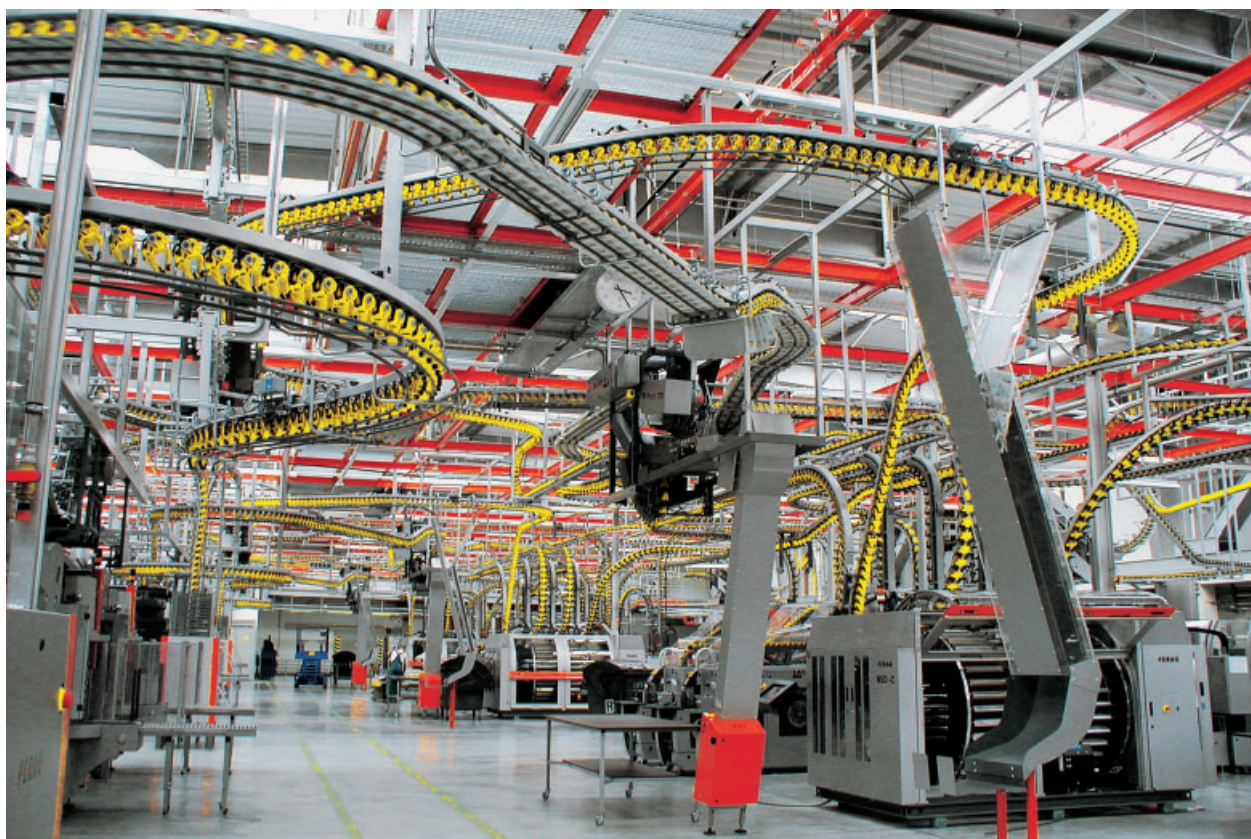


EtherCAT durchgängig in allen Ebenen

## EtherCAT für die Fabrikvernetzung

Auf der Feldebene wird das bereits weit verbreitete und bekannte EtherCAT Device Protocol, häufig auch einfach als EtherCAT-Protokoll bezeichnet, für die I/O-Kommunikation innerhalb einer Maschine oder eines Maschinenteils genutzt. Besondere Eigenschaften sind unter anderen die hochgenaue Deterministik mit sehr geringen Zykluszeiten (bis  $< 100 \mu\text{s}$ ), die präzise Synchronisierung für Antriebs- und Messtechnikwendungen sowie die niedrigen Anschaltkosten zur Nutzung der Technologie bis in die I/O-Ebene. Die Prozessleitebene erfordert zum Betrieb einer Anlage oder einer Fabrik weitere Kommunikationsmöglichkeiten. Hierfür bietet EtherCAT mit dem EtherCAT Automation Protocol eine Lösung.



**Printing on the Fly:** Für ein einwandfreies Ergebnis müssen die Positionierung der Zeitung und der Druck der Adresse auf das vorgesehene Feld des Außenbogens exakt synchronisiert sein.

Die Geräte der Feldebene werden von einer übergeordneten Steuerung kontrolliert. Große Anlagen, wie in der Automobilindustrie, werden aus mehreren Maschinen aufgebaut. Hierbei müssen die Steuerungen der einzelnen Anlagenteile Daten austauschen. Unter Umständen kann sogar der Datenaustausch zwischen I/O-Geräten aus verschiedenen Anlagenteilen sinnvoll sein.

Für ein Kommunikationsprotokoll, das im Bereich der Produktionsleitebene Anwendung finden soll, ergeben sich folgende Anforderungen:

- | Datenaustausch zwischen EtherCAT-Mastern (Master-Master-Kommunikation) = Datenaustausch auf Steuerungsebene
- | Datenaustausch zwischen EtherCAT-Master und Visualisierung
- | Durchgriff von überlagerten Steuerungen auf Teilnehmer in unterlagerten EtherCAT-Segmenten (Routing)
- | Datenaustausch zwischen EtherCAT-Mastern und anderen Teilnehmern wie Konfigurations-Tools

Zusätzliche Anforderungen:

- | Standard-Ethernet-Kommunikationsschnittstelle
- | keine strengen Anforderungen an Zykluszeit und Synchronität
- | zyklische Kommunikation im Millisekundenbereich
- | Einsatz von Standard-Infrastrukturelementen wie Switchen zur Vernetzung der Teilnehmer

Diese Anforderungen werden vom EtherCAT Automation Protocol (EAP) erfüllt, welches damit die vertikale Integration von EtherCAT im Gesamtsystem stärkt.

### EtherCAT-Protokolltypen

Das EtherCAT-Protokoll kann über Ethernet (EtherType 0x88A4), über UDP (User Datagram Protocol, UDP-Port 0x88A4) oder per TCP (Transmission Control Protocol, TCP-Port 0x88A4) übertragen werden. Das Übertragungsmedium ist dabei nicht relevant: Fast-Ethernet- oder Gigabit-Ethernet-Verbindungen auf Kupfer

oder Glasfaser sind ebenso möglich wie die Nutzung des Protokolls über eine Wireless-Verbindung. Somit können auch Anlagenteile integriert werden, die nicht über eine feste Leitung verbunden werden können (z.B. Flurförderfahrzeuge, Regalbediensysteme).

In den Nutzdaten dieser Telegramme schließt sich der EtherCAT-Frame an (Abbildung 1).

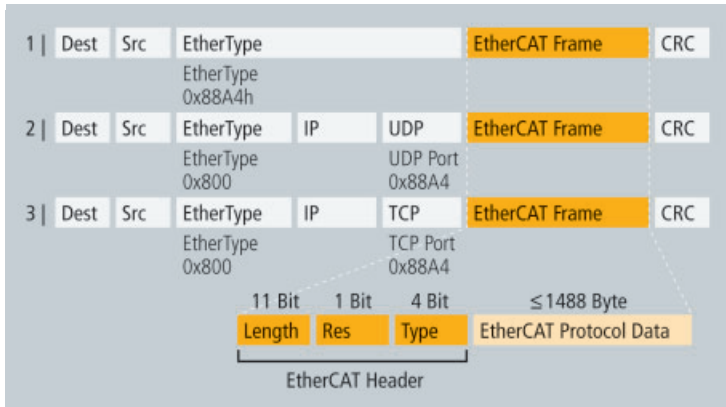


Abb. 1: EtherCAT-Protokollübertragung

Der Header des EtherCAT-Frames spezifiziert den EtherCAT-Protokolltyp.

EtherCAT Device Protocol	Typ 1:	EtherCAT-Datagramm(e)
EtherCAT Automation Protocol	Typ 4:	Prozessdatenkommunikation
	Typ 5:	Mailbox-Kommunikation

EtherCAT-Slaves, die an einen EtherCAT-Master angeschlossen sind, verwenden immer das EtherCAT Device Protocol.

Für die Umsetzung des Protokolls wird in den EtherCAT-Slaves ein EtherCAT-Kommunikations-Chip, der EtherCAT Slave Controller (ESC), verwendet. Dieser wertet ausschließlich Telegramme vom Typ 1 aus. Sowohl zyklische als auch azyklische Daten können in einem EtherCAT-Frame mit diesem Typ übertragen werden.

Für das EtherCAT Automation Protocol werden die Typen 4 und 5 verwendet. Folgen Prozessdaten, wird der Typ 4 verwendet, bei Mailbox-Daten steht der Typ 5 im EtherCAT-Frame-Header. Die Mailbox-Protokolle (CoE, SoE und FoE), wie sie bereits mit dem EtherCAT Device Protocol genutzt werden, können ebenso mit dem EtherCAT Automation Protocol verwendet werden.

### EAP-Nachrichten-Routing

Damit Mastergeräte untereinander Daten austauschen können oder z. B. ein Konfigurations-Tool Parameter in einem Antrieb im unterlagerten EtherCAT-Segment einstellen kann, muss ein Nachrichten-Routing vom Konfigurations-Tool über den EtherCAT-Master bis zum -Slave möglich sein.

Hierfür wird das EtherCAT Mailbox Protocol AoE (Automation Device Protocol over EtherCAT) verwendet. Dieses Protokoll hat den Vorteil, dass es routingfähig ist und somit durch mehrere Ebenen durchgereicht werden kann, um unterlagerte Objektverzeichnisse zu erreichen.

Jeder Ethernet-Port im Master wird als AoE-Gerät implementiert und erhält eine eigene AoE-NetID. Im Beispiel (Abbildung 2) erhalten Port 1 und Port 2 je eine AoE-NetID. Das Routing von einem Port zum anderen wird von einem AoE-Router im Master übernommen, welcher ebenfalls eine AoE-NetID zugewiesen bekommt.

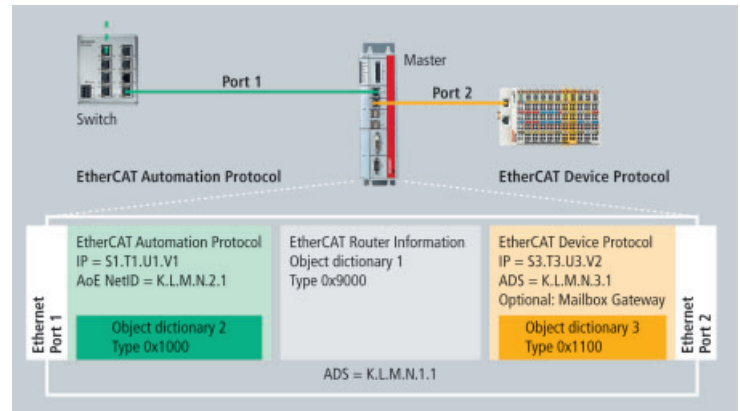


Abb. 2: Identifikation von EAP-Teilnehmern

Über die AoE-NetID lässt sich auf jeden Port bzw. auf jedes AoE-Gerät und die mit diesem verbundenen Informationen zugreifen. Die Informationen, die dem Netzwerk zur Verfügung gestellt werden, sind in Objektverzeichnissen strukturiert. Deren genauer Inhalt unterscheidet sich je nach Aufgabe des AoE-Geräts/-Ports.

## Strukturierung der Gerätedaten

### Objektverzeichnis

Ein Objektverzeichnis ist eine Liste mit Variablen und Parametern. Jeder Eintrag wird über einen eigenen Index und ggf. einen Subindex adressiert. Der gesamte Indexraum ist in verschiedene Bereiche unterteilt, so wie in der Tabelle beschrieben.

Aufteilung des Index-Raums			
Index-Bereich	Verwendung		
0x0000 – 0x0FFF	Datentyp-Beschreibung		
0x1000 – 0x1FFF	Standard-MDP-Objekte (Device-Name, Hardware/Softwareversion, Identity, Zusammenstellung der Prozessdaten)		
0x2000 – 0x5FFF	herstellerspezifischer Indexbereich (z. B. Geräteparameter) profilspezifischer Bereich (hier Profil 5001 – MDP-Profil)		
	<b>Subprofil 1000</b>	<b>Subprofil 1100</b>	<b>Subprofil 9000</b>
0x6000 – 0x6FFF	Eingangsvariablen	--	--
0x7000 – 0x7FFF	Ausgangsvariablen	--	--
0x8000 – 0x8FFF	Process-Data-Konfiguration	Information über alle angeschlossenen Slaves	AoE-Geräte-Informationen
.....	.....		
0xD000 – 0xDFFF	Ausgangs-Process-Data	--	--
0xE000 – 0xEFFF	Eingangs-Process-Data	--	--
0xF000 – 0xFFFF	gerätespezifische Daten (z. B. Index-Abstand, AoE-NetID)		

Durch die Unterteilung des Indexraums in festgelegte Bereiche wird die Strukturierung der Daten übersichtlich. Außerdem ist dies die Grundlage für die Verwendung von Algorithmen, mit denen die Zusammenstellung der Prozessdaten (PDO-Konfiguration und PDO-Zuordnung) organisiert wird.

**Profilspezifischer Bereich für das EAP**

Der profilspezifische Bereich ab Index 0x6000 wird durch sogenannte Geräteprofile festgelegt. Für viele Geräteklassen (Antriebe, I/O-Geräte) wurden entsprechende Geräteprofile spezifiziert. Für das EAP wird das Modular Device Profile (MDP, Profilnummer 5001) verwendet. Dieses Profil legt die Verwendung des Indexraums entsprechend der Tabelle fest. Spezielle Geräteklassen werden dabei durch Subprofile in dieses Raster eingeordnet (z.B. für Gateway-Geräte). Das EtherCAT Automation Protocol besitzt eine eigene Subprofilnummer (1000) ebenso wie das EtherCAT Device Protocol (1100) für die EtherCAT-Master-Funktion zum unterlagerten EtherCAT-Segment. Der AoE-Router besitzt die Subprofilnummer 9000.

Auf einem Master, der sowohl das EtherCAT Automation Protocol als auch das EtherCAT Device Protocol unterstützt, haben die Ports zum jeweiligen Netzwerk ein eigenes Objektverzeichnis mit der entsprechenden Subprofilnummer (vergleiche auch Abbildung 2).

**1. Subprofil 1000 – EtherCAT Automation Protocol**

Das Objektverzeichnis für das Subprofil 1000 wird für die Konfiguration der Kommunikationsbeziehung zwischen zwei EAP-Geräten verwendet. Es beschreibt die Prozessdaten, wie sie für das EAP verwendet werden.

**2. Subprofil 1100 – EtherCAT Device Protocol**

Das Objektverzeichnis des EtherCAT-Segments listet alle angeschlossenen Slaves auf. Ein Slave wird dabei im Objektverzeichnis als Modul beschrieben.

**3. Subprofil 9000 – EtherCAT Router Information**

Im Router-Objektverzeichnis stehen alle verfügbaren Schnittstellen des Geräts und deren AoE-NetIDs.

**EAP – zyklischer Datenaustausch**

Der Prozessdatenaustausch im EAP kann nach dem „Pushed“- oder dem „Polled“-Prinzip erfolgen. Im „Pushed“-Betrieb sendet jeder Kommunikationsteilnehmer seine Daten zyklisch oder in einem Vielfachen des eigenen Zyklus. Im Empfänger kann konfiguriert werden, von welchem Sender welche Daten empfangen werden sollen. Die Konfiguration zwischen Sender- und Empfängerdaten erfolgt dabei über das Objektverzeichnis nach dem Subprofil 1000.

Im „Polled“-Betrieb werden die Daten von den Teilnehmern abgefragt. Hierzu schickt ein Teilnehmer, häufig der übergeordnete Leitreehner, ein Telegramm zu den Teilnehmern, woraufhin diese mit einem eigenen Telegramm antworten. Hierdurch ist auch eine Synchronisierung der Teilnehmer möglich.

**Aufbau der Prozessdaten**

Der Inhalt eines Process-Data-Telegramms wird wie die Prozessdaten eines EtherCAT-Slaves beschrieben. Ein Telegramm entspricht einer SyncManager-Area, so dass der Aufbau der Prozessdaten eines Telegramms über die PDO-Zuordnung und PDO-Konfiguration festgelegt wird (Abb. 3).

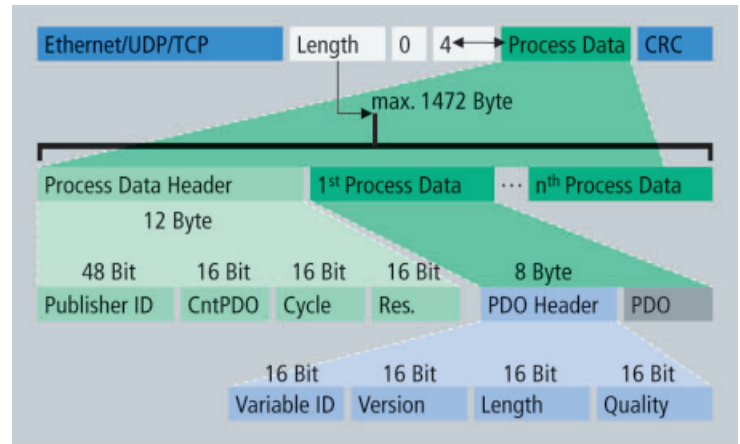


Abb. 3: Aufbau des Prozessdatentelegramms

Die Telegrammsegmente haben folgende Bedeutung:

Header	Header-Info	Beschreibung
<b>Process Data Header</b>	Publisher ID	AoE-NetID des Publishers (0xD000:06+4*n)
	CntPDO	PDO-Anzahl im Frame
	Cycle	inkrementiert mit jedem Zyklus
<b>PDO Header</b>	Variable ID	ID jeder PDO (0xD000:SI3+4*n)
	Version	Version dieser PDO (0xD000:SI4+4*n)
	Length	Länge der PDO
	Quality	Alter der Daten (in 100 µs)

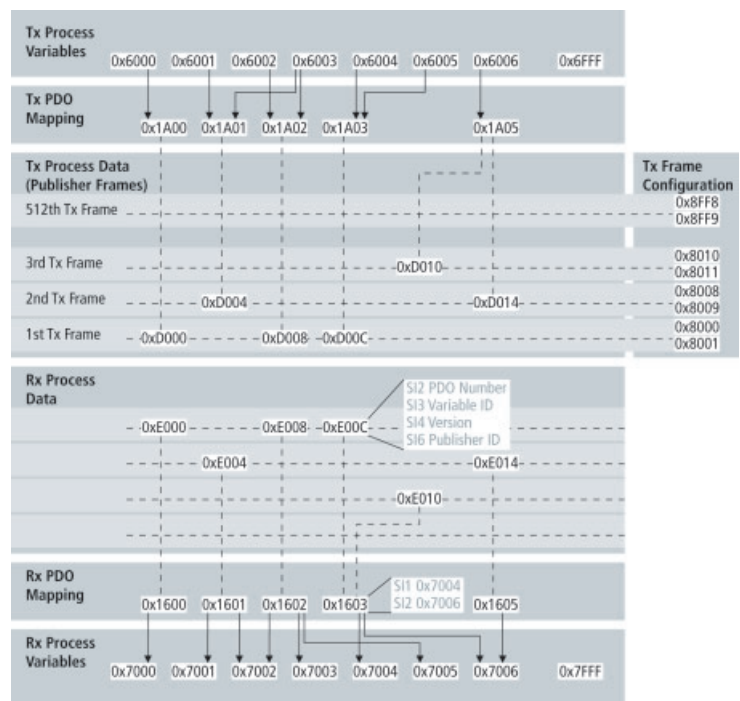


Abb. 4: Beispiel für den Aufbau der Prozessdatenkonfiguration. Der obere Teil zeigt den Aufbau der Tx Process Data und der Telegrammkonfiguration, der untere Teil die empfangenen Rx Process Data mit den zugeordneten Rx-Variablen.



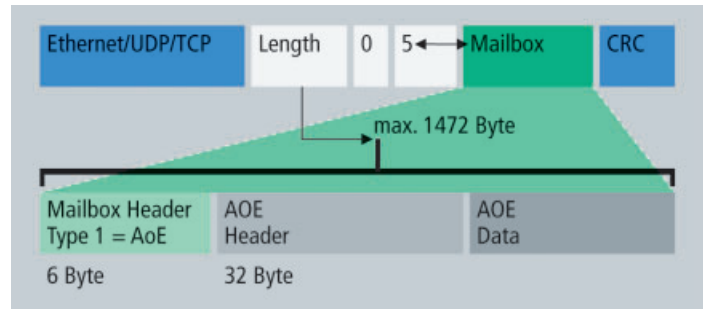


Abb. 5: Aufbau des AoE-Mailbox-Telegramms

Ein Gerät definiert seine Ausgangsvariablen (Tx *Process Variables*) im Indexbereich ab 0x6000. Für jede Variable wird ein Index verwendet (*Variable 1* → 0x6000, *Variable 2* → 0x6001 etc.). Mit dem tatsächlichen Wert der Variable wird der Name, die Länge und der Variablentyp, z. B. Prozessdatum oder Diagnosedatum, verbunden. Im *Process-Data*-Telegramm wird nur der Variablenwert selbst übertragen.

Die *Process Variables* können dann beliebig mit Hilfe der Tx-Mapping-PDOs ab 0x1A00 gruppiert werden. Dabei kann eine *Variable* auch in mehrere *PDOs* übertragen werden.

Die über das Mapping beschriebenen Strukturen von Variablen werden mit einer *Variable ID*, *Version (Version)* und *Alter (Quality)* kombiniert. Dies geschieht im Tx-*Process-Data*-Bereich ab 0xD000.

Die *Process Data* müssen nun noch für die Übertragung einem Ethernet-Frame zugeordnet werden. Die Zuordnung und Reihenfolge jedes *Process Data* wird über die Assign-Objekte im Bereich ab 0x8000 festgelegt.

Auf der Empfängerseite wird der Absender des empfangenen Frames durch die Publisher-ID im Frame-Header bestimmt. Mit dieser und der *Variable ID* und *Version* aus dem *PDO*-Header kann der Empfänger den *Process-Data*-Index (ab 0xE000) finden, in dem der Verweis auf seine interne Zielvariable steht. Über das gefundene Mapping-PDO (ab 0x1600) wird also der Index der Zielvariablen bestimmt, in dem der empfangene Variablenwert als Eingangsvariable gespeichert wird.

Die Konfiguration der Prozessdatenstruktur auf der Empfängerseite ist, wie auf der Absenderseite, beliebig. Welcher Index einer Rx *Process Variable* zugeordnet wird, ist unabhängig von dem Index der Tx *Process Variable*.

Da die Ausgangs- und Eingangsvariablen vorab konfiguriert werden, ist ein zusätzlicher Verbindungsaufbau vor Kommunikationsbeginn nicht mehr notwendig.

**EAP – azyklischer Datenaustausch**

Zur Konfiguration der Ausgangs- und Eingangsvariablen und um auf die Objektverzeichnisse des Masters oder der EtherCAT-Slaves zuzugreifen, wird ein azyklischer Zugriff auf diese ermöglicht. Hierzu wird das AoE-Protokoll (ADS over EtherCAT) verwendet. Damit können mehrere Objektverzeichnisse adressiert werden.

Als Transportprotokoll wird ein Telegramm vom Typ 5 (Mailbox-Kommunikation) verwendet. Der Aufbau des Frames ist dabei identisch mit der Mailbox-Kommunikation innerhalb eines EtherCAT-Segments (Abb. 5).

Auf das AoE-Protokoll können wiederum die Mailbox-Protokolle CoE, SoE und FoE abgebildet werden. Dadurch kann zum Beispiel ein Konfigurations-Tool an den Master angeschlossen werden, um auf einen Antrieb im EtherCAT-Netzwerk für Konfigurationszwecke zuzugreifen.

**Beispiel einer Anlagenautomation mit EAP**

Die Fertigung von Solarmodulen integriert eine Vielzahl von Prozessschritten, für welche Markierungs- und Identifikationssysteme, messtechnische Einheiten und spezielle Handlingsmodule eingesetzt werden.

Die verwendete Förderanlage ist in bis zu 14 Prozessinseln aufgeteilt, wobei jedes Segment mit einem Steuerungs- und einem Bedienrechner ausgestattet ist. Hinzu kommen Bedienpanel, die bei Bedarf an jedem Punkt der Fertigungslinie anschließbar sind.

Der Datenaustausch ist über das EAP realisiert. Jede Station tauscht bidirektional mit der jeweils vorherigen und der nachfolgenden Station Status- und Kontrollinformationen aus: 600 Byte in jede Richtung mit einer Zykluszeit von zehn Millisekunden. Hinzu kommt die Kommunikation zum Kontrollrechner, der zusätzlich mit jeder Station bis zu 1 kByte Daten bidirektional austauscht.

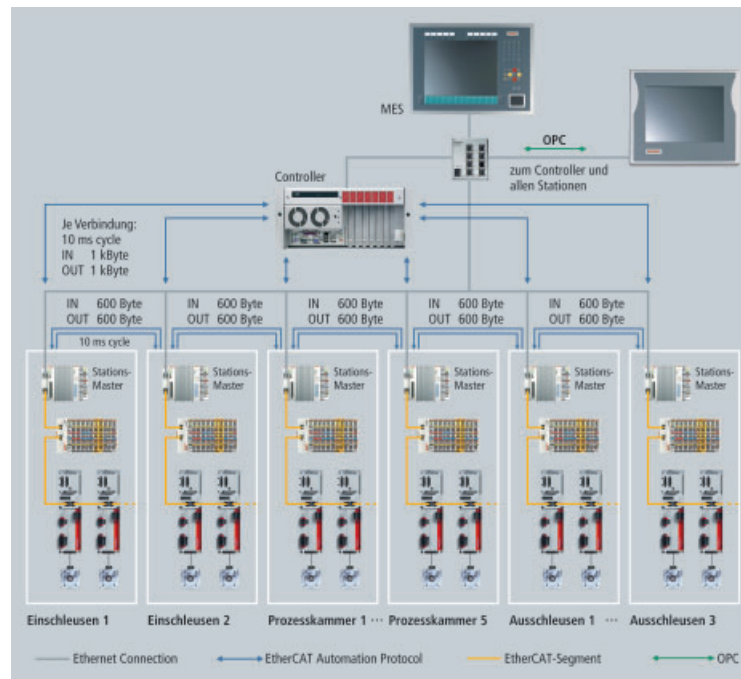


Abb. 6: Kommunikationsarchitektur einer Produktionsanlage für Solarmodule