



OPC UA Pub/Sub für das IIoT

REIBUNGSLOSER DATENAUSTAUSCH

OPC UA hat sich mittlerweile als führender Standard für den Datenaustausch in Industrie-4.0-Anwendungen etabliert. So lassen sich Daten sicher und problemlos zwischen sämtlichen Systemen innerhalb eines Unternehmens austauschen. Ein neues OPC-UA-Embedded-Toolkit ermöglicht nun Systemen mit geringem verfügbarem Speicher und eingeschränkter Rechenleistung die einfache Integration.

TEXT: Thomas Rummel, Softing Industrial Automation **BILDER:** Softing Industrial Automation; iStock, triloks

Das bisherige Client/Server-Modell in der OPC-UA-Spezifikation erfüllt nicht mehr alle Anforderungen einer IIoT- beziehungsweise Industrie-4.0-Kommunikation. So stößt man auf Szenarien, die die Umsetzung der Kommunikations-

prinzipien Einer-an-Viele (One-to-Many), Viele-an-Einen (Many-to-One) und Viele-an-Viele (Many-to-Many) erfordern. Die neue Version 1.04 der Spezifikation der OPC Foundation (OPC UA Part 14 – PubSub RC 1.04.24, OPC Foun-

dation, 2.9.2017) wird daher um das Kommunikationsmodell Publisher/Subscriber (Pub/Sub) erweitert, das sich besser für IIoT-Implementierungen eignet als die im bisherigen OPC-UA-Standard definierte Client/Server-Architektur.



Pub/Sub-Kommunikation

Beim Client/Server-Kommunikationsmodell sendet ein Client eine Anfrage zum Server. Der Server bearbeitet die Anfrage und sendet dann eine Antwort zurück an den Client. Beim Pub/Sub-Kommunikationsmodell agieren Publisher und Subscriber unabhängig voneinander. Der Publisher versendet eigenständige Nachrichten, die von einem oder mehreren Subscribern empfangen werden können. Das Senden der Nachrichten durch den Publisher erfolgt unabhängig davon, ob Subscriber für den Empfang der Nachrichten existieren. Die Verbindung zwischen Publishern und Subscribern wird über eine nachrichtenorientierte Zwischenschicht hergestellt. Das ist

eine Software- oder Hardware-Infrastruktur, mit der Nachrichten in einem verteilten System empfangen und gesendet werden können und die eine einfache Anbindung von OPC UA an IoT-Cloud-Dienste ermöglicht. Die Zwischenschichten sind in der OPC-UA-Pub/Sub-Spezifikation definiert und basieren auf den Protokollen UDP (User Datagram Protocol), AMQP (Advanced Message Queuing Protocol) und MQTT (Message Queuing Telemetry Transport).

Die UDP-Zwischenschicht nutzt die Multicast-Funktionalität von UDP. Sie wird über die Infrastruktur des Ethernet-Netzes und den UDP/IP-Protokoll-Stack realisiert und ermöglicht einen zyklischen, determi-



Publisher-Subscriber-Modell aus der OPC-UA-Spezifikation

nistischen Datentransfer vom Publisher zu den Subscribern. Die AMQP- und MQTT-Zwischenschichten benutzen eine Message-Broker-Software zur Verbindung von Publisher und Subscriber. Eine TSN-Zwischenschicht gibt es in der neuen Spezifikation noch nicht, sie soll aber für die nächste Version definiert werden. Die OPC-UA-Pub/Sub-Spezifikation definiert auch das Format der Nachrichten, die über die unterschiedlichen Zwischenschichten verschickt werden und entweder binär (UADP, UA Datagram Protocol) oder mit JSON (JavaScript Object Notation) codiert werden. Zusätzlich zum Kommunikationsprotokoll wird in der OPC-UA-Pub/Sub-Spezifikation auch ein optionales Konfigurationsmodell für Publisher und Subscriber definiert. Zum Zugriff auf das Konfigurationsmodell wird die OPC-UA-Client/Server-Kommunikation genutzt. Ein Publisher oder Subscriber muss also für die Konfiguration zusätzlich einen Server realisieren.

Der Publisher wird über die PublishedDataSet-Objekte und die DataSetWriter-Objekte konfiguriert. Ein PublishedDataSet-Objekt beschreibt die gesendeten Daten und ein DataSetWriter-Objekt die Transporteinstellungen zum Versenden der Daten. Analog zum

Publisher wird der Subscriber über die SubscribedDataSet-Objekte und DataSetReader-Objekte konfiguriert. Ein DataSetReader-Objekt beschreibt die Transporteinstellungen zum Empfangen der Daten und ein SubscribedDataSet die Verteilung der empfangenen Daten im Subscriber.

Pub/Sub wird attraktiv

Mit der OPC-UA-Pub/Sub-Spezifikation wird OPC UA sehr attraktiv für den Einsatz in Steuerungen und Feldgeräten. OPC UA bietet nun für den zyklischen, deterministischen Datenaustausch Mechanismen an, die vergleichbar mit Ethernet/IP oder Profinet RT sind. Aus diesem Grund hat Softing seine OPC-UA-Toolkit-Familie um ein OPC-UA-Embedded-Toolkit erweitert.

Das neue Toolkit wurde speziell für Embedded-Systeme mit geringem verfügbaren Speicher und eingeschränkter Rechenleistung entwickelt. So werden bei der Implementierung eines OPC-UA-Publishers mit dem OPC-UA-Embedded-Toolkit nur 150 KB Flash-Speicher für den Code und die statischen Daten und 10 KB Arbeitsspeicher für die Laufzeitdaten benötigt. Die Implementierung eines OPC-UA-Pub-

lishers mit einem OPC-UA-Server nach dem Nano-Embedded-Device-Server-Profil erfordert dagegen 230 KB Flash- und 40 KB Arbeitsspeicher. OPC UA bietet auch auf Embedded-Plattformen eine gute Performance. Auf einem Raspberry Pi 1 können von einem mit dem OPC-UA-Embedded-Toolkit erzeugten Server über 140.000 Variablenwerte in der Sekunde gelesen oder geschrieben werden. Das OPC-UA-Embedded-Toolkit wird als C-Quellcode geliefert. Der Code kann einfach auf Embedded-Betriebssysteme portiert werden, die einen IP-Stack beinhalten und Tasks mit Synchronisationsmechanismen bieten. Es beinhaltet beispielsweise die Unterstützung für FreeRTOS auf einem Espressif-ESP32-Prozessor mit 520 KB RAM. Mit der im ersten Quartal 2018 erscheinenden Version 1.10 des OPC-UA-Embedded-Toolkits werden die drei Embedded-OPC-UA-Server-Profile der OPC-UA-Spezifikation unterstützt: Embedded-UA-Server, Micro-Embedded-Device-Server und Nano-Embedded-Device-Server. Eine Testversion, mit der die Entwicklung mit dem OPC-UA-Embedded-Toolkit unter Linux und Windows evaluiert werden kann, ist auf der Softing-Industrial-Website (<https://industrial.softing.com>) zum Download verfügbar. □