

# Siemens S7-200-Treiber

© 2021 PTC Inc. Alle Rechte vorbehalten.

# Inhaltsverzeichnis

<b>Siemens S7-200-Treiber</b> .....	<b>1</b>
<b>Inhaltsverzeichnis</b> .....	<b>2</b>
Siemens S7-200-Treiber .....	3
Übersicht .....	3
<b>Setup</b> .....	<b>4</b>
Kanaleigenschaften - Allgemein .....	5
Tag-Zähler .....	5
Kanaleigenschaften - Serielle Kommunikation .....	6
Kanaleigenschaften - Schreiboptimierungen .....	8
Kanaleigenschaften - Erweitert .....	9
Kanaleigenschaften - Kommunikationsserialisierung .....	10
Kanaleigenschaften - Siemens Client-ID .....	11
Geräteeigenschaften - Allgemein .....	11
Geräteeigenschaften - Scan-Modus .....	12
Geräteeigenschaften - Zeitvorgabe .....	13
Geräteeigenschaften - Automatische Herabstufung .....	14
Geräteeigenschaften - Redundanz .....	15
<b>Datentypbeschreibung</b> .....	<b>16</b>
<b>Adressbeschreibungen</b> .....	<b>17</b>
S7-200-Adressierung .....	17
S7-200-PPM-Adressierung .....	19
<b>Ereignisprotokollmeldungen</b> .....	<b>23</b>
Der Block weist möglicherweise Adressen außerhalb des zulässigen Bereichs auf.   Block- startadresse = '<Adresse>', Blockgröße = <Anzahl> (Byte). .....	23
Fehlermaskendefinitionen .....	23
<b>Index</b> .....	<b>24</b>

## Siemens S7-200-Treiber

Hilfe-Version [1.042](#)

### INHALT

#### [Übersicht](#)

Was ist Siemens S7-200-Treiber?

#### [Setup](#)

Wie konfiguriere ich ein Gerät für die Verwendung mit diesem Treiber?

#### [Datentypbeschreibung](#)

Welche Datentypen unterstützt dieser Treiber?

#### [Adressbeschreibungen](#)

Wie adressiere ich einen Datenspeicherort auf einem S7-200-Gerät von Siemens?

#### [Ereignisprotokollmeldungen](#)

Welche Meldungen können bei Siemens S7-200-Treiber auftreten?

### Übersicht

---

Siemens S7-200-Treiber bietet eine zuverlässige Möglichkeit, S7-200-Geräte von Siemens mit OPC-Client-Anwendungen, u.a. HMI, SCADA, Historian, MES, ERP und zahlreichen benutzerdefinierten Anwendungen, zu verbinden. Es ist für eine Verwendung mit S7-200-Geräten von Siemens vorgesehen und unterstützt eine 10- oder 11-Bit-Einstellung für das PPI-Programmierkabel. Bei Nutzung des 10-Bit-Modus (insbesondere für das EM 241-Modemmodul) sollte der PPM-Modus von S7-200 ausgewählt werden. Bei Nutzung des 11-Bit-Modus sollte das S7-200-Modell ausgewählt werden.

## Setup

---

### Kanal- und Gerätegrenzwerte

Die von diesem Treiber unterstützte maximale Anzahl von Kanälen liegt bei 256. Die maximale Anzahl von Geräten, die von diesem Treiber unterstützt werden, liegt bei 127 pro Kanal.

Unterstützte Kommunikationsparameter

Baudrate: 9600 oder 19200

Parität: Gerade (11-Bit-Modus) und Keine (10-Bit-PPM-Modus)

Daten-Bits: 8

Stopp-Bits: 1

● **Hinweis:** Nicht alle Geräte unterstützen die aufgelisteten Konfigurationen.

### Ethernet-Kapselung

Dieser Treiber unterstützt Ethernet-Kapselung, wodurch er mit seriellen Geräten kommunizieren kann, die über einen Terminal- oder Geräteserver mit einem Ethernet-Netzwerk verbunden sind. Sie kann in den Kanaleigenschaften über die COM-ID aufgerufen werden. *Weitere Informationen dazu finden Sie in der Hilfedokumentation zum Hauptserver.*

### Kommunikationsprotokolle

Kommunikationsprotokoll für PPI (Point-to-Point) S7-200 (11-Bit-Modus).

Kommunikationsprotokoll für PPM (Point-to-Point Modem) S7-200 (10-Bit-Modus).

Siemens S7-200-Treiber verwendet normalerweise das 11-Bit-PPI-Standardprotokoll. Wenn das EM 241-Modemmodul erforderlich ist, muss das S7-200-PPM-Modell ausgewählt werden. Durch dieses Modell kann der Treiber im 10-Bit-Modus verwendet werden, der mit vielen handelsüblichen Modems kompatibel ist. Der 10-Bit-PPM-Modus kann auch direkt am Programmier-Port der SPS verwendet werden. Um den 10-Bit-PPM-Modus zu aktivieren, legen Sie für das S7-200-Programmierskabel den 10-Bit-Modus fest.

● **Siehe auch:** [Geräteeigenschaften](#)

### Flusssteuerung

Bei Verwendung eines RS232/RS485-Konverters hängt die Art der erforderlichen Flusssteuerung von den Anforderungen des Konverters ab. Einige Konverter benötigen keine Flusssteuerung, bei anderen ist RTS-Flusssteuerung erforderlich. Der Dokumentation zum Konverter entnehmen Sie seine Flussanforderungen. Ein RS485-Konverter, der automatische Flusssteuerung bietet, ist empfehlenswert.

● **Hinweis:** Wird das mitgelieferte Kommunikationskabel des Herstellers verwendet, ist es manchmal erforderlich, in den Kanaleigenschaften eine Flusssteuerungseinstellung **RTS** oder **RTS immer** zu wählen.

● **Siehe auch:** [Kanal-Kommunikationseigenschaften](#)

### Unterstützte Geräte

Siemens-S7-200-Geräte

### Unterstützte Kabel

Ein spezielles Kabel ist für die Kommunikation mit der S7-200-SPS erforderlich. Das vom Hersteller empfohlene Kabel sollte verwendet werden.

## Kanaleigenschaften - Allgemein

Dieser Server unterstützt die Verwendung von mehreren gleichzeitigen Kommunikationstreibern. Jedes Protokoll oder jeder Treiber, das/der in einem Serverprojekt verwendet wird, wird als Kanal bezeichnet. Ein Serverprojekt besteht unter Umständen aus vielen Kanälen mit demselben Kommunikationstreiber oder mit eindeutigen Kommunikationstreibern. Ein Kanal fungiert als grundlegender Baustein eines OPC-Links. Diese Gruppe wird verwendet, um allgemeine Kanaleigenschaften (wie z.B. die ID-Attribute und den Betriebsmodus) anzugeben.

Eigenschaftengruppen	Allgemein	
	Serielle Kommunikation	
	Schreiboptimierungen	
	Erweitert	
	Kommunikationsserialisierung	
	ID	
	Name	Channel 1
	Beschreibung	
	Treiber	
	Diagnose	
Diagnoseerfassung	Deaktivieren	

### Identifikation

**Name:** Geben Sie die benutzerdefinierte ID dieses Kanals an. Bei jedem Serverprojekt muss jeder Kanalname eindeutig sein. Zwar können Namen bis zu 256 Zeichen lang sein, doch haben einige Client-Anwendungen beim Durchsuchen des Tag-Raums des OPC-Servers ein eingeschränktes Anzeigefenster. Der Kanalname ist ein Teil der OPC-Browserinformationen. Die Eigenschaft ist erforderlich, um einen Kanal zu erstellen.

• Informationen über reservierte Zeichen finden Sie in der Serverhilfe unter „So benennen Sie Kanäle, Geräte, Tags und Tag-Gruppen richtig“.

**Beschreibung:** Geben Sie benutzerdefinierte Informationen über diesen Kanal an.

• Viele dieser Eigenschaften, einschließlich der Beschreibung, verfügen über ein zugeordnetes System-Tag.

**Treiber:** Geben Sie das Protokoll/den Treiber für diesen Kanal an. Diese Eigenschaft gibt den Gerätetreiber an, der während der Kanalerstellung ausgewählt wurde. Es ist eine deaktivierte Einstellung in den Kanaleigenschaften. Die Eigenschaft ist erforderlich, um einen Kanal zu erstellen.

• **Hinweis:** Beim Online-Vollzeitbetrieb des Servers können diese Eigenschaften jederzeit geändert werden. Dies schließt das Ändern des Kanalnamens ein, um zu verhindern, dass Clients Daten am Server registrieren. Wenn ein Client bereits ein Element vom Server abgerufen hat, bevor der Kanalname geändert wurde, sind die Elemente davon nicht beeinflusst. Wenn die Client-Anwendung das Element nach der Änderung des Kanalnamens freigibt und versucht, es mit dem alten Kanalnamen erneut abzurufen, wird das Element nicht akzeptiert. Es sollten keine Änderungen an den Eigenschaften erfolgen, sobald eine große Client-Anwendung entwickelt wurde. Verwenden Sie die richtige Benutzerrollen- und Berechtigungsverwaltung, um zu verhindern, dass Operatoren Eigenschaften ändern oder auf Serverfunktionen zugreifen.

### Diagnose

**Diagnoseerfassung:** Bei Aktivierung dieser Option stehen die Diagnoseinformationen des Kanals für OPC-Anwendungen zur Verfügung. Da für die Diagnosefunktionen des Servers eine minimale Mehraufwandsverarbeitung erforderlich ist, wird empfohlen, dass sie bei Bedarf verwendet werden und ansonsten deaktiviert sind. Die Standardeinstellung ist deaktiviert.

• **Hinweise:** Diese Eigenschaft ist nicht verfügbar, wenn der Treiber Diagnosen nicht unterstützt.

• Weitere Informationen finden Sie unter "Kommunikationsdiagnosen" und "Statistik-Tags" in der Serverhilfe.

### Tag-Zähler

**Statische Tags:** Gibt die Gesamtanzahl der definierten statischen Tags auf dieser Ebene (Gerät oder Kanal) an. Diese Informationen können bei der Problembehandlung und beim Lastenausgleich hilfreich sein.

## Kanaleigenschaften - Serielle Kommunikation

Eigenschaften für serielle Kommunikation stehen seriellen Treibern zur Verfügung und sind je nach Treiber, Verbindungstyp und ausgewählten Optionen unterschiedlich. Unten finden Sie eine Übermenge der möglichen Eigenschaften.

Klicken Sie auf, um zu einem der Abschnitte zu springen: [Verbindungstyp](#), [Serielle Port-Einstellungen](#) oder [Ethernet-Einstellungen](#) und [Betriebsverhalten](#).

● **Hinweis:** Während des Online-Vollzeitbetriebs des Servers können diese Eigenschaften jederzeit geändert werden. Verwenden Sie die richtige Benutzerrollen- und Berechtigungsverwaltung, um zu verhindern, dass Operatoren Eigenschaften ändern oder auf Serverfunktionen zugreifen.

Eigenschaftengruppen	<input type="checkbox"/> <b>Verbindungstyp</b>	
Allgemein	Physisches Medium	COM-Port
<b>Serielle Kommunikation</b>	Gemeinsam genutzt	Nein
Schreiboptimierungen	<input type="checkbox"/> <b>Serielle Port-Einstellungen</b>	
Erweitert	COM-ID	3
Kommunikationsserialisierung	Baudrate	19200
Verknüpfungseinstellungen	Daten-Bits	8
	Parität	Keine
	Stopp-Bits	1
	Flusssteuerung	Keine
	<input type="checkbox"/> <b>Betriebsverhalten</b>	
	Bericht Kommunikationsfehler	Aktivieren

### Verbindungstyp

**Physisches Medium:** Wählen Sie den Hardware-Gerätetyp für Datenkommunikation. Zu den Optionen gehören Modem, Ethernet-Kapselung, COM-Port und Keine. Die Standardeinstellung ist COM-Port.

1. **Keine:** Wählen Sie "Keine" aus, um anzugeben, dass keine physische Verbindung vorhanden ist. Dadurch wird der Abschnitt [Operation ohne Kommunikation](#) angezeigt.
2. **COM-Port:** Wählen Sie "COM-Port" aus, um den Abschnitt [Serielle Port-Einstellungen](#) anzuzeigen und zu konfigurieren.
3. **Modem:** Wählen Sie "Modem" aus, wenn für die Kommunikation Telefonleitungen verwendet werden. Dies wird im Abschnitt [Modemeinstellungen](#) konfiguriert.
4. **Ethernet-Kapselung:** Wählen Sie diese Option aus, wenn für die Kommunikation Ethernet-Kapselung verwendet wird. Dadurch wird der Abschnitt [Ethernet-Einstellungen](#) angezeigt.
5. **Gemeinsam genutzt:** Überprüfen Sie, ob für die Verbindung korrekt angegeben ist, dass die aktuelle Konfiguration mit einem anderen Kanal gemeinsam genutzt wird. Dies ist eine schreibgeschützte Eigenschaft.

### Serielle Port-Einstellungen

**COM-ID:** Gibt die zu verwendende Kommunikations-ID bei der Kommunikation mit dem Kanal zugewiesenen Geräten an. Der gültige Bereich ist 1 bis 9991 bis 16. Die Standardeinstellung ist 1.

**Baudrate:** Geben Sie die Baudrate an, die zur Konfiguration des ausgewählten Kommunikationsports verwendet werden soll.


**Daten-Bits:** Geben Sie die Anzahl der Daten-Bits pro Datenwort an. Zu den Optionen gehören 5, 6, 7 oder 8.

**Parität:** Geben Sie den Paritätstyp für die Daten an. Zu den Optionen gehören "Ungerade", "Gerade" oder "Keine".

**Stopp-Bits:** Geben Sie die Anzahl der Stopp-Bits pro Datenwort an. Zu den Optionen gehören 1 oder 2.

**Flusssteuerung:** Wählen Sie aus, wie die RTS- und DTR-Steuereleitungen verwendet werden. Flusssteuerung ist für die Kommunikation mit einigen seriellen Geräten erforderlich. Es gibt folgende Optionen:


- **Keine:** Mit dieser Option werden keine Steuerleitungen umgeschaltet oder in den aktiven Zustand gebracht.
- **DTR:** Mit dieser Option wird die DTR-Leitung in den aktiven Zustand gebracht, wenn der Kommunikationsport geöffnet ist und es auch bleibt.
- **RTS:** Mit dieser Option wird angegeben, dass die RTS-Leitung hoch ist, wenn Byte für die Übertragung zur Verfügung stehen. Nachdem alle gepufferten Byte gesendet wurden, ist die RTS-Leitung niedrig. Dies wird normalerweise mit der RS232/RS485-Konverter-Hardware verwendet.
- **RTS, DTR:** Diese Option ist eine Kombination aus DTR und RTS.
- **RTS immer:** Mit dieser Option wird die RTS-Leitung in den aktiven Zustand gebracht, wenn der Kommunikationsport geöffnet ist und es auch bleibt.
- **RTS manuell:** Mit dieser Option wird die RTS-Leitung basierend auf den für RTS-Leitungssteuerung eingegebenen Zeitvorgaben-Eigenschaften in den aktiven Zustand gebracht. Sie steht nur zur Verfügung, wenn der Treiber manuelle RTS-Leitungssteuerung unterstützt (oder wenn die Eigenschaften gemeinsam benutzt werden und mindestens einer der Kanäle zu einem Treiber gehört, der diese Unterstützung bereitstellt). Durch "RTS manuell" wird die Eigenschaft **RTS-Leitungssteuerung** mit den folgenden Optionen hinzugefügt:
  - **Anstieg:** Diese Eigenschaft gibt an, wie lang die RTS-Leitung vor der Datenübertragung ansteigt. Der gültige Bereich liegt zwischen 0 und 9999 Millisekunden. Die Standardeinstellung ist 10 Millisekunden.
  - **Abfall:** Diese Eigenschaft gibt die Zeitdauer an, während der die RTS-Leitung nach der Datenübertragung hoch bleibt. Der gültige Bereich liegt zwischen 0 und 9999 Millisekunden. Die Standardeinstellung ist 10 Millisekunden.
  - **Abrufverzögerung:** Diese Eigenschaft gibt die Zeit an, um die der Abruf für die Kommunikation verzögert ist. Der gültige Bereich liegt zwischen 0 und 9999. Die Standardeinstellung ist 10 Millisekunden.

 **Tip:** Bei Verwendung von doppeladrigen RS-485-Kabeln können "Echos" in den Kommunikationsleitungen auftreten. Da diese Kommunikation keine Echounterdrückung unterstützt, wird empfohlen, Echos zu deaktivieren oder einen RS-485-Konverter zu verwenden.

## Betriebsverhalten

- **Kommunikationsfehler melden:** Aktivieren oder deaktivieren Sie die Berichterstattung über geringfügige Kommunikationsfehler. Wenn diese Option aktiviert ist, werden geringfügige Fehler beim Auftreten im Ereignisprotokoll angezeigt. Wenn diese Option deaktiviert ist, werden dieselben Fehler nicht angezeigt, selbst wenn es normale Anforderungsfehler sind. Die Standardeinstellung ist "Aktivieren".
- **Inaktive Verbindung schließen:** Wählen Sie diese Option, um die Verbindung zu schließen, wenn es keinerlei Tags mehr gibt, die von einem Client im Kanal referenziert werden. Die Standardeinstellung ist "Aktivieren".
- **Inaktivitätsdauer bis Schließen:** Geben Sie an, wie lang der Server warten soll, bis alle Tags vor dem Schließen des COM-Ports entfernt wurden. Der Standardwert ist 15 Sekunden.

## Ethernet-Einstellungen

 **Hinweis:** Nicht alle seriellen Treiber unterstützen Ethernet-Kapselung. Wird diese Gruppe nicht angezeigt, wird die Funktion nicht unterstützt.

Ethernet-Kapselung ermöglicht die Kommunikation mit seriellen Geräten, die im Ethernet-Netzwerk mit Terminalservern verbunden sind. Ein Terminalserver ist im Wesentlichen ein virtueller serieller Port, der TCP/IP-Meldungen im Ethernet-Netzwerk in serielle Daten konvertiert. Sobald die Meldung konvertiert wurde, können Benutzer Standardgeräte verbinden, die eine serielle Kommunikation mit dem Terminalserver unterstützen. Der serielle Port des Terminalservers muss richtig konfiguriert werden, um den Anforderungen des seriellen Geräts zu entsprechen, mit dem er verbunden ist. *Weitere Informationen dazu finden Sie in der Serverhilfe unter "Ethernet-Kapselung verwenden".*

- **Netzwerkadapter:** Geben Sie für Ethernet-Geräte in diesem Kanal einen zu bindenden Netzwerkadapter an. Wählen Sie einen Netzwerkadapter für die Bindung, oder lassen Sie die Standardeinstellung vom Betriebssystem auswählen.

Bestimmte Treiber zeigen unter Umständen zusätzliche Eigenschaften für Ethernet-Kapselung an. Weitere Informationen dazu finden Sie unter [Kanaleigenschaften - Ethernet-Kapselung](#).

## Modemeinstellungen

- **Modem:** Geben Sie das installierte Modem an, das für die Kommunikation verwendet werden soll.
- **Verbindungs-Timeout:** Diese Eigenschaft gibt an, wie lang auf das Herstellen von Verbindungen gewartet werden soll, bevor ein Lese- oder Schreibvorgang fehlschlägt. Der Standardwert ist 60 Sekunden.
- **Modemeigenschaften:** Konfigurieren Sie die Modem-Hardware. Durch Klicken auf diese Schaltfläche werden händlerspezifische Modemeigenschaften geöffnet.
- **Automatisches Wählen:** Ermöglicht das automatische Wählen von Einträgen im Telefonbuch. Die Standardeinstellung ist "Deaktivieren". Weitere Informationen finden Sie unter "Modem Auto-Dial" in der Serverhilfe.
- **Kommunikationsfehler melden:** Aktivieren oder deaktivieren Sie die Berichterstellung über geringfügige Kommunikationsfehler. Wenn diese Option aktiviert ist, werden geringfügige Fehler beim Auftreten im Ereignisprotokoll angezeigt. Wenn diese Option deaktiviert ist, werden dieselben Fehler nicht angezeigt, selbst wenn es normale Anforderungsfehler sind. Die Standardeinstellung ist "Aktivieren".
- **Inaktive Verbindung schließen:** Wählen Sie diese Option, um die Modemverbindung zu schließen, wenn es keinerlei Tags mehr gibt, die von einem Client im Kanal referenziert werden. Die Standardeinstellung ist "Aktivieren".
- **Inaktivitätsdauer bis Schließen:** Geben Sie an, wie lang der Server warten soll, bis alle Tags vor dem Schließen der Modemverbindung entfernt wurden. Der Standardwert ist 15 Sekunden.

## Operation ohne Kommunikation

- **Leseverarbeitung:** Wählen Sie aus, welche Maßnahmen ergriffen werden sollen, wenn ein expliziter Gerätelesevorgang angefordert wird. Zu den Optionen gehören Ignorieren und Fehlgeschlagen. Bei Ignorieren geschieht nichts, bei Fehlgeschlagen wird das Fehlschlagen dem Client durch eine Aktualisierung angezeigt. Die Standardeinstellung ist Ignorieren.

## Kanaleigenschaften - Schreiboptimierungen

Der Server muss sicherstellen, dass die von der Client-Anwendung geschriebenen Daten rechtzeitig auf das Gerät gelangen. In Anbetracht dieses Ziels stellt der Server Optimierungseigenschaften bereit, um die jeweiligen Anforderungen zu erfüllen bzw. die Reaktionsfähigkeit der Anwendungen zu verbessern.

Eigentengruppen	<input checked="" type="checkbox"/> <b>Schreiboptimierungen</b>	
Allgemein	Optimierungsmethode	Nur den letzten Wert für alle Tags schr...
Serielle Kommunikation	Servicezyklus	10
<b>Schreiboptimierungen</b>		

## Schreiboptimierungen

**Optimierungsmethode:** Mit dieser Option wird gesteuert, wie Schreibdaten an den zugrunde liegenden Kommunikationstreiber weitergeleitet werden. Die Optionen sind:

- **Alle Werte für alle Tags schreiben:** Mit dieser Option wird der Server gezwungen, für jeden Wert einen Schreibvorgang auf dem Controller zu versuchen. In diesem Modus sammelt der Server weiterhin Schreibenanforderungen und fügt sie der internen Schreibwarteschlange des Servers hinzu. Der Server verarbeitet die Schreibwarteschlange und versucht, sie zu leeren, indem er so schnell wie möglich Daten auf das Gerät schreibt. In diesem Modus wird sichergestellt, dass alles, was von den Client-Anwendungen geschrieben wird, an das Zielgerät gesendet wird. Dieser Modus sollte ausgewählt werden, wenn die Reihenfolge des Schreibvorgangs oder der Inhalt des Schreibelements eindeutig auf dem Zielgerät zu finden sein muss.
- **Nur den letzten Wert für nicht boolesche Tags schreiben:** Viele aufeinander folgende Schreibvorgänge für denselben Wert können sich aufgrund der Zeit, die tatsächlich zum Senden der Daten auf das Gerät erforderlich ist, in der Schreibwarteschlange ansammeln. Wenn der Server einen Schreibwert



aktualisiert, der bereits in die Schreibwarteschlange eingefügt wurde, sind weitaus weniger Schreibvorgänge erforderlich, um denselben Endausgabewert zu erhalten. Auf diese Weise sammeln sich keine zusätzlichen Schreibvorgänge in der Warteschlange des Servers an. Wenn der Benutzer den Schieberegler nicht mehr verschiebt, erreicht der Wert im Gerät praktisch in derselben Zeit den richtigen Wert. Dem Modus entsprechend wird jeder Wert, der kein boolescher Wert ist, in der internen Warteschlange des Servers aktualisiert und bei der nächstmöglichen Gelegenheit an das Gerät gesendet. Dies kann die Anwendungsleistung erheblich verbessern.

- **Hinweis:** Mit dieser Option wird nicht versucht, Schreibvorgänge in Boolesche Werte zu optimieren. Dadurch können Benutzer den HMI-Datenvorgang optimieren, ohne Probleme mit Booleschen Operationen (z.B. eine vorübergehende Schaltfläche) zu verursachen.
- **Nur den letzten Wert für alle Tags schreiben:** Mit dieser Option wird die hinter der zweiten Optimierungsmethode stehende Theorie auf alle Tags angewendet. Sie ist besonders nützlich, wenn die Anwendung nur den letzten Wert an das Gerät senden muss. In diesem Modus werden alle Schreibvorgänge optimiert, indem die derzeit in der Schreibwarteschlange befindlichen Tags vor dem Senden aktualisiert werden. Dies ist der Standardmodus.

**Servicezyklus:** Wird verwendet, um das Verhältnis von Schreib- und Lesevorgängen zu steuern. Das Verhältnis basiert immer auf einem Lesevorgang für jeden zehnten Schreibvorgang. Für den Servicezyklus wird standardmäßig 10 festgelegt. Dies bedeutet, dass 10 Schreibvorgänge für jeden Lesevorgang erfolgen. Zwar führt die Anwendung eine große Anzahl fortlaufender Schreibvorgänge durch, doch muss sichergestellt werden, dass es für Lesedaten weiterhin Verarbeitungszeit gibt. Die Einstellung 1 hat zur Folge, dass ein Lesevorgang für jeden Schreibvorgang erfolgt. Wenn es keine durchzuführenden Schreibvorgänge gibt, werden Lesevorgänge fortlaufend verarbeitet. Dies ermöglicht eine Optimierung für Anwendungen mit fortlaufenden Schreibvorgängen gegenüber einem ausbalancierteren Datenzufluss und -abfluss.

● **Hinweis:** Es wird empfohlen, dass für die Anwendung die Kompatibilität mit den Verbesserungen zur Schreiboptimierung charakteristisch ist, bevor sie in einer Produktionsumgebung verwendet wird.

## Kanaleigenschaften - Erweitert

Diese Gruppe wird verwendet, um erweiterte Kanaleigenschaften anzugeben. Nicht alle Treiber unterstützen alle Eigenschaften; so wird die Gruppe "Erweitert" für jene Geräte nicht angezeigt.

Eigenschaftengruppen	<input type="checkbox"/> <b>Nicht normalisierte Float-Handhabung</b>	
Allgemein	Gleitkommawerte	Durch Null ersetzen
Serielle Kommunikation	<input type="checkbox"/> <b>Verzögerung zwischen Geräten</b>	
Schreiboptimierungen	Verzögerung zwischen Geräten...	0
<b>Erweitert</b>		
Kommunikationsserialisierung		

**Nicht normalisierte Float-Handhabung:** Ein nicht normalisierter Wert wird als "Unendlich", "Nichtzahlenwert (NaN)" oder als "Denormalisierte Zahl" definiert. Die Standardeinstellung ist Durch Null ersetzen. Für Treiber, die eine native Float-Handhabung aufweisen, wird standardmäßig unter Umständen "Nicht geändert" verwendet. Durch Behandlung nicht normalisierter Gleitkommazahlen können Benutzer festlegen, wie ein Treiber mit nicht normalisierten IEEE-754-Gleitkommandaten umgeht. Es folgen Beschreibungen der Optionen:

- **Durch Null ersetzen:** Diese Option ermöglicht es einem Treiber, nicht normalisierte IEEE-754-Gleitkommawerte durch Null zu ersetzen, bevor sie an Clients übertragen werden.
- **Nicht geändert:** Diese Option ermöglicht es einem Treiber, denormalisierte, normalisierte IEEE-754-Nichtzahlenwerte und unendliche IEEE-754-Werte ohne jegliche Konvertierung oder Änderungen an Clients zu übertragen.

● **Hinweis:** Diese Eigenschaft ist deaktiviert, wenn der Treiber keine Gleitkommawerte unterstützt, oder wenn er nur die angezeigte Option unterstützt. Gemäß der Float-Normalisierungseinstellung des Kanals unterliegen nur Echtzeit-Treiber-Tags (wie z.B. Werte und Arrays) der Float-Normalisierung. Beispielsweise werden EFM-Daten nicht durch diese Einstellung beeinflusst.

● **Weitere Informationen über die Gleitkommawerte finden Sie unter "Mit nicht normalisierten Gleitkommawerten arbeiten" in der Serverhilfe.**

**Verzögerung zwischen Geräten:** Geben Sie die Zeitdauer an, in der der Kommunikationskanal das Senden einer Anforderung an das nächste Gerät verzögert, nachdem Daten vom aktuellen Gerät in demselben Kanal empfangen wurden. Null (0) deaktiviert die Verzögerung.

● **Hinweis:** Diese Eigenschaft ist nicht für alle Treiber, Modelle und abhängige Einstellungen verfügbar.

## Kanaleigenschaften - Kommunikationsserialisierung

Die Multithreading-Architektur des Servers ermöglicht Kanälen die parallele Kommunikation mit Geräten. Zwar ist das effizient, doch kann die Kommunikation in Fällen mit physischen Netzwerkeinschränkungen (wie Ethernet-Funksignale) serialisiert werden. Kommunikationsserialisierung schränkt die Kommunikation auf einen Kanal gleichzeitig innerhalb eines virtuellen Netzwerks ein.

Der Begriff "virtuelles Netzwerk" beschreibt eine Sammlung von Kanälen und zugeordneten Geräten, die dieselbe Pipeline für die Kommunikation verwenden. Beispielsweise ist die Pipeline eines Ethernet-Radios das Client-Radio. Alle Kanäle mit demselben Client-Radio werden demselben virtuellen Netzwerk zugeordnet. Kanäle dürfen jeweils nacheinander im Roundrobin-Verfahren kommunizieren. Standardmäßig kann ein Kanal eine Transaktion verarbeiten, bevor die Kommunikation an einen anderen Kanal übergeben wird. Eine Transaktion kann einen oder mehrere Tags einschließen. Wenn der steuernde Kanal ein Gerät enthält, das nicht auf eine Anfrage antwortet, kann der Kanal die Steuerung erst bis zum Timeout der Transaktion freigeben. Dies hat Datenaktualisierungsverzögerungen für die anderen Kanäle im virtuellen Netzwerk zur Folge.

Eigenschaftengruppen	[-] <b>Einstellungen auf Kanalebene</b>	
Allgemein	Virtuelles Netzwerk	Keine
Serielle Kommunikation	Transaktionen pro Zyklus	1
Schreiboptimierungen	[-] <b>Globale Einstellungen</b>	
Erweitert	Netzwerkmodus	Lastausgleich
<b>Kommunikationsserialisier...</b>		

### Einstellungen auf Kanalebene

**Virtuelles Netzwerk:** Geben Sie den Kanalmodus der Kommunikationsserialisierung an. Zu den Optionen gehören "Keine" sowie "Netzwerk 1 - Netzwerk 500". Die Standardeinstellung ist "Keine". Es folgen Beschreibungen der Optionen:

- **Keine:** Mit dieser Option wird die Kommunikationsserialisierung für den Kanal deaktiviert.
- **Netzwerk 1 - Netzwerk 500:** Mit dieser Option wird das virtuelle Netzwerk angegeben, dem der Kanal zugewiesen wird.

**Transaktionen pro Zyklus:** Geben Sie die Anzahl einzelner blockierter/nicht blockierter Lese-/Schreibtransaktionen an, die auf dem Kanal vorkommen können. Wird einem Kanal die Möglichkeit zur Kommunikation gegeben, wird versucht, diese Anzahl von Transaktionen auszuführen. Der gültige Bereich liegt zwischen 1 und 99. Die Standardeinstellung ist 1.

### Globale Einstellungen

**Netzwerkmodus:** Mit dieser Eigenschaft wird gesteuert, wie die Kanalkommunikation delegiert wird. Im Modus **Lastausgleich** wird jedem Kanal die Möglichkeit gegeben, nacheinander zu kommunizieren. Im Modus **Priorität** wird Kanälen die Möglichkeit gegeben, nach den folgenden Regeln (von der höchsten zur niedrigsten Priorität) zu kommunizieren:

1. Kanäle mit ausstehenden Schreibvorgängen haben den höchsten Vorrang.
2. Kanäle mit ausstehenden expliziten Lesevorgängen (durch interne Plugins oder externe Client-Schnittstellen) werden je nach Priorität des Lesevorgangs priorisiert.
3. Gescannte Lesevorgänge und andere periodische Ereignisse (treiberspezifisch).

Die Standardeinstellung ist "Lastausgleich" und wirkt sich auf *alle* virtuellen Netzwerke und Kanäle aus.

● Geräte, die sich auf unangeforderte Antworten verlassen, sollten nicht in ein virtuelles Netzwerk eingefügt werden. In Situationen, wo die Kommunikationen serialisiert werden muss, wird empfohlen, dass "Automatische Herabstufung" aktiviert wird.

Aufgrund von Unterschieden in der Art und Weise, wie Treiber Daten lesen und schreiben (wie z.B. einzelne blockierte oder nicht blockierte Transaktionen) muss die Eigenschaft "Transaktionen pro Zyklus" der Anwendung möglicherweise angepasst werden. Berücksichtigen Sie dabei die folgenden Faktoren:

- Wie viele Tags müssen von jedem Kanal gelesen werden?
- Wie oft werden Daten in jeden Kanal geschrieben?
- Verwendet der Kanal einen seriellen oder einen Ethernet-Treiber?
- Liest der Treiber Tags in separaten Anfragen, oder werden mehrere Tags in einem Block gelesen?
- Wurden die Zeitvorgabe-Eigenschaften des Geräts (wie z.B. Anforderungs-Timeout und Fehlgeschlagen nach x aufeinander folgenden Timeouts) für das Kommunikationsmedium des virtuellen Netzwerks optimiert?

## Kanaleigenschaften - Siemens Client-ID

Property Groups General Serial Communications Write Optimizations Advanced Communication Serialization <b>Siemens Client ID</b>	<table border="1"> <tr> <td colspan="2"><b>Siemens Client ID</b></td> </tr> <tr> <td>Siemens Client ID</td> <td>0</td> </tr> </table>	<b>Siemens Client ID</b>		Siemens Client ID	0
<b>Siemens Client ID</b>					
Siemens Client ID	0				

Siemens-Client-ID Siemens S7-200-Treiber: Geben Sie die von im Netzwerk verwendete Knotennummer an. Jeder Kanal muss eine eindeutige Siemens-Client-ID aufweisen. Der gültige Bereich liegt zwischen 0 und 126.

## Geräteeigenschaften - Allgemein

<b>Identifikation</b>	
Name	S7-200
Beschreibung	
Treiber	Siemens S7-200
Modell	S7-200
Kanalzuweisung	Siemens S7-200
ID-Format	Dezimal
ID	1
<b>Betriebsmodus</b>	
Datensammlung	Aktivieren
Simuliert	Nein
<b>Tag-Zähler</b>	
Statische Tags	1
<b>Name</b> Geben Sie die Identität dieses Objekts an.	

### Identifikation

**Name:** Benutzerdefinierte ID dieses Geräts.

**Beschreibung:** Benutzerdefinierte Informationen über dieses Gerät.

**Kanalzuweisung:** Benutzerdefinierter Name des Kanals, zu dem dieses Gerät derzeit gehört.

**Treiber:** Ausgewählter Protokolltreiber für dieses Gerät.

**Modell:** Wählen Sie die jeweilige Version des Geräts aus.

**ID-Format:** Wählen Sie aus, wie die Geräte-ID formatiert wird. Zu den Optionen zählen Formatierungen als Dezimal-, Oktal- oder Hexadezimalzahl.

**ID:** Die eindeutige ID des Geräts für die Kommunikation mit dem Treiber. Der gültige Bereich liegt zwischen 0 und 126. Alle unter diesem Kanal definierten Geräte sollten keine ID verwenden, die im Konflikt mit der Siemens-Client-ID steht.

## Betriebsmodus

**Datensammlung:** Diese Eigenschaft steuert den aktiven Status des Geräts. Zwar sind Gerätekommunikationen standardmäßig aktiviert, doch kann diese Eigenschaft verwendet werden, um ein physisches Gerät zu deaktivieren. Kommunikationen werden nicht versucht, wenn ein Gerät deaktiviert ist. Vom Standpunkt eines Clients werden die Daten als ungültig markiert und Schreibvorgänge werden nicht akzeptiert. Diese Eigenschaft kann jederzeit durch diese Eigenschaft oder die System-Tags des Geräts geändert werden.

**Simuliert:** Diese Option versetzt das Gerät in den Simulationsmodus. In diesem Modus versucht der Treiber nicht, mit dem physischen Gerät zu kommunizieren, aber der Server gibt weiterhin gültige OPC-Daten zurück. Durch Auswählen von "Simuliert" wird die physische Kommunikation mit dem Gerät angehalten, OPC-Daten können jedoch als gültige Daten dem OPC-Client zurückgegeben werden. Im Simulationsmodus behandelt der Server alle Gerätedaten als reflektierend: was auch immer in das simulierte Gerät geschrieben wird, wird zurückgelesen, und jedes OPC-Element wird einzeln behandelt. Die Speicherzuordnung des Elements basiert auf der Gruppenaktualisierungsrate. Die Daten werden nicht gespeichert, wenn der Server das Element entfernt (z.B., wenn der Server neu initialisiert wird). Die Standardeinstellung ist "Nein".

### Hinweise:

1. Dieses System-Tag (\_ Simulated) ist schreibgeschützt und kann für den Laufzeitschutz nicht geschrieben werden. Das System-Tag ermöglicht es, dass diese Eigenschaft vom Client überwacht wird.
2. Im Simulationsmodus basiert die Speicherzuordnung des Elements auf Client-Aktualisierungsraten (Gruppenaktualisierungsrate für OPC-Clients oder Scan-Intervall für native und DDE-Schnittstellen). Das bedeutet, dass zwei Clients, die dasselbe Element mit unterschiedlichen Aktualisierungsraten referenzieren, verschiedene Daten zurückgeben.

Der Simulationsmodus ist nur für Test- und Simulationszwecke. Es sollte niemals in einer Produktionsumgebung nie verwendet werden.

## Geräteeigenschaften - Scan-Modus

Der Scan-Modus gibt das vom abonnierten Client angeforderte Scan-Intervall für Tags an, die Gerätekommunikation erfordern. Synchrone und asynchrone Lese- und Schreibvorgänge des Geräts werden so bald wie möglich verarbeitet; unbeeinflusst von den Eigenschaften für den Scan-Modus.

Eigenschaftengruppen	<input type="checkbox"/> <b>Scan-Modus</b>	
Allgemein	Scan-Modus	Vom Client angegebenes Scan-Intervall...
<b>Scan-Modus</b>	Anfangsaktualisierungen aus ...	Deaktivieren

**Scan-Modus:** Geben Sie an, wie Tags im Gerät für an abonnierende Clients gesendete Aktualisierungen gesannt werden. Es folgen Beschreibungen der Optionen:

- **Vom Client angegebenes Scan-Intervall berücksichtigen:** Dieser Modus verwendet das vom Client angeforderte Scan-Intervall.

- **Datenanfrage nicht schneller als Scan-Intervall:** Dieser Modus gibt den Wert an, der als maximales Scan-Intervall festgelegt wurde. Der gültige Bereich liegt zwischen 10 und 99999990 Millisekunden. Die Standardeinstellung ist 1000 Millisekunden.
  - **Hinweis:** Wenn der Server über einen aktiven Client und Elemente für das Gerät verfügt und der Wert für das Scan-Intervall erhöht wird, werden die Änderungen sofort wirksam. Wenn der Wert für das Scan-Intervall verringert wird, werden die Änderungen erst wirksam, wenn alle Client-Anwendungen getrennt wurden.
- **Alle Datenanfragen im Scan-Intervall:** Dieser Modus erzwingt, dass Tags im angegebenen Intervall nach abonnierten Clients gescannt werden. Der gültige Bereich liegt zwischen 10 und 99999990 Millisekunden. Die Standardeinstellung ist 1000 Millisekunden.
- **Nicht scannen, nur Abruf anfordern:** In diesem Modus werden Tags, die zum Gerät gehören, nicht periodisch abgerufen, und es wird auch kein Lesevorgang durchgeführt, um den Anfangswert eines Elements abzurufen, sobald es aktiv wird. Es liegt in der Verantwortung des OPC-Clients, Aktualisierungen abzurufen, entweder durch Schreiben in das \_DemandPoll-Tag oder durch Ausgeben expliziter Lesevorgänge des Geräts für einzelne Elemente. *Weitere Informationen finden Sie unter "Geräte-Bedarfsabruf" in der Serverhilfe.*
- **Durch Tag angegebenes Scan-Intervall berücksichtigen:** Dieser Modus erzwingt das Scannen statischer Tags im Intervall, das in ihrer statischen Konfiguration Tag-Eigenschaften angegeben wurde. Dynamische Tags werden in dem vom Client angegebenen Scan-Intervall gescannt.

**Anfangsaktualisierungen aus Cache:** Wenn diese Option aktiviert ist, kann der Server die ersten Aktualisierungen für neu aktivierte Tag-Referenzen aus gespeicherten (Cache-)Daten zur Verfügung stellen. Cache-Aktualisierungen können nur bereitgestellt werden, wenn die neue Elementreferenz dieselben Eigenschaften für Adresse, Scan-Intervall, Datentyp, Client-Zugriff und Skalierung gemeinsam nutzt. Ein Lesevorgang des Geräts wird nur für die Anfangsaktualisierung für die erste Client-Referenz verwendet. Der Standardeinstellung ist "Deaktiviert"; immer wenn ein Client eine Tag-Referenz aktiviert, versucht der Server, den Anfangswert vom Gerät zu lesen.

## Geräteeigenschaften - Zeitvorgabe

Mithilfe der Zeitvorgabe-Eigenschaften des Geräts kann die Antwort des Treibers auf Fehlerbedingungen so angepasst werden, dass sie den Anforderungen der Anwendung entspricht. In vielen Fällen erfordert die Umgebung für eine optimale Leistung Änderungen an diesen Eigenschaften. Faktoren wie elektrisch generiertes Rauschen, Modemverzögerungen und fehlerhafte physische Verbindungen können beeinflussen, wie viele Fehler oder Timeouts ein Kommunikationstreiber feststellt. Zeitvorgabe-Eigenschaften sind für jedes konfigurierte Gerät spezifisch.

Property Groups	☐ <b>Communication Timeouts</b>	
General	Connect Timeout (s)	3
Scan Mode	Request Timeout (ms)	1000
<b>Timing</b>	Attempts Before Timeout	3

## Kommunikations-Timeouts

**Verbindungs-Timeout:** Mit dieser Eigenschaft (die in erster Linie von Ethernet-basierten Treibern verwendet wird) wird die Zeitdauer gesteuert, die zum Herstellen einer Socket-Verbindung mit einem Remote-Gerät erforderlich ist. Die Verbindungszeit des Gerät ist häufig länger als normale Kommunikationsanforderungen mit demselben Gerät. Der gültige Bereich liegt zwischen 1 und 30 Sekunden. Die Standardeinstellung ist normalerweise 3 Sekunden, kann jedoch abhängig vom jeweiligen Treiber unterschiedlich sein. Wenn diese Einstellung nicht vom Treiber unterstützt wird, ist sie deaktiviert.

● **Hinweis:** Aufgrund der Art der UDP-Verbindungen ist die Einstellung für Verbindungs-Timeout nicht anwendbar, wenn die Kommunikation über UDP erfolgt.

**Anforderungs-Timeout:** Geben Sie ein von allen Treibern verwendetes Intervall an, um zu bestimmen, wie lange der Treiber abschließend auf eine Antwort vom Zielgerät wartet. Der gültige Bereich liegt zwischen 50 und 9.999.999 Millisekunden (167,6667 Minuten). Die Standardeinstellung ist im Allgemeinen 1000 Millisekunden, kann jedoch abhängig vom Treiber unterschiedlich sein. Das Standard-Timeout für die meisten seriellen Treiber basiert auf einer Baudrate von 9600 Baud oder besser. Wenn ein Treiber bei niedrigeren Bau-

draten verwendet wird, erhöhen Sie das Timeout, um die erhöhte Zeit auszugleichen, die zum Abrufen von Daten erforderlich ist.

**Versuche vor Timeout:** Geben Sie an, wie oft der Treiber eine Kommunikationsanforderung wiederholt, bevor er die Anforderung als fehlgeschlagen und das Gerät als fehlerhaft erachtet. Der gültige Bereich liegt zwischen 1 und 10. Die Standardeinstellung ist normalerweise 3, kann sich jedoch abhängig vom jeweiligen Treiber ändern. Die Anzahl der für eine Anwendung konfigurierten Wiederholungen hängt größtenteils von der Kommunikationsumgebung ab. Diese Eigenschaft trifft sowohl auf Verbindungsversuche als auch auf Anforderungsversuche zu.

## Zeitvorgabe

**Verzögerung zwischen Anfragen:** Geben Sie an, wie lange der Treiber wartet, bevor er die nächste Anforderung an das Zielgerät sendet. Sie setzt das dem Gerät zugewiesene normale Tag-Abfrageintervall sowie einmalige Lese- und Schreibvorgänge außer Kraft. Diese Verzögerung kann bei Geräten mit langsamen Durchlaufzeiten und in Situationen nützlich sein, in denen die Netzwerklast problematisch ist. Das Konfigurieren einer Verzögerung für ein Gerät wirkt sich auf die Kommunikation mit allen anderen Geräten im Kanal aus. Es wird empfohlen, dass Benutzer jedes Gerät trennen, das eine Verzögerung zwischen Anfragen für einen separaten Kanal erfordert (sofern möglich). Andere Kommunikationseigenschaften (wie z.B. Kommunikationsserialisierung) können diese Verzögerung verlängern. Der gültige Bereich liegt zwischen 0 und 300000 Millisekunden; jedoch können einige Treiber ggf. den maximalen Wert wegen einer Funktion ihrer spezifischen Konstruktion beschränken. Die Standardeinstellung ist 0. Dies weist darauf hin, dass es keine Verzögerung zwischen Anfragen mit dem Zielgerät gibt.

● **Hinweis:** Nicht alle Treiber unterstützen Verzögerung zwischen Anfragen. Diese Einstellung wird nicht angezeigt, wenn sie nicht zur Verfügung steht.

Property Groups	[-] <b>Timing</b>	
General	Inter-Request Delay (ms)	0
Scan Mode		
<b>Timing</b>		

## Geräteeigenschaften - Automatische Herabstufung

Die Eigenschaften für automatische Herabstufung können ein Gerät vorübergehend in den Nicht-Scan-Modus versetzen, falls das Gerät nicht antwortet. Dadurch, dass ein nicht reagierendes Gerät für einen bestimmten Zeitraum offline gestellt wird, kann der Treiber weiterhin seine Kommunikation mit anderen Geräten in demselben Kanal optimieren. Nach Ablauf dieses Zeitraums versucht der Treiber die Kommunikation mit dem nicht reagierenden Gerät erneut. Wenn das Gerät reagiert, wird es wieder zum Scannen freigegeben. Andernfalls wird sein Nicht-Scan-Zeitraum erneut gestartet.

Eigenschaftengruppen	[-] <b>Automatische Herabstufung</b>	
Allgemein	Herabstufen bei Fehler	Aktivieren
Scan-Modus	Timeout bis zum Herabstufen	3
Zeitvorgabe	Herabstufungszeitraum (ms)	10000
<b>Automatische Herabstufung</b>	Anfragen verwerfen, wenn herabgestuft	Deaktivieren

**Herabstufen bei Fehler:** Wird diese Option aktiviert, wird das Gerät automatisch in den Nicht-Scan-Modus versetzt, bis es wieder antwortet.

● **Tipp:** Ermitteln Sie, wenn sich ein Gerät im Nicht-Scan-Modus befindet, indem Sie seinen herabgestuften Status mit dem `_AutoDemoted-System-Tag` überwachen.

**Timeout bis zum Herabstufen:** Legen Sie fest, wie viele aufeinander folgende Zyklen von Anforderungs-Timeouts und Wiederholungen vorkommen, bevor das Gerät in den Nicht-Scan-Modus versetzt wird. Der gültige Bereich ist 1 bis 30 aufeinander folgende Fehlschläge. Die Standardeinstellung ist 3.

**Herabstufungszeitraum:** Gibt an, wie lange das Gerät im Nicht-Scan-Modus sein sollte, wenn der Timeout-Wert erreicht wird. Während dieses Zeitraums werden keine Leseanforderungen an das Gerät gesendet, und für alle den Leseanforderungen zugeordneten Daten wird schlechte Qualität festgelegt. Wenn dieser Zeitraum

abgelaufen ist, versetzt der Treiber das Gerät in den Scan-Modus und ermöglicht einen weiteren Kommunikationsversuch. Der gültige Bereich liegt zwischen 100 und 3600000 Millisekunden. Die Standardeinstellung ist 10000 Millisekunden.

**Anfragen verwerfen, wenn herabgestuft:** Durch Aktivieren dieser Option wird ausgewählt, ob Schreibanforderungen während des Nicht-Scan-Zeitraums versucht werden sollten. Deaktivieren Sie diese Option, damit Schreibanforderungen unabhängig vom Herabstufungszeitraum immer gesendet werden. Aktivieren Sie diese Option, um Schreibvorgänge zu verwerfen; auf dem Server schlägt jede von einem Client empfangene Schreibanforderung automatisch fehl, und es wird keine Meldung im Ereignisprotokoll angezeigt.

## Geräteeigenschaften - Redundanz

Eigenschaftengruppen	☐ <b>Redundanz</b>	
Allgemein	Pfad des Sekundärgeräts	
Scan-Modus	Betriebsmodus	Fehler beim Einschalten
Zeitvorgabe	Überwachungselement	
Redundanz	Überwachungsintervall (s)	300
	Baldmöglichste Rückkehr zum Primärgerät	Ja

Redundanz steht mit dem Plugin für Redundanz auf Medienebene zur Verfügung.

• Weitere Informationen dazu erhalten Sie auf der Website, von einem Vertriebsrepräsentanten oder im [Benutzerhandbuch](#).

## Datentypbeschreibung

Datentyp	Beschreibung
Boolean	Einzelnes Bit eines 16-Bit-Werts*
Byte	8-Bit-Wert ohne Vorzeichen  Bit 0 ist das Low-Bit Bit 7 ist das High-Bit
Word	16-Bit-Wert ohne Vorzeichen  Bit 0 ist das Low-Bit Bit 15 ist das High-Bit
Short	16-Bit-Wert mit Vorzeichen  Bit 0 ist das Low-Bit Bit 14 ist das High-Bit Bit 15 ist das Vorzeichen-Bit
DWord	32-Bit-Wert ohne Vorzeichen  Bit 0 ist das Low-Bit Bit 31 ist das High-Bit
Long	32-Bit-Wert mit Vorzeichen  Bit 0 ist das Low-Bit Bit 30 ist das High-Bit Bit 31 ist das Vorzeichen-Bit
Float	32-Bit-Gleitkommawert  Der Treiber interpretiert zwei aufeinanderfolgende Register als Gleitkommawert, indem das erste Register als Low-Wort und das zweite Register als High-Wort bewertet wird.
String	Mit Null beendete ASCII-Zeichenfolge

• \*Weitere Informationen dazu finden Sie unter [Adressbeschreibungen](#).



## Adressbeschreibungen

Adressspezifikationen sind je nach verwendetem Modell unterschiedlich. Wählen Sie einen Link aus der folgenden Liste aus, um bestimmte Adressinformationen für das entsprechende Modell zu erhalten.

[S7-200-Adressierung](#)

[S7-200-PPM-Adressierung](#)

## S7-200-Adressierung

Das S7-200-Adressierungsformat ist mit dem S7-200-PPM-Adressierungsformat identisch. Die Modellauswahl bestimmt in diesem Fall, ob der Treiber das PPI-Protokoll (normaler S7-200-Modus) oder den PPM-Modus (S7-200 im Punkt-zu-Punkt-Modem) verwendet. In beiden Fällen ist die Adressierung dieselbe.

Die Standard-Datentypen für dynamisch definierte Tags werden **fett** dargestellt.

Adresstyp	Bereich	Typ	Zugriff
Einzelne Eingaben	I00000-I65535 I00000-I65534 I00000-I65532  I00000.bb-I65535.bb I00000.bb-I65534.bb I00000.bb-I65532.bb	Byte <b>Word</b> , Short DWord, Long, Float  <b>Byte</b> Boolean, <b>Word</b> , Short <b>DWord</b> , Long	Lesen/Schreiben
Einzelne Ausgaben	Q00000-Q65535 Q00000-Q65534 Q00000-Q65532  Q00000.bb- Q65535.bb Q00000.bb- Q65534.bb Q00000.bb- Q65532.bb	Byte <b>Word</b> , Short DWord, Long, Float  <b>Byte</b> Boolean, <b>Word</b> , Short <b>DWord</b> , Long	Lesen/Schreiben
Interner Speicher	M00000-M65535 M00000-M65534 M00000-M65532  M00000.bb- M65535.bb M00000.bb- M65534.bb M00000.bb- M65532.bb	Byte, <b>Word</b> , Short DWord, Long, Float  <b>Byte</b> Boolean, <b>Word</b> , Short <b>DWord</b> , Long	Lesen/Schreiben
Spezieller Speicher	SM00000-SM65535 SM00000-SM65534 SM00000-SM65532  SM00000.bb- SM65535.bb SM00000.bb- SM65534.bb SM00000.bb- SM65532.bb	Byte <b>Word</b> , Short DWord, Long, Float  <b>Byte</b> Boolean, <b>Word</b> , Short <b>DWord</b> , Long	Lesen/Schreiben  SM0-SM29 sind schreib- geschützt
Variabler Speicher	V00000-V65535 V00000-V65534 V00000-V65532  V00000.bb-	Byte, <b>Word</b> , Short DWord, Long, Float, String	Lesen/Schreiben

Adresstyp	Bereich	Typ	Zugriff
	V65535.bb V00000.bb- V65534.bb V00000.bb- V65532.bb	<b>Byte</b> Boolean, <b>Word</b> , Short <b>DWord</b> , Long, String	
Zeitgeber aktuelle Werte	T00000-T65535	<b>DWord</b> , Long	Lesen/Schreiben
Zeitgeberstatus-Bits	T00000-T65535	Boolean*	Schreibgeschützt
Zähler aktuelle Werte	C00000-C65535	<b>Word</b> , Short	Lesen/Schreiben
Zählerstatus-Bits	C00000-C65535	Boolean*	Schreibgeschützt
Hochgeschwindigkeitszähler	HC00000-HC65535	<b>DWord</b> , Long	Schreibgeschützt
Analoge Eingaben	AI00000-AI65534**	<b>Word</b> , Short	Schreibgeschützt
Analoge Ausgaben	AQ00000-AQ65534**	<b>Word</b> , Short	Lesegeschützt

\*Für Zeitgeber- und Zählerstatus-Bits wird keine Punkt-Bit-Notation verwendet. Das Status-Bit für Zeitgeber 7 ist T7 und als Boolean deklariert.

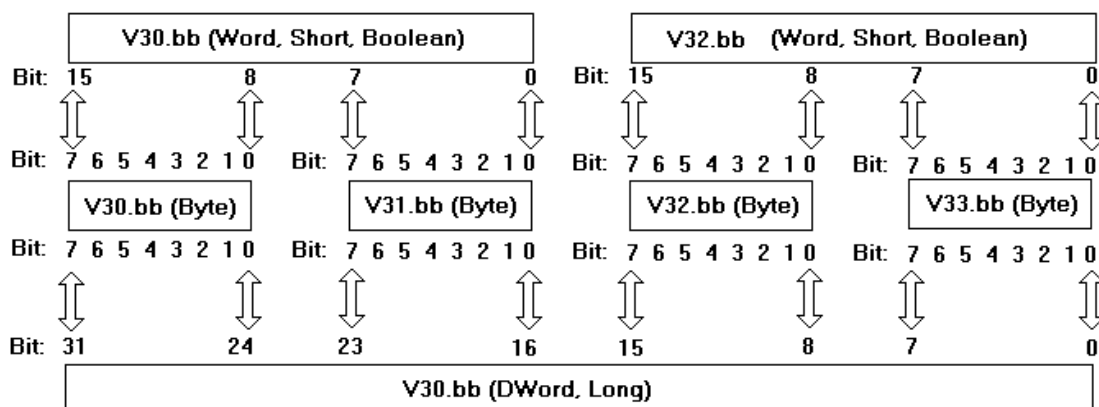
\*\*Für analoge Eingaben und Ausgaben muss die Adresse gerade sein (AI0, AI2, AI4 usw.). Analoge Ausgaben (AQ) sind schreibgeschützt: Es gibt keine Möglichkeit zum Lesen des Werts von analogen Ausgaben vom Gerät. Lesegeschützte Datentypen in diesem Treiber geben beim Lesen den zuletzt geschriebenen Wert zurück, wenn das anfängliche Schreiben auf das Gerät abgeschlossen ist. Wenn das anfängliche Schreiben nicht abgeschlossen wurde, gibt der Treiber beim Lesen einen Wert 0 zurück. Das gilt nur, während ein Client mit dem Server verbunden ist.

Die tatsächliche Anzahl von Adressen jedes Typs hängt vom verwendeten Siemens S7-200-Gerät ab. Nicht jeder Typ unterstützt notwendigerweise eine Adresse zwischen 0 und 65535. Informationen zu Adressbereichen finden Sie in der Dokumentation zum Gerät.

### Optional Punkt-Bits

Für die Datentypen "Byte", "Word", "Short", "DWord" oder "Long" kann ein optionales .bb (Punkt-Bit) an die Adresse angehängt werden, um ein Bit mit einem bestimmten Wert zu referenzieren. Die gültigen Bereiche für das optionale Bit sind 0-7 für Byte-Typen, 0-15 für die Typen "Word", "Short" und "Boolean", 0-31 für die Typen "DWord" und "Long" sowie 1-211 für "String"-Typen. "Float"-Typen unterstützen keine Bit-Operationen. Die Typen "Boolean" und "String" erfordern eine Bit-Nummer. Die Bit-Nummer für "String"-Typen gibt die Anzahl von Zeichen in der Zeichenfolge an.

Für dynamische Adressen mit Bit-Nummern im Bereich von 0-7 wird standardmäßig "Byte", im Bereich von 8-15 wird standardmäßig "Word" und im Bereich von 16-31 wird standardmäßig "DWord" verwendet. Für V-Speicheradressen mit einer Bit-Nummer größer als 31 wird standardmäßig "String" verwendet. Das folgende Diagramm veranschaulicht, wie der Treiber Bits innerhalb des Controllers zuordnet.



● **Hinweis:** V30.10@bool, V30.2@byte und V30.26@DWord referenzieren alle dasselbe Bit im Controller.

## Arrays

Bestimmte Speichertypen (I, Q, M, SM, V, AI und AQ) unterstützen eine Array-Operation. Arrays des Typs "Boolean" sind derzeit nicht zulässig. Um eine Array-Adresse anzugeben, hängen Sie *[Zeilen][Spalten]* an das Ende einer Adresse an. Wenn nur [Spalten] angegeben ist, wird für [Zeilen] standardmäßig 1 festgelegt. Bei Verwendung des Array-Typs ist es möglich, einen Block von 200 Byte gleichzeitig zu lesen und zu schreiben.

Die maximale Array-Größe für die Typen "Word" und "Short" beträgt 100 und für die Typen "DWord", "Long" und "Float" beträgt sie 50. Die Array-Größe wird durch die Multiplikation von Zeilen und Spalten bestimmt.

● **Hinweis:** Die maximale Array-Größe hängt auch von der maximalen Blockgröße des verwendeten Geräts ab.

## Beispiele

1. Um ein Array von 10 Gleitkommazahlen des variablen Speichers beginnend mit V10 zu lesen und zu schreiben, deklarieren Sie eine Adresse wie folgt: V10 [1][10]. Wählen Sie den Datentyp "Float".

● **Hinweis:** Dieses Array liest und schreibt Werte in die Register V10, V14, V18, V22 usw. bis V46.

2. Um einen Lese- und Schreibvorgang in Bit 23 des internen Speichers M20 "Long" durchzuführen, deklarieren Sie eine Adresse wie folgt: M20.23. Wählen Sie den Datentyp "Long".

## Zeichenfolgen

Der Treiber erlaubt das Speichern von Zeichenfolgen mit einer variablen Länge an variablen Speicherorten. Die Bit-Nummer gibt die Zeichenfolgenlänge (1-211) in Zeichen an. An das Gerät gesendete Zeichenfolgedaten, die jedoch kürzer sind als die Zeichenfolge-Zeichenanzahl (Bit-Nummer), enden mit Null. Zeichenfolgedaten, die die Zeichenlänge einhalten oder überschreiten, werden auf die Zeichenanzahl gekürzt und ohne ein Nullabschlusszeichen an das Gerät gesendet.

Um eine Zeichenfolge zu lesen und zu schreiben, die bei V5 beginnt und eine Länge von 10 Zeichen besitzt, deklarieren Sie eine Adresse wie folgt: V5.10. Wählen Sie den Datentyp "String".

● **Hinweise:**

1. Es würden die V-Speicherorte V5-V14 verwendet werden, um diese 10 Zeichen lange Zeichenfolge zu speichern.
2. Nicht alle Geräte unterstützen Anfragen für bis zu 211 Zeichen in einer einzelnen Transaktion. Weitere Informationen zur maximalen Anzahl von Zeichen, die in einer Transaktion angefordert werden können, finden Sie in der Dokumentation zum Gerät. Dieser Wert ist die längste Zeichenfolge, die der Treiber in das und vom Gerät lesen bzw. schreiben kann.

● **Achtung:** Wenn Sie die Typen "Word", "Short", "DWord", "Long" oder "Float" ändern, müssen Sie bedenken, dass jede Adresse an einem Byte-Offset im Gerät beginnt. Deshalb überlappen sich die Wörter V0 und V1 bei Byte 1. Durch das Schreiben in V0 wird der in V1 enthaltene Wert geändert. Auf ähnliche Weise können auch die Typen "DWord", "Long" und "Float" überlappen. Es wird empfohlen, diese Arbeitsspeichertypen so zu verwenden, dass kein Überlappen auftritt. Bei Verwendung von DWord kann beispielsweise durch V0, V4 und V8 verhindert werden, dass Bytes überlappen.

## S7-200-PPM-Adressierung

Das S7-200-PPM-Adressierungsformat ist mit dem S7-200-Adressierungsformat identisch. Die Modellauswahl bestimmt in diesem Fall, ob der Treiber das PPI-Protokoll (normaler S7-200-Modus) oder den PPM-Modus (S7-200 im Punkt-zu-Punkt-Modem) verwendet. In beiden Fällen ist die Adressierung dieselbe. Der PPM-Modus wird verwendet, wenn die Ziel-SPS über das EM241-Modemmodul oder über den im 10-Bit-Modus ausgeführten Programmier-Port verbunden wird.

Die Standard-Datentypen für dynamisch definierte Tags werden **fett** dargestellt.

Adresstyp	Bereich	Typ	Zugriff
Einzelne Eingaben	I00000-I65535 I00000-I65534	Byte <b>Word, Short</b>	Lesen/Schreiben

Adresstyp	Bereich	Typ	Zugriff
	I00000-I65532  I00000.bb-I65535.bb I00000.bb-I65534.bb I00000.bb-I65532.bb	DWord, Long, Float  <b>Byte</b> Boolean, <b>Word</b> , Short <b>DWord</b> , Long	
Einzelne Ausgaben	Q00000-Q65535 Q00000-Q65534 Q00000-Q65532  Q00000.bb- Q65535.bb Q00000.bb- Q65534.bb Q00000.bb- Q65532.bb	Byte <b>Word</b> , Short DWord, Long, Float  <b>Byte</b> Boolean, <b>Word</b> , Short <b>DWord</b> , Long	Lesen/Schreiben
Interner Speicher	M00000-M65535 M00000-M65534 M00000-M65532  M00000.bb- M65535.bb M00000.bb- M65534.bb M00000.bb- M65532.bb	Byte <b>Word</b> , Short DWord, Long, Float  <b>Byte</b> Boolean, <b>Word</b> , Short <b>DWord</b> , Long	Lesen/Schreiben
Spezieller Speicher	SM00000-SM65535 SM00000-SM65534 SM00000-SM65532  SM00000.bb- SM65535.bb SM00000.bb- SM65534.bb SM00000.bb- SM65532.bb	Byte <b>Word</b> , Short DWord, Long, Float  <b>Byte</b> Boolean, <b>Word</b> , Short <b>DWord</b> , Long	Lesen/Schreiben  SM0-SM29 sind schreib- geschützt
Variabler Speicher	V00000-V65535 V00000-V65534 V00000-V65532  V00000.bb- V65535.bb V00000.bb- V65534.bb V00000.bb- V65532.bb	Byte <b>Word</b> , Short DWord, Long, Float, String  <b>Byte</b> Boolean, <b>Word</b> , Short <b>DWord</b> , Long, String	Lesen/Schreiben
Zeitgeber aktuelle Werte	T00000-T65535	<b>DWord</b> , Long	Lesen/Schreiben
Zeitgeberstatus-Bits	T00000-T65535	Boolean*	Schreibgeschützt
Zähler aktuelle Werte	C00000-C65535	<b>Word</b> , Short	Lesen/Schreiben
Zählerstatus-Bits	C00000-C65535	Boolean*	Schreibgeschützt
Hochgeschwindigkeitszähler	HC00000-HC65535	<b>DWord</b> , Long	Schreibgeschützt
Analoge Eingaben	AI00000-AI65534***	<b>Word</b> , Short	Schreibgeschützt
Analoge Ausgaben	AQ00000- AQ65534***	<b>Word</b> , Short	Lesegeschützt

\*Für Zeitgeber- und Zählerstatus-Bits wird keine Punkt-Bit-Notation verwendet. Das Status-Bit für Zeitgeber 7 ist "T7" und als Boolean deklariert.

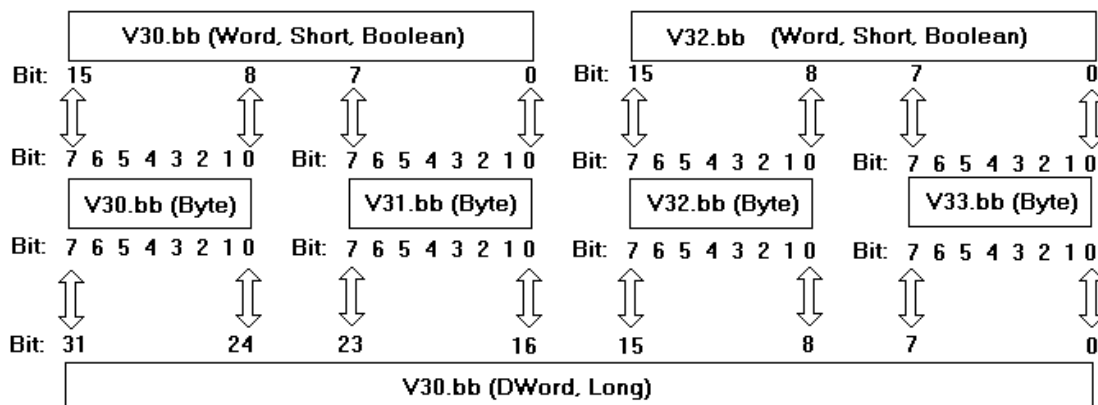
\*\*Für analoge Eingaben und Ausgaben muss die Adresse gerade sein (AI0, AI2, AI4 usw.). Analoge Ausgaben (AQ) sind schreibgeschützt: Es gibt keine Möglichkeit zum Lesen des Werts von analogen Ausgaben vom Gerät. Leseschutztypen dieses Treibers geben beim Lesen den zuletzt geschriebenen Wert zurück, wenn das anfängliche Schreiben auf dem Gerät abgeschlossen ist. Wenn das anfängliche Schreiben nicht abgeschlossen wurde, gibt der Treiber beim Lesen einen Wert 0 zurück. Das gilt nur, während ein Client mit dem Server verbunden ist.

Die tatsächliche Anzahl von Adressen jedes Typs hängt vom verwendeten Siemens S7-200-Gerät ab. Nicht jeder Typ unterstützt notwendigerweise eine Adresse zwischen 0 und 65535. Informationen zu Adressbereichen finden Sie in der Dokumentation zum Gerät.

### Optional Punkt-Bits

Für die Datentypen "Byte", "Word", "Short", "DWord" oder "Long" kann ein optionales .bb (Punkt-Bit) an die Adresse angehängt werden, um ein Bit mit einem bestimmten Wert zu referenzieren. Die gültigen Bereiche für das optionale Bit sind 0-7 für Byte-Typen, 0-15 für die Typen "Word", "Short" und "Boolean", 0-31 für die Typen "DWord" und "Long" sowie 1-211 für "String"-Typen. "Float"-Typen unterstützen keine Bit-Operationen. Die Typen "Boolean" und "String" erfordern eine Bit-Nummer. Die Bit-Nummer für "String"-Typen gibt die Anzahl von Zeichen in der Zeichenfolge an.

Für dynamische Adressen mit Bit-Nummern im Bereich von 0-7 wird standardmäßig "Byte", im Bereich von 8-15 wird standardmäßig "Word" und im Bereich von 16-31 wird standardmäßig "DWord" verwendet. Für V-Speicheradressen mit einer Bit-Nummer größer als 31 wird standardmäßig "String" verwendet. Das folgende Diagramm veranschaulicht, wie der Treiber Bits innerhalb des Controllers zuordnet.



● **Hinweis:** V30.10@bool, V30.2@byte und V30.26@DWord referenzieren alle dasselbe Bit im Controller.

### Arrays

Zusätzlich zu den oben aufgelisteten Adressformaten unterstützen bestimmte Speichertypen (I, Q, M, SM, V, AI, AQ) eine Array-Operation. Arrays des Typs "Boolean" sind derzeit nicht zulässig. Um eine Array-Adresse anzugeben, hängen Sie [Zeilen][Spalten] an das Ende einer Adresse an. Wenn nur [Spalten] angegeben ist, wird für [Zeilen] standardmäßig 1 festgelegt. Bei Verwendung des Array-Typs ist es möglich, einen Block von 200 Byte gleichzeitig zu lesen und zu schreiben.

Die maximale Array-Größe für die Typen "Word" und "Short" beträgt 100 und für die Typen "DWord", "Long" und "Float" beträgt sie 50. Die Array-Größe wird durch die Multiplikation von Zeilen und Spalten bestimmt.

● **Hinweis:** Die maximale Array-Größe hängt auch von der maximalen Blockgröße des verwendeten Geräts ab.

### Beispiele

1. Um ein Array von 10 Gleitkommazahlen des variablen Speichers beginnend mit V10 zu lesen und zu schreiben, deklarieren Sie eine Adresse wie folgt:

V10 [1][10]. Wählen Sie den Datentyp "Float". Dieses Array liest und schreibt Werte in die Register V10, V14, V18, V22 usw. bis V46.

2. Um einen Lese- und Schreibvorgang in Bit 23 des internen Speichers M20 "Long" durchzuführen, deklarieren Sie eine Adresse wie folgt: M20.23. Wählen Sie "Long" als Datentyp aus.

### Zeichenfolgen

Der Treiber erlaubt das Speichern von Zeichenfolgen mit einer variablen Länge an variablen Speicherorten. Die Bit-Nummer gibt die Zeichenfolgenlänge (1-211) in Zeichen an. An das Gerät gesendete Zeichenfolgedaten, die jedoch kürzer sind als die Zeichenfolge-Zeichenanzahl (Bit-Nummer), enden mit Null. Zeichenfolgedaten, die die Zeichenlänge einhalten oder überschreiten, werden auf die Zeichenanzahl gekürzt und ohne ein Nullabschlusszeichen an das Gerät gesendet.

Um eine Zeichenfolge zu lesen und zu schreiben, die bei V5 beginnt und eine Länge von 10 Zeichen besitzt, deklarieren Sie eine Adresse wie folgt: V5.10. Wählen Sie den Datentyp "String".

#### ● Hinweise:

1. Es werden die V-Speicherorte V5-V14 verwendet, um diese 10 Zeichen lange Zeichenfolge zu speichern.
2. Nicht alle Geräte unterstützen Anfragen für bis zu 211 Zeichen in einer einzelnen Transaktion. Weitere Informationen zur maximalen Anzahl von Zeichen, die in einer Transaktion angefordert werden können, finden Sie in der Dokumentation zum Gerät. Dieser Wert ist die längste Zeichenfolge, die der Treiber in das und vom Gerät lesen bzw. schreiben kann.

● Wenn Sie die Typen "Word", "Short", "DWord", "Long" oder "Float" ändern, müssen Sie bedenken, dass jede Adresse an einem Byte-Offset innerhalb des Geräts beginnt. Deshalb überlappen sich die Wörter V0 und V1 bei Byte 1. Durch das Schreiben in V0 wird der in V1 enthaltene Wert geändert. Auf ähnliche Weise können auch die Typen "DWord", "Long" und "Float" überlappen. Es wird empfohlen, diese Arbeitsspeichertypen so zu verwenden, dass kein Überlappen auftritt. Bei Verwendung von DWord kann beispielsweise durch V0, V4 und V8 verhindert werden, dass Bytes überlappen.

# Ereignisprotokollmeldungen

Die folgenden Informationen betreffen Meldungen, die im Fensterbereich Ereignisprotokoll in der Hauptbenutzeroberfläche angezeigt werden. Informationen zum Filtern und Sortieren der Detailansicht Ereignisprotokoll finden Sie in der OPC-Serverhilfe. In der Serverhilfe sind viele allgemeine Meldungen enthalten, die also auch gesucht werden sollten. Im Allgemeinen werden die Art der Meldung (Information, Warnung) sowie Fehlerbehebungsinformationen bereitgestellt (sofern möglich).

## **Der Block weist möglicherweise Adressen außerhalb des zulässigen Bereichs auf. | Blockstartadresse = '<Adresse>', Blockgröße = <Anzahl> (Byte).**

---

### **Fehlertyp:**

Warnung

### **Mögliche Ursache:**

Es wurde versucht, einen Speicherblock mit mindestens einem nicht vorhandenen Speicherort auf dem angegebenen Gerät zu referenzieren.

### **Mögliche Lösung:**

Überprüfen Sie, ob die Tags, die Adressen zugewiesen sind, im angegebenen Bereich auf dem Gerät liegen, und entfernen Sie alle, die ungültige Speicherorte referenzieren.

## **Fehlermaskendefinitionen**

---

**B** = Hardwareunterbrechung festgestellt

**F** = Framing-Fehler

**E** = E/A-Fehler

**O** = Zeichenpufferüberlauf

**R** = RX-Pufferüberlauf

**P** = Erhaltener Byte-Paritätsfehler

**T** = TX-Puffer voll

# Index

## A

Abfall 7  
Abrufverzögerung 7  
Adressbeschreibungen 17  
Alle Werte für alle Tags schreiben 8  
Analoge Ausgaben 18, 20  
Analoge Eingaben 18, 20  
Anfangsaktualisierungen aus Cache 13  
Anforderungs-Timeout 14  
Anfragen verwerfen, wenn herabgestuft 15  
Anstieg 7  
Arrays 19, 21  
Automatische Herabstufung 14  
Automatisches Wählen 8

## B

Baudrate 6  
Betriebsverhalten 7  
Boolean 16  
Byte 16

## C

COM-ID 6  
COM-Port 6

## D

Daten-Bits 6  
Datensammlung 12  
Datentypbeschreibung 16  
Der Block weist möglicherweise Adressen außerhalb des zulässigen Bereichs auf. | Blockstartadresse = '<Adresse>', Blockgröße = <Anzahl> (Byte). 23  
Diagnose 5  
DTR 7  
Durch Null ersetzen 9  
Durch Tag angegebenes Scan-Intervall berücksichtigen 13  
DWord 16



**E**

E/A 23  
Einstellungen auf Kanalebene 10  
Einzelne Ausgaben 17, 20  
Einzelne Eingaben 17, 19  
Ereignisprotokollmeldungen 23  
Ethernet-Einstellungen 7  
Ethernet-Kapselung 6

**F**

Fehlermaskendefinitionen 23  
Float 16  
Flusssteuerung 4, 7  
Framing 23

**G**

Geräteeigenschaften - Automatische Herabstufung 14  
Geräteeigenschaften - Redundanz 15  
Geräteeigenschaften - Zeitvorgabe 13  
Geteilt 6  
Globale Einstellungen 10

**H**

Hardware 23  
Herabstufen bei Fehler 14  
Herabstufungszeitraum 15  
Hochgeschwindigkeitszähler 18, 20

**I**

ID 5, 12  
ID-Format 12  
Identifikation 11  
Inaktive Verbindung schließen 7-8  
Inaktivitätsdauer bis Schließen 7-8  
Interner Speicher 17, 20

**K**

Kanaleigenschaften - Allgemein 5  
Kanaleigenschaften - Erweitert 9  
Kanaleigenschaften - Kommunikationsserialisierung 10  
Kanaleigenschaften - Schreiboptimierungen 8  
Kanaleigenschaften - Serielle Kommunikation 6  
Kanalzuweisung 12  
Keine 6  
Kommunikations-Timeouts 13  
Kommunikationsfehler melden 7-8  
Kommunikationsprotokolle 4

**L**

Lastausgleich 10  
Leseverarbeitung 8  
Long 16

**M**

Mit Vorzeichen 16  
Modell 12  
Modem 6, 8  
Modemeinstellungen 8

**N**

Netzwerk 4  
Netzwerk 1 - Netzwerk 500 10  
Netzwerkadapter 7  
Netzwerksmodus 10  
Nicht geändert 9  
Nicht normalisierte Float-Handhabung 9  
Nicht scannen, nur Abruf anfordern 13  
Nur den letzten Wert für alle Tags schreiben 9  
Nur den letzten Wert für nicht boolesche Tags schreiben 9

**O**

Ohne Vorzeichen 16  
OPC-Client 3

Operation ohne Kommunikation 8  
Optimierungsmethode 8

## P

Parität 6, 23  
Physisches Medium 6  
PPI-Programmierskabel 3  
PPI (Point-to-Point) 4  
PPM-Modem (Point-to-Point) 4  
Priorität 10  
Punkt-Bits 18, 21

## R

Redundanz 15  
RS-485 7  
RS232 4  
RS485 4  
RTS 7  
RX-Pufferüberlauf 23

## S

S7-200-Adressierung 17  
S7-200-PPM-Adressierung 19  
Scan-Modus 12  
Serielle Kommunikation 6  
Serielle Port-Einstellungen 6  
Servicezyklus 9  
Setup 4  
Short 16  
Siemens-Client-ID 11  
Siemens-S7-200-Gerät 3  
Simuliert 12  
Spezieller Speicher 17, 20  
Stopp-Bits 6  
String 16

## T

Tag-Zähler 5  
Timeout bis zum Herabstufen 14

Transaktionen pro Zyklus 10  
Treiber 12  
TX-Puffer 23

## U

Überlauf 23  
Übersicht 3  
Unterstützte Geräte 4  
Unterstützte Kabel 4  
Unterstützte Kommunikationsparameter 4

## V

Variabler Speicher 17, 20  
Verbindungs-Timeout 8, 13  
Verbindungstyp 6  
Versuche vor Timeout 14  
Verzögerung zwischen Geräten 10  
Virtuelles Netzwerk 10

## W

Word 16

## Z

Zähler aktuelle Werte 18, 20  
Zählerstatus-Bits 18, 20  
Zeichenfolgen 19, 22  
Zeitgeber aktuelle Werte 18, 20  
Zeitgeberstatus-Bits 18, 20  
Zeitvorgabe 13