die maximale Anzahl der Verbindungen kann durch die Anzahl der ST\_IEC870\_5\_104ServerConnection [► 552]-Arrayelemente festgelegt werden. Die Adresse der Arrayvariablen und die Bytegröße der Arrayvariablen müssen dann an die Instanz des FB\_IEC870\_5\_104SlaveGrp-Funktionsbausteins übergeben werden.

#### VAR\_IN\_OUT

```

VAR_IN_OUT
hServer : T_HSERVER;
END_VAR
    
```

**hServer:** TCP/IP Server-Handle. Die internen Parameter der Server-Handle-Variable müssen vorher mit der Funktion F\_CreateServerHnd initialisiert werden.

#### VAR\_INPUT

```

VAR_INPUT
protPara : ST_IEC870_5_104ProtocolParams;
sysPara : ST_IEC870_5_101SystemParams;
pAOEntries : POINTER TO ARRAY[0..IEC870_ANYSIZE_ARRAY] OF ST_IEC870_5_101AODBEntry := 0;
cbAOEntries : UDINT := 0;
pInputs : PVOID := 0;
cbInputs : UDINT := 0;
pOutputs : PVOID := 0;
cbOutputs : UDINT := 0;
pMemory : PVOID := 0;
cbMemory : UDINT := 0;
pData : PVOID := 0;
cbData : UDINT := 0;
pConnections : POINTER TO ARRAY[0..IEC870_ANYSIZE_ARRAY] OF ST_IEC870_5_104ServerConnection := 0;
cbConnections : UDINT := 0;
bEnable : BOOL := TRUE;
END_VAR
    
```

**protPara:** IEC 60870-5-104-Protokolparameter. [► 549]

**sysPara:** Systemparameter [► 487].

**pAOEntries:** Adresse der ersten Applikationsobjekt-Datenbankvariablen [► 482].

**cbAOEntries:** Bytegröße der Applikationsobjekt-Datenbankvariablen.

**pInputs:** Adresse des SPS-Prozessdatenbereichs der Eingänge.

**cbInputs:** Bytegröße des SPS-Prozessdatenbereichs der Eingänge.

**pOutputs:** Adresse des SPS-Prozessdatenbereichs der Ausgänge.

**cbOutputs:** Bytegröße des SPS-Prozessdatenbereichs der Ausgänge.

**pMemory:** Adresse des SPS-Prozessdatenbereichs der Merker.

**cbMamory**: Bytegröße des SPS-Prozessdatenbereichs der Merker.

**pData**: Adresse des SPS-Datenbereichs.

**cbData**: Bytegröße des SPS-Datenbereichs.

**pConnections**: Adresse der [ST\\_IEC870\\_5\\_104ServerConnection](#) [[▶ 552](#)]-Arrayvariablen.

**cbConnections**: Bytegröße der [ST\\_IEC870\\_5\\_104ServerConnection](#) [[▶ 552](#)]-Arrayvariablen.

**bEnable**: Aktiviert/Deaktiviert den Funktionsbaustein (Kommunikation und Verbindungen).

Die Adressen können mit dem ADR- und die Bytegrößen mit dem SIZEOF-Operator ermittelt werden.

## VAR\_OUTPUT

```
VAR_OUTPUT
  system : ST_IEC870_5_104SystemInterface;
  status : ST_IEC870_5_104GrpStatus;
END_VAR
```

**system**: [System-Interface](#) [[▶ 553](#)]. Diese Variable dient anderen IEC-Applikationsfunktionen als Kommunikationsschnittstelle zum IEC-Gerät (hier: Unterstation).

- Membervariable *system.device* wird z.B. von der [F\\_iecSetAOQuality](#) [[▶ 437](#)]-Funktion als VAR\_IN\_OUT-Parameter erwartet.
- Membervariable *system.device.errors* ist ein Gerätefehler-Fifo. Die registrierten Fehler können von der SPS-Applikation ausgelesen und ausgewertet werden.

**status**: [Verbindungs- und Datentransfer-Statusinformationen](#) [[▶ 553](#)].

### Beispiel (High-Level Interface):

- [IEC 60870-5-104 Unterstation \(slave\)](#); [[▶ 676](#)]

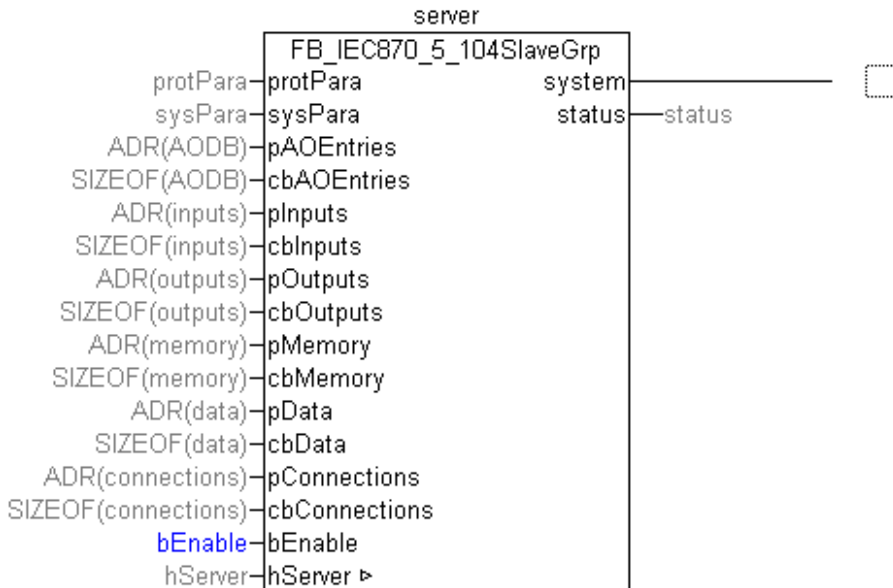
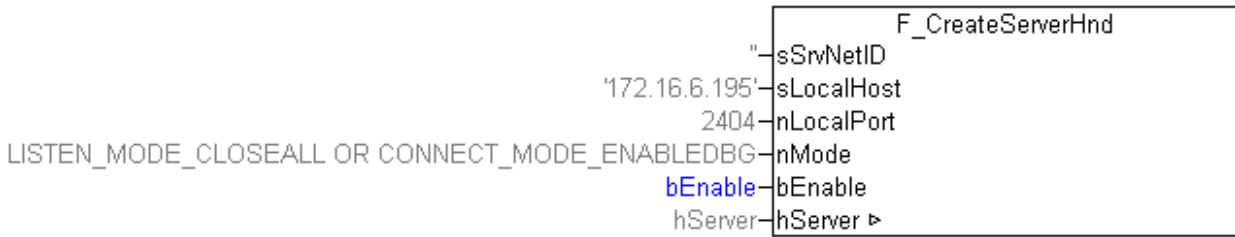
### Weitere Beispiele:

Aufruf in FUP mit max. 2 Master-Verbindungen:

```
PROGRAM test
VAR
  AODB : ARRAY[0..49] OF ST_IEC870_5_101AODBEntry;
  inputs AT%IB0 : ARRAY[0..999] OF BYTE;
  outputs AT%QB0 : ARRAY[0..999] OF BYTE;
  memory AT%MB0 : ARRAY[0..999] OF BYTE;
  data : ARRAY[0..999] OF BYTE;

  hServer : T_HSERVER;
  server : FB_IEC870_5_104SlaveGrp;
  connections : ARRAY[0..1] OF ST_IEC870_5_104ServerConnection; (* Two master connections *)

  bEnable : BOOL := TRUE;
  protPara : ST_IEC870_5_104ProtocolParams;
  sysPara : ST_IEC870_5_101SystemParams := ( asduAddr := 7 );
  status : ST_IEC870_5_104GrpStatus;
  bError : BOOL;
  iecError : ST_IEC870_5_101ErrorFifoEntry;
END_VAR
```



Im folgenden ST-Beispiel wird der Gerätefehler-Fifo zyklisch ausgelesen und die registrierten Fehler ins TwinCAT XAE->"Error List"-Fenster geschrieben.

```

REPEAT
  server.system.device.errors.RemoveError( getError=>iecError, bOk=>bError );
  IF bError THEN
    ADSLOGSTR( ADSLOG_MSGTYPE_ERROR OR ADSLOG_MSGTYPE_LOG, 'IEC 60870-5-104 slave error: 0x%s',
      DWORD_TO_HEXSTR( iecError.nErrId, 8, FALSE ) );
  END_IF
UNTIL NOT bError
END_REPEAT
  
```

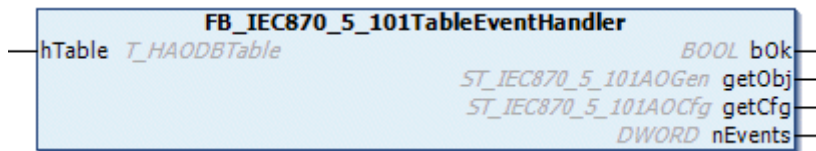
**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.1.1.6 FB\_IEC870\_5\_101TableEventHandler**



Das Auslesen der Ereignisse ist optional und nicht zwingend notwendig. D.h. die SPS-Applikation muss diesen Baustein nicht unbedingt instanzieren. Zur Zeit wird diese Funktionalität nur vom High-Level IEC 60870-5-101/104 Master unterstützt!



Mit diesem Funktionsbaustein kann die SPS-Applikation einige Änderungen in der IEC-Applikationsobjekt Datenbank erkennen und, wenn nötig, darauf entsprechend reagieren. Die Änderungen werden als Ereignisse bezeichnet. Jeder Ereignis-Typ wird in einer separaten internen Liste verwaltet. Die SPS-Applikation kann durch den Aufruf der Bausteinaktionen die anstehenden Ereignisse aus einer der Listen auslesen. Da auch mehrere Ereignisse pro SPS-Zyklus auftreten können werden die Ereignisse intern gezählt. Der Zähler wird beim Auftreten eines Ereignisses immer inkrementiert. An dem Bausteinausgang wird nur der letzte Wert und der Zählerstand ausgegeben.

Folgende Ereignisse werden von dem Funktionsbaustein registriert:

- **OnCreate-Ereignisse** werden immer dann gemeldet wenn ein neues Applikationsobjekt (Single Point, Double Point, Measured Value...) der Applikationsdatenbank hinzugefügt wurde.
- **OnChange-Ereignisse** werden dann gemeldet wenn ein Applikationsobjekt von der unteren Transportschicht empfangen wurde (Rx-Frames) oder verschickt wird (Tx-Frames), unabhängig davon ob sich der Wert des Informationsobjektes tatsächlich geändert hat oder nicht. Bei einem direkten Kommando z.B. C\_SC\_NA\_1 in Steuerungsrichtung (Master->Slave) werden im Normalfall bei folgenden Übertragungsursachen Ereignisse gemeldet: eIEC870\_COT\_ACT (Aktivierung), eIEC870\_COT\_ACT\_CON (Bestätigung der Aktivierung) und bei eIEC870\_COT\_TERM (Beendigung der Aktivierung). Bei einem Datenpunkt in Überwachungsrichtung (Slave->Master) z.B. M\_SP\_NA\_1 können bei folgenden Übertragungsursachen Ereignisse gemeldet werden: eIEC870\_COT\_SPONTAN, eIEC870\_COT\_INROGEN, eIEC870\_COT\_BACKGROUND usw.

Der Funktionsbaustein besitzt zwei Aktionen:

- **RemoveOnCreateEvent** (Liest einen Eintrag aus der OnCreate-Ereignisliste aus);
- **RemoveOnChangeEvent** (Liest einen Eintrag aus der OnChange-Ereignisliste aus);

## VAR\_IN\_OUT

```
VAR_IN_OUT
    hTable : T_HAODBTable;
END_VAR
```

**hTable:** [Applikationsobjekt-Datenbankhandle \[► 531\]](#) (Hash-Tabellenhandle). Das Tabellenhandle muss vor der Benutzung einmalig mit der Funktion [F\\_iecCreateTableHnd \[► 452\]](#) initialisiert werden.

## VAR\_OUTPUT

```
VAR_OUTPUT
    bOk      : BOOL := FALSE;
    getObj   : ST_IEC870_5_101AObj;
    getCfg   : ST_IEC870_5_101AOCfg;
    nEvents  : DWORD := 0;
END_VAR
```

**bOk:** Diese Variable wird TRUE, wenn ein Ereignis erfolgreich ausgelesen wurde. Bei FALSE ist die zuletzt gelesene Ereignisliste leer.

**getObj:** Der aktuelle [Wert der Dateneinheit \[► 483\]](#) (ASDU).

**getCfg:** Die aktuellen [Konfigurationsparameter der Dateneinheit \[► 483\]](#) (ASDU).

**nEvents:** Ereignis-Zähler (Multiplikator). Wertebereich: (0 bis 16#FFFFFFFF). Beim Erreichen des Maximalwertes wird nicht mehr inkrementiert.

## Beispiel in ST:

Im folgenden Programmausschnitt werden die anstehenden Ereignisse in REPEAT-Schleifen ausgelesen und in Windows Application Log geschrieben. Die gesuchten Datenpunkte sind bereits als Hash-Tabelleneinträge konfiguriert worden. Siehe in der Beschreibung der Funktion: [F\\_iecAddTableEntry \[► 453\]](#).

```

PROGRAM P_LogEvents
VAR_IN_OUT
  hTable      : T_HAODBTable;
END_VAR
VAR
  fbHandler : FB_IEC870_5_101TableEventHandler;
END_VAR
REPEAT
  fbHandler.RemoveOnChangeEvent( hTable := hTable );
  IF fbHandler.bOk THEN
    ADSLOGSTR( ADSLOG_MSGTYPE_HINT OR ADSLOG_MSGTYPE_LOG,
      'RemoveOnChangeEvent(), IOA: %s',
      DWORD_TO_STRING( fbHandler.getObj.info.objAddr ) );
  END_IF
UNTIL NOT fbHandler.bOk
END_REPEAT

REPEAT
  fbHandler.RemoveOnCreateEvent( hTable := hTable );
  IF fbHandler.bOk THEN
    ADSLOGSTR( ADSLOG_MSGTYPE_HINT OR ADSLOG_MSGTYPE_LOG,
      'RemoveOnCreateEvent(), IOA: %s',
      DWORD_TO_STRING( fbHandler.getObj.info.objAddr ) );
  END_IF
UNTIL NOT fbHandler.bOk
END_REPEAT

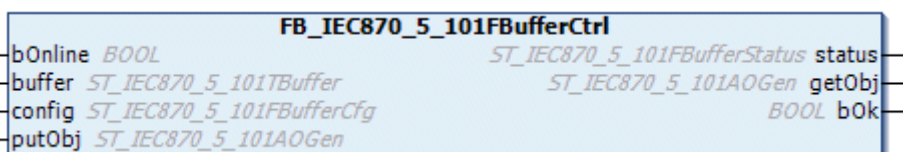
```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.1.2 Low Level**

**5.1.2.1 FB\_IEC870\_5\_101FBufferCtrl**



Mit diesem Funktionsbaustein kann der Inhalt des TX/RX-Datenpuffers manipuliert werden, der bei der Kommunikation über das IEC 60870-5-101/104 (Low Level) Transport Interface benutzt wird. Zusätzlich werden die zu versickenden ASDUs (nur TX-Richtung) in die Datei gepuffert wenn die Verbindung zur Zentralstation unterbrochen wurde (im Offline-Mode). Die Funktionalität ähnelt der Funktionalität des FB\_IEC870\_5\_101TBufferCtrl-Funktionsbausteins.

Der Funktionsbaustein besitzt folgende Aktionen:

- **RxRemoveObj** (entfernt den ältesten Fifoeintrag aus dem RX-Fifo);
- **RxReset** (löscht alle RX-Fifoeinträge, setzt den RX-Fifo zurück);
- **TxAddObj** (fügt einen neuen Fifoeintrag in den TX-Fifo);
- **TxReset** (löscht alle TX-Fifoeinträge, setzt den TX-Fifo zurück)

Durch den Aufruf der oben aufgelisteten Aktionen kann der Inhalt des TX/RX-Datenpuffers verändert werden.

## VAR\_IN\_OUT

```
VAR_IN_OUT
  bOnline : BOOL;
  buffer  : ST_IEC870_5_101TBuffer;
END_VAR
```

**bOnline:** Über diesen Eingang wird dem Funktionsbaustein mitgeteilt ob die Verbindung sich im Offline oder Online-Mode befindet. TRUE = Online, FALSE = Offline. Im Offline-Mode werden die zu versendenden ASDUs in einer Datei gepuffert. Im Online-Mode werden dann die in der Datei gepufferten ASDUs aus der Datei entfernt und an die Zentralstation verschickt.

**buffer:** TX/RX-Datenpuffer [► 481]. Die TX/RX-Pufferparameter (wie z.B. asduSize ) müssen vor der Benutzung konfiguriert werden.

## VAR\_INPUT

```
VAR_INPUT
  config : ST_IEC870_5_101FBufferCfg;
  putObj : ST_IEC870_5_101AObj;
END_VAR
```

**config:** Offline-Dateipuffer-Konfigurationseinstellungen [► 536].

**putObj:** Dateneinheit [► 483] (ASDU), die gesendet werden soll.

## VAR\_OUTPUT

```
VAR_OUTPUT
  status : ST_IEC870_5_101FBufferStatus;
  getObj : ST_IEC870_5_101AObj;
  bOk    : BOOL;
END_VAR
```

**status:** Offline-Datenpuffer-Statusinformationen [► 537].

**getObj:** Empfangene Dateneinheit [► 483] (ASDU).

**bOk:** Diese Variable wird TRUE, wenn ein neuer Eintrag erfolgreich hinzugefügt oder aus dem Fifo entfernt wurde. Diese Variable wird FALSE beim Pufferüberlauf und wenn kein Eintrag entfernt werden konnte weil der Fifo bereits leer ist.

## Beispiel in ST:

Folgender Beispielcode (Ausschnitt) demonstriert die Benutzung der Baustein-Aktionen. Alle ~1s (tCycle) wird eine neue ASDU (M\_BO\_TB\_1) mit der Übertragungsursache: Spontan und Zeitstempel generiert und in den TX-Fifo abgelegt.

Die empfangenen Testkommandos (C\_TS\_TA\_1) und Uhrzeitsynchronisationskommandos (C\_CS\_NA\_1) werden aus dem RX-Fifo entfernt und mit gespiegelten ASDUs beantwortet.

Über die VAR\_IN\_OUT-Variable *bOnline* wird das Speichern oder Laden der ASDUs in die Datei gesteuert. Bei *bOnline* = FALSE wird die Datei geöffnet und die anfallenden TX-ASDUs im Hintergrund in die Datei geschrieben. Bei *bOnline* = TRUE werden die ASDUs aus der Datei im Hintergrund geladen und verschickt. Damit das Speichern und Laden der Pufferdatei im Hintergrund durchgeführt werden kann muss der FB\_IEC870\_5\_101FBufferCtrl-Funktionsbaustein zyklisch aufgerufen werden. Das Hinzufügen neuer TX-ASDUs oder das Bearbeiten alter RX-ASDUs wird von dem Speichern/Laden in/aus der Datei nicht beeinflusst.

```
PROGRAM P_ProcessSlaveBufferData
VAR_IN_OUT
  bOnline : BOOL;
  buffer  : ST_IEC870_5_101TBuffer;
END_VAR
VAR
  asduAddr : DWORD := 7; (* Common asdu address *)

  fbBuffer : FB_IEC870_5_101FBufferCtrl :=( config := ( sPathName := 'c:\tmp\OfflineAsdu.dat',
    bOverwrite := TRUE,
    cbBuffer := 16#100000 )); (* RX/TX buffer control function block *)

  txAsdu : ST_IEC870_5_101AObj; (* asdu to send *)
  txTT   : T_CP56Time2a; (* time tag to send *)
```



```

rxAsdu   : ST_IEC870_5_101AOGen; (* received asdu *)
rxTT     : T_CP56Time2a; (* received time tag *)

rxQOI    : BYTE; (* qualifier of interrogation command *)
txBSI    : DWORD := 1; (* bit string value *)
txQDS    : BYTE; (* bit string quality descriptor *)
tCycle   : TIME := T#1s;
bSpont   : BOOL := TRUE;
timer    : TON;
fbRTC    : RTC_EX2 := ( EN := TRUE, PDT := ( wYear := 2006, wMonth := 8, wDay := 17, wHour := 12
, wMinute := 23 ) );
END_VAR

timer( IN := bSpont, PT := tCycle );
IF timer.Q THEN
  timer( IN := FALSE ); timer( IN := bSpont );

  txBSI := ROL( txBSI, 1); (* Modify bit string value *)
  txQDS.7 := NOT txQDS.7; (* Toggle IV quality flag *) (* create dummy time tag *)
  fbRTC();
  txTT := SYSTEMTIME_TO_CP56Time2a( fbRTC.CDT, TRUE );

  (* create asdu *)
  txAsdu.ident.eType := M_BO_TB_1; (* Bit string with time tag *)
  txAsdu.ident.bsQ := FALSE;
  txAsdu.ident.nObj := 1;
  txAsdu.ident.eCOT := eIEC870_COT_SPONTAN;
  txAsdu.ident.nORG := 1;
  txAsdu.ident.bPN := FALSE;
  txAsdu.ident.bt := FALSE;
  txAsdu.ident.eClass := eIEC870_Class_1;
  txAsdu.ident.asduAddr := asduAddr;
  txAsdu.info.objAddr := 100;
  F_iecResetStream( 0, txAsdu.info.stream ); (* clear previous data (this sets the stream length =
0 *)
  F_iecCopyBufferToStream( ADR( txBSI ), SIZEOF( txBSI ), txAsdu.info.stream ); (* put BSI to stre
am *)
  F_iecCopyBufferToStream( ADR( txQDS ), SIZEOF( txQDS ), txAsdu.info.stream ); (* put QDS to stre
am *)
  F_iecCopyBufferToStream( ADR( txTT ), SIZEOF( txTT ), txAsdu.info.stream ); (* put time tag to s
tream *)

  fbBuffer.TxAddObj( bOnline := bOnline, putObj := txAsdu, buffer := buffer ); (* put asdu to the
TX fifo *)
  IF NOT fbBuffer.bOk THEN
    RETURN;
    (* TODO: Report send buffer overflow error *)
  END_IF
END_IF

REPEAT
  fbBuffer.RxRemoveObj( bOnline := bOnline, getObj=>rxAsdu, buffer := buffer ); (* Try to remove a
sdu from RX fifo *)
  IF fbBuffer.RxRemoveObj.bOk THEN (* success *)
    CASE rxAsdu.ident.eType OF

      C_TS_NA_1: (* Simple test command implementation *)

        txAsdu := rxAsdu;
        txAsdu.ident.eCOT := eIEC870_COT_ACT_CON; (* send activation confirmation *)
        fbBuffer.TxAddObj( bOnline := bOnline, putObj := txAsdu, buffer := buffer ); (* put asdu to
the TX fifo *)
        IF NOT fbBuffer.bOk THEN
          EXIT;
          (* TODO: Report send buffer overflow error *)
        END_IF

      C_CS_NA_1: (* Simple clock synchronisation command implementation *)

        F_iecCopyStreamToBuffer( ADR( rxTT ), SIZEOF( rxTT ), rxAsdu.info.stream );

        (*... *)

        txAsdu := rxAsdu; (* dummy old time value *)
        txAsdu.ident.eCOT := eIEC870_COT_ACT_CON; (* send activation confirmation *)
        fbBuffer.TxAddObj( bOnline := bOnline, putObj := txAsdu, buffer := buffer ); (* put asdu to
the TX fifo *)

```

```

    IF NOT fbBuffer.bOk THEN
        EXIT;
        (* TODO: Report send buffer overflow error *)
    END_IF

    C_IC_NA_1: (* Simple interrogation command implementation *)

    txAsdu := rxAsdu;
    txAsdu.ident.eCOT := eIEC870_COT_ACT_CON; (* send activation confirmation *)
    fbBuffer.TxAddObj( bOnline := bOnline, putObj := txAsdu, buffer := buffer ); (* put asdu to
the TX fifo *)
    IF NOT fbBuffer.bOk THEN
        EXIT;
        (* TODO: Report send buffer overflow error *)
    END_IF

    F_iecCopyStreamToBuffer( ADR( rxQOI ), SIZEOF(rxQOI), rxAsdu.info.stream );

    (* create asdu *)
    txAsdu.ident.eType := M_BO_NA_1; (* Bit string without time tag! *)
    txAsdu.ident.bsQ := FALSE;
    txAsdu.ident.nObj := 1;
    txAsdu.ident.eCOT := BYTE_TO_INT( rxQOI );
    txAsdu.ident.nORG := 1;
    txAsdu.ident.bPN := FALSE;
    txAsdu.ident.bT := FALSE;
    txAsdu.ident.eClass := eIEC870_Class_1;
    txAsdu.ident.asduAddr := asduAddr;
    txAsdu.info.objAddr := 100;
    F_iecResetStream( 0, txAsdu.info.stream ); (* clear previous data (this sets the stream leng
th = 0 *)
    F_iecCopyBufferToStream( ADR( txBSI ), SIZEOF( txBSI ), txAsdu.info.stream ); (* put BSI to
stream *)
    F_iecCopyBufferToStream( ADR( txQDS ), SIZEOF( txQDS ), txAsdu.info.stream ); (* put QDS to
stream *)
    fbBuffer.TxAddObj( bOnline := bOnline, putObj := txAsdu, buffer := buffer ); (* put asdu to
the TX fifo *)
    IF NOT fbBuffer.bOk THEN
        EXIT;
        (* TODO: Report send buffer overflow error *)
    END_IF

    txAsdu := rxAsdu;
    txAsdu.ident.eCOT := eIEC870_COT_ACT_TERM; (* send activation termination *)
    fbBuffer.TxAddObj( bOnline := bOnline, putObj := txAsdu, buffer := buffer ); (* put asdu to
the TX fifo *)
    IF NOT fbBuffer.bOk THEN
        EXIT;
        (* TODO: Report send buffer overflow error *)
    END_IF

    ELSE
        (* TODO: Report invalid asdu type...*)
        EXIT;
    END_CASE

    END_IF
UNTIL NOT fbBuffer.bOk (* RX fifo is empty *)
END_REPEAT

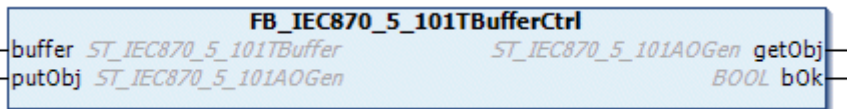
(* Offline frames are written to the file. Execute this function block in every cycle! *)
fbBuffer(bOnline := bOnline, buffer:= buffer );
IF fbBuffer.status.eState = eIEC870_FBUFFER_ERROR THEN
    (*TODO: Report file access error *)
    ;
END_IF

```

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

5.1.2.2 FB\_IEC870\_5\_101TBufferCtrl



Mit diesem Funktionsbaustein kann der Inhalt des TX/RX-Datenpuffers manipuliert werden, der bei der Kommunikation über das IEC 60870-5-101/104 Low-Level Interface benutzt wird.

Der Funktionsbaustein besitzt folgende Aktionen:

- **RxRemoveObj** (entfernt den ältesten Fifoeintrag aus dem RX-Fifo);
- **RxReset** (löscht alle RX-Fifoeinträge, setzt den RX-Fifo zurück);
- **TxAddObj** (fügt einen neuen Fifoeintrag in den TX-Fifo);
- **TxReset** (löscht alle TX-Fifoeinträge, setzt den TX-Fifo zurück)

Durch den Aufruf der oben aufgelisteten Aktionen kann der Inhalt des TX/RX-Datenpuffers verändert werden.

VAR\_IN\_OUT

```
VAR_IN_OUT
  buffer : ST_IEC870_5_101TBuffer;
END_VAR
```

**buffer:** TX/RX-Datenpuffer [▶ 481]. Die TX/RX-Pufferparameter (wie z.B. asduSize ) müssen vor der Benutzung konfiguriert werden.

VAR\_INPUT

```
VAR_INPUT
  putObj : ST_IEC870_5_101AObj;
END_VAR
```

**putObj:** Dateneinheit [▶ 483] (ASDU), die gesendet werden soll.

VAR\_OUTPUT

```
VAR_OUTPUT
  getObj : ST_IEC870_5_101AObj;
  bOk : BOOL;
END_VAR
```

**getObj:** Empfangene Dateneinheit [▶ 483] (ASDU).

**bOk:** Diese Variable wird TRUE, wenn ein neuer Eintrag erfolgreich hinzugefügt oder aus dem Fifo entfernt wurde. Diese Variable wird FALSE beim Pufferüberlauf und wenn kein Eintrag entfernt werden konnte weil der Fifo bereits leer ist.

Beispiele (Low-Level Interface):

- IEC 60870-5-101 Zentralstation (master, unbalanced mode); [▶ 587]

- [IEC 60870-5-101 Unterstation \(slave, unbalanced mode\) \[► 618\]](#);
- [IEC 60870-5-101 Zentralstation \(master, balanced mode\) \[► 588\]](#);
- [IEC 60870-5-101 Unterstation \(slave, balanced mode\) \[► 619\]](#);

### Weitere Beispiele:

Folgendes Beispielcode (Ausschnitt) demonstriert die Benutzung der Baustein-Aktionen. Alle ~100ms (tCycle) wird eine neue ASDU (M\_BO\_TB\_1) mit der Übertragungsursache: Spontan und Zeitstempel generiert und in den TX-Fifo abgelegt.

Die empfangenen Testkommandos (C\_TS\_TA\_1) und Uhrzeitsynchronisationskommandos (C\_CS\_NA\_1) werden aus dem RX-Fifo entfernt und mit gespiegelten ASDUs beantwortet.

```

PROGRAM P_SAMPLE_1ms
VAR
  (* TX/RX data buffer *)
  fbBuffer : FB_IEC870_5_101TBufferCtrl;
  buffer    : ST_IEC870_5_101TBuffer := ( asduSize := 253 );

  tCycle    : TIME := T#100ms;
  timer     : TON;
  dtStart   : DT := DT#2006-07-05-12:34:56;

  txAsdu    : ST_IEC870_5_101AOGen;
  rxAsdu    : ST_IEC870_5_101AOGen;

  txBSI     : DWORD := 1;
  txQDS     : BYTE;
  txTT      : T_CP56Time2a;
  rxTT      : T_CP56Time2a;
END_VAR

timer( IN := TRUE, PT := tCycle );
IF timer.Q THEN
  timer( IN := FALSE );

  txAsdu.ident.eType := M_BO_TB_1; (* Bit string with time tag *)
  txAsdu.ident.bsQ := FALSE;
  txAsdu.ident.nObj := 1;
  txAsdu.ident.eCOT := eIEC870_COT_SPONTAN;
  txAsdu.ident.nORG := 1;
  txAsdu.ident.bPN := FALSE;
  txAsdu.ident.bT := FALSE;
  txAsdu.ident.eClass := eIEC870_Class_1;
  txAsdu.ident.asduAddr := 7;
  txAsdu.info.objAddr := 100;

  txBSI := ROL( txBSI, 1); (* Modify bit string value *)
  txQDS.7 := NOT txQDS.7; (* Toggle IV quality flag *) (* create dummy time tag *)
  dtStart := dtStart + tCycle;
  txTT := SYSTEMTIME_TO_CP56Time2a( DT_TO_SYSTEMTIME( dtStart ), TRUE );

  F_iecResetStream( 0, txAsdu.info.stream ); (* clear previous data (this sets the stream length =
0 *)
  F_iecCopyBufferToStream( ADR( txBSI ), SIZEOF( txBSI ), txAsdu.info.stream ); (* put BSI to stre
am *)
  F_iecCopyBufferToStream( ADR( txQDS ), SIZEOF( txQDS ), txAsdu.info.stream ); (* put QDS to stre
am *)
  F_iecCopyBufferToStream( ADR( txTT ), SIZEOF( txTT ), txAsdu.info.stream ); (* put time tag to s
tream *)

  fbBuffer.TxAddObj( putObj := txAsdu, buffer := buffer ); (* put asdu to the TX fifo *)
  IF NOT fbBuffer.bOk THEN
    ;(* Report send buffer overflow error *)
  END_IF
END_IF

REPEAT
fbBuffer.RxRemoveObj( getObj=>rxAsdu, buffer := buffer ); (* Try to remove asdu from RX fifo *)
IF fbBuffer.bOk THEN (* success *)

  CASE rxAsdu.ident.eType OF
    C_TS_TA_1: (* Test command *)
      txAsdu := rxAsdu;

```

```

txAsdu.ident.eCOT := eIEC870_COT_ACT_CON; (* send activation confirmation *)

fbBuffer.TxAddObj( putObj := txAsdu, buffer := buffer ); (* put asdu to the TX fifo *)
IF NOT fbBuffer.bOk THEN
    ;(* Report send buffer overflow error *)
END_IF

C_CS_NA_1: (* clock synchronisation *)

F_iecCopyStreamToBuffer( ADR( rxTT ), SIZEOF( rxTT ), rxAsdu.info.stream );

(*... *)

txAsdu := rxAsdu; (* dummy old time value *)
txAsdu.ident.eCOT := eIEC870_COT_ACT_CON; (* send activation confirmation *)

fbBuffer.TxAddObj( putObj := txAsdu, buffer := buffer ); (* put asdu to the TX fifo *)
IF NOT fbBuffer.bOk THEN
    ;(* Report send buffer overflow error *)
END_IF

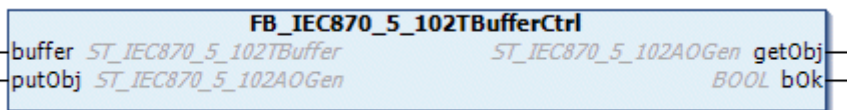
END_CASE

END_IF
UNTIL NOT fbBuffer.bOk (* RX fifo is empty *)
END_REPEAT
    
```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.1.2.3 FB\_IEC870\_5\_102TBufferCtrl**



Mit diesem Funktionsbaustein kann der Inhalt des TX/RX-Datenpuffers manipuliert werden, der bei der Kommunikation über das IEC60870-5-102 Low-Level Interface benutzt wird.

Der Funktionsbaustein besitzt folgende Aktionen:

- **RxRemoveObj** (entfernt den ältesten Fifoeintrag aus dem RX-Fifo);
- **RxReset** (löscht alle RX-Fifoeinträge, setzt den RX-Fifo zurück);
- **TxAddObj** (fügt einen neuen Fifoeintrag in den TX-Fifo);
- **TxReset** (löscht alle TX-Fifoeinträge, setzt den TX-Fifo zurück)

Durch den Aufruf der oben aufgelisteten Aktionen kann der Inhalt des TX/RX-Datenpuffers verändert werden.

**VAR\_IN\_OUT**

```

VAR_IN_OUT
    buffer : ST_IEC870_5_102TBuffer;
END_VAR
    
```

**buffer:** TX/RX-Datenpuffer [► 544]. Die TX/RX-Pufferparameter (wie z.B. asduSize ) müssen vor der Benutzung konfiguriert werden.

**VAR\_INPUT**

```
VAR_INPUT
  putObj : ST_IEC870_5_102AObj;
END_VAR
```

**putObj:** Dateneinheit [► 545] (ASDU), die gesendet werden soll.

**VAR\_OUTPUT**

```
VAR_OUTPUT
  getObj : ST_IEC870_5_102AObj;
  bOk    : BOOL;
END_VAR
```

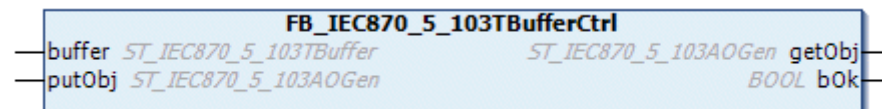
**getObj:** Empfangene Dateneinheit [► 545] (ASDU).

**bOk:** Diese Variable wird TRUE, wenn ein neuer Eintrag erfolgreich hinzugefügt oder aus dem Fifo entfernt wurde. Diese Variable wird FALSE beim Pufferüberlauf und wenn kein Eintrag entfernt werden konnte weil der Fifo bereits leer ist.

**Beispiel (Low-Level Interface):** IEC 60870-5-102 Zentralstation (master) [► 624];

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.1.2.4 FB\_IEC870\_5\_103TBufferCtrl**

Mit diesem Funktionsbaustein kann der Inhalt des TX/RX-Datenpuffers manipuliert werden, der bei der Kommunikation über das IEC60870-5-103 Low-Level Interface benutzt wird.

Der Funktionsbaustein besitzt folgende Aktionen:

- **RxRemoveObj** (entfernt den ältesten Fifoeintrag aus dem RX-Fifo);
- **RxReset** (löscht alle RX-Fifoeinträge, setzt den RX-Fifo zurück);
- **TxAddObj** (fügt einen neuen Fifoeintrag in den TX-Fifo);
- **TxReset** (löscht alle TX-Fifoeinträge, setzt den TX-Fifo zurück)

Durch den Aufruf der oben aufgelisteten Aktionen kann der Inhalt des TX/RX-Datenpuffers verändert werden.

**VAR\_IN\_OUT**

```
VAR_IN_OUT
  buffer : ST_IEC870_5_103TBuffer;
END_VAR
```

**buffer:** TX/RX-Datenpuffer [► 547]. Die TX/RX-Pufferparameter (wie z.B. asduSize ) müssen vor der Benutzung konfiguriert werden.

**VAR\_INPUT**

```
VAR_INPUT
  putObj : ST_IEC870_5_103A0Gen;
END_VAR
```

**putObj:** Dateneinheit [► 547] (ASDU), die gesendet werden soll.

**VAR\_OUTPUT**

```
VAR_OUTPUT
  getObj : ST_IEC870_5_103A0Gen;
  bOk : BOOL;
END_VAR
```

**getObj:** Empfangene Dateneinheit [► 547] (ASDU).

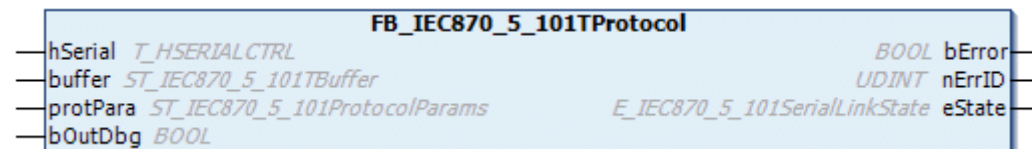
**bOk:** Diese Variable wird TRUE, wenn ein neuer Eintrag erfolgreich hinzugefügt oder aus dem Fifo entfernt wurde. Diese Variable wird FALSE beim Pufferüberlauf und wenn kein Eintrag entfernt werden konnte weil der Fifo bereits leer ist.

**Beispiel (Low-Level Interface):** IEC 60870-5-103 Zentralstation (master) [► 627]

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.1.2.5 FB\_IEC870\_5\_101TProtocol**



Der Kommunikationsbaustein FB\_IEC870\_5\_101TProtocol implementiert die Übertragungsprozeduren der Verbindungsschicht nach der IEC 60870-5-1 und IEC 60870-5-2-Norm. Beim Protokollfehler wird ein entsprechender Fehlercode am Ausgang des Funktionsbausteins ausgegeben und die Datenübertragung unterbrochen. Um den Datenaustausch erneut aktivieren zu können, muss die Aktion INIT aufgerufen werden. Es werden dabei z.B. die TX/RX-Datenpuffer zurückgesetzt. Der Kommunikationsbaustein erwartet eine TX/RX-Datenpuffervariable. Diese Variable muss per VAR\_IN\_OUT an den Baustein übergeben werden.

Der Funktionsbaustein besitzt folgende Aktionen:

- **INIT** (Führt eine Initialisierung des Funktionsbausteins durch);

**Protokollkonfiguration**

Der Kommunikationsbaustein besitzt eine protPara-Variable vom strukturierten Typ. Über diese Variable können Protokollparameter z.B. RX/TX-Timeoutzeiten usw. konfiguriert werden.

**VAR\_IN\_OUT**

```
VAR_IN_OUT
  hSerial : T_HSERIALCTRL;
  buffer : ST_IEC870_5_101TBuffer;
END_VAR
```

**hSerial:** [Verbindungs-Handle](#) [▶ 530] zum [FB\\_IEC870\\_SerialLineCtrl](#) [▶ 417]-Funktionsbaustein. Über diese Variable werden mit dem [FB\\_IEC870\\_SerialLineCtrl](#)-Funktionsbaustein die zu sendenden und empfangenen Daten ausgetauscht.

**buffer:** [TX/RX Datenpuffer](#) [▶ 481].

### VAR\_INPUT

```
VAR_INPUT
  protPara      : ST_IEC870_5_101ProtocolParams;
  bOutDbg       : BOOL;
```

**protPara:** [IEC 60870-5-101-Protokollparameter](#) [▶ 538].

**bOutDbg:** Aktiviert/deaktiviert die Debug-Ausgabe der Frames im TwinCAT XAE->"Error List"-Fenster.

### VAR\_OUTPUT

```
VAR_OUTPUT
  bError        : BOOL;
  nErrID        : UDINT;
  eState        : E_IEC870_5_101SerialLinkState := eSERIALLINK_DISCONNECTED;
```

**bError:** Dieser Ausgang wird auf TRUE gesetzt, sobald ein Fehler bei der Datenübertragung aufgetreten ist.

**nErrID:** Liefert bei einem gesetzten bError-Ausgang einen [Fehlercode](#) [▶ 685].

**eState:** [Verbindungsstatus](#) [▶ 528] zu dem Kommunikationspartner.

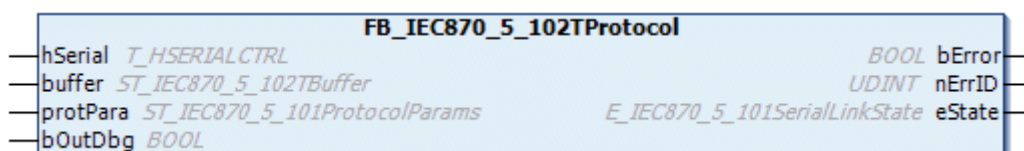
### Beispiele (Low-Level Interface):

- [IEC 60870-5-101 Zentralstation \(master, unbalanced mode\)](#); [▶ 587]
- [IEC 60870-5-101 Unterstation \(slave, unbalanced mode\)](#) [▶ 618];
- [IEC 60870-5-101 Zentralstation \(master, balanced mode\)](#) [▶ 588];
- [IEC 60870-5-101 Unterstation \(slave, balanced mode\)](#) [▶ 619];

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.1.2.6 FB\_IEC870\_5\_102TProtocol



Der Kommunikationsbaustein [FB\\_IEC870\\_5\\_102TProtocol](#) implementiert die Übertragungsprozeduren der Verbindungsschicht nach der IEC 60870-5-1 und IEC 60870-5-2-Norm. Beim Protokollfehler wird ein entsprechender Fehlercode am Ausgang des Funktionsbausteins ausgegeben



und die Datenübertragung unterbrochen. Um den Datenaustausch erneut aktivieren zu können, muss die Aktion INIT aufgerufen werden. Der Kommunikationsbaustein erwartet eine TX/RX-Datenpuffervariable. Diese Variable muss per VAR\_IN\_OUT an den Baustein übergeben werden.

Der Funktionsbaustein besitzt folgende Aktionen:

- **INIT** (Führt eine Initialisierung des Funktionsbausteins durch). In der Default-Konfiguration werden die TX/RX-Datenpuffer zurückgesetzt. Das Löschen der Puffer kann aber durch das Setzen der bRetainBuffer-Variablen in der Protokollparameterstruktur unterbunden werden.

**Protokollkonfiguration**

Der Kommunikationsbaustein besitzt eine protPara-Variable vom strukturierten Typ. Über diese Variable können Protokollparameter z.B. RX/TX-Timeoutzeiten usw. konfiguriert werden.

**VAR\_IN\_OUT**

```
VAR_IN_OUT
  hSerial : T_HSERIALCTRL;
  buffer  : ST_IEC870_5_102TBuffer;
END_VAR
```

**hSerial:** Verbindungs-Handle [▶ 530] zum FB IEC870\_SerialLineCtrl [▶ 417]-Funktionsbaustein. Über diese Variable werden mit dem FB\_IEC870\_SerialLineCtrl-Funktionsbaustein die zu sendenden und empfangenen Daten ausgetauscht.

**buffer:** TX/RX Datenpuffer [▶ 544].

**VAR\_INPUT**

```
VAR_INPUT
  protPara : ST_IEC870_5_101ProtocolParams;
  bOutDbg  : BOOL;
END_VAR
```

**protPara:** IEC 60870-5-101/102-Protokollparameter [▶ 538].

**bOutDbg::** Aktiviert/deaktiviert die Debug-Ausgabe der Frames im TwinCAT XAE->"Error List"-Fenster.

**VAR\_OUTPUT**

```
VAR_OUTPUT
  bError   : BOOL;
  nErrID   : UDINT;
  eState   : E_IEC870_5_101SerialLinkState := eSERIALLINK_DISCONNECTED;
END_VAR
```

**bError:** Dieser Ausgang wird auf TRUE gesetzt, sobald ein Fehler bei der Datenübertragung aufgetreten ist.

**nErrID:** Liefert bei einem gesetzten bError-Ausgang einen Fehlercode [▶ 694].

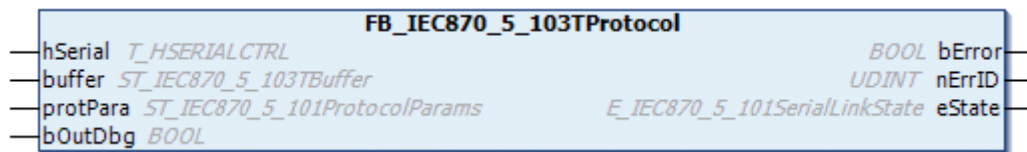
**eState:** Verbindungsstatus [▶ 528] zu dem Kommunikationspartner.

**Beispiel (Low-Level Interface):** IEC 60870-5-102 Zentralstation (master) [▶ 624];

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.1.2.7 FB\_IEC870\_5\_103TProtocol



Der Kommunikationsbaustein FB\_IEC870\_5\_103TProtocol implementiert die Übertragungsprozeduren der Verbindungsschicht nach der IEC 60870-5-1 und IEC 60870-5-2-Norm.

Beim Protokollfehler wird ein entsprechender Fehlercode am Ausgang des Funktionsbausteins ausgegeben und die Datenübertragung unterbrochen. Um den Datenaustausch erneut aktivieren zu können, muss die Aktion INIT aufgerufen werden. Der Kommunikationsbaustein erwartet eine TX/RX-Datenpuffervariable. Diese Variable muss per VAR\_IN\_OUT an den Baustein übergeben werden.

Der Funktionsbaustein besitzt folgende Aktionen:

- **INIT** (Führt eine Initialisierung des Funktionsbausteins durch). In der Default-Konfiguration werden die TX/RX-Datenpuffer zurückgesetzt. Das Löschen der Puffer kann aber durch das Setzen der bRetainBuffer-Variablen in der Protokollparameterstruktur unterbunden werden.

#### Protokollkonfiguration

Der Kommunikationsbaustein besitzt eine protPara-Variable vom strukturierten Typ. Über diese Variable können Protokollparameter z.B. RX/TX-Timeoutzeiten usw. konfiguriert werden.

#### VAR\_IN\_OUT

```
VAR_IN_OUT
  hSerial : T_HSERIALCTRL;
  buffer  : ST_IEC870_5_103TBuffer;
END_VAR
```

**hSerial:** [Verbindungs-Handle \[▶ 530\]](#) zum [FB\\_IEC870\\_SerialLineCtrl \[▶ 417\]](#)-Funktionsbaustein. Über diese Variable werden mit dem FB\_IEC870\_SerialLineCtrl-Funktionsbaustein die zu sendenden und empfangenen Daten ausgetauscht.

**buffer:** [TX/RX Datenpuffer \[▶ 547\]](#).

#### VAR\_INPUT

```
VAR_INPUT
  protPara : ST_IEC870_5_101ProtocolParams;
  bOutDbg  : BOOL;
END_VAR
```

**protPara:** [IEC 60870-5-101/103-Protokollparameter \[▶ 538\]](#).

**bOutDbg:** Aktiviert/deaktiviert die Debug-Ausgabe der Frames im TwinCAT XAE->"Error List"-Fenster.

#### VAR\_OUTPUT

```
VAR_OUTPUT
  bError   : BOOL;
  nErrID   : UDINT;
  eDState  : E_IEC870_5_101SerialLinkState := eSERIALLINK_DISCONNECTED;
END_VAR
```

**bError:** Dieser Ausgang wird auf TRUE gesetzt, sobald ein Fehler bei der Datenübertragung aufgetreten ist.

**nErrID:** Liefert bei einem gesetzten bError-Ausgang einen [Fehlercode \[▶ 701\]](#).

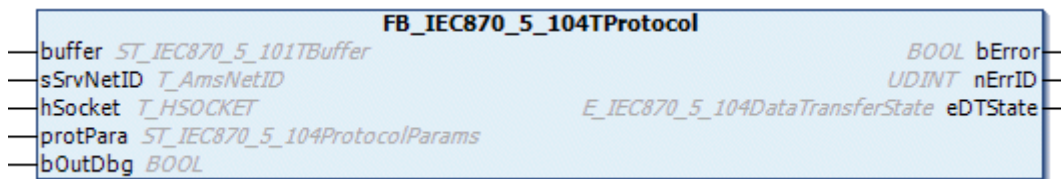
**eState:** [Verbindungsstatus \[▶ 528\]](#) zum anderen Kommunikationspartner.

**Beispiel (Low-Level Interface):** [IEC 60870-5-103 Zentralstation \(master\) \[▶ 627\]](#);

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

5.1.2.8 FB\_IEC870\_5\_104TProtocol



Der Kommunikationsbaustein FB\_IEC870\_5\_104TProtocol implementiert die ACPI-Funktionen der IEC 60870-5-104-Norm (Start/Stop-Data Transfer, Test-Frames, Send/Receive Frame-Counter usw.). Beim Protokollfehler wird ein entsprechender Fehlercode am Ausgang des Funktionsbausteins ausgegeben und die Datenübertragung unterbrochen. Um den Datenaustausch erneut aktivieren zu können, muss die Aktion INIT aufgerufen werden. Es werden dabei z.B. die Framezähler, Sende- und der TX/RX-Datenpuffer zurückgesetzt. Der Kommunikationsbaustein erwartet eine TX/RX-Datenpuffervariable. Diese Variable muss per VAR\_IN\_OUT an den Baustein übergeben werden.

Der Funktionsbaustein besitzt folgende Aktionen:

- **INIT** (Führt eine Initialisierung des Funktionsbausteins durch);
- **STARTDT** (Sendet ein Start-Data-Transfer-Frame an den Kommunikationspartner);
- **STOPDT** (Sendet Stopp-Data-Transfer-Frame an den Kommunikationspartner);

Verbindungsaufbau

Mit einem gesonderten Baustein z.B. FB\_ServerClientConnection muss die TCP/IP-Verbindung auf- und abgebaut werden. Die SPS-Applikation kann dadurch selber auf mögliche Protokollfehler reagieren und die Verbindung schließen oder z.B. den Dienst Process-Reset implementieren. Dieser Baustein liefert am Ausgang ein Verbindungshandle, den Verbindungsstatus und Informationen über Fehler die beim Verbindungsaufbau/-abbau aufgetreten sind.

Das Verbindungshandle wird von dem Kommunikationsbaustein benötigt.

Protokollkonfiguration

Der Kommunikationsbaustein besitzt eine protPara-Variable vom strukturierten Typ. Über diese Variable können Protokollparameter z.B. iK, iW, Start/Stop-Datentransfer-Verhalten usw. konfiguriert werden.

VAR\_IN\_OUT

```
VAR_IN_OUT
    buffer : ST_IEC870_5_101TBuffer;
END_VAR
```

**buffer:** TX/RX Datenpuffer [► 481].

VAR\_INPUT

```
VAR_INPUT
    sSrvNetID : T_AmsNetID;
    hSocket : T_HSOCKET;
```

```

protPara : ST_IEC870_5_104ProtocolParams;
bOutDbg  : BOOL;
END_VAR

```

**sSrvNetID:** String mit der Netzwerkadresse des TwinCAT TCP/IP Connection Servers. Für den lokalen Rechner (default) kann auch ein Leerstring angegeben werden.

**hSocket:** Das TCP/IP-Verbindungshandle des Kommunikationspartners dessen Daten empfangen oder gesendet werden sollen.

**protPara:** [IEC 60870-5-104-Protokollparameter](#) [► 549].

**bOutDbg::** Aktiviert/deaktiviert die Debug-Ausgabe der TCP/IP-Frames in der TwinCAT System Manager-Loggeransicht.

## VAR\_OUTPUT

```

VAR_OUTPUT
  bError      : BOOL;
  nErrID     : UDINT;
  eDTState   : E_IEC870_5_104DataTransferState := eIEC870_STOPDT;
END_VAR

```

**bError:** Dieser Ausgang wird auf TRUE gesetzt, sobald ein Fehler bei der Datenübertragung aufgetreten ist.

**nErrID:** Liefert bei einem gesetzten bError-Ausgang einen Fehlercode [► 708];

**eDTState:** Status des IEC 60870-5-104-Datenaustauschs [► 530] (STARTDT, STOPDT).

### Beispiele:

- [IEC 60870-5-104 Zentralstation \(master\)](#); [► 651]
- [IEC 60870-5-104 Unterstation \(slave\)](#); [► 680]

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.1.3 FB\_IEC870\_PartyLineCtrl



Mit dem Funktionsbaustein FB\_IEC870\_PartyLineCtrl kann der Datenaustausch zur Zentralstation im Linienbetrieb (partyline) betrieben werden. Der Funktionsbaustein muss in der SPS-Task zyklisch aufgerufen werden. Mit dem eMode-Eingang kann der Linienbetrieb aktiviert/deaktiviert werden. Wenn Sie den Funktionsbaustein nicht benutzen (Default Einstellung) wird kein Linienbetrieb betrieben.

Die hSerial-Variable ist eine Struktur und dient dem internen Datenaustausch zwischen der schnellen und langsamen Kommunikationstask. Jedes Mal wenn der IEC-Slave senden will wird der bRTS-Ausgang (request to send) zuerst auf TRUE gesetzt. Nachdem der Sendebetrieb eingeschaltet wurde wird dies dem IEC-Slave durch das Setzen des bCTS-Eingangs (clear to send) auf TRUE mitgeteilt. Danach beginnt der IEC-Slave zu senden. Nachdem die Daten gesendet wurden (interne Hardware-Puffer sind leer) setzt der

IEC-Slave den *bRTS*-Ausgang auf FALSE zurück. Jetzt kann die Sendeleitung für den anderen Teilnehmer freigegeben werden. Wenn dies geschehen ist, muss dies ebenfalls am *bCTS*-Eingang mit FALSE dem IEC-Slave mitgeteilt werden. D.h. der Zustand des *bCTS*-Eingangs folgt immer dem Zustand des *bRTS*-Ausgangs.

Bei der seriellen PC-Schnittstelle wird der *bRTS*-Ausgang erst dann auf FALSE gesetzt (Daten gesendet), wenn die Abfrage des internen Hardware-Sendepuffers Null Bytes im Puffer zurückliefert.

### VAR\_IN\_OUT

```
VAR_IN_OUT
  hSerial      : T_HSERIALCTRL;
END_VAR
```

**hSerial** : [Verbindungs-Handle \[► 530\]](#) zum [FB\\_IEC870\\_SerialLineCtrl \[► 417\]](#)-Funktionsbaustein. Über diese Variable werden mit dem [FB\\_IEC870\\_SerialLineCtrl](#)-Funktionsbaustein die zu sendenden und empfangenen Daten ausgetauscht.

### VAR\_INPUT

```
VAR_INPUT
  eMode       : E_IEC870_5_101PartylineMode := eIEC870_PartylineMode_Off;
  deviceID    : UDINT := 0;
  bCTS        : BOOL := FALSE;
END_VAR
```

**eMode** : [Partyline-Aktivierungsmodus \[► 528\]](#).

**deviceID** : Dieser Parameter wird in TwinCAT 3.1 nicht verwendet und muss nicht gesetzt werden.

**bCTS** : Clear to send (für den IEC-Link-Layer).

### VAR\_OUTPUT

```
VAR_IN_OUT
  bError       : BOOL;
  nErrID       : UDINT;
  bRTS         : BOOL := FALSE;
END_VAR
```

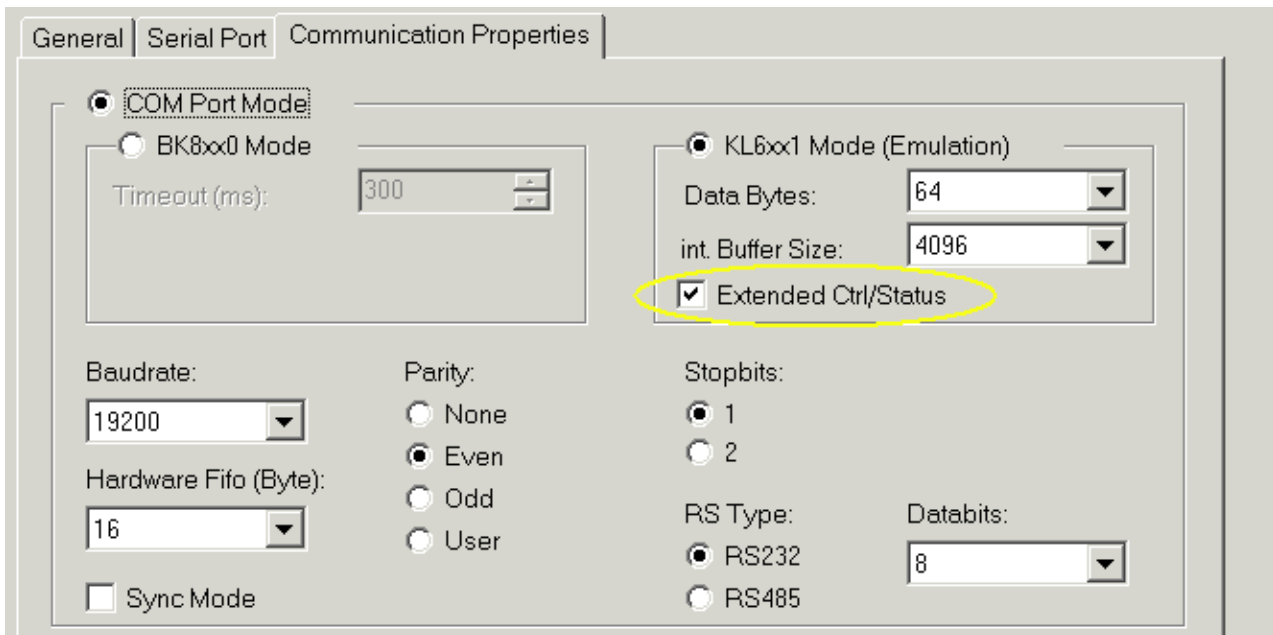
**bError**: Wird TRUE, sobald ein Fehler aufgetreten ist.

**nErrID**: Liefert bei einem gesetzten *bError*-Ausgang den [Fehlercode \[► 685\]](#).

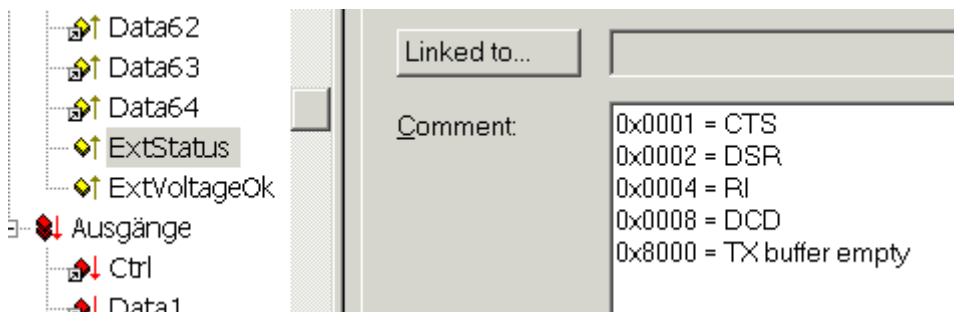
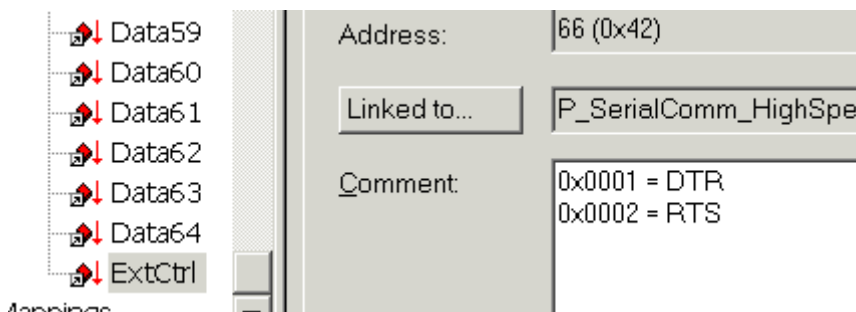
**bRTS**: Request to send (vom IEC-Link-Layer).

### Beispiel für Linienbetrieb-Konfiguration:

Die RTS-Leitung kann direkt als ein IO-Ausgang in die SPS gemappt werden. Sie müssen im TwinCAT System Manager die "**Extended Ctrl/Status**" Option aktivieren. Der *eMode*-Parameter an dem Funktionsbaustein muss *eMode := eIEC870\_PartylineMode\_Ext\_On* gesetzt werden.



Die "serial\_ExtCtrl"-Variable muss mit "ExtCtrl" (RTS-Ausgang) und die "serial\_ExtStatus"-Variable mit "ExtStatus" verknüpft werden.



```

PROGRAM P_SerialComm_HighSpeed
VAR
  serial_In      AT%I*   : PcComInData;
  serial_Out    AT%Q*   : PcComOutData;
  KL6_In        AT%I*   : KL6inData5B;
  KL6_Out       AT%Q*   : KL6outData5B;
  serial_ExtStatus AT%I* : WORD;
  serial_ExtCtrl AT%Q*   : WORD;

  hSerial       : T_HSERIALCTRL; (* Serial connection control handle *)
  fbSerialLineCtrl : FB_IEC870_SerialLineCtrl := ( Mode := SERIALLINEMODE_PC_COM_PORT(*SERIALLINEMODE_KL6_5B_STANDARD *) ,
    Baudrate := 19200,
    NoDatabits := 8,
    Parity := PARITY_EVEN,
    Stopbits := 1,
    Handshake := HANDSHAKE_NONE,
    ContinousMode := FALSE );

  fbPartyLineCtrl : FB_IEC870_PartyLineCtrl := ( eMode := eIEC870_PartylineMode_Ext_On );
  delay           : TON;

```

```

    tRTS_DEALY_ON      : TIME := T#100ms;
    tRTS_DELAY_OFF    : TIME := T#100MS;
END_VAR

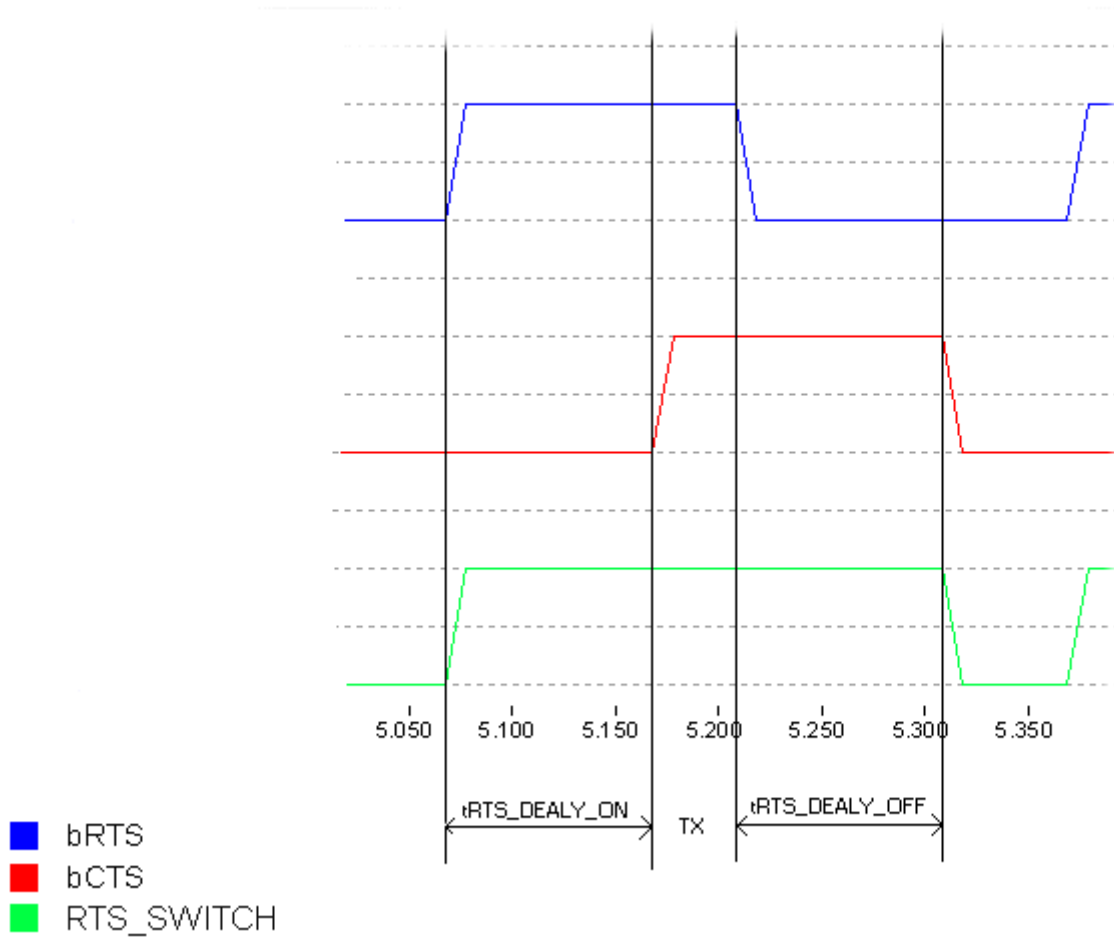
fbSerialLineCtrl( pComIn := SEL( fbSerialLineCtrl.Mode = SERIALLINEMODE_KL6_5B_STANDARD, ADR( serial_In ), ADR( KL6_In ) ),
                 pComOut := SEL( fbSerialLineCtrl.Mode = SERIALLINEMODE_KL6_5B_STANDARD, ADR( serial_Out ), ADR( KL6_Out ) ),
                 SizeComIn := SEL( fbSerialLineCtrl.Mode = SERIALLINEMODE_KL6_5B_STANDARD, SIZEOF( serial_In ), SIZEOF( KL6_In ) ),
                 hSerial := hSerial );

fbPartyLineCtrl( hSerial:= hSerial );
IF fbPartyLineCtrl.bRTS <> fbPartyLineCtrl.bCTS THEN
  IF fbPartyLineCtrl.bRTS THEN
    serial_ExtCtrl.1 := TRUE;(* Switch RTS line ON *)
    delay( in := TRUE, PT := tRTS_DEALY_ON );(* Wait until line enabled *)
    IF delay.Q THEN
      delay( in := FALSE );
      fbPartyLineCtrl.bCTS := TRUE;(* Set clear to send *)
    END_IF
  ELSE
    IF serial_ExtStatus.15 THEN
      delay( in := TRUE, PT := tRTS_DELAY_OFF );(* Wait until all data send *)
      IF delay.Q THEN
        delay( in := FALSE );
        serial_ExtCtrl.1 := FALSE;(* Switch RTS line OFF *)
        fbPartyLineCtrl.bCTS := FALSE;(* Reset clear to send *)
      END_IF
    END_IF
  END_IF
END_IF

```

Implementierung des Linienbetriebs in der schnellen Kommunikationstask: Über die *RTS\_SWITCH*-Variable (**serial\_ExtCtrl.1**) wird die Leitung für den Sendebetrieb EIN- und AUS-geschaltet. Die *tRTS\_DEALY\_ON*-Verzögerungszeit (Vorlaufzeit) stellt sicher, dass die Freischaltung der Leitung für den IEC-Slave abgeschlossen wurde, die *tRTS\_DELAY\_OFF*-Verzögerungszeit (Nachlaufzeit) stellt sicher, dass auch das letzte gesendete Datenbyte von der Zentralstation empfangen wurde (**serial\_ExtStatus.15**) .

## FB\_IEC870\_PartyLineCtrl



## Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)



## 5.1.4 FB\_IEC870\_SerialLineCtrl

FB_IEC870_SerialLineCtrl	
hSerial	T_HSERIALCTRL
Mode	ComSerialLineMode_t
Baudrate	UDINT
NoDatabits	BYTE
Parity	ComParity_t
Stopbits	BYTE
Handshake	ComHandshake_t
ContinousMode	BOOL
pComIn	PVOID
pComOut	PVOID
SizeComIn	UINT

Der Funktionsbaustein FB\_IEC870\_SerialLineCtrl wickelt die Kommunikation zwischen einer seriellen Schnittstelle (KL60xx, EL60xx oder COM-Schnittstelle) und den IEC60870-5-10x SPS-Bausteinen ab. Wenn Sie für die Kommunikation eine serielle Klemme **KL60xx** benutzen, dann wird die Busklemme zuerst von dem Baustein initialisiert und konfiguriert (Baudrate, Parity, usw.). Die Konfiguration der **PC-COM-Schnittstelle** und der Klemme **EL60xx** muss aber in TwinCAT System Manager durchgeführt werden. Die empfangenen und zu sendenden Daten werden in den internen Puffern der hSerial-Variablen gehalten. Der Funktionsbaustein muss in der SPS-Task zyklisch aufgerufen werden.

### VAR\_IN\_OUT

```
VAR_IN_OUT
  hSerial      : T_HSERIALCTRL;
END_VAR
```

**hSerial** : [Verbindungs-Handle](#) [► 530] zu anderen IEC 60870-5-10x Protokoll-Funktionsbausteinen. Über diese Variable werden mit dem FB\_IEC870\_SerialLineCtrl-Funktionsbaustein die zu sendenden und empfangenen Daten ausgetauscht.

### VAR\_INPUT

```
VAR_INPUT
  Mode          : ComSerialLineMode_t := SERIALLINEMODE_PC_COM_PORT;
  Baudrate      : UDINT               := 19200;
  NoDatabits    : BYTE                := 8;
  Parity        : ComParity_t := PARITY_EVEN;
  Stopbits      : BYTE                := 1;
  Handshake     : ComHandshake_t := HANDSHAKE_NONE;
  ContinousMode: BOOL                := FALSE;
  pComIn        : PVOID               := 0;
  pComOut       : PVOID               := 0;
  SizeComIn     : UINT                := 0;
END_VAR
```

**Mode**: Der Mode-Eingang legt eindeutig fest, welche serielle Hardware verwendet wird.

**Baudrate**: Die Baudrate, soweit durch die serielle Hardware unterstützt (nur bei KL60xx-Mode: 115200, 57600, 38400, 19200, 9600, 4800, 2400, 1200).

**NoDatabits**: Anzahl der Nutzdatenbits in einem Datenbyte (nur bei KL60xx-Mode: 7 or 8).

**Parity**: Typ des Paritybits eines Datenbytes (nur bei KL60xx-Mode: PARITY\_NONE=0, PARITY\_EVEN=1, PARITY\_ODD=2).

**Stopbits**: Anzahl der Stopbits pro Datenbyte (nur bei KL60xx-Mode: 1 or 2).

**Handshake** : Typ des verwendeten Handshakes soweit durch die serielle Hardware unterstützt (nur bei KL60xx-Mode: HANDSHAKE\_NONE=0, HANDSHAKE\_RTSCCTS=1, HANDSHAKE\_XONXOFF=2).

**ContinousMode**: Schaltet das kontinuierliche Senden ein, wenn es durch die serielle Hardware unterstützt wird.

Wenn ContinuousMode TRUE ist, werden gesendete Daten erst dann aus der seriellen Hardware abgeschickt, wenn der Hardware-Sendepuffer voll ist. Dadurch wird ein zeitlückenfreies Senden gewährleistet, solange die Datenmenge in der Größenordnung des Hardware-Sendepuffers liegt. Der continuous mode wird nur in besonderen Fällen benötigt, wenn das Endgerät auf Zeitlücken mit einem Timeout reagiert.

**pComIn:** Universeller Pointer auf die Eingangsvariable der Prozessdaten der seriellen Hardware (Datentypen KL6inData, KL6inData5b, PcComInData, EL6inData22B). Der Pointer wird mit der ADR() Funktion zugewiesen.

**pComOut:** Universeller Pointer auf die Ausgangsvariable der Prozessdaten der seriellen Hardware (Datentypen KL6outData, KL6outData5b, PcComOutData, EL6outData22B). Der Pointer wird mit der ADR() Funktion zugewiesen.

**SizeComIn:** Größe des Eingangs-Prozessabbildes der verwendeten seriellen Hardware. Die Größe wird mit der SIZEOF() Funktion ermittelt und zugewiesen.

### Beispiel 1:

Das Beispiel zeigt einen Aufruf in ST. Durch das Setzen der Mode-Variablen kann zwischen zwei Kommunikationswegen umgeschaltet werden.

Bei Mode = SERIALLINEMODE\_PC\_COM\_PORT wird über eine serielle COM-Schnittstelle des PC's und bei Mode = SERIALLINEMODE\_KL6\_5B\_STANDARD über eine KL6001 Busklemme kommuniziert (5 Byte mode).

```
PROGRAM P_SerialComm_HighSpeed
VAR
    fbSerialLineCtrl    : FB_IEC870_SerialLineCtrl;
    Mode                : ComSerialLineMode_t := SERIALLINEMODE_KL6_5B_STANDARD; (* SERIALLINEMODE_PC
    _COM_PORT *)

    serial_in AT%I*     : PcComInData;
    serial_out AT%Q*    : PcComOutData;
    KL6_in AT%I*        : KL6inData5B;
    KL6_out AT%Q*       : KL6outData5B;

    hSerial             : T_HSERIALCTRL;
END_VAR

fbSerialLineCtrl( Mode := Mode,
    Baudrate := 19200,
    NoDatabits := 8,
    Parity := PARITY_EVEN,
    Stopbits := 1,
    Handshake := HANDSHAKE_NONE,
    ContinuousMode := FALSE,
    pComIn := SEL( Mode = SERIALLINEMODE_KL6_5B_STANDARD, ADR( serial_in ), ADR( KL6_in ) ),
    pComOut := SEL( Mode = SERIALLINEMODE_KL6_5B_STANDARD, ADR( serial_out ), ADR( KL6_out ) ),
    SizeComIn := SEL( Mode = SERIALLINEMODE_KL6_5B_STANDARD, SIZEOF( serial_in ), SIZEOF( KL6_in
    ) ),
    hSerial := hSerial );
```

### Beispiel 2:

In diesem Beispiel wird über eine EL6001 kommuniziert (22 Byte mode). Die Konfiguration der EL6001 Klemme (Baudrate, Parity usw.) muss in TwinCAT System Manager durchgeführt werden.

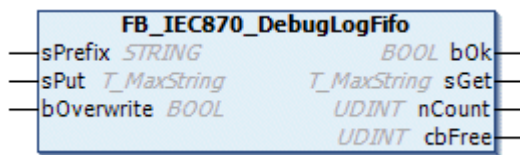
```
PROGRAM P_SerialComm_HighSpeed
VAR
    fbSerialLineCtrl    : FB_IEC870_SerialLineCtrl := ( Mode := SERIALLINEMODE_EL6_22B );
    EL6_in AT%I*        : EL6inData22B;
    EL6_out AT%Q*       : EL6outData22B;
    hSerial             : T_HSERIALCTRL;
END_VAR

fbSerialLineCtrl( pComIn := ADR( EL6_in ), pComOut := ADR( EL6_out ), SizeComIn := SIZEOF( EL6_in ),
    hSerial := hSerial );
```

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

5.1.5 FB\_IEC870\_DebugLogFifo



Dieser Funktionsbaustein ist ein Fifo für Status-/Debug-/Fehler-Meldungen. In der Standardeinstellung wird der älteste Eintrag immer überschrieben. Der Fifo hat eine konstante interne Puffergröße. Die Größe wird durch die Konstante: MAX\_IEC870\_DEBUGLOG\_FIFOSIZE bestimmt (default: 10000 Bytes).

Der Funktionsbaustein besitzt folgende Aktionen:

- **A\_Add** (fügt eine neue Meldung dem Fifo hinzu);
- **A\_Remove** (entfernt die älteste Meldung aus dem Fifo);
- **A\_Reset** (löscht alle Meldungen, setzt den Fifo zurück);
- **A\_LogHint** (fügt eine neue Meldung dem Fifo hinzu und schreibt diese Meldung als Hinweis zusätzlich in das TwinCAT XAE->"Error List" Fenster);
- **A\_LogWarning** (fügt eine neue Meldung dem Fifo hinzu und schreibt diese Meldung als Warnung zusätzlich in das TwinCAT XAE->"Error List" Fenster);
- **A\_LogError** (fügt eine neue Meldung dem Fifo hinzu und schreibt diese Meldung als Fehler zusätzlich in das TwinCAT XAE->"Error List" Fenster);

VAR\_INPUT

```
VAR_INPUT
  sPrefix      : STRING;
  sPut         : T_MaxString;
  bOverwrite   : BOOL := TRUE;
END_VAR
```

**sPrefix:** Zusätzlicher Prefix der an die Meldung vorangestellt wird.

**sPut :** Meldung die dem Fifo hinzugefügt werden soll.

**bOverwrite :** TRUE => die ältesten Einträge werden überschrieben, FALSE => die ältesten Einträge werden nicht überschrieben.

VAR\_OUTPUT

```
VAR_OUTPUT
  bOk         : BOOL;
  sGet        : T_MaxString;
  nCount      : UDINT;
  cbFree      : UDINT;
END_VAR
```

**bOk:** Diese Variable wird TRUE, wenn ein neuer Eintrag erfolgreich hinzugefügt oder aus dem Fifo entfernt wurde. Der Wert bleibt FALSE beim Pufferüberlauf und wenn der älteste Eintrag nicht überschrieben werden konnte.

**sGet:** Meldung, die aus dem Fifo entfernt wurde.

**nCount:** Liefert die aktuelle Anzahl Fifo-Einträge.

**cbFree :** Anzahl der freien Fifo-Datenbytes.

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.1.6 FB\_IEC870\_5\_101ErrorFifo



IEC 60870-5-10x Fehler-Fifo. Der älteste Eintrag wird immer überschrieben. Der Fifo hat eine konstante Größe. Die Größe wird durch die Konstante: `IEC870_MAX_ERROR_FIFO_SIZE` bestimmt (default: 10 Elemente).

Der Funktionsbaustein besitzt drei Aktionen:

- **AddError** (fügt eine neue Fehlermeldung dem Fifo hinzu);
- **RemoveError** (entfernt die älteste Fehlermeldung aus dem Fifo);
- **Reset** (löscht alle Fehlermeldungen, setzt den Fifo zurück);

Im Normalfall werden die Fehlermeldungen durch die internen IEC 60870-5-10x Gerätefunktionen dem Fifo hinzugefügt. Die SPS-Applikation kann diese Fehlermeldungen durch den Aufruf der Aktion: **RemoveError** auslesen und auswerten.

### VAR\_INPUT

```
VAR_INPUT
  putError : ST_IEC870_5_101ErrorFifoEntry;
  putObj   : ST_IEC870_5_101AObj;
END_VAR
```

**putError:** Fehlermeldung [► 490], die dem Fifo hinzugefügt werden soll.

**putObj :** Zusätzliche Information zum Applikationsobjekt (Datenpunkt) auf den sich die Fehlermeldung bezieht. Diese Information ist optional und nicht immer vorhanden.

### VAR\_OUTPUT

```
VAR_OUTPUT
  getError: ST_IEC870_5_101ErrorFifoEntry;
  nErrors : UDINT;
  bOk     : BOOL;
  getObj  : ST_IEC870_5_101AObj;
END_VAR
```

**getError:** Fehlermeldung [► 490], die aus dem Fifo entfernt wurde.

**nErrors:** Liefert die aktuelle Anzahl Fifo-Einträge (Fehlermeldungen im Fifo).

**bOk:** Diese Variable wird TRUE, wenn ein neuer Eintrag erfolgreich hinzugefügt oder aus dem Fifo entfernt wurde.

**getObj :** Zusätzliche Information zum Applikationsobjekt (Datenpunkt) auf den sich die Fehlermeldung bezieht. Diese Information ist optional und nicht immer vorhanden.

**Beispiel in ST:**

Im folgenden ST-Beispiel wird der Gerätefehler-Fifo ausgelesen und die registrierten Fehler ins TwinCAT XAE->"Error List" geschrieben.

```
PROGRAM P_LogErrors
VAR_IN_OUT
    fbErrors : FB_IEC870_5_101ErrorFifo;
END_VAR

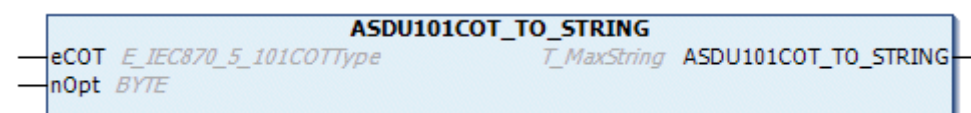
REPEAT
    fbErrors.RemoveError();
    IF fbErrors.bOk THEN
        ADSLOGSTR( ADSLOG_MSGTYPE_ERROR OR ADSLOG_MSGTYPE_LOG, 'IEC 60870-5-10x device error: 0x%s',
            DWORD_TO_HEXSTR( fbErrors.getError.nErrId, 8, FALSE) );
    END_IF
UNTIL NOT fbErrors.bOk
END_REPEAT
```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.2 Funktionen

### 5.2.1 ASDU101COT\_TO\_STRING



Die Funktion konvertiert die IEC 60870-5-101/104 Übertragungsursache in einen String (für Debug/Fehlersuche).

**Syntax**

```
FUNCTION ASDU101COT_TO_STRING: T_MaxString
VAR_INPUT
    eCOT : E_IEC870_101COTType;
    nOpt : BYTE;
END_VAR
```

**Eingänge**

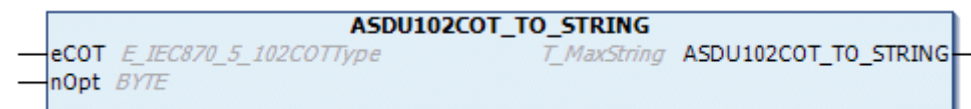
**eCOT:** Übertragungsursache [[▶ 509](#)].

**nOpt:** Optionen. Bit 0 := FALSE => Exakte Typbezeichnung, Bit 0 := TRUE => Kurze Typbezeichnung. Bit 1 := TRUE => Numerischer hexadezimaler/dezimaler Wert, Bit 2 := TRUE => Zusätzlicher Beschreibungstext.

## Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.2.2 ASDU102COT\_TO\_STRING



Die Funktion konvertiert die IEC 60870-5-102 Übertragungsursache in einen String (für Debug/Fehlersuche).

## Syntax

```

FUNCTION ASDU102COT_TO_STRING: T_MaxString
VAR_INPUT
    eCOT : E_IEC870_102COTType;
    nOpt : BYTE;
END_VAR

```

## Eingänge

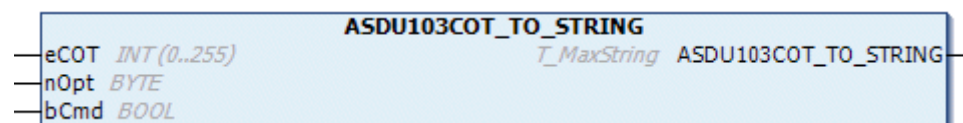
**eCOT:** Übertragungsursache [► 512].

**nOpt:** Optionen. Bit 0 := FALSE => Exakte Typbezeichnung, Bit 0 := TRUE => Kurze Typbezeichnung. Bit 1 := TRUE => Numerischer hexadezimaler/dezimaler Wert, Bit 2 := TRUE => Zusätzlicher Beschreibungstext.

## Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.2.3 ASDU103COT\_TO\_STRING



Die Funktion konvertiert die IEC 60870-5-103 Übertragungsursache in einen String (für Debug/Fehlersuche).

**Syntax**

```
FUNCTION ASDU103COT_TO_STRING: T_MaxString
VAR_INPUT
    eCOT : INT(0..255);
    nOpt : BYTE;
    bCmd : BOOL;
END_VAR
```

**Eingänge**

**eCOT:** Übertragungsursache in Überwachungsrichtung [▶ 513] oder in Steuerungsrichtung [▶ 514].

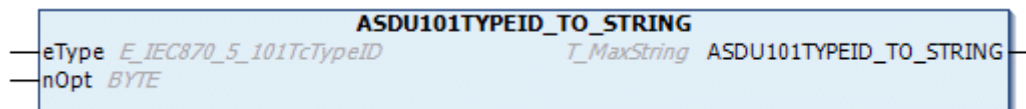
**nOpt:** Optionen. Bit 0 := FALSE => Exakte Typbezeichnung, Bit 0 := TRUE => Kurze Typbezeichnung. Bit 1 := TRUE => Numerischer hexadezimaler/dezimaler Wert, Bit 2 := TRUE => Zusätzlicher Beschreibungstext.

**bCmd:** FALSE=>Überwachungsrichtung, TRUE =>Steuerungsrichtung.

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uutilities (System)

**5.2.4 ASDU101TYPEID\_TO\_STRING**



Die Funktion konvertiert die IEC 60870-5-101/104 ASDU-Typbezeichnung in einen String (für Debug/Fehlersuche).

**Syntax**

```
FUNCTION ASDU101TYPEID_TO_STRING: T_MaxString
VAR_INPUT
    eType : E_IEC870_101TcTypeID;
    nOpt : BYTE;
END_VAR
```

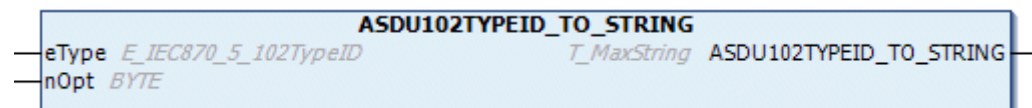
**Eingänge**

**eCOT:** ASDU-Typbezeichnung. [▶ 500]

**nOpt:** Optionen. Bit 0 := FALSE => Exakte Typbezeichnung, Bit 0 := TRUE => Kurze Typbezeichnung. Bit 1 := TRUE => Numerischer hexadezimaler/dezimaler Wert, Bit 2 := TRUE => Zusätzlicher Beschreibungstext.

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.2.5 ASDU102TYPEID\_TO\_STRING**

Die Funktion konvertiert die IEC 60870-5-102 ASDU-Typbezeichnung in einen String (für Debug/Fehlersuche).

**Syntax**

```

FUNCTION ASDU102TYPEID_TO_STRING: T_MaxString
VAR_INPUT
    eType : E_IEC870_102TcTypeID;
    nOpt  : BYTE;
END_VAR

```

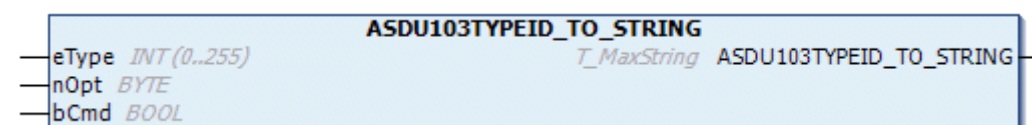
**🚪 Eingänge**

**eCOT:** ASDU-Typbezeichnung [[▶ 503](#)].

**nOpt:** Optionen. Bit 0 := FALSE => Exakte Typbezeichnung, Bit 0 := TRUE => Kurze Typbezeichnung. Bit 1 := TRUE => Numerischer hexadezimaler/dezimaler Wert, Bit 2 := TRUE => Zusätzlicher Beschreibungstext.

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.2.6 ASDU103TYPEID\_TO\_STRING**

Die Funktion konvertiert die IEC 60870-5-103 ASDU-Typbezeichnung in einen String (für Debug/Fehlersuche).



**Syntax**

```
FUNCTION ASDU103TYPEID_TO_STRING: T_MaxString
VAR_INPUT
    eType : INT(0..255);
    nOpt  : BYTE;
    bCmd  : BOOL;
END_VAR
```

**Eingänge**

**eCOT:** ASDU-Typbezeichnung in Überwachungsrichtung [▶ 505] oder in Steuerungsrichtung [▶ 505].

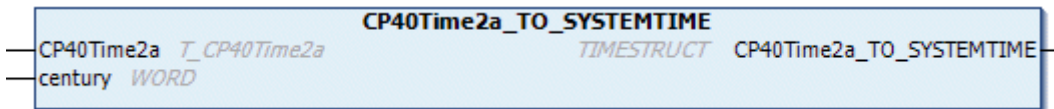
**nOpt:** Optionen. Bit 0 := FALSE => Exakte Typbezeichnung, Bit 1 := TRUE => Kurze Typbezeichnung. Bit 2 := TRUE => Numerischer hexadezimaler/dezimaler Wert, Bit 3 := TRUE => Zusätzlicher Beschreibungstext.

**bCmd:** FALSE => Überwachungsrichtung, TRUE => Steuerungsrichtung.

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uutilities (System)

**5.2.7 CP40Time2a\_TO\_SYSTEMTIME**



Die Funktion konvertiert das CP40Time2a-Zeitformat [▶ 533] in Windows-Systemzeit-Format. Das SU-Flag wird nicht benutzt.

**Syntax**

```
FUNCTION CP40Time2a_TO_SYSTEMTIME: Timestruct
VAR_INPUT
    CP40Time2a : T_CP40Time2a;
    century     : WORD;
END_VAR
```

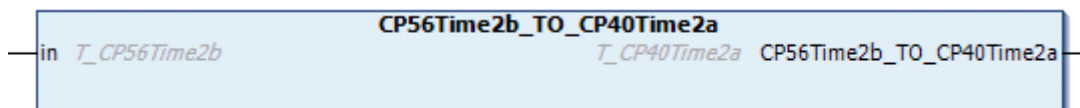
**Eingänge**

**CP40Time2a:** Die zu konvertierende Zeit im CP40Time2a-Format.

**century:** Das Jahrhundert ( z.B. 20 für das Jahr 2005 ). Diese Information ist im CP40Time2a-Format nicht enthalten und aus diesem Grund muss sie zusätzlich angegeben werden.

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.2.8 CP56Time2b\_TO\_CP40Time2a**

Die Funktion konvertiert das CP56Time2b-Zeitformat [► 532] in CP40Time2a-Zeitformat [► 533].

**Syntax**

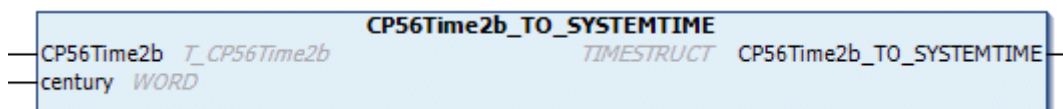
```
FUNCTION CP56Time2b_TO_CP40Time2a: T_CP40Time2a
VAR_INPUT
    in : T_CP56Time2b;
END_VAR
```

**🚩 Eingänge**

**in:** Die zu konvertierende Zeit im CP56Time2b-Format.

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.2.9 CP56Time2b\_TO\_SYSTEMTIME**

Die Funktion konvertiert das CP56Time2b-Zeitformat [► 532] in Windows-System-Zeitformat. Das SU-Flag wird nicht benutzt.

**Syntax**

```
FUNCTION CP56Time2b_TO_SYSTEMTIME: TIMESTRUCT
VAR_INPUT
    CP56Time2b : T_CP56Time2b;
    century    : WORD;
END_VAR
```

**Eingänge**

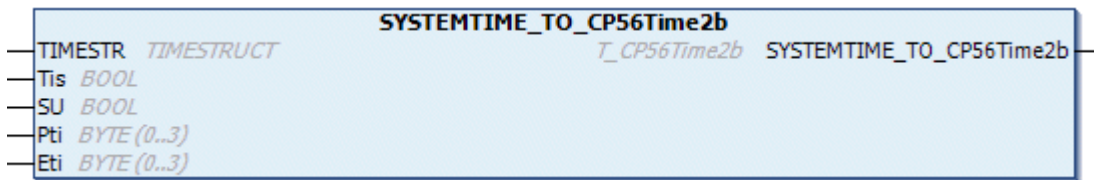
**CP56Time2b:** Die zu konvertierende Zeit im CP56Time2b-Format.

**century:** Das Jahrhundert ( z.B. 20 für das Jahr 2005 ). Diese Information ist im CP56Time2b-Format nicht enthalten und aus diesem Grund muss sie zusätzlich angegeben werden.

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.2.10 SYSTEMTIME\_TO\_CP56Time2b**



Die Funktion konvertiert das Windows-System-Zeitformat in das CP56Time2b-Zeitformat [► 532]. Alle reservierten Bits sind Null.

**Syntax**

```
FUNCTION SYSTEMTIME_TO_CP56Time2b: T_CP56Time2b
VAR_INPUT
    TIMESTR : Timestruct;
    Tis      : Bool;
    SU       : Bool;
    Pti      : Byte(0..3);
    Eti      : Byte(0..3);
END_VAR
```

**Eingänge**

**TIMESTR:** Die zu konvertierende Systemzeit.

**Tis:** Tarifinformation.

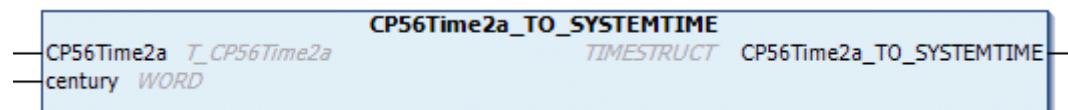
**SU:** Sommer-/Winterzeitformat. Diese Information ist in dem TIMESTR-Format nicht vorhanden und muss zusätzlich angegeben werden. TRUE = Sommerzeit, FALSE = Winterzeit (Normalzeit).

**Pti:** Power Tarifinformation.

**Eti:** Energie Tarifinformation.

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.2.11 CP56Time2a\_TO\_SYSTEMTIME**

Die Funktion konvertiert das CP56Time2a-Zeitformat [► 531] in Windows-Systemzeit-Format. Das SU-Flag wird nicht benutzt.

**Syntax**

```
FUNCTION CP56Time2a_TO_SYSTEMTIME: Timestruct
VAR_INPUT
    CP56Time2a : T_CP56Time2a;
    century    : WORD;
END_VAR
```

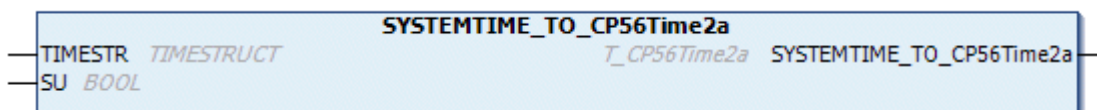
**📌 Eingänge**

**CP56Time2a:** Die zu konvertierende Zeit im CP56Time2a-Format.

**century:** Das Jahrhundert ( z.B. 20 für das Jahr 2005 ). Diese Information ist im CP56Time2a-Format nicht enthalten und aus diesem Grund muss sie zusätzlich angegeben werden.

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.2.12 SYSTEMTIME\_TO\_CP56Time2a**

Die Funktion konvertiert das Windows-System-Zeitformat in das CP56Time2a-Zeitformat [► 531]. Alle reservierten Bits sind Null.

**Syntax**

```
FUNCTION SYSTEMTIME_TO_CP56Time2a: T_CP56Time2a
VAR_INPUT
    TIMESTR : TIMESTRUCT;
    SU      : BOOL;
END_VAR
```

 **Eingänge**

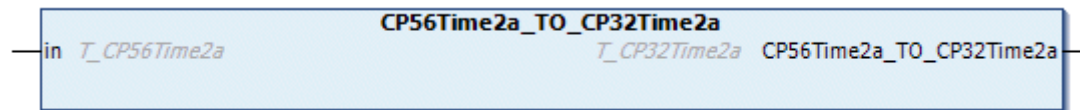
**TIMESTR:** Die zu konvertierende Systemzeit.

**SU:** Sommer-/Winterzeitformat. Diese Information ist in dem TIMESTR-Format nicht vorhanden und muss zusätzlich angegeben werden. TRUE = Sommerzeit, FALSE = Winterzeit (Normalzeit).

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.2.13 CP56Time2a\_TO\_CP32Time2a**



Die Funktion konvertiert das CP56Time2a-Zeitformat [▶ 531] in CP32Time2a-Zeitformat [▶ 533].

**Syntax**

```
FUNCTION CP56Time2a_TO_CP32Time2a: T_CP32Time2a
VAR_INPUT
    in : T_CP56Time2a;
END_VAR
```

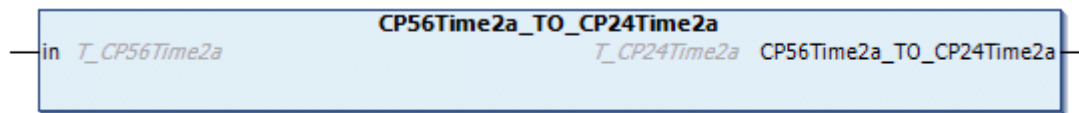
 **Eingänge**

**in:** Die zu konvertierende Zeit im CP56Time2a-Format.

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.2.14 CP56Time2a\_TO\_CP24Time2a



Die Funktion konvertiert das CP56Time2a-Zeitformat [► 531] in CP24Time2a-Zeitformat [► 534].

### Syntax

```
FUNCTION CP56Time2a_TO_CP24Time2a: T_CP24Time2a
VAR_INPUT
    in : T_CP56Time2a;
END_VAR
```

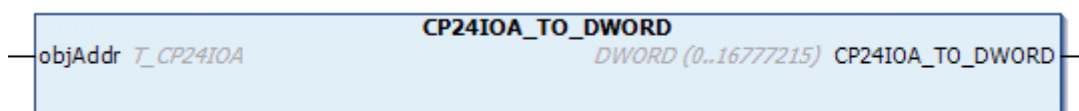
### 📌 Eingänge

**in:** Die zu konvertierende Zeit im CP56Time2a-Format.

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.2.15 CP24IOA\_TO\_DWORD



Die Funktion generiert eine strukturierte TwinCAT-Objektadresse (3 Oktete). Siehe auch: DWORD TO\_CP24IOA [► 431].

### Syntax

```
FUNCTION CP24IOA_TO_DWORD: DWORD
VAR_INPUT
    objAddr : T_CP24IOA;
END_VAR
```

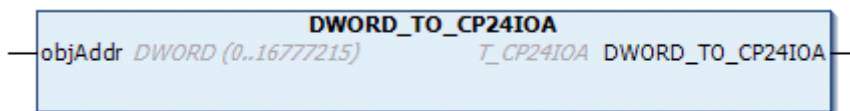
### 📌 Eingänge

**objAddr:** Parameter der strukturierten TwinCAT-Objektadresse [► 535].

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.2.16 DWORD\_TO\_CP24IOA



Die Funktion konvertiert eine strukturierte TwinCAT-Objektadresse [► 535] in einzelne Adressparameter. Siehe auch: CP24IOA TO DWORD [► 430].

Syntax

```
FUNCTION DWORD_TO_CP24IOA: T_CP24IOA
VAR_INPUT
    objAddr : DWORD(0..16777215);
END_VAR
```

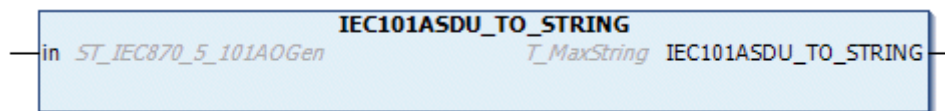
Eingänge

**objAddr:** Strukturierte TwinCAT-Objektadresse (3 oktete).

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.2.17 IEC101ASDU\_TO\_STRING



Die Funktion konvertiert die Informationen einer IEC 60870-5-101/104 ASDU in einen String (für Debugzwecke/Fehlersuche).

Syntax

```
FUNCTION IEC101ASDU_TO_STRING: T_MaxString
VAR_INPUT
    in : ST_IEC870_5_101AOGen;
END_VAR
```

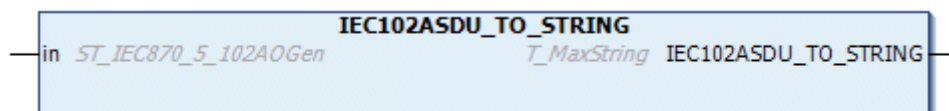
### 🚩 Eingänge

**in:** Strukturierte Variable mit der zu konvertierenden ASDU.

#### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.2.18 IEC102ASDU\_TO\_STRING



Die Funktion konvertiert die Informationen einer IEC 60870-5-102 ASDU in einen String (für Debugzwecke/ Fehlersuche).

#### Syntax

```

FUNCTION IEC102ASDU_TO_STRING: T_MaxString
VAR_INPUT
    in : ST_IEC870_5_102A0Gen;
END_VAR
  
```

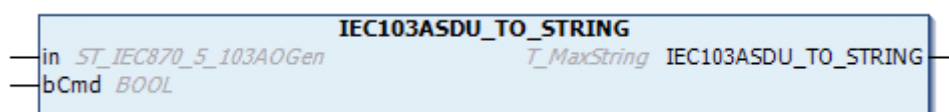
### 🚩 Eingänge

**in:** Strukturierte Variable mit der zu konvertierenden ASDU.

#### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.2.19 IEC103ASDU\_TO\_STRING



Die Funktion konvertiert die Informationen einer IEC 60870-5-103 ASDU in einen String (für Debugzwecke/ Fehlersuche).



**Syntax**

```
FUNCTION IEC103ASDU_TO_STRING: T_MaxString
VAR_INPUT
    in      : ST_IEC870_5_103A0Gen;
    bCmd    : BOOL;
END_VAR
```

**Eingänge**

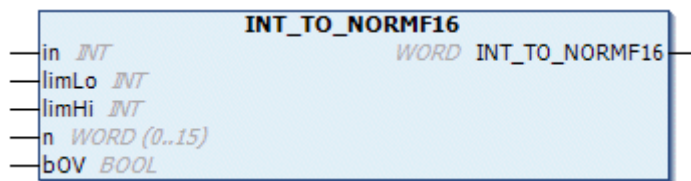
**in:** Strukturierte Variable mit der zu konvertierenden ASDU.

**bCmd:** FALSE => ASDU in Überwachungsrichtung, TRUE => ASDU in Steuerungsrichtung.

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.2.20 INT\_TO\_NORMF16**



Diese Funktion normalisiert einen 16 Bit integer Wert und konvertiert ihn in einen NVA-Wert (normalisierter Wert) im Festkommazahlformat.

**Syntax**

```
FUNCTION INT_TO_NORMF16: WORD
VAR_INPUT
    in      : INT;
    limLo   : INT;
    limHi   : INT;
    n       : WORD(0..15) := 15;
    bOV     : BOOL;
END_VAR
```

**Eingänge**

**in:** Wert der konvertiert werden soll.

**limLo:** Min. Wertgrenze.

**limHi:** Max. Wertgrenze.

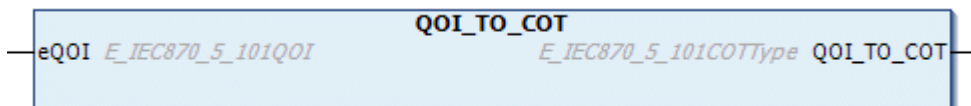
**n:** Anzahl der Nachkommastellen der Festkommazahl.

**bOV:** TRUE => Wert-Bereichsüberschreitung oder -Unterschreitung.

## Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.2.21 QOI\_TO\_COT



Die Funktion konvertiert den "Qualifier" des Stationsabfragebefehls in die zu erwartende Übertragungsursache für die abgefragten Daten.

## Syntax

```
FUNCTION QOI_TO_COT: E_IEC870_5_101COTType
VAR_INPUT
    eQOI : E_IEC870_5_101QOI;
END_VAR
```

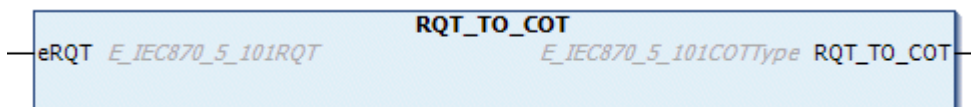
## Eingänge

**eQOI:** Qualifier des Stationsabfragebefehls [► 521].

## Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.2.22 RQT\_TO\_COT



Die Funktion konvertiert den "Qualifier" des Zählerabfragebefehls in die zu erwartende Übertragungsursache für die abgefragten Daten.

## Syntax

```
FUNCTION RQT_TO_COT: E_IEC870_5_101COTType
VAR_INPUT
    eRQT : E_IEC870_5_101RQT;
END_VAR
```

 **Eingänge**

**eRQT:** Qualifier des Zählerabfragebefehls [▶ 523].

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.2.23 F\_ieclnitAOEntry**



Die Funktion F\_ieclnitAOEntry konfiguriert die Applikationsobjekte (Single Points, Double Points, Measured Values...) in der Applikationsdatenbank als Linear-Tabelleneinträge. Das zu konfigurierende Tabellenelement (Arrayelement) muss als VAR\_IN\_OUT-Funktionsparameter an die Funktion übergeben werden.

**Syntax**

```

FUNCTION F_ieclnitAOEntry: UDINT
VAR_INPUT
    eType      : E_IEC870_5_101TcTypeID := ASDU_TYPEUNDEF;
    objAddr    : DWORD := 0;
    group      : DWORD := 0;
    multiplier  : BYTE := 0;
    ioMapType  : E_IEC870_5_101IOMappingType := MAP_AREA_NONE;
    byteOffs   : UXINT := 0;
    bitOffs    : UDINT := 0;
END_VAR
VAR_IN_OUT
    dbEntry    : ST_IEC870_5_101AODBEntry;
END_VAR
    
```

 **Eingänge**

**eType:** Applikationsobjekt Typ [▶ 500], ASDU identifier (z.B.: M\_SP\_NA\_1 für Single-Point oder M\_DP\_NA\_1 für Double-Point ).

**objAddr:** Objektadresse, frei wählbar.

**group:** Object-Group-Konfigurationsflags. Hier finden sie die Beschreibung aller Group-Flags [▶ 554]. Die Flags können mit OR-Verknüpfung kombiniert werden. Nicht alle Kombinationen sind aber sinnvoll!

**multiplier:** Basis-Zykluszeit-Multiplikator für zyklische/periodische Datenübertragung. 0 = Deaktiviert. Die Basis-Zykluszeit kann über den *tPerCyclicBase*-Parameter in den Systemparametern [▶ 487] konfiguriert werden.

**ioMapType:** TwinCAT SPS-Prozessdatenbereich [▶ 506]. Dieser Parameter legt fest wie die TwinCAT SPS und IEC-Applikationsobjekt Prozessdaten gemappt werden sollen.

**byteOffs:** TwinCAT SPS-Prozessdaten-Byte-Offset (x86 Plattform => 32 Bit, x64 Plattform => 64 Bit).

**bitOffs:** TwinCAT SPS-Prozessdaten-Bit-Offset.

## Eing-/Ausgänge

**dbEntry:** Das zu konfigurierende [Tabellenelement \[► 482\]](#) (Arrayelement).

## 🚩 Rückgabewert

Rückgabeparameter	Beschreibung
0	Kein Fehler.
<> 0	Fehler: IEC60870-5-10x Fehlercode <a href="#">[► 557]</a>

## Beispiel in ST

Im folgenden Beispiel werden einige Datenpunkte als Linear-Tabelleneinträge konfiguriert.

eType	objAddr	group	multiplier	ioMapType	byteOffs	bitOffs
M_SP_NA_1	100	IEC870_GRP_INRO1	0	MAP_AREA_MEMORY	100	0
M_SP_TB_1	101	IEC870_GRP_INRO1	0	MAP_AREA_MEMORY	101	0
M_DP_NA_1	200	IEC870_GRP_INROGEN	0	MAP_AREA_DATA	200	0
M_IT_NA_1	800	IEC870_GRP_REQCOGEN	0	MAP_AREA_MEMORY	800	0

```

VAR_GLOBAL CONSTANT
    MAX_TABLE_IDX : INT := 49;
END_VAR

PROGRAM P_iecInitAOEntry
VAR_IN_OUT
    AODB : ARRAY[1..MAX_TABLE_IDX] OF ST_IEC870_5_101AODBEntry;
END_VAR

VAR
    bInit : BOOL := TRUE;
    nError : UDINT;
END_VAR

IF bInit THEN
    bInit := FALSE;
    IF ( nError := F_iecInitAOEntry ( eType := M_SP_NA_1,
        objAddr := 100,
        group := IEC870_GRP_INRO1,
        multiplier := 0,
        ioMapType := MAP_AREA_MEMORY,
        byteOffs := 100,
        bitOffs := 0,
        dbEntry := AODB[1] ) ) <> 0 THEN
        ADSLOGSTR( ADSLOG_MSGTYPE_ERROR OR ADSLOG_MSGTYPE_LOG, 'F_iecInitAOEntry() error: %s',
            DWORD_TO_HEXSTR( nError, 8, FALSE ) );
    END_IF

    IF ( nError := F_iecInitAOEntry ( eType := M_SP_TB_1,
        objAddr := 101,
        group := IEC870_GRP_INRO1,
        multiplier := 0,
        ioMapType := MAP_AREA_MEMORY,
        byteOffs := 101,
        bitOffs := 0,
        dbEntry := AODB[2] ) ) <> 0 THEN
        ADSLOGSTR( ADSLOG_MSGTYPE_ERROR OR ADSLOG_MSGTYPE_LOG, 'F_iecInitAOEntry() error: %s',
            DWORD_TO_HEXSTR( nError, 8, FALSE ) );
    END_IF

    IF ( nError := F_iecInitAOEntry ( eType := M_DP_NA_1,
        objAddr := 200,
        group := IEC870_GRP_INROGEN,

```

```

        multipliert := 0,
        ioMapType := MAP_AREA_DATA,
        byteOffs := 200,
        bitOffs := 0,
        dbEntry := AODB[3] ) ) <> 0 THEN
    ADSLOGSTR( ADSLOG_MSGTYPE_ERROR OR ADSLOG_MSGTYPE_LOG, 'F_iecInitAOEntry() error: %s',
        DWORD_TO_HEXSTR( nError, 8, FALSE ) );
END_IF

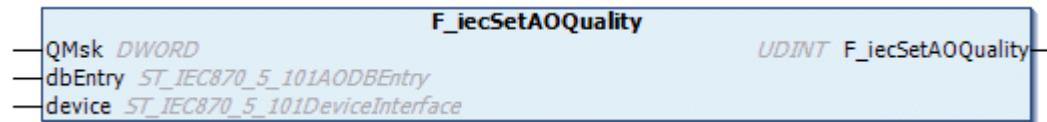
IF ( nError := F_iecInitAOEntry ( eType := M_IT_NA_1,
    objAddr := 800,
    group := IEC870_GRP_REQCOGEN,
    multipliert := 0,
    ioMapType := MAP_AREA_MEMORY,
    byteOffs := 800,
    bitOffs := 0,
    dbEntry := AODB[4] ) ) <> 0 THEN
    ADSLOGSTR( ADSLOG_MSGTYPE_ERROR OR ADSLOG_MSGTYPE_LOG, 'F_iecInitAOEntry() error: %s',
        DWORD_TO_HEXSTR( nError, 8, FALSE ) );
END_IF
END_IF

```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.2.24 F\_iecSetAOQuality**



Mit dieser Funktion können Quality-Flags eines Applikationsobjekts auf einen bestimmten Wert gesetzt/ zurückgesetzt werden.

**Syntax**

```

FUNCTION F_iecSetAOQuality: UDINT
VAR_INPUT
    QMsk : DWORD;
END_VAR
VAR_IN_OUT
    dbEntry : ST_IEC870_5_101AODBEntry;
    device : ST_IEC870_5_101DeviceInterface;
END_VAR

```

**Eingänge**

**QMsk:** Quality-Flags. Die Quality-Flags können mit OR-Verknüpfung kombiniert werden. Bei QMsk = Null werden keine Flags gesetzt/zurückgesetzt. Eine Liste der verfügbaren Quality-Flags finden Sie hier: [Quality-Flags \[► 556\]](#).

**Ein-/Ausgänge**

**dbEntry:** [Applikationsobjekt \[► 482\]](#) dessen Status der Quality-Flags gesetzt werden soll.

**device:** [Kommunikationsschnittstelle \[► 489\]](#) des IEC-Geräts.

## 🚩 Rückgabewert

Rückgabeparameter	Bedeutung
0	Kein Fehler.
<> 0	Fehler: <a href="#">IEC60870-5-10x Fehlercode</a>   <a href="#">557</a>

## Beispiel in ST

```

VAR_GLOBAL CONSTANT
    MAX_TABLE_IDX : INT := 49;
END_VAR

PROGRAM P_iecGetAOQuality
VAR_IN_OUT
    AODB          : ARRAY[1..MAX_TABLE_IDX] OF ST_IEC870_5_101AODBEntry;
END_VAR

VAR
    server1       : FB_IEC870_5_104Slave;

    bBlock        : BOOL;
    bUnblock      : BOOL;
    bIsBlocked    : BOOL;

    qualityFlags  : DWORD;
END_VAR

IF bBlock THEN
    bBlock := FALSE;
    F_iecSetAOQuality( IECQ_BL_ON, AODB[1], server1.system.device );
END_IF

IF bUnblock THEN
    bUnblock := FALSE;
    F_iecSetAOQuality( IECQ_BL_OFF, AODB[1], server1.system.device );
END_IF

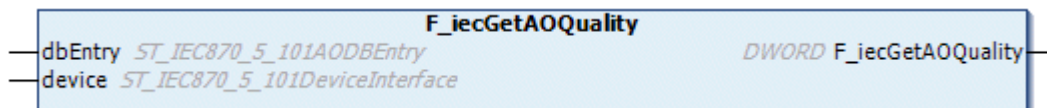
qualityFlags := F_iecGetAOQuality ( AODB[1], server1.system.device );
bIsBlocked := SEL( (qualityFlags AND IECQ_BL_ON) = IECQ_BL_ON, FALSE, TRUE );

```

## Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.2.25 F\_iecGetAOQuality



Mit der Funktion kann der Status der Quality-Flags eines Applikationsobjekts gelesen werden.

### Syntax

```

FUNCTION F_iecGetAOQuality: DWORD
VAR_IN_OUT
    dbEntry : ST_IEC870_5_101AODBEntry;
    device  : ST_IEC870_5_101DeviceInterface;
END_VAR

```

**Ein-/Ausgänge**

**dbEntry:** Applikationsobjekt [[▶ 482](#)] dessen Quality-Flags gelesen werden sollen.

**device:** Kommunikationsschnittstelle [[▶ 489](#)] zum IEC-Gerät.

 **Rückgabewert**

Rückgabeparameter	Bedeutung
0	Fehler, keine Quality-Flags für dieses Applikationsobjekt verfügbar.
<> 0	Kein Fehler. Der Rückgabeparameter liefert den Status der Quality-Flags. Eine Liste der verfügbaren Quality-Flags finden Sie hier: <a href="#">Quality-Flags</a> [ <a href="#">▶ 556</a> ].

**Beispiel in ST**

```

VAR_GLOBAL CONSTANT
    MAX_TABLE_IDX : INT := 49;
END_VAR

PROGRAM P_iecGetAOQuality
VAR_IN_OUT
    AODB          : ARRAY[1..MAX_TABLE_IDX] OF ST_IEC870_5_101AODBEntry;
END_VAR

VAR
    server1       : FB_IEC870_5_104Slave;

    bBlock        : BOOL;
    bUnblock      : BOOL;
    bIsBlocked    : BOOL;

    qualityFlags  : DWORD;
END_VAR

IF bBlock THEN
    bBlock := FALSE;
    F_iecSetAOQuality( IECQ_BL_ON, AODB[1], server1.system.device );
END_IF

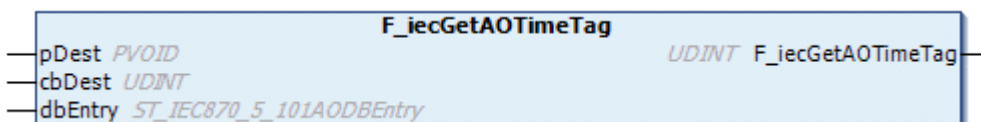
IF bUnblock THEN
    bUnblock := FALSE;
    F_iecSetAOQuality( IECQ_BL_OFF, AODB[1], server1.system.device );
END_IF

qualityFlags := F_iecGetAOQuality ( AODB[1], server1.system.device );
bIsBlocked := SEL( (qualityFlags AND IECQ_BL_ON ) = IECQ_BL_ON, FALSE, TRUE );
    
```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.2.26 F\_iecGetAOTimeTag**



Mit dieser Funktion kann der aktuelle Zeitstempel eines Applikationsobjekts in einen Bytepuffer gelesen werden.

### Syntax

```
FUNCTION F_iecGetAOTimeTag: UDINT
VAR_INPUT
    pDest      : PVOID;
    cbDest     : UDINT;
END_VAR
VAR_IN_OUT
    dbEntry   : ST_IEC870_5_101AODBEntry;
END_VAR
```

### Eingänge

**pDest:** Zielpufferadresse.

**cbDest:** Bytegröße des Zielpuffers.

### Ein-/Ausgänge

**dbEntry:** Applikationsobjekt [[▶ 482](#)] dessen Zeitstempel gelesen werden soll.

### Rückgabewert

Rückgabeparameter	Bedeutung
0	Fehler, das Applikationsobjekt besitzt keinen Zeitstempel.
<> 0	Anzahl der erfolgreich kopierten Zeitstempeldatenbytes. Bei einem CP24Time2a-Zeitstempelformat sind es z.B. 3 Bytes und bei einem CP56Time2a-Zeitstempelformat sind es 7 Bytes.

### Beispiel in ST

```
PROGRAM P_iecGetAOTimeTag
VAR_IN_OUT
    AODB      : ARRAY[1..MAX_TABLE_IDX] OF ST_IEC870_5_101AODBEntry;
END_VAR

VAR
    bGet      : BOOL := TRUE;
    index     : DINT;
    cbTagSize : UDINT;
    tTag      : T_CP56Time2a;
END_VAR

IF bGet THEN
    bGet      := FALSE;

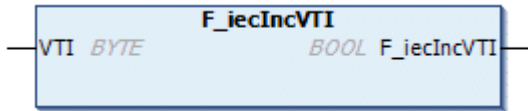
    FOR index := 1 TO MAX_TABLE_IDX BY 1 DO
        cbTagSize := F_iecGetAOTimeTag ( ADR( tTag ), SIZEOF( tTag ), AODB[index] );
    END_FOR
END_IF
```

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)



### 5.2.27 F\_iecIncVTI



Diese Funktion inkrementiert den INT7 regelnden Schrittwert. Das transiente Bit wird nicht verändert.

#### Syntax

```
FUNCTION F_iecIncVTI: BOOL
VAR_IN_OUT
    VTI : BYTE;
END_VAR
```

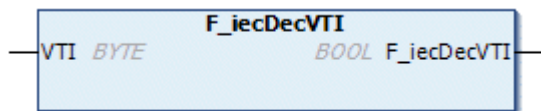
#### Ein-/Ausgänge

**VTI:** Das zu inkrementierende Byte.

#### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.2.28 F\_iecDecVTI



Diese Funktion dekrementiert den INT7 regelnden Schrittwert. Das transiente Bit wird nicht verändert.

#### Syntax

```
FUNCTION F_iecDecVTI: BOOL
VAR_IN_OUT
    VTI : BYTE;
END_VAR
```

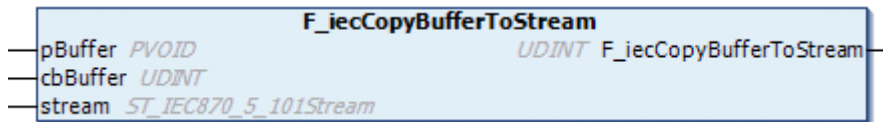
#### Ein-/Ausgänge

**VTI:** Das zu dekrementierende Byte.

#### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.2.29 F\_iecCopyBufferToStream



Diese Funktion kopiert Datenbytes von einer externen Puffervariablen in die *stream*-Variable. Der Speicherinhalt der *stream*-Variablen wird vergrößert. Der Rückgabeparameter der Funktion liefert die Anzahl der erfolgreich kopierten Datenbytes.

### Syntax

```
FUNCTION F_iecCopyBufferToStream: UDINT
VAR_INPUT
    pBuffer : PVOID;
    cbBuffer: UDINT;
END_VAR
VAR_IN_OUT
    stream : ST_IEC870_5_101Stream;
END_VAR
```

### Eingänge

**pBuffer:** Pointer (Adresse) einer externen Puffervariablen.

**cbBuffer:** Anzahl der Datenbytes die von der externen Puffervariablen in die *stream*-Variable kopiert werden sollen.

### Ein-/Ausgänge

**stream:** [Zieldatenpuffer](#) [▶ 485].

### Beispiel in ST

Bei einer steigenden Flanke an der *bTx* werden 2 x 4 Datenbytes der *txBuffer*-Variablen in die *stream*-Variable hineinkopiert.

```
PROGRAM P_iecCopyBufferToStream
VAR
    stream : ST_IEC870_5_101Stream;
    txBuffer : ARRAY[0..3] OF BYTE := [1, 2, 3, 4];
    cbResult : UDINT;
    bTx : BOOL;
END_VAR

IF bTx THEN
    bTx := FALSE;
    cbResult := F_iecResetStream( 0, stream );
    cbResult := F_iecCopyBufferToStream ( ADR( txBuffer ), SIZEOF( txBuffer ), stream );
    cbResult := F_iecCopyBufferToStream ( ADR( txBuffer ), SIZEOF( txBuffer ), stream );
END_IF
```

Speicherdarstellung der *stream*-Variablen vor dem ersten Funktionsaufruf:

length	data												
0	16#00	16#00	16#00	16#00	16#00	16#00	16#00	16#00	16#00	16#00	16#00	...	IEC870 _MAX_ ASDU_ DATA_ BYTE

Speicherdarstellung der *stream*-Variablen nach dem ersten Funktionsaufruf:

length	data												
4	16#01	16#02	16#03	16#04	16#00	16#00	16#00	16#00	16#00	16#00	16#00	...	IEC870 _MAX_ _ASDU_ _DATA_ _BYTE

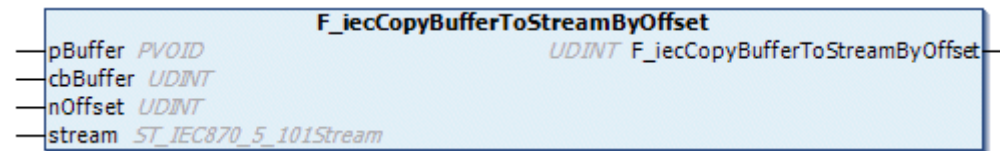
Speicherdarstellung der *stream*-Variablen nach dem zweiten Funktionsaufruf:

length	data											
8	16#01	16#02	16#03	16#04	16#01	16#02	16#03	16#04	16#00	16#00	...	IEC870 _MAX_ _ASDU_ _DATA_ _BYTE

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.2.30 F\_iecCopyBufferToStreamByOffset**



Diese Funktion kopiert Datenbytes von einer externen Puffervariablen in die *stream*-Variable, beginnend ab einer Byteoffsetposition. Der Speicherinhalt der *stream*-Variablen wird vergrößert abhängig von der vorherigen Länge und dem angegebenen Byteoffset. Der Rückgabeparameter der Funktion liefert die Anzahl der erfolgreich kopierten Datenbytes.

**Syntax**

```
FUNCTION F_iecCopyBufferToStreamByOffset: UDINT
VAR_INPUT
    pBuffer : PVOID;
    cbBuffer : UDINT;
    nOffset : UDINT;
END_VAR
VAR_IN_OUT
    stream : ST_IEC870_5_101Stream;
END_VAR
```

**Eingänge**

**pBuffer:** Pointer (Adresse) einer externen Puffervariablen.

**cbBuffer:** Anzahl der Datenbytes die von der externen Puffervariablen in die *stream*-Variable kopiert werden sollen.

**nOffset:** Byteoffsetposition in dem Zieldatenpuffer.

## Ein-/Ausgänge

**stream:** [Zielpuffer \[► 485\]](#).

## Beispiel in ST

Bei einer steigenden Flanke an der *bTx* werden 4 Datenbytes der *txAtOffs*-Variablen in die *stream*-Variable hineinkopiert (ab der Byteoffsetposition: 5).

```
PROGRAM P_iecCopyBufferToStreamByOffset
VAR
  bTx      : BOOL;
  stream   : ST_IEC870_5_101Stream := (length := 8, data := [16#01, 16#02, 16#03, 16#04, 16#05, 16#06, 16#07, 16#08] );
  txAtOffs : DWORD := 16#DDCCBBAA;
  cbResult : UDINT;
END_VAR

IF bTx THEN
  bTx := FALSE;
  cbResult := F_iecCopyBufferToStreamByOffset ( ADR( txAtOffs), SIZEOF(txAtOffs), 5, stream );
  (* stream.length == 9 *)
END_IF
```

Speicherdarstellung der *stream*-Variablen nach dem Programmstart:

length	data											
8	16#01	16#02	16#03	16#04	16#05	16#06	16#07	16#08	16#00	16#00	...	IEC870 MAX_ ASDU_ DATA_ BYTE

Speicherdarstellung der *stream*-Variablen nach dem `F_iecCopyBufferToStreamByOffset`-Funktionsaufruf:

length	data											
9	16#01	16#02	16#03	16#04	16#05	16#AA	16#BB	16#CC	16#DD	16#00	...	IEC870 MAX_ ASDU_ DATA_ BYTE

## Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.2.31 F\_iecCopyStreamToBuffer

F_iecCopyStreamToBuffer	
pBuffer <i>PVOID</i>	UDINT F_iecCopyStreamToBuffer
cbBuffer <i>UDINT</i>	
stream <i>ST_IEC870_5_101Stream</i>	

Diese Funktion kopiert Datenbytes von der *stream*-Variablen in eine externe Puffervariable. Der Speicherinhalt der *stream*-Variablen bleibt unverändert. Der Rückgabeparameter der Funktion liefert die Anzahl der erfolgreich kopierten Datenbytes.

**Syntax**

```
FUNCTION F_iecCopyStreamToBuffer: UDINT
VAR_INPUT
    pBuffer : PVOID;
    cbBuffer: UDINT;
END_VAR
VAR_IN_OUT
    stream : ST_IEC870_5_101Stream;
END_VAR
```

 **Eingänge**

**pBuffer:** Pointer (Adresse) einer externen Puffervariablen.

**cbBuffer:** Maximale Anzahl der Datenbytes die aus der *stream*-Variablen in den externen Puffer kopiert werden sollen.

**Ein-/Ausgänge**

**stream:** [Quelldatenpuffer \[▶ 485\]](#).

**Beispiel in ST**

Bei einer steigenden Flanke am bRx werden die ersten vier Datenbytes der *stream*-Variablen in die *rxBufferA*- und *rxBufferB*-Variable kopiert.

```
PROGRAM P_iecCopyStreamToBuffer
VAR
    stream : ST_IEC870_5_101Stream := ( length := 4, data := [16#01, 16#02, 16#03, 16#04] );
    rxBufferA : ARRAY[0..3] OF BYTE;
    rxBufferB : ARRAY[0..3] OF BYTE;
    cbResult : UDINT;
    bRx : BOOL;
END_VAR
IF bRx THEN
    bRx := FALSE;
    cbResult := F_iecCopyStreamToBuffer ( ADR( rxBufferA ), SIZEOF( rxBufferA ), stream );
    cbResult := F_iecCopyStreamToBuffer ( ADR( rxBufferB ), SIZEOF( rxBufferB ), stream );
END_IF
```

Speicherdarstellung der *stream*-Variablen nach dem Programmstart:

length	data											
4	16#01	16#02	16#03	16#04	...	...	...	...	...	...	...	IEC870 MAX ASDU DATA BYTE

Speicherdarstellung der *stream*-Variablen nach dem ersten und zweiten F\_CopyStreamToBuffer-Funktionsaufruf:

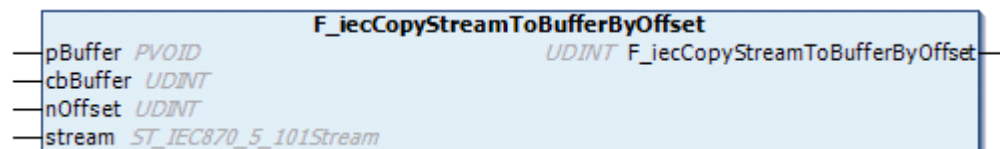
length	data											
4	16#01	16#02	16#03	16#04	...	...	...	...	...	...	...	IEC870 MAX ASDU DATA BYTE

Die *rxBufferA* und *rxBufferB* Array-Variable haben nach dem Aufruf folgenden Wert: [16#01, 16#02, 16#03, 16#04].

## Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.2.32 F\_iecCopyStreamToBufferByOffset



Diese Funktion kopiert Datenbytes von der *stream*-Variablen ab einer Byteoffsetposition in eine externe Puffervariable. Der Speicherinhalt der *stream*-Variablen bleibt unverändert. Der Rückgabeparameter der Funktion liefert die Anzahl der erfolgreich kopierten Datenbytes.

### Syntax

```

FUNCTION F_iecCopyStreamToBufferByOffset: UDINT
VAR_INPUT
    pBuffer : PVOID;
    cbBuffer: UDINT;
    nOffset : UDINT;
END_VAR
VAR_IN_OUT
    stream : ST_IEC870_5_101Stream;
END_VAR
  
```

### Eingänge

**pBuffer:** Pointer (Adresse) einer externen Puffervariablen.

**cbBuffer:** Maximale Anzahl der Datenbytes die aus der *stream*-Variablen in den externen Puffer kopiert werden sollen.

**nOffset:** Byteoffsetposition des Quelldatenpuffers.

### Ein-/Ausgänge

**stream:** [Quelldatenpuffer \[► 485\]](#).

### Beispiel in ST

Bei einer steigenden Flanke am bRx werden vier Datenbytes ab dem Byteoffset 1 der *stream*-Variablen in die *rxAtOffs*-Variable kopiert.

```

PROGRAM P_iecCopyStreamToBufferByOffset
VAR
    stream      : ST_IEC870_5_101Stream := ( length := 8, data := [16#01, 16#AA, 16#BB, 16#CC, 16#DD, 1
6#06, 16#07, 16#08] );
    bRx         : BOOL;
    rxAtOffs    : DWORD := 0;
    cbResult    : UDINT := 0;
END_VAR

IF bRx THEN
    bRx := FALSE;
    cbResult := F_iecCopyStreamToBufferByOffset ( ADR( rxAtOffs ), SIZEOF( rxAtOffs ), 1, stream );
END_IF
  
```

Speicherdarstellung der *stream*-Variablen nach dem Programmstart:

length	data											
8	16#01	16#AA	16#BB	16#CC	16#DD	16#06	16#07	16#08	16#00	16#00	...	IEC870_MAX_ASDU_DATA_BYTE

Speicherdarstellung der *stream*-Variablen nach dem F\_iecCopyStreamToBufferByOffset-Funktionsaufruf:

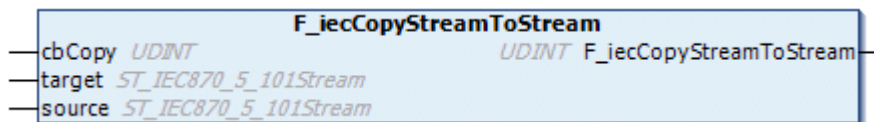
length	data											
8	16#01	16#AA	16#BB	16#CC	16#DD	16#06	16#07	16#08	16#00	16#00	...	IEC870_MAX_ASDU_DATA_BYTE

Die Variable *rxAtOffs* hat den Wert: 16#DDCCBBAA.

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.2.33 F\_iecCopyStreamToStream**



Diese Funktion kopiert Datenbytes von der *source*-Variablen in die *target*-Variable. Der Speicherinhalt der *source*-Variablen bleibt unverändert. Der Speicherinhalt der *target*-Variablen wird vergrößert. Der Rückgabeparameter der Funktion liefert die Anzahl der erfolgreich kopierten Datenbytes.

**Syntax**

```
FUNCTION F_iecCopyStreamToStream: UDINT
VAR_INPUT
    cbCopy : UDINT;
END_VAR
VAR_IN_OUT
    target : ST_IEC870_5_101Stream;
    source : ST_IEC870_5_101Stream;
END_VAR
```

**Eingänge**

**cbCopy:** Anzahl der Bytes die von der *source*-Variablen in die *target*-Variable kopiert werden sollen.

**Ein-/Ausgänge**

**target:** Zieldatenpuffer [[▶ 485](#)].

**source:** Quelldatenpuffer [[▶ 485](#)].

## Beispiel in ST

Bei einer steigenden Flanke am *bCopy* wird zuerst *srcValue* inkrementiert und in *srcStream* kopiert. Danach werden die ersten 4 Datenbytes von *srcStream* nach *dstStream* kopiert. Zum Schluss werden die ersten 4 Datenbytes von *dstStream* in die *dstValue*-Variable kopiert.

```
PROGRAM P_iecCopyStreamToStream
VAR
  srcStream   : ST_IEC870_5_101Stream;
  srcValue    : DWORD;

  dstStream   : ST_IEC870_5_101Stream;
  dstValue    : DWORD;

  bCopy       : BOOL;
END_VAR

IF bCopy THEN
  bCopy := FALSE;
  srcValue := srcValue + 1;
  F_iecCopyBufferToStream( ADR( srcValue ), SIZEOF( srcValue ), srcStream );
  F_iecCopyStreamToStream ( SIZEOF( srcValue ), dstStream, srcStream );
  F_iecCopyStreamToBuffer( ADR( dstValue ), SIZEOF( dstValue ), dstStream );
END_IF
```

Speicherdarstellung der *srcStream*- und *dstStream*-Variablen nach dem ersten *F\_iecCopyStreamToStream*-Funktionsaufruf:

Tab. 80: *srcStream*:

length	data												
4	16#01	16#00	16#00	16#00	...	...	...	...	...	...	...	...	IEC870 _MAX_ ASDU_ DATA_ BYTE

Tab. 81: *dstStream*:

length	data												
4	16#01	16#00	16#00	16#00	...	...	...	...	...	...	...	...	IEC870 _MAX_ ASDU_ DATA_ BYTE

Speicherdarstellung der *srcStream*- und *dstStream*-Variablen nach dem zweiten *F\_iecCopyStreamToStream*-Funktionsaufruf:

Tab. 82: *srcStream*:

length	data												
8	16#01	16#00	16#00	16#00	16#02	16#00	16#00	16#00	...	...	...	...	IEC870 _MAX_ ASDU_ DATA_ BYTE

Tab. 83: *dstStream*:

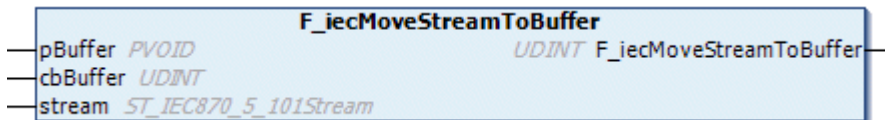
length	data												
8	16#01	16#00	16#00	16#00	16#01	16#00	16#00	16#00	...	...	...	...	IEC870 _MAX_ ASDU_ DATA_ BYTE



Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

5.2.34 F\_iecMoveStreamToBuffer



Diese Funktion kopiert Datenbytes von der *stream*-Variablen in eine externe Puffervariable und löscht anschließend die kopierten Datenbytes aus der *stream*-Variablen. Der Speicherinhalt der *stream*-Variablen wird verkleinert. Der Rückgabeparameter der Funktion liefert die Anzahl der erfolgreich kopierten Datenbytes.

Syntax

```
FUNCTION F_iecMoveStreamToBuffer : UDINT
VAR_INPUT
    pBuffer : PVOID;
    cbBuffer : UDINT;
END_VAR
VAR_IN_OUT
    stream : ST_IEC870_5_101Stream;
END_VAR
```

Eingänge

**pBuffer:** Pointer (Adresse) einer externen Puffervariablen.

**cbBuffer:** Maximale Anzahl der Datenbytes die aus der *stream*-Variablen in den externen Puffer kopiert werden sollen.

Ein-/Ausgänge

**stream:** [Quelldatenpuffer \[▶ 485\]](#).

Beispiel in ST

Nach dem Programmstart sind 8 Byte-Werte in der *stream*-Variable initialisiert. Bei einer steigenden Flanke an *bRx* werden jedes Mal 4 Datenbytes aus der *stream*-Variablen in die *rxBuffer*-Variable kopiert.

```
PROGRAM P_iecMoveStreamToBuffer
VAR
    stream : ST_IEC870_5_101Stream := ( length := 8, data := [16#78, 16#56, 16#34, 16#12, 16#01, 16#EF, 16#CD, 16#AB] );
    rxBuffer : DWORD;
    cbResult : UDINT;
    bRx : BOOL := TRUE;
END_VAR
IF bRx THEN
    bRx := FALSE;
    cbResult := F_iecMoveStreamToBuffer( ADR( rxBuffer ), sizeof( rxBuffer ), stream );
END_IF
```

Speicherdarstellung der *stream*-Variablen nach dem Programmstart:

length	data											
8	16#78	16#56	16#34	16#12	16#01	16#EF	16#CD	16#AB	...	...	...	IEC870 _MAX_ _ASDU_ _DATA_ _BYTE

Speicherdarstellung der stream-Variablen nach dem ersten F\_iecMoveStreamToBuffer-Funktionsaufruf:

length	data											
4	16#01	16#EF	16#CD	16#AB	16#01	16#EF	16#CD	16#AB	...	...	...	IEC870 _MAX_ _ASDU_ _DATA_ _BYTE

Speicherdarstellung der stream-Variablen nach dem zweiten F\_iecMoveStreamToBuffer-Funktionsaufruf:

length	data											
0	16#01	16#EF	16#CD	16#AB	16#01	16#EF	16#CD	16#AB	...	...	...	IEC870 _MAX_ _ASDU_ _DATA_ _BYTE

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.2.35 F\_iecMoveStreamToStream

F_iecMoveStreamToStream	
cbCopy	UDINT UDINT F_iecMoveStreamToStream
target	ST_IEC870_5_101Stream
source	ST_IEC870_5_101Stream

Diese Funktion kopiert Datenbytes von der *source*-Variablen zur *target*-Variablen und löscht anschließend die kopierten Datenbytes aus der *source*-Variablen. Der Speicherinhalt der *source*-Variablen wird verkleinert und der, der *target*-Variablen vergrößert.

### Syntax

```
FUNCTION F_iecMoveStreamToStream : UDINT
VAR_INPUT
    cbCopy : UDINT;
END_VAR
VAR_IN_OUT
    target : ST_IEC870_5_101Stream;
    source : ST_IEC870_5_101Stream;
END_VAR
```

### Eingänge

**cbCopy:** Anzahl der Datenbytes die von der *source*-Variablen in die *target*-Variablen kopiert werden sollen.

**Ein-/Ausgänge**

**target:** Zieldatenpuffer [▶ 485].

**source:** Quelldatenpuffer [▶ 485].

**Beispiel in ST**

```
PROGRAM P_iecMoveStreamToStream
VAR
  target : ST_IEC870_5_101Stream := ( length := 2, data := [16#AA, 16#BB] );
  source : ST_IEC870_5_101Stream := ( length := 8, data := [1,2,3,4,5,6,7,8] );
  cbReturn : UDINT;
  bMove : BOOL := TRUE;
END_VAR

IF bMove THEN
  bMove := FALSE;
  cbReturn := F_iecMoveStreamToStream( source.length, target, source );
END_IF
```

Inhalt der *target*-Variablen nach Programmstart:

length	data											
10	16#AA	16#BB	16#01	16#02	16#03	16#04	16#05	16#06	16#07	16#08	...	IEC870 MAX ASDU DATA BYTE

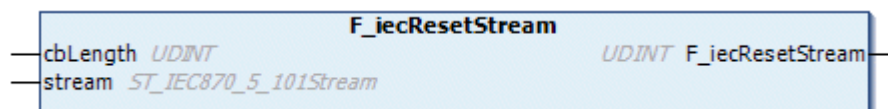
Inhalt der *source*-Variablen nach dem Programmstart:

length	data											
0	16#01	16#02	16#03	16#04	16#05	16#06	16#07	16#08	16#00	16#00	...	IEC870 MAX ASDU DATA BYTE

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.2.36 F\_iecResetStream**



Die Funktion initialisiert und setzt die *stream*-Variable zurück. Optional kann der interne Datenpuffer der *stream*-Variablen mit einer bestimmten Anzahl an Null-Bytes initialisiert werden. Der Rückgabewert der Funktion liefert die Anzahl der erfolgreich initialisierten Null-Bytes.

**Syntax**

```
FUNCTION F_iecResetStream: UDINT
VAR_INPUT
  cbLength : UDINT;
```

```

END_VAR
VAR_IN_OUT
  stream : ST_IEC870_5_101Stream;
END_VAR

```

### 🚪 Eingänge

**cbLength:** Anzahl der zu initialisierten Null-Bytes.

### Ein-/Ausgänge

**stream:** Puffervariable [► 485], die initialisiert werden soll.

### Beispiel in ST

Nach dem Programmstart wird der interne Puffer der *stream*-Variablen zurückgesetzt und mit 5 Null-Bytes initialisiert.

```

PROGRAM P_ResetStream
VAR
  stream : ST_IEC870_5_101Stream;
  bReset : BOOL := TRUE;
END_VAR

IF bReset THEN
  bReset := FALSE;
  F_iecResetStream( 5, stream );
END_IF

```

Speicherinhalt der *stream*-Variablen nach dem Programmstart:

length	data											
5	16#00	16#00	16#00	16#00	16#00	...	...	...	...	...	...	IEC870 MAX_ ASDU_ DATA_ BYTE

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.2.37 F\_iecCreateTableHnd

```

F_iecCreateTableHnd
pAOEntries POINTER TO ARRAY[0..IEC870_ANYSIZE_ARRAY] OF ST_IEC870_5_101AODBEntry
cbAOEntries UDINT
hTable T_HAODBTable
UDINT F_iecCreateTableHnd

```

Die Funktion *F\_iecCreateTableHnd* initialisiert das Applikationsobjekt-Datenbankhandle (Hash-Tabellenhandle). Das Tabellenhandle muss vor der Benutzung einmalig initialisiert werden.

### Syntax

```

FUNCTION F_iecCreateTableHnd: UDINT
VAR_INPUT
  pAOEntries : POINTER TO ST_IEC870_5_101AODBEntry := 0;
  cbAOEntries : UDINT := 0;
END_VAR

```

```
VAR_IN_OUT
  hTable      : T_HAODBTable;
END_VAR
```

**Eingänge**

**pAOEntries:** Adresse der ersten Applikationsobjekt-Datenbankvariablen.

**cbAOEntries:** Bytegröße der Applikationsobjekt-Datenbankvariablen.

**Ein-/Ausgänge**

**hTable:** Das zu initialisierende Applikationsobjekt-Datenbankhandle [|> 531](#)] (Hash-Tabellenhandle).

**Rückgabewert**

Rückgabeparameter	Beschreibung
0	Kein Fehler.
<> 0	Fehler: IEC60870-5-10x Fehlercode <a href="#"> &gt; 557</a> ]

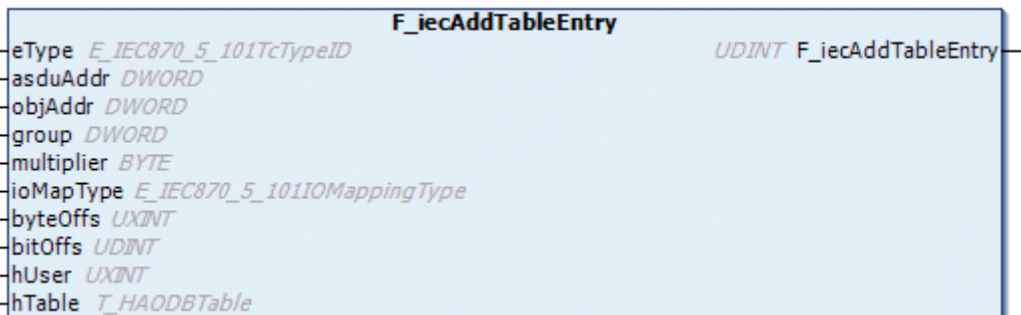
**Beispiel in ST**

Siehe in der Beschreibung der F\_iecAddTableEntry [|> 453](#)]-Funktion.

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.2.38 F\_iecAddTableEntry**



Die Funktion F\_iecAddTableEntry konfiguriert die Applikationsobjekte (Single Points, Double Points, Measured Values...) in der Applikationsdatenbank als Hash-Tabelleneinträge. Die Funktion sucht automatisch nach einem freien, noch nicht belegten Tabellenelement (Arrayelement) und setzt dessen Konfigurationsparameter.

**Syntax**

```
FUNCTION F_iecAddTableEntry: UDINT
VAR_INPUT
  eType      : E_IEC870_5_101TcTypeID := ASDU_TYPEUNDEF;
  asduAddr   : DWORD := 0;
```

```

objAddr      : DWORD := 0;
group        : DWORD := 0;
multiplier   : BYTE  := 0;
ioMapType    : E_IEC870_5_101IOMappingType := MAP_AREA_NONE;
byteOffs     : UXINT := 0;
bitOffs      : UDINT := 0;
hUser        : UXINT := 0;
END_VAR
VAR_IN_OUT
  hTable      : T_HAODBTable;
END_VAR

```

## Eingänge

**eType:** Applikationsobjekt Typ, [ASDU identifier \[► 500\]](#) (z.B.: M\_SP\_NA\_1 für Single-Point oder M\_DP\_NA\_1 für Double-Point ).

**asduAddr:** Gemeinsame ASDU-Adresse.

**objAddr:** Objektadresse, frei wählbar.

**group:** Object-Group-Konfigurationsflags. Hier finden sie die [Beschreibung aller Group-Flags \[► 554\]](#). Die Flags können mit OR-Verknüpfung kombiniert werden. Nicht alle Kombinationen sind aber sinnvoll!

**multiplier:** Basis-Zykluszeit-Multiplikator für zyklische/periodische Datenübertragung. 0 = Deaktiviert. Die Basis-Zykluszeit kann über den *tPerCyclicBase*-Parameter in den [Systemparametern \[► 487\]](#) konfiguriert werden.

**ioMapType:** TwinCAT SPS-Prozessdatenbereich [\[► 506\]](#). Dieser Parameter legt fest wie die TwinCAT SPS und IEC-Applikationsobjekt Prozessdaten gemappt werden sollen.

**byteOffs:** TwinCAT SPS-Prozessdaten-Byte-Offset (x86 Plattform => 32 Bit, x64 Plattform => 64 Bit).

**bitOffs:** TwinCAT SPS-Prozessdaten-Bit-Offset.

**hUser:** Frei definierbarer Wert, auch Pointer (x86 Plattform => 32 Bit, x64 Plattform => 64 Bit). Dieser Wert wird in den Konfigurationsdaten des Applikationsobjekts abgelegt.

## Ein-/Ausgänge

**hTable:** [Applikationsobjekt-Datenbankhandle \[► 531\]](#) (Hash-Tabellenhandle). Das Tabellenhandle muss vor der Benutzung einmalig mit der Funktion [F\\_iecCreateTableHnd \[► 452\]](#) initialisiert werden.

## Rückgabewert

Rückgabeparameter	Beschreibung
0	Kein Fehler.
<> 0	Fehler: IEC60870-5-10x Fehlercode <a href="#">[► 557]</a>

## Beispiel in ST

Im folgenden Beispiel werden einige Datenpunkte in die Applikationsdatenbank als Hash-Tabelleneinträge hinzugefügt:

eType	asduAddr	objAddr	group	multiplier	ioMapType	byteOffs	bitOffs	hUser
M_SP_NA_1	11	100	IEC870_GRP_INRO1	0	MAP_AREA_MEMORY	100	0	16#00BECF01
M_SP_TB_1	11	101	IEC870_GRP_INRO1	0	MAP_AREA_MEMORY	101	0	16#00BECF11
M_DP_NA_1	11	200	IEC870_GRP_INROGEN	0	MAP_AREA_DATA	200	0	16#00BECF02
M_IT_NA_1	11	800	IEC870_GRP_REQCOGEN	0	MAP_AREA_MEMORY	800	0	16#00BECF03

```

VAR_GLOBAL CONSTANT
    MAX_TABLE_IDX : INT := 49;
END_VAR

PROGRAM P_iecAddTableEntry
VAR_IN_OUT
    hTable : T_HAODBTable;
    AODB : ARRAY[1..MAX_TABLE_IDX] OF ST_IEC870_5_101AODBEntry;
END_VAR

VAR
    bAdd : BOOL := TRUE;
    nError : UDINT;
END_VAR

IF bAdd THEN
    bAdd := FALSE;

    IF ( nError := F_iecCreateTableHnd( ADR( AODB ), SIZEOF( AODB ), hTable ) <> 0 THEN
        ADSLOGSTR( ADSLOG_MSGTYPE_ERROR OR ADSLOG_MSGTYPE_LOG, 'F_iecCreateTableHnd() error: %s',
            DWORD_TO_HEXSTR( nError, 8, FALSE ) );
        RETURN;
    END_IF

    IF ( nError := F_iecAddTableEntry( eType := M_SP_NA_1,
        asduAddr := 11,
        objAddr := 100,
        group := IEC870_GRP_INRO1,
        multiplier := 0,
        ioMapType := MAP_AREA_MEMORY,
        byteOffs := 100,
        bitOffs := 0,
        hUser := 16#00BECF01,
        hTable := hTable ) ) <> 0 THEN

        ADSLOGSTR( ADSLOG_MSGTYPE_ERROR OR ADSLOG_MSGTYPE_LOG, 'F_iecAddTableEntry() error: %s',
            DWORD_TO_HEXSTR( nError, 8, FALSE ) );
    END_IF

    IF ( nError := F_iecAddTableEntry( eType := M_SP_TB_1,
        asduAddr := 11,
        objAddr := 101,
        group := IEC870_GRP_INRO1,
        multiplier := 0,
        ioMapType := MAP_AREA_MEMORY,
        byteOffs := 101,
        bitOffs := 0,
        hUser := 16#00BECF11,
        hTable := hTable ) ) <> 0 THEN

        ADSLOGSTR( ADSLOG_MSGTYPE_ERROR OR ADSLOG_MSGTYPE_LOG, 'F_iecAddTableEntry() error: %s',
            DWORD_TO_HEXSTR( nError, 8, FALSE ) );
    END_IF

    IF ( nError := F_iecAddTableEntry( eType := M_DP_NA_1,
        asduAddr := 11,
        objAddr := 200,
        group := IEC870_GRP_INROGEN,
        multiplier := 0,
        ioMapType := MAP_AREA_DATA,
        byteOffs := 200,

```

```

        bitOffs      := 0,
        hUser       := 16#00BECF02,
        hTable     := hTable ) ) <> 0 THEN
    ADSLOGSTR( ADSLOG_MSGTYPE_ERROR OR ADSLOG_MSGTYPE_LOG, 'F_iecAddTableEntry() error: %s',
        DWORD_TO_HEXSTR( nError, 8, FALSE ) );
END_IF

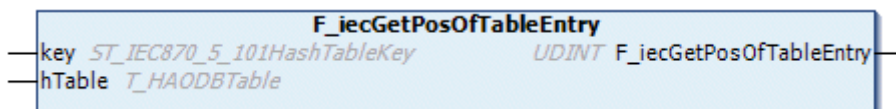
IF ( nError := F_iecAddTableEntry( eType := M_IT_NA_1,
    asduAddr      := 11,
    objAddr       := 800,
    group         := IEC870_GRP_REQCOGEN,
    multiplier    := 0,
    ioMapType     := MAP_AREA_MEMORY,
    byteOffs      := 800,
    bitOffs       := 0,
    hUser         := 16#00BECF03,
    hTable        := hTable ) ) <> 0 THEN
    ADSLOGSTR( ADSLOG_MSGTYPE_ERROR OR ADSLOG_MSGTYPE_LOG, 'F_iecAddTableEntry() error: %s',
        DWORD_TO_HEXSTR( nError, 8, FALSE ) );
END_IF
END_IF

```

## Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.2.39 F\_iecGetPosOfTableEntry



Die Funktion `F_iecGetPosOfTableEntry` liefert den zum Lookup-Schlüssel passenden Tabellenindex (Arrayindex) eines Hash-Tabelleneintrags. Das erste Arrayelement hat die Positionsnummer eins (non-zero-based array position).

### Syntax

```

FUNCTION F_iecGetPosOfTableEntry: UDINT
VAR_INPUT
    key    : ST_IEC870_5_101HashTableKey;
END_VAR
VAR_IN_OUT
    hTable : T_HAODBTable;
END_VAR

```

### Eingänge

**key:** [Lookup-Schlüssel](#) [[▶ 497](#)].

### Ein-/Ausgänge

**hTable:** [Applikationsobjekt-Datenbankhandle](#) [[▶ 531](#)] (Hash-Tabellenhandle). Das Tabellenhandle muss vor der Benutzung einmalig mit der Funktion `F_iecCreateTableHnd` [[▶ 452](#)] initialisiert werden.



**🚩 Rückgabewert**

Rückgabeparameter	Beschreibung
0	Es wurde kein zum Schlüssel passender Tabelleneintrag gefunden.
<> 0	Kein Fehler. Der Rückgabeparameter liefert den gesuchten Tabellenindex (non-zero-based array position).

**Beispiel in ST**

Es wird nach dem Linearen-Tabellenindex von drei Datenpunkten gesucht. Die gesuchten Datenpunkte sind bereits als Hash-Tabelleneinträge konfiguriert worden. Siehe in der Beschreibung der Funktion:

F\_iecAddTableEntry [▶ 453].

eType	objAddr	asduAddr	group
M_SP_NA_1	100	11	IEC870_GRP_INRO1
M_SP_TB_1	101	11	IEC870_GRP_INRO1
M_DP_NA_1	200	11	IEC870_GRP_INROGEN
M_IT_NA_1	800	11	IEC870_GRP_REQCOGEN

```

VAR_GLOBAL CONSTANT
    MAX_TABLE_IDX : INT := 49;
END_VAR

VAR_IN_OUT
    hTable : T_HAODBTable;
    AODB : ARRAY[1..MAX_TABLE_IDX] OF ST_IEC870_5_101AODBEntry;
END_VAR

VAR
    bGet : BOOL := TRUE;
    key1 : ST_IEC870_5_101HashTableKey := ( eType := M_SP_NA_1, asduAddr := 11, objAddr := 100, group := IEC870_GRP_INRO1, lookup := IEC870_LOOKUP_KEY_ALL_ON );
    key2 : ST_IEC870_5_101HashTableKey := ( eType := M_SP_TB_1, asduAddr := 11, objAddr := 101, group := IEC870_GRP_INRO1, lookup := IEC870_LOOKUP_KEY_ALL_ON );
    key3 : ST_IEC870_5_101HashTableKey := ( eType := M_DP_NA_1, asduAddr := 11, objAddr := 200, group := IEC870_GRP_INROGEN, lookup := IEC870_LOOKUP_KEY_ALL_ON );
    key4 : ST_IEC870_5_101HashTableKey := ( eType := M_IT_NA_1, asduAddr := 11, objAddr := 800, group := IEC870_GRP_REQCOGEN, lookup := IEC870_LOOKUP_KEY_ALL_ON );

    nPosition : UDINT;
    hUser1 : UXINT;
    hUser2 : UXINT;
    hUser3 : UXINT;
    hUser4 : UXINT;
END_VAR

IF bGet THEN
    bGet := FALSE;

    IF ( nPosition := F_iecGetPosOfTableEntry( key := key1, hTable := hTable ) ) = 0 THEN
        ADSLOGSTR( ADSLOG_MSGTYPE_ERROR OR ADSLOG_MSGTYPE_LOG, 'F_iecGetPosOfTableEntry(key1) failed! %s', '' );
    ELSE
        hUser1 := AODB[nPosition].aObj.cfg.hUser;
    END_IF

    IF ( nPosition := F_iecGetPosOfTableEntry( key := key2, hTable := hTable ) ) = 0 THEN
        ADSLOGSTR( ADSLOG_MSGTYPE_ERROR OR ADSLOG_MSGTYPE_LOG, 'F_iecGetPosOfTableEntry(key2) failed! %s', '' );
    ELSE
        hUser2 := AODB[nPosition].aObj.cfg.hUser;
    END_IF

    IF ( nPosition := F_iecGetPosOfTableEntry( key := key3, hTable := hTable ) ) = 0 THEN
        ADSLOGSTR( ADSLOG_MSGTYPE_ERROR OR ADSLOG_MSGTYPE_LOG, 'F_iecGetPosOfTableEntry(key3) failed! %s', '' );
    ELSE
        hUser3 := AODB[nPosition].aObj.cfg.hUser;
    END_IF

```

```

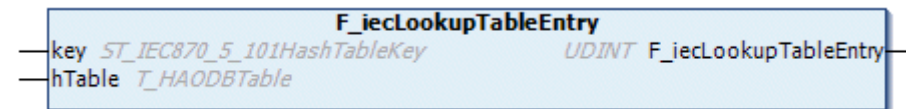
IF ( nPosition := F_iecGetPosOfTableEntry( key := key4, hTable := hTable ) ) = 0 THEN
    ADSLOGSTR( ADSLOG_MSGTYPE_ERROR OR ADSLOG_MSGTYPE_LOG, 'F_iecGetPosOfTableEntry(key4) failed
! %s', '' );
ELSE
    hUser4 := AODB[nPosition].aObj.cfg.hUser;
END_IF
END_IF

```

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.2.40 F\_iecLookupTableEntry



Die Funktion F\_iecLookupTableEntry prüft ob ein zum Schlüssel passender Tabelleneintrag bereits vorhanden ist.

### Syntax

```

FUNCTION F_iecLookupTableEntry: UDINT
VAR_INPUT
    key    : ST_IEC870_5_101HashTableKey;
END_VAR
VAR_IN_OUT
    hTable : T_HAODBTable;
End_VAR

```

### Eingänge

**key:** Lookup-Schlüssel [[▶ 497](#)].

### Ein-/Ausgänge

**hTable:** Applikationsobjekt-Datenbankhandle [[▶ 531](#)] (Hash-Tabellenhandle). Das Tabellenhandle muss vor der Benutzung einmalig mit der Funktion F\_iecCreateTableHnd [[▶ 452](#)] initialisiert werden.

### Rückgabewert

Rückgabeparameter	Beschreibung
0	Kein Fehler. Es existiert ein zum Schlüssel passender Tabelleneintrag.
<> 0	Es wurde kein Tabelleneintrag gefunden. Fehler: IEC60870-5-10x Fehlercode [ <a href="#">▶ 557</a> ]

### Beispiel in ST

Es wird die Existenz von Datenpunkten in der Applikationsdatenbank überprüft. Die gesuchten Datenpunkte sind bereits als Hash-Tabelleneinträge konfiguriert worden. Siehe in der Beschreibung der Funktion: F\_iecAddTableEntry [[▶ 453](#)].

eType	objAddr	asduAddr	group
M_SP_NA_1	100	11	IEC870_GRP_INRO1
M_SP_TB_1	101	11	IEC870_GRP_INRO1
M_DP_NA_1	200	11	IEC870_GRP_INROGEN
M_IT_NA_1	800	11	IEC870_GRP_REQCOGEN

```

PROGRAM P_iecLookupTableEntry
VAR_IN_OUT
    hTable : T_HAODBTable;
END_VAR

VAR
    bLookup : BOOL := TRUE;
    key1 : ST_IEC870_5_101HashTableKey := ( eType := M_SP_NA_1, asduAddr := 11, objAddr := 100, group := IEC870_GRP_INRO1, lookup := IEC870_LOOKUP_KEY_ALL_ON );
    key2 : ST_IEC870_5_101HashTableKey := ( eType := M_SP_TB_1, asduAddr := 11, objAddr := 101, group := IEC870_GRP_INRO1, lookup := IEC870_LOOKUP_KEY_ALL_ON );
    key3 : ST_IEC870_5_101HashTableKey := ( eType := M_DP_NA_1, asduAddr := 11, objAddr := 200, group := IEC870_GRP_INROGEN, lookup := IEC870_LOOKUP_KEY_ALL_ON );
    key4 : ST_IEC870_5_101HashTableKey := ( eType := M_IT_NA_1, asduAddr := 11, objAddr := 800, group := IEC870_GRP_REQCOGEN, lookup := IEC870_LOOKUP_KEY_ALL_ON );

    nError : UDINT;
END_VAR

IF bLookup THEN
    bLookup := FALSE;

    IF ( nError := F_iecLookupTableEntry( key := key1, hTable := hTable ) ) <> 0 THEN
        ADSLOGSTR( ADSLOG_MSGTYPE_ERROR OR ADSLOG_MSGTYPE_LOG, 'F_iecLookupTableEntry(key1) error: %s',
            DWORD_TO_HEXSTR( nError, 8, FALSE ) );
    END_IF

    IF ( nError := F_iecLookupTableEntry( key := key2, hTable := hTable ) ) <> 0 THEN
        ADSLOGSTR( ADSLOG_MSGTYPE_ERROR OR ADSLOG_MSGTYPE_LOG, 'F_iecLookupTableEntry(key2) error: %s',
            DWORD_TO_HEXSTR( nError, 8, FALSE ) );
    END_IF

    IF ( nError := F_iecLookupTableEntry( key := key3, hTable := hTable ) ) <> 0 THEN
        ADSLOGSTR( ADSLOG_MSGTYPE_ERROR OR ADSLOG_MSGTYPE_LOG, 'F_iecLookupTableEntry(key3) error: %s',
            DWORD_TO_HEXSTR( nError, 8, FALSE ) );
    END_IF

    IF ( nError := F_iecLookupTableEntry( key := key4, hTable := hTable ) ) <> 0 THEN
        ADSLOGSTR( ADSLOG_MSGTYPE_ERROR OR ADSLOG_MSGTYPE_LOG, 'F_iecLookupTableEntry(key4) error: %s',
            DWORD_TO_HEXSTR( nError, 8, FALSE ) );
    END_IF
END_IF

```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uutilities (System)

## 5.2.41 F\_iecRemoveTableEntry

F_iecRemoveTableEntry	
key	ST_IEC870_5_101HashTableKey UDINT F_iecRemoveTableEntry
hTable	T_HAODBTable

Die Funktion F\_iecRemoveTableEntry entfernt einen zum Schlüssel passenden Hash-Tabelleneintrag.

### Syntax

```
FUNCTION F_iecRemoveTableEntry: UDINT
VAR_INPUT
    key      : ST_IEC870_5_101HashTableKey;
END_VAR
VAR_IN_OUT
    hTable   : T_HAODBTable;
END_VAR
```

### Eingänge

**key:** Lookup-Schlüssel [► 497].

### Ein-/Ausgänge

**hTable:** Applikationsobjekt-Datenbankhandle [► 531] (Hash-Tabellenhandle). Das Tabellenhandle muss vor der Benutzung einmalig mit der Funktion F\_iecCreateTableHnd [► 452] initialisiert werden.

### Rückgabewert

Rückgabeparameter	Beschreibung
0	Kein Fehler, der Tabelleneintrag wurde erfolgreich entfernt.
<> 0	Fehler: IEC60870-5-10x Fehlercode [► 557]

### Beispiel in ST

Es werden drei Hash-Tabelleneinträge aus der Applikationsdatenbank entfernt. Die gesuchten Datenpunkte sind bereits als Hash-Tabelleneinträge konfiguriert worden. Siehe in der Beschreibung der Funktion: F\_iecAddTableEntry [► 453].

eType	objAddr	asduAddr	group
M_SP_NA_1	100	11	IEC870_GRP_INRO1
M_DP_NA_1	200	11	IEC870_GRP_INROGEN
M_IT_NA_1	800	11	IEC870_GRP_REQCOGE N

```
PROGRAM P_RemoveEntry
VAR_IN_OUT
    hTable   : T_HAODBTable;
END_VAR

VAR
    key      : ST_IEC870_5_101HashTableKey;
    bRemove  : BOOL;
    nError   : UDINT;
END_VAR

IF bRemove THEN
    bRemove   := FALSE;

    key.eType   := M_SP_NA_1;
    key.asduAddr := 11;
    key.objAddr  := 100;
    key.group    := IEC870_GRP_INRO1;
    key.lookup   := IEC870_LOOKUP_KEY_ALL_ON;
```

```

nError      := F_iecRemoveTableEntry( key := key, hTable := hTable );
IF nError <> 0 THEN
    RETURN;
END_IF

key.eType   := M_DP_NA_1;
key.asduAddr := 11;
key.objAddr  := 200;
key.group   := IEC870_GRP_INROGEN;
key.lookup  := IEC870_LOOKUP_KEY_ALL_ON;
nError      := F_iecRemoveTableEntry( key := key, hTable := hTable );
IF nError <> 0 THEN
    RETURN;
END_IF

key.eType   := M_IT_NA_1;
key.asduAddr := 11;
key.objAddr  := 800;
key.group   := IEC870_GRP_REQCOGEN;
key.lookup  := IEC870_LOOKUP_KEY_ALL_ON;
nError      := F_iecRemoveTableEntry( key := key, hTable := hTable );
IF nError <> 0 THEN
    RETURN;
END_IF
END_IF

```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.2.42 F\_iecCmpAddrOctets**



Diese Funktion vergleicht zwei Adressen (z.B. Idie Verbindungsadresse, Objektadresse oder die gemeinsame ASDU-Adresse).

**Syntax**

```

FUNCTION F_iecCmpAddrOctets: BOOL
VAR_INPUT
    a    : DWORD;
    b    : DWORD;
    size : INT(0..4);
END_VAR

```

**Eingänge**

- a:** Erste Adresse.
- b:** Zweite Adresse.
- size:** Bytegrösse der Adresse (0..3, 4 ist reserviert).

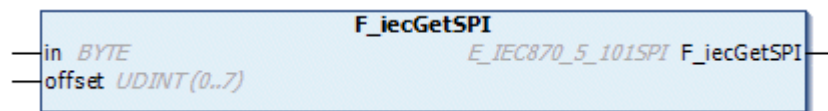
### 🔴 Rückgabewert

Rückgabeparameter	Bedeutung
TRUE	(a und b sind gleich) oder (b ist eine Adresse an alle(broadcast)) oder (size ist 0),
FALSE	Alle anderen Fälle.

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.2.43 F\_iecGetSPI



Mit dieser Hilfsfunktion kann aus einer Byte-Variablen die Information einer Einzelmeldung [► 526] (single point information) ausmaskiert werden. In der TwinCAT SPS belegt die Information einer Einzelmeldung 1 Bit an Prozessdaten. In einer Byte-Variablen können somit Informationen von bis zu 8 Einzelmeldungen gemappt werden.

### Syntax

```
FUNCTION F_iecGetSPI: E_IEC870_5_101SPI
VAR_INPUT
    in      : BYTE;
    offset  : UDINT(0..7);
END_VAR
```

### 🔴 Eingänge

**in:** Byte Variable aus der die Information einer Einzelmeldung ausmaskiert werden soll.

**offset:** Bitoffset der Einzelmeldung.

### Beispiel in ST

Die Information der vier Einzelmeldungen wird ausmaskiert.

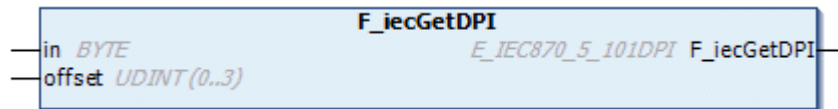
```
PROGRAM MAIN
VAR
    memarea AT%M* : ARRAY[0..10] OF BYTE;
    eSPI          : E_IEC870_5_101SPI;
End_VAR

eSPI := F_iecGetSPI( memarea[0], 0 );
eSPI := F_iecGetSPI( memarea[0], 1 );
eSPI := F_iecGetSPI( memarea[0], 2 );
eSPI := F_iecGetSPI( memarea[0], 3 );
```

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

5.2.44 F\_iecGetDPI



Mit dieser Hilfsfunktion kann aus einer Byte-Variablen die Information einer Doppelmeldung [► 527] (double point information) ausmaskiert werden. In der TwinCAT SPS belegt die Information einer Doppelmeldung 2 Bit an Prozessdaten. In einer Byte-Variablen kann somit die Information von bis zu 4 Doppelmeldungen gemappt werden.

Syntax

```
FUNCTION F_iecGetDPI: E_IEC870_5_101DPI
VAR_INPUT
    in      : BYTE;
    offset  : UDINT(0..3);
END_VAR
```

👉 Eingänge

**in:** Variable aus der die Information einer Doppelmeldung ausmaskiert werden soll.

**offset:** Bitoffset der Information einer Doppelmeldung.

Beispiel in ST

Die Information der vier Doppelmeldungen wird ausmaskiert.

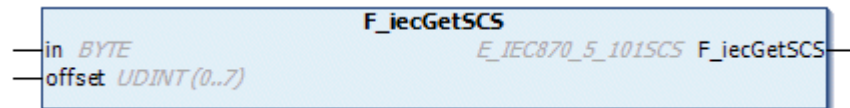
```
PROGRAM MAIN
VAR
    memarea AT%M* : ARRAY[0..10] OF BYTE;
    eDPI          : E_IEC870_5_101DPI;
End_VAR

eDPI := F_iecGetDPI( memarea[0], 0 );
eDPI := F_iecGetDPI( memarea[0], 1 );
eDPI := F_iecGetDPI( memarea[0], 2 );
eDPI := F_iecGetDPI( memarea[0], 3 );
```

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.2.45 F\_iecGetSCS



Mit dieser Hilfsfunktion kann aus einer Byte-Variablen der Status eines Einzelbefehls [► 520] (single command state) ausmaskiert werden. In der TwinCAT SPS belegt der Status eines Einzelbefehls 1 Bit an Prozessdaten. In einer Byte-Variablen kann somit der Status von bis zu 8 Einzelbefehlen gemappt werden.

### Syntax

```
FUNCTION F_iecGetSCS: E_IEC870_5_101SCS
VAR_INPUT
    in      : BYTE;
    offset  : UDINT(0..7);
END_VAR
```

### Eingänge

**in:** Byte Variable aus der der Status eines Einzelbefehls extrahiert werden soll.

**offset:** Bitoffset des Einzelbefehls.

### Beispiel in ST

Der Status der vier Einzelbefehle wird ausmaskiert.

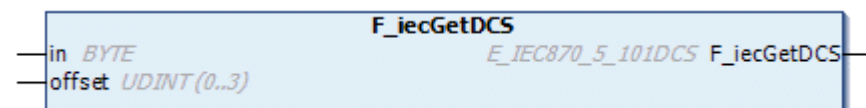
```
PROGRAM MAIN
VAR
    memarea AT%M* : ARRAY[0..10] OF BYTE;
    eSCS          : E_IEC870_5_101SCS;
END_VAR

eSCS := F_iecGetSCS( memarea[0], 0 );
eSCS := F_iecGetSCS( memarea[0], 1 );
eSCS := F_iecGetSCS( memarea[0], 2 );
eSCS := F_iecGetSCS( memarea[0], 3 );
```

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.2.46 F\_iecGetDCS



Mit dieser Hilfsfunktion kann aus einer Byte-Variablen der Status eines Doppelbefehls [► 520] (double command state) ausmaskiert werden. In der TwinCAT SPS belegt der Status eines Doppelbefehls 2 Bit an Prozessdaten. In einer Byte-Variablen kann somit der Status von bis zu 4 Doppelbefehlen gemappt werden.



**Syntax**

```
FUNCTION F_iecGetDCS: E_IEC870_5_101DCS
VAR_INPUT
    in      : BYTE;
    offset  : UDINT(0..3);
END_VAR
```

 **Eingänge**

**in:** Variable aus der der Status des Doppelbefehls ausmaskiert werden soll.

**offset:** Bitoffset des Status eines Doppelbefehls.

**Beispiel in ST**

Der Status der vier Doppelbefehle wird ausmaskiert.

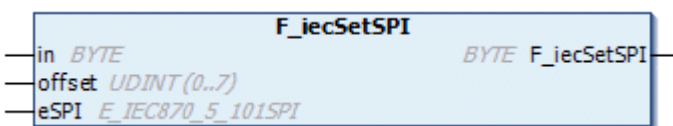
```
PROGRAM MAIN
VAR
    memarea AT%M* : ARRAY[0..10] OF BYTE;
    eDCS          : E_IEC870_5_101DCS;
END_VAR

eDCS := F_iecGetDCS( memarea[0], 0 );
eDCS := F_iecGetDCS( memarea[0], 1 );
eDCS := F_iecGetDCS( memarea[0], 2 );
eDCS := F_iecGetDCS( memarea[0], 3 );
```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uutilities (System)

**5.2.47 F\_iecSetSPI**



Diese Hilfsfunktion kopiert die Information einer Einzelmeldung [\[ 526 \]](#) (single point information) in eine Byte-Variable. In der TwinCAT SPS belegt die Information einer Einzelmeldung 1 Bit an Prozessdaten. In einer Byte-Variablen können somit Informationen von bis zu 8 Einzelmeldungen gemappt werden.

**Syntax**

```
FUNCTION F_iecSetSPI: BYTE
VAR_INPUT
    in      : BYTE;
    offset  : UDINT(0..7);
    eSPI    : E_IEC870_5_101SPI;
END_VAR
```

 **Eingänge**

**in:** Zielvariable in die die Information der Einzelmeldung hineinkopiert werden soll.

**offset:** Bitoffset der Information der Einzelmeldung.

**eSPI:** Der neue Wert der Information der Einzelmeldung.

## Beispiel in ST

Die Information der vier Einzelmeldungen wird auf ON gesetzt.

```
PROGRAM MAIN
VAR
  memarea AT%M* : ARRAY[0..10] OF BYTE;
END_VAR

memarea[0] := F_iecSetSPI( memarea[0], 0, eIEC870_SPI_ON );
memarea[0] := F_iecSetSPI( memarea[0], 1, eIEC870_SPI_ON );
memarea[0] := F_iecSetSPI( memarea[0], 2, eIEC870_SPI_ON );
memarea[0] := F_iecSetSPI( memarea[0], 3, eIEC870_SPI_ON );
```

## Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.2.48 F\_iecSetDPI



Diese Hilfsfunktion kopiert die Information einer Doppelmeldung [► 527] (double point information) in eine Byte-Variable. In der TwinCAT SPS belegt die Information einer Doppelmeldung 2 Bit an Prozessdaten. In einer Byte-Variablen kann somit die Information von bis zu 4 Doppelmeldungen gemappt werden.

### Syntax

```
FUNCTION F_iecSetDPI: BYTE
VAR_INPUT
  in      : BYTE;
  offset  : UDINT(0..3);
  eDPI    : E_IEC870_5_101DPI;
END_VAR
```

### 👉 Eingänge

**in:** Zielvariable in die die Information der Doppelmeldung hineinkopiert werden soll.

**offset:** Bitoffset der Information der Doppelmeldung.

**eDPI:** Der neue Wert der Information der Doppelmeldung.

## Beispiel in ST

Die Information der vier Doppelmeldungen wird auf ON gesetzt.

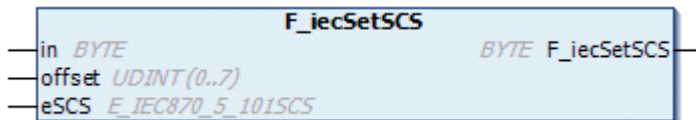
```
PROGRAM MAIN
VAR
  memarea AT%M* : ARRAY[0..10] OF BYTE;
END_VAR

memarea[0] := F_iecSetDPI( memarea[0], 0, eIEC870_DPI_ON );
memarea[0] := F_iecSetDPI( memarea[0], 1, eIEC870_DPI_ON );
memarea[0] := F_iecSetDPI( memarea[0], 2, eIEC870_DPI_ON );
memarea[0] := F_iecSetDPI( memarea[0], 3, eIEC870_DPI_ON );
```

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

5.2.49 F\_iecSetSCS



Diese Hilfsfunktion kopiert den Status eines Einzelbefehls [► 520] (single command state) in eine Byte-Variable. In der TwinCAT SPS belegt der Status eines Einzelbefehls 1 Bit an Prozessdaten. In einer Byte-Variablen kann somit der Status von bis zu 8 Einzelbefehlen gemappt werden.

Syntax

```
FUNCTION F_iecSetSCS: BYTE
VAR_INPUT
    in      : BYTE;
    offset  : UDINT(0..7);
    eSCS    : E_IEC870_5_101SCS;
END_VAR
```

📌 Eingänge

**in:** Zielvariable in die der Status des Einzelbefehls hineinkopiert werden soll.

**offset:** Bitoffset des Statuses des Einzelbefehls.

**eSCS:** Der neue Wert des Statuses des Einzelbefehls.

Beispiel in ST

Der Status der vier Einzelbefehle wird auf ON gesetzt.

```
PROGRAM MAIN
VAR
    memarea AT%M* : ARRAY[0..10] OF BYTE;
END_VAR

memarea[0] := F_iecSetSCS( memarea[0], 0, eIEC870_SCS_ON );
memarea[0] := F_iecSetSCS( memarea[0], 1, eIEC870_SCS_ON );
memarea[0] := F_iecSetSCS( memarea[0], 2, eIEC870_SCS_ON );
memarea[0] := F_iecSetSCS( memarea[0], 3, eIEC870_SCS_ON );
```

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.2.50 F\_iecSetDCS



Diese Hilfsfunktion kopiert den Status eines Doppelbefehls [► 520] (double command state) in eine Byte-Variable. In der TwinCAT SPS belegt der Status eines Doppelbefehls 2 Bit an Prozessdaten. In einer Byte-Variablen kann somit der Status von bis zu 4 Doppelbefehlen gemappt werden.

### Syntax

```
FUNCTION F_iecSetDCS: BYTE
VAR_INPUT
    in      : BYTE;
    offset  : UDINT(0..3);
    eDCS    : E_IEC870_5_101DCS;
END_VAR
```

### 👉 Eingänge

**in:** Zielvariable in die der Status des Doppelbefehls hineinkopiert werden soll.

**offset:** Bitoffset des Status des Doppelbefehls.

**eDCS:** Der neue Wert des Status des Doppelbefehls [► 520].

### Beispiel in ST

Der Status der vier Doppelbefehle wird auf ON gesetzt.

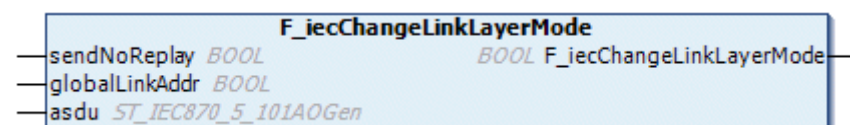
```
PROGRAM MAIN
VAR
    memarea AT%M* : ARRAY[0..10] OF BYTE;
END_VAR

memarea[0] := F_iecSetDCS( memarea[0], 0, eIEC870_DCS_ON );
memarea[1] := F_iecSetDCS( memarea[1], 1, eIEC870_DCS_ON );
memarea[2] := F_iecSetDCS( memarea[2], 2, eIEC870_DCS_ON );
memarea[3] := F_iecSetDCS( memarea[3], 3, eIEC870_DCS_ON );
```

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.2.51 F\_iecChangeLinkLayerMode



Mit dieser Funktion kann das Defaultverhalten jedes einzelnen ASDU-Frames auf der Link-Layer-Ebene in der Primärstation festgelegt bzw. verändert werden. Das zu versendende ASDU-Frame kann als SEND/NO REPLY-Frame (Funktion 4) oder Broadcast-Frame gekennzeichnet werden. Die Kennzeichnung der Frames erfolgt vor dem Ablegen des Frames in dem TX-Puffer. D.h. vor dem Aufruf der [FB IEC870\\_5\\_101TBufferCtrl](#) [► 403].TxAddObj-Aktion.

Wenn Sie die Funktion nicht verwenden, dann werden in der Primärstation alle Frames als SEND/CONFIRM-Frames (Funktion 3) versendet. Als Linkadresse wird in diesem Fall die konfigurierte Stationsadresse verwendet.



- Im Unbalanced-Modus hat die Verwendung dieser Funktion in der Unterstation keine Bedeutung. Die Unterstation agiert in diesem Modus niemals als Primärstation.
- Bei der Verwendung des IEC 6087-5-104 Protokolls hat die Funktion keine Bedeutung.
- Diese Funktionalität wird nur bei der Verwendung der "Low level"-Schnittstelle unterstützt.

**Syntax**

```
FUNCTION F_iecChangeLinkLayerMode: BOOL
VAR_INPUT
    sendNoReplay    : BOOL;
    globalLinkAddr  : BOOL;
END_VAR
VAR_IN_OUT
    asdu            : ST_IEC870_5_101AObj;
END_VAR
```

**Eingänge**

**sendNoReplay:** Dieser Parameter legt fest ob beim Versenden des Frames die Link-Layer-Funktion: SEND/NO REPLY (TRUE = Funktion 4) oder SEND/CONFIRM (FALSE = Funktion 3) verwendet werden soll;

**globalLinkAddr:** Dieser Parameter legt fest ob beim Versenden des Frames statt der konfigurierten Stations-Linkadresse eine globale (Broadcast) Adresse verwendet werden soll. Beim Wert TRUE wird als Linkadresse im gesendeten Frame 16#FF bzw. 16#FFFF verwendet (one octet size, two octets size link address);

**Ein-/Ausgänge**

**asdu:** Die zu versendende [Dateneinheit](#) [► 483] als VAR\_IN\_OUT-Variable;

**Rückgabewert**

Rückgabeparameter	Beschreibung
FALSE	Funktion fehlgeschlagen.
TRUE	Kein Fehler.

**Beispiel 1 in ST (Ausschnitt)**

Die spontanen Daten eines Bitstrings sollen mit der Hilfe der Funktion SEND/NO REPLY an die Zentralstation gesendet werden (Balanced-Mode).

```
...
(* Send spontaneous bitstring data *)
IF ( txQDS <> BITSTRING_QUALITY_100 ) OR ( txBSI <> BITSTRING_100 ) THEN

    txBSI          := BITSTRING_100;
    txQDS          := BITSTRING_QUALITY_100;(* Get quality *)
    txTT           := SYSTEMTIME_TO_CP56Time2a( fbRTC.CDT, TRUE );
(* Get current time stamp *)(* create asdu *)
    txAsdu.ident.eType := M_BO_TB_1; (* Bit string with time tag *)
    txAsdu.ident.bsQ   := FALSE;
    txAsdu.ident.nObj  := 1;
    txAsdu.ident.eCOT  := eIEC870_COT_SPONTAN;
    txAsdu.ident.nORG  := sysPara.nOrg;(* Set originator address *)
    txAsdu.ident.bPN   := FALSE;
```

```

txAsdu.ident.bT      := FALSE;
txAsdu.ident.eClass := eIEC870_Class_1;(* Put to the high priority tx buffer *)
txAsdu.ident.asduAddr := sysPara.asduAddr;(* Set common asdu address *)
txAsdu.info.objAddr  := 100;(* Set information object address *)
F_iecResetStream( 0, txAsdu.info.stream ); (* clear previous data (this sets the stream length =
0 *)
F_iecCopyBufferToStream( ADR( txBSI ), SIZEOF( txBSI ), txAsdu.info.stream ); (* put BSI to stre
am *)
F_iecCopyBufferToStream( ADR( txQDS ), SIZEOF( txQDS ), txAsdu.info.stream ); (* put QDS to stre
am *)
F_iecCopyBufferToStream( ADR( txTT ), 7 (*SIZEOF( txTT )-1*), txAsdu.info.stream ); (* put time
tag to stream *)F_iecChangeLinkLayerMode( TRUE, FALSE, txAsdu );

fbBuffer.TxAddObj( putObj := txAsdu, buffer := buffer ); (* put asdu to the TX fifo *)F_iecChang
eLinkLayerMode( FALSE, FALSE, txAsdu );

IF fbBuffer.bOk THEN
    fbLog( put := CONCAT( '<=', IEC101ASDU_TO_STRING(txAsdu) ) );
ELSE(* Report send buffer overflow error *)
    fbLog( put := 'TX buffer overflow (spontaneous bitstring data)!' );
END_IF
END_IF
...

```

Die txAsdu-Variable wird zum Versenden weiterer Datenpunkte verwendet. Die Default-Konfiguration des txAsdu-Frames wird durch einen weiteren F\_iecChangeLinkLayerMode(FALSE, FALSE, ...)-Funktionsaufruf hergestellt. Andere ASDUs sollen nicht als SEND/NO REPLY-Telegramme gesendet werden.

### Beispiel 2 in ST (Ausschnitt)

Ein Einzelbefehl soll mit der Hilfe der Funktion SEND/NO REPLAY an die Unterstation gesendet werden (Unbalanced-Mode).

```

...
(* Send one single command *)
IF SND_SCS_2100 THEN
    SND_SCS_2100 := FALSE; (* Reset flag *)

    txAsdu.ident.eType      := C_SC_NA_1; (* Single command *)
    txAsdu.ident.bSQ       := FALSE;
    txAsdu.ident.nObj      := 1;
    txAsdu.ident.eCOT      := eIEC870_COT_ACT; (* Command activation *)
    txAsdu.ident.nORG      := sysPara.nOrg;(* Set originator address *)
    txAsdu.ident.bPN       := FALSE;
    txAsdu.ident.bT        := FALSE;
    txAsdu.ident.eClass    := eIEC870_Class_1;(* Put to the high priority tx buffer *)
    txAsdu.ident.asduAddr  := sysPara.asduAddr;(* Set common asdu address *)
    txAsdu.info.objAddr    := 2100;(* Set information object address *)
    tmpByte                := INT_TO_BYTE(SCS_2100);(* Set single command state *)
    tmpByte.7              := 0;(* Set select/execute bit *)

    F_iecResetStream( 0, txAsdu.info.stream ); (* Clear previous data (this sets the stream length =
0 *)
    F_iecCopyBufferToStream( ADR( tmpByte ), SIZEOF( tmpByte ), txAsdu.info.stream ); (* put QCC to
stream *)      F_iecChangeLinkLayerMode( TRUE, FALSE, txAsdu );

    fbBuffer.TxAddObj( putObj := txAsdu, buffer := buffer ); (* put asdu to the TX fifo *)
    F_iecChangeLinkLayerMode( FALSE, FALSE, txAsdu );

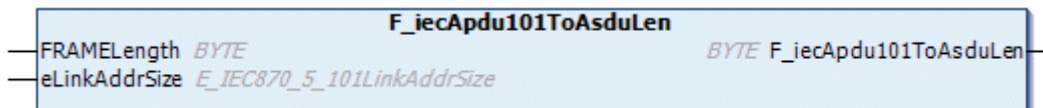
    IF fbBuffer.bOk THEN
        timerCON( IN := FALSE );(* Reset timer *)
        timerTERM( IN := FALSE );(* Reset timer *)
        fbLog( put := CONCAT( '<=', IEC101ASDU_TO_STRING(txAsdu) ) );
        state := 80;(* Wait for command confirmation *)
    ELSE(* Report send buffer overflow error *)
        fbLog( put := 'TX buffer overflow (single command)!' );
        state := 1;
    END_IF
END_IF
...

```

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

5.2.52 F\_iecApdu101ToAsduLen



Die Funktion berechnet für das IEC 60870-5-101 Protokoll die maximal verfügbare ASDU-Octetlänge anhand der konfigurierten APDU-Telegrammlänge und der Adressfeldlänge der Verbindungsschicht. Die maximal verfügbare ASDU-Länge wird z.B. bei der Konfiguration der ST\_IEC870\_5\_101TBuffer-Variablen benötigt. Diese Datenstruktur (TX/RX-Datenpuffer) wird beim Datenaustausch über das IEC60870-5-101 Low-Level Interface benutzt.

Syntax

```
FUNCTION F_iecApdu101ToAsduLen: BYTE
VAR_INPUT
    FRAMELength    : BYTE;
    eLinkAddrSize  : E_IEC870_5_101LinkAddrSize;
END_VAR
```

Eingänge

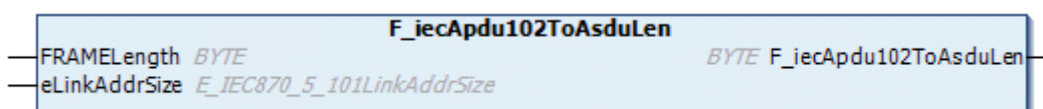
**FRAMELength:** Die max. verfügbare APDU-Telegrammlänge (siehe Kompatibilitätsliste).

**eLinkAddrSize:** Adressfeldlänge [[▶ 507](#)] der Verbindungsschicht.

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

5.2.53 F\_iecApdu102ToAsduLen



Die Funktion berechnet für das IEC 60870-5-102 Protokoll die maximal verfügbare ASDU-Octetlänge anhand der konfigurierten APDU-Telegrammlänge und der Adressfeldlänge der Verbindungsschicht. Die maximal verfügbare ASDU-Länge wird z.B. bei der Konfiguration der ST\_IEC870\_5\_102TBuffer-Variablen benötigt. Diese Datenstruktur (TX/RX-Datenpuffer) wird beim Datenaustausch über das IEC60870-5-102 Low-Level Interface benutzt.

### Syntax

```
FUNCTION F_iecApdu102ToAsduLen: BYTE
VAR_INPUT
    FRAMELength    : BYTE;
    eLinkAddrSize  : E_IEC870_5_101LinkAddrSize;
END_VAR
```

### 🔴 Eingänge

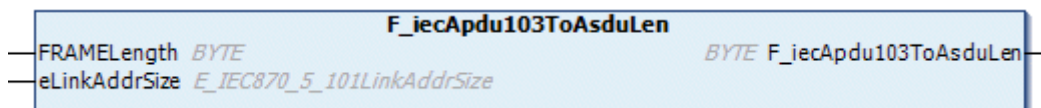
**FRAMELength:** Die max. verfügbare APDU-Telegrammlänge (siehe Kompatibilitätsliste).

**eLinkAddrSize:** Adressfeldlänge [► 507] der Verbindungsschicht.

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.2.54 F\_iecApdu103ToAsduLen



Die Funktion berechnet für das IEC 60870-5-103 Protokoll die maximal verfügbare ASDU-Octetlänge anhand der konfigurierten APDU-Telegrammlänge und der Adressfeldlänge der Verbindungsschicht. Die maximal verfügbare ASDU-Länge wird z.B. bei der Konfiguration der ST\_IEC870\_5\_103TBuffer-Variablen benötigt. Diese Datenstruktur (TX/RX-Datenpuffer) wird beim Datenaustausch über das IEC60870-5-103 Low-Level Interface benutzt.

### Syntax

```
FUNCTION F_iecApdu103ToAsduLen: BYTE
VAR_INPUT
    FRAMELength    : BYTE;
    eLinkAddrSize  : E_IEC870_5_101LinkAddrSize;
END_VAR
```

### 🔴 Eingänge

**FRAMELength:** Die max. verfügbare APDU-Telegrammlänge (siehe Kompatibilitätsliste).

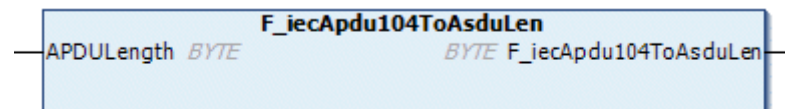
**eLinkAddrSize:** Adressfeldlänge [► 507] der Verbindungsschicht.



Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.2.55 F\_iecApdu104ToAsduLen



Die Funktion berechnet für das IEC 60870-5-104 Protokoll die maximal verfügbare ASDU-Octetlänge anhand der konfigurierten APDU-Länge. Die maximal verfügbare ASDU-Länge wird z.B. bei der Konfiguration der ST\_IEC870\_5\_101TBuffer-Variablen benötigt. Diese Datenstruktur (TX/RX-Datenpuffer) wird beim Datenaustausch über das IEC60870-5-104 Low-Level Interface benutzt.

Syntax

```
FUNCTION F_iecApdu104ToAsduLen: BYTE
VAR_INPUT
    APDULength : BYTE;
END_VAR
```

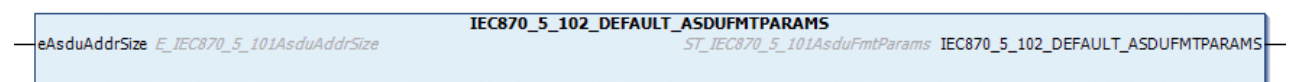
Eingänge

**APDULength:** Die verfügbare APDU-Länge (siehe Kompatibilitätsliste).

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.2.56 IEC870\_5\_102\_DEFAULT\_ASDFMTPARAMS



Diese Funktion setzt die durch die IEC 60870-5-102 Norm vorgegebenen Frame-Formatparameter [► 490]:

- Bytelänge der Übertragungsursache;
- Bytelänge der gemeinsamen ASDU-Adresse;
- Bytelänge der Informationsobjektadresse;

**Syntax**

```
FUNCTION IEC870_5_102_DEFAULT_ASDFMTPARAMS: ST_IEC870_5_101AsduFmtParams
VAR_INPUT
    eAsduAddrSize : E_IEC870_5_101AsduAddrSize;
END_VAR
```

 **Eingänge**

**eAsduAddrSize:** Oktetlänge der gemeinsamen ASDU-Adresse [► 506].

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.2.57 IEC870\_5\_104\_DEFAULT\_ASDFMTPARAMS**

**IEC870\_5\_104\_DEFAULT\_ASDFMTPARAMS**

*ST\_IEC870\_5\_101AsduFmtParams* IEC870\_5\_104\_DEFAULT\_ASDFMTPARAMS

Diese Funktion setzt die durch die IEC 60870-5-104 Norm vorgegebenen Frame-Formatparameter [► 490] auf die Standardwerte:

- Übertragungsursache: 2 Byte;
- Gemeinsame ASDU Adresse: 2 Byte;
- Informationsobjektadresse: 3 Byte.

**Syntax**

```
FUNCTION IEC870_5_104_DEFAULT_ASDFMTPARAMS: ST_IEC870_5_101AsduFmtParams
VAR_INPUT
    (* keine Parameter *)
END_VAR
```

 **Engänge**

Keine.

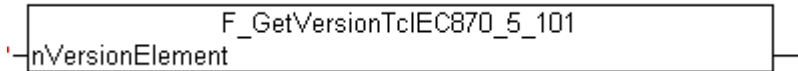
**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.2.58 Veraltet

### 5.2.58.1 F\_GetVersionTcIEC870\_5\_101

Bei dieser Funktion handelt es sich um eine veraltete Funktion die nur aus Kompatibilitätsgründen in der Bibliothek enthalten ist! Die aktuelle [Versionsinformation der Bibliothek befindet sich in einer globalen Konstanten \[► 560\]](#).



Mit dieser Funktion können Versionsinformationen der SPS-Bibliothek ausgelesen werden.

#### FUNCTION F\_GetVersionTcIEC870\_5\_101: UINT

```

VAR_INPUT
  nVersionElement : INT;
END_VAR
  
```

**nVersionElement** : Versionselement, das gelesen werden soll. Mögliche Parameter:

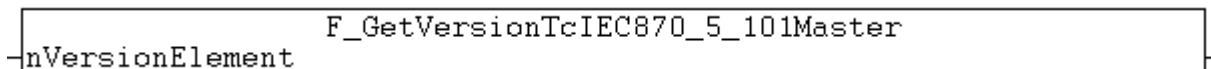
- 1 : major number;
- 2 : minor number;
- 3 : revision number;

#### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.2.58.2 F\_GetVersionTcIEC870\_5\_101Master

Bei dieser Funktion handelt es sich um eine veraltete Funktion die nur aus Kompatibilitätsgründen in der Bibliothek enthalten ist! Die aktuelle [Versionsinformation der Bibliothek befindet sich in einer globalen Konstanten \[► 560\]](#).



Mit dieser Funktion können Versionsinformationen der SPS-Bibliothek ausgelesen werden.

#### FUNCTION F\_GetVersionTcIEC870\_5\_101Master: UINT

```

VAR_INPUT
  nVersionElement : INT;
END_VAR
  
```

**nVersionElement** : Versionselement, das gelesen werden soll. Mögliche Parameter:

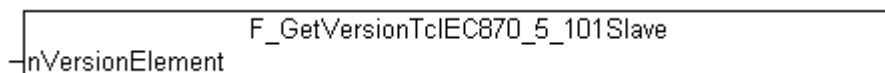
- 1 : major number;
- 2 : minor number;
- 3 : revision number;

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.2.58.3 F\_GetVersionTcIEC870\_5\_101Slave**

Bei dieser Funktion handelt es sich um eine veraltete Funktion die nur aus Kompatibilitätsgründen in der Bibliothek enthalten ist! Die aktuelle Versionsinformation der Bibliothek befindet sich in einer globalen Konstanten [► 560].



Mit dieser Funktion können Versionsinformationen der SPS-Bibliothek ausgelesen werden.

**FUNCTION F\_GetVersionTcIEC870\_5\_101Slave: UINT**

```

VAR_INPUT
    nVersionElement : INT;
END_VAR
  
```

**nVersionElement** : Versionselement, das gelesen werden soll. Mögliche Parameter:

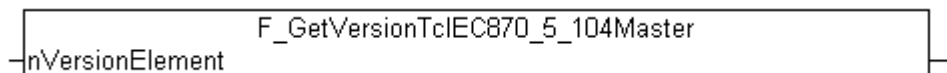
- 1 : major number;
- 2 : minor number;
- 3 : revision number;

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.2.58.4 F\_GetVersionTcIEC870\_5\_104Master**

Bei dieser Funktion handelt es sich um eine veraltete Funktion die nur aus Kompatibilitätsgründen in der Bibliothek enthalten ist! Die aktuelle Versionsinformation der Bibliothek befindet sich in einer globalen Konstanten [► 560].



Mit dieser Funktion können Versionsinformationen der SPS-Bibliothek ausgelesen werden.

**FUNCTION F\_GetVersionTcIEC870\_5\_104Master: UINT**

```
VAR_INPUT
    nVersionElement : INT;
END_VAR
```

**nVersionElement** : Versionselement, das gelesen werden soll. Mögliche Parameter:

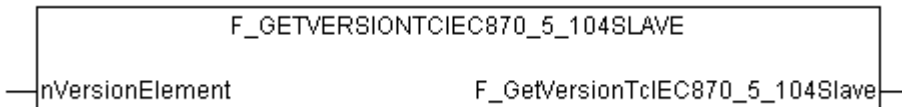
- 1 : major number;
- 2 : minor number;
- 3 : revision number;

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.2.58.5 F\_GetVersionTcIEC870\_5\_104Slave**

Bei dieser Funktion handelt es sich um eine veraltete Funktion die nur aus Kompatibilitätsgründen in der Bibliothek enthalten ist! Die aktuelle [Versionsinformation der Bibliothek befindet sich in einer globalen Konstanten \[► 560\]](#).



Mit dieser Funktion können Versionsinformationen der SPS-Bibliothek ausgelesen werden.

**FUNCTION F\_GetVersionTcIEC870\_5\_104Slave: UINT**

```
VAR_INPUT
    nVersionElement : INT;
END_VAR
```

**nVersionElement** : Versionselement, das gelesen werden soll. Mögliche Parameter:

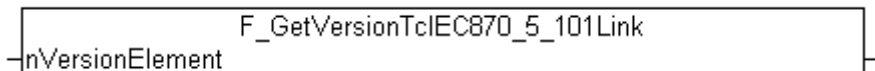
- 1 : major number;
- 2 : minor number;
- 3 : revision number;

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.2.58.6 F\_GetVersionTcIEC870\_5\_101Link

Bei dieser Funktion handelt es sich um eine veraltete Funktion die nur aus Kompatibilitätsgründen in der Bibliothek enthalten ist! Die aktuelle [Versionsinformation](#) der Bibliothek befindet sich in einer globalen Konstanten [[► 560](#)].



Mit dieser Funktion können Versionsinformationen der SPS-Bibliothek ausgelesen werden.

#### FUNCTION F\_GetVersionTcIEC870\_5\_101Link: UINT

```

VAR_INPUT
    nVersionElement : INT;
END_VAR

```

**nVersionElement** : Versionselement, das gelesen werden soll. Mögliche Parameter:

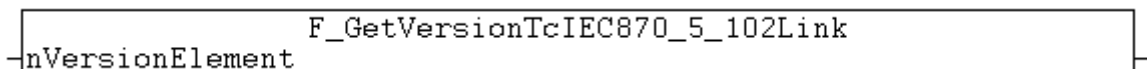
- 1 : major number;
- 2 : minor number;
- 3 : revision number;

#### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uutilities (System)

### 5.2.58.7 F\_GetVersionTcIEC870\_5\_102Link

Bei dieser Funktion handelt es sich um eine veraltete Funktion die nur aus Kompatibilitätsgründen in der Bibliothek enthalten ist! Die aktuelle [Versionsinformation](#) der Bibliothek befindet sich in einer globalen Konstanten [[► 560](#)].



Mit dieser Funktion können Versionsinformationen der SPS-Bibliothek ausgelesen werden.

#### FUNCTION F\_GetVersionTcIEC870\_5\_102Link: UINT

```

VAR_INPUT
    nVersionElement : INT;
END_VAR

```

**nVersionElement** : Versionselement, das gelesen werden soll. Mögliche Parameter:

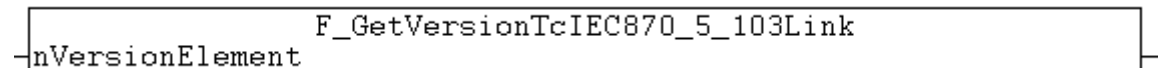
- 1 : major number;
- 2 : minor number;
- 3 : revision number;

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.2.58.8 F\_GetVersionTcIEC870\_5\_103Link**

Bei dieser Funktion handelt es sich um eine veraltete Funktion die nur aus Kompatibilitätsgründen in der Bibliothek enthalten ist! Die aktuelle Versioninformation der Bibliothek befindet sich in einer globalen Konstanten [► 560].



Mit dieser Funktion können Versionsinformationen der SPS-Bibliothek ausgelesen werden.

**FUNCTION F\_GetVersionTcIEC870\_5\_103Link: UINT**

```

VAR_INPUT
  nVersionElement : INT;
END_VAR
  
```

**nVersionElement** : Versionselement, das gelesen werden soll. Mögliche Parameter:

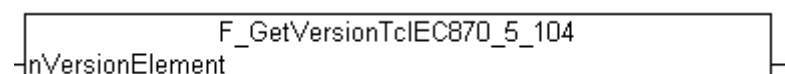
- 1 : major number;
- 2 : minor number;
- 3 : revision number;

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.2.58.9 F\_GetVersionTcIEC870\_5\_104**

Bei dieser Funktion handelt es sich um eine veraltete Funktion die nur aus Kompatibilitätsgründen in der Bibliothek enthalten ist! Die aktuelle Versioninformation der Bibliothek befindet sich in einer globalen Konstanten [► 560].



Mit dieser Funktion können Versionsinformationen der SPS-Bibliothek ausgelesen werden.

**FUNCTION F\_GetVersionTcIEC870\_5\_104: UINT**

```
VAR_INPUT
  nVersionElement : INT;
END_VAR
```

**nVersionElement** : Versionselement, das gelesen werden soll. Mögliche Parameter:

- 1 : major number;
- 2 : minor number;
- 3 : revision number;

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3 Datentypen****5.3.1 ST\_IEC870\_5\_101ExSystemInterface****Syntax**

```
TYPE ST_IEC870_5_101ExSystemInterface :
STRUCT
  device      : ST_IEC870_5_101DeviceInterface;
  service     : ST_IEC870_5_101SystemServices;
  hSOTable    : T_HAODBTABLE;
END_STRUCT
END_TYPE
```

**device**: Interne Kommunikationsschnittstelle des IEC-Gerätes [[▶ 489](#)].

**service**: IEC-Gerätedienste [[▶ 481](#)];

**hSOTable**: Systemobjekt Datenbank Handle [[▶ 531](#)];

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.2 ST\_IEC870\_5\_101SystemInterface****Syntax**

```
TYPE ST_IEC870_5_101SystemInterface :
STRUCT
  device : ST_IEC870_5_101DeviceInterface;
END_STRUCT
```



```

service : ST_IEC870_5_101SlaveServices;
END_STRUCT
END_TYPE

```

**device:** Kommunikationsschnittstelle [► 489] des IEC-Gerätes. Diese Variable dient anderen SPS Funktionen oder Funktionsbausteinen als Kommunikationsschnittstelle zum IEC-Gerät (hier: Unterstation).

**service:** IEC-Gerätedienste;

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.3 ST\_IEC870\_5\_101SystemServices

Diese strukturierte Variable wird von der Station nur für interne Kommunikation verwendet.

**Syntax**

```

TYPE ST_IEC870_5_101SystemServices
STRUCT
(**)
END_STRUCT
END_TYPE

```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.4 ST\_IEC870\_5\_101TBuffer

Diese Datenstruktur (TX/RX-Datenpuffer) wird beim Datenaustausch über das IEC60870-5-104/101 Low-Level-Interface benutzt.

**Syntax**

```

TYPE ST_IEC870_5_101TBuffer :
STRUCT
eDbg          : E_IEC870_5_101FifoDbgFlags :=eIEC870_FIFO_DBG_OFF;
asduFmt       : ST_IEC870_5_101AsduFmtParams;
asduSize      : BYTE := 0;
mode          : DWORD := 0;
dataLink      : ST_IEC870_5_101DataLink;
bOverwrite    : BOOL := FALSE;
END_STRUCT
END_TYPE

```

**eDbg:** Konfiguriert die Debug-Ausgabe [► 514] der ASDU's.

**asduFmt:** ASDU-Formatparameter [► 490].

**asduSize:** Maximale Bytelänge der ASDU.

**mode:** Reserviert, wird zur Zeit nicht benutzt. Dieser Wert sollte Null sein.

**dataLink:** Auf die Elemente dieser Datenstruktur sollte nicht direkt, sondern nur mit einer Instanz des `FB_IEC870_5_101TBufferCtrl` [► 403]-Funktionsbausteins zugegriffen werden.

Der TX/RX-Datenpuffer verwendet intern zwei Sende-Fifos und einen Empfangs-Fifo:

1. Class 1 Sende-Fifo mit (hochprioren) Daten;
2. Class 2 Sende-Fifo mit (niederprioren) Daten;
3. Receive-Fifo (für Class 1 und Class 2 Daten);

Die unteren Transportfunktionen der Bibliothek leeren zuerst den Class 1-Fifo und dann den Class 2-Fifo. Die Class 2 Daten werden nur dann versendet wenn der Class 1-Fifo keine zu versendenden Daten enthält.

Jeder der internen Fifos hat eine feste Größe von 200000 Bytes. Dies dürfte für die meisten Anwendungen ausreichen, weil die Anzahl der Frames die einmalig verschickt werden können durch die `iK` und `iW`-Protokollparameter begrenzt wird. Erfahrungsgemäß können in jedem Fifo ca. 2000 ASDUs mit einem Informations-Element (Objekt) oder ca. 600 ASDUs mit einer Sequence von 100 Informations-Elementen (Objekten) abgelegt werden.

Wenn eine größere Anzahl der zu versendenden oder zu empfangenen Frames zwischengespeichert werden soll (z.B. >2000), so können diese in externen, vom SPS-Programmierer festgelegten Puffern/Fifos zwischengehalten werden. Die SPS-Applikation kann dann zur Laufzeit die TwinCAT-Sende-Fifos mit den eigenen Fifo-Einträgen nachfüllen oder bei vielen empfangenen Frames den TwinCAT-Receive-Fifo leeren. Eine andere Möglichkeit ist z.B. zwei Puffer zu benutzen und diese abwechselnd zu füllen/lesen und an den Kommunikationsbaustein übergeben.

**bOverwrite:** Aktiviert/deaktiviert das Überschreiben der ältesten Meldungen im Sendepuffer bei der Überschreitung der max. zulässigen Puffergröße. Dieser Parameter sollte nur im Offline-Mode aktiviert werden (d.h. wenn die Verbindung unterbrochen wurde) und wenn Offline-Datenspeicherung zusätzlich über den **bRetainBuffer**-Parameter in den Protokollparametern (`ST_IEC870_5_10xProtPara`) aktiviert wurde. Im Online-Mode (d.h. wenn der Datentransfer gestartet wurde) sollten keine älteren Meldungen überschrieben werden (sonst fehlen möglicherweise einige Zwischenwerte).

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.3.5 ST\_IEC870\_5\_101AODBEntry

Ein IEC Applikationsobjekt-Datenbankeintrag. Die IEC-Applikationsobjekt Datenbank wird als Array-Variable von Typ `ST_IEC870_5_101AODBEntry` deklariert. Die Membervariablen dieses strukturierten Typs werden nicht direkt, sondern nur mit Hilfe der zur Verfügung stehenden Funktionen oder Funktionsbausteine verändert. Die `F_iecInIAOEntry` [► 435] gehört z.B. zu einer solchen Funktion.

### Syntax

```
TYPE ST_IEC870_5_101AODBEntry :
STRUCT
    aObj : ST_IEC870_5_101AOEntry;
    ctrl : FB_IEC870_5_101AOctrl;
END_STRUCT
END_TYPE
```

Beispiel für eine Deklaration von einer Applikationsdatenbank mit 2001 Objekten:

**Beispiel**

```
VAR_GLOBAL
  slaveIAODB : ARRAY[0..2000] OF ST_IEC870_5_101AODBEntry;
END_VAR
```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.6 ST\_IEC870\_5\_101AOGen**

Variablen von diesem Typ repräsentieren ein ASDU-Objekt.

**Syntax**

```
TYPE ST_IEC870_5_101AOGen:
STRUCT
  head : ST_IEC870_5_101FifoHead :=( source := ( link := 0, addr := 0 ),
                                     target := ( link := 0, addr := 0 ),
                                     ctrl := 0 );

  ident: ST_IEC870_5_101DataUnit_Ident := ( eType      := ASDU_TYPEUNDEF,
                                           nObj       := 0,
                                           bSQ        := FALSE,
                                           bT          := FALSE,
                                           bPN         := FALSE,
                                           nORG        := 0,
                                           asduAddr:= 0,
                                           eCOT        := eIEC870_COT_UNUSED,
                                           eClass      := eIEC870_Class_None );
  info  : ST_IEC870_5_101AOInfoObj := ( objAddr := 0, stream := ( length := 0 ) );
END_STRUCT
END_TYPE
```

**head:** Header (reserviert).

**ident:** Identifikationsfelder der Dateneinheit [▶ 484] (ASDU).

**info:** Informationsobjekt-/Informationselement-Datenfeld [▶ 485].

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.7 ST\_IEC870\_5\_101AOCfg**

ASDU-Objekt-Konfigurationsparameter. Diese Parameter werden während der Konfiguration der Datenpunkte ( F\_ieclnitAOEntry oder F\_iecAddTableEntry ) gesetzt und sollten nicht direkt aus der SPS-Applikation beschrieben werden.

**Syntax**

```

TYPE ST_IEC870_5_101AOCfg:
STRUCT
  group      : DWORD := IEC870_GRP_INROGEN;
  multiplier : BYTE := 0;
  ioMapType  : E_IEC870_5_101IOMappingType := MAP_AREA_NONE;
  byteOfs   : UXINT := 0;
  bitOfs    : UDINT := 0;
  hUser     : UXINT := 0;
END_STRUCT
END_TYPE

```

**group:** Object-Group-Konfigurationsflags. Hier finden Sie die [Beschreibung aller Group-Flags \[► 554\]](#). Die Flags können mit OR-Verknüpfung kombiniert werden. Nicht alle Kombinationen sind aber sinnvoll!

**multiplier:** Basis-Zykluszeit-Multiplikator für zyklische/periodische Datenübertragung. 0 = Deaktiviert. Die Basis-Zykluszeit kann über den *tPerCyclicBase*-Parameter in den [Systemparametern \[► 487\]](#) konfiguriert werden.

**ioMapType:** [TwinCAT SPS-Prozessdatenbereich \[► 506\]](#). Dieser Parameter legt fest wie die TwinCAT SPS und IEC-Applikationsobjekt Prozessdaten gemappt werden sollen.

**byteOfs:** TwinCAT SPS-Prozessdaten-Byte-Offset (x86 Plattform => max. 32 Bit, x64 Plattform => max. 64 Bit).

**bitOfs:** TwinCAT SPS-Prozessdaten-Bit-Offset.

**hUser:** Optionales Benutzer-Handle. Frei definierbarer Wert, auch Zeiger (x86 Plattform => 32 Bit, x64 Plattform => 64 Bit).

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.8 ST\_IEC870\_5\_101DataUnit\_Ident**

Identifikationsfeld der Dateneinheit (ASDU).

**Syntax**

```

TYPE ST_IEC870_5_101DataUnit_Ident:
STRUCT
  eType      : E_IEC870_5_101TcTypeID;
  bSQ       : BOOL;
  nObj      : BYTE;
  bT        : BOOL;
  bPN       : BOOL;
  eCOT      : E_IEC870_5_101COTType;
  nORG      : BYTE;
  asduAddr  : DWORD;
  eClass    : E_IEC870_5_101ClassType;
END_STRUCT
END_TYPE

```

**eType:** [IEC 60870-5-101/104 ASDU-Typkennung \[► 500\]](#) := <0..255>.

**bSQ:** Sequenz-Flag := <FALSE = Single, TRUE = Sequence>.

**nObj:** Anzahl der Informationsobjekte oder Informationselemente := <0..127>.

**bT**: Test-Flag (wird kaum benutzt).

**bPN**: Positive/negative Bestätigung := <FALSE = Positiv, TRUE = Negativ>.

**eCOT**: IEC 60870-5-101/104 Übertragungsursache [[▶ 509](#)] (COT) := <0..63>.

**nORG**: Quelladresse.

**asduAddr**: Gemeinsame ASDU-Adresse (CAA).

**eClass**: [Prioritätsklasse](#) [[▶ 509](#)] (TX-Daten).

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.9 ST\_IEC870\_5\_101AOInfoObj**

Informationsobjektbeschreibung.

**Syntax**

```

TYPE ST_IEC870_5_101AOInfoObj :
STRUCT
    objAddr : DWORD;
    stream : ST_IEC870_5_101Stream;
END_STRUCT
END_TYPE
    
```

**objAddr** : Informationsobjektadresse (IOA).

**stream** : [Informationselemente-Datenpuffer](#) [[▶ 485](#)].

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.10 ST\_IEC870\_5\_101Stream**

Variablen von diesem Typ werden als Datenpuffer (Stream) benutzt. Im Stream werden "rohe" Datenbytes zwischengespeichert, mit der Reihenfolge, wie sie später verschickt werden sollen oder wie sie empfangen wurden. Beim Senden oder Empfang wird zuerst das Datenbyte Null verschickt bzw. empfangen.

**Syntax**

```

TYPE ST_IEC870_5_101Stream :
STRUCT
    length : DWORD := 0;
    
```

```

data : ARRAY[0..IEC870_MAX_ASDU_DATA_BYTE] OF BYTE;
END_STRUCT
END_TYPE

```

**length:** Aktuelle Anzahl der Datenbytes im Stream.

**data:** Datenpuffer (Byte-Array).

Speicherdarstellung einer Stream-Variablen mit einer zwischengespeicherten DWORD-Variablen mit dem Wert: **16#BECF1234**. Beachten Sie die vertauschten Datenbytes im Intel-Format!

length	data											
4	16#34	16#12	16#CF	16#BE	...	...	...	...	...	...	...	IEC870 _MAX_ ASDU_ DATA_ BYTE

Benutzen Sie die in der Tabelle aufgeführten Funktionen um den Speicherinhalt einer Stream-Variablen zu verändern:

Function	Description
<a href="#">F_iecResetStream [► 451]</a>	Stream-Initialisierung/-Reset.
<a href="#">F_iecCopyBufferToStream [► 442]</a>	Kopiert Datenbytes von einer externen Puffervariablen in den Stream. Der Speicherinhalt der Stream-Variablen wird vergrößert.
<a href="#">F_iecCopyBufferToStreamByOffset [► 443]</a>	Kopiert Datenbytes von einer externen Puffervariablen in den Stream ab einem Byteoffset. Der Speicherinhalt der Stream-Variablen wird vergrößert.
<a href="#">F_iecCopyStreamToBuffer [► 444]</a>	Kopiert Datenbytes vom Stream in eine externe Puffervariable. Der Speicherinhalt der Stream-Variablen bleibt unverändert.
<a href="#">F_iecCopyStreamToBufferByOffset [► 446]</a>	Kopiert Datenbytes vom Stream ab einem Byteoffset in eine externe Puffervariable. Der Speicherinhalt der Stream-Variablen bleibt unverändert.
<a href="#">F_iecCopyStreamToStream [► 447]</a>	Kopiert Datenbytes von einem Stream zum anderen Stream. Der Speicherinhalt der Quell-Variablen bleibt unverändert. Der Speicherinhalt der Ziel-Variablen wird vergrößert.
<a href="#">F_iecMoveStreamToBuffer [► 449]</a>	Kopiert Datenbytes vom Stream in eine externe Puffervariable und löscht anschließend die kopierten Datenbytes im Stream. Der Speicherinhalt der Stream-Variablen wird verkleinert.
<a href="#">F_iecMoveStreamToStream [► 450]</a>	Kopiert Datenbytes vom Quell-Stream zum Ziel-Stream und löscht anschließend die kopierten Datenbytes im Quell-Stream. Der Speicherinhalt der Quell-Variablen wird verkleinert und der Ziel-Variablen vergrößert.

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.3.11 ST\_IEC870\_5\_101SystemParams

Systemparameter der Station.

### Syntax

```

TYPE ST_IEC870_5_101SystemParams :
STRUCT
  asduAddr      : DWORD := 11;
  asduAddrRev   : DWORD := 22;

  asduFmt       : ST_IEC870_5_101AsduFmtParams := ( eCOTSize := eIEC870_COT_TwoOctets,
                                                    eAsduAddrSize := eIEC870_AsduAddr_TwoOctets,
                                                    eObjAddrSize := eIEC870_ObjAddr_ThreeOctets
);
  bEndOfInit    : BOOL := TRUE;

  bSyncTime     : BOOL := TRUE;
  bSyncPCTime   : BOOL := FALSE;

  bUsePCTime    : BOOL := TRUE;
  tSyncToPCTime : TIME := T#0s;
  sPCTimeNetID  : T_AmsNetID := '';

  bTimeCOT3     : BOOL := FALSE;
  tSyncTimeout  : TIME := T#0m;

  bACTCONRes    : BOOL := TRUE;
  bACTTERMRes   : BOOL := TRUE;

  bPerCyclic    : BOOL := FALSE;
  tPerCyclicBase : TIME := T#60s;

  bBackScan     : BOOL := FALSE;
  tBackScanCycle : TIME := T#60s;

  bPerFRZ       : BOOL := FALSE;
  tPerFRZCycle  : TIME := T#60s;
  bSyncPerFRZtoHour : BOOL := FALSE;

  tSelExecTimeout : TIME := T#10s;
  tActConTimeout  : TIME := T#5s;
  tActTermTimeout : TIME := T#30s;
  tDeactConTimeout : TIME := T#5s;
  tReadResTimeout : TIME := T#5s;

  dbgMode       : DWORD := 0;
  orgAddr       : BYTE := 1;

  bOverwrite    : BOOL := FALSE;
  bCBOActTerm   : BOOL := FALSE;
  bAdoptSuccTSC : BOOL := FALSE;
END_STRUCT
END_TYPE

```

**asduAddr:** Gemeinsame ASDU-Adresse (CAA) in Standardrichtung. Default: 11.

**asduAddrRev:** Gemeinsame ASDU-Adresse (CAA) in Umkehrrichtung. Default: 22.

**asduFmt:** [ASDU-Formatparameter \[► 490\]](#) (z.B. Oktetlänge der Übertragungsursache-, ASDU-Adresse- und der Object-Adresse);

**bEndOfInit:** Wenn TRUE, sende M\_EI\_NA\_1 (End of init) nachdem die Stationsinitialisierung abgeschlossen wurde. Eine Stationsinitialisierung wird beim Systemstart oder nach einem Prozess Reset durchgeführt. Default: TRUE.

**bSyncTime:** Wenn TRUE, aktiviere IEC-Systemzeit Synchronisation beim Empfang des C\_CS\_NA\_1-Kommandos. Default: TRUE.

**bSyncPCTime:** Wenn TRUE, synchronisiere neben der IEC-Systemzeit auch die Systemzeit des TwinCAT PC's (die Windows-Systemzeit in der Taskleiste). Default: FALSE.

**bUsePCTime:** Wenn TRUE, synchronisiere die IEC-Systemzeit mit der Systemzeit des PC's. Nach der Initialisierung der Unterstation wird die IEC-Systemzeit zuerst mit der Systemzeit des TwinCAT PC's (Windows-Systemzeit) synchronisiert. Default: TRUE.

**tSyncToPCTime** : Steuert die zyklische Synchronisation der IEC-Systemzeit mit der Systemzeit des TwinCAT PC's (im Online- und Offline-Mode). Implementiert in IEC870-5-104 slave Bibliothek v3.0.3 und höher.

Die Zeit bestimmt die Zykluszeit in der die Synchronisation durchgeführt wurde. Bei einem Wert = T#0s ist die zyklische Synchronisation deaktiviert. Default: T#0s.

**sPCTimeNetID**: TwinCAT Netzwerkadresse des PCs dessen Systemzeit bei der Synchronisierung benutzt werden soll. Für den lokalen PC kann auch ein Leerstring angegeben werden. Default: Leerstring = Lokaler PC.

**bTimeCOT3**: Wenn TRUE, sende die Systemzeit zur Zentralstation mit der Übertragungsursache <3> *Spontan* beim Stundenwechsel. Default: FALSE.

**tSyncTimeout**: Uhrzeitsynchronisationsintervall-Timeoutüberwachung. Implementiert in IEC870-5-101/104 slave Bibliothek v2.0.0 und höher. In früheren Versionen wird dieser Parameter nicht benutzt. Die Zeitstempel haben einen IV-Quality-Flag (invalid). Bei einem Wert <> T#0m (z.B. T#60m) wird das IV-Quality-Flag bei allen darauf folgenden Zeitstempeln auf Invalid gesetzt wenn innerhalb einer Stunde keine Uhrzeitsynchronisation durchgeführt wurde. Das IV-Quality-Flag des Zeitstempels kann vom Master als Indikator für die Qualität des Zeitstempels benutzt werden. Bei tSyncTimeout = T#0m ist die Überwachung nicht aktiv.

**bACTCONRes**: Wenn TRUE, sende ACTCON response.

**bACTTERMRes**: Wenn TRUE, sende ACTTERM response.

**bPerCyclic**: Wenn TRUE, aktiviere zyklische/periodische Datenübertragung.

**tPerCyclicBase**: Basiszeit der zyklischen/periodischen Datenübertragung.

**bBackScan**: Wenn TRUE, aktiviere Hintergrundabfrage.

**tBackScanCycle**: Hintergrundabfrage-Zykluszeit.

**bPerFRZ**: Wenn TRUE, aktiviere lokales Umspeichern/Rücksetzen der Zählerwerte.

**tPerFRZCycle**: Zykluszeit für lokales Umspeichern/Rücksetzen.

**bSyncPerFRZtoHour** : Synchronisiere periodisches/zyklisches Umspeichern der Zähler zur ganzen Stunde. Default: FALSE (deaktiviert).

**tSelExecTimeout**: Max. Timeoutzeit zwischen dem Anwahlbefehl und dem Ausführungsbefehl (nur Zentralstation).

**tActConTimeout** :Max. Timeoutzeit für den Empfang der Bestätigung der Befehllaktivierung (nur Zentralstation).

**tActTermTimeout** : Max. Timeoutzeit für den Empfang der Befehlterminierung (nur Zentralstation).

**tDeactConTimeout** : Max. Timeoutzeit für den Empfang der Bestätigung des Befehlabbruchs (nur Zentralstation).

**tReadResTimeout** : Max. Timeoutzeit für die Ausführung des Lesebefehls (C\_RD\_NA\_1) (nur Zentralstation).

**dbgMode**:Debug-Flags:



Wert	Beschreibung
IEC870_DEBUGMODE_DISABLED	Loggen deaktiviert
IEC870_DEBUGMODE_ASDU	Die ASDUs werden als hexadezimale Ausgabe im Application-Log geloggt.
IEC870_DEBUGMODE_DEVSTATE	Statusänderungen der Station werden im Application-Log geloggt.
IEC870_DEBUGMODE_LINKLAYER	Link layer frames werden im Application-Log geloggt IEC870-5-101 slave Bibliothek: bei allen Versionen verfügbar; IEC870-5-104 slave Bibliothek: ab v2.0.0 und höher verfügbar;
IEC870_DEBUGMODE_LINKERROR	Link layer Fehlermeldungen werden im Application-Log geloggt.

Die Flags können in der gewünschten Kombination verodert werden.

**orgAddr:** Quelladresse (wird zur Zeit nicht benutzt).

**bOverwrite:** Aktiviert/deaktiviert das Überschreiben der ältesten Meldungen im Sendepuffer bei der Überschreitung der max. zulässigen Puffergröße. Dieser Parameter ist nur im Offline-Mode aktiv (d.h. wenn die Verbindung unterbrochen wurde) und wenn Offline-Datenspeicherung zusätzlich über den **bRetainBuffer**-Parameter in den Protokollparametern (ST\_IEC870\_5\_104ProtPara) aktiviert wurde. Im Online-Mode (d.h. wenn der Datentransfer gestartet wurde) werden keine älteren Meldungen überschrieben (sonst würden möglicherweise einige Zwischenwerte fehlen).

**bCBOActTerm :** Nur bei: C\_BO\_NA\_1, C\_BO\_TA\_1, C\_BO\_TB\_1. Verwende/erwarte Übertragungsursachen: actTerm, deact, deactCon Bitstringbefehl. Default: FALSE (nicht verwenden).

**bAdoptSuccTSC :** Nur bei: C\_TS\_TA\_1: Übernehme/synchronisiere zum empfangenen Zählerwert.

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.12 ST\_IEC870\_5\_101DeviceInterface

**Syntax**

```

TYPE ST_IEC870_5_101DeviceInterface :
STRUCT
    asduAddr      : DWORD;
    asduAddrRev  : DWORD;
    dataLink      : ST_IEC870_5_101DataLink;
    clock         : ST_IEC870_5_101SystemRTC;
    comp         : ST_IEC870_5_101DeviceCompatibility;
    errors        : FB_IEC870_5_101ErrorFifo;
    status        : DWORD;
END_STRUCT
END_TYPE
    
```

**asduAddr:** Gemeinsame ASDU-Adresse in Standardrichtung;

**asduAddrRev:** Gemeinsame ASDU-Adresse in Umkehrrichtung.

**dataLink:** Interne ASDU-Send/Receive-Fifos;

**clock:** IEC-Systemzeitobjekt;

**comp:** Kompatibilitätsliste mit der Zuordnung der ASDU-Typen zu den Übertragungsursachen;

**errors:** Gerätefehler-Fifo [► 420];

**status:** Globaler-Gerätestatus. 1=OK, 0=Not OK;

#### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.13 ST\_IEC870\_5\_101AsduFmtParams

ASDU Frameformatparameter.

#### Syntax

```

TYPE ST_IEC870_5_101AsduFmtParams :
STRUCT
  eCOTSize      : E_IEC870_5_101COTSize      := eIEC870_COT_TwoOctets;
  eAsduAddrSize : E_IEC870_5_101AsduAddrSize := eIEC870_AsduAddr_TwoOctets;
  eObjAddrSize  : E_IEC870_5_101ObjAddrSize := eIEC870_ObjAddr_ThreeOctets;
END_STRUCT
END_TYPE

```

**eCOTSize** : Oktetgröße der Übertragungsursache [► 507] (COT).

**eAsduAddrSize** : Oktetgröße der gemeinsamen ASDU adresse [► 506] (CAA).

**eObjAddrSize** : Oktetgröße der Informationsobjektadresse [► 508] (IOA).

#### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.14 ST\_IEC870\_5\_101ErrorFifoEntry

TwinCAT IEC 60870-5-10x Gerätefehler.

#### Syntax

```

TYPE ST_IEC870_5_101ErrorFifoEntry :
STRUCT
  nErrId : UDINT;
  eSrcId : E_IEC870_5_101ErrorSourceID;
END_STRUCT
END_TYPE

```

**nErrID:** Fehlercode (abhängig vom Protokoll):

- [TwinCAT IEC 60870-5-101 Fehlercodes \[▶ 685\]](#);
- [TwinCAT IEC 60870-5-102 Fehlercodes \[▶ 694\]](#);
- [TwinCAT IEC 60870-5-103 Fehlercodes \[▶ 701\]](#);
- [TwinCAT IEC 60870-5-104 Fehlercodes \[▶ 708\]](#);

**eSrcID:** Fehlerquelle [▶ 508]. Die Fehlerquelle gibt nähere Informationen über die Fehlerursache, die Komponente oder den Dienst der den Fehler gemeldet hat.

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.15 ST\_IEC870\_5\_101AcquisitionParams**

Konfigurationsparameter für die zyklische Datenerfassung. In den meisten Fällen soll von der Zentralstation einmalig eine Initialisierungssequenz ausgeführt werden. In dieser Sequenz sind einige Befehle enthalten, die nach der Herstellung der Kommunikationsverbindung und/oder nach dem Empfang von ASDU: M\_EI\_NA\_1 (Ende der Initialisierung) ausgeführt werden. Zu solchen Befehlen gehört z.B. der Uhrzeitsynchronisationsbefehl. Danach soll die Zentralstation z. B. kontinuierlich zyklische Abfragebefehle zur Unterstation senden. Bei einer IEC 60870-5-104 Zentralstation wird der Datenaustausch aktiviert, nachdem die Kommunikationsverbindung hergestellt und „STARTDT“ (Start Data Transfer) empfangen wurde. Im Fall einer IEC 60870-5-101 Zentralstation nach dem Verbindungsreset (Reset Of Link).

Ab der Tc2\_IEC60870\_5\_10x.compiled-library v3.0.7.0 kann der Zeitpunkt der Abarbeitung der Initialisierungssequenz und der zyklischen Abfragebefehle konfiguriert werden. Die Applikation kann wählen ob die Initialisierungssequenz nach dem STARTDT oder erst nach dem Empfang von M\_EI\_NA\_1 ausgeführt werden soll.

Die Strukturkomponente *arrInitSeq* dient der Konfiguration der Initialisierungssequenz. Dort können Sie einige unterschiedliche Befehle vorkonfigurieren.

Die Strukturkomponente *eMode* legt fest wann diese Befehle ausgeführt werden sollen. Die Initialisierungssequenz kann z. B. nach der Herstellung der Verbindung (nach STARTDT oder Reset Of Link) oder erst nach dem Empfang von ASDU: M\_EI\_NA\_1 (Ende der Initialisierung) ausgeführt werden.

Die zyklischen Abfragebefehle werden über die Strukturkomponenten *testCmd*, *clockSync*, *arrGenro*, *arrCoro* oder *genCmd* konfiguriert.

**Syntax**

```

TYPE ST_IEC870_5_101AcquisitionParams :
STRUCT
  eMode : E_IEC870_5_101AcquisitionMode := E_IEC870_5_101AcquisitionMode.InitSeq_CyclicPoll;

  arrInitSeq : ARRAY[IEC870_MIN_ISTEP..IEC870_MAX_ISTEP] OF E_IEC870_5_101InitSeqStep :=
    [eIEC870_ISTEP_TEST,
     eIEC870_ISTEP_CLOCK,
     eIEC870_ISTEP_GENRO,
     eIEC870_ISTEP_CORO,
     eIEC870_ISTEP_UNUSED,
     eIEC870_ISTEP_UNUSED,
     eIEC870_ISTEP_UNUSED];

  testCmd : ST_IEC870_5_101TestPollParams :=
    (asduAddr := IEC870_ASDUADDR_SYSPARA, tPollCycle := T#60S, bEnable := TRUE);

```

```

clockSync : ST_IEC870_5_101ClockPollParams :=
  (asduAddr := IEC870_ASUADDR_SYSPARA, tPollCycle := T#60S, bEnable := TRUE);

arrGenro : ARRAY[0..16] OF ST_IEC870_5_101GenroPollParams :=
  [(eQOI := eIEC870_QOI_INROGEN, tPollCycle := T#60S, bEnable := TRUE),
  (eQOI := eIEC870_QOI_INRO1),
  (eQOI := eIEC870_QOI_INRO2),
  (eQOI := eIEC870_QOI_INRO3),
  (eQOI := eIEC870_QOI_INRO4),
  (eQOI := eIEC870_QOI_INRO5),
  (eQOI := eIEC870_QOI_INRO6),
  (eQOI := eIEC870_QOI_INRO7),
  (eQOI := eIEC870_QOI_INRO8),
  (eQOI := eIEC870_QOI_INRO9),
  (eQOI := eIEC870_QOI_INRO10),
  (eQOI := eIEC870_QOI_INRO11),
  (eQOI := eIEC870_QOI_INRO12),
  (eQOI := eIEC870_QOI_INRO13),
  (eQOI := eIEC870_QOI_INRO14),
  (eQOI := eIEC870_QOI_INRO15),
  (eQOI := eIEC870_QOI_INRO16)];

arrCoro : ARRAY[0..4] OF ST_IEC870_5_101CoroPollParams :=
  [(eRQT := eIEC870_RQT_REQCOGEN, eFRZ := eIEC870_FRZ_READ, tPollCycle := T#60S, bEnable := TR
UE),
  (eRQT := eIEC870_RQT_REQCO1),
  (eRQT := eIEC870_RQT_REQCO2),
  (eRQT := eIEC870_RQT_REQCO3),
  (eRQT := eIEC870_RQT_REQCO4)];

genCmd : ST_IEC870_5_101GenCmdPollParams :=
  (tPollCycle := T#1H, bEnable := FALSE);

delayCmd : ST_IEC870_5_101DelayPollParams := (tDelay := T#5S);

eAODBType : E_IEC870_5_101AODBType := eIEC870_AODB_STATIC;
END_STRUCT
END_TYPE

```

**eMode:** [Konfiguration des Triggers/Auslösers \[► 498\]](#) für die Abarbeitung der Initialisierungssequenz (arrInitSeq) und der zyklischen Abfragebefehle. Da nicht jeder Server immer die ASDU: M\_EI\_NA\_1 (Ende der Initialisierung) sendet, wurde folgende Standardeinstellung gewählt:  
E\_IEC870\_5\_101AcquisitionMode.InitSeq\_CyclicPoll.

In der Standardeinstellung werden nach dem Start des Datentransfers (IEC 60870-5-104, STARTDT) oder dem Reset der Verbindung (IEC 60870-5-101) zuerst alle Befehle der Initialisierungssequenz einmalig, danach die zyklischen Abfragebefehle fortlaufend ausgeführt. Die ASDU: M\_EI\_NA\_1 wird nicht ausgewertet und ignoriert.

Wenn die Initialisierungssequenz beim Empfang von M\_EI\_NA\_1 ausgeführt werden soll, muss eine andere Einstellung gewählt werden, z. B.:  
E\_IEC870\_5\_101AcquisitionMode.CyclicPoll\_OnEndOfInit\_InitSeq\_CyclicPoll oder  
E\_IEC870\_5\_101AcquisitionMode.OnEndOfInit\_InitSeq\_CyclicPoll.

**arrInitSeq:** [Initialisierungssequenz \[► 515\]](#). Die Standardkonfiguration der Initialisierungssequenz führt folgende Befehle aus:

- Testbefehl (eIEC870\_ISTEP\_TEST);
- Uhrzeitsynchronisationsbefehl (eIEC870\_ISTEP\_CLOCK);
- Generalabfrage (eIEC870\_ISTEP\_GENRO, Gruppe Allgemein);
- Zählerabfrage (eIEC870\_ISTEP\_CORO, Zählergruppe Allgemein, Zähler lesen);

**testCmd:** [Parameter für zyklische Testbefehle \[► 493\]](#). Standardwert: Aktiv, ein Testkommando alle 60s.

**clockSync:** [Parameter für zyklische Uhrzeitsynchronisationsbefehle \[► 494\]](#). Standardwert: Aktiv, Uhrzeitsynchronisation alle 60s.

**arrGenro:** [Parameter für zyklische Stationsabfragebefehle \[► 494\]](#). Es können Stationsabfragen von bis zu 17 Datenpunktgruppen konfiguriert werden. Standardwert: Aktiv, eine Stationsabfrage der Gruppe 'Allgemein' alle 60s.

**arrCoro:** Parameter für zyklische Zählerabfragebefehle [► 495]. Es können Zählerabfragen von bis zu 5 Zählergruppen konfiguriert werden. Standardwert: Aktiv, eine Zählerabfrage der Gruppe Allgemein alle 60s.

**genCmd:** Parameter für zyklische Übertragung der Datenpunkte in Steuerungsrichtung [► 496] (Einzelbefehl, Doppelbefehl, Sollwerte usw.). Standardwert: Nicht aktiv, die Befehle werden alle 60min. übertragen.

**delayCmd:** Parameter für die Verzögerung des nächsten Initialisierungsschrittes [► 497] während der Ausführung der Initialisierungssequence. Standardwert: Der nächste Initialisierungsschritt wird um 5s verzögert.

**eAODBType:** Applikationsdatenbank-Typ [► 515]. Dieser Parameter legt fest wie die Datenpunkte in der Applikationsdatenbank abgespeichert werden.

**Beispiel in ST**

Im folgenden Programmausschnitt wird die zyklische Datenerfassung wie folgt konfiguriert: Alle Initialisierungsschritte werden deaktiviert. Das zyklische Testkommando und Uhrzeitsynchronisationskommando werden auch deaktiviert. Zusätzlich zu der Standard-Stationsabfrage wird noch eine weitere Stationsabfrage der Gruppe 1 alle 100s konfiguriert. Außerdem wird zu der Standard-Zählerabfrage eine Zählerabfrage (Umspeichern) der Zählergruppe 1 alle 200s konfiguriert.

```
PROGRAM P_AcquisitionConfig
VAR_IN_OUT
    acqPara : ST_IEC870_5_101AcquisitionParams;
END_VAR
acqPara.arrInitSeq[0]      := eIEC870_ISTEP_UNUSED;
acqPara.arrInitSeq[1]      := eIEC870_ISTEP_UNUSED;
acqPara.arrInitSeq[2]      := eIEC870_ISTEP_UNUSED;
acqPara.arrInitSeq[3]      := eIEC870_ISTEP_UNUSED;
acqPara.arrInitSeq[4]      := eIEC870_ISTEP_UNUSED;

acqPara.testCmd.bEnable    := FALSE;

acqPara.clockSync.bEnable  := FALSE;

acqPara.arrGenro[1].asduAddr := IEC870_ASUADDR_SYSPARA;
acqPara.arrGenro[1].eQOI    := eIEC870_QOI_INRO1;
acqPara.arrGenro[1].tPollCycle:= T#100s;
acqPara.arrGenro[1].bEnable  := TRUE;

acqPara.arrCoro[1].asduAddr := IEC870_ASUADDR_SYSPARA;
acqPara.arrCoro[1].eFRZ     := eIEC870_FRZ_FREEZE;
acqPara.arrCoro[1].eRQT     := eIEC870_RQT_REQCO1;
acqPara.arrCoro[1].tPollCycle := T#200s;
acqPara.arrCoro[1].bEnable  := TRUE;
```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uutilities (System)

**5.3.16 ST\_IEC870\_5\_101TestPollParams**

Konfigurationsparameter für den zyklischen Testbefehl.

**Syntax**

```
TYPE ST_IEC870_5_101TestPollParams:
STRUCT
    asduAddr      : DWORD := IEC870_ASUADDR_SYSPARA;
    tPollCycle    : TIME := T#60s;
```

```

    bEnable      : BOOL := FALSE;
END_STRUCT
END_TYPE

```

**asduAddr:** Zieladresse des Testbefehls (Gemeinsame ASDU-Adresse);

**tPollCycle:** Zykluszeit des Testbefehls.

**bEnable:** Aktiviert (TRUE) oder deaktiviert (FALSE) die zyklischen Testbefehle.

#### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.17 ST\_IEC870\_5\_101ClockPollParams

Konfigurationsparameter für den zyklischen Uhrzeitsynchronisationsbefehl. Die Station, welche die Uhrzeitsynchronisationsbefehle sendet, besitzt eine eigene interne Software-Uhr. Diese Uhr wird während der Stationsinitialisierung mit der lokalen Windows-Systemzeit (der Uhrzeit in der Windows Taskleiste) synchronisiert. Im Betrieb werden die Zielstationen über die Uhrzeitsynchronisationsbefehle mit der Uhrzeit der internen Software-Uhr synchronisiert.

#### Syntax

```

TYPE ST_IEC870_5_101ClockPollParams :
STRUCT
    asduAddr      : DWORD := IEC870_ASDUADDR_SYSPARA;
    tPollCycle    : TIME := T#60s;
    bEnable       : BOOL := FALSE;
END_STRUCT
END_TYPE

```

**asduAddr:** Zieladresse des Uhrzeitsynchronisationsbefehls (Gemeinsame ASDU adresse).

**tPollCycle:** Zykluszeit des Uhrzeitsynchronisationsbefehls.

**bEnable:** Aktiviert/deaktiviert die zyklischen Uhrzeitsynchronisationsbefehle.

#### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.18 ST\_IEC870\_5\_101GenroPollParams

Konfigurationsparameter für den zyklischen Stationsabfragebefehl.

**Syntax**

```
TYPE ST_IEC870_5_101GenroPollParams:
STRUCT
  asduAddr      : DWORD := IEC870_ASDUADDR_SYSPARA;
  tPollCycle    : TIME := T#60s;
  eQOI          : E_IEC870_5_101QOI := eIEC870_QOI_INROGEN;
  bEnable       : BOOL := FALSE;
END_STRUCT
END_TYPE
```

**asduAddr:** Zieladresse des Stationsabfragebefehls (CAA).

**tPollCycle:** Stationsabfrage-Zykluszeit.

**eQOI:** Abfragebefehlparameter/Kennung [► 521].

**bEnable:** Aktiviert (TRUE) oder deaktiviert (FALSE) die Stationsabfragebefehle.

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.19 ST\_IEC870\_5\_101CoroPollParams**

Konfigurationsparameter für den zyklischen Zählerabfragebefehl.

**Syntax**

```
TYPE ST_IEC870_5_101CoroPollParams:
STRUCT
  asduAddr      : DWORD := IEC870_ASDUADDR_SYSPARA;
  tPollCycle    : TIME := T#60s;
  eRQT          : E_IEC870_5_101RQT := eIEC870_RQT_REQCOGEN;
  eFRZ          : E_IEC870_5_101FRZ := eIEC870_FRZ_READ;
  bEnable       : BOOL := FALSE;
END_STRUCT
END_TYPE
```

**asduAddr:** Zieladresse des Zählerabfragebefehls (Gemeinsame ASDU Adresse).

**tPollCycle:** Zykluszeit des Zählerabfragebefehls.

**eRQT:** Kennung für die Zählerabfrage [► 523].

**eFRZ:** FREEZE-/RESET-Kennung [► 522].

**bEnable:** Aktiviert/deaktiviert die zyklischen Zählerabfragebefehle.

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.20 ST\_IEC870\_5\_101GenCmdPollParams

Parameter für zyklische Übertragung der Datenpunkte in Steuerungsrichtung (Einzelbefehl, Doppelbefehl, Sollwerte usw.).

Die (High-Level) Zentralstation-Funktionsbausteine FB\_IEC870\_5\_101Master und FB\_IEC870\_5\_104Master implementieren eine einfache/automatische Befehlsausführung. In deren Implementierung wird ein Befehl aktiviert/ausgeführt, sobald sich der Prozesswert eines Kontrolldatenpunkts geändert hat (Beispiel der Wert von Einzelbefehl (C\_SC\_NA\_1) ändert sich von OFF zu ON). In einigen Fällen kann es aber notwendig sein, die Werte aller Kontrolldatenpunkte, besonders nach dem Verlust und der Wiederherstellung der Kommunikationsverbindung, erneut zur Unterstation zu senden.

In diesem Fall kann die zyklische Übertragung der Datenpunkte vom Nutzen sein. Bitte beachten Sie, dass es einen konkurrierenden Zugriff auf die Befehlsausführung geben kann, wenn z. B. die zyklische Übertragung der Datenpunkte in Steuerungsrichtung (Befehlsausführung) ansteht und ein Kontrolldatenpunkt bereits einen Befehl ausführt (durch spontane Wertänderung am Datenpunkt in Steuerungsrichtung). Der aktive Befehl muss zuerst zu Ende ausgeführt werden. Erst danach kann ein neuer Befehl aktiviert werden. Da an einer Befehlsausführung mehrere Primitive (Aktivierung, Bestätigung der Aktivierung und Terminierung) beteiligt sind, kann je nach System eine Befehlsausführungszeit von einigen Millisekunden bis zu Sekunden/Minuten auftreten. Dies sollte bei der Implementierung einer Zentralstation-Applikation berücksichtigt werden. Bei sich schnell ändernden SPS-Sollwerten sollte dann eine Filterung vorgenommen werden. Z. B. bei Analoogsollwerten können die niederwertigsten Bits ausmaskiert werden. Auf diese Weise kann die Anzahl der Befehlsaktivierungen reduziert werden.

Bei einem konkurrierenden Zugriff auf die Befehlsausführung können Befehle zum späteren Zeitpunkt (wiederholt) aktiviert werden. Der wiederholte Befehl wird aber mit dem zum späteren Zeitpunkt vorliegenden Sollwert aktiviert.

#### Syntax

```
TYPE ST_IEC870_5_101GenCmdPollParams:
STRUCT
  tPollCycle          : TIME := T#1h;
  bEnable             : BOOL := FALSE;
  options             : DWORD := 0;
  bQueueConcurrentCmds : BOOL := FALSE;
END_STRUCT
END_TYPE
```

**tPollCycle:** Übertragungszykluszeit.

**bEnable:** Aktiviert/deaktiviert die zyklische Übertragung der Datenpunkte.

**options:** Zusätzliche Parameter.

**bQueueConcurrentCmds:** Kontrolliert das Verhalten der Befehlsausführung, wenn zum Zeitpunkt der zyklischen Befehlsaktivierung bereits ein Befehl ausgeführt wird (ausgelöst z .B. durch eine spontane Wertänderung in den SPS-Prozessdaten):

- FALSE (Defaultwert). Bei Datenpunkten, die gerade einen Befehl ausführen, werden keine wiederholten Befehle zum späteren Zeitpunkt aktiviert.
- TRUE: Bei Datenpunkten, die gerade einen Befehl ausführen, wird der Befehl zum späteren Zeitpunkt aktiviert (nachdem die aktuell aktive Befehlsausführung abgeschlossen wurde).

#### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)



### 5.3.21 ST\_IEC870\_5\_101DelayPollParams

Konfigurationsparameter für die Verzögerung des nächsten Initialisierungsschrittes.

**Syntax**

```
TYPE ST_IEC870_5_101DelayPollParams :
STRUCT
    tDelay : TIME := T#5s;
END_STRUCT
END_TYPE
```

**tDelay:** Verzögerungszeit.

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.22 ST\_IEC870\_5\_101HashTableKey

Applikationsobjekt-Datenbank-Lookup-Schlüssel. Mit Hilfe des Schlüssels können die Hash-Tabelleneinträge lokalisiert und modifiziert werden.

**Syntax**

```
TYPE ST_IEC870_5_101HashTableKey :
STRUCT
    eType      : E_IEC870_5_101TcTypeID := ASDU_TYPEUNDEF;
    asduAddr   : DWORD := 0;
    objAddr    : DWORD := 0;
    group      : DWORD := 0;
    lookup     : DWORD := IEC870_LOOKUP_KEY_ALL_ON;
END_STRUCT
END_TYPE
```

**eType:** IEC 60870-5-101/104 ASDU-Typkennung [▶ 500] (z.B.: M\_SP\_NA\_1 für Single-Point oder M\_DP\_NA\_1 für Double-Point usw.).

**asduAddr:** Gemeinsame ASDU-Adresse (CAA).

**objAddr:** Informationsobjektadresse (IOA).

**group:** Object-Group-Konfigurationsflags [▶ 554]. Die Flags können mit OR-Verknüpfung kombiniert werden. Wenn dieser Parameter den Wert Null hat werden die group-Parameter ignoriert.

**lookup:** Zusätzliche Lookup-Schlüsselparameter. Die verfügbaren Parameter sind als Konstanten deklariert (siehe untere Tabelle). Diese können mit OR-Verknüpfung kombiniert werden.

Konstante	Wert	Beschreibung
IEC870_LOOKUP_KEY_ALL_ON	0	Bei der Suche werden alle Parameter berücksichtigt ( <i>eType</i> , <i>asduAddr</i> , <i>objAddr</i> , <i>group</i> ).
IEC870_LOOKUP_KEY_TYPE_OF F	1	Bei der Suche wird der <i>eType</i> -Parameter ignoriert.
IEC870_LOOKUP_KEY_ASDUADDR_OFF	2	Bei der Suche wird der <i>asduAddr</i> -Parameter ignoriert.
IEC870_LOOKUP_KEY_GROUP_OFF	4	Bei der Suche wird der <i>group</i> -Parameter ignoriert.
IEC870_LOOKUP_KEY_OBJADDR_OFF	8	Bei der Suche wird der <i>objAddr</i> -Parameter ignoriert. Dieser Parameter ist nicht zu empfehlen da alle Datenpunkte über eine eindeutige Objektadresse identifizierbar sein müssen.

### Beispiel in ST:

Siehe in der Beschreibung der Funktion: [F\\_iecLookupTableEntry \[► 458\]](#), [F\\_iecRemoveTableEntry \[► 460\]](#), [F\\_iecGetPosOfTableEntry \[► 456\]](#).

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.3.23 E\_IEC870\_5\_101AcquisitionMode

Ab der Tc2\_IEC60870\_5\_10x.compiled-library v3.0.7.0 kann der Trigger/Auslöser der Befehle in der Initialisierungssequenz und/oder der zyklischen Abfragebefehle konfiguriert werden. Die Befehle der Initialisierungssequenz und die zyklischen Abfragebefehle werden nach der Herstellung der Kommunikationsverbindung „STARTDT“ (IEC 60870-5-104) oder „reset of link“ (IEC 60870-5-101) und/oder nach dem Empfang der ASDU: M\_EI\_NA\_1 (Ende der Initialisierung) ausgeführt. Die detaillierte Konfiguration der zyklischen Datenerfassung erfolgt über die Komponenten des strukturierten Typs: [ST\\_IEC870\\_5\\_101AcquisitionParams \[► 491\]](#).

Zur Initialisierungssequenz gehören Befehle, die in der [ST\\_IEC870\\_5\\_101AcquisitionParams.arrInitSeq \[► 492\]](#)-Komponente konfiguriert wurden.

Bei den zyklischen Abfragebefehlen handelt es sich um Befehle, die über folgende Strukturkomponenten konfiguriert wurden:

- [ST\\_IEC870\\_5\\_101AcquisitionParams.testCmd \[► 492\]](#)
- [ST\\_IEC870\\_5\\_101AcquisitionParams.clockSync \[► 492\]](#)
- [ST\\_IEC870\\_5\\_101AcquisitionParams.arrGenro \[► 492\]](#)
- [ST\\_IEC870\\_5\\_101AcquisitionParams.arrCoro \[► 493\]](#)
- [ST\\_IEC870\\_5\\_101AcquisitionParams.genCmd \[► 493\]](#)

**Syntax**

```

TYPE E_IEC870_5_101AcquisitionMode :
(
  CyclicPoll := 0,
  InitSeq_CyclicPoll := 1,
  CyclicPoll_OnEndOfInit_InitSeq_CyclicPoll := 2,
  InitSeq_CyclicPoll_OnEndOfInit_InitSeq_CyclicPoll := 3,
  OnEndOfInit_InitSeq_CyclicPoll := 4
);
END_TYPE
    
```

**Werte**

Name	Beschreibung
CyclicPoll	Nach dem Empfang von STARTDT/ResetOfLink wird die Abarbeitung der zyklischen Abfragebefehle gestartet und kontinuierlich fortgesetzt. Befehle in der Initialisierungssequenz (auch wenn konfiguriert) werden nicht ausgeführt. Die ASDU: M_EI_NA_1 wird nicht nicht ausgewertet und ignoriert.
InitSeq_CyclicPoll	Nach dem Empfang von STARTDT/ResetOfLink wird zuerst die Abarbeitung der Befehle in der Initialisierungssequenz gestartet. Wenn alle Befehle in der Initialisierungssequenz ausgeführt wurden, wird die Abarbeitung der zyklischen Abfragebefehle gestartet und kontinuierlich fortgesetzt. Die ASDU: M_EI_NA_1 wird nicht ausgewertet und ignoriert.
CyclicPoll_OnEndOfInit_InitSeq_CyclicPoll	Nach dem Empfang von STARTDT/ResetOfLink wird die Abarbeitung der zyklischen Abfragebefehle gestartet. Die zyklischen Abfragebefehle werden aber nur bis zum Empfang von ASDU: M_EI_NA_1 abgearbeitet. Beim Empfang von ASDU: M_EI_NA_1 wird die Abarbeitung der zyklischen Abfragebefehle unterbrochen (der zuletzt aktive Befehl wird aber noch komplettiert) und die Abarbeitung der Befehle in der Initialisierungssequenz gestartet. Nach der Abarbeitung aller Befehle in der Initialisierungssequenz wird erneut die Abarbeitung der zyklischen Abfragebefehle gestartet und kontinuierlich fortgesetzt.
InitSeq_CyclicPoll_OnEndOfInit_InitSeq_CyclicPoll	Nach dem Empfang von STARTDT/ResetOfLink und bis zum Empfang von ASDU: M_EI_NA_1 wird die Abarbeitung der Befehle in der Initialisierungssequenz gestartet. Danach (wenn alle vor dem Empfang von M_EI_NA_1 ausgeführt werden konnten) wird die Abarbeitung der zyklischen Abfragebefehle gestartet. Beim Empfang von ASDU: M_EI_NA_1 wird die Abarbeitung der Befehle in der Initialisierungssequenz oder der zyklischen Abfragebefehle unterbrochen (der zuletzt aktive Befehl wird aber noch komplettiert) und die Abarbeitung der Befehle in der Initialisierungssequenz erneut gestartet. Nach der Abarbeitung aller Befehle in der Initialisierungssequenz wird erneut die Abarbeitung der zyklischen Abfragebefehle gestartet und kontinuierlich fortgesetzt.
OnEndOfInit_InitSeq_CyclicPoll	Es wird auf den Empfang der ASDU: M_EI_NA_1 gewartet. Beim Empfang von M_EI_NA_1 wird zuerst die Abarbeitung der Befehle in der Initialisierungssequenz gestartet. Nach der Abarbeitung aller Befehle in der Initialisierungssequenz wird die Abarbeitung der zyklischen Abfragebefehle gestartet und kontinuierlich fortgesetzt.

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4022.28 und höher	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x v3.0.7.0 und höher (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uutilities (System)

## 5.3.24 E\_IEC870\_5\_101TcTypeID

Datenpunkt-Typbezeichner nach IEC 60870-5-101/104.

### Syntax

```

TYPE E_IEC870_5_101TcTypeID :
(
  ASDU_TYPEUNDEF := 0, (* (000, 0x00) not allowed *)
  (* reserved standard asdu types *)
  M_SP_NA_1, (* (001, 0x01) Single-point information *)
  M_SP_TA_1, (* (002, 0x02) Single-point information with time tag *)
  M_DP_NA_1, (* (003, 0x03) Double-point information *)
  M_DP_TA_1, (* (004, 0x04) Double-point information with time tag *)
  M_ST_NA_1, (* (005, 0x05) Step position information *)
  M_ST_TA_1, (* (006, 0x06) Step position information with time tag *)
  M_BO_NA_1, (* (007, 0x07) Bitstring of 32 bit *)
  M_BO_TA_1, (* (008, 0x08) Bitstring of 32 bit with time tag *)
  M_ME_NA_1, (* (009, 0x09) Measured value, normalised value *)
  M_ME_TA_1, (* (010, 0x0A) Measured value, normalized value with time tag *)
  M_ME_NB_1, (* (011, 0x0B) Measured value, scaled value *)
  M_ME_TB_1, (* (012, 0x0C) Measured value, scaled value wit time tag *)
  M_ME_NC_1, (* (013, 0x0D) Measured value, short floating point number *)
  M_ME_TC_1, (* (014, 0x0E) Measured value, short floating point number with time tag *)
  M_IT_NA_1, (* (015, 0x0F) Integrated totals *)
  M_IT_TA_1, (* (016, 0x10) Integrated totals with time tag *)
  M_EP_TA_1, (* (017, 0x11) Event of protection equipment with time tag *)
  M_EP_TB_1, (* (018, 0x12) Packed start events of protection equipment with time tag *)
  M_EP_TC_1, (* (019, 0x13) Packed output circuit information of protection equipment with time ta
g *)
  M_PS_NA_1, (* (020, 0x14) Packed single point information with status change detection *)
  M_ME_ND_1, (* (021, 0x15) Measured value, normalized value without quality descriptor *)
  ASDU_TYPE_22,
  ASDU_TYPE_23,
  ASDU_TYPE_24,
  ASDU_TYPE_25,
  ASDU_TYPE_26,
  ASDU_TYPE_27,
  ASDU_TYPE_28,
  ASDU_TYPE_29,
  M_SP_TB_1, (* (030, 0x1E) Single-point information with time tag CP56Time2a *)
  M_DP_TB_1, (* (031, 0x1F) Double-point information with time tag CP56Time2a *)
  M_ST_TB_1, (* (032, 0x20) Step position information with time tag CP56Time2a *)
  M_BO_TB_1, (* (033, 0x21) Bitstring of 32 bit with time tag CP56Time2a *)
  M_ME_TD_1, (* (034, 0x22) Measured value, normalised value with time tag CP56Time2a *)
  M_ME_TE_1, (* (035, 0x23) Measured value, scaled value with time tag CP56Time2a *)
  M_ME_TF_1, (* (036, 0x24) Measured value, short floating point number with time tag CP56Time2a *
)
  M_IT_TB_1, (* (037, 0x25) Integrated totals with time tag CP56Time2a *)
  M_EP_TD_1, (* (038, 0x26) Event of protection equipment with time tag CP56Time2a *)
  M_EP_TE_1, (* (039, 0x27) Packed start events of protection equipment with time tag CP56Time2a *
)
  M_EP_TF_1, (* (040, 0x28) Packed output circuit information of protection equipment with time ta
g CP56Time2a *)
  ASDU_TYPE_41,
  ASDU_TYPE_42,
  ASDU_TYPE_43,
  ASDU_TYPE_44,
  C_SC_NA_1, (* (045, 0x2D) Single command *)
  C_DC_NA_1, (* (046, 0x2E) Double command *)
  C_RC_NA_1, (* (047, 0x2F) Regulating step command *)
  C_SE_NA_1, (* (048, 0x30) Set-point Command, normalised value *)
  C_SE_NB_1, (* (049, 0x31) Set-point Command, scaled value *)
  C_SE_NC_1, (* (050, 0x32) Set-point Command, short floating point number *)
  C_BO_NA_1, (* (051, 0x33) Bitstring 32 bit command *)
  ASDU_TYPE_52,
  ASDU_TYPE_53,
  ASDU_TYPE_54,
  ASDU_TYPE_55,
  ASDU_TYPE_56,
  ASDU_TYPE_57,
  C_SC_TA_1, (* (058, 0x3A) Single command with time tag CP56Time2a *)
  C_DC_TA_1, (* (059, 0x3B) Double command with time tag CP56Time2a *)
  C_RC_TA_1, (* (060, 0x3C) Regulating step command with time tag CP56Time2a *)
  C_SE_TA_1, (* (061, 0x3D) Measured value, normalised value command with time tag CP56Time2a *)
  C_SE_TB_1, (* (062, 0x3E) Measured value, scaled value command with time tag CP56Time2a *)
  C_SE_TC_1, (* (063, 0x3F) Measured value, short floating point number command with time tag CP56
Time2a *)
  C_BO_TA_1, (* (064, 0x40) Bitstring of 32 bit command with time tag CP56Time2a *)
)

```

```

ASDU_TYPE_65,
ASDU_TYPE_66,
ASDU_TYPE_67,
ASDU_TYPE_68,
ASDU_TYPE_69,
M_EI_NA_1, (* (070, 0x46) End of Initialisation *)
ASDU_TYPE_71,
ASDU_TYPE_72,
ASDU_TYPE_73,
ASDU_TYPE_74,
ASDU_TYPE_75,
ASDU_TYPE_76,
ASDU_TYPE_77,
ASDU_TYPE_78,
ASDU_TYPE_79,
ASDU_TYPE_80,
ASDU_TYPE_81,
ASDU_TYPE_82,
ASDU_TYPE_83,
ASDU_TYPE_84,
ASDU_TYPE_85,
ASDU_TYPE_86,
ASDU_TYPE_87,
ASDU_TYPE_88,
ASDU_TYPE_89,
ASDU_TYPE_90,
ASDU_TYPE_91,
ASDU_TYPE_92,
ASDU_TYPE_93,
ASDU_TYPE_94,
ASDU_TYPE_95,
ASDU_TYPE_96,
ASDU_TYPE_97,
ASDU_TYPE_98,
ASDU_TYPE_99,
C_IC_NA_1, (* (100, 0x64) Interrogation command *)
C_CI_NA_1, (* (101, 0x65) Counter interrogation command *)
C_RD_NA_1, (* (102, 0x66) Read Command*)
C_CS_NA_1, (* (103, 0x67) Clock synchronisation command *)
C_TS_NA_1, (* (104, 0x68) Test command *)
C_RP_NA_1, (* (105, 0x69) Reset process command *)
C_CD_NA_1, (* (106, 0x6A) C_CD_NA_1 Delay acquisition command *)
C_TS_TA_1, (* (107, 0x6B) Test command with time tag CP56Time2a *)
ASDU_TYPE_108,
ASDU_TYPE_109,
P_ME_NA_1, (* (110, 0x6E) Parameter of measured values, normalized value *)
P_ME_NB_1, (* (111, 0x6F) Parameter of measured values, scaled value *)
P_ME_NC_1, (* (112, 0x70) Parameter of measured values, short floating point number *)
P_AC_NA_1, (* (113, 0x71) Parameter activation *)
ASDU_TYPE_114,
ASDU_TYPE_115,
ASDU_TYPE_116,
ASDU_TYPE_117,
ASDU_TYPE_118,
ASDU_TYPE_119,
F_FR_NA_1, (* (120, 0x78) File ready *)
F_SR_NA_1, (* (121, 0x79) Section ready *)
F_SC_NA_1, (* (122, 0x7A) Call directory, select file, call file, call section *)
F_LS_NA_1, (* (123, 0x7B) Last section, last segment *)
F_FA_NA_1, (* (124, 0x7C) ACK file, ACK section *)
F_SG_NA_1, (* (125, 0x7D) Segment *)
F_DR_TA_1, (* (126, 0x7E) Directory *)
ASDU_TYPE_127,
(* reserved user asdu types *)
ASDU_TYPE_128,
ASDU_TYPE_129,
ASDU_TYPE_130,
ASDU_TYPE_131,
ASDU_TYPE_132,
ASDU_TYPE_133,
ASDU_TYPE_134,
ASDU_TYPE_135,
ASDU_TYPE_136,
ASDU_TYPE_137,
ASDU_TYPE_138,
ASDU_TYPE_139,
ASDU_TYPE_140,
ASDU_TYPE_141,
ASDU_TYPE_142,
ASDU_TYPE_143,

```

```
ASDU_TYPE_144,  
ASDU_TYPE_145,  
ASDU_TYPE_146,  
ASDU_TYPE_147,  
ASDU_TYPE_148,  
ASDU_TYPE_149,  
ASDU_TYPE_150,  
ASDU_TYPE_151,  
ASDU_TYPE_152,  
ASDU_TYPE_153,  
ASDU_TYPE_154,  
ASDU_TYPE_155,  
ASDU_TYPE_156,  
ASDU_TYPE_157,  
ASDU_TYPE_158,  
ASDU_TYPE_159,  
ASDU_TYPE_160,  
ASDU_TYPE_161,  
ASDU_TYPE_162,  
ASDU_TYPE_163,  
ASDU_TYPE_164,  
ASDU_TYPE_165,  
ASDU_TYPE_166,  
ASDU_TYPE_167,  
ASDU_TYPE_168,  
ASDU_TYPE_169,  
ASDU_TYPE_170,  
ASDU_TYPE_171,  
ASDU_TYPE_172,  
ASDU_TYPE_173,  
ASDU_TYPE_174,  
ASDU_TYPE_175,  
ASDU_TYPE_176,  
ASDU_TYPE_177,  
ASDU_TYPE_178,  
ASDU_TYPE_179,  
ASDU_TYPE_180,  
ASDU_TYPE_181,  
ASDU_TYPE_182,  
ASDU_TYPE_183,  
ASDU_TYPE_184,  
ASDU_TYPE_185,  
ASDU_TYPE_186,  
ASDU_TYPE_187,  
ASDU_TYPE_188,  
ASDU_TYPE_189,  
ASDU_TYPE_190,  
ASDU_TYPE_191,  
ASDU_TYPE_192,  
ASDU_TYPE_193,  
ASDU_TYPE_194,  
ASDU_TYPE_195,  
ASDU_TYPE_196,  
ASDU_TYPE_197,  
ASDU_TYPE_198,  
ASDU_TYPE_199,  
ASDU_TYPE_200,  
ASDU_TYPE_201,  
ASDU_TYPE_202,  
ASDU_TYPE_203,  
ASDU_TYPE_204,  
ASDU_TYPE_205,  
ASDU_TYPE_206,  
ASDU_TYPE_207,  
ASDU_TYPE_208,  
ASDU_TYPE_209,  
ASDU_TYPE_210,  
ASDU_TYPE_211,  
ASDU_TYPE_212,  
ASDU_TYPE_213,  
ASDU_TYPE_214,  
ASDU_TYPE_215,  
ASDU_TYPE_216,  
ASDU_TYPE_217,  
ASDU_TYPE_218,  
ASDU_TYPE_219,  
ASDU_TYPE_220,  
ASDU_TYPE_221,  
ASDU_TYPE_222,  
ASDU_TYPE_223,
```

```

ASDU_TYPE_224,
ASDU_TYPE_225,
ASDU_TYPE_226,
ASDU_TYPE_227,
ASDU_TYPE_228,
ASDU_TYPE_229,
ASDU_TYPE_230,
ASDU_TYPE_231,
ASDU_TYPE_232,
ASDU_TYPE_233,
ASDU_TYPE_234,
ASDU_TYPE_235,
ASDU_TYPE_236,
ASDU_TYPE_237,
ASDU_TYPE_238,
ASDU_TYPE_239,
ASDU_TYPE_240,
ASDU_TYPE_241,
ASDU_TYPE_242,
ASDU_TYPE_243,
ASDU_TYPE_244,
ASDU_TYPE_245,
ASDU_TYPE_246,
ASDU_TYPE_247,
ASDU_TYPE_248,
ASDU_TYPE_249,
ASDU_TYPE_250,
ASDU_TYPE_251,
ASDU_TYPE_252,
ASDU_TYPE_253,
ASDU_TYPE_254,
ASDU_TYPE_255,
ASDU_TYPEMAX (* not used *)
);
END_TYPE

```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.25 E\_IEC870\_5\_102TypeID**

ASDU-Typkennungen nach IEC 60870-5-102.

**Syntax**

```

TYPE E_IEC870_5_102TypeID:
(
ASDU_TYPEUNDEF_2 := 0, (* (000) not allowed *) (* Process information in monitor direction *)
M_SP_TA_2 := 1, (* (001) Single-point information with time tag *)
M_IT_TA_2 := 2, (* (002) Accounting integrated totals, 4 octets each *)
M_IT_TB_2 := 3, (* (003) Accounting integrated totals, 3 octets each *)
M_IT_TC_2 := 4, (* (004) Accounting integrated totals, 2 octets each *)
M_IT_TD_2 := 5, (* (005) Periodical reset accounting integrated totals, 4 octets each *)
M_IT_TE_2 := 6, (* (006) Periodical reset accounting integrated totals, 3 octets each *)
M_IT_TF_2 := 7, (* (007) Periodical reset accounting integrated totals, 2 octets each *)
M_IT_TG_2 := 8, (* (008) Operational integrated totals, 4 octets each *)
M_IT_TH_2 := 9, (* (009) Operational integrated totals, 3 octets each *)
M_IT_TI_2 := 10, (* (010) Operational integrated totals, 2 octets each *)
M_IT_TK_2 := 11, (* (011) Periodical reset operational integrated totals, 4 octets each *)
M_IT_TL_2 := 12, (* (012) Periodical reset operational integrated totals, 3 octets each *)
M_IT_TM_2 := 13, (* (013) Periodical reset operational integrated totals, 2 octets each *)
(* System information in monitor direction *)
M_EI_NA_2 := 70, (* (070) End of initialization *)
P_MP_NA_2 := 71, (* (071) Manufacturer and product specification of integrated total DTE *)
M_TI_TA_2 := 72, (* (072) Current system time of integrated total DTE *)

```

```

(* System information in control direction *)
C_RD_NA_2 := 100, (* (100) Read manufacturer and product specification *)
C_SP_NA_2 := 101, (* (101) Read record of single-point information with time tag *)
C_SP_NB_2 := 102, (* (102) Read record of single-
point information with time tag of a selected time range *)
C_TI_NA_2 := 103, (* (103) Read current system time of integrated total DTE *)
C_CI_NA_2 := 104, (* (104) Read accounting integrated totals of the oldest integration period *)
C_CI_NB_2 := 105, (* (105) Read accounting integrated totals of the oldest integration period and
of a selected range of addresses *)
C_CI_NC_2 := 106, (* (106) Read accounting integrated totals of a specific past integration peri
od *)
C_CI_ND_2 := 107, (* (107) Read accounting integrated totals of a specific past integration peri
od and of a selected range of addresses *)
C_CI_NE_2 := 108, (* (108) Read periodical reset accounting integrated totals of the oldest inte
gration period *)
C_CI_NF_2 := 109, (* (109) Read periodical reset accounting integrated totals of the oldest inte
gration period and of a selected range of addresses *)
C_CI_NG_2 := 110, (* (110) Read periodical reset accounting integrated totals of a specific past
integration period *)
C_CI_NH_2 := 111, (* (111) Read periodical reset accounting integrated totals of a specific past
integration period and of a selected range of addresses *)
C_CI_NI_2 := 112, (* (112) Read operational integrated totals of the oldest integration period *)
)
C_CI_NK_2 := 113, (* (113) Read operational integrated totals of the oldest integration period a
nd of a selected range of addresses *)
C_CI_NL_2 := 114, (* (114) Read operational integrated totals of a specific past integration peri
od *)
C_CI_NM_2 := 115, (* (115) Read operational integrated totals of a specific past integration peri
od and of a selected range of addresses *)
C_CI_NN_2 := 116, (* (116) Read periodical reset operational integrated totals of the oldest int
egration period *)
C_CI_NO_2 := 117, (* (117) Read periodical reset operational integrated totals of the oldest int
egration period and of a selected range of addresses *)
C_CI_NP_2 := 118, (* (118) Read periodical reset operational integrated totals of a specific pas
t integration period *)
C_CI_NQ_2 := 119, (* (119) Read periodical reset operational integrated totals of a specific pas
t integration period and of a selected range of addresses *)
C_CI_NR_2 := 120, (* (120) Read accounting integrated totals of a specific past integration peri
od of a selected time range and of a selected range of addresses *)
C_CI_NS_2 := 121, (* (121) Read periodical reset accounting integrated totals of a specific past
integration period of a selected time range and of a selected range of addresses *)
C_CI_NT_2 := 122, (* (122) Read operational integrated totals of a specific past integration peri
od of a selected time range and of a selected range of addresses *)
C_CI_NU_2 := 123, (* (123) Read periodical reset operational integrated totals of a specific pas
t integration period of a selected time range and of a selected range of addresses *)

M_DS_TA_2 := 128, (* (128) *)
P_ME_NA_2 := 129, (* Parameters of the measuring point *)
M_DS_TB_2 := 130, (**)
M_CH_TA_2 := 131, (**)
C_PK_2 := 132, (* Load private key *)
C_TA_VC_2 := 133, (* Read tariff information ( current values ) *)
C_TA_VM_2 := 134, (* Read tariff information ( stored values ) *)
M_TA_VC_2 := 135, (* Tariff information ( current values ) *)
M_TA_VM_2 := 136, (* Tariff information ( stored values ) *)
C_TA_CP_2 := 137, (* Close accounting period *)
M_IB_TG_2 := 139, (* Block of operational integrated totals ( absolute values ) *)
M_IB_TK_2 := 140, (* Block of periodical reset operational integrated totals ( increment values
) *)
C_RM_NA_2 := 141, (* Read configuration data of the meter device *)
M_RM_NA_2 := 142, (* Configuration of the meter device *)
C_MR_NA_2 := 143, (* Change configuration data of the meter device *)
C_PC_NA_2 := 144, (* (144) *)
M_PC_NA_2 := 145, (* (145) *)
C_MC_NA_2 := 146, (* (146) *)
C_DF_NA_2 := 147, (* (147) *)
M_DF_NA_2 := 148, (* (148) *)
C_MF_NA_2 := 149, (* (149) *)

C_DS_TA_2 := 180, (* (180) *)
C_CS_TA_2 := 181, (* (181) Change date and time ( Time synchronization ) *)
C_PI_NA_2 := 182, (* (182) Read parameters of the measuring point *)
C_AC_NA_2 := 183, (* (183) Start session and send access key *)

C_DS_TB_2 := 184, (* (184) *)
C_CH_TA_2 := 185, (* (185) *)
C_MH_TA_2 := 186, (* (186) *)
C_FS_NA_2 := 187, (* (187) Finish session *)
C_MP_NA_2 := 188, (* (188) *)
C_CB_NT_2 := 189, (* (189) Read a block of operational integrated totals of a time period and a

```



```
selected address *)
  C_CB_UN_2 := 190 (* (190) Read a block of periodical reset operational integrated totals of a time period and a selected address *)
);
END_TYPE
```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.26 E\_IEC870\_5\_103MTypeID**

ASDU-Typkennungen in Überwachungsrichtung (slave -> master) nach IEC 60870-5-103.

**Syntax**

```
TYPE E_IEC870_5_103MTypeID:
(
  M_TYPEUNDEF_3 := 0, (* Not used *)
  M_TTM_TA_3 := 1, (* Time-tagged message *)
  M_TMR_TA_3 := 2, (* Time-tagged message with relative time *)
  M_MEI_NA_3 := 3, (* Measurands I *)
  M_TME_TA_3 := 4, (* Time-tagged measurands with relative time *)
  M_IRC_NA_3 := 5, (* Identification *)
  M_SYN_TA_3 := 6, (* Time synchronisation *)
  M_TGI_NA_3 := 8, (* General interrogation *)
  M_MEII_NA_3 := 9, (* Measurands II *)
  M_GD_XA_3 := 10, (* Generic data *)
  M_GI_XA_3 := 11, (* Generic identification *)
  M_LRD_TA_3 := 23, (* List of recorded disturbances *)
  M_RTD_TA_3 := 26, (* Ready for transmission of disturbance data *)
  M_RTC_NA_3 := 27, (* Ready for transmission of channel *)
  M_RTT_NA_3 := 28, (* Ready for transmission of tags *)
  M_TOT_NA_3 := 29, (* Transmission of tags *)
  M_TOV_NA_3 := 30, (* Transmission of disturbance values *)
  M_EOT_NA_3 := 31 (* End of transmission *)
);
END_TYPE
```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.27 E\_IEC870\_5\_103CTypeID**

ASDU-Typkennungen in Steuerungsrichtung (master -> slave) nach IEC 60870-5-103.

**Syntax**

```
TYPE E_IEC870_5_103CTypeID:
(
  C_TYPEUNDEF_3 := 0, (* Not used *)
```

```

C_SYN_TA_3 := 6, (* Time synchronisation *)
C_IGI_NA_3 := 7, (* General interrogation *)
C_GD_NA_3 := 10, (* Generic data *)
C_GRC_NA_3 := 20, (* General command *)
C_GC_NA_3 := 21, (* Generic command *)
C_ODT_NA_3 := 24, (* Order for disturbance data transmission *)
C_ADT_NA_3 := 25 (* Acknowledgement for disturbance data transmission *)
);
END_TYPE

```

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.3.28 E\_IEC870\_5\_101IOMappingType

TwinCAT SPS-Prozessdatenbereich (inputs, outputs, memory, data) in den bzw. aus dem die IEC-Prozessdaten gemappt (kopiert) werden sollen.

### Syntax

```

TYPE E_IEC870_5_101IOMappingType :
(
  MAP_AREA_NONE := 0,
  MAP_AREA_MEMORY := 1,
  MAP_AREA_INPUT := 2,
  MAP_AREA_OUTPUT := 4,
  MAP_AREA_DATA := 8
);
END_TYPE

```

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.3.29 E\_IEC870\_5\_101AsduAddrSize

Oktetgrösse (Bytegrösse) der gemeinsamen ASDU-Adresse.

### Syntax

```

TYPE E_IEC870_5_101AsduAddrSize :
(
  eIEC870_AsdAddr_OneOctet := 1, (* Ein Byte *)
  eIEC870_AsdAddr_TwoOctets := 2 (* Zwei Byte *)
);
END_TYPE

```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.30 E\_IEC870\_5\_101COTSize**

Oktetgröße (Bytegröße) der Übertragungsursache.

**Syntax**

```
TYPE E_IEC870_5_101COTSize :
(
  eIEC870_COT_OneOctet := 1, (* Ein Byte *)
  eIEC870_COT_TwoOctets := 2 (* Zwei Byte (inklusive der Quelladresse) *)
);
END_TYPE
```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.31 E\_IEC870\_5\_101LinkAddrSize**

Oktetgröße (Bytegröße) der Linkadresse.

**Syntax**

```
TYPE E_IEC870_5_101LinkAddrSize :
(
  eIEC870_LinkAddr_None := 0, (* Keine (wird nicht benutzt) *)
  eIEC870_LinkAddr_OneOctet := 1, (* Ein Byte *)
  eIEC870_LinkAddr_TwoOctets := 2 (* Zwei Byte *)
);
END_TYPE
```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.32 E\_IEC870\_5\_101ObjAddrSize

Oktetgrösse (Bytegrösse) der Informationsobjektadresse (IOA).

#### Syntax

```
TYPE E_IEC870_5_101ObjAddrSize :
(
  eIEC870_ObjAddr_OneOctet      := 1, (* Ein Byte *)
  eIEC870_ObjAddr_TwoOctets     := 2, (* Zwei Byte *)
  eIEC870_ObjAddr_ThreeOctets  := 3 (* Drei Byte *)
);
END_TYPE
```

#### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.33 E\_IEC870\_5\_101ErrorSourceID

Fehlerquelle.

#### Syntax

```
TYPE E_IEC870_5_101ErrorSourceID :
(
  eIEC870_ESRC_NONE, (* Fehlerquelle ist unbekannt *)
  eIEC870_ESRC_COUNTER_INTERROGATION, (* Fehler während der Ausführung der Zählwertabfrage *)
  eIEC870_ESRC_SYNC_CLOCK_CTRL, (* Fehler während der Ausführung der Uhrzeitsynchronisation *)
  eIEC870_ESRC_CLOCK_EVENT, (* Fehler während der spontanen Übertragung der Uhrzeit *)
  eIEC870_ESRC_GETPCTIME, (* Fehler während der Synchronisierung der Geräte-
Uhrzeit mit der Uhrzeit des PC's *)
  eIEC870_ESRC_SETPCTIME, (* Fehler während der Synchronisierung der PC-
Uhrzeit mit der Uhrzeit des Gerätes *)
  eIEC870_ESRC_STATION_INTERROGATION, (* Fehler während der Generalabfrage *)
  eIEC870_ESRC_READ_DATA_CMD, (* Fehler beim Lesekommando *)
  eIEC870_ESRC_RESET_PROCESS, (* Fehler beim Process-Reset-Kommando *)
  eIEC870_ESRC_TEST_CMD, (* Fehler beim Testkommando *)
  eIEC870_ESRC_ENDOFINIT, (* Fehler bei M_EI_NA_1 (Ende der Initialisierung) *)
  eIEC870_ESRC_BACKGROUND_SCAN, (* Fehler bei der Ausführung der Hintergrundabfrage *)
  eIEC870_ESRC_COMMAND_CTRL, (* Fehler bei der Initialisierung des Befehls (single-
command, double-command, setpoint-command ...) *)
  eIEC870_ESRC_COMMAND_EXEC, (* Fehler bei der Ausführung des Befehls *)
  eIEC870_ESRC_LOCAL_FREEZE_RESET, (* Fehler beim Umspeichern / Zurücksetzen *)
  eIEC870_ESRC_PERIODIC_CYCLIC, (* Fehler bei der periodischen/zyklischen Datenübertragung *)
  eIEC870_ESRC_USERAPP_OBJECT, (* Fehler im Applikationsobjekt (fehlerhafte Konfiguration/
Wert) *)
  eIEC870_ESRC_USERAPP_SETQUALITY, (* Fehler bei der Quality-Flag-Bearbeitung *)
  eIEC870_ESRC_IEC60870_5_104LINK, (* Fehler im IEC60870-5-104 Link Layer *)
  eIEC870_ESRC_IEC60870_5_101LINK, (* Fehler im IEC60870-5-101 Link Layer *)
  eIEC870_ESRC_COMLIB, (* Fehler in der unterlagerten Kommunikation über die seriellen
Schnittstellen *)
  eIEC870_ESRC_POLLING_SEVICE (* Fehler in der Acquisition-State-
Machine (Abarbeitung der zyklische Pollanfragen)*)
);
END_TYPE
```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.34 E\_IEC870\_5\_101ClassType**

Prioritätsklasse der ASDU. Hochpriore Daten werden der Klasse 1 zugeordnet und niederpriore Daten der Klasse 2.

**Syntax**

```
TYPE E_IEC870_5_101ClassType :
(
  eIEC870_Class_None,
  eIEC870_Class_1,
  eIEC870_Class_2
);
END_TYPE
```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.35 E\_IEC870\_5\_101COTType**

IEC 60870-5-101/104 Übertragungsursachen (Cause of transfer).

**Syntax**

```
TYPE E_IEC870_5_101COTType:
(
  eIEC870_COT_UNUSED := 0,
  eIEC870_COT_CYCLIC := 1,
  eIEC870_COT_BACKGROUND := 2,
  eIEC870_COT_SPONTAN := 3,
  eIEC870_COT_INIT := 4,
  eIEC870_COT_REQ := 5,
  eIEC870_COT_ACT := 6,
  eIEC870_COT_ACT_CON := 7,
  eIEC870_COT_DEACT := 8,
  eIEC870_COT_DEACT_CON := 9,
  eIEC870_COT_ACT_TERM := 10,
  eIEC870_COT_RETREM := 11,
  eIEC870_COT_RETLOC := 12,
  eIEC870_COT_FILE := 13,
  eIEC870_COT_14 := 14,
  eIEC870_COT_15 := 15,
  eIEC870_COT_16 := 16,
  eIEC870_COT_17 := 17,
  eIEC870_COT_18 := 18,
  eIEC870_COT_19 := 19,
  eIEC870_COT_INROGEN := 20,
```

```
eIEC870_COT_INRO1 := 21,  
eIEC870_COT_INRO2 := 22,  
eIEC870_COT_INRO3 := 23,  
eIEC870_COT_INRO4 := 24,  
eIEC870_COT_INRO5 := 25,  
eIEC870_COT_INRO6 := 26,  
eIEC870_COT_INRO7 := 27,  
eIEC870_COT_INRO8 := 28,  
eIEC870_COT_INRO9 := 29,  
eIEC870_COT_INRO10 := 30,  
eIEC870_COT_INRO11 := 31,  
eIEC870_COT_INRO12 := 32,  
eIEC870_COT_INRO13 := 33,  
eIEC870_COT_INRO14 := 34,  
eIEC870_COT_INRO15 := 35,  
eIEC870_COT_INRO16 := 36,  
eIEC870_COT_REQCOGEN := 37,  
eIEC870_COT_REQCO1 := 38,  
eIEC870_COT_REQCO2 := 39,  
eIEC870_COT_REQCO3 := 40,  
eIEC870_COT_REQCO4 := 41,  
eIEC870_COT_42 := 42,  
eIEC870_COT_43 := 43,  
eIEC870_COT_UNKNOWN_TYPE := 44,  
eIEC870_COT_UNKNOWN_CAUSE := 45,  
eIEC870_COT_UNKNOWN_ASDU_ADDRESS := 46,  
eIEC870_COT_UNKNOWN_OBJECT_ADDRESS := 47,  
eIEC870_COT_48 := 48,  
eIEC870_COT_49 := 49,  
eIEC870_COT_50 := 50,  
eIEC870_COT_51 := 51,  
eIEC870_COT_52 := 52,  
eIEC870_COT_53 := 53,  
eIEC870_COT_54 := 54,  
eIEC870_COT_55 := 55,  
eIEC870_COT_56 := 56,  
eIEC870_COT_57 := 57,  
eIEC870_COT_58 := 58,  
eIEC870_COT_59 := 59,  
eIEC870_COT_60 := 60,  
eIEC870_COT_61 := 61,  
eIEC870_COT_62 := 62,  
eIEC870_COT_63 := 63  
);  
END_TYPE
```

Wert	Beschreibung
eIEC870_COT_UNUSED	Wird nicht benutzt
eIEC870_COT_CYCLIC	Zyklische Daten
eIEC870_COT_BACKGROUND	Hintergrundabfrage
eIEC870_COT_SPONTAN	Spontane Daten
eIEC870_COT_INIT	Ende der Initialisierung
eIEC870_COT_REQ	Read-Request
eIEC870_COT_ACT	Kommando-Aktivierung
eIEC870_COT_ACT_CON	Bestätigung der Kommando-Aktivierung
eIEC870_COT_DEACT	Kommando-Abbruch
eIEC870_COT_DEACT_CON	Bestätigung des Kommando-Abbruchs
eIEC870_COT_ACT_TERM	Terminierung der Kommando-Aktivierung
eIEC870_COT_RETREM	Erwidert wegen einem Fern-Kommando
eIEC870_COT_RETLOC	Erwidert wegen einem Lokal-Kommando
eIEC870_COT_FILE	Dateizugriff
eIEC870_COT_INROGEN	Stationsabfrage (allgemein)
eIEC870_COT_INRO1	Stationsabfrage der Gruppe 1
eIEC870_COT_INRO2	Stationsabfrage der Gruppe 2
eIEC870_COT_INRO3	Stationsabfrage der Gruppe 3
eIEC870_COT_INRO4	Stationsabfrage der Gruppe 4
eIEC870_COT_INRO5	Stationsabfrage der Gruppe 5
eIEC870_COT_INRO6	Stationsabfrage der Gruppe 6
eIEC870_COT_INRO7	Stationsabfrage der Gruppe 7
eIEC870_COT_INRO8	Stationsabfrage der Gruppe 8
eIEC870_COT_INRO9	Stationsabfrage der Gruppe 9
eIEC870_COT_INRO10	Stationsabfrage der Gruppe 10
eIEC870_COT_INRO11	Stationsabfrage der Gruppe 11
eIEC870_COT_INRO12	Stationsabfrage der Gruppe 12
eIEC870_COT_INRO13	Stationsabfrage der Gruppe 13
eIEC870_COT_INRO14	Stationsabfrage der Gruppe 14
eIEC870_COT_INRO15	Stationsabfrage der Gruppe 15
eIEC870_COT_INRO16	Stationsabfrage der Gruppe 16
eIEC870_COT_REQCOGEN	Zählerabfrage (allgemein)
eIEC870_COT_REQCO1	Zählerabfrage der Gruppe 1
eIEC870_COT_REQCO2	Zählerabfrage der Gruppe 2
eIEC870_COT_REQCO3	Zählerabfrage der Gruppe 3
eIEC870_COT_REQCO4	Zählerabfrage der Gruppe 4
eIEC870_COT_UNKNOWN_TYPE	Unbekannter Typ
eIEC870_COT_UNKNOWN_CAUSE	Unbekannte Übertragungsursache
eIEC870_COT_UNKNOWN_ASDU_ADDRESS	Unbekannte gemeinsame Asdu-Adresse
eIEC870_COT_UNKNOWN_OBJECT_ADDRESS	Unbekannte Objekt-Adresse
eIEC870_COT_14..eIEC870_COT_19 eIEC870_COT_42..eIEC870_COT_43 eIEC870_COT_48..eIEC870_COT_63	Bereich, der reserviert/unbenutzt ist.

## Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.3.36 E\_IEC870\_5\_102COTType

Übertragungsursachen (Überwachungsrichtung und/oder Steuerungsrichtung) nach der IEC 60870-5-102.

## Syntax

```

TYPE E_IEC870_5_102COTType:
(
  eIEC870_2COT_0      := 0,
  eIEC870_2COT_1      := 1,
  eIEC870_2COT_2      := 2,
  eIEC870_2COT_SPONTAN := 3, (* spontaneous *)
  eIEC870_2COT_INIT   := 4, (* initialization *)
  eIEC870_2COT_REQ    := 5, (* request or requested *)
  eIEC870_2COT_ACT    := 6, (* command activation *)
  eIEC870_2COT_ACT_CON := 7, (* command activation confirmation *)
  eIEC870_2COT_DEACT  := 8, (* command deactivation *)
  eIEC870_2COT_DEACT_CON := 9, (* command deactivation confirmation *)
  eIEC870_2COT_ACT_TERM := 10, (* command activation termination *)
  eIEC870_2COT_11     := 11,
  eIEC870_2COT_12     := 12,
  eIEC870_2COT_RECORD_NOT_FOUND := 13, (* requested record list is not available *)
  eIEC870_2COT_UNKNOWN_ASDU_TYPE := 14, (* unknown ASDU type identifier *)
  eIEC870_2COT_UNKNOWN_RECORD_NUMBER := 15, (* unknown record number *)
  eIEC870_2COT_UNKNOWN_RECORD_ADDRESS := 16, (* unknown record address *)
  eIEC870_2COT_OBJECT_NOT_FOUND := 17, (* information object not available *)
  eIEC870_2COT_PERIOD_NOT_AVAILABLE := 18, (* requested measurement period not available *)
  eIEC870_2COT_19 := 19,
  eIEC870_2COT_20 := 20,
  eIEC870_2COT_21 := 21,
  eIEC870_2COT_22 := 22,
  eIEC870_2COT_23 := 23,
  eIEC870_2COT_24 := 24,
  eIEC870_2COT_25 := 25,
  eIEC870_2COT_26 := 26,
  eIEC870_2COT_27 := 27,
  eIEC870_2COT_28 := 28,
  eIEC870_2COT_29 := 29,
  eIEC870_2COT_30 := 30,
  eIEC870_2COT_31 := 31,
  eIEC870_2COT_32 := 32,
  eIEC870_2COT_33 := 33,
  eIEC870_2COT_34 := 34,
  eIEC870_2COT_35 := 35,
  eIEC870_2COT_36 := 36,
  eIEC870_2COT_37 := 37,
  eIEC870_2COT_38 := 38,
  eIEC870_2COT_39 := 39,
  eIEC870_2COT_40 := 40,
  eIEC870_2COT_41 := 41,
  eIEC870_2COT_42 := 42,
  eIEC870_2COT_43 := 43,
  eIEC870_2COT_44 := 44,
  eIEC870_2COT_45 := 45,
  eIEC870_2COT_46 := 46,
  eIEC870_2COT_47 := 47,
  eIEC870_2COT_48 := 48,
  eIEC870_2COT_49 := 49,
  eIEC870_2COT_50 := 50,
  eIEC870_2COT_51 := 51,
  eIEC870_2COT_52 := 52,
  eIEC870_2COT_53 := 53,

```



```
eIEC870_2COT_54 := 54,
eIEC870_2COT_55 := 55,
eIEC870_2COT_56 := 56,
eIEC870_2COT_57 := 57,
eIEC870_2COT_58 := 58,
eIEC870_2COT_59 := 59,
eIEC870_2COT_60 := 60,
eIEC870_2COT_61 := 61,
eIEC870_2COT_62 := 62,
eIEC870_2COT_63 := 63
);
END_TYPE
```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.37 E\_IEC870\_5\_103MCOT**

Übertragungsursachen in Überwachungsrichtung (slave -> master) nach IEC 60870-5-103.

**Syntax**

```
TYPE E_IEC870_5_103MCOT:
(
eIEC870_MCOT_UNUSED := 0, (* Not used *)
eIEC870_MCOT_SPONTAN := 1, (* Spontaneous data *)
eIEC870_MCOT_CYCLIC := 2, (* Cyclic data *)
eIEC870_MCOT_FCB := 3, (* Reset FCB bit *)
eIEC870_MCOT_CU := 4, (* Reset communication unit *)
eIEC870_MCOT_SR := 5, (* Start/Restart *)
eIEC870_MCOT_ON := 6, (* Power on *)
eIEC870_MCOT_TST := 7, (* Test mode *)
eIEC870_MCOT_SYN := 8, (* Time synchronisation *)
eIEC870_MCOT_GI := 9, (* General interrogation *)
eIEC870_MCOT_TGI := 10, (* Termination of general interrogation *)
eIEC870_MCOT_LO := 11, (* Local operation *)
eIEC870_MCOT_RO := 12, (* Remote operation *)
eIEC870_MCOT_CP := 20, (* Positive ack of command *)
eIEC870_MCOT_CN := 21, (* Negative ack of command *)
eIEC870_MCOT_TOV := 31, (* Transmission of disturbance values *)
eIEC870_MCOT_WP := 40, (* Positive ack of generic write command *)
eIEC870_MCOT_WN := 41, (* Negative ack of generic write command *)
eIEC870_MCOT_RP := 42, (* Valid data response to generic read command *)
eIEC870_MCOT_RN := 43, (* Invalid data response to generic read command *)
eIEC870_MCOT_CWC := 44 (* Confirmation of generic write *)
);
END_TYPE
```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.38 E\_IEC870\_5\_103CCOT

Übertragungsursachen in Steuerungsrichtung (master -> slave) nach IEC 60870-5-103.

#### Syntax

```
TYPE E_IEC870_5_103CCOT:
(
  eIEC870_CCOT_UNUSED := 0, (* Not used *)
  eIEC870_CCOT_SYN := 8, (* Time synchronisation *)
  eIEC870_CCOT_IGI := 9, (* Initialisation of general interrogation *)
  eIEC870_CCOT_GRC := 20, (* General command *)
  eIEC870_CCOT_TOV := 31, (* Transmission of disturbance values *)
  eIEC870_CCOT_WC := 40, (* Generic write command *)
  eIEC870_CCOT_RC := 42 (* Generic read command *)
);
END_TYPE
```

#### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.39 E\_IEC870\_5\_101FifoDbgFlags

Debug-Ausgabe-Flags. Bei einer aktivierten Debug-Ausgabe werden die ASDU-Hexbytes im TwinCAT XAE "Error List" Fenster geloggt.

#### Syntax

```
TYPE E_IEC870_5_101FifoDbgFlags:
(
  eIEC870_FIFO_DBG_OFF := 0,
  eIEC870_FIFO_DBG_PUT := 1,
  eIEC870_FIFO_DBG_GET := 2,
  eIEC870_FIFO_DBG_ALL := 3
);
END_TYPE
```



Beachten Sie, dass die Debug-Ausgabe eine höhere Systemauslastung erzeugt.

Wert	Beschreibung
eIEC870_FIFO_DBG_OFF	Keine Debug-Ausgabe
eIEC870_FIFO_DBG_PUT	Debug-Ausgabe beim Hinzufügen der Fifo-Elemente
eIEC870_FIFO_DBG_GET	Debug-Ausgabe beim Entfernen der Fifo-Elemente
eIEC870_FIFO_DBG_ALL	Debug-Ausgabe beim Hinzufügen und Entfernen der Fifo-Elemente

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.40 E\_IEC870\_5\_101AODBType**

Typ der Applikationsdatenbank. Zu Zeit wird nur der Typ eIEC870\_AODB\_STATIC unterstützt. D.h. alle Applikationsobjekte (Datenpunkte) müssen von der SPS-Applikation konfiguriert werden.

**Syntax**

```
TYPE E_IEC870_5_101AODBType :
(
    eIEC870_AODB_STATIC := 0,
    eIEC870_AODB_DYNAMIC := 1
);
END_TYPE
```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.41 E\_IEC870\_5\_101InitSeqStep**

Auswahl an Befehlen, die eine Zentralstation (nur High-Level) nach der Herstellung der Kommunikationsverbindung und/oder nach dem Empfang der ASDU M\_EI\_NA\_1 (Ende der Initialisierung) zur Unterstation senden soll. Mehrere dieser Befehle können zu einer Initialisierungssequenz (aufeinanderfolgende Initialisierungsschritte) konfiguriert werden. Die gesamte Konfiguration der zyklischen Datenerfassung (und darunter auch der Initialisierungssequenz) erfolgt über die Komponenten des strukturierten Typs: [ST IEC870\\_5\\_101AcquisitionParams \[▶ 491\]](#).

Die Initialisierungssequenz wird hauptsächlich in der [ST IEC870\\_5\\_101AcquisitionParams.arrInitSeq \[▶ 492\]](#)-Komponente konfiguriert. Einige der Befehle benötigen aber zusätzliche Konfigurationsparameter. Diese Parameter befinden sich dann in weiteren [ST IEC870\\_5\\_101AcquisitionParams \[▶ 491\]](#)-Komponenten. Beispiele:

- eIEC870\_ISTEP\_GENRO-Befehl verwendet standardmäßig den Parameterwert: [ST IEC870\\_5\\_101AcquisitionParams.arrGenro \[▶ 492\]\[0\].QOI := eIEC870\\_QOI\\_INROGEN](#).
- eIEC870\_ISTEP\_GENRO10-Befehl verwendet standardmäßig den Parameterwert [ST IEC870\\_5\\_101AcquisitionParams.arrGenro \[▶ 492\]\[10\].QOI := eIEC870\\_QOI\\_INRO10](#).
- eIEC870\_ISTEP\_CORO-Befehl verwendet standardmäßig die Parameterwerte: [ST IEC870\\_5\\_101AcquisitionParams.arrCoro \[▶ 493\]\[0\].eRQT := eIEC870\\_RQT\\_REQCOGEN](#) und [ST IEC870\\_5\\_101AcquisitionParams.arrCoro \[▶ 493\]\[0\].eFRZ := eIEC870\\_FRZ\\_READ](#).

- eIEC870\_ISTEP\_CORO1-Befehl verwendet standardmäßig die Parameterwerte:  
ST IEC870 5 101AcquisitionParams.arrCoro [▶ 493][1].eRQT := eIEC870\_RQT\_REQCO1 und  
ST IEC870 5 101AcquisitionParams.arrCoro [▶ 493][1].eFRZ := eIEC870\_FRZ\_READ.
- eIEC870\_ISTEP\_DELAY-Befehl verwendet standardmäßig den Parameterwert:  
ST IEC870 5 101AcquisitionParams.delayCmd [▶ 493].tDelay := T#5S.

Ab der Tc2\_IEC60870\_5\_10x.compiled-library v3.0.7.0 kann der Trigger/Zeitpunkt der Abarbeitung der Initialisierungssequenz über die ST IEC870 5 101AcquisitionParams.eMode [▶ 492]-Komponente konfiguriert werden. Standardmäßig wird die Abarbeitung der Initialisierungssequenz nach dem „STARTDT“ (IEC 60870-5-104) oder „Reset of link“ (IEC 60870-5-101) gestartet.

Die Zentralstation-Funktionsbausteine (High-Level) definieren bereits eine Eingangsvariable: FB IEC870 5 101Master [▶ 385].acqPara oder FB IEC870 5 104Master [▶ 389].acqPara vom Typ: ST IEC870 5 101AcquisitionParams. Mehr Kontrolle über die zyklische Datenerfassung bieten die Low-Level-Funktionsbausteine. Die Low-Level-Funktionsbausteine erfordern aber mehr Programmieraufwand.

## Syntax

```

TYPE E_IEC870_5_101InitSeqStep :
(
  eIEC870_ISTEP_UNUSED      := 0,
  eIEC870_ISTEP_CLOCK      := 1,
  eIEC870_ISTEP_TEST       := 2,
  eIEC870_ISTEP_GENRO      := 3,
  eIEC870_ISTEP_CORO       := 4,
  eIEC870_ISTEP_COMMAND    := 5,
  eIEC870_ISTEP_DELAY      := 6,

  eIEC870_ISTEP_CORO1      := 7,
  eIEC870_ISTEP_CORO2      := 8,
  eIEC870_ISTEP_CORO3      := 9,
  eIEC870_ISTEP_CORO4      := 10,

  eIEC870_ISTEP_GENRO1     := 11,
  eIEC870_ISTEP_GENRO2     := 12,
  eIEC870_ISTEP_GENRO3     := 13,
  eIEC870_ISTEP_GENRO4     := 14,
  eIEC870_ISTEP_GENRO5     := 15,
  eIEC870_ISTEP_GENRO6     := 16,
  eIEC870_ISTEP_GENRO7     := 17,
  eIEC870_ISTEP_GENRO8     := 18,
  eIEC870_ISTEP_GENRO9     := 19,
  eIEC870_ISTEP_GENRO10    := 20,
  eIEC870_ISTEP_GENRO11    := 21,
  eIEC870_ISTEP_GENRO12    := 22,
  eIEC870_ISTEP_GENRO13    := 23,
  eIEC870_ISTEP_GENRO14    := 24,
  eIEC870_ISTEP_GENRO15    := 25,
  eIEC870_ISTEP_GENRO16    := 26
);
END_TYPE

```

**Werte**

Name	Bedeutung
eIEC870_ISTEP_UNUSED	Der Initialisierungsschritt wird nicht benutzt.
eIEC870_ISTEP_CLOCK	Die Zentralstation soll einen Uhrzeitsynchronisationsbefehl zur Unterstation senden.
eIEC870_ISTEP_TEST	Die Zentralstation soll einen Testbefehl zur Unterstation senden.
eIEC870_ISTEP_GENRO	Die Zentralstation soll einen Generalabfragebefehl (Gruppe Allgemein) zur Unterstation senden. Weitere Konfigurationsparameter für diesen Befehl befinden sich in: <a href="#">ST_IEC870_5_101AcquisitionParams.arrGenro</a> [ <a href="#">▶_492</a> ][0].
eIEC870_ISTEP_CORO	Die Zentralstation soll einen Zählerabfragebefehl (Gruppe Allgemein) zur Unterstation senden. Weitere Konfigurationsparameter für diesen Befehl befinden sich in: <a href="#">ST_IEC870_5_101AcquisitionParams.arrCoro</a> [ <a href="#">▶_493</a> ][0].
eIEC870_ISTEP_COMMAND	Die Zentralstation soll alle Kommandos einmalig zur Unterstation senden (Datenpunkte in Steuerungsrichtung: Einzelbefehle, Doppelbefehle, Sollwerte, usw.).
eIEC870_ISTEP_DELAY	Die Zentralstation soll mit der Ausführung des nächsten Initialisierungsschrittes warten. Der Wartezeitparameter für diesen Befehl befindet sich in: <a href="#">ST_IEC870_5_101AcquisitionParams.delayCmd</a> [ <a href="#">▶_493</a> ].
eIEC870_ISTEP_CORO1	Die Zentralstation soll einen Zählerabfragebefehl (Gruppe 1) zur Unterstation senden. Weitere Konfigurationsparameter für diesen Befehl befinden sich in: <a href="#">ST_IEC870_5_101AcquisitionParams.arrCoro</a> [1].
eIEC870_ISTEP_CORO2	Die Zentralstation soll einen Zählerabfragebefehl (Gruppe 2) zur Unterstation senden. Weitere Konfigurationsparameter für diesen Befehl befinden sich in: <a href="#">ST_IEC870_5_101AcquisitionParams.arrCoro</a> [2].
eIEC870_ISTEP_CORO3	Die Zentralstation soll einen Zählerabfragebefehl (Gruppe 3) zur Unterstation senden. Weitere Konfigurationsparameter für diesen Befehl befinden sich in: <a href="#">ST_IEC870_5_101AcquisitionParams.arrCoro</a> [3].
eIEC870_ISTEP_CORO4	Die Zentralstation soll einen Zählerabfragebefehl (Gruppe 4) zur Unterstation senden. Weitere Konfigurationsparameter für diesen Befehl befinden sich in: <a href="#">ST_IEC870_5_101AcquisitionParams.arrCoro</a> [4].
eIEC870_ISTEP_GENRO1	Die Zentralstation soll einen Generalabfragebefehl (Gruppe 1) zur Unterstation senden. Weitere Konfigurationsparameter für diesen Befehl befinden sich in: <a href="#">ST_IEC870_5_101AcquisitionParams.arrGenro</a> [1].
eIEC870_ISTEP_GENRO2	Die Zentralstation soll einen Generalabfragebefehl (Gruppe 2) zur Unterstation senden. Weitere Konfigurationsparameter für diesen Befehl befinden sich in: <a href="#">ST_IEC870_5_101AcquisitionParams.arrGenro</a> [2].
eIEC870_ISTEP_GENRO3	Die Zentralstation soll einen Generalabfragebefehl (Gruppe 3) zur Unterstation senden. Weitere Konfigurationsparameter für diesen Befehl befinden sich in: <a href="#">ST_IEC870_5_101AcquisitionParams.arrGenro</a> [3].
eIEC870_ISTEP_GENRO4	Die Zentralstation soll einen Generalabfragebefehl (Gruppe 4) zur Unterstation senden. Weitere Konfigurationsparameter für diesen Befehl befinden sich in: <a href="#">ST_IEC870_5_101AcquisitionParams.arrGenro</a> [4].
eIEC870_ISTEP_GENRO5	Die Zentralstation soll einen Generalabfragebefehl (Gruppe 5) zur Unterstation senden. Weitere Konfigurationsparameter für diesen Befehl befinden sich in: <a href="#">ST_IEC870_5_101AcquisitionParams.arrGenro</a> [5].

Name	Bedeutung
eIEC870_ISTEP_GENRO6	Die Zentralstation soll einen Generalabfragebefehl (Gruppe 6) zur Unterstation senden. Weitere Konfigurationsparameter für diesen Befehl befinden sich in: ST_IEC870_5_101AcquisitionParams.arrGenro[6].
eIEC870_ISTEP_GENRO7	Die Zentralstation soll einen Generalabfragebefehl (Gruppe 7) zur Unterstation senden. Weitere Konfigurationsparameter für diesen Befehl befinden sich in: ST_IEC870_5_101AcquisitionParams.arrGenro[7].
eIEC870_ISTEP_GENRO8	Die Zentralstation soll einen Generalabfragebefehl (Gruppe 8) zur Unterstation senden. Weitere Konfigurationsparameter für diesen Befehl befinden sich in: ST_IEC870_5_101AcquisitionParams.arrGenro[8].
eIEC870_ISTEP_GENRO9	Die Zentralstation soll einen Generalabfragebefehl (Gruppe 9) zur Unterstation senden. Weitere Konfigurationsparameter für diesen Befehl befinden sich in: ST_IEC870_5_101AcquisitionParams.arrGenro[9].
eIEC870_ISTEP_GENRO10	Die Zentralstation soll einen Generalabfragebefehl (Gruppe 10) zur Unterstation senden. Weitere Konfigurationsparameter für diesen Befehl befinden sich in: ST_IEC870_5_101AcquisitionParams.arrGenro[10].
eIEC870_ISTEP_GENRO11	Die Zentralstation soll einen Generalabfragebefehl (Gruppe 11) zur Unterstation senden. Weitere Konfigurationsparameter für diesen Befehl befinden sich in: ST_IEC870_5_101AcquisitionParams.arrGenro[11].
eIEC870_ISTEP_GENRO12	Die Zentralstation soll einen Generalabfragebefehl (Gruppe 12) zur Unterstation senden. Weitere Konfigurationsparameter für diesen Befehl befinden sich in: ST_IEC870_5_101AcquisitionParams.arrGenro[12].
eIEC870_ISTEP_GENRO13	Die Zentralstation soll einen Generalabfragebefehl (Gruppe 13) zur Unterstation senden. Weitere Konfigurationsparameter für diesen Befehl befinden sich in: ST_IEC870_5_101AcquisitionParams.arrGenro[13].
eIEC870_ISTEP_GENRO14	Die Zentralstation soll einen Generalabfragebefehl (Gruppe 14) zur Unterstation senden. Weitere Konfigurationsparameter für diesen Befehl befinden sich in: ST_IEC870_5_101AcquisitionParams.arrGenro[14].
eIEC870_ISTEP_GENRO15	Die Zentralstation soll einen Generalabfragebefehl (Gruppe 15) zur Unterstation senden. Weitere Konfigurationsparameter für diesen Befehl befinden sich in: ST_IEC870_5_101AcquisitionParams.arrGenro[15].
eIEC870_ISTEP_GENRO16	Die Zentralstation soll einen Generalabfragebefehl (Gruppe 16) zur Unterstation senden. Weitere Konfigurationsparameter für diesen Befehl befinden sich in: ST_IEC870_5_101AcquisitionParams.arrGenro[16].

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.42 E\_IEC870\_5\_101FBufferState

Aktueller Offline-ASDU-Dateipuffer-Status.

#### Syntax

```
TYPE E_IEC870_5_101QU :
(
  eIEC870_FBUFFER_IDLE := 0,
  eIEC870_FBUFFER_SAVING := 1,
  eIEC870_FBUFFER_LOADING := 2,
  eIEC870_FBUFFER_ERROR := 3
);
END_TYPE
```

#### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.43 E\_IEC870\_5\_101SCS

Single command state.

#### Syntax

```
TYPE E_IEC870_5_101SCS :
(
  eIEC870_SCS_OFF := 0,
  eIEC870_SCS_ON := 1
);
END_TYPE
```

#### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.44 E\_IEC870\_5\_101DCS

Double command state.

#### Syntax

```
TYPE E_IEC870_5_101DCS :
(
  eIEC870_DCS_INDETERMINATE0 := 0,
  eIEC870_DCS_OFF := 1,
  eIEC870_DCS_ON := 2,
  eIEC870_DCS_INDETERMINATE3 := 3
);
END_TYPE
```



**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.45 E\_IEC870\_5\_101COI**

Cause of initialization.

**Syntax**

```
TYPE E_IEC870_5_101COI :
(
  eIEC870_COI_LOCAL_POWER_ON      := 0, (* Local power ON *)
  eIEC870_COI_LOCAL_MANUAL_RESET  := 1, (* Local manual reset *)
  eIEC870_COI_REMOTE_RESET        := 2 (* Remote reset *)
  (* <3..31> := Reserved for future norm definitions *)
  (* <32..127> := Reserved for user definitions (private range) *)
);
END_TYPE
```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.46 E\_IEC870\_5\_101QOI**

Qualifier of interrogation command.

**Syntax**

```
TYPE E_IEC870_5_101QOI :
(
  eIEC870_QOI_UNUSED              := 0, (* Not used*)
  (* <1..19> := Reserved for standard definitions of this companion standard *)
  eIEC870_QOI_INROGEN              := 20, (* Global station interrogation *)
  eIEC870_QOI_INRO1                := 21, (* Group 1 station interrogation *)
  eIEC870_QOI_INRO2                := 22, (* Group 2 station interrogation...*)
  eIEC870_QOI_INRO3                := 23,
  eIEC870_QOI_INRO4                := 24,
  eIEC870_QOI_INRO5                := 25,
  eIEC870_QOI_INRO6                := 26,
  eIEC870_QOI_INRO7                := 27,
  eIEC870_QOI_INRO8                := 28,
  eIEC870_QOI_INRO9                := 29,
  eIEC870_QOI_INRO10               := 30,
  eIEC870_QOI_INRO11               := 31,
  eIEC870_QOI_INRO12               := 32,
  eIEC870_QOI_INRO13               := 33,
  eIEC870_QOI_INRO14               := 34,
  eIEC870_QOI_INRO15               := 35,
  eIEC870_QOI_INRO16               := 36 (* Group 16 station interrogation*)
  (* <37..63> := Reserved for future norm definitions of this companion standard (compatible range)

```

```

) *)
(* <64..255> := Reserved for user definitions (private range) *)
);
END_TYPE

```

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.3.47 E\_IEC870\_5\_101QL

Qualifier of set-point command.

### Syntax

```

TYPE E_IEC870_5_101QL :
(
  eIEC870_QL_DEFAULT := 0
  (* <1..63> := Reserved for standard definitions (compatible range) *)
  (* <64..127> := Reserved for special use (private range) *)
);
END_TYPE

```

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.3.48 E\_IEC870\_5\_101FRZ

Freeze/reset qualifier of counter interrogation command.

### Syntax

```

TYPE E_IEC870_5_101FRZ :
(
  eIEC870_FRZ_READ := 0, (* Read only (no freeze or reset) *)
  eIEC870_FRZ_FREEZE := 1, (* Counter freeze without reset (value frozen represents integrated total) *)
  eIEC870_FRZ_FREEZE_AND_RESET := 2, (* Counter freeze with reset (value frozen represents incremental information) *)
  eIEC870_FRZ_RESET := 3 (* Counter reset *)
);
END_TYPE

```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.49 E\_IEC870\_5\_101RQT**

Request qualifier of counter interrogation command.

**Syntax**

```

TYPE E_IEC870_5_101RQT :
(
  eIEC870_RQT_NONE           := 0, (* No counter read *)
  eIEC870_RQT_REQCO1        := 1, (* Group 1 counter interrogation *)
  eIEC870_RQT_REQCO2        := 2, (* Group 2 counter interrogation *)
  eIEC870_RQT_REQCO3        := 3, (* Group 3 counter interrogation *)
  eIEC870_RQT_REQCO4        := 4, (* Group 4 counter interrogation *)
  eIEC870_RQT_REQCOGEN      := 5 (* General counter interrogation *)
  (* <6..31> := Reserved for future norm definitions *)
  (* <32..63> := Reserved for user definitions (private range) *)
);
END_TYPE
    
```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.50 E\_IEC870\_5\_101QRP**

Qualifier des Prozess-Reset-Befehls.

**Syntax**

```

TYPE E_IEC870_5_101QRP :
(
  eIEC870_QRP_UNUSED        := 0, (* Not used *)
  eIEC870_QRP_GENERAL        := 1, (* General process reset *)
  eIEC870_QRP_TTEVENTS      := 2 (* Reset pending events with time tag *)
  (* <3..127> := Reserved for future norm definitions *)
  (* <128..255> := Reserved for user definitions (private range) *)
);
END_TYPE
    
```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.51 E\_IEC870\_5\_101QU**

Qualifier der Befehle.

**Syntax**

```

TYPE E_IEC870_5_101QU :
(
  eIEC870_QU_UNSPECIFIED := 0,
  eIEC870_QU_SHORTPULSE := 1,
  eIEC870_QU_LONGPULSE := 2,
  eIEC870_QU_PERSISTENT := 3
  (* <4..8> := Reserved for standrad definitions of companion standard (compatible range) *)
  (* <9..15> := Reserved for the selection of other predefined functions *)
  (* <16..31> := Reserved for special use (private range) *)
);
END_TYPE

```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.52 E\_IEC870\_5\_101ES**

Event state (single event of protection equipment).

**Syntax**

```

TYPE E_IEC870_5_101ES:
(
  eIEC870_ES_INDETERMINATE0 := 0,
  eIEC870_ES_OFF := 1,
  eIEC870_ES_ON := 2,
  eIEC870_ES_INDETERMINATE3 := 3
);
END_TYPE

```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.53 E\_IEC870\_5\_101KPA**

Kind of parameter.

**Syntax**

```

TYPE E_IEC870_5_101KPA:
(
  eIEC870_KPA_UNUSED := 0,
  eIEC870_KPA_THRESH := 1,
  eIEC870_KPA_FILTER := 2,
  eIEC870_KPA_LOLIMIT := 3,
  eIEC870_KPA_HILIMIT := 4
  (* <5..31> := Reserved for standard definitions (compatible range) *)
  (* <32..63> := Reserved for special use (private range) *)
);
END_TYPE
    
```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.54 E\_IEC870\_5\_101QPA**

Qualifier der Parameteraktivierung.

**Syntax**

```

TYPE E_IEC870_5_101QPA:
(
  eIEC870_QPA_UNUSED := 0,
  eIEC870_QPA_GENERAL := 1,
  eIEC870_QPA_OBJECT := 2,
  eIEC870_QPA_TRANSMISSION := 3
  (* <4..127> := Reserved for standard definitions (compatible range) *)
  (* <128..255> := Reserved for special use (private range) *)
);
END_TYPE
    
```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.55 E\_IEC870\_5\_101RCS**

Statusinformation des Schrittbefehls.

**Syntax**

```
TYPE E_IEC870_5_101RCS :
(
  eIEC870_RCS_NOTALLOWED0 := 0,
  eIEC870_RCS_DECREMENT  := 1,
  eIEC870_RCS_INCREMENT  := 2,
  eIEC870_RCS_NOTALLOWED3 := 3
);
END_TYPE
```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.56 E\_IEC870\_5\_101SPI**

Single-point information.

**Syntax**

```
TYPE E_IEC870_5_101SPI :
(
  eIEC870_SPI_OFF := 0,
  eIEC870_SPI_ON  := 1
);
END_TYPE
```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.57 E\_IEC870\_5\_101DPI

Double-point information.

#### Syntax

```
TYPE E_IEC870_5_101DPI :
(
  eIEC870_DPI_INDETERMINATE0 := 0,
  eIEC870_DPI_OFF             := 1,
  eIEC870_DPI_ON              := 2,
  eIEC870_DPI_INDETERMINATE3 := 3
);
END_TYPE
```

#### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.58 E\_IEC870\_5\_101FrameType

IEC 60870-5-10x Frame Type. Zur Zeit wird nur FT1.2 unterstützt.

#### Syntax

```
TYPE E_IEC870_5_101FrameType :
(
  eIEC870_FrameType_FT1_2
);
END_TYPE
```

#### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.59 E\_IEC870\_5\_101LinkMode

Konfiguriert den Datenübertragungsmodus.

#### Syntax

```
TYPE E_IEC870_5_101LinkMode :
(
  eIEC870_LinkMode_Unbalanced,
  eIEC870_LinkMode_Balanced
);
END_TYPE
```

**eIEC870\_LinkMode\_Unbalanced:** Asymmetrische/Unsymmetrische Datenübertragung. In diesem Übertragungsmodus steuert die Leitstation (master) den Datenaustausch mit den Unterstationen (slaves). Die Leitstation initiiert immer den Datenaustausch. Die Leitstation fragt auf diese Weise sequentiell alle Unterstationen ab.

**eIEC870\_LinkMode\_Balanced:** Symmetrische Datenübertragung. In diesem Übertragungsmodus kann jede Station den Datenaustausch initiieren und kann nur in "point-to-point" und "multiple-point-to-point"-Konfigurationen verwendet werden.

#### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.60 E\_IEC870\_5\_101SerialLinkState

IEC 60870-5-101/102/103 Verbindungsstatus.

#### Syntax

```
TYPE E_IEC870_5_101SerialLinkState :
(
  eSERIALLINK_DISCONNECTED,
  eSERIALLINK_ESTABLISHED,
  eSERIALLINK_SUSPENDED
);
END_TYPE
```

#### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.61 E\_IEC870\_5\_101PartyLineMode

Partyline Aktivierungsmodus. Siehe auch in der Beschreibung des Funktionsbausteins:

[FB IEC870\\_PartyLineCtrl](#) [▶ 412].

#### Syntax

```
TYPE E_IEC870_5_101PartyLineMode :
(
  eIEC870_PartylineMode_Off := 0, (* Deaktiviert (wird nicht verwendet) *)
  eIEC870_PartylineMode_On := 1, (* Aktiviert *)
  eIEC870_PartylineMode_Ext_On := 2 (* Aktiviert, verwendet "Extended Ctrl/Status" Prozessdaten *)
);
END_TYPE
```



**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.62 E\_IEC870\_5\_101LinkReset**

Reset-Typ während der Initialisierung der Verbindungsschicht.

**Syntax**

```
TYPE E_IEC870_5_101LinkReset :
(
  eIEC870_LinkReset_None := 0, (* Disabled *)
  eIEC870_LinkReset_CU   := 1, (* Reset communication unit *)
  eIEC870_LinkReset_UP   := 2, (* Reset user process *)
  eIEC870_LinkReset_FCB  := 3 (* Reset FCB bit *)
);
END_TYPE
```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.63 E\_IEC870\_DEVICE\_TYPE**

Protokoll- bzw. Stationstyp.

**Syntax**

```
TYPE E_IEC870_DEVICE_TYPE :
(
  eIEC870_101_SLAVE := 0, (* Secondary (responding) station *)
  eIEC870_101_MASTER := 1, (* Primary (initiating) station *)

  eIEC870_102_SLAVE := 2, (* Secondary (responding) station *)
  eIEC870_102_MASTER := 3, (* Primary (initiating) station *)

  eIEC870_103_SLAVE := 4, (* Secondary (responding) station *)
  eIEC870_103_MASTER := 5 (* Primary (initiating) station *)
);
END_TYPE
```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.64 E\_IEC870\_5\_104DataTransferState**

Status des IEC 60870-5-104 Datentransfers.

**Syntax**

```

TYPE E_IEC870_5_104DataTransferState :
(
  eIEC870_STOPDT, (* data exchange deactivated *)
  eIEC870_STARTDT, (* data exchange activated *)
  eIEC870_STOPDT_PENDING, (* waiting for STOPDT confirmation (master only)*)
  eIEC870_STARTDT_PENDING (* waiting for STARTDT confirmation (master only)*)
);
END_TYPE

```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.65 T\_HSERIALCTRL**

Eine Variable von diesem Typ repräsentiert ein Verbindungs-Handle der seriellen Schnittstelle. Die Strukturelemente sollen nicht direkt beschrieben oder verändert werden. Variablen von diesem Typ werden für den Datenaustausch der zu sendeten oder empfangenen Daten zwischen dem Funktionsbaustein [FB\\_IEC870\\_SerialLineCtrl](#) [▶ 417] und anderen IEC 60870-5-101 SPS-Bausteinen benutzt.

**Syntax**

```

TYPE T_HSERIALCTRL
STRUCT
(*...*)
END_STRUCT
END_TYPE

```

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.66 T\_HAODBTable**

Applikationsobjekt-Datenbankhandle (Hash-Tabellenhandle). Das Tabellenhandle muss vor der Benutzung einmalig mit der Funktion [F\\_iecCreateTableHnd](#) [▶ 452] initialisiert werden.

**Syntax**

```
TYPE T_HAODBTable :
STRUCT
(*...*)
END_STRUCT
END_TYPE
```

Auf Variablen dieses strukturierten Typs wird nicht direkt, sondern nur mit Hilfe der zur Verfügung stehenden Funktionen oder Funktionsbausteine zugegriffen. Benutzen Sie die in der Tabelle aufgeführten Funktionen:

Function	Description
<a href="#">F_iecCreateTableHnd</a> [▶ 452]	Initialisiert das Hash-Tabellenhandle
<a href="#">F_iecAddTableEntry</a> [▶ 453]	Konfiguriert und fügt einen einen neuen Hash-Tabelleneintrag
<a href="#">F_iecRemoveTableEntry</a> [▶ 460]	Entfernt einen Hash-Tabelleneintrag
<a href="#">F_iecLookupTableEntry</a> [▶ 458]	Prüft ob ein bestimmter Hash-Tabelleneintrag existiert
<a href="#">F_iecGetPosOfTableEntry</a> [▶ 456]	Ermittelt die lineare Position eines Hash-Tabelleneintrags

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.67 T\_CP56Time2a**

Binäres (7 Byte) Zeitformat (Typ a).

**Syntax**

```
TYPE T_CP56Time2a :
STRUCT
  Milliseconds : WORD;
  IVResMinute : BYTE;
  SRes2Hour : BYTE;
  DOWDay : BYTE;
```

```

    Res3Month      : BYTE;
    Res4Year       : BYTE;
END_STRUCT
END_TYPE

```

**Milliseconds** : Millisekunden := <0..59999>.

**IVResMinute** : Bit 7 = IV := <0=gültige, 1=ungültige Zeit>, Bit 6 = Res (reserviert) := <0>, Bit 0..5 = Minute := <0..59>.

**SURes2Hour** : Bit 7 = SU := <0=Normalzeit, 1=Sommerzeit>, Bits 5..6 = Res2 (reserviert) := <0>, Bits 0..4 = Hour (Stunde) := <0..23>.

**DOWDay** : Bits 5..7 = DOW (Wochentag) := <0=nicht benutzt, 1=Montag..7=Sonntag>, Bits 0..4 = Day (Monatstag) := <1..31>.

**Res3Month** : Bits 4..7 = Res3 (reserviert) := <0>, Bits 0..3 = Month (Monat) := <1..12>.

**Res4Year** : Bit 7 = Res4 (reserviert) := <0>, Bits 0..6 = Year (Jahr) := <0..99>.

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.3.68 T\_CP56Time2b

Binäres (7 Byte) Zeitformat (Typ b).

### Syntax

```

TYPE T_CP56Time2b :
STRUCT
    MsSecond      : WORD := 16#00;
    IVTisMinute   : BYTE := 16#00;
    SURes1Hour    : BYTE := 16#00;
    DOWDay        : BYTE := 16#01;
    PtiEtiMonth   : BYTE := 16#01;
    Res2Year      : BYTE := IEC870_DEFAULT_CP56TIME2A_YEAR;
END_STRUCT
END_TYPE

```

**MsSecond** : Bits 11..15 = Second (Sekunde) := <0..59>, Bits 0..9 = Ms (Millisekunde) := <0..999>.

**IVTisMinute** : Bit 7 = IV := <0=gültige, 1=ungültige Zeit>, Bit 6 = Tif (Tarifinformation) := <0..1>, Bit 0..5 = Minute := <0..59>.

**SURes1Hour** : Bit 7 = SU := <0=Normalzeit, 1=Sommerzeit>, Bits 5..6 = Res1 (reserviert) := <0>, Bits 0..4 = Hour (Stunde) := <0..23>.

**DOWDay** : Bits 5..7 = DOW (Wochentag) := <0=nicht benutzt, 1=Montag..7=Sonntag>, Bits 0..4 = Day (Monatstag) := <1..31>.

**PtiEtiMonth** : Bits 6..7 = Pti (Powertarifinformation) := <0..3>, Bits 4..5 = Eti (Energietarifinformation) := <0..3>, Bits 0..3 = Month (Monat) := <1..12>.

**Res2Year** : Bit 7 = Res2 (reserviert) := <0>, Bits 0..6 = Year (Jahr) := <0..99>.

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.69 T\_CP40Time2a

Binäres (5 Byte) Zeitformat.

Syntax

```

TYPE T_CP40Time2a :
STRUCT
    IVTisMinute : BYTE := 16#00;
    SUREs1Hour : BYTE := 16#00;
    DOWDay : BYTE := 16#01;
    PtiEtiMonth : BYTE := 16#01;
    Res2Year : BYTE := IEC870_DEFAULT_CP56TIME2A_YEAR;
END_STRUCT
END_TYPE
    
```

**IVTisMinute** : Bit 7 = IV := <0=gültige, 1=ungültige Zeit>, Bit 6 = Tif (Tarifinformation) := <0..1>, Bit 0..5 = Minute := <0..59>.

**SUREs1Hour** : Bit 7 = SU := <0=Normalzeit, 1=Sommerzeit>, Bits 5..6 = Res1 (reserviert) := <0>, Bits 0..4 = Hour (Stunde) := <0..23>.

**DOWDay** : Bits 5..7 = DOW (Wochentag) := <0=nicht benutzt, 1=Montag..7=Sonntag>, Bits 0..4 = Day (Monatstag) := <1..31>.

**PtiEtiMonth** : Bits 6..7 = Pti (Powertarifinformation) := <0..3>, Bits 4..5 = Eti (Energietarifinformation) := <0..3>, Bits 0..3 = Month (Monat) := <1..12>.

**Res2Year** : Bit 7 = Res2 (reserviert) := <0>, Bits 0..6 = Year (Jahr) := <0..99>.

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.70 T\_CP32Time2a

Binäres (4 Byte) Zeitformat.

Syntax

```

TYPE T_CP32Time2a :
STRUCT
    Milliseconds : WORD;
    IVResMinute : BYTE;
    
```

```

    SRes2Hour      : BYTE;
END_STRUCT
END_TYPE

```

**Milliseconds** : Millisekunden := <0..59999>.

**IVResMinute** : Bit 7 = IV := <0=gültige, 1=ungültige Zeit>, Bit 6 = Res (reserviert) := <0>, Bit 0..5 = Minute := <0..59>.

**SRes2Hour** : Bit 7 = SU := <0=Normalzeit, 1=Sommerzeit>, Bits 5..6 = Res2 (reserviert) := <0>, Bits 0..4 = Hour (Stunde) := <0..23>.

#### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.71 T\_CP24Time2a

Binäres (3 Byte) Zeitformat.

#### Syntax

```

TYPE T_CP24Time2a :
STRUCT
    Milliseconds      : WORD;
    IVResMinute       : BYTE;
END_STRUCT
END_TYPE

```

**Milliseconds** : Millisekunden := <0..59999>.

**IVResMinute** : Bit 7 = IV := <0=gültige, 1=ungültige Zeit>, Bit 6 = Res (reserviert) := <0>, Bit 0..5 = Minute := <0..59>.

#### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.72 T\_CP16Time2a

Binäres (2 Byte) Zeitformat.

#### Syntax

```

TYPE T_CP16Time2a :
STRUCT
    Milliseconds      : WORD;
END_STRUCT
END_TYPE

```

**Milliseconds** : Millisekunden := <0..59999>.

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.73 T\_CP24IOA**

Strukturierte TwinCAT Objektadresse.

**Syntax**

```

TYPE T_CP24IOA:
STRUCT
    ioMap      : E_IEC870_5_101IOMappingType := MAP_AREA_NONE;
    byteOffs   : WORD := 0;
    bitOffs    : BYTE(0..7) := 0;
END_STRUCT
END_TYPE
    
```

**ioMapType:** TwinCAT SPS-Prozessdatenbereich [[▶ 506](#)]. Dieser Parameter legt fest wie die TwinCAT SPS und IEC-Applikationsobjekt Prozessdaten gemappt werden sollen.

**byteOffs:** TwinCAT SPS-Prozessdaten Byte-Offset. Wertebereich: 0..65535.

**bitOffs:** TwinCAT SPS-Prozessdaten Bit-Offset. Wertebereich: 0..7.

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.3.74 T\_IEC870\_5\_101COTBits**

Übertragungsursache Bitmaske.

**Syntax**

```

TYPE T_IEC870_5_101COTBits : ARRAY[0..7] OF BYTE;
END_TYPE
    
```

## Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.75 ST\_IEC870\_5\_101FBufferCfg

Konfigurationseinstellungen für den Offline-ASDU-Dateipuffer. Diese Struktur wird von dem [FB IEC870\\_5\\_101FBufferCtrl](#) [► 399] Funktionsbaustein benutzt.

#### Syntax

```

TYPE ST_IEC870_5_101FBufferCfg :
STRUCT
  sNetID      : T_AmsNetID := '';
  sPathName   : T_MaxString := 'c:\Temp\data.dat';
  ePath       : E_OpenPath := PATH_GENERIC;
  cbBuffer    : UDINT := 16#100000;
  bOverwrite  : BOOL := TRUE;
  bFilter     : BOOL := FALSE;
  cotFilter   : T_IEC870_5_101COTBits := 8(0);
  tTimeout    : TIME := DEFAULT_ADS_TIMEOUT;
  bContinueAfterOverflow : BOOL := FALSE;
  bResetTxBufferOnOverwrite : BOOL := FALSE;
END_STRUCT
END_TYPE

```

**sNetID:** Hier kann die Netzwerkadresse des TwinCAT-Rechners angegeben werden, auf dem die Pufferdatei erstellt werden soll. Für den lokalen PC kann auch ein Leerstring angegeben werden.

**sPathName:** Enthält den Pfad- und Dateinamen der zu öffnenden Datei (max. 255 Zeichen). Der Pfad kann nur auf das lokale Dateisystem des Rechners zeigen. Das bedeutet, Netzwerkpfade können hier nicht angegeben werden.

**ePath:** Über diesen Eingang kann ein TwinCAT-Systemdateipfad auf dem Zielgerät zum Öffnen der Datei angewählt werden.

**cbBuffer:** Max. Bytegröße der Pufferdatei. Wenn **bOverwrite** = FALSE gesetzt wurde und die max. Größe überschritten wurde wird ein Fehler zurückgemeldet. Defaulteinstellung: 16#100000 = 1MB.

**bOverwrite:** Beim Erreichen der max. Größe werden die ältesten Einträge überschrieben wenn diese Variable auf TRUE gesetzt wurde.

**bFilter:** Zur Zeit noch nicht implementiert. Aktiviert/deaktiviert einen COT-Filter. Nur ASDUs mit bestimmten Übertragungsursachen werden in die Datei gepuffert.

**cotFilter:** Zur Zeit noch nicht implementiert. Über diese Variable können Übertragungsursachen ([COT-Filter](#) [► 535]) konfiguriert werden, die in die Datei gepuffert werden sollen.

**tTimeout:** Maximale Timeoutzeit die bei dem Dateizugriff nicht überschritten werden sollte.

**bContinueAfterOverflow:** TRUE => Ignoriere Dateiüberlauffehler wenn max. Dateigröße erreicht wurde (nur bei **bOverwrite** = FALSE und TF6500 v3.0.6.0 und höher).

**bResetTxBufferOnOverwrite:** TRUE => Setze die noch nicht gesendeten Tx-Daten im Memory-Buffer zurück sobald der erste Eintrag im Dateipuffer überschrieben wurde (nur bei **bOverwrite** = TRUE und TF6500 v3.0.6.0 und höher).



Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.76 ST\_IEC870\_5\_101FBufferStatus

Statusinformationen zum Offline-ASDU-Dateipuffer. Diese Struktur wird von dem FB IEC870 5\_101FBufferCtrl [[▶ 399](#)] Funktionsbaustein benutzt.

Syntax

```

TYPE ST_IEC870_5_101FBufferStatus:
STRUCT
  eState      : E_IEC870_5_101FBufferState := eIEC870_FBUFFER_IDLE;
  nErrID      : UDINT;
  bCorrupted  : BOOL;
  nCount      : UDINT;
  bOverwriting: BOOL;
  nOverwrites : UDINT;
  bOverflowing: BOOL;
END_STRUCT
END_TYPE
    
```

**eState:** Liefert den Dateipufferstatus [[▶ 520](#)] (Datei wird geschrieben, gelesen, ist geschlossen oder es ist ein Fehler beim Schreiben/Lesen aufgetreten).

**nErrID:** Liefert bei einem Dateizugriffsfehler die ADS-Fehlernummer.

**bCorrupted:** Bei TRUE war die zuletzt geöffnete Datei als korrupt erkannt und wurde durch eine neue leere Datei ersetzt. Eine korrupte Datei kann z.B. dann entstehen wenn die max. Dateipuffergröße geändert wurde oder die Datei nicht richtig geschlossen wurde.

**nCount:** Aktuelle Anzahl der gepufferten Einträge im Dateipuffer. Der Dateipuffer muss zuerst geöffnet werden um die Anzahl der Einträge in einer existierenden Datei ermitteln zu können. D.h. die Verbindung muss kurz in den Offline Mode gehen.

**bOverwriting:** TRUE => Der älteste Dateieintrag wird gerade überschrieben (nur bei TF6500 v3.0.6.0 und höher).

**nOverwrites:** Anzahl der Dateieinträge die überschrieben wurden (nur bei TF6500 v3.0.6.0 und höher).

**bOverflowing:** TRUE => Max. Dateigröße wurde erreicht oder der letzte Eintrag konnte nicht geschrieben werden (nur bei TF6500 v3.0.6.0 und höher).

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.77 ST\_IEC870\_5\_101ProtocolParams

Protokollparameter der Verbindungsschicht. Die Parameter haben abhängig von dem Protokoll- oder Stationstyp unterschiedliche Bedeutung.

#### Syntax

```

TYPE ST_IEC870_5_101ProtocolParams :
STRUCT
  eType           : E_IEC870_DEVICE_TYPE           := eIEC870_101_SLAVE;
  eLinkReset      : E_IEC870_5_101LinkReset        := eIEC870_LinkReset_CU;
  linkAddr        : DWORD                           := 1;
  eLinkAddrSize   : E_IEC870_5_101LinkAddrSize     := eIEC870_LinkAddr_TwoOctets;
  eLinkMode       : E_IEC870_5_101LinkMode         := eIEC870_LinkMode_Unbalanced;
  eFrameType      : E_IEC870_5_101FrameType        := eIEC870_FrameType_FT1_2;
  tRxTimeout      : TIME := T#5s;
  tTxTimeout      : TIME := T#5s;
  bForceC1Res     : BOOL := TRUE;
  bForceC2Res     : BOOL := TRUE;
  tClass1Poll     : TIME := T#200ms;
  tClass2Poll     : TIME := T#200ms;
  nRetries        : BYTE := 3;
  tRetry          : TIME := T#100ms;
  tResponse       : TIME := T#1s;
  tTestLink       : TIME := T#5s;
  tPollDFC        : TIME := T#1s;
  FRAMELength     : BYTE (MIN_IEC870_5_101Link_FRAMELEN..MAX_IEC870_5_101Link_FRAMELEN) := MAX_IEC870_5_101Link_FRAMELEN;
  bRetainBuffer   : BOOL := FALSE;
  tMaxPollDelay   : TIME := T#0s;
  tLinkPollCycle  : TIME := T#10s;
END_STRUCT
END_TYPE

```

In der unterstehenden Tabelle finden Sie Hinweise zur Konfiguration. Bei fixen Werten muss der Parameter dem Wert aus der Tabelle entsprechen (nicht konfigurierbar).

#### Legende:

- **X** Der Parameter wird benutzt und ist konfigurierbar;
- **N/A** Der Parameter wird nicht benutzt und ist nicht konfigurierbar;

Parametername	Initialisierungswert	IEC 60870-5-101 controlled station (slave, unbalanced mode)	IEC 60870-5-101 control station (master, unbalanced mode)	IEC 60870-5-103 control station (master, unbalanced mode)	IEC 60870-5-102 control station (master, unbalanced mode)	Beschreibung
<a href="#">eType</a> [► 529]	eIEC870_101_SLAVE	fix, eIEC870_101_SLAVE	fix, eIEC870_101_MASTER	fix, eIEC870_103_MASTER	fix, eIEC870_102_MASTER	Konfiguriert den Protokoll- und Stationstyp.
<a href="#">eLinkReset</a> [► 529]	eIEC870_Link_Reset_CU	N/A	X	X	fix, eIEC870_Link_Reset_CU	Reset-Typ bei der Intilialisierung der Verbindungsschicht.
<b>linkAddr</b>	1	eigene Adresse	Adresse des Remote-Teilnehmers	Adresse des Remote-Teilnehmers	Adresse des Remote-Teilnehmers	Verbindungsadresse.
<a href="#">eLinkAddrSize</a> [► 507]	eIEC870_Link_Addr_TwoOctets	X	X	X	X	Octetgröße der Verbindungsadresse.
<a href="#">eLinkMode</a> [► 527]	eIEC870_Link_Mode_Unbalanced	fix, eIEC870_Link_Mode_Unbalanced	fix, eIEC870_Link_Mode_Unbalanced	fix, eIEC870_Link_Mode_Unbalanced	fix, eIEC870_Link_Mode_Unbalanced	Verbindungs-Mode (balanced/unbalanced).
<a href="#">eFrameType</a> [► 527]	eIEC870_FrameType_FT1_2	fix, eIEC870_FrameType_FT1_2	fix, eIEC870_FrameType_FT1_2	fix, eIEC870_FrameType_FT1_2	fix, eIEC870_FrameType_FT1_2	Telegramm-Frameformat. Zur Zeit wird nur das Format F1.2 unterstützt.
<b>tRxTimeout</b>	T#5s	X	X	X	X	Max. Frame-Receive-Timeout-Zeit.
<b>tTxTimeout</b>	T#5s	X	X	X	X	Max. Frame-Send-Timeout-Zeit.

Parametername	Initialisierungswert	IEC 60870-5-101 controlled station (slave, unbalanced mode)	IEC 60870-5-101 control station (master, unbalanced mode)	IEC 60870-5-103 control station (master, unbalanced mode)	IEC 60870-5-102 control station (master, unbalanced mode)	Beschreibung
bForceC1Res	TRUE	X	N/A	N/A	N/A	<p>Ist dieser Parameter gesetzt (TRUE), dann werden Class 1 Daten als Antwort auf Class 2 Anfrage gesendet wenn Class 1 Daten vorliegen (optimierter, höherer Datendurchsatz für Class 1).</p> <p>Ist dieser Parameter nicht gesetzt (FALSE) dann wird negative Antwort auf Class 2 Anfrage gesendet wenn Class 1 Daten vorliegen (kleinerer Datendurchsatz, leere Telegramme).</p>

Parametername	Initialisierungswert	IEC 60870-5-101 controlled station (slave, unbalanced mode)	IEC 60870-5-101 control station (master, unbalanced mode)	IEC 60870-5-103 control station (master, unbalanced mode)	IEC 60870-5-102 control station (master, unbalanced mode)	Beschreibung
<b>bForceC2Res</b>	TRUE	X	N/A	N/A	N/A	<p>Ist dieser Parameter gesetzt (TRUE), dann werden Class 2 Daten als Antwort auf Class 2 Anfrage gesendet obwohl Class 1 Daten vorliegen (optimierter, höherer Datendurchsatz für Class 2).</p> <p>Ist dieser Parameter nicht gesetzt (FALSE) dann wird negative Antwort auf Class 2 Anfrage gesendet wenn Class 1 Daten vorliegen (kleinerer Datendurchsatz, leere Telegramme)</p>
<b>tClass1Poll</b>	T#200ms	N/A	X	X	X	Zykluszeit in der die Class 1 Daten abgefragt werden sollen (unbalanced mode only).
<b>tClass2Poll</b>	T#200ms	N/A	X	X	X	Zykluszeit in der die Class 2 Daten abgefragt werden sollen (unbalanced mode only).

Parametername	Initialisierungswert	IEC 60870-5-101 controlled station (slave, unbalanced mode)	IEC 60870-5-101 control station (master, unbalanced mode)	IEC 60870-5-103 control station (master, unbalanced mode)	IEC 60870-5-102 control station (master, unbalanced mode)	Beschreibung
<b>nRetries</b>	3	bis zur v3.0.6 nicht benutzt (Fehler beim ersten gestörten Frame), ab der v3.0.8 und höher konfigurierbar	X	X	X	Maximale Anzahl der Telegrammwiederholungen beim gestörten Datenaustausch.
<b>tRetry</b>	T#100ms	N/A	X	X	X	Verzögerungszeit für Telegrammwiederholungen beim gestörten Datenaustausch.
<b>tResponse</b>	T#1s	N/A	X	X	X	Dieser Parameter bestimmt die max. Wartezeit auf eine Telegrammbestätigung. Beim Überschreiten dieser Zeit wird das Telegramm wiederholt, aber max. bis zu <i>nRetry</i> -Male, übertragen.
<b>tTestLink</b>	T#5s	N/A	N/A	N/A	N/A	Zykluszeit in der Test-Telegramme verschickt werden sollen (balanced mode only)

Parametername	Initialisierungswert	IEC 60870-5-101 controlled station (slave, unbalanced mode)	IEC 60870-5-101 control station (master, unbalanced mode)	IEC 60870-5-103 control station (master, unbalanced mode)	IEC 60870-5-102 control station (master, unbalanced mode)	Beschreibung
tPollDFC	T#1s	N/A	X	X	X	Zykluszeit in der beim gesetzten DFC-Bit der Verbindungsstatus abgefragt werden soll. Das DFC-Bit wird von dem Remote-Teilnehmer dann gesetzt wenn weitere Telegramme einen Überlauf verursachen könnten.
FRAMELength	MAX_IEC870_5_101Link_FRAMELEN	X	X	X	X	Telegrammlänge, max. Länge L.
bRetainBuffer	FALSE	ab der Produktversion v3.0.2 und höher	X	X	X	Ist dieser Parameter gesetzt, dann werden die internen TX/RX-Puffer bei der Initialisierung der Verbindungsschicht nicht zurückgesetzt (gelöscht).
tMaxPollDelay	T#0s (= 0: deaktiviert, <> 0: aktiv)	ab der Produktversion v3.0.2 und höher	N/A	N/A	N/A	Beim Fehlen der Poll-Anfragen wird nach der Überschreitung dieser Zeit der Verbindungsstatus als OFFLINE deklariert.
tLinkPollCycle	T#10s	N/A	X	X	X	Zykluszeit in der bei einer fehlenden Verbindung link status telegramme an die Unterstation gesendet werden.

Zusätzliche Hinweise zu **bForceC1Res** und **bForceC2Res**-Parametern:

Unabhängig davon ob Sie beide oder nur eines der Parameter setzen oder nicht wird das ACD-Bit auf jeden Fall passend gesetzt um der Leitstation mitzuteilen ob Class 1 oder Class 2 Daten als nächstes abgefragt werden sollen. Nur bei Classe 2 Abfragen wird das Verhalten der Unterstation von diesen Parametern beeinflusst.

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.3.78 ST\_IEC870\_5\_102TBuffer

Diese Datenstruktur (TX/RX-Datenpuffer) wird beim Datenaustausch über das IEC 60870-5-102 Low-Level-Interface benutzt.

### Syntax

```

TYPE ST_IEC870_5_102TBuffer :
STRUCT
  eDbg          : E_IEC870_5_101FifoDbgFlags :=eIEC870_FIFO_DBG_OFF;
  eAsduAddrSize: E_IEC870_5_101AsduAddrSize;
  asduSize     : BYTE := 253;
  mode        : DWORD := 0;
  dataLink     : ST_IEC870_5_101DataLink;
END_STRUCT
END_TYPE

```

**eDbg**: Debug-Ausgabe-Parameter [[▶ 514](#)].

**eAsduAddrSize**: Oktetlänge der gemeinsamen ASDU-Adresse [[▶ 506](#)] (CAA) := <1..2>.

**asduSize**: Maximale Bytelänge der ASDU := <50..253>.

**mode**: Reserviert, wird zur Zeit nicht benutzt. Dieser Wert sollte Null sein.

**dataLink**: Auf die Elemente dieser Datenstruktur sollte nicht direkt, sondern nur mit einer Instanz des [FB IEC870\\_5\\_102TBufferCtrl](#) [[▶ 405](#)]-Funktionsbausteins zugegriffen werden.

Der TX/RX-Datenpuffer verwendet intern zwei Sendefifos und einen Empfangs-Fifo:

1. Class 1 Sendefifo mit (hochprioren) Daten;
2. Class 2 Sendefifo mit (niederprioren) Daten;
3. Receive-Fifo (für Class 1 und Class 2 Daten);

Die unteren Transportfunktionen der Bibliothek leeren zuerst den Class 1-Fifo und dann den Class 2-Fifo. Die Class 2 Daten werden nur dann versendet wenn der Class 1-Fifo keine zu versendenden Daten enthält.

Jeder der internen Fifos hat eine feste Größe von 200000 Bytes. Erfahrungsgemäß können in jedem Fifo ca. 2000 ASDUs mit einem Informations-Element (Objekt) mit der maximalen Größe oder ca. 600 ASDUs mit einer Sequence von 100 Informations-Elementen (Objekten) abgelegt werden.

Wenn eine größere Anzahl der zu versendenden oder zu empfangenen Frames zwischengespeichert werden soll (z.B. >2000), so können diese in externen, vom SPS-Programmierer festgelegten Puffern/Fifos zwischengehalten werden. Die SPS-Applikation kann dann zur Laufzeit die TwinCAT-Sendefifos mit den eigenen Fifo-Einträgen nachfüllen oder bei vielen empfangenen Frames den TwinCAT-Receive-Fifo leeren. Eine andere Möglichkeit ist z.B. zwei Puffer zu benutzen und diese abwechselnd zu füllen/lesen und an den Kommunikationsbaustein übergeben.



Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.79 ST\_IEC870\_5\_102AOGen

Variablen von diesem Typ repräsentieren ein ASDU-Objekt.

Syntax

```

TYPE ST_IEC870_5_102AOGen:
STRUCT
  head : ST_IEC870_5_101FifoHead :=( source := ( link := 0, addr := 0 ),
                                     target := ( link := 0, addr := 0 ),
                                     ctrl := 0 );

  ident : ST_IEC870_5_102DataUnit_Ident := ( eType := ASDU_TYPEUNDEF_2,
                                             nObj := 0,
                                             bSQ := FALSE,
                                             bT := FALSE,
                                             bPN := FALSE,
                                             eCOT := 0,
                                             asduAddr := 0,
                                             eClass := eIEC870_Class_None );
  info : ST_IEC870_5_102AOInfoObj := ( rcdAddr := 0,
                                       n := 0,
                                       stream := ( length := 0 ) );
END_STRUCT
END_TYPE
    
```

**head:** Header (reserviert).

**ident:** Identifikationsfelder der Dateneinheit [► 546] (ASDU).

**info:** IEC 60870-5-102 Informationsobjektbeschreibung [► 545].

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.80 ST\_IEC870\_5\_102AOInfoObj

IEC 60870-5-102 Informationsobjektbeschreibung.

Syntax

```

TYPE ST_IEC870_5_102AOInfoObj :
STRUCT
  rcdAddr : BYTE;
    
```

```

    stream : ST_IEC870_5_101Stream;
END_STRUCT
END_TYPE

```

**rcdAddr:** Listenadresse einer Messperiode oder Listenadresse [► 295] einer Einzelmeldung.

**stream:** Informationselemente oder Informationsobjekte [► 485] (Byte-Array-Puffer).

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.3.81 ST\_IEC870\_5\_102DataUnit\_Ident

Identifikationsfeld der Dateneinheit (ASDU).

### Syntax

```

TYPE ST_IEC870_5_102DataUnit_Ident:
STRUCT
    eType : E_IEC870_5_102TypeID;
    bSQ : BOOL;
    nObj : BYTE(0..127);
    bT : BOOL;
    bPN : BOOL;
    eCOT : E_IEC870_5_102COTType;
    asduAddr: DWORD;
    eClass : E_IEC870_5_101ClassType;
END_STRUCT
END_TYPE

```

**eType:** IEC 60870-5-102 ASDU-Typkennung [► 503] := <0..255>.

**bSQ:** Sequenz-Flag := <FALSE = Einzeln, TRUE = Sequenz>.

**nObj:** Anzahl der Informationsobjekte oder Informationselemente. Zulässiger Bereich: <0..127>.

**bT:** Test-Bit (wird kaum benutzt).

**bPN:** Positive/Negative Bestätigung := <FALSE = Positive, TRUE = Negative>.

**eCOT:** IEC 60870-5-102 Übertragungsursache [► 512] (COT). Zulässiger Bereich: <0..63>.

**asduAddr:** Gemeinsame ASDU-Adresse (CAA).

**eClass:** Prioritätsklasse [► 509].

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.82 ST\_IEC870\_5\_103TBuffer

Diese Datenstruktur (TX/RX-Datenpuffer) wird beim Datenaustausch über das IEC 60870-5-103 Low-Level-Interface benutzt.

#### Syntax

```

TYPE ST_IEC870_5_103TBuffer :
STRUCT
  eDbg      : E_IEC870_5_101FifoDbgFlags :=eIEC870_FIFO_DBG_OFF;
  asduSize: BYTE := 249;
  mode     : DWORD := 0;
  dataLink: ST_IEC870_5_101DataLink;
END_STRUCT
END_TYPE
    
```

**eDbg:** [Debug-Ausgabe-Parameter \[► 514\]](#).

**asduSize:** Maximale Bytelänge der ASDU.

**mode:** Reserviert, wird zur Zeit nicht benutzt. Dieser Wert sollte Null sein.

**dataLink:** Auf die Elemente dieser Datenstruktur sollte nicht direkt, sondern nur mit einer Instanz des [FB IEC870\\_5\\_103TBufferCtrl \[► 406\]](#)-Funktionsbausteins zugegriffen werden.

Der TX/RX-Datenpuffer verwendet intern zwei Sendefifos und einen Empfangs-Fifo:

1. Class 1 Sendefifo mit (hochprioren) Daten;
2. Class 2 Sendefifo mit (niederprioren) Daten;
3. Receive-Fifo (für Class 1 und Class 2 Daten);

Die unteren Transportfunktionen der Bibliothek leeren zuerst den Class 1-Fifo und dann den Class 2-Fifo. Die Class 2 Daten werden nur dann versendet wenn der Class 1-Fifo keine zu versendenden Daten enthält.

Jeder der internen Fifos hat eine feste Größe von 200000 Bytes. Erfahrungsgemäß können in jedem Fifo ca. 2000 ASDUs mit einem Informations-Element (Objekt) mit der maximalen Größe oder ca. 600 ASDUs mit einer Sequence von 100 Informations-Elementen (Objekten) abgelegt werden.

Wenn eine größere Anzahl der zu versendenden oder zu empfangenen Frames zwischengespeichert werden soll (z.B. >2000), so können diese in externen, vom SPS-Programmierer festgelegten Puffern/Fifos zwischengehalten werden. Die SPS-Applikation kann dann zur Laufzeit die TwinCAT-Sendefifos mit den eigenen Fifo-Einträgen nachfüllen oder bei vielen empfangenen Frames den TwinCAT-Receive-Fifo leeren. Eine andere Möglichkeit ist z.B. zwei Puffer zu benutzen und diese abwechselnd zu füllen/lesen und an den Kommunikationsbaustein übergeben.

#### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.83 ST\_IEC870\_5\_103AOGen

Variablen von diesem Typ repräsentieren ein ASDU-Objekt.

#### Syntax

```

TYPE ST_IEC870_5_103AOGen:
STRUCT
  head : ST_IEC870_5_101FifoHead :=( source := ( link := 0, addr := 0 ),
    
```

```

        target := ( link := 0, addr := 0 ),
        ctrl   := 0 );

    ident    : ST_IEC870_5_103DataUnit_Ident := ( e := 0,
        nObj   := 0,
        bSQ    := FALSE,
        eCOT   := 0,
        eClass := eIEC870_Class_None,
        asduAddr:= 0 );
    info     : ST_IEC870_5_103AOInfoObj := ( fc := 0,
        n      := 0,
        stream := ( length := 0 ) );
END_STRUCT
END_TYPE

```

**head:** Header (reserviert).

**ident:** Identifikationsfelder der Dateneinheit [► 549] (ASDU).

**info:** Informationsobjekt-/Informationselement-Datenfeld [► 548].

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.3.84 ST\_IEC870\_5\_103AOInfoObj

IEC 60870-5-103 Informationsobjektbeschreibung.

### Syntax

```

TYPE ST_IEC870_5_103AOInfoObj :
STRUCT
    fc      : BYTE;
    n       : BYTE;
    stream  : ST_IEC870_5_101Stream;
END_STRUCT
END_TYPE

```

**fc:** Funktionsnummer/Funktionscode [► 325].

**n:** Informationsnummer [► 325].

**stream :** Information element [► 485] of max. length.

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.85 ST\_IEC870\_5\_103DataUnit\_Ident

Identifikationsfeld der Dateneinheit (ASDU).

#### Syntax

```

TYPE ST_IEC870_5_103DataUnit_Ident:
STRUCT
  eType      : INT(0..255);
  bSQ       : BOOL;
  nObj      : BYTE(0..127);
  eCOT      : INT(0..255);
  asduAddr  : BYTE;
  eClass    : E_IEC870_5_101ClassType;
END_STRUCT
END_TYPE
    
```

**eType:** IEC 60870-5-103 ASDU-Typkennung in Steuerungsrichtung [▶ 505] oder ASDU-Typkennung in Überwachungsrichtung [▶ 505] := <0..255>.

**bSQ:** Sequence-Flag := <FALSE = Single, TRUE = Sequenz>.

**nObj:** Anzahl der Informationsobjekte oder Informationselemente := <0..127>.

**eCOT:** IEC 60870-5-103 Übertragungsursache in Steuerungsrichtung [▶ 514] oder Übertragungsursache in Überwachungsrichtung [▶ 513] := <0..255>.

**asduAddr:** Gemeinsame ASDU-Adresse (CAA, ein Byte).

**eClass:** Prioritätsklasse [▶ 509].

#### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.86 ST\_IEC870\_5\_104ProtocolParams

#### Syntax

```

TYPE ST_IEC870_5_104ProtocolParams :
STRUCT
  t0          : TIME := T#30s;
  t1          : TIME := T#15s;
  t2          : TIME := T#10s;
  t3          : TIME := T#20s;
  iK          : INT := 12;
  iW          : INT := 8;

  bSFrameACK  : BOOL := TRUE;
  bTESTFRAct  : BOOL := TRUE;
  bDTControlled: BOOL := TRUE;
  bControlDT  : BOOL := FALSE;

  bSTARTDTCon : BOOL := TRUE;
  bSTOPDTCon  : BOOL := TRUE;
  bTESTFRCon  : BOOL := TRUE;

  eAcceptMode : E_SocketAcceptMode := eACCEPT_ALL;
  sRemoteHost : STRING(15) := '';
  nRemotePort : UDINT := 0;

  APDULength  : BYTE(MIN_IEC870_5_104_APDULEN..MAX_IEC870_5_104_APDULEN) := MAX_IEC870_5_104_APDU
    
```

```

LEN;
  bThrottleMode: BOOL := FALSE;
  bPackFrames  : BOOL := FALSE;
  bRetainBuffer: BOOL := FALSE;
  bCOTFilter   : BOOL := TRUE;
  cotFilter    : T_IEC870_5_101COTBits := 2#10001111, 7(16#FF);
  eSwitchMode  : E_IEC870_5_104GrpSwitchMode := eIEC870_SWITCHMODE_OFF;
END_STRUCT
END_TYPE

```

**t0** : Zykluszeit für Verbindungsversuche (nur Leitstation).

**t1** : Max. Timeout-Zeit für eine Antwort auf ein STARTDAct-, STOPDAct- oder TESTFRAct-Frame.

**t2** : Spätestens nach dieser Zeit wird ein S-Frame versendet.

**t3** : Spätestens nach dieser Zeit wird ein Test-Frame versendet.

**iK** : Spätestens nach diesem gesendeten APDUs im I-Format, welches nicht quittiert wurde, wird die Verbindung geschlossen.

**iW** : Spätestens nach dem Empfang von w APDUs im I-Format quittiert der Empfänger den Empfang.

**bSFrameACK** : Sende S-Frames.

**bTESTFRAct** : Sende Test-Frames.

**bDTControlled** : Warte auf STARTDT-, STOPDT-Frame vom Master.

**bControlIDT** : Nur bei der Masterkonfiguration: Sende STARTDT zum Slave.

**bSTARTDTCon** : Sende STARTDT Bestätigung

**bSTOPDTCon** : Sende STOPDT Bestätigung.

**bTESTFRCon** : Sende TESTFR Bestätigung.

**eAcceptMode** : Legt fest ob Verbindungen zu allen Remote-Clients, oder nur zu Clients mit bestimmter Host- und Port-Adresse zugelassen werden sollen. Default: Akzeptiere alle ankommenden Verbindungen. Auf der Seite der Unterstation funktioniert dies nur bei einer logischen Verbindung.

**sRemoteHost** : Host-Adresse des Remote-Clients. Bei eAcceptMode = eACCEPT\_ALL wird dieser Parameter ignoriert.

**nRemotePort** : Port-Adresse des Remote-Clients. Bei eAcceptMode = eACCEPT\_ALL wird dieser Parameter ignoriert.

**APDULength** : Maximale Länge der APDU. Defaultwerte:

Max. length of APDU = 255 bytes - 1 start octet - 1 length octet = 253 octets;

Max. length of ASDU = 253 - 4 control octets = 249 octets;

**bThrottleMode**: Die TCP/IP sockets werden pollend aus der SPS abgefragt. Mit diesem Parameter kann die Anzahl der pollenden Lesezugriffe und damit auch die Systemauslastung reduziert werden, besonders dann wenn nur selten Daten empfangen werden (z.B. Generalabfrage oder Uhrzeitsynchronisationsbefehle).

- FALSE: Nach jedem Lesezugriff folgt der nächste auch wenn keine neuen Daten empfangen wurden (höhere Systemauslastung);
- TRUE: Nach jedem Lesezugriff, der keine neuen Daten liefert wird eine Verzögerung eingefügt. Der nächste Lesezugriff wird danach verzögert ausgeführt (kleinere Systemauslastung). Nach jedem Lesezugriff, der neue Daten liefert, wird der nächste ohne Verzögerung ausgeführt. Die max. Verzögerungszeit <= 2 Sekunden.

**bPackFrames**: Per Default wird mit einem TCP/IP-Send-Aufruf nur eine einzelne APDU verschickt. Wenn Sie diesen Parameter auf TRUE setzen, können Sie die Sende-Performance wesentlich erhöhen und so die Sende-Buffer-Überläufe verringern.

- FALSE: Für eine APDU wird ein TCP/IP-Send-Aufruf benötigt. Ca. alle 3 SPS-Zyklen kann max. eine APDU verschickt werden;

- TRUE: Mehrere APDUs werden zu einem größeren TCP/IP-Block (maximal aber iK-Frames) zusammengefasst und mit einem TCP/IP-Send-Aufruf verschickt;

**bRetainBuffer:** In der Standardeinstellung (FALSE) werden die internen Tc/Rx-Puffer beim Verbindungsabbruch gelöscht. Wenn dieses Flag auf TRUE gesetzt ist dann werden die noch nicht gesendeten ASDU's im internen Sendepuffer nicht gelöscht. Somit ist eine Offline-Pufferung von von ca. 100-200 Messwerten (abhängig von der ASDU-Größe) im RAM-Speicher möglich. Die Station entfernt immer dann die ASDUs aus dem Sendepuffer wenn deren Empfang bestätigt wurde. Bitte beachten Sie, dass die bereits gesendeten aber noch nicht bestätigten ASDUs auch im Puffer verbleiben und nächstes Mal erneut gesendet werden. Möglicherweise empfängt die andere Station dann die Werte zwei Mal. Sie können dieses Verhalten über zwei weitere Parameter: **bCOTFilter** und **cotFilter** konfigurieren.

**bCOTFilter:** Aktiviert/deaktiviert die Filtermaske mit den Übertragungsursachen. Dieser Parameter ist nur dann gültig wenn auch der *bRetainBuffer*-Parameter auf TRUE gesetzt wurde. Bei jedem neuen Verbindungsaufbau wird intern eine SessionID hochgezählt. Diese SessionID wird immer an die empfangenen und gesendeten ASDUs angehängt. Dadurch können die ASDUs die noch nicht gesendet wurden und im Offline-Puffer verbleiben der alten Verbindung zugeordnet werden. Diese ASDUs können dann mit Hilfe der COT-Maske (**cotFilter**) gefiltert und verworfen werden (COT = cause of transfer). Dies ist manchmal notwendig wenn der andere Kommunikationspartner die wiederholten ASDUs nicht akzeptiert.

**cotFilter:** Filtermaske mit Übertragungsursachen (COT = Cause of transfer). Dieser Parameter ist nur dann gültig wenn auch der *bRetainBuffer*-Parameter und *bCOTFilter*-Parameter gesetzt wurde. Die Übertragungsursache der zu sendenden ASDUs wird nur dann geprüft wenn deren SessionID nicht zu der aktuellen SessionID passt (d.h. die ASDUs die aus der vorherigen Verbindung stammen). Jedes Bit entspricht einer Übertragungsursache. Die Übertragungsursache wird nur dann überprüft wenn das entsprechende Bit gesetzt wurde.

Die Übertragungsursachen sind auf folgende Weise in den Bits kodiert:

```

cotFilter[0].7 = eIEC870_COT_UNUSED
cotFilter[0].6 = eIEC870_COT_CYCLIC
cotFilter[0].5 = eIEC870_COT_BACKGROUND
cotFilter[0].4 = eIEC870_COT_SPONTAN
cotFilter[0].3 = eIEC870_COT_INIT
cotFilter[0].2 = eIEC870_COT_REQ
cotFilter[0].1 = eIEC870_COT_ACT
cotFilter[0].0 = eIEC870_COT_ACT_CON
cotFilter[1].7 = eIEC870_COT_DEACT
cotFilter[1].6 = eIEC870_COT_DEACT_CON
cotFilter[1].5 = eIEC870_COT_ACT_TERM
    
```

... usw.

Default: Filtern von COT: eIEC870\_COT\_CYCLIC oder eIEC870\_COT\_BACKGROUND oder eIEC870\_COT\_SPONTAN aktiviert.

**eSwitchMode** : Redundanzgruppe-Umschaltmodus (reserviert).

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.87 ST\_IEC870\_5\_104ExSystemInterface

#### Syntax

```

TYPE ST_IEC870_5_104ExSystemInterface :
STRUCT
    device      : ST_IEC870_5_101DeviceInterface;
    service     : ST_IEC870_5_101SystemServices;
    hSOTable    : T_HAODBTABLE;
END_STRUCT
END_TYPE

```

**device:** Interne Kommunikationsschnittstelle des IEC-Gerätes [[▶ 489](#)].

**service:** IEC-Gerätedienste [[▶ 481](#)].

**hSOTable:** Systemobjekt Datenbank Handle [[▶ 531](#)].

#### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.88 ST\_IEC870\_5\_104ServerConnection

Eine IEC 60870-5-104 Unterstation agiert als TCP/IP Server. Ein TCP/IP Server kann theoretisch Verbindungen zu mehreren Clients aufbauen. Eine Variable von diesem Typ repräsentiert eine Kommunikationsverbindung der IEC 870-5-104 Unterstation zur Leitstation.

Beispiel einer Deklaration für eine dreifache Verbindung:

#### Code

```
connections : ARRAY[0..2] OF ST_IEC870_5_104ServerConnection;
```

Beispiel einer Deklaration für eine einfache Verbindung:

#### Code

```
connections : ARRAY[0..0] OF ST_IEC870_5_104ServerConnection;
```

#### HINWEIS

Die Strukturelemente sollen nicht direkt beschrieben oder verändert werden.

#### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)



### 5.3.89 ST\_IEC870\_5\_104GrpStatus

Status einer IEC870-5-104 Slave-Gruppe.

#### Syntax

```

TYPE ST_IEC870_5_104GrpStatus:
STRUCT
    nConnected : DWORD := 0;
    nSuspended : DWORD := 0;
    nDTStarted : DWORD := 0;
END_STRUCT
END_TYPE
    
```

**nConnected:** Anzahl der hergestellten TCP/IP-Verbindungen (ESTABLISHED).

**nSuspended:** Anzahl der Verbindungen deren Zustand gerade wechselt (CONNECTED->DISCONNECTED oder DISCONNECTED->CONNECTED).

**nDTStarted:** Anzahl der Verbindungen mit aktivem Datentransfer.

#### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

### 5.3.90 ST\_IEC870\_5\_104SystemInterface

#### Syntax

```

TYPE ST_IEC870_5_104SystemInterface :
STRUCT
    device : ST_IEC870_5_101DeviceInterface;
    service : ST_IEC870_5_104SlaveServices;
END_STRUCT
END_TYPE
    
```

**device:** Kommunikationsschnittstelle [▶ 489] des IEC-Gerätes.

**service:** IEC-Gerätedienst.

#### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

## 5.4 Globale Konstanten

### 5.4.1 Group Konfigurationsflags

Mit den Group-Flags werden Applikationsobjekte (Datenpunkte) bestimmten Gruppen zugeordnet. Die Group-Flags für Stationsabfrage und Zählwertübertragung sind auf folgende Weise definiert worden:

- Stationsabfrage: Alle Applikationsobjekte, die der Gruppe 1 bis 16 zugeordnet wurden, gehören automatisch auch zur globalen Gruppe: IEC870\_GRP\_INROGEN;
- Zählwertübertragung: Alle Applikationsobjekte, die der Gruppe 1 bis 4 zugeordnet wurden, gehören automatisch auch zur globalen Gruppe: IEC870\_GRP\_REQCOGEN;

Sie können aber die automatische Zugehörigkeit der Applikationsobjekte zur globalen Gruppe verhindern in dem Sie die entsprechenden Bits während der Konfiguration ausmaskieren.

Beispiele:

```
IEC870_GRP_INRO3 AND NOT IEC870_GRP_INROGEN
```

oder

```
IEC870_GRP_REQCO1 AND NOT IEC870_GRP_REQCOGEN
```

Konstante	Wert	Beschreibung
<b>Stationsabfrage</b>		
IEC870_GRP_INROGEN	16#00000001	Abgefragt durch Generalabfrage (station or global)
IEC870_GRP_INRO1	16#00000003	Abgefragt durch Stationsabfrage der Gruppe 1
IEC870_GRP_INRO2	16#00000005	Abgefragt durch Stationsabfrage der Gruppe 2
IEC870_GRP_INRO3	16#00000009	Abgefragt durch Stationsabfrage der Gruppe 3
IEC870_GRP_INRO4	16#00000011	Abgefragt durch Stationsabfrage der Gruppe 4
IEC870_GRP_INRO5	16#00000021	Abgefragt durch Stationsabfrage der Gruppe 5
IEC870_GRP_INRO6	16#00000041	Abgefragt durch Stationsabfrage der Gruppe 6
IEC870_GRP_INRO7	16#00000081	Abgefragt durch Stationsabfrage der Gruppe 7
IEC870_GRP_INRO8	16#00000101	Abgefragt durch Stationsabfrage der Gruppe 8
IEC870_GRP_INRO9	16#00000201	Abgefragt durch Stationsabfrage der Gruppe 9
IEC870_GRP_INRO10	16#00000401	Abgefragt durch Stationsabfrage der Gruppe 10
IEC870_GRP_INRO11	16#00000801	Abgefragt durch Stationsabfrage der Gruppe 11
IEC870_GRP_INRO12	16#00001001	Abgefragt durch Stationsabfrage der Gruppe 12
IEC870_GRP_INRO13	16#00002001	Abgefragt durch Stationsabfrage der Gruppe 13
IEC870_GRP_INRO14	16#00004001	Abgefragt durch Stationsabfrage der Gruppe 14
IEC870_GRP_INRO15	16#00008001	Abgefragt durch Stationsabfrage der Gruppe 15
IEC870_GRP_INRO16	16#00010001	Abgefragt durch Stationsabfrage der Gruppe 16
<b>Zählwertübertragung</b>		
IEC870_GRP_REQCOGEN	16#00020000	Abgefragt durch allgemeine Zählerabfrage
IEC870_GRP_REQCO1	16#00060000	Abgefragt durch Zählerabfrage der Gruppe 1
IEC870_GRP_REQCO2	16#000A0000	Abgefragt durch Zählerabfrage der Gruppe 2
IEC870_GRP_REQCO3	16#00120000	Abgefragt durch Zählerabfrage der Gruppe 3
IEC870_GRP_REQCO4	16#00220000	Abgefragt durch Zählerabfrage der Gruppe 4
IEC870_GRP_LOCFREEZE	16#00400000	Aktiviert zyklisches lokales Umspeichern der Zählerwerte
IEC870_GRP_LOCRESET	16#00800000	Aktiviert zyklisches lokales Rücksetzen der Zählerwerte
<b>Andere</b>		
IEC870_GRP_IV_OFF	16#01000000	Deaktiviert das Setzen des Qualitätsflags: Invalid (IV)
IEC870_GRP_REVERSE	16#02000000	Objekt wird in Umkehrrichtung verwendet (reserviert)
IEC870_GRP_SELECTCMD	16#04000000	Erzwingt Kommando-Anwählen/-Ausführen für diesen Datenpunkt (reserviert)
not defined	16#08000000	nicht benutzt (reserviert)
IEC870_GRP_USERCMD	16#10000000	Benutzerdefiniertes Kommando (reserviert)
IEC870_GRP_BACKGROUND	16#20000000	Aktiviert Hintergrundabfrage für diesen Datenpunkt
IEC870_GRP_PERCYC	16#40000000	Aktiviert periodischen/zyklischen Datentransfer für diesen Datenpunkt
IEC870_GRP_SPONTOFF	16#80000000	Deaktiviert spontane Datenübertragung für diesen Datenpunkt

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.4.2 Qualitätsflags**

Konstante	Wert	Beschreibung
IECQ_BL_ON	16#0001	Blockiert
IECQ_BL_OFF	16#0002	Nicht blockiert
IECQ_SB_ON	16#0004	Ersetzt
IECQ_SB_OFF	16#0008	Nicht ersetzt
IECQ_NT_ON	16#0010	Nicht aktuell
IECQ_NT_OFF	16#0020	Aktuell
IECQ_IV_ON	16#0040	Ungültig
IECQ_IV_OFF	16#0080	Gültig
IECQ_OV_ON	16#0100	Überlauf
IECQ_OV_OFF	16#0200	Kein Überlauf
IECQ_EI_ON	16#0400	Abgelaufene Zeit ist ungültig
IECQ_EI_OFF	16#0800	Abgelaufene Zeit ist gültig
IECQ_CY_ON	16#1000	Übertrag
IECQ_CY_OFF	16#2000	Kein Übertrag
IECQ_CA_ON	16#4000	Zähler wurde eingestellt
IECQ_CA_OFF	16#8000	Zähler wurde nicht eingestellt

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.4.3 IEC 60870-5-10x Fehlercodes**

Fehlercode (Hex)	Fehlercode (Dez)	Symbolische Konstante	Beschreibung
0x00008101	33025	IEC870_COMMERR_INVALIDSTARTBYTE	Invalid frame start character <> 0x68
0x00008102	33026	IEC870_COMMERR_RXBUFFERFLOW	Receive buffer overflow
0x00008103	33027	IEC870_COMMERR_TXBUFFERFLOW	Send buffer overflow
0x00008104	33028	IEC870_COMMERR_INVALIDUFFMT	Invalid U-Frame format, more than one function (STARTDT, STOPDT, TESTFR) activated
0x00008105	33029	IEC870_COMMERR_INVALIDSFFMT	Invalid S-Frame format, invalid length parameter
0x00008106	33030	IEC870_COMMERR_T1RESPONSE	t1 (response timeout) expired
0x00008107	33031	IEC870_COMMERR_SENDSEQ	Send sequence error
0x00008108	33032	IEC870_COMMERR_KOVERTFLOW	k reached
0x00008109	33033	IEC870_COMMERR_FATALERR	Fatal internal error
0x0000810A	33034	IEC870_COMMERR_INVALIDSTATE	Device is in invalid state (disconnected?)
0x0000810B	33035	IEC870_COMMERR_INVALIDSIZE	Invalid parameter size
0x0000810C	33036	IEC870_COMMERR_INVALIDVALUE	Invalid parameter value
0x0000810D	33037	IEC870_COMMERR_INVALIDTYPE	Invalid asdu (object) type
0x0000810F	33039	IEC870_COMMERR_TIMEOUT	Communication timeout
0x00008110	33040	IEC870_COMMERR_LENGTH1	Invalid length field value
0x00008111	33041	IEC870_COMMERR_LENGTH2	Length field and length field copy differs
0x00008112	33042	IEC870_COMMERR_STARTCHAR2	Invalid second start character
0x00008113	33043	IEC870_COMMERR_CHECKSUM	Invalid checksum
0x00008114	33044	IEC870_COMMERR_ENDCHAR	Invalid end character
0x00008115	33045	IEC870_COMMERR_LINKADDR	Invalid link address size
0x00008116	33046	IEC870_COMMERR_SRVFUNCTION	Invalid link service function code
0x00008117	33047	IEC870_COMMERR_FRAMETYPE	Invalid frame type
0x00008118	33048	IEC870_COMMERR_UNSUPPORTEDMODE	Unsupported communication mode (balanced mode)
0x00008119	33049	IEC870_COMMERR_T2KOVERFLOW	k reached and t2 (response timeout) expired
0x0000811A	33050	IEC870_COMMERR_INVALIDCONFIG	Invalid object configuration/initialization
0x0000811B	33051	IEC870_COMMERR_UNKNOWNTYPE	Unknown asdu type
0x0000811C	33052	IEC870_COMMERR_UNKNOWNCOT	Unknown cause of transfer

Fehlercode (Hex)	Fehlercode (Dez)	Symbolische Konstante	Beschreibung
0x0000811D	33053	IEC870_COMMERR_UNKNOWN ASDUADDR	Unknown asdu address
0x0000811E	33054	IEC870_COMMERR_UNKNOWN OBJADDR	Unknown object address
0x0000811F	33055	IEC870_COMMERR_NEGACTCON	Negative activation confirmation
0x00008120	33056	IEC870_COMMERR_NEGACTTERM	Negative activation termination
0x00008121	33057	IEC870_COMMERR_NEGDEACTCON	Negative deactivation confirmation
0x00008122	33058	IEC870_COMMERR_BUSY	Allready in busy state
0x00008123	33059	IEC870_COMMERR_AODBOVERFLOW	Application object database overflow
0x00008124	33060	IEC870_COMMERR_AODBNOTFOUND	Application object not in database
0x00008125	33061	IEC870_COMMERR_ACTCONTIMEOUT	Activation confirmation timeout error
0x00008126	33062	IEC870_COMMERR_ACTTERMIMEOUT	Activation termination timeout error
0x00008127	33063	IEC870_COMMERR_DEACTCONTIMEOUT	Deactivation termination timeout error
0x00008128	33064	IEC870_COMMERR_SELEXECTIMEOUT	Command select/execute timeout error
0x00008129	33065	IEC870_COMMERR_READRESTIMEOUT	Read command response timeout error
0x00008130	33072	IEC870_COMMERR_LIBNOTCOMPAT	Product libraries are incompatible
0x00008131	33073	IEC870_COMMERR_DIR	Invalid DIR bit value
0x00008132	33074	IEC870_COMMERR_PRM	Invalid PRM bit value
0x00008133	33075	IEC870_COMMERR_FCV	Invalid FCV bit value
0x00008134	33076	IEC870_COMMERR_SCANTIMEOUT	Station scan cycle timeout error

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.4.4 Parameterliste**

Konstante	Wert	Beschreibung
IEC870_MAX_RXTX_FIFO_SIZE	200000	Maximale Tx/Rx-Fifogröße (in Byte)

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4018.26 TF6500 v3.0.5.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)

**5.4.5 Bibliotheksversion**

Alle Bibliotheken haben eine spezifische Version. Diese Versionsinformation wird in der SPS-Bibliotheksrepository angezeigt. Mit der Funktion F\_CmpLibVersion (definiert in der Tc2\_System-Bibliothek) kann die vorhandene Versionsinformation mit der benötigten verglichen werden. Eine globale Konstante in der Bibliothek beinhaltet die Versionsinformation:

**Global\_Version**

```
VAR_GLOBAL CONSTANT
    stLibVersion_Tc2_IEC60870_5_10x : ST_LibVersion;
END_VAR
```



Die bekannten Möglichkeiten aus TwinCAT 2.x um die Versionsinformation abzufragen sind veraltet.

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken (Kategoriegruppe)
TwinCAT v3.1.4012.0	PC oder CX (x86, x64, ARM)	Tc2_IEC60870_5_10x (Communication->IEC60870) Tc2_SerialCom (Communication->Serial) Tc2_Tcplp (Communication->Tcplp) Tc2_Uilities (System)



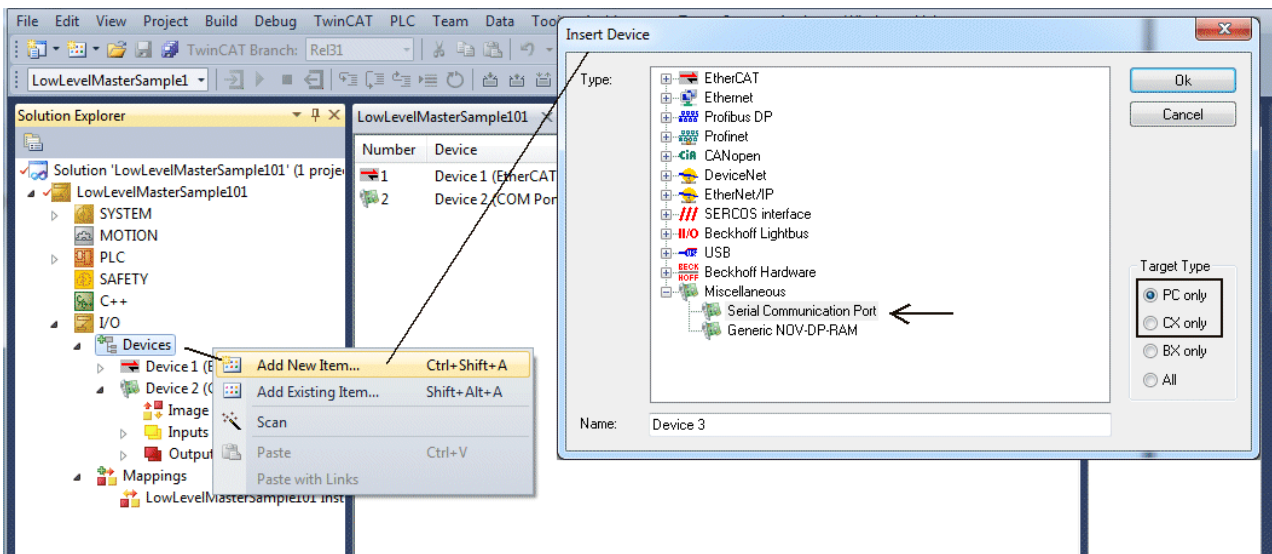
## 6 Beispiele

### 6.1 Konfiguration der seriellen Schnittstellen

- [Standard PC COMx-Schnittstelle](#) [▶ 561]
- [Serielle Busklemmen KL6xxx](#) [▶ 562]
- [Serielle Busklemmen EL6xxx](#) [▶ 563]

#### 1. Standard PC COMx-Schnittstellen

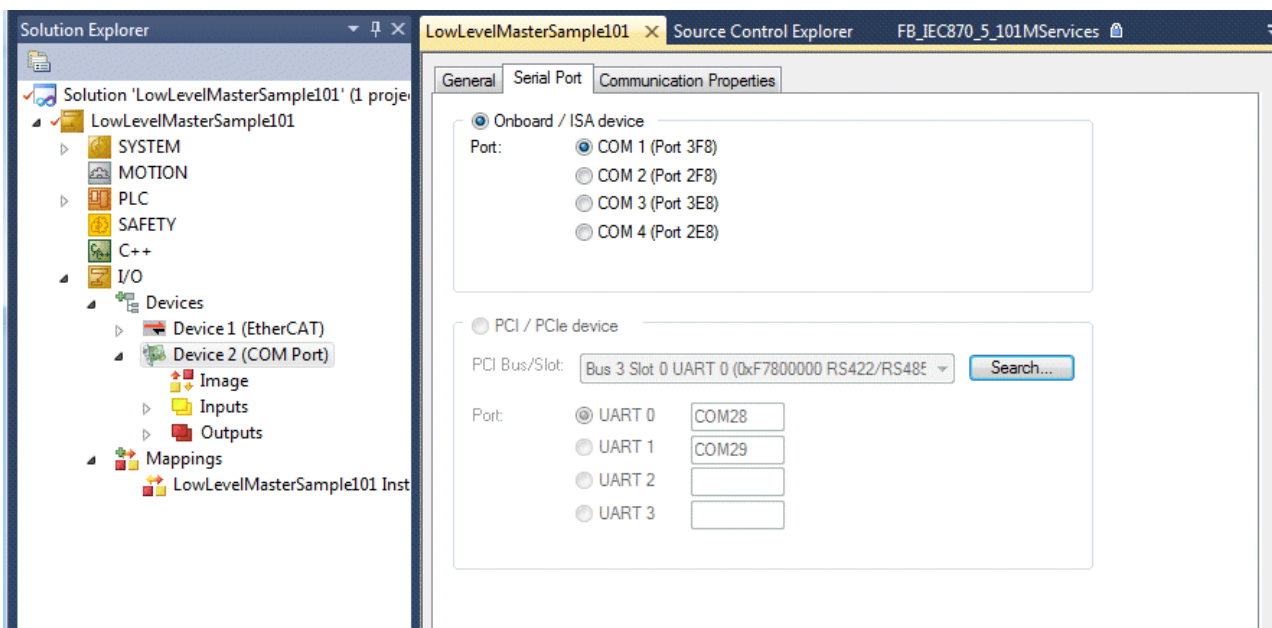
Die serielle Schnittstelle kann nur in PC/CX-Systemen eingefügt werden. Mit der rechten Maustaste den Knoten "E/A->Geräte" anklicken. Im Kontextmenü "Neues Element hinzufügen..." auswählen. Unter "Verschiedenes" die "Serielle Schnittstelle" auswählen.



Anschließend nachfolgende Einstellungen vornehmen.

#### 1.1. Karteireiter Serial Port

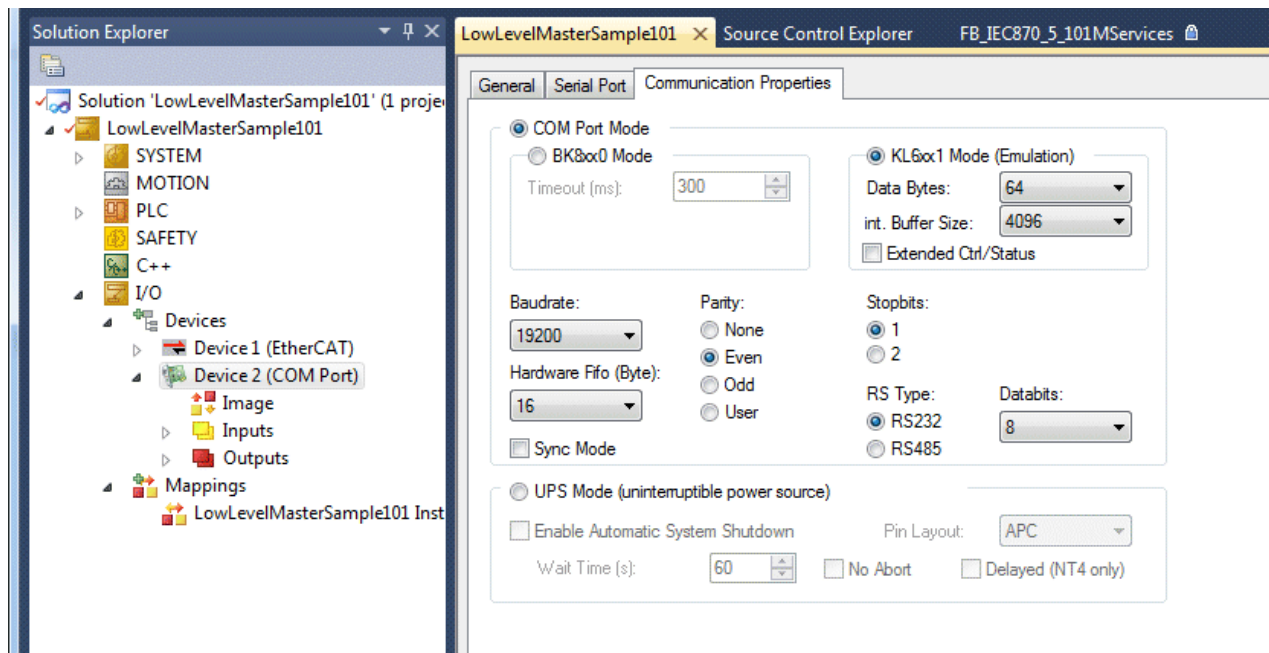
Serielle Schnittstelle auswählen (integrierte COM1..COM4 oder wenn vorhanden eine der PCI/PCIe).



## 1.2. Karteireiter Eigenschaften

Die Option KL6xxx1 Mode (Emulation) anwählen und dann die Kommunikationsparameter einstellen:

Baudrate, hier 19200 Baud, 8 Datenbits, Parity=even, 1 Stopbit



## 2. Serielle Busklemmen KL6xxx

Die Schnittstelle wird im TwinCAT SPS Projekt durch die Instanz des FB\_IEC870\_SerialLineCtrl-Funktionsbausteins konfiguriert. Die Kommunikationsparameter wie Baudrate, Parity usw. sind an diesem Baustein einzustellen.

```
(* High speed task. Serial background communication. *)
PROGRAM P_SerialComm_HighSpeed
VAR
  serial_In      AT%I*   : PcComInData;
  serial_Out     AT%Q*   : PcComOutData;
  KL6_In        AT%I*   : KL6inData5B;
  KL6_Out       AT%Q*   : KL6outData5B;
  serial_ExtStatus AT%I* : WORD;
  serial_ExtCtrl AT%Q*   : WORD;

  hSerial       : T HSERIALCTRL; (* Serial connection control handle *)
  fbSerialLineCtrl : FB_IEC870_SerialLineCtrl := ( Mode := SERIALLINEMODE_KL6_5B_STANDARD (*SERIALLINEMODE_PC_COM_PORT*),
    Baudrate := 19200,
    NoDatabits := 8,
    Parity := PARITY_EVEN,
    Stopbits := 1,
    Handshake := HANDSHAKE_NONE,
    ContinuousMode := FALSE );
```

### 2.1 RS485 Betriebsmodus

Bei der RS485 Betriebsart (z.B. KL6041) werden die Daten in halbduplex Übertragung ausgetauscht. Im RS485 Betriebsmodus werden die Send- und Empfangsleitungen miteinander verbunden. Dadurch empfängt die Klemme nicht nur die Daten anderer Teilnehmer, sondern auch ihre eigenen Sendedaten. Diese Daten können die Kommunikation stören. Für den Halbduplex-Modus muss der *Handshake*-Eingang des FB\_IEC870\_SerialLineCtrl-Funktionsbausteins auf den Wert: RS485\_HALFDUPLEX gesetzt werden.

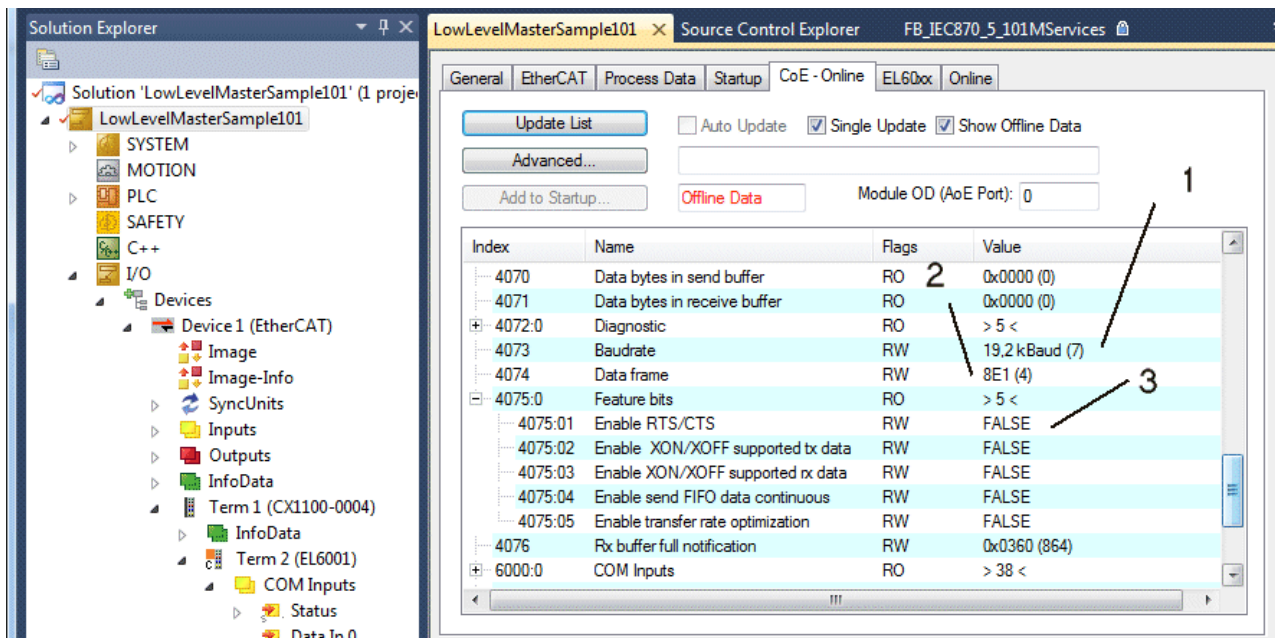
### 3. Serielle Busklemmen EL6xxx

Die serielle Schnittstelle wird im TwinCAT System Manager Karteireiter CoE-Online (CoE=CanOpen over EtherCAT) eingestellt. Der Karteireiter steht nur bei Online-Zugriff auf die Klemme zur Verfügung, d.h., ist der System Manger nicht mit der Hardware verbunden, können keine Einstellungen verändert werden. Die Einstellungen werden in der Klemme Spannungsausfallsicher abgespeichert. Nach einem Scannen der Klemmen müssen diese Daten nicht neu eingegeben werden. Bei einem Tausch der Klemme werden die Daten aus der neuen Klemme aktiv und müssen demzufolge wieder angepasst werden. Um das zu verhindern, müssen im Karteireiter *Startup* die geänderten Daten eingetragen werden. Beim Systemstart werden die Daten aus dem Reiter *Startup* in die Klemme übertragen. Ein Tausch der Klemme ist somit Problemlos möglich. Nach einem Scannen der Klemmen müssen die Daten im Reiter *Startup* neu eingegeben werden.

#### 3.1. Karteireiter CoE

- (1) Baudrate, hier 19200 Baud
- (2) Date frame, 8E1 entspricht 8 Datenbits, Parity=even, 1 Stoppbit
- (3) Feature bits -> Enable RTS/CTS = FALSE bei EL6001

Mit einem Doppelklick auf die entsprechende Zeile wird ein Menü geöffnet, in dem die Einstellungen geändert werden können.



#### 3.1.1 RS485 Betriebsmodus

Bei der RS485 Betriebsart werden die Daten in halbduplex Übertragung ausgetauscht. Im RS485 Betriebsmodus werden die Sende- und Empfangsleitungen miteinander verbunden. Dadurch empfängt die Klemme nicht nur die Daten anderer Teilnehmer, sondern auch ihre eigenen Sendedaten. Diese Daten können die Kommunikation stören. Mit der Option "Enable half duplex" im "COM Settings"-Objekt kann dies unterdrückt werden (Index 8000:06).

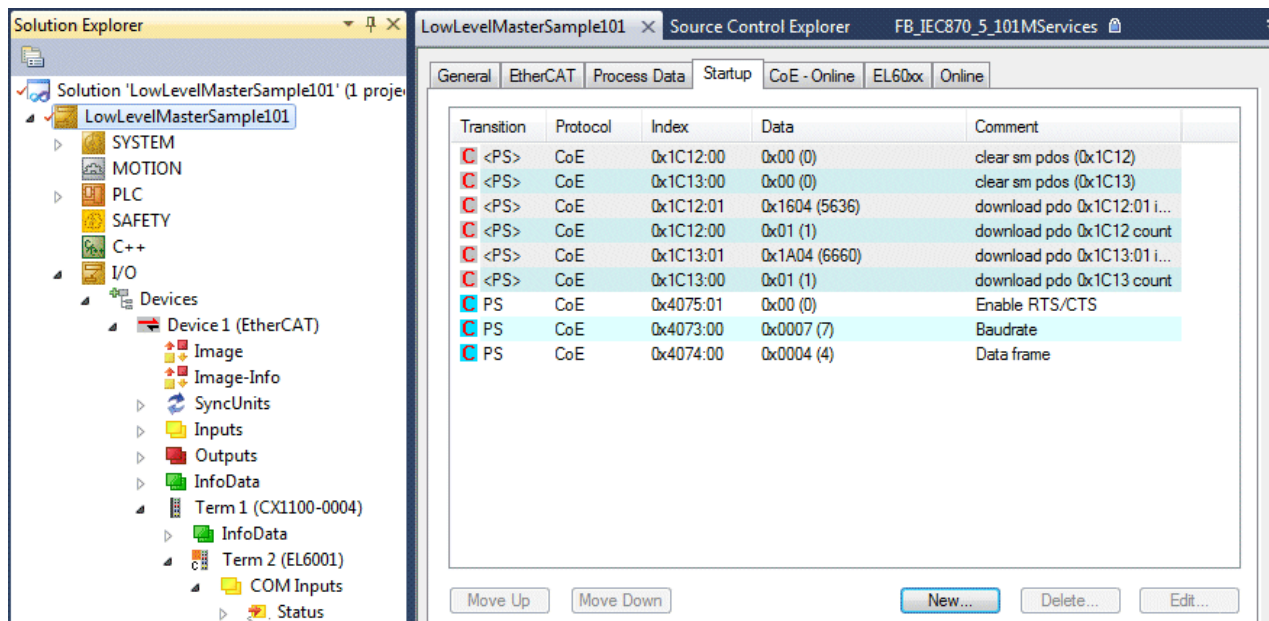
#### 3.2. Karteireiter Startup

Alle vom Standard abweichenden Einstellungen müssen in die Startup Liste eingetragen werden. Beim Systemstart werden die Daten aus dem Reiter *Startup* in die Klemme übertragen. Ein Tausch der Klemme ist somit Problemlos möglich. Nach einem Scannen der Klemmen müssen die Daten im Reiter *Startup* neu eingegeben werden.

Baudrate, hier 19200 Baud

Date frame, 8E1 entspricht 8 Datenbits, Parity=even, 1 Stoppbit

Feature bits -> Enable RTS/CTS = *FALSE* bei EL6001



Transition	Protocol	Index	Data	Comment
<PS>	CoE	0x1C12:00	0x00 (0)	clear sm pdos (0x1C12)
<PS>	CoE	0x1C13:00	0x00 (0)	clear sm pdos (0x1C13)
<PS>	CoE	0x1C12:01	0x1604 (5636)	download pdo 0x1C12:01 i...
<PS>	CoE	0x1C12:00	0x01 (1)	download pdo 0x1C12 count
<PS>	CoE	0x1C13:01	0x1A04 (6660)	download pdo 0x1C13:01 i...
<PS>	CoE	0x1C13:00	0x01 (1)	download pdo 0x1C13 count
PS	CoE	0x4075:01	0x00 (0)	Enable RTS/CTS
PS	CoE	0x4073:00	0x0007 (7)	Baudrate
PS	CoE	0x4074:00	0x0004 (4)	Data frame

## 6.2 IEC 60870-5-101

### 6.2.1 Zentralstation

#### 6.2.1.1 High-Level Interface

##### 6.2.1.1.1 Quick start

Einfaches Projekt mit kompletten Quellen finden Sie hier: [IEC 60870-5-101 Zentralstation-High-Level-Beispiel](#) [▶ 565].

Kompatibilitätsliste finden Sie hier: [Interoperability check list](#) [▶ 14]

Übersicht der Fehlercodes finden Sie hier: [Fehlercodes](#) [▶ 685]

#### Kurzanleitung

##### Applikationsobjekt-Datenbank

Die Applikationsobjekt-Datenbank der Zentralstation muss mit der Funktion: [F\\_iecCreateTableHnd](#) [▶ 452] als Hash-Tabelle konfiguriert werden. Die einzelnen Arrayelemente werden dabei als Hash-Tabelle untereinander verlinkt. Dies ermöglicht u. a. einen schnelleren Zugriff auf die einzelnen Datenpunkte, bringt aber auch einige Nachteile mit sich, die beachtet werden müssen:

- Die Größe der Applikationsdatenbank (Arraygröße) darf nicht zur Laufzeit (z.B. durch Online-Change) verändert werden. Die Zentralstation stoppt sofort die Ausführung und meldet einen Fehler. Der Grund: Die Verlinkung der Hash-Tabelle passt nicht mehr. Bei Programmänderungen laden Sie am besten das komplette Projekt ins Laufzeitsystem.
- Auf die einzelnen Array-Elemente darf nicht per Index sondern nur mit Hilfe der speziellen Funktionen zugegriffen werden (z.B.: [F\\_iecAddTableEntry](#) [▶ 453] usw. ).
- Bei einem indizierten Zugriff auf die Tabellenelemente dürfen die internen Konfigurationsparameter nicht beschrieben oder verändert werden. Bei einer Änderung des Typs, der ASDU-Adresse oder der Objektadresse kann der Datenpunkt nicht mehr gefunden werden. Ein Datenpunkt, der umkonfiguriert werden soll wird aus der Tabelle durch den Funktionsaufruf [F\\_iecRemoveTableEntry](#) [▶ 460] zuerst entfernt. Danach kann der neue Datenpunkt hinzugefügt werden.

Bei einer Implementierung als lineare Tabelle müsste die Zentralstation bei jeder empfangenen ASDU (Dateneinheit) das komplette Array nach dem passenden Element durchsuchen. Dies würde bei vielen Datenpunkten sehr lange Ausführungszeiten generieren.

### Protokollparameter

Die meisten Protokollparameter sind bereits mit Defaultwerten vorbelegt und müssen nicht explizit gesetzt werden.

### Systemparameter

Die Systemparameter sind ebenfalls mit Defaultwerten vorbelegt. Während der Inbetriebnahme ist es nützlich die Debug-Ausgabe zu aktivieren (*dbgMode*) um mögliche Fehler lokalisieren zu können.

### Parameter für die zyklische Datenerfassung

Folgende Parameter sind bereits mit Defaultwerten vorkonfiguriert:

- Initialisierungssequenz (besteht aus einem Testbefehl, Uhrzeitsynchronisation, Stationsabfrage und Zählerabfrage);
- Zyklische Befehle:
  - Testbefehl alle 60s;
  - Uhrzeitsynchronisation alle 60s;
  - Stationsabfrage der Gruppe: Allgemein alle 60s;
  - Zählerabfrage der Gruppe: Allgemein alle 60s;

#### 6.2.1.1.1 Download: Zentralstation Minimal (master, unbalanced mode)

Einfache Zentralstation-Applikation die das TwinCAT IEC 60870-5-101 High-Level Interface verwendet.

Download TwinCAT XAE Project (\*.zip): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/984601099.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/984601099.zip)

Im Beispielprojekt sind folgende Stationsparameter für die Zentralstation eingestellt:

- Standard PC interface: **COM1**.
- Baudrate: **19200 Baud**.
- Link address: **220**.
- Link address size: **2 octets**.
- Cause of transfer size: **2 octets (inclusive originator address)**.
- Originator address: **1**.
- Common ASDU address: **7**.
- Common ASDU address size: **2 octets**.
- Information object address size: **3 octets**.
- Link mode: **Unbalanced**.

Datenpunktconfiguration (sie sollte gleich sein mit der Datenpunktconfiguration der Unterstationen):

- Single point information: M\_SP\_NA\_1 (IOA := 100).
- Single command: C\_SC\_NA\_1 (IOA := 10).

Initialisierungssequenz, die nach dem Verbindungsaufbau einmalig/automatisch ausgeführt wird:

- Testbefehl (C\_TS\_NA\_1).
- Uhrzeitsynchronisationsbefehl (C\_CS\_NA\_1).
- Generalabfragebefehl (C\_IC\_NA\_1) der Gruppe: Allgemein (eIEC870\_QOI\_INROGEN).
- Zählerabfragebefehl (C\_CI\_NA\_1) der Gruppe: Allgemein (eIEC870\_RQT\_REQCOGEN).  
Zählerabfrage Mode C

Beispielimplementierung folgender Dienste:

- Testbefehl (C\_TS\_NA\_1) alle 60 Sekunden.
- Uhrzeitsynchronisationsbefehl (C\_CS\_NA\_1) alle 60 Sekunden.
- Generalabfragebefehl (C\_IC\_NA\_1) der Gruppe: Allgemein (eIEC870\_QOI\_INROGEN) alle 60 Sekunden.
- Zählerabfragebefehl (C\_CI\_NA\_1) der Gruppe: Allgemein (eIEC870\_RQT\_REQCOGEN) alle 60 Sekunden. Zählerabfrage Mode C.
- Bei Wertänderung an der **bSCS** -Variablen wird der Einzelbefehl (C\_SC\_NA\_1) gesendet (direkte Kommandoausführung)
- Beim Empfang spontaner Daten vom Typ (M\_SP\_NA\_1, IOA := 100) wird der ON/OFF-Wert in die SPS-Variable **bSPI** gemappt..

Folgende Tabelle beschreibt kurz die Hauptkomponenten des Projekts:

Komponente	Typ	Beschreibung
P_MAIN_LowSpeed	PRG	Dieses Programmteil implementiert das Fernwirkprotokoll (langsame Task).
P_SerialComm_HighSpeed	PRG	Dieses Programmteil implementiert die TwinCAT Hintergrundkommunikation über die seriellen Schnittstellen des PCs (schnelle Task).
fbClient	FB_IEC870_5_101Master (FB)	Implementiert die Zentralstation. Eine Instanz dieses Bausteins kann nur mit einer Unterstation kommunizieren. Für jede weitere Unterstation mit der kommuniziert werden soll wird eine weitere Instanz dieses Funktionsbausteins benötigt.
bSPI	BOOL	Der empfangene ON/OFF-Wert wird in diese SPS-Variable direkt an die lokierte Merkeradresse: <b>AT%MX0.0</b> gemappt.
bSCS	BOOL	Der zu sendende ON/OFF-Wert wird direkt von der an der Merkeradresse <b>AT%MX1.5</b> dieser lokierten SPS-Variablen gemappt.
AODB	ARRAY of ST_IEC870_5_101AODBEntry (STRUCT)	Applikationsobjektdatenbank. Es ist eine Array-Variable. Für jeden Datenpunkt wird ein Arrayelement benötigt.
hTable	T_HAODBTable (STRUCT)	Applikationsobjektdatenbank-Handle. Mit Hilfe dieser Variablen wird auf die Applikationsobjektdatenbank zugegriffen und nie direkt auf die einzelnen Arrayelemente.
F_iecCreateTableHnd	FUNCTION	Im Initialisierungsschritt werden alle Arrayelemente von AODB mit diesem Funktionsaufruf einmalig initialisiert. Beim Erfolg liefert dieser Funktionsaufruf das Handle.
F_iecAddTableEntry	FUNCTION	Im Initialisierungsschritt werden die einzelnen Datenpunkte der Station mit diesem Funktionsaufruf einmalig konfiguriert.

### 6.2.1.1.2 Tutorial

Die Einführung ist eine Anleitung wie Sie in der TwinCAT SPS eine IEC60870-5-101 Zentralstation (master) implementieren und konfigurieren können. Das Beispiel kann [hier](#) [[566](#)] heruntergeladen werden.

#### 6.2.1.1.2.1 Download: Zentralstation Tutorial (master, unbalanced mode)

Eine komplexere Zentralstation-Applikation mit unterschiedlichen Datenpunkten in beide Richtungen und einer Verbindung zur Unterstation. Die Beispielapplikation verwendet das TwinCAT IEC 60870-5-101 High-Level Interface.

Download TwinCAT XAE Project (\*.zip): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/984604043.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/984604043.zip)

Im Beispielprojekt sind folgende Stationsparameter für die Zentralstation eingestellt:

- Standard PC interface: **COM1**.
- Baudrate: **19200 Baud** .
- Link address: **220**.
- Link address size: **2 octets**.
- Cause of transfer size: **2 octets (inclusive originator address)**.
- Originator address: **1**.
- Common ASDU address: **7**.
- Common ASDU address size: **2 octets**.
- Information object address size: **3 octets**.
- Link mode: **Unbalanced**.
- Class 1 poll cycle: **100ms**.
- Class 2 poll cycle: **100ms**.

Datenpunktconfiguration (sie sollte gleich sein mit der Datenpunktconfiguration der Unterstation):

- Single point information: M\_SP\_NA\_1 (IOA := 100), M\_SP\_TA\_1 (IOA := 101), M\_SP\_TB\_1 (IOA := 102).
- Double point information: M\_DP\_NA\_1 (IOA := 200), M\_DP\_TA\_1 (IOA := 201), M\_DP\_TB\_1 (IOA := 202).
- Regulating step value: M\_ST\_NA\_1 (IOA := 300), M\_ST\_TA\_1 (IOA := 301), M\_ST\_TB\_1 (IOA := 302).
- Bitstring value: M\_BO\_NA\_1 (IOA := 400), M\_BO\_TA\_1 (IOA := 401), M\_BO\_TB\_1 (IOA := 402).
- Measured value, normalized value: M\_ME\_NA\_1 (IOA := 500), M\_ME\_TA\_1 (IOA := 501), M\_ME\_TD\_1 (IOA := 502).
- Measured value, scaled value: M\_ME\_NB\_1 (IOA := 600), M\_ME\_TB\_1 (IOA := 601), M\_ME\_TD\_1 (IOA := 602).
- Measured value, short floating point value: M\_ME\_NC\_1 (IOA := 700), M\_ME\_TC\_1 (IOA := 701), M\_ME\_TF\_1 (IOA := 702).
- Binary counter value: M\_IT\_NA\_1 (IOA := 800), M\_IT\_TA\_1 (IOA := 801), M\_IT\_TB\_1 (IOA := 802).
- Single command: C\_SC\_NA\_1 (IOA := 10), C\_SC\_TA\_1 (IOA := 11), C\_SC\_TB\_1 (IOA := 12).
- Double command: C\_DC\_NA\_1 (IOA := 20), C\_DC\_TA\_1 (IOA := 21), C\_DC\_TB\_1 (IOA := 22).
- Bitstring command: C\_BO\_NA\_1 (IOA := 40), C\_BO\_TA\_1 (IOA := 41), C\_BO\_TB\_1 (IOA := 42).
- Setpoint command, normalized value: C\_SE\_NA\_1 (IOA := 50), C\_SE\_TA\_1 (IOA := 51), C\_SE\_TB\_1 (IOA := 52).
- Setpoint command, scaled value: C\_SE\_NB\_1 (IOA := 60), C\_SE\_TB\_1 (IOA := 61), C\_SE\_TB\_1 (IOA := 62).
- Setpoint command, short floating point value: C\_SE\_NC\_1 (IOA := 70), C\_SE\_TC\_1 (IOA := 71), C\_SE\_TC\_1 (IOA := 72).

Initialisierungssequenz, die nach dem Verbindungsaufbau einmalig/automatisch ausgeführt wird:

- Kein Testbefehl (C\_TS\_NA\_1, deaktiviert).
- Uhrzeitsynchronisationsbefehl (C\_CS\_NA\_1).
- Generalabfragebefehl (C\_IC\_NA\_1) der Gruppe: Allgemein (eIEC870\_QOI\_INROGEN).
- Kein Zählerabfragebefehl (C\_IC\_NA\_1) der Gruppe: Allgemein (eIEC870\_RQT\_REQCOGEN, deaktiviert). Zählerabfrage Mode A (Zähler werden lokal in der Unterstation umgespeichert und als spontane Daten an die Zentralstation gesendet).

Beispielimplementierung folgender Dienste:

- Testbefehl (C\_TS\_NA\_1) alle 60 Sekunden.

- Uhrzeitsynchronisationsbefehl (C\_CS\_NA\_1) alle 60 Sekunden.
- Generalabfragebefehl (C\_IC\_NA\_1) der Gruppe: Allgemein (eIEC870\_QOI\_INROGEN) alle 60 Sekunden.
- Kein Zählerabfragebefehl (C\_IC\_NA\_1) der Gruppe: Allgemein (eIEC870\_RQT\_REQCOGEN, deaktiviert). Zählerabfrage Mode A (Zähler werden lokal in der Unterstation umgespeichert und als spontane Daten an die Zentralstation gesendet).
- Simulation und Übertragung der Befehle alle 10 Sekunden wenn die Variable **bSendCmd** auf TRUE gesetzt wurde (siehe obere Datenpunktkonfiguration).
- Empfang spontaner Daten.
- Empfang von M\_EI\_NA\_1 (Ende der Initialisierung).

Folgende Tabelle beschreibt kurz die Hauptkomponenten des Projekts:

Komponente	Typ	Beschreibung
P_MAIN_LowSpeed	PRG	Dieses Programmteil implementiert das Fernwirkprotokoll (langsame Task).
P_SerialComm_HighSpeed	PRG	Dieses Programmteil implementiert die TwinCAT Hintergrundkommunikation über die seriellen Schnittstellen des PCs (schnelle Task).
fbClient	FB_IEC870_5_101Master (FB)	Implementiert die Zentralstation. Eine Instanz dieses Bausteins kann nur mit einer Unterstation kommunizieren. Für jede weitere Unterstation mit der kommuniziert werden soll wird eine weitere Instanz dieses Funktionsbausteins benötigt.
AODB	ARRAY OF ST_IEC870_5_101AODBEntry (STRUCT)	Applikationsobjektdatenbank. Es ist eine Array-Variable. Für jeden Datenpunkt wird ein Arrayelement benötigt.
hTable	T_HAODBTable (STRUCT)	Applikationsobjektdatenbank-Handle. Mit Hilfe dieser Variablen wird auf die Applikationsobjektdatenbank zugegriffen und nie direkt auf die einzelnen Arrayelemente.
F_iecCreateTableHnd	FUNCTION	Im Initialisierungsschritt werden alle Arrayelemente von AODB mit diesem Funktionsaufruf einmalig initialisiert. Beim Erfolg liefert dieser Funktionsaufruf das Handle.
F_iecAddTableEntry	FUNCTION	Im Initialisierungsschritt werden die einzelnen Datenpunkte der Station mit diesem Funktionsaufruf einmalig konfiguriert.
memory	ST_MemoryData (STRUCT)	Diese Struktur beinhaltet die empfangenen und die zu sendende Prozessdaten die ständig aktualisiert werden.
fbHandler	FB_IEC870_5_101TableEventHandler (FB)	Dieser Funktionsbaustein teilt der Applikation mit wenn Datenpunktdaten geändert oder aktualisiert wurden.



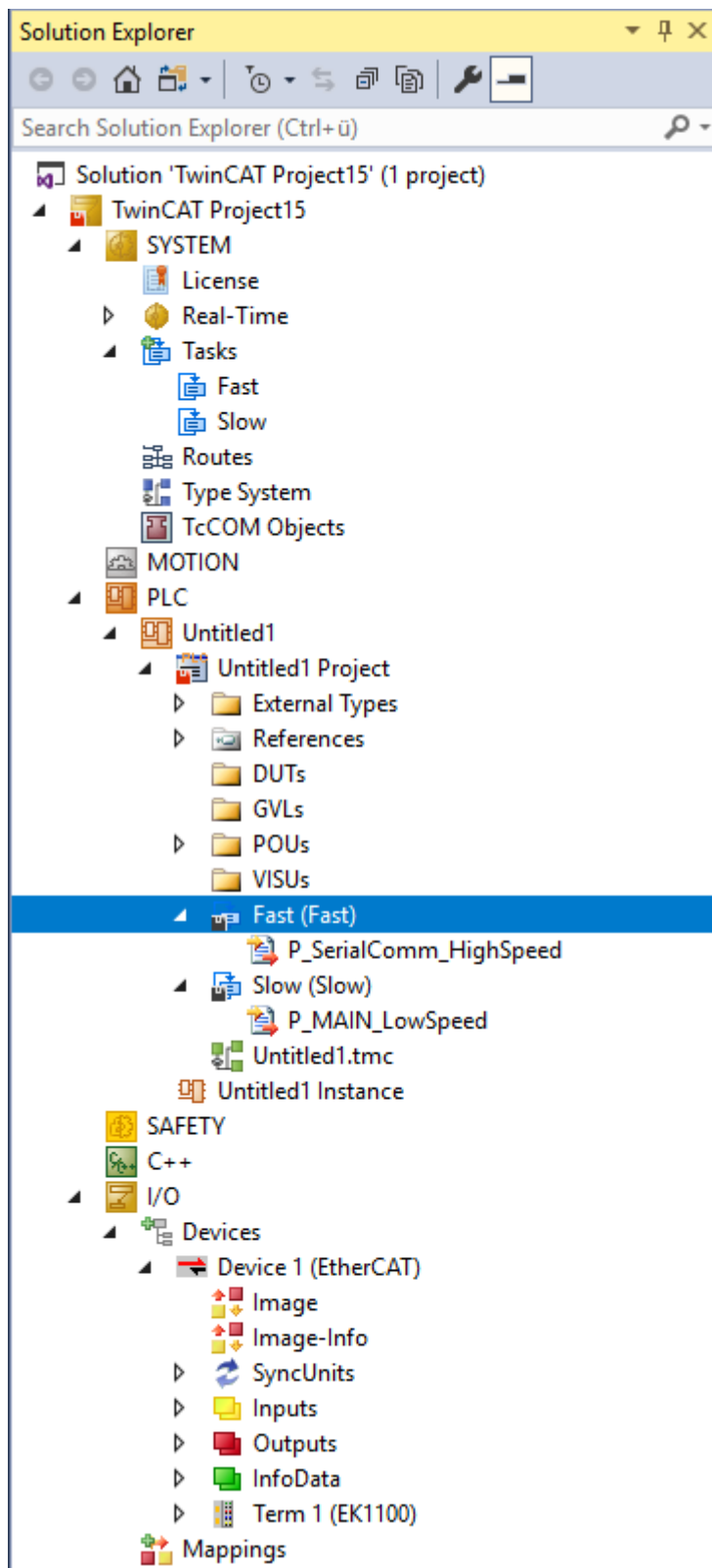
**6.2.1.1.2.2 Glossar**

<b>Begriff</b>	<b>Beschreibung</b>
Unterstation, Slave, Server	Synonyme für eine untergeordnete Station (sie wird überwacht)
Zentralstation, Leitstation, Master, Client	Synonyme für eine übergeordnete Station (Leitstation, sie überwacht andere Stationen)
Steuerungsrichtung	Datenübertragungsrichtung von der Zentralstation zur Unterstation
Überwachungsrichtung	Datenübertragungsrichtung von der Unterstation zur Zentralstation
Applikationsobjekte	IEC-Informationsobjekte in der TwinCAT SPS-Applikation (Single Points, Double Points, Measured Values, Short Floating Point Values usw. )
APDU	Protokolldateneinheit der Anwendungsschicht (application protocol data unit)
APCI	Protokollsteuerinformation der Anwendungsschicht (application protocol control information)
ASDU	Dienstdateneinheit der Anwendungsschicht (application service data unit)
IOA, Adresse des Informationsobjekts	Adresse des Single Points, Double Points usw. (information object address)
Primärstation	Die Primärstation sendet Befehle (Anfragen) an die Sekundärstation und steuert/kontrolliert so die Datenübertragung der Sekundärstation.
Sekundärstation	Die Sekundärstation antwortet auf die Anfragen der Primärstation.
Kombinierte Station	Kombinierte Stationen können die Rolle der Primärstation und Sekundärstation annehmen (symmetrische Übertragung, balanced mode).
Symmetrische Übertragung (balanced mode)	Beide Stationen können als Primärstation oder Sekundärstation agieren und die Datenübertragung initialisieren.
Unsymmetrische Übertragung (unbalanced mode)	Die Datenübertragung wird immer durch die Primärstation initiiert und gesteuert. Die Zentralstation agiert immer als Primärstation und die Unterstation als Sekundärstation.

**6.2.1.1.2.3 SPS-Projekt anlegen, SPS-Bibliotheken einbinden**

- ✓ Starten Sie TwinCAT XAE.
- 1. Mit **File > New > Project** öffnet sich ein Wizard. In diesem wählen Sie **TwinCAT Project** und das Template **TwinCAT XAE Project** aus.
- 2. Gehen Sie auf den PLC Knoten im Solution Explorer und wählen Sie über das Kontextmenu **Add New Item**. Wählen Sie im Wizard ein **Standard PLC Project** aus.
- 3. Neben der automatisch angelegten MAIN, fügen Sie einen weiteren Programmierbaustein hinzu. Gehen Sie auf den Ordner **POU** und von dort über das Kontextmenu auf **Add POU**. Wählen Sie den Typ **Program** aus und nennen es *P\_MAIN\_LowSpeed* .
- 4. Fügen Sie einen weiteren Programmbaustein hinzu und nennen diesen *P\_SerialComm\_HighSpeed*.

5. Konfigurieren Sie in der Taskkonfiguration unter System 2 Tasks, eine schnelle (T#1ms) und eine langsame (T#10ms). Ordnen Sie den Programmbaustein *P\_MAIN\_HighSpeed* der schnellen Task und den *P\_MAIN\_LowSpeed* der langsamen Task zu (siehe Bild).



6. Im PLC Project Knoten ist die Bibliotheksverwaltung im Unterknoten References zu finden. Dort wählen Sie im Kontextmenu über **Add Library** die Bibliothek **Tc2\_IEC60870\_5\_10x.compiled-library** aus und bestätigen dies.
- ⇒ Nun sind alle Voraussetzungen erfüllt, um das Tutorial schrittweise in Konfiguration und Programmierung durchzuarbeiten.

### 6.2.1.1.2.4 Die schnelle SPS-Task

Fügen Sie im Deklarationsteil folgenden SPS-Code hinzu:

```
PROGRAM P_SerialComm_HighSpeed
VAR
  fbSerialLineCtrl    : FB_IEC870_SerialLineCtrl := (
    Mode                := SERIALLINEMODE_PC_COM_PORT, (*SERIALLINEMODE_KL6_5B_STANDARD *)
    Baudrate            := 19200,
    NoDatabits          := 8,
    Parity              := PARITY_EVEN,
    Stopbits            := 1,
    Handshake           := HANDSHAKE_NONE,
    ContinousMode       := FALSE );

  serial_in            AT%IB4000      : PcComInData;
  serial_out           AT%QB4000      : PcComOutData;

  KL6_in               AT%IB4100      : KL6inData5B;
  KL6_out              AT%QB4100      : KL6outData5B;

  hSerial              : T_HSERIALCTRL;
END_VAR
```

und im Programmcode:

```
fbSerialLineCtrl( pComIn := SEL( fbSerialLineCtrl.Mode = SERIALLINEMODE_KL6_5B_STANDARD,
ADR( serial_in ), ADR( KL6_in ) ),
pComOut := SEL( fbSerialLineCtrl.Mode = SERIALLINEMODE_KL6_5B_STANDARD, ADR(
serial_out ), ADR( KL6_out ) ),
SizeComIn := SEL( fbSerialLineCtrl.Mode = SERIALLINEMODE_KL6_5B_STANDARD,
SIZEOF( serial_in ), SIZEOF( KL6_in ) ),
hSerial := hSerial );
```

Eine zu diesem Beispiel passende TwinCAT System Manager Konfiguration finden Sie auf der Beispiele-Übersichtsseite. Die Mode-Variable kann dazu verwendet werden um zwischen zwei Kommunikationswegen umzuschalten.

#### Kommunikation über die standard PC COMx-Schnittstellen

- In diesem Fall wird der Mode-Parameter auf den Wert: **SERIALLINEMODE\_PC\_COM\_PORT** gesetzt.
- Im TwinCAT System Manager werden die *serial\_in*- und *serial\_out*- Variablen mit den entsprechenden IO-Variablen der seriellen Schnittstelle verknüpft.
- Die Schnittstelle wird und muss im TwinCAT System Manager konfiguriert werden (Baudrate, Parity usw.). Andere Kommunikationsparameter an dem FB\_IEC870\_SerialLineCtrl-Funktionsbaustein sind in diesem Mode irrelevant.

#### Kommunikation über die seriellen Beckhoff Busklemmen KL6xxx

- In diesem Fall wird der Mode-Parameter auf den Wert: **SERIALLINEMODE\_KL6\_5B\_STANDARD** gesetzt.
- Im TwinCAT System Manager werden die *KL6\_in* - und *KL6\_out* - Variablen mit den entsprechenden I/O-Variablen der seriellen Klemme KL6xxx verknüpft.
- Die Schnittstelle wird in der TwinCAT SPS durch die Instanz des FB\_IEC870\_SerialLineCtrl-Funktionsbausteins konfiguriert. Die Kommunikationsparameter wie Baudrate, Parity usw. sind an diesem Baustein einzustellen.

#### Kommunikation über die seriellen Beckhoff Busklemmen EL6xxx

```
PROGRAM P_SerialComm_HighSpeed
VAR
  fbSerialLineCtrl : FB_IEC870_SerialLineCtrl := ( Mode := SERIALLINEMODE_EL6_22B );

  EL6_in AT%IB4100 : EL6inData22B;
  EL6_out AT%QB4100 : EL6outData22B;
  hSerial : T_HSERIALCTRL;
END_VAR

fbSerialLineCtrl( pComIn := ADR( EL6_in ),
pComOut := ADR( EL6_out ),
SizeComIn := SIZEOF( EL6_in ),
hSerial := hSerial );
```

- In diesem Fall wird der Mode-Parameter auf den Wert: **SERIALLINEMODE\_EL6\_22B** gesetzt.
- Im TwinCAT System Manager werden die *EL6\_in* - und *EL6\_out* - Variablen mit den entsprechenden I/O-Variablen der seriellen Klemme EL6xxx verknüpft.
- Die Schnittstelle wird und muss im TwinCAT System Manager konfiguriert werden (Baudrate, Parity usw.). Andere Kommunikationsparameter an dem FB\_IEC870\_SerialLineCtrl-Funktionsbaustein sind in diesem Mode irrelevant.

#### 6.2.1.1.2.5 Applikationsobjekt-Datenbank der Zentralstation definieren und konfigurieren

Applikationsobjekte = Single Points, Double Points, Measured Values, Short Floating Point Values usw.

In diesem Beispiel wurden die Befehle so konfiguriert, dass die Prozessdaten in Steuerungsrichtung (Befehle) im gleichen Speicherbereich aber auf einem anderen Byte-/Bit-Offset wie die Daten der Information in Überwachungsrichtung liegen. Sie können aber auch die Befehle auf den gleichen Byte- Bit-Offset wie die Information in Überwachungsrichtung legen.

Beispiel:

C\_SC\_NA\_1 mit IOA = 10 auf den gleichen Byte- und Bit-Offset wie M\_SP\_NA\_1 mit IOA = 100 (beide Byte-Offset = 100 und Bit-Offset = 0). Bei einer Wertänderung von M\_SP\_NA\_1 wird ein neues C\_SC\_NA\_1-Kommando ausgelöst.

Als Beispiel konfigurieren wir in dem Einführungsprojekt folgende Applikationsobjekte:

ASDU identifizier	Objekt-adresse IOA	Group-Konfigurationsparameter	Basiszeit-multiplikator	SPS-Prozessdatenbereich	Offset Byte	Offset Bit	Prozessdatenbreite in der TwinCAT SPS
M_SP_NA_1	100	Generalabfrage	0	Merker	100	0	1 Bit
M_SP_TA_1	101	Generalabfrage	0	Merker	100	1	1 Bit
M_SP_TB_1	102	Generalabfrage	0	Merker	100	2	1 Bit
M_DP_NA_1	200	Generalabfrage	0	Merker	200	0	2 Bits
M_DP_TA_1	201	Generalabfrage	0	Merker	200	2	2 Bits
M_DP_TB_1	202	Generalabfrage	0	Merker	200	4	2 Bits
M_ST_NA_1	300	Generalabfrage	0	Merker	300	0	1 Byte
M_ST_TA_1	301	Generalabfrage	0	Merker	301	0	1 Byte
M_ST_TB_1	302	Generalabfrage	0	Merker	302	0	1 Byte
M_BO_NA_1	400	Generalabfrage	0	Merker	400	0	4 Byte
M_BO_TA_1	401	Generalabfrage	0	Merker	404	0	4 Byte
M_BO_TB_1	402	Generalabfrage	0	Merker	408	0	4 Byte
M_ME_NA_1	500	Generalabfrage	0	Merker	500	0	2 Byte
M_ME_TA_1	501	Generalabfrage	0	Merker	502	0	2 Byte
M_ME_TD_1	502	Generalabfrage	0	Merker	504	0	2 Byte
M_ME_NB_1	600	Generalabfrage	0	Merker	600	0	2 Byte
M_ME_TB_1	601	Generalabfrage	0	Merker	602	0	2 Byte
M_ME_TE_1	602	Generalabfrage	0	Merker	604	0	2 Byte
M_ME_NC_1	700	Generalabfrage	0	Merker	700	0	4 Byte
M_ME_TC_1	701	Generalabfrage	0	Merker	704	0	4 Byte
M_ME_TF_1	702	Generalabfrage	0	Merker	708	0	4 Byte
M_IT_NA_1	800	Generalzählerabfrage und Mode A (lokal Umspeichern mit Spontanübertragung alle 15s)	0	Merker	800	0	4 Byte

ASDU identifizier	Objekt-adresse IOA	Group-Konfigurationsparameter	Basiszeit-multiplikator	SPS-Prozessdatenbereich	Offset Byte	Offset Bit	Prozessdatenbreite in der TwinCAT SPS
M_IT_TA_1	801	Generalzählerabfrage und Mode A (lokal Umspeichern mit Spontanübertragung alle 15s)	0	Merker	804	0	4 Byte
M_IT_TB_1	802	Generalzählerabfrage und Mode A (lokal Umspeichern mit Spontanübertragung alle 15s)	0	Merker	808	0	4 Byte
<b>Commands</b>							
C_SC_NA_1	10	-	0	Merker	2100	0	1 Bit
C_SC_NA_1	11	-	0	Merker	2100	1	1 Bit
C_SC_TA_1	12	-	0	Merker	2100	2	1 Bit
C_DC_NA_1	20	-	0	Merker	2200	0	2 Bit
C_DC_NA_1	21	-	0	Merker	2200	2	2 Bit
C_DC_TA_1	22	-	0	Merker	2200	4	2 Bit
C_BO_NA_1	40	-	0	Merker	2400	0	4 Byte
C_BO_NA_1	41	-	0	Merker	2404	0	4 Byte
C_BO_TA_1	42	-	0	Merker	2408	0	4 Byte
C_SE_NA_1	50	-	0	Merker	2500	0	2 Byte
C_SE_NA_1	51	-	0	Merker	2502	0	2 Byte
C_SE_TA_1	52	-	0	Merker	2504	0	2 Byte
C_SE_NB_1	60	-	0	Merker	2600	0	2 Byte
C_SE_NB_1	61	-	0	Merker	2602	0	2 Byte
C_SE_TB_1	62	-	0	Merker	2604	0	2 Byte
C_SE_NC_1	70	-	0	Merker	2700	0	4 Byte
C_SE_NC_1	71	-	0	Merker	2704	0	4 Byte
C_SE_TC_1	72	-	0	Merker	2708	0	4 Byte

### Datenbankvariable deklarieren

Die Applikationsobjekt-Datenbank ist eine Array-Variable vom Typ ST\_IEC870\_5\_101AODBEntry. Jedes Array-Element entspricht einem Applikationsobjekt. Die Array-Elemente werden aber nicht direkt, sondern nur mit Hilfe der speziell dafür zur Verfügung gestellten Funktionen und ein Datenbank-Handle (Tabellen-

Handle) manipuliert. Das Datenbank-Handle muss vor der Benutzung durch einen einmaligen F\_iecCreateTableHnd-Funktionsaufruf initialisiert werden. Dabei werden auch die Array-Elemente miteinander als Hash-Tabelle verknüpft. Bei einer größeren Anzahl der Datenpunkte ermöglicht die Hash-Tabelle einen schnelleren Zugriff auf einen einzelnen Datenpunkt.

Die maximale Anzahl der Applikationsobjekte ist frei wählbar und nur durch den verfügbaren Speicher begrenzt. Sie müssen sich auf eine konstante maximale Anzahl während der SPS-Programmierung festlegen. Zur Laufzeit kann die maximale Anzahl der Applikationsobjekte nicht mehr verändert werden. In unserem Beispiel werden 50 Applikationsobjekte deklariert. Diese Anzahl reicht für die meisten Anwendungen aus. Beachten Sie, dass sehr viele Applikationsobjekte auch entsprechend viel Speicher und Laufzeit benötigen.

Definieren Sie folgende Variablen in P\_MAIN\_LowSpeed:

```
PROGRAM P_MAIN_LowSpeed
VAR
    AODB          : ARRAY[0..49] OF ST_IEC870_5_101AODBEntry;
    hTable        : T_HAODBTable;
END_VAR
```

**Applikationsobjekte konfigurieren**

Die Konfiguration der gewünschten Applikationsobjekte wird zur Programmlaufzeit durchgeführt. Während der Konfiguration werden unter anderem der Objekt-Typ ( M\_SP\_NA\_1, M\_DP\_NA\_1, M\_ST\_NA\_1 usw.), die gemeinsame ASDU-Adresse, die Objekt-Adresse und weitere Objekt-Parameter festgelegt.

Nach der Initialisierung des Datenbank-Handles ist die Applikationsobjekt-Datenbank (Datenbank-Array) leer und muss mit den gewünschten Daten (Datenpunkten) gefüllt werden. Die Konfiguration der Datenpunkte der Zentralstation muss der Konfiguration der Datenpunkte in der Unterstation entsprechen! D.h. in der Zentralstation müssen Datenpunkte vom gleichen Typ, gleicher gemeinsamen ASDU-Adresse und mit der gleichen Informationsobjekt-Adresse wie in der Unterstation konfiguriert werden. Andere Parameter wie z.B. das Mapping-Bereich, Byte-, Bit-Offset können beliebig konfiguriert werden.

Es stehen folgende Funktionen zur Manipulation der Applikationsdatenbank zur Verfügung:

Function	Description
F_iecCreateTableHnd	Initialisiert das Hash-Tabellenhandle
F_iecAddTableEntry	Konfiguriert und fügt einen neuen Hash-Tabelleneintrag
F_iecRemoveTableEntry	Entfernt einen Hash-Tabelleneintrag
F_iecLookupTableEntry	Prüft ob ein bestimmter Hash-Tabelleneintrag existiert
F_iecGetPosOfTableEntry	Ermittelt die lineare Position eines Hash-Tabelleneintrags

Das Datenbank-Handle muss an die Funktion per VAR\_IN\_OUT übergeben werden. Im Regelfall wird die Konfiguration beim SPS-Programmstart einmalig in einer Init-Routine durchgeführt.

Um die Applikationsobjekte beim Programmstart zu konfigurieren wird in P\_MAIN\_LowSpeed folgender SPS-Code hinzugefügt:

```
PROGRAM P_MAIN_LowSpeed
VAR
    AODB : ARRAY[0..49] OF ST_IEC870_5_101AODBEntry;
    hTable : T_HAODBTable;

    init : BOOL := TRUE;
    initError : UDINT;
    asduAddr : UDINT := 7;
END_VAR

IF init THEN
    init := FALSE;

    initError := F_iecCreateTableHnd( ADR( AODB ), SIZEOF( AODB ), hTable );
```

```

IF initError <> 0 THEN
    ADSLOGSTR( ADSLOG_MSGTYPE_HINT OR ADSLOG_MSGTYPE_LOG,
        'F_iecCreateTableHnd() error: %s',
        DWORD_TO_HEXSTR( initError, 8, FALSE ) );
    RETURN;
END_IF

(* Monitored Single Points *)
initError := F_iecAddTableEntry( M_SP_NA_1, asduAddr, 100, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
100, 0, 0, hTable );
    initError := F_iecAddTableEntry( M_SP_NA_1, asduAddr, 101, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
Y, 100, 1, 0, hTable );
    initError := F_iecAddTableEntry( M_SP_TB_1, asduAddr, 102, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
Y, 100, 2, 0, hTable );
(* Double Points*)
initError := F_iecAddTableEntry( M_DP_NA_1, asduAddr, 200, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
200, 0, 0, hTable );
    initError := F_iecAddTableEntry( M_DP_NA_1, asduAddr, 201, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
Y, 200, 2, 0, hTable );
    initError := F_iecAddTableEntry( M_DP_TB_1, asduAddr, 202, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
Y, 200, 4, 0, hTable );
(* Regulating step value *)
initError := F_iecAddTableEntry( M_ST_NA_1, asduAddr, 300, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
300, 0, 0, hTable );
    initError := F_iecAddTableEntry( M_ST_NA_1, asduAddr, 301, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
Y, 301, 0, 0, hTable );
    initError := F_iecAddTableEntry( M_ST_TB_1, asduAddr, 302, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
Y, 302, 0, 0, hTable );
(* 32 bit string *)
initError := F_iecAddTableEntry( M_BO_NA_1, asduAddr, 400, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
400, 0, 0, hTable );
    initError := F_iecAddTableEntry( M_BO_NA_1, asduAddr, 401, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
Y, 404, 0, 0, hTable );
    initError := F_iecAddTableEntry( M_BO_TB_1, asduAddr, 402, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
Y, 408, 0, 0, hTable );
(* Measured value, normalized value *)
initError := F_iecAddTableEntry( M_ME_NA_1, asduAddr, 500, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
500, 0, 0, hTable );
    initError := F_iecAddTableEntry( M_ME_NA_1, asduAddr, 501, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
Y, 502, 0, 0, hTable );
    initError := F_iecAddTableEntry( M_ME_TD_1, asduAddr, 502, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
Y, 504, 0, 0, hTable );
(* Measured value, scaled value *)
initError := F_iecAddTableEntry( M_ME_NB_1, asduAddr, 600, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
600, 0, 0, hTable );
    initError := F_iecAddTableEntry( M_ME_NB_1, asduAddr, 601, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
Y, 602, 0, 0, hTable );
    initError := F_iecAddTableEntry( M_ME_TE_1, asduAddr, 602, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
Y, 604, 0, 0, hTable );
(* Measured value, short floating point value *)
initError := F_iecAddTableEntry( M_ME_NC_1, asduAddr, 700, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
700, 0, 0, hTable );
    initError := F_iecAddTableEntry( M_ME_NC_1, asduAddr, 701, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
Y, 704, 0, 0, hTable );
    initError := F_iecAddTableEntry( M_ME_TF_1, asduAddr, 702, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
Y, 708, 0, 0, hTable );
(* Integrated totals *)
initError := F_iecAddTableEntry( M_IT_NA_1, asduAddr, 800, IEC870_GRP_REQCOGEN, 0, MAP_AREA_MEMORY,
800, 0, 0, hTable );
    initError := F_iecAddTableEntry( M_IT_NA_1, asduAddr, 801, IEC870_GRP_REQCOGEN, 0, MAP_AREA_MEMORY,
RY, 804, 0, 0, hTable );
    initError := F_iecAddTableEntry( M_IT_TB_1, asduAddr, 802, IEC870_GRP_REQCOGEN, 0, MAP_AREA_MEMORY,
RY, 808, 0, 0, hTable );

(* Single commands *)
initError := F_iecAddTableEntry( C_SC_NA_1, asduAddr, 10, 0, 0, MAP_AREA_MEMORY, 2100, 0, 0, hTable
);
    initError := F_iecAddTableEntry( C_SC_NA_1, asduAddr, 11, 0, 0, MAP_AREA_MEMORY, 2100, 1, 0, hTa
ble );
    initError := F_iecAddTableEntry( C_SC_TA_1, asduAddr, 12, 0, 0, MAP_AREA_MEMORY, 2100, 2, 0, hTa
ble );
(* Double commands *)
initError := F_iecAddTableEntry( C_DC_NA_1, asduAddr, 20, 0, 0, MAP_AREA_MEMORY, 2200, 0, 0, hTable
);
    initError := F_iecAddTableEntry( C_DC_NA_1, asduAddr, 21, 0, 0, MAP_AREA_MEMORY, 2200, 2, 0, hTa
ble );
    initError := F_iecAddTableEntry( C_DC_TA_1, asduAddr, 22, 0, 0, MAP_AREA_MEMORY, 2200, 4, 0, hTa
ble );
(* 32 bit string commands *)

```



```

initError := F_iecAddTableEntry( C_BO_NA_1, asduAddr, 40, 0, 0, MAP_AREA_MEMORY, 2400, 0, 0, hTable
);
  initError := F_iecAddTableEntry( C_BO_NA_1, asduAddr, 41, 0, 0, MAP_AREA_MEMORY, 2404, 0, 0, hTa
ble );
  initError := F_iecAddTableEntry( C_BO_TA_1, asduAddr, 42, 0, 0, MAP_AREA_MEMORY, 2408, 0, 0, hTa
ble );
  (* Set point, normalized values*)
  initError := F_iecAddTableEntry( C_SE_NA_1, asduAddr, 50, 0, 0, MAP_AREA_MEMORY, 2500, 0, 0, hTable
);
  initError := F_iecAddTableEntry( C_SE_NA_1, asduAddr, 51, 0, 0, MAP_AREA_MEMORY, 2502, 0, 0, hTa
ble );
  initError := F_iecAddTableEntry( C_SE_TA_1, asduAddr, 52, 0, 0, MAP_AREA_MEMORY, 2504, 0, 0, hTa
ble );
  (* Set point, scaled values *)
  initError := F_iecAddTableEntry( C_SE_NB_1, asduAddr, 60, 0, 0, MAP_AREA_MEMORY, 2600, 0, 0, hTable
);
  initError := F_iecAddTableEntry( C_SE_NB_1, asduAddr, 61, 0, 0, MAP_AREA_MEMORY, 2602, 0, 0, hTa
ble );
  initError := F_iecAddTableEntry( C_SE_TB_1, asduAddr, 62, 0, 0, MAP_AREA_MEMORY, 2604, 0, 0, hTa
ble );
  (* Set point, short floating point values *)
  initError := F_iecAddTableEntry( C_SE_NC_1, asduAddr, 70, 0, 0, MAP_AREA_MEMORY, 2700, 0, 0, hTable
);
  initError := F_iecAddTableEntry( C_SE_NC_1, asduAddr, 71, 0, 0, MAP_AREA_MEMORY, 2704, 0, 0, hTa
ble );
  initError := F_iecAddTableEntry( C_SE_TC_1, asduAddr, 72, 0, 0, MAP_AREA_MEMORY, 2708, 0, 0, hTa
ble );
END_IF

```

### 6.2.1.1.2.6 Mapping der SPS- und IEC-Prozessdaten

Die TwinCAT SPS-Prozessdaten werden zur Programmlaufzeit zyklisch in die IEC-Prozessdaten (Applikationsobjekte) und umgekehrt gemappt (kopiert). Für das Mapping der IEC<->SPS Prozessdaten können bis zu 4 Prozessdatenbereiche (IO-Eingänge, IO-Ausgänge, Merkerbereich, Datenbereich) als Puffervariablen im SPS-Programm deklariert werden. Die Bytegröße der Puffer ist frei wählbar und kann für jeden Bereich unterschiedlich gewählt werden. Unbenutzte Bereiche müssen nicht unbedingt deklariert werden.

In unserem Einführungsbeispiel deklarieren wir 4 SPS-Prozessdatenbereiche mit jeweils 3000 Bytes:

```

PROGRAM P_MAIN_LowSpeed
VAR
  AODB      : ARRAY[0..49] OF ST_IEC870_5_101AODBEntry;
  hTable    : T_HAODBTable;

  init      : BOOL := TRUE;
  initError : UDINT;
  asduAddr  : UDINT := 7;

  inputs AT%IB0 : ARRAY[0..2999] OF BYTE;
  outputs AT%QB0 : ARRAY[0..2999] OF BYTE;
  memory AT%MB0 : ARRAY[0..2999] OF BYTE;
  data      : ARRAY[0..2999] OF BYTE;
END_VAR

```

Die Zuordnung, wie die Prozessdaten zur Laufzeit gemappt werden sollen, wird während der Konfiguration der Applikationsobjekte mit der `F_iecAddTableEntry` [► 453] Funktion festgelegt.

Siehe auch in: [Applikationsobjekte definieren und konfigurieren](#) [► 572].

Die Puffervariablen wurden nun als Byte-Arrays deklariert. Um auf die gewünschten Daten besser zugreifen zu können definieren wir die einzelnen Variablen ein zweites Mal und legen diese auf die entsprechenden Byte/Bit-Offsetadressen. Bei einer Änderung im Byte-Array wird die entsprechende einzelne Variable gleichzeitig geändert und umgekehrt. Dies ist aber nicht zwingend notwendig. Sie können direkt auf die Bytes/Bits der Byte-Array-Puffervariablen zugreifen.

```

VAR_GLOBAL(* Memory offset 0..99 unused *)
  (* Single points *)
  msgSingle_0 AT%MX100.0 : BOOL;
  msgSingle_1 AT%MX100.1 : BOOL;
  msgSingle_2 AT%MX100.2 : BOOL;

```

```

(* Double points *)
(*   Bit 0..1 = first double point,
   Bit 2..3 = second double point,
   Bit 4..5 = third double point,
   Bit 6..7 = fourth double point *)
msgDouble_0   AT%MB200   : BYTE;

(* Regulating step values *)
msgStep_0     AT%MB300   : BYTE;
msgStep_1     AT%MB301   : BYTE;
msgStep_2     AT%MB302   : BYTE;

(* 32 bit strings *)
msgBitStr_0   AT%MD400   : DWORD;
msgBitStr_1   AT%MD404   : DWORD;
msgBitStr_2   AT%MD408   : DWORD;

(* Measured values, normalized values *)
msgNormalized_0 AT%MW500   : WORD;
msgNormalized_1 AT%MW502   : WORD;
msgNormalized_2 AT%MW504   : WORD;

(* Measured values, scaled values *)
msgScaled_0    AT%MW600   : INT;
msgScaled_1    AT%MW602   : INT;
msgScaled_2    AT%MW604   : INT;

(* Measured values, short floating point values *)
msgFloating_0  AT%MD700   : REAL;
msgFloating_1  AT%MD704   : REAL;
msgFloating_2  AT%MD708   : REAL;

(* Integrated totals *)
msgTotal_0     AT%MD800   : UDINT;
msgTotal_1     AT%MD804   : UDINT;
msgTotal_2     AT%MD808   : UDINT;

(* Single commands *)
cmdSingle_0    AT%MX2100.0 : BOOL;
cmdSingle_1    AT%MX2100.1 : BOOL;
cmdSingle_2    AT%MX2100.2 : BOOL;

(* Double commands *)
(*   Bit 0..1 = first double command,
   Bit 2..3 = second double command,
   Bit 4..5 = third double command,
   Bit 6..7 = fourth double command *)
cmdDouble_0    AT%MB2200   : BYTE;

(* 32 bit string commands *)
cmdBitStr_0    AT%MD2400   : DWORD;
cmdBitStr_1    AT%MD2404   : DWORD;
cmdBitStr_2    AT%MD2408   : DWORD;

(* Set point, normalized values *)
cmdNormalized_0 AT%MW2500   : WORD;
cmdNormalized_1 AT%MW2502   : WORD;
cmdNormalized_2 AT%MW2504   : WORD;

(* Set point, scaled values *)
cmdScaled_0    AT%MW2600   : INT;
cmdScaled_1    AT%MW2602   : INT;
cmdScaled_2    AT%MW2604   : INT;

(* Set point, short floating point values *)
cmdFloating_0  AT%MD2700   : REAL;
cmdFloating_1  AT%MD2704   : REAL;
cmdFloating_2  AT%MD2708   : REAL;
END_VAR

```

## Mapping der IEC<->SPS Prozessdaten in der Zentralstation

### Prozessdaten in Überwachungsrichtung (Slave->Master information)

#### Beispiel 1

Single point information (M\_SP\_NA\_1) mit der IOA = 100, SPS Merkerbereich, Byteoffset = 100, Bitoffset = 0.

*Unterstation -> ... -> Zentralstation FB -> memory[100].0 == msgSingle\_0*

*Beispiel 2*

Measured value, short floating point value (M\_ME\_NC\_1) mit der IOA = 700, SPS Merkerbereich, Byteoffset = 700, Bitoffset = 0 (bedeutungslos).

*Unterstation -> ... -> Zentralstation FB -> memory[700..703] == msgFloating\_0*

**Prozessdaten in Steuerungsrichtung (Master->Slave commands)**

*Beispiel 1*

Single command state (C\_SC\_NA\_1) mit der IOA = 10, SPS Merkerbereich, Byteoffset = 2100, Bitoffset = 0.

*cmdSingle\_0 == memory[2100].0 -> Zentralstation FB -> ... -> Unterstation*

*Beispiel 2*

Set point, short floating point value (C\_SE\_NC\_1) mit der IOA = 70, SPS Merkerbereich, Byteoffset = 2700, Bitoffset = 0 (bedeutungslos).

*cmdFloating\_0 == memory[2700..2703] -> Zentralstation FB -> ... -> Unterstation*

**6.2.1.1.2.7 Instanz der IEC60870-5-101 Zentralstation deklarieren und aufrufen**

Die gesamte Funktionalität einer Zentralstation ist im Funktionsbaustein FB\_IEC870\_5\_101Master gekapselt. Mit einer Instanz kann eine Verbindung zur Unterstation aufgebaut werden. Für eine weitere Verbindung deklarieren Sie eine weitere Instanz des Funktionsbausteins.

Fügen Sie im Deklarationsteil von P\_MAIN\_LowSpeed folgenden SPS-Code ein:

```
PROGRAM P_MAIN_LowSpeed
VAR
  AODB          : ARRAY[0..49] OF ST_IEC870_5_101AODBEntry;
  hTable        : T_HAODBTable;

  init          : BOOL := TRUE;
  initError     : UDINT;
  asduAddr      : UDINT := 7;

  inputs AT%IB0 : ARRAY[0..2999] OF BYTE;
  outputs AT%QB0 : ARRAY[0..2999] OF BYTE;
  memory AT%MB0 : ARRAY[0..2999] OF BYTE;
  data          : ARRAY[0..2999] OF BYTE;

  bEnable       : BOOL := TRUE;
  client        : FB_IEC870_5_101Master;
END_VAR
```

und im Programmteil wird die Instanz aufgerufen:

```
IF init THEN
  init := FALSE;
  ...
ELSE
  ...
  client(
    pAOEntries := ADR( AODB ),
    cbAOEntries := SIZEOF( AODB ),
    pInputs := ADR( inputs ),
    cbInputs := SIZEOF( inputs ),
    pOutputs := ADR( outputs ),
    cbOutputs := SIZEOF( outputs ),
    pMemory := ADR( memory ),
    cbMemory := SIZEOF( memory ),
    pData := ADR( data ),
```

```

    cbData := SIZEOF( data ),
    bEnable := bEnable,
    hSerial := P_SerialComm_HighSpeed.hSerial,
    hTable := hTable );
...
END_IF

```

### 6.2.1.1.2.8 IEC60870-5-101-Protokollparameter

Über die IEC60870-5-101-Protokollparameter kann das Verhalten der Zentralstation an die Anforderungen der Unterstation angepasst werden. Die meisten Parameter sind mit Defaultwerten vorbelegt, so dass diese nicht verändert werden müssen.

Im unserem Beispiel konfigurieren wir die Verbindungsadresse, die Oktetlänge der Verbindungsadresse und die Zykluszeit in der die Class 1- und Class 2-Daten gepollt werden sollen.

```

IF init THEN
    init := FALSE;
...
(*Configure protocol parameter *)
client.protPara.linkAddr := 220; (* link address of remote slave *)
client.protPara.eLinkAddrSize := eIEC870_LinkAddr_TwoOctets; (* link address octet size *)
client.protPara.tClass1Poll := T#0ms; (* poll class 1 data with max. speed *)
client.protPara.tClass2Poll := T#0ms; (* poll class 2 data with max. speed *)
...
ELSE
    client( pInputs := ADR( inputs ),
           cbInputs := SIZEOF( inputs ),
           pOutputs := ADR( outputs ),
           ...
    );
END_IF

```

Die Dokumentation aller Übertragungsprotokoll-Parameter finden Sie hier: [ST IEC870 5 101PotocolParams \[► 538\]](#).

### 6.2.1.1.2.9 Systemparameter

Über die Systemparameter werden z. B. die gemeinsame ASDU-Adresse und die Anwenderfunktionen der Zentralstation konfiguriert.

In unserer Einführung konfigurieren wir folgende Systemparameter:

- Die gemeinsame ASDU-Adresse wird auf 7 gesetzt. (*asduAddr*);
- Die Oktetlänge der Übertragungsursache wird auf 2 gesetzt (1 Oktet für COT + 1 Oktet für Quelladresse) (*eCOTSize*);
- Die Oktetlänge der gemeinsamen ASDU Adresse wird auf 2 gesetzt (*eAsduAddrSize*);
- Die Oktetlänge der Informationsobjektadresse wird auf 3 gesetzt (*eObjAddrSize*);
- Das loggen der Debugmeldungen im Application-Log wird aktiviert (*dbgMode*). Es werden Änderungen im Gerätestatus und Fehlermeldungen der Verbindungsschicht geloggt;

Fügen Sie folgenden SPS-Code in Ihr SPS-Projekt ein:

```

IF init THEN
    init := FALSE;
...
    client.sysPara.asduAddr := 7;
    client.sysPara.asduFmt.eCOTSize := eIEC870_COT_TwoOctets; (* cause of transfer octet size *)
    client.sysPara.asduFmt.eAsduAddrSize := eIEC870_AsduAddr_TwoOctets;
    (* common ASDU address octet size *)
    client.sysPara.asduFmt.eObjAddrSize := eIEC870_ObjAddr_ThreeOctets;
    (* information object address octet size *)
    client.sysPara.dbgMode := IEC870_DEBUGMODE_DEVSTATE OR IEC870_DEBUGMODE_LINKERROR; (* IEC870_DEBUGMODE_ASDU OR IEC870_DEBUGMODE_LINKLAYER *)
...
ELSE
    client( pInputs := ADR( inputs ),
           cbInputs := SIZEOF( inputs ),
           ...
    );
END_IF

```

```
pOutputs := ADR( outputs ),
...
END_IF
```

Die Dokumentation aller Systemparameter finden Sie hier: [ST\\_IEC870\\_5\\_101SystemParams \[▶ 487\]](#).

### 6.2.1.1.2.10 Initialisierungssequence

```
client.acqPara.arrInitSeq[0] := eIEC870_ISTEP_TEST; (* Send test command *)
client.acqPara.arrInitSeq[1] := eIEC870_ISTEP_CLOCK; (* Send clock synchronization command *)
client.acqPara.arrInitSeq[2] := eIEC870_ISTEP_GENRO; (* Send general interrogation command *)
client.acqPara.arrInitSeq[3] := eIEC870_ISTEP_CORO; (* Send counter interrogation command *)
client.acqPara.arrInitSeq[5] := eIEC870_ISTEP_UNUSED; (* not used *)
```

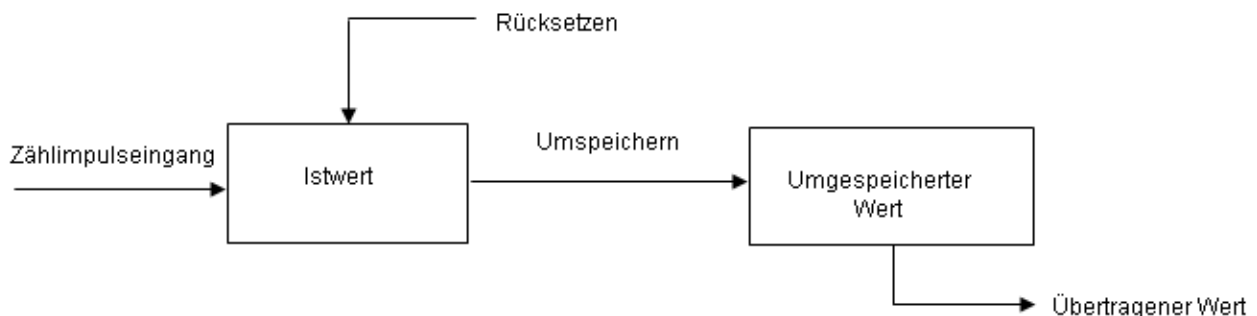
### 6.2.1.1.2.11 Stationsabfrage

Der Stationsabfragebefehl wird von der Zentralstation eingeleitet. Im Kennungsfeld des Befehls ist auch die Gruppe (1 bis 16 oder allgemein) festgelegt. Die Unterstation überträgt die zu dieser Gruppe dazugehörigen Applikationsobjekte mit der Übertragungsursache <20> bis <36> an die Zentralstation. Applikationsobjekte mit Zeitmarken werden ohne Zeitmarken übertragen.

```
client.acqPara.arrGenro[0].tPollCycle := T#60s;
client.acqPara.arrGenro[0].eQOI := eIEC870_QOI_INROGEN;
client.acqPara.arrGenro[0].bEnable := TRUE;
```

### 6.2.1.1.2.12 Zählwertübertragung (counter interrogation)

Allgemeines Modell für die Zählwertübertragung:



Die Istwerte werden durch Zähler aufsummiert. Die Istwerte können durch einen Umspeicherbefehl, der entweder von der Zentralstation empfangen oder örtlich (lokal in der Unterstation) erzeugt wird, periodisch in umgespeicherte Werte umgespeichert (kopiert) werden. Nach dem Umspeichern wird der erfasste Wert entweder auf Null zurückgesetzt (Erfassen von Inkrementalwerten) oder der Zähler fährt mit seinem Betrieb fort (Erfassen von Zählerständen).

Applikationsobjekte mit Zählwerten werden Gruppen zugeordnet. Die Gruppen werden einzeln umgespeichert (frozen), zurückgesetzt (reset) oder übertragen. Die Zentralstation sendet Zählwertabfragebefehle an die Unterstation. In einem Kennungsfeld des Befehls (QCC) wird die durchzuführende Aktion (FRZ) und Gruppe (RQT) festgelegt.

Die Zuordnung der Applikationsobjekte zu den einzelnen Gruppen (1 bis 4 oder allgemein) wird während der Konfiguration durch den Group-Flag-Parameter festgelegt. Es gibt vier Betriebsarten für die Erfassung von Zählerständen und Inkrementalwerten. Zu jeder Betriebsart sind einige Hinweise zur Konfiguration der Systemparameter oder der Applikationsobjekte aufgeführt.

#### Betriebsart A: Örtlich Umspeichern mit Spontanübertragung

Die Unterstation initiiert intern das Umspeichern oder Umspeichern mit Rücksetzen. Die umgespeicherten Zählwerte werden spontan übertragen, nachdem die Funktion Umspeichern oder Umspeichern mit Rücksetzen durchgeführt wurde. Die Zentralstation gibt in dieser Betriebsart keine Zählwertabfragebefehle aus.

**Konfiguration der Systemparameter:****Konfiguration der Applikationsobjekte:****Betriebsart B: Örtliches Umspeichern mit Zählerabfrage**

Die Unterstation initiiert intern das Umspeichern oder Umspeichern mit Rücksetzen. Die umgespeicherten Zählwerte werden durch Zählwertabfragebefehle von der Zentralstation abgefragt. In diesem Fall darf die Zentralstation im Befehlskennungsfeld das Umspeichern oder Umspeichern mit Rücksetzen nicht benutzen (FRZ=0). Die Zählwerte werden allgemein oder in Gruppen (groups) 1 bis 4 abgefragt.

**Konfiguration der Systemparameter:****Konfiguration der Applikationsobjekte:****Betriebsart C: Zentralstation leitet das Umspeichern, Umspeichern mit Rücksetzen oder Rücksetzen ein**

Ein Zählwertabfragebefehl wird periodisch von der Zentralstation an die Unterstation ausgegeben, um das Umspeichern oder (und) Rücksetzen zu steuern. Dieser Befehl hat aber noch keine Übertragung der Zählwerte zur Folge. Erst ein nachfolgender Zählwertabfragebefehl wird von der Zentralstation gesendet um die umgespeicherten Zählwerte einzusammeln. Ähnlich, wie bei der Betriebsart B.

**Konfiguration der Systemparameter:****Konfiguration der Applikationsobjekte:****Betriebsart D: Zentralstation leitet das Umspeichern und (oder) Rücksetzen ein und die umgespeicherten Werte werden spontan übertragen**

Diese Betriebsart ist eine Kombination des Zählwertbefehls von der Zentralstation wie für Betriebsart C mit einer spontanen Übertragung der Zählwerte wie bei der Betriebsart A.

**Konfiguration der Systemparameter:****Konfiguration der Applikationsobjekte:****6.2.1.1.2.13 Test der Kommunikation**

Durch das Setzen der *bExecuteCmd*-Variable auf TRUE wird eine einfache Simulation der Befehle in Steuerungsrichtung aktiviert und mit FALSE deaktiviert. Bei einer aktiven Verbindung wird in verwendeten Beispiel ein Einzel-Befehl (C\_SC\_NA\_1, IOA = 10 ) zyklisch alle 10 Sekunden zur Unterstation übertragen.

```
PROGRAM MAIN
VAR
    ...

    bExecuteCmd : BOOL;
    timer : TON;

    ...
END_VAR

...

(* Simple command simulation *)
timer( IN := bExecuteCmd, PT := T#10s ); (* Send cyclic command *)
IF timer.Q THEN
    timer( IN := FALSE );
    cmdSingle_0 := NOT cmdSingle_0; (* toggle single command ON<->OFF *)

(*     cmdDouble_0 := SEL( cmdDouble_0 = 1, 1, 2 );

    cmdBitStr_0 := cmdBitStr_0 + 1;

    cmdNormalized_0 := cmdNormalized_0 + 2;

    cmdScaled_0 := cmdScaled_0 + 4;

    cmdFloating_0 := cmdFloating_0 + 1.2; *)
```

```
END_IF
```

```
...
```

### 6.2.1.1.2.14 Übertragungs- und Kommunikationsfehler

Die Stationsfehlermeldungen werden in einem FIFO abgelegt. Es können bis zu 10 Fehlermeldungen zwischengespeichert werden. Bei fatalen Kommunikationsfehlern (z.B. Fehler der Verbindungsschicht, die Checksumme des Frames passt nicht) wird die Verbindung unterbrochen und muss neu aufgebaut werden. Fehler in der Applikationsschicht (z.B. der ASDU-Sendepuffer ist wegen zu vieler Frames übergelaufen) werden nur geloggt und führen nicht zum Verbindungsabbruch. Es immer noch möglich auch bei diesen Fehlern die Verbindung aus der Applikation zu unterbrechen. Neben dem Fehler-Code wird auch die Fehlerquelle in der Fehlermeldung abgelegt. Dieses erleichtert die Lokalisierung des Fehlers.

#### Beispiel

Die anfallenden Fehlermeldungen einer IEC 60870-5-101 Zentralstation können durch folgenden Aufruf ausgelesen werden:

```
PROGRAM MAIN
VAR
...
    client : FB_IEC870_5_101Master;
...
END_VAR

....

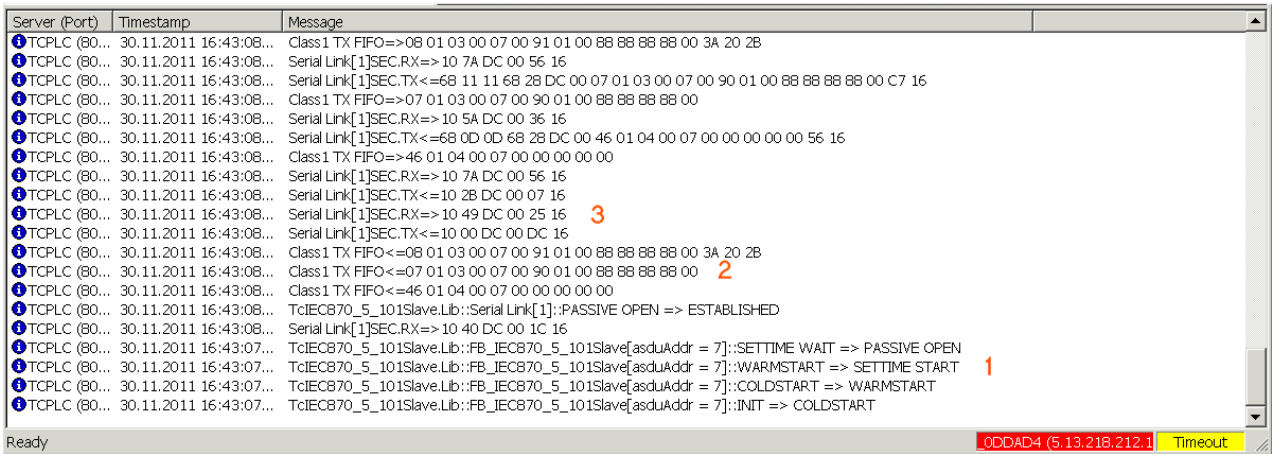
REPEAT
    client.system.device.errors.RemoveError( );
    IF client.system.device.errors.bOk THEN
        ADSLOGSTR( ADSLOG_MSGTYPE_ERROR OR ADSLOG_MSGTYPE_LOG,
            'IEC60870-5-101 master error: 0x%s',
            DWORD_TO_HEXSTR( client.system.device.errors.getError.nErrId, 8, FALSE) );
    END_IF
UNTIL NOT client.system.device.errors.bOk
END_REPEAT
...
```

### 6.2.1.1.2.15 Fehlersuche und Diagnose

Debug-Meldungen, die ins Application-Log geschrieben werden erleichtern eine Fehlersuche im System. Derzeit können drei Stufen der Debugmeldungen in einer IEC-Applikation aktiviert werden. Diese Meldungen können durch den dbgMode-Systemparameter der Leitstation aktiviert werden (ST IEC870\_5\_101SystemParams [▶ 487](#)).

1. Stationsstatus-Meldungen (dbgMode: IEC870\_DEBUGMODE\_DEVSTATE);
2. Hexadezimale Ausgabe der ASDU's (ohne Link-Layer-Control-Header, dbgMode: IEC870\_DEBUGMODE\_ASDU). Pro Zeile werden 32-ASDU-Datenbytes als hexadezimale Zahlen ausgegeben. Längere ASDU's werden auf mehrere Zeilen verteilt;
3. Hexadezimale Ausgabe der APDU's (Serial-Port-Telegramme, dbgMode: IEC870\_DEBUGMODE\_LINKLAYER). Pro Zeile werden 32-APDU-Datenbytes als hexadezimale Zahlen ausgegeben. Ähnlich wie bei 2. werden längere APDU's auf mehrere Zeilen verteilt;

Optional können auch Link-Layer-Fehler ausgegeben werden (dbgMode: IEC870\_DEBUGMODE\_LINKERROR). Um die aktivierten Debugmeldungen zu sehen starten Sie TwinCAT System Manager und aktivieren Sie die Loggeransicht. Die nachfolgende Grafik zeigt eine Debugausgabe. Die drei unterschiedlichen Typen der Meldungen wurden mit entsprechenden Zahlen markiert.



Weitere Diagnosetools:

- Portmon for Windows (v3.02, Windows Sysinternals);
- Diverse Protokoll-Test-Suit-Produkte;

### 6.2.1.1.2.16 Konfiguration der seriellen Schnittstellen

- [Konfiguration der standard PC COMx-Schnittstellen \[► 584\]](#)
- [Konfiguration der seriellen Busklemmen KL6xxx \[► 585\]](#)
- [Konfiguration der seriellen Busklemmen EL6xxx \[► 585\]](#)

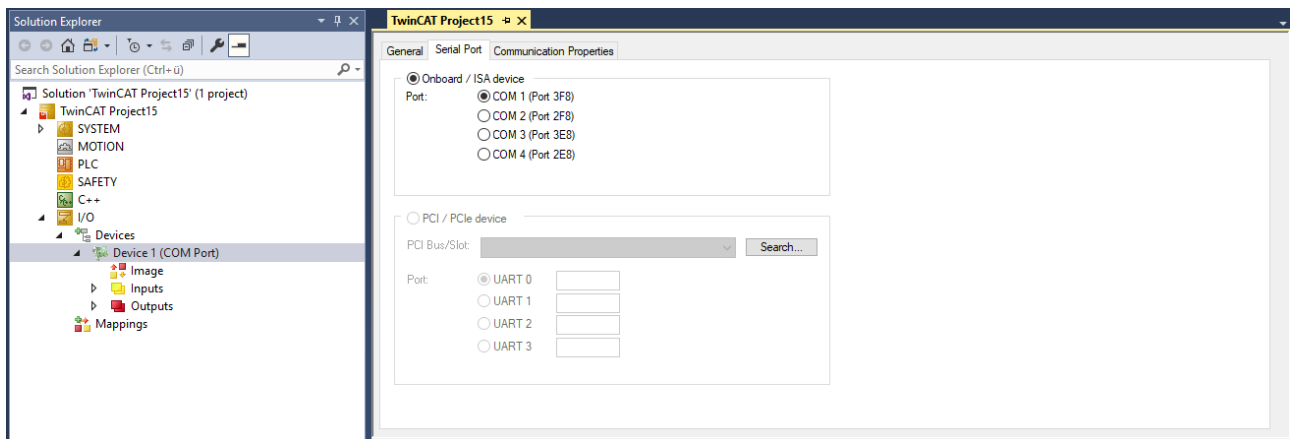
#### 1. Konfiguration der standard PC COMx-Schnittstellen

Die serielle Schnittstelle kann nur in PC/CX-Systemen eingefügt werden.

Mit der rechten Maustaste "E/A Geräte" anklicken. "Gerät anfügen" auswählen. Unter "Verschiedenes" die "Serielle Schnittstelle" auswählen. Anschließend nachfolgende Einstellungen vornehmen.

##### 1.1. Karteireiter Serial Port

Serielle Schnittstelle COM1

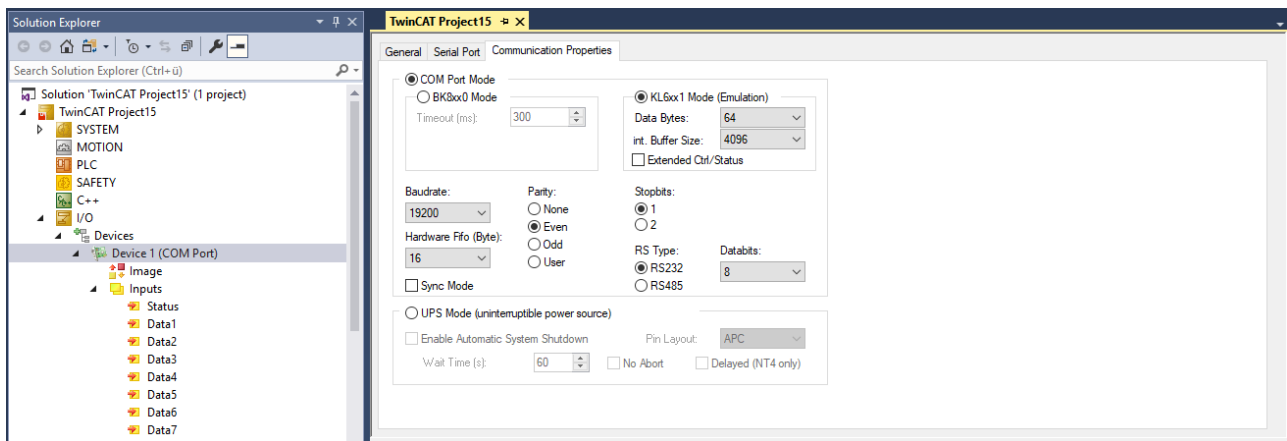


##### 1.2. Karteireiter Eigenschaften

Die Option KL6xxx1 Mode (Emulation) anwählen und dann die Kommunikationsparameter einstellen:

Baudrate, hier 19200 Baud, 8 Datenbits, Parity=even, 1 Stopbit





## 2. Konfiguration der seriellen Busklemmen KL6xxx

Die Schnittstelle wird in der TwinCAT SPS durch die Instanz des FB\_IEC870\_SerialLineCtrl-Funktionsbausteins konfiguriert. Die Kommunikationsparameter wie Baudrate, Parity usw. sind an diesem Baustein einzustellen.

### 2.1 RS485 Betriebsmodus

Bei der RS485 Betriebsart (z.B. KL6041) werden die Daten in halbduplex Übertragung ausgetauscht. Im RS485 Betriebsmodus werden die Sende- und Empfangsleitungen miteinander verbunden. Dadurch empfängt die Klemme nicht nur die Daten anderer Teilnehmer, sondern auch ihre eigenen Sendedaten. Diese Daten können die Kommunikation stören. Für den Halbduplex-Modus muss der Handshake-Eingang des FB\_IEC870\_SerialLineCtrl-Funktionsbausteins auf den Wert: RS485\_HALF DUPLEX gesetzt werden.

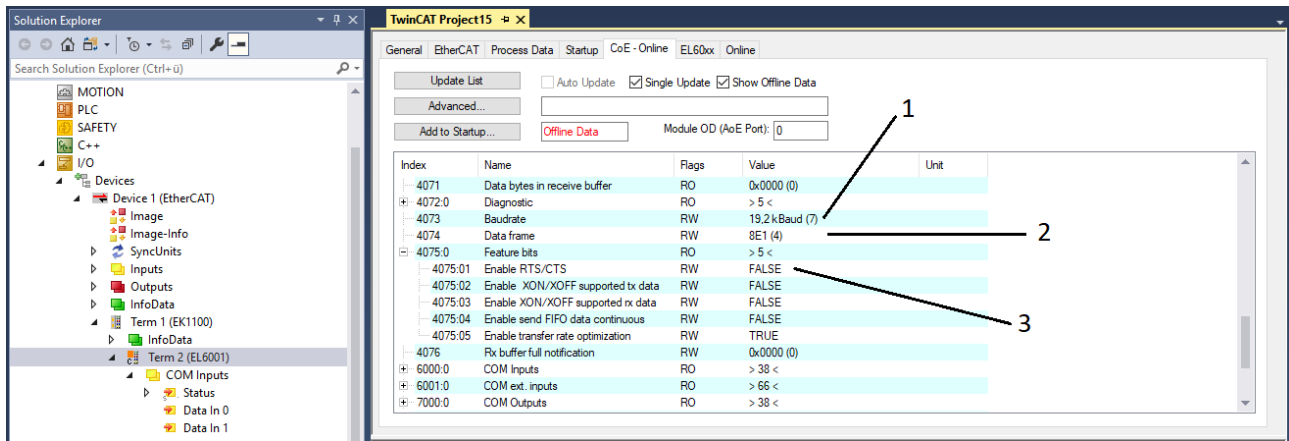
## 3. Konfiguration der seriellen Busklemmen EL6xxx

Die serielle Schnittstelle wird im TwinCAT System Manager Karteireiter CoE-Online (CoE=CanOpen over EtherCAT) eingestellt. Der Karteireiter steht nur bei Online-Zugriff auf die Klemme zur Verfügung, d.h., ist der System Manager nicht mit der Hardware verbunden, können keine Einstellungen verändert werden. Die Einstellungen werden in der Klemme Spannungsausfallsicher abgespeichert. Nach einem Scannen der Klemmen müssen diese Daten nicht neu eingegeben werden. Bei einem Tausch der Klemme werden die Daten aus der neuen Klemme aktiv und müssen demzufolge wieder angepasst werden. Um das zu verhindern, müssen im Karteireiter *Startup* die geänderten Daten eingetragen werden. Beim Systemstart werden die Daten aus dem Reiter *Startup* in die Klemme übertragen. Ein Tausch der Klemme ist somit problemlos möglich. Nach einem Scannen der Klemmen müssen die Daten im Reiter *Startup* neu eingegeben werden.

### 3.1. Karteireiter CoE

- (1) Baudrate, hier 19200 Baud
- (2) Date frame, 8E1 entspricht 8 Datenbits, Parity=even, 1 Stoppbit
- (3) Feature bits -> Enable RTS/CTS = *FALSE* bei EL6001

Mit einem Doppelklick auf die entsprechende Zeile wird ein Menü geöffnet, in dem die Einstellungen geändert werden können.



### 3.1.1 RS485 Betriebsmodus

Bei der RS485 Betriebsart werden die Daten in halbduplex Übertragung ausgetauscht. Im RS485 Betriebsmodus werden die Sende- und Empfangsleitungen miteinander verbunden. Dadurch empfängt die Klemme nicht nur die Daten anderer Teilnehmer, sondern auch ihre eigenen Sendedaten. Diese Daten können die Kommunikation stören. Mit der Option "Enable half duplex" im "COM Settings"-Objekt kann dies unterdrückt werden (Index 8000:06).

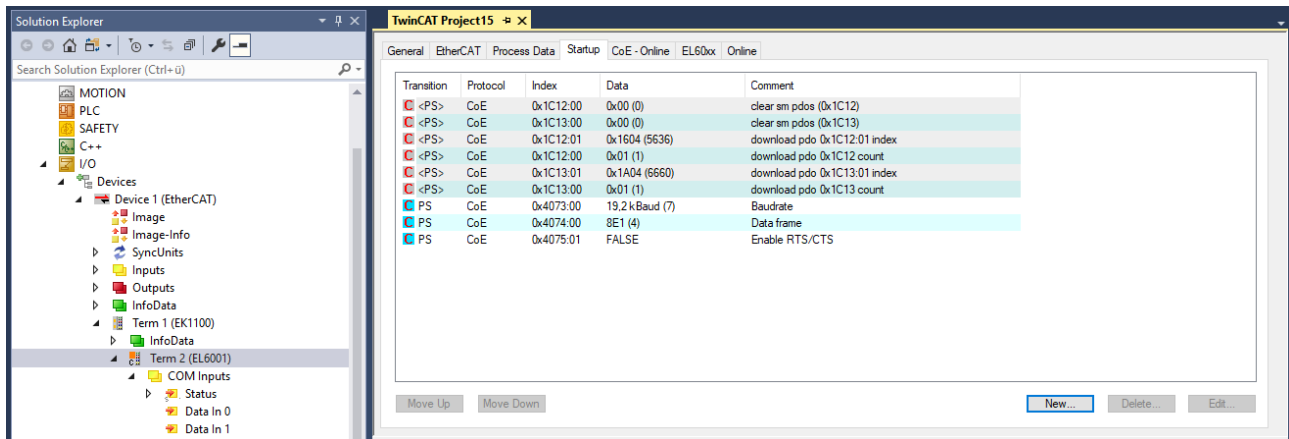
### 3.2. Karteireiter Startup

Alle vom Standard abweichenden Einstellungen müssen in die Startup Liste eingetragen werden. Beim Systemstart werden die Daten aus dem Reiter *Startup* in die Klemme übertragen. Ein Tausch der Klemme ist somit problemlos möglich. Nach einem Scannen der Klemmen müssen die Daten im Reiter *Startup* neu eingegeben werden.

Baudrate, hier 19200 Baud

Date frame, 8E1 entspricht 8 Datenbits, Parity=even, 1 Stoppsbit

Feature bits -> Enable RTS/CTS = *FALSE* bei EL6001



## 6.2.1.2 Low-Level Interface

### 6.2.1.2.1 Quick start

Einfaches Projekt mit kompletten Quellen finden Sie hier: [IEC 60870-5-101 Zentralstation-Low-Level-Beispiel \[► 587\]](#).

Eine SPS-Applikation, die über das "Low-Level"-Interface mit einer Unterstation kommunizieren soll, benötigt folgende Ressourcen:

- Eine Instanz des Kommunikationsbausteins: [FB IEC870\\_5\\_101TProtocol \[► 407\]](#);

- Eine Instanz des TX/RX-Datenpuffers: [ST\\_IEC870\\_5\\_101TBuffer \[► 481\]](#);
- Eine Instanz des Funktionsbausteins zur Manipulation des TX/RX-Datenpuffers: [FB\\_IEC870\\_5\\_101TBufferCtrl \[► 403\]](#);

### 6.2.1.2.1.1 Download: Zentralstation (master, unbalanced mode)

Einfache Zentralstation-Applikation die das TwinCAT IEC 60870-5-101 Low-Level Interface verwendet. Diese Beispielapplikation kann mit bis zu drei Unterstationen über eine RS232-Schnittstelle kommunizieren. Im Programm ist aber nur die Kommunikation zur ersten Unterstation aktiviert.

Download TwinCAT XAE Project (\*.zip): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/984606987.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/984606987.zip)

Im Beispielprojekt sind folgende Stationsparameter für die Zentralstation eingestellt:

- Standard PC interface: **COM1**.
- Baudrate: **19200 Baud**.
- Link address: **220 (erste Unterstation), 221 (zweite Unterstation), 222 (dritte Unterstation)**.
- Link address size: **1 octet**.
- Cause of transfer size: **1 octet**.
- Originator address: **not used (0)**.
- Common ASDU address: **7 (erste Unterstation), 8 (zweite Unterstation), 9 (dritte Unterstation)**.
- Common ASDU address size: **1 octet**.
- Information object address size: **2 octets**.
- Link mode: **Unbalanced**.

Datenpunktconfiguration (sie sollte gleich sein mit der Datenpunktconfiguration aller drei Unterstationen):

- Single point information: M\_SP\_NA\_1 (IOA := 100), M\_SP\_TA\_1 (IOA := 101), M\_SP\_TB\_1 (IOA := 102).
- Double point information: M\_DP\_NA\_1 (IOA := 200).
- Regulating step value: M\_ST\_NA\_1 (IOA := 300).
- Bitstring value: M\_BO\_NA\_1 (IOA := 400).
- Measured value, normalized value: M\_ME\_NA\_1 (IOA := 500).
- Measured value, scaled value: M\_ME\_NB\_1 (IOA := 600).
- Measured value, short floating point value: M\_ME\_NC\_1 (IOA := 700).
- Binary counter value: M\_IT\_NA\_1 (IOA := 800), M\_IT\_TA\_1 (IOA := 801), M\_IT\_TB\_1 (IOA := 802).
- Single command: C\_SC\_NA\_1 (IOA := 10).
- Double command: C\_DC\_NA\_1 (IOA := 20).
- Regulating step command: C\_RC\_NA\_1 (IOA := 30).
- Bitstring command: C\_BO\_NA\_1 (IOA := 40).
- Setpoint command, normalized value: C\_SE\_NA\_1 (IOA := 50).
- Setpoint command, scaled value: C\_SE\_NB\_1 (IOA := 60).
- Setpoint command, short floating point value: C\_SE\_NC\_1 (IOA := 70).

Beispielimplementierung folgender Dienste:

- Testbefehl (C\_TS\_NA\_1) alle 10 Sekunden.
- Uhrzeitsynchronisationsbefehl (C\_CS\_NA\_1) alle 20 Sekunden.
- Generalabfragebefehl (C\_IC\_NA\_1) der Gruppe: Allgemein (eIEC870\_QOI\_INROGEN) alle 40 Sekunden.
- Zählerabfragebefehl (C\_CI\_NA\_1) der Gruppe: Allgemein (eIEC870\_RQT\_REQCOGEN) alle 60 Sekunden. Zählerabfrage Mode A.

- Simulation und Übertragung der Befehle alle 60 Sekunden wenn die Variable **bSimulate** auf TRUE gesetzt wurde (siehe obere Datenpunktconfiguration).
- Empfang spontaner Daten.
- Empfang von M\_EI\_NA\_1 (Ende der Initialisierung).

Folgende Tabelle beschreibt kurz die Hauptkomponenten des Projekts:

Komponente	Typ	Beschreibung
P_MAIN_LowSpeed	PRG	Dieses Programmteil implementiert das Fernwirkprotokoll (langsame Task).
P_SerialComm_HighSpeed	PRG	Dieses Programmteil implementiert die TwinCAT Hintergrundkommunikation über die seriellen Schnittstellen des PCs (schnelle Task).
aClients	ARRAY OF FB_IEC870_5_101M (FB)	Implementiert die Zentralstation. Eine Instanz dieses Bausteins kann nur mit einer Unterstation kommunizieren. Für jede weitere Unterstation mit der kommuniziert werden soll wird eine weitere Instanz dieses Funktionsbausteins benötigt.
fbServices	FB_IEC870_5_101MS ervices (FB)	Beinhaltet eine einfache Implementierung einiger Dienste wie z.B. Generalabfrage, Uhrzeitsynchronisation, Testbefehle usw.
aProcData	ARRAY OF ST_IEC870_5_101MP rocData (STRUCT)	Hier befinden sich die empfangenen und die zu sendenden Prozessdaten die ständig aktualisiert werden.
sysPara	ST_IEC870_5_101MS ysPara (STRUCT)	Konfiguration der Systemparameter der Zentralstation und der implementierten Dienste.
fbLog	FB_IEC870_DebugLo gFifo (FB)	Ein einfacher Logpuffer für Debugzwecke/ Fehlersuche.

### 6.2.1.2.1.2 Download: Zentralstation (master, balanced mode)

Einfache Zentralstation-Applikation die das TwinCAT IEC 60870-5-101 Low-Level Interface verwendet. In der Beispielapplikation kann mit einer Unterstation über eine RS232-Schnittstelle kommuniziert werden.

Download TwinCAT XAE Project (\*.zip): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/984609931.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/984609931.zip)

Im Beispielprojekt sind folgende Stationsparameter für die Zentralstation eingestellt:

- Standard PC interface: **COM1**.
- Baudrate: **19200 Baud**.
- Link address: **220**.
- Link address size: **1 octet**.
- Cause of transfer size: **1 octet**.
- Originator address: **0 (not used)**.
- Common ASDU address: **7**.
- Common ASDU address size: **1 octet**.
- Information object address size: **2 octets**.
- Link mode: **Balanced**.

Datenpunktconfiguration (sie sollte gleich sein mit der Datenpunktconfiguration der Unterstationen):

- Single point information: M\_SP\_NA\_1 (IOA := 100), M\_SP\_TA\_1 (IOA := 101), M\_SP\_TB\_1 (IOA := 102).
- Double point information: M\_DP\_NA\_1 (IOA := 200).

- Regulating step value: M\_ST\_NA\_1 (IOA := 300).
- Bitstring value: M\_BO\_NA\_1 (IOA := 400).
- Measured value, normalized value: M\_ME\_NA\_1 (IOA := 500).
- Measured value, scaled value: M\_ME\_NB\_1 (IOA := 600).
- Measured value, short floating point value: M\_ME\_NC\_1 (IOA := 700).
- Binary counter value: M\_IT\_NA\_1 (IOA := 800), M\_IT\_TA\_1 (IOA := 801), M\_IT\_TB\_1 (IOA := 802).
- Single command: C\_SC\_NA\_1 (IOA := 10).
- Double command: C\_DC\_NA\_1 (IOA := 20).
- Regulating step command: C\_RC\_NA\_1 (IOA := 30).
- Bitstring command: C\_BO\_NA\_1 (IOA := 40).
- Setpoint command, normalized value: C\_SE\_NA\_1 (IOA := 50).
- Setpoint command, scaled value: C\_SE\_NB\_1 (IOA := 60).
- Setpoint command, short floating point value: C\_SE\_NC\_1 (IOA := 70).

Beispielimplementierung folgender Dienste:

- Testbefehl (C\_TS\_NA\_1) alle 10 Sekunden.
- Uhrzeitsynchronisationsbefehl (C\_CS\_NA\_1) alle 20 Sekunden.
- Generalabfragebefehl (C\_IC\_NA\_1) der Gruppe: Allgemein (eIEC870\_QOI\_INROGEN) alle 40 Sekunden.
- Zählerabfragebefehl (C\_IC\_NA\_1) der Gruppe: Allgemein (eIEC870\_RQT\_REQCOGEN) alle 60 Sekunden. Zählerabfrage Mode A.
- Simulation und Übertragung der Befehle alle 60 Sekunden wenn die Variable **bSimulate** auf TRUE gesetzt wurde (siehe obere Datenpunktkonfiguration).
- Empfang spontaner Daten.
- Empfang von M\_EI\_NA\_1 (Ende der Initialisierung).

Folgende Tabelle beschreibt kurz die Hauptkomponenten des Projekts:

Komponente	Typ	Beschreibung
P_MAIN_LowSpeed	PRG	Dieses Programmteil implementiert das Fernwirkprotokoll (langsame Task).
P_SerialComm_HighSpeed	PRG	Dieses Programmteil implementiert die TwinCAT Hintergrundkommunikation über die seriellen Schnittstellen des PCs (schnelle Task).
fbClient	FB_IEC870_5_101M (FB)	Implementiert die Zentralstation. Eine Instanz dieses Bausteins kann nur mit einer Unterstation kommunizieren. Für jede weitere Unterstation mit der kommuniziert werden soll wird eine weitere Instanz dieses Funktionsbausteins benötigt.
fbServices	FB_IEC870_5_101MS ervices (FB)	Beinhaltet eine einfache Implementierung einiger Dienste wie z.B. Generalabfrage, Uhrzeitsynchronisation, Testbefehle usw.
stProcData	ST_IEC870_5_101MP rocData (STRUCT)	Hier befinden sich die empfangenen und die zu sendenden Prozessdaten die ständig aktualisiert werden.
sysPara	ST_IEC870_5_101MS ysPara (STRUCT)	Konfiguration der Systemparameter der Zentralstation und der implementierten Dienste.
fbLog	FB_IEC870_DebugLo gFifo (FB)	Ein einfacher Logpuffer für Debugzwecke/ Fehlersuche.

### 6.2.1.2.1.3 Konfiguration der seriellen Schnittstellen

- [Konfiguration der standard PC COMx-Schnittstellen \[► 590\]](#)

- [Konfiguration der seriellen Busklemmen KL6xxx \[► 590\]](#)
- [Konfiguration der seriellen Busklemmen EL6xxx \[► 591\]](#)

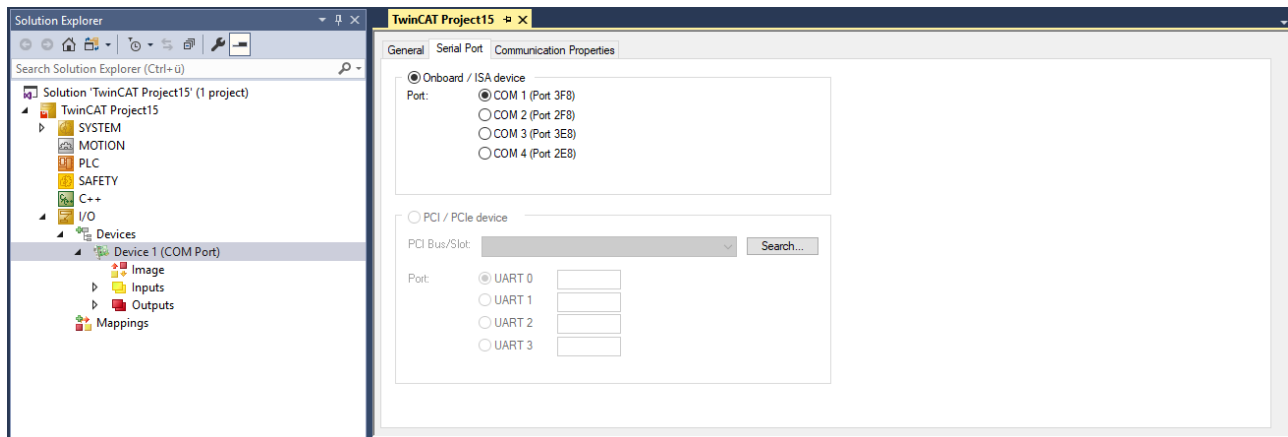
## 1. Konfiguration der standard PC COMx-Schnittstellen

Die serielle Schnittstelle kann nur in PC/CX-Systemen eingefügt werden.

Mit der rechten Maustaste "E/A Geräte" anklicken. "Gerät anfügen" auswählen. Unter "Verschiedenes" die "Serielle Schnittstelle" auswählen. Anschließend nachfolgende Einstellungen vornehmen.

### 1.1. Karteireiter Serial Port

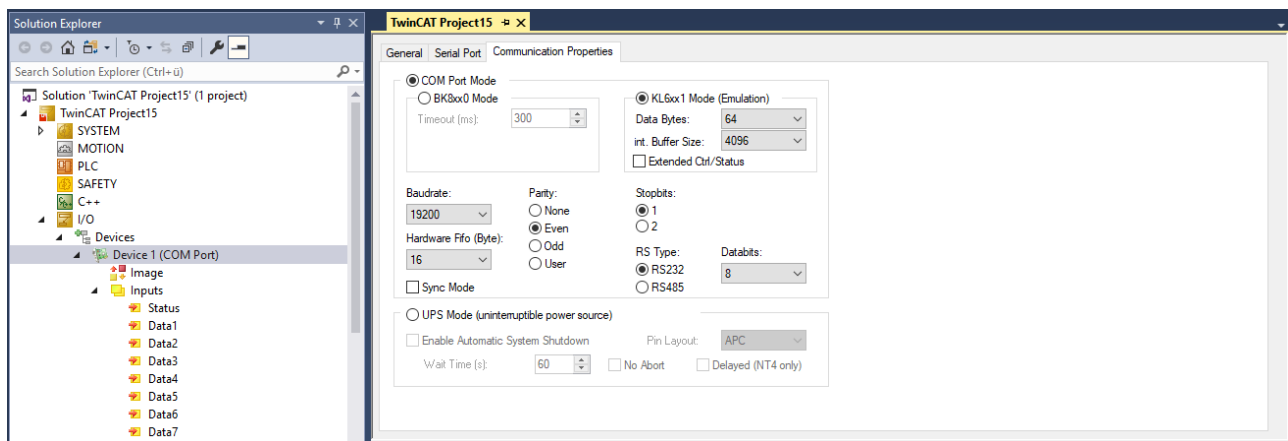
Serielle Schnittstelle COM1



### 1.2. Karteireiter Eigenschaften

Die Option KL6xxx1 Mode (Emulation) anwählen und dann die Kommunikationsparameter einstellen:

Baudrate, hier 19200 Baud, 8 Datenbits, Parity=even, 1 Stopbit



## 2. Konfiguration der seriellen Busklemmen KL6xxx

Die Schnittstelle wird in der TwinCAT SPS durch die Instanz des FB\_IEC870\_SerialLineCtrl-Funktionsbausteins konfiguriert. Die Kommunikationsparameter wie Baudrate, Parity usw. sind an diesem Baustein einzustellen.

### 2.1 RS485 Betriebsmodus

Bei der RS485 Betriebsart (z.B. KL6041) werden die Daten in halbduplex Übertragung ausgetauscht. Im RS485 Betriebsmodus werden die Sende- und Empfangsleitungen miteinander verbunden. Dadurch empfängt die Klemme nicht nur die Daten anderer Teilnehmer, sondern auch ihre eigenen Sendedaten. Diese Daten können die Kommunikation stören. Für den Halbduplex-Modus muss der Handshake-Eingang des FB\_IEC870\_SerialLineCtrl-Funktionsbausteins auf den Wert: RS485\_HALFDUPLEX gesetzt werden.

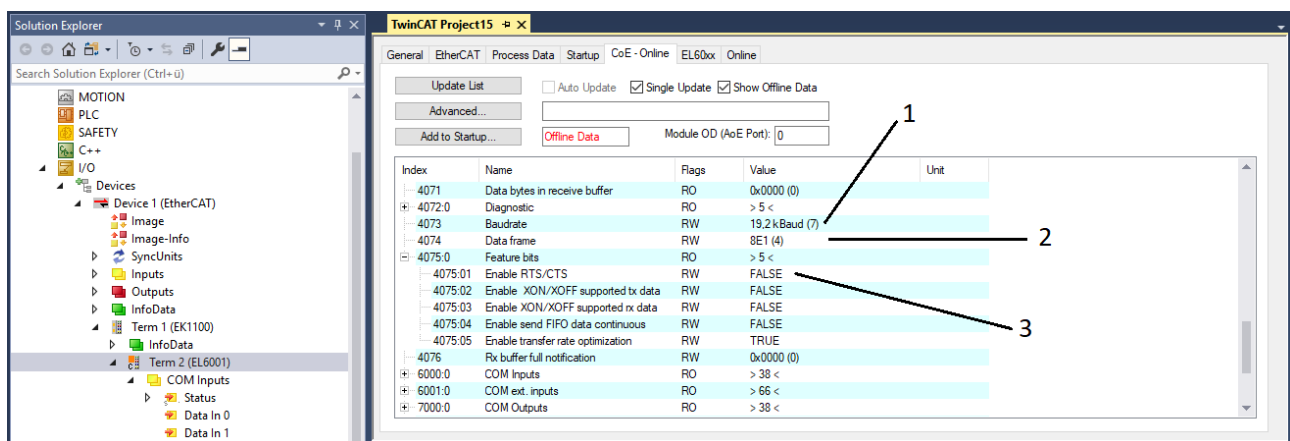
### 3. Konfiguration der seriellen Busklemmen EL6xxx

Die serielle Schnittstelle wird im TwinCAT System Manager Karteireiter CoE-Online (CoE=CanOpen over EtherCAT) eingestellt. Der Karteireiter steht nur bei Online-Zugriff auf die Klemme zur Verfügung, d.h., ist der System Manager nicht mit der Hardware verbunden, können keine Einstellungen verändert werden. Die Einstellungen werden in der Klemme Spannungsausfallsicher abgespeichert. Nach einem Scannen der Klemmen müssen diese Daten nicht neu eingegeben werden. Bei einem Tausch der Klemme werden die Daten aus der neuen Klemme aktiv und müssen demzufolge wieder angepasst werden. Um das zu verhindern, müssen im Karteireiter *Startup* die geänderten Daten eingetragen werden. Beim Systemstart werden die Daten aus dem Reiter *Startup* in die Klemme übertragen. Ein Tausch der Klemme ist somit problemlos möglich. Nach einem Scannen der Klemmen müssen die Daten im Reiter *Startup* neu eingegeben werden.

#### 3.1. Karteireiter CoE

- (1) Baudrate, hier 19200 Baud
- (2) Date frame, 8E1 entspricht 8 Datenbits, Parity=even, 1 Stopbit
- (3) Feature bits -> Enable RTS/CTS = *FALSE* bei EL6001

Mit einem Doppelklick auf die entsprechende Zeile wird ein Menü geöffnet, in dem die Einstellungen geändert werden können.



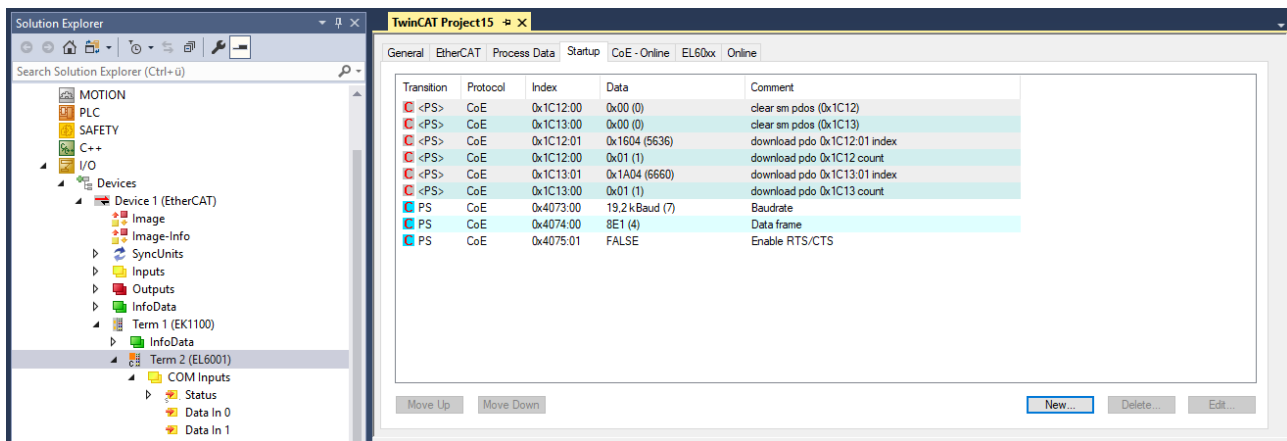
#### 3.1.1 RS485 Betriebsmodus

Bei der RS485 Betriebsart werden die Daten in halbduplex Übertragung ausgetauscht. Im RS485 Betriebsmodus werden die Sende- und Empfangsleitungen miteinander verbunden. Dadurch empfängt die Klemme nicht nur die Daten anderer Teilnehmer, sondern auch ihre eigenen Sendedaten. Diese Daten können die Kommunikation stören. Mit der Option "Enable half duplex" im "COM Settings"-Objekt kann dies unterdrückt werden (Index 8000:06).

#### 3.2. Karteireiter Startup

Alle vom Standard abweichenden Einstellungen müssen in die Startup Liste eingetragen werden. Beim Systemstart werden die Daten aus dem Reiter *Startup* in die Klemme übertragen. Ein Tausch der Klemme ist somit problemlos möglich. Nach einem Scannen der Klemmen müssen die Daten im Reiter *Startup* neu eingegeben werden.

- Baudrate, hier 19200 Baud
- Date frame, 8E1 entspricht 8 Datenbits, Parity=even, 1 Stopbit
- Feature bits -> Enable RTS/CTS = *FALSE* bei EL6001



## 6.2.2 Unterstation

### 6.2.2.1 High-Level Interface

#### 6.2.2.1.1 Quick start

Einfaches Projekt mit kompletten Sourcen finden Sie hier: [IEC 60870-5-101 Unterstation-High-Level-Beispiel \[► 592\]](#).

Kompatibilitätsliste finden Sie hier: [Interoperability check list \[► 32\]](#)

Übersicht der Fehlercodes finden Sie hier: [Fehlercodes \[► 685\]](#)

#### Kurzanleitung

1. Erstellen Sie ein neues SPS-Projekt erstellen binden Sie die SPS-Bibliothek: **Tc2\_IEC60870\_5\_10x** ein.
2. Legen Sie zwei SPS-Tasks an, eine schnelle (z.B. mit Zykluszeit T#1ms) und eine langsame (z.B. mit Zykluszeit T#10ms).  
Legen Sie zwei Programmbausteine an (z.B. *P\_SerialComm\_HighSpeed* und *P\_MAIN\_LowSpeed*).  
*P\_SerialComm\_HighSpeed* wird von der schnellen und *P\_MAIN\_LowSpeed* von der langsamen Task aufgerufen.
3. Legen Sie im *P\_SerialComm\_HighSpeed* eine Instanz des Funktionsbausteins [FB\\_IEC870\\_SerialLineCtrl \[► 417\]](#) an, konfigurieren Sie diese und rufen Sie diese auf.  
Je nach dem, ob über die seriellen Beckhoff-Klemmen oder über die serielle Schnittstelle des PCs kommuniziert wird, die Puffer: *KL6inData5B*, *KL6outData5B* oder *PcComInData*, *PcComOutData* anlegen und mit den entsprechen IO-Prozessdaten im TwinCAT Systemmanager verknüpfen.
4. Die Instanz der [T\\_HSERIALCTRL \[► 530\]](#)-Variablen dient dem Austausch der Tx/Rx-Telegramme zwischen den beiden Tasks (Programmen). Legen Sie diese Variable z.B. als globale Variable an.
5. Konfiguration der Datenpunkte: Legen Sie eine Array-Variable vom Typ [ST\\_IEC870\\_5\\_101AODBEntry \[► 482\]](#) an. Jedes Arrayelement entspricht einem Datenpunkt. Konfigurieren Sie die Datenpunkte mit Hilfe der Funktion [F\\_ieclnitAOEntry \[► 435\]](#) zur Laufzeit (z.B. in einem Init-Schritt).
6. Legen Sie im *P\_MAIN\_LowSpeed* eine Instanz des Protokoll-Bausteins [FB\\_IEC870\\_5\\_101Slave \[► 387\]](#) an, konfigurieren Sie diese und rufen Sie diese auf.
7. Konfigurieren Sie die System- und Protokoll-Parameter passend zu den Parametern der Leitstation.

#### 6.2.2.1.1.1 Download: Unterstation Minimal (slave, unbalanced mode)

Einfache Unterstation-Applikation die das TwinCAT IEC 60870-5-101 High-Level Interface nutzt.

Download TwinCAT XAE Project (\*.zip): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/984612875.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/984612875.zip)



Im Beispielprojekt sind folgende Stationsparameter für die Unterstation eingestellt:

- Standard PC interface: **COM1**.
- Baudrate: **19200 Baud**.
- Link address: **220**.
- Link address size: **2 octets**.
- Cause of transfer size: **2 octets (inclusive originator address)**.
- Originator address: **1**.
- Common ASDU address: **7**.
- Common ASDU address size: **2 octets**.
- Information object address size: **3 octets**.
- Link mode: **Unbalanced**.

Datenpunktkonfiguration (sie sollte gleich sein mit der Datenpunktkonfiguration der Zentralstation):

- Single point information: M\_SP\_NA\_1 (IOA := 100).
- Single command: C\_SC\_NA\_1 (IOA := 10).

Beispielimplementierung folgender Dienste:

- Ende der Initialisierung (M\_EI\_NA\_1) wird einmalig gesendet.
- Testbefehl (C\_TS\_NA\_1).
- Uhrzeitsynchronisationsbefehl (C\_CS\_NA\_1).
- Generalabfragebefehl (C\_IC\_NA\_1) der Gruppe: Allgemein (eIEC870\_QOI\_INROGEN).
- Zählerabfragebefehl (C\_CI\_NA\_1) der Gruppe: Allgemein (eIEC870\_RQT\_REQCOGEN). Zählerabfrage Mode C.
- Einzelbefehl-Bearbeitung (C\_SC\_NA\_1) (direkte Kommandoausführung)
- Das Senden spontaner Daten vom Typ (M\_SP\_NA\_1, IOA := 100) bei Wertänderung an der **bSPI**-Variablen.

Folgende Tabelle beschreibt kurz die Hauptkomponenten des Projekts:

Komponente	Typ	Beschreibung
P_MAIN_LowSpeed	PRG	Dieses Programmteil implementiert das Fernwirkprotokoll (langsame Task).
P_SerialComm_HighSpeed	PRG	Dieses Programmteil implementiert die TwinCAT Hintergrundkommunikation über die seriellen Schnittstellen des PCs (schnelle Task).
fbServer	FB_IEC870_5_101Slave (FB)	Implementiert die Unterstation. Eine Instanz dieses Bausteins kann nur mit einer Zentralstation kommunizieren. Für jede weitere Zentralstation mit der kommuniziert werden soll wird eine weitere Instanz dieses Funktionsbausteins benötigt.
bSPI	BOOL	Der zu sendende ON/OFF-Wert wird direkt von der an der Merkeradresse <b>AT%MX1.5</b> dieser lokierten SPS-Variablen gemappt.
bSCS	BOOL	Der empfangene ON/OFF-Wert wird in diese SPS-Variable direkt an die lokierte Merkeradresse: <b>AT%MX0.0</b> gemappt.
AODB	ARRAY OF ST_IEC870_5_101AOEntry (STRUCT)	Applikationsobjektdatenbank. Es ist eine Array-Variable. Für jeden Datenpunkt wird ein Arrayelement benötigt.
F_iecInitAOEntry	STRUCT	Im Initialisierungsschritt werden die einzelnen Datenpunkte der Station mit diesem Funktionsaufruf einmalig konfiguriert.

### 6.2.2.1.2 Tutorial

Das Tutorial ist eine Anleitung wie Sie in der TwinCAT SPS eine IEC60870-5-101 Unterstation (Slave) implementieren und konfigurieren können. Das Beispiel kann [hier \[► 594\]](#) heruntergeladen werden.

#### 6.2.2.1.2.1 Download: Unterstation Tutorial (slave, unbalanced mode)

Eine komplexere Unterstation-Applikation mit unterschiedlichen Datenpunkten in beide Richtungen und einer Verbindung zur Zentralstation. Die Beispielapplikation verwendet das TwinCAT IEC 60870-5-101 High-Level Interface.

Download TwinCAT XAE Project (\*.zip): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/984615819.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/984615819.zip)

Im Beispielprojekt sind folgende Stationsparameter für die Unterstation eingestellt:

- Standard PC interface: **COM1**.
- Baudrate: **19200 Baud**.
- Link address: **220**.
- Link address size: **2 octets**.
- Cause of transfer size: **2 octets (inclusive originator address)**.
- Originator address: **1**.
- Common ASDU address: **7**.
- Common ASDU address size: **2 octets**.
- Information object address size: **3 octets**.
- Link mode: **Unbalanced**.

Datenpunktconfiguration (sie sollte gleich sein mit der Datenpunktconfiguration der Zentralstation):

- Single point information: M\_SP\_NA\_1 (IOA := 100), M\_SP\_TA\_1 (IOA := 101), M\_SP\_TB\_1 (IOA := 102).
- Double point information: M\_DP\_NA\_1 (IOA := 200), M\_DP\_TA\_1 (IOA := 201), M\_DP\_TB\_1 (IOA := 202).
- Regulating step value: M\_ST\_NA\_1 (IOA := 300), M\_ST\_TA\_1 (IOA := 301), M\_ST\_TB\_1 (IOA := 302).
- Bitstring value: M\_BO\_NA\_1 (IOA := 400), M\_BO\_TA\_1 (IOA := 401), M\_BO\_TB\_1 (IOA := 402).
- Measured value, normalized value: M\_ME\_NA\_1 (IOA := 500), M\_ME\_TA\_1 (IOA := 501), M\_ME\_TD\_1 (IOA := 502).
- Measured value, scaled value: M\_ME\_NB\_1 (IOA := 600), M\_ME\_TB\_1 (IOA := 601), M\_ME\_TD\_1 (IOA := 602).
- Measured value, short floating point value: M\_ME\_NC\_1 (IOA := 700), M\_ME\_TC\_1 (IOA := 701), M\_ME\_TF\_1 (IOA := 702).
- Binary counter value: M\_IT\_NA\_1 (IOA := 800), M\_IT\_TA\_1 (IOA := 801), M\_IT\_TB\_1 (IOA := 802).
- Single command: C\_SC\_NA\_1 (IOA := 10), C\_SC\_TA\_1 (IOA := 11), C\_SC\_TB\_1 (IOA := 12).
- Double command: C\_DC\_NA\_1 (IOA := 20), C\_DC\_TA\_1 (IOA := 21), C\_DC\_TB\_1 (IOA := 22).
- Bitstring command: C\_BO\_NA\_1 (IOA := 40), C\_BO\_TA\_1 (IOA := 41), C\_BO\_TB\_1 (IOA := 42).
- Setpoint command, normalized value: C\_SE\_NA\_1 (IOA := 50), C\_SE\_TA\_1 (IOA := 51), C\_SE\_TB\_1 (IOA := 52).
- Setpoint command, scaled value: C\_SE\_NB\_1 (IOA := 60), C\_SE\_TB\_1 (IOA := 61), C\_SE\_TB\_1 (IOA := 62).
- Setpoint command, short floating point value: C\_SE\_NC\_1 (IOA := 70), C\_SE\_NC\_1 (IOA := 71), C\_SE\_TC\_1 (IOA := 72).

Beispielimplementierung folgender Dienste:

- Ende der Initialisierung (M\_EI\_NA\_1) wird einmalig gesendet.

- Testbefehl (C\_TS\_NA\_1).
- Uhrzeitsynchronisationsbefehl (C\_CS\_NA\_1).
- Generalabfragebefehl (C\_IC\_NA\_1) der Gruppe: Allgemein (eIEC870\_QOI\_INROGEN).
- Zählerabfragebefehl (C\_IC\_NA\_1) der Gruppe: Allgemein (eIEC870\_RQT\_REQCOGEN, Mode A). Zähler werden lokal in der Unterstation alle 15 Sekunden umgespeichert und als spontane Daten an die Zentralstation übertragen.
- Befehlsausführung einiger Befehle (direkte Befehle, siehe obere Datenpunktkonfiguration).
- Simulation und Übertragung der Spontanen Daten alle 10 Sekunden wenn die Variable **bSendSpont** auf TRUE gesetzt wurde (siehe obere Datenpunktkonfiguration).

Folgende Tabelle beschreibt kurz die Hauptkomponenten des Projekts:

Komponente	Typ	Beschreibung
P_MAIN_LowSpeed	PRG	Dieses Programmteil implementiert das Fernwirkprotokoll (langsame Task).
P_SerialComm_HighSpeed	PRG	Dieses Programmteil implementiert die TwinCAT Hintergrundkommunikation über die seriellen Schnittstellen des PCs (schnelle Task).
fbServer	FB_IEC870_5_101Slave (FB)	Implementiert die Unterstation. Eine Instanz dieses Bausteins kann nur mit einer Zentralstation kommunizieren. Für jede weitere Zentralstation mit der kommuniziert werden soll wird eine weitere Instanz dieses Funktionsbausteins benötigt.
AODB	ARRAY OF ST_IEC870_5_101AO DBEntry (STRUCT)	Applikationsobjektdatenbank. Es ist eine Array-Variable. Für jeden Datenpunkt wird ein Arrayelement benötigt.
F_iecInitAOEntry	FUNCTION	Im Initialisierungsschritt werden die einzelnen Datenpunkte der Station mit diesem Funktionsaufruf einmalig konfiguriert.
memory	ST_MemoryData (STRUCT)	Diese Struktur beinhaltet die empfangenen und die zu sendenden Prozessdaten die ständig aktualisiert werden.
F_iecSetAOQuality	FUNCTION	Diese Funktion demonstriert wie die Quality-Flags von M_SP_NA_1(IOA := 100) gesetzt/zurückgesetzt werden können.
F_iecGetAOQuality	FUNCTION	Diese Funktion demonstriert wie die Quality-Flags von M_SP_NA_1(IOA := 100) abgefragt werden können.

### 6.2.2.1.2.2 Glossar

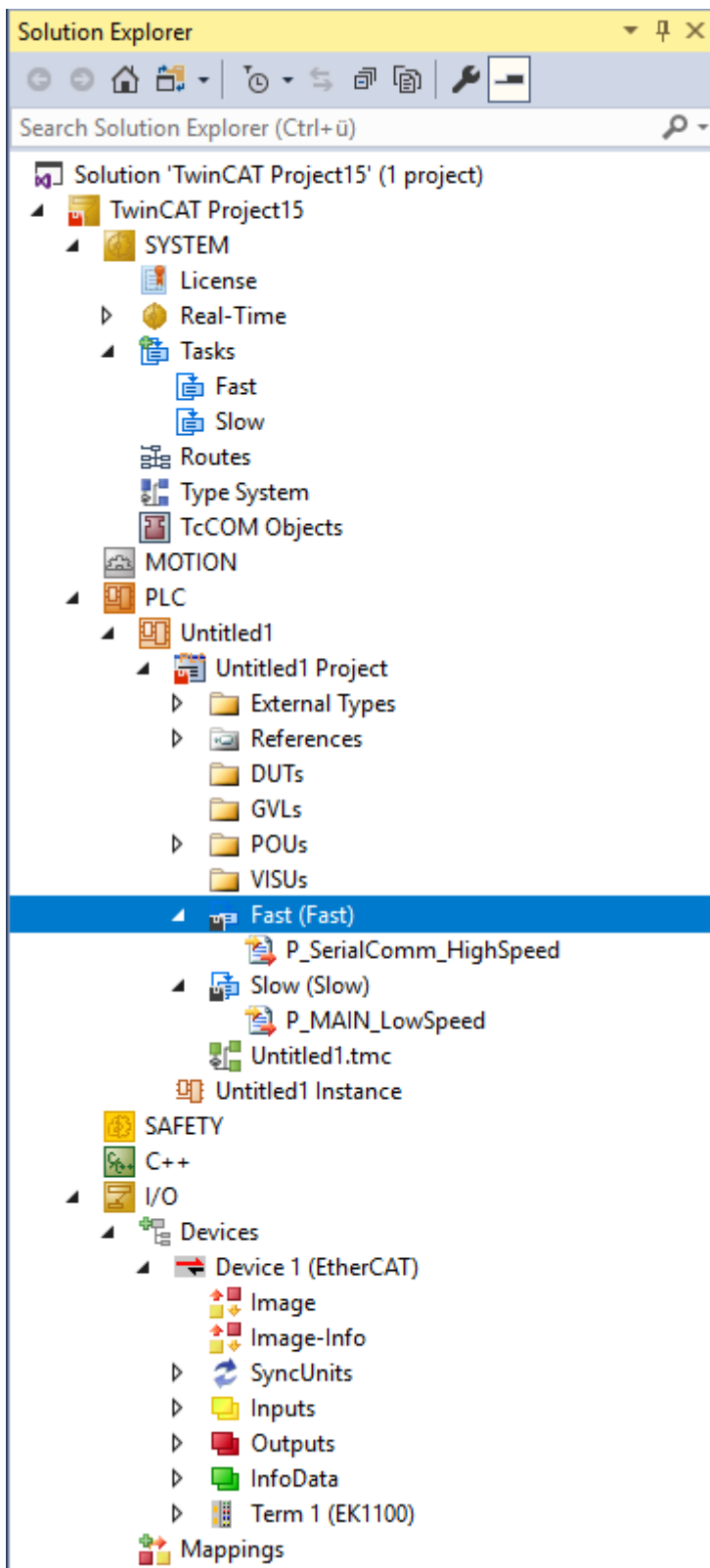
Begriff	Beschreibung
Unterstation, Slave, Server	Synonyme für eine untergeordnete Station (sie wird überwacht)
Zentralstation, Leitstation, Master, Client	Synonyme für eine übergeordnete Station (Leitstation, sie überwacht andere Stationen)
Steuerungsrichtung	Datenübertragungsrichtung von der Zentralstation zur Unterstation
Überwachungsrichtung	Datenübertragungsrichtung von der Unterstation zur Zentralstation
Applikationsobjekte	IEC-Informationsobjekte in der TwinCAT SPS-Applikation (Single Points, Double Points, Measured Values, Short Floating Point Values usw. )
APDU	Protokolldateneinheit der Anwendungsschicht (application protocol data unit)
APCI	Protokollsteuerinformation der Anwendungsschicht (application protocol control information)
ASDU	Dienstdateneinheit der Anwendungsschicht (application service data unit)
IOA, Adresse des Informationsobjekts	Adresse des Single Points, Double Points usw. (information object address)
Primärstation	Die Primärstation sendet Befehle (Anfragen) an die Sekundärstation und steuert/kontrolliert so die Datenübertragung der Sekundärstation.
Sekundärstation	Die Sekundärstation antwortet auf die Anfragen der Primärstation.
Kombinierte Station	Kombinierte Stationen können die Rolle der Primärstation und Sekundärstation annehmen (symmetrische Übertragung, balanced mode).
Symmetrische Übertragung (balanced mode)	Beide Stationen können als Primärstation oder Sekundärstation agieren und die Datenübertragung initialisieren.
Unsymmetrische Übertragung (unbalanced mode)	Die Datenübertragung wird immer durch die Primärstation initiiert und gesteuert. Die Zentralstation agiert immer als Primärstation und die Unterstation als Sekundärstation.

### 6.2.2.1.2.3 SPS-Projekt anlegen, SPS-Bibliotheken einbinden

✓ Starten Sie TwinCAT XAE.

1. Mit **File > New > Project** öffnet sich ein Wizard. In diesem wählen Sie **TwinCAT Project** und das Template **TwinCAT XAE Project** aus.
2. Gehen Sie auf den PLC Knoten im Solution Explorer und wählen Sie über das Kontextmenu **Add New Item**. Wählen Sie im Wizard ein **Standard PLC Project** aus.
3. Neben der automatisch angelegten MAIN, fügen Sie einen weiteren Programmierbaustein hinzu. Gehen Sie auf den Ordner **POU** und von dort über das Kontextmenu auf **Add POU**. Wählen Sie den Typ **Program** aus und nennen es *P\_MAIN\_LowSpeed* .
4. Fügen Sie einen weiteren Programmbaustein hinzu und nennen diesen *P\_SerialComm\_HighSpeed*.

5. Konfigurieren Sie in der Taskkonfiguration unter System 2 Tasks, eine schnelle (T#1ms) und eine langsame (T#10ms). Ordnen Sie den Programmbaustein *P\_MAIN\_HighSpeed* der schnellen Task und den *P\_MAIN\_LowSpeed* der langsamen Task zu (siehe Bild).



6. Im PLC Project Knoten ist die Bibliotheksverwaltung im Unterknoten References zu finden. Dort wählen Sie im Kontextmenu über **Add Library** die Bibliothek **Tc2\_IEC60870\_5\_10x.compiled-library** aus und bestätigen dies.  
 ⇒ Nun sind alle Voraussetzungen erfüllt, um das Tutorial schrittweise in Konfiguration und Programmierung durchzuarbeiten.

### 6.2.2.1.2.4 Die schnelle SPS-Task

Fügen Sie im Deklarationsteil folgenden SPS-Code hinzu:

```
PROGRAM P_SerialComm_HighSpeed
VAR
    fbSerialLineCtrl    : FB_IEC870_SerialLineCtrl := (
        Mode             := SERIALLINEMODE_PC_COM_PORT, (*SERIALLINEMODE_KL6_5B_STANDARD *)
        Baudrate         := 19200,
        NoDatabits       := 8,
        Parity           := PARITY_EVEN,
        Stopbits         := 1,
        Handshake        := HANDSHAKE_NONE,
        ContinousMode    := FALSE );

    serial_in           AT%IB4000           : PcComInData;
    serial_out          AT%QB4000           : PcComOutData;

    KL6_in             AT%IB4100           : KL6inData5B;
    KL6_out            AT%QB4100           : KL6outData5B;

    hSerial            : T_HSERIALCTRL;
END_VAR
```

und im Programmcode:

```
fbSerialLineCtrl( pComIn := SEL( fbSerialLineCtrl.Mode = SERIALLINEMODE_KL6_5B_STANDARD,
ADR( serial_in ), ADR( KL6_in ) ),
pComOut := SEL( fbSerialLineCtrl.Mode = SERIALLINEMODE_KL6_5B_STANDARD, ADR(
serial_out ), ADR( KL6_out ) ),
SizeComIn := SEL( fbSerialLineCtrl.Mode = SERIALLINEMODE_KL6_5B_STANDARD,
SIZEOF( serial_in ), SIZEOF( KL6_in ) ),
hSerial := hSerial );
```

Eine zu diesem Beispiel passende TwinCAT System Manager Konfiguration finden Sie auf der Beispiele-Übersichtsseite. Die Mode-Variable kann dazu verwendet werden um zwischen zwei Kommunikationswegen umzuschalten.

#### Kommunikation über die Standard PC COMx-Schnittstellen

- In diesem Fall wird der Mode-Parameter auf den Wert: **SERIALLINEMODE\_PC\_COM\_PORT** gesetzt.
- Im TwinCAT System Manager werden die *serial\_in*- und *serial\_out*- Variablen mit den entsprechenden IO-Variablen der seriellen Schnittstelle verknüpft.
- Die Schnittstelle wird und muss im TwinCAT System Manager konfiguriert werden (Baudrate, Parity usw.). Andere Kommunikationsparameter an dem FB\_IEC870\_SerialLineCtrl-Funktionsbaustein sind in diesem Mode irrelevant.

#### Kommunikation über die seriellen Beckhoff Busklemmen KL6xxx

- In diesem Fall wird der Mode-Parameter auf den Wert: **SERIALLINEMODE\_KL6\_5B\_STANDARD** gesetzt.
- Im TwinCAT System Manager werden die *KL6\_in* - und *KL6\_out* - Variablen mit den entsprechenden I/O-Variablen der seriellen Klemme KL6xxx verknüpft.
- Die Schnittstelle wird in der TwinCAT SPS durch die Instanz des FB\_IEC870\_SerialLineCtrl-Funktionsbausteins konfiguriert. Die Kommunikationsparameter wie Baudrate, Parity usw. sind an diesem Baustein einzustellen.

#### Kommunikation über die seriellen Beckhoff Busklemmen EL6xxx

```
PROGRAM P_SerialComm_HighSpeed
VAR
    fbSerialLineCtrl : FB_IEC870_SerialLineCtrl := ( Mode := SERIALLINEMODE_EL6_22B );

    EL6_in AT%IB4100 : EL6inData22B;
    EL6_out AT%QB4100 : EL6outData22B;
    hSerial : T_HSERIALCTRL;
END_VAR

fbSerialLineCtrl( pComIn := ADR( EL6_in ),
pComOut := ADR( EL6_out ),
SizeComIn := SIZEOF( EL6_in ),
hSerial := hSerial );
```

- In diesem Fall wird der Mode-Parameter auf den Wert: **SERIALLINEMODE\_EL6\_22B** gesetzt.
- Im TwinCAT System Manager werden die *EL6\_in* - und *EL6\_out* - Variablen mit den entsprechenden I/O-Variablen der seriellen Klemme EL6xxx verknüpft.
- Die Schnittstelle wird und muss im TwinCAT System Manager konfiguriert werden (Baudrate, Parity usw.). Andere Kommunikationsparameter an dem FB\_IEC870\_SerialLineCtrl-Funktionsbaustein sind in diesem Mode irrelevant.

#### 6.2.2.1.2.5 Applikationsobjekt-Datenbank der Unterstation definieren und konfigurieren

Applikationsobjekte = Single Points, Double Points, Measured Values, Short Floating Point Values usw.

In diesem Beispiel wurden die Befehle so konfiguriert, dass die Prozessdaten der Befehle im gleichen Speicherbereich aber auf einem anderen Byte-/Bit-Offset wie die Daten der Information in Überwachungsrichtung liegen. Sie können aber auch die Befehle auf den gleichen Byte- Bit-Offset wie die Information in Überwachungsrichtung legen.

##### Beispiel:

C\_SC\_NA\_1 mit IOA = 10 auf den gleichen Byte- und Bit-Offset wie M\_SP\_NA\_1 mit IOA = 100 (beide Byte-Offset = 100 und Bit-Offset = 0). In diesem Fall wird eine Wertänderung durch ein Kommando von der Leitstation eine Übertragung des M\_SP\_NA\_1 mit der Objektadresse 100 und Übertragungsursache <11> (returned by remote command) zur Folge haben.

Als Beispiel konfigurieren wir in dem Einführungsprojekt folgende Applikationsobjekte:

Arrayelement	ASDU identifizier	Objektadresse IOA	Group-Konfigurationsparameter	Basiszeitmultiplikator	SPS-Prozessdatenbereich	Offset Byte	Offset Bit	Prozessdatenbreite in der TwinCAT SPS
0	M_SP_NA_1	100	Generalabfrage	0	Merker	100	0	1 Bit
1	M_SP_NA_1	101	Generalabfrage	0	Merker	100	1	1 Bit
2	M_SP_TB_1	102	Generalabfrage	0	Merker	100	2	1 Bit
3	M_DP_NA_1	200	Generalabfrage	0	Merker	200	0	2 Bits
4	M_DP_NA_1	201	Generalabfrage	0	Merker	200	2	2 Bits
5	M_DP_TB_1	202	Generalabfrage	0	Merker	200	4	2 Bits
6	M_ST_NA_1	300	Generalabfrage	0	Merker	300	0	1 Byte
7	M_ST_NA_1	301	Generalabfrage	0	Merker	301	0	1 Byte
8	M_ST_TB_1	302	Generalabfrage	0	Merker	302	0	1 Byte
9	M_BO_NA_1	400	Generalabfrage	0	Merker	400	0	4 Byte
10	M_BO_NA_1	401	Generalabfrage	0	Merker	404	0	4 Byte
11	M_BO_TB_1	402	Generalabfrage	0	Merker	408	0	4 Byte
12	M_ME_NA_1	500	Generalabfrage	0	Merker	500	0	2 Byte
13	M_ME_NA_1	501	Generalabfrage	0	Merker	502	0	2 Byte
14	M_ME_TD_1	502	Generalabfrage	0	Merker	504	0	2 Byte
15	M_ME_NB_1	600	Generalabfrage	0	Merker	600	0	2 Byte
16	M_ME_NB_1	601	Generalabfrage	0	Merker	602	0	2 Byte
17	M_ME_TE_1	602	Generalabfrage	0	Merker	604	0	2 Byte
18	M_ME_NC_1	700	Generalabfrage	0	Merker	700	0	4 Byte
19	M_ME_NC_1	701	Generalabfrage	0	Merker	704	0	4 Byte
20	M_ME_TF_1	702	Generalabfrage	0	Merker	708	0	4 Byte



Arrayelement	ASDU identifizier	Objektadresse IOA	Group-Konfigurationsparameter	Basiszeitmultiplikator	SPS-Prozessdatenbereich	Offset Byte	Offset Bit	Prozessdatenbreite in der TwinCAT SPS
21	M_IT_NA_1	800	Generalzählerabfrage und Mode A (lokal Umspeichern mit Spontanübertragung alle 15s)	0	Merker	800	0	4 Byte
22	M_IT_NA_1	801	Generalzählerabfrage und Mode A (lokal Umspeichern mit Spontanübertragung alle 15s)	0	Merker	804	0	4 Byte
23	M_IT_TB_1	802	Generalzählerabfrage und Mode A (lokal Umspeichern mit Spontanübertragung alle 15s)	0	Merker	808	0	4 Byte
<b>Commands</b>								
24	C_SC_NA_1	10	-	0	Merker	2100	0	1 Bit
25	C_SC_NA_1	11	-	0	Merker	2100	1	1 Bit
26	C_SC_TA_1	12	-	0	Merker	2100	2	1 Bit
27	C_DC_NA_1	20	-	0	Merker	2200	0	2 Bit
28	C_DC_NA_1	21	-	0	Merker	2200	2	2 Bit
29	C_DC_TA_1	22	-	0	Merker	2200	4	2 Bit
30	C_RC_NA_1	30	-	0	Merker	2300	0	1 Byte
31	C_RC_NA_1	31	-	0	Merker	2301	0	1 Byte
32	C_RC_TA_1	32	-	0	Merker	2302	0	1 Byte
33	C_BO_NA_1	40	-	0	Merker	2400	0	4 Byte

Arrayelement	ASDU-identifizier	Objekt-adresse IOA	Group-Konfigurationsparameter	Basiszeit-multiplikator	SPS-Prozessdatenbereich	Offset Byte	Offset Bit	Prozessdatenbreite in der TwinCAT SPS
34	C_BO_NA_1	41	-	0	Merker	2404	0	4 Byte
35	C_BO_TA_1	42	-	0	Merker	2408	0	4 Byte
36	C_SE_NA_1	50	-	0	Merker	2500	0	2 Byte
37	C_SE_NA_1	51	-	0	Merker	2502	0	2 Byte
38	C_SE_TA_1	52	-	0	Merker	2504	0	2 Byte
39	C_SE_NB_1	60	-	0	Merker	2600	0	2 Byte
40	C_SE_NB_1	61	-	0	Merker	2602	0	2 Byte
41	C_SE_TB_1	62	-	0	Merker	2604	0	2 Byte
42	C_SE_NC_1	70	-	0	Merker	2700	0	4 Byte
43	C_SE_NC_1	71	-	0	Merker	2704	0	4 Byte
44	C_SE_TC_1	72	-	0	Merker	2708	0	4 Byte

### Datenbankvariable deklarieren

Die Applikationsobjekt-Datenbank ist eine Array-Variable vom Typ [ST\\_IEC870\\_5\\_101AODBEntry](#) [► 482]

Jedes Array-Element entspricht einem Applikationsobjekt. Die maximale Anzahl der Applikationsobjekte ist frei wählbar und nur durch den verfügbaren Speicher begrenzt. Sie müssen sich auf eine konstante maximale Anzahl während der SPS-Programmierung festlegen. Zur Laufzeit kann die maximale Anzahl der Applikationsobjekte nicht mehr verändert werden.

In unserem Beispiel werden 50 Applikationsobjekte deklariert. Diese Anzahl reicht für die meisten Anwendungen aus. Beachten Sie, dass sehr viele Applikationsobjekte auch entsprechend viel Speicher und Laufzeit benötigen.

Definieren Sie folgende Variable in MAIN:

```
PROGRAM MAIN
VAR
    AODB : ARRAY[0..49] OF ST_IEC870_5_101AODBEntry;
END_VAR
```

### Applikationsobjekte konfigurieren

Während der Konfiguration der einzelnen Applikationsobjekte werden unter anderem der Objekt-Typ (M\_SP\_NA\_1, M\_DP\_NA\_1, M\_ST\_NA\_1 usw.), die Objekt-Adresse und weitere Objekt-Parameter festgelegt.

Die Konfiguration der gewünschten Applikationsobjekte wird zur Programmlaufzeit durchgeführt. Jedes Applikationsobjekt (Datenbank-Array-Element) wird durch einen einmaligen Aufruf der [F\\_iecInitAOEntry](#) [► 435] Funktion konfiguriert. Das zu konfigurierende Array-Element wird an die Funktion per VAR\_IN\_OUT

übergeben. Im Regelfall wird die Konfiguration beim SPS-Programmstart einmalig in einer Init-Routine durchgeführt. Die Funktion [F\\_iecInitAOEntry](#) [► 435] erwartet folgende Funktionsparameter (von links nach rechts):

```
FUNCTION F_iecInitAOEntry : UDINT
VAR_INPUT
    eType      : E_IEC870_5_101TcTypeID;
    objAddr    : DWORD := 0;
    group      : DWORD := 0;
    multiplier  : BYTE := 0;
    ioMapType  : E_IEC870_5_;
    byteOffs   : UDINT := 0;
    bitOffs    : UDINT := 0;
END_VAR
VAR_IN_OUT
    dbEntry    : ST_IEC870_5_101AODBEntry;
END_VAR
```

**eType:** Applikationsobjekt-Typ (ASDU identifier, z.B.: M\_SP\_NA\_1 für Single Point oder M\_DP\_NA\_1 für Double Point ). Beachten Sie, dass nur die in der Kompatibilitätsliste aufgeführten ASDU-Typen verwendet werden können. Unzulässige Typen werden ignoriert.

**objAddr :** Objektadresse, z.B. 100. Jedes Applikationsobjekt sollte mit einer eindeutigen Adresse konfiguriert werden.

**group:** Group-Konfigurationsparameter. Die verfügbaren Group-Parameter sind als Konstanten definiert und können mit ODER-Operator kombiniert werden. Z.B.: IEC870\_GRP\_INROGEN OR IEC870\_GRP\_PERCYC.

Hier finden Sie die Beschreibung aller [Group-Konfigurationsparameter](#) [► 554]

**multiplier:** Basiszeitmultiplikator für die zyklische/periodische Datenübertragung ( 0=Deaktiviert). Die Basiszeit wird über die Systemparameter konfiguriert. Wurde die Basiszeit z.B. auf T#10s gesetzt und der Multiplikator auf den Wert 2, dann werden die periodischen/zyklischen Daten des Applikationsobjekts alle 20 Sekunden gesendet.

**ioMapType:** Dieser Parameter legt fest, aus oder in welchen Prozessdatenbereich der TwinCAT SPS die IEC-Prozessdaten zur Laufzeit gemappt werden sollen (inputs, outputs, memory, data).

**byteOffs:** Prozessdatenbereich Byteoffset;

**bitOffs:** Prozessdatenbereich Bitoffset;

**dbEntry:** Applikationsobjekt das konfiguriert werden soll (ein Datenbankvariable-Arrayelement, das an die Funktion per VAR\_IN\_OUT übergeben wird).

Um die Applikationsobjekte beim Programmstart zu konfigurieren wird in MAIN folgender SPS-Code hinzugefügt:

```
PROGRAM MAIN
VAR
    AODB          : ARRAY[0..49] OF ST_IEC870_5_101AODBEntry;
    init          : BOOL := TRUE;
    initError     : UDINT;
END_VAR

IF init THEN
    init := FALSE;

    (* Monitored Single Points *)
    initError := F_iecInitAOEntry( M_SP_NA_1, 100, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 100, 0, AODB
[0] );
    initError := F_iecInitAOEntry( M_SP_NA_1, 101, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 100, 1, A
ODB[1] );
    initError := F_iecInitAOEntry( M_SP_TB_1, 102, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 100, 2, A
ODB[2] );
    (* Double Points*)
    initError := F_iecInitAOEntry( M_DP_NA_1, 200, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 200, 0, AODB
[3] );
    initError := F_iecInitAOEntry( M_DP_NA_1, 201, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 200, 2, A
ODB[4] );
    initError := F_iecInitAOEntry( M_DP_TB_1, 202, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 200, 4, A
ODB[5] );
    (* Regulating step value *)
    initError := F_iecInitAOEntry( M_ST_NA_1, 300, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 300, 0, AODB
[6] );
```

```

    initError := F_iecInitAOEntry( M_ST_NA_1, 301, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 301, 0, A
ODB[7] );
    initError := F_iecInitAOEntry( M_ST_TB_1, 302, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 302, 0, A
ODB[8] );
    (* 32 bit string*)
    initError := F_iecInitAOEntry( M_BO_NA_1, 400, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 400, 0, AODB
[9] );
    initError := F_iecInitAOEntry( M_BO_NA_1, 401, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 404, 0, A
ODB[10] );
    initError := F_iecInitAOEntry( M_BO_TB_1, 402, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 408, 0, A
ODB[11] );
    (* Measured value, normalized value *)
    initError := F_iecInitAOEntry( M_ME_NA_1, 500, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 500, 0, AODB
[12] );
    initError := F_iecInitAOEntry( M_ME_NA_1, 501, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 502, 0, A
ODB[13] );
    initError := F_iecInitAOEntry( M_ME_TD_1, 502, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 504, 0, A
ODB[14] );
    (* Mesured value, scaled value *)
    initError := F_iecInitAOEntry( M_ME_NB_1, 600, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 600, 0, AODB
[15] );
    initError := F_iecInitAOEntry( M_ME_NB_1, 601, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 602, 0, A
ODB[16] );
    initError := F_iecInitAOEntry( M_ME_TE_1, 602, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 604, 0, A
ODB[17] );
    (* Measured value , short floating point value *)
    initError := F_iecInitAOEntry( M_ME_NC_1, 700, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 700, 0, AODB
[18] );
    initError := F_iecInitAOEntry( M_ME_NC_1, 701, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 704, 0, A
ODB[19] );
    initError := F_iecInitAOEntry( M_ME_TF_1, 702, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 708, 0, A
ODB[20] );
    (* Integrated totals *)
    initError := F_iecInitAOEntry( M_IT_NA_1, 800, IEC870_GRP_REQCOGEN OR IEC870_GRP_LOCFREEZE, 0, MAP_
AREA_MEMORY, 800, 0, AODB[21] );
    initError := F_iecInitAOEntry( M_IT_NA_1, 801, IEC870_GRP_REQCOGEN OR IEC870_GRP_LOCFREEZE, 0, M
AP_AREA_MEMORY, 804, 0, AODB[22] );
    initError := F_iecInitAOEntry( M_IT_TB_1, 802, IEC870_GRP_REQCOGEN OR IEC870_GRP_LOCFREEZE, 0, M
AP_AREA_MEMORY, 808, 0, AODB[23] );

    (* Single commands *)
    initError := F_iecInitAOEntry( C_SC_NA_1, 10, 0, 0, MAP_AREA_MEMORY, 2100, 0, AODB[24] );
    initError := F_iecInitAOEntry( C_SC_NA_1, 11, 0, 0, MAP_AREA_MEMORY, 2100, 1, AODB[25] );
    initError := F_iecInitAOEntry( C_SC_TA_1, 12, 0, 0, MAP_AREA_MEMORY, 2100, 2, AODB[26] );
    (* Double commands *)
    initError := F_iecInitAOEntry( C_DC_NA_1, 20, 0, 0, MAP_AREA_MEMORY, 2200, 0, AODB[27] );
    initError := F_iecInitAOEntry( C_DC_NA_1, 21, 0, 0, MAP_AREA_MEMORY, 2200, 2, AODB[28] );
    initError := F_iecInitAOEntry( C_DC_TA_1, 22, 0, 0, MAP_AREA_MEMORY, 2200, 4, AODB[29] );
    (* Regulating step commands *)
    initError := F_iecInitAOEntry( C_RC_NA_1, 30, 0, 0, MAP_AREA_MEMORY, 2300, 0, AODB[30] );
    initError := F_iecInitAOEntry( C_RC_NA_1, 31, 0, 0, MAP_AREA_MEMORY, 2301, 0, AODB[31] );
    initError := F_iecInitAOEntry( C_RC_TA_1, 32, 0, 0, MAP_AREA_MEMORY, 2302, 0, AODB[32] );
    (* 32 bit string commands *)
    initError := F_iecInitAOEntry( C_BO_NA_1, 40, 0, 0, MAP_AREA_MEMORY, 2400, 0, AODB[33] );
    initError := F_iecInitAOEntry( C_BO_NA_1, 41, 0, 0, MAP_AREA_MEMORY, 2404, 0, AODB[34] );
    initError := F_iecInitAOEntry( C_BO_TA_1, 42, 0, 0, MAP_AREA_MEMORY, 2408, 0, AODB[35] );
    (* Set point, normalized values*)
    initError := F_iecInitAOEntry( C_SE_NA_1, 50, 0, 0, MAP_AREA_MEMORY, 2500, 0, AODB[36] );
    initError := F_iecInitAOEntry( C_SE_NA_1, 51, 0, 0, MAP_AREA_MEMORY, 2502, 0, AODB[37] );
    initError := F_iecInitAOEntry( C_SE_TA_1, 52, 0, 0, MAP_AREA_MEMORY, 2504, 0, AODB[38] );
    (* Set point, scaled values *)
    initError := F_iecInitAOEntry( C_SE_NB_1, 60, 0, 0, MAP_AREA_MEMORY, 2600, 0, AODB[39] );
    initError := F_iecInitAOEntry( C_SE_NB_1, 61, 0, 0, MAP_AREA_MEMORY, 2602, 0, AODB[40] );
    initError := F_iecInitAOEntry( C_SE_TB_1, 62, 0, 0, MAP_AREA_MEMORY, 2604, 0, AODB[41] );
    (* Set point, short floating point values *)
    initError := F_iecInitAOEntry( C_SE_NC_1, 70, 0, 0, MAP_AREA_MEMORY, 2700, 0, AODB[42] );
    initError := F_iecInitAOEntry( C_SE_NC_1, 71, 0, 0, MAP_AREA_MEMORY, 2704, 0, AODB[43] );
    initError := F_iecInitAOEntry( C_SE_TC_1, 72, 0, 0, MAP_AREA_MEMORY, 2708, 0, AODB[44] );

END_IF

```

### 6.2.2.1.2.6 Mapping der SPS- und IEC-Prozessdaten

Die TwinCAT SPS-Prozessdaten werden zur Programmlaufzeit zyklisch in die IEC-Prozessdaten (Applikationsobjekte) und umgekehrt gemappt (kopiert). Für das Mapping der IEC<->SPS Prozessdaten können bis zu 4 Prozessdatenbereiche (IO-Eingänge, IO-Ausgänge, Merkerbereich, Datenbereich) als

Puffervariablen im SPS-Programm deklariert werden. Die Bytegröße der Puffer ist frei wählbar und kann für jeden Bereich unterschiedlich gewählt werden. Unbenutzte Bereiche müssen nicht unbedingt deklariert werden.

In unserem Einführungsbeispiel deklarieren wir 4 SPS-Prozessdatenbereiche mit jeweils 3000 Bytes:

```
PROGRAM MAIN
VAR
  AODB : ARRAY[0..49] OF ST_IEC870_5_101AODBEntry;

  init : BOOL := TRUE;
  initError : UDINT;

  inputs AT%IB0 : ARRAY[0..2999] OF BYTE;
  outputs AT%QB0 : ARRAY[0..2999] OF BYTE;
  memory AT%MB0 : ARRAY[0..2999] OF BYTE;
  data : ARRAY[0..2999] OF BYTE;

END_VAR
```

Die Zuordnung, wie die Prozessdaten zur Laufzeit gemappt werden sollen, wird während der Konfiguration der Applikationsobjekte mit der F\_ieclnitAOEntry-Funktion festgelegt.

Siehe auch in: [Applikationsobjekte definieren und konfigurieren \[► 435\]](#).

Die Puffervariablen wurden nun als Byte-Arrays deklariert. Um auf die gewünschten Daten besser zugreifen zu können definieren wir die einzelnen Variablen ein zweites Mal und legen diese auf die entsprechenden Byte/Bit-Offsetadressen. Bei einer Änderung im Byte-Array wird die entsprechende einzelne Variable gleichzeitig geändert und umgekehrt. Dies ist aber nicht zwingend notwendig. Sie können direkt auf die Bytes/Bits der Byte-Array-Puffervariablen zugreifen.

```
VAR_GLOBAL(* Memory offset 0..99 unused *)
(* Single points *)
msgSingle_0 AT%MX100.0 : BOOL;
msgSingle_1 AT%MX100.1 : BOOL;
msgSingle_2 AT%MX100.2 : BOOL;

(* Double points *)
(* Bit 0..1 = first double point,
   Bit 2..3 = second double point,
   Bit 4..5 = third double point,
   Bit 6..7 = fourth double point *)
msgDouble_0 AT%MB200 : BYTE;

(* Regulating step values *)
msgStep_0 AT%MB300 : BYTE;
msgStep_1 AT%MB301 : BYTE;
msgStep_2 AT%MB302 : BYTE;

(* 32 bit strings *)
msgBitStr_0 AT%MD400 : DWORD := 2#10001000_10001000_10001000_10001000;
msgBitStr_1 AT%MD404 : DWORD := 2#10001000_10001000_10001000_10001000;
msgBitStr_2 AT%MD408 : DWORD := 2#10001000_10001000_10001000_10001000;

(* Measured values, normalized values *)
msgNormalized_0 AT%MW500 : WORD;
msgNormalized_1 AT%MW502 : WORD;
msgNormalized_2 AT%MW504 : WORD;

(* Measured values, scaled values *)
msgScaled_0 AT%MW600 : INT;
msgScaled_1 AT%MW602 : INT;
msgScaled_2 AT%MW604 : INT;

(* Measured values, short floating point values *)
msgFloating_0 AT%MD700 : REAL;
msgFloating_1 AT%MD704 : REAL;
msgFloating_2 AT%MD708 : REAL;

(* Integrated totals *)
msgTotal_0 AT%MD800 : UDINT;
msgTotal_1 AT%MD804 : UDINT;
msgTotal_2 AT%MD808 : UDINT;

(*#####*)
(* Single commands *)
cmdSingle_0 AT%MX2100.0 : BOOL;
cmdSingle_1 AT%MX2100.1 : BOOL;
```

```

cmdSingle_2      AT%MX2100.2 : BOOL;

(* Double commands *)
(*      Bit 0..1 = first double command,
      Bit 2..3 = second double command,
      Bit 4..5 = third double command,
      Bit 6..7 = fourth double command *)
cmdDouble_0     AT%MB2200      : BYTE;

(* Regulating step commands *)
cmdStep_0       AT%MB2300      : BYTE;
cmdStep_1       AT%MB2301      : BYTE;
cmdStep_2       AT%MB2302      : BYTE;

(* 32 bit string commands *)
cmdBitStr_0     AT%MD2400      : DWORD;
cmdBitStr_1     AT%MD2404      : DWORD;
cmdBitStr_2     AT%MD2408      : DWORD;

(* Set point, normalized values *)
cmdNormalized_0 AT%MW2500      : WORD;
cmdNormalized_1 AT%MW2502      : WORD;
cmdNormalized_2 AT%MW2504      : WORD;

(* Set point, scaled values *)
cmdScaled_0     AT%MW2600      : INT;
cmdScaled_1     AT%MW2602      : INT;
cmdScaled_2     AT%MW2604      : INT;

(* Set point, short floating point values *)
cmdFloating_0   AT%MD2700      : REAL;
cmdFloating_1   AT%MD2704      : REAL;
cmdFloating_2   AT%MD2708      : REAL;
END_VAR

```

## Mapping der IEC<->SPS Prozessdaten in der Unterstation

### Prozessdaten in Überwachungsrichtung (Slave->Master information)

#### Beispiel 1

Single point information (M\_SP\_NA\_1) mit der IOA = 100, SPS Merkerbereich, Byteoffset = 100, Bitoffset = 0.

*msgSingle\_0 == memory[100].0 -> Unterstation FB -> ... -> Leitstation*

#### Beispiel 2

Measured value, short floating point value (M\_ME\_NC\_1) mit der IOA = 700, SPS Merkerbereich, Byteoffset = 700, Bitoffset = 0 (bedeutungslos).

*msgFloating\_0 == memory[700..703] -> Unterstation FB -> ... -> Leitstation*

### Prozessdaten in Steuerungsrichtung (Master->Slave commands)

#### Beispiel 1

Single command state (C\_SC\_NA\_1) mit der IOA = 10, SPS Merkerbereich, Byteoffset = 2100, Bitoffset = 0.

*Leitstation -> ... -> Unterstation FB -> memory[2100].0 == cmdSingle\_0*

#### Beispiel 2

Set point, short floating point value (C\_SE\_NC\_1) mit der IOA = 70, SPS Merkerbereich, Byteoffset = 2700, Bitoffset = 0 (bedeutungslos).

*Leitstation -> ... -> Unterstation FB -> memory[2700..2703] == cmdFloating\_0*

## 6.2.2.1.2.7 Instanz der IEC60870-5-101 Unterstation deklarieren und aufrufen

Hier [[▶ 594](#)] können Sie die kompletten SPS-Sourcen herunterladen.

Die gesamte Funktionalität einer Unterstation ist im Funktionsbaustein FB\_IEC870\_5\_101Slave gekapselt. Mit einer Instanz kann eine Verbindung zum Master aufgebaut werden. Das *hSerial*-Verbindungshandle der schnellen Task muss als VAR\_IN\_OUT-Variable an die Unterstation übergeben werden.

Fügen Sie im Deklarationsteil von *P\_MAIN\_LowSpeed* folgenden SPS-Code ein:

```
PROGRAM P_MAIN_LowSpeed
VAR
  AODB          : ARRAY[0..49] OF ST_IEC870_5_101AODBEntry;

  init          : BOOL := TRUE;
  error         : UDINT;

  inputs AT%IB0 : ARRAY[0..2999] OF BYTE;
  outputs AT%QB0 : ARRAY[0..2999] OF BYTE;
  memory AT%MB0 : ARRAY[0..2999] OF BYTE;
  data          : ARRAY[0..2999] OF BYTE;

  bEnable       : BOOL := TRUE;
  server        : FB_IEC870_5_101Slave;
END_VAR
```

und im Programmteil wird die Instanz aufgerufen:

```
IF init THEN
  init := FALSE;
  ...

ELSE
  ...
  server( pInputs := ADR( inputs ), cbInputs := SIZEOF( inputs ), pOutputs := ADR( outputs ),
          cbOutputs := SIZEOF( outputs ), pMemory := ADR( memory ), cbMemory := SIZEOF( memory ),
          pData := ADR( data ), cbData := SIZEOF( data ), pAOEntries := ADR( AODB ), cbAOEntries := SIZEOF( AODB ),
          hSerial := P_SerialComm_HighSpeed.hSerial, (* serial link interface connection handle from fast task *)
          bEnable := bEnable );
  ...
END_IF
```

### 6.2.2.1.2.8 IEC60870-5-101-Protokollparameter

Über die IEC60870-5-101-Protokollparameter kann das Verhalten der Unterstation an die Anforderungen des Masters angepasst werden. Die meisten Parameter sind mit Defaultwerten vorbelegt, so dass diese nicht verändert werden müssen.

Im unserem Beispiel setzen wir die Linkadresse und die Bytelänge der Linkadresse:

```
IF init THEN
  init := FALSE;
  ...

  server.protPara.linkAddr := 220; (* slave link address *)
  server.protPara.eLinkAddrSize := eIEC870_LinkAddr_TwoOctets; (* link address octet size *)
  ...

ELSE
  server( pInputs := ADR( inputs ),
          cbInputs := SIZEOF( inputs ),
          pOutputs := ADR( outputs ),
          ...
  ...
END_IF
```

Die Dokumentation aller Übertragungsprotokoll-Parameter finden Sie hier: [ST\\_IEC870\\_5\\_101PotocolParams](#) [► 538]

### 6.2.2.1.2.9 Systemparameter

Über die Systemparameter werden z. B. die gemeinsame ASDU-Adresse und die Anwenderfunktionen der Unterstation konfiguriert.

In unserer Einführung konfigurieren wir folgende Systemparameter:

- Die gemeinsame ASDU-Adresse wird auf 7 gesetzt. (*asduAddr*)
- Die Systemzeit der Unterstation wird während der Initialisierung mit der Systemzeit des lokalen TwinCAT PCs synchronisiert (*bUsePCTime*).
- Die Synchronisierung der Systemzeit der Unterstation mit dem Uhrzeitsynchronisationsbefehl wird aktiviert (*bSyncTime*).
- Während der Synchronisierung der Systemzeit in der Unterstation soll die Systemzeit des TwinCAT PCs nicht synchronisiert werden (*bSyncPCTime*).
- Das Senden von M\_EI\_NA\_1 (End of init) an die Zentralstation wird aktiviert (*bEndOfInit*).
- Das Senden der periodischen/zyklischen Daten wird deaktiviert (*bPerCyclic*). Die Basiszeit fürs Senden dieser Daten wird auf 5s gesetzt.
- Hintergrundabfrage wird deaktiviert (*bBackScan*). Die Zykluszeit für Hintergrundabfrage wird auf 30s gesetzt.
- Das lokale Umspeichern und Reset der Zählerstände wird aktiviert (*bPerFRZ*) und die Zykluszeit fürs Umspeichern und Reset auf 15s gesetzt.
- Das loggen der Debugmeldungen im Application-Log wird aktiviert (*dbgMode*). Es werden Änderungen im Gerätestatus geloggt.

Fügen Sie folgenden SPS-Code in Ihr SPS-Projekt ein:

```
IF init THEN
  init := FALSE;
  ...

  server.sysPara.asduAddr := 7;
  server.sysPara.bUsePCTime := TRUE;
  server.sysPara.bSyncTime := TRUE;
  server.sysPara.bSyncPCTime := FALSE;
  server.sysPara.bEndOfInit := TRUE;
  server.sysPara.bPerCyclic := FALSE;
  server.sysPara.tPerCyclicBase := T#5s;
  server.sysPara.bBackScan := FALSE;
  server.sysPara.tBackScanCycle := T#30s;
  server.sysPara.bPerFRZ := TRUE;
  server.sysPara.tPerFRZCycle := T#15s;
  server.sysPara.dbgMode := (*IEC870_DEBUGMODE_ASDU OR*) IEC870_DEBUGMODE_DEVSTATE;

  ...
ELSE
  server( pInputs := ADR( inputs ),
        cbInputs := SIZEOF( inputs ),
        pOutputs := ADR( outputs ),
        ...
  );
END_IF
```

Die Dokumentation aller Systemparameter finden Sie hier: [ST IEC870 5 101SystemParams \[▶ 487\]](#).

### 6.2.2.1.2.10 Stationsabfrage

Der Stationsabfragebefehl wird von der Zentralstation eingeleitet. Im Kennungsfeld des Befehls ist auch die Gruppe (1 bis 16 oder allgemein) festgelegt. Die Unterstation überträgt die zu dieser Gruppe dazugehörigen Applikationsobjekte mit der Übertragungsursache <20> bis <36> an die Zentralstation. Applikationsobjekte mit Zeitmarken werden ohne Zeitmarken übertragen.

#### Konfiguration der Systemparameter

- Es müssen keine speziellen Systemparameter gesetzt werden;

#### Konfiguration der Applikationsobjekte

- Der Datenpunkt muss einer oder mehreren Gruppen zugeordnet werden. Der Group-Parameter muss gesetzt werden. Eine Übersicht aller verfügbaren Gruppen finden Sie hier: [Group configuration flags \[▶ 554\]](#).

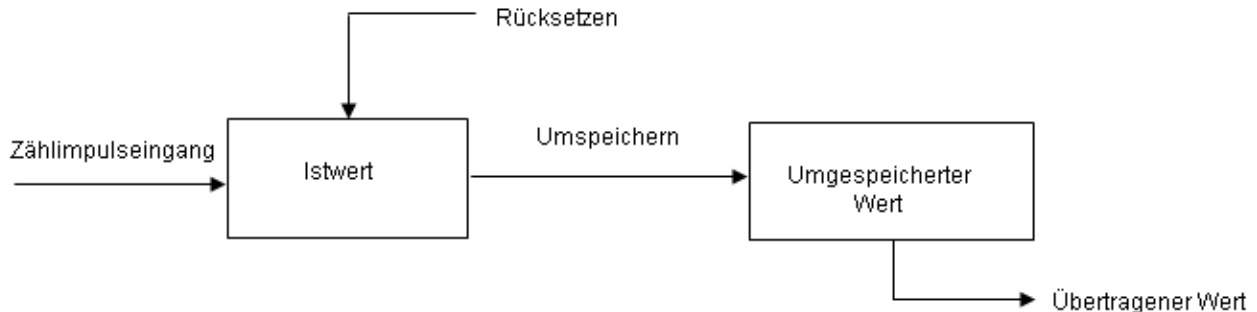
Beispielkonfiguration für einen Datenpunkt der der Gruppe: 1 und der Gruppe: Allgemein zugeordnet wurde.



```
F_iecInitAOEntry( M_SP_NA_1, 100, IEC870_GRP_INROGEN OR IEC870_GRP_INRO1, 0, MAP_AREA_MEMORY, 100, 0, AODB[0] );
```

### 6.2.2.1.2.11 Zählwertübertragung (counter interrogation)

Allgemeines Modell für die Zählwertübertragung:



Die Istwerte werden durch Zähler aufsummiert. Die Istwerte können durch einen Umspeicherbefehl, der entweder von der Zentralstation empfangen oder örtlich (lokal in der Unterstation) erzeugt wird, periodisch in umgespeicherte Werte umgespeichert (kopiert) werden. Nach dem Umspeichern wird der erfasste Wert entweder auf Null zurückgesetzt (Erfassen von Inkrementalwerten) oder der Zähler fährt mit seinem Betrieb fort (Erfassen von Zählerständen).

Applikationsobjekte mit Zählwerten werden Gruppen zugeordnet. Die Gruppen werden einzeln umgespeichert (frozen), zurückgesetzt (reset) oder übertragen. Die Zentralstation sendet Zählwertabfragebefehle an die Unterstation. In einem Kennungsfeld des Befehls (QCC) wird die durchzuführende Aktion (FRZ) und Gruppe (RQT) festgelegt.

Die Zuordnung der Applikationsobjekte zu den einzelnen Gruppen (1 bis 4 oder allgemein) wird während der Konfiguration durch den Group-Flag-Parameter festgelegt. Es gibt vier Betriebsarten für die Erfassung von Zählerständen und Inkrementalwerten. Zu jeder Betriebsart sind einige Hinweise zur Konfiguration der Systemparameter oder der Applikationsobjekte aufgeführt.

#### Betriebsart A: Örtlich Umspeichern mit Spontanübertragung

Die Unterstation initiiert intern das Umspeichern oder Umspeichern mit Rücksetzen. Die umgespeicherten Zählwerte werden spontan übertragen, nachdem die Funktion Umspeichern oder Umspeichern mit Rücksetzen durchgeführt wurde. Die Zentralstation gibt in dieser Betriebsart keine Zählwertabfragebefehle aus.

#### Konfiguration der Systemparameter:

```
bPerFRZ := TRUE
tPerFRZCycle := T#60s
```

Der erste Parameter aktiviert das örtliche Umspeichern oder Umspeichern mit Rücksetzen. Der zweite Parameter gibt die Zykluszeit an in der das umspeichern oder Umspeichern mit Rücksetzen durchgeführt wird (z. B.: alle 60 Sekunden).

#### Konfiguration der Applikationsobjekte:

- IEC870\_GRP\_SPONTOFF-Group-Parameter darf nicht gesetzt werden. Dieser würde die spontane Datenübertragung der Zählwerte verhindern.
- Der Zählwert wird umgespeichert wenn IEC870\_GRP\_LOCFREEZE-Group-Parameter gesetzt wurde.
- Der Zählwert wird zurückgesetzt wenn IEC870\_GRP\_LOCRESET-Group-Parameter gesetzt wurde.
- Das Örtliche Umspeichern oder Umspeichern mit Rücksetzen wird gleichzeitig für alle Gruppen (1 bis 4 oder allgemein) durchgeführt.

### Betriebsart B: Örtliches Umspeichern mit Zählerabfrage

Die Unterstation initiiert intern das Umspeichern oder Umspeichern mit Zurücksetzen. Die umgespeicherten Zählwerte werden durch Zählwertabfragebefehle von der Zentralstation abgefragt. In diesem Fall darf die Zentralstation im Befehlskennungsfeld das Umspeichern oder Umspeichern mit Zurücksetzen nicht benutzen (FRZ=0). Die Zählwerte werden allgemein oder in Gruppen (groups) 1 bis 4 abgefragt.

#### Konfiguration der Systemparameter:

```
bPerFRZ := TRUE
tPerFRZCycle := T#60s
```

Der erste Parameter aktiviert das örtliche Umspeichern oder (und) Zurücksetzen. Der zweite Parameter gibt die Zykluszeit an in der das umspeichern oder Umspeichern mit Zurücksetzen durchgeführt wird (z.B.: alle 60 Sekunden).

#### Konfiguration der Applikationsobjekte:

- IEC870\_GRP\_SPONTOFF-Groupparameter muss gesetzt werden. Die Zählwerte sollen nicht spontan zur Zentralstation übertragen werden.
- Der Zählwert wird umgespeichert wenn IEC870\_GRP\_LOCFREEZE-Group-Parameter gesetzt wurde.
- Der Zählwert wird zurückgesetzt wenn IEC870\_GRP\_LOCRESET-Group-Parameter gesetzt wurde.
- Das örtliche Umspeichern oder Umspeichern mit Zurücksetzen wird gleichzeitig für alle Gruppen (1 bis 4 oder allgemein) durchgeführt.

### Betriebsart C: Zentralstation leitet das Umspeichern, Umspeichern mit Zurücksetzen oder Zurücksetzen ein

Ein Zählwertabfragebefehl wird periodisch von der Zentralstation an die Unterstation ausgegeben, um das Umspeichern oder (und) Zurücksetzen zu steuern. Dieser Befehl hat aber noch keine Übertragung der Zählwerte zur Folge. Erst ein nachfolgender Zählwertabfragebefehl wird von der Zentralstation gesendet um die umgespeicherten Zählwerte einzusammeln. Ähnlich, wie bei der Betriebsart B.

#### Konfiguration der Systemparameter:

```
bPerFRZ := FALSE
tPerFRZCycle := T#60s
```

Das örtliche Umspeichern oder (und) Zurücksetzen muss deaktiviert werden. Der zweite Parameter wird ignoriert.

#### Konfiguration der Applikationsobjekte:

- IEC870\_GRP\_SPONTOFF muss gesetzt werden. Die Zählwerte sollen nicht spontan zur Zentralstation übertragen werden.
- IEC870\_GRP\_LOCFREEZE- und IEC870\_GRP\_LOCRESET-Group-Parameter dürfen nicht gesetzt werden. Die Zentralstation leitet das Umspeichern oder (und) Zurücksetzen ein.
- Die Zählwerte können einzelnen Gruppen (1 bis 4 oder allgemein) zugeordnet und abgefragt werden (IEC870\_GRP\_REQCOGEN, IEC870\_GRP\_REQCO1, IEC870\_GRP\_REQCO2, IEC870\_GRP\_REQCO3, IEC870\_GRP\_REQCO4).

### Betriebsart D: Zentralstation leitet das Umspeichern und (oder) Zurücksetzen ein und die umgespeicherten Werte werden spontan übertragen

Diese Betriebsart ist eine Kombination des Zählwertbefehls von der Zentralstation wie für Betriebsart C mit einer spontanen Übertragung der Zählwerte wie bei der Betriebsart A.

#### Konfiguration der Systemparameter:

```
bPerFRZ := FALSE
tPerFRZCycle := T#60s
```

Das örtliche Umspeichern oder (und) Zurücksetzen muss deaktiviert werden. Der zweite Parameter wird ignoriert.

#### Konfiguration der Applikationsobjekte:

- IEC870\_GRP\_SPONTOFF-Group-Parameter darf nicht gesetzt werden. Dieser würde die spontane Datenübertragung der Zählwerte verhindern.
- IEC870\_GRP\_LOCFREEZE- und IEC870\_GRP\_LOCRESET-Group-Parameter dürfen nicht gesetzt werden. Die Zentralstation leitet das Umspeichern oder (und) Rücksetzen ein.
- Die Zählwerte können einzelnen Gruppen (1 bis 4 oder allgemein) zugeordnet und abgefragt werden (IEC870\_GRP\_REQCOGEN, IEC870\_GRP\_REQCO1, IEC870\_GRP\_REQCO2, IEC870\_GRP\_REQCO3, IEC870\_GRP\_REQCO4).

### 6.2.2.1.2.12 Uhrzeitsynchronisation

Das Verhalten, wie die Systemzeit der Unterstation synchronisiert werden soll, ist über die Systemparameter [► 487] konfigurierbar.

- Die Systemzeit der Unterstation kann während der Initialisierung mit der Systemzeit des lokalen TwinCAT PCs synchronisiert werden;
- Beim Empfang eines Uhrzeit-Synchronisationsbefehls von der Zentralstation kann die Systemzeit der Unterstation ebenfalls synchronisiert werden;
- Die Systemzeit des lokalen TwinCAT PCs kann beim Empfang eines Uhrzeit-Synchronisationsbefehls auch synchronisiert werden.

### 6.2.2.1.2.13 Hintergrundabfrage

Die Hintergrundabfrage wird zum Auffrischen der Prozessinformationen von der Unterstation zur Zentralstation als zusätzlicher Sicherheitsbeitrag zur Stationsabfrage und spontanen Übertragung angewendet.

Applikationsobjekte mit Typkennungen wie für die Stationsabfrage, dürfen mit der Übertragungsursache <2> Hintergrundabfrage stetig mit niedriger Priorität übertragen werden. Die zulässigen ASDU-Typkennungen sind in der Kompatibilitätsliste der Station aufgelistet (Tabelle Typkennung <-> Übertragungsursache). Die Hintergrundabfrage wird von der Unterstation eingeleitet und ist unabhängig von den Stationsabfragebefehlen.

#### Konfiguration der Systemparameter

Der Übertragungszyklus wird durch Systemparameter [► 487] in der Unterstation festgelegt.

```
bBackScan := TRUE;
tBackScanCycle := T#30s;
```

#### Konfiguration der Applikationsobjekte

Applikationsobjekte, deren Prozessdaten als Hintergrundabfrage übertragen werden sollen, müssen mit dem Group-Flag: IEC870\_GRP\_BACKGROUND konfiguriert werden.

Beispiel:

```
F_iecInitAOEntry( M_SP_NA_1, 100, IEC870_GRP_INROGEN OR IEC870_GRP_BACKGROUND, 0, MAP_AREA_MEMORY, 100, 0, AODB[0] );
```

### 6.2.2.1.2.14 Zyklische Datenübertragung

Die zyklische Datenübertragung wird ähnlich wie die Hintergrundabfrage von der Unterstation eingeleitet und ist unabhängig von anderen Befehlen aus der Zentralstation. Durch die zyklische Datenübertragung werden die Prozessdaten der Zentralstation kontinuierlich aufgefrischt. Bei den Prozessdaten handelt es sich meistens um Messwerte, die in regulären Zeitabständen erfasst werden. Die zyklische Datenübertragung wird oft benutzt um nicht-zeitkritische, oder sich nicht so schnell ändernde Prozessdaten zu überwachen (z.B. ein Temperatursensor). Die zyklischen/periodischen Daten werden zur Zentralstation mit der Übertragungsursache <1> *Periodic/Cyclic* übertragen. Die zulässigen ASDU-Typkennungen sind in der Kompatibilitätsliste der Station aufgeführt (Tabelle Typkennung <-> Übertragungsursache). Die zyklische Datenübertragung kann über die Systemparameter und die Konfigurationsparameter der Applikationsobjekte konfiguriert werden.

**Konfiguration der Systemparameter:**

```
bPerCyclic      : BOOL := TRUE;
tPerCyclicBase  : TIME := T#60s;
```

Der erste Parameter aktiviert die zyklische Übertragung. Der zweite Parameter ist die Basiszeit der zyklischen/periodischen Datenübertragung (hier z.B. 60 Sekunden).

**Konfiguration der Applikationsobjekte:**

- IEC870\_GRP\_PERCYC-Gruppenparameter muss gesetzt werden;
- Der Multiplikator-Parameter (*multiplier*) der F\_iecInitAOEntry-Funktion muss auf einen Wert <> Null gesetzt werden. Beispiel: Bei einem Multiplikator = 2 und Basiszeit von 60 Sekunden werden die die Prozessdaten des Applikationsobjekts alle 120 Sekunden zur Zentralstation gesendet;

Beispielkonfiguration für einen Messwert der zyklisch alle 120 Sekunden zur Zentralstation übertragen werden soll (measured value, normalized value without time tag, M\_ME\_NA\_1).

```
F_iecInitAOEntry( M_ME_NA_1, 222, IEC870_GRP_INROGEN OR IEC870_GRP_PERCYC, 2, MAP_AREA_MEMORY, 6, 0, AODB[2] );
```

**6.2.2.1.2.15 Befehlsübertragung**

Befehle können von der Zentralstation in Steuerungsrichtung (zur Unterstation) gesendet werden. Ein Einzelbefehl mit der Typkennung 45 (C\_SC\_NA\_1) wird zur Steuerung eines Applikationsobjekts benutzt, das in Überwachungsrichtung als Einzelmeldung (M\_SP\_NA\_1, M\_SP\_TA\_1 oder M\_SP\_TB\_1) übertragen wird. Ein Doppelbefehl (C\_DC\_NA\_1) wird zur Steuerung eines Applikationsobjekts benutzt, das in Überwachungsrichtung als Doppelmeldung (M\_DP\_NA\_1, M\_DP\_TA\_1 oder M\_DP\_TB\_1) übertragen wird, usw.

**Konfiguration der Systemparameter:**

- Es müssen keine speziellen Systemparameter gesetzt werden;

**Konfiguration der Applikationsobjekte:**

- Die Applikationsobjekte müssen als Befehle (Datentypen in Steuerungsrichtung) konfiguriert werden;
- Die Adressen der Informationsobjekte (IOA's) müssen den Adressen in der Leitstation entsprechen;

Beispiele:

Single command mit der IOA = 10. Der empfangene Wert wird in den Merkerbereichspuffer, Byteoffset = 100, Bitoffset = 0 kopiert.

```
F_iecInitAOEntry( C_SC_NA_1, 10, 0, 0, MAP_AREA_MEMORY, 100, 0, AODB[24] );
```

Double command mit der IOA = 20. Der empfangene Wert wird in den Merkerbereichspuffer, Byteoffset = 200, Bitoffset = 0..1 kopiert.

```
F_iecInitAOEntry( C_DC_NA_1, 20, 0, 0, MAP_AREA_MEMORY, 200, 0, AODB[27] );
```

Set point, scaled value mit der IOA = 60. Der empfangene Wert wird in den Merkerbereichspuffer, Byteoffset = 600..601, Bitoffset = 0 kopiert.

```
F_iecInitAOEntry( C_SE_NB_1, 60, 0, 0, MAP_AREA_MEMORY, 600, 0, AODB[39] );
```

**6.2.2.1.2.16 Abfrage-/Lese-Prozedur**

Die Zentralstation sendet Abfragebefehle an die Unterstation.

In dem Abfragebefehl wird die Adresse des abzufragenden Applikationsobjekts übertragen. Die Daten dieses Applikationsobjekts sollen an die Zentralstation gesendet werden. Die Unterstation sendet die Daten mit der Übertragungsursache <5> *Abfrage oder abgefragt*. Die zulässigen ASDU-Typkennungen sind in der Kompatibilitätsliste der Station aufgelistet (Tabelle Typkennung <-> Übertragungsursache).

**Konfiguration der Systemparameter**

- Es müssen keine speziellen Systemparameter gesetzt werden;

**Konfiguration der Applikationsobjekte**

- Es müssen keine speziellen Parameter gesetzt werden;

### 6.2.2.1.2.17 Doppelübertragung

Alle Applikationsobjekte (Informationsobjekte), die mit der Übertragungsursache <3> *Spontan* übertragen werden, dürfen zweimal übertragen werden, mit und ohne Zeitmarke. Diese Betriebsart wird "Doppelübertragung" genannt.

Die Doppelübertragung wird von der Unterstation derzeit nicht unterstützt. Eine eigene Implementierung ist über das [Low-Level Interface \[► 114\]](#) möglich.

### 6.2.2.1.2.18 Quality-Flags

Die Quality-Flags (Quality-Descriptor) liefern der Zentralstation zusätzliche Informationen zur Qualität eines Applikationsobjekts. Die Quality-Flags können aus der SPS-Applikation mit Hilfe der [F\\_iecSetAOQuality \[► 437\]](#) Funktion unabhängig voneinander gesetzt/zurückgesetzt werden. Mit der [F\\_iecGetAOQuality](#) Funktion kann der Status der Quality-Flags abgefragt werden. Jede Änderung der Quality-Flags führt zu einer spontanen Übertragung der Daten zur Zentralstation.

Folgende Quality-Flags werden intern von der Unterstation zur Laufzeit ausgewertet:

- IECQ\_BL\_ON (Blocked). Wurden die Prozessdaten des Applikationsobjekts für die Übertragung blockiert, dann wird das Mapping der SPS- und IEC-Prozessdaten für dieses Applikationsobjekt nicht ausgeführt;

Folgende Quality-Flags werden intern von der Unterstation zur Laufzeit gesetzt/zurückgesetzt:

- IECQ\_IV\_ON (Invalid). Die Unterstation setzt das Invalid-Flag wenn das Mapping der SPS- und IEC-Prozessdaten nicht durchgeführt werden konnte (z.B. wegen fehlerhafter Konfiguration des Applikationsobjekts). Dieses Verhalten kann durch einen gesetzten Group-Parameter: IEC870\_GRP\_IV\_OFF deaktiviert werden.

Alle anderen Quality-Flags werden unverändert zur Zentralstation gesendet.

### 6.2.2.1.2.19 Test der Kommunikation

Durch das Setzen der *bChangeIO*-Variable auf TRUE wird eine einfache Simulation der Datenpunkte in Überwachungsrichtung aktiviert und mit FALSE deaktiviert. Bei einer aktiven Verbindung werden die Werte Zyklisch alle 3 Sekunden zur Leitstation übertragen.

```
PROGRAM MAIN
VAR
    ...

    bChangeIO : BOOL; (* TRUE => simulate/modify plc process data *)
    timer : TON;
    i : INT;

    ...
END_VAR

...

(*modify plc process data *)
timer( IN := bChangeIO, PT := T#3s );
IF timer.Q THEN
    timer( IN := FALSE );

    msgSingle_0 := NOT msgSingle_0;
    msgSingle_1 := NOT msgSingle_1;
    msgSingle_2 := NOT msgSingle_2;

    FOR i:= 0 TO 3 DO
        IF F_iecGetDPI(msgDouble_0, i) = eIEC870_DPI_ON THEN (* the value of double point already ON? *)
            msgDouble_0 := F_iecSetDPI( msgDouble_0, i, eIEC870_DPI_OFF ); (* change ON => OFF *)
        )
        ELSE
```

```

        msgDouble_0 := F_iecSetDPI(      msgDouble_0, i, eIEC870_DPI_ON ); (* change OFF => ON *)
    END_IF
END_FOR

F_iecIncVTI( msgStep_0 );
F_iecDecVTI( msgStep_1 );

msgBitStr_0 := ROL( msgBitStr_0, 1 );
msgBitStr_1 := ROR( msgBitStr_1, 1 );

msgNormalized_0 := msgNormalized_0 + 1;
msgNormalized_1 := msgNormalized_1 + 2;

msgScaled_0 := msgScaled_0 + 3;
msgScaled_1 := msgScaled_1 - 3;

msgFloating_0 := msgFloating_0 + 0.1;
msgFloating_1 := msgFloating_1 + 1.5;

msgTotal_0 := msgTotal_0 + 1;
msgTotal_1 := msgTotal_1 + 2;
END_IF
...

```

### 6.2.2.1.2.20 Übertragungs- und Kommunikationsfehler

Die Stationsfehlermeldungen werden in einem FIFO abgelegt. Es können bis zu 10 Fehlermeldungen zwischengespeichert werden. Bei fatalen Kommunikationsfehlern (z. B. Fehler der Verbindungsschicht, die Checksumme des Frames passt nicht) wird die Verbindung unterbrochen und muss neu aufgebaut werden. Fehler in der Applikationsschicht (z. B. der ASDU-Sendepuffer ist wegen zu vieler Frames übergelaufen) werden nur geloggt und führen nicht zum Verbindungsabbruch. Es immer noch möglich auch bei diesen Fehlern die Verbindung aus der Applikation zu unterbrechen. Neben dem Fehler-Code wird auch die Fehlerquelle in der Fehlermeldung abgelegt. Dieses erleichtert die Lokalisierung des Fehlers.

#### Beispiel

Die anfallenden Fehlermeldungen einer IEC 60870-5-101 Unterstation können durch folgenden Aufruf ausgelesen werden:

```

PROGRAM MAIN
VAR
...
    server : FB_IEC870_5_101Slave;
...
END_VAR

....

REPEAT
    server.system.device.errors.RemoveError();
    IF server.system.device.errors.bOk THEN
        ADSLOGSTR( ADSLOG_MSGTYPE_ERROR OR ADSLOG_MSGTYPE_LOG,
            'IEC60870-5-101 slave error: 0x%s',
            DWORD_TO_HEXSTR( server.system.device.errors.getError.nErrId, 8, FALSE) );
    END_IF
UNTIL NOT server.system.device.errors.bOk
END_REPEAT
...

```

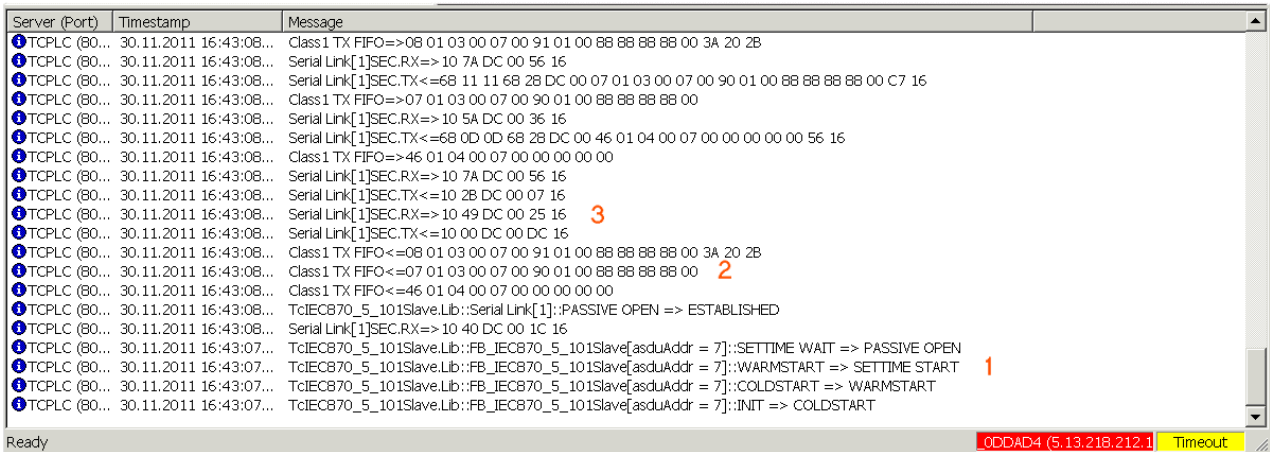
### 6.2.2.1.2.21 Fehlersuche und Diagnose

Debug-Meldungen, die ins Application-Log geschrieben werden erleichtern eine Fehlersuche im System. Derzeit können drei Stufen der Debugmeldungen in einer IEC-Applikation aktiviert werden. Diese Meldungen können durch den dbgMode-Systemparameter der Unterstation aktiviert werden ([ST\\_IEC870\\_5\\_101SystemParams](#) ▶ 487]).

1. Stationsstatus-Meldungen (dbgMode: IEC870\_DEBUGMODE\_DEVSTATE);

2. Hexadezimale Ausgabe der ASDU's (ohne Link-Layer-Control-Header, dbgMode: IEC870\_DEBUGMODE\_ASDU). Pro Zeile werden 32-ASDU-Datenbytes als hexadezimale Zahlen ausgegeben. Längere ASDU's werden auf mehrere Zeilen verteilt;
3. Hexadezimale Ausgabe der APDU's (Serial-Port-Telegramme, dbgMode: IEC870\_DEBUGMODE\_LINKLAYER). Pro Zeile werden 32-APDU-Datenbytes als hexadezimale Zahlen ausgegeben. Ähnlich wie bei 2. werden längere APDU's auf mehrere Zeilen verteilt;

Optional können auch Link-Layer-Fehler ausgegeben werden (dbgMode: IEC870\_DEBUGMODE\_LINKERROR). Um die aktivierten Debugmeldungen zu sehen starten Sie TwinCAT System Manager und aktivieren Sie die Loggeransicht. Die nachfolgende Grafik zeigt eine Debugausgabe. Die drei unterschiedlichen Typen der Meldungen wurden mit entsprechenden Zahlen markiert.



Weitere Diagnosetools:

- Portmon for Windows (v3.02, Windows Sysinternals)
- Diverse Protokoll-Test-Suit-Produkte

## 6.2.2.1.2.22 Konfiguration der seriellen Schnittstellen

- [Konfiguration der standard PC COMx-Schnittstellen \[► 615\]](#)
- [Konfiguration der seriellen Busklemmen KL6xxx \[► 616\]](#)
- [Konfiguration der seriellen Busklemmen EL6xxx \[► 616\]](#)

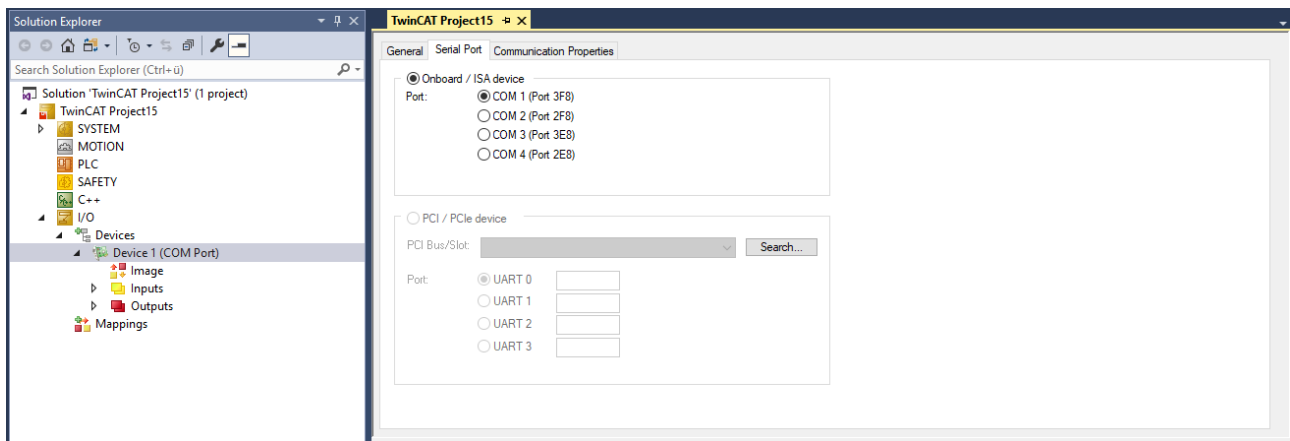
### 1. Konfiguration der standard PC COMx-Schnittstellen

Die serielle Schnittstelle kann nur in PC/CX-Systemen eingefügt werden.

Mit der rechten Maustaste "E/A Geräte" anklicken. "Gerät anfügen" auswählen. Unter "Verschiedenes" die "Serielle Schnittstelle" auswählen. Anschließend nachfolgende Einstellungen vornehmen.

#### 1.1. Karteireiter Serial Port

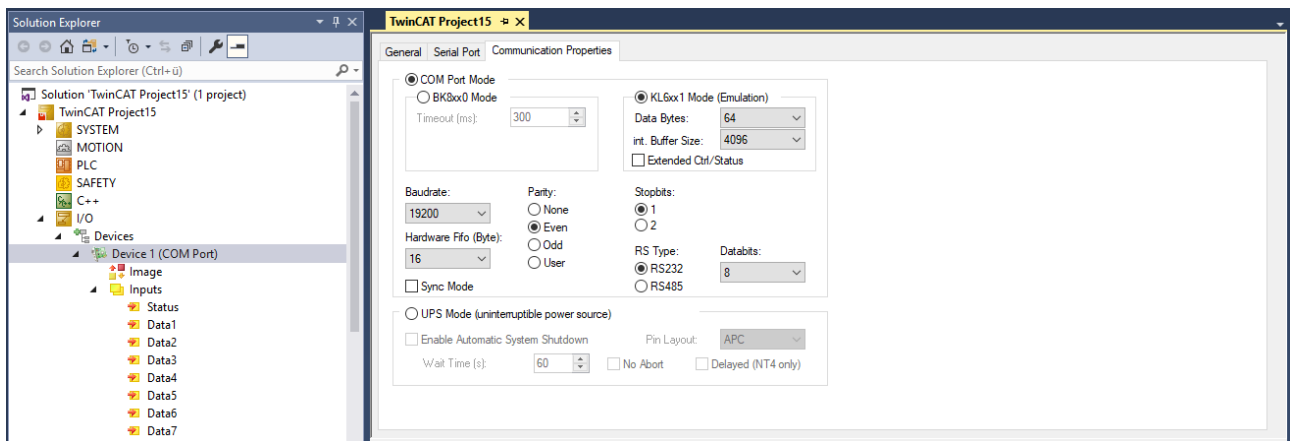
Serielle Schnittstelle COM1



## 1.2. Karteireiter Eigenschaften

Die Option KL6xxx1 Mode (Emulation) anwählen und dann die Kommunikationsparameter einstellen:

Baudrate, hier 19200 Baud, 8 Datenbits, Parity=even, 1 Stopbit



## 2. Konfiguration der seriellen Busklemmen KL6xxx

Die Schnittstelle wird in der TwinCAT SPS durch die Instanz des FB\_IEC870\_SerialLineCtrl-Funktionsbausteins konfiguriert. Die Kommunikationsparameter wie Baudrate, Parity usw. sind an diesem Baustein einzustellen.

### 2.1 RS485 Betriebsmodus

Bei der RS485 Betriebsart (z.B. KL6041) werden die Daten in halbduplex Übertragung ausgetauscht. Im RS485 Betriebsmodus werden die Send- und Empfangsleitungen miteinander verbunden. Dadurch empfängt die Klemme nicht nur die Daten anderer Teilnehmer, sondern auch ihre eigenen Sendedaten. Diese Daten können die Kommunikation stören. Für den Halbduplex-Modus muss der Handshake-Eingang des FB\_IEC870\_SerialLineCtrl-Funktionsbausteins auf den Wert: RS485\_HALFDUPLEX gesetzt werden.

## 3. Konfiguration der seriellen Busklemmen EL6xxx

Die serielle Schnittstelle wird im TwinCAT System Manager Karteireiter CoE-Online (CoE=CanOpen over EtherCAT) eingestellt. Der Karteireiter steht nur bei Online-Zugriff auf die Klemme zur Verfügung, d.h., ist der System Manager nicht mit der Hardware verbunden, können keine Einstellungen verändert werden. Die Einstellungen werden in der Klemme Spannungsausfallsicher abgespeichert. Nach einem Scannen der Klemmen müssen diese Daten nicht neu eingegeben werden. Bei einem Tausch der Klemme werden die Daten aus der neuen Klemme aktiv und müssen demzufolge wieder angepasst werden. Um das zu verhindern, müssen im Karteireiter *Startup* die geänderten Daten eingetragen werden. Beim Systemstart werden die Daten aus dem Reiter *Startup* in die Klemme übertragen. Ein Tausch der Klemme ist somit problemlos möglich. Nach einem Scannen der Klemmen müssen die Daten im Reiter *Startup* neu eingegeben werden.



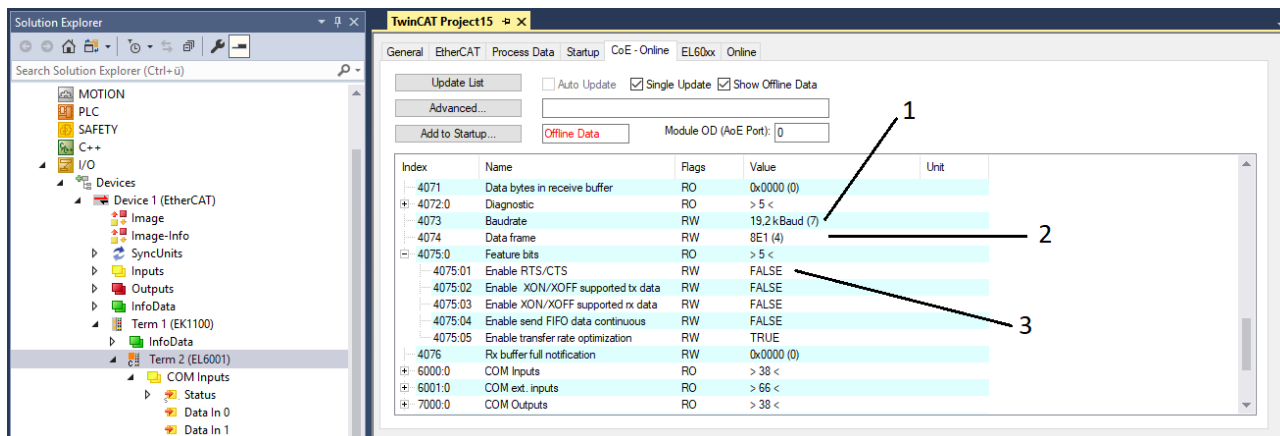
### 3.1. Karteireiter CoE

(1) Baudrate, hier 19200 Baud

(2) Date frame, 8E1 entspricht 8 Datenbits, Parity=even, 1 Stoppbit

(3) Feature bits -> Enable RTS/CTS = FALSE bei EL6001

Mit einem Doppelklick auf die entsprechende Zeile wird ein Menü geöffnet, in dem die Einstellungen geändert werden können.



#### 3.1.1 RS485 Betriebsmodus

Bei der RS485 Betriebsart werden die Daten in halbduplex Übertragung ausgetauscht. Im RS485 Betriebsmodus werden die Sendeleitungen miteinander verbunden. Dadurch empfängt die Klemme nicht nur die Daten anderer Teilnehmer, sondern auch ihre eigenen Sendedaten. Diese Daten können die Kommunikation stören. Mit der Option "Enable half duplex" im "COM Settings"-Objekt kann dies unterdrückt werden (Index 8000:06).

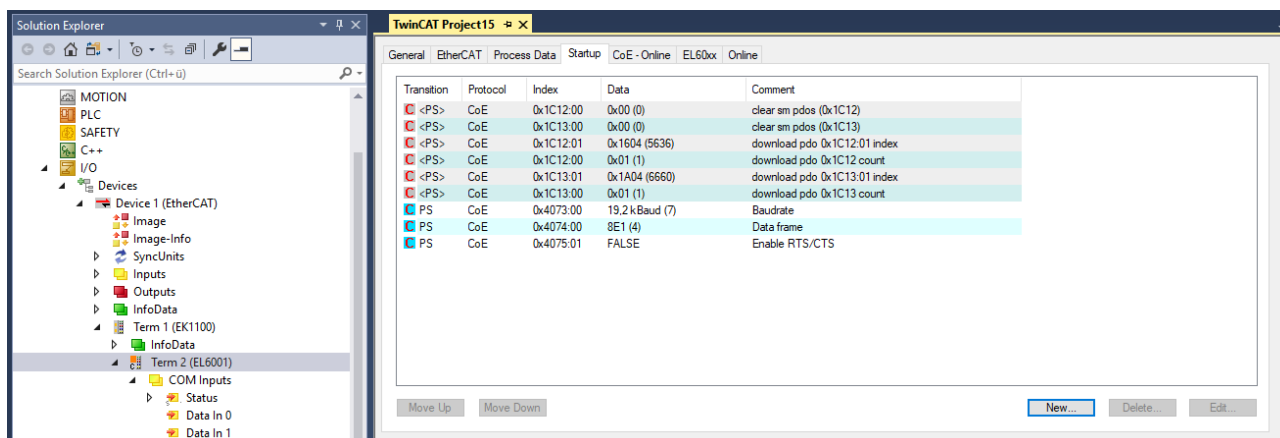
### 3.2. Karteireiter Startup

Alle vom Standard abweichenden Einstellungen müssen in die Startup Liste eingetragen werden. Beim Systemstart werden die Daten aus dem Reiter *Startup* in die Klemme übertragen. Ein Tausch der Klemme ist somit problemlos möglich. Nach einem Scannen der Klemmen müssen die Daten im Reiter *Startup* neu eingegeben werden.

Baudrate, hier 19200 Baud

Date frame, 8E1 entspricht 8 Datenbits, Parity=even, 1 Stoppbit

Feature bits -> Enable RTS/CTS = FALSE bei EL6001



## 6.2.2.2 Low-Level Interface

### 6.2.2.2.1 Quick start

Einfaches Projekt mit kompletten Quellen finden Sie hier: [IEC 60870-5-101 Unterstation-Low-Level-Beispiel \[▶ 618\]](#).

Eine SPS-Applikation, die über das "Low-Level"-Interface mit einer Zentralstation kommunizieren soll, benötigt folgende Ressourcen:

- Eine Instanz des Kommunikationsbausteins: [FB\\_IEC870\\_5\\_101TProtocol \[▶ 407\]](#);
- Eine Instanz des TX/RX-Datenpuffers: [ST\\_IEC870\\_5\\_101TBuffer \[▶ 481\]](#);
- Eine Instanz des Funktionsbausteins zur Manipulation des TX/RX-Datenpuffers: [FB\\_IEC870\\_5\\_101TBufferCtrl \[▶ 403\]](#);

#### 6.2.2.2.1.1 Download: Unterstation (slave, unbalanced)

Einfache Unterstation-Applikation die das TwinCAT IEC 60870-5-101 Low-Level Interface verwendet. Diese Beispiellapplikation kann mit einer Zentralstation über die RS232-Schnittstelle kommunizieren.

Download TwinCAT XAE Project (\*.zip): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/984618763.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/984618763.zip)

Im Beispielprojekt sind folgende Stationsparameter für die Unterstation eingestellt:

- Standard PC interface: **COM1**.
- Baudrate: **19200 Baud**.
- Link address: **220**.
- Link address size: **1 octet**.
- Cause of transfer size: **1 octet**.
- Originator address: **0 (not used)**.
- Common ASDU address: **7**.
- Common ASDU address size: **1 octet**.
- Information object address size: **2 octets**.
- Link mode: **Unbalanced**.

Datenpunktconfiguration (sie sollte gleich sein mit der Datenpunktconfiguration der Zentralstation):

- Single point information: M\_SP\_NA\_1 (IOA := 100), M\_SP\_TA\_1 (IOA := 101), M\_SP\_TB\_1 (IOA := 102).
- Double point information: M\_DP\_NA\_1 (IOA := 200).
- Regulating step value: M\_ST\_NA\_1 (IOA := 300).
- Bitstring value: M\_BO\_NA\_1 (IOA := 400).
- Measured value, normalized value: M\_ME\_NA\_1 (IOA := 500).
- Measured value, scaled value: M\_ME\_NB\_1 (IOA := 600).
- Measured value, short floating point value: M\_ME\_NC\_1 (IOA := 700).
- Binary counter value: M\_IT\_NA\_1 (IOA := 800), M\_IT\_TA\_1 (IOA := 801), M\_IT\_TB\_1 (IOA := 802).
- Single command: C\_SC\_NA\_1 (IOA := 10).
- Double command: C\_DC\_NA\_1 (IOA := 20).
- Regulating step command: C\_RC\_NA\_1 (IOA := 30).
- Bitstring command: C\_BO\_NA\_1 (IOA := 40).
- Setpoint command, normalized value: C\_SE\_NA\_1 (IOA := 50).
- Setpoint command, scaled value: C\_SE\_NB\_1 (IOA := 60).

- Setpoint command, short floating point value: C\_SE\_NC\_1 (IOA := 70).

Beispielimplementierung folgender Dienste:

- Testbefehl (C\_TS\_NA\_1).
- Uhrzeitsynchronisationsbefehl (C\_CS\_NA\_1).
- Generalabfragebefehl (C\_IC\_NA\_1) der Gruppe: Allgemein (eIEC870\_QOI\_INROGEN).
- Zählerabfragebefehl (C\_IC\_NA\_1) der Gruppe: Allgemein (eIEC870\_RQT\_REQCOGEN), Mode A. Die Zählwerte werden lokal alle 60 Sekunden umgespeichert.
- Befehlsausführung einiger Befehle (siehe obere Datenpunktconfiguration).
- Ende der Initialisierung (M\_EI\_NA\_1) wird einmalig zur Zentralstation gesendet (nach dem die Verbindung hergestellt wurde).
- Zyklische Datenübertragung alle 60 Sekunden.
- Simulation und Übertragung der spontanen Daten alle 60 Sekunden wenn die Variable **bSimulate** auf TRUE gesetzt wurde.

Folgende Tabelle beschreibt kurz die Hauptkomponenten des Projekts:

Komponente	Typ	Beschreibung
P_MAIN_LowSpeed	PRG	Dieses Programmteil implementiert das Fernwirkprotokoll (langsame Task).
P_SerialComm_HighSpeed	PRG	Dieses Programmteil implementiert die TwinCAT Hintergrundkommunikation über die seriellen Schnittstellen des PCs (schnelle Task).
fbServer	FB_IEC870_5_101S (FB)	Implementiert die Unterstation. Eine Instanz dieses Bausteins kann nur mit einer Zentralstation kommunizieren. Für jede weitere Zentralstation mit der kommuniziert werden soll wird eine weitere Instanz dieses Funktionsbausteins benötigt.
fbServices	FB_IEC870_5_101SS ervices (FB)	Beinhaltet eine einfache Implementierung einiger Dienste wie z.B. Generalabfrage, Uhrzeitsynchronisation, Testbefehle usw.
stProcData	ST_IEC870_5_101SP rocData (STRUCT)	Hier befinden sich die empfangenen und die zu sendenden Prozessdaten die ständig aktualisiert werden.
sysPara	ST_IEC870_5_101SS ysPara (STRUCT)	Konfiguration der Systemparameter der Unterstation und der implementierten Dienste.
fbLog	FB_IEC870_DebugLo gFifo (FB)	Ein einfacher Logpuffer für Debugzwecke/ Fehlersuche.

### 6.2.2.2.1.2 Download: Unterstation (slave, balanced mode)

Einfache Unterstation-Applikation die das TwinCAT IEC 60870-5-101 Low-Level Interface verwendet. Diese Beispielapplikation kann mit einer Zentralstation über eine RS232-Schnittstelle kommunizieren.

Download TwinCAT XAE Project (\*.zip): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/984621707.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/984621707.zip)

Im Beispielprojekt sind folgende Stationsparameter für die Unterstation eingestellt:

- Standard PC interface: **COM1**.
- Baudrate: **19200 Baud**.
- Link address: **220**.
- Link address size: **1 octet**.
- Cause of transfer size: **1 octet**.
- Originator address: **0 (not used)**.
- Common ASDU address: **7**.

- Common ASDU address size: **1 octet**.
- Information object address size: **2 octets**.
- Link mode: **Balanced**.

Datenpunktconfiguration (sie sollte gleich sein mit der Datenpunktconfiguration der Zentralstation):

- Single point information: M\_SP\_NA\_1 (IOA := 100), M\_SP\_TA\_1 (IOA := 101), M\_SP\_TB\_1 (IOA := 102)
- Double point information: M\_DP\_NA\_1 (IOA := 200)
- Regulating step value: M\_ST\_NA\_1 (IOA := 300)
- Bitstring value: M\_BO\_NA\_1 (IOA := 400)
- Measured value, normalized value: M\_ME\_NA\_1 (IOA := 500)
- Measured value, scaled value: M\_ME\_NB\_1 (IOA := 600)
- Measured value, short floating point value: M\_ME\_NC\_1 (IOA := 700)
- Binary counter value: M\_IT\_NA\_1 (IOA := 800), M\_IT\_TA\_1 (IOA := 801), M\_IT\_TB\_1 (IOA := 802)
- Single command: C\_SC\_NA\_1 (IOA := 10)
- Double command: C\_DC\_NA\_1 (IOA := 20)
- Regulating step command: C\_RC\_NA\_1 (IOA := 30)
- Bitstring command: C\_BO\_NA\_1 (IOA := 40)
- Setpoint command, normalized value: C\_SE\_NA\_1 (IOA := 50)
- Setpoint command, scaled value: C\_SE\_NB\_1 (IOA := 60)
- Setpoint command, short floating point value: C\_SE\_NC\_1 (IOA := 70)

Beispielimplementierung folgender Dienste:

- Testbefehl (C\_TS\_NA\_1).
- Uhrzeitsynchronisationsbefehl (C\_CS\_NA\_1).
- Generalabfragebefehl (C\_IC\_NA\_1) der Gruppe: Allgemein (eIEC870\_QOI\_INROGEN).
- Zählerabfragebefehl (C\_IC\_NA\_1) der Gruppe: Allgemein (eIEC870\_RQT\_REQCOGEN), Mode A. Die Zählwerte werden lokal alle 60 Sekunden umgespeichert.
- Befehlsausführung einiger Befehle (siehe obere Datenpunktconfiguration).
- Ende der Initialisierung (M\_EI\_NA\_1) wird einmalig zur Zentralstation gesendet (nach dem die Verbindung hergestellt wurde).
- Zyklische Datenübertragung alle 60 Sekunden.
- Simulation und Übertragung der spontanen Daten alle 60 Sekunden wenn die Variable **bSimulate** auf TRUE gesetzt wurde.

Folgende Tabelle beschreibt kurz die Hauptkomponenten des Projekts:

Komponente	Typ	Beschreibung
P_MAIN_LowSpeed	PRG	Dieses Programmteil implementiert das Fernwirkprotokoll (langsame Task).
P_SerialComm_HighSpeed	PRG	Dieses Programmteil implementiert die TwinCAT Hintergrundkommunikation über die seriellen Schnittstellen des PCs (schnelle Task).
fbServer	FB_IEC870_5_101S (FB)	Implementiert die Unterstation. Eine Instanz dieses Bausteins kann nur mit einer Zentralstation kommunizieren. Für jede weitere Zentralstation mit der kommuniziert werden soll wird eine weitere Instanz dieses Funktionsbausteins benötigt.
fbServices	FB_IEC870_5_101SS ervices (FB)	Beinhaltet eine einfache Implementierung einiger Dienste wie z.B. Generalabfrage, Uhrzeitsynchronisation, Testbefehle usw.
stProcData	ST_IEC870_5_101SP rocData (STRUCT)	Hier befinden sich die empfangenen und die zu sendenden Prozessdaten die ständig aktualisiert werden.
sysPara	ST_IEC870_5_101SS ysPara (STRUCT)	Konfiguration der Systemparameter der Unterstation und der implementierten Dienste.
fbLog	FB_IEC870_DebugLo gFifo (FB)	Ein einfacher Logpuffer für Debugzwecke/ Fehlersuche.

### 6.2.2.2.1.3 Konfiguration der seriellen Schnittstellen

- [Konfiguration der standard PC COMx-Schnittstellen \[► 621\]](#)
- [Konfiguration der seriellen Busklemmen KL6xxx \[► 622\]](#)
- [Konfiguration der seriellen Busklemmen EL6xxx \[► 622\]](#)

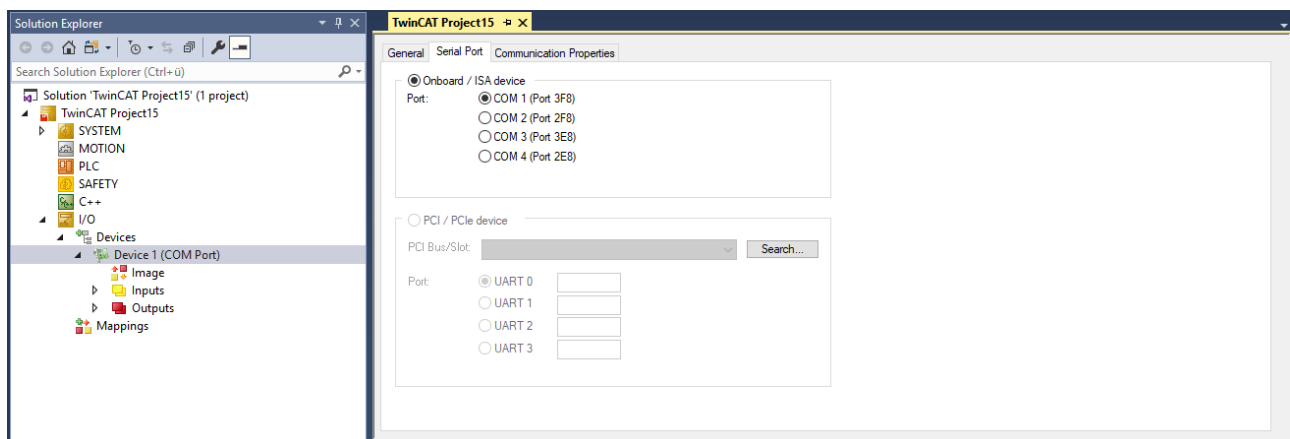
#### 1. Konfiguration der standard PC COMx-Schnittstellen

Die serielle Schnittstelle kann nur in PC/CX-Systemen eingefügt werden.

Mit der rechten Maustaste "E/A Geräte" anklicken. "Gerät anfügen" auswählen. Unter "Verschiedenes" die "Serielle Schnittstelle" auswählen. Anschließend nachfolgende Einstellungen vornehmen.

##### 1.1. Karteireiter Serial Port

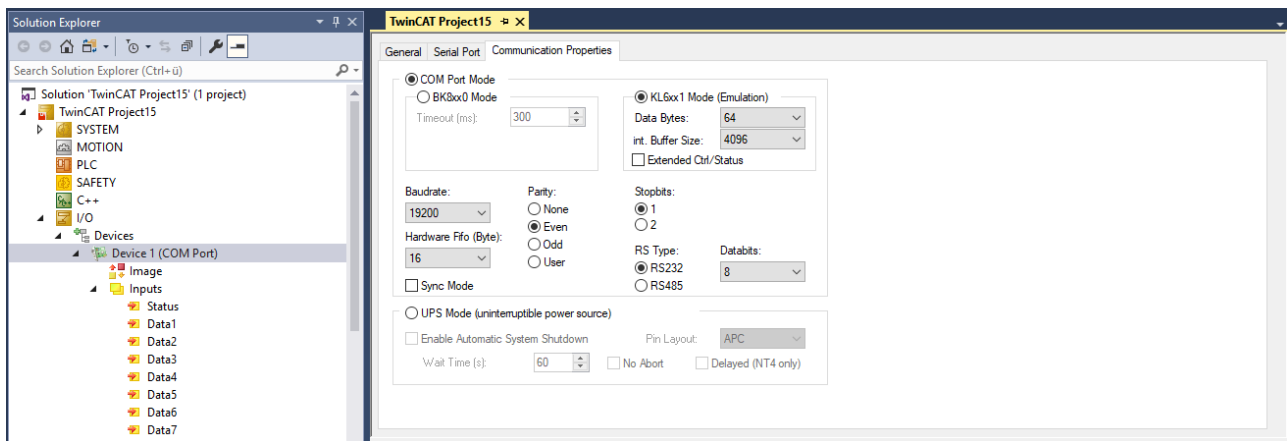
Serielle Schnittstelle COM1



##### 1.2. Karteireiter Eigenschaften

Die Option KL6xxx1 Mode (Emulation) anwählen und dann die Kommunikationsparameter einstellen:

Baudrate, hier 19200 Baud, 8 Datenbits, Parity=even, 1 Stoppbit



## 2. Konfiguration der seriellen Busklemmen KL6xxx

Die Schnittstelle wird in der TwinCAT SPS durch die Instanz des FB\_IEC870\_SerialLineCtrl-Funktionsbausteins konfiguriert. Die Kommunikationsparameter wie Baudrate, Parity usw. sind an diesem Baustein einzustellen.

### 2.1 RS485 Betriebsmodus

Bei der RS485 Betriebsart (z.B. KL6041) werden die Daten in halbduplex Übertragung ausgetauscht. Im RS485 Betriebsmodus werden die Send- und Empfangsleitungen miteinander verbunden. Dadurch empfängt die Klemme nicht nur die Daten anderer Teilnehmer, sondern auch ihre eigenen Sendedaten. Diese Daten können die Kommunikation stören. Für den Halbduplex-Modus muss der Handshake-Eingang des FB\_IEC870\_SerialLineCtrl-Funktionsbausteins auf den Wert: RS485\_HALFDUPLEX gesetzt werden.

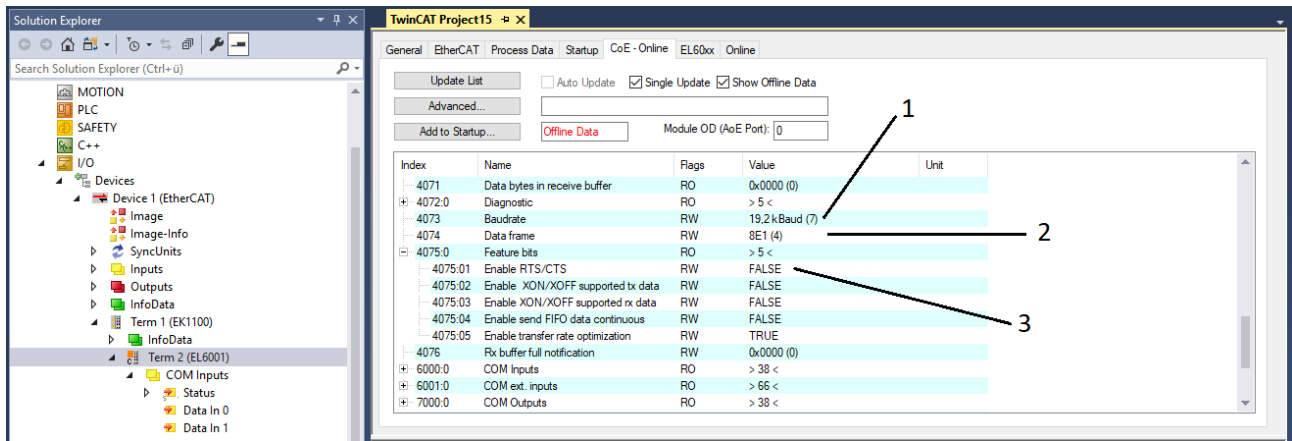
## 3. Konfiguration der seriellen Busklemmen EL6xxx

Die serielle Schnittstelle wird im TwinCAT System Manager Karteireiter CoE-Online (CoE=CanOpen over EtherCAT) eingestellt. Der Karteireiter steht nur bei Online-Zugriff auf die Klemme zur Verfügung, d.h., ist der System Manager nicht mit der Hardware verbunden, können keine Einstellungen verändert werden. Die Einstellungen werden in der Klemme Spannungsausfallsicher abgespeichert. Nach einem Scannen der Klemmen müssen diese Daten nicht neu eingegeben werden. Bei einem Tausch der Klemme werden die Daten aus der neuen Klemme aktiv und müssen demzufolge wieder angepasst werden. Um das zu verhindern, müssen im Karteireiter *Startup* die geänderten Daten eingetragen werden. Beim Systemstart werden die Daten aus dem Reiter *Startup* in die Klemme übertragen. Ein Tausch der Klemme ist somit problemlos möglich. Nach einem Scannen der Klemmen müssen die Daten im Reiter *Startup* neu eingegeben werden.

### 3.1. Karteireiter CoE

- (1) Baudrate, hier 19200 Baud
- (2) Date frame, 8E1 entspricht 8 Datenbits, Parity=even, 1 Stoppbit
- (3) Feature bits -> Enable RTS/CTS = *FALSE* bei EL6001

Mit einem Doppelklick auf die entsprechende Zeile wird ein Menü geöffnet, in dem die Einstellungen geändert werden können.



### 3.1.1 RS485 Betriebsmodus

Bei der RS485 Betriebsart werden die Daten in halbduplex Übertragung ausgetauscht. Im RS485 Betriebsmodus werden die Sendeleitungen miteinander verbunden. Dadurch empfängt die Klemme nicht nur die Daten anderer Teilnehmer, sondern auch ihre eigenen Sendedaten. Diese Daten können die Kommunikation stören. Mit der Option "Enable half duplex" im "COM Settings"-Objekt kann dies unterdrückt werden (Index 8000:06).

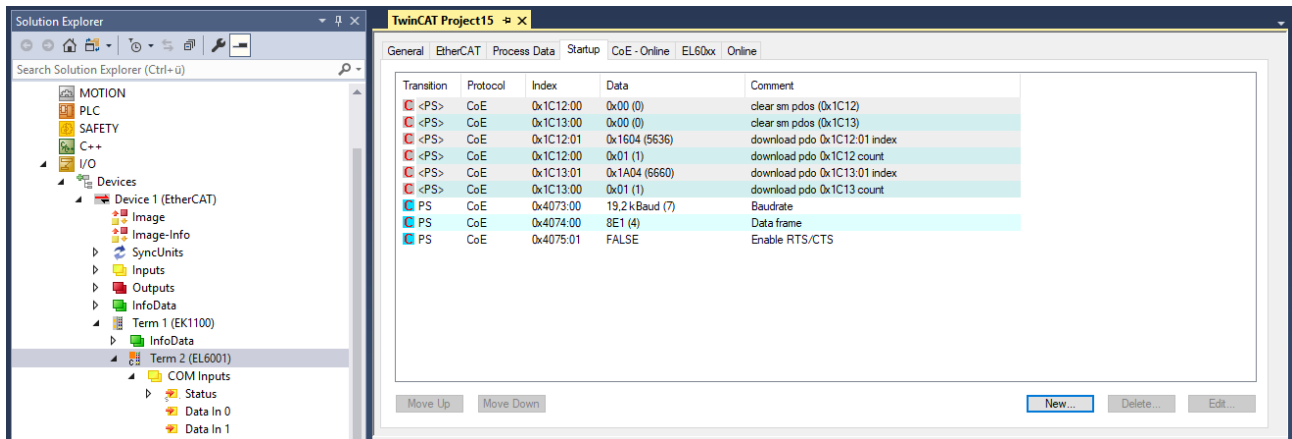
### 3.2. Karteireiter Startup

Alle vom Standard abweichenden Einstellungen müssen in die Startup Liste eingetragen werden. Beim Systemstart werden die Daten aus dem Reiter *Startup* in die Klemme übertragen. Ein Tausch der Klemme ist somit problemlos möglich. Nach einem Scannen der Klemmen müssen die Daten im Reiter *Startup* neu eingegeben werden.

Baudrate, hier 19200 Baud

Date frame, 8E1 entspricht 8 Datenbits, Parity=even, 1 Stopbit

Feature bits -> Enable RTS/CTS = *FALSE* bei EL6001



## 6.3 IEC 60870-5-102

### 6.3.1 Low-Level Interface

#### 6.3.1.1 Quick start

Einfaches Projekt mit kompletten Quellen finden Sie hier: [IEC 60870-5-102 Zentralstation-Low-Level-Beispiel \[▶ 624\]](#).

Eine SPS-Applikation, die über das Transport Interface mit einer Unterstation kommunizieren soll, benötigt folgende Ressourcen:

- Eine Instanz des Kommunikationsbausteins: [FB\\_IEC870\\_5\\_102TProtocol](#) [► 408];
- Eine Instanz des TX/RX-Datenpuffers: [ST\\_IEC870\\_5\\_102TBuffer](#) [► 544];
- Eine Instanz des Funktionsbausteins zur Manipulation des TX/RX-Datenpuffers: [FB\\_IEC870\\_5\\_102TBufferCtrl](#) [► 405];

### 6.3.1.1.1 Download: Zentralstation (master)

Einfache Zentralstation-Applikation die das TwinCAT IEC 60870-5-102 Low-Level Interface verwendet. Diese Beispielapplikation kann mit bis zu drei Unterstationen über eine RS232-Schnittstelle kommunizieren. Im Programm ist aber nur die Kommunikation zur ersten Unterstation aktiviert.

Download TwinCAT XAE Project (\*.zip): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/984624651.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/984624651.zip)

Im Beispielprojekt sind folgende Stationsparameter für die Zentralstation eingestellt:

- Standard PC interface: **COM1**.
- Baudrate: **19200 Baud**.
- Link address: **220 (erste Unterstation), 221 (zweite Unterstation), 222 (dritte Unterstation)**.
- Link address size: **2 octets**.
- Common ASDU address: **7 (erste Unterstation), 8 (zweite Unterstation), 9 (dritte Unterstation)**.
- Common ASDU address size: **2 octets**.

Beispielimplementierung folgender Dienste:

- Abruf einer Liste von Einzelmeldungen mit Zeitmarke eines ausgewählten Zeitraums (C\_SP\_NB\_2) alle 30 Sekunden.
- Abruf der Hersteller- und Produktspezifikation (C\_RD\_NA\_2) alle 10 Sekunden.

Folgende Tabelle beschreibt kurz die Hauptkomponenten des Projekts:

Komponente	Typ	Beschreibung
P_MAIN_LowSpeed	PRG	Dieses Programmteil implementiert das Fernwirkprotokoll (langsame Task).
P_SerialComm_HighSpeed	PRG	Dieses Programmteil implementiert die TwinCAT Hintergrundkommunikation über die seriellen Schnittstellen des PCs (schnelle Task).
aClients	ARRAY OF FB_IEC870_5_102M (FB)	Implementiert die Zentralstation. Eine Instanz dieses Bausteins kann nur mit einer Unterstation kommunizieren. Für jede weitere Unterstation mit der kommuniziert werden soll wird eine weitere Instanz dieses Funktionsbausteins benötigt.
fbServices	FB_IEC870_5_102MS ervices (FB)	Beinhaltet eine einfache Implementierung einiger Dienste wie z.B. Lesen der Hersteller/Produkt-Spezifikation, Abruf der Liste von Einzelmeldungen usw.
aProcData	ARRAY OF ST_IEC870_5_102MProcData (STRUCT)	Hier befinden sich die empfangenen und die zu sendenden Prozessdaten die ständig aktualisiert werden.
sysPara	ST_IEC870_5_102MSysPara (STRUCT)	Konfiguration der Systemparameter der Zentralstation und der implementierten Dienste.
fbLog	FB_IEC870_DebugLogFifo (FB)	Ein einfacher Logpuffer für Debugzwecke/ Fehlersuche.



### 6.3.1.1.2 Konfiguration der seriellen Schnittstellen

- [Konfiguration der standard PC COMx-Schnittstellen \[► 625\]](#)
- [Konfiguration der seriellen Busklemmen KL6xxx \[► 625\]](#)
- [Konfiguration der seriellen Busklemmen EL6xxx \[► 626\]](#)

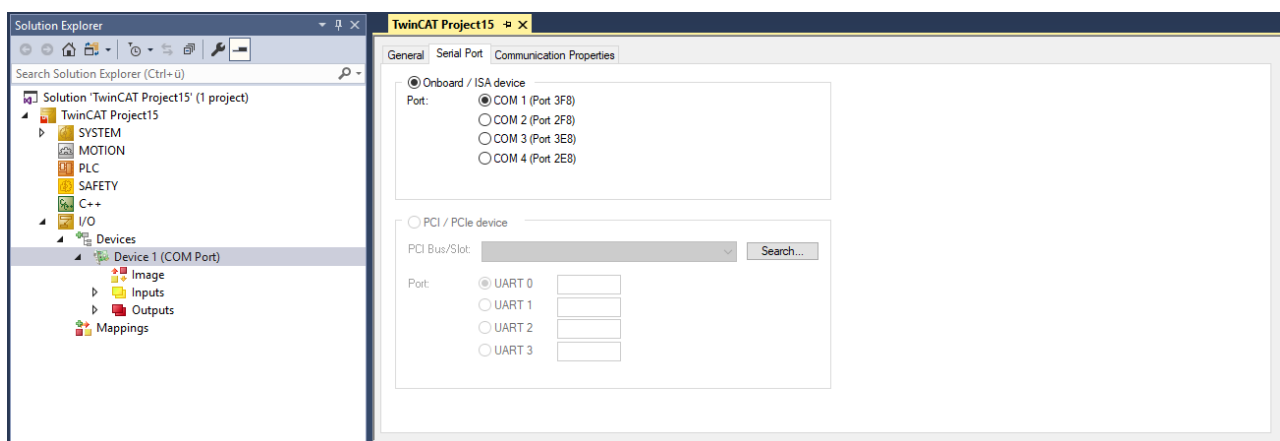
#### 1. Konfiguration der standard PC COMx-Schnittstellen

Die serielle Schnittstelle kann nur in PC/CX-Systemen eingefügt werden.

Mit der rechten Maustaste "E/A Geräte" anklicken. "Gerät anfügen" auswählen. Unter "Verschiedenes" die "Serielle Schnittstelle" auswählen. Anschließend nachfolgende Einstellungen vornehmen.

##### 1.1. Karteireiter Serial Port

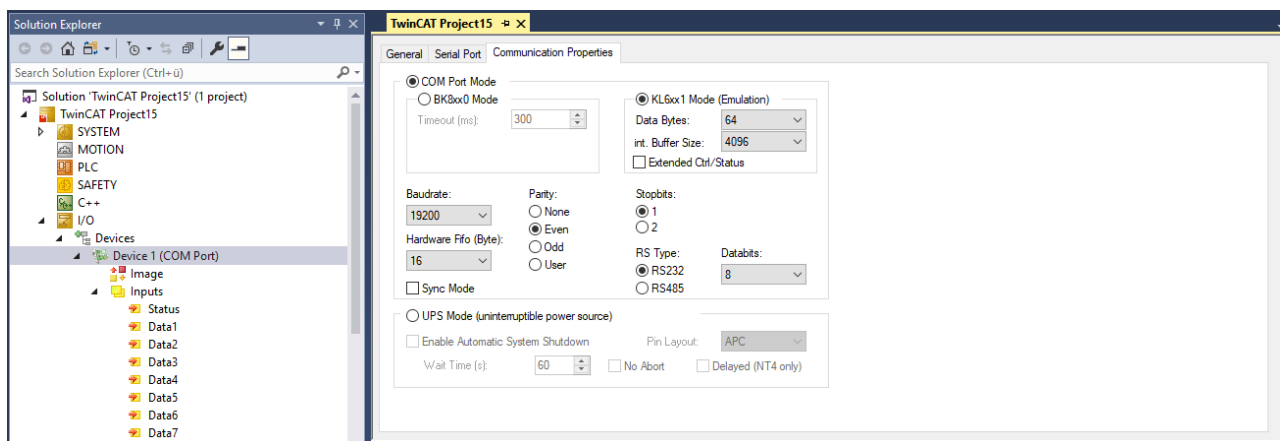
Serielle Schnittstelle COM1



##### 1.2. Karteireiter Eigenschaften

Die Option KL6xxx1 Mode (Emulation) anwählen und dann die Kommunikationsparameter einstellen:

Baudrate, hier 19200 Baud, 8 Datenbits, Parity=even, 1 Stopbit



#### 2. Konfiguration der seriellen Busklemmen KL6xxx

Die Schnittstelle wird in der TwinCAT SPS durch die Instanz des FB\_IEC870\_SerialLineCtrl-Funktionsbausteins konfiguriert. Die Kommunikationsparameter wie Baudrate, Parity usw. sind an diesem Baustein einzustellen.

## 2.1 RS485 Betriebsmodus

Bei der RS485 Betriebsart (z.B. KL6041) werden die Daten in halbduplex Übertragung ausgetauscht. Im RS485 Betriebsmodus werden die Sendeleitungen miteinander verbunden. Dadurch empfängt die Klemme nicht nur die Daten anderer Teilnehmer, sondern auch ihre eigenen Sendedaten. Diese Daten können die Kommunikation stören. Für den Halbduplex-Modus muss der Handshake-Eingang des FB\_IEC870\_SerialLineCtrl-Funktionsbausteins auf den Wert: RS485\_HALFDUPLEX gesetzt werden.

## 3. Konfiguration der seriellen Busklemmen EL6xxx

Die serielle Schnittstelle wird im TwinCAT System Manager Karteireiter CoE-Online (CoE=CanOpen over EtherCAT) eingestellt. Der Karteireiter steht nur bei Online-Zugriff auf die Klemme zur Verfügung, d.h., ist der System Manager nicht mit der Hardware verbunden, können keine Einstellungen verändert werden. Die Einstellungen werden in der Klemme Spannungsausfallsicher abgespeichert. Nach einem Scannen der Klemmen müssen diese Daten nicht neu eingegeben werden. Bei einem Tausch der Klemme werden die Daten aus der neuen Klemme aktiv und müssen demzufolge wieder angepasst werden. Um das zu verhindern, müssen im Karteireiter *Startup* die geänderten Daten eingetragen werden. Beim Systemstart werden die Daten aus dem Reiter *Startup* in die Klemme übertragen. Ein Tausch der Klemme ist somit problemlos möglich. Nach einem Scannen der Klemmen müssen die Daten im Reiter *Startup* neu eingegeben werden.

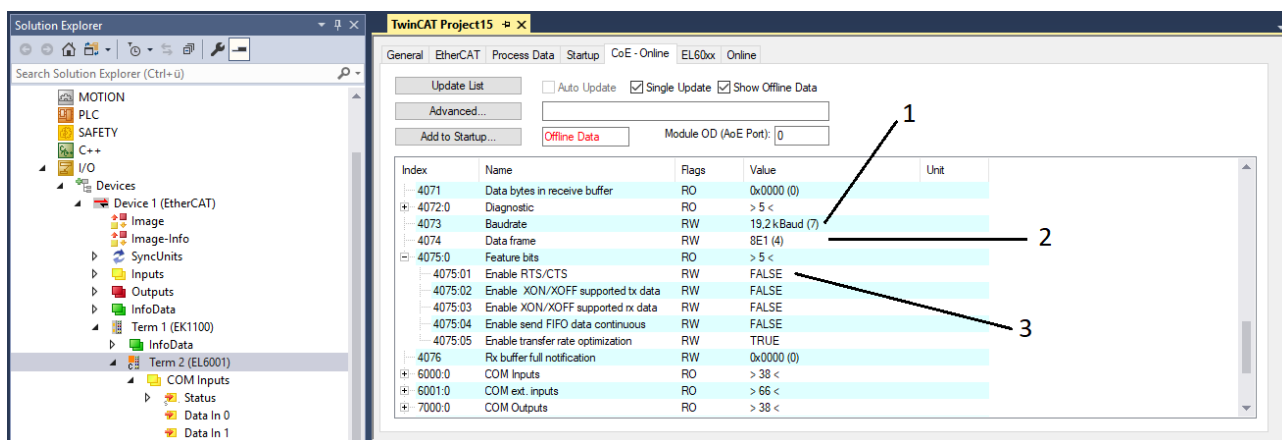
### 3.1. Karteireiter CoE

(1) Baudrate, hier 19200 Baud

(2) Data frame, 8E1 entspricht 8 Datenbits, Parity=even, 1 Stopbit

(3) Feature bits -> Enable RTS/CTS = FALSE bei EL6001

Mit einem Doppelklick auf die entsprechende Zeile wird ein Menü geöffnet, in dem die Einstellungen geändert werden können.



#### 3.1.1 RS485 Betriebsmodus

Bei der RS485 Betriebsart werden die Daten in halbduplex Übertragung ausgetauscht. Im RS485 Betriebsmodus werden die Sendeleitungen miteinander verbunden. Dadurch empfängt die Klemme nicht nur die Daten anderer Teilnehmer, sondern auch ihre eigenen Sendedaten. Diese Daten können die Kommunikation stören. Mit der Option "Enable half duplex" im "COM Settings"-Objekt kann dies unterdrückt werden (Index 8000:06).

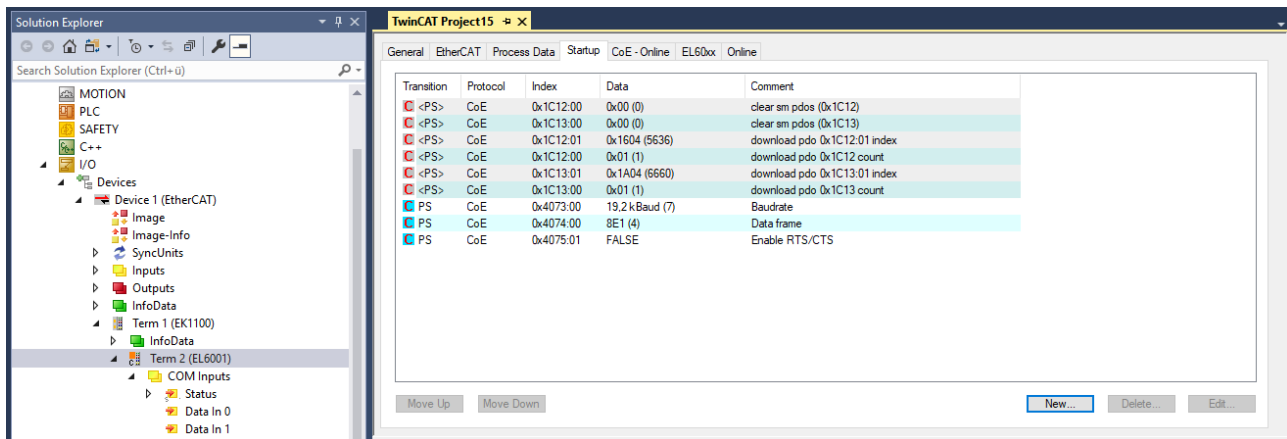
### 3.2. Karteireiter Startup

Alle vom Standard abweichenden Einstellungen müssen in die Startup Liste eingetragen werden. Beim Systemstart werden die Daten aus dem Reiter *Startup* in die Klemme übertragen. Ein Tausch der Klemme ist somit problemlos möglich. Nach einem Scannen der Klemmen müssen die Daten im Reiter *Startup* neu eingegeben werden.

Baudrate, hier 19200 Baud

Date frame, 8E1 entspricht 8 Datenbits, Parity=even, 1 Stopbit

Feature bits -> Enable RTS/CTS = *FALSE* bei EL6001



## 6.4 IEC 60870-5-103

### 6.4.1 Low-Level Interface

#### 6.4.1.1 Quick start

Einfaches Projekt mit kompletten Quellen finden Sie hier: [IEC 60870-5-103 Zentralstation-Low-Level-Beispiel \[► 627\]](#).

Eine SPS-Applikation, die über das Transport Interface mit einer Unterstation kommunizieren soll, benötigt folgende Ressourcen:

- Eine Instanz des Kommunikationsbausteins: [FB IEC870\\_5\\_103TProtocol \[► 410\]](#);
- Eine Instanz des TX/RX-Datenpuffers: [ST IEC870\\_5\\_103TBuffer \[► 547\]](#);
- Eine Instanz des Funktionsbausteins zur Manipulation des TX/RX-Datenpuffers: [FB IEC870\\_5\\_103TBufferCtrl \[► 406\]](#);

##### 6.4.1.1.1 Download: Zentralstation (master)

Einfache Zentralstation-Applikation die das TwinCAT IEC 60870-5-103 Low-Level Interface verwendet. Diese Beispielapplikation kann mit bis zu zehn Unterstationen über eine RS232-Schnittstelle kommunizieren. Im Programm ist aber nur die Kommunikation zur ersten Unterstation aktiviert.

Download TwinCAT XAE Project (\*.zip): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/984627595.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/984627595.zip)

Im Beispielprojekt sind folgende Stationsparameter für die Zentralstation eingestellt:

- Standard PC interface: **COM1**.
- Baudrate: **19200 Baud**.
- Link address: **220 (erste Unterstation), 221 (zweite Unterstation), 222 (dritte Unterstation), ..., 229 (zehnte Unterstation)**.
- Link address size: **1 octet (fix)**.
- Common ASDU address: **220 (erste Unterstation), 221 (zweite Unterstation), 222 (dritte Unterstation), ..., 229 (zehnte Unterstation)**.
- Common ASDU address size: **1 octet (fix)**.

Beispielimplementierung folgender Dienste:

- Uhrzeitsynchronisation (C\_SYN\_TA\_3) alle 60 Sekunden.
- Generalabfrage (C\_IGI\_NA\_3) alle 60 Sekunden.

- General command (C\_GRC\_NA\_3) wenn die Membervariable **ST\_IEC870\_5\_103MProcData.eCmdCtrl** auf den Wert eIEC870\_CMD\_EXECUTE gesetzt wurde.
- Empfang von Meldungen mit Zeitstempel (M\_TTM\_TA\_3).

Folgende Tabelle beschreibt kurz die Hauptkomponenten des Projekts:

Komponente	Typ	Beschreibung
P_MAIN_LowSpeed	PRG	Dieses Programmteil implementiert das Fernwirkprotokoll (langsame Task).
P_SerialComm_HighSpeed	PRG	Dieses Programmteil implementiert die TwinCAT Hintergrundkommunikation über die seriellen Schnittstellen des PCs (schnelle Task).
aClients	ARRAY OF FB_IEC870_5_103M (FB)	Implementiert die Zentralstation. Eine Instanz dieses Bausteins kann nur mit einer Unterstation kommunizieren. Für jede weitere Unterstation mit der kommuniziert werden soll wird eine weitere Instanz dieses Funktionsbausteins benötigt.
fbServices	FB_IEC870_5_103M Services (FB)	Beinhaltet eine einfache Implementierung einiger Dienste wie z.B. Generalabfrage, Uhrzeitsynchronisation usw.
aProcPara	ARRAY OF ST_IEC870_5_103M ProcData (STRUCT)	Hier befinden sich die empfangenen und die zu sendenden Prozessdaten die ständig aktualisiert werden.
sysPara	ST_IEC870_5_103M SysPara (STRUCT)	Konfiguration der Systemparameter der Zentralstation und der implementierten Dienste.
fbLog	FB_IEC870_DebugLogFifo (FB)	Ein einfacher Logpuffer für Debugzwecke/ Fehlersuche.

### 6.4.1.1.2 Konfiguration der seriellen Schnittstellen

- [Konfiguration der standard PC COMx-Schnittstellen \[► 628\]](#)
- [Konfiguration der seriellen Busklemmen KL6xxx \[► 629\]](#)
- [Konfiguration der seriellen Busklemmen EL6xxx \[► 629\]](#)

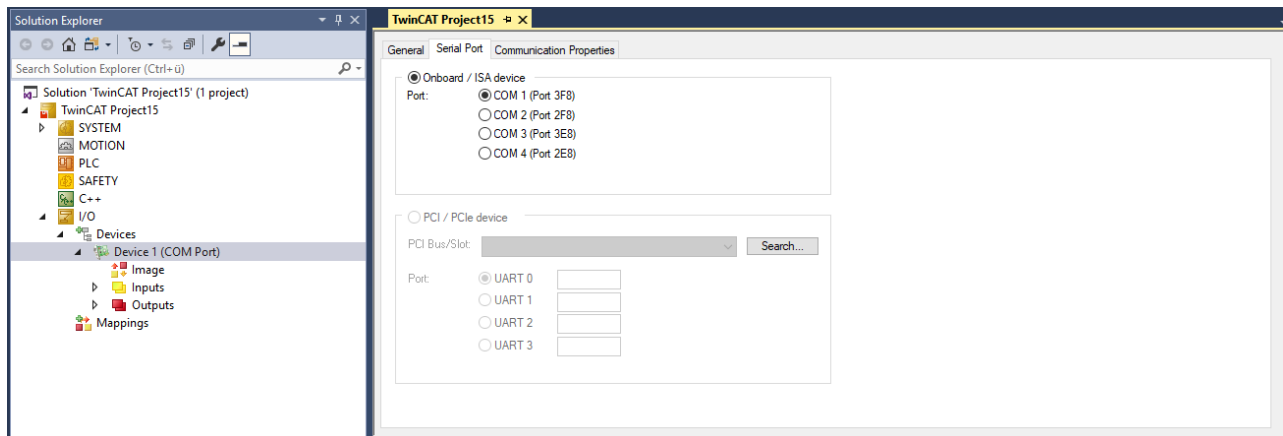
#### 1. Konfiguration der standard PC COMx-Schnittstellen

Die serielle Schnittstelle kann nur in PC/CX-Systemen eingefügt werden.

Mit der rechten Maustaste "E/A Geräte" anklicken. "Gerät anfügen" auswählen. Unter "Verschiedenes" die "Serielle Schnittstelle" auswählen. Anschließend nachfolgende Einstellungen vornehmen.

#### 1.1. Karteireiter Serial Port

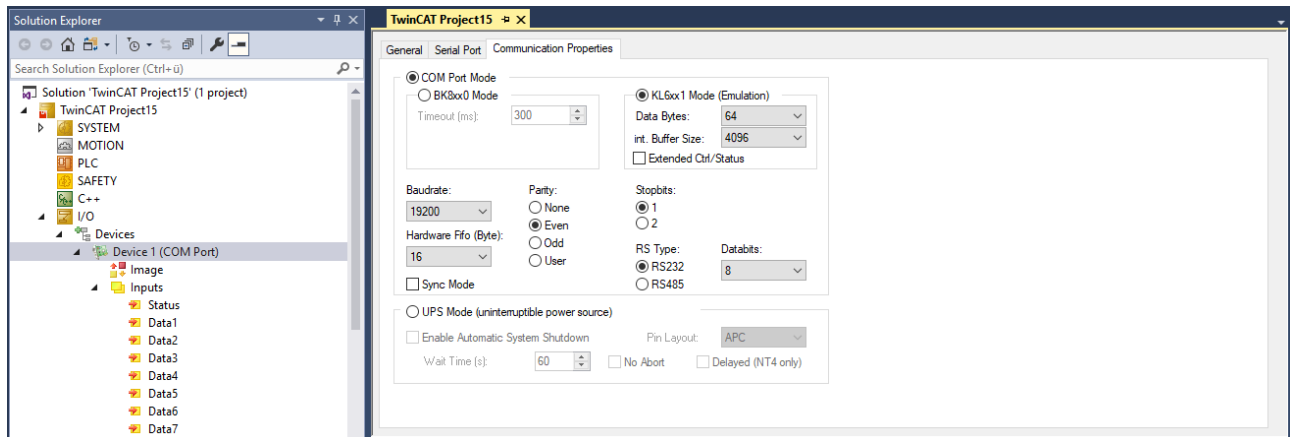
Serielle Schnittstelle COM1



## 1.2. Karteireiter Eigenschaften

Die Option KL6xxx1 Mode (Emulation) anwählen und dann die Kommunikationsparameter einstellen:

Baudrate, hier 19200 Baud, 8 Datenbits, Parity=even, 1 Stoppbit



## 2. Konfiguration der seriellen Busklemmen KL6xxx

Die Schnittstelle wird in der TwinCAT SPS durch die Instanz des FB\_IEC870\_SerialLineCtrl-Funktionsbausteins konfiguriert. Die Kommunikationsparameter wie Baudrate, Parity usw. sind an diesem Baustein einzustellen.

### 2.1 RS485 Betriebsmodus

Bei der RS485 Betriebsart (z.B. KL6041) werden die Daten in halbduplex Übertragung ausgetauscht. Im RS485 Betriebsmodus werden die Sende- und Empfangsleitungen miteinander verbunden. Dadurch empfängt die Klemme nicht nur die Daten anderer Teilnehmer, sondern auch ihre eigenen Sendedaten. Diese Daten können die Kommunikation stören. Für den Halbduplex-Modus muss der Handshake-Eingang des FB\_IEC870\_SerialLineCtrl-Funktionsbausteins auf den Wert: RS485\_HALFDUPLEX gesetzt werden.

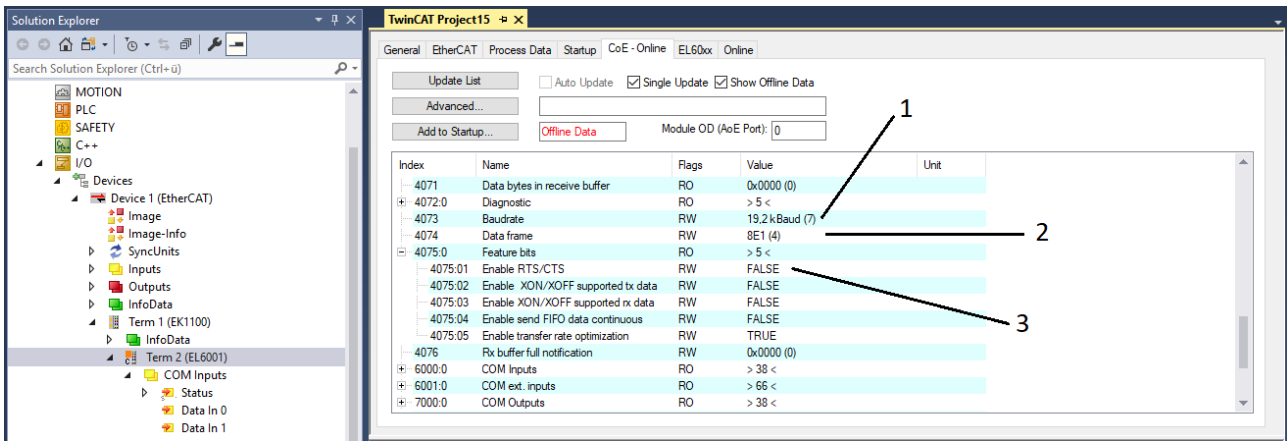
## 3. Konfiguration der seriellen Busklemmen EL6xxx

Die serielle Schnittstelle wird im TwinCAT System Manager Karteireiter CoE-Online (CoE=CanOpen over EtherCAT) eingestellt. Der Karteireiter steht nur bei Online-Zugriff auf die Klemme zur Verfügung, d.h., ist der System Manager nicht mit der Hardware verbunden, können keine Einstellungen verändert werden. Die Einstellungen werden in der Klemme Spannungsausfallsicher abgespeichert. Nach einem Scannen der Klemmen müssen diese Daten nicht neu eingegeben werden. Bei einem Tausch der Klemme werden die Daten aus der neuen Klemme aktiv und müssen demzufolge wieder angepasst werden. Um das zu verhindern, müssen im Karteireiter *Startup* die geänderten Daten eingetragen werden. Beim Systemstart werden die Daten aus dem Reiter *Startup* in die Klemme übertragen. Ein Tausch der Klemme ist somit problemlos möglich. Nach einem Scannen der Klemmen müssen die Daten im Reiter *Startup* neu eingegeben werden.

### 3.1. Karteireiter CoE

- (1) Baudrate, hier 19200 Baud
- (2) Date frame, 8E1 entspricht 8 Datenbits, Parity=even, 1 Stoppbit
- (3) Feature bits -> Enable RTS/CTS = FALSE bei EL6001

Mit einem Doppelklick auf die entsprechende Zeile wird ein Menü geöffnet, in dem die Einstellungen geändert werden können.



### 3.1.1 RS485 Betriebsmodus

Bei der RS485 Betriebsart werden die Daten in halbduplex Übertragung ausgetauscht. Im RS485 Betriebsmodus werden die Sendeleitungen miteinander verbunden. Dadurch empfängt die Klemme nicht nur die Daten anderer Teilnehmer, sondern auch ihre eigenen Sendedaten. Diese Daten können die Kommunikation stören. Mit der Option "Enable half duplex" im "COM Settings"-Objekt kann dies unterdrückt werden (Index 8000:06).

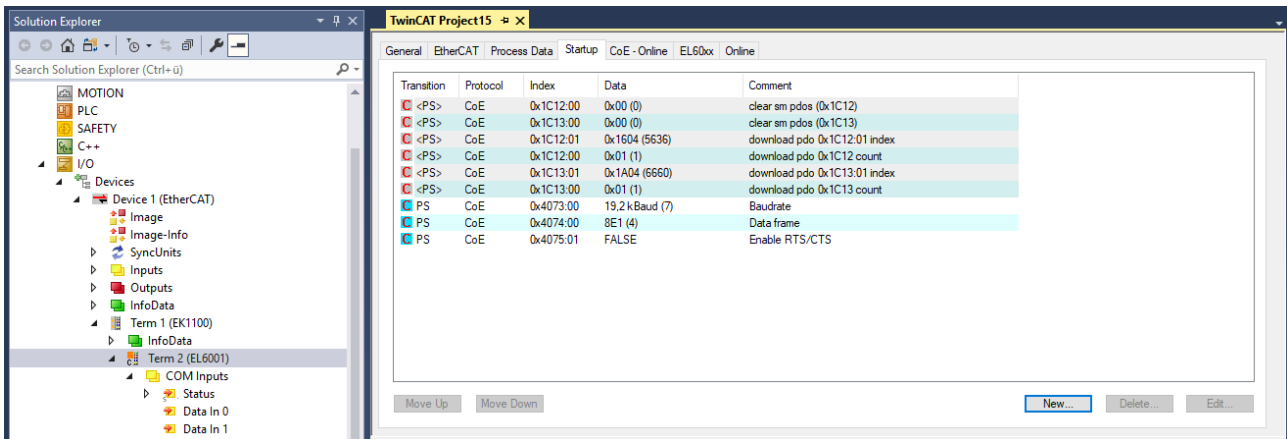
### 3.2. Karteireiter Startup

Alle vom Standard abweichenden Einstellungen müssen in die Startup Liste eingetragen werden. Beim Systemstart werden die Daten aus dem Reiter *Startup* in die Klemme übertragen. Ein Tausch der Klemme ist somit problemlos möglich. Nach einem Scannen der Klemmen müssen die Daten im Reiter *Startup* neu eingegeben werden.

Baudrate, hier 19200 Baud

Date frame, 8E1 entspricht 8 Datenbits, Parity=even, 1 Stopbit

Feature bits -> Enable RTS/CTS = *FALSE* bei EL6001



## 6.5 IEC 60870-5-104

### 6.5.1 Zentralstation

#### 6.5.1.1 High-Level Interface

##### 6.5.1.1.1 Quick start

Einfaches Projekt mit kompletten Sourcen finden Sie hier: [IEC 60870-5-104 Zentralstation-High-Level-Beispiel](#) [▶ 632].

Kompatibilitätsliste finden Sie hier: [Interoperability check list](#) [▶ 68]

Übersicht der Fehlercodes finden Sie hier: [Fehlercodes](#) [▶ 708]

#### Kurzanleitung

##### Applikationsobjekt-Datenbank

Die Applikationsobjekt-Datenbank der Zentralstation muss mit der Funktion: [F\\_iecCreateTableHnd](#) [▶ 452] als Hash-Tabelle konfiguriert werden. Die einzelnen Arrayelemente werden dabei als Hash-Tabelle untereinander verlinkt. Dies ermöglicht u. a. einen schnelleren Zugriff auf die einzelnen Datenpunkte, bringt aber auch einige Nachteile mit sich, die beachtet werden müssen:

- Die Größe der Applikationsdatenbank (Arraygröße) darf nicht zur Laufzeit (z.B. durch Online-Change) verändert werden. Die Zentralstation stoppt sofort die Ausführung und meldet einen Fehler. Der Grund: Die Verlinkung der Hash-Tabelle passt nicht mehr. Bei Programmänderungen laden Sie am besten das komplette Projekt ins Laufzeitsystem.
- Auf die einzelnen Array-Elemente darf nicht per Index sondern nur mit Hilfe der speziellen Funktionen zugegriffen werden (z.B.: [F\\_iecAddTableEntry](#) [▶ 453] usw.).
- Bei einem indizierten Zugriff auf die Tabellenelemente dürfen die internen Konfigurationsparameter nicht beschrieben oder verändert werden. Bei einer Änderung des Typs, der ASDU-Adresse oder der Objektadresse kann der Datenpunkt nicht mehr gefunden werden. Ein Datenpunkt, der umkonfiguriert werden soll wird aus der Tabelle durch den Funktionsaufruf [F\\_iecRemoveTableEntry](#) [▶ 460] zuerst entfernt. Danach kann der neue Datenpunkt hinzugefügt werden.

Bei einer Implementierung als lineare Tabelle müsste die Zentralstation bei jeder empfangenen ASDU (Dateneinheit) das komplette Array nach dem passenden Element durchsuchen. Dies würde bei vielen Datenpunkten sehr lange Ausführungszeiten generieren.

##### Protokollparameter

Über die Protokollparameter wird das Verhalten der TCP/IP-Transportschicht konfiguriert. Die meisten Protokollparameter sind bereits mit Defaultwerten vorbelegt und müssen nicht explizit gesetzt werden. Die SPS-Applikation muss aber einmalig die IP-Adresse (*sRemoteHost*) und die Port-Adresse (*nRemotePort*) der Unterstation konfigurieren.

##### Systemparameter

Die Systemparameter sind ebenfalls mit Defaultwerten vorbelegt. Während der Inbetriebnahme ist es nützlich die Debug-Ausgabe zu aktivieren (*dbgMode*) um mögliche Fehler lokalisieren zu können.

##### Parameter für die zyklische Datenerfassung

Folgende Parameter sind bereits mit Defaultwerten vorkonfiguriert:

- Initialisierungssequenz (besteht aus einem Testbefehl, Uhrzeitsynchronisation, Stationsabfrage und Zählerabfrage);
- Zyklische Befehle:

- Testbefehl alle 60s;
- Uhrzeitsynchronisation alle 60s;
- Stationsabfrage der Gruppe: Allgemein alle 60s;
- Zählerabfrage der Gruppe: Allgemein alle 60s;

### 6.5.1.1.2 Zentralstation Minimal (Master)

Einfache Zentralstation-Applikation die das TwinCAT IEC 60870-5-104 High-Level Interface verwendet.

Download TwinCAT XAE Project (\*.zip): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/984630539.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/984630539.zip)

Im Beispielprojekt sind folgende Stationsparameter für die Zentralstation eingestellt:

- Server host address (IP-Adresse der Unterstation): 127.0.0.1 (Sie müssen mindestens diesen Parameter an Ihre Zielplattform anpassen!).
- Server port address (Portadresse der Unterstation): **2404**.
- k: **12**.
- w: **8**.
- t0: **30s**.
- t1: **15s**.
- t2: **10s**.
- t3: **20s**.
- Cause of transfer size: **2 octets (inclusive originator address)**.
- Common ASDU address size: **2 octets**.
- Information object address size: **3 octets**.
- Originator address: **1**.
- Common ASDU address: **7**.
- Max. APDU length: **253**.

Datenpunktconfiguration (sie sollte gleich sein mit der Datenpunktconfiguration der Unterstationen):

- Single point information: M\_SP\_NA\_1 (IOA := 100).
- Single command: C\_SC\_NA\_1 (IOA := 10).

Initialisierungssequenz, die nach dem Verbindungsaufbau einmalig/automatisch ausgeführt wird:

- Testbefehl (C\_TS\_TA\_1).
- Uhrzeitsynchronisationsbefehl (C\_CS\_NA\_1).
- Generalabfragebefehl (C\_IC\_NA\_1) der Gruppe: Allgemein (eIEC870\_QOI\_INROGEN).
- Zählerabfragebefehl (C\_CI\_NA\_1) der Gruppe: Allgemein (eIEC870\_RQT\_REQCOGEN).  
Zählerabfrage Mode C.

Beispielimplementierung folgender Dienste:

- Testbefehl (C\_TS\_NA\_1) alle 60 Sekunden.
- Uhrzeitsynchronisationsbefehl (C\_CS\_NA\_1) alle 60 Sekunden.
- Generalabfragebefehl (C\_IC\_NA\_1) der Gruppe: Allgemein (eIEC870\_QOI\_INROGEN) alle 60 Sekunden.
- Zählerabfragebefehl (C\_CI\_NA\_1) der Gruppe: Allgemein (eIEC870\_RQT\_REQCOGEN) alle 60 Sekunden. Zählerabfrage Mode C.
- Bei Wertänderung an der **bSCS** -Variablen wird der Einzelbefehl (C\_SC\_NA\_1) gesendet (direkte Kommandoausführung)
- Beim Empfang spontaner Daten vom Typ (M\_SP\_NA\_1, IOA := 100) wird der ON/OFF-Wert in die SPS-Variable **bSPI** gemappt..



Folgende Tabelle beschreibt kurz die Hauptkomponenten des Projekts:

Komponente	Typ	Beschreibung
MAIN	PRG	Dieses Programmteil implementiert das Fernwirkprotokoll (langsame Task).
fbClient	FB_IEC870_5_104Master (FB)	Implementiert die Zentralstation. Eine Instanz dieses Bausteins kann nur mit einer Unterstation kommunizieren. Für jede weitere Unterstation mit der kommuniziert werden soll wird eine weitere Instanz dieses Funktionsbausteins benötigt.
bSPI	BOOL	Der empfangene ON/OFF-Wert wird in diese SPS-Variable direkt an die lokierte Merkeradresse: <b>AT%MX0.0</b> gemappt.
bSCS	BOOL	Der zu sendende ON/OFF-Wert wird direkt von der an der Merkeradresse <b>AT%MX1.5</b> dieser lokierten SPS-Variablen gemappt.
AODB	ARRAY OF ST_IEC870_5_101AODBEntry (STRUCT)	Applikationsobjektdatenbank. Es ist eine Array-Variable. Für jeden Datenpunkt wird ein Arrayelement benötigt.
hTable	T_HAODBTable (STRUCT)	Applikationsobjektdatenbank-Handle. Mit Hilfe dieser Variablen wird auf die Applikationsobjektdatenbank zugegriffen und nie direkt auf die einzelnen Arrayelemente.
F_iecCreateTableHnd	FUNCTION	Im Initialisierungsschritt werden alle Arrayelemente von AODB mit diesem Funktionsaufruf einmalig initialisiert. Beim Erfolg liefert dieser Funktionsaufruf das Handle.
F_iecAddTableEntry	FUNCTION	Im Initialisierungsschritt werden die einzelnen Datenpunkte der Station mit diesem Funktionsaufruf einmalig konfiguriert.

### 6.5.1.1.3 Tutorial

Dieses Tutorial ist eine ausführliche Schritt für Schritt Anleitung für die Konfiguration und Programmierung einer IEC 60870-5-104 Zentralstation. Den hier aufgeführten Beschreibungen liegt dieses [Tutorial SPS-Beispiel |▶ 647|](#) zugrunde.

#### 6.5.1.1.3.1 Programmierumgebung konfigurieren

Gehen Sie bei der grundlegenden Konfiguration der Programmierumgebung wie folgt vor.

- ✓ Öffnen Sie das Microsoft Visual Studio
  - 1. Gehen Sie auf: File / New / Project
  - 2. Wählen Sie ein TwinCAT Projekt aus
  - 3. Wenn die Solution mit dem TwinCAT Projekt geöffnet ist, können Sie mit einem Rechtsklick auf den Eintrag PLC das Kontextmenu öffnen und Add New Item wählen
  - 4. Wählen Sie ein Standard PLC Project aus
  - 5. Den PLC Project Baum bitte aufklappen und über das Kontextmenu des Unterpunktes References Add library auswählen
  - 6. Fügen Sie die folgenden Bibliotheken als Grundlage Ihrer IEC 60870-5-10x Applikation hinzu:
    - Tc2\_Utilities
    - Tc2\_Tcplp
    - Tc2\_IEC60870\_5\_10x
- ⇒ Damit haben Sie die grundlegende Konfiguration der Programmierumgebung abgeschlossen.

### 6.5.1.1.3.2 Applikationsobjekte definieren und konfigurieren

Mit Applikationsobjekten sind die zu definierenden Datentypen wie Single Points, Double Points, Measured Values, Short Floating Point Values usw. gemeint.

In diesem Beispiel wurden die Befehle so konfiguriert, dass die Prozessdaten der Befehle im gleichen Speicherbereich aber auf einem anderen Byte- und Bit-Offset wie die Daten der Information in Überwachungsrichtung liegen. Sie können aber auch die Befehle auf den gleichen Byte- und Bit-Offset wie die Information in Überwachungsrichtung legen.

Beispiel:

C\_SC\_NA\_1 mit IOA = 10 auf den gleichen Byte- und Bit-Offset wie M\_SP\_NA\_1 mit IOA = 100 (beide Byte-Offset = 100 und Bit-Offset = 0). In diesem Fall würde eine Wertänderung des Single-Points in der Unterstation (M\_SP\_NA\_1 mit der Objektadresse 100) eine Datenübertragung mit der Übertragungsursache <3> (Spontan) zur Leitstation auslösen. In der Leitstation wird der neue Wert auf die Bit- und Byte-Offsetadresse des Befehls (C\_SC\_NA\_1 mit der Objektadresse 10) kopiert und löst eine Befehlsübertragung zur Unterstation aus.

Als Beispiel konfigurieren wir in dem Einführungsprojekt folgende Applikationsobjekte:

ASDU identifizier	Objektadre sse IOA	Group- Konfigurationspar ameter	Basiszeitmulti plikator	SPS- Prozessdatenb ereich	Offse t Byte	Offse t Bit	Prozessdaten breite in der TwinCAT SPS
M_SP_NA _1	100	Generalabfrage	0	Merker	100	0	1 Bit
M_SP_NA _1	101	Generalabfrage	0	Merker	100	1	1 Bit
M_SP_TB _1	102	Generalabfrage	0	Merker	100	2	1 Bit
M_DP_N A_1	200	Generalabfrage	0	Merker	200	0	2 Bits
M_DP_N A_1	201	Generalabfrage	0	Merker	200	2	2 Bits
M_DP_TB _1	202	Generalabfrage	0	Merker	200	4	2 Bits
M_ST_NA _1	300	Generalabfrage	0	Merker	300	0	1 Byte
M_ST_NA _1	301	Generalabfrage	0	Merker	301	0	1 Byte
M_ST_TB _1	302	Generalabfrage	0	Merker	302	0	1 Byte
M_BO_N A_1	400	Generalabfrage	0	Merker	400	0	4 Byte
M_BO_N A_1	401	Generalabfrage	0	Merker	404	0	4 Byte
M_BO_TB _1	402	Generalabfrage	0	Merker	408	0	4 Byte
M_ME_N A_1	500	Generalabfrage	0	Merker	500	0	2 Byte
M_ME_N A_1	501	Generalabfrage	0	Merker	502	0	2 Byte
M_ME_T D_1	502	Generalabfrage	0	Merker	504	0	2 Byte
M_ME_N B_1	600	Generalabfrage	0	Merker	600	0	2 Byte
M_ME_N B_1	601	Generalabfrage	0	Merker	602	0	2 Byte
M_ME_TE _1	602	Generalabfrage	0	Merker	604	0	2 Byte
M_ME_N C_1	700	Generalabfrage	0	Merker	700	0	4 Byte
M_ME_N C_1	701	Generalabfrage	0	Merker	704	0	4 Byte
M_ME_TF _1	702	Generalabfrage	0	Merker	708	0	4 Byte
M_IT_NA _1	800	Generalzählerabfra ge	0	Merker	800	0	4 Byte
M_IT_NA _1	801	Generalzählerabfra ge	0	Merker	804	0	4 Byte
M_IT_TB _1	802	Generalzählerabfra ge	0	Merker	808	0	4 Byte
<b>Comman ds</b>							

C_SC_NA _1	10	-	0	Merker	2100	0	1 Bit
C_SC_NA _1	11	-	0	Merker	2100	1	1 Bit
C_SC_TA _1	12	-	0	Merker	2100	2	1 Bit
C_DC_NA _1	20	-	0	Merker	2200	0	2 Bit
C_DC_NA _1	21	-	0	Merker	2200	2	2 Bit
C_DC_TA _1	22	-	0	Merker	2200	4	2 Bit
C_BO_NA _1	40	-	0	Merker	2400	0	4 Byte
C_BO_NA _1	41	-	0	Merker	2404	0	4 Byte
C_BO_TA _1	42	-	0	Merker	2408	0	4 Byte
C_SE_NA _1	50	-	0	Merker	2500	0	2 Byte
C_SE_NA _1	51	-	0	Merker	2502	0	2 Byte
C_SE_TA _1	52	-	0	Merker	2504	0	2 Byte
C_SE_NB _1	60	-	0	Merker	2600	0	2 Byte
C_SE_NB _1	61	-	0	Merker	2602	0	2 Byte
C_SE_TB _1	62	-	0	Merker	2604	0	2 Byte
C_SE_NC _1	70	-	0	Merker	2700	0	4 Byte
C_SE_NC _1	71	-	0	Merker	2704	0	4 Byte
C_SE_TC _1	72	-	0	Merker	2708	0	4 Byte

### Datenbankvariable deklarieren

Die Applikationsobjekt-Datenbank ist eine Array-Variable vom Typ `ST_IEC870_5_101AODBEntry` [► 482]. Jedes Array-Element entspricht einem Applikationsobjekt. Die Array-Elemente werden aber nicht direkt, sondern nur mit Hilfe der speziell dafür zur Verfügung gestellten Funktionen und einem Datenbank-Handle (Tabellen-Handle) manipuliert. Das Datenbank-Handle muss vor der Benutzung durch einen einmaligen `F_iecCreateTableHnd` [► 452]-Funktionsaufruf initialisiert werden. Dabei werden auch die Array-Elemente miteinander als Hash-Tabelle verknüpft. Bei einer größeren Anzahl der Datenpunkte ermöglicht die Hash-Tabelle einen schnelleren Zugriff auf einen einzelnen Datenpunkt.

Die maximale Anzahl der Applikationsobjekte ist frei wählbar und nur durch den verfügbaren Speicher begrenzt. Sie müssen sich auf eine konstante maximale Anzahl während der SPS-Programmierung festlegen. Zur Laufzeit kann die maximale Anzahl der Applikationsobjekte nicht mehr verändert werden. In unserem Beispiel werden 50 Applikationsobjekte deklariert. Diese Anzahl reicht für die meisten Anwendungen aus. Beachten Sie, dass sehr viele Applikationsobjekte auch entsprechend viel Speicher und Laufzeit benötigen.

Definieren Sie die folgenden Variablen in MAIN:

```
PROGRAM MAIN
VAR
```

```
AODB : ARRAY[0..49] OF ST_IEC870_5_101AODBEntry [▶ 482];
hTable : T_HAODBTable [▶ 531];
END_VAR
```

**Applikationsobjekte konfigurieren**

Die Konfiguration der gewünschten Applikationsobjekte wird zur Programmlaufzeit durchgeführt. Während der Konfiguration werden unter anderem der Objekt-Typ (M\_SP\_NA\_1, M\_DP\_NA\_1, M\_ST\_NA\_1 usw.), die gemeinsame ASDU-Adresse, die Objekt-Adresse und weitere Objekt-Parameter festgelegt.

Nach der Initialisierung des Datenbank-Handles ist die Applikationsobjekt-Datenbank (Datenbank-Array) leer und muss mit den gewünschten Daten (Datenpunkten) gefüllt werden. Die Konfiguration der Datenpunkte der Zentralstation muss der Konfiguration der Datenpunkte in der Unterstation entsprechen! D.h. in der Zentralstation müssen Datenpunkte vom gleichen Typ, gleicher gemeinsamen ASDU-Adresse und mit der gleichen Informationsobjekt-Adresse wie in der Unterstation konfiguriert werden. Andere Parameter wie z.B. der Mapping-Bereich, Byte-, Bit-Offset können beliebig konfiguriert werden.

Es stehen folgende Funktionen zur Manipulation der Applikationsdatenbank zur Verfügung:

Funktion	Beschreibung
<a href="#">F_iecCreateTableHnd [▶ 452]</a>	Initialisiert das Hash-Tabellenhandle
<a href="#">F_iecAddTableEntry [▶ 453]</a>	Konfiguriert und fügt einen neuen Hash-Tabelleneintrag ein
<a href="#">F_iecRemoveTableEntry [▶ 460]</a>	Entfernt einen Hash-Tabelleneintrag
<a href="#">F_iecLookupTableEntry [▶ 458]</a>	Prüft ob ein bestimmter Hash-Tabelleneintrag existiert
<a href="#">F_iecGetPosOfTableEntry [▶ 456]</a>	Ermittelt die lineare Position eines Hash-Tabelleneintrags

Das Datenbank-Handle muss an die Funktion per VAR\_IN\_OUT übergeben werden. Im Regelfall wird die Konfiguration beim SPS-Programmstart einmalig in einer Init-Routine durchgeführt.

Um die Applikationsobjekte beim Programmstart zu konfigurieren wird in MAIN folgender SPS-Code hinzugefügt:

```
PROGRAM MAIN
VAR
AODB : ARRAY[0..49] OF ST_IEC870_5_101AODBEntry [▶ 482];
hTable : T_HAODBTable [▶ 531];

init : BOOL := TRUE;
initError : UDINT;
asduAddr : UDINT := 7;
END_VAR

IF init THEN
init := FALSE;

initError := F_iecCreateTableHnd [▶ 452]( ADR( AODB ), SIZEOF( AODB ), hTable );
IF initError <> 0 THEN
ADSLOGSTR( ADSLOG_MSGTYPE_HINT OR ADSLOG_MSGTYPE_LOG,
'F_iecCreateTableHnd() error: %s',
DWORD_TO_HEXSTR( initError, 8, FALSE ) );
RETURN;
END_IF

(* Monitored Single Points *)
initError := F_iecAddTableEntry( M_SP_NA_1, asduAddr, 100, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
100, 0, 0, hTable );
initError := F_iecAddTableEntry( M_SP_NA_1, asduAddr, 101, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
100, 1, 0, hTable );
initError := F_iecAddTableEntry( M_SP_TB_1, asduAddr, 102, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
100, 2, 0, hTable );
(* Double Points*)
initError := F_iecAddTableEntry( M_DP_NA_1, asduAddr, 200, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
200, 0, 0, hTable );
initError := F_iecAddTableEntry( M_DP_NA_1, asduAddr, 201, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
200, 2, 0, hTable );
```

```

initError := F_iecAddTableEntry( M_DP_TB_1, asduAddr, 202, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
200, 4, 0, hTable );
(* Regulating step value *)
initError := F_iecAddTableEntry( M_ST_NA_1, asduAddr, 300, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
300, 0, 0, hTable );
initError := F_iecAddTableEntry( M_ST_NA_1, asduAddr, 301, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
301, 0, 0, hTable );
initError := F_iecAddTableEntry( M_ST_TB_1, asduAddr, 302, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
302, 0, 0, hTable );
(* 32 bit string *)
initError := F_iecAddTableEntry( M_BO_NA_1, asduAddr, 400, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
400, 0, 0, hTable );
initError := F_iecAddTableEntry( M_BO_NA_1, asduAddr, 401, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
404, 0, 0, hTable );
initError := F_iecAddTableEntry( M_BO_TB_1, asduAddr, 402, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
408, 0, 0, hTable );
(* Measured value, normalized value *)
initError := F_iecAddTableEntry( M_ME_NA_1, asduAddr, 500, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
500, 0, 0, hTable );
initError := F_iecAddTableEntry( M_ME_NA_1, asduAddr, 501, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
502, 0, 0, hTable );
initError := F_iecAddTableEntry( M_ME_TD_1, asduAddr, 502, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
504, 0, 0, hTable );
(* Mesured value, scaled value *)
initError := F_iecAddTableEntry( M_ME_NB_1, asduAddr, 600, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
600, 0, 0, hTable );
initError := F_iecAddTableEntry( M_ME_NB_1, asduAddr, 601, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
602, 0, 0, hTable );
initError := F_iecAddTableEntry( M_ME_TE_1, asduAddr, 602, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
604, 0, 0, hTable );
(* Measured value , short floating point value *)
initError := F_iecAddTableEntry( M_ME_NC_1, asduAddr, 700, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
700, 0, 0, hTable );
initError := F_iecAddTableEntry( M_ME_NC_1, asduAddr, 701, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
704, 0, 0, hTable );
initError := F_iecAddTableEntry( M_ME_TF_1, asduAddr, 702, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY,
708, 0, 0, hTable );
(* Integrated totals *)
initError := F_iecAddTableEntry( M_IT_NA_1, asduAddr, 800, IEC870_GRP_REQCOGEN, 0, MAP_AREA_MEMORY,
800, 0, 0, hTable );
initError := F_iecAddTableEntry( M_IT_NA_1, asduAddr, 801, IEC870_GRP_REQCOGEN, 0, MAP_AREA_MEMORY,
804, 0, 0, hTable );
initError := F_iecAddTableEntry( M_IT_TB_1, asduAddr, 802, IEC870_GRP_REQCOGEN, 0, MAP_AREA_MEMORY,
808, 0, 0, hTable );

(* Single commands *)
initError := F_iecAddTableEntry( C_SC_NA_1, asduAddr, 10, 0, 0, MAP_AREA_MEMORY, 2100, 0, 0,
hTable );
initError := F_iecAddTableEntry( C_SC_NA_1, asduAddr, 11, 0, 0, MAP_AREA_MEMORY, 2100, 1, 0,
hTable );
initError := F_iecAddTableEntry( C_SC_TA_1, asduAddr, 12, 0, 0, MAP_AREA_MEMORY, 2100, 2, 0,
hTable );
(* Double commands *)
initError := F_iecAddTableEntry( C_DC_NA_1, asduAddr, 20, 0, 0, MAP_AREA_MEMORY, 2200, 0, 0,
hTable );
initError := F_iecAddTableEntry( C_DC_NA_1, asduAddr, 21, 0, 0, MAP_AREA_MEMORY, 2200, 2, 0,
hTable );
initError := F_iecAddTableEntry( C_DC_TA_1, asduAddr, 22, 0, 0, MAP_AREA_MEMORY, 2200, 4, 0,
hTable );
(* 32 bit string commands *)
initError := F_iecAddTableEntry( C_BO_NA_1, asduAddr, 40, 0, 0, MAP_AREA_MEMORY, 2400, 0, 0,
hTable );
initError := F_iecAddTableEntry( C_BO_NA_1, asduAddr, 41, 0, 0, MAP_AREA_MEMORY, 2404, 0, 0,
hTable );
initError := F_iecAddTableEntry( C_BO_TA_1, asduAddr, 42, 0, 0, MAP_AREA_MEMORY, 2408, 0, 0,
hTable );
(* Set point, normalized values*)
initError := F_iecAddTableEntry( C_SE_NA_1, asduAddr, 50, 0, 0, MAP_AREA_MEMORY, 2500, 0, 0,
hTable );
initError := F_iecAddTableEntry( C_SE_NA_1, asduAddr, 51, 0, 0, MAP_AREA_MEMORY, 2502, 0, 0,
hTable );
initError := F_iecAddTableEntry( C_SE_TA_1, asduAddr, 52, 0, 0, MAP_AREA_MEMORY, 2504, 0, 0,
hTable );
(* Set point, scaled values *)
initError := F_iecAddTableEntry( C_SE_NB_1, asduAddr, 60, 0, 0, MAP_AREA_MEMORY, 2600, 0, 0,
hTable );
initError := F_iecAddTableEntry( C_SE_NB_1, asduAddr, 61, 0, 0, MAP_AREA_MEMORY, 2602, 0, 0,
hTable );
initError := F_iecAddTableEntry( C_SE_TB_1, asduAddr, 62, 0, 0, MAP_AREA_MEMORY, 2604, 0, 0,
hTable );

```

```
(* Set point, short floating point values *)
initError := F_iecAddTableEntry( C_SE_NC_1, asduAddr, 70, 0, 0, MAP_AREA_MEMORY, 2700, 0, 0,
hTable );
initError := F_iecAddTableEntry( C_SE_NC_1, asduAddr, 71, 0, 0, MAP_AREA_MEMORY, 2704, 0, 0,
hTable );
initError := F_iecAddTableEntry( C_SE_TC_1, asduAddr, 72, 0, 0, MAP_AREA_MEMORY, 2708, 0, 0,
hTable );

END_IF
```

Das zugehörige Tutorial SPS-Beispiel kann [hier \[► 647\]](#) heruntergeladen werden.

### 6.5.1.1.3.3 Mapping der SPS- und IEC-Prozessdaten

Die TwinCAT SPS-Prozessdaten werden zur Programmlaufzeit zyklisch in die IEC-Prozessdaten (Applikationsobjekte) und umgekehrt gemappt (kopiert). Für das Mapping der IEC<->SPS Prozessdaten können bis zu 4 Prozessdatenbereiche (IO-Eingänge, IO-Ausgänge, Merkerbereich, Datenbereich) als Puffervariablen im SPS-Programm deklariert werden. Die Bytegröße der Puffer ist frei wählbar und kann für jeden Bereich unterschiedlich gewählt werden. Unbenutzte Bereiche müssen nicht unbedingt deklariert werden.

In unserem Einführungsbeispiel deklarieren wir 4 SPS-Prozessdatenbereiche mit jeweils 3000 Bytes:

```
PROGRAM MAIN
VAR
AODB : ARRAY[0..49] OF ST_IEC870_5_101AODBEntry [► 482];
hTable : T_HAODBTable;

init : BOOL := TRUE;
initError : UDINT;
asduAddr : UDINT := 7;

inputs AT%IB0 : ARRAY[0..2999] OF BYTE;
outputs AT%QB0 : ARRAY[0..2999] OF BYTE;
memory AT%MB0 : ARRAY[0..2999] OF BYTE;
data : ARRAY[0..2999] OF BYTE;

END_VAR
```

Die Zuordnung, wie die Prozessdaten zur Laufzeit gemappt werden sollen, wird während der Konfiguration der Applikationsobjekte mit der [F\\_iecAddTableEntry \[► 453\]](#)-Funktion festgelegt.

Siehe auch in: [Applikationsobjekte definieren und konfigurieren \[► 634\]](#).

Die Puffervariablen wurden nun als Byte-Arrays deklariert. Um auf die gewünschten Daten besser zugreifen zu können definieren wir die einzelnen Variablen ein zweites Mal und legen diese auf die entsprechenden Byte/Bit-Offsetadressen. Bei einer Änderung im Byte-Array wird die entsprechende einzelne Variable gleichzeitig geändert und umgekehrt. Dies ist aber nicht zwingend notwendig. Sie können direkt auf die Bytes/Bits der Byte-Array-Puffervariablen zugreifen.

```
VAR GLOBAL(* Memory offset 0..99 unused *)
(* Single points *)
msgSingle_0 AT%MX100.0 : BOOL;
msgSingle_1 AT%MX100.1 : BOOL;
msgSingle_2 AT%MX100.2 : BOOL;

(* Double points *)
(* Bit 0..1 = first double point,
Bit 2..3 = second double point,
Bit 4..5 = third double point,
Bit 6..7 = fourth double point *)
msgDouble_0 AT%MB200 : BYTE;

(* Regulating step values *)
msgStep_0 AT%MB300 : BYTE;
msgStep_1 AT%MB301 : BYTE;
msgStep_2 AT%MB302 : BYTE;

(* 32 bit strings *)
msgBitStr_0 AT%MD400 : DWORD;
msgBitStr_1 AT%MD404 : DWORD;
msgBitStr_2 AT%MD408 : DWORD;
```

```

(* Measured values, normalized values *)
msgNormalized_0 AT%MW500 : WORD;
msgNormalized_1 AT%MW502 : WORD;
msgNormalized_2 AT%MW504 : WORD;

(* Measured values, scaled values *)
msgScaled_0 AT%MW600 : INT;
msgScaled_1 AT%MW602 : INT;
msgScaled_2 AT%MW604 : INT;

(* Measured values, short floating point values *)
msgFloating_0 AT%MD700 : REAL;
msgFloating_1 AT%MD704 : REAL;
msgFloating_2 AT%MD708 : REAL;

(* Integrated totals *)
msgTotal_0 AT%MD800 : UDINT;
msgTotal_1 AT%MD804 : UDINT;
msgTotal_2 AT%MD808 : UDINT;

(* Single commands *)
cmdSingle_0 AT%MX2100.0 : BOOL;
cmdSingle_1 AT%MX2100.1 : BOOL;
cmdSingle_2 AT%MX2100.2 : BOOL;

(* Double commands *)
(* Bit 0..1 = first double command,
Bit 2..3 = second double command,
Bit 4..5 = third double command,
Bit 6..7 = fourth double command *)
cmdDouble_0 AT%MB2200 : BYTE;

(* 32 bit string commands *)
cmdBitStr_0 AT%MD2400 : DWORD;
cmdBitStr_1 AT%MD2404 : DWORD;
cmdBitStr_2 AT%MD2408 : DWORD;

(* Set point, normalized values *)
cmdNormalized_0 AT%MW2500 : WORD;
cmdNormalized_1 AT%MW2502 : WORD;
cmdNormalized_2 AT%MW2504 : WORD;

(* Set point, scaled values *)
cmdScaled_0 AT%MW2600 : INT;
cmdScaled_1 AT%MW2602 : INT;
cmdScaled_2 AT%MW2604 : INT;

(* Set point, short floating point values *)
cmdFloating_0 AT%MD2700 : REAL;
cmdFloating_1 AT%MD2704 : REAL;
cmdFloating_2 AT%MD2708 : REAL;
END_VAR

```

## Mapping der IEC<->SPS Prozessdaten in der Zentralstation

### Prozessdaten in Überwachungsrichtung (Slave->Master Information)

#### Beispiel 1

Single point information (M\_SP\_NA\_1) mit der IOA = 100, SPS Merkerbereich, Byteoffset = 100, Bitoffset = 0.

Unterstation -> ... -> Zentralstation FB -> Memory[100].0 == msgSingle\_0

#### Beispiel 2

Measured value, short floating point value (M\_ME\_NC\_1) mit der IOA = 700, SPS Merkerbereich, Byteoffset = 700, Bitoffset = 0 (bedeutungslos).

Unterstation -> ... -> Zentralstation FB -> Memory[700..703] == msgFloating\_0

### Prozessdaten in Steuerungsrichtung (Master->Slave Commands)

#### Beispiel 1



Single command state (C\_SC\_NA\_1) mit der IOA = 10, SPS Merkerbereich, Byteoffset = 2100, Bitoffset = 0.

cmdSingle\_0 == memory[2100].0 -> Zentralstation FB -> ... -> Unterstation

Beispiel 2

Set point, short floating point value (C\_SE\_NC\_1) mit der IOA = 70, SPS Merkerbereich, Byteoffset = 2700, Bitoffset = 0 (bedeutungslos).

cmdFloating\_0 == memory[2700..2703] -> Zentralstation FB -> ... -> Unterstation

Das zugehörige Tutorial SPS-Beispiel kann [hier \[▶ 647\]](#) heruntergeladen werden.

#### 6.5.1.1.3.4 Aufruf und Deklaration der Zentralstation

Die gesamte Funktionalität einer Zentralstation ist im Funktionsbaustein FB\_IEC870\_5\_104Master gekapselt. Mit einer Instanz kann eine Verbindung zur Unterstation aufgebaut werden. Für eine weitere Verbindung deklarieren Sie eine weitere Instanz des Funktionsbausteins.

Fügen Sie im Deklarationsteil von MAIN folgenden SPS-Code ein:

```
PROGRAM MAIN
VAR
AODB : ARRAY[0..49] OF ST_IEC870_5_101AODBEntry;
hTable : T_HAODBTable;

init : BOOL := TRUE;
initError : UDINT;
asduAddr : UDINT := 7;

inputs AT%IB0 : ARRAY[0..2999] OF BYTE;
outputs AT%QB0 : ARRAY[0..2999] OF BYTE;
memory AT%MB0 : ARRAY[0..2999] OF BYTE;
data : ARRAY[0..2999] OF BYTE;

bEnable : BOOL := TRUE;
client : FB_IEC870_5_104Master;

END_VAR
```

Im Programmteil wird die Instanz aufgerufen:

```
IF init THEN
init := FALSE;
...

ELSE
...
client( pAOEntries := ADR( AODB ),
cbAOEntries := SIZEOF( AODB ),
pInputs := ADR( inputs ),
cbInputs := SIZEOF( inputs ),
pOutputs := ADR( outputs ),
cbOutputs := SIZEOF( outputs ),
pMemory := ADR( memory ),
cbMemory := SIZEOF( memory ),
pData := ADR( data ),
cbData := SIZEOF( data ),
bEnable := bEnable,
hTable := hTable );
...
END_IF
```

Das zugehörige Tutorial SPS-Beispiel kann [hier \[▶ 647\]](#) heruntergeladen werden.

#### 6.5.1.1.3.5 Basiseinstellungen: Protokollparameter

Über die IEC60870-5-104-Protokollparameter kann das Verhalten der Zentralstation an die Anforderungen der Unterstation angepasst werden. Die meisten Parameter sind mit Defaultwerten vorbelegt, so dass diese nicht verändert werden müssen.

Im unserem Beispiel verändern wir die Werte der iK- und iW-Parameter und konfigurieren die IP-Adresse und Portnummer der Unterstation zu der die Verbindung aufgebaut werden soll.

```
IF init THEN
    init := FALSE;
    ...

    client.protPara.sRemoteHost := '127.0.0.1';
    client.protPara.nRemotePort := 2404;
    client.protPara.iK := 12;
    client.protPara.iW := 8;
    client.protPara.bThrottleMode := TRUE;
    client.protPara.bPackFrames := TRUE;

ELSE
    client( pInputs := ADR( inputs ),
           cbInputs := SIZEOF( inputs ),
           pOutputs := ADR( outputs ),
           ...
    );
END_IF
```

Die Dokumentation aller Übertragungsprotokoll-Parameter ist unter [ST IEC870 5 104ProtocolParams \[► 549\]](#) zu finden.

Das zugehörige Tutorial SPS-Beispiel kann [hier \[► 647\]](#) heruntergeladen werden.

### 6.5.1.1.3.6 Basiseinstellungen: Systemparameter

Über die Systemparameter werden z.B. die gemeinsame ASDU-Adresse und die Anwenderfunktionen der Zentralstation konfiguriert. In unserer Einführung konfigurieren wir folgende Systemparameter:

Die gemeinsame ASDU-Adresse wird auf 7 gesetzt. (*asduAddr*)

Das loggen der Debug-Meldungen im Application-Log wird aktiviert (*dbgMode*). Es werden Änderungen im Gerätestatus geloggt.

Fügen Sie folgenden SPS-Code in Ihr SPS-Projekt ein:

```
IF init THEN
    init := FALSE;
    ...

    client.sysPara.asduAddr := 7;
    client.sysPara.dbgMode := IEC870_DEBUGMODE_DEVSTATE (* OR IEC870_DEBUGMODE_LINKERROR OR
    IEC870_DEBUGMODE_ASDU OR IEC870_DEBUGMODE_LINKLAYER *);
    ...
ELSE
    client( pInputs := ADR( inputs ),
           cbInputs := SIZEOF( inputs ),
           pOutputs := ADR( outputs ),
           ...
    );
END_IF
```

Die Dokumentation aller Systemparameter finden Sie hier: [ST IEC870 5 101SystemParams \[► 487\]](#).

Das zugehörige Tutorial SPS-Beispiel kann [hier \[► 647\]](#) heruntergeladen werden.

### 6.5.1.1.3.7 Initialisierungssequenz

Die Initialisierungssequenz sieht wie folgt aus.

```
client.acqPara.arrInitSeq[0] := eIEC870_ISTEP_TEST; (* Send test command *)
client.acqPara.arrInitSeq[1] := eIEC870_ISTEP_CLOCK; (* Send clock synchronization command *)
client.acqPara.arrInitSeq[2] := eIEC870_ISTEP_GENRO; (* Send general interrogation command *)
client.acqPara.arrInitSeq[3] := eIEC870_ISTEP_CORO; (* Send counter interrogation command *)
client.acqPara.arrInitSeq[5] := eIEC870_ISTEP_UNUSED; (* not used *)
```

Das zugehörige Tutorial SPS-Beispiel kann [hier \[► 647\]](#) heruntergeladen werden.

### 6.5.1.1.3.8 Stationsabfrage / Generalabfrage

Der Stationsabfragebefehl oder auch Generalabfrage genannt, wird von der Zentralstation eingeleitet. Im Kennungsfeld des Befehls ist auch die Gruppe (1 bis 16 oder allgemein) festgelegt. Die Unterstation überträgt die zu dieser Gruppe dazugehörigen Applikationsobjekte mit der Übertragungsursache <20> bis <36> an die Zentralstation.

#### ● Zeitmarken bei Generalabfragen

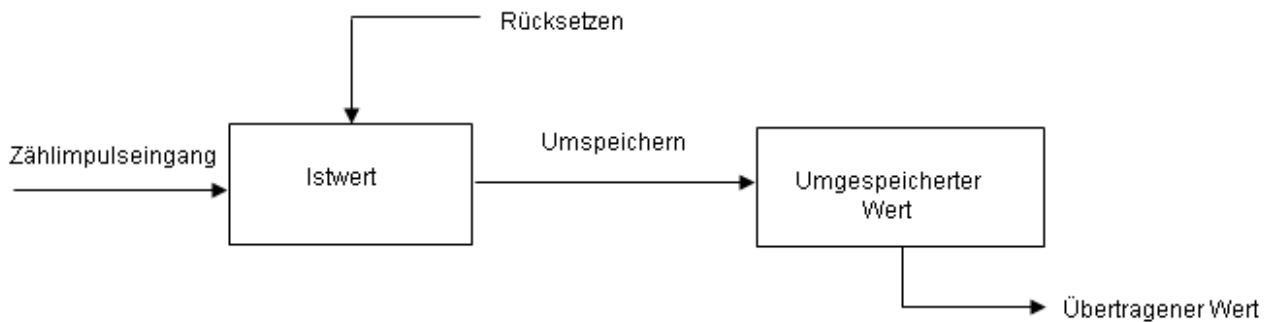
**i** Applikationsobjekte mit Zeitmarken werden bei der Generalabfrage ohne Zeitmarken übertragen.

```
client.acqPara.arrGenro[0].tPollCycle := T#60s;
client.acqPara.arrGenro[0].eQOI := eIEC870_QOI_INROGEN;
client.acqPara.arrGenro[0].bEnable := TRUE;
```

Das zugehörige Tutorial SPS-Beispiel kann [hier \[▶ 647\]](#) heruntergeladen werden.

### 6.5.1.1.3.9 Zählwertübertragung

Zur Erläuterung ein allgemeines Modell für die Zählwertübertragung:



Die Istwerte werden durch Zähler aufsummiert. Die Istwerte können durch einen Umspeicherbefehl, der entweder von der Zentralstation empfangen oder örtlich (lokal in der Unterstation) erzeugt wird, periodisch in umgespeicherte Werte umgespeichert (kopiert) werden. Nach dem Umspeichern wird der erfasste Wert entweder auf Null zurückgesetzt (Erfassen von Inkrementalwerten) oder der Zähler fährt mit seinem Betrieb fort (Erfassen von Zählerständen).

Applikationsobjekte mit Zählwerten werden Gruppen zugeordnet. Die Gruppen werden einzeln umgespeichert (frozen), zurückgesetzt (reset) oder übertragen. Die Zentralstation sendet Zählwertabfragebefehle an die Unterstation. In einem Kennungsfeld des Befehls (QCC) wird die durchzuführende Aktion (FRZ) und Gruppe (RQT) festgelegt.

Die Zuordnung der Applikationsobjekte zu den einzelnen Gruppen (1 bis 4 oder allgemein) wird während der Konfiguration durch den Group-Flagparameter festgelegt. Es gibt vier Betriebsarten für die Erfassung von Zählerständen und Inkrementalwerten. Zu jeder Betriebsart sind einige Hinweise zur Konfiguration der Systemparameter oder der Applikationsobjekte aufgeführt.

#### Betriebsart A: Örtlich Umspeichern mit Spontanübertragung

Die Unterstation initiiert intern das Umspeichern oder Umspeichern mit Rücksetzen. Die umgespeicherten Zählwerte werden spontan übertragen, nachdem die Funktion Umspeichern oder Umspeichern mit Rücksetzen durchgeführt wurde. Die Zentralstation gibt in dieser Betriebsart keine Zählwertabfragebefehle aus.

Konfiguration der Systemparameter:

Konfiguration der Applikationsobjekte:

### Betriebsart B: Örtlich Umspeichern mit Zählerabfrage

Die Unterstation initiiert intern das Umspeichern oder Umspeichern mit Rücksetzen. Die umgespeicherten Zählwerte werden durch Zählwertabfragebefehle von der Zentralstation abgefragt. In diesem Fall darf die Zentralstation im Befehlskennungsfeld das Umspeichern oder Umspeichern mit Rücksetzen nicht benutzen (FRZ=0). Die Zählwerte werden allgemein oder in Gruppen (groups) 1 bis 4 abgefragt.

Konfiguration der Systemparameter:

Konfiguration der Applikationsobjekte:

### Betriebsart C: Zentralstation leitet das Umspeichern, Umspeichern mit Rücksetzen oder Rücksetzen ein

Ein Zählwertabfragebefehl wird periodisch von der Zentralstation an die Unterstation ausgegeben, um das Umspeichern oder (und) Rücksetzen zu steuern. Dieser Befehl hat aber noch keine Übertragung der Zählwerte zur Folge. Erst ein nachfolgender Zählwertabfragebefehl wird von der Zentralstation gesendet um die umgespeicherten Zählwerte einzusammeln. Ähnlich, wie bei der Betriebsart B.

Konfiguration der Systemparameter:

Konfiguration der Applikationsobjekte:

### Betriebsart D: Zentralstation leitet das Umspeichern und (oder) Rücksetzen ein und die umgespeicherten Werte werden spontan übertragen

Diese Betriebsart ist eine Kombination des Zählwertbefehls von der Zentralstation wie für Betriebsart C mit einer spontanen Übertragung der Zählwerte wie bei der Betriebsart A.

Konfiguration der Systemparameter:

Konfiguration der Applikationsobjekte:

Das zugehörige Tutorial SPS-Beispiel kann [hier \[► 647\]](#) heruntergeladen werden.

## 6.5.1.1.3.10 Uhrzeitsynchronisation

In den [Acquisitions-Parametern \[► 491\]](#) stehen die Konfigurationsparameter für den zyklischen Datenaustausch. Auch die Uhrzeitsynchronisation kann dort eingeschaltet und konfiguriert werden. Im Initialisierungsschritt müssen folgende Werte gesetzt werden:

```
//activate clock sync
fbMaster.acqPara.clockSync.bEnable := TRUE;
fbMaster.acqPara.clockSync.tPollCycle := T#120S;
```

Des Weiteren können bezüglich der Zeitsynchronisation einige Einstellungen in den [Systemparametern \[► 487\]](#) vorgenommen werden. Für die Zentralstation ist relevant:

**bSyncPCTime:** Wenn der Wert TRUE ist, soll neben der IEC-Systemzeit auch die Systemzeit des TwinCAT PCs (die Windows-Systemzeit in der Taskleiste) synchronisiert werden. Der Default-Wert ist FALSE.

Das zugehörige Tutorial SPS-Beispiel kann [hier \[► 647\]](#) heruntergeladen werden.

## 6.5.1.1.3.11 Test der Kommunikation

Dieses Kapitel bezieht sich direkt auf das Tutorial Beispiel. Durch das Setzen der *bExecuteCmd*-Variable auf TRUE wird eine einfache Simulation der Befehle in Steuerungsrichtung aktiviert und mit FALSE deaktiviert. Bei einer aktiven Verbindung wird in diesem Beispiel ein Einzel-Befehl (C\_SC\_NA\_1, IOA = 10) zyklisch alle 10 Sekunden zur Unterstation übertragen.

```
PROGRAM MAIN
VAR
...

bExecuteCmd : BOOL;
timer : TON;
```

```

...
END_VAR

...

(* Simple command simulation *)
timer( IN := bExecuteCmd, PT := T#10s ); (* Send cyclic command *)
IF timer.Q THEN
timer( IN := FALSE );
cmdSingle_0 := NOT cmdSingle_0; (* toggle single command ON<->OFF *)

(* cmdDouble_0 := SEL( cmdDouble_0 = 1, 1, 2 );

cmdBitStr_0 := cmdBitStr_0 + 1;

cmdNormalized_0 := cmdNormalized_0 + 2;

cmdScaled_0 := cmdScaled_0 + 4;

cmdFloating_0 := cmdFloating_0 + 1.2; *)
END_IF

...

```

Das zugehörige Tutorial SPS-Beispiel kann [hier \[► 647\]](#) heruntergeladen werden.

### 6.5.1.1.3.12 Protokoll- und Stationsfehler

Die Stationsfehlermeldungen werden in einem FIFO abgelegt. Es können bis zu 10 Fehlermeldungen zwischengespeichert werden. Bei fatalen Kommunikationsfehlern (z.B. Fehler der Verbindungsschicht, die Checksumme des Frames passt nicht) wird die Verbindung unterbrochen und muss neu aufgebaut werden. Fehler in der Applikationsschicht (z.B. der ASDU-Sendepuffer ist wegen zu vieler Frames übergelaufen) werden nur geloggt und führen nicht zum Verbindungsabbruch. Es ist immer noch möglich auch bei diesen Fehlern die Verbindung aus der Applikation heraus zu unterbrechen. Neben dem Fehler-Code wird auch die Fehlerquelle in der Fehlermeldung abgelegt. Dieses erleichtert die Lokalisierung des Fehlers.

#### Beispiel

Die anfallenden Fehlermeldungen einer 60870-5-104 Zentralstation können durch folgenden Aufruf ausgelesen werden:

```

PROGRAM MAIN
VAR
...

client : FB_IEC870_5_104Master;

...
END_VAR

...

REPEAT
client.system.device.errors.RemoveError( );
IF client.system.device.errors.bOk THEN
ADSLOGSTR( ADSLOG_MSGTYPE_ERROR OR ADSLOG_MSGTYPE_LOG,
'IEC60870-5-104 master error: 0x%s',
DWORD_TO_HEXSTR( client.system.device.errors.getError.nErrId, 8, FALSE ) );
END_IF
UNTIL NOT client.system.device.errors.bOk
END_REPEAT

...

```

Das zugehörige Tutorial SPS-Beispiel kann [hier \[► 647\]](#) heruntergeladen werden.

### 6.5.1.1.3.13 Erweiterte Diagnose

Debug-Meldungen, die ins Application-Log geschrieben werden erleichtern eine Fehlersuche im System. Zurzeit können vier Stufen der Debug-Meldungen in einer IEC-Applikation aktiviert werden. Die Meldungen werden durch den dbgMode-Systemparameter der Leitstation aktiviert ([ST\\_IEC870\\_5\\_101SystemParams \[► 487\]](#)).

Debug-Meldungen, die beim Aufbauen oder Abbauen der TCP/IP-Verbindung geloggt werden und Link-Layer-Fehlermeldungen (dbgMode: IEC870\_DEBUGMODE\_LINKERROR);

Stationsstatus-Meldungen (dbgMode: IEC870\_DEBUGMODE\_DEVSTATE);

Hexadezimale Ausgabe der ASDU's (dbgMode: IEC870\_DEBUGMODE\_ASDU). Pro Zeile werden 32-ASDU-Datenbytes als hexadezimale Zahlen ausgegeben. Längere ASDU's werden auf mehrere Zeilen verteilt.

Hexadezimale Ausgabe der APDU's (TCP/IP Telegramme, dbgMode: IEC870\_DEBUGMODE\_LINKLAYER). Pro Zeile werden 32-APDU-Datenbytes als hexadezimale Zahlen ausgegeben. Ähnlich wie bei 3. werden längere APDU's auf mehrere Zeilen verteilt.

Um die aktivierten Debug-Meldungen zu sehen starten Sie TwinCAT System Manager und aktivieren Sie die Loggeransicht. Die nachfolgende Grafik zeigt eine Debug-Ausgabe. Die ersten drei unterschiedlichen Typen der Meldungen wurden mit entsprechenden Zahlen markiert.

Server (Port)	Timestamp	Meldung
TCPLC (80...	4/25/2005 1:08:29 P...	Class1 TX FIFO==>6B 01 07 01 07 00 00 00 00 00 00 63 7F 08 0D 39 04 05
TCPLC (80...	4/25/2005 1:08:29 P...	Class1 TX FIFO<==6B 01 07 01 07 00 00 00 00 00 00 63 7F 08 0D 39 04 05
TCPLC (80...	4/25/2005 1:08:29 P...	RX Fifo==>6B 01 06 01 07 00 00 00 00 00 00 AF 7B 08 0D 19 04 05
TCPLC (80...	4/25/2005 1:08:29 P...	RX Fifo<==6B 01 06 01 07 00 00 00 00 00 00 AF 7B 08 0D 19 04 05
TCPLC (80...	4/25/2005 1:08:29 P...	Class1 TX FIFO==>64 01 0A 01 07 00 00 00 00 14
TCPLC (80...	4/25/2005 1:08:29 P...	Class1 TX FIFO<==64 01 0A 01 07 00 00 00 00 14
TCPLC (80...	4/25/2005 1:08:29 P...	Class1 TX FIFO==>0D B2 14 01 07 00 BC 02 00 00 00 00 00 00 00 00 00 00
TCPLC (80...	4/25/2005 1:08:29 P...	Class1 TX FIFO<==0B 82 14 01 07 00 58 02 00 00 00 00 00 00 00
TCPLC (80...	4/25/2005 1:08:29 P...	Class1 TX FIFO==>09 B2 14 01 07 00 F4 01 00 00 00 00 00 00 00
TCPLC (80...	4/25/2005 1:08:29 P...	Class1 TX FIFO<==0D B2 14 01 07 00 BC 02 00 00 00 00 00 00 00 00
TCPLC (80...	4/25/2005 1:08:29 P...	Class1 TX FIFO==>07 B2 14 01 07 00 90 01 00 00 00 00 00 00 00 00
TCPLC (80...	4/25/2005 1:08:29 P...	Class1 TX FIFO==>05 B2 14 01 07 00 2C 01 00 00 00 00 00
TCPLC (80...	4/25/2005 1:08:29 P...	Class1 TX FIFO==>03 B2 14 01 07 00 C8 00 00 00 00
TCPLC (80...	4/25/2005 1:08:29 P...	Class1 TX FIFO==>01 B2 14 01 07 00 64 00 00 00 00
TCPLC (80...	4/25/2005 1:08:29 P...	Class1 TX FIFO<==0B 82 14 01 07 00 58 02 00 00 00 00 00 00
TCPLC (80...	4/25/2005 1:08:29 P...	Class1 TX FIFO<==09 B2 14 01 07 00 F4 01 00 00 00 00 00 00
TCPLC (80...	4/25/2005 1:08:29 P...	Class1 TX FIFO<==07 B2 14 01 07 00 90 01 00 00 00 00 00 00 00
TCPLC (80...	4/25/2005 1:08:29 P...	Class1 TX FIFO<==05 B2 14 01 07 00 2C 01 00 00 00 00 00
TCPLC (80...	4/25/2005 1:08:29 P...	Class1 TX FIFO<==03 B2 14 01 07 00 C8 00 00 00 00
TCPLC (80...	4/25/2005 1:08:29 P...	Class1 TX FIFO<==01 B2 14 01 07 00 64 00 00 00 00
TCPLC (80...	4/25/2005 1:08:29 P...	Class1 TX FIFO==>64 01 07 01 07 00 00 00 00 14
TCPLC (80...	4/25/2005 1:08:29 P...	Class1 TX FIFO<==64 01 07 01 07 00 00 00 00 14
TCPLC (80...	4/25/2005 1:08:29 P...	RX Fifo==>64 01 06 01 07 00 00 00 00 14
TCPLC (80...	4/25/2005 1:08:29 P...	RX Fifo<==64 01 06 01 07 00 00 00 00 14
TCPLC (80...	4/25/2005 1:08:29 P...	Class1 TX FIFO==>46 01 04 00 07 00 00 00 00 00
TCPLC (80...	4/25/2005 1:08:29 P...	Class1 TX FIFO<==46 01 04 00 07 00 00 00 00 00
TCPLC (80...	4/25/2005 1:08:29 P...	TcIEC870_5_104Slave.Lib::ESTABLISHED (STOPDT) => ESTABLISHED (STARTDT) [2]
TCPLC (80...	4/25/2005 1:08:29 P...	TcIEC870_5_104Slave.Lib::PASSIVE_OPEN => ESTABLISHED (STOPDT) [2]
TCPLC (80...	4/25/2005 1:08:29 P...	TcSocketHelper.lib::FB_SocketAccept::internal handle:2, error code: 0

Abb. 1:

Weitere Diagnose-Tools stehen zur Verfügung:

TwinCAT ADS Monitor

Netzwerkmonitor

Wireshark

Ethereal

Diverse Protokoll-Test-Suit-Produkte

**6.5.1.1.3.14 Glossar**

Begriff	Beschreibung
Unterstation, Slave, Server	Synonyme für eine untergeordnete Station (sie wird überwacht)
Zentralstation, Leitstation, Master, Client	Synonyme für eine übergeordnete Station (Leitstation, sie überwacht andere Stationen)
Steuerungsrichtung	Datenübertragungsrichtung von der Zentralstation zur Unterstation
Überwachungsrichtung	Datenübertragungsrichtung von der Unterstation zur Zentralstation
Applikationsobjekte	IEC-Informationsobjekte in der TwinCAT SPS-Applikation (Single Points, Double Points, Measured Values, Short Floating Point Values usw.)
APDU	Protokolldateneinheit der Anwendungsschicht (application protocol data unit)
APCI	Protokollsteuerinformation der Anwendungsschicht (application protocol control information)
ASDU	Dienstdateneinheit der Anwendungsschicht (application service data unit)
IOA, Adresse des Informationsobjekts	Adresse des Single Points, Double Points usw. (information object address)

**6.5.1.1.4 Zentralstation Tutorial (Master)**

Eine komplexere Zentralstation-Applikation mit unterschiedlichen Datenpunkten in beide Richtungen und einer Verbindung zur Unterstation. Die Beispielapplikation verwendet das TwinCAT IEC 60870-5-104 High-Level Interface.

Download TwinCAT XAE Project (\*.zip): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/984633483.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/984633483.zip)

Im Beispielprojekt sind folgende Stationsparameter für die Zentralstation eingestellt:

- Server host address (IP-Adresse der Unterstation): **127.0.0.1** (Sie müssen mindestens diesen Parameter an Ihre Zielplattform anpassen!).
- Server port address (Portadresse der Unterstation): **2404**.
- k: **12**.
- w: **8**.
- t0: **30s**.
- t1: **15s**.
- t2: **10s**.
- t3: **20s**.
- Cause of transfer size: **2 octets (inclusive originator address)**.
- Common ASDU address size: **2 octets**.
- Information object address size: **3 octets**.
- Originator address: **1**.
- Common ASDU address: **7**.
- Max. APDU length: **253**.

Datenpunktconfiguration (sollte gleich sein mit der Datenpunktconfiguration der Unterstation):

- Single point information: M\_SP\_NA\_1 (IOA := 100), M\_SP\_NA\_1(IOA := 101), M\_SP\_TB\_1(IOA := 102).
- Double point information: M\_DP\_NA\_1 (IOA := 200), M\_DP\_NA\_1 (IOA := 201), M\_DP\_TB\_1 (IOA := 202).

- Regulating step value: M\_ST\_NA\_1 (IOA := 300), M\_ST\_NA\_1 (IOA := 301), M\_ST\_TB\_1 (IOA := 302).
- Bitstring value: M\_BO\_NA\_1 (IOA := 400), M\_BO\_NA\_1 (IOA := 401), M\_BO\_TB\_1 (IOA := 402).
- Measured value, normalized value: M\_ME\_NA\_1 (IOA := 500), M\_ME\_NA\_1 (IOA := 501), M\_ME\_TD\_1 (IOA := 502).
- Measured value, scaled value: M\_ME\_NB\_1 (IOA := 600), M\_ME\_NB\_1 (IOA := 601), M\_ME\_TD\_1 (IOA := 602).
- Measured value, short floating point value: M\_ME\_NC\_1 (IOA := 700), M\_ME\_NC\_1 (IOA := 701), M\_ME\_TF\_1 (IOA := 702).
- Binary counter value: M\_IT\_NA\_1 (IOA := 800), M\_IT\_NA\_1 (IOA := 801), M\_IT\_TB\_1 (IOA := 802).
- Single command: C\_SC\_NA\_1 (IOA := 10), C\_SC\_NA\_1 (IOA := 11), C\_SC\_TA\_1 (IOA := 12).
- Double command: C\_DC\_NA\_1 (IOA := 20), C\_DC\_NA\_1 (IOA := 21), C\_DC\_TA\_1 (IOA := 22).
- Bitstring command: C\_BO\_NA\_1 (IOA := 40), C\_BO\_NA\_1 (IOA := 41), C\_BO\_TA\_1 (IOA := 42).
- Setpoint command, normalized value: C\_SE\_NA\_1 (IOA := 50), C\_SE\_NA\_1 (IOA := 51), C\_SE\_TA\_1 (IOA := 52).
- Setpoint command, scaled value: C\_SE\_NB\_1 (IOA := 60), C\_SE\_NB\_1 (IOA := 61), C\_SE\_TB\_1 (IOA := 62).
- Setpoint command, short floating point value: C\_SE\_NC\_1 (IOA := 70), C\_SE\_NC\_1 (IOA := 71), C\_SE\_TC\_1 (IOA := 72).

Initialisierungssequenz, die nach dem Verbindungsaufbau einmalig/automatisch ausgeführt wird:

- Kein Testbefehl (C\_TS\_NA\_1, deaktiviert).
- Uhrzeitsynchronisationsbefehl (C\_CS\_NA\_1).
- Generalabfragebefehl (C\_IC\_NA\_1) der Gruppe: Allgemein (eIEC870\_QOI\_INROGEN).
- Kein Zählerabfragebefehl (C\_IC\_NA\_1) der Gruppe: Allgemein (eIEC870\_RQT\_REQCOGEN, deaktiviert). Zählerabfrage Mode A (Zähler werden lokal in der Unterstation umgespeichert und als spontane Daten an die Zentralstation gesendet).

Beispielimplementierung folgender Dienste:

- Testbefehl (C\_TS\_NA\_1) alle 60 Sekunden.
- Uhrzeitsynchronisationsbefehl (C\_CS\_NA\_1) alle 60 Sekunden.
- Generalabfragebefehl (C\_IC\_NA\_1) der Gruppe: Allgemein (eIEC870\_QOI\_INROGEN) alle 60 Sekunden.
- Kein Zählerabfragebefehl (C\_IC\_NA\_1) der Gruppe: Allgemein (eIEC870\_RQT\_REQCOGEN, deaktiviert). Zählerabfrage Mode A (Zähler werden lokal in der Unterstation umgespeichert und als spontane Daten an die Zentralstation gesendet).
- Simulation und Übertragung der Befehle alle 10 Sekunden wenn die Variable **bSendCmd** auf TRUE gesetzt wurde (siehe obere Datenpunktkonfiguration).
- Empfang spontaner Daten.
- Empfang von M\_EI\_NA\_1 (Ende der Initialisierung).

Folgende Tabelle beschreibt kurz die Hauptkomponenten des Projekts:



Komponente	Typ	Beschreibung
MAIN	PRG	Dieses Programmteil implementiert das Fernwirkprotokoll (langsame Task).
fbClient	FB_IEC870_5_104Master (FB)	Implementiert die Zentralstation. Eine Instanz dieses Bausteins kann nur mit einer Unterstation kommunizieren. Für jede weitere Unterstation mit der kommuniziert werden soll wird eine weitere Instanz dieses Funktionsbausteins benötigt.
AODB	ARRAY OF ST_IEC870_5_101AODBEntry (STRUCT)	Applikationsobjektdatenbank. Es ist eine Array-Variable. Für jeden Datenpunkt wird ein Arrayelement benötigt.
hTable	T_HAODBTable (STRUCT)	Applikationsobjektdatenbank-Handle. Mit Hilfe dieser Variablen wird auf die Applikationsobjektdatenbank zugegriffen und nie direkt auf die einzelnen Arrayelemente.
F_iecCreateTableHnd	FUNCTION	Im Initialisierungsschritt werden alle Arrayelemente von AODB mit diesem Funktionsaufruf einmalig initialisiert. Beim Erfolg liefert dieser Funktionsaufruf das Handle.
F_iecAddTableEntry	FUNCTION	Im Initialisierungsschritt werden die einzelnen Datenpunkte der Station mit diesem Funktionsaufruf einmalig konfiguriert.
memory	ST_MemoryData (STRUCT)	Diese Struktur beinhaltet die empfangenen und die zu sendenden Prozessdaten die ständig aktualisiert werden.
fbHandler	FB_IEC870_5_101TableEventHandler (FB)	Dieser Funktionsbaustein teilt der Applikation mit wenn Datenpunktdaten geändert oder aktualisiert wurden.

### 6.5.1.1.5 Szenarios

#### 6.5.1.1.5.1 Spontane Datenübertragung und Generalabfrage

Das hier vorgestellte Beispielprogramm zeigt einen IEC 60870-5-104 Master mit High Level Schnittstelle und einer einfachen Kommunikation von zwei Datenpunkten. Ein Datenpunkt wird in Control und ein anderer in Monitor Direction jeweils spontan übertragen. Für den Datenpunkt in Monitor Direction ist zusätzlich eine Generalabfrage konfiguriert.

Download TwinCAT XAE Project (\*.zip): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/1243747211.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/1243747211.zip)

Im Beispiel sind folgende Parameter für die Zentralstation (Master) eingestellt:

- Server Host-Adresse (IP-Adresse des Slaves): 127.0.0.1 (Mindestens dieser Parameter muss individuell angepasst werden)
- Server Port-Adresse (Port-Adresse des Slaves): 2404
- k: 12
- w: 8
- t0: 30s
- t1: 15s
- t2: 10s
- t3: 20s
- Allgemeine ASDU-Adresse: 3
- Cause of transfer size: 2 octets
- Common ASDU address size: 2 octets
- Information object address size: 3 octets

- Max. APDU length: 253



- Die Applikationen stellen eine beispielhafte Umsetzung der High-Level Schnittstelle dar. Es wird kein Anspruch auf Vollständigkeit und Richtigkeit übernommen. Stellen Sie die IP-Adresse des Slaves sowohl im Master- als auch im Slave-Teil ein! Überprüfen Sie, dass der Port 2404 nicht von einer Firewall blockiert wird!
- Die Verbindungsherstellung beginnt nach dem Starten des Programms. Die Datenpunkte werden in der Globalen Variablenliste (GVL) verwaltet.

### 6.5.1.1.5.2 Generalabfrage und Gruppen

Das hier vorgestellte Beispielprogramm zeigt einen IEC 60870-5-104 Master mit High Level Schnittstelle. Das Beispiel geht darauf ein Gruppen von Datentypen zu definieren. Es wird eine Generalabfrage und eine gesonderte Gruppe 1 konfiguriert. Die Generalabfrage erfolgt alle 30s und die Gruppe 1 wird alle 10s abgefragt. Es werden unter anderem Single Points, Bitstrings und Measured Values kommuniziert.

Download TwinCAT XAE Project (\*.zip): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/1243751819.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/1243751819.zip)

Im Beispiel sind folgende Parameter für die Zentralstation (Master) eingestellt:

- Server Host-Adresse (IP-Adresse des Slaves): 127.0.0.1 (Mindestens dieser Parameter muss individuell angepasst werden)
- Server Port-Adresse (Port-Adresse des Slaves): 2404
- k: 12
- w: 8
- t0: 30s
- t1: 15s
- t2: 10s
- t3: 20s
- Allgemeine ASDU-Adresse: 3
- Cause of transfer size: 2 octets
- Common ASDU address size: 2 octets
- Information object address size: 3 octets
- Max. APDU length: 253



- Die Applikationen stellen eine beispielhafte Umsetzung der High-Level Schnittstelle dar. Es wird kein Anspruch auf Vollständigkeit und Richtigkeit übernommen. Stellen Sie die IP-Adresse des Slaves sowohl im Master- als auch im Slave-Teil ein! Überprüfen Sie, dass der Port 2404 nicht von einer Firewall blockiert wird!
- Die Verbindungsherstellung beginnt nach dem Starten des Programms. Die Datenpunkte werden in der Globalen Variablenliste (GVL) verwaltet.

### 6.5.1.1.5.3 Zählwertabfrage und Zeitsynchronisation

Das hier vorgestellte Beispielprogramm zeigt einen IEC 60870-5-104 Master mit High Level Schnittstelle. Kern des Beispiels ist die Uhrzeit-Synchronisation von Master zu Slave und die Zählwertabfrage. Die Synchronisation der Uhrzeit findet hier beispielhaft alle 120s statt. Ein Datentyp M\_IT\_NA\_1 sorgt für die Zählwert-Übertragung durch eine General-Zähler-Abfrage.

Download TwinCAT XAE Project (\*.zip): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/1243756427.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/1243756427.zip)

Im Beispiel sind folgende Parameter für die Zentralstation (Master) eingestellt:

- Server Host-Adresse (IP-Adresse des Slaves): 127.0.0.1 (Mindestens dieser Parameter muss individuell angepasst werden)
- Server Port-Adresse (Port-Adresse des Slaves): 2404

- k: 12
- w: 8
- t0: 30s
- t1: 15s
- t2: 10s
- t3: 20s
- Allgemeine ASDU-Adresse: 3
- Cause of transfer size: 2 octets
- Common ASDU address size: 2 octets
- Information object address size: 3 octets
- Max. APDU length: 253



- Die Applikationen stellen eine beispielhafte Umsetzung der High-Level Schnittstelle dar. Es wird kein Anspruch auf Vollständigkeit und Richtigkeit übernommen. Stellen Sie die IP-Adresse des Slaves sowohl im Master- als auch im Slave-Teil ein! Überprüfen Sie, dass der Port 2404 nicht von einer Firewall blockiert wird!
- Die Verbindungsherstellung beginnt nach dem Starten des Programms. Die Datenpunkte werden in der Globalen Variablenliste (GVL) verwaltet.

## 6.5.1.2 Low-Level Interface

### 6.5.1.2.1 Quick start

Einfaches Projekt mit kompletten Quellen finden Sie hier: [IEC 60870-5-104 Zentralstation-Low-Level-Beispiel \[▶ 651\]](#).

Eine SPS-Applikation, die über das "Low-Level"-Interface mit einer Unterstation kommunizieren soll, benötigt folgende Ressourcen:

- Eine Instanz des Kommunikationsbausteins: [FB IEC870\\_5\\_104TProtocol \[▶ 411\]](#);
- Eine Instanz des Funktionsbausteins zur Herstellung der TCP/IP-Verbindung: [FB\\_ServerClientConnection](#) oder [FB\\_ClientServerConnection](#)
- Eine Instanz des TX/RX-Datenpuffers: [ST IEC870\\_5\\_101TBuffer \[▶ 481\]](#);
- Eine Instanz des Funktionsbausteins zur Manipulation des TX/RX-Datenpuffers: [FB IEC870\\_5\\_101TBufferCtrl \[▶ 403\]](#);

### 6.5.1.2.2 Zentralstation (Master)

Einfache Zentralstation-Applikation die das TwinCAT IEC 60870-5-104 Low-Level Interface verwendet. Diese Beispielapplikation kann nur mit einer Unterstation kommunizieren.

Download TwinCAT XAE Project (\*.zip): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resourcen/984636427.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resourcen/984636427.zip)

Im Beispielprojekt sind folgende Stationsparameter für die Zentralstation eingestellt:

- Server host address (IP-Adresse der Unterstation): **127.0.0.1 (Sie müssen mindestens diesen Parameter an Ihre Zielplattform anpassen!).**
- Server port address (Portadresse der Unterstation): **2404.**
- k: **12.**
- w: **8.**
- t0: **30s.**
- t1: **15s.**
- t2: **10s.**

- t3: **20s**.
- Cause of transfer size: **2 octets (inclusive originator address)**.
- Common ASDU address size: **2 octets**.
- Information object address size: **3 octets**.
- Originator address: **0**.
- Common ASDU address: **7**.
- Max. APDU length: **253**.

Datenpunktconfiguration (sie sollte gleich sein mit der Datenpunktconfiguration der Unterstation):

- Single point information: M\_SP\_NA\_1 (IOA := 100), M\_SP\_NA\_1 (IOA := 101), M\_SP\_TB\_1 (IOA := 102).
- Double point information: M\_DP\_NA\_1 (IOA := 200).
- Regulating step value: M\_ST\_NA\_1 (IOA := 300).
- Bitstring value: M\_BO\_NA\_1 (IOA := 400).
- Measured value, normalized value: M\_ME\_NA\_1 (IOA := 500).
- Measured value, scaled value: M\_ME\_NB\_1 (IOA := 600).
- Measured value, short floating point value: M\_ME\_NC\_1 (IOA := 700).
- Binary counter value: M\_IT\_NA\_1 (IOA := 800), M\_IT\_NA\_1 (IOA := 801), M\_IT\_TB\_1 (IOA := 802).
- Single command: C\_SC\_NA\_1 (IOA := 10).
- Double command: C\_DC\_NA\_1 (IOA := 20).
- Regulating step command: C\_RC\_NA\_1 (IOA := 30).
- Bitstring command: C\_BO\_NA\_1 (IOA := 40).
- Setpoint command, normalized value: C\_SE\_NA\_1 (IOA := 50).
- Setpoint command, scaled value: C\_SE\_NB\_1 (IOA := 60).
- Setpoint command, short floating point value: C\_SE\_NC\_1 (IOA := 70).

Beispielimplementierung folgender Dienste:

- Testbefehl (C\_TS\_TA\_1) alle 10 Sekunden.
- Uhrzeitsynchronisationsbefehl (C\_CS\_NA\_1) alle 20 Sekunden.
- Generalabfragebefehl (C\_IC\_NA\_1) der Gruppe: Allgemein (eIEC870\_QOI\_INROGEN) alle 40 Sekunden.
- Zählerabfragebefehl (C\_IC\_NA\_1) der Gruppe: Allgemein (eIEC870\_RQT\_REQCOGEN) alle 60 Sekunden. Zählerabfrage Mode A.
- Simulation und Übertragung der Befehle alle 60 Sekunden wenn die Variable **bSimulate** auf TRUE gesetzt wurde (siehe obere Datenpunktconfiguration).
- Empfang spontaner Daten.
- Empfang von M\_EI\_NA\_1 (Ende der Initialisierung).
- 

Folgende Tabelle beschreibt kurz die Hauptkomponenten des Projekts:

Komponente	Typ	Beschreibung
MAIN	PRG	Dieses Programmteil implementiert das Fernwirkprotokoll
fbClient	FB_IEC870_5_104M (FB)	Implementiert die Zentralstation. Eine Instanz dieses Bausteins kann nur mit einer Unterstation kommunizieren. Für jede weitere Unterstation mit der kommuniziert werden soll wird eine weitere Instanz dieses Funktionsbausteins benötigt.
fbServices	FB_IEC870_5_104MServices (FB)	Beinhaltet eine einfache Implementierung einiger Dienste wie z.B. Generalabfrage, Uhrzeitsynchronisation, Testbefehle usw.
stProcPara	ST_IEC870_5_104MProcData (STRUCT)	Hier befinden sich die empfangenen und die zu sendenden Prozessdaten die ständig aktualisiert werden.
sysPara	ST_IEC870_5_104MSysPara (STRUCT)	Konfiguration der Systemparameter der Zentralstation und der implementierten Dienste.
fbLog	FB_IEC870_DebugLogFifo (FB)	Ein einfacher Logpuffer für Debugzwecke/ Fehlersuche.

### 6.5.1.2.3 Szenarios

#### Grundsätzliche Erläuterungen zu den Low Level Szenarios:

Der Initialisierungsschritt beginnt mit der Konfiguration der TCP/IP-Schnittstelle, wobei IPAdresse und Kommunikationsport festgelegt werden. Anschließend werden verschiedene Protokollparameter, sowie Parameter der Sende- beziehungsweise Empfangspuffer eingestellt. Danach wird die TCP/IP-Verbindung mit Hilfe von Funktionsbausteinen aus der TCP/IP-Funktion hergestellt.

Daraufhin erfolgt das Empfangen der Daten, welche solange mit der Aktion **fbBuffer.RxRemoveObj** aus dem Empfangspuffer geladen werden, bis dieser leer ist. Die empfangenen Daten werden nach Typkennung und Objektadresse sortiert. Nach der Sortierung werden die Entsprechenden Daten mit der Funktion **F\_iecCopyStreamToBuffer** in eine Variable gespeichert und anschließend entsprechend der in der Norm festgelegten Bestandteile aufgeteilt. Wird im Slave der Befehl für eine Generalabfrage oder der Abfragebefehl für eine bestimmte Abfragegruppe empfangen, so wird sofort die entsprechende Übertragungsprozedur für Abfragebefehle angestoßen und die für die jeweilige Gruppe definierten Datenpunkte in den Sendepuffer geschrieben. Das Senden der Daten wird im nachfolgenden Absatz genauer beschrieben.

Anschließend erfolgt sowohl beim Master als auch beim Slave das Senden der Daten, die außerhalb oder zusätzlich zu einer Abfragegruppe übermittelt werden sollen. Dabei werden zuerst gemeinsame Sendeinformationen der Datenpunkte wie ASDU Adresse, Anzahl der zu übertragenden Objekte, Grund der Datenübertragung und Prioritätsklasse in die Struktur **stTxAsdu.ident** geschrieben. Anschließend werden die Protokollframes der einzelnen Datenpunkte, welche aus Typkennung, Objektadresse und objektspezifischen Daten bestehen in die Struktur **stTxAsdu** geschrieben. Anschließend wird diese mit der Funktion **F\_iecCopyBufferToStream** umkopiert und der Aktion **fbBuffer.TxAddObj** in den Sendepuffer geschrieben. Abschließend werden mögliche Fehler aus dem Fehlerspeicher ausgelesen und in einer Fehlerausgabe gelistet.

Der Ablauf des Lesens, Schreibens und das anschließende Fehlerhandling wiederholen sich zyklisch.

#### 6.5.1.2.3.1 Spontane Datenübertragung und Generalabfrage

Das hier vorgestellte Beispielprogramm zeigt einen IEC 60870-5-104 Master mit Low Level Schnittstelle und einer einfachen Kommunikation von zwei Datenpunkten. Ein Datenpunkt wird in Control und ein anderer in Monitor Direction jeweils spontan übertragen. Für den Datenpunkt in Monitor Direction ist zusätzlich eine Generalabfrage konfiguriert.

Download TwinCAT XAE Project (\*.zip): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/1244576139.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/1244576139.zip)

Im Beispiel sind folgende Parameter für die Zentralstation (Master) eingestellt:

- Server Host-Adresse (IP-Adresse des Slaves): 127.0.0.1 (Mindestens dieser Parameter muss individuell angepasst werden)
- Server Port-Adresse (Port-Adresse des Slaves): 2404
- k: 12
- w: 8
- t0: 30s
- t1: 15s
- t2: 10s
- t3: 20s
- Allgemeine ASDU-Adresse: 3
- Cause of transfer size: 2 octets
- Common ASDU address size: 2 octets
- Information object address size: 3 octets
- Max. APDU length: 253



- Es wird kein Anspruch auf Vollständigkeit und Richtigkeit übernommen. Stellen Sie die IP-Adresse des Slaves sowohl im Master- als auch im Slave-Teil ein! Überprüfen Sie, dass der Port 2404 nicht von einer Firewall blockiert wird!
  - Die Verbindungsherstellung beginnt nach dem Starten des Programms. Die Datenpunkte werden in der Globalen Variablenliste (GVL) verwaltet.
- 

### 6.5.1.2.3.2 Generalabfrage und Gruppen

Das hier vorgestellte Beispielprogramm zeigt einen IEC 60870-5-104 Master mit Low Level Schnittstelle. Das Beispiel geht darauf ein Gruppen von Datentypen zu definieren. Es wird eine Generalabfrage und eine gesonderte Gruppe 1 konfiguriert. Die Generalabfrage erfolgt alle 30s und die Gruppe 1 wird alle 10s abgefragt. Es werden unter anderem Single Points, Bitstrings und Measured Values kommuniziert.

Download TwinCAT XAE Project (\*.zip): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resourcen/1244593547.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resourcen/1244593547.zip)

Im Beispiel sind folgende Parameter für die Zentralstation (Master) eingestellt:

- Server Host-Adresse (IP-Adresse des Slaves): 127.0.0.1 (Mindestens dieser Parameter muss individuell angepasst werden)
- Server Port-Adresse (Port-Adresse des Slaves): 2404
- k: 12
- w: 8
- t0: 30s
- t1: 15s
- t2: 10s
- t3: 20s
- Allgemeine ASDU-Adresse: 3
- Cause of transfer size: 2 octets
- Common ASDU address size: 2 octets
- Information object address size: 3 octets
- Max. APDU length: 253



- Es wird kein Anspruch auf Vollständigkeit und Richtigkeit übernommen. Stellen Sie die IP-Adresse des Slaves sowohl im Master- als auch im Slave-Teil ein! Überprüfen Sie, dass der Port 2404 nicht von einer Firewall blockiert wird!
  - Die Verbindungsherstellung beginnt nach dem Starten des Programms. Die Datenpunkte werden in der Globalen Variablenliste (GVL) verwaltet.
-

### 6.5.1.2.3.3 Zählwertabfrage und Zeitsynchronisation

Das hier vorgestellte Beispielprogramm zeigt einen IEC 60870-5-104 Master mit Low Level Schnittstelle. Kern des Beispiels ist die Uhrzeit-Synchronisation von Master zu Slave und die Zählwertabfrage. Die Synchronisation der Uhrzeit findet hier beispielhaft alle 120s statt. Ein Datentyp M\_IT\_NA\_1 sorgt für die Zählwert-Übertragung durch eine General-Zähler-Abfrage.

Download TwinCAT XAE Project (\*.zip): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/1244598155.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/1244598155.zip)

Im Beispiel sind folgende Parameter für die Zentralstation (Master) eingestellt:

- Server Host-Adresse (IP-Adresse des Slaves): 127.0.0.1 (Mindestens dieser Parameter muss individuell angepasst werden)
- Server Port-Adresse (Port-Adresse des Slaves): 2404
- k: 12
- w: 8
- t0: 30s
- t1: 15s
- t2: 10s
- t3: 20s
- Allgemeine ASDU-Adresse: 3
- Cause of transfer size: 2 octets
- Common ASDU address size: 2 octets
- Information object address size: 3 octets
- Max. APDU length: 253



- Es wird kein Anspruch auf Vollständigkeit und Richtigkeit übernommen. Stellen Sie die IP-Adresse des Slaves sowohl im Master- als auch im Slave-Teil ein! Überprüfen Sie, dass der Port 2404 nicht von einer Firewall blockiert wird!
- Die Verbindungsherstellung beginnt nach dem Starten des Programms. Die Datenpunkte werden in der Globalen Variablenliste (GVL) verwaltet.

## 6.5.2 Unterstation

### 6.5.2.1 High-Level Interface

#### 6.5.2.1.1 Quick start

Einfaches Projekt mit kompletten Quellen finden Sie hier: [IEC 60870-5-104 Unterstation-High-Level-Beispiel \[▶ 656\]](#).

Kompatibilitätsliste finden Sie hier: [Interoperability check list \[▶ 86\]](#)

Übersicht der Fehlercodes finden Sie hier: [Fehlercodes \[▶ 708\]](#)

#### Kurzanleitung

1. Erstellen Sie ein neues SPS-Projekt erstellen binden Sie die SPS-Bibliothek **Tc2\_IEC60870\_5\_10x** ein.
2. Legen Sie im Hauptprogramm eine Instanz der T\_HSERVER-Variablen an (Verbindungs-Handle) und initialisieren Sie im Programmteil einmalig durch den Aufruf der F\_CreateServerHnd-Funktion mit passenden Parametern.

3. Konfiguration der Datenpunkte: Legen Sie eine Array-Variable vom Typ `ST_IEC870_5_101AODBEntry` [► 482] an. Jedes Arrayelement entspricht einem Datenpunkt. Konfigurieren Sie die Datenpunkte mit Hilfe der Funktion `F_ieclnitAOEntry` [► 435] zur Laufzeit (z.B. in einem Init-Schritt).
4. Legen Sie im Hauptprogramm eine Instanz des Protokoll-Bausteins `FB_IEC870_5_104Slave` [► 392] an, konfigurieren Sie diese und rufen Sie diese auf. Konfigurieren Sie die System- und Protokoll-Parameter passend zu den Parametern der Leitstation.

### 6.5.2.1.2 Unterstation Minimal (Slave)

Einfache Unterstation-Applikation die das TwinCAT IEC 60870-5-104 High-Level Interface nutzt.

Download TwinCAT XAE Project (\*.zip): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/984639371.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/984639371.zip)

Im Beispielprojekt sind folgende Stationsparameter für die Unterstation eingestellt:

- Server host address (IP-Adresse der Unterstation): **127.0.0.1**(**Sie müssen mindestens diesen Parameter an Ihre Zielplattform anpassen!**) .
- Server port address (Portadresse der Unterstation): **2404**.
- k: **12**.
- w: **8**.
- t0: **30s**.
- t1: **15s**.
- t2: **10s**.
- t3: **20s**.
- Cause of transfer size: **2 octets (inclusive originator address)**.
- Common ASDU address size: **2 octets**.
- Information object address size: **3 octets**.
- Originator address: **1**.
- Common ASDU address: **7**.
- Max. APDU length: **253**.

Datenpunktconfiguration (sie sollte gleich sein mit der Datenpunktconfiguration der Zentralstation):

- Single point information: `M_SP_NA_1` (IOA := 100).
- Single command: `C_SC_NA_1` (IOA := 10).

Beispielimplementierung folgender Dienste:

- End of Initialization (`M_EI_NA_1`) wird einmalig gesendet.
- Testbefehl (`C_TS_TA_1`) .
- Uhrzeitsynchronisationsbefehl (`C_CS_NA_1`).
- Generalabfragebefehl (`C_IC_NA_1`) der Gruppe: Allgemein (`eIEC870_QOI_INROGEN`).
- Zählerabfragebefehl (`C_CI_NA_1`) der Gruppe: Allgemein (`eIEC870_RQT_REQCOGEN`). Zählerabfrage Mode C.
- Einzelbefehl-Bearbeitung (`C_SC_NA_1`) (direkte Kommandoausführung)
- Das Senden spontaner Daten vom Typ (`M_SP_NA_1`, IOA := 100) bei Wertänderung an der **bSPI**-Variablen.

Folgende Tabelle beschreibt kurz die Hauptkomponenten des Projekts:



Komponente	Typ	Beschreibung
MAIN	PRG	Dieses Programmteil implementiert das Fernwirkprotokoll (langsame Task).
fbServer	FB_IEC870_5_104Slave (FB)	Implementiert die Unterstation. Eine Instanz dieses Bausteins kann nur mit einer Zentralstation kommunizieren. Für jede weitere Zentralstation mit der kommuniziert werden soll wird eine weitere Instanz dieses Funktionsbausteins benötigt.
bSPI	BOOL	Der zu sendende ON/OFF-Wert wird direkt von der an der Merkeradresse <b>AT%MX1.5</b> dieser lokierten SPS-Variablen gemappt.
bSCS	BOOL	Der empfangene ON/OFF-Wert wird in diese SPS-Variable direkt an die lokierte Merkeradresse: <b>AT%MX0.0</b> gemappt.
AODB	ARRAY OF ST_IEC870_5_101AO DBEntry (STRUCT)	Applikationsobjektdatenbank. Es ist eine Array-Variable. Für jeden Datenpunkt wird ein Arrayelement benötigt.
F_iecInitAOEntry	FUNCTION	Im Initialisierungsschritt werden die einzelnen Datenpunkte der Station mit diesem Funktionsaufruf einmalig konfiguriert.

### 6.5.2.1.3 Tutorial

#### 6.5.2.1.3.1 Programmierumgebung konfigurieren

Gehen Sie bei der grundlegenden Konfiguration der Programmierumgebung wie folgt vor.

✓ Öffnen Sie das Microsoft Visual Studio

1. Gehen Sie auf: File / New / Project
2. Wählen Sie ein TwinCAT Projekt aus
3. Wenn die Solution mit dem TwinCAT Projekt geöffnet ist, können Sie mit einem Rechtsklick auf den Eintrag PLC das Kontextmenu öffnen und Add New Item wählen
4. Wählen Sie ein Standard PLC Project aus
5. Den PLC Project Baum bitte aufklappen und über das Kontextmenu des Unterpunktes References Add library auswählen
6. Fügen Sie die folgenden Bibliotheken als Grundlage Ihrer IEC 60870-5-10x Applikation hinzu:  
Tc2\_Utillities  
Tc2\_Tcplp  
Tc2\_IEC60870\_5\_10x

⇒ Damit haben Sie die grundlegende Konfiguration der Programmierumgebung abgeschlossen.

#### 6.5.2.1.3.2 Applikationsobjekte definieren und konfigurieren

Applikationsobjekte = Single Points, Double Points, Measured Values, Short Floating Point Values usw.

In diesem Beispiel wurden die Befehle so konfiguriert, dass die Prozessdaten der Befehle im gleichen Speicherbereich aber auf einem anderen Byte-/Bit-Offset wie die Daten der Information in Überwachungsrichtung liegen. Sie können aber auch die Befehle auf den gleichen Byte- Bit-Offset wie die Information in Überwachungsrichtung legen.

**Beispiel:**

C\_SC\_NA\_1 mit IOA = 10 auf den gleichen Byte- und Bit-Offset wie M\_SP\_NA\_1 mit IOA = 100 (beide Byte-Offset = 100 und Bit-Offset = 0). In diesem Fall wird eine Wertänderung durch ein Kommando von der Leitstation eine Übertragung des M\_SP\_NA\_1 mit der Objektadresse 100 und Übertragungsursache <11> (returned by remote command) zur Folge haben.

Als Beispiel für das Tutorial werden folgende Applikationsobjekte:

Arrayelement	ASDU identifizier	Objektadresse IOA	Group-Konfigurationsparameter	Basiszeitmultiplikator	SPS-Prozessdatenbereich	Offset Byte	Offset Bit	Prozessdatenbreite in der TwinCAT SPS
0	M_SP_N A_1	100	Generalabfrage	0	Merker	100	0	1 Bit
1	M_SP_N A_1	101	Generalabfrage	0	Merker	100	1	1 Bit
2	M_SP_T B_1	102	Generalabfrage	0	Merker	100	2	1 Bit
3	M_DP_N A_1	200	Generalabfrage	0	Merker	200	0	2 Bits
4	M_DP_N A_1	201	Generalabfrage	0	Merker	200	2	2 Bits
5	M_DP_T B_1	202	Generalabfrage	0	Merker	200	4	2 Bits
6	M_ST_N A_1	300	Generalabfrage	0	Merker	300	0	1 Byte
7	M_ST_N A_1	301	Generalabfrage	0	Merker	301	0	1 Byte
8	M_ST_T B_1	302	Generalabfrage	0	Merker	302	0	1 Byte
9	M_BO_N A_1	400	Generalabfrage	0	Merker	400	0	4 Byte
10	M_BO_N A_1	401	Generalabfrage	0	Merker	404	0	4 Byte
11	M_BO_T B_1	402	Generalabfrage	0	Merker	408	0	4 Byte
12	M_ME_N A_1	500	Generalabfrage	0	Merker	500	0	2 Byte
13	M_ME_N A_1	501	Generalabfrage	0	Merker	502	0	2 Byte
14	M_ME_T D_1	502	Generalabfrage	0	Merker	504	0	2 Byte
15	M_ME_N B_1	600	Generalabfrage	0	Merker	600	0	2 Byte
16	M_ME_N B_1	601	Generalabfrage	0	Merker	602	0	2 Byte
17	M_ME_T E_1	602	Generalabfrage	0	Merker	604	0	2 Byte
18	M_ME_N C_1	700	Generalabfrage	0	Merker	700	0	4 Byte
19	M_ME_N C_1	701	Generalabfrage	0	Merker	704	0	4 Byte
20	M_ME_T F_1	702	Generalabfrage	0	Merker	708	0	4 Byte
21	M_IT_N A_1	800	Generalzählerabfrage und Mode A (lokal Umspeichern mit Spontanübertragung alle 15s)	0	Merker	800	0	4 Byte

22	M_IT_N A_1	801	Generalzählerabfrage und Mode A (lokal Umspeichern mit Spontanübertragung alle 15s)	0	Merker	804	0	4 Byte
23	M_IT_TB _1	802	Generalzählerabfrage und Mode A (lokal Umspeichern mit Spontanübertragung alle 15s)	0	Merker	808	0	4 Byte
<b>Commands</b>								
24	C_SC_N A_1	10	-	0	Merker	2100	0	1 Bit
25	C_SC_N A_1	11	-	0	Merker	2100	1	1 Bit
26	C_SC_T A_1	12	-	0	Merker	2100	2	1 Bit
27	C_DC_N A_1	20	-	0	Merker	2200	0	2 Bit
28	C_DC_N A_1	21	-	0	Merker	2200	2	2 Bit
29	C_DC_T A_1	22	-	0	Merker	2200	4	2 Bit
30	C_RC_N A_1	30	-	0	Merker	2300	0	1 Byte
31	C_RC_N A_1	31	-	0	Merker	2301	0	1 Byte
32	C_RC_T A_1	32	-	0	Merker	2302	0	1 Byte
33	C_BO_N A_1	40	-	0	Merker	2400	0	4 Byte
34	C_BO_N A_1	41	-	0	Merker	2404	0	4 Byte
35	C_BO_T A_1	42	-	0	Merker	2408	0	4 Byte
36	C_SE_N A_1	50	-	0	Merker	2500	0	2 Byte
37	C_SE_N A_1	51	-	0	Merker	2502	0	2 Byte
38	C_SE_T A_1	52	-	0	Merker	2504	0	2 Byte
39	C_SE_N B_1	60	-	0	Merker	2600	0	2 Byte
40	C_SE_N B_1	61	-	0	Merker	2602	0	2 Byte
41	C_SE_T B_1	62	-	0	Merker	2604	0	2 Byte
42	C_SE_N C_1	70	-	0	Merker	2700	0	4 Byte
43	C_SE_N C_1	71	-	0	Merker	2704	0	4 Byte
44	C_SE_T C_1	72	-	0	Merker	2708	0	4 Byte

## Datenbankvariable deklarieren

Die Applikationsobjekt-Datenbank ist eine Array-Variable vom Typ [ST\\_IEC870\\_5\\_101AODBEntry \[▶ 482\]](#). Jedes Array-Element entspricht einem Applikationsobjekt. Die maximale Anzahl der Applikationsobjekte ist frei wählbar und nur durch den verfügbaren Speicher begrenzt. Sie müssen sich auf eine konstante maximale Anzahl während der SPS-Programmierung festlegen. Zur Laufzeit kann die maximale Anzahl der Applikationsobjekte nicht mehr verändert werden.

In unserem Beispiel werden 50 Applikationsobjekte deklariert. Diese Anzahl reicht für die meisten Anwendungen aus. Beachten Sie, dass sehr viele Applikationsobjekte auch entsprechend viel Speicher und Laufzeit benötigen.

Definieren Sie folgende Variable in MAIN:

```
PROGRAM MAIN
VAR
AODB : ARRAY[0..49] OF ST\_IEC870\_5\_101AODBEntry \[▶ 482\];
END_VAR
```

## Applikationsobjekte konfigurieren

Während der Konfiguration der einzelnen Applikationsobjekte werden unter anderem der Objekt-Typ ( M\_SP\_NA\_1, M\_DP\_NA\_1, M\_ST\_NA\_1 usw.), die Objekt-Adresse und weitere Objekt-Parameter festgelegt.

Die Konfiguration der gewünschten Applikationsobjekte wird zur Programmlaufzeit durchgeführt. Jedes Applikationsobjekt (Datenbank-Array-Element) wird durch einen einmaligen Aufruf der [F\\_ieclnitAOEntry \[▶ 435\]](#)-Funktion konfiguriert. Das zu konfigurierende Array-Element wird an die Funktion per VAR\_IN\_OUT übergeben. Im Regelfall wird die Konfiguration beim SPS-Programmstart einmalig in einer Init-Routine durchgeführt. Die Funktion [F\\_ieclnitAOEntry \[▶ 435\]](#) erwartet folgende Funktionsparameter (von links nach rechts):

```
FUNCTION F_ieclnitAOEntry : UDINT
VAR_INPUT
eType : E\_IEC870\_5\_101TcTypeID \[▶ 500\];
objAddr : DWORD := 0;
group : DWORD := 0;
multiplier : BYTE := 0;
ioMapType : E\_IEC870\_5\_101IOMappingType \[▶ 506\];
byteOffs : UDINT := 0;
bitOffs : UDINT := 0;
END_VAR
VAR_IN_OUT
dbEntry : ST\_IEC870\_5\_101AODBEntry;
END_VAR
```

**eType:** Applikationsobjekt-Typ (ASDU identifier, z.B.: M\_SP\_NA\_1 für Single Point oder M\_DP\_NA\_1 für Double Point ). Beachten Sie, dass nur die in der Kompatibilitätsliste aufgeführten ASDU-Typen verwendet werden können. Unzulässige Typen werden ignoriert.

**objAddr :** Objektadresse, z.B. 100. Jedes Applikationsobjekt sollte mit einer eindeutigen Adresse konfiguriert werden.

**group:** Group-Konfigurationsparameter. Die verfügbaren Group-Parameter sind als Konstanten definiert und können mit ODER-Operator kombiniert werden. Z.B.: IEC870\_GRP\_INROGEN OR IEC870\_GRP\_PERCYC.

Hier finden Sie die [Beschreibung aller Group-Konfigurationsparameter](#).

**multiplier:** Basiszeitmultiplikator für die zyklische/periodische Datenübertragung ( 0=Deaktiviert). Die Basiszeit wird über die Systemparameter konfiguriert. Wurde die Basiszeit z.B. auf T#10s gesetzt und der Multiplikator auf den Wert 2, dann werden die periodischen/zyklischen Daten des Applikationsobjekts alle 20 Sekunden gesendet.

**ioMapType:** Dieser Parameter legt fest, aus oder in welchen Prozessdatenbereich der TwinCAT SPS die IEC-Prozessdaten zur Laufzeit gemappt werden sollen (inputs, outputs, memory, data).

**byteOffs:** Prozessdatenbereich Byteoffset;

**bitOffs:** Prozessdatenbereich Bitoffset;

**dbEntry:** Applikationsobjekt das konfiguriert werden soll (ein Datenbankvariable-Arrayelement, das an die Funktion per VAR\_IN\_OUT übergeben wird).

Um die Applikationsobjekte beim Programmstart zu konfigurieren wird in MAIN folgender SPS-Code hinzugefügt:

```
PROGRAM MAIN
VAR
AODB : ARRAY[0..49] OF ST_IEC870_5_101AODBEntry;

init : BOOL := TRUE;
initError : UDINT;
END_VAR

IF init THEN
init := FALSE;

(* Monitored Single Points *)
initError := F_iecInitAOEntry( M_SP_NA_1, 100, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 100, 0,
AODB[0] );
initError := F_iecInitAOEntry( M_SP_NA_1, 101, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 100, 1,
AODB[1] );
initError := F_iecInitAOEntry( M_SP_TB_1, 102, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 100, 2,
AODB[2] );
(* Double Points*)
initError := F_iecInitAOEntry( M_DP_NA_1, 200, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 200, 0,
AODB[3] );
initError := F_iecInitAOEntry( M_DP_NA_1, 201, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 200, 2,
AODB[4] );
initError := F_iecInitAOEntry( M_DP_TB_1, 202, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 200, 4,
AODB[5] );
(* Regulating step value *)
initError := F_iecInitAOEntry( M_ST_NA_1, 300, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 300, 0,
AODB[6] );
initError := F_iecInitAOEntry( M_ST_NA_1, 301, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 301, 0,
AODB[7] );
initError := F_iecInitAOEntry( M_ST_TB_1, 302, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 302, 0,
AODB[8] );
(* 32 bit string*)
initError := F_iecInitAOEntry( M_BO_NA_1, 400, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 400, 0,
AODB[9] );
initError := F_iecInitAOEntry( M_BO_NA_1, 401, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 404, 0,
AODB[10] );
initError := F_iecInitAOEntry( M_BO_TB_1, 402, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 408, 0,
AODB[11] );
(* Measured value, normalized value *)
initError := F_iecInitAOEntry( M_ME_NA_1, 500, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 500, 0,
AODB[12] );
initError := F_iecInitAOEntry( M_ME_NA_1, 501, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 502, 0,
AODB[13] );
initError := F_iecInitAOEntry( M_ME_TD_1, 502, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 504, 0,
AODB[14] );
(* Mesured value, scaled value *)
initError := F_iecInitAOEntry( M_ME_NB_1, 600, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 600, 0,
AODB[15] );
initError := F_iecInitAOEntry( M_ME_NB_1, 601, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 602, 0,
AODB[16] );
initError := F_iecInitAOEntry( M_ME_TE_1, 602, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 604, 0,
AODB[17] );
(* Measured value , short floating point value *)
initError := F_iecInitAOEntry( M_ME_NC_1, 700, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 700, 0,
AODB[18] );
initError := F_iecInitAOEntry( M_ME_NC_1, 701, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 704, 0,
AODB[19] );
initError := F_iecInitAOEntry( M_ME_TF_1, 702, IEC870_GRP_INROGEN, 0, MAP_AREA_MEMORY, 708, 0,
AODB[20] );
(* Integrated totals *)
initError := F_iecInitAOEntry( M_IT_NA_1, 800, IEC870_GRP_REQCOGEN OR IEC870_GRP_LOCFREEZE, 0,
MAP_AREA_MEMORY, 800, 0, AODB[21] );
initError := F_iecInitAOEntry( M_IT_NA_1, 801, IEC870_GRP_REQCOGEN OR IEC870_GRP_LOCFREEZE, 0,
MAP_AREA_MEMORY, 804, 0, AODB[22] );
initError := F_iecInitAOEntry( M_IT_TB_1, 802, IEC870_GRP_REQCOGEN OR IEC870_GRP_LOCFREEZE, 0,
MAP_AREA_MEMORY, 808, 0, AODB[23] );

(* Single commands *)
initError := F_iecInitAOEntry( C_SC_NA_1, 10, 0, 0, MAP_AREA_MEMORY, 2100, 0, AODB[24] );
initError := F_iecInitAOEntry( C_SC_NA_1, 11, 0, 0, MAP_AREA_MEMORY, 2100, 1, AODB[25] );
initError := F_iecInitAOEntry( C_SC_TA_1, 12, 0, 0, MAP_AREA_MEMORY, 2100, 2, AODB[26] );
(* Double commands *)
```

```

initError := F_iecInitAOEntry( C_DC_NA_1, 20, 0, 0, MAP_AREA_MEMORY, 2200, 0, AODB[27] );
initError := F_iecInitAOEntry( C_DC_NA_1, 21, 0, 0, MAP_AREA_MEMORY, 2200, 2, AODB[28] );
initError := F_iecInitAOEntry( C_DC_TA_1, 22, 0, 0, MAP_AREA_MEMORY, 2200, 4, AODB[29] );
(* Regulating step commands *)
initError := F_iecInitAOEntry( C_RC_NA_1, 30, 0, 0, MAP_AREA_MEMORY, 2300, 0, AODB[30] );
initError := F_iecInitAOEntry( C_RC_NA_1, 31, 0, 0, MAP_AREA_MEMORY, 2301, 0, AODB[31] );
initError := F_iecInitAOEntry( C_RC_TA_1, 32, 0, 0, MAP_AREA_MEMORY, 2302, 0, AODB[32] );
(* 32 bit string commands *)
initError := F_iecInitAOEntry( C_BO_NA_1, 40, 0, 0, MAP_AREA_MEMORY, 2400, 0, AODB[33] );
initError := F_iecInitAOEntry( C_BO_NA_1, 41, 0, 0, MAP_AREA_MEMORY, 2404, 0, AODB[34] );
initError := F_iecInitAOEntry( C_BO_TA_1, 42, 0, 0, MAP_AREA_MEMORY, 2408, 0, AODB[35] );
(* Set point, normalized values*)
initError := F_iecInitAOEntry( C_SE_NA_1, 50, 0, 0, MAP_AREA_MEMORY, 2500, 0, AODB[36] );
initError := F_iecInitAOEntry( C_SE_NA_1, 51, 0, 0, MAP_AREA_MEMORY, 2502, 0, AODB[37] );
initError := F_iecInitAOEntry( C_SE_TA_1, 52, 0, 0, MAP_AREA_MEMORY, 2504, 0, AODB[38] );
(* Set point, scaled values *)
initError := F_iecInitAOEntry( C_SE_NB_1, 60, 0, 0, MAP_AREA_MEMORY, 2600, 0, AODB[39] );
initError := F_iecInitAOEntry( C_SE_NB_1, 61, 0, 0, MAP_AREA_MEMORY, 2602, 0, AODB[40] );
initError := F_iecInitAOEntry( C_SE_TB_1, 62, 0, 0, MAP_AREA_MEMORY, 2604, 0, AODB[41] );
(* Set point, short floating point values *)
initError := F_iecInitAOEntry( C_SE_NC_1, 70, 0, 0, MAP_AREA_MEMORY, 2700, 0, AODB[42] );
initError := F_iecInitAOEntry( C_SE_NC_1, 71, 0, 0, MAP_AREA_MEMORY, 2704, 0, AODB[43] );
initError := F_iecInitAOEntry( C_SE_TC_1, 72, 0, 0, MAP_AREA_MEMORY, 2708, 0, AODB[44] );

END_IF

```

Das zugehörige Tutorial SPS-Beispiel kann [hier \[► 674\]](#) heruntergeladen werden.

### 6.5.2.1.3.3 Mapping der SPS- und IEC-Prozessdaten

Die TwinCAT SPS-Prozessdaten werden zur Programmlaufzeit zyklisch in die IEC-Prozessdaten (Applikationsobjekte) und umgekehrt gemappt (kopiert). Für das Mapping der IEC<->SPS Prozessdaten können bis zu vier Prozessdatenbereiche (IO-Eingänge, IO-Ausgänge, Merkerbereich, Datenbereich) als Puffervariablen im SPS-Programm deklariert werden. Die Bytegröße der Puffer ist frei wählbar und kann für jeden Bereich unterschiedlich gewählt werden. Unbenutzte Bereiche müssen nicht unbedingt deklariert werden.

In unserem Einführungsbeispiel deklarieren wir 4 SPS-Prozessdatenbereiche mit jeweils 3000 Bytes:

```

PROGRAM MAIN
VAR
AODB : ARRAY[0..49] OF ST_IEC870_5_101AODBEntry;

init : BOOL := TRUE;
initError : UDINT;

inputs AT%IB0 : ARRAY[0..2999] OF BYTE;
outputs AT%QB0 : ARRAY[0..2999] OF BYTE;
memory AT%MB0 : ARRAY[0..2999] OF BYTE;
data : ARRAY[0..2999] OF BYTE;

END_VAR

```

Die Zuordnung, wie die Prozessdaten zur Laufzeit gemappt werden sollen, wird während der Konfiguration der Applikationsobjekte mit der `F_iecInitAOEntry`-Funktion [\[► 435\]](#) festgelegt.

Siehe auch in: [Applikationsobjekte definieren und konfigurieren \[► 657\]](#).

Die Puffervariablen wurden nun als Byte-Arrays deklariert. Um auf die gewünschten Daten besser zugreifen zu können definieren wir die einzelnen Variablen ein zweites Mal und legen diese auf die entsprechenden Byte/Bit-Offsetadressen. Bei einer Änderung im Byte-Array wird die entsprechende einzelne Variable gleichzeitig geändert und umgekehrt. Dies ist aber nicht zwingend notwendig. Sie können direkt auf die Bytes/Bits der Byte-Array-Puffervariablen zugreifen.

```

VAR_GLOBAL(* Memory offset 0..99 unused *)
(* Single points *)
msgSingle_0 AT%MX100.0 : BOOL;
msgSingle_1 AT%MX100.1 : BOOL;
msgSingle_2 AT%MX100.2 : BOOL;

(* Double points *)
(* Bit 0..1 = first double point,
Bit 2..3 = second double point,
Bit 4..5 = third double point,

```

```

Bit 6..7 = fourth double point *)
msgDouble_0 AT%MB200 : BYTE;

(* Regulating step values *)
msgStep_0 AT%MB300 : BYTE;
msgStep_1 AT%MB301 : BYTE;
msgStep_2 AT%MB302 : BYTE;

(* 32 bit strings *)
msgBitStr_0 AT%MD400 : DWORD := 2#10001000_10001000_10001000_10001000;
msgBitStr_1 AT%MD404 : DWORD := 2#10001000_10001000_10001000_10001000;
msgBitStr_2 AT%MD408 : DWORD := 2#10001000_10001000_10001000_10001000;

(* Measured values, normalized values *)
msgNormalized_0 AT%MW500 : WORD;
msgNormalized_1 AT%MW502 : WORD;
msgNormalized_2 AT%MW504 : WORD;

(* Measured values, scaled values *)
msgScaled_0 AT%MW600 : INT;
msgScaled_1 AT%MW602 : INT;
msgScaled_2 AT%MW604 : INT;

(* Measured values, short floating point values *)
msgFloating_0 AT%MD700 : REAL;
msgFloating_1 AT%MD704 : REAL;
msgFloating_2 AT%MD708 : REAL;

(* Integrated totals *)
msgTotal_0 AT%MD800 : UDINT;
msgTotal_1 AT%MD804 : UDINT;
msgTotal_2 AT%MD808 : UDINT;

(******)
(* Single commands *)
cmdSingle_0 AT%MX2100.0 : BOOL;
cmdSingle_1 AT%MX2100.1 : BOOL;
cmdSingle_2 AT%MX2100.2 : BOOL;

(* Double commands *)
(* Bit 0..1 = first double command,
Bit 2..3 = second double command,
Bit 4..5 = third double command,
Bit 6..7 = fourth double command *)
cmdDouble_0 AT%MB2200 : BYTE;

(* Regulating step commands *)
cmdStep_0 AT%MB2300 : BYTE;
cmdStep_1 AT%MB2301 : BYTE;
cmdStep_2 AT%MB2302 : BYTE;

(* 32 bit string commands *)
cmdBitStr_0 AT%MD2400 : DWORD;
cmdBitStr_1 AT%MD2404 : DWORD;
cmdBitStr_2 AT%MD2408 : DWORD;

(* Set point, normalized values *)
cmdNormalized_0 AT%MW2500 : WORD;
cmdNormalized_1 AT%MW2502 : WORD;
cmdNormalized_2 AT%MW2504 : WORD;

(* Set point, scaled values *)
cmdScaled_0 AT%MW2600 : INT;
cmdScaled_1 AT%MW2602 : INT;
cmdScaled_2 AT%MW2604 : INT;

(* Set point, short floating point values *)
cmdFloating_0 AT%MD2700 : REAL;
cmdFloating_1 AT%MD2704 : REAL;
cmdFloating_2 AT%MD2708 : REAL;
END_VAR

```

## Mapping der IEC<->SPS Prozessdaten in der Unterstation

### Prozessdaten in Überwachungsrichtung (Slave->Master information)

#### Beispiel 1

Single point information (M\_SP\_NA\_1) mit der IOA = 100, SPS Merkerbereich, Byteoffset = 100, Bitoffset = 0.

msgSingle\_0 == memory[100].0 -> Unterstation FB -> ... -> Leitstation

Beispiel 2

Measured value, short floating point value (M\_ME\_NC\_1) mit der IOA = 700, SPS Merkerbereich, Byteoffset = 700, Bitoffset = 0 (bedeutungslos).

msgFloating\_0 == memory[700..703] -> Unterstation FB -> ... -> Leitstation

### Prozessdaten in Steuerungsrichtung (Master->Slave commands)

Beispiel 1

Single command state (C\_SC\_NA\_1) mit der IOA = 10, SPS Merkerbereich, Byteoffset = 2100, Bitoffset = 0.

Leitstation -> ... -> Unterstation FB -> memory[2100].0 == cmdSingle\_0

Beispiel 2

Set point, short floating point value (C\_SE\_NC\_1) mit der IOA = 70, SPS Merkerbereich, Byteoffset = 2700, Bitoffset = 0 (bedeutungslos).

Leitstation -> ... -> Unterstation FB -> memory[2700..2703] == cmdFloating\_0

Das zugehörige Tutorial SPS-Beispiel kann [hier \[▶ 674\]](#) heruntergeladen werden.

### 6.5.2.1.3.4 Aufruf und Deklaration der Unterstation

Die gesamte Funktionalität einer Unterstation ist im Funktionsbaustein FB\_IEC870\_5\_104Slave gekapselt. Mit einer Instanz kann eine Verbindung zum Master aufgebaut werden. Für eine weitere Verbindung deklarieren Sie eine weitere Instanz und übergeben an diese zweite Instanz das gleiche Server-Handle (*hServer*-Variable), oder Sie verwenden den FB\_IEC870\_5-104SlaveGrp [[▶ 395](#)]-Funktionsbaustein (empfohlen). Die IP-Adresse müssen Sie passend zu der IP-Adresse Ihres Zielsystems setzen.

Fügen Sie im Deklarationsteil von MAIN folgenden SPS-Code ein:

```
PROGRAM MAIN
VAR
AODB : ARRAY[0..49] OF ST_IEC870_5_101AODBEntry;

init : BOOL := TRUE;
initError : UDINT;

inputs AT%IB0 : ARRAY[0..2999] OF BYTE;
outputs AT%QB0 : ARRAY[0..2999] OF BYTE;
memory AT%MB0 : ARRAY[0..2999] OF BYTE;
data : ARRAY[0..2999] OF BYTE;

bEnable : BOOL := TRUE;
hServer : T_HSERVER;
server : FB_IEC870_5_104Slave[▶ 392];

END_VAR
```

und im Programmteil wird die Instanz aufgerufen:

```
IF init THEN
init := FALSE;
...

F_CreateServerHnd( '', '127.0.0.1'(* change this! *), 2404, nMode := LISTEN_MODE_CLOSEALL OR
CONNECT_MODE_ENABLEDBG, bEnable, hServer );

ELSE
server( pInputs := ADR( inputs ),
cbInputs := SIZEOF( inputs ),
pOutputs := ADR( outputs ),
```



```

cbOutputs := SIZEOF( outputs ),
pMemory := ADR( memory ),
cbMemory := SIZEOF( memory ),
pData := ADR( data ),
cbData := SIZEOF( data ),
pAOEntries := ADR( AODB ),
cbAOEntries := SIZEOF( AODB ),
hServer := hServer,
bEnable := bEnable );
END_IF

```

Das zugehörige Tutorial SPS-Beispiel kann [hier \[▶ 674\]](#) heruntergeladen werden.

### 6.5.2.1.3.5 Basiseinstellungen: Protokollparameter

Über die IEC60870-5-104-Protokollparameter kann das Verhalten der Unterstation an die Anforderungen des Masters angepasst werden. Die meisten Parameter sind mit Defaultwerten vorbelegt, so dass diese nicht verändert werden müssen.

In unserem Beispiel werden die Werte der iK- und iW-Parameter verändert:

```

IF init THEN
init := FALSE;
...
F_CreateServerHnd( '', '127.0.0.1'(* change this! *), 2404, nMode := LISTEN_MODE_CLOSEALL OR
CONNECT_MODE_ENABLEDBG, bEnable, hServer);

server.protPara.iK := 12;
server.protPara.iW := 8;

ELSE
server( pInputs := ADR( inputs ),
cbInputs := SIZEOF( inputs ),
pOutputs := ADR( outputs ),
...
END_IF

```

Die Dokumentation aller Übertragungsprotokoll-Parameter finden Sie hier: [ST IEC870\\_5\\_104PotocolParams \[▶ 549\]](#).

Das zugehörige Tutorial SPS-Beispiel kann [hier \[▶ 674\]](#) heruntergeladen werden.

### 6.5.2.1.3.6 Basiseinstellungen: Systemparameter

Über die Systemparameter werden z.B. die gemeinsame ASDU-Adresse und die Anwenderfunktionen der Unterstation konfiguriert.

In unserer Einführung konfigurieren wir folgende Systemparameter:

Die gemeinsame ASDU-Adresse wird auf 7 gesetzt (*asduAddr*).

Die Systemzeit der Unterstation wird während der Initialisierung mit der Systemzeit des lokalen TwinCAT PC's synchronisiert (*bUsePCTime*).

Die Synchronisierung der Systemzeit der Unterstation mit dem Uhrzeitsynchronisationsbefehl wird aktiviert (*bSyncTime*).

Während der Synchronisierung der Systemzeit in der Unterstation soll die Systemzeit des TwinCAT PCs nicht synchronisiert werden (*bSyncPCTime*).

Das Senden von M\_EI\_NA\_1 (End of init) an die Zentralstation wird aktiviert (*bEndOfInit*).

Das Senden der periodischen/zyklischen Daten wird deaktiviert (*bPerCyclic*). Die Basiszeit fürs Senden dieser Daten wird auf 5s gesetzt.

Hintergrundabfrage wird deaktiviert (*bBackScan*). Die Zykluszeit für Hintergrundabfrage wird auf 30s gesetzt.

Das lokale Umspeichern und Reset der Zählerstände wird aktiviert (*bPerFRZ*) und die Zykluszeit fürs Umspeichern und Reset auf 15s gesetzt.

Das loggen der Debugg-Meldungen im Application-Log wird aktiviert (*dbgMode*). Es werden Änderungen im Gerätestatus geloggt.

Fügen Sie folgenden SPS-Code in Ihr SPS-Projekt ein:

```
IF init THEN
init := FALSE;
...

server.sysPara.asduAddr := 7;
server.sysPara.bUsePCTime := TRUE;
server.sysPara.bSyncTime := TRUE;
server.sysPara.bSyncPCTime := FALSE;
server.sysPara.bEndOfInit := TRUE;
server.sysPara.bPerCyclic := FALSE;
server.sysPara.tPerCyclicBase := T#5s;
server.sysPara.bBackScan := FALSE;
server.sysPara.tBackScanCycle := T#30s;
server.sysPara.bPerFRZ := TRUE;
server.sysPara.tPerFRZCycle := T#15s;
server.sysPara.dbgMode := (*IEC870_DEBUGMODE_ASDU OR*) IEC870_DEBUGMODE_DEVSTATE;

...
ELSE
server( pInputs := ADR( inputs ),
cbInputs := SIZEOF( inputs ),
pOutputs := ADR( outputs ),
...
END_IF
```

Die Dokumentation aller Systemparameter finden Sie hier: [ST IEC870 5 101SystemParams \[► 487\]](#).

Das zugehörige Tutorial SPS-Beispiel kann [hier \[► 674\]](#) heruntergeladen werden.

### 6.5.2.1.3.7 Stationsabfrage / Generalabfrage

Der Stationsabfragebefehl wird von der Zentralstation eingeleitet. Im Kennungsfeld des Befehls ist auch die Gruppe (1 bis 16 oder allgemein) festgelegt. Die Unterstation überträgt die zu dieser Gruppe dazugehörigen Applikationsobjekte mit der Übertragungsursache <20> bis <36> an die Zentralstation. Applikationsobjekte mit Zeitmarken werden ohne Zeitmarken übertragen.

Konfiguration der Systemparameter

Es müssen keine speziellen Systemparameter gesetzt werden;

Konfiguration der Applikationsobjekte

Der Datenpunkt muss einer oder mehreren Gruppen zugeordnet werden. Der Group-Parameter muss gesetzt werden. Eine Übersicht aller verfügbaren Gruppen finden Sie hier: [Group configuration flags \[► 554\]](#).

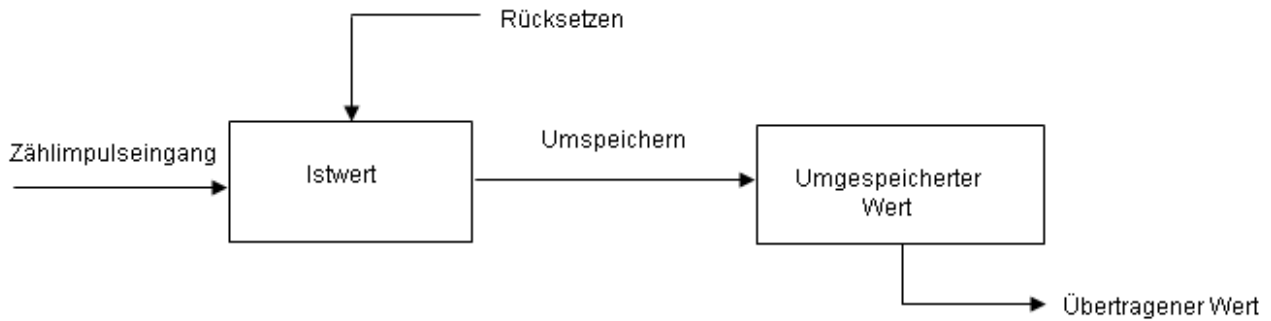
Beispielkonfiguration für einen Datenpunkt der der Gruppe: 1 und der Gruppe: Allgemein zugeordnet wurde.

```
F_iecInitAOEntry( M_SP_NA_1, 100, IEC870_GRP_INROGEN OR IEC870_GRP_INRO1, 0, MAP_AREA_MEMORY, 100,
0, AODB[0] );
```

Das zugehörige Tutorial SPS-Beispiel kann [hier \[► 674\]](#) heruntergeladen werden.

### 6.5.2.1.3.8 Zählwertübertragung

Zur Erläuterung ein allgemeines Modell für die Zählwertübertragung:



Die Istwerte werden durch Zähler aufsummiert. Die Istwerte können durch einen Umspeicherbefehl, der entweder von der Zentralstation empfangen oder örtlich (lokal in der Unterstation) erzeugt wird, periodisch in umgespeicherte Werte umgespeichert (kopiert) werden. Nach dem Umspeichern wird der erfasste Wert entweder auf Null zurückgesetzt (Erfassen von Inkrementalwerten) oder der Zähler fährt mit seinem Betrieb fort (Erfassen von Zählerständen).

Applikationsobjekte mit Zählwerten werden Gruppen zugeordnet. Die Gruppen werden einzeln umgespeichert (frozen), zurückgesetzt (reset) oder übertragen. Die Zentralstation sendet Zählwertabfragebefehle an die Unterstation. In einem Kennungsfeld des Befehls (QCC) wird die durchzuführende Aktion (FRZ) und Gruppe (RQT) festgelegt.

Die Zuordnung der Applikationsobjekte zu den einzelnen Gruppen (1 bis 4 oder allgemein) wird während der Konfiguration durch den Group-Flagparameter festgelegt. Es gibt vier Betriebsarten für die Erfassung von Zählerständen und Inkrementalwerten. Zu jeder Betriebsart sind einige Hinweise zur Konfiguration der Systemparameter oder der Applikationsobjekte aufgeführt.

### Betriebsart A: Örtlich Umspeichern mit Spontanübertragung

Die Unterstation initiiert intern das Umspeichern oder Umspeichern mit Rücksetzen. Die umgespeicherten Zählwerte werden spontan übertragen, nachdem die Funktion Umspeichern oder Umspeichern mit Rücksetzen durchgeführt wurde. Die Zentralstation gibt in dieser Betriebsart keine Zählwertabfragebefehle aus.

#### Konfiguration der Systemparameter:

```
bPerFRZ := TRUE
tPerFRZCycle := T#60s
```

Der erste Parameter aktiviert das örtliche Umspeichern oder Umspeichern mit Rücksetzen. Der zweite Parameter gibt die Zykluszeit an in der das Umspeichern oder Umspeichern mit Rücksetzen durchgeführt wird (z.B.: alle 60 Sekunden).

#### Konfiguration der Applikationsobjekte:

IEC870\_GRP\_SPONTOFF-Gruppenparameter darf nicht gesetzt werden. Dieser würde die spontane Datenübertragung der Zählwerte verhindern.

Der Zählwert wird umgespeichert wenn IEC870\_GRP\_LOCFREEZE-Gruppenparameter gesetzt wurde.

Der Zählwert wird zurückgesetzt wenn IEC870\_GRP\_LOCRESET-Gruppenparameter gesetzt wurde.

Das Örtliche Umspeichern oder Umspeichern mit Rücksetzen wird gleichzeitig für alle Gruppen (1 bis 4 oder allgemein) durchgeführt.

### Betriebsart B: Örtliches Umspeichern mit Zählerabfrage

Die Unterstation initiiert intern das Umspeichern oder Umspeichern mit Rücksetzen. Die umgespeicherten Zählwerte werden durch Zählwertabfragebefehle von der Zentralstation abgefragt. In diesem Fall darf die Zentralstation im Befehlskennungsfeld das Umspeichern oder Umspeichern mit Rücksetzen nicht benutzen (FRZ=0). Die Zählwerte werden allgemein oder in Gruppen (groups) 1 bis 4 abgefragt.

#### Konfiguration der Systemparameter:

```
bPerFRZ := TRUE
```

```
tPerFRZCycle := T#60s
```

Der erste Parameter aktiviert das örtliche Umspeichern oder (und) Rücksetzen. Der zweite Parameter gibt die Zykluszeit an in der das umspeichern oder Umspeichern mit Rücksetzen durchgeführt wird (z.B.: alle 60 Sekunden).

#### Konfiguration der Applikationsobjekte:

IEC870\_GRP\_SPONTOFF-Gruppenparameter muss gesetzt werden. Die Zählwerte sollen nicht spontan zur Zentralstation übertragen werden.

Der Zählwert wird umgespeichert wenn IEC870\_GRP\_LOCFREEZE-Gruppenparameter gesetzt wurde.

Der Zählwert wird zurückgesetzt wenn IEC870\_GRP\_LOCRESET-Gruppenparameter gesetzt wurde.

Das örtliche Umspeichern oder Umspeichern mit Rücksetzen wird gleichzeitig für alle Gruppen (1 bis 4 oder allgemein) durchgeführt.

#### Betriebsart C: Zentralstation leitet das Umspeichern, Umspeichern mit Rücksetzen oder Rücksetzen ein

Ein Zählwertabfragebefehl wird periodisch von der Zentralstation an die Unterstation ausgegeben, um das Umspeichern oder (und) Rücksetzen zu steuern. Dieser Befehl hat aber noch keine Übertragung der Zählwerte zur Folge. Erst ein nachfolgender Zählwertabfragebefehl wird von der Zentralstation gesendet um die umgespeicherten Zählwerte einzusammeln. Ähnlich, wie bei der Betriebsart B.

#### Konfiguration der Systemparameter:

```
bPerFRZ := FALSE
```

```
tPerFRZCycle := T#60s
```

Das örtliche Umspeichern oder (und) Rücksetzen muss deaktiviert werden. Der zweite Parameter wird ignoriert.

#### Konfiguration der Applikationsobjekte:

IEC870\_GRP\_SPONTOFF muss gesetzt werden. Die Zählwerte sollen nicht spontan zur Zentralstation übertragen werden.

IEC870\_GRP\_LOCFREEZE- und IEC870\_GRP\_LOCRESET-Gruppenparameter dürfen nicht gesetzt werden. Die Zentralstation leitet das Umspeichern oder (und) Rücksetzen ein.

Die Zählwerte können einzelnen Gruppen (1 bis 4 oder allgemein) zugeordnet und abgefragt werden (IEC870\_GRP\_REQCOGEN, IEC870\_GRP\_REQCO1, IEC870\_GRP\_REQCO2, IEC870\_GRP\_REQCO3, IEC870\_GRP\_REQCO4).

#### Betriebsart D: Zentralstation leitet das Umspeichern und (oder) Rücksetzen ein und die umgespeicherten Werte werden spontan übertragen

Diese Betriebsart ist eine Kombination des Zählwertbefehls von der Zentralstation wie für Betriebsart C mit einer spontanen Übertragung der Zählwerte wie bei der Betriebsart A.

#### Konfiguration der Systemparameter:

```
bPerFRZ := FALSE
```

```
tPerFRZCycle := T#60s
```

Das örtliche Umspeichern oder (und) Rücksetzen muss deaktiviert werden. Der zweite Parameter wird ignoriert.

#### Konfiguration der Applikationsobjekte:

IEC870\_GRP\_SPONTOFF-Gruppenparameter darf nicht gesetzt werden. Dieser würde die spontane Datenübertragung der Zählwerte verhindern.

IEC870\_GRP\_LOCFREEZE- und IEC870\_GRP\_LOCRESET-Gruppenparameter dürfen nicht gesetzt werden. Die Zentralstation leitet das Umspeichern oder (und) Rücksetzen ein.

Die Zählwerte können einzelnen Gruppen (1 bis 4 oder allgemein) zugeordnet und abgefragt werden (IEC870\_GRP\_REQCOGEN, IEC870\_GRP\_REQCO1, IEC870\_GRP\_REQCO2, IEC870\_GRP\_REQCO3, IEC870\_GRP\_REQCO4).

Das zugehörige Tutorial SPS-Beispiel kann [hier \[► 674\]](#) heruntergeladen werden.

### 6.5.2.1.3.9 Uhrzeitsynchronisation

Die Unterstation kann mit der Uhrzeit der Zentralstation synchronisiert werden. Die dafür nötigen Einstellungen nimmt man in den Systemparametern vor:

```
fbSlave.sysPara.bSyncTime := TRUE; // sync IEC system time after reception of C_CS_NA_1
commando
fbSlave.sysPara.bSyncPCTime := TRUE; // sync local PC system time from master
fbSlave.sysPara.tSyncToPCTime := T#30S;
```

Eine Detailbeschreibung:

**bSyncTime:** Wenn TRUE, aktiviere IEC-Systemzeit Synchronisation beim Empfang des C\_CS\_NA\_1-Kommandos. Default: TRUE.

**bSyncPCTime:** Wenn TRUE, synchronisiere neben der IEC-Systemzeit auch die Systemzeit des TwinCAT PC's (die Windows-Systemzeit in der Taskleiste). Default: FALSE.

**bUsePCTime:** Wenn TRUE, synchronisiere die IEC-Systemzeit mit der Systemzeit des PC's. Nach der Initialisierung der Unterstation wird die IEC-Systemzeit zuerst mit der Systemzeit des TwinCAT PC's (Windows-Systemzeit) synchronisiert. Default: TRUE.

**tSyncToPCTime:** Steuert die zyklische Synchronisation der IEC-Systemzeit mit der Systemzeit des TwinCAT PC's (im Online- und Offline-Mode). Die Zeit bestimmt die Zykluszeit in der die Synchronisation durchgeführt wird. Bei einem Wert = T#0s ist die zyklische Synchronisation deaktiviert. Default: T#0s.

**sPCTimeNetID:** TwinCAT Netzwerkadresse des PCs dessen Systemzeit bei der Synchronisierung benutzt werden soll. Für den lokalen PC kann auch ein Leerstring angegeben werden. Default: Leerstring = Lokaler PC.

**bTimeCOT3:** Wenn TRUE, sende die Systemzeit zur Zentralstation mit der Übertragungsursache <3> *Spontan* beim Stundenwechsel. Default: FALSE

Das zugehörige Tutorial SPS-Beispiel kann [hier \[► 674\]](#) heruntergeladen werden.

### 6.5.2.1.3.10 Hintergrundabfrage

Die Hintergrundabfrage wird zum Auffrischen der Prozessinformationen von der Unterstation zur Zentralstation als zusätzlicher Sicherheitsbeitrag zur Stationsabfrage und spontanen Übertragung angewendet.

Applikationsobjekte mit Typkennungen wie für die Stationsabfrage, dürfen mit der Übertragungsursache <2> Hintergrundabfrage stetig mit niedriger Priorität übertragen werden. Die zulässigen ASDU-Typkennungen sind in der Kompatibilitätsliste der Station aufgelistet (Tabelle Typkennung <-> Übertragungsursache). Die Hintergrundabfrage wird von der Unterstation eingeleitet und ist unabhängig von den Stationsabfragebefehlen.

#### Konfiguration der Systemparameter

Der Übertragungszyklus wird durch [Systemparameter \[► 487\]](#) in der Unterstation festgelegt.

```
bBackScan := TRUE;
tBackScanCycle := T#30s;
```

#### Konfiguration der Applikationsobjekte

Applikationsobjekte, deren Prozessdaten als Hintergrundabfrage übertragen werden sollen, müssen mit dem Group-Flag: IEC870\_GRP\_BACKGROUND konfiguriert werden.

#### Beispiel:

```
F_iecInitAOEntry( M_SP_NA_1, 100, IEC870_GRP_INROGEN OR IEC870_GRP_BACKGROUND, 0, MAP_AREA_MEMORY,
100, 0, AODB[0] );
```

Das zugehörige Tutorial SPS-Beispiel kann [hier \[► 674\]](#) heruntergeladen werden.

### 6.5.2.1.3.11 Zyklische Datenübertragung

Die zyklische Datenübertragung wird ähnlich wie die Hintergrundabfrage von der Unterstation eingeleitet und ist unabhängig von anderen Befehlen aus der Zentralstation. Durch die zyklische Datenübertragung werden die Prozessdaten der Zentralstation kontinuierlich aufgefrischt. Bei den Prozessdaten handelt es sich meistens um Messwerte, die in regulären Zeitabständen erfasst werden. Die zyklische Datenübertragung wird oft benutzt um nicht-zeitkritische, oder sich nicht so schnell ändernde Prozessdaten zu überwachen (z.B. ein Temperatursensor). Die zyklischen/periodischen Daten werden zur Zentralstation mit der Übertragungsursache <1> *Periodic/Cyclic* übertragen. Die zulässigen ASDU-Typkennungen sind in der Kompatibilitätsliste der Station aufgeführt (Tabelle Typkennung <-> Übertragungsursache). Die zyklische Datenübertragung kann über die Systemparameter und die Konfigurationsparameter der Applikationsobjekte konfiguriert werden.

#### Konfiguration der Systemparameter:

```
bPerCyclic : BOOL := TRUE;
tPerCyclicBase : TIME := T#60s;
```

Der erste Parameter aktiviert die zyklische Übertragung. Der zweite Parameter ist die Basiszeit der zyklischen/periodischen Datenübertragung (hier z.B. 60 Sekunden).

#### Konfiguration der Applikationsobjekte:

IEC870\_GRP\_PERCYC-Gruppenparameter muss gesetzt werden;

Der Multiplikator-Parameter (*multiplier*) der `F_iecInitAOEntry`-Funktion muss auf einen Wert <> Null gesetzt werden. Beispiel: Bei einem Multiplikator = 2 und Basiszeit von 60 Sekunden werden die die Prozessdaten des Applikationsobjekts alle 120 Sekunden zur Zentralstation gesendet;

Beispielkonfiguration für einen Messwert der zyklisch alle 120 Sekunden zur Zentralstation übertragen werden soll (measured value, normalized value without time tag, `M_ME_NA_1`).

```
F_iecInitAOEntry( M_ME_NA_1, 222, IEC870_GRP_INROGEN OR IEC870_GRP_PERCYC, 2, MAP_AREA_MEMORY, 6, 0,
AODB[2] );
```

Das zugehörige Tutorial SPS-Beispiel kann [hier \[► 674\]](#) heruntergeladen werden.

### 6.5.2.1.3.12 Befehlsübertragung

Befehle können von der Zentralstation in Steuerungsrichtung (zur Unterstation) gesendet werden.

Ein Einzelbefehl mit der Typkennung 45 (`C_SC_NA_1`) wird zur Steuerung eines Applikationsobjekts benutzt, das in Überwachungsrichtung als Einzelmeldung (`M_SP_NA_1`, `M_SP_TA_1` oder `M_SP_TB_1`) übertragen wird. Ein Doppelbefehl (`C_DC_NA_1`) wird zur Steuerung eines Applikationsobjekts benutzt, das in Überwachungsrichtung als Doppelmeldung (`M_DP_NA1`, `M_DP_TA_1` oder `M_DP_TB_1`) übertragen wird, usw.

#### Konfiguration der Systemparameter:

Es müssen keine speziellen Systemparameter gesetzt werden;

#### Konfiguration der Applikationsobjekte:

Die Applikationsobjekte müssen als Befehle (Datentypen in Steuerungsrichtung) konfiguriert werden;

Die Adressen der Informationsobjekte (IOA's) müssen den Adressen in der Leitstation entsprechen;

Beispiele:

Single Command mit der IOA = 10. Der empfangene Wert wird in den Merkerbereichspuffer, Byteoffset = 100, Bitoffset = 0 kopiert.

```
F_iecInitAOEntry( C_SC_NA_1, 10, 0, 0, MAP_AREA_MEMORY, 100, 0, AODB[24] );
```

Double command mit der IOA = 20. Der empfangene Wert wird in den Merkerbereichspuffer, Byteoffset = 200, Bitoffset = 0..1 kopiert.

```
F_iecInitAOEntry( C_DC_NA_1, 20, 0, 0, MAP_AREA_MEMORY, 200, 0, AODB[27] );
```

Set point, scaled value mit der IOA = 60. Der empfangene Wert wird in den Merkerbereichspuffer, Byteoffset = 600..601, Bitoffset = 0 kopiert.

```
F_iecInitAOEntry( C_SE_NB_1, 60, 0, 0, MAP_AREA_MEMORY, 600, 0, AODB[39] )
```

Das zugehörige Tutorial SPS-Beispiel kann [hier \[▶ 674\]](#) heruntergeladen werden.

### 6.5.2.1.3.13 Abfrage-/Lese-Prozedur

Die Zentralstation sendet Abfragebefehle an die Unterstation.

In dem Abfragebefehl wird die Adresse des abzufragenden Applikationsobjekts übertragen. Die Daten dieses Applikationsobjekts sollen an die Zentralstation gesendet werden. Die Unterstation sendet die Daten mit der Übertragungsursache <5> *Abfrage oder abgefragt*. Die zulässigen ASDU-Typkennungen sind in der Kompatibilitätsliste der Station aufgelistet (Tabelle Typkennung <-> Übertragungsursache).

#### Konfiguration der Systemparameter:

Es müssen keine speziellen Systemparameter gesetzt werden.

#### Konfiguration der Applikationsobjekte:

Es müssen keine speziellen Parameter gesetzt werden.

Das zugehörige Tutorial SPS-Beispiel kann [hier \[▶ 674\]](#) heruntergeladen werden.

### 6.5.2.1.3.14 Doppelübertragung

Alle Applikationsobjekte (Informationsobjekte), die mit der Übertragungsursache <3> *Spontan* übertragen werden, dürfen zweimal übertragen werden, mit und ohne Zeitmarke. Diese Betriebsart wird "Doppelübertragung" genannt.

Doppelübertragung wird von der Unterstation zurzeit nicht unterstützt.

Das zugehörige Tutorial SPS-Beispiel kann [hier \[▶ 674\]](#) heruntergeladen werden.

### 6.5.2.1.3.15 Quality-Flags

Die Quality-Flags (Quality-Descriptor) liefern der Zentralstation zusätzliche Informationen zur Qualität eines Applikationsobjekts. Die Quality-Flags können aus der SPS-Applikation mit Hilfe der `F_iecSetAOQuality` [▶ 437]-Funktion unabhängig voneinander gesetzt/zurückgesetzt werden. Mit der `F_iecGetAOQuality` [▶ 438]-Funktion kann der Status der Quality-Flags abgefragt werden. Jede Änderung der Quality-Flags führt zu einer spontanen Übertragung der Daten zur Zentralstation.

Folgende Quality-Flags werden intern von der Unterstation zur Laufzeit ausgewertet:

IECQ\_BL\_ON (Blocked). Wurden die Prozessdaten des Applikationsobjekts für die Übertragung blockiert, dann wird das Mapping der SPS- und IEC-Prozessdaten für dieses Applikationsobjekt nicht ausgeführt;

Folgende Quality-Flags werden intern von der Unterstation zur Laufzeit gesetzt/zurückgesetzt:

IECQ\_IV\_ON (Invalid). Die Unterstation setzt das Invalid-Flag wenn das Mapping der SPS- und IEC-Prozessdaten nicht durchgeführt werden konnte (z.B. wegen fehlerhafter Konfiguration des Applikationsobjekts). Dieses Verhalten kann durch einen gesetzten Group-Parameter: IEC870\_GRP\_IV\_OFF deaktiviert werden.

Alle anderen Quality-Flags werden unverändert zur Zentralstation gesendet.

Das zugehörige Tutorial SPS-Beispiel kann [hier \[▶ 674\]](#) heruntergeladen werden.

### 6.5.2.1.3.16 Test der Kommunikation

Durch das Setzen der *bChangeIO*-Variable auf TRUE wird eine einfache Simulation der Datenpunkte in Überwachungsrichtung aktiviert und mit FALSE deaktiviert. Bei einer aktiven Verbindung werden die Werte Zyklisch alle 3 Sekunden zur Leitstation übertragen.

```
PROGRAM MAIN
VAR
...
bChangeIO : BOOL; (* TRUE => simulate/modify plc process data *)
timer : TON;
i : INT;
...
END_VAR
...

(*modify plc process data *)
timer( IN := bChangeIO, PT := T#3s );
IF timer.Q THEN
timer( IN := FALSE );

msgSingle_0 := NOT msgSingle_0;
msgSingle_1 := NOT msgSingle_1;
msgSingle_2 := NOT msgSingle_2;

FOR i:= 0 TO 3 DO
IF F_iecGetDPI(msgDouble_0, i) = eIEC870_DPI_ON THEN (* the value of double point allready ON? *)
msgDouble_0 := F_iecSetDPI( msgDouble_0, i, eIEC870_DPI_OFF ); (* change ON => OFF *)
ELSE
msgDouble_0 := F_iecSetDPI( msgDouble_0, i, eIEC870_DPI_ON );(* change OFF => ON *)
END_IF
END_FOR

F_iecIncVTI( msgStep_0 );
F_iecDecVTI( msgStep_1 );

msgBitStr_0 := ROL( msgBitStr_0, 1 );
msgBitStr_1 := ROR( msgBitStr_1, 1 );

msgNormalized_0 := msgNormalized_0 + 1;
msgNormalized_1 := msgNormalized_1 + 2;

msgScaled_0 := msgScaled_0 + 3;
msgScaled_1 := msgScaled_1 - 3;

msgFloating_0 := msgFloating_0 + 0.1;
msgFloating_1 := msgFloating_1 + 1.5;

msgTotal_0 := msgTotal_0 + 1;
msgTotal_1 := msgTotal_1 + 2;
END_IF
...

```

Das zugehörige Tutorial SPS-Beispiel kann [hier \[► 674\]](#) heruntergeladen werden.

### 6.5.2.1.3.17 Protokoll- und Stationsfehler

Die Stationsfehlermeldungen werden in einem FIFO abgelegt. Es können bis zu 10 Fehlermeldungen zwischengespeichert werden. Bei fatalen Kommunikationsfehlern (z.B. Fehler der Verbindungsschicht, die Checksumme des Frames passt nicht) wird die Verbindung unterbrochen und muss neu aufgebaut werden. Fehler in der Applikationsschicht (z.B. der ASDU-Sendepuffer ist wegen zu vieler Frames übergelaufen) werden nur geloggt und führen nicht zum Verbindungsabbruch. Es immer noch möglich auch bei diesen Fehlern die Verbindung aus der Applikation zu unterbrechen. Neben dem Fehler-Code wird auch die Fehlerquelle in der Fehlermeldung abgelegt. Dieses erleichtert die Lokalisierung des Fehlers.

#### Beispiel

Die anfallenden Fehlermeldungen einer IEC 60870-5-104 Unterstation können durch folgenden Aufruf ausgelesen werden:



```

PROGRAM MAIN
VAR
...

server : FB_IEC870_5_104Slave;

...
END_VAR

....

REPEAT
server.system.device.errors.RemoveError();
IF server.system.device.errors.bOk THEN
ADSLOGSTR( ADSLOG_MSGTYPE_ERROR OR ADSLOG_MSGTYPE_LOG,
'IEC60870-5-104 slave error: 0x%$s',
DWORD_TO_HEXSTR( server.system.device.errors.getError.nErrId, 8, FALSE) );
END_IF
UNTIL NOT server.system.device.errors.bOk
END_REPEAT

...

```

Das zugehörige Tutorial SPS-Beispiel kann [hier \[▶ 674\]](#) heruntergeladen werden.

### 6.5.2.1.3.18 Erweiterte Diagnose

Debugmeldungen, die ins Application-Log geschrieben werden erleichtern eine Fehlersuche im System. Zur Zeit können 5 Stufen der Debugmeldungen in einer IEC-Applikation aktiviert werden. Vier Stufen werden durch den dbgMode-Systemparameter der Unterstation aktiviert ([ST IEC870 5 101SystemParams \[▶ 487\]](#)).

Debugmeldungen, die beim Aufbauen oder Abbauen der TCP/IP-Verbindung geloggt werden (Meldungen der TcSocketHelper.Lib). Diese Meldungen können durch den nMode-Parameter: CONNECT\_MODE\_ENABLEDBG beim Aufruf der F\_CreateServerHnd()-Funktion aktiviert werden;

Stationsstatus-Meldungen (dbgMode: IEC870\_DEBUGMODE\_DEVSTATE);

Hexadezimale Ausgabe der ASDU's (dbgMode: IEC870\_DEBUGMODE\_ASDU). Pro Zeile werden 32-ASDU-Datenbytes als hexadezimale Zahlen ausgegeben. Längere ASDU's werden auf mehrere Zeilen verteilt;

Hexadezimale Ausgabe der APDU's (TCP/IP Telegramme, dbgMode: IEC870\_DEBUGMODE\_LINKLAYER). Pro Zeile werden 32-APDU-Datenbytes als hexadezimale Zahlen ausgegeben. Ähnlich wie bei 3. werden längere APDU's auf mehrere Zeilen verteilt;

Ausgabe der Link-Layer-Fehler als Meldungen (dbgMode: IEC870\_DEBUGMODE\_LINKERROR);

Um die aktivierten Debugmeldungen zu sehen starten Sie TwinCAT System Manager und aktivieren Sie die Loggeransicht. Die nachfolgende Grafik zeigt eine Debugausgabe. Die ersten drei unterschiedlichen Typen der Meldungen wurden mit entsprechenden Zahlen markiert.

```

Server (Port) | Timestamp | Meldung
TCPLC (80... | 4/25/2005 1:08:29 P... | Class1 TX FIFO==>6B 01 07 01 07 00 00 00 00 00 63 7F 08 0D 39 04 05
TCPLC (80... | 4/25/2005 1:08:29 P... | Class1 TX FIFO<==6B 01 07 01 07 00 00 00 00 00 63 7F 08 0D 39 04 05
TCPLC (80... | 4/25/2005 1:08:29 P... | RX Fifo==>6B 01 06 01 07 00 00 00 00 00 AF 7B 08 0D 19 04 05
TCPLC (80... | 4/25/2005 1:08:29 P... | RX Fifo<==6B 01 06 01 07 00 00 00 00 00 AF 7B 08 0D 19 04 05
TCPLC (80... | 4/25/2005 1:08:29 P... | Class1 TX FIFO==>64 01 0A 01 07 00 00 00 00 14
TCPLC (80... | 4/25/2005 1:08:29 P... | Class1 TX FIFO<==64 01 0A 01 07 00 00 00 00 14
TCPLC (80... | 4/25/2005 1:08:29 P... | Class1 TX FIFO==>0D B2 14 01 07 00 BC 02 00 00 00 00 00 00 00 00 00 00
TCPLC (80... | 4/25/2005 1:08:29 P... | Class1 TX FIFO==>0B B2 14 01 07 00 58 02 00 00 00 00 00 00 00 00
TCPLC (80... | 4/25/2005 1:08:29 P... | Class1 TX FIFO==>09 B2 14 01 07 00 F4 01 00 00 00 00 00 00 00
TCPLC (80... | 4/25/2005 1:08:29 P... | Class1 TX FIFO<==0D B2 14 01 07 00 BC 02 00 00 00 00 00 00 00 00 00
TCPLC (80... | 4/25/2005 1:08:29 P... | Class1 TX FIFO==>07 B2 14 01 07 00 90 01 00 00 00 00 00 00 00 00 00
TCPLC (80... | 4/25/2005 1:08:29 P... | Class1 TX FIFO==>05 B2 14 01 07 00 2C 01 00 00 00 00 00 00 00
TCPLC (80... | 4/25/2005 1:08:29 P... | Class1 TX FIFO==>03 B2 14 01 07 00 C8 00 00 00 00 00 00 00
TCPLC (80... | 4/25/2005 1:08:29 P... | Class1 TX FIFO==>01 B2 14 01 07 00 64 00 00 00 00 00 00 00
TCPLC (80... | 4/25/2005 1:08:29 P... | Class1 TX FIFO<==0B B2 14 01 07 00 58 02 00 00 00 00 00 00 00 00
TCPLC (80... | 4/25/2005 1:08:29 P... | Class1 TX FIFO<==09 B2 14 01 07 00 F4 01 00 00 00 00 00 00 00
TCPLC (80... | 4/25/2005 1:08:29 P... | Class1 TX FIFO<==07 B2 14 01 07 00 90 01 00 00 00 00 00 00 00 00 00
TCPLC (80... | 4/25/2005 1:08:29 P... | Class1 TX FIFO<==05 B2 14 01 07 00 2C 01 00 00 00 00 00 00 00
TCPLC (80... | 4/25/2005 1:08:29 P... | Class1 TX FIFO<==03 B2 14 01 07 00 C8 00 00 00 00 00 00 00
TCPLC (80... | 4/25/2005 1:08:29 P... | Class1 TX FIFO<==01 B2 14 01 07 00 64 00 00 00 00 00 00 00
TCPLC (80... | 4/25/2005 1:08:29 P... | Class1 TX FIFO==>64 01 07 01 07 00 00 00 00 14
TCPLC (80... | 4/25/2005 1:08:29 P... | Class1 TX FIFO<==64 01 07 01 07 00 00 00 00 14
TCPLC (80... | 4/25/2005 1:08:29 P... | RX Fifo==>64 01 06 01 07 00 00 00 00 14
TCPLC (80... | 4/25/2005 1:08:29 P... | RX Fifo<==64 01 06 01 07 00 00 00 00 14
TCPLC (80... | 4/25/2005 1:08:29 P... | Class1 TX FIFO==>46 01 04 00 07 00 00 00 00 00
TCPLC (80... | 4/25/2005 1:08:29 P... | Class1 TX FIFO<==46 01 04 00 07 00 00 00 00 00
TCPLC (80... | 4/25/2005 1:08:29 P... | TcIEC870_5_104Slave.Lib::ESTABLISHED (STOPDT) => ESTABLISHED (STARTDT) [2]
TCPLC (80... | 4/25/2005 1:08:29 P... | TcIEC870_5_104Slave.Lib::PASSIVE_OPEN => ESTABLISHED (STOPDT) [2]
TCPLC (80... | 4/25/2005 1:08:29 P... | TcSocketHelper.lib::FB_SocketAccept::internal handle:2, error code: 0
    
```

**6.5.2.1.3.19 Glossar**

Begriff	Beschreibung
Unterstation, Slave, Server	Synonyme für eine untergeordnete Station (sie wird überwacht)
Zentralstation, Leitstation, Master, Client	Synonyme für eine übergeordnete Station (Leitstation, sie überwacht andere Stationen)
Steuerungsrichtung	Datenübertragungsrichtung von der Zentralstation zur Unterstation
Überwachungsrichtung	Datenübertragungsrichtung von der Unterstation zur Zentralstation
Applikationsobjekte	IEC-Informationsobjekte in der TwinCAT SPS-Applikation (Single Points, Double Points, Measured Values, Short Floating Point Values usw.)
APDU	Protokolldateneinheit der Anwendungsschicht (application protocol data unit)
APCI	Protokollsteuerinformation der Anwendungsschicht (application protocol control information)
ASDU	Dienstdateneinheit der Anwendungsschicht (application service data unit)
IOA, Adresse des Informationsobjekts	Adresse des Single Points, Double Points usw. (information object address)

**6.5.2.1.4 Unterstation Tutorial (Slave)**

Eine komplexere Unterstation-Applikation mit unterschiedlichen Datenpunkten in beide Richtungen und einer Verbindung zur Zentralstation. Die Beispielapplikation verwendet das TwinCAT IEC 60870-5-104 High-Level Interface.

Download TwinCAT XAE Project (\*.zip): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/984642315.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/984642315.zip)

Im Beispielprojekt sind folgende Stationsparameter für die Unterstation eingestellt:

- Server host address (IP-Adresse der Unterstation): **127.0.0.1(Sie müssen mindestens diesen Parameter an Ihre Zielplattform anpassen!).**
- Server port address (Portadresse der Unterstation): **2404.**
- k: **12.**
- w: **8.**
- t0: **30s.**
- t1: **15s.**
- t2: **10s.**
- t3: **20s.**
- Cause of transfer size: **2 octets (inclusive originator address).**
- Common ASDU address size: **2 octets.**
- Information object address size: **3 octets.**
- Originator address: **1.**
- Common ASDU address: **7.**
- Max. APDU length: **253.**

Datenpunktconfiguration (sie sollte gleich sein mit der Datenpunktconfiguration der Zentralstation):

- Single point information: M\_SP\_NA\_1 (IOA := 100), M\_SP\_NA\_1 (IOA := 101), M\_SP\_TB\_1 (IOA := 102).
- Double point information: M\_DP\_NA\_1 (IOA := 200), M\_DP\_NA\_1 (IOA := 201), M\_DP\_TB\_1 (IOA := 202).
- Regulating step value: M\_ST\_NA\_1 (IOA := 300), M\_ST\_NA\_1 (IOA := 301), M\_ST\_TB\_1 (IOA := 302).
- Bitstring value: M\_BO\_NA\_1 (IOA := 400), M\_BO\_NA\_1 (IOA := 401), M\_BO\_TB\_1 (IOA := 402).
- Measured value, normalized value: M\_ME\_NA\_1 (IOA := 500), M\_ME\_NA\_1 (IOA := 501), M\_ME\_TD\_1 (IOA := 502).
- Measured value, scaled value: M\_ME\_NB\_1 (IOA := 600), M\_ME\_NB\_1 (IOA := 601), M\_ME\_TD\_1 (IOA := 602).
- Measured value, short floating point value: M\_ME\_NC\_1 (IOA := 700), M\_ME\_NC\_1 (IOA := 701), M\_ME\_TF\_1 (IOA := 702).
- Binary counter value: M\_IT\_NA\_1 (IOA := 800), M\_IT\_NA\_1 (IOA := 801), M\_IT\_TB\_1 (IOA := 802).
- Single command: C\_SC\_NA\_1 (IOA := 10), C\_SC\_NA\_1 (IOA := 11), C\_SC\_TA\_1 (IOA := 12).
- Double command: C\_DC\_NA\_1 (IOA := 20), C\_DC\_NA\_1 (IOA := 21), C\_DC\_TA\_1 (IOA := 22).
- Bitstring command: C\_BO\_NA\_1 (IOA := 40), C\_BO\_NA\_1 (IOA := 41), C\_BO\_TA\_1 (IOA := 42).
- Setpoint command, normalized value: C\_SE\_NA\_1 (IOA := 50), C\_SE\_NA\_1 (IOA := 51), C\_SE\_TA\_1 (IOA := 52).
- Setpoint command, scaled value: C\_SE\_NB\_1 (IOA := 60), C\_SE\_NB\_1 (IOA := 61), C\_SE\_TB\_1 (IOA := 62).
- Setpoint command, short floating point value: C\_SE\_NC\_1 (IOA := 70), C\_SE\_NC\_1 (IOA := 71), C\_SE\_TC\_1 (IOA := 72).

Beispielimplementierung folgender Dienste:

- Ende der Initialisierung (M\_EI\_NA\_1) wird einmalig gesendet.
- Testbefehl (C\_TS\_TA\_1).
- Uhrzeitsynchronisationsbefehl (C\_CS\_NA\_1).
- Generalabfragebefehl (C\_IC\_NA\_1) der Gruppe: Allgemein (eIEC870\_QOI\_INROGEN).
- Zählerabfragebefehl (C\_IC\_NA\_1) der Gruppe: Allgemein (eIEC870\_RQT\_REQCOGEN, Mode A). Zähler werden lokal in der Unterstation alle 15 Sekunden umgespeichert und als spontane Daten an die Zentralstation übertragen.
- Befehlsausführung einiger Befehle (direkte Befehle, siehe obere Datenpunktconfiguration).

- Simulation und Übertragung der Spontanen Daten alle 10 Sekunden wenn die Variable **bSendSpont** auf TRUE gesetzt wurde (siehe obere Datenpunktkonfiguration).

Folgende Tabelle beschreibt kurz die Hauptkomponenten des Projekts:

Komponente	Typ	Beschreibung
MAIN	PRG	Dieses Programmteil implementiert das Fernwirkprotokoll.
hServer	T_HSERVER (STRUCT)	Das Server-Verbindungshandle.
fbServer	FB_IEC870_5_104Slave (FB)	Implementiert die Unterstation. Eine Instanz dieses Bausteins kann nur mit einer Zentralstation kommunizieren. Für jede weitere Zentralstation mit der kommuniziert werden soll wird eine weitere Instanz dieses Funktionsbausteins benötigt.
F_CreateServerHnd	FUNCTION	Diese Funktion initialisiert das Server-Verbindungshandle.
AODB	ARRAY OF ST_IEC870_5_101AODBEntry (STRUCT)	Applikationsobjektdatenbank. Es ist eine Array-Variable. Für jeden Datenpunkt wird ein Arrayelement benötigt.
F_iecInitAOEntry	FUNCTION	Im Initialisierungsschritt werden die einzelnen Datenpunkte der Station mit diesem Funktionsaufruf einmalig konfiguriert.
memory	ST_MemoryData (STRUCT)	Diese Struktur beinhaltet die empfangenen und die zu sendenden Prozessdaten die ständig aktualisiert werden.
F_iecSetAOQuality	FUNCTION	Diese Funktion demonstriert wie die Quality-Flags von M_SP_NA_1( IOA := 100) gesetzt/zurückgesetzt werden können.
F_iecGetAOQuality	FUNCTION	Diese Funktion demonstriert wie die Quality-Flags von M_SP_NA_1( IOA := 100) abgefragt werden können.

### 6.5.2.1.5 Unterstation Verbindungs-Redundanz (Slave)

Eine komplexere Unterstation-Applikation mit unterschiedlichen Datenpunkten in beide Richtungen und zwei Verbindungen zur Zentralstation. Die Beispielapplikation verwendet das TwinCAT IEC 60870-5-104 High-Level Interface.

Download TwinCAT XAE Project (\*.zip): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/984645259.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/984645259.zip)

Im Beispielprojekt sind folgende Stationsparameter für die Unterstation eingestellt:

- Server host address (IP-Adresse der Unterstation): **127.0.0.1(Sie müssen mindestens diesen Parameter an Ihre Zielplattform anpassen!).**
- Server port address (Portadresse der Unterstation): **2404.**
- k: **12.**
- w: **8.**
- t0: **30s.**
- t1: **15s.**
- t2: **10s.**
- t3: **20s.**
- Cause of transfer size: **2 octets (inclusive originator address).**
- Common ASDU address size: **2 octets.**
- Information object address size: **3 octets.**
- Originator address: **1.**

- Common ASDU address: **7**.
- Max. APDU length: **253**.

Datenpunktconfiguration (sollte gleich sein mit der Datenpunktconfiguration der Zentralstation):

- Single point information: M\_SP\_NA\_1 (IOA := 100), M\_SP\_NA\_1 (IOA := 101), M\_SP\_TB\_1 (IOA := 102).
- Double point information: M\_DP\_NA\_1 (IOA := 200), M\_DP\_NA\_1 (IOA := 201), M\_DP\_TB\_1 (IOA := 202).
- Regulating step value: M\_ST\_NA\_1 (IOA := 300), M\_ST\_NA\_1 (IOA := 301), M\_ST\_TB\_1 (IOA := 302).
- Bitstring value: M\_BO\_NA\_1 (IOA := 400), M\_BO\_NA\_1 (IOA := 401), M\_BO\_TB\_1 (IOA := 402).
- Measured value, normalized value: M\_ME\_NA\_1 (IOA := 500), M\_ME\_NA\_1 (IOA := 501), M\_ME\_TD\_1 (IOA := 502).
- Measured value, scaled value: M\_ME\_NB\_1 (IOA := 600), M\_ME\_NB\_1 (IOA := 601), M\_ME\_TD\_1 (IOA := 602).
- Measured value, short floating point value: M\_ME\_NC\_1 (IOA := 700), M\_ME\_NC\_1 (IOA := 701), M\_ME\_TF\_1 (IOA := 702).
- Binary counter value: M\_IT\_NA\_1 (IOA := 800), M\_IT\_NA\_1 (IOA := 801), M\_IT\_TB\_1 (IOA := 802).
- Single command: C\_SC\_NA\_1 (IOA := 10), C\_SC\_NA\_1 (IOA := 11), C\_SC\_TA\_1 (IOA := 12).
- Double command: C\_DC\_NA\_1 (IOA := 20), C\_DC\_NA\_1 (IOA := 21), C\_DC\_TA\_1 (IOA := 22).
- Bitstring command: C\_BO\_NA\_1 (IOA := 40), C\_BO\_NA\_1 (IOA := 41), C\_BO\_TA\_1 (IOA := 42).
- Setpoint command, normalized value: C\_SE\_NA\_1 (IOA := 50), C\_SE\_NA\_1 (IOA := 51), C\_SE\_TA\_1 (IOA := 52).
- Setpoint command, scaled value: C\_SE\_NB\_1 (IOA := 60), C\_SE\_NB\_1 (IOA := 61), C\_SE\_TB\_1 (IOA := 62).
- Setpoint command, short floating point value: C\_SE\_NC\_1 (IOA := 70), C\_SE\_NC\_1 (IOA := 71), C\_SE\_TC\_1 (IOA := 72).

Beispielimplementierung folgender Dienste:

- Ende der Initialisierung (M\_EI\_NA\_1) wird einmalig gesendet.
- Testbefehl (C\_TS\_TA\_1).
- Uhrzeitsynchronisationsbefehl (C\_CS\_NA\_1).
- Generalabfragebefehl (C\_IC\_NA\_1) der Gruppe: Allgemein (eIEC870\_QOI\_INROGEN).
- Zählerabfragebefehl (C\_IC\_NA\_1) der Gruppe: Allgemein (eIEC870\_RQT\_REQCOGEN, Mode A). Zähler werden lokal in der Unterstation alle 15 Sekunden umgespeichert und als spontane Daten an die Zentralstation übertragen.
- Befehlsausführung einiger Befehle (direkte Befehle, siehe obere Datenpunktconfiguration).
- Simulation und Übertragung der Spontanen Daten alle 10 Sekunden wenn die Variable **bSendSpont** auf TRUE gesetzt wurde (siehe obere Datenpunktconfiguration).

Folgende Tabelle beschreibt kurz die Hauptkomponenten des Projekts:

Komponente	Typ	Beschreibung
MAIN	PRG	Dieses Programmteil implementiert das Fernwirkprotokoll
hServer	T_HSERVER (STRUCT)	Das Server-Verbindungshandle.
fbServer	FB_IEC870_5_104SlaveGrp (FB)	Implementiert die Unterstation. Eine Instanz dieses Bausteins kann nur mit einer Zentralstation kommunizieren. Für jede weitere Zentralstation mit der kommuniziert werden soll wird eine weitere Instanz dieses Funktionsbausteins benötigt.
F_CreateServerHnd	FUNCTION	Diese Funktion initialisiert das Server-Verbindungshandle.
AODB	ARRAY OF ST_IEC870_5_101AO DBEntry (STRUCT)	Applikationsobjektdatenbank. Es ist eine Array-Variable. Für jeden Datenpunkt wird ein Arrayelement benötigt.
F_iecnitAOEntry	FUNCTION	Im Initialisierungsschritt werden die einzelnen Datenpunkte der Station mit diesem Funktionsaufruf einmalig konfiguriert.
memory	ST_MemoryData (STRUCT)	Diese Struktur beinhaltet die empfangenen und die zu sendenden Prozessdaten die ständig aktualisiert werden.
F_iecSetAOQuality	FUNCTION	Diese Funktion demonstriert wie die Quality-Flags von M_SP_NA_1( IOA := 100) gesetzt/zurückgesetzt werden können.
F_iecGetAOQuality	FUNCTION	Diese Funktion demonstriert wie die Quality-Flags von M_SP_NA_1( IOA := 100) abgefragt werden können.
aConnections	ARRAY OF ST_IEC870_5_104ServerConnection (STRUCT)	Jedes Arrayelement repräsentiert eine Verbindung zur Zentralstation.

### 6.5.2.1.6 Szenarios

#### 6.5.2.1.6.1 Spontane Datenübertragung und Generalabfrage

Das hier vorgestellte Beispielprogramm zeigt einen IEC 60870-5-104 Slave mit High Level Schnittstelle und einer einfachen Kommunikation von zwei Datenpunkten. Ein Datenpunkt wird in Control und ein anderer in Monitor Direction jeweils spontan übertragen. Für den Datenpunkt in Monitor Direction ist zusätzlich eine Generalabfrage konfiguriert.

Download TwinCAT XAE Project (\*.zip): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/1243749515.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/1243749515.zip)

Im Beispiel sind folgende Parameter für die Unterstation (Slave) eingestellt:

- Server Host-Adresse (IP-Adresse des Slaves/die eigene IP): 127.0.0.1 (Mindestens dieser Parameter muss individuell angepasst werden)
- Server Port-Adresse (Port-Adresse des Slaves/der eigene Port): 2404
- k: 12
- w: 8
- t0: 30s
- t1: 15s
- t2: 10s
- t3: 20s
- Allgemeine ASDU-Adresse: 3

- Cause of transfer size: 2 octets
- Common ASDU address size: 2 octets
- Information object address size: 3 octets
- Max. APDU length: 253



- Die Applikationen stellen eine beispielhafte Umsetzung der High-Level Schnittstelle dar. Es wird kein Anspruch auf Vollständigkeit und Richtigkeit übernommen. Stellen Sie die IP-Adresse des Slaves sowohl im Master- als auch im Slave-Teil ein! Überprüfen Sie, dass der Port 2404 nicht von einer Firewall blockiert wird!
- Die Verbindungsherstellung beginnt nach dem Starten des Programms. Die Datenpunkte werden in der Globalen Variablenliste (GVL) verwaltet.

### 6.5.2.1.6.2 Generalabfrage und Gruppen

Das hier vorgestellte Beispielprogramm zeigt einen IEC 60870-5-104 Slave mit High Level Schnittstelle. Das Beispiel geht darauf ein Gruppen von Datentypen zu definieren. Es wird eine Generalabfrage und eine gesonderte Gruppe 1 konfiguriert. Die Generalabfrage erfolgt alle 30s und die Gruppe 1 wird alle 10s abgefragt. Es werden unter anderem Single Points, Bitstrings und Measured Values kommuniziert.

Download TwinCAT XAE Project (\*.zip): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/1243754123.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/1243754123.zip)

Im Beispiel sind folgende Parameter für die Unterstation (Slave) eingestellt:

- Server Host-Adresse (IP-Adresse des Slaves): 127.0.0.1 (Mindestens dieser Parameter muss individuell angepasst werden)
- Server Port-Adresse (Port-Adresse des Slaves): 2404
- k: 12
- w: 8
- t0: 30s
- t1: 15s
- t2: 10s
- t3: 20s
- Allgemeine ASDU-Adresse: 3
- Cause of transfer size: 2 octets
- Common ASDU address size: 2 octets
- Information object address size: 3 octets
- Max. APDU length: 253



- Die Applikationen stellen eine beispielhafte Umsetzung der High-Level Schnittstelle dar. Es wird kein Anspruch auf Vollständigkeit und Richtigkeit übernommen. Stellen Sie die IP-Adresse des Slaves sowohl im Master- als auch im Slave-Teil ein! Überprüfen Sie, dass der Port 2404 nicht von einer Firewall blockiert wird!
- Die Verbindungsherstellung beginnt nach dem Starten des Programms. Die Datenpunkte werden in der Globalen Variablenliste (GVL) verwaltet.

### 6.5.2.1.6.3 Zählwertabfrage und Zeitsynchronisation

Das hier vorgestellte Beispielprogramm zeigt einen IEC 60870-5-104 Slave mit High Level Schnittstelle. Kern des Beispiels ist die Uhrzeit-Synchronisation von Master zu Slave und die Zählwertabfrage. Die Synchronisation der Uhrzeit findet hier beispielhaft alle 120s statt. Ein Datentyp M\_IT\_NA\_1 sorgt für die Zählwert-Übertragung durch eine General-Zähler-Abfrage.

Download TwinCAT XAE Project (\*.zip): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/1243758731.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/1243758731.zip)

Im Beispiel sind folgende Parameter für die Unterstation (Slave) eingestellt:

- Server Host-Adresse (IP-Adresse des Slaves): 127.0.0.1 (Mindestens dieser Parameter muss individuell angepasst werden)
- Server Port-Adresse (Port-Adresse des Slaves): 2404
- k: 12
- w: 8
- t0: 30s
- t1: 15s
- t2: 10s
- t3: 20s
- Allgemeine ASDU-Adresse: 3
- Cause of transfer size: 2 octets
- Common ASDU address size: 2 octets
- Information object address size: 3 octets
- Max. APDU length: 253



- Die Applikationen stellen eine beispielhafte Umsetzung der High-Level Schnittstelle dar. Es wird kein Anspruch auf Vollständigkeit und Richtigkeit übernommen. Stellen Sie die IP-Adresse des Slaves sowohl im Master- als auch im Slave-Teil ein! Überprüfen Sie, dass der Port 2404 nicht von einer Firewall blockiert wird!
- Die Verbindungsherstellung beginnt nach dem Starten des Programms. Die Datenpunkte werden in der Globalen Variablenliste (GVL) verwaltet.

## 6.5.2.2 Low-Level Interface

### 6.5.2.2.1 Quick start

Einfaches Projekt mit kompletten Quellen finden Sie hier: [IEC 60870-5-104 Unterstation-Low-Level-Beispiel \[▶ 680\]](#).

Eine SPS-Applikation, die über das "Low-Level"-Interface mit einer Zentralstation kommunizieren soll, benötigt folgende Ressourcen:

- Eine Instanz des Kommunikationsbausteins: [FB\\_IEC870\\_5\\_104TProtocol \[▶ 411\]](#);
- Eine Instanz des Funktionsbausteins zur Herstellung der TCP/IP-Verbindung: [FB\\_ServerClientConnection](#) oder [FB\\_ClientServerConnection](#);
- Eine Instanz des TX/RX-Datenpuffers: [ST\\_IEC870\\_5\\_101TBuffer \[▶ 481\]](#);
- Eine Instanz des Funktionsbausteins zur Manipulation des TX/RX-Datenpuffers: [FB\\_IEC870\\_5\\_101TBufferCtrl \[▶ 403\]](#);

### 6.5.2.2.2 Unterstation (Slave)

Einfache Unterstation-Applikation die das TwinCAT IEC 60870-5-104 Low-Level Interface verwendet. Diese Beispielapplikation kann mit einer Zentralstation über eine Netzwerk-Schnittstelle kommunizieren.

Download TwinCAT XAE Project (\*.zip): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/984648203.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/984648203.zip)

Im Beispielprojekt sind folgende Stationsparameter für die Unterstation eingestellt:

- Server host address (IP-Adresse der Unterstation): **127.0.0.1 (Sie müssen mindestens diesen Parameter an Ihre Zielplattform anpassen!)**
- Server port address (Portadresse der Unterstation): **2404.**
- k: **12.**



- w: **8**.
- t0: **30s**.
- t1: **15s**.
- t2: **10s**.
- t3: **20s**.
- Cause of transfer size: **2 octets (inclusive originator address)**.
- Common ASDU address size: **2 octets**.
- Information object address size: **3 octets**.
- Originator address: **0**.
- Common ASDU address: **7**.
- Max. APDU length: **253**.

Datenpunktconfiguration (sie sollte gleich sein mit der Datenpunktconfiguration der Zentralstation):

- Single point information: M\_SP\_NA\_1 (IOA := 100), M\_SP\_NA\_1 (IOA := 101), M\_SP\_TB\_1 (IOA := 102).
- Double point information: M\_DP\_NA\_1 (IOA := 200).
- Regulating step value: M\_ST\_NA\_1 (IOA := 300).
- Bitstring value: M\_BO\_NA\_1 (IOA := 400).
- Measured value, normalized value: M\_ME\_NA\_1 (IOA := 500).
- Measured value, scaled value: M\_ME\_NB\_1 (IOA := 600).
- Measured value, short floating point value: M\_ME\_NC\_1 (IOA := 700).
- Binary counter value: M\_IT\_NA\_1 (IOA := 800), M\_IT\_NA\_1 (IOA := 801), M\_IT\_TB\_1 (IOA := 802).
- Single command: C\_SC\_NA\_1 (IOA := 10).
- Double command: C\_DC\_NA\_1 (IOA := 20).
- Regulating step command: C\_RC\_NA\_1 (IOA := 30).
- Bitstring command: C\_BO\_NA\_1 (IOA := 40).
- Setpoint command, normalized value: C\_SE\_NA\_1 (IOA := 50).
- Setpoint command, scaled value: C\_SE\_NB\_1 (IOA := 60).
- Setpoint command, short floating point value: C\_SE\_NC\_1 (IOA := 70).

Beispielimplementierung folgender Dienste:

- Testbefehl (C\_TS\_TA\_1).
- Uhrzeitsynchronisationsbefehl (C\_CS\_NA\_1).
- Generalabfragebefehl (C\_IC\_NA\_1) der Gruppe: Allgemein (eIEC870\_QOI\_INROGEN).
- Zählerabfragebefehl (C\_IC\_NA\_1) der Gruppe: Allgemein (eIEC870\_RQT\_REQCOGEN), Mode A. Die Zählwerte werden lokal alle 60 Sekunden umgespeichert.
- Befehlsausführung einiger Befehle (siehe obere Datenpunktconfiguration).
- Ende der Initialisierung (M\_EI\_NA\_1) wird einmalig zur Zentralstation gesendet (nach dem die Verbindung hergestellt wurde).
- Zyklische Datenübertragung alle 60 Sekunden.
- Simulation und Übertragung der spontanen Daten alle 60 Sekunden wenn die Variable **bSimulate** auf TRUE gesetzt wurde.

Folgende Tabelle beschreibt kurz die Hauptkomponenten des Projekts:

Komponente	Typ	Beschreibung
MAIN	PRG	Dieses Programmteil implementiert das Fernwirkprotokoll.
fbServer	FB_IEC870_5_104S (FB)	Implementiert die Unterstation. Eine Instanz dieses Bausteins kann nur mit einer Zentralstation kommunizieren. Für jede weitere Zentralstation mit der kommuniziert werden soll wird eine weitere Instanz dieses Funktionsbausteins benötigt.
fbServices	FB_IEC870_5_104SS ervices (FB)	Beinhaltet eine einfache Implementierung einiger Dienste wie z.B. Generalabfrage, Uhrzeitsynchronisation, Testbefehle usw.
stProcPara	ST_IEC870_5_104SP rocData (STRUCT)	Hier befinden sich die empfangenen und die zu sendenden Prozessdaten die ständig aktualisiert werden.
sysPara	ST_IEC870_5_104SS ysPara (STRUCT)	Konfiguration der Systemparameter der Unterstation und der implementierten Dienste.
fbLog	FB_IEC870_DebugLo gFifo (FB)	Ein einfacher Logpuffer für Debugzwecke/ Fehlersuche.

### 6.5.2.2.3 Szenarios

#### Grundsätzliche Erläuterungen zu den Low Level Szenarios:

Der Initialisierungsschritt beginnt mit der Konfiguration der TCP/IP-Schnittstelle, wobei IPAdresse und Kommunikationsport festgelegt werden. Anschließend werden verschiedene Protokollparameter, sowie Parameter der Sende- beziehungsweise Empfangspuffer eingestellt. Danach wird die TCP/IP-Verbindung mit Hilfe von Funktionsbausteinen aus der TCP/IP-Funktion hergestellt.

Daraufhin erfolgt das Empfangen der Daten, welche solange mit der Aktion **fbBuffer.RxRemoveObj** aus dem Empfangspuffer geladen werden, bis dieser leer ist. Die empfangenen Daten werden nach Typkennung und Objektadresse sortiert. Nach der Sortierung werden die Entsprechenden Daten mit der Funktion **F\_iecCopyStreamToBuffer** in eine Variable gespeichert und anschließend entsprechend der in der Norm festgelegten Bestandteile aufgeteilt. Wird im Slave der Befehl für eine Generalabfrage oder der Abfragebefehl für eine bestimmte Abfragegruppe empfangen, so wird sofort die entsprechende Übertragungsprozedur für Abfragebefehle angestoßen und die für die jeweilige Gruppe definierten Datenpunkte in den Sendepuffer geschrieben. Das Senden der Daten wird im nachfolgenden Absatz genauer beschrieben.

Anschließend erfolgt sowohl beim Master als auch beim Slave das Senden der Daten, die außerhalb oder zusätzlich zu einer Abfragegruppe übermittelt werden sollen. Dabei werden zuerst gemeinsame Sendeinformationen der Datenpunkte wie ASDU Adresse, Anzahl der zu übertragenden Objekte, Grund der Datenübertragung und Prioritätsklasse in die Struktur **stTxAsdu.ident** geschrieben. Anschließend werden die Protokollframes der einzelnen Datenpunkte, welche aus Typkennung, Objektadresse und objektspezifischen Daten bestehen in die Struktur **stTxAsdu** geschrieben. Anschließend wird diese mit der Funktion **F\_iecCopyBufferToStream** umkopiert und der Aktion **fbBuffer.TxAddObj** in den Sendepuffer geschrieben. Abschließend werden mögliche Fehler aus dem Fehlerspeicher ausgelesen und in einer Fehlerausgabe gelistet.

Der Ablauf des Lesens, Schreibens und das anschließende Fehlerhandling wiederholen sich zyklisch.

#### 6.5.2.2.3.1 Spontane Datenübertragung und Generalabfrage

Das hier vorgestellte Beispielprogramm zeigt einen IEC 60870-5-104 Slave mit Low Level Schnittstelle und einer einfachen Kommunikation von zwei Datenpunkten. Ein Datenpunkt wird in Control und ein anderer in Monitor Direction jeweils spontan übertragen. Für den Datenpunkt in Monitor Direction ist zusätzlich eine Generalabfrage konfiguriert.

Download TwinCAT XAE Project (\*.zip): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/1244578443.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/1244578443.zip)

Im Beispiel sind folgende Parameter für die Unterstation (Slave) eingestellt:

- Server Host-Adresse (IP-Adresse des Slaves): 127.0.0.1 (Mindestens dieser Parameter muss individuell angepasst werden)
- Server Port-Adresse (Port-Adresse des Slaves): 2404
- k: 12
- w: 8
- t0: 30s
- t1: 15s
- t2: 10s
- t3: 20s
- Allgemeine ASDU-Adresse: 3
- Cause of transfer size: 2 octets
- Common ASDU address size: 2 octets
- Information object address size: 3 octets
- Max. APDU length: 253



- Es wird kein Anspruch auf Vollständigkeit und Richtigkeit übernommen. Stellen Sie die IP-Adresse des Slaves sowohl im Master- als auch im Slave-Teil ein! Überprüfen Sie, dass der Port 2404 nicht von einer Firewall blockiert wird!
  - Die Verbindungsherstellung beginnt nach dem Starten des Programms. Die Datenpunkte werden in der Globalen Variablenliste (GVL) verwaltet.
- 

#### 6.5.2.2.3.2 Generalabfrage und Gruppen

Das hier vorgestellte Beispielprogramm zeigt einen IEC 60870-5-104 Slave mit Low Level Schnittstelle. Das Beispiel geht darauf ein Gruppen von Datentypen zu definieren. Es wird eine Generalabfrage und eine gesonderte Gruppe 1 konfiguriert. Die Generalabfrage erfolgt alle 30s und die Gruppe 1 wird alle 10s abgefragt. Es werden unter anderem Single Points, Bitstrings und Measured Values kommuniziert.

Download TwinCAT XAE Project (\*.zip): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resourcen/1244595851.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resourcen/1244595851.zip)

Im Beispiel sind folgende Parameter für die Unterstation (Slave) eingestellt:

- Server Host-Adresse (IP-Adresse des Slaves): 127.0.0.1 (Mindestens dieser Parameter muss individuell angepasst werden)
- Server Port-Adresse (Port-Adresse des Slaves): 2404
- k: 12
- w: 8
- t0: 30s
- t1: 15s
- t2: 10s
- t3: 20s
- Allgemeine ASDU-Adresse: 3
- Cause of transfer size: 2 octets
- Common ASDU address size: 2 octets
- Information object address size: 3 octets
- Max. APDU length: 253



- Es wird kein Anspruch auf Vollständigkeit und Richtigkeit übernommen. Stellen Sie die IP-Adresse des Slaves sowohl im Master- als auch im Slave-Teil ein! Überprüfen Sie, dass der Port 2404 nicht von einer Firewall blockiert wird!
  - Die Verbindungsherstellung beginnt nach dem Starten des Programms. Die Datenpunkte werden in der Globalen Variablenliste (GVL) verwaltet.
-

### 6.5.2.2.3.3 Zählwertabfrage und Zeitsynchronisation

Das hier vorgestellte Beispielprogramm zeigt einen IEC 60870-5-104 Slave mit Low Level Schnittstelle. Kern des Beispiels ist die Uhrzeit-Synchronisation von Master zu Slave und die Zählwertabfrage. Die Synchronisation der Uhrzeit findet hier beispielhaft alle 120s statt. Ein Datentyp M\_IT\_NA\_1 sorgt für die Zählwert-Übertragung durch eine General-Zähler-Abfrage.

Download TwinCAT XAE Project (\*.zip): [https://infosys.beckhoff.com/content/1031/TF6500\\_TC3\\_IEC60870\\_5\\_10x/Resources/1244600459.zip](https://infosys.beckhoff.com/content/1031/TF6500_TC3_IEC60870_5_10x/Resources/1244600459.zip)

Im Beispiel sind folgende Parameter für die Unterstation (Slave) eingestellt:

- Server Host-Adresse (IP-Adresse des Slaves): 127.0.0.1 (Mindestens dieser Parameter muss individuell angepasst werden)
- Server Port-Adresse (Port-Adresse des Slaves): 2404
- k: 12
- w: 8
- t0: 30s
- t1: 15s
- t2: 10s
- t3: 20s
- Allgemeine ASDU-Adresse: 3
- Cause of transfer size: 2 octets
- Common ASDU address size: 2 octets
- Information object address size: 3 octets
- Max. APDU length: 253



- Es wird kein Anspruch auf Vollständigkeit und Richtigkeit übernommen. Stellen Sie die IP-Adresse des Slaves sowohl im Master- als auch im Slave-Teil ein! Überprüfen Sie, dass der Port 2404 nicht von einer Firewall blockiert wird!
  - Die Verbindungsherstellung beginnt nach dem Starten des Programms. Die Datenpunkte werden in der Globalen Variablenliste (GVL) verwaltet.
-

## 7 Anhang

### 7.1 IEC 60870-5-101

#### 7.1.1 TwinCAT IEC 60870-5-101 Fehlercodes

##### Voraussetzungen

Fehlercodes (Hex)	Fehlercodes (Dez)	Fehlerquelle	Beschreibung
0x00000000-0x00002000	0-8192	TF6340 Serial-Communication Fehlercodes	TF6340 Serial-Communication Fehler (aber nur wenn die Fehlerquelle [▶ 508] = eIEC870_ESRC_IEC60870_5_101LINK ).
0x00000000-0x00007800	0-30720	TwinCAT System Fehlercodes [▶ 685]	TwinCAT System Fehler (ADS-Fehlercodes inklusive wenn Fehlerquelle [▶ 508] <> eIEC870_ESRC_IEC60870_5_101LINK ).
0x00008100-0x000081FF	32768-33023	TwinCAT IEC 60870-5-10x Fehlercodes [▶ 557]	TwinCAT IEC 60870-5-10x Protokollfehler/ Kommunikationsfehler (z.B.: Timeoutfehler, Telegrammfehler, Konfigurationsfehler).

##### 7.1.1.1 ADS Return Codes

Gruppierung der Fehlercodes:

Globale Fehlercodes: 0x0000 [▶ 685]... (0x9811\_0000 ...)

Router Fehlercodes: 0x0500 [▶ 686]... (0x9811\_0500 ...)

Allgemeine ADS Fehler: 0x0700 [▶ 687]... (0x9811\_0700 ...)

RTime Fehlercodes: 0x1000 [▶ 689]... (0x9811\_1000 ...)

##### Globale Fehlercodes

Hex	Dec	HRESULT	Name	Beschreibung
0x0	0	0x98110000	ERR_NOERROR	Kein Fehler.
0x1	1	0x98110001	ERR_INTERNAL	Interner Fehler.
0x2	2	0x98110002	ERR_NORTIME	Keine Echtzeit.
0x3	3	0x98110003	ERR_ALLOCLOCKEDMEM	Zuweisung gesperrt - Speicherfehler.
0x4	4	0x98110004	ERR_INSERTMAILBOX	Postfach voll – Es konnte die ADS Nachricht nicht versendet werden. Reduzieren der Anzahl der ADS Nachrichten pro Zyklus bringt Abhilfe.
0x5	5	0x98110005	ERR_WRONGRECEIVEHMSG	Falsches HMSG.
0x6	6	0x98110006	ERR_TARGETPORTNOTFOUND	Ziel-Port nicht gefunden – ADS Server ist nicht gestartet, nicht erreichbar oder nicht installiert.
0x7	7	0x98110007	ERR_TARGETMACHINENOTFOUND	Zielrechner nicht gefunden – AMS Route wurde nicht gefunden.
0x8	8	0x98110008	ERR_UNKNOWNCMDID	Unbekannte Befehl-ID.
0x9	9	0x98110009	ERR_BADTASKID	Ungültige Task-ID.
0xA	10	0x9811000A	ERR_NOIO	Kein IO.
0xB	11	0x9811000B	ERR_UNKNOWNAMSCMD	Unbekannter AMS-Befehl.
0xC	12	0x9811000C	ERR_WIN32ERROR	Win32 Fehler.
0xD	13	0x9811000D	ERR_PORTNOTCONNECTED	Port nicht verbunden.
0xE	14	0x9811000E	ERR_INVALIDAMSLENGTH	Ungültige AMS-Länge.
0xF	15	0x9811000F	ERR_INVALIDAMSNETID	Ungültige AMS Net ID.
0x10	16	0x98110010	ERR_LOWINSTLEVEL	Installations-Level ist zu niedrig –TwinCAT 2 Lizenzfehler.
0x11	17	0x98110011	ERR_NODEBUGINTAVAILABLE	Kein Debugging verfügbar.
0x12	18	0x98110012	ERR_PORTDISABLED	Port deaktiviert – TwinCAT System Service nicht gestartet.
0x13	19	0x98110013	ERR_PORTALREADYCONNECTED	Port bereits verbunden.
0x14	20	0x98110014	ERR_AMSSYNC_W32ERROR	AMS Sync Win32 Fehler.
0x15	21	0x98110015	ERR_AMSSYNC_TIMEOUT	AMS Sync Timeout.
0x16	22	0x98110016	ERR_AMSSYNC_AMSERROR	AMS Sync Fehler.
0x17	23	0x98110017	ERR_AMSSYNC_NOINDEXINMAP	Keine Index-Map für AMS Sync vorhanden.
0x18	24	0x98110018	ERR_INVALIDAMSPORT	Ungültiger AMS-Port.
0x19	25	0x98110019	ERR_NOMEMORY	Kein Speicher.
0x1A	26	0x9811001A	ERR_TCPSEND	TCP Sendefehler.
0x1B	27	0x9811001B	ERR_HOSTUNREACHABLE	Host nicht erreichbar.
0x1C	28	0x9811001C	ERR_INVALIDAMSFRAGMENT	Ungültiges AMS Fragment.
0x1D	29	0x9811001D	ERR_TLSEND	TLS Sendefehler – Secure ADS Verbindung fehlgeschlagen.
0x1E	30	0x9811001E	ERR_ACCESSDENIED	Zugriff Verweigert – Secure ADS Zugriff verweigert.

## Router Fehlercodes

Hex	Dec	HRESULT	Name	Beschreibung
0x500	1280	0x98110500	ROUTERERR_NOLOCKEDMEMORY	Lockierter Speicher kann nicht zugewiesen werden.
0x501	1281	0x98110501	ROUTERERR_RESIZEMEMORY	Die Größe des Routerspeichers konnte nicht geändert werden.
0x502	1282	0x98110502	ROUTERERR_MAILBOXFULL	Das Postfach hat die maximale Anzahl der möglichen Meldungen erreicht.
0x503	1283	0x98110503	ROUTERERR_DEBUGBOXFULL	Das Debug Postfach hat die maximale Anzahl der möglichen Meldungen erreicht.
0x504	1284	0x98110504	ROUTERERR_UNKNOWNPORTTYPE	Der Porttyp ist unbekannt.
0x505	1285	0x98110505	ROUTERERR_NOTINITIALIZED	Router ist nicht initialisiert.
0x506	1286	0x98110506	ROUTERERR_PORTALREADYINUSE	Die Portnummer ist bereits vergeben.
0x507	1287	0x98110507	ROUTERERR_NOTREGISTERED	Der Port ist nicht registriert.
0x508	1288	0x98110508	ROUTERERR_NOMOREQUEUES	Die maximale Portanzahl ist erreicht.
0x509	1289	0x98110509	ROUTERERR_INVALIDPORT	Der Port ist ungültig.
0x50A	1290	0x9811050A	ROUTERERR_NOTACTIVATED	Der Router ist nicht aktiv.
0x50B	1291	0x9811050B	ROUTERERR_FRAGMENTBOXFULL	Das Postfach hat die maximale Anzahl für fragmentierte Nachrichten erreicht.
0x50C	1292	0x9811050C	ROUTERERR_FRAGMENTTIMEOUT	Fragment Timeout aufgetreten.
0x50D	1293	0x9811050D	ROUTERERR_TOBEREMOVED	Port wird entfernt.

**Allgemeine ADS Fehlercodes**

Hex	Dec	HRESULT	Name	Beschreibung
0x700	1792	0x98110700	ADSERR_DEVICE_ERROR	Allgemeiner Gerätefehler.
0x701	1793	0x98110701	ADSERR_DEVICE_SRVNOTSUPP	Service wird vom Server nicht unterstützt.
0x702	1794	0x98110702	ADSERR_DEVICE_INVALIDGRP	Ungültige Index-Gruppe.
0x703	1795	0x98110703	ADSERR_DEVICE_INVALIDOFFSET	Ungültiger Index-Offset.
0x704	1796	0x98110704	ADSERR_DEVICE_INVALIDACCESS	Lesen oder Schreiben nicht gestattet.
0x705	1797	0x98110705	ADSERR_DEVICE_INVALIDSIZE	Parametergröße nicht korrekt.
0x706	1798	0x98110706	ADSERR_DEVICE_INVALIDDATA	Ungültige Daten-Werte.
0x707	1799	0x98110707	ADSERR_DEVICE_NOTREADY	Gerät nicht betriebsbereit.
0x708	1800	0x98110708	ADSERR_DEVICE_BUSY	Gerät beschäftigt.
0x709	1801	0x98110709	ADSERR_DEVICE_INVALIDCONTEXT	Ungültiger Kontext vom Betriebssystem - Kann durch Verwendung von ADS Bausteinen in unterschiedlichen Tasks auftreten. Abhilfe kann die Multitasking-Synchronisation in der SPS geben.
0x70A	1802	0x9811070A	ADSERR_DEVICE_NOMEMORY	Nicht genügend Speicher.
0x70B	1803	0x9811070B	ADSERR_DEVICE_INVALIDPARM	Ungültige Parameter-Werte.
0x70C	1804	0x9811070C	ADSERR_DEVICE_NOTFOUND	Nicht gefunden (Dateien,...).
0x70D	1805	0x9811070D	ADSERR_DEVICE_SYNTAX	Syntax-Fehler in Datei oder Befehl.
0x70E	1806	0x9811070E	ADSERR_DEVICE_INCOMPATIBLE	Objekte stimmen nicht überein.
0x70F	1807	0x9811070F	ADSERR_DEVICE_EXISTS	Objekt ist bereits vorhanden.
0x710	1808	0x98110710	ADSERR_DEVICE_SYMBOLNOTFOUND	Symbol nicht gefunden.
0x711	1809	0x98110711	ADSERR_DEVICE_SYMBOLVERSIONINVALID	Symbol-Version ungültig – Kann durch einen Online-Change auftreten. Erzeuge einen neuen Handle.
0x712	1810	0x98110712	ADSERR_DEVICE_INVALIDSTATE	Gerät (Server) ist im ungültigen Zustand.
0x713	1811	0x98110713	ADSERR_DEVICE_TRANSMODENOTSUPP	AdsTransMode nicht unterstützt.
0x714	1812	0x98110714	ADSERR_DEVICE_NOTIFYHANDINVALID	Notification Handle ist ungültig.
0x715	1813	0x98110715	ADSERR_DEVICE_CLIENTUNKNOWN	Notification-Client nicht registriert.
0x716	1814	0x98110716	ADSERR_DEVICE_NOMOREHDL	Keine weiteren Handles verfügbar.
0x717	1815	0x98110717	ADSERR_DEVICE_INVALIDWATCHSIZE	Größe der Notification zu groß.
0x718	1816	0x98110718	ADSERR_DEVICE_NOTINIT	Gerät nicht initialisiert.
0x719	1817	0x98110719	ADSERR_DEVICE_TIMEOUT	Gerät hat einen Timeout.
0x71A	1818	0x9811071A	ADSERR_DEVICE_NOINTERFACE	Interface Abfrage fehlgeschlagen.
0x71B	1819	0x9811071B	ADSERR_DEVICE_INVALIDINTERFACE	Falsches Interface angefordert.
0x71C	1820	0x9811071C	ADSERR_DEVICE_INVALIDCLSID	Class-ID ist ungültig.
0x71D	1821	0x9811071D	ADSERR_DEVICE_INVALIDOBJID	Object-ID ist ungültig.
0x71E	1822	0x9811071E	ADSERR_DEVICE_PENDING	Anforderung steht aus.
0x71F	1823	0x9811071F	ADSERR_DEVICE_ABORTED	Anforderung wird abgebrochen.
0x720	1824	0x98110720	ADSERR_DEVICE_WARNING	Signal-Warnung.
0x721	1825	0x98110721	ADSERR_DEVICE_INVALIDARRAYIDX	Ungültiger Array-Index.
0x722	1826	0x98110722	ADSERR_DEVICE_SYMBOLNOTACTIVE	Symbol nicht aktiv.
0x723	1827	0x98110723	ADSERR_DEVICE_ACCESSDENIED	Zugriff verweigert.
0x724	1828	0x98110724	ADSERR_DEVICE_LICENSENOTFOUND	Fehlende Lizenz.
0x725	1829	0x98110725	ADSERR_DEVICE_LICENSEEXPIRED	Lizenz abgelaufen.
0x726	1830	0x98110726	ADSERR_DEVICE_LICENSEEXCEEDED	Lizenz überschritten.
0x727	1831	0x98110727	ADSERR_DEVICE_LICENSEINVALID	Lizenz ungültig.
0x728	1832	0x98110728	ADSERR_DEVICE_LICENSESYSTEMID	Lizenzproblem: System-ID ist ungültig.
0x729	1833	0x98110729	ADSERR_DEVICE_LICENSENOTIMELIMIT	Lizenz nicht zeitlich begrenzt.
0x72A	1834	0x9811072A	ADSERR_DEVICE_LICENSEFUTUREISSUE	Lizenzproblem: Zeitpunkt in der Zukunft.
0x72B	1835	0x9811072B	ADSERR_DEVICE_LICENSETIMETOLONG	Lizenz-Zeitraum zu lang.
0x72C	1836	0x9811072C	ADSERR_DEVICE_EXCEPTION	Exception beim Systemstart.
0x72D	1837	0x9811072D	ADSERR_DEVICE_LICENSEDUPLICATED	Lizenz-Datei zweimal gelesen.
0x72E	1838	0x9811072E	ADSERR_DEVICE_SIGNATUREINVALID	Ungültige Signatur.
0x72F	1839	0x9811072F	ADSERR_DEVICE_CERTIFICATEINVALID	Zertifikat ungültig.
0x730	1840	0x98110730	ADSERR_DEVICE_LICENSEOEMNOTFOUND	Public Key vom OEM nicht bekannt.
0x731	1841	0x98110731	ADSERR_DEVICE_LICENSERESTRICTED	Lizenz nicht gültig für diese System.ID.
0x732	1842	0x98110732	ADSERR_DEVICE_LICENSEDEMODENIED	Demo-Lizenz untersagt.
0x733	1843	0x98110733	ADSERR_DEVICE_INVALIDFNCID	Funktions-ID ungültig.
0x734	1844	0x98110734	ADSERR_DEVICE_OUTOFRANGE	Außerhalb des gültigen Bereiches.
0x735	1845	0x98110735	ADSERR_DEVICE_INVALIDALIGNMENT	Ungültiges Alignment.



Hex	Dec	HRESULT	Name	Beschreibung
0x736	1846	0x98110736	ADSERR_DEVICE_LICENSEPLATFORM	Ungültiger Plattform Level.
0x737	1847	0x98110737	ADSERR_DEVICE_FORWARD_PL	Kontext – Weiterleitung zum Passiv-Level.
0x738	1848	0x98110738	ADSERR_DEVICE_FORWARD_DL	Kontext – Weiterleitung zum Dispatch-Level.
0x739	1849	0x98110739	ADSERR_DEVICE_FORWARD_RT	Kontext – Weiterleitung zur Echtzeit.
0x740	1856	0x98110740	ADSERR_CLIENT_ERROR	Clientfehler.
0x741	1857	0x98110741	ADSERR_CLIENT_INVALIDPARM	Dienst enthält einen ungültigen Parameter.
0x742	1858	0x98110742	ADSERR_CLIENT_LISTEMPTY	Polling-Liste ist leer.
0x743	1859	0x98110743	ADSERR_CLIENT_VARUSED	Var-Verbindung bereits im Einsatz.
0x744	1860	0x98110744	ADSERR_CLIENT_DUPLINVOKEID	Die aufgerufene ID ist bereits in Benutzung.
0x745	1861	0x98110745	ADSERR_CLIENT_SYNC TIMEOUT	Timeout ist aufgetreten – Die Gegenstelle antwortet nicht im vorgegebenen ADS Timeout. Die Routeneinstellung der Gegenstelle kann falsch konfiguriert sein.
0x746	1862	0x98110746	ADSERR_CLIENT_W32ERROR	Fehler im Win32 Subsystem.
0x747	1863	0x98110747	ADSERR_CLIENT_TIMEOUTINVALID	Ungültiger Client Timeout-Wert.
0x748	1864	0x98110748	ADSERR_CLIENT_PORTNOTOPEN	Port nicht geöffnet.
0x749	1865	0x98110749	ADSERR_CLIENT_NOAMSADDR	Keine AMS Adresse.
0x750	1872	0x98110750	ADSERR_CLIENT_SYNCINTERNAL	Interner Fehler in Ads-Sync.
0x751	1873	0x98110751	ADSERR_CLIENT_ADDHASH	Überlauf der Hash-Tabelle.
0x752	1874	0x98110752	ADSERR_CLIENT_REMOVEHASH	Schlüssel in der Tabelle nicht gefunden.
0x753	1875	0x98110753	ADSERR_CLIENT_NOMORESVM	Keine Symbole im Cache.
0x754	1876	0x98110754	ADSERR_CLIENT_SYNCRESINVALID	Ungültige Antwort erhalten.
0x755	1877	0x98110755	ADSERR_CLIENT_SYNCPORTLOCKED	Sync Port ist verriegelt.
0x756	1878	0x98110756	ADSERR_CLIENT_REQUESTCANCELLED	Die Anfrage wurde abgebrochen.

**RTime Fehlercodes**

Hex	Dec	HRESULT	Name	Beschreibung
0x1000	4096	0x98111000	RTERR_INTERNAL	Interner Fehler im Echtzeit-System.
0x1001	4097	0x98111001	RTERR_BADTIMERPERIODS	Timer-Wert nicht gültig.
0x1002	4098	0x98111002	RTERR_INVALIDTASKPTR	Task-Pointer hat den ungültigen Wert 0 (null).
0x1003	4099	0x98111003	RTERR_INVALIDSTACKPTR	Stack-Pointer hat den ungültigen Wert 0 (null).
0x1004	4100	0x98111004	RTERR_PrioEXISTS	Die Request Task Priority ist bereits vergeben.
0x1005	4101	0x98111005	RTERR_NOMORETCB	Kein freier TCB (Task Control Block) verfügbar. Maximale Anzahl von TCBs beträgt 64.
0x1006	4102	0x98111006	RTERR_NOMORESEMAS	Keine freien Semaphoren zur Verfügung. Maximale Anzahl der Semaphoren beträgt 64.
0x1007	4103	0x98111007	RTERR_NOMOREQUEUES	Kein freier Platz in der Warteschlange zur Verfügung. Maximale Anzahl der Plätze in der Warteschlange beträgt 64.
0x100D	4109	0x9811100D	RTERR_EXTIRQALREADYDEF	Ein externer Synchronisations-Interrupt wird bereits angewandt.
0x100E	4110	0x9811100E	RTERR_EXTIRQNOTDEF	Kein externer Sync-Interrupt angewandt.
0x100F	4111	0x9811100F	RTERR_EXTIRQINSTALLFAILED	Anwendung des externen Synchronisierungs-Interrupts ist fehlgeschlagen.
0x1010	4112	0x98111010	RTERR_IRQNOTLESSOREQUAL	Aufruf einer Service-Funktion im falschen Kontext
0x1017	4119	0x98111017	RTERR_VMXNOTSUPPORTED	Intel VT-x Erweiterung wird nicht unterstützt.
0x1018	4120	0x98111018	RTERR_VMXDISABLED	Intel VT-x Erweiterung ist nicht aktiviert im BIOS.
0x1019	4121	0x98111019	RTERR_VMXCONTROLSMISSING	Fehlende Funktion in Intel VT-x Erweiterung.
0x101A	4122	0x9811101A	RTERR_VMXENABLEFAILS	Aktivieren von Intel VT-x schlägt fehl.

**Spezifische positive HRESULT Return Codes:**

HRESULT	Name	Beschreibung
0x0000_0000	S_OK	Kein Fehler.
0x0000_0001	S_FALSE	Kein Fehler. Bsp.: erfolgreiche Abarbeitung, bei der jedoch ein negatives oder unvollständiges Ergebnis erzielt wurde.
0x0000_0203	S_PENDING	Kein Fehler. Bsp.: erfolgreiche Abarbeitung, bei der jedoch noch kein Ergebnis vorliegt.
0x0000_0256	S_WATCHDOG_TIMEOUT	Kein Fehler. Bsp.: erfolgreiche Abarbeitung, bei der jedoch eine Zeitüberschreitung eintrat.

### TCP Winsock-Fehlercodes

Hex	Dec	Name	Beschreibung
0x274C	10060	WSAETIMEDOUT	Verbindungs Timeout aufgetreten - Fehler beim Herstellen der Verbindung, da die Gegenstelle nach einer bestimmten Zeitspanne nicht ordnungsgemäß reagiert hat, oder die hergestellte Verbindung konnte nicht aufrecht erhalten werden, da der verbundene Host nicht reagiert hat.
0x274D	10061	WSAECONNREFUSED	Verbindung abgelehnt - Es konnte keine Verbindung hergestellt werden, da der Zielcomputer dies explizit abgelehnt hat. Dieser Fehler resultiert normalerweise aus dem Versuch, eine Verbindung mit einem Dienst herzustellen, der auf dem fremden Host inaktiv ist—das heißt, einem Dienst, für den keine Serveranwendung ausgeführt wird.
0x2751	10065	WSAEHOSTUNREACH	Keine Route zum Host - Ein Socketvorgang bezog sich auf einen nicht verfügbaren Host.

Weitere Winsock-Fehlercodes: [Win32-Fehlercodes](#) [▶ 713]

## 7.1.2 Fehlersuche/Diagnose

### Allgemeine Überprüfungen:

- Überprüfen Sie die in dieser Dokumentation beschriebenen Hardware- und Softwareanforderungen (TwinCAT-Version, CE Image-Version usw.);
- Überprüfen Sie die IO-Konfiguration und das Mapping der SPS-Variablen in TwinCAT System Manager ([Konfiguration der seriellen Schnittstellen](#) [▶ 561], Baudrate, Parity, Stopbits usw.). Vergleichen Sie die Parameter mit den Parametern des Kommunikationspartners;
- Überprüfen Sie die Konfiguration der Datenpunkte (Typ, Adresse des Informationsobjektes usw.);
- Überprüfen Sie ob der andere Kommunikationspartner einen Fehlercode ausgibt;
- Aktivieren Sie die [Debugausgaben](#) [▶ 691] beim Aufbauen und Abbauen der Verbindung und/oder der ASDU Daten. Öffnen Sie den TwinCAT System Manager und aktivieren das LogView-Fenster. Prüfen Sie die Debugausgaben;

### High-Level Interface:

- Speziell bei der Zentralstation:
  - Überprüfen Sie ob der [FB IEC870\\_5\\_101Master](#) [▶ 385]-Funktionsbaustein einen [Fehlercode/Fehlerquelle](#) [▶ 685] ausgibt;
  - Vergleichen/überprüfen Sie die [Kompatibilitätsliste der Zentralstation](#) [▶ 14] mit der Kompatibilitätsliste der Unterstation;
  - Überprüfen Sie die am [FB\\_IEC870\\_5\\_101Master](#)-Funktionsbaustein eingestellten [Protokollparameter](#) [▶ 538] (Link-Adresse, Länge der Link-Adresse, FRAMELength, usw.). Vergleichen Sie die Protokollparameter mit den Parametern in der Unterstation;
  - Überprüfen Sie die am [FB\\_IEC870\\_5\\_101Master](#)-Funktionsbaustein eingestellten [Systemparameter](#) [▶ 487] (ASDU-Adresse, Länge der ASDU-Adresse, Länge der Informationsobjekt-Adresse, Länge der Übertragungsursache COT, usw. ). Vergleichen Sie die Systemparameter mit den Parametern in der Unterstation;
  - Überprüfen Sie die am [FB\\_IEC870\\_5\\_101Master](#)-Funktionsbaustein eingestellten Parameter für die [zyklische Datenerfassung](#) [▶ 491] (Akquisition) (Initialisierungssequenz, zyklische Generalabfrage, zyklische Zählerabfrage, zyklische Testkommandos, usw. );
- Speziell bei der Unterstation:

- Überprüfen Sie ob der [FB IEC870\\_5\\_101Slave \[▶ 387\]](#)-Funktionsbaustein einen [Fehlercode/ Fehlerquelle \[▶ 685\]](#) ausgibt;
- Überprüfen Sie die [Kompatibilitätsliste der Unterstation \[▶ 32\]](#) mit der Kompatibilitätsliste der Zentralstation;
- Überprüfen Sie die am [FB\\_IEC870\\_5\\_101Slave](#)-Funktionsbaustein eingestellten [Protokollparameter \[▶ 538\]](#) (Link-Adresse, Länge der Link-Adresse, FRAMELength, usw.). Vergleichen Sie die Protokollparameter mit den Parametern in der Zentralstation;
- Überprüfen Sie die am [FB\\_IEC870\\_5\\_101Slave](#)-Funktionsbaustein eingestellten [Systemparameter \[▶ 487\]](#) (ASDU-Adresse, Länge der ASDU-Adresse, Länge der Informationsobjekt-Adresse, Länge der Übertragungsursache COT, usw. ). Vergleichen Sie die Systemparameter mit den Parametern in der Leitstation;

#### Low-Level Interface:

- Überprüfen Sie ob der [FB\\_IEC870\\_5\\_101TProtocol \[▶ 407\]](#)-Funktionsbaustein einen [Fehlercode \[▶ 685\]](#) ausgibt;
- Vergleichen/überprüfen Sie die Kompatibilitätsliste der Zentralstation mit der Kompatibilitätsliste der Unterstation;
- Überprüfen Sie die am [FB\\_IEC870\\_5\\_101TProtocol](#)-Funktionsbaustein eingestellten [Protokollparameter \[▶ 538\]](#) (Link-Adresse, Länge der Link-Adresse, FRAMELength, usw.). Vergleichen Sie die Protokollparameter mit den Parametern des Kommunikationspartners;
- Überprüfen Sie die am TX/RX-Datenpuffer (Instanz von [ST\\_IEC870\\_5\\_101TBuffer \[▶ 481\]](#)) konfigurierten Adresslängen: Länge der ASDU-Adresse, Länge der Informationsobjekt-Adresse, Länge der Übertragungsursache, max. ASDU-Länge. Vergleichen Sie die Adresslängen mit der Konfiguration des Kommunikationspartners;

### 7.1.3 Debugmeldungen in TwinCAT->Error List

Debugmeldungen, die in TwinCAT XAE->Error List geschrieben werden erleichtern eine Fehlersuche im System. Um die aktivierten Debugmeldungen zu sehen starten Sie TwinCAT XAE und wählen im Menue: View->Other Windows->Error List. Die nachfolgende Grafik zeigt eine Debugausgabe. Die unterschiedlichen Typen der Meldungen wurden mit Zahlen markiert.

Error List		
<span>0 Errors</span> <span>0 Warnings</span> <span>1866 Messages</span> <span>Clear</span>		
	Description	
1	29.10.2013 08:50:33 829 ms   'LowSpeed' (351): Tc2_IEC60870_5_10x.Library::FB_IEC870_5_101Master[asduAddr = 7]::INIT => COLDSTART	3
2	29.10.2013 08:50:33 839 ms   'LowSpeed' (351): Tc2_IEC60870_5_10x.Library::FB_IEC870_5_101Master[asduAddr = 7]::COLDSTART => WARMSTART	
3	29.10.2013 08:50:33 849 ms   'LowSpeed' (351): Tc2_IEC60870_5_10x.Library::FB_IEC870_5_101Master[asduAddr = 7]::WARMSTART => SETTIME START	
4	29.10.2013 08:50:33 939 ms   'LowSpeed' (351): Tc2_IEC60870_5_10x.Library::FB_IEC870_5_101Master[asduAddr = 7]::SETTIME WAIT => ACTIVE OPEN	
5	29.10.2013 08:50:34 9 ms   'LowSpeed' (351): Serial Link[1]PRM.TX<=10 49 DC 00 25 16	2
6	29.10.2013 08:50:45 59 ms   'LowSpeed' (351): Serial Link[1]PRM.TX<=10 49 DC 00 25 16	
7	29.10.2013 08:50:45 119 ms   'LowSpeed' (351): Serial Link[1]PRM.RX=>10 0B DC 00 E7 16	
8	29.10.2013 08:50:45 159 ms   'LowSpeed' (351): Serial Link[1]PRM.TX<=10 40 DC 00 1C 16	
9	29.10.2013 08:50:45 219 ms   'LowSpeed' (351): Serial Link[1]PRM.RX=>10 00 DC 00 DC 16	
10	29.10.2013 08:50:45 229 ms   'LowSpeed' (351): Tc2_IEC60870_5_10x.Library::Serial Link[1]::ACTIVE OPEN => ESTABLISHED	3
11	29.10.2013 08:50:45 249 ms   'LowSpeed' (351): Serial Link[1]PRM.TX<=10 7B DC 00 57 16	2
12	29.10.2013 08:50:45 319 ms   'LowSpeed' (351): Serial Link[1]PRM.RX=>68 0D 0D 68 08 DC 00 46 01 04 00 07 00 00 00 00 00 36 16	
13	29.10.2013 08:50:45 329 ms   'LowSpeed' (351): RX FIFO<=46 01 04 00 07 00 00 00 00 00	1
14	29.10.2013 08:50:45 329 ms   'LowSpeed' (351): RX FIFO=>46 01 04 00 07 00 00 00 00 00	
15	29.10.2013 08:50:45 339 ms   'LowSpeed' (351): Class1 TX FIFO<=68 01 06 01 07 00 00 00 00 AA 55	
16	29.10.2013 08:50:45 359 ms   'LowSpeed' (351): Serial Link[1]PRM.TX<=68 0E 0E 68 53 DC 00 68 01 06 01 07 00 00 00 00 AA 55 A5 16	
17	29.10.2013 08:50:45 429 ms   'LowSpeed' (351): Serial Link[1]PRM.RX=>10 00 DC 00 DC 16	
18	29.10.2013 08:50:45 439 ms   'LowSpeed' (351): Class1 TX FIFO=>68 01 06 01 07 00 00 00 00 AA 55	1
19	29.10.2013 08:50:45 669 ms   'LowSpeed' (351): Serial Link[1]PRM.TX<=10 7B DC 00 57 16	2
20	29.10.2013 08:50:45 789 ms   'LowSpeed' (351): Serial Link[1]PRM.RX=>68 0E 0E 68 28 DC 00 68 01 07 01 07 00 00 00 00 AA 55 7B 16	
21	29.10.2013 08:50:45 749 ms   'LowSpeed' (351): RX FIFO<=68 01 07 01 07 00 00 00 00 AA 55	1
22	29.10.2013 08:50:45 749 ms   'LowSpeed' (351): RX FIFO=>68 01 07 01 07 00 00 00 00 AA 55	

Error List		
<span>2 Errors</span> <span>0 Warnings</span> <span>57 Messages</span> <span>Clear</span>		
	Description	
48	29.10.2013 08:58:13 449 ms   'LowSpeed' (351): Serial Link[1]PRM.RX=>10 09 DC 00 E5 16	2
49	29.10.2013 08:58:13 689 ms   'LowSpeed' (351): Serial Link[1]PRM.TX<=10 7B DC 00 57 16	
50	29.10.2013 08:58:15 709 ms   'LowSpeed' (351): Serial Link[1]PRM.TX<=10 7B DC 00 57 16	
51	29.10.2013 08:58:17 729 ms   'LowSpeed' (351): Serial Link[1]PRM.TX<=10 7B DC 00 57 16	
52	29.10.2013 08:58:19 749 ms   'LowSpeed' (351): Serial Link[1]PRM.TX<=10 7B DC 00 57 16	
53	29.10.2013 08:58:20 779 ms   'LowSpeed' (351): Tc2_IEC60870_5_10x.Library::Serial Link[1] IEC60870-5-101 protocol error: 0x0000810F	4
54	29.10.2013 08:58:20 779 ms   'LowSpeed' (351): Tc2_IEC60870_5_10x.Library::Serial Link[1]::ESTABLISHED => ACTIVE OPEN	3
55	29.10.2013 08:58:20 779 ms   'LowSpeed' (351): IEC60870-5-101 master error: 0x0000810F	5
56	29.10.2013 08:58:20 789 ms   'LowSpeed' (351): Tc2_IEC60870_5_10x.Library::Serial Link[1]::LOCAL RESET	3
57	29.10.2013 08:58:20 849 ms   'LowSpeed' (351): Serial Link[2]PRM.TX<=10 49 DC 00 25 16	2
58	29.10.2013 08:58:31 899 ms   'LowSpeed' (351): Serial Link[2]PRM.TX<=10 49 DC 00 25 16	
59	29.10.2013 08:58:42 949 ms   'LowSpeed' (351): Serial Link[2]PRM.TX<=10 49 DC 00 25 16	

Die Meldungen werden auf unterschiedliche Weise (durch unterschiedliche Parameter) aktiviert, je nach dem ob Sie die "Low-Level" oder "High-Level"-Schnittstelle verwenden. Unten folgt die Auflistung der unterschiedlichen Parameternamen.

1. Hexadezimale Ausgabe der ASDU's (ohne Link-Layer-Control-Header). Pro Zeile werden 32-ASDU-Datenbytes als hexadezimale Zahlen ausgegeben. Längere ASDU's werden auf mehrere Zeilen verteilt:

Name	AN	AUS	Unterstation		Zentralstation	
			Low-Level	High-Level	Low-Level	High-Level
ST IEC870 5 101TBuffer [▶ 481].eDbg	eIEC870_FIF O_DBG_ALL	eIEC870_FIF O_DBG_OFF	Ja	-	Ja	-
FB IEC870 5 101Slave [▶ 387].sysPa ra.dbgMode	IEC870_DEB UGMODE_A SDU	IEC870_DEB UGMODE_DI SABLED	-	Ja	-	-
FB IEC870 5 101Master [▶ 385].sysPa ra.dbgMode	IEC870_DEB UGMODE_A SDU	IEC870_DEB UGMODE_DI SABLED	-	-	-	Ja

2. Hexadezimale Ausgabe der APDU's (RS232/RS485 Telegramme). Pro Zeile werden 32-APDU-Datenbytes als hexadezimale Zahlen ausgegeben. Längere APDU's werden auf mehrere Zeilen verteilt:

Name	AN	AUS	Unterstation		Zentralstation	
			Low-Level	High-Level	Low-Level	High-Level
FB IEC870 5 101TProtocol [▶ 407].bOut Dbg	TRUE	FALSE	Ja	-	Ja	-
FB IEC870 5 101Slave [▶ 387].sysPa ra.dbgMode	IEC870_DEB UGMODE_LI NKLAYER	IEC870_DEB UGMODE_DI SABLED	-	Ja	-	-
FB IEC870 5 101Master [▶ 385].sysPa ra.dbgMode	IEC870_DEB UGMODE_LI NKLAYER	IEC870_DEB UGMODE_DI SABLED	-	-	-	Ja-

3. Stationsstatus-Meldungen:

Name	AN	AUS	Unterstation		Zentralstation	
			Low-Level	High-Level	Low-Level	High-Level
FB IEC870 5 101Slave [▶ 387].sysPa ra.dbgMode	IEC870_DEB UGMODE_D EVSTATE	IEC870_DEB UGMODE_DI SABLED	-	Ja	-	-
FB IEC870 5 101Master [▶ 385].sysPa ra.dbgMode	IEC870_DEB UGMODE_D EVSTATE	IEC870_DEB UGMODE_DI SABLED	-	-	-	Ja

4: Ausgabe der Link-Layer-Fehler als Meldungen:

Name	AN	AUS	Unterstation		Zentralstation	
			Low-Level	High-Level	Low-Level	High-Level
FB IEC870 5 101Slave [▶ 387].sysPara.dbgMode	IEC870_DEB UGMODE_LI NKERROR	IEC870_DEB UGMODE_DI SABLED	-	Ja	-	-
FB IEC870 5 101Master [▶ 385].sysPara.dbgMode	IEC870_DEB UGMODE_LI NKERROR	IEC870_DEB UGMODE_DI SABLED	-	-	-	Ja

5. Meldungen, die durch die SPS-Applikation (Beispielprojekt) mit Hilfe der ADSLOGSTR, ... Funktionen geloggt werden.

Weitere Diagnosetools:

- Portmon for Windows (v3.02, Windows Sysinternals);
- Diverse Protokoll-Test-Suit-Produkte;

## 7.2 IEC 60870-5-102

### 7.2.1 TwinCAT IEC 60870-5-102 Fehlercodes

#### Voraussetzungen

Fehlercodes (Hex)	Fehlercodes (Dez)	Fehlerquelle	Beschreibung
0x00000000-0x00002000	0-8192	TF6340 Serial-Communication Fehlercodes	TF6340 Serial-Communication Fehler (aber nur wenn die Fehlerquelle [▶ 508] = eIEC870_ESRC_IEC60870_5_101LINK ).
0x00000000-0x00007800	0-30720	TwinCAT System Fehlercodes [▶ 694]	TwinCAT System Fehler (ADS-Fehlercodes inklusive wenn Fehlerquelle [▶ 508] <> eIEC870_ESRC_IEC60870_5_101LINK ).
0x00008100-0x000081FF	32768-33023	TwinCAT IEC 60870-5-10x Fehlercodes [▶ 557]	TwinCAT IEC 60870-5-10x Protokollfehler/ Kommunikationsfehler (z.B.: Timeoutfehler, Telegrammfehler, Konfigurationsfehler).

#### 7.2.1.1 ADS Return Codes

Gruppierung der Fehlercodes:

Globale Fehlercodes: 0x0000 [▶ 694]... (0x9811\_0000 ...)

Router Fehlercodes: 0x0500 [▶ 695]... (0x9811\_0500 ...)

Allgemeine ADS Fehler: 0x0700 [▶ 696]... (0x9811\_0700 ...)

RTime Fehlercodes: 0x1000 [▶ 698]... (0x9811\_1000 ...)

#### Globale Fehlercodes

Hex	Dec	HRESULT	Name	Beschreibung
0x0	0	0x98110000	ERR_NOERROR	Kein Fehler.
0x1	1	0x98110001	ERR_INTERNAL	Interner Fehler.
0x2	2	0x98110002	ERR_NORTIME	Keine Echtzeit.
0x3	3	0x98110003	ERR_ALLOCLOCKEDMEM	Zuweisung gesperrt - Speicherfehler.
0x4	4	0x98110004	ERR_INSERTMAILBOX	Postfach voll – Es konnte die ADS Nachricht nicht versendet werden. Reduzieren der Anzahl der ADS Nachrichten pro Zyklus bringt Abhilfe.
0x5	5	0x98110005	ERR_WRONGRECEIVEHMSG	Falsches HMSG.
0x6	6	0x98110006	ERR_TARGETPORTNOTFOUND	Ziel-Port nicht gefunden – ADS Server ist nicht gestartet, nicht erreichbar oder nicht installiert.
0x7	7	0x98110007	ERR_TARGETMACHINENOTFOUND	Zielrechner nicht gefunden – AMS Route wurde nicht gefunden.
0x8	8	0x98110008	ERR_UNKNOWNCMDID	Unbekannte Befehl-ID.
0x9	9	0x98110009	ERR_BADTASKID	Ungültige Task-ID.
0xA	10	0x9811000A	ERR_NOIO	Kein IO.
0xB	11	0x9811000B	ERR_UNKNOWNAMSCMD	Unbekannter AMS-Befehl.
0xC	12	0x9811000C	ERR_WIN32ERROR	Win32 Fehler.
0xD	13	0x9811000D	ERR_PORTNOTCONNECTED	Port nicht verbunden.
0xE	14	0x9811000E	ERR_INVALIDAMSLENGTH	Ungültige AMS-Länge.
0xF	15	0x9811000F	ERR_INVALIDAMSNETID	Ungültige AMS Net ID.
0x10	16	0x98110010	ERR_LOWINSTLEVEL	Installations-Level ist zu niedrig –TwinCAT 2 Lizenzfehler.
0x11	17	0x98110011	ERR_NODEBUGINTAVAILABLE	Kein Debugging verfügbar.
0x12	18	0x98110012	ERR_PORTDISABLED	Port deaktiviert – TwinCAT System Service nicht gestartet.
0x13	19	0x98110013	ERR_PORTALREADYCONNECTED	Port bereits verbunden.
0x14	20	0x98110014	ERR_AMSSYNC_W32ERROR	AMS Sync Win32 Fehler.
0x15	21	0x98110015	ERR_AMSSYNC_TIMEOUT	AMS Sync Timeout.
0x16	22	0x98110016	ERR_AMSSYNC_AMSERROR	AMS Sync Fehler.
0x17	23	0x98110017	ERR_AMSSYNC_NOINDEXINMAP	Keine Index-Map für AMS Sync vorhanden.
0x18	24	0x98110018	ERR_INVALIDAMSPORT	Ungültiger AMS-Port.
0x19	25	0x98110019	ERR_NOMEMORY	Kein Speicher.
0x1A	26	0x9811001A	ERR_TCPSEND	TCP Sendefehler.
0x1B	27	0x9811001B	ERR_HOSTUNREACHABLE	Host nicht erreichbar.
0x1C	28	0x9811001C	ERR_INVALIDAMSFRAGMENT	Ungültiges AMS Fragment.
0x1D	29	0x9811001D	ERR_TLSEND	TLS Sendefehler – Secure ADS Verbindung fehlgeschlagen.
0x1E	30	0x9811001E	ERR_ACCESSDENIED	Zugriff Verweigert – Secure ADS Zugriff verweigert.

**Router Fehlercodes**

Hex	Dec	HRESULT	Name	Beschreibung
0x500	1280	0x98110500	ROUTERERR_NOLOCKEDMEMORY	Lockierter Speicher kann nicht zugewiesen werden.
0x501	1281	0x98110501	ROUTERERR_RESIZEMEMORY	Die Größe des Routerspeichers konnte nicht geändert werden.
0x502	1282	0x98110502	ROUTERERR_MAILBOXFULL	Das Postfach hat die maximale Anzahl der möglichen Meldungen erreicht.
0x503	1283	0x98110503	ROUTERERR_DEBUGBOXFULL	Das Debug Postfach hat die maximale Anzahl der möglichen Meldungen erreicht.
0x504	1284	0x98110504	ROUTERERR_UNKNOWNPORTTYPE	Der Porttyp ist unbekannt.
0x505	1285	0x98110505	ROUTERERR_NOTINITIALIZED	Router ist nicht initialisiert.
0x506	1286	0x98110506	ROUTERERR_PORTALREADYINUSE	Die Portnummer ist bereits vergeben.
0x507	1287	0x98110507	ROUTERERR_NOTREGISTERED	Der Port ist nicht registriert.
0x508	1288	0x98110508	ROUTERERR_NOMOREQUEUES	Die maximale Portanzahl ist erreicht.
0x509	1289	0x98110509	ROUTERERR_INVALIDPORT	Der Port ist ungültig.
0x50A	1290	0x9811050A	ROUTERERR_NOTACTIVATED	Der Router ist nicht aktiv.
0x50B	1291	0x9811050B	ROUTERERR_FRAGMENTBOXFULL	Das Postfach hat die maximale Anzahl für fragmentierte Nachrichten erreicht.
0x50C	1292	0x9811050C	ROUTERERR_FRAGMENTTIMEOUT	Fragment Timeout aufgetreten.
0x50D	1293	0x9811050D	ROUTERERR_TOBEREMOVED	Port wird entfernt.

### Allgemeine ADS Fehlercodes



Hex	Dec	HRESULT	Name	Beschreibung
0x700	1792	0x98110700	ADSERR_DEVICE_ERROR	Allgemeiner Gerätefehler.
0x701	1793	0x98110701	ADSERR_DEVICE_SRVNOTSUPP	Service wird vom Server nicht unterstützt.
0x702	1794	0x98110702	ADSERR_DEVICE_INVALIDGRP	Ungültige Index-Gruppe.
0x703	1795	0x98110703	ADSERR_DEVICE_INVALIDOFFSET	Ungültiger Index-Offset.
0x704	1796	0x98110704	ADSERR_DEVICE_INVALIDACCESS	Lesen oder Schreiben nicht gestattet.
0x705	1797	0x98110705	ADSERR_DEVICE_INVALIDSIZE	Parametergröße nicht korrekt.
0x706	1798	0x98110706	ADSERR_DEVICE_INVALIDDATA	Ungültige Daten-Werte.
0x707	1799	0x98110707	ADSERR_DEVICE_NOTREADY	Gerät nicht betriebsbereit.
0x708	1800	0x98110708	ADSERR_DEVICE_BUSY	Gerät beschäftigt.
0x709	1801	0x98110709	ADSERR_DEVICE_INVALIDCONTEXT	Ungültiger Kontext vom Betriebssystem - Kann durch Verwendung von ADS Bausteinen in unterschiedlichen Tasks auftreten. Abhilfe kann die Multitasking-Synchronisation in der SPS geben.
0x70A	1802	0x9811070A	ADSERR_DEVICE_NOMEMORY	Nicht genügend Speicher.
0x70B	1803	0x9811070B	ADSERR_DEVICE_INVALIDPARM	Ungültige Parameter-Werte.
0x70C	1804	0x9811070C	ADSERR_DEVICE_NOTFOUND	Nicht gefunden (Dateien,...).
0x70D	1805	0x9811070D	ADSERR_DEVICE_SYNTAX	Syntax-Fehler in Datei oder Befehl.
0x70E	1806	0x9811070E	ADSERR_DEVICE_INCOMPATIBLE	Objekte stimmen nicht überein.
0x70F	1807	0x9811070F	ADSERR_DEVICE_EXISTS	Objekt ist bereits vorhanden.
0x710	1808	0x98110710	ADSERR_DEVICE_SYMBOLNOTFOUND	Symbol nicht gefunden.
0x711	1809	0x98110711	ADSERR_DEVICE_SYMBOLVERSIONINVALID	Symbol-Version ungültig – Kann durch einen Online-Change auftreten. Erzeuge einen neuen Handle.
0x712	1810	0x98110712	ADSERR_DEVICE_INVALIDSTATE	Gerät (Server) ist im ungültigen Zustand.
0x713	1811	0x98110713	ADSERR_DEVICE_TRANSMODENOTSUPP	AdsTransMode nicht unterstützt.
0x714	1812	0x98110714	ADSERR_DEVICE_NOTIFYHANDINVALID	Notification Handle ist ungültig.
0x715	1813	0x98110715	ADSERR_DEVICE_CLIENTUNKNOWN	Notification-Client nicht registriert.
0x716	1814	0x98110716	ADSERR_DEVICE_NOMOREHDL	Keine weiteren Handles verfügbar.
0x717	1815	0x98110717	ADSERR_DEVICE_INVALIDWATCHSIZE	Größe der Notification zu groß.
0x718	1816	0x98110718	ADSERR_DEVICE_NOTINIT	Gerät nicht initialisiert.
0x719	1817	0x98110719	ADSERR_DEVICE_TIMEOUT	Gerät hat einen Timeout.
0x71A	1818	0x9811071A	ADSERR_DEVICE_NOINTERFACE	Interface Abfrage fehlgeschlagen.
0x71B	1819	0x9811071B	ADSERR_DEVICE_INVALIDINTERFACE	Falsches Interface angefordert.
0x71C	1820	0x9811071C	ADSERR_DEVICE_INVALIDCLSID	Class-ID ist ungültig.
0x71D	1821	0x9811071D	ADSERR_DEVICE_INVALIDOBJID	Object-ID ist ungültig.
0x71E	1822	0x9811071E	ADSERR_DEVICE_PENDING	Anforderung steht aus.
0x71F	1823	0x9811071F	ADSERR_DEVICE_ABORTED	Anforderung wird abgebrochen.
0x720	1824	0x98110720	ADSERR_DEVICE_WARNING	Signal-Warnung.
0x721	1825	0x98110721	ADSERR_DEVICE_INVALIDARRAYIDX	Ungültiger Array-Index.
0x722	1826	0x98110722	ADSERR_DEVICE_SYMBOLNOTACTIVE	Symbol nicht aktiv.
0x723	1827	0x98110723	ADSERR_DEVICE_ACCESSDENIED	Zugriff verweigert.
0x724	1828	0x98110724	ADSERR_DEVICE_LICENSENOTFOUND	Fehlende Lizenz.
0x725	1829	0x98110725	ADSERR_DEVICE_LICENSEEXPIRED	Lizenz abgelaufen.
0x726	1830	0x98110726	ADSERR_DEVICE_LICENSEEXCEEDED	Lizenz überschritten.
0x727	1831	0x98110727	ADSERR_DEVICE_LICENSEINVALID	Lizenz ungültig.
0x728	1832	0x98110728	ADSERR_DEVICE_LICENSESYSTEMID	Lizenzproblem: System-ID ist ungültig.
0x729	1833	0x98110729	ADSERR_DEVICE_LICENSENOTIMELIMIT	Lizenz nicht zeitlich begrenzt.
0x72A	1834	0x9811072A	ADSERR_DEVICE_LICENSEFUTUREISSUE	Lizenzproblem: Zeitpunkt in der Zukunft.
0x72B	1835	0x9811072B	ADSERR_DEVICE_LICENSETIMETOLONG	Lizenz-Zeitraum zu lang.
0x72C	1836	0x9811072C	ADSERR_DEVICE_EXCEPTION	Exception beim Systemstart.
0x72D	1837	0x9811072D	ADSERR_DEVICE_LICENSEDUPLICATED	Lizenz-Datei zweimal gelesen.
0x72E	1838	0x9811072E	ADSERR_DEVICE_SIGNATUREINVALID	Ungültige Signatur.
0x72F	1839	0x9811072F	ADSERR_DEVICE_CERTIFICATEINVALID	Zertifikat ungültig.
0x730	1840	0x98110730	ADSERR_DEVICE_LICENSEOEMNOTFOUND	Public Key vom OEM nicht bekannt.
0x731	1841	0x98110731	ADSERR_DEVICE_LICENSERESTRICTED	Lizenz nicht gültig für diese System.ID.
0x732	1842	0x98110732	ADSERR_DEVICE_LICENSEDEMOTDENIED	Demo-Lizenz untersagt.
0x733	1843	0x98110733	ADSERR_DEVICE_INVALIDFNCID	Funktions-ID ungültig.
0x734	1844	0x98110734	ADSERR_DEVICE_OUTOFRANGE	Außerhalb des gültigen Bereiches.
0x735	1845	0x98110735	ADSERR_DEVICE_INVALIDALIGNMENT	Ungültiges Alignment.

Hex	Dec	HRESULT	Name	Beschreibung
0x736	1846	0x98110736	ADSERR_DEVICE_LICENSEPLATFORM	Ungültiger Plattform Level.
0x737	1847	0x98110737	ADSERR_DEVICE_FORWARD_PL	Kontext – Weiterleitung zum Passiv-Level.
0x738	1848	0x98110738	ADSERR_DEVICE_FORWARD_DL	Kontext – Weiterleitung zum Dispatch-Level.
0x739	1849	0x98110739	ADSERR_DEVICE_FORWARD_RT	Kontext – Weiterleitung zur Echtzeit.
0x740	1856	0x98110740	ADSERR_CLIENT_ERROR	Clientfehler.
0x741	1857	0x98110741	ADSERR_CLIENT_INVALIDPARM	Dienst enthält einen ungültigen Parameter.
0x742	1858	0x98110742	ADSERR_CLIENT_LISTEMPTY	Polling-Liste ist leer.
0x743	1859	0x98110743	ADSERR_CLIENT_VARUSED	Var-Verbindung bereits im Einsatz.
0x744	1860	0x98110744	ADSERR_CLIENT_DUPLINVOKEID	Die aufgerufene ID ist bereits in Benutzung.
0x745	1861	0x98110745	ADSERR_CLIENT_SYNC TIMEOUT	Timeout ist aufgetreten – Die Gegenstelle antwortet nicht im vorgegebenen ADS Timeout. Die Routeneinstellung der Gegenstelle kann falsch konfiguriert sein.
0x746	1862	0x98110746	ADSERR_CLIENT_W32ERROR	Fehler im Win32 Subsystem.
0x747	1863	0x98110747	ADSERR_CLIENT_TIMEOUTINVALID	Ungültiger Client Timeout-Wert.
0x748	1864	0x98110748	ADSERR_CLIENT_PORTNOTOPEN	Port nicht geöffnet.
0x749	1865	0x98110749	ADSERR_CLIENT_NOAMSADDR	Keine AMS Adresse.
0x750	1872	0x98110750	ADSERR_CLIENT_SYNCINTERNAL	Interner Fehler in Ads-Sync.
0x751	1873	0x98110751	ADSERR_CLIENT_ADDHASH	Überlauf der Hash-Tabelle.
0x752	1874	0x98110752	ADSERR_CLIENT_REMOVEHASH	Schlüssel in der Tabelle nicht gefunden.
0x753	1875	0x98110753	ADSERR_CLIENT_NOMORESVM	Keine Symbole im Cache.
0x754	1876	0x98110754	ADSERR_CLIENT_SYNCRESINVALID	Ungültige Antwort erhalten.
0x755	1877	0x98110755	ADSERR_CLIENT_SYNCPORTLOCKED	Sync Port ist verriegelt.
0x756	1878	0x98110756	ADSERR_CLIENT_REQUESTCANCELLED	Die Anfrage wurde abgebrochen.

### RTime Fehlercodes

Hex	Dec	HRESULT	Name	Beschreibung
0x1000	4096	0x98111000	RTERR_INTERNAL	Interner Fehler im Echtzeit-System.
0x1001	4097	0x98111001	RTERR_BADTIMERPERIODS	Timer-Wert nicht gültig.
0x1002	4098	0x98111002	RTERR_INVALIDTASKPTR	Task-Pointer hat den ungültigen Wert 0 (null).
0x1003	4099	0x98111003	RTERR_INVALIDSTACKPTR	Stack-Pointer hat den ungültigen Wert 0 (null).
0x1004	4100	0x98111004	RTERR_PrioEXISTS	Die Request Task Priority ist bereits vergeben.
0x1005	4101	0x98111005	RTERR_NOMORETCB	Kein freier TCB (Task Control Block) verfügbar. Maximale Anzahl von TCBs beträgt 64.
0x1006	4102	0x98111006	RTERR_NOMORESEMAS	Keine freien Semaphoren zur Verfügung. Maximale Anzahl der Semaphoren beträgt 64.
0x1007	4103	0x98111007	RTERR_NOMOREQUEUES	Kein freier Platz in der Warteschlange zur Verfügung. Maximale Anzahl der Plätze in der Warteschlange beträgt 64.
0x100D	4109	0x9811100D	RTERR_EXTIRQALREADYDEF	Ein externer Synchronisations-Interrupt wird bereits angewandt.
0x100E	4110	0x9811100E	RTERR_EXTIRQNOTDEF	Kein externer Sync-Interrupt angewandt.
0x100F	4111	0x9811100F	RTERR_EXTIRQINSTALLFAILED	Anwendung des externen Synchronisierungs-Interrupts ist fehlgeschlagen.
0x1010	4112	0x98111010	RTERR_IRQLNOTLESSOREQUAL	Aufruf einer Service-Funktion im falschen Kontext
0x1017	4119	0x98111017	RTERR_VMXNOTSUPPORTED	Intel VT-x Erweiterung wird nicht unterstützt.
0x1018	4120	0x98111018	RTERR_VMXDISABLED	Intel VT-x Erweiterung ist nicht aktiviert im BIOS.
0x1019	4121	0x98111019	RTERR_VMXCONTROLSMISSING	Fehlende Funktion in Intel VT-x Erweiterung.
0x101A	4122	0x9811101A	RTERR_VMXENABLEFAILS	Aktivieren von Intel VT-x schlägt fehl.

### Spezifische positive HRESULT Return Codes:

HRESULT	Name	Beschreibung
0x0000_0000	S_OK	Kein Fehler.
0x0000_0001	S_FALSE	Kein Fehler. Bsp.: erfolgreiche Abarbeitung, bei der jedoch ein negatives oder unvollständiges Ergebnis erzielt wurde.
0x0000_0203	S_PENDING	Kein Fehler. Bsp.: erfolgreiche Abarbeitung, bei der jedoch noch kein Ergebnis vorliegt.
0x0000_0256	S_WATCHDOG_TIMEOUT	Kein Fehler. Bsp.: erfolgreiche Abarbeitung, bei der jedoch eine Zeitüberschreitung eintrat.

**TCP Winsock-Fehlercodes**

Hex	Dec	Name	Beschreibung
0x274C	10060	WSAETIMEDOUT	Verbindungs Timeout aufgetreten - Fehler beim Herstellen der Verbindung, da die Gegenstelle nach einer bestimmten Zeitspanne nicht ordnungsgemäß reagiert hat, oder die hergestellte Verbindung konnte nicht aufrecht erhalten werden, da der verbundene Host nicht reagiert hat.
0x274D	10061	WSAECONNREFUSED	Verbindung abgelehnt - Es konnte keine Verbindung hergestellt werden, da der Zielcomputer dies explizit abgelehnt hat. Dieser Fehler resultiert normalerweise aus dem Versuch, eine Verbindung mit einem Dienst herzustellen, der auf dem fremden Host inaktiv ist—das heißt, einem Dienst, für den keine Serveranwendung ausgeführt wird.
0x2751	10065	WSAEHOSTUNREACH	Keine Route zum Host - Ein Socketvorgang bezog sich auf einen nicht verfügbaren Host.

Weitere Winsock-Fehlercodes: [Win32-Fehlercodes](#) [[↗ 713](#)]

**7.2.2 Fehlersuche/Diagnose**

- Überprüfen Sie die in dieser Dokumentation beschriebenen Hardware- und Softwareanforderungen (TwinCAT-Version, CE Image-Version usw.);
- Vergleichen/überprüfen Sie die [Kompatibilitätsliste der Zentralstation](#) [[↗ 49](#)] mit der Kompatibilitätsliste der Unterstation;
- Überprüfen Sie die IO-Konfiguration und das Mapping der SPS-Variablen in TwinCAT System Manager ([Konfiguration der seriellen Schnittstellen](#) [[↗ 561](#)], Baudrate, Parity, Stopbits usw.). Vergleichen Sie die Parameter mit den Parametern der Unterstation;
- Überprüfen Sie ob der [FB IEC870\\_5\\_102TProtocol](#) [[↗ 408](#)]-Funktionsbaustein einen [Fehlercode](#) [[↗ 694](#)] ausgibt;
- Überprüfen Sie die am [FB IEC870\\_5\\_102TProtocol](#) [[↗ 408](#)]-Funktionsbaustein eingestellten [Protokollparameter](#) [[↗ 538](#)] (Link-Adresse, FRAMELength, usw.). Vergleichen Sie die Protokollparameter mit den Parametern der Unterstation;
- Überprüfen Sie die am TX/RX-Datenpuffer (instanz von [ST IEC870\\_5\\_102TBuffer](#) [[↗ 544](#)]) konfigurierten Adresslängen: Länge der ASDU-Adresse, max. ASDU-Länge. Vergleichen Sie die Adresslängen mit der Konfiguration der Unterstation;
- Überprüfen Sie die Konfiguration der Datenpunkte (Typ, Adresse des Informationsobjektes usw.);
- Überprüfen Sie ob die Unterstation einen Fehlercode ausgibt;
- Aktivieren Sie die [Debugausgaben](#) [[↗ 699](#)] beim Aufbauen und Abbauen der Verbindung und/oder der ASDU Daten. Öffnen Sie den TwinCAT System Manager und aktivieren das LogView-Fenster. Prüfen Sie die Debugausgaben;

**7.2.3 Debugmeldungen in TwinCAT->Error List**

Debugmeldungen, die in TwinCAT XAE->Error List geschrieben werden erleichtern eine Fehlersuche im System. Um die aktivierten Debugmeldungen zu sehen starten Sie TwinCAT XAE und wählen im Menue: View->Other Windows->Error List. Die nachfolgende Grafik zeigt eine Debugausgabe. Die unterschiedlichen Typen der Meldungen wurden mit Zahlen markiert.

	Description	
①	29.10.2013 11:47:33 856 ms   LowSpeed (B51): Serial Link2 PRM TX<=10 49 DC 00 25 16	2
②	29.10.2013 11:47:34 606 ms   LowSpeed (B51): Serial Link2 PRM TX<=10 49 DC 00 25 16	
③	29.10.2013 11:47:34 646 ms   LowSpeed (B51): Serial Link2 PRM RX<=10 08 DC 00 E7 16	
④	29.10.2013 11:47:34 686 ms   LowSpeed (B51): Serial Link2 PRM TX<=10 40 DC 00 1C 16	
⑤	29.10.2013 11:47:34 716 ms   LowSpeed (B51): Serial Link2 PRM RX<=E5	
⑥	29.10.2013 11:47:34 746 ms   LowSpeed (B51): Serial Link2 PRM TX<=10 7B DC 00 57 16	
⑦	29.10.2013 11:47:34 746 ms   LowSpeed (B51): Class1 TX FIFO<=66 01 06 07 00 33 00 00 01 01 07 00 00 01 0C 07	1
⑧	29.10.2013 11:47:34 746 ms   LowSpeed (B51): IEC 60870-5-102M, linkAddr220, CAA:7<=TYPE:C_SP_NB_2(102, 0x66), COT:ACT(006, 0x06), Command activation, SQ:0, NO:1, T:0, PN:0, CAA:7, RCA:S1, LEN:10, DATA:00 00 01 01 07 00 00 01 0C 07	3
⑨	29.10.2013 11:47:34 776 ms   LowSpeed (B51): Serial Link2 PRM RX<=E5	2
⑩	29.10.2013 11:47:34 816 ms   LowSpeed (B51): Serial Link2 PRM TX<=68 13 13 68 53 DC 00 66 01 06 07 00 33 00 00 01 01 07 00 00 01 0C 07 F3 16	
⑪	29.10.2013 11:47:34 876 ms   LowSpeed (B51): Serial Link2 PRM RX<=10 20 DC 00 FC 16	
⑫	29.10.2013 11:47:34 886 ms   LowSpeed (B51): Class1 TX FIFO<=66 01 06 07 00 33 00 00 01 01 07 00 00 01 0C 07	1
⑬	29.10.2013 11:47:35 116 ms   LowSpeed (B51): Serial Link2 PRM TX<=10 7A DC 00 56 16	2
⑭	29.10.2013 11:47:35 176 ms   LowSpeed (B51): IEC 60870-5-102M, linkAddr220, CAA:7<=TYPE:C_RD_NA_2(100, 0x64), COT:REQ(005, 0x05), Request or requested, SQ:0, NO:0, T:0, PN:0, CAA:7, RCA:0	3
⑮	29.10.2013 11:47:35 186 ms   LowSpeed (B51): RX FIFO<=66 01 07 07 00 33 00 00 01 01 07 00 00 01 0C 07	1
⑯	29.10.2013 11:47:35 186 ms   LowSpeed (B51): RX FIFO<=66 01 07 07 00 33 00 00 01 01 07 00 00 01 0C 07	
⑰	29.10.2013 11:47:35 186 ms   LowSpeed (B51): IEC 60870-5-102M, linkAddr220, CAA:7<=TYPE:C_SP_NB_2(102, 0x66), COT:ACT_CON(007, 0x07), Command activation confirmation, SQ:0, NO:1, T:0, PN:0, CAA:7, RCA:S1, LEN:10, DATA:00 00 01 01 07 00 00 01 0C 07	3
⑱	29.10.2013 11:47:35 216 ms   LowSpeed (B51): Class1 TX FIFO<=64 00 05 07 00 00	1
⑲	29.10.2013 11:47:35 216 ms   LowSpeed (B51): IEC 60870-5-102M, linkAddr220, CAA:7<=TYPE:C_RD_NA_2(100, 0x64), COT:REQ(005, 0x05), Request or requested, SQ:0, NO:0, T:0, PN:0, CAA:7, RCA:0	3
⑳	29.10.2013 11:47:35 246 ms   LowSpeed (B51): Serial Link2 PRM TX<=68 09 09 68 53 DC 00 64 00 05 07 00 00 9F 16	2
㉑	29.10.2013 11:47:35 286 ms   LowSpeed (B51): Serial Link2 PRM RX<=E5	
㉒	29.10.2013 11:47:35 296 ms   LowSpeed (B51): Class1 TX FIFO<=64 00 05 07 00 00	1
㉓	29.10.2013 11:47:35 416 ms   LowSpeed (B51): Serial Link2 PRM TX<=10 7B DC 00 57 16	
㉔	29.10.2013 11:47:35 446 ms   LowSpeed (B51): Serial Link2 PRM RX<=E5	
㉕	29.10.2013 11:47:35 686 ms   LowSpeed (B51): Serial Link2 PRM TX<=10 58 DC 00 37 16	
㉖	29.10.2013 11:47:35 716 ms   LowSpeed (B51): Serial Link2 PRM RX<=E5	

Die Meldungen werden auf unterschiedliche Weise (durch unterschiedliche Parameter) aktiviert. Unten folgt die Auflistung der unterschiedlichen Parameternamen.

1. Hexadezimale Ausgabe der ASDU's (ohne Link-Layer-Control-Header). Pro Zeile werden 32-ASDU-Datenbytes als hexadezimale Zahlen ausgegeben. Längere ASDU's werden auf mehrere Zeilen verteilt:

Name	AN	AUS
ST_IEC870_5_102TBuffer [▶ 544].eDbg	eIEC870_FIFO_DBG_ALL	eIEC870_FIFO_DBG_OFF

2. Hexadezimale Ausgabe der APDU's (RS232/RS485 Telegramme). Pro Zeile werden 32-APDU-Datenbytes als hexadezimale Zahlen ausgegeben. Längere APDU's werden auf mehrere Zeilen verteilt:

Name	AN	AUS
FB_IEC870_5_102TProtocol [▶ 408].bOutDbg	TRUE	FALSE

3. Meldungen, die durch die SPS-Applikation (Beispielprojekt) mit Hilfe des FB\_IEC870\_DebugLogFifo-Funktionsbausteins geloggt werden.

Name	AN	AUS
FB_IEC870_DebugLogFifo [▶ 419]	Aktionsaufruf: A_LogError, A_LogWarning, A_LogHint	-

Weitere Diagnosetools:

- Portmon for Windows (v3.02, Windows Sysinternals);
- Diverse Protokoll-Test-Suit-Produkte;

## 7.3 IEC 60870-5-103

### 7.3.1 TwinCAT IEC 60870-5-103 Fehlercodes

#### Voraussetzungen

Fehlercodes (Hex)	Fehlercodes (Dez)	Fehlerquelle	Beschreibung
0x00000000-0x00002000	0-8192	TF6340 Serial-Communication Fehlercodes	TF6340 Serial-Communication Fehler (aber nur wenn die Fehlerquelle [▶ 508] = eIEC870_ESRC_IEC60870_5_101LINK ).
0x00000000-0x00007800	0-30720	<u>TwinCAT System Fehlercodes [▶ 701]</u>	TwinCAT System Fehler (ADS-Fehlercodes inklusive wenn Fehlerquelle [▶ 508] <> eIEC870_ESRC_IEC60870_5_101LINK ).
0x00008100-0x000081FF	32768-33023	<u>TwinCAT IEC 60870-5-10x Fehlercodes [▶ 557]</u>	TwinCAT IEC 60870-5-10x Protokollfehler/ Kommunikationsfehler (z.B.: Timeoutfehler, Telegrammfehler, Konfigurationsfehler).

#### 7.3.1.1 ADS Return Codes

Gruppierung der Fehlercodes:

Globale Fehlercodes: 0x0000 [▶ 701]... (0x9811\_0000 ...)

Router Fehlercodes: 0x0500 [▶ 702]... (0x9811\_0500 ...)

Allgemeine ADS Fehler: 0x0700 [▶ 703]... (0x9811\_0700 ...)

RTime Fehlercodes: 0x1000 [▶ 705]... (0x9811\_1000 ...)

#### Globale Fehlercodes

Hex	Dec	HRESULT	Name	Beschreibung
0x0	0	0x98110000	ERR_NOERROR	Kein Fehler.
0x1	1	0x98110001	ERR_INTERNAL	Interner Fehler.
0x2	2	0x98110002	ERR_NORTIME	Keine Echtzeit.
0x3	3	0x98110003	ERR_ALLOCLOCKEDMEM	Zuweisung gesperrt - Speicherfehler.
0x4	4	0x98110004	ERR_INSERTMAILBOX	Postfach voll – Es konnte die ADS Nachricht nicht versendet werden. Reduzieren der Anzahl der ADS Nachrichten pro Zyklus bringt Abhilfe.
0x5	5	0x98110005	ERR_WRONGRECEIVEHMSG	Falsches HMSG.
0x6	6	0x98110006	ERR_TARGETPORTNOTFOUND	Ziel-Port nicht gefunden – ADS Server ist nicht gestartet, nicht erreichbar oder nicht installiert.
0x7	7	0x98110007	ERR_TARGETMACHINENOTFOUND	Zielrechner nicht gefunden – AMS Route wurde nicht gefunden.
0x8	8	0x98110008	ERR_UNKNOWNCMDID	Unbekannte Befehl-ID.
0x9	9	0x98110009	ERR_BADTASKID	Ungültige Task-ID.
0xA	10	0x9811000A	ERR_NOIO	Kein IO.
0xB	11	0x9811000B	ERR_UNKNOWNAMSCMD	Unbekannter AMS-Befehl.
0xC	12	0x9811000C	ERR_WIN32ERROR	Win32 Fehler.
0xD	13	0x9811000D	ERR_PORTNOTCONNECTED	Port nicht verbunden.
0xE	14	0x9811000E	ERR_INVALIDAMSLENGTH	Ungültige AMS-Länge.
0xF	15	0x9811000F	ERR_INVALIDAMSNETID	Ungültige AMS Net ID.
0x10	16	0x98110010	ERR_LOWINSTLEVEL	Installations-Level ist zu niedrig –TwinCAT 2 Lizenzfehler.
0x11	17	0x98110011	ERR_NODEBUGINTAVAILABLE	Kein Debugging verfügbar.
0x12	18	0x98110012	ERR_PORTDISABLED	Port deaktiviert – TwinCAT System Service nicht gestartet.
0x13	19	0x98110013	ERR_PORTALREADYCONNECTED	Port bereits verbunden.
0x14	20	0x98110014	ERR_AMSSYNC_W32ERROR	AMS Sync Win32 Fehler.
0x15	21	0x98110015	ERR_AMSSYNC_TIMEOUT	AMS Sync Timeout.
0x16	22	0x98110016	ERR_AMSSYNC_AMSERROR	AMS Sync Fehler.
0x17	23	0x98110017	ERR_AMSSYNC_NOINDEXINMAP	Keine Index-Map für AMS Sync vorhanden.
0x18	24	0x98110018	ERR_INVALIDAMSPORT	Ungültiger AMS-Port.
0x19	25	0x98110019	ERR_NOMEMORY	Kein Speicher.
0x1A	26	0x9811001A	ERR_TCPSEND	TCP Sendefehler.
0x1B	27	0x9811001B	ERR_HOSTUNREACHABLE	Host nicht erreichbar.
0x1C	28	0x9811001C	ERR_INVALIDAMSFRAGMENT	Ungültiges AMS Fragment.
0x1D	29	0x9811001D	ERR_TLSEND	TLS Sendefehler – Secure ADS Verbindung fehlgeschlagen.
0x1E	30	0x9811001E	ERR_ACCESSDENIED	Zugriff Verweigert – Secure ADS Zugriff verweigert.

## Router Fehlercodes

Hex	Dec	HRESULT	Name	Beschreibung
0x500	1280	0x98110500	ROUTERERR_NOLOCKEDMEMORY	Lockierter Speicher kann nicht zugewiesen werden.
0x501	1281	0x98110501	ROUTERERR_RESIZEMEMORY	Die Größe des Routerspeichers konnte nicht geändert werden.
0x502	1282	0x98110502	ROUTERERR_MAILBOXFULL	Das Postfach hat die maximale Anzahl der möglichen Meldungen erreicht.
0x503	1283	0x98110503	ROUTERERR_DEBUGBOXFULL	Das Debug Postfach hat die maximale Anzahl der möglichen Meldungen erreicht.
0x504	1284	0x98110504	ROUTERERR_UNKNOWNPORTTYPE	Der Porttyp ist unbekannt.
0x505	1285	0x98110505	ROUTERERR_NOTINITIALIZED	Router ist nicht initialisiert.
0x506	1286	0x98110506	ROUTERERR_PORTALREADYINUSE	Die Portnummer ist bereits vergeben.
0x507	1287	0x98110507	ROUTERERR_NOTREGISTERED	Der Port ist nicht registriert.
0x508	1288	0x98110508	ROUTERERR_NOMOREQUEUES	Die maximale Portanzahl ist erreicht.
0x509	1289	0x98110509	ROUTERERR_INVALIDPORT	Der Port ist ungültig.
0x50A	1290	0x9811050A	ROUTERERR_NOTACTIVATED	Der Router ist nicht aktiv.
0x50B	1291	0x9811050B	ROUTERERR_FRAGMENTBOXFULL	Das Postfach hat die maximale Anzahl für fragmentierte Nachrichten erreicht.
0x50C	1292	0x9811050C	ROUTERERR_FRAGMENTTIMEOUT	Fragment Timeout aufgetreten.
0x50D	1293	0x9811050D	ROUTERERR_TOBEREMOVED	Port wird entfernt.

**Allgemeine ADS Fehlercodes**

Hex	Dec	HRESULT	Name	Beschreibung
0x700	1792	0x98110700	ADSERR_DEVICE_ERROR	Allgemeiner Gerätefehler.
0x701	1793	0x98110701	ADSERR_DEVICE_SRVNOTSUPP	Service wird vom Server nicht unterstützt.
0x702	1794	0x98110702	ADSERR_DEVICE_INVALIDGRP	Ungültige Index-Gruppe.
0x703	1795	0x98110703	ADSERR_DEVICE_INVALIDOFFSET	Ungültiger Index-Offset.
0x704	1796	0x98110704	ADSERR_DEVICE_INVALIDACCESS	Lesen oder Schreiben nicht gestattet.
0x705	1797	0x98110705	ADSERR_DEVICE_INVALIDSIZE	Parametergröße nicht korrekt.
0x706	1798	0x98110706	ADSERR_DEVICE_INVALIDDATA	Ungültige Daten-Werte.
0x707	1799	0x98110707	ADSERR_DEVICE_NOTREADY	Gerät nicht betriebsbereit.
0x708	1800	0x98110708	ADSERR_DEVICE_BUSY	Gerät beschäftigt.
0x709	1801	0x98110709	ADSERR_DEVICE_INVALIDCONTEXT	Ungültiger Kontext vom Betriebssystem - Kann durch Verwendung von ADS Bausteinen in unterschiedlichen Tasks auftreten. Abhilfe kann die Multitasking-Synchronisation in der SPS geben.
0x70A	1802	0x9811070A	ADSERR_DEVICE_NOMEMORY	Nicht genügend Speicher.
0x70B	1803	0x9811070B	ADSERR_DEVICE_INVALIDPARM	Ungültige Parameter-Werte.
0x70C	1804	0x9811070C	ADSERR_DEVICE_NOTFOUND	Nicht gefunden (Dateien,...).
0x70D	1805	0x9811070D	ADSERR_DEVICE_SYNTAX	Syntax-Fehler in Datei oder Befehl.
0x70E	1806	0x9811070E	ADSERR_DEVICE_INCOMPATIBLE	Objekte stimmen nicht überein.
0x70F	1807	0x9811070F	ADSERR_DEVICE_EXISTS	Objekt ist bereits vorhanden.
0x710	1808	0x98110710	ADSERR_DEVICE_SYMBOLNOTFOUND	Symbol nicht gefunden.
0x711	1809	0x98110711	ADSERR_DEVICE_SYMBOLVERSIONINVALID	Symbol-Version ungültig – Kann durch einen Online-Change auftreten. Erzeuge einen neuen Handle.
0x712	1810	0x98110712	ADSERR_DEVICE_INVALIDSTATE	Gerät (Server) ist im ungültigen Zustand.
0x713	1811	0x98110713	ADSERR_DEVICE_TRANSMODENOTSUPP	AdsTransMode nicht unterstützt.
0x714	1812	0x98110714	ADSERR_DEVICE_NOTIFYHANDINVALID	Notification Handle ist ungültig.
0x715	1813	0x98110715	ADSERR_DEVICE_CLIENTUNKNOWN	Notification-Client nicht registriert.
0x716	1814	0x98110716	ADSERR_DEVICE_NOMOREHDL	Keine weiteren Handles verfügbar.
0x717	1815	0x98110717	ADSERR_DEVICE_INVALIDWATCHSIZE	Größe der Notification zu groß.
0x718	1816	0x98110718	ADSERR_DEVICE_NOTINIT	Gerät nicht initialisiert.
0x719	1817	0x98110719	ADSERR_DEVICE_TIMEOUT	Gerät hat einen Timeout.
0x71A	1818	0x9811071A	ADSERR_DEVICE_NOINTERFACE	Interface Abfrage fehlgeschlagen.
0x71B	1819	0x9811071B	ADSERR_DEVICE_INVALIDINTERFACE	Falsches Interface angefordert.
0x71C	1820	0x9811071C	ADSERR_DEVICE_INVALIDCLSID	Class-ID ist ungültig.
0x71D	1821	0x9811071D	ADSERR_DEVICE_INVALIDOBJID	Object-ID ist ungültig.
0x71E	1822	0x9811071E	ADSERR_DEVICE_PENDING	Anforderung steht aus.
0x71F	1823	0x9811071F	ADSERR_DEVICE_ABORTED	Anforderung wird abgebrochen.
0x720	1824	0x98110720	ADSERR_DEVICE_WARNING	Signal-Warnung.
0x721	1825	0x98110721	ADSERR_DEVICE_INVALIDARRAYIDX	Ungültiger Array-Index.
0x722	1826	0x98110722	ADSERR_DEVICE_SYMBOLNOTACTIVE	Symbol nicht aktiv.
0x723	1827	0x98110723	ADSERR_DEVICE_ACCESSDENIED	Zugriff verweigert.
0x724	1828	0x98110724	ADSERR_DEVICE_LICENSENOTFOUND	Fehlende Lizenz.
0x725	1829	0x98110725	ADSERR_DEVICE_LICENSEEXPIRED	Lizenz abgelaufen.
0x726	1830	0x98110726	ADSERR_DEVICE_LICENSEEXCEEDED	Lizenz überschritten.
0x727	1831	0x98110727	ADSERR_DEVICE_LICENSEINVALID	Lizenz ungültig.
0x728	1832	0x98110728	ADSERR_DEVICE_LICENSESYSTEMID	Lizenzproblem: System-ID ist ungültig.
0x729	1833	0x98110729	ADSERR_DEVICE_LICENSENOTIMELIMIT	Lizenz nicht zeitlich begrenzt.
0x72A	1834	0x9811072A	ADSERR_DEVICE_LICENSEFUTUREISSUE	Lizenzproblem: Zeitpunkt in der Zukunft.
0x72B	1835	0x9811072B	ADSERR_DEVICE_LICENSETIMETOLONG	Lizenz-Zeitraum zu lang.
0x72C	1836	0x9811072C	ADSERR_DEVICE_EXCEPTION	Exception beim Systemstart.
0x72D	1837	0x9811072D	ADSERR_DEVICE_LICENSEDUPLICATED	Lizenz-Datei zweimal gelesen.
0x72E	1838	0x9811072E	ADSERR_DEVICE_SIGNATUREINVALID	Ungültige Signatur.
0x72F	1839	0x9811072F	ADSERR_DEVICE_CERTIFICATEINVALID	Zertifikat ungültig.
0x730	1840	0x98110730	ADSERR_DEVICE_LICENSEOEMNOTFOUND	Public Key vom OEM nicht bekannt.
0x731	1841	0x98110731	ADSERR_DEVICE_LICENSERESTRICTED	Lizenz nicht gültig für diese System.ID.
0x732	1842	0x98110732	ADSERR_DEVICE_LICENSEDEMODENIED	Demo-Lizenz untersagt.
0x733	1843	0x98110733	ADSERR_DEVICE_INVALIDFNCID	Funktions-ID ungültig.
0x734	1844	0x98110734	ADSERR_DEVICE_OUTOFRANGE	Außerhalb des gültigen Bereiches.
0x735	1845	0x98110735	ADSERR_DEVICE_INVALIDALIGNMENT	Ungültiges Alignment.



Hex	Dec	HRESULT	Name	Beschreibung
0x736	1846	0x98110736	ADSERR_DEVICE_LICENSEPLATFORM	Ungültiger Plattform Level.
0x737	1847	0x98110737	ADSERR_DEVICE_FORWARD_PL	Kontext – Weiterleitung zum Passiv-Level.
0x738	1848	0x98110738	ADSERR_DEVICE_FORWARD_DL	Kontext – Weiterleitung zum Dispatch-Level.
0x739	1849	0x98110739	ADSERR_DEVICE_FORWARD_RT	Kontext – Weiterleitung zur Echtzeit.
0x740	1856	0x98110740	ADSERR_CLIENT_ERROR	Clientfehler.
0x741	1857	0x98110741	ADSERR_CLIENT_INVALIDPARM	Dienst enthält einen ungültigen Parameter.
0x742	1858	0x98110742	ADSERR_CLIENT_LISTEMPTY	Polling-Liste ist leer.
0x743	1859	0x98110743	ADSERR_CLIENT_VARUSED	Var-Verbindung bereits im Einsatz.
0x744	1860	0x98110744	ADSERR_CLIENT_DUPLINVOKEID	Die aufgerufene ID ist bereits in Benutzung.
0x745	1861	0x98110745	ADSERR_CLIENT_SYNC TIMEOUT	Timeout ist aufgetreten – Die Gegenstelle antwortet nicht im vorgegebenen ADS Timeout. Die Routeneinstellung der Gegenstelle kann falsch konfiguriert sein.
0x746	1862	0x98110746	ADSERR_CLIENT_W32ERROR	Fehler im Win32 Subsystem.
0x747	1863	0x98110747	ADSERR_CLIENT_TIMEOUTINVALID	Ungültiger Client Timeout-Wert.
0x748	1864	0x98110748	ADSERR_CLIENT_PORTNOTOPEN	Port nicht geöffnet.
0x749	1865	0x98110749	ADSERR_CLIENT_NOAMSADDR	Keine AMS Adresse.
0x750	1872	0x98110750	ADSERR_CLIENT_SYNCINTERNAL	Interner Fehler in Ads-Sync.
0x751	1873	0x98110751	ADSERR_CLIENT_ADDHASH	Überlauf der Hash-Tabelle.
0x752	1874	0x98110752	ADSERR_CLIENT_REMOVEHASH	Schlüssel in der Tabelle nicht gefunden.
0x753	1875	0x98110753	ADSERR_CLIENT_NOMORESVM	Keine Symbole im Cache.
0x754	1876	0x98110754	ADSERR_CLIENT_SYNCRESINVALID	Ungültige Antwort erhalten.
0x755	1877	0x98110755	ADSERR_CLIENT_SYNCPORTLOCKED	Sync Port ist verriegelt.
0x756	1878	0x98110756	ADSERR_CLIENT_REQUESTCANCELLED	Die Anfrage wurde abgebrochen.

**RTime Fehlercodes**

Hex	Dec	HRESULT	Name	Beschreibung
0x1000	4096	0x98111000	RTERR_INTERNAL	Interner Fehler im Echtzeit-System.
0x1001	4097	0x98111001	RTERR_BADTIMERPERIODS	Timer-Wert nicht gültig.
0x1002	4098	0x98111002	RTERR_INVALIDTASKPTR	Task-Pointer hat den ungültigen Wert 0 (null).
0x1003	4099	0x98111003	RTERR_INVALIDSTACKPTR	Stack-Pointer hat den ungültigen Wert 0 (null).
0x1004	4100	0x98111004	RTERR_PrioEXISTS	Die Request Task Priority ist bereits vergeben.
0x1005	4101	0x98111005	RTERR_NOMORETCB	Kein freier TCB (Task Control Block) verfügbar. Maximale Anzahl von TCBs beträgt 64.
0x1006	4102	0x98111006	RTERR_NOMORESEMAS	Keine freien Semaphoren zur Verfügung. Maximale Anzahl der Semaphoren beträgt 64.
0x1007	4103	0x98111007	RTERR_NOMOREQUEUES	Kein freier Platz in der Warteschlange zur Verfügung. Maximale Anzahl der Plätze in der Warteschlange beträgt 64.
0x100D	4109	0x9811100D	RTERR_EXTIRQALREADYDEF	Ein externer Synchronisations-Interrupt wird bereits angewandt.
0x100E	4110	0x9811100E	RTERR_EXTIRQNOTDEF	Kein externer Sync-Interrupt angewandt.
0x100F	4111	0x9811100F	RTERR_EXTIRQINSTALLFAILED	Anwendung des externen Synchronisierungs-Interrupts ist fehlgeschlagen.
0x1010	4112	0x98111010	RTERR_IRQLNOTLESSOREQUAL	Aufruf einer Service-Funktion im falschen Kontext
0x1017	4119	0x98111017	RTERR_VMXNOTSUPPORTED	Intel VT-x Erweiterung wird nicht unterstützt.
0x1018	4120	0x98111018	RTERR_VMXDISABLED	Intel VT-x Erweiterung ist nicht aktiviert im BIOS.
0x1019	4121	0x98111019	RTERR_VMXCONTROLSMISSING	Fehlende Funktion in Intel VT-x Erweiterung.
0x101A	4122	0x9811101A	RTERR_VMXENABLEFAILS	Aktivieren von Intel VT-x schlägt fehl.

**Spezifische positive HRESULT Return Codes:**

HRESULT	Name	Beschreibung
0x0000_0000	S_OK	Kein Fehler.
0x0000_0001	S_FALSE	Kein Fehler. Bsp.: erfolgreiche Abarbeitung, bei der jedoch ein negatives oder unvollständiges Ergebnis erzielt wurde.
0x0000_0203	S_PENDING	Kein Fehler. Bsp.: erfolgreiche Abarbeitung, bei der jedoch noch kein Ergebnis vorliegt.
0x0000_0256	S_WATCHDOG_TIMEOUT	Kein Fehler. Bsp.: erfolgreiche Abarbeitung, bei der jedoch eine Zeitüberschreitung eintrat.

### TCP Winsock-Fehlercodes

Hex	Dec	Name	Beschreibung
0x274C	10060	WSAETIMEDOUT	Verbindungs Timeout aufgetreten - Fehler beim Herstellen der Verbindung, da die Gegenstelle nach einer bestimmten Zeitspanne nicht ordnungsgemäß reagiert hat, oder die hergestellte Verbindung konnte nicht aufrecht erhalten werden, da der verbundene Host nicht reagiert hat.
0x274D	10061	WSAECONNREFUSED	Verbindung abgelehnt - Es konnte keine Verbindung hergestellt werden, da der Zielcomputer dies explizit abgelehnt hat. Dieser Fehler resultiert normalerweise aus dem Versuch, eine Verbindung mit einem Dienst herzustellen, der auf dem fremden Host inaktiv ist—das heißt, einem Dienst, für den keine Serveranwendung ausgeführt wird.
0x2751	10065	WSAEHOSTUNREACH	Keine Route zum Host - Ein Socketvorgang bezog sich auf einen nicht verfügbaren Host.

Weitere Winsock-Fehlercodes: [Win32-Fehlercodes \[▶ 713\]](#)

## 7.3.2 Fehlersuche/Diagnose

- Überprüfen Sie die in dieser Dokumentation beschriebenen Hardware- und Softwareanforderungen (TwinCAT-Version, CE Image-Version usw.);
- Vergleichen/überprüfen Sie die [Kompatibilitätsliste der Zentralstation \[▶ 57\]](#) mit der Kompatibilitätsliste der Unterstation;
- Überprüfen Sie die IO-Konfiguration und das Mapping der SPS-Variablen in TwinCAT System Manager ([Konfiguration der seriellen Schnittstellen \[▶ 561\]](#), Baudrate, Parity, Stopbits usw.). Vergleichen Sie die Parameter mit den Parametern der Unterstation;
- Überprüfen Sie ob der [FB IEC870\\_5\\_103TProtocol \[▶ 410\]](#)-Funktionsbaustein einen [Fehlercode \[▶ 701\]](#) ausgibt;
- Überprüfen Sie die am [FB IEC870\\_5\\_103TProtocol \[▶ 410\]](#)-Funktionsbaustein eingestellten [Protokolparameter \[▶ 538\]](#) (Link-Adresse, Länge der Link-Adresse, FRAMELength, usw.). Vergleichen Sie die Protokolparameter mit den Parametern der Unterstation;
- Überprüfen Sie die am TX/RX-Datenpuffer (instanz von [ST IEC870\\_5\\_103TBuffer \[▶ 547\]](#)) konfigurierte max. ASDU-Länge. Vergleichen Sie die Länge mit der Konfiguration der Unterstation;
- Überprüfen Sie die Konfiguration der Datenpunkte (Typ, ASDU-Adresse, Funktionsnummer, Informationsnummer usw.);
- Überprüfen Sie ob die Unterstation einen Fehlercode ausgibt;
- Aktivieren Sie die [Debugausgaben \[▶ 706\]](#) beim Aufbauen und Abbauen der Verbindung und/oder der ASDU Daten. Öffnen Sie den TwinCAT System Manager und aktivieren das LogView-Fenster. Prüfen Sie die Debugausgaben;

## 7.3.3 Debugmeldungen in TwinCAT->Error List

Debugmeldungen, die in TwinCAT XAE->Error List geschrieben werden erleichtern eine Fehlersuche im System. Um die aktivierten Debugmeldungen zu sehen starten Sie TwinCAT XAE und wählen im Menue: View->Other Windows->Error List. Die nachfolgende Grafik zeigt eine Debugausgabe. Die unterschiedlichen Typen der Meldungen wurden mit Zahlen markiert.

	Description
0 1	29.10.2013 12:11:13 925 ms   'LowPrio' (351): Serial Link(1)PRM.TX<=10 5B DC 37 16
① 2	29.10.2013 12:11:16 185 ms   'LowPrio' (351): Serial Link(1)PRM.TX<=10 5B DC 37 16 2
① 3	29.10.2013 12:11:18 445 ms   'LowPrio' (351): Serial Link(1)PRM.TX<=10 5B DC 37 16
① 4	29.10.2013 12:11:19 625 ms   'LowPrio' (351): IEC 60870-5-103M, linkAddr220, CAA:220=>Transport interface error: 0x0000810F
① 5	29.10.2013 12:11:19 865 ms   'LowPrio' (351): Serial Link(2)PRM.TX<=10 40 DC 1C 16 2
① 6	29.10.2013 12:11:19 905 ms   'LowPrio' (351): Serial Link(2)PRM.RX=>10 20 DC FC 16
① 7	29.10.2013 12:11:20 85 ms   'LowPrio' (351): Class1 TX FIFO<=06 81 08 DC FF 00 6E 4E 0B 0C 5D 0A 0D 1
① 8	29.10.2013 12:11:20 85 ms   'LowPrio' (351): IEC 60870-5-103M, linkAddr220, CAA:220<=>TYPE:C_SYN_TA_3(006, 0x06), COT:SYN(008, 0x08), Time synchronisation, SQ1, NO1, CAA:220, FC:255, FN:0, LEN:7, DATA:6E 4E 0B 0C 5D 0A 0D 3
① 9	29.10.2013 12:11:20 185 ms   'LowPrio' (351): Serial Link(2)PRM.TX<=10 7B DC 57 16 2
① 10	29.10.2013 12:11:20 225 ms   'LowPrio' (351): Serial Link(2)PRM.RX=>10 29 DC 05 16
① 11	29.10.2013 12:11:20 525 ms   'LowPrio' (351): Serial Link(2)PRM.TX<=68 0F 68 53 DC 06 81 08 DC FF 00 6E 4E 0B 0C 5D 0A 0D E0 16
① 12	29.10.2013 12:11:20 565 ms   'LowPrio' (351): Serial Link(2)PRM.RX=>10 20 DC FC 16
① 13	29.10.2013 12:11:20 705 ms   'LowPrio' (351): Class1 TX FIFO=>06 81 08 DC FF 00 6E 4E 0B 0C 5D 0A 0D 1
① 14	29.10.2013 12:11:21 65 ms   'LowPrio' (351): Serial Link(2)PRM.TX<=10 7A DC 56 16 2
① 15	29.10.2013 12:11:21 125 ms   'LowPrio' (351): Serial Link(2)PRM.RX=>68 0F 68 08 DC 06 81 08 DC FF 00 6E 4E 0B 0C 5D 0A 0D 95 16
① 16	29.10.2013 12:11:21 245 ms   'LowPrio' (351): RX FIFO<=06 81 08 DC FF 00 6E 4E 0B 0C 5D 0A 0D 1
① 17	29.10.2013 12:11:21 245 ms   'LowPrio' (351): RX FIFO=>06 81 08 DC FF 00 6E 4E 0B 0C 5D 0A 0D
① 18	29.10.2013 12:11:21 245 ms   'LowPrio' (351): IEC 60870-5-103M, linkAddr220, CAA:220=>TYPE:M_SYN_TA_3(006, 0x06), COT:SYN(008, 0x08), Time synchronisation, SQ1, NO1, CAA:220, FC:255, FN:0, LEN:7, DATA:6E 4E 0B 0C 5D 0A 0D 3
① 19	29.10.2013 12:11:21 285 ms   'LowPrio' (351): Class1 TX FIFO<=07 81 09 DC FF 05 02 1
① 20	29.10.2013 12:11:21 285 ms   'LowPrio' (351): IEC 60870-5-103M, linkAddr220, CAA:220<=>TYPE:C_I_GL_NA_3(007, 0x07), COT:IGI(009, 0x09), Initialisation of general interrogation, SQ1, NO1, CAA:220, FC:255, FN:5, LEN:1, DATA:02 3
① 21	29.10.2013 12:11:21 445 ms   'LowPrio' (351): Serial Link(2)PRM.TX<=68 09 09 68 53 DC 07 81 09 DC FF 05 02 A2 16 2
① 22	29.10.2013 12:11:21 485 ms   'LowPrio' (351): Serial Link(2)PRM.RX=>10 20 DC FC 16
① 23	29.10.2013 12:11:21 625 ms   'LowPrio' (351): Class1 TX FIFO=>07 81 09 DC FF 05 02 1
① 24	29.10.2013 12:11:21 985 ms   'LowPrio' (351): Serial Link(2)PRM.TX<=10 7A DC 56 16 2
① 25	29.10.2013 12:11:22 25 ms   'LowPrio' (351): Serial Link(2)PRM.RX=>68 09 09 68 08 DC 08 81 0A DC FF 00 02 54 16
① 26	29.10.2013 12:11:22 165 ms   'LowPrio' (351): RX FIFO<=08 81 0A DC FF 00 02 1

Die Meldungen werden auf unterschiedliche Weise (durch unterschiedliche Parameter) aktiviert. Unten folgt die Auflistung der unterschiedlichen Parameternamen.

1. Hexadezimale Ausgabe der ASDUs (ohne Link-Layer-Control-Header). Pro Zeile werden 32-ASDU-Datenbytes als hexadezimale Zahlen ausgegeben. Längere ASDUs werden auf mehrere Zeilen verteilt:

Name	AN	AUS
ST_IEC870_5_103TBuffer [▶ 547].eDbg	eIEC870_FIFO_DBG_ALL	eIEC870_FIFO_DBG_OFF

2. Hexadezimale Ausgabe der APDUs (RS232/RS485 Telegramme). Pro Zeile werden 32-APDU-Datenbytes als hexadezimale Zahlen ausgegeben. Längere APDUs werden auf mehrere Zeilen verteilt:

Name	AN	AUS
FB_IEC870_5_103TProtocol [▶ 410].bOutDbg	TRUE	FALSE

3. Meldungen, die durch die SPS-Applikation (Beispielprojekt) mit Hilfe des FB\_IEC870\_DebugLogFifo-Funktionsbausteins geloggt werden.

Name	AN	AUS
FB_IEC870_DebugLogFifo [▶ 419]	Aktionsaufruf: A_LogError, A_LogWarning, A_LogHint	-

Weitere Diagnosetools:

- Portmon for Windows (v3.02, Windows Sysinternals);
- Diverse Protokoll-Test-Suit-Produkte;

## 7.4 IEC 60870-5-104

### 7.4.1 TwinCAT IEC 60870-5-104 Fehlercodes

#### Voraussetzungen

Fehlercodes (Hex)	Fehlercodes (Dez)	Fehlerquelle	Beschreibung
0x00000000-0x00007800	0-30720	<u>TwinCAT System Fehlercodes</u> [ <a href="#">▶ 708</a> ]	TwinCAT System Fehler (ADS-Fehlercodes inklusive).
0x00008000-0x000080FF	32768-33023	TF6310 TCP/IP Fehlercodes	TF6310 TCP/IP Fehler.
0x00008100-0x000081FF	33024-33279	<u>TwinCAT IEC 60870-5-10x Fehlercodes</u> [ <a href="#">▶ 557</a> ]	TwinCAT IEC 60870-5-10x Protokollfehler/ Kommunikationsfehler (z.B.: Timeoutfehler, Telegrammfehler, Konfigurationsfehler).
0x80070000-0x8007FFFF	2147942400-2148007935	Higher word = 0x8007 (fix) Lower word = Win32 <u>System Fehlercodes</u> [ <a href="#">▶ 713</a> ]	Win32 Systemfehler (Windows Sockets Fehlercodes inklusive). Der eigentliche Fehlercode befindet sich in den unteren 16 Bit. Die oberen 16 Bit bestimmen nur die Fehlerquelle.

#### 7.4.1.1 ADS Return Codes

Gruppierung der Fehlercodes:

Globale Fehlercodes: [0x0000](#) [[▶ 708](#)]... (0x9811\_0000 ...)

Router Fehlercodes: [0x0500](#) [[▶ 709](#)]... (0x9811\_0500 ...)

Allgemeine ADS Fehler: [0x0700](#) [[▶ 710](#)]... (0x9811\_0700 ...)

RTime Fehlercodes: [0x1000](#) [[▶ 712](#)]... (0x9811\_1000 ...)

#### Globale Fehlercodes

Hex	Dec	HRESULT	Name	Beschreibung
0x0	0	0x98110000	ERR_NOERROR	Kein Fehler.
0x1	1	0x98110001	ERR_INTERNAL	Interner Fehler.
0x2	2	0x98110002	ERR_NORTIME	Keine Echtzeit.
0x3	3	0x98110003	ERR_ALLOCLOCKEDMEM	Zuweisung gesperrt - Speicherfehler.
0x4	4	0x98110004	ERR_INSERTMAILBOX	Postfach voll – Es konnte die ADS Nachricht nicht versendet werden. Reduzieren der Anzahl der ADS Nachrichten pro Zyklus bringt Abhilfe.
0x5	5	0x98110005	ERR_WRONGRECEIVEHMSG	Falsches HMSG.
0x6	6	0x98110006	ERR_TARGETPORTNOTFOUND	Ziel-Port nicht gefunden – ADS Server ist nicht gestartet, nicht erreichbar oder nicht installiert.
0x7	7	0x98110007	ERR_TARGETMACHINENOTFOUND	Zielrechner nicht gefunden – AMS Route wurde nicht gefunden.
0x8	8	0x98110008	ERR_UNKNOWNCMDID	Unbekannte Befehl-ID.
0x9	9	0x98110009	ERR_BADTASKID	Ungültige Task-ID.
0xA	10	0x9811000A	ERR_NOIO	Kein IO.
0xB	11	0x9811000B	ERR_UNKNOWNAMSCMD	Unbekannter AMS-Befehl.
0xC	12	0x9811000C	ERR_WIN32ERROR	Win32 Fehler.
0xD	13	0x9811000D	ERR_PORTNOTCONNECTED	Port nicht verbunden.
0xE	14	0x9811000E	ERR_INVALIDAMSLENGTH	Ungültige AMS-Länge.
0xF	15	0x9811000F	ERR_INVALIDAMSNETID	Ungültige AMS Net ID.
0x10	16	0x98110010	ERR_LOWINSTLEVEL	Installations-Level ist zu niedrig –TwinCAT 2 Lizenzfehler.
0x11	17	0x98110011	ERR_NODEBUGINTAVAILABLE	Kein Debugging verfügbar.
0x12	18	0x98110012	ERR_PORTDISABLED	Port deaktiviert – TwinCAT System Service nicht gestartet.
0x13	19	0x98110013	ERR_PORTALREADYCONNECTED	Port bereits verbunden.
0x14	20	0x98110014	ERR_AMSSYNC_W32ERROR	AMS Sync Win32 Fehler.
0x15	21	0x98110015	ERR_AMSSYNC_TIMEOUT	AMS Sync Timeout.
0x16	22	0x98110016	ERR_AMSSYNC_AMSERROR	AMS Sync Fehler.
0x17	23	0x98110017	ERR_AMSSYNC_NOINDEXINMAP	Keine Index-Map für AMS Sync vorhanden.
0x18	24	0x98110018	ERR_INVALIDAMSPORT	Ungültiger AMS-Port.
0x19	25	0x98110019	ERR_NOMEMORY	Kein Speicher.
0x1A	26	0x9811001A	ERR_TCPSEND	TCP Sendefehler.
0x1B	27	0x9811001B	ERR_HOSTUNREACHABLE	Host nicht erreichbar.
0x1C	28	0x9811001C	ERR_INVALIDAMSFRAGMENT	Ungültiges AMS Fragment.
0x1D	29	0x9811001D	ERR_TLSEND	TLS Sendefehler – Secure ADS Verbindung fehlgeschlagen.
0x1E	30	0x9811001E	ERR_ACCESSDENIED	Zugriff Verweigert – Secure ADS Zugriff verweigert.

**Router Fehlercodes**

Hex	Dec	HRESULT	Name	Beschreibung
0x500	1280	0x98110500	ROUTERERR_NOLOCKEDMEMORY	Lockierter Speicher kann nicht zugewiesen werden.
0x501	1281	0x98110501	ROUTERERR_RESIZEMEMORY	Die Größe des Routerspeichers konnte nicht geändert werden.
0x502	1282	0x98110502	ROUTERERR_MAILBOXFULL	Das Postfach hat die maximale Anzahl der möglichen Meldungen erreicht.
0x503	1283	0x98110503	ROUTERERR_DEBUGBOXFULL	Das Debug Postfach hat die maximale Anzahl der möglichen Meldungen erreicht.
0x504	1284	0x98110504	ROUTERERR_UNKNOWNPORTTYPE	Der Porttyp ist unbekannt.
0x505	1285	0x98110505	ROUTERERR_NOTINITIALIZED	Router ist nicht initialisiert.
0x506	1286	0x98110506	ROUTERERR_PORTALREADYINUSE	Die Portnummer ist bereits vergeben.
0x507	1287	0x98110507	ROUTERERR_NOTREGISTERED	Der Port ist nicht registriert.
0x508	1288	0x98110508	ROUTERERR_NOMOREQUEUES	Die maximale Portanzahl ist erreicht.
0x509	1289	0x98110509	ROUTERERR_INVALIDPORT	Der Port ist ungültig.
0x50A	1290	0x9811050A	ROUTERERR_NOTACTIVATED	Der Router ist nicht aktiv.
0x50B	1291	0x9811050B	ROUTERERR_FRAGMENTBOXFULL	Das Postfach hat die maximale Anzahl für fragmentierte Nachrichten erreicht.
0x50C	1292	0x9811050C	ROUTERERR_FRAGMENTTIMEOUT	Fragment Timeout aufgetreten.
0x50D	1293	0x9811050D	ROUTERERR_TOBEREMOVED	Port wird entfernt.

### Allgemeine ADS Fehlercodes

Hex	Dec	HRESULT	Name	Beschreibung
0x700	1792	0x98110700	ADSERR_DEVICE_ERROR	Allgemeiner Gerätefehler.
0x701	1793	0x98110701	ADSERR_DEVICE_SRVNOTSUPP	Service wird vom Server nicht unterstützt.
0x702	1794	0x98110702	ADSERR_DEVICE_INVALIDGRP	Ungültige Index-Gruppe.
0x703	1795	0x98110703	ADSERR_DEVICE_INVALIDOFFSET	Ungültiger Index-Offset.
0x704	1796	0x98110704	ADSERR_DEVICE_INVALIDACCESS	Lesen oder Schreiben nicht gestattet.
0x705	1797	0x98110705	ADSERR_DEVICE_INVALIDSIZE	Parametergröße nicht korrekt.
0x706	1798	0x98110706	ADSERR_DEVICE_INVALIDDATA	Ungültige Daten-Werte.
0x707	1799	0x98110707	ADSERR_DEVICE_NOTREADY	Gerät nicht betriebsbereit.
0x708	1800	0x98110708	ADSERR_DEVICE_BUSY	Gerät beschäftigt.
0x709	1801	0x98110709	ADSERR_DEVICE_INVALIDCONTEXT	Ungültiger Kontext vom Betriebssystem - Kann durch Verwendung von ADS Bausteinen in unterschiedlichen Tasks auftreten. Abhilfe kann die Multitasking-Synchronisation in der SPS geben.
0x70A	1802	0x9811070A	ADSERR_DEVICE_NOMEMORY	Nicht genügend Speicher.
0x70B	1803	0x9811070B	ADSERR_DEVICE_INVALIDPARM	Ungültige Parameter-Werte.
0x70C	1804	0x9811070C	ADSERR_DEVICE_NOTFOUND	Nicht gefunden (Dateien,...).
0x70D	1805	0x9811070D	ADSERR_DEVICE_SYNTAX	Syntax-Fehler in Datei oder Befehl.
0x70E	1806	0x9811070E	ADSERR_DEVICE_INCOMPATIBLE	Objekte stimmen nicht überein.
0x70F	1807	0x9811070F	ADSERR_DEVICE_EXISTS	Objekt ist bereits vorhanden.
0x710	1808	0x98110710	ADSERR_DEVICE_SYMBOLNOTFOUND	Symbol nicht gefunden.
0x711	1809	0x98110711	ADSERR_DEVICE_SYMBOLVERSIONINVALID	Symbol-Version ungültig – Kann durch einen Online-Change auftreten. Erzeuge einen neuen Handle.
0x712	1810	0x98110712	ADSERR_DEVICE_INVALIDSTATE	Gerät (Server) ist im ungültigen Zustand.
0x713	1811	0x98110713	ADSERR_DEVICE_TRANSMODENOTSUPP	AdsTransMode nicht unterstützt.
0x714	1812	0x98110714	ADSERR_DEVICE_NOTIFYHANDINVALID	Notification Handle ist ungültig.
0x715	1813	0x98110715	ADSERR_DEVICE_CLIENTUNKNOWN	Notification-Client nicht registriert.
0x716	1814	0x98110716	ADSERR_DEVICE_NOMOREHDL	Keine weiteren Handles verfügbar.
0x717	1815	0x98110717	ADSERR_DEVICE_INVALIDWATCHSIZE	Größe der Notification zu groß.
0x718	1816	0x98110718	ADSERR_DEVICE_NOTINIT	Gerät nicht initialisiert.
0x719	1817	0x98110719	ADSERR_DEVICE_TIMEOUT	Gerät hat einen Timeout.
0x71A	1818	0x9811071A	ADSERR_DEVICE_NOINTERFACE	Interface Abfrage fehlgeschlagen.
0x71B	1819	0x9811071B	ADSERR_DEVICE_INVALIDINTERFACE	Falsches Interface angefordert.
0x71C	1820	0x9811071C	ADSERR_DEVICE_INVALIDCLSID	Class-ID ist ungültig.
0x71D	1821	0x9811071D	ADSERR_DEVICE_INVALIDOBJID	Object-ID ist ungültig.
0x71E	1822	0x9811071E	ADSERR_DEVICE_PENDING	Anforderung steht aus.
0x71F	1823	0x9811071F	ADSERR_DEVICE_ABORTED	Anforderung wird abgebrochen.
0x720	1824	0x98110720	ADSERR_DEVICE_WARNING	Signal-Warnung.
0x721	1825	0x98110721	ADSERR_DEVICE_INVALIDARRAYIDX	Ungültiger Array-Index.
0x722	1826	0x98110722	ADSERR_DEVICE_SYMBOLNOTACTIVE	Symbol nicht aktiv.
0x723	1827	0x98110723	ADSERR_DEVICE_ACCESSDENIED	Zugriff verweigert.
0x724	1828	0x98110724	ADSERR_DEVICE_LICENSENOTFOUND	Fehlende Lizenz.
0x725	1829	0x98110725	ADSERR_DEVICE_LICENSEEXPIRED	Lizenz abgelaufen.
0x726	1830	0x98110726	ADSERR_DEVICE_LICENSEEXCEEDED	Lizenz überschritten.
0x727	1831	0x98110727	ADSERR_DEVICE_LICENSEINVALID	Lizenz ungültig.
0x728	1832	0x98110728	ADSERR_DEVICE_LICENSESYSTEMID	Lizenzproblem: System-ID ist ungültig.
0x729	1833	0x98110729	ADSERR_DEVICE_LICENSENOTIMELIMIT	Lizenz nicht zeitlich begrenzt.
0x72A	1834	0x9811072A	ADSERR_DEVICE_LICENSEFUTUREISSUE	Lizenzproblem: Zeitpunkt in der Zukunft.
0x72B	1835	0x9811072B	ADSERR_DEVICE_LICENSETIMETOLONG	Lizenz-Zeitraum zu lang.
0x72C	1836	0x9811072C	ADSERR_DEVICE_EXCEPTION	Exception beim Systemstart.
0x72D	1837	0x9811072D	ADSERR_DEVICE_LICENSEDUPLICATED	Lizenz-Datei zweimal gelesen.
0x72E	1838	0x9811072E	ADSERR_DEVICE_SIGNATUREINVALID	Ungültige Signatur.
0x72F	1839	0x9811072F	ADSERR_DEVICE_CERTIFICATEINVALID	Zertifikat ungültig.
0x730	1840	0x98110730	ADSERR_DEVICE_LICENSEOEMNOTFOUND	Public Key vom OEM nicht bekannt.
0x731	1841	0x98110731	ADSERR_DEVICE_LICENSERESTRICTED	Lizenz nicht gültig für diese System.ID.
0x732	1842	0x98110732	ADSERR_DEVICE_LICENSEDEMODENIED	Demo-Lizenz untersagt.
0x733	1843	0x98110733	ADSERR_DEVICE_INVALIDFNCID	Funktions-ID ungültig.
0x734	1844	0x98110734	ADSERR_DEVICE_OUTOFRANGE	Außerhalb des gültigen Bereiches.
0x735	1845	0x98110735	ADSERR_DEVICE_INVALIDALIGNMENT	Ungültiges Alignment.

Hex	Dec	HRESULT	Name	Beschreibung
0x736	1846	0x98110736	ADSERR_DEVICE_LICENSEPLATFORM	Ungültiger Plattform Level.
0x737	1847	0x98110737	ADSERR_DEVICE_FORWARD_PL	Kontext – Weiterleitung zum Passiv-Level.
0x738	1848	0x98110738	ADSERR_DEVICE_FORWARD_DL	Kontext – Weiterleitung zum Dispatch-Level.
0x739	1849	0x98110739	ADSERR_DEVICE_FORWARD_RT	Kontext – Weiterleitung zur Echtzeit.
0x740	1856	0x98110740	ADSERR_CLIENT_ERROR	Clientfehler.
0x741	1857	0x98110741	ADSERR_CLIENT_INVALIDPARG	Dienst enthält einen ungültigen Parameter.
0x742	1858	0x98110742	ADSERR_CLIENT_LISTEMPTY	Polling-Liste ist leer.
0x743	1859	0x98110743	ADSERR_CLIENT_VARUSED	Var-Verbindung bereits im Einsatz.
0x744	1860	0x98110744	ADSERR_CLIENT_DUPLINVOKEID	Die aufgerufene ID ist bereits in Benutzung.
0x745	1861	0x98110745	ADSERR_CLIENT_SYNC_TIMEOUT	Timeout ist aufgetreten – Die Gegenstelle antwortet nicht im vorgegebenen ADS Timeout. Die Routeneinstellung der Gegenstelle kann falsch konfiguriert sein.
0x746	1862	0x98110746	ADSERR_CLIENT_W32ERROR	Fehler im Win32 Subsystem.
0x747	1863	0x98110747	ADSERR_CLIENT_TIMEOUTINVALID	Ungültiger Client Timeout-Wert.
0x748	1864	0x98110748	ADSERR_CLIENT_PORTNOTOPEN	Port nicht geöffnet.
0x749	1865	0x98110749	ADSERR_CLIENT_NOAMSADDR	Keine AMS Adresse.
0x750	1872	0x98110750	ADSERR_CLIENT_SYNCINTERNAL	Interner Fehler in Ads-Sync.
0x751	1873	0x98110751	ADSERR_CLIENT_ADDHASH	Überlauf der Hash-Tabelle.
0x752	1874	0x98110752	ADSERR_CLIENT_REMOVEHASH	Schlüssel in der Tabelle nicht gefunden.
0x753	1875	0x98110753	ADSERR_CLIENT_NOMORESVM	Keine Symbole im Cache.
0x754	1876	0x98110754	ADSERR_CLIENT_SYNCRESINVALID	Ungültige Antwort erhalten.
0x755	1877	0x98110755	ADSERR_CLIENT_SYNCPORTLOCKED	Sync Port ist verriegelt.
0x756	1878	0x98110756	ADSERR_CLIENT_REQUESTCANCELLED	Die Anfrage wurde abgebrochen.

### RTime Fehlercodes

Hex	Dec	HRESULT	Name	Beschreibung
0x1000	4096	0x98111000	RTERR_INTERNAL	Interner Fehler im Echtzeit-System.
0x1001	4097	0x98111001	RTERR_BADTIMERPERIODS	Timer-Wert nicht gültig.
0x1002	4098	0x98111002	RTERR_INVALIDTASKPTR	Task-Pointer hat den ungültigen Wert 0 (null).
0x1003	4099	0x98111003	RTERR_INVALIDSTACKPTR	Stack-Pointer hat den ungültigen Wert 0 (null).
0x1004	4100	0x98111004	RTERR_PrioEXISTS	Die Request Task Priority ist bereits vergeben.
0x1005	4101	0x98111005	RTERR_NOMORETCB	Kein freier TCB (Task Control Block) verfügbar. Maximale Anzahl von TCBs beträgt 64.
0x1006	4102	0x98111006	RTERR_NOMORESEMAS	Keine freien Semaphoren zur Verfügung. Maximale Anzahl der Semaphoren beträgt 64.
0x1007	4103	0x98111007	RTERR_NOMOREQUEUES	Kein freier Platz in der Warteschlange zur Verfügung. Maximale Anzahl der Plätze in der Warteschlange beträgt 64.
0x100D	4109	0x9811100D	RTERR_EXTIRQALREADYDEF	Ein externer Synchronisations-Interrupt wird bereits angewandt.
0x100E	4110	0x9811100E	RTERR_EXTIRQNOTDEF	Kein externer Sync-Interrupt angewandt.
0x100F	4111	0x9811100F	RTERR_EXTIRQINSTALLFAILED	Anwendung des externen Synchronisierungs-Interrupts ist fehlgeschlagen.
0x1010	4112	0x98111010	RTERR_IRQLNOTLESSOREQUAL	Aufruf einer Service-Funktion im falschen Kontext
0x1017	4119	0x98111017	RTERR_VMXNOTSUPPORTED	Intel VT-x Erweiterung wird nicht unterstützt.
0x1018	4120	0x98111018	RTERR_VMXDISABLED	Intel VT-x Erweiterung ist nicht aktiviert im BIOS.
0x1019	4121	0x98111019	RTERR_VMXCONTROLSMISSING	Fehlende Funktion in Intel VT-x Erweiterung.
0x101A	4122	0x9811101A	RTERR_VMXENABLEFAILS	Aktivieren von Intel VT-x schlägt fehl.

### Spezifische positive HRESULT Return Codes:



HRESULT	Name	Beschreibung
0x0000_0000	S_OK	Kein Fehler.
0x0000_0001	S_FALSE	Kein Fehler. Bsp.: erfolgreiche Abarbeitung, bei der jedoch ein negatives oder unvollständiges Ergebnis erzielt wurde.
0x0000_0203	S_PENDING	Kein Fehler. Bsp.: erfolgreiche Abarbeitung, bei der jedoch noch kein Ergebnis vorliegt.
0x0000_0256	S_WATCHDOG_TIMEOUT	Kein Fehler. Bsp.: erfolgreiche Abarbeitung, bei der jedoch eine Zeitüberschreitung eintrat.

**TCP Winsock-Fehlercodes**

Hex	Dec	Name	Beschreibung
0x274C	10060	WSAETIMEDOUT	Verbindungs Timeout aufgetreten - Fehler beim Herstellen der Verbindung, da die Gegenstelle nach einer bestimmten Zeitspanne nicht ordnungsgemäß reagiert hat, oder die hergestellte Verbindung konnte nicht aufrecht erhalten werden, da der verbundene Host nicht reagiert hat.
0x274D	10061	WSAECONNREFUSED	Verbindung abgelehnt - Es konnte keine Verbindung hergestellt werden, da der Zielcomputer dies explizit abgelehnt hat. Dieser Fehler resultiert normalerweise aus dem Versuch, eine Verbindung mit einem Dienst herzustellen, der auf dem fremden Host inaktiv ist—das heißt, einem Dienst, für den keine Serveranwendung ausgeführt wird.
0x2751	10065	WSAEHOSTUNREACH	Keine Route zum Host - Ein Socketvorgang bezog sich auf einen nicht verfügbaren Host.
Weitere Winsock-Fehlercodes: <a href="#">Win32-Fehlercodes</a> [▶ 713]			

**7.4.1.2 Win32 Error Codes**

Die folgende Tabelle enthält eine Liste der Win32-Fehlercodes.

[0](#) [▶ 714], [100](#) [▶ 716], [200](#) [▶ 719], [1001](#) [▶ 721], [1100](#) [▶ 724], [1200](#) [▶ 727], [1400](#) [▶ 733], [1600](#) [▶ 735], [1800](#) [▶ 740], [2000](#) [▶ 742], [3000](#) [▶ 743], [5000](#) [▶ 746], [6000](#) [▶ 750], [8000](#) [▶ 752], [8500](#) [▶ 762], [9001](#) [▶ 767], [10004](#) [▶ 769], [12000](#) [▶ 772],

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
0	0x00000000	ERROR_SUCCESS	Der Vorgang wurde erfolgreich abgeschlossen.
1	0x00000001	ERROR_INVALID_FUNCTION	Unzulässige Funktion.
2	0x00000002	ERROR_FILE_NOT_FOUND	Das System kann die angegebene Datei nicht finden.
3	0x00000003	ERROR_PATH_NOT_FOUND	Das System kann den angegebenen Pfad nicht finden.
4	0x00000004	ERROR_TOO_MANY_OPEN_FILES	Das System kann die Datei nicht öffnen.
5	0x00000005	ERROR_ACCESS_DENIED	Zugriff wird verweigert.
6	0x00000006	ERROR_INVALID_HANDLE	Das Handle ist ungültig.
7	0x00000007	ERROR_ARENA_TRASHED	Die Speicher-Kontrollblöcke wurden zerstört.
8	0x00000008	ERROR_NOT_ENOUGH_MEMORY	Es ist nicht genügend Speicher verfügbar, um diesen Befehl zu verarbeiten.
9	0x00000009	ERROR_INVALID_BLOCK	Die Speicher-Blockadresse ist ungültig.
10	0x0000000A	ERROR_BAD_ENVIRONMENT	Die Umgebung ist nicht korrekt.
11	0x0000000B	ERROR_BAD_FORMAT	Es wurde versucht, ein Programm mit einem falschen Format zu laden.
12	0x0000000C	ERROR_INVALID_ACCESS	Der Zugangscode ist ungültig.
13	0x0000000D	ERROR_INVALID_DATA	Die Daten sind ungültig.
14	0x0000000E	ERROR_OUTOFMEMORY	Nicht genügend Speicherplatz ist zum Abschließen dieses Vorgangs verfügbar.
15	0x0000000F	ERROR_INVALID_DRIVE	Das System kann das angegebene Laufwerk nicht finden.
16	0x00000010	ERROR_CURRENT_DIRECTORY	Das Verzeichnis kann nicht entfernt werden.
17	0x00000011	ERROR_NOT_SAME_DEVICE	Das System kann nicht die Datei in ein anderes Laufwerk verschieben.
18	0x00000012	ERROR_NO_MORE_FILES	Es gibt keine weiteren Dateien.
19	0x00000013	ERROR_WRITE_PROTECT	Das Medium ist schreibgeschützt.
20	0x00000014	ERROR_BAD_UNIT	Das System kann das angegebene Gerät nicht finden.
21	0x00000015	ERROR_NOT_READY	Das Gerät ist nicht bereit.
22	0x00000016	ERROR_BAD_COMMAND	Das Gerät erkennt den Befehl nicht.
23	0x00000017	ERROR_CRC	Datenfehler (CRC-Prüfung).
24	0x00000018	ERROR_BAD_LENGTH	Das Programm hat einen Befehl ausgegeben, aber die Befehlslänge ist falsch.
25	0x00000019	ERROR_SEEK	Das Laufwerk kann nicht suchen Sie einen bestimmten Bereich oder auf der Festplatte zu verfolgen.
26	0x0000001A	ERROR_NOT_DOS_DISK	Die angegebene Festplatte oder Diskette kann nicht zugegriffen werden.
27	0x0000001B	ERROR_SECTOR_NOT_FOUND	Das Laufwerk kann den angeforderten Sektor nicht finden.
28	0x0000001C	ERROR_OUT_OF_PAPER	Der Drucker hat kein Papier.
29	0x0000001D	ERROR_WRITE_FAULT	Das System kann nicht auf das angegebene Gerät schreiben.
30	0x0000001E	ERROR_READ_FAULT	Das System kann nicht vom angegebenen Gerät lesen.
31	0x0000001F	ERROR_GEN_FAILURE	Ein an das System angeschlossenes Gerät funktioniert nicht.
32	0x00000020	ERROR_SHARING_VIOLATION	Der Prozess kann nicht die Datei zugreifen, da es von einem anderen Prozess verwendet wird.
33	0x00000021	ERROR_LOCK_VIOLATION	Der Prozess kann nicht die Datei zugreifen, da ein anderer Prozess einen Teil der Datei gesperrt hat.
34	0x00000022	ERROR_WRONG_DISK	Die falsche Diskette befindet sich im Laufwerk. Fügen Sie %2 (Seriennummer des Datenträgers: %3) in das Laufwerk %1 ein.
36	0x00000024	ERROR_SHARING_BUFFER_EXCEEDED	Zu viele Dateien, die für die gemeinsame Nutzung geöffnet.
38	0x00000026	ERROR_HANDLE_EOF	Das Ende der Datei wurde erreicht.
39	0x00000027	ERROR_HANDLE_DISK_FULL	Der Datenträger ist voll.
50	0x00000032	ERROR_NOT_SUPPORTED	Die Anforderung wird nicht unterstützt.
51	0x00000033	ERROR_REM_NOT_LIST	Der Remotecomputer ist nicht verfügbar.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
52	0x00000034	ERROR_DUP_NAME	Ein doppelter Name ist im Netzwerk vorhanden.
53	0x00000035	ERROR_BAD_NETPATH	Der Netzwerkpfad wurde nicht gefunden.
54	0x00000036	ERROR_NETWORK_BUSY	Das Netzwerk ist ausgelastet.
55	0x00000037	ERROR_DEV_NOT_EXIST	Die angegebene Netzwerkressource oder das Gerät ist nicht mehr verfügbar.
56	0x00000038	ERROR_TOO_MANY_CMDS	Das Netzwerk-BIOS-Befehl-Limit wurde erreicht.
57	0x00000039	ERROR_ADAP_HDW_ERR	Ein Netzwerk-Adapter-Hardware-Fehler aufgetreten.
58	0x0000003A	ERROR_BAD_NET_RESP	Der angegebene Server kann nicht den angeforderten Vorgang ausführen.
59	0x0000003B	ERROR_UNEXP_NET_ERR	Ein unerwarteter Netzwerkfehler ist aufgetreten.
60	0x0000003C	ERROR_BAD_REM_ADAP	Der Remoteadapter ist nicht kompatibel.
61	0x0000003D	ERROR_PRINTQ_FULL	Die Druckerwarteschlange ist voll.
62	0x0000003E	ERROR_NO_SPOOL_SPACE	Speicherplatz zum Speichern der Datei, die darauf warten, gedruckt werden, ist nicht auf dem Server verfügbar.
63	0x0000003F	ERROR_PRINT_CANCELLED	Ihre zum Drucken, wartende Datei wurde gelöscht.
64	0x00000040	ERROR_NETNAME_DELETED	Der angegebene Netzwerkname ist nicht mehr verfügbar.
65	0x00000041	ERROR_NETWORK_ACCESS_DENIED	Netzwerkzugriff wurde verweigert.
66	0x00000042	ERROR_BAD_DEV_TYPE	Der Netzwerk-Ressource-Typ ist nicht korrekt.
67	0x00000043	ERROR_BAD_NET_NAME	Der Netzwerkname kann nicht gefunden werden.
68	0x00000044	ERROR_TOO_MANY_NAMES	Die Namensbegrenzung für den lokalen Computer Netzwerkarte wurde überschritten.
69	0x00000045	ERROR_TOO_MANY_SESS	Die Netzwerk-BIOS-Sitzungsbegrenzung wurde überschritten.
70	0x00000046	ERROR_SHARING_PAUSED	Der Remoteserver wurde angehalten wurde oder wird gerade gestartet wird.
71	0x00000047	ERROR_REQ_NOT_ACCEP	Keine weitere Verbindungen können zu dieser Zeit mit diesem Remotecomputer erfolgen, denn es gibt bereits so viele Verbindungen wie der Computer annehmen kann.
72	0x00000048	ERROR_REDIR_PAUSED	Der angegebene Drucker oder das angegebene Datenträgergerät wurde angehalten.
80	0x00000050	ERROR_FILE_EXISTS	Die Datei ist vorhanden.
82	0x00000052	ERROR_CANNOT_MAKE	Das Verzeichnis oder die Datei kann nicht erstellt werden.
83	0x00000053	ERROR_FAIL_I24	Fehler auf INT 24.
84	0x00000054	ERROR_OUT_OF_STRUCTURES	Speicher zur Verarbeitung dieser Anforderung ist nicht verfügbar.
85	0x00000055	ERROR_ALREADY_ASSIGNED	Der lokale Gerätenamen wird bereits verwendet.
86	0x00000056	ERROR_INVALID_PASSWORD	Das angegebene Netzwerkennwort ist falsch.
87	0x00000057	ERROR_INVALID_PARAMETER	Die Parameter ist falsch.
88	0x00000058	ERROR_NET_WRITE_FAULT	Ein Schreibfehler ist aufgetreten im Netzwerk.
89	0x00000059	ERROR_NO_PROC_SLOTS	Das System kann zur Zeit keinen weiteren Prozess starten.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
100	0x00000064	ERROR_TOO_MANY_SEMAPHORES	Ein weiteres System Semaphore kann nicht erstellt werden.
101	0 x 00000065	ERROR_EXCL_SEM_ALREADY_OWNED	Die exklusive Semaphore gehört einem anderen Prozess.
102	0x00000066	ERROR_SEM_IS_SET	Das Semaphor ist festgelegt und kann nicht geschlossen werden.
103	0x00000067	ERROR_TOO_MANY_SEM_REQUESTS	Die Semaphore kann nicht erneut festgelegt werden.
104	0x00000068	ERROR_INVALID_AT_INTERRUPT_TIME	Keine anfordern exklusive Semaphore Interrupt Zeitpunkt.
105	0x00000069	ERROR_SEM_OWNER_DIED	Das frühere Eigentum an dieser Semaphore ist beendet.
106	0x0000006A	ERROR_SEM_USER_LIMIT	Legen Sie die Diskette für Laufwerk %1.
107	0x0000006B	ERROR_DISK_CHANGE	Das Programm beendet, da eine Alternative Diskette nicht eingelegt wurde.
108	0x0000006C	ERROR_DRIVE_LOCKED	Der Datenträger wird verwendet oder von einem anderen Prozess gesperrt.
109	0x0000006D	ERROR_BROKEN_PIPE	Die Pipe wurde beendet.
110	0x0000006E	ERROR_OPEN_FAILED	Das System kann das Gerät oder die angegebene Datei nicht öffnen.
111	0x0000006F	ERROR_BUFFER_OVERFLOW	Der Dateiname ist zu lang.
112	0x00000070	ERROR_DISK_FULL	Es ist nicht genügend Speicherplatz auf dem Datenträger.
113	0x00000071	ERROR_NO_MORE_SEARCH_HANDLES	Nicht mehr interne Datei-IDs zur Verfügung.
114	0x00000072	ERROR_INVALID_TARGET_HANDLE	Der interne Dateibezeichner Ziel ist falsch.
117	0x00000075	ERROR_INVALID_CATEGORY	Der vom Anwendungsprogramm IOCTL-Aufruf ist nicht korrekt.
118	0x00000076	ERROR_INVALID_VERIFY_SWITCH	Der Wert des Parameters überprüfen-on-Write Switch ist nicht korrekt.
119	0x00000077	ERROR_BAD_DRIVER_LEVEL	Das System unterstützt den angeforderten Befehl nicht.
120	0x00000078	ERROR_CALL_NOT_IMPLEMENTED	Diese Funktion ist auf diesem System nicht unterstützt.
121	0x00000079	ERROR_SEM_TIMEOUT	Das Zeitlimit für die Semaphore ist abgelaufen.
122	0x0000007A	ERROR_INSUFFICIENT_BUFFER	An einen Systemaufruf übergebene Datenbereich ist zu klein.
123	0x0000007B	ERROR_INVALID_NAME	Der Dateiname, Verzeichnisname oder Datenträgerbezeichnung ist falsch.
124	0x0000007C	ERROR_INVALID_LEVEL	Die Systemaufrufebene ist nicht korrekt.
125	0x0000007D	ERROR_NO_VOLUME_LABEL	Der Datenträger hat keine Datenträgerbezeichnung.
126	0x0000007E	ERROR_MOD_NOT_FOUND	Das angegebene Modul konnte nicht gefunden werden.
127	0x0000007F	ERROR_PROC_NOT_FOUND	Die angegebene Prozedur konnte nicht gefunden werden.
128	0x00000080	ERROR_WAIT_NO_CHILDREN	Es gibt keine untergeordneten Prozesse warten.
129	0x00000081	ERROR_CHILD_NOT_COMPLETE	Die Anwendung %1 kann nicht im Win32-Modus ausgeführt werden.
130	0x00000082	ERROR_DIRECT_ACCESS_HANDLE	Versuch, ein Dateihandle einer offenen Datenträgerpartition für einen anderen Vorgang als raw Disk i/o zu verwenden.
131	0x00000083	ERROR_NEGATIVE_SEEK	Wurde versucht, den Dateizeiger vor den Anfang der Datei zu bewegen.
132	0x00000084	ERROR_SEEK_ON_DEVICE	Der Dateizeiger kann auf dem angegebenen Gerät oder Datei festgelegt werden.
133	0x00000085	ERROR_IS_JOIN_TARGET	Ein Join- oder SUBST-Befehl kann nicht für ein Laufwerk verwendet werden, die zuvor verbundene Laufwerke enthält.
134	0x00000086	ERROR_IS_JOINED	Wurde versucht, einen Join- oder SUBST-Befehl auf einem Laufwerk zu verwenden, die bereits verbunden worden ist.
135	0x00000087	ERROR_IS_SUBSTED	Es wurde versucht, einen Join- oder SUBST-Befehl auf einem Laufwerk zu verwenden, die bereits ersetzt worden ist.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
136	0x00000088	ERROR_NOT_JOINED	Das System versucht, die Verknüpfung eines Laufwerks zu löschen, die nicht Mitglied ist.
137	0x00000089	ERROR_NOT_SUBSTED	Das System versucht, die Substitution eines Laufwerks zu löschen, die nicht ersetzt wird.
138	0x0000008A	ERROR_JOIN_TO_JOIN	Das System versucht, ein Laufwerk zu einem Verzeichnis auf einem verknüpften Laufwerk anzuschließen.
139	0x0000008B	ERROR_SUBST_TO_SUBST	Das System versucht, ein Laufwerk zu einem Verzeichnis auf einem subst Laufwerk zu ersetzen.
140	0x0000008C	ERROR_JOIN_TO_SUBST	Das System versucht, mit dem Auto in ein Verzeichnis auf einem subst Laufwerk anzuschließen.
141	0x0000008D	ERROR_SUBST_TO_JOIN	Das System versucht, SUBST ein Laufwerk zu einem Verzeichnis auf einem verknüpften Laufwerk.
142	0x0000008E	ERROR_BUSY_DRIVE	Das System kann nicht zu dieser Zeit ein Join- oder SUBST ausführen.
143	0x0000008F	ERROR_SAME_DRIVE	Das System kann nicht join oder ersetzen Sie mit dem Auto, oder für ein Verzeichnis auf dem gleichen Laufwerk.
144	0x00000090	ERROR_DIR_NOT_ROOT	Das Verzeichnis ist kein Unterverzeichnis des Stammverzeichnisses.
145	0x00000091	ERROR_DIR_NOT_EMPTY	Das Verzeichnis ist nicht leer.
146	0x00000092	ERROR_IS_SUBST_PATH	Der angegebene Pfad wird in Ersatz verwendet.
147	0x00000093	ERROR_IS_JOIN_PATH	Nicht genügend Ressourcen verfügbar, um diesen Befehl zu verarbeiten.
148	0x00000094	ERROR_PATH_BUSY	Der angegebene Pfad kann zu diesem Zeitpunkt verwendet werden.
149	0x00000095	ERROR_IS_SUBST_TARGET	Es wurde versucht, beitreten oder für die ein Verzeichnis auf dem Laufwerk das Ziel des vorherigen Ersatz ist mit dem Auto zu ersetzen.
150	0x00000096	ERROR_SYSTEM_TRACE	System-Trace-Informationen wurde in der CONFIG nicht angegeben.SYS-Datei, oder die Ablaufverfolgung ist nicht erlaubt.
151	0x00000097	ERROR_INVALID_EVENT_COUNT	Die Anzahl der angegebenen Semaphore-Ereignisse für DosMuxSemWait ist nicht korrekt.
152	0x00000098	ERROR_TOO_MANY_MUXWAITERS	DosMuxSemWait wurde nicht ausgeführt; zu viele Semaphoren sind bereits festgelegt.
153	0x00000099	ERROR_INVALID_LIST_FORMAT	Die DosMuxSemWait-Liste ist nicht korrekt.
154	0x0000009A	ERROR_LABEL_TOO_LONG	Die eingegebene Datenträgerbezeichnung übersteigt die Etikett-Zeichen-Grenze des Dateisystems Ziel.
155	0x0000009B	ERROR_TOO_MANY_TCBS	Ein anderer Thread kann nicht erstellt werden.
156	0x0000009C	ERROR_SIGNAL_REFUSED	Der Empfängerprozess hat das Signal abgelehnt.
157	0x0000009D	ERROR_DISCARDED	Das Segment ist bereits verworfen und kann nicht gesperrt werden.
158	0x0000009E	ERROR_NOT_LOCKED	Das Segment ist bereits freigeschaltet.
159	0x0000009F	ERROR_BAD_THREADID_ADDR	Die Adresse für die Thread-ID ist nicht korrekt.
160	0x000000A0	ERROR_BAD_ARGUMENTS	Die an DosExecPgm übergebene Argumentzeichenfolge ist falsch.
161	0x000000A1	ERROR_BAD_PATHNAME	Der angegebene Pfad ist ungültig.
162	0x000000A2	ERROR_SIGNAL_PENDING	Ein Signal ist bereits anhängig.
164	0x000000A4	ERROR_MAX_THRDS_REACHED	Keine weitere Threads können im System erstellt werden.
167	0x000000A7	ERROR_LOCK_FAILED	Nicht in der Lage, einen Bereich einer Datei zu sperren.
170	0x000000AA	ERROR_BUSY	Die angeforderte Ressource ist in Verwendung.
173	0x000000AD	ERROR_CANCEL_VIOLATION	Eine Sperranforderung war nicht hervorragend für die mitgelieferten "Abbrechen"-Region.
174	0x000000AE	ERROR_ATOMIC_LOCKS_NOT_SUPPORTED	Das Dateisystem unterstützt keine atomare Änderungen an der Sperrentyp.
180	0x000000B4	ERROR_INVALID_SEGMENT_NUMBER	Erkennung des Systems durch eine Segment-Zahl, die nicht korrekt war.
182	0x000000B6	ERROR_INVALID_ORDINAL	Das Betriebssystem kann nicht %1 ausgeführt.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
183	0x000000B7	ERROR_ALREADY_EXISTS	Eine Datei kann nicht erstellt werden, wenn die Datei bereits vorhanden ist.
186	0x000000BA	ERROR_INVALID_FLAG_NUMBER	Das Flag übergeben ist nicht korrekt.
187	0x000000BB	ERROR_SEM_NOT_FOUND	Der Namen des angegebenen Semaphore wurde nicht gefunden.
188	0x000000BC	ERROR_INVALID_STARTING_CODESEG	Das Betriebssystem kann nicht %1 ausgeführt.
189	0x000000BD	ERROR_INVALID_STACKSEG	Das Betriebssystem kann nicht %1 ausgeführt.
190	0x000000BE	ERROR_INVALID_MODULETYPE	Das Betriebssystem kann nicht %1 ausgeführt.
191	0x000000BF	ERROR_INVALID_EXE_SIGNATURE	%1 Kann nicht im Win32-Modus ausgeführt werden.
192	0x000000C0	ERROR_EXE_MARKED_INVALID	Das Betriebssystem kann nicht %1 ausgeführt.
193	0x000000C1	ERROR_BAD_EXE_FORMAT	%1 ist keine zulässige Win32-Anwendung.
194	0x000000C2	ERROR_ITERATED_DATA_EXCEEDS_64k	Das Betriebssystem kann nicht %1 ausgeführt.
195	0x000000C3	ERROR_INVALID_MINALLOCSIZE	Das Betriebssystem kann nicht %1 ausgeführt.
196	0x000000C4	ERROR_DYNLINK_FROM_INVALID_RING	Das Betriebssystem kann nicht dieses Anwendungsprogramm ausgeführt.
197	0x000000C5	ERROR_IOPL_NOT_ENABLED	Das Betriebssystem ist momentan nicht konfiguriert um diese Anwendung auszuführen.
198	0x000000C6	ERROR_INVALID_SEGDPL	Das Betriebssystem kann nicht %1 ausgeführt.
199	0x000000C7	ERROR_AUTODATASEG_EXCEEDS_64k	Das Betriebssystem kann nicht dieses Anwendungsprogramm ausgeführt.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
200	0x000000C8	ERROR_RING2SEG_MUST_BE_MOVABLE	Das Code-Segment kann nicht größer als oder gleich 64 KB sein.
201	0x000000c9 DRIVER_VERIFIER_IOMANAGER_VIOLATION	ERROR_RELOC_CHAIN_XEEDS_SEGLIM	Das Betriebssystem kann nicht %1 ausgeführt.
202	0x000000CA	ERROR_INFLOOP_IN_RELOC_CHAIN	Das Betriebssystem kann nicht %1 ausgeführt.
203	0x000000CB	ERROR_ENVVAR_NOT_FOUND	Das System nicht die Umgebungsoption gefunden, die eingegeben wurde.
205	0x000000CD	ERROR_NO_SIGNAL_SENT	Kein Prozess in der Unterstruktur Befehl hat einen Signalhandler.
206	0x000000CE	ERROR_FILENAME_EXCED_RANGE	Der Dateiname oder die Erweiterung ist zu lang.
207	0x000000CF	ERROR_RING2_STACK_IN_USE	Der Ring 2-Stapel wird verwendet.
208	0x000000D0	ERROR_META_EXPANSION_TOO_LONG	Die globale Dateinamenszeichen * oder?, sind falsch eingegeben oder es wurden zu viele globale Dateinamenszeichen angegeben.
209	0x000000D1	ERROR_INVALID_SIGNAL_NUMBER	Das Signal, das gesendet wird, ist nicht korrekt.
210	0x000000D2	ERROR_THREAD_1_INACTIVE	Der Signalhandler kann nicht festgelegt werden.
212	0x000000D4	ERROR_LOCKED	Das Segment ist gesperrt und kann nicht neu zugewiesen werden.
214	0x000000D6	ERROR_TOO_MANY_MODULES	An dieses Programm oder Dynamic Link Modul sind zu viele Dynamic Link Module angefügt.
215	0x000000D7	ERROR_NESTING_NOT_ALLOWED	Aufrufe von LoadModule können nicht verschachtelt werden.
216	0x000000D8	ERROR_EXE_MACHINE_TYPE_MISMATCH	Die Image-Datei "% 1" ist gültig, aber ist für einen Maschinentyp als dem aktuellen Computer.
230	0x000000E6	ERROR_BAD_PIPE	Der Rohr-Status ist ungültig.
231	0x000000E7	ERROR_PIPE_BUSY	Alle Pipe-Instanzen sind beschäftigt.
232	0x000000E8	ERROR_NO_DATA	Das Rohr ist geschlossen.
233	0x000000E9	ERROR_PIPE_NOT_CONNECTED	Kein Prozess ist am anderen Ende der Leitung.
234	0x000000EA	ERROR_MORE_DATA	Weitere Daten sind verfügbar.
240	0x000000F0	ERROR_VC_DISCONNECTED	Die Sitzung wurde abgebrochen.
254	0x000000FE	ERROR_INVALID_EA_NAME	Das angegebene erweiterte Attribut-Name ist ungültig.
255	0x000000FF	ERROR_EA_LIST_INCONSISTENT	Die erweiterten Attribute sind inkonsistent.
258	0x00000102	WAIT_TIMEOUT	Der Wartevorgang Zeitüberschreitung.
259	0x00000103	ERROR_NO_MORE_ITEMS	Keine weiteren Daten ist verfügbar.
266	0x0000010A	ERROR_CANNOT_COPY	Die Kopierfunktionen können nicht verwendet werden.
267	0x0000010B	ERROR_DIRECTORY	Der Verzeichnisname ist ungültig.
275	0x00000113	ERROR_EAS_DIDNT_FIT	Die erweiterten Attribute passten nicht in den Puffer.
276	0x00000114	ERROR_EA_FILE_CORRUPT	Die erweiterte-Attribute-Datei auf dem System bereitgestellte Datei ist beschädigt.
277	0x00000115	ERROR_EA_TABLE_FULL	Die erweiterte-Attribute-Datei ist voll.
278	0x00000116	ERROR_INVALID_EA_HANDLE	Das angegebene erweiterte Attribut Handle ist ungültig.
282	0x0000011A	ERROR_EAS_NOT_SUPPORTED	Das bereitgestellte Dateisystem unterstützt keine erweiterten Attribute.
288	0x00000120	ERROR_NOT_OWNER	Versuchen Sie, nicht im Besitz des Aufrufers Mutex freizugeben.
298	0x0000012A	ERROR_TOO_MANY_POSTS	Zu viele Beiträge wurden auf einen Semaphore.
299	0x0000012B	ERROR_PARTIAL_COPY	Nur ein Teil der ReadProcessMemory- oder WriteProcessMemory-Anforderung wurde abgeschlossen.
300	0x0000012C	ERROR_OPLOCK_NOT_GRANTED	Die Oplock-Anforderung wurde verweigert.
301	0x0000012D	ERROR_INVALID_OPLOCK_PROTOCOL	Eine ungültige Oplock-Bestätigung wurde vom System empfangen.
302	0x0000012E	ERROR_DISK_TOO_FRAGMENTED	Das Volume ist zu fragmentiert, um diesen Vorgang abzuschließen.
303	0x0000012F	ERROR_DELETE_PENDING	Die Datei kann nicht geöffnet werden, da es wird gerade gelöscht wird.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
317	0x0000013D	ERROR_MR_MID_NOT_FOUND	Das System kann nicht finden Meldungstext für die Meldungsnummer 0 x %1 in der Meldungsdatei für %2.
487	0x000001E7	ERROR_INVALID_ADDRESS	Versuch, ungültige Adresse zuzugreifen.
534	0x00000216	ERROR_ARITHMETIC_OVERFLOW	Arithmetisches Ergebnis übertraf 32 Bits.
535	0x00000217	ERROR_PIPE_CONNECTED	Es ist ein Prozess am anderen Ende der Pipe.
536	0x00000218	ERROR_PIPE_LISTENING	Ich warte auf ein Prozess das andere Ende der Pipe zu öffnen.
994	0x000003E2	ERROR_EA_ACCESS_DENIED	Zugriff auf die erweiterte-Attribute wurde verweigert.
995	0x000003E3	ERROR_OPERATION_ABORTED	Die i/o-Operation wurde wegen eines Threadendes oder einer Anwendungsanforderung abgebrochen.
996	0x000003E4	ERROR_IO_INCOMPLETE	Überlappendes I/O Ereignis befindet sich nicht in einem signalisierten Zustand.
997	0x000003E5	ERROR_IO_PENDING	Überlappenden i/o-Operation ist im Gange.
998	0x000003E6	ERROR_NOACCESS	Ungültiger Zugriff auf Speicherbereich.
999	0x000003E7	ERROR_SWAPERROR	Fehler beim Speicherseite.



Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
1001	0x000003E9	ERROR_STACK_OVERFLOW	Rekursion zu tief; der Stapel ist übergelaufen.
1002	0x000003EA	ERROR_INVALID_MESSAGE	Das Fenster kann nicht auf die gesendete Nachricht fungieren.
1003	0x000003EB	ERROR_CAN_NOT_COMPLETE	Diese Funktion kann nicht abgeschlossen werden.
1004	0x000003EC	ERROR_INVALID_FLAGS	Ungültige Flags.
1005	0x000003ED	ERROR_UNRECOGNIZED_VOLUME	Das Volumen ist keine anerkannte Dateisystem enthalten. Bitte achten Sie darauf, dass alle erforderliche Datei-System-Treiber geladen sind und dass die Lautstärke nicht beschädigt ist.
1006	0x000003EE	ERROR_FILE_INVALID	Das Volumen für eine Datei wurde extern geändert, so dass die geöffnete Datei nicht mehr gültig ist.
1007	0x000003EF	ERROR_FULLSCREEN_MODE	Der angeforderte Vorgang kann nicht im Vollbild Modus ausgeführt werden.
1008	0x000003F0	ERROR_NO_TOKEN	Es wurde versucht, auf ein Token zu verweisen, die nicht vorhanden ist.
1009	0x000003F1	ERROR_BADDB	Die Konfiguration-Registry-Datenbank ist beschädigt.
1010	0x000003F2	ERROR_BADKEY	Der Schlüssel der Konfigurationsregistrierung ist ungültig.
1011	0x000003F3	ERROR_CANTOPEN	Der Schlüssel der Konfigurationsregistrierung konnte nicht geöffnet werden.
1012	0x000003F4	ERROR_CANTREAD	Der Schlüssel der Konfigurationsregistrierung konnte nicht gelesen werden.
1013	0x000003F5	ERROR_CANTWRITE	Der Schlüssel der Konfigurationsregistrierung konnte nicht geschrieben werden.
1014	0x000003F6	ERROR_REGISTRY_RECOVERED	Eine der Dateien in der Registrierungsdatenbank musste durch Verwendung von ein Protokoll oder eine Sicherungskopie wiederhergestellt werden. Die Wiederherstellung war erfolgreich.
1015	0x000003F7	ERROR_REGISTRY_CORRUPT	Die Registrierung ist beschädigt. Die Struktur einer der Dateien, die Registrierungsdaten enthält, ist beschädigt, oder Speicherabzug der Datei beschädigt ist oder die Datei konnte nicht wiederhergestellt werden, da die Sicherungskopie oder das Protokoll fehlte oder beschädigt.
1016	0x000003F8	ERROR_REGISTRY_IO_FAILED	Eine i/o-Operation durch die Registrierung ausgelöst, ist fehlgeschlagen. Die Registrierung konnte nicht lesen, oder ausschreiben, oder flush, eine der Dateien, die das Systemabbild der Registrierung enthalten.
1017	0x000003F9	ERROR_NOT_REGISTRY_FILE	Das System hat versucht, zu laden oder eine Datei in die Registrierung wiederherstellen, aber die angegebene Datei ist nicht in einem Registrierungs-Dateiformat.
1018	0x000003FA	ERROR_KEY_DELETED	Unzulässiger Vorgang auf einen Registrierungsschlüssel, der zum Löschen markiert wurde versucht.
1019	0x000003FB	ERROR_NO_LOG_SPACE	Das System konnte den benötigten Speicherplatz in einem Registrierungsprotokoll nicht reservieren.
1020	0x000003FC	ERROR_KEY_HAS_CHILDREN	Eine symbolische Verknüpfung kann nicht in einem Registrierungsschlüssel erstellt werden, die bereits Unterschlüssel oder Werte hat.
1021	0x000003FD	ERROR_CHILD_MUST_BE_VOLATILE	Einen dauerhaften Unterschlüssel unter einem temporären übergeordneten Schlüssel kann nicht erstellt werden.
1022	0x000003FE	ERROR_NOTIFY_ENUM_DIR	Eine Anforderung einer Änderungsnachricht ist abgeschlossen, und die Informationen wird nicht im Puffer des Aufrufers zurückgegeben. Der Aufrufer muss jetzt die Dateien um die Änderungen zu finden auflisten.
1051	0x0000041B	ERROR_DEPENDENT_SERVICES_RUNNING	Ein Stopzeichen wurde an einen Dienst gesendet, die andere Dienste abhängen.
1052	0x0000041C	ERROR_INVALID_SERVICE_CONTROL	Das angeforderte Steuerelement ist nicht gültig für diesen Dienst.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
1053	0x0000041D	ERROR_SERVICE_REQUEST_TIMEOUT	Der Dienst hat nicht rechtzeitig auf die Start- oder reagiert.
1054	0x0000041E	ERROR_SERVICE_NO_THREAD	Ein Thread konnte nicht für den Dienst erstellt werden.
1055	0x0000041F	ERROR_SERVICE_DATABASE_LOCKED	Die Dienstdatenbank ist gesperrt.
1056	0x00000420	ERROR_SERVICE_ALREADY_RUNNING	Bereits ist eine Instanz des Dienstes ausgeführt.
1057	0x00000421	ERROR_INVALID_SERVICE_ACCOUNT	Der Kontoname ist ungültig oder nicht vorhanden, oder das Kennwort ist ungültig für den Kontonamen angegeben.
1058	0x00000422	ERROR_SERVICE_DISABLED	Der Dienst kann nicht gestartet werden, weil es deaktiviert ist oder weil sie keine aktivierten Geräte zugeordnet hat.
1059	0x00000423	ERROR_CIRCULAR_DEPENDENCY	Kreisförmige Dienstabhängigkeit wurde angegeben.
1060	0x00000424	ERROR_SERVICE_DOES_NOT_EXIST	Der angegebene Dienst ist nicht als installierter Dienst vorhanden.
1061	0x00000425	ERROR_SERVICE_CANNOT_ACCEPT_CTRL	Der Dienst kann nicht diesmal Steuerungsmeldungen annehmen.
1062	0x00000426	ERROR_SERVICE_NOT_ACTIVE	Der Dienst wurde nicht gestartet.
1063	0x00000427	ERROR_FAILED_SERVICE_CONTROLLER_CONNECT	Der Dienstprozess konnte keine Verbindung den Dienstcontroller herstellen.
1064	0x00000428	ERROR_EXCEPTION_IN_SERVICE	Eine Ausnahme ist in den Dienst beim Umgang mit der Anforderung.
1065	0x00000429	ERROR_DATABASE_DOES_NOT_EXIST	Die angegebene Datenbank existiert nicht.
1066	0x0000042A	ERROR_SERVICE_SPECIFIC_ERROR	Der Dienst hat einen dienstspezifischen Fehlercode zurückgegeben.
1067	0x0000042B	ERROR_PROCESS_ABORTED	Der Prozess wurde unerwartet beendet.
1068	0x0000042C	ERROR_SERVICE_DEPENDENCY_FAIL	Der Abhängigkeitsdienst oder die Gruppe konnte nicht gestartet werden.
1069	0x0000042D	ERROR_SERVICE_LOGON_FAILED	Der Dienst wurde aufgrund eines Anmeldefehlers nicht gestartet.
1070	0x0000042E	ERROR_SERVICE_START_HANG	Nach dem Start hing der Dienst einen Startausstehende Zustand.
1071	0x0000042F	ERROR_INVALID_SERVICE_LOCK	Die angegebene Sperre für die Datenbank ist ungültig.
1072	0x00000430	ERROR_SERVICE_MARKED_FOR_DELETE	Der angegebene Dienst wurde zum Löschen markiert.
1073	0x00000431	ERROR_SERVICE_EXISTS	Der angegebene Dienst ist bereits vorhanden.
1074	0x00000432	ERROR_ALREADY_RUNNING_LKG	Das System wird derzeit mit der letzten funktionierenden Konfiguration ausgeführt.
1075	0x00000433	ERROR_SERVICE_DEPENDENCY_DELETED	Der Abhängigkeitsdienst existiert nicht oder wurde zum Löschen markiert.
1076	0x00000434	ERROR_BOOT_ALREADY_ACCEPTED	Die aktuelle Startkonfiguration wurde bereits für den Einsatz als letzte als funktionierend Steuerelementsatz akzeptiert.
1077	0x00000435	ERROR_SERVICE_NEVER_STARTED	Keine Versuche zum Starten des Dienstes wurden seit dem letzten Start.
1078	0x00000436	ERROR_DUPLICATE_SERVICE_NAME	Der Name wird bereits verwendet als einen Dienstnamen oder einen Service-Anzeigenamen.
1079	0x00000437	ERROR_DIFFERENT_SERVICE_ACCOUNT	Für diesen Dienst angegebene Konto unterscheidet sich von dem für andere Dienste, die in demselben Prozess ausgeführt angegebene Konto.
1080	0x00000438	ERROR_CANNOT_DETECT_DRIVER_FAILURE	Fehleraktionen können nur für Win32-Dienste nicht für Treiber festgelegt werden.
1081	0x00000439	ERROR_CANNOT_DETECT_PROCESS_ABORT	Dieser Dienst wird in demselben Prozess wie der Dienststeuerungs-Manager ausgeführt. Daher kann nicht der Dienststeuerungs-Manager Maßnahmen zu ergreifen, falls dieser Dienst Prozess unerwartet beendet wird.
1082	0x0000043A	ERROR_NO_RECOVERY_PROGRAM	Kein Wiederherstellungsprogramm wurde für diesen Dienst konfiguriert.
1083	0x0000043B	ERROR_SERVICE_NOT_IN_EXE	Das ausführbare Programm, dass dieser Dienst für die Ausführung konfiguriert wird den Dienst nicht implementiert.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
1084	0x0000043C	ERROR_NOT_SAFEBOOT_SERVICE	Dieser Dienst kann nicht im abgesicherten Modus gestartet werden.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
1100	0x0000044C	ERROR_END_OF_MEDIA	Das physische Ende des Bandes wurde erreicht.
1101	0x0000044D	ERROR_FILEMARK_DETECTED	Ein Bandzugriff erreichte eine Dateimark.
1102	0x0000044E	ERROR_BEGINNING_OF_MEDIA	Am Anfang des Bandes oder eine Partition wurde gefunden.
1103	0x0000044F	ERROR_SETMARK_DETECTED	Ein Bandzugriff erreichte das Ende eines Satzes von Dateien.
1104	0x00000450	ERROR_NO_DATA_DETECTED	Keine weiteren Daten ist auf dem Band.
1105	0x00000451	ERROR_PARTITION_FAILURE	Band konnte nicht partitioniert werden.
1106	0x00000452	ERROR_INVALID_BLOCK_LENGTH	Beim Zugriff auf ein neues Band einer Partition über mehrere Datenträger ist die aktuelle Blockgröße falsch.
1107	0x00000453	ERROR_DEVICE_NOT_PARTITIONED	Band-Partitionsinformationen konnte nicht gefunden werden, beim Laden eines Bandes.
1108	0x00000454	ERROR_UNABLE_TO_LOCK_MEDIA	Konnte nicht gesperrt werden Auswerfen Mechanismus.
1109	0x00000455	ERROR_UNABLE_TO_UNLOAD_MEDIA	Nicht in der Lage, die Medien zu entladen.
1110	0x00000456	ERROR_MEDIA_CHANGED	Das Medium im Laufwerk möglicherweise geändert.
1111	0x00000457	ERROR_BUS_RESET	Der i/o-Bus wurde zurückgesetzt.
1112	0x00000458	ERROR_NO_MEDIA_IN_DRIVE	Kein Medium im Laufwerk.
1113	0x00000459	ERROR_NO_UNICODE_TRANSLATION	Keine Zuordnung für das Unicode-Zeichen existiert in der Ziel-Multi-Byte-Codepage.
1114	0x0000045A	ERROR_DLL_INIT_FAILED	Eine dynamic Link Library (DLL) Initialisierungsroutine ist fehlgeschlagen.
1115	0x0000045B	ERROR_SHUTDOWN_IN_PROGRESS	Ein Herunterfahren des Systems ist im Gange.
1116	0x0000045C	ERROR_NO_SHUTDOWN_IN_PROGRESS	Nicht in der Lage, das System Herunterfahren abubrechen, da kein Herunterfahren ausgeführt wurde.
1117	0x0000045D	ERROR_IO_DEVICE	Die Anforderung konnte nicht aufgrund eines i/o Device Fehler ausgeführt werden.
1118	0x0000045E	ERROR_SERIAL_NO_DEVICE	Kein serielle Gerät wurde erfolgreich initialisiert. Der serielle Treiber wird entladen.
1119	0x0000045F	ERROR_IRQ_BUSY	Nicht in der Lage, ein Gerät zu öffnen, die eine Interruptanforderung (IRQ) zusammen mit anderen Geräten teilnahm. Mindestens ein anderes Gerät, das diesen IRQ verwendet wurde bereits eröffnet.
1120	0x00000460	ERROR_MORE_WRITES	Eine serielle i/o-Operation wurde durch ein weiteres Schreiben an den seriellen Port abgeschlossen. (Die vervollständigt. \\n(IOCTL_SERIAL_XOFF_COUNTER erreicht NULL.)
1121	0x00000461	ERROR_COUNTER_TIMEOUT	Eine serielle i/o-Operation abgeschlossen, da das Timeout ist abgelaufen. (Die vervollständigt. \\n(IOCTL_SERIAL_XOFF_COUNTER hat nicht Null erreicht.)
1122	0x00000462	ERROR_FLOPPY_ID_MARK_NOT_FOUND	Auf der Diskette wurde keine ID-Adresse-Mark gefunden.
1123	0x00000463	ERROR_FLOPPY_WRONG_CYLINDER	Spuradresse Diskette Sektor-ID-Feld der Diskette Controller Track.
1124	0x00000464	ERROR_FLOPPY_UNKNOWN_ERROR	Der Diskettencontroller hat einen Fehler, der vom Diskettentreiber nicht erkannt wird gemeldet.
1125	0x00000465	ERROR_FLOPPY_BAD_REGISTERS	Der Diskettencontroller kehrte inkonsistente Ergebnisse in seinen Registern zurück.
1126	0x00000466	ERROR_DISK_RECALIBRATE_FAILED	Beim Zugriff auf die Festplatte, schlug ein neu kalibrieren, auch nach Wiederholungen.
1127	0x00000467	ERROR_DISK_OPERATION_FAILED	Bei Festplattenzugriff, schlug ein Festplattenvorgang auch nach Wiederholungen.
1128	0x00000468	ERROR_DISK_RESET_FAILED	Beim Zugriff auf die Festplatte, ein Festplatten-Controller-Reset war nötig, jedoch auch versagt.
1129	0x00000469	ERROR_EOM_OVERFLOW	Physischen Ende Bandende wurde erreicht.
1130	0x0000046A	ERROR_NOT_ENOUGH_SERVER_MEMORY	Es ist nicht genügend Server-Speicherplatz verfügbar, um diesen Befehl zu verarbeiten.
1131	0x0000046B	ERROR_POSSIBLE_DEADLOCK	Eine potenzielle Deadlock-Bedingung wurde erkannt.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
1132	0x0000046C	ERROR_MAPPED_ALIGNMENT	Die Basisadresse oder der angegebene Dateioffset hat nicht die richtige Anordnung.
1140	0x00000474	ERROR_SET_POWER_STATE_VETOED	Ein Versuch, den Energiestatus zu ändern wurde durch eine andere Anwendung oder ein Treiber Veto.
1141	0x00000475	ERROR_SET_POWER_STATE_FAILED	Das System-BIOS konnte einen Versuch, den Energiestatus zu ändern.
1142	0x00000476	ERROR_TOO_MANY_LINKS	Wurde versucht, mehr Verknüpfungen auf eine Datei als das Dateisystem unterstützt erstellen.
1150	0x0000047E	ERROR_OLD_WIN_VERSION	Das angegebene Programm benötigt eine neuere Version von Windows.
1151	0x0000047F	ERROR_APP_WRONG_OS	Das angegebene Programm ist kein Windows- oder MS-DOS-Programm.
1152	0x00000480	ERROR_SINGLE_INSTANCE_APP	Mehr als eine Instanz der das angegebene Programm kann nicht gestartet werden.
1153	0x00000481	ERROR_RMODE_APP	Das angegebene Programm wurde für eine frühere Version von Windows geschrieben.
1154	0x00000482	ERROR_INVALID_DLL	Eine Bibliothekdatei, die zum Ausführen dieser Anwendung erforderlich ist beschädigt.
1155	0x00000483	ERROR_NO_ASSOCIATION	Keine Anwendung ist der angegebenen Datei für diesen Vorgang zugeordnet.
1156	0x00000484	ERROR_DDE_FAIL	Fehler beim Senden des Befehls an die Anwendung.
1157	0x00000485	ERROR_DLL_NOT_FOUND	Eine Bibliothekdatei, die zum Ausführen dieser Anwendung erforderlich kann nicht gefunden werden.
1158	0x00000486	ERROR_NO_MORE_USER_HANDLES	Der aktuelle Prozess hat alle Managerobjekte Griffe für Fenster-Manager-Objekte verwendet.
1159	0x00000487	ERROR_MESSAGE_SYNC_ONLY	Die Nachricht kann nur mit synchronen Vorgängen verwendet werden.
1160	0x00000488	ERROR_SOURCE_ELEMENT_EMPTY	Das angegebenen Quellelement verfügt über keine Medien.
1161	0x00000489	ERROR_DESTINATION_ELEMENT_FULL	Das angegebene Zielelement enthält bereits ein Medium.
1162	0x0000048A	ERROR_ILLEGAL_ELEMENT_ADDRESS	Das angegebene Element ist nicht vorhanden.
1163	0x0000048B	ERROR_MAGAZINE_NOT_PRESENT	Das angegebene Element ist Teil einer Zeitschrift, die nicht vorhanden ist.
1164	0x0000048C	ERROR_DEVICE_REINITIALIZATION_NEEDED	Das angegebene Gerät muss wegen Hardwarefehler.
1165	0x0000048D	ERROR_DEVICE_REQUIRES_CLEANING	Das Gerät hat angezeigt, dass die Reinigung erforderlich ist, bevor weitere Operationen durchgeführt werden.
1166	0x0000048E	ERROR_DEVICE_DOOR_OPEN	Das Gerät hat angezeigt, dass die Tür offen ist.
1167	0x0000048F	ERROR_DEVICE_NOT_CONNECTED	Das Gerät ist nicht angeschlossen.
1168	0x00000490	ERROR_NOT_FOUND	Element nicht gefunden.
1169	0x00000491	ERROR_NO_MATCH	Es gab keine Übereinstimmung für den angegebenen Schlüssel im Index.
1170	0x00000492	ERROR_SET_NOT_FOUND	Der angegebene Eigenschaftensatz ist nicht im Objekt vorhanden.
1171	0x00000493	ERROR_POINT_NOT_FOUND	Der Punkt an GetMouseMovePointsEx übergeben, ist nicht im Puffer.
1172	0x00000494	ERROR_NO_TRACKING_SERVICE	Der Überwachungsdienst (Arbeitsstation) wird nicht ausgeführt.
1173	0x00000495	ERROR_NO_VOLUME_ID	Die Volume-ID konnte nicht gefunden werden.
1175	0x00000497	ERROR_UNABLE_TO_REMOVE_REPLACED	Kann nicht zum Entfernen der Datei ersetzt werden.
1176	0x00000498	ERROR_UNABLE_TO_MOVE_REPLACEMENT	Können nicht die Ersatzdatei in die Datei ersetzt werden. Die Datei ersetzt werden, hat ihren ursprünglichen Namen behalten.
1177	0x00000499	ERROR_UNABLE_TO_MOVE_REPLACEMENT_2	Können nicht die Ersatzdatei in die Datei ersetzt werden. Die Datei ersetzt werden wurde mithilfe der Name der Sicherungskopie umbenannt.
1178	0x0000049A	ERROR_JOURNAL_DELETE_IN_PROGRESS	Das Band-Änderungsjournal wird gelöscht.
1179	0x0000049B	ERROR_JOURNAL_NOT_ACTIVE	Das Band-Änderungsjournal ist nicht aktiv.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
1180	0x0000049C	ERROR_POTENTIAL_FILE_FOUND	Eine Datei wurde gefunden, aber es kann nicht sein, die richtige Datei.
1181	0x0000049D	ERROR_JOURNAL_ENTRY_DELETED	Der Journaleintrag wurde aus dem Journal gelöscht.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
1200	0x000004B0	ERROR_BAD_DEVICE	Der angegebene Geräte-name ist ungültig.
1201	0x000004B1	ERROR_CONNECTION_UNAVAIL	Das Gerät ist momentan nicht verbunden, aber es ist eine Erinnerung Verbindung.
1202	0x000004B2	ERROR_DEVICE_ALREADY_REMEMBERED	Der lokale Geräte-name hat eine erinnerte Verbindung zu anderen Netzwerkressource.
1203	0x000004B3	ERROR_NO_NET_OR_BAD_PATH	Kein Netzwerkanbieter akzeptiert den angegebene Netzwerkpfad.
1204	0x000004B4	ERROR_BAD_PROVIDER	Der angegebene Netzwerk-Provider-Name ist ungültig.
1205	0x000004B5	ERROR_CANNOT_OPEN_PROFILE	Es kann nicht das Netzwerkverbindungsprofil geöffnet werden.
1206	0x000004B6	ERROR_BAD_PROFILE	Das Netzwerkverbindungsprofil ist beschädigt.
1207	0x000004B7	ERROR_NOT_CONTAINER	Eine der Nichtcontainer nicht aufgelistet werden.
1208	0x000004B8	ERROR_EXTENDED_ERROR	Ein erweiterter Fehler ist aufgetreten.
1209	0x000004B9	ERROR_INVALID_GROUPNAME	Das Format des angegebenen Gruppennamens ist ungültig.
1210	0x000004BA	ERROR_INVALID_COMPUTERNAME	Das Format des angegebenen Computernamens ist ungültig.
1211	0x000004BB	ERROR_INVALID_EVENTNAME	Das Format des angegebenen Ereignisnamens ist ungültig.
1212	0x000004BC	ERROR_INVALID_DOMAINNAME	Das Format des angegebenen Domännennamens ist ungültig.
1213	0x000004BD	ERROR_INVALID_SERVICENAME	Das Format des angegebenen Dienstnamens ist ungültig.
1214	0x000004BE	ERROR_INVALID_NETNAME	Das Format des angegebenen Netzwerknamens ist ungültig.
1215	0x000004BF	ERROR_INVALID_SHARENAME	Das Format des angegebenen Freigabenamens ist ungültig.
1216	0x000004C0	ERROR_INVALID_PASSWORDNAME	Das Format des angegebenen Kennworts ist ungültig.
1217	0x000004C1	ERROR_INVALID_MESSAGE_NAME	Das Format des angegebenen Meldung-Name ist ungültig.
1218	0x000004C2	ERROR_INVALID_MESSAGEDEST	Das Format des angegebenen Meldung Ziel ist ungültig.
1219	0x000004C3	ERROR_SESSION_CREDENTIAL_CONFLICT	Die Anmeldeinformationen angegeben Konflikt mit einem vorhandenen Satz von Anmeldeinformationen.
1220	0x000004C4	ERROR_REMOTE_SESSION_LIMIT_EXCEEDED	Es wurde versucht, eine Sitzung mit einem Netzwerkserver herzustellen, aber es gibt bereits zu viele Sitzungen mit diesem Server hergestellt.
1221	0x000004C5	ERROR_DUP_DOMAINNAME	Der Arbeitsgruppe oder Domäne-Name ist bereits von einem anderen Computer im Netzwerk verwendet.
1222	0x000004C6	ERROR_NO_NETWORK	Das Netzwerk ist nicht vorhanden oder nicht gestartet.
1223	0x000004C7	ERROR_CANCELLED	Der Vorgang wurde vom Benutzer abgebrochen.
1224	0x000004C8	ERROR_USER_MAPPED_FILE	Der angeforderte Vorgang kann nicht auf eine Datei mit einem Benutzer zugeordnet Abschnitt öffnen ausgeführt werden.
1225	0x000004C9	ERROR_CONNECTION_REFUSED	Das remote-System verweigert die Netzwerkverbindung.
1226	0x000004CA	ERROR_GRACEFUL_DISCONNECT	Die Netzwerkverbindung wurde erfolgreich geschlossen.
1227	0x000004CB	ERROR_ADDRESS_ALREADY_ASSOCIATED	Der Netzwerk-Transport-Endpunkt hat bereits eine Adresse zugeordnet.
1228	0x000004CC	ERROR_ADDRESS_NOT_ASSOCIATED	Eine Adresse wurde noch nicht mit den Netzwerkendpunkt verbunden.
1229	0x000004CD	ERROR_CONNECTION_INVALID	Ein Vorgang bezog sich auf eine nicht vorhandene Netzwerkverbindung.
1230	0x000004CE	ERROR_CONNECTION_ACTIVE	Ein ungültiger Vorgang bezog sich auf eine aktive Netzwerkverbindung.
1231	0x000004CF	ERROR_NETWORK_UNREACHABLE	Der Speicherort im Netzwerk nicht erreichbar. Informationen zur Fehlerbehebung in Netzwerken finden Sie in der Windows-Hilfe.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
1232	0x000004D0	ERROR_HOST_UNREACHABLE	Der Speicherort im Netzwerk nicht erreichbar. Informationen zur Fehlerbehebung in Netzwerken finden Sie in der Windows-Hilfe.
1233	0x000004D1	ERROR_PROTOCOL_UNREACHABLE	Der Speicherort im Netzwerk nicht erreichbar. Informationen zur Fehlerbehebung in Netzwerken finden Sie in der Windows-Hilfe.
1234	0x000004D2	ERROR_PORT_UNREACHABLE	Kein Dienst ist auf die Ziel-Netzwerk-Endpunkt auf dem entfernten System betrieben.
1235	0x000004D3	ERROR_REQUEST_ABORTED	Die Anforderung wurde abgebrochen.
1236	0x000004D4	ERROR_CONNECTION_ABORTED	Die Netzwerkverbindung wurde durch das lokale System abgebrochen.
1237	0x000004D5	ERROR_RETRY	Der Vorgang konnte nicht abgeschlossen werden. Eine Wiederholung sollte durchgeführt werden.
1238	0x000004D6	ERROR_CONNECTION_COUNT_LIMIT	Eine Verbindung zum Server konnte nicht erfolgen, da das Limit für die Anzahl der gleichzeitigen Verbindungen für dieses Konto erreicht wurde.
1239	0x000004D7	ERROR_LOGIN_TIME_RESTRICTION	Der Versuch, zu einer nicht autorisierten Tageszeit für dieses Konto einloggen.
1240	0x000004D8	ERROR_LOGIN_WKSTA_RESTRICTION	Das Konto ist nicht berechtigt, von dieser Station aus anmelden.
1241	0x000004D9	ERROR_INCORRECT_ADDRESS	Die Netzwerkadresse konnte für den angeforderten Vorgang nicht verwendet werden.
1242	0x000004DA	ERROR_ALREADY_REGISTERED	Der Dienst ist bereits registriert.
1243	0x000004DB	ERROR_SERVICE_NOT_FOUND	Der angegebene Dienst existiert nicht.
1244	0x000004DC	ERROR_NOT_AUTHENTICATED	Der angeforderte Vorgang wurde nicht durchgeführt, weil der Benutzer nicht authentifiziert wurde.
1245	0x000004DD	ERROR_NOT_LOGGED_ON	Der angeforderte Vorgang wurde nicht durchgeführt, weil der Benutzer nicht mit dem Netzwerk angemeldet hat. Der angegebene Dienst existiert nicht.
1246	0x000004DE	ERROR_CONTINUE	Fahren Sie mit bei der Arbeit.
1247	0x000004DF	ERROR_ALREADY_INITIALIZED	Es wurde versucht, einen Initialisierungsvorgang auszuführen, wenn die Initialisierung bereits abgeschlossen wurde.
1248	0x000004E0	ERROR_NO_MORE_DEVICES	Keine weiteren lokalen Geräte.
1249	0x000004E1	ERROR_NO_SUCH_SITE	Die angegebene Site ist nicht vorhanden.
1250	0x000004E2	ERROR_DOMAIN_CONTROLLER_EXISTS	Ein Domänencontroller mit dem angegebenen Namen ist bereits vorhanden.
1251	0x000004E3	ERROR_ONLY_IF_CONNECTED	Dieser Vorgang wird unterstützt, nur, wenn Sie mit dem Server verbunden sind.
1252	0x000004E4	ERROR_OVERRIDE_NOCHANGES	Die Gruppenrichtlinienumgebung sollte die Erweiterung aufrufen, auch wenn keine Änderungen vorliegen.
1253	0x000004E5	ERROR_BAD_USER_PROFILE	Der angegebene Benutzer muss kein gültiges Profil.
1254	0x000004E6	ERROR_NOT_SUPPORTED_ON_SBS	Dieser Vorgang wird auf einem Microsoft Small Business Server nicht unterstützt.
1255	0x000004E7	ERROR_SERVER_SHUTDOWN_IN_PROGRESS	Der Server-Computer wird heruntergefahren.
1256	0x000004E8	ERROR_HOST_DOWN	Das remote-System ist nicht verfügbar. Informationen zur Fehlerbehebung in Netzwerken finden Sie in der Windows-Hilfe.
1257	0x000004E9	ERROR_NON_ACCOUNT_SID	Der angegebene Sicherheitsbezeichner stammt nicht von einer Kontodomäne.
1258	0x000004EA	ERROR_NON_DOMAIN_SID	Der angegebene Sicherheitsbezeichner verfügt über keine Domänenkomponente.
1259	0x000004EB	ERROR_APPHELP_BLOCK	AppHelp Dialog abgebrochen, wodurch die Anwendung gestartet.
1260	0x000004EC	ERROR_ACCESS_DISABLED_BY_POLICY	Zugriff auf die angeforderte Ressource wurde vom Systemadministrator deaktiviert.
1261	0x000004ED	ERROR_REG_NAT_CONSUMPTION	Ein Programm-Versuch, verwenden Sie ein ungültiges registrieren Wert. Normalerweise verursacht durch einen nicht initialisierten Register. Dieser Fehler ist Itanium bestimmte.
1262	0x000004EE	ERROR_CSCSHARE_OFFLINE	Die Freigabe ist derzeit offline oder existiert nicht.



Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
1300	0x00000514	ERROR_NOT_ALL_ASSIGNED	Nicht alle Privilegien verwiesen werden an den Aufrufer zugewiesen.
1301	0x00000515	ERROR_SOME_NOT_MAPPED	Einige Zuordnung von Kontennamen und Sicherheits-IDs nicht geschah.
1302	0x00000516	ERROR_NO_QUOTAS_FOR_ACCOUNT	Keine Kontingentgrenzen System sind für dieses Konto festgelegt worden.
1303	0x00000517	ERROR_LOCAL_USER_SESSION_KEY	Es ist kein Chiffrierschlüssel verfügbar. Ein bekannter Verschlüsselungsschlüssel wurde zurückgegeben.
1304	0x00000518	ERROR_NULL_LM_PASSWORD	Das Kennwort ist zu komplex, um in ein LAN Manager-Kennwort umgewandelt werden. Das zurückgegebene LAN Manager-Kennwort ist eine NULL-Zeichenfolge.
1305	0x00000519	ERROR_UNKNOWN_REVISION	Die Revisionsstufe ist nicht bekannt.
1306	0x0000051A	ERROR_REVISION_MISMATCH	Gibt an, dass zwei Revisionsstufen nicht kompatibel sind.
1307	0x0000051B	ERROR_INVALID_OWNER	Diese Sicherheits-ID kann nicht als Besitzer dieses Objekts zugewiesen werden.
1308	0x0000051C	ERROR_INVALID_PRIMARY_GROUP	Diese Sicherheits-ID kann nicht als primäre Gruppe eines Objekts zugewiesen werden.
1309	0x0000051D	ERROR_NO_IMPERSONATION_TOKEN	Eine hat versucht, ein Identitätstoken operieren von einem Thread, die einen Client derzeit keinen Identitätswechsel ist.
1310	0x0000051E	ERROR_CANT_DISABLE_MANDATORY	Die Gruppe kann nicht deaktiviert werden.
1311	0x0000051F	ERROR_NO_LOGON_SERVERS	Es sind momentan keine Anmeldeserver zur Verfügung, um die Anmeldeanforderung zu verarbeiten.
1312	0x00000520	ERROR_NO_SUCH_LOGON_SESSION	Eine angegebene Anmeldesitzung ist nicht vorhanden. Es kann bereits gekündigt haben.
1313	0x00000521	ERROR_NO_SUCH_PRIVILEGE	Angegebenen Privileg ist nicht vorhanden.
1314	0x00000522	ERROR_PRIVILEGE_NOT_HELD	Eine erforderliche Berechtigung ist nicht durch den Kunden statt.
1315	0x00000523	ERROR_INVALID_ACCOUNT_NAME	Der angegebene Name ist keine korrekt formatierte Kontonamen.
1316	0x00000524	ERROR_USER_EXISTS	Der angegebene Benutzer ist bereits vorhanden.
1317	0x00000525	ERROR_NO_SUCH_USER	Der angegebene Benutzer ist nicht vorhanden.
1318	0x00000526	ERROR_GROUP_EXISTS	Die angegebene Gruppe ist bereits vorhanden.
1319	0x00000527	ERROR_NO_SUCH_GROUP	Die angegebene Gruppe existiert nicht.
1320	0x00000528	ERROR_MEMBER_IN_GROUP	Entweder das angegebene Benutzerkonto ist bereits ein Mitglied der angegebenen Gruppe, oder die angegebene Gruppe kann nicht gelöscht werden, da sie ein Mitglied enthält.
1321	0x00000529	ERROR_MEMBER_NOT_IN_GROUP	Das angegebene Benutzerkonto ist kein Mitglied der angegebenen Gruppe Account.
1322	0x0000052A	ERROR_LAST_ADMIN	Die letzte verbleibende Administratorkonto kann nicht deaktiviert oder gelöscht werden.
1323	0x0000052B	ERROR_WRONG_PASSWORD	Nicht in der Lage, das Kennwort zu aktualisieren. Der Wert als das aktuelle Kennwort ist falsch.
1324	0x0000052C	ERROR_ILL_FORMED_PASSWORD	Nicht in der Lage, das Kennwort zu aktualisieren. Für das neue Kennwort angegebene Wert enthält Werte, die in Kennwörtern nicht zulässig sind.
1325	0x0000052D	ERROR_PASSWORD_RESTRICTION	Nicht in der Lage, das Kennwort zu aktualisieren. Der Wert für das neue Kennwort erfüllt nicht die Länge, der Komplexität oder der Geschichte Bedarf der Domäne.
1326	0x0000052E	ERROR_LOGON_FAILURE	Anmeldung fehlgeschlagen: Unbekannter Benutzername oder falsches Kennwort.
1327	0x0000052F	ERROR_ACCOUNT_RESTRICTION	Anmeldung fehlgeschlagen: Benutzerkontenbeschränkung.
1328	0x00000530	ERROR_INVALID_LOGON_HOURS	Anmeldung fehlgeschlagen: Konto-Anmeldung Zeit Einschränkung Verletzung.
1329	0x00000531	ERROR_INVALID_WORKSTATION	Anmeldung fehlgeschlagen: Benutzer nicht gestattet, an diesem Computer anmelden.
1330	0x00000532	ERROR_PASSWORD_EXPIRED	Anmeldung fehlgeschlagen: das angegebene Konto-Kennwort ist abgelaufen.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
1331	0x00000533	ERROR_ACCOUNT_DISABLED	Anmeldung fehlgeschlagen: Konto zurzeit deaktiviert.
1332	0x00000534	ERROR_NONE_MAPPED	Zuordnungen von Kontennamen und Sicherheitskennungen wurden nicht durchgeführt.
1333	0x00000535	ERROR_TOO_MANY_LUIDS_REQUESTED	Zu viele lokale Benutzer-IDs (LUIDs) wurden gleichzeitig angefordert.
1334	0x00000536	ERROR_LUIDS_EXHAUSTED	Keine weiteren lokalen Benutzerkennungen (LUIDs) stehen zur Verfügung.
1335	0x00000537	ERROR_INVALID_SUB_AUTHORITY	Die Teilautoritätskomponente einer Sicherheits-ID ist ungültig für diese spezielle Verwendung.
1336	0x00000538	ERROR_INVALID_ACL	Die Struktur der Access Control List (ACL) ist ungültig.
1337	0x00000539	ERROR_INVALID_SID	Die Struktur der Sicherheits-ID ist ungültig.
1338	0x0000053A	ERROR_INVALID_SECURITY_DESCR	Die Sicherheit Deskriptor Struktur ist ungültig.
1340	0x0000053C	ERROR_BAD_INHERITANCE_ACL	Die vererbte Zugriffssteuerungsliste (ACL) oder Zugriffssteuerungseintrag (ACE) konnte nicht erstellt werden.
1341	0x0000053D	ERROR_SERVER_DISABLED	Der Server ist derzeit deaktiviert.
1342	0x0000053E	ERROR_SERVER_NOT_DISABLED	Der Server ist zurzeit aktiviert.
1343	0x0000053F	ERROR_INVALID_ID_AUTHORITY	Der angegebene Wert wurde ein ungültiger Wert für Bezeichnerautorität.
1344	0x00000540	ERROR_ALLOTTED_SPACE_EXCEEDED	Kein Speicher mehr verfügbar ist für Sicherheits-Informationen-Updates verfügbar.
1345	0x00000541	ERROR_INVALID_GROUP_ATTRIBUTES	Die angegebenen Attribute sind ungültig oder nicht kompatibel mit den Attributen der Gruppe als Ganzes.
1346	0x00000542	ERROR_BAD_IMPERSONATION_LEVEL	Eine erforderliche Identitätswechselebene wurde nicht bereitgestellt oder die angegebene Identitätswechselebene ist ungültig.
1347	0x00000543	ERROR_CANT_OPEN_ANONYMOUS	Ein anonymer Sicherheitsebenentoken kann nicht geöffnet werden.
1348	0x00000544	ERROR_BAD_VALIDATION_CLASS	Die Validierung Informationsklasse angefordert war ungültig.
1349	0x00000545	ERROR_BAD_TOKEN_TYPE	Der Typ des Tokens ist für den versuchten Einsatz ungeeignet.
1350	0x00000546	ERROR_NO_SECURITY_ON_OBJECT	Nicht möglich eine Sicherheit für ein Objekt ohne zugeordnete Sicherheit aufweist.
1351	0x00000547	ERROR_CANT_ACCESS_DOMAIN_INFO	Konfigurationsinformationen konnte nicht vom Domänencontroller, gelesen werden da der Computer nicht verfügbar ist oder Zugriff wurde verweigert.
1352	0x00000548	ERROR_INVALID_SERVER_STATE	Die Sicherheitskontenverwaltung (SAM) oder lokale Sicherheits-Autorität (LSA) Server befand sich in den Zustand um die Sicherheit durchzuführen.
1353	0x00000549	ERROR_INVALID_DOMAIN_STATE	Die Domäne befand sich in einem Zustand, in den Sicherheitsvorgang auszuführen.
1354	0x0000054A	ERROR_INVALID_DOMAIN_ROLE	Dieser Vorgang ist nur für den primären Domänencontroller der Domäne zulässig.
1355	0x0000054B	ERROR_NO_SUCH_DOMAIN	Die angegebene Domäne ist nicht vorhanden oder konnte keine Verbindung hergestellt werden.
1356	0x0000054C	ERROR_DOMAIN_EXISTS	Die angegebene Domäne ist bereits vorhanden.
1357	0x0000054D	ERROR_DOMAIN_LIMIT_EXCEEDED	Wurde versucht, die Grenze für die Anzahl der Domänen je Server zu überschreiten.
1358	0x0000054E	ERROR_INTERNAL_DB_CORRUPTION	Kann nicht die angeforderte Operation wegen einer katastrophalen Datenträgerfehler oder eine Beschädigung von Daten-Struktur auf der Festplatte abgeschlossen.
1359	0x0000054F	ERROR_INTERNAL_ERROR	Ein interner Fehler aufgetreten.
1360	0x00000550	ERROR_GENERIC_NOT_MAPPED	Generische Zugriffarten wurden in einer Zugriffsmaske enthalten, die bereits nicht generischen Typen zugeordnet sein sollten.
1361	0x00000551	ERROR_BAD_DESCRIPTOR_FORMAT	Eine Sicherheitsbeschreibung ist nicht im richtigen Format (absolut oder selbstbezogen).

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
1362	0x00000552	ERROR_NOT_LOGON_PROCESS	Die angeforderte Aktion ist auf die Verwendung von Anmeldeprozessen nur beschränkt. Der aufrufende Prozess ist nicht als einen Anmeldevorgang registriert.
1363	0x00000553	ERROR_LOGON_SESSION_EXISTS	Eine neue Anmeldesitzung kann nicht mit der ID gestartet werden, die bereits verwendet wird.
1364	0x00000554	ERROR_NO_SUCH_PACKAGE	Ein angegebenen Authentifizierungspaket ist unbekannt.
1365	0x00000555	ERROR_BAD_LOGON_SESSION_STATE	Die Anmeldesitzung ist nicht in einem Staat, der die angeforderte Operation entspricht.
1366	0x00000556	ERROR_LOGON_SESSION_COLLISION	Die Anmeldesitzung, die, der ID bereits wird, verwendet.
1367	0x00000557	ERROR_INVALID_LOGON_TYPE	Eine Anmeldeanforderung enthielt einen Typwert ungültige Anmeldeversuche.
1368	0x00000558	ERROR_CANNOT_IMPERSONATE	Nicht mit mit einer named Pipe, bis Daten aus dem Rohr gelesen wurde.
1369	0x00000559	ERROR_RXACT_INVALID_STATE	Der Transaktionszustand einer Teilstruktur der Registrierung ist nicht kompatibel mit dem angeforderten Vorgang.
1370	0x0000055A	ERROR_RXACT_COMMIT_FAILURE	Eine innere Sicherheit-Datenbankfehler ist aufgetreten.
1371	0x0000055B	ERROR_SPECIAL_ACCOUNT	Diesen Vorgang für integrierte Konten nicht möglich.
1372	0x0000055C	ERROR_SPECIAL_GROUP	Dieser Vorgang auf integrierte Gruppe kann nicht ausgeführt werden.
1373	0x0000055D	ERROR_SPECIAL_USER	Dieser Vorgang auf diesen integrierten spezielle Benutzer kann nicht durchgeführt werden.
1374	0x0000055E	ERROR_MEMBERS_PRIMARY_GROUP	Der Benutzer kann nicht aus einer Gruppe entfernt werden, da die Gruppe momentan die primäre Gruppe des Benutzers ist.
1375	0x0000055F	ERROR_TOKEN_ALREADY_IN_USE	Das Token wird bereits als primäres Token verwendet.
1376	0x00000560	ERROR_NO_SUCH_ALIAS	Die angegebene lokale Gruppe ist nicht vorhanden.
1377	0x00000561	ERROR_MEMBER_NOT_IN_ALIAS	Der angegebene Kontenname ist kein Mitglied der lokalen Gruppe.
1378	0x00000562	ERROR_MEMBER_IN_ALIAS	Der angegebene Kontenname ist bereits Mitglied der lokalen Gruppe.
1379	0x00000563	ERROR_ALIAS_EXISTS	Die angegebene lokale Gruppe ist bereits vorhanden.
1380	0x00000564	ERROR_LOGON_NOT_GRANTED	Anmeldung fehlgeschlagen: der Benutzer wurde den angeforderte Anmeldetyp auf diesem Computer nicht erteilt.
1381	0x00000565	ERROR_TOO_MANY_SECRETS	Die maximale Anzahl der Geheimnisse, die in einem einzigen System gespeichert werden können, wurde überschritten.
1382	0x00000566	ERROR_SECRET_TOO_LONG	Die Länge eines Geheimnisses überschreitet die maximale zulässige Länge.
1383	0x00000567	ERROR_INTERNAL_DB_ERROR	Die lokale Sicherheits-Autorität-Datenbank enthält eine interne Inkonsistenz.
1384	0x00000568	ERROR_TOO_MANY_CONTEXT_IDS	Im Verlauf eines Anmeldeversuchs angesammelt Sicherheitskontext des Benutzers zu viele Sicherheits-IDs.
1385	0x00000569	ERROR_LOGON_TYPE_NOT_GRANTED	Anmeldung fehlgeschlagen: der Benutzer wurde den angeforderte Anmeldetyp auf diesem Computer nicht erteilt.
1386	0x0000056A	ERROR_NT_CROSS_ENCRYPTION_REQUIRED	Ein querverschlüsseltes Kennwort ist erforderlich, ein Benutzerkennwort zu ändern.
1387	0x0000056B	ERROR_NO_SUCH_MEMBER	Ein neues Mitglied konnte nicht hinzugefügt oder entfernt aus der lokalen Gruppe, da das Mitglied nicht vorhanden ist.
1388	0x0000056C	ERROR_INVALID_MEMBER	Ein neues Mitglied konnte nicht zu einer lokalen Gruppe hinzugefügt werden, da das Mitglied den falschen Kontentyp aufweist.
1389	0x0000056D	ERROR_TOO_MANY_SIDS	Zu viele Sicherheits-IDs es wurden angegeben.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
1390	0x0000056E	ERROR_LM_CROSS_ENCRYPTION_REQUIRED	Ein querverschlüsseltes Kennwort ist erforderlich, um dieses Benutzerkennwort zu ändern.
1391	0x0000056F	ERROR_NO_INHERITANCE	Gibt an, dass eine Zugriffssteuerungsliste keine vererbaren Komponenten enthält.
1392	0x00000570	ERROR_FILE_CORRUPT	Die Datei oder das Verzeichnis ist beschädigt und nicht lesbar.
1393	0x00000571	ERROR_DISK_CORRUPT	Die Datenträgerstruktur ist beschädigt und nicht lesbar.
1394	0x00000572	ERROR_NO_USER_SESSION_KEY	Es ist kein Benutzersitzungsschlüssel für die angegebene Anmeldesitzung.
1395	0x00000573	ERROR_LICENSE_QUOTA_EXCEEDED	Der Dienst, auf die zugegriffen wird für eine bestimmte Anzahl von Verbindungen lizenziert. Keine weitere Verbindungen können zu dieser Zeit zum Dienst erfolgen, denn es gibt bereits so viele Verbindungen wie der Service annehmen kann.
1396	0x00000574	ERROR_WRONG_TARGET_NAME	Anmeldung fehlgeschlagen: Der Zielkontenname ist falsch.
1397	0x00000575	ERROR_MUTUAL_AUTH_FAILED	Gegenseitiger Authentifizierung ist fehlgeschlagen. Das Serverkennwort ist nicht mehr aktuell auf dem Domänencontroller.
1398	0x00000576	ERROR_TIME_SKEW	Gibt es einen Zeitunterschied zwischen Client und Server.
1399	0x00000577	ERROR_CURRENT_DOMAIN_NOT_ALLOWED	Dieser Vorgang kann nicht auf der aktuellen Domäne ausgeführt werden.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
1400	0x00000578	ERROR_INVALID_WINDOW_HANDLE	Ungültiges Fensterhandle.
1401	0x00000579	ERROR_INVALID_MENU_HANDLE	Ungültiges Menü-Handle.
1402	0x0000057A	ERROR_INVALID_CURSOR_HANDLE	Ungültiger Cursor-Handle.
1403	0x0000057B	ERROR_INVALID_ACCEL_HANDLE	Ungültige Beschleuniger Tabelle Handle.
1404	0x0000057C	ERROR_INVALID_HOOK_HANDLE	Ungültiger Hook-Handle.
1405	0x0000057D	ERROR_INVALID_DWP_HANDLE	Ungültiges Handle für eine Multiple-Fensterposition-Struktur.
1406	0x0000057E	ERROR_TLW_WITH_WSCHILD	Auf der obersten Ebene untergeordnetes Fenster kann nicht erstellt werden.
1407	0x0000057F	ERROR_CANNOT_FIND_WND_CLASS	Fensterklasse wurde nicht gefunden.
1408	0x00000580	ERROR_WINDOW_OF_OTHER_THREAD	Ungültiges Fenster; Es gehört zum anderen Thread.
1409	0x00000581	ERROR_HOTKEY_ALREADY_REGISTERED	Abkürzungstaste ist bereits registriert.
1410	0x00000582	ERROR_CLASS_ALREADY_EXISTS	Klasse ist bereits vorhanden.
1411	0x00000583	ERROR_CLASS_DOES_NOT_EXIST	Klasse ist nicht vorhanden.
1412	0x00000584	ERROR_CLASS_HAS_WINDOWS	Klasse hat noch geöffnete Fenster.
1413	0x00000585	ERROR_INVALID_INDEX	Ungültiger Index.
1414	0x00000586	ERROR_INVALID_ICON_HANDLE	Ungültiges Symbolhandle.
1415	0x00000587	ERROR_PRIVATE_DIALOG_INDEX	Private DIALOG-Fenster Wörter verwenden.
1416	0x00000588	ERROR_LISTBOX_ID_NOT_FOUND	Der Listenfeldbezeichner wurde nicht gefunden.
1417	0x00000589	ERROR_NO_WILDCARD_CHARACTERS	Es fanden sich keine Platzhalter.
1418	0x0000058A	ERROR_CLIPBOARD_NOT_OPEN	Thread muss keine Zwischenablage öffnen.
1419	0x0000058B	ERROR_HOTKEY_NOT_REGISTERED	Abkürzungstaste ist nicht registriert.
1420	0x0000058C	ERROR_WINDOW_NOT_DIALOG	Das Fenster ist kein gültiges Dialogfenster.
1421	0x0000058D	ERROR_CONTROL_ID_NOT_FOUND	Steuerelement-ID nicht gefunden.
1422	0x0000058E	ERROR_INVALID_COMBOBOX_MESSAGE	Ungültige Nachricht für ein Kombinationsfeld, da es kein Edit-Steuerelement verfügt.
1423	0x0000058F	ERROR_WINDOW_NOT_COMBOBOX	Das Fenster ist kein Kombinationsfeld.
1424	0x00000590	ERROR_INVALID_EDIT_HEIGHT	Höhe muss weniger als 256 sein.
1425	0x00000591	ERROR_DC_NOT_FOUND	Ungültiges Gerät Kontexthandle (DC).
1426	0x00000592	ERROR_INVALID_HOOK_FILTER	Ungültiger Hook-Prozedur-Typ.
1427	0x00000593	ERROR_INVALID_FILTER_PROC	Ungültiger Hook-Prozedur.
1428	0x00000594	ERROR_HOOK_NEEDS_HMOD	Kann nicht-lokaler Hook ohne ein Modulhandle gesetzt.
1429	0x00000595	ERROR_GLOBAL_ONLY_HOOK	Diese Hookprozedur kann nur global festgelegt werden.
1430	0x00000596	ERROR_JOURNAL_HOOK_SET	Die Journal-Hookprozedur ist bereits installiert.
1431	0x00000597	ERROR_HOOK_NOT_INSTALLED	Die Hookprozedur ist nicht installiert.
1432	0x00000598	ERROR_INVALID_LB_MESSAGE	Ungültige Nachricht für Listenfeld für Einfachauswahl.
1433	0x00000599	ERROR_SETCOUNT_ON_BAD_LB	LB_SETCOUNT an non-lazy-Listenfeld gesendet.
1434	0x0000059A	ERROR_LB_WITHOUT_TABSTOPS	Dieses Listenfeld unterstützt keine Tabstopps.
1435	0x0000059B	ERROR_DESTROY_OBJECT_OF_OTHER_THREAD	Von einem anderen Thread erstellte Objekt kann nicht zerstört werden.
1436	0x0000059C	ERROR_CHILD_WINDOW_MENU	Untergeordnete Fenster können nicht Menüs haben.
1437	0x0000059D	ERROR_NO_SYSTEM_MENU	Das Fenster muss kein Systemmenü.
1438	0x0000059E	ERROR_INVALID_MSGBOX_STYLE	Ungültige Nachricht Knotenart.
1439	0x0000059F	ERROR_INVALID_SPI_VALUE	Ungültige systemweite (SPI_*)-Parameter.
1440	0x000005A0	ERROR_SCREEN_ALREADY_LOCKED	Bildschirm bereits gesperrt.
1441	0x000005A1	ERROR_HWNDSDS_HAVE_DIFF_PARENT	Alle Handles für Windows in einer Multiple-Fenster-Position-Struktur müssen die gleichen Eltern.
1442	0x000005A2	ERROR_NOT_CHILD_WINDOW	Das Fenster ist kein untergeordnetes Fenster.
1443	0x000005A3	ERROR_INVALID_GW_COMMAND	Ungültiger GW_*-Befehl.
1444	0x000005A4	ERROR_INVALID_THREAD_ID	Ungültiger Threadbezeichner.
1445	0x000005A5	ERROR_NON_MDICHILD_WINDOW	Eine Nachricht aus einem Fenster, die kein mehrere Document Interface (MDI)-Fenster ist nicht verarbeitet werden.
1446	0x000005A6	ERROR_POPUP_ALREADY_ACTIVE	Popup-Menü bereits aktiv.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
1447	0x000005A7	ERROR_NO_SCROLLBARS	Das Fenster hat keine Bildlaufleisten.
1448	0x000005A8	ERROR_INVALID_SCROLLBAR_RANGE	Scroll Bar Bereich sein nicht größer als MAXLONG.
1449	0x000005A9	ERROR_INVALID_SHOWWIN_COMMAND	Nicht anzeigen oder entfernen das Fenster in der Art und Weise angeben.
1450	0x000005AA	ERROR_NO_SYSTEM_RESOURCES	Es gibt nicht genügend Systemressourcen um den angeforderten Dienst auszuführen.
1451	0x000005AB	ERROR_NONPAGED_SYSTEM_RESOURCES	Es gibt nicht genügend Systemressourcen um den angeforderten Dienst auszuführen.
1452	0x000005AC	ERROR_PAGED_SYSTEM_RESOURCES	Es gibt nicht genügend Systemressourcen um den angeforderten Dienst auszuführen.
1453	0x000005AD	ERROR_WORKING_SET_QUOTA	Nicht genügend Quoten, um den angeforderten Dienst auszuführen.
1454	0x000005AE	ERROR_PAGEFILE_QUOTA	Nicht genügend Quoten, um den angeforderten Dienst auszuführen.
1455	0x000005AF	ERROR_COMMITMENT_LIMIT	Die Auslagerungsdatei ist zu klein für diesen Vorgang abgeschlossen.
1456	0x000005B0	ERROR_MENU_ITEM_NOT_FOUND	Ein Menüelement wurde nicht gefunden.
1457	0x000005B1	ERROR_INVALID_KEYBOARD_HANDLE	Ungültige Tastatur Layout Handle.
1458	0x000005B2	ERROR_HOOK_TYPE_NOT_ALLOWED	Hooktyp ist nicht zulässig.
1459	0x000005B3	ERROR_REQUIRES_INTERACTIVE_WINDOWSTATION	Dieser Vorgang erfordert eine interaktive Arbeitsstation.
1460	0x000005B4	ERROR_TIMEOUT	Dieser Vorgang zurückgegeben, da das Timeout ist abgelaufen.
1461	0x000005B5	ERROR_INVALID_MONITOR_HANDLE	Ungültige Monitor Handle.
1500	0x000005DC	ERROR_EVENTLOG_FILE_CORRUPT	Die Ereignisprotokolldatei ist beschädigt.
1501	0x000005DD	ERROR_EVENTLOG_CANT_START	Keine Ereignisprotokolldatei konnte geöffnet werden, so dass der Ereignis-Protokollierung-Dienst nicht gestartet werden konnte.
1502	0x000005DE	ERROR_LOG_FILE_FULL	Die Ereignisprotokolldatei ist voll.
1503	0x000005DF	ERROR_EVENTLOG_FILE_CHANGED	Die Ereignisprotokolldatei wurde zwischen den Lesevorgängen geändert.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
1601	0x00000641	ERROR_INSTALL_SERVICE_FAILURE	Der Windows Installer-Dienst konnte nicht zugegriffen werden. Kontaktieren Sie den Support, um sicherzustellen, dass der Windows-Installationsdienst ordnungsgemäß registriert ist.
1602	0x00000642	ERROR_INSTALL_USEREXIT	Benutzer hat die Installation abgebrochen.
1603	0x00000643	ERROR_INSTALL_FAILURE	Schwerwiegender Fehler während der Installation.
1604	0x00000644	ERROR_INSTALL_SUSPEND	Installation angehalten, unvollständig.
1605	0x00000645	ERROR_UNKNOWN_PRODUCT	Diese Aktion gilt nur für Produkte, die derzeit installiert sind.
1606	0x00000646	ERROR_UNKNOWN_FEATURE	Funktionskennung ist nicht registriert.
1607	0x00000647	ERROR_UNKNOWN_COMPONENT	Komponentenkennung ist nicht registriert.
1608	0x00000648	ERROR_UNKNOWN_PROPERTY	Unbekannte Eigenschaft.
1609	0x00000649	ERROR_INVALID_HANDLE_STATE	Handle ist in einem ungültigen Zustand.
1610	0x0000064A	ERROR_BAD_CONFIGURATION	Die Konfigurationsdaten für dieses Produkt ist beschädigt. Kontaktieren Sie Ihren Support.
1611	0x0000064B	ERROR_INDEX_ABSENT	Komponente-Qualifizierer ist nicht vorhanden.
1612	0x0000064C	ERROR_INSTALL_SOURCE_ABSENT	Die Installationsquelle für dieses Produkt ist nicht verfügbar. Stellen Sie sicher, dass die Quelle vorhanden ist und dass Sie darauf zugreifen können.
1613	0x0000064D	ERROR_INSTALL_PACKAGE_VERSION	Dieses Installationspaket kann nicht vom Windows-Installationsdienst installiert werden. Sie müssen ein Windows Servicepack installieren, das eine neuere Version des Windows Installer-Dienstes enthält.
1614	0x0000064E	ERROR_PRODUCT_UNINSTALLED	Produkt deinstalliert wird.
1615	0x0000064F	ERROR_BAD_QUERY_SYNTAX	SQL-Abfragesyntax ungültig oder wird nicht unterstützt.
1616	0x00000650	ERROR_INVALID_FIELD	Eintragsfeld ist nicht vorhanden.
1617	0x00000651	ERROR_DEVICE_REMOVED	Das Gerät wurde entfernt.
1618	0x00000652	ERROR_INSTALL_ALREADY_RUNNING	Eine andere Installation wird bereits ausgeführt. Schließen Sie die Installation bevor Sie mit der Installation fortfahren.
1619	0x00000653	ERROR_INSTALL_PACKAGE_OPEN_FAILED	Dieses Installationspaket konnte nicht geöffnet werden. Stellen Sie sicher, dass das Paket vorhanden ist und dass Sie darauf zugreifen können, oder wenden Sie sich an den Anwendungshersteller, um sicherzustellen, dass dies ein gültiges Windows Installer-Paket ist.
1620	0x00000654	ERROR_INSTALL_PACKAGE_INVALID	Dieses Installationspaket konnte nicht geöffnet werden. Wenden Sie sich an den Anwendungshersteller, um sicherzustellen, dass dies ein gültiges Windows Installer-Paket ist.
1621	0x00000655	ERROR_INSTALL_UI_FAILURE	Es gab einen Fehler beim Starten der Benutzeroberfläche des Windows Installer-Dienst. Kontaktieren Sie Ihren Support.
1622	0x00000656	ERROR_INSTALL_LOG_FAILURE	Fehler beim Öffnen der Installationsprotokolldatei. Stellen Sie sicher, dass der Speicherort der angegebenen Protokolldatei vorhanden ist und dass Sie darauf schreiben können.
1623	0x00000657	ERROR_INSTALL_LANGUAGE_UNSUPPORTED	Die Sprache dieses Installationspakets wird von Ihrem System nicht unterstützt.
1624	0x00000658	ERROR_INSTALL_TRANSFORM_FAILURE	Fehler beim Anwenden der Transformationen. Stellen Sie sicher, dass die angegebenen Transformationspfade gültig sind.
1625	0x00000659	ERROR_INSTALL_PACKAGE_REJECTED	Diese Installation ist von Systemrichtlinien verboten. Kontaktieren Sie Ihren Systemadministrator.
1626	0x0000065A	ERROR_FUNCTION_NOT_CALLED	Funktion konnte nicht ausgeführt werden.
1627	0x0000065B	ERROR_FUNCTION_FAILED	Funktion ist während der Ausführung fehlgeschlagen.
1628	0x0000065C	ERROR_INVALID_TABLE	Ungültige oder unbekannte Tabelle angegeben.
1629	0x0000065D	ERROR_DATATYPE_MISMATCH	Bereitgestellten Daten sind vom falschen Typ.
1630	0x0000065E	ERROR_UNSUPPORTED_TYPE	Daten dieser Art werden nicht unterstützt.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
1631	0x0000065F	ERROR_CREATE_FAILED	Der Windows Installer-Dienst konnte nicht gestartet werden. Kontaktieren Sie Ihren Support.
1632	0x00000660	ERROR_INSTALL_TEMP_UNWRITABLE	Der temp-Ordner ist entweder voll oder unzugänglich. Überprüfen Sie, ob der temp-Ordner vorhanden ist und ob Sie darauf schreiben können.
1633	0x00000661	ERROR_INSTALL_PLATFORM_UNSUPPORTED	Dieses Installationspaket wird von diesem Prozessortyp nicht unterstützt. Wenden Sie sich an Ihren Hersteller.
1634	0x00000662	ERROR_INSTALL_NOTUSED	Komponente auf diesem Computer nicht verwendet.
1635	0x00000663	ERROR_PATCH_PACKAGE_OPEN_FAILED	Das Patchpaket konnte nicht geöffnet werden. Stellen Sie sicher, dass das Patchpaket vorhanden ist und dass Sie darauf zugreifen können, oder wenden Sie sich an den Anwendungshersteller, um sicherzustellen, dass dies ein gültiges Windows Installer-Patchpaket ist.
1636	0x00000664	ERROR_PATCH_PACKAGE_INVALID	Das Patchpaket konnte nicht geöffnet werden. Wenden Sie sich an den Anwendungshersteller, um sicherzustellen, dass dies ein gültiges Windows Installer-Patchpaket ist.
1637	0x00000665	ERROR_PATCH_PACKAGE_UNSUPPORTED.	Dieses Patchpaket kann vom Windows-Installationsdienst nicht verarbeitet werden. Sie müssen ein Windows Servicepack installieren, das eine neuere Version des Windows Installer-Dienstes enthält.
1638	0x00000666	ERROR_PRODUCT_VERSION	Eine andere Version dieses Produkts ist bereits installiert. Installation dieser Version kann nicht fortgesetzt werden. Zum Konfigurieren oder die vorhandene Version dieses Produkts entfernen, verwenden Sie Programme hinzufügen/entfernen in der Systemsteuerung.
1639	0x00000667	ERROR_INVALID_COMMAND_LINE	Ungültiges Befehlszeilenargument. Ausführliche Befehlszeilenhilfe finden Sie im Windows Installer SDK.
1640	0x00000668	ERROR_INSTALL_REMOTE_DISALLOWED	Nur Administratoren haben die Berechtigung zum Hinzufügen, Entfernen oder während einer Terminaldienste-Remotesitzung Serversoftware konfigurieren. Wenn Sie installieren oder Software auf dem Server konfigurieren möchten, wenden Sie sich an den Netzwerkadministrator.
1641	0x00000669	ERROR_SUCCESS_REBOOT_INITIATED	Der angeforderte Vorgang wurde erfolgreich abgeschlossen. Das System wird neu gestartet werden, damit die Änderungen wirksam werden.
1642	0x0000066A	ERROR_PATCH_TARGET_NOT_FOUND	Das Updatepatch kann nicht vom Windows-Installationsdienst installiert werden, da das Programm aktualisiert werden, fehlen möglicherweise, oder das Updatepatch kann eine andere Version des Programms aktualisieren. Stellen Sie sicher, dass das Programm aktualisiert werden auf Ihrem Computer vorhanden ist und dass Sie das richtige Updatepatch.
1643	0x0000066B	ERROR_PATCH_PACKAGE_REJECTED	Das Patchpaket ist von der Systemrichtlinie nicht gestattet. Es ist nicht mit einem entsprechenden Zertifikat signiert.
1644	0x0000066C	ERROR_INSTALL_TRANSFORM_REJECTED	Eine oder mehrere Anpassungen sind von der Systemrichtlinie nicht zulässig. Sie werden nicht mit einem entsprechenden Zertifikat signiert.
1700	0x000006A4	RPC_S_INVALID_STRING_BINDING	Die String-Bindung ist ungültig.
1701	0x000006A5	RPC_S_WRONG_KIND_OF_BINDING	Die Bindungs-Zugriffsnummer hat den falschen Typ.
1702	0x000006A6	RPC_S_INVALID_BINDING	Die Bindungs-Zugriffsnummer ist ungültig.
1703	0x000006A7	RPC_S_PROTSEQ_NOT_SUPPORTED	Die RPC-Protokollsequenz wird nicht unterstützt.
1704	0x000006A8	RPC_S_INVALID_RPC_PROTSEQ	Die RPC-Protokollsequenz ist ungültig.
1705	0x000006A9	RPC_S_INVALID_STRING_UUID	Der Zeichenfolge universelle eindeutige Bezeichner (UUID) ist ungültig.
1706	0x000006AA	RPC_S_INVALID_ENDPOINT_FORMAT	Das Endpunktformat ist ungültig.
1707	0x000006AB	RPC_S_INVALID_NET_ADDR	Die Netzwerkadresse ist ungültig.



Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
1708	0x000006AC	RPC_S_NO_ENDPOINT_FOUND	Kein Endpunkt es wurde gefunden.
1709	0x000006AD	RPC_S_INVALID_TIMEOUT	Der Timeoutwert ist ungültig.
1710	0x000006AE	RPC_S_OBJECT_NOT_FOUND	Der Objekt universelle eindeutige Bezeichner (UUID) wurde nicht gefunden.
1711	0x000006AF	RPC_S_ALREADY_REGISTERED	Der Objekt universelle eindeutige Bezeichner (UUID) wurde bereits registriert.
1712	0x000006B0	RPC_S_TYPE_ALREADY_REGISTERED	Der Typ universelle eindeutige Bezeichner (UUID) wurde bereits registriert.
1713	0x000006B1	RPC_S_ALREADY_LISTENING	Der RPC-Server ist bereits überwacht.
1714	0x000006B2	RPC_S_NO_PROTSEQS_REGISTERED	Keine Protokollsequenzen wurden registriert.
1715	0x000006B3	RPC_S_NOT_LISTENING	Der RPC-Server hört nicht.
1716	0x000006B4	RPC_S_UNKNOWN_MGR_TYPE	Der Managertyp ist unbekannt.
1717	0x000006B5	RPC_S_UNKNOWN_IF	Die Schnittstelle ist unbekannt.
1718	0x000006B6	RPC_S_NO_BINDINGS	Es gibt keine Bindungen.
1719	0x000006B7	RPC_S_NO_PROTSEQS	Es gibt keine Protokollsequenzen.
1720	0x000006B8	RPC_S_CANT_CREATE_ENDPOINT	Der Endpunkt kann nicht erstellt werden.
1721	0x000006B9	RPC_S_OUT_OF_RESOURCES	Sind nicht genügend Ressourcen zum Abschließen dieses Vorgangs verfügbar.
1722	0x000006BA	RPC_S_SERVER_UNAVAILABLE	Der RPC-Server ist nicht verfügbar.
1723	0x000006BB	RPC_S_SERVER_TOO_BUSY	Der RPC-Server ist ausgelastet, um diesen Vorgang abzuschließen.
1724	0x000006BC	RPC_S_INVALID_NETWORK_OPTIONS	Die Netzwerkoptionen sind ungültig.
1725	0x000006BD	RPC_S_NO_CALL_ACTIVE	Es sind keine Remoteprozeduraufrufe in diesem Thread aktiv.
1726	0x000006BE	RPC_S_CALL_FAILED	Der Remoteprozeduraufruf ist fehlgeschlagen.
1727	0x000006BF	RPC_S_CALL_FAILED_DNE	Der Remoteprozeduraufruf ist fehlgeschlagen und wurde nicht ausgeführt.
1728	0x000006C0	RPC_S_PROTOCOL_ERROR	Ein remote Prozedur Call (RPC)-Protokollfehler ist aufgetreten.
1730	0x000006C2	RPC_S_UNSUPPORTED_TRANS_SYN	Die Transfer-Syntax wird von der RPC-Server nicht unterstützt.
1732	0x000006C4	RPC_S_UNSUPPORTED_TYPE	Der universelle eindeutige Bezeichner (UUID)-Typ wird nicht unterstützt.
1733	0x000006C5	RPC_S_INVALID_TAG	Das Tag ist ungültig.
1734	0x000006C6	RPC_S_INVALID_BOUND	Die Datenfeldgrenzen sind ungültig.
1735	0x000006C7	RPC_S_NO_ENTRY_NAME	Die Bindung enthält keinem Eintragsnamen.
1736	0x000006C8	RPC_S_INVALID_NAME_SYNTAX	Die Namenssyntax ist ungültig.
1737	0x000006C9	RPC_S_UNSUPPORTED_NAME_SYNTAX	Die Namenssyntax wird nicht unterstützt.
1739	0x000006CB	RPC_S_UUID_NO_ADDRESS	Keine Netzwerkadresse steht verwenden, um eine universelle eindeutige Bezeichner (UUID) zu erstellen.
1740	0x000006CC	RPC_S_DUPLICATE_ENDPOINT	Der Endpunkt ist ein Duplikat.
1741	0x000006CD	RPC_S_UNKNOWN_AUTHN_TYPE	Der Authentifizierungstyp ist unbekannt.
1742	0x000006CE	RPC_S_MAX_CALLS_TOO_SMALL	Die maximale Anzahl der Aufrufe ist zu klein.
1743	0x000006CF	RPC_S_STRING_TOO_LONG	Die Zeichenfolge ist zu lang.
1744	0x000006D0	RPC_S_PROTSEQ_NOT_FOUND	Die RPC-Protokollsequenz wurde nicht gefunden.
1745	0x000006D1	RPC_S_PROCNUM_OUT_OF_RANGE	Die Prozedur ist außerhalb des gültigen Bereichs.
1746	0x000006D2	RPC_S_BINDING_HAS_NO_AUTH	Die Bindung enthält keine Authentifizierungsinformationen nicht.
1747	0x000006D3	RPC_S_UNKNOWN_AUTHN_SERVICE	Der Echtheitsbestätigungsdienst ist unbekannt.
1748	0x000006D4	RPC_S_UNKNOWN_AUTHN_LEVEL	Die Authentifizierungsebene ist unbekannt.
1749	0x000006D5	RPC_S_INVALID_AUTH_IDENTITY	Der Sicherheitskontext ist ungültig.
1750	0x000006D6	RPC_S_UNKNOWN_AUTHZ_SERVICE	Der Autorisierungsdienst ist unbekannt.
1751	0x000006D7	EPT_S_INVALID_ENTRY	Der Eintrag ist ungültig.
1752	0x000006D8	EPT_S_CANT_PERFORM_OP	Der Serverendpunkt kann den Vorgang nicht ausführen.
1753	0x000006D9	EPT_S_NOT_REGISTERED	Es sind keine weiteren Endpunkte verfügbar in der Endpunktzusammenfassung.
1754	0x000006DA	RPC_S_NOTHING_TO_EXPORT	Keine Schnittstellen wurden exportiert.
1755	0x000006DB	RPC_S_INCOMPLETE_NAME	Der Eintragsname ist unvollständig.
1756	0x000006DC	RPC_S_INVALID_VERS_OPTION	Die Versionsoption ist ungültig.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
1757	0x000006DD	RPC_S_NO_MORE_MEMBERS	Es gibt keine weitere Mitglieder.
1758	0x000006DE	RPC_S_NOT_ALL_OBJS_UNEXPORTED	Es gibt nichts zu unexport.
1759	0x000006DF	RPC_S_INTERFACE_NOT_FOUND	Die Schnittstelle wurde nicht gefunden.
1760	0x000006E0	RPC_S_ENTRY_ALREADY_EXISTS	Der Eintrag ist bereits vorhanden.
1761	0x000006E1	RPC_S_ENTRY_NOT_FOUND	Der Eintrag wurde nicht gefunden.
1762	0x000006E2	RPC_S_NAME_SERVICE_UNAVAILABLE	Der Namensdienst ist nicht verfügbar.
1763	0x000006E3	RPC_S_INVALID_NAF_ID	Die Familie der Netzwerk-Adresse ist ungültig.
1764	0x000006E4	RPC_S_CANNOT_SUPPORT	Der angeforderte Vorgang wird nicht unterstützt.
1765	0x000006E5	RPC_S_NO_CONTEXT_AVAILABLE	Kein Sicherheitskontext steht Identitätswechsel zu erlauben.
1766	0x000006E6	RPC_S_INTERNAL_ERROR	Ein Remoteprozeduraufruf (RPC) ist ein interner Fehler aufgetreten.
1767	0x000006E7	RPC_S_ZERO_DIVIDE	Der RPC-Server versuchte eine Ganzzahldivision durch Null.
1768	0x000006E8	RPC_S_ADDRESS_ERROR	Adressierung Fehler im RPC-Server.
1769	0x000006E9	RPC_S_FP_DIV_ZERO	Eine Gleitkommaoperation beim RPC-Server verursachte eine Division durch Null.
1770	0x000006EA	RPC_S_FP_UNDERFLOW	Ein Gleitkommaunterlauf trat beim RPC-Server.
1771	0x000006EB	RPC_S_FP_OVERFLOW	Ein Gleitkommaüberlauf trat beim RPC-Server.
1772	0x000006EC	RPC_X_NO_MORE_ENTRIES	Die Liste der RPC-Server für die Bindung von Auto-Handles ist aufgebraucht.
1773	0x000006ED	RPC_X_SS_CHAR_TRANS_OPEN_FAIL	Kann nicht die Tabelle Zeichenübersetzungsdatei geöffnet.
1774	0x000006EE	RPC_X_SS_CHAR_TRANS_SHORT_FILE	Die Datei mit der Zeichenübersetzungstabelle hat weniger als 512 Bytes.
1775	0x000006EF	RPC_X_SS_IN_NULL_CONTEXT	Ein null-Kontexthandle wurde während eines Remoteprozeduraufrufs vom Client an den Host übergeben.
1777	0x000006F1	RPC_X_SS_CONTEXT_DAMAGED	Das Kontexthandle während eines Remoteprozeduraufrufs geändert.
1778	0x000006F2	RPC_X_SS_HANDLES_MISMATCH	Die Bindungszugriffsnummern, die an einen entfernten Prozeduraufruf übergeben wurden, stimmen nicht überein.
1779	0x000006F3	RPC_X_SS_CANNOT_GET_CALL_HANDLE	Die Stub kann das remote Prozedur Call-Handle zu erhalten.
1780	0x000006F4	RPC_X_NULL_REF_POINTER	An den Stub wurde ein null-Verweis-Zeiger übergeben.
1781	0x000006F5	RPC_X_DEUM_VALUE_OUT_OF_RANGE	Der Enumerationswert ist außerhalb des gültigen Bereichs.
1782	0x000006F6	RPC_X_BYTE_COUNT_TOO_SMALL	Die Byteanzahl ist zu klein.
1783	0x000006F7	RPC_X_BAD_STUB_DATA	Die Stub erhielt fehlerhafte Daten.
1784	0x000006F8	ERROR_INVALID_USER_BUFFER	Der angegebene Benutzerpuffer ist für den angeforderten Vorgang ungültig.
1785	0x000006F9	ERROR_UNRECOGNIZED_MEDIA	Dem Speichermedium wird nicht erkannt. Es kann nicht formatiert werden.
1786	0x000006FA	ERROR_NO_TRUST_LSA_SECRET	Die Arbeitsstation muss kein Geheimnis Vertrauen.
1787	0x000006FB	ERROR_NO_TRUST_SAM_ACCOUNT	Die Sicherheitsdatenbank auf dem Server muss kein Computerkonto für diese Arbeitsstationsvertrauensstellung.
1788	0x000006FC	ERROR_TRUSTED_DOMAIN_FAILURE	Die Vertrauensstellung zwischen der primären Domäne und der vertrauenswürdigen Domäne scheiterte.
1789	0x000006FD	ERROR_TRUSTED_RELATIONSHIP_FAILURE	Die Vertrauensstellung zwischen dieser Arbeitsstation und der primären Domäne scheiterte.
1790	0x000006FE	ERROR_TRUST_FAILURE	Die Netzwerkanmeldung ist fehlgeschlagen.
1791	0x000006FF	RPC_S_CALL_IN_PROGRESS	Fortschritt für diesen Thread wird bereits ein remote Procedure Call.
1792	0x00000700	ERROR_NETLOGON_NOT_STARTED	Es wurde versucht, sich anzumelden, aber der Netzwerk Logon-Dienst wurde nicht gestartet.
1793	0x00000701	ERROR_ACCOUNT_EXPIRED	Das Benutzerkonto ist abgelaufen.
1794	0x00000702	ERROR_REDIRECTOR_HAS_OPEN_HANDLES	Der Redirector wird verwendet und kann nicht entladen werden.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
1795	0x00000703	ERROR_PRINTER_DRIVER_ALREADY_INSTALLED	Der angegebene Druckertreiber ist bereits installiert.
1796	0x00000704	ERROR_UNKNOWN_PORT	Der angegebene Anschluss ist unbekannt.
1797	0x00000705	ERROR_UNKNOWN_PRINTER_DRIVER	Der Druckertreiber ist unbekannt.
1798	0x00000706	ERROR_UNKNOWN_PRINTPROCESSOR	Der Druckprozessor ist unbekannt.
1799	0x00000707	ERROR_INVALID_SEPARATOR_FILE	Die angegebene Trenndatei ist ungültig.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
1800	0x00000708	ERROR_INVALID_PRIORITY	Die angegebene Priorität ist ungültig.
1801	0x00000709	ERROR_INVALID_PRINTER_NAME	Der Druckername ist ungültig.
1802	0x0000070A	ERROR_PRINTER_ALREADY_EXISTS	Der Drucker ist bereits vorhanden.
1803	0x0000070B	ERROR_INVALID_PRINTER_COMMAND	Der Druckerbefehl ist ungültig.
1804	0x0000070C	ERROR_INVALID_DATATYPE	Der angegebene Datentyp ist ungültig.
1805	0x0000070D	ERROR_INVALID_ENVIRONMENT	Die angegebene Umgebung ist ungültig.
1806	0x0000070E	RPC_S_NO_MORE_BINDINGS	Es gibt keine weitere Bindungen.
1807	0x0000070F	ERROR_NOLOGON_INTERDOMAIN_TRUST_ACCOUNT	Das verwendete Konto ist ein domänenübergreifendes Vertrauenskonto. Verwenden Sie Ihr globales Benutzerkonto oder lokales Benutzerkonto auf diesen Server zugreifen.
1808	0x00000710	ERROR_NOLOGON_WORKSTATION_TRUST_ACCOUNT	Das verwendete Konto ist ein Arbeitsstationsvertrauenskonto. Verwenden Sie Ihr globales Benutzerkonto oder lokales Benutzerkonto auf diesen Server zugreifen.
1809	0x00000711	ERROR_NOLOGON_SERVER_TRUST_ACCOUNT	Das verwendete Konto ist ein Serververtrauenskonto. Verwenden Sie Ihr globales Benutzerkonto oder lokales Benutzerkonto auf diesen Server zugreifen.
1810	0x00000712	ERROR_DOMAIN_TRUST_INCONSISTENT	Der Name oder die Sicherheitskennung (SID) der angegebenen Domäne ist unvereinbar mit den Vertrauensinformationen dieser Domäne.
1811	0x00000713	ERROR_SERVER_HAS_OPEN_HANDLES	Der Server wird verwendet und kann nicht entladen werden.
1812	0x00000714	ERROR_RESOURCE_DATA_NOT_FOUND	Die angegebene Bilddatei enthält keinen Ressourcenabschnitt.
1813	0x00000715	ERROR_RESOURCE_TYPE_NOT_FOUND	Der angegebene Ressourcentyp kann nicht in der Bilddatei gefunden werden.
1814	0x00000716	ERROR_RESOURCE_NAME_NOT_FOUND	Der angegebene Ressourcenname kann nicht in der Bilddatei gefunden werden.
1815	0x00000717	ERROR_RESOURCE_LANG_NOT_FOUND	Die angegebene Ressource-Sprach-ID kann nicht in der Bilddatei gefunden werden.
1816	0x00000718	ERROR_NOT_ENOUGH_QUOTA	Nicht genug Quote ist verfügbar, um diesen Befehl zu verarbeiten.
1817	0x00000719	RPC_S_NO_INTERFACES	Keine Schnittstellen wurden registriert.
1818	0x0000071A	RPC_S_CALL_CANCELLED	Der Remoteprozeduraufruf wurde abgesagt.
1819	0x0000071B	RPC_S_BINDING_INCOMPLETE	Die Bindungsnummer enthält nicht alle benötigten Informationen.
1820	0x0000071C	RPC_S_COMM_FAILURE	Während ein remote Procedure Call ist ein Kommunikationsfehler aufgetreten.
1821	0x0000071D	RPC_S_UNSUPPORTED_AUTHN_LEVEL	Die angeforderte Echtheitsbestätigungsebene wird nicht unterstützt.
1822	0x0000071E	RPC_S_NO_PRINC_NAME	Kein Prinzipalname registriert.
1823	0x0000071F	RPC_S_NOT_RPC_ERROR	Der angegebene Fehler ist kein gültiger Windows-RPC Fehler-Code.
1824	0x00000720	RPC_S_UUID_LOCAL_ONLY	Eine UUID, die nur für diesen Computer gültig ist wurde zugewiesen.
1825	0x00000721	RPC_S_SEC_PKG_ERROR	Ein Paket spezifischen Sicherheitsfehler.
1826	0x00000722	RPC_S_NOT_CANCELLED	Thread wird nicht abgebrochen.
1827	0x00000723	RPC_X_INVALID_ES_ACTION	Ungültiger Vorgang auf dem Handle Kodierung/Dekodierung.
1828	0x00000724	RPC_X_WRONG_ES_VERSION	Inkompatible Version des Pakets serialisieren.
1829	0x00000725	RPC_X_WRONG_STUB_VERSION	Inkompatible Version des RPC-Stub.
1830	0x00000726	RPC_X_INVALID_PIPE_OBJECT	Das RPC-Pipe-Objekt ist ungültig oder beschädigt.
1831	0x00000727	RPC_X_WRONG_PIPE_ORDER	Ein ungültiger Vorgang bezog sich auf eine RPC-Pipe-Objekt.
1832	0x00000728	RPC_X_WRONG_PIPE_VERSION	Nicht unterstützte RPC-Pipe-Version.
1898	0x0000076A	RPC_S_GROUP_MEMBER_NOT_FOUND	Mitglied der Gruppe wurde nicht gefunden.
1899	0x0000076B	EPT_S_CANT_CREATE	Der Endpunkt-Mapper-Datenbank-Eintrag konnte nicht erstellt werden.
1900	0x0000076C	RPC_S_INVALID_OBJECT	Der Objekt universelle eindeutige Bezeichner (UUID) ist die nil-UUID.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
1901	0x0000076D	ERROR_INVALID_TIME	Die angegebene Zeit ist ungültig.
1902	0x0000076E	ERROR_INVALID_FORM_NAME	Der angegebene Formularname ist ungültig.
1903	0x0000076F	ERROR_INVALID_FORM_SIZE	Die angegebene Formulargröße ist ungültig.
1904	0x00000770	ERROR_ALREADY_WAITING	Auf wird das angegebene Druckerhandle bereits gewartet.
1905	0x00000771	ERROR_PRINTER_DELETED	Der angegebene Drucker wurde gelöscht.
1906	0x00000772	ERROR_INVALID_PRINTER_STATE	Der Status des Druckers ist ungültig.
1907	0x00000773	ERROR_PASSWORD_MUST_CHANGE	Das Kennwort des Benutzers muss geändert werden, bevor Sie zum ersten Mal anmelden.
1908	0x00000774	ERROR_DOMAIN_CONTROLLER_NOT_FOUND	Den Domänencontroller konnte für diese Domäne nicht finden werden.
1909	0x00000775	ERROR_ACCOUNT_LOCKED_OUT	Das angesprochene Konto ist momentan gesperrt und kann nicht werden angemeldet.
1910	0x00000776	OR_INVALID_OXID	Der Objekt-Ausführer angegeben wurde nicht gefunden.
1911	0x00000777	OR_INVALID_OID	Das angegebene Objekt wurde nicht gefunden.
1912	0x00000778	OR_INVALID_SET	Die angegebene Objektauswertung wurde nicht gefunden.
1913	0x00000779	RPC_S_SEND_INCOMPLETE	Einige Daten verbleiben in der Anfrage-Puffer gesendet werden.
1914	0x0000077A	RPC_S_INVALID_ASYNC_HANDLE	Ungültiger asynchroner remote Prozedur Call-Handle.
1915	0x0000077B	RPC_S_INVALID_ASYNC_CALL	Ungültige asynchrone RPC-Call-Handle für diesen Vorgang.
1916	0x0000077C	RPC_X_PIPE_CLOSED	Das RPC-Pipe-Objekt wurde bereits geschlossen.
1917	0x0000077D	RPC_X_PIPE_DISCIPLINE_ERROR	Der RPC-Aufruf abgeschlossen, bevor alle Pipes verarbeitet wurden.
1918	0x0000077E	RPC_X_PIPE_EMPTY	Keine weiteren Daten ist von der RPC-Pipe verfügbar.
1919	0x0000077F	ERROR_NO_SITENAME	Kein Sitename steht für diese Maschine.
1920	0x00000780	ERROR_CANT_ACCESS_FILE	Die Datei kann nicht vom System zugegriffen werden.
1921	0x00000781	ERROR_CANT_RESOLVE_FILENAME	Der Name der Datei kann nicht vom System nicht aufgelöst werden.
1922	0x00000782	RPC_S_ENTRY_TYPE_MISMATCH	Der Eintrag ist nicht dem erwarteten Typ.
1923	0x00000783	RPC_S_NOT_ALL_OBJS_EXPORTED	Nicht alle Objekt-UUIDs konnten in den angegebenen Eintrag exportiert werden.
1924	0x00000784	RPC_S_INTERFACE_NOT_EXPORTED	Schnittstelle konnte nicht in den angegebenen Eintrag exportiert werden.
1925	0x00000785	RPC_S_PROFILE_NOT_ADDED	Der angegebene Profileintrag konnte nicht hinzugefügt werden.
1926	0x00000786	RPC_S_PRF_ELT_NOT_ADDED	Das angegebene Profilelement konnte nicht hinzugefügt werden.
1927	0x00000787	RPC_S_PRF_ELT_NOT_REMOVED	Das angegebene Profilelement konnte nicht entfernt werden.
1928	0x00000788	RPC_S_GRP_ELT_NOT_ADDED	Das Gruppenelement konnte nicht hinzugefügt werden.
1929	0x00000789	RPC_S_GRP_ELT_NOT_REMOVED	Das Gruppenelement konnte nicht entfernt werden.
1930	0x0000078A	ERROR_KM_DRIVER_BLOCKED	Der Druckertreiber ist nicht kompatibel mit einer Richtlinie auf Ihrem Computer NT 4.0-Treiber blockiert aktiviert.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
2000	0x000007D0	ERROR_INVALID_PIXEL_FORMAT	Das Pixelformat ist ungültig.
2001	0x000007D1	ERROR_BAD_DRIVER	Der angegebene Treiber ist ungültig.
2002	0x000007D2	ERROR_INVALID_WINDOW_STYLE	Das Fenster-Stil oder Class-Attribut ist ungültig für diesen Vorgang.
2003	0x000007D3	ERROR_METAFILE_NOT_SUPPORTED	Der angeforderte Metafilevorgang wird nicht unterstützt.
2004	0x000007D4	ERROR_TRANSFORM_NOT_SUPPORTED	Der Transformation der angeforderte Vorgang wird nicht unterstützt.
2005	0x000007D5	ERROR_CLIPPING_NOT_SUPPORTED	Der angeforderte Clipping-Vorgang wird nicht unterstützt.
2010	0x000007DA	ERROR_INVALID_CMM	Das angegebene Farbverwaltungsmodul ist ungültig.
2011	0x000007DB	ERROR_INVALID_PROFILE	Das angegebene Farbprofil ist ungültig.
2012	0x000007DC	ERROR_TAG_NOT_FOUND	Das angegebene Tag wurde nicht gefunden.
2013	0x000007DD	ERROR_TAG_NOT_PRESENT	Ein erforderliche Tag ist nicht vorhanden.
2014	0x000007DE	ERROR_DUPLICATE_TAG	Das angegebene Tag ist bereits vorhanden.
2015	0x000007DF	ERROR_PROFILE_NOT_ASSOCIATED_WITH_DEVICE	Das angegebene Farbprofil ist nicht mit jedem Gerät verbunden.
2016	0x000007E0	ERROR_PROFILE_NOT_FOUND	Das angegebene Farbprofil wurde nicht gefunden.
2017	0x000007E1	ERROR_INVALID_COLORSPACE	Der angegebenen Farbraum ist ungültig.
2018	0x000007E2	ERROR_ICM_NOT_ENABLED	Farbverwaltung ist nicht aktiviert.
2019	0x000007E3	ERROR_DELETING_ICM_XFORM	Beim Löschen der Farbe-Transformation ist ein Fehler aufgetreten.
2020	0x000007E4	ERROR_INVALID_TRANSFORM	Die angegebene Farbe-Transformation ist ungültig.
2021	0x000007E5	ERROR_COLORSPACE_MISMATCH	Die angegebene Transformation stimmt nicht überein, die Bitmap-Farbraum.
2022	0x000007E6	ERROR_INVALID_COLORINDEX	Der angegebene benannte Farbindex ist nicht im Profil vorhanden.
2108	0x0000083C	ERROR_CONNECTED_OTHER_PASSWORD	Die Netzwerkverbindung wurde erfolgreich hergestellt, aber der Benutzer musste aufgefordert, ein Kennwort nicht dem ursprünglich angegebenen übereinstimmte.
2202	0x0000089A	ERROR_BAD_USERNAME	Der angegebene Benutzername ist ungültig.
2250	0x000008CA	ERROR_NOT_CONNECTED	Diese Netzwerkverbindung ist nicht vorhanden.
2401	0x00000961	ERROR_OPEN_FILES	Diese Netzwerkverbindung hat Dateien öffnen oder anstehende Anforderungen.
2402	0x00000962	ERROR_ACTIVE_CONNECTIONS	Aktive Verbindungen sind noch vorhanden.
2404	0x00000964	ERROR_DEVICE_IN_USE	Das Gerät wird von einem aktiven Prozess verwendet und kann nicht getrennt werden.
2500	0x000009C4	ERROR_PKINIT_FAILURE	Das Kerberos-Protokoll-Fehler während der Überprüfung des KDC-Zertifikats während der Smartcard-Anmeldung.
2501	0x000009C5	ERROR_SMARTCARD_SUBSYSTEM_FAILURE	Das Kerberos-Protokoll-Fehler beim Versuch, das Smartcard-Subsystem zu nutzen.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
3000	0x00000BB8	ERROR_UNKNOWN_PRINT_MONITOR	Der angegebene Druckmonitor ist unbekannt.
3001	0x00000BB9	ERROR_PRINTER_DRIVER_IN_USE	Der angegebene Druckertreiber wird zurzeit verwendet.
3002	0x00000BBA	ERROR_SPOOL_FILE_NOT_FOUND	Die Spooldatei wurde nicht gefunden.
3003	0x00000BBB	ERROR_SPL_NO_STARTDOC	Ein StartDocPrinter wurde nicht ausgeführt.
3004	0x00000BBC	ERROR_SPL_NO_ADDJOB	Ein AddJob-Aufruf wurde nicht ausgestellt.
3005	0x00000BBD	ERROR_PRINT_PROCESSOR_ALREADY_INSTALLED	Der angegebene Druckprozessor ist bereits installiert.
3006	0x00000BBE	ERROR_PRINT_MONITOR_ALREADY_INSTALLED	Der angegebene Druckmonitor ist bereits installiert.
3007	0x00000BBF	ERROR_INVALID_PRINT_MONITOR	Der angegebene Druckmonitor muss nicht die benötigten Funktionen.
3008	0x00000BC0	ERROR_PRINT_MONITOR_IN_USE	Der angegebene Druckmonitor ist momentan in Gebrauch.
3009	0x00000BC1	ERROR_PRINTER_HAS_JOBS_QUEUED	Der angeforderte Vorgang ist nicht zulässig, wenn gibt es Arbeitsplätze, die an den Drucker in eine Warteschlange gestellt.
3010	0x00000BC2	ERROR_SUCCESS_REBOOT_REQUIRED	Der angeforderte Vorgang ist erfolgreich. Änderungen werden nicht wirksam, bis das System neu gestartet wird.
3011	0x00000BC3	ERROR_SUCCESS_RESTART_REQUIRED	Der angeforderte Vorgang ist erfolgreich. Änderungen werden nicht wirksam, bis der Dienst neu gestartet wird.
3012	0x00000BC4	ERROR_PRINTER_NOT_FOUND	Keine Drucker es wurden gefunden.
4000	0x00000FA0	ERROR_WINS_INTERNAL	WINS-Fehler beim Verarbeiten des Befehls.
4001	0x00000FA1	ERROR_CAN_NOT_DEL_LOCAL_WINS	Der lokale WINS können nicht gelöscht werden.
4002	0x00000FA2	ERROR_STATIC_INIT	Die Einfuhr aus der Datei ist fehlgeschlagen.
4003	0x00000FA3	ERROR_INC_BACKUP	Die Sicherung ist fehlgeschlagen. Erfolgte eine vollständige Sicherung vor?
4004	0x00000FA4	ERROR_FULL_BACKUP	Die Sicherung ist fehlgeschlagen. Überprüfen Sie das Verzeichnis, in dem Sie die Datenbank sichern möchten.
4005	0x00000FA5	ERROR_REC_NON_EXISTENT	Der Name ist nicht in der WINS-Datenbank vorhanden.
4006	0x00000FA6	ERROR_RPL_NOT_ALLOWED	Replikation mit einem nicht konfigurierten Partner ist nicht erlaubt.
4100	0x00001004	ERROR_DHCP_ADDRESS_CONFLICT	Der DHCP-Client hat eine IP-Adresse erhalten, die bereits im Netzwerk verwendet wird. Die lokale Schnittstelle wird deaktiviert werden, bis der DHCP-Client eine neue Adresse abrufen kann.
4200	0x00001068	ERROR_WMI_GUID_NOT_FOUND	Die GUID übergeben wurde von einem WMI-Datenprovider nicht als gültig anerkannt.
4201	0x00001069	ERROR_WMI_INSTANCE_NOT_FOUND	Den Namen der Instanz übergeben wurde von einem WMI-Datenprovider nicht als gültig anerkannt.
4202	0x0000106A	ERROR_WMI_ITEMID_NOT_FOUND	Die Daten-Element-ID übergeben wurde von einem WMI-Datenprovider nicht als gültig anerkannt.
4203	0x0000106B	ERROR_WMI_TRY_AGAIN	Die WMI-Anforderung konnte nicht abgeschlossen werden und sollte wiederholt werden.
4204	0x0000106C	ERROR_WMI_DP_NOT_FOUND	Der WMI-Datenanbieter konnte nicht gefunden werden.
4205	0x0000106D	ERROR_WMI_UNRESOLVED_INSTANCE_REF	Der WMI-Datenanbieter verweist auf eine Instanz-Gruppe, die nicht registriert wurde.
4206	0x0000106E	ERROR_WMI_ALREADY_ENABLED	Der WMI-Datenblock oder die Ereignisbenachrichtigung wurde bereits aktiviert.
4207	0x0000106F	ERROR_WMI_GUID_DISCONNECTED	Der WMI-Datenblock ist nicht mehr verfügbar.
4208	0x00001070	ERROR_WMI_SERVER_UNAVAILABLE	Der WMI-Datendienst ist nicht verfügbar.
4209	0x00001071	ERROR_WMI_DP_FAILED	Der WMI-Datenanbieter konnte die Anforderung durchzuführen.
4210	0x00001072	ERROR_WMI_INVALID_MOF	Die WMI-MOF-Informationen sind ungültig.
4211	0x00001073	ERROR_WMI_INVALID_REGINFO	Die WMI-Registrierungsinformationen ist nicht gültig.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
4212	0x00001074	ERROR_WMI_ALREADY_DISABLED	Der WMI-Datenblock oder die Ereignisbenachrichtigung wurde bereits deaktiviert.
4213	0x00001075	ERROR_WMI_READ_ONLY	Der WMI-Objekt oder Daten-Datenblock ist schreibgeschützt.
4214	0x00001076	ERROR_WMI_SET_FAILURE	Der WMI-Objekt oder Daten-Datenblock konnte nicht geändert werden.
4300	0x000010CC	ERROR_INVALID_MEDIA	Die Medienkennung stellt kein gültiges Medium dar.
4301	0x000010CD	ERROR_INVALID_LIBRARY	Die Bibliothek-ID stellt keine gültige Bibliothek dar.
4302	0x000010CE	ERROR_INVALID_MEDIA_POOL	Die Medienkennung Pool stellt keinen gültigen Medienpool dar.
4303	0x000010CF	ERROR_DRIVE_MEDIA_MISMATCH	Das Laufwerk und das Medium sind nicht kompatibel, oder in verschiedenen Bibliotheken vorhanden.
4304	0x000010D0	ERROR_MEDIA_OFFLINE	Das Medium muss derzeit in einer Offlinebibliothek vorhanden und online sein, um diesen Vorgang auszuführen.
4305	0x000010D1	ERROR_LIBRARY_OFFLINE	Der Vorgang kann nicht auf einer Offlinebibliothek ausgeführt werden.
4306	0x000010D2	ERROR_EMPTY	Der Bibliothek, das Laufwerk oder Medien-Pool ist leer.
4307	0x000010D3	ERROR_NOT_EMPTY	Bibliothek, das Laufwerk oder der Medienpool muss zum Ausführen dieses Vorgangs leer sein.
4308	0x000010D4	ERROR_MEDIA_UNAVAILABLE	Kein Medium ist derzeit in diesem Medienpool oder Bibliothek.
4309	0x000010D5	ERROR_RESOURCE_DISABLED	Eine Ressource, die für diesen Vorgang erforderlich ist deaktiviert.
4310	0x000010D6	ERROR_INVALID_CLEANER	Die Medienkennung vertritt keinen gültigen Cleaner.
4311	0x000010D7	ERROR_UNABLE_TO_CLEAN	Das Laufwerk kann nicht gesäubert werden oder Reinigung nicht unterstützt.
4312	0x000010D8	ERROR_OBJECT_NOT_FOUND	Die Objekt-ID stellt kein gültiges Objekt dar.
4313	0x000010D9	ERROR_DATABASE_FAILURE	Nicht in der Lage zu lesen oder schreiben in die Datenbank.
4314	0x000010DA	ERROR_DATABASE_FULL	Die Datenbank ist voll.
4315	0x000010DB	ERROR_MEDIA_INCOMPATIBLE	Das Medium ist nicht kompatibel mit dem Gerät oder Medium-Pool.
4316	0x000010DC	ERROR_RESOURCE_NOT_PRESENT	Die Ressource benötigt für diesen Vorgang ist nicht vorhanden.
4317	0x000010DD	ERROR_INVALID_OPERATION	Die Vorgangs-ID ist ungültig.
4318	0x000010DE	ERROR_MEDIA_NOT_AVAILABLE	Das Medium ist nicht eingebaut oder für den Einsatz bereit.
4319	0x000010DF	ERROR_DEVICE_NOT_AVAILABLE	Das Gerät ist nicht für den Einsatz bereit.
4320	0x000010E0	ERROR_REQUEST_REFUSED	Der Operator oder der Administrator hat die Anforderung abgelehnt.
4321	0x000010E1	ERROR_INVALID_DRIVE_OBJECT	Die Laufwerk-ID repräsentiert kein gültiges Laufwerk.
4322	0x000010E2	ERROR_LIBRARY_FULL	Bibliothek ist voll. Kein Steckplatz steht zur Verfügung.
4323	0x000010E3	ERROR_MEDIUM_NOT_ACCESSIBLE	Der Transport kann nicht das Medium zugreifen.
4324	0x000010E4	ERROR_UNABLE_TO_LOAD_MEDIUM	Nicht in der Lage, das Medium in das Laufwerk zu laden.
4325	0x000010E5	ERROR_UNABLE_TO_INVENTORY_DRIVE	Kann nicht zum Abrufen des Status über das Laufwerk.
4326	0x000010E6	ERROR_UNABLE_TO_INVENTORY_SLOT	Kann nicht zum Abrufen des Status über den Steckplatz.
4327	0x000010E7	ERROR_UNABLE_TO_INVENTORY_TRANSPORT	Kann nicht zum Abrufen des Status über den Transport.
4328	0x000010E8	ERROR_TRANSPORT_FULL	Den Transport kann nicht verwenden werden, da es bereits verwendet wird.
4329	0x000010E9	ERROR_CONTROLLING_IEPORT	Kann nicht öffnen oder schließen das Einlegen bzw. Ausgabefach.



Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
4330	0x000010EA	ERROR_UNABLE_TO_EJECT_MOUNTED_MEDIA	Nicht in der Lage, den Datenträger auszuwerfen, weil es im Auto ist.
4331	0x000010EB	ERROR_CLEANER_SLOT_SET	Ein Reinigungssteckplatz ist bereits reserviert.
4332	0x000010EC	ERROR_CLEANER_SLOT_NOT_SET	Ein Reinigungssteckplatz ist nicht reserviert.
4333	0x000010ED	ERROR_CLEANER_CARTRIDGE_SPENT	Die Reinigungskassette hat die maximale Anzahl von Laufwerk Reinigungen durchgeführt.
4334	0x000010EE	ERROR_UNEXPECTED_OMID	Unerwartete auf Medium Bezeichner.
4335	0x000010EF	ERROR_CANT_DELETE_LAST_ITEM	Das letzte verbleibende Element in dieser Gruppe oder Ressource kann nicht gelöscht werden.
4336	0x000010F0	ERROR_MESSAGE_EXCEEDS_MAX_SIZE	Der jeweiligen Meldung überschreitet die maximale zulässige Größe für diesen Parameter.
4337	0x000010F1	ERROR_VOLUME_CONTAINS_SYS_FILES	Der Band enthält System- oder Auslagerungsdateien.
4338	0x000010F2	ERROR_INDIGENOUS_TYPE	Der Medientyp kann nicht aus dieser Bibliothek entfernt werden, da mindestens ein Laufwerk in der Bibliothek meldet, dass es diesen Medientyp unterstützen kann.
4339	0x000010F3	ERROR_NO_SUPPORTING_DRIVES	Dieses Offlinemedium kann auf diesem System bereitgestellt werden, da keine aktivierten Laufwerke vorhanden sind, die benutzt werden können.
4340	0x000010F4	ERROR_CLEANER_CARTRIDGE_INSTALLED	Eine Reinigungskassette ist vorhanden in der Bandbibliothek.
4350	0x000010FE	ERROR_FILE_OFFLINE	Der Remotespeicherdienst konnte nicht die Datei erinnern.
4351	0x000010FF	ERROR_REMOTE_STORAGE_NOT_ACTIVE	Der Remotespeicherdienst ist derzeit nicht betriebsbereit.
4352	0x00001100	ERROR_REMOTE_STORAGE_MEDIA_ERROR	Der Remotespeicherdienst ist einen Medienfehler aufgetreten.
4390	0x00001126	ERROR_NOT_A_REPARSE_POINT	Die Datei oder das Verzeichnis ist kein Analysepunkt.
4391	0x00001127	ERROR_REPARSE_ATTRIBUTE_CONFLICT	Das Analysepunkte Punkt-Attribut kann nicht festgelegt werden, weil es mit einem vorhandenen Attribut widerspricht.
4392	0x00001128	ERROR_INVALID_REPARSE_DATA	Die im Analysepunktpuffer vorhandenen Daten ist ungültig.
4393	0x00001129	ERROR_REPARSE_TAG_INVALID	Die Marke vorhanden im Analysepunktpuffer ist ungültig.
4394	0x0000112A	ERROR_REPARSE_TAG_MISMATCH	Es besteht eine Diskrepanz zwischen dem in der Anfrage angegebenen Tag und dem Tag in der Analysepunkt vorhanden.
4500	0x00001194	ERROR_VOLUME_NOT_SIS_ENABLED	Single Instance Storage steht nicht auf diesem Volume.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
5001	0x00001389	ERROR_DEPENDENT_RESOURCE_EXISTS	Die Clusterressource kann nicht in eine andere Gruppe verschoben werden, da andere Ressourcen abhängig sind.
5002	0x0000138A	ERROR_DEPENDENCY_NOT_FOUND	Die Clusterressourcenabhängigkeit wurde nicht gefunden.
5003	0x0000138B	ERROR_DEPENDENCY_ALREADY_EXISTS	Die Clusterressource kann nicht der angegebenen Ressource abhängig gemacht werden, da er bereits angewiesen ist.
5004	0x0000138C	ERROR_RESOURCE_NOT_ONLINE	Die Cluster-Ressource ist nicht online.
5005	0x0000138D	ERROR_HOST_NODE_NOT_AVAILABLE	Ein Clusterknoten ist nicht verfügbar für diesen Vorgang.
5006	0x0000138E	ERROR_RESOURCE_NOT_AVAILABLE	Die Cluster-Ressource ist nicht verfügbar.
5007	0x0000138F	ERROR_RESOURCE_NOT_FOUND	Die Clusterressource konnte nicht gefunden werden.
5008	0x00001390	ERROR_SHUTDOWN_CLUSTER	Der Cluster wird heruntergefahren.
5009	0x00001391	ERROR_CANT_EVICT_ACTIVE_NODE	Ein Clusterknoten kann nicht vom Cluster entfernt, es sei denn der Knoten ausfällt.
5010	0x00001392	ERROR_OBJECT_ALREADY_EXISTS	Das Objekt ist bereits vorhanden.
5011	0x00001393	ERROR_OBJECT_IN_LIST	Das Objekt ist bereits in der Liste.
5012	0x00001394	ERROR_GROUP_NOT_AVAILABLE	Die Cluster-Gruppe ist nicht verfügbar für alle neuen Anforderungen.
5013	0x00001395	ERROR_GROUP_NOT_FOUND	Die Cluster-Gruppe konnte nicht gefunden werden.
5014	0x00001396	ERROR_GROUP_NOT_ONLINE	Der Vorgang konnte nicht abgeschlossen werden, da die Clustergruppe nicht online ist.
5015	0x00001397	ERROR_HOST_NODE_NOT_RESOURCE_OWNER	Der Clusterknoten ist nicht der Besitzer der Ressource.
5016	0x00001398	ERROR_HOST_NODE_NOT_GROUP_OWNER	Der Clusterknoten ist nicht der Besitzer der Gruppe.
5017	0x00001399	ERROR_RESMON_CREATE_FAILED	Die Clusterressource konnte in dem angegebenen Ressourcenmonitor nicht erstellt werden.
5018	0x0000139A	ERROR_RESMON_ONLINE_FAILED	Die Clusterressource konnte vom Ressourcenmonitor nicht online geschaltet werden.
5019	0x0000139B	ERROR_RESOURCE_ONLINE	Der Vorgang konnte nicht abgeschlossen werden, da die Cluster-Ressource online ist.
5020	0x0000139C	ERROR_QUORUM_RESOURCE	Die Clusterressource konnte nicht gelöscht oder offline geschaltet werden, da es die Quorumressource ist.
5021	0x0000139D	ERROR_NOT_QUORUM_CAPABLE	Der Cluster konnte nicht der angegebenen Ressource eine Quorumressource, da es nicht wird eine Quorumressource kann.
5022	0x0000139E	ERROR_CLUSTER_SHUTTING_DOWN	Die Clustersoftware wird heruntergefahren.
5023	0x0000139F	ERROR_INVALID_STATE	Die Gruppe oder Ressource ist nicht im richtigen Zustand zum Ausführen des angeforderten Vorgangs.
5024	0x000013A0	ERROR_RESOURCE_PROPERTIES_STORED	Die Eigenschaften wurden gespeichert, aber nicht alle Änderungen werden wirksam, bis das nächste Mal, das die Ressource online geschaltet wird.
5025	0x000013A1	ERROR_NOT_QUORUM_CLASS	Der Cluster konnte nicht der angegebenen Ressource eine Quorumressource, da es nicht zu einer freigegebenen Speicherklasse gehört.
5026	0x000013A2	ERROR_CORE_RESOURCE	Die Clusterressource konnte nicht gelöscht werden, da es sich um eine Kernressource handelt.
5027	0x000013A3	ERROR_QUORUM_RESOURCE_ONLINE_FAILED	Die Quorumressource konnte nicht online geschaltet.
5028	0x000013A4	ERROR_QUORUMLOG_OPEN_FAILED	Die Quorum-Protokolldatei konnten nicht erstellt oder montiert erfolgreich.
5029	0x000013A5	ERROR_CLUSTERLOG_CORRUPT	Die Clusterprotokolldatei ist fehlerhaft.
5030	0x000013A6	ERROR_CLUSTERLOG_RECORD_EXCEEDS_MAXSIZE	Der Datensatz konnte nicht in das Clusterprotokoll geschrieben werden, da sie die maximale Größe überschreitet.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
5031	0x000013A7	ERROR_CLUSTERLOG_EXCEEDS_MAXSIZE	Das Clusterprotokoll überschreitet die maximale Größe.
5032	0x000013A8	ERROR_CLUSTERLOG_CHKPOINT_NOT_FOUND	Kein Prüfpunktdatensatz wurde im Clusterprotokoll gefunden.
5033	0x000013A9	ERROR_CLUSTERLOG_NOT_ENOUGH_SPACE	Der minimal benötigte Speicherplatz für die Protokollierung erforderlich nicht verfügbar ist.
5034	0x000013AA	ERROR_QUORUM_OWNER_ALIVE	Der Clusterknoten konnte die Kontrolle über die Quorumressource zu nehmen, da die Ressource einem anderen aktiven Knoten gehört.
5035	0x000013AB	ERROR_NETWORK_NOT_AVAILABLE	Ein Cluster-Netzwerk ist nicht verfügbar für diesen Vorgang.
5036	0x000013AC	ERROR_NODE_NOT_AVAILABLE	Ein Clusterknoten ist nicht verfügbar für diesen Vorgang.
5037	0x000013AD	ERROR_ALL_NODES_NOT_AVAILABLE	Alle Clusterknoten müssen ausgeführt werden, um diesen Vorgang auszuführen.
5038	0x000013AE	ERROR_RESOURCE_FAILED	Eine Clusterressource ist fehlgeschlagen.
5039	0x000013AF	ERROR_CLUSTER_INVALID_NODE	Der Clusterknoten ist nicht gültig.
5040	0x000013B0	ERROR_CLUSTER_NODE_EXISTS	Der Clusterknoten ist bereits vorhanden.
5041	0x000013B1	ERROR_CLUSTER_JOIN_IN_PROGRESS	Ein Knoten ist dabei, den Cluster beitreten.
5042	0x000013B2	ERROR_CLUSTER_NODE_NOT_FOUND	Der Clusterknoten wurde nicht gefunden.
5043	0x000013B3	ERROR_CLUSTER_LOCAL_NODE_NOT_FOUND	Die Informationen zur Cluster-lokalen Knoten wurde nicht gefunden.
5044	0x000013B4	ERROR_CLUSTER_NETWORK_EXISTS	Das Clusternetzwerk ist bereits vorhanden.
5045	0x000013B5	ERROR_CLUSTER_NETWORK_NOT_FOUND	Das Clusternetzwerk wurde nicht gefunden.
5046	0x000013B6	ERROR_CLUSTER_NETINTERFACE_EXISTS	Die Cluster-Netzwerk-Schnittstelle ist bereits vorhanden.
5047	0x000013B7	ERROR_CLUSTER_NETINTERFACE_NOT_FOUND	Die Cluster-Netzwerk-Schnittstelle wurde nicht gefunden.
5048	0x000013B8	ERROR_CLUSTER_INVALID_REQUEST	Die Cluster-Anforderung gilt nicht für dieses Objekt.
5049	0x000013B9	ERROR_CLUSTER_INVALID_NETWORK_PROVIDER	Die Cluster-Netzwerk-Provider ist nicht gültig.
5050	0x000013BA	ERROR_CLUSTER_NODE_DOWN	Der Clusterknoten ist unten.
5051	0x000013BB	ERROR_CLUSTER_NODE_UNREACHABLE	Der Clusterknoten ist nicht erreichbar.
5052	0x000013BC	ERROR_CLUSTER_NODE_NOT_MEMBER	Der Clusterknoten ist kein Mitglied des Clusters.
5053	0x000013BD	ERROR_CLUSTER_JOIN_NOT_IN_PROGRESS	Eine Cluster-Join-Operation wird nicht ausgeführt.
5054	0x000013BE	ERROR_CLUSTER_INVALID_NETWORK	Das Clusternetzwerk ist nicht gültig.
5056	0x000013C0	ERROR_CLUSTER_NODE_UP	Der Clusterknoten ist oben.
5057	0x000013C1	ERROR_CLUSTER_IPADDR_IN_USE	Die IP-Adresse des Clusters wird bereits verwendet.
5058	0x000013C2	ERROR_CLUSTER_NODE_NOT_PAUSED	Der Clusterknoten wurde nicht angehalten.
5059	0x000013C3	ERROR_CLUSTER_NO_SECURITY_CONTEXT	Keine Cluster-Sicherheitskontext steht zur Verfügung.
5060	0x000013C4	ERROR_CLUSTER_NETWORK_NOT_INTERNAL	Das Clusternetzwerk ist nicht für die interne Clusterkommunikation konfiguriert.
5061	0x000013C5	ERROR_CLUSTER_NODE_ALREADY_UP	Der Clusterknoten ist bereits oben.
5062	0x000013C6	ERROR_CLUSTER_NODE_ALREADY_DOWN	Der Clusterknoten ist bereits nach unten.
5063	0x000013C7	ERROR_CLUSTER_NETWORK_ALREADY_ONLINE	Das Clusternetzwerk ist bereits online.
5064	0x000013C8	ERROR_CLUSTER_NETWORK_ALREADY_OFFLINE	Das Clusternetzwerk ist bereits offline.
5065	0x000013C9	ERROR_CLUSTER_NODE_ALREADY_MEMBER	Der Clusterknoten ist bereits Mitglied des Clusters.
5066	0x000013CA	ERROR_CLUSTER_LAST_INTERNAL_NETWORK	Das Clusternetzwerk ist das einzige, das für die interne Clusterkommunikation zwischen mindestens zwei aktiven Clusterknoten konfiguriert. Die interne Kommunikationsfähigkeit kann aus dem Netzwerk entfernt werden.
5067	0x000013CB	ERROR_CLUSTER_NETWORK_HAS_DEPENDENTS	Einem oder mehreren Clusterressourcen abhängen im Netzwerk, Service für Kunden. Die Möglichkeit des Client-Zugriff kann nicht aus dem Netzwerk entfernt werden.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
5068	0x000013CC	ERROR_INVALID_OPERATION_ON_QUORUM	Dieser Vorgang kann nicht werden ausgeführt auf die Cluster-Ressource, da es die Quorumressource. Sie können nicht die Quorumressource offline zu bringen oder ändern die Liste der möglichen Besitzer.
5069	0x000013CD	ERROR_DEPENDENCY_NOT_ALLOWED	Die Cluster-Quorum-Ressource ist nicht zulässig, keine Abhängigkeiten haben.
5070	0x000013CE	ERROR_CLUSTER_NODE_PAUSED	Der Clusterknoten ist angehalten.
5071	0x000013CF	ERROR_NODE_CANT_HOST_RESOURCE	Die Clusterressource konnte nicht online geschaltet werden. Der Besitzerknoten kann diese Ressource nicht ausführen.
5072	0x000013D0	ERROR_CLUSTER_NODE_NOT_READY	Der Clusterknoten ist nicht bereit sind, die angeforderte Operation auszuführen.
5073	0x000013D1	ERROR_CLUSTER_NODE_SHUTTING_DOWN	Der Clusterknoten wird heruntergefahren.
5074	0x000013D2	ERROR_CLUSTER_JOIN_ABORTED	Der Cluster-Join-Vorgang wurde abgebrochen.
5075	0x000013D3	ERROR_CLUSTER_INCOMPATIBLE_VERSIONS	Die Cluster-Join-Operation scheiterte inkompatible Softwareversionen zwischen beitretenden Knotens und seiner Auftraggeber.
5076	0x000013D4	ERROR_CLUSTER_MAXNUM_OF_RESOURCES_EXCEEDED	Diese Ressource kann nicht erstellt werden, da der Cluster den Grenzwert für die Anzahl der Ressourcen erreicht hat, die sie überwachen kann.
5077	0x000013D5	ERROR_CLUSTER_SYSTEM_CONFIG_CHANGE	Die Konfiguration des Systems während des Cluster beitreten oder Form verändert. Der Join oder Formular-Vorgang wurde abgebrochen.
5078	0x000013D6	ERROR_CLUSTER_RESOURCE_TYPE_NOT_FOUND	Der angegebene Ressourcentyp wurde nicht gefunden.
5079	0x000013D7	ERROR_CLUSTER_RESTYPE_NOT_SUPPORTED	Der angegebene Knoten wird eine Ressource dieses Typs nicht unterstützt. Dies kann durch Versionsinkonsistenzen oder durch das Fehlen der Ressourcen-DLL auf diesem Knoten sein.
5080	0x000013D8	ERROR_CLUSTER_RESNAME_NOT_FOUND	Der angegebene Ressourcename wird von dieser Ressourcen-DLL unterstützt. Dies ist möglicherweise aufgrund einer schlechten (oder geänderte) Nachnamen auf die Ressourcen-DLL.
5081	0x000013D9	ERROR_CLUSTER_NO_RPC_PACKAGES_REGISTERED	Mit dem RPC-Server konnte kein Authentifizierungspaket registriert werden.
5082	0x000013DA	ERROR_CLUSTER_OWNER_NOT_IN_PREFLIST	Sie können nicht die Gruppe nicht online schalten, weil der Besitzer der Gruppe nicht in der Liste der bevorzugten für die Gruppe. Um den Besitzerknoten für die Gruppe zu ändern, verschieben Sie die Gruppe.
5083	0x000013DB	ERROR_CLUSTER_DATABASE_SEQMISMATCH	Der Beitrittsvorgang ist fehlgeschlagen, da die Sequenznummer der Cluster-Datenbank geändert hat oder nicht kompatibel mit dem Spind-Knoten ist. Dies kann während einer Verknüpfungoperation passieren, wenn die Clusterdatenbank während der Verknüpfung geändert wurde.
5084	0x000013DC	ERROR_RESMON_INVALID_STATE	Der Ressourcenmonitor ist nicht möglich, die Fail-Operation durchgeführt werden, während die Ressource im aktuellen Zustand ist. Dies kann passieren, wenn die Ressource in einem wartenden Zustand befindet.
5085	0x000013DD	ERROR_CLUSTER_GUM_NOT_LOCKER	Ein nicht Locker-Code hat eine Anforderung, die Sperre für eine globale Aktualisierungen zu reservieren.
5086	0x000013DE	ERROR_QUORUM_DISK_NOT_FOUND	Der Quorumdatenträger konnte vom Clusterdienst nicht gefunden werden.
5087	0x000013DF	ERROR_DATABASE_BACKUP_CORRUPT	Die Sicherung Cluster-Datenbank ist möglicherweise beschädigt.
5088	0x000013E0	ERROR_CLUSTER_NODE_ALREADY_HAS_DFS_ROOT	Ein DFS-Stamm existiert bereits in diesem Clusterknoten.
5089	0x000013E1	ERROR_RESOURCE_PROPERTY_UNCHANGEABLE	Ein Versuch, eine Ressourceneigenschaft zu ändern ist fehlgeschlagen, da der Konflikt mit einer anderen vorhandenen Eigenschaft.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
5890	0x00001702	ERROR_CLUSTER_MEMBERSHIP_INVALID_STATE	Ein Vorgang bezog, die inkompatibel mit dem aktuellen Status der Mitgliedschaft des Knotens ist.
5891	0x00001703	ERROR_CLUSTER_QUORUMLOG_NOT_FOUND	Die Quorumressource enthält nicht die Quorum-Protokolldatei.
5892	0x00001704	ERROR_CLUSTER_MEMBERSHIP_HALT	Das Mitgliedschaftsmodul angefordert Herunterfahren des Clusterdienstes auf diesem Knoten.
5893	0x00001705	ERROR_CLUSTER_INSTANCE_ID_MISMATCH	Der Beitrittsvorgang ist fehlgeschlagen, da die Cluster-Instanz-ID des beitretenden Knotens nicht mit die Cluster-Instanz-ID des Sponsorknoten übereinstimmt.
5894	0x00001706	ERROR_CLUSTER_NETWORK_NOT_FOUND_FOR_IP	Eine passende Netzwerk für die angegebene IP-Adresse konnte nicht gefunden werden. Bitte geben Sie eine Subnetzmaske und ein Clusternetzwerk.
5895	0x00001707	ERROR_CLUSTER_PROPERTY_DATA_TYPE_MISMATCH	Der eigentliche Datentyp der Eigenschaft entsprach nicht den erwarteten Datentyp der Eigenschaft.
5896	0x00001708	ERROR_CLUSTER_EVICT_WITHOUT_CLEANUP	Der Clusterknoten wurde erfolgreich vom Cluster entfernt. Der Knoten wurde nicht bereinigt, da es die entfernen-Cleanup-Funktionalität nicht unterstützt.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
6000	0x00001770	ERROR_ENCRYPTION_FAILED	Die angegebene Datei konnte nicht verschlüsselt werden.
6001	0x00001771	ERROR_DECRYPTION_FAILED	Die angegebene Datei konnte nicht entschlüsselt werden.
6002	0x00001772	ERROR_FILE_ENCRYPTED	Die angegebene Datei ist verschlüsselt und der Benutzer verfügt nicht über die Fähigkeit zur Entschlüsselung.
6003	0x00001773	ERROR_NO_RECOVERY_POLICY	Es gibt keine gültige Verschlüsselung Wiederherstellungsrichtlinie für dieses System.
6004	0x00001774	ERROR_NO_EFS	Der erforderliche Verschlüsselung-Treiber ist nicht für dieses System geladen.
6005	0x00001775	ERROR_WRONG_EFS	Die Datei wurde mit einem Treiber andere Verschlüsselung verschlüsselt, als derzeit geladen wird.
6006	0x00001776	ERROR_NO_USER_KEYS	Es gibt keine EFS-Schlüssel für den Benutzer definiert.
6007	0x00001777	ERROR_FILE_NOT_ENCRYPTED	Die angegebene Datei ist nicht verschlüsselt.
6008	0x00001778	ERROR_NOT_EXPORT_FORMAT	Die angegebene Datei ist nicht in das definierte EFS-Exportformat.
6009	0x00001779	ERROR_FILE_READ_ONLY	Die angegebene Datei ist schreibgeschützt.
6010	0x0000177A	ERROR_DIR_EFS_DISALLOWED	Das Verzeichnis wurde für die Verschlüsselung deaktiviert.
6011	0x0000177B	ERROR_EFS_SERVER_NOT_TRUSTED	Der Server ist nicht für RAS Verschlüsselungsvorgang vertraut.
6012	0x0000177C	ERROR_BAD_RECOVERY_POLICY	Wiederherstellungsrichtlinie für dieses System enthält ungültiges Wiederherstellungszertifikat.
6013	0x0000177D	ERROR_EFS_ALG_BLOB_TOO_BIG	Auf die Quelldatei verwendete Verschlüsselungsalgorithmus benötigt einen größeren zentralen Puffer als diejenige auf die Zieldatei.
6014	0x0000177E	ERROR_VOLUME_NOT_SUPPORT_EFS	Die Datenträgerpartition unterstützt keine Dateiverschlüsselung.
6118	0x000017E6	ERROR_NO_BROWSER_SERVERS_FOUND	Die Liste der Server für diese Arbeitsgruppe ist derzeit nicht verfügbar.
6200	0x00001838	SCHED_E_SERVICE_NOT_LOCALSYSTEM	Der Taskplannerdienst muss konfiguriert werden, um in das System-Konto ordnungsgemäß auszuführen. Einzelne Aufgaben können so konfiguriert werden, in anderen Konten ausgeführt werden.
7001	0x00001B59	ERROR_CTX_WINSTATION_NAME_INVALID	Der angegebene Sitzungsname ist ungültig.
7002	0x00001B5A	ERROR_CTX_INVALID_PD	Der angegebene Protokolltreiber ist ungültig.
7003	0x00001B5B	ERROR_CTX_PD_NOT_FOUND	Der angegebene Protokolltreiber wurde im Systempfad nicht gefunden.
7004	0x00001B5C	ERROR_CTX_WD_NOT_FOUND	Der angegebene Anschluss-Treiber wurde im Systempfad nicht gefunden.
7005	0x00001B5D	ERROR_CTX_CANNOT_MAKE_EVENTLOG_ENTRY	Ein Registrierungsschlüssel für die Ereignisprotokollierung konnte nicht für diese Sitzung erstellt werden.
7006	0x00001B5E	ERROR_CTX_SERVICE_NAME_COLLISION	Ein Dienst mit dem gleichen Namen ist bereits auf dem System vorhanden.
7007	0x00001B5F	ERROR_CTX_CLOSE_PENDING	Ein Schließvorgang steht für die Sitzung.
7008	0x00001B60	ERROR_CTX_NO_OUTBUF	Es sind keine freien Ausgabepuffer verfügbar.
7009	0x00001B61	ERROR_CTX_MODEM_INF_NOT_FOUND	Das MODEM.INF-Datei wurde nicht gefunden.
7010	0x00001B62	ERROR_CTX_INVALID_MODEMNAME	Der Modemname wurde in MODEM.INF nicht gefunden.
7011	0x00001B63	ERROR_CTX_MODEM_RESPONSE_ERROR	Das Modem hat den Befehl, die ihm nicht angenommen. Stellen Sie sicher, dass der Modemname des konfigurierten angeschlossenen Modems entspricht.
7012	0x00001B64	ERROR_CTX_MODEM_RESPONSE_TIMEOUT	Das Modem antwortet nicht auf den Befehl gesendet. Überprüfen Sie, ob das Modem richtig angeschlossen und eingeschaltet.
7013	0x00001B65	ERROR_CTX_MODEM_RESPONSE_NO_CARRIER	Erkennung des Trägersignals ist fehlgeschlagen oder Carrier wurde gelöscht wegen um zu trennen.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
7014	0x00001B66	ERROR_CTX_MODEM_RESPONSE_NO_DIALTONE	Freizeichen nicht innerhalb der erforderlichen Zeit erkannt. Stellen Sie sicher, dass das Telefonkabel richtig angeschlossen und funktionsfähig ist.
7015	0x00001B67	ERROR_CTX_MODEM_RESPONSE_BUSY	Busy Signal zur remote-Site auf Rückruf erkannt.
7016	0x00001B68	ERROR_CTX_MODEM_RESPONSE_VOICE	An remote-Standort beim Rückruf wurde ein Sprachsignal.
7017	0x00001B69	ERROR_CTX_TD_ERROR	Transport-Treiber-Fehler
7022	0x00001B6E	ERROR_CTX_WINSTATION_NOT_FOUND	Die angegebene Sitzung wurde nicht gefunden.
7023	0x00001B6F	ERROR_CTX_WINSTATION_ALREADY_EXISTS	Der angegebene Sitzungsname wird bereits verwendet.
7024	0x00001B70	ERROR_CTX_WINSTATION_BUSY	Der angeforderte Vorgang kann nicht werden abgeschlossen, da die Terminalverbindung momentan Verarbeitung einer verbinden, trennen, zurücksetzen oder Löschvorgang.
7025	0x00001B71	ERROR_CTX_BAD_VIDEO_MODE	Ein Versuch wurde unternommen, zum Herstellen einer Verbindung mit einer Sitzungs, deren Videomodus vom aktuellen Client nicht unterstützt wird.
7035	0x00001B7B	ERROR_CTX_GRAPHICS_INVALID	Die Anwendung hat versucht, die DOS-Grafikmodus zu aktivieren. DOS-Grafikmodus wird nicht unterstützt.
7037	0x00001B7D	ERROR_CTX_LOGON_DISABLED	Ihr Privileg der interaktiven Anmeldung wurde deaktiviert. Bitte kontaktieren Sie Ihren Administrator.
7038	0x00001B7E	ERROR_CTX_NOT_CONSOLE	Der angeforderte Vorgang kann nur an der Systemkonsole erfolgen. Dies ist meist das Ergebnis von einem Treiber oder System DLL erfordert direkten Konsolenzugriff.
7040	0x00001B80	ERROR_CTX_CLIENT_QUERY_TIMEOUT	Der Client an den Server reagiert herstellen Nachricht.
7041	0x00001B81	ERROR_CTX_CONSOLE_DISCONNECT	Trennen der Konsolensitzung wird nicht unterstützt.
7042	0x00001B82	ERROR_CTX_CONSOLE_CONNECT	Wiederherstellen einer getrennten Sitzungs auf der Konsole wird nicht unterstützt.
7044	0x00001B84	ERROR_CTX_SHADOW_DENIED	Die Anforderung zur Remoteüberwachung einer anderen Sitzung wurde verweigert.
7045	0x00001B85	ERROR_CTX_WINSTATION_ACCESS_DENIED	Der angeforderte Sitzungszugriff verweigert.
7049	0x00001B89	ERROR_CTX_INVALID_WD	Der angegebene Anschluss Treiber ist ungültig.
7050	0x00001B8A	ERROR_CTX_SHADOW_INVALID	Die angeforderte Sitzung kann nicht remote gesteuert werden. Dies kann sein, weil die Sitzung getrennt wird oder verfügt derzeit nicht über ein Benutzer angemeldet.
7051	0x00001B8B	ERROR_CTX_SHADOW_DISABLED	Die angeforderte Sitzung ist nicht für die Fernbedienung zulassen konfiguriert.
7052	0x00001B8C	ERROR_CTX_CLIENT_LICENSE_IN_USE	Ihre Anfrage zum Herstellen einer Verbindung mit diesem Terminalserver ist abgelehnt worden. Ihre Lizenznummer für Terminal Server-Client ist derzeit von einem anderen Benutzer verwendet wird. Rufen Sie bitte Ihren Systemadministrator, um eine eindeutige Lizenznummer zu erhalten.
7053	0x00001B8D	ERROR_CTX_CLIENT_LICENSE_NOT_SET	Ihre Anfrage zum Herstellen einer Verbindung mit diesem Terminalserver ist abgelehnt worden. Ihre Lizenznummer für Terminal Server-Client wurde nicht für diese Kopie des Terminal Server-Clients eingegeben. Bitte kontaktieren Sie Ihren Systemadministrator.
7054	0x00001B8E	ERROR_CTX_LICENSE_NOT_AVAILABLE	Das System hat seine lizenzierte Anmelde-limit erreicht. Bitte versuchen Sie es später erneut.
7055	0x00001B8F	ERROR_CTX_LICENSE_CLIENT_INVALID	Der Client, den Sie verwenden ist nicht lizenziert, um dieses System zu verwenden. Die Anmeldeanforderung wurde verweigert.
7056	0x00001B90	ERROR_CTX_LICENSE_EXPIRED	Die System-Lizenz ist abgelaufen. Die Anmeldeanforderung wurde verweigert.
7057	0x00001B91	ERROR_CTX_SHADOW_NOT_RUNNING	Fernbedienung konnte nicht abgebrochen werden, weil die angegebene Sitzung nicht gerade aus der Ferne gesteuert wird.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
8001	0x00001F41	FRS_ERR_INVALID_API_SEQUENCE	Die Datei Replikation Dienst-API wurde falsch aufgerufen.
8002	0x00001F42	FRS_ERR_STARTING_SERVICE	Der Dateireplikationsdienst kann nicht gestartet werden.
8003	0x00001F43	FRS_ERR_STOPPING_SERVICE	Der Dateireplikationsdienst kann nicht beendet werden.
8004	0x00001F44	FRS_ERR_INTERNAL_API	Die Datei Replikation Dienst-API wurde die Anforderung beendet. Das Ereignisprotokoll möglicherweise weitere Informationen.
8005	0x00001F45	FRS_ERR_INTERNAL	Der Dateireplikationsdienst beendet die Anforderung. Das Ereignisprotokoll möglicherweise weitere Informationen.
8006	0x00001F46	FRS_ERR_SERVICE_COMM	Der Dateireplikationsdienst kann nicht kontaktiert werden. Das Ereignisprotokoll möglicherweise weitere Informationen.
8007	0x00001F47	FRS_ERR_INSUFFICIENT_PRIV	Der Dateireplikationsdienst kann nicht die Anforderung zu erfüllen, weil der Benutzer nicht über ausreichende Berechtigungen verfügt. Das Ereignisprotokoll möglicherweise weitere Informationen.
8008	0x00001F48	FRS_ERR_AUTHENTICATION	Der Dateireplikationsdienst kann nicht der Anforderung erfüllen, weil authentifizierte RPC nicht verfügbar ist. Das Ereignisprotokoll möglicherweise weitere Informationen.
8009	0x00001F49	FRS_ERR_PARENT_INSUFFICIENT_PRIV	Der Dateireplikationsdienst kann nicht die Anforderung zu erfüllen, weil der Benutzer nicht über ausreichende Berechtigungen auf dem Domänencontroller verfügt. Das Ereignisprotokoll möglicherweise weitere Informationen.
8010	0x00001F4A	FRS_ERR_PARENT_AUTHENTICATION	Der Dateireplikationsdienst kann nicht der Anforderung erfüllen, weil authentifizierte RPC nicht verfügbar auf dem Domänencontroller ist. Das Ereignisprotokoll möglicherweise weitere Informationen.
8011	0x00001F4B	FRS_ERR_CHILD_TO_PARENT_COMM	Der Dateireplikationsdienst kann nicht mit der Dateireplikationsdienst auf dem Domänencontroller kommunizieren. Das Ereignisprotokoll möglicherweise weitere Informationen.
8012	0x00001F4C	FRS_ERR_PARENT_TO_CHILD_COMM	Der Dateireplikationsdienst des Domänencontrollers kann nicht mit der Dateireplikationsdienst auf diesem Computer kommunizieren. Das Ereignisprotokoll möglicherweise weitere Informationen.
8013	0x00001F4D	FRS_ERR_SYSVOL_POPULATE	Der Dateireplikationsdienst kann wegen eines internen Fehlers nicht das Systemvolumen ausfüllen. Das Ereignisprotokoll möglicherweise weitere Informationen.
8014	0x00001F4E	FRS_ERR_SYSVOL_POPULATE_TIMEOUT	Der Dateireplikationsdienst kann wegen einer internen Zeitüberschreitung nicht das Systemvolumen ausfüllen. Das Ereignisprotokoll möglicherweise weitere Informationen.
8015	0x00001F4F	FRS_ERR_SYSVOL_IS_BUSY	Der Dateireplikationsdienst kann die Anforderung nicht verarbeiten. Das Systemvolumen ist mit einer vorherigen Anforderung ausgelastet.
8016	0x00001F50	FRS_ERR_SYSVOL_DEMOTE	Der Dateireplikationsdienst kann nicht beendet die Replikation von Systemdatenträger wegen eines internen Fehlers. Das Ereignisprotokoll möglicherweise weitere Informationen.
8017	0x00001F51	FRS_ERR_INVALID_SERVICE_PARAMETER	Der Dateireplikationsdienst hat einen ungültigen Parameter festgestellt.
8200	0x00002008	ERROR_DS_NOT_INSTALLED	Fehler beim Installieren des Verzeichnisdiensts. Weitere Informationen finden Sie im Ereignisprotokoll.
8201	0x00002009	ERROR_DS_MEMBERSHIP_EVALUATED_LOCALY	Der Verzeichnisdienst ausgewertet Gruppenmitgliedschaften lokal.
8202	0x0000200A	ERROR_DS_NO_ATTRIBUTE_OR_VALUE	Die angegebene Verzeichnisdienstattribut oder der Wert ist nicht vorhanden.



Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
8203	0x0000200B	ERROR_DS_INVALID_ATTRIBUTE_SYNTAX	Die Attributsyntax, die an den Verzeichnisdienst ist ungültig.
8204	0x0000200C	ERROR_DS_ATTRIBUTE_TYPE_UNDEFINED	Der Typ des Attributs angegeben, um den Verzeichnisdienst ist nicht definiert.
8205	0x0000200D	ERROR_DS_ATTRIBUTE_OR_VALUE_EXISTS	Das angegebene Verzeichnisdienstattribut oder Wert ist bereits vorhanden.
8206	0x0000200E	ERROR_DS_BUSY	Der Verzeichnisdienst ist ausgelastet.
8207	0x0000200F	ERROR_DS_UNAVAILABLE	Der Verzeichnisdienst ist nicht verfügbar.
8208	0x00002010	ERROR_DS_NO_RIDS_ALLOCATED	Der Verzeichnisdienst konnte einen relativen Bezeichner zuweisen.
8209	0x00002011	ERROR_DS_NO_MORE_RIDS	Der Verzeichnisdienst hat den Pool der relative IDs erschöpft.
8210	0x00002012	ERROR_DS_INCORRECT_ROLE_OWNER	Der angeforderte Vorgang konnte nicht ausgeführt werden, da der Verzeichnisdienst nicht den Master für diese Art von Vorgang ist.
8211	0x00002013	ERROR_DS_RIDMGR_INIT_ERROR	Der Verzeichnisdienst konnte das Teilsystem, das relative Identifikatoren zuweist, zu initialisieren.
8212	0x00002014	ERROR_DS_OBJ_CLASS_VIOLATION	Der angeforderte Vorgang hat eine oder mehrere Einschränkungen mit der Klasse des Objekts nicht erfüllt.
8213	0x00002015	ERROR_DS_CANT_ON_NON_LEAF	Der Verzeichnisdienst kann den angeforderten Vorgang nur an einem Endknotenobjekt durchführen.
8214	0x00002016	ERROR_DS_CANT_ON_RDN	Der Verzeichnisdienst kann nicht den angeforderten Vorgang an das RDN-Attribut eines Objekts durchführen.
8215	0x00002017	ERROR_DS_CANT_MOD_OBJ_CLASS	Der Verzeichnisdienst erkennt, einen Versuch, die Objektklasse eines Objekts zu ändern.
8216	0x00002018	ERROR_DS_CROSS_DOM_MOVE_ERROR	Der angeforderte domänenübergreifende Verschiebevorgang konnte nicht ausgeführt werden.
8217	0x00002019	ERROR_DS_GC_NOT_AVAILABLE	Nicht in der Lage, den globalen Katalogserver zu kontaktieren.
8218	0x0000201A	ERROR_SHARED_POLICY	Das Richtlinienobjekt ist freigegeben und kann nur im Stammverzeichnis geändert werden.
8219	0x0000201B	ERROR_POLICY_OBJECT_NOT_FOUND	Das Richtlinienobjekt ist nicht vorhanden.
8220	0x0000201C	ERROR_POLICY_ONLY_IN_DS	Die angeforderten Richtlinieninformationen ist nur im Verzeichnisdienst.
8221	0x0000201D	ERROR_PROMOTION_ACTIVE	Eine Heraufstufung von Domänencontrollern ist derzeit aktiv.
8222	0x0000201E	ERROR_NO_PROMOTION_ACTIVE	Eine Heraufstufung von Domänencontrollern ist derzeit nicht aktiv
8224	0x00002020	ERROR_DS_OPERATIONS_ERROR	Operationen Fehler.
8225	0x00002021	ERROR_DS_PROTOCOL_ERROR	Ein Protokollfehler ist aufgetreten.
8226	0x00002022	ERROR_DS_TIMELIMIT_EXCEEDED	Das Zeitlimit für diese Anforderung wurde überschritten.
8227	0x00002023	ERROR_DS_SIZELIMIT_EXCEEDED	Das Größenlimit für diese Anforderung wurde überschritten.
8228	0x00002024	ERROR_DS_ADMIN_LIMIT_EXCEEDED	Die administrative Grenze für diese Anforderung wurde überschritten.
8229	0x00002025	ERROR_DS_COMPARE_FALSE	Die Vergleich-Antwort war falsch.
8230	0x00002026	ERROR_DS_COMPARE_TRUE	Die Vergleich-Antwort galt.
8231	0x00002027	ERROR_DS_AUTH_METHOD_NOT_SUPPORTED	Die angeforderte Authentifizierungsmethode wird vom Server nicht unterstützt.
8232	0x00002028	ERROR_DS_STRONG_AUTH_REQUIRED	Eine sicherere Authentifizierungsmethode wird für diesen Server benötigt.
8233	0x00002029	ERROR_DS_INAPPROPRIATE_AUTH	Unangemessene Authentifizierung.
8234	0x0000202A	ERROR_DS_AUTH_UNKNOWN	Der Authentifizierungsmechanismus ist unbekannt.
8235	0x0000202B	ERROR_DS_REFERRAL	Ein Verweis wurde vom Server zurückgegeben.
8236	0x0000202C	ERROR_DS_UNAVAILABLE_CRIT_EXTENSION	Der Server unterstützt nicht die angeforderte kritische Erweiterung.
8237	0x0000202D	ERROR_DS_CONFIDENTIALITY_REQUIRED	Diese Anforderung erfordert eine sichere Verbindung.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
8238	0x0000202E	ERROR_DS_INAPPROPRIATE_MATCHING	Unangemessene Abgleich verwendet werden.
8239	0x0000202F	ERROR_DS_CONSTRAINT_VIOLATION	Eine Beschränkungsverletzung aufgetreten.
8240	0x00002030	ERROR_DS_NO_SUCH_OBJECT	Es gibt kein solches Objekt auf dem Server.
8241	0x00002031	ERROR_DS_ALIAS_PROBLEM	Es gibt eine Alias-Problem.
8242	0x00002032	ERROR_DS_INVALID_DN_SYNTAX	Eine ungültige dn-Syntax es wurde angegeben.
8243	0x00002033	ERROR_DS_IS_LEAF	Das Objekt ist ein Leaf-Objekt.
8244	0x00002034	ERROR_DS_ALIAS_DEREF_PROBLEM	Es ist ein Alias Dereferenzierung Problem.
8245	0x00002035	ERROR_DS_UNWILLING_TO_PERFORM	Der Server ist nicht bereit, die Anforderung zu verarbeiten.
8246	0x00002036	ERROR_DS_LOOP_DETECT	Eine Schleife wurde gefunden.
8247	0x00002037	ERROR_DS_NAMING_VIOLATION	Es ist eine Benennung Verletzung.
8248	0x00002038	ERROR_DS_OBJECT_RESULTS_TOO_LARGE	Die Ergebnismenge ist zu groß.
8249	0x00002039	ERROR_DS_AFFECTS_MULTIPLE_DSAS	Der Vorgang beeinflusst mehrere DSAs
8250	0x0000203A	ERROR_DS_SERVER_DOWN	Der Server ist nicht funktionstüchtig.
8251	0x0000203B	ERROR_DS_LOCAL_ERROR	Ein lokaler Fehler ist aufgetreten.
8252	0x0000203C	ERROR_DS_ENCODING_ERROR	Ein Codierung Fehler aufgetreten.
8253	0x0000203D	ERROR_DS_DECODING_ERROR	Ein Decodierungsfehler ist aufgetreten.
8254	0x0000203E	ERROR_DS_FILTER_UNKNOWN	Im Such-Filter wird nicht erkannt.
8255	0x0000203F	ERROR_DS_PARAM_ERROR	Ein oder mehrere Parameter sind ungültig.
8256	0x00002040	ERROR_DS_NOT_SUPPORTED	Die angegebene Methode wird nicht unterstützt.
8257	0x00002041	ERROR_DS_NO_RESULTS_RETURNED	Keine Ergebnisse wurden zurückgegeben.
8258	0x00002042	ERROR_DS_CONTROL_NOT_FOUND	Das angegebene Steuerelement wird vom Server nicht unterstützt.
8259	0x00002043	ERROR_DS_CLIENT_LOOP	Eine Weiterleitungsschleife wurde vom Client erkannt.
8260	0x00002044	ERROR_DS_REFERRAL_LIMIT_EXCEEDED	Die voreingestellten Empfehlung wurde überschritten.
8261	0x00002045	ERROR_DS_SORT_CONTROL_MISSING	Die Suche erfordert eine SORT-Steuerung.
8262	0x00002046	ERROR_DS_OFFSET_RANGE_ERROR	Die Suchergebnisse überschreiten den angegebenen Offsetbereich.
8301	0x0000206D	ERROR_DS_ROOT_MUST_BE_NC	Das Stammobjekt muss der Kopf eines Namenskontextes sein. Das Stammobjekt kann kein instanziiertes übergeordnetes Element haben.
8302	0x0000206E	ERROR_DS_ADD_REPLICA_INHIBITED	Add-Replikat-Operation kann nicht ausgeführt werden. Der Namenskontext muss um das Replikat erstellen beschreibbar sein.
8303	0x0000206F	ERROR_DS_ATT_NOT_DEF_IN_SCHEMA	Ein Verweis auf ein Attribut, das nicht in das Schema definiert ist aufgetreten.
8304	0x00002070	ERROR_DS_MAX_OBJ_SIZE_EXCEEDED	Die maximale Größe eines Objekts wurde überschritten.
8305	0x00002071	ERROR_DS_OBJ_STRING_NAME_EXISTS	Es wurde versucht, ein Objekt in das Verzeichnis mit einem Namen hinzuzufügen, die bereits verwendet wird.
8306	0x00002072	ERROR_DS_NO_RDN_DEFINED_IN_SCHEMA	Wurde versucht, ein Objekt einer Klasse hinzuzufügen, die keine RDN im Schema definiert.
8307	0x00002073	ERROR_DS_RDN_DOESNT_MATCH_SCHEMA	Wurde versucht, fügen Sie ein Objekt mithilfe eines RDN, die nicht der RDN im Schema definiert ist.
8308	0x00002074	ERROR_DS_NO_REQUESTED_ATTRS_FOUND	Keines der angeforderten Attribute wurden auf die Objekte gefunden.
8309	0x00002075	ERROR_DS_USER_BUFFER_TOO_SMALL	Der Benutzerpuffer ist zu klein.
8310	0x00002076	ERROR_DS_ATT_IS_NOT_ON_OBJ	Das in den Vorgang angegebene Attribut ist nicht vorhanden, auf das Objekt.
8311	0x00002077	ERROR_DS_ILLEGAL_MOD_OPERATION	Illegale Änderungsvorgang. Einige Aspekte der Änderung ist nicht zulässig.
8312	0x00002078	ERROR_DS_OBJ_TOO_LARGE	Das angegebene Objekt ist zu groß.
8313	0x00002079	ERROR_DS_BAD_INSTANCE_TYPE	Der angegebene Instanzentyp ist nicht gültig.
8314	0x0000207A	ERROR_DS_MASTERDSA_REQUIRED	Der Vorgang muss an einer master-DSA ausgeführt werden.
8315	0x0000207B	ERROR_DS_OBJECT_CLASS_REQUIRED	Das Objekt-Class-Attribut muss angegeben werden.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
8316	0x0000207C	ERROR_DS_MISSING_REQUIRED_ATT	Ein erforderliches Attribut fehlt.
8317	0x0000207D	ERROR_DS_ATT_NOT_DEF_FOR_CLASS	Es wurde versucht, ein Objekt, um ein Attribut zu enthalten, die nicht legal für seine Klasse ist zu ändern
8318	0x0000207E	ERROR_DS_ATT_ALREADY_EXISTS	Das angegebene Attribut ist bereits für das Objekt.
8320	0x00002080	ERROR_DS_CANT_ADD_ATT_VALUES	Das angegebene Attribut ist nicht vorhanden oder hat keine Werte.
8321	0x00002081	ERROR_DS_SINGLE_VALUE_CONSTRAINT	Für ein Attribut, die nur einen Wert haben kann, wurden mehrere Werte angegeben.
8322	0x00002082	ERROR_DS_RANGE_CONSTRAINT	Ein Wert für das Attribut wurde nicht in den zulässigen Wertebereich.
8323	0x00002083	ERROR_DS_ATT_VAL_ALREADY_EXISTS	Der angegebene Wert ist bereits vorhanden.
8324	0x00002084	ERROR_DS_CANT_REM_MISSING_ATT	Das Attribut kann nicht entfernt werden, da sie nicht auf das Objekt vorhanden ist.
8325	0x00002085	ERROR_DS_CANT_REM_MISSING_ATT_VAL	Der Attributwert kann nicht entfernt werden, da sie nicht auf das Objekt vorhanden ist.
8326	0x00002086	ERROR_DS_ROOT_CANT_BE_SUBREF	Das angegebene Stammobjekt kann nicht Objekt einer Subref sein.
8327	0x00002087	ERROR_DS_NO_CHAINING	Verkettung ist nicht gestattet.
8328	0x00002088	ERROR_DS_NO_CHAINED_EVAL	Verkettete Bewertung ist nicht gestattet.
8329	0x00002089	ERROR_DS_NO_PARENT_OBJECT	Der Vorgang konnte nicht ausgeführt werden, weil das Objekt übergeordneten instanziiert oder gelöscht wird.
8330	0x0000208A	ERROR_DS_PARENT_IS_AN_ALIAS	Dass ein Elternteil, das ein Alias ist, ist nicht gestattet. Aliase sind Endobjekte.
8331	0x0000208B	ERROR_DS_CANT_MIX_MASTER_AND_REPS	Das Objekt und das übergeordnete müssen denselben Typ haben, entweder beide Master oder beide Replikat sein.
8332	0x0000208C	ERROR_DS_CHILDREN_EXIST	Der Vorgang kann nicht ausgeführt werden, da untergeordnete Objekte vorhanden sind. Dieser Vorgang kann nur auf einem Blatt-Objekt ausgeführt werden.
8333	0x0000208D	ERROR_DS_OBJ_NOT_FOUND	Verzeichnis-Objekt nicht gefunden.
8334	0x0000208E	ERROR_DS_ALIASED_OBJ_MISSING	Das Alias-Objekt fehlt.
8335	0x0000208F	ERROR_DS_BAD_NAME_SYNTAX	Die zu verwendenden Objektname ist Syntax nicht.
8336	0x00002090	ERROR_DS_ALIAS_POINTS_TO_ALIAS	Es ist nicht gestattet, für einen Alias auf einen anderen Alias verweisen.
8337	0x00002091	ERROR_DS_CANT_DEREF_ALIAS	Der Alias kann nicht dereferenziert werden.
8338	0x00002092	ERROR_DS_OUT_OF_SCOPE	Die Bedienung ist außerhalb des gültigen Bereichs.
8339	0x00002093	ERROR_DS_OBJECT_BEING_REMOVED	Der Vorgang kann nicht fortgesetzt werden, da das Objekt wird gerade entfernt wird.
8340	0x00002094	ERROR_DS_CANT_DELETE_DSA_OBJ	Das DSA-Objekt kann nicht gelöscht werden.
8341	0x00002095	ERROR_DS_GENERIC_ERROR	Ein Verzeichnisdienstfehler ist aufgetreten.
8342	0x00002096	ERROR_DS_DSA_MUST_BE_INT_MASTER	Der Vorgang kann nur auf einem internen master-DSA-Objekt ausgeführt werden.
8343	0x00002097	ERROR_DS_CLASS_NOT_DSA	Das Objekt muss der Klasse DSA sein.
8344	0x00002098	ERROR_DS_INSUFF_ACCESS_RIGHTS	Unzureichende Berechtigungen zum Ausführen des Vorgangs.
8345	0x00002099	ERROR_DS_ILLEGAL_SUPERIOR	Das Objekt kann nicht hinzugefügt werden, da das übergeordnete Element nicht auf der Liste der möglichen Vorgesetzten ist.
8346	0x0000209A	ERROR_DS_ATTRIBUTE_OWNED_BY_SAM	Zugriff auf das Attribut ist unzulässig, da das Attribut von Security Accounts Manager (SAM) gehört.
8347	0x0000209B	ERROR_DS_NAME_TOO_MANY_PARTS	Der Name hat zu viele Teile.
8348	0x0000209C	ERROR_DS_NAME_TOO_LONG	Der Name ist zu lang.
8349	0x0000209D	ERROR_DS_NAME_VALUE_TOO_LONG	Der Namenswert ist zu lang.
8350	0x0000209E	ERROR_DS_NAME_UNPARSEABLE	Der Verzeichnisdienst-Fehler analysieren einen Namen.
8351	0x0000209F	ERROR_DS_NAME_TYPE_UNKNOWN	Der Verzeichnisdienst kann nicht den Attributtyp für einen Namen erhalten.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
8352	0x000020A0	ERROR_DS_NOT_AN_OBJECT	Der Name identifiziert kein Objekt; der Name bezeichnet ein Phantom.
8353	0x000020A1	ERROR_DS_SEC_DESC_TOO_SHORT	Die Sicherheitsbeschreibung ist zu kurz.
8354	0x000020A2	ERROR_DS_SEC_DESC_INVALID	Die Sicherheitsbeschreibung ist ungültig.
8355	0x000020A3	ERROR_DS_NO_DELETED_NAME	Fehler beim Namen für gelöschte Objekt zu erstellen.
8356	0x000020A4	ERROR_DS_SUBREF_MUST_HAVE_PARENT	Das übergeordnete Objekt einer neuen Subref muss vorhanden sein.
8357	0x000020A5	ERROR_DS_NCNAME_MUST_BE_NC	Das Objekt muss ein Namenskontext sein.
8358	0x000020A6	ERROR_DS_CANT_ADD_SYSTEM_ONLY	Es ist nicht erlaubt, ein Attribut hinzuzufügen, die das System gehört.
8359	0x000020A7	ERROR_DS_CLASS_MUST_BE_CONCRETE	Die Klasse des Objekts muss strukturelle sein; eine abstrakte Klasse kann nicht instanziiert werden.
8360	0x000020A8	ERROR_DS_INVALID_DMD	Das Schema-Objekt konnte nicht gefunden werden.
8361	0x000020A9	ERROR_DS_OBJ_GUID_EXISTS	Ein lokales Objekt mit dieser GUID (tot oder lebendig) ist bereits vorhanden.
8362	0x000020AA	ERROR_DS_NOT_ON_BACKLINK	Der Vorgang kann nicht auf einen Backlink ausgeführt werden.
8363	0x000020AB	ERROR_DS_NO_CROSSREF_FOR_NC	Der Querverweis für den angegebenen Namenskontext konnte nicht gefunden werden.
8364	0x000020AC	ERROR_DS_SHUTTING_DOWN	Der Vorgang konnte nicht ausgeführt werden, da der Verzeichnisdienst heruntergefahren wird.
8365	0x000020AD	ERROR_DS_UNKNOWN_OPERATION	Die Verzeichnis-Service-Anforderung ist ungültig.
8366	0x000020AE	ERROR_DS_INVALID_ROLE_OWNER	Das Attribut Role Besitzer konnte nicht gelesen werden.
8367	0x000020AF	ERROR_DS_COULDNT_CONTACT_FSMO	Der angeforderte FSMO-Vorgang ist fehlgeschlagen. Der aktuelle FSMO-Halter konnten nicht erreicht werden.
8368	0x000020B0	ERROR_DS_CROSS_NC_DN_RENAME	Änderung der DN über einen Namenskontext ist nicht gestattet.
8369	0x000020B1	ERROR_DS_CANT_MOD_SYSTEM_ONLY	Das Attribut kann nicht geändert werden, da es dem System gehört.
8370	0x000020B2	ERROR_DS_REPLICATOR_ONLY	Nur der Replikator kann diese Funktion auszuführen.
8371	0x000020B3	ERROR_DS_OBJ_CLASS_NOT_DEFINED	Die angegebene Klasse ist nicht definiert.
8372	0x000020B4	ERROR_DS_OBJ_CLASS_NOT_SUBCLASS	Die angegebene Klasse ist keine Unterklasse.
8373	0x000020B5	ERROR_DS_NAME_REFERENCE_INVALID	Der Namensbezug ist ungültig.
8374	0x000020B6	ERROR_DS_CROSS_REF_EXISTS	Ein Querverweis ist bereits vorhanden.
8375	0x000020B7	ERROR_DS_CANT_DEL_MASTER_CROSSREF	Es ist nicht gestattet, ein master Querverweis zu löschen.
8376	0x000020B8	ERROR_DS_SUBTREE_NOTIFY_NOT_NC_HEAD	Unterstruktur Benachrichtigungen werden nur von NC-Köpfen unterstützt.
8377	0x000020B9	ERROR_DS_NOTIFY_FILTER_TOO_COMPLEX	Benachrichtigungsfilter ist zu komplex.
8378	0x000020BA	ERROR_DS_DUP_RDN	Schemaaktualisierung ist fehlgeschlagen: RDN zu duplizieren.
8379	0x000020BB	ERROR_DS_DUP_OID	Schemaaktualisierung ist fehlgeschlagen: OID duplizieren
8380	0x000020BC	ERROR_DS_DUP_MAPI_ID	Schemaaktualisierung ist fehlgeschlagen: MAPI-ID doppelt.
8381	0x000020BD	ERROR_DS_DUP_SCHEMA_ID_GUID	Schemaaktualisierung ist fehlgeschlagen: doppelte Schema-Id-GUID.
8382	0x000020BE	ERROR_DS_DUP_LDAP_DISPLAY_NAME	Schemaaktualisierung ist fehlgeschlagen: doppelte LDAP-Anzeigenname.
8383	0x000020BF	ERROR_DS_SEMANTIC_ATT_TEST	Schemaaktualisierung ist fehlgeschlagen: Spektrum-untere weniger als obere Bereich
8384	0x000020C0	ERROR_DS_SYNTAX_MISMATCH	Schemaaktualisierung ist fehlgeschlagen: Syntax-Konflikt
8385	0x000020C1	ERROR_DS_EXISTS_IN_MUST_HAVE	Löschen des Schemas ist fehlgeschlagen: Attribut wird verwendet in muss enthalten
8386	0x000020C2	ERROR_DS_EXISTS_IN_MAY_HAVE	Löschen des Schemas ist fehlgeschlagen: Attribut wird verwendet, im Mai-enhalten

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
8387	0x000020C3	ERROR_DS_NONEXISTENT_MAY_HAVE	Schemaaktualisierung ist fehlgeschlagen: Attribut in Mai-enthalten ist nicht vorhanden
8388	0x000020C4	ERROR_DS_NONEXISTENT_MUST_HAVE	Schemaaktualisierung ist fehlgeschlagen: Attribut in muss enthalten-existiert nicht
8389	0x000020C5	ERROR_DS_AUX_CLS_TEST_FAIL	Schemaaktualisierung ist fehlgeschlagen: Klasse in der Aux-Klasse Liste existiert nicht oder ist keine Erweiterungsklasse
8390	0x000020C6	ERROR_DS_NONEXISTENT_POSS_SUP	Schemaaktualisierung ist fehlgeschlagen: Klasse in Poss-Vorgesetzten ist nicht vorhanden
8391	0x000020C7	ERROR_DS_SUB_CLS_TEST_FAIL	Schemaaktualisierung ist fehlgeschlagen: Klasse in der Subclassof-Liste ist nicht vorhanden oder erfüllt nicht Hierarchieregeln
8392	0x000020C8	ERROR_DS_BAD_RDN_ATT_ID_SYNTAX	Schemaaktualisierung ist fehlgeschlagen: Rdn-Att-Id hat falsche Syntax
8393	0x000020C9	ERROR_DS_EXISTS_IN_AUX_CLS	Löschen des Schemas ist fehlgeschlagen: Klasse wird als Erweiterungsklasse verwendet
8394	0x000020CA	ERROR_DS_EXISTS_IN_SUB_CLS	Löschen des Schemas ist fehlgeschlagen: Klasse wird als Sub-Klasse verwendet
8395	0x000020CB	ERROR_DS_EXISTS_IN_POSS_SUP	Löschen des Schemas ist fehlgeschlagen: Klasse wird verwendet, als Poss-Superior
8396	0x000020CC	ERROR_DS_RECALCSHEMA_FAILED	Schemaaktualisierung ist fehlgeschlagen in Validierung Cache eine Neuberechnung.
8397	0x000020CD	ERROR_DS_TREE_DELETE_NOT_FINISHED	Der Strukturlöschung ist noch nicht fertig.
8398	0x000020CE	ERROR_DS_CANT_DELETE	Der angeforderte Löschvorgang konnte nicht ausgeführt werden.
8399	0x000020CF	ERROR_DS_ATT_SCHEMA_REQ_ID	Kann nicht gelesen werden die Klassen-ID für das Schema-Protokoll regelt.
8400	0x000020D0	ERROR_DS_BAD_ATT_SCHEMA_SYNTAX	Das Attributschema ist Syntax nicht.
8401	0x000020D1	ERROR_DS_CANT_CACHE_ATT	Das Attribut konnte nicht zwischengespeichert werden.
8402	0x000020D2	ERROR_DS_CANT_CACHE_CLASS	Die Klasse konnte nicht zwischengespeichert werden.
8403	0x000020D3	ERROR_DS_CANT_REMOVE_ATT_CACHE	Das Attribut konnte nicht aus dem Cache entfernt werden.
8404	0x000020D4	ERROR_DS_CANT_REMOVE_CLASS_CACHE	Die Klasse konnte nicht aus dem Cache entfernt werden.
8405	0x000020D5	ERROR_DS_CANT_RETRIEVE_DN	Das distinguished Name-Attribut konnte nicht gelesen werden.
8406	0x000020D6	ERROR_DS_MISSING_SUPREF	Objekt eine erforderliche Subref fehlt.
8407	0x000020D7	ERROR_DS_CANT_RETRIEVE_INSTANCE	Das Type-Instanzattribut konnte nicht abgerufen werden.
8408	0x000020D8	ERROR_DS_CODE_INCONSISTENCY	Ein interner Fehler aufgetreten.
8409	0x000020D9	ERROR_DS_DATABASE_ERROR	Ein Datenbankfehler ist aufgetreten.
8410	0x000020DA	ERROR_DS_GOVERNSID_MISSING	Das Attribut GOVERNSID fehlt.
8411	0x000020DB	ERROR_DS_MISSING_EXPECTED_ATT	Eine erwartete Attribut fehlt.
8412	0x000020DC	ERROR_DS_NCNAME_MISSING_CR_REF	Im angegebene Namenskontext fehlt ein Querverweis.
8413	0x000020DD	ERROR_DS_SECURITY_CHECKING_ERROR	Eine Sicherheitsüberprüfung-Fehler ist aufgetreten.
8414	0x000020DE	ERROR_DS_SCHEMA_NOT_LOADED	Das Schema ist nicht geladen.
8415	0x000020DF	ERROR_DS_SCHEMA_ALLOC_FAILED	Schema Zuweisung ist fehlgeschlagen. Bitte prüfen Sie, ob die Maschine Arbeitsspeicher knapp ist.
8416	0x000020E0	ERROR_DS_ATT_SCHEMA_REQ_SYNTAX	Erhalten Sie die erforderliche Syntax für das Attributschema konnte nicht.
8417	0x000020E1	ERROR_DS_GCVERIFY_ERROR	Die Überprüfung des globalen Katalogs ist fehlgeschlagen. Der globale Katalog ist nicht verfügbar oder unterstützt nicht den Betrieb. Ein Teil des Verzeichnisses ist zurzeit nicht verfügbar.
8418	0x000020E2	ERROR_DS_DRA_SCHEMA_MISMATCH	Der Replikationsvorgang ist aufgrund einer Schemaübereinstimmung zwischen den beteiligten Servern fehlgeschlagen.
8419	0x000020E3	ERROR_DS_CANT_FIND_DSA_OBJ	Das DSA-Objekt konnte nicht gefunden werden.
8420	0x000020E4	ERROR_DS_CANT_FIND_EXPECTED_NC	Der Namenskontext konnte nicht gefunden werden.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
8421	0x000020E5	ERROR_DS_CANT_FIND_NC_IN_CACHE	Der Namenskontext konnte nicht im Cache gefunden werden.
8422	0x000020E6	ERROR_DS_CANT_RETRIEVE_CHILD	Das untergeordnete Objekt konnte nicht abgerufen werden.
8423	0x000020E7	ERROR_DS_SECURITY_ILLEGAL_MODIFY	Die Änderung war aus Sicherheitsgründen nicht gestattet.
8424	0x000020E8	ERROR_DS_CANT_REPLACE_HIDDEN_REC	Der Vorgang kann nicht den versteckten Eintrag ersetzen.
8425	0x000020E9	ERROR_DS_BAD_HIERARCHY_FILE	Die Hierarchie-Datei ist ungültig.
8426	0x000020EA	ERROR_DS_BUILD_HIERARCHY_TABLE_FAILED	Der Versuch, die Hierarchietabelle zu erstellen gescheitert.
8427	0x000020EB	ERROR_DS_CONFIG_PARAM_MISSING	Der Verzeichnis-Konfigurations-Parameter fehlt in der Registrierung.
8428	0x000020EC	ERROR_DS_COUNTING_AB_INDICES_FAILED	Der Versuch, die Adresse Buch Indizes zählen konnte.
8429	0x000020ED	ERROR_DS_HIERARCHY_TABLE_MALLOC_FAILED	Die Reservierung der Hierarchietabelle ist fehlgeschlagen.
8430	0x000020EE	ERROR_DS_INTERNAL_FAILURE	Der Verzeichnisdienst ist einen internen Fehler aufgetreten.
8431	0x000020EF	ERROR_DS_UNKNOWN_ERROR	Der Verzeichnisdienst ist einen unbekanntem Fehler aufgetreten.
8432	0x000020F0	ERROR_DS_ROOT_REQUIRES_CLASS_TOP	Ein Stammobjekt erfordert eine Klasse der 'Top'.
8433	0x000020F1	ERROR_DS_REFUSING_FSMO_ROLES	Dieser Verzeichnisserver wird heruntergefahren und kann nicht übernehmen des Besitzes von neuen schwimmenden Einzelmaster-Operation-Rollen.
8434	0x000020F2	ERROR_DS_MISSING_FSMO_SETTINGS	Der Verzeichnisdienst fehlen verbindliche Konfigurationsinformationen, und ist nicht in der Lage, die Besitzer von schwimmenden Einzelmaster-Operation-Rollen zu bestimmen.
8435	0x000020F3	ERROR_DS_UNABLE_TO_SURRENDER_ROLES	Der Verzeichnisdienst konnte den Besitz von einem oder mehreren schwimmenden Einzelmaster-Operation Rollen an andere Server übertragen.
8436	0x000020F4	ERROR_DS_DRA_GENERIC	Der Replikationsvorgang ist fehlgeschlagen.
8437	0x000020F5	ERROR_DS_DRA_INVALID_PARAMETER	Ein ungültiger Parameter wurde für diesen Replikationsvorgang angegeben.
8438	0x000020F6	ERROR_DS_DRA_BUSY	Der Verzeichnisdienst ist zu beschäftigt, um den Replikationsvorgang zu diesem Zeitpunkt abgeschlossen.
8439	0x000020F7	ERROR_DS_DRA_BAD_DN	Der distinguished Name für diesen Replikationsvorgang angegebene ist ungültig.
8440	0x000020F8	ERROR_DS_DRA_BAD_NC	Der für diesen Replikationsvorgang angegebene Namenskontext ist ungültig.
8441	0x000020F9	ERROR_DS_DRA_DN_EXISTS	Der distinguished Name für diesen Replikationsvorgang bereits angegeben vorhanden ist.
8442	0x000020FA	ERROR_DS_DRA_INTERNAL_ERROR	Im Replikationssystem ist einen internen Fehler aufgetreten.
8443	0x000020FB	ERROR_DS_DRA_INCONSISTENT_DIT	Der Replikationsvorgang ist eine Datenbankinkonsistenz aufgetreten.
8444	0x000020FC	ERROR_DS_DRA_CONNECTION_FAILED	Der für diesen Replikationsvorgang angegebene Server konnte nicht kontaktiert werden.
8445	0x000020FD	ERROR_DS_DRA_BAD_INSTANCE_TYPE	Der Replikationsvorgang hat ein Objekt mit einem ungültigen Instanzentyp feststellt.
8446	0x000020FE	ERROR_DS_DRA_OUT_OF_MEM	Der Replikationsvorgang ist fehlgeschlagen, Speicher.
8447	0x000020FF	ERROR_DS_DRA_MAIL_PROBLEM	Der Replikationsvorgang wurde ein Fehler in das Mailsystem.
8448	0x00002100	ERROR_DS_DRA_REF_ALREADY_EXISTS	Die Replikationsverweisinformationen für den Zielservers bereits vorhanden ist.
8449	0x00002101	ERROR_DS_DRA_REF_NOT_FOUND	Die Replikationsverweisinformationen für den Zielservers ist nicht vorhanden.
8450	0x00002102	ERROR_DS_DRA_OBJ_IS_REP_SOURCE	Der Namenskontext kann nicht entfernt werden, da es auf einen anderen Server repliziert werden.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
8451	0x00002103	ERROR_DS_DRA_DB_ERROR	Der Replikationsvorgang ist einen Datenbankfehler aufgetreten.
8452	0x00002104	ERROR_DS_DRA_NO_REPLICA	Der Namenskontext wird gerade entfernt oder wird nicht vom angegebenen Server repliziert.
8453	0x00002105	ERROR_DS_DRA_ACCESS_DENIED	Der Replikationszugriff wurde verweigert.
8454	0x00002106	ERROR_DS_DRA_NOT_SUPPORTED	Der angeforderte Vorgang wird von dieser Version des Verzeichnisdienstes nicht unterstützt.
8455	0x00002107	ERROR_DS_DRA_RPC_CANCELLED	Der Remoteprozeduraufruf Replikation wurde abgebrochen.
8456	0x00002108	ERROR_DS_DRA_SOURCE_DISABLED	Der Quellserver weist derzeit Replikationsanforderungen.
8457	0x00002109	ERROR_DS_DRA_SINK_DISABLED	Der Zielserver ist derzeit Replikationsanforderungen ablehnen.
8458	0x0000210A	ERROR_DS_DRA_NAME_COLLISION	Der Replikationsvorgang ist aufgrund einer Kollision von Objektnamen fehlgeschlagen.
8459	0x0000210B	ERROR_DS_DRA_SOURCE_REINSTALLED	Die Replikationsquelle wurde erneut installiert wurde.
8460	0x0000210C	ERROR_DS_DRA_MISSING_PARENT	Der Replikationsvorgang ist fehlgeschlagen, da ein erforderliches übergeordnetes Objekt fehlt.
8461	0x0000210D	ERROR_DS_DRA_PREEMPTED	Der Replikationsvorgang wurde präemptiv verdrängt.
8462	0x0000210E	ERROR_DS_DRA_ABANDON_SYNC	Der Replikationssynchronisierungsversuch wurde aus Mangel an Updates aufgegeben.
8463	0x0000210F	ERROR_DS_DRA_SHUTDOWN	Der Replikationsvorgang wurde abgebrochen, da das System heruntergefahren wird.
8464	0x00002110	ERROR_DS_DRA_INCOMPATIBLE_PARTIAL_SE T	Der Replikationssynchronisierungsversuch ist fehlgeschlagen, da der Ziel-Teilattributsatz keine Teilmenge der Quelle Teilattributsatz.
8465	0x00002111	ERROR_DS_DRA_SOURCE_IS_PARTIAL_REPLI CA	Der Replikationssynchronisierungsversuch ist fehlgeschlagen, da ein Masterreplikat versucht, aus einem partiellen Replikat synchronisiert.
8466	0x00002112	ERROR_DS_DRA_EXTN_CONNECTION_FAILED	Der für diesen Replikationsvorgang angegebene Server wurde kontaktiert, aber diesen Server konnte keine Verbindung einen zusätzlichen Server benötigt, um den Vorgang abzuschließen.
8467	0x00002113	ERROR_DS_INSTALL_SCHEMA_MISMATCH	Die Version des Active Directory-Schemas der Quellgesamtstruktur ist nicht kompatibel mit der Version von Active Directory auf diesem Computer. Sie müssen das Betriebssystem auf einem Domänencontroller in der Quellgesamtstruktur aktualisieren, bevor dieser Computer als Domänencontroller in dieser Gesamtstruktur hinzugefügt werden kann.
8468	0x00002114	ERROR_DS_DUP_LINK_ID	Schemaaktualisierung ist fehlgeschlagen: Attribut mit der gleichen Verbindungskennung bereits vorhanden ist.
8469	0x00002115	ERROR_DS_NAME_ERROR_RESOLVING	Namensübersetzung: allgemeiner Verarbeitungsfehler.
8470	0x00002116	ERROR_DS_NAME_ERROR_NOT_FOUND	Namensübersetzung: konnte nicht gefunden werden, den Namen oder die unzureichende Recht, Namen zu sehen.
8471	0x00002117	ERROR_DS_NAME_ERROR_NOT_UNIQUE	Namensübersetzung: Input Name mehreren Ausgabenamen zugeordnet.
8472	0x00002118	ERROR_DS_NAME_ERROR_NO_MAPPING	Namensübersetzung: Input Name gefunden, aber nicht die zugehörigen Ausgabe-Format.
8473	0x00002119	ERROR_DS_NAME_ERROR_DOMAIN_ONLY	Namensübersetzung: nicht vollständig lösen nur die Domain wurde gefunden.
8474	0x0000211A	ERROR_DS_NAME_ERROR_NO_SYNTACTICAL MAPPING	Namensübersetzung: nicht in der Lage, rein syntaktische Zuordnung auf dem Client auszuführen, ohne auf den Draht ausgehen.
8475	0x0000211B	ERROR_DS_CONSTRUCTED_ATT_MOD	Änderung der konstruierten Att ist nicht zulässig.
8476	0x0000211C	ERROR_DS_WRONG_OM_OBJ_CLASS	Die OM-Objektklasse angegeben ist für ein Attribut mit der angegebenen Syntax falsch.
8477	0x0000211D	ERROR_DS_DRA_REPL_PENDING	Die Replikationsanforderung wurde gebucht; warten auf Antwort.
8478	0x0000211E	ERROR_DS_DS_REQUIRED	Der angeforderte Vorgang erfordert einen Verzeichnisdienst, und keiner war verfügbar.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
8479	0x0000211F	ERROR_DS_INVALID_LDAP_DISPLAY_NAME	Der angezeigte LDAP-Name der Klasse oder des Attributs enthält ASCII fremde Zeichen.
8480	0x00002120	ERROR_DS_NON_BASE_SEARCH	Der angeforderte Suchvorgang wird nur für base-Suche unterstützt.
8481	0x00002121	ERROR_DS_CANT_RETRIEVE_ATTS	Die Suche konnte nicht Attribute aus der Datenbank abzurufen.
8482	0x00002122	ERROR_DS_BACKLINK_WITHOUT_LINK	Der Schema Update-Vorgang versucht, ein Backwardlink-Attribut hinzuzufügen, die keine entsprechenden Forwardlink aufweist.
8483	0x00002123	ERROR_DS_EPOCH_MISMATCH	Quelle und Ziel einer domänenübergreifenden Verschiebung einverstanden nicht auf die Objekt Nummer Epoche. Quelle oder das Ziel muss nicht die neueste Version des Objekts.
8484	0x00002124	ERROR_DS_SRC_NAME_MISMATCH	Quelle und Ziel einer domänenübergreifenden Verschiebung einverstanden nicht auf der aktuellen Objektname. Quelle oder das Ziel muss nicht die neueste Version des Objekts.
8485	0x00002125	ERROR_DS_SRC_AND_DST_NC_IDENTICAL	Quelle und Ziel einer domänenübergreifenden Verschiebevorgangs sind identisch. Anrufer sollten lokale Verschiebevorgang anstelle von cross Domain Verschiebevorgang verwenden.
8486	0x00002126	ERROR_DS_DST_NC_MISMATCH	Quelle und Ziel einer domänenübergreifenden Verschiebung sind auf die Namenskontexte in der Gesamtstruktur nicht einig. Quelle oder das Ziel muss nicht die neueste Version von den Partitionscontainer.
8487	0x00002127	ERROR_DS_NOT_AUTHORITY_FOR_DST_NC	Ziel einer domänenübergreifenden Verschiebung ist nicht autorisierend für den Ziel-Namenskontext.
8488	0x00002128	ERROR_DS_SRC_GUID_MISMATCH	Quelle und Ziel einer domänenübergreifenden Verschiebung einverstanden nicht über die Identität des Quellobjekts. Quelle oder das Ziel muss nicht die neueste Version des Quellobjekts.
8489	0x00002129	ERROR_DS_CANT_MOVE_DELETED_OBJECT	Objekt, das domänenübergreifend verschoben werden ist bereits bekannt, vom Zielserver gelöscht werden. Der Quellserver muss nicht die neueste Version des Quellobjekts.
8490	0x0000212A	ERROR_DS_PDC_OPERATION_IN_PROGRESS	Ein anderer Vorgang, der exklusiven Zugriff auf die PDC PSMO benötigt, ist bereits im Gange.
8491	0x0000212B	ERROR_DS_CROSS_DOMAIN_CLEANUP_REQD	Eines domänenübergreifenden Verschiebevorgangs ist fehlgeschlagen, so dass die zwei Versionen des verschobenen Objekts vorhanden - jeweils in den Quell- und Ziel-Domänen. Das Zielobjekt muss entfernt werden, um das System in einen konsistenten Zustand wiederherzustellen.
8492	0x0000212C	ERROR_DS_ILLEGAL_XDOM_MOVE_OPERATION	Dieses Objekt kann nicht verschoben werden über Domänengrenzen entweder weil domänenübergreifenden bewegt für diese Klasse sind nicht zulässig, oder das Objekt einige Besonderheiten, z. B. hat: Treuhandkonto oder eingeschränkt RID, die verhindern, dass seine Bewegung.
8493	0x0000212D	ERROR_DS_CANT_WITH_ACCT_GROUP_MEMBERSHPS	Objekte mit Mitgliedschaften kann nicht über Domänengrenzen verschoben werden, da einmal bewegt, das Bedingungen für die Mitgliedschaft von der Gruppe der verletzen würde. Entfernen Sie das Objekt von jedem Konto Gruppenmitgliedschaften und wiederholen.
8494	0x0000212E	ERROR_DS_NC_MUST_HAVE_NC_PARENT	Ein naming Kontext Kopf muss die unmittelbar untergeordneten ein weiteres naming Kontext Kopf, keinen inneren Knoten sein.



Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
8495	0x0000212F	ERROR_DS_CR_IMPOSSIBLE_TO_VALIDATE	Das Verzeichnis kann nicht den vorgeschlagenen Kontextnamen für naming überprüfen, weil es kein Replikat des Namenskontextes oberhalb des vorgeschlagenen Namenskontextes hält. Bitte stellen Sie sicher, dass die Domänennamenmaster-Rolle von einem Server stattfindet, die als globale Katalogserver konfiguriert ist und der Server auf dem neuesten Stand der Replikationspartner ist. (Gilt nur für Windows 2000 Domain Naming Master)
8496	0x00002130	ERROR_DS_DST_DOMAIN_NOT_NATIVE	Zieldomäne muss sich im einheitlichen Modus.
8497	0x00002131	ERROR_DS_MISSING_INFRASTRUCTURE_CONTAINER	Der Vorgang kann nicht ausgeführt werden, weil der Server nicht verfügbar einen Infrastruktur-Container in der Domäne von Interesse sind.
8498	0x00002132	ERROR_DS_CANT_MOVE_ACCOUNT_GROUP	Domänenübergreifende Verschieben von nicht leeren Kontogruppen ist nicht erlaubt.
8499	0x00002133	ERROR_DS_CANT_MOVE_RESOURCE_GROUP	Domänenübergreifende Verschieben von nicht leeren Ressourcengruppen ist nicht erlaubt.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
8500	0x00002134	ERROR_DS_INVALID_SEARCH_FLAG	Die Search-Flags für das Attribut sind ungültig. Das ANR-Bit ist nur für Attribute von Unicode- oder Teletex-Zeichenfolgen gültig.
8501	0x00002135	ERROR_DS_NO_TREE_DELETE_ABOVE_NC	Baum-Löschungen, die an ein-mit einem NC-Kopf als abhängiges Objekt beginnen, sind nicht zulässig.
8502	0x00002136	ERROR_DS_COULDNT_LOCK_TREE_FOR_DELETE	Der Verzeichnisdienst konnte eine Struktur zur Vorbereitung einer Strukturlöschung zu sperren, da die Struktur verwendet wurde.
8503	0x00002137	ERROR_DS_COULDNT_IDENTIFY_OBJECTS_FOR_TREE_DELETE	Der Verzeichnisdienst konnte die Liste der zu löschenden beim Versuch einer Strukturlöschung Objekte zu identifizieren.
8504	0x00002138	ERROR_DS_SAM_INIT_FAILURE	Sicherheitskontenverwaltung ist aufgrund des folgenden Fehlers fehlgeschlagen: %1. Fehlerstatus: 0 x %2. Klicken Sie auf OK, um das System Herunterfahren und Neustart in Verzeichnisdienste wiederherstellen. Überprüfen Sie das Ereignisprotokoll detaillierte Informationen.
8505	0x00002139	ERROR_DS_SENSITIVE_GROUP_VIOLATION	Nur ein Administrator kann die Liste der Mitglieder einer administrativen Gruppe ändern.
8506	0x0000213A	ERROR_DS_CANT_MOD_PRIMARYGROUPID	Die primäre Gruppenkennung eines Domänencontrollerkontos kann nicht geändert werden.
8507	0x0000213B	ERROR_DS_ILLEGAL_BASE_SCHEMA_MOD	Wird versucht, das Basisschema zu ändern.
8508	0x0000213C	ERROR_DS_NONSAFE_SCHEMA_CHANGE	Hinzufügen eines neuen obligatorischen Attributs zu einer vorhandenen Klasse, ein obligatorisches Attribut aus einer vorhandenen Klasse löschen oder Hinzufügen von ein optionales Attribut auf Sonderklasse ist Spitze, die kein Backlink Attribut (direkt oder durch Vererbung, z. B. durch Hinzufügen oder Löschen einer Erweiterungsklasse) ist nicht zulässig.
8509	0x0000213D	ERROR_DS_SCHEMA_UPDATE_DISALLOWED	Schemaaktualisierung ist auf diesem DC nicht zulässig, weil der DC nicht das Schema-FSMO-Funktionsbesitzer ist.
8510	0x0000213E	ERROR_DS_CANT_CREATE_UNDER_SCHEMA	Ein Objekt dieser Klasse kann unter dem Schemacontainer nicht erstellt werden. Sie können nur Objekte unter dem Schemacontainer Attributschema und Klasse-Schema erstellen.
8511	0x0000213F	ERROR_DS_INSTALL_NO_SRC_SCH_VERSION	Installation des Replikats bzw. untergeordneten konnte das Attribut ObjectVersion auf dem Schemacontainer auf der Quell-DC. Entweder fehlt auf dem Schemacontainer das Attribut, oder die angegebenen Anmeldeinformationen sind nicht berechtigt, es zu lesen.
8512	0x00002140	ERROR_DS_INSTALL_NO_SCH_VERSION_IN_IFILE	Die Installation des Replikats bzw. untergeordneten konnte das Attribut ObjectVersion im Abschnitt SCHEMA der Datei im system32-Verzeichnis.
8513	0x00002141	ERROR_DS_INVALID_GROUP_TYPE	Der angegebene Gruppentyp ist ungültig.
8514	0x00002142	ERROR_DS_NO_NEST_GLOBALGROUP_IN_MIXEDDOMAIN	Globale Gruppen in einer Domäne im gemischten Modus können nicht verschachtelt werden, wenn die Gruppe Sicherheit aktiviert ist.
8515	0x00002143	ERROR_DS_NO_NEST_LOCALGROUP_IN_MIXEDDOMAIN	Lokale Gruppen in einer Domäne im gemischten Modus können nicht verschachtelt werden, wenn die Gruppe Sicherheit aktiviert ist.
8516	0x00002144	ERROR_DS_GLOBAL_CANT_HAVE_LOCAL_MEMBER	Eine globale Gruppe nicht als Mitglied eine lokale Gruppe.
8517	0x00002145	ERROR_DS_GLOBAL_CANT_HAVE_UNIVERSAL_MEMBER	Eine globale Gruppe kann keine universelle Gruppe als Mitglied haben.
8518	0x00002146	ERROR_DS_UNIVERSAL_CANT_HAVE_LOCAL_MEMBER	Eine universelle Gruppe kann nicht als Mitglied eine lokale Gruppe haben.
8519	0x00002147	ERROR_DS_GLOBAL_CANT_HAVE_CROSSDOMAIN_MEMBER	Eine globale Gruppe kann nicht Mitglied domänenübergreifende haben.
8520	0x00002148	ERROR_DS_LOCAL_CANT_HAVE_CROSSDOMAIN_LOCAL_MEMBER	Eine lokale Gruppe kann nicht als Mitglied einer anderen domänenübergreifende lokale Gruppe haben.

Fehler			Beschreibung
Dezi- mal	Hexadezimal	Name	
8521	0x00002149	ERROR_DS_HAVE_PRIMARY_MEMBERS	Eine Gruppe mit primären Mitgliedern kann nicht in eine Gruppe mit deaktivierter Sicherheit ändern.
8522	0x0000214A	ERROR_DS_STRING_SD_CONVERSION_FAILED	Der Cacheload des Schemas konnte die Zeichenfolge Standard-SD bei einem Klassenschemaobjekt konvertieren.
8523	0x0000214B	ERROR_DS_NAMING_MASTER_GC	Nur DSAs, die Globalkatalogserver konfiguriert darf die Domain Naming Master FSMO-Rolle zu halten. (Gilt nur für Windows 2000 Server)
8524	0x0000214C	ERROR_DS_LOOKUP_FAILURE	Der DSA-Vorgang kann aufgrund eines DNS-Lookup nicht fortfahren.
8525	0x0000214D	ERROR_DS_COULDNT_UPDATE_SPNS	Beim Verarbeiten einer Änderung des DNS-Hostnamens für ein Objekt, konnte der Dienstprinzipalnamens-Werte nicht synchron gehalten werden.
8526	0x0000214E	ERROR_DS_CANT_RETRIEVE_SD	Das Attribut Sicherheitsbeschreibung konnte nicht gelesen werden.
8527	0x0000214F	ERROR_DS_KEY_NOT_UNIQUE.	Das angeforderte Objekt wurde nicht gefunden, aber ein Objekt mit diesem Schlüssel wurde gefunden.
8528	0x00002150	ERROR_DS_WRONG_LINKED_ATT_SYNTAX	Die Syntax für die verknüpfte attributierten hinzugefügt wird, ist falsch. Forwardlinks haben nur die Syntax 2.5.5.1, 2.5.5.7 und 2.5.5.14 und Backlinks nur die Syntax 2.5.5.1 möglich.
8529	0x00002151	ERROR_DS_SAM_NEED_BOOTKEY_PASSWOR D	Die Sicherheitskontenverwaltung muss das Boot-Passwort zu erhalten.
8530	0x00002152	ERROR_DS_SAM_NEED_BOOTKEY_FLOPPY	Die Sicherheitskontenverwaltung muss den Schlüssel Boot von Diskette zu erhalten.
8531	0x00002153	ERROR_DS_CANT_START	Verzeichnisdienst kann nicht gestartet werden.
8532	0x00002154	ERROR_DS_INIT_FAILURE	Verzeichnisdienste konnten nicht gestartet werden.
8533	0x00002155	ERROR_DS_NO_PKT_PRIVACY_ON_CONNECTI ON	Die Verbindung zwischen Client und Server erfordert Paketsicherheit oder besser.
8534	0x00002156	ERROR_DS_SOURCE_DOMAIN_IN_FOREST	Die Quelldomäne möglicherweise nicht in der gleichen Gesamtstruktur als Ziel.
8535	0x00002157	ERROR_DS_DESTINATION_DOMAIN_NOT_IN_F OREST	Die Zieldomäne muss sich im Wald.
8536	0x00002158	ERROR_DS_DESTINATION_AUDITING_NOT_EN ABLED	Der Vorgang erfordert, dass Ziel Domain Überwachung aktiviert werden.
8537	0x00002159	ERROR_DS_CANT_FIND_DC_FOR_SRC_DOMAI N	Der Vorgang konnte keinen DC für die Quelldomäne finden.
8538	0x0000215A	ERROR_DS_SRC_OBJ_NOT_GROUP_OR_USER	Das Quellobjekt muss eine Gruppe oder einen Benutzer sein.
8539	0x0000215B	ERROR_DS_SRC_SID_EXISTS_IN_FOREST	SID des Quellobjekts ist bereits in Ziel-Gesamtstruktur.
8540	0x0000215C	ERROR_DS_SRC_AND_DST_OBJECT_CLASS_M ISMATCH	Das Quell- und Ziel-Objekt muss vom gleichen Typ sein.
8541	0x0000215D	ERROR_SAM_INIT_FAILURE	Sicherheitskontenverwaltung ist aufgrund des folgenden Fehlers fehlgeschlagen: %1. Fehlerstatus: 0 x %2. Klicken Sie auf OK, um das System Herunterfahren und starten Sie im abgesicherten Modus neu. Überprüfen Sie das Ereignisprotokoll detaillierte Informationen.
8542	0x0000215E	ERROR_DS_DRA_SCHEMA_INFO_SHIP	Schemainformationen konnten nicht in die Replikationsanforderung einbezogen.
8543	0x0000215F	ERROR_DS_DRA_SCHEMA_CONFLICT	Der Replikationsvorgang konnte nicht abgeschlossen werden, aufgrund einer Inkompatibilität Schema.
8544	0x00002160	ERROR_DS_DRA_EARLIER_SCHEMA_CONFLICT	Der Replikationsvorgang konnte nicht abgeschlossen werden, aufgrund einer vorherigen Schema-Inkompatibilität.
8545	0x00002161	ERROR_DS_DRA_OBJ_NC_MISMATCH	Die Replikationsaktualisierung konnte nicht angewendet werden, da die Quelle oder das Ziel hat, noch nicht erhaltenen Informationen bezüglich eines letzten domänenübergreifenden verschieben-Operation.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
8546	0x00002162	ERROR_DS_NC_STILL_HAS_DSAS	Die angeforderte Domäne konnte nicht gelöscht werden, denn es gibt ein Domänencontroller, die immer noch diese Domain hosten.
8547	0x00002163	ERROR_DS_GC_REQUIRED	Der angeforderte Vorgang kann nur auf einem globalen Katalogserver ausgeführt werden.
8548	0x00002164	ERROR_DS_LOCAL_MEMBER_OF_LOCAL_ONLY	Eine lokale Gruppe kann nur Mitglied in anderen lokalen Gruppen in derselben Domäne sein.
8549	0x00002165	ERROR_DS_NO_FPO_IN_UNIVERSAL_GROUPS	Fremde Sicherheitsprinzipale können keine Mitglieder von universellen Gruppen sein.
8550	0x00002166	ERROR_DS_CANT_ADD_TO_GC	Das Attribut ist nicht zulässig, aus Sicherheitsgründen auf GC repliziert werden.
8551	0x00002167	ERROR_DS_NO_CHECKPOINT_WITH_PDC	Der Prüfpunkt mit dem PDC konnte nicht eingenommen werden, denn es gibt zu viele Änderungen derzeit verarbeitet wird.
8552	0x00002168	ERROR_DS_SOURCE_AUDITING_NOT_ENABLED	Der Vorgang erfordert die Quelle Domäne Überwachung aktiviert werden.
8553	0x00002169	ERROR_DS_CANT_CREATE_IN_NONDOMAIN_NC	Sicherheitsprinzipalobjekte können nur im Domänennamenkontext erstellt werden.
8554	0x0000216A	ERROR_DS_INVALID_NAME_FOR_SPN	Ein Service Principal Name (SPN) konnte nicht erstellt werden, da der angegebene Hostname nicht im notwendigen Format ist.
8555	0x0000216B	ERROR_DS_FILTER_USES_CONSTRUCTED_ATTRIBUTES	Ein Filter wurde übergeben, die konstruierten Attribute verwendet.
8556	0x0000216C	ERROR_DS_UNICODEPWD_NOT_IN_QUOTES	Der Attributwert "unicodePWD" muss in Anführungszeichen eingeschlossen werden.
8557	0x0000216D	ERROR_DS_MACHINE_ACCOUNT_QUOTA_EXCEEDED	Ihr Computer konnte der Domäne nicht hinzugefügt. Sie haben die maximale Anzahl der Computerkonten überschritten, die Sie berechtigt sind, in dieser Domäne zu erstellen. Wenden Sie sich an den Systemadministrator, um diese Grenze zurücksetzen oder erhöht haben.
8558	0x0000216E	ERROR_DS_MUST_BE_RUN_ON_DST_DC	Aus Sicherheitsgründen muss der Vorgang auf der Ziel-DC ausgeführt werden.
8559	0x0000216F	ERROR_DS_SRC_DC_MUST_BE_SP4_OR_GREATER	Aus Sicherheitsgründen muss der Quell-DC NT4SP4 sein oder größer.
8560	0x00002170	ERROR_DS_CANT_TREE_DELETE_CRITICAL_OBJ	Kritische Directory Service System-Objekte können während der Baum-Delete-Operationen gelöscht werden. Das Baum löschen kann teilweise durchgeführt wurden.
8561	0x00002171	ERROR_DS_INIT_FAILURE_CONSOLE	Verzeichnisdienste konnte aufgrund des folgenden Fehlers nicht gestartet: %1. Fehlerstatus: 0 x %2. Bitte klicken Sie auf OK zum Herunterfahren des Systems. Sie können die Wiederherstellungskonsole verwenden, um das System weiter zu diagnostizieren.
8562	0x00002172	ERROR_DS_SAM_INIT_FAILURE_CONSOLE	Sicherheitskontenverwaltung ist aufgrund des folgenden Fehlers fehlgeschlagen: %1. Fehlerstatus: 0 x %2. Bitte klicken Sie auf OK zum Herunterfahren des Systems. Sie können die Wiederherstellungskonsole verwenden, um das System weiter zu diagnostizieren.
8563	0x00002173	ERROR_DS_FOREST_VERSION_TOO_HIGH	Diese Version von Windows ist zu alt, um das aktuelle Verzeichnis Wald Verhalten zu unterstützen. Sie müssen das Betriebssystem auf diesem Server aktualisieren, bevor sie einen Domänencontroller in dieser Gesamtstruktur werden kann.
8564	0x00002174	ERROR_DS_DOMAIN_VERSION_TOO_HIGH	Diese Windows-Version ist zu alt, um das aktuelle Domain-Verhalten zu unterstützen. Sie müssen das Betriebssystem auf diesem Server aktualisieren, bevor sie einen Domänencontroller in dieser Domäne werden kann.
8565	0x00002175	ERROR_DS_FOREST_VERSION_TOO_LOW	Diese Windows-Version unterstützt nicht mehr die Verhaltensversion in diesem Directory-Gesamtstruktur verwendet. Sie müssen die Verhaltensversion Wald ausbauen, bevor dieser Server ein Domänencontroller in der Gesamtstruktur werden kann.

Fehler			Beschreibung
Dezi- mal	Hexadezimal	Name	
8566	0x00002176	ERROR_DS_DOMAIN_VERSION_TOO_LOW	Diese Windows-Version unterstützt nicht mehr die Verhaltensversion in dieser Domäne verwendet. Sie müssen die Verhaltensversion der Domäne voraus, bevor dieser Server ein Domänencontroller in der Domäne werden kann.
8567	0x00002177	ERROR_DS_INCOMPATIBLE_VERSION	Der Windows-Version ist inkompatibel mit der Verhaltensversion der Domäne oder Gesamtstruktur.
8568	0x00002178	ERROR_DS_LOW_DSA_VERSION	Die Verhaltensversion kann nicht auf den angeforderten Wert erhöht werden, weil noch Domänencontroller mit Versionen kleiner als der angeforderte Wert vorhanden sind.
8569	0x00002179	ERROR_DS_NO_BEHAVIOR_VERSION_IN_MIXEDDOMAIN	Der Versionswert Verhalten kann nicht erhöht werden, während die Domäne noch im gemischten Domänenmodus befindet. Zuerst müssen Sie die Domäne in den einheitlichen Modus ändern, bevor die Verhaltensversion erhöht.
8570	0x0000217A	ERROR_DS_NOT_SUPPORTED_SORT_ORDER	Die angeforderte Sortierreihenfolge wird nicht unterstützt.
8571	0x0000217B	ERROR_DS_NAME_NOT_UNIQUE	Gefunden Sie ein Objekt mit nicht eindeutigen Namen.
8572	0x0000217C	ERROR_DS_MACHINE_ACCOUNT_CREATED_PRENT4	Das Computerkonto wurde vor NT4 erstellt. Das Konto muss neu erstellt werden.
8573	0x0000217D	ERROR_DS_OUT_OF_VERSION_STORE	Die Datenbank ist außerhalb des Versionspeichers.
8574	0x0000217E	ERROR_DS_INCOMPATIBLE_CONTROLS_USED	Nicht fortsetzen, weil mehrere widersprüchliche Steuerelemente verwendet wurden.
8575	0x0000217F	ERROR_DS_NO_REF_DOMAIN	Keine gültige Sicherheitsbeschreibung-Referenzdomäne für diese Partition gefunden.
8576	0x00002180	ERROR_DS_RESERVED_LINK_ID	Schemaaktualisierung ist fehlgeschlagen: der Verknüpfungskennung ist reserviert.
8577	0x00002181	ERROR_DS_LINK_ID_NOT_AVAILABLE	Schemaaktualisierung ist fehlgeschlagen: Es gibt keine Link-Bezeichner.
8578	0x00002182	ERROR_DS_AG_CANT_HAVE_UNIVERSAL_MEMBER	Eine Kontogruppe kann keine universelle Gruppe als Mitglied haben.
8579	0x00002183	ERROR_DS_MODIFYDN_DISALLOWED_BY_INSTANCE_TYPE	Umbenennen Sie oder verschieben Sie, sind auf Benennung Kontext Köpfe oder schreibgeschützten Objekten nicht zulässig.
8580	0x00002184	ERROR_DS_NO_OBJECT_MOVE_IN_SCHEMA_NAMESPACE	Verschiebevorgänge sind auf Objekten im Schemanamenskontext nicht zulässig.
8581	0x00002185	ERROR_DS_MODIFYDN_DISALLOWED_BY_FLAG	Ein Systemflag festgelegt wurde, auf das Objekt und lässt nicht das Objekt verschoben oder umbenannt werden.
8582	0x00002186	ERROR_DS_MODIFYDN_WRONG_GRANDPARENT	Dieses Objekt ist nicht zulässig, dessen Großeltern-Container ändern. Bewegungen sind für dieses Objekt nicht verboten, aber beschränken sich auf Geschwister-Container.
8583	0x00002187	ERROR_DS_NAME_ERROR_TRUST_REFERRAL	Nicht vollständig auflösen, eine Überweisung in eine andere Gesamtstruktur erstellt wird.
8584	0x00002188	ERROR_NOT_SUPPORTED_ON_STANDARD_SERVER	Die angeforderte Aktion ist auf standard-Server nicht unterstützt.
8585	0x00002189	ERROR_DS_CANT_ACCESS_REMOTE_PART_OF_AD	Eine Partition von Active Directory befindet sich auf einem Remoteserver konnte nicht zugegriffen werden. Stellen Sie sicher, dass mindestens ein Server für die Partition ausgeführt wird, in Frage.
8586	0x0000218A	ERROR_DS_CR_IMPOSSIBLE_TO_VALIDATE	Das Verzeichnis kann nicht den vorgeschlagenen Namen naming Context (oder Partition) überprüfen, denn es kein Replikat hält noch kann es ein Replikat des Namenskontextes oberhalb des vorgeschlagenen Namenskontextes kontaktieren. Bitte stellen Sie sicher, dass der übergeordnete Namenskontext richtig im DNS registriert ist und dass mindestens ein Replikat dieses Namenskontextes vom Domänennamensmaster erreichbar.
8587	0x0000218B	ERROR_DS_THREAD_LIMIT_EXCEEDED	Der Thread für diese Anforderung wurde überschritten.

Fehler			Beschreibung
Dezi- mal	Hexadezimal	Name	
8588	0x0000218C	ERROR_DS_NOT_CLOSEST	Der globale Katalogserver ist nicht im Schrank- Standort.

		Fehler	Beschreibung
Dezimal	Hexadezimal	Name	
9001	0x00002329	DNS_ERROR_RCODE_FORMAT_ERROR	DNS-Server nicht in der Lage, Format zu interpretieren.
9002	0x0000232A	DNS_ERROR_RCODE_SERVER_FAILURE	DNS-Server-Fehler.
9003	0x0000232B	DNS_ERROR_RCODE_NAME_ERROR	DNS-Name ist nicht vorhanden.
9004	0x0000232C	DNS_ERROR_RCODE_NOT_IMPLEMENTED	DNS-Anfrage vom Name-Server nicht unterstützt.
9005	0x0000232D	DNS_ERROR_RCODE_REFUSED	DNS Betrieb verweigert.
9006	0x0000232E	DNS_ERROR_RCODE_YXDOMAIN	DNS-Namen, die nicht existieren sollte, ist vorhanden.
9007	0x0000232F	DNS_ERROR_RCODE_YXRRSET	DNS-RR-Menge, die nicht vorhanden sein sollte, ist vorhanden.
9008	0x00002330	DNS_ERROR_RCODE_NXRRSET	DNS-RR-Reihe, die vorhanden sind, ist nicht vorhanden.
9009	0x00002331	DNS_ERROR_RCODE_NOTAUTH	DNS-Server nicht autorisierend für die Zone.
9010	0x00002332	DNS_ERROR_RCODE_NOTZONE	DNS-Name im Update oder Prereq ist nicht in der Zone.
9016	0x00002338	DNS_ERROR_RCODE_BADSIG	Die DNS-Signatur konnte nicht überprüft werden.
9017	0x00002339	DNS_ERROR_RCODE_BADKEY	DNS-bad Taste.
9018	0x0000233A	DNS_ERROR_RCODE_BADTIME	Gültigkeit der DNS-Signatur ist abgelaufen.
9501	0x0000251D	DNS_INFO_NO_RECORDS	Keine Datensätze gefunden für angegebenen DNS-Abfrage.
9502	0x0000251E	DNS_ERROR_BAD_PACKET	Schlechten DNS-Paket.
9503	0x0000251F	DNS_ERROR_NO_PACKET	Kein DNS-Paket.
9504	0x00002520	DNS_ERROR_RCODE	DNS-Fehler, Rcode überprüfen.
9505	0x00002521	DNS_ERROR_UNSECURE_PACKET	Ungesicherte DNS-Paket.
9551	0x0000254F	DNS_ERROR_INVALID_TYPE	Ungültiger DNS-Typ.
9552	0x00002550	DNS_ERROR_INVALID_IP_ADDRESS	Ungültige IP-Adresse.
9553	0x00002551	DNS_ERROR_INVALID_PROPERTY	Ungültige Eigenschaft.
9554	0x00002552	DNS_ERROR_TRY_AGAIN_LATER	Versuchen Sie DNS-Vorgang später noch einmal.
9555	0x00002553	DNS_ERROR_NOT_UNIQUE	Eintrag für den angegebenen Namen und Typ ist nicht eindeutig.
9556	0x00002554	DNS_ERROR_NON_RFC_NAME	DNS-Name entspricht nicht RFC-Spezifikationen.
9557	0x00002555	DNS_STATUS_FQDN	DNS-Name ist ein vollständig qualifizierter DNS-Name.
9558	0x00002556	DNS_STATUS_DOTTED_NAME	DNS-Name ist (Multilabel) punktiert.
9559	0x00002557	DNS_STATUS_SINGLE_PART_NAME	DNS-Name ist eine einteilige.
9560	0x00002558	DNS_ERROR_INVALID_NAME_CHAR	DSN-Name enthält ein ungültiges Zeichen.
9561	0x00002559	DNS_ERROR_NUMERIC_NAME	DNS-Name ist vollständig numerisch.
9601	0x00002581	DNS_ERROR_ZONE_DOES_NOT_EXIST	DNS-Zone ist nicht vorhanden.
9602	0x00002582	DNS_ERROR_NO_ZONE_INFO	DNS-Zoneninformationen nicht verfügbar.
9603	0x00002583	DNS_ERROR_INVALID_ZONE_OPERATION	Ungültiger Vorgang für die DNS-Zone.
9604	0x00002584	DNS_ERROR_ZONE_CONFIGURATION_ERROR	Ungültige DNS-Zonenkonfiguration.
9605	0x00002585	DNS_ERROR_ZONE_HAS_NO_SOA_RECORD	DNS-Zone ist kein Start des Autoritätsursprung (SOA)-Eintrags.
9606	0x00002586	DNS_ERROR_ZONE_HAS_NO_NS_RECORDS	DNS-Zone besitzt keinen Namenservereintrag (NS).
9607	0x00002587	DNS_ERROR_ZONE_LOCKED	DNS-Zone ist gesperrt.
9608	0x00002588	DNS_ERROR_ZONE_CREATION_FAILED	Fehler beim DNS-Zone erstellen.
9609	0x00002589	DNS_ERROR_ZONE_ALREADY_EXISTS	DNS-Zone ist bereits vorhanden.
9610	0x0000258A	DNS_ERROR_AUTOZONE_ALREADY_EXISTS	Automatische DNS-Zone ist bereits vorhanden.
9611	0x0000258B	DNS_ERROR_INVALID_ZONE_TYPE	Ungültiger DNS-Zonentyp.
9612	0x0000258C	DNS_ERROR_SECONDARY_REQUIRES_MASTER_IP	Sekundäre DNS-Zone erfordert master IP-Adresse.
9613	0x0000258D	DNS_ERROR_ZONE_NOT_SECONDARY	DNS-Zone ist nicht sekundär.
9614	0x0000258E	DNS_ERROR_NEED_SECONDARY_ADDRESSES	Benötigen Sie sekundäre IP-Adresse.
9615	0x0000258F	DNS_ERROR_WINS_INIT_FAILED	WINS-Initialisierung ist fehlgeschlagen.
9616	0x00002590	DNS_ERROR_NEED_WINS_SERVERS	Benötigen Sie WINS-Server.
9617	0x00002591	DNS_ERROR_NBSTAT_INIT_FAILED	NBSTAT-Initialisierungsaufwurf ist fehlgeschlagen.
9618	0x00002592	DNS_ERROR_SOA_DELETE_INVALID	Ungültiges Löschen des Autoritätsursprung (SOA)

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
9619	0x00002593	DNS_ERROR_FORWARDER_ALREADY_EXISTS	Eine bedingte Weiterleitung-Zone ist für diesen Namen bereits vorhanden.
9651	0x000025B3	DNS_ERROR_PRIMARY_REQUIRES_DATAFILE	Primäre DNS-Zone erfordert eine Datendatei.
9652	0x000025B4	DNS_ERROR_INVALID_DATAFILE_NAME	Ungültige Datafile Name für DNS-Zone.
9653	0x000025B5	DNS_ERROR_DATAFILE_OPEN_FAILURE	Öffnen Sie die Datendatei für die DNS-Zone konnte nicht.
9654	0x000025B6	DNS_ERROR_FILE_WRITEBACK_FAILED	Konnte die Datendatei für die DNS-Zone zu schreiben.
9655	0x000025B7	DNS_ERROR_DATAFILE_PARSING	Fehler beim Lesen der Datendatei für die DNS-Zone.
9701	0x000025E5	DNS_ERROR_RECORD_DOES_NOT_EXIST	DNS-Eintrag ist nicht vorhanden.
9702	0x000025E6	DNS_ERROR_RECORD_FORMAT	DNS-Datensatzformat Fehler.
9703	0x000025E7	DNS_ERROR_NODE_CREATION_FAILED	Knoten-Erstellung-Fehler im DNS.
9704	0x000025E8	DNS_ERROR_UNKNOWN_RECORD_TYPE	Unbekannten DNS-Datensatztyp.
9705	0x000025E9	DNS_ERROR_RECORD_TIMED_OUT	DNS-Eintrag Zeitüberschreitung.
9706	0x000025EA	DNS_ERROR_NAME_NOT_IN_ZONE	Name nicht in der DNS-Zone.
9707	0x000025EB	DNS_ERROR_CNAME_LOOP	CNAME-Schleife gefunden.
9708	0x000025EC	DNS_ERROR_NODE_IS_CNAME	Knoten ist ein CNAME DNS-Eintrag.
9709	0x000025ED	DNS_ERROR_CNAME_COLLISION	Ein CNAME-Eintrag für angegebenen Namen ist bereits vorhanden.
9710	0x000025EE	DNS_ERROR_RECORD_ONLY_AT_ZONE_ROOT	Notieren Sie nur am DNS-Zonenstamm.
9711	0x000025EF	DNS_ERROR_RECORD_ALREADY_EXISTS	DNS-Eintrag ist bereits vorhanden.
9712	0x000025F0	DNS_ERROR_SECONDARY_DATA	Sekundäre DNS-Zone Datenfehler.
9713	0x000025F1	DNS_ERROR_NO_CREATE_CACHE_DATA	DNS-Cachedaten konnten nicht erstellt werden.
9714	0x000025F2	DNS_ERROR_NAME_DOES_NOT_EXIST	DNS-Name ist nicht vorhanden.
9715	0x000025F3	DNS_WARNING_PTR_CREATE_FAILED	Zeigereintrag (PTR) konnte nicht erstellt werden.
9716	0x000025F4	DNS_WARNING_DOMAIN_UNDELETED	DNS-Domäne wurde wiederhergestellt.
9717	0x000025F5	DNS_ERROR_DS_UNAVAILABLE	Der Verzeichnisdienst ist nicht verfügbar.
9718	0x000025F6	DNS_ERROR_DS_ZONE_ALREADY_EXISTS	DNS-Zone ist bereits im Verzeichnisdienst vorhanden.
9719	0x000025F7	DNS_ERROR_NO_BOOTFILE_IF_DS_ZONE	DNS-Server nicht erstellen oder lesen die Startdatei für den Verzeichnisdienst integrierte DNS-Zone.
9751	0x00002617	DNS_INFO_AXFR_COMPLETE	DNS-AXFR (Zonenübertragung) abgeschlossen.
9752	0x00002618	DNS_ERROR_AXFR	DNS-Zonenübertragung ist fehlgeschlagen.
9753	0x00002619	DNS_INFO_ADDED_LOCAL_WINS	Zusätzlichen lokalen WINS-Server.
9801	0x00002649	DNS_STATUS_CONTINUE_NEEDED	Sichere Updates Aufruf muss Aktualisierungsanforderung fortsetzen.
9851	0x0000267B	DNS_ERROR_NO_TCPIP	Netzwerkprotokoll TCP/IP nicht installiert
9852	0x0000267C	DNS_ERROR_NO_DNS_SERVERS	Keine DNS-Server für das lokale System konfiguriert.
9901	0x000026AD	DNS_ERROR_DP_DOES_NOT_EXIST	Die angegebene Verzeichnispertition existiert nicht.
9902	0x000026AE	DNS_ERROR_DP_ALREADY_EXISTS	Die angegebene Verzeichnispertition ist bereits vorhanden.
9903	0x000026AF	DNS_ERROR_DP_NOT_ENLISTED	Der DS ist nicht in der angegebenen Verzeichnispertition eingetragen.
9904	0x000026B0	DNS_ERROR_DP_ALREADY_ENLISTED	Der DS wird in die angegebene Verzeichnispertition bereits eingetragen.



Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
10004	0x00002714	WSAEINTR	Ein Blockierungsvorgang wurde durch einen Aufruf von WSACancelBlockingCall unterbrochen.
10009	0x00002719	WSAEBADF	Gelieferten Datei-Handle ist ungültig.
10013	0x0000271D	WSAEACCES	Es wurde versucht, auf einen Socket in einer Weise, die durch die Zugriffsberechtigungen verboten.
10014	0x0000271E	WSAEFAULT	Das System entdeckt eine ungültige Zeigeradresse bei dem Versuch, das Zeigerargument in einem Aufruf verwenden.
10022	0x00002726	WSAEINVAL	Ein ungültiges Argument wurde angegeben.
10024	0x00002728	WSAEMFILE	Zu viele geöffnete Sockets.
10035	0x00002733	WSAEWOULDBLOCK	Ein nicht blockierender Socketvorgang konnte nicht sofort abgeschlossen werden.
10036	0x00002734	WSAEINPROGRESS	Ein Blockierungsvorgang wird momentan ausgeführt.
10037	0x00002735	WSAEALREADY	Ein Vorgang bezog sich auf einem nicht blockierenden Socket, die bereits einen Vorgang ausgeführt.
10038	0x00002736	WSAENOTSOCK	Ein Vorgang bezog sich auf etwas, das kein Socket ist.
10039	0x00002737	WSAEDESTADDRREQ	Eine erforderliche Adresse wurde aus einem Vorgang auf einem Socket weggelassen.
10040	0x00002738	WSAEMSGSIZE	Eine Meldung, die über einen Datagrammsocket gesendet wurde größer als den internen Nachrichtenpuffer oder einige andere Netzwerk-Limit, oder der Puffer verwendet, um ein Datagramm zu empfangen war kleiner als das Datagramm selbst.
10041	0x00002739	WSAEPROTOTYPE	Im Funktionsaufruf Socket wurde ein Protokoll angegeben, das Semantik des angeforderten Sockettyps nicht unterstützt.
10042	0x0000273A	WSAENOPROTOOPT	In einem Aufruf von Getsockopt oder Setsockopt wurde eine nicht unterstützte Option bzw. Ebene angegeben.
10043	0x0000273B	WSAEPROTONOSUPPORT	Das angeforderte Protokoll wurde nicht in das System konfiguriert, oder keine Implementierung dafür vorhanden.
10044	0x0000273C	WSAESOCKTOSUPPORT	Die Unterstützung für den angegebenen Sockettyp existiert nicht in dieser Adressfamilie.
10045	0x0000273D	WSAEOPNOTSUPP	Der Vorgang wird nicht unterstützt für den Typ des Objekts, auf die verwiesen wird.
10046	0x0000273E	WSAEPFNOSUPPORT	Die Protokollfamilie wurde nicht in das System konfiguriert oder keine Implementierung dafür vorhanden.
10047	0x0000273F	WSAEAFNOSUPPORT	Eine Adresse, die nicht kompatibel mit dem angeforderten Protokoll wurde verwendet.
10048	0x00002740	WSAEADDRINUSE	Regel nicht nur eine Verwendung von jede Socketadresse (Protokoll-Netzwerk-Adresse/ Anschluss) ist zulässig.
10049	0x00002741	WSAEADDRNOTAVAIL	Die angeforderte Adresse ist in seinem Kontext ungültig.
10050	0x00002742	WSAENETDOWN	Bei einem Socketvorgang ist ein tot Netzwerk aufgetreten.
10051	0x00002743	WSAENETUNREACH	Ein Socketvorgang bezog sich auf ein Netzwerk nicht erreichbar.
10052	0x00002744	WSAENETRESET	Die Verbindung wurde aufgrund von KeepAlive-Aktivitäten erkennen eines Fehlers während die Operation im Gange war gebrochen.
10053	0x00002745	WSAECONNABORTED	Eine bestehende Verbindung wurde von der Software in Ihrem Hostcomputer abgebrochen.
10054	0x00002746	WSAECONNRESET	Eine vorhandene Verbindung wurde vom Remotehost geschlossen.
10055	0x00002747	WSAENOBUFS	Ein Vorgang auf einem Socket konnte nicht ausgeführt werden, weil das System ausreichend Pufferspeicher fehlte oder eine Warteschlange voll war.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
10056	0x00002748	WSAEISCONN	Eine Verbindungsanforderung wurde auf einen bereits verbundenen Socket.
10057	0x00002749	WSAENOTCONN	Eine Anforderung zum Senden oder empfangen von Daten wurde verhindert, da der Socket nicht verbunden ist und (beim Senden über einen Datagrammsocket mit einem Sendto-Aufruf) keine Adresse angegeben wurde.
10058	0x0000274A	WSAESHUTDOWN	Eine Anforderung zum Senden oder empfangen von Daten wurde verhindert, da der Socket hatte bereits in diese Richtung mit einem vorangegangenen Shutdown-Aufruf heruntergefahren worden.
10059	0x0000274B	WSAETOOMANYREFS	Zu viele Verweise auf einige Kernelobjekt.
10060	0x0000274C	WSAETIMEDOUT	Ein Verbindungsversuch ist fehlgeschlagen, da die verbundene Partei nicht richtig geantwortet hat, nach einer gewissen Zeit, oder die hergestellte Verbindung ist fehlgeschlagen, da der verbundene Host nicht reagiert hat.
10061	0x0000274D	WSAECONNREFUSED	Keine Verbindung konnte hergestellt werden, da der Zielcomputer aktiv es verweigert.
10062	0x0000274E	WSAELOOP	Name kann nicht übersetzt werden.
10063	0x0000274F	WSAENAMETOOLONG	Namenskomponente oder Name war zu lang.
10064	0x00002750	WSAEHOSTDOWN	Bei einem Socketvorgang ist fehlgeschlagen, da der Zielhost gedrückt wurde.
10065	0x00002751	WSAEHOSTUNREACH	Ein Socketvorgang bezog sich auf einen Host nicht erreichbar.
10066	0x00002752	WSAENOTEMPTY	Ein Verzeichnis kann nicht entfernt werden, die nicht leer ist.
10067	0x00002753	WSAEPROCLIM	Eine Windows Sockets-Implementation kann ein Limit für die Anzahl der Anwendungen bestehen, die zugleich auf diese zugreifen kann.
10068	0x00002754	WSAEUSERS	Lief der Quote.
10069	0x00002755	WSAEDQUOT	Datenträgerkontingent lief.
10070	0x00002756	WSAESTALE	Datei-Handle-Verweis ist nicht mehr verfügbar.
10071	0x00002757	WSAEREMOTE	Element ist nicht lokal verfügbar.
10091	0x0000276B	WSASYSNOTREADY	WSAStartup funktioniert Funktion nicht zu dieser Zeit, da das zugrunde liegende System verwendeten Netzwerkdienste derzeit nicht verfügbar ist.
10092	0x0000276C	WSAVERNOTSUPPORTED	Die angeforderte Windows Sockets-Version wird nicht unterstützt.
10093	0x0000276D	WSANOTINITIALISED	Die Anwendung hat WSAStartup nicht aufgerufen, oder WSAStartup fehlgeschlagen.
10101	0x00002775	WSAEDISCON	Wird von WSARcv oder WSARcvFrom, um anzugeben, dass die Gegenseite eine ordnungsgemäßes Herunterfahren-Sequenz eingeleitet hat.
10102	0x00002776	WSAENOMORE	Von WSALookupServiceNext können keine weiteren Ergebnisse zurückgegeben werden.
10103	0x00002777	WSAECANCELLED	WSALookupServiceEnd wurde aufgerufen, während dieser Aufruf noch verarbeitet wurde. Der Aufruf wurde abgebrochen.
1050°	0x00002778	WSAEINVALIDPROCTABLE	Die Prozedur-Aufruf-Tabelle ist ungültig.
10105	0x00002779	WSAEINVALIDPROVIDER	Der angeforderte Dienstanbieter ist ungültig.
10106	0x0000277A	WSAEPROVIDERFAILEDINIT	Der angeforderte Dienstanbieter konnte nicht geladen oder initialisiert werden.
10107	0x0000277B	WSASYSALLFAILURE	Ein Systemaufruf, der nie ausfällt ist fehlgeschlagen.
10108	0x0000277C	WSASERVICE_NOT_FOUND	Keine solchen Dienst ist bekannt. Der Dienst kann nicht im angegebenen Namespace gefunden werden.
10109	0x0000277D	WSATYPE_NOT_FOUND	Die angegebene Klasse wurde nicht gefunden.
10110	0x0000277E	WSA_E_NO_MORE	Von WSALookupServiceNext können keine weiteren Ergebnisse zurückgegeben werden.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
10111	0x0000277F	WSA_E_CANCELLED	WSALookupServiceEnd wurde aufgerufen, während dieser Aufruf noch verarbeitet wurde. Der Aufruf wurde abgebrochen.
10112	0x00002780	WSAEREFUSED	Eine Datenbankabfrage ist fehlgeschlagen, da es aktiv verweigert wurde.
11001	0x00002AF9	WSAHOST_NOT_FOUND	Keine solche Host ist unbekannt.
11002	0x00002AFA	WSATRY_AGAIN	Dies ist normalerweise ein zeitweiliger Fehler bei der Auflösung von Hostnamen und bedeutet, dass der lokale Server keine Antwort aus einem autorisierenden Server erhalten.
11003	0x00002AFB	WSANO_RECOVERY	Während eine Datenbanksuche ist ein nicht behebbarer Fehler aufgetreten.
11004	0x00002AFC	WSANO_DATA	Der angeforderte Name ist gültig und wurde in der Datenbank gefunden, aber es verfügt nicht über die richtigen zugeordneten Daten, die für gelöst.
11005	0x00002AFD	WSA_QOS_RECEIVERS	Mindestens eine Reserve ist angekommen.
11006	0x00002AFE	WSA_QOS_SENDERS	Mindestens einen Pfad ist angekommen.
11007	0x00002AFF	WSA_QOS_NO_SENDERS	Es gibt keine Absender.
11008	0x00002B00	WSA_QOS_NO_RECEIVERS	Es gibt keine Empfänger.
11009	0x00002B01	WSA_QOS_REQUEST_CONFIRMED	Reserve ist bestätigt worden.
11010	0x00002B02	WSA_QOS_ADMISSION_FAILURE	Fehler aufgrund mangelnder Ressourcen.
11011	0x00002B03	WSA_QOS_POLICY_FAILURE	Aus administrativen Gründen - schlechte Anmeldeinformationen abgelehnt.
11012	0x00002B04	WSA_QOS_BAD_STYLE	Unbekannte oder widersprüchlichen Stil.
11013	0x00002B05	WSA_QOS_BAD_OBJECT	Problem mit einen Teil des Puffers Filter- oder Providerspecific im Allgemeinen.
11014	0x00002B06	WSA_QOS_TRAFFIC_CTRL_ERROR	Problem mit einem Teil der Flowspec.
11015	0x00002B07	WSA_QOS_GENERIC_ERROR	General QOS-Fehler.
11016	0x00002B08	WSA_QOS_ESERVICETYPE	Eine ungültige oder unbekannte Dienstyp wurde der Flowspec gefunden.
11017	0x00002B09	WSA_QOS_EFLOWSPEC	In der QOS-Struktur wurde eine ungültige oder inkonsistente Flowspec gefunden.
11018	0x00002B0A	WSA_QOS_EPROVSPECBUF	Ungültige QOS-spezifischen Anbieterpuffer.
11019	0x00002B0B	WSA_QOS_EFILTERSTYLE	Ein ungültiger QOS-Filter-Stil wurde verwendet.
11020	0x00002B0C	WSA_QOS_EFILTERTYPE	Ein ungültiger QOS-Filter-Typ verwendet wurde.
11021	0x00002B0D	WSA_QOS_EFILTERCOUNT	Eine falsche Anzahl an QOS FILTERSPECs wurden in die FLOWDESCRIPTOR angegeben.
11022	0x00002B0E	WSA_QOS_EOBJLENGTH	Im QOS-spezifischen Anbieterpuffer wurde ein Objekt mit einem ungültigen ObjectLength-Feld angegeben.
11023	0x00002B0F	WSA_QOS_EFLOWCOUNT	In der QOS-Struktur wurde eine falsche Anzahl von Flow-Deskriptoren angegeben.
11024	0x00002B10	WSA_QOS_EUNKNOWNPSOBJ	Im QOS-spezifischen Anbieterpuffer wurde ein unbekanntes Objekt gefunden.
11025	0x00002B11	WSA_QOS_EPOLICYOBJ	Im QOS-spezifischen Anbieterpuffer wurde ein ungültiges Richtlinienobjekt gefunden.
11026	0x00002B12	WSA_QOS_EFLOWDESC	Ein ungültiger QOS-Flow-Deskriptor wurde in die Liste der Flow-Deskriptoren gefunden.
11027	0x00002B13	WSA_QOS_EPSFLOWSPEC	Im QOS-spezifischen Anbieterpuffer wurde eine ungültige oder inkonsistente Flowspec gefunden.
11028	0x00002B14	WSA_QOS_EPSFILTERSPEC	Im QOS-spezifischen Anbieterpuffer wurde ein ungültiger FILTERSPEC gefunden.
11029	0x00002B15	WSA_QOS_ESDMODEOBJ	Im QOS-spezifischen Anbieterpuffer wurde eine ungültige Formobjekt verwerfen Modus gefunden.
11030	0x00002B16	WSA_QOS_ESHAPERATEOBJ	Im QOS-spezifischen Anbieterpuffer wurde ein ungültiges Strukturierungsrateobjekt gefunden.
11031	0x00002B17	WSA_QOS_RESERVED_PETYPE	Im QOS-spezifischen Anbieterpuffer wurde ein reserviertes Richtlinielement gefunden.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
12000	0x00002EE0	ERROR_SXS_SECTION_NOT_FOUND	Der angeforderte Abschnitt war nicht anwesend in den Aktivierungskontext.
12001	0x00002EE1	ERROR_SXS_CANT_GEN_ACTCTX	Diese Anwendung konnte nicht gestartet werden, weil die Anwendungskonfiguration falsch ist. Durch eine Neuinstallation der Anwendung könnte das Problem beheben.
12002	0x00002EE2	ERROR_SXS_INVALID_ACTCTXDATA_FORMAT	Das Anwendungsformat Datenbindung ist ungültig.
12003	0x00002EE3	ERROR_SXS_ASSEMBLY_NOT_FOUND	Die referenzierte Assemblierung ist nicht auf Ihrem System installiert.
12004	0x00002EE4	ERROR_SXS_MANIFEST_FORMAT_ERROR	Die manifest-Datei beginnt nicht mit der erforderlichen Tag und Format.
12005	0x00002EE5	ERROR_SXS_MANIFEST_PARSE_ERROR	Die manifest-Datei enthält einen oder mehrere Syntaxfehler.
12006	0x00002EE6	ERROR_SXS_ACTIVATION_CONTEXT_DISABLED	Die Anwendung hat versucht, einen deaktivierten Aktivierungskontext zu aktivieren.
12007	0x00002EE7	ERROR_SXS_KEY_NOT_FOUND	Der angeforderte Nachschlüssel wurde nicht in jedem aktiven Aktivierungskontext gefunden.
12008	0x00002EE8	ERROR_SXS_VERSION_CONFLICT	Eine Komponentenversion benötigt, die durch die Anwendungskonflikte mit einer anderen bereits aktiven Komponentenversion.
12009	0x00002EE9	ERROR_SXS_WRONG_SECTION_TYPE	Der Typ angefordert, Aktivierung Kontext Abschnitt nicht die verwendeten API-Abfrage entspricht.
12010	0x00002EEA	ERROR_SXS_THREAD_QUERIES_DISABLED	Mangelnde Systemressourcen erforderte isolierten Aktivierung für den aktuellen Ausführungsthread an deaktiviert werden.
12011	0x00002EEB	ERROR_SXS_PROCESS_DEFAULT_ALREADY_SET	Beim Versuch, den Prozess-Standard-Aktivierungskontext festzulegen ist fehlgeschlagen, da der Prozess-Standard-Aktivierung-Kontext bereits festgelegt wurde.
12012	0x00002EEC	ERROR_SXS_UNKNOWN_ENCODING_GROUP	Bezeichner der Codierung angegeben wird nicht erkannt.
12013	0x00002EED	ERROR_SXS_UNKNOWN_ENCODING	Die Kodierung angefordert wird nicht erkannt.
12014	0x00002EEE	ERROR_SXS_INVALID_XML_NAMESPACE_URI	Das Manifest enthält einen Verweis auf einen ungültigen URI.
12015	0x00002EEF	ERROR_SXS_ROOT_MANIFEST_DEPENDENCY_NOT_INSTALLED	Das Anwendungsmanifest enthält einen Verweis auf eine abhängige Assembly, die nicht installiert ist.
12016	0x00002EF0	ERROR_SXS_LEAF_MANIFEST_DEPENDENCY_NOT_INSTALLED	Das Manifest für eine Assembly, die von der Anwendung verwendeten hat einen Verweis auf eine abhängige Assembly, die nicht installiert ist.
12017	0x00002EF1	ERROR_SXS_INVALID_ASSEMBLY_IDENTITY_ATTRIBUTE	Das Manifest enthält ein Attribut für die Identität der Assembly, die nicht gültig ist.
12018	0x00002EF2	ERROR_SXS_MANIFEST_MISSING_REQUIRED_DEFAULT_NAMESPACE	Das Manifest fehlt die erforderliche Standardspezifikation Namespace auf das Assembly-Element.
12019	0x00002EF3	ERROR_SXS_MANIFEST_INVALID_REQUIRED_DEFAULT_NAMESPACE	Das Manifest hat einen Standardnamespace auf Assembly-Elements angegeben, aber sein Wert ist nicht "Urn: Schemas-Microsoft-com".
12020	0x00002EF4	ERROR_SXS_PRIVATE_MANIFEST_CROSS_PATH_WITH_REPARSE_POINT	Die private Manifeste Sonde hat den Analysepunkte Punkt-assoziierten Weg gekreuzt.
12021	0x00002EF5	ERROR_SXS_DUPLICATE_DLL_NAME	Zwei oder mehr Komponenten, vom Anwendungsmanifest direkt oder indirekt referenzierte haben Dateien mit demselben Namen.
12022	0x00002EF6	ERROR_SXS_DUPLICATE_WINDOWCLASS_NAME	Zwei oder mehr Komponenten, vom Anwendungsmanifest direkt oder indirekt referenzierte haben Fensterklassen mit demselben Namen.
12023	0x00002EF7	ERROR_SXS_DUPLICATE_CLSID	Zwei oder mehr Komponenten verwiesen direkt oder indirekt vom Anwendungsmanifest haben den gleichen COM-Server CLSIDs.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
12024	0x00002EF8	ERROR_SXS_DUPLICATE_IID	Zwei oder mehr Komponenten, vom Anwendungsmanifest direkt oder indirekt referenzierte haben Proxies für die gleichen IIDs der COM-Schnittstelle.
12025	0x00002EF9	ERROR_SXS_DUPLICATE_TLBID	Zwei oder mehr Komponenten, vom Anwendungsmanifest direkt oder indirekt referenzierte haben den gleichen COM-Typ-Bibliothek TLBIDs.
12026	0x00002EFA	ERROR_SXS_DUPLICATE_PROGID	Zwei oder mehr Komponenten verwiesen direkt oder indirekt vom Anwendungsmanifest haben den gleichen COM-ProgIDs.
12027	0x00002EFB	ERROR_SXS_DUPLICATE_ASSEMBLY_NAME	Zwei oder mehr Komponenten, vom Anwendungsmanifest direkt oder indirekt referenzierte sind verschiedene Versionen derselben Komponente nicht zulässig ist.
12028	0x00002EFC	ERROR_SXS_FILE_HASH_MISMATCH	Eine Komponentendatei stimmt nicht mit den Überprüfungsinformationen im Komponentenmanifest überein.
12029	0x00002EFD	ERROR_SXS_POLICY_PARSE_ERROR	Das Politik-Manifest enthält einen oder mehrere Syntaxfehler.
12030	0x00002EFE	ERROR_SXS_XML_E_MISSINGQUOTE	Manifestverarbeitungsfehler: Ein Zeichenfolgenliteral wurde erwartet, aber keine Öffnung Anführungsstrich Zeichen wurde gefunden.
12031	0x00002EFF	ERROR_SXS_XML_E_COMMENTSyntax	Manifestverarbeitungsfehler: Falsche Syntax wurde in einem Kommentar verwendet.
12032	0x00002F00	ERROR_SXS_XML_E_BADSTARTNAMECHAR	Manifestverarbeitungsfehler: Ein Name wurde mit einem ungültigen Zeichen gestartet.
12033	0x00002F01	ERROR_SXS_XML_E_BADNAMECHAR	Manifestverarbeitungsfehler: Ein Name enthält ein ungültiges Zeichen.
12034	0x00002F02	ERROR_SXS_XML_E_BADCHARINSTRING	Manifestverarbeitungsfehler: Ein String-literal enthält ein ungültiges Zeichen.
12035	0x00002F03	ERROR_SXS_XML_E_XMLDECLSYNTAX	Manifestverarbeitungsfehler: Ungültige Syntax für eine XML-Deklaration.
12036	0x00002F04	ERROR_SXS_XML_E_BADCHARDATA	Manifestverarbeitungsfehler: Ein ungültiges Zeichen wurde im Textinhalt gefunden.
12037	0x00002F05	ERROR_SXS_XML_E_MISSINGWHITESPACE	Manifestverarbeitungsfehler: Erforderliche Leerraum fehlte.
12038	0x00002F06	ERROR_SXS_XML_E_EXPECTINGTAGEND	Manifestverarbeitungsfehler: Das Zeichen '>' wurde erwartet.
12039	0x00002F07	ERROR_SXS_XML_E_MISSINGSEMICOLON	Manifestverarbeitungsfehler: Es wurde ein Semikolon erwartet.
12040	0x00002F08	ERROR_SXS_XML_E_UNBALANCEDPAREN	Manifestverarbeitungsfehler: Unausgeglichene Klammern.
12041	0x00002F09	ERROR_SXS_XML_E_INTERNALERROR	Manifestverarbeitungsfehler: Interner Fehler.
12042	0x00002F0A	ERROR_SXS_XML_E_UNEXPECTED_WHITESPACE	Manifestverarbeitungsfehler: Leerraum ist an dieser Stelle nicht zulässig.
12043	0x00002F0B	ERROR_SXS_XML_E_INCOMPLETE_ENCODING	Manifestverarbeitungsfehler: Ende der Datei erreicht ungültigen Status für die aktuelle Codierung.
12044	0x00002F0C	ERROR_SXS_XML_E_MISSING_PAREN	Manifestverarbeitungsfehler: Fehlende Klammer.
12045	0x00002F0D	ERROR_SXS_XML_E_EXPECTINGCLOSEQUOTE	Manifestverarbeitungsfehler: Ein Einzel- oder Doppelzitat schließendes Anführungszeichen Zeichen (' oder ') fehlt.
12046	0x00002F0E	ERROR_SXS_XML_E_MULTIPLE_COLONS	Manifestverarbeitungsfehler: Mehrere Doppelpunkte dürfen nicht in einen Namen.
12047	0x00002F0F	ERROR_SXS_XML_E_INVALID_DECIMAL	Manifestverarbeitungsfehler: Ungültiges Zeichen für Dezimalziffer.
12048	0x00002F10	ERROR_SXS_XML_E_INVALID_HEXIDECIMAL	Manifestverarbeitungsfehler: Ungültiges Zeichen für die Hexadezimalziffer.
12049	0x00002F11	ERROR_SXS_XML_E_INVALID_UNICODE	Manifestverarbeitungsfehler: Ungültige Unicode-Zeichenwert für diese Plattform.
12050	0x00002F12	ERROR_SXS_XML_E_WHITESPACEORQUESTIONMARK	Manifestverarbeitungsfehler: Erwartet Leerraum oder '?'.
12051	0x00002F13	ERROR_SXS_XML_E_UNEXPECTEDENDTAG	Manifestverarbeitungsfehler: End-Tag wurde an dieser Stelle nicht erwartet.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
12052	0x00002F14	ERROR_SXS_XML_E_UNCLOSEDTAG	Manifestverarbeitungsfehler: Die folgenden Tags waren nicht geschlossen: %1.
12053	0x00002F15	ERROR_SXS_XML_E_DUPLICATEATTRIBUTE	Manifestverarbeitungsfehler: Doppeltes Attribut.
12054	0x00002F16	ERROR_SXS_XML_E_MULTIPLEROOTS	Manifestverarbeitungsfehler: Einziger Top-Level-Element darf in einem XML-Dokument.
12055	0x00002F17	ERROR_SXS_XML_E_INVALIDATROOTLEVEL	Manifestverarbeitungsfehler: Ungültig auf der obersten Ebene des Dokuments.
12056	0x00002F18	ERROR_SXS_XML_E_BADXMLDECL	Manifestverarbeitungsfehler: Ungültige XML-Deklaration.
12057	0x00002F19	ERROR_SXS_XML_E_MISSINGROOT	Manifestverarbeitungsfehler: XML-Dokument muss ein Element der obersten Ebene.
12058	0x00002F1A	ERROR_SXS_XML_E_UNEXPECTEDEOF	Manifestverarbeitungsfehler: Unerwartetes Dateende.
12059	0x00002F1B	ERROR_SXS_XML_E_BADPEREFINSUBSET	Manifestverarbeitungsfehler: Parameter-Entitäten können nicht innerhalb von Markupdeklarationen in einer internen Teilmenge verwendet werden.
12060	0x00002F1C	ERROR_SXS_XML_E_UNCLOSEDSTARTTAG	Manifestverarbeitungsfehler: Element wurde nicht geschlossen.
12061	0x00002F1D	ERROR_SXS_XML_E_UNCLOSEDENDTAG	Manifestverarbeitungsfehler: Endelement fehlte das Zeichen '>'.
12062	0x00002F1E	ERROR_SXS_XML_E_UNCLOSEDSTRING	Manifestverarbeitungsfehler: Ein Zeichenfolgenliteral wurde nicht geschlossen.
12063	0x00002F1F	ERROR_SXS_XML_E_UNCLOSEDCOMMENT	Manifestverarbeitungsfehler: Ein Kommentar wurde nicht geschlossen.
12064	0x00002F20	ERROR_SXS_XML_E_UNCLOSEDDECL	Manifestverarbeitungsfehler: Eine Erklärung wurde nicht geschlossen.
12065	0x00002F21	ERROR_SXS_XML_E_UNCLOSEDCDATA	Manifestverarbeitungsfehler: Ein CDATA-Abschnitt wurde nicht geschlossen.
12066	0x00002F22	ERROR_SXS_XML_E_RESERVEDNAMESPACE	Manifestverarbeitungsfehler: Das Namespacepräfix darf nicht mit der reservierten Zeichenkette "Xml" beginnen.
12067	0x00002F23	ERROR_SXS_XML_E_INVALIDENCODING	Manifestverarbeitungsfehler: System der angegebenen Codierung unterstützt nicht.
12068	0x00002F24	ERROR_SXS_XML_E_INVALIDSWITCH	Manifestverarbeitungsfehler: Wechsel von der aktuellen Codierung auf angegebene Codierung nicht unterstützt.
12069	0x00002F25	ERROR_SXS_XML_E_BADXMLCASE	Manifestverarbeitungsfehler: Der Name "Xml" ist reserviert und muss klein geschrieben sein.
12070	0x00002F26	ERROR_SXS_XML_E_INVALID_STANDALONE	Manifestverarbeitungsfehler: Das Standalone-Attribut muss den Wert "Ja" oder "Nein".
12071	0x00002F27	ERROR_SXS_XML_E_UNEXPECTED_STANDALONE	Manifestverarbeitungsfehler: Das Standalone-Attribut kann nicht in externe Entitäten verwendet werden.
12072	0x00002F28	ERROR_SXS_XML_E_INVALID_VERSION	Manifestverarbeitungsfehler: Ungültige Versionsnummer.
12073	0x00002F29	ERROR_SXS_XML_E_MISSINGEQUALS	Manifestverarbeitungsfehler: Fehlendes Gleichheitszeichen zwischen und Attributwert.
13000	0x000032C8	ERROR_IPSEC_QM_POLICY_EXISTS	Die angegebene Schnellmodusrichtlinie ist bereits vorhanden.
13001	0x000032C9	ERROR_IPSEC_QM_POLICY_NOT_FOUND	Die angegebene Schnellmodusrichtlinie wurde nicht gefunden.
13002	0x000032CA	ERROR_IPSEC_QM_POLICY_IN_USE	Die angegebene Schnellmodusrichtlinie wird verwendet.
13003	0x000032CB	ERROR_IPSEC_MM_POLICY_EXISTS	Die angegebene Hauptmodusrichtlinie ist bereits vorhanden.
13004	0x000032CC	ERROR_IPSEC_MM_POLICY_NOT_FOUND	Die angegebene Hauptmodusrichtlinie wurde nicht gefunden.
13005	0x000032CD	ERROR_IPSEC_MM_POLICY_IN_USE	Die angegebene Hauptmodusrichtlinie wird verwendet.
13006	0x000032CE	ERROR_IPSEC_MM_FILTER_EXISTS	Der angegebene Hauptmodusfilter ist bereits vorhanden.
13007	0x000032CF	ERROR_IPSEC_MM_FILTER_NOT_FOUND	Die angegebene Hauptmodusfilter wurde nicht gefunden.
13008	0x000032D0	ERROR_IPSEC_TRANSPORT_FILTER_EXISTS	Der angegebenen Transport-Mode-Filter ist bereits vorhanden.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
13009	0x000032D1	ERROR_IPSEC_TRANSPORT_FILTER_NOT_FOUND	Der angegebenen Transport-Mode-Filter ist nicht vorhanden.
13010	0x000032D2	ERROR_IPSEC_MM_AUTH_EXISTS	Die angegebene Hauptmodus-Authentifizierungsliste ist vorhanden.
13011	0x000032D3	ERROR_IPSEC_MM_AUTH_NOT_FOUND	Die angegebene Hauptmodus-Authentifizierungsliste wurde nicht gefunden.
13012	0x000032D4	ERROR_IPSEC_MM_AUTH_IN_USE	Die angegebene Schnellmodusrichtlinie wird verwendet.
13013	0x000032D5	ERROR_IPSEC_DEFAULT_MM_POLICY_NOT_FOUND	Die angegebene Hauptmodusrichtlinie wurde nicht gefunden.
13014	0x000032D6	ERROR_IPSEC_DEFAULT_MM_AUTH_NOT_FOUND	Die angegebene Schnellmodusrichtlinie wurde nicht gefunden.
13015	0x000032D7	ERROR_IPSEC_DEFAULT_QM_POLICY_NOT_FOUND	Die manifest-Datei enthält einen oder mehrere Syntaxfehler.
13016	0x000032D8	ERROR_IPSEC_TUNNEL_FILTER_EXISTS	Die Anwendung hat versucht, einen deaktivierten Aktivierungskontext zu aktivieren.
13017	0x000032D9	ERROR_IPSEC_TUNNEL_FILTER_NOT_FOUND	Der angeforderte Nachschlüssel wurde nicht in jedem aktiven Aktivierungskontext gefunden.
13018	0x000032DA	ERROR_IPSEC_MM_FILTER_PENDING_DELETION	Der Hauptmodus-Filter steht zum Löschen.
13019	0x000032DB	ERROR_IPSEC_TRANSPORT_FILTER_PENDING_DELETION	Der Transportfilter steht zum Löschen.
13020	0x000032DC	ERROR_IPSEC_TUNNEL_FILTER_PENDING_DELETION	Der Tunnel-Filter steht zum Löschen.
13021	0x000032DD	ERROR_IPSEC_MM_POLICY_PENDING_DELETION	Die Hauptmodus-Politik steht zum Löschen.
13022	0x000032DE	ERROR_IPSEC_MM_AUTH_PENDING_DELETION	Das Hauptmodus-Authentifizierung-Bundle steht zum Löschen.
13023	0x000032DF	ERROR_IPSEC_QM_POLICY_PENDING_DELETION	Die Schnellmodusrichtlinie steht zum Löschen.
13801	0x000035E9	ERROR_IPSEC_IKE_AUTH_FAIL	IKE-Authentifizierung-Anmeldeinformationen sind inakzeptabel.
13802	0x000035EA	ERROR_IPSEC_IKE_ATTRIB_FAIL	IKE-Sicherheitsattribute sind inakzeptabel.
13803	0x000035EB	ERROR_IPSEC_IKE_NEGOTIATION_PENDING	IKE-Aushandlung im Gange.
13804	0x000035EC	ERROR_IPSEC_IKE_GENERAL_PROCESSING_ERROR	Allgemeine Verarbeitungsfehler.
13805	0x000035ED	ERROR_IPSEC_IKE_TIMED_OUT	Verhandlung Zeitüberschreitung.
13806	0x000035EE	ERROR_IPSEC_IKE_NO_CERT	IKE wurde kein gültiges Zertifikat gefunden.
13807	0x000035EF	ERROR_IPSEC_IKE_SA_DELETED	IKE-Sicherheitszuordnung gelöscht von Peer vor Gründung abgeschlossen.
13808	0x000035F0	ERROR_IPSEC_IKE_SA_REAPED	IKE-Sicherheitszuordnung gelöscht vor Gründung abgeschlossen.
13809	0x000035F1	ERROR_IPSEC_IKE_MM_ACQUIRE_DROP	Aushandlungsanforderung saß in der Warteschlange zu lang.
13810	0x000035F2	ERROR_IPSEC_IKE_QM_ACQUIRE_DROP	Aushandlungsanforderung saß in der Warteschlange zu lang.
13811	0x000035F3	ERROR_IPSEC_IKE_QUEUE_DROP_MM	Aushandlungsanforderung saß in der Warteschlange zu lang.
13812	0x000035F4	ERROR_IPSEC_IKE_QUEUE_DROP_NO_MM	Aushandlungsanforderung saß in der Warteschlange zu lang.
13813	0x000035F5	ERROR_IPSEC_IKE_DROP_NO_RESPONSE	Keine Antwort von Peer.
13814	0x000035F6	ERROR_IPSEC_IKE_MM_DELAY_DROP	Verhandlung dauerte zu lange.
13815	0x000035F7	ERROR_IPSEC_IKE_QM_DELAY_DROP	Verhandlung dauerte zu lange.
13816	0x000035F8	ERROR_IPSEC_IKE_ERROR	Unbekannter Fehler aufgetreten.
13817	0x000035F9	ERROR_IPSEC_IKE_CRL_FAILED	Zertifikat Widerruf überprüfen konnte.
13818	0x000035FA	ERROR_IPSEC_IKE_INVALID_KEY_USAGE	Ungültiges Zertifikat Schlüsselverwendung.
13819	0x000035FB	ERROR_IPSEC_IKE_INVALID_CERT_TYPE	Ungültiges Zertifikat-Typ.
13820	0x000035FC	ERROR_IPSEC_IKE_NO_PRIVATE_KEY	Kein Zertifikat zugeordneten privaten Schlüssel.
13822	0x000035FE	ERROR_IPSEC_IKE_DH_FAIL	Fehler bei Diffie-Helman-Berechnung.
13824	0x00003600	ERROR_IPSEC_IKE_INVALID_HEADER	Ungültiger Header.
13825	0x00003601	ERROR_IPSEC_IKE_NO_POLICY	Keine Richtlinie konfiguriert.
13826	0x00003602	ERROR_IPSEC_IKE_INVALID_SIGNATURE	Fehler beim Überprüfen der Signatur.

Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
13827	0x00003603	ERROR_IPSEC_IKE_KERBEROS_ERROR	Fehler beim Authentifizieren mit Kerberos.
13828	0x00003604	ERROR_IPSEC_IKE_NO_PUBLIC_KEY	Zertifikat des Peers haben keinen öffentlichen Schlüssel.
13829	0x00003605	ERROR_IPSEC_IKE_PROCESS_ERR	Fehler Verarbeitung Fehler Nutzlast.
13830	0x00003606	ERROR_IPSEC_IKE_PROCESS_ERR_SA	Fehler Verarbeitung SA Nutzlast.
13831	0x00003607	ERROR_IPSEC_IKE_PROCESS_ERR_PROP	Fehler beim Verarbeiten Vorschlag Nutzlast.
13832	0x00003608	ERROR_IPSEC_IKE_PROCESS_ERR_TRANS	Fehler beim Verarbeiten Transform Nutzlast.
13833	0x00003609	ERROR_IPSEC_IKE_PROCESS_ERR_KE	Fehler Verarbeitung KE Nutzlast.
13834	0x0000360A	ERROR_IPSEC_IKE_PROCESS_ERR_ID	Fehler-Verarbeitung-ID-Nutzlast.
13835	0x0000360B	ERROR_IPSEC_IKE_PROCESS_ERR_CERT	Fehler beim Verarbeiten Cert Nutzlast.
13836	0x0000360C	ERROR_IPSEC_IKE_PROCESS_ERR_CERT_REQ	Fehler beim Verarbeiten Zertifikatanforderung Nutzlast.
13837	0x0000360D	ERROR_IPSEC_IKE_PROCESS_ERR_HASH	Fehler beim Verarbeiten Hash Nutzlast.
13838	0x0000360E	ERROR_IPSEC_IKE_PROCESS_ERR_SIG	Fehler beim Verarbeiten Signatur Nutzlast.
13839	0x0000360F	ERROR_IPSEC_IKE_PROCESS_ERR_NONCE	Fehler beim Verarbeiten Nonce Nutzlast.
13840	0x00003610	ERROR_IPSEC_IKE_PROCESS_ERR_NOTIFY	Fehler beim Verarbeiten Notify-Nutzlast.
13841	0x00003611	ERROR_IPSEC_IKE_PROCESS_ERR_DELETE	Fehler beim Verarbeiten der Nutzlast löschen.
13842	0x00003612	ERROR_IPSEC_IKE_PROCESS_ERR_VENDOR	Fehler beim Verarbeiten VendorId Nutzlast.
13843	0x00003613	ERROR_IPSEC_IKE_INVALID_PAYLOAD	Ungültige Nutzlast empfangen.
13844	0x00003614	ERROR_IPSEC_IKE_LOAD_SOFT_SA	Soft-Sicherheitszuordnung geladen.
13845	0x00003615	ERROR_IPSEC_IKE_SOFT_SA_TORN_DOWN	Soft-Sicherheitszuordnung abgerissen.
13846	0x00003616	ERROR_IPSEC_IKE_INVALID_COOKIE	Ungültige Cookie empfangen...
13847	0x00003617	ERROR_IPSEC_IKE_NO_PEER_CERT	Peer konnte gültiges Zertifikat senden.
13848	0x00003618	ERROR_IPSEC_IKE_PEER_CRL_FAILED	Zertifizierung Widerruf Kontrolle der Peerzertifikats ist fehlgeschlagen.
13849	0x00003619	ERROR_IPSEC_IKE_POLICY_CHANGE	Neue Richtlinie für ungültig erklärt SAs mit alten Politik gebildet.
13850	0x0000361A	ERROR_IPSEC_IKE_NO_MM_POLICY	Es gibt keine verfügbaren Main Mode IKE-Richtlinie.
13851	0x0000361B	ERROR_IPSEC_IKE_NOTCBPRIV	Fehler beim TCB-Berechtigung aktiviert.
13852	0x0000361C	ERROR_IPSEC_IKE_SECLoadFAIL	Fehler beim Laden von Sicherheit.DLL.
13853	0x0000361D	ERROR_IPSEC_IKE_FAILSSPINIT	Sicherheit Tabelle Versand Funktionsadresse von SSPI erhalten konnte.
13854	0x0000361E	ERROR_IPSEC_IKE_FAILQUERYSSP	Fehler bei der Abfrage von Kerberos-Paket um max token-Größe zu erhalten.
13855	0x0000361F	ERROR_IPSEC_IKE_SRVACQFAIL	Fehler beim Abrufen der Kerberos-Serveranmeldeinformationen für ISAKMP/ ERROR_IPSEC_IKE Dienst. Kerberos-Authentifizierung funktioniert nicht. Der wahrscheinlichste Grund dafür ist mangelnde Domänenmitgliedschaft. Dies ist normal, wenn Ihr Computer Mitglied einer Arbeitsgruppe ist.
13856	0x00003620	ERROR_IPSEC_IKE_SRVQUERYCRED	Fehler beim Bestimmen der SSPI-Prinzipalname für ISAKMP/ERROR_IPSEC_IKE Dienst (QueryCredentialsAttributes).
13857	0x00003621	ERROR_IPSEC_IKE_GETSPIFAIL	Konnte keine neue SPI für die eingehende SA vom Ipsec-Treiber zu erhalten. Die häufigste Ursache hierfür ist, dass der Treiber nicht den richtigen Filter. Überprüfen Sie Ihre Politik um die Filter zu überprüfen.
13858	0x00003622	ERROR_IPSEC_IKE_INVALID_FILTER	Da Filter ist ungültig.
13859	0x00003623	ERROR_IPSEC_IKE_OUT_OF_MEMORY	Speicherzuordnung fehlgeschlagen.
13860	0x00003624	ERROR_IPSEC_IKE_ADD_UPDATE_KEY_FAILED	Fehler beim Hinzufügen von Security Association zur IPsec-Treiber. Die häufigste Ursache hierfür ist, wenn die IKE-Aushandlung zu lange dauerte. Wenn das Problem weiterhin auftritt, reduzieren Sie die Auslastung auf dem fehlerhaften Computer.
13861	0x00003625	ERROR_IPSEC_IKE_INVALID_POLICY	Ungültige Richtlinien.
13862	0x00003626	ERROR_IPSEC_IKE_UNKNOWN_DOI	Ungültige DOI.
13863	0x00003627	ERROR_IPSEC_IKE_INVALID_SITUATION	Ungültige Lage.
13864	0x00003628	ERROR_IPSEC_IKE_DH_FAILURE	Diffie-Hellman-Fehler.
13865	0x00003629	ERROR_IPSEC_IKE_INVALID_GROUP	Ungültige Diffie-Hellman-Gruppe.



Fehler			Beschreibung
Dezimal	Hexadezimal	Name	
13866	0x0000362A	ERROR_IPSEC_IKE_ENCRYPT	Fehler verschlüsseln Nutzlast.
13867	0x0000362B	ERROR_IPSEC_IKE_DECRYPT	Fehler entschlüsseln Nutzlast.
13868	0x0000362C	ERROR_IPSEC_IKE_POLICY_MATCH	Fehler der Politik überein.
13869	0x0000362D	ERROR_IPSEC_IKE_UNSUPPORTED_ID	Nicht unterstützte ID.
13870	0x0000362E	ERROR_IPSEC_IKE_INVALID_HASH	Hashüberprüfung ist fehlgeschlagen.
13871	0x0000362F	ERROR_IPSEC_IKE_INVALID_HASH_ALG	Ungültige Hash-Algorithmus.
13872	0x00003630	ERROR_IPSEC_IKE_INVALID_HASH_SIZE	Ungültige Hash-Größe.
13873	0x00003631	ERROR_IPSEC_IKE_INVALID_ENCRYPT_ALG	Ungültige Verschlüsselungsalgorithmus.
13874	0x00003632	ERROR_IPSEC_IKE_INVALID_AUTH_ALG	Ungültiger Authentifizierungsalgorithmus.
13875	0x00003633	ERROR_IPSEC_IKE_INVALID_SIG	Ungültiges Zertifikat-Signatur.
13876	0x00003634	ERROR_IPSEC_IKE_LOAD_FAILED	Laden fehlgeschlagen ist.
13877	0x00003635	ERROR_IPSEC_IKE_RPC_DELETE	Über RPC-Aufruf gelöscht.
13878	0x00003636	ERROR_IPSEC_IKE_BENIGN_REINIT	Vorübergehender Zustand erstellt, um Reinit auszuführen. Dies ist keine echte Fehler.
13879	0x00003637	ERROR_IPSEC_IKE_INVALID_RESPONDER_LIFETIME_NOTIFY	Den Lebenszeitwert erhielt der Responder Lifetime Benachrichtigen ist unter dem Mindestwert von Windows 2000 konfiguriert. Bitte korrigieren Sie die Richtlinie auf dem Peer-Computer.
13880	0x00003638	ERROR_IPSEC_IKE_QM_LIMIT_REAP	SA geerntet weil QM Grenze erreicht wurde.
13881	0x00003639	ERROR_IPSEC_IKE_INVALID_CERT_KEYLEN	Schlüssellänge im Zertifikat ist zu klein für die konfigurierten Sicherheitsanforderungen.
13882	0x0000363A	ERROR_IPSEC_IKE_MM_LIMIT	Maximale Anzahl von etablierten MM SAs, peer überschritten.
13883	0x0000363B	ERROR_IPSEC_IKE_NEGOTIATION_DISABLED	IKE erhielt eine Politik, die Aushandlung deaktiviert.
13884	0x0000363C	ERROR_IPSEC_IKE_NEG_STATUS_END	

## 7.4.2 Fehlersuche/Diagnose

### Allgemeine Überprüfungen:

- Überprüfen Sie die in dieser Dokumentation beschriebenen Hardware- und Softwareanforderungen (TwinCAT-Version, CE Image-Version usw.);
- Überprüfen Sie die Installationshinweise (z.B. Installation der CAB-Files auf einem CE System);
- Bei Verbindungsproblemen kann der PING-Befehl dazu benutzt werden, um festzustellen ob der Kommunikationspartner über die Netzwerkverbindung erreichbar ist. Wenn dies nicht der Fall ist überprüfen Sie die Netzwerkkonfiguration und die [Firewall-Einstellungen](#) [[781](#)];
- Eine komplette Aufzeichnung der Netzwerkkommunikation kann mit Sniffer-Tools wie Wireshark durchgeführt werden. Die Aufnahme kann dann vom Beckhoff-Supportpersonal analysiert werden;
- Überprüfen Sie die Konfiguration der Datenpunkte (Typ, Adresse des Informationsobjektes usw.);
- Überprüfen Sie ob der andere Kommunikationspartner einen Fehlercode ausgibt;
- Aktivieren Sie die [Debugausgaben](#) [[778](#)] beim Aufbau und Abbau der Verbindung und/oder der ASDU Daten. Öffnen Sie den TwinCAT System Manager und aktivieren das LogView-Fenster. Prüfen Sie die Debugausgaben.
- Überprüfen Sie die Verwendung des FB\_SocketCloseAll Funktionsbausteins und des LISTEN\_MODE\_CLOSEALL-Parameters wenn Sie über mehrere TCP/IP Verbindungen (Server/Clients) in einem Laufzeitsystem kommunizieren.  
Bei mehreren Verbindungen aktivieren Sie nur eine Instanz des FB\_SocketCloseAll-Funktionsbausteins einmalig im Initialisierungsschritt beim Programmstart.  
Der LISTEN\_MODE\_CLOSEALL-Parameter darf in diesem Fall nicht mehr verwendet werden;

### High-Level Interface:

- Speziell bei der Zentralstation:
  - Überprüfen Sie ob der [FB\\_IEC870\\_5\\_104Master](#) [[389](#)]-Funktionsbaustein einen [Fehlercode/Fehlerquelle](#) [[708](#)] ausgibt;

- Vergleichen/überprüfen Sie die Kompatibilitätsliste der Zentralstation [▶ 68] mit der Kompatibilitätsliste der Unterstation;
- Überprüfen Sie ob die Netzwerkadresse, Portnummer die Sie an den FB\_IEC870\_5\_104Master-Funktionsbaustein übergeben richtig sind;
- Überprüfen Sie die am FB\_IEC870\_5\_104Master-Funktionsbaustein eingestellten Protokollparameter [▶ 549] (iK, iW, t0, t1, t2, t3, APDULength, usw.). Vergleichen Sie die Protokollparameter mit den Parametern in der Unterstation;
- Überprüfen Sie die am FB\_IEC870\_5\_104Master-Funktionsbaustein eingestellten Systemparameter [▶ 487] (ASDU-Adresse, Länge der ASDU-Adresse, Länge der Informationsobjekt-Adresse, Länge der Übertragungsursache COT, usw. ). Vergleichen Sie die Systemparameter mit den Parametern in der Unterstation;
- Überprüfen Sie die am FB\_IEC870\_5\_104MasterFunktionsbaustein eingestellten Parameter für die zyklische Datenerfassung [▶ 491] (Akquisition) (Initialisierungssequenz, zyklische Generalabfrage, zyklische Zählerabfrage, zyklische Testkommandos, usw. );
- Speziell bei der Unterstation:
  - Überprüfen Sie ob der FB\_IEC870\_5\_104Slave [▶ 392]-Funktionsbaustein einen Fehlercode/ Fehlerquelle [▶ 708] ausgibt;
  - Überprüfen Sie die Kompatibilitätsliste der Unterstation [▶ 86] mit der Kompatibilitätsliste der Zentralstation;
  - Überprüfen Sie ob die Eingangsparameter, die Sie an die F\_CreateServerHnd Funktion übergeben richtig sind (Netzwerkadresse, Portnummer, usw.);
  - Überprüfen Sie die am FB\_IEC870\_5\_104Slave-Funktionsbaustein eingestellten Protokollparameter [▶ 549] (iK, iW, t0, t1, t2, t3, APDULength, usw.). Vergleichen Sie die Protokollparameter mit den Parametern in der Leitstation;
  - Überprüfen Sie die am FB\_IEC870\_5\_104Slave-Funktionsbaustein eingestellten Systemparameter [▶ 487] (ASDU-Adresse, Länge der ASDU-Adresse, Länge der Informationsobjekt-Adresse, Länge der Übertragungsursache COT, usw. ). Vergleichen Sie die Systemparameter mit den Parametern in der Leitstation;

#### Low-Level Interface:

- Überprüfen Sie ob der FB\_IEC870\_5\_104TProtocol [▶ 411]-Funktionsbaustein einen Fehlercode [▶ 708] ausgibt;
- Vergleichen/überprüfen Sie die Kompatibilitätsliste der Zentralstation mit der Kompatibilitätsliste der Unterstation;
- Überprüfen Sie ob die Netzwerkadresse und Portnummer die Sie an die F\_CreateServerHnd Funktion bzw. an den FB\_ClientServerConnection oder FB\_ServerClientConnection Funktionsbaustein übergeben richtig sind;
- Überprüfen Sie die am FB\_IEC870\_5\_104TProtocol-Funktionsbaustein eingestellten Protokollparameter [▶ 549] (iK, iW, t0, t1, t2, t3, APDULength, usw.). Vergleichen Sie die Protokollparameter mit den Parametern des Kommunikationspartners;
- Überprüfen Sie die am TX/RX-Datenpuffer (Instanz von ST\_IEC870\_5\_101TBuffer [▶ 407]) konfigurierten Adresslängen: Länge der ASDU-Adresse, Länge der Informationsobjekt-Adresse, Länge der Übertragungsursache, max. ASDU-Länge. Vergleichen Sie die Adresslängen mit der Konfiguration des Kommunikationspartners;

### 7.4.3 Debugmeldungen in TwinCAT->Error List

Debugmeldungen, die in TwinCAT XAE->Error List geschrieben werden erleichtern eine Fehlersuche im System. Um die aktivierten Debugmeldungen zu sehen starten Sie TwinCAT XAE und wählen im Menü: View->Other Windows->Error List. Die nachfolgende Grafik zeigt eine Debugausgabe. Die unterschiedlichen Typen der Meldungen wurden mit Zahlen markiert.

Error List	
3 Errors	0 Warnings
447 Messages Clear	
Description	
28.10.2013 16:24:14 993 ms   'Standard' (350): Tc2_IEC60870_5_10x.Library::FB_IEC870_5_104Master[asduAddr = 7]::INIT => COLDSTART	4
28.10.2013 16:24:15 3 ms   'Standard' (350): Tc2_IEC60870_5_10x.Library::FB_IEC870_5_104Master[asduAddr = 7]::COLDSTART => WARMSTART	
28.10.2013 16:24:15 13 ms   'Standard' (350): Tc2_IEC60870_5_10x.Library::FB_IEC870_5_104Master[asduAddr = 7]::WARMSTART => SETTIME START	
28.10.2013 16:24:15 103 ms   'Standard' (350): Tc2_IEC60870_5_10x.Library::FB_IEC870_5_104Master[asduAddr = 7]::SETTIME WAIT => ACTIVE OPEN	
28.10.2013 16:24:15 113 ms   'Standard' (350): Tc2_IEC60870_5_10x.Library::TCP/IP Link[0]::DISCONNECTED => ACTIVE OPEN::Remote:172.17.61.5:2404	
28.10.2013 16:24:16 143 ms   'Standard' (350): Tc2_Tcplp.library::FB_SocketConnect(CLIENT), Handle:0x0 Remote:172.17.61.5:2404, error:0x8007274D	1
28.10.2013 16:24:16 143 ms   'Standard' (350): Tc2_IEC60870_5_10x.Library::FB_ClientServerConnection( Remote:172.17.61.5:2404 ) error: 0x8007274D	5
28.10.2013 16:24:16 143 ms   'Standard' (350): IEC60870-5-104 master error:0x8007274D	6
28.10.2013 16:24:45 243 ms   'Standard' (350): Tc2_Tcplp.library::FB_SocketConnect(CLIENT), Handle:0x1010002 Local:172.17.60.233:61282 Remote:172.17.61.5:2404, error:0x00000000	1
28.10.2013 16:24:45 253 ms   'Standard' (350): Tc2_IEC60870_5_10x.Library::TCP/IP Link[1]::ACTIVE OPEN => ESTABLISHED::Handle:0x1010002 Local:172.17.60.233:61282 Remote:172.17.61.5:2404	4
28.10.2013 16:24:45 283 ms   'Standard' (350): Tc2_IEC60870_5_10x.Library::TCP/IP Link[1]<=68 04 07 00 00 00	3
28.10.2013 16:24:45 873 ms   'Standard' (350): Tc2_IEC60870_5_10x.Library::TCP/IP Link[1]>=68 04 0B 00 00 00	
28.10.2013 16:24:45 873 ms   'Standard' (350): Tc2_IEC60870_5_10x.Library::TCP/IP Link[1]::STOPDT => STARTDT::Handle:0x1010002 Local:172.17.60.233:61282 Remote:172.17.61.5:2404	4
28.10.2013 16:24:45 903 ms   'Standard' (350): Tc2_IEC60870_5_10x.Library::TCP/IP Link[1]>=68 0E 00 00 00 00 48 01 04 00 07 00 00 00 00 00	3
28.10.2013 16:24:45 903 ms   'Standard' (350): RX FIFO<=46 01 04 00 07 00 00 00 00 00	2
28.10.2013 16:24:45 903 ms   'Standard' (350): RX FIFO=>46 01 04 00 07 00 00 00 00 00	
28.10.2013 16:24:45 923 ms   'Standard' (350): Class1 TX FIFO<=6B 01 06 01 07 00 00 00 00 00 08 B3 18 10 3C 0A 0D	
28.10.2013 16:24:45 923 ms   'Standard' (350): Tc2_IEC60870_5_10x.Library::TCP/IP Link[1]<=68 16 00 00 02 00 6B 01 06 01 07 00 00 00 00 00 08 B3 18 10 3C 0A 0D	3
28.10.2013 16:24:46 53 ms   'Standard' (350): Tc2_IEC60870_5_10x.Library::TCP/IP Link[1]>=68 16 02 00 02 00 6B 01 07 01 07 00 00 00 00 00 08 B3 18 10 3C 0A 0D	
28.10.2013 16:24:46 53 ms   'Standard' (350): RX FIFO<=6B 01 07 01 07 00 00 00 00 00 08 B3 18 10 3C 0A 0D	2
28.10.2013 16:24:46 53 ms   'Standard' (350): Class1 TX FIFO=>6B 01 06 01 07 00 00 00 00 00 08 B3 18 10 3C 0A 0D	
28.10.2013 16:24:46 53 ms   'Standard' (350): RX FIFO=>6B 01 07 01 07 00 00 00 00 00 08 B3 18 10 3C 0A 0D	
28.10.2013 16:24:46 133 ms   'Standard' (350): Class1 TX FIFO<=6B 01 06 01 07 00 00 00 00 01 00 DA B3 18 10 3C 0A 0D	
28.10.2013 16:24:46 133 ms   'Standard' (350): Tc2_IEC60870_5_10x.Library::TCP/IP Link[1]<=68 16 02 00 04 00 6B 01 06 01 07 00 00 00 00 01 00 DA B3 18 10 3C 0A 0D	
28.10.2013 16:24:46 263 ms   'Standard' (350): Tc2_IEC60870_5_10x.Library::TCP/IP Link[1]>=68 16 04 00 04 00 6B 01 07 01 07 00 00 00 00 01 00 DA B3 18 10 3C 0A 0D	

Die Meldungen werden auf unterschiedliche Weise (durch unterschiedliche Parameter) aktiviert, je nachdem ob Sie die "Low-Level" oder "High-Level"-Schnittstelle verwenden. Unten folgt die Auflistung der unterschiedlichen Parameternamen.

1. Debugmeldungen, die beim Aufbau oder Abbau der TCP/IP-Verbindung geloggt werden:

Name	AN	AUS	Unterstation		Zentralstation	
			Low-Level	High-Level	Low-Level	High-Level
F_CreateServerHnd	CONNECT_M ODE_ENABL EDBG	0	Ja	Ja	-	-
FB_ClientServerConnection	CONNECT_M ODE_ENABL EDBG	0	-	-	Ja	-
FB_IEC870_5_104Master [> 389].sysPara.dbgMode	IEC870_DEB UGMODE_LI NKERROR	IEC870_DEB UGMODE_DI SABLED	-	-	-	Ja

2. Hexadezimale Ausgabe der ASDUs (ohne Link-Layer-Control-Header). Pro Zeile werden 32-ASDU-Datenbytes als hexadezimale Zahlen ausgegeben. Längere ASDUs werden auf mehrere Zeilen verteilt:

Name	AN	AUS	Unterstation		Zentralstation	
			Low-Level	High-Level	Low-Level	High-Level
ST_IEC870_5_101TBuffer [> 481].eDbg	eIEC870_FIF O_DBG_ALL	eIEC870_FIF O_DBG_OFF	Ja	-	Ja	-
FB_IEC870_5_104Slave [> 392].sysPara.dbgMode	IEC870_DEB UGMODE_AS DU	IEC870_DEB UGMODE_DI SABLED	-	Ja	-	-
FB_IEC870_5_104Master [> 389].sysPara.dbgMode	IEC870_DEB UGMODE_AS DU	IEC870_DEB UGMODE_DI SABLED	-	-	-	Ja

3. Hexadezimale Ausgabe der APDUs (TCP/IP Telegramme). Pro Zeile werden 32-APDU-Datenbytes als hexadezimale Zahlen ausgegeben. Längere APDUs werden auf mehrere Zeilen verteilt:

Name	AN	AUS	Unterstation		Zentralstation	
			Low-Level	High-Level	Low-Level	High-Level
FB IEC870 5 1 04TProtocol [▶ 411].bOutD bg	TRUE	FALSE	Ja	-	Ja	-
FB IEC870 5 1 04Slave [▶ 392].sysPar a.dbgMode	IEC870_DEB UGMODE_LI NKLAYER	IEC870_DEB UGMODE_DI SABLED	-	Ja	-	-
FB IEC870 5 1 04Master [▶ 389].sysPar a.dbgMode	IEC870_DEB UGMODE_LI NKLAYER	IEC870_DEB UGMODE_DI SABLED	-	-	-	Ja-

4. Stationsstatus-Meldungen:

Name	AN	AUS	Unterstation		Zentralstation	
			Low-Level	High-Level	Low-Level	High-Level
FB IEC870 5 1 04Slave [▶ 392].sysPar a.dbgMode	IEC870_DEB UGMODE_DE VSTATE	IEC870_DEB UGMODE_DI SABLED	-	Ja	-	-
FB IEC870 5 1 04Master [▶ 389].sysPar a.dbgMode	IEC870_DEB UGMODE_DE VSTATE	IEC870_DEB UGMODE_DI SABLED	-	-	-	Ja

5: Ausgabe der Link-Layer-Fehler als Meldungen:

Name	AN	AUS	Unterstation		Zentralstation	
			Low-Level	High-Level	Low-Level	High-Level
FB IEC870 5 1 04Slave [▶ 392].sysPar a.dbgMode	IEC870_DEB UGMODE_LI NKERROR	IEC870_DEB UGMODE_DI SABLED	-	Ja	-	-
FB IEC870 5 1 04Master [▶ 389].sysPar a.dbgMode	IEC870_DEB UGMODE_LI NKERROR	IEC870_DEB UGMODE_DI SABLED	-	-	-	Ja

6. Meldungen, die durch die SPS-Applikation (Beispielprojekt) mit Hilfe der ADSLOGSTR, ... Funktionen geloggt werden.

Weitere Diagnosetools:

- TwinCAT ADS Monitor;
- Netzwerkmonitor;
- Wireshark;
- Ethereal;
- Diverse Protokoll-Test-Suit-Produkte;

### 7.4.4 Firewall Einstellungen -104 Protokoll

IEC 60870-5-104 verwendet während der Kommunikation das TCP/IP als Transportprotokoll. Es ist daher darauf zu achten, dass der entsprechende TCP-Port bei Benutzung einer Firewall frei geschaltet wird. Die untere Tabelle listet Standard Ports auf, die bei der Benutzung einer Firewall zu berücksichtigen sind.

Beschreibung	Typ	Protokoll	Port
IEC 60870-5-104 telecontrol protocol	Protocol	TCP	2404

Die Konfiguration der Windows Firewall wird über den entsprechenden Dialog in der Systemsteuerung vorgenommen. Weitere Informationen zur Konfiguration finden Sie in der Windows bzw. Firewall Dokumentation.

## 7.5 Glossar

Begriff	Beschreibung
Unterstation, Slave, Server	Synonyme für eine untergeordnete Station (sie wird überwacht)
Zentralstation, Leitstation, Master, Client	Synonyme für eine übergeordnete Station (Leitstation, sie überwacht andere Stationen)
Steuerungsrichtung	Datenübertragungsrichtung von der Zentralstation zur Unterstation
Überwachungsrichtung	Datenübertragungsrichtung von der Unterstation zur Zentralstation
Applikationsobjekte	IEC-Informationsobjekte in der TwinCAT SPS-Applikation (Single Points, Double Points, Measured Values, Short Floating Point Values usw. )
APDU	Protokolldateneinheit der Anwendungsschicht (application protocol data unit)
APCI	Protokolsteuerinformation der Anwendungsschicht (application protocol control information)
ASDU	Dienstdateneinheit der Anwendungsschicht (application service data unit)
IOA, Adresse des Informationsobjekts	Adresse des Single Points, Double Points usw. (information object address)
Primärstation	Die Primärstation sendet Befehle (Anfragen) an die Sekundärstation und steuert/kontrolliert so die Datenübertragung der Sekundärstation.
Sekundärstation	Die Sekundärstation antwortet auf die Anfragen der Primärstation.
Kombinierte Station	Kombinierte Stationen können die Rolle der Primärstation und Sekundärstation annehmen (symetrische Übertragung, balanced mode).
Symetrische Übertragung (balanced mode)	Beide Stationen können als Primärstation oder Sekundärstation agieren und die Datenübertragung initialisieren.
Unsymetrische Übertragung (unbalanced mode)	Die Datenübertragung wird immer durch die Primärstation initiiert und gesteuert. Die Zentralstation agiert immer als Primärstation und die Unterstation als Sekundärstation.
CAA	Gemeinsame ASDU-Adresse.
COT	Übertragungsursache.

## 7.6 FAQ - Häufig gestellte Fragen und Antworten

In diesem Bereich werden häufig gestellte Fragen beantwortet, um Ihnen die Arbeit mit der TwinCAT Fernwirktechnik zu erleichtern.

Wenn Sie noch weitere Fragen haben, kontaktieren Sie bitte unseren Support (-157)

Wie viele Datenpunkte kann man über TCP mit dem IEC 60870-5-104 Protokoll kommunizieren? [► 782]

### **? Wie viele Datenpunkte kann man über TCP mit dem IEC 60870-5-104 Protokoll kommunizieren?**

! Diese Frage ist nicht pauschal zu beantworten. Beckhoff integriert alle Kommunikationsprotokolle der IEC 60870-5 in die PC-basierte Steuerungstechnik. Daher ist die Anzahl der möglichen Datenpunkte abhängig von den Systemressourcen und der eingestellten Zykluszeit. Bei dem -104 Protokoll kommt noch die Qualität des Netzwerks hinzu. Bei den seriellen Protokollen ist die Baudrate mitentscheidend.



Mehr Informationen:  
**[www.beckhoff.com/tf6500](http://www.beckhoff.com/tf6500)**

Beckhoff Automation GmbH & Co. KG  
Hülshorstweg 20  
33415 Verl  
Deutschland  
Telefon: +49 5246 9630  
[info@beckhoff.com](mailto:info@beckhoff.com)  
[www.beckhoff.com](http://www.beckhoff.com)

