

# FABRIKAUTOMATION



## BEDIENUNGSANLEITUNG

PROFIBUS VB<sub>3</sub>X



CE

Es gelten die Allgemeinen Lieferbedingungen für Erzeugnisse und Leistungen der Elektroindustrie,  
herausgegeben vom Zentralverband Elektrotechnik und Elektroindustrie (ZVEI) e.V.  
in ihrer neuesten Fassung sowie die Ergänzungsklausel: "Erweiterter Eigentumsvorbehalt"

Wir von PEPPERL+FUCHS/VISOLUX fühlen uns verpflichtet, einen Beitrag für die Zukunft zu leisten,  
deshalb ist diese Druckschrift auf chlorfrei gebleichtem Papier gedruckt.

<b>1</b>	<b>Überblick</b> .....	<b>5</b>
1.1	<b>Beschreibung</b> .....	<b>5</b>
1.2	<b>Profibus-DP</b> .....	<b>5</b>
<b>2</b>	<b>Buskommunikation</b> .....	<b>6</b>
2.1	<b>Datenaustausch</b> .....	<b>6</b>
2.2	<b>Konfigurationseinstellungen</b> .....	<b>7</b>
2.2.1	Buskommunikationsparameter .....	8
2.2.2	Anleitung zur Auswahl von Profibus-Adressen über die Tastatur: .....	8
2.3	<b>Flusssteuerungsmodus (FSM)</b> .....	<b>9</b>
2.3.1	Flusssteuerungstreiber .....	11
2.3.2	FSM mit DAD-Treiber .....	12
	Control-Feld .....	13
	SAP-Feld .....	14
	Length-Feld .....	14
	Datenübertragung von VB3X zur SPS .....	14
	Datenübertragung von der SPS zu VB3X .....	17
	Resynchronisation .....	18
	Fragmentierung und Wiederaussetzen .....	20
	SAP-Dienste .....	22
	DAD-interne Warteschlangen .....	23
2.3.3	FSM mit DPD-Treiber .....	23
2.4	<b>Datenkonsistenz</b> .....	<b>24</b>
2.4.1	Datenkonsistenz mit DAD-Treiber .....	25
2.4.2	Datenkonsistenz mit DPD-Treiber .....	27
<b>3</b>	<b>Digitale I/O-Konditionierung</b> .....	<b>27</b>
3.1	<b>Digitale Eingang-Konditionierung</b> .....	<b>29</b>
3.1.1	Phase Echo .....	30
3.2	<b>Digitale Ausgang-Konditionierung</b> .....	<b>31</b>
3.2.1	Phase Trigger .....	32
3.2.2	Lesephase über Profibus .....	32
<b>4</b>	<b>Netzwerkkonfiguration</b> .....	<b>33</b>
4.1	<b>GSD-Datei</b> .....	<b>33</b>
4.2	<b>GSD-Installation</b> .....	<b>34</b>
4.3	<b>Scannerprogrammierung über die GSD-Datei</b> .....	<b>36</b>
4.3.1	Projektmodule .....	36
	[M1] Operating Mode .....	38
	[M2] Start Input .....	38
	[M3] Stop Input .....	38

# PROFIBUS VB3X

## Inhaltsverzeichnis und allgemeine Informationen

[M4] OnLine Timeout Options .....	38
[M5] Automatic Mode Options .....	39
[M6] Continuous Mode Options .....	39
[M7] Packtrack Mode Options .....	39
[M8] Reading Parameters .....	39
[M10] Profibus Settings .....	40
[M11] Code Label Setting 1 ... [M15] Code Label Setting 5 .....	40
[M21] Restore Default .....	41
[M31] Oscillating Mirror Settings .....	41
[M32] Modul ScanLine Amplitude Settings .....	41
[M33] Modul D-Flash Settings .....	41

## 5 Notizen .....

### 43

### Allgemeine Informationen

Diese Betriebsanleitung enthält Informationen zur bestimmungsgemäßen Verwendung des Produkts. Sie muss von allen Personen, die dieses Produkt implementieren, einsetzen, pflegen, warten und kontrollieren, gelesen und beachtet werden. Dieses Produkt kann seine bestimmungsgemäßen Aufgaben nur erfüllen, wenn es entsprechend den Anweisungen von Pepperl+Fuchs/Visolux eingesetzt, gepflegt, gewartet und kontrolliert wird.

Die von Pepperl+Fuchs/Visolux für dieses Produkt übernommene Garantie erlischt, wenn es nicht entsprechend den Anweisungen von Pepperl+Fuchs/Visolux eingesetzt, gepflegt, gewartet und kontrolliert wird.

Veränderungen an den Geräten oder Bauteilen sowie der Einsatz von defekten oder unvollständigen Geräten oder Bauteilen sind unzulässig. Instandsetzungen an Geräten oder Bauteilen dürfen nur von Pepperl+Fuchs/Visolux durchgeführt werden.

Instandsetzungsarbeiten am Produkt, die nicht von Pepperl+Fuchs/Visolux durchgeführt wurden, unterliegen nicht dem Einfluss von Pepperl+Fuchs/Visolux. Unsere Haftung ist daher beschränkt auf Instandsetzungsarbeiten, die von Pepperl+Fuchs/Visolux durchgeführt werden.

Das Vorstehende ändert nicht die Angaben über Gewährleistungen und Haftung in den Verkaufs- und Lieferbedingungen von Pepperl+Fuchs.

### Verwendete Symbole

In dieser Anleitung werden Symbole verwendet, um auf wichtige Informationen zur Bedienung des VB3X und zum sicheren Arbeiten damit hinzuweisen. Die Bedeutung dieser Symbole ist:



Note

*Empfehlung für den Anwender*

*Bei Beachtung dieser Hinweise wird es Ihnen erleichtert, das System in Betrieb zu nehmen und mit dem VB3X zu arbeiten.*

Technische Änderungen vorbehalten.

Ausgabedatum: 06/15/2005

# 1 Überblick

## 1.1 Beschreibung

Profibus ist der am weitesten verbreitete FieldBus der Welt.

Profibus ist der am breitesten akzeptierte Netzwerkstandard. In Europa wird er fast universell eingesetzt und ist auch in Nordamerika, Südamerika und Teilen Afrikas und Asiens verbreitet. Profibus kann große Datenmengen bei hoher Geschwindigkeit handhaben und wird den Anforderungen der Mehrzahl von Automatisierungsanwendungen gerecht.

Profibus wurde 1989 unter der Leitung der deutschen Regierung in Zusammenarbeit mit Automatisierungsherstellern (Siemens) entwickelt. Heute wird er hauptsächlich in der Prozessführung sowie in großen Montage- und Materialverarbeitungsanlagen eingesetzt. Nur ein einziges Kabel ist in der Lage, Multi-Input-Sensorblocks, Druckventile, komplexe intelligente Geräte, kleinere Subnetzwerke, Bediener-schnittstellen und viele andere Geräte zu verbinden.

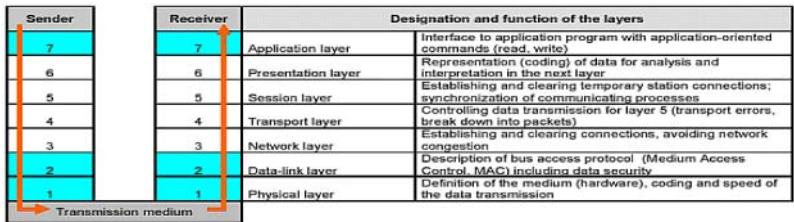


Abbildung 1.1 ISO/OSI-Referenzmodell

Das ISO/OSI-Referenzmodell beschreibt die Kommunikation zwischen den Stationen eines Kommunikationssystems: Wenn ein Kommunikationssystem bestimmte Funktionen nicht benötigt, erfüllen die entsprechenden Schichten keine Zweck und werden umgangen. Profibus verwendet die Schichten 1, 2 und 7.

## 1.2 Profibus-DP

Profibus ist grundsätzlich in zwei verschiedenen Versionen verfügbar:

- Profibus-DP (Dezentrale Peripherie)  
Mit Profibus-DP sind mehrere Master möglich. In diesem Fall wird jeder Slave einem Master zugewiesen. Das bedeutet, dass mehrere Master Eingaben von dem Gerät lesen können, aber nur ein Master Ausgaben an das Gerät schreiben kann.
- Profibus-FMS  
Ein Peer-to-Peer-Nachrichtenformat, das die Kommunikation der Master untereinander ermöglicht. Wie beim Profibus-DP sind bis zu 126 Teilnehmer verfügbar, die, falls gewünscht, alle Master sein können. FMS-Nachrichten verbrauchen mehr Kosten als DP-Nachrichten.
- Profibus-PA  
Das PA-Protokoll entspricht dem neuesten Profibus-DP mit der Ausnahme, dass Spannung und Strompegel reduziert werden, um den Anforderungen der Eigensicherheit (Klasse I Div. II) für die Verfahrenstechnik gerecht zu werden.

Pepperl+Fuchs unterstützt ausschließlich Profibus-DP, da diese Version speziell für die Fabrikautomation ausgelegt ist. Die Systemversion muss diesem Typ entsprechen.

Hauptmerkmale:

- Maximale Anzahl von Teilnehmern: 126
- Entfernung: 100 m bis 24 km (mit Repeatern und Lichtleiterübertragung)
- Baudrate 9600 bis 12 M bps
- Nachrichtenformate: Polling, Peer-to-Peer

Im Folgenden beziehen sich alle Angaben ausschließlich auf Profibus-DP

## 2 Buskommunikation

### 2.1 Datenaustausch

Der **Master**-Profibus ist meist eine SPS (Siemens S7 oder andere). Manchmal kann es sich auch einer computerbasiertes Gerät handeln.

Das Gerät VB3X-xxxx-P ist im Profibus-Netzwerk immer ein **Slave**.

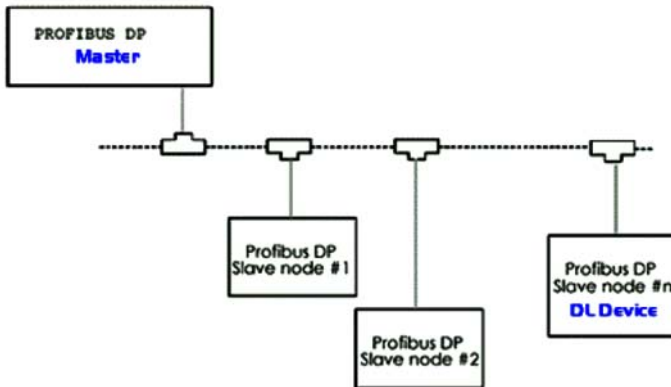


Abbildung 2.1 PROFIBUS-Netzwerktopologie

Grundsätzlich gibt es zwischen SLAVE und MASTER zwei gemeinsame Speicherbereiche (Austauschbereiche), so dass die Geräte miteinander Informationen austauschen können. Physisch befinden sich die Austauschbereiche auf der VB3X-Profibus-Karte.

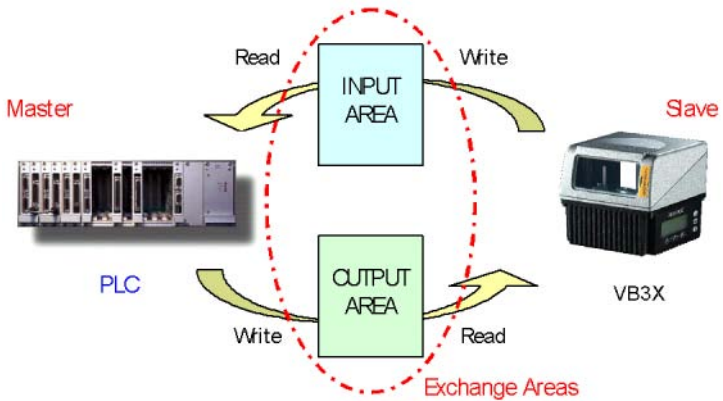


Abbildung 2.2 Datenaustausch

Eingangs- und Ausgangsbereiche beziehen sich immer auf den Master: Das bedeutet, dass der Scanner in den Eingangsspeicher und die SPS in den Ausgangsspeicher schreibt.

Die Größen der Austauschbereiche können durch die SPS über die GSD-Datei auf unterschiedliche Werte gesetzt werden: Die integrierten Profibus-Modelle VB33/ VB34 lassen bis zu **128 Byte für den Eingangsbereich** und **32 Byte für den Ausgangsbereich** zu.

## 2.2 Konfigurationseinstellungen

Das Konfigurationstool ermöglicht eine vollständige Parametrierung der Profibus-Kommunikation.

Um die Konnektivität des Profibus herzustellen, muss zuerst der Parameter Data Tx ausgewählt werden (als Standard):

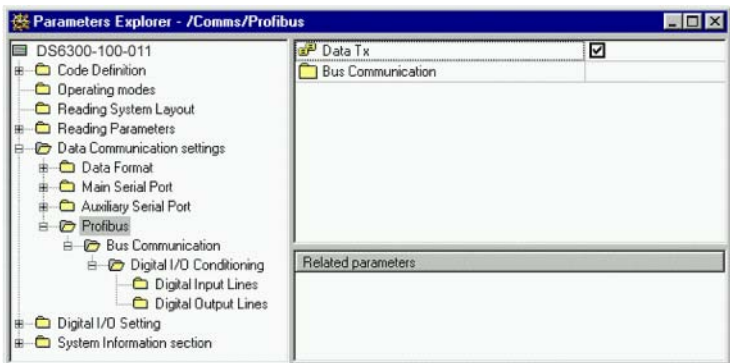


Abbildung 2.3 Parametrierung Data Tx

### 2.2.1 Buskommunikationsparameter

Der Slave wird über seine eigene Adresse (Parameter Node Address) erkannt. Dieser Wert, der für jeden Slave eindeutig ist, muss der Adresse entsprechen, die bei der Konfiguration Ihres Netzwerks angegeben wurde, damit die SPS den Slave richtig erkennen kann.

### 2.2.2 Anleitung zur Auswahl von Profibus-Adressen über die Tastatur:

Um in das Untermenü BUS zu gelangen und den Scanner zu konfigurieren, führen Sie folgende Schritte aus:

1. Halten Sie die Taste ▲ (Pfeil oben) und ▼ (Pfeil unten) für ungefähr 2 Sekunden gedrückt, um ins Hauptmenü zu gelangen.
2. Wählen Sie mithilfe der Taste ▲ (Pfeil oben) oder ▼ (Pfeil unten) den Menüpunkt "Bus", und drücken Sie dann zur Bestätigung die Eingabetaste.
3. Wählen Sie mithilfe der Taste ▲ (Pfeil oben) oder ▼ (Pfeil unten) den Menüpunkt "BusAddrSel", und drücken Sie dann zur Bestätigung die Eingabetaste.
4. Wählen mithilfe der Tasten ▲ (Pfeil oben) oder ▼ (Pfeil unten) Ihre Profibus-Adresse von 0 bis 125. Drücken Sie dann zur Bestätigung die Eingabetaste.
5. Wählen Sie mithilfe der Taste ▲ (Pfeil oben) oder ▼ (Pfeil unten) den Menüpunkt "Exit", und drücken Sie dann zur Bestätigung die Eingabetaste. Wiederholen Sie diesen Schritt, um das Hauptmenü zu verlassen und zur aktuellen Betriebsart des Scanners zurückzukehren.

Die Datenflusssteuerung bietet eine leistungsstarke Möglichkeit, die Kommunikation mit dem Profibus-Master zu steuern und zu optimieren. Ein dediziertes Programm, das auf der SPS läuft, ist erforderlich, um diese Funktion nutzen zu können (Details siehe Abschnitt "Flusssteuerungsmodus (FSM)").

**In der endgültigen Implementierung sollte Flusssteuerung eingesetzt werden, um eine maximale Zuverlässigkeit und eine optimale Synchronisation zwischen Master und Slave zu erreichen.** Als erster Ansatz wird Data Flow Control = Disable vorgeschlagen, um die Hardware-/Softwarekonfiguration des Profibus-Netzwerks zu prüfen.

Im Abschnitt Bus Communication kann auf die Parameter Baud Rate, Master Input Area Size und Master Output Data Size nur lesend zugegriffen werden, da diese Parameter vom Profibus-Master festgelegt werden.



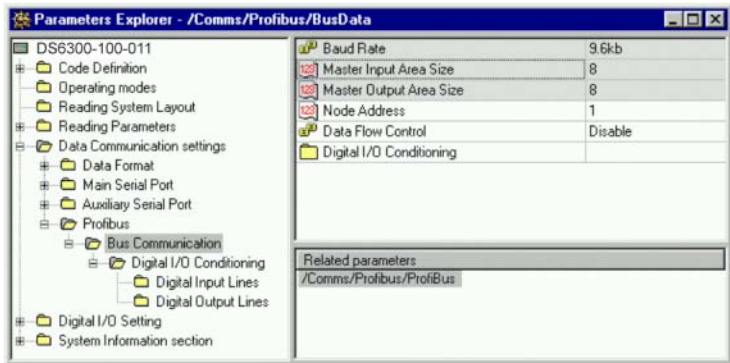


Abbildung 2.4 Parametrierung Baud Rate

## 2.3 Flusssteuerungsmodus (FSM)

Der Flusssteuerungsmodus ist eine leistungsstarke Möglichkeit, um die Kommunikation mit dem Profibus-Master zu steuern und zu optimieren. Durch Aktivierung des FSM werden einige Bytes des Austauschbereichs für Treiberoperationen reserviert. Der Rest wird von der Anwendungsschicht verwendet. Die reservierten Bytes werden zur Realisierung vieler verschiedener Funktionen genutzt, z. B.:

- Flusssteuerung und entsprechendes Puffern in beide Richtungen
- Fragmentierung und Wiederaussetzen von Daten, die die Größe des Austauschbereichs überschreiten
- Synchronisation von Flusssteuerungsnummern
- Dienstzugangspunktorientierte Kommunikation
- Längeninformation



Note

*Wenn die Flusssteuerung deaktiviert ist, werden alle Bytes des Austauschbereichs von der Anwendungsschicht genutzt. Der Eingangsbereich wird immer aktualisiert, wenn ein neues Leseereignis zur Masterstation übertragen werden soll.*

*→ In dieser Situation muss der Master den Eingangsbereich lesen, bevor dieser sich aufgrund einer neuen Nachricht, normalerweise ein neuer Barcode, verändert.*

*→ Außerdem wird nicht erkannt, wenn derselbe Barcode zweimal auftritt, da der Eingangsbereich sich nicht ändert.*

*→ Wenn die Anwendungsdaten die Größe des Eingangsbereichs überschreiten, werden die Daten automatisch abgeschnitten.*

FSM kann über den Parameter Data Flow Control ausgewählt werden.

# PROFIBUS VB3X Buskommunikation

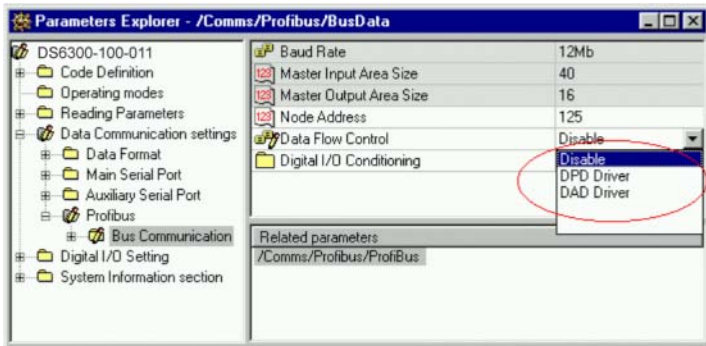


Abbildung 2.5 Parametrierung Data Flow Control

Grundsätzlich stehen drei Optionen zur Verfügung:

- DAD Driver → FSM kompatibel mit "Flow Control = Anybus", verwendet in neuen Geräten
- DPD Driver → FSM kompatibel mit "Flow Control = Profibus", verwendet in Multiplexer
- Disable → Keine Flusssteuerung

Obwohl beide Treiber dieselben oben genannten Funktionen realisieren, stehen derzeit zwei Optionen zur Verfügung, um eine maximale Kompatibilität mit Geräten unterschiedlicher Familien herzustellen.

Zu den unterstützten FSM siehe die folgende Tabelle:

VB3x-Gerät	Identifizierungscode der Konfigurationssoftware.	DAD-Treiber	DPD-Treiber
VB33-2000-P	DS6300-100-011	x	x
VB33-2000-OM-P	DS6300-105-011	x	x
VB34-2500-P	DS6400-100-011	x	x
VB34-2500-OM-P	DS6400-105-011	x	x
Sonstiges Gerät			
MX4000-1100			x
MX4000-1100 SB2498 12MB/s		x	
CBOX-300 Profibus-DP		x	x

Bei der Verwendung von FSM kann die Option Data Consistency aktiviert werden, um die Gesamtsicherheit der Datenkommunikation zu verbessern. Diese Option ist obligatorisch wenn die Größe des Eingangsbereichs des Masters 32 Byte überschreitet.

Weitere Details finden Sie in den nächsten Abschnitten.

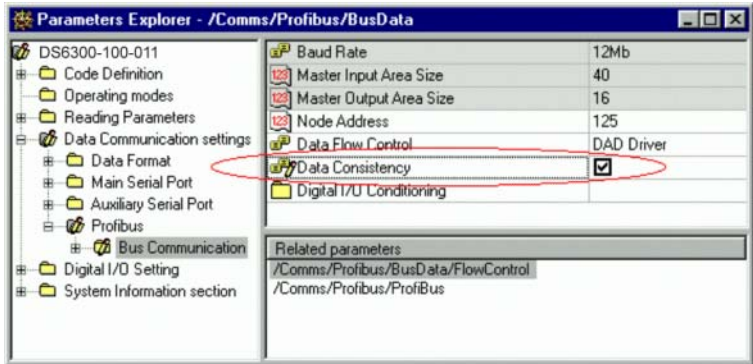


Abbildung 2.6 Datenkonsistenz

### 2.3.1 Flusssteuerungstreiber

Der Flusssteuerungstreiber ist eine Schicht, die auf dem integrierten DP-Datenaustauschmechanismus aufsetzt. Grundsätzlich ist so eine Schicht erforderlich, da der integrierte DP-Profibus-Mechanismus nicht nachrichtenorientiert ist.

In der folgenden Abbildung wird der gesamte Stack dargestellt:

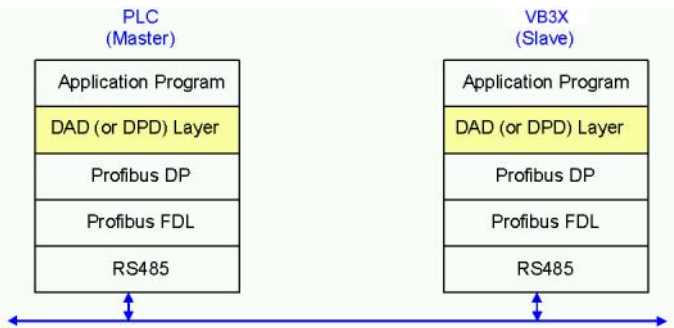


Abbildung 2.7 Flusssteuerungstreiber

Es sind zwei Flusssteuerungstreiber verfügbar:

- DA-Treiber, kompatibel mit Flusssteuerung = Anybus
- DPD-Treiber, kompatibel mit Flusssteuerung = Profibus

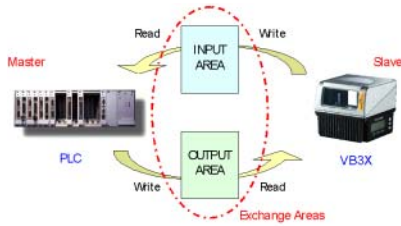
Wie bereits beschrieben können diese bei VB3X einzeln über das Konfigurationstool gewählt werden. Natürlich muss der entsprechende Treiber SPS-seitig implementiert werden.

Der DAD-Treiber sollte bei ganz neuen Installationen die bevorzugte Lösung sein.

## 2.3.2 FSM mit DAD-Treiber

Um die flussgesteuerte Version des Treibers zu implementieren, müssen Austauschbereiche einheitlich in beide Richtungen kompiliert werden.

Im Folgenden bezieht sich der Eingangsbereich auf einen Zwischenspeicher aus InputAreaSize-Bytes:



IN[0]	IN[1]	IN[2]	...	IN[InputAreaSize - 1]
-------	-------	-------	-----	-----------------------

und der Ausgangsbereich auf einen Zwischenspeichern aus OutputAreaSize-Bytes:

OUT[0]	OUT[1]	OUT[2]	...	OUT[OutputAreaSize - 1]
--------	--------	--------	-----	-------------------------

In beiden Speichern werden nur die ersten drei Bytes von der DAD-Treiberschicht genutzt:

- Control-Feld** (Byte 0) dient zum Starten und zur Steuerung der Basiselemente des Treibers wie Flusssteuerung, Fragmentierung und Resynchronisation.
- Service Access Point-Feld** (Byte 1) dient zur Unterscheidung von unterschiedlichen Diensten und zur Sicherstellung zukünftiger Erweiterbarkeit. (Da diese Definition für SAP (Service Access Point = Dienstzugangspunkt) durch den DAD-Treiber eingeführt wurde, darf dies nicht mit dem von der internationalen Norm definierten SAP verwechselt werden).
- Length-Feld** (Byte 2) enthält die von der Anwendungsschicht verwendete Anzahl von Bytes:  
 Length-Feld = (InputAreaSize - 3) für den Eingangsbereich  
 Length-Feld = (OutputAreaSize - 3) für den Ausgangsbereich.

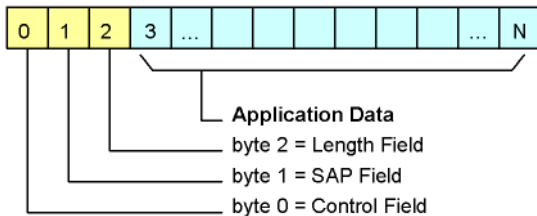


Abbildung 2.8 Speicher Ein-/Ausgangsaustauschbereich

Der Anwendungsdatenspeicher enthält nützliche Informationen, normalerweise die Barcode-Nachrichten, die von dem Anwendungsprogramm bearbeitet wurden. IN[3] enthält das erste signifikante Byte des Anwendungsdatenspeichers (dasselbe erste

Byte, das man sehen würde, wenn VB3X den Barcode-Speicher an den Hilfsschnittstelle anstatt an die Profibus-Schnittstelle übertragen würde).

Die Struktur des Anwendungsspeichers und seine Länge hängen stark von dem gewählten Datenformat des VB3X ab. Barcode-Nachrichten, die länger sind als (InputAreaSize – 3), werden durch einen automatischen Fragmentierungsprozess in einzelne Teile aufgeteilt (Details siehe Abschnitt "Fragmentierung und Wiedierzusammensetzen").

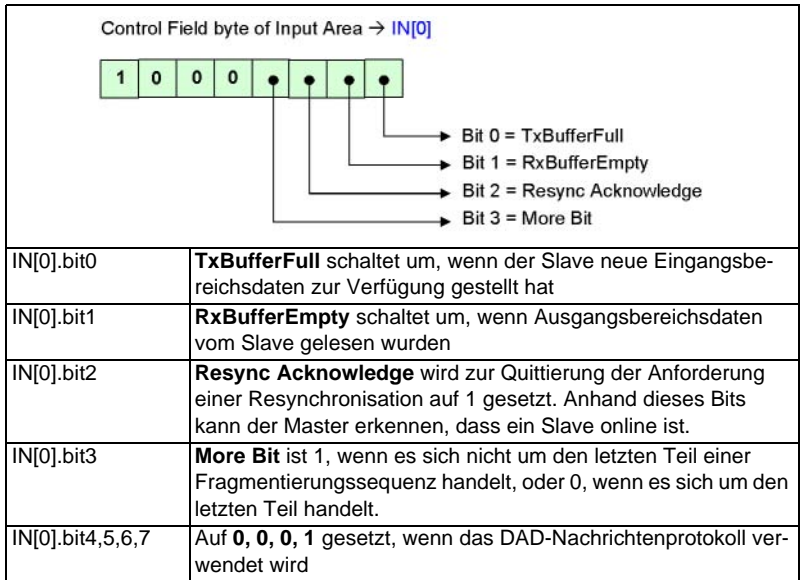
**Control-Feld**

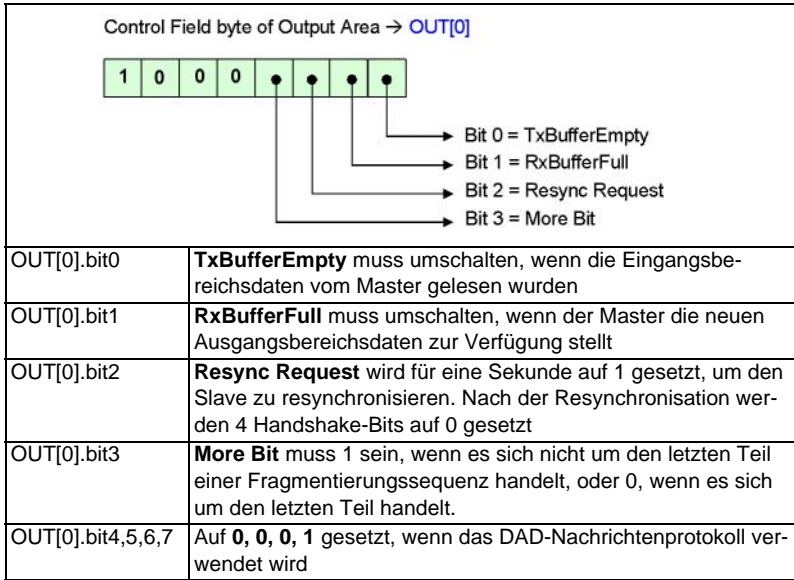
Dies ist der Kern der flussgesteuerten Kommunikation.

Im Eingangsbereich sind Bit 0 und Bit 1 von IN[0] für die Handshake-Funktion reserviert, während im Ausgangsbereich, der symmetrisch ist, Bit 0 und 1 von OUT[0] reserviert sind.

Die Masterstation kann jederzeit über Bit 2 des Ausgangsbereichs eine Resynchronisation anfordern. Dieser Prozess, der die Synchronisierungsbits (Bit 0 und Bit 1 des Eingangs- und Ausgangsbereichs) zurücksetzt, muss von dem Slave an Bit 2 des Eingangsbereichs quittiert werden.

Bit 3 dient zur Steuerung einer Fragmentierungssequenz in beide Richtungen.





## SAP-Feld

SAP (Dienstzugangspunkt) ist ein Bezeichner, der zur Implementierung mehrerer Dienste verwendet wird, die denselben Kommunikationskanal zwischen zwei Remote-Stationen verwenden.

Die folgenden Werte wurden definiert:

- SAP = 0 Dient zur Übertragung von Informationen zwischen VB3X und SPS
- SAP = 255 Reserviert für Treiberdienste (Details siehe Abschnitt "SAP-Dienste")

Alle anderen SAP-Werte sind frei wählbar und können nach Abstimmung zwischen den Anwendungsprogrammen selbst von dedizierten Anwendungsprogrammen verwendet werden.

## Length-Feld

Die Anwendungsschicht verwendet alle oder einen Teil der übrigen Bytes des Austauschbereichsspeichers, die nicht von dem DAD-Treiber genutzt werden. Das Length-Feld wurde eingeführt, um Informationen darüber zu speichern, wie viele Bytes tatsächlich von der Anwendungsschicht verwendet werden.

Ein Fragment, bei dem es sich nicht um den letzten Teil einer Fragmentierungssequenz handelt, muss in dieses Feld [InputAreaSize - 3] (oder [OutputAreaSize - 3]) eintragen, je nachdem, ob es sich um ein Eingangs- oder Ausgangsfragment handelt. Andernfalls erhält dieses Feld einen Wert, der kleiner oder gleich [InputAreaSize - 3] ist.

## Datenübertragung von VB3X zur SPS

In diesem Abschnitt wird beschrieben, wie Nachrichten mit Flusststeuerung ausgetauscht werden. Der Kommunikationsmechanismus ist sehr einfach:

- IN[0].bit0 [A] wird von VB3X verwendet, um zu melden, dass "ein Slave neue Daten geschrieben hat und der Master sie lesen kann"
- OUT[0].bit0 [B] muss von der SPS verwendet werden, um zu melden, dass der "Master die letzten Daten gelesen hat und der Slave die nächste Nachricht senden kann"

Dies passiert immer, wenn Bit A (oder B) seinen Status ändert (umschaltet). Der Bitpegel spielt keine Rolle, nur der Übergang muss berücksichtigt werden.

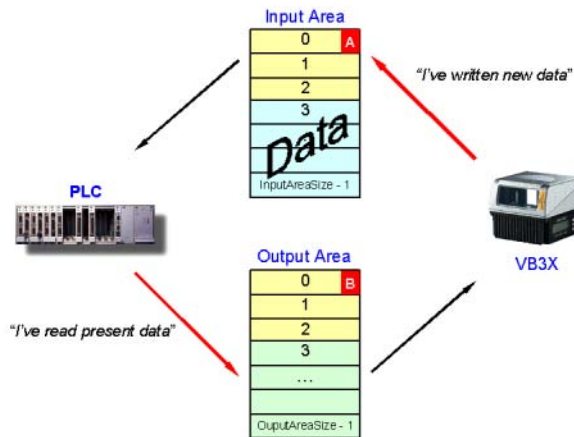


Abbildung 2.9 Nachrichtenaustausch mit Flusststeuerung

Der folgende Zustandsautomat zeigt die Datenübertragung vom Slave zum Master. Bitte beachten Sie, dass jeder Zyklus zwei Datennachrichten überträgt.

# PROFIBUS VB3X Buskommunikation

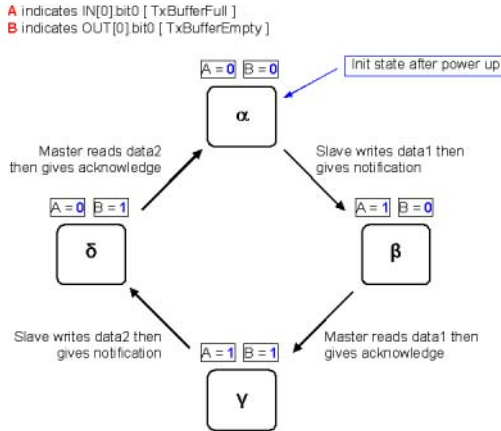


Abbildung 2.10 Zustandsautomat

Im Folgenden wird ein typischer Datenaustausch mit den folgenden Einstellungen analysiert:

- Flow Control = DAD Driver
- Input Area Size = 16
- Output Area Size = 8
- Nach dem Einschalten enthalten die Eingangs- und Ausgangsbereiche normalerweise null. Gemäß der DAD-Treiberimplementierung hat der Eingangsbereich die Werte Control-Feld = 80h und SAP = 00h.

Eingangsbereich	80h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h
Ausgangsbereich	00h 00h 00h 00h 00h 00h 00h 00h

- Außerdem muss die SPS das Control-Feld des Ausgangsbereichs richtig setzen, wenn das DAD-Nachrichtenprotokoll verwendet wird.

Eingangsbereich	80h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h
Ausgangsbereich	80h 00h 00h 00h 00h 00h 00h 00h

- VB3X liest einen Barcode "123456". Es wird von einem Standarddatenformat mit <STX> als Kopf und <CR><LF> als Endezeichen ausgegangen. VB3X schaltet Bit A.

Eingangsbereich	81h 00h 09h 02h 31h 32h 33h 34h 35h 36h 0Dh 0Ah 00h 00h 00h 00h
Ausgangsbereich	80h 00h 00h 00h 00h 00h 00h 00h

- SPS erkennt den Übergang von Bit A und kann nun die eingehenden Daten lesen (sie kopiert 9 Bytes in ihren Speicher ab IN[3]) und schaltet dann Bit B zur Quittierung um.

Ausgabedatum: 06/15/2005



**Hinweis** Vor der Quittierung werden alle anderen von VB3X gelesene Barcodes zwischengespeichert.

Eingangsbereich	81h 00h 09h 02h 31h 32h 33h 34h 35h 36h 0Dh 0Ah 00h 00h 00h 00h
Ausgangsbereich	81h 00h 00h 00h 00h 00h 00h 00h

- VB3X liest einen Barcode "10DL" und schaltet Bit A um.

Eingangsbereich	80h 00h 07h 02h 31h 30h 44h 4Ch 0Dh 0Ah 00h 00h 00h 00h 00h 00h
Ausgangsbereich	81h 00h 00h 00h 00h 00h 00h 00h

- SPS liest die neue Nachricht (sie kopiert 7 Bytes in ihren Speicher ab IN[3]) und schaltet dann Bit B zur Quittierung um.

Eingangsbereich	80h 00h 07h 02h 31h 30h 44h 4Ch 0Dh 0Ah 00h 00h 00h 00h 00h 00h
Ausgangsbereich	80h 00h 00h 00h 00h 00h 00h 00h

- VB3X führt No Read aus und schaltet Bit A um. Es wird von <CAN> als globalem No-Read-Zeichen ausgegangen.

Eingangsbereich	81h 00h 04h 02h 18h 0Dh 0Ah 00h 00h 00h 00h 00h 00h 00h 00h 00h
Ausgangsbereich	80h 00h 00h 00h 00h 00h 00h 00h

- SPS liest die neue Nachricht (sie kopiert 4 Bytes in ihren Speicher ab IN[3]) und schaltet dann Bit B zur Quittierung um.

Eingangsbereich	81h 00h 04h 02h 18h 0Dh 0Ah 00h 00h 00h 00h 00h 00h 00h 00h 00h
Ausgangsbereich	81h 00h 00h 00h 00h 00h 00h 00h

- Der Datenaustausch läuft weiter ...

### Datenübertragung von der SPS zu VB3X

Analog zum vorherigen Abschnitt funktioniert die Flusststeuerung selbst dann, wenn die Daten vom Master zum Slave übertragen werden. Der Kommunikationsmechanismus basiert auf denselben Konzepten:

- rOUT[0].bit1 [C] muss von der SPS verwendet werden, um zu melden, dass der "Master neue Daten geschrieben hat und der Slave diese lesen kann"
- IN[0].bit1 [D] wird von VB3X verwendet, um zu melden, dass "ein Slave Daten gelesen hat und der Master die nächste Nachricht senden kann"

Dies passiert immer, wenn Bit C (oder D) seinen Status ändert (umschaltet). Der Bitpegel spielt keine Rolle, nur der Übergang muss berücksichtigt werden.

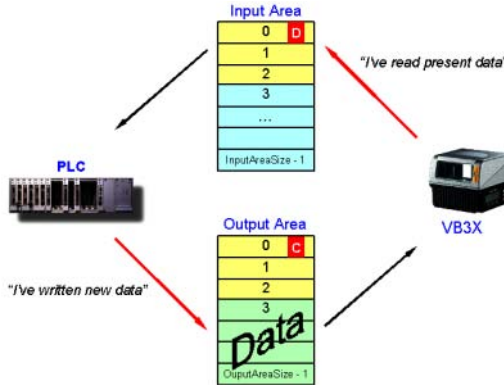


Abbildung 2.11 Nachrichtenaustausch mit Flusststeuerung

Der folgende Zustandsautomat zeigt die Datenübertragung vom Master zum Slave. Bitte beachten Sie, dass jeder Zyklus zwei Datennachrichten überträgt.

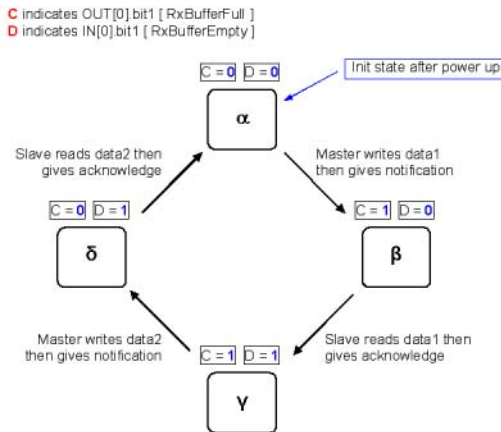


Abbildung 2.12 Zustandsautomat

## Resynchronisation

Der Resynchronisationsprozess startet das Nachrichtenprotokoll ab einem vorgegebenen Status neu.

Er kann entweder beim Starten des Masters verwendet werden, um festzustellen, ob der Slave online ist, oder während des normalen Betriebs, falls Fehler das Zurücksetzen des Protokolls erfordern.

Der Prozess basiert auf Bit zwei:

- OUT[0].bit2 [E] muss von der SPS verwendet werden, um die Resynchronisation anzufordern
- IN[0].bit2 [F] wird von VB3X verwendet, um die Anfrage zu quittieren

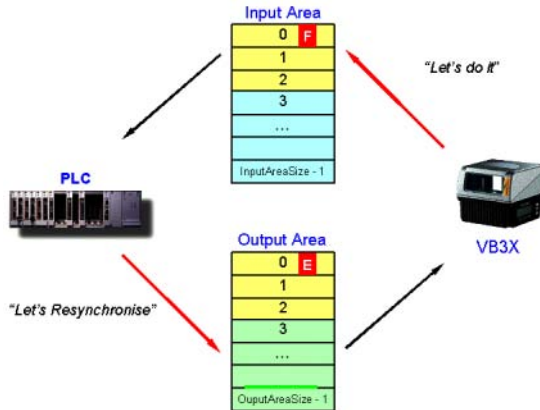


Abbildung 2.13 Resynchronisation

Der folgende Zustandsautomat zeigt den Resynchronisationszyklus, der von der SPS angefordert und zusammen mit VB3X ausgeführt wird:

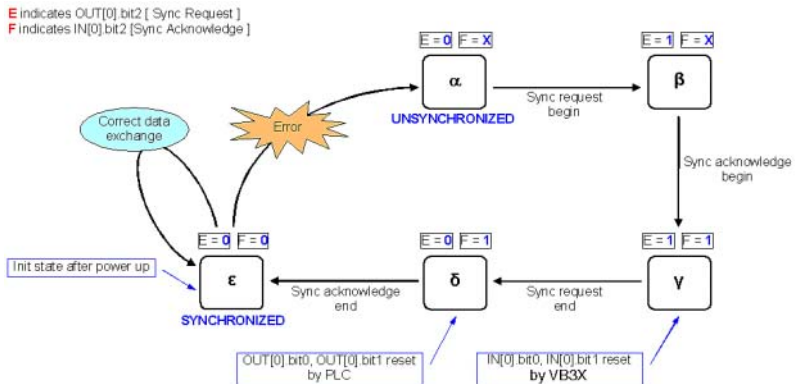


Abbildung 2.14 Zustandsautomat

Im Folgenden wird der Resynchronisationsprozess ab dem vorherigen Datenaustausch, der unter "Datenübertragung von VB3X zur SPS" ... beschrieben wird, analysiert.

Eingangsbe- reich	81h 00h 04h 02h 18h 0Dh 0Ah 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h
Ausgangsbe- reich	81h 00h 00h 00h 00h 00h 00h 00h

- Die SPS fordert durch Setzen von Bit E = 1 eine Resynchronisation an.

Eingangsbereich	81h 00h 04h 02h 18h 0Dh 0Ah 00h 00h 00h 00h 00h 00h 00h 00h
Ausgangsbereich	85h 00h 00h 00h 00h 00h 00h 00h

- VB3X erkennt die Anfrage und setzt daher IN[0].bit0 und IN[0].bit1 zurück. Dann quittiert er dies gegenüber der SPS mithilfe von Bit F.

Eingangsbereich	84h 00h 04h 02h 18h 0Dh 0Ah 00h 00h 00h 00h 00h 00h 00h 00h
Ausgangsbereich	85h 00h 00h 00h 00h 00h 00h 00h

- SPS muss OUT[0].bit0 und OUT[0].bit1 zurücksetzen, bevor die Anfrage mit E = 0 abgeschlossen wird.

Eingangsbereich	84h 00h 04h 02h 18h 0Dh 0Ah 00h 00h 00h 00h 00h 00h 00h 00h
Ausgangsbereich	80h 00h 00h 00h 00h 00h 00h 00h

- VB3X schließt den Quittierungsvorgang durch Setzen von Bit F = 0 ab.

Eingangsbereich	80h 00h 04h 02h 18h 0Dh 0Ah 00h 00h 00h 00h 00h 00h 00h 00h
Ausgangsbereich	80h 00h 00h 00h 00h 00h 00h 00h

- Die Flusssteuerung ist jetzt in einen vorgegebenen Status zurückgekehrt. Alle Datenaustausch-Bits im Control-Feld sind mit Sicherheit null, und die Datenübertragung kann sicher ablaufen.

### Fragmentierung und Wiederzusammensetzen

Der Fragmentierungsprozess wird immer aktiviert, wenn die Anwendungsdaten nicht in den entsprechenden Austauschbereich passen. Grundsätzlich werden lange Nachrichten in Teile zerlegt, die dann einzeln übertragen werden. Beim Wiederzusammensetzen wird die gesamte Nachricht wiederhergestellt.

VB3X implementiert diese Funktionen bereits in der DAD-Schicht, während die SPS eine entsprechende Managementfunktion benötigt.

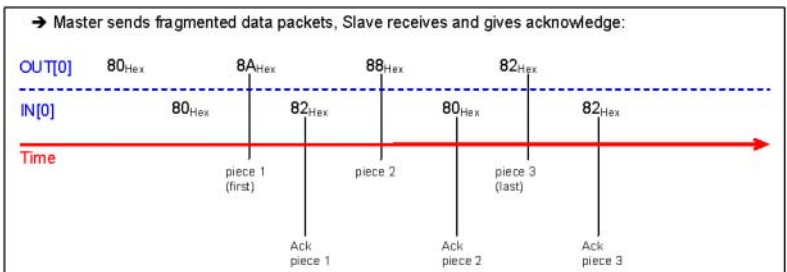
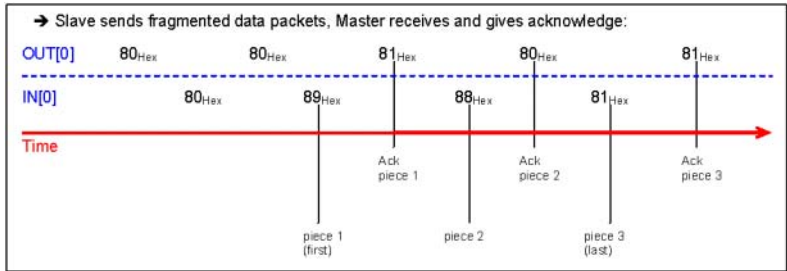
Die Fragmentierung basiert auf dem **More Bit** (Bit 3) in dem Byte des Control-Felds. More Bit = 0 gibt an, dass alle Informationen in der aktuellen Nachricht enthalten sind. Wenn die Anwendungsdaten länger sind als (exchange area size – 3), wird die erste Teilnachricht mit More Bit = 1 übertragen. Bei den folgenden Nachrichten ist ebenfalls More Bit = 1, während beim letzten Teil wieder More Bit = 0 ist. Aufgrund dieses Mechanismus kann der Empfänger den letzten Teil erkennen und so die gesamten Informationen wieder zusammensetzen.

Einige Hinweise:

- VB3X kann Nachrichten von bis zu 256 Byte handhaben.
- Bei Zwischenfragmenten ist das Length-Feld = (exchange area size – 3)
- Beim letzten Fragment ist das Length-Feld = (exchange area size – 3)
- Bit0 und Bit1 der Eingangs- und Ausgangsbereiche werden für jedes Fragment unabhängig verwaltet

Ausgabedatum: 06/15/2005

Die folgenden Abbildungen zeigen, wie sich das Control Byte je nach Fragmentierungsprozess ändert. Beide Datenflussrichtungen werden berücksichtigt.



Im Folgenden wird ein Datenaustausch in Fragmenten mit den folgenden Einstellungen analysiert:

- Flow Control = DAD Driver
  - Input Area Size = 16
  - Output Area Size = 8
- Nach dem Einschalten enthalten die Eingangs- und Ausgangsbereiche normalerweise null. Gemäß der DAD-Treiberimplementierung hat der Eingangsbereich die Werte Control-Feld = 80Hex und SAP = 00Hex.

Eingangsbereich	80h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h
Ausgangsbereich	00h 00h 00h 00h 00h 00h 00h 00h

- Außerdem muss die SPS das Control-Feld des Ausgangsbereichs richtig setzen, wenn das DAD-Nachrichtenprotokoll verwendet wird.

Eingangsbereich	80h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h
Ausgangsbereich	80h 00h 00h 00h 00h 00h 00h 00h

- VB3X liest einen Barcode mit dem Inhalt "1234567890abcde1234567890abcde". Es wird von einem Standarddatenformat mit <STX> als Kopf und <CR><LF> als Endezeichen ausgegangen. Da in diesem Fall nicht die gesamte Nachricht in den Eingangsbereich passt, überträgt VB3X zunächst nur das Fragment "<STX>1234567890ab" (mit More Bit = 1) und schaltet dann Bit A um.

# PROFIBUS VB3X

## Buskommunikation

Eingangsbe- reich	89h 00h 0Dh 02h 31h 32h 33h 34h 35h 36h 37h 38h 39h 30h 61h 62h
Ausgangsbe- reich	80h 00h 00h 00h 00h 00h 00h 00h

- SPS erkennt den Übergang von Bit A und kann nun das erste eingehende Fragment lesen (sie kopiert 13 Bytes in ihren Speicher ab IN[3]) und schaltet dann Bit B zur Quittierung um.

Eingangsbe- reich	89h 00h 0Dh 02h 31h 32h 33h 34h 35h 36h 37h 38h 39h 30h 61h 62h
Ausgangsbe- reich	81h 00h 00h 00h 00h 00h 00h 00h

- VB3X erkennt den Übergang von Bit B, sendet das zweite Fragment "cde1234567890" (immer noch More Bit = 1) und schaltet Bit A um.

Eingangsbe- reich	88h 00h 0Dh 63h 64h 65h 31h 32h 33h 34h 35h 36h 37h 38h 39h 30h
Ausgangsbe- reich	81h 00h 00h 00h 00h 00h 00h 00h

- SPS liest das zweite Fragment (sie kopiert 13 Bytes in ihren Speicher ab IN[3]) und schaltet dann Bit B zur Quittierung um.

Eingangsbe- reich	88h 00h 0Dh 63h 64h 65h 31h 32h 33h 34h 35h 36h 37h 38h 39h 30h
Ausgangsbe- reich	80h 00h 00h 00h 00h 00h 00h 00h

- VB3X sendet das dritte (letzte) Fragment "abcde<CR><LF>" (schließlich More Bit = 0) und schaltet Bit A um.

Eingangsbe- reich	81h 00h 07h 61h 62h 63h 64h 65h 0Dh 0Ah 00h 00h 00h 00h 00h 00h
Ausgangsbe- reich	80h 00h 00h 00h 00h 00h 00h 00h

- SPS liest das letzte Fragment (sie kopiert 7 Bytes in ihren Speicher ab IN[3]), und das Wiederaussetzen kann abgeschlossen werden. Dann schaltet sie Bit B zur Quittierung um.

Eingangsbe- reich	81h 00h 07h 61h 62h 63h 64h 65h 0Dh 0Ah 00h 00h 00h 00h 00h 00h
Ausgangsbe- reich	81h 00h 00h 00h 00h 00h 00h 00h

- Die ganze Nachricht wurde vollständig übertragen.

### SAP-Dienste

FLUSH QUEUE ist der einzige derzeit verfügbare Treiberdienst. Er leert die internen Warteschlangen und kann jederzeit gestartet werden.

FLUSH QUEUE-Dienst
--------------------

Ausgabedatum: 06/15/2005

Anforderung:	Pufferspeicher leeren (gestellt von der Masterstation an den VB3X)
Aktion:	Alle Informationen aus vorherigen Entschlüsselungsvorgängen löschen
Antwort:	Befehl angenommen/Befehl abgelehnt (erzeugt von VB3X an Master)

Die Anwendungsdatenbereiche müssen folgendermaßen formatiert sein:

Aufforderungsbehl	Byte 3	Byte 4
Pufferspeicher leeren	'I' (5B Hex)	'F' (46 Hex)

Antwortbehl	Byte 3	Byte 4
Befehl angenommen	'A' (41 Hex)	' ' (20 Hex)
Befehl abgelehnt	'B' (43 Hex)	' ' (20 Hex)

### DAD-interne Warteschlangen

VB3X verfügt über zwei interne Warteschlangen (eine pro Richtung) für die Anwendungsereignisse: Eingangswarteschlange und Ausgangswarteschlange.

Die Eingangswarteschlange wird verwendet, wenn eine Nachricht (üblicherweise ein Barcode) von VB3X übertragen werden soll, bevor die Masterstation alle Acknowledge-Handshakes pro die vorherige Übertragung generiert hat.

Die Ausgangswarteschlange wird derzeit selten verwendet.

Die Größe der Warteschlangen beträgt:

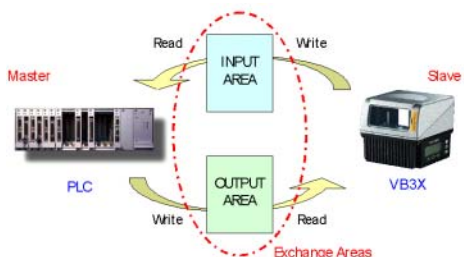
- 350 Elemente von je 128 Byte für die Eingangswarteschlange (Datenfluss vom Slave zum Master)
- 26 Elemente von je 32 Byte für die Ausgangswarteschlange (Datenfluss vom Master zum Slave)

Die Warteschlangen können von der Masterstation über SAP = 255 einfach geleert werden. Dies geschieht normalerweise beim Starten des Masters, falls die Masterstation alle vorherigen Zwischenspeicher, die vor dem Starten generiert wurden, leeren möchte. Die Masterstation kann jedoch auch beschließen, diese nicht zu leeren.

### 2.3.3 FSM mit DPD-Treiber

Um die flussgesteuerte Version des Treibers zu implementieren, müssen Austauschbereiche einheitlich in beide Richtungen kompiliert werden.

Dieser Treiber unterstützt alle im Abschnitt "FSM mit DAD-Treiber" aufgeführten Funktionen und alle oben beschriebenen Handshake-Mechanismen.



# PROFIBUS VB3X Buskommunikation

Das Feld Profibus Station Address ist die einzige abweichende Anforderung. Die ersten vier Bytes werden von der DPD-Treiberschicht in beiden Speichern wie folgt genutzt

<b>Control-Feld</b> (Byte 0)	dient zum Speichern und zur Steuerung der Basiselemente des DPD-Treibers wie Flusssteuerung, Fragmentierung und Resynchronisation.
<b>Profibus Station Address-Feld</b> (Byte 1)	enthält Informationen zu der gewählten VB3X-Adresse des Profibus-Netzwerks.
<b>Profibus Station Address-Feld</b> (Byte 2)	dient zur Unterscheidung von unterschiedlichen Diensten und zur Sicherstellung zukünftiger Erweiterbarkeit. (Da diese Definition für SAP (Service Access Point = Dienstzugangspunkt) durch den DPD-Treiber eingeführt wurde, darf dies nicht mit dem von der internationalen Norm definierten SAP verwechselt werden).
<b>Length-Feld</b> (Byte 3)	enthält die von der Anwendungsschicht verwendete Anzahl von Bytes: Length-Feld = (InputAreaSize – 4) für den Eingangsbereich Length-Feld = (OutputAreaSize – 4) für den Ausgangsbereich.

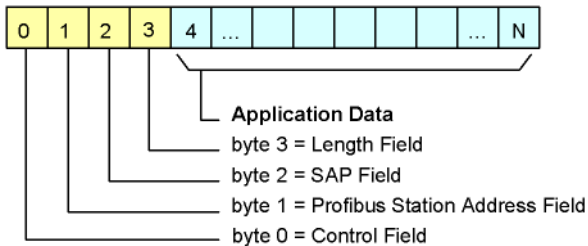


Abbildung 2.15 Speicher Ein-/Ausgangsaustauschbereich

Der Anwendungsdatenspeicher enthält die Informationen ab Byte IN[4]. Der Fragmentierungsprozess wird von VB3X gesteuert, wenn die Barcode-Nachrichten länger sind als (InputAreaSize – 4).

## 2.4 Datenkonsistenz

Bei der Verwendung von Flusssteuerung (entweder DAD- oder DPD-Treiber) ist die Option Data Consistency verfügbar.

Ist diese Option aktiviert, implementiert der Treiber einen bestimmten Mechanismus, um die Konsistenz der übertragenen und empfangenen Daten über die gesamte Größe des Austauschbereichs zu gewährleisten.



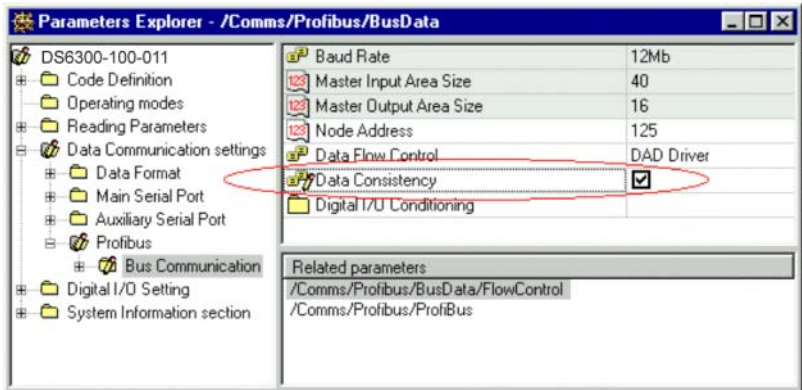


Abbildung 2.16 Datenkonsistenz

### 2.4.1 Datenkonsistenz mit DAD-Treiber

Wie in der folgenden Abbildung gezeigt, kopiert die Anwendungsschicht das Byte des Control-Felds in die letzte Position des Austauschbereichs. Wenn sich das Control-Feld von Nachricht zu Nachricht ändert, bleibt auf diese Weise die gesamte Nachricht konsistent, solange das erste und letzte Byte übereinstimmen.

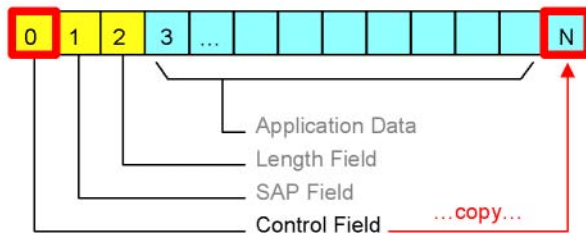


Abbildung 2.17 Speicher Ein-/Ausgangsaustauschbereich

Aus praktischer Sicht gilt eine vollständige Nachricht im Eingangsbereich als konsistent, sobald die SPS bestätigt, dass  $IN[0]$  gleich  $IN[InputAreaSize - 1]$  ist.

Die SPS sollte dafür sorgen,  $OUT[0]$  in  $OUT[OutputAreaSize - 1]$  zu doppeln, um den Treiber die Konsistenz prüfen zu lassen.

Aufgrund des neuen, vom DAD-Treiber verwendeten Bytes werden Barcode-Nachrichten, die länger sind als  $(InputAreaSize - 4)$ , durch einen automatischen Fragmentierungsprozess in einzelne Teile aufgeteilt.

Im Folgenden wird ein konsistenter Datenaustausch in Fragmenten mit den folgenden Einstellungen analysiert:

- Flow Control = DAD Driver
- Data Consistency = Enable
- Input Area Size = 16
- Output Area Size = 8

# PROFIBUS VB3X

## Buskommunikation

- Nach dem Einschalten enthalten die Eingangs- und Ausgangsbereiche normalerweise null. Gemäß der DAD-Treiberimplementierung hat der Eingangsbereich die Werte Control-Feld IN[0] = IN[15] = 80Hex und SAP = 00Hex.

Eingangsbereich	80h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h
Ausgangsbereich	00h 00h 00h 00h 00h 00h 00h 00h

- PLC muss gemäß dem DAD-Nachrichtenprotokoll mit Datenkonsistenz Control-Feld = OUT[0] = OUT[7] = 80Hex setzen.

Eingangsbereich	80h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 00h 80h
Ausgangsbereich	80h 00h 00h 00h 00h 00h 00h 80h

- VB3X liest einen Barcode mit dem Inhalt "1234567890abcde1234567890abcde". Es wird von einem Standarddatenformat mit <STX> als Kopf und <CR><LF> als Endezeichen ausgegangen. VB3X überträgt nur das erste Fragment "<STX>1234567890a" (mit More Bit = 1) und schaltet dann Bit A um.

Eingangsbereich	89h 00h 0Ch 02h 31h 32h 33h 34h 35h 36h 37h 38h 39h 30h 61h 89h
Ausgangsbereich	80h 00h 00h 00h 00h 00h 00h 80h

- SPS erkennt den Übergang von Bit A und kann nun das erste eingehende Fragment lesen (sie kopiert 12 Bytes in ihren Speicher ab IN[3]) und schaltet dann Bit B zur Quittierung um.

Eingangsbereich	89h 00h 0Ch 02h 31h 32h 33h 34h 35h 36h 37h 38h 39h 30h 61h 89h
Ausgangsbereich	81h 00h 00h 00h 00h 00h 00h 81h

- VB3X erkennt den Übergang von Bit B, sendet das zweite Fragment "bcde12345678" (immer noch More Bit = 1) und schaltet Bit A um.

Eingangsbereich	88h 00h 0Ch 62h 63h 64h 65h 31h 32h 33h 34h 35h 36h 37h 38h 88h
Ausgangsbereich	81h 00h 00h 00h 00h 00h 00h 81h

- SPS liest das zweite Fragment (sie kopiert 12 Bytes in ihren Speicher ab IN[3]) und schaltet dann Bit B zur Quittierung um.

Eingangsbereich	88h 00h 0Ch 62h 63h 64h 65h 31h 32h 33h 34h 35h 36h 37h 38h 88h
Ausgangsbereich	80h 00h 00h 00h 00h 00h 00h 80h

- VB3X sendet das letzte Fragment "90abcde<CR><LF>" (schließlich More Bit = 0) und schaltet Bit A um.

Eingangsbereich	81h 00h 09h 39h 30h 61h 62h 63h 64h 65h 0Dh 0Ah 00h 00h 00h 81h
Ausgangsbereich	80h 00h 00h 00h 00h 00h 00h 80h

Ausgabedatum: 06/15/2005

- SPS liest das letzte Fragment (sie kopiert 9 Bytes in ihren Speicher ab IN[3]), und das Wiederzusammensetzen kann abgeschlossen werden. Dann schaltet sie Bit B zur Quittierung um.

Eingangsbe- reich	81h 00h 09h 39h 30h 61h 62h 63h 64h 65h 0Dh 0Ah 00h 00h 00h 81h
Ausgangsbe- reich	81h 00h 00h 00h 00h 00h 00h 81h

- Die ganze Nachricht wurde vollständig übertragen.

### 2.4.2 Datenkonsistenz mit DPD-Treiber

Auch in diesem Fall kopiert die Anwendungsschicht das Byte des Control-Felds in die letzte Position des Austauschbereichs. Wie die folgende Abbildung zeigt, ist die Datenaustauschstruktur jedoch leicht unterschiedlich:

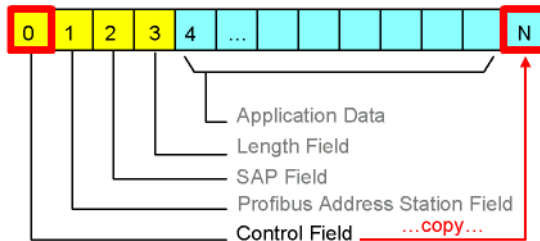


Abbildung 2.18 Speicher Ein-/Ausgangsaustauschbereich

Wird der DPD-Treiber verwendet, werden Barcode-Nachrichten, die länger sind als (InputAreaSize – 5), durch einen automatischen Fragmentierungsprozess in einzelne Teile aufgeteilt.

## 3 Digitale I/O-Konditionierung

Diese Funktion bietet die Möglichkeit, den Scanner-Eingang/Ausgang als Remote-Peripheriegerät des Hosts zu nutzen:

- Die digitale Ausgang-Konditionierung ermöglicht der SPS, den Pegel der drei physikalischen VB3X-Ausgänge zu erzwingen
- Die digitale Eingang-Konditionierung ermöglicht es dem VB3X, den Status seiner vier physikalischen Eingänge der SPS zu melden

Einige der Vorteile sind:

- Der Profibus-Master kennt den Status jedes einzelnen Scannereingangs, d.h., ein Paket kommt an, ein übergroßes Datenpaket wurde erkannt, oder ein Datenpaket hat die Tx-Datenleitung erreicht usw...
- Der Profibus-Master kann den Scanner in den Lesemodus zwingen
- Der Profibus-Master kann den Status der Scannerausgänge je nach empfangener Barcode-Nachricht ändern (Sortieranwendung)

# PROFIBUS VB3X

## Digitale I/O-Konditionierung

- Auf dem Scanner ist keine dedizierte Software zur Steuerung der Ein-/Ausgänge notwendig

Die digitale I/O Konditionierung des VB3X wird aktiviert, wenn einer der nachfolgenden Parameter überprüft wird:

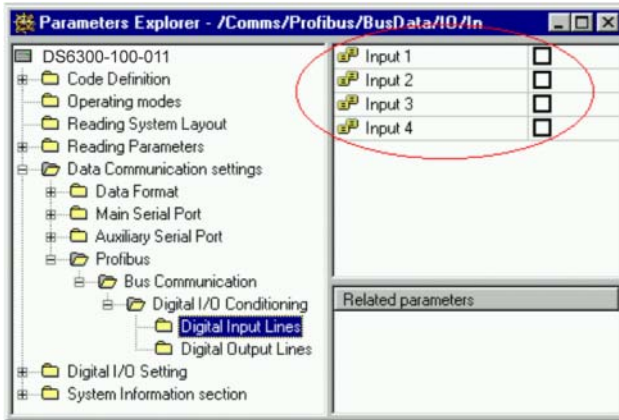


Abbildung 3.1 Digitale Eingangsleitungen

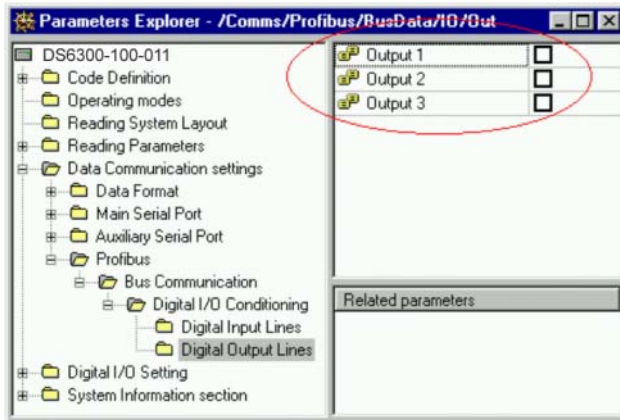


Abbildung 3.2 Digitale Ausgangsleitungen

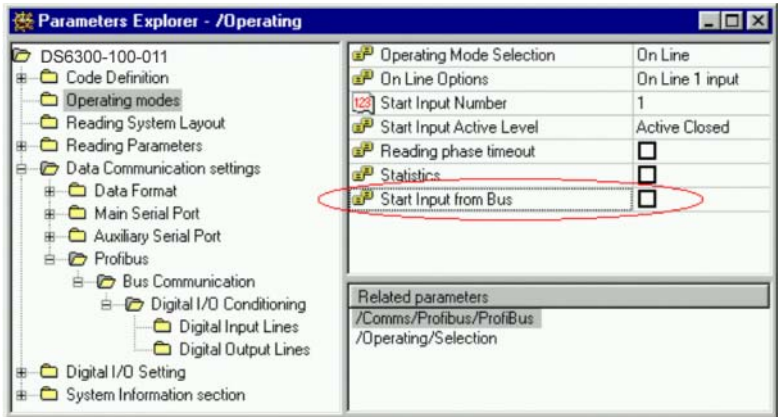


Abbildung 3.3 Betriebsarten

Die digitalen I/Os des VB3X werden daher, wie in den nächsten Abschnitten beschrieben, auf das erste Byte des Eingangs- und Ausgangsaustauschbereichs abgebildet.

### 3.1 Digitale Eingang-Konditionierung



Die I/O-Konditionierungsfunktion arbeitet unabhängig von den Optionen der Flusssteuerung.

Ohne einen Flusssteuerungstreiber ist das folgende Eingangs-Mapping verfügbar:

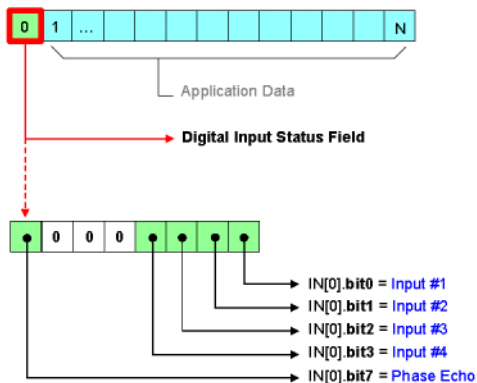


Abbildung 3.4 Speicher Eingangsaustauschbereich

Mit Flusststeuerungstreibern sieht die Struktur des Eingangsbereichs folgendermaßen aus:

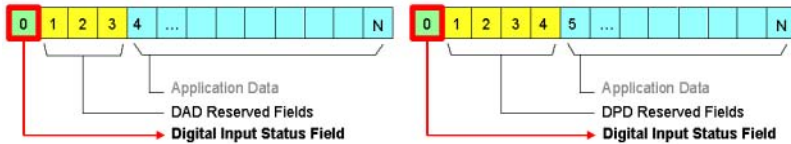


Abbildung 3.5 Digitale Eingangskonditionierung mit DAD-Treiberpuffer und mit DPD-Treiberpuffer

Das Feld Digital Input Status wird durch den VB3X ständig entsprechend den Status der physikalischen Eingänge des Scanners aktualisiert. Nur die ausgewählten Eingänge werden wiedergegeben, wie z. B. Input#1 und Input#3 in der Abbildung unten:

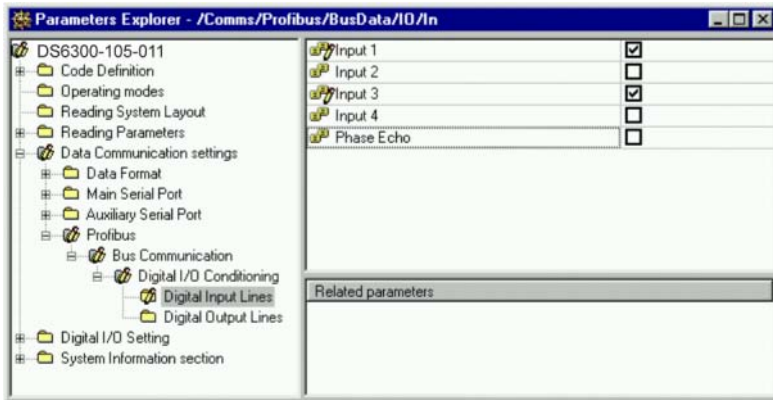


Abbildung 3.6 Digitale Eingangsleitungen

### 3.1.1 Phase Echo

Das Bit Phase Echo gibt den Status der Lesephase an, die entfernt auf dem VB3X aktiviert wurde.

Mithilfe dieser Funktion erhält der Master Feedback über die aktuell aktivierten Lesevorgänge über das Profibus-Netzwerk.

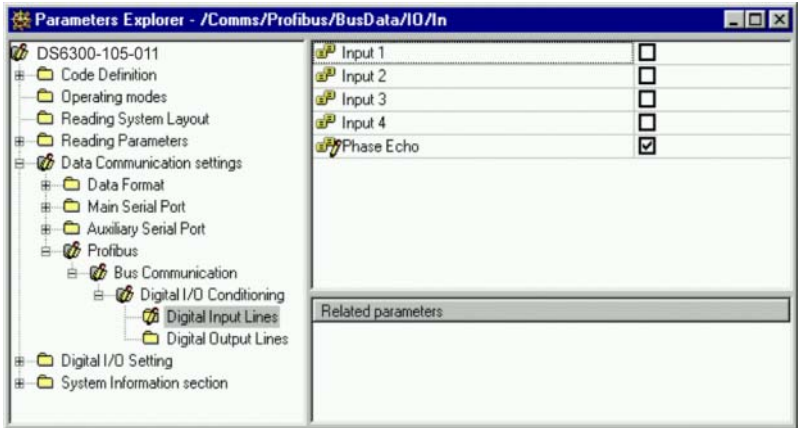


Abbildung 3.7 Parametrierung Phase Echo

Der Parameter Phase Echo kann aktiviert werden, nachdem der Parameter Start Input From Bus ausgewählt wurde. Details siehe Abschnitt "Lesephase über Profibus".

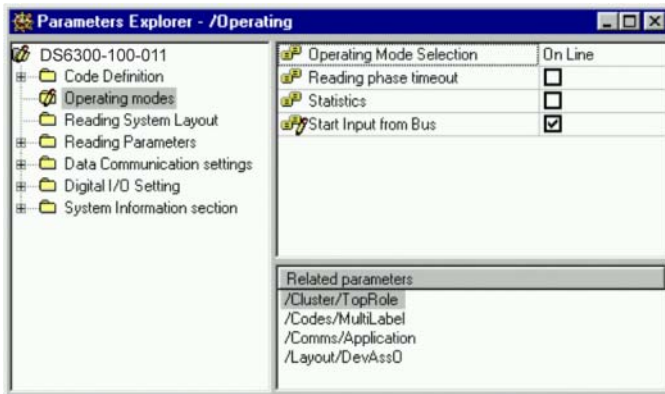


Abbildung 3.8 Eingang vom Bus starten

### 3.2 Digitale Ausgang-Konditionierung



Note

Die I/O-Konditionierungsfunktion arbeitet unabhängig von den Optionen der Flusststeuerung.

Ohne einen Flussteuerungstreiber ist das folgende Ausgangs-Mapping verfügbar:

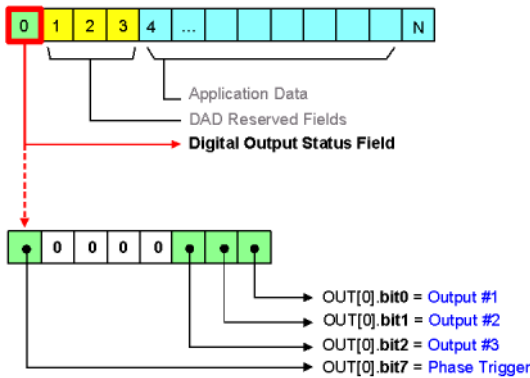


Abbildung 3.9 Speicher Ausgangsaustauschbereich

Mit Flussteuerungstreibern sieht die Struktur des Ausgangsbereichs folgendermaßen aus:

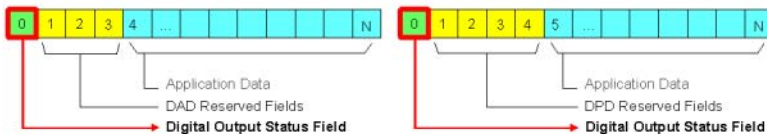


Abbildung 3.10 Digitale Ausgangskonditionierung mit DAD-Treiberpuffer und mit DPD-Treiberpuffer

Das Feld Digital Output Status kann von der SPS jederzeit gesetzt werden. Die physikalischen Ausgänge des VB3X ändern sich dann entsprechend. Es werden also nur die ausgewählten Ausgänge kontrolliert.

### 3.2.1 Phase Trigger

Das Bit Phase Trigger wird vom Master verwendet, um die Lese-Phase des Scanners zu erzwingen. Die Aktivierung des Lesevorgangs erfolgt über das Profibus-Netzwerk. Daher sind keine externen Fotozellen oder I/O-Signale erforderlich.

Phase Trigger ist aktiviert, sobald der Parameter Start Input From Bus ausgewählt wird. Details siehe Abschnitt "Lese-Phase über Profibus".



*Eine alternative Möglichkeit, jede Lese-Phase über Profibus zu öffnen und zu schließen, ist es, den VB3X im Modus Serial On Line zu betreiben und dann einen vom Anwender programmierbaren Serial Start String und Serial Stop String durch die SPS zu senden.*

### 3.2.2 Lese-Phase über Profibus

Diese Implementierung erfordert fünf Schritte:

1. Verbinden Sie den Scanner mit dem Konfigurationstool.



- Überprüfen Sie den Parameter Start Input from Bus in dem Ordner Operating Modes, so dass der Scanner weiß, dass der Phase Trigger über den Profibus geliefert wird.

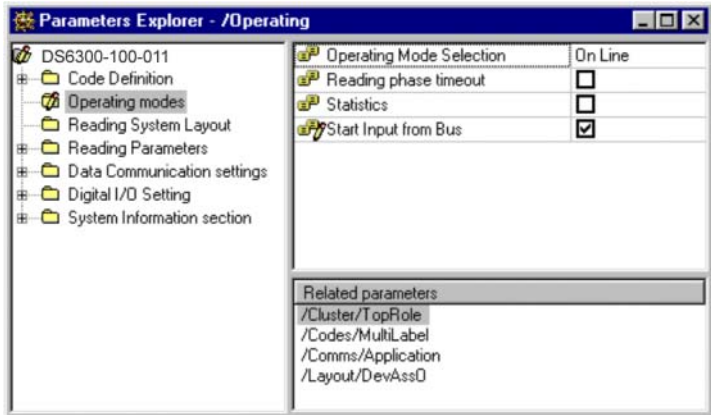


Abbildung 3.11 Eingang vom Bus starten

- Senden Sie die geänderte Konfiguration an den permanenten Speicher (EEPROM) des VB3X

Die Lesephase des Scanners wird nun folgendermaßen gesteuert:

- Die Lesephase startet, sobald der Master OUT[0].bit7 setzt
- Die Lesephase endet, sobald der Master OUT[0].bit7 löscht

## 4 Netzwerkkonfiguration

### 4.1 GSD-Datei

Eine GSD-Datei ist eine lesbare ASCII-Textdatei, die eine vollständige Beschreibung des jeweiligen Geräts beinhaltet. Grundsätzlich enthält die GSD-Datei sowohl allgemeine Informationen (d. h. Name des Anbieters und des Geräts, Hardware-/ Software-Versionen) als auch gerätespezifische Informationen (Größe des Eingangs- und Ausgangsbereichs, Kommunikationsparameter, Einstellungsparameter des Scanners usw.).

Für die Einrichtung eines Profibus-Netzwerks stehen leistungsstarke Konfigurationstools (z. B. Siemens SIMATIC Manager) zur Verfügung. Auf der Grundlage der GSD-Dateien ermöglichen diese Tools die einfache Konfiguration von Profibus-Netzwerken mit Geräten unterschiedlicher Hersteller.



Zunächst muss eine GSD-Datei in der SPS-Umgebung installiert werden, damit ein neues Gerät erkannt wird und im Profibus-Netzwerk arbeiten kann. Folgen Sie den Anweisungen im Abschnitt "GSD-Installation".

# PROFIBUS VB3X Netzwerkconfiguration

VB3X ist mit den folgenden Dateien ausgestattet:

- PF\_07F0.GSD Gerätebeschreibungsdatei
- PFVB34.DIB
- PFVB34DI.DIB Kundenspezifische Erkennungssymbole
- PFVB34SF.DIB

Die GSD-Datei ist ein zertifizierter Teil des Geräts und darf nicht manuell geändert werden. Die Datei wird durch das Konfigurationstool nicht geändert.

## 4.2 GSD-Installation

Im ersten Schritt muss der VB3X als neuer Profibus-DP-Slave dem Katalog von geeigneten Geräten für die SPS hinzugefügt werden.

1. Installieren Sie die neue GSD-Datei ...



2. Suchen Sie die GSD-Datei ...

(GSD- und \*.DIB-Dateien müssen sich im selben Verzeichnis befinden)

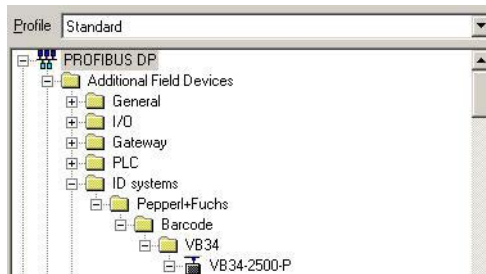


3. Aktualisieren Sie den Katalog ...



#### 4. Suchen Sie das neue Gerät ...

Ein neues VB3X-Gerät erscheint im SPS-Katalog unter Profibus-DP → Additional Field Devices → ID Systems → Pepperl+Fuchs → Barcode → VB3X-Ordner



#### Fügen Sie das Gerät in das Profibus-Netzwerk ein ...

Die einfache Drag&Drop-Funktion ermöglicht das Einfügen des VB3X-Geräts in Ihr eigenes Netzwerk.

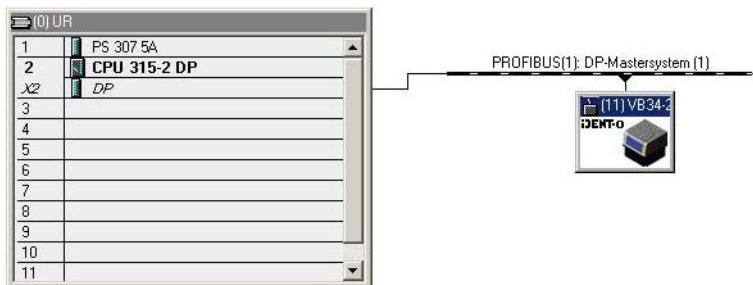


Abbildung 4.1 Einfügen des VB3X-Geräts in Ihr eigenes Netzwerk

## 4.3 Scannerprogrammierung über die GSD-Datei

Der Hauptvorteil der GSD-Datei ist die Einrichtung des Scanners mithilfe des Profibus-Masters, ohne das Konfigurationstool einzusetzen: Es ist möglich, die Barcodes, die Betriebsart des Scanners und viele andere wichtige Parameter auszuwählen.

Verschiedene Scannerparameter werden in Projektmodule unterteilt, die zur Erstellung kundenspezifischer Konfigurationen verwendet werden können. Der Anwender sollte die gewünschten Module in sein eigenes SPS-Projekt integrieren und den richtigen Wert für jeden Parameter auswählen. Sobald die Einstellung abgeschlossen ist, parametrisiert die SPS den Scanner automatisch bei jedem Einschalten und jedes Mal, wenn dieser im Netzwerk wiedererkannt (wiederangeschlossen) wird.

Da die Scannerkonfiguration in der SPS gespeichert ist, kann der Austausch des Scanners einfach und schnell erfolgen.

### 4.3.1 Projektmodule

Es gibt drei Arten von Modulen:

- [IOMCxx]-Module → I/O-Modul mit Datenkonsistenz über den gesamten String
- [IOMxx]-Module → I/O-Modul mit Datenkonsistenz über Bytes/Wörter
- [Mxx]-Modul → Scannerparametermodul

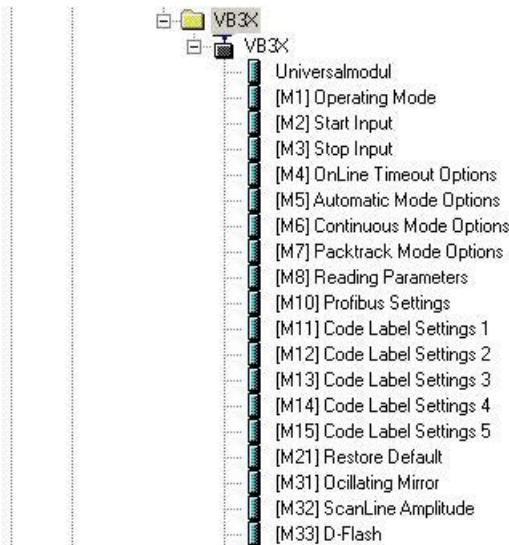


Abbildung 4.2 Überblick über die Module

Mindestens ein I/O-Modul ist erforderlich, um die Größe der Eingangs- und Ausgangsaustauschbereiche einzustellen. Falls erforderlich, können ein oder mehrere [IOMCxx]- (oder [IOMxx]-) Module kombiniert werden.

Eine Parametrierung des Scanners ist optional. In diesem Fall muss das Modul Operating Mode [M1] immer vorhanden sein.

Es wird empfohlen, das Modul Restore Default [M21] zuerst einzusetzen, um eine einheitliche Konfiguration zu ermöglichen. Alle verfügbaren Module mit Ausnahme der I/O-Module werden normalerweise in der Reihenfolge ausgeführt, in der sie hinzugefügt werden.

Im Folgenden werden Details zu den einzelnen Modulen aufgeführt:

### Modul [M1] Operating Mode

Parameter	Optionen
Operating Mode	"On Line 1In" "On Line 2In" "On Line 1In/Timeout" "On Line ProfiBus" "On Line ProfiBus/Timeout" "Serial On Line" "Serial On Line/Timeout" "Continuous" "Test""Packtrack" "Automatic"

### Modul [M2] Start Input Modul [M3] Stop Input

Parameter	Optionen
Presence Sensor	"Input 1" "Input 2" "Input 3" "Input 4"
PS Active Level	"Active Closed" "Active Open"

### Modul [M4] OnLine Timeout Options

Parameter	Optionen
Timeout	"40(ms)" "50(ms)" "70(ms)" "100(ms)" "150(ms)" "200(ms)" "500(ms)" "1000(ms)" "2000(ms)" "3000(ms)" "4000(ms)" "5000(ms)" "7000(ms)" "10000(ms)" "12000(ms)" "15000(ms)"

**Modul [M5] Automatic Mode Options  
Modul [M6] Continuous Mode Options**

Parameter	Optionen
Code Filter Depth	"0" ... "10" (in Schritten von 1) "15" ... "50" (in Schritten von 5)
Threshold	"10 scans" "20 scans" "50 scans" "80 scans" "100 scans" "150 scans" "200 scans" "250 scans" "300 scans" "350 scans" "400 scans" "500 scans" "750 scans" "1000 scans" "5000 scans" "10000 scans" "15000 scans" "20000 scans" "30000 scans"

**Modul [M7] Packtrack Mode Options**

Parameter	Optionen
Physical Encoder	"Enable" "Disable"
Tx Edge	"Trailing" "Leading"
Max Number Of Packs	"2" ... "15"

**Modul [M8] Reading Parameters**

Parameter	Optionen
Reading Mode	"Reconstruction" "Linear"
Overflow Ratio	"0" ... "50"
Beam Shutter	"Disable" "Triggered" "Enable"

### Modul [M10] Profibus Settings

Parameter	Optionen
Flow Control	"Disable" "DAD Driver" "DPD Driver"
Data Consistency	"Enable" "Disable"
Input 1	"Enable" "Disable"
Input 2	"Enable" "Disable"
Input 3	"Enable" "Disable"
Input 4	"Enable" "Disable"
Phase Echo	"Enable" "Disable"
Output 1	"Enable" "Disable"
Output 2	"Enable" "Disable"
Output 3	"Enable" "Disable"

### Module [M11] Code Label Setting 1 ... [M15] Code Label Setting 5

Parameter	Optionen
Code Type	"Code 128" "Interleaved 2 of 5" "Code 39" "Code EAN 128" "EAN-13" "EAN-8" "UPC-A" "UPC-E" "All EAN_UPC" "CODABAR" "Code 93"
Label Length	"Variable" "1" ... "48"



**Modul [M21] Restore Default**

Parameter	Optionen
<Keine Parameter>	<Keine Optionen>

**Modul [M31] Oscillating Mirror Settings**

Parameter	Optionen
OM Mode	Continuous Fixed
OM Angle	"-2,5" ... "37,5" (in Schritten von 0,5)
OM Frequency	"0,0" ... "6,0" (in Schritten von 0,1) "6,5" ... "19,5" (in Schritten von 0,5)

**[M32] Modul ScanLine Amplitude Settings**

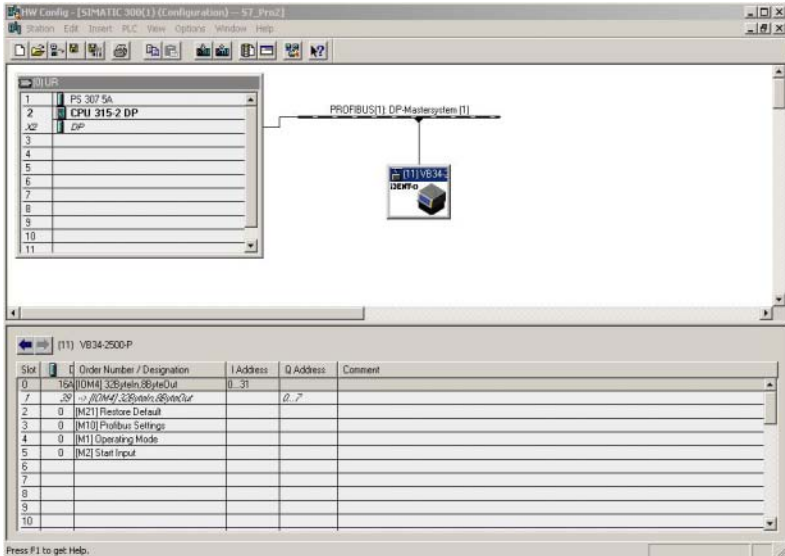
Parameter	Optionen
Scan Line Position	"0" ... "255" (in Schritten von 5)

**[M33] Modul D-Flash Settings**

Parameter	Optionen
Offset Distance	"-255" ... "255" (in Schritten von 5)
Behaviour	"Nearest" "Background Suppression"
StandBy Distance	"1" ... "8" (in Schritten von 1)
Distance 1 up to Distance 8	"45" ... "190" (in Schritten von 5)

# PROFIBUS VB3X Netzwerkkonfiguration

Ein typisches Scannerparametrierungsprojekt sieht folgendermaßen aus:



Hier sehen Sie:

- **[IOMC4]-Modul** zum Setzen der Ein-/Ausgangsaustauschbereiche.
- **[M21]-Modul** zum Wiederherstellen der Standardeinstellungen der Scannerkonfiguration.
- **[M10]-Modul** zum Definieren möglicher Optionen für Flusssteuerung und/oder Datenkonsistenz.
- **[M1]-Modul** zum Auswählen der Betriebsart des Scanners, z. B. OnLine.
- **[M2]-Modul** zur Angabe, welcher digitale Eingang den Trigger liefert.

Vollständige Beschreibungen der Parameter enthält die Online-Hilfe des Konfigurationstools.

## **Notizen**

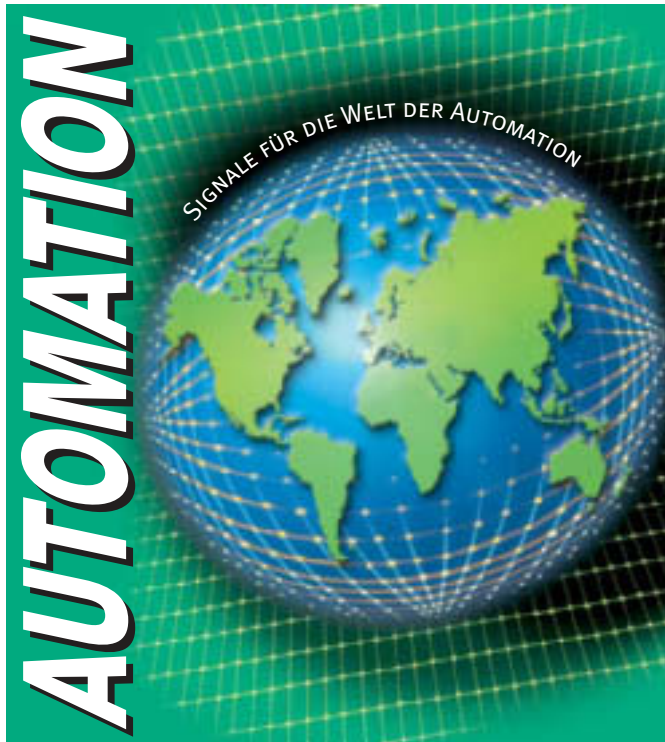






Es gelten die Allgemeinen Lieferbedingungen für Erzeugnisse und Leistungen der Elektroindustrie,  
herausgegeben vom Zentralverband Elektrotechnik und Elektroindustrie (ZVEI) e.V.  
in ihrer neuesten Fassung sowie die Ergänzungsklausel: "Erweiterter Eigentumsvorbehalt"

Wir von PEPPERL+FUCHS/VISOLUX fühlen uns verpflichtet, einen Beitrag für die Zukunft zu leisten,  
deshalb ist diese Druckschrift auf chlorfrei gebleichtem Papier gedruckt.



[www.pepperl-fuchs.com](http://www.pepperl-fuchs.com)

Tel. (0621) 776-1111 · Fax (0621) 776-27-1111 · E-Mail: [fa-info@de.pepperl-fuchs.com](mailto:fa-info@de.pepperl-fuchs.com)

#### Zentrale weltweit

Pepperl+Fuchs GmbH · Königsberger Allee 87  
68307 Mannheim · Deutschland  
Tel. 0621 776-0 · Fax 0621 776-1000  
E-Mail: [info@de.pepperl-fuchs.com](mailto:info@de.pepperl-fuchs.com)

#### Zentrale USA

Pepperl+Fuchs Inc. · 1600 Enterprise Parkway  
Twinsburg, Ohio 44087 · USA  
Tel. +1 330 4253555 · Fax +1 330 4254607  
E-Mail: [sales@us.pepperl-fuchs.com](mailto:sales@us.pepperl-fuchs.com)

#### Zentrale Asien

Pepperl+Fuchs Pte Ltd. · P+F Building  
18 Ayer Rajah Crescent · Singapore 139942  
Tel. +65 67799091 · Fax +65 68731637  
E-Mail: [sales@sg.pepperl-fuchs.com](mailto:sales@sg.pepperl-fuchs.com)

 **PEPPERL+FUCHS**  
SIGNALS FÜR DIE WELT DER AUTOMATION