

Allen-Bradley ControlLogix Server-Treiber

© 2026 Kepware. Alle Rechte vorbehalten.

Inhaltsverzeichnis

Allen-Bradley ControlLogix Server-Treiber	1
Inhaltsverzeichnis	2
Willkommen bei der Allen-Bradley ControlLogix Server-Treiber Hilfe	4
Übersicht	4
Setup	5
Kanaleigenschaften – Allgemein	5
Tag-Zähler	6
Channel Properties — Ethernet Communications	6
Channel Properties — Write Optimizations	7
Channel Properties — Advanced	8
Channel Properties — EtherNet/IP Module	8
Geräteeigenschaften – Allgemein	9
Betriebsmodus	10
Tag-Zähler	11
Device Properties — Scan Mode	11
Geräteeigenschaften – Tag-Generierung	11
Device Properties — Controller Module	14
Device Properties — Native Tag Database	14
Tag Hierarchy	15
Geräteeigenschaften – Optionen	17
ControlLogix-Client-Gerätekonfiguration	17
Logix Config Object	19
Logix Config CSV Import	19
Defining New Logix Addresses	22
Data Types Description	23
Adressbeschreibungen	24
Adressformate	26
Tag Scope	28
Erweiterte Adressierung	28
Error Codes	31
EtherNet/IP Encapsulation Error Codes	31
CIP Error Codes	31
0x01 Extended Error Codes	32
0xFF Erweiterte Fehlercodes	32
Event Log Messages	33

Fehler beim Importieren der nativen Tag-Datenbank. Datei kann nicht geöffnet werden. Betriebssystemfehler = '<Fehler>'.	33
Fehler beim Importieren der nativen Tag-Datenbank. Datei kann nicht geöffnet werden, da ein allgemeiner Lesefehler aufgetreten ist.	33
Fehler beim Importieren der nativen Tag-Datenbank. Dateikodierung wird nicht unterstützt.	33
Fehler beim Importieren der nativen Tag-Datenbank.	33
Fehler beim Importieren der nativen Tag-Datenbank. Feldname wurde nicht erkannt. Nicht erkannter Feldname = '<Feldname>'.	34
Fehler beim Importieren der nativen Tag-Datenbank. Feldname ist doppelt vorhanden. Doppelter Feldname = '<Feld>'.	34
Fehler beim Importieren der nativen Tag-Datenbank. Tag-Feld-Identifikations-Datensatz fehlt.	34
Fehler beim Importieren der nativen Tag-Datenbank. Tag-Feld-Identifikations-Datensatz ist unvollständig.	35
Fehler beim Starten des unangeforderten Logix-Servers.	35
Ungültige Tag-Adresse. Natives Tag wurde nicht importiert. Ungültige Adresse = '<Adresse>'. ...	35
Zwei Kanäle wurden für die Verwendung derselben Netzwerkadapter-IP bzw. desselben TCP-Ports konfiguriert. Jeder Kanal muss an eine eindeutige lokale IP bzw. einen eindeutigen Port gebunden sein. Erster Kanal = '<Kanal>', zweiter Kanal = '<Kanal>'.	35
Zwei Geräte wurden für die Verwendung des gleichen Pfads vom EtherNet/IP-Modul konfiguriert. Jeder Kanal muss einen eindeutigen Pfad vom EtherNet/IP-Modul aufweisen. Erstes Gerät = '<Adresse>', zweites Gerät = '<Adresse>'.	35
Ungültige Tag-Adresse. Doppelte Tag-Adressen sind nicht zulässig. Ungültige Adresse = '<Adresse>'.	36
Speicherplatz für Tag konnte nicht zugeordnet werden. Tag-Adresse = '<Adresse>'.	36
Ungültiges natives Tag. Größe der einzelnen Tags ist auf 128 KB begrenzt. Tag-Adresse = '<Adresse>'.	36
Fehler beim Importieren des nativen Tags. Gesamtgröße der Datenbank-Tag-Daten ist auf 512 KB begrenzt. Tag-Adresse = '<Adresse>'.	37
Der angegebene TCP/IP-Port liegt außerhalb des zulässigen Bereichs. Der Standard-Port wird verwendet. Gültiger Bereich = <Nummer> bis <Nummer>, Standard-Port = <Nummer>.	37
Ein weiteres Gerät im Kanal ist bereits als lokale CPU registriert. Gerät = <Gerät>.	37
CSV-Import fehlgeschlagen. Vorhandene Logix-Adressen können nicht gelöscht werden.	37
CSV-Import fehlgeschlagen. Es können keine neuen Logix-Adressen erstellt werden.	37
Unerwartetes Fragment beim Verarbeiten fragmentierter Schreibvorgänge entdeckt. Logix-Adresse = <Tag>.	38
Native Tags importiert. Tag-Anzahl = <Anzahl>, Tag-Datenbank-Pfad = '<Pfad>'.	38
Tags generiert. Tag-Anzahl = <Anzahl>, Tag-Hierarchie-Modus = '<Modus>'.	38
Aufgrund geringer Ressourcen ist die automatische Tag-Generierung fehlgeschlagen.	38
Index	39

Willkommen bei der Allen-Bradley ControlLogix Server-Treiber Hilfe

Dies ist die Benutzerdokumentation für Kepware Allen-Bradley ControlLogix Server-Treiber. Dieses Hilfe-Center wird regelmäßig aktualisiert, um die neuesten Funktionen und Informationen widerzuspiegeln.

Übersicht

Was ist Allen-Bradley ControlLogix Server-Treiber?

Kanal-Setup

Wie gebe ich einen eindeutigen TCP/IP-Port für den Kanal an?

Geräte-Setup

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

Logix Config-Objekt

Wie konfiguriere ich ein Objekt für dieses Gerät?

Datentypbeschreibung

Welche Datentypen unterstützt dieser Treiber?

Adressbeschreibungen

Wie adressiere ich ein Tag auf einem ControlLogix-Gerät von Allen-Bradley?

Automatische Tag-Datenbankgenerierung

Wie kann ich Tags für Allen-Bradley ControlLogix Server-Treiber konfigurieren?

Fehlercodes

Welche Fehlercodes gibt es für das ControlLogix-Gerät von Allen-Bradley?

Ereignisprotokollmeldungen

Welche Meldungen erzeugt der Treiber?

Version 1.038

© 2026 Kepware. Alle Rechte vorbehalten.

Übersicht

Allen-Bradley ControlLogix Server-Treiber bietet eine zuverlässige Möglichkeit, ControlLogix-SPS von Allen-Bradley mit Client-Anwendungen zu verbinden. Es wird ein Rack der ControlLogix 5000-Reihe simuliert, in dem sich ein EtherNet/IP-Modul und bis zu 16 ControlLogix-CPU's befinden. SPS der ControlLogix 5000-Reihe können so konfiguriert werden, dass Lese-/Schreibvorgänge von CIP-Datentabellen vom/im Treiber mit der MSG-Kontaktplananweisung durchgeführt werden.

🔴 **Hinweis:** Allen-Bradley ControlLogix Server-Treiber beschränkt die Anzahl der Kanäle auf einen. Wenn der Netzwerkadapter und Port mit einer anderen Anwendung auf dem Host-Rechner in Konflikt steht, kann der Treiber keine eingehenden EtherNet/IP-Verbindungen annehmen.

🔵 *Weitere Informationen zur Konfiguration von SPS der ControlLogix 5000-Reihe für die Kommunikation mit dem Treiber finden Sie in der Rockwell/Allen-Bradley-Dokumentation.*

Setup

Allen-Bradley ControlLogix Server-Treiber fungiert als simuliertes Rack der ControlLogix 5000-Reihe, in dem sich ein Ethernet/IP-Modul befindet. Im Rack können bis zu 16 ControlLogix-CPU's enthalten sein, eine davon im EtherNet/IP-Modul (als lokal angesehen) sowie bis zu 15 einzelne CPU-Module (die für das EtherNet/IP-Modul als remote angesehen werden). Die von diesem Treiber unterstützte maximale Anzahl von Kanälen liegt bei 1. Die maximale Anzahl von Geräten, die von diesem Treiber unterstützt werden, liegt bei 16 pro Kanal. Bis zu 256 Geräte können ggf. gleichzeitig mit dem simulierten EtherNet/IP-Modul verbunden sein.

Unterstützte Geräte

Alle SPS der ControlLogix 5000-Reihe, die MSG-Anweisungen zum Lesen/Schreiben der CIP-Datentabelle unterstützen und mit Firmware-Revision 16 oder höher ausgeführt werden. Es werden verbundene und nicht verbundene CIP-Datentabellen-Lese- oder -Schreibvorgänge unterstützt.

Kommunikationsprotokoll

EtherNet/IP

SPS-Konfiguration

Geräte im Netzwerk müssen so programmiert werden, dass sie die MSG-Anweisung der CIP-Datentabelle zum Lesen/Schreiben für die Kommunikation mit dem Treiber verwenden sowie zurückgegebene Daten verarbeiten können.

• *Weitere Informationen zur Konfiguration der MSG-Anweisung finden Sie in den Programmiermeldungen in einem ControlLogix-System von Allen-Bradley.*

Sockets

Bis zu 256 eingehende Verbindungen werden gleichzeitig bedient. Die Verbindungen bleiben so lange offen, bis sie vom Absender geschlossen 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.

• **Siehe auch:** *API-Dokumentation zu Eigenschaftsinformationen finden Sie unter dem Endpunkt /config/v1/doc/drivers.*

Eigenschaftengruppen	Identifikation	
Allgemein	Name	
Scan-Modus	Beschreibung	
Zeitvorgabe	Treiber	
Automatische Herabstufung	Modell	
Tag-Generierung	Kanalzuweisung	
Protokolleinstellungen	ID	
Tag-Import	Betriebsmodus	
Ermittlung	Datensammlung	Aktivieren
	Simuliert	Nein
	Tag-Zähler	
	Statische Tags	1

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. Geben Sie 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: Wenn diese Option aktiviert ist, 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.

Hinweis: Diese Eigenschaft ist nicht verfügbar, wenn der Treiber oder das Betriebssystem die Diagnose nicht unterstützt.

Weitere Informationen dazu 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.

Channel Properties — Ethernet Communications

Ethernet Communication can be used to communicate with devices.

Eigenschaftengruppen	Ethernet-Einstellungen	
Allgemein	Netzwerkadapter	Standard
Ethernet-Kommunikation		

Ethernet Settings

Network Adapter: Specify the network adapter to bind. When left blank or Default is selected, the operating system selects the default adapter.

Channel Properties — Write Optimizations

The server must ensure that the data written from the client application gets to the device on time. Given this goal, the server provides optimization properties to meet specific needs or improve application responsiveness.

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

Write Optimizations

Optimization Method: Controls how write data is passed to the underlying communications driver. The options are:

- **Write All Values for All Tags:** This option forces the server to attempt to write every value to the controller. In this mode, the server continues to gather write requests and add them to the server's internal write queue. The server processes the write queue and attempts to empty it by writing data to the device as quickly as possible. This mode ensures that everything written from the client applications is sent to the target device. This mode should be selected if the write operation order or the write item's content must uniquely be seen at the target device.
- **Write Only Latest Value for Non-Boolean Tags:** Many consecutive writes to the same value can accumulate in the write queue due to the time required to actually send the data to the device. If the server updates a write value that has already been placed in the write queue, far fewer writes are needed to reach the same final output value. In this way, no extra writes accumulate in the server's queue. When the user stops moving the slide switch, the value in the device is at the correct value at virtually the same time. As the mode states, any value that is not a Boolean value is updated in the server's internal write queue and sent to the device at the next possible opportunity. This can greatly improve the application performance.
 - **Note:** This option does not attempt to optimize writes to Boolean values. It allows users to optimize the operation of HMI data without causing problems with Boolean operations, such as a momentary push button.
- **Write Only Latest Value for All Tags:** This option takes the theory behind the second optimization mode and applies it to all tags. It is especially useful if the application only needs to send the latest value to the device. This mode optimizes all writes by updating the tags currently in the write queue before they are sent. This is the default mode.

Duty Cycle: is used to control the ratio of write to read operations. The ratio is always based on one read for every one to ten writes. The duty cycle is set to ten by default, meaning that ten writes occur for each read operation. Although the application is performing a large number of continuous writes, it must be ensured that read data is still given time to process. A setting of one results in one read operation for every write operation. If there are no write operations to perform, reads are processed continuously. This allows optimization for applications with continuous writes versus a more balanced back and forth data flow.

● **Note:** It is recommended that the application be characterized for compatibility with the write optimization enhancements before being used in a production environment.

Channel Properties — Advanced

This group is used to specify advanced channel properties. Not all drivers support all properties; so the Advanced group does not appear for those devices.

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		

Non-Normalized Float Handling: A non-normalized value is defined as Infinity, Not-a-Number (NaN), or as a Denormalized Number. The default is Replace with Zero. Drivers that have native float handling may default to Unmodified. Non-normalized float handling allows users to specify how a driver handles non-normalized IEEE-754 floating point data. Descriptions of the options are as follows:

- **Replace with Zero:** This option allows a driver to replace non-normalized IEEE-754 floating point values with zero before being transferred to clients.
- **Unmodified:** This option allows a driver to transfer IEEE-754 denormalized, normalized, non-number, and infinity values to clients without any conversion or changes.

● **Note:** This property is disabled if the driver does not support floating-point values or if it only supports the option that is displayed. According to the channel's float normalization setting, only real-time driver tags (such as values and arrays) are subject to float normalization. For example, EFM data is not affected by this setting.

● *For more information on the floating-point values, refer to "How To ... Work with Non-Normalized Floating-Point Values" in the server help.*

Inter-Device Delay: Specify the amount of time the communications channel waits to send new requests to the next device after data is received from the current device on the same channel. Zero (0) disables the delay.

● **Note:** This property is not available for all drivers, models, and dependent settings.

Channel Properties — EtherNet/IP Module

Eigenschaftengruppen	<input type="checkbox"/> EtherNet/IP-Modul	
Allgemein	TCP/IP-Port	44818
Ethernet-Kommunikation		
Schreiboptimierungen		
Erweitert		
EtherNet/IP-Modul		

TCP/IP Port: Specify the TCP/IP and UDP port that provides a unique communication channel to the EtherNet/IP module. The valid range is 1 to 65535. The default is 44818.

● **Note:** The Allen-Bradley ControlLogix Server-Treiber currently limits the number of channels to one. If the network adapter and port conflicts with another application on the host machine, the driver cannot accept inbound EtherNet/IP connections.

● *For more information, refer to [Event Log Messages](#).*

Geräteeigenschaften – Allgemein

Ein Gerät stellt ein einzelnes Ziel in einem Kommunikationskanal dar. Wenn der Treiber mehrere Controller unterstützt, müssen Benutzer eine Geräte-ID für jeden Controller eingeben.

• **Siehe auch:** API-Dokumentation zu Eigenschaftsinformationen finden Sie unter dem Endpunkt `/config/v1/doc/drivers`.

Eigenschaftengruppen	
Allgemein	
Scan-Modus	

Identifikation	
Name	Device1
Beschreibung	
Treiber	Simulator
Modell	16 Bit Device
Kanalzuweisung	Channel1
ID-Format	Dezimal
ID	1

Identifikation

Name: Geben Sie den Namen des Geräts an. Es ist ein logischer, benutzerdefinierter Name, der bis zu 256 Zeichen lang sein und auf mehreren Kanälen verwendet werden kann.

• **Hinweis:** Zwar sind beschreibende Namen allgemein eine gute Idee, doch haben einige OPC-Client-Anwendungen beim Durchsuchen des Tag-Raums des OPC-Servers möglicherweise ein eingeschränktes Anzeigefenster. Der Geräte- und Kanalname werden ebenfalls Teil der Informationen zum Durchsuchen der Hierarchiebaumstruktur. Innerhalb eines OPC-Clients würde die Kombination aus Kanalname und Gerätenamenname als "ChannelName.DeviceName" angezeigt werden.

• **Weitere Informationen dazu finden Sie in der Serverhilfe unter "So benennen Sie Kanäle, Geräte, Tags und Tag-Gruppen richtig".**

Beschreibung: Geben Sie die benutzerdefinierten Informationen über dieses Gerät an.

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

Kanalzuweisung: Geben Sie den benutzerdefinierten Namen des Kanals an, zu dem dieses Gerät derzeit gehört.

Treiber: Ausgewählter Protokolltreiber für dieses Gerät.

Modell: Geben Sie den Gerätetyp an, der dieser ID zugeordnet ist. Der Inhalt des Dropdown-Menüs hängt vom Typ des verwendeten Kommunikationstreibers ab. Modelle, die von einem Treiber nicht unterstützt werden, sind deaktiviert. Wenn der Kommunikationstreiber mehrere Gerätemodelle unterstützt, kann die Modellauswahl nur geändert werden, wenn keine Client-Anwendungen mit dem Gerät verbunden sind.

• **Hinweis:** Wenn der Kommunikationstreiber mehrere Modelle unterstützt, sollten Benutzer versuchen, die Modellauswahl mit dem physischen Gerät abzugleichen. Wenn das Gerät im Dropdown-Menü nicht dargestellt wird, wählen Sie ein Modell aus, das dem Zielgerät am ehesten entspricht. Einige Treiber unterstützen die Modellauswahl "Offen", wodurch Benutzer kommunizieren können, ohne bestimmte Details des Zielgeräts zu kennen. *Weitere Informationen dazu finden Sie in der Dokumentation für den Treiber.*

ID: Geben Sie die treiberspezifische Station oder den treiberspezifischen Knoten für das Gerät an. Der Typ der eingegebenen ID hängt vom verwendeten Kommunikationstreiber ab. Für viele Kommunikationstreiber ist die ID ein numerischer Wert. Treiber, die eine numerische ID unterstützen, stellen Benutzern die Option

zum Eingeben eines numerischen Werts bereit, dessen Format den Anforderungen der Anwendung oder der Charakteristik des ausgewählten Kommunikationstreibers entsprechend angepasst werden kann. Das Format wird standardmäßig durch den Treiber festgelegt. Zu den Optionen gehören "Dezimal", "Oktal" und "Hexadezimal".

● **Hinweis:** Wenn der Treiber Ethernet-basiert ist oder eine unkonventionelle Station oder einen unkonventionellen Knotennamen unterstützt, kann die TCP/IP-Adresse des Geräts ggf. als Geräte-ID verwendet werden. TCP/IP-Adressen bestehen aus vier Werten, die durch Punkte getrennt sind, wobei jeder Wert im Bereich von 0 bis 255 liegt. Einige Geräte-IDs sind zeichenfolgenbasiert. Abhängig vom Treiber gibt es möglicherweise zusätzliche zu konfigurierende Eigenschaften innerhalb des ID-Felds.

Betriebsmodus

Eigenschaftengruppen	+ Identifikation	
Allgemein	- Betriebsmodus	
Scan-Modus	Datensammlung	Aktivieren
Automatische Herabstufung	Simuliert	Nein
Tag-Generierung	+ Tag-Zähler	

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 Daten werden nicht gespeichert, wenn der Server das Element entfernt (z.B., wenn der Server neu initialisiert wird). Die Standardeinstellung ist "Nein".

● Hinweise:

1. Aktualisierungen werden erst nach dem Trennen von Clients und nach dem Wiederherstellen von deren Verbindung angewendet.
2. Das 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.
3. 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.
4. Wird ein Gerät simuliert, so werden Aktualisierungen möglicherweise nicht schneller als innerhalb einer Sekunde auf dem Client angezeigt.

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

Tag-Zähler

Eigenschaftengruppen	+ Identifikation	
Allgemein	+ Betriebsmodus	
Scan-Modus	- Tag-Zähler	
	Statische Tags	0

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.

Device Properties — Scan Mode

The Scan Mode specifies the subscribed-client requested scan rate for tags that require device communications. Synchronous and asynchronous device reads and writes are processed as soon as possible; unaffected by the Scan Mode properties.

Eigenschaftengruppen	- Scan-Modus	
Allgemein	Scan-Modus	Vom Client angegebenes Scan-Intervall...
Scan-Modus	Anfangsaktualisierungen aus ...	Deaktivieren

Scan Mode: Specify how tags in the device are scanned for updates sent to subscribing clients. Descriptions of the options are:

- **Respect Client-Specified Scan Rate:** This mode uses the scan rate requested by the client.
- **Request Data No Faster than Scan Rate:** This mode specifies the value set as the maximum scan rate. The valid range is 10 to 99999990 milliseconds. The default is 1000 milliseconds.
 - **Note:** When the server has an active client and items for the device and the scan rate value is increased, the changes take effect immediately. When the scan rate value is decreased, the changes do not take effect until all client applications have been disconnected.
- **Request All Data at Scan Rate:** This mode forces tags to be scanned at the specified rate for subscribed clients. The valid range is 10 to 99999990 milliseconds. The default is 1000 milliseconds.
- **Do Not Scan, Demand Poll Only:** This mode does not periodically poll tags that belong to the device nor perform a read to get an item's initial value once it becomes active. It is the OPC client's responsibility to poll for updates, either by writing to the `_DemandPoll` tag or by issuing explicit device reads for individual items. *For more information, refer to "Device Demand Poll" in server help.*
- **Respect Tag-Specified Scan Rate:** This mode forces static tags to be scanned at the rate specified in their static configuration tag properties. Dynamic tags are scanned at the client-specified scan rate.

Initial Updates from Cache: When enabled, this option allows the server to provide the first updates for newly activated tag references from stored (cached) data. Cache updates can only be provided when the new item reference shares the same address, scan rate, data type, client access, and scaling properties. A device read is used for the initial update for the first client reference only. The default is disabled; any time a client activates a tag reference the server attempts to read the initial value from the device.

Geräteeigenschaften – Tag-Generierung

Mithilfe der Funktionen zur automatischen Tag-Datenbankgenerierung wird die Einrichtung einer Anwendung zu einem Plug-and-Play-Vorgang. Ausgewählte Kommunikationstreiber können so konfiguriert werden, dass automatisch eine Liste von Tags erstellt wird, die gerätespezifischen Daten entsprechen. Diese

automatisch generierten Tags (die von der Art des unterstützenden Treibers abhängen) können von den Clients durchsucht werden.

☛ *Nicht alle Geräte und Treiber unterstützen die automatische Tag-Datenbankgenerierung und nicht alle unterstützen die gleichen Datentypen. Lesen Sie für weitere Informationen die Beschreibungen der Datentypen oder die Liste der unterstützten Datentypen für jeden Treiber.*

Wenn das Zielgerät seine eigene lokale Tag-Datenbank unterstützt, liest der Treiber die Tag-Informationen des Geräts und verwendet die Daten zum Generieren von Tags innerhalb des Servers. Wenn das Gerät benannte Tags nicht nativ unterstützt, erstellt der Treiber eine Liste von auf treiberspezifischen Informationen basierenden Tags. Ein Beispiel dieser beiden Bedingungen sieht wie folgt aus:

1. Wenn ein Datenerfassungssystem seine eigene lokale Tag-Datenbank unterstützt, verwendet der Kommunikationstreiber die im Gerät gefundenen Tag-Namen, um die Tags des Servers zu erstellen.
2. Wenn ein Ethernet-E/A-System die Erkennung seiner eigenen verfügbaren E/A-Modultypen unterstützt, generiert der Kommunikationstreiber automatisch Tags auf dem Server, die auf den E/A-Modultypen im Ethernet-E/A-Rack basieren.

☛ **Hinweis:** Der Betriebsmodus zur automatischen Tag-Datenbankgenerierung ist komplett konfigurierbar. Weitere Informationen finden Sie in den nachfolgenden Eigenschaftsbeschreibungen.

Eigenschaftengruppen	☐ Tag-Generierung	
Allgemein	Bei Gerätestart	Nicht beim Start erstellen
Scan-Modus	Bei doppeltem Tag	Bei Erstellen löschen
Zeitvorgabe	Elmenteilgruppe	
Automatische Herabstufung	Automatisch generierte Untergruppen zulassen	Aktivieren
Tag-Generierung		

Bei Eigenschaftsänderung: Unterstützt das Gerät automatische Tag-Generierung, wenn bestimmte Eigenschaften geändert werden, so wird die Option **Bei Eigenschaftsänderung** angezeigt. Diese Option ist standardmäßig auf **Ja** festgelegt, kann jedoch auf **Nein** gesetzt werden, um zu steuern, wann eine Tag-Generierung stattfindet. In diesem Fall muss die Aktion **Tags erstellen** manuell aufgerufen werden, damit eine Tag-Generierung stattfindet.

Bei Gerätestart: Geben Sie an, wann OPC-Tags automatisch generiert werden. Es folgen Beschreibungen der Optionen:

- **Nicht beim Start generieren:** Mit dieser Option wird verhindert, dass der Treiber irgendwelche OPC-Tags dem Tag-Raum des Servers hinzufügt. Dies ist die Standardeinstellung.
- **Immer beim Start generieren:** Das Auswählen dieser Option hat zur Folge, dass der Treiber das Gerät für Tag-Informationen bewertet. Es werden auch jedes Mal, wenn der Server gestartet wird, Tags dem Tag-Raum des Servers hinzugefügt.
- **Beim ersten Start erstellen:** Das Auswählen dieser Option hat zur Folge, dass der Treiber das Zielgerät für Tag-Informationen bewertet, wenn das Projekt zum ersten Mal ausgeführt wird. Es werden bei Bedarf auch sämtliche OPC-Tags dem Tag-Raum des Servers hinzugefügt.

☛ **Hinweis:** Wenn die Option zum automatischen Generieren von OPC-Tags ausgewählt wird, müssen sämtliche Tags, die dem Tag-Raum des Servers hinzugefügt werden, mit dem Projekt gespeichert werden. Benutzer können das Projekt konfigurieren, um automatisch über das Menü **Tools | Optionen** zu speichern.

Bei doppeltem Tag: Wenn die automatische Tag-Datenbankgenerierung aktiviert wird, muss der Server wissen, wie mit Tags, die er möglicherweise zuvor hinzugefügt hat, oder mit Tags, die nach dem Kommunikationstreiber seit ihrer ursprünglichen Erstellung hinzugefügt oder geändert wurden, zu verfahren ist. Mit dieser Einstellung wird gesteuert, wie der Server OPC-Tags behandelt, die automatisch generiert wurden und derzeit im Projekt vorhanden sind. Es wird auch verhindert, dass sich automatisch generierte Tags auf dem Server ansammeln.

Beispiel: Wenn ein Benutzer die E/A-Module im Rack mit dem für **Immer beim Start erstellen** konfigurierten Server ändert, würden neue Tags jedes Mal dem Server hinzugefügt werden, wenn der Kommunikationstreiber ein neues E/A-Modul erkannt hat. Wenn die alten Tags nicht entfernt wurden, könnten sich viele unbenutzte Tags im Tag-Raum des Servers ansammeln. Die Optionen sind:

- **Bei Erstellen löschen:** Mit dieser Option werden sämtliche Tags gelöscht, die zuvor dem Tag-Raum hinzugefügt wurden, bevor sämtliche neuen Tags hinzugefügt werden. Dies ist die Standardeinstellung.
- **Nach Bedarf überschreiben:** Mit dieser Option wird der Server angewiesen, nur die Tags zu entfernen, die der Kommunikationstreiber durch neue Tags ersetzt. Sämtliche Tags, die nicht überschrieben werden, bleiben im Tag-Raum des Servers.
- **Nicht überschreiben:** Mit dieser Option wird verhindert, dass der Server sämtliche Tags entfernt, die zuvor generiert wurden oder bereits auf dem Server vorhanden waren. Der Kommunikationstreiber kann nur Tags hinzufügen, die völlig neu sind.
- **Nicht überschreiben, Fehler protokollieren:** Diese Option hat denselben Effekt wie die vorherige Option und sendet auch eine Fehlermeldung an das Ereignisprotokoll des Servers, wenn eine Tag-Überschreibung stattgefunden hätte.

● **Hinweis:** Das Entfernen von OPC-Tags wirkt sich auf Tags, die automatisch vom Kommunikationstreiber generiert wurden, sowie auf sämtliche Tags aus, die unter Verwendung von Namen, die generierten Tags entsprechen, hinzugefügt wurden. Benutzer sollten es vermeiden, Tags dem Server unter Verwendung von Namen hinzuzufügen, die möglicherweise den Tags entsprechen, die automatisch vom Treiber generiert werden.

Elternteilgruppe: Mit dieser Eigenschaft wird verhindert, dass sich automatisch generierte Tags mit Tags vermischen, die manuell eingegeben wurden, indem eine Gruppe festgelegt wurde, die für automatisch generierte Tags verwendet werden soll. Der Name der Gruppe kann bis zu 256 Zeichen lang sein. Diese Elternteilgruppe stellt einen Stammzweig bereit, dem alle automatisch generierten Tags hinzugefügt werden.

Automatisch generierte Untergruppen zulassen: Mit dieser Eigenschaft wird gesteuert, ob der Server automatisch Untergruppen für die automatisch generierten Tags erstellt. Dies ist die Standardeinstellung. Wenn diese Option deaktiviert ist, generiert der Server die Tags des Geräts in einer unstrukturierten Liste ohne jede Gruppierung. Im Serverprojekt werden die resultierenden Tags mit dem Adresswert benannt. Beispielsweise werden die Tag-Namen während des Generierungsprozesses nicht beibehalten.

● **Hinweis:** Wenn beim Generieren von Tags durch den Server einem Tag derselbe Name wie einem bestehenden Tag zugewiesen wird, erhöht das System automatisch auf die nächste höchste Nummer, sodass der Tag-Name nicht dupliziert wird. Beispiel: Wenn der Generierungsprozess das Tag "AI22" erstellt, das bereits existiert, wird stattdessen das Tag als "AI23" erstellt.

Erstellen: Initiiert die Erstellung automatisch generierter OPC-Tags. Wenn die Konfiguration des Geräts geändert wurde, wird der Treiber durch die Option **Tags erstellen** gezwungen, das Gerät erneut auf mögliche Tag-Änderungen zu bewerten. Ihre Fähigkeit, über die System-Tags aufgerufen zu werden, ermöglicht einer Client-Anwendung das Initiieren der Tag-Datenbankerstellung.

● **Hinweis:** **Tags erstellen** ist deaktiviert, wenn die Konfiguration ein Projekt offline bearbeitet.

Device Properties — Controller Module

Eigenschaftengruppen Allgemein Scan-Modus Tag-Generierung Controller-Modul Native Tag-Datenbank Optionen	Pfad von EtherNet/IP-Modul	
	Modultyp	Remote
	Steckplatz	1
	Pfad	10.80.68.217:44818,1,1

Module Type: Verify if the device is Local (part of the simulated EtherNet/IP Module) or Remote (which require a slot number for EtherNet/IP routing). There can be one Local CPU and up to fifteen Remote CPUs.

● **Note:** Each ControlLogix server device must be configured to represent a ControlLogix 5000 Series controller.

- **Local:** When Local, the Controller Module is treated as a CPU local to the simulated EtherNet/IP Module. There can only be one Local Controller Module per channel. The default is enabled.
- **Remote:** When Remote, the Controller Module is treated as a CPU separate from the simulated EtherNet/IP Module. There can be up to fifteen remote controller modules per channel. When this option is enabled, a slot must also be specified. The default is disabled.
- **Note:** This is disabled and forced to Remote (1) if another device is already configured as Local.

Slot: This property is part of the routing path to the Controller Module. It only contains slots that are currently available for the channel / device being configured. When a new slot is specified, the previous slot is made available for use in another device.

Path: This property represents the routing path to the Controller Module from the ControlLogix client device. It should be used during ControlLogix Client Configuration in the PLC.

● For more information on configuring ControlLogix server devices, refer to [ControlLogix Client Device Configuration](#).

Device Properties — Native Tag Database

The Allen-Bradley ControlLogix Server-Treiber can be configured to automatically generate a list of server tags that correspond to the Native Tag Database. This is also referred to as the Logix Configuration (or Logix Config). Logix Addresses in the Native Tag Database must be of pre-defined Logix Atomic Data Types, but may also be a part of a structured type.

The driver generates an OPC server tag for each Atomic Tag defined in the Native Tag Database. For array types, a server tag is defined for each element of the array. Array Tags can quickly increase the number of tags imported, as well as the number of tags available in the OPC server. Automatically generated tags are always configured with a client access of Read/Write.

● For more information, refer to [Logix Config](#).

Eigenschaftengruppen Allgemein Scan-Modus Tag-Generierung Controller-Modul Native Tag-Datenbank Optionen	Automatische Tag-Generierung	
	Tag-Hierarchie	Erweitert

Auto Tag Generation

• For more information, refer to [Device Properties — Tag Generation](#).

Tag Hierarchy: Select if the tag display "tree" organization should be Expanded or Condensed. The default is Expanded.

- **Expanded:** generated client tags are grouped similarly to RSLogix, with groups created for each segment (following a period in the tag address), structure, substructure, and array.
- **Condensed:** generated client tags are grouped similarly to the tag addressing, with groups created for each segment preceding a period.

• For more information on the Native Tag Database CSV format required for import, refer to [Native Tag Database CSV Import](#).

Tag Hierarchy

The automatically generated OPC server tags can follow one of two hierarchies: Expanded or Condensed. The default is Expanded Mode.

Expanded Mode

In Expanded Mode, the automatically generated server tags follow a Group/Tag hierarchy consistent with the tag hierarchy in RSLogix5000. Groups are created for each segment that precedes a period and are created in logical groupings. Groups created include the following:

- Global (Controller) Scope
- Structures and Substructures
- Arrays

• **Note:** Groups are not created for .bit addresses.

Basic Global Tags

Basic Global Tags (or non-structure, non-array tags) are placed under the Global group. Each Structure and Array Tag is provided with its own subgroup of the parent group. By organizing the data in this fashion, the OPC server's Tag View mimics RSLogix5000.

• **Note:** The name of the Structure/Array subgroup also provides a description. For example, an array "tag1 [1,6]" defined in the controller has a subgroup name "tag1[x,y]". In this example, x signifies that dimension 1 exists, and y signifies that dimension 2 exists. Furthermore, the tags within an array subgroup are the elements of that array unless explicitly limited. The tags within a structure subgroup are the structure members themselves. A structure that contains an array has an array subgroup of the structure group created as well.

Array Tags

A group is created for each array that contains the array's elements. Group names have the notation <array name>[x,y,z], where:

- **[x,y,z]:** 3 dimensional array.
- **[x,y]:** 2 dimensional array
- **[x]:** 1 dimensional array

• **Note:** Array Tags have the notation <tag element>_XXXX_YYYYY_ZZZZ. For example, element "tag1 [12,2,987]" has the tag name "tag1_12_2_987".

Simple Example

Name	Value	Force Mask	Style	Data Type
MyTag	{...}	{...}		MyDataType
MyTag.Member1	{...}	{...}	Decimal	DINT[10]
MyTag.Member1[0]	0		Decimal	DINT
MyTag.Member1[1]	0		Decimal	DINT
MyTag.Member1[2]	0		Decimal	DINT
MyTag.Member1[3]	0		Decimal	DINT

Projekt	Logix-Adresse	Logix-Datentyp	Zugriffstyp	Beschreibung
Konnektivität	Mitglied1	DINT	Schreibgeschützt	Mitglied1
Kanal	Mitglied2	DINT	Schreibgeschützt	Mitglied2
Gerät	Mitglied3	DINT	Schreibgeschützt	Mitglied3
Global	Mitglied4	DINT	Schreibgeschützt	Mitglied4
MeinEtikett	Mitglied5	DINT	Schreibgeschützt	Mitglied5
Mitglied	Mitglied6	DINT	Schreibgeschützt	Mitglied6
	Mitglied7	DINT	Schreibgeschützt	Mitglied7
	Mitglied8	DINT	Schreibgeschützt	Mitglied8
	Mitglied9	DINT	Schreibgeschützt	Mitglied9

Complex Example

A Logix Address defined with the address "MyStructArray[0].MySubStruct.Data" would be represented in the following groups: "Global," "MYSTRUCTARRAY[x]," "MYSTRUCTARRAY[0]," and "MYSUBSTRUCT". The tag "DATA" would be in the last group. The static reference to "DATA" would be "Channel1.Device1.Global.MYSTRUCTARRAY[x].MYSTRUCTARRAY[0].MYSUBSTRUCT.DATA". The dynamic reference would be "Channel1.Device1.MyStructArray[0].MySubStruct.Data".

For more information, refer to "Static Tags (User-Defined)" and "Dynamic Tags" in server help file.

Condensed Mode

In Condensed Mode, the automatically generated server tags follow a group / tag hierarchy consistent with the tag's address. Groups are created for each segment that precedes the period. Groups created include the following:

- Program Scope
- Structures and Substructures

Notes:

1. Groups are not created for arrays or .bit addresses.
2. Tag or structure member names that start with an underscore are converted to "U". This is required because the server does not support leading underscores in tag name fields.

Simple Example

Name	Value	Force Mask	Style	Data Type
MyTag	{...}	{...}		MyDataType
MyTag.Member1	{...}	{...}	Decimal	DINT[10]
MyTag.Member1[0]	0		Decimal	DINT
MyTag.Member1[1]	0		Decimal	DINT
MyTag.Member1[2]	0		Decimal	DINT
MyTag.Member1[3]	0		Decimal	DINT

Projekt	Logix-Adresse /	Logix-Datentyp	Zugriffstyp	Beschreibung
Konnektivität	Mitglied1	DINT	Schreibgeschützt	Mitglied1
Kanal	Mitglied2	DINT	Schreibgeschützt	Mitglied2
Gerät	Mitglied3	DINT	Schreibgeschützt	Mitglied3
MeinEtikett	Mitglied4	DINT	Schreibgeschützt	Mitglied4
	Mitglied5	DINT	Schreibgeschützt	Mitglied5
	Mitglied6	DINT	Schreibgeschützt	Mitglied6
	Mitglied7	DINT	Schreibgeschützt	Mitglied7
	Mitglied8	DINT	Schreibgeschützt	Mitglied8
	Mitglied9	DINT	Schreibgeschützt	Mitglied9

Complex Example

A Logix Address defined with address "MyStructArray[0].MySubStruct.Data" would be represented in the following groups: "MYSTRUCTARRAY[0]" and "MYSUBSTRUCT". The tag "DATA" would be in the last group. The static reference to "DATA" would be "Channel1.Device1.MYSTRUCTARRAY[0].MYSUBSTRUCT.DATA" and the dynamic reference would be "Channel1.Device1.MyStructArray[0].MySubStruct.Data".

Geräteigenschaften - Optionen

Eigenschaftengruppen	Optionen	
Allgemein	OPC-Qualität schlecht bis Schreiben	Deaktivieren
Scan-Modus	Zeichenfolgen bündeln	Deaktivieren
Tag-Generierung		
Controller-Modul		
Native Tag-Datenbank		
Optionen		

OPC-Qualität schlecht bis Schreiben: Wählen Sie "Aktivieren" aus, um den Treiber dazu zu zwingen, schlechte Qualität zurückzugeben, bis ein Schreibvorgang in das native Tag erfolgt. Der Schreibvorgang kann ggf. über eine Client-Schnittstelle (wie OPC) oder ein ControlLogix-Client-Gerät (wie ein SPS der ControlLogix 5000-Reihe) erfolgen. Wenn ein Schreibvorgang in ein einzelnes Element-Array erfolgt, wird das gesamte Array initialisiert und gute Qualität zurückgegeben. Die Standardeinstellung ist deaktiviert. Beim Start initialisiert Allen-Bradley ControlLogix Server-Treiber Werte von Ganzzahl- bzw. numerischen Datentypen auf Null (0) und Zeichenfolgen auf "Leer". Clients erhalten standardmäßig Erstaktualisierungen mit guter Qualität; dieses Verhalten kann jedoch für jedes Gerät geändert werden.

Zeichenfolgen bündeln: Diese Option ist für alle im Zeichenfolgen-Tag angezeigten Zeichenfolgen aktiviert, um alle Byte der Array-Elemente in gepackter Form einzuschließen. Ist diese Option deaktiviert, liegt die im Zeichenfolgen-Tag angezeigte Zeichenfolge in einem ungepackten Format vor, wobei nur die Low-Byte jedes Elements angezeigt werden.

ControlLogix-Client-Gerätekonfiguration

ControlLogix-SPS von Allen-Bradley müssen so programmiert werden, dass Meldungen zum Lesen/Schreiben von CIP-Datentabellen mit der MSG-Kontaktplananweisung an diesen Treiber ausgegeben werden. Es sollte ein Routing-Pfad, der die Konfiguration des Treibers darstellt, verwendet werden. Dazu gehört die IP-Adresse, Steckplatznummer und ein optionaler Port. *Weitere Informationen zur MSG-Kontaktplananweisung finden Sie in der Dokumentation zur SPS-Programmierung von Rockwell/Allen-Bradley.* Der Routing-Pfad, der einem bestimmten ControlLogix-Servergerät zugeordnet ist, wird im Controller-Modul unter "Geräteigenschaften" angegeben. *Weitere Informationen dazu finden Sie unter [Geräteigenschaften - Controller-Modul](#).*

Unterstützte Dienste

- Nicht fragmentiertes Lesen
- Fragmentiertes Lesen
- Nicht fragmentiertes Schreiben
- Fragmentiertes Schreiben
- Lesen/Bearbeiten/Schreiben

● **Hinweis:** Die MSG-Kontaktplananweisung von ControlLogix entscheidet automatisch, ob je nach der Größe der Anforderung fragmentierte oder nicht fragmentierte Dienste verwendet werden sollen. Dies ist keine vom Benutzer zu konfigurierende Option.

Unterstützte Logix-Typen

- BOOL
- DWORD (BOOL-Array)
- SINT
- INT
- DINT
- LINT
- REAL

● *Weitere Informationen zu unterstützten Logix-Typen finden Sie unter [Adressbeschreibungen](#).*

Fehlercodes

Allen-Bradley ControlLogix Server-Treiber antwortet auf alle korrekt formatierten, empfangenen Meldungen. Wenn die Anforderung nicht abgeschlossen werden kann, wird eine Antwortmeldung mit einem Fehlerstatus ungleich Null und ein optionaler, erweiterter Fehlerstatus in den ERR- und EXERR-Tags der MESSAGE-Struktur zurückgegeben. Kontaktpläne müssen zur Verarbeitung dieser Fehler geschrieben werden.

● *Weitere Informationen zu den Fehlercodes, die an ControlLogix-Client-Geräte zurückgegeben werden können, finden Sie unter [Fehlercodes](#).*

● **Hinweis:** Dieser Treiber unterstützt CIP-Datentabellen-Lesevorgänge und -Schreibvorgänge für die oben aufgelisteten unteilbaren Logix-Typen. Zwar werden strukturierte Typen nicht unterstützt, doch kann die MSG-Kontaktplananweisung zum Schreiben/Lesen von/in einzelne(n) unteilbaren Logix-Typen innerhalb eines strukturierten Typs verwendet werden. Beispiel: Wenn das Tag "MyString @ STRING" in den Treiber geschrieben werden muss, muss ein CIP-Datentabellen-Lesevorgang für "MyString.DATA" und "MyString.LEN" separat durchgeführt werden.

Logix Config Object

When a device is created, an object named Logix Config is created. Logix Config is also referred to as the Native Tag Database. This object defines the address space of the device, defining what addresses will be accepted for incoming Logix or OPC clients. The objects that represent this namespace are referred to as Logix Addresses or Native Tags throughout this help file. Logix Addresses may be created individually or imported in bulk from a CSV file.

Notes:

1. The total memory size of a Native Tag (Logix Address) must not exceed 128 kB.
2. The total memory size of a Native Tag Database (Logix Config) must not exceed 512 kB.

Logix Config CSV Import

The device address space may be configured by importing Logix Addresses from a CSV file. To import a CSV file, right click **Logix Config** and select **Import CSV**.

A CSV file specifies the Native Tags that each device can represent (its address space). It is used once for tag import and is not required for automatic tag database generation or remote deployment. When a CSV import is triggered by the user, all existing Logix Addresses in the Logix Config object will be deleted and replaced with the imported Logix Addresses. After the Logix Addresses have been imported, an ATG will be triggered creating static tags that represent the address space of the device. The following CSV header must be used for Native Tag import:

Logix Address	Logix DataType	External Access	Description
---------------	----------------	-----------------	-------------

Definitions and requirements for each column in the CSV file are discussed below.

Tip: A template Native Tag Database CSV file is included for reference in `<Server Installation Directory>/Drivers/controllogix_unsolicited_ethernet/import_template.csv`.

Importing Native Tags as SINT, INT, and DINT Arrays

Native Tags that are imported as SINT, INT, and DINT arrays also have a string tag defined that uses the number of elements of the corresponding array in the tag address. Examples are as follows:

- If a Native Tag called "MySINTarray @ SINT[100]" is imported, a Static Tag with the address "MYSINTARRAY / 100" and String data type is generated.
- If a Native Tag called "MyINTarray @ INT[100]" is imported, a Static Tag with the address "MYINTARRAY / 100" and String data type is generated.
- If a Native Tag called "MyDINTarray @ DINT[100]" is imported, a Static Tag with the address "MYDINTARRAY / 100" and String data type is generated.

Tip: To import RSLogix5000 pre-defined Strings, the two elements contained within the String types ("STRING.DATA" and "STRING.LEN") should be defined in the Native Tag Database CSV file before the import is performed.

Logix Address

Restrictions on the Logix Address are consistent with RSLogix5000 requirements, which correspond to the following IEC 1131-3 identifier rules:

- Must begin with an alphabetic character (A-Z, a-z) or an underscore.
- Can only contain alphanumeric characters and underscores.
- Can have as many as 40 characters in each segment.
- Cannot have consecutive underscores.
- Are not case sensitive.

Tags that do not have a unique Logix Address or meet the identifier requirements above fail to import, causing a message with the specified Logix Address to be posted to the OPC server's Event Log.

Logix DataType

The following pre-defined Logix Atomic Types are supported:

Logix Data Type	Supported Data Types
BOOL	Boolean
SINT	Char, Byte
INT	Short, Word
DINT	Long, DWord
LINT	Double, Date
REAL	Float

Array Syntax: When defining the data type, up to three dimensional arrays are allowed for most types. Array syntax is specified using brackets "[]" (for example a DataType of DINT[5,5] specifies a two-dimensional array with five elements in each dimension).

Special Rules: The following rules apply to array data types:

1. Boolean arrays are limited to a single dimension.
2. Boolean arrays must be DWORD-aligned; the dimension size must be a multiple of 32 bits.
3. The total memory size of a Native Tag (Logix Address) must not exceed 128 kB.
4. The total memory size of a Native Tag Database (Logix Config) must not exceed 512 kB.

Note: Other pre-defined or user-defined complex (structured) data types are not supported. Structured data can be imported by qualifying the Logix Address down to the atomic type. For example, there exists a structured type called TIME, which is described as the following:

```
TIME
{
  HOUR @ SINT
  MIN @ SINT
  SEC @ SINT
}
```

The structure can be broken down and the atomic member imported as "TIME.HOUR," "TIME.MIN," and "TIME.SEC" with the associated Logix data type, External Access, and Description following the CSV format outlined above. All unsupported Logix data type values specified in the CSV import are defaulted to DINT so that the import succeeds.

Access Type

The Access Type specifies the ControlLogix client devices' read / write privileges. This access does not apply to OPC client tags, which always have a client access of read / write by default (this setting is configurable using static OPC server tags). The following external access types are supported: all other values specified are set to read / write.

- **R/W**: ControlLogix client devices have Read / Write permissions to the Native Tag.
- **RO**: ControlLogix client devices have Read Only permissions. All write attempts fail with the appropriate error (CIP error 0x0F).

Description

Descriptions are used during automatic tag database generation, truncated to 64 characters. The Description field must be present, but may be left blank.

Defining New Logix Addresses

To define a new Logix Address, right-click **Logix Config** in the Tree View and select **New Logix Address**. Alternatively, select **Logix Config** in the Tree View, right-click the Detail View, then select **New Logix Address**.

Defining Logix Address Properties

Eigenschaftengrupp... Allgemein	[-] Identifikation	
	Beschreibung	
	[-] Definition der Logix-Adresse	
	Logix-Adresse	
	Logix-Datentyp	DINT
	Zugriffstyp	Schreibgeschützt

Description: This property provides a brief summary of the object or its use.

Logix Address: This property specifies the address for the Logix Address. See [CSV Import](#) or [Logix Tag-Based Addressing](#) for more information about valid Logix addresses.

Logix DataType: This property specifies the data type for the Logix Address. See [CSV Import](#) for more information about valid Logix data types.

External Access: This property specifies whether the tag has Read-Only or Read / Write access. After creating a new Logix Address, it will appear in the tag list in the Detail View

Data Types Description

The following describes server data types supported by the Allen-Bradley ControlLogix Server-Treiber when defining static tags. For data types allowed when defining the address space of the device, see [Logix Tag-Based Addressing](#).

Data Types	Description
Boolean	Single bit
Char	Signed 8-bit value bit 0 is the low bit bit 6 is the high bit bit 7 is the sign bit
Byte	Unsigned 8-bit value bit 0 is the low bit bit 7 is the high bit
Short	Signed 16-bit value bit 0 is the low bit bit 14 is the high bit bit 15 is the sign bit
Word	Unsigned 16-bit value bit 0 is the low bit bit 15 is the high bit
Long	Signed 32-bit value bit 0 is the low bit bit 30 is the high bit bit 31 is the sign bit
DWord	Unsigned 32-bit value bit 0 is the low bit bit 31 is the high bit
Float	32-bit floating point value bit 0 is the low bit bit 31 is the high bit
Double	64-bit floating point value bit 0 is the low bit bit 63 is the high bit
String	Typically null-terminated, null padded, or blank padded ASCII string.
Date	64-bit floating point value.

● For a description of Logix platform-specific data types, refer to [Address Descriptions](#).

Adressbeschreibungen

Allen-Bradley ControlLogix Server-Treiber unterstützt auf symbolischen Tags basierte Adressierung.

Auf Logix-Tags basierte Adressierung

Dieser Treiber verwendet eine Tag- oder symbolbasierte Adressstruktur, die im Allgemeinen als Logix- oder native Tags bezeichnet werden (kompatibel mit Integrated Architecture von Rockwell Automation). Diese Tags unterscheiden sich von herkömmlichen SPS-Datenelementen dahingehend, dass der Tag-Name die Adresse ist und keine physische oder logische Adresse.

Der Treiber ermöglicht Benutzern den Zugriff auf die unteilbaren Datentypen BOOL, SINT, INT, DINT, LINT und REAL des Controllers. Zwar handelt es sich bei einigen der vordefinierten Typen um Strukturen, doch basieren sie letztendlich auf diesen unteilbaren Datentypen. Dementsprechend sind alle nicht strukturierten (unteilbaren) Mitglieder einer Struktur zugänglich. Beispiel: Ein TIMER (ZEITGEBER) kann keinem OPC-Server-Tag zugewiesen werden, wohingegen ein unteilbares TIMER-Mitglied zugewiesen werden kann (beispielsweise TIMER.EN, TIMER.ACC usw.). Wenn es sich bei einem Strukturmitglied um eine Struktur selbst handelt, müssen beide Strukturen entsprechend erweitert werden, damit auf ein unteilbares Mitglied der Substruktur zugegriffen werden kann. Dies ist üblicherweise bei benutzer- und moduldefinierten Typen und bei keinem der vordefinierten Typen zu finden.

Unteilbarer Datentyp	Beschreibung		Bereich
BOOL	Einzel-Bit-Wert	VT_BOOL	0, 1
SINT	8-Bit-Wert mit Vorzeichen	VT_I1	-128 bis 127
INT	16-Bit-Wert mit Vorzeichen	VT_I2	-32.768 bis 32.767
DINT	32-Bit-Wert mit Vorzeichen	VT_I4	-2.147.483.648 bis 2.147.483.647
LINT	64-Bit-Wert mit Vorzeichen	VT_R8	-9.22337E18 bis 9.22336E18
REAL	32-Bit-IEEE-Gleitkommazahl	VT_R4	1.1755 E-38 bis 3.403E38 0 -3.403E-38 bis -1.1755

Tag-Adressregeln für OPC-Client/Server

Logix-Adressen entsprechen Tag-Adressen für OPC-Client/Server. Für Logix-Adressen, die über RSLogix5000 eingegeben werden, gelten die IEC 1131-3-ID-Regeln. Für Tag-Adressen für OPC-Client/Server gelten dieselben Regeln. Die folgenden Symbole werden verwendet:

- Müssen mit einem Buchstaben (A-Z, a-z) oder Unterstrich beginnen.
- Dürfen nur Buchstaben und Unterstriche enthalten.
- Dürfen in jedem Segment bis zu 40 Zeichen lang sein.
- Dürfen keine aufeinander folgenden Unterstriche enthalten.
- Dabei wird die Groß-/Kleinschreibung nicht beachtet.

Hinweise

1. Die Zuweisung des Tag-Namens im OPC-Server unterscheidet sich von der Adresszuweisung dahingehend, dass Namen nicht mit einem Unterstrich beginnen dürfen.
2. Damit Tags ordnungsgemäß validiert werden, müssen native Tags, die für das statische OPC-Client-Tag stehen, in der nativen Tag-Datenbank vorhanden sein.

Adressformate

Es gibt mehrere Möglichkeiten, Logix-Adressen statisch im OPC-Server oder dynamisch von einem OPC-Client aus zu adressieren. Das ausgewählte Format hängt vom Typ des Tags und seiner Verwendung ab. Beispiel: Das Bit-Format würde beim Zugriff auf ein Bit innerhalb eines Tags des SINT-Typs verwendet werden.

• *Weitere Informationen zum Adressformat und zur Syntax finden Sie in der Tabelle unten.*

• **Hinweis:** Alle Formate sind für RSLogix5000 nativ, und zwar mit Ausnahme von Array und String. Beim Referenzieren eines unteilbaren Datentyps können Benutzer einen RSLogix5000-Tag-Namen kopieren und in den Parameter "Tag-Adresse" des OPC-Servers einfügen: Er ist gültig, solange das entsprechende native Tag in der nativen Tag-Datenbank vorhanden ist.

Format	Syntax	Beispiel	Hinweise
Standard	<Logix-Tag-Name>	tag_1	Das Tag kann kein Array sein.
Array-Element	<Tag-Name für Logix-Array> [dim1, dim2, dim3]	tag_1 [2, 58, 547] tag_1 [0, 3]	Dimensionsbereich = 1 bis 3. Elementbereich = 0 bis 65535.
Array ohne Offset*	<Tag-Name für Logix-Array> {Spaltenanzahl} <Tag-Name für Logix-Array> {Zeilenanzahl} {Spaltenanzahl}	tag_1 {8} tag_1 {2} {4}	Dimensionsbereich = 1 bis 2. Elementbereich = 1 bis 65535. Die Anzahl der Elemente zum Lesen/Schreiben ist gleich der Zeilenanzahl multipliziert mit der Spaltenanzahl. Wenn keine Zeilen angegeben werden, liegt die Zeilenanzahl standardmäßig bei 1. Das Array beginnt bei einem Null-Offset (Array-Index gleich 0 für alle Dimensionen).
Array mit Offset*	<Logix-Array-Element-Tag> {Spaltenanzahl} <Logix-Array-Element-Tag> {Zeilenanzahl}{Spaltenanzahl}	tag_1 [2, 3] {10} tag_1 [2, 3] 2}{5}	Das Array beginnt bei einem durch die Dimensionen im Array-Element-Tag festgelegten Offset. Das Array deckt immer die höchste Dimension ab. Somit würde "tag_1[2,3]{10}" ein Array von Elementen tag_1[2,3] -> tag_1[2,13] erzeugen.
Bit	<Logix-Tag-Name>.bit <Logix-Tag-Name>.[Bit]	tag_1.0 tag_1.[0]	Bit-Bereich = 0 bis 31. Wenn das Tag ein Array ist, muss es ein BOOL-Array sein; andernfalls kann das Tag kein Array sein.
String	<Logix-Tag-Name>.Data/<Maximale Zeichenfolgenlänge>	tag_1.Data/4	Längenbereich = 1 bis 65535. Die maximale Anzahl Zeichen, die ein Lesen/Schreiben in der Zeichenfolge ermöglichen.

*Da dieses Format möglicherweise mehr als ein Element anfordert, hängt die Reihenfolge, in der Array-Daten weitergeleitet werden, von der Dimension des Logix-Array-Tags ab. Beispiel: Wenn die Zeilen multipliziert mit den Spalten 4 ergeben und das Controller-Tag ein 3X3-Element-Array ist, handelt es sich bei den Elementen, die referenziert werden, um array_tag [0,0], array_tag [0,1], array_tag [0,2] und array_tag [1,0] in dieser Reihenfolge. Die Ergebnisse wären unterschiedlich, wenn es sich beim Controller-Tag um ein 2X10-Element-Array handeln würde.

• *Weitere Informationen zum Referenzieren von Elementen für Arrays mit 1, 2 und 3 Dimensionen finden Sie unter [Reihenfolge von Logix-Array-Daten](#).*

Tag Scope

Global Tags

Global Tags are Logix Addresses that have global scope in the controller. Any program or task can access Global Tags; however, the number of ways a Global Tag can be referenced depends on its Logix data type and the address format being used.

Program Tags

Program Tags are identical to Global Tags except that their scope is local to the program in which it is defined. The driver does not currently support importing Native Tags with a program designation.

Structure Tag Addressing

Logix Structure Tags are tags with one or more member tags (which can be atomic or structured).

`<structure name> . <atomic-type tag>`

This implies that a substructure would be addressed as the following:

`<structure name> . <substructure name> . <atomic-type tag>`

Arrays of structures would be addressed as the following:

`<structure array name> [dim1, dim2, dim3]. <atomic-type tag>`

This implies that an array of substructures would be addressed as the following:

`<structure name> . <substructure array name> [dim1, dim2, dim3]. <atomic-type tag>`

Note: The examples above display a few of the addressing possibilities that involve structures and are only provided as an introduction to structure addressing. *For more information, refer to Rockwell/Allen-Bradley documentation.*

Erweiterte Adressierung

Benutzer haben mehrere Optionen für die symbolische Adressierung, die in der Adresse mit symbolischen Tags enthalten sein können. Die folgenden Einschränkungen wurden dem Datentyp für die Bit- und Array-Adressierungssyntax auferlegt:

- Was die Bit-Syntax angeht, darf der Index die Bit-Größe für den Datentyp nicht überschreiten. Beispiel: "MyDint @ Dint" wird als natives Tag importiert. Der Bit-Index darf 31 nicht überschreiten (da DINTs 32-Bit-Werte mit Vorzeichen sind).
- Was die Array-Syntax angeht, dürfen das Offset und die Anzahl der Elemente im Array die Anzahl der Elemente im zugeordneten nativen Tag nicht überschreiten. Beispiel: "MyDintArray @ DINT[10]" wird als natives Tag importiert. Ein statisches Tag mit den Adressen "MYDINTARRAY[0]{5}" und "MYDINTARRAY[4]{5}" ist gültig, weil in den Arrays jeweils die ersten und letzten 5 Elemente des nativen Tags enthalten sind. Ein statisches Tag mit der Adresse "MYDINTARRAY[5]{10}" ist ungültig, weil das Tag nach 10 DINTs fragt, die bei Offset 5 beginnen und das native Tag-Array nicht so groß ist.

• Weitere Informationen zu erweiterten Themen finden Sie in der Tabelle unten.

Element	Syntax	Beispiel	Hinweise
Standard	<code><Tag-Name></code>	tag_1	k.A.
Array ohne Offset	<code><Array-Tag-Name> {Spaltenanzahl}</code>	tag_1 {8} tag_1 {2} {4}	Die Anzahl der Elemente zum Lesen/Schreiben ist gleich der Zeilenanzahl multipliziert mit der Spaltenanzahl. Wenn keine Zeilen angegeben werden, liegt die Zeilenanzahl standardmäßig bei 1. Min-

Element	Syntax	Beispiel	Hinweise
	<Array-Tag-Name> {Zeilenanzahl} {Spaltenanzahl}		destens 1 Element des Arrays muss adressiert werden. Das Array beginnt bei einem Null-Offset (Array-Index gleich 0 für alle Dimensionen).
Array mit Offset	<Array-Element-Tag> [Offset] {Spaltenanzahl} <Array-Element-Tag> [Offset]{Zeilenanzahl} {Spaltenanzahl}	tag_1 [5] {8} tag_1[5] {2} {4}	Die Anzahl der Elemente zum Lesen/Schreiben ist gleich der Zeilenanzahl multipliziert mit der Spaltenanzahl. Wenn keine Zeilen angegeben werden, liegt die Zeilenanzahl standardmäßig bei 1. Mindestens 1 Element des Arrays muss adressiert werden. Das Array beginnt bei einem Null-Offset (Array-Index gleich 0 für alle Dimensionen).
Bit	<Tag-Name> . bit <Tag-Name> . [Bit]	tag_1 . 0 tag_1 . [0]	k.A. k.A.
Zeichenfolge	<Tag-Name> / <Elementanzahl>	tag_1 / 4	Die Anzahl der Elemente muss mindestens 1 sein. Die Anzahl der Zeichen zum Lesen/Schreiben ist von der Eigenschaft "Zeichenfolgen bündeln" abhängig.*

*Bei Aktivierung entspricht die Zeichenanzahl der Elementanzahl multipliziert mit der Elementgröße (4 Elemente eines INT-Arrays weisen auf 8 Zeichen hin). Bei Deaktivierung entspricht die Zeichenanzahl der Elementanzahl (4 Elemente eines INT-Arrays weisen auf 4 Zeichen hin).

• Weitere Informationen dazu finden Sie unter [Optionen](#).

Reihenfolge von Logix-Array-Daten

Da native Tags bis zu dreidimensionale Arrays unterstützen, wird die Reihenfolge der Logix-Array-Daten einem zweidimensionalen OPC-Array zugeordnet.

1. Dimensionale Arrays – array [dim1]

1-dimensionale Array-Daten werden dem bzw. vom Controller in aufsteigender Reihenfolge weitergeleitet.
for (dim1 = 0; dim1 < dim1_max; dim1++)

Beispiel: Array mit 3 Elementen

```
array [0]
array [1]
array [2]
```

2. Dimensionale Arrays – array [dim1, dim2]

2-dimensionale Array-Daten werden dem bzw. vom Controller in aufsteigender Reihenfolge weitergeleitet.
for (dim1 = 0; dim1 < dim1_max; dim1++)
for (dim2 = 0; dim2 < dim2_max; dim2++)

Beispiel: Array mit 3x3 Elementen

```
array [0, 0]
array [0, 1]
array [0, 2]
array [1, 0]
array [1, 1]
```

```
array [1, 2]
array [2, 0]
array [2, 1]
array [2, 2]
```

3. Dimensionale Arrays - array [dim1, dim2, dim3]

3-dimensionale Array-Daten werden dem bzw. vom Controller in aufsteigender Reihenfolge weitergeleitet.

```
for (dim1 = 0; dim1 < dim1_max; dim1++)
for (dim2 = 0; dim2 < dim2_max; dim2++)
for (dim3 = 0; dim3 < dim3_max; dim3++)
```

Beispiel: 3X3x3 Element-Arrays

```
array [0, 0, 0]
array [0, 0, 1]
array [0, 0, 2]
array [0, 1, 0]
array [0, 1, 1]
array [0, 1, 2]
array [0, 2, 0]
array [0, 2, 1]
array [0, 2, 2]
array [1, 0, 0]
array [1, 0, 1]
array [1, 0, 2]
array [1, 1, 0]
array [1, 1, 1]
array [1, 1, 2]
array [1, 2, 0]
array [1, 2, 1]
array [1, 2, 2]
array [2, 0, 0]
array [2, 0, 1]
array [2, 0, 2]
array [2, 1, 0]
array [2, 1, 1]
array [2, 1, 2]
array [2, 2, 0]
array [2, 2, 1]
array [2, 2, 2]
```

Error Codes

The Allen-Bradley ControlLogix Server-Treiber may return the following error codes.

• For more information on a specific type of error code, select a link from the list below.

[EtherNet/IP Encapsulation Error Codes](#)

[CIP Error Codes](#)

[0x01 Extended Error Codes](#)

[0xFF Extended Error Codes](#)

EtherNet/IP Encapsulation Error Codes

The Allen-Bradley ControlLogix Server-Treiber may return the following error codes.

• **Note:** The error codes are in hexadecimal.

Error	Description
0001	Command not handled
0002	Memory not available for command
0003	Poorly formed or incomplete data
0064	Invalid Session ID
0065	Invalid length in header
0069	Requested protocol version not supported

CIP Error Codes

The error codes are in hexadecimal.

Error	Description
01	Connection failure*
02	Insufficient resources
03	Parameter value invalid
04	IOI could not be deciphered or tag does not exist
05	Unknown destination
06	Data requested would not fit in response packet
08	Unsupported service
0F	Permission denied
13	Insufficient command data / parameter specified to execute service
26	The number of IOI words specified does not match IOI word count
FF	General Error**

• ***See Also:** [0x01 Extended Error Codes](#)

• ****See Also:** [0xFF Extended Error Codes](#)

0x01 Extended Error Codes

The Allen-Bradley ControlLogix Server-Treiber may return the following extended errors for CIP error 0x01.

● **Note:** The error codes are in hexadecimal.

Error	Description
0x0205	Unconnected Send parameter error.
0x0312	Link address is not available.
0x0318	Link address to self is invalid.

0xFF Erweiterte Fehlercodes


Allen-Bradley ControlLogix Server-Treiber gibt möglicherweise die folgenden erweiterten Fehlercodes für den CIP-Fehler 0xFF zurück.

● **Hinweis:** Die Fehlercodes liegen in hexadezimaler Schreibweise vor.

Fehler	Beschreibung
2104	Adresse außerhalb des zulässigen Bereichs.
2105	Versuch, über das Ende des Datenobjekts hinaus zuzugreifen.
2107	Der Datentyp ist ungültig oder wird nicht unterstützt.

Event Log Messages

The following information concerns messages posted to the Event Log pane in the main user interface. Consult the OPC server help on filtering and sorting the Event Log detail view. Server help contains many common messages, so should also be searched. Generally, the type of message (informational, warning) and troubleshooting information is provided whenever possible.

 **Tip:** Messages that originate from a data source (such as third-party software, including databases) are presented through the Event Log. Troubleshooting steps should include researching those messages online and in vendor documentation.

Fehler beim Importieren der nativen Tag-Datenbank. Datei kann nicht geöffnet werden. | Betriebssystemfehler = '<Fehler>'.

Fehlertyp:

Fehler

Fehler beim Importieren der nativen Tag-Datenbank. Datei kann nicht geöffnet werden, da ein allgemeiner Lesefehler aufgetreten ist.

Fehlertyp:

Fehler

Mögliche Ursache:

Die Datei fehlt, ist fehlerhaft oder nicht ordnungsgemäß formatiert.

Mögliche Lösung:

Suchen Sie die Tag-Datenbank-Datei, und überprüfen Sie Header-Zeile, Format und Speicherort, bevor Sie den Vorgang erneut durchführen.

Fehler beim Importieren der nativen Tag-Datenbank. Dateikodierung wird nicht unterstützt.

Fehlertyp:

Fehler

Mögliche Ursache:

Die importierte CSV-Datei verwendet nicht unterstützte Dateikodierung.

Mögliche Lösung:

Aktualisieren Sie die CSV-Datei, um die ANSI- oder UTF-8-Kodierungsmethode zu verwenden.

Fehler beim Importieren der nativen Tag-Datenbank.

Fehlertyp:

Fehler

Mögliche Ursache:

Es ist ein unerwarteter Fehler aufgetreten.

Mögliche Lösung:

Vergewissern Sie sich, dass die importierte CSV-Datei ordnungsgemäß formatiert ist.

Fehler beim Importieren der nativen Tag-Datenbank. Feldname wurde nicht erkannt. | Nicht erkannter Feldname = '<Feldname>'.

Fehlertyp:

Fehler

Mögliche Ursache:

Die CSV-Datei definiert ein Feld, das beim Importieren der nativen Tag-Datenbank nicht unterstützt wird.

Mögliche Lösung:

Vergewissern Sie sich, dass in der CSV-Datei keine unbeabsichtigten Felder vorhanden sind.

● Hinweis:

Zu den unterstützten Feldnamen zählen 'Logix Address', 'Logix DataType', 'External Access' und 'Description'.

Fehler beim Importieren der nativen Tag-Datenbank. Feldname ist doppelt vorhanden. | Doppelter Feldname = '<Feld>'.

Fehlertyp:

Fehler

Mögliche Ursache:

Die importierte CSV-Datei enthält mehrere Definitionen des gleichen Feldnamens.

Mögliche Lösung:

Entfernen oder korrigieren Sie doppelte Felder in der CSV-Datei. Zu den unterstützten Feldnamen zählen 'Logix Address', 'Logix Data Type', 'External Access' und 'Description'.

Fehler beim Importieren der nativen Tag-Datenbank. Tag-Feld-Identifikations-Datensatz fehlt.

Fehlertyp:

Fehler

Mögliche Ursache:

Die importierte CSV-Datei enthält keinen Header.

Mögliche Lösung:

Vergewissern Sie sich, dass keine Felder in der CSV-Datei fehlen, und bestätigen oder korrigieren Sie die Header-Zeile.

● Hinweis:

Zu den unterstützten Feldnamen zählen 'Logix Address', 'Logix Data Type', 'External Access' und 'Description'.

Fehler beim Importieren der nativen Tag-Datenbank. Tag-Feld-Identifikations-Datensatz ist unvollständig.

Fehlertyp:

Fehler

Mögliche Ursache:

Die importierte CSV-Datei enthält keinen vollständigen und gültigen Header.

Mögliche Lösung:

Vergewissern Sie sich, dass keine Felder in der CSV-Datei fehlen, und bestätigen oder korrigieren Sie die Header-Zeile.

● Hinweis:

Zu den unterstützten Feldnamen zählen 'Logix Address', 'Logix Data Type', 'External Access' und 'Description'.

Fehler beim Starten des unangeforderten Logix-Servers.

Fehlertyp:

Fehler

Mögliche Ursache:

Der Treiber konnte keine Bindung und Überwachung an der angegebenen IP bzw. dem Port durchführen.

Mögliche Lösung:

Vergewissern Sie sich, dass der angegebene Port (TCP oder UDP) nicht von einer anderen Anwendung verwendet wird, und beheben Sie mögliche Konflikte.

Ungültige Tag-Adresse. Natives Tag wurde nicht importiert. | Ungültige Adresse = '<Adresse>'.

Fehlertyp:

Warnung

Zwei Kanäle wurden für die Verwendung derselben Netzwerkadapter-IP bzw. desselben TCP-Ports konfiguriert. Jeder Kanal muss an eine eindeutige lokale IP bzw. einen eindeutigen Port gebunden sein. | Erster Kanal = '<Kanal>', zweiter Kanal = '<Kanal>'.

Fehlertyp:

Warnung

Zwei Geräte wurden für die Verwendung des gleichen Pfads vom EtherNet/IP-Modul konfiguriert. Jeder Kanal muss einen eindeutigen Pfad vom EtherNet/IP-Modul aufweisen. | Erstes Gerät = '<Adresse>', zweites Gerät = '<Adresse>'.

Fehlertyp:

Warnung

Mögliche Ursache:

Das Gerät ist für einen CPU-Typ konfiguriert, und die Steckplatznummer wird bereits verwendet.

Mögliche Lösung:

1. Wählen Sie einen anderen CPU-Typ (lokal oder remote) aus.
2. Wählen Sie eine andere Steckplatznummer aus, wenn das Gerät als ein Remote-Controller-Modul konfiguriert wurde.

Ungültige Tag-Adresse. Doppelte Tag-Adressen sind nicht zulässig. | Ungültige Adresse = '<Adresse>'.

Fehlertyp:

Warnung

Mögliche Ursache:

Die importierte CSV-Datei enthält einen oder mehrere Tags mit der gleichen Logix-Adresse.

Mögliche Lösung:

Entfernen Sie in der importierten CSV-Datei der nativen Tag-Datenbank die nativen Tags, die eine doppelte Logix-Adresse aufweisen.

Speicherplatz für Tag konnte nicht zugeordnet werden. | Tag-Adresse = '<Adresse>'.

Fehlertyp:

Warnung

Mögliche Ursache:

Die zum Erstellen eines Tags erforderlichen Ressourcen konnten nicht zugeordnet werden. Das Tag wurde nicht zum Projekt hinzugefügt.

Mögliche Lösung:

Schließen Sie nicht verwendete Anwendungen, und/oder erhöhen Sie den virtuellen Arbeitsspeicher. Versuchen Sie es anschließend erneut.

Ungültiges natives Tag. Größe der einzelnen Tags ist auf 128 KB begrenzt. | Tag-Adresse = '<Adresse>'.

Fehlertyp:

Warnung

Mögliche Ursache:

Die importierte CSV-Datei enthält eine einzelne Tag-Definition, die mehr als 128 KB Speicher für die Darstellung erfordert, beispielsweise in einem großen Array. Native Tags, die mehr als 128 KB Speicher erfordern, können nicht definiert werden.

Mögliche Lösung:

Verringern Sie die Array-Größe, oder entfernen Sie das Tag aus der zu importierenden CSV-Datei.

Fehler beim Importieren des nativen Tags. Gesamtgröße der Datenbank-Tag-Daten ist auf 512 KB begrenzt. | Tag-Adresse = '<Adresse>'.

Fehlertyp:

Warnung

Mögliche Ursache:

Die importierte CSV-Datei enthält Tag-Definitionen, die mehr als 512 KB Speicher erfordern, beispielsweise in einem großen Array.

Mögliche Lösung:

Wenn mehr als 512 KB Speicher erforderlich sind, erstellen Sie ein neues Gerät, und verteilen Sie die Tag-Datenbank auf mehrere Geräte.

Der angegebene TCP/IP-Port liegt außerhalb des zulässigen Bereichs. Der Standard-Port wird verwendet. | Gültiger Bereich = <Nummer> bis <Nummer>, Standard-Port = <Nummer>.

Fehlertyp:

Warnung

Mögliche Ursache:

Ein Projekt wurde geladen, für das ein TCP/IP-Port von 0 angegeben wurde, aber diese Option wird nicht unterstützt.

Mögliche Lösung:

Verwenden Sie den Standard-Port (44818), oder wählen Sie einen Port in einem gültigen Bereich (1 bis 65535) aus.

Ein weiteres Gerät im Kanal ist bereits als lokale CPU registriert. | Gerät = <Gerät>.

Fehlertyp:

Warnung

CSV-Import fehlgeschlagen. Vorhandene Logix-Adressen können nicht gelöscht werden.

Fehlertyp:

Warnung

CSV-Import fehlgeschlagen. Es können keine neuen Logix-Adressen erstellt werden.

Fehlertyp:

Warnung

Unerwartetes Fragment beim Verarbeiten fragmentierter Schreibvorgänge entdeckt. | Logix-Adresse = <Tag>.

Fehlertyp:

Warnung

Native Tags importiert. | Tag-Anzahl = <Anzahl>, Tag-Datenbank-Pfad = '<Pfad>'.

Fehlertyp:

Informationen

Tags generiert. | Tag-Anzahl = <Anzahl>, Tag-Hierarchie-Modus = '<Modus>'.

Fehlertyp:

Informationen

Aufgrund geringer Ressourcen ist die automatische Tag-Generierung fehlgeschlagen.

Fehlertyp:

Informationen

Index

0

- 0x01 Extended Error Codes 32
- 0xFF Erweiterte Fehlercodes 32

A

- Adressbeschreibungen 24
- Adressformate 26
- Allgemein 9
- Array-Element 26
- Array mit Offset 26
- Array ohne Offset 26
- Auf Logix-Tags basierte Adressierung 24
- Aufgrund geringer Ressourcen ist die automatische Tag-Generierung fehlgeschlagen. 38

B

- Bei doppeltem Tag 13
- Bei Eigenschaftsänderung 12
- Bei Gerätestart 12
- Betriebsmodus 10
- Bit 26
- BOOL 24
- Boolean 23
- Byte 23

C

- Channel Properties — Advanced 8
- Channel Properties — Ethernet Communications 6
- Channel Properties — Write Optimizations 7
- Char 23
- CIP Error Codes 31
- Condensed 15
- Controller Module 14

ControlLogix-Client-Gerätekonfiguration 17

CSV-Import fehlgeschlagen. Es können keine neuen Logix-Adressen erstellt werden. 37

CSV-Import fehlgeschlagen. Vorhandene Logix-Adressen können nicht gelöscht werden. 37

CSV file 19

D

Data Types Description 23

Date 23

Datensammlung 10

Defining New Logix Addresses 22

Der angegebene TCP/IP-Port liegt außerhalb des zulässigen Bereichs. Der Standard-Port wird verwendet.
| Gültiger Bereich = <Nummer> bis <Nummer>, Standard-Port = <Nummer>. 37

Diagnose 6

DINT 24

Do Not Scan, Demand Poll Only 11

Double 23

Duty Cycle 7

DWord 23

E

Ein weiteres Gerät im Kanal ist bereits als lokale CPU registriert. | Gerät = <Gerät>. 37

Elternteilgruppe 13

Error Codes 31

Erstellen 13

Erweiterte Adressierung 28

Ethernet Settings 6

EtherNet/IP Encapsulation Error Codes 31

EtherNet/IP Module 8

Event Log Messages 33

Expanded 15

External Access 22

F

Fehler beim Importieren der nativen Tag-Datenbank. 33

Fehler beim Importieren der nativen Tag-Datenbank. Datei kann nicht geöffnet werden, da ein allgemeiner Lesefehler aufgetreten ist. 33

Fehler beim Importieren der nativen Tag-Datenbank. Datei kann nicht geöffnet werden. | Betriebssystemfehler = '<Fehler>'. 33

Fehler beim Importieren der nativen Tag-Datenbank. Dateikodierung wird nicht unterstützt. 33

Fehler beim Importieren der nativen Tag-Datenbank. Feldname ist doppelt vorhanden. | Doppelter Feldname = '<Feld>'. 34

Fehler beim Importieren der nativen Tag-Datenbank. Feldname wurde nicht erkannt. | Nicht erkannter Feldname = '<Feldname>'. 34

Fehler beim Importieren der nativen Tag-Datenbank. Tag-Feld-Identifikations-Datensatz fehlt. 34

Fehler beim Importieren der nativen Tag-Datenbank. Tag-Feld-Identifikations-Datensatz ist unvollständig. 35

Fehler beim Importieren des nativen Tags. Gesamtgröße der Datenbank-Tag-Daten ist auf 512 KB begrenzt. | Tag-Adresse = '<Adresse>'. 37

Fehler beim Starten des unangeforderten Logix-Servers. 35

Float 23

G

Generieren 12

Geräteeigenschaften – Allgemein 9

Geräteeigenschaften – Tag-Generierung 11

I

ID 10

Identifikation 6, 9

Initial Updates from Cache 11

INT 24

Inter-Device Delay 8

K

Kanaleigenschaften – Allgemein 5

Kanalzuweisung 9

L

LINT 24

Local 14

Logix Address 22

Logix Config CSV Import 19

Logix Config Object 19

Logix DataType 22

Long 23

Löschen 13

M

Modell 9

Module Type 14

N

Name 9

Native Tag Database 14

Native Tags importiert. | Tag-Anzahl = <Anzahl>, Tag-Datenbank-Pfad = '<Pfad>'. 38

Network Adapter 7

Non-Normalized Float Handling 8

O

OPC-Qualität schlecht bis Schreiben 17

Optimization Method 7

Optionen 17

P

Path 14

R

REAL 24

Reihenfolge von Logix-Array-Daten 29

Remote 14

Replace with Zero 8

Respect Tag-Specified Scan Rate 11

S

Scan Mode 11

Setup 5

Short 23

Simuliert 10

SINT 24

Slot 14

Speicherplatz für Tag konnte nicht zugeordnet werden. | Tag-Adresse = '<Adresse>'. 36

Standard 26

String 23, 26

T

Tag-Generierung 11

Tag-Zähler 6, 11

Tag Hierarchy 15

Tag Scope 28

Tags generiert. | Tag-Anzahl = <Anzahl>, Tag-Hierarchie-Modus = '<Modus>'. 38

TCP/IP Port 8

Treiber 9

U

Überschreiben 13

Übersicht 4

Unerwartetes Fragment beim Verarbeiten fragmentierter Schreibvorgänge entdeckt. | Logix-Adresse = <Tag>. 38

Ungültige Tag-Adresse. Doppelte Tag-Adressen sind nicht zulässig. | Ungültige Adresse = '<Adresse>'. 36

Ungültige Tag-Adresse. Natives Tag wurde nicht importiert. | Ungültige Adresse = '<Adresse>'. 35

Ungültiges natives Tag. Größe der einzelnen Tags ist auf 128 KB begrenzt. | Tag-Adresse = '<Adresse>'. 36

Unmodified 8

Untergruppen zulassen 13

W

Word 23

Write All Values for All Tags 7

Write Only Latest Value for All Tags 7

Write Only Latest Value for Non-Boolean Tags 7

Z

Zeichenfolgen bündeln 17

Zwei Geräte wurden für die Verwendung des gleichen Pfads vom EtherNet/IP-Modul konfiguriert. Jeder Kanal muss einen eindeutigen Pfad vom EtherNet/IP-Modul aufweisen. | Erstes Gerät = '<Adresse>', zweites Gerät = '<Adresse>'. 35

Zwei Kanäle wurden für die Verwendung derselben Netzwerkadapter-IP bzw. desselben TCP-Ports konfiguriert. Jeder Kanal muss an eine eindeutige lokale IP bzw. einen eindeutigen Port gebunden sein. | Erster Kanal = '<Kanal>', zweiter Kanal = '<Kanal>'. 35