# **Applikation zur Kommunikation**

# applications TOOLS

Einfache Anlagenvisualisierung mit dem OCX Data Control unter VBA for Excel



**Applikation** 

**SIEMENS** 

OPC mit Excel VBA

Beitrags-ID: 23829402

Die Applikationsbeispiele sind unverbindlich und erheben keinen Anspruch auf Vollständigkeit hinsichtlich Konfiguration und Ausstattung sowie jeglicher Eventualitäten. Die Applikationsbeispiele stellen keine kundenspezifische Lösungen dar, sondern sollen lediglich Hilfestellung bieten bei typischen Aufgabenstellungen. Sie sind für den sachgemäßen Betrieb der beschriebenen Produkte selbst verantwortlich. Diese Applikationsbeispiele entheben Sie nicht der Verpflichtung zu sicherem Umgang bei Anwendung, Installation, Betrieb und Wartung. Durch Nutzung dieser Applikationsbeispiele erkennen Sie an, dass Siemens über die beschriebene Haftungsregelung hinaus nicht für etwaige Schäden haftbar gemacht werden kann. Wir behalten uns das Recht vor, Änderungen an diesen Applikationsbeispielen jederzeit ohne Ankündigung durchzuführen. Bei Abweichungen zwischen den Vorschlägen in diesen Applikationsbeispiel und anderen Siemens Publikationen, wie z.B. Katalogen, hat der Inhalt der anderen Dokumentation Vorrang.

#### Gewährleistung, Haftung und Support

Für die in diesem Dokument enthaltenen Informationen übernehmen wir keine Gewähr.

Unsere Haftung, gleich aus welchem Rechtsgrund, für durch die Verwendung der in diesem Applikationsbeispiel beschriebenen Beispiele, Hinweise, Programme, Projektierungs- und Leistungsdaten usw. verursachte Schäden ist ausgeschlossen, soweit nicht z.B. nach dem Produkthaftungsgesetz in Fällen des Vorsatzes, der grober Fahrlässigkeit, wegen der Verletzung des Lebens, des Körpers oder der Gesundheit, wegen einer Übernahme der Garantie für die Beschaffenheit einer Sache, wegen des arglistigen Verschweigens eines Mangels oder wegen Verletzung wesentlicher Vertragspflichten zwingend gehaftet wird. Der Schadensersatz wegen Verletzung wesentlicher Vertragspflichten ist jedoch auf den vertragstypischen, vorhersehbaren Schaden begrenzt, soweit nicht Vorsatz oder grobe Fahrlässigkeit vorliegt oder wegen der Verletzung des Lebens, des Körpers oder der Gesundheit zwingend gehaftet wird. Eine Änderung der Beweislast zu Ihrem Nachteil ist hiermit nicht verbunden.

Copyright© 2006 Siemens A&D. Weitergabe oder Vervielfältigung dieser Applikationsbeispiele oder Auszüge daraus sind nicht gestattet, soweit nicht ausdrücklich von Siemens A&D zugestanden.

Bei Fragen zu diesem Beitrag wenden Sie sich bitte über folgende E-Mail-Adresse an uns:

mailto:csweb@ad.siemens.de



Beitrags-ID: 23829402

#### Vorwort

#### Ziel der Applikation

Der internationale Standard OPC ist für den Zugriff auf Prozessdaten einer SIMATIC S7-Station aus einer Windowsapplikation heraus eine optimierte Schnittstelle.

Die vorliegende Applikation zeigt eine sehr einfache und kostengünstige Vorgehensweise, um individuelle Visualisierungsoberflächen mit Microsoft Excel und VBA zu erstellen. Hierzu wird in dieser Applikation eine einfache Produktionsanlage mit einer SIMATIC S7-Steuerung simuliert. Die dafür benötigten Prozessinformationen werden zwischen der Excel Applikation (OPC-Client) und der S7-CPU ausgetauscht.

#### Kerninhalte dieser Applikation

Folgende Kernpunkte werden in dieser Applikation behandelt:

- Visualisierung/Programmierung
  - Erstellung einer Microsoft Excel/ VBA Applikation und den OPC ActiveX Controls von SIMATIC NET
  - Umgang mit der OPC Standard Schnittstelle für OPC Data Access über das OCX Data Control
  - Verschaltung der SIMATIC NET ActiveX Controls mit dem OCX Data Control
  - Verschaltung von Excel Standard Controls mit S7Variablen
- Projektierung
  - Projektierung einer PC-Station mit OPC-Server und eine Verbindung über die MPI Schnittstelle zur S7-CPU

#### **Abgrenzung**

Diese Applikation enthält keine

- Grundlagen zu Microsoft Excel
- Grundlagenvermittlung zur Programmiersprache Visual Basic for Applikation (VBA)
- Grundlagen zu KOP/ FUP/ AWL.

Grundlegende Kenntnisse über diese Themen werden voraus gesetzt.



Beitrags-ID: 23829402

#### **Aufbau des Dokuments**

Die Dokumentation der vorliegenden Applikation ist in folgende Hauptteile gegliedert.

Teil	Beschreibung
Applikationsbeschreibung	Hier erfahren Sie alles, um sich einen Überblick zu verschaffen. Sie lernen die verwendeten Komponenten kennen (Standard Hard- und Softwarekomponenten sowie die eigens erstellte Anwender Software).
Funktionsprinzipien, und Programmstrukturen	Hier wird auf die detaillierten Funktionsabläufe der beteiligten Hard- und Softwarekomponenten, die Lösungsstrukturen und wo sinnvoll auf die konkrete Implementierung dieser Applikation eingegangen. Sie benötigen diesen Teil, wenn Sie das Zusammenspiel der Lösungskomponenten kennen lernen wollen, um diese z.B. als Basis für eigene Entwicklungen zu verwenden.
Aufbau, Projektierung und Bedienung der Applikation	Dieser Teil führt Sie Schritt für Schritt durch den Aufbau, wichtige Projektierungsschritte, Inbetriebnahme und Bedienung der Applikation.
Anhang	Hier finden Sie weiter führende Informationen, wie z. B. Literaturangaben, Glossare etc

#### Referenz zum Automation and Drives Service & Support

Dieser Beitrag stammt aus dem Internet Applikationsportal des Automation and Drives Service & Support. Durch den folgenden Link gelangen Sie direkt zur Downloadseite dieses Dokuments.

http://support.automation.siemens.com/WW/view/de/23829402

#### Beitrags-ID: 23829402

#### Inhaltsverzeichnis

nhalts	verzeichnis	5	
Applika	ntionsbeschreibung	7	
	Automatisierungsaufgabe		
1.1	Übersicht	7	
1.2	Anforderungen im Detail	8	
2	Automatisierungslösung		
2.1	Übersicht zur Gesamtlösung		
2.2	Beschreibung der Kernfunktionalität		
2.3	Benötigte Hard- und Software-Komponenten		
2.4	Alternativlösungen		
2.4.1	Alternative zum OCX Data Control unter Excel		
2.4.2	Programmierung eines OPC-Client unter einer anderen Hochsprache		
2.4.3	Nutzung eines Standard B&B Systems	. 16	
unktic	onsprinzipien und Programmstrukturen	. 17	
3	Generelle Funktionsmechanismen	. 17	
3.1	Grundlagen zum Thema OPC	. 17	
3.2	Multi Point Interface	. 20	
ļ.	Funktionsmechanismen dieser Applikation	. 21	
<b>l</b> .1	Funktionsprinzipien des OCX Data Control	. 21	
1.2	Exemplarische Projektierung des OCX Data Control	. 29	
1.3	STEP7 Beispiel-Programm	. 31	
5	Erläuterungen zum Beispielprogramm	. 33	
5.1	Fehlerhandling im OPC Client	. 33	
5.2	Erläuterungen zum VBA-Beispielprogramm		
5.3	Erläuterungen zum STEP7 Programm	. 38	
Aufbau	, Projektierung und Bedienung der Applikation	. 40	
6	Installation und Inbetriebnahme		
6.1	Installation der Hard- und Software		
6.2	Projektierung des OPC Clients		
6.3	Konfiguration des Komponeten Konfigurators		
6.4	Konfiguration des Engineering PG/PC		
6.5	Inbetriebnahme der SIMATIC PC-Station		
6.6	Inbetriebnahme der SIMATIC S7-Station	. 47	
7	Konfiguration und Projektierung		
7.1	Projektierung einer neuen CPU in der SIMATIC S7-Station		
7.2	Änderung der ItemIDs im VBA-Projekt	. 53	

# **SIEMENS**

OPC mit Excel VBA

Beitrags-ID: 23829402

8	Bedienung der Applikation	56
Anha	ing und Literaturhinweise	59
9	Literaturhinweise	59
9.1	Literaturangaben	59
9.2	Internet-Link-Angaben	59
9.3	Verweise auf weitere OPC-Applikationen	60

OPC mit Excel VBA Beitrags-ID: 23829402

#### **Applikationsbeschreibung**

#### Inhalt

Hier verschaffen Sie sich einen Überblick über diese Applikation Sie lernen die verwendeten Komponenten kennen (Standard Hard- und Softwarekomponenten sowie die eigens erstellte Anwender Software).

#### 1 Automatisierungsaufgabe

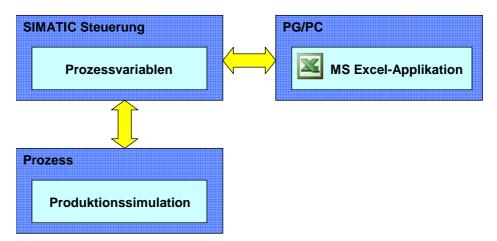
#### Hier erfahren Sie...

welche Automatisierungsaufgabe in der vorliegenden Dokumentation thematisiert wird.

#### 1.1 Übersicht

#### Überblick über die Automatisierungsaufgabe

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



#### Beschreibung der Automatisierungsaufgabe

Durch eine möglichst kostengünstige Ankopplung an eine S7-CPU soll eine einfache individuelle Visualisierungsoberfläche für eine kleine Produktionsanlage mit MS Office Mitteln für den unteren Leistungsbereich realisiert werden.

Mit Hilfe eines in VBA programmierten OPC-Client zeigt diese Applikation, wie ein Produktionsprozess visualisiert werden kann. Durch Betätigung von Elementen auf der Bedienoberfläche wird das Anwenderprogramm auf der S7-CPU, das diesen Prozess simuliert, beeinflusst. Anderseits wird es ermöglicht größere Datenmengen aus der Steuerung zu lesen und auf Visualisierungselementen darzustellen.

8/60

# **SIEMENS**

OPC mit Excel VBA

#### 1.2 Anforderungen im Detail

#### Anforderungen durch die Automatisierungsaufgabe

- Die Applikation soll kostengünstig sein.
- Der gesamte Produktionsprozess soll mit einer VBA-Bedienoberfläche in Excel visualisiert werden.

Beitrags-ID: 23829402

- Die Verbindung zur Steuerung erfolgt über die MPI-Schnittstelle der S7-CPU.
- Die Applikation soll zeigen, wie der SIMATIC NET OPC-Server mittels dem OCX Data Control in VBA hier eingesetzt werden kann.
- Realisierung einer einfachen
  - Bedienoberfläche zum Steuern der Anlage
  - Rezeptverwaltung
  - Darstellung zeitlicher Verläufe

#### Anforderung an die Visualisierungsoberfläche

- Die Bedienoberfläche soll einfach gestaltet sein.
- Über eine graphische Oberfläche sollen Prozessvariablen in der S7-CPU gesteuert und visualisiert werden.
- Es soll prinzipiell gezeigt werden wie S7-Prozessdaten mit
  - SIMATIC NET ActiveX Controls (ohne Programmierkenntnisse) und
  - Excel Standard Controls (individuelle Programmierung)

verschaltet und angezeigt werden.

- Es soll gezeigt werden wie auch größere Datenmengen in die S7-CPU geschrieben oder von dieser gelesen werden
- Es soll gezeigt werden wie die g\u00e4ngigsten S7-Variablentypen in VBA konvertiert und dargestellt werden

#### Anforderungen an die Steuerung

- · Simulation eines technischen Prozesses
- Die Kommunikation soll ohne spezielle Kommunikationsbausteine erfolgen (Variablendienste der S7-Kommunikation)

9/60

OPC mit Excel VBA

# Automatisierungslösung

#### Hier erfahren Sie...

welche Lösung für die Automatisierungsaufgabe gewählt wurde.

Beitrags-ID: 23829402

#### 2.1 Übersicht zur Gesamtlösung

#### **Schema**

2

Die folgende Abbildung zeigt schematisch die wichtigsten Komponenten der Lösung:

Abbildung 2-1

# S7-300 Station PG/PC - PS307 5A - Windows XP Professional - CPU 313-C - Office 2003 -STEP7 V5.4 -SIMATIC NET PC Software 2005 V 6.3 -CP5611 - CP5611

#### **Aufbau**

Der Hardwareaufbau der Automatisierungslösung besteht aus einer S7-300 Station und einem PG/PC, die über MPI miteinander verbunden sind:

- das PG/PC über den CP5611 A2
- die CPU über die integrierte MPI-Schnittstelle

#### 2.2 Beschreibung der Kernfunktionalität

#### Funktionsumfang des Beispiels

Die Excel Projektmappe beinhaltet drei Tabellenblätter, die beispielhaft eine vereinfachte Eisproduktion simulieren.

In der folgenden Tabelle sehen Sie, welches Tabellenblatt welche Hauptfunktionen erfüllt.



Beitrags-ID: 23829402

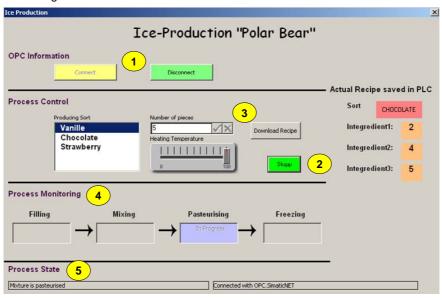
Tabelle 2-1

Tabellenblatt Name	Funktion	
Process	UserForm mit Bedienelementen für die Produktion	
	Zustandsanzeige einer Schrittkette	
Recipe	<ul> <li>Rezeptverwaltung f ür die Produktion</li> </ul>	
Archive	Diagrammaufzeichnungen eines Prozessverlaufs	
	Archivierte Produktionsdaten	

#### Bedienmaske "Process"

Der Dialog "Ice Production" ist eine in VBA eingefügte UserForm und wird beim Aktivieren des Excel-Sheet "Process" aktiviert. Hier ist das OCX Data Control eingebunden.

#### Abbildung 2-2



#### Funktionalität der Oberfläche

Die Oberfläche dieses Dialoges erfüllt dabei folgende Funktionalitäten:

Tabelle 2-2

Bedienelement	Funktion
1.	Verbindung zum OPC Server aufbauen und abbauen.
2.	Steuerung starten und stoppen.
3.	Produktionssorte, Stückzahl und Temperatur auswählen und Rezeptdaten in die CPU laden
4.	Prozessüberwachung
5.	Prozessstatus-Anzeige

Copyright © Siemens AG 2006 All rights reserved 23829402\_OPC\_DATCON\_EXCEL\_DOKU\_v10\_d.doc

OPC mit Excel VBA Beitrags-ID: 23829402

#### Bedienmaske "Recipe"

Auf der Oberfläche dieses Excel Tabellenblattes werden die Sollwerte für die Zutaten der einzelnen Produktionsarten angezeigt. Diese Werte können geändert werden und bei Aktivieren des Buttons Download Recipe im Dialog in die CPU übernommen.

Folgende Abbildung zeigt das Excel Tabellenblatt Recipe.

Abbildung 2-3

# Indication of the quantity required for each recipe and ingredient in process

required	for each recipe	and ingredient	in procent
Vanillle			
	Integredient1:	9	
	Integredient2:	10	
	Integredient3:	5	
Chocolat	e		_
	Integredient1:	2	
	Integredient2:	4	
	Integredient3:	5	T
Strawber	ry		
	Integredient1:	30	
	Integredient2:	30	N. C.
	Integredient3:	30	10



#### Beitrags-ID: 23829402

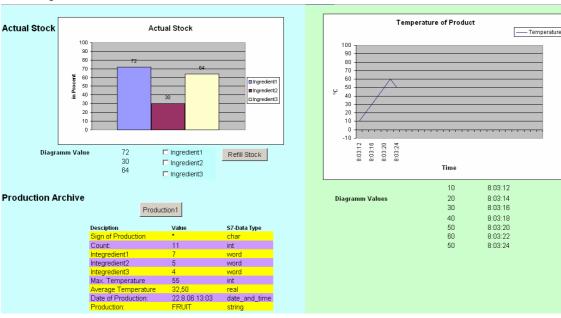
#### Bedienmaske "Archive"

Die Oberfläche dieses Tabellenblattes erfüllt dabei drei unterschiedliche Funktionalitäten:

- Actual Stock: Diagrammanzeige mit dem aktuellen Lagerbestand.
- Production Archive: Rezept der letzten Produktion
- Recording of Temperature: Aufzeichnung der Temperaturänderung während eines Produktionsvorganges in Diagrammform

Folgender Screenshot zeigt das Tabellenblatt "Archive".

#### Abbildung 2-4



#### Vorteile der Applikationslösung

 Kostengünstig: MPI Schnittstelle standardmäßig auf jeder CPU vorhanden; somit keine teure CPU nötig

Beitrags-ID: 23829402

- Minimaler Hardwareaufbau
- Keine zusätzliche Entwicklungsumgebung nötig, da VBA-Editor bei Microsoft Office integriert.

#### Vorteile des Einsatzes von OPC

Der Einsatz des SIMATIC NET OPC Server zur Anlagenvisualisierung hat folgende Vorteile:

- Kostengünstig, da OPC-Server im Lieferumfang der SIMATIC NET Software enthalten.
- Einfache Projektierung des OPC-Servers (wie gewohnt in STEP 7).
- Effizienter Datenaustausch von einer Prozessvariablen zu einer weiterverarbeitenden Applikation.
- Pflege und Verbreitung durch OPC-Foundation.
- Datenpakete bis zu 65KByte.
- Verwendung einer einfachen Script Sprache für die OPC-Programmierung.



2.3

#### Benötigte Hard- und Software-Komponenten

#### Hardware-Komponenten

Tabelle 2-3

Komponente	MLFB/Bestellnummer	Hinweis
PS305 5A	6ES7 307-1EA00-0AA0	Oder ähnliche PS
CPU 313C	6ES7 313-5BE01-0AB0	Alternativ kann auch eine vergleichbare CPU verwendet werden.
MICRO MEMORY CARD 8 MByte	6ES7953-8LP11-0AA0	Oder ähnliche
Field-PG oder gleichwertiger PC mit PB/ MPI-Karte	6ES7 711-xxxxx-xxxx	Projektierungsrechner Link zum PG-Konfigurator
CP5611 A2	6GK1561-1AA01	Optional: MPI Karte zum Anschluss in einem PC
MPI Kabel	6ES7 901-0BF00-0AA0	

Beitrags-ID: 23829402

#### **Standard Software-Komponenten**

Tabelle 2-4

Komponente	MLFB/Bestellnummer	Hinweis
SIMATIC S7 STEP7 V 5.4	6ES7810-4CC08-0YA5	
SIMATIC NET SOFTWARE EDITION 2005	6GK1704-0AA07-3AA0	OPC-Server, OCX Data Control
SIMATIC NET PB SOFTNET-S7/2005	6GK1704-5CW63-3AA0	Software + Lizenz
Microsoft Office 2003		Bestellbar über Microsoft

#### Beispieldateien und Projekte

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

Tabelle 2-5

Komponente	Hinweis
23829402_OPC_DATCON_EXCEL_CODE_v10.zip	Diese gepackte Datei enthält das STEP 7 Projekt und die Excel Arbeitsmappe
23829402_OPC_DATCON_EXCEL_DOKU_v10_d.pdf	Dieses Dokument.

Copyright © Siemens AG 2006 All rights reserved 23829402\_OPC\_DATCON\_EXCEL\_DOKU\_v10\_d.doc

OPC mit Excel VBA Beitrags-ID: 23829402

#### 2.4 Alternativlösungen

#### 2.4.1 Alternative zum OCX Data Control unter Excel

Programmierung eines OPC-Client unter Verwendung der Automation Schnittstelle des SIMATIC NET OPC-Servers.

#### Vorteile:

- Leicht programmierbar durch Verwendung der Scriptsprachen VBA
- Schnelle Erstellung von Oberflächen direkt im Excel-Sheet oder als UserForm
- Integration von ActiveX-Komponenten

#### Nachteile:

- Eingeschränkte Leistungsfähigkeit durch Skriptsprache
- Nur mittlere Anzahl an Variablen möglich

#### 2.4.2 Programmierung eines OPC-Client unter einer anderen Hochsprache

#### C/C++/C# unter Verwendung der Custom Schnittstelle

#### Vorteile:

- Hohe Performanz durch Hochsprachenprogrammierung.
- Sehr hohe Ausdrucksstärke und Flexibilität
- Hohe Anzahl an Variablen möglich
- Paralleles Ausführen mehrerer Funktionalitäten möglich
- Web-Anwendungen mit .NET Sprache C# programmierbar

#### Nachteile:

- lange Einarbeitungszeiten erforderlich
- höherer Aufwand für die Erstellung komplexerer Oberflächen
- bei C# zusätzlicher Runtime Callable Wrapper als Zwischenschicht für den Zugriff auf den OPC-Server nötig.

## Visual Basic V 6.0 unter Verwendung der Automation Schnittstelle

#### Vorteile:

- Integration von ActiveX-Komponenten
- Entwicklung einer Anwendung innerhalb kurzer Zeit möglich, da nur kurze Einarbeitungszeit.

Copyright © Siemens AG 2006 All rights reserved 23829402\_OPC\_DATCON\_EXCEL\_DOKU\_v10\_d.doc

OPC mit Excel VBA

#### Nachteile:

- Schlechte Performance durch die Visual Basic Runtime und Wrapper als zusätzlicher Softwareschicht
- Paralleles Ausführen mehrerer Funktionalitäten nicht möglich

Beitrags-ID: 23829402

#### 2.4.3 Nutzung eines Standard B&B Systems

Verwendung von WinCC bzw. WinCC flexible als OPC-Client.

#### Vorteile:

- Integrierte Funktionen zum Bedienen Beobachten, Melden, Quittieren und Archivieren
- · Keine Programmierung möglich
- Schnelle Erstellung einer Bedienoberfläche
- Zugriff auf unterschiedliche OPC-Server gleichzeitig möglich

#### Nachteile:

 Hohe Kosten durch die Anschaffung der WinCC- und WinCC flexible-Software



### Funktionsprinzipien und Programmstrukturen

Beitrags-ID: 23829402

#### Inhalt

Hier wird auf die detaillierten Funktionsabläufe der beteiligten Hard- und Softwarekomponenten, die Lösungsstrukturen und wo sinnvoll auf die konkrete Implementierung dieser Applikation eingegangen.

Sie benötigen diesen Teil nur, wenn Sie das Zusammenspiel der Lösungskomponenten kennen lernen wollen.

#### 3 Generelle Funktionsmechanismen

#### Hier erfahren Sie...

welche generellen Funktionsmechanismen bezüglich OPC gelten.

#### 3.1 Grundlagen zum Thema OPC

#### Was ist OPC?

OPC ist eine herstellerunabhängige Software-Schnittstelle auf Basis von COM/DCOM, die einen Datenaustausch zwischen Hard- und Software auch verschiedener Hersteller ermöglicht. Unter der OPC-Foundation, einer Interessengemeinschaft namhafter Hersteller, entstanden unter anderem diese vier OPC- Spezifikationen.

Tabelle 3-1

Spezifikation	Anwendung
OPC Data Access (DA)	Zugriff auf Prozessdaten
OPC Alarm& Events (A&E)	Schnittstelle für ereignisbasierte Information inklusive Quittung
OPC Historical Data Access (HDA)	Funktionen für archivierte Daten
OPC Data eXchange (DX)	Server- Server Querkommunikation.

Dieses Beispiel verwendet ausschließlich die "OPC Data Access" Spezifikation.

OPC ist eine Client/Server Architektur. Hersteller für Baugruppen, die Prozessdaten liefern, stellen für ihre Baugruppen einen OPC-Server zur Verfügung, der die Anbindung an die jeweilige Datenquelle übernimmt. Ein OPC-Client kontaktiert den Server und liest bzw. beschreibt die Daten.

#### Beitrags-ID: 23829402

#### **OPC DA-Schnittstelle**

**SIEMENS** 

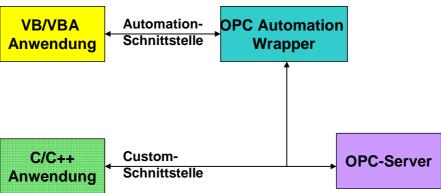
Die OPC DA-Spezifikation war der erste OPC Standard. Sie wird verwendet, um Prozessdaten zwischen Steuerungen und HMI Geräten oder anderen Clients auszutauschen. Die gelesenen Daten enthalten Datentyp, Zeitstempel und Quality, die Aussagen über die Qualität des Wertes macht. Die Clients können sowohl Prozessdaten lesen als auch Befehle an den OPC-Server schreiben. Der Server reicht die Steuerdaten dann an die Steuerung weiter.

#### **Automation- und Custom Schnittstelle**

OPC stellt zwei Schnittstellen für das Data Access zur Verfügung. Die Custom und Automation Schnittstellen ermöglichen einen Aufruf der Server Funktionen durch den Client. Die Schnittstellen gibt es für den Zugriff auf Prozessvariablen (Data Access) als auch für die Verarbeitung von Ereignissen und Alarmen (Alarms & Events).

Nachfolgendes Bild zeigt das Schema:

Abbildung 3-1





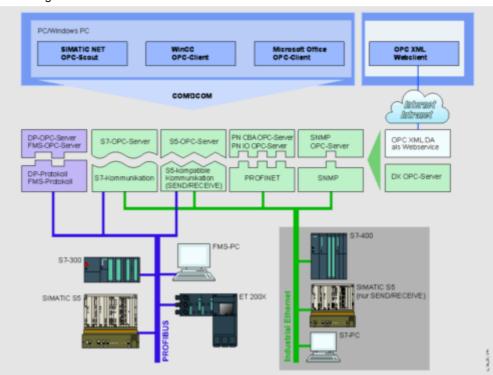
#### SIMATIC NET OPC-Server

Nachfolgende Grafik zeigt eine Struktur mit den verschiedenen SIMATIC NET OPC-Server und den zugehörigen Protokolltreiber.

Beitrags-ID: 23829402

Für diese Applikation wurde der S7-OPC-Server verwendet. Über MPI wird eine S7-Kommunikation zu einer S7-300 aufgebaut.

Abbildung 3-2



Der SIMATIC NET OPC-Server stellt folgende Zugänge zur Verfügung:

- Industrial Ethernet
  - S7-Kommunkation
  - S5-kompatible Kommunikation
  - PROFINET
  - SNMP
- PROFIBUS
  - DP-Protokoll
  - FMS-Protokoll
  - S7-Kommunikation
  - S5-kompatible Kommunikation

Copyright © Siemens AG 2006 All rights reserved 23829402\_OPC\_DATCON\_EXCEL\_DOKU\_v10\_d.doc

OPC mit Excel VBA Beitrags-ID: 23829402

#### 3.2 Multi Point Interface

Jede SIMATIC S7-CPU besitzt eine MPI-Schnittstelle (MultiPoint Interface). Sie ermöglicht den Aufbau eines Subnetzes, in dem Automatisierungssystem, Bedien- und Beobachtungsgeräte und Programmiergeräte untereinander Daten austauschen können.

Tabelle 3-2

Kriterium	Technische Daten
Anzahl der Stationen	Max. 32
Übertragungsraten	19,2 kBit/s 187,5 kBit/s 12 MBit/s
Netzausdehnung	Segmentlänge 50 m
Übertragungsmedium	Geschirmte Zweidrahtleitung

OPC:

OPC mit Excel VBA

#### 4 Funktionsmechanismen dieser Applikation

#### Hier erfahren Sie...

welche Funktionalitäten der OPC-Client anbietet, wie die OPC-Server und OPC-Client Funktionsmechanismen zusammenarbeiten und wie dies in einem VBA Projekt realisiert ist.

Beitrags-ID: 23829402

#### 4.1 Funktionsprinzipien des OCX Data Control

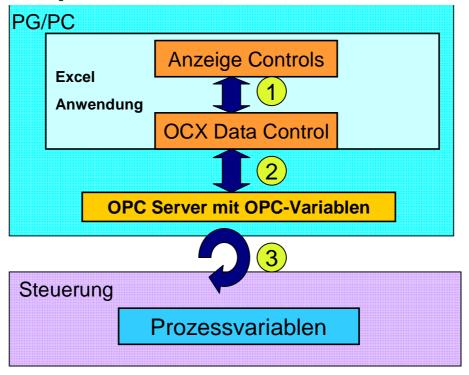
Die zentrale Komponente ist das SIMATIC NET OCX Data Control.

Ohne dieses Control kann kein anderes SIMATIC Control auf Prozessdaten zugreifen. Das OCX Data Control stellt die Verbindung zum SIMATIC NET OPC-Server her und greift auf Prozessdaten zu. Während des Programmablaufs ist dieses Control unsichtbar.

#### Schema

Nachfolgendes Bild verdeutlicht die Verbindung zwischen den Anzeige Controls und den OPC-Prozessvariablen

Abbildung 4-1





Beitrags-ID: 23829402

#### Die Nummerierungen haben folgende Bedeutung:

Tabelle 4-1

Nr.	Aktion
1.	Das OCX Data Control versorgt die Controls mit Daten. Anderseits erteilen die Controls Schreibbefehle an das OCX Data Control.
2.	Der OPC-Server versorgt das OCX Data Control mit Prozesswerten über das Data Access Interface.
3.	Der OPC-Server überwacht ständig die zugeordneten Prozessvariablen auf Änderung und nimmt Schreibaufträge auf Prozessvariablen vor.

#### Die verfügbaren Anzeige Controls von SIMATIC NET

Die Anzeige Controls sind Elemente zur Visualisierung von Prozessdaten. Sie erhalten ihre Daten über das SIMATIC NET OCX Data Control und nicht direkt durch Zugriff auf OPC oder eine andere Schnittstelle. Mit diesen Controls wird es ermöglicht, eine Bedien- und Beobachtungsoberfläche ohne Programmierung zu erstellen.

Nachfolgende Tabelle zeigt die existierenden Anzeige Controls:

Tabelle 4-2

Control	Darstellung	Kurzbeschreibung
Button	Stopp	Greift auf einzelne im OPC-Server gespeicherten Bits zu. Das Button Control kennt nur die 2 Werte 0 = Start 1 = Stopp Bei Betätigung des Buttons, ändert sich auch das zugeordnete Bit.
Slider		Greift über Variablen auf Prozessdaten im Byte, Word und Doppelwortformat zu. Wenn das Slider Control im Wert verändern wird, ändert sich auch der Wert der zugeordneten Variable
Number	5 ××	Greift über Variablen auf Prozessdaten im Byte, Word und Doppelwortformat zu. Wenn im Number Control ein neuer Wert eingegeben wird, ändert sich auch der Wert der zugeordneten Variable
Label	VANILLE	Stellt den Wert einer Variablen dar. Das Label control kann nur Werte von Prozessdaten darstellen

#### **Hinweis**

Die Anzeige Controls müssen in eine UserForm von Visual Basic eingefügt werden.



#### Methoden des OCX Data Control

Das OCX Data Control besitzt einige Methoden um Informationen aus dem OPC-Server zu holen oder Aufträge abzuwickeln. Im Folgenden werden die wichtigsten Methoden aufgeführt.

Beitrags-ID: 23829402

#### **Connect/ Disconnect**

Mit der Methode Connect bzw. Disconnect kann man sich manuell mit dem OPC-Server verbinden oder die Verbindung abbauen. Benötigt werden diese Methoden wenn das automatische Verbinden mit dem OPC-Server deaktiviert ist.

#### ValueChanged

Die Methode ValueChanged wird dann aufgerufen, wenn ein oder mehrere Ereignisse durch eine Wert- oder Qualitätsänderung der zugeordneten Variablen ausgelöst werden. Der Ereignisname und der neue Wert bzw. Quality wird im Methodenaufruf als Array übergeben.

#### Die Syntax der Methode ValueChanged ist:

DatCon1 ValueChanged(ByVal Count As Long,
ByVal UserIDs As Variant,
ByVal ItemIDs As Variant,
ByVal Values As Variant,
ByVal Qualities As Variant,
ByVal TimeStamps As Variant)

#### Die Parameter haben dabei folgende Bedeutung:

#### Tabelle 4-3

Parameter	Bedeutung	
Count	Anzahl der ausgelösten Ereignisse	
UserIDs	Ereignisnamen	
ItemIDs	ItemIDs der zugeordneten Variablen	
Values	geänderte Werte der Variablen	
Qualities	Qualitäten der Variablen	
TimeStamps	Zeitpunkt des Ereignisses	



#### ReadVariable

Die Methode ReadVariable liest den Wert einer durch die ItemID spezifizierten Prozessvariable.

Beitrags-ID: 23829402

Die Syntax der Methode ist:

result = DatCon1.ReadVariable	(ItemID as String,
	Value as Variant,
	Quality as Long,
	Timestamp as Date)

#### Die Parameter haben dabei folgende Bedeutung:

Tabelle 4-4

Parameter	Bedeutung
ItemID	ItemID der zu lesenden Variablen
Value	Werte der Variablen
Quality	Qualität der Variablen
TimeStamp	Zeitpunkt des Leseauftrages

#### ReadMultiVariable

Die Methode ReadMultiVariable liest die Werte aus mehreren durch die ItemID spezifizierten Prozessvariablen.

Die Syntax der Methode ist:

result =	DatCon1.ReadMultiVariable	(ItemIDs as String,
		Values as Variant,
		Errors as Variant
		Qualities as Long,
		Timestamps as Date)

#### Die Parameter haben dabei folgende Bedeutung:

Tabelle 4-5

Parameter	Bedeutung
ItemIDs	String-Array mit den ItemIDs der zu lesenden Variablen
Values	Variant-Array für die Werte
Errors	Variant-Array für den Fehlerstatus
Quality	Long-Array mit den Lesequalitäten
TimeStamp	Date-Array mit den Lesezeiten



#### WriteVariable

Die Methode WriteVariable schreibt einen bestimmten Wert in eine durch die ItemID spezifizierte Prozessvariable.

Beitrags-ID: 23829402

Die Syntax der Methode ist:

Die Parameter haben dabei folgende Bedeutung:

Tabelle 4-6

Parameter	Bedeutung
ItemID	ItemID der zu beschreibenden Variablen
Value	Neuer Wert

#### WriteMultiVariable

Die Methode WriteMultiVariable schreibt bestimmte Werte in mehrere durch ihre ItemID spezifizierte Prozessvariablen.

Die Syntax der Methode ist:

#### Die Parameter haben folgende Bedeutung:

Tabelle 4-7

Parameter	Bedeutung
ItemIDs	String-Array mit den ItemIDs der zu beschreibenden Prozessvariablen
Values	Variant-Array mit den neuen Werten
Errors	Variant-Array für den Fehlerstatus

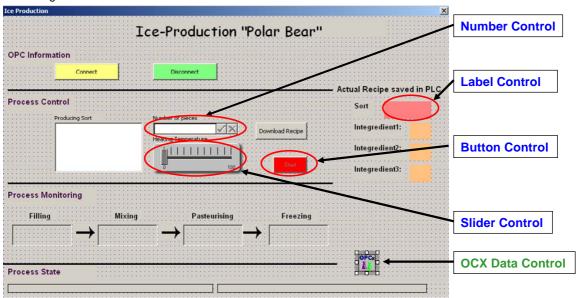


# Beitrags-ID: 23829402

#### Die verwendeten ActiveX Controls in der Bedienoberfläche

Die folgende Abbildung zeigt die Bedienoberfläche des Dialoges im Entwicklungsmodus. Sie enthält alle Anzeige Controls von SIMATIC NET und das Data Control, die farblich hervorgehoben sind.

#### Abbildung 4-2



#### Die OPC- Prozessvariablen

Prozessvariablen werden im OPC-Server durch eine OPC-Klasse vertreten. Die so genannte ItemID ist eine String-Folge und identifiziert die Prozessvariable eindeutig. Sie gibt dem Server an, welche Prozessvariable dem OPC-Item zugeordnet wird. Durch das OPC-Item kann dann auf die Prozessvariable zugegriffen werden. Die ItemID kann dabei die Prozessvariable absolut oder symbolisch identifizieren.

#### **Absolute Adressierung**

Für die ItemID mit einer absoluten Adressierung gibt es drei Möglichkeiten:

S7:[<Verbindungsname>]DB <Nr>, {<Typ>} <Adresse> {,<Anzahl>}

S7:[<Verbindungsname>]DI <Nr>, {<Typ>} <Adresse> {,<Anzahl>}

S7:[<Verbindungsname>]<Objekt> {<Typ> }<Adresse> {,<Anzahl>}

Die Bedeutung der einzelnen Adressierbereiche ist:

Copyright © Siemens AG 2006 All rights reserved 23829402\_OPC\_DATCON\_EXCEL\_DOKU\_v10\_d.doc

OPC mit Excel VBA Beitrags-ID: 23829402

Tabelle 4-8

Adressierbereiche	Bedeutung	Hinweis
S7	S7-Protokoll für den Zugriff auf Prozessvariablen	
<verbindungsname></verbindungsname>	Name der Verbindung	In NetPro projektierbar
DB	Datenbaustein; Kennzeichen für eine Variable in einem DB	
DI	Instanzdatenbaustein; Kennzeichen für eine Variable in einem DI	
<nr.></nr.>	Nummer des DBs oder DIs	
<typ></typ>	S7-Datentyp	Ein S7-Datentyp wird im OPC-Server in den entsprechenden OLE- (COM)Datentyp umgewandelt
<objekt></objekt>	S7-Datentyp	z.B. Merker M, Eingang E, Ausgang A
<adresse></adresse>	Adresse der ersten Variablen, die angesprochen werden soll.	-Byte-Offset -Byte-Offset.Bit (nur Datentyp X) -Byte-Offset.String -Länge (nur Datentyp String, String-Länge 1 Byte bis 254 Byte)
<anzahl></anzahl>	Anzahl der Variablen eines Typs, die ab dem im Parameter Adresse angegebenen Offset angesprochen werden sollen (Wertebereich 065535).	Beim Datentyp <b>X</b> Eingabe der Anzahl für Schreibzugriff nur in Vielfachen von 8 möglich. Die Bitadresse muss dann Null sein.

#### **Symbolische Adressierung**

Sind die Prozessvariablen in der Steuerung als Symbol deklariert, kann die ItemID auch symbolisch sein.

- <Stationsname>.<CPU-Name>.<DB-Name>.<Variablenname>
- <Stationsname>.<CPU-Name>.<DI-Name>.<Variablenname>
- <Stationsname>.<CPU-Name>.<Variablenname>

Die Bedeutung der Adressierbereiche ist:

Tabelle 4-9

Adressierbereiche	Bedeutung	Hinweis
<stationsname></stationsname>	Name der SIMATIC Station	Projektierbar in STEP7
<cpu-name></cpu-name>	Name der CPU	Projektierbar in STEP7
<db-name></db-name>	Symbolischer Name des Datenbausteins	Projektierbar in STEP7
<di-name></di-name>	Symbolischer Name des Instanzdatenbausteins	Projektierbar in STEP7
<variablenname></variablenname>	Symbolischer Name der Variablen	Projektierbar in STEP7



Beitrags-ID: 23829402

#### Hinweis

Damit der OPC-Client symbolische Variablenzugriffe über den OPC-Server absetzten kann, müssen die in STEP 7 projektierten Symbole in die Systemdaten für den OPC-Server aufgenommen werden. (Siehe Kap.6)

#### ItemIDs dieser Applikation

Nachfolgende Tabelle listet die ItemIDs auf, die in dieser Applikation symbolisch verwendet werden.

Tabelle 4-10

ItemID symbolisch	ItemID absolut
SIMATIC 300.CPU 313C.PROD_RECIPE.Ingredient1	S7:[OPC]DB6,INT0
SIMATIC 300.CPU 313C.PROD_RECIPE.Ingredient2	S7:[OPC]DB6,INT2
SIMATIC 300.CPU 313C.PROD_RECIPE.Ingredient3	S7:[OPC]DB6,INT4
SIMATIC 300.CPU 313C.DB_ACT_PROD.PRO_SORT	S7:[OPC]DB1,STRING0,30
SIMATIC 300.CPU 313C.DB_ACT_PROD.PRO_COUNT	S7:[OPC]DB1,INT32
SIMATIC 300.CPU 313C.DB_ACT_PROD.PRO_MAX_TEMP	S7:[OPC]DB1,INT40
SIMATIC 300.CPU 313C.DB_ACT_PROD.Start	S7:[OPC]DB1,X44.0
SIMATIC 300.CPU 313C.DB_ACT_PROD.Download	S7:[OPC]DB1,X44.1
SIMATIC 300.CPU 313C.DB_ACT_PROD.Filling	S7:[OPC]DB1,X44.2
SIMATIC 300.CPU 313C.DB_ACT_PROD.Mixing	S7:[OPC]DB1,X44.3
SIMATIC 300.CPU 313C.DB_ACT_PROD.Pasteurizing	S7:[OPC]DB1,X44.4
SIMATIC 300.CPU 313C.DB_ACT_PROD.Freezing	S7:[OPC]DB1,X44.5

#### **Hinweis**

Mehr Informationen über die OPC-Prozessvariablen finden Sie im Handbuch "Industrielle Kommunikation mit PG/PC Bd2 Schnittstellen" unter SIMATIC -> Dokumentation.

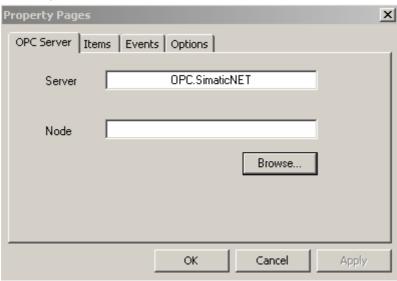
Copyright © Siemens AG 2006 All rights reserved 23829402\_OPC\_DATCON\_EXCEL\_DOKU\_v10\_d.doc

OPC mit Excel VBA Beitrags-ID: 23829402

#### 4.2 Exemplarische Projektierung des OCX Data Control

Das OCX Data Control ist der wichtigste Bestandteil des OPC-Client in dieser Applikation. Für die Konfiguration verfügt das OCX Data Control über ein Eigenschaftsfenster mit mehreren Laschen.

Abbildung 4-3



Dieser Abschnitt zeigt exemplarisch wie man eine Variable und ein Ereignis verschaltet.

Tabelle 4-11

Schritt	Handlung	Dialoglasche
1.	OPC-Server auswählen	OPC-Server
2.	Anzeige Controls mit Prozessvariablen verbinden	Items
3.	Prozessvariablen zu Ereignissen zuordnen	Events
4.	Einstellungen vornehmen	Options

#### **OPC-Server auswählen**

Das SIMATIC NET OCX Data Control kann als OPC-Client jeden beliebigen OPC-Server nutzen. Der SIMATIC NET OPC Server heißt **OPC.SimaticNET.** Der OPC-Server wird dabei vom OCX Data Control so konfiguriert, dass alle zugeordneten Variablen ständig beobachtet werden und bei Wertänderung einer Prozessvariable der neue Wert an das Data Control übertragen wird und von dort aus an das entsprechende Anzeige Control.

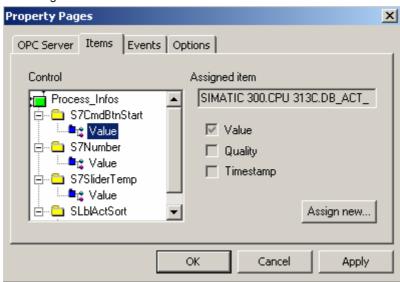


# Verknüpfung der Anzeige Controls mit Prozessvariablen

Der Zugriff auf die Prozessvariablen erfolgt über das Data Control. Die Anzeige Controls werden über das OCX Data Control mit einer Eigenschaft (Value, Quality, Zeitstempel) einer Prozessvariablen des OPC-Servers verbunden.

Beitrags-ID: 23829402

Abbildung 4-4



#### Prozessvariablen zu Ereignissen anbinden

Oft reichen die Anzeige Controls nicht aus, um eine individuellere und komplexere Oberfläche zu gestalten. Neben den Anzeige Controls besteht die Möglichkeit, Prozessvariablen mit dem OCX Data Control in Form von Ereignissen zu verbinden. Ändert sich der Zustand oder Wert der Variablen, wird das zugehörige Ereignis ausgelöst und das VB Programm kann entsprechend reagieren.

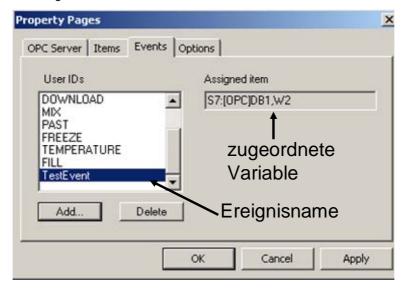
Nachfolgendes Bild zeigt ein Ereignis mit der zugeordneten Variablen. Sobald sich der Wert der verknüpften Variablen ändert, wird ein Ereignis mit dem Namen "TestEvent" ausgelöst.

# **SIEMENS**

OPC mit Excel VBA

Beitrags-ID: 23829402

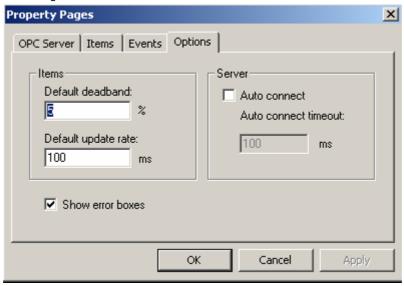
Abbildung 4-5



#### Einstellungen vornehmen

Im Abschnitt Einstellungen können Default Werte für die OPC-Verbindung gesetzt werden. Dazu gehört die Aktualisierungszeit der Variablen, die Totzeit der Verbindung aber auch das Abschalten der automatischen Verbindung mit dem OPC-Server bei Aufruf des Data Controls.

Abbildung 4-6



#### 4.3 STEP7 Beispiel-Programm

Die folgende Tabelle gibt eine Übersicht über die STEP7- Bausteine für die Simulation der Eisproduktion.



Beitrags-ID: 23829402

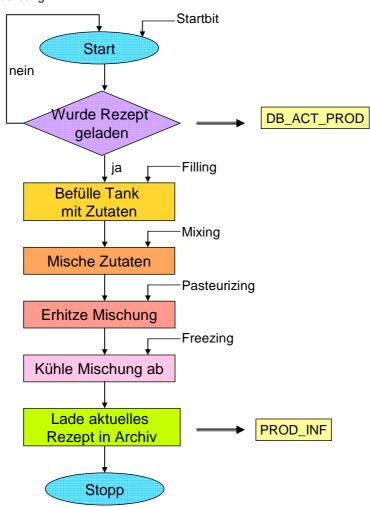
Tabelle 4-12

Baustein	Funktion
FC 1	Die Funktion "PRO_CONTROL" simuliert die Eisproduktion in einer Schrittkette.
DB 1	Der Datenbaustein "DB_ACT_PROD" enthält die aktuellen Rezeptdaten und Steuervariablen
DB 2	Der Datenbaustein "PRO_INF" enthält Produktionsinformationen des Produktionsvorganges
DB 6	Der Datenbaustein "PROD_STOCK" enthält die Istwerte des Lagerbestands

#### Der Ablauf des Simulationsprogramms

Nachfolgende Grafik zeigt das Flussdiagramm des Simulationsprogramms. Die Simulation der Produktion ist im FC1 "PRO\_CONTROL" hinterlegt, der zyklisch vom OB1 aufgerufen wird. Der FC1 ist als Schrittkette realisiert.

Abbildung 4-7





#### 5 Erläuterungen zum Beispielprogramm

#### Hier erfahren Sie...

Details aus dem Code einiger Kernprogrammteile des VBA-Programms, wie die Fehlerroutine im OPC-Client realisiert wurde und Hinweise zum Ablauf des STEP7 Programms.

Beitrags-ID: 23829402

#### 5.1 Fehlerhandling im OPC Client

Dieser Abschnitt zeigt, wie das VBA Programm auf bestimmte Fehler reagiert. Folgende Fehler werden ausgewertet:

- beim Verbindungsaufbau und -abbau
- beim Lesen von Prozessvariablen aus der Steuerung
- beim Schreiben von Werten in die Steuerung

#### Fehlerhandling in der Oberfläche:

- Die Oberfläche enthält ein Ausgabefenster für Fehlermeldungen.
- Ein Fehler im Verbindungsaufbau mit dem OPC-Server ruft die Methode ConnectionError auf.
- Die Rückgabewerte aller OPC Aufrufe werden auf Fehler überprüft.
- Im Fehlerfall wird eine Meldung im Ausgabefenster ausgegeben.
- Viele Funktionen verfügen über einen ErrorHandler, der im Fehlerfall eine Meldung ausgibt.

#### 5.2 Erläuterungen zum VBA-Beispielprogramm

In diesem Kapitel erfahren Sie mehr Details aus dem Visual Basic Code. Anhand von Codeauszügen werden folgende Kernprogrammteile näher erläutert:

- Ereignisauswertung mit dem OCX Data Control
- Lesen von Blockdaten
- Konvertierung von S7-Datentypen in VBA Typen



Beitrags-ID: 23829402

#### Methode ValueChanged

Die Methode ValueChanged des OCX Data Controls wird dann aufgerufen, wenn ein oder mehrere Ereignisse ausgelöst wurden. Die Methode merkt sich die Ereignisnamen, sodass das VB-Programm auf die jeweiligen Ereignisse reagieren kann. Nachfolgende Grafik zeigt einen Ausschnitt aus der ValueChange-Methode dieser Applikation.

#### Abbildung 5-1

```
Private Sub DatCon1 ValueChanged(ByVal Count As Long,
                                    ByVal UserIDs As Variant, _
                                    ByVal ItemIDs As Variant, _
                                    ByVal Values As Variant,
                                    ByVal Qualities As Variant,
                                    ByVal TimeStamps As Variant)
'A loop with 'count' passes
'the value of 'count' is the number of changed variables in PLC
For i = 0 To Count -1
Debug.Print (i)
Debug.Print (UserIDs(i))
Events = UserIDs(i)
Debug.Print (Events)
Select Case Events
'Changes of Actual Stock of ingredients
    Case "INGRED1"
       Tabelle3.Cells(19, 3) = Values(i)
    Case "INGRED2"
        Tabelle3.Cells(20, 3) = Values(i)
    Case "INGRED3"
        Tabelle3.Cells(21, 3) = Values(i)
'Change of State of Process: Filling
    Case "MIX"
        If (Values(i) = True) Then
            TxtBoxSTEP1.Text = "In Progress"
            TxtBoxSTEP1.BackStyle = fmBackStyleOpaque
            Txt State.Caption = " Integredients are mixed"
        Else
            TxtBoxSTEP1.Text = " "
            TxtBoxSTEP1.BackStyle = fmBackStyleTransparent
        End If
'===...===
End Select
Next i
End Sub
```



#### Lesen von größeren Datenmengen

Zur Entlastung des OPC-Servers besteht die Möglichkeit, Daten unterschiedlicher Typen en bloc aus der CPU zu lesen. Die Daten werden mit der ReadVariable-Methode in einem festgelegten Format in einem Puffer gelesen. Nachfolgende Grafik zeigt den Aufruf zum Lesen eines Datenblocks von Bytes.

#### Abbildung 5-2

```
result = Process_Infos.DatCon1.ReadVariable("S7:[OPC]DB2,B0,58", _ Archiv, _ quality, _ timestamp)
```

Beitrags-ID: 23829402

#### Datenkonvertierungsroutinen im VBA

In diesem Abschnitt zeigen wir Ihnen, wie Sie den en bloc gelesenen Byte-Datenblock in die Ursprungstypen aufsplittern können. Dabei müssen die Datenkonvertierungsvorschriften von S7-Datentypen in die entsprechenden VBA-Typen beachtet werden.

#### Bytes ←→VBA-Integer

Folgende Grafik zeigt die Konvertierung in ein VBA-Integer. Die gelesenen Bytes stellen in der CPU ein Integer dar.

#### Abbildung 5-3

35/60



Beitrags-ID: 23829402

#### Bytes ←→VBA-Zeichen

Folgende Grafik zeigt die Konvertierung in ein VBA-Zeichen. Die gelesenen Bytes stellen in der CPU ein Char dar.

#### Abbildung 5-4

#### Bytes ←→VBA-Word

Folgende Grafik zeigt die Konvertierung in ein VBA-Word. Die gelesenen Bytes stellen in der CPU ein Word dar.

#### Abbildung 5-5

#### Byte ←→VBA-Byte

Folgende Grafik zeigt die Konvertierung in ein VBA-Byte. Die gelesenen Bytes stellen in der CPU ein Byte dar.

#### Abbildung 5-6

```
Function S7ByteToByte(ByVal S7Byte As Byte) As Byte
S7ByteToByte = BCDToDec(S7Byte)
End Function
```

36/60

Copyright © Siemens AG 2006 All rights reserved 23829402\_OPC\_DATCON\_EXCEL\_DOKU\_v10\_d.doc

OPC mit Excel VBA

Beitrags-ID: 23829402

## Bytes ←→VBA-String

Folgende Grafik zeigt die Konvertierung in ein VBA-String. Die gelesenen Bytes stellen in der CPU ein String dar.

#### Abbildung 5-7

```
Function BytesToString(buf As Variant, i As Integer) As String
Dim Sort As String
Dim k
Dim lenght As Integer |
lenght = buf(i + 1)
For k = i + 2 To i + 2 + lenght
Sort = Sort & Chr((buf(k)))
Next k
BytesToString = Sort
End Function
```

### Bytes ← → VBA-Single

Folgende Grafik zeigt die Konvertierung in ein VBA-Single. Die gelesenen Bytes stellen in der CPU ein Real dar.

#### Abbildung 5-8

```
Function S7BytesToSingle(buf As Variant, i As Integer) As Single
Dim z As Single
    CopyMemory ByVal VarPtr(z) + 2, CLng(TwoByteToInt(buf(i + 1), buf(i))), 2
    CopyMemory ByVal VarPtr(z), CLng(TwoByteToInt(buf(i + 3), buf(i + 2))), 2
    S7BytesToSingle = z
End Function
```

## Bytes ←→VBA-Date

Folgende Grafik zeigt die Konvertierung in ein VBA-Date. Die gelesenen Bytes stellen in der CPU ein Date\_and\_Time dar.

#### Abbildung 5-9

Copyright © Siemens AG 2006 All rights reserved 23829402\_OPC\_DATCON\_EXCEL\_DOKU\_v10\_d.doc

OPC mit Excel VBA Beitrags-ID: 23829402

## 5.3 Erläuterungen zum STEP7 Programm

In diesem Kapitel erfahren Sie mehr Details aus dem STEP7 Programm.:

- Visualisierungsschnittstelle
- Archiv-Datenbaustein
- Lagerbestand

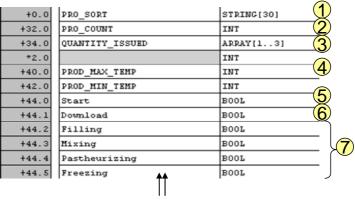
## Visualisierungsschnittstelle

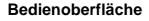
Der DB1 "DB\_ACT\_PROD" fungiert als Datenquelle für die Bedienoberfläche. Er fasst alle für das Bedienen & Beobachten notwendigen Variablen zusammen.

Nachfolgende Grafik verdeutlicht den Zusammenhang zwischen Bedienoberfläche und Datenbaustein.

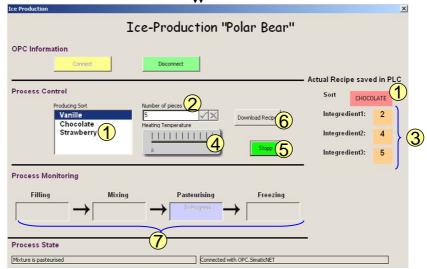
Abbildung 5-10













#### Der Archiv-Datenbaustein DB 2

Der Datenbaustein "PROD\_INF" enthält Informationen über den abgeschlossenen Produktionsvorgang wie die Rezeptdaten, die Produktionszeit und die Durchschnittstemperatur. Diese Daten können im Archiv-Bereich des Excel-Sheets "Archive" geladen und gelesen werden.

Beitrags-ID: 23829402

Nachfolgende Grafik zeigt die Struktur des DB 2

## Abbildung 5-11

+0.0	PRO_SORT	STRING[30]	1.1
+32.0	PRO_COUNT	INT	0
+34.0	QUANTITY_ISSUED	ARRAY[13]	
*2.0		INT	
+40.0	PROD_MAX_TEMP	INT	80
+42.0	PROD_MIN_TEMP	INT	-10
+44.0	AVERAGE_TEMP	REAL	0.000000e+000
+48.0	PROD_TIME	DATE_AND_TIME	DT#90-1-1-0:0:0.000
+56.0	PROD_SIGN	CHAR	1 * 1

## Modifikationsmöglichkeit

In dieser Applikation werden die Archiv-Daten im Datenbaustein mit der nächsten Produktion überschrieben. Sollen die Informationen über die Produktionsvorgänge über einen längeren Zeitraum gespeichert werden, muss das STEP7 Projekt entsprechend angepasst werden. Dabei besteht z.B. die Möglichkeit den Datenbaustein als Ringpuffer zum Speichern der Produktionsinformationen zu programmieren.

#### **Der Lagerbestand**

Der Datenbaustein "PROD\_STOCK" verwaltet den Lagerbestand für die einzelnen Zutaten.

Nachfolgende Grafik zeigt die Struktur des DB 6

#### Abbildung 5-12

+0.0	Integredientl	INT	100
+2.0	Integredient2	INT	100
+4.0	Integredient3	INT	100



# Aufbau, Projektierung und Bedienung der Applikation

Beitrags-ID: 23829402

#### Inhalt

Dieser Teil führt Sie Schritt für Schritt durch den Aufbau, wichtige Projektierungsschritte, Inbetriebnahme und Bedienung der Applikation.

## 6 Installation und Inbetriebnahme

#### Hier erfahren Sie...

welche Hard- und Software Sie installieren müssen und welche Schritte zur Inbetriebnahme des Beispiels notwendig sind.

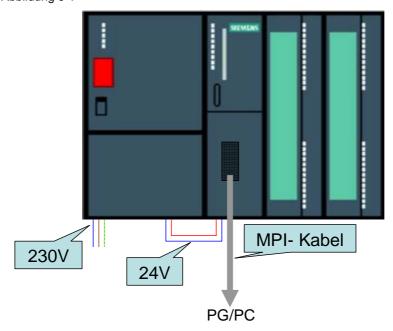
### 6.1 Installation der Hard- und Software

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

Bauen Sie die SIMATIC Hardwarekomponenten laut folgendem Bild und der Stückliste in Kapitel 2.3 auf.

Abbildung 6-1





Hinweis

Die Aufbaurichtlinien für die Installation der einzelnen Komponenten sind generell zu beachten.

#### Installation der Standard Software

Tabelle 6-1

Nr.	Aktion	Anmerkung
5.	Installieren Sie STEP 7 V5.4	Beachten Sie die Anweisungen des Installationsprogramms
6.	Installieren Sie die SIMATIC NET SOFTWARE EDITION 2005	Beachten Sie die Anweisungen des Installationsprogramms

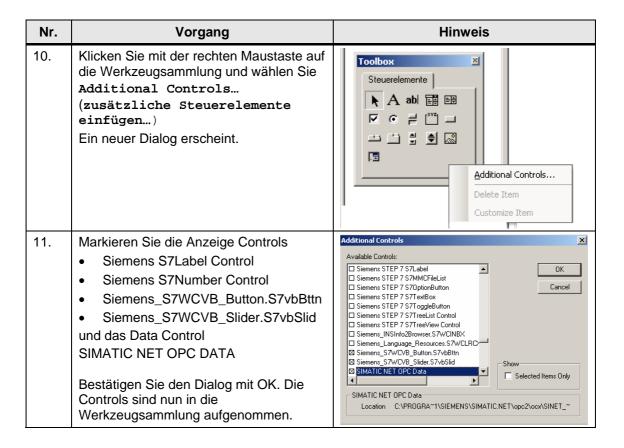
## 6.2 Projektierung des OPC Clients

Die folgende Anleitung zum Einbinden der S7Controls in VBA und der Projektierung des OCX Data Control dient nur dem Verständnis. Diese Projektierungen sind in der Beispielapplikation bereits enthalten.

## Einbinden der S7Controls in VBA

#### Tabelle 6-2

Nr.	Vorgang	Hinweis
7.	Öffnen Sie die Excel-Arbeitsmappe der Beispielapplikation	
8.	Öffnen Sie den VBA-Editor in der Arbeitsmappe.	Unter Extras -> Macros -> Visual Basic Editor
9.	Markieren Sie das VBA Projekt. Klicken Sie mit der rechten Maustaste und fügen Sie unter Insert -> UserForm (Einfügen -> UserForm) eine UserForm ein.	Microsoft Visual Basic - Mappe2.xls - [DieseArbeitsmappe (Code]]  File Edit View Insert Format Debug Run Iools Add-Ir  Project - VBAProject  VBAProjec





## **Projektierung des OCX Data Control**

Tabelle 6-3

Nr.	Vorgang	Hinweis	
1.	Markieren Sie das Symbol für das OCX Data Control in ihrer UserForm. Öffnen Sie mit der rechten Maustaste die Eigenschaften.	OPCx 1	
2.	Markieren Sie das Feld Custom (Benutzerdefiniert). Der Button [] wird sichtbar. Betätigen Sie den Button.	Properties - DatCon1  Alphabetic   Categorized    (About)    (Custom)    (Name)   DatCon1    Activated   False    AutoConnect   False    AutoConnectTimeou   100    DefaultDeadBand   5    DefaultUpdateRate   100    Height   24    Left   582    NodeName   ServerName   OPC.SimaticNET    ShowErrorBoxes   True    Tag    Top   294    Width   24	
3.	Die Eigenschaftsseite des OCX Data Control erscheint. Hier können Sie den OPC Server konfigurieren, Items mit Anzeige-Controls verschalten und Events anlegen.	Property Pages  OPC Server Items   Events   Options   Server OPC SimaticNET  Node  Browse  OK Cancel Apply	

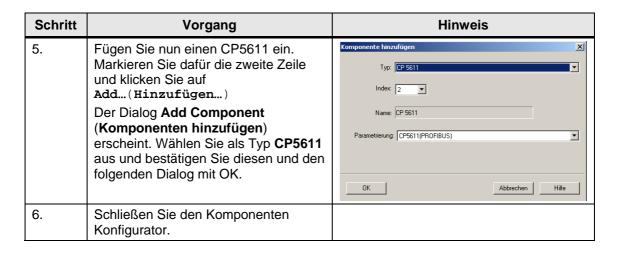


# 6.3 Konfiguration des Komponeten Konfigurators

Tabelle 6-4

Schritt	Vorgang	Hinweis
1.	Öffnen Sie den Komponenten Konfigurator	Start -> Station Configurator (Start -> Komponenten Konfigurator)
2.	Geben Sie der PC Station unter Station Name(Stationsname) denselben Namen wie im STEP7 Programm. In diesem Beispiel: SIMATIC PC-Station(1)	Station Configuration Editor - [OFFLINE]  Components Diagnostics Configuration Info  Station: SIMATIC PC-Station(1)
3.	Fügen Sie nun einen OPC-Server ein. Markieren Sie dafür die erste Zeile und klicken Sie auf Add(Hinzufügen) Der Dialog Add Component (Komponenten hinzufügen) erscheint. Wählen Sie als Typ OPC Server aus und bestätigen Sie diesen und den folgenden Dialog mit OK.	Station Configuration Editor - [OFFLINE]  Components Diagnostics Configuration Info  Station: SIMATIC PC-Station(1)
4.	Der OPC-Server ist nun im Komponenten Konfigurator integriert.	

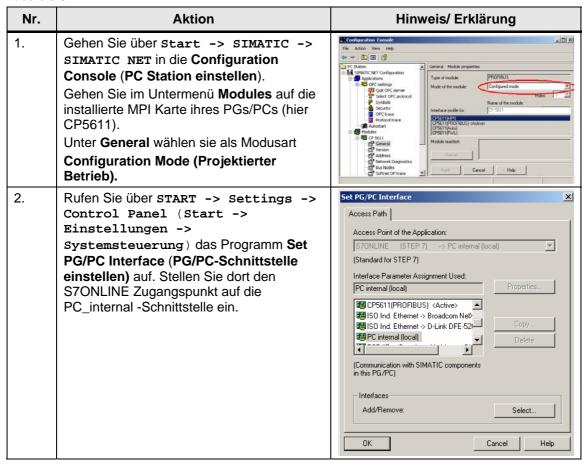




## 6.4 Konfiguration des Engineering PG/PC

#### Einstellen der PG/PC-Schnittstelle

Tabelle 6-5





## 6.5 Inbetriebnahme der SIMATIC PC-Station

Entpacken Sie die Datei 23829402\_OPC\_DATCON\_EXCEL\_CODE\_v10.zip in ein Verzeichnis ihrer Wahl.

Beitrags-ID: 23829402

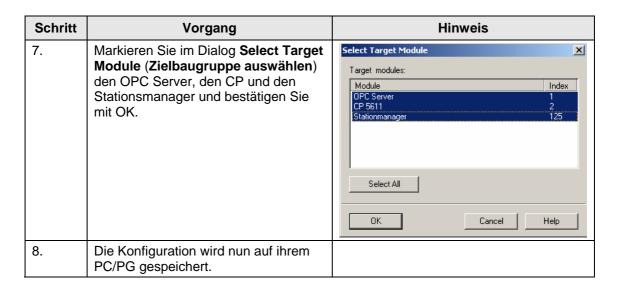
In diesem File befindet sich

- das STEP7 Projekt opc\_data.zip
- die Excel Datei IceProduktion.xls

Tabelle 6-6

Schritt	Vorgang	Hinweis
1.	Starten Sie den SIMATIC MANAGER	
2.	Dearchivieren Sie das Projekt opc_data.zip	<pre>UnterFile -&gt; Retrieve (Datei -&gt; Dearchivieren)</pre>
3.	Wechseln Sie in die Konfiguration der SIMATIC PC-Station	OPC_DataControl_Excel SIMATIC 300 CPU 313C S7 Program(2) Sources Blocks SIMATIC PC-Station(1)
4.	Markieren Sie den OPC-Server und öffnen Sie durch Doppelklick die Eigenschaften.	Index         Baugruppe         Bestellnu         Fi         M           1         OPC Server         V6.3           2         FCP 5611         6GK1 561-1A/V6.0.83           3         FCP 5611
5.	Wechseln Sie in die Lasche S7 und aktivieren Sie All bei den User Symbols. Bestätigen Sie den Dialog mit OK.  Damit wird die ganze Symbolik in die Systemdaten des OPC-Servers übernommen.	Properties - OPC Server  SNMP
6.	Übersetzen Sie Ihre Konfiguration und laden Sie sie auf Ihren PC/PG.	





## 6.6 Inbetriebnahme der SIMATIC S7-Station

Dieses Kapitel beschreibt die Inbetriebnahme der SIMATIC S7-Station.

#### **Hinweis**

Das mit diesem Beispiel ausgelieferte STEP 7 Projekt enthält die vollständig konfigurierte SIMATIC S7-Station mit Steuerungsprogramm. Dieses Projekt kann nur dann ohne Anpassung verwendet werden, wenn die Hardware identisch mit der Projektierung ist.

Tabelle 6-7

N	۱r.	Aktion	Anmerku	ung
	1.	Markieren Sie die Station SIMATIC 300 und laden Sie die Station auf die Steuerung.	OPC_DataControl_Excel  SIMATIC 300  CPU 313C  Fig. S7 Program(2)  SIMATIC PC-Station(1)	Object name  Hardware CPU 313C



## 7 Konfiguration und Projektierung

#### Hier erfahren Sie...

welche Konfigurationsschritte notwendig sind, wenn die Hardware von der im Beispiel verwendeten Hardware abweicht.

Beitrags-ID: 23829402

## 7.1 Projektierung einer neuen CPU in der SIMATIC S7-Station

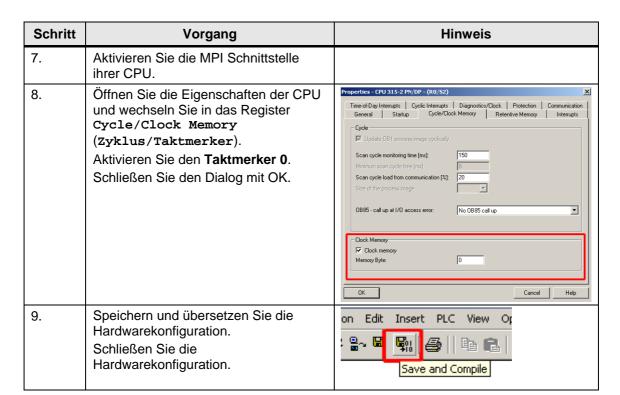
Die Konfiguration und Projektierung der SIMATIC S7-Station wird mittels STEP 7 vorgenommen und schrittweise am Beispiel einer CPU315-2PN beschrieben.

## Einbinden einer anderen CPU

Tabelle 7-1

Schritt	Vorgang	Hinweis
1.	Starten Sie den SIMATIC MANAGER	
2.	Dearchivieren Sie das Projekt opc_data.zip	<pre>UnterFile -&gt; Retrieve (Datei -&gt; Dearchivieren)</pre>
3.	Öffnen Sie die Hardwarekonfiguration durch Doppelklicken auf Hardware im Ordner SIMATIC 300	OPC_DataControl_Excel  SIMATIC 300 SIMATIC PC-Station(1)
4.	Markieren Sie in der Hardwarekonfiguration die CPU313C und löschen Sie aus dem Projekt. Bestätigen Sie den Löschvorgang im folgenden Dialog mit Yes (Ja).	1 PS 307 5A  2 CPU 313C  2.2 DI24/D016  2.3 AI5/A02  2.4 Count
5.	Löschen Sie das zugehörige S7- Programm <b>nicht</b> ! Verlassen Sie den Dialog mit No (Nein)!	Delete (13:192)  Do you also want to delete moduleCPU 313C which belongs to program OPC_DataControl_Excel\SIMATIC 300\CPU 313C\S7 Program(2)?  Yes No Help
6.	Wählen Sie ihre neue CPU im Hardwarekatalog und ziehen Sie sie mit Drag&Drop auf Steckplatz 2.	Purple Code   Substitute   Su

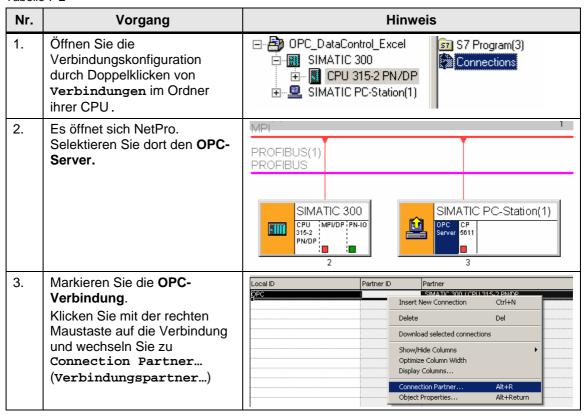
OPC mit Excel VBA



Beitrags-ID: 23829402

## Verbindungsprojektierung

Tabelle 7-2

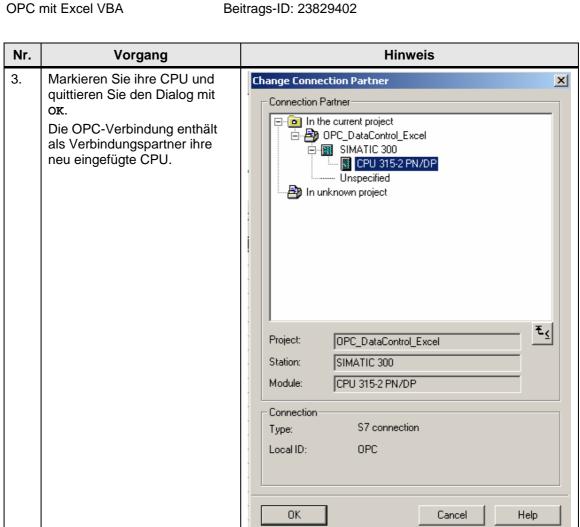


4.

Speichern und übersetzen Sie die NetPro Konfiguration. Schließen Sie NetPro.

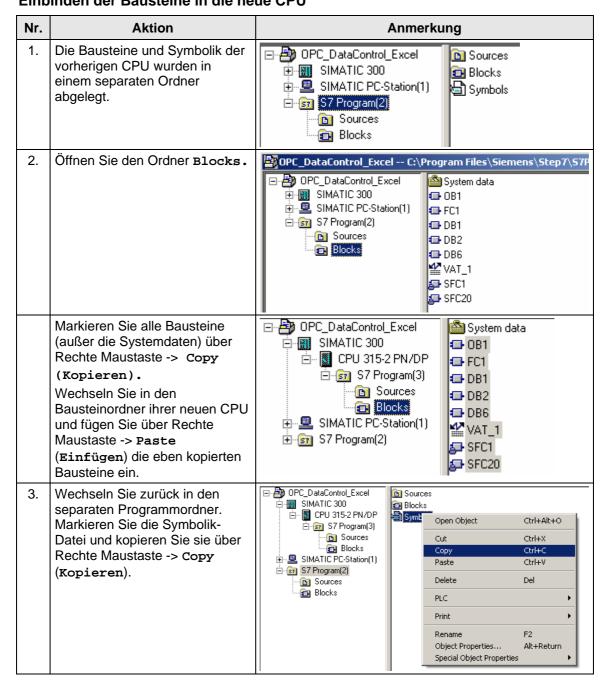
# **SIEMENS**

OPC mit Excel VBA





## Einbinden der Bausteine in die neue CPU



Beitrags-ID: 23829402

Beitrags-ID: 23829402

Fügen Sie die Symbolik-Datei in □ ⊕ OPC\_DataControl\_Excel 🛅 Sources Blocks ☐-- SIMATIC 300 den Programmordner ihre neuen Ē-- CPU 315-2 PN/DP Open Object Ctrl+Alt+O CPU über Rechte Maustaste -> - 57 Program(3) **6** Sources Ctrl+X Paste (Einfügen) ein. 🛅 Blocks Ctrl+C Сору SIMATIC PC-Station(1) Ctrl+V Paste ⊕ ggr S7 Program(2) Sources Blocks Insert New Object F2 Object Properties... Alt+Return Special Object Properties 5. Löschen Sie den separaten Programmordner der vorherigen CPŪ. 6. Markieren Sie die Station □ ♣ OPC\_DataControl\_Excel 👊 Hardware SIMATIC 300 und laden Sie die Ė-∰ SIMATIC 300 CPU 315-2 PN/DP Station auf die Steuerung. ± -- SIMATIC PC-Station(1) Markieren Sie die SIMATIC PC-7. Station und laden diese auf ihren PC.

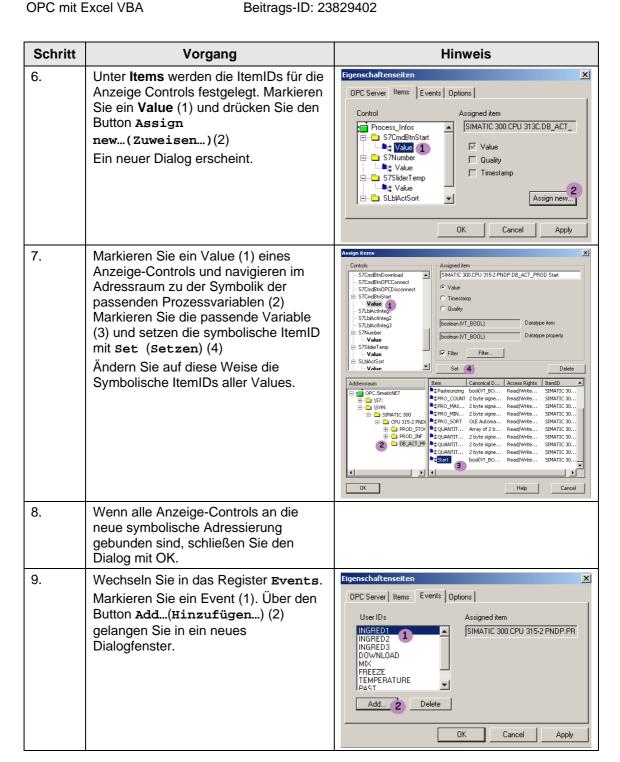
## 7.2 Änderung der ItemIDs im VBA-Projekt

Die symbolische Adressierung der ItemID enthält als Adressbereich den CPU-Namen. Wird dieser CPU-Name geändert, müssen auch die ItemIDs entsprechend angepasst werden. Die Änderung erfolgt im VBA-Editor.

Tabelle 7-3

Schritt	Vorgang	Hinweis
1.	Öffnen Sie die Excel-Arbeitsmappe der Beispielapplikation	
2.	Öffnen Sie den VBA-Editor in der Arbeitsmappe.	Unter Extras -> Macros -> Visual Basic Editor
3.	Markieren Sie das Symbol für das OCX Data Control in ihrer UserForm. Öffnen Sie mit der rechten Maustaste die Eigenschaften.	OPCX D
4.	Markieren Sie das Feld Custom	Properties - DatCon1
	(Benutzerdefiniert). Der Button	DatCon1 DatCon
	[] wird sichtbar.  Betätigen Sie den Button.	Alphabetic   Categorized
	Betaugen die den Batton.	(About)
		(Custom)
		(Name) DatCon1
		Activated False
		AutoConnect False
		AutoConnectTimeou 100
		DefaultDeadBand 5
		DefaultUpdateRate 100
		Height 24
		Left 582
		NodeName ODG 5: 1: NET
		ServerName OPC.SimaticNET
		ShowErrorBoxes True
		Tag Top 294
		Width 24
5.	Die Eigenschaftsseite des OCX Data	Property Pages X
	Control erscheint.	OPC Server Items   Events   Options
	Wechseln Sie in das Register Items	Server OPC.SimaticNET
	, and the second	Server   OF C. Simalicity
		Node
		Browse
		DIOWSS
		OK Cancel Apply

OPC mit Excel VBA



Schritt	Vorgang	Hinweis
10.	Markieren Sie einen Eventnamen (1) und navigieren im Adressraum zu der Symbolik der passenden Prozessvariablen (2) Markieren Sie die Variable (3) und ändern Sie die symbolische ItemID mit Change (Ändern) (4) Ändern Sie auf diese Weise die Symbolische ItemIDs aller Ereignisse.	Addressaum    Addressaum
11.	Wenn alle Events an die neue symbolische Adressierung gebunden sind, schließen Sie den Dialog mit OK.	
12.	Schließen Sie das Eigenschaftsfenster des OCX Data Control.	Alle symbolischen ItemIDs wurden der neuen CPU angepasst.
13.	Speichern Sie das VBA Projekt.	



# 8 Bedienung der Applikation

## Hier erfahren Sie...

wie Sie alle Funktionen dieser Applikation bedienen können.

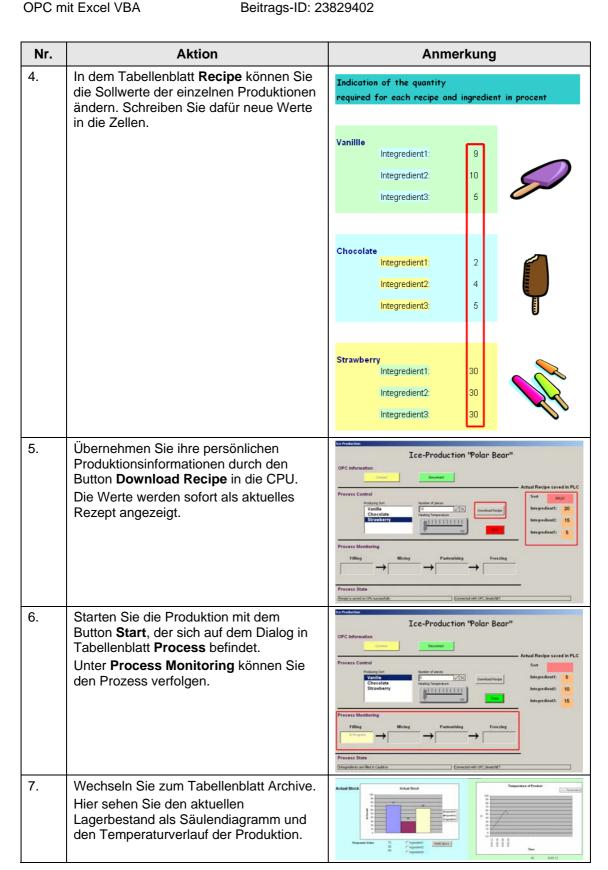
Beitrags-ID: 23829402

## Bedienung der Eisproduktion

Tabelle 8-1

Nr.	Aktion	Anmerkung
1.	Öffnen Sie die Excel Datei IceProduktion.xIs. Der Dialog Ice Produktion erscheint.	Tice-Production "Polar Bear"  OPC Information  Correct  Process Central  Actual Recipe saved in PLC  Set  Variety  Checking  Checking  Free Checking
1.	Mit dem Button Connect verbinden Sie sich mit dem OPC-Server.  Mit dem Button Disconnect bauen Sie die Verbindung ab.	Ce-Production "Polar Bear"  OPC Information
2.	Wählen Sie aus dem Auswahlfeld eine Sorte aus, die produziert werden soll.	Tice-Production "Polar Bear"  OPC Information  Correct  Process Control  Process Control  Set  Set  Indicate of pure  Process Manifering  Free Manifering  Free State  Process State
3.	Geben Sie eine Stückzahl ein und verändert sie die maximale Produktionstemperatur durch Verschieben des Sliders.	Tice-Production "Polar Bear"  OPC Information  Correct  Process Central  Actual Recipe saved in PLC  Set  Markor of pure  Check like  Check like  Check like  Process Manitoring  Filling  Mining  Pasteuthing  Freezing  Freezing  Freezing

OPC mit Excel VBA



Nr.	Aktion	Aı	nmerkung	
8.	Im Produktionsarchiv können Sie sich durch Betätigen des Buttons das	Production1		
	Produktionsrezept herunterladen	Desciption	Value	S7-Data Type
	'	Sign of Production	*	char
		Count:	11	int
		Integredient1	7	word
		Integredient2	5	word
		Integredient3	4	word
		Max. Temperature	55	int
		Average Temperature	32,50	real
		Date of Production:	17.8.06 14:34	date_and_time
		Production:	CHOCOLATE	string

DPC mit Excel VBA Beitrags-ID: 23829402

# **Anhang und Literaturhinweise**

## 9 Literaturhinweise

## 9.1 Literaturangaben

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

Abbildung 9-1

Nr.	Themengebiet	Titel
/1/	STEP7	Automatisieren mit STEP7 in AWL und SCL
		Hans Berger
		Publicis MCD Verlag
		ISBN 3-89578-113-4

## 9.2 Internet-Link-Angaben

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

Nr.	Themengebiet	Titel
1	Referenz auf den Beitrag	http://support.automation.siemens.com /WW/view/de/23829402
2	Siemens A&D Customer Support	http://www.ad.siemens.de/support
3	OPC Custom Interface, Interfacebeschreibung Auf der Dokumentations- CD-Rom von SIMATICNET Oder auf der OPC Foundation Webseite zum download www.opcfoundation.org	OPC Data Access 3.0 Spezifikation
4	Beschreibung bzw. Informationen zu:  • Allgemeine Informationen zu OPC  • OPC-Server nutzen Wird von SIMATIC NET installiert, siehe: Start → SIMATIC → Dokumentation → Deutsch Im Produktsupport unter der Beitrags ID: 2044387	SIMATIC NET – Industrielle Kommunikation mit PG/PC
5	Handbuch zum Einstellen der PC-Station Wird von SIMATIC NET installiert, siehe: Start → SIMATIC → Dokumentation → Deutsch Im Produktsupport unter der Beitrags ID. 13542666	SIMATIC NET -PC-Stationen in Betrieb nehmen - Anleitung und Schnelleinstieg
6	Modifizierte Installationsanleitungen für CP 5411, CP 5511 und CP 5611 Im Produktsupport unter der Beitrags ID: 284796	Modifizierte Installationsanleitungen für CP 5411, CP 5511 und CP 5611
7	Handbuch / Betriebsanleitung CP5611 Im Produktsupport unter der Beitrags ID: 13654902 zu finden.	SIMATIC NET CP 5611 Installationsanleitung/Produktinformation

OPC mit Excel VBA Beitrags-ID: 23829402

# 9.3 Verweise auf weitere OPC-Applikationen

Nr.	Themengebiet	Beschreibung
1	Applikation: OPC-Kommunikation über das SEND/RECEIVE-Protokoll mit einem Visual Basic .NET OPC-Client Im Produktsupport unter der Beitrags ID: 21523291 zu finden.	OPC-Kommunikation über das SEND/RECEIVE-Protokoll mit einem Visual Basic .NET OPC-Client
2	Applikation: Datenaustausch zwischen verschiedenen Bussystemen mit OPC Data eXchange (OPC DX) Im Produktsupport unter der Beitrags ID: 21523569 zu finden.	Datenaustausch zwischen verschiedenen Bussystemen mit OPC Data eXchange (OPC DX)
3	Applikation: PC-basierte Automatisierung: Anbindung von Datenbanken über offene Schnittstellen mittels OPC-XML, programmiert in C# .net Im Produktsupport unter der Beitrags ID: 21576581 zu finden.	PC-basierte Automatisierung: Anbindung von Datenbanken über offene Schnittstellen mittels OPC- XML, programmiert in C# .net
4	Applikation: Nutzung der XML-DA Schnittstelle des SIMATIC NET OPC Servers mit Visual Basic .NET Im Produktsupport unter der Beitrags ID: 21402169 zu finden.	Nutzung der XML-DA Schnittstelle des SIMATIC NET OPC Servers mit Visual Basic .NET
5	Applikation: Externe Bausteinverwaltung mit den S7-Bausteindiensten des SIMATIC NET OPC Servers Im Produktsupport unter der Beitrags ID: 21495347 zu finden.	Externe Bausteinverwaltung mit den S7-Bausteindiensten des SIMATIC NET OPC Servers
6	Applikation: Massendatenerfassung mit einem OPC-Client in C# auf Basis von .NET Im Produktsupport unter der Beitrags ID: 21447513 zu finden.	Massendatenerfassung mit einem OPC-Client in C# auf Basis von .NET
7	Applikation: Individuelle Visualisierung mit OPC auf Basis von Microsoft® .NET und der Programmiersprache C# Im Produktsupport unter der Beitrags ID: 21043779 zu finden.	Individuelle Visualisierung mit OPC auf Basis von Microsoft® .NET und der Programmiersprache C#
8	Applikation: Steuern mit dem PC - Der SIMATIC NET OPC Inproc Server als DP Master Im Produktsupport unter der Beitrags ID: 21045282 zu finden.	Steuern mit dem PC - Der SIMATIC NET OPC Inproc Server als DP Master
9	Applikation: Dezentralisieren mit dem PC - Der SIMATIC NET OPC Server als DP Slave Im Produktsupport unter der Beitrags ID: 21040390 zu finden.	Dezentralisieren mit dem PC - Der SIMATIC NET OPC Server als DP Slave