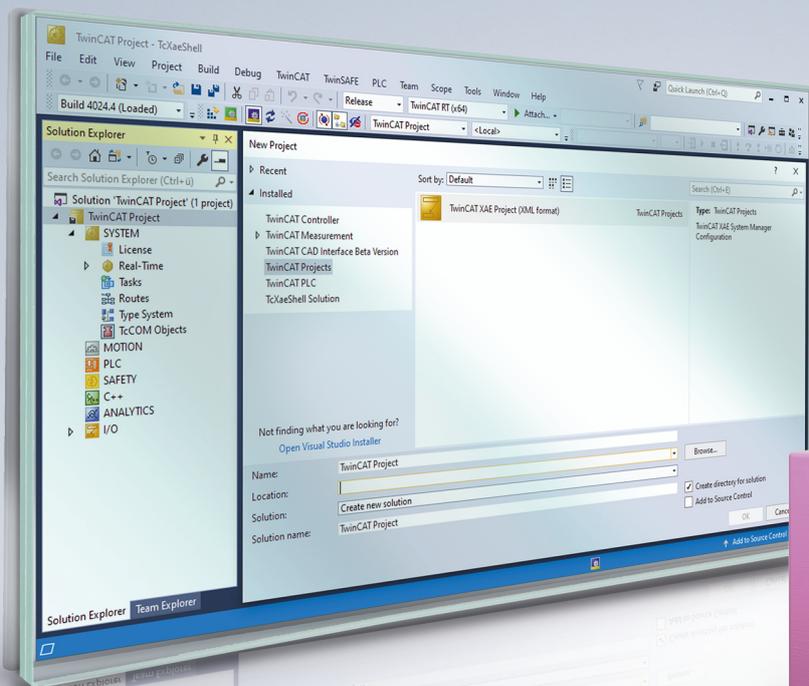


BECKHOFF New Automation Technology

Manual | DE

TF6105

TwinCAT 3 | OPC UA Pub/Sub



Inhaltsverzeichnis

1	Vorwort	5
1.1	Hinweise zur Dokumentation	5
1.2	Zu Ihrer Sicherheit	6
1.3	Hinweise zur Informationssicherheit	7
2	Überblick	8
3	Installation	9
3.1	Systemanforderungen	9
3.2	Installation	9
3.3	Lizenzierung	10
4	Technische Einführung	13
4.1	Quick Start	13
4.1.1	UDP	13
4.1.2	MQTT	17
4.2	Unterstützte Features	21
4.3	Protokollübersicht	23
4.4	Publisher/Subscriber	25
4.5	DataSets	27
4.6	Transportprotokolle	32
4.6.1	UDP	32
4.6.2	MQTT	32
4.7	Header-Layouts	35
4.8	Konfiguration Import/Export	36
4.9	In-depth	38
4.9.1	PublishingInterval	38
4.9.2	KeyFrames, DeltaFrames, KeepAlive	39
4.9.3	MessageReceiveTimeout	40
5	Samples	41
5.1	UDP	41
5.2	MQTT	42
5.3	EtherCAT Master	43
6	Tutorials	50
7	Anhang	51
7.1	Diagnose	51
7.2	Troubleshooting	53

1 Vorwort

1.1 Hinweise zur Dokumentation

Diese Beschreibung wendet sich ausschließlich an ausgebildetes Fachpersonal der Steuerungs- und Automatisierungstechnik, das mit den geltenden nationalen Normen vertraut ist.

Zur Installation und Inbetriebnahme der Komponenten ist die Beachtung der Dokumentation und der nachfolgenden Hinweise und Erklärungen unbedingt notwendig.

Das Fachpersonal ist verpflichtet, stets die aktuell gültige Dokumentation zu verwenden.

Das Fachpersonal hat sicherzustellen, dass die Anwendung bzw. der Einsatz der beschriebenen Produkte alle Sicherheitsanforderungen, einschließlich sämtlicher anwendbaren Gesetze, Vorschriften, Bestimmungen und Normen erfüllt.

Disclaimer

Diese Dokumentation wurde sorgfältig erstellt. Die beschriebenen Produkte werden jedoch ständig weiterentwickelt.

Wir behalten uns das Recht vor, die Dokumentation jederzeit und ohne Ankündigung zu überarbeiten und zu ändern.

Aus den Angaben, Abbildungen und Beschreibungen in dieser Dokumentation können keine Ansprüche auf Änderung bereits gelieferter Produkte geltend gemacht werden.

Marken

Beckhoff®, ATRO®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, MX-System®, Safety over EtherCAT®, TC/BSD®, TwinCAT®, TwinCAT/BSD®, TwinSAFE®, XFC®, XPlanar® und XTS® sind eingetragene und lizenzierte Marken der Beckhoff Automation GmbH.

Die Verwendung anderer in dieser Dokumentation enthaltenen Marken oder Kennzeichen durch Dritte kann zu einer Verletzung von Rechten der Inhaber der entsprechenden Kennzeichnungen führen.



EtherCAT® ist eine eingetragene Marke und patentierte Technologie, lizenziert durch die Beckhoff Automation GmbH, Deutschland.

Copyright

© Beckhoff Automation GmbH & Co. KG, Deutschland.

Weitergabe sowie Vervielfältigung dieses Dokuments, Verwertung und Mitteilung seines Inhalts sind verboten, soweit nicht ausdrücklich gestattet.

Zuwendungen verpflichten zu Schadenersatz. Alle Rechte für den Fall der Patent-, Gebrauchsmuster- oder Geschmacksmustereintragung vorbehalten.

Fremdmarken

In dieser Dokumentation können Marken Dritter verwendet werden. Die zugehörigen Markenvermerke finden Sie unter: <https://www.beckhoff.com/trademarks>.

1.2 Zu Ihrer Sicherheit

Sicherheitsbestimmungen

Lesen Sie die folgenden Erklärungen zu Ihrer Sicherheit.
Beachten und befolgen Sie stets produktspezifische Sicherheitshinweise, die Sie gegebenenfalls an den entsprechenden Stellen in diesem Dokument vorfinden.

Haftungsausschluss

Die gesamten Komponenten werden je nach Anwendungsbestimmungen in bestimmten Hard- und Software-Konfigurationen ausgeliefert. Änderungen der Hard- oder Software-Konfiguration, die über die dokumentierten Möglichkeiten hinausgehen, sind unzulässig und bewirken den Haftungsausschluss der Beckhoff Automation GmbH & Co. KG.

Qualifikation des Personals

Diese Beschreibung wendet sich ausschließlich an ausgebildetes Fachpersonal der Steuerungs-, Automatisierungs- und Antriebstechnik, das mit den geltenden Normen vertraut ist.

Signalwörter

Im Folgenden werden die Signalwörter eingeordnet, die in der Dokumentation verwendet werden. Um Personen- und Sachschäden zu vermeiden, lesen und befolgen Sie die Sicherheits- und Warnhinweise.

Warnungen vor Personenschäden

GEFAHR

Es besteht eine Gefährdung mit hohem Risikograd, die den Tod oder eine schwere Verletzung zur Folge hat.

WARNUNG

Es besteht eine Gefährdung mit mittlerem Risikograd, die den Tod oder eine schwere Verletzung zur Folge haben kann.

VORSICHT

Es besteht eine Gefährdung mit geringem Risikograd, die eine mittelschwere oder leichte Verletzung zur Folge haben kann.

Warnung vor Umwelt- oder Sachschäden

HINWEIS

Es besteht eine mögliche Schädigung für Umwelt, Geräte oder Daten.

Information zum Umgang mit dem Produkt



Diese Information beinhaltet z. B.:
Handlungsempfehlungen, Hilfestellungen oder weiterführende Informationen zum Produkt.

1.3 Hinweise zur Informationssicherheit

Die Produkte der Beckhoff Automation GmbH & Co. KG (Beckhoff) sind, sofern sie online zu erreichen sind, mit Security-Funktionen ausgestattet, die den sicheren Betrieb von Anlagen, Systemen, Maschinen und Netzwerken unterstützen. Trotz der Security-Funktionen sind die Erstellung, Implementierung und ständige Aktualisierung eines ganzheitlichen Security-Konzepts für den Betrieb notwendig, um die jeweilige Anlage, das System, die Maschine und die Netzwerke gegen Cyber-Bedrohungen zu schützen. Die von Beckhoff verkauften Produkte bilden dabei nur einen Teil des gesamtheitlichen Security-Konzepts. Der Kunde ist dafür verantwortlich, dass unbefugte Zugriffe durch Dritte auf seine Anlagen, Systeme, Maschinen und Netzwerke verhindert werden. Letztere sollten nur mit dem Unternehmensnetzwerk oder dem Internet verbunden werden, wenn entsprechende Schutzmaßnahmen eingerichtet wurden.

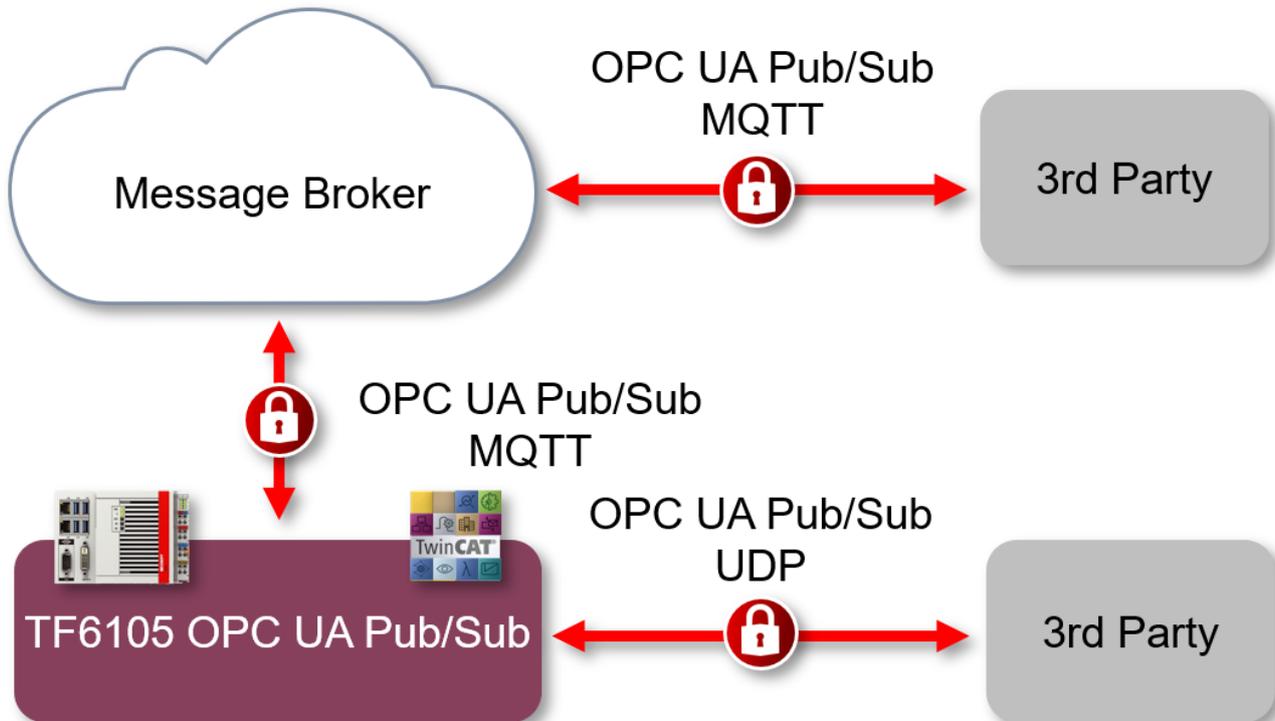
Zusätzlich sollten die Empfehlungen von Beckhoff zu entsprechenden Schutzmaßnahmen beachtet werden. Weiterführende Informationen über Informationssicherheit und Industrial Security finden Sie in unserem <https://www.beckhoff.de/secguide>.

Die Produkte und Lösungen von Beckhoff werden ständig weiterentwickelt. Dies betrifft auch die Security-Funktionen. Aufgrund der stetigen Weiterentwicklung empfiehlt Beckhoff ausdrücklich, die Produkte ständig auf dem aktuellen Stand zu halten und nach Bereitstellung von Updates diese auf die Produkte aufzuspielen. Die Verwendung veralteter oder nicht mehr unterstützter Produktversionen kann das Risiko von Cyber-Bedrohungen erhöhen.

Um stets über Hinweise zur Informationssicherheit zu Produkten von Beckhoff informiert zu sein, abonnieren Sie den RSS Feed unter <https://www.beckhoff.de/secinfo>.

2 Überblick

Das Produkt TF6105 TC3 OPC UA Pub/Sub beinhaltet Konnektivität über verschiedene Transportprotokolle, wie sie in der Pub/Sub-Erweiterung der OPC UA Spezifikation (OPC10000-14) spezifiziert sind. Viele Funktionen der Spezifikation wurden implementiert und stehen zur Nutzung bereit.



Dieses Produkt beinhaltet:

- einen Echtzeittreiber, der die OPC UA Pub/Sub-Kommunikation über UDP Unicast oder Multicast ermöglicht
- einen Echtzeittreiber für die OPC UA Pub/Sub-Kommunikation über MQTT
- eine Erweiterung für den TwinCAT System Manager zur Konfiguration von OPC UA Pub/Sub
- eine Erweiterung für den TwinCAT System Manager zum Scannen entfernter OPC-UA-Server-Geräte
- eine optionale Integration in den TwinCAT OPC UA Server (TF6100 TC3 OPC UA) zur Nutzung der Konfigurationsschnittstelle auf Basis des standardisierten OPC-UA-Pub/Sub-Informationsmodells.

3 Installation

3.1 Systemanforderungen

Technische Daten	Beschreibung
Betriebssystem	Windows 10, Windows 11 (TwinCAT 3.1 Build >=4026.16)
Zielpattform	PC-Architektur (x86,x64)
Minimale TwinCAT-Version	TwinCAT 3.1 Build 4026
Erforderliches TwinCAT-Setup-Level	TwinCAT 3 XAE, XAR
Benötigte TwinCAT-Lizenz	TF6105 TC3 OPC UA Pub/Sub
Hardwareanforderungen	Dieses Produkt bietet Unterstützung für verschiedene Transportprotokolle. Wenn Sie UDP verwenden möchten, benötigen Sie eine Netzwerkinterfacekarte, die mit dem Treiber des TwinCAT Realtime-Ethernet Adapters kompatibel ist, siehe unten. Wenn Sie MQTT verwenden möchten, ist keine spezielle Netzwerkinterfacekarte erforderlich.

Realtime-Ethernet Adapter

Für die UDP-Kommunikation basiert dieses Produkt auf dem Treiber des TwinCAT Realtime-Ethernet Adapters. Um zu überprüfen, ob Ihre Netzwerkinterfacekarte für diesen Treiber geeignet ist, führen Sie bitte die folgenden Schritte aus. Bitte beachten Sie, dass die Installation dieses Treibers nicht erforderlich ist, wenn Sie nur OPC UA Pub/Sub über MQTT verwenden möchten.

1. Erstellen Sie ein OPC UA RT Gerät, indem Sie mit der rechten Maustaste auf **Devices** klicken und **Add New Item** wählen.
2. Klappen Sie die Kategorie OPC UA auf und wählen Sie **Real-Time OPC UA Device**.
3. Wählen Sie nun den Adapter aus und suchen Sie die entsprechende Ethernet-Schnittstelle, indem Sie **Compatible Devices** auswählen.
4. Versuchen Sie, den Treiber des TwinCAT RT-Ethernet Adapters für Ihre Netzwerkinterfacekarte zu installieren.

i "Nur für Demonstrationszwecke"

Wenn Ihre Netzwerkinterfacekarte in der Kategorie "Nur für Demonstrationszwecke" aufgeführt ist, wurde der Treiber des TwinCAT RT-Ethernet Adapters zwar installiert, bietet aber möglicherweise keine Echtzeitfunktionen. Sie können es jedoch weiterhin zur Bewertung und für Tests verwenden.

3.2 Installation

Wenn Sie TwinCAT 3.1 Build 4026 (und höher) auf dem Windows-Betriebssystem verwenden, können Sie diese Funktion über den TwinCAT Package Manager installieren, siehe [Dokumentation zur Installation](#). Normalerweise installieren Sie die Funktion über den entsprechenden Workload. Sie können die im Workload enthaltenen Pakete aber auch einzeln installieren. Diese Dokumentation beschreibt im Folgenden kurz den Installationsvorgang über den Workload.

Kommandozeilenprogramm TcPkg

Über das TcPkg Command Line Interface (CLI) können Sie sich die verfügbaren Workloads auf dem System anzeigen lassen:

```
tcpkg list -t workload
```

Über das folgende Kommando können Sie den Workload der Funktion installieren.

```
tcpkg install TF610x.OpcUaClientPubSub.XAE
tcpkg install TF610x.OpcUaClientPubSub.XAR
```

TwinCAT Package Manager UI

Über das User Interface (UI) können Sie sich alle verfügbaren Workloads anzeigen lassen und diese bei Bedarf installieren.

Folgen Sie hierzu den entsprechenden Anweisungen in der Oberfläche.

Die Installation des entsprechenden Pakets oder Workloads kann entweder über TcPkg CLI oder TcPkg UI erfolgen.



Unvorbereiteter TwinCAT-Neustart kann Datenverlust erzeugen

Die Installation dieser Funktion hat unter Umständen einen TwinCAT-Neustart zur Folge.

Stellen Sie sicher, dass keine kritischen TwinCAT-Applikationen auf dem System laufen oder fahren. Sie diese zunächst geordnet herunter.

3.3 Lizenzierung

Die TwinCAT 3 Function ist als Vollversion oder als 7-Tage-Testversion freischaltbar. Beide Lizenztypen sind über die TwinCAT-3-Entwicklungsumgebung (XAE) aktivierbar.

Lizenzierung der Vollversion einer TwinCAT 3 Function

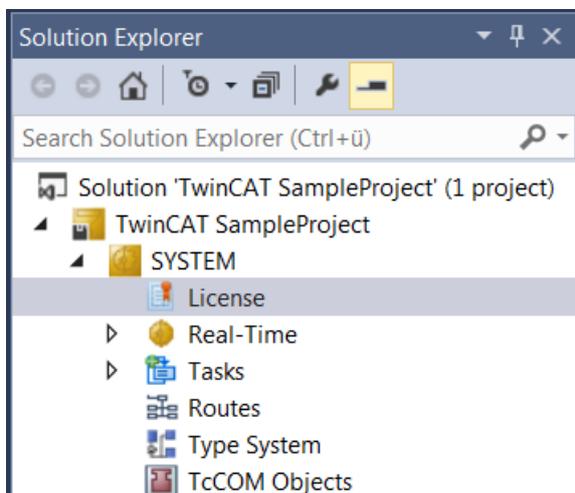
Die Beschreibung der Lizenzierung einer Vollversion finden Sie im Beckhoff Information System in der Dokumentation „[TwinCAT-3-Lizenzierung](#)“.

Lizenzierung der 7-Tage-Testversion einer TwinCAT 3 Function



Eine 7-Tage-Testversion kann nicht für einen [TwinCAT-3-Lizenz-Dongle](#) freigeschaltet werden.

1. Starten Sie die TwinCAT-3-Entwicklungsumgebung (XAE).
2. Öffnen Sie ein bestehendes TwinCAT-3-Projekt oder legen Sie ein neues Projekt an.
3. Wenn Sie die Lizenz für ein Remote-Gerät aktivieren wollen, stellen Sie das gewünschte Zielsystem ein. Wählen Sie dazu in der Symbolleiste in der Drop-down-Liste **Choose Target System** das Zielsystem aus.
 - ⇒ Die Lizenzierungseinstellungen beziehen sich immer auf das eingestellte Zielsystem. Mit der Aktivierung des Projekts auf dem Zielsystem werden automatisch auch die zugehörigen TwinCAT-3-Lizenzen auf dieses System kopiert.
4. Klicken Sie im **Solution Explorer** im Teilbaum **SYSTEM** doppelt auf **License**.



⇒ Der TwinCAT-3-Lizenzmanager öffnet sich.

- Öffnen Sie die Registerkarte **Manage Licenses**. Aktivieren Sie in der Spalte **Add License** das Auswahlkästchen für die Lizenz, die Sie Ihrem Projekt hinzufügen möchten (z. B. „TF4100 TC3 Controller Toolbox“).

Order No	License	Add License
TF3601	TC3 Condition Monitoring Level 2	<input type="checkbox"/> cpu license
TF3650	TC3 Power Monitoring	<input type="checkbox"/> cpu license
TF3680	TC3 Filter	<input type="checkbox"/> cpu license
TF3800	TC3 Machine Learning Inference Engine	<input type="checkbox"/> cpu license
TF3810	TC3 Neural Network Inference Engine	<input type="checkbox"/> cpu license
TF3900	TC3 Solar-Position-Algorithm	<input type="checkbox"/> cpu license
TF4100	TC3 Controller Toolbox	<input checked="" type="checkbox"/> cpu license
TF4110	TC3 Temperature-Controller	<input type="checkbox"/> cpu license
TF4500	TC3 Speech	<input type="checkbox"/> cpu license

- Öffnen Sie die Registerkarte **Order Information (Runtime)**.
 ⇒ In der tabellarischen Übersicht der Lizenzen wird die zuvor ausgewählte Lizenz mit dem Status „missing“ angezeigt.
- Klicken Sie auf **7 Days Trial License...**, um die 7-Tage-Testlizenz zu aktivieren.

- Es öffnet sich ein Dialog, der Sie auffordert, den im Dialog angezeigten Sicherheitscode einzugeben.

- Geben Sie den Code genauso ein, wie er angezeigt wird, und bestätigen Sie ihn.
- Bestätigen Sie den nachfolgenden Dialog, der Sie auf die erfolgreiche Aktivierung hinweist.

⇒ In der tabellarischen Übersicht der Lizenzen gibt der Lizenzstatus nun das Ablaufdatum der Lizenz an.

10. Starten Sie das TwinCAT-System neu.

⇒ Die 7-Tage-Testversion ist freigeschaltet.

4 Technische Einführung

4.1 Quick Start

Das folgende Kapitel führt Sie durch die erstmalige Einrichtung einer OPC UA Pub/Sub-Konfiguration in TwinCAT. Bitte stellen Sie sicher, dass die [Installationsanweisungen \[► 9\]](#) befolgt wurden, bevor Sie fortfahren, und dass Ihr System die [Systemanforderungen \[► 9\]](#) erfüllt. Die folgenden Quick Start Tutorials sind verfügbar:

Tutorial	Beschreibung
UDP [► 13]	Zeigt, wie man ein SPS-Projekt, ein OPC UA RT-Gerät und einen Publisher einrichtet, der ein DataSet mit einer Variablen über UDP sendet. Die Variable ist über das Prozessabbild mit einer Variablen aus dem SPS-Projekt verknüpft.
MQTT [► 17]	Zeigt, wie man ein SPS-Projekt, ein OPC UA RT-Gerät und einen Publisher einrichtet, der ein DataSet mit einer Variablen über MQTT sendet. Die Variable wird über den TwinCAT Target Browser mit einer Variablen aus dem SPS-Projekt verknüpft.

Offline-Konfigurationsaustausch

OPC UA Pub/Sub definiert ein Dateiformat für den Offline-Konfigurationsaustausch, das auf dem UA Binary Format basiert (OPC UA Spezifikation, Kapitel 5.2). Wenn Sie eine OPC UA Pub/Sub-Konfiguration zwischen zwei verschiedenen Geräten einrichten wollen, empfehlen wir dringend die Verwendung dieser gemeinsamen Austauschdatei. Das Kapitel [Konfiguration Import/Export \[► 36\]](#) enthält weitere Informationen zu diesem Thema.

4.1.1 UDP

Dieses Quick Start Tutorial zeigt, wie man ein SPS-Projekt, ein OPC UA RT Device und einen Publisher einrichtet, der ein DataSet mit einer Variablen über UDP sendet. Die Variable ist über das Prozessabbild mit einer Variablen aus dem SPS-Projekt verknüpft.

Vorbereitung des SPS-Projektes

Bereiten Sie ein SPS-Projekt vor, das eine Ausgangsvariable mit dem Datentyp INT definiert, z. B.:

```
PROGRAM MAIN
VAR
  nCounter AT%Q* : INT;
END_VAR

nCounter := nCounter + 1;
```

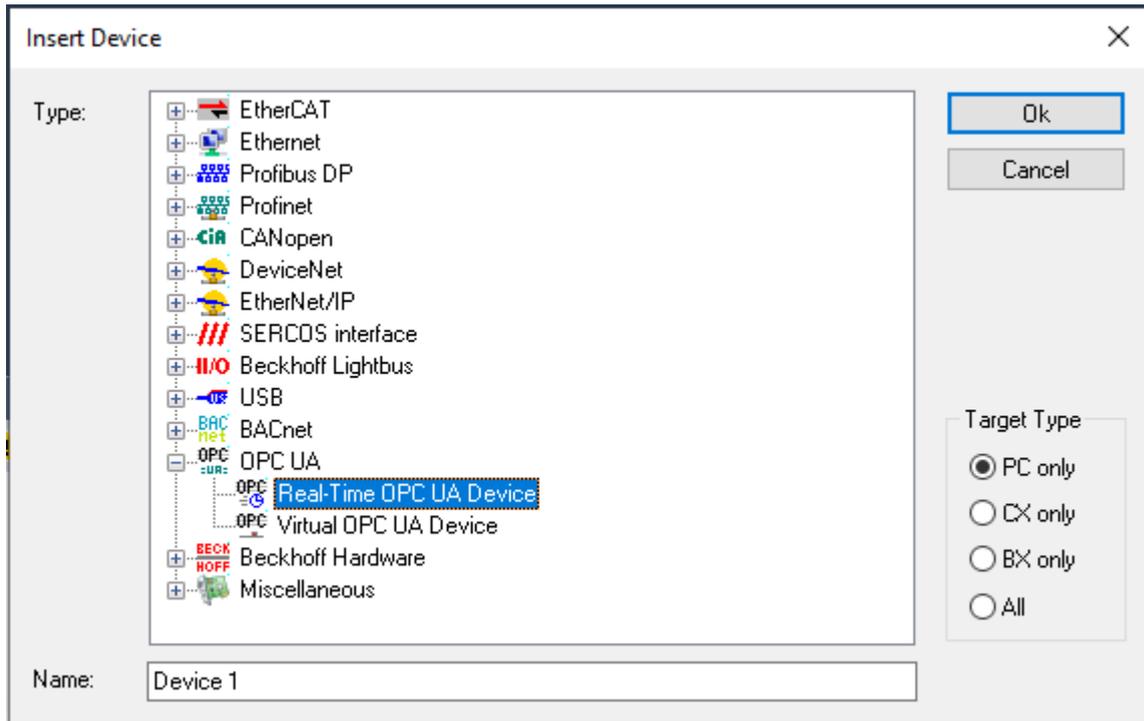
Kompilieren Sie das SPS-Projekt, damit das SPS-Prozessabbild erstellt wird.

Publisher konfigurieren

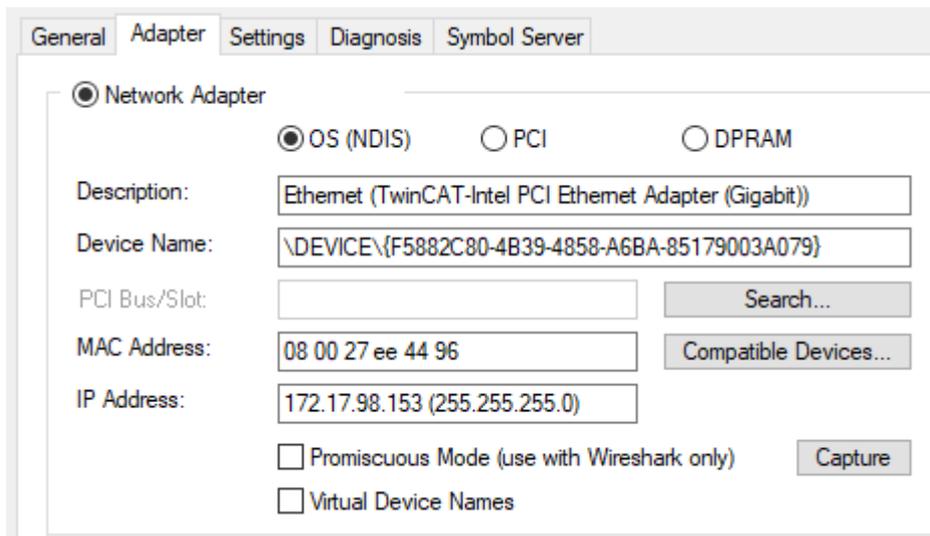
Bitte führen Sie die folgenden Konfigurationsschritte durch, um ein OPC UA Publisher-Gerät einzurichten:

1. Stellen Sie sicher, dass sich TwinCAT im Config-Modus befindet.
2. Erweitern Sie im Projektmappen-Explorer den Knoten I/O und wählen Sie das Kindelement Devices.
3. Nach einem Rechtsklick auf Devices **Add new element...** auswählen

4. Erweitern Sie die Kategorie **OPC UA**, wählen Sie das **Real-Time OPC UA Device** und klicken Sie auf **Ok**.

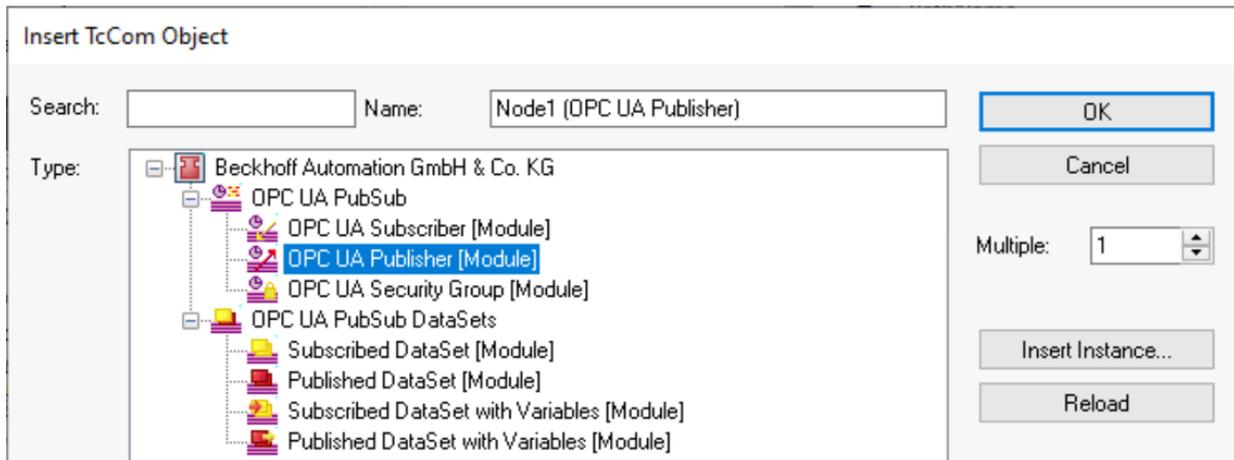


5. Doppelklicken Sie auf das hinzugefügte Gerät und navigieren Sie zur Registerkarte **Adapter**.

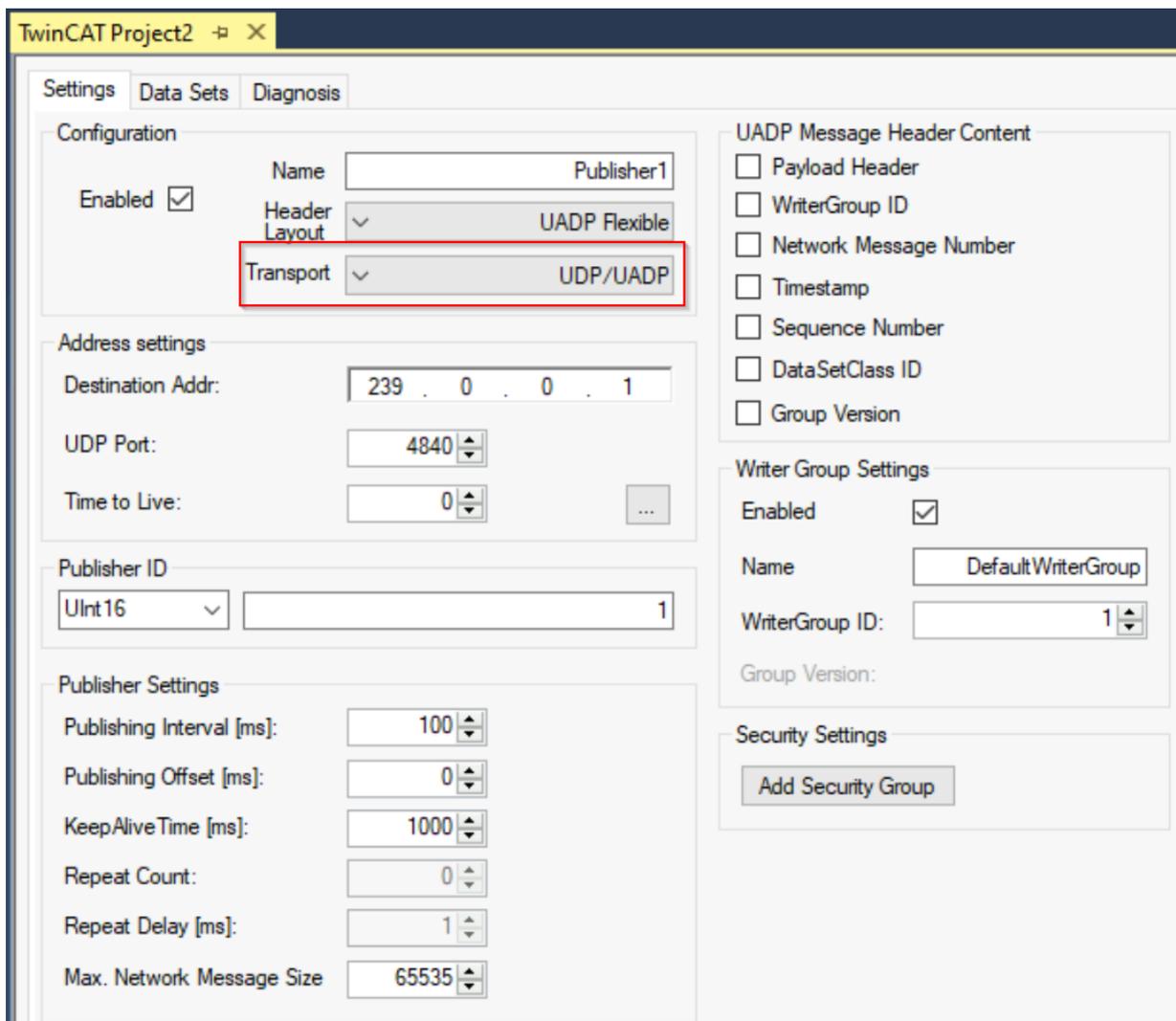


6. Binden Sie das Gerät an eine Ihrer Netzwerkkarten. Stellen Sie sicher, dass Sie den Treiber des TwinCAT-Intel PCI Ethernet Adapters auf dieser Netzwerkkarte installiert haben.
7. Nach einem Rechtsklick auf das Gerät **Add new element...** auswählen.

8. Wählen Sie den Eintrag **OPC UA Publisher (Module)** und klicken Sie auf **Ok**.

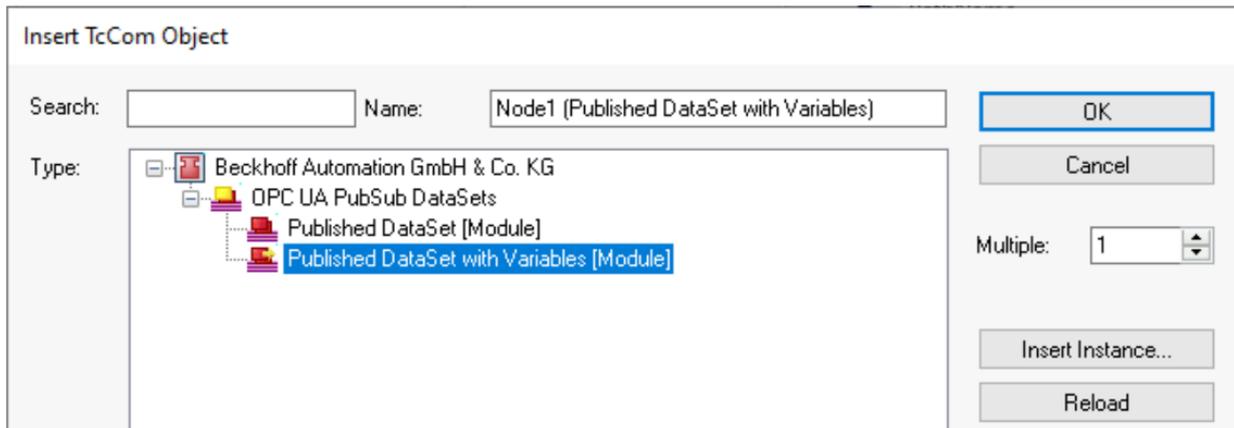


9. Um den hinzugefügten **Publisher**-Knoten zu konfigurieren, doppelklicken Sie ihn einfach in der Strukturansicht. Es gibt viele Konfigurationsoptionen für den Publisher, z. B. den zu verwendenden Transport (UDP, MQTT), das Datenformat (binär, JSON), optionale Header-Felder und so weiter. Wir wollen diese Einstellungen vorerst auf ihren Standardwerten belassen. Stellen Sie nur sicher, dass der Transport **UDP/UADP** ausgewählt ist.



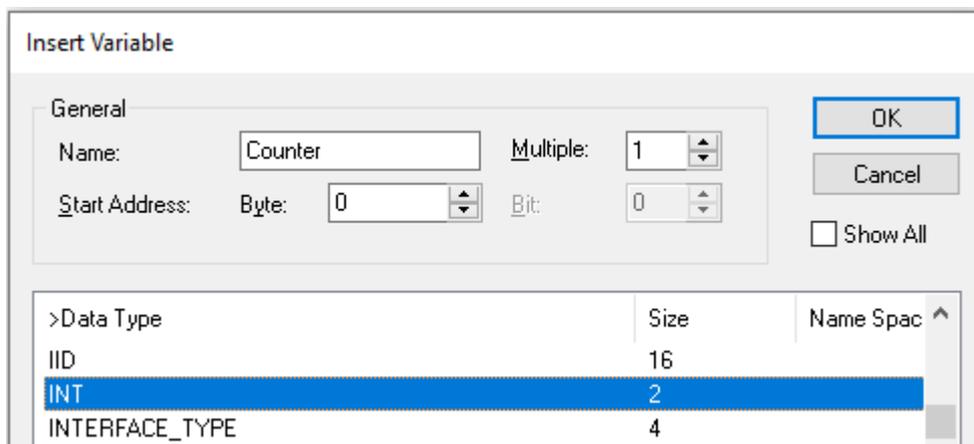
10. Im nächsten Schritt wollen Sie ein sogenanntes DataSet konfigurieren, um Variablen zu definieren, die vom Publisher versendet werden sollen. Nach Rechtsklick auf den **Publisher**-Knoten **Add new element...** auswählen.

11. Wählen Sie den Eintrag **Published DataSet with Variables (Module)** und klicken Sie auf **Ok**.

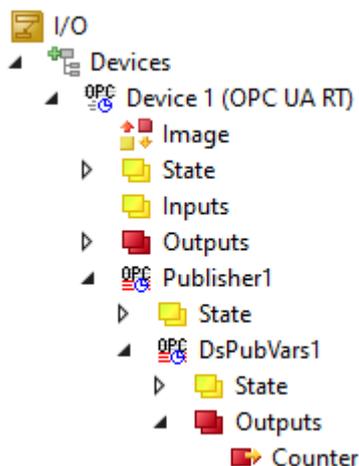


12. Rechtsklicken Sie in dem hinzugefügten DataSet auf den Knoten **Outputs** und wählen Sie **Add new element...** aus.

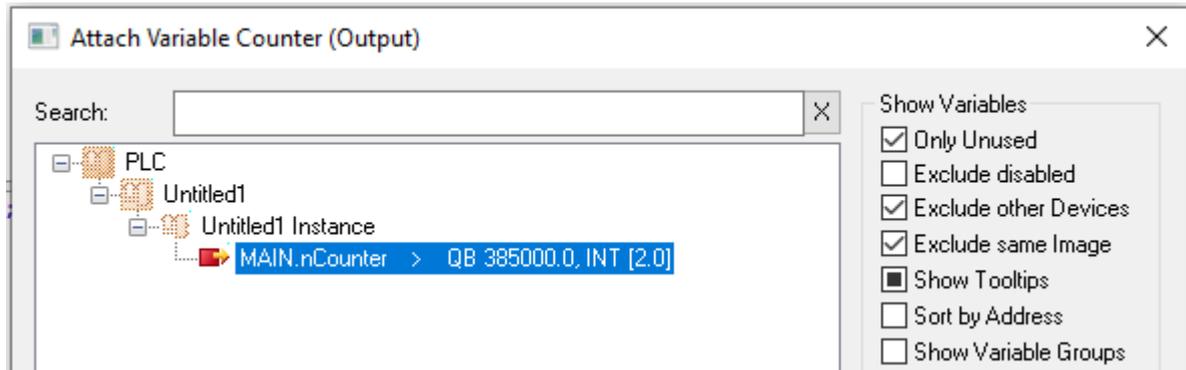
13. Geben Sie einen Namen für die Prozessvariable ein, z. B. **Counter**, und wählen Sie **INT** als Datentyp. Klicken Sie auf **Ok**, um die Variable zum Prozessabbild des DataSets hinzuzufügen.



⇒ Ihre Konfiguration sollte nun wie folgt aussehen:



14. Die hinzugefügte Variable mit rechts anklicken und im Kontextmenü **Change Link...** auswählen. Wählen Sie die SPS-Variable, die Sie zuvor erstellt haben, und klicken Sie auf **Ok**.



Aktivieren Sie das Projekt

Aktivieren Sie das TwinCAT-Projekt, indem Sie auf die Schaltfläche **Activate** in der TwinCAT XAE-Symbolleiste klicken. Stellen Sie sicher, dass Sie das richtige Zielgerät ausgewählt haben (wir gehen davon aus, dass Sie Ihr lokales Gerät verwenden) und dass Sie die richtigen Real-Time-Einstellungen verwenden, damit Sie die TwinCAT Runtime auf Ihrem Gerät aktivieren können. Bitte konsultieren Sie die reguläre TwinCAT-Dokumentation, um weitere Informationen darüber zu erhalten, wie Sie eine TwinCAT-Konfiguration aktivieren und TwinCAT in den Run-Modus schalten.

In diesem Quick Start Tutorial haben wir UDP als Transportprotokoll verwendet. Sie können nun entweder einen OPC UA Pub/Sub-Subscriber mit den Daten verbinden oder auch einen Blick auf die Daten mit Wireshark werfen. Wenn Sie einen Subscriber anschließen wollen, empfehlen wir Ihnen dringend, [Konfiguration Import/Export \[► 36\]](#) zu nutzen.

4.1.2 MQTT

Dieses Quick Start Tutorial zeigt, wie man ein SPS-Projekt, ein OPC UA RT Device und einen Publisher einrichtet, der ein DataSet mit einer Variablen über MQTT sendet. Die Variable ist über das Prozessabbild mit einer Variablen aus dem SPS-Projekt verknüpft.

Vorbereitung des SPS-Projektes

Bereiten Sie ein SPS-Projekt vor, das eine Variable mit dem Datentyp INT definiert, z. B.:

```
PROGRAM MAIN
VAR
  nCounter : INT;
END_VAR

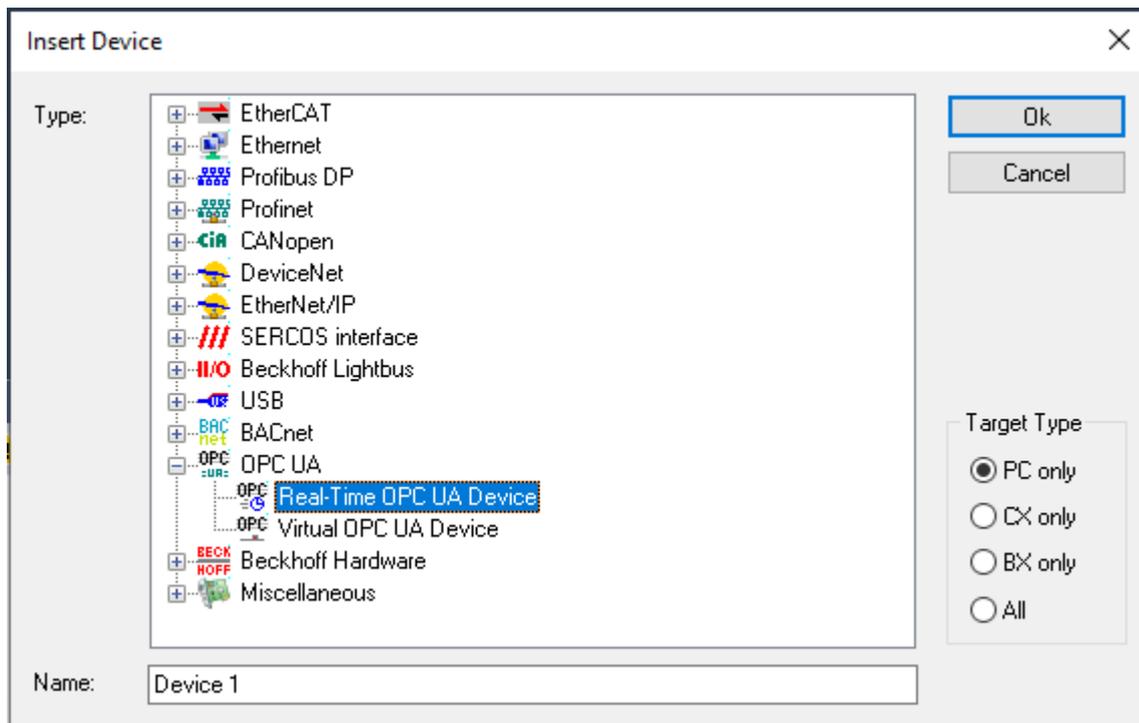
nCounter := nCounter + 1;
```

Bitte beachten Sie, dass Sie keine Input/Output-Compiler-Anweisung setzen müssen, da wir das Prozessabbild nicht zur Verknüpfung der Variablen verwenden werden. Aktivieren Sie diese Konfiguration und starten Sie das SPS-Programm.

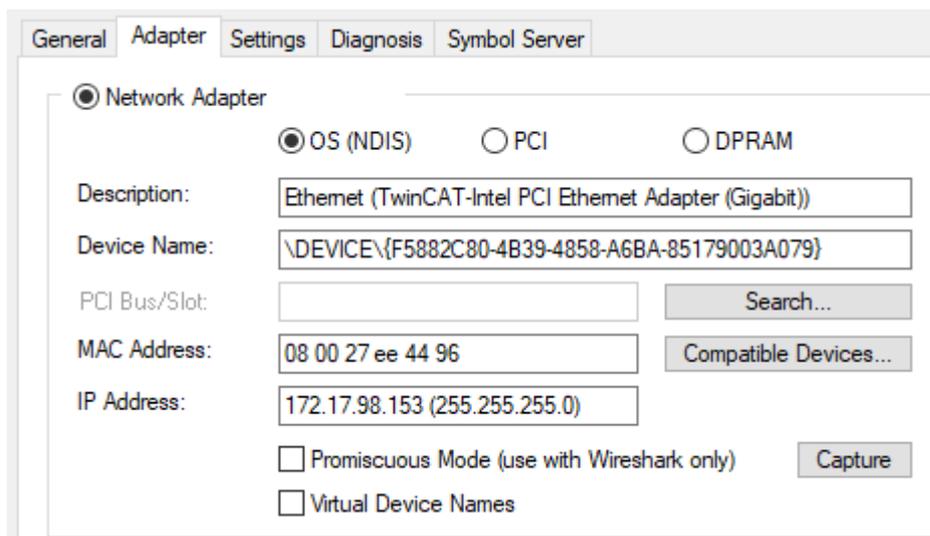
Publisher konfigurieren

Bitte führen Sie die folgenden Konfigurationsschritte durch, um ein OPC UA Publisher-Gerät einzurichten:

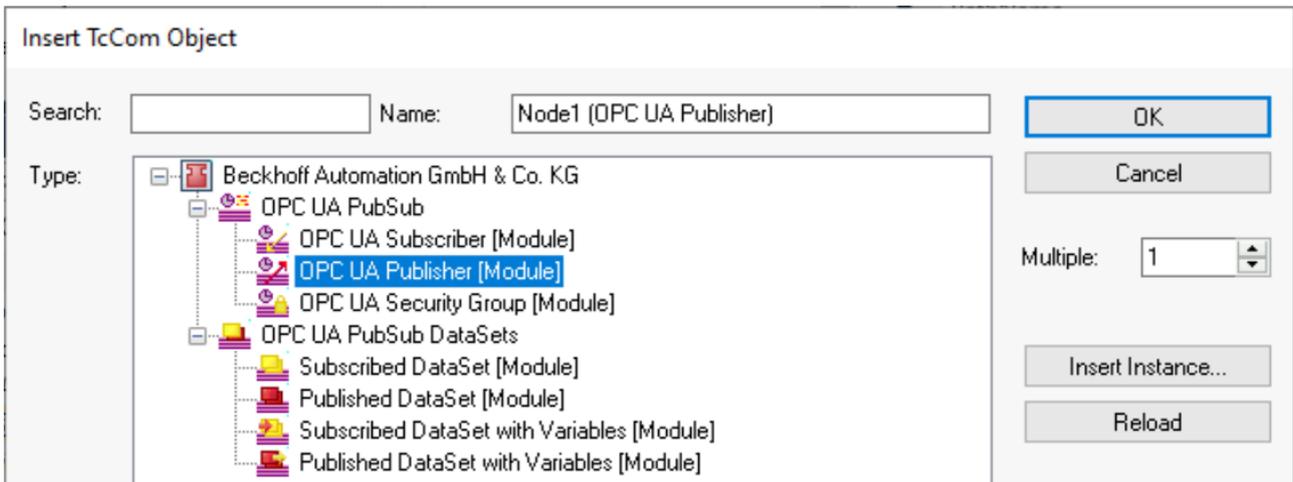
- Stellen Sie sicher, dass sich TwinCAT im Config-Modus befindet.
- Erweitern Sie im Projektmappen-Explorer den Knoten I/O und wählen Sie das Kindelement **Devices** aus.
- Nach einem Rechtsklick auf **Devices Add new element...** auswählen.
- Erweitern Sie die Kategorie **OPC UA**, wählen Sie das **Real-Time OPC UA Device** und klicken Sie auf **Ok**.



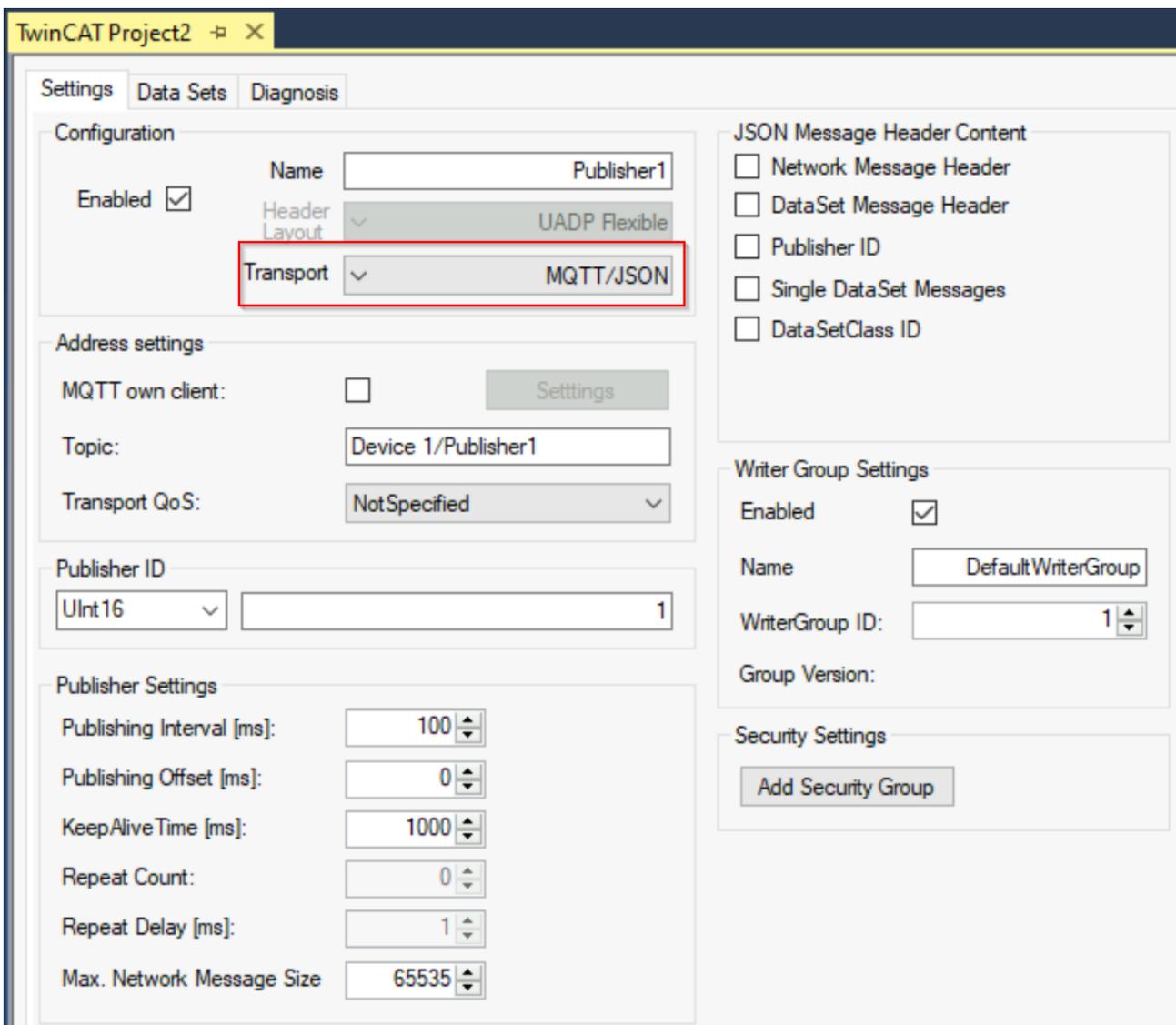
- Doppelklicken Sie auf das hinzugefügte Gerät und navigieren Sie zur Registerkarte **Adapter**.



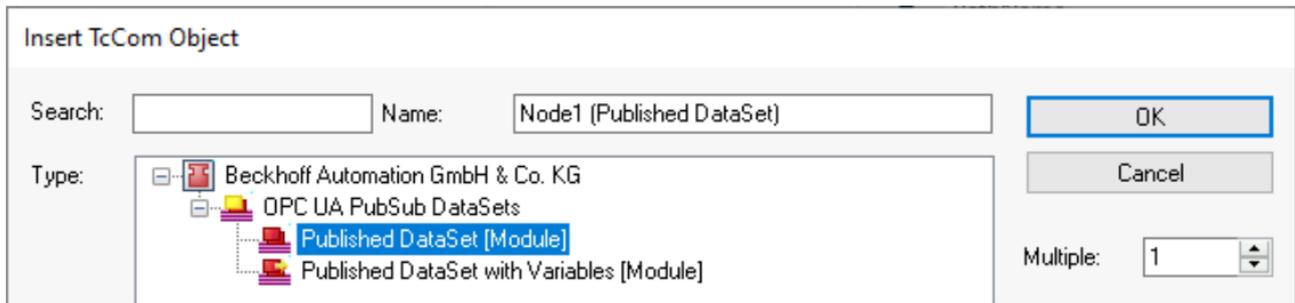
- Binden Sie das Gerät an eine Ihrer Netzwerkinterfacekarten. Stellen Sie sicher, dass Sie den Treiber des TwinCAT-Intel PCI Ethernet Adapters auf dieser Netzwerkinterfacekarte installiert haben.
- Nach einem Rechtsklick auf das Gerät **Add new element...** auswählen.
- Wählen Sie den Eintrag **OPC UA Publisher (Module)** und klicken Sie auf **Ok**.



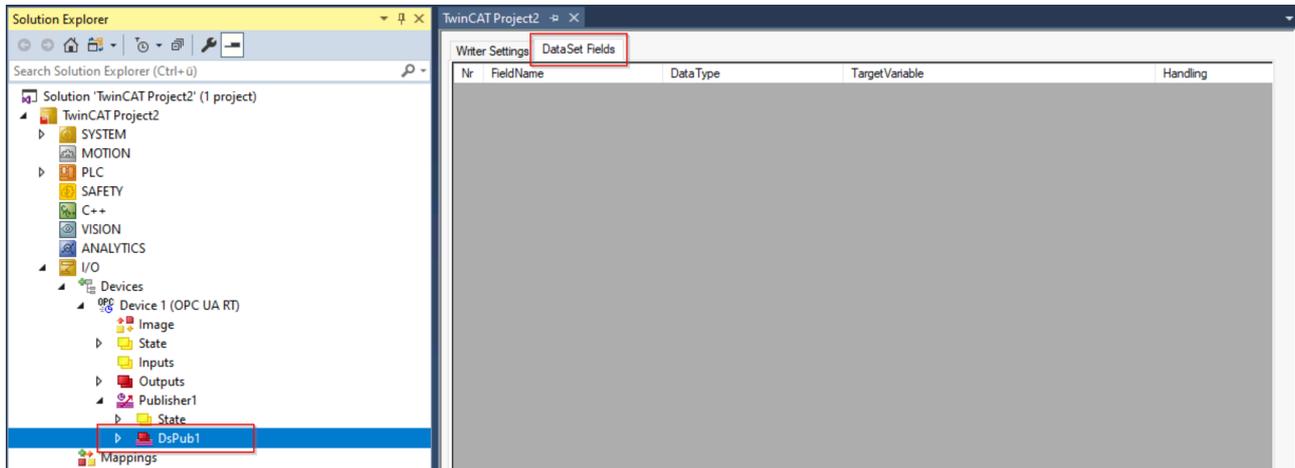
- Um den hinzugefügten **Publisher**-Knoten zu konfigurieren, doppelklicken Sie ihn einfach in der Strukturansicht. Es gibt viele Konfigurationsoptionen für den Publisher, z. B. den zu verwendenden Transport (UDP, MQTT), das Datenformat (binär, JSON), optionale Header-Felder und so weiter. Wir wollen diese Einstellungen vorerst auf ihren Standardwerten belassen und nur die Transporteinstellung auf **MQTT/JSON** ändern.



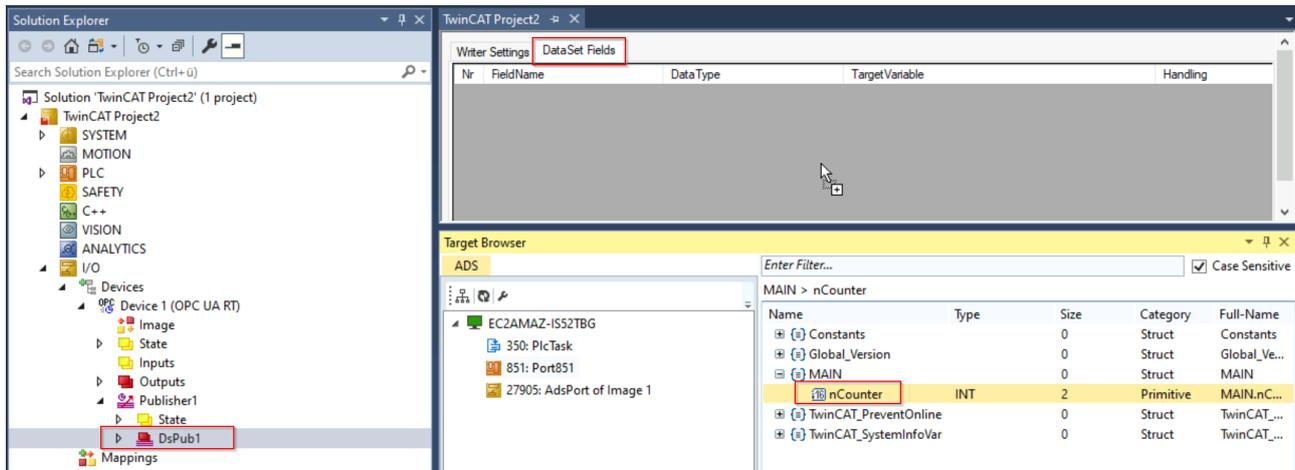
- Im nächsten Schritt wollen Sie ein sogenanntes DataSet konfigurieren, um Variablen zu definieren, die vom Publisher versendet werden sollen. Nach Rechtsklick auf den **Publisher**-Knoten **Add new element...** auswählen.
- Wählen Sie den Eintrag **Published DataSet (Module)** und klicken Sie auf **Ok**.



- Doppelklicken Sie auf den DataSet und öffnen Sie die Registerkarte **Data Set fields**.



- Öffnen Sie den TwinCAT Target Browser, navigieren Sie zum MAIN-Programm Ihres SPS-Projekts und ziehen Sie die Variable nCounter per Drag&Drop in die Liste der DataSet-Felder.



Aktivieren des Projekts

Aktivieren Sie das TwinCAT-Projekt, indem Sie auf die Schaltfläche **Activate** in der TwinCAT XAE-Symbolleiste klicken. Stellen Sie sicher, dass Sie das richtige Zielgerät ausgewählt haben (wir gehen davon aus, dass Sie Ihr lokales Gerät verwenden) und dass Sie die richtigen Real-Time-Einstellungen verwenden, damit Sie die TwinCAT Runtime auf Ihrem Gerät aktivieren können. Bitte konsultieren Sie die reguläre TwinCAT-Dokumentation, um weitere Informationen darüber zu erhalten, wie Sie eine TwinCAT-Konfiguration aktivieren und TwinCAT in den Run-Modus schalten.

In diesem Quick Start Tutorial haben wir MQTT als Transportprotokoll verwendet. Sie können nun jeden MQTT-Client, der Zugriff auf den Message Broker hat, verwenden, um das veröffentlichte DataSet zu abonnieren. Der folgende Screenshot zeigt das Kommandozeilentool **mosquitto_sub.exe**, das für die Daten abonniert wurde. Dieses Tool ist Teil der Softwareanwendung Mosquitto.

```

Administrator: PowerShell 7 (x64)
PS C:\Program Files\mosquitto> .\mosquitto_sub.exe -t "Device 1/Publisher1"
{
  "MessageId": "{00000002-FFFE-FFFF-FEFF-FFFFFFEFFFFFFF}",
  "MessageType": "ua-data",
  "Messages": [
    {
      "Payload": {
        "nCounter": {
          "Type": 4,
          "Body": 0
        }
      }
    }
  ]
}
{
  "MessageId": "{00000002-FFFE-FFFF-FEFF-FFFFFFEFFFFFFF}",
  "MessageType": "ua-data",
  "Messages": [
    {
      "Payload": {
        "nCounter": {
          "Type": 4,
          "Body": 0
        }
      }
    }
  ]
}

```

4.2 Unterstützte Features

OPC UA Pub/Sub verfügt über eine große Anzahl von Features. Es kann sein, dass dieses Produkt nicht alle Features von Anfang an unterstützt. Weitere Features werden im Laufe der Zeit durch Produktaktualisierungen hinzugefügt. Die folgende Tabelle zeigt eine Liste aller derzeit unterstützten Funktionalitäten.

● Kompatibilität zur OPC UA Pub/Sub Spezifikation



Die aktuelle Version dieses Produkts ist kompatibel zur OPC UA Pub/Sub Spezifikation Version 1.05.03.

Header-Layouts

Name	Unterstützt
Flexible	Ja
UADP Dynamic	Ja
UADP PeriodicFixed	Ja
JSON DataSetMessage	Ja
JSON Minimal	Ja
JSON NetworkMessage	Ja

UDP Transport

Feature	Unterstützt
Publisher	Ja
Subscriber	Ja
UADP chunking	Ja
IP-Fragmentierung	Ja
DataKey-Frames	Ja
Delta-Frames	Ja
KeepAlive	Ja
Timeouts-Handling bei Nachrichten	Ja
UADP raw encoding	Ja
UADP variant encoding	Ja
Pub/Sub-Sicherheitsnachrichten signieren	Noch nicht unterstützt
Pub/Sub-Sicherheitsnachrichtenverschlüsselung	Noch nicht unterstützt
Unterstützung für primitive Datentypen (INT, REAL, ...)	Ja
Unterstützung für Arrays von primitiven Datentypen (INT, REAL, ...)	Ja
Unterstützung für Datenstrukturen	Ja
Unterstützung für Arrays von Datenstrukturen	Ja
Unterstützung für Unions	Noch nicht unterstützt
Unterstützung für Varianten	Noch nicht unterstützt
Unterstützung für lokale DataSets	Ja
Unterstützung für vorkonfigurierte (globale) DataSets	Ja
Unterstützung für Unicast/Multicast	Ja
Unterstützung für die Konfiguration des UDP-Ports	Ja
Unterstützung für Ereignismeldungen	Noch nicht unterstützt
Unterstützung für feldspezifische Daten in DataSets (Delta-Deadband, Deadband-Typ, Ersatzwerte)	Noch nicht unterstützt
Unterstützung für Ethernet-VLAN-Transport	Noch nicht unterstützt
Unterstützung für MetaDaten-Dienste	Noch nicht unterstützt

MQTT Transport

Feature	Unterstützt
Publisher	Ja
Subscriber	Ja
Key-Frames	Ja
Delta-Frames UADP	Ja
Delta-Frames JSON	Noch nicht unterstützt
KeepAlive	Noch nicht unterstützt
Unterstützung für MQTT Version 3.1.1	Ja
Unterstützung für MQTT Version 5.0	Ja
Unterstützung für MQTT / TLS Versionen 1.2 und 1.3	Ja
JSON-Kodierung (umkehrbar)	Ja
JSON-Kodierung (nicht umkehrbar)	Ja
UADP-Kodierung	Ja
Unterstützung für Ereignismeldungen	Noch nicht unterstützt
Unterstützung für primitive Datentypen (INT, REAL, ...)	Ja
Unterstützung für Arrays von primitiven Datentypen (INT, REAL, ...)	Ja
Unterstützung für Datenstrukturen	Ja
Unterstützung für Arrays von Datenstrukturen	Ja
Unterstützung für Unions	Noch nicht unterstützt
Unterstützung für Varianten	Noch nicht unterstützt
Unterstützung für lokale DataSets	Ja
Unterstützung für globale DataSets	Ja
Unterstützung für MetaDaten-Dienste	Noch nicht unterstützt

Unterstützte Message Broker

Unser TwinCAT-MQTT-Protokolltreiber unterstützt die MQTT-Versionen 3.1.1 und 5.0. Daher werden alle Message Broker unterstützt, die diese Versionen ebenfalls unterstützen. Wir haben die Kommunikation mit Mosquitto, HiveMQ und AWS IoT Core erfolgreich getestet.

OPC UA Pub/Sub Sicherheit

Feature	Unterstützt
Pre-Shared-Key (PSK)	Noch nicht unterstützt
Local SKS	Noch nicht unterstützt
Remote SKS	Noch nicht unterstützt

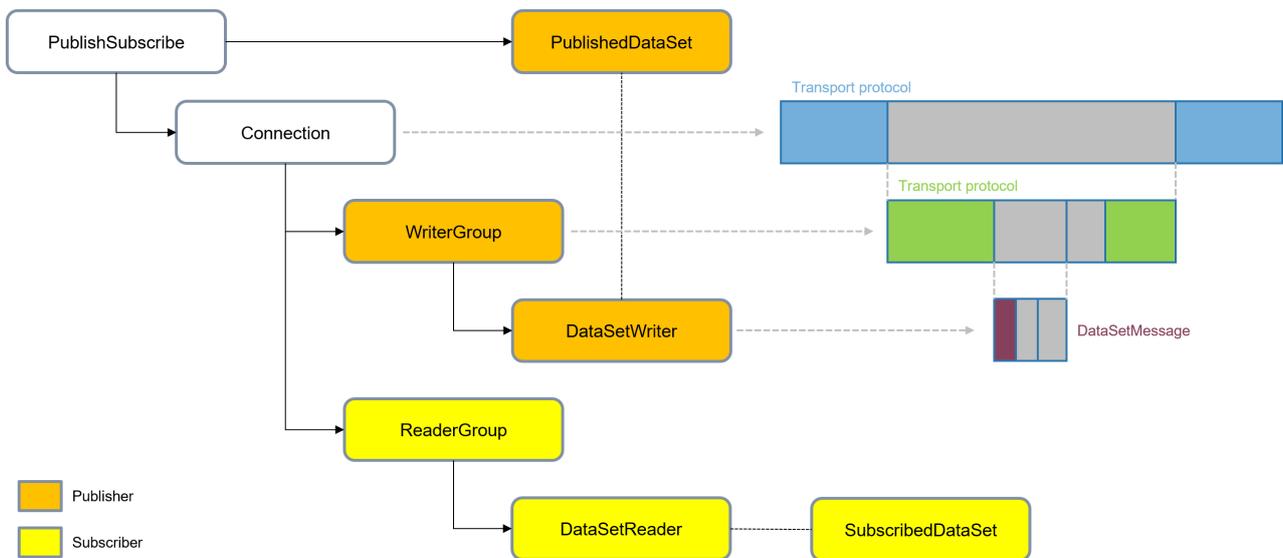
Handhabung der Konfiguration

Feature	Unterstützt
Offline-Konfigurationsaustausch (UA-Binary)	Ja
Offline-Konfigurationsaustausch (read)	Noch nicht unterstützt
Offline-Konfigurationsaustausch (read)	Noch nicht unterstützt

4.3 Protokollübersicht

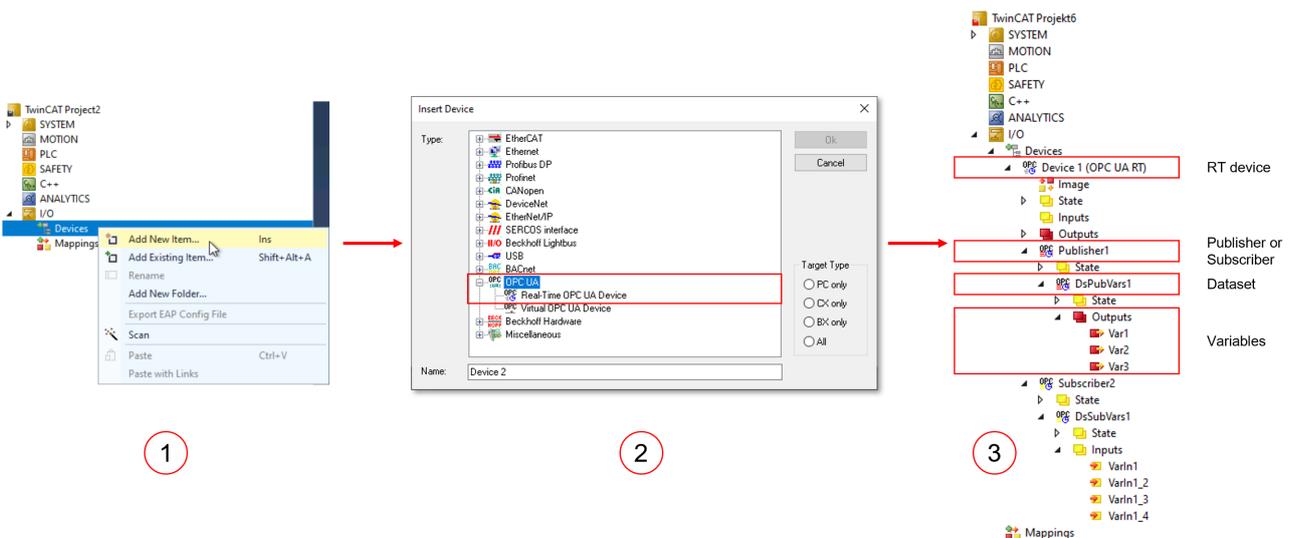
OPC UA Pub/Sub definiert unterschiedliche Konfigurationsparameter für die verschiedenen Komponenten. Sie definieren das Verhalten von Publisher und Subscriber. Die Konfiguration dieser Parameter kann über das OPC-UA-Informationsmodell für PubSub oder über herstellerspezifische Mechanismen erfolgen, d. h. im Falle des Produkts TF6105 TC3 OPC UA Pub/Sub über die TwinCAT-XAE-Umgebung.

Das folgende Diagramm zeigt die verschiedenen Konfigurationskomponenten und ihre Beziehung zueinander. Die Komponenten WriterGroup, DataSetWriter und PublishedDataSet definieren den Publisher, während ReaderGroup, DataSetReader und SubscribedDataSet den Subscriber definieren.



Für detailliertere Informationen zu den einzelnen Komponenten empfehlen wir, die OPC UA Spezifikation, Teil 14 (Pub/Sub) zu konsultieren.

Bei der Einrichtung der Konfiguration in der TwinCAT-XAE-Umgebung kann der Engineering-Workflow wie folgt visualisiert werden:



Schritt	Beschreibung
1	Ein neues Gerät wird zur TwinCAT-I/O-Konfiguration hinzugefügt.
2	Aus der Geräteliste können Sie dann das "Real-Time OPC UA Device" auswählen, das den Einstiegspunkt für alle OPC UA Pub/Sub-Einstellungen darstellt.
3	Publisher, Subscriber, DataSets und Variablen werden ebenso hinzugefügt wie die entsprechenden Parameter.

Die folgenden Kapitel geben einen Überblick über die verschiedenen Komponenten und verweisen auf ausführlichere Dokumentationsartikel.

Device

Das OPC UA RT (Realtime) Device ist der Einstiegspunkt in die OPC UA Pub/Sub-Konfiguration und wird mit einer Netzwerkkarte verbunden, die auf dem Treiber des TwinCAT Realtime-Ethernet Adapters basiert. Das Device kann einen oder mehrere Publisher und/oder Subscriber sowie (globale) DataSets enthalten.

Publisher/Subscriber

Ein Publisher- oder Subscriber-Knoten definiert, ob die angeschlossenen DataSets entweder gesendet (veröffentlicht) oder empfangen (abonniert) werden. Jeder [Publisher/Subscriber \[► 25\]](#)-Knoten enthält Adressinformationen für das zu verwendende [Transportprotokoll \[► 32\]](#).

DataSet

DataSets können in zwei verschiedenen Bereichen angegeben werden. Sie können einen DataSet entweder lokal auf einem Publisher oder Subscriber oder global auf dem Device definieren. Globale DataSets können von mehreren Publishern oder Subscribern gemeinsam genutzt werden. Es gibt zwei verschiedene Arten von DataSets, die festlegen, wie Variablen erstellt und behandelt werden:

- DataSet mit Variablen: Dieser DataSet-Typ kann mit Variablen konfiguriert werden, die im Prozessabbild des DataSets erscheinen. Die Variablen können dann mit anderen Prozessabbildvariablen verknüpft werden.
- DataSet ohne Variablen: Dieser DataSet-Typ kann mit Variablen über den TwinCAT Target Browser konfiguriert werden. Die Variablen werden dann automatisch intern verknüpft und erscheinen nicht im Prozessabbild.

Weitere Informationen finden Sie in unserem Dokumentationsartikel über [DataSets \[► 27\]](#).

Variablen

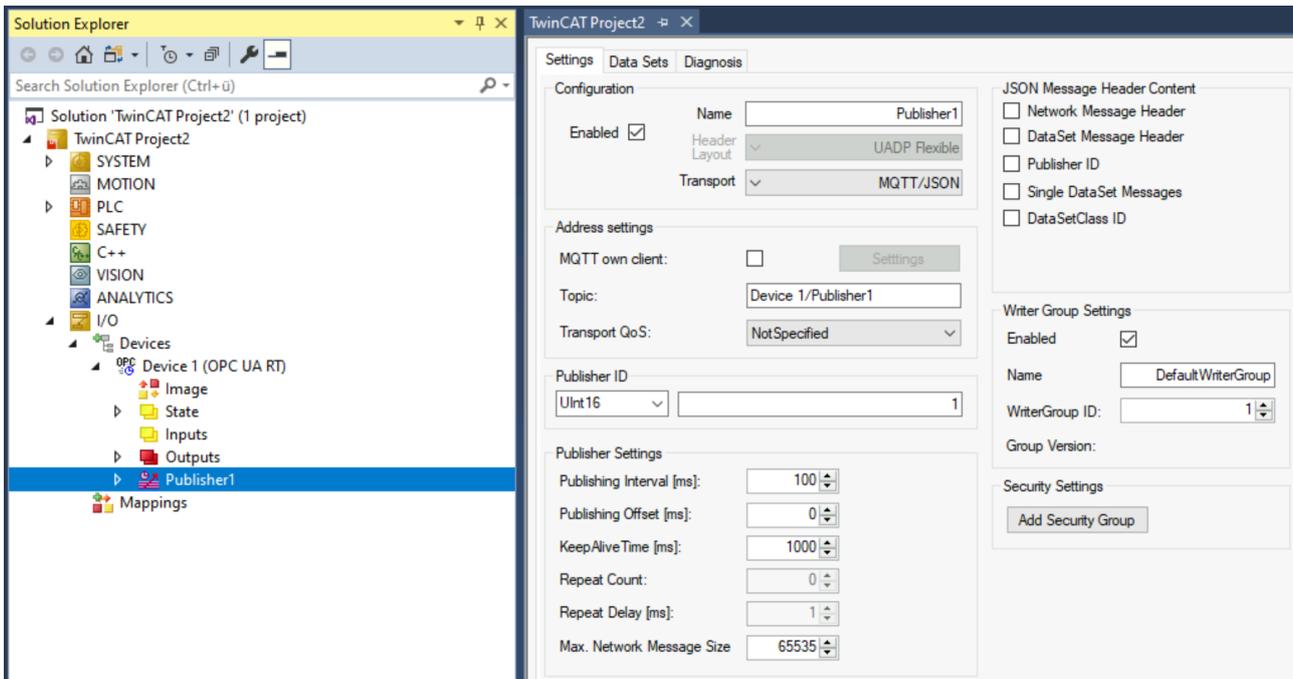
Variablen werden dem DataSet als so genannte "DataSet-Felder" hinzugefügt. Sie können entweder mit anderen Prozessabbildvariablen verknüpft werden oder werden über den TwinCAT Target Browser direkt mit anderen Variablen verbunden. Weitere Informationen finden Sie in unserem Dokumentationsartikel über [DataSets \[► 27\]](#).

4.4 Publisher/Subscriber

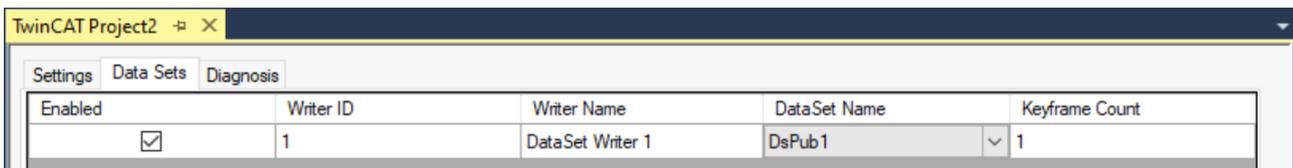
OPC UA Pub/Sub basiert auf dem Kommunikationsmuster Publisher/Subscriber. TF6105 TC3 OPC UA Pub/Sub enthält verschiedene Konfigurationsmechanismen zur Einrichtung eines Publishers oder Subscribers einschließlich der entsprechenden [DataSets \[► 27\]](#). Wir empfehlen auch, das Kapitel [Protokollübersicht \[► 23\]](#) zu konsultieren, um einen Überblick über die verschiedenen Konfigurationskomponenten zu erhalten.

Publisher

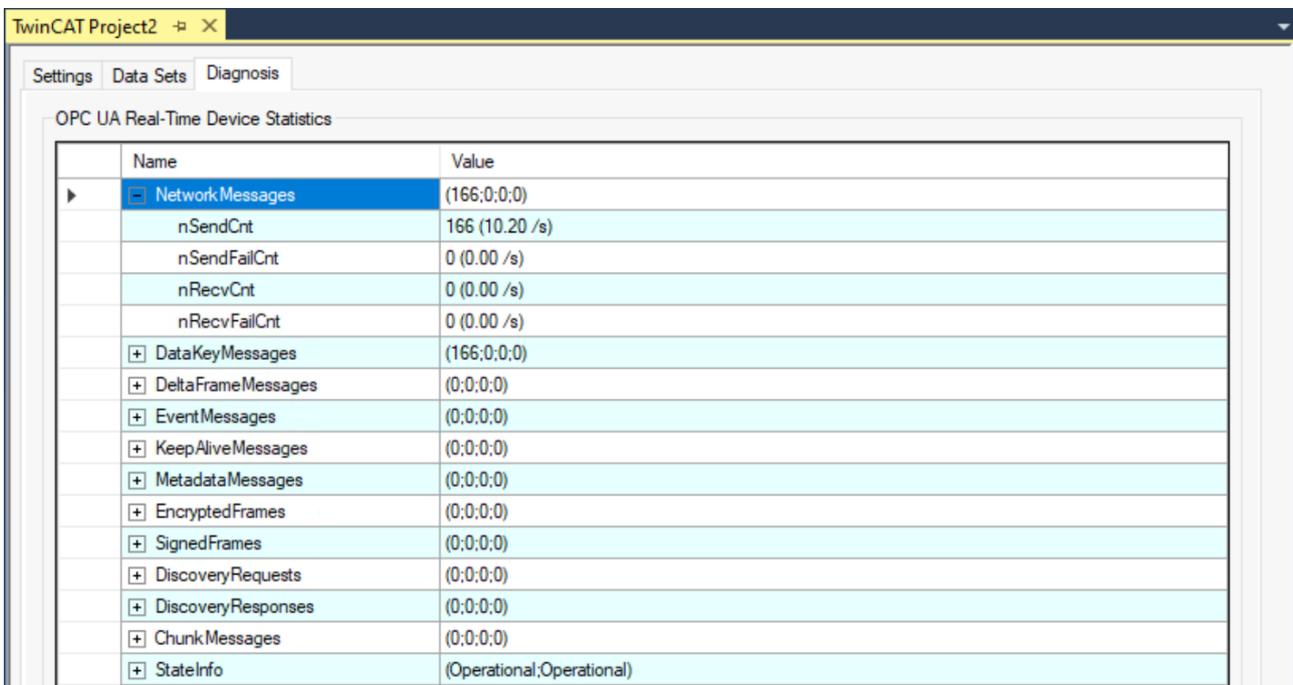
Mit dem Publisher kann die Kommunikation zum Senden von Daten eingerichtet werden. Er wird direkt zum OPC UA RT Device hinzugefügt. Die Einstellungsseite enthält verschiedene Konfigurationsparameter zur Festlegung des [Transportprotokolls \[► 32\]](#) sowie zusätzliche Einstellungen, die die Datenkommunikation weiter definieren.



Die Registerkarte **Data Sets** enthält Informationen über alle verknüpften lokalen oder globalen DataSets.

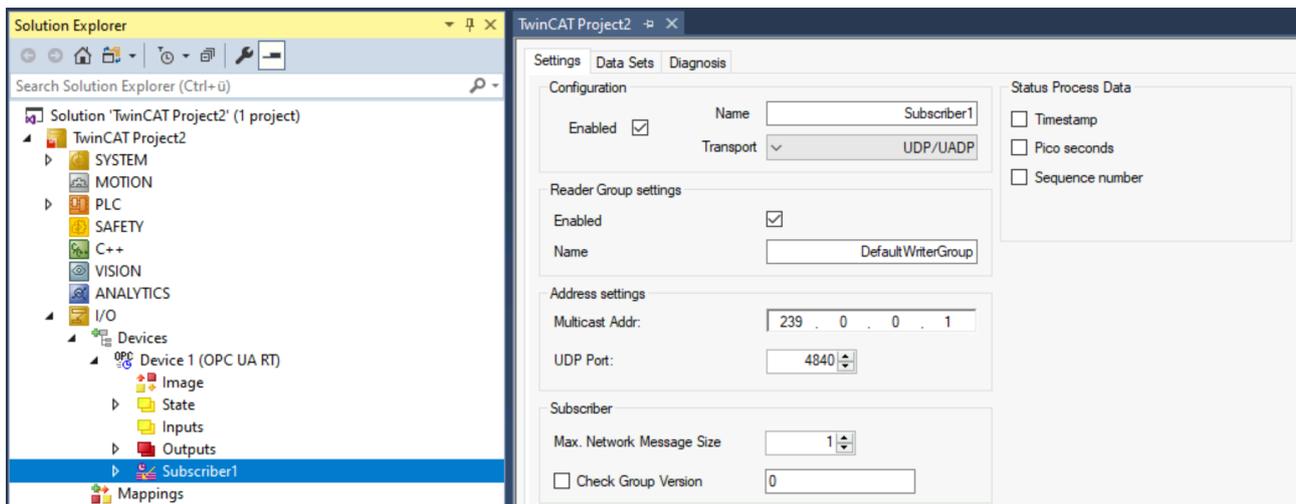


Die Registerkarte **Diagnosis** enthält Diagnoseinformationen zur Datenkommunikation.



Subscriber

Mit dem Publisher kann die Kommunikation für den Datenempfang eingerichtet werden. Er wird direkt zum OPC UA RT Device hinzugefügt. Die Einstellungsseite enthält verschiedene Konfigurationsparameter zur Festlegung des Transportprotokolls [► 32] sowie zusätzliche Einstellungen, die die Datenkommunikation weiter definieren.



Ähnlich wie beim Publisher enthält die Registerkarte **Data Sets** Informationen über alle verknüpften lokalen oder globalen DataSets und die Registerkarte **Diagnosis** enthält Diagnoseinformationen über die Datenkommunikation.

4.5 DataSets

DataSets können zu einem Publisher und Subscriber hinzugefügt werden und definieren eine Liste von Namens- und Wertepaaren, die eine Liste von Variablenwerten darstellen. Solche Variablen werden dann durch so genannte DataSet-Felder dargestellt. Einem DataSet können Metadaten hinzugefügt werden, die zusätzliche Informationen über die Struktur und den Inhalt eines DataSets liefern.

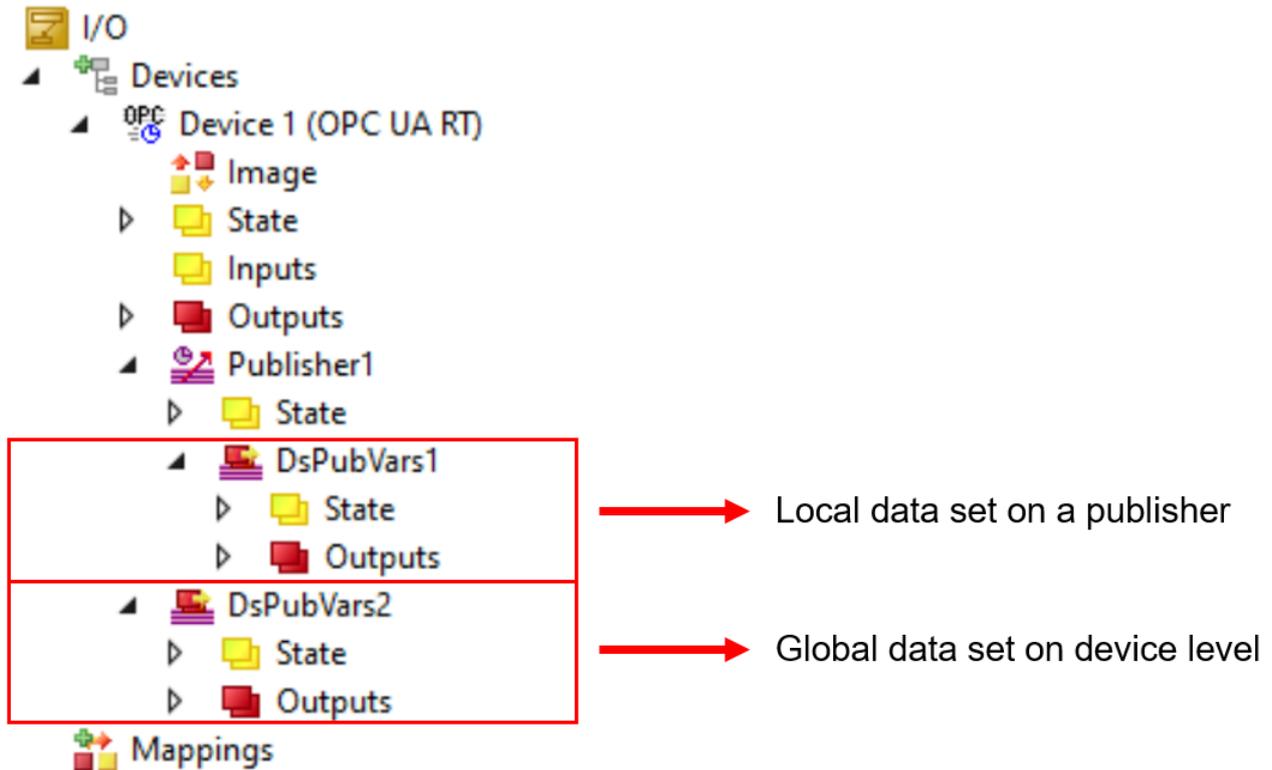
TF6105 TC3 OPC UA Pub/Sub stellt zwei verschiedene Arten von DataSets zur Verfügung, die definieren, wie Variablen konfiguriert und behandelt werden.

- DataSet mit Variablen
- DataSet ohne Variablen

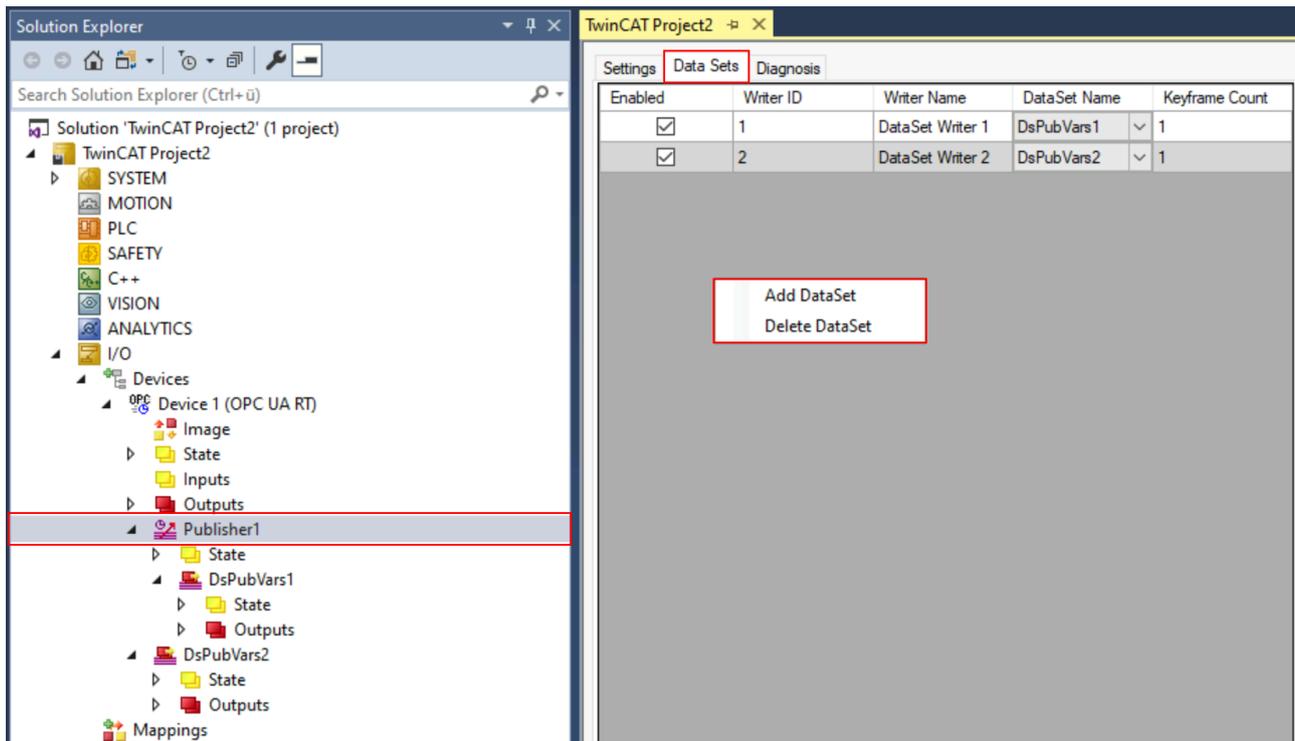
Diese DataSet-Typen werden im Folgenden ausführlicher beschrieben.

Scope

DataSets können in zwei verschiedenen Bereichen konfiguriert werden: lokal oder global. Lokale DataSets werden ausschließlich für einen einzelnen Publisher oder Subscriber konfiguriert, während globale DataSets auf Geräteebeane hinzugefügt werden und von Publishern oder Subscribern gemeinsam genutzt werden können, z. B. wenn Sie dieselben Datenpunkte auf zwei verschiedenen Publishern veröffentlichen möchten.



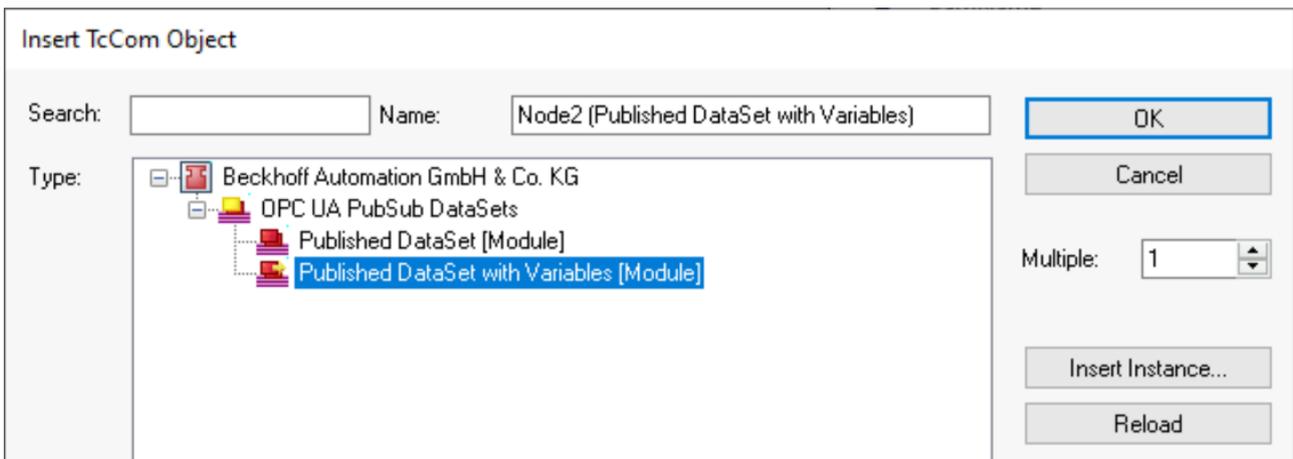
Globale DataSets können einem Publisher/Subscriber über die Registerkarte **Data Sets** hinzugefügt werden:



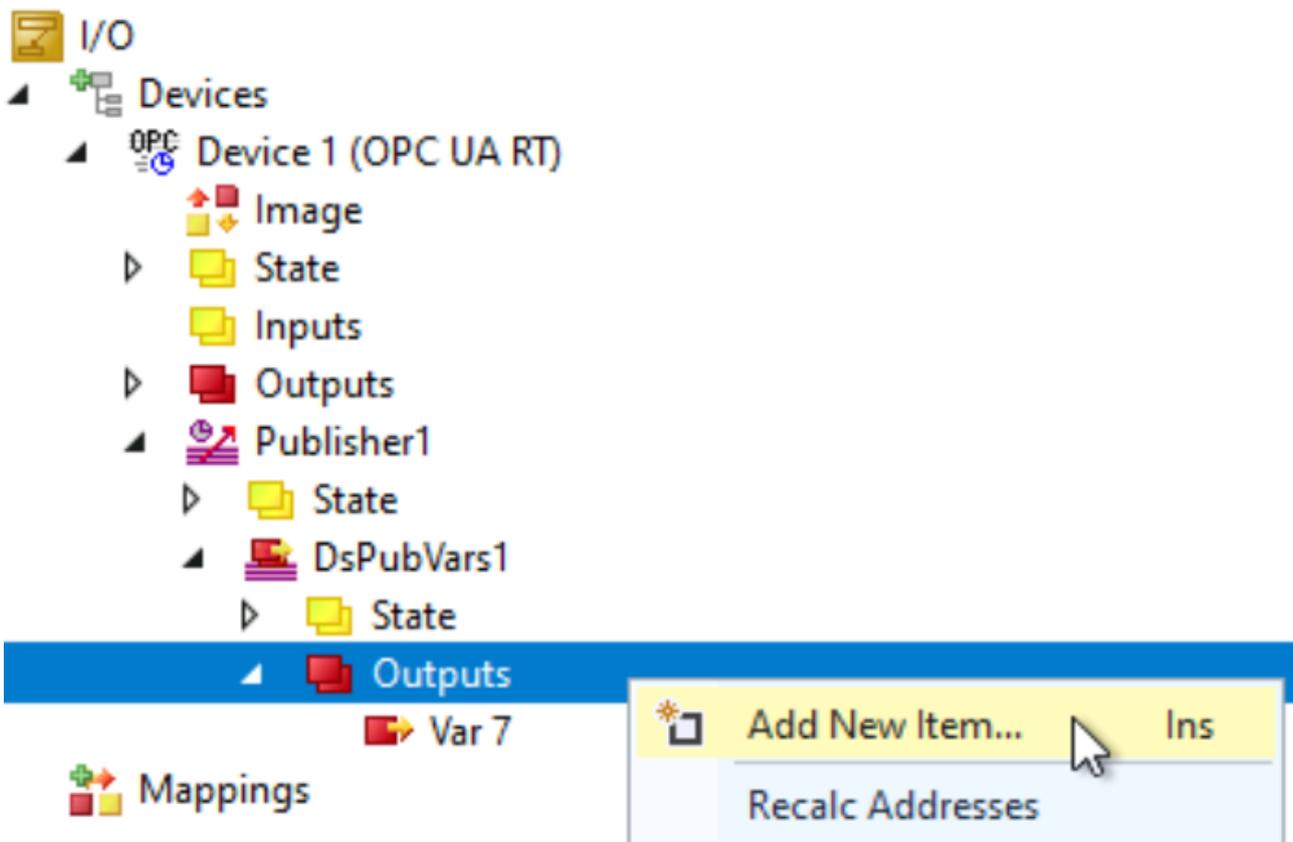
Im obigen Beispiel wurde der Publisher mit einem lokalen DataSet namens "DsPubVars1" und einem globalen Satz namens "DsPubVars2" konfiguriert.

DataSet mit Variablen

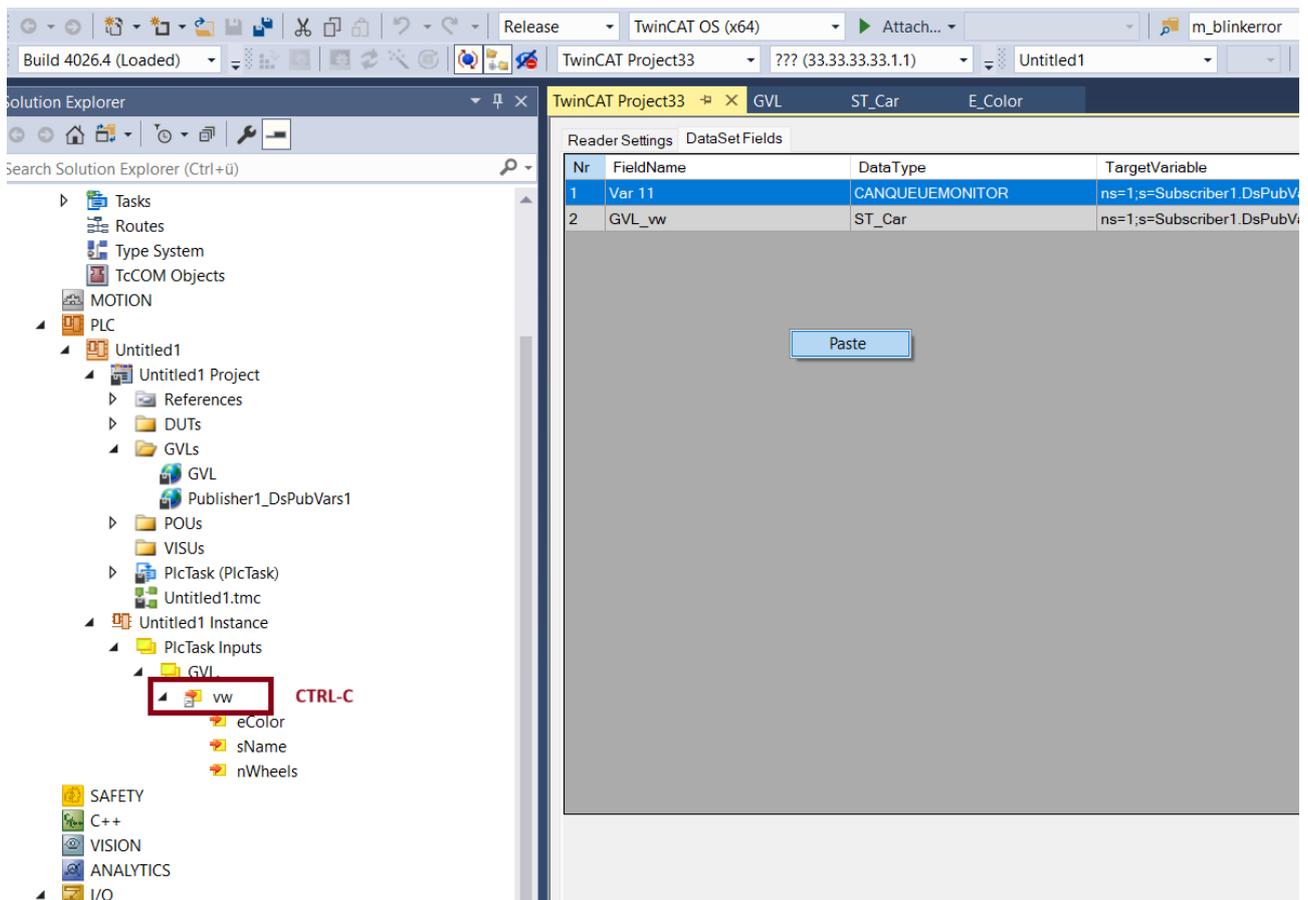
DataSets mit Variablen konfigurieren alle Variablen als Teil des Prozessabbildes. Diese DataSets können entweder global oder lokal, z. B. bei einem Publisher, hinzugefügt werden:



Die Variablen können dann direkt in das Prozessabbild des DataSets eingefügt werden:



Alternativ können Sie auch Copy&Paste auf eine Variable aus dem SPS-Prozessabbild anwenden, um sie dem DataSet hinzuzufügen, wie im folgenden Screenshot gezeigt. Kopieren Sie die Variable einfach mit STRG+C in die Zwischenablage und fügen Sie sie dann über das Kontextmenü in das Ziel-DataSet ein.



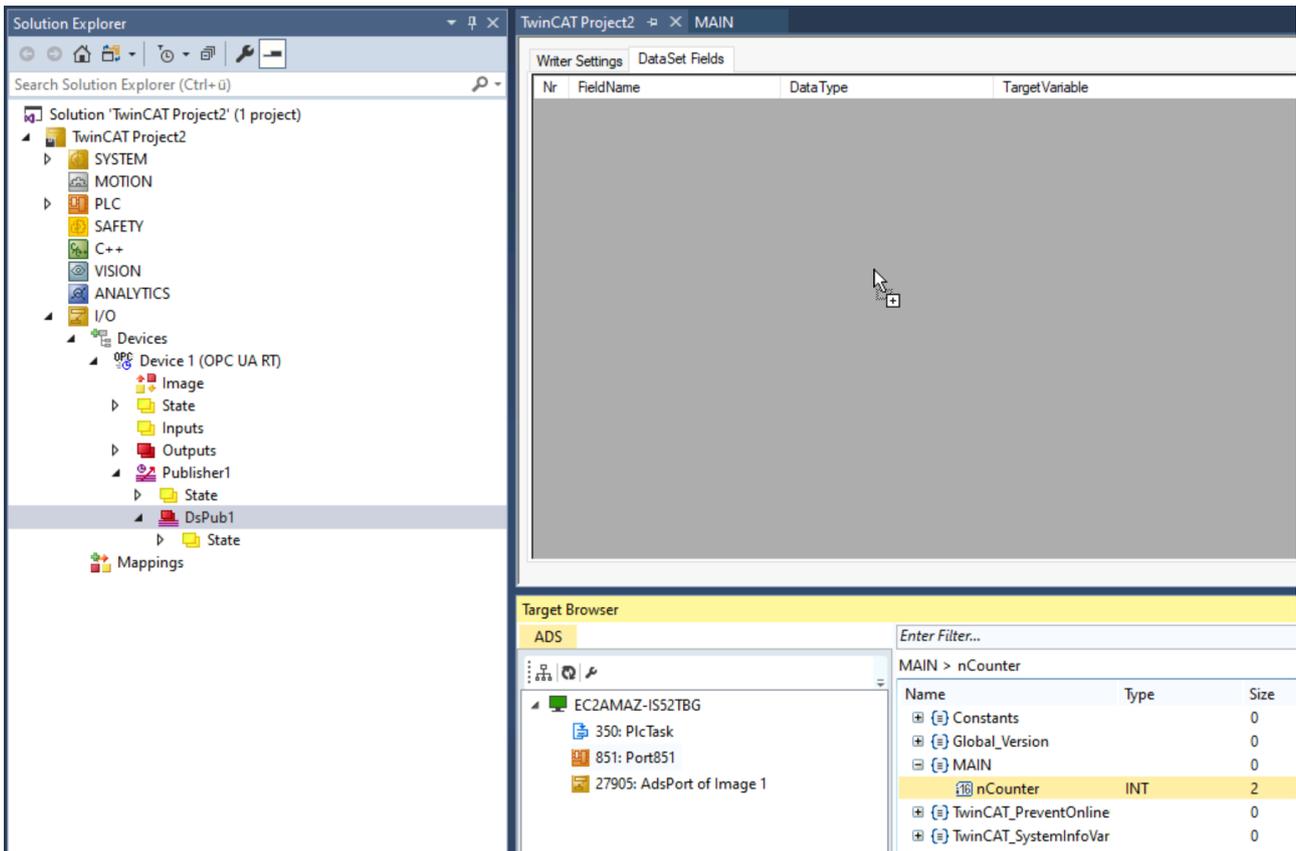
DataSet ohne Variablen

Bei DataSets ohne Variablen werden die Variablen nicht als Teil des Prozessabbildes konfiguriert. Stattdessen können Variablen entweder über den TwinCAT Target Browser oder direkt aus einem SPS-Prozessabbild in den DataSet eingefügt werden. Die Variablen werden dann automatisch mit dem entsprechenden Symbol verknüpft.

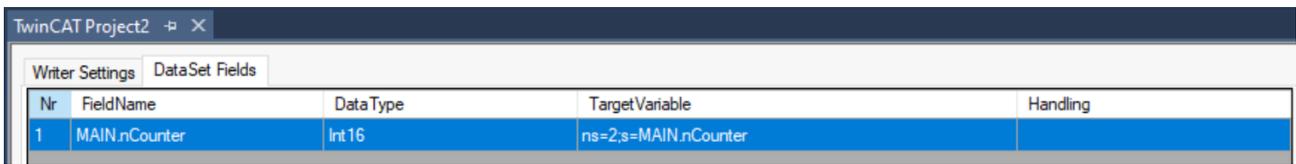
Änderungen am Symbol

i Bitte beachten Sie, dass bei einer Änderung des verknüpften Symbols die Variablen aus dem DataSet entfernt und über den TwinCAT Target Browser wieder hinzugefügt werden müssen, da die symbolischen Informationen im Rahmen des Drag&Drop-Prozesses abgefragt werden.

Um eine Variable zu dieser Art von DataSet hinzuzufügen, öffnen Sie bitte den TwinCAT Target Browser und ziehen Sie eine Variable auf die Registerkarte **Data Set Fields** des DataSets, wie im folgenden Screenshot gezeigt.

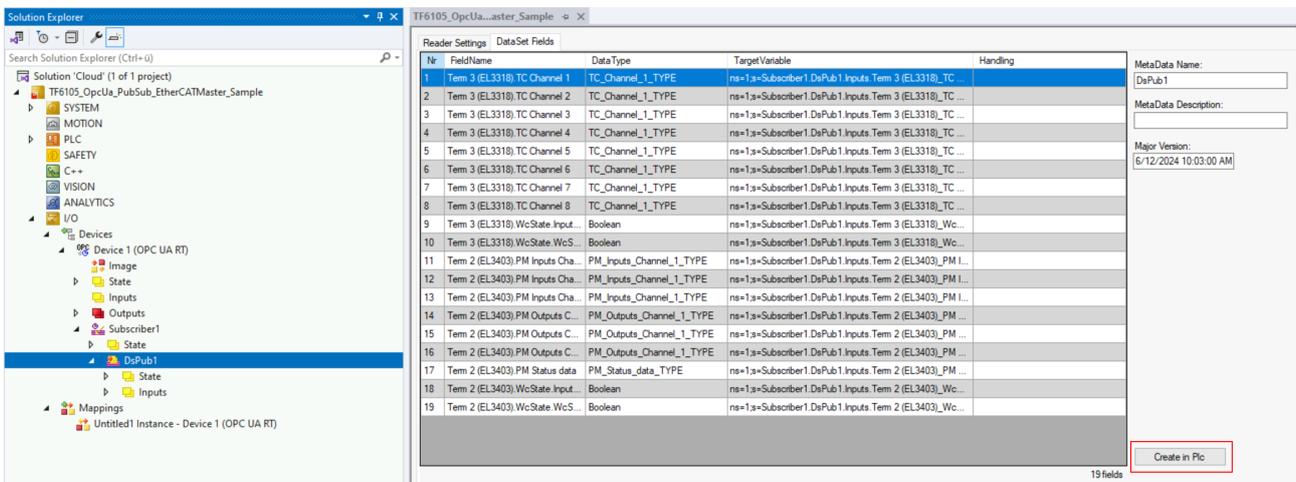


Die Feldkonfiguration des DataSets enthält nun die Variable und sie wird veröffentlicht, sobald die Konfiguration aktiviert ist.

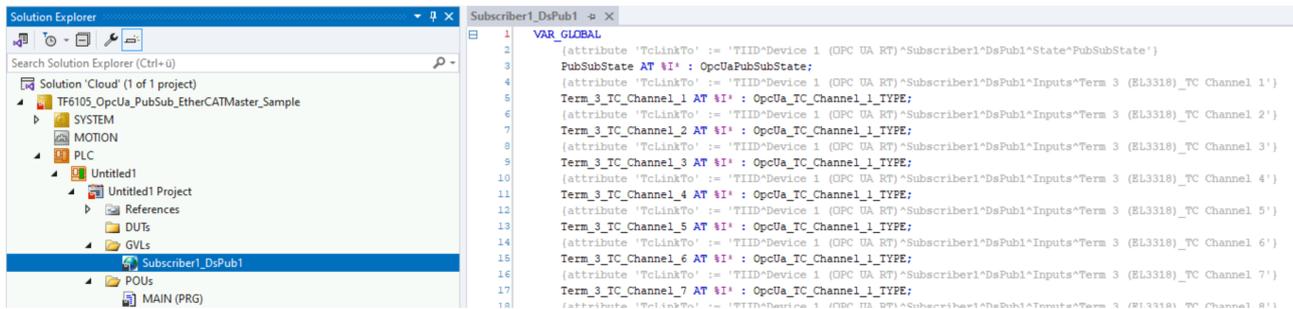


Codegenerierung

Die Subscriber-Datasets enthalten eine automatische Codegenerierung, um die für die Erstellung und Verknüpfung passender SPS-Variablen erforderliche Engineeringzeit zu reduzieren. Die Codegenerierung ist über die entsprechende Schaltfläche auf dem Subscriber-Dataset zugänglich.



Diese Funktionalität erzeugt eine globale Variablenliste, die alle Variablen aus dem Subscriber-Dataset mit übereinstimmenden Datentypen und dem Attribut "TcLinkTo" enthält, dass sie automatisch mit ihren Gegenständen verknüpft.



4.6 Transportprotokolle

Die OPC UA Pub/Sub Erweiterung definiert verschiedene Transportprotokolle, über die Daten übertragen werden können. Zu diesen Transportprotokollen gehören:

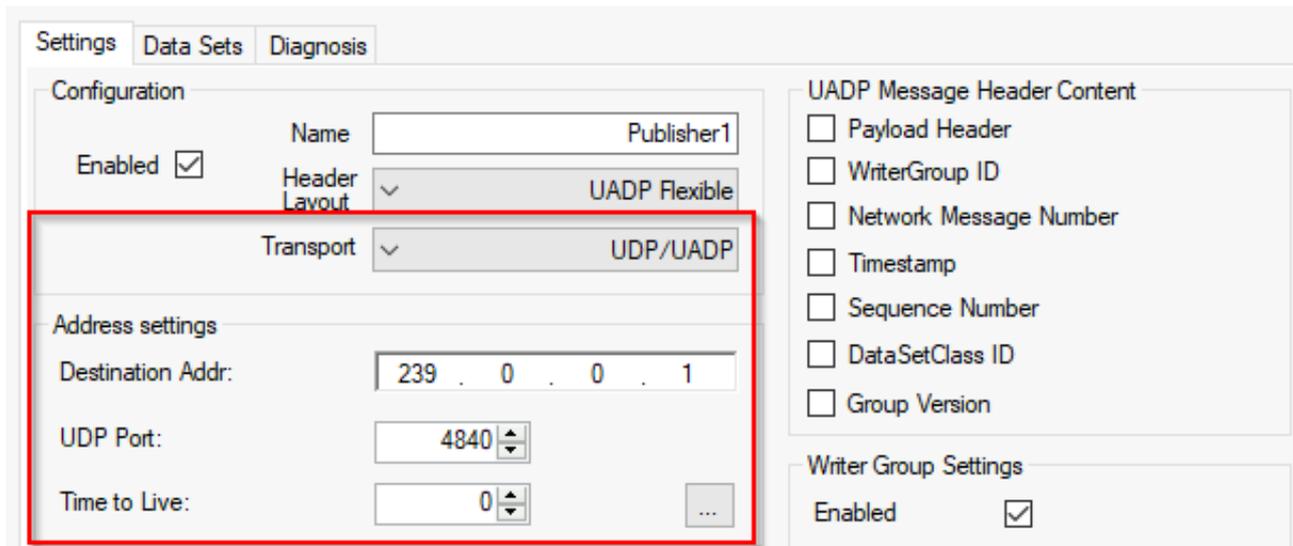
- UDP für schnelle Maschine-zu-Maschine-Kommunikation
- MQTT für die Maschine-zu-Cloud-Kommunikation

Jedes Protokoll hat seine eigenen Vor- und Nachteile, und es hängt vom Anwendungsprojekt ab, welcher Transport der richtige ist. Obwohl MQTT als "IoT"-Protokoll gilt und häufig in Verbindung mit Cloud-Diensten verwendet wird, ist es natürlich nicht auf Cloud-bezogene Anwendungsfälle beschränkt, da MQTT-Konnektivität auch innerhalb eines privaten Netzwerks eingerichtet werden kann.

Das Produkt TF6105 TC3 OPC UA Pub/Sub unterstützt beide Transportprotokolle. In den folgenden Kapiteln werden die einzelnen Konfigurationsparameter für jedes Transportprotokoll beschrieben.

4.6.1 UDP

Um einen Publisher oder Subscriber für den UDP-Transport zu konfigurieren, öffnen Sie einfach die Publisher/Subscriber-Konfigurationsseite und wählen den UDP-Transport aus. UDP unterstützt die (binäre) UADP-Kodierung und verschiedene Header-Layouts (► 35). Es stehen verschiedene Adresseinstellungen zur Verfügung, um die Verbindung zu konfigurieren, z. B. die Zieladresse, den UDP-Port und die Time-to-Live.

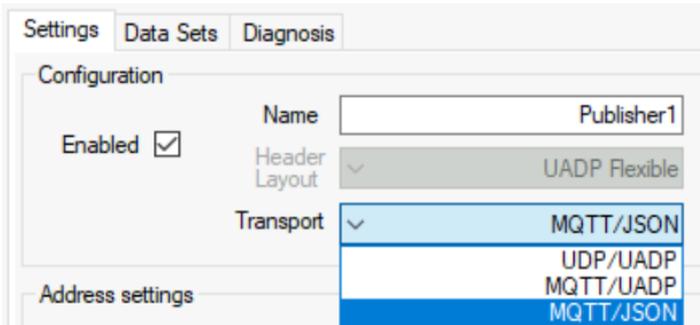


4.6.2 MQTT

Um einen Publisher oder Subscriber für die Verwendung des MQTT-Transports zu konfigurieren, öffnen Sie einfach die Publisher/Subscriber-Konfigurationsseite und wählen den MQTT-Transport aus. MQTT unterstützt sowohl die (binäre) UADP-Kodierung als auch JSON und kann mit einem bestimmten Header-Layout (► 35) konfiguriert werden.

Kodierung

Es stehen zwei Kodierungen zur Auswahl: MQTT/UADP und MQTT/JSON. Das erste Datenformat verwendet die binäre UADP-Kodierung zur Kodierung von Nachrichten und das letzte ein JSON-Format.



Je nach gewähltem Header-Layout kann die MQTT/JSON-Kodierung wie folgt aussehen:

```
{
  "MessageId": "{0CF7F27A-56FF-1F85-CFC2-12C886365C1D}",
  "MessageType": "ua-data",
  "Messages": [
    {
      "DataSetWriterId": 1,
      "Payload": {
        "nCounter": {
          "Type": 4,
          "Body": 18
        },
        "bToggle": {
          "Type": 1,
          "Body": false
        }
      }
    },
    {
      "DataSetWriterId": 2,
      "Payload": {
        "myNewVariable": {
          "Type": 4,
          "Body": 42
        }
      }
    }
  ]
}
```

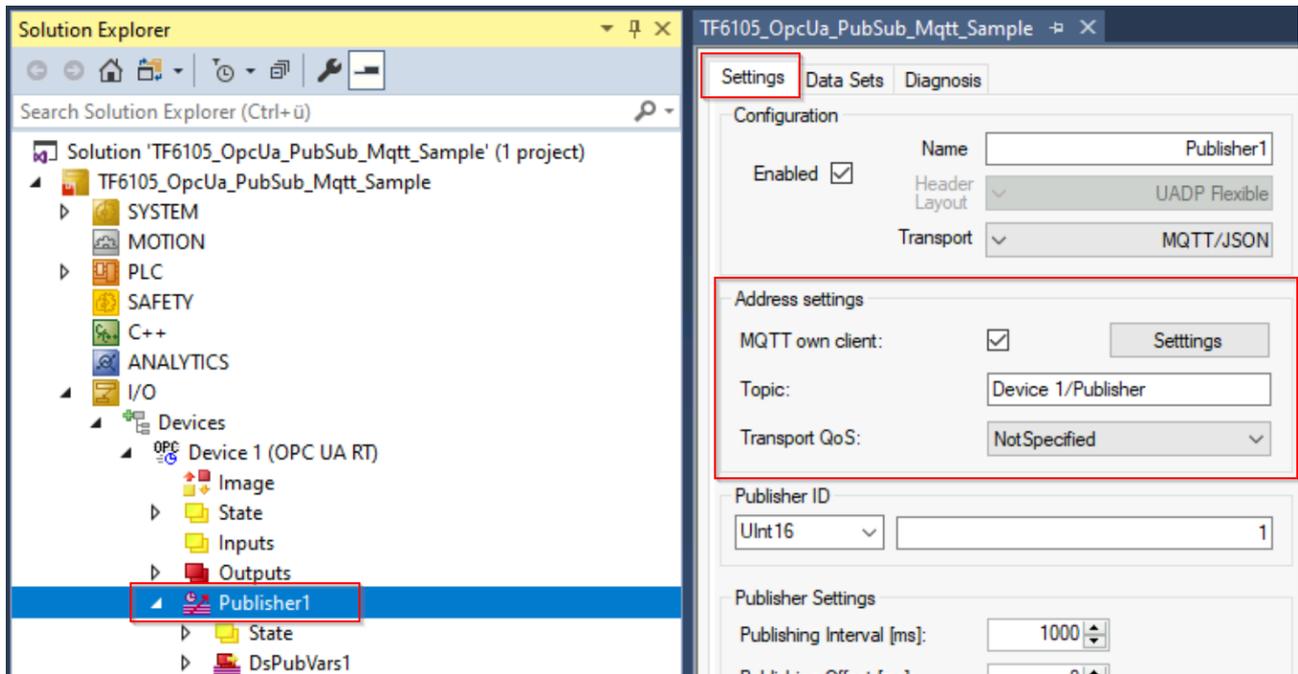
Das JSON-Format ist in der OPC UA Pub/Sub Spezifikation standardisiert. Eine vollständige Beschreibung des JSON-Schemas und jedes einzelnen Feldes finden Sie in der Spezifikation.

Verbindungsparameter

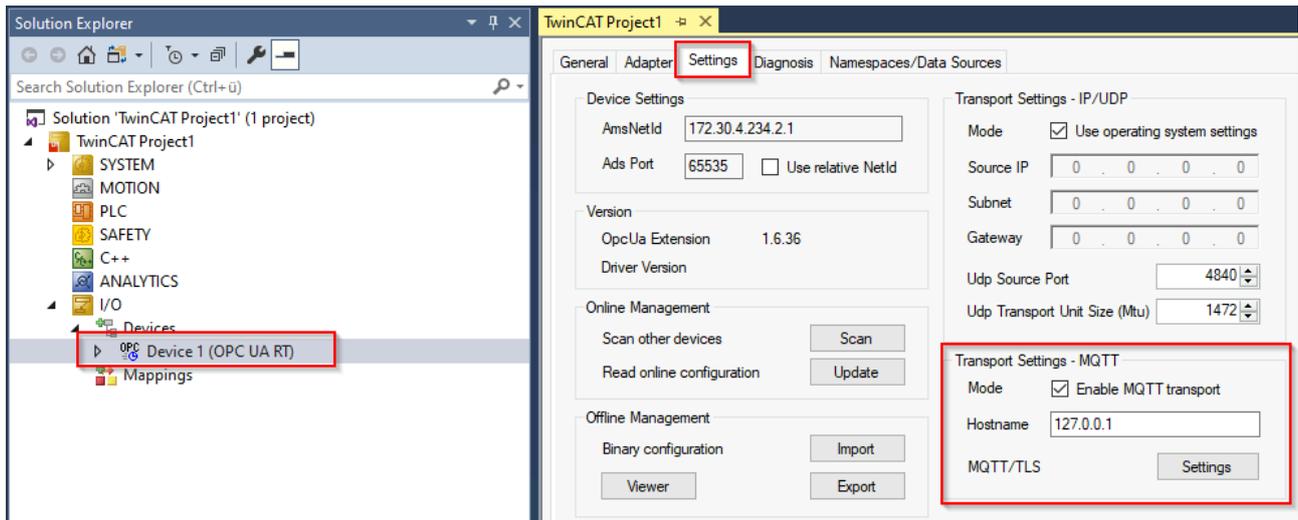
Wenn Sie MQTT als Transportprotokoll verwenden, müssen Sie auch Verbindungsparameter für den MQTT Message Broker definieren. Diese Parameter können, je nach Anwendungsfall, an zwei verschiedenen Stellen angegeben werden:

- Sie können entweder einzelne MQTT-Verbindungen auf jedem Publisher oder Subscriber konfigurieren
- oder Sie können eine "globale" MQTT Verbindung in den OPC UA RT Device Settings konfigurieren. In diesem Fall teilen sich alle Publisher und Subscriber eine Verbindung zum MQTT Message Broker.

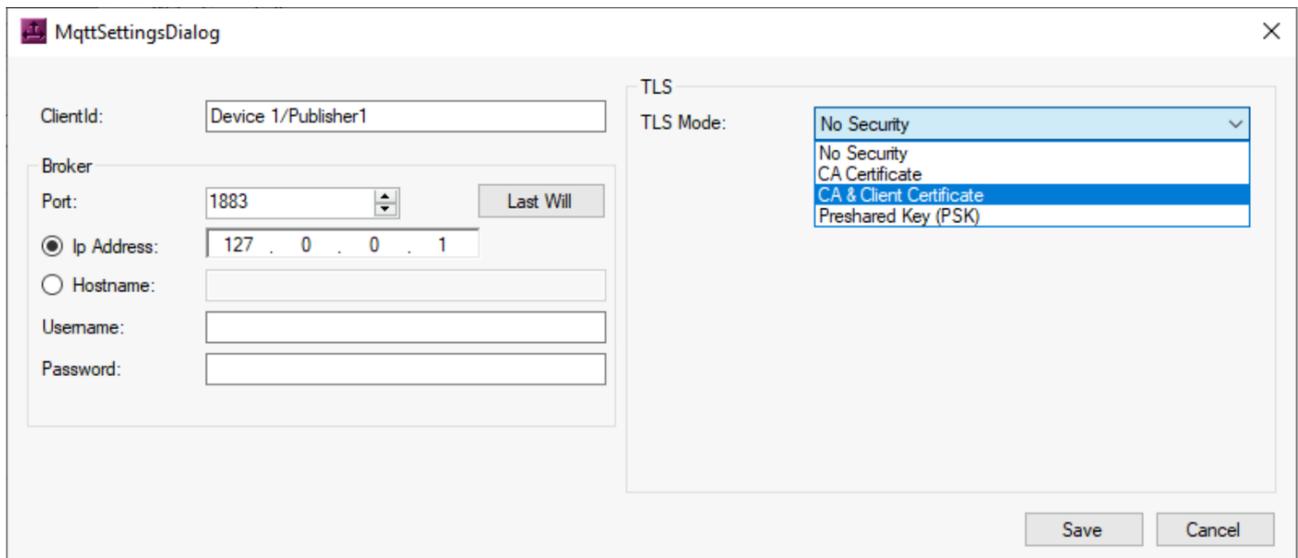
Der folgende Screenshot zeigt die MQTT-Verbindungsparameter für einen Publisher:



Der folgende Screenshot zeigt, wo die globale MQTT-Verbindung auf dem OPC UA RT Gerät eingerichtet werden kann:



In beiden Fällen ist der Dialog für die Eingabe der Verbindungsparameter gleich und sieht wie folgt aus:



Topics

Die Daten des MQTT Message Brokers sind in sogenannten "Topics" organisiert. Aus Sicht von OPC UA Pub/Sub können Sie Topics auf zwei Ebenen spezifizieren:

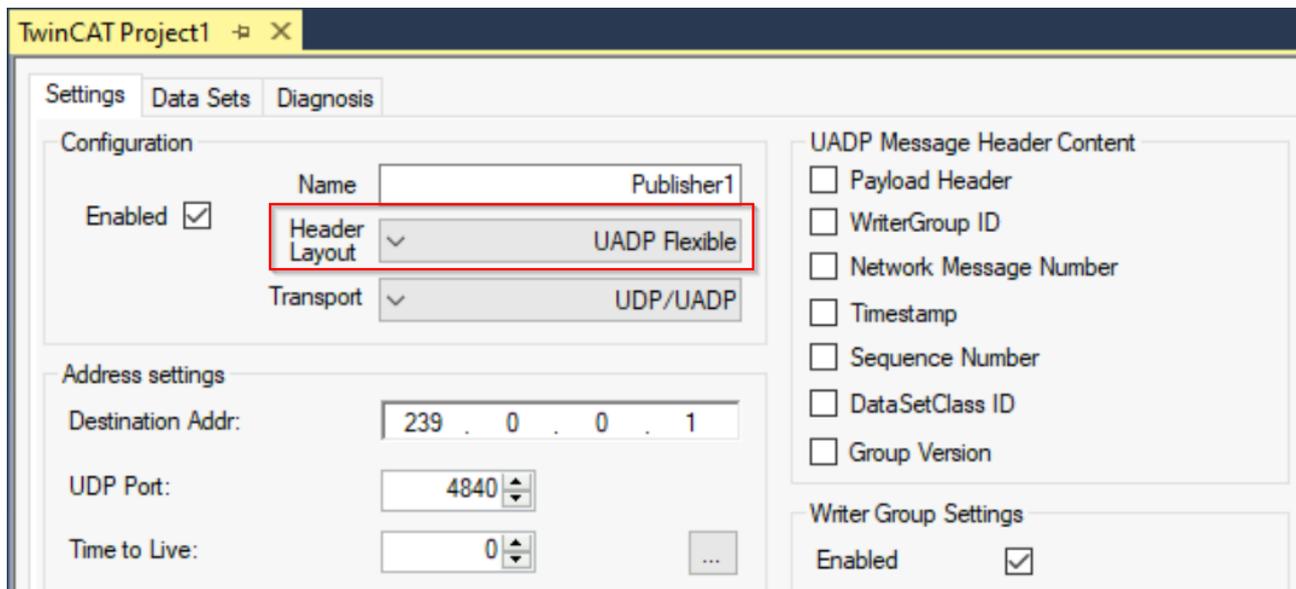
- Das Topic eines Publishers wird in den Publisher-Einstellungen konfiguriert.
- Das Topic eines Subscribers wird für jedes DataSet individuell konfiguriert.

4.7 Header-Layouts

Die Header-Inhalte und Nachrichtenlayouts wurden so gestaltet, dass sie flexibel sind und verschiedene Anwendungsfälle unterstützen, indem einzelne Felder in den Headern aktiviert oder deaktiviert werden. Diese Flexibilität ermöglicht es zwar, viele verschiedene Anwendungsfälle mit Pub/Sub zu unterstützen, aber die Anzahl der möglichen Headerfeld-Kombinationen erhöht auch den Aufwand für die Implementierung und Verifizierung. Andererseits können innerhalb eines bestimmten Anwendungsbereichs oder für verschiedene Anwendungsfälle einige Konfigurationen besser geeignet sein als andere.

Benutzerdefinierte Header-Layout-Konfigurationen (auch "flexibles" Header-Layout genannt) können verwendet werden, sollten aber auf Anwendungen beschränkt werden, die nicht unter die Anwendungsfälle für die Layouts fallen.

Das Header-Layout kann für einen Publisher oder Subscriber konfiguriert werden, wie in der folgenden Abbildung gezeigt.



Die folgenden Header-Layouts werden unterstützt. Die Namen der Header-Layouts stammen aus der OPC UA Spezifikation, Teil 14, Anhang A.

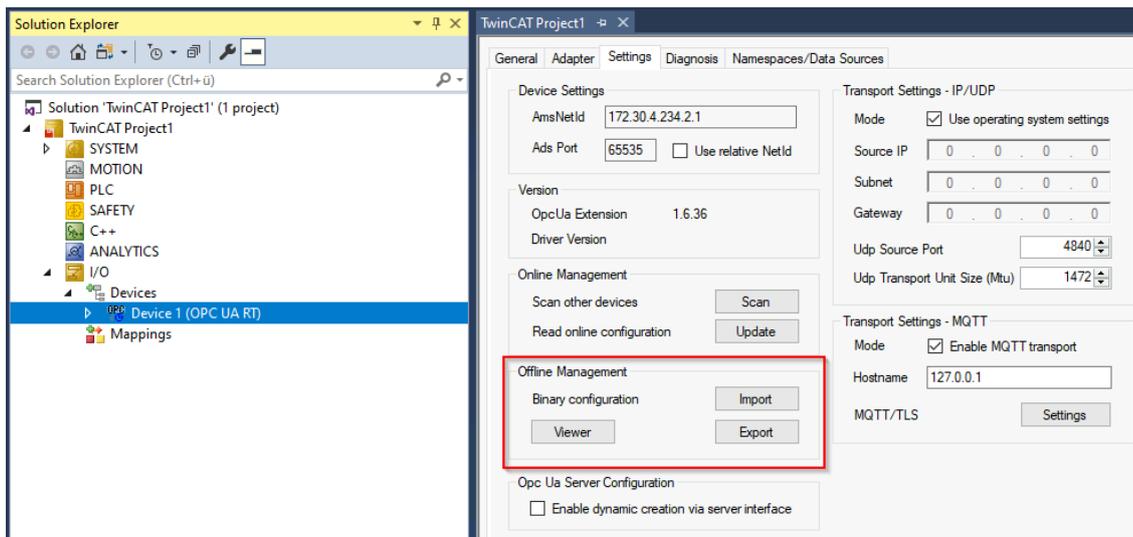
Name	Unterstützt
Flexible	Ja
UADP Dynamic	Ja
UADP PeriodicFixed	Ja
JSON DataSetMessage	Ja
JSON Minimal	Ja
JSON NetworkMessage	Ja

4.8 Konfiguration Import/Export

Der Austausch von Konfigurationen kann ein sehr wichtiger Schritt in der Engineeringphase eines Projekts sein, insbesondere wenn eine Pub/Sub-Konfiguration zwischen verschiedenen Geräten und/oder Anbietern ausgetauscht werden muss. OPC UA Pub/Sub enthält ein standardisiertes, dateibasiertes Format zum Austausch von Konfigurationen, das als "UA-Binary Import/Export" oder einfach als "Configuration Import/Export" bezeichnet wird. Der Arbeitsablauf sieht in der Regel folgendermaßen aus:

- Richten Sie Ihre Publisher auf Gerät A ein.
- Exportieren Sie die Konfigurationsdatei.
- Importieren Sie die Konfigurationsdatei auf dem/den anderen Gerät(en), um einen entsprechenden Subscriber einzurichten.

Ein Konfigurationsdialog auf dem OPC UA RT Device in TwinCAT XAE ermöglicht den Zugriff auf diese Import/Export-Funktionalität.



Änderungen an der Konfiguration

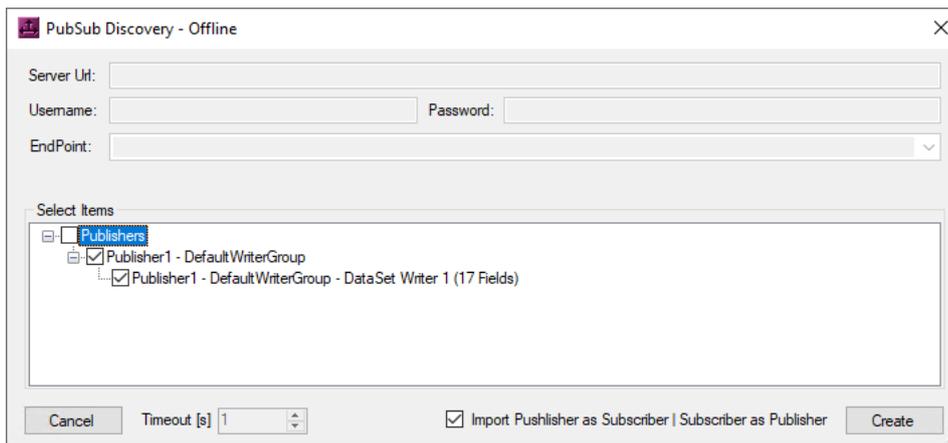
I Bitte beachten Sie, dass jede Änderung der Konfiguration auch die Erstellung einer neuen Konfigurationsdatei erfordert.

Export

Beim Exportieren einer Konfigurationsdatei wird ein Dialog geöffnet, in dem Sie das Ziel auswählen können, in dem die Datei gespeichert werden soll. Sie können dann die erstellte Datei nehmen und sie an die entsprechende Partei weitergeben.

Import

Beim Importieren einer Konfigurationsdatei wird ein Dialog geöffnet, in dem die Publisher/Subscriber ausgewählt werden können, die aus der Datei importiert werden sollen. In der Standardeinstellung wird ein Publisher als Subscriber importiert und umgekehrt.



i MQTT-Verbindungseinstellungen

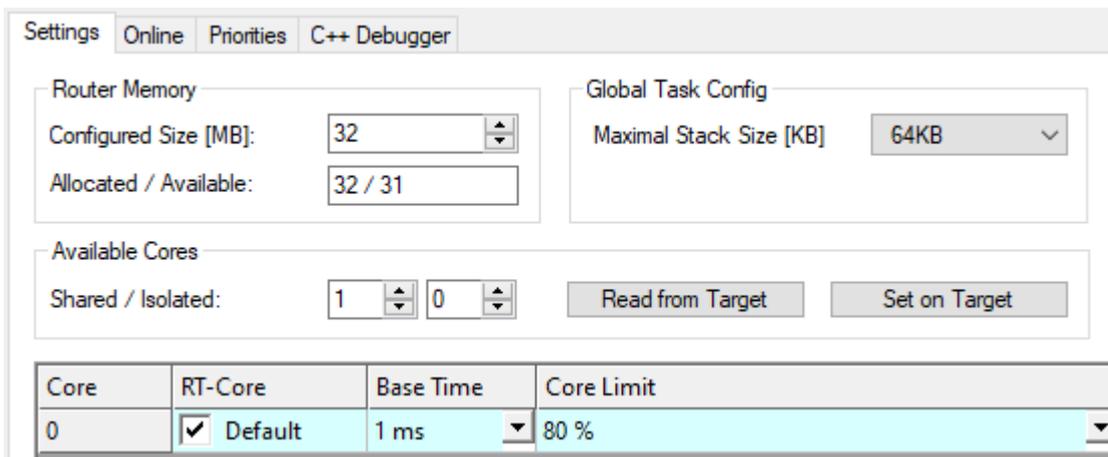
Wenn Sie eine Konfigurationsdatei importieren, die MQTT-Verbindungseinstellungen enthält, werden die Verbindungsdetails wie folgt importiert:

Wenn die MQTT-Verbindungseinstellungen für alle Konfigurationsobjekte gleich sind, werden die Einstellungen importiert und auf dem OPC UA RT Device hinzugefügt.

Wenn die MQTT-Verbindungseinstellungen bei einigen Konfigurationselementen unterschiedlich sind, werden die Einstellungen auf den Publisher/Subscriber-Knoten importiert.

Configuration Viewer

Der gleiche Einstellungsbereich enthält auch einen so genannten "Configuration Viewer", der eine Vorschau auf den Inhalt der binären Pub/Sub-Konfigurationsdatei ermöglicht.



Bei der Konfiguration eines Publishers korreliert das PublishingInterval nun mit dieser BaseTime. Der folgende Screenshot zeigt ein konfiguriertes PublishingInterval von 100 ms (100 Ticks * 1 ms BaseTime).

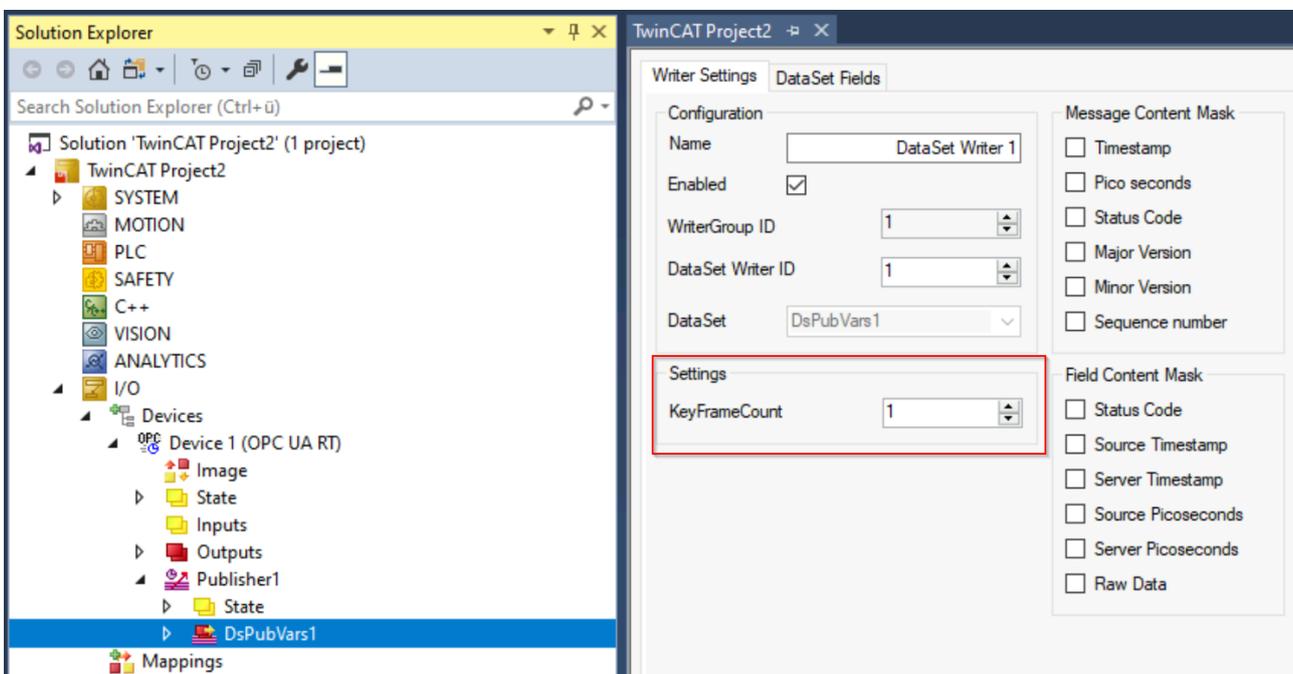


4.9.2 KeyFrames, DeltaFrames, KeepAlive

In diesem Dokumentationsartikel wird erläutert, wie KeyFrames, DeltaFrames und KeepAlive miteinander korrelieren, wenn sie für einen Publisher oder einen Publisher-DataSet konfiguriert werden.

KeyFrames und DeltaFrames

KeyFrames enthalten immer den vollständigen DataSet, während DeltaFrames nur inkrementelle Aktualisierungen enthalten (Felder, die sich geändert haben). Bei der Konfiguration eines Publisher-DataSet gibt der KeyFrameCount an, wie oft ein KeyFrame in Abhängigkeit vom PublishingInterval veröffentlicht werden soll.



Bei der Einstellung "1" wird in jedem Publisher-Zyklus ein KeyFrame veröffentlicht. Bei einem Wert größer als "1" wird alle n Zyklen ein KeyFrame veröffentlicht, andernfalls wird ein DeltaFrame veröffentlicht. Bitte beachten Sie, dass Delta-Frames nur dann veröffentlicht werden, wenn es eine Datenänderung gibt, was bedeutet, dass es Zeiten geben kann, in denen überhaupt keine Frames veröffentlicht werden.

KeepAlive

Die KeepAlive-Zeit, die auf einem Publisher konfiguriert wird, wird verwendet, um Frames zu senden, wenn es keine Datenänderungen (und damit Delta-Frames) gibt, die über einen längeren Zeitraum gesendet werden. Damit ist sichergestellt, dass die Subscriber weiterhin wissen, dass der Publisher online ist. Bitte beachten Sie bei der Konfiguration des KeepAlive für einen Subscriber, dass dies auch mit dem MessageReceiveTimeout korreliert.

4.9.3 MessageReceiveTimeout

Der MessageReceiveTimeout wird als Teil einer Subscriber-Konfiguration konfiguriert. Er gibt an, nach welcher Zeit eine eingehende Nachricht für einen Subscriber-DataSet zu erwarten ist - andernfalls kommt es zu einem Timeout und die Prozessvariable PubSubState geht in den Fehlerzustand. Der MessageReceiveTimeout kann durch das [KeepAlive](#) [► 39] zurückgesetzt werden.

5 Samples

Beispielcode und -konfigurationen für dieses Produkt können über das entsprechende Repository auf GitHub bezogen werden: https://github.com/Beckhoff/TF6105_Samples. Sie haben dort die Möglichkeit, das Repository zu clonen oder eine ZIP-Datei mit dem Sample herunterzuladen. Gegenwärtig sind die folgenden Samples vorhanden:

Sample	Beschreibung
UDP [▶ 41]	Zeigt, wie man einen Beispiel-Publisher und -Subscriber basierend auf OPC UA Pub/Sub UDP einrichtet.
MQTT [▶ 42]	Zeigt, wie man einen Beispiel-Publisher und -Subscriber auf Basis von OPC UA Pub/Sub MQTT einrichtet.
Daten vom EtherCAT-Master veröffentlichen	Erklärt, wie man einen EtherCAT Master ADS Symbol Server einrichtet und einen OPC UA Publisher konfiguriert, um dessen Datenpunkte zu veröffentlichen. Dieses Beispiel ist nicht als Download verfügbar, aber alle erforderlichen Schritte werden in diesem Dokumentationsartikel erläutert.

Wir empfehlen Ihnen auch unsere [Quick Start Tutorials](#) [▶ 13], die einen geführten Durchgang durch verschiedene Anwendungsfälle der Konfiguration bieten.

5.1 UDP

Dieses Beispiel zeigt, wie man einen Publisher und Subscriber auf Basis von OPC UA Pub/Sub UDP einrichtet. Die Sample-Solution enthält ein TwinCAT-Projekt, das die folgende SPS-Konfiguration enthält:

- ein SPS-Projekt, das zufällige Daten generiert und die Daten mit den [Data Sets](#) [▶ 27] der OPC UA RT Device's Publisher verknüpft.
- ein SPS-Projekt, das die von den Subscribern empfangenen Daten auf dem OPC UA RT Device speichert.

Die E/A-Konfiguration umfasst ein OPC UA RT Device mit zwei Publishern und zwei Subscribern:

- einen Publisher, der so konfiguriert ist, dass er ein DataSet mit Variablen enthält. Die Variablen werden über das Prozessabbild mit den Ausgangsvariablen des ersten SPS-Projekts verknüpft.
- einen Publisher, der so konfiguriert ist, dass er ein DataSet ohne Variablen enthält. Die Variablen sind intern über ihre ADS-Symbolnamen mit Variablen im ersten SPS-Projekt verknüpft.
- einen Subscriber, der so konfiguriert ist, dass er die vom ersten Publisher veröffentlichten Daten abonnieren kann. Die Eingangsvariablen werden mit Variablen aus dem zweiten SPS-Projekt verknüpft.
- einen Subscriber, der so konfiguriert ist, dass er die von dem zweiten Publisher veröffentlichten Daten abonnieren kann. Die Eingangsvariablen werden mit Variablen aus dem zweiten SPS-Projekt verknüpft.

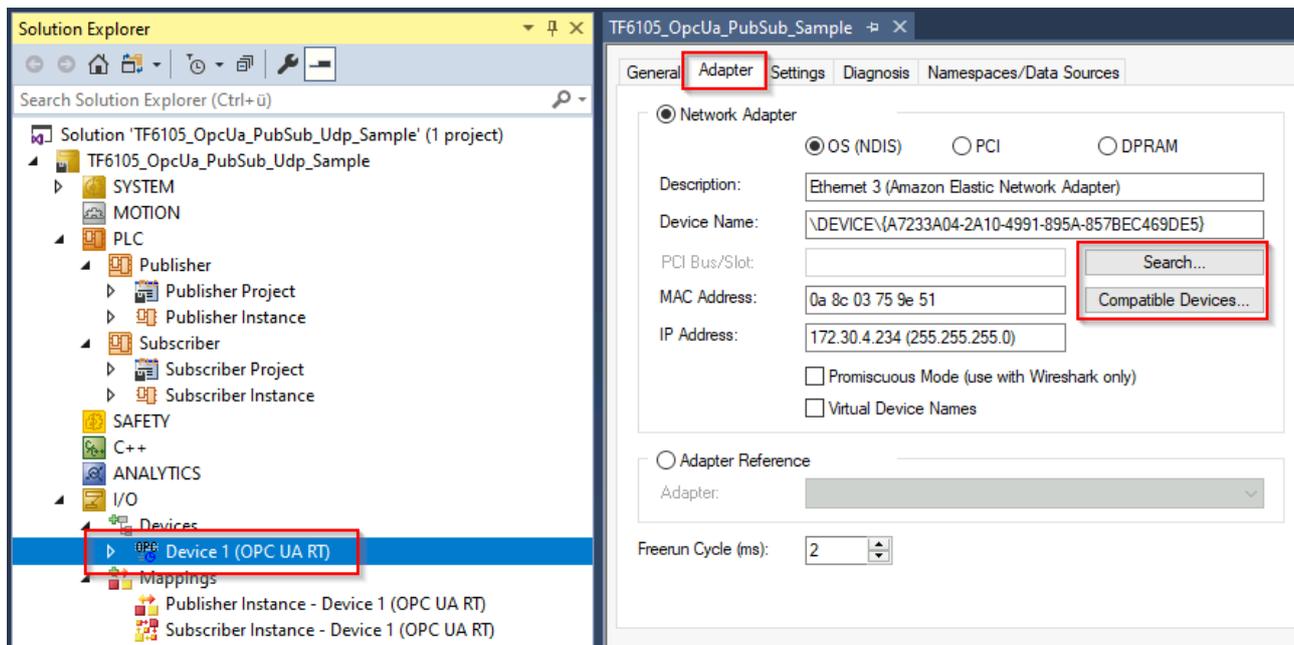
● Erstellung der Subscriber



Die Subscriber wurden automatisch auf der Grundlage von [Konfiguration Import/Export](#) [▶ 36] erstellt. Wir empfehlen dringend, dieses Austauschformat zu verwenden, wo immer es möglich ist.

So starten Sie das Beispiel

Bevor Sie das Beispielprojekt aktivieren, muss das OPC UA RT Device für eine Netzwerkkarte eingerichtet werden, auf der der Treiber des TwinCAT Realtime-Ethernet Adapters installiert ist. Dies kann über den Dialog OPC UA RT Geräteeinstellungen erfolgen.



Bitte vergewissern Sie sich auch, dass Ihr System die [Systemanforderungen](#) [► 9] erfüllt.

5.2 MQTT

Dieses Beispiel zeigt, wie man einen Publisher und Subscriber auf Basis von OPC UA Pub/Sub MQTT einrichtet. Die Sample-Solution enthält ein TwinCAT-Projekt, das die folgende SPS-Konfiguration enthält:

- ein SPS-Projekt, das zufällige Daten generiert und die Daten mit den [Data Sets](#) [► 27] der OPC UA RT Device's Publisher verknüpft.
- ein SPS-Projekt, das die von den Subscribern empfangenen Daten auf dem OPC UA RT Device speichert.

Die E/A-Konfiguration umfasst ein OPC UA RT Device mit zwei Publishern und zwei Subscribern:

- einen Publisher, der so konfiguriert ist, dass er einen DataSet mit Variablen enthält. Die Variablen werden über das Prozessabbild mit den Ausgangsvariablen des ersten SPS-Projekts verknüpft.
- einen Publisher, der so konfiguriert ist, dass er einen DataSet ohne Variablen enthält. Die Variablen sind intern über ihre ADS-Symbolnamen mit Variablen im ersten SPS-Projekt verknüpft.
- einen Subscriber, der so konfiguriert ist, dass er die vom ersten Publisher veröffentlichten Daten abonnieren kann. Die Eingangsvariablen werden mit Variablen aus dem zweiten SPS-Projekt verknüpft.
- einen Subscriber, der so konfiguriert ist, dass er die von dem zweiten Publisher veröffentlichten Daten abonnieren kann. Die Eingangsvariablen werden mit Variablen aus dem zweiten SPS-Projekt verknüpft.

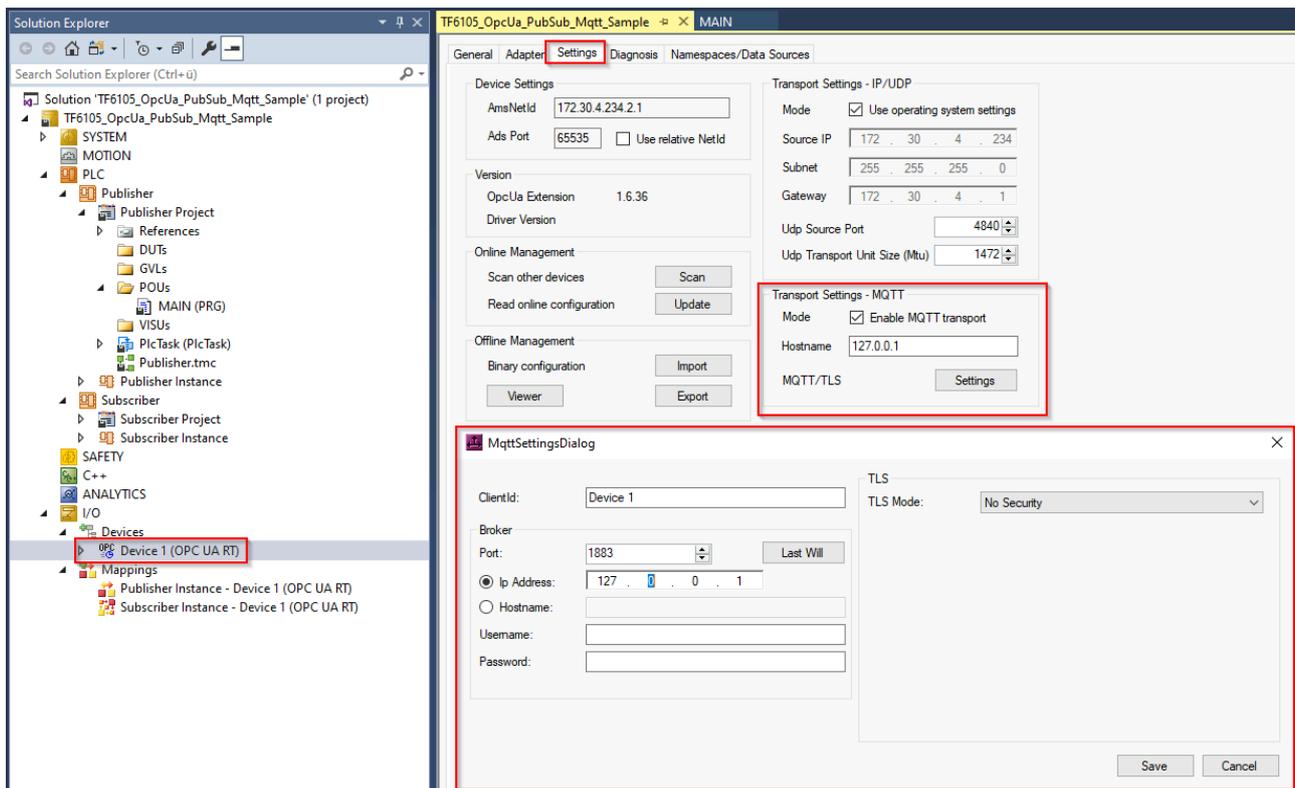
i Erstellung der Subscriber

Die Subscriber wurden automatisch auf der Grundlage von [Konfiguration Import/Export](#) [► 36] erstellt. Wir empfehlen dringend, dieses Austauschformat zu verwenden, wo immer es möglich ist.

So starten Sie das Beispiel

Bitte stellen Sie sicher, dass Ihr System die [Systemanforderungen](#) [► 9] erfüllt.

Bevor Sie das TwinCAT-Projekt aktivieren, vergewissern Sie sich bitte, dass Sie einen MQTT Message Broker lokal auf demselben System laufen haben. Der Message Broker sollte ohne Verschlüsselung oder Benutzerauthentifizierung auf Port 1883/tcp hören. Bitte beachten Sie, dass dies nur zu Demonstrationszwecken dient! Bitte passen Sie diese Konfiguration an Ihre Umgebung an. Die MQTT-Einstellungen sind auf dem OPC UA RT Device zu finden.



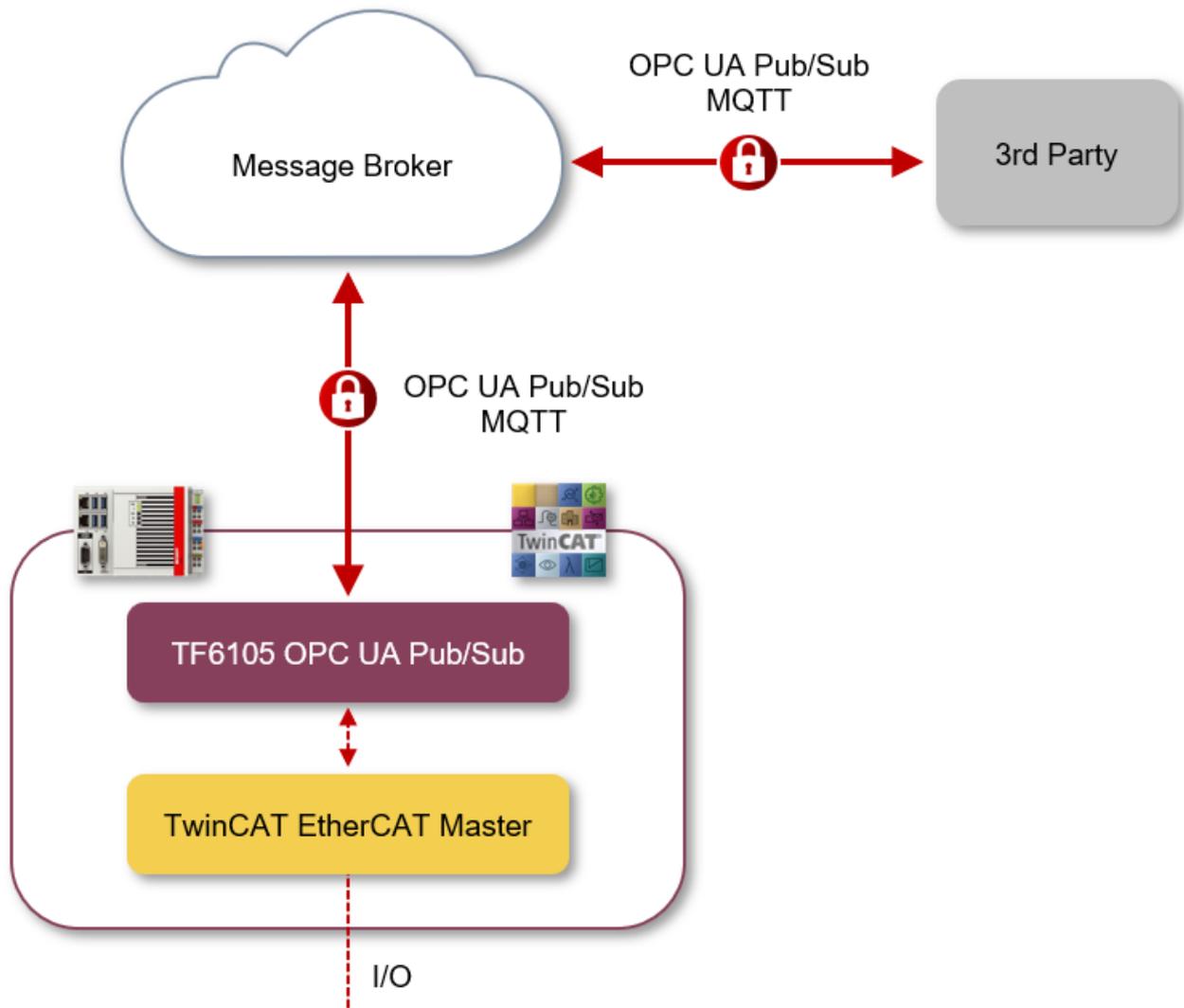
Sie können dann das TwinCAT-Projekt aktivieren und sich in die SPS-Projekte einloggen, um die SPS-Laufzeit zu debuggen. Sie sollten sehen, dass das Subscriber-SPS-Projekt Daten empfängt (die von dem anderen SPS-Projekt veröffentlicht werden).

5.3 EtherCAT Master

Sie können auf einfache Weise Datenpunkte aus einem EtherCAT-Master-Abbild veröffentlichen. Ein SPS-Zwischenprojekt ist nicht erforderlich, kann aber hinzugefügt werden, falls Sie eine Datenkonvertierung oder -umwandlung benötigen. Die folgende Dokumentation beschreibt alle notwendigen Schritte, um diesen Anwendungsfall einzurichten. Diese Schritte bestehen aus:

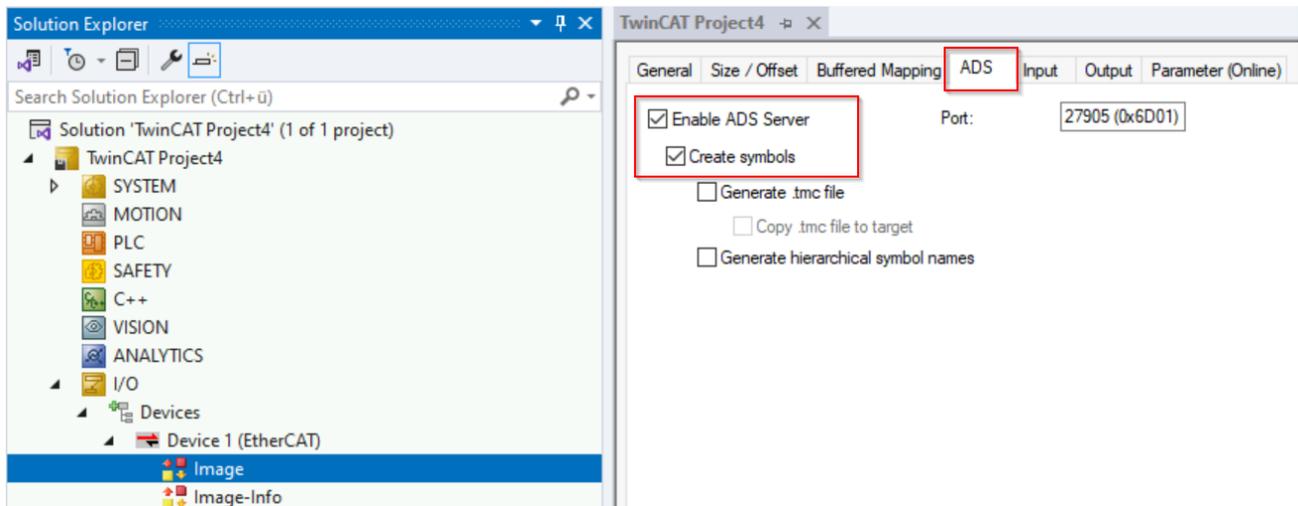
1. Aktivierung des ADS Symbol Server auf dem EtherCAT Master
2. Einrichten eines OPC UA RT Device
3. Konfiguration eines Publishers mit einem DataSet
4. Hinzufügen von Datenpunkten aus dem EtherCAT-Master zum Publisher-DataSet
5. [optional] Einrichten eines TwinCAT Subscriber
6. [optional] Einrichten eines generischen MQTT-Client-Subscriber

Dieses Beispiel demonstriert diesen Aufbau anhand eines Anwendungsfalls, in dem I/O-Datenpunkte vom EtherCAT-Master an einen Message Broker in der Cloud veröffentlicht werden sollen. Eine Anwendung von Drittherstellern sollte sich dann mit demselben Broker verbinden und die Daten abonnieren. Dieses Beispiel kann an verschiedene Varianten dieses Anwendungsfalls angepasst werden, je nach Bedarf der Anwendung.



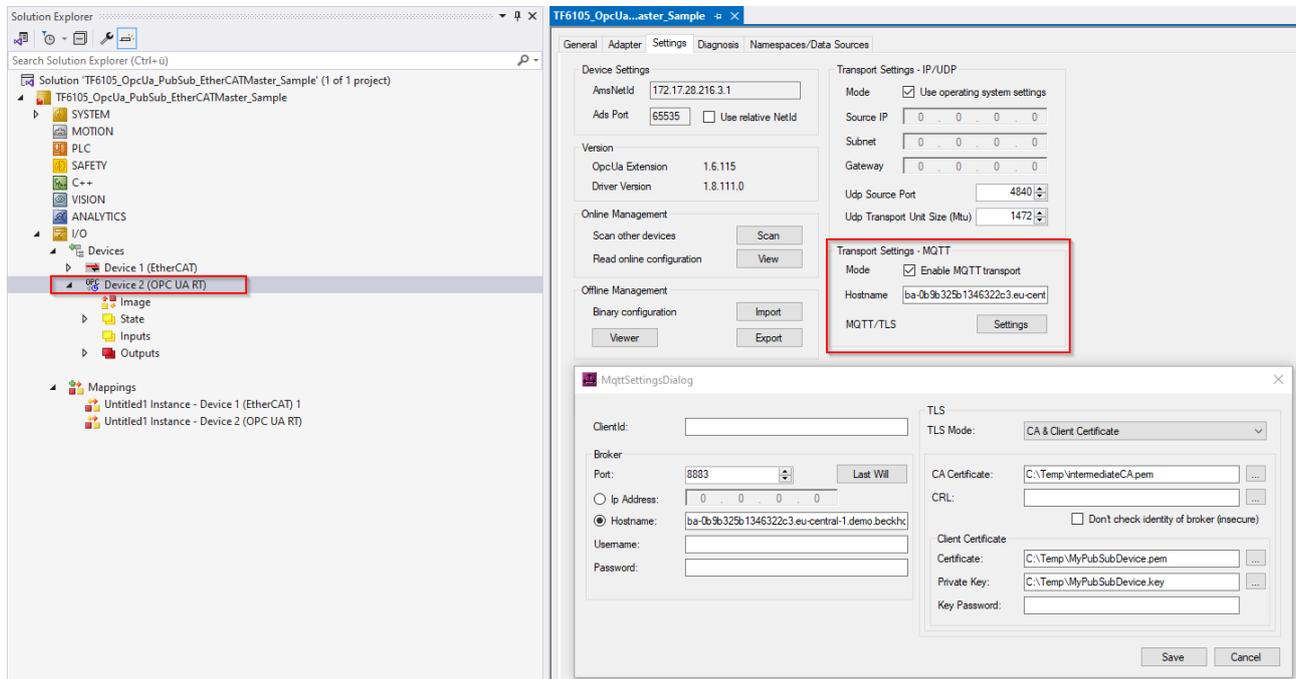
Aktivierung des ADS Symbol Server auf dem EtherCAT Master

Der TwinCAT EtherCAT Master bietet die Funktionalitäten eines ADS Symbol Servers. Alle Klemmen und Variablen des EtherCAT-Masters werden dann über ADS zur Verfügung gestellt. ADS-Clients können sich mit dem ADS Symbol Server verbinden, um dessen Namensraum zu durchsuchen und Daten zu lesen/schreiben. Sie können diese Funktion auf der ADS-Einstellungsseite des EtherCAT Master Image aktivieren.



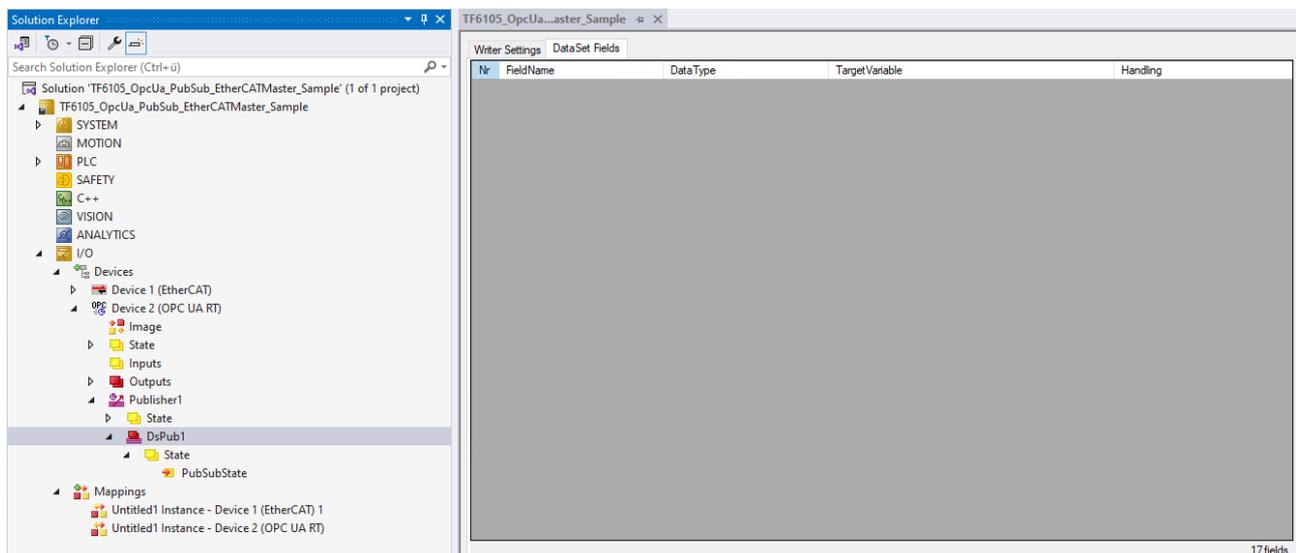
Einrichten eines OPC UA RT Device

Richten Sie ein OPC UA RT Device ein, wie in unseren [Quick Start \[▶ 13\]](#) Tutorials beschrieben. In diesem Beispiel wird das OPC UA RT Device für eine sichere MQTT Verbindung zu einem entfernten Message Broker konfiguriert, der in der Cloud läuft, aber Sie können dieses Beispiel auch lokal ausführen. Bitte geben Sie die MQTT-Verbindungsdaten entsprechend Ihrer Umgebung ein.



Konfiguration eines Publishers mit einem DataSet

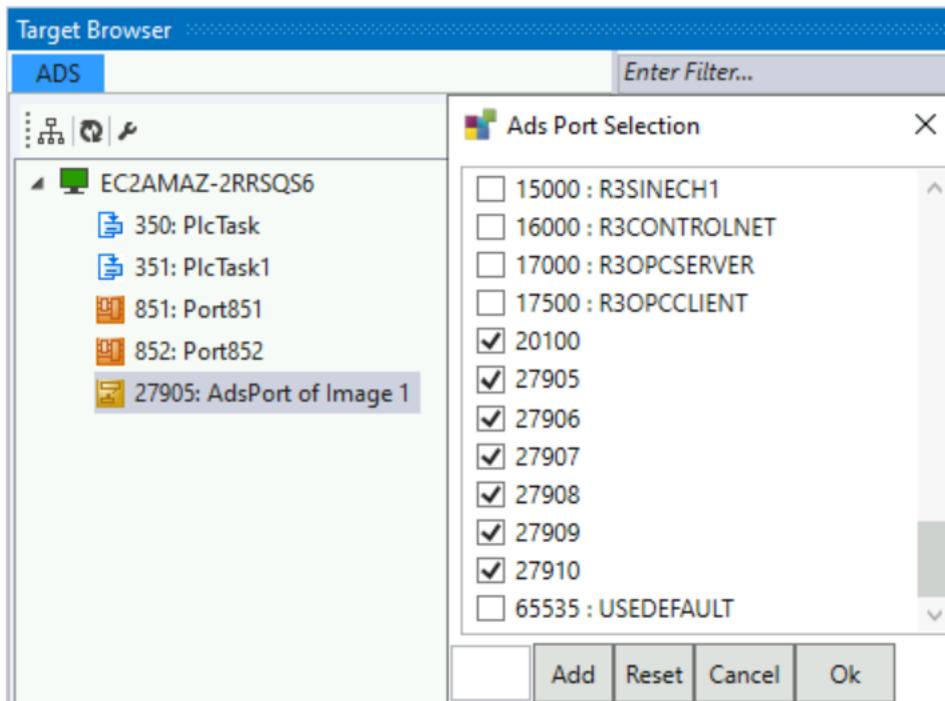
Als Nächstes möchten Sie einen Publisher erstellen, der einen DataSet mit Variablen enthält, die mit den Datenpunkten des EtherCAT Master Image verknüpft sind. Rechtsklicken Sie auf das OPC UA RT Device und wählen Sie **Add New Item** und dann **OPC UA Publisher**. Rechtsklicken Sie dann auf den hinzugefügten Publisher und wählen Sie **Add New Item** und dann **Published DataSet**. Beachten Sie, dass der DataSet noch keine Felder enthält, wie im folgenden Screenshot zu sehen ist.



Sie können nun mit dem TwinCAT Target Browser Datenpunkte aus dem EtherCAT Master hinzufügen.

Hinzufügen von Datenpunkten aus dem EtherCAT-Master zum Publisher-DataSet

Öffnen Sie den TwinCAT Target Browser und navigieren Sie zu dem ADS Symbol Server des EtherCAT Masters. Wenn dieser ADS Server noch nicht angezeigt wird, können Sie seinen Port über den Dialog ADS Port Selection hinzufügen, wie im folgenden Screenshot gezeigt.

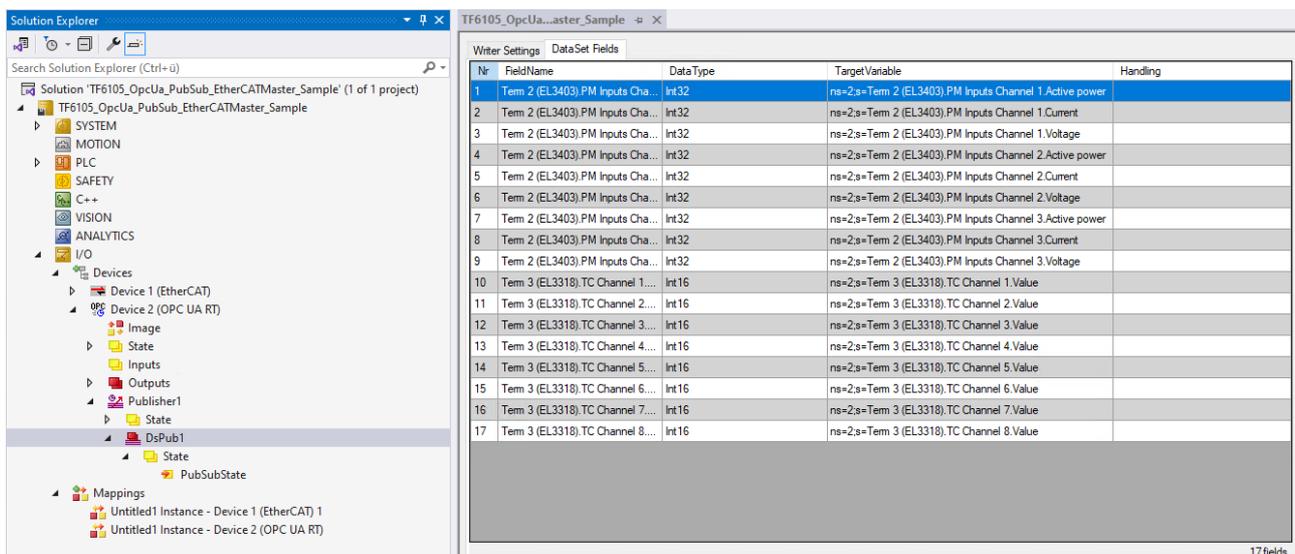


Den ADS Port des EtherCAT Master Image finden Sie auf der Einstellungsseite in Schritt 1.

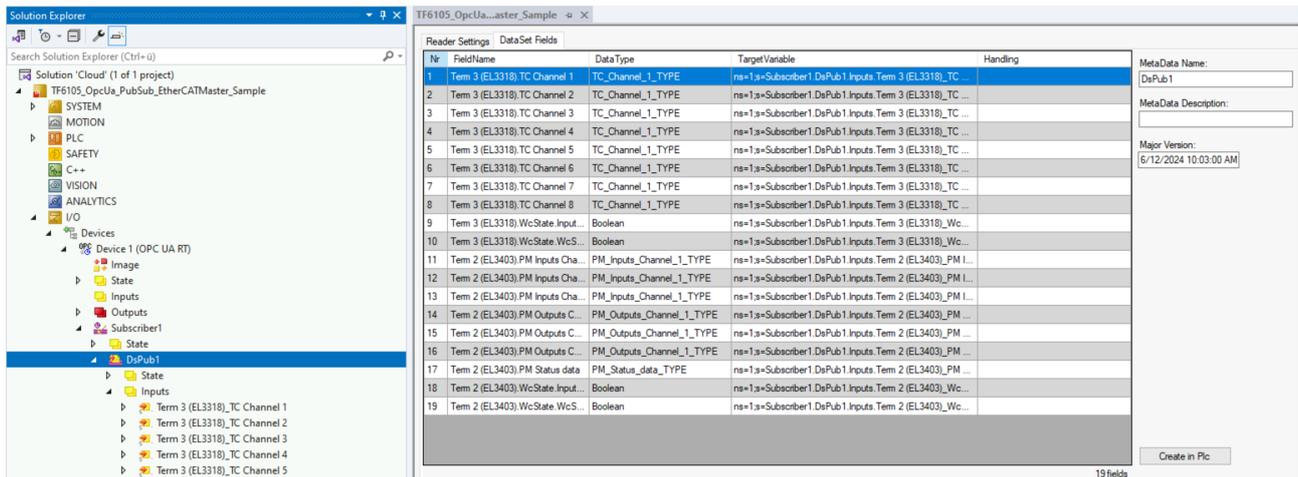
Sie können nun durch den Namensraum des EtherCAT Master Image navigieren und die Datenpunkte auswählen, die Sie veröffentlichen möchten. Beim Hinzufügen von Datenpunkten haben Sie vier Möglichkeiten:

1. Wählen Sie jeden Datenpunkt einzeln aus und fügen Sie ihn per Drag&Drop dem DataSet hinzu.
2. Wählen Sie das gesamte Prozessabbild aus und fügen Sie es per Drag&Drop dem DataSet hinzu. In diesem Fall wird jede Klemme auf dem Bild als strukturierter Datentyp zum DataSet hinzugefügt.
3. Wählen Sie das gesamte Prozessabbild aus und fügen Sie es mit SHIFT + Drag&Drop dem DataSet hinzu. In diesem Fall werden nur die Variablen auf jeder Klemme zum DataSet hinzugefügt (als einfache Datentypen). Diese Option kann erforderlich sein, wenn der OPC UA Pub/Sub Subscriber keine strukturierten Typen unterstützt.
4. Wählen Sie die Klemmen aus und fügen Sie sie per Drag&Drop dem DataSet hinzu.

Der folgende Screenshot zeigt das Ergebnis einer Drag&Drop-Operation wie in Option 1 beschrieben für zwei Klemmen (EL3403 und EL3318).



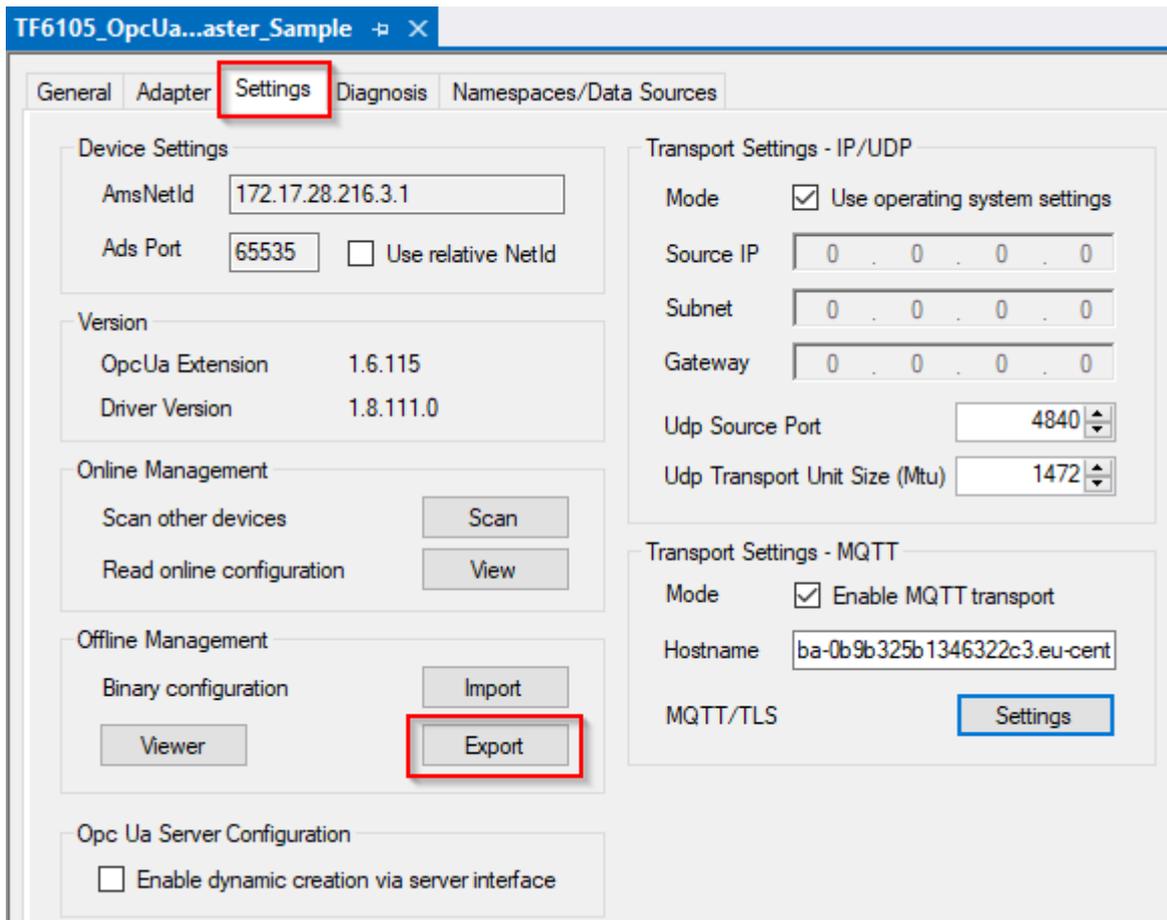
Der folgende Screenshot zeigt das Ergebnis einer Drag&Drop-Operation wie in Option 4 beschrieben für zwei Klemmen (EL3403 und EL3318).



[optional] Einrichten eines TwinCAT Subscriber

Sie können ganz einfach einen Subscriber für die veröffentlichten Datenpunkte einrichten, indem Sie die [Konfiguration \[► 36\]](#) in eine OPC UA Binäraustauschdatei exportieren. Diese Datei kann dann auf der anderen Seite importiert werden, um automatisch einen Subscriber zu erstellen, der der Konfiguration des Publishers entspricht.

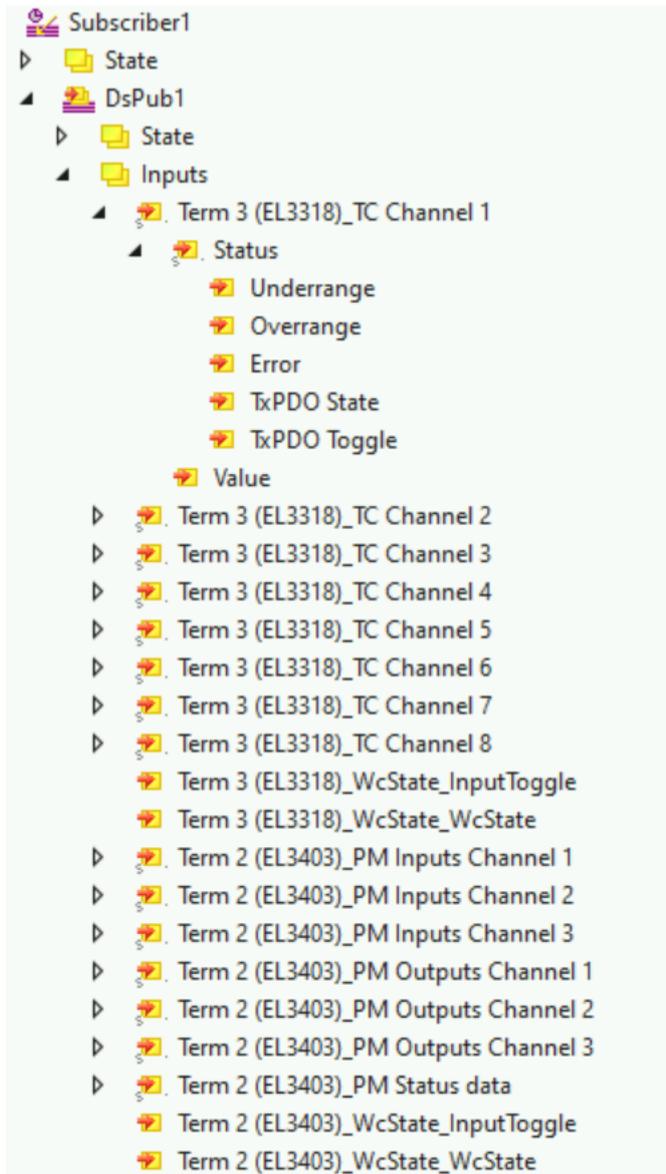
Um die Konfiguration zu exportieren, doppelklicken Sie auf das OPC UA RT Device und wählen Sie **Settings**. Klicken Sie auf die Schaltfläche **Export** und speichern Sie die Konfiguration in einer Datei.



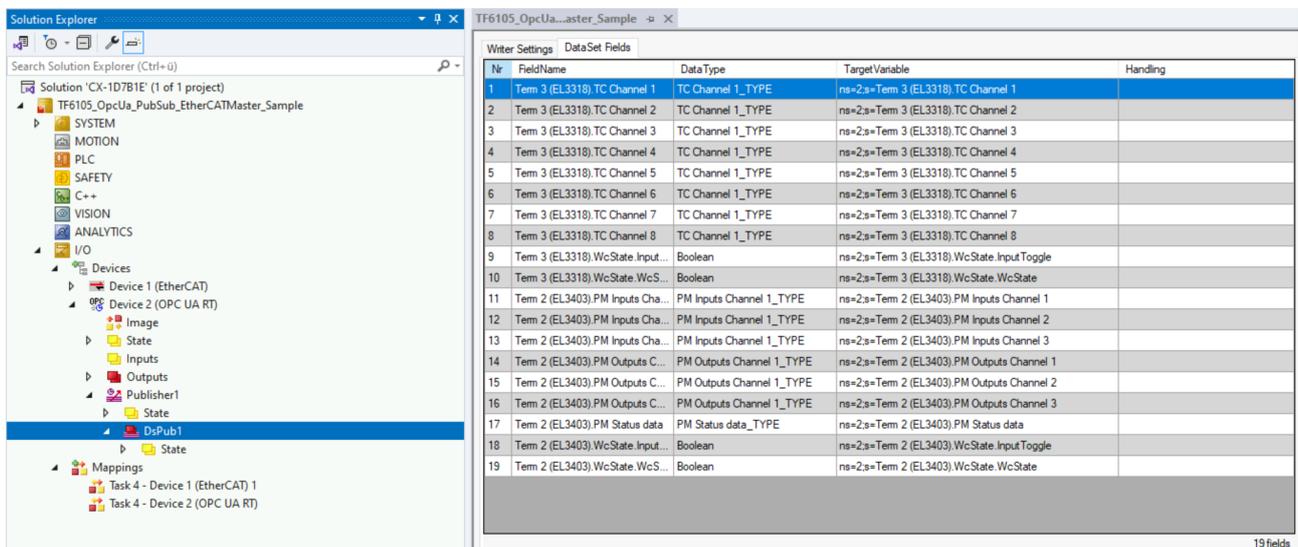
Importieren Sie diese Konfigurationsdatei auf das Gerät, das den Subscriber aufnehmen soll. Wenn das Gerät auch ein System ist, auf dem TwinCAT läuft, können Sie die Datei in das entsprechende TwinCAT-Projekt importieren, indem Sie die Schaltfläche **Import** auf dem OPC UA RT Device verwenden.

Sie können auch einen Subscriber in demselben Projekt erstellen, in dem auch der Publisher läuft. Die Steuerung fungiert also als Publisher UND Subscriber.

Nach dem Importieren der Konfigurationsdatei wird der Subscriber automatisch mit einem DataSet konfiguriert, der die Felder des Publishers enthält.



Sie können dann ein Mapping mit Variablen z. B. aus einer SPS-Anwendung erstellen oder über die Schaltfläche **Create PLC code** in der DataSet-Konfiguration automatisch SPS-Code generieren.



Mit dieser Operation wird eine GVL erstellt, die Variablen enthält, die über das Attribut "TcLinkTo" mit den Prozessabbildvariablen verknüpft sind, z. B.:

```
Subscriber1_DsPub1 # x
1  VAR_GLOBAL
2      {attribute 'TcLinkTo' := 'TIID^Device 1 (OPC UA RT)^Subscriber1^DsPub1^State^PubSubState'}
3      PubSubState AT %I* : OpcUaPubSubState;
4      {attribute 'TcLinkTo' := 'TIID^Device 1 (OPC UA RT)^Subscriber1^DsPub1^Inputs^Term 3 (EL3318)_TC Channel 1'}
5      Term_3_TC_Channel_1 AT %I* : OpcUa_TC_Channel_1_TYPE;
6      {attribute 'TcLinkTo' := 'TIID^Device 1 (OPC UA RT)^Subscriber1^DsPub1^Inputs^Term 3 (EL3318)_TC Channel 2'}
7      Term_3_TC_Channel_2 AT %I* : OpcUa_TC_Channel_1_TYPE;
8      {attribute 'TcLinkTo' := 'TIID^Device 1 (OPC UA RT)^Subscriber1^DsPub1^Inputs^Term 3 (EL3318)_TC Channel 3'}
9      Term_3_TC_Channel_3 AT %I* : OpcUa_TC_Channel_1_TYPE;
10     {attribute 'TcLinkTo' := 'TIID^Device 1 (OPC UA RT)^Subscriber1^DsPub1^Inputs^Term 3 (EL3318)_TC Channel 4'}
11     Term_3_TC_Channel_4 AT %I* : OpcUa_TC_Channel_1_TYPE;
12     {attribute 'TcLinkTo' := 'TIID^Device 1 (OPC UA RT)^Subscriber1^DsPub1^Inputs^Term 3 (EL3318)_TC Channel 5'}
13     Term_3_TC_Channel_5 AT %I* : OpcUa_TC_Channel_1_TYPE;
14     {attribute 'TcLinkTo' := 'TIID^Device 1 (OPC UA RT)^Subscriber1^DsPub1^Inputs^Term 3 (EL3318)_TC Channel 6'}
15     Term_3_TC_Channel_6 AT %I* : OpcUa_TC_Channel_1_TYPE;
```

[optional] Einrichten eines generischen MQTT-Client-Subscriber

Sie können jeden MQTT-Client für die Daten abonnieren, die im MQTT Message Broker veröffentlicht werden. In diesem Beispiel haben wir das MQTT/JSON-Datenformat verwendet, das eine ASCII-basierte, einfache Interpretation der Daten ermöglicht. Der folgende Abschnitt zeigt, wie das resultierende JSON-Format aussehen kann.

```
{
  "DataSetWriterId": 1,
  "PublisherId": "1",
  "SequenceNumber": 5228,
  "MinorVersion": 771501780,
  "Timestamp": "2024-06-12T10:14:04.734Z",
  "Status": 0,
  "Payload": {
    "Term 3 (EL3318).TC Channel 1": {
      "Status": {
        "Underrange": false,
        "Overrange": false,
        "Error": false,
        "TxPDO State": false,
        "TxPDO Toggle": false
      },
      "Value": 233
    },
    "Term 3 (EL3318).TC Channel 2": {
      "Status": {
        "Underrange": true,
        "Overrange": false,
        "Error": false,
        "TxPDO State": false,
        "TxPDO Toggle": false
      },
      "Value": 13720
    },
    ...
    ...
  }
}
```

6 Tutorials

Video-Tutorials zu diesem Produkt finden Sie auf unserer Website unter <https://www.beckhoff.com/tutorials>. Die Video-Tutorials bieten eine schnelle Einführung in das Produkt und die einzelnen Produktfacetten.



TwinCAT 3 C++

14.03.2024 | Multimedia

Tutorial: TwinCAT 3 C++ overview

Get an overview of the C++ integration into TwinCAT.

[Learn more →](#)



TwinCAT 3 Scope View

12.02.2024 | Multimedia

Tutorial: Chart synchronization in TwinCAT Scope

Learn how to dock and synchronize charts in TwinCAT Scope.

[Learn more →](#)



TwinCAT 3 OPC UA

12.02.2024 | Multimedia

Tutorial: Getting started with TF6100 TwinCAT 3 OPC UA Server

Learn how to configure the TwinCAT OPC UA Server.

[Learn more →](#)

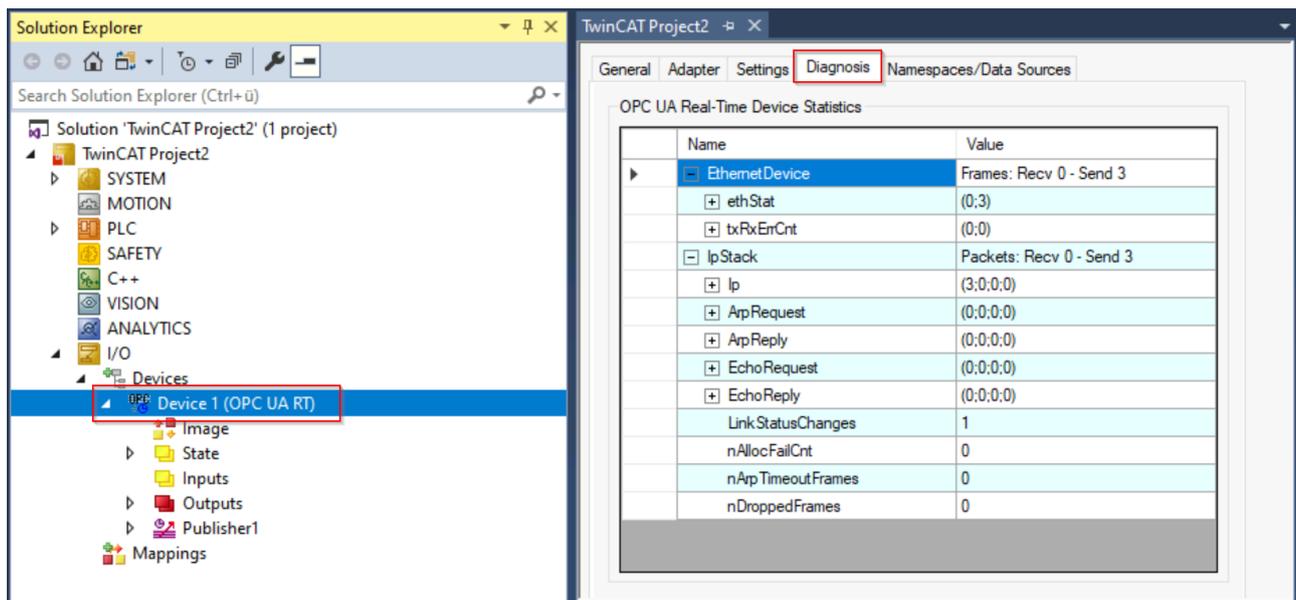
7 Anhang

7.1 Diagnose

Es gibt verschiedene Konfigurationsebenen, auf denen Sie Diagnoseinformationen finden können. Diese Informationen sind auf einer separaten Registerkarte des entsprechenden Konfigurationselements verfügbar.

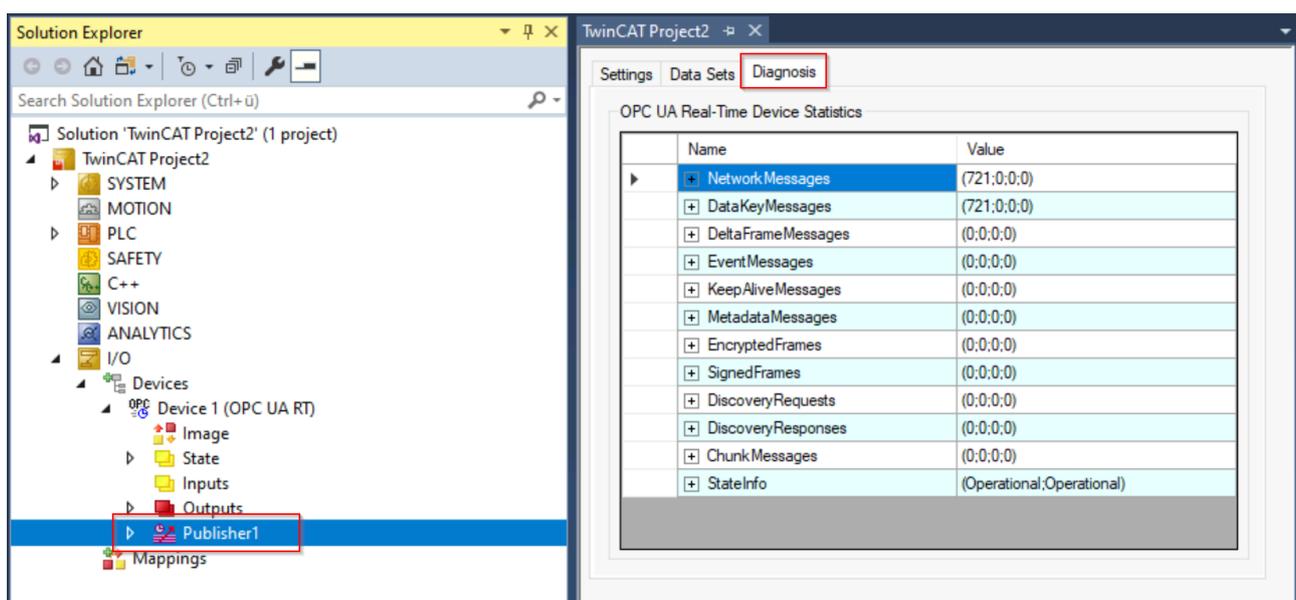
Device

Das OPC UA RT Device enthält die Registerkarte **Diagnosis**, die verschiedene Metriken enthält, um herauszufinden, ob und wie viele Pakete vom zugrunde liegenden IP-Stack gesendet oder empfangen werden.



Publisher/Subscriber

Jeder Publisher oder Subscriber enthält eine Registerkarte **Diagnosis**, die verschiedene Metriken für den jeweiligen Publisher oder Subscriber enthält.



TwinCAT-Fehlerliste

Der TwinCAT-OPC-UA-Pub/Sub-Treiber sendet auch Logging-Informationen an das TwinCAT-Fehlerlistenfenster, das von TwinCAT XAE aus zugänglich ist.

Wireshark

Wireshark ist ein gutes Software-Tool für fortgeschrittenes Debugging und Diagnose. Sie können auch Wireshark verwenden, um OPC UA Pub/Sub Pakete zu diagnostizieren.

Beispiel: Der folgende Wireshark-Trace zeigt den Verkehr, der durch den konfigurierten Publisher verursacht wird, der alle 1000 ms (Zeitstempel in der ersten Spalte) Pakete an die UDP-Multicast-Gruppe 224.0.0.1 sendet.

16	2.740000000	172.17.98.153	224.0.0.1	OPCUA UADP	52	4840	→	4840	Len=10
20	3.740000000	172.17.98.153	224.0.0.1	OPCUA UADP	52	4840	→	4840	Len=10
24	4.740000000	172.17.98.153	224.0.0.1	OPCUA UADP	52	4840	→	4840	Len=10
27	5.740000000	172.17.98.153	224.0.0.1	OPCUA UADP	52	4840	→	4840	Len=10
30	6.740000000	172.17.98.153	224.0.0.1	OPCUA UADP	52	4840	→	4840	Len=10
37	7.740000000	172.17.98.153	224.0.0.1	OPCUA UADP	52	4840	→	4840	Len=10
46	8.740000000	172.17.98.153	224.0.0.1	OPCUA UADP	52	4840	→	4840	Len=10
49	9.740000000	172.17.98.153	224.0.0.1	OPCUA UADP	52	4840	→	4840	Len=10
54	10.740000000	172.17.98.153	224.0.0.1	OPCUA UADP	52	4840	→	4840	Len=10

Jedes Paket enthält den konfigurierten DataSet und seine Variable (in unserem Fall "counter") vom Datentyp INT (entspricht Int16).

```

UADP Network Message
  > NetworkMessage Header
  Payload Header
  < DataSet Payload
    < DataSetMessage[0] (Data Key Frame Data)
      > DataSetFlags1
      < Data Key Frame
        Field Count: 1
        < DataSetField[0] (Variant Encoding)
          > Variant Encoding Mask: 4
          < Value (Int16)
            18889
  
```

● Wireshark-Plugin/Dissector

i Sie benötigen den OPC UA Pub/Sub Wireshark Dissector, wenn Sie die Pub/Sub-Nachrichten in einer lesbaren Weise verfolgen und interpretieren wollen.

7.2 Troubleshooting

Verhalten	Hinweise
Ich kann keine Daten auf meinem Subscriber empfangen und die TwinCAT-Fehlerliste zeigt mehrere Fehler an, die besagen "Invalid minor version received".	Dieser Fehler tritt in der Regel auf, wenn sich die Quelldaten geändert haben, z. B. wenn einem Publisher-DataSet zusätzliche Felder hinzugefügt wurden. In diesem Fall wird die Versionsinformation innerhalb des OPC UA Pub/Sub Pakets inkrementiert und es ist daher erforderlich, die neue Konfiguration dem Subscriber bekannt zu machen. Am einfachsten ist es, die aktualisierte Publisher-Konfiguration in eine <u>Konfigurationsdatei</u> [► 36] zu exportieren und sie in den Subscribern zu importieren.
Eine Verbindung zum Message Broker kann nicht hergestellt werden und die TwinCAT-Fehlerliste zeigt mehrere Fehler an, die besagen "Error connecting MQTT client MQTT_ERR_NOT_FOUND".	Bitte vergewissern Sie sich, dass Sie den richtigen Hostnamen oder die richtige IP-Adresse Ihres MQTT Message Brokers verwendet haben.
Eine Verbindung zum Message Broker kann nicht hergestellt werden und die TwinCAT-Fehlerliste zeigt mehrere Fehler an, die besagen "Error connecting MQTT client MQTT_ERR_CONN_TIMEDOUT".	Bitte vergewissern Sie sich, dass Sie den richtigen Hostnamen oder die richtige IP-Adresse Ihres MQTT Message Brokers verwendet haben und dass der Broker unter dieser Adresse erreichbar ist. Vergewissern Sie sich auch, dass Sie den richtigen TCP/IP-Port Ihres Message Brokers angegeben haben, z. B. 1883 oder 8883.
Eine Verbindung zum Message Broker kann nicht hergestellt werden und die TwinCAT-Fehlerliste zeigt mehrere Fehler an, die besagen "Error connecting MQTT client MQTT_ERR_TLS_VERIFY_FAIL".	Diese Fehlermeldung wird in der Regel ausgelöst, wenn der Pub/Sub-Treiber das Message Broker-Zertifikat nicht akzeptiert, was in der Regel mit der Verwendung eines falschen CA-Zertifikats zusammenhängt. Bitte stellen Sie sicher, dass Sie das richtige CA-Zertifikat angegeben haben.
Eine Verbindung zum Message Broker kann nicht hergestellt werden und die TwinCAT-Fehlerliste zeigt mehrere Fehler an, die besagen "Error connecting MQTT client MQTT_ERR_TLS_HANDSHAKE_FAIL".	Diese Fehlermeldung wird normalerweise ausgegeben, wenn der Message Broker Ihr Client-Zertifikat nicht akzeptiert. Er kann auch ausgelöst werden, wenn Sie überhaupt kein Client-Zertifikat angegeben haben. Vergewissern Sie sich, dass Sie das richtige Client-Zertifikat angegeben haben und dass es vom Message Broker als vertrauenswürdig eingestuft wird.
Eine Verbindung zum Message Broker kann nicht hergestellt werden und die TwinCAT-Fehlerliste zeigt mehrere Fehler an, die besagen "Error connecting MQTT client MQTT_ERR_TLS_KEY_NOTFOUND".	Diese Fehlermeldung wird normalerweise ausgegeben, wenn der Client-Key nicht gefunden werden kann. Stellen Sie sicher, dass der Client-Key unter dem Pfad existiert, den Sie in den MQTT-Verbindungseinstellungen angegeben haben.
Eine Verbindung zum Message Broker kann nicht hergestellt werden und die TwinCAT-Fehlerliste zeigt mehrere Fehler an, die besagen "Error connecting MQTT client MQTT_ERR_TLS_KEY_INVALID".	Diese Fehlermeldung wird normalerweise ausgegeben, wenn der Client-Key nicht geöffnet werden kann. Bitte stellen Sie sicher, dass Sie eine gültige Client-Key-Datei in den MQTT-Verbindungseinstellungen angegeben haben und dass Sie den Client-Key nicht mit dem Client-Zertifikat verwechselt haben.
Eine Verbindung zum Message Broker kann nicht hergestellt werden und die TwinCAT-Fehlerliste zeigt mehrere Fehler an, die besagen "Error connecting MQTT client MQTT_ERR_TLS_CERT_NOTFOUND".	Diese Fehlermeldung wird normalerweise ausgegeben, wenn das Client-Zertifikat nicht gefunden werden kann. Stellen Sie sicher, dass das Client-Zertifikat unter dem Pfad existiert, den Sie in den MQTT-Verbindungseinstellungen angegeben haben.

Verhalten	Hinweise
<p>Eine Verbindung zum Message Broker kann nicht hergestellt werden und die TwinCAT-Fehlerliste zeigt mehrere Fehler an, die besagen "Error connecting MQTT client MQTT_ERR_TLS_CERT_INVALID".</p>	<p>Diese Fehlermeldung wird normalerweise ausgegeben, wenn das Client-Zertifikat nicht geöffnet werden kann.</p> <p>Bitte stellen Sie sicher, dass Sie in den MQTT-Verbindungseinstellungen eine gültige Client-Zertifikatsdatei angegeben haben und dass Sie das Client-Zertifikat nicht mit dem Client-Key verwechselt haben.</p>
<p>Beim Setzen von TwinCAT in den Config Mode und anschließender Bestätigung des Load I/O Devices Dialogs, erhalte ich die Fehlermeldung „SAFEOP to OP of ... failed. Invalid parameter 0x9811070B.“</p>	<p>Durch den TwinCAT Statuswechsel wird auch der OPC UA Pub/Sub Treiber einem Kontextwechsel unterzogen. Da jedoch die SPS im Config Mode nicht läuft, wird diese Fehlermeldung angezeigt. Dasselbe Verhalten tritt auch auf, wenn man auf den Reload Devices Button in der TwinCAT XAE Toolbar klickt. Dieses Verhalten ist by-design.</p>

Trademark statements

Beckhoff®, TwinCAT®, TwinCAT/BSD®, TC/BSD®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, Safety over EtherCAT®, TwinSAFE®, XFC®, XTS® and XPlanar® are registered trademarks of and licensed by Beckhoff Automation GmbH.

Third-party trademark statements

Intel, the Intel logo, Intel Core, Xeon, Intel Atom, Celeron and Pentium are trademarks of Intel Corporation or its subsidiaries.

Microsoft, Microsoft Azure, Microsoft Edge, PowerShell, Visual Studio, Windows and Xbox are trademarks of the Microsoft group of companies.

Wireshark is a registered trademark of Sysdig, Inc.

Mehr Informationen:
www.beckhoff.com/tf6105

Beckhoff Automation GmbH & Co. KG
Hülshorstweg 20
33415 Verl
Deutschland
Telefon: +49 5246 9630
info@beckhoff.com
www.beckhoff.com

