

Plattform MX6 / PI

Softwareoption S118

CODESYS OPC UA PubSub

1 Identifikation

Identifikation	
Optionskennung	S118
Bestellnummer	S-05000319-0000
Kurzbezeichnung	CODESYS OPC UA PubSub
Kurzbeschreibung	Mit Hilfe dieser Softwareoption ist es möglich, Nachrichten über das von OPC UA Foundation definierte Pub/Sub Protokoll auszutauschen.
Revisionskennung Dokument	V1.0

2 Systemvoraussetzungen und Einschränkungen

Systemvoraussetzungen und Einschränkungen	
Unterstützte Plattformen oder Geräte	Berghof SPS Geräte der MX6 und PI-Plattform (z.B.: MC, CC, DC). Weitere Informationen bezüglich Verfügbarkeit und Kompatibilität finden Sie im Produktkatalog im Abschnitt Optionen.
Firmware	MX6-SPS ab Version 1.27.0, CODESYS ab 3.5 SP18 Patch 3 PI-SPS ab Version 1.3.1, CODESYS ab 3.5. SP18 Patch 4
Weitere Anforderungen	<ul style="list-style-type: none"> – IP Netzwerkschnittstelle Netzwerkzugriff – Publish/Subscribe von Nachrichten gemäß OPC 10000-14: OPC Unified Architecture Part 14: PubSub Release 1.04
Einschränkungen	<ul style="list-style-type: none"> – Maximale Größe einer NetworkMessage: 1500 – Die aktuelle Packetgröße und Größe der Nutzdaten (Payload) ist über die Konfiguration der jeweiligen ReaderGroup/WriterGroup und DataSet Bausteine konfigurierbar

3 Produktbeschreibung

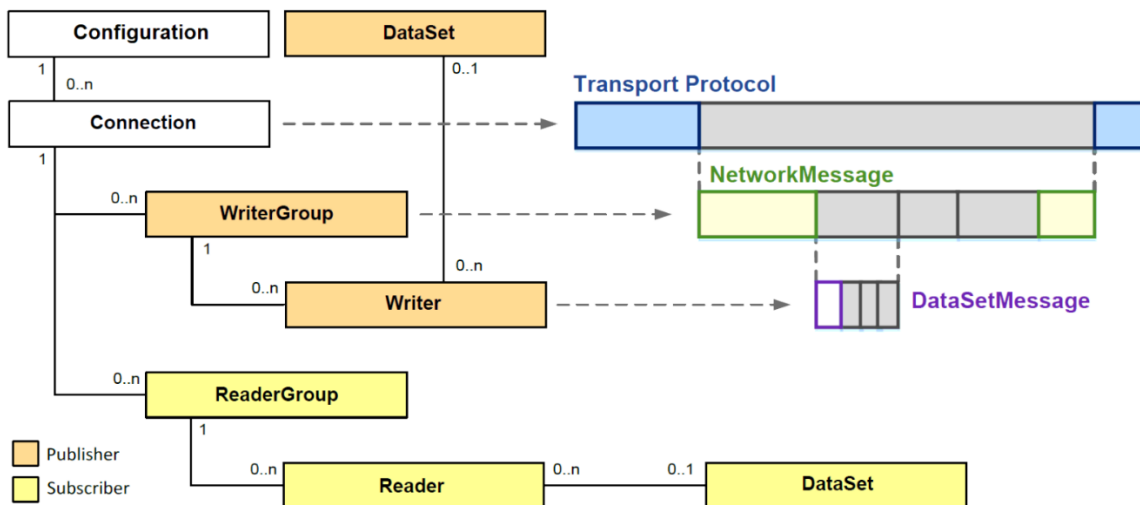
Diese Softwareoption schaltet die Lizenz für die CODESYS Bibliothek OPC UA PubSub für das Gerät frei.

Die Kommunikation über das OPC UA PubSub Protokoll stellt neben der Client/Server Kommunikation eine weitere Möglichkeit bereit, konform zu den Regeln der OPC UA Foundation Daten zwischen den Teilnehmern eines Netzwerkes auszutauschen. Die Struktur der Daten (DataSet) kann frei definiert werden und wird im Vorfeld zwischen Sender und Empfänger abgesprochen. Damit entfällt der Transport zusätzlicher (Meta-)Daten. Der Datentransfer erfolgt mit Hilfe der OPC UA PubSub Bibliothek über UDP/IP nach den Regeln, die für UADP festgelegt sind. Ein sogenannter Publisher veröffentlicht seine Daten an eine ihm unbekannte Anzahl von Subscriber. Sender und Empfänger kennen sich also nicht. Daher hat die Anzahl der Empfänger auch keinen Rückwirkung auf den Sender. Eine Rückmeldung an den Sender ob seine Nachrichten die Empfänger erreicht hat, kann prinzipbedingt nicht über das Protokoll erfolgen. Bei Bedarf muss eine solche Rückmeldung applikationsspezifisch verwaltet werden. Die Daten werden in einem binären Format gemäß den Regeln der OPC UA Foundation übertragen. Die Umwandlung der IEC Datentypen in die entsprechenden OPC UA Datentypen und umgekehrt übernimmt die OPC UA PubSub Bibliothek. Dabei folgt die Implementierung folgenden Profilen:

- Publisher: PubSub Publisher UADP Periodic Fixed Settings
- Subscriber: PubSub Subscriber UADP Periodic Fixed Settings

Die Länge der Nachrichten ist auf 1500 Bytes (MTU) beschränkt (Chunked NetworkMessages not supported). Solange die Regeln für ein Time Sensitive Network noch nicht zur Verfügung stehen, können harte Echtzeitbedingungen nicht eingehalten werden. Die Implementierung der OPCuA PubSub Bibliothek versucht aber die Jitter so klein wie möglich zu halten.

- Folgende Funktionsbausteine sind in der Bibliothek enthalten:
- DataSet: Funktionsbaustein zur Definition einer DataSetMessage
- Configuration: Funktionsbaustein Verwaltung gemeinsamer Ressourcen (Connection, Group, ...)
- RootDiagnostics: Funktionsbaustein für übergeordnete Diagnosedaten
- Connection: Funktionsbaustein zur Verwaltung der Verbindung zu Publisher und Subscriber
- ConnectionDiagnostics: Funktionsbaustein für Diagnosedaten der Verbindung (Connection)
- ReaderGroup: Funktionsbaustein zur Verwaltung von Reader Bausteinen (Bildung einer NetworkMessage aus DataSetMessages)
- ReaderGroupDiagnostics: Funktionsbaustein für Diagnosedaten einer ReaderGroup
- Reader: Funktionsbaustein zur Verwaltung eines DataSet Bausteins (Subscriber)
- ReaderDiagnostics: Funktionsbaustein für Diagnosedaten eines Reader Bausteins
- WriterGroup: Funktionsbaustein zur Verwaltung von Writer Bausteinen (Bildung einer NetworkMessage aus DataSetMessages)
- WriterGroupDiagnostics: Funktionsbaustein für Diagnosedaten einer WriterGroup
- Writer: Funktionsbaustein zur Verwaltung eines DataSet Bausteins (Publisher)
- WriterDiagnostics: Funktionsbaustein für Diagnosedaten eines Writer Bausteins



4 Technische Daten

Technische Daten	
Tasking	Unterstützung einer Hintergrundtask für den jeweiligen Connection Baustein
Unterstütztes Profil	Publisher: PubSub Publisher UADP Periodic Fixed Settings Subscriber: PubSub Subscriber UADP Periodic Fixed Settings

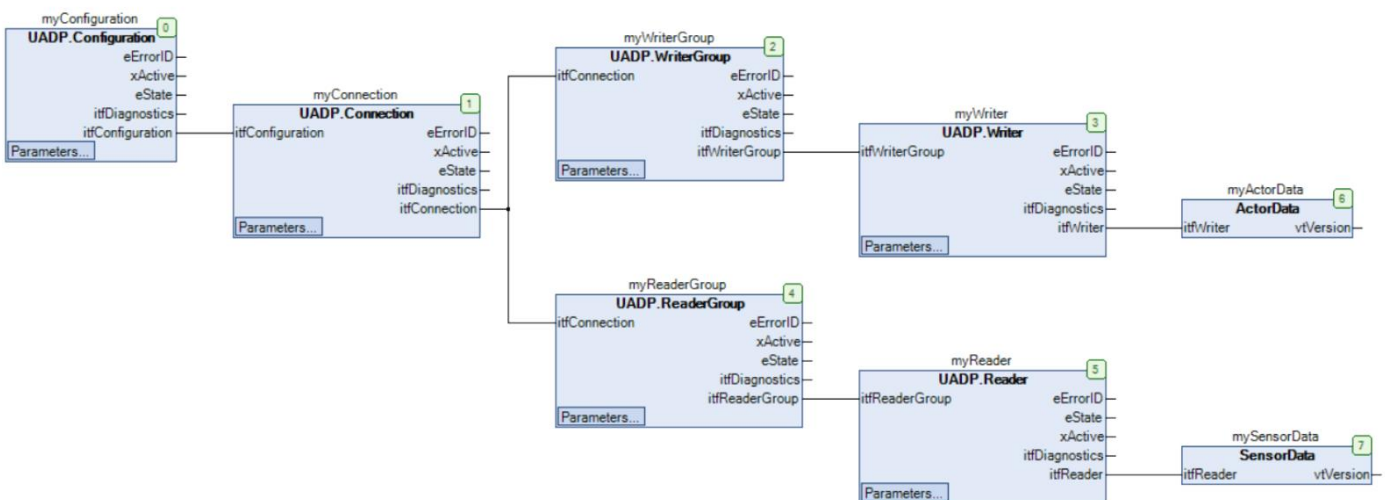
5 Quick Start Guide

Bei der Installation des Packages wird das Beispielprojekt „OPC UA PubSub SL Example.project“ in das gewählte Zielverzeichnis installiert. Die Applikation Device_1 demonstriert das Lesen einer NetworkMessage. Die Applikation Device_2 zeigt, wie Nachrichten über die enthaltenen Funktionsbausteine gesendet werden können.

5.1 Verwendung des Beispielprojekts

1. Hardware
 - Sie benötigen zwei Berghof SPS, welche über ein Netzwerk verbunden sind.
 - Passen Sie die Geräte in dem Projekt an Ihre Konfiguration an, indem Sie Device_1 und Device_2 auf die verwendete Hardware aktualisieren
2. Anpassungen im Projekt
 - Setzen Sie die IP-Adressen in der GVL GlobalIP auf die von Ihnen konfigurierte Hardware. Sie finden diese GVL im Bereich POU's.
 - Setzen Sie die Multicast Adresse auf eine freie Adresse in Ihrem Netzwerk. Hinweis: Wenn Sie sich nicht in einem privaten Netzwerk befinden, lassen Sie sich eine freie Adresse von Ihrem Netzwerk Administrator zuweisen
 - Setzen Sie den Port. Der voreingestellte Port 4840 ist der standardmäßig empfohlene Port für OPC UA PubSub
3. Loggen Sie sich auf beiden Steuerungen ein und starten Sie die Applikationen
4. Starten Sie sowohl Sender (Device_2: PLC_PRG.xEnable := TRUE) als auch Empfänger (Device_1: Communication_PRG.xEnable := TRUE)
5. Jetzt sollten Sie auf Seite des Empfängers sehen, wie sich die Werte des Sinus ändern Die anderen Werte im Sender können Sie manuell ändern und die Änderungen im Empfänger sehen
6. Vorschlag zum Erweitern des Beispiels (Hinzufügen einer Variable zum übertragenen DataSet)
 - Erweitern Sie SensorDataSet._aIndex um einen Eintrag. Wählen Sie einen beliebigen Datentyp
 - Passen Sie in SensorDataSet.Init die Version des DataSet an, indem Sie den Datumseintrag aktualisieren
 - Erstellen Sie eine Variable des gewählten Datentyps in PLC_PRG des Sender und weisen Sie diese in txSensorDataSet.PrepareValues zu
 - Wiederholen Sie diesen Schritt auf der Empfänger-Seite in rxSensorDataSet.PrepareValues. Erstellen Sie auch hier zuvor eine neue Variable desselben Typs
 - Nach einem erneuten Download sollte auch diese Variable mit übertragen werden

Beispiel für eine typische Konfiguration:



Ihre Ansprechpartner erreichen Sie unter:

Vertriebsteam | T +49.7121.894-131 | controls@berghof.com

Berghof Automation GmbH | Arbachtalstraße 26 | 72800 Eningen | www.berghof-automation.com
SoftwareOption_S118_CODESYS_OPcUA_PubSub_V1_DE