

Ethernet-Treiber von Allen-Bradley

© 2021 PTC Inc. Alle Rechte vorbehalten.

Inhaltsverzeichnis

Ethernet-Treiber von Allen-Bradley	1
Inhaltsverzeichnis	2
Ethernet-Treiber von Allen-Bradley	4
Übersicht	5
Setup	6
Kanaleigenschaften - Allgemein	7
Tag-Zähler	7
Kanaleigenschaften - Ethernet-Kommunikation	8
Kanaleigenschaften - Schreiboptimierungen	8
Kanaleigenschaften - Erweitert	9
Kanaleigenschaften - Kommunikationsserialisierung	10
Geräteeigenschaften - Allgemein - ID	11
Betriebsmodus	11
Geräteeigenschaften - Scan-Modus	12
Geräteeigenschaften - Zeitvorgabe	13
Geräteeigenschaften - Automatische Herabstufung	14
Geräteeigenschaften - Kommunikationsparameter	14
Geräteeigenschaften - Protokollparameter	15
Geräteeigenschaften - Steckplatzkonfiguration	15
Geräteeigenschaften - Redundanz	16
Richtlinien für modulare E/A-Auswahl	18
Kommunikation optimieren	21
Datentypbeschreibung	22
Adressbeschreibungen	23
Allgemeine Adressierung	23
Ausgabedateien	23
Eingabedateien	25
Statusdateien	26
Binäre Dateien	27
Zeitgeberdateien	28
Zählerdateien	28
Steuerdateien	29
Ganzzahldateien	30
Float-Dateien	31
ASCII-Dateien	31
String-Dateien	32
Zeichenfolgenlänge	33
Offene Adressierung für SLC5/05	33
Adressierung für PLC-5-Familie und SoftPLC	33
BCD-Dateien	33
PID-Dateien	34
Meldungsdateien	35

Blocktransferdateien	36
Ereignisprotokollmeldungen	38
Datenblock von Gerät kann nicht gelesen werden. Empfangener Frame enthält Fehler. Blockstartadresse = '<Adresse>'.	38
Datenblock von Gerät kann nicht gelesen werden. Tag deaktiviert. Blockstartadresse = '<Adresse>', Statuscode = <Code>, erweiterter Statuscode = <Code>.	38
In Adresse auf Gerät kann nicht geschrieben werden. Empfangener Frame enthält Fehler. Adresse = '<Adresse>'.	39
Datenblock von Gerät kann nicht gelesen werden. Blockstartadresse = '<Adresse>', Statuscode = <Code>, erweiterter Statuscode = <Code>.	39
Datenblock von Gerät kann nicht gelesen werden. Tag deaktiviert. Blockstartadresse = '<Adresse>', Statuscode = <Code>.	40
In Adresse auf Gerät kann nicht geschrieben werden. Adresse = '<Adresse>', Statuscode = <Code>, erweiterter Statuscode = <Code>.	41
Datenblock von Gerät kann nicht gelesen werden. Blockstartadresse = '<Adresse>', Statuscode = <Code>.	41
In Adresse auf Gerät kann nicht geschrieben werden. Adresse = '<Adresse>', Statuscode = <Code>.	42
In Adresse auf Gerät kann nicht geschrieben werden. Paketlänge liegt außerhalb des zulässigen Bereichs. Adresse = '<Adresse>', erwartete Paketlänge = <niedrig> bis <hoch> (Byte).	42
In Adresse auf Gerät kann nicht geschrieben werden. TNS liegt außerhalb des zulässigen Bereichs. Adresse = '<Adresse>', erwarteter TNS-Bereich = <niedrig> bis <hoch>.	42
Index	43

Ethernet-Treiber von Allen-Bradley

Hilfeversion [1.057](#)

INHALT

[Übersicht](#)

Was ist Ethernet-Treiber von Allen-Bradley?

[Geräte-Setup](#)

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

[Kommunikation optimieren](#)

Wie erziele ich die beste Leistung mit Ethernet-Treiber von Allen-Bradley?

[Datentypbeschreibung](#)

Welche Datentypen werden von diesem Treiber unterstützt?

[Adressbeschreibungen](#)

Wie adressiere ich eine Datenposition auf einem Ethernet-Gerät von Allen-Bradley?

[Ereignisprotokollmeldungen](#)

Welche Meldungen werden vom Treiber erzeugt?

Übersicht

Ethernet-Treiber von Allen-Bradley bietet eine zuverlässige Möglichkeit, Ethernet-Geräte von Allen-Bradley mit Client-Anwendungen, u.a. HMI, SCADA, Historian, MES, ERP und zahlreichen benutzerdefinierten Anwendungen, zu verbinden. Dieser Treiber unterstützt SPS der SLC 5/05-, PLC-5-Reihe und SoftPLC von Allen-Bradley. Adressbereiche sind offen, um zukünftige Modelle für diese SPS-Reihen zu unterstützen.

Setup

Kommunikationsprotokoll

Allen-Bradley Ethernet

Unterstützte Geräte

SLC 5/05-Prozessor*

PLC-5-Reihe (ausschließlich PLC-5/250-Reihe)

SoftPLC

*Adressbereiche sind im Treiber offen, um neue Geräte zu ermöglichen. Der Treiber kann ggf. ein Gerät unterstützen, selbst wenn es oben nicht aufgelistet ist.

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 1024 pro Kanal.

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	<input type="checkbox"/> ID	
Allgemein	Name	Channel1
Serielle Kommunikation	Beschreibung	
Schreiboptimierungen	Treiber	
Erweitert	<input type="checkbox"/> Diagnose	
Kommunikationsserialisierung	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 - Ethernet-Kommunikation

Ethernet-Kommunikation kann für die Kommunikation mit Geräten verwendet werden.

Eigenschaftengruppen	Ethernet-Einstellungen	
Allgemein	Netzwerkadapter	Standard
Ethernet-Kommunikation		

Ethernet-Einstellungen

Netzwerkadapter: Geben Sie den zu bindenden Netzwerkadapter an. Wird dieses Feld leer gelassen oder es wird "Standard" ausgewählt, so wählt das Betriebssystem den Standardadapter aus.

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.

Eigenschaftengruppen	Schreiboptimierungen	
Allgemein	Optimierungsmethode	Nur den letzten Wert für alle Tags schr...
Serielle Kommunikation	Servicezyklus	10
Schreiboptimierungen		

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 Schiebescalter 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"/> Nicht normalisierte Float-Handhabung	
Allgemein	Gleitkommawerte	Durch Null ersetzen
Serielle Kommunikation	<input type="checkbox"/> Verzögerung zwischen Geräten	
Schreiboptimierungen	Verzögerung zwischen Geräten...	0
Erweitert		
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-Gleitkommadata 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	<input type="checkbox"/> Einstellungen auf Kanalebene	
Allgemein	Virtuelles Netzwerk	Keine
Serielle Kommunikation	Transaktionen pro Zyklus	1
Schreiboptimierungen	<input type="checkbox"/> Globale Einstellungen	
Erweitert	Netzwerkmodus	Lastausgleich
Kommunikationsserialisier...		

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?

Geräteeigenschaften - Allgemein - ID

Property Groups	Identification	
General	Name	Device1
Scan Mode	Description	
Timing	Channel Assignment	Channel1
Auto-Demotion	Driver	Allen-Bradley Ethernet
Communication Parameters	Model	SLC 5/05 Open
Protocol Parameters	ID	255.255.255.25
Slot Configuration	Operating Mode	
Redundancy	Data Collection	Enable
	Simulated	No

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: Die jeweilige Version des Geräts.

ID: Bei der Geräte-ID handelt es sich um die Netzwerkadresse der SPS.

• **Siehe auch:** [Betriebsmodus](#)

Betriebsmodus

Property Groups	Identification	
General	Operating Mode	
Scan Mode	Data Collection	Enable
	Simulated	No

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: Versetzen Sie das Gerät in den Simulationsmodus, oder beenden Sie den Modus. 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 Speichervorgabe 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	☐ Scan-Modus	
Allgemein	Scan-Modus	Vom Client angegebenes Scan-Intervall...
Scan-Modus	Anfangsaktualisierungen aus ...	Deaktivieren

Scan-Modus: Geben Sie an, wie Tags im Gerät für an abonnierende Clients gesendete Aktualisierungen gescannt 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. Die 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	☐ Communication Timeouts	
General	Connect Timeout (s)	3
Scan Mode	Request Timeout (ms)	1000
Timing	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 Baudraten 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	<input checked="" type="checkbox"/> Timing	
General	Inter-Request Delay (ms)	0
Scan Mode		
Timing		

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	<input checked="" type="checkbox"/> Automatische Herabstufung	
Allgemein	Herabstufen bei Fehler	Aktivieren
Scan-Modus	Timeout bis zum Herabstufen	3
Zeitvorgabe	Herabstufungszeitraum (ms)	10000
Automatische Herabstufung	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.

Tip: 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 Schreibenanforderungen während des Nicht-Scan-Zeitraums versucht werden sollten. Deaktivieren Sie diese Option, damit Schreibenanforderungen unabhängig vom Herabstufungszeitraum immer gesendet werden. Aktivieren Sie diese Option, um Schreibvorgänge zu verwerfen; auf dem Server schlägt jede von einem Client empfangene Schreibenanforderung automatisch fehl, und es wird keine Meldung im Ereignisprotokoll angezeigt.

Geräteeigenschaften - Kommunikationsparameter

Property Groups	<input checked="" type="checkbox"/> Communication Parameters	
General	Port	2222
Scan Mode	Request Size (bytes)	512
Timing		
Auto-Demotion		
Communication Parameters		
Protocol Parameters		
Slot Configuration		
Redundancy		

Port: Gibt die Port-Nummer an, für deren Verwendung das Remote-Gerät konfiguriert ist. Die Standardeinstellung ist 2222.

Anforderungsgröße: Gibt die maximale Anzahl an Bytes an, die von einem Gerät gleichzeitig angefordert werden können. Wählen Sie zur Optimierung der Leistung des Treibers beim Konfigurieren der Anforderungsgröße eine der folgenden Einstellungen aus: 32, 64, 128, 256, 512, 1024 oder 2000 Byte. Der Standardwert ist 512 Byte.

● **Tipp:** Für boolesche Arrays entspricht die Blockgröße dem Bit-Äquivalent (oder Blockgröße multipliziert mit 8). Beispiel: Eine Blockgröße von 512 Byte ist gleich $512 * 8 = 4096$ Bit.

Geräteeigenschaften - Protokollparameter

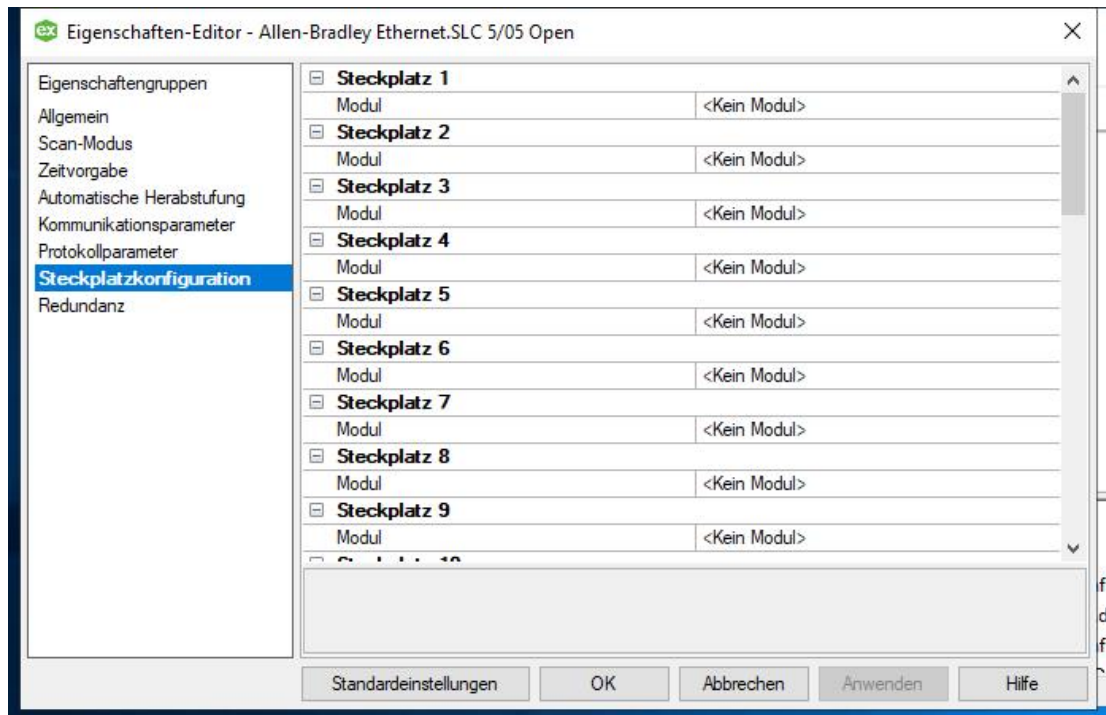
Property Groups General Scan Mode Timing Auto-Demotion Communication Parameters Protocol Parameters Slot Configuration Redundancy	<table border="1"> <tr> <td colspan="2">Protocol Parameters</td> </tr> <tr> <td>Destination Node Address (DST)</td> <td>0</td> </tr> <tr> <td colspan="2"> </td> </tr> </table>	Protocol Parameters		Destination Node Address (DST)	0		
Protocol Parameters							
Destination Node Address (DST)	0						

Zielknotenadresse (DST): Gibt die Zielknotenadresse an. Wählen Sie für DF1-Gateway-Anwendungen die Knotenadresse des Zielgeräts aus. Behalten Sie für Nicht-DF1-Gateway-Anwendungen die Knotenadresse mit der Standardeinstellung 0 bei.

● **Hinweis:** Beim Zielgerät handelt es sich um das DH+- oder DH-485-Gerät.

Geräteeigenschaften - Steckplatzkonfiguration

SLC 500-Modelle (mit modularen E/A-Racks) müssen für die Verwendung mit diesem Treiber konfiguriert werden, wenn durch den Treiber auf die E/A zugegriffen werden soll. Bis zu 30 Steckplätze können pro Gerät konfiguriert werden.



So verwenden Sie die Steckplatzkonfiguration:

1. Wählen Sie den zu konfigurierenden Steckplatz aus, indem Sie im Modul-Listenfeld auf die Zeile klicken.
2. Klicken Sie zum Auswählen eines Moduls in der Dropdown-Liste mit den verfügbaren Modulen darauf.
3. Konfigurieren Sie bei Bedarf die Eingabewörter und Ausgabewörter.
4. Wenn Sie einen Steckplatz bzw. ein Modul entfernen möchten, wählen Sie in der Dropdown-Liste mit den verfügbaren Modulen **Kein Modul** aus.
5. Klicken Sie nach Abschluss auf **OK**.

Tipp: Mit dem 0000-Generic Module können Sie die nicht in der Liste mit den verfügbaren Modulen enthaltene E/A konfigurieren.

Hinweis: Es ist üblich, offene Steckplätze im Rack zu haben, in denen sich kein physisches Modul befindet. Damit ein korrekter Zugriff auf Daten für die verschiedenen Steckplätze, die ein Modul aufweisen, erfolgen kann, muss den vorhergehenden Modulen die richtige Anzahl von Wörtern zugeordnet sein. Beispiel: Wenn nur Interesse an E/A in Steckplatz 3 besteht, die Steckplätze 1 und 2 jedoch E/A-Module enthalten, müssen von dieser Gruppe zur Steckplatzkonfiguration aus die richtigen Module für die Steckplätze 1, 2 und 3 ausgewählt werden.

0000-Generic Module

Mit dem Generic Module können Sie Eingabe- und Ausgabewörter für Module zuordnen, die in der Liste der verfügbaren Module nicht enthalten sind. Um das Generic Module ordnungsgemäß zu verwenden, müssen Benutzer die Anzahl der für jedes Modul erforderlichen Eingabe- und Ausgabewörter kennen.

Schlagen Sie zur Bestätigung von Eingabe- und Ausgabeanforderungen in der E/A-Benutzerhandbuchdokumentation von Allen-Bradley nach und denken Sie daran, dass die Anforderungen je nach Operationen der Klasse 1 oder 3 unterschiedlich sein können.

Weitere Informationen zur Anzahl der für jedes E/A-Modul zur Verfügung stehenden Eingabe- und Ausgabewörter finden Sie unter [Richtlinien für modulare E/A-Auswahl](#).

Geräteigenschaften - Redundanz

Eigenschaftengruppen	Redundanz	
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](#).

Richtlinien für modulare E/A-Auswahl

In der folgenden Tabelle wird die Anzahl der Eingabe- und Ausgabewörter aufgelistet, die für jedes E/A-Modul in der Liste der Steckplatzkonfiguration zur Verfügung stehen.

Tip: Mit dem Generic Module können Sie Eingabe- und Ausgabewörter für Module zuordnen, die in der Liste der verfügbaren Module nicht enthalten sind. Der Bereich der akzeptierten Werte wird in der Tabelle unten angezeigt.

Schlagen Sie im Benutzerhandbuch von Allen-Bradley das jeweilige zu konfigurierende E/A-Modul nach, um die Eingabe- und Ausgabeanforderungen zu bestätigen. Die Anforderungen sind je nach Operation der Klasse 1 oder 3 möglicherweise unterschiedlich.

Modultyp	Eingabewörter	Ausgabewörter
0000-Generic Module	0-255	0-255
1203-SM1 - SCANport-Kommunikationsmodul - Einfach	8	8
1203-SM1 - SCANport-Kommunikationsmodul - Erweitert	32	32
1394-SJT - GMC-Turbosystem	32	32
1746-BAS - Basismodul 500 5/01 (Konfiguration)	8	8
1746-BAS - Basismodul 5/02 (Konfiguration)	8	8
1746-HS - Einzelachsen-Bewegungs-Controller	4	4
1746-HSCE - Hochgeschwindigkeitszähler/Encoder	8	1
1746-HSRV - Bewegungssteuerungsmodul	12	8
1746-HSTP1 - Stepper-Controller-Modul	8	8
1746-I*16 - Jedes einzelne 16-pt-Eingabemodul	1	0
1746-I*32 - Jedes einzelne 32-pt-Eingabemodul	2	0
1746-I*8 - Jedes einzelne 8-pt-Eingabemodul	1	0
1746-IA16 - 16 Input 100/120 VAC	1	0
1746-IA4 - 4 Input 100/120 VAC	1	0
1746-IA8 - 8 Input 100/120 VAC	1	0
1746-IB16 - 16 Input (Senke) 24 VDC	1	0
1746-IB32 - 32 Input (Senke) 24 VDC	2	0
1746-IB8 - 8 Input (Senke) 24 VDC	1	0
1746-IC16 - 16 Input (Senke) 48 VDC	1	0
1746-IG16 - 16 Input [TTL] (Quelle) 5 VDC	1	0
1746-IH16 - 16 Input [Trans] (Senke) 125 VDC	1	0
1746-IM16 - 16 Input 200/240 VAC	1	0
1746-IM4 - 4 Input 200/240 VAC	1	0
1746-IM8 - 8 Input 200/240 VAC	1	0
1746-IN16 - 16 Input 24 VAC/VDC	1	0
1746-INI4I - Analog 4 Ch. Isol. Stromeingang	8	8
1746-INI4VI - Analog 4 Ch. Isol. Spannungs-/Stromeingang	8	8
1746-INO4I Analog 4 Ch. Isol. Stromeingang	8	8
1746-INO4VI Analog 4 Ch. Isol. Spannungs-/Stromeingang	8	8
1746-INT4 4 Ch. Isolierter Thermoelementeingang	8	8
1746-IO12 - 6 In 100/120 VAC 6 Out [Rly] VAC/VDC	1	1
1746-IO12DC - 6 Input 12 VDC, 6 Output [Rly]	1	1
1746-IO4 - 2 In 100/120 VAC 2 Out [Rly] VAC/VDC3	1	1
1746-IO8 - 4 In 100/120 VAC 4 Out [Rly] VAC/VDC4	1	1
1746-ITB16 - 16 Input [Schnell] (Senke) 24 VDC	1	0

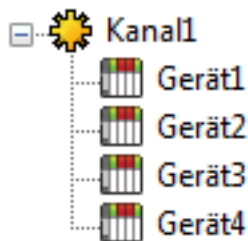
Modultyp	Eingabewörter	Ausgabewörter
1746-ITV16 - 16 Input [Schnell] (Quelle) 24 VDC	1	0
1746-IV16 - 16 Input (Quelle) 24 VDC	1	0
1746-IV32 - 32 Input (Quelle) 24 VDC	2	0
1746-IV8 - 8 Input (Quelle) 24 VDC	1	0
1746-NI4 - 4 Ch Analoge Eingabe	4	0
1746-NI8 - 8 Ch Analoge Eingabe, Klasse 1	8	8
1746-NI8 - 8 Ch Analoge Eingabe, Klasse 3	16	12
1746-NIO4I - Analog Comb 2 in & 2 Stromausgang	2	2
1746-NIO4V - Analog Comb 2 in & 2 Spannungsausgang	2	2
1746-NO4I - 4 Ch Analoger Stromausgang	0	4
1746-NO4V - 4 Ch Analoger Spannungsausgang	0	4
1746-NR4 - 4 Ch Rtd/Widerstand-Eingangsmodul	8	8
1746-NT4 - 4 Ch Thermoelement-Eingangsmodul	8	8
1746-NT8 Analog 8 Ch - Thermoelementeingang	8	8
1746-O*16 - Jedes einzelne 16-pt-Ausgabemodul	0	1
1746-O*32 - Jedes einzelne 32-pt-Ausgabemodul	0	2
1746-O*8 - Jedes einzelne 8-pt-Ausgabemodul	0	1
1746-OA16 - 16 Output (Triac) 100/240 VAC	0	1
1746-OA8 - 8 Output (Triac) 100/240 VAC	0	1
1746-OAP12 - 12 Output [Triac] 120/240 VDC	0	1
1746-OB16 - 16 Output [Trans] (Quelle) 10/50 VDC	0	1
1746-OB16E - 16 Output [Trans] (Quelle) Geschützt	0	1
1746-OB32 - 32 Output [Trans] (Quelle) 10/50 VDC	0	2
1746-OB32E - 32 Output [Trans] (Quelle) 10/50 VDC	0	2
1746-OB6EI - 6 Output [Trans] (Quelle) 24 VDC	0	1
1746-OB8 - 8 Output [Trans] (Quelle) 10/50 VDC	0	1
1746-OBP16 - 16 Output [Trans 1 amp] (SRC) 24 VDC	0	1
1746-OBP8 - 8 Output [Trans 2 amp] (Quelle) 24 VDC	0	1
1746-OG16 - 16 Output [TLL] (SENKE) 5 VDC	0	1
1746-OV16 - 16 Output [Trans] (Senke) 10/50 VDC	0	1
1746-OV32 - 32 Output [Trans] (Senke) 10/50 VDC	0	2
1746-OV8 - 8 Output [Trans] (Senke) 10/50 VDC	0	1
1746-OVP16 - 16 Output [Trans 1 amp] (Senke) 24VDC3	0	1
1746-OW16 - 16 Output [Relais] VAC/VDC	0	1
1746-OW4 - 4 Output [Relais] VAC/VDC	0	1
1746-OW8 - 8 Output [Relais] VAC/VDC	0	1
1746-OX8 - 8 Output [Isoliertes Relais] VAC/VDC	0	1
1747-DCM - Direktkommunikationsmodul (1/2 Rack)	4	4
1747-DCM - Direktkommunikationsmodul (1/4 Rack)	2	2
1747-DCM - Direktkommunikationsmodul (3/4 Rack)	6	6
1747-DCM - Direktkommunikationsmodul (Volles Rack)	8	8
1747-DSN - Verteilter E/A-Scanner - 30 Blöcke	32	32
1747-DSN - Verteilter E/A-Scanner - 7 Blöcke	8	8
1747-KE - Schnittstellenmodul, Reihe A	1	0
1747-KE - Schnittstellenmodul - Reihe B	8	8
1747-MNET - MNET-Netzwerk-Kommunikationsmodul	0	0

Modultyp	Eingabewörter	Ausgabewörter
1746-QS - Synchronisiertes Achsenmodul	32	32
1747-QV - Open Loop-Geschwindigkeitsregelung	8	8
1747-RCIF - Robot-Steuerung-Schnittstellenmodul	32	32
1747-SCNR - ControlNet-SLC-Scanner	32	32
1747-SDN - DeviceNet-Scanner-Modul	32	32
1747-SN - Remote-E/A-Scanner	32	32
AMCI-1561 AMCI Series 1561 Resolver Module	8	8

Kommunikation optimieren

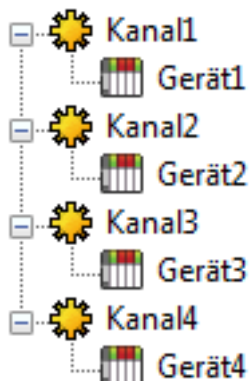
Ethernet-Treiber von Allen-Bradley wurde dafür konzipiert, eine optimale Leistung mit der geringsten Auswirkung auf die Gesamtleistung des Systems zu erzielen. Zwar ist der Treiber schnell, doch gibt es eine Reihe von Richtlinien zur Optimierung der Anwendung und zum Erreichen der maximalen Leistung.

Der Server bezeichnet Kommunikationsprotokolle wie Ethernet von Allen-Bradley als Kanal. Jeder in der Anwendung definierte Kanal stellt einen separaten Ausführungspfad im Server dar. Sobald ein Kanal festgelegt wurde, kann eine Reihe von Geräten unter diesem Kanal definiert werden. Jedes dieser Geräte stellt eine einzelne SPS von Allen-Bradley dar, von der Daten gesammelt werden. Zwar ermöglicht diese Methode zum Definieren der Anwendung ein hohes Leistungsniveau, doch kann damit nicht vollständig Nutzen aus Ethernet-Treiber von Allen-Bradley oder dem Netzwerk gezogen werden. Ein Beispiel dafür, wie die Anwendung bei Konfiguration mit einem Kanal möglicherweise aussehen kann, wird im Folgenden gezeigt.



Jedes Gerät wird unter einem einzelnen Ethernet-Kanal von Allen-Bradley angezeigt. In dieser Konfiguration muss sich der Treiber schnellstmöglich von einem Gerät zum nächsten bewegen, um Informationen in einem effektiven Intervall zu sammeln. Je mehr Geräte hinzugefügt oder je mehr Informationen von einem einzelnen Gerät angefordert werden, desto mehr leidet die Aktualisierungsrate insgesamt.

Wenn Ethernet-Treiber von Allen-Bradley nur einen Kanal definieren könnte, würde das Beispiel oben die einzige verfügbare Option darstellen. Der Treiber kann jedoch bis zu 256 Kanäle definieren. Durch Verwenden mehrerer Kanäle wird die Arbeitsbelastung bei der Datensammlung verteilt, indem mehrere Anfragen gleichzeitig an das Netzwerk gestellt werden. Ein Beispiel dafür, wie dieselbe Anwendung aussehen kann, wenn sie mit mehreren Kanälen konfiguriert wird, um die Leistung zu verbessern, wird im Folgenden gezeigt.



Jedes Gerät wird unter seinem eigenen Kanal definiert. In dieser Konfiguration wird ein einziger Ausführungspfad dediziert für das Sammeln von Daten von jedem Gerät eingesetzt. Wenn die Anwendung über 256 oder weniger Geräte verfügt, kann sie wie hier gezeigt optimiert werden.

Die Leistung kann sich verbessern, selbst wenn die Anwendung mehr als 256 Geräte hat. Zwar sind 256 oder weniger Geräte ideal, jedoch zieht die Anwendung weiterhin Nutzen aus zusätzlichen Kanälen. Obwohl das Verteilen der Gerätebelastung auf alle Kanäle zur Folge hat, dass sich der Server erneut von Gerät zu Gerät bewegt, kann dies mit weit weniger zu verarbeitenden Geräten auf einem einzigen Kanal erfolgen.

Datentypbeschreibung

Datentyp	Beschreibung
Boolean	Einzelnes Bit
Byte	8-Bit-Wert ohne Vorzeichen
Char	8-Bit-Wert mit Vorzeichen
Word	16-Bit-Wert ohne Vorzeichen
Short	16-Bit-Wert mit Vorzeichen
DWord	32-Bit-Wert ohne Vorzeichen
Long	32-Bit-Wert mit Vorzeichen
BCD	BCD mit zwei Byte gepackt, vier Dezimalstellen
LBCD	BCD mit vier Byte gepackt, acht Dezimalstellen
Float	32-Bit-IEEE-Gleitkommazahl
String	Mit Null beendetes Zeichen-Array

● **Hinweis:** Die Datentypen "DWord", "Long" und "LBCD" sind für keines der SPS-Modelle nativ. Beim Referenzieren einer 16-Bit-Position als 32-Bit-Wert handelt es sich bei der referenzierten Position um das Low-Wort und bei der nachfolgenden Position um das High-Wort. Beispiel: Wenn N7:10 als DWord-Datentyp ausgewählt wird, ist N7:10 das Low-Wort und N7:11 das High-Wort.

Adressbeschreibungen

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

Modell	Output	Input	Status	Binary	Timer	Counter	Control	Integer	Float	ASCII	String	BCD	Long	PID	Message	Block Transfer	Function
SLC5/05	X	X	X	X	X	X	X	X	X	X	X						
PLC5	X	X	X	X	X	X	X	X	X	X	X	X		X	X	X	

[Allgemeine Adressierung](#)

[Offene Adressierung für SLC 5/05](#)

[Adressierung für PLC-5-Familie und Soft-SPS](#)

Allgemeine Adressierung

Die allgemeinen Adressen unten beziehen sich auf SLC 5/05, PLC-5 und SoftPLC.

[Ausgabedateien](#)

[Eingabedateien](#)

[Statusdateien](#)

[Binäre Dateien](#)

[Zeitgeberdateien](#)

[Zählerdateien](#)

[Steuerdateien](#)

[Ganzzahldateien](#)

[Float-Dateien](#)

[ASCII-Dateien](#)

[String-Dateien](#)

Siehe auch:

[Offene Adressierung für SLC 5/05](#)

[Adressierung für PLC-5-Familie und SoftPLC](#)

Ausgabedateien

Die Syntax für den Zugriff auf Daten in der Ausgabedatei ist je nach SPS-Modell unterschiedlich. Datenpositionen sind für PLC-5- und SoftPLC-Modelle zum Lesen/Schreiben und sind für alle anderen Modelle schreibgeschützt. Der Standard-Datentyp für sämtliche Syntax wird **fett** dargestellt.

Syntax für PLC-5- und SoftPLC-Modell

Syntax	Datentyp
O:<Wort>	Short, Word , BCD
O:<Wort>/<Bit>	Boolean
O:<Wort>/<Bit>[Zeilen][Spalten]	Boolean*
O:<Wort>/<Bit>[Spalten]	Boolean*
O/Bit	Boolean
O/Bit[Zeilen][Spalten]	Boolean*
O/Bit[Spalten]	Boolean*

*Array-Typen

● **Hinweis:** Wort- und Bit-Adressinformationen liegen für PLC-5- und SoftPLC-Modelle im Oktalformat vor. Dies entspricht der Konvention der Programmiersoftware.

Syntax für offene SLC 5/05-Modelle (modulare E/A)

Syntax	Datentyp
O:<Steckplatz>	Short, Word , BCD
O:<Steckplatz>.<Wort>	Short, Word , BCD
O:<Steckplatz>/<Bit>	Boolean
O:<Steckplatz>/<Bit>[Zeilen][Spalten]	Boolean*
O:<Steckplatz>/<Bit>[Spalten]	Boolean*
O:<Steckplatz>.<Wort>/<Bit>	Boolean
O:<Steckplatz>.<Wort>/<Bit>[Zeilen][Spalten]	Boolean*
O:<Steckplatz>.<Wort>/<Bit>[Spalten]	Boolean*

*Array-Typen

Steckplatz- und Wortkonfigurationen

Die folgenden Steckplatz- und Wortpositionen sind für jedes Modell zulässig. Informationen dazu finden Sie unter [Geräte-Setup](#).

SPS-Modell	Min. Steckplatz	Max. Steckplatz	Max. Wort
SLC 5/05 offen	1	30	*
PLC-5-Familie	k.A.	k.A.	277 (oktal)
SoftPLC	k.A.	k.A.	777 (oktal)

*Die Anzahl der für jedes E/A-Modul verfügbaren Eingabe- oder Ausgabewörter ist unter [Richtlinien für modulare E/A-Auswahl](#) zu finden.

Beispiele

Alle Adressen liegen im Oktalformat vor.

PLC-5/SoftPLC	Adressen
O:0	Wort 0
O:37	Wort 31 (37 oktal = 31 dezimal)
O/42	Bit 34 (42 oktal = 34 dezimal)
O:2/2	Bit 2 Wort 2 (identisch mit O/42)
O/20[9]	Boolean-Array mit 9 Elementen, beginnend bei Bit 16 (20 oktal = 16 dezimal)
O/37[8][11]	Boolean Array mit 8 von 11 Elementen, beginnend bei Bit 31 (37 oktal = 31 dezimal)
O:47/5[3]	Boolean-Array mit 3 Elementen, beginnend bei Bit 5 Wort 39 (47 oktal = 39 dezimal)
O:11/13[3][7]	Boolean-Array mit 3 von 7 Elementen, beginnend bei Bit 11 (13 oktal = 11 dezimal) Wort 9 (11 oktal = 9 dezimal)

SLC 5/05	Adressen
O:1	Wort 0 Steckplatz 1
O:1.0	Wort 0 Steckplatz 1 (identisch mit O:1)
O:12	Wort 0 Steckplatz 12
O:12.2	Wort 2 Steckplatz 12
O:4.0/0	Bit 0 Wort 0 Steckplatz 4
O:4/0	Bit 0 Steckplatz 4 (identisch mit O:4.0/0)

SLC 5/05	Adressen
O:4.2/0	Bit 0 Wort 2 Steckplatz 4
O:4/32	Bit 32 Steckplatz 4 (identisch mit O:4.2/0)
O:2.12/3[17]	Boolean-Array mit 17 Elementen, beginnend bei Bit 3 Wort 12 Steckplatz 2
O:2.2/0[12][12]	Boolean-Array mit 12 von 12 Elementen, beginnend bei Bit 0 Wort 2 Steckplatz 2
O:2/43[5]	Boolean-Array mit 5 Elementen, beginnend bei Bit 43 Steckplatz 2
O:2/11[6][12]	Boolean-Array mit 6 von 12 Elementen, beginnend bei Bit 11 Steckplatz 2

Eingabedateien

Die Syntax für den Zugriff auf Daten in der Eingabedatei ist je nach SPS-Modell unterschiedlich. Datenpositionen sind für PLC-5-Modelle zum Lesen/Schreiben und sind für alle anderen Modelle schreibgeschützt. Der Standard-Datentyp für sämtliche Syntax wird **fett** dargestellt.

Syntax für PLC-5- und SoftPLC-Modell

Syntax	Datentyp
I:<Wort>	Short, Word , BCD
I:<Wort>/<Bit>	Boolean
I:<Wort>/<Bit>[Zeilen][Spalten]	Boolean*
I:<Wort>/<Bit>[Spalten]	Boolean*
I/Bit	Boolean
I/Bit[Zeilen][Spalten]	Boolean*
I/Bit[Spalten]	Boolean*

*Array-Typen

● **Hinweis:** Wort- und Bit-Adressinformationen liegen für PLC-5- und SoftPLC-Modelle im Oktalformat vor. Dies entspricht der Konvention der Programmiersoftware.

Syntax für offene SLC 5/05-Modelle (modulare E/A)

Syntax	Datentyp
I:<Steckplatz>	Short, Word , BCD
I:<Steckplatz>.<Wort>	Short, Word , BCD
I:<Steckplatz>/<Bit>	Boolean
I:<Steckplatz>/<Bit>[Zeilen][Spalten]	Boolean*
I:<Steckplatz>/<Bit>[Spalten]	Boolean*
I:<Steckplatz>.<Wort>/<Bit>	Boolean
I:<Steckplatz>.<Wort>/<Bit>[Zeilen][Spalten]	Boolean*
I:<Steckplatz>.<Wort>/<Bit>[Spalten]	Boolean*

*Array-Typen

Steckplatz- und Wortpositionen

Die folgenden Steckplatz- und Wortpositionen sind für jedes Modell zulässig. Weitere Informationen dazu finden Sie unter [Geräte-Setup](#).

SPS-Modell	Min. Steckplatz	Max. Steckplatz	Max. Wort
SLC 5/05 offen	1	30	*
PLC-5-Familie	k.A.	k.A.	277 (oktal)
SoftPLC-Familie	k.A.	k.A.	777 (oktal)

*Die Anzahl der für jedes E/A-Modul verfügbaren Eingabe- oder Ausgabewörter ist unter [Richtlinien für modulare E/A-Auswahl](#) zu finden.

Beispiele

Alle Adressen liegen im Oktalformat vor.

PLC-5/SoftPLC	Adressen
I:0	Wort 0
I:10	Word 8 (10 oktal = 8 dezimal)
I/20	Bit 16 (20 oktal = 16 dezimal)
I:1/0	Bit 0 Wort 1 (identisch mit I/20)
I/20[9]	Boolean-Array mit 9 Elementen, beginnend bei Bit 16 (20 oktal = 16 dezimal)
I/37[8][11]	Boolean Array mit 8 von 11 Elementen, beginnend bei Bit 31 (37 oktal = 31 dezimal)
I:47/5[3]	Boolean Array mit 3 Elementen, beginnend bei Bit 5 Wort 39 (47 oktal = 39 dezimal)
I:11/13[3][7]	Boolean-Array mit 3 von 7 Elementen, beginnend bei Bit 11 (13 oktal = 11 dezimal) Wort 9 (11 oktal = 9 dezimal)

SLC 5/05	Adressen
I:1	Wort 0 Steckplatz 1
I:1.0	Wort 0 Steckplatz 1 (identisch mit I:1)
I:12	Wort 0 Steckplatz 12
I:12.2	Wort 2 Steckplatz 12
I:4.0/0	Bit 0 Wort 0 Steckplatz 4
I:4/0	Bit 0 Steckplatz 4 (identisch mit I:4.0/0)
I:4.2/0	Bit 0 Wort 2 Steckplatz 4
I:4/32	Bit 32 Steckplatz 4 (identisch mit I:4.2/0)
I:2.12/3[17]	Boolean-Array mit 17 Elementen, beginnend bei Bit 3 Wort 12 Steckplatz 2
I:2.2/0[12][12]	Boolean-Array mit 12 von 12 Elementen, beginnend bei Bit 0 Wort 2 Steckplatz 2
I:2/43[5]	Boolean-Array mit 5 Elementen, beginnend bei Bit 43 Steckplatz 2
I:2/11[6][12]	Boolean-Array mit 6 von 12 Elementen, beginnend bei Bit 11 Steckplatz 2

Statusdateien

Geben Sie für den Zugriff auf Statusdateien ein Wort (und optional ein Bit innerhalb des Worts) an. Der Standard-Datentyp für sämtliche Syntax wird **fett** dargestellt.

Syntax	Datentyp
S:<Wort>	Short, Word , BCD, DWord, Long, LBCD
S:<Wort> [Zeilen][Spalten]	Short, Word , BCD, DWord, Long, LBCD*
S:<Wort> [Spalten]	Short, Word , BCD, DWord, Long, LBCD*
S:<Wort>/<Bit>	Boolean
S:<Wort>/<Bit> [Zeilen][Spalten]	Boolean *
S:<Wort>/<Bit> [Spalten]	Boolean *
S/Bit	Boolean
S/Bit [Zeilen][Spalten]	Boolean *
S/Bit [Spalten]	Boolean *

*Array-Typen

● **Hinweis:** Die Anzahl der Array-Elemente (in Byte) darf die angegebene Blockanforderungsgröße nicht überschreiten. Dies bedeutet, dass die Array-Größe 16 Wortwerte bei einer Blockanforderungsgröße von 32 Byte nicht überschreiten darf. Weitere Informationen dazu finden Sie unter [Blockanforderungsgröße](#).

Wortpositionen

Die folgenden Wortpositionen sind für jedes Modell zulässig. Die maximale Wortposition ist eine weniger, wenn als 32-Bit-Datentyp (Long, DWord oder "Long BCD") zugegriffen wird.

SPS-Modell	Max. Wort
SLC 5/05 offen	999
PLC-5-Familie	999
SoftPLC	31

Beispiel	Description
S:0	Wort 0.
S/26	Bit 26.
S:4/15	Bit 15 Wort 4.
S:10 [16]	Array mit 16 Elementen, beginnend bei Wort 10.
S:0 [4][8]	Array mit 4 mal 8 Elementen, beginnend bei Wort 0.
S/9 [5]	Boolean-Array mit 5 Elementen, beginnend bei Bit 9.
S/11 [3][7]	Boolean-Array mit 3 von 7 Elementen, beginnend bei Bit 11.
S:6/1 [6]	Boolean-Array mit 6 Elementen, beginnend bei Bit 1 Wort 6
S:13/5 [2][3]	Boolean-Array mit 2 von 3 Elementen, beginnend bei Bit 5 Wort 13.

Binäre Dateien

Geben Sie für den Zugriff auf binäre Dateien eine Dateinummer und ein Wort (und optional ein Bit im Wort) ein. Der Standard-Datentyp für sämtliche Syntax wird **fett** dargestellt.

Syntax	Datentyp
B<Datei>:<Wort>	Short, Word , BCD, DWord, Long, LBCD
B<Datei>:<Wort> [Zeilen][Spalten]	Short, Word , BCD, DWord, Long, LBCD*
B<Datei>:<Wort> [Spalten]	Short, Word , BCD, DWord, Long, LBCD*
B<Datei>:<Wort>/<Bit>	Boolean
B<Datei>:<Wort>/<Bit> [Zeilen][Spalten]	Boolean *
B<Datei>:<Wort>/<Bit> [Spalten]	Boolean *
B<Datei>/Bit	Boolean
B<Datei>/Bit [Zeilen][Spalten]	Boolean *
B<Datei>/Bit [Spalten]	Boolean *

*Array-Typen

● **Hinweis:** Die Anzahl der Array-Elemente (in Byte) darf die angegebene Blockanforderungsgröße nicht überschreiten. Dies bedeutet, dass die Array-Größe 16 Wortwerte bei einer Blockanforderungsgröße von 32 Byte nicht überschreiten darf. Weitere Informationen dazu finden Sie unter [Blockanforderungsgröße](#).

Dateinummern und Wortpositionen

Die folgenden Dateinummern und Wortpositionen sind für jedes Modell zulässig. Die maximale Wortposition ist eine weniger, wenn als 32-Bit-Datentyp (Long, DWord oder "Long BCD") zugegriffen wird.

SPS-Modell	Dateinummer	Max. Wort
SLC 5/05 offen	3, 9-999	999

SPS-Modell	Dateinummer	Max. Wort
PLC-5-Familie	3-999	1999
SoftPLC	3-9999	9999

Beispiel	Description
B3:0	Wort 0
B3/26	Bit 26
B12:4/15	Bit 15, Wort 4
B3:10 [20]	Array mit 20 Elementen, beginnend bei Wort 10
B15:0 [6][6]	Array mit 6 mal 6 Elementen, beginnend bei Wort 0
B3/7 [8]	Boolean-Array mit 8 Elementen, beginnend bei Bit 7
B3/32 [6][9]	Boolean-Array mit 6 mal 9 Elementen, beginnend bei Bit 32
B3:11/2 [12]	Boolean-Array mit 12 Elementen, beginnend bei Bit 2, Wort 11
B3:23/4 [5][8]	Boolean-Array mit 5 mal 8 Elementen, beginnend bei Bit 4, Wort 23

Zeitgeberdateien

Bei Zeitgeberdateien handelt es sich um einen strukturierten Typ, auf dessen Daten durch Festlegen einer Dateinummer, eines Elements und eines Felds zugegriffen wird. Der Standard-Datentyp hängt vom Feld ab, auf das zugegriffen wird. Ganzzahlfelder erhalten den Standard-Datentyp Word.

Syntax	Datentyp
T<Datei>:<Element>.<Feld>	Hängt von Feld ab

Dateinummern und Elemente

Die folgenden Dateinummern und maximalen Elemente sind für jedes Modell zulässig.

SPS-Modell	Dateinummer	Max. Element
SLC 5/05 offen	4, 9-999	999
PLC-5-Familie	3-999	1999
SoftPLC	3-9999	9999

Die folgenden Felder sind für jedes Element zulässig. Weitere Informationen zur Bedeutung der einzelnen Felder finden Sie in der SPS-Dokumentation.

Elementfeld	Datentyp	Zugriff
ACC	Short, Word	Lesen/Schreiben
PRE	Short, Word	Lesen/Schreiben
DN	Boolean	Schreibgeschützt
TT	Boolean	Schreibgeschützt
EN	Boolean	Schreibgeschützt

Beispiel	Description
T4:0.ACC	Akkumulator von Zeitgeber 0, Datei 4.
T4:10.DN	Abschluss-Bit von Zeitgeber 10, Datei 4.
T15:0.PRE	Voreinstellung von Zeitgeber 0, Datei 15.

Zählerdateien

Bei Zählerdateien handelt es sich um einen strukturierten Typ, auf dessen Daten durch Festlegen einer Dateinummer, eines Elements und eines Felds zugegriffen wird. Der Standard-Datentyp hängt vom Feld ab, auf das

zugegriffen wird. Ganzzahlfelder erhalten den Standard-Datentyp Word.

Syntax	Datentyp
C<Datei>:<Element>.<Feld>	Hängt von Feld ab

Dateinummern und Elemente

Die folgenden Dateinummern und maximalen Elemente sind für jedes Modell zulässig.

SPS-Modell	Dateinummer	Max. Element
SLC 5/05 offen	5, 9-999	999
PLC-5-Familie	3-999	1999
SoftPLC	3-9999	9999

Die folgenden Felder sind für jedes Element zulässig. Weitere Informationen zur Bedeutung der einzelnen Felder finden Sie in der SPS-Dokumentation.

Elementfeld	Datentyp	Zugriff
ACC	Short, Word	Lesen/Schreiben
PRE	Short, Word	Lesen/Schreiben
UA	Boolean	Schreibgeschützt
UN	Boolean	Schreibgeschützt
OV	Boolean	Schreibgeschützt
DN	Boolean	Schreibgeschützt
CD	Boolean	Schreibgeschützt
CU	Boolean	Schreibgeschützt

Beispiel	Description
C5:0.ACC	Akkumulator von Zähler 0, Datei 5
C5:10.DN	Abschluss-Bit von Zähler 10, Datei 5
C15:0.PRE	Voreinstellung von Zähler 0, Datei 15

Steuerdateien

Bei Steuerdateien handelt es sich um einen strukturierten Typ, auf dessen Daten durch Festlegen einer Dateinummer, eines Elements und eines Felds zugegriffen wird. Der Standard-Datentyp hängt vom Feld ab, auf das zugegriffen wird. Ganzzahlfelder erhalten den Standard-Datentyp Word.

Syntax	Datentyp
R<Datei>:<Element>.<Feld>	Hängt von Feld ab

Dateinummern und Elemente

Die folgenden Dateinummern und maximalen Elemente sind für jedes Modell zulässig.

SPS-Modell	Dateinummer	Max. Element
SLC 5/05 offen	6, 9-999	999
PLC-5-Familie	3-999	1999
SoftPLC	3-9999	9999

Die folgenden Felder sind für jedes Element zulässig. Weitere Informationen zur Bedeutung der einzelnen Felder finden Sie in der SPS-Dokumentation.

Elementfeld	Datentyp	Zugriff
LEN	Short, Word	Lesen/Schreiben
POS	Short, Word	Lesen/Schreiben
FD	Boolean	Schreibgeschützt
IN	Boolean	Schreibgeschützt
UL	Boolean	Schreibgeschützt
ER	Boolean	Schreibgeschützt
EM	Boolean	Schreibgeschützt
DN	Boolean	Schreibgeschützt
EU	Boolean	Schreibgeschützt
EN	Boolean	Schreibgeschützt

Beispiele	Description
R6:0.LEN	Längenfeld von Steuerung 0, Datei 6
R6:10.DN	Abschluss-Bit von Steuerung 10, Datei 6
R15:18.POS	Positionsfeld von Steuerung 18, Datei 15

Ganzzahldateien

Geben Sie für den Zugriff auf Ganzzahldateien eine Dateinummer und ein Wort (und optional ein Bit im Wort) an. Der Standard-Datentyp für sämtliche Syntax wird **fett** dargestellt.

Syntax	Datentyp
N<Datei>:<Wort>	Short, Word , BCD, DWord, Long, LBCD
N<Datei>:<Wort> [rows][cols]	Short, Word , BCD, DWord, Long, LBCD*
N<Datei>:<Wort> [cols]	Short, Word , BCD, DWord, Long, LBCD*
N<Datei>:<Wort>/<Bit>	Boolean
N<Datei>:<Wort>/<Bit> [Zeilen][Spalten]	Boolean*
N<Datei>:<Wort>/<Bit> [Spalten]	Boolean*
N<Datei>/Bit	Boolean
N<Datei>/Bit [Zeilen][Spalten]	Boolean*
N<Datei>/Bit [Spalten]	Boolean*

*Array-Typen

● **Hinweis:** Die Anzahl der Array-Elemente (in Byte) darf die angegebene Blockanforderungsgröße nicht überschreiten. Dies bedeutet, dass die Array-Größe 16 Wortwerte bei einer Blockanforderungsgröße von 32 Byte nicht überschreiten darf. Weitere Informationen dazu finden Sie unter [Blockanforderungsgröße](#).

Dateinummern und Wortpositionen

Die folgenden Dateinummern und maximalen Wortpositionen sind für jedes Modell zulässig. Die maximale Wortposition ist eine weniger, wenn als 32-Bit-Datentyp (Long, DWord oder Long-BCD) zugegriffen wird.

SPS-Modell	Dateinummer	Max. Wort
SLC 5/05 offen	7, 9-999	999
PLC-5-Familie	3-999	1999
SoftPLC	3-9999	9999

Beispiel	Beschreibung
N7:0	Wort 0

Beispiel	Beschreibung
N7/26	Bit 26
N12:4/15	Bit 15, Wort 4
N7:10 [8]	Array mit 8 Elementen, beginnend bei Wort 10
N15:0 [4][5]	Array mit 4 von 5 Elementen, beginnend bei Wort 0
N7/12 [9]	Boolean-Array mit 9 Elementen, beginnend bei Bit 12
N7/19 [3][11]	Boolean-Array mit 3 von 11 Elementen, beginnend bei Bit 19
N7:7/0 [10]	Boolean-Array mit 10 Elementen, beginnend bei Bit 0 Wort 7
N7:29/13 [2][15]	Boolean-Array mit 2 von 15 Elementen, beginnend bei Bit 13 Wort 29

Float-Dateien

Geben Sie für den Zugriff auf Float-Dateien eine Dateinummer und ein Element an. Der einzige zulässige Datentyp ist "Float".

Syntax	Datentyp
F<Datei>:<Element>	Float
F<Datei>:<Element> [rows][cols]	Float-Array
F<Datei>:<Element> [cols]	Float-Array

● **Hinweis:** Die Anzahl der Array-Elemente (in Byte) darf die angegebene Blockanforderungsgröße nicht überschreiten. Dies bedeutet, dass die Array-Größe 8 Float-Werte bei einer Blockanforderungsgröße von 32 Byte nicht überschreiten darf. Weitere Informationen dazu finden Sie unter [Blockanforderungsgröße](#).

Dateinummern und Wortpositionen

Die folgenden Dateinummern und maximalen Wortpositionen sind für jedes Modell zulässig.

SPS-Modell	Dateinummer	Max. Wort
SLC 5/05 offen	8-999	999
PLC-5-Familie	3-999	1999
SoftPLC	3-9999	9999

Beispiel	Beschreibung
F8:0	Float (Gleitkommazahl) 0
F8:10 [16]	Array mit 16 Elementen, beginnend bei Wort 10
F15:0 [4][4]	Array mit 16 Elementen, beginnend bei Wort 0

ASCII-Dateien

Geben Sie für den Zugriff auf Daten in einer ASCII-Datei eine Dateinummer und Zeichenposition an. Der Standard-Datentyp für sämtliche Syntax wird **fett** dargestellt.

Syntax	Datentyp
A<Datei>:<Char>	Char , Byte*
A<Datei>:<Char> [Zeilen][Spalten]	Char , Byte*
A<Datei>:<Char> [Spalten]	Char , Byte*
A<Datei>:<Wort-Offset>/Länge	String **

● **Hinweis:** Die Anzahl der Array-Elemente darf die angegebene Blockanforderungsgröße nicht überschreiten. Weitere Informationen dazu finden Sie unter [Blockanforderungsgröße](#).

*Die SPS packt zwei Zeichen pro Wort in die Datei, wobei das High-Byte das erste Zeichen und das Low-Byte das zweite Zeichen enthält. Die SPS-Programmiersoftware ermöglicht Zugriff auf der Wortebene oder der Zwei-

Zeichen-Ebene. Der AB-Ethernet-Treiber ermöglicht Zugriff auf die Zeichenebene. Dazu gibt es folgende Beispiele:

- Wenn Sie die Programmiersoftware verwenden, würde bei "A10:0 = AB" der Wert 'A' im High-Byte von A10:0 und 'B' im Low-Byte gespeichert.
- Unter Verwendung des AB-Ethernet-Treibers würden die beiden Zuweisungen A10:0 = A und A10:1 = B zur Folge haben, dass dieselben Daten im SPS-Speicher gespeichert werden.

**Das Referenzieren dieser Datei als Zeichenfolgendaten ermöglicht Zugriff auf Daten an Wortgrenzen wie die Programmiersoftware. Die Länge kann aus bis zu 236 Zeichen bestehen. Wenn eine an das Gerät gesendete Zeichenfolge kürzer ist als die durch die Adresse angegebene Länge, beendet der Treiber die Zeichenfolge mit Null, bevor sie an den Controller gesendet wird.

Dateinummern und Zeichenpositionen

Die folgenden Dateinummern und maximalen Zeichenpositionen sind für jedes Modell zulässig.

SPS-Modell	Dateinummer	Max. Zeichen
SLC 5/05 offen	9-999	1999
PLC-5-Familie	3-999	1999
SoftPLC	k.A.	k.A.

● **Hinweis:** Alle SLC 500-SPS unterstützen keine ASCII-Dateitypen. Weitere Informationen dazu finden Sie in der SPS-Dokumentation.

Beispiel	Description
A9:0	Zeichen 0 (High-Byte von Wort 0)
A27:10 [80]	Char-Array aus 80 Zeichen, das bei Zeichen 10 beginnt
A15:0 [4][16]	Char-Array aus 4 mal 16 Zeichen, das bei Zeichen 0 beginnt
A62:0/32	String aus 32 Zeichen, der bei Wort-Offset 0 beginnt

String-Dateien

Geben Sie für den Zugriff auf Daten in einer String-Datei eine Dateinummer und ein Element an. Der einzige zulässige Datentyp ist String, wobei es sich um mit Null beendete Arrays mit 82 Zeichen handelt. Der Treiber platziert das Nullabschlusszeichen basierend auf der von der SPS zurückgegebenen Zeichenfolgenlänge.

Syntax	Datentyp
ST<Datei>:<Element>	Zeichenfolge

● **Hinweis:** String-Arrays werden nicht unterstützt.

● **Tipp:** Die Zeichenfolgenlänge kann mit der [COPY- oder MOVE-Funktion](#) abgerufen werden.

Dateinummern und Wortpositionen

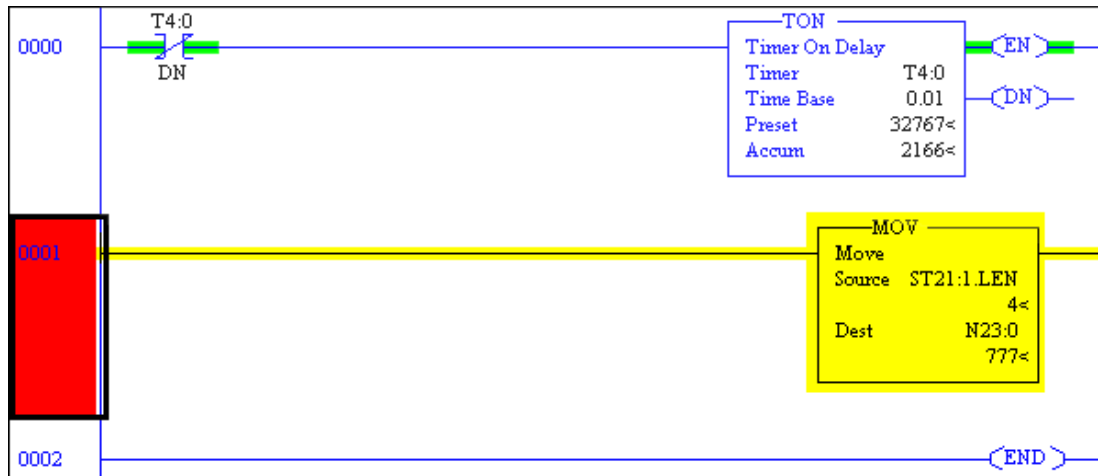
Die folgenden Dateinummern und maximalen Wortpositionen sind für jedes Modell zulässig.

SPS-Modell	Dateinummer	Max. Wort
SLC 5/05 offen	9-999	999
PLC-5-Familie	3-999	999
SoftPLC	3-9990	9999

Beispiel	Description
ST9:0	Zeichenfolge 0
ST18:10	Zeichenfolge 10

Zeichenfolgenlänge

Zwar wird das .LEN-Feld nicht unterstützt, doch kann die Zeichenfolgenlänge mit der COPY- oder MOVE-Funktion wie unten gezeigt abgerufen werden.



Offene Adressierung für SLC5/05

Die tatsächliche Anzahl der zur Verfügung stehenden Adressen ist vom SPS-Modell abhängig. Die Bereiche wurden ausgeweitet, um maximale Flexibilität mit zukünftigen Modellen zu ermöglichen. Wenn der Treiber bei Laufzeit feststellt, dass eine Adresse im Gerät nicht vorhanden ist, sendet er eine Fehlermeldung und entfernt das Tag aus seiner Scanliste.

- **Hinweis:** Dieses Modell hat keine bestimmte Adressierung.
- **Siehe auch:** [Allgemeine Adressierung](#)

Adressierung für PLC-5-Familie und SoftPLC

Allgemeine Adressierung

[Allgemeine Adressierung](#)

Modellspezifische Adressierung

[BCD-Dateien](#)

[PID-Dateien](#)

[Meldungsdateien](#)

[Blocktransferdateien](#)

BCD-Dateien

Geben Sie für den Zugriff auf BCD-Dateien eine Dateinummer und ein Wort an. Die einzigen zulässigen Datentypen sind BCD und Long-BCD. Der Standard-Datentyp ist immer BCD.

Syntax	Datentyp
D<Datei>:<Wort>	BCD, LBCD
D<Datei>:<Wort> [Zeilen][Spalten]	BCD, LBCD*
D<Datei>:<Wort> [Spalten]	BCD, LBCD*

*Array-Typen

- **Hinweis:** Die Anzahl der Array-Elemente (in Byte) darf die angegebene Blockanforderungsgröße nicht überschreiten. Dies bedeutet, dass die Array-Größe 16 BCD-Werte bei einer Blockanforderungsgröße von 32 Byte nicht überschreiten darf. Weitere Informationen dazu finden Sie unter [Blockanforderungsgröße](#).

Dateinummern und Wortpositionen

Die folgenden Dateinummern und maximalen Wortpositionen sind für jedes Modell zulässig.

SPS-Modell	Dateinummer	Max. Wort
SLC 5/05 offen	k.A.	k.A.
PLC-5-Familie	3-999	1999
SoftPLC	3-9999	9999

Beispiel	Beschreibung
D9:0	Wort 0
D27:10 [16]	Array mit 16 Elementen, beginnend bei Wort 10.
D15:0 [4][8]	Array mit 32 Elementen, beginnend bei Wort 0.

PID-Dateien

Bei PID-Dateien handelt es sich um einen strukturierten Typ, auf dessen Daten durch Festlegen einer Dateinummer, eines Elements und eines Felds zugegriffen wird. Der Standard-Datentyp hängt vom Feld ab, auf das zugegriffen wird. Ganzzahlfelder erhalten den Standard-Datentyp Word.

Syntax	Datentyp
PD<Datei>:<Element>.<Feld>	Hängt von Feld ab

Dateinummern und Elemente

Die folgenden Dateinummern und maximalen Elemente sind für jedes Modell zulässig.

SPS-Modell	Dateinummer	Max. Element
SLC 5/05 offen	k.A.	k.A.
PLC-5-Familie	3-999	999
SoftPLC	3-9999	9999

Die folgenden Felder sind für jedes Element zulässig. Weitere Informationen zur Bedeutung der einzelnen Felder finden Sie in der SPS-Dokumentation.

Elementfeld	Datentyp	Zugriff
SP	Real	Lesen/Schreiben
KP	Real	Lesen/Schreiben
KI	Real	Lesen/Schreiben
KD	Real	Lesen/Schreiben
BIAS	Real	Lesen/Schreiben
MAXS	Real	Lesen/Schreiben
MINS	Real	Lesen/Schreiben
DB	Real	Lesen/Schreiben
SO	Real	Lesen/Schreiben
MAXO	Real	Lesen/Schreiben
MINO	Real	Lesen/Schreiben
UPD	Real	Lesen/Schreiben
PV	Real	Lesen/Schreiben
ERR	Real	Lesen/Schreiben
OUT	Real	Lesen/Schreiben

Elementfeld	Datentyp	Zugriff
PVH	Real	Lesen/Schreiben
PVL	Real	Lesen/Schreiben
DVP	Real	Lesen/Schreiben
DVN	Real	Lesen/Schreiben
PVDB	Real	Lesen/Schreiben
DVDB	Real	Lesen/Schreiben
MAXI	Real	Lesen/Schreiben
MINI	Real	Lesen/Schreiben
TIE	Real	Lesen/Schreiben
FILE	Short, Word	Lesen/Schreiben
ELEM	Short, Word	Lesen/Schreiben
EN	Boolean	Lesen/Schreiben
CT	Boolean	Lesen/Schreiben
CL	Boolean	Lesen/Schreiben
PVT	Boolean	Lesen/Schreiben
DO	Boolean	Lesen/Schreiben
SWM	Boolean	Lesen/Schreiben
CA	Boolean	Lesen/Schreiben
MO	Boolean	Lesen/Schreiben
PE	Boolean	Lesen/Schreiben
INI	Boolean	Lesen/Schreiben
SPOR	Boolean	Lesen/Schreiben
OLL	Boolean	Lesen/Schreiben
OLH	Boolean	Lesen/Schreiben
EWD	Boolean	Lesen/Schreiben
DVNA	Boolean	Lesen/Schreiben
DVHA	Boolean	Lesen/Schreiben
PVLA	Boolean	Lesen/Schreiben
PVHA	Boolean	Lesen/Schreiben

Beispiel	Description
PD14:0.SP	Setpoint-Feld von PD 0 Datei 14
PD18:6.EN	Statusaktivierungs-Bit von PD 6 Datei 18

Meldungsdateien

Bei Meldungsdateien handelt es sich um einen strukturierten Typ, auf dessen Daten durch Festlegen einer Dateinummer, eines Elements und eines Felds zugegriffen wird. Der Standard-Datentyp hängt vom Feld ab, auf das zugegriffen wird. Ganzzahlfelder erhalten den Standard-Datentyp Word.

Syntax	Datentyp
MG<Datei>:<Element>.<Feld>	Hängt von Feld ab

Dateinummern und Elemente

Die folgenden Dateinummern und maximalen Elemente sind für jedes Modell zulässig.

SPS-Modell	Dateinummer	Max. Element
SLC 5/05 offen	k.A.	k.A.
PLC-5-Familie	3-999	999
SoftPLC	3-9999	9999

Die folgenden Felder sind für jedes Element zulässig. Weitere Informationen zur Bedeutung der einzelnen Felder finden Sie in der SPS-Dokumentation.

Elementfeld	Datentyp	Zugriff
ERR	Short, Word	Lesen/Schreiben
RLEN	Short, Word	Lesen/Schreiben
DLEN	Short, Word	Lesen/Schreiben
EN	Boolean	Lesen/Schreiben
ST	Boolean	Lesen/Schreiben
DN	Boolean	Lesen/Schreiben
ER	Boolean	Lesen/Schreiben
CO	Boolean	Lesen/Schreiben
EW	Boolean	Lesen/Schreiben
NR	Boolean	Lesen/Schreiben
TO	Boolean	Lesen/Schreiben

Beispiel	Description
MG14:0.RLEN	Angefordertes Längenfild von MG 0 Datei 14
MG18:6.CO	Fortsetzungs-Bit von MG 6 Datei 18

Blocktransferdateien

Bei Blocktransferdateien handelt es sich um einen strukturierten Typ, auf dessen Daten durch Festlegen einer Dateinummer, eines Elements und eines Felds zugegriffen wird. Der Standard-Datentyp hängt vom Feld ab, auf das zugegriffen wird. Ganzzahlfelder erhalten den Standard-Datentyp Word.

Syntax	Datentyp
BT<Datei>:<Element>.<Feld>	Hängt von Feld ab

Dateinummern und Elemente

Die folgenden Dateinummern und maximalen Elemente sind für jedes Modell zulässig.

SPS-Modell	Dateinummer	Max. Element
SLC 5/05 offen	k.A.	k.A.
PLC-5-Familie	3-999	1999
SoftPLC	3-9999	9999

Die folgenden Felder sind für jedes Element zulässig. Weitere Informationen zur Bedeutung der einzelnen Felder finden Sie in der SPS-Dokumentation.

Elementfeld	Datentyp	Zugriff
RLEN	Short, Word	Lesen/Schreiben
DLEN	Short, Word	Lesen/Schreiben
FILE	Short, Word	Lesen/Schreiben
ELEM	Short, Word	Lesen/Schreiben

Elementfeld	Datentyp	Zugriff
RW	Boolean	Lesen/Schreiben
ST	Boolean	Lesen/Schreiben
DN	Boolean	Lesen/Schreiben
ER	Boolean	Lesen/Schreiben
CO	Boolean	Lesen/Schreiben
EW	Boolean	Lesen/Schreiben
NR	Boolean	Lesen/Schreiben
TO	Boolean	Lesen/Schreiben

Beispiel	Description
BT14:0.RLEN	Angefordertes Längenfeld von BT 0, Datei 14
BT18:6.CO	Fortsetzungs-Bit von BT 6, Datei 18

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).

Datenblock von Gerät kann nicht gelesen werden. Empfangener Frame enthält Fehler. | Blockstartadresse = '<Adresse>'.

Fehlertyp:

Warnung

Mögliche Ursache:

1. Eine falsche Frame-Größe wurde empfangen.
2. TNS stimmt nicht überein.
3. Vom Gerät wurde ein ungültiger Antwortbefehl zurückgegeben.
4. Fehlausrichtung von Paketen aufgrund von Verbindung/Trennung zwischen PC und Gerät.
5. Fehlerhafte Verbindungskabel zwischen den Geräten verursachen Störungen.

Mögliche Lösung:

Zwar kann der Treiber ohne weitere Maßnahmen nach dem Fehler wiederhergestellt werden, es besteht jedoch möglicherweise ein Problem bei der Verkabelung oder dem Gerät selbst, das behoben werden sollte.

Datenblock von Gerät kann nicht gelesen werden. Tag deaktiviert. | Blockstartadresse = '<Adresse>', Statuscode = <Code>, erweiterter Statuscode = <Code>.

Fehlertyp:

Warnung

Mögliche Ursache:

1. Die angeforderte Adresse ist in der SPS nicht vorhanden.
2. Auf die angeforderte Adresse kann nicht zugegriffen werden, da die SPS einen Fehlerstatus aufweist.
3. Die Kommunikationsparameter für die Ethernet-Verbindung sind falsch.

Mögliche Lösung:

1. Vergewissern Sie sich, dass die Adresse in der SPS vorhanden ist.
2. Vergewissern Sie sich, dass die SPS keinen Fehlerstatus aufweist, oder stellen Sie den Betrieb der SPS wieder her.
3. Vergewissern Sie sich, dass die Kommunikationsparameter für die Ethernet-Verbindung richtig sind.
4. Vergewissern Sie sich, dass der richtige Port für das benannte Gerät angegeben wurde.
5. Vergewissern Sie sich, dass die IP-Adresse des benannten Geräts mit der des eigentlichen Geräts übereinstimmt.

🔸 Hinweis:

1. Überprüfen Sie den Statuscode und den erweiterten Statuscode, die von der SPS zurückgegeben werden. Der erweiterte Statuscode wird möglicherweise nicht immer zurückgegeben; Fehlerinformationen sind im Statuscode enthalten. Die Codes werden im Hexadezimalformat angezeigt.
2. Statuscodefehler im Low-Nibble weisen auf vom lokalen Knoten festgestellte Fehler hin. Der Treiber versucht diese Datenblöcke in regelmäßigen Abständen erneut zu lesen. Vom lokalen Knoten festgestellte Fehler treten auf, wenn das KF-Modul die Ziel-SPS nicht im Netzwerk erkennen kann.
3. Statuscodefehler im High-Nibble weisen auf von der SPS festgestellte Fehler hin. Diese Fehler werden generiert, wenn der vom Treiber angeforderte Datenblock in der SPS nicht verfügbar ist. Der Treiber fordert diese Blöcke nicht erneut an, nachdem dieser Fehler empfangen wurde. Dieser Fehler kann generiert werden, wenn die Adresse nicht in der SPS vorhanden ist.

In Adresse auf Gerät kann nicht geschrieben werden. Empfangener Frame enthält Fehler. | Adresse = '<Adresse>'.

Fehlertyp:

Warnung

Mögliche Ursache:

1. Falsche Frame-Größe wurde empfangen.
2. TNS stimmt nicht überein.
3. Vom Gerät wurde ein ungültiger Antwortbefehl zurückgegeben.
4. Fehlausrichtung von Paketen wurde durch Verbindung/Trennung zwischen PC und Gerät verursacht.
5. Fehlerhafte Verbindungskabel zwischen den Geräten verursachen Störungen.

Mögliche Lösung:

Zwar kann der Treiber ohne weitere Maßnahmen nach diesem Fehler wiederhergestellt werden, es besteht jedoch möglicherweise ein Problem bei der Verkabelung oder dem Gerät selbst, das behoben werden sollte.

Datenblock von Gerät kann nicht gelesen werden. | Blockstartadresse = '<Adresse>', Statuscode = <Code>, erweiterter Statuscode = <Code>.

Fehlertyp:

Warnung

Mögliche Ursache:

1. Die angeforderte Adresse ist in der SPS nicht vorhanden.
2. Auf die angeforderte Adresse kann nicht zugegriffen werden, da die SPS einen Fehlerstatus aufweist.
3. Die Kommunikationsparameter für die Ethernet-Verbindung sind falsch.

Mögliche Lösung:

1. Vergewissern Sie sich, dass die Adresse in der SPS vorhanden ist.
2. Vergewissern Sie sich, dass die SPS keinen Fehlerstatus aufweist, oder stellen Sie den Betrieb der SPS wieder her.
3. Vergewissern Sie sich, dass die Kommunikationsparameter für die Ethernet-Verbindung richtig sind.
4. Vergewissern Sie sich, dass der richtige Port für das benannte Gerät angegeben wurde.
5. Vergewissern Sie sich, dass die IP-Adresse des benannten Geräts mit der des eigentlichen Geräts übereinstimmt.

Hinweis:

1. Überprüfen Sie den Statuscode und den erweiterten Statuscode, die von der SPS zurückgegeben werden. Der erweiterte Statuscode wird möglicherweise nicht immer zurückgegeben; Fehlerinformationen sind im Statuscode enthalten. Die Codes werden im Hexadezimalformat angezeigt.
2. Statuscodefehler im Low-Nibble weisen auf vom lokalen Knoten festgestellte Fehler hin. Der Treiber versucht diese Datenblöcke in regelmäßigen Abständen erneut zu lesen. Vom lokalen Knoten festgestellte Fehler treten auf, wenn das KF-Modul die Ziel-SPS nicht im Netzwerk erkennen kann.
3. Statuscodefehler im High-Nibble weisen auf von der SPS festgestellte Fehler hin. Diese Fehler werden generiert, wenn der vom Treiber angeforderte Datenblock in der SPS nicht verfügbar ist. Der Treiber fordert diese Blöcke nicht erneut an, nachdem dieser Fehler empfangen wurde. Dieser Fehler kann generiert werden, wenn die Adresse nicht in der SPS vorhanden ist.

Datenblock von Gerät kann nicht gelesen werden. Tag deaktiviert. | Blockstartadresse = '<Adresse>', Statuscode = <Code>.

Fehlertyp:

Warnung

Mögliche Ursache:

1. Die angeforderte Adresse ist in der SPS nicht vorhanden.
2. Auf die angeforderte Adresse kann nicht zugegriffen werden, da die SPS einen Fehlerstatus aufweist.
3. Die Kommunikationsparameter für die Ethernet-Verbindung sind falsch.

Mögliche Lösung:

1. Vergewissern Sie sich, dass die Adresse in der SPS vorhanden ist.
2. Vergewissern Sie sich, dass die SPS keinen Fehlerstatus aufweist, oder stellen Sie den Betrieb der SPS wieder her.
3. Vergewissern Sie sich, dass die Kommunikationsparameter für die Ethernet-Verbindung richtig sind.
4. Vergewissern Sie sich, dass der richtige Port für das benannte Gerät angegeben wurde.
5. Vergewissern Sie sich, dass die IP-Adresse des benannten Geräts mit der des eigentlichen Geräts übereinstimmt.

Hinweis:

1. Überprüfen Sie den Statuscode und den erweiterten Statuscode, die von der SPS zurückgegeben werden. Der erweiterte Statuscode wird möglicherweise nicht immer zurückgegeben; Fehlerinformationen sind im Statuscode enthalten. Die Codes werden im Hexadezimalformat angezeigt.
2. Statuscodefehler im Low-Nibble weisen auf vom lokalen Knoten festgestellte Fehler hin. Der Treiber versucht diese Datenblöcke in regelmäßigen Abständen erneut zu lesen. Vom lokalen Knoten festgestellte Fehler treten auf, wenn das KF-Modul die Ziel-SPS nicht im Netzwerk erkennen kann.
3. Statuscodefehler im High-Nibble weisen auf von der SPS festgestellte Fehler hin. Diese Fehler werden generiert, wenn der vom Treiber angeforderte Datenblock in der SPS nicht verfügbar ist. Der Treiber fordert diese Blöcke nicht erneut an, nachdem dieser Fehler empfangen wurde. Dieser Fehler kann generiert werden, wenn die Adresse nicht in der SPS vorhanden ist.

In Adresse auf Gerät kann nicht geschrieben werden. | Adresse = '<Adresse>', Statuscode = <Code>, erweiterter Statuscode = <Code>.

Fehlertyp:

Warnung

Mögliche Ursache:

Die Adresse, in die geschrieben wurde, ist in der SPS nicht vorhanden.

Mögliche Lösung:

Vergewissern Sie sich, dass die Adresse in der SPS vorhanden ist.

● Hinweis:

1. Überprüfen Sie den Statuscode und den erweiterten Statuscode, die von der SPS zurückgegeben werden. Der erweiterte Statuscode wird möglicherweise nicht immer zurückgegeben; Fehlerinformationen sind im Statuscode enthalten. Die Codes werden im Hexadezimalformat angezeigt.
2. Statuscodefehler im Low-Nibble weisen auf vom lokalen Knoten festgestellte Fehler hin. Der Treiber versucht diese Datenblöcke in regelmäßigen Abständen erneut zu lesen. Vom lokalen Knoten festgestellte Fehler treten auf, wenn das KF-Modul die Ziel-SPS nicht im Netzwerk erkennen kann.
3. Statuscodefehler im High-Nibble weisen auf von der SPS festgestellte Fehler hin. Diese Fehler werden generiert, wenn der vom Treiber angeforderte Datenblock in der SPS nicht verfügbar ist. Der Treiber fordert diese Blöcke nicht erneut an, nachdem dieser Fehler empfangen wurde. Dieser Fehler kann generiert werden, wenn die Adresse nicht in der SPS vorhanden ist.

Datenblock von Gerät kann nicht gelesen werden. | Blockstartadresse = '<Adresse>', Statuscode = <Code>.

Fehlertyp:

Warnung

Mögliche Ursache:

1. Die angeforderte Adresse ist in der SPS nicht vorhanden.
2. Auf die angeforderte Adresse kann nicht zugegriffen werden, da die SPS einen Fehlerstatus aufweist.
3. Die Kommunikationsparameter für die Ethernet-Verbindung sind falsch.

Mögliche Lösung:

1. Vergewissern Sie sich, dass die Adresse in der SPS vorhanden ist.
2. Vergewissern Sie sich, dass die SPS keinen Fehlerstatus aufweist, oder stellen Sie den Betrieb der SPS wieder her.
3. Vergewissern Sie sich, dass die Kommunikationsparameter für die Ethernet-Verbindung richtig sind.
4. Vergewissern Sie sich, dass der richtige Port für das benannte Gerät angegeben wurde.
5. Vergewissern Sie sich, dass die IP-Adresse des benannten Geräts mit der des eigentlichen Geräts übereinstimmt.

● Hinweis:

1. Überprüfen Sie den Statuscode und den erweiterten Statuscode, die von der SPS zurückgegeben werden. Der erweiterte Statuscode wird möglicherweise nicht immer zurückgegeben; Fehlerinformationen sind im Statuscode enthalten. Die Codes werden im Hexadezimalformat angezeigt.

2. Statuscodefehler im Low-Nibble weisen auf vom lokalen Knoten festgestellte Fehler hin. Der Treiber versucht diese Datenblöcke in regelmäßigen Abständen erneut zu lesen. Vom lokalen Knoten festgestellte Fehler treten auf, wenn das KF-Modul die Ziel-SPS nicht im Netzwerk erkennen kann.
3. Statuscodefehler im High-Nibble weisen auf von der SPS festgestellte Fehler hin. Diese Fehler werden generiert, wenn der vom Treiber angeforderte Datenblock in der SPS nicht verfügbar ist. Der Treiber fordert diese Blöcke nicht erneut an, nachdem dieser Fehler empfangen wurde. Dieser Fehler kann generiert werden, wenn die Adresse nicht in der SPS vorhanden ist.

In Adresse auf Gerät kann nicht geschrieben werden. | Adresse = '<Adresse>', Statuscode = <Code>.

Fehlertyp:

Warnung

Mögliche Ursache:

1. Die Ethernet-Verbindung zwischen Gerät und Host-PC ist unterbrochen.
2. Die Kommunikationsparameter für die Ethernet-Verbindung sind falsch.
3. Möglicherweise wurde dem benannten Gerät eine falsche IP-Adresse zugewiesen.

Mögliche Lösung:

1. Überprüfen Sie die Verkabelung zwischen dem PC und dem Gerät.
2. Vergewissern Sie sich, dass für das benannte Gerät der richtige Port angegeben ist.
3. Vergewissern Sie sich, dass die IP-Adresse des benannten Geräts mit der des eigentlichen Geräts übereinstimmt.

In Adresse auf Gerät kann nicht geschrieben werden. Paketlänge liegt außerhalb des zulässigen Bereichs. | Adresse = '<Adresse>', erwartete Paketlänge = <niedrig> bis <hoch> (Byte).

Fehlertyp:

Informationen

In Adresse auf Gerät kann nicht geschrieben werden. TNS liegt außerhalb des zulässigen Bereichs. | Adresse = '<Adresse>', erwarteter TNS-Bereich = <niedrig> bis <hoch>.

Fehlertyp:

Informationen

Index

A

Adressbeschreibungen 23
Alle Werte für alle Tags schreiben 8
Allgemeine Adressierung 23
Anfangsaktualisierungen aus Cache 12
Anforderungs-Timeout 13
Anforderungsgröße 15
Anfragen verwerfen, wenn herabgestuft 14
ASCII-Dateien 31
Ausgabedateien 23
Automatische Herabstufung 14

B

BCD 22
BCD-Dateien 33
Betriebsmodus 11
Binäre Dateien 27
Blocktransferdateien 36
Boolean 22
Byte 22

C

Char 22

D

Datenblock von Gerät kann nicht gelesen werden. | Blockstartadresse = '<Adresse>', Statuscode = <Code>, erweiterter Statuscode = <Code>. 39
Datenblock von Gerät kann nicht gelesen werden. | Blockstartadresse = '<Adresse>', Statuscode = <Code>. 41
Datenblock von Gerät kann nicht gelesen werden. Empfangener Frame enthält Fehler. | Blockstartadresse = '<Adresse>'. 38
Datenblock von Gerät kann nicht gelesen werden. Tag deaktiviert. | Blockstartadresse = '<Adresse>', Statuscode = <Code>, erweiterter Statuscode = <Code>. 38
Datenblock von Gerät kann nicht gelesen werden. Tag deaktiviert. | Blockstartadresse = '<Adresse>', Statuscode = <Code>. 40
Datensammlung 11
Datentypbeschreibung 22
Diagnose 7

Durch Null ersetzen 9
Durch Tag angegebenes Scan-Intervall berücksichtigen 12
DWord 22

E

Eingabedateien 25
Einstellungen auf Kanalebene 10
Ereignisprotokollmeldungen 38
Ethernet-Einstellungen 8

F

Float 22
Float-Dateien 31

G

Ganzzahldateien 30
Geräteeigenschaften - Automatische Herabstufung 14
Geräteeigenschaften - Redundanz 16
Geräteeigenschaften - Zeitvorgabe 13
Globale Einstellungen 10

H

Herabstufen bei Fehler 14
Herabstufungszeitraum 14

I

ID 7, 11
Identifikation 11
In Adresse auf Gerät kann nicht geschrieben werden. | Adresse = '<Adresse>', Statuscode = <Code>, erweiterter Statuscode = <Code>. 41
In Adresse auf Gerät kann nicht geschrieben werden. | Adresse = '<Adresse>', Statuscode = <Code>. 42
In Adresse auf Gerät kann nicht geschrieben werden. Empfänger Frame enthält Fehler. | Adresse = '<Adresse>'. 39
In Adresse auf Gerät kann nicht geschrieben werden. Paketlänge liegt außerhalb des zulässigen Bereichs. | Adresse = '<Adresse>', erwartete Paketlänge = <niedrig> bis <hoch> (Byte). 42
In Adresse auf Gerät kann nicht geschrieben werden. TNS liegt außerhalb des zulässigen Bereichs. | Adresse = '<Adresse>', erwarteter TNS-Bereich = <niedrig> bis <hoch>. 42

K

Kanaleigenschaften - Allgemein 7
Kanaleigenschaften - Erweitert 9
Kanaleigenschaften - Ethernet-Kommunikation 8
Kanaleigenschaften - Kommunikationsserialisierung 10
Kanaleigenschaften - Schreiboptimierungen 8
Kanalzuweisung 11
Kommunikation optimieren 21
Kommunikations-Timeouts 13
Kommunikationsparameter 14

L

Lastausgleich 10
LBCD 22
Long 22

M

Meldung 35
Modell 11

N

Netzwerk 1 - Netzwerk 500 10
Netzwerkadapter 8
Netzwerksmodus 10
Nicht geändert 9
Nicht normalisierte Float-Handhabung 9
Nicht scannen, nur Abruf anfordern 12
Nur den letzten Wert für alle Tags schreiben 8
Nur den letzten Wert für nicht boolesche Tags schreiben 8

O

Optimierungsmethode 8

P

PID-Dateien 34
PLC5-Adressierung 33

Port 15
Priorität 10
Protokollparameter 15

R

Redundanz 16
Richtlinien für modulare E/A-Auswahl 18

S

Scan-Modus 12
Servicezyklus 9
Setup 6
Short 22
Simuliert 12
SLC5/05 33
Statusdateien 26
Steckplatzkonfiguration 15
Steuerdateien 29
String 22
String-Dateien 32

T

Tag-Zähler 7
Timeout bis zum Herabstufen 14
Transaktionen pro Zyklus 10
Treiber 11

U

Übersicht 5

V

Verbindungs-Timeout 13
Versuche vor Timeout 13
Verzögerung zwischen Geräten 9
Virtuelles Netzwerk 10

W

Word 22

Z

Zählerdateien 28

Zeichenfolgenlänge 33

Zeitgeberdateien 28

Zeitvorgabe 13

Zielknotenadresse 15