

**SIEMENS**

*Ingenuity for life*

*Industry Online Support*

*Home*

# S7-Anwenderbaustein für den OPC UA-Client einer SIMATIC S7-1500

SIMATIC S7-1500 / FW V2.6 / OPC UA-Client-Baustein

<https://support.industry.siemens.com/cs/ww/de/view/109762770>

Siemens  
Industry  
Online  
Support



# 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

<b>Rechtliche Hinweise .....</b>	<b>2</b>
<b>1 Einführung.....</b>	<b>4</b>
1.1 Überblick.....	4
1.2 Funktionsweise.....	5
1.3 Verwendete Komponenten.....	8
<b>2 Engineering .....</b>	<b>9</b>
2.1 Bausteinbeschreibung.....	9
2.1.1 Schnittstellenbeschreibung .....	9
2.1.2 Funktionsbeschreibung .....	11
2.1.3 Funktionsablauf .....	13
2.1.4 Bausteindiagnose.....	17
2.2 Projektierung und Programmierung .....	19
2.2.1 OPC UA-Client aktivieren.....	19
2.2.2 OPC UA-Client-Schnittstelle erstellen.....	20
2.2.3 OPC UA-Client-Baustein parametrieren .....	24
2.3 Inbetriebnahme und Bedienung des Beispiels.....	27
2.3.1 Inbetriebnahme des Beispiels .....	27
2.3.2 Beschreibung der Oberfläche .....	27
2.3.3 Bedienung des Beispiels .....	28
<b>3 Wissenswertes.....</b>	<b>32</b>
3.1 Verbindungsüberwachung im Baustein "OpcUaClient" .....	32
3.2 OPC UA-Server im Beispiel .....	35
3.3 Mengengerüst und Lizenzen des OPC UA-Clients.....	36
3.4 Tipps & Tricks.....	37
<b>4 Anhang.....</b>	<b>39</b>
4.1 Service und Support.....	39
4.2 Links und Literatur .....	40
4.3 Änderungsdokumentation .....	40

# 1 Einführung

## 1.1 Überblick

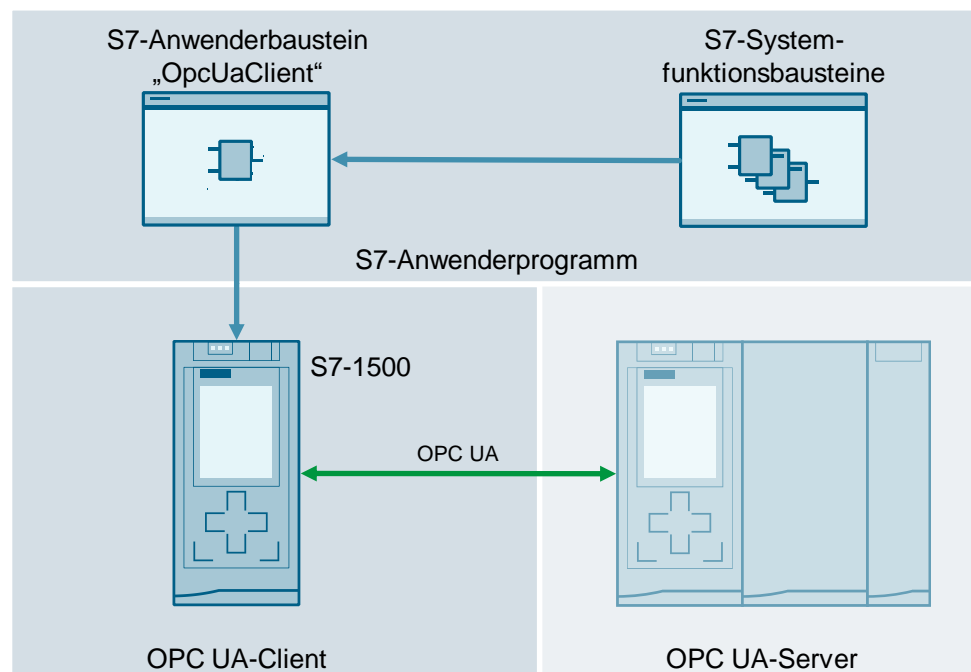
Ab der Firmware V2.6 der SIMATIC S7-1500 PLC-Familie hält ein OPC UA-Client Einzug in das Steuerungssystem. Mit dieser Erweiterung können Sie eine M2M-Kommunikation komplett über OPC UA auf einem SIMATIC S7-1500 Controller realisieren.

Im Gegensatz zum OPC UA-Server der Steuerung, wird der OPC UA-Client über eine Vielzahl von Systemfunktionsbausteinen programmiert und nicht nur projiziert. Der Client unterstützt alle gängigen Security-Policies und -Modes sowie die Benutzerauthentifizierung über Benutzername und Passwort oder anonym.

Mit diesem Anwendungsbeispiel erstellen wir für Sie den S7-Anwenderbaustein "OpcUaClient", der die wichtigsten Funktionen der OPC UA-Systemfunktionsbausteine zusammenfasst, die Implementierung für Sie beschleunigt und die Programmierung vereinfacht.

Als OPC UA-Server im Beispiel dient eine S7-1500 Steuerung mit einem einfachen Simulationsprogramm für Prozesswerte. Alternativ können Sie jeden OPC UA-Server verwenden.

Abbildung 1-1



Die folgenden Zugriffsarten werden vom OPC UA-Client und dem Anwenderbaustein unterstützt:

- Registered Read
- Registered Write
- Registered Method Call

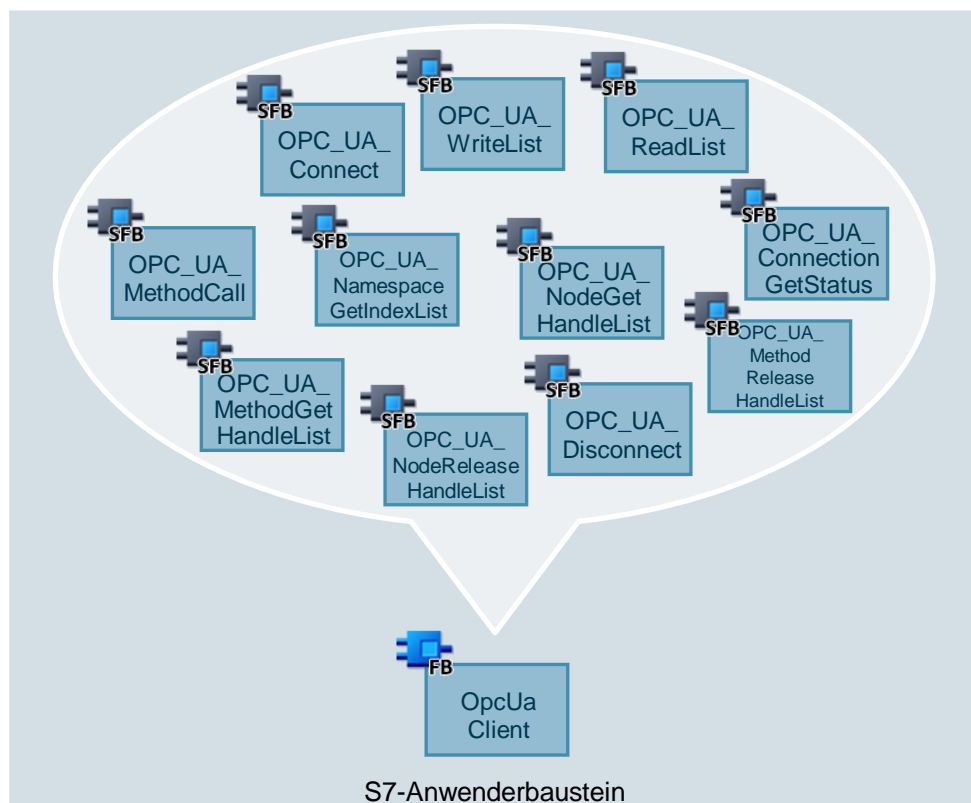


## 1.2 Funktionsweise

Um die OPC UA-Client-Funktionalitäten zu realisieren sind im S7-Anwenderbaustein "OpcUaClient" die folgenden Systemfunktionsbausteine für Sie aufgerufen, parametrisiert und ausgewertet:

- Verbindungsaufbau
  - OPC\_UA\_Connect
  - OPC\_UA\_NamespaceGetIndexList
  - OPC\_UA\_NodeGetHandleList
  - OPC\_UA\_MethodGetHandleList
- Lesen, schreiben, Methoden aufrufen
  - OPC\_UA\_ReadList
  - OPC\_UA\_WriteList
  - OPC\_UA\_MethodCall
- Verbindungsabbau
  - OPC\_UA\_NodeReleaseHandleList
  - OPC\_UA\_MethodReleaseHandleList
  - OPC\_UA\_Disconnect
- Diagnose
  - OPC\_UA\_ConnectionGetStatus

Abbildung 1-2



Die folgende Tabelle gibt Ihnen einen Überblick über die Funktionen der einzelnen SFBs innerhalb des S7-Anwenderbausteins "OpcUaClient":

Tabelle 1-1

<b>SFB</b>	<b>Funktion</b>
OPC-UA_Connect	Baut eine Verbindung und Session zu einem OPC UA-Server auf
OPC-UA_Disconnect	Baut eine Verbindung und Session zu einem OPC UA-Server ab
OPC-UA_Namespace GetIndexList	Erfragt die aktuellen Indizes der Namespaces des verbundenen OPC UA-Servers
OPC-UA_Node GetHandleList	Registriert die zu lesenden oder schreibenden OPC UA-Node-IDs auf einem OPC UA-Server
OPC-UA_Method GetHandleList	Registriert die aufzurufenden OPC UA-Methoden-Node-IDs auf einem OPC UA-Server
OPC-UA_ReadList	Liest die registrierten Variablen eines OPC UA-Servers
OPC-UA_WriteList	Schreibt die registrierten Variablen eines OPC UA-Servers
OPC-UA_MethodCall	Ruft eine registrierte Methode eines OPC UA-Servers auf
OPC-UA_Node ReleaseHandleList	Gibt die registrierten Node-IDs auf einem OPC UA-Server wieder frei
OPC-UA_Method ReleaseHandleList	Gibt die registrierten Methoden-Node-IDs auf einem OPC UA-Server wieder frei
OPC-UA_Connection GetStatus	Gibt die Qualität und den Status der Verbindung zu einem OPC UA-Server zurück

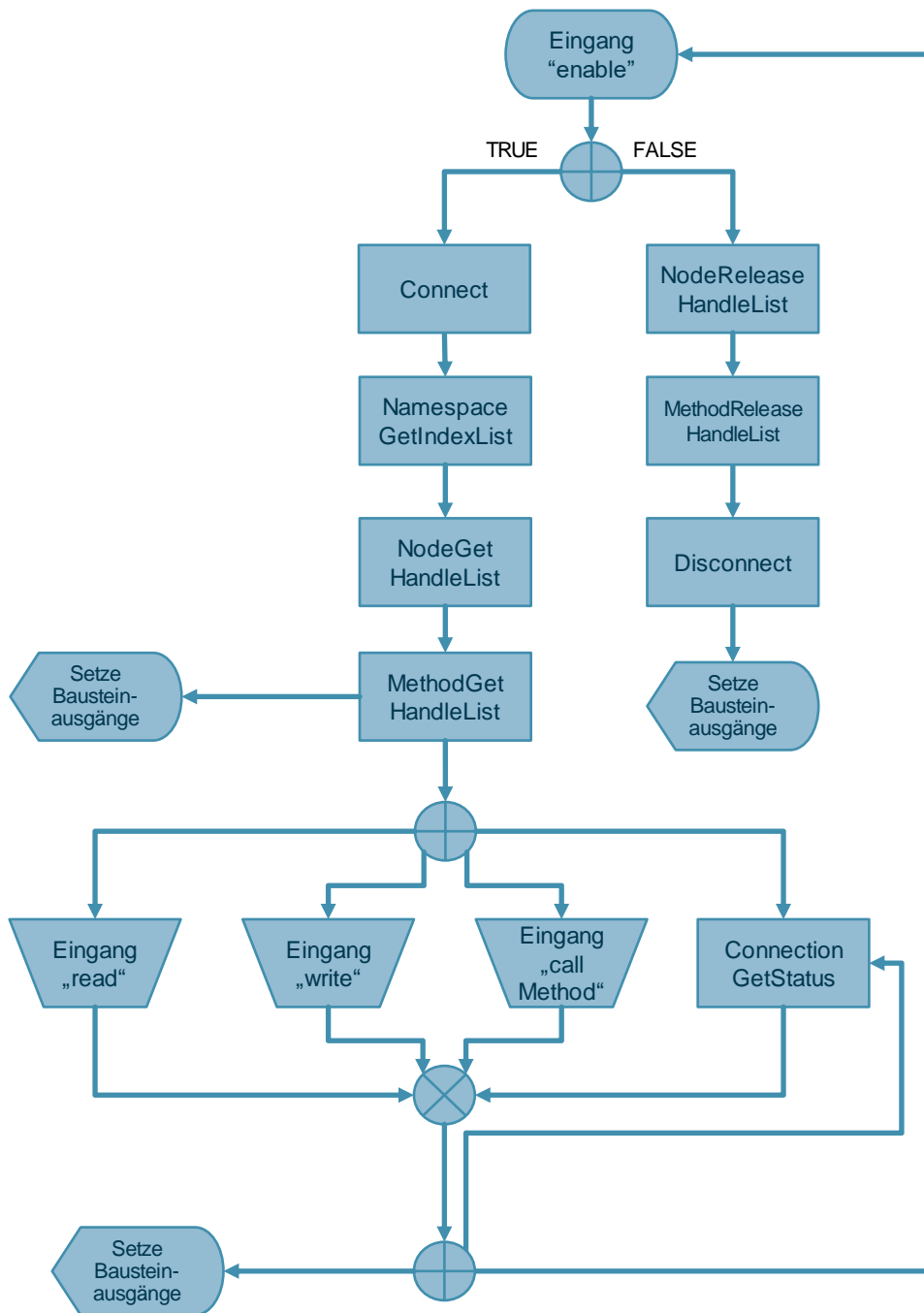
**Funktionsablauf**

Nachdem Sie den Baustein parametriert und aufgerufen haben, benötigen Sie zum Steuern der OPC UA-Funktionen lediglich vier Eingangsparameter: "enable", "read", "write" und "callMethod".

Über den Eingangsparameter "connect" bauen Sie eine Verbindung/Session auf und ab. Der Eingang "read" startet einen Leseauftrag, der Eingang "write" einen Schreibauftrag. Über "callMethod" rufen Sie eine ausgewählte Methode auf dem verbundenen OPC UA-Server auf.

Der Baustein diagnostiziert und hält für Sie die Verbindung zum Server und verbindet sich automatisch neu, wenn es zu einem Verbindungsabbruch kommt. Bei Fehlverhalten werden Sie über die Bausteinausgänge informiert.

Abbildung 1-3



## 1.3 Verwendete Komponenten

Dieses Anwendungsbeispiel wurde mit den folgenden Hard- und Softwarekomponenten erstellt:

Tabelle 1-2

Komponente	Anzahl	Artikelnummer	Hinweis
S7-1500 CPU 1513-1 PN/DP	1	6ES7513-1AL01-0AB0	Client: FW 2.6 für TIA Portal V15.1 FW 2.8 für TIA Portal V16
S7-1500 CPU 1516F-3 PN/DP	1	6ES7 516-3FN01-0AB0	Server: Ab FW 2.0
STEP 7 Professional V15.1	1	6ES7822-1AA05-0YA5	TIA Portal V15.1
STEP 7 Professional V16	1	6ES7822-1AA06-0YA5	TIA Portal V16

Dieses Anwendungsbeispiel besteht aus den folgenden Komponenten:

Tabelle 1-3

Komponente	Dateiname	Hinweis
Dokumentation	109762770_OPC_UA_PLC-Client_DOC_V1_2_1_de.pdf	Dieses Dokument
Beispielprojekt	109762770_OPC_UA_PLC-Client_15_PROJ_V1_2.zip	Dieses ZIP-Archiv enthält das Beispielprojekt für TIA Portal V15.1 und FW 2.6
Beispielprojekt	109762770_OPC_UA_PLC-Client_16_PROJ_V1_3.zip	Dieses ZIP-Archiv enthält das Beispielprojekt für TIA Portal V16 und FW 2.8

### Hinweis

Die enthaltenen Projekte für TIA Portal V15.1 (S7-1500 FW 2.6) und V16 (ab S7-1500 FW 2.8) unterscheiden sich im Ablauf der Verbindungsüberwachung. Achten Sie darauf das entsprechende Projekt und die entsprechende Firmware zu wählen.

Falls Sie TIA Portal V16 mit einer S7-1500 mit FW 2.6 nutzen, verwenden Sie das Projekt für TIA Portal V15 und rüsten Sie es auf TIA Portal V16 hoch.



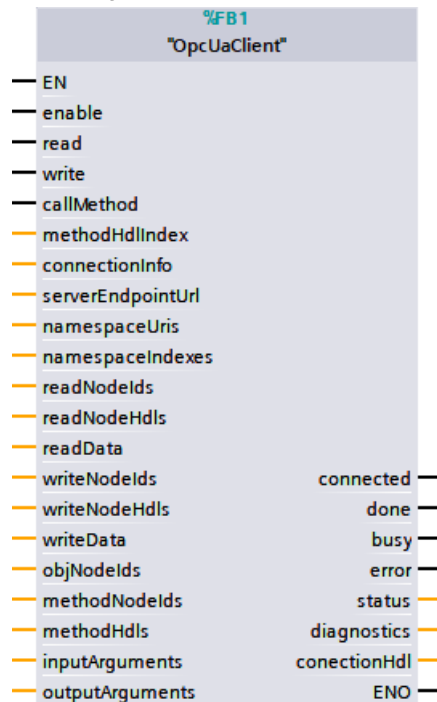
## 2 Engineering

### 2.1 Bausteinbeschreibung

#### 2.1.1 Schnittstellenbeschreibung

Die folgende Abbildung zeigt den OPC UA-Client-Baustein "OpcUaClient":

Abbildung 2-1



Die folgende Tabelle erläutert die Input- beziehungsweise die InOut-Parameter des Bausteins:

Tabelle 2-1

Parameter	Datentyp	Beschreibung
enable	Bool	TRUE = Verbindungsaufbau FALSE = Verbindungsabbau
read	Bool	Startet Read-Auftrag bei positiver Flanke
write	Bool	Startet Write-Auftrag bei positiver Flanke
callMethod	Bool	Ruft die Methode von "methodHdlIndex" bei positiver Flanke auf
methodHdlIndex	Int	Die aufzurufende Methode aus der Methoden-Liste
connectionInfo	"OPC_UA_SessionConnectInfo"	Verbindungsinformationen zum Session-Aufbau mit dem OPC UA-Server
serverEndpointUrl	String	Die Endpoint-URL des OPC UA-Servers
namespaceUris	Variant	Zeiger auf den Speicherbereich der abgefragten Namespace-URIs
namespaceIndexes	Variant	Zeiger auf den Speicherbereich der abgefragten Namespace-Indizes

Parameter	Datentyp	Beschreibung
readNodeIds	Variant	Zeiger auf den Speicherbereich (Array of "OPC_UA_NodeId") der zu lesenden Node-IDs
readNodeHdls	Variant	Zeiger auf den Speicherbereich (Array of DWORD) der zu lesenden Node-Handles
readData	Variant	Zeiger auf den Speicherbereich (UDT oder STRUCT) der gelesenen Werte
writeNodeIds	Variant	Zeiger auf den Speicherbereich (Array of "OPC_UA_NodeId") der zu schreibenden Node-IDs
writeNodeHdls	Variant	Zeiger auf den Speicherbereich (Array of DWORD) der zu schreibenden Node-Handles
writeData	Variant	Zeiger auf den Speicherbereich (UDT) der zu schreibenden Werte
objNodeIds	Variant	Zeiger auf den Speicherbereich (Array of "OPC_UA_NodeId") der aufzurufenden Object-Node-IDs von Methoden
methodNodeIds	Variant	Zeiger auf den Speicherbereich (Array of DWORD) der aufzurufenden Node-IDs von Methoden
methodNodeHdls	Variant	Zeiger auf den Speicherbereich (DWORD) der aufzurufenden Methoden-Handles
inputArguments	Variant	Zeiger auf die Eingangsparameter (UDT oder STRUCT) der aufzurufenden Methode
outputArguments	Variant	Zeiger auf die Ausgangsparameter (UDT oder STRUCT) der aufzurufenden Methode

**Hinweis** Bei Verwendung mehrerer Methoden müssen Sie die InOut-Parameter „inputArguments“ und „outputArguments“ zur Laufzeit des Programms an die aufzurufenden Methoden anpassen. Beachten Sie hierzu Kapitel ["3.4 Tipps & Tricks"](#).

Die folgende Tabelle erläutert die Output-Parameter des Bausteins:

Tabelle 2-2

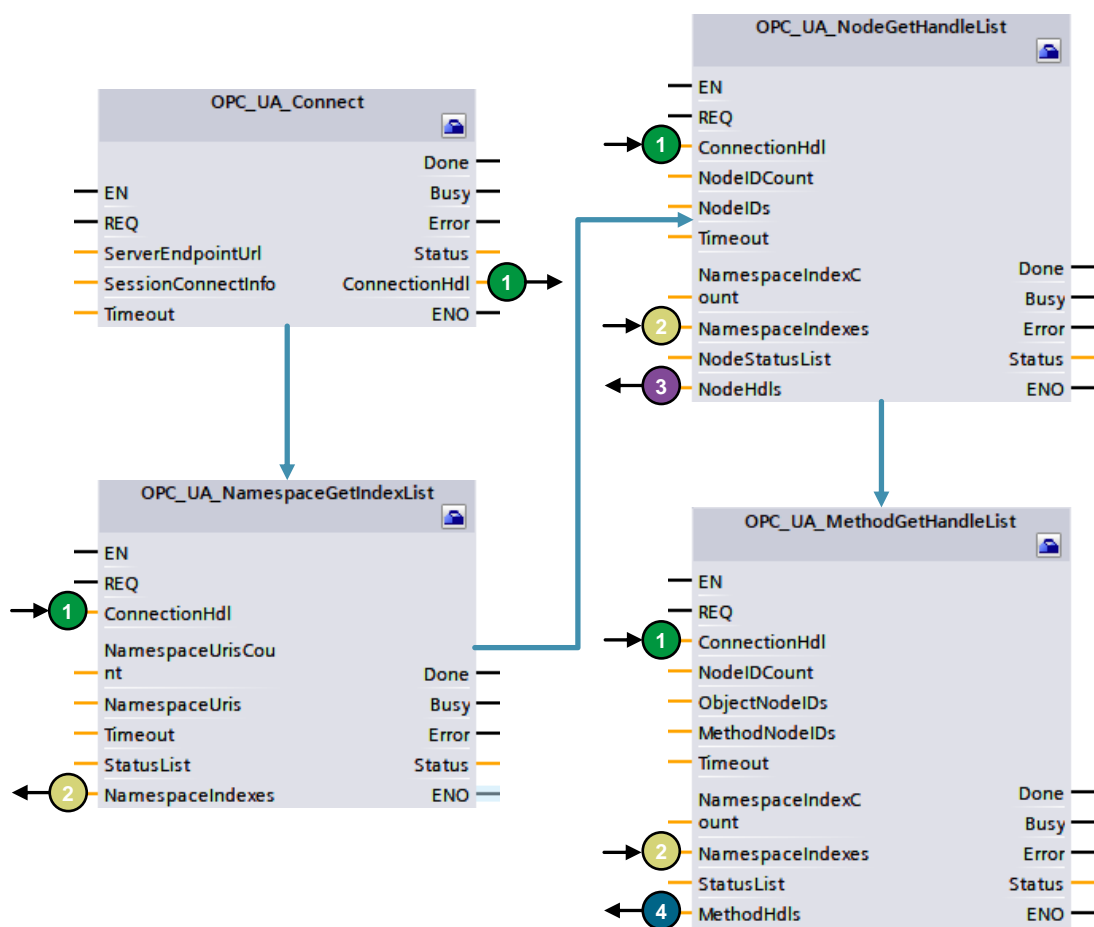
Parameter	Datentyp	Beschreibung
connected	Bool	TRUE = mit Server verbunden und Session aufgebaut
done	Bool	TRUE = der letzte Auftrag wurde ohne Fehler beendet
busy	Bool	TRUE = ein Auftrag wird bearbeitet
error	Bool	TRUE = der letzte Auftrag wurde mit einem Fehler beendet
status	Word	Ursache eines Fehlers
diagnostics	"typeFbDiagnostics"	Ausgang für erweiterte Diagnose
connectionHdl	DWord	Verbindungs-Handle der aktuellen Session zur Verwendung bei weiteren Aufrufen der OPC UA-Systemfunktionsbausteine

### 2.1.2 Funktionsbeschreibung

Der S7-Anwenderbaustein "OpcUaClient" implementiert für Sie die OPC UA-Client-Funktionen. Hierfür enthält der Baustein Schrittketten, in denen die OPC UA-Systemfunktionsbausteine aufgerufen und ausgewertet werden. Der Baustein benötigt die im TIA Portal projektierte Client-Schnittstelle, die alle Informationen zur OPC UA-Kommunikation bereitstellt. Weitere Informationen zur Client-Schnittstelle finden Sie unter Kapitel [2.2.2 OPC UA-Client-Schnittstelle erstellen](#).

Die folgenden Abbildungen erläutern allgemein die Abhängigkeiten und den Workflow zwischen den OPC UA-Systemfunktionsbausteinen:

Abbildung 2-2 Verbindungs Aufbau



© Siemens AG 2019. All rights reserved

Abbildung 2-3 Variablen lesen/schreiben und Methoden aufrufen

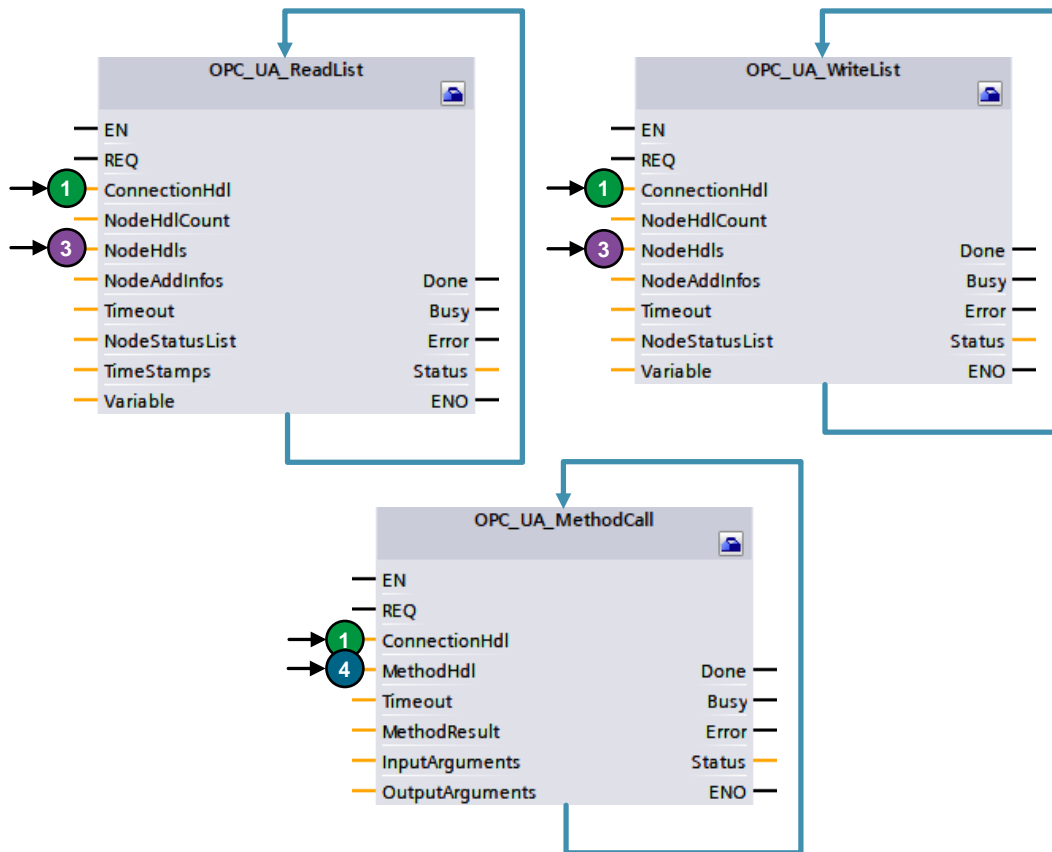
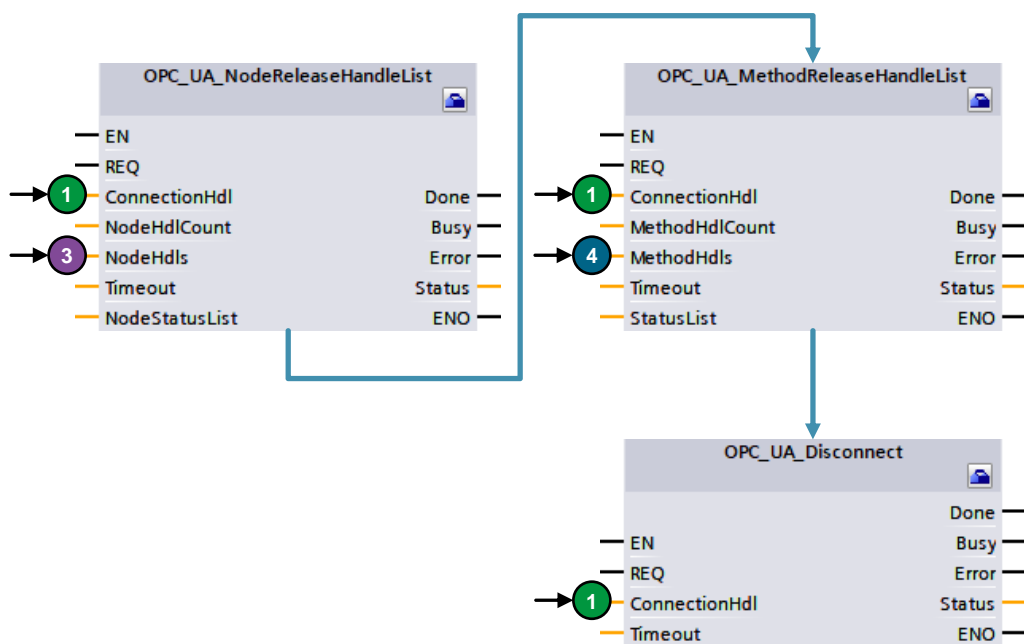


Abbildung 2-4 Verbindungsabbau und Ressourcen freigeben



### 2.1.3 Funktionsablauf

Die folgenden Erläuterungen erklären Ihnen den Ablauf der Funktionen innerhalb des Bausteins „OpcUaClient“:

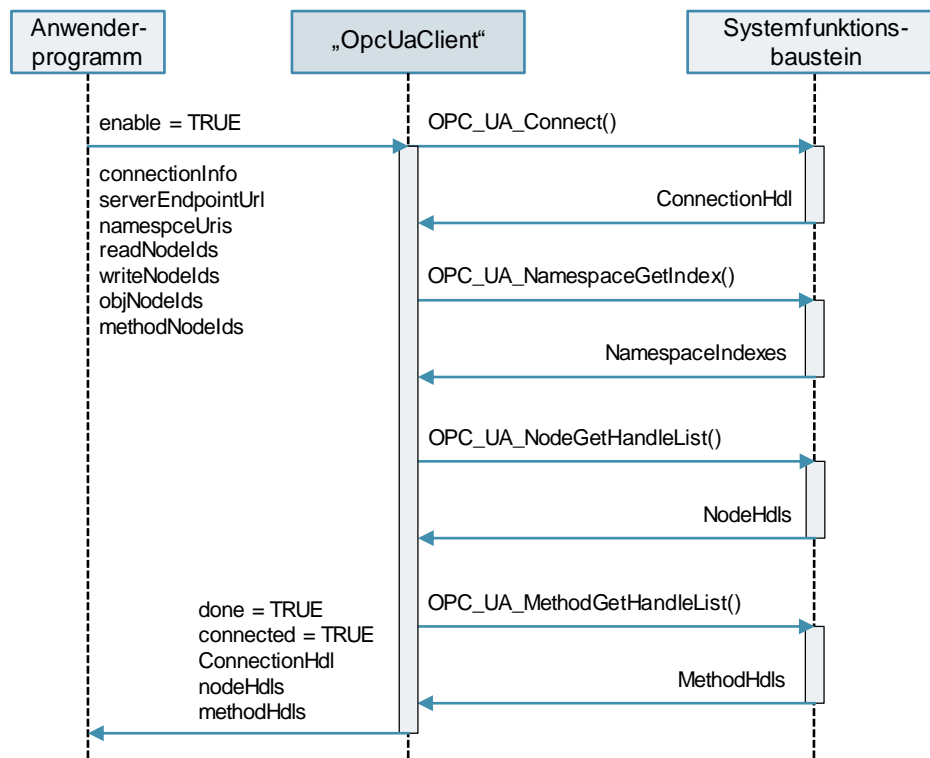
#### Verbindungsaufbau

Für den Verbindungsaufbau wird der Systemfunktionsbaustein "OPC\_UA\_Connect" ausgeführt. Dieser Baustein baut eine Verbindung zum Server auf und aktiviert eine Session. Nach erfolgreichem Verbindungsaufbau liefert der Baustein ein "Connection Handle" zurück. Dieses Handle referenziert für alle weiteren OPC UA-Funktionen auf die bestehende Verbindung/Session.

Nachdem der Baustein die Verbindung aufgebaut hat, wird der Systemfunktionsbaustein "OPC\_UA\_NamespaceGetIndex" ausgeführt. Dieser Baustein liefert die Namespace-Indexe für die in der Client-Schnittstelle projektierten Nodes zurück. Die Indexe sind obligatorisch für die Adressierung der Nodes im späteren Verlauf.

Anschließend werden die Nodes der Lese-, Schreib- und Methodenlisten am Server registriert. Hierfür werden die Systemfunktionsbausteine "OPC\_UA\_NodeGetHandleList" und "OPC\_UA\_MethodGetHandleList" ausgeführt. Für jede registrierte Node wird ein Handle zurückgegeben, das zum Lesen oder Schreiben der Node im späteren Verlauf benötigt wird.

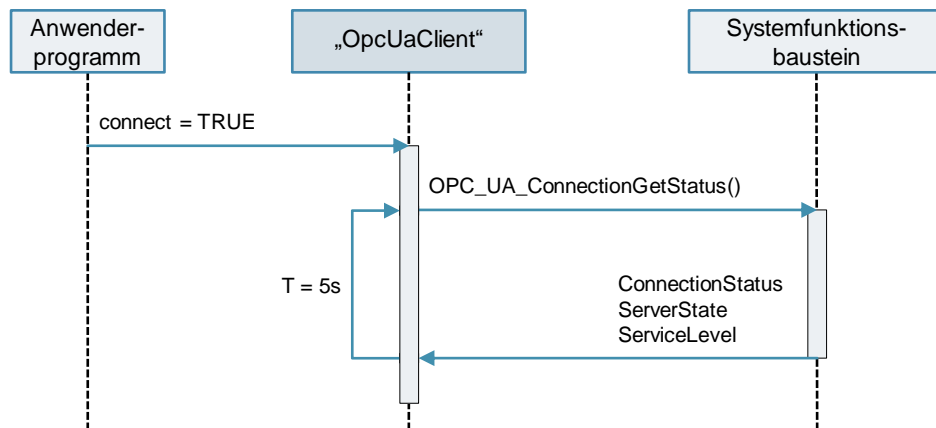
Abbildung 2-5



### Verbindungsüberwachung

Für die Verbindungsüberwachung wird der Systemfunktionsbaustein "OPC-UA\_ConnectionGetStatus" zyklisch (5 s) aufgerufen, nachdem Sie eine Verbindung erfolgreich aufgebaut haben. Nach jedem Aufruf werden die vom Baustein gelieferten Daten ausgewertet, um Verbindungsabbrüche oder anderweitige Fehler zu diagnostizieren. Weitere Informationen zur Verbindungsüberwachung erhalten Sie in Kapitel [3.1 Verbindungsüberwachung im Baustein "OpcUaClient"](#).

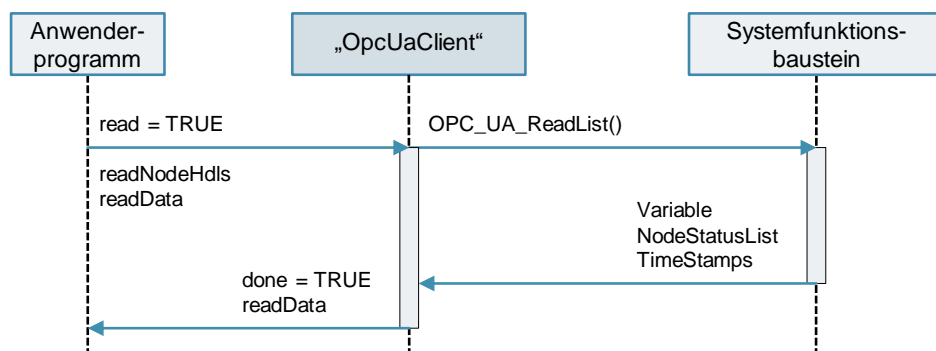
Abbildung 2-6



### Variablen lesen

Um die Variablen vom Server zu lesen wird der Systemfunktionsbaustein "OPC-UA\_ReadList" ausgeführt. Der Baustein verwendet die beim Verbindungsaufbau registrierten Handles, um die Variablen über den "RegisteredRead"-Dienst zu lesen. Der SFB liefert die Werte jeder Node zurück. Zusätzlich werden ein Status-Code und ein Zeitstempel pro Node ausgegeben. Der Anwenderbaustein "OpcUaClient" wertet die Status-Codes für Sie aus und setzt im Fehlerfall die entsprechenden Ausgänge ("error" = "TRUE"; "errorId" = "71", "status" = "<StatusCode>").

Abbildung 2-7

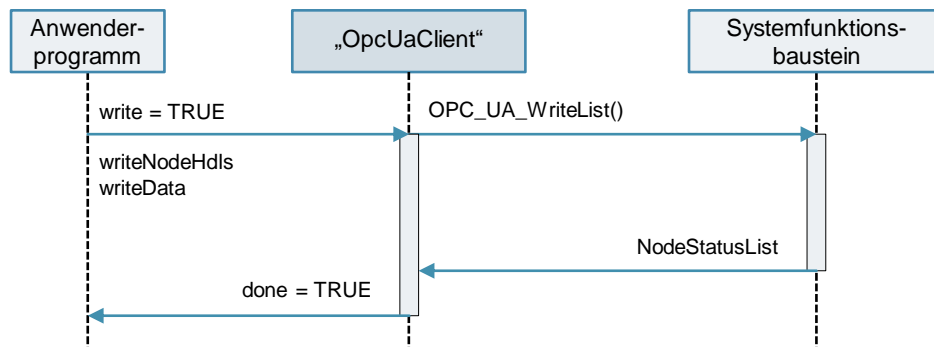




### Variablen schreiben

Um die Variablen auf den Server zu schreiben wird der Systemfunktionsbaustein "OPC\_UA\_WriteList" ausgeführt. Der Baustein verwendet die beim Verbindungsaufbau registrierten Handles, um die Variablen über den "RegisteredWrite"-Dienst zu schreiben. Der SFB gibt einen Status-Code pro geschriebenem Node zurück. Der Anwenderbaustein "OpcUaClient" wertet die Status-Codes für Sie aus und setzt im Fehlerfall die entsprechenden Ausgänge ("error" = "TRUE"; "errorId" = "72", "status" = "<StatusCode>").

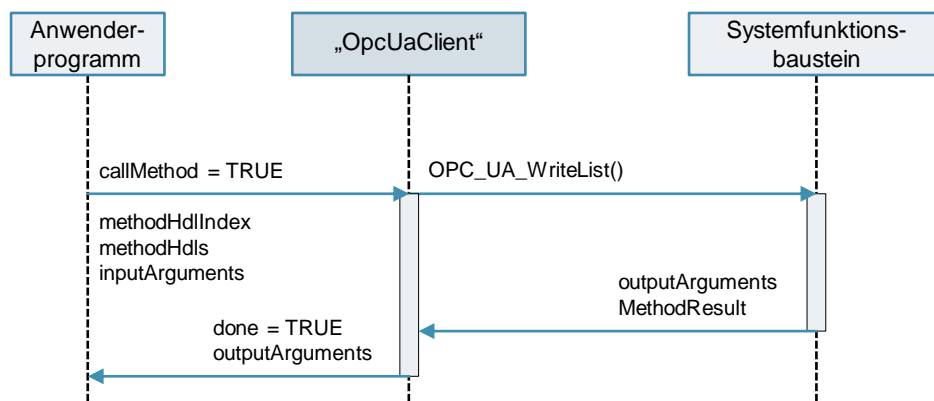
Abbildung 2-8



### Methoden aufrufen

Um die Methode auf dem Server auszuführen wird der Systemfunktionsbaustein "OPC\_UA\_MethodCall" ausgeführt. Der Baustein verwendet die beim Verbindungsaufbau registrierten Handles und die Werte am Eingang "inputArguments", um die Methoden über den "RegisteredMethodCall"-Dienst aufzurufen. Der SFB gibt ein Method-Result zurück. Der Anwenderbaustein "OpcUaClient" wertet dieses Result für Sie aus und setzt im Fehlerfall die entsprechenden Ausgänge ("error" = "TRUE"; "errorId" = "73", "status" = "<MethodResult>"). Zusätzlich werden die Rückgabewerte der Methode am Ausgang "outputAruguments" ausgegeben.

Abbildung 2-9

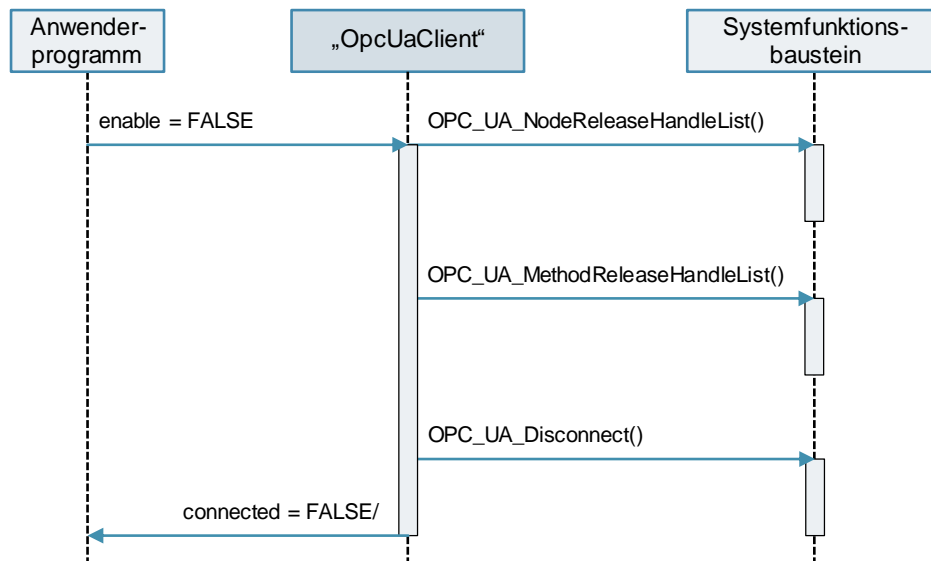


## Verbindungsabbau

Für den Verbindungsabbau wird der Systemfunktionsbaustein "OPC-UA\_Disconnect" ausgeführt. Dieser Baustein baut die Verbindung zum Server ab, deaktiviert die bestehende Session und gibt verwendeten Ressourcen der Client-CPU wieder frei.

Vor dem Verbindungsabbau werden die Systemfunktionsbausteine "OPC-UA\_NodeReleaseHandleList" und "OPC-UA\_MethodReleaseHandleList" ausgeführt. Diese SFBs geben die Handles wieder frei, die beim Verbindungsaufbau für alle Nodes der Lese-, Schreib- und Methodenlisten vom Server erstellt wurden.

Abbildung 2-10



## 2.1.4 Bausteindiagnose

Die folgenden Erläuterungen beschreiben die Diagnose des FBs "OpcUaClient".

### Bausteingänge ohne Fehler

Die folgende Tabelle erläutert die Werte des Parameters "status" in Abhängigkeit von "done" und "busy", wenn kein Fehler vorliegt:

Tabelle 2-3

"status"	"done"	"busy"	Beschreibung
16#0000_0000	FALSE	FALSE	Client nicht verbunden.
16#0000_0000	TRUE	FALSE	Auftrag erfolgreich ausgeführt.
16#0000_7000	FALSE	FALSE	Client verbunden und bereit für Auftrag.
16#0000_7000	TRUE	FALSE	Verbindung erfolgreich aufgebaut und bereit für Auftrag.
16#0000_7001	FALSE	TRUE	Baustein bearbeitet Auftrag.

### Bausteingänge im Fehlerfall

Über die Schnittstellenparameter "error", "status" und "diagnostics" diagnostizieren Sie den Baustein. Wenn ein Fehler ansteht, wird "error" gesetzt. Der Ausgang "status" liefert Ihnen die dazugehörige Fehlerquelle des FB. Die erweiterten Diagnoseinformationen entnehmen Sie dem Ausgang "diagnostics". Die Werte dieser Struktur bleiben bis zum nächsten Fehler bestehen.

Die folgende Tabelle erläutert die Struktur der Variable "diagnostics":

Tabelle 2-4

Variable	Beschreibung
status	Letzter Statuscode des Schnittstellenparameters "status" des FBs "OpcUaClient".
subfunctionStatus	Status bzw. Rückgabewert des internen SBFs, bei dem der Fehler aufgetreten ist. Detaillierte Informationen entnehmen Sie der Online-Hilfe ("F1") zum jeweiligen SFB, beziehungsweise den Statuslisten im Instanz-Datenbaustein. (16#80FF_0000 = Lese-, Schreib- oder Methodenliste nicht registriert)

Die folgende Tabelle erläutert die Fehlerquellen zu den Statuscodes des Schnittstellenparameter "status":

Tabelle 2-5

Statuscode	Fehlerquelle
16#8601	Fehler beim Verbindungsaufbau (SFB "OPC_UA_Connect").
16#8602	Fehler beim Auslesen des Namespace-Indexes (SFB "OPC_UA_NamespaceGetIndexList").
16#8604	Fehler beim Lesen einer Leseliste (SFB "OPC_UA_ReadList").
16#8605	Fehler beim Schreiben einer Schreibliste (SFB "OPC_UA_WriteList").
16#8606	Fehler beim Aufrufen einer Methode aus einer Methodenliste (SFB "OPC_UA_MethodCall").

Statuscode	Fehlerquelle
16#8608	Fehler beim Verbindungsabbau (SFB "OPC_UA_Disconnect").
16#8610	Fehler beim Ausführen der Verbindungsüberwachung (SFB "OPC_UA_ConnectionGetStatus").
16#8611	Fehler bei der Verbindungsüberwachung bezogen auf den Verbindungsstatus ("CSTATE").
16#8612	Fehler bei der Verbindungsüberwachung bezogen auf den Serverstatus ("SSTATE").
16#8613	Fehler bei der Verbindungsüberwachung bezogen auf das Servicelevel ("SLVL").
16#8614	Fehler in der Statusliste beim Lesen einer Leseliste.
16#8615	Fehler in der Statusliste beim Schreiben einer Schreibliste.
16#8616	Fehler in der Serialisierung beim Methodenaufruf (SFB "Serialize").
16#8617	Fehler in der Statusliste beim Auslesen der Namespace-Indexes.
16#8626	Fehler im OPC UA-Status eines Methodenaufrufs.
16#8631	Fehler beim Registrieren einer Leseliste (SFB "OPC_UA_NodeGetHandleList").
16#8632	Fehler beim Registrieren einer Schreibliste (SFB "OPC_UA_NodeGetHandleList").
16#8633	Fehler beim Registrieren einer Methodenliste (SFB "OPC_UA_MethodGetHandleList").
16#8671	Fehler bei der Freigabe einer Leseliste (SFB "OPC_UA_NodeReleaseHandleList").
16#8672	Fehler bei der Freigabe einer Schreibliste (SFB "OPC_UA_NodeReleaseHandleList").
16#8673	Fehler bei der Freigabe einer Methodenliste (SFB "OPC_UA_MethodReleaseHandleList").
16#86F4	Fehler beim Lesen einer Leseliste; Leseliste nicht registriert.
16#86F5	Fehler beim Schreiben einer Schreibliste; Schreibliste nicht registriert.
16#86F6	Fehler beim Aufrufen einer Methode; Methodenliste nicht registriert.
16#8711	Fehler bei der Verbindungsüberwachung bezogen auf den Verbindungsstatus ("CSTATE"); Verbindung wurde abgebrochen.
16#8731	Fehler in der Statusliste beim Registrieren einer Leseliste.
16#8732	Fehler in der Statusliste beim Registrieren einer Schreibliste.
16#8733	Fehler in der Statusliste beim Registrieren einer Methodenliste.
16#8771	Fehler in der Statusliste beim Freigeben einer Leseliste.
16#8772	Fehler in der Statusliste beim Freigeben einer Schreibliste.
16#8773	Fehler in der Statusliste beim Freigeben einer Methodenliste.

## 2.2 Projektierung und Programmierung

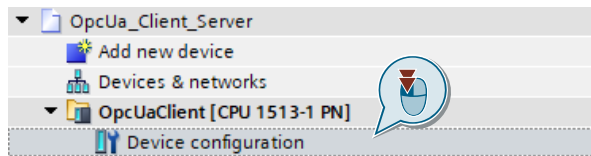
Bevor Sie den OPC UA-Baustein nutzen können, müssen Sie die folgenden Schritte durchführen:

- OPC UA-Client in der Gerätekonfiguration aktivieren
- OPC UA-Client-Schnittstelle erstellen
- OPC UA-Baustein "OpcUaClient" parametrieren

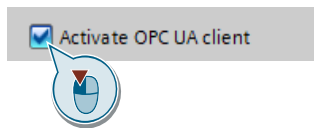
### 2.2.1 OPC UA-Client aktivieren

Um die Funktionen des OPC UA-Clients nutzen zu können, müssen Sie den Client in den Eigenschaften der CPU aktivieren. Gehen Sie folgendermaßen vor:

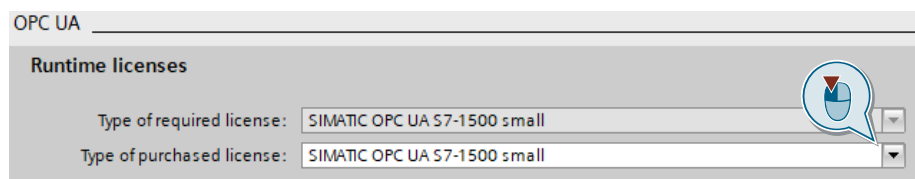
1. Navigieren Sie im Projektbaum in TIA Portal zur "Gerätekonfiguration" ("Device configuration") Ihrer CPU.



2. Navigieren Sie im Inspektorfenster zu "OPC UA > Client > Allgemein" ("OPC UA > Client > General") und aktivieren Sie das Kontrollkästchen "OPC UA-Client aktivieren" ("Activate OPC UA client").



3. Bestätigen Sie anschließend, dass Sie die benötigte Lizenz für OPC UA besitzen. Navigieren Sie zu "Runtime-Lizenzen" ("Runtime licenses") und wählen Sie in der Kategorie "OPC UA" die entsprechende Lizenz aus.



#### Hinweis

Je nach verwendeter SIMATIC S7-1500 CPU benötigen Sie eine passende OPC UA-Lizenz ("small", "medium" oder "large"). Beachten Sie hierzu Kapitel [3.3 Mengengerüst und Lizenzen des OPC UA-Clients](#).

## 2.2.2 OPC UA-Client-Schnittstelle erstellen

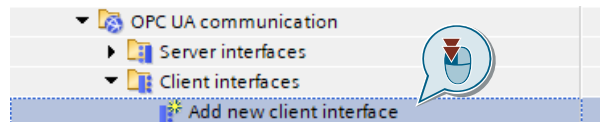
Um die OPC UA-Funktionen der S7-Funktionsbausteine zu nutzen, müssen Sie zunächst eine Client-Schnittstelle erstellen. Die Schnittstelle enthält alle relevanten Informationen, die der Client-Baustein "OpcUaClient" benötigt. Das TIA Portal legt diese Informationen in zwei automatisch generierten Datenbausteinen ab:

- "<InterfaceName>\_Configuration"  
Dieser DB enthält die Verbindungsinformationen, Node-IDs und Datentypen.
- "<InterfaceName>\_Data"  
In diesem DB werden die gelesenen bzw. die zu schreibenden Werte gespeichert. Der DB enthält auch eine Status-Liste zu den einzelnen Variablen.

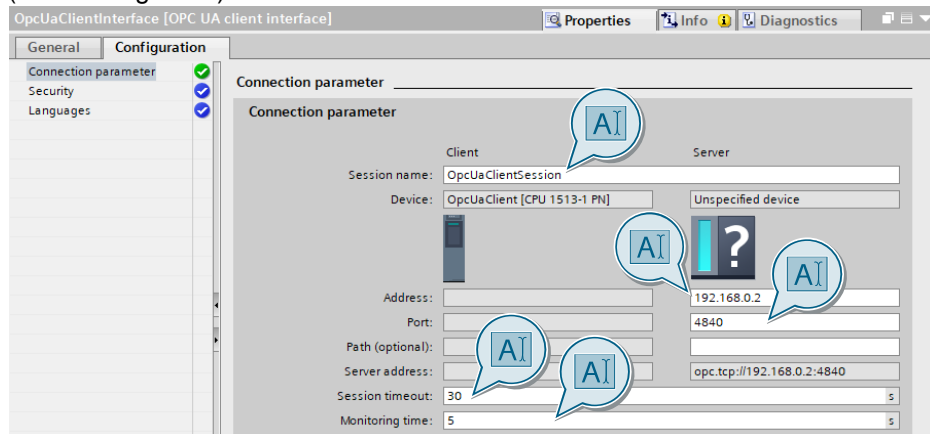
### Verbindungsparameter, Autorisierung und Authentifizierung konfigurieren

Gehen Sie zur Konfiguration der Verbindungsinformationen folgendermaßen vor:

1. Navigieren Sie im Projektbaum in TIA Portal zu "OPC UA-Kommunikation > Client-Schnittstellen" ("OPC UA communication > Client interfaces") und doppelklicken Sie auf "Neue Client-Schnittstelle hinzufügen" ("Add new client interface") aus.

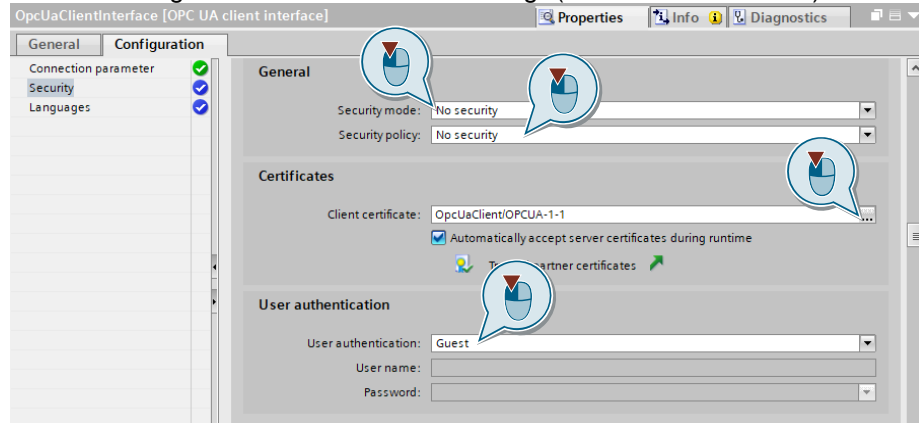


2. Tragen Sie im Inspektorfenster unter "Eigenschaften > Konfiguration > Verbindungsparameter" ("Properties > Configuration > Connection parameter") einen Session-Name ("Session name"), die "Adresse" ("Address") und den "Port" des Servers ein. Vergeben Sie außerdem ein sinnvolles "Session-Timeout" ("Session timeout") und die gewünschte "Überwachungszeit" ("Monitoring time").





3. Wählen Sie im Inspektorfenster unter "Eigenschaften > Konfiguration > Security" ("Properties > Configuration > Security") den benötigten "Security-Modus" ("Security mode") und die "Security Policy" ("Security policy") aus. Erstellen Sie zudem ein "Client-Zertifikat" ("Client certificate"), falls Sie eine signierte und verschlüsselte Verbindung aufbauen möchten, und konfigurieren Sie die benötigte "Benutzer-Authentifizierung" ("User authentication").

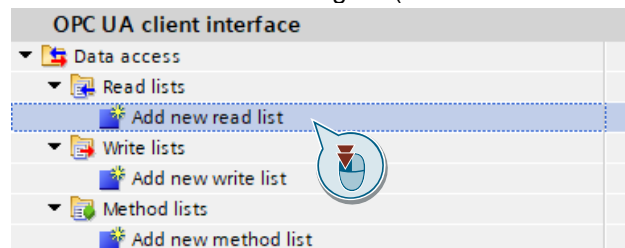


**Hinweis** Die Security von OPC UA wird in diesem Dokument nicht weiter erläutert.

### Leselisten, Schreiblisten und Methodenlisten konfigurieren

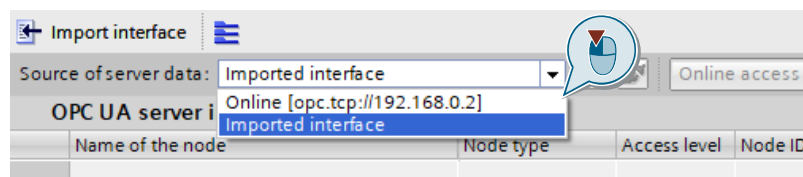
Gehen Sie zur Konfiguration der Listen folgendermaßen vor:

1. Navigieren Sie im Arbeitsbereich in TIA Portal im linken Abschnitt zu "Datenzugriff > Leselisten" ("Data access > Read lists") und doppelklicken Sie auf "Neue Leseliste hinzufügen" ("Add new read list").

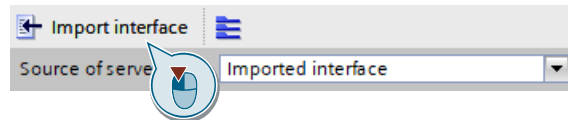


2. Sie haben nun zwei Möglichkeiten, die Leseliste zu befüllen: Über ein importiertes Server-Interface als XML-Datei (a) oder über eine Online-Verbindung direkt zu einem Server (b).

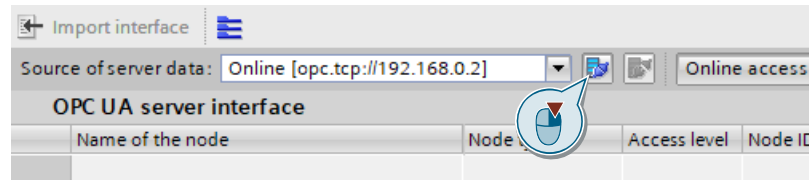
Wählen Sie in der Klappliste "Quelle der Server-Daten" ("Source of server data") im rechten Abschnitt des Arbeitsbereichs eine der beiden Optionen aus.



- a. Klicken Sie auf die Schaltfläche "Schnittstelle importieren" ("Import interface"), wenn Sie die Option "Importierte Schnittstelle" ("Imported interface") verwenden. Wählen Sie im erscheinenden Dialog die XML-Datei Ihrer Sever-Schnittstelle aus und bestätigen Sie mit "Importieren" ("Import").



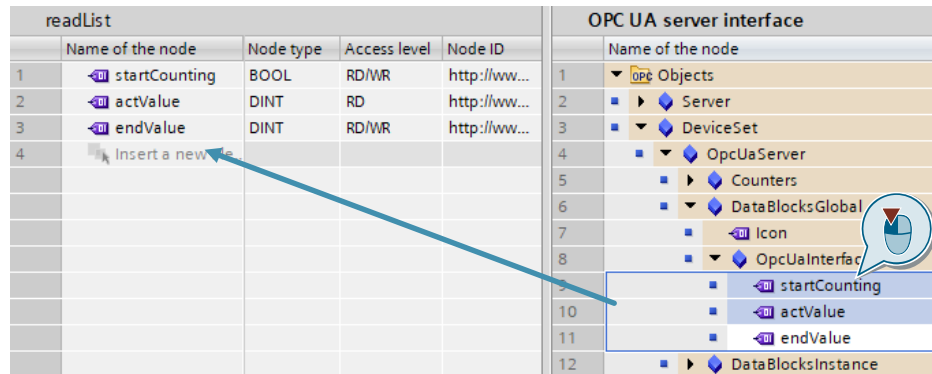
- b. Klicken Sie auf die Schaltfläche "Mit Online-Server verbinden" ("Connect to online server"), wenn Sie die Option "Online" verwenden.



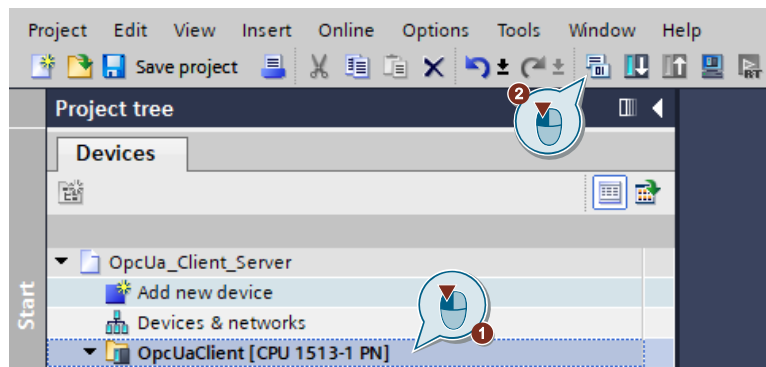
Hierbei baut TIA Portal eine Online-Verbindung zum bereits projektierten OPC UA-Server auf. Alternativ können Sie über die Schaltfläche "Online-Zugänge" ("Online access") eine Verbindung zu einem anderen OPC UA-Server aufbauen.

**Hinweis** Vergewissern Sie sich, dass der OPC UA-Server von Ihrer Engineering-Station aus erreichbar ist, bevor Sie eine Online-Verbindung aufbauen.

3. Ziehen Sie per Drag & Drop die zu lesenden Variablen des Server-Interface in die Leseliste.



4. Wiederholen Sie die Schritte 1 bis 3 für die "Schreiblisten" ("Write lists") und "Methodenlisten" ("Method lists").
5. Markieren Sie Ihre CPU in der Projektnavigation und klicken Sie auf das Symbol "Übersetzen" ("Compile"), damit TIA Portal die Datenbausteine "Configuration" und "Data" generiert. In diesen beiden Datenbausteinen sind nun alle OPC UA-Übergabeparameter für den S7-Anwenderbaustein "OpUaClient" enthalten.



Nach jeder Änderung in der OPC UA-Client-Schnittstelle müssen Sie die CPU neu kompilieren, damit die Parameter in die Datenbausteine übertragen werden.

## 2.2.3 OPC UA-Client-Baustein parametrieren

Erstellen Sie zunächst ein Client-Interface wie in Kapitel [2.2.2 OPC UA-Client-Schnittstelle erstellen](#) beschrieben.

### Baustein in Ihr Projekt kopieren

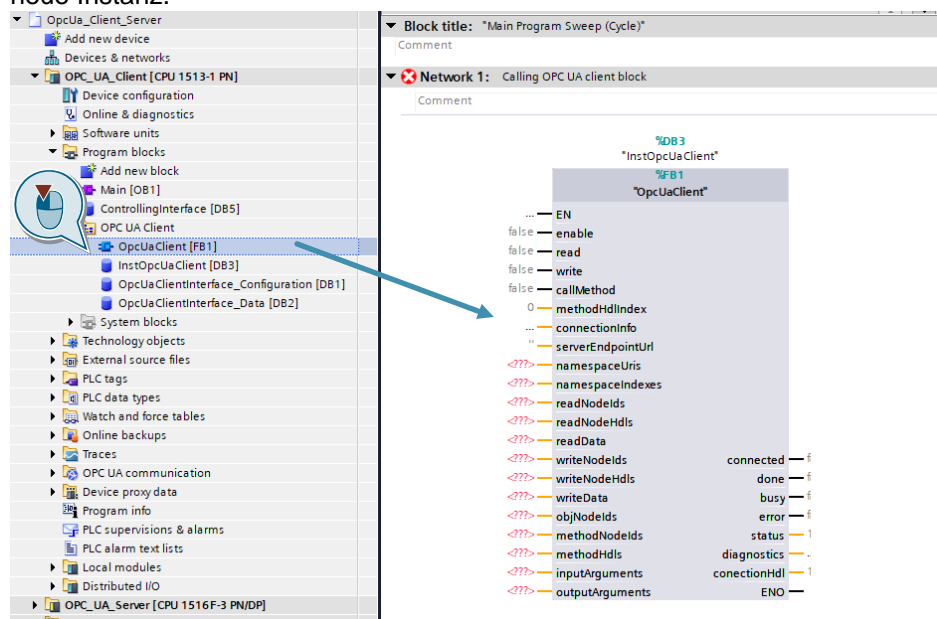
Kopieren Sie den OPC UA-Clientbaustein "OpcUaClient" folgendermaßen in Ihr Projekt:

1. Laden Sie das Beispielprojekt "109762770 OPC-UA-PLC-Client\_15\_PROJ\_V11.zip" bzw. "109762770 OPC-UA-PLC-Client\_16\_PROJ\_V11.zip" herunter und entpacken Sie das ZIP-Archiv.
2. Öffnen Sie die Projekt-Datei "OpcUaClientServer15.ap15\_1" bzw. "OpcUaClientServer16.ap16".
3. Navigieren Sie im Projektbaum zu "OPC-UA-Client > Programmbausteine > OPC UA Client" ("OPC-UA-Client > Program blocks > OPC UA Client") und kopieren Sie den Baustein "OpcUaClient" in Ihr eigenes Projekt.

### Baustein parametrieren

Parametrieren Sie die Schnittstelle des Bausteins "OpcUaClient" in Ihrem Anwenderprogramm folgendermaßen:

1. Rufen Sie den Baustein in einem zyklischen OB auf und erstellen Sie eine neue Instanz.



2. Verschalten Sie die Ein- und Ausgabeparameter zur Steuerung und Diagnose des Bausteins. Erstellen Sie hierfür geeignete Variablen.

Bausteinparameter	Datentyp
connect	Bool
read	Bool
write	Bool
callMethod	Bool
methodHdlIndex	SInt
connected	Bool
done	Bool
busy	Bool
error	Bool
status	DWord
diagnostics	"typeFbDiagnostics"
connectionHdl	DWord

3. Verschalten Sie die Client-Schnittstellen-Parameter des Bausteins "OpcUaClient". Die zu verschaltenden Variablen finden Sie in den Datenbausteinen, die über die Client-Schnittstelle generiert werden. Belegen Sie nicht benötigte Blöcke der folgenden Tabelle mit "NULL". Die Parameter des Blocks "Connection" sind obligatorisch.

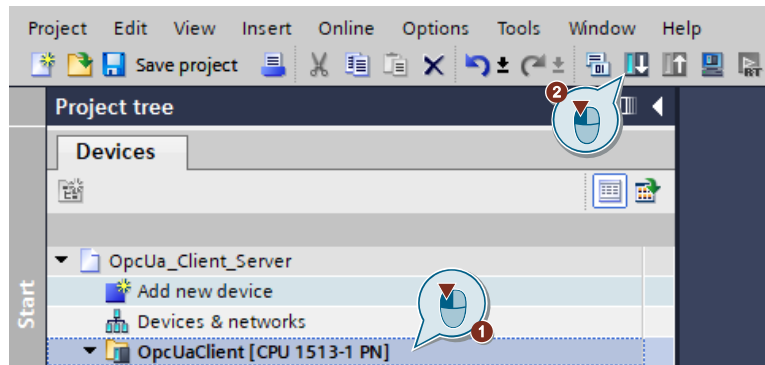
Bausteinparameter	Datenbaustein und Datenbausteinparameter
<b>Connection</b>	
connectionInfo	"<InterfaceName>_Configuration".Connection.ConnectInfo
serverEndpointUrl	"<InterfaceName>_Configuration".Connection.ServerEndpointUrl
namespaceUris	"<InterfaceName>_Configuration".Namespaces.NamespaceURIs
namespaceIndexes	"<InterfaceName>_Configuration".Namespaces.ServerNameSpaceIndexes
connectionHdl	(Optional) "<InterfaceName>_Configuration".Connection.ConenctionHdl
<b>Read</b>	
readNodeIds	"<InterfaceName>_Configuration".ReadLists.<ReadListName>.Nodes
readNodeHdls	"<InterfaceName>_Configuration".ReadLists.<ReadListName>.NodeHdls
readData	"<InterfaceName>_Data".<ReadListName>.Variable
<b>Write</b>	
writeNodeIds	"<InterfaceName>_Configuration".WriteLists.<WriteListName>.Nodes
writeNodeHdls	"<InterfaceName>_Configuration".WriteLists.<WriteListName>.NodeHdls
writeData	"<InterfaceName>_Data".<WriteListName>.Variable
<b>CallMethod</b>	
objNodeIds	"<InterfaceName>_Configuration".MethodLists.<MethodListName>.ObjectNodes

Bausteinparameter	Datenbaustein und Datenbausteinparameter
methodNodeIds	"<InterfaceName>_Configuration".MethodLists. <MethodListName>.MethodNodes
methodeNodeHdls	"<InterfaceName>_Configuration".MethodLists. <MethodListName>.MethodHdls
inputArguments	"<InterfaceName>_Data".<MethodListName>.<MethodName>. Inputs
outputArguments	"<InterfaceName>_Data".<MethodListName>.<MethodName>. Outputs

**Hinweis**

Falls eine oder mehrere Funktionen des OpcUaClient-Bausteins (Read, Write oder CallMethod) nicht benötigt werden, verschalten Sie die entsprechenden Parameter mit "NULL".

4. Markieren Sie Ihre CPU in der Projektnavigation und klicken Sie auf das Symbol "Herunterladen in Geräte" ("Download to device"), um das Projekt in Ihre CPU zu laden.





## 2.3 Inbetriebnahme und Bedienung des Beispiels

In diesem Anwendungsbeispiel erfolgt die Bedienung über eine Beobachtungstabelle aus TIA Portal.

In Kapitel [3.2 OPC UA-Server im Beispiel](#) finden Sie einen Überblick über die Demoapplikation des OPC UA-Servers im Beispiel

### 2.3.1 Inbetriebnahme des Beispiels

Gehen Sie folgendermaßen vor, um das Beispiel in Betrieb zu nehmen:

1. Laden Sie das Beispielprojekt "109762770 OPC\_UA\_PLC-Client\_15\_PROJ\_V1\_2.zip" beziehungsweise "109762770 OPC\_UA\_PLC-Client\_16\_PROJ\_V1\_2.zip" herunter und entpacken Sie das ZIP-Archiv.
2. Öffnen Sie die Projekt-Datei "OpcUaClientServer.ap15\_1" bzw. "OpcUaClientServer.ap16".
3. (Optional) Tauschen Sie Ihre CPUs im Projekt, falls Sie andere SIMATIC S7-1500 Controller verwenden. Klicken Sie hierzu im Projektbaum mit der rechten Maustaste auf die CPUs und anschließend im Kontextmenü auf "Gerät tauschen..." ("Change device...").
4. Laden Sie die beiden projektierten CPUs ("OpcUaClient" und "OpcUaServer") in Ihre Steuerungen.

### 2.3.2 Beschreibung der Oberfläche

Die folgende Abbildung zeigt die Beobachtungstabelle "ControllingOpcUa" des Beispiels:

Abbildung 2-11

	Name	Display for...	Monitor value	
1	// Inputs			
2	"ControllingInterface".opcUaConnect	Bool		<input type="checkbox"/>
3	"ControllingInterface".opcUaRead	Bool	1	<input type="checkbox"/>
4	"ControllingInterface".opcUaWrite	Bool		<input type="checkbox"/>
5	"ControllingInterface".opcUaMethodIndex	DEC+/-		<input type="checkbox"/>
6	"ControllingInterface".opcUaMethodCall	Bool		<input type="checkbox"/>
7				
8	// Outputs			
9	"ControllingInterface".opcUaConnected	Bool		<input type="checkbox"/>
10	"ControllingInterface".opcUaDone	Bool		<input type="checkbox"/>
11	"ControllingInterface".opcUaBusy	Bool	2	<input type="checkbox"/>
12	"ControllingInterface".opcUaError	Bool		<input type="checkbox"/>
13	"ControllingInterface".opcUaErrorId	DEC+/-		<input type="checkbox"/>
14	"ControllingInterface".opcUaStatus	Hex		<input type="checkbox"/>
15				
16	// Data Read			
17	"OpcUaClientInterface_Data".readList.Variable.actValue	DEC+/-		<input type="checkbox"/>
18	"OpcUaClientInterface_Data".readList.Variable.endValue	DEC+/-	3	<input type="checkbox"/>
19	"OpcUaClientInterface_Data".readList.Variable.startCounting	Bool		<input type="checkbox"/>
20				
21	// Data Write			
22	"OpcUaClientInterface_Data".writeList.Variable.endValue	DEC+/-		<input type="checkbox"/>
23	"OpcUaClientInterface_Data".writeList.Variable.startCounting	Bool	4	<input type="checkbox"/>
24				
25	// Data Method			
26	"OpcUaClientInterface_Data".methodList.Method1.Inputs.DateTime	DATE_AN...	5	<input type="checkbox"/>
27				

Die folgende Tabelle erläutert die Unterteilung der Beobachtungstabelle

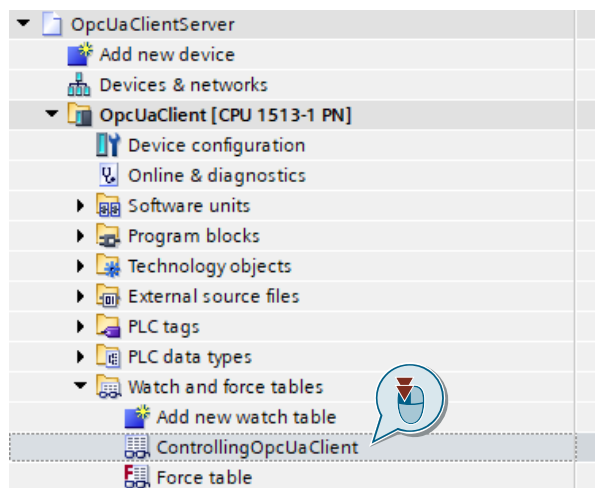
Tabelle 2-6

Nr.	Beschreibung
1.	Im Bereich "Inputs" befinden sich die Variablen zur Steuerung des Bausteins.
2.	Im Bereich "Outputs" befinden sich die Variablen zur Diagnose des Bausteins.
3.	Im Bereich "Data Read" befinden sich die Variablen, die vom OPC UA-Server gelesen werden.
4.	Im Bereich "Data Write" befinden sich die Variablen, die auf den OPC UA-Server geschrieben werden.
5.	Im Bereich "Data Method" befindet sich das "InputArgument" für die Methode, die auf dem OPC UA-Server ausgeführt wird.

### 2.3.3 Bedienung des Beispiels

Gehen Sie folgendermaßen vor, um das Beispiel zu bedienen:

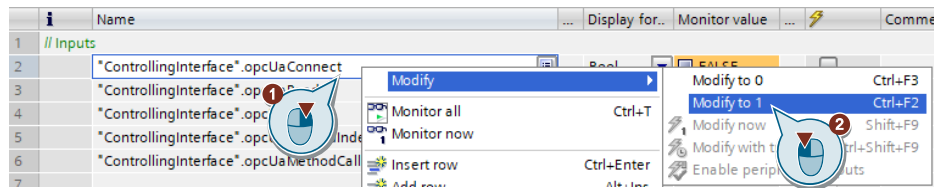
1. Navigieren Sie Im Projektbaum zu "OpcUaClient > Beobachtungs- und Forcetabellen" ("OpcUaClient > Watch and force tables") und öffnen Sie die Beobachtungstabelle "ControllingOpcUaClient" mit einem Doppelklick.



2. Klicken Sie im Arbeitsbereich auf das Symbol "Alle beobachten" ("Monitor all"), um die Variablen der Steuerung live zu beobachten.



3. Verbinden Sie Ihren Client zunächst mit dem Server, indem Sie die Variable "opcUaConnect" im Bereich "Inputs" auf "TRUE" steuern. Rechtsklicken Sie hierzu auf die Variable und wählen Sie "Steuern > Steuern auf 1" ("Modify > Modify to 1") aus.



4. Überprüfen Sie den Verbindungsaufbau. Beobachten Sie hierzu die Variable "opcUaConnected" im Bereich "Outputs":  
 "TRUE" - Client ist verbunden.  
 "FALSE" - Client ist nicht verbunden.

Name	Display for...	Monitor value	Comme
// Inputs			
*ControllingInterface".opcUaConnect	Bool	FALSE	
*ControllingInterface".opcUaDone	Bool	FALSE	
*ControllingInterface".opcUaBusy	Bool	FALSE	
*ControllingInterface".opcUaError	Bool	FALSE	
*ControllingInterface".opcUaErrorId	DEC+/-	0	
*ControllingInterface".opcUaStatus	Hex	16#0000_0000	
// Outputs			
*ControllingInterface".opcUaConnected	Bool	TRUE	
*ControllingInterface".opcUaDone	Bool	FALSE	
*ControllingInterface".opcUaBusy	Bool	FALSE	
*ControllingInterface".opcUaError	Bool	FALSE	
*ControllingInterface".opcUaErrorId	DEC+/-	0	
*ControllingInterface".opcUaStatus	Hex	16#0000_0000	

Weitere Informationen zur Diagnose des Bausteins finden Sie in Kapitel [2.1.4 Bausteindiagnose](#).

Nachdem Sie Ihren Client mit dem OPC UA-Server verbunden haben, können Sie folgende Operationen durchführen:

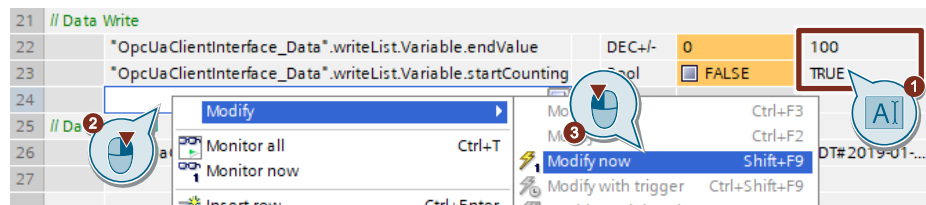
- Variablen vom Server lesen
- Variablen zum Server schreiben
- Methode ausführen

### Variablen des Servers schreiben

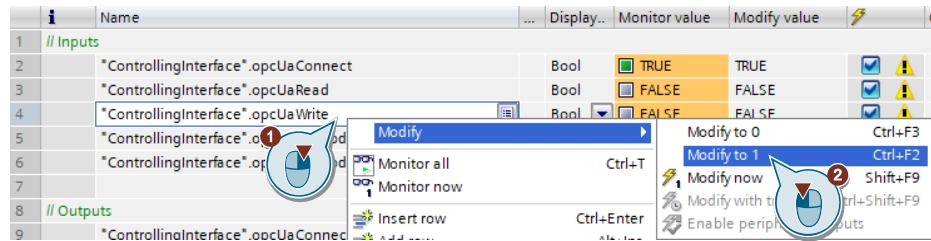
Gehen Sie folgendermaßen vor, um die Variablen des Servers zu schreiben:

1. Steuern Sie die Variablen im Bereich "Data Write". Tragen Sie die gewünschten zu schreibenden Werte in die Spalte "Steuerwert" ("Modify value") ein. Die Variable "endValue" bestimmt, wie hoch der Counter im OPC UA-Server zählen soll (im Beispiel: "100"). Die Variable "startCounting" bestimmt, ob der Counter des Servers aktiv ist oder nicht (im Beispiel: "TRUE").

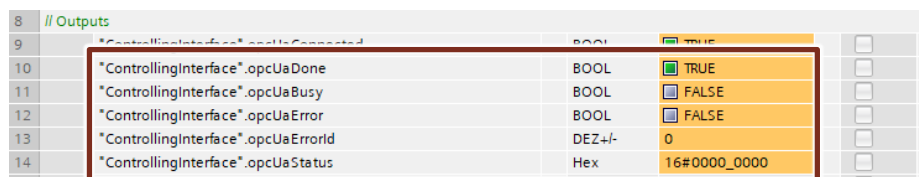
Rechtsklicken Sie in den Arbeitsbereich und wählen Sie "Steuern > Sofort steuern" ("Modify > Modify now") aus, um die Werte in die Steuerung zu übertragen.



- Schreiben Sie die OPC UA-Variablen des Servers, indem Sie die Variable "OpcUaWrite" im Bereich "Inputs" auf "TRUE" steuern. Rechtsklicken Sie hierzu auf die Variable und wählen Sie "Steuern > Steuern auf 1" ("Modify > Modify to 1") aus.



- Überprüfen Sie den Schreibauftrag. Beobachten Sie hierzu die Variablen "opcUaDone", "opcUaError", "opcUaErrorId" und "opcUaStatus" im Bereich "Outputs":

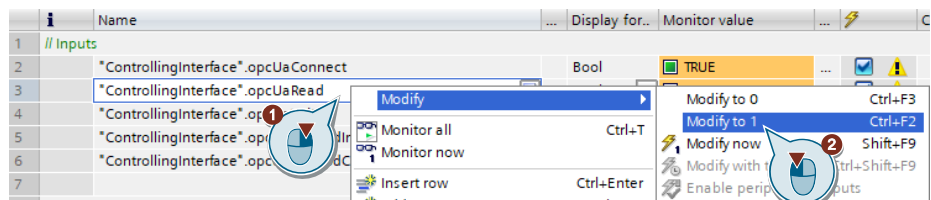


Weitere Informationen zur Diagnose des Bausteins finden Sie in Kapitel [2.1.4 Bausteindiagnose](#).

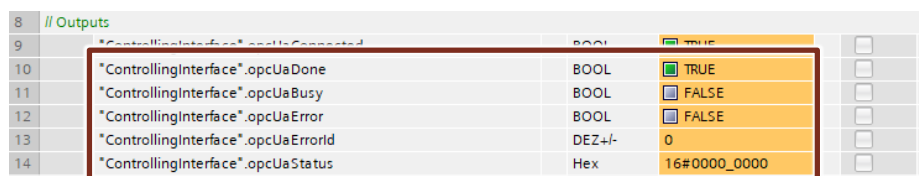
### Variablen des Servers lesen

Gehen Sie folgendermaßen vor, um die Variablen des Servers zu lesen:

- Lesen Sie die OPC UA-Variablen des Servers, indem Sie die Variable "OpcUaRead" im Bereich "Inputs" auf "TRUE" steuern. Rechtsklicken Sie hierzu auf die Variable und wählen Sie "Steuern > Steuern auf 1" ("Modify > Modify to 1") aus.



- Überprüfen Sie den Leseauftrag. Beobachten Sie hierzu die Variablen "opcUaDone", "opcUaError", "opcUaErrorId" und "opcUaStatus" im Bereich "Outputs":



Weitere Informationen zur Diagnose des Bausteins finden Sie in Kapitel [2.1.4 Bausteindiagnose](#).

3. Beobachten Sie die Werte im Bereich "Data Read". Wenn Sie zuvor einen Wert für "endValue" vorgegeben und "startCounting" gesetzt haben, dann verändert sich der Wert "actValue" jede Sekunde. Überprüfen Sie die Wertänderung mit weiteren Leseaufträgen.

16	// Data Read			
17		*OpcUaClientInterface_Data*.readList.Variable.actValue	DEZ+/-	68
18		*OpcUaClientInterface_Data*.readList.Variable.endValue	DEZ+/-	100
19		*OpcUaClientInterface_Data*.readList.Variable.startCounting	BOOL	<input checked="" type="checkbox"/> TRUE FALSE

### Methode des Servers aufrufen

Gehen Sie folgendermaßen vor, um die OPC UA-Methode des Servers aufzurufen:

1. Steuern Sie die Variable "dateTime" im Bereich "Data Method". Tragen Sie die zu übergebende PLC-Zeit in die Spalte "Steuerwert" ("Modify value") ein. Rechtsklicken Sie in den Arbeitsbereich und wählen Sie "Steuern > Sofort steuern" ("Modify > Modify now") aus, um den Wert in die Steuerung zu übertragen.

25	// Data Method			
26		*OpcUaClientInterface_Data*.methodList.Method1.Inputs.DateTime	DATE	2019-01-25 13:26:30
27				

2. Rufen Sie die OPC UA-Methode des Servers auf, indem Sie die Variable "OpcUaMethodCall" im Bereich "Inputs" auf "TRUE" steuern. Rechtsklicken Sie hierzu auf die Variable und wählen Sie "Steuern > Steuern auf 1" ("Modify > Modify to 1") aus.

	Name	Anzeig...	Beobachtung...	Steuerwert
1	// Inputs			
2	*ControllingInterface*.opcUaConnect	BOOL	<input checked="" type="checkbox"/> TRUE	TRUE
3	*ControllingInterface*.opcUaRead	BOOL	<input type="checkbox"/> FALSE	FALSE
4	*ControllingInterface*.opcUaWrite	BOOL	<input type="checkbox"/> FALSE	FALSE
5	*ControllingInterface*.opcUaMethodIndex	DEZ+/-	0	
6	*ControllingInterface*.opcUaMethodCall	BOOL	<input type="checkbox"/> FALSE	FALSE
7	// Outputs			
8	*ControllingInterface*.opcUaDone	BOOL	<input type="checkbox"/> FALSE	FALSE
9	*ControllingInterface*.opcUaError	BOOL	<input type="checkbox"/> FALSE	FALSE
10	*ControllingInterface*.opcUaErrorId	DEZ+/-	0	
11	*ControllingInterface*.opcUaStatus	Hex	16#0000_0000	

3. Überprüfen Sie den Methodenaufruf. Beobachten Sie hierzu die Variablen "opcUaDone", "opcUaError", "opcUaErrorId" und "opcUaStatus" im Bereich "Outputs":

8	// Outputs			
9		*ControllingInterface*.opcUaConnect	BOOL	<input checked="" type="checkbox"/> TRUE
10		*ControllingInterface*.opcUaDone	BOOL	<input checked="" type="checkbox"/> TRUE
11		*ControllingInterface*.opcUaBusy	BOOL	<input type="checkbox"/> FALSE
12		*ControllingInterface*.opcUaError	BOOL	<input type="checkbox"/> FALSE
13		*ControllingInterface*.opcUaErrorId	DEZ+/-	0
14		*ControllingInterface*.opcUaStatus	Hex	16#0000_0000

Weitere Informationen zur Diagnose des Bausteins finden Sie in Kapitel [2.1.4 Bausteindiagnose](#).

4. Überprüfen Sie die neue PLC-Zeit auf dem Display Ihrer Server-CPU.

### 3 Wissenswertes

#### 3.1 Verbindungsüberwachung im Baustein "OpcUaClient"

##### Verbindungsüberwachung für S7-1500 FW 2.6

Das folgende Diagramm beschreibt das Fehlerhandling der implementierten Verbindungsüberwachung durch den Systemfunktionsbaustein "OPC\_UA\_ConnectionGetStatus" (CGS):

Abbildung 3-1

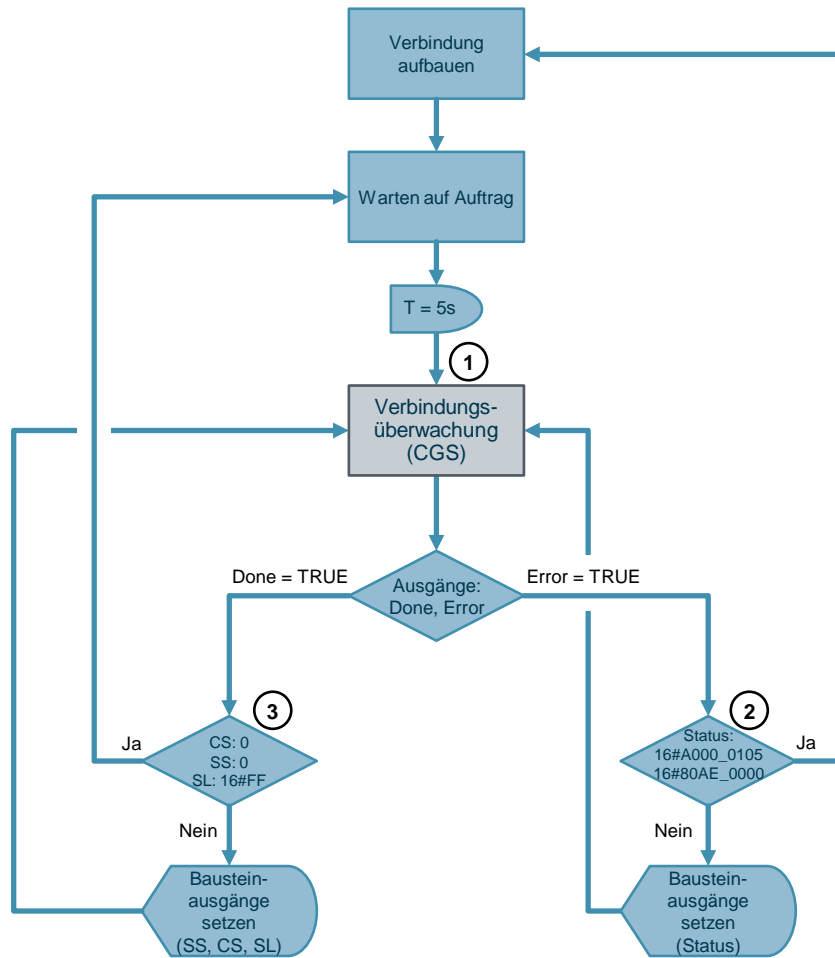


Tabelle 3-1

Nr.	Beschreibung
1.	Nachdem eine Verbindung vom Baustein "OpcUaClient" erfolgreich zu einem Server aufgebaut ist, wird der Systemfunktionsbaustein "OPC_UA_ConnectionGetStatus" periodisch (5 Sekunden) aufgerufen, um die Verbindung zum Server zu überwachen.
2.	Wenn das "Connection Handle" zum Server ungültig ist, erhalten Sie einen Fehler am SFB "OPC_UA_ConnectionGetStatus" ("Error" = "TRUE"). Bei den dazugehörigen "Status" = "16#A000_0105" oder "16#80AE_0000" wird eine neue Verbindung aufgebaut. Andere Werte signalisieren Fehler, die vom Anwender behoben werden müssen, bevor eine neue Verbindung aufgebaut werden kann. In diesem Fall werden die Ausgänge des Bausteins "OpcUaClient" gesetzt und der SFB erneut ausgeführt. Weitere Informationen zu den Ausgängen finden Sie in der TIA Online-Hilfe (F1).

Nr.	Beschreibung
3.	Wenn ein gültiges "Connection Handle" zum Server besteht, wird der SFB "OPC_UA_ConnectionGetStatus" erfolgreich ausgeführt ("Done" = "TRUE"). Der Baustein "OpcUaClient" wertet in diesem Fall die Ausgänge "ConnectionStatus" (CS), " ServerState " (SS) und "ServiceLevel" (SL) aus. Die Werte CS = "0", SS = "0" und SL = "16#FF" signalisieren eine korrekte Verbindung, der Baustein "OpcUaClient" ist bereit weitere Aufträge auszuführen. Andere Werte signalisieren Fehler, die vom Anwender behoben werden müssen, bevor eine neue Verbindung aufgebaut werden kann. In diesem Fall werden die Ausgänge des Bausteins "OpcUaClient" gesetzt und der SFB erneut ausgeführt. Weitere Informationen zu den Ausgängen finden Sie in der TIA Online-Hilfe (F1).

**Verbindungsüberwachung für S7-1500 FW 2.8 oder neuer**

Das folgende Diagramm beschreibt das Fehlerhandling der implementierten Verbindungsüberwachung durch den Systemfunktionsbaustein "OPC\_UA\_ConnectionGetStatus" (CGS):

Abbildung 3-2

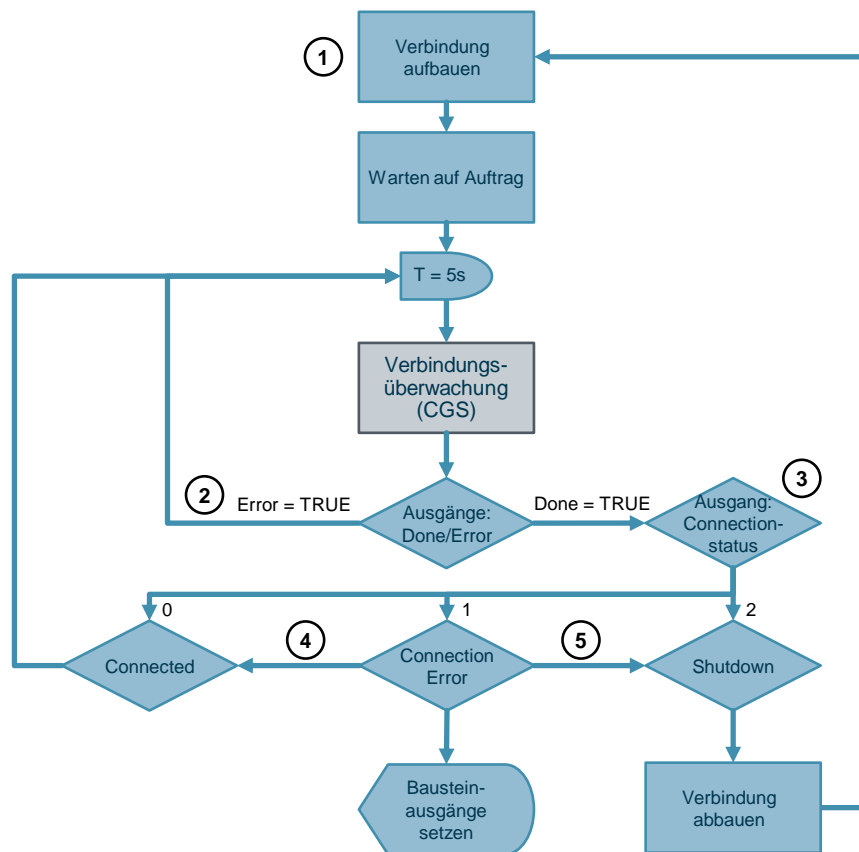


Tabelle 3-2

Nr.	Beschreibung
1.	Nachdem eine Verbindung vom Baustein "OpcUaClient" erfolgreich zu einem Server aufgebaut ist, wird der Systemfunktionsbaustein "OPC-UA_ConnectionGetStatus" periodisch (5 Sekunden) ausgeführt, um die Verbindung zum Server zu überwachen.
2.	Liefert der Baustein "OPC-UA_ConnectionGetStatus" bei der Verbindungsüberwachung einen Fehler ("Error" = TRUE) zurück, ist davon auszugehen, dass gerade ein unterlagerter Leseauftrag der Knoten "ServerState" und "ServiceLevel" stattfindet. Der Baustein muss in diesem Fall erneut ausgeführt werden.
3.	Läuft die Verbindungsüberwachung über den Baustein "OPC-UA_ConnectionGetStatus" erfolgreich ("Done" = TRUE) ab, muss der Bausteinausgang "ConnectionStatus" ausgewertet werden: "0": Verbindung besteht ("UACS_Connected") "1": Verbindung unterbrochen ("UACS_ConnectionError") "2": Verbindung abgebaut ("UACS_Shutdown")
4.	Bei behobener Ursache eines Verbindungsfehlers reaktiviert ("UACS_ConnectionError" > "UACS_Connected") der OPC UA-Client einer SIMATIC S7-1500 die Verbindung automatisch.
5.	Bei behobener Ursache eines Verbindungsfehlers aber bereits deaktivierter Session am Server ("UACS_ConnectionError" > "UACS_Shutdown") muss die Verbindung abgebaut (SFB "OPC-UA_Disconnect") und anschließend erneut aufgebaut (SFB "OPC-UA_Connect") werden.



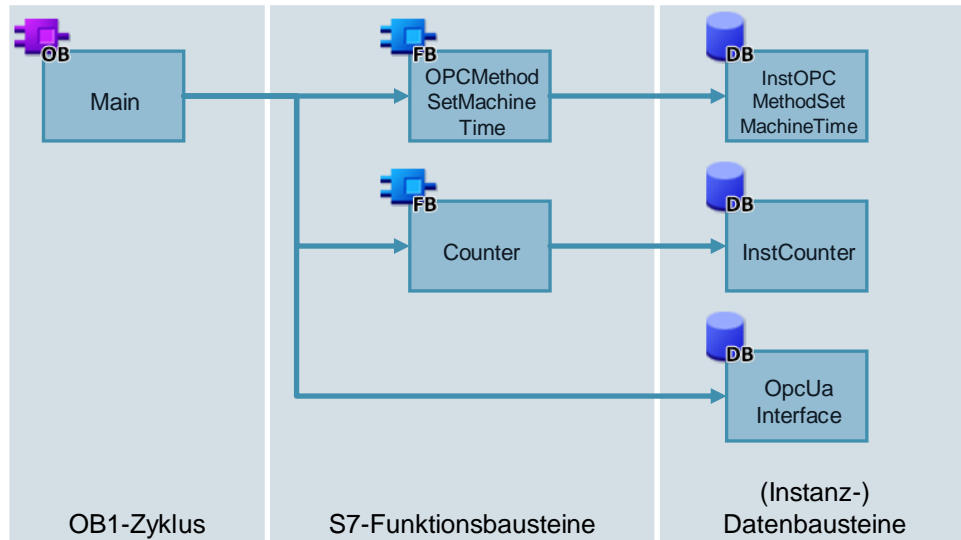
## 3.2 OPC UA-Server im Beispiel

Das S7-Anwenderprogramm der zweiten CPU "OpcUaServer" im Beispielprojekt beinhaltet den OB1, zwei Anwenderbausteine und einen Datenbaustein. Die CPU dient lediglich als Beispielservers.

### Aufrufhierarchie

Die nachfolgende Abbildung zeigt die Aufrufhierarchie des S7-Anwenderprogramms der CPU "OpcUaServer" im Beispielprojekt:

Abbildung 3-3



### Erläuterung der Bausteine

Im zyklischen Anwenderprogramm werden lediglich die Funktionsbausteine "Counter" und "OPCMethodSetMachineTime" aufgerufen.

Der Baustein "Counter" zählt im Sekundentakt die Variable "actValue" des Datenbausteins "OpcUaInterface" hoch, wenn die ebenfalls enthaltene Variable "startCounting" gesetzt ist. Über die Variable "endValue" können Sie bestimmen, wie weit "Counter" zählen kann. Bei Erreichen von "endValue" fängt der Counter wieder bei "0" an.

Der Baustein "OPCMethodSetMachineTime" implementiert eine OPC UA-Methode, welche die PLC-Zeit setzt.

### 3.3 Mengengerüst und Lizenzen des OPC UA-Clients

Die folgende Tabelle erläutert das Mengengerüst und die Lizenzen des OPC UA-Clients der SIMATIC S7-1500:

Tabelle 3-3

Runtime-Lizenz/ CPU-Typ	Small ET 200SP CPU CPU 1511 CPU 1513	Medium CPU 1515 CPU 1516 CPU 1507S	Large CPU 1517 CPU1518 CPU 1508S
Max. Anzahl der Verbindungen:	4	10	40
Max. Elemente in einer Lese-/Schreib-/Methodenliste:	300	300	300
Max. Nodes in einem Client-Interface:	1000	2000	5000
Max. parallele Lese-/Schreib-/Methoden-Aufträge:	5	5	5
Max. Anzahl von Methoden Handles:	100	100	100
Max. Anzahl von Input- /Output-Argumenten:	20	20	20

**Hinweis** Die SIMATIC S7-1500 CPU 1507S bietet trotz der Lizenz "Medium" das Mengengerüst "Large".

## 3.4 Tipps & Tricks

### PLC-Zeit einstellen

Bei der zertifikatsbasierten Authentifizierung wird das Gültigkeitsdatum der Zertifikate geprüft. Hierzu ist es erforderlich, dass Sie die PLC-Zeit korrekt einstellen. Wir empfehlen Ihnen dazu die Verwendung eines NTP-Servers.

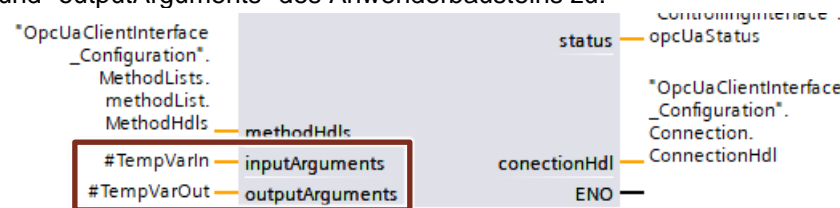
### Input- und Output-Argument für Methoden zur Laufzeit ändern

Da der Baustein "OpcUaClient" nur jeweils einen Eingang für die Input- und Output-Argumente bereitstellt, müssen Sie die Eingänge zur Laufzeit zuweisen, wenn Sie mehrere unterschiedliche Methoden aufrufen wollen. Gehen Sie folgendermaßen vor:

1. Erstellen Sie jeweils eine temporäre Variable vom Typ "Variant" in der Bausteinschnittstelle des Bausteins, für die Input- und OutputArguments:

Temp	
TempVarIn	Variant
TempVarOut	Variant

2. Weisen Sie die temporären Variablen den InOut-Parametern "inputArguments" und "outputArguments" des Anwenderbausteins zu:



3. Referenzieren Sie die temporären Variablen auf das gewünschte Input- bzw. Output-Argument mit Hilfe der Anweisung "REF":

```
#TempVarIn :=
REF("OpcUaClientInterface_Data".methodList.Method1.Inputs);
#TempVarOut :=
REF("OpcUaClientInterface_Data".methodList.Method1.Outputs);
```

4. Weisen Sie anschließend den temporären Variablen, abhängig von der gewählten Methode, ein Input- und Output-Argument mit Hilfe der Anweisung "VariantPut" zu:

```
IF methodHdlIndex = 1 THEN
  VariantPut (SRC:="OpcUaClientInterface_Data".methodList.
  Method1.Inputs, DST:=#TempVarIn);
  VariantPut (SRC:="OpcUaClientInterface_Data".methodList.
  Method1.Outputs, DST:=#TempVarOut);
END_IF;

IF methodHdlIndex = 2 THEN
  VariantPut (SRC:="OpcUaClientInterface_Data".methodList.
  Method2.Inputs, DST:=#TempVarIn);
  VariantPut (SRC:="OpcUaClientInterface_Data".methodList.
  Method2.Outputs, DST:=#TempVarOut);
END_IF;
```

Durch diese Vorgehensweise haben Sie die Möglichkeit, zur Laufzeit die Zuweisung der Bausteinschnittstellen zu ändern. Gehen Sie nach obenstehendem Schema vor, um je nach ausgewählter Methode (Eingangsparameter: "methodHdlIndex") die passenden Input- und Output-Argumente des Bausteins zuzuweisen. Sie finden die Argumente in folgendem Speicherbereich des Datenbausteins der Client-Schnittstelle:

- "<InterfaceName>\_Data".methodList.<MethodName>.Inputs
- "<InterfaceName>\_Data".methodList.<MethodName>.Outputs

## 4 Anhang

### 4.1 Service und Support

#### Industry Online Support

Sie haben Fragen oder brauchen Unterstützung?

Über den Industry Online Support greifen Sie rund um die Uhr auf das gesamte Service und Support Know-how sowie auf unsere Dienstleistungen zu.

Der Industry Online Support ist die zentrale Adresse für Informationen zu unseren Produkten, Lösungen und Services.

Produktinformationen, Handbücher, Downloads, FAQs und Anwendungsbeispiele – alle Informationen sind mit wenigen Mausklicks erreichbar:

[support.industry.siemens.com](https://support.industry.siemens.com)

#### Technical Support

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

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

[support.industry.siemens.com/cs/my/src](https://support.industry.siemens.com/cs/my/src)

#### SITRAIN – Digital Industry Academy

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

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

[siemens.de/sitrain](https://siemens.de/sitrain)

#### Serviceangebot

Unser Serviceangebot umfasst folgendes:

- Plant Data Services
- Ersatzteilservices
- Reparaturservices
- Vor-Ort und Instandhaltungsservices
- Retrofit- und Modernisierungsservices
- Serviceprogramme und Verträge

Ausführliche Informationen zu unserem Serviceangebot finden Sie im Servicekatalog:

[support.industry.siemens.com/cs/sc](https://support.industry.siemens.com/cs/sc)

#### Industry Online Support App

Mit der App "Siemens Industry Online Support" erhalten Sie auch unterwegs die optimale Unterstützung. Die App ist für iOS und Android verfügbar:

[support.industry.siemens.com/cs/ww/de/sc/2067](https://support.industry.siemens.com/cs/ww/de/sc/2067)

## 4.2 Links und Literatur

Tabelle 4-1

Nr.	Thema
\1\	Siemens Industry Online Support <a href="https://support.industry.siemens.com">https://support.industry.siemens.com</a>
\2\	Link auf die Beitragsseite des Anwendungsbeispiels <a href="https://support.industry.siemens.com/cs/ww/de/view/109762770">https://support.industry.siemens.com/cs/ww/de/view/109762770</a>

## 4.3 Änderungsdokumentation

Tabelle 4-2

Version	Datum	Änderung
V1.0	03/2019	Erste Ausgabe
V1.1	01/2020	Fehlerbereinigung; Erweitertes Fehlerhandling; Update für TIA Portal V16 und SIMATIC S7-1500 FW 2.8
V1.2	10/2020	Fehlerbereinigung im Projekt für TIA Portal V16
V1.2.1	08/2021	Fehlerbereinigung im Projekt für TIA Portal V16