

IP Autokonfiguration für industrielle Netzwerkkomponenten in Industrie 4.0 Anwendungen basierend auf der OPC UA Protokollsuite

Markus Rentschler	Henning Trsek	Lars Dürkop
Business Unit Networking Balluff GmbH Schurwaldstrasse 9 73765 Neuhausen, Germany Markus.Rentschler@balluff.de	Industrial Security rt-solutions.de GmbH Oberländer Ufer 190a 50968 Köln, Germany trsek@rt-solutions.de	Institut für industrielle Informationstechnik (inIT) Liebigstr. 87 32657 Lemgo, Germany lars.duerkop@hs-owl.de

Abstract: Die Inbetriebnahme oder der Austausch verschiedener vernetzter Komponenten in industriellen Netzwerken setzt immer eine Erstkonfiguration der betroffenen Geräte voraus. Diese Erstkonfiguration von industriellen Automatisierungsgeräten mit IP-Zugangsparametern ist immer noch sehr heterogen durch inkompatible Ansätze gelöst, wobei einem einheitlichen standardisierten Ansatz vorwiegend politische Gründe im Wege standen. Im Rahmen der Industrie 4.0 Standardisierungsbemühungen besteht die realistische Chance, diese politische Hürde überwinden zu können. Die vorliegende Arbeit gibt einen Überblick über bestehende Ansätze sowohl der IT- als auch der Automatisierungsbranche zur IP-Erstkonfiguration und schlägt einen Lösungsansatz vor, der beispielsweise im Rahmen der OPC UA-Spezifikation umgesetzt werden könnte und somit in einem einheitlichen und standardisierten Mechanismus zur Geräteerkennung und IP-Adressvergabe resultieren könnte.

1 Motivation

Bei der Inbetriebnahme oder dem Austausch von netzwerkfähigen Komponenten in industriellen Netzwerken ist immer eine Erstkonfiguration der betroffenen Geräte mit ihren IP-Zugangsparametern notwendig, während im laufenden Betrieb für Überwachung und Diagnose der Anlage jederzeit der Zugriff auf diese Konfigurationsparameter erforderlich ist. In diesem Kontext müssen typischerweise die folgenden Anwendungsfälle (Use Cases) abgedeckt werden können:

1. **Geräteerkennung:** Für die Verwaltung und Diagnose bestehender Anlagennetze müssen ohne vorherige Kenntnis der Netzwerktopologie und Geräteidentitäten im Netzwerk angeschlossenen Geräte über eine Suchfunktion erkennbar und ansprechbar sein und Ihre jeweiligen Identifikationsparameter an die anfragende Station zurückliefern.
2. **Geräteinstallation:** Beim Einbringen von neuen, bislang unbekannten, Komponenten in das Netzwerk muss eine Erreichbarkeit auch ohne IP-Konfiguration und eine nachvollziehbare IP-Parametervergabe an diese Geräte möglich sein.
3. **Gerätetausch:** Wenn eine (fehlerhafte) Komponente durch ein baugleiches Gerät ersetzt wird, muss die Identität des ausgetauschten Geräts eindeutig ermittelbar sein, um anschließend einen automatischen Konfigurationstransfer auf das ausgetauschte Gerät durchführen zu können.
4. **IP-Parametervergabe:** Manuell oder automatisiert muss entweder durch ein autorisiertes zentrales Konfigurationswerkzeug oder durch ein anderes Gerät im Netzwerk einem Gerät abhängig von bestimmten Vergabekriterien ein IP-Parametersatz zugewiesen werden können. Ein IP-Parametersatz besteht dabei im Wesentlichen aus einer eigenen IP-Adresse, der Netzmaske und einer IP-Adresse eines Routers, darüber hinaus aus einem eindeutigen Gerätenamen.
5. **Namensauflösung:** In modernen Netzwerken werden Geräte zunehmend über eindeutig vergebene Namen identifiziert und angesprochen, die schlecht durch Menschen interpretierbare IP- und MAC-Adresse tritt dabei in den Hintergrund. In der Installationstechnik kann dieser Ansatz konzeptionell sehr gut mit den etablierten Betriebsmittel-Kennzeichnungssystemen verbunden werden.

Diese Prozesse sollten in Industrie-4.0-Umgebungen möglichst autonom ablaufen können, wodurch Anforderungen an netzwerkfähige Komponenten und die übergeordneten Anwendungen entstehen. Hierzu müssen als notwendige Grundlage bestimmte Funktionalitäten in den Protokoll-Stacks der Netzwerkschnittstellen der Geräte vorgehalten werden. Die wichtigste Funktion dabei ist die Fähigkeit zur Erkennung des Gerätes im Netzwerk und dessen Konfiguration mit einer IP-Adresse, auch wenn es bis dahin noch über keine dediziert vergebene IP-Konfiguration verfügt.

Um diese Anforderungen zu adressieren, werden bislang oftmals Protokolle aus der IT-Welt herangezogen, wie beispielsweise das *Dynamic Host Configuration Protocol* (DHCP), und es existieren eine Anzahl Feldbus- und herstellerabhängige Mechanismen und Protokolle, die meist nicht kompatibel zueinander sind und auch unterschiedlichen Anwendungsparadigmen folgen. Für den Anwender von heterogenen Automatisierungsnetzwerken bedeutet dies, mit einer ebenso heterogenen Landschaft von Anwendungstools umgehen und diese pflegen zu müssen. Für den grundlegenden Fall der Erstinbetriebnahme, wenn also die Geräte im Netzwerk erkannt und für eine Erreichbarkeit über das TCP/IP-Netzwerk mit einer IP-Adresse versehen werden müssen, ist das besonders aufwändig.

Zwei typische Beispiele für diese Toollandschaft sind in Abbildung 1 dargestellt, Hilscher NetIdent (www.hilscher.com) und das das Primary Setup Tool (www.siemens.com). Weitere Beispiele für häufig eingesetzte Tools sind das Westermo IP config protocol, (www.westermo.com), Winford ETH32 UDP conf (www.winford.com), Turck IP Address Tool (www.turck.com) oder auch Control Port Vision DX (www.comtrol.com).

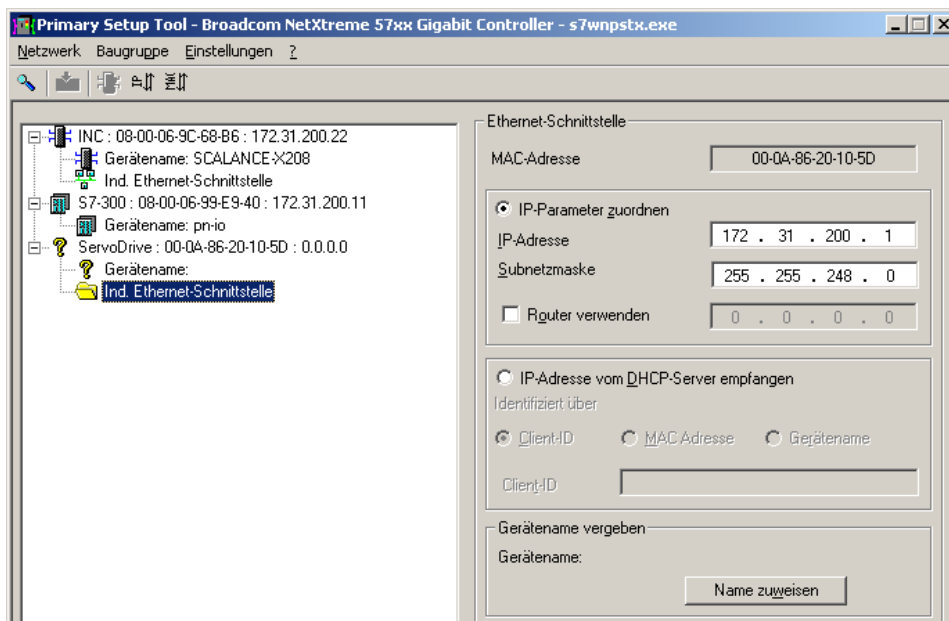
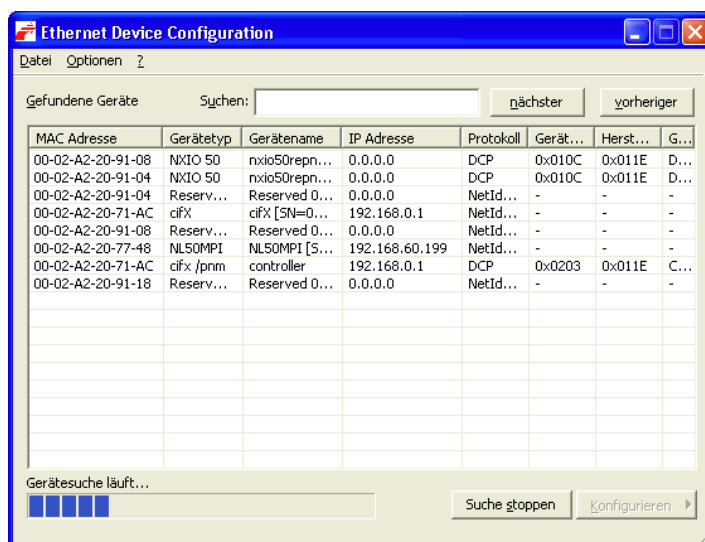


Abbildung 1: Typische Discovery- und Konfigurationstools im Bereich industrieller Netzwerke

Obwohl eine zentrale Anforderung an Komponenten für Automatisierungsanlagen der Industrie 4.0 die Selbstkonfiguration und Selbstdiagnose ist [1], gibt es bislang keinen allgemein akzeptierten und definierten Mechanismus für die Geräteerkennung und IP-Erstkonfiguration in Automatisierungsnetzwerken [2], der auch eine einheitliche Toolkette ermöglichen würde.

Die vorliegende Arbeit analysiert daher die bestehenden Anwendungsfälle und schlägt als Lösungsansatz einen Mechanismus zur Geräteerkennung und IP-Adressvergabe vor, der im Kontext der Industrie-4.0-Definitionen standardisiert werden könnte. Der vorgestellte Lösungsansatz soll dabei möglichst auf bestehenden Verfahren aufsetzen und in die OPC UA Protokollsuite eingebettet werden können, da hierdurch möglicherweise eine realistische Chance zur Standardisierung entstehen könnte.

2 Automatisierungstopologie

Das betrachtete Anwendungsszenario für diese Arbeit ist eine wandlungsfähige Produktionsanlage basierend auf einem modularen mechatronischen Ansatz. Die Gesamtarchitektur ist in Abbildung 2 veranschaulicht und besteht aus flexiblen Modulen, die als intelligente und autonome Fertigungszellen bezeichnet werden können und somit den bekannten Industrie 4.0 Konzepten folgen.

Die Module verfügen über Selbst-Konfigurations- und Selbst-Diagnosemechanismen und können sich somit flexibel und autonom an sich ändernde Umgebungsbedingungen anpassen. Die Module können hinzugefügt oder entfernt werden, um den Fertigungsprozess je nach Anforderung anzupassen. Hierzu ist jedes Modul mit einer eigenständigen Speicherprogrammierbaren Steuerung (SPS) ausgestattet, die wiederum auf dezentrale IO Geräte zugreift um Sensorwerte zu erfassen und Aktorwerte zu schreiben. Hierbei müssen die in Abschnitt 1 beschriebenen Anwendungsfälle abgedeckt werden.

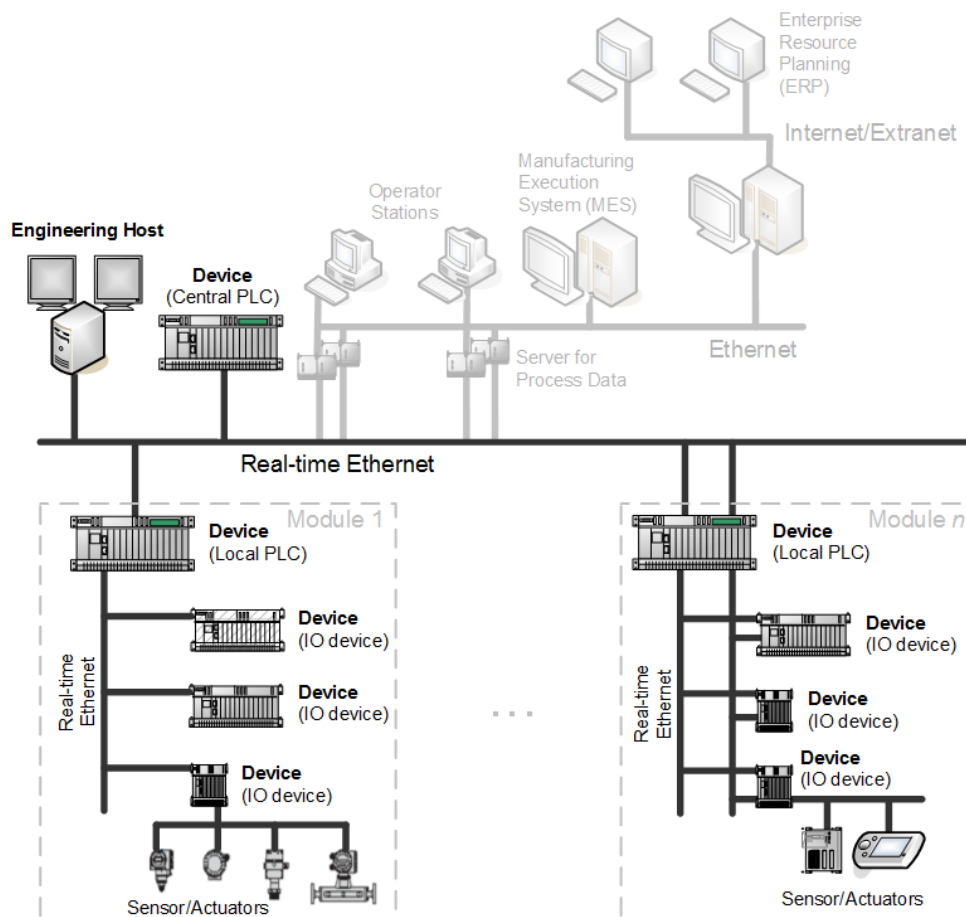


Abbildung 2: Architektur, Topologie und Terminologie in einer modularisierten Fertigungsumgebung

Im vorliegenden Kontext dieser Arbeit werden als „Hosts“ jene Instanzen bezeichnet, welche die Funktion der IP-Parametervergabe an „Devices“ bereitstellen, während „Devices“ die Geräte sind, die für eine derartige IP-Parametervergabe über das Netzwerk bereitstehen. Dabei wird zwischen „Device-initiierten“ und „Host-initiierten“ Mechanismen unterschieden. Während bei den IT-Lösungen vorwiegend „Device-initiierte“ Verfahren wie DHCP und ZeroConf etabliert sind, ist in Automatisierungsumgebungen der „Host-initiierte“ Ansatz verbreitet, meist manuell von einem Engineering Host-PC ausgehend. Als Beispiel sei hier PROFINET DCP und das zugehörige Tooling angeführt.

3 Bewertung bestehender IP-Konfigurationsverfahren

In [2] wurde zu dem vorliegenden Sachverhalt bereits eine umfassende Recherche über existierende Verfahren durchgeführt. Das Ergebnis ist in Tabelle 1 dargestellt und zeigt die wesentlichen relevanten Eigenschaften der Verfahren zusammenfassend auf.

Tabelle 1: Bestehende Verfahren zur Adressvergabe und deren Bewertung

	DHCPv4	PROFINET DCP	„ZeroConf“	IPv6
Erreichbarkeit ohne IP-Konfiguration	Eingeschränkt Device-initiiert	Host-initiiert (L2)	Über LinkLocal-Adresse (L3)	Über LinkLocal-Adresse (L3)
Geräteerkennung	Device-initiiert	Host-initiiert (L2)	Über ICMP (L3)	Über ICMP (L3)
Geräteidentifikation	MAC-Adresse, Systemname	MAC-Adresse, Systemname, Position in der Topologie	MAC-Adresse	MAC-Adresse
IP-Parametrierung	Device-initiiert	Host-initiiert	Host-initiiert über zusätzlichen Request-Response-Mechanismus, z.B. SNMP	Device-initiiert über DHCPv6, Host-initiiert über zusätzlichen Request-Response-Mechanismus, z.B. SNMP

Bei Betrachtung der obigen Matrix wird deutlich, dass IPv6 mit der *Stateless Address Autoconfiguration* nach [RFC 4862](#) bereits die breiteste funktionale Abdeckung mitliefert. Leider ist IPv6 in Automatisierungsnetzwerken noch nicht verbreitet und weist ebenso wie IPv4 auch Definitionslücken für die einheitliche hostbasierte IP-Parametrierung auf.

Naheliegender wäre es daher, für IPv4 Mechanismen zu definieren, die sich an IPv6 anlehnen. Dieser Ansatz wird durch „ZeroConf“ nach RFC 3927 (*Dynamic Configuration of IPv4 Link-Local Addresses*) verfolgt und würde durch den weitgehend identischen konzeptionellen Ansatz auch die Migration zwischen oder den Parallelbetrieb von IPv4 und IPv6-Netzwerken erleichtern.

Analog zu IPv6 (RFC 4291) hat die IANA (Internet Assigned Numbers Authority) für IPv4 den Bereich 169.254.0.0-169.254.255.255 definiert, in dem ein Netzwerkinterface sich selbst automatisch eine derartige, sog. „Link-Local“ IP-Adresse zuweisen kann, ohne Konflikte mit routingfähigen Adressen zu verursachen. RFC 3927 (*Dynamic Configuration of IPv4 Link-Local Addresses*) beschreibt das diesbezügliche Protokoll.

Nachdem dem Netzwerkadapter auf diese Weise eine IP-Adresse zugewiesen wurde, kann das Gerät TCP/IP verwenden, um mit anderen Geräten zu kommunizieren, die Verbindung zu demselben LAN haben und auch für die automatische Zuweisung privater IP-Adressen konfiguriert sind oder deren IP-Adresse manuell auf den Adressbereich 169.254.x.y mit einer Subnetzmaske 255.255.0.0. konfiguriert ist. Die automatische Zuweisung privater IP-Adressen sollte standardmäßig aktiviert sein, wenn keine IP-Adresse aus einer anderen Quelle (manuelle Konfiguration oder DHCP) vorliegt.

Die im IT-Bereich entwickelten „ZeroConf“-Mechanismen werden dort schon länger sehr erfolgreich eingesetzt (u.a. Apple's „Bonjour“) und teilweise bereits für die Automatisierungstechnik adaptiert, insbesondere mDNS für die OPC UA-Protokollsuite.

Im Gegensatz zu MAC-Layer-basierten Verfahren wie PROFINET-DCP würde dieser „ZeroConf“-Ansatz über die selbstzugewiesene Link-Local-IP-Adresse von Anfang an volle TCP/IP-Konnektivität mit anderen Gerät im lokalen Subnetzwerk ermöglichen. Sowohl für den IPv4- als auch den IPv6-basierten Ansatz fehlt ergänzend lediglich noch die Definition eines einheitlichen Request-Response-Mechanismus zur hostinitiierten (IP-)Parametervergabe. Hierfür könnten verschiedene existierende Mechanismen ergänzt werden, wie beispielsweise SNMP oder OPC UA-Konfigurationsobjekte.

Im Falle von SNMP müsste eine IP-Parameter-MIB definiert werden, die neben dem Lesen auch das Schreiben der Parameter im Gerät erlaubt. Im Falle von OPC UA könnte die Companion Specification „Device Integration“ um IP-Parameter-Objekte erweitert werden.

Der Anwendungsfall Gerätetausch ist laut [3] zufriedenstellend mittels eindeutiger Topologie-Erkennung lösbar, was ebenso wie in PROFINET-DCP über das Standard-Protokoll LLDP gelöst werden kann.

4 OPC UA

OPC UA ist eine Plattform-unabhängige Middleware-Technologie für den industriellen Einsatz, welche sich insbesondere für zukünftige Industrie 4.0-Architekturen eignet. In diesem Abschnitt wird eine Erweiterung des OPC UA Discovery-Dienstes beschrieben, wodurch dieser als alternative Methode zum Verteilen von IP-Adressen eingesetzt werden kann.

4.1 OPC UA Hintergrund

Im Allgemeinen ist OPC UA darauf angelegt, Interoperabilität zwischen heterogenen System zu ermöglichen - auch wenn diese verschiedene Arten von Netzwerken einsetzen. Hierzu definiert OPC UA einerseits Methoden für die Modellierung von Daten und andererseits für den Transport von Daten. Die Modellierung erfolgt Objekt-orientiert, wobei OPC UA selbst nur die grundlegenden Strukturen vorschreibt ohne dabei eine bestimmte Semantik der Daten zu definieren. Für Letzteres müssen Standardisierungsgremien eigene domänenspezifische Informationsmodelle erarbeiten und spezifizieren.

Die OPC UA-Architektur orientiert sich am Server/Client-Paradigma. Jeder Server kann mehrere Endpunkte enthalten, über dessen Adressräume der Zugriff auf die nach dem zugrundeliegenden Informationsmodell modellierten Daten ermöglicht wird. Ein Client kann den Adressraum eines Endpunktes durchsuchen und somit auf die für ihn relevanten Daten zugreifen.

4.2 OPC UA Discovery - global und lokal

Der OPC UA Discovery-Dienst ist in Teil 12 der OPC UA-Spezifikation beschrieben. In diesem werden verschiedene Methoden definiert, durch welche OPC UA Clients Informationen über vorhandene Server einholen können. Es können zwei verschiedene Discovery-Verfahren unterschieden werden: Beim globalen Discovery-Modus registriert sich jeder Server selbstständig bei einer zentralen Instanz, welche Global Discovery Server genannt wird. Bei dieser Variante muss allerdings die Adresse des Global Discovery Servers bei jedem Client vorkonfiguriert werden.

Bei dem zweiten Discovery-Verfahren ist eine solche manuelle Konfiguration nicht notwendig - hier werden sogenannte Local Discovery Server eingesetzt. Deren Einschränkung auf lokale Subnetze ist für die meisten automatisierungstechnischen Anlagen wie der SFWL nicht relevant, da hier die Netzwerke in der Regel nur aus einem Subnetz bestehen. Der lokale Discovery-Prozess von OPC UA basiert auf dem Protokoll Multicast DNS (mDNS) gemäß RFC 6762. Der Ablauf wird exemplarisch in Abbildung 3 gezeigt.

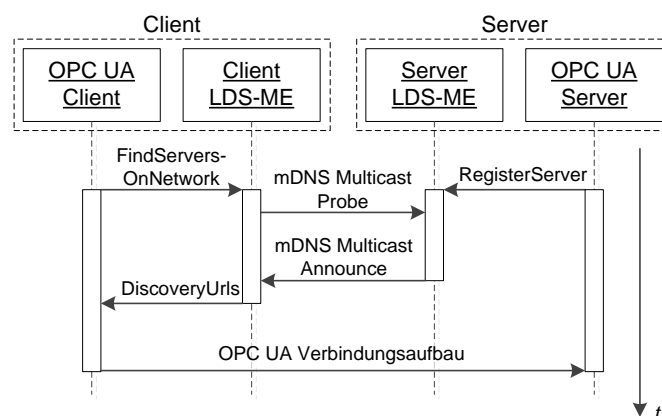


Abbildung 3: Lokales Discovery Verfahren in OPC UA

Jedes OPC UA-Gerät, sowohl Server als auch Client, enthält bei dieser Variante einen Local Discovery Server mit Multicast-Erweiterungen (LDS-ME). Um vorhandene Server im Netz zu entdecken, sendet ein Client ein FindServersOnNetwork-Kommando an seinen eigenen LDS-ME, welcher daraufhin eine Probe-Nachricht an die im mDNS-Standard definierte Multicast-Adresse schickt. Die serverseitigen LDS-MEs antworten mit einer Announce-Nachricht, welche die DiscoveryURL des OPC UA Servers enthält. Mittels der DiscoveryURL kann der OPC UA Client eine direkte Verbindung zum OPC UA Server aufbauen. Aus technischer Sicht beinhalten die Multicast-Nachrichten DNS Service Records (SRV Records). Im Allgemeinen werden diese dazu genutzt, um Netzwerkdienste zu suchen und anzubieten. Das Service Field als Bestandteil eines SRV Records gibt an, um was für einen Dienst es sich handelt, der gesucht bzw. angeboten wird. Zum Beispiel wird im OPC UA Standard für das lokale Discovery der Dienst `_opcua-tcp` definiert. Alle LDS-ME, welche TCP unterstützen, geben diese Kennung in ihrem Service Field an. Die DiscoveryURL wird nicht in einem SRV Record, sondern mittels eines DNS Text (TXT) Resource Record übermittelt.

Interessanterweise wird in Teil 12 der OPC UA-Spezifikation eingangs DHCP als Referenz aufgeführt, aber dann nicht weiter darauf eingegangen. In [4] wird im Kontext der Autokonfiguration mit OPC UA für PROFINET-Geräte DHCP ebenfalls erwähnt.

5 Lösungsvorschlag

Für eine zukünftige standardisierte und Industrie 4.0-fähige Netzwerkschnittstelle für TCP/IP-basierte Automatisierungsgeräte schlagen wir daher zwei Möglichkeiten vor. Die IP-Autokonfiguration analog zu den in IPv6 integrierten Mechanismen und eine automatisierte IP-Adressvergabe, die in das OPC UA Discovery eingebettet ist. Beide Ansätze werden nachfolgend erläutert und diskutiert.

5.1 IP-Autokonfiguration mit Link-Local-Adresse

Analog zu den Mechanismen in IPv6-basierten Netzwerkgeräten soll für TCP/IP-Netzwerkschnittstellen von Devices basierend auf IPv4 das „ZeroConf“-Protokoll gemäß RFC 3927 (*Dynamic Configuration of IPv4 Link-Local Addresses*) implementiert werden. Falls nach dem Booten des Devices keine IP-Parameter über andere Mechanismen (z.B. bereits vorkonfiguriert im NVM oder über DHCP) bezogen werden können, soll das Device sich selbst eine „Link-Local“-Adresse automatisch zuweisen. Dabei macht sich das Gerät durch Senden eines ARP Announcements im lokalen Netzwerksegment bekannt. Anschließend kann im lokalen Netzwerksegment über die normalen TCP/IP-basierten Protokolle, wie bspw. auch OPC-UA, mit dem Gerät kommuniziert werden.

Da die Link-Local-Adresse nur Kommunikation im lokalen Netzwerksegment erlaubt, wird empfohlen, diese nur für den Fall der Erstparametrierung von Geräten zu verwenden und in diesem Zuge möglichst immer durch eine routenfähige IP-Adresse zu ersetzen.

5.2 Automatisierbare IP-Adressvergabe mittels OPC UA Discovery

Mittels des lokalen Discovery-Dienstes kann eine OPC UA-basierte IP-Adressvergabe realisiert werden, welche im Gegensatz zu DHCP auch die Möglichkeit einer hostinitiierten Gerätekonfiguration bietet. Hierdurch entfällt die Notwendigkeit des Einsatzes herstellerspezifischer Tools und Protokolle zur Adresskonfiguration.

Im Folgenden wird nicht mehr zwischen dem OPC UA Client und Server unterschieden. Entsprechend der in Abschnitt 2 definierten Terminologie werden stattdessen die Begriffe Device und Engineering Host verwendet. Sowohl Device als auch Engineering Host beinhalten jeweils einen LDS-ME. Zur Realisierung der IP-Adressvergabe wird der Einsatz eines neuen OPC UA SRV Records vorgeschlagen, welcher beispielsweise mit `_opcua-devicemanagement` bezeichnet werden könnte. Der jeweilige Operationstyp (Adressanforderung oder Adressvergabe) kann in einem TXT Resource Record definiert werden.

Wie in Anforderungen beschrieben wurde (siehe Kapitel 1), muss die IP-Adressvergabe mit OPC UA hauptsächlich zwei Anwendungsfälle unterstützen: (i) Device-initiierte Adressvergabe und (ii) vom Engineering Host-initiierte Adressvergabe. Der Ansatz für beide Fälle wird im Folgenden beschrieben.

5.2.1 Device-initiierte Adressvergabe (vergleichbar mit DHCP)

In diesem Anwendungsfall kann die IP-Adressvergabe in Analogie zum lokalen OPC UA Discovery-Prozess realisiert werden. Der Ablauf wird in Abbildung 4 dargestellt. Die Anforderung einer IP-Adresse (IP request) wird hier mittels einer mDNS Multicast Nachricht vom Device an den Engineering Host gesendet. Die Anforderung bezieht sich in ihrem SRV Record-Feld auf den Dienst `_opcua-devicemanagement`. Im TXT Record wird angegeben, dass es sich um eine IP-Anforderung handelt.

Der Host antwortet nach Erhalt der Anforderung mit einer Nachricht, die per Unicast direkt an das Device gesendet wird. Die Nachricht enthält den gleichen SRV Record, im TXT Record steht die dem Device zugeteilte IP-Adresse.

Weitere Operationen wie das DHCP FORCERENEW-Kommando oder die in [5] vorgeschlagenen DHCP-Erweiterungen sind hier nicht notwendig, da der Engineering Host in der Lage ist, selbstständig Verbindungen zu den Devices aufzubauen.

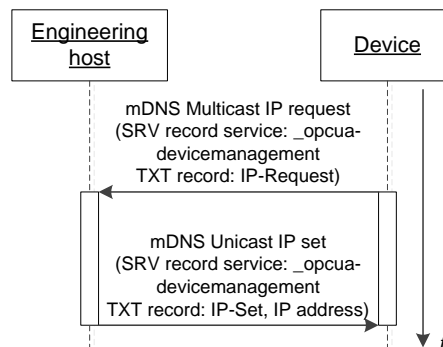


Abbildung 4: Device-initiierte Adressvergabe mittels OPC UA

5.2.2 Host-initiierte Adressvergabe (vergleichbar mit DCP)

In diesem Anwendungsfall muss der Engineering Host in der Lage sein, das Netzwerk nach vorhandenen Geräten zu erkunden (vergleichbar mit dem Identify Request von DCP) und diesen Adressen zuzuweisen. Die gesamte Adressvergabe ist in Abbildung 5 dargestellt. Der Erkundungsprozess kann auch hier mittels mDNS Multicast Requests realisiert werden, welche den SRV Record `_opcua-devicemanagement` verwenden. Im TXT Record kann zusätzlich angegeben werden, ob der Host alle Devices erkunden will oder nur bestimmte Devices, welche durch geeignete Selektoren, wie beispielsweise Vendor-ID oder Product-ID, ausgewählt werden. Die entsprechenden Devices senden ihre Antworten (Identify Response) an den Host. In einem dritten Schritt kann der Engineering Host schließlich allen Geräten, die geantwortet haben, mittels der bereits im ersten Anwendungsfall beschriebenen IP set-Nachricht eine Adresse zuweisen.

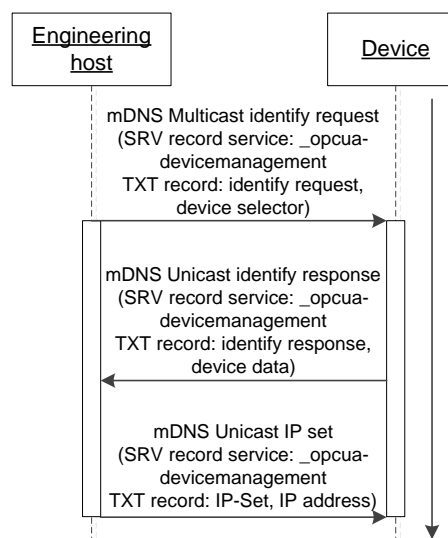


Abbildung 5: Host-initiierte Adressvergabe mittels OPC UA

5.3 IP-Parameterobjekte für OPC UA

Ein wesentlicher Aspekt für den vorliegenden Ansatz ist die Definition von einheitlich modellierten Konfigurationsobjekten für IP-Parameter. Nur dann können Tools herstellerübergreifend diese Parameter einfach adressieren. Deshalb sollte die Companion Specification „Device Integration“ überarbeitet und um entsprechende Objekte erweitert werden.

6 Fazit

In dieser Arbeit wird das Problem der initialen Adressvergabe für IP basierte Kommunikationssysteme adressiert. Bestehende Lösungen aus dem IT Bereich erfüllen nicht oder nur teilweise die bestehenden Anforderungen der Automation. Daher existieren in diesem Bereich eine Vielzahl an herstellerabhängigen Mechanismen und Protokollen, die nicht interoperabel sind. In einem System von heterogenen Automatisierungsnetzen bedeutet dies, dass eine große Anzahl verschiedener Tools eingesetzt werden muss.

Daher werden zwei Lösungsansätze in dieser Arbeit vorgestellt, die die identifizierten Anforderungen abdecken. Die IP-Autokonfiguration analog zu den in IPv6 integrierten Mechanismen und eine automatisierte IP-Adressvergabe, die in das OPC UA Discovery eingebettet ist. Beide Ansätze können problemlos in eine standardisierte Lösung überführt bzw. integriert werden.

Die technische Lösung des angesprochenen Problems ist relativ einfach und gut umsetzbar. Am Ende stellt sich daher die Frage, wie eine standardisierte Lösung politisch herbeigeführt werden kann? Da in der Vergangenheit auf Herstellerseite oft eine wesentliche Interessenlage in der Definition eines proprietären Marktsegments bestanden hat, sind die meist durch einzelne Hersteller getriebenen inkompatiblen Feldbussuiten entstanden.

Im Zuge der Industrie 4.0-Initiativen ist die klare Tendenz erkennbar, diesen Fehler nicht mehr wiederholen zu wollen. Das gibt beispielsweise herstellerunabhängigen Organisationen wie der OPC Foundation den notwendigen ideellen Rückhalt, um den OPC UA-Ansatz in dieser Richtung vorantreiben zu können. Grundvoraussetzung ist aber immer das menschliche Engagement, d.h. dass sich genügend Teilnehmer in der Automatisierungsbranche finden, die einen derartigen Ansatz voranzutreiben bereit sind. Hiermit möchten wir um Mitstreiter werben.

Danksagung und Hinweise

Diese Arbeit wurde mit Mitteln des Bundesministeriums für Bildung und Forschung (BMBF) im Rahmen des Spitzenclusters „Intelligente Technische Systeme OstWestfalenLippe (it's OWL)“ gefördert. Sie basiert auf dem Beitrag [2] und ergänzt diesem um zusätzliche Aspekte.

Literaturverzeichnis

- [1] L. Dürkop, H. Trsek, J. Jasperneite, L. Wisniewski. “Towards Autoconfiguration of Industrial Automation Systems: A Case Study using Profinet IO,” Emerging Technologies & Factory Automation (ETFA), 2012 IEEE 17th Conference on, Sep. 2012.
- [2] M. Rentschler, H. Trsek, L. Dürkop. “OPC UA extension for IP Auto-Configuration in Cyber-Physical Systems,” Industrial Informatics (INDIN), 2016 IEEE 17th Conference on, July. 2016.
- [3] M. Rentschler. “Faulty Device Replacement” Industrial Informatics (INDIN), 2012 IEEE 17th Conference on, July 2012.
- [4] Dürkop, L.; Imtiaz, J.; Trsek, H.; Wisniewski, L.; Jasperneite, J., “Using OPC UA for the Auto configuration of Real-time Ethernet Systems.” 11th IEEE International Conference on Industrial Informatics (INDIN) 2013.
- [5] <https://datatracker.ietf.org/doc/draft-rentschler-dhc-discovery/>