

Mitsubishi CNC Ethernetドライバー

© 2019 PTC Inc. All Rights Reserved.

目次

Mitsubishi CNC Ethernet ドライバー	1
目次	2
Mitsubishi CNC Ethernet ドライバー	3
概要	3
設定	4
チャンネルのプロパティ - 一般	4
チャンネルのプロパティ - イーサネット 通信	5
チャンネルのプロパティ - 書き込み最適化	5
チャンネルのプロパティ - 詳細	6
デバイスのプロパティ - 一般	6
動作モード	7
デバイスのプロパティ - スキャンモード	8
デバイスのプロパティ - タイミング	8
デバイスのプロパティ - 自動格下げ	9
デバイスのプロパティ - ネットワークパラメータ	10
デバイスのプロパティ - 冗長	11
マルチレベルネットワーク	13
通信の最適化	14
データ型の説明	15
アドレスの説明	16
イベントログメッセージ	19
デバイスでのタグの読み取りに失敗しました。 タグアドレス = '<アドレス>'。	19
デバイスでのタグの書き込みに失敗しました。デバイスがエラーコードを返しました。 タグアドレス = '<アドレス>', エラーコード = <コード>。	19
デバイスでのブロック読み取りに失敗しました。デバイスはエラーコードを返しました。 ブロックサイズ = <数値> (ポイント)、ブロック開始アドレス = '<アドレス>', エラーコード = <コード>。	19
デバイスでのタグの書き込みに失敗しました。フレーミングエラー。 タグアドレス = '<アドレス>'。	19
デバイスでのタグの書き込みに失敗しました。デバイスが不正なトランザクション ID を返しました。 タグアドレス = '<アドレス>'。	19
デバイスでのブロック読み取りに失敗しました。フレーミングエラー。 ブロックサイズ = <数値> (ポイント)、ブロック開始アドレス = '<アドレス>'。	20
デバイスでのブロック読み取りに失敗しました。デバイスが不正なトランザクション ID を返しました。 ブロックサイズ = <数値> (ポイント)、ブロック開始アドレス = '<アドレス>'。	20
デバイスでのタグの書き込みに失敗しました。接続エラー。 タグアドレス = '<アドレス>'。	20
デバイスでのブロック読み取りに失敗しました。接続エラー。 ブロックサイズ = <数値> (ポイント)、ブロック開始アドレス = '<アドレス>'。	20
索引	21

Mitsubishi CNC Ethernet ドライバー

ヘルプバージョン 1.041

目次

概要

Mitsubishi CNC Ethernet ドライバーとは

デバイスの設定

このドライバーを使用するためにデバイスを構成する方法

イーサネット通信の最適化

Mitsubishi CNC Ethernet 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

Mitsubishi CNC Ethernet デバイスでデータ位置のアドレスを指定する方法

エラーの説明

ドライバーで生成されるエラーメッセージ

概要

Mitsubishi CNC Ethernet ドライバー は三菱 CNC イーサネットコントローラが 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。

設定

サポートされるデバイス

C64 CNC コントローラ

通信プロトコル

Winsock V1.1 以上を使用したイーサネット
TCP/IP

サポートされる通信パラメータ

バイナリフォーマットのみ

モデル

AJ71QE71 と互換性のあるイーサネットモジュールが搭載された三菱 C64

チャンネルのプロパティ - 一般

このサーバーは、複数の通信ドライバーの同時使用をサポートしています。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。

プロパティグループ	識別	
一般	名前	Channel1
シリアル通信	説明	
書き込み最適化	ドライバー	
詳細	診断	
通信シリアル化	診断取り込み	無効化

識別

「名前」: このチャンネルのユーザー定義の識別情報。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の情報。

● 「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」: このチャンネルに選択されているプロトコルドライバー。このプロパティでは、チャンネル作成時に選択されたデバイスドライバーが示されます。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。このことを念頭において、大規模なクライアントアプリケーションを開発した後はプロパティに対する変更を行わないようにします。サーバー機能へのアクセス権を制限してオペレータがプロパティを変更できないようにするには、ユーザーマネージャを使用します。

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**: ドライバーで診断機能がサポートされていない場合、このプロパティは使用できません。

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

チャンネルのプロパティ - イーサネット通信

イーサネット通信を使用してデバイスと通信できます。

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		

イーサネット設定

「**ネットワークアダプタ**」: バインドするネットワークアダプタを指定します。空白のままにするか、「デフォルト」を選択した場合、オペレーティングシステムはデフォルトのアダプタを選択します。

チャンネルのプロパティ - 書き込み最適化

サーバーと同様に、デバイスへのデータの書き込みはアプリケーションの最も重要な要素です。サーバーは、クライアントアプリケーションから書き込まれたデータがデバイスに遅延なく届くようにします。このため、サーバーに用意されている最適化プロパティを使用して、特定のニーズを満たしたり、アプリケーションの応答性を高めたりできます。

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		

書き込み最適化

「**最適化方法**」: 基礎となる通信ドライバーに書き込みデータをどのように渡すかを制御します。以下のオプションがあります。

- 「**すべてのタグのすべての値を書き込み**」: このオプションを選択した場合、サーバーはすべての値をコントローラに書き込もうとします。このモードでは、サーバーは書き込み要求を絶えず収集し、サーバーの内部書き込みキューにこれらの要求を追加します。サーバーは書き込みキューを処理し、デバイスにできるだけ早くデータを書き込むことによって、このキューを空にしようとする。このモードでは、クライアントアプリケーションから書き込まれたすべてのデータがターゲットデバイスに送信されます。ターゲットデバイスで書き込み操作の順序または書き込みアイテムのコンテンツが一意に表示される必要がある場合、このモードを選択します。
- 「**非 Boolean タグの最新の値のみを書き込み**」: デバイスにデータを実際に送信するのに時間がかかっているために、同じ値への多数の連続書き込みが書き込みキューに累積することがあります。書き込みキューにすでに置かれている書き込み値をサーバーが更新した場合、同じ最終出力値に達するまでに必要な書き込み回数ははるかに少なくなります。このようにして、サーバーのキューに余分な書き込みが累積することがなくなります。ユーザーがスライドスイッチを動かすのをやめると、ほぼ同時にデバイス内の値が正確な値になります。モード名からもわかるように、Boolean 値でない値はサーバーの内部書き込みキュー内で更新され、次の機会にデバイスに送信されます。これによってアプリケーションのパフォーマンスが大幅に向上します。
 - **注記**: このオプションを選択した場合、Boolean 値への書き込みは最適化されません。モーメンタリッシュボタンなどの Boolean 操作で問題が発生することなく、HMI データの操作を最適化できます。
- 「**すべてのタグの最新の値のみを書き込み**」: このオプションを選択した場合、2 つ目の最適化モードの理論がすべてのタグに適用されます。これはアプリケーションが最新の値だけをデバイスに送信する必要がある場合に特に役立ちます。このモードでは、現在書き込みキューに入っているタグを送信する前に更新することによって、すべての書き込みが最適化されます。これがデフォルトのモードです。

「**デューティサイクル**」: 読み取り操作に対する書き込み操作の比率を制御するときに使用します。この比率は必ず、読み取り 1 回につき書き込みが 1 から 10 回の間であることが基になっています。デューティサイクルはデフォルトで 10 に設定されており、1 回の読み取り操作につき 10 回の書き込みが行われます。アプリケーションが多数の連続書き込みを

行っている場合でも、読み取りデータを処理する時間が確実に残っている必要があります。これを設定すると、書き込み操作が 1 回行われるたびに読み取り操作が 1 回行われるようになります。実行する書き込み操作がない場合、読み取りが連続処理されます。これにより、連続書き込みを行うアプリケーションが最適化され、データの送受信フローがよりバランスのとれたものとなります。

● **注記:** 本番環境で使用する前に、強化された書き込み最適化機能との互換性が維持されるようにアプリケーションのプロパティを設定することをお勧めします。

チャンネルのプロパティ - 詳細

このグループは、チャンネルの詳細プロパティを指定するときに使用します。すべてのドライバーがすべてのプロトコルをサポートしているわけではないので、サポートしていないデバイスには詳細グループが表示されません。

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		

「**非正規化浮動小数点処理**」: 非正規化値は無限、非数 (NaN)、または非正規化数として定義されます。デフォルトは「ゼロで置換」です。ネイティブの浮動小数点処理が指定されているドライバーはデフォルトで「未修正」になります。「非正規化浮動小数点処理」では、ドライバーによる非正規化 IEEE-754 浮動小数点データの処理方法を指定できます。オプションの説明は次のとおりです。

- 「**ゼロで置換**」: このオプションを選択した場合、ドライバーが非正規化 IEEE-754 浮動小数点値をクライアントに転送する前にゼロで置き換えることができます。
- 「**未修正**」: このオプションを選択した場合、ドライバーは IEEE-754 非正規化、正規化、非数、および無限の値を変換または変更せずにクライアントに転送できます。

● **注記:** ドライバーが浮動小数点値をサポートしていない場合や、表示されているオプションだけをサポートする場合、このプロパティは使用できません。チャンネルの浮動小数点正規化の設定に従って、リアルタイムのドライバータグ (値や配列など) が浮動小数点正規化の対象となります。たとえば、EFM データはこの設定の影響を受けません。

● 浮動小数点値の詳細については、サーバーのヘルプで「非正規化浮動小数点値を使用する方法」を参照してください。

「**デバイス間遅延**」: 通信チャンネルが同じチャンネルの現在のデバイスからデータを受信した後、次のデバイスに新しい要求を送信するまで待機する時間を指定します。ゼロ (0) を指定すると遅延は無効になります。

● **注記:** このプロパティは、一部のドライバー、モデル、および依存する設定では使用できません。

デバイスのプロパティ - 一般

デバイスは、通信チャンネル上の 1 つのターゲットを表します。ドライバーが複数のコントローラをサポートしている場合、ユーザーは各コントローラのデバイス ID を入力する必要があります。

Property Groups	<input type="checkbox"/> Identification	
General	Name	
Scan Mode	Description	
	Channel Assignment	
	Driver	
	Model	
	ID Format	Decimal
	ID	2

識別

「**名前**」: このプロパティでは、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:** わかりやすい名前をすること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の情報。

● 「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の名前。

「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」: このプロパティでは、この ID に関連付けられるデバイスのタイプを指定します。このドロップダウンメニューの内容は、使用されている通信ドライバーのタイプによって異なります。ドライバーによってサポートされていないモデルは無効になります。通信ドライバーが複数のデバイスモデルをサポートしている場合、デバイスにクライアントアプリケーションが 1 つも接続していない場合にのみモデル選択を変更できます。

● **注記:** 通信ドライバーが複数のモデルをサポートしている場合、ユーザーは物理デバイスに合わせてモデルを選択する必要があります。このドロップダウンメニューにデバイスが表示されない場合、ターゲットデバイスに最も近いモデルを選択します。一部のドライバーは "オープン" と呼ばれるモデル選択をサポートしており、ユーザーはターゲットデバイスの詳細を知らなくても通信できます。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。

「ID」: このプロパティでは、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。

動作モード

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

「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」: このオプションは、デバイスをシミュレーションモードにします。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。

デバイスのプロパティ - スキャンモード

「スキャンモード」では、デバイスとの通信を必要とする、サブスクリプション済みクライアントが要求したタグのスキャン速度を指定します。同期および非同期デバイスの読み取りと書き込みは可能なかぎりただちに処理され、「スキャンモード」のプロパティの影響を受けません。

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		

「スキャンモード」: 購読しているクライアントに送信される更新についてデバイス内のタグをどのようにスキャンするかを指定します。オプションの説明は次のとおりです。

- 「**クライアント固有のスキャン速度を適用**」: このモードでは、クライアントによって要求されたスキャン速度を使用します。
- 「**指定したスキャン速度以下でデータを要求**」: このモードでは、最大スキャン速度として設定されている値を指定します。有効な範囲は 10 から 99999990 ミリ秒です。デフォルトは 1000 ミリ秒です。
 - **注記**: サーバーにアクティブなクライアントがあり、デバイスのアイテム数とスキャン速度の値が増加している場合、変更はただちに有効になります。スキャン速度の値が減少している場合、すべてのクライアントアプリケーションが切断されるまで変更は有効になりません。
- 「**すべてのデータを指定したスキャン速度で要求**」: このモードでは、指定した速度で購読済みクライアント用にタグがスキャンされます。有効な範囲は 10 から 99999990 ミリ秒です。デフォルトは 1000 ミリ秒です。
- 「**スキャンしない、要求ポールのみ**」: このモードでは、デバイスに属するタグは定期的にポーリングされず、アクティブになった後はアイテムの初期値の読み取りは実行されません。更新のポーリングは、_DemandPoll タグに書き込むか、個々のアイテムについて明示的なデバイス読み取りを実行することによって、クライアントが行います。詳細については、サーバーのヘルプで「デバイス要求ポール」を参照してください。
- 「**タグに指定のスキャン速度を適用**」: このモードでは、静的構成のタグプロパティで指定されている速度で静的タグがスキャンされます。動的タグはクライアントが指定したスキャン速度でスキャンされます。

「**キャッシュからの初回更新**」: このオプションを有効にした場合、サーバーは保存 (キャッシュ) されているデータから、新たにアクティブ化されたタグ参照の初回更新を行います。キャッシュからの更新は、新しいアイテム参照が同じアドレス、スキャン速度、データ型、クライアントアクセス、スケール設定のプロパティを共有している場合にのみ実行できます。1 つ目のクライアント参照についてのみ、初回更新にデバイス読み取りが使用されます。デフォルトでは無効になっており、クライアントがタグ参照をアクティブ化したときにはいつでも、サーバーがデバイスから初期値の読み取りを試みます。

デバイスのプロパティ - タイミング

デバイスのタイミングのプロパティでは、エラー状態に対するデバイスの応答をアプリケーションのニーズに合わせて調整できます。多くの場合、最適なパフォーマンスを得るためにはこれらのプロパティを変更する必要があります。電気的に発生するノイズ、モデムの遅延、物理的な接続不良などの要因が、通信ドライバーで発生するエラーやタイムアウトの数に影響します。タイミングのプロパティは、設定されているデバイスごとに異なります。

プロパティグループ	☐ 通信タイムアウト	
一般	要求のタイムアウト (ミリ秒)	5000
スキャンモード	再試行回数	3
タイミン	☐ タイミング	
自動格下げ	要求間遅延 (ミリ秒)	0

通信タイムアウト

「**接続タイムアウト**」: このプロパティ (イーサネットベースのドライバーで主に使用) は、リモートデバイスとのソケット接続を確立するために必要な時間を制御します。デバイスの接続時間は、同じデバイスへの通常の通信要求よりも長くなるのがよくあります。有効な範囲は 1 から 30 秒です。デフォルトは通常は 3 秒ですが、各ドライバーの特性によって異なる場合があります。この設定がドライバーでサポートされていない場合、無効になります。

- **注記**: UDP 接続の特性により、UDP を介して通信する場合には接続タイムアウトの設定は適用されません。

「**要求のタイムアウト**」: このプロパティでは、ターゲットデバイスからの応答を待つのをいつやめるかを判断する際にすべてのドライバーが使用する間隔を指定します。有効な範囲は 50 から 9,999,999 ミリ秒 (167.6667 分) です。デフォルトは通常は 1000 ミリ秒ですが、ドライバーによって異なる場合があります。ほとんどのシリアルドライバーのデフォルトのタイムアウトは 9600 ボー以上のもので、低いボーレートに基づきます。低いボーレートでドライバーを使用している場合、データの取得に必要な時間が増えることを補うため、タイムアウト時間を増やします。

「**タイムアウト前の試行回数**」: このプロパティでは、ドライバーが通信要求を発行する回数を指定します。この回数を超えると、要求が失敗してデバイスがエラー状態にあると見なされます。有効な範囲は 1 から 10 です。デフォルトは通常は 3 ですが、各ドライバーの特性によって異なる場合があります。アプリケーションに設定される試行回数は、通信環境に大きく依存します。このプロパティは、接続の試行と要求の試行の両方に適用されます。

タイミング

「**要求間遅延**」: このプロパティでは、ドライバーがターゲットデバイスに次の要求を送信するまでの待ち時間を指定します。デバイスに関連付けられているタグおよび 1 回の読み取りと書き込みの標準のポーリング間隔がこれによってオーバーライドされます。この遅延は、応答時間が長いデバイスを扱う際や、ネットワークの負荷が問題である場合に役立ちます。デバイスの遅延を設定すると、そのチャンネル上のその他すべてのデバイスとの通信に影響が生じます。可能な場合、要求間遅延を必要とするデバイスは別々のチャンネルに分けて配置することをお勧めします。その他の通信プロパティ (通信シリアル化など) によってこの遅延が延長されることがあります。有効な範囲は 0 から 300,000 ミリ秒ですが、一部のドライバーでは独自の設計の目的を果たすために最大値が制限されている場合があります。デフォルトは 0 であり、ターゲットデバイスへの要求間に遅延はありません。

● **注記:** すべてのドライバーで「要求間遅延」がサポートされているわけではありません。使用できない場合にはこの設定は表示されません。

デバイスのプロパティ - 自動格下げ

自動格下げのプロパティを使用することで、デバイスが応答していない場合にそのデバイスを一時的にスキャン停止にできます。応答していないデバイスを一定期間オフラインにすることで、ドライバーは同じチャンネル上のほかのデバイスとの通信を引き続き最適化できます。停止期間が経過すると、ドライバーは応答していないデバイスとの通信を再試行します。デバイスが応答した場合はスキャンが開始され、応答しない場合はスキャン停止期間が再開します。

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

「**エラー時に格下げ**」: 有効にした場合、デバイスは再び応答するまで自動的にスキャン停止になります。

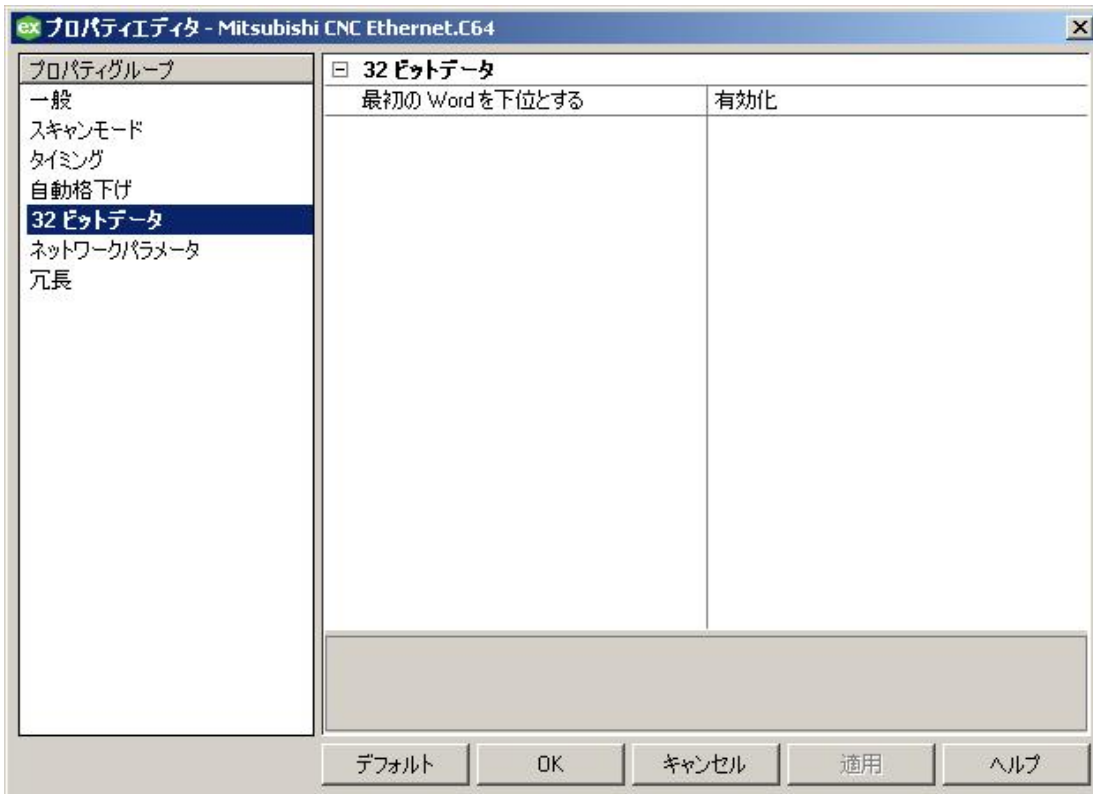
● **ヒント:** システムタグ `_AutoDemoted` を使用して格下げ状態をモニターすることで、デバイスがいつスキャン停止になったかを把握できます。

「**格下げまでのタイムアウト回数**」: デバイスをスキャン停止にするまでに要求のタイムアウトと再試行のサイクルを何回繰り返すかを指定します。有効な範囲は 1 から 30 回の連続エラーです。デフォルトは 3 です。

「**格下げ期間**」: タイムアウト値に達したときにデバイスをスキャン停止にする期間を指定します。この期間中、そのデバイスには読み取り要求が送信されず、その読み取り要求に関連するすべてのデータの品質は不良に設定されます。この期間が経過すると、ドライバーはそのデバイスのスキャンを開始し、通信での再試行が可能になります。有効な範囲は 100 から 3600000 ミリ秒です。デフォルトは 10000 ミリ秒です。

「**格下げ時に要求を破棄**」: スキャン停止期間中に書き込み要求を試行するかどうかを選択します。格下げ期間中も書き込み要求を必ず送信するには、無効にします。書き込みを破棄するには有効にします。サーバーはクライアントから受信した書き込み要求をすべて自動的に破棄し、イベントログにメッセージを書き込みません。

デバイスのプロパティ - 32 ビットデータ



「最初の Word を下位とする」: 最初のデータ Word が環境下で (上位ではなく) 下位であることを示します。三菱デバイスでは 32 ビットデータ型に 2 つの連続するレジスタアドレスが使用されます。ドライバーが最初の Word を 32 ビット値の下位 Word とするか上位 Word とするかを指定できます。デフォルトでは「最初の Word を下位とする」が「有効化」になっています。

● **注記:** デバイスにアクティブな参照がある間はこのプロパティを変更できません。

デバイスのプロパティ - ネットワークパラメータ



「ポート番号」: このプロパティでは宛先 CNC (またはマルチレイヤーネットワークが要求を受信するよう設定されている場合にはゲートウェイデバイス) の UDP ポートを指定します。デフォルトの設定は 5001 です。

● 注記: デフォルト設定はプログラミングツールとモニタリングツールで必ず使用可能なので、デフォルト設定を使用することをお勧めします。

「ソースネットワーク」: このパラメータでは、PC が存在するソースネットワークの番号を指定します。有効な範囲は 1 から 239 です。デフォルトの設定は 1 です。この設定はドライバーが (ゲートウェイデバイスなしで) 直接 CNC と通信するかどうかとは関係ありません。

「ソースステーション」: このパラメータでは、PC に割り当てられているステーション番号を指定します。有効な範囲は 1 から 239 です。デフォルトの設定は 1 です。ソースネットワーク上のすべてのデバイスに一意のステーション番号を割り当てる必要があります。この設定はドライバーが (ゲートウェイデバイスなしで) 直接 CNC と通信するかどうかとは関係ありません。

「宛先ネットワーク」: このパラメータでは、CNC が存在するネットワークの番号を指定します。有効な範囲は 0 から 239 です。デフォルトの設定は 1 です。この設定はドライバーが (ゲートウェイデバイスなしで) 直接 CNC と通信するかどうかとは関係ありません。

「宛先ステーション」: このパラメータでは、CNC に割り当てられているステーション番号を指定します。有効な範囲は 0 から 239 です。デフォルトの設定は 1 です。宛先ネットワーク上のすべてのデバイスに一意のステーション番号を割り当てる必要があります。この設定はドライバーが (ゲートウェイデバイスなしで) 直接 CNC と通信するかどうかとは関係ありません。

デバイスのプロパティ - 冗長

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

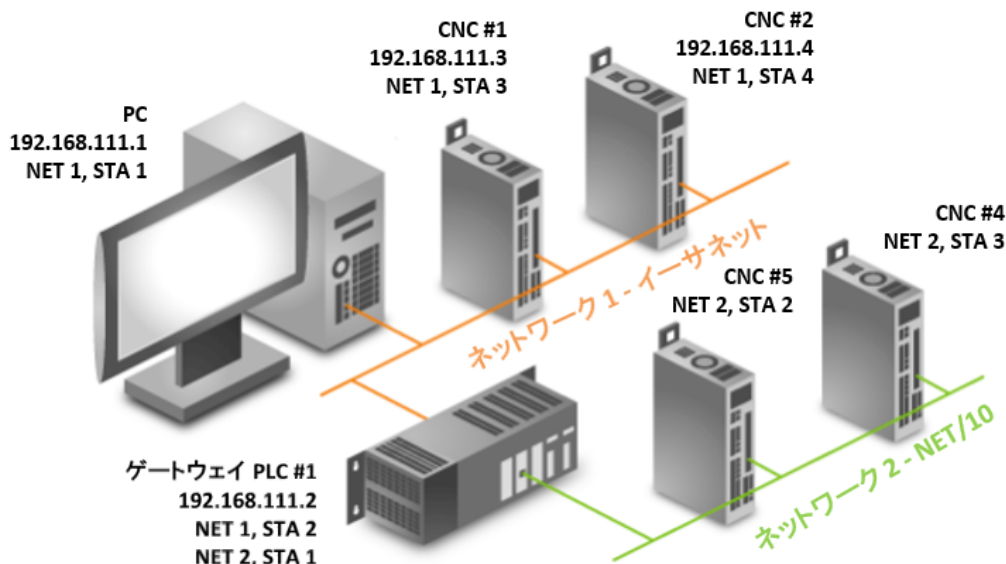
冗長設定はメディアレベルの冗長プラグインで使用できます。

● 詳細については、Web サイトまたはユーザーマニュアルを参照するか、営業担当者までお問い合わせください。

マルチレベルネットワーク

このドライバーを使用してリモートネットワーク上のデバイスと通信できます。以下に示す例では、CNC 1 と CNC 2 はローカルイーサネットネットワーク上にあります。CNC 3 と CNC 4 はネットワーク 2 - NET/10 上にあります。PLC 1 は 2 つのネットワークを接続するリレーデバイスとして機能します。

● 関連項目: [デバイスの設定](#)



ゲートウェイには AJ71QE71 イーサネット モジュールと NET/10 モジュールが組み込まれています。CNC 1 と CNC 2 には AJ 71QE 71 イーサネット モジュールと NET/10 モジュールが組み込まれています。CNC 3 と CNC 2 には NET/10 モジュールが組み込まれています。この例では、以下の表に示すデバイス ID を使用してサーバープロジェクト内に 4 つのデバイスが作成されています。

CNC	デバイス ID	SRC NET	SRC STA	DST NET	DST STA	コメント
1	192.168.111.2	1	1	1	3	直接
2	192.168.111.2	1	1	1	4	直接
3	192.168.111.2	1	1	2	2	PLC 1 経由
4	192.168.111.2	1	1	2	3	PLC 1 経由

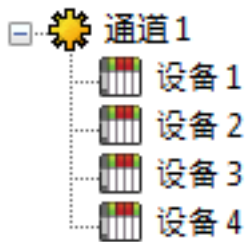
ユーザーはゲートウェイのイーサネットカードを設定できます (オープン方式 UDP と IP 192.168.111.2)。宛先 IP (255.255.255.255) と宛先ポート (0xFFFF) を使用して、PC によって使用される可能性がある任意の IP とポートに対応できます。

● 注記: リレーデバイスがリモートデバイスとの間の読み取り書き込みの失敗について報告するまでに 5 秒以上かかることがあります。リモートデバイスの要求タイムアウトを適宜設定することをお勧めします。

通信の最適化

Mitsubishi CNC Ethernet ドライバーは、システム全体のパフォーマンスへの影響を最小限に抑えながら最大のパフォーマンスが得られるように設計されています。このドライバーは高速ですが、このアプリケーションを制御および最適化して最大のパフォーマンスを得るためのガイドラインがあります。

このサーバーでは、Mitsubishi CNC Ethernet ドライバーなどの通信プロトコルのことをチャンネルと呼びます。アプリケーションで定義されている各チャンネルは、サーバーでの個々の実行パスを表します。チャンネルが定義された後、そのチャンネルの下に一連のデバイスを定義できます。これらのデバイスそれぞれが、データの収集元となる単一のイーサネットデバイスを表します。このアプローチに従ってアプリケーションを定義することで高いパフォーマンスが得られますが、ドライバーやネットワークがフルに利用されるわけではありません。単一のチャンネルを使用して構成されているアプリケーションの表示例を次に示します。



デバイスそれぞれが単一のデバイスチャンネルの下に表示されます。この構成では、ドライバーは効果的な速度で情報を収集するために、できるだけ速やかにあるデバイスから次のデバイスに移動する必要があります。さらにデバイスが追加されたり、1つのデバイスからより多くの情報が要求されたりするにしがたい、全体的な更新レートが低下していきます。

Mitsubishi CNC Ethernet ドライバーがチャンネルを1つだけ定義可能な場合、上に示した例が唯一可能なオプションとなりますが、このドライバーは最大 256 チャンネルまで定義できます。複数のチャンネルを使用して複数の要求をネットワークに同時に発行することで、データ収集のワークロードが分散されます。パフォーマンスを改善するために同じアプリケーションを複数のチャンネルを使用して構成した場合の例を次に示します。



ここではそれぞれのデバイスが各自のチャンネルの下に定義されています。この新しい構成では、各デバイスからのデータ収集タスクごとに1つの実行パスが割り当てられます。アプリケーションのデバイスの数が 256 以下である場合、まさにここで示したように最適化できます。

アプリケーションのデバイスの数が 256 を超える場合でもパフォーマンスは改善されます。デバイスの数は 256 以下であるのが理想的ですが、そうでない場合でもアプリケーションは追加のチャンネルから恩恵を受けます。デバイスの負荷を 256 個のチャンネルすべてに分散してもサーバーはデバイスを切り替えますが、単一のチャンネルで処理するデバイスの数ははるかに少なくなります。

データ型の説明

Mitsubishi CNC Ethernetドライバーでは次のデータ型がサポートされています。

データ型	説明
Word	符号なし 16 ビット値 ビット 0 が下位ビット ビット 15 が上位ビット
Short	符号付き 16 ビット値 ビット 0 が下位ビット ビット 14 が上位ビット ビット 15 が符号ビット
DWord	符号なし 32 ビット値 ビット 0 が下位ビット ビット 31 が上位ビット
Long*	符号付き 32 ビット値 ビット 0 が下位ビット ビット 30 が上位ビット ビット 31 が符号ビット
Long型の例	レジスタ 40001 が Long 型として指定されている場合、レジスタ 40001 のビット 0 は 32 ビットデータ型のビット 0 になり、レジスタ 40002 のビット 15 は 32 ビットデータ型のビット 31 になります。これが選択されていない場合にはこの逆になります。
Float*	32 ビット浮動小数点値
Floatの例	レジスタ 40001 が Float として指定されている場合、レジスタ 40001 のビット 0 は 32 ビットデータ型のビット 0 になり、レジスタ 40002 のビット 15 は 32 ビットデータ型のビット 31 になります。これが選択されていない場合にはこの逆になります。

ドライバーは1つ目のレジスタを下位 Word、2つ目のレジスタを上位 Word とすることで、連続する2つのレジスタを単精度値として解釈します。これが選択されていない場合にはこの逆になります。

アドレスの説明

アドレスの様子は使用されているモデルによって異なります。動的に定義されるタグのデフォルトのデータ型を太字で示しています。

デバイスタイプ	範囲*	データ型	アクセス
入力	X0000-X1FFF (16 進数)	Boolean 、Short、Word、Long、DWord	読み取り/書き込み
出力	Y0000-X1FFF (16 進数)	Boolean 、Short、Word、Long、DWord	読み取り/書き込み
リンクリレー	B0000-B1FFF (16 進数)	Boolean 、Short、Word、Long、DWord	読み取り/書き込み
特殊リンクリレー	SB0000-SB01FF (16 進数)	Boolean 、Short、Word、Long、DWord	読み取り/書き込み
内部リレー	M0000-M10239	Boolean 、Short、Word、Long、DWord	読み取り/書き込み
特殊内部リレー	SM0000-SM1023	Boolean 、Short、Word、Long、DWord	読み取り/書き込み
ラッチリレー	L0000-L0511	Boolean 、Short、Word、Long、DWord	読み取り/書き込み
アナンシエータリレー	F0000-F1023	Boolean 、Short、Word、Long、DWord	読み取り/書き込み
タイマー接点	TS0000-TS0703	Boolean 、Short、Word、Long、DWord	読み取り/書き込み
タイマーコイル	TC0000-TC0703	Boolean 、Short、Word、Long、DWord	読み取り/書き込み
カウンタ接点	CS0000-CS0255	Boolean 、Short、Word、Long、DWord	読み取り/書き込み
カウンタコイル	CC0000-CC0255	Boolean 、Short、Word、Long、DWord	読み取り/書き込み

*開始アドレスが有効なメモリ範囲内にある場合、このデバイスはそのメモリ範囲を超えるブロック読み取りにตอบสนองします。その場合、デバイスはこのメモリ範囲外のすべての値にゼロを返します。

● **注記:** Boolean 型のいずれのデバイスにも Short 型、Word 型、Long 型、DWord 型としてアクセスできます。ただし、デバイスが 16 ビット境界でアドレス指定されている必要があります。

デバイスタイプ	範囲	データ型	アクセス
タイマーの値	TN0000-TN0703	Short 、 Word	読み取り/書き込み
カウンタの値	CN0000-CN0255	Short、 Word	読み取り/書き込み
データレジスタ	D00000-D08191 D00000-D08190	Short 、 Word 、Long、DWord、Float	読み取り/書き込み
データレジスタのビットアクセス	D00000.00-D08191.15* D00000.00-D08190.31*	Short 、 Word 、 Boolean** 、Long、DWord	読み取り/書き込み
String データレジスタ、HiLo	DSH00000.002-DSH08190.002	String	読み取

デバイスタイプ	範囲	データ型	アクセス
バイトオーダリングでのアクセス	DSH00000.128-DSH08127.128 ピリオドの後ろのビット番号を使用して2-128バイトの文字列長を指定します。長さは偶数でなければなりません。		リ/書き込み
String データレジスタ、LoHi バイトオーダリングでのアクセス	DSL00000.002-DSL08190.002 DSL00000.128-DSL08127.128 ピリオドの後ろのビット番号を使用して2-128バイトの文字列長を指定します。長さは偶数でなければなりません。	String	読み取り/書き込み
特殊データレジスタ	SD0000-SD1023 SD0000-SD1022	Short、 Word Long、 DWord、 Float	読み取り/書き込み
ビットアクセス特殊データレジスタ	SD0000.00-SD1023.15* SD0000.00-SD1022.31*	Short、 Word、 Boolean** Long、 DWord	読み取り/書き込み
リンクレジスタ	W0000-W1FFF (16進数) W0000-W1FFE (16進数)	Short、 Word Long、 DWord、 Float	読み取り/書き込み
ビットアクセスリンクレジスタ	W0000.00-W1FFF.15* W0000.00-W1FFE.31*	Short、 Word、 Boolean** Long、 DWord	読み取り/書き込み
String リンクレジスタ、HiLo バイトオーダリングでのアクセス	WSH0000.002-WSH1FFE.002 WSH0000.128-WSH1FBF.128 ピリオドの後ろのビット番号を使用して2-128バイトの文字列長を指定します。長さは偶数でなければなりません。	String	読み取り/書き込み
String リンクレジスタ、LoHi バイトオーダリングでのアクセス	WSL0000.002-WSL1FFE.002 WSL0000.128-WSL1FBF.128 ピリオドの後ろのビット番号を使用して2-128バイトの文字列長を指定します。長さは偶数でなければなりません。	String	読み取り/書き込み
特殊リンクレジスタ	SW0000-SW01FF (16進数) SW0000-SW01FE (16進数)	Short、 Word Long、 DWord、 Float	読み取り/書き込み
特殊リンクレジスタ ビットアクセス	SW0000.00-SW01FF.15* SW0000.00-SW01FE.31*	Short、 Word、 Boolean** Long、 DWord	読み取り/書き込み
ファイルレジスタ	R00000-R13311 R00000-R13310	Short、 Word Long、 DWord、 Float	読み取り/書き込み
ビットアクセスファイルレジスタ	R00000.00-R13311.15* R00000.00-R13310.31*	Short、 Word、	読み取り/書き込み

デバイスタイプ	範囲	データ型	アクセス
		Boolean** Long、 DWord	込み
String ファイルレジスタ アクセス HiLo バイトオーダリング	RSH0000.002-RSH13310.002 RSH0000.128-RSH13183.128 ピリオドの後ろのビット番号を使用して 2-128 バイトの文字列長を指定します。長さは偶数でなければなりません。	String	読み取り/書き込み
String ファイルレジスタ アクセス LoHi バイトオーダリング	RSL0000.002-RSL13310.002 RSL0000.128-RSL.128 ピリオドの後ろのビット番号を使用して 2-128 バイトの文字列長を指定します。長さは偶数でなければなりません。	String	読み取り/書き込み
インデックスレジスタ	Z00-Z13 Z00-Z12	Short、 Word Long、 DWord、 Float	読み取り/書き込み
ビットアクセスインデックスレジスタ	Z00.00-Z13.15* Z00.00-Z12.31*	Short、 Word、 Boolean** Long、 DWord	読み取り/書き込み

*レジスタメモリの場合、Short、Word、DWord、Long、および Boolean データ型では、特定の値のビットを参照するため、オプションの .bb (ドットビット) をアドレスの末尾に追加できます。オプションのビットの有効な範囲は Short、Word、Boolean では 0 から 15 であり、Long 型と DWord 型では 0 から 31 です。文字列ではビット番号を使用して長さを指定します。D メモリ内の文字列の有効な長さは 2 から 128 バイトです。さらに、文字列の長さは偶数でなければなりません。Float 型ではビット操作はサポートされません。ビット番号は必ず 10 進表記で指定します。

**レジスタメモリに Boolean としてアクセスする場合、ビット番号を指定する必要があります。

配列へのアクセス

すべてのタイプのデバイスに Short、Word、Long、DWord、または Float フォーマットの配列としてアクセスできます。配列のサイズはデータ型とデバイスタイプによって異なります。あらゆるタイプのレジスタデバイスが、Short および Word 型では最大 254 個の要素、Long、DWord、Float では 127 個の要素にアクセスできます。あらゆるタイプのビットメモリが、Short および Word 型では最大 125 個の要素、Long、DWord、Float では 62 個の要素にアクセスできます。配列は 1 次元または 2 次元です。次元にかかわらず、事前に定義されている上限を配列のサイズが超えてはいけません。通常のデバイス参照上に配列表記を追加すると配列に入力されます。

● **注記:** 配列タグ (すべてのデバイスタイプ) のデフォルトは Word 型です。

例

1. D100[4] 1 次元配列には次のレジスタアドレスが含まれています: D100、D101、D102、D103。
2. M016[3][4] 2 次元配列には次のデバイスアドレスが Word 型として含まれています: M016、M032、M048、M064、M080、M096、M112、M128、M144、M160、M176、M192 3 行 x 4 列 = 12 Word 12 x 16 (Word) = 合計 192 ビット。

その他のデバイスの例

1. X デバイスマemoryに Word 型としてアクセス: X??? ここで、??? は 16 ビット境界に基づく 16 進数です (010、020、030 など)。
2. M デバイスマemoryに Long 型としてアクセス: M???? ここで、???? は 16 ビット境界に基づく 10 進数です (0、16、32、48 など)。

イベント ログメッセージ

次の情報は、メインユーザーインターフェースの「イベントログ」枠に記録されたメッセージに関するものです。「イベントログ」詳細ビューのフィルタと並べ替えについては、サーバーのヘルプを参照してください。サーバーのヘルプには共通メッセージが多数含まれているので、これらも参照してください。通常は、可能な場合、メッセージのタイプ (情報、警告) とトラブルシューティングに関する情報が提供されています。

デバイスでのタグの読み取りに失敗しました。| 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:

指定されているアドレスはデバイスの範囲外です。

解決策:

デバイスでサポートされているアドレス範囲を確認し、それに応じてタグの構成を修正してください。

デバイスでのタグの書き込みに失敗しました。デバイスがエラーコードを返しました。| タグアドレス = '<アドレス>'、エラーコード = <コード>。

エラータイプ:

警告

考えられる原因:

エラーコードはエラーメッセージの理由を示しています。

解決策:

エラーコードについてはドキュメントを参照してください。

デバイスでのブロック読み取りに失敗しました。デバイスはエラーコードを返しました。| ブロックサイズ = <数値> (ポイント)、ブロック開始アドレス = '<アドレス>'、エラーコード = <コード>。

エラータイプ:

警告

考えられる原因:

エラーコードはエラーメッセージの理由を示しています。

解決策:

エラーコードについてはドキュメントを参照してください。

デバイスでのタグの書き込みに失敗しました。フレーミングエラー。| 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:

ソースステーション番号などのフィールドの値が無効であるパケットを受信しました。

デバイスでのタグの書き込みに失敗しました。デバイスが不正なトランザクション ID を返しました。| タグアドレス = '<アドレス>'。

エラータイプ:

警告

デバイスでのブロック読み取りに失敗しました。フレーミングエラー。| ブロックサイズ = <数値> (ポイント)、ブロック開始アドレス = '<アドレス>'。

エラータイプ:

警告

考えられる原因:

ソースステーション番号などのフィールドの値が無効であるパケットを受信しました。

デバイスでのブロック読み取りに失敗しました。デバイスが不正なトランザクション ID を返しました。| ブロックサイズ = <数値> (ポイント)、ブロック開始アドレス = '<アドレス>'。

エラータイプ:

警告

デバイスでのタグの書き込みに失敗しました。接続エラー。| 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:

このエラーは Winsock エラーによるものです (ソケット作成の失敗など)。

デバイスでのブロック読み取りに失敗しました。接続エラー。| ブロックサイズ = <数値> (ポイント)、ブロック開始アドレス = '<アドレス>'。

エラータイプ:

警告

考えられる原因:

このエラーは Winsock エラーによるものです (ソケット作成の失敗など)。

索引

D

DWord 15

I

ID 7

L

Long 15

S

Short 15

W

Word 15

あ

アドレスの説明 16

い

イベントログメッセージ 19

え

エラー時に格下げ 9

き

キャッシュからの初回更新 8

さ

サポートされるデバイス 4

し

シミュレーション 7

す

スキャンしない、要求ボールのみ 8

スキャンモード 8

そ

ソースステーション 11

ソースネットワーク 11

た

タイムアウト前の試行回数 9

タグに指定のスキャン速度を適用 8

ち

チャンネル割り当て 7

て

データコレクション 7

データ型の説明 15

デバイスでのタグの書き込みに失敗しました。デバイスがエラーコードを返しました。| タグアドレス = '<アドレス>'、エラーコード = <コード>。 19

デバイスでのタグの書き込みに失敗しました。デバイスが不正なトランザクション ID を返しました。| タグアドレス = '<アドレス>'。 19

デバイスでのタグの書き込みに失敗しました。フレーミングエラー。| タグアドレス = '<アドレス>'。 19

デバイスでのタグの書き込みに失敗しました。接続エラー。| タグアドレス = '<アドレス>'。 20

デバイスでのタグの読み取りに失敗しました。| タグアドレス = '<アドレス>'。 19

デバイスでのブロック読み取りに失敗しました。デバイスが不正なトランザクション ID を返しました。| ブロックサイズ = <数値> (ポイント)、ブロック開始アドレス = '<アドレス>'。 20

デバイスでのブロック読み取りに失敗しました。デバイスはエラーコードを返しました。| ブロックサイズ = <数値> (ポイント)

ト)、ブロック開始アドレス = '<アドレス>', エラーコード = <コード>。 19
デバイスでのブロック読み取りに失敗しました。フレーミングエラー。| ブロックサイズ = <数値> (ポイント)、ブロック開始
アドレス = '<アドレス>'。 20
デバイスでのブロック読み取りに失敗しました。接続エラー。| ブロックサイズ = <数値> (ポイント)、ブロック開始アドレ
ス = '<アドレス>'。 20

と

ドライバー 7

ね

ネットワークパラメータ 10

は

バイナリ 4

ほ

ポート 11

ま

マルチレベルネットワーク 13

も

モデル 4, 7

嬢

宛先ステーション 11

宛先ネットワーク 11

盛

一般 6

梱

概要 3

柜

格下げまでのタイムアウト回数 9

格下げ期間 9

格下げ時に要求を破棄 9

髙

最初のWordを下位とする 10

膊

自動格下げ 9

諸

識別 6

儷

冗長 11

捅

接続のタイムアウト 8

覘

設定 4

辺

通信タイムアウト 8-9

通信の最適化 14

通信プロトコル 4

創

動作モード 7

厭

名前 6

裕

要求のタイムアウト 9

要求間遅延 9