

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. Sie sind selbst für den sachgemäßen und sicheren Betrieb der Produkte innerhalb der geltenden Vorschriften verantwortlich und müssen dazu die Funktion des jeweiligen Anwendungsbeispiels überprüfen und auf Ihre Anlage individuell anpassen.

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. Die Anwendungsbeispiele unterliegen nicht zwingend den üblichen Tests und Qualitätsprüfungen eines kostenpflichtigen Produkts, können Funktions- und Leistungsmängel enthalten und mit Fehlern behaftet sein. Sie sind verpflichtet, die Nutzung so zu gestalten, dass eventuelle Fehlfunktionen nicht zu Sachschäden oder der Verletzung von Personen führen.

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. 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://support.industry.siemens.com).

Securityhinweise

Siemens bietet Produkte und Lösungen mit Industrial Security-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 Security-Konzept zu implementieren (und kontinuierlich aufrechtzuerhalten), das dem aktuellen Stand der Technik entspricht. Die Produkte und Lösungen von Siemens formen nur einen Bestandteil eines solchen Konzepts.

Der Kunde ist dafür verantwortlich, unbefugten Zugriff auf seine Anlagen, Systeme, Maschinen und Netzwerke zu verhindern. Systeme, Maschinen und Komponenten sollten nur mit dem Unternehmensnetzwerk oder dem Internet verbunden werden, wenn und soweit dies notwendig ist und entsprechende Schutzmaßnahmen (z.B. Nutzung von Firewalls und Netzwerksegmentierung) ergriffen wurden.

Zusätzlich sollten die Empfehlungen von Siemens zu entsprechenden Schutzmaßnahmen beachtet werden. Weiterführende Informationen über Industrial Security finden Sie unter: https://www.siemens.com/industrialsecurity.

Die Produkte und Lösungen von Siemens werden ständig weiterentwickelt, um sie noch sicherer zu machen. Siemens empfiehlt ausdrücklich, Aktualisierungen durchzuführen, sobald die entsprechenden Updates 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 Security RSS Feed unter: https://www.siemens.com/industrialsecurity.

Inhaltsverzeichnis

Rec	htliche Hi	nweise	2
1	Aufgab	e	4
	1.1 1.2	Übersicht Anforderungen	
2	Lösung	J	6
	2.1 2.2 2.3	Beschreibung der Kernfunktionalität Verwendete Hard- und Software-Komponenten Alternativlösungen	9
3	Funktio	onsmechanismen dieser Applikation	13
4	3.1 3.1.1 3.1.2 3.1.3 3.1.4 3.1.5 3.2 3.2.1 3.2.2 Projekt 4.1 4.2	Programmierung des Clients unter Verwendung des .NET OPC Data Controls Einbinden der Assemblies Einstellen des Compilers Erstellen der Oberfläche Verbinden der OPC Datenpunkte (Variablen) mit dem Control Schreiben von OPC-Items Erläuterungen zum Simulationsprogramm in der Steuerung Ablauf der Simulation Funktion der Bausteine ierung eines OPC-Servers Konfiguration der OPC Server Station Endkontrolle der Einstellungen	13 15 16 17 19 21 22 23
5	Installa	tion und Inbetriebnahme	29
	5.1 5.2 5.3 5.4 5.5	Installation der Hard- und SoftwareLaden der der PC Station über STEP 7 V1xImportieren der XDB-Datei in den Komponenten KonfiguratorInstallation des OPC-Clients auf dem PC/PGLaden der Simulation auf die S7-Stationen	31 36 39
6	Bedien	ung der Applikation	41
7	Glossa	r	42
8	Literatu	ırhinweise	43
	8.1	Internet-Link-Angaben	43
9	Historie	9	44

1.1 Übersicht

1 Aufgabe

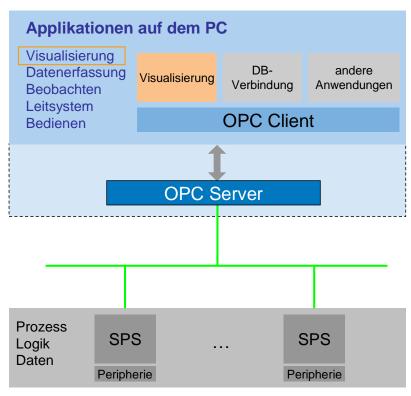
1.1 Übersicht

Einführung/Einleitung

Dieses Applikationsbeispiel zeigt exemplarisch die Kopplung eines Produktionsprozesses an einen Windows basierten PC mit einem sehr einfachen und schnell zu realisierenden Datenaustausch über OPC. Mit diesem Wirkungsprinzip können beispielsweise eigene, spezialisierte Bedienoberflächen und Prozessvisualisierung oder Datenerfassung realisiert werden.

Überblick über die Automatisierungsaufgabe

Folgendes Bild gibt einen Überblick über die Automatisierungsaufgabe. Abbildung 1-1



Beschreibung der Automatisierungsaufgabe

Ein Prozess wird mit zwei verschiedenen SPSen simuliert. Die Daten aus diesem Prozess sollen angezeigt und modifiziert werden können. Diese Applikation zeigt, wie - unter zu Hilfenahme des SIMATIC NET OPC Data Control für .NET – Prozessdaten auf einem PC angezeigt werden können, unabhängig davon aus welcher Prozesssteuerung sie stammen.

1.2 Anforderungen

1.2 Anforderungen

Anforderungen an die Bedien- und Beobachtungssoftware zur Visualisierung

Die Software soll eine schnelle und einfache Oberflächenerstellung ermöglichen. Dazu muss sie folgenden Anforderungen entsprechen:

• Verwendung des vereinfachten .NET OPC DA Connector zur Verschaltung mit .NET Oberflächencontrols.

Anforderung an die Datenschnittstelle zwischen Visualisierung und Steuerung

Der Austausch der Prozessdaten erfolgt über die standardisierte OPC-DA Schnittstelle:

- Anbindung an die Prozessdaten über Industrial Ethernet sowie den SIMATIC NET OPC-Server V8.2 (oder höher).
- Nutzung der OPC-DataAccess Schnittstelle
- Symbolische Adressierung der Prozessdaten
- · Lesen und Schreiben von Prozessdaten
- Beobachten von Prozessdaten

Anforderungen an die einzusetzende Entwicklungsumgebung

Es soll die aktuelle Windows-Entwicklungsumgebung eingesetzt werden:

- Einsatz des Microsoft Visual Studio® .NET 2008 SP1 (oder höher)
- Einsatz der .NET-Programmiersprache Visual C#

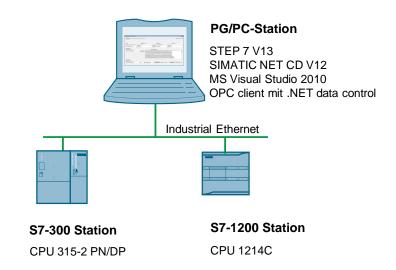
1.2 Anforderungen

2 Lösung

Schema

Die folgende Abbildung zeigt schematisch die wichtigsten Komponenten der Lösung:

Abbildung 2-1



S7-Station

Auf der Steuerungsseite befindet sich eine SIMATIC S7-300 mit einer CPU 315-2 PN/DP sowie optional dem CP 343-1. Weiterhin eine S7-1214C mit Ethernetanschluss.

PC Station

Eine PC-Station wird über einen Switch an eine S7-300 Steuerung und eine S7-1200 angeschlossen. Dabei wird eine handelsübliche Ethernet-Netzwerkkarte benutzt. Auf der PC-Station läuft sowohl der SIMATIC NET OPC Server, als auch der OPC-Client. Ein sehr einfach gestalteter Client zeigt Ihnen alle Basisfunktionen für den Einstieg.

Vorteile

Die hier gezeigte Lösung verwendet den SIMATIC NET OPC DA Server und kann unabhängig vom Typ der eingesetzten SIMATIC Steuerung verwendet werden. Die gezeigte Ankopplung der SIMATIC and den OPC-Server über S7 Protokoll und Ethernet ist für den Client transparent. Die hier gezeigte Lösung kann ebenso für alle anderen vom SIMATIC NET OPC Server unterstützten Protokolle und Busvarianten genutzt werden.

2.1 Beschreibung der Kernfunktionalität

Abgrenzung

Diese Applikation enthält keine vollständige Beschreibung

- des .NET-Frameworks,
- von C# bzw. VB.NET,
- der OPC-Spezifikation sowie
- tiefgehender COM-Mechanismen

Vorausgesetzte Kenntnisse

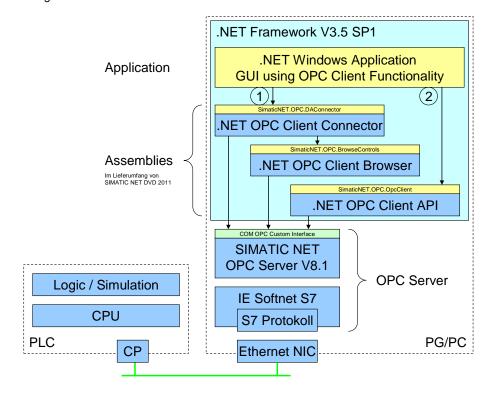
Grundlegende Kenntnisse im Bereich der objektorientierten Programmierung sowie im COM-Umfeld werden vorausgesetzt. Des Weiteren sind Kenntnisse in UML (Unified Modelling Language) von Vorteil.

2.1 Beschreibung der Kernfunktionalität

Beteiligte Software-Komponenten

Folgendes Bild zeigt die beteiligten Software-Komponenten sowie die in diesem Beispiel gezeigte Nutzungsvariante (1):

Abbildung 2-2



PC/PG

Auf dem PC/PG ist zur Visualisierung ein C#-OPC-Client realisiert.

Der OPC-Client nutzt zur Ankopplung an den Prozess die SIMATIC NET .NET Data Controls, die mit dem SIMATIC NET OPC-Servers ab V8.2 automatisch

2.1 Beschreibung der Kernfunktionalität

mitinstalliert werden. Die beiden in diesem Beispiel verwendeten Assemblies (SimaticNET.OPC.DaConnector und SimaticNET.OPC.BrowseControls) kapseln die OPC DA Schnittstelle und bieten dem Client eine einfache Nutzungsmöglichkeit aus .NET Applikationen heraus.

Über die SIMATIC NET SOFTNET-S7-Anbindung und das S7-Protokoll baut der SIMATIC NET OPC-Server die Verbindung zur Steuerung auf.

Steuerung

Die Steuerung liefert die zu visualisierenden Daten. Dazu ist ein einfaches S7-Programm zur Simulation der verschiedenen Datentypen implementiert.

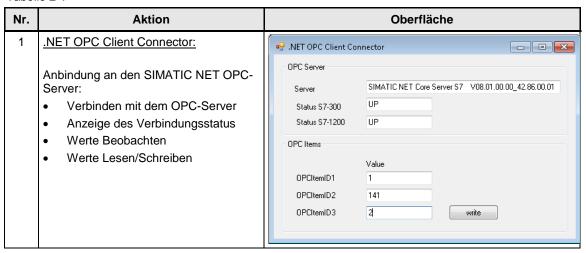
Erstellte Softwarekomponenten

- C# OPC-Client
- STEP 7-Simulationsprogramm

Allgemeiner Ablauf der Applikation

Die Applikation zeigt eine einfache Applikationsvariante, um die Möglichkeiten zum Zugriff zu demonstrieren. Kleinvisualisierungen mit einfachen Oberflächen nutzen das .NET OPC Client Control mit DAConnector Assembly und können ohne Programmieraufwand OPC Items mit Windows Controls "verschalten".

Tabelle 2-1



Hinweis

Die hier verwendeten SIMATIC NET OPC Data Controls für .NET können ausschließlich für die Kopplung mit SIMATIC OPC Server verwendet werden.

2.2 Verwendete Hard- und Software-Komponenten

2.2 Verwendete Hard- und Software-Komponenten

Die Applikation wurde mit den nachfolgenden Komponenten erstellt:

Hardware-Komponenten für die Steuerung

Tabelle 2-2

Komponente	Anz.	Bestellnummer	Hinweis
PS307 5A	1	6ES7307-1EA00-0AA0	
CPU 315-2 PN/DP	1	6ES7315-2AG10-0AB0	Oder eine vergleichbare S7-300 CPU mit PROFINET- Schnittstelle
S7-1200 PM 1207	1	6EP1332-1SH71	
CPU 1214C DC/DC/DC	1	6ES7 214-1AE30-0XB0	Oder eine vergleichbare S7- 1200 CPU
Handelsüblicher Switch	1	Abhängig vom Produkt	

Hardware-Komponenten für den PC

Tabelle 2-3

Komponente	Anz.	Bestellnummer	Hinweis
Field PG M4	1	6ES77160	Oder ein handelsüblicher PC mit entsprechender Software
NDIS-fähige Netzwerkkarte	1	Abhängig vom Produkt	Im Field-PG integriert

Standard Software-Komponenten

Tabelle 2-4

Komponente	Anz.	Bestellnummer	Hinweis
STEP 7 Professional V15.1	1	6ES7822-4AA03-0YA5	
WinCC Professional V15.	1	6AV2103-0DA00-0AM0	Passen Sie die Bestellnummer entsprechend der nötigen Powertags an.
SIMATIC NET IE SOFTNET-S7 (V12)	1	6GK1704-1LW12-0AA0 6GK1704-1CW12-0AA0	LW=8 S7-Verbindungen (Lean), CW=64 S7-Verbindungen
Microsoft Visual Studio 2010	1	Express Edition Standard Edition Professional Edition	Erhältlich im Microsoft Store (http://emea.microsoftstore.com)
.NET Framework 3.5	1	Frei downloadbar bei http://www.microsoft.com/	Wird durch SIMATIC NET installiert

Beispieldateien und Projekte

Die folgende Liste enthält alle Dateien und Projekte, die in diesem Beispiel verwendet werden.

2.2 Verwendete Hard- und Software-Komponenten

Tabelle 2-5

Komponente	Hinweis
21043779_OPCDAConnector_CODE.zip	C# Source-Code der Benutzeroberfläche und archiviertes STEP7 V11 und STEP7 V13- Projekt
21043779_OPCDAConnector_DOKU_V1_1_d.pdf	Dieses Dokument

2.3 Alternativlösungen

2.3 Alternativlösungen

Hier erfahren Sie welche Alternativen zur Lösung des Automatisierungsproblems existieren und welche Charakteristika die Alternativen besitzen.

Nutzung des Standard OPC Foundation .NET Wrapper (Raw Interface)

Prinzipiell kann auf die Standard OPC .NET API zurückgegriffen werden. Dies ist im Dokument "Programmierung eines OPC DA .NET Clients mit C# für den SIMATIC NET OPC Server (COM/DCOM)" beschrieben.

Dies hat folgende Vor- und Nachteile:

Tabelle 2-6

	Vorteile		Nachteile
•	Anbindung an OPC Server anderer Hersteller möglich.	•	Komplexe API erschwert deren Verwendung.
•	Volle Funktion der API nutzbar (volle Flexibilität)	•	Basis Funktionalität eines Clients (z. B. reconnect bei Verbindungsabbruch) muss selbst implementiert werden.
		•	Verwaltungsfunktionen müssen selbst implementiert werden.
		•	Einfaches Verschalten mit Standard .NET Controls nicht ohne Weiteres möglich.

Gegenüberstellung von OPC .NET Wrapper (Raw Interface) und SIMATIC NET OPC Data Controls (Toolkit)

Aufgrund der in der OPC-Spezifikation festgelegten Datenstrukturen, ist das OPC-Interface als Custom-Interface implementiert. OPC-Clients, die in C# implementiert sind, greifen über den RCW (runtime callable wrapper) zu. Dieser Wrapper wird von der OPC Foundation zur Verfügung gestellt und reicht das rohe OPC-Interface durch (COM Interop). Der OPC .NET Wrapper bietet damit die volle Funktionalität und die volle Flexibilität. Allerdings müssen auch alle Verwaltungsfunktionen vom Client aus implementiert werden. Ebenso wie generisches Clientverhalten z. B. automatisches Wiederverbinden nach Verbindungsabbruch.

Die SIMATIC NET OPC Data Controls für .NET bilden ein Client-Toolkit mit einer stark vereinfachten .NET API. Damit werden wesentliche Verwaltungsfunktionen und logisches Verhalten in einem Data Control gekapselt und vor dem C# Programmierer verdeckt. Dadurch vereinfacht sich der Implementierungsaufwand erheblich.

Hinweis

Die SIMATIC NET OPC-Client Controls können sich nur mit SIMATIC NET OPC-Servern verbinden nicht aber mit OPC Servern anderer Hersteller.

2.3 Alternativlösungen

Entscheidungskriterien für die Nutzung verschiedener Bibliotheksvarianten

Die folgende Tabelle zeigt Ihnen die wichtigsten Entscheidungskriterien, die ausschlaggebend sind für die Wahl der einzusetzenden Programmiersprache:

Tabelle 2-7

Kriterium	Beschreibung
Multivendor Fähigkeit	Ist die Nutzung des OPC-Client auch mit anderen OPC Servern gefordert?
Mengengerüst	Welche Datenmengen sollen vom OPC-Client verarbeitet werden?
OPC DA und OPC UA	Soll neben OPC DA auch OPC UA Verbindungen genutzt werden?
Einfache Implementierung	Inwieweit ist die Sprache geeignet, Code relativ einfach und schnell zu implementieren?
OPC Hintergrund Know-how	Wie viel OPC-KowHow ist vorhanden um den Client zu realisieren?
.NET Programmiererfahrung	Wie viel C# Programmiererfahrung ist vorhanden um den Client zu realisieren?
Use Case	Wie komplex ist die Applikation, die die OPC-Daten verarbeitet, wie viel Flexibilität wird am OPC-Interface benötigt?

Gegenüberstellung verschiedener Bibliotheksvarianten

In der nachfolgenden Tabelle werden nun die Programmiervarianten anhand der oben beschriebenen Kriterien gegenübergestellt.

Tabelle 2-8

Kriterium	SIMATIC NET OPC Client Control	SIMATIC NET OPC Client API	OPC Foundation .NET Wrapper
Multivendor Fähigkeit	-	•	✓
Mengengerüst	-	✓	✓
OPC DA und OPC UA	✓ (*)	✓ (*)	nur DA
Einfachheit	++	+	-
OPC Know-how	keins	minimal	+
.NET Know-how	keins	+	++
Flexibilität	-	√	√

(*) klassisch OPC DA und OPC UA parallel, limitiert auf SIMATIC NET OPC Server

3 Funktionsmechanismen dieser Applikation

Einführung

IN den folgenden Kapiteln zeigen wir Ihnen Details zur Programmierung im PC und der S7-CPU.

3.1 Programmierung des Clients unter Verwendung des .NET OPC Data Controls

Übersicht

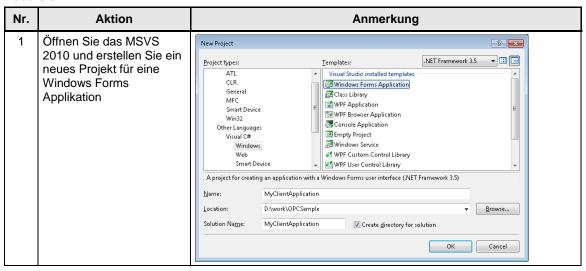
In den folgenden Abschnitten zeigen wir Ihnen, wie die .NET Assemblies in die Entwicklungsumgebung eingebunden werden und wie Sie eine einfache Oberflächen erstellen können.

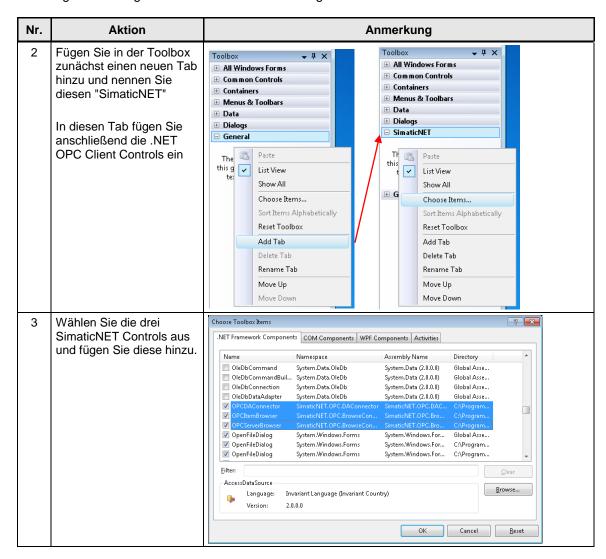
3.1.1 Einbinden der Assemblies

Allgemeines

Um die Funktionalität der .NET OPC Client Controls zu nutzen, müssen die Assemblies zunächst im MS Visual Studio referenziert werden.

Tabelle 3-1



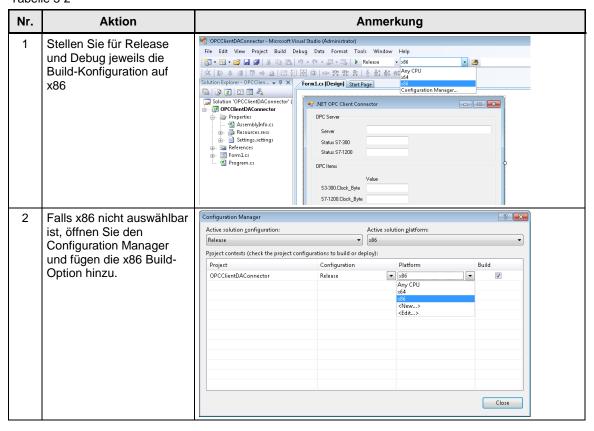


3.1.2 Einstellen des Compilers

Allgemeines

Die zu erstellende Beispielapplikation verwendet Assemblies, die vorkompiliert geliefert werden. Diese sind für x86 Systeme erstellt. Dies bedeutet dass auch die Beispielapplikation für x86 gebaut werden muss. Hierzu wird die Build-Konfiguration im MS Visual Studio für Debug und Release auf x86 eingestellt.

Tabelle 3-2



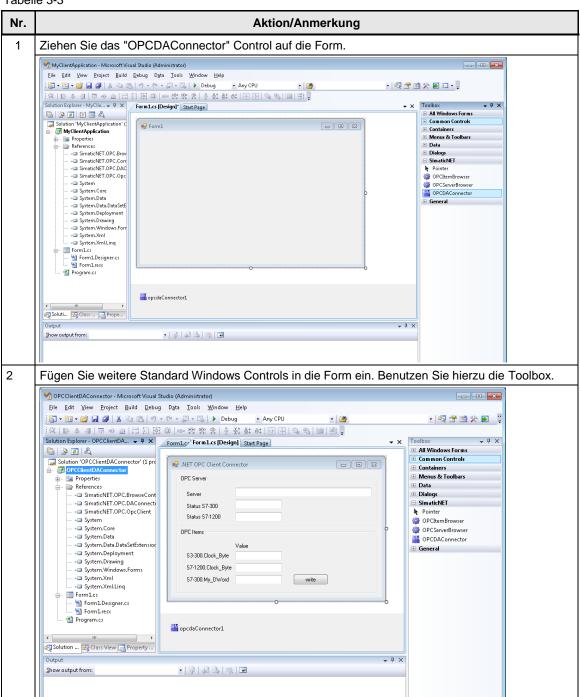
3.1.3 Erstellen der Oberfläche

Allgemeines

Um eine einfache Oberfläche zu erstellen, müssen Sie selbst nichts programmieren. Ziehen Sie die gewünschten Controls auf die Form und geben Sie Ihnen aussagekräftige Namen.

Ziehen Sie das Control "OPCDAConnector" mit Drag&Drop auf die Form. Es hat keine Oberfläche und wird daher im unteren Teil des Designerfensters angezeigt. Dieses Control enthält die vollständige gekapselte OPC-Client Funktionalität.

Tabelle 3-3



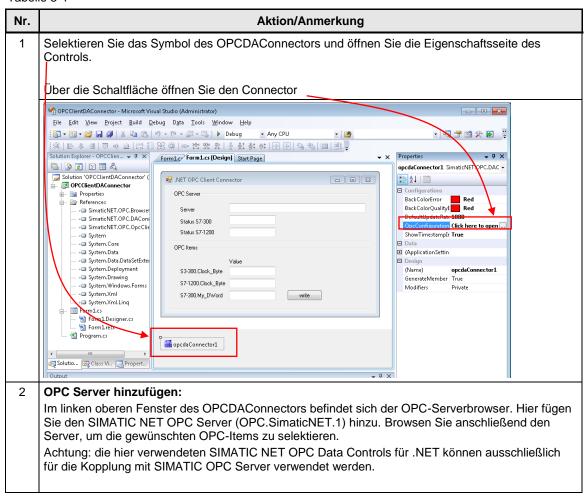
3.1.4 Verbinden der OPC Datenpunkte (Variablen) mit dem Control

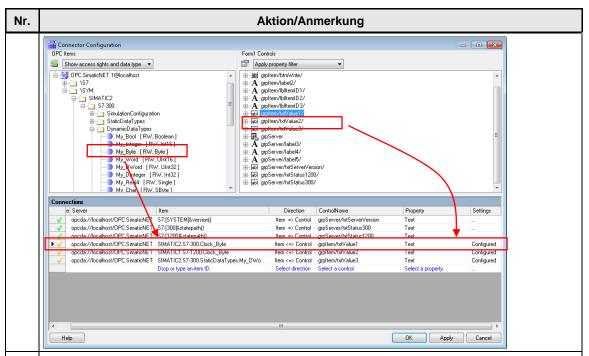
Einführung

Um eine Verschaltung der OPC-Datenpunkte mit den Properties der Standard Windows Controls vorzunehmen, muss der OPCDAConnector gestartet werden.

Projektierungsschritte

Tabelle 3-4





3 Datenübertragungsrichtung

Die Übertragungsrichtung der Daten (Datenfluss) wird für jedes verknüpfte Paar getrennt eingestellt. Möglich sind

- Item => Control
 - Sobald sich das OPC-Item ändert wird der Prozesswert in das Control geschrieben.
- Item <= Control

Es werden nur Werte vom Control in Richtung Item übertragen. Hierzu ist ein Triggerereignis erforderlich.

Item <=> Control

Es ist eine bidirektionale Übertragung möglich, zusätzlich kann ein Triggerereignis konfiguriert werden, dass die Übertragung vom Control zum OPC-Item auslöst (z. B. ein Button_Click_Event)

4 Datentypkonvertierung

Bei der Verschaltung von Properties und OPC-Items ist, je nach Übertragungsrichtung, eine Datentypkonvertierungen erforderlich. Der OPCDAConnector versucht, soweit möglich die Konvertierung automatisch vorzunehmen. Das Validierungsergebnis bezüglich der verschalteten Datentypen wird pro Verbindung mit einem Symbol in der Verbindungsliste angezeigt.

- Grünes Häkchen
 - Ok: die Datentypen passen.
- Oranges Häckchen

Warnung: die Datentypen können ineinander konvertiert werden, allerdings besteht die Gefahr das eine Wertebereichsüberschreitung passiert.

Rotes Kreuz

Fehler: die Datentypen können für die gewählte Übertragungsrichtung nicht ineinander konvertiert werden.

3.1.5 Schreiben von OPC-Items

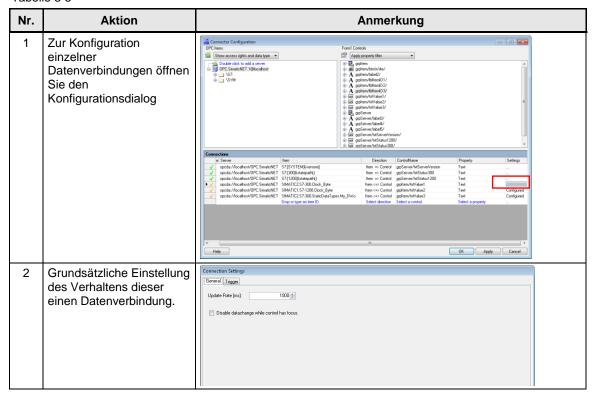
Einführung

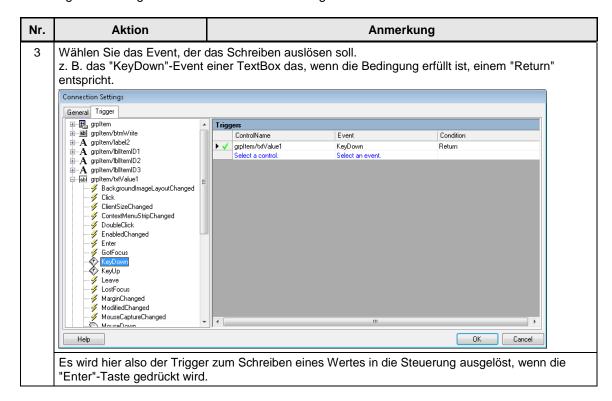
Das Schreiben der OPC-Items direkt aus dem Windows Control kann auf verschiedene Weise erfolgen. Abhängig von den Anforderungen an die Steuerungslogik und der Applikation, ist der OPCDAConnector entsprechend konfigurierbar.

Projektierungsschritte

Typischerweise löst ein Ereignis den Schreibvorgang aus. Dies kann ein Button Click Event sein, der beim Drücken eines Buttons ausgelöst wird, oder aber der Key Press Event, der beim Editieren einer Textbox ausgelöst wird.

Tabelle 3-5





3.2 Erläuterungen zum Simulationsprogramm in der Steuerung

3.2 Erläuterungen zum Simulationsprogramm in der Steuerung

3.2.1 Ablauf der Simulation

Beschreibung des Ablaufs

Die Abläufe innerhalb der Steuerung sind denkbar einfach.

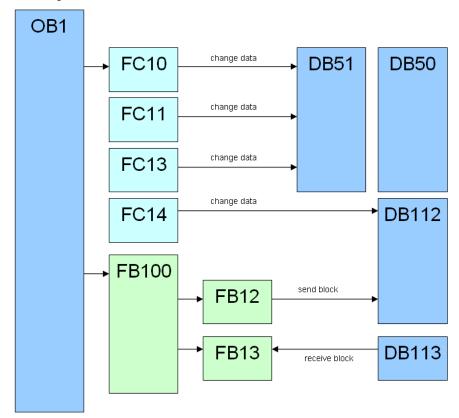
Der Baustein OB1 ruft zyklisch die Funktionen FC10, FC11, FC13 und FC14 auf. Diese Funktionen "bewegen" die Daten im Datenbaustein DB51. Zusätzlich kann in diesem allgemeinen Simulationsprogramm der Funktionsbaustein FB100 aufgerufen werden, der die BSEND, BRCV hantiert um einen Datenblock zu senden bzw. zu empfangen. Der Datenpuffer ist auf 4096 Byte voreingestellt. Die BSEND / BRCV Funktion wird in diesem Beispiel nicht benötigt.

Die Simulation enthält alle einfachen Datentypen und kann einfach um Arrays erweitert werden.

Aufrufpfad der Bausteine

Folgendes Diagramm veranschaulicht den Aufrufpfad:

Abbildung 3-1



3.2 Erläuterungen zum Simulationsprogramm in der Steuerung

3.2.2 Funktion der Bausteine

Aufgaben und Funktion der Bausteine

Die einfache Simulation soll verschiedenen Datentypen bereitstehen und die Daten selbstständig inkrementieren, um unter anderem den DataChange über OPC zu veranschaulichen.

Tabelle 3-6

Baustein	Beschreibung
OB1 Main	Hauptschleife des Programms, ruft alle Unterprogramme auf
OB100 Init	Anlauf OB wird beim Hochlauf einmalig durchlaufen.
FC10 ChangeDateAndTime	Ändert Datum und Zeit in einer Variablen im Datenbaustein 51
FC11 ChangeSimpleTypes	Inkrementiert einfache Datentypen im Datenbaustein 51
FC13 ChangeString	Ändert eine Stringvariable im Datenbaustein 51 inhaltlich und in ihrer Länge
FC14 ChangeSendData	Inkrementiert einen Datenblock im Datenbaustein 112
FB100 InvokeBSENDandBRCV	Ruft die Funktionsbausteine der Blockdienste auf
DB50 StaticDataTypes	Globaler Datenbaustein mit verschiedenen einfachen Datentypen
DB51 DynamicDataTypes	Globaler Datenbaustein mit verschiedenen einfachen Datentypen, die durch die Simulation bewegt werden.
DB112 SendData	Sendepuffer 4096Byte
DB113 RcvData	Empfangspuffer 4096Byte

4 Projektierung eines OPC-Servers

Kernfokus dieser Applikation ist die Nutzung von OPC. Hierzu ist eine Reihe von Projektierungen notwendig, die auf unterschiedlicher Weise vorgenommen werden können.

In diesem Beispiel wird die Projektierung in STEP 7 V15.1 gezeigt.

Hinweis

Dieses Kapitel ist nur relevant, wenn Details zur Projektierung von Interesse sind.

Im mitgelieferten STEP 7-Projekt ist die komplette Projektierung bereits durchgeführt.

4.1 Konfiguration der OPC Server Station

Die Konfiguration und Projektierung der SIMATIC PC Station wird mittels STEP 7 vorgenommen und schrittweise beschrieben. Alternativ kann auch mit Hilfe des NCM-PC Softwarepakets eine Projektierung vorgenommen werden. Die Vorgehensweise ist identisch, es werden jedoch einseitig projektierte Verbindungen verwendet.

Tabelle 4-1

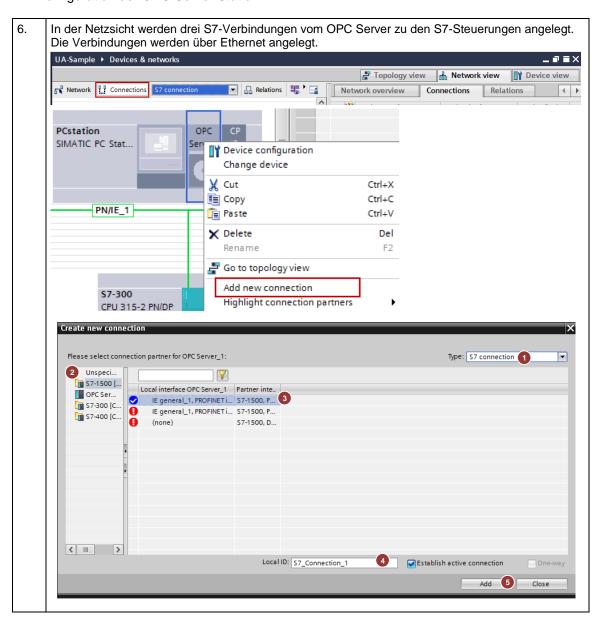
Nr.	Aktion Anmerkung					
1.			ΓΕΡ 7 V15.1.			
	Öffn	en des zu	vor angelegten	Projektes "UA-Sample".		
2.				und Namen vergeben.		
				ss identisch mit dem "Windows chaften → Computername		(siehe
	etwo	▼ 🛅 UA-San				7500
	~ ~		new device			51.
	ces		Add new device			
	Device name:					
		PC station				
		• 🙀				
		▶ 🛅		▼ 🛅 PC systems	Device:	
		→ 🚂	_	▼ 🛄 PC general		
		→ [iii]	_	PC station Industrial PCs		
		• 👼	Controllers	► Industrial PCs ► Industrial PCs ► Industrial PCs ► Industrial PCs	d Controller	
		→ 🛅		SINUMERIK operator o		
		<u> </u>		▶ 🛅 SIMATIC Controller Ap		SIMATIC PC station
				► 🙀 SIMATIC HMI applicati	ion	SIMATIC DC Station
				▶ 🧑 User applications	Order no.:	SIMATIC PC-Statio
		→ [i]	HMI		Version:	V1.0

3. Öffnen Sie die Gerätesicht der PC Station. Fügen Sie dort die Ethernetkarte und den OPC-Server V8.2 ein. **Hinweis** Wollen Sie eine S7-1200 oder eine S7-1500 CPU mit optimierten Datenbausteinen verwenden, dann müssen Sie den OPC Server V12 verwenden. Der Steckplatz muss identisch mit dem Index sein, der in der Konfigurationskonsole vergeben wurde, hier Index "2" für Ethernetkarte. Die Applikation "OPC Server" wurde auf Steckplatz "1" gesteckt. ✓ Filter PC station ☐ SIMATIC Controller Application R SIMATIC HMI application PC station User applications OPC server Application Communications modules ROFINET/Ethernet CP 1604 CP 1612 (A2) CP 1613 (A2) CP 1616 onboard CP 1616 CP 1623 CP 1628 IE general 4. Für die Ethernetkarte wird in den Eigenschaften eine IP-Adresse (hier "192.168.172.1") vergeben und die Karte wird mit dem Ethernetnetzwerk verbunden. IP protocol **✓** Use IP protocol IP address: 192 . 168 . 172 . 1 Subnet mask: 255 . 255 . 255 . 0 Use router Router address: 0 5. In den Eigenschaftsseiten des OPC Servers wird im Register "S7" die Verwendung der symbolischen Adressierung aktiviert. OPC tags

Configuring...

None
All
Configured

Field elements visible during runtime

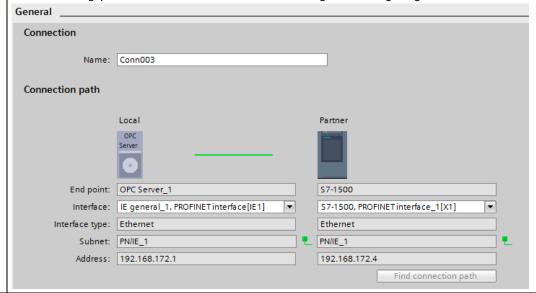


7. S7-Verbindung über Ethernet:

Nachdem die S7-Verbindungen angelegt wurden, werden über einen Klick auf die Verbindung und den Reiter Eigenschaften im Inspektorfenster weitere Einstellungen vorgenommen.

Nachdem der Verbindungsweg gewählt wurde, kann der Verbindungsname geändert werden (hier "Conn001" für die Verbindung zur S7-300, "Conn002" für die Verbindung zur S7-400 und "Conn003" für die Verbindung zur S7-1500).

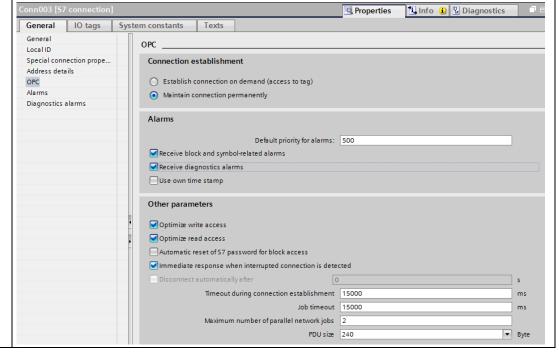
Die Verbindungspartner und die Parameter der Verbindung werden angezeigt.



8. Im den Eigenschaften der S7-Verbindungen wird das Menü "OPC" selektiert. Hier werden verbindungsspezifische Einstellungen vorgenommen.

Der Verbindungsaufbau wird permanent eingestellt, damit die Verbindung gehalten wird, auch wenn gerade nicht kommuniziert wird.

Die Verbindung wird für die Übertragung von Alarmen und Diagnose-Events konfiguriert. Weiterhin wird die sofortige Reaktion auf Verbindungsunterbrechungen aktiviert um unnötige Wartezeiten auf Timeouts zu vermeiden.



4.2 Endkontrolle der Einstellungen

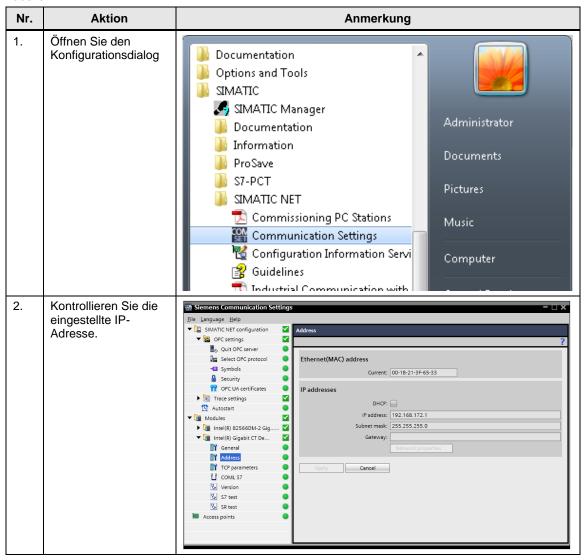
9. Nachdem die Verbindungen angelegt sind, muss die neue Konfiguration in die Stationen geladen werden.

Die PC Station kann auch mit der XDB-Datei, wie in Kapitel 6 beschrieben, geladen werden.

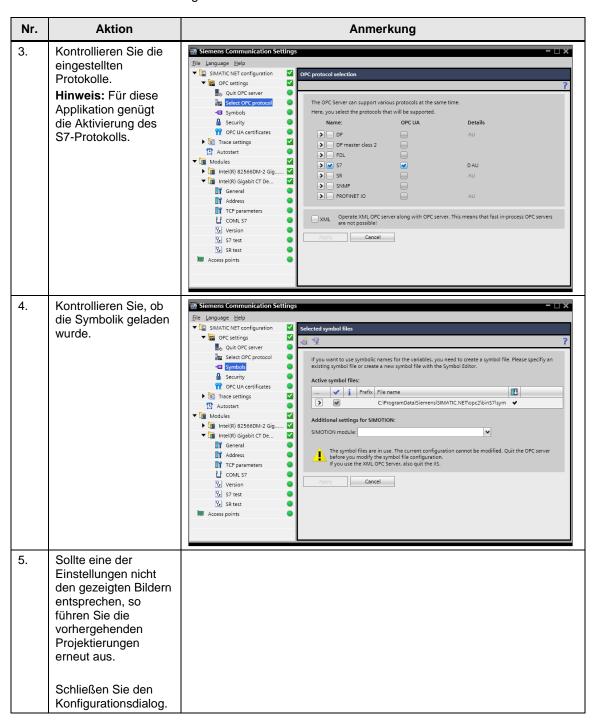
4.2 Endkontrolle der Einstellungen

Die Einstellungen können mit dem Konfigurations-Konsole **Kommunikations-Einstellung** (**Communication Settings**) kontrolliert werden.

Tabelle 4-2



4.2 Endkontrolle der Einstellungen



5.1 Installation der Hard- und Software

5 Installation und Inbetriebnahme

5.1 Installation der Hard- und Software

In diesem Kapitel wird beschrieben, welche Hardware- und Softwarekomponenten installiert werden müssen. Die Beschreibungen und Handbücher sowie Lieferinformationen, die mit den entsprechenden Produkten ausgeliefert werden, sollten in jedem Fall beachtet werden.

Installation der Hardware

Die Hardwarekomponenten entnehmen Sie bitte dem Kapitel 2.2. Gehen Sie für den Hardwareaufbau gemäß folgender Tabelle vor:

ACHTUNG Schalten Sie die Spannungsversorgung erst nach dem letzten Schritt zu.

Tabelle 5-1

Nr.	Fokus	Aktion
1	Steuerung – S7-300 Station	Bauen Sie die Station analog zur Abbildung in Kap. 2 auf.
2	Steuerung – S7-1200 Station	Bauen Sie die Station analog zur Abbildung in Kap. 2 auf.
3	PG/PC Station	Bauen Sie die Station analog zur Abbildung in Kap. 2 auf.
4	Industrial Ethernet	Verbinden Sie die Steuerung mit dem PG analog zur Abbildung in Kap. 2.

Installation der Standard Software

Auf dem PG/PC muss STEP 7 V15.1 und SIMATIC NET installiert werden. Auf SIMATIC PGs ist STEP 7 V15.1 bereits vorinstalliert.

Auf die Beschreibung der Installation von STEP 7 V15. und SIMATIC NET wird an dieser Stelle verzichtet. Die Installation findet in gewohnter Windowsumgebung statt und ist selbsterklärend bzw. in den entsprechenden Handbüchern beschrieben.

Adressübersicht der beteiligten Baugruppen

Falls Sie das Projekt an einem vorhandenen Industrial Ethernet betreiben wollen, müssen Sie folgende Adressvergabe beachten:

Tabelle 5-2

Fokus Baugruppe		IP-Adresse
PG/PC NDIS-Netzwerkkarte		192.168.172.1
Steuerung	CP 343-1	192.168.172.2
Steuerung	CPU 1214C	192.168.172.4

5.1 Installation der Hard- und Software

Hinweise

- Beachten Sie die richtige Subnetz-Maske 255.255.25.0.
- Alternativ ist es auch möglich, im STEP 7-Projekt die zugewiesenen IP-Adressen zu ändern.

Einstellen der IP Adresse

Die Ethernet-Netzwerkkarte muss in den projektierten Betrieb geschaltet werden. Hierzu muss die PC-Station konfiguriert werden.

Hinweis

Stellen Sie sicher, dass die Netzwerkkarte die feste IP-Adresse 192.168.172.1 hat (diese können Sie über die Netzwerkeinstellungen und den TCP/IP-Eigenschaften einstellen), wenn Sie das mitgelieferte Projekt verwenden wollen.

5.2 Laden der der PC Station über STEP 7 V1x

Installation des STEP 7 V1x Projekts via TIA Portal

Die PC-Station kann direkt aus dem TIA-Portal geladen werden. Alternativ hierzu kann eine PC-Station auch über den Komponenten Konfigurator und die XDB-Datei konfiguriert werden (Siehe Kap.5.3).

Tabelle 5-3

Nr.	Aktion	Anmerkung
1	Entpacken Sie das TIA-Projekt: STEP7_TIA15.1.zip	Entpacken Sie das Projekt in einem Pfad in dem Sie Lese- und Schreibrechte besitzen.
2	Öffnen Sie das TIA-Portal und navigieren Sie zum Projekt mittels der Browserfunktion	Takethy biorgrand Anthropisch Start Copes einteling project Copes ei
3	Bestätigen Sie mit Öffnen.	OPCSample OPCSample.ap15 OPCSample OPCSample
4	Nach dem Öffnen wechseln Sie in die Projektansicht	Somewhat Continued Somewhat

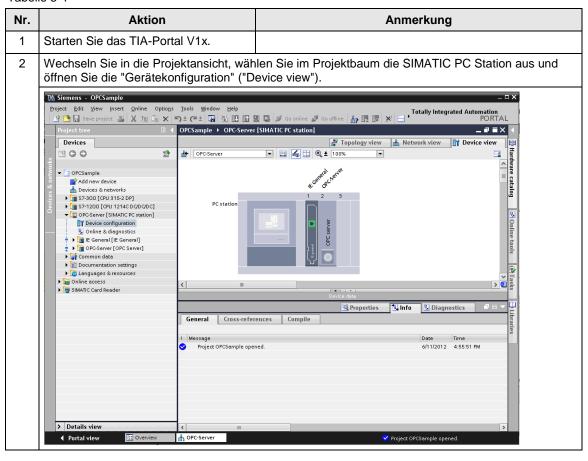
Nr.	Aktion	Anmerkung
5	Laden Sie die PC-Station	Stronger Of Comple
	Alternative: Sie können die PC-Station auch über den Import der mitgelieferten XDB-Datei konfigurieren. Siehe Kap 5.3	Contract Contract

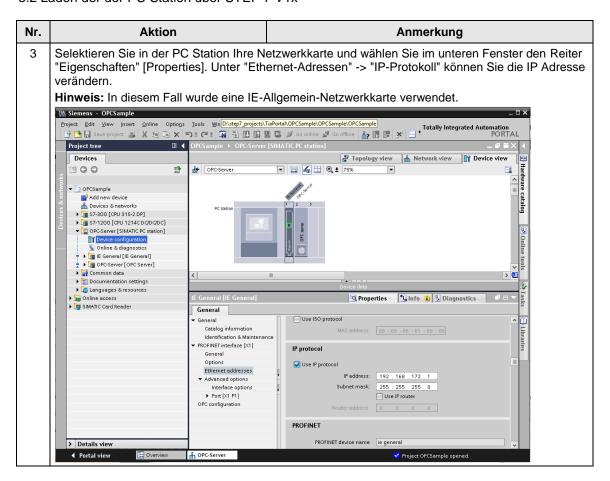
Ändern der IP Adresse der PC Station in STEP 7 V1x

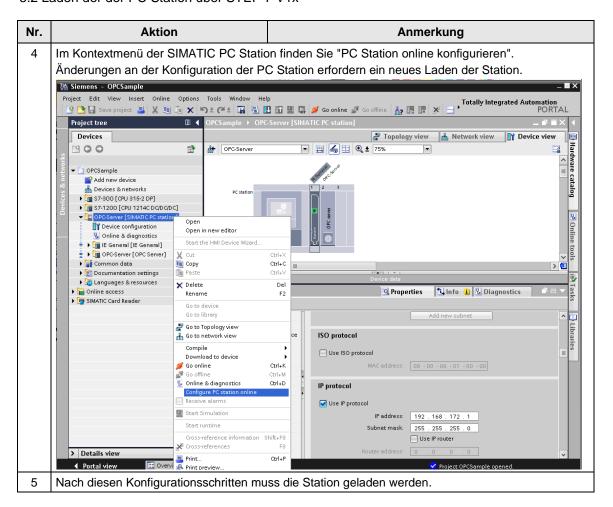
Hinweis

Die IP-Adresse im STEP7-Projekt muss mit der IP-Adresse Ihrer physikalischen Ethernet-Schnittstelle übereinstimmen.

Tabelle 5-4







5.3 Importieren der XDB-Datei in den Komponenten Konfigurator

5.3 Importieren der XDB-Datei in den Komponenten Konfigurator

Einleitung

Eine PC-Station kann alternativ zum Laden aus dem TIA-Portal (siehe Kap. 5.2) auch über den Komponenten Konfigurator und die XDB-Datei konfiguriert werden. Die XDB-Datei ist im mitgelieferten TIA-Projekt bereits vorhanden.

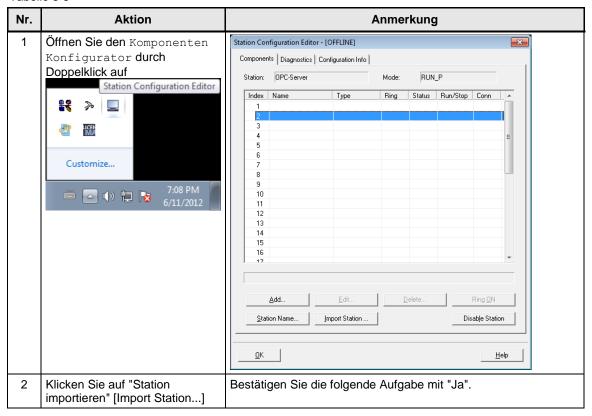
Einstellen der IP Adresse

Die Ethernet-Netzwerkkarte muss in den projektierten Betrieb geschaltet werden. Hierzu muss die PC-Station konfiguriert werden.

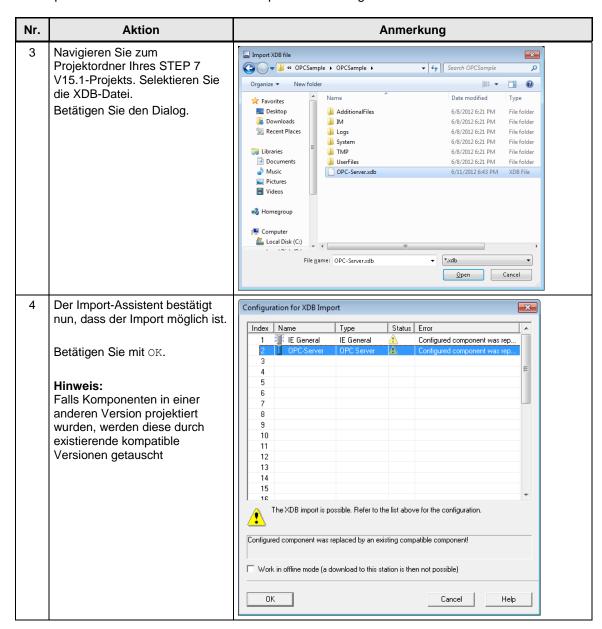
Hinweis

Stellen Sie sicher, dass die Netzwerkkarte die feste IP-Adresse 192.168.172.1 hat (diese können Sie über die Netzwerkeinstellungen und den TCP/IP-Eigenschaften einstellen), wenn Sie das mitgelieferte Projekt verwenden wollen.

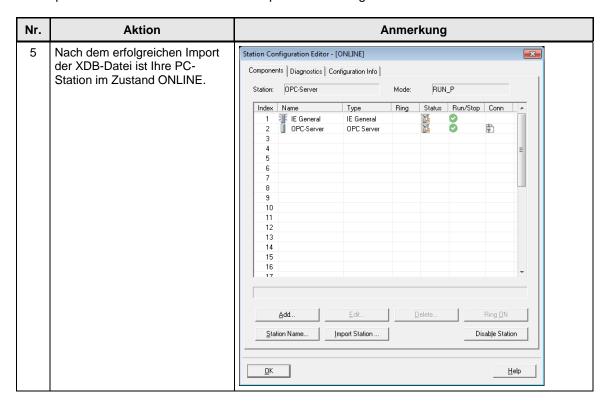
Tabelle 5-5



5.3 Importieren der XDB-Datei in den Komponenten Konfigurator



5.3 Importieren der XDB-Datei in den Komponenten Konfigurator



5.4 Installation des OPC-Clients auf dem PC/PG

5.4 Installation des OPC-Clients auf dem PC/PG

Die Applikationssoftware wird in einer Archivdatei geliefert. Entpacken Sie das Archiv in einem Verzeichnis auf dem PG/PC. Beachten Sie das Sie Lese- und Schreibrechte benötigen.

ACHTUNG

Falls Sie ein älteres Betriebssystem als Windows 7 SP1 nutzen und dort nicht die SIMATIC NET PC Software V8.x installiert wurde, so müssen Sie zuerst das .NET-Framework 3.5 +SP1 installieren.

Informationen hierzu erhalten Sie auf den Microsoft Internet Seiten http://www.microsoft.com/

Gehen Sie zur Installation der Bedienoberfläche folgendermaßen vor:

Tabelle 5-6

Nr.	Aktion	Anmerkung
1	Entpacken Sie die Datei : 21043779_OPCDAConnector_CODE.zip	Diese zip-Datei enthält sowohl das STEP 7 V15.1 Projekt, als auch den OPC-Client mit C# Source-Code.
2	Entpacken Sie die Datei : Csharp_OPCDAConnector_CODE.zip	
3	Im Verzeichnis \OPCClientDAConnector\bin\x86\Release finden Sie die Datei: OPCClientDAConnector.exe	Die EXE kann nur ausgeführt werden, wenn die dazugehörigen Assemblies im selben Verzeichnis liegen

Enthaltene Dateien

Die Archivdatei enthält die MS Visual Studio Solution Datei und den Source Code, sowie vorkompilierte Binärdateien für x86 Systeme. Im Unterordner befindet sich die ausführbare Datei (EXE), sowie die benötigten Assemblies.

Verzeichnis: \OPCClientDAConnector\bin\x86\Release

Tabelle 5-7

Datei	Gehört zu
OPCClientDAConnector.exe	Die hier beschriebene OPC-Client Applikation
OPCClientDAConnector.pdb	Symbolinformation zum Debuggen
OPCClientDAConnector.vshost.exe	Hostprozess für den Debugger
SimaticNET.OPC.BrowseControls.dll	Browse Control
SimaticNET.OPC.Common.dll	OPC-RC-Wrapper
SimaticNET.OPC.DAConnector.dll	Connector Control
SimaticNET.OPC.OpcClient.dll	Client API

Deinstallation

Die Deinstallation erfolgt durch Löschen des Verzeichnisses.

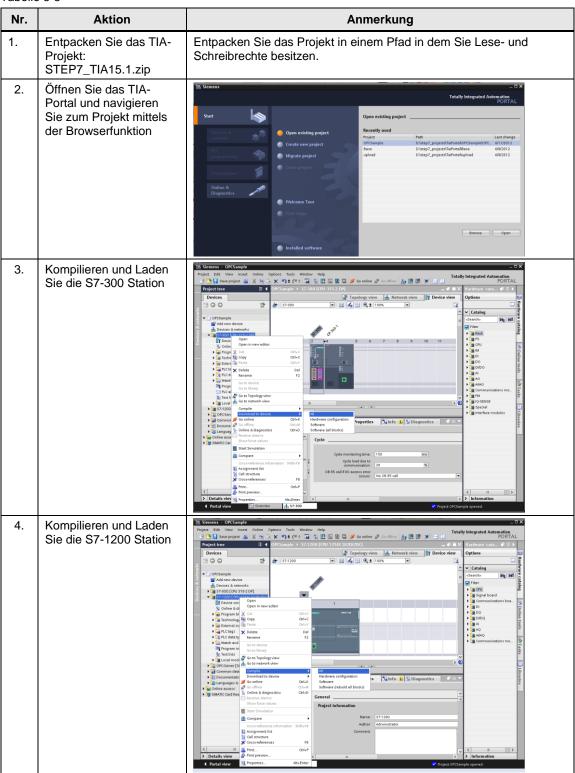
5.5 Laden der Simulation auf die S7-Stationen

5.5 Laden der Simulation auf die S7-Stationen

Laden des TIA-Projekts

Gehen Sie dazu wie folgt vor:

Tabelle 5-8

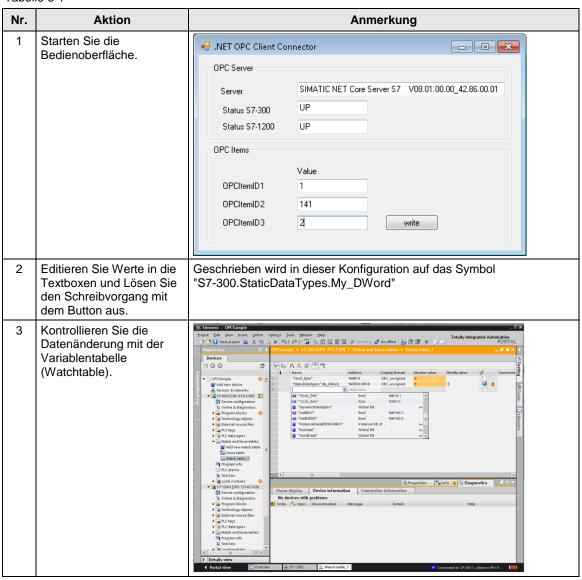


6 Bedienung der Applikation

Inbetriebnahme der Simulation

Um die Simulation in Betrieb zu nehmen, sind keine weiteren Schritte nötig:

Tabelle 6-1



7 Glossar

COM / DCOM

COM: Component Object Modell: Softwaremodell zur Kommunikation zwischen Komponenten, das auf einer einheitlichen Schnittstelle aufsetzt; DCOM: Softwaremodell zur Kommunikation über Rechnergrenzen hinweg, welches auf COM aufsetzt.

Event-Handler

Ein Event-Handler bearbeitet aufgetretene Ereignisse bzw. Windowsnachrichten.

Exception

Unter einer Exception versteht man eine Ausnahmesituation. Diese kann entweder vom Betriebssystem (z. B. Division durch Null) oder vom Anwenderprogramm generiert werden.

Exception-Handler

Ein Exception-Handler bearbeitet aufgetretene Ausnahmesituationen. Dies ist in der Regel ein gesichertes Fehlverhalten und/oder eine Nachricht an den Nutzer.

HRESULT

Rückgabe-Datentyp von COM-Objekten.

IDL

Interface Definition Language: eine von Microsoft standardisierte Sprache zur Definition von Funktions- und Parameterschnittstellen.

Polling

Englischer Begriff, mit dem das (meist zyklische) Abfragen bestimmter Werte oder Zustände gemeint ist.

Sinkinterface

Mit Hilfe des Sink-Interfaces können Nachrichten zwischen Komponenten gesendet werden. Das Sinkinterface setzt auf COM-Mechanismen auf.

Thread

Durch "Threads" lassen sich innerhalb einer Anwendung bzw. eines Prozesses mehrere Codefragmente quasi parallel, also gleichzeitig, ausführen.

Verwendet eine Applikation mehrere Threads, so besitzt die Applikation die Eigenschaft "multi-threaded".

Hat eine Applikation ausschließlich einen Thread, so heißt sie "single-threaded". Bei diesen Applikationen werden somit alle Codefragmente immer sequentiell abgearbeitet.

Windowsnachricht

In den gängigen Microsoft Windows Betriebssystemen werden zur Mitteilung von Ereignissen, z. B. das Paint-Ereignis, Nachrichten ausgetauscht.

Wrapper

Als "Wrapper" wird in der Regel ein Klassenverbund bezeichnet, der andere Klassenverbände zur Datenkonvertierung oder leichteren Verwendung kapselt. Er kann somit als "Hülle" gesehen werden, der die "gewrappten" Klassen umschließt und nach außen verbirgt.

8 Literaturhinweise

Literaturangaben

Diese Liste ist keinesfalls vollständig und spiegelt nur eine Auswahl an geeigneter Literatur wieder.

	Themengebiet		Titel
\1\	OPC	•	OPC DA 2.05 Specification bei http://www.opcfoundation.org/ mn_opc_0.pdf und mn_opc_76.pdf von SIMATIC.NET V8.0
\2\	.NET	•	Inside C#, Tom Archer
		•	.NET Crashkurs, Clemens Vasters, Oellers, Javidi, Jung, Freiberger, DePetrillo
		•	Microsoft .NET Framework Programmierung, Jeffrey Richter

8.1 Internet-Link-Angaben

Diese Liste ist keinesfalls vollständig und spiegelt nur eine Auswahl an geeigneten Informationen wieder.

Tabelle 8-1

	Themengebiet	Titel
\1\	Referenz auf den Beitrag	http://support.automation.siemens.com/WW/view/de/21043779
\2\	Siemens Industry Online Support	http://support.automation.siemens.com
/3/	Beispiel zum Alarm & Events Server	http://support.automation.siemens.com/WW/view/de/26548467
\4\	Beispielclient für .NET zum OPC UA -Server	http://support.automation.siemens.com/WW/view/de/42014088

9 Historie

Tabelle 9-1

Version	Datum	Änderung
V1.0	12/2012	Erste Ausgabe
V1.1	06/2014	Migration auf - STEP 7 V13 - Visual Studio 2010
V1.2	07/2019	Aktualisierung für STEP 7 V15.1