

Beschreibung
EtherNet/IP
Prozessschnittstelle

Inhaltsverzeichnis

1.	Vorbemerkung	3
2.	Allgemeines	3
3.	EtherNet/IP Einstellungen.....	3
3.1	Kontrollbytes	4
3.2	Segmentierung	4
3.2.1	Segmentierungsbeispiel 1	5
3.2.2	Segmentierungsbeispiel 2	5
4.	Datenaustausch über Ethernet/IP.....	5

1. Vorbemerkung

Dieses Dokument dient zur Erläuterung einer EtherNet/IP-Kommunikation zwischen einem ifm Vision-Sensor mit EtherNet/IP Schnittstelle und einer EtherNet/IP fähigen SPS.

2. Allgemeines

Das Ethernet Industrial Protocol (EtherNet/IP) ist ein offener Standard für industrielle Netzwerke. EtherNet/IP dient zur Übertragung zyklischer E/A-Daten sowie azyklischer Parameterdaten. EtherNet/IP bildet damit eine breite Basis für effektive Datenkommunikation in der Industrie.

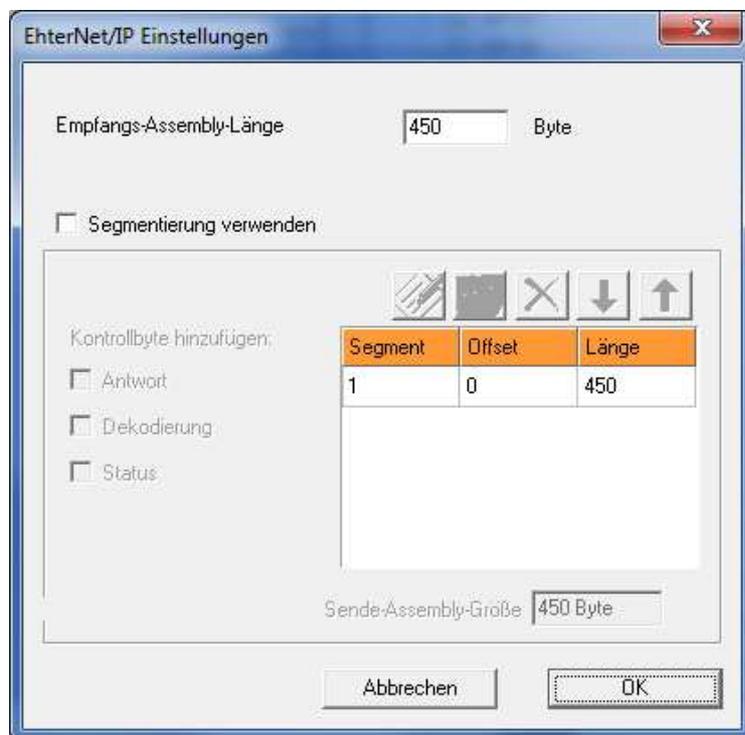
EtherNet/IP erweitert Ethernet um ein modernes Industrieprotokoll (CIP, Common Industrial Protocol) als Applikationsschicht für Automatisierungsanwendungen. Damit ist Ethernet für den Bereich der industriellen Steuerungstechnik bestens qualifiziert.

3. EtherNet/IP Einstellungen

Der ifm Objekterkennungssensor unterstützt als EtherNet/IP Adapter Device die Kommunikation mit einem als EtherNet/IP Scanner konfigurierten Gerät. In der Regel ist dies der Prozessrechner (z.B. eine SPS).

Die Kommunikation erfolgt mit Hilfe von 2 EtherNet/IP Assemblies, eines zum Datentransport von der Steuerung zum Sensor („Output Assembly Instance“, ID Adresse 100 / 0x64) und eines zum Datentransport vom Sensor zur Steuerung („Input Assembly Instance“, ID Adresse 101 / 0x65).

Die Längen der Assemblies müssen im Sensor und in der Steuerung gleich eingestellt werden.



- „Empfangs-Assembly-Länge“ definiert die Länge der „Output Assembly Instance“ (ID 100)
- „Segmentierung verwenden“ aktiviert die Definition der verschiedenen „Input Assembly Instance“ (ID 101) Parameter.

3.1 Aufbau des Assemblies bei deaktivierter Segmentierung

i	Die „Input Assembly Instance“ (101) ist 450 Byte lang und ist in 3 Bereiche segmentiert:			
		Offset	Länge	Inhalt
	Bereich 1	0	215 Byte	Antwort auf die eingehenden Nachrichten
	Bereich 2	215	215 Byte	Byte Ergebnis der Code oder Bildauswertungen
Bereich 3	430	20 Byte	Feste Geräte und Ergebnis Informationen	

Das letzte Byte jedes Segments dient als Kontrollbyte, das während der Bearbeitung durch den Objekterkennungssensor hoch gezählt wird. Bei gleichem Codeinhalt dienen diese Kontrollbytes zur Unterscheidung der Eingangsdaten.

Test	Offset	Länge	Ankreuzfeld Bezeichnung
Kontrollbyte Bereich 1	214	1 Byte	Antwort
Kontrollbyte Bereich 2	429	1 Byte	Auswertung
Kontrollbyte Bereich 3	449	1 Byte	Geräteinformationen

3.2 Segmentierung

In der "Output Assembly Instance" werden Nachrichten immer ab Adresse 0 geschrieben, hier lässt sich nur die Länge festlegen. Diese muss mindestens der Länge der längsten möglichen Nachricht entsprechen (max. 450 Byte).

Um Speicherplatz zu sparen, lässt sich die "Input Assembly Instance" segmentieren. Auf diese Weise werden nur die tatsächlich für die Applikation benötigten Daten übertragen. Für jedes Segment kann in der Segment-Auswahlliste ein „Offset“ und die benötigte „Länge“ ausgewählt werden.

Die vordefinierte „Input Assembly Instance“ Unterteilung lässt sich mit Hilfe der Segmentierungstabelle umkonfigurieren. Die Segmentierungstabelle bildet eine neue Reihenfolge der Bytes für die „Input Assembly Instance“ ab. Ein Segment ist definiert durch seinen Index, eine Anzahl von Bytes (Segment Länge) und eine Byte-Adresse aus der vordefinierten „Input Assembly Instance“ (Segment Offset).

Damit wird die „Input Assembly Instance“ neu unterteilt. Der Segment-Index stellt die Zuordnungsreihenfolge fest. Die Anzahl an zugeordneten Bytes wird durch die Segment-Länge definiert, und der Segment-Offset zeigt auf die Adresse aus der vordefinierten „Input Assembly Instance“, von der die Bytes extrahiert werden.

Das letzte Byte jedes Segmentes kann als Kontrollbyte aktiviert werden. Dieses wird dann während der Bearbeitung durch den Objekterkennungssensor hoch gezählt. Bei gleichem Codeinhalt dienen diese Kontrollbytes zur Unterscheidung der Eingangsdaten. Durch einen Klick auf das entsprechende Feld kann das Kontrollbyte aktiviert bzw. deaktiviert werden.

Vorsicht! Für eine korrekte Segmentierung müssen alle Bytes aus der „Input Assembly Instance“ eindeutig abgebildet werden.

3.2.1 Segmentierungsbeispiel 1

Segmentierungstabelle

Index	Offset	Länge
1	0	450 Byte

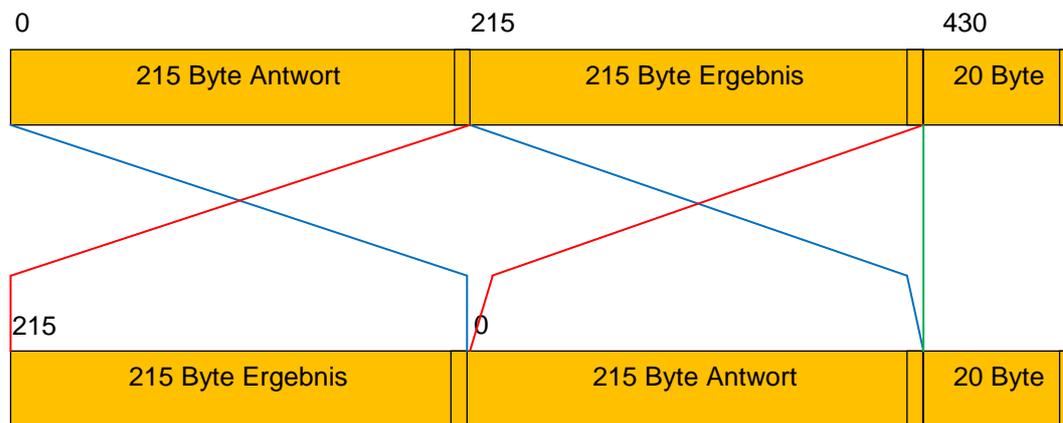
Erläuterung:

Default Segmentierungstabelle. Nimmt 450 Bytes (alle!) aus der vordefinierte „Input Assembly Instance“ und platziert diese auf die Adresse 0. Diese Segmentierungstabelle hat deshalb keine tatsächliche Wirkung auf die „Input Assembly Instance“.

3.2.2 Segmentierungsbeispiel 2

Segmentierungstabelle

Index	Offset	Länge
1	215	215 Byte
2	0	215 Byte
3	430	20 Byte



Erläuterung:

Bereiche 1 und 2 aus der vordefinierten „Input Assembly Instance“ werden getauscht: Erst werden 215 Bytes ab Byte-Adresse 215 aus der vordefinierte „Input Assembly Instance“ neu platziert. Danach 215 Bytes ab Byte-Adresse 0 und dann 20 Bytes ab Byte-Adresse 430 aus der vordefinierten „Input Assembly Instance“.

4. Datenaustausch über EtherNet/IP

Der Datenaustausch zwischen einem EtherNet/IP fähigen Sensor und einer SPS erfolgt zyklisch. D.h. die im Sensor gespeicherten Daten im Output-Assembly Bereich (ID 100) werden in jedem Zyklus von der angeschlossenen SPS abgeholt und in dem in der SPS definierten Datenbereich gespeichert.

Ändern sich die Daten im Sensor, so werden diese im nächsten Zyklus der SPS in den definierten Datenbereich übernommen und stehen so lange zur Verfügung, bis der Sensor seinen Output-Assembly Bereich überschreibt.