

Inhaltsverzeichnis

- Funktionen** 1
- Zweck & Anwendung** 1
 - Zweck 1
 - Anwendung 2
- Installation** 2
 - Anforderungen 2
 - SPS-Einstellungen 2
- Konfiguration** 2
 - Codabix verwenden 2
 - Channel 3
 - Variablen 5
 - Import/Export 7
 - S7-XML-Konfigurationsdatei verwenden 7
 - Struktur 7
 - Verwendung 13
 - Synchronisation 13
 - Beispiel Konfigurationsdatei 13
 - Beispiel Konfigurationsschema-Datei 14
 - Anwendung verwenden (nur unter Windows) 18
 - Überblick 18
 - Verwendung 18
- Fehlerdiagnose** 19
 - Kanal 19
 - Variablen 20
 - Logdatei 20
 - Status Codes 21
- Entities** 22
 - Device 22
 - Channel 22
 - Variable 22
- Ordner & Dateien** 23
 - Ordner 23
 - Dateien 23
- Versionsinformation** 23
 - Dieses Dokument 23
 - Plugin 23
 - Assembly 23

S7 Device Plugin

Das S7 Device Plugin ermöglicht das Lesen und Schreiben von Daten von physischen SIMATIC S7-Geräten über TCP/IP.

Folgende Gerätetypen werden unterstützt:

- S7-1500 (siehe [SPS-Einstellungen](#))
- S7-1200 (siehe [SPS-Einstellungen](#))
- S7-300
- S7-400
- WinAC
- S7-SoftPLC
- LOGO! (siehe [SPS-Einstellungen](#))
- S7-200
- SIMATIC S5 über S5-LAN
- S7-LAN
- VIPA-S7 und jede andere S7-TCP/IP kompatible SPS

Funktionen

- Optimierte Lese- und Schreibzugriffe durch bestmögliche Ausnutzung der Paketgröße.
- Automatisiertes Verbindungshandling, auch auto-reconnect.
- Zugriff auf den SPS-Speicher mit benutzerdefinierten Typen.
- Nutzung der vorhandenen SPS-Projekte, um Kanäle und Variablen einzurichten. Folgende Projektformate werden unterstützt:
 - STEP7 Projekt Dateien (*.s7p)
 - IP-S7 Projekt Dateien (*.ips7)
 - S7 Watch Projekt Dateien (*.wproj)
 - CSV-to-S7 Projekt Dateien (*.ini)

Zweck & Anwendung

Zweck

Die angeschlossenen Geräte können einfach mit CoDaBix® gesteuert werden. Durch die Verknüpfung der SPS-Speicher mit den in CoDaBix® definierten Nodes kann die SPS direkt mit vielen anderen Nodes, Geräten, Diensten usw., die in CoDaBix® gewartet werden, interagieren. Auch andere CoDaBix® Teilnehmer können mit den durch das S7 Device Plugin bereitgestellten SPS-Geräten interagieren.

Anwendung

- Dynamische Generierung von Fertigungsdaten basierend auf verschiedenen Bedingungen und Daten, die von Maschinen, Benutzern, Aufträgen, Zuständen, Diensten usw. generiert werden
- Zusätzliche Sicherheit durch die Überwachung der gesamten Anlage, einschließlich der betriebsinternen Interaktionen
- Zentrale Steuerung und Durchflussüberwachung zur Früherkennung von möglichen Störungen
- Verbesserung des Anlagenleitstands, indem die von den Geräten erzeugten Produktionsdaten überwacht und aufgezeichnet werden

Installation

Dieses Plugin ist Bestandteil des CoDaBix® Setups. Bitte konsultieren Sie [CoDaBix® Setup und erster Start](#) für weitere Informationen darüber, wie dieses Plugin installiert und deinstalliert werden kann.

Anforderungen

- Standardanforderungen von CoDaBix
- aktivierte ausgehende Verbindung via TCP-/IP-Verbindung über Port 102

SPS-Einstellungen

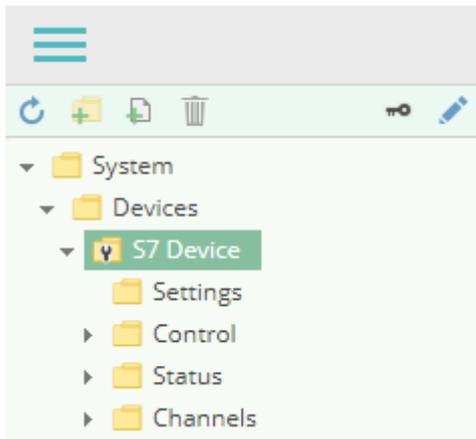
Bitte beachten Sie, dass für den Zugriff auf die S7-1200, S7-1500 sowie LOGO! bei den Datenbausteinattributen der **optimierte Datenbausteinzugriff deaktiviert** sein muss.

Eine Anleitung hierzu finden Sie unter: [Optimierten Datenbausteinzugriff deaktivieren](#).

Konfiguration

Codabix verwenden

Die gesamte S7 Device Plugin-Konfiguration befindet sich unter dem Nodepfad `/System/Devices/S7 Device`.



Channel

Ein S7 Device Channel repräsentiert die Verbindung zu einer S7-SPS.

Settings

Address

Die IP-Adresse oder der Hostname der S7-SPS.

Rack

Die Rack-Nummer der SPS.

Slot

Die Slot-Nummer der SPS.

Device Type

Der Gerätetyp der SIMATIC S7-SPS. Folgende Gerätetypen werden unterstützt:

- LOGO!
- S7-200
- S7-300
- S7-400
- S7-1200
- S7-1500

Channel Type

Der Typ des Kanals, der benutzt wird, um mit der SPS zu kommunizieren.

DateTime Interpretation

Gibt an, wie die Zeitzone von gelesenen und geschriebenen DateTime-Werten interpretiert werden soll (werden die Datums-/Zeitwerte in der SPS als UTC-Zeit oder als lokale Zeit abgelegt).

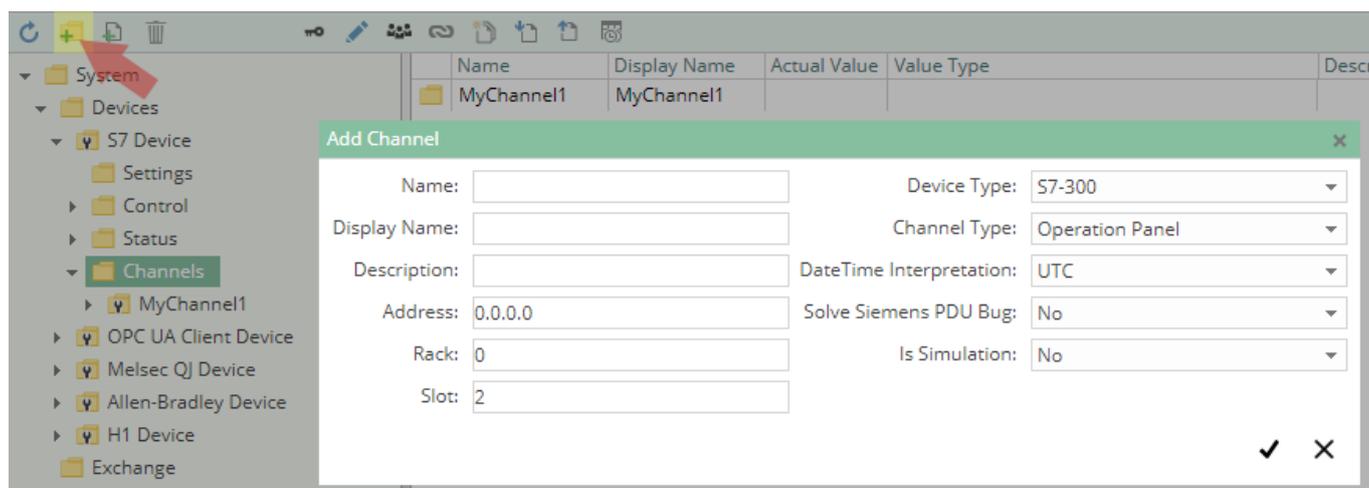
Solve Siemens PDU Bug

Legt fest, ob eine reduzierte PDU-Size verwendet werden soll, um ein Problem in einigen S7-400-SPSen zu umgehen, welche sonst in zufälliger Weise große Datenpakete abschneiden könnten, was sich in Fehlern wie „specified data area doesn't exist“ äußern würde.

Is Simulation

Legt fest, ob dieses Device im Simulationsmodus läuft, d.h. dass keine Verbindung zu einer physischen SPS hergestellt wird.

Hinzufügen eines Channels



Um einen neuen S7-Channel zu erstellen, gehen Sie wie folgt vor:

1. Fügen Sie einen Folder Node unter dem Node S7 Device/Channels hinzu, oder machen Sie einen Rechtsklick auf den S7 Device/Channels-Node und wählen Sie Add Channel aus.

2. Tragen Sie im Add Channel-Dialog die Settings für die S7-Verbindung ein.
3. Nachdem Sie „Save“ geklickt haben, wird die Channel-Node erstellt.
4. Sie können den Kanal starten, indem Sie die Channel-Node auswählen und den Startbutton klicken.

Variablen

Unter dem Variables-Node können Sie Datenpunktnodes erstellen, die aus der S7 gelesen und in diese geschrieben werden. Zusätzlich dazu können Sie Variablen auch z.B. aus einem **STEP7**-Projekt importieren.

Die Value Type-Eigenschaft muss dabei auf den zugehörigen Datentyp festgelegt werden. Aktuell werden folgenden Typen im S7 Device Plugin unterstützt:

S7 Device Type	SPS-Typ	Bevorzugter CoDaBix Value Type	Beschreibung
Bool	BOOL	Boolean or Boolean-Array	Eine Variable vom SPS-Typ BOOL.
Byte	BYTE	Byte or Byte-Array	Eine Variable vom SPS-Typ BYTE.
Char	CHAR	String or String-Array	Eine Variable vom SPS-Typ CHAR.
Int	INT	Int16 or Int16-Array	Eine Variable vom SPS-Typ INT. Repräsentiert einen vorzeichenbehafteten 16-Bit Integer.
Word	WORD	UInt16 or UInt16-Array	Eine Variable vom SPS-Typ WORD. Repräsentiert einen vorzeichenlosen 16-Bit Integer.
DInt	DINT	Int32 or Int32-Array	Eine Variable vom SPS-Typ DINT. Repräsentiert einen vorzeichenbehafteten 32-Bit Integer.
DWord	DWORD	UInt32 or UInt32-Array	Eine Variable vom SPS-Typ DWORD. Repräsentiert einen vorzeichenlosen 32-Bit Integer.
LInt	LINT	Int64 or Int64-Array	Eine Variable vom SPS-Typ LINT. Repräsentiert einen vorzeichenbehafteten 64-Bit Integer.
LWord	LWORD	UInt64 or UInt64-Array	Eine Variable vom SPS-Typ LWORD. Repräsentiert einen vorzeichenlosen 64-Bit Integer.
Real	REAL	Single or Single-Array	Eine Variable vom SPS-Typ REAL. Repräsentiert eine Gleitkommazahl mit einfacher Genauigkeit.
Double	REAL	Double or Double-Array	Eine Variable vom SPS-Typ REAL. Repräsentiert eine Gleitkommazahl mit doppelter Genauigkeit.
LReal	LREAL	Double or Double-Array	Eine Variable vom SPS-Typ LREAL. Repräsentiert eine Gleitkommazahl mit doppelter Genauigkeit.
Date	DATE	DateTime or DateTime-Array	Eine Variable vom SPS-Typ DATE.
Time	TIME	TimeSpan or TimeSpan-Array	Eine Variable vom SPS-Typ TIME.

S7 Device Type	SPS-Typ	Bevorzugter CoDaBix Value Type	Beschreibung
TimeOfDay	TOD/TIME_OF_DAY	TimeSpan or TimeSpan-Array	Eine Variable vom SPS-Typ TOD bzw. TIME_OF_DAY.
S5Time	S5TIME	TimeSpan or TimeSpan-Array	Eine Variable vom SPS-Typ S5TIME.
DateTime	DT/DATE_AND_TIME	DateTime or DateTime-Array	Eine Variable vom SPS-Typ DT bzw. DATE_AND_TIME.
DateTimeLong	DTL	DateTime or DateTime-Array	Eine Variable vom SPS-Typ DTL.
String	STRING	String	Eine Variable vom SPS-Typ STRING.
S5String	BYTE	String	Eine Variable vom SPS-Typ BYTE. Eine feste Anzahl von Bytes wird als String interpretiert.

Path

Über die Path-Eigenschaft des Nodes wird die Adresse, optional eine Typangabe, sowie (bei Arrays oder Strings) die Länge der Daten definiert:

```
<Address>
<Address>, <Length>
<Address>, <Type>
<Address>, <Type>[<Length>]
```

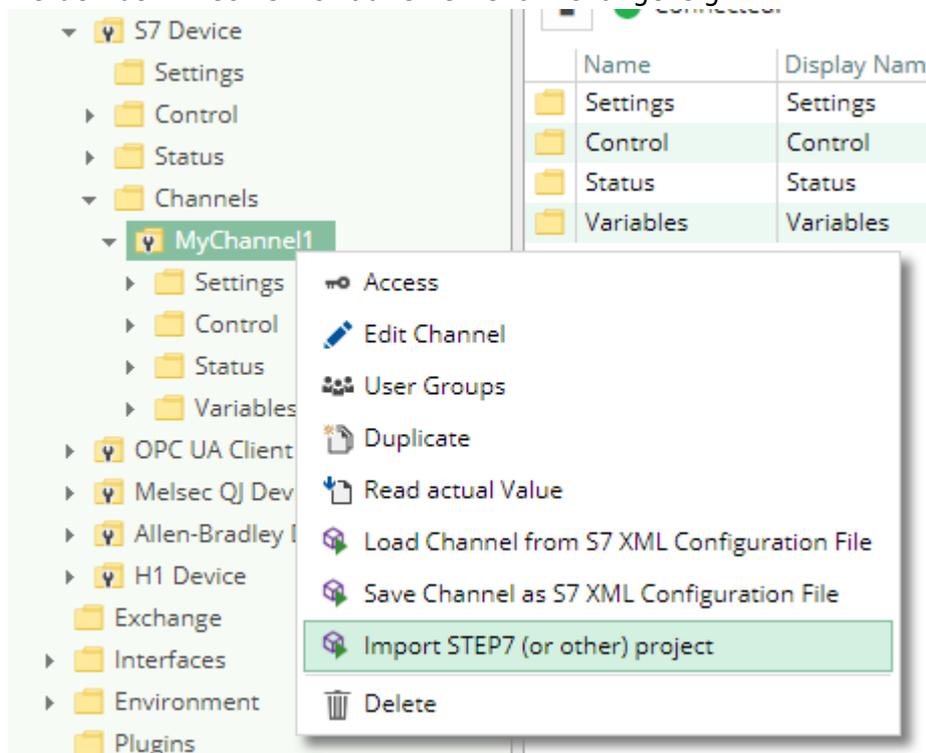
Platzhalter	Beschreibung
<Address>	Die Adresse der Daten. Beispiel: DB10.DBW 16
<Length>	Falls die Variable ein Array oder String ist, geben Sie hier die Länge des Arrays/String an. Diese wird nur beim Lesen verwendet; beim Schreiben wird die Länge aus dem zu schreibenden/String Array ermittelt.
<Type>	Falls angegeben, gibt den S7 Device Type an (siehe obige Tabelle), der den standardmäßig abgeleiteten Typen aus dem CoDaBix Value Type überschreibt. Beachten Sie: Beim Verwenden des CoDaBix Value Type Double wird daraus nicht automatisch ein S7 Device Type abgeleitet; Sie müssen explizit Double angeben.

Beispiele

CoDaBix Value Type	Path	Erklärung
Boolean	DB10.DBX 3.2	Einzelnes Bit (BOOL) bei Adresse DB10.DBX 3.2
Boolean-Array	DB10.DBX 3.4, 18	Bit-(BOOL)-Array von Adresse DB10.DBX 3.4 bis DB10.DBX 5.6 (exklusiv)
Int32	DB10.DBD 12, DInt	Einzelnes DInt (DINT) bei Adresse DB10.DBW 12
Double-Array	DB10.DBD 20, Double[5]	Double-Array (REAL) von Adresse DB10.DBD 20 bis DB10.DBD 40 (exklusiv)
String	DB10.DBB 40, 100	String (STRING) von Adresse DB10.DBB 40 bis DB10.DBB 142 (exklusiv)
String	DB10.DBB 40, S5String[100]	S5String (BYTE) von Adresse DB10.DBB 40 bis DB10.DBB 140 (exklusiv)

Import/Export

Das S7 Device Plugins unterstützt den Import und Export eines Channels als **S7-XML-Konfigurationsdatei** (siehe nächster Abschnitt). Zusätzlich können Sie in einen Channel ein **STEP7-Projekt** (.s7p) importieren, das in eine ZIP-Datei gepackt wurde. Die Import- und Export-Optionen werden beim Rechtsklick auf einen Channel angezeigt:



Um eine Konfigurationsdatei bzw. ein STEP7-Projekt als neuen Channel zu importieren, legen Sie zuerst einen neuen Channel (mit Standardsettings) an, und rechtsklicken Sie dann auf ihn, um den Import-Dialog anzuzeigen.

S7-XML-Konfigurationsdatei verwenden

Struktur

Das S7 Device Plugin definiert die Wurzel seines Elementbaums durch das `PluginSettings` Element wie in [Plugin Konfiguration - Verwenden einer Konfigurationsdatei](#) beschrieben, und setzt dessen XML-Struktur anhand des `Channel`-Elements fort.

Channel Element

Das `Channel` Element dient als Container für die Elemente `Settings` und `Variables`. Ein Channel identifiziert eine SPS-Verbindung, zu welcher die Kanaleinstellungen gehören. Diese Informationen werden dann vom S7 Device Plugin benutzt, um die SPS mit CoDaBix® zu verbinden.

Jedes `Channel` Element stellt folgende Attribute bereit:

	Verpflichtend	Typ	Zweck
Identifizier	nein	GUID	Eindeutiger generischer Identifier der Entity, die mit dem Kanal in Verbindung steht. <i>Dies ist ein allgemeines Entity Attribut, für mehr Information über dessen Nutzung siehe Plugin Konfiguration - Verwenden einer Konfigurationsdatei: Das Entity Identifier Attribut.</i>
ChangeType	nein	ChangeType	Der Zustand der Entity Konfiguration, die dazu verwendet wird, um den Kanal zu repräsentieren. <i>Dies ist ein allgemeines Entity Attribut, für mehr Informationen über dessen Nutzung siehe Plugin Konfiguration - Verwenden einer Konfigurationsdatei: Das Entities ChangeType Attribut.</i>
Name	ja	String	Eindeutiger Name (innerhalb des Channels Elements) des Kanals.

Das Channel Element kann wie folgt aussehen:

```
<Channel>
  <Settings />
  <Variables />
</Channel>
```

Settings Element

Das Settings Element legt die Attribute zur Erstellung des Channels fest. Diese Attribute konfigurieren die Verbindungsparameter des Kanals.

Das Settings Element kann wie folgt aussehen:

```
<Settings Address="192.168.0.80"
  Rack="0"
  Slot="2"
  ChannelType="OperationPanel"
  DeviceType="S7400" />
```

Das Settings Element stellt folgende Attribute bereit:

	Verpflichtend	Typ	Zweck
Address	ja	String	IP Adresse der SPS zur Verbindung.
Rack	nein	Int32	Rack Nummer der SPS.
Slot	nein	Int32	Slot Nummer der SPS.
ChannelType	nein	S7 Device Channel Type	Der Typ des kanals, der benutzt wird, um mit der SPS zu kommunizieren.
DeviceType	nein	S7 Device Type	SPS Gerätetyp.

S7 Device Kanaltyp

Folgende Werte sind als Attribute dieses Typs gültig:

Wert	Beschreibung
„OperationPanel“	um sich via OP Kanal mit dem Gerät zu verbinden.
„ProgrammerDevice“	um sich via PG Kanal mit dem Gerät zu verbinden.
„Other“	um sich über anderen Kanäle mit dem Gerät zu verbinden.

S7 Gerätetyp

Folgende Werte sind als Attribute dieses Typs gültig:

Wert	Beschreibung
„Logo“	SIEMENS LOGO!
„S7200“	SIMATIC S7-200
„S7300“	SIMATIC S7-300
„S7400“	SIMATIC S7-400
„S71200“	SIMATIC S7-1200
„S71500“	SIMATIC S7-1500

Variables Element

Das Variables Element dient als Container für ein oder mehrere Variable Elemente. Dieses Element wartet alle Variablen, die mit dem Channel in Verbindung stehen.

Das Variables Element kann wie folgt aussehen:

```
<Variables>
  <!-- 0-n Variable elements -->
</Variables>
```

Variable Element

Das Variable Element dient als Container für das Element Variables. Eine Variable identifiziert einen adressierbaren Bereich im Speicher der SPS oder eine Reihe von aufeinanderfolgenden Variablen identifiziert mehrere adressierbare Bereiche im Speicher der SPS. Diese Informationen werden dann vom S7 Device Plugin benutzt, um den SPS-Speicher mit CoDaBix® Nodes zu verbinden.

Jedes Variable Element stellt folgende Attribute bereit:

	Verpflichtend	Type	Zweck
Identifizier	nein	GUID	Der generische eindeutige Identifier der Entity, die mit der Variable in Verbindung steht. <i>Dies ist ein allgemeines Entity Attribut, für mehr Informationen zu dessen Nutzung siehe Plugin Konfiguration - Verwenden einer Konfigurationsdatei: <u>Das Entities Identifier Attribut</u>.</i>
ChangeType	nein	ChangeType	Der Zustand der Entity Konfiguration, die dazu verwendet wird, um die Variable zu repräsentieren. <i>Dies ist ein allgemeines Entity Attribut, für mehr Informationen zu dessen Nutzung siehe Plugin Konfiguration - Verwenden einer Konfigurationsdatei: <u>Das Entities ChangeType Attribut</u>.</i>

	Verpflichtend	Type	Zweck
Name	ja	String	Eindeutiger Variablenname innerhalb des Variables Elements.
Description	nein	String	Beschreibung der Benutzung und des Zwecks des adressierten Speicherbereichs.
Address	ja	PLC Address	Operand, der benutzt werden soll, um den Speicher der SPS zu adressieren (hier nicht unterstützt: Type=Object“).
Type	ja	PLC Variable Type	Der Datentyp, der im adressierten Speicher abgelegt ist und wie er interpretiert werden muss.
Length	nein	Int32	Länge einer Array oder String Variable (nur unterstützt im Type=„String“ und numerischen Typen). Ist dieses Attribut definiert, definiert die Variable einen Arraywert (falls unterstützt), andernfalls einen Skalarwert.

SPS Variablentyp

Folgende Werte sind als Attribute dieses Typs gültig:

Wert	Beschreibung
"Bool"	Eine Variable vom SPS-Typ BOOL.
"Byte"	Eine Variable vom SPS-Typ BYTE.
"Char"	Eine Variable vom SPS-Typ CHAR.
"Int"	Eine Variable vom SPS-Typ INT. Repräsentiert einen vorzeichenbehafteten 16 bit Integer.
"Word"	Eine Variable vom SPS-Typ WORD. Repräsentiert einen vorzeichenlosen 16 bit Integer.
"DInt"	Eine Variable vom SPS-Typ DINT. Repräsentiert einen vorzeichenbehafteten 32 bit Integer.
"DWord"	Eine Variable vom SPS-Typ DWORD. Repräsentiert einen vorzeichenlosen 32 bit Integer.
"Real"	Eine Variable vom SPS-Typ REAL. Repräsentiert eine Gleitkommazahl mit einfacher Genauigkeit.
"Double"	Eine Variable vom SPS-Typ REAL. Repräsentiert eine Gleitkommazahl mit doppelter Genauigkeit.
"Date"	Eine Variable vom SPS-Typ DATE.
"Time"	Eine Variable vom SPS-Typ TIME.
"TimeOfDay"	Eine Variable vom SPS-Typ TOD/TIME_OF_DAY.
"S5Time"	Eine Variable vom SPS-Typ S5TIME.
"DateTime"	Eine Variable vom SPS-Typ DT/DATE_AND_TIME.
"DateTimeLong"	Eine Variable vom SPS-Typ DTL.
"String"	Eine Variable vom SPS-Typ STRING.
"S5String"	Eine Variable vom SPS-Typ BYTE. Eine feste Anzahl von Bytes wird als STRING interpretiert.

SPS Variablenadresse

Die folgenden Kombinationen aus Operand und Datentyp sind gültig, um eine gültige SPS Variablenadresse zu konstruieren:

Operands

Operand	Siemens, DE	IEC
Eingang	E	I
Ausgang	A	Q
Merker	M	M
Peripherie	P	P
Zähler	Z	C
Datenbaustein	DB	DB
Timer	T	T

Datentypen

Datentyp	Operand	Bits	Bereich	Beschreibung
BOOL	X	1	0 bis 1	Einzelnes bit, das true (1) or false (0) repräsentiert.
BYTE	B	8	0 bis 255	Ein vorzeichenloser 8-bit Integer.
WORD	W	16	0 bis 65.535	Ein vorzeichenloser 16-bit Integer (Word).
DWORD	D	32	0 bis $2^{32} - 1$	Ein vorzeichenloser 32-bit Integer (Double Word).
CHAR	B	8	A+00 bis A+ff	Ein ASCII-Code als vorzeichenlose 8-bit Zeichenfolge.
INT	W	16	-32.768 bis 32.767	Ein vorzeichenbehafteter 16-bit Integer.
DINT	D	32	-2^{31} bis $2^{31}-1$	Ein vorzeichenbehafteter 32-bit Integer (Double Word).
REAL	D	32	+/-1.5e-45 bis +/-3.4e38	Eine IEEE754 als 32-bit Gleitkommazahl mit einfacher Genauigkeit.
S5TIME	W	16	00.00:00:00.100 bis 00.02:46:30.000	Eine binär codierte dezimale (BCD) Zahl, die eine Zeitspanne in Millisekunden repräsentiert.
TIME	D	32	00.00:00:00.000 bis 24.20:31:23.647	Ein vorzeichenbehafteter 16-bit Integer, der eine Zeitspanne in Millisekunden repräsentiert.
TIME_OF_DAY	D	32	00.00:00:00.000 bis 00.23:59:59.999	Ein vorzeichenloser 16-bit Integer, der eine Zeitspanne in Millisekunden repräsentiert.
DATE	W	16	01.01.1990 bis 31.12.2168	Ein vorzeichenloser 16-bit Integer, der ein Datum in Tagen repräsentiert.
DATE_AND_TIME	D	64	00:00:00.000 01.01.1990 bis 23:59:59.999 31.12.2089	Eine binär codierte dezimale (BCD) Zahl, die Datum und Zeit repräsentiert.

Beispiele

Beispiel	Datentyp	Siemens	IEC
Eingang Byte 1, Bit 0	BOOL	E 1.0	I 1.0
Ausgang Byte 1, Bit 7	BOOL	A 1.7	Q 1.7

Beispiel	Datentyp	Siemens	IEC
Merker Byte 10, Bit 1	BOOL	M 10.1	M 10.1
Datenbaustein 1, Byte 1, Bit 0	BOOL	DB1.DBX 1.0	DB1.DBX 1.0
Eingang Byte 1	BYTE	EB 1	IB 1
Ausgang Byte 10	BYTE	AB 10	QB 10
Merker Byte 100	BYTE	MB 100	MB 100
Peripherie Eingang Byte 0	BYTE	PEB 0	PIB 0
Peripherie Ausgang Byte 1	BYTE	PAB 1	PQB 1
Datenbaustein 1, Byte 1	BYTE	DB1.DBB 1	DB1.DBB 1

Das Variable Element kann wie folgt aussehen:

```
<Variable Name="Active Rotations"
  Type="DInt"
  Address="DB101.DBD 0" />
```

Das Variable Element mit dem Type Attribut und Attributwert „Object“ kann wie folgt aussehen:

```
<Variable Name="Mill Job No. 1" Description="Identifies the mill job
no. 1" Type="Object">
  <Variables>
    <Variable Name="Rotations" Type="DInt" Address="DB200.DBD 0" />
    <Variable Name="Use Colling" Type="Bool" Address="DB200.DBX 4.0"
  />
  <Variable Name="Use Fan" Type="Bool" Address="DB200.DBX 4.1" />
  <Variable Name="Point 1" Type="Object">
    <Variables>
      <Variable Name="X" Description="The x portion of the drill."
Type="Int" Address="DB200.DBW 5" />
      <Variable Name="Y" Description="The y portion of the drill."
Type="Int" Address="DB200.DBW 7" />
    </Variables>
  </Variable>
  <Variable Name="Point 2" Type="Object">
    <Variables>
      <Variable Name="X" Description="The x portion of the drill."
Type="Int" Address="DB200.DBW 9" />
      <Variable Name="Y" Description="The y portion of the drill."
Type="Int" Address="DB200.DBW 11" />
    </Variables>
  </Variable>
  <Variable Name="Point 3" Type="Object">
    <Variables>
      <Variable Name="X" Description="The x portion of the drill."
Type="Int" Address="DB200.DBW 13" />
      <Variable Name="Y" Description="The y portion of the drill."
Type="Int" Address="DB200.DBW 15" />
    </Variables>
  </Variable>
</Variable>
```

```
</Variables>  
</Variable>
```

Verwendung

Es wird empfohlen, einen professionellen XML-Editor zum manuellen Editieren der Konfigurationsdatei zu verwenden. Um den Vorteil der XML-Schemadefinition (am Ende der Dokumentation erwähnt) zu nutzen, müssen Sie wie folgt auf das `xsi:noNamespaceSchemaLocation` Attribut im Wurzelement des Dokuments `PluginSettings` verweisen (die XSD-Datei muss neben der XML-Datei abgelegt werden):

```
<?xml version="1.0" encoding="utf-8" ?>  
<PluginSettings xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
  xsi:noNamespaceSchemaLocation="CoDaBix.S7DevicePlugin.Settings.xsd">  
  <!-- child elements -->  
</PluginSettings>
```

Unabhängig davon, ob die Konfigurationsdatei manuell oder automatisiert bearbeitet / erstellt wird, muss der oben dokumentierte Elementbaum vollständig sein, um eine gültige, wohlgeformte und benutzbare Konfigurationsdatei zu erzeugen.

Synchronisation

Eine S7-XML-Konfigurationsdatei wird einem S7 Device Channel in CoDaBix zur **automatischen Synchronisierung** zugeordnet, wenn die folgende Bedingung erfüllt ist:

- Eine Datei mit dem Namen `CoDaBix.S7DevicePlugin.<ChannelName>.Settings.xml` existiert im `ConfigFolder (<CodabixProjectDir>/plugins/S7DevicePlugin)` wenn das S7-Plugin gestartet wird (d.h. wenn CoDaBix gestartet wird) oder wenn in CoDaBix ein neuer Channel erzeugt wird.

In diesem Fall wird die Konfigurationsdatei, wenn sie sich ändert, automatisch nach CoDaBix synchronisiert. Umgekehrt werden bei Änderungen in CoDaBix (oder in den Channel-Settings) diese in die Konfigurationsdatei synchronisiert.

Unabhängig von der automatische Synchronisation können Sie auch eine manuelle Synchronisation durch den Rechtsklick auf einen S7 Device Channel in CoDaBix und das Wählen des entsprechenden Import/Export-Menüeintrages durchführen.

Beispiel Konfigurationsdatei

[CoDaBix.S7DevicePlugin.Settings.xml](#)

```
<?xml version="1.0" encoding="utf-8"?>  
<PluginSettings>  
  <S7Device>  
    <Channels>  
      <Channel Name="Line Controller">  
        <Settings Address="192.168.0.80"  
ChannelType="OperationPanel" DeviceType="S7400" />
```

```

    <Variables>
      <Variable Name="Active Job Name" Type="String"
Address="DB100.DBB 0" Length="64" />
      <Variable Name="Active Job Number" Type="String"
Address="DB100.DBB 80" Length="8" />
      <Variable Name="Active Rotations" Type="DInt"
Address="DB101.DBD 0" />
      <Variable Name="Mill Job No. 1" Description="Identifies
the mill job no. 1" Type="Object">
        <Variables>
          <Variable Name="Rotations" Type="DInt"
Address="DB200.DBD 0" />
          <Variable Name="Use Colling" Type="Bool"
Address="DB200.DBX 4.0" />
          <Variable Name="Use Fan" Type="Bool"
Address="DB200.DBX 4.1" />
          <Variable Name="Point 1" Type="Object">
            <Variables>
              <Variable Name="X" Description="The x portion of
the drill." Type="Int" Address="DB200.DBW 5" />
              <Variable Name="Y" Description="The y portion of
the drill." Type="Int" Address="DB200.DBW 7" />
            </Variables>
          </Variable>
          <Variable Name="Point 2" Type="Object">
            <Variables>
              <Variable Name="X" Description="The x portion of
the drill." Type="Int" Address="DB200.DBW 9" />
              <Variable Name="Y" Description="The y portion of
the drill." Type="Int" Address="DB200.DBW 11" />
            </Variables>
          </Variable>
          <Variable Name="Point 3" Type="Object">
            <Variables>
              <Variable Name="X" Description="The x portion of
the drill." Type="Int" Address="DB200.DBW 13" />
              <Variable Name="Y" Description="The y portion of
the drill." Type="Int" Address="DB200.DBW 15" />
            </Variables>
          </Variable>
        </Variables>
      </Variable>
    </Variables>
  </Channel>
</Channels>
</S7Device>
</PluginSettings>

```

Beispiel Konfigurationsschema-Datei

CoDaBix.S7DevicePlugin.Settings.xsd

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema id="CoDaBix.S7DevicePlugin.Settings"
  elementFormDefault="qualified"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name="Guid">
    <xs:restriction base="xs:string">
      <xs:pattern value="([0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12})|(\{[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}\})"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="PlcAddress">
    <xs:restriction base="xs:string">
      <xs:pattern value="^[ \t]*((DB[ \t]*([\d+])[ \t]*\.[ \t]**(DB))|(((Z|C)|DB|M|(E|I)|L|(A|Q)|(PE|PI)|(PA|PQ)|T)))[ \t]*((X|B|W|D))?[ \t]*([\d+](\.[\d+))?\b" />
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="SettingsChangeTypeEnum">
    <xs:restriction base="xs:string">
      <xs:enumeration value="None" />
      <xs:enumeration value="Created" />
      <xs:enumeration value="Updated" />
      <xs:enumeration value="Deleted" />
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="VariableEnumType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="Bool" />
      <xs:enumeration value="Byte" />
      <xs:enumeration value="Char" />
      <xs:enumeration value="Int" />
      <xs:enumeration value="Word" />
      <xs:enumeration value="DInt" />
      <xs:enumeration value="DWord" />
      <xs:enumeration value="Real" />
      <xs:enumeration value="Double" />
      <xs:enumeration value="Date" />
      <xs:enumeration value="Time" />
      <xs:enumeration value="TimeOfDay" />
      <xs:enumeration value="S5Time" />
      <xs:enumeration value="DateTime" />
      <xs:enumeration value="String" />
    </xs:restriction>
  </xs:simpleType>
```

```
<xs:complexType name="VariableType">
  <xs:sequence>
    <xs:element name="Variables" type="VariablesType"
minOccurs="0" maxOccurs="1" />
  </xs:sequence>

  <xs:attribute name="Identifier" type="Guid" use="required" />
  <xs:attribute name="Name" type="xs:string" use="required" />
  <xs:attribute name="Description" type="xs:string"
use="optional" />
  <xs:attribute name="Address" type="PlcAddress" use="optional"
/>
  <xs:attribute name="Type" type="VariableEnumType"
use="required" />
  <xs:attribute name="Length" type="xs:integer" use="optional"
default="-1" />
  <xs:attribute name="ChangeType"
type="SettingsChangeTypeEnumType" use="optional" />
</xs:complexType>

<xs:complexType name="VariablesType">
  <xs:sequence>
    <xs:element name="Variable" type="VariableType"
minOccurs="0" maxOccurs="unbounded" />
  </xs:sequence>
</xs:complexType>

<xs:simpleType name="ChannelDeviceEnumType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Logo" />
    <xs:enumeration value="S7200" />
    <xs:enumeration value="S7300" />
    <xs:enumeration value="S7400" />
    <xs:enumeration value="S71200" />
    <xs:enumeration value="S71500" />
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="ChannelEnumType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="OperationPanel" />
    <xs:enumeration value="ProgrammerDevice" />
    <xs:enumeration value="Other" />
  </xs:restriction>
</xs:simpleType>

<xs:complexType name="ChannelSettingsType">
  <xs:attribute name="Address" type="xs:string" use="required"
/>
  <xs:attribute name="Rack" type="xs:integer" use="optional"
default="0" />
</xs:complexType>
```

```
<xs:attribute name="Slot" type="xs:integer" use="optional"
default="2" />
<xs:attribute name="ChannelType" type="ChannelEnumType"
use="optional" default="OperationPanel" />
<xs:attribute name="DeviceType" type="ChannelDeviceEnumType"
use="optional" default="S7300" />
<xs:attribute name="ChangeType"
type="SettingsChangeTypeEnumType" use="optional" />
</xs:complexType>

<xs:complexType name="ChannelType">
  <xs:sequence>
    <xs:element name="Settings" type="ChannelSettingsType"
minOccurs="0" maxOccurs="1" />
    <xs:element name="Variables" type="VariablesType"
minOccurs="0" maxOccurs="1" />
  </xs:sequence>

  <xs:attribute name="Identifier" type="Guid" use="required" />
  <xs:attribute name="Name" type="xs:string" use="required" />
  <xs:attribute name="ChangeType"
type="SettingsChangeTypeEnumType" use="optional" />
</xs:complexType>

<xs:complexType name="ChannelsType">
  <xs:sequence>
    <xs:element name="Channel" type="ChannelType" minOccurs="0"
maxOccurs="unbounded" />
  </xs:sequence>
</xs:complexType>

<xs:complexType name="DeviceType">
  <xs:sequence>
    <xs:element name="Channels" type="ChannelsType"
minOccurs="0" maxOccurs="1" />
  </xs:sequence>
</xs:complexType>

<xs:complexType name="PluginSettingsType">
  <xs:sequence>
    <xs:element name="S7Device" type="DeviceType" minOccurs="0"
maxOccurs="1" />
  </xs:sequence>
</xs:complexType>

  <xs:element name="PluginSettings" type="PluginSettingsType" />
</xs:schema>
```

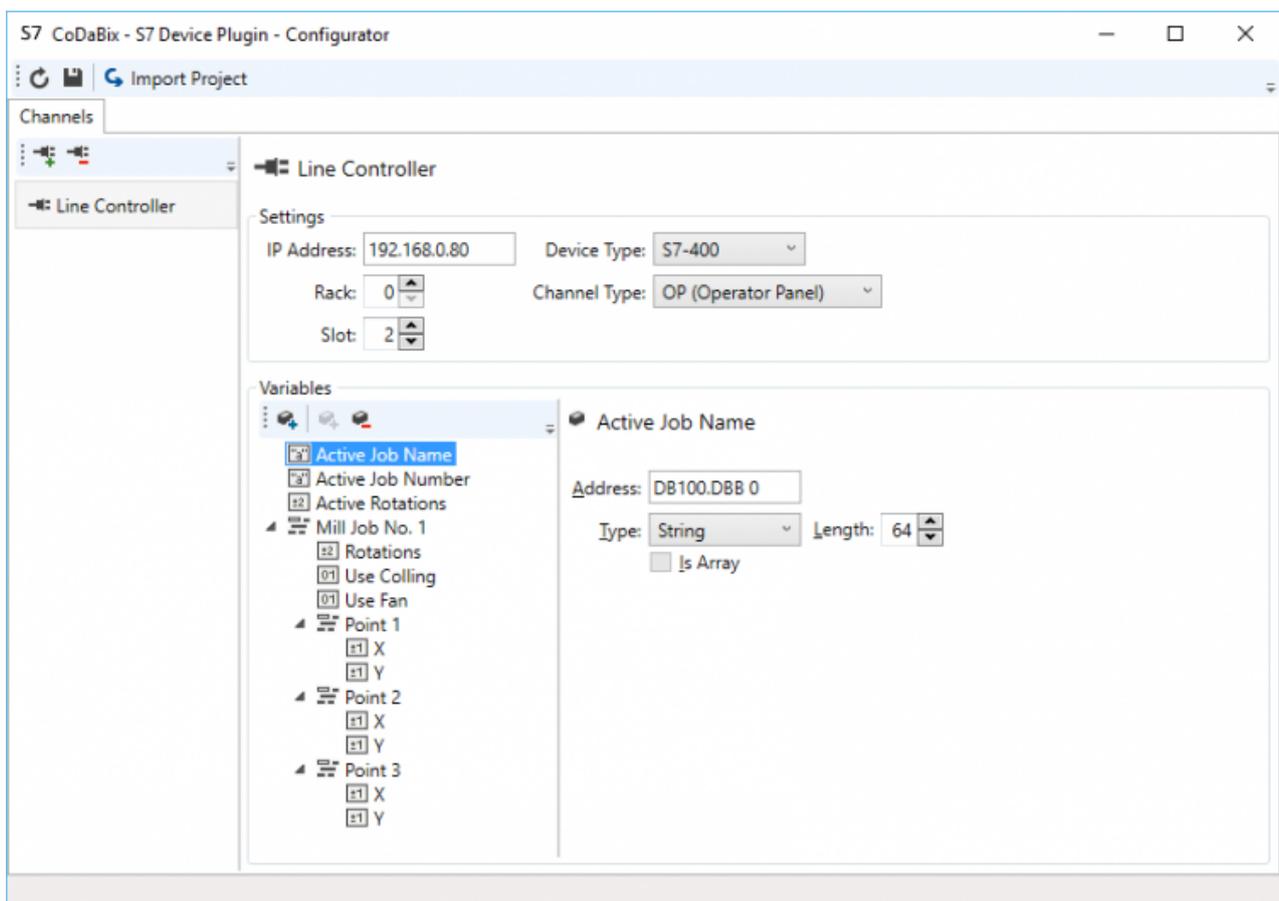
Anwendung verwenden (nur unter Windows)

Beachten Sie: Diese Konfigurationsanwendung ermöglicht es, S7-XML-Konfigurationsdateien (.xml) zu laden und zu speichern, die in CoDaBix bei einem S7 Device Channel über das Kontextmenü importiert und exportiert werden können, oder die einem CoDaBix S7 Device Channel zur automatischen Synchronisierung zugeordnet sind.

(siehe Abschnitt [S7-XML-Konfigurationsdatei verwenden](#))

Die Konfigurationsanwendung befindet sich im Ordner <CoDaBixInstallDir> und wird durch einen Doppelklick auf die Datei mit dem Namen CoDaBix.S7DevicePlugin.Configurator.exe gestartet.

Überblick



Verwendung

- Einen neuen Channel in der linken Liste hinzufügen, indem Sie auf den oberen Plus-Button klicken.
 - Namen vergeben, indem Sie im rechten Ausschnitt den Kanalnamen klicken.
 - Die Channel Settings einstellen wie die IP Adresse (mindestens erforderlich).
 - Optional den passenden Device Type und den zu benutzenden Channel Type

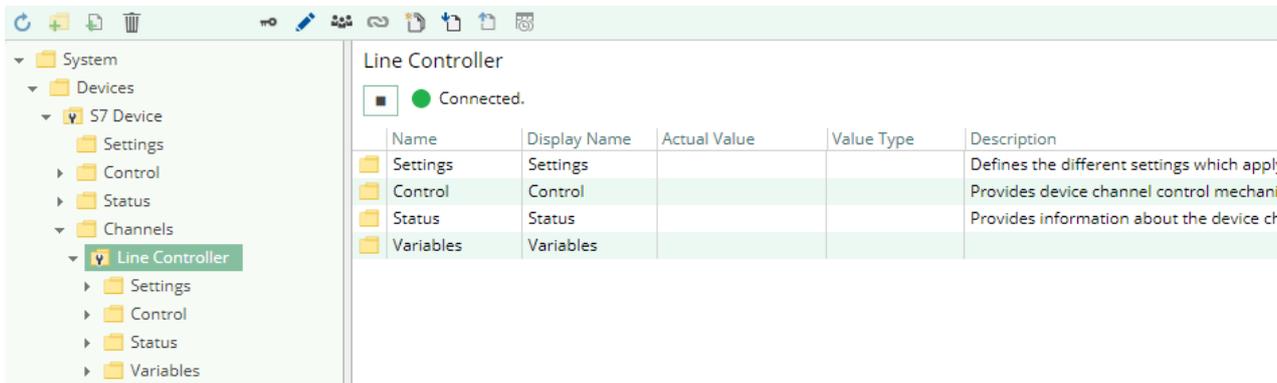
- auswählen.
- Optional den Rack und Slot einstellen (falls nicht gleich benutzerdefiniert).
- In der zweiten Liste eine neue Variable zum Channel hinzufügen, indem Sie auf den oberen ersten Plus-Button klicken.
 - Mit Klick auf den Namen im rechten Ausschnitt ändern Sie den Variablennamen.
 - Optional in den freien Bereich unterhalb des Variablennamens eine Beschreibung der Variable hinzufügen.
 - Den SPS-Adressoperanden festlegen, der benutzt werden soll, um auf den SPS-Speicher für die Variable zuzugreifen.
 - SPS-Variablentyp in der Combo auswählen und im Falle einer String oder Array (Option Is Array auswählen) Variable die passende Länge eingeben.
 - Fügen Sie optional zusätzliche Variablen hinzu, falls die vorher hinzugefügte Variable vom Typ „Object“ ist. Dies machen Sie, indem Sie die Object Variable im linken Baum auswählen und dann den zweiten Plus-Button klicken.
- Zum Löschen einer Variablen wählen Sie diese im linken Baum aus und klicken Sie den Minus-Button.
- Zum Löschen eines Channels wählen Sie diesen in der linken Liste aus und klicken Sie den Minus-Button.
- Bei Klick auf den Import Project-Button ist es möglich, folgende Projektformate zu importieren:
 - STEP7 Projektdateien (*.s7p)
 - IP-S7 Projektdateien (*.ips7)
 - S7 Watch Projektdateien (*.wproj)
 - CSV-to-S7 Projektdateien (*.ini)

Fehlerdiagnose

Das S7 Device Plugin stellt je nach zu inspizierender Schicht verschiedene Statusinformationen zur Verfügung. Generell wird die Channel-basierte Diagnoseinformationen vom Verbindungsstatus vom kanal zur S7 SPS. Die Variable-basierte Diagnoseinformation wird während des Lese- und Schreibzugriffes verschiedener Variablen generiert.

Kanal

Um den Status des S7-Kanals zu überwachen und zu diagnostizieren, werfen Sie einen Blick auf das folgende Bild:



Das obige Bild zeigt das Bedienfeld des S7-Kanals, das alle statusrelevanten Informationen anzeigt. Das Bedienfeld aktualisiert automatisch seine Statusinformation, wenn ein neuer Status verfügbar ist.

Statuskreis

Farbe	Bedeutung
●	Der Kanal ist gestoppt. Klicken Sie den ►-Button, um ihn zu starten.
●	Der Kanal startet oder stoppt gerade oder wartet auf den Verbindungsaufbau.
●	Der Kanal läuft und es wurde erfolgreich eine Verbindung hergestellt. Sie können ihn durch Klick auf den ■-Button stoppen.
●	Der Kanal läuft, aber die Verbindung ist momentan fehlerhaft. Bitte überprüfen Sie den Statustext für weitere Informationen.

Variablen

Zur Überwachung und Diagnose der verschiedenen S7 Variablen werfen Sie einen Blick auf die Status Code-Eigenschaft der Variable, die in CoDaBix angezeigt wird. Falls die Status Code-Spalte den Wert Bad anzeigt, ist in den meisten Fällen der adressierte Datenbereich nicht erreichbar.



Logdatei

Alle Device Channel-bezogenen Statusinformationen werden auch in der kanalspezifischen Logdatei im [LoggingFolder] protokolliert. Jede Logdatei wird im Namensschema S7 Device.<Channel>.log benannt. Der Inhalt einer solchen Logdatei kann wie folgt aussehen:

```

...
[15:31:46 05.09.2016] - Error (Severity=High): Code=[-6], Text=[The
specified CPU could not be found.]
...
    
```

Beim Benutzen des Beispielkanals wäre der Name der Logdatei: S7 Device.Line Controller.log.

Status Codes

Die folgende Tabelle zeigt die verschiedenen Statusinformationen, die möglich sind:

Code	Kategorie	Schweregrad	Text	Bedeutung
-88	Error	High	The operation has been canceled.	Der Vorgang wurde abgebrochen.
-34	Error	High	The requested PLC block could not be found.	Der angefragte SPS-Bereich wurde nicht gefunden.
-33	Error	High	The PLC does not support bit based operations.	Die SPS unterstützt keine Bit-basierten Operationen.
-32	Error	High	The PLC supplied truncated data.	Die SPS hat unvollständige Daten gesendet.
-31	Error	High	The requested PLC or PC type of data can not be transformed from one into the other.	Der angeforderten SPS- oder PC-Datentyp kann nicht vom einen zum anderen umgewandelt werden.
-30	Error	High	The requested PLC or PC type of data is not available.	Der angeforderte SPS- oder PC-Datentyp ist nicht verfügbar.
-21	Error	High	A connection to the device has already been established.	Es wurde bereits eine Verbindung zum Gerät aufgebaut.
-20	Error	High	The size of the supplied buffer is lower than the amount of data available.	Die Größe des übergebenen Puffers ist geringer als die Menge der verfügbaren Daten.
-11	Error	High	The type or format of data supplied is not supported.	Datentyp oder -format wird nicht unterstützt.
-10	Error	High	The supplied access mode is not supported or unknown.	Der übergebene Zugriffsmodus wird nicht unterstützt oder ist unbekannt.
-9	Error	High	The operation failed upon a value range error.	Vorgang aufgrund eines Wertebereichfehlers fehlgeschlagen.
-8	Error	High	Failed to allocate required memory.	Angeforderter Speicher konnte nicht zugeteilt werden.
-7	Error	High	The operation failed upon a socket error.	Vorgang aufgrund eines Socket Fehlers fehlgeschlagen.
-6	Error	High	The specified CPU could not be found.	Die angegebene CPU konnte nicht gefunden werden.
-5	Error	High	A general error occurred.	Ein allgemeiner Fehler ist aufgetreten.
-2	Error	High	The necessary memory could not be allocated (out of memory).	Die erforderliche Speicher konnte nicht allokiert werden (nicht genügend Arbeitsspeicher).

Code	Kategorie	Schweregrad	Text	Bedeutung
-1	Error	High	The operation has timed out.	Der Vorgang hat zu lange gedauert.
0	Information	Moderate	The operation completed without any kind of error.	Der Vorgang wurde ohne Fehler beendet.
1	Information	Moderate	The operation completed successfully.	Der Vorgang wurde erfolgreich abgeschlossen.
2	Information	Moderate	The addressed data area does not exist.	Der adressierte Datenbereich existiert nicht.

Entities

Wie jedes Device Plugin erweitert das S7 Device Plugin das CoDaBix [Device Modell](#).

Device

Der Device Typ S7Device des Plugins definiert auch den S7DeviceChannel und erweitert somit die grundlegenden CodabixDevice und CodabixDeviceChannel Entities. Während das S7Device nur eine Konkretisierung des CodabixDevice darstellt, erweitert der S7DeviceChannel den CodabixDeviceChannel mit dem S7 Variable Entities.

Channel

Jeder Kanal wird von einem Channel Worker behandelt, der eine physische Verbindung zur S7 SPS herstellt. Zur Fehlerdiagnosezwecken liest der Worker automatisch alle 10 Sekunden die SPS-Adresse „MB 0“, um den Channel Statuscode und seine Beschreibung zu aktualisieren und Verbindungsfehler zu erkennen.

Der Worker liest standardmäßig keine Werte. Wenn ein Anwender oder Plugin in CoDaBix einen synchronen Lesevorgang der Channel Variablen anfordert (z.B. mit der „Read actual value“-Funktion in der CoDaBix Webkonfiguration), liest der Channel Worker diese aus dem zugrundeliegenden S7 Device und schreibt diese in die entsprechenden CoDaBix® Nodes.

Ähnlich schreibt der Channel Worker auch die Werte in die zugrundeliegende S7, wenn ein Client oder Plugin Werte in die Channel Variablen schreibt.

Damit eine S7 Variable regelmäßig gelesen wird, können Sie in der Webkonfiguration bei dem Node „History Options“ auf Yes stellen (was eine interne Subscription erstellt), oder Sie können zum Beispiel einen OPC UA Client verwenden, der mit dem OPC UA Serverplugin verbunden ist und damit eine Subscription für die S7 Variablenodes erstellen. In diesen Fällen liest der Channel Worker in regelmäßigen Intervallen die Variablen aus der S7 und schreibt den neuen Wert nach einer Wertänderung automatisch in die entsprechende CoDaBix® Node.

Variable

Jede S7 Variable kann anhand des gleichen SPS-Adressoperanden auf denselben SPS-Speicher zugreifen. Seine Interpretation hängt jedoch von der SPS-Datentyp Auswahl ab. Unterstützte Variablenformate sind Skalar, Array und Objektvariablen. Variablen vom Typ Object können weitere Variablen besitzen und sind als UserDefinedTypes (UDT in STEP7) definiert.

Ordner & Dateien

Ordner

Name	Pfad	Zweck / Verwendung
AssemblyFolder	<CodabixInstallDir>/plugins/S7DevicePlugin/	Beinhaltet die Plugin Assembly Datei.
ConfigFolder	<CodabixDataDir>/plugins/S7DevicePlugin/	Beinhaltet die Plugin Konfigurationsdatei.
LoggingFolder	<CodabixDataDir>/log/	Beinhaltet die Plugin Log Dateien.

Dateien

Typ	Pfad	Zweck / Verwendung
Assembly	[AssemblyFolder]/CoDaBix.S7DevicePlugin.dll	Die Plugin Assembly Datei.
Config App	<CodabixInstallDir>/CoDaBix.S7DevicePlugin.Configurator.exe	Die Plugin Konfigurationsdatei.
Config File	[ConfigFolder]/CoDaBix.S7DevicePlugin.<ChannelName>.Settings.xml	Die optionale Channel-Konfigurationsdatei.
Logging	[LoggingFolder]/S7 Device.<ChannelName>.log	Die Log Datei.

Versionsinformation

Dieses Dokument

Datum	2020-21-0
Version	1.3

Plugin

Name	S7 Device Plugin
Node	/System/Devices/S7 Device
Version	1.2.0

Assembly

Name	CoDaBix.S7DevicePlugin.dll
Datum	2020-12-01
Version	1.2.0.0

From:

<https://www.codabix.de/> - **CoDaBix®**

Permanent link:

<https://www.codabix.de/de/plugins/device/s7deviceplugin>

Last update: **2022/07/28 17:07**