

# Omron Host Linkドライバー

© 2021 PTC Inc. All Rights Reserved.

# 目次

<b>Omron Host Link ドライバー</b> .....	<b>1</b>
<b>目次</b> .....	<b>2</b>
Omron Host Link ドライバー .....	3
概要 .....	3
<b>設定</b> .....	<b>4</b>
チャンネルのプロパティ- 一般 .....	5
タグ数 .....	5
チャンネルのプロパティ- シリアル通信 .....	6
チャンネルのプロパティ- 書き込み最適化 .....	8
チャンネルのプロパティ- 詳細 .....	9
デバイスのプロパティ- 一般 .....	9
デバイスのプロパティ- スキャンモード .....	11
デバイスのプロパティ- タイミング .....	12
デバイスのプロパティ- 自動格下げ .....	13
デバイスのプロパティ- 文字間遅延 .....	13
デバイスのプロパティ- 冗長 .....	15
<b>データ型の説明</b> .....	<b>16</b>
<b>アドレスの説明</b> .....	<b>17</b>
C20H のアドレス指定 .....	17
C200H のアドレス指定 .....	20
CQM1 のアドレス指定 .....	23
オープンアドレス指定 .....	26
<b>イベントログメッセージ</b> .....	<b>32</b>
要求された型と受信した型が一致しません。  タグアドレス = '<アドレス>', 要求された型 = <タイプ>, 受信した型 = <タイプ>。 .....	32
ブロックに不良アドレスがあります。  タグアドレス = '<アドレス>', ブロック = <start> ~ <end>。 .....	32
エラーマスクの定義 .....	32
<b>索引</b> .....	<b>33</b>

## Omron Host Linkドライバー

---

ヘルプバージョン 1.039

### 目次

#### 概要

Omron Host Linkドライバーとは

#### 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

#### 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

Omron Host Link 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

Omron Host Linkドライバー で生成されるメッセージ

### 概要

---

Omron Host Linkドライバー は、オムロンホストリンクデバイスを次のような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。HMI、SCADA、Historian、MES、ERP など。これは、SYSMAC C シリーズデバイスで使用するためのものです。

## 設定

---

### サポートされるデバイス

C20H  
C200H  
CQM1  
オープン

### 通信プロトコル

オムロン上位リンク

### サポートされる通信パラメータ

ボーレート: 9600  
パリティ: 偶数または奇数  
データビット: 7 または 8  
ストップビット: 1 または 2

### 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 100 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 32 です。

### イーサネットカプセル化

このドライバーではイーサネットカプセル化がサポートされているため、ドライバーはターミナルサーバーを使用してイーサネットネットワークに接続されているシリアルデバイスとの通信が可能です。これはチャンネルプロパティを介して設定できます。詳細については、サーバーのヘルプドキュメントを参照してください。

### フロー制御

RS232/RS485 コンバータを使用している場合、必要なフロー制御のタイプはコンバータの要件によって異なります。コンバータには、フロー制御を必要としないものと、RTS フローを必要とするものがあります。コンバータのフロー要件についてはコンバータのドキュメントを参照してください。自動フロー制御を備えた RS485 コンバータが推奨されます。

#### ● 注記:

1. 製造メーカーから供給されている通信ケーブルを使用している場合、チャンネルプロパティでフロー制御の設定として「RTS」または「RTS 常時」を選択する必要があることがあります。
2. 適切なフロー制御が行われていないプラットフォームで実行している場合、サーバーの通信設定でフロー制御を設定する必要があることがあります。

## 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。

プロパティグループ	<input type="checkbox"/> 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	<input type="checkbox"/> 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。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。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、allows the usage of statistics tags that provide feedback to client applications regarding the operation of the channel. 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、サーバーのヘルプの「通信診断」および「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。

## チャンネルのプロパティ - シリアル通信

シリアル通信のプロパティはシリアルドライバーで設定でき、選択されているドライバー、接続タイプ、オプションによって異なります。使用可能なプロパティのスーパーセットを以下に示します。

次のいずれかのセクションをクリックしてください: [接続タイプ](#)、[シリアルポートの設定](#) または [イーサネット設定](#)、および [実行動作](#)。

● **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。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。

プロパティグループ	<input type="checkbox"/> <b>接続タイプ</b>	
一般	物理メディア	COM ポート
<b>シリアル通信</b>	共有	いいえ
書き込み最適化	<input type="checkbox"/> <b>シリアルポートの設定</b>	
詳細	COM ID	3
通信シリアル化	ボーレート	19200
リンク設定	データビット	8
	パリティ	なし
	ストップビット	1
	フロー制御	なし
	<input type="checkbox"/> <b>実行動作</b>	
	通信エラーを報告	有効化

### 接続タイプ

「物理メディア」: データ通信に使用するハードウェアデバイスのタイプを選択します。次のオプションがあります: 「モデム」、「イーサネットカプセル化」、「COM ポート」、「なし」。デフォルトは「COM ポート」です。

1. 「なし」: 物理的な接続がないことを示すには「なし」を選択します。これによって [通信なしの動作](#) セクションが表示されます。
2. 「COM ポート」: [シリアルポートの設定](#) セクションを表示して設定するには、「COM ポート」を選択します。
3. 「モデム」: 通信に電話回線を使用する場合 ([モデム設定](#) セクションで設定)、「モデム」を選択します。
4. 「イーサネットカプセル化」: イーサネットカプセル化機能を使用して通信を行う場合は、このオプションを選択します。この機能については、[イーサネット設定](#) セクションを参照してください。
5. 「共有」: 現在の構成を別のチャンネルと共有するよう接続が正しく識別されていることを確認します。これは読み取り専用プロパティです。

### シリアルポートの設定

「COM ID」: チャンネルに割り当てられているデバイスと通信するときに使用する通信 ID を指定します。有効な範囲は 1 から 9991 から 16 です。デフォルトは 1 です。

「ボーレート」: 選択した通信ポートを設定するときに使用するボーレートを指定します。

「データビット」: データワードあたりのデータビット数を指定します。オプションは 5、6、7、8 です。

「パリティ」: データのパリティのタイプを指定します。オプションには「奇数」、「偶数」、「なし」があります。

「ストップビット」: データワードあたりのストップビット数を指定します。オプションは 1 または 2 です。

「フロー制御」: RTS および DTR 制御回線の利用方法を選択します。一部のシリアルデバイスと通信する際にはフロー制御が必要です。以下のオプションがあります。

- 「なし」: このオプションでは、制御回線はトグル(アサート)されません。
- 「DTR」: このオプションでは、通信ポートが開いてオンのままになっている場合に DTR 回線がアサートされます。

- 「RTS」: このオプションでは、バイトを転送可能な場合に RTS 回線がハイになります。バッファ内のすべてのバイトが送信されると、RTS 回線はローになります。これは通常、RS232/RS485 コンバータハードウェアで使用されます。
- 「RTS、DTR」: このオプションは DTR と RTS を組み合わせたものです。
- 「RTS 常時」: このオプションでは、通信ポートが開いてオンのままになっている場合に、RTS 回線がアサートされます。
- 「RTS 手動」: このオプションでは、「RTS 回線制御」で入力したタイミングプロパティに基づいて RTS 回線がアサートされます。これは、ドライバーが手動による RTS 回線制御をサポートしている場合 (またはプロパティが共有され、このサポートを提供するドライバーに 1 つ以上のチャンネルが属している場合) にのみ使用できます。「RTS 手動」を選択した場合、次のオプションから成る「RTS 回線制御」プロパティが追加されます。
  - 「事前オン」: このプロパティでは、データ転送のどれだけ前に RTS 回線を事前にオンにするかを指定します。有効な範囲は 0 から 9999 ミリ秒です。デフォルトは 10 ミリ秒です。
  - 「遅延オフ」: このプロパティでは、データ転送後に RTS 回線をハイのままにする時間を指定します。有効な範囲は 0 から 9999 ミリ秒です。デフォルトは 10 ミリ秒です。
  - 「ポーリング遅延」: このプロパティでは、通信のポーリングが遅延する時間を指定します。有効な範囲は 0 から 9999 です。デフォルトは 10 ミリ秒です。

● ヒント: 2 回線 RS 485 を使用している場合、通信回線上で "エコー" が発生することがあります。この通信はエコー除去をサポートしていないので、エコーを無効にするか、RS-485 コンバータを使用することをお勧めします。

## 実行動作

- 「通信エラーを報告」: 低レベル通信エラーに関するレポートを有効または無効にします。オンにした場合、低レベルのエラーが発生するとイベントログに書き込まれます。オフにした場合、通常の要求の失敗は書き込まれますが、これと同じエラーは書き込まれません。デフォルトは「有効化」です。
- 「アイドル接続を閉じる」: チャンネル上のクライアントによっていずれのタグも参照されなくなった場合、接続を閉じます。デフォルトは「有効化」です。
- 「クローズするまでのアイドル時間」: すべてのタグが除去されてから COM ポートを閉じるまでサーバーが待機する時間を指定します。デフォルトは 15 秒です。

## イーサネット設定

● 注記: すべてのシリアルドライバーがイーサネットカプセル化をサポートするわけではありません。このグループが表示されない場合、機能はサポートされていません。

イーサネットカプセル化は、イーサネットネットワーク上のターミナルサーバーに接続しているシリアルデバイスとの通信を可能にします。ターミナルサーバーは基本的には仮想のシリアルポートであり、イーサネットネットワーク上の TCP/IP メッセージをシリアルデータに変換します。メッセージが変換されると、ユーザーはシリアル通信をサポートする標準デバイスをターミナルサーバーに接続可能になります。ターミナルサーバーのシリアルポートが接続先のシリアルデバイスの要件に合うように適切に設定されている必要があります。詳細については、サーバーヘルプの「Using Ethernet Encapsulation」を参照してください。

- 「ネットワークアダプタ」: このチャンネルのイーサネットデバイスがバインドするネットワークアダプタを指定します。バインド先のネットワークアダプタを選択するか、OS がデフォルトを選択可能にします。
  - 一部のドライバーでは追加のイーサネットカプセル化プロパティが表示されることがあります。詳細については、[チャンネルのプロパティ - イーサネットカプセル化](#)を参照してください。

## モデム設定

- 「モデム」: 通信に使用するインストール済みモデムを指定します。
- 「接続タイムアウト」: 接続が確立される際に待機する時間を指定します。この時間を超えると読み取りまたは書き込みが失敗します。デフォルトは 60 秒です。
- 「モデムのプロパティ」: モデムハードウェアを設定します。クリックした場合、ベンダー固有のモデムプロパティが開きます。
- 「自動ダイヤル」: 電話帳内のエントリに自動ダイヤルできます。デフォルトは「無効化」です。詳細については、サーバーのヘルプで「モデム自動ダイヤル」を参照してください。
- 「通信エラーを報告」: 低レベル通信エラーに関するレポートを有効または無効にします。オンにした場合、低レベルのエラーが発生するとイベントログに書き込まれます。オフにした場合、通常の要求の失敗は書き込まれますが、これと同じエラーは書き込まれません。デフォルトは「有効化」です。

- ・「**アイドル接続を閉じる**」: チャネル上のクライアントによっていずれのタグも参照されなくなった場合、モデム接続を閉じます。デフォルトは「有効化」です。
- ・「**クローズするまでのアイドル時間**」: すべてのタグが除去されてからモデム接続を閉じるまでサーバーが待機する時間を指定します。デフォルトは 15 秒です。

### 通信なしの動作

- ・「**読み取り処理**」: 明示的なデバイス読み取りが要求された場合の処理を選択します。オプションには「無視」と「失敗」があります。「無視」を選択した場合には何も行われません。「失敗」を選択した場合、失敗したことがクライアントに通知されます。デフォルト設定は「無視」です。

## チャネルのプロパティ - 書き込み最適化

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をデバイスに遅延なく届ける必要があります。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することができます。

プロパティグループ	<input checked="" type="checkbox"/> <b>書き込み最適化</b>	
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
<b>書き込み最適化</b>		

### 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しています。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。モーメンタリブ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を使用します。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"/> <b>非正規化浮動小数点処理</b>	
一般	浮動小数点値	ゼロで置換
シリアル通信	<input type="checkbox"/> <b>デバイス間遅延</b>	
書き込み最適化	デバイス間遅延 (ミリ秒)	0
<b>詳細</b>		
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。

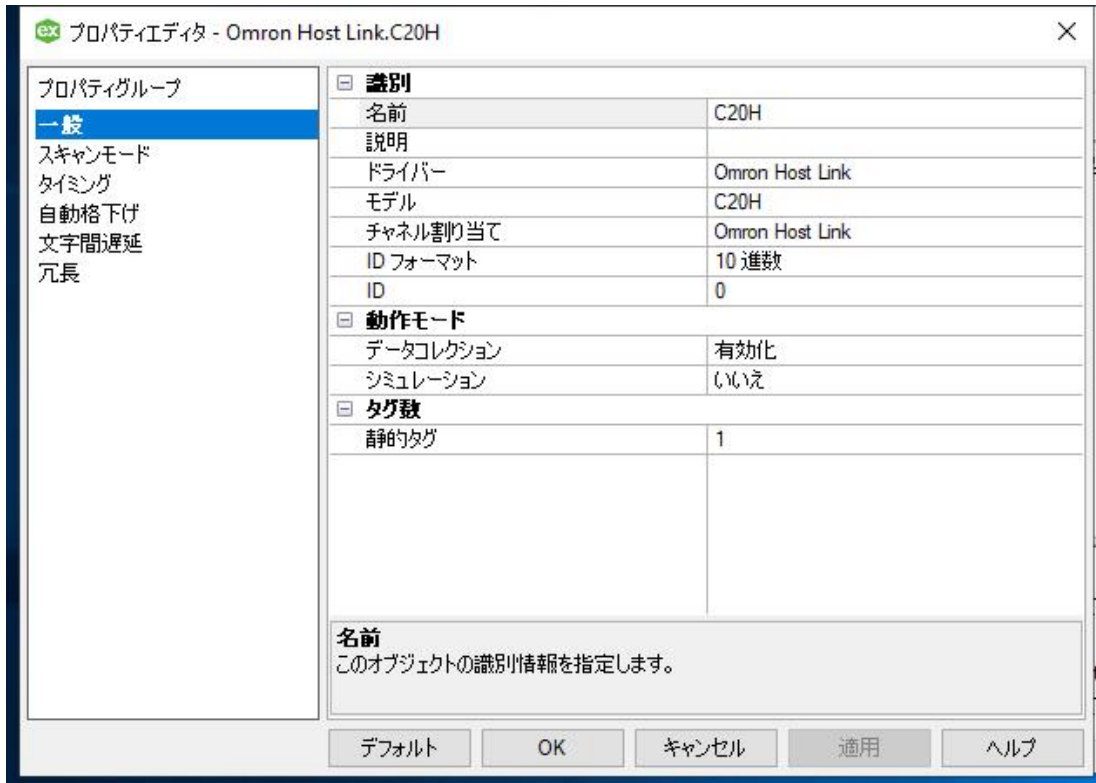
● **注記:** ドライバーが浮動小数点値をサポートしていない場合や、表示されているオプションだけをサポートする場合、このプロパティは無効になります。チャンネルの浮動小数点正規化の設定に従って、リアルタイムのドライバータグ (値や配列など) が浮動小数点正規化の対象となります。たとえば、EFM データはこの設定の影響を受けません。

● 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



## 識別

「名前」: このデバイスのユーザー定義の識別情報。

「説明」: このデバイスに関するユーザー定義の情報。

「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」: このデバイスのバージョン。

● FINS 通信 サービスをサポートするモデルのリストについては、製造メーカーの Web サイトを参照してください。

「ID フォーマット」: デバイス識別情報のフォーマット方法を選択します。オプションには「10 進数」、「8 進数」、「16 進数」があります。

「ID」: 「ID」にはデバイスの一意的ユニット番号を指定します。有効な範囲は 0 から 31 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」: 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限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
タイミング		

「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、OPC 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。電氣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。

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

### 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ことよくあります。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**」: すべてのドライバーがターゲットデバイスからの応答の完了を待機する時間を決定するために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**」: 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**」: 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。

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

## 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。

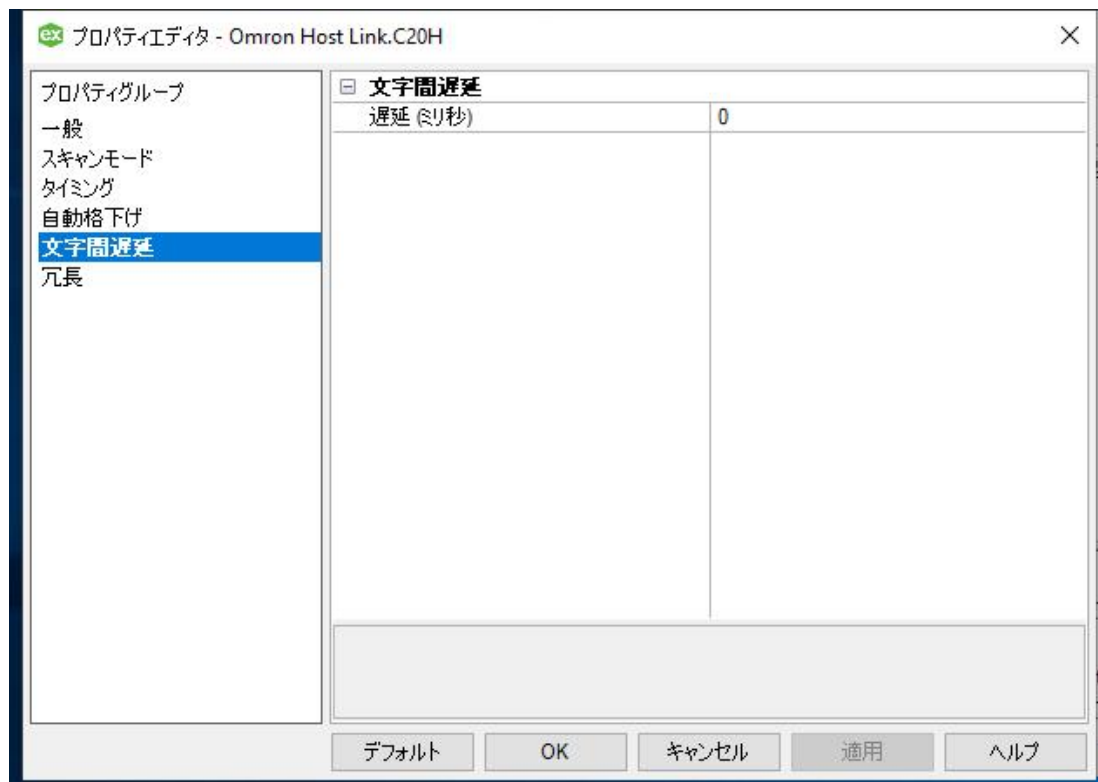
「格下げまでのタイムアウト回数」: デバイスをスキャン停止にするまでに要求のタイムアウトと再試行のサイクルを何回繰り返すかを指定します。有効な範囲は 1 から 30 回の連続エラーです。デフォルトは 3 です。

「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 - 文字間遅延

文字間遅延はモデル通信のためのものですが、一部のデバイスはモデムを介して通信する際に遅延を必要としません。この値によって生じる遅延によって通信速度が低下します。



「遅延」: ドライバーがオムロン製デバイスに送信する各文字間の制御遅延を指定します。有効な範囲は0から200ミリ秒です。デフォルトの設定は0ミリ秒です。

● ヒント: 一般的にはこの値をデフォルト設定のままにすることをお勧めしますが、モデム経由でこのドライバーを使用している場合には、正常な通信を行うためにこの値を調整しても構いません。1200 ボーのモデム接続では、標準的な設定は30ミリ秒です。

## 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。

## データ型の説明

データ型	説明
Boolean	1 ビット
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
BCD	2 バイトパックされた BCD 値の範囲は 0-9999 です。この範囲外の値には動作が定義されていません。
LBCD	4 バイトパックされた BCD 値の範囲は 0-99999999 です。この範囲外の値には動作が定義されていません。
Long	符号付き 32 ビット値
DWord	符号なし 32 ビット値
浮動小数点数	32 ビット浮動小数点値 ドライバーは 2 つ目のレジスタを上位 Word、1 つ目のレジスタを下位 Word とすることで、連続する 2 つのレジスタを浮動小数点値として解釈します。
Float の例	レジスタ DM100 が Float として指定されている場合、レジスタ DM100 のビット 0 は 32 ビット Float のビット 0 になり、レジスタ DM101 のビット 15 は 32 ビット Float のビット 31 になります。
String	Null 終端 ASCII 文字列。 最大 58 文字の文字列長、HiLo バイトオーダー、LoHi バイトオーダー、上位バイトのみ、下位バイトのみの選択がサポートされています。



## 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。対象のモデルのアドレス情報を取得するには、次のリストからリンクを選択してください。

[C20H のアドレス指定](#)

[C200H のアドレス指定](#)

[CQM1 のアドレス指定](#)

[オープンアドレス指定](#)

## C20H のアドレス指定

デフォルトのデータ型を太字で示しています。

●詳細については、[文字列のサポート](#)および[配列のサポート](#)を参照してください。

デバイスタイプ	範囲	データ型	アクセス
補助リレー	AR00-AR27 AR00-AR26 ARxx.00-ARxx.15	<b>Word</b> 、Short、BCD、Long、DWord、Float、LBCD  Boolean	読み取り/書き込み
String の補助リレー、HiLo バイトオーダー	AR00.56H-AR27.02H ピリオドの後ろの長さは2 から 56 文字の範囲の文字列長を示します	<b>String</b>	読み取り/書き込み
String の補助リレー、LoHi バイトオーダー	AR00.56L-AR27.02L ピリオドの後ろの長さは2 から 56 文字の範囲の文字列長を示します	<b>String</b>	読み取り/書き込み
String の補助リレー、各 Word の上位バイトのみ使用	AR00.28D-AR27.01D ピリオドの後ろの長さは1 から 28 文字の範囲の文字列長を示します	<b>String</b>	読み取り/書き込み
String の補助リレー、各 Word の下位バイトのみ使用	AR00.28E-AR27.01E ピリオドの後ろの長さは1 から 28 文字の範囲の文字列長を示します	<b>String</b>	読み取り/書き込み
データメモリ	DM0000-DM0999 DM0000-DM0998 DMxxxx.00-DMxxxx.15 DM1000-DM1999 DM1000-DM1998 DMxxxx.00-DMxxxx.15	<b>Word</b> 、Short、BCD、Long、DWord、Float、LBCD  Boolean  <b>Word</b> 、Short、BCD、Long、DWord、Float、LBCD  Boolean	読み取り/書き込み   読み取り専用
String のデータメモリ、HiLo バイトオーダー	DM0000.58H-DM0999.02H ピリオドの後ろの長さは2 から 58 文字の範囲の文字列長を示します DM1000.58H-DM1999.02H ピリオドの後ろの長さは2 から 58 文字の範囲の文字列長を示します	<b>String</b>   <b>String</b>	読み取り/書き込み   読み取り専用

デバイスタイプ	範囲	データ型	アクセス
	囲の文字列長を示します		
String のデータメモリ、LoHi バイト オーダー	DM0000.58L-DM0999.02L ピリオドの後ろの長さは2 から 58 文字の範囲の文字列長を示します	String	読み取り/ 書き込み
	DM1000.58L-DM1999.02L ピリオドの後ろの長さは2 から 58 文字の範囲の文字列長を示します	String	読み取り 専用
String のデータメモリ、各 Word の 上位バイトのみ使用	DM0000.29D-DM0999.01D ピリオドの後ろの長さは1 から 29 文字の範囲の文字列長を示します	String	読み取り/ 書き込み
	DM1000.29D-DM1999.01D ピリオドの後ろの長さは1 から 29 文字の範囲の文字列長を示します	String	読み取り 専用
String のデータメモリ、各 Word の 下位バイトのみ使用	DM0000.29E-DM0999.01E ピリオドの後ろの長さは1 から 29 文字の範囲の文字列長を示します	String	読み取り/ 書き込み
	DM1000.29E-DM1999.01E ピリオドの後ろの長さは1 から 29 文字の範囲の文字列長を示します	String	読み取り 専用
保持リレー	HR00-HR99	Word、Short、 BCD、Long、 DWord、Float、 LBCD  Boolean	読み取り/ 書き込み
	HR00-HR98		
	HRxx.00-HRxx.15		
String の保持リレー、HiLo バイト オーダー	HR00.58H-HR99.02H ピリオドの後ろの長さは2 から 58 文字の範囲の文字列長を示します	String	読み取り/ 書き込み
String の保持リレー、LoHi バイト オーダー	HR00.58L-HR99.02L ピリオドの後ろの長さは2 から 58 文字の範囲の文字列長を示します	String	読み取り/ 書き込み
String の保持リレー、各 Word の 上位バイトのみ使用	HR00.29D-HR99.01D ピリオドの後ろの長さは1 から 29 文字の範囲の文字列長を示します	String	読み取り/ 書き込み
String の保持リレー、各 Word の 下位バイトのみ使用	HR00.29E-HR99.01E ピリオドの後ろの長さは1 から 29 文字の範囲の文字列長を示します	String	読み取り/ 書き込み
内部リレー	IR000-IR255	Word、Short、 BCD、Long、 DWord、Float、 LBCD  Boolean	読み取り/ 書き込み
	IR000-IR254		
	IRxxx.00-IRxxx.15		

デバイスタイプ	範囲	データ型	アクセス
String の内部リレー、HiLo バイトオーダー	IR000.58H-IR255.02H ピリオドの後ろの長さは2 から 58 文字の範囲の文字列長を示します	String	読み取り/書き込み
String の内部リレー、LoHi バイトオーダー	IR000.58L-IR255.02L ピリオドの後ろの長さは2 から 58 文字の範囲の文字列長を示します	String	読み取り/書き込み
String の内部リレー、各 Word の上位バイトのみ使用	IR000.29D-IR255.01D ピリオドの後ろの長さは1 から 29 文字の範囲の文字列長を示します	String	読み取り/書き込み
String の内部リレー、各 Word の下位バイトのみ使用	IR000.29E-IR255.01E ピリオドの後ろの長さは1 から 29 文字の範囲の文字列長を示します	String	読み取り/書き込み
リンクリレー	LR00-LR63 LR00-LR62 LRxx.00-LRxx.15	Word、Short、BCD、Long、DWord、Float、LBCD Boolean	読み取り/書き込み
String のリンクリレー、HiLo バイトオーダー	LR00.58H-LR63.02H ピリオドの後ろの長さは2 から 58 文字の範囲の文字列長を示します	String	読み取り/書き込み
String のリンクリレー、LoHi バイトオーダー	LR00.58L-LR63.02L ピリオドの後ろの長さは2 から 58 文字の範囲の文字列長を示します	String	読み取り/書き込み
String のリンクリレー、各 Word の上位バイトのみ使用	LR00.29D-LR63.01D ピリオドの後ろの長さは1 から 29 文字の範囲の文字列長を示します	String	読み取り/書き込み
String のリンクリレー、各 Word の下位バイトのみ使用	LR00.29E-LR63.01E ピリオドの後ろの長さは1 から 29 文字の範囲の文字列長を示します	String	読み取り/書き込み
タイマー/カウンタ	RC000-RC511 RCxxx.00-RCxxx.15	Word、Short、BCD Boolean	読み取り/書き込み
タイマー/カウンタのステータス	TC000-TC511	Boolean	読み取り/書き込み

### 文字列のサポート

C20H モデルでは、補助リレー (AR)、データメモリ (DM)、保持リレー (HR)、内部リレー (IR)、リンクリレー (LR) を ASCII 文字列として読み書きできます。これらのタイプのデバイスを文字列データに使用している場合、各レジスタは2 バイト (2 文字) または1 バイトの ASCII データを格納できます。文字列を定義する際に、そのレジスタにおける ASCII データの順序、またはそのレジスタ内で使用するバイトを選択できます。

レジスタあたり2 バイトの ASCII データを使用する場合、文字列の長さは2 から 58 文字 (または AR では2 から 56) となり、ビット番号の位置に入力します。この長さは偶数として入力する必要があります。文字列によって指定されるレジスタの範囲がデバイスタイプの範囲を超えてはなりません。バイトオーダーはアドレスの末尾に "H" または "L" を付けることによって指定します。

レジスタあたり 1 バイトの ASCII データを使用する場合、文字列の長さは 1 から 29 文字 (または AR では 1 から 28) となり、ビット番号の位置に入力します。文字列によって指定されるレジスタの範囲がデバイスタイプの範囲を超えてはなりません。レジスタ内で使用するバイトはアドレスに "D" または "E" を追加することによって指定します。

**例**

- DM100 で開始し、長さが 50 バイト、HiLo バイトオーダーの文字列をアドレス指定するには、DM100.50H と入力します。
- DM110 で開始し、長さが 8 バイト、LoHi バイトオーダーの文字列をアドレス指定するには、DM110.08L と入力します。
- DM200 で開始し、長さが 15 バイト、上位バイトのみの文字列をアドレス指定するには、DM200.15D と入力します。
- DM220 で開始し、長さが 7 バイト、下位バイトのみの文字列をアドレス指定するには、DM220.07E と入力します。

**配列のサポート**

Boolean と String 以外のすべてのデータ型で配列がサポートされています。配列のアドレス指定には 2 つの方法があります。例ではデータメモリの位置が使用されています。

DMxxxx [行数] [列数]  
DMxxxx [列数]\*

\*この方法では "行数" が 1 であるものと見なされます。

行数と列数を掛けた値にデータサイズのバイト数を掛けた値が 116 バイトを超えてはなりません。これはプロトコルによる制限です。このドライバーは ASCII プロトコルを使用するため、Word、Short、BCD は 4 バイト、DWord、Long、LBCD、Float は 8 バイトあります。たとえば、4 X 7 の Word の配列では配列サイズが 28 Word x 4 バイト = 112 バイトになり、これは最大要求サイズである 116 バイトに収まります。

●32 ビット値 (DWord、Long、LBCD、Float) を修正する場合には注意してください。これらのデータ型を使用可能な各アドレスは、デバイス内で 1 Word ずつオフセットして開始しています。このため、DWord DM0 と DM1 は Word DM1 で重複します。DM0 に書き込んだ場合も DM1 に格納されている値が修正されます。これらのデータ型を使用する際には重複が生じないようにすることをお勧めします。たとえば、DWord を使用している場合、DM0、DM2、DM4 などを使用することで Word が重複しないようにします。

**C200H のアドレス指定**

デフォルトのデータ型を太字で示しています。

●詳細については、[文字列のサポート](#)および[配列のサポート](#)を参照してください。

デバイスタイプ	範囲	データ型	アクセス
補助リレー	AR000-AR999 AR000-AR998 ARxxx.00-ARxxx.15	<b>Word</b> 、Short、BCD、Long、DWord、Float、LBCD  Boolean	読み取り/書き込み
String の補助リレー、HiLo バイトオーダー	AR000.58H-AR999.02H ピリオドの後ろの長さは 2 から 58 文字の範囲の文字列長を示します	<b>String</b>	読み取り/書き込み
String の補助リレー、LoHi バイトオーダー	AR000.58L-AR999.02L ピリオドの後ろの長さは 2 から 58 文字の範囲の文字列長を示します	<b>String</b>	読み取り/書き込み
String の補助リレー、各 Word の上位バイトのみ使用	AR000.29D-AR999.01D ピリオドの後ろの長さは 1 から 29 文字の範囲の文字列長を示します	<b>String</b>	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
String の補助リレー、各 Word の下位バイトのみ使用	AR000.29E-AR999.01E ピリオドの後ろの長さは1から29文字の範囲の文字列長を示します	String	読み取り/書き込み
データメモリ	DM0000-DM9999 DM0000-DM9998 DMxxxx.00-DMxxxx.15	Word、Short、BCD、Long、DWord、Float、LBCD Boolean	読み取り/書き込み
String のデータメモリ、HiLo バイトオーダー	DM0000.58H-DM0999.02H ピリオドの後ろの長さは2から58文字の範囲の文字列長を示します	String	読み取り/書き込み
String のデータメモリ、LoHi バイトオーダー	DM0000.58L-DM0999.02L ピリオドの後ろの長さは2から58文字の範囲の文字列長を示します	String	読み取り/書き込み
String のデータメモリ、各 Word の上位バイトのみ使用	DM0000.29D-DM0999.01D ピリオドの後ろの長さは1から29文字の範囲の文字列長を示します	String	読み取り/書き込み
String のデータメモリ、各 Word の下位バイトのみ使用	DM0000.29E-DM0999.01E ピリオドの後ろの長さは1から29文字の範囲の文字列長を示します	String	読み取り/書き込み
保持リレー	HR000-HR999 HR000-HR998 HRxxx.00-HRxxx.15	Word、Short、BCD、Long、DWord、Float、LBCD Boolean	読み取り/書き込み
String の保持リレー、HiLo バイトオーダー	HR000.58H- HR999.02H ピリオドの後ろの長さは2から58文字の範囲の文字列長を示します	String	読み取り/書き込み
String の保持リレー、LoHi バイトオーダー	HR000.58L-HR999.02L ピリオドの後ろの長さは2から58文字の範囲の文字列長を示します	String	読み取り/書き込み
String の保持リレー、各 Word の上位バイトのみ使用	HR000.29D-HR999.01D ピリオドの後ろの長さは1から29文字の範囲の文字列長を示します	String	読み取り/書き込み
String の保持リレー、各 Word の下位バイトのみ使用	HR000.29E-HR999.01E ピリオドの後ろの長さは1から29文字の範囲の文字列長を示します	String	読み取り/書き込み
内部リレー	IR000-IR999 IR000-IR998 IRxxx.00-IRxxx.15	Word、Short、BCD、Long、DWord、Float、LBCD Boolean	読み取り/書き込み
String の内部リレー、HiLo バイト	IR000.58H-IR999.02H	String	読み取り

デバイスタイプ	範囲	データ型	アクセス
オーダー	ピリオドの後ろの長さは2から58文字の範囲の文字列長を示します		書き込み
String の内部リレー、LoHi バイトオーダー	IR000.58L-IR999.02L ピリオドの後ろの長さは2から58文字の範囲の文字列長を示します	String	読み取り 書き込み
String の内部リレー、各 Word の上位バイトのみ使用	IR000.29D-IR999.01D ピリオドの後ろの長さは1から29文字の範囲の文字列長を示します	String	読み取り 書き込み
String の内部リレー、各 Word の下位バイトのみ使用	IR000.29E-IR999.01E ピリオドの後ろの長さは1から29文字の範囲の文字列長を示します	String	読み取り 書き込み
リンクリレー	LR000-LR999 LR000-LR998 LRxxx.00-LRxxx.15	Word、Short、BCD、Long、DWord、Float、LBCD  Boolean	読み取り 書き込み
String のリンクリレー、HiLo バイトオーダー	LR000.58H-LR999.02H ピリオドの後ろの長さは2から58文字の範囲の文字列長を示します	String	読み取り 書き込み
String のリンクリレー、LoHi バイトオーダー	LR000.58L-LR999.02L ピリオドの後ろの長さは2から58文字の範囲の文字列長を示します	String	読み取り 書き込み
String のリンクリレー、各 Word の上位バイトのみ使用	LR000.29D-LR999.01D ピリオドの後ろの長さは1から29文字の範囲の文字列長を示します	String	読み取り 書き込み
String のリンクリレー、各 Word の下位バイトのみ使用	LR000.29E-LR999.01E ピリオドの後ろの長さは1から29文字の範囲の文字列長を示します	String	読み取り 書き込み
タイマー/カウンタ	RC000-RC999 RCxxx.00-RCxxx.15	Word、Short、BCD  Boolean	読み取り 書き込み
タイマー/カウンタのステータス	TC000-TC999	Boolean	読み取り 書き込み

### 文字列のサポート

C200H モデルでは、補助リレー (AR)、データメモリ (DM)、保持リレー (HR)、内部リレー (IR)、リンクリレー (LR) を ASCII 文字列として読み書きできます。これらのタイプのデバイスを文字列データに使用している場合、各レジスタは2バイト (2文字) または1バイトの ASCII データを格納できます。文字列を定義する際に、そのレジスタにおける ASCII データの順序、またはそのレジスタ内で使用するバイトを選択できます。

レジスタあたり2バイトの ASCII データを使用する場合、文字列の長さは2から58文字となり、ビット番号の位置に入力します。この長さは偶数として入力する必要があります。文字列によって指定されるレジスタの範囲がデバイスタイプの範囲を超えてはなりません。バイトオーダーはアドレスの末尾に "H" または "L" を付けることによって指定します。

レジスタあたり1バイトのASCIIデータを使用する場合、文字列の長さは1から29文字となり、ビット番号の位置に入力します。文字列によって指定されるレジスタの範囲がデバイスタイプの範囲を超えてはなりません。レジスタ内で使用するバイトはアドレスに"D"または"E"を追加することによって指定します。

#### 例

- DM100 で開始し、長さが50バイト、HiLoバイトオーダーの文字列をアドレス指定するには、DM100.50H と入力します。
- DM110 で開始し、長さが8バイト、LoHiバイトオーダーの文字列をアドレス指定するには、DM110.08L と入力します。
- DM200 で開始し、長さが15バイト、上位バイトのみの文字列をアドレス指定するには、DM200.15D と入力します。
- DM220 で開始し、長さが7バイト、下位バイトのみの文字列をアドレス指定するには、DM220.07E と入力します。

#### 配列のサポート

Boolean と String 以外のすべてのデータ型で配列がサポートされています。配列のアドレス指定には2つの方法があります。例ではデータメモリの位置が使用されています。

DMxxxx [行数] [列数]

DMxxxx [列数]\*

\*この方法では"行数"が1であるものと見なされます。

行数と列数を掛けた値にデータサイズのバイト数を掛けた値が116バイトを超えてはなりません。これはプロトコルによる制限です。このドライバーはASCIIプロトコルを使用するため、Word、Short、BCDは4バイト、DWord、Long、LBCD、Floatは8バイトあります。たとえば、4×7のWordの配列では配列サイズが28Word×4バイト=112バイトになり、これは最大要求サイズである116バイトに収まります。

◆32ビット値(DWord、Long、LBCD、Float)を修正する場合には注意してください。これらのデータ型を使用可能な各アドレスは、デバイス内で1Wordずつオフセットして開始しています。このため、DWord DM0とDM1はWord DM1で重複します。DM0に書き込んだ場合もDM1に格納されている値が修正されます。これらのデータ型を使用する際には重複が生じないようにすることをお勧めします。たとえばDWordを使用している場合、DM0、DM2、DM4などを使用すると、Wordの重複を避けることができます。

#### CQM1のアドレス指定

デフォルトのデータ型を太字で示しています。

◆詳細については、[文字列のサポート](#)および[配列のサポート](#)を参照してください。

デバイスタイプ	範囲	データ型	アクセス
補助リレー	AR00-AR27	<b>Word</b> 、Short、BCD、Long	読み取り 書き込み
	AR00-AR26	DWord、Float、LBCD	
	ARxx.00-ARxx.15	Boolean	
Stringの補助リレー、HiLoバイトオーダー	AR00.56H-AR27.02H ピリオドの後ろの長さは2から56文字の範囲の文字列長を示します	<b>String</b>	読み取り 書き込み
Stringの補助リレー、LoHiバイトオーダー	AR00.56L-AR27.02L ピリオドの後ろの長さは2から56文字の範囲の文字列長を示します	<b>String</b>	読み取り 書き込み
Stringの補助リレー、各Wordの上位バイトのみ使用	AR00.28D-AR27.01D ピリオドの後ろの長さは1から28文字の範囲の文字列長を示します	<b>String</b>	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
String の補助リレー、各 Word の下位バイトのみ使用	AR00.28E-AR27.01E ピリオドの後ろの長さは 1 から 28 文字の範囲の文字列長を示します	String	読み取り/書き込み
データメモリ	DM0000-DM6655 DM0000-DM6654 DMxxxx.00-DMxxxx.15	Word、Short、BCD、Long DWord、Float、LBCD Boolean	読み取り/書き込み
String のデータメモリ、HiLo バイトオーダー	DM0000.58H-DM6655.02H ピリオドの後ろの長さは 2 から 58 文字の範囲の文字列長を示します	String	読み取り/書き込み
String のデータメモリ、LoHi バイトオーダー	DM0000.58L-DM6655.02L ピリオドの後ろの長さは 2 から 58 文字の範囲の文字列長を示します	String	読み取り/書き込み
String のデータメモリ、各 Word の上位バイトのみ使用	DM0000.29D-DM6655.01D ピリオドの後ろの長さは 1 から 29 文字の範囲の文字列長を示します	String	読み取り/書き込み
String のデータメモリ、各 Word の下位バイトのみ使用	DM0000.29E-DM6655.01E ピリオドの後ろの長さは 1 から 29 文字の範囲の文字列長を示します	String	読み取り/書き込み
保持リレー	HR00-HR99 HR00-HR98 HRxx.00-HRxx.15	Word、Short、BCD、Long DWord、Float、LBCD Boolean	読み取り/書き込み
String の保持リレー、HiLo バイトオーダー	HR00.58H-HR99.02H ピリオドの後ろの長さは 2 から 58 文字の範囲の文字列長を示します	String	読み取り/書き込み
String の保持リレー、LoHi バイトオーダー	HR00.58L-HR99.02L ピリオドの後ろの長さは 2 から 58 文字の範囲の文字列長を示します	String	読み取り/書き込み
String の保持リレー、各 Word の上位バイトのみ使用	HR00.29D-HR99.01D ピリオドの後ろの長さは 1 から 29 文字の範囲の文字列長を示します	String	読み取り/書き込み
String の保持リレー、各 Word の下位バイトのみ使用	HR00.29E-HR99.01E ピリオドの後ろの長さは 1 から 29 文字の範囲の文字列長を示します	String	読み取り/書き込み
内部リレー	IR000-IR255 IR000-IR254 IRxxx.00-IRxxx.15	Word、Short、BCD、Long DWord、Float、LBCD	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	
String の内部リレー、HiLo バイトオーダー	IR000.58H-IR255.02H ピリオドの後ろの長さは2 から 58 文字の範囲の文字列長を示します	String	読み取り/ 書き込み
String の内部リレー、LoHi バイトオーダー	IR000.58L-IR255.02L ピリオドの後ろの長さは2 から 58 文字の範囲の文字列長を示します	String	読み取り/ 書き込み
String の内部リレー、各 Word の上位バイトのみ使用	IR000.29D-IR255.01D ピリオドの後ろの長さは1 から 29 文字の範囲の文字列長を示します	String	読み取り/ 書き込み
String の内部リレー、各 Word の下位バイトのみ使用	IR000.29E-IR255.01E ピリオドの後ろの長さは1 から 29 文字の範囲の文字列長を示します	String	読み取り/ 書き込み
リンクリレー	LR00-LR63 LR00-LR62 LRxx.00-LRxx.15	Word、Short、BCD、Long DWord、Float、LBCD Boolean	読み取り/ 書き込み
String のリンクリレー、HiLo バイトオーダー	LR00.58H-LR63.02H ピリオドの後ろの長さは2 から 58 文字の範囲の文字列長を示します	String	読み取り/ 書き込み
String のリンクリレー、LoHi バイトオーダー	LR00.58L-LR63.02L ピリオドの後ろの長さは2 から 58 文字の範囲の文字列長を示します	String	読み取り/ 書き込み
String のリンクリレー、各 Word の上位バイトのみ使用	LR00.29D-LR63.01D ピリオドの後ろの長さは1 から 29 文字の範囲の文字列長を示します	String	読み取り/ 書き込み
String のリンクリレー、各 Word の下位バイトのみ使用	LR00.29E-LR63.01E ピリオドの後ろの長さは1 から 29 文字の範囲の文字列長を示します	String	読み取り/ 書き込み
タイマー/カウンタ	RC000-RC511 RCxxx.00-RCxxx.15	Word、Short、BCD Boolean	読み取り/ 書き込み
タイマー/カウンタのステータス	TC000-TC511	Boolean	読み取り/ 書き込み

### 文字列のサポート

CQM1 モデルでは、補助リレー (AR)、データメモリ (DM)、保持リレー (HR)、内部リレー (IR)、リンクリレー (LR) を ASCII 文字列として読み書きできます。これらのタイプのデバイスを文字列データに使用している場合、各レジスタは2 バイト (2 文字) または1 バイトの ASCII データを格納できます。文字列を定義する際に、そのレジスタにおける ASCII データの順序、またはそのレジスタ内で使用するバイトを選択できます。

レジスタあたり2 バイトの ASCII データを使用する場合、文字列の長さは2 から 58 文字 (または AR では2 から 56) となり、ビット番号の位置に入力します。この長さは偶数として入力する必要があります。文字列によって指定されるレジス

タの範囲がデバイスタイプの範囲を超えてはなりません。バイトオーダーはアドレスの末尾に"H" または "L" を付けることによって指定します。

レジスタあたり1バイトのASCIIデータを使用する場合、文字列の長さは1から29文字(またはARでは1から28)となり、ビット番号の位置に入力します。文字列によって指定されるレジスタの範囲がデバイスタイプの範囲を超えてはなりません。レジスタ内で使用するバイトはアドレスに"D" または "E" を追加することによって指定します。

### 例

- DM100 で開始し、長さが50バイト、HiLo バイトオーダーの文字列をアドレス指定するには、DM100.50H と入力します。
- DM110 で開始し、長さが8バイト、LoHi バイトオーダーの文字列をアドレス指定するには、DM110.08L と入力します。
- DM200 で開始し、長さが15バイト、上位バイトのみの文字列をアドレス指定するには、DM200.15D と入力します。
- DM220 で開始し、長さが7バイト、下位バイトのみの文字列をアドレス指定するには、DM220.07E と入力します。

### 配列のサポート

Boolean と String 以外のすべてのデータ型で配列がサポートされています。配列のアドレス指定には2つの方法があります。例ではデータメモリの位置が使用されています。

DMxxxx [行数] [列数]

DMxxxx [列数]\*

\*この方法では"行数"が1であるものと見なされます。

行数と列数を掛けた値にデータサイズのバイト数を掛けた値が116バイトを超えてはなりません。これはプロトコルによる制限です。このドライバーはASCIIプロトコルを使用するため、Word、Short、BCDは4バイト、DWord、Long、LBCD、Floatは8バイトあります。たとえば、4 X 7のWordの配列では配列サイズが28 Word x 4 バイト = 112 バイトになり、これは最大要求サイズである116バイトに収まります。

●32ビット値(DWord、Long、LBCD、Float)を修正する場合には注意してください。これらのデータ型を使用可能な各アドレスは、デバイス内で1Wordずつオフセットして開始しています。このため、DWord DM0とDM1はWord DM1で重複します。DM0に書き込んだ場合もDM1に格納されている値が修正されます。これらのデータ型を使用する際には重複が生じないようにすることをお勧めします。たとえばDWordを使用している場合、DM0、DM2、DM4などを使用すると、Wordの重複を避けることができます。

### オープンアドレス指定

次のメモリマップは、新しいデバイスをサポートするためにすべてのタイプのメモリに公開されています。デバイス固有のアドレス範囲については、製造メーカーのドキュメントを参照してください。デフォルトのデータ型を太字で示しています。

●詳細については、[文字列のサポート](#)および[配列のサポート](#)を参照してください。

デバイスタイプ	範囲	データ型	アクセス
補助リレー	AR0000-AR9999 AR0000-AR9998 ARxxxx.00-ARxxxx.15	<b>Word</b> 、Short、BCD、Long、 DWord、Float、LBCD  Boolean	読み取り 書き込み
Stringの補助リレー HiLo バイトオーダー	AR0000.58H-AR9999.02H  ピリオドの後ろの長さは2から58文字の範囲の文字列長を示します	<b>String</b>	読み取り

デバイスタイプ	範囲	データ型	アクセス
			書き込み
String の補助リレー LoHi バイトオーダー	AR0000.58L-AR9999.02L ピリオドの後ろの長さは2 から 58 文字の範囲の文字列長を示します	String	読み取り書き込み
String の補助リレー 各 Word の上位バイトのみ使用 xxxxx	AR0000.29D-AR9999.01D ピリオドの後ろの長さは1 から 29 文字の範囲の文字列長を示します	String	読み取り書き込み
String の補助リレー 各 Word の下位バイトのみ使用 xxxxx	AR0000.29E-AR9999.01E ピリオドの後ろの長さは1 から 29 文字の範囲の文字列長を示します	String	読み取り書き込み
データメモリ	DM0000-DM9999 DM0000-DM9998 DMxxxx.00-DMxxxx.15	Word、Short、BCD、Long、DWord、Float、LBCD  Boolean	読み取り書き込み
String のデータメモリ HiLo バイトオーダー	DM0000.58H-DM9999.02H ピリオドの後ろの長さは2 から 58 文字の範囲の文字列長を示します	String	読み取り書き込み
String のデータメモリ LoHi バイトオーダー	DM0000.58L-DM9999.02L ピリオドの後ろの長さは2 から 58 文字の範囲の文字列長を示します	String	読み取り書き込み
String のデータメモリ 各 Word の上位バイトのみ使用	DM0000.29D-DM9999.01D	String	読

デバイスタイプ	範囲	データ型	アクセス
xxxxx	ピリオドの後ろの長さは1から29文字の範囲の文字列長を示します		み取り書き込み
String のデータメモリ 各 Word の下位バイトのみ使用 xxxxx	DM0000.29E-DM9999.01E ピリオドの後ろの長さは1から29文字の範囲の文字列長を示します	String	読み取り書き込み
保持リレー	HR0000-HR9999 HR0000-HR9998 HRxxxx.00-HRxxxx.15	Word、Short、BCD、Long、DWord、Float、LBCD  Boolean	読み取り書き込み
String の保持リレー HiLo バイトオーダー	HR0000.58H-HR9999.02H ピリオドの後ろの長さは2から58文字の範囲の文字列長を示します	String	読み取り書き込み
String の保持リレー LoHi バイトオーダー	HR0000.58L-HR9999.02L ピリオドの後ろの長さは2から58文字の範囲の文字列長を示します	String	読み取り書き込み
String の保持リレー 各 Word の上位バイトのみ使用 xxxxx	HR0000.29D-HR9999.01D ピリオドの後ろの長さは1から29文字の範囲の文字列長を示します	String	読み取り書き込み
String の保持リレー 各 Word の下位バイトのみ使用 xxxxx	HR0000.29E-HR9999.01E ピリオドの後ろの長さは1から29文字の範囲の文字列長を示します	String	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
			み
内部リレー	IR0000-IR9999 IR0000-IR9998 IRxxxx.00-IRxxxx.15	Word、Short、BCD、Long、DWord、Float、LBCD  Boolean	読み取り書き込み
String の内部リレー HiLo バイトオーダー	IR0000.58H-IR9999.02H ピリオドの後ろの長さは2から58文字の範囲の文字列長を示します	String	読み取り書き込み
String の内部リレー LoHi バイトオーダー	IR0000.58L-IR9999.02L ピリオドの後ろの長さは2から58文字の範囲の文字列長を示します	String	読み取り書き込み
String の内部リレー 各 Word の上位バイトのみ使用	IR0000.29D-IR9999.01D ピリオドの後ろの長さは1から29文字の範囲の文字列長を示します	String	読み取り書き込み
String の内部リレー 各 Word の下位バイトのみ使用 xxxxx	IR0000.29E-IR9999.01E ピリオドの後ろの長さは1から29文字の範囲の文字列長を示します	String	読み取り書き込み
リンクリレー	LR0000-LR9999 LR0000-LR9998 LRxxxx.00-LRxxxx.15	Word、Short、BCD、Long、DWord、Float、LBCD  Boolean	読み取り書き込み
String のリンクリレー HiLo バイトオーダー	LR0000.58H-LR9999.02H ピリオドの後ろの長さは2から58文字の範囲の文字列長を示します	String	読み取り書き

デバイスタイプ	範囲	データ型	アクセス
			き込み
String のリンクリレー LoHi バイトオーダー	LR0000.58L-LR9999.02L  ピリオドの後ろの長さは2 から 58 文字の範囲の文字列長を示します	String	読み取り書き込み
String のリンクリレー 各 Word の上位バイトのみ使用 xxxxx	LR0000.29D-LR9999.01D  ピリオドの後ろの長さは1 から 29 文字の範囲の文字列長を示します	String	読み取り書き込み
String のリンクリレー 各 Word の下位バイトのみ使用 xxxxx	LR0000.29E-LR9999.01E  ピリオドの後ろの長さは1 から 29 文字の範囲の文字列長を示します	String	読み取り書き込み
タイマー/カウンタ	RC0000-RC9999  RCxxxx.00-RCxxxx.15	Word、Short、BCD  Boolean	読み取り書き込み
タイマー/カウンタのステータス*	TC0000-TC9999	Boolean	読み取り書き込み

\*注記: 一部のモデルはタイマー/カウンタのステータス (TCxxxx) への書き込みをサポートしていません。

### 文字列のサポート

オープンモデルでは、補助リレー (AR)、データメモリ (DM)、保持リレー (HR)、内部リレー (IR)、リンクリレー (LR) を ASCII 文字列として読み書きできます。これらのタイプのデバイスを文字列データに使用している場合、各レジスタは 2 バイト (2 文字) または 1 バイトの ASCII データを格納できます。文字列を定義する際に、そのレジスタにおける ASCII データの順序、またはそのレジスタ内で使用するバイトを選択できます。

レジスタあたり2バイトのASCIIデータを使用する場合、文字列の長さは2から58文字となり、ビット番号の位置に入力します。この長さは偶数として入力する必要があります。文字列によって指定されるレジスタの範囲がデバイスタイプの範囲を超えてはなりません。バイトオーダーはアドレスの末尾に"H"または"L"を付けることによって指定します。

レジスタあたり1バイトのASCIIデータを使用する場合、文字列の長さは1から29文字となり、ビット番号の位置に入力します。文字列によって指定されるレジスタの範囲がデバイスタイプの範囲を超えてはなりません。レジスタ内で使用するバイトはアドレスに"D"または"E"を追加することによって指定します。

#### 例

- DM100 で開始し、長さが50バイト、HiLoバイトオーダーの文字列をアドレス指定するには、DM100.50Hと入力します。
- DM110 で開始し、長さが8バイト、LoHiバイトオーダーの文字列をアドレス指定するには、DM110.08Lと入力します。
- DM200 で開始し、長さが15バイト、上位バイトのみの文字列をアドレス指定するには、DM200.15Dと入力します。
- DM220 で開始し、長さが7バイト、下位バイトのみの文字列をアドレス指定するには、DM220.07Eと入力します。

#### 配列のサポート

BooleanとString以外のすべてのデータ型で配列がサポートされています。配列のアドレス指定には2つの方法があります。例ではデータメモリの位置が使用されています。

DMxxxx [行数] [列数]  
DMxxxx [列数]\*

\*この方法では"行数"が1であるものと見なされます。

行数と列数を掛けた値にデータサイズのバイト数を掛けた値が116バイトを超えてはなりません。これはプロトコルによる制限です。このドライバーはASCIIプロトコルを使用するため、Word、Short、BCDは4バイト、DWord、Long、LBCD、Floatは8バイトあります。たとえば、4 X 7のWordの配列では配列サイズが28 Word x 4バイト = 112バイトになり、これは最大要求サイズである116バイトに収まります。

●32ビット値(DWord、Long、LBCD、Float)を修正する場合には注意してください。これらのデータ型を使用可能な各アドレスは、デバイス内で1Wordずつオフセットして開始しています。このため、DWord DM0とDM1はWord DM1で重複します。DM0に書き込んだ場合もDM1に格納されている値が修正されます。これらのデータ型を使用する際には重複が生じないようにすることをお勧めします。たとえばDWordを使用している場合、DM0、DM2、DM4などを使用すると、Wordの重複を避けることができます。

## イベント ログメッセージ

次の情報は、メインユーザ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リングとソートについては、OPC 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。

**要求された型と受信した型が一致しません。| タグアドレス = '<アドレス>'、要求された型 = <タイプ>、受信した型 = <タイプ>。**

---

**エラータイプ:**

エラー

**ブロックに不良アドレスがあります。| タグアドレス = '<アドレス>'、ブロック = <start> ~ <end>。**

---

**エラータイプ:**

警告

**考えられる原因:**

指定されたデバイスに存在しない位置を参照しようとした。

**解決策:**

タグに割り当てられたアドレスがデバイスの指定された範囲内にあることを確認します。無効な位置を参照するタグを削除または修正してください。

### エラーマスクの定義

---

B = ハードウェアの破損が検出されました

F = フレーミングエラー

E = I/O エラー

O = 文字バッファオーバーラン

R = RX バッファオーバーラン

P = 受信バイトパリティエラー

T = TX バッファフル



# 索引

## B

BCD 16  
Boolean 16

## C

C200H のアドレス指定 20  
C20H 17  
COM ID 6  
COM ポート 6  
CQM1 のアドレス指定 23

## D

DTR 6  
DWord 16

## E

ERP 3

## F

Float 16

## H

HMI 3

## I

I/O エラー 32  
ID 10

## L

LBCD 16

Long 16

## M

MES 3

## R

RS-485 7

RTS 7

RX バッファオーバーラン 32

## S

SCADA 3

Short 16

String 16

SYSMAC C シリーズ 3

## T

TX バッファフル 32

## W

Word 16

## あ

アイドル接続を閉じる 7-8

アドレスの説明 17

## い

イーサネットカプセル化 4, 6

イーサネット設定 7

イベントログメッセージ 32

## え

エラーマスクの定義 32

エラー時に格下げ 13

## お

オーバーラン 32

オープンアドレス指定 26

## き

キャッシュからの初回更新 11

## く

クローズするまでのアイドル時間 7-8

## し

シミュレーション 10

シリアルポートの設定 6

シリアル通信 6

## す

スキャンしない、要求ポールのみ 11

スキャンモード 11

ストップビット 4, 6

すべてのタグのすべての値を書き込み 8

すべてのタグの最新の値のみを書き込み 8

## せ

ゼロで置換 9

## た

タイミング 12

タイムアウト前の試行回数 12

タグに指定のスキャン速度を適用 11

タグ数 5

## ち

チャンネルのプロパティ-シリアル通信 6

チャンネルのプロパティ-一般 5

チャンネルのプロパティ-書き込み最適化 8

チャンネルのプロパティ-詳細 9

チャンネル割り当て 10

## て

データコレクション 10

データビット 4, 6

データ型の説明 16

デバイスのプロパティ-タイミング 12

デバイスのプロパティ-自動格下げ 13

デバイスのプロパティ-冗長 15

デバイス間遅延 9

デューティサイクル 8

## と

ドライバー 10

## な

なし 6

## ね

ネットワーク 4

ネットワークアダプタ 7

## は

ハードウェアの破損 32

パリティ 4, 6, 32

## ひ

ヒストリアン 3

## ふ

フレーミング 32

フロー制御 4, 6

ブロックに不良アドレスがあります。| タグアドレス = '<アドレス>', ブロック = <start> ~ <end>。 32

## ほ

ポーリング遅延 7

ポーレート 4, 6

## ま

マスク 32

## も

モデム 6-7

モデム設定 7

モデル 10

## 漢字

概要 3

格下げまでのタイムアウト回数 13

格下げ期間 13

格下げ時に要求を破棄 13

共有 6

最適化方法 8

事前オン 7

自動ダイヤル 7

自動格下げ 13

識別 5, 9

実行動作 7

冗長 15

診断 5

接続タイプ 6

接続のタイムアウト 7, 12

- 設定 4
- 遅延 14
- 遅延オフ 7
- 通信エラーを報告 7
- 通信タイムアウト 12
- 通信なしの動作 8
- 通信プロトコル 4
- 読み取り処理 8
- 非 Boolean タグの最新の値のみを書き込み 8
- 非正規化浮動小数点処理 9
- 物理メディア 6
- 文字間遅延 13
- 未修正 9
- 要求された型と受信した型が一致しません。| タグアドレス = '<アドレス>'、要求された型 = <タイプ>、受信した型 = <タイプ>。 32
- 要求のタイムアウト 12