



Siemens
Industry
Online
Support

ANWENDUNGSBEISPIEL

Bibliothek für IO-Link (LIOLink)

SIMATIC / TIA Portal / Bausteinbibliothek

SIEMENS

Rechtliche Hinweise

Nutzung der Anwendungsbeispiele

In den Anwendungsbeispielen wird die Lösung von Automatisierungsaufgaben im Zusammenspiel mehrerer Komponenten in Form von Text, Grafiken und/oder Software-Bausteinen beispielhaft dargestellt. Die Anwendungsbeispiele sind ein kostenloser Service der Siemens AG und/oder einer Tochtergesellschaft der Siemens AG ("Siemens"). Sie sind unverbindlich und erheben keinen Anspruch auf Vollständigkeit und Funktionsfähigkeit hinsichtlich Konfiguration und Ausstattung. Die Anwendungsbeispiele stellen keine kundenspezifischen Lösungen dar, sondern bieten lediglich Hilfestellung bei typischen Aufgabenstellungen. **Die Anwendungsbeispiele unterliegen nicht den Standardtests und Qualitätsprüfungen eines kostenpflichtigen Produkts und können funktionale und Leistungsdefekte sowie andere Fehler und Sicherheitslücken enthalten Sie sind verantwortlich für den ordnungsgemäßen und sicheren Betrieb der Produkte gemäß allen geltenden Vorschriften, einschließlich der Überprüfung und Anpassung des Anwendungsbeispiels für Ihr System, und stellen sicher, dass nur geschultes Personal es so verwendet, dass Sachschäden oder Verletzungen von Personen vermieden werden. Sie sind allein verantwortlich für jede produktive Nutzung.**

Sie erhalten von Siemens das nicht ausschließliche, nicht unterlizenzierbare und nicht übertragbare Recht, die Anwendungsbeispiele durch fachlich geschultes Personal zu nutzen. Jede Änderung an den Anwendungsbeispielen erfolgt auf Ihre Verantwortung. Die Weitergabe an Dritte oder Vervielfältigung der Anwendungsbeispiele oder von Auszügen daraus ist nur in Kombination mit Ihren eigenen Produkten gestattet. Jede weitere Verwendung der Anwendungsbeispiele ist ausdrücklich nicht gestattet und es werden keine weiteren Rechte gewährt. Sie dürfen die Anwendungsbeispiele in keiner anderen Weise verwenden, insbesondere ist jegliches direkte oder indirekte Training oder Verbesserungen von KI-Modellen ausdrücklich untersagt.

Haftungsausschluss

Siemens schließt seine Haftung, gleich aus welchem Rechtsgrund, insbesondere für die Verwendbarkeit, Verfügbarkeit, Vollständigkeit und Mangelfreiheit der Anwendungsbeispiele, sowie dazugehöriger Hinweise, Projektierungs- und Leistungsdaten und dadurch verursachte Schäden aus. Dies gilt nicht, soweit Siemens zwingend haftet, z.B. nach dem Produkthaftungsgesetz, in Fällen des Vorsatzes, der groben Fahrlässigkeit, wegen der schuldhaften Verletzung des Lebens, des Körpers oder der Gesundheit, bei Nichteinhaltung einer übernommenen Garantie, wegen des arglistigen Verschweigens eines Mangels oder wegen der schuldhaften Verletzung wesentlicher Vertragspflichten. Der Schadensersatzanspruch für die Verletzung wesentlicher Vertragspflichten ist jedoch auf den vertragstypischen, vorhersehbaren Schaden begrenzt, soweit nicht Vorsatz oder grobe Fahrlässigkeit vorliegen oder wegen der Verletzung des Lebens, des Körpers oder der Gesundheit gehaftet wird. Eine Änderung der Beweislast zu Ihrem Nachteil ist mit den vorstehenden Regelungen nicht verbunden. Von in diesem Zusammenhang bestehenden oder entstehenden Ansprüchen Dritter stellen Sie Siemens frei, soweit Siemens nicht gesetzlich zwingend haftet.

Durch Nutzung der Anwendungsbeispiele erkennen Sie an, dass Siemens über die beschriebene Haftungsregelung hinaus nicht für etwaige Schäden haftbar gemacht werden kann.

Weitere Hinweise

Siemens behält sich das Recht vor, Änderungen an den Anwendungsbeispielen jederzeit ohne Ankündigung durchzuführen und Ihnen die Lizenz jederzeit zu kündigen. Bei Abweichungen zwischen den Vorschlägen in den Anwendungsbeispielen und anderen Siemens Publikationen, wie z. B. Katalogen, hat der Inhalt der anderen Dokumentation Vorrang.

Ergänzend gelten die Siemens Nutzungsbedingungen (<https://www.siemens.com/global/en/general/terms-of-use.html>).

Cybersecurity-Hinweise

Siemens bietet Produkte und Lösungen mit Industrial Cybersecurity-Funktionen an, die den sicheren Betrieb von Anlagen, Systemen, Maschinen und Netzwerken unterstützen.

Um Anlagen, Systeme, Maschinen und Netzwerke gegen Cyber-Bedrohungen zu sichern, ist es erforderlich, ein ganzheitliches Industrial Cybersecurity-Konzept zu implementieren (und kontinuierlich aufrechtzuerhalten), das dem aktuellen Stand der Technik entspricht. Die Produkte und Lösungen von Siemens formen einen Bestandteil eines solchen Konzepts.

Die Kunden sind dafür verantwortlich, unbefugten Zugriff auf ihre Anlagen, Systeme, Maschinen und Netzwerke zu verhindern. Diese Systeme, Maschinen und Komponenten sollten nur mit dem Unternehmensnetzwerk oder dem Internet verbunden werden, wenn und soweit dies notwendig ist und nur wenn entsprechende Schutzmaßnahmen (z.B. Firewalls und/oder Netzwerksegmentierung) ergriffen wurden.

Weiterführende Informationen zu möglichen Schutzmaßnahmen im Bereich Industrial Cybersecurity finden Sie unter www.siemens.com/cybersecurity-industry.

Die Produkte und Lösungen von Siemens werden ständig weiterentwickelt, um sie noch sicherer zu machen. Siemens empfiehlt ausdrücklich, Produkt-Updates anzuwenden, sobald sie zur Verfügung stehen und immer nur die aktuellen Produktversionen zu verwenden. Die Verwendung veralteter oder nicht mehr unterstützter Versionen kann das Risiko von Cyber-Bedrohungen erhöhen.

Um stets über Produkt-Updates informiert zu sein, abonnieren Sie den Siemens Industrial Cybersecurity RSS Feed unter <https://www.siemens.com/cert>.

Inhaltsverzeichnis

1.	Einführung	6
2.	Bestandteile der Bibliothek.....	7
3.	Basisbausteine	11
3.1.	LIOLink_Device.....	11
3.1.1.	Beschreibung	11
3.1.2.	Parameter	12
3.1.3.	Funktionsweise	13
3.1.4.	Fehlerhandling	15
3.2.	LIOLink_Diagnose.....	16
3.2.1.	Beschreibung	16
3.2.2.	Parameter	16
3.2.3.	Funktionsweise	17
3.2.4.	Fehlerhandling	17
3.2.5.	Events.....	18
3.2.6.	Event-Codes für IO-Link Devices	19
3.2.7.	Event-Codes für die IO-Link-Master Ports.....	20
4.	Master-spezifische Bausteine	22
4.1.	LIOLink_Master.....	22
4.1.1.	Beschreibung	22
4.1.2.	Parameter	23
4.1.3.	Funktionsweise	23
4.1.4.	Fehlerhandling	24
4.2.	LIOLink_Toolchanger	25
4.2.1.	Beschreibung	25
4.2.2.	Parameter	26
4.2.3.	Status und Fehlerhandling	27
5.	Geräte-spezifische Bausteine.....	28
5.1.	SIMATIC RF200.....	28
5.1.1.	LIOLink_RF200	28
5.1.2.	LIOLink_RF200_ReadTag	31
5.1.3.	LIOLink_RF200_SwitchAntenna.....	32
5.1.4.	LIOLink_RF200_WriteTag	33
5.1.5.	Integration ins Anwenderprojekt	34
5.1.6.	Fehlerhandling	34
5.2.	SIRIUS	35

5.2.1.	Beschreibung	35
5.2.2.	Parameter	36
5.2.3.	Integration ins Anwenderprojekt	38
5.2.4.	Fehlerhandling	38
6.	Profilbausteine.....	39
6.1.	LIOLink_AdjSwitchingSensor	39
6.1.1.	Beschreibung	39
6.1.2.	Parameter	40
6.1.3.	Bedienung.....	42
6.1.3.1.	Funktionen.....	42
6.1.3.2.	Teach-In.....	43
6.1.4.	Fehlerhandling	45
6.2.	LIOLink_IdentAndDiag	46
6.2.1.	Beschreibung	46
6.2.2.	Parameter	46
6.2.3.	Funktionsweise	48
6.2.4.	Fehlerhandling	50
6.3.	LIOLink_MeasuredDataChannel.....	52
6.3.1.	Beschreibung	52
6.3.2.	Parameter	52
6.3.3.	Funktionsweise	53
6.3.4.	Fehlerhandling	56
6.4.	LIOLink_MultiAdjSwitchingSensor	57
6.4.1.	Beschreibung	57
6.4.2.	Parameter	58
6.4.3.	Bedienung.....	60
6.4.3.1.	Konfigurationseinstellungen und Sollwerte.....	60
6.4.3.2.	Schaltpunktmodi	61
6.4.3.3.	Funktionen.....	63
6.4.3.4.	Kanal einlernen (Teach Channel).....	63
6.4.3.5.	Teach-In.....	64
7.	Kopiervorlagen	67
7.1.	LIOLink_MeasuredData	67
7.1.1.	Beschreibung	67
7.1.2.	Parameter	67
7.1.3.	Funktionsweise	68
7.1.4.	Fehlerhandling	71
8.	Wissenswertes	72

8.1.	Grundlagen IO-Link.....	72
8.1.1.	Was ist IO-Link?	72
8.1.2.	Projektierung des IO-Link Masters.....	72
8.2.	Bibliotheken im TIA Portal	75
9.	Anhang	76
9.1.	Service und Support.....	76
9.2.	Links und Literatur.....	77
9.3.	Änderungsdokumentation	77

1. Einführung

Überblick

Systematische Diagnosekonzepte und der effiziente Umgang von Parameterdaten werden auf allen Ebenen der Automatisierungstechnik verlangt. Dazu ist es unerlässlich, dass Sensoren und Schaltgeräte in den Automatisierungsverbund eingebunden werden.

Der Kommunikationsstandard IO-Link bietet hierbei durch die intelligente Anbindung von Sensoren und Schaltgeräten an die Steuerungsebene neue Möglichkeiten. Kernpunkte sind das Schalten, Schützen und Überwachen auf der Feldebene.

Das IO-Link-System besteht aus einem IO-Link Master und einem oder mehreren IO-Link Devices (Sensoren oder Aktoren). Der IO-Link Master fungiert als Schnittstelle zur überlagerten Steuerung und steuert die Kommunikation mit den angeschlossenen IO-Link Devices.

Diese Bibliothek stellt Bausteine und PLC-Datentypen zur Verfügung, um die Kommunikation zwischen SIMATIC Controller und IO-Link Master oder IO-Link Device zu ermöglichen.

Funktionsumfang

Die Bausteine dieser Bibliothek sind in drei Gruppen unterteilt:

- Allgemein verwendbare **Basisbausteine** zur Kommunikation mit SIMATIC IO-Link Master und beliebigen IO-Link Devices und zum Auslesen von deren IO-Link-Diagnose.
- **Master-spezifische Bausteine**, die für Backup&Restore von SIMATIC IO-Link-Mastern und zur Ausführung von Portfunktionen verwendet werden.
- **Geräte-spezifische Bausteine** vereinfachen die Kommunikation mit jeweils einem bestimmten IO-Link Device durch eine darauf zugeschnittene Schnittschnelle und vordefinierte PLC-Datentypen. Diese Bausteine bauen auf den Basisbausteinen auf.
- **Profilbausteine** vereinfachen die Kommunikation mit einem Gerät über ein bestimmtes IO-Link Profil. Diese Bausteine bauen auf den Basisbausteinen auf.

Gültigkeit

Diese Bibliothek ist verfügbar für TIA Portal V16 und höher.

Alle Bausteine der Bibliothek stehen für SIMATIC S7-1200/1500 Controller und größtenteils auch für SIMATIC S7-300/400 Controller zur Verfügung.

2. Bestandteile der Bibliothek

Nachfolgend finden Sie eine Auflistung aller Bausteine dieser Bibliothek mit der aktuellen Version.

Basisbausteine

Tabelle 2-1: Basisbausteine

Name	Version	Beschreibung
LIOLink_Device	V 5.1.0	Ermöglicht das Lesen und Schreiben azyklischer Daten eines IO-Link Device
LIOLink_Diagnose	V 2.0.0	Ermöglicht das Auslesen der Diagnose aller Ports des IO-Link-Masters

Master-spezifische Bausteine

Tabelle 2-2: Master-spezifische Bausteine

Name	Version	Beschreibung
LIOLink_Master	V 4.0.0	Ermöglicht das Sichern (Backup) und Wiederherstellen (Restore) von Geräteparametern und -einstellungen eines IO-Link Masters über das S7-Programm (Mastertausch ohne Engineering Tool)
LIOLink_Toolchanger	V1.0.0	Ermöglicht das Suspendieren und die Wiederaufnahme des Portbetriebs oder der Diagnose von Ports des IO-Link-Masters

Geräte-spezifische Bausteine

Tabelle 2-3: Geräte-spezifische Bausteine

Name	Version	Beschreibung
LIOLink_3RA	V 4.1.0	Dient zur einfachen Kommunikation mit SIRIUS 3RA Geräten
LIOLink_3RB24	V 4.1.0	Dient zur einfachen Kommunikation mit SIRIUS 3RB24 Geräten
LIOLink_3RR24	V 4.1.0	Dient zur einfachen Kommunikation mit SIRIUS 3RR24 Geräten
LIOLink_3RS1	V 4.1.0	Dient zur einfachen Kommunikation mit SIRIUS 3RS1 Geräten
LIOLink_3RS2	V 2.1.0	Dient zur einfachen Kommunikation mit SIRIUS 3RS2 Geräten
LIOLink_3SU1_ElecModule	V 3.1.0	Dient zur einfachen Kommunikation mit SIRIUS 3SU1 Elektronikmodulen
LIOLink_3SU1_IDKeySwitch	V 3.1.0	Dient zur einfachen Kommunikation mit SIRIUS 3SU1 ID-Schlüsselschaltern
LIOLink_8WD46	V 1.1.0	Dient zur einfachen Kommunikation mit SIRIUS 8WD46 Signalleuchten
LIOLink_3UG481x	V 4.1.0	Dient zur einfachen Kommunikation mit SIRIUS 3UG481 Geräten
LIOLink_3UG4822	V 4.1.0	Dient zur einfachen Kommunikation mit SIRIUS 3UG4822 Geräten
LIOLink_3UG4825	V 4.1.0	Dient zur einfachen Kommunikation mit SIRIUS 3UG4825 Geräten
LIOLink_3UG4832	V 4.1.0	Dient zur einfachen Kommunikation mit SIRIUS 3UG4832 Geräten
LIOLink_3UG4841	V 4.1.0	Dient zur einfachen Kommunikation mit SIRIUS 3UG4841 Geräten
LIOLink_3UG4851	V 4.1.0	Dient zur einfachen Kommunikation mit SIRIUS 3UG4851 Geräten
LIOLink_RF200	V 4.1.0	Dient zur einfachen Kommunikation mit SIMATIC RF200 Readern
LIOLink_RF200_ReadTag	V 3.0.1	Dient speziell zum Lesen eines Transponders über einen SIMATIC RF200 Reader
LIOLink_RF200_SwitchAntenna	V 3.0.0	Dient speziell zum Ein- und Ausschalten des Antennenfelds eines SIMATIC RF200 Readers
LIOLink_RF200_WriteTag	V 3.0.1	Dient speziell zum Lesen eines Transponders über einen SIMATIC RF200 Reader

Profilbausteine

Tabelle 2-4: Profilbausteine

Name	Version	Beschreibung
LIOLink_AdjSwitchingSensor	V 2.1.0	Dient zum Einstellen oder Einlernen des Sollwerts und zum Ändern der Schaltungsklogik von verstellbaren Schaltsensoren oder Adjustable Switching Sensors (AdSS)
LIOLink_IdentAndDiag	V 2.1.0	Liest und schreibt azyklisch Identifikations- und Diagnosedaten und gibt den Status des angeschlossenen IO-Link Device aus. Diese Funktionsbaustein unterstützt Profil "Identification und Diagnosis"
LIOLink_MeasuredDataChannel	V 2.0.0	Mit diesem Funktionsbaustein können zyklisch Messwerte der Sensorik erfasst werden. Die gemessenen Rohwerte der Sensoren werden verarbeitet und stehen als Real- oder DInt-Messwert an den jeweiligen Ausgängen zur Verfügung. Diese Funktionsbaustein unterstützt Smart Sensor-Profil Typ 3 ("Digital Measuring Sensors") und Typ 4 ("Digital Measuring Switching Sensors").
LIOLink_MultiAdjSwitchingSensor	V 2.0.0	Bietet eine einheitliche Schnittstelle für den Zugriff und die Parametrierung von IO Link Devices, die das Smart Sensor-Profil unterstützen. Insbesondere kann der Baustein von Sensoren eingesetzt werden, die dem Messgerät Profil Typ 2 zugeordnet werden können, d. h. IO-Link Devices, die das Smart Sensor-Profil "Multiple Adjustable Switching Sensors" unterstützen.

Kopiervorlagen

Tabelle 2-5: Kopiervorlagen

Name	Version	Beschreibung
LIOLink_MeasuredData		Dient der automatischen Erfassung und Aufbereitung von Prozessdaten von IO-Link Sensoren unabhängig vom Sensortyp.

PLC-Datentypen

Tabelle 2-6: PLC-Datentypen

Name	Version	Beschreibung
LIOLink_3RA_typeAll	V 3.0.0	Struktur für alle unterstützen Datensätze für SIRIUS 3RA Geräte
LIOLink_3RA_typeConfig	V 3.0.0	Beschreibt die Datensätze "Istkonfiguration" (105) und "Sollkonfiguration" (130)
LIOLink_3RA_typeConfigStarterLong	V 1.0.0	Beschreibt die Konfigurationen für Starter mit Typ "long"
LIOLink_3RA_typeConfigStarterShort	V 1.0.0	beschreibt die Konfigurationen für Starter mit Typ "short"
LIOLink_3RA_typeDiag	V 3.0.0	Beschreibt den Datensatz "Diagnose" (92)
LIOLink_3RA_typeDiagStarterLong	V 1.0.0	Beschreibt die Diagnose von Startern mit dem Typ "long"
LIOLink_3RA_typeDiagStarterShort	V 1.0.0	Beschreibt die Diagnose von Startern mit dem Typ "short"
LIOLink_3RA_typeParameterPage1	V 3.0.0	Beschreibt den Datensatz "Parameterseite 1"
LIOLink_3RA_typePII	V 1.0.0	Beschreibt das Prozessabbild der Eingänge
LIOLink_3RA_typePIQ	V 1.0.0	Beschreibt das Prozessabbild der Ausgänge
LIOLink_3RB24_typeAll	V 3.0.0	Struktur für alle unterstützen Datensätze für SIRIUS 3RB24 Geräte
LIOLink_3RB24_typeDiag	V 3.0.0	Beschreibt den Datensatz "Diagnose" (92)
LIOLink_3RB24_typeMeasure	V 3.0.0	Beschreibt den Datensatz "Messwerte" (94)
LIOLink_3RB24_typePII	V 1.0.0	Beschreibt das Prozessabbild der Eingänge
LIOLink_3RB24_typePIQ	V 1.0.0	Beschreibt das Prozessabbild der Ausgänge
LIOLink_3RB24_typePresetConfig	V 3.0.0	Beschreibt den Datensatz "Sollkonfiguration" (130)
LIOLink_3RB24_typeTechFunctions	V 3.0.0	Beschreibt den Datensatz "Technologiefunktionen" (131)
LIOLink_3RR24_typeAll	V 3.0.0	Struktur für alle unterstützen Datensätze für SIRIUS 3RR24 Geräte
LIOLink_3RR24_typeDiag	V 3.0.0	Beschreibt den Datensatz "Diagnose" (92)

Name	Version	Beschreibung
LIOLink_3RR24_typeMeasure	V 3.0.0	Beschreibt den Datensatz "Messwerte" (94)
LIOLink_3RR24_typeParam	V 3.0.0	Beschreibt den Datensatz "Parameter" (131)
LIOLink_3RR24_typePII	V 1.0.0	Beschreibt das Prozessabbild der Eingänge
LIOLink_3RR24_typePIQ	V 1.0.0	Beschreibt das Prozessabbild der Ausgänge
LIOLink_3RS1_typeAll	V 3.0.0	Struktur für alle unterstützten Datensätze für SIRIUS 3RS1 Geräte
LIOLink_3RS1_typeDiag	V 3.0.0	Beschreibt den Datensatz "Diagnose" (92)
LIOLink_3RS1_typeMeasure	V 3.0.0	Beschreibt den Datensatz "Messwerte" (94)
LIOLink_3RS1_typeParam	V 3.0.0	Beschreibt den Datensatz "Parameter" (131)
LIOLink_3RS1_typePII	V 1.0.0	Beschreibt das Prozessabbild der Eingänge
LIOLink_3RS1_typePIQ	V 1.0.0	Beschreibt das Prozessabbild der Ausgänge
LIOLink_3RS1_typeSensor	V 3.0.0	Beschreibt den Zustand eines Temperatursensors
LIOLink_3RS2_typeAll	V 1.0.0	Struktur für alle unterstützten Datensätze für SIRIUS 3RS2 Geräte
LIOLink_3RS2_typeCurrentInput	V 1.0.0	Beschreibt den Stromeingang 4...20mA
LIOLink_3RS2_typeDiag	V 1.0.0	Beschreibt den Datensatz "Diagnose" (92)
LIOLink_3RS2_typeMeasure	V 1.0.0	Beschreibt den Datensatz "Messwerte" (94)
LIOLink_3RS2_typeMeasureFloat	V 1.0.0	Beschreibt den Datensatz "Messwerte" (94) im Fließkommaformat
LIOLink_3RS2_typeParam	V 1.0.0	Beschreibt den Datensatz "Parameter" (131)
LIOLink_3RS2_typePII	V 1.0.0	Beschreibt das Prozessabbild der Eingänge
LIOLink_3RS2_typePIQ	V 1.0.0	Beschreibt das Prozessabbild der Ausgänge
LIOLink_3RS2_typeSensor	V 1.0.0	Beschreibt den Zustand eines Temperatursensors
LIOLink_3SU1_ElecModule_typeAll	V 2.0.0	Struktur für alle unterstützten Datensätze für SIRIUS ACT 3SU1 Elektronikmodule
LIOLink_3SU1_ElecModule_typeDiag92	V 2.0.0	Beschreibt den Datensatz "Diagnose" (92)
LIOLink_3SU1_ElecModule_typeDiag94	V 2.0.0	Beschreibt den Datensatz "Diagnose" (94)
LIOLink_3SU1_ElecModule_typeParam	V 2.0.0	Beschreibt den Datensatz "Parameter" (131)
LIOLink_3SU1_ElecModule_typePII	V 1.0.1	Beschreibt das Prozessabbild der Eingänge
LIOLink_3SU1_ElecModule_typePIQ	V 1.0.1	Beschreibt das Prozessabbild der Ausgänge
LIOLink_3SU1_IDKeySwitch_typeAll	V 2.0.0	Struktur für alle unterstützten Datensätze für SIRIUS ACT 3SU1 ID-Schlüsselschalter
LIOLink_3SU1_IDKeySwitch_typeDiag92	V 2.0.0	Beschreibt den Datensatz "Diagnose" (92)
LIOLink_3SU1_IDKeySwitch_typeDiag94	V 2.0.0	Beschreibt den Datensatz "Diagnose" (94)
LIOLink_3SU1_IDKeySwitch_typeKeyList1	V 2.0.0	Beschreibt den Datensatz "Individuell codierbare ID Schlüssel (1-30)" (81)
LIOLink_3SU1_IDKeySwitch_typeKeyList2	V 2.0.0	Beschreibt den Datensatz "Individuell codierbare ID Schlüssel (31-50)" (82)
LIOLink_3SU1_IDKeySwitch_typeKeyState	V 2.0.0	Beschreibt den Zustand eines ID-Schlüssels
LIOLink_3SU1_IDKeySwitch_typeParam	V 2.0.0	Beschreibt den Datensatz "Parameter" (131)
LIOLink_3SU1_IDKeySwitch_typePII	V 1.0.0	Beschreibt das Prozessabbild der Eingänge
LIOLink_3UG481_typeAll	V 3.0.0	Struktur für alle unterstützten Datensätze für SIRIUS 3UG481 Geräte
LIOLink_3UG481_typeDiag	V 3.0.0	Beschreibt den Datensatz "Diagnose" (92)
LIOLink_3UG481_typeMeasure	V 3.0.0	Beschreibt den Datensatz "Messwerte" (94)
LIOLink_3UG481_typeParam	V 3.0.0	Beschreibt den Datensatz "Parameter" (131)
LIOLink_3UG4822_typeAll	V 3.0.0	Struktur für alle unterstützten Datensätze für SIRIUS 3UG4822 Geräte
LIOLink_3UG4822_typeDiag	V 3.0.0	Beschreibt den Datensatz "Diagnose" (92)
LIOLink_3UG4822_typeMeasure	V 3.0.0	Beschreibt den Datensatz "Messwerte" (94)
LIOLink_3UG4822_typeParam	V 3.0.0	Beschreibt den Datensatz "Parameter" (131)
LIOLink_3UG4825_typeAll	V 3.0.0	Struktur für alle unterstützten Datensätze für SIRIUS 3UG4825 Geräte
LIOLink_3UG4825_typeDiag	V 3.0.0	Beschreibt den Datensatz "Diagnose" (92)

Name	Version	Beschreibung
LIOLink_3UG4825_typeMeasure	V 3.0.0	Beschreibt den Datensatz "Messwerte" (94)
LIOLink_3UG4825_typeParam	V 3.0.0	Beschreibt den Datensatz "Parameter" (131)
LIOLink_3UG4832_typeAll	V 3.0.0	Struktur für alle unterstützten Datensätze für SIRIUS 3UG4832 Geräte
LIOLink_3UG4832_typeDiag	V 3.0.0	Beschreibt den Datensatz "Diagnose" (92)
LIOLink_3UG4832_typeMeasure	V 3.0.0	Beschreibt den Datensatz "Messwerte" (94)
LIOLink_3UG4832_typeParam	V 3.0.0	Beschreibt den Datensatz "Parameter" (131)
LIOLink_3UG4841_typeAll	V 3.0.0	Struktur für alle unterstützten Datensätze für SIRIUS 3UG4841 Geräte
LIOLink_3UG4841_typeDiag	V 3.0.0	Beschreibt den Datensatz "Diagnose" (92)
LIOLink_3UG4841_typeMeasure	V 3.0.0	Beschreibt den Datensatz "Messwerte" (94)
LIOLink_3UG4841_typeParam	V 3.0.0	Beschreibt den Datensatz "Parameter" (131)
LIOLink_3UG4851_typeAll	V 3.0.0	Struktur für alle unterstützten Datensätze für SIRIUS 3UG4851 Geräte
LIOLink_3UG4851_typeDiag	V 3.0.0	Beschreibt den Datensatz "Diagnose" (92)
LIOLink_3UG4851_typeMeasure	V 3.0.0	Beschreibt den Datensatz "Messwerte" (94)
LIOLink_3UG4851_typeParam	V 3.0.0	Beschreibt den Datensatz "Parameter" (131)
LIOLink_8WD46_typeAll	V 1.0.0	Struktur, die alle möglichen Parameter der 8WD46 zusammenfasst
LIOLink_8WD46_typeLightEffects	V 1.0.0	Struktur zur Parametrierung der Lichteffekte
LIOLink_8WD46_typeSoundMode	V 1.0.0	Struktur zur Parametrierung der Soundeffekte
LIOLink_RF200_typeAll	V 3.0.0	Struktur für alle unterstützten Datensätze für SIMATIC RF200 Reader
LIOLink_RF200_typeEventHistory	V 3.0.1	Beschreibt den Datensatz "Ereignisverlauf" (0x4A)
LIOLink_RF200_typeParameters	V 1.0.0	Beschreibt den Datensatz "Leser-Parameter" (0x40)
LIOLink_RF200_typePII	V 1.0.0	Beschreibt das Prozessabbild der Eingänge
LIOLink_RF200_typePIQ	V 1.0.0	Beschreibt das Prozessabbild der Ausgänge
LIOLink_RF200_typeReaderStatus	V 3.0.0	Beschreibt den Datensatz "Reader-Status" (0x5A)
LIOLink_RF200_typeTagStatus	V 3.0.0	Beschreibt den Datensatz "Transponder-Status" (0x5B)
LIOLink_RF200_typeUIDHistory	V 3.0.0	Beschreibt den Datensatz "UID-Verlauf" (0x5C)
LIOLink_typeDiagnostics	V 1.0.0	Stellt eine Diagnosestruktur für diverse Bausteine der Bibliothek bereit, um im Fehlerfall detaillierte Informationen zu liefern.
LIOLink_typeEvents	V 1.0.0	Beschreibt die Ereignisse der einzelnen Ports eines IO-Link-Masters
LIOLink_typePortEventCodes	V 1.0.0	Struktur für Ereignisinformationen: Ereigniscodes und EventQualifier
LIOLink_typePortEventQualifier	V 1.0.0	Struktur für EventQualifier: Instanz, Quelle, Typ, Modus
LIOLink_typePortEvents	V 1.0.0	Struktur zur Anzeige der letzten 5 Ereignisse eines Ports
LIOLink_typeIdentificationObjects	V 1.0.1	Beschreibt alle Profil-relevanten Geräteparameter gemäß Common-Profil
LIOLink_typeParameterPage0	V 3.0.0	Beschreibt die Parameterseite 0, die alle IO-Link Devices standardmäßig bereitstellen.
LIOLink_typeConfigMultiAdjSwSensor	V 1.0.0	Definiert die Werte für die Konfigurationseinstellungen, die bei einer Anfrage mit der Funktion rd_all / wr_conf gelesen/geschrieben werden sollen
LIOLink_typeParamMultiAdjSwSensor	V 1.0.0	Definiert die Werte für die zu lesenden/schreibenden Sollwertparameter bei einer Anfrage mit der Funktion rd_all / wr_conf
LIOLink_typeSensorInfo		Informationen über den Sensor. Diese Informationen sind erforderlich, um den Sensor zu erkennen und die vom Sensor kommenden Daten zu verarbeiten

HINWEIS Weitere Informationen zu den einzelnen Datensätzen finden Sie in den zugehörigen Gerätehandbüchern.

3. Basisbausteine

3.1. LIOLink_Device

3.1.1. Beschreibung

Durch Lesen und Schreiben azyklischer Daten können Sie Geräteparameter zu einem IO-Link Device schreiben bzw. Parameter, Messwerte und Diagnosedaten von einem IO-Link Device lesen.

Der Funktionsbaustein unterstützt Sie bei folgenden Aufgaben:

- (Um)Parametrierung eines IO-Link Device
- Diagnose eines IO-Link Device
- Ausführen von IO-Link Portfunktionen
- Sichern/Wiederherstellung von IO-Link Geräteparametern

Die Daten auf dem IO-Link Device werden mit Index und Subindex eindeutig adressiert. Zudem können mit dem Funktionsbaustein auch Port-Funktionen ausgeführt werden.

HINWEIS Der Aufbau der Datenobjekte und der Port-Funktionen ist der jeweiligen Dokumentation des IO-Link Device und IO-Link Master zu entnehmen.

Der Funktionsbaustein basiert auf einem standardisierten Protokoll (PROFIBUS DP/PROFINET IO), das den Zugriff auf Daten eines IO-Link Device über den IO-Link Master sicherstellt. Grundsätzlich wird eine Sequenz von azyklischen Lese- und Schreibzugriffen genutzt, die durch die SIMATIC Systemfunktionen "RDREC" und "WRREC" repräsentiert werden.

Der Baustein ist ein asynchron arbeitender Funktionsbaustein, d. h. die Bearbeitung erstreckt sich über mehrere SPS-Zyklen.

HINWEIS Wenn ein DPV1-Slave über GSD-Datei projektiert ist (GSD ab Rev. 3) und die DP-Schnittstelle des DP-Master auf "S7-kompatibel" eingestellt ist, funktioniert der Baustein nicht korrekt.
Abhilfe: Schnittstelle des DP-Master auf "DPV1" umstellen.

HINWEIS Dieser Baustein ersetzt die bisher verfügbaren Bausteine "IO_LINK_DEVICE" und "IO_LINK_CALL".

Gültigkeit

Der FB steht in zwei Varianten für S7-1200/1500 und S7-300/400 zur Verfügung. Dabei existieren allerdings geringfügige Unterschiede. (vgl. [Tabelle 3-1: Parameter von LIOLink_Device](#))

3.1.2. Parameter

Abbildung 3-1: LIOLink_Device

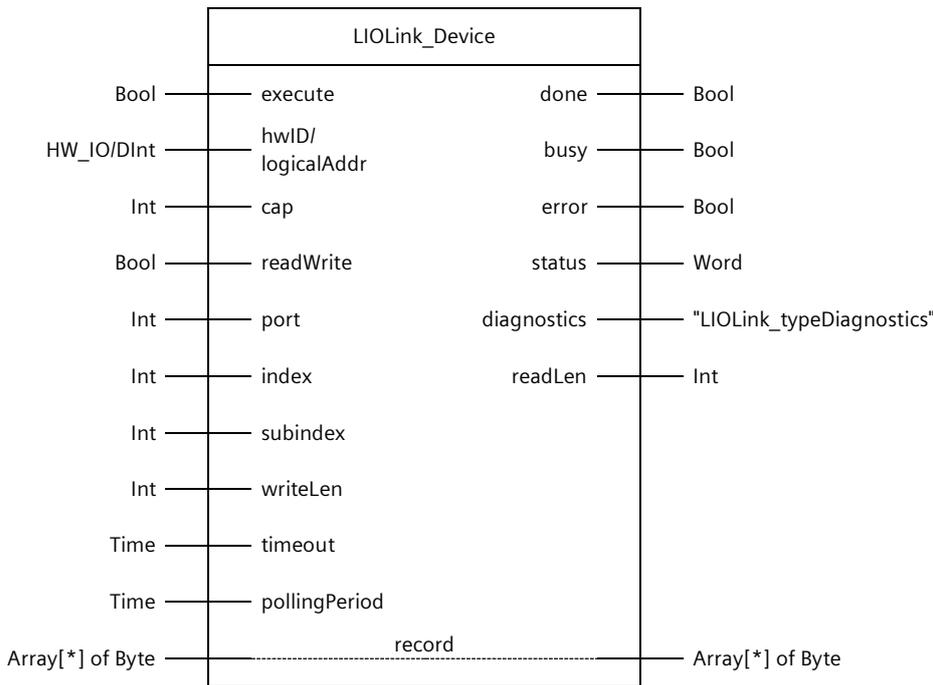


Tabelle 3-1: Parameter von LIOLink_Device

Name	P-Typ	Datentyp	Kommentar
execute	IN	Bool	Anforderung, die Funktion auszuführen
hwID/ logicalAddr	IN	HW_IO/DInt	S7-1200/1500: Hardwarekennung des IO-Link Masters oder des ersten Submoduls. S7-300/400: Logische Adresse des IO-Link Masters (Modul oder Submodul)
cap (optional)	IN	Int	Zugangspunkt (Client Access Point): S7-1200/1500: Wird bei der Verwendung von Siemens IO-Link Mastern automatisch erkannt und ist daher optional. Wird der Parameter verändert, ist die automatische Erkennung nicht aktiv. S7-300/400: Die automatische Erkennung ist in dieser Version nicht implementiert. Der Zugangspunkt muss daher manuell gesetzt werden. Für gewöhnlich besitzt er den Wert 0xB400 oder 0x00E3 und ist im jeweiligen Handbuch angegeben.
readWrite	IN	Bool	Modus FALSE: lesen TRUE: schreiben
port	IN	Int	Portnummer an dem das IO-Link Device betrieben wird Mögliche Werte: 0..63
index	IN	Int	Parameterindex Mögliche Werte: 0..32767 65535 (0xFFFF): IOL-D - Portfunktionen
subindex	IN	Int	Parametersubindex 0: gesamter Record 1..255: Parameter aus Record

Name	P-Typ	Datentyp	Kommentar
writeLen	IN	Int	Länge der zu schreibenden Daten in Bytes+ (Nettodaten) Mögliche Werte beim Schreiben: 1..232 Nicht relevant beim Lesen
timeout	IN	Time	Zeit, nach der ein Auftrag abgebrochen wird
pollingPeriod (optional)	IN	Time	S7-1200/1500: Variabel einstellbare Zeit, die der Baustein wartet, bis der Datensatz übermittelt wird. Defaultwert: 100 ms S7-300/400: Die Funktionalität ist in dieser Version nicht implementiert, weswegen der Parameter hier nicht existiert.
done	OUT	Bool	TRUE: Auftrag wurde erfolgreich ausgeführt
busy	OUT	Bool	TRUE: Auftrag wird gerade bearbeitet
error	OUT	Bool	TRUE: Ein Fehler ist bei der Bearbeitung des FB aufgetreten
status	OUT	Word	16#0000–16#7FFF: Status des FB 16#8000–16#FFFF: Fehlercodes (siehe Kapitel 3.1.4)
diagnostics	OUT	"LIOLink_ typeDiagnostics"	Detaillierte Diagnoseinformationen des FB (siehe Kapitel 3.1.4)
readLen	OUT	Int	Länge der gelesenen Daten in Bytes (Nettodaten)
record	IN_OUT	Array[*] of Byte	S7-1200/1500: Quell-/Zielbereich für die zu lesenden/schreibenden Daten. Möglicher Wertebereich: 0..231 S7-300/400: Datentyp: Array[0...231] of Byte

3.1.3. Funktionsweise

Adressierung

Der gewünschte Datensatz wird mit den Parametern "index" und "subindex" eindeutig adressiert.

Beim Schreiben von Daten wird die am Parameter "writeLen" spezifizierte Datenmenge zum IO-Link Device übertragen. Beim Lesen ist der Parameter irrelevant.

Über den Parameter "cap" (Client Access Point) wird der Zugangspunkt zum IO-Link Master definiert. Bei der Verwendung von IO-Link Mastern der Siemens AG erfolgt eine automatische Erkennung. Wird der Parameter allerdings verändert, ist die automatische Erkennung nicht aktiv. Für gewöhnlich ist der Zugangspunkt 0xB400 oder 0x00E3.

HINWEIS

Die automatische Erkennung des Parameters "cap" ist nur in den Bausteinen der S7-1200/1500 implementiert. Für S7-300/400 muss dieser manuell am Baustein gesetzt werden.

Zeitlicher Ablauf einer Übertragung

Die Datenübertragung wird mit einer positiven Flanke am Eingang "execute" gestartet.

Die Ausgänge "done", "busy", "error" und "status" zeigen den Zustand des Auftrags an.

Nach erfolgreicher Ausführung zeigt der Parameter "len" die Länge der empfangenen oder geschriebenen Daten an.

Solange der Eingang "execute" gesetzt ist, behalten die Ausgangsparameter ihren Wert bei. Wird der Eingang "execute" zurückgesetzt, bevor die Bearbeitung des FB abgeschlossen ist, werden die Werte der Ausgangsparameter nach der Bearbeitung des Auftrages für einen Zyklus ausgegeben.

Überschreitet die Ausführung des Auftrags die Zeit am Parameter "timeout", wird die Bearbeitung abgebrochen und ein Fehler ausgegeben.

HINWEIS

Die Datenübertragung erfolgt in Form von Rohdaten (ARRAY of Byte), d. h. die Daten können in dieser Form nicht interpretiert werden.

Es steht dem Anwender frei, die Daten entsprechend der Vorgaben des Geräteherstellers zu formatieren (Kopie in eine Datenstruktur oder Datentyp).

Parameteränderungen bestätigen

Nachdem alle gewünschten Änderungen der IO-Link Device-Parameter mit dem Baustein geschrieben wurden, müssen die Daten durch ein Systemkommando bestätigt werden, damit sie als Backup im Master bereitstehen.

Dieses Systemkommando führen Sie durch den folgenden Auftrag aus:

Tabelle 3-2: Parameteränderungen bestätigen

Parameter	Wert
mode	1
index	2
subindex	0
writeLen	1
record[0]	0x05

Abbildung 3-2: Auszug aus der IO-Link Spezifikation

Table B.9 – Coding of SystemCommand (ISDU)

Command (hex)	Command (dec)	Command name	M/O	Definition
0x00	0	Reserved		
0x01	1	ParamUploadStart	O	Start parameter upload
0x02	2	ParamUploadEnd	O	Stop parameter upload
0x03	3	ParamDownloadStart	O	Start parameter download
0x04	4	ParamDownloadEnd	O	Stop parameter download
0x05	5	ParamDownloadStore	O	Finalize parameterization and start Data Storage

3.1.4. Fehlerhandling

Der Ausgang "status" gibt den aktuellen Status sowie Fehler aus, während der Ausgang "diagnostics" im Fehlerfall eine Diagnosestruktur mit detaillierten Informationen bereitstellt.

status

Tabelle 3-3: Ausgang "status" von LIOLink_Device

Status	Bedeutung
16#0000	Auftrag abgeschlossen, keine Warnung und keine weitere Detaillierung
16#7000	Kein Auftrag in Bearbeitung (Initialwert)
16#7001	Erster Aufruf nach Eingang eines neuen Auftrags (steigende Flanke "execute")
16#7002	Folgeaufruf
16#8201	Nicht-unterstützter Port
16#8202	Nicht-unterstützter Index
16#8203	Nicht-unterstützter Subindex
16#8205	Die Länge am Parameter "writeLen" passt nicht zum Datensatz, der geschrieben werden soll
16#8401	IO-Link Master hat einen Fehlercode zurückgemeldet, siehe "diagnostics"
16#8402	Empfangener Datensatz passt nicht zum Auftrag
16#8403	Auftrag konnte nicht in der vorgegebenen Zeit abgeschlossen werden
16#8600	Interner Zustandsautomat hat einen undefinierten Zustand erreicht
16#8601	Systemfunktion WRREC meldet einen Fehler, siehe "diagnostics"
16#8602	Systemfunktion RDREC meldet einen Fehler, siehe "diagnostics"

diagnostics

Der Ausgang "diagnostics" gibt im Fehlerfall detaillierte Informationen zu dem vorliegenden Fehler aus.

Tabelle 3-4: Ausgang "diagnostics" von LIOLink_Device

Variable	Beschreibung
status	Letzter Statuscode des Schnittstellenparameters "status" des FB.
subfunctionStatus	Status von Systemfunktionen RDREC/WRREC oder Fehlercode vom IO-Link Master (%W1: Fehlercode vom IO-Link Master, %W0: ISDU-Fehlercode). Detaillierte Informationen entnehmen Sie der Online-Hilfe zur jeweiligen Systemfunktion bzw. dem Gerätehandbuch des IO-Link Masters/Devices.
stateNumber	Zustand des Zustandsautomaten des FB, in dem der Fehler aufgetreten ist.

3.2. LIOLink_Diagnose

3.2.1. Beschreibung

Mithilfe dieses Bausteins ist es möglich detaillierte Diagnosedaten eines Siemens IO-Link-Masters und der daran angeschlossenen Geräte auszulesen.

HINWEIS

Dieser Funktionsbaustein liefert die Diagnose von IO-Link-Mastern mit maximal 8 Ports. Bei mehr als 8 Ports liefert dieser Baustein nur die Diagnosedaten für die ersten 8 Ports.

Gültigkeit

Der Baustein ist für die folgenden IO-Link Master freigegeben:

- ET 200SP, CM 4xIO-Link (6ES7137-6BD00-0BA0) ab FW V2.2.0
- ET 200pro, EM 4 IO-LINK HF (6ES7147-4JD00-0AB0) ab FW V1.0.0
- ET 200AL, CM 4xIO-Link (6ES7147-5JD00-0BA0) ab FW V1.1.0
- ET 200eco PN, IO-Link Master (6ES7148-6JD00-0AB0) ab FW V1.0.0
- ET 200eco PN, CM 8x IO-Link, M12-L (6ES7148-6JG00-0BB0) ab FW V1.0.0
- ET 200eco PN, CM 4x IO-Link + DIQ 12x 24VDC (6ES7148-6JE00-0BB0) ab FW V1.0.0
- ET 200eco PN, CM 8x IO-Link + DIQ 8x 24VDC (6ES7148-6JJ00-0BB0) ab FW V1.0.0
- ET 200clean, CM 8x IO-Link + DIQ 4x 24V DC (6ES7148-7JH00-0BB0) ab FW V1.1.0
- S7-1200, SM 1278 4 IO-Link (6ES7278-4BD32-0XB0) ab FW V2.0.0
- S7-1500, CM 8xIO-Link (6ES7547-1JF00-0AB0) ab FW V1.0.0

Der FB steht in einer Variante für S7-1200/1500 zur Verfügung.

3.2.2. Parameter

Abbildung 3-4: LIOLink_Diagnose

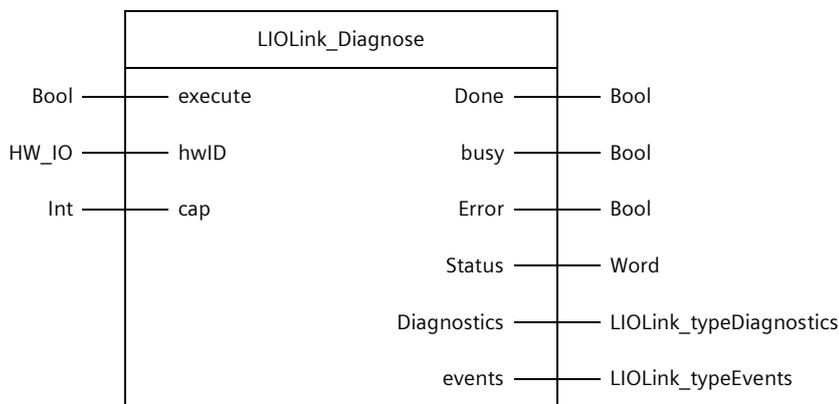


Tabelle 3-9: Parameter von LIOLink_Diagnose

Name	P-Typ	Datentyp	Kommentar
execute	IN	Bool	Anforderung, die Funktion auszuführen
hwID	IN	HW_IO	Hardwareerkennung des IO-Link Masters oder des ersten Submoduls.

Name	P-Typ	Datentyp	Kommentar
cap (optional)	IN	Int	Zugangspunkt (Client Access Point): Wird bei der Verwendung von Siemens IO-Link Mastern automatisch erkannt und ist daher optional. Wird der Parameter verändert, ist die automatische Erkennung nicht aktiv. Für gewöhnlich ist der Zugangspunkt 0xB400 oder 0x00E3 und ist im jeweiligen Handbuch angegeben.
done	OUT	Bool	TRUE: Auftrag wurde erfolgreich ausgeführt
busy	OUT	Bool	TRUE: Auftrag wird gerade bearbeitet
error	OUT	Bool	TRUE: Ein Fehler ist bei der Bearbeitung des FB aufgetreten
status	OUT	Word	16#0000–16#7FFF: Status des FB 16#8000–16#FFFF: Fehlercodes (siehe Kapitel 3.2.4)
diagnostics	OUT	"LIOLink_ typeDiagnostics"	Detaillierte Diagnoseinformationen des FB (siehe Kapitel 3.2.4)
events	OUT	"LIOLink_ typeEvents"	Die Ereignisse der einzelnen Ports eines IO-Link-Masters (siehe Kapitel 3.2.5)

3.2.3. Funktionsweise

Die Datenübertragung wird mit einer positiven Flanke am Eingang "execute" gestartet.

Die Ausgänge "done", "busy", "error" und "status" zeigen den Zustand des Auftrags an.

Nach erfolgreicher Ausführung zeigt der Parameter "events" die Ereignisse/Diagnosen aller IO-Link-Master-Ports und der angeschlossenen Geräte.

Solange der Eingang "execute" gesetzt ist, behalten die Ausgangsparameter ihren Wert bei. Wird der Eingang "execute" zurückgesetzt, bevor die Bearbeitung des FB abgeschlossen ist, werden die Werte der Ausgangsparameter "done", "busy", "error" und "status" nach der Bearbeitung des Auftrages für einen Zyklus ausgegeben. Der Ausgangsparameter "events" wird jedes Mal aktualisiert, wenn der Eingang "execute" ausgelöst wird.

3.2.4. Fehlerhandling

Das Fehlerhandling geschieht über die zwei Ausgänge "status" und "diagnostics" und wird im Folgenden genauer erläutert.

status

Der Ausgang "status" gibt den aktuellen Zustand des Bausteins, sowie Fehlercodes der ausgeführten Systemfunktion RDREC aus.

Tabelle 3-5: Ausgang "status" von LIOLink_Diagnose

Code / Value	Identifizier / Description
16#0000	Auftrag abgeschlossen, keine Warnung und keine weitere Detaillierung
16#7000	Kein Auftrag in Bearbeitung (Initialwert)
16#7001	Erster Aufruf nach Eingang eines neuen Auftrags (steigende Flanke "execute")
16#7002	Folgeaufruf
16#8600	Interner Zustandsautomat hat einen undefinierten Zustand erreicht
16#8602	Systemfunktion RDREC meldet einen Fehler, siehe "diagnostics"
16#8603	Fehler: der manuell eingegebene CAP wird nicht unterstützt; nur CAP=227 oder CAP=0xB400 (-19456) werden unterstützt

diagnostics

Der Ausgang "diagnostics" gibt im Fehlerfall detaillierte Informationen zu dem vorliegenden Fehler aus.

Tabelle 3-6: Ausgang "diagnostics" von LIOLink_Diagnose

Variable	Beschreibung
status	Letzter Statuscode des Schnittstellenparameters "status" des FB.
subfunctionStatus	Status von Systemfunktionen RDREC/WRREC oder Fehlercode vom IO-Link Master (%W1: Fehlercode vom IO-Link Master, %W0: ISDU-Fehlercode). Detaillierte Informationen entnehmen Sie der Online-Hilfe zur jeweiligen Systemfunktion bzw. dem Gerätehandbuch des IO-Link Masters/Devices.
stateNumber	Zustand des Zustandsautomaten des FB, in dem der Fehler aufgetreten ist.

3.2.5. Events

Der Ausgangsparameter "events" liefert die letzten laufenden 5 Events für jeden Port des IO-Link Masters. Diese Events basieren dabei auf dem IO-Link-Standard und können entweder Events des Ports oder des Devices sein. Unterschieden wird dies durch das Bit "source" in [Tabelle 3-10: "LIOLink_typePortEventQualifier"](#).

Grundsätzlich ist der Parameter "events" durch verschiedene UDT's untergliedert, die in den folgenden Tabellen hierarchisch aufgeführt sind. Dem Ausgangsparameter ist zunächst der UDT "LIOLink_typeEvents" zugewiesen. Dieser besitzt ein Array, das die Events aller 8 Ports beinhaltet. (vgl. [Tabelle 3-7: "LIOLink_typeEvents"](#))

Tabelle 3-7: "LIOLink_typeEvents"

Variable	Datentyp	Beschreibung
port	Array[1..8] of LIOLink_typePortEvents	Beinhaltet die Events aller 8 Ports.

Der UDT "LIOLink_typePortEvents" in [Tabelle 3-8](#) beinhaltet wiederum ein Array mit 5 Feldern zur Speicherung der letzten 5 Events.

Tabelle 3-8: "LIOLink_typePortEvents"

Variable	Datentyp	Beschreibung
event	Array[0..4] of LIOLink_typePortEventCodes	Beschreibt die letzten 5 aufgetretenen Events

Jedes Event wird dabei durch einen "eventCode" und durch einen "eventQualifier" beschrieben (vgl. [Tabelle 3-9](#)). Definiert ist dies durch die IO-Link-Spezifikation.

Mögliche Event-Codes sind in [Tabelle 3-11](#) und in [Tabelle 3-12](#) aufgeführt.

Tabelle 3-9: "LIOLink_typePortEventCodes"

Variable	Datentyp	Beschreibung
eventCode	Word	IO-Link EventCode
eventQualifier	LIOLink_typePortEventQualifier	IO-Link EventQualifier

EventQualifier besteht aus den Variablen "instance", "source", "type" und "mode". Dabei gibt "instance" die spezifische Quelle (Instanz) eines Events an und verfeinern somit dessen Bewertung auf der Empfängerseite. Zusätzlich gibt die Variable "source" die Quelle des Ereignisses an, "type" die Event-Kategorie und mode den Event-Modus. Spezifiziert sind diese EventQualifier in der IO-Link-Spezifikation.

Tabelle 3-10: "LIOLink_typePortEventQualifier"

Variable	Datentyp	Beschreibung
instance	Byte	0: Unknown / 1-3: reserved / 4: Application / 5-7: reserved
source	Bool	0: Device (remote) / 1: Master/Port

Variable	Datentyp	Beschreibung
type	Byte	0: reserved / 1: Notification / 2: Warning / 3: Error
mode	Byte	0: reserved / 1: Event single shot / 2: Event disappears / 3: Event appears

HINWEIS

Abhängig vom "events.Port[x].event[y].eventQualifier.source"-Bit gibt es zwei verschiedene Tabellen zur Definition Event-Codes für IO-Link-Devices und Event-Codes für die Ports des Masters.

3.2.6. Event-Codes für IO-Link Devices

[Tabelle 3-11](#) listet alle Event-Codes auf, die für die IO-Link Devices definiert sind. Das "source" Bit (vgl. [Tabelle 3-10](#)) ist demzufolge nicht gesetzt. Die Auflistung ist aus der IO-Link-Spezifikation V1.1.3 entnommen. Herstellerspezifische Ereigniscodes sind in der IO-DD der jeweiligen Geräte unter den Event Codes zu finden: 0x8CA0 - 0x8DFF.

Tabelle 3-11: Event-Codes für IO-Link Devices

EventCode	Definition und empfohlene Wartungsmaßnahme	Event-Kategorie
0x0000	No malfunction	Notification
0x1000	General malfunction – unknown error	Error
0x1001 - 0x17FF	Reserved	
0x1800 - 0x18FF	Vendor specific	
0x1900 - 0x3FF	Reserved	
0x4000	Temperature fault – Overload	Error
0x4001 - 0x420F	Reserved	
0x4210	Device temperature overrun – Clear source of heat	Warning
0x4211 - 0x421F	Reserved	
0x4220	Device temperature underrun – Insulate Device	Warning
0x4221 - 0x4FFF	Reserved	
0x5000	Device Hardware fault – Device exchange	Error
0x5001 - 0x500F	Reserved	
0x5010	Component malfunction – Repair of exchange	Error
0x5011	Non volatile memory loss – Check batteries	Error
0x5012	Batteries low -Exchange batteries	Warning
0x5013 - 0x50FF	Reserved	
0x5100	General power supply fault – Check availability	Error
0x5101	Fuse blown/open – Exchange fuse	Error
0x5102 - 0x510F	Reserved	
0x5110	Primary supply voltage overrun – Check tolerance	Warning
0x5111	Primary supply voltage underrun – Check tolerance	Warning
0x5112	Secondary supply voltage fault (Port Class B) – Check tolerance	Warning
0x5113 - 0x5FFF	Reserved	
0x6000	Device software fault – Check firmware version	Error
0x6001 - 0x631F	Reserved	
0x6320	Parameter error – Check data sheet and values	Error
0x6321	Parameter missing – Check data sheet	Error
0x6322 - 0x634F	Reserved	
0x6350	Reserved	
0x6351 - 0x76FF	Reserved	
0x7700	Wire break of a subordinate device – Check installation	Error
0x7701 - 0x770F	Wire break of subordinate device 1 ...device 15 – Check installation	Error
0x7710	Short circuit – Check installation	Error
0x7711	Ground fault – Check installation	Error

EventCode	Definition und empfohlene Wartungsmaßnahme	Event-Kategorie
0x7712 - 0x8BFF	Reserved	
0x8C00	Technology specific application fault – Reset Device	Error
0x8C01	Simulation active – Check operational mode	Warning
0x8C01 - 0x8C0F	Reserved	
0x8C10	Process variable range overrun – Process Data uncertain	Warning
0x8C11 - 0x8C1F	Reserved	
0x8C20	Measurement range exceeded – Check application	Error
0x8C21 - 0x8C2F	Reserved	
0x8C30	Process variable range underrun – Process Data uncertain	Warning
0x8C31 - 0x8C3F	Reserved	
0x8C40	Maintenance required – Cleaning	Warning
0x8C41	Maintenance required – Refill	Warning
0x8C42	Maintenance required – Exchange wear and tear parts	Warning
0x8C43 - 0x8C9F	Reserved	
0x8CA0 - 0x8DFF	Vendor specific	
0x8E00 - 0xAFFF	Reserved	
0xB000 - 0xB0FF	Reserved for Safety extensions	
0xB100 - 0xBFFF	Reserved for profiles	
0xC000 - 0xFF90	Reserved	
0xFF91	Data Storage upload request ("DS_UPLOAD_REQ") – internal, not visible to user	Notification (single shot)
0xFF92 - 0xFFAF	Reserved	
0xFFB0 - 0xFFB7	Reserved for Wireless extensions	
0xFFB8 - 0xFFFF	Reserved	

3.2.7. Event-Codes für die IO-Link-Master Ports

[Tabelle 3-12](#) listet alle Event-Codes auf, die für die Ports des IO-Link Masters definiert sind. Das "source" Bit (vgl. [Tabelle 3-10](#)) ist demzufolge gesetzt. Die Auflistung ist aus der IO-Link-Spezifikation V1.1.3 entnommen.

Tabelle 3-12: Event-Codes für IO-Link-Master Ports

EventCode	Definition und empfohlene Wartungsmaßnahme	Event-Kategorie
0x0000 to 0x17FF	Reserved	
0x1800	No device (communication)	Error
0x1801	Startup parametrization error – check parameter	Error
0x1802	Incorrect VendorID – Inspection Level mismatch	Error
0x1803	Incorrect DeviceID – Inspection Level mismatch	Error
0x1804	Short circuit at C/Q – check wire connection	Error
0x1805	PHY overtemperature – check Master temperature and load	Error
0x1806	Short circuit at L+ - check wire connection	Error
0x1807	Overcurrent at L+ - check power supply (e.g. L1+)	Error
0x1808	Device Event overflow	Error
0x1809	Backup inconsistency – memory out of range (2048 octets)	Error
0x180A	Backup inconsistency – identity fault	Error
0x180B	Backup inconsistency – Data Storage unspecific error	Error
0x180C	Backup inconsistency – upload fault	Error
0x180D	Parameter inconsistency - download fault	Error
0x180E	P24 (Class B) missing or undervoltage	Error
0x180F	Short circuit at P24 (Class B) – check wire connection (e.g. L2+)	Error

EventCode	Definition und empfohlene Wartungsmaßnahme	Event-Kategorie
0x1810	Short circuit at I/Q – check wiring	Error
0x1811	Short circuit at C/Q (if digital output) – check wiring	Error
0x1812	Overcurrent at I/Q – check load	Error
0x1813	Overcurrent at C/Q (if digital output) – check load	Error
0x1814 to 0x1EFF	Reserved	
0x1F00	Vendor specific	
0x2000 to 0x2FFF	Safety extensions	
0x3000 to 0x3FFF	Wireless extensions	
0x4000 to 0x5FFF	Reserved	
0x6000	Invalid cycle time	Error
0x6001	Revision fault – incompatible protocol version	Error
0x6002	ISDU batch failed – parameter inconsistency?	Error
0x6003 to 0xFF20	Reserved	
0xFF21	DL: Device plugged in ("NEW SLAVE") – PD stop	Notification
0xFF22	Device communication lost	Notification
0xFF23	Data Storage identification mismatch	Notification
0xFF24	Data Storage buffer overflow	Notification
0xFF25	Data Storage parameter access denied	Notification
0xFF26	Port status changed	Notification
0xFF27	Data Storage upload completed and new data object available	Notification
0xFF28 to 0xFF30	Reserved	
0xFF31	DL: Incorrect Event signalling	Notification
0xFF32 to 0xFFFF	Reserved	

4. Master-spezifische Bausteine

4.1. LIOLink_Master

4.1.1. Beschreibung

Mithilfe dieses Bausteins können Sie die Geräteparameter und -einstellungen eines IO-Link Masters über das S7-Programm sichern (Backup) und wiederherstellen (Restore) (Mastertausch ohne Engineering Tool).

HINWEIS Dieser Baustein ersetzt die bisher verfügbaren Bausteine "IO_LINK_MASTER_4" und "IO_LINK_MASTER_8".

Gültigkeit

Der Baustein ist für die folgenden IO-Link Master freigegeben:

- ET 200SP, CM 4xIO-Link (6ES7137-6BD00-0BA0) ab FW V2.2.0
- ET 200pro, EM 4 IO-LINK HF (6ES7147-4JD00-0AB0) ab FW V1.0.0
- ET 200AL, CM 4xIO-Link (6ES7147-5JD00-0BA0) ab FW V1.1.0
- ET 200eco PN, IO-Link Master (6ES7148-6JD00-0AB0) ab FW V1.0.0
- ET 200eco PN, CM 8x IO-Link, M12-L (6ES7148-6JG00-0BB0) ab FW V1.0.0
- ET 200eco PN, CM 4x IO-Link + DIQ 12x 24VDC (6ES7148-6JE00-0BB0) ab FW V1.0.0
- ET 200eco PN, CM 8x IO-Link + DIQ 8x 24VDC (6ES7148-6JJ00-0BB0) ab FW V1.0.0
- ET 200clean, CM 8x IO-Link + DIQ 4x 24V DC (6ES7148-7JH00-0BB0) ab FW V1.1.0
- S7-1200, SM 1278 4 IO-Link (6ES7278-4BD32-0XB0) ab FW V2.0.0
- S7-1500, CM 8xIO-Link (6ES7547-1JF00-0AB0) ab FW V1.0.0

HINWEIS Die Funktion "Master Backup" ist nur für IO-Link Devices verfügbar, die für den IO-Link Standard ab V1.1 spezifiziert sind.
IO-Link Master von Drittanbietern werden durch diesen Baustein nicht unterstützt.

Der FB steht in zwei Varianten für S7-1200/1500 und S7-300/400 zur Verfügung. Dabei existieren geringfügige Unterschiede. (vgl. [Tabelle 3-1: Parameter von LIOLink Device](#))

4.1.2. Parameter

Abbildung 4-1: LIOLink_Master

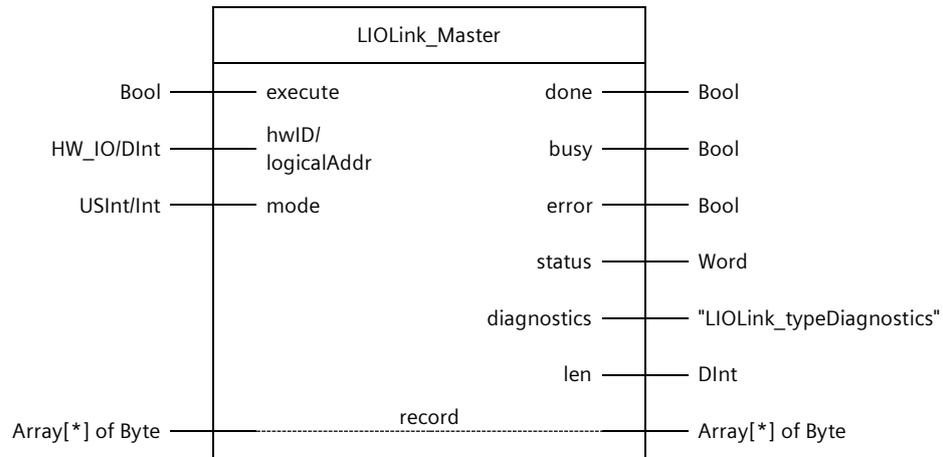


Tabelle 4-1: Parameter von LIOLink_Master

Name	P-Typ	Datentyp	Kommentar
execute	IN	Bool	Anforderung, die Funktion auszuführen
hwID/ logicalAddr	IN	HW_IO/DInt	S7-1200/1500: Hardwarekennung des IO-Link Masters oder des ersten Submoduls. S7-300/400: Logische Adresse des IO-Link Masters (Modul oder Submodul)
mode	IN	USInt/Int	Modus 0: Backup 1: Restore
done	OUT	Bool	TRUE: Auftrag wurde erfolgreich ausgeführt
busy	OUT	Bool	TRUE: Auftrag wird gerade bearbeitet
error	OUT	Bool	TRUE: Ein Fehler ist bei der Bearbeitung des FB aufgetreten
status	OUT	Word	16#0000–16#7FFF: Status des FB 16#8000–16#FFFF: Fehlercodes (siehe Kapitel 4.1.4)
diagnostics	OUT	"LIOLink_ typeDiagnostics"	Detaillierte Diagnoseinformationen des FB (siehe Kapitel 4.1.4)
len	OUT	DInt	Länge der gelesenen/geschriebenen Daten in Bytes (Nettodaten)
record	IN_OUT	Array[*] of Byte	Quell-/Zielbereich für die zu lesenden/schreibenden Daten. Zulässige Grenzen: <ul style="list-style-type: none"> • 4-Port-Master: 0..10239 • 8-Port-Master: 0..17549

4.1.3. Funktionsweise

Die Datenübertragung wird mit einer positiven Flanke am Eingang "execute" gestartet.

Der Baustein arbeitet asynchron, d. h. die Bearbeitung erstreckt sich über mehrere SPS-Zyklen. Grundsätzlich nutzt der Baustein eine Sequenz von azyklischen Lese-/Schreibzugriffen, die durch die Systemfunktionen "RDREC" und "WRREC" erfolgen.

Die Ausgänge "done", "busy", "error" und "status" zeigen den Zustand des Auftrags an.

Nach erfolgreicher Ausführung zeigt der Parameter "len" die Länge der empfangenen oder geschriebenen Daten an.

HINWEIS Um möglichst wenig Ressourcen zu belegen, arbeitet der Baustein direkt in dem Speicherbereich am Parameter "record". Die Daten in dem Speicherbereich dürfen sich während eines Schreibvorgangs nicht ändern und dürfen erst nach Abschluss eines Lesevorgangs gelesen werden. Werten Sie dafür den Ausgangsparameter "done" oder "status" aus.

Solange der Eingang "execute" gesetzt ist, behalten die Ausgangsparameter ihren Wert bei. Wird der Eingang "execute" zurückgesetzt, bevor die Bearbeitung des FB abgeschlossen ist, werden die Werte der Ausgangsparameter nach der Bearbeitung des Auftrages für einen Zyklus ausgegeben.

HINWEIS Die Datenübertragung erfolgt in Form von Rohdaten (Array of Byte), d. h. die Daten können nicht interpretiert werden und dürfen nicht verändert bzw. manipuliert werden!
Bevor ein Datensatz geschrieben werden kann, muss dieser erst gesichert werden.

HINWEIS Wenn ein DPV1-Slave über GSD-Datei projiziert ist (GSD ab Rev. 3) und die DP-Schnittstelle des DP-Master auf "S7-kompatibel" eingestellt ist, arbeitet der Baustein "LIOLink_Master" nicht korrekt. Stellen Sie die Schnittstelle des DP-Masters stattdessen auf "DPV1".

ACHTUNG **Datenverlust**
Die Kombination einer SIMATIC S7-1200 CPU mit einem 8-Port IO-Link Master übersteigt die Höhe der remanenten Daten, die in der CPU zur Verfügung stehen.
Falls diese Kombination zum Einsatz kommt, führt dies zu Datenverlust, da die Daten nicht vollständig remanent in der CPU gespeichert werden können.

4.1.4. Fehlerhandling

Der Ausgang "status" gibt den aktuellen Status sowie Fehler aus, während der Ausgang "diagnostics" im Fehlerfall eine Diagnosestruktur mit detaillierten Informationen bereitstellt.

status

Tabelle 4-2: Ausgang "status" von LIOLink_Master

Status	Bedeutung
16#0000	Auftrag abgeschlossen, keine Warnung und keine weitere Detaillierung
16#7000	Kein Auftrag in Bearbeitung (Initialwert)
16#7001	Erster Aufruf nach Eingang eines neuen Auftrags (steigende Flanke "execute")
16#7002	Folgeaufruf während aktiver Bearbeitung ohne weitere Detaillierung
16#71xx	Folgeaufruf während Backup, xx = aktuelle Sequenznummer
16#72xx	Folgeaufruf während Restore, xx = aktuelle Sequenznummer
16#80B0	Unbekannter Modultyp
16#8200	Nicht-unterstützter Wert am Parameter "mode"
16#8201	Array am Parameter "record" entspricht nicht den erwarteten Grenzen
16#8401	IO-Link Master hat eine Sequenznummer zurückgemeldet, die einen Fehler indiziert, siehe "diagnostics"
16#8600	Interner Zustandsautomat hat einen undefinierten Zustand erreicht
16#8601	Systemfunktion WRREC meldet einen Fehler während des Resets, siehe "diagnostics"
16#8602	Systemfunktion RDREC meldet einen Fehler während des Backups, siehe "diagnostics"
16#8603	Systemfunktion WRREC meldet einen Fehler während des Restores, siehe "diagnostics"

Status	Bedeutung
16#8604	Systemfunktion RDREC meldet einen Fehler während der Verifikation des Restores, siehe "diagnostics"

diagnostics

Der Ausgang "diagnostics" gibt im Fehlerfall detaillierte Informationen zu dem vorliegenden Fehler aus.

Tabelle 4-3: Ausgang "diagnostics" von LIOLink_Master

Variable	Beschreibung
status	Letzter Statuscode des Schnittstellenparameters "status" des FB.
subfunctionStatus	Status von Systemfunktionen RDREC/WRREC oder Fehlercode vom IO-Link Master. Detaillierte Informationen entnehmen Sie der Online-Hilfe zur jeweiligen Systemfunktion bzw. dem Gerätehandbuch des IO-Link Masters.
stateNumber	Zustand des Zustandsautomaten des FB, in dem der Fehler aufgetreten ist.

Tabelle 4-4: Ausgang "subfunctionStatus" Fehlercode von LIOLink_Master

subfunctionStatus	Bedeutung
16#FFFF_FF01	Wiederherstellen (Restore) läuft
16#FFFF_FF02	Aktivierung läuft
16#FFFF_FF03	Aktivierung fertig gestellt
16#FFFF_FF04	CRC check fehlgeschlagen
16#FFFF_FF05	Blob- Daten ungültig

4.2. LIOLink_Toolchanger

4.2.1. Beschreibung

Über Portfunktionen kann eine Portdiagnose oder Fehlermeldung je nach Anwendungsfall ausgesetzt und wieder aufgenommen werden.

Der Funktionsbaustein unterstützt Sie bei folgenden Aufgaben:

- "Port-Betrieb suspendieren"
- "Port-Betrieb wieder aufnehmen"

Das Konzept hinter "Port-Betrieb suspendieren" ist es, die gesamten Störungsmeldungen an das System zu unterdrücken, da es sich um eine beabsichtigte Aktion handelt. Grundsätzlich werden nach der Suspendierung alle Fehlermeldungen des betreffenden Ports und Devices gelöscht. In diesem Zustand wird davon ausgegangen, dass das Anwenderprogramm weiterhin Prozessdaten und Port Qualifier Information (PQI) verarbeitet und diese daher weiterhin aktiv sind und aktualisiert werden. Ein Device kann jedoch im suspendierten Zustand von einem Port entfernt werden, ohne bestimmte Ereignisse oder Diagnosen auszulösen. Die Systemdiagnose wird keinen Fehler melden.

Das Konzept hinter "Port-Betrieb wieder aufnehmen" besteht darin, den Diagnosemechanismus wieder aufzunehmen. Die aktuelle Diagnose wird wiederhergestellt und aktualisiert.

Ein möglicher Anwendungsfall kann eine Maschine sein, die ein bestimmtes Werkzeug, z. B. einen Greifer, in einem Magazin abdockt und ein anderes andockt. Dieses An- und Abdocken kann Ereignisse und Diagnosen im System oder dem Anwenderprogramm auslösen, da die Kommunikation und die Stromversorgung des Geräts oder Werkzeugs unterbrochen werden. Die Unterdrückung der Diagnose vor dem Abdocken des "vorherigen" Werkzeugs und die Wiederaufnahme der Diagnose nach dem Andocken des "neuen" Werkzeugs beseitigt dieses Problem während des Betriebs.

HINWEIS

Weitere Informationen zu Tool Changer Use Case und Portfunktionen finden Sie unter [IO-Link Integration – Profile for Profinet](#).

Gültigkeit

Der Baustein ist für die folgenden IO-Link Master freigegeben:

- ET 200eco PN, CM 8x IO-Link, M12-L (6ES7148-6JG00-0BB0) ab FW V5.2.0
- ET 200eco PN, CM 4x IO-Link + DIQ 12x 24VDC (6ES7148-6JE00-0BB0) ab FW V5.1.0
- ET 200eco PN, CM 8x IO-Link + DIQ 8x 24VDC (6ES7148-6JJ00-0BB0) ab FW V5.1.0
- ET 200clean, CM 8x IO-Link + DIQ 4x 24V DC (6ES7148-7JH00-0BB0) ab FW V1.1.0

4.2.2. Parameter

Abbildung 4-2: LIOLink_Toolchanger

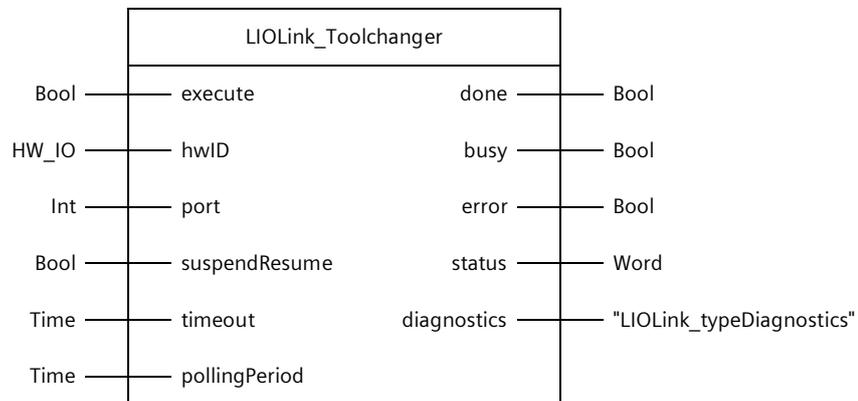


Tabelle 4-5: Parameter von LIOLink_Toolchanger

Name	P-Typ	Datentyp	Kommentar
execute	IN	Bool	Anforderung, die Funktion auszuführen
hwID	IN	HW_IO	S7-1200/1500: Hardwarekennung des IO-Link Masters oder des ersten Submoduls.
port	IN	Int	Portnummer an dem das IO-Link Device betrieben wird
suspendResume	IN	Bool	Portfunktion; FALSE: Portbetrieb suspendieren, TRUE: Portbetrieb wieder aufnehmen
timeout	IN	Time	Zeit, nach der ein Auftrag abgebrochen wird Defaultwert: 20s
pollingPeriod (optional)	IN	Time	Variabel einstellbare Zeit, die der Baustein wartet, bis der Datensatz übermittelt wird Defaultwert: 100 ms
done	OUT	Bool	TRUE: Auftrag wurde erfolgreich ausgeführt
busy	OUT	Bool	TRUE: Auftrag wird gerade bearbeitet
error	OUT	Bool	TRUE: Ein Fehler ist bei der Bearbeitung des FB aufgetreten
status	OUT	Word	16#0000–16#7FFF: Status des FB 16#8000–16#FFFF: Fehlercodes (siehe Kapitel 3.1.4)
diagnostics	OUT	"LIOLink_typeDiagnostics"	Detaillierte Diagnoseinformationen des FB (siehe Kapitel 3.1.4)

HINWEIS

Über den Eingang "suspendResume" wird der Betrieb eines bestimmten Ports (Eingang "port") unterbrochen oder wiederaufgenommen. Es ist zu beachten, dass die Prozessdaten und die PQI auch nach dem Suspendieren eines Ports noch aktiv sind und aktualisiert werden. Wird ein Gerät entfernt, sind nach dem Suspendieren des Ports die Prozessdaten leer oder 0 und PQI zeigt entsprechend an, dass die IO-Link Daten ungültig sind.

Es wird dem Benutzer überlassen, die Prozessdaten im Falle einer Unterbrechung/Wiederaufnahme entsprechend zu behandeln. Der FB [LIOLink MeasuredData](#) ist ein Beispiel, wie man das machen kann.

4.2.3. Status und Fehlerhandling

Der Ausgang "status" gibt den aktuellen Status sowie Fehler aus, während der Ausgang "diagnostics" im Fehlerfall eine Diagnosestruktur mit detaillierten Informationen bereitstellt.

status

Tabelle 4-6: Ausgang "status" von LIOLink_Toolchanger

Status	Bedeutung
16#0000	Auftrag abgeschlossen, keine Warnung und keine weitere Detaillierung im Ausgang "diagnostics"
16#7000	Kein Auftrag in Bearbeitung (Initialwert)
16#7001	Erster Aufruf nach Eingang eines neuen Auftrags (steigende Flanke "execute")
16#7002	Folgeaufruf
16#7003	Suspendieren wurde erfolgreich ausgeführt
16#7004	Wiederaufnahme wurde erfolgreich ausgeführt
16#8201	Nicht-unterstützter Port
16#8401	IO-Link Master hat einen Fehlercode zurückgemeldet, siehe "diagnostics"
16#8402	Empfangener Datensatz passt nicht zum Auftrag
16#8403	Auftrag konnte nicht in der vorgegebenen Zeit abgeschlossen werden
16#8600	Interner Zustandsautomat hat einen undefinierten Zustand erreicht
16#8601	Systemfunktion WRREC meldet einen Fehler, siehe "diagnostics"
16#8602	Systemfunktion RDREC meldet einen Fehler, siehe "diagnostics"

diagnostics

Der Ausgang "diagnostics" gibt im Fehlerfall detaillierte Informationen zu dem vorliegenden Fehler aus.

Tabelle 4-7: Ausgang "diagnostics" von LIOLink_Toolchanger

Variable	Beschreibung
status	Letzter Statuscode des Schnittstellenparameters "status" des FB.
subfunctionStatus	Status von Systemfunktionen RDREC/WRREC oder Fehlercode vom IO-Link Master (%W1: Fehlercode vom IO-Link Master, %W0: ISDU-Fehlercode). Detaillierte Informationen entnehmen Sie der Online-Hilfe zur jeweiligen Systemfunktion bzw. dem Gerätehandbuch des IO-Link Masters/Devices.
stateNumber	Zustand des Zustandsautomaten des FB, in dem der Fehler aufgetreten ist.

5. Geräte-spezifische Bausteine

Die Geräte-spezifischen Bausteine dieser Bibliothek vereinfachen die Kommunikation mit jeweils einem bestimmten IO-Link Device durch eine darauf zugeschnittene Schnittschnelle und vordefinierte PLC-Datentypen.

Der Anwender muss dabei nicht die notwendigen Parameter kennen, um einen bestimmten Datensatz zu lesen oder zu schreiben, sondern wählt diesen über die Schnittstelle aus.

Für jeden unterstützten Datensatz werden PLC-Datentypen bereitgestellt, die ein einfaches Lesen bzw. Schreiben des Datensatzes ermöglichen.

5.1. SIMATIC RF200

Mit den hier beschriebenen Bausteinen der Bibliothek können Sie den SIMATIC RF200 IO-Link Reader (V1.0/V1.1) über eine einfache Anwenderschnittstelle ansteuern.

Der FB "LIOLink_RF200" ermöglicht das Lesen und Schreiben diverser Datensätze, während die anderen FBs Funktionen zum Lesen eines Transponders, Schreiben eines Transponders und Ein-/Ausschalten des Antennenfelds am Readers bieten.

Gültigkeit

Diese FBs stehen jeweils in zwei Varianten für S7-1200/1500 und S7-300/400 zur Verfügung.

5.1.1. LIOLink_RF200

Beschreibung

Der FB "LIOLink_RF200" ermöglicht das Lesen und Schreiben diverser Datensätze über eine einfache Anwenderschnittstelle.

Parameter

Abbildung 5-1: LIOLink_RF200

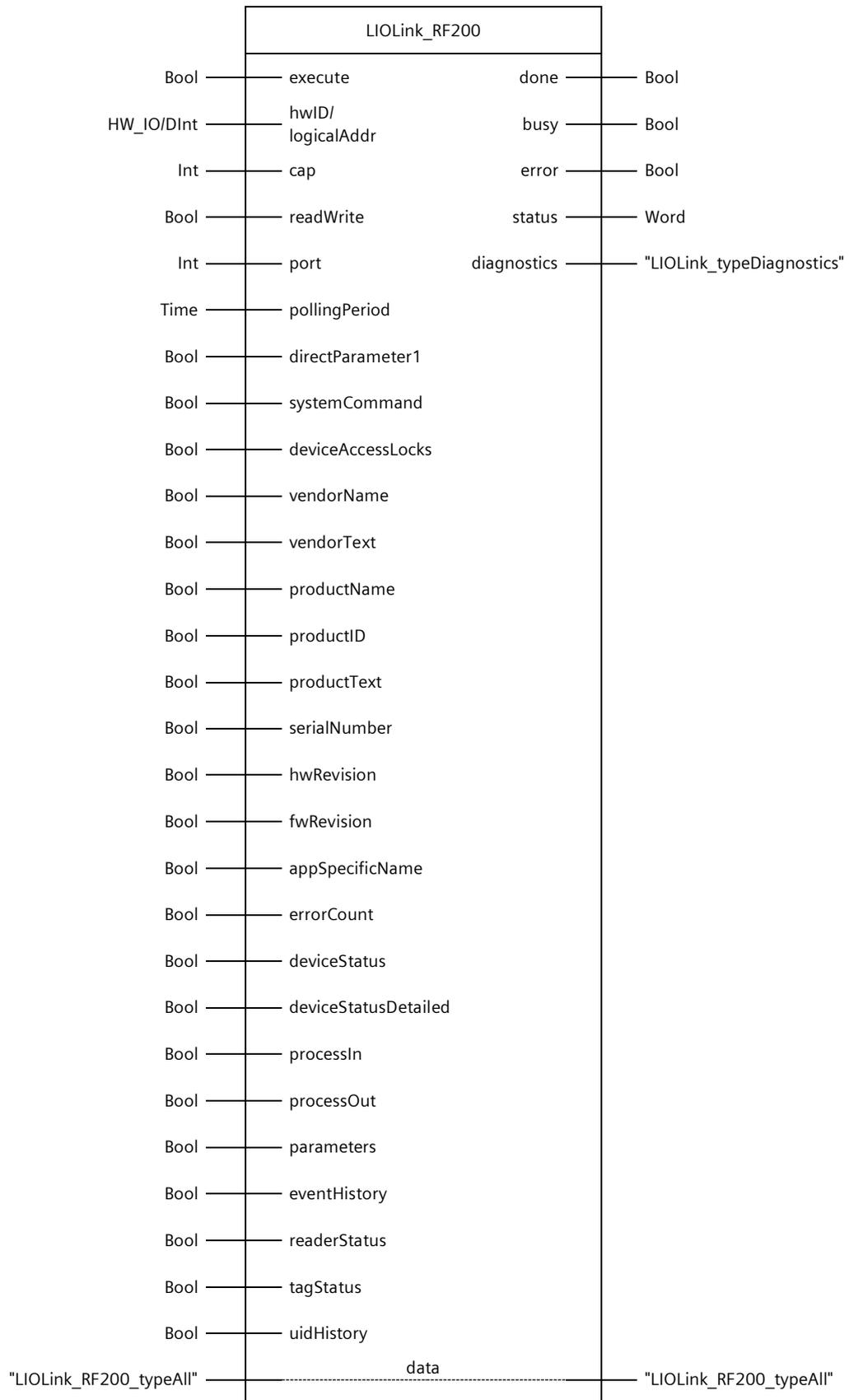


Tabelle 5-1: Parameter von LIOLink_RF200

Name	P-Typ	Datentyp	Kommentar
execute	IN	Bool	Anforderung, die Funktion auszuführen
hwID/logicalAddr	IN	HW_IO/DInt	S7-1200/1500: Hardwarekennung des IO-Link Masters oder des ersten Submoduls. S7-300/400: Logische Adresse des IO-Link Masters (Modul oder Submodul)
cap (optional)	IN	Int	Zugangspunkt (Client Access Point): S7-1200/1500: Wird bei der Verwendung von Siemens IO-Link Mastern automatisch erkannt und ist daher optional. Wird der Parameter verändert, ist die automatische Erkennung nicht aktiv. S7-300/400: Die automatische Erkennung ist in dieser Version nicht implementiert. Der Zugangspunkt muss daher manuell gesetzt werden. Für gewöhnlich besitzt er den Wert 0xB400 oder 0x00E3 und ist im jeweiligen Handbuch angegeben.
readWrite	IN	Bool	Modus FALSE: lesen TRUE: schreiben
port	IN	Int	Portnummer an dem das IO-Link Device betrieben wird Mögliche Werte: 0..63
pollingPeriod	IN	Time	S7-1200/1500: Variabel einstellbare Zeit, die der Baustein wartet, bis der Datensatz übermittelt wird. Defaultwert: 100 ms S7-300/400: Die Funktionalität ist in dieser Version nicht implementiert, weswegen der Parameter hier nicht existiert.
directParameter1	IN	Bool	Lies "Direct parameter page 1"
systemCommand	IN	Bool	Schreibe Systemkommando: Device- Reset: 0x80 Rücksetzen auf Werkseinstellung: 0x82
deviceAccessLocks	IN	Bool	Lies/schreib Sperrfunktionen für Gerätezugriff
vendorName	IN	Bool	Lies Herstellername
vendorText	IN	Bool	Lies Herstellertext
productName	IN	Bool	Lies Produktname
productID	IN	Bool	Lies Produkt ID
productText	IN	Bool	Lies Produkt-Text
serialNumber	IN	Bool	Lies Seriennummer
hwRevision	IN	Bool	Lies Hardwareversion
fwRevision	IN	Bool	Lies Firmwareversion
appSpecificName	IN	Bool	Lies/schreib Anwenderspezifische Markierung
errorCount	IN	Bool	Lies Fehlerzähler
deviceStatus	IN	Bool	Lies Geräte-Status
deviceStatusDetailed	IN	Bool	Lies detaillierten Geräte-Status
processIn	IN	Bool	Lies letztes Prozessabbild der Eingänge

Name	P-Typ	Datentyp	Kommentar
processOut	IN	Bool	Lies letztes Prozessabbild der Ausgänge
parameters	IN	Bool	Lies Parameter
eventHistory	IN	Bool	Lies Ereignisverlauf
readerStatus	IN	Bool	Lies Reader-status
tagStatus	IN	Bool	Lies Transponder-Status
uidHistory	IN	Bool	Lies UID-Verlauf
done	OUT	Bool	TRUE: Auftrag wurde erfolgreich ausgeführt
busy	OUT	Bool	TRUE: Auftrag wird gerade bearbeitet.
error	OUT	Bool	TRUE: Ein Fehler ist bei der Bearbeitung des FB aufgetreten
status	OUT	Word	Status von LIOLink_Device (siehe Kapitel 3.1.3)
diagnostics	OUT	"LIOLink_typeDiagnostics"	Diagnoseinformationen von LIOLink_Device (siehe Kapitel 3.1.3)
data	IN_OUT	"LIOLink_RF200_typeAll"	Geräte-spezifischer Ziel-/Quellbereich

HINWEIS

Pro Auftrag kann nur ein Datensatz gelesen/geschrieben werden. Der FB liest/schreibt den zum ersten gesetzten Parameter zugehörigen Datensatz.

Fehlerhandling

Der Baustein gibt die Status- und Diagnoseinformationen vom intern aufgerufenen FB "LIOLink_Device" an den Ausgängen "status" und "diagDeviceFB" aus (siehe Kapitel [3.1.3](#)).

5.1.2. LIOLink_RF200_ReadTag**Beschreibung**

Der FB "LIOLink_RF200_Read" liest einen Datenblock vom Transponder.

Parameter

Abbildung 5-2: LIOLink_RF200_ReadTag

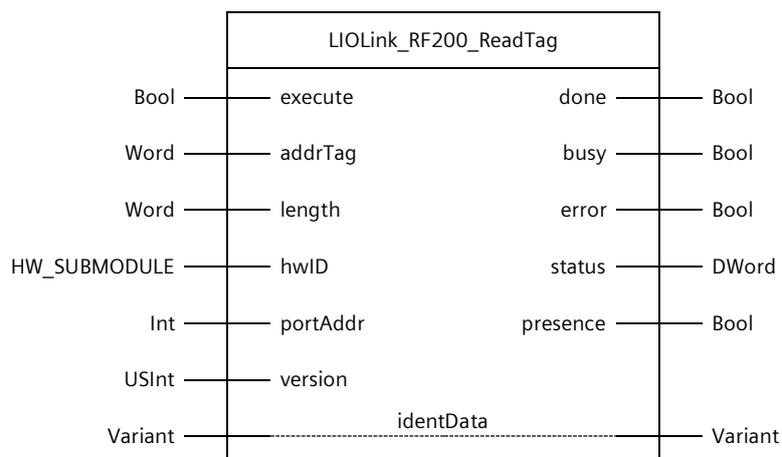
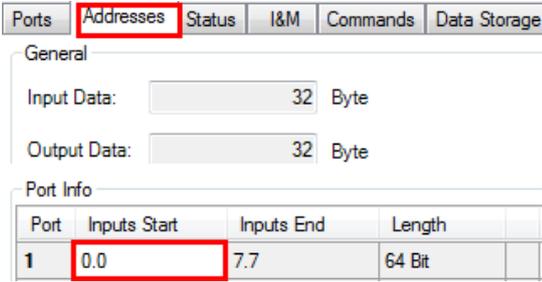


Tabelle 5-2: Parameter von LIOLink_RF200_ReadTag

Name	P-Typ	Datentyp	Kommentar
execute	IN	Bool	Aktiviert den Leseauftrag bei positiver Flanke.
addrTag	IN	Word	Anfangsadresse der zu lesenden Daten auf dem Transponder

Name	P-Typ	Datentyp	Kommentar
length	IN	Word	Länge der Daten, die vom Transponder gelesen werden
hwID	IN	HW_SUBMODULE	Hardware-Kennung des IO-Link Masters
portAddr	IN	Int	Anfangsadresse des angeschlossenen Readers (PCT Tool).  <p>Beispiel: Anfangsadresse 0.0, muss bei "portAddr" der Wert "0" eingetragen werden.</p>
version	IN	USInt	IO-Link-Version des Readers 11: IO-Link-Version 1.1 10: IO-Link-Version 1.0
done	OUT	Bool	TRUE: Auftrag wurde erfolgreich ausgeführt
busy	OUT	Bool	TRUE: Auftrag wird gerade bearbeitet
error	OUT	Bool	TRUE: Ein Fehler ist bei der Bearbeitung des FB aufgetreten
status	OUT	DWord	Fehlercode im Fehlerfall, siehe Kapitel 5.1.6 .
presence	OUT	Bool	Anwesenheits-Bit Dieses Bit wird nur gesetzt, wenn sich ein Transponder im Feld des Readers befindet
identData	IN_OUT	Variant	Datenbereich, in den die gelesenen Daten geschrieben werden.

5.1.3. LIOLink_RF200_SwitchAntenna

Beschreibung

Der FB "LIOLink_RF200_Antenna" schaltet die Antenne eines SIMATIC RF200 IO-Link- Readers ein oder aus.

HINWEIS Im Normalbetrieb wird dieser Befehl nicht benötigt, da nach dem Einschalten des Readers die Antenne immer eingeschaltet ist.

Parameter

Abbildung 5-3: LIOLink_RF200_SwitchAntenna

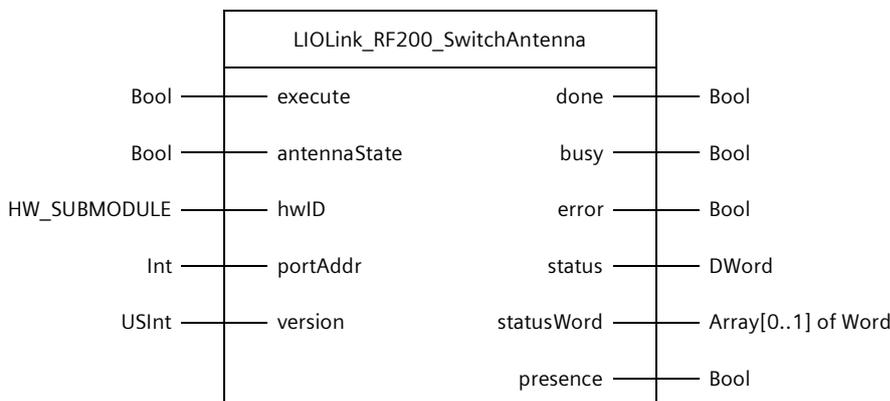
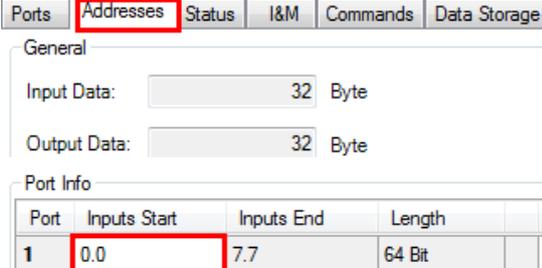


Tabelle 5-3: Parameter von LIOLink_RF200_SwitchAntenna

Name	P-Typ	Datentyp	Kommentar
execute	IN	Bool	Anforderung, die Funktion auszuführen
antennaState	IN	Bool	Antennenzustand TRUE: Aktiviere Antenne FALSE: Deaktiviere Antenne
hwID	IN	HW_SUBMODULE	Hardware-Kennung des IO-Link Masters
portAddr	IN	Int	Anfangsadresse des angeschlossenen Readers (PCT Tool). 
version	IN	USInt	IO-Link-Version des Readers 11: IO-Link-Version 1.1 10: IO-Link-Version 1.0
done	OUT	Bool	TRUE: Auftrag wurde erfolgreich ausgeführt
busy	OUT	Bool	TRUE: Auftrag wird gerade bearbeitet
error	OUT	Bool	TRUE: Ein Fehler ist bei der Bearbeitung des FB aufgetreten
status	OUT	DWord	Fehlercode im Fehlerfall, siehe Kapitel 5.1.6 .
presence	OUT	Bool	Anwesenheits-Bit Dieses Bit wird nur gesetzt, wenn sich ein Transponder im Feld des Readers befindet

5.1.4. LIOLink_RF200_WriteTag

Beschreibung

Der FB "LIOLink_RF200_Write" schreibt einen Datenblock aus dem Anwenderprogramm auf einen Transponder.

Parameter

Abbildung 5-4: LIOLink_RF200_WriteTag

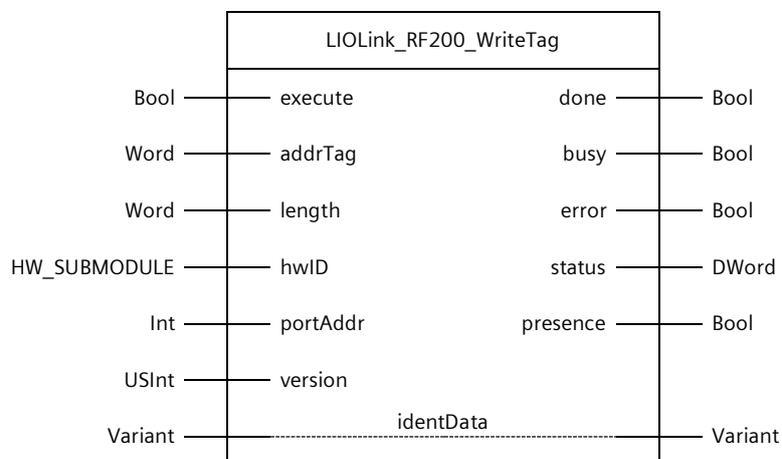
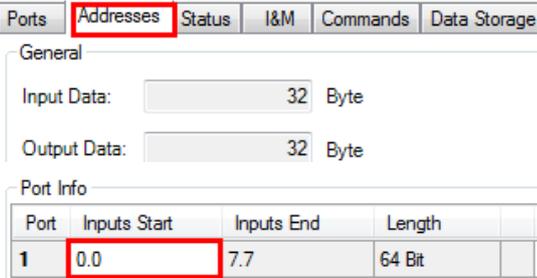


Tabelle 5-4: Parameter von LIOLink_RF200_WriteTag

Name	P-Typ	Datentyp	Kommentar
execute	IN	Bool	Aktiviert den Schreibauftrag bei positiver Flanke.
addrTag	IN	Word	Anfangsadresse der zu schreibenden Daten auf dem Transponder.
length	IN	Word	Länge der Daten, die auf den Transponder geschrieben werden. Hinweis: Der Reader schreibt mindestens 4 (V1.0) bzw. 28 Byte (V1.1) auf den Transponder. Deshalb muss die Länge größer 4 (V1.0) bzw. 28 Byte (V1.1) gewählt werden.
hwID	IN	HW_SUBMODULE	Hardware-Kennung des IO-Link Masters
portAddr	IN	Int	Anfangsadresse des angeschlossenen Readers (PCT Tool). 
version	IN	USInt	IO-Link-Version des Readers 11: IO-Link-Version 1.1 10: IO-Link-Version 1.0
done	OUT	Bool	TRUE: Auftrag wurde erfolgreich ausgeführt
busy	OUT	Bool	TRUE: Auftrag wird gerade bearbeitet
error	OUT	Bool	TRUE: Ein Fehler ist bei der Bearbeitung des FB aufgetreten
status	OUT	DWord	Fehlercode im Fehlerfall, siehe Kapitel 5.1.6 .
presence	OUT	Bool	Anwesenheits-Bit Dieses Bit wird nur gesetzt, wenn sich ein Transponder im Feld des Readers befindet.
identData	IN_OUT	Variant	Datenbereich, aus dem die zu schreibenden Daten gelesen werden

5.1.5. Integration ins Anwenderprojekt

Sie finden ein ausführliches Anwendungsbeispiel zur Integration der Bausteine in Ihr Anwenderprojekt im Siemens Industry Online Support:

<https://support.industry.siemens.com/cs/ww/de/view/73565887>

5.1.6. Fehlerhandling

HINWEIS Die nachfolgenden Informationen gelten nur für die FBs "LIOLink_RF200_ReadTag", "LIOLink_RF200_SwitchAntenna" und "LIOLink_RF200_WriteTag".

Fehlermeldungen der FBs

Tabelle 5-5: Fehlermeldungen der FBs

Status	Beschreibung	Abhilfe
16#00018101	Der Transponder hat vor Ablauf des Lese-/Schreibvorgangs das Feld verlassen.	Prozess erneut starten
16#00018102	Der vorherige Auftrag ist noch nicht abgeschlossen. Der Auftrag wird zu dem nächstmöglichen Zeitpunkt beendet.	Prozess erneut starten

Status	Beschreibung	Abhilfe
16#00018103	Es wurde kein Transponder innerhalb der angegebenen Zeitdauer im Feld des Readers erkannt.	Prozess erneut starten
16#00018104	Die angegebene Länge ist kleiner als 4 (IO-Link V1.0) oder 28 (IO-Link V1.1).	Eine Länge größer 4 Byte (IO-Link V1.0) oder 28 Byte (IO-Link V1.1) angeben
16#000180xy 16#000187xy 16#000185xy 16#00018xyy	Fehlermeldungen der erweiterten Anweisungen "BLKMOV", "DPRD_DAT" und "DPWR_DAT".	Siehe TIA Portal Informationssystem
16#001100xx	Fehlermeldungen des angeschlossenen Readers.	Siehe Gerätehandbuch

Fehlermeldungen des angeschlossenen RF200 IO-Link-Readers

Den Fehler des angeschlossenen Readers des können Sie auf folgenden Wegen ermitteln:

- Direkt am Reader durch Abzählen des Blinkmusters der roten Fehler-LED
- Über den Parameter "status" (16#001100xx)

Eine detailliertere Übersicht dieser Fehler finden Sie im Handbuch "RFID-Systeme SIMATIC RF200 IO-Link":
<https://support.industry.siemens.com/cs/ww/de/view/109766065>

5.2. SIRIUS

5.2.1. Beschreibung

Mit den hier beschriebenen Bausteinen der Bibliothek können Sie diverse IO-Link-fähige SIRIUS-Geräte über eine einfache Anwenderschnittstelle ansteuern.

Dies umfasst die folgenden Geräte:

- Kompaktabzweige SIRIUS 3RA
- Elektronische Überlastrelais SIRIUS 3RB24
- Überwachungsrelais SIRIUS 3RR24
- Temperaturüberwachungsrelais SIRIUS 3RS1
- Temperaturüberwachungsrelais SIRIUS 3RS2
- Elektronikmodul SIRIUS ACT 3SU1
- ID-Schlüsselschalter SIRIUS ACT 3SU1
- Überwachungsrelais SIRIUS 3UG48
- Signalsäulen SIRIUS 8WD46 (nur S7-1200/1500)

Gültigkeit

Diese FBs stehen jeweils in zwei Varianten für S7-1200/1500 und S7-300/400 zur Verfügung.

5.2.2. Parameter

Die Parameter der Bausteine für die unterschiedlichen Gerätetypen sind nahezu identisch. Daher werden die Parameter exemplarisch anhand des FB "LIOLink_3SU1_ElecModule" erläutert.

Abbildung 5-5: LIOLink_3SU1_ElecModule

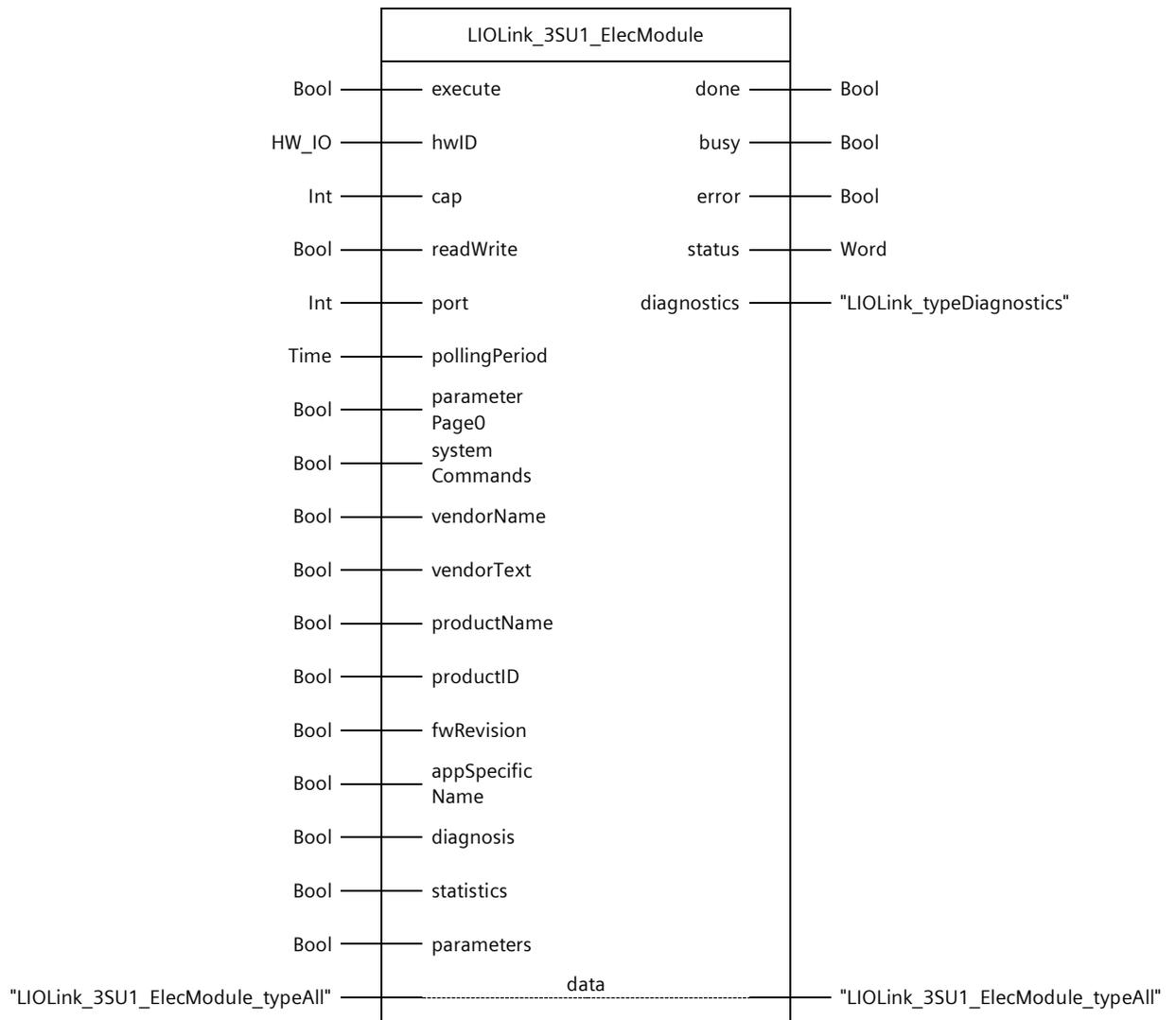


Tabelle 5-6: Parameter von LIOLink_3SU1_ElecModule

Name	P-Typ	Datentyp	Kommentar
execute	IN	Bool	Anforderung, die Funktion auszuführen
hwID	IN	HW_IO	S7-1200/1500: Hardwareerkennung des IO-Link Masters oder des ersten Submoduls. S7-300/400: Logische Adresse des IO-Link Masters (Modul oder Submodul)

Name	P-Typ	Datentyp	Kommentar
cap (optional)	IN	Int	Zugangspunkt (Client Access Point): S7-1200/1500: Wird bei der Verwendung von Siemens IO-Link Mastern automatisch erkannt und ist daher optional. Wird der Parameter verändert, ist die automatische Erkennung nicht aktiv. S7-300/400: Die automatische Erkennung ist in dieser Version nicht implementiert. Der Zugangspunkt muss daher manuell gesetzt werden. Für gewöhnlich besitzt er den Wert 0xB400 oder 0x00E3 und ist im jeweiligen Handbuch angegeben.
readWrite	IN	Bool	Modus FALSE: lesen TRUE: schreiben
port	IN	Int	Portnummer an dem das IO-Link Device betrieben wird Mögliche Werte: 0..63
pollingPeriod	IN	Time	S7-1200/1500: Variabel einstellbare Zeit, die der Baustein wartet, bis der Datensatz übermittelt wird. Defaultwert: 100 ms S7-300/400: Die Funktionalität ist in dieser Version nicht implementiert, weswegen der Parameter hier nicht existiert.
parameterPage0	IN	Bool	Lies Parameterseite 0
systemCommands	IN	Bool	Schreib Systembefehle
vendorName	IN	Bool	Lies Herstellername
vendorText	IN	Bool	Lies Herstellertext
productName	IN	Bool	Lies Produktname
productID	IN	Bool	Lies Produkt-ID
fwRevision	IN	Bool	Lies Firmwareversion
appSpecificName	IN	Bool	Lies/schreib Anwenderspezifische Markierung
diagnosis	IN	Bool	Lies Diagnose (Datensatz 92)
statistics	IN	Bool	Lies Diagnose (Datensatz 94)
parameters	IN	Bool	Lies/schreib Parameter (Datensatz 131)
done	OUT	Bool	TRUE: Auftrag wurde erfolgreich ausgeführt
busy	OUT	Bool	TRUE: Auftrag wird gerade bearbeitet.
error	OUT	Bool	TRUE: Ein Fehler ist bei der Bearbeitung des FB aufgetreten
status	OUT	Word	Status von LIOLink_Device (siehe Kapitel 3.1.3)
diagnostics	OUT	"LIOLink_ typeDiagnostics"	Diagnoseinformationen von LIOLink_Device (siehe Kapitel 3.1.3)
data	IN_OUT	"LIOLink_3SU1_ ElecModule_typeAll"	Geräte-spezifischer Ziel-/Quellbereich

HINWEIS

Pro Auftrag kann nur ein Datensatz gelesen/geschrieben werden. Der FB liest/schreibt den zum ersten gesetzten Parameter zugehörigen Datensatz.

5.2.3. Integration ins Anwenderprojekt

Sie finden ein ausführliches Anwendungsbeispiel zur Integration der Bausteine in Ihr Anwenderprojekt im Siemens Industry Online Support:

<https://support.industry.siemens.com/cs/ww/de/view/90529409>

5.2.4. Fehlerhandling

Der Baustein gibt die Status- und Diagnoseinformationen vom intern aufgerufenen FB "LIOLink_Device" an den Ausgängen "status" und "diagDeviceFB" aus (siehe Kapitel [3.1.3](#)).

6. Profilbausteine

Die IO-Link Community definiert standardisierte Geräteprofile für IO-Link Devices, um die Zugriffe aus dem Anwenderprogramm einer Steuerung auf IO-Link Devices zu vereinheitlichen.

Die Geräteprofile für IO-Link basieren auf einheitlichen Datenstrukturen, Dateninhalten und Basisfunktionalitäten für die IO-Link Devices. Dadurch kann für eine Vielzahl unterschiedlicher IO-Link Devices, die demselben Geräteprofil entsprechen, eine einheitliche Schnittstelle im Programm erstellt werden und die Anzahl unterschiedlicher Funktionsblöcke von unterschiedlichen Herstellern auf ein Minimum reduziert werden.

Voraussetzung zur Anwendung der Bausteine ist, dass die Datenstruktur des verwendeten IO-Link Device das entsprechende IO-Link-Profil unterstützt.

Die Spezifikation zu den einzelnen Geräteprofilen finden Sie auf der Webseite der IO-Link Community:

<https://IO-Link.com/>

Die Profilbausteine dieser Bibliothek ermöglichen den standardisierten Zugriff über die Common- und Smart Sensor-Profile.

Common-Profil

Das Common-Profil liefert einheitliche Informationen zur Identifizierung und Diagnose des IO-Link Device. Es ist allgemein gültig für IO-Link Devices.

Smart Sensor-Profil

Das Smart Sensor-Profil unterteilt sich zusätzlich in schaltende (adjustable switching sensors) und messende (measurement data channel) Profile. D. h. bei diesen Sensorprofilen werden Schaltpunkte oder Messwerte übertragen. Bei "Smart Sensor" Devices handelt es sich um Gerätegruppen, die einem implementierten Profil entsprechen.

Gültigkeit

Diese FBs stehen nur für S7-1200/1500 zur Verfügung.

HINWEIS

Beachten Sie, dass die Geräteprofile nur für IO-Link Devices verfügbar sind, die für den IO-Link Standard ab V1.1 spezifiziert sind.

6.1. LIOLink_AdjSwitchingSensor

6.1.1. Beschreibung

Der FB "LIOLink_AdjSwitchingSensor" bietet eine einheitliche Schnittstelle für den Zugriff und die Parametrierung von IO-Link Devices, die das Smart Sensor-Profil unterstützen.

Insbesondere kann der Baustein von Sensoren eingesetzt werden, die dem Messgerät Profil Typ 2 zugeordnet werden können, d. h. IO-Link Devices, die das Smart Sensor-Profil "Adjustable Switching Sensors" unterstützen.

Verstellbare Schaltsensoren oder Adjustable Switching Sensors (AdSS) innerhalb des Smart Sensor-Profils sind Geräte, die genau ein binäres Ausgangssignal (Schaltsignal) bereitstellen. Der Sollwert dieses Schaltausgangs kann von der Anwendung entweder durch Eingabe eines eigenen Sollwerts bei der Konfiguration oder mit Hilfe eines Teach-In-Verfahrens definiert werden.

Darüber hinaus sind verschiedene Teach-In-Verfahren, wie Einzelwert Teach-In, Zweiwert Teach-In oder dynamischer Teach-In möglich, was die Inbetriebnahme der Anwendung erleichtert. Je nach Sensortyp sind individuelle Kombinationen dieser Teach-In-Verfahren möglich.

Die Schaltpunktlogik (High-aktiv/Low-aktiv) kann von der Anwendung definiert werden.

HINWEIS Beachten Sie, dass bei mehrfachem gleichzeitigem Aufrufen des FB "LIOLink_AdjSwitchingSensor" für den gleichen Master (z. B. Informationsabruf für mehrere Ports gleichzeitig) nur ein Bausteinaufruf erfolgreich beendet wird. An den anderen Bausteinen wird ein Statuskonflikt "diagnostics.status" = 16#8402 des unterlagerten FB "LIOLink_Device" ausgegeben (Sende- und Response Daten inkonsistent).

6.1.2. Parameter

Abbildung 6-1: LIOLink_AdjSwitchingSensor

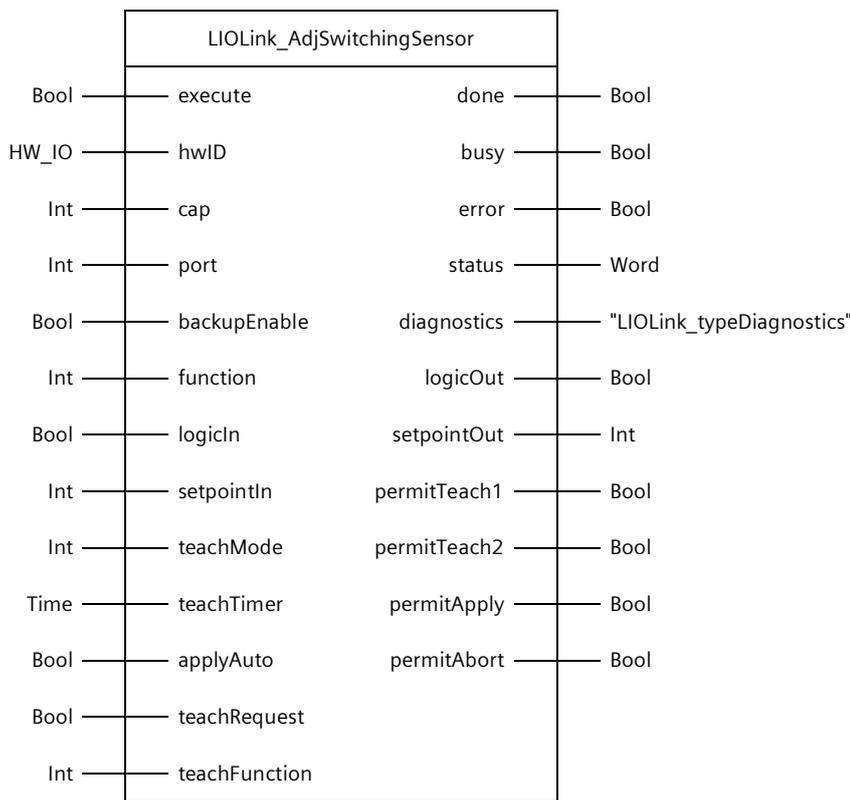


Tabelle 6-1: Parameter von LIOLink_AdjSwitchingSensor

Name	P-Typ	Datentyp	Kommentar
execute	IN	Bool	Anforderung, die Funktion auszuführen
hwID	IN	HW_IO	Hardwarekennung des IO-Link Masters oder des ersten Submoduls.
cap (optional)	IN	Int	Zugangspunkt (Client Access Point): Wird bei der Verwendung von Siemens IO-Link Mastern automatisch erkannt und ist daher optional. Wird der Parameter verändert, ist die automatische Erkennung nicht aktiv. Für gewöhnlich ist der Zugangspunkt 0xB400 oder 0x00E3 und ist im jeweiligen Handbuch angegeben.
port	IN	Int	Portnummer an dem das IO-Link Device betrieben wird Mögliche Werte: 0..63
backupEnable	IN	Bool	TRUE: über das Kommando "wr_ident" werden die an den Eingängen "applicationSpecificTagIn", "locationTagIn" und "functionTagIn" anliegenden Informationen im Device gespeichert FALSE: der Backup-Mechanismus wird nicht vom Baustein ausgeführt

Name	P-Typ	Datentyp	Kommentar
function	IN	Int	Funktionsauswahl: 0: no_func 1: rd_all 2: wr_conf 3: wr_param 4: teach Siehe Kapitel 6.1.3.1
logicIn	IN	Bool	TRUE: der gerade am Sensor anliegende Schalterpunkt, wird über die Funktion "wr_conf" in den Sensor geschrieben FALSE: das Auslesen des aktuellen Schalterpunkts am Sensor ist inaktiv
setpointIn	IN	Int	Definiert den Wert für einen neuen Sollwert, der bei einer Anforderung mit der Funktion "wr_param" in den Sensor geschrieben wird
teachMode	IN	Int	Auswahl des Teach-Vorgangs: 0: no_teach: Keine Teach-In-Aktion 1: single_value: Einzelwert Teach-In 2: two_value: Zweiwert Teach-In 3: dynamic: Dynamisches Teach-In Siehe Kapitel 6.1.3.2
teachTimer	IN	Time	Definiert die Dauer der dynamischen Teach-Zeit. Ein Wert von "0" deaktiviert die Aktivierung des automatischen Stoppbefehls. Die Teach-Funktion "teach_Stop" kann immer zum Auslösen des dynamischen Teach-Stopps verwendet werden und überschreibt damit den Teach Timer.
applyAuto	IN	Bool	Definiert das Verhalten bei einem zweiwertigen Teach-Vorgang. FALSE = automatische Übernahme deaktiviert Die Übernahmefunktion muss vom Anwender ausgelöst werden, um die gesammelten Teach-Points auszuwerten und den neuen Sollwert zu aktivieren. TRUE = automatische Übernahme aktiviert Wurden zwei Teach-Punkte erfolgreich eingelernt, wird ein automatisches Übernehmen ausgelöst. Es ist keine Aktivität des Anwenders erforderlich.
teachRequest	IN	Bool	Eine steigende Flanke löst einen Teach-Schritt entsprechend der gewählten Teach-Funktion am Eingang "teachFunction" aus.
teachFunction	IN	Int	Teach-Funktion, die mit einer Teach-Anforderung am Eingang "teachRequest" ausgeführt wird: 0: keine Teach-Funktion 1: Starte Teach-Schritt 1 2: Starte Teach-Schritt 2 3: Zweiwertige Teach-Ergebnisse übernehmen 4: Abbruch der aktuellen Teach-Sequenz Siehe Kapitel 6.1.3.2
done	OUT	Bool	TRUE: Auftrag wurde erfolgreich ausgeführt
busy	OUT	Bool	TRUE: Auftrag wird gerade bearbeitet
error	OUT	Bool	TRUE: Ein Fehler ist bei der Bearbeitung des FB aufgetreten
status	OUT	Word	16#0000–16#7FFF: Status des FB 16#8000–16#FFFF: Fehlercodes (siehe Kapitel 6.1.4)
diagnostics	OUT	"LIOLink_type Diagnostics"	Detaillierte Diagnoseinformationen des FB (siehe Kapitel 6.1.4)

Name	P-Typ	Datentyp	Kommentar
logicOut	OUT	Bool	Dieser Ausgang stellt den aktuellen Wert des Parameters "Logik" des Sensors dar. Die Variable wird bei jedem Abschluss eines Teach-Vorgangs, eines Schreibvorgangs oder bei einem Anforderungssignal mit der Funktion "rd_all" aktualisiert.
setpointOut	OUT	Int	Dieser Ausgang stellt den aktuellen Wert des Parameters "Sollwert" des Sensors dar. Die Variable wird bei jedem Abschluss eines Teach-Vorgangs, eines Schreibvorgangs oder bei einem Anforderungssignal mit der Funktion "rd_all" aktualisiert.
permitTeach1	OUT	Bool	Das Signal wird gesetzt, wenn gemäß dem anstehenden Status des FB ein Triggersignal für die Teach-Funktion "teach_1" möglich ist.
permitTeach2	OUT	Bool	Das Signal wird gesetzt, wenn gemäß dem anstehenden Status des FB ein Triggersignal für die Teach-Funktion "teach_2" möglich ist.
permitApply	OUT	Bool	Das Signal wird gesetzt, wenn gemäß dem anstehenden Status des FB ein Triggersignal für die Teach-Funktion "apply" möglich ist.
permitAbort	OUT	Bool	Das Signal wird gesetzt, wenn gemäß dem anstehenden Status des FB ein Triggersignal für die Teach-Funktion "abort" möglich ist.

6.1.3. Bedienung

6.1.3.1. Funktionen

Der Baustein bietet fünf Funktionen. Eine Funktion kann durch Angabe der entsprechenden Nummer am Eingang "function" ausgewählt werden. Eine steigende Flanke am Eingang "execute" löst die ausgewählte Funktion aus.

- **no_func (0)**
Es wird keine Funktion ausgeführt.
- **rd_all (1)**
Bei dieser Funktion werden die aktuellen Schaltsignale und Parameterwerte vom Sensor gelesen. Die gelesenen Werte stehen an den Ausgängen "logicOut" und "setpointOut" zur Verfügung.
- **wr_conf (2)**
Diese Funktion bewirkt, dass ein zuvor angelegter Wert für "logicIn" in den Sensor geschrieben wird.
- **wr_param (3)**
Diese Funktion bewirkt, dass ein zuvor angelegter Wert für "setpointIn" in den Sensor geschrieben wird.
- **teach (4)**
Diese Funktion bewirkt, dass der Baustein in den Teach-Vorgang übergeht (siehe Kapitel [6.1.3.2.](#))

Wenn der Eingang "backupEnable" gesetzt ist, wird der IO-Link Systembefehl "ParameterDownloadStore" aufgerufen. Dadurch wird der Datenspeichermechanismus (DataStorage) gestartet und die neue Parametrierung im IO-Link Master gespeichert.

6.1.3.2. Teach-In

Übersicht

In diesem Abschnitt werden Erläuterungen zu den unterschiedlichen Teach-Funktionen gegeben. Über die Teach-In-Funktionen werden Sollwerte zum Schalten des Schaltausgangs am Sensor eingelernt. Es sind verschiedene Teach-In-Verfahren, wie Einzelwert Teach-In, Zweiwert Teach-In oder dynamischer Teach-In möglich, was die Inbetriebnahme der Anwendung erleichtert. Je nach Sensortyp sind individuelle Kombinationen dieser Teach-In-Verfahren möglich.

Es stehen vier Teach-Vorgänge zur Verfügung, die über den Eingang "teachMode" ausgewählt werden können:

- **no_teach (0):** Keine Teach-In-Aktion
- **single_value (1):** Einzelwert Teach-In
- **two_value (2):** Zweiwert Teach-In
- **dynamic (3):** Dynamisches Teach-In

Während eines Teach-Vorgangs, werden die einzelnen Teach-Schritte über die Eingänge "teachRequest" und "teachFunction" gesteuert.

Die unterschiedlichen Teach-Mechanismen werden im nächsten Abschnitt näher erläutert.

Der Ausgang "status" stellt Informationen zum aktuell anstehenden Einlernschritt zur Verfügung. Wird eine "teachFunction" angefordert, die zu diesem Zeitpunkt nicht erlaubt ist, dann stoppt der Funktionsblock den Teach-Vorgang und gibt einen Fehler zurück.

Einzelwert Teach-In

Nachfolgend werden die Schritte für ein Einzelwert Teach-In aufgeführt:

1. Eingänge "function" = 4 und "teachMode" = 1 setzen.
2. Eingang "execute" = TRUE setzen.
3. Abwarten bis Ausgänge "permitTeach1" = TRUE und "status" = 16#7110 (Teach-Prozess wartet auf weiteren Befehl).
4. Objekt in die gewünschte Position bringen.
5. Eingang "teachFunction" = 1 setzen.
6. Eingang "teachRequest" = TRUE setzen.

Der Teach-Vorgang war erfolgreich, wenn Ausgänge "done" = TRUE und "status" = 16#0000.

Zweiwert Teach-In

Nachfolgend werden die Schritte für einen Zweiwert Teach-In aufgeführt:

1. Eingänge "function" = 4 und "teachMode" = 2 setzen.
2. Eingang "execute" = TRUE setzen.
3. Abwarten bis Ausgänge "permitTeach1" = TRUE, "permitTeach2" = TRUE und "status" = 16#7120 (Teach-Prozess wartet auf weiteren Befehl).
4. Objekt in die gewünschte Position für Teach-Point 1 bringen.
5. Eingang "teachFunction" = 1 setzen.
6. Eingang "teachRequest" = TRUE setzen.
7. Abwarten bis Ausgänge "permitTeach1" = TRUE, "permitTeach2" = TRUE und "status" = 16#7120 (Teach-Prozess wartet auf weiteren Befehl).
8. Objekt in die gewünschte Position für Teach-Point 2 bringen.
9. Eingang "teachFunction" = 2 setzen.

10. Eingang "teachRequest" = FALSE setzen.
11. Eingang "teachRequest" = TRUE setzen.
12. Wenn "applyAuto" = FALSE, muss "teachApply" manuell ausgelöst werden:
 - a. Abwarten bis Ausgänge "permitApply" = TRUE und "status" = 16#7140 (Teach-Prozess wartet auf Befehl zur Übernahme der eingelernten Werte).
 - b. Eingang "teachFunction" = 3 setzen.
 - c. Eingang "teachRequest" = FALSE setzen.
 - d. Eingang "teachRequest" = TRUE setzen.

Wenn "applyAuto" = TRUE, wird "teachApply" automatisch ausgelöst.

Der Teach-Vorgang war erfolgreich, wenn Ausgänge "done" = TRUE und "status" = 16#0000.

Alternativ kann "teachPoint2" auch zuerst eingelernt werden.

Ein Teach-Point kann auch mehrfach während des zweiwertigen Teach-In-Prozesses eingelernt werden.

Ein Abbruch der Teach-Funktion ist mit "teachFunction" = 4 nach dem Einlernen des ersten Teachpunkts verfügbar. Den Teach-Vorgang können Sie abbrechen, indem Sie "teachFunction" = 4 setzen und eine steigende Flanke am Eingang "teachRequest" durchführen.

Dynamisches Teach-In

Nachfolgend werden die Schritte für ein dynamisches Teach-In aufgeführt:

1. Eingänge "function" = 4 und "teachMode" = 3 setzen (Dynamisches Teach-In).
2. Eingang "execute" = TRUE setzen.
3. Abwarten bis Ausgänge "permitTeach1" = TRUE und "status" = 16#7130 (Teach-Prozess wartet auf weiteren Befehl).
4. Eingang "teachFunction" = 1 setzen.
5. Eingang "teachRequest" = TRUE setzen.
6. Abwarten bis Ausgang "status" = 16#7131 (Teach-Prozess ist aktiv).
Der dynamische Einlernvorgang hat begonnen.
7. Objekt innerhalb des gewünschten Bereichs verschieben.
8. Wenn "teachTimer" = 0 s, muss der Teach-Point 2 manuell angestoßen werden:
 - a. Abwarten bis Ausgang "permitTeach2" = TRUE.
 - b. Eingang "teachFunction" = 2 setzen.
 - c. Eingang "teachRequest" = FALSE setzen.
 - d. Eingang "teachRequest" = TRUE setzen.

Wenn "teachTimer" \neq 0 s, wird der Teach-Point 2 automatisch angestoßen und der Teach-Prozess endet, wenn die Zeit abgelaufen ist. Der Timer startet nach dem erfolgreichen Senden des Teach-Auftrags an das IO-Link Device.

Der Teach-Vorgang war erfolgreich, wenn Ausgänge "done" = TRUE und "status" = 16#0000.

Ein Abbruch der Teach-Funktion ist nach dem Start des Teach-Vorgangs verfügbar. Diesen können Sie durch "teachFunction" = 4 und eine steigende Flanke am Eingang "teachRequest" abbrechen.

6.1.4. Fehlerhandling

Der Ausgang "status" gibt den aktuellen Status sowie Fehler aus, während der Ausgang "diagnostics" im Fehlerfall eine Diagnosestruktur mit detaillierten Informationen bereitstellt.

status

Tabelle 6-2: Fehlercodes des Ausgangs "status"

Ausgangsstatus	Erläuterung
Funktionsblock interner Status	
16#0000	Auftrag erfolgreich ausgeführt
16#7000	Kein Auftrag in Bearbeitung (Initialwert)
16#7011	Beschäftigt Daten zu lesen
16#7012	Beschäftigt Daten zu schreiben
16#7026	Beschäftigt mit Backup
16#7101	Teach-Prozess gestartet
16#7102	Teach-Prozess, Folgeaufruf
16#7110	Teach-Prozess, Einzelwert Teach-Prozess wartet auf weiteren Befehl
16#7111	Teach-Prozess, Einzelwert Teach-Prozess ist beschäftigt
16#7120	Teach-Prozess, Zweiwert Teach-Prozess wartet auf weiteren Befehl
16#7121	Teach-Prozess, Zweiwert Teach-Prozess ist beschäftigt
16#7130	Teach-Prozess, Dynamischer Teach-Prozess wartet auf weiteren Befehl
16#7131	Teach-Prozess, Dynamischer Teach-Prozess ist beschäftigt
16#7140	Teach-Prozess wartet auf Befehl eingelernte Werte zu übernehmen
16#7141	Teach-Prozess, eingelernte Werte werden übernommen
16#7151	Teach-Prozess wird abgebrochen
16#8030	Der Wert am Eingang "function" liegt außerhalb des definierten Bereichs
16#8330	Während des Teach-Prozesses ist ein Fehler aufgetreten
16#8331	Fehler, Anforderung Teach-Funktion in diesem Zustand nicht erlaubt
16#8332	Der Wert am Eingang "teachMode" liegt außerhalb des definierten Bereichs
16#8600	Fehler im unterlagerten FB "LIOLink_Device", siehe "diagnostics"

diagnostics

Der Ausgang "diagnostics" gibt im Fehlerfall detaillierte Informationen zu dem vorliegenden Fehler aus.

Tabelle 6-3: Ausgang "diagnostics"

Variable	Beschreibung
status	Letzter Statuscode des Schnittstellenparameters "status" des FB.
subfunctionStatus	Status von LIOLink_Device, siehe Kapitel 3.1.4 .
stateNumber	Zustand des Zustandsautomaten des FB, in dem der Fehler aufgetreten ist.

6.2. LIOLink_IdentAndDiag

6.2.1. Beschreibung

Der FB "LIOLink_IdentAndDiag" liest und schreibt azyklisch Identifikation und Diagnose-Daten und gibt den Status des angeschlossenen IO-Link Device aus. Der Funktionsbaustein unterstützt das "Common Profile" aus der IO-Link Spezifikation.

Diese Profil-ID beinhaltet die Funktionsklassen "Deviceidentification", "DeviceDiagnosis", "ProcessDataVariable" und "ExtendedIdentification". Sie werden zu einem Device-Profil zusammengefasst.

HINWEIS

Beachten Sie, dass bei mehrfachem gleichzeitigem Aufrufen des FB "LIOLink_IdentAndDiag" für den gleichen Master (z. B. Informationsabruf für mehrere Ports gleichzeitig) nur ein Bausteinaufruf erfolgreich beendet wird. An den anderen Bausteinen wird ein Statuskonflikt "diagnostics.status" = 16#8402 des unterlagerten FB "LIOLink_Device" ausgegeben (Sende- und Responsedaten inkonsistent).

6.2.2. Parameter

Abbildung 6-2: LIOLink_IdentAndDiag

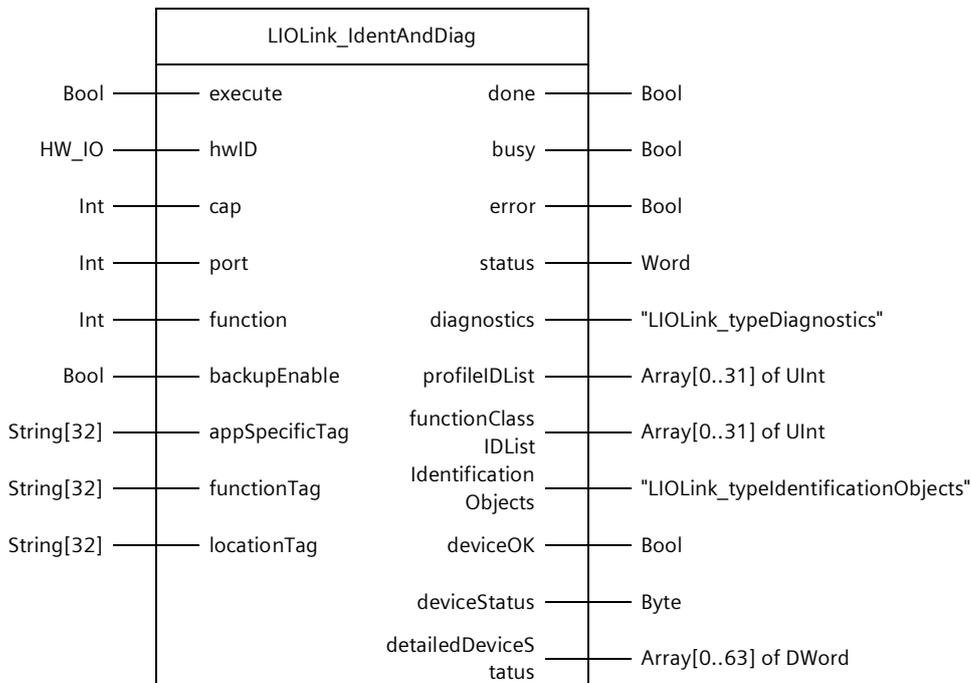


Tabelle 6-4: Parameter von LIOLink_IdentAndDiag

Name	P-Typ	Datentyp	Kommentar
execute	IN	Bool	Anforderung, die Funktion auszuführen
hwID	IN	HW_IO	Hardwareerkennung des IO-Link Masters oder des ersten Submoduls.
cap (optional)	IN	Int	Zugangspunkt (Client Access Point): Wird bei der Verwendung von Siemens IO-Link Mastern automatisch erkannt und ist daher optional. Wird der Parameter verändert, ist die automatische Erkennung nicht aktiv. Für gewöhnlich ist der Zugangspunkt 0xB400 oder 0x00E3 und ist im jeweiligen Handbuch angegeben.
port	IN	Int	Portnummer an dem das IO-Link Device betrieben wird Mögliche Werte: 0..63

Name	P-Typ	Datentyp	Kommentar
function	IN	Int	<p>Funktionsauswahl</p> <p>0: no_func Keine Funktion</p> <p>1: rd_all Identifikations- und Diagnose-Daten des gewählten IO-Link Device werden ausgelesen.</p> <p>2: rd_diag Diagnose-Daten des gewählten IO-Link Device werden ausgelesen.</p> <p>3: wr_ident Werte an den Eingängen "appSpecificTag", "locationTag" und "functionTag" werden auf das gewählte IO-Link Device geschrieben.</p> <p>Eine ausführliche Beschreibung der Funktionen finden Sie in Kapitel 6.2.3.</p>
backupEnable	IN	Bool	<p>TRUE: über das Kommando "wr_ident" werden die an den Eingängen "appSpecificTag", "locationTag" und "functionTag" anliegenden Informationen im Device gespeichert</p> <p>FALSE: der Backup-Mechanismus wird nicht vom Baustein ausgeführt</p>
appSpecificTag	IN	String[32]	Anwenderspezifisches Kennzeichen, das über wr_ident auf das IO-Link Device geschrieben wird.
functionTag	IN	String[32]	Funktionskennzeichen, das über wr_ident auf das IO-Link Device geschrieben wird.
locationTag	IN	String[32]	Ortskennzeichen, das über wr_ident auf das IO-Link Device geschrieben wird.
done	OUT	Bool	TRUE: Auftrag wurde erfolgreich ausgeführt
busy	OUT	Bool	TRUE: Auftrag wird gerade bearbeitet
error	OUT	Bool	TRUE: Ein Fehler ist bei der Bearbeitung des FB aufgetreten
status	OUT	Word	<p>16#0000–16#7FFF: Status des FB</p> <p>16#8000–16#FFFF: Fehlercodes (siehe Kapitel 6.2.4)</p>
diagnostics	OUT	"LIOLink_type Diagnostics"	Detaillierte Diagnoseinformationen des FB (siehe Kapitel 6.2.4)
profileIDList	OUT	Array[0..31] of UInt	Gibt eine Liste der Profile IDs aus, welche das Gerät unterstützt
functionClassIDList	OUT	Array[0..31] of UInt	Gibt eine Liste der Funktionsklassen aus, welche das Gerät unterstützt
identificationObjects	OUT	"LIOLink_type IdentificationObjects"	In diesem PLC-Datentyp werden die Informationen der Identifikationsobjekte des Sensors gespeichert (siehe Tabelle 6-5)
deviceOK	OUT	Bool	<p>Geräteinformation</p> <p>TRUE: es stehen keine weiteren Diagnoseinformationen zur Verfügung</p> <p>FALSE: zusätzliche Informationen werden am Ausgang "deviceStatus" und "detailedDeviceStatus" ausgegeben</p>
deviceStatus	OUT	Byte	Gibt den aktuellen Status des Geräts aus (siehe Kapitel 6.2.4)
detailedDeviceStatus	OUT	Array[0..63] of DWord	Stellt zusätzliche Informationen des IO-Link Device zur Verfügung, die der Hersteller definiert (siehe Kapitel 6.2.4)

6.2.3. Funktionsweise

Funktionsauswahl

Am Funktionsbaustein können über den Eingangsparameter "function" unterschiedliche Funktionen angesteuert werden. Mit einer steigenden Flanke am Eingang "execute" wird die eingestellte Funktion ausgeführt:

- **no_func (0)**
Es wird keine Funktion ausgeführt.
- **rd_all (1)**
Es werden alle aktuellen Identifikations- und Diagnosedaten des IO-Link Device ausgelesen und an den entsprechenden Ausgängen ausgegeben. Alle in [Tabelle 6-6](#) hinterlegten Parameter werden am Device gelesen. Falls ein optionaler Parameter nicht gelesen werden kann, wird der Default-Wert ausgegeben.
- **rd_diag (2)**
Rücklesen der aktuellen Diagnoseparameterwerte aus dem Gerät. Im Gegensatz zu "function" = 1 wird nur "deviceStatus" und "detailedDeviceStatus" zurückgelesen. Wenn diese Parameter nicht gelesen werden können, werden die Standardwerte bereitgestellt.
- **wr_ident (3)**
Die Werte, die an den Eingängen "applicationSpecificTagIn", "locationTagIn" und "functionTagIn" bereitgestellt werden, werden auf das IO-Link Device geschrieben. Diese Eingänge werden auch direkt auf ihre jeweiligen Ausgänge geschrieben. Wenn "locationTagIn" oder "functionTagIn" nicht geschrieben werden können, werden die Default-Werte auf die Ausgänge geschrieben und der Ausgang "status" = 16#4000.

Backup

Wenn der Eingang "backupEnable" auf TRUE gesetzt ist, wird der IO-Link Systembefehl "ParameterDownloadStore" aufgerufen. Dadurch wird der Datenspeichermechanismus (DataStorage) gestartet und die neue Parametrierung im IO-Link Master gespeichert.

Es wird ein Fehler angezeigt, wenn obligatorische (M) Parameter (siehe [Tabelle 6-6](#)) nicht gelesen oder geschrieben werden können. In diesem Fall stoppt der Schreibe- oder Leseauftrag im FB und ein detaillierter Status wird an den Ausgängen "status" und "diagDeviceFB" ausgegeben (siehe Kapitel [6.2.4](#)).

PLC-Datentyp "typIdentificationObjects"

Am Ausgangsparameter "identificationObjects" wird der PLC-Datentyp "typIdentificationObjects" verschaltet. In diesem Datentyp ist die Identifikationsstruktur des IO-Link Device hinterlegt.

In einem Datenbaustein können alle Informationen zu den unterschiedlichen Profilsensoren gespeichert werden.

Tabelle 6-5: Parameter von LIOLink_typIdentificationObjects

Name	Typ	Kommentar
vendorID	Word	Eindeutige Herstelleridentifikation, die von der IO-Link Community vergeben wird
deviceID	DWord	Eindeutige Gerätekennung, die vom Hersteller vergeben wird
vendorName	String[64]	Herstellername
vendorText	String[64]	Zusätzliche Informationen über den Hersteller
productName	String[64]	Produktname
productID	String[64]	Produkt-ID
productText	String[64]	Produkttext
serialNumber	String[16]	Seriennummer
hwRevision	String[64]	Hardware-Version
fwRevision	String[64]	Firmware-Version
appSpecificTag	String[32]	Anwenderspezifische Markierung
locationTag	String[32]	Erweiterter Identifikationsparameter, der für die allgemeine Lokalisierung eines Geräts verwendet werden kann
functionTag	String[32]	Erweiterter Identifikationsparameter, der für die allgemeine Identifizierung eines Geräts verwendet werden kann

IO-Link Device-Parameter

Die folgende Tabelle zeigt, welche Parameter von einem IO-Link Device mit integriertem Common-Profil ausgelesen werden können.

Es gibt Parameter, die verpflichtend vom Hersteller ausgegeben werden müssen (M) oder die optional (O) zur Verfügung stehen.

Tabelle 6-6: IO-Link Device Parameter

Parameter	Name	Typ	Defaultwert	Verpflichtend (M)/ optional (O)	Beschreibung
16#0000 Byte 0x07, 0x08	Vendor ID	WORD	-	M	Eindeutige Lieferantenidentifikation, die von der IO-Link Community vergeben wird
16#0000 Byte 0x09, 0x0A, 0x0B	Device ID	DWORD	-	M	Eindeutige Gerätekennung, die von einem Lieferanten vergeben wird
16#000D	Profile Charakteristik	Array [32] of UINT	0	M	Die Profile basieren auf der Definition von FunctionClasses. Diese FunctionClasses können als eigenständige Eigenschaft verwendet oder mit dem Profileidentifizier kombiniert werden, wie z. B. <ul style="list-style-type: none"> • DeviceProfileIDs für bestimmte Klassen von Geräten oder • CommonApplicationProfileIDs zur allgemeinen Verwendung in allen Geräten
16#0010	Vendor Name	String [64]	-	M	Lieferantenname, welcher der VendorID zugeordnet ist
16#0011	Vendor Text	String [64]	"na"	O	Zusätzliche Informationen über den Lieferanten
16#0012	Produkt Name	String [64]	-	M	Produktname zur Unterscheidung von Varianten
16#0013	Produkt ID	String [64]	-	M	Herstellerspezifische Produkt- oder Typenbezeichnung
16#0014	Produkt Text	String [64]	"na"	O	Zusätzliche Produktinformationen, wie z. B. Produktkategorie
16#0015	Seriennumm er	String [16]	"na"	M	Eindeutiger herstellerspezifischer Code für jedes einzelne Gerät
16#0016	Hardware Revision	String [64]	"na"	M	Herstellerspezifische Kodierung für die Hardware-Revision des Geräts
16#0017	Firmware Revision	String [64]	-	M	Herstellerspezifische Kodierung der Firmware-Revision des Geräts
16#0018	Application Specific Tag	String [32]	-	M	Lese-/Schreibdatenobjekt für die Benutzeranwendung zur Identifizierung der spezifischen Vorrichtung
16#0019	Location Tag	String [32]	"na"	M	Erweiterter Identifikationsparameter, der für die allgemeine Lokalisierung eines Geräts verwendet werden kann. Der Inhalt ist nicht vordefiniert, es kann jede sichtbare Zeichenkette nach eigenen Namensregeln geschrieben werden.

Parameter	Name	Typ	Defaultwert	Verpflichtend (M)/ optional (O)	Beschreibung
16#001A	Function Tag	String[32]	"na"	M	Erweiterter Identifikationsparameter, der für die allgemeine Identifizierung eines Geräts verwendet werden kann. Der Inhalt ist nicht vordefiniert, es kann jede sichtbare Zeichenkette nach eigenen Namensregeln geschrieben werden.
16#0024	Device Status	BYTE	0	M	Die Informationen, die ausgegeben werden, entsprechen den Werten der Tabelle 6-9: Werte "deviceStatus" .
16#0025	Detailed Device Status	Array[64] of DWORD	0	M	In diesem Array werden zusätzliche detaillierte Geräteinformationen, die der Hersteller des IO-Link Devices definiert, ausgegeben.

6.2.4. Fehlerhandling

Der Ausgang "status" gibt den aktuellen Status sowie Fehler aus, während der Ausgang "diagnostics" im Fehlerfall eine Diagnosestruktur mit detaillierten Informationen bereitstellt.

status

Tabelle 6-7: Fehlercodes des Ausgangs "status"

Status	Bedeutung
16#0000	Auftrag abgeschlossen, keine Warnung und keine weitere Detaillierung
16#0003	Auftrag abgeschlossen, min. ein optionaler Parameter konnte nicht geschrieben werden
16#7000	Kein Auftrag in Bearbeitung (Initialwert)
16#7001	Erster Aufruf nach Eingang eines neuen Auftrags (steigende Flanke "execute")
16#7002	Folgeaufruf
16#8001	Beim Ausführen des Bausteins ist ein Fehler aufgetreten. Prüfen Sie Ihre Eingabewerte
16#8203	Der Wert am Eingang "function" liegt außerhalb des definierten Bereichs
16#8400	Fehler bei der Abarbeitung von außen. Optionaler Parameter ohne Default-Wert.
16#8600	Fehler im unterlagerten FB "LIOLink_Device", siehe "diagnostics"
16#87xx	Ein Parameter konnte nicht gelesen werden. xx enthält den Index (hex) des Parameters.
16#88xx	Ein Parameter konnte nicht geschrieben werden. xx enthält den Index (hex) des Parameters.

diagnostics

Der Ausgang "diagnostics" gibt im Fehlerfall detaillierte Informationen zu dem vorliegenden Fehler aus.

Tabelle 6-8: Ausgang "diagnostics"

Variable	Beschreibung
status	Letzter Statuscode des Schnittstellenparameters "status" des FB.
subfunctionStatus	Status von LIOLink_Device, siehe Kapitel 3.1.4 .
stateNumber	Zustand des Zustandsautomaten des FB, in dem der Fehler aufgetreten ist.

deviceStatus

Tabelle 6-9: Werte "deviceStatus"

Wert	Beschreibung
16#00	Das Gerät arbeitet ordnungsgemäß.
16#01	"Wartungsbedarf" Obwohl die Prozessdaten gültig sind, zeigt die interne Diagnose an, dass das Gerät kurz davor steht, seine Fähigkeit zur korrekten Funktion zu verlieren. z. B.: Optische Linsen werden verstaubt, Ablagerungen bilden sich, Schmierstoffspiegel niedrig
16#02	"Außerhalb der Spezifikation" Obwohl die Prozessdaten gültig sind, zeigt die interne Diagnose an, dass das Gerät außerhalb des angegebenen Messbereichs oder der Umgebungsbedingungen arbeitet. z. B.: Stromversorgung, Hilfsenergie, Temperatur, Luftdruck, magnetische Störungen, Vibrationen, Beschleunigung, Störlicht, Blasenbildung in Flüssigkeiten
16#03	"Funktionsprüfung" Prozessdaten sind aufgrund von beabsichtigten Manipulationen am Gerät vorübergehend ungültig. z. B.: Kalibrierungen, Teach-In, Positionsanpassungen, Simulation
16#04	"Fehler" Prozessdaten sind ungültig aufgrund von Fehlfunktionen im Gerät oder dessen Peripheriegeräten. Das Gerät kann seine vorgesehene Funktion nicht erfüllen.
16#05..FF	Reserviert

detailedDeviceStatus

Der Parameter gibt Informationen über aktuell anstehende Ereignisse zum IO-Link Device. Ereignisse vom Typ "Fehler" oder "Warnung" und Modus "Ereignis erscheint" werden in der Liste des detaillierten Gerätestatus mit EventQualifier und EventCode ausgegeben. Beim Eintreten eines Ereignisses mit Modus "Ereignis verschwindet" wird der entsprechende Eintrag im Detailed Device Status auf EventQualifier "0x00" und EventCode "0x000000" gesetzt. Auf diese Weise liefert dieser Parameter immer den aktuellen Diagnosestatus des Geräts. Der Parameter ist ein schreibgeschütztes Datenobjekt. Es können maximal 64 Array-Elementen (Event Einträge) angezeigt werden, wobei jedoch die Anzahl der Array-Elemente dieses Parameters gerätespezifisch ist. Auch die gelieferten Event-Codes sind gerätespezifisch und werden vom Hersteller definiert. Beim Ausschalten oder Zurücksetzen des IO-Link Device wird der Inhalt aller Array-Elemente auf die Grundeinstellungen gesetzt - EventQualifier "0x00", EventCode "0x000000".

Tabelle 6-10: Aufteilung "detailedDeviceStatus"

Eintrag	Ereignis	Datentyp	Comment
1	Error/Warnung_1	DWORD	Alle Einträge: 16#0000, es steht kein Fehler oder keine Warnung an
2	Error/Warnung_2	DWORD	
3	Error/Warnung_3	DWORD	1. BYTE: EventQualifier
...	-	DWORD	2.,3. BYTE: EventCode
n (max.64)	Error/Warnung_n	DWORD	4. BYTE: enthält keine weitere Information

Die IO-Link Device-Hersteller können die Implementierung einer statischen Liste wählen, d. h. eine feste Array-Position für jedes Event mit einem bestimmten EventCode, oder eine dynamische Liste, d. h. jeder Event Eintrag wird in der nächsten freien Array-Position gespeichert. Für eine dynamische Liste ist der Zugriff auf den Subindex nicht erlaubt.

6.3. LIOLink_MeasuredDataChannel

6.3.1. Beschreibung

Der FB "LIOLink_MeasuredDataChannel" bietet eine einheitliche Schnittstelle für den Zugriff auf IO-Link Devices, die das Smart Sensor-Profil unterstützen.

Insbesondere kann der Baustein von Sensoren eingesetzt werden, die das Smart Sensor-Profil Typ 3 ("Digital Measuring Sensors") und Typ 4 ("Digital Measuring Switching Sensors") unterstützen. In der aktuellen Version wird allerdings für den Typ 4 nur 4.1 und 4.2 unterstützt.

6.3.2. Parameter

Abbildung 6-3: LIOLink_MeasuredDataChannel

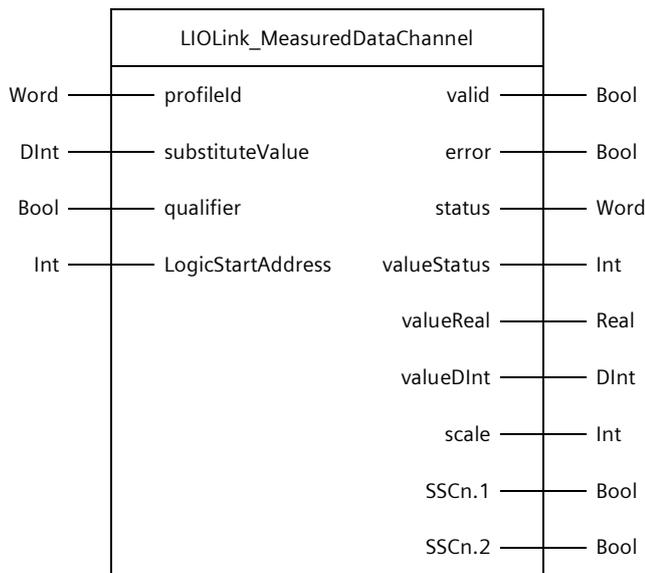


Tabelle 6-11: Parameter von LIOLink_MeasuredDataChannel

Name	P-Typ	Datentyp	Kommentar
profileId	IN	Word	ausgewählte Profil-ID bzw. Prozessdatenstruktur 1 = SSP 3.1 2 = SSP 3.2 3 = SSP 3.3 4 = SSP 3.4 5 = SSP 4.1.x 6 = SSP 4.2.x (SSP = Smart Sensor Profil)
substituteValue	IN	DInt	Der angegebene Ersatzwert wird auf den "valueReal" und "valueDINT" angewendet, wenn "valueStatus" ungleich 0 ist.
qualifier	IN	Bool	Dieses Signal entspricht der Port Qualifier-Information des Sensors. FALSE = Prozessdaten sind ungültig TRUE = Prozessdaten sind gültig Das Port Qualifier-Bit kann im PCT-Tool aktiviert werden. Es wird für jeden IO-Link Port ein Bit reserviert.
LogicStartAddress	IN	Int	Logische Adresse des ersten Bytes des Prozessdatenabbilds der Eingänge.
valid	OUT	Bool	Wenn der Wert TRUE ist, sind die angegebenen Werte gültig und können für weitere Berechnungen verwendet werden.
error	OUT	Bool	Wenn der Wert TRUE ist, tritt ein interner Fehler auf und weitere Informationen werden am Funktionsbaustein über den Ausgang "status" bereitgestellt.

Name	P-Typ	Datentyp	Kommentar
status	OUT	Word	Bietet interne Fehlercodes (siehe Tabelle 6-16)
valueStatus	OUT	Int	Status der Prozessdateneingabe 0 = ok 1 = Prozessdaten ungültig 2 = Keine Daten 3 = Außerhalb des Bereichs (+) 4 = Außerhalb des Bereichs (-) 5 = nicht definiert
valueReal	OUT	Real	Prozessdaten im Realformat zur Auswertung innerhalb der SPS
valueDInt	OUT	DInt	Prozessdaten im Double-Integer-Format
scale	OUT	Int	Prozessdaten-Skalierungsfaktor (abhängig vom Eingang "scaleIn" und vom angeschlossenen Sensor)
SSCn.1	OUT	Bool	Schaltinformation Kanal 1
SSCn.2	OUT	Bool	Schaltinformation Kanal 2

6.3.3. Funktionsweise

Mit dem Funktionsbaustein können zyklisch Messwerte der Sensorik erfasst werden. Die gemessenen Rohwerte der Sensoren werden verarbeitet und als Real- oder DInt-Messwerte an den jeweiligen Ausgängen wieder ausgegeben. In dem ausgegebenen Messwert wird bereits die Skalierung des Sensors über den Skalierungsfaktor aus der Prozessdaten-Eingabestruktur berücksichtigt und muss nicht extra ausgewertet werden, d. h. es werden sowohl die physikalische Einheit des Sensors als auch der Messwert interpretiert.

Der Eingang 'LogicStartAddress' ist die Logikadresse des ersten Bytes im Prozessdatenabbild der Eingänge und wird verwendet, um den Messwert, die Ausgänge 'scale' und 'SSCn.1'/'SSCn.2' direkt aus der Prozessdateneingangsstruktur zu lesen. Die Ausgänge 'SSCn.1' und 'SSCn.2' sind nur für SSP 4.1.x und 4.2.x relevant. Ansonsten sind 'SSCn.1' und 'SSCn.2' auf 0 gesetzt.

Der Eingang 'LogicStartAddress' kann entweder direkt über die Konfiguration im TIA Portal oder über das PCT Tool ausgelesen werden.

Der Eingang "profileId" bestimmt, welche Art der Smart Sensor Profile (SSP) vom IO-Link Device verwendet wird und ob die Länge des Messwertes, der aus dem Prozessabbild gelesen wird, ein 16-Bit- oder 32-Bit-Wert ist.

SSP 3.1 und SSP 3.3 verwenden die Prozessdatenstruktur "PDI32.INT16_INT8" (16 Bit Messwert), es gibt keinen Unterschied zwischen "profileId" = 1 und "profileId" = 3. Gleiches gilt für "profileId" = 2 (SSP 3.2) und "profileId" = 4 (SSP 3.4), da sie beide die Datenstruktur "PDI48.INT32_INT8" (32 Bit Messwert) verarbeiten. "profileId" = 5 (SSP 4.1.x) verwendet einen 16-Bit-Messwert (Struktur MSDC48) und "profileId" = 6 (SSP 4.2.x) verwendet einen 32-Bit-Messwert (Struktur MSDC48)

In der folgenden Tabelle sehen Sie eine Übersicht der Profiltypen 3 und 4 mit zugehöriger Prozessdatenstruktur.

Tabelle 6-12: Messgeräte-Profiltypen 3

Profil-Typ	Profil-ID	Name des Profilmerkmals	Funktionsklasse		Prozessdatenstruktur
			Messung	Deaktivierung Wandler	
SSP 3.1	0x000A	Messsensor	0x800A		PDI32.INT16_INT8
SSP 3.2	0x000B	Messsensor, hochauflösend	0x800B	-	PDI48.INT32_INT8
SSP 3.3	0x000C	Messsensor, Sperrfunktion	0x800A	0x800C	PDI32.INT16_INT8 PDO8.BOOL1
SSP 3.4	0x000D	Messsensor, hochauflösend, Sperrfunktion	0x800B		PDI48.INT32_INT8 PDO8.BOOL1
SSP 4.1.1	0x0010	Mess- und Schaltsensor, Kanal 1			PDI32.MSDC32_1
SSP 4.1.2	0x0011	Mess- und Schaltsensor, Kanal 2		0x800A	PDI64.MSDC32_2
SSP 4.1.3	0x0012	Mess- und Schaltsensor, Kanal 3			PDI96.MSDC32_3
SSP 4.1.4	0x0013	Mess- und Schaltsensor, Kanal 4			PDI128.MSDC32_4
SSP 4.2.1	0x0014	Mess- und Schaltsensor, hochauflösend, Kanal 1	0x800D 0x8010		PDI48.MSDC48_1
SSP 4.2.2	0x0015	Mess- und Schaltsensor, hochauflösend, Kanal 2		0x800B	PDI96.MSDC48_2
SSP 4.2.3	0x0016	Mess- und Schaltsensor, hochauflösend, Kanal 3			PDI144.MSDC48_3
SSP 4.2.4	0x0017	Mess- und Schaltsensor, hochauflösend, Kanal 4			PDI192.MSDC48_4

Wenn kein Fehler ansteht, wird der gelesene Messwert direkt auf den Ausgang "valueDInt" weitergeleitet. Der Wert am Ausgang "valueReal" wird berechnet aus: "measurement value" * 10 ^ "scale".

Es gibt mehrere Signalzustände, die während des Prozesses auftreten können:

Tabelle 6-13: Unterschiedliche Szenarios der Signalzustände

Fall	Signalzustände an den Eingängen	Signalzustände an den Ausgängen
1	"qualifier" = FALSE	"valid" = FALSE "valueStatus" = 1 (Prozessdaten ungültig) "valueReal" = Ersatzwert "valueDInt" = Ersatzwert "scale" = "scaleIn"
2	"qualifier" = TRUE "profilId" = 1, 2, 3, 4, 5 oder 6 unterer Grenzwert <= "measurementValue" <= oberer Grenzwert (siehe Tabelle 6-14)	"valid" = TRUE "valueStatus" = 0 (ok) "valueReal" = "measurementValue" * 10 ^ "scaleIn" "valueDInt" = "measurementValue" "scale" = "scaleIn"
3	"qualifier" = TRUE "profilId" = 1, 2, 3, 4, 5 oder 6 "measurementValue" = Außerhalb des Bereichs (+) (siehe Tabelle 6-15)	"valid" = FALSE "valueStatus" = 3 (Außerhalb des Bereichs (+)) "valueReal" = Ersatzwert "valueDInt" = Ersatzwert "scale" = "scaleIn"
4	"qualifier" = TRUE "profilId" = 1, 2, 3, 4, 5 oder 6 "measurementValue" = Außerhalb des Bereichs (-) (siehe Tabelle 6-15)	"valid" = FALSE "valueStatus" = 4 (Außerhalb des Bereichs (-)) "valueReal" = Ersatzwert "valueDInt" = Ersatzwert "scale" = "scaleIn"
5	"qualifier" = TRUE "profilId" = 1, 2, 3, 4, 5 oder 6 "measurementValue" = keine Messwerte (siehe Tabelle 6-15)	"valid" = FALSE "valueStatus" = 2 (Keine Daten) "valueReal" = Ersatzwert "valueDInt" = Ersatzwert "scale" = "scaleIn"
6	"qualifier" = TRUE "profilId" = 1, 2, 3, 4, 5 oder 6 "measurementValue" = Messwert entspricht keinem der Werte der Fälle 2-5.	"valid" = FALSE "valueStatus" = 5 (nicht definiert) "valueReal" = Ersatzwert "valueDInt" = Ersatzwert "scale" = "scaleIn"

Tabelle 6-14: Grenzen der Messwerte

Grenzwert	16 Bit	32 Bit
unterer Grenzwert	-32000	-2147482880
	16#8300	16#80000300
oberer Grenzwert	32000	2147482880
	16#7D00	16#7FFFFD00

Tabelle 6-15: Feste Sonderwerte (Ersatzwerte)

Grenzwert	16 Bit	32 Bit
Außerhalb des Bereichs (-)	-32760	-2147483640
	16#8008	16#80000008
Außerhalb des Bereichs (+)	32760	2147483640
	16#7FF8	16#7FFFFFF8
keine Messwerte	32764	2147483644
	16#7FFC	16#7FFFFC

6.3.4. Fehlerhandling

Im Fall eines Fehlers wird der Ausgang "error" gesetzt und der Ausgang "status" liefert die internen Fehlercodes des FB.

Tabelle 6-16: Fehlercodes am Ausgang "status"

status	Bedeutung	Erläuterung
16#0000	Kein Fehler	Es liegt kein Fehler an.
16#8001	Unbekannte Profil-ID oder falscher Datentyp	Überprüfen Sie die Profil-ID des Sensors oder den verwendeten Datentyp. Es wurde eine unbekannte Profil-ID verwendet oder der Datentyp passt nicht zur Profil-ID. Profil-ID1: WORD Profil-ID2: DWORD Profil-ID3: WORD Profil-ID4: DWORD

Über den Ausgang "valueStatus" werden Informationen über die Qualität der Prozessdaten geliefert. Wenn kein Fehler anliegt, ist "valueStatus" = 0.

Wenn ein Fehler anliegt, können folgende Zustände am Ausgang "valueStatus" anliegen:

Tabelle 6-17: Bedeutung Ausgang "valueStatus"

valueStatus	Beschreibung
1	Die Prozessdaten sind ungültig.
2	Es liegen keine Daten vor.
3	Die Prozessdaten liegen außerhalb des oberen Grenzwerts.
4	Die Prozessdaten liegen außerhalb des unteren Grenzwerts.
5	Der Status ist nicht definiert.

6.4. LIOLink_MultiAdjSwitchingSensor

6.4.1. Beschreibung

Der FB "LIOLink_MultiAdjSwitchingSensor" bietet eine einheitliche Schnittstelle für den Zugriff und die Parametrierung von IO Link Devices, die das Smart Sensor-Profil unterstützen.

Insbesondere kann der Baustein von Sensoren eingesetzt werden, die den folgenden Messgeräte-Profilen zugeordnet werden können:

- Typ 2.1 bis 2.6: Einstellbare Schaltsensoren mit einem Kanal
- Typ 2.7: Mehrkanalige einstellbare Schaltsensoren
- Typ 4.x: Mehrkanalige einstellbare Schaltsensoren

Der Funktionsbaustein (FB) kann verwendet werden, um Sollwerte einzustellen oder anzulernen und die Schaltpunktlogik eines einstellbaren Schaltsensors zu ändern. Das bedeutet, dass der Baustein mit IO-Link-Geräten kompatibel ist, die das Smart Sensor-Profil 'Multiple Adjustable Switching Sensors' unterstützen.

Der Funktionsblock stellt die Zustandsmaschinen für den Zugriff auf die profilspezifischen Parameter und die Abläufe für die drei Teach-Modi (Einzelwert Teach-In, Zweiwert Teach-In und dynamischer Teach-In) bereit. Die dargestellten Signale ermöglichen den Zugriff auf Funktionen für verschiedene Anwendungsfälle und Betriebsarten:

- Auswahl des Teach-Kanals
- Schaltsignal-Kanal-Parameter lesen
- Parameter des Schaltsignalkanals schreiben
- Einwertiger Teach
- Zweiwerte-Teach
- Dynamischer Teach-In

HINWEIS

Beachten Sie, dass bei mehrfachem gleichzeitigem Aufrufen des FB "LIOLink_MultiAdjSwitchingSensor" für den gleichen Master (z. B. Informationsabruf für mehrere Ports gleichzeitig) nur ein Bausteinaufruf erfolgreich beendet wird. An den anderen Bausteinen wird ein Statuskonflikt "diagnostics.status" = 16#8402 des unterlagerten FB "LIOLink_Device" ausgegeben (Sende- und Responseedaten inkonsistent).

6.4.2. Parameter

Abbildung 6-4: LIOLink_MultiAdjSwitchingSensor

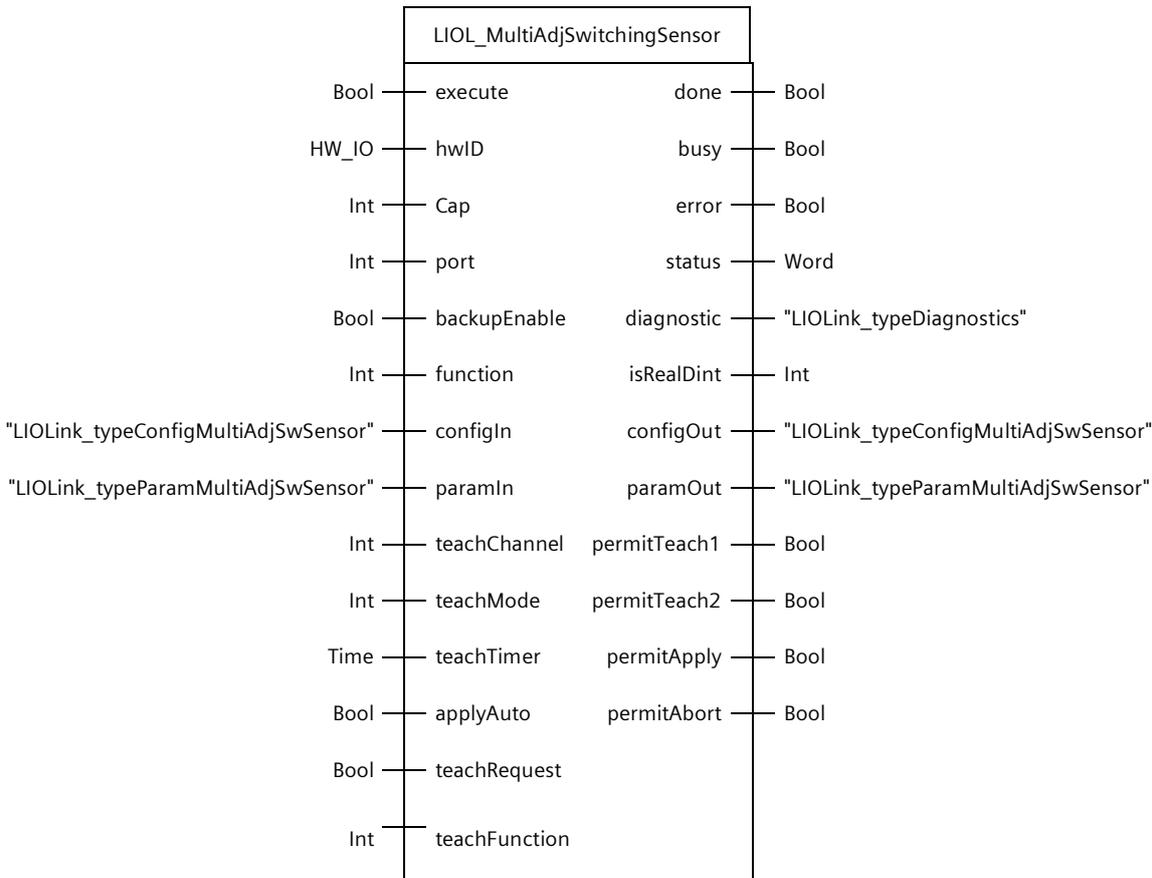


Tabelle 6-18: Parameter von LIOLink_MultiAdjSwitchingSensor

Name	P-Typ	Datentyp	Kommentar
execute	IN	Bool	Anforderung, die Funktion auszuführen
hwID	IN	HW_IO	Hardwarekennung des IO-Link Masters (des Submoduls für ET 200eco PN)
cap (optional)	IN	Int	Zugangspunkt (Client Access Point): Wird bei der Verwendung von Siemens IO-Link Mastern automatisch erkannt und ist daher optional. Wird der Parameter verändert, ist die automatische Erkennung nicht aktiv. Für gewöhnlich ist der Zugangspunkt 0xB400 oder 0x00E3 und ist im jeweiligen Handbuch angegeben.
port	IN	Int	Portnummer an dem das IO-Link Device betrieben wird Mögliche Werte: 0..63
backupEnable	IN	Bool	TRUE: über das Kommando "wr_ident" werden die an den Eingängen "applicationSpecificTagIn", "locationTagIn" und "functionTagIn" anliegenden Informationen im Device gespeichert FALSE: der Backup-Mechanismus wird nicht vom Baustein ausgeführt
function	IN	Int	Funktionsauswahl: 0: no_func 1: rd_all 2: wr_conf 3: wr_param 4: teach Siehe Kapitel 6.4.3.3

Name	P-Typ	Datentyp	Kommentar
configIn	IN	LIOLink_type ConfigMultiAdjSwSensor	Definiert die Werte für die Konfigurationseinstellungen, die bei einer Anforderung mit der Funktion 'wr_conf' in den Sensor geschrieben werden. Siehe Kapitel 6.4.3.1
paramIn	IN	LIOLink_type ParamMultiAdjSwSensor	Definiert die Werte für die neuen Sollwerte, die bei einer Anforderung mit der Funktion 'wr_param' in den Sensor geschrieben werden. Siehe Kapitel 6.4.3.1
teachChannel	IN	Int	Definiert den verwendeten SSP-Typ; SSP-Typen 2.1 bis 2.6, SSC = -1; SSC1.1=1; SSC1.2=2; SSC2.1=11; SSC2.2=12; SSC3.1=21; SSC3.2=22; SSC4.1=31; SSC4.2=32 Siehe Kapitel 6.4.3.4
teachMode	IN	Int	Auswahl des Teach-Vorgangs; 0: no_teach: Keine Teach-In-Aktion 1: single_value: Einzelwert Teach-In SP1 2: two_value: Zweiwert Teach-In SP1 3: dynamic: Dynamisches Teach-In SP1 Für SSP-Typen 2.7 und 4.x gilt zusätzlich: 11: single_value: Einzelwert Teach-In SP2 12: two_value: Zweiwert Teach-In SP2 13: dynamic: Dynamisches Teach-In SP2 Siehe Kapitel 6.4.3.5
teachTimer	IN	Time	Definiert die Dauer der dynamischen Teach-Zeit. Ein Wert von "0" deaktiviert die Aktivierung des automatischen Stoppbefehls. Die Teach-Funktion "teach_Stop" kann immer zum Auslösen des dynamischen Teach-Stopps verwendet werden und überschreibt damit den Teach Timer.
applyAuto	IN	Bool	Definiert das Verhalten bei einem zweiwertigen Teach-Vorgang. FALSE = automatische Übernahme deaktiviert Die Übernahmefunktion muss vom Anwender ausgelöst werden, um die gesammelten Teach-Points auszuwerten und den neuen Sollwert zu aktivieren. TRUE = automatische Übernahme aktiviert Wurden zwei Teach-Punkte erfolgreich eingelernt, wird ein automatisches Übernehmen ausgelöst. Es ist keine Aktivität des Anwenders erforderlich.
teachRequest	IN	Bool	Eine steigende Flanke löst einen Teach-Schritt entsprechend der gewählten Teach-Funktion am Eingang "teachFunction" aus.

Name	P-Typ	Datentyp	Kommentar
teachFunction	IN	Int	Teach-Funktion, die mit einer Teach-Anforderung am Eingang "teachRequest" ausgeführt wird: 0: keine Teach-Funktion 1: Starte Teach-Schritt 1 2: Starte Teach-Schritt 2 3: Zweiwertige Teach-Ergebnisse übernehmen 4: Abbruch der aktuellen Teach-Sequenz Siehe Kapitel 6.1.3.2
done	OUT	Bool	TRUE: Auftrag wurde erfolgreich ausgeführt
busy	OUT	Bool	TRUE: Auftrag wird gerade bearbeitet
error	OUT	Bool	TRUE: Ein Fehler ist bei der Bearbeitung des FB aufgetreten
status	OUT	Word	16#0000–16#7FFF: Status des FB 16#8000–16#FFFF: Fehlercodes (siehe Kapitel 6.1.4)
diagnostics	OUT	"LIOLink_type Diagnostics"	Detaillierte Diagnoseinformationen des FB (siehe Kapitel 6.1.4)
isRealDint	OUT	Int	Zeigt den Datentyp an, der für Konfigurations- und Sollwertparameter (Hyst, SP1, SP2) verwendet wird; je nach Sensor können diese Werte das Format Real oder Dint haben; 0: Funktionsklasse nicht unterstützt, 1: Real, 2: Dint
configOut	OUT	LIOLink_type ConfigMultiAdjSwSensor	Dieser Ausgang stellt den aktuellen Wert der Konfigurationsparameter des Sensors dar. Die Variablen werden jedes Mal aktualisiert, wenn ein Teach-Vorgang, ein Schreibvorgang oder ein Anforderungssignal mit der Funktion "rd_all" abgeschlossen ist. Siehe Kapitel 6.4.3.1
paramOut	OUT	LIOLink_type ParamMultiAdjSwSensor	Dieser Ausgang stellt die aktuellen Werte der Sollwerte des Sensors dar. Die Variablen werden mit der Funktion "rd_all" jedes Mal aktualisiert, wenn ein Teach-Vorgang, ein Schreibvorgang oder ein Anforderungssignal abgeschlossen ist. Siehe Kapitel 6.4.3.1
permitTeach1	OUT	Bool	Das Signal wird gesetzt, wenn gemäß dem anstehenden Status des FB ein Triggersignal für die Teach-Funktion "teach_1" möglich ist.
permitTeach2	OUT	Bool	Das Signal wird gesetzt, wenn gemäß dem anstehenden Status des FB ein Triggersignal für die Teach-Funktion "teach_2" möglich ist.
permitApply	OUT	Bool	Das Signal wird gesetzt, wenn gemäß dem anstehenden Status des FB ein Triggersignal für die Teach-Funktion "apply" möglich ist.
permitAbort	OUT	Bool	Das Signal wird gesetzt, wenn gemäß dem anstehenden Status des FB ein Triggersignal für die Teach-Funktion "abort" möglich ist.

6.4.3. Bedienung

6.4.3.1. Konfigurationseinstellungen und Sollwerte

Die Eingangsvariable "configIn" und die Ausgangsvariable "configOut" haben den Datentyp struct "LIOLink_typeConfigMultiAdjSwSensor". Dieser Datentyp besteht aus allen für einen Kanal möglichen Konfigurationsparametern.

Tabelle 6-19 "LIOLink_typeConfigMultiAdjSwSensor" ¹

LIOLink_typeConfigMultiAdjSwSensor		
Variable	Data type	Beschreibung
logic	Int	Schaltpunkt Logik; 0: high aktiv; 1: low aktiv;

¹ Hyst ist nicht relevant, wenn der Schaltpunkt Modus gleich Zweipunkt ist.

LIOLink_typeConfigMultiAdjSwSensor

Variable	Data type	Beschreibung
mode	Int	Schaltpunktmodus; 0: deaktiviert; 1: Einzelpunkt, 2: Fenster, 3: Zweipunkt;
hystDint	DInt	Schaltpunkt Hysterese (Dint)
hystReal	Real	Schaltpunkt Hysterese (Real)

Die Eingangsvariable "paramIn" und die Ausgangsvariable "paramOut" haben den Datentyp Struct "LIOLink_typeParamConfigMultiAdjSwSensor". Dieser Datentyp besteht aus allen für einen Kanal möglichen Sollwerten.

Tabelle 6-20

LIOLink_typeParamMultiAdjSwSensor

Variable	Data type	Beschreibung
sp1Dint	DInt	Sollwert 1 (DInt)
sp1Real	Real	Sollwert 1 (Real)
sp2Dint	DInt	Sollwert 2 (DInt)
sp2Real	Real	Sollwert 2 (Real)

Weitere Informationen über die Funktionen der einzelnen Konfigurationsparameter und Sollwerte sind zu finden unter dem Abschnitt "Profile" in [Webseite der IO Link Community](#).

Die Konfigurationsparameter "hyst" sowie die Sollwerte "sp1" und "sp2" können je nach unterstützter Funktionsklasse des Geräts entweder im Dint- oder Real-Format vorliegen.

Der FB erkennt, welche Funktionsklasse vom Gerät unterstützt wird, und liest bzw. schreibt diese Informationen im korrekten Format. Als hilfreicher Tipp, der Ausgang "isRealDint" zeigt an, in welchem Format das Gerät die Werte unterstützt.

Es wird empfohlen, zunächst eine rd_all-Funktion auszuführen, um alle Konfigurationen und Sollwerte auszulesen. Dadurch wird der Ausgang "isRealDint" aktualisiert, und anschließend können Sie die entsprechenden Variablen an das unterstützte Format anpassen.

6.4.3.2. Schaltpunktmodi

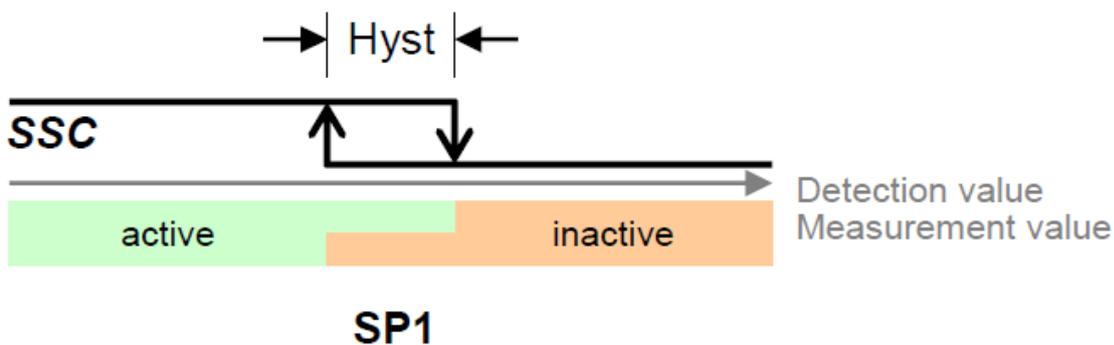
Der Parameter "mode" des Konfigurationsparameters "LIOLink_typeConfigMultiAdjSwSensor" definiert das Verhalten des Schaltsignals in Abhängigkeit von Sollwertparametern und dem aktuellen Erfassungs- oder Messwert. Es gibt vier verschiedene Schaltpunktmodi:

- Deaktivierungsmodus
- Einzelpunktmodus
- Fenstermodus
- Zweipunktmodus

Einzelpunktmodus

Im Einzelpunktmodus ändert sich der Schaltzustand, wenn der aktuelle Wert den Sollwert SP1 erreicht. Dieser Wechsel erfolgt bei steigenden oder fallenden Werten. Der Sollwert SP2 ist in diesem Modus nicht relevant. Dieser Modus wird typischerweise zur Objekterkennung oder zur Mengenerfassung verwendet.

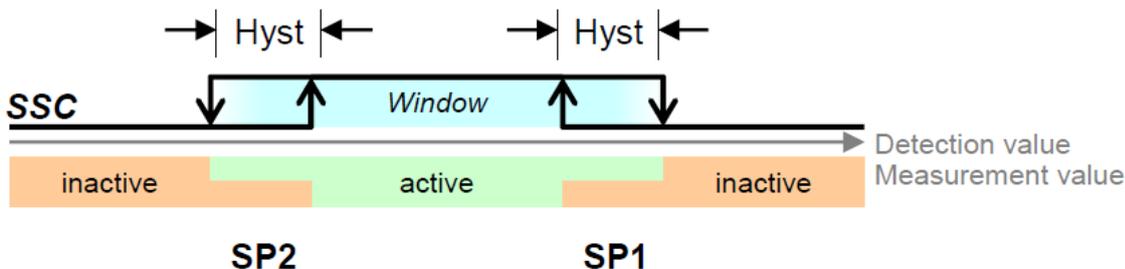
Abbildung 6-5 Beispiel für einen Einzelpunktmodus zur Objekterkennung²



Fenstermodus

Im Fenstermodus ändert sich der Schaltzustand, wenn der aktuelle Messwert entweder den Sollwert SP1 oder den Sollwert SP2 erreicht. Dieser Wechsel erfolgt bei steigenden oder fallenden Werten.

Abbildung 6-6 Beispiel für den Fenstermodus³



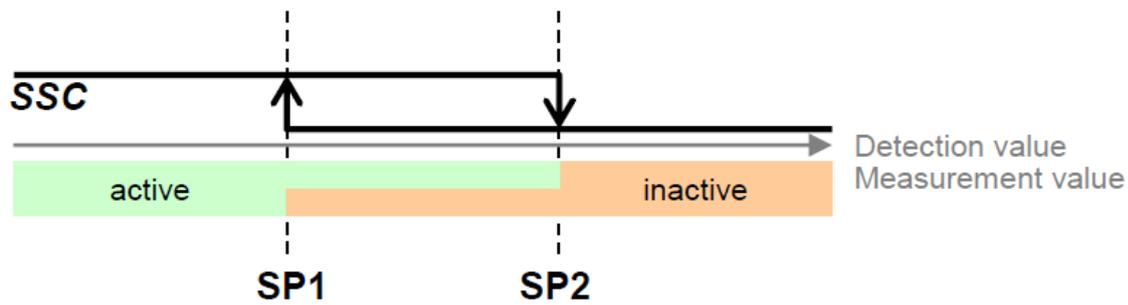
Zweipunktmodus (ohne Hysterese)

Im Zweipunktmodus ändert sich der Schaltzustand, wenn der aktuelle Messwert den Sollwert SP1 erreicht. Dieser Wechsel erfolgt nur bei steigenden Messwerten. Der Schaltzustand ändert sich auch, wenn der Stromwert den Sollwert SP2 erreicht. Dieser Wechsel erfolgt nur bei fallenden Messwerten. Die Hysterese wird in diesem Fall ignoriert.

Wenn der Messwert beim Einschalten des Smart Sensors zwischen SP1 und SP2 liegt, hängt das Verhalten von der hersteller-/anbieterspezifischen Auslegung des Geräts ab.

² Quelle: IO-Link Smart Sensor 2nd Edition, <https://io-link.com/de/Download/Download.php>

³ Quelle: IO-Link Smart Sensor 2nd Edition, <https://io-link.com/de/Download/Download.php>

Abbildung 6-7 Beispiel für den Zwei-Punkt-Modus für die Objekterkennung⁴

Deaktivierungsmodus

Der Schaltzustand im Deaktivierungsmodus muss "inaktiv" sein.

6.4.3.3. Funktionen

Der Baustein bietet fünf Funktionen. Eine Funktion kann durch Angabe der entsprechenden Nummer am Eingang "function" ausgewählt werden. Eine steigende Flanke am Eingang "execute" löst die ausgewählte Funktion aus.

- **no_func (0)**
Es wird keine Funktion ausgeführt.
- **rd_all (1)**
Bei dieser Funktion werden die aktuellen Schaltsignale und Parameterwerte vom Sensor gelesen. Die gelesenen Werte stehen an den Ausgängen "configOut" und "paramOut" zur Verfügung.
- **wr_conf (2)**
Diese Funktion bewirkt, dass die zuvor angelegten Werte innerhalb der "configIn" in den Sensor geschrieben wird.
- **wr_param (3)**
Diese Funktion bewirkt, dass die zuvor angelegten Werte innerhalb der "paramIn" in den Sensor geschrieben wird.
- **teach (4)**
Diese Funktion bewirkt, dass der Baustein in den Teach-Vorgang übergeht (siehe Kapitel [Teach-In](#)).

Wenn der Eingang "backupEnable" gesetzt ist, wird der IO-Link Systembefehl "ParameterDownloadStore" aufgerufen. Dadurch wird der Datenspeichermechanismus (DataStorage) gestartet und die neue Parametrierung im IO-Link Master gespeichert.

Für jede Funktion wird der "teachChannel" verwendet, um zuerst den Kanal auszuwählen und dann die Funktion zu bearbeiten. Die Auswahl des Kanals und die Bearbeitung der Funktion erfolgen in einer Ausführung (in einer aufsteigenden Flanke im Eingang "execute") des Bausteins. Mehr Informationen zu "teachChannel" sind hier zu finden [6.4.3.4](#).

6.4.3.4. Kanal einlernen (Teach Channel)

Der Baustein ermöglicht die Auswahl des Kanals, von dem je nach gewählter Funktion Sollwerte, Konfigurationsparameter und Teachpunkte zum/vom Sensor gelesen/geschrieben werden sollen. Ein Kanal kann durch Eingabe der entsprechenden Nummer am Eingang "teachChannel" ausgewählt werden.

Der Kanal hängt von dem vom Sensor unterstützten SSP-Typ (Smart Sensor Profile) ab.

- Switching Signal Channel (SSC) = -1: gilt nur für die SSP-Typen 2.1 bis 2.6
- SSC1.1=1;

⁴ Quelle: IO-Link Smart Sensor 2nd Edition, <https://io-link.com/de/Download/Download.php>

- SSC1.2=2;
- SSC2.1=11;
- SSC2.2=12;
- SSC3.1=21;
- SSC3.2=22;
- SSC4.1=31;
- SSC4.2=32

Weitere Informationen über SSP-Typen und ihre Eigenschaften sind auf dem Abschnitt "Profile" der [IOLink-Webseite](#) zu finden.

6.4.3.5. Teach-In

Übersicht

In diesem Abschnitt werden Erläuterungen zu den unterschiedlichen Teach-Funktionen gegeben. Über die Teach-In-Funktionen werden Sollwerte zum Schalten des Schaltausgangs am Sensor eingelernt. Es sind verschiedene Teach-In-Verfahren, wie Einzelwert Teach-In, Zweiwert Teach-In oder dynamischer Teach-In möglich, was die Inbetriebnahme der Anwendung erleichtert. Je nach Sensortyp sind individuelle Kombinationen dieser Teach-In-Verfahren möglich.

Es stehen sieben Teach-Vorgänge zur Verfügung, die über den Eingang "teachMode" ausgewählt werden können:

- **no_teach (0):** Keine Teach-In-Aktion
- **single_value (1):** Einzelwert Teach-In
- **two_value (2):** Zweiwert Teach-In
- **dynamic (3):** Dynamisches Teach-In
- Für SSP-Typen 2.7 und 4.x gilt zusätzlich:
- **single_value (11):** Einzelwert Teach-In SP2
- **two_value (12):** Zweiwert Teach-In SP2
- **dynamic (13):** Dynamisches Teach-In SP2

Während eines Teach-Vorgangs, werden die einzelnen Teach-Schritte über die Eingänge "teachRequest" und "teachFunction" gesteuert.

Die unterschiedlichen Teach-Mechanismen werden im nächsten Abschnitt näher erläutert.

Der Ausgang "status" stellt Informationen zum aktuell anstehenden Einlernschritt zur Verfügung. Wird eine "teachFunction" angefordert, die zu diesem Zeitpunkt nicht erlaubt ist, dann stoppt der Funktionsblock den Teach-Vorgang und gibt einen Fehler zurück.

Bevor eine Teach-In-Funktion gestartet wird, muss zuerst der Schalterpunktmodus für den Kanal konfiguriert werden. Je nach Schalterpunktmodus sind einige Teach-Prozeduren nicht relevant/erlaubt, z.B. ist der Einzelwert-Teach für SP2 im Einzelpunktmodus nicht relevant.

Der Teach-Kanal muss immer gewählt werden, wenn ein Teach-In-Vorgang durchgeführt wird. Der Teach-In-Vorgang wird auf dem gewählten Kanal ausgeführt.

Einzelwert Teach-In

Nachfolgend werden die Schritte für ein Einzelwert Teach-In am SSC2.1 aufgeführt:

1. Eingänge "teachChannel" = 11 (SSC2.1), "function" = 4 und "teachMode" = 1 setzen.

2. Eingang "execute" = TRUE setzen.
3. Abwarten bis Ausgänge "permitTeach1" = TRUE und "status" = 16#7110 (Teach-Prozess wartet auf weiteren Befehl).
4. Objekt in die gewünschte Position bringen.
5. Eingang "teachFunction" = 1 setzen.
6. Eingang "teachRequest" = TRUE setzen.

Der Teach-Vorgang war erfolgreich, wenn Ausgänge "done" = TRUE und "status" = 16#0000.

Zweiwert Teach-In

Nachfolgend werden die Schritte für einen Zweiwert Teach-In am SSC1.2 aufgeführt:

1. Eingänge "teachChannel" = 2 (SSC1.2), "function" = 4 und "teachMode" = 2 setzen.
2. Eingang "execute" = TRUE setzen.
3. Abwarten bis Ausgänge "permitTeach1" = TRUE, "permitTeach2" = TRUE und "status" = 16#7120 (Teach-Prozess wartet auf weiteren Befehl).
4. Objekt in die gewünschte Position für Teach-Point 1 bringen.
5. Eingang "teachFunction" = 1 setzen.
6. Eingang "teachRequest" = TRUE setzen.
7. Abwarten bis Ausgänge "permitTeach1" = TRUE, "permitTeach2" = TRUE und "status" = 16#7120 (Teach-Prozess wartet auf weiteren Befehl).
8. Objekt in die gewünschte Position für Teach-Point 2 bringen.
9. Eingang "teachFunction" = 2 setzen.
10. Eingang "teachRequest" = FALSE setzen.
11. Eingang "teachRequest" = TRUE setzen.
12. Wenn "applyAuto" = FALSE, muss "teachApply" manuell ausgelöst werden:
 - a. Abwarten bis Ausgänge "permitApply" = TRUE und "status" = 16#7140 (Teach-Prozess wartet auf Befehl zur Übernahme der eingelernten Werte).
 - b. Eingang "teachFunction" = 3 setzen.
 - c. Eingang "teachRequest" = FALSE setzen.
 - d. Eingang "teachRequest" = TRUE setzen.

Wenn "applyAuto" = TRUE, wird "teachApply" automatisch ausgelöst.

Der Teach-Vorgang war erfolgreich, wenn Ausgänge "done" = TRUE und "status" = 16#0000.

Alternativ kann "teachPoint2" auch zuerst eingelernt werden.

Ein Teach-Point kann auch mehrfach während des zweiwertigen Tech-In-Prozesses eingelernt werden.

Ein Abbruch der Teach-Funktion ist mit "teachFunction" = 4 nach dem Einlernen des ersten Teachpunkts verfügbar. Den Teach-Vorgang können Sie abbrechen, indem Sie "teachFunction" = 4 setzen und eine steigende Flanke am Eingang "teachRequest" durchführen.

Dynamisches Teach-In

Nachfolgend werden die Schritte für ein dynamisches Teach-In am SSC2.2 aufgeführt:

1. Eingänge "teachChannel" = 12 (SSC2.2), "function" = 4 und "teachMode" = 3 setzen (Dynamisches Teach-In).
2. Eingang "execute" = TRUE setzen.

3. Abwarten bis Ausgänge "permitTeach1" = TRUE und "status" = 16#7130 (Teach-Prozess wartet auf weiteren Befehl).
4. Eingang "teachFunction" = 1 setzen.
5. Eingang "teachRequest" = TRUE setzen.
6. Abwarten bis Ausgang "status" = 16#7131 (Teach-Prozess ist aktiv).
Der dynamische Einlernvorgang hat begonnen.
7. Objekt innerhalb des gewünschten Bereichs verschieben.
8. Wenn "teachTimer" = 0 s, muss der Teach-Point 2 manuell angestoßen werden:
 - a. Abwarten bis Ausgang "permitTeach2" = TRUE.
 - b. Eingang "teachFunction" = 2 setzen.
 - c. Eingang "teachRequest" = FALSE setzen.
 - d. Eingang "teachRequest" = TRUE setzen.

Wenn "teachTimer" ≠ 0 s, wird der Teach-Point 2 automatisch angestoßen und der Teach-Prozess endet, wenn die Zeit abgelaufen ist. Der Timer startet nach dem erfolgreichen Senden des Teach-Auftrags an das IO-Link Device.

Der Teach-Vorgang war erfolgreich, wenn Ausgänge "done" = TRUE und "status" = 16#0000.

Ein Abbruch der Teach-Funktion ist nach dem Start des Teach-Vorgangs verfügbar. Diesen können Sie durch "teachFunction" = 4 und eine steigende Flanke am Eingang "teachRequest" abbrechen.

7. Kopiervorlagen

7.1. LIOLink_MeasuredData

7.1.1. Beschreibung

Dieser FB dient zum Lesen und Verarbeiten von Prozessdaten, unabhängig davon, welches Device am IO-Link Master angeschlossen ist. Der FB findet zuerst das angeschlossene Device heraus und liest und bereitet dann dessen Prozessdaten entsprechend auf.

Unterstützte Geräte für diesen FB sind in einer "Datenbank" in Form eines statischen Arrays vom Datentyp "Struct" zu finden. Von jedem Gerät oder Sensor sind einige Informationen notwendig, damit der FB korrekt arbeitet und das Gerät unterstützt, wie z.B. Vendor ID, Device ID, Offset, Scale.

Im Falle eines Devicefehlers oder einer bestimmten Operation, wie z.B. bei einem Werkzeugwechsler (siehe Kapitel 4.1), kann der Benutzer einen Ersatzwert angeben, der dann als Prozesswert für den spezifischen Sensor verwendet wird.

Dieser FB dient als Master-Template und kann vom Benutzer auf der Grundlage seines Anwendungsfalls und seiner Sensoren und Devices angepasst werden.

Dieser FB kann im Falle eines Sensortausches hilfreich sein, insbesondere dann, wenn verschiedene Sensoren desselben oder unterschiedlicher Hersteller ersetzt werden müssen. Wenn ein anderer Sensortyp eingesetzt wird, muss dieser zwar neu parametrisiert werden, aber der FB würde den neuen Sensortyp automatisch erkennen und dessen Prozesswert auslesen, solange der neue Sensortyp vom FB unterstützt wird. Somit wäre ein Sensortausch mit einem anderen Sensortyp ohne Änderungen im Anwenderprogramm oder TIA-Projekt möglich.

Gültigkeit

Der FB steht nur für S7-1200/1500 zur Verfügung.

Voraussetzungen

Die Ports des IO-Link Masters, die verwendet werden, müssen in der Gerätekonfiguration des IO-Link Masters aktiviert werden (Port Enable).

Wegen der Auswertung des PQI ist es zwingend erforderlich, die Prozessdatenlänge des Ports auf IO-Link 32 I / 32 O + PQI (maximum) einzustellen. Damit bereitet auch eine unterschiedliche Prozessdatenlänge von verschiedenen Sensoren bei deren Austausch keine Probleme.

7.1.2. Parameter

Abbildung 7-1: LIOLink_MeasuredData

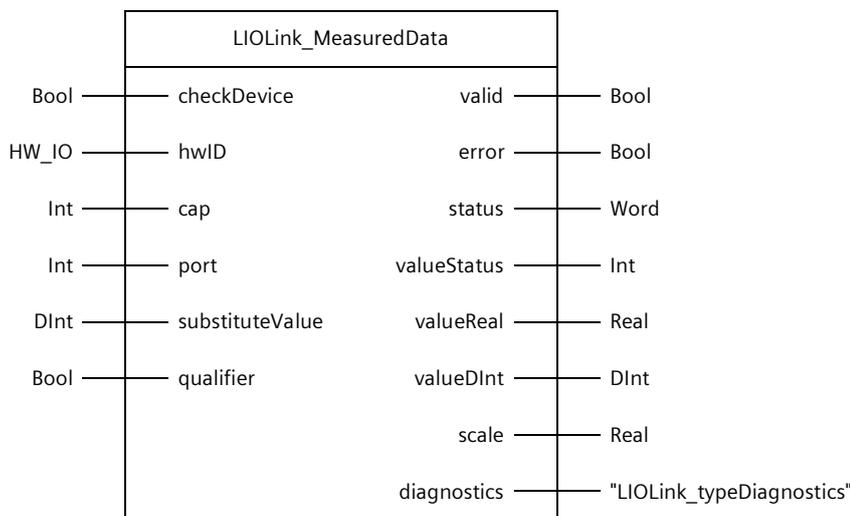


Tabelle 7-1: Parameter von LIOLink_MeasuredData

Name	P-Typ	Datentyp	Kommentar
checkDevice	IN	Bool	Prüfen des angeschlossenen Geräts anfordern (falls ein neues Gerät angeschlossen ist)
hwID	IN	HW_IO	Hardware-Identifizier des Ports des IO-Link Masters
cap (optional)	IN	Int	Zugangspunkt (Client Access Point): Wird bei der Verwendung von Siemens IO-Link Mastern automatisch erkannt und ist daher optional. Wird der Parameter verändert, ist die automatische Erkennung nicht aktiv. Für gewöhnlich besitzt er den Wert 0xB400 oder 0x00E3 und ist im jeweiligen Handbuch angegeben.
port	IN	Int	Portnummer an dem das IO-Link Device betrieben wird
substituteValue	IN	DInt	Dieser Wert wird an den Ausgängen ValueReal und ValueDInt verwendet, wenn ValueStatus ungleich 0 ist ValueReal := DINT_TO_REAL(substituteValue)
qualifier	IN	Bool	Qualifier der Prozessdaten FALSE: ProcessData sind ungültig TRUE: ProcessData sind ungültig (siehe Kapitel 7.1.3)
valid	OUT	Bool	TRUE: Auftrag wurde erfolgreich ausgeführt
error	OUT	Bool	TRUE: Auftrag wird gerade bearbeitet
status	OUT	Word	16#0000–16#7FFF: Status des FB 16#8000–16#FFFF: Fehlercodes (siehe Kapitel 3.1.4)
valueStatus	OUT	Int	Status der Prozessdateneingabe: 0 = OK; 1 = PD ungültig; 2 = Keine Daten; 3 = Außerhalb des Bereichs (+); 4 = Außerhalb des Bereichs (-); 5 = Nicht definiert;
valueReal	OUT	Real	Prozesswert im Real-Format zur Auswertung in der SPS
valueDInt	OUT	DInt	Prozesswert im Double-Integer-Format zur Auswertung in der SPS
scale	OUT	Real	Skalierungsfaktor der Prozessdaten
diagnostics	OUT	"LIOLink_ typeDiagnostics"	Detaillierte Diagnoseinformationen des FB (siehe Kapitel 3.1.4)

7.1.3. Funktionsweise

Qualifier

Der Eingang "qualifier" zeigt die Gültigkeit der Daten an, wobei FALSE signalisiert, dass die Prozessdaten ungültig sind und der Ersatzwert (Eingang "substituteValue") stattdessen verwendet wird, während TRUE bedeutet, dass die Prozessdaten gültig sind. Zum Beispiel muss der Eingang "qualifier" auf FALSE gesetzt werden, wenn der Port-Betrieb unterbrochen (siehe Kapitel [4.2.1](#)) werden soll, damit der Ersatzwert anstelle der Prozessdaten genutzt wird. Ebenso gilt dies, wenn ein Fehler auftritt oder wenn Wartungsarbeiten durchgeführt werden müssen; in solchen Fällen wird der Eingang "qualifier" auf FALSE gesetzt, um sicherzustellen, dass die ungültigen Prozessdaten durch den Ersatzwert ersetzt werden.

Device-Erkennung

Der FB versucht zuerst, das Device zu finden bzw. zu erkennen, das an dem gegebenen Port (Eingang "port") angeschlossen ist. Die Device-Erkennung wird beim ersten CPU-Zyklus angestoßen oder, wenn der Eingang "checkDevice" ausgelöst wird.

Der FB benutzt den FB LIOLink_Device, um die Device ID und Vendor ID des Devices zu lesen. Auf diese Weise erkennt er das angeschlossene Device. Dann wird versucht, diese Informationen, Device ID und Vendor ID, in der "Sensordatenbank" (statisches Array "statSensors") zu finden.

Die statische Variable in Form eines Arrays "statSensors" ist ein Array mit verschiedenen Sensortypen. Jedes Element dieses Arrays repräsentiert einen Sensortyp. Jeder Sensortyp hat die gleichen Informationen, die zu seiner Identifizierung und zur Verarbeitung der Prozessdaten benötigt werden. Diese Informationen sind in dem Datentyp "LIOLink_typeSensorInfo" strukturiert.

Tabelle 7-2 "LIOLink_typeSensorInfo"

LIOLink_typeSensorInfo		
Variable	Data type	Beschreibung
vendorID	DInt	Hersteller-ID des Sensors
deviceID	DInt	Geräte-ID des Sensors
offset	Int	Offset für den Wert vom Sensor
scale	Real	Skalierung des Werts vom Sensor Defaultwert: 1.0 (= keine Skalierung)
fbState	DInt	FBState, in dem die Vorverarbeitung der IO-Daten dieses spezifischen Sensors durchgeführt wird. FBState muss für jeden Sensor eindeutig sein.

Nachdem das Gerät im "statSensors"-Array gefunden wurde, indem die ausgelesene Geräte-ID und Hersteller-ID mit den entsprechenden Informationen im Array verglichen wurden, wird das Element, das exakt dieselben Informationen enthält, der statischen Variable "statCurrentSensor" zugewiesen. Diese Variable repräsentiert dann den aktuell angeschlossenen Sensor und wird zur Verarbeitung der Prozessdaten verwendet.

Da der Prozess der Geräteerkennung ein azyklischer Vorgang ist und mehrere Zyklen dauern kann, wird der Ersatzwert (Eingang "substituteValue") den Prozesswerten oder Ausgängen zugewiesen. Dieses Szenario bleibt gültig, bis die azyklischen Aufrufe abgeschlossen sind.

Tabelle 7-3: Ausgangszustände während eines Geräteerkennungsverfahrens

Output Variable	Value
valid	FALSE
error	FALSE
status	16#7003
valueStatus	1 (PD Invalid)
valueReal	DINT_TO_REAL(substituteValue)
valueDInt	substituteValue
scale	1.0

Auslesen und Verarbeitung von Prozessdaten

Der FB verwendet intern den Befehl GETIO und liest über den Eingang "hwID" das gesamte Prozessabbild der Eingänge eines Ports, also 33 Byte (32 Byte Eingang + PQI). Das gesamte Prozessabbild wird dann in einer statischen Variablen "statiOData" gespeichert.

Nach dem Lesen des Prozessabbilds wird der PQI ausgewertet, um zu prüfen, ob die IO-Link-Daten gültig sind. Wenn die Daten nicht gültig sind, geht der FB in den Fehlerzustand und der Ersatzwert wird als Prozesswert verwendet.

HINWEIS

Für den Eingang "hwID" verwenden Sie den Hardware Identifier des Ports des IO-Link Masters, an dem der Sensor angeschlossen ist. Jeder Port des IO-Link Masters hat einen Hardware Identifier.

Wenn die IO-Link-Daten gültig sind, wechselt der FB zu einem Zustandsautomaten, bei dem jeder Zustand einem bestimmten Sensortyp entspricht. In jedem dieser Zustände werden die Sensordaten aus dem gesamten (zuvor mit GETIO ausgelesenen) Prozessabbild extrahiert und je nach Sensortyp entsprechend weiterverarbeitet.

Da verschiedene Sensoren Prozessdaten an unterschiedlichen Positionen im Prozessabbild und mit unterschiedlichen Genauigkeiten, Offsets und Skalierungen liefern können, wird für jeden Sensortyp ein separater Zustand verwendet. Dies ermöglicht eine hohe Flexibilität, da das System so in der Lage ist, unterschiedliche Sensortypen nahtlos in die gleiche Anwendung zu integrieren.

Der Zustand, in dem der Automat arbeitet, wird direkt durch den erkannten Sensor bestimmt, und der aktive Zustand wird in der statischen Struktur "statCurrentSensor" gespeichert.

Integration von neuen Sensoren in LIOLink_MeasuredData

In der FB sind zwei Sensoren als Beispiel integriert, um zu zeigen, wie es aussehen sollte. Diese beiden Sensoren sind von unterschiedlichen Typen und Anbietern. Ihr Prozesswert befindet sich an unterschiedlichen Positionen im Prozessabbild und sie haben eine unterschiedliche Skalierung.

Hier wird gezeigt, wie man einen neuen Sensor in den LIOLink_MeasuredData FB integriert:

1. Die Konstante "NUMBER_OF_SENSORS" um 1 erhöhen. Diese Konstante wird verwendet, um den Bereich des oben erwähnten Arrays "statSensors" zu bestimmen. Damit wird ein neuer Sensor zur Sensor-"Datenbank" "statSensors" hinzugefügt. Dieser Sensor wird am Ende des Arrays eingefügt, das heißt, er ist das letzte Element des Arrays.
2. Eine neue Konstante FB_SENSOR_STATE_XYZ hinzufügen, die als neuer Zustand in der Zustandsmaschine für den neuen Sensor verwendet wird. Wie oben erwähnt, muss jeder Sensor eine eindeutige Zustandsnummer haben, daher muss der Wert des FB_SENSOR_STATE_XYZ im Vergleich zu anderen Zuständen eindeutig sein.

Abbildung 7-2: Vor/Nach der Ausführung von Schritt 1 & 2

Vor der Ausführung von Schritt 1 & 2:

49	Constant							
50	FB_SENSOR_STATE_3RS2	DInt	1					State for sensor 3RS2 (statSensor[1])
51	FB_SENSOR_STATE_TCC511	DInt	2					State for sensor TCC511 (statSensor[2])
52	NUMBER_OF_SENSORS	Int	2					Number of all possible sensors

Nach der Ausführung von Schritt 1 & 2:

49	Constant							
50	FB_SENSOR_STATE_3RS2	DInt	1					State for sensor 3RS2 (statSensor[1])
51	FB_SENSOR_STATE_TCC511	DInt	2					State for sensor TCC511 (statSensor[2])
52	FB_SENSOR_STATE_XYZ	DInt	3					State for sensor XYZ
53	NUMBER_OF_SENSORS	Int	3					Number of all possible sensors

3. Der neue Sensor, der in Schritt 1 zu "statSensors" hinzugefügt wurde, hat Default-Werte. Diese Werte müssen wir mit denen des neuen Sensors ändern. Hier sind vendorID, deviceID, offset, scale erforderlich und können auf der IODfinder-Website gefunden werden. Außerdem ist fbState erforderlich und entspricht dem neuen Zustand, den im Schritt 2 hinzugefügt wurde.
4. Fügen Sie den in Schritt 2 erstellten neuen Zustand zum Zustandsautomaten in der Region "Prepare Measured Data" hinzu. Zuerst extrahieren Sie den Prozesswert aus statIOData, wobei zu berücksichtigen ist, dass die Position des Wertes je nach Sensor variieren kann. Konvertieren Sie den Prozesswert entsprechend seiner Genauigkeit und seines Datentyps. Sie können die temporären Variablen tempWordMeasurementValue und tempIntMeasurementValue verwenden, um die Datenextraktion und -konvertierung zu unterstützen.

Sobald der Wert verarbeitet ist, weisen Sie ihn den statischen Ausgangsvariablen statValueDInt und statValueReal zu. Das statValid-Flag sollte auf true gesetzt werden, und statValueStatus sollte aktualisiert werden, um zu zeigen, dass die Werte gültig sind. Die Gültigkeit (statValid) und der Status (statValueStatus) des Prozesswerts sollten jedoch auf der Grundlage bestimmter Kriterien bestimmt werden, z. B. ob der Wert innerhalb des erwarteten Bereichs liegt. Die Behandlung von Szenarien, wie z. B. Werte, die außerhalb des erwarteten Bereichs liegen, liegt in der Entscheidung des Anwenders, wie er mit solchen Situationen am besten umgeht.

Mögliche Werte für diese Ausgänge sind im Kommentarbereich zu sehen (siehe Kapitel [7.1.2](#)).

7.1.4. Fehlerhandling

Der Baustein gibt die Status- und Diagnoseinformationen vom intern aufgerufenen FB "LIOLink_Device" an den Ausgängen "status" und "diagDeviceFB" aus (siehe Kapitel [3.1.3](#)).

Status

Zusätzlich zu den Statusinformationen, die aus "LIOLink_Device" ausgelesen werden, hat der FB die folgenden zusätzlichen Statuscodes.

Tabelle 7-4: Ausgang "status" von LIOLink_MeasuredData

Status	Bedeutung
16#0000	Auftrag abgeschlossen, keine Warnung und keine weitere Detaillierung
16#7000	Kein Auftrag in Bearbeitung (Initialwert)
16#7001	Erster Aufruf nach Eingang eines neuen Auftrags (steigende Flanke 'checkDevice')
16#7002	Nachfolgender Aufruf während aktiver Bearbeitung ohne weitere Angaben
16#7003	Untergeordneter FB (LIOLink_Device) ist busy, Maschinenstatus ist FB_STATE_ENABLING
16#7004	Lesen des Prozessabbildes ist aktiv und erfolgreich
16#8600	Fehler aufgrund eines undefinierten Zustands im Zustandsautomaten
16#8601	Fehler aufgrund eines undefinierten Zustands in der Zustandsmaschine; Sensorzustand wurde in FB_STATE_ENABLING nicht gefunden;
16#8603	IO Link Master hat einen Fehlercode zurückgemeldet, siehe "diagnostics"; Ungültige IO-Link Daten
16#8604	IO Link Master hat einen Fehlercode zurückgemeldet, siehe "diagnostics"
16#8605	Qualifier ist falsch

8. Wissenswertes

8.1. Grundlagen IO-Link

Einführung

Die Punkt-zu-Punkt-Schnittstelle IO-Link, über die Prozess- und Diagnosedaten, sowie Parameter übertragen werden, wurde speziell für den Anschluss beliebiger Sensoren und Aktoren an ein Steuerungssystem entwickelt. Dabei bedient sich IO-Link nicht der klassischen Busverdrahtung, sondern behält die für Sensoren und Aktoren übliche Parallelverdrahtung bei.

Der Kommunikationsstandard IO-Link ist unterhalb der Feldbusebene angesiedelt. Er ermöglicht eine zentrale Fehlerdiagnose und -ortung bis zur Sensor-/Aktorebene und erleichtert die Inbetriebsetzung und die Instandhaltung, indem sich Parameterdaten direkt aus der Applikation heraus dynamisch ändern lassen.

Als offene Schnittstelle lässt sich IO-Link in alle gängigen Feldbus- und Automatisierungssysteme integrieren. Konsequente Interoperabilität sorgt dabei für hohen Investitionsschutz. Dies gilt auch im Rahmen von bestehenden Maschinenkonzepten für die weitere Nutzung von Sensoren, die über keine IO-Link-Schnittstelle verfügen.

8.1.1. Was ist IO-Link?

Überblick

"IO-Link ist die erste, weltweit standardisierte IO-Technologie (IEC 61131-9) um mit Sensoren und auch Aktoren zu kommunizieren. Die leistungsfähige Punkt-zu-Punkt Kommunikation basiert dabei auf dem schon lange bekannten 3-Leiter Sensor und Aktor Anschluss ohne weitere zusätzliche Anforderungen an das Kabelmaterial. IO-Link ist somit kein Feldbus, sondern die evolutionäre Weiterentwicklung der bisherigen, erprobten Anschlusstechnik für Sensoren und Aktoren."⁵

Hersteller

IO-Link wird von vielen namhaften Herstellern, wie auch Siemens, in einem technischen Komitee gefördert.

Anschluss

Die IO-Link Kommunikation kann über bereits vorhandene Leitungen für digitale Ein- und Ausgänge in 3-Leiter Technik angeschlossen werden. Dadurch entsteht ein homogener und deutlich reduzierter Verdrahtungsaufwand.

8.1.2. Projektierung des IO-Link Masters

Die Konfiguration des IO-Link Masters und der daran angeschlossenen Devices erfolgt im Port Configuration Tool, kurz: S7-PCT:

<https://support.industry.siemens.com/cs/ww/de/view/32469496>

Das S7-PCT kann entweder Stand-Alone betrieben oder aus einem bestehenden TIA Portal-Projekt heraus aufgerufen werden.

Um das S7-PCT aus STEP 7 V16 heraus zu öffnen, haben Sie in der Gerätesicht des IO-Link Masters die folgenden Möglichkeiten:

⁵ <https://io-link.com/>

- Klicken Sie in der Gerätesicht im Editorbereich mit Rechtsklick auf das IO-Link Master Modul und wählen Sie "Device-Tool starten..." ("Start device tool...").



- Klicken Sie in der Geräte-Auflistung mit Rechtsklick auf das IO-Link Master Modul und wählen Sie "Device-Tool starten..." ("Start device tool...").



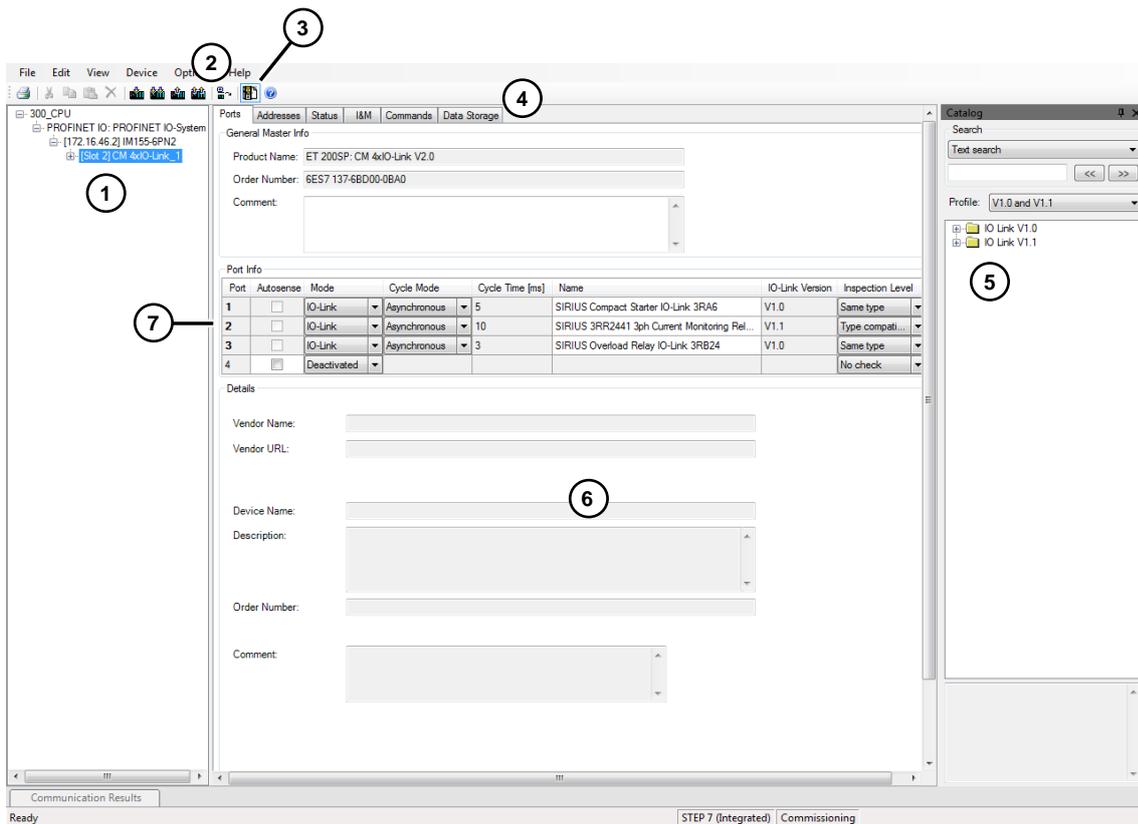
Übersicht und Beschreibung der Benutzeroberfläche von S7-PCT

Das S7-PCT erfüllt die Funktionen:

- Zuordnung der jeweiligen Sensoren und Aktoren (z.B. IO-Link Devices) zu den Ports des IO-Link Masters.
- Zuordnung der Peripherie-Adressen der IO-Link Devices zur S7-CPU.
- Festlegen der Parametrierung der Devices.

Die nachfolgende Abbildung zeigt ein bereits konfiguriertes IO-Link Master-Modul:

Abbildung 8-1

**Legende:**

1. Projektbaum
2. Schaltflächen für Up- und Download der aktuellen IO-Link Konfiguration
3. Schaltfläche für Zu- und Abschalten des Produktkatalogs
4. Registerkarten
5. Produktkatalog für IO-Link Devices
6. Arbeitsbereich
7. Portnummern mit parametrisierten IO-Link Devices

8.2. Bibliotheken im TIA Portal

Der Großteil der Bausteine ist als Typ in der Bibliothek abgelegt. Somit sind die Bausteine versioniert und können folgende Vorteile nutzen:

- Zentrale Updatefunktion von Bibliothekselementen
- Versionierung von Bibliothekselementen

HINWEIS

Informationen zum generellen Umgang mit Bibliotheken finden Sie im Leitfaden zur Bibliothekshandhabung:

<https://support.industry.siemens.com/cs/ww/de/view/109747503>

HINWEIS

Alle Bausteine in der Bibliothek wurden nach dem Programmierstyleguide erstellt:

<https://support.industry.siemens.com/cs/ww/de/view/81318674>

Weitere Informationen zu Bibliotheken im TIA Portal:

- Wie können Sie globale Bibliotheken im TIA Portal öffnen, bearbeiten und hochrüsten?
<https://support.industry.siemens.com/cs/ww/de/view/37364723>
- In weniger als 10 Minuten TIA Portal: Time Savers – Globale Bibliotheken
<https://support.industry.siemens.com/cs/ww/de/view/78529894>
- Welche Elemente aus STEP 7 (TIA Portal) und WinCC (TIA Portal) können in einer Bibliothek als Typ oder als Kopiervorlage abgelegt werden?
<https://support.industry.siemens.com/cs/ww/de/view/109476862>
- Wie können Sie beim Starten von TIA Portal ab V13 eine globale Bibliothek automatisch öffnen und z. B. als Unternehmensbibliothek verwenden?
<https://support.industry.siemens.com/cs/ww/de/view/100451450>

9. Anhang

9.1. Service und Support

SiePortal

Die integrierte Plattform für Produktauswahl, Einkauf und Support – und Verbindung von Industry Mall und Online Support. Die neue Startseite ersetzt die bisherigen Startseiten der Industry Mall sowie des Online Support Portals (SIOS) und fasst diese zusammen.

- **Produkte & Services**
Unter Produkte & Services finden Sie alle unsere Angebote, die bisher im Mall Katalog verfügbar waren.
- **Support**
Im Bereich Support finden Sie alle Informationen, die für die Lösung technischer Probleme mit unseren Produkten hilfreich sind.
- **mySieportal**
mySiePortal ist Ihr persönlicher Bereich, der Funktionen, wie z.B. die Warenkorbverwaltung oder die Bestellübersicht anzeigt. Den vollen Funktionsumfang sehen Sie hier erst nach erfolgtem Login.

Das SiePortal rufen Sie über diese Adresse auf:

sieportal.siemens.com

Technical Support

Der Technical Support von Siemens Industry unterstützt Sie schnell und kompetent bei allen technischen Anfragen mit einer Vielzahl maßgeschneiderter Angebote – von der Basisunterstützung bis hin zu individuellen Supportverträgen.

Anfragen an den Technical Support stellen Sie per Web-Formular:

support.industry.siemens.com/cs/my/src

SITRAIN – Digital Industry Academy

Mit unseren weltweit verfügbaren Trainings für unsere Produkte und Lösungen unterstützen wir Sie praxisnah, mit innovativen Lernmethoden und mit einem kundenspezifisch abgestimmten Konzept.

Mehr zu den angebotenen Trainings und Kursen sowie deren Standorte und Termine erfahren Sie unter:

siemens.de/sitrain

Industry Online Support App

Mit der App "Industry Online Support" erhalten Sie auch unterwegs die optimale Unterstützung.

Die App ist für iOS und Android verfügbar:



9.2. Links und Literatur

Tabelle 9-1: Links und Literatur

Nr.	Thema
11)	Siemens Industry Online Support https://support.industry.siemens.com
12)	Link auf die Beitragsseite des Anwendungsbeispiels https://support.industry.siemens.com/cs/ww/de/view/82981502
13)	IO-Link bei Siemens http://www.siemens.de/IO-Link
14)	Webseite der IO-Link Community https://io-link.com/
15)	Herstellerübergreifende Datenbank für Beschreibungsdateien (IODDs) https://ioddfinder.IO-Link.com/
16)	Port Configuration Tool https://support.industry.siemens.com/cs/ww/de/view/32469496
17)	Website der IO-Link Community – Dateien https://io-link.com/de/Download/Download.php
18)	IO-Link Integration – Profile for PROFINET https://io-link.com/de/Download/Download.php

9.3. Änderungsdokumentation

Tabelle 9-2: Änderungsdokumentation

Version	Datum	Änderung
V1.0	04.11.2013	Erste Ausgabe
V2.0	30.03.2015	<ul style="list-style-type: none"> • Komplette Überarbeitung und Aktualisierung auf STEP 7 V13 SP1 Kopierschutz entfernt • IO_LINK_MASTER hinzugefügt
V2.1	21.08.2015	Grenzen bei der Portnummer bei IO_LINK_DEVICE geändert
V3.0	27.10.2016	FB "IO_LINK_MASTER" um die Funktion "Master Backup" für den ET 200eco PN, IO-Link Master (6ES7148-6JD00-0AB0) erweitert
V3.1	16.03.2017	Bibliothek um STEP 7 V14 Basic-Variante erweitert
V5.0	16.11.2017	Neue Master-Funktion für neuen IO-Link Master für STEP 7 V14 SP1 integriert
V5.1	10.01.2018	Ergänzung der Bibliothek für STEP 7 V5.5 SP4 Hotfix 11
V5.2	21.12.2018	Funktionserweiterung für S7-1500, CM 8xIO-Link mit dem Funktionsbaustein IO_LINK_MASTER_8
V6.0	22.09.2020	<ul style="list-style-type: none"> • Bausteine gemäß Programmierstyleguide für S7-1200/1500 und PLCopen komplett überarbeitet • FBs für STEP 7 Basic und STEP 7 Professional zusammengeführt • FBs "IO_LINK_MASTER_4" und "IO_LINK_MASTER_8" zusammengeführt • Geräte-spezifischen Bausteine aus Beitrag 90529409 integriert • Profilbausteine aus Beitrag 109766016 integriert • FB für SIRIUS 3RS2 Geräte hinzugefügt

Version	Datum	Änderung
V7.0	01.04.2023	<ul style="list-style-type: none"> • Automatisierte CAP-Erkennung und optionale Polling-Period implementiert. • Diagnosebaustein "LIOLink_Diagnose" hinzugefügt • Profilbaustein "LIOLink_MultiAdjSwitchingSensor" hinzugefügt • Profilbaustein "LIOLink_MeasuredDataChannel" überarbeitet, Profil 4 ist integriert • Optimierung und Standardisierung Geräte-spezifischer Bausteine. • Geräte-spezifische Baustein "LIOLink_8WD46" hinzugefügt
V7.1	27.01.2025	<ul style="list-style-type: none"> • Profilbaustein "LIOLink_MultiAdjSwitchingSensor" aktualisiert und unterstützt alle SSP-Profile. Zuvor implementierte er nur die Funktionen des Profilbausteins "LIOLink_AdjSwitchingSensor" • Bugfix: "LIOLink_Diagnose"; er unterstützt CAP-Input, und verbesserte Ereignis-Zuordnung • Neuer Basisbaustein "LIOLink_Toolchanger" hinzugefügt • Neuer Basisbaustein als Master Template "LIOLink_MeasuredData" hinzugefügt • Verbesserung: "LIOLink_Device"; robuster • Update aller "LIOLink_Device" abhängigen FBs mit der neuen Version V5.1
V7.2	30.01.2025	<ul style="list-style-type: none"> • SIMATIC RF200 Bausteine „LIOLink_RF200_ReadTag“ und „LIOLink_RF200_WriteTag“ Error Status an das Error bit angepasst.
V7.3	15.04.2025	<ul style="list-style-type: none"> • Einführung v. ET 200clean, CM 8x IO-Link + DIQ 4x 24V DC