

Siemens S7 Plus イーサネットドライバー

© 2026 Kepware. All Rights Reserved.

目次

Siemens S7 Plus イーサネットドライバー	1
目次	2
Siemens S7 Plus イーサネットドライバー ヘルプへようこそ	4
概要	4
設定	5
チャンネルのプロパティ - 一般	6
タグ数	7
Channel Properties — Ethernet Communications	8
Channel Properties — Write Optimizations	8
Channel Properties — Advanced	9
デバイスのプロパティ - 一般	9
動作モード	11
タグ数	11
Device Properties — Scan Mode	11
Device Properties — Timing	12
Device Properties — Auto-Demotion	13
デバイスのプロパティ - タグ生成	13
デバイスのプロパティ - 通信	15
Device Properties — Redundancy	17
Optimizing Communications	19
Data Types Description	21
Data Type Mapping	22
Symbolic Address Descriptions	25
Event Log Messages	26
タグを読み取れません。 タグアドレス = '<アドレス>'、	26
読み取り要求に失敗しました。 	26
タグに書き込めません。 タグアドレス = '<アドレス>'、	26
書き込み要求に失敗しました。 	27
ノードに対してタグが生成されていません。 ノードアドレス = '<アドレス>'、	27
コントローラで検出された変更と一致するように、配列の定義が更新されました。 タグアドレス = '<アドレス>'。	27
PLC 詳細 IP = '<アドレス>'、ポート = '<ポート>'、PLC ファミリー = '<ファミリー>'、タイプ = '<タイプ>'、MLFB = '<mlfb>'、ファームウェア = '<ファームウェア>'。	27
Siemens 通信ライブラリ バージョン = '<バージョン>'、ビルド日 = '<ビルド日>'。	28
PLC からシンボルをロードしています。 	28
セキュリティで保護されている通信の証明書がロードされましたが、無効な証明書が検出されました。 ロード済み = '<数>'、期限切れ = '<数>'、無効な署名 = '<数>'。	28

セキュリティで保護されている通信の証明書が適切にロードされました。 ロード済み = '<数>'。	28
接続の詳細 	28
理由の説明	29
Appendix — Reloading Symbols	39
索引	40

Siemens S7 Plus イーサネットドライバー ヘルプへようこそ

これは、Kepware Siemens S7 Plus イーサネットドライバー のユーザードキュメントです。このドキュメントは、最新の機能と情報を反映して定期的に更新されます。

概要

Siemens S7 Plus イーサネットドライバー とは

設定

このドライバーを使用するためにチャンネルとデバイスを構成する方法

通信の最適化

このドライバーから最高のパフォーマンスを得る方法

データ型の説明

このドライバーでサポートされるデータ型

アドレスの説明

Siemens デバイスでデータ位置のアドレスを指定する方法

イベントログメッセージ

Siemens S7 Plus イーサネットドライバー で生成されるメッセージ

バージョン 1.052

© 2026 Kepware. All Rights Reserved.

概要

Siemens S7 Plus イーサネットドライバーは、Siemens イーサネットデバイスが HMI、SCADA、Historian、MES、ERP や多数のカスタムアプリケーションを含む OPC クライアントアプリケーションに接続するための信頼性の高い手段を提供します。これはシンボリックアドレス指定をサポートする Siemens S7 1200 および 1500 PLC で使用することを目的とし、最適化されたブロックへのアクセスが提供されます。

ドライバーは追加のライブラリやハードウェアを必要としません。標準イーサネットカードが必要です。

設定

このセクションでは、S7-1200 コントローラと S7-1500 コントローラに接続するための設定情報について説明します。

通信プロトコル

S7 Comm Plus

サポートされるデバイス

デバイスはシンボリックアドレス指定をサポートしている必要があります。

- S7-1200
- S7-1500

これらのデバイスには、組み込みのイーサネットモジュールがあります。

サポートされているデバイスは次のファームウェアバージョンをサポートしています。

Kepware リリース	S7-1200	S7-1500
6.14 以降	FW 4.6、4.5、4.4、4.2.1、SPS 4.2	FW 3.0.x、2.9.x、2.8.1、2.6、2.5、SPS 2.0、<1.7
6.11 - 6.13	FW 4.5、4.4、4.2.1、SPS 4.2	FW 2.9.2、2.8.1、2.6、2.5、SPS 2.0、<1.7

セキュリティプロトコル

TLS バージョン 1.3 (古いデバイスではこれ以前のバージョン)

チャンネルとデバイスの制限値

このドライバーでサポートされているチャンネルの最大数は 256 です。このドライバーでサポートされているデバイスの最大数は、1 つのチャンネルにつき 16 です。

● **関連項目:** [チャンネルのプロパティ](#)、[デバイスのプロパティ](#)、[Siemens と Media-Level Redundancy のサポート](#)、およびサポート [ナレッジベース](#) で Siemens TCP/IP から Siemens S7 Plus への移行に関するアーティクル (ログインが必要) を参照してください。

チャンネルのプロパティ - 一般

このサーバーでは、複数の通信ドライバーを同時に使用することができます。サーバープロジェクトで使用される各プロトコルおよびドライバーをチャンネルと呼びます。サーバープロジェクトは、同じ通信ドライバーまたは一意の通信ドライバーを使用する多数のチャンネルから成ります。チャンネルは、OPC リンクの基本的な構成要素として機能します。このグループは、識別属性や動作モードなどの一般的なチャンネルプロパティを指定するときに使用します。

● **関連項目:** プロパティ情報に関するAPIドキュメントについては、`/config/v1/doc/drivers` エンドポイントを参照してください。

プロパティグループ	<input type="checkbox"/> 識別	
一般	名前	
イーサネット通信	説明	
書き込み最適化	ドライバー	
詳細	<input type="checkbox"/> 診断	
プロトコル設定	診断取り込み	無効化
	<input type="checkbox"/> タグ数	
	静的タグ	1

識別

「**名前**」: このチャンネルのユーザー定義識別情報を指定します。各サーバープロジェクトで、それぞれのチャンネル名が一意でなければなりません。名前は最大 256 文字ですが、一部のクライアントアプリケーションでは OPC サーバーのタグ空間をブラウズする際の表示ウィンドウが制限されています。チャンネル名は OPC ブラウザ情報の一部です。チャンネルの作成にはこのプロパティが必要です。

● 予約済み文字の詳細については、サーバーのヘルプで「チャンネル、デバイス、タグ、およびタググループに適切な名前を付ける方法」を参照してください。

「**説明**」: このチャンネルに関するユーザー定義情報を指定します。

● 「説明」などのこれらのプロパティの多くには、システムタグが関連付けられています。

「**ドライバー**」: このチャンネル用のプロトコルドライバーを指定します。チャンネル作成時に選択されたデバイスドライバーを指定します。チャンネルのプロパティではこの設定を変更することはできません。チャンネルの作成にはこのプロパティが必要です。

● **注記:** サーバーがオンラインで常時稼働している場合、これらのプロパティをいつでも変更できます。これには、クライアントがデータをサーバーに登録できないようにチャンネル名を変更することも含まれます。チャンネル名を変更する前にクライアントがサーバーからアイテムをすでに取得している場合、それらのアイテムは影響を受けません。チャンネル名が変更された後で、クライアントアプリケーションがそのアイテムを解放し、古いチャンネル名を使用して再び取得しようとしても、そのアイテムは取得されません。大規模なクライアントアプリケーションを開発した場合は、プロパティを変更しないようにしてください。オペレータがプロパティを変更したりサーバーの機能にアクセスしたりすることを防ぐため、適切なユーザー役割を使用し、権限を正しく管理する必要があります。

診断

「**診断取り込み**」: このオプションが有効な場合、チャンネルの診断情報が OPC アプリケーションに取り込まれます。サーバーの診断機能は最小限のオーバーヘッド処理を必要とするので、必要なときにだけ利用し、必要がないときには無効にしておくことをお勧めします。デフォルトでは無効になっています。

● **注記:** ドライバーまたはオペレーティングシステムが診断をサポートしていない場合、このプロパティは使用できません。

● **ヒント:** 診断情報を ASCII で表示できます。

● **詳細**については、サーバーのヘルプで「通信診断」と「統計タグ」を参照してください。

タグ数

「静的タグ」: デバイスレベルまたはチャンネルレベルで定義される静的タグの数を指定します。この情報は、トラブルシューティングと負荷分散を行う場合に役立ちます。

Channel Properties — Ethernet Communications

Ethernet Communication can be used to communicate with devices.

プロパティグループ	イーサネット設定	
一般	ネットワークアダプタ	デフォルト
イーサネット通信		

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.

プロパティグループ	書き込み最適化	
一般	最適化方法	すべてのタグの最新の値のみを書き込み
シリアル通信	デューティサイクル	10
書き込み最適化		

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.

プロパティグループ	<input type="checkbox"/> 非正規化浮動小数点処理	
一般	浮動小数点値	ゼロで置換
シリアル通信	<input type="checkbox"/> デバイス間遅延	
書き込み最適化	デバイス間遅延 (ミリ秒)	0
詳細		
通信シリアル化		

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.

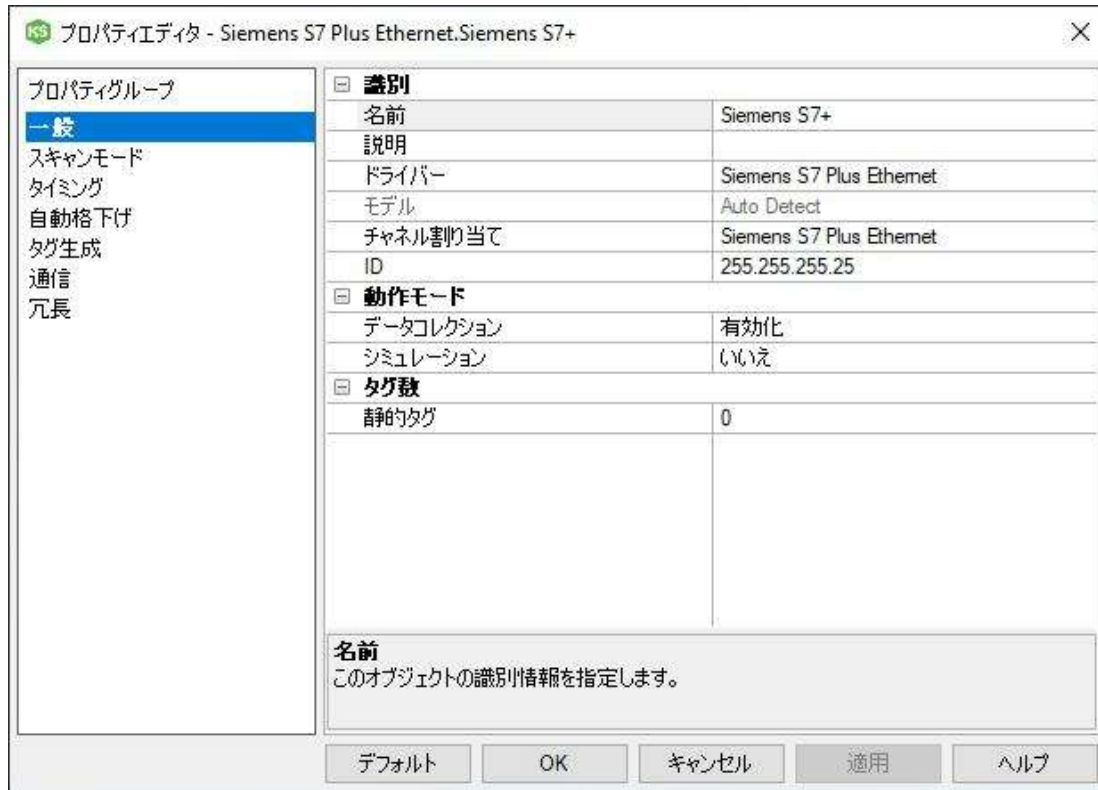
● *For more information on the floating-point values, refer to "How To ... Work with Non-Normalized Floating-Point Values" in the server help.*

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.

デバイスのプロパティ - 一般

デバイスは、通信チャネル上の 1 つのターゲットを表します。



識別

「名前」: デバイスの名前を指定します。これは最大 256 文字のユーザー定義の論理名であり、複数のチャンネルで使用できます。

● **注記**: わかりやすい名前にするを一般的にはお勧めしますが、一部の OPC クライアントアプリケーションでは OPC サーバーのタグ空間をブラウズする際の表示ウィンドウが制限されています。デバイス名とチャンネル名はブラウズツリー情報の一部にもなります。OPC クライアント内では、チャンネル名とデバイス名の組み合わせが "<チャンネル名>.<デバイス名>" として表示されます。

● 詳細については、サーバーのヘルプで「チャンネル、デバイス、タグ、およびタググループに適切な名前を付ける方法」を参照してください。

「説明」: このデバイスに関するユーザー定義情報を指定します。

● 「説明」などのこれらのプロパティの多くには、システムタグが関連付けられています。

「チャンネル割り当て」: このデバイスが現在属しているチャンネルのユーザー定義名を指定します。

「ドライバー」: このデバイスに設定されているプロトコルドライバー。

「モデル」: ドライバーはモデルを自動的に検出します。

「ID」: デバイスのドライバー固有のステーションまたはノードを指定します。入力する ID のタイプは、使用されている通信ドライバーによって異なります。多くの通信ドライバーでは、ID は数値です。数値 ID をサポートするドライバーでは、ユーザーは数値を入力でき、そのフォーマットはアプリケーションのニーズまたは選択した通信ドライバーの特性に合わせて変更できます。フォーマットはデフォルトではドライバーによって設定されます。オプションには「10 進数」、「8 進数」、「16 進数」があります。

● **注記**:

- ドライバーがイーサネットベースであるか、通常とは異なるステーションまたはノード名をサポートしている場合、デバイスの TCP/IP アドレスをデバイス ID として使用できます。TCP/IP アドレスはピリオドで区切った 4 つの値から成り、各値の範囲は 0 から 255 です。一部のデバイス ID は文字列ベースです。ドライバーによっては、ID フィールドで追加のプロパティを設定する必要があります。詳細については、ドライバーのヘルプドキュメントを参照してください。
- このドライバーの ID は、PLC の一意のネットワークアドレスを表す文字列です。通常は IP アドレスまたは構成されたホスト名のフォーマットです。この文字列の最大長は 63 文字です。

動作モード

「**データコレクション**」: このプロパティでは、デバイスのアクティブな状態を制御します。デバイスの通信はデフォルトで有効になっていますが、このプロパティを使用して物理デバイスを無効にできます。デバイスが無効になっている場合、通信は試みられません。クライアントから見た場合、そのデータは無効としてマークされ、書き込み操作は許可されません。このプロパティは、このプロパティまたはデバイスのシステムタグを使用していつでも変更できます。

「**シミュレーション**」: デバイスをシミュレーションモードに切り替えるかどうかを指定します。このモードでは、ドライバーは物理デバイスとの通信を試みませんが、サーバーは引き続き有効な OPC データを返します。シミュレーションモードではデバイスとの物理的な通信は停止しますが、OPC データは有効なデータとして OPC クライアントに返されます。シミュレーションモードでは、サーバーはすべてのデバイスデータを自己反映的データとして扱います。つまり、シミュレーションモードのデバイスに書き込まれたデータはすべて再び読み取られ、各 OPC アイテムは個別に処理されます。アイテムのメモリマップはグループ更新レートに基づきます。(サーバーが再初期化された場合などに) サーバーがアイテムを除去した場合、そのデータは保存されません。デフォルトは「いいえ」です。

● 注記:

1. システムタグ (_Simulated) は読み取り専用であり、ランタイム保護のため、書き込みは禁止されています。このシステムタグを使用することで、このプロパティをクライアントからモニターできます。
2. シミュレーションモードでは、アイテムのメモリマップはクライアントの更新レート (OPC クライアントではグループ更新レート、ネイティブおよび DDE インタフェースではスキャン速度) に基づきます。つまり、異なる更新レートで同じアイテムを参照する 2 つのクライアントは異なるデータを返します。

●シミュレーションモードはテストとシミュレーションのみを目的としています。本番環境では決して使用しないでください。

タグ数

「**静的タグ**」: デバイスレベルまたはチャネルレベルで定義される静的タグの数を指定します。この情報は、トラブルシューティングと負荷分散を行う場合に役立ちます。

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.

プロパティグループ	☐ スキャンモード	
一般	スキャンモード	クライアント固有のスキャン速度を適用 ▼
スキャンモード	キャッシュからの初回更新	無効化
タイミング		

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.

Device Properties — Timing

The device Timing properties allow the driver's response to error conditions to be tailored to fit the application's needs. In many cases, the environment requires changes to these properties for optimum performance. Factors such as electrically generated noise, modem delays, and poor physical connections can influence how many errors or timeouts a communications driver encounters. Timing properties are specific to each configured device.

プロパティグループ	☐ 通信タイムアウト	
一般	接続タイムアウト (秒)	3
スキャンモード	要求のタイムアウト (ミリ秒)	1000
タイミング	タイムアウト前の試行回数	3

Communications Timeouts

Connect Timeout: This property (which is used primarily by Ethernet based drivers) controls the amount of time required to establish a socket connection to a remote device. The device's connection time often takes longer than normal communications requests to that same device. The valid range is 1 to 30 seconds. The default is typically 3 seconds, but can vary depending on the driver's specific nature. If this setting is not supported by the driver, it is disabled.

● **Note:** Due to the nature of UDP connections, the connection timeout setting is not applicable when communicating via UDP.

Request Timeout: Specify an interval used by all drivers to determine how long the driver waits for a response from the target device to complete. The valid range is 50 to 9999999 milliseconds (167 minutes). The default is usually 1000 milliseconds, but can vary depending on the driver. The default timeout for most serial drivers is based on a baud rate of 9600 baud or better. When using a driver at lower baud rates, increase the timeout to compensate for the increased time required to acquire data.

Attempts Before Timeout: Specify how many times the driver issues a communications request before considering the request to have failed and the device to be in error. The valid range is 1 to 10. The default is typically 3, but can vary depending on the driver's specific nature. The number of attempts configured for an application depends largely on the communications environment. This property applies to both connection attempts and request attempts.

Device Properties — Auto-Demotion

The Auto-Demotion properties can temporarily place a device off-scan in the event that a device is not responding. By placing a non-responsive device offline for a specific time period, the driver can continue to optimize its communications with other devices on the same channel. After the time period has been reached, the driver re-attempts to communicate with the non-responsive device. If the device is responsive, the device is placed on-scan; otherwise, it restarts its off-scan time period.

プロパティグループ	自動格下げ	
一般	エラー時に格下げ	有効化
スキャンモード	格下げまでのタイムアウト回数	3
タイミング	格下げ期間 (ミリ秒)	10000
自動格下げ	格下げ時に要求を破棄	無効化

Demote on Failure: When enabled, the device is automatically taken off-scan until it is responding again.

● **Tip:** Determine when a device is off-scan by monitoring its demoted state using the `_AutoDemoted` system tag.

Timeouts to Demote: Specify how many successive cycles of request timeouts and retries occur before the device is placed off-scan. The valid range is 1 to 30 successive failures. The default is 3.

Demotion Period: Indicate how long the device should be placed off-scan when the timeouts value is reached. During this period, no read requests are sent to the device and all data associated with the read requests are set to bad quality. When this period expires, the driver places the device on-scan and allows for another attempt at communications. The valid range is 100 to 3600000 milliseconds. The default is 10000 milliseconds.

Discard Requests when Demoted: Select whether or not write requests should be attempted during the off-scan period. Disable to always send write requests regardless of the demotion period. Enable to discard writes; the server automatically fails any write request received from a client and does not post a message to the Event Log.

デバイスのプロパティ - タグ生成

自動タグデータベース生成機能によって、アプリケーションの設定がプラグアンドプレイ操作になります。デバイス固有のデータに対応するタグのリストを自動的に構築するよう通信ドライバーを設定できます。これらの自動生成されたタグ(サポートしているドライバーの特性によって異なる)をクライアントからブラウズできます。

● 一部のデバイスやドライバーは自動タグデータベース生成のフル機能をサポートしていません。また、すべてのデバイスやドライバーが同じデータ型をサポートするわけではありません。詳細については、データ型の説明を参照するか、各ドライバーがサポートするデータ型のリストを参照してください。

ターゲットデバイスが独自のローカルタグデータベースをサポートしている場合、ドライバーはそのデバイスのタグ情報を読み取って、そのデータを使用してサーバー内にタグを生成します。デバイスが名前付きのタグをネイティブにサポートしてい

ない場合、ドライバーはそのドライバー固有の情報に基づいてタグのリストを作成します。この2つの条件の例は次のとおりです。

1. データ取得システムが独自のローカルタグデータベースをサポートしている場合、通信ドライバーはデバイスで見つかったタグ名を使用してサーバーのタグを構築します。
2. イーサネット I/O システムが独自の使用可能な I/O モジュールタイプの検出をサポートしている場合、通信ドライバーはイーサネット I/O ラックにプラグイン接続している I/O モジュールのタイプに基づいてサーバー内にタグを自動的に生成します。

● **注記:** 自動タグデータベース生成の動作モードを詳細に設定できます。詳細については、以下のプロパティの説明を参照してください。

プロパティグループ	☐ タグ生成	
一般	デバイス起動時	起動時に生成しない
スキャンモード	重複タグ	作成時に削除
タイミング	親グループ	
自動格下げ	自動生成されたサブグループを許可	有効化
タグ生成		

「**プロパティ変更時**」: デバイスが、特定のプロパティが変更された際の自動タグ生成をサポートする場合、「**プロパティ変更時**」オプションが表示されます。これはデフォルトで「はい」に設定されていますが、「いいえ」に設定してタグ生成を実行する時期を制御できます。この場合、タグ生成を実行するには「**タグを作成**」操作を手動で呼び出す必要があります。

「**デバイス起動時**」: OPC タグを自動的に生成するタイミングを指定します。オプションの説明は次のとおりです。

- 「**起動時に生成しない**」: このオプションを選択した場合、ドライバーは OPC タグをサーバーのタグ空間に追加しません。これはデフォルトの設定です。
- 「**起動時に常に生成**」: このオプションを選択した場合、ドライバーはデバイスのタグ情報を評価します。さらに、サーバーが起動するたびに、サーバーのタグ空間にタグを追加します。
- 「**最初の起動時に生成**」: このオプションを選択した場合、そのプロジェクトが初めて実行されたときに、ドライバーがデバイスのタグ情報を評価します。さらに、必要に応じて OPC タグをサーバーのタグ空間に追加します。

● **注記:** OPC タグを自動生成するオプションを選択した場合、サーバーのタグ空間に追加されたタグをプロジェクトとともに保存する必要があります。ユーザーは「**ツール**」 | 「**オプション**」メニューから、自動保存するようプロジェクトを設定できます。

「**重複タグ**」: 自動タグデータベース生成が有効になっている場合、サーバーが以前に追加したタグや、通信ドライバーが最初に作成した後で追加または修正されたタグを、サーバーがどのように処理するかを設定する必要があります。この設定では、自動生成されてプロジェクト内に現在存在する OPC タグをサーバーがどのように処理するかを制御します。これによって、自動生成されたタグがサーバーに累積することもなくなります。

たとえば、「**起動時に常に生成**」に設定されているサーバーのラックで I/O モジュールを変更した場合、通信ドライバーが新しい I/O モジュールを検出するたびに新しいタグがサーバーに追加されます。古いタグが削除されなかった場合、多数の未使用タグがサーバーのタグ空間内に累積することがあります。以下のオプションがあります。

- 「**作成時に削除**」: このオプションを選択した場合、新しいタグが追加される前に、以前にタグ空間に追加されたタグがすべて削除されます。これはデフォルトの設定です。
- 「**必要に応じて上書き**」: このオプションを選択した場合、サーバーは通信ドライバーが新しいタグに置き換えているタグだけ削除します。上書きされていないタグはすべてサーバーのタグ空間に残ります。
- 「**上書きしない**」: このオプションを選択した場合、サーバーは以前に生成されたタグやサーバーにすでに存在するタグを除去しません。通信ドライバーは完全に新しいタグだけを追加できます。

- 「**上書きしない、エラーを記録**」: このオプションには上記のオプションと同じ効果がありますが、タグの上書きが発生した場合にはサーバーのイベントログにエラーメッセージも書き込まれます。

● **注記**: OPC タグの除去は、通信ドライバーによって自動生成されたタグ、および生成されたタグと同じ名前を使用して追加されたタグに影響します。ドライバーによって自動生成されるタグと一致する可能性がある名前を使用してサーバーにタグを追加しないでください。

「**親グループ**」: このプロパティでは、自動生成されたタグに使用するグループを指定することで、自動生成されたタグと、手動で入力したタグを区別します。グループの名前は最大 256 文字です。この親グループは、自動生成されたすべてのタグが追加されるルートブランチとなります。

「**自動生成されたサブグループを許可**」: このプロパティでは、自動生成されたタグ用のサブグループをサーバーが自動的に作成するかどうかを制御します。これはデフォルトの設定です。無効になっている場合、サーバーはグループを作成しないで、デバイスのタグをフラットリスト内に生成します。サーバープロジェクトで、生成されたタグには名前としてアドレスの値が付きます。たとえば、生成プロセス中はタグ名は維持されません。

● **注記**: サーバーがタグを生成しているときに、タグに既存のタグと同じ名前が割り当てられた場合、タグ名が重複しないようにするため、番号が自動的に 1 つ増分します。たとえば、生成プロセスによってすでに存在する "AI22" という名前のタグが作成された場合、代わりに "AI23" としてタグが作成されます。

「**作成**」: 自動生成 OPC タグの作成を開始します。「**タグを作成**」が有効な場合、デバイスの構成が修正されると、ドライバーはタグ変更の可能性についてデバイスを再評価します。システムタグからアクセスできるため、クライアントアプリケーションはタグデータベース作成を開始できます。

● **注記**: 構成がプロジェクトをオフラインで編集する場合、「**タグを作成**」は無効になります。

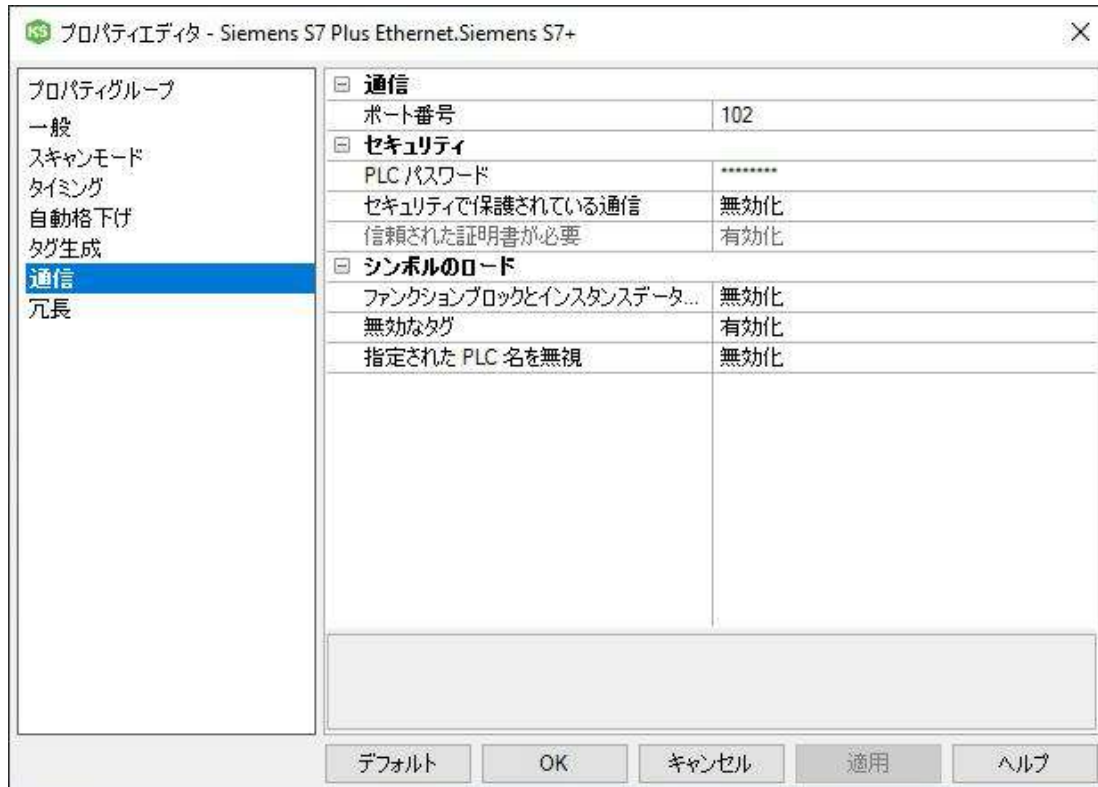
● **注記**:

1. このドライバーは、グループ名のバックスラッシュをアンダースコアに置き換えます。
2. 3 つの HMI プロパティ (Accessible、Writable、Visible) がすべて有効になっている場合、または TIA ポータルプログラミングソフトウェアで Accessible と Visible の両方が有効になっている場合にのみ、タグが生成されます。これらのプロパティの意味は、次のとおりです。
 - HMI/OPC UA/Web API からアクセス可能 (Accessible)
 - HMI/OPC UA/Web API から書き込み可能 (Writable)
 - HMI Engineering で表示可能 (Visible)
3. タグを生成するたびに、コントローラからシンボルが再ロードされます。
4. 一部の Siemens データ型は、このドライバーではサポートされていません。それらのノードでは、タグは生成されません。タグが存在しないノードを警告する **イベントログメッセージ** が出力されます。単一のタグでサポートされていない構造の場合、メッセージは出力されません。

デバイスのプロパティ - 通信

PLC で保護レベルのパスワードが設定されている場合、パスワードを使用して、少なくとも読み取りアクセスを許可するように「PLC パスワード」プロパティを設定する必要があります。

● アクセスレベルの設定の詳細については、PLC のプログラミングソフトウェアを参照してください。



通信

「**ポート番号**」: デバイスに設定されている TCP/IP ポート番号を指定します。有効な範囲は 1 から 65535 です。デフォルト値は「102」です。

● **ヒント**: サーバーと PLC が同じネットワーク上に存在する場合、ほとんどのアプリケーションではデフォルトポートを使用することが推奨されます。ファイアウォールと高度なルータを介してインターネットを使用するアプリケーションの場合、これらの処理が行われるようにポート番号を変更できます。ただし、ほとんどの場合、PLC はポート 102 でのみ接続を受け付けるため、ルータ転送が必要になることがあります。

セキュリティ

「**PLC パスワード**」: PLC で設定されている必要なアクセスレベルのパスワードを指定します。パスワードでサポートされている最大長は 256 文字です。ワイド文字がサポートされています。

「**セキュリティで保護されている通信**」: PLC とのセキュリティで保護された接続を確立するように指定します。セキュリティで保護された PG/PC/HMI 通信は、TIA Portal V17 以降で設定されている PLC で使用できます。デフォルトは「無効化」です。ただし、PLC でサポートされている場合は、セキュリティで保護された通信を有効にすることをお勧めします。有効にすると、すべての情報が暗号化されます。

「**信頼された証明書が必要**」: セキュリティで保護された接続には信頼された証明書が必要であることを指定します。このプロパティは、「セキュリティで保護されている通信」が有効になっている場合にのみ使用できます。デフォルトは「有効化」です。

● 注記:

- 。「信頼された証明書が必要」を有効にする場合、Siemens PLC 証明書を Siemens S7 Plus Ethernet 信頼ストアに手動でインポートする必要があります。

- 証明書をインポートおよび構成するには、サーバー管理ツールで「**証明書ストア**」タブを使用します。「Siemens S7 Plus Ethernet」機能を選択します。詳細は、サーバーのヘルプドキュメントの「**管理**」|「**設定**」|「**証明書ストア**」を参照してください。
- 新しい証明書をインポートした場合、サーバーランタイムを再起動する必要があります。

シンボルのロード

「**ファンクションブロックとインスタンスデータブロックを含める**」: 有効にした場合、ドライバーは Siemens PLC Function Blocks (FB) とそれに関連するインスタンスデータブロック (IDB) のシンボル情報を取得します。これらのシンボルにより、ドライバーは自動タグ生成中にこれらの変数のタグを生成したり、これらのブロック内で定義された変数に対して読み書きを行ったりすることができます。デフォルトは「**無効**」です。

● 注記:

- FB と IDB が含まれている場合、シンボルのロード時間が長くなり、メモリの使用量が増加する可能性があります。
変更は PLC ロジックの実行に影響を与える可能性があるため、FB/IDB 変数に書き込む前に、適切なアクセス許可と安全性チェックを確認してください。

● **関連項目:** 「[理由の説明](#)」セクションの「[シンボルのロードの設定が変更された理由](#)」を参照してください。

「**無効なタグ**」: 有効にした場合、無効なタグが検出されるとデバイスからシンボルがダウンロードされます。デフォルトは「**有効**」です。

● **注記:** デバイスレベルのシステムタグ "_ForceSymbolReload" は、ゼロ以外の値で書き込まれたときにデバイスからシンボルを強制的に再ロードします。

「**指定した PLC 名を無視**」: 有効にすると、シンボルのロードによって、指定された名前に関係なく、PLC 名が 'PLC' として解釈されます。これは冗長性を確保する目的で使用され、プライマリ PLC とセカンダリ PLC との間のシームレスな切り替えが可能になります。自動生成されたタグアドレスには PLC 名が含まれますが、自動生成されたタグ名には含まれません。デフォルトは「**無効**」です。

● 注記:

- このプロパティを変更すると、以前の PLC 名を含む以前のタグアドレスは無効になります。接続を試行する前にタグを再生成してください。
- シンボルの変更により現在の構成が無効になるため、デバイスが接続されている間はこのプロパティを変更できません。

関連項目: 「[理由の説明](#)」セクションの「[シンボルのロードの設定が変更された理由](#)」を参照してください。

Device Properties — Redundancy

プロパティグループ	☐ 冗長	
一般	セカンダリパス	
スキャンモード	動作モード	障害時に切り替え
タイミング	モニターアイテム	
冗長	モニター間隔 (秒)	300
	できるだけ速やかにプライマリに...	はい

Redundancy is available with the Media-Level Redundancy Plug-In.

● *Consult the website, a sales representative, or the [user manual](#) for more information.*

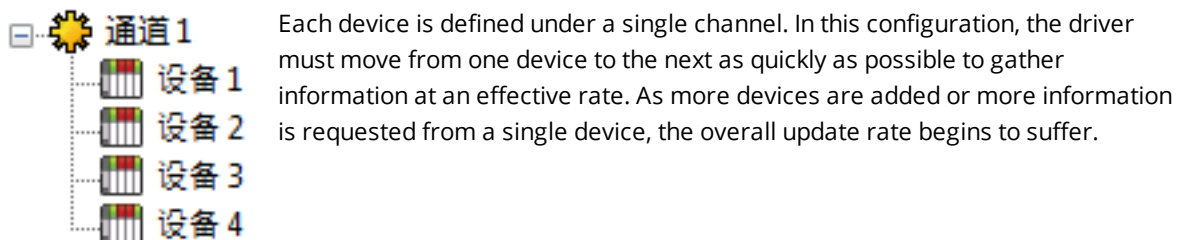
● **注記:** Siemens S7 Plus イーサネットドライバーとMedia-Level Redundancy Plug-In の間の対話については、次の点に注意してください。

- Siemens デバイスの冗長設定では、S7-15xxR または S7-15xxH デバイスのペアで独自の冗長システムを提供できます (Kepware とは無関係)。
- これらの冗長設定が構成されている場合、Media-Level Redundancy Plug-In は機能しません (Siemens 冗長ペアはオーバーライドされません)。
- これらの設定が構成されておらず、PLC が Siemens の冗長または IP アドレス共有を使用していない場合、コントローラはその他の S7-15xx デバイスと同様に動作し、Media-Level Redundancy Plug-In と互換性があるはずですが。

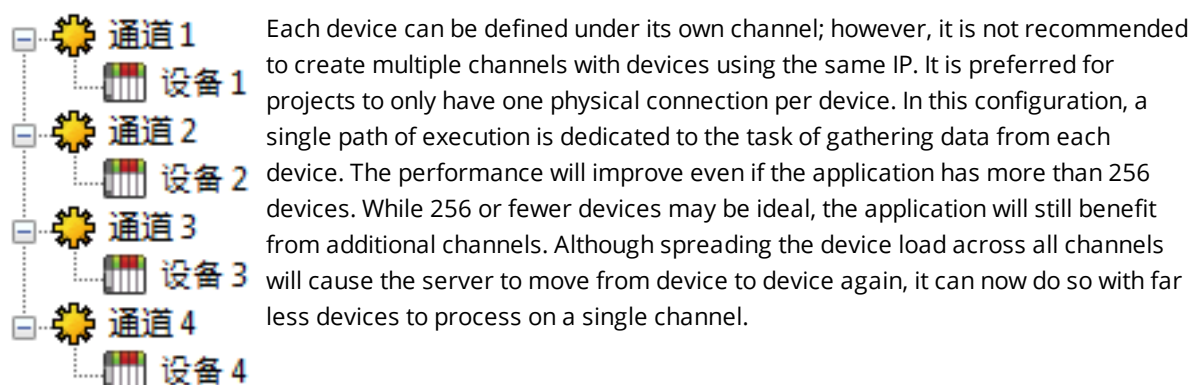
Optimizing Communications

The Siemens S7 Plus イーサネットドライバー was designed to provide the best performance with the least amount of impact on the system's overall performance. While the Siemens S7 Plus イーサネットドライバー is fast, there are a couple of guidelines that can be used to optimize the application and gain maximum performance.

This server refers to communications protocols like Siemens S7 Plus Ethernet as a channel. Each channel defined in the application represents a separate path of execution in the server. Once a channel has been defined, a series of devices can then be defined under that channel. It is not recommended for the series of devices to use the same device IP. It is preferred for projects to only have one physical connection per device. Each of these devices represents a single Siemens Ethernet controller from which data will be collected. Although this approach to defining the application provides a high level of performance, it does not take full advantage of the Siemens S7 Plus イーサネットドライバー or the network. An example of how the application may appear when configured using a single channel is shown below.



If the Siemens S7 Plus イーサネットドライバー could only define one channel, the example above would be the only option available; however, the driver can define up to 256 channels, each with support for 16 devices. Using multiple channels distributes the data collection workload by simultaneously issuing multiple requests to the network. An example of how the same application may appear when configured using multiple channels is shown below.



Although the maximum number of channels is 256, the device ultimately determines the number of allowed connections. This constraint comes from the fact that some devices cannot support so many connections. For these devices, the maximum number of channels defined should equal the maximum number of connections allowed. For devices that support more connections, the maximum of 256 channels should be defined, with devices spread evenly over these channels. If a device's connection limit is exceeded, bad tag quality may occur. *Refer to Siemens documentation for maximum allowed connections.*

Tip: Using arrays can help improve performance.

Tip: For best performance, keep the total number of tags down. Remove tags in the server if the client is not using them. Memory consumption increases as the number of tags increase. It is recommended to keep

server_runtime private bytes from approaching a size of 800 MB. Private bytes exceeding this can result in bad behavior and “Out of memory” event log error messages.

• For more information on device connections, refer to [Device Properties](#).

Data Types Description

The following data types are supported when creating tags and for tag reads and writes:

Data Type	Description
Default	The tag is assigned one of the data types below based upon the S7 data type. This data type is assigned upon the first successful read from the device.  See Also: Data Type Mapping
Boolean	Single bit Range: 0 to 1
Byte	Unsigned 8-bit value Range: 0 to 255
Char	Signed 8-bit value Range: -128 to 127
Word	Unsigned 16-bit value Range: 0 to 65,535
Short	Signed 16-bit value Range: 32,768 to 32,767
DWord	Unsigned 32-bit value Range: 0 to 4,294,967,295
Long	Signed 32-bit value Range: -2,147,483,648 to 2,147,483,647
QWord	Unsigned 64-bit value Range: 0 to 18,446,744,073,709,551,615
LLong	Signed 64-bit value Range: -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807
Float	32-bit floating point value Range: $\pm 1.17154943508222875E-38$ to $\pm 3.4028234663852886E+38$ (normalized) 0 $\pm 1.4012984643248170E-38$ to $\pm 1.1754942106924411E-38$ (denormalized)
Double	64-bit floating point value Range: $\pm 2.2250738585072014E-308$ to $\pm 1.7976931348623157+308$ (normalized) 0 $\pm 4.9406564584124654E-324$ to $\pm 2.2250738585072010E-308$ (denormalized)
String	Null-terminated ASCII string The minimum size of strings is 1. The maximum size of strings is 254 characters. Strings are considered narrow (8 bits in length). This data type supports ASCII and extended-ASCII characters.

 **See Also:** [Data Type Mapping](#)

Data Type Mapping

This table shows data types supported by this driver with data types on the controller on the left and their server equivalents on the right. If multiple server data types are supported for an S7 Data Type, the default data type is shown in bold. Unless stated otherwise, these data types are supported by both the S7-1200 and S7-1500 devices. All data types below are also supported as Read / Write unless otherwise indicated.

S7 Data Type	Server Data Types
Bool	Boolean, Boolean Array
Byte	Byte, Byte Array
Char	String , Byte, Char, String Array, Byte Array, Char Array ● Notes: <ul style="list-style-type: none"> For the String data type, each Char tag is read in as a 1-character string (ex. 65 is 'A'). This means it is possible to write a string value to characters (ex. 'A' can be written instead of 65), assuming the data type is string. If a string of length greater than 1 is written to the char tag, only the first character is written to the device (ex. Writing 'foo' to a char tag truncates and writes 'f'). For Byte data type, the value is the ASCII / Extended ASCII value in decimal format, ranging from 0 – 255 (ex. 'A' is 65). For Char data type, the value is the ASCII / Extended ASCII value in decimal format, ranging from -128 – 127 (ex. 'A' is 65).
INT	Short, Short Array
DINT	Long, Long Array
LINT	LLong ● Notes: This type is only supported by S7-1500 devices. Array of the LINT data type is not supported (as of this release).
Word	Word, Word Array
DWord	DWord, DWord Array
LWord	QWord ● Notes: This type is only supported by S7-1500 devices. Array of the LWORD data type is not supported (as of this release).
REAL	Float, Float Array
LREAL	Double
SINT	Char, Char Array
UDINT	DWord, DWord Array
UINT	Word, Word Array
USINT	Byte, Byte Array
ULINT	QWord ● Note: This type is only supported by S7-1500 devices. Array of the ULINT data type is not supported (as of this release).
STRING	String, String Array
Date And Time (DT)	String ● Note: Date_And_Time is a read-only tag. It displays in standard time with the following format: mm/dd/yyyy hh/min/ss tt. Example of a read on this tag:

S7 Data Type	Server Data Types
	05/21/1991 05:30:21 PM. Not including leap years, the minimum value for this type is 01/01/1990 12:00:00 AM and the maximum value is 12/31/2089 11:59:59 PM. ● Note: This type is only supported by S7-1500 devices. Array of the DT data type is not supported (as of this release).
Time Of Day (TOD)	String ● Note: Time_Of_Day indicates the elapsed time from midnight. It displays in the following format: hh:mm:ss.msc. An example of a read on this tag: 23:31:21.999. The minimum value for this tag is: 00:00:00.000 and the maximum value for this tag is 23:59:59.999. Array of the TOD data type is not supported (as of this release).
Time	String, Long ● Note: Time is displayed as a string in the following format: ddD_hhH_mmM_ssS_hhhMS. An example of a read on this tag: 21D_15H_12M_60S_333MS. The minimum value for this tag is -24D_20H_31M_23S_648MS and the maximum value for this tag is 24D_20H_31M_23S_647MS. ● Note: Array of the Time data type is not supported (as of this release).
S5Time	Long ● Note: The range of this type is 0 to 9990000. The value stored in the server represents S5Time in milliseconds. When writing the value to the server, it should be written as a millisecond value. Negative values cannot be written and any value above the maximum range is automatically set to 9990000. If the value in the PLC is an invalid S5Time value, the server returns bad quality for tags accessing this type. ● Note: This type is only supported by S7-1500 devices.
Date	String ● Note: Date is displayed in the following format: yyyy-mm-dd. An example of a read on this tag: 1991-02-03. The minimum value for this tag is 1990-01-01 and the maximum value for this tag is 2169-06-06. Array of the Date data type is not supported (as of this release).
OB TOD	Short

● **Notes:**

- The number of elements for a client array tag is assigned on the first read of that tag and equals the total number of elements for that array node in the controller. For example, given a 2D array node [2,3], the number of elements assigned to the array tag is 6. If the number of elements for the array node changes in the controller, the driver updates the client array tag with the new element count. In the example above, if the array node changes from a 2D [2,3] to 1D [5], the array tag is assigned a new element count of 5. If instead the array node changed from 2D [2,3] to 1D [6], there is no change in element count so there is no change to the array tag.
- Arrays of server data types are supported for the data types indicated above. The server array maximum size is 65535 elements. The controller arrays can be defined with dimensions of 1D to 6D arrays; these are mapped to a 1-dimensional client array. If arrays in the PLC are defined with more than 65535 elements, automatic tag generation does not create the array tag itself, but all the array element tags are generated. An event log message is posted stating that arrays exceeding 65535

elements are not supported. A read or write on this same array tag also fails with an event log message.

Symbolic Address Descriptions

When manually creating a tag, some restrictions apply to the symbolic address being created. The rules and data types are below.

Examples:

PLC_1.Blocks.Data_block_1.Tag1

PLC_1.Blocks.Data_block_2.Tag2

Address Syntax Rules

- The length of the address must not exceed 1350 characters.
- The address length cannot be 0.
- The symbolic address cannot contain spaces only.
- The string for the address must come with a valid data type (*listed in the [Data Types Mapping](#)*).
- Symbolic address strings cannot have leading and trailing spaces within the name.
- If the following characters are included in a single node name, the node name must be wrapped in double quotation marks:
 - decimal point
 - (open parenthesis
 -) close parenthesis
 - [open square bracket
 -] close square bracket

Examples:

PLC_1.Blocks.Data_block_1."Tag("

PLC_1.Blocks."Data_block.1".Tag

Tip: If the PLC node name contains a double quotation mark, the double quotation mark must be escaped with a double quotation mark and the node name must be wrapped in double quotation marks.

Example:

"PLC""Name".Blocks.Data_block_1.Tag

Notes:

- Reading or writing to a symbolic address that does not exist in the PLC triggers the driver to load symbols from the PLC on every read or write request. For best performance, tags that reference invalid symbolic addresses must be removed to prevent them from being included in read or write requests and continuously loading symbols.
- There is no syntax for specifying the number of rows and columns for client array tags. Since 1D – 6D array tags in the controller are flattened to 1D (one dimension) client arrays, the number of rows is fixed at 1. The driver automatically determines and assigns the number of columns to the tag at runtime.

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.

タグを読み取れません。 | タグアドレス = '<アドレス>',

エラータイプ:

エラー

考えられる原因:

タグの読み取り中にエラーが発生しました。イベントログのメッセージで理由を参照してください。

解決策:

理由の説明セクションの理由をレビューしてください。

● 関連項目:

理由の説明

読み取り要求に失敗しました。 |

エラータイプ:

エラー

考えられる原因:

読み取り要求中にエラーが発生しました。イベントログのメッセージで理由を参照してください。

解決策:

理由の説明セクションの理由をレビューしてください。

● 関連項目:

理由の説明

タグに書き込めません。 | タグアドレス = '<アドレス>',

エラータイプ:

エラー

考えられる原因:

タグの書き込み中にエラーが発生しました。イベントログのメッセージで理由を参照してください。

解決策:

理由の説明セクションの理由をレビューしてください。

● 関連項目:

理由の説明

書き込み要求に失敗しました。 |

エラータイプ:

エラー

考えられる原因:

書き込み要求中にエラーが発生しました。イベントログのメッセージで理由を参照してください。

解決策:

理由の説明セクションの理由をレビューしてください。

● 関連項目:

理由の説明

ノードに対してタグが生成されていません。 | ノードアドレス = '<アドレス>',

エラータイプ:

警告

考えられる原因:

ノードのブラウズ中にエラーが発生しました。イベントログのメッセージで理由を参照してください。

解決策:

理由の説明セクションの理由をレビューしてください。

● 関連項目:

理由の説明

**コントローラで検出された変更と一致するように、配列の定義が更新されました。 | タグ
アドレス = '<アドレス>'。**

エラータイプ:

警告

考えられる原因:

コントローラのこの配列で変更された要素の合計数。

解決策:

操作は必要ありません。タグ<アドレス>は新規要素数に自動的に割り当てられました。

**PLC 詳細 | IP = '<アドレス>', ポート = '<ポート>', PLC ファミリー = '<ファミリー>', タイプ
= '<タイプ>', MLFB = '<mlfb>', ファームウェア = '<ファームウェア>'。**

エラータイプ:

情報

Siemens 通信ライブラリ | バージョン = '<バージョン>', ビルド日 = '<ビルド日>'。

エラータイプ:

情報

PLC からシンボルをロードしています。 |

エラータイプ:

情報

考えられる原因:

PLC からのシンボルのロードが必要です。イベントログのメッセージで理由を確認してください。

解決策:

理由の説明セクションの理由をレビューしてください。

● 関連項目:

理由の説明

セキュリティで保護されている通信の証明書がロードされましたが、無効な証明書が検出されました。 | ロード済み = '<数>', 期限切れ = '<数>', 無効な署名 = '<数>'。

エラータイプ:

セキュリティ

考えられる原因:

1. 証明書ストアの証明書が期限切れになっています。
2. 証明書ストアの証明書に無効な署名があります。

解決策:

1. 期限切れの証明書を除去します。将来の有効期限を持つ証明書をインポートします。
2. 無効な署名証明書を除去します。有効な署名を持つ証明書をインポートします。

セキュリティで保護されている通信の証明書が適切にロードされました。 | ロード済み = '<数>'。

エラータイプ:

セキュリティ

接続の詳細 |

エラータイプ:

セキュリティ

考えられる原因:

接続が適切に確立されました。このメッセージに追加されたイベントログで詳細を参照してください。

理由の説明

一部のイベントログメッセージでは、理由フィールドの下に追加情報が表示されます。以下のリンクをクリックすると、エラーの理由が表示されます。

[パスへのアクセスが拒否されました](#)

[65535 個の要素を超える配列はサポートされていません](#)

[配列範囲が無効です](#)

[SSL を初期化できませんでした](#)

[日付文字列に構文エラーが含まれています。予期されるフォーマットは yyyy-mm-dd です](#)

[デバイスが応答していません](#)

[デバイスが応答していません。接続がデバイスによって切断されました](#)

[エラーコード = <16 進エラーコード>](#)

[ホスト名の解決に失敗しました](#)

[内部ドライバーエラーが発生しました](#)

[S5Time の読み取り時に無効な値が見つかりました](#)

[S5Time の書き込み時に無効な値が見つかりました](#)

[新規接続](#)

[ノードパスが無効です](#)

[接続外です](#)

[メモリ不足です](#)

[パスワードが変更されました](#)

[パスワードが必要です](#)

[この Siemens データ型からの読み取りはサポートされていません](#)

[セキュリティで保護されている通信が必要です](#)

[シンボルのロードの設定が変更されました](#)

[ノードのデータ型がサポートされていません](#)

[時刻文字列に構文エラーが含まれています。予期されるフォーマットは 'hh:mm:ss.hhh' です](#)

[時間文字列に構文エラーが含まれています。予期されるフォーマットは ddD_hhH_mmM_ssS_hhhMS です](#)

[サポートされていない通信構成が検出されました](#)

[このアドレスに対してサポートされていないデータ型](#)

[信頼されていない証明書](#)

[ユーザーが強制した再ロード](#)

[この Siemens データ型への書き込みはサポートされていません](#)

[パスワードが無効です](#)

理由 = 配列範囲が無効です。

考えられる原因:

コントローラ内の配列の要素数がクライアント配列の要素数よりも少なく、クライアントが配列タグに書き込もうとしました。

考えられる解決策:

1. クライアントが、コントローラ内の配列の要素数以下の数の要素を書き込んでいることを確認します。
2. コントローラ配列が変更された場合、クライアントはタグに書き込む前にタグを読み取り、配列構成を同期化する必要があります。

理由 = SSL を初期化できませんでした。

考えられる原因:

PLC が、セキュリティで保護された通信をサポートしない TIA Portal のバージョンでプログラミングされました。セキュリティで保護された PG/PC および HMI の通信は、TIA Portal V17 以降を使用して構成された PLC で使用できます。

考えられる解決策:

セキュリティで保護された通信を使用するには、PLC プロジェクトが V17 以降でなければなりません。

理由 = デバイスが応答していません: ID = <IP アドレス>。

考えられる原因:

1. IP アドレスが正しくない。
2. PLC が応答するように構成されていない。
3. 設定されている接続タイムアウトが短すぎる。
4. 設定されている要求タイムアウトが短すぎる。
5. 接続数が多すぎるため、デバイスが接続を拒否した。

考えられる解決策:

1. IP アドレスを確認します。
2. PLC が応答可能な状態になっているかどうか、正しいプログラムがロードされているかどうかを確認します。
3. PLC が接続要求を受け入れることができるように、設定されている接続タイムアウトの値を大きくします。
4. PLC が接続要求に応答できるように、設定されている要求タイムアウトの値を大きくします。
5. PLC で許可されている接続数を確認します。

理由 = デバイスが応答していません。接続がデバイスによって切断されました。ID = <IP アドレス>

考えられる原因:

接続数が多すぎるため、デバイスが接続を拒否した。

考えられる解決策:

PLC で許可されている接続数を確認します。

理由 = ホスト名の解決に失敗しました。ホスト名 = <ホスト名>。

考えられる原因:

1. IP アドレスのフォーマットが、4 つのオクテットを持つ必要な IP アドレスに従っていない。
2. 指定されたホスト名が無効であり、IP アドレスに解決されない。

考えられる解決策:

1. IP アドレスのフォーマットを確認または修正します。
2. 文字列が IP アドレスに解決される有効なホスト名であることを確認するか、または有効なホスト名になるよう修正します。

理由 = ノードパスが無効です。

考えられる原因:

1. コントローラ内にノードパスが存在しない。
2. 配列内に配列要素が含まれていない。
3. ノードパスが UDT のメンバーではない。
4. ノードパスが使用可能な状態になっていない。
5. ノードパスが適用可能な状態になっていない。
6. 次のいずれかの特殊文字が含まれているノードパスの部分が引用符で囲まれていない: 「.」、「[」、「]」、「(」、「)」
7. 配列要素の表記に無効な文字が含まれている。
8. ノードパスが配列になっているが、インデックス構文がない。
9. ノードパスが UDT メンバーのパスになっているが、UDT がインスタンスではない。
10. 記号のロード後にノードパスが変更または削除された。
11. 記号のロード後にノードパスのデータ型が変更された。

考えられる解決策:

1. ノードパスの構文が正しいかどうかを確認します。
2. コントローラ内にノードが存在するかどうかを確認します。
3. ノードパスが、配列や UDT などの複雑なタイプになっていないかどうかを確認します。

理由 = パスへのアクセスが拒否されました。

考えられる原因:

1. ノードパスが読み取り専用になっている。
2. ノードパスが HMI にアクセスできない状態になっている。
3. ノードパスが保護されている。

考えられる解決策:

1. HMI からノードにアクセスできるかどうかを確認します。
2. 書き込みできない場合は、ノードに書き込みアクセス許可が設定されているかどうかを確認します。

理由 = 65535 個の要素を超える配列はサポートされていません。

考えられる原因:

コントローラ内の配列に、65535 個を超える数の要素があります。

考えられる解決策:

コントローラの配列を、最大サイズの 65535 個の要素に減少させます。

理由 = この Siemens データ型からの読み取りはサポートされていません。

考えられる原因:

ドライバーは、サポートされていない Siemens データ型で定義されたノードの読み取りをサポートしていません。

考えられる解決策:

有効な Siemens データ型のリストについては、[「データ型のマッピング」](#)を参照してください。問題を解決できない場合は、テクニカルサポートにお問い合わせください。

理由 = この Siemens データ型への書き込みはサポートされていません。

考えられる原因:

Siemens Date and Time データ型で定義されているノードへの書き込みがドライバーでサポートされていない。

考えられる解決策:

有効な Siemens データ型のリストについては、[「データ型のマッピング」](#)を参照してください。問題を解決できない場合は、テクニカルサポートにお問い合わせください。

理由 = 時刻文字列に構文エラーが含まれています。予期されるフォーマットは hh:mm:ss.hhh です。

考えられる原因:

1. 値の構文が正しい形式になっていない。
2. 時間、分、秒、またはミリ秒の部分に範囲外の値が指定されている。

考えられる解決策:

1. 書き込む文字列の値を「hh:mm:ss.hhh」という形式に設定して、もう一度書き込み操作を実行します。
2. 時間、分、秒、ミリ秒の部分に正しい値が設定されているかどうかを確認します。

理由 = このアドレスではサポートされていないデータ型。データ型 = <data type>

考えられる原因:

タグのデータ型がノードのデータ型と一致していない。

考えられる解決策:

ノードのデータ型に一致するように、タグのデータ型を修正します。

● **関連項目:** 有効な Siemens データ型のリストについては、[「データ型のマッピング」](#)を参照してください。

理由 = 日付文字列に構文エラーが含まれています。予期されるフォーマットは yyyy-mm-dd です。

考えられる原因:

1. 値の構文が正しい形式になっていない。
2. 日、月、または年の部分に範囲外の値が指定されている。

考えられる解決策:

1. 書き込む文字列の値を「yyyy-mm-dd」という形式に設定して、もう一度書き込み操作を実行します。
2. 日、月、年の部分に正しい値が設定されているかどうかを確認します。

理由 = 時間文字列に構文エラーが含まれています。予期されるフォーマットは ddD_hhH_mmM_ssS_hhhMS です。

考えられる原因:

1. 値の構文が正しい形式になっていない。
2. 日、時間、分、秒、またはミリ秒の部分に範囲外の値が指定されている。

考えられる解決策:

1. 書き込む文字列の値を「ddD_hhH_mmM_ssS_hhhMS」という形式に設定して、もう一度書き込み操作を実行します。
2. 日、時間、分、秒、ミリ秒の部分に正しい値が設定されているかどうかを確認します。

理由 = ノードのデータ型がサポートされていません; ノード タイプ = <node type>

考えられる原因:

ドライバーで Siemens データ型がサポートされていない。

考えられる解決策:

テクニカルサポートに連絡します。

● **関連項目:** 有効な S7 データ型のリストについては、[「データ型のマッピング」](#)を参照してください。

理由 = パスワードが無効です

考えられる原因:

設定されているデバイスのパスワードが PLC で設定されている保護レベルのパスワードと一致しない。

考えられる解決策:

設定したパスワードを確認します。

理由 = パスワードが変更されました。

考えられる原因:

PLC またはサーバープロジェクトのパスワードが変更された。

考えられる解決策:

PLC 内とサーバープロジェクト内のパスワードが一致し、認証が成功し、更新が続行されることを確認します。

理由 = パスワードが必要です。

考えられる原因:

PLC に保護レベルのパスワードが設定されている。

考えられる解決策:

デバイスのパスワードを設定します。

理由 = S5Time の読み取り時に無効な値が見つかりました。このタイプの有効な範囲は 0 から 9990000 の間です。

考えられる原因:

アクセス中のデバイスで設定されている S5Time が正しくない。

考えられる解決策:

デバイスに有効な S5Time 値を書き込み、情報を正しく読み取ることができるようにします。これで解決しない場合は、テクニカルサポートにお問い合わせください。

理由 = S5Time の書き込み時に無効な値が見つかりました。このタイプの有効な範囲は 0 から 9990000 の間です。

考えられる原因:

S5Time タグに負の値が書き込まれている。

考えられる解決策:

有効な S5Time 値を入力します。これで解決しない場合は、テクニカルサポートにお問い合わせください。

エラーコード = <hex error code>

考えられる原因:

一般的ではないエラーが原因で要求が失敗しました。

考えられる解決策:

1. アイテムの構成内容とアクセス権を確認します。
2. テクニカルサポートに連絡します。

理由 = 内部ドライバーエラーが発生しました。

考えられる原因:

ドライバー内で予期しないエラーが発生したため、要求が失敗しました。

考えられる解決策:

1. アイテムの構成内容とアクセス権を確認します。
2. テクニカルサポートに連絡します。

サポートされていない通信構成が検出されました、ID = < IP アドレス >。

考えられる原因:

1. PLC が、現在処理されていないエラーを返しました。
2. PLC デバイス構成はサポートされていません。

考えられる解決策:

問題が引き続き発生する場合は、テクニカルサポートに連絡してください。

理由 = 新規接続。

考えられる原因:

1. クライアントの読み取り/書き込みまたはタグ生成要求で PLC への接続が確立された。
2. ネットワークの問題により、以前の接続が切断され、PLC への新しい接続が確立された。

考えられる解決策:

ネットワークの安定性について考えられる問題を確認します。

理由 = 接続外です。

考えられる原因:

接続数が多すぎるため、デバイスが接続を拒否した。

考えられる解決策:

PLC で許可されている接続数を確認します。

理由 = メモリ不足です。

考えられる原因:

ランタイムプロセスが、システム上のアプリケーションで使用可能なメモリ容量に達しました。

考えられる解決策:

1. ランタイムを再起動します。
2. プロジェクト内の静的タグの数を減らします。クライアントがアクセスするかどうかにかかわらず、プロジェクト内のすべての静的タグがランタイムメモリにロードされます。そのため、不要な静的タグをすべて除去します。
3. 各コントローラへの接続数を減らします。各接続は、コントローラシンボルを読み取ったり書き込んだりするために、それらのコントローラシンボルをメモリにロードする必要があります。
4. チャネルとデバイスの数を減らします。ランタイムを起動した後、クライアントを接続し、すべてのタグの品質が良好であるかどうかを確認します。ランタイムのプライベートバイトサイズを確認して、システム上のアプリケーションの許容限度に近づいているかどうかを確認します。サイズは、接続数、プロジェクト内のタグ数、および各コントローラのシンボル数によって異なります。

理由 = セキュリティで保護されている通信が必要です。

考えられる原因:

PLC は、セキュリティで保護された PG/PC と HMI の通信のみを許可するように構成されています。

考えられる解決策:

セキュリティで保護された通信 デバイスのプロパティを有効にします。

理由 = 信頼されていない証明書。

考えられる原因:

信頼された証明書がありません。

考えられる解決策:

1. 証明書ストアで PLC の証明書を追加または置換する。
2. 「信頼された証明書が必要」デバイスプロパティを無効にして、混合モードで設定されている PLC との通信を許可する。

理由 = シンボルのロードの設定が変更されました。

考えられる原因:

「インスタンスデータブロックおよびファンクションブロックを含める」プロパティが変更されました (デバイスプロパティの「シンボルのロード」グループにあります)。

考えられる解決策:

プロパティが有効になっている場合、「シンボルのロード」にインスタンスデータブロックおよびファンクションブロックのシンボルを含めます。

理由 = ユーザーが強制した再ロード。

考えられる原因:

ゼロ以外の値がデバイスレベルのシステムタグ "_ForceSymbolReload" に書き込まれました。

考えられる解決策:

シンボルが再ロードされます。

Appendix — Reloading Symbols

If getting bad quality after controller project edits, reload symbols to update. Scenarios where a tag can change to bad quality because of controller project changes:

- When a tag's symbolic name in the controller is modified;
- When a tag's data type in the controller is modified;
- When a tag is removed; or
- If the Access Level of the device is changed to HMI access or below and the password in the server is not updated.

To make sure all other tags are reporting accurately, the server reloads tag information from the controller to maintain the accuracy of all other tags. When the server identifies that a tag has been modified, removed, or if an ATG occurs; it reloads tag information. All other scenarios require a manual reload of tag information. This is accomplished by executing a runtime reinitialize, which allows the server to reload tags. If a tag is still bad quality after the tag information update, consider manually correcting that tag or perform an ATG to correct tag information for many tags. In cases where tags are added in the controller, but not in the server; those tags must be manually entered into the server with correct information for it to read or write properly.

Note: Reading or writing to a symbolic address that does not exist in the PLC triggers the driver to load symbols from the PLC on every read or write request. For best performance, tags that reference invalid symbolic addresses must be removed to prevent them from being included in read or write requests and continuously loading symbols.

索引

A

Access Level 39
Address Descriptions 25
Address Syntax 25
Attempts Before Timeout 13
Auto-Demotion 13

B

Boolean 21-22
Byte 21-22

C

Channel Properties — Advanced 9
Channel Properties — Ethernet Communications 8
Channel Properties — Write Optimizations 8
Char 21-22
Communications Timeouts 12
Connect Timeout 12

D

Data Type Mapping 22
Data Types Description 21
Date 23
Default 21
Demote on Failure 13
Demotion Period 13
Device Properties — Auto-Demotion 13
Device Properties — Redundancy 17
Device Properties — Timing 12
DINT 22
Discard Requests when Demoted 13
Do Not Scan, Demand Poll Only 12

Double 21-22

Duty Cycle 9

DWord 21-22

E

Ethernet Settings 8

Event Log Messages 26

F

Float 21-22

I

ID 10

Initial Updates from Cache 12

INT 22

Inter-Device Delay 9

L

LINT 22

LLong 21-22

Long 21-22

LREAL 22

N

Network Adapter 8

Non-Normalized Float Handling 9

O

Optimization Method 8

Optimizing Communications 19

P

PLC からシンボルをロードしています。 | 28

PLC パスワード 16-17

PLC 詳細 | IP = '<アドレス>', ポート = '<ポート>', PLC ファミリー = '<ファミリー>', タイプ = '<タイプ>', MLFB = '<mlfb>', ファームウェア = '<ファームウェア>'。 27

Q

QWord 21-22

R

REAL 22

Redundancy 17

Reloading Symbols 39

Replace with Zero 9

Request Timeout 12

Respect Tag-Specified Scan Rate 12

S

S5Time 23

S7-1200 5

S7-1500 5

S7 Comm Plus 5

Scan Mode 11

Short 21-23

Siemens 通信ライブラリ | バージョン = '<バージョン>', ビルド日 = '<ビルド日>'。 28

SINT 22

String 21-23

symbolic name 39

T

Time 23

Timeouts to Demote 13

Timing 12

TOD 23

U

UDINT 22

UINT 22

ULINT 22

Unmodified 9

USINT 22

W

Word 21-22

Write All Values for All Tags 8

Write Only Latest Value for All Tags 9

Write Only Latest Value for Non-Boolean Tags 8

い

イーサネットカード 4

こ

コントローラで検出された変更と一致するように、配列の定義が更新されました。 | タグアドレス = '<アドレス>'。 27

さ

サブグループを許可 15

サポートされていない通信構成が検出されました、ID = '< IP アドレス >'。 36

し

シミュレーション 11

シンボルのロード 17

せ

セキュリティ 16

セキュリティで保護されている通信の証明書がロードされましたが、無効な証明書が検出されました。 | ロード済み = '<数>', 期限切れ = '<数>', 無効な署名 = '<数>'。 28

セキュリティで保護されている通信の証明書が適切にロードされました。 | ロード済み = '<数>'。 28

た

タグに書き込めません。 | タグアドレス = '<アドレス>', 26

タグを読み取れません。 | タグアドレス = '<アドレス>', 26

タグ数 7, 11

タグ生成 13

ち

チャンネルのプロパティ - 一般 6

チャンネル割り当て 10

て

データコレクション 11

デバイスのプロパティ - タグ生成 13

デバイスのプロパティ - 一般 9

デバイス起動時 14

と

ドライバー 10

の

ノードに対してタグが生成されていません。 | ノードアドレス = '<アドレス>', 27

ふ

プロトコル 5

プロパティ変更時 14

ほ

ポート番号 16

漢字

一般 9

概要 4

作成 15

削除 14

識別 6, 10

重複タグ 14

書き込み要求に失敗しました。 | 27

上書き 14

親グループ 15

診断 6

生成 14

接続の詳細 | 28

通信 15-16

動作モード 11

読み取り要求に失敗しました。 | 26

名前 10

理由 29

理由 = シンボルのロードの設定が変更されました 37

理由 = セキュリティで保護されている通信が必要です 37

理由 = メモリ不足です 36

理由 = ユーザーが強制した再ロード 38

理由 = 信頼されていない証明書 37

理由 = 新規接続 36

理由 = 接続外です 36