Modbus RTU シリアルドライバー

© 2025 PTC Inc. All Rights Reserved.

目次

Modbus RTU シリアルドライバー 1
目次 2
Modbus RTU シリアルドライバー ヘルプセンターへようこそ 5
概要 5
設定 6
チャネルのプロパティ - 一般 6
タグ数 7
チャネルのプロパティ - 通信シリアル化 7
チャネルのプロパティ - 書き込み最適化 9
チャネルのプロパティ - シリアル通信 9
チャネルのプロパティ - 詳細 12
デバイスのプロパティ - 一般 12
動作モード 13
タグ数 14
デバイスのプロパティ - スキャンモード 14
デバイスのプロパティ - イーサネットカプセル化 15
デバイスのプロパティ - タイミング 16
デバイスのプロパティ - 自動格下げ 17
デバイスのプロパティ - タグ生成 17
デバイスのプロパティ - 設定 19
デバイスのプロパティ - ブロックサイズ 23
デバイスのプロパティ - 変数のインポート設定 25
デバイスのプロパティ - フレーミング 25
デバイスのプロパティ - エラー処理 26
デバイスのプロパティ - 冗長 27
自動タグデータベース生成 28
統計アイテム 29
データ型の説明 32
アドレスの説明 33
Modbus のアドレス指定 33
Magnetek GPD 515 ドライブのアドレス指定 36
Elliott フローコンピュータのアドレス指定 37
Daniels S500 フローコンピュータのアドレス指定 38
ダイナミックフルードメーターのアドレス指定 38
Omni フローコンピュータのアドレス指定 40
Omni カスタムパケット 45

Omni 生データアーカイブ 48
Omni テキストレポート 53
Omni テキストアーカイブ 55
ファンクションコードの説明 57
Configuration API Service - チャネルのプロパティ 58
Configuration API Services - デバイスのプロパティ 58
イベントログメッセージ 60
ブロック範囲に不良アドレスがあります。| アドレス範囲 = <開始> ～ <終了>。 60
不良配列。| 配列範囲 = <開始> ～ <終了>。 60
ブロックアドレスが例外コードを返しました。| アドレス範囲 = <開始> ～ <終了>、例外コード = <コード>。 61
アドレスに書き込めません。デバイスは例外コードを返しました。| アドレス = '<アドレス>'、例外コード = <コード>。 61
アドレスから読み取れません。デバイスは例外コードを返しました。| アドレス = '<アドレス>'、例外コード = <コード>。 61
メモリリソース量の低下によりタグインポートが失敗しました。 61
タグのインポート中にファイル例外が発生しました。 62
インポートファイルのレコードの解析でエラーが発生しました。| レコード番号 = <数値>、フィールド = <名前>。 62
インポートファイルのレコードの説明が切り詰められました。| レコード番号 = <数値>。 62
インポートされたタグ名が無効のため変更されました。| タグ名 = '<タグ>'、変更後のタグ名 = '<タグ>'。 62
データ型がサポートされていないため、タグをインポートできませんでした。| タグ名 = '<タグ>'、サポートされていないデータ型 = '<タイプ>'。 63
メモリ割り当ての問題により、Omni テキストバッファを読み取れませんでした。 63
指定された日付範囲では使用可能な Omni テキストアーカイブデータがありません。 63
Omni テキストレポートへの書き込みが切り詰められました。| レポート番号 = <数値>。 63
パケット数の制限により、Omni テキストレポートを読み取れませんでした。| レポート番号 = <数値>。 64
書き込みに失敗しました。パスの最大長を超えています。| タグアドレス = '<アドレス>'、最大長 = <数値>。 64
Omni テキストデータをファイルに書き込んでいるときにエラーが発生しました。| タグアドレス = '<アドレス>'、理由 = '<理由>'。 64
Omni テキスト出力ファイルを開くことができませんでした。| タグアドレス = '<アドレス>'、理由 = '<理由>'。 64
アドレスに書き込めません。予期しない文字が応答に含まれています。| タグアドレス = '<アドレス>'。 65
アドレスから読み取れません。予期しない文字が応答に含まれています。| タグアドレス = '<アドレス>'。 65
ブロックのアドレスを読み取れません。予期しない文字が応答に含まれています。| アドレス範囲 = <開始> ～ <終了>。 65
Omni テキスト出力ファイルを変更できませんでした。| タグアドレス = '<アドレス>', 理由 = 指定したパスは許可されません。 65
Omni テキスト出力ファイルを変更できませんでした。| タグアドレス = '<アドレス>', 理由 = 指定したファイルの拡張子は '.txt' または '.log' でなければなりません。 65
ファイルからタグデータベースをインポートしています。| ファイル名 = '<名前>'。 66
エラーマスクの定義 66

Modbus 例外コード 67
索引 **69**

Modbus RTU シリアルドライバー ヘルプセンターへようこそ

このヘルプセンターは、Kepware Modbus RTU シリアルドライバー のユーザードキュメンテーションです。このヘルプセンターは、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
Modbus RTU Serialドライバーとは

設定
このドライバーとともに使用するためにチャネルとデバイスを設定する方法

自動タグデータベース生成
Modbus RTU Serialドライバー用にタグを設定する方法

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
Modbus 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
Modbus RTU Serialドライバーで生成されるメッセージ

Additional Resources

- Search the **Technical Support knowledge base** to find known answers to technical questions and issues (login required).
- For links to Help Centers for all PTC products, including Help Centers for other releases of this product, see the **PTC Help Centers**.
- For a collaborative approach to development and troubleshooting, post questions or solutions in the **Community forum**.

We Want Your Feedback

We are interested in your feedback on the content of this Help Center.

- Click in the upper right corner of any page to send feedback about that page.
- Email **documentation@ptc.com** to provide suggestions or comments about the Help Center.

バージョン 1.098
© 2025 PTC Inc. All Rights Reserved.

概要

Modbus RTU シリアルドライバー は Modbus シリアルデバイスが 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 Modbus RTU プロトコルをサポートするシリアルデバイスで使用するためのものです。Modbus RTU シリアルドライバー は Modbus RTU と互換性のある広範なデバイスをサポートするために開発されました。

設定

通信プロトコル

Modbus RTU プロトコル

サポートされる通信パラメータ

「ボーレート」: 1200、2400、9600、19200
「パリティ」: 奇数、偶数、なし
データビット: 8
「ストップビット」: 1 および 2

サポートされるデバイス

- Modbus 対応デバイス
- Elliott フローコンピュータ
- Magnetek GPD 515 ドライブ
- Omni フローコンピュータ
- Daniels S500 フローコンピュータ
- ダイナミックフルードメーター (DFM) SFC3
- TSXCUSBMBP USB アダプタ

チャ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ネルの最大数は 2048 です。
- **シリアル接続**を使用する場合、1 チャネルあたりの推奨最大デバイス数は 32 です (ハードウェア信号増幅デバイスを使用する場合を除く)。
- **イーサネットカプセル化**を使用する場合、1 チャネルあたりの最大デバイス数は 255 になります。

チャネルのプロパティ - 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ネルから成ります。チャネルは、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ネルプロパティを指定するときに使用します。

識別

「名前」: このチャ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。チャネル名は OPC ブラウザ情報の一部です。チャ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」: このチャネル用のプロトコル/ドライバーを指定します。チャ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ネル名を変更することも含まれます。チャ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。
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。

チャネルのプロパティ - 通信シリアル化

サーバーのマルチスレッドアーキテクチャにより、チャネルはデバイスとの並列通信が可能になります。これは効率的ですが、物理ネットワークに制約がある (無線イーサネットなど) 場合には通信をシリアル化できます。通信シリアル化によって、仮想ネットワーク内で同時に通信可能なチャネルは 1 つに制限されます。

"仮想ネットワーク" という用語は、通信に同じパイプラインを使用するチャネルと関連デバイスの集合を表します。たとえば、無線イーサネットのパイプラインはクライアント無線です。同じクライアント無線を使用しているチャネルは、すべて同じ仮想ネットワークに関連付けられています。チャネルは "ラウンドロビン" 方式で 1 つずつ順番に通信できます。デフォルトでは、チャネルが 1 つのトランザクションを処理した後で、通信を別のチャネルに渡します。トランザクションには 1 つ以上のタグが含まれることがあります。要求に応答しないデバイスが制御チャネルに含まれている場合、そのトランザクションがタイムアウトになるまでチャネルは制御を解放できません。これによって、仮想ネットワーク内のその他のチャネルでデータ更新の遅延が生じます。

プロパティグループ		
一般	⊟ チャネルレベルの設定	
シリアル通信	仮想ネットワーク	なし
書き込み最適化	サイクルあたりのトランザクション数	1
詳細	⊟ グローバル設定	
通信シリアル化	ネットワークモード	負荷分散

チャネルレベルの設定

「仮想ネットワーク」: 通信シリアル化のチャネルのモードを指定します。オプションには「なし」、「ネットワーク 1」 - 「ネットワーク 500」があります。デフォルトは「なし」です。オプションの説明は次のとおりです。

- **「なし」**: このオプションを選択した場合、チャネルの通信シリアル化は無効になります。
- **「ネットワーク 1」 - 「ネットワーク 500」**: このオプションでは、チャネルを割り当てる仮想ネットワークを指定します。

「サイクルあたりのトランザクション数」: チャネルで実行可能な単一ブロック/非ブロック読み取り/書き込みトランザクションの数を指定します。あるチャネルが通信する機会を得ると、この数だけトランザクションが試みられます。有効な範囲は 1 から 99 です。デフォルトは 1 です。

グローバル設定

「ネットワークモード」: このプロパティでは、チャネル通信を委譲する方法を制御します。**「負荷分散」**モードでは、各チャネルが 1 つずつ順番に通信する機会を得ます。**「優先順位」**モードでは、チャネルは次の規則 (最も高い優先順位から最も低い優先順位の順) に従って通信する機会を得ます。

1. 書き込みが保留中になっているチャネルの優先順位が最も高くなります。
2. (内部のプラグインまたは外部のクライアントインタフェースによって) 明示的な読み取りが保留中になっているチャネルは、その読み取りの優先順位に基づいて優先順位が決まります。
3. スキャン読み取りおよびその他の定期的イベント (ドライバー固有)。

デフォルトは「負荷分散」であり、すべての仮想ネットワークとチャネルに影響します。
非送信請求応答に依存するデバイスを仮想ネットワーク内に配置してはなりません。通信をシリアル化する必要がある場合、「自動格下げ」を有効にすることをお勧めします。

データを読み書きする方法はドライバーによって異なるので (単一ブロック/非ブロックトランザクションなど)、アプリケーションの「サイクルあたりのトランザクション数」プロパティを調整する必要があります。その場合、次の要因について検討します。

- 各チャネルから読み取る必要があるタグの数
- 各チャネルにデータを書き込む頻度
- チャネルが使用しているのはシリアルドライバーかイーサネットドライバーか?
- ドライバーは複数の要求に分けてタグを読み取るか、複数のタグをまとめて読み取るか?
- デバイスのタイミングプロパティ (「要求のタイムアウト」や「連続した *x* 回のタイムアウト後の失敗」など) が仮想ネットワークの通信メディアに最適化されているか?

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
一般
シリアル通信
書き込み最適化

書き込み最適化	
最適化方法	すべてのタグの最新の値のみを書き込み
デューティサイクル	10

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。モーメンタリプ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。

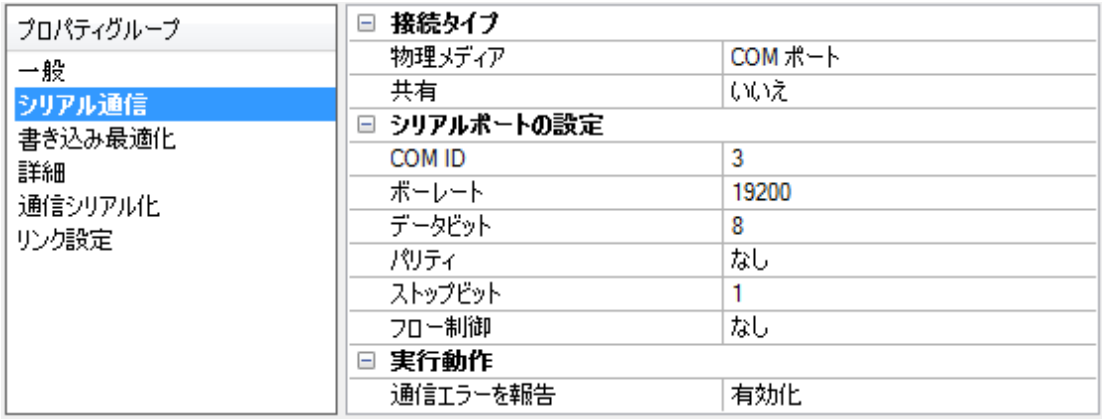
「デューティサイクル」: 読み取り操作に対する書き込み操作の比率を制御するときに使用します。この比率は必ず、読み取り 1 回につき書き込みが 1 から 10 回の間であることが基になっています。デューティサイクルはデフォルトで 10 に設定されており、1 回の読み取り操作につき 10 回の書き込みが行われます。アプリケーションが多数の連続書き込みを行っている場合でも、読み取りデータを処理する時間が確実に残っている必要があります。これを設定すると、書き込み操作が 1 回行われるたびに読み取り操作が 1 回行われるようになります。実行する書き込み操作がない場合、読み取りが連続処理されます。これにより、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。
- 使用する特定の通信パラメータを定義する必要があります。ドライバーによっては、チャネルが同一の通信パラメータを共有できる場合とできない場合があります。仮想ネットワークに設定できる共有シリアル接続は 1 つだけです (「チャネルのプロパティ - シリアル通信」を参照してください)。



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」、「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. **「共有」**: 現在の構成を別のチャ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」**: データ転送の前にRTS 回線を事前にオン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。
    - **「遅延オフ」**: データ転送後にRTS 回線を解放するまでの時間を指定します。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」**: 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。

モデム設定

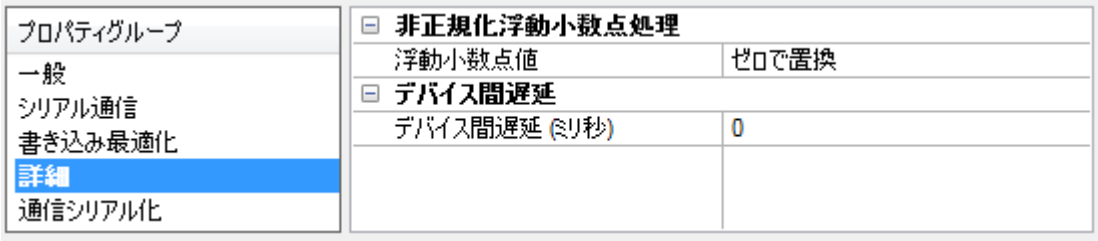
- **「モデム」**: 通信に使用するインストール済みモデムを指定します。
- **「接続タイムアウト」**: 接続が確立される際に待機する時間を指定します。この時間を超えると読み取りまたは書き込みが失敗します。デフォルトは60 秒です。
- **「モデムのプロパティ」**: モデムハードウェアを設定します。クリックした場合、ベンダー固有のモデムプロパティが開きます。
- **「自動ダイヤル」**: 電話帳内のエントリに自動ダイヤルできます。デフォルトは「無効化」です。詳細については、サーバーのヘルプで「モデム自動ダイヤル」を参照してください。
- **「通信エラーを報告」**: 低レベル通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 - 詳細

このグループは、チャ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。



「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」: 通信チャネルが同じチャ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。

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、複数のチャネルで使用できます。

注記: わかりやすい名前に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、チャネル名とデバイス名の組み合わせが "<チャネル名>.<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。

「チャネル割り当て」: このデバイスが現在属しているチャ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」: この ID に関連付けられてい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、ドライバーに関するマニュアルを参照してください。

「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。

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。(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. クライアントが切断して再接続するまで、更新は適用されません。
2. 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。
3. 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。
4. デバイスをシミュレートしたときに、クライアントで更新が 1 秒未満で表示されない場合があります。

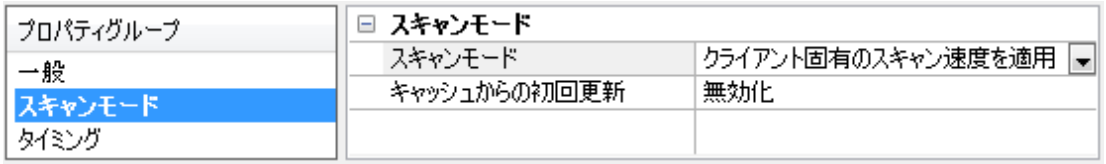
シミュレーションモードはテストとシミュレーションのみを目的としています。本番環境では決して使用しないでください。

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。

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。



「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。

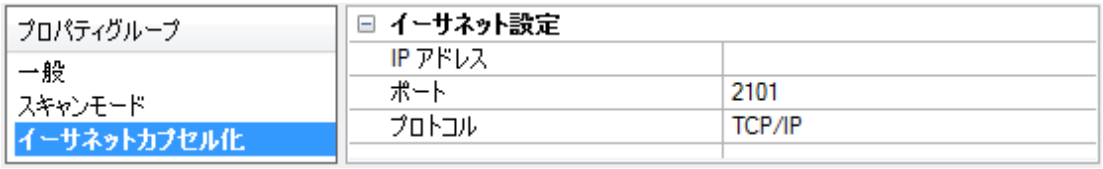
「キャッシュからの初期更新」: このオプションを有効にした場合、サーバーは保存 (キャッシュ) されているデータから、新たにアクティブ化されたタグ参照の初回更新を行います。キャッシュからの更新は、新しいアイテム参照が同じアドレス、スキャン速度、データ型、クライアントアクセス、スケール設定のプロパティを共有している場合にのみ実行できます。1 つ目のクライアント参照についてのみ、初期更新にデバイス読み取りが使用されます。デフォルトでは無効になっており、クライアントがタグ参照をアクティブ化したときにはいつでも、サーバーがデバイスから初期値の読み取りを試みます。

デバイスのプロパティ - イーサネットカプセル化

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用に設計されています。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す。ターミナルサーバーは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。メッセージがシリアル形式に変換されると、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。

詳細については、サーバーのヘルプで「イーサネットカプセル化の使用方法」を参照してください。

イーサネットカプセル化はドライバーに対して透過的なので、残りのプロパティを、これらがあたかもローカルシリアルポート上で直接デバイスに接続しているかのように設定します。



「IP アドレス」: このデバイスが接続されているターミナルサーバーの 4 つのフィールドから構成される IP アドレスを入力します。IP は YYY.YYY.YYY.YYY として指定します。YYY は IP アドレスを示します。各 YYY バイトが 0 から 255 の範囲でなければなりません。各シリアルデバイスは独自の IP アドレスを持つことができますが、単一のターミナルサーバーからマルチドロップされた複数のデバイスがある場合、複数のデバイスが同じ IP アドレスを持つことがあります。

「ポート」: リモートターミナルサーバーに接続するためのイーサネットポートを設定します。

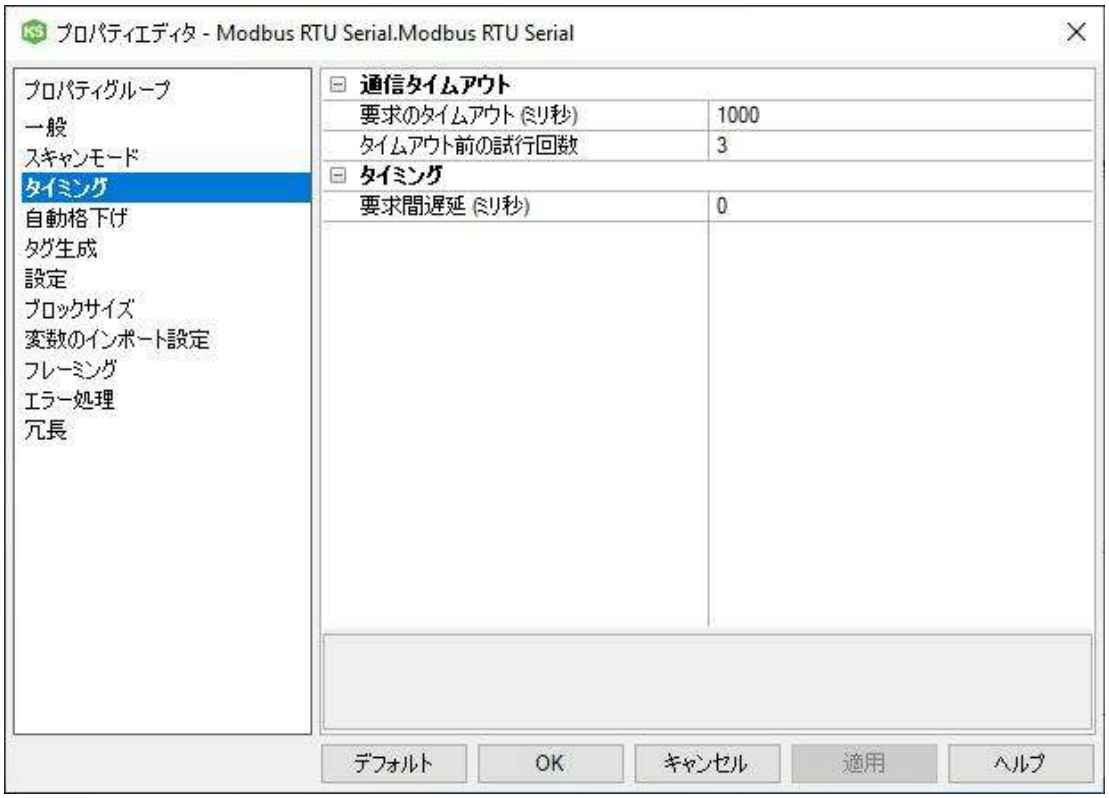
「プロトコル」: TCP/IP 通信または UDP 通信を設定します。この選択は使用されているターミナルサーバーの特性によります。デフォルトのプロトコル選択は TCP/IP です。使用可能なプロトコルの詳細については、ターミナルサーバーのヘルプドキュメントを参照してください。

注記

1. 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。
2. IP アドレスの有効な範囲は 0.0.0.0 から 255.255.255.255 です (0.0.0.0 と 255.255.255.255 は含まれません)。

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。



通信タイムアウト

「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
一般
スキャンモード
タイミング
自動格下げ

自動格下げ	
エラー時に格下げ	有効化
格下げまでのタイムアウト回数	3
格下げ期間 (ミリ秒)	10000
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 - 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
一般
スキャンモード
タイミング
自動格下げ
タグ生成

タグ生成	
デバイス起動時	起動時に生成しない
重複タグ	作成時に削除
親グループ	
自動生成されたサブグループを許可	有効化

「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、サーバーのタグスペース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除去します。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。

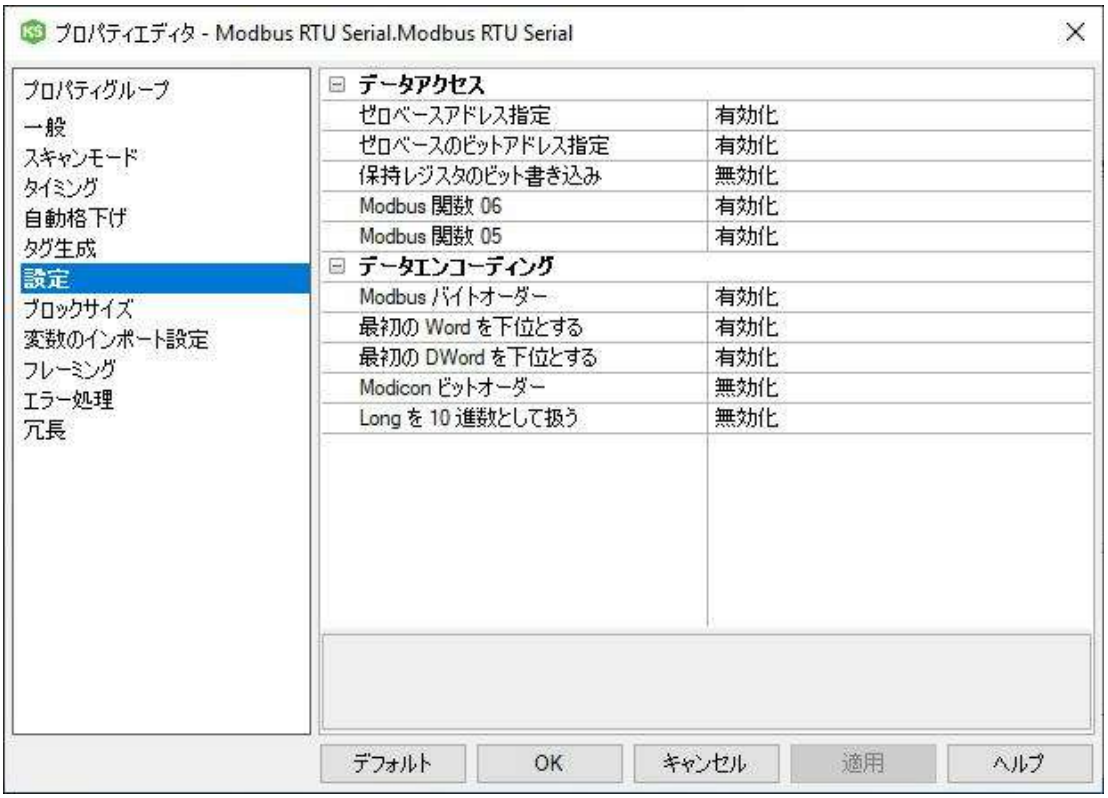
「自動生成されたサブグループを許可」: このプロパティでは、自動生成されたタグ用のサブグループをサーバーが自動的に作成するかどうかを制御します。これはデフォルトの設定です。無効になっている場合、サーバーはグループを作成しないで、デバイスのタグをフラットリスト内に生成します。サーバープロジェクトで、生成されたタグには名前としてアドレスの値が付きます。たとえば、生成プロセス中はタグ名は維持されません。

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。

デバイスのプロパティ - 設定



データアクセス

「ゼロベースアドレス指定」: デバイスのアドレス番号付けの規則で番号がゼロではなく 1 で開始する場合、デバイスのパラメータを定義する際にこの値を指定できます。デフォルトでは、Modbus デバイスと通信するためにフレームを構築する場合はユーザーが入力したアドレスから 1 が引かれます。デバイスがこの規則に従わない場合、「無効」を選択します。デフォルトの動作は Modicon PLC の規則に従います。

「ゼロベースのビットアドレス指定」: レジスタ内で、Word 内のビットをブールとして参照可能なメモリタイプ。アドレス指定の表記は <アドレス>.<ビット> であり、ここで <ビット> は Word 内のビット番号を表します。このオプションによって、ある Word 内の 1 ビットを 2 つの方法 (ゼロベースまたは 1 ベース) によってアドレス指定できます。ゼロベースとは最初のビットが 0 で始まることを意味し (範囲 = 0-15)、1 ベースとは最初のビットが 1 で始まることを意味します (範囲 = 1-16)。

「保持レジスタのビット書き込み」: 保持レジスタ内のビット位置に書き込む際、ドライバーは対象のビットのみを修正する必要があります。一部のデバイスはレジスタ内の 1 ビットを操作する特別なコマンドをサポートしています (ファンクションコード 0x16 (16 進) または 22 (10 進))。デバイスがこの機能をサポートしていない場合、ドライバーは 1 ビットだけが変更されるように読み取り/修正/書き込み操作を実行する必要があります。有効になっている場合、この単一レジスタ書き込みの設定に関係なく、レジスタへの書き込みにファンクションコード 0x16 を使用します。無効になっている場合、ドライバーは単一レジスタ書き込みの「Modbus 関数 06」の選択に応じて、ファンクションコード 0x06 または 0x10 を使用します。デフォルトでは無効に設定されています。
注記: Modbus バイトオーダーが無効になっている場合、このコマンドで送信されるマスクのバイトオーダーは Intel バイトオーダーになります。

「Modbus 関数 06」: このドライバーは、保持レジスタのデータをターゲットデバイスに書き込む Modbus プロトコルファンクションをサポートしています。ほとんどの場合、このドライバーは書き込み対象のレジスタの数に基づいてファンクション 06 と 16 を切り替えます。単一の 16 ビットレジスタに書き込む場合、このドライバーは通常、Modbus ファンクション 06 を使用します。32 ビット値を 2 つのレジスタに書き込む場合、このドライバーは Modbus ファンクション 16 を使用します。標準の Modicon PLC では、このどちらのファンクションを使用しても問題ありません。ただし、Modbus プロトコルを使用する多くのサードパーティデバイスとこれらのデバイスの多くは、保持レジスタへの書き込みに Modbus ファンクション 16 のみをサポートしています。この選択はデフォルトで有効になっており、ドライバーは必要に応じて 06 と 16 を切り替えることができます。デバイスが Modbus ファンクション 16 のみを使用してすべての書き込みを行う必要がある場合、この選択を無効にします。
注記: Word 内のビットの書き込みでは、「保持レジスタのビット書き込み」プロパティがこのオプションよりも優先されます。「保持レジスタのビット書き込み」が有効になっている場合、このプロパティにかかわらず、ファンクションコード 0x16 が使用されます。無効になっている場合、Word 内のビットの書き込みにはファンクションコード 0x06 または 0x10 が使用されます。

「Modbus 関数 05」: このドライバーは、出力コイルのデータをターゲットデバイスに書き込む Modbus プロトコルファンクションをサポートしています。ほとんどの場合、このドライバーは書き込み対象のコイルの数に基づいてこの 2 つのファンクションを切り替えます。単一のコイルに書き込む場合、このドライバーは Modbus ファンクション 05 を使用します。コイルの配列に書き込む場合、このドライバーは Modbus ファンクション 15 を使用します。標準の Modicon PLC では、このどちらのファンクションを使用しても問題ありません。ただし、Modbus プロトコルを使用する多くのサードパーティデバイスとこれらのデバイスの多くは、コイルの数に関係なく、出力コイルへの書き込みに Modbus ファンクション 15 の使用のみをサポートしています。この選択はデフォルトで有効になっており、ドライバーは必要に応じて 05 と 15 を切り替えることができます。デバイスが Modbus ファンクション 15 のみを使用してすべての書き込みを行う必要がある場合、この選択を無効にします。

データエンコーディング

「Modbus バイトオーダー」: 各レジスタ/16 ビット値のデータエンコーディングを設定します。この選択を使用することで、バイトオーダーをデフォルトの Modbus バイトオーダーから Intel バイトオーダーに変更できます。デフォルトでは有効になっており、Modbus 対応デバイスでは標準の設定です。デバイスが Intel バイトオーダーを使用する場合、このプロパティを無効にすることで Intel フォーマットのデータを読み取ります。

注記: この設定は Omni モデルには適用されません。このモデルでは必ず Modbus バイトオーダーが使用されます。

「最初の Word を下位とする」: 32 ビット値と、64 ビット値の DWord のデータエンコーディングを設定します。Modbus 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。ドライバーはこのオプションに基づいて、最初の Word を 32 ビット値の下位 Word または上位 Word として読み取ることができます。デフォルトは「有効」で最初の Word が下位となり、Modicon Modsoft プログラミングソフトウェアの規則に従います。

注記: この設定は Omni モデルには適用されません。このモデルでは必ず Modbus バイトオーダーが使用されます。

「最初の DWord を下位とする」: 64 ビット値のデータエンコーディングを設定します。Modbus デバイスでは 64 ビットデータ型に 4 つの連続するレジスタアドレスが使用されます。ドライバーは最初の DWord を 64 ビット値の下位 DWord または上位 DWord として読み取ることができます。デフォルトは「有効」で最初の DWord が下位となり、32 ビットデータ型のデフォルトの規則に従います。

注記: この設定は Omni モデルには適用されません。このモデルでは必ず Modbus バイトオーダーが使用されます。

「Modicon ビットオーダー」: 有効な場合、ドライバーはレジスタに対する読み書きの際にビットオーダーを反転して Modicon Modsoft プログラミングソフトウェアの規則に従います。たとえば、このオプションが有効になっている場合、アドレス 40001.0/1 への書き込みはこのデバイスのビット 15/16 に影響します。このオプションはデフォルトで無効になっています。

次の例では、そのドライバーが使用しているレジスタ内のビットアドレス指定がゼロベースか 1 ベースかに応じて、1 から 16 番目のビットは 0-15 ビットまたは 1-16 ビットを表します。
MSB = 最上位ビット
LSB = 最下位ビット

「Modicon ビットオーダー」が有効

MSB															LSB
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

「Modicon ビットオーダー」が無効

MSB															LSB
16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1

「Long を 10 進数として扱う」: 有効な場合、ドライバーは倍精度符号なしの Long データ型と DWord データ型を 0 から 99999999 の範囲の値としてエンコード/デコードします。このフォーマットでは各 Word が 0 から 9999 の値を表します。この範囲を超える読み取られた値はクランプされませんが、動作は定義されていません。読み取られた値はすべて [読み取られた値] = 上位 Word * 10000 + 下位 Word、という式を使用してデコードされます。99999999 より大きい書き込まれた値は最大値にクランプされます。書き込まれた値はすべて、生データ = [書き込まれた値]/10000 + [書き込まれた値] % 10000 という式を使用してエンコードされます。

設定に関するヒント

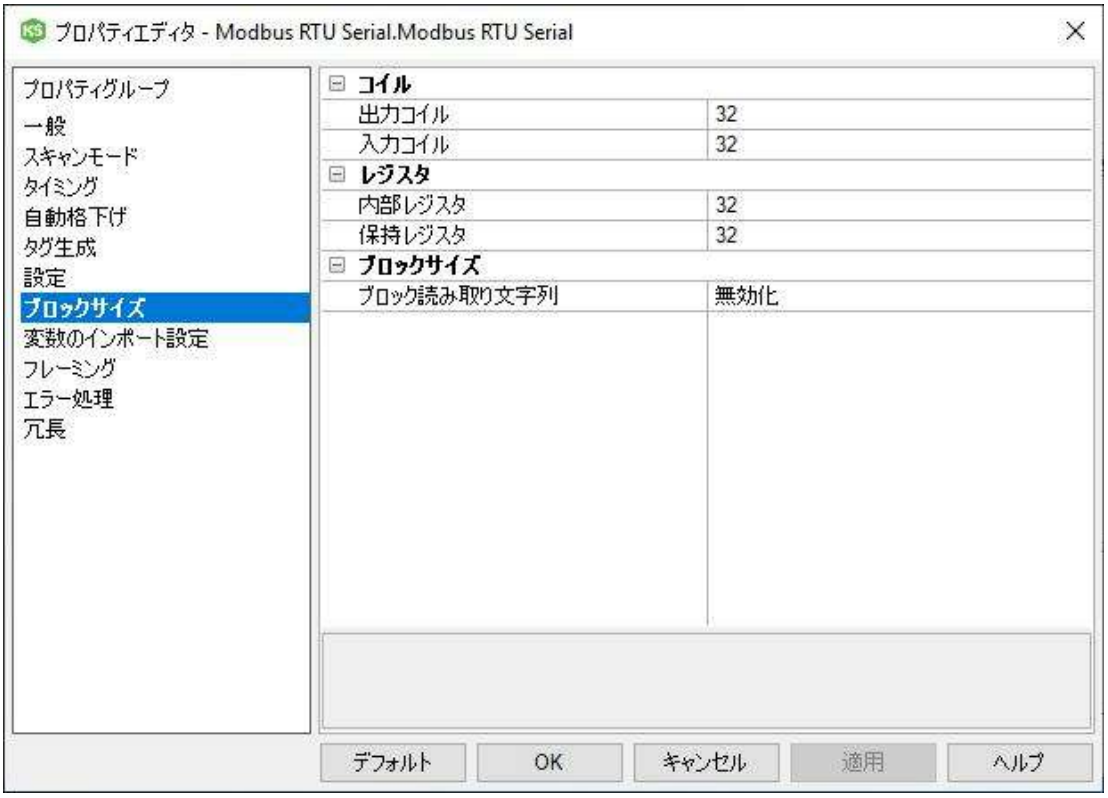
データ型	Modbus バイトオーダー	最初の Word を下位とする	最初の DWord を下位とする
Word、Short、BCD	適用可能	なし	なし
Float、DWord、Long、LBCD	適用可能	適用可能	なし
Double	適用可能	適用可能	適用可能

必要な場合、以下の情報とデバイスのドキュメントを参照して、データエンコーディングオプションの正しい設定を調べてください。

ほとんどの Modbus デバイスではデフォルト設定で問題ありません。

データエンコーディングのオプション	データエンコーディング	
Modbus バイトオーダー	上位バイト (15..8)	下位バイト (7..0)
Modbus バイトオーダー	下位バイト (7..0)	上位バイト (15..8)
最初の Word を下位とする	上位 Word (31..16) 64 ビットデータ型での DWord の上位 Word (63..48)	下位 Word (15..0) 64 ビットデータ型での DWord の下位 Word (47..32)
最初の Word を下位とする	下位 Word (15..0) 64 ビットデータ型での DWord の下位 Word (47..32)	上位 Word (31..16) 64 ビットデータ型での DWord の上位 Word (63..48)
最初の DWord を下位とする	上位 DWord (63..32)	下位 DWord (31..0)
最初の DWord を下位とする	下位 DWord (31..0)	上位 DWord (63..32)

デバイスのプロパティ - ブロックサイズ



コイル

「出力コイル」: 出力ブロックのサイズをビット数で指定します。コイルは 8 から 2000 ポイント (ビット) の範囲で一度に読み取ることができます。ブロックサイズを大きくすると、1 回の要求でより多くのポイントがデバイスから読み取られます。ブロックサイズを小さくすると、デバイス内の連続しない位置からデータを読み取ることができます。デフォルトの設定は 32 です。

「入力コイル」: 入力ブロックのサイズをビット数で指定します。コイルは 8 から 2000 ポイント (ビット) の範囲で一度に読み取ることができます。ブロックサイズを大きくすると、1 回の要求でより多くのポイントがデバイスから読み取られます。ブロックサイズを小さくすると、デバイス内の連続しない位置からデータを読み取ることができます。デフォルトの設定は 32 です。

「レジスタ」

「内部レジスタ」: 内部レジスタのブロックサイズをビット数で指定します。1 から 125 の標準 16 ビット Modbus レジスタを一度で読み取ることができます。ブロックサイズを大きくすると、1 回の要求でより多くのレジスタ値がデバイスから読み取られます。ブロックサイズを小さくすると、デバイス内の連続しない位置からデータを読み取ることができます。デフォルトの設定は 32 です。

「保持レジスタ」: 保持レジスタのブロックサイズをビット数で指定します。1 から 125 の標準 16 ビット Modbus レジスタを一度で読み取ることができます。ブロックサイズを大きくすると、1 回の要求でより多くのレジスタ値がデバイスから読み取られます。ブロックサイズを小さくすると、デバイス内の連続しない位置からデータを読み取ることができます。デフォルトの設定は 32 です。

警告: レジスタのブロックサイズとして 120 より大きい値が設定され、任意のタグに 32 ビットまたは 64 ビットデータ型が使用されている場合、「ブロックに不良アドレスがあります」というエラーが発生することがあります。これを防止するには、ブロックサイズの値を 120 に減らしてください。

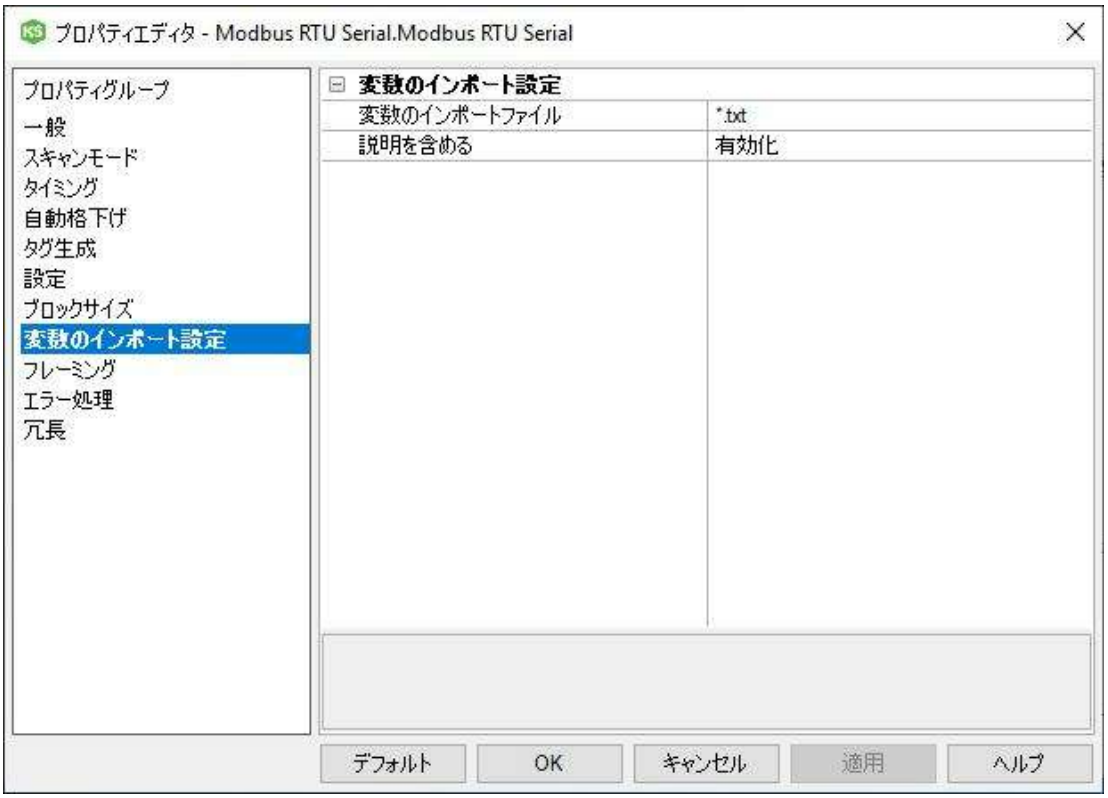
ブロックサイズ

「ブロック読み取り文字列」: 通常は個別に読み取る文字列タグをグループ/ブロックで読み取ります。文字列タグは選択したブロックサイズに応じてグループ化されます。ブロック読み取りは Modbus モデルの文字列タグに対してのみ実行できます。デフォルトの設定は「無効」です。

デバイスのプロパティ - 変数のインポート設定

「変数のインポート設定」の各パラメータでは、自動タグデータベース生成に使用する変数インポートファイルの場所を指定します。

Modbus ドライバー向け CSV ファイルの詳細については、「Modbus ドライバー向け CSV ファイルの作成」を参照してください。



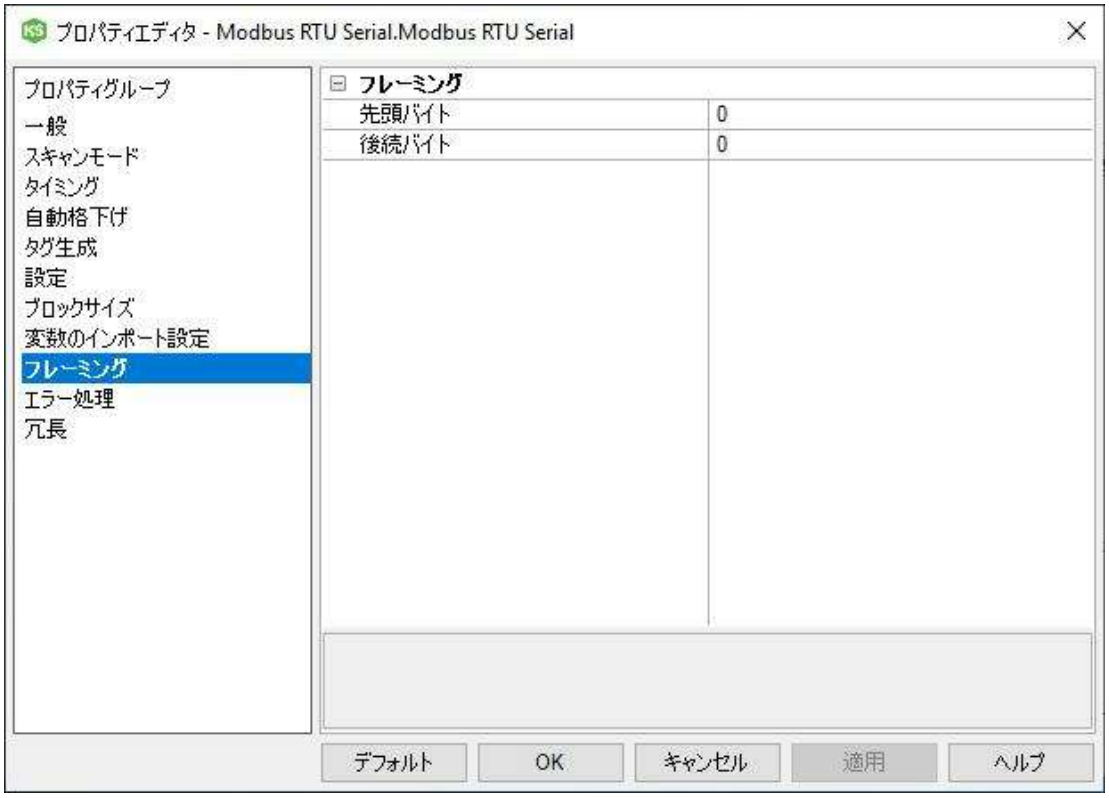
「変数のインポートファイル」: このパラメータは、自動タグデータベース生成に使用する変数インポートファイルの正確な場所にブラウズするときに使用します。

「説明を含める」: 有効な場合、インポートされたタグの説明が使用されます (ファイル内に存在する場合)。

自動タグデータベース生成機能の設定 (および変数インポートファイルの作成方法) については、自動タグデータベース生成を参照してください。

デバイスのプロパティ - フレーミング

一部のターミナルサーバーでは Modbus フレームにデータが追加されます。このため、フレーミングパラメータを使用して、応答メッセージ内の追加バイトを無視するようドライバーを設定できます。



フレーミング

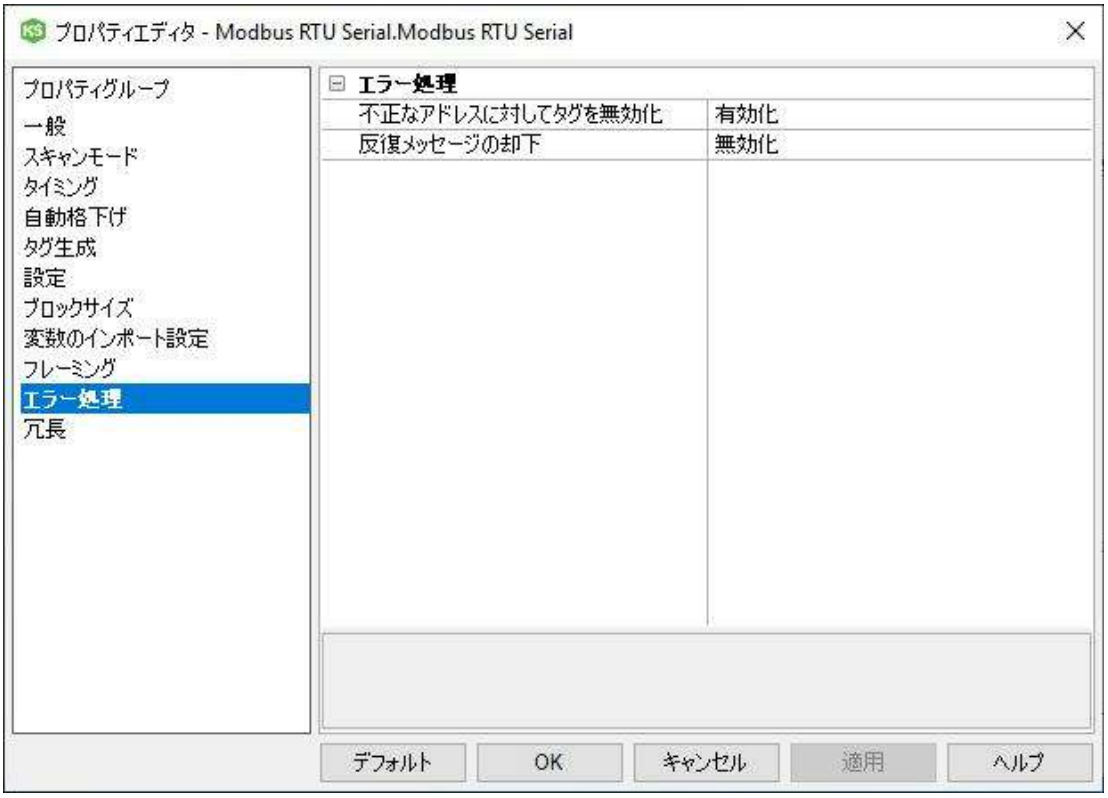
「Modbus TCP フレーミング」: ドライバーが Modbus TCP フレームと MBAP ヘッダーを使用している場合は、**「有効化」**を選択します。デフォルトでは無効になっています。
ヒント: この設定は、ネイティブの Modbus TCP デバイスとの通信時に有効にする必要があります。

「先頭バイト」: Modbus の応答の先頭に付加されるバイト数を指定します。値の範囲は 0 から 8 です。

「後続バイト」: Modbus の応答の末尾に付加されるバイト数を指定します。値の範囲は 0 から 8 です。

デバイスのプロパティ - エラー処理

エラー処理パラメータは、デバイスからのエラーへの対処方法を指定します。

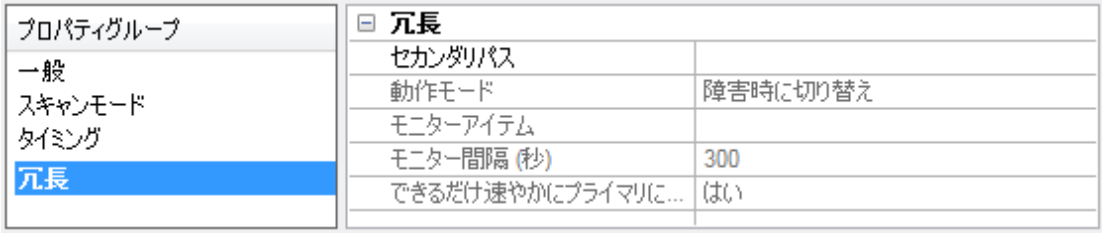


「不正なアドレスでタグを無効化」: 有効にした場合、デバイスがデータブロックの読み取りに応答して Modbus 例外コード 2 (不正なアドレス) または 3 (ポイント数などの不正なデータ) を返した場合にドライバーはそのブロックのポーリングを停止します。無効にした場合、ドライバーはそのデータブロックを引き続きポーリングします。デフォルト設定では有効になっています。

「反復メッセージの却下」: 有効にした場合、ドライバーは反復メッセージを無効な応答として解釈してその要求を再試行します。デフォルト設定では有効になっています。無効にした場合、ドライバーは反復メッセージを受け付けます。

注記: 一部のメッセージリレー装置は Modbus の要求をドライバーにエコーバックします。

デバイスのプロパティ - 冗長



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。

自動タグデータベース生成

Modbus RTU シリアルドライバーでは自動タグデータベース生成が利用されるため、ドライバーはデバイスのラダープログラムによって使用されるデータポイントにアクセスするタグを自動的に作成できます。タグデータベースの構築に必要な情報をデバイスに対して照会可能な場合もありますが、このドライバーは代わりに**変数インポートファイル**を使用する必要があります。変数インポートファイルは Concept および ProWORX デバイスプログラミングアプリケーションを使用して生成できます。

変数インポートファイルの作成

このインポートファイルは、多くのデバイスプログラミングアプリケーションのデフォルトのエクスポートファイルフォーマットであるセミコロン区切りテキスト .TXT フォーマットでなければなりません。

変数インポートファイルの作成方法については、技術情報「Modbus ドライバー向け CSV ファイルの作成」を参照してください。

このドライバーは、自動タグデータベース生成をサポートするすべてのドライバーに共通する基本設定に加え、追加の設定を必要とします。特別な設定として変数インポートファイルの名前や場所などを指定する必要があり、これらはデバイスウィザードの「変数のインポート設定」ステップの実行中に指定します。または、後で指定する場合は**「デバイスのプロパティ」「変数のインポート設定」**の順に選択します。

詳細については、変数のインポート設定を参照してください。

サーバー構成

アプリケーション固有のニーズに合わせて自動タグデータベース生成をカスタマイズできます。プライマリ制御オプションは、デバイスウィザードのデータベース作成ステップの実行中に指定します。または、後で指定する場合は**「デバイスのプロパティ」**｜「タグ生成」の順に選択します。

操作

構成に応じて、タグ生成はサーバープロジェクトが開始したときに自動的に開始するか、後から手動で開始できます。「イベントログ」には、タグ生成プロセスの開始時刻、変数インポートファイルの処理中に発生したエラー、このプロセスの完了時刻が示されます。

統計アイテム

統計アイテムには、デフォルトでは集められない、追加の診断情報から集められたデータが使用されます。統計アイテムを使用するには、通信診断が有効になっている必要があります。通信診断を有効にするには、Project View でチャネルを右クリックし、**「プロパティ」 | 「診断を有効にする」**の順にクリックします。または、チャネルをダブルクリックし、**「Enable Diagnostics」**を選択します。

チャネルレベルの統計アイテム

チャネルレベルの統計アイテムの構文は <チャネル>.*_Statistics* です。

注記: チャネルレベルの統計は、デバイスレベルの同じアイテムを合計したものです。

アイテム	データ型	アクセス	説明
_CommFailures	DWord	読み取り/書き込み	通信が失敗した(つまり再試行回数の上限に達した) 回数の合計
_ErrorResponses	DWord	読み取り/書き込み	受信した有効なエラー応答の総数
_ExpectedResponses	DWord	読み取り/書き込み	予想される受信応答の総数
_LastResponseTime	String	読み取り専用	最後に有効な応答を受信した時刻
_LateData	DWord	読み取り/書き込み	(指定されているスキャン速度に基づいて) 予想よりも遅いタグの読み取りの合計回数。この値は、DNR エラー状態によって増加しません。タグは、通信失敗の後の最初の読み取りで、(実際には遅延であっても) 遅延としてカウントされません。これは仕様によるものです。
_MsgResent	DWord	読み取り/書き込み	再試行として送信されたメッセージの総数
_MsgSent	DWord	読み取り/書き込み	初回に送信されたメッセージの総数
_MsgTotal	DWord	読み取り専用	送信されたメッセージ (_MsgSent + _MsgResent) の総数
_PercentReturn	Float	読み取り専用	初回の送信 (Sent) に対する予想される応答 (Received) の割合 (パーセント)
_PercentValid	Float	読み取り専用	送信された要求の総数 (_MsgTotal) に対する受信した有効な応答の総数 (_TotalResponses) の割合 (パーセント)
_Reset	Bool	読み取り/書き込み	すべての診断カウンタをリセットします。_Reset タグに書き込むと、すべての診断カウンタがこのレベルでリセットされます。
_RespBadChecksum	DWord	読み取り/書き込み	チェックサムエラーがある応答の総数
_RespTimeouts	DWord	読み取り/書き込	いかなる種類の応答も受信できなかったメッセージの総数

アイテム	データ型	アクセス	説明
		み	
_RespTruncated	DWord	読み取り/書き込み	部分的な応答のみを受信したメッセージの総数
_TotalResponses	DWord	読み取り専用	受信した有効な応答の総数 (_ErrorResponses + _ExpectedResponses)

シミュレーションモードでは統計アイテムは更新されません *(デバイスの一般プロパティを参照)*。

デバイスレベルの統計アイテム

デバイスレベルの統計アイテムの構文は <チャネル>.<デバイス>.*_Statistics* です。

アイテム	データ型	アクセス	説明
_CommFailures	DWord	読み取り/書き込み	通信が失敗した (つまり再試行回数の上限に達した) 回数の合計
_ErrorResponses	DWord	読み取り/書き込み	受信した有効なエラー応答の総数
_ExpectedResponses	DWord	読み取り/書き込み	予想される受信応答の総数
_LastResponseTime	String	読み取り専用	最後に有効な応答を受信した時刻
_LateData	DWord	読み取り/書き込み	(指定されているスキャン速度に基づいて) 予想よりも遅いタグの読み取りの合計回数。この値は、DNR エラー状態によって増加しません。タグは、通信失敗の後の最初の読み取りで、(実際には遅延であっても) 遅延としてカウントされません。これは仕様によるものです。
_MsgResent	DWord	読み取り/書き込み	再試行として送信されたメッセージの総数
_MsgSent	DWord	読み取り/書き込み	初回に送信されたメッセージの総数
_MsgTotal	DWord	読み取り専用	送信されたメッセージ (_MsgSent + _MsgResent) の総数
_PercentReturn	Float	読み取り専用	初回の送信 (Sent) に対する予想される応答 (Received) の割合 (パーセント)
_PercentValid	Float	読み取り専用	送信された要求の総数 (_MsgTotal) に対する受信した有効な応答の総数 (_TotalResponses) の割合 (パーセント)
_Reset	Bool	読み取り/書き込み	すべての診断カウンタをリセットします。_Reset タグに書き込むと、すべての診断カウンタがこのレベルでリセットされます。
_RespBadChecksum	DWord	読み取り/書き込み	チェックサムエラーがある応答の総数

アイテム	データ型	アクセス	説明
_RespTimeouts	DWord	読み取り/書き込み	いかなる種類の応答も受信できなかったメッセージの総数
_RespTruncated	DWord	読み取り/書き込み	部分的な応答のみを受信したメッセージの総数
_TotalResponses	DWord	読み取り専用	受信した有効な応答の総数 (_ErrorResponses + _ExpectedResponses)

シミュレーションモードでは統計アイテムは更新されません *(デバイスの一般プロパティを参照)*。

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
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
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。
String	Null 終端 ASCII 文字列 Modbus モデルでは、HiLo LoHi バイトオーダー、8 バイトおよび 16 バイトの Omni フローコンピュータ文字列データがサポートされています。
Double*	64 ビット浮動小数点値 ドライバーは最後の 2 つのレジスタを上位 DWord、最初の 2 つのレジスタを下位 DWord とすることで、連続する 4 つのレジスタを倍精度値として解釈します。
Double の例	レジスタ 40001 が Double として指定されている場合、レジスタ 40001 のビット 0 は 64 ビットデータ型のビット 0 になり、レジスタ 40004 のビット 15 は 64 ビットデータ型のビット 63 になります。
Float*	32 ビット浮動小数点値 ドライバーは最後のレジスタを上位 Word、最初のレジスタを下位 Word とすることで、連続する 2 つのレジスタを単精度値として解釈します。
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。

*この説明は、64 ビットデータ型では最初の DWord を下位とし、32 ビットデータ型では最初の Word を下位とするデフォルトのデータ処理を前提としています。

アドレスの説明

アドレスの仕様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。

Modbus のアドレス指定
Magnetek GPD 515 ドライブのアドレス指定
Elliott フローコンピュータのアドレス指定
Daniels S500 フローコンピュータのアドレス指定
ダイナミックフルードメーターのアドレス指定
Omni フローコンピュータのアドレス指定
統計

関連項目: ファンクションコードの説明

Modbus のアドレス指定

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。ファンクションコードは 10 進数で表示されます。

詳細については、*ファンクションコードの説明*を参照してください。

5 桁のアドレス指定と 6 桁のアドレス指定

Modbus のアドレス指定では、アドレスの最初の桁はプライマリテーブルを示します。以降の桁はデバイスのデータアイテムを表します。データアイテムの最大値は 2 バイトの符号なし整数 (65,535) です。内部では、このドライバーはアドレステーブルとアイテム全体を表すのに 6 桁を必要とします。Modbus デバイスの多くはすべてのデータアイテムをサポートしていないことに注意してください。そのようなデバイスのアドレスを入力する際の混乱を回避するため、このドライバーはアドレスフィールドに入力されたものに従ってアドレスに "パディング" (桁を追加) します。プライマリテーブルタイプの後ろに最大 4 桁ある場合 (例: 4x、4xx、4xxx、4xxxx)、アドレスはそのままになるか、5 桁までゼロが追加されます。プライマリテーブルタイプの後ろに 5 桁ある場合 (例: 4xxxxx)、アドレスは変わりません。内部では、41、401、4001、40001、または 400001 として入力されたアドレスはすべて、プライマリテーブルタイプ 4 とデータアイテム 1 を示すアドレスを表します。

プライマリテーブル	説明
0	出力コイル
1	入力コイル
3	内部レジスタ
4	保持レジスタ

Modbus のアドレス指定 (10 進フォーマット)

アドレス	範囲	データ型	アクセス *	ファンクションコード
出力コイル	000001-065536	**Boolean**	読み取り/書き込み	01, 05, 15
入力コイル	100001-165536	**Boolean**	読み取り専用	02
内部レジスタ	300001-365536 300001-365535	**Word**、Short、BCD Float、DWord、	読み取り専用	04

アドレス	範囲	データ型	アクセス *	ファンクションコード
	300001-365533 3xxxxx.0/1-3xxxxx.15/16***	Long、LBCD Double Boolean		
String の内部レジスタ、HiLo バイトオーダー	300001.2H-365536.240H ピリオドの後ろのビット番号は 2 から 240 バイトの範囲の文字列長を示します。	**String****	読み取り専用	04
String の内部レジスタ、LoHi バイトオーダー	300001.2L-365536.240L ピリオドの後ろのビット番号は 2 から 240 バイトの範囲の文字列長を示します。	**String****	読み取り専用	04
保持レジスタ	400001-465536 400001-465535 400001-465533 4xxxxx.0/1-4xxxxx.15/16***	**Word**、Short、BCD Float、DWord、 Long、LBCD Double Boolean	読み取り/書き込み	03, 06, 16 03, 06, 16, 22
String の保持レジスタ、HiLo バイトオーダー	400001.2H-465536.240H ピリオドの後ろのビット番号は 2 から 240 バイトの範囲の文字列長を示します。	**String****	読み取り/書き込み	03, 16
String の保持レジスタ、LoHi バイトオーダー	400001.2L-465536.240L ピリオドの後ろのビット番号は 2 から 240 バイトの範囲の文字列長を示します。	**String****	読み取り/書き込み	03, 16

*"W40001" などのように、アドレスの先頭に "W" を付けることによって、すべての読み取り/書き込み可能アドレスを書き込み専用として設定できます。これによって、ドライバーはレジスタの指定したアドレスを読み取れなくなります。クライアントが書き込み専用タグを読み取ろうとすると、指定したアドレスへの最後に成功した書き込みの値が取得されます。成功した書き込みがない場合、クライアントは数値/文字列値の初期値である 0/NULL を受信します。

警告: 書き込み専用タグのクライアントアクセス権限を読み取り専用に設定した場合、これらのタグへの書き込みは失敗し、クライアントは数値/文字列値に対して必ず 0/NULL を受信します。

***詳細については、文字列のサポートを参照してください。*

****詳細については、設定の「ゼロベースのビットアドレス指定」を参照してください。*

Modbus のアドレス指定 (16 進フォーマット)

アドレス	範囲	データ型	アクセス	ファンクションコード
出力コイル	H000001-H0FFFF	**Boolean**	読み取り/書き込み	01, 05, 15
入力コイル	H100001-H1FFFF	**Boolean**	読み取り専用	02
内部レジスタ	H300001-H310000 H300001-H3FFFF H300001-H3FFFD	**Word**、Short、BCD Float、DWord、 Long、LBCD Double	読み取り専用	04

アドレス	範囲	データ型	アクセス	ファンクションコード
	H3xxxxx.0/1-H3xxxxx.F/10*	Boolean		
String の内部レジスタ、HiLo バイトオーダー	H300001.2H-H3FFFF.240H。 ビット番号は 2 から 240 バイトの範囲の文字列長を示します。	**String****	読み取り専用	04
String の内部レジスタ、LoHi バイトオーダー	H300001.2L-H3FFFF.240L。 ビット番号は 2 から 240 バイトの範囲の文字列長を示します。	**String****	読み取り専用	04
保持レジスタ	H400001-H410000 H400001-H4FFFF H400001-H4FFFD H4xxxxx.0/1-H4xxxxx.F/10*	**Word**、Short、BCD、Float、DWord、Long、LBCD、Double、Boolean	読み取り/書き込み	03, 06, 16 03, 06, 16, 22
String の保持レジスタ、HiLo バイトオーダー	H400001.2H-H4FFFF.240H。 ビット番号は 2 から 240 バイトの範囲の文字列長を示します。	**String****	読み取り/書き込み	03, 16
String の保持レジスタ、LoHi バイトオーダー	H400001.2L-H4FFFF.240L。 ビット番号は 2 から 240 バイトの範囲の文字列長を示します。	**String****	読み取り/書き込み	03, 16

**詳細については、設定の「ゼロベースのビットアドレス指定」を参照してください。*
***詳細については、文字列のサポートを参照してください。*

文字列のサポート

Modbus モデルでは保持レジスタメモリを ASCII 文字列として読み書きできます。文字列データに保持レジスタを使用している場合、各レジスタに 2 バイトの ASCII データが格納されます。文字列を定義する際に、そのレジスタにおける ASCII データの順序を選択できます。文字列の長さは 2 から 240 バイトの範囲で指定でき、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。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。

文字列の例

1. 40200 で開始し、長さが 100 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、"40200.100H" と入力します。
2. 40500 で開始し、長さが 78 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、"40500.78L" と入力します。

注記: デバイスで許可される書き込み要求の最大サイズによって文字列の長さが制限されることがあります。文字列タグを使用しているときに、サーバーのイベントウィンドウで「デバイス <デバイス> のアドレス <アドレス> に書き込めません: デバイスは例外コード 3 を返しました」というエラーメッセージを受信した場合、これは文字列の長さがそのデバイスに適していなかったことを意味します。可能な場合、文字列を短くします。

通常のアドレスの例

1. 255 番目の出力コイルのアドレスは、10 進アドレス指定を使用した場合には '0255' となり、16 進アドレス指定を使用した場合には 'H0FF' となります。

2. 一部のドキュメントでは Modbus のアドレスはファンクションコードと位置によって示されています。たとえば、ファンクションコード 3、位置 2000 のアドレスは '42000' または 'H47D0' となります。先頭の '4' は保持レジスタ (ファンクションコード 3) を表しています。
3. 一部のドキュメントでは Modbus のアドレスはファンクションコードと位置によって示されています。たとえば、ファンクションコード 5、位置 100 を設定するとアドレスは '0100' または 'H064' となります。先頭の '0' は出力コイル (ファンクションコード 5) を表しています。このアドレスに 1 を書き込むとコイルが設定され、0 を書き込むとコイルがリセットされます。

配列のサポート

内部レジスタと保持レジスタの位置では Boolean 型と String 型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。入力コイルと出力コイルでも配列がサポートされます (Boolean データ型)。配列のアドレス指定には 2 つの方法があります。次の例では保持レジスタの位置が使用されています。

4xxxx [行数] [列数]
4xxxx [列数] - この方法では行数が 1 であるものと見なされます。

配列の場合、行数と列数を掛けた値が、デバイスのレジスタ/コイルタイプに割り当てられているブロックサイズを超えてはなりません。32 ビットデータ型のレジスタ配列の場合、行数と列数を掛けた値を 2 倍にした値がブロックサイズを超えてはなりません。

パックタイプのコイルアドレス

パックタイプのコイルアドレスでは、複数の連続するコイルにアナログ値としてアクセスできます。この機能は入力コイルと出力コイルの両方で、ポーリングモードでのみ使用できます。有効な唯一のデータ型が Word です。構文は以下のとおりです。

出力コイル: 0*xxxxx#nn* Word 読み取り/書き込み
入力コイル: 1*xxxxx#nn* Word 読み取り専用

ここで、*xxxxx* は 1 つ目のコイルのアドレス (10 進と 16 進の値を使用可能) であり、*nn* はアナログ値にパックされるコイルの数 (1-16、10 進のみ) です。

開始アドレスがアナログ値の LSB (最下位ビット) となるビットオーダーになります。

Magnetek GPD 515 ドライブのアドレス指定

この表には、Magnetek GPD 515 ドライブから使用可能なデータの一般的な範囲が示されています。Modbus RTU アドレス指定によって特定のドライブパラメータにアクセスする方法については、Magnetek Modbus RTU Technical Manual、部品番号 TM4025 を参照してください。いずれの場合も、必要なアドレスの先頭に文字 H (16 進数によるアドレス指定を示す) を指定する必要があります。該当する場合、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。

Magnetek GPD 515 のアドレス指定 (16 進フォーマット)

アドレス	範囲	データ型	アクセス
ビットレベルアクセスのコマンドレジスタ	H40001-H4000F H4xxxx.0/1-H4xxxx.F/10*	**Word**、Short Boolean	読み取り/書き込み
ビットレベルアクセスのモニターレジスタ	H40010-H4001A H4xxxx.0/1-H4xxxx.F/10*	**Word**、Short Boolean	読み取り専用
ビットレベルアクセスのドライブパラメータレジスタ (モニターのみ)	H40020-H40097 H4xxxx.0/1-H4xxxx.F/10*	**Word**、Short Boolean	読み取り専用

アドレス	範囲	データ型	アクセス
ビットレベルアクセスのドライブパラメータレジスタ	H40100-H4050D H4xxxx.0/1-H4xxxx.F/10*	**Word**、Short Boolean	読み取り/書き込み
特殊レジスタ	H4FFDD ACCEPT H4FFFD ENTER	Word、Short	書き込み専用

詳細については、設定の「ゼロベースのビットアドレス指定」を参照してください。

例

ドライバーのオペレーションステータス (アドレス 02BH) にアクセスするには、アドレス H4002B を入力します。

注記: Magnetek デバイスを OPC サーバープロジェクトに追加する場合、「ゼロベースアドレス指定」が無効になっていることを確認する必要があります。このパラメータが正しく設定されていない場合、Modbus RTU ドライバーでは Magnetek のすべてのアドレスが 1 だけオフセットしてしまいます。

配列のサポート

保持レジスタの位置では Boolean 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。次の例では保持レジスタの位置が使用されています。

4xxxx [行数] [列数]
4xxxx [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が、デバイスのレジスタタイプに割り当てられているブロックサイズを超えてはなりません。

Elliott フローコンピュータのアドレス指定

該当する場合、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。

アドレス	範囲	データ型	アクセス
出力コイル	000001-065536	**Boolean**	読み取り/書き込み
入力コイル	100001-165536	**Boolean**	読み取り専用
内部レジスタ	300001-365536 300001-365535 3xxxxx.0/1-3xxxxx.15/16*	**Word**、Short、BCD Float、DWord、Long、LBCD Boolean	読み取り専用
保持レジスタ	400001-465536 400001-465535 4xxxxx.0/1-4xxxxx.15/16*	**Word**、Short、BCD** Float、DWord、Long、LBCD Boolean	読み取り/書き込み

詳細については、設定の「ゼロベースのビットアドレス指定」を参照してください。
**アドレス範囲 405001 から 405315 と 407001 から 407315 は 32 ビットレジスタです。405001 から 405315 の範囲のアドレスではデフォルトのデータ型として Long が使用されます。

407001 から 407315 の範囲のアドレスではデフォルトのデータ型として Float が使用されます。これらのアドレスレジスタは 32 ビットなので、Float、DWord、Long、LBCD データ型のみを使用できます。配列は使用できません。

配列のサポート

内部レジスタと保持レジスタの位置では Boolean 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。次の例では保持レジスタの位置が使用されています。

4xxxx [行数] [列数]
4xxxx [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が、デバイスのレジスタタイプに割り当てられているブロックサイズを超えてはなりません。32 ビットデータ型の配列の場合、行数と列数を掛けた値を 2 倍にした値がブロックサイズを超えてはなりません。

Daniels S500 フローコンピュータのアドレス指定

該当する場合、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。ファンクションコードは 10 進数で表示されます。

詳細については、ファンクションコードの説明を参照してください。

アドレス	16 進範囲	10 進範囲	データ型	ファンクションコード	アクセス
合計	000-0FF	4096-4351	Double	03	読み取り専用
計算/測定された変数	100-24F	4352-4687	Float	03, 16	読み取り/書き込み
計算定数	250-28F	4688-4751	Float	03, 16	読み取り/書き込み
キーパッドのデフォルト値	290-2AF	4752-4783	Float	03, 16	読み取り/書き込み
アラームとスケール変換の定数	2B0-5FF	4784-5631	Float	03, 16	読み取り/書き込み
ステータス/制御	700-7FF	5888-6143	**Boolean**	01, 05	読み取り/書き込み
アラーム	800-FFF	6144-8191	**Boolean**	02	読み取り専用

ダイナミックフルードメーターのアドレス指定

該当する場合、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。

ダイナミックフルードメーターのアドレス指定 (10 進フォーマット)

アドレス	範囲	データ型	アクセス
保持レジスタ (16 ビット)	400000-407000 400000-406999 408001-465535 408001-465534 4xxxxx.0/1-4xxxxx.15/16*	**Word**、Short、BCD Float、DWord、Long、LBCD **Word**、Short、BCD	読み取り/書き込み

アドレス	範囲	データ型	アクセス
		Float、DWord、Long、LBCD Boolean	
保持レジスタ (32 ビット)	407001-408000	**Float**	読み取り/書き込み
String の保持レジスタ、HiLo バイトオーダー	400000.2H-407000.240H 408001.2H-465535.240H ピリオドの後ろのビット番号は 2 から 240 バイトの範囲の文字列長を示します。	**String**	読み取り/書き込み
String の保持レジスタ、LoHi バイトオーダー	400000.2L-407000.240L 408001.2L-465535.240L ピリオドの後ろのビット番号は 2 から 240 バイトの範囲の文字列長を示します。	**String**	読み取り/書き込み

詳細については、設定の「ゼロベースのビットアドレス指定」を参照してください。

ダイナミックフルードメーターのアドレス指定 (16 進フォーマット)

アドレス	範囲	データ型	アクセス
保持レジスタ (16 ビット)	H400000-H401B58 H400000-H401B57 H401F41-H40FFFF H401F41-H40FFFE H4xxxxx.0/1-H4xxxxx.F/10*	**Word**、Short、BCD Float、DWord、Long、LBCD **Word**、Short、BCD Float、DWord、Long、LBCD Boolean	読み取り/書き込み
保持レジスタ (32 ビット)	H401B59-H401F40	**Float**	読み取り/書き込み
String の保持レジスタ、HiLo バイトオーダー	H400000.2H-H401B58.240H H401F41.2H-H40FFFF.240H ピリオドの後ろのビット番号は 2 から 240 バイトの範囲の文字列長を示します。	**String**	読み取り/書き込み
String の保持レジスタ、LoHi バイトオーダー	H400000.2L-H401B58.240L H401F41.2L-H0FFFF.240L ピリオドの後ろのビット番号は 2 から 240 バイトの範囲の文字列長を示します。	**String**	読み取り/書き込み

詳細については、設定の「ゼロベースのビットアドレス指定」を参照してください。

注記: このドライバーでは、ダイナミックフルードメーターモデルのすべてのアドレスが "4" で始まる必要があります。この 4 はダイナミックフルードメーターのドキュメントに必ずしも明記されていないことがあります。たとえば、ユーザーは "アドレス 3001 にあるユニット ID" などという記述を目にすることがあります。この値はサーバーで "403001" としてアドレス指定する必要があります。

文字列のサポート

ダイナミックフルードメーターモデルでは保持レジスタメモリを ASCII 文字列として読み書きできます。文字列データに保持レジスタを使用している場合、各レジスタに 2 バイトの ASCII データが格納されます。文字列を定義する際に、そのレジスタにおける ASCII データの順序を選択できます。文字列の長さは 2 から 240 バイトの範囲で指定でき、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。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。

文字列の例

1. 40200 で開始し、長さが 100 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、"40200.100H" と入力します。
2. 40500 で開始し、長さが 78 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、"40500.78L" と入力します。

注記: デバイスで許可される書き込み要求の最大サイズによって文字列の長さが制限されることがあります。文字列タグを使用しているときに、サーバーのイベントウィンドウで「デバイス <デバイス> のアドレス <アドレス> に書き込めません: デバイスは例外コード 3 を返しました」というエラーメッセージを受信した場合、これは文字列の長さがそのデバイスに適していなかったことを意味します。可能な場合、文字列を短くしてみてください。

Omni フローコンピュータのアドレス指定

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。

アドレス	範囲	データ型	アクセス
デジタル I/O ポイント	1001-1024	**Boolean**	読み取り/書き込み
プログラミング可能 Boolean ポイント	1025-1088	**Boolean**	読み取り/書き込み
計測管ステータス/アラームポイント	1n01-001n59 1n76-1n99 n = 計測管の数	**Boolean**	読み取り/書き込み
マイクロモーションアラームステータスポイント	1n60-1n75 n = 計測管の数	**Boolean**	読み取り/書き込み
ユーザースクラッチパッド Boolean ポイント	1501-1599 1601 -1649	**Boolean**	読み取り/書き込み
ユーザースクラッチパッドワンショットポイント	1650-1699	**Boolean**	読み取り/書き込み
コマンド Boolean ポイント/変数	1700-1798	**Boolean**	読み取り/書き込み
メーターステーションアラーム/ステータスポイント	1801-1899	**Boolean**	読み取り/書き込み

アドレス	範囲	データ型	アクセス
プローバーアラーム/ステータスポイント	1901-1967	**Boolean**	読み取り/書き込み
メーター積算計ロールオーバーフラグ	2n01-2n37 n = 計測管の数	**Boolean**	読み取り/書き込み
その他のメーターステーションアラーム/ステータス	2601-2623	**Boolean**	読み取り/書き込み
ステーション積算計ロールオーバーフラグ	2801-2851	**Boolean**	読み取り/書き込み
ステーション積算計 10 進分解能	2852-2862 2865-2999	**Boolean**	読み取り/書き込み

16 ビット整数データアドレス	範囲	データ型	アクセス
カスタムデータパケット #1	3001-3040	**Short**、Word、BCD	読み取り/書き込み
カスタムデータパケット #2	3041-3056	**Short**、Word、BCD	読み取り/書き込み
カスタムデータパケット #3	3057-3096	**Short**、Word、BCD	読み取り/書き込み
その他16 ビット整数データ	3097-3099 3737-3799 3875-3899	**Short**、Word、BCD	読み取り/書き込み
計測管 16 ビット整数データ	3n01-3n52 n = 計測管の数	**Short**、Word、BCD	読み取り/書き込み
スクラッチパッド 16 ビット整数データ	3501-3599	**Short**、Word、BCD	読み取り/書き込み
ユーザーディスプレイ #1	3601-3608	**Short**、Word、BCD	読み取り/書き込み
ユーザーディスプレイ #2	3609-3616	**Short**、Word、BCD	読み取り/書き込み
ユーザーディスプレイ #3	3617-3624	**Short**、Word、BCD	読み取り/書き込み
ユーザーディスプレイ #4	3625-3632	**Short**、Word、BCD	読み取り/書き込み
ユーザーディスプレイ #5	3633-3640	**Short**、Word、BCD	読み取り/書き込み
ユーザーディスプレイ #6	3641-3648	**Short**、Word、BCD	読み取り/書き込み
ユーザーディスプレイ #7	3649-3656	**Short**、Word、BCD	読み取り/書き込み
ユーザーディスプレイ #8	3657-3664	**Short**、Word、BCD	読み取り/書き込み
生データアーカイブレコードへのアクセス	3701-3736	**Short**、Word、BCD	読み取り/書き込み

16 ビット整数データアドレス	範囲	データ型	アクセス
メーターステーション 16 ビット整数データ	3800-3842	**Short**、Word、BCD	読み取り/書き込み
メーター #1 バッチシーケンス	3843-3848	**Short**、Word、BCD	読み取り/書き込み
メーター #2 バッチシーケンス	3849-3854	**Short**、Word、BCD	読み取り/書き込み
メーター #3 バッチシーケンス	3855-3860	**Short**、Word、BCD	読み取り/書き込み
メーター #4 バッチシーケンス	3861-3866	**Short**、Word、BCD	読み取り/書き込み
フローコンピュータ日時	3867-3874	**Short**、Word、BCD	読み取り/書き込み
プローバー 16 ビット整数データ	3901-3999	**Short**、Word、BCD	読み取り/書き込み

8 文字 ASCII 文字列データ	範囲	データ型	アクセス
計測管 ASCII データ	4n01-4n39 n = 計測管の数	**String**	読み取り/書き込み
スクラッチパッド ASCII データ	4501-4599	**String**	読み取り/書き込み
ユーザーディスプレイ定義変数	4601 -4640	**String**	読み取り/書き込み
ステーション補助入力変数	4707-4710	**String**	読み取り/書き込み
メーターステーション ASCII データ	4801-4851	**String**	読み取り/書き込み
メーター #1 バッチ ID	4852-4863	**String**	読み取り/書き込み
メーター #2 バッチ ID	4864-4875	**String**	読み取り/書き込み
メーター #3 バッチ ID	4876-4887	**String**	読み取り/書き込み
メーター #4 バッチ ID	4888-4899	**String**	読み取り/書き込み
プローバー ASCII 文字列データ	4901-4942	**String**	読み取り/書き込み

32 ビット整数データ	範囲	データ型	アクセス
計測管 32 ビット整数データ	5n01-5n99 n = 計測管の数	**Long**、DWord、LBCD、Float	読み取り/書き込み
スクラッチパッド 32 ビット整数データ	5501-5599	**Long**、DWord、	読み取り/書き込み

32 ビット整数データ	範囲	データ型	アクセス
		LBCD、Float	
ステーション 32 ビット整数データ	5801-5818	**Long**、DWord、LBCD、Float	読み取り/書き込み
メーター #1 バッチサイズ	5819-5824	**Long**、DWord、LBCD、Float	読み取り/書き込み
メーター #2 バッチサイズ	5825-5830	**Long**、DWord、LBCD、Float	読み取り/書き込み
メーター #3 バッチサイズ	5831-5836	**Long**、DWord、LBCD、Float	読み取り/書き込み
メーター #4 バッチサイズ	5837-5842	**Long**、DWord、LBCD、Float	読み取り/書き込み
その他の 32 ビット計測管データ	5843-5899	**Long**、DWord、LBCD、Float	読み取り/書き込み
プローバー 32 ビット整数データ	5901-5973	**Long**、DWord、LBCD、Float	読み取り/書き込み
コンパクトプローバー TDVOL/TDFMP パルス	5974-5999	**Long**、DWord、LBCD、Float	読み取り/書き込み

32 ビット IEEE 浮動小数点データ	範囲	データ型	アクセス
予約済みデータ	6001-7000	**Float**、Long、DWord、LBCD	読み取り/書き込み
デジタルからアナログへの出力	7001-7024	**Float**、Long、DWord、LBCD	読み取り/書き込み
ユーザー変数	7025-7088	**Float**、Long、DWord、LBCD	読み取り/書き込み
プログラミング可能アキュムレータ	7089-7099	**Float**、Long、DWord、LBCD	読み取り/書き込み
計測管データ	7n01 - 7n99 n = 計測管の数	**Float**、Long、DWord、LBCD	読み取り/書き込み
スクラッチパッド データ	7501-7599	**Float**、Long、DWord、LBCD	読み取り/書き込み
PID 制御データ	7601-7623	**Float**、Long、DWord、LBCD	読み取り/書き込み
その他の計測管データ	7624-7699	**Float**、Long、DWord、LBCD	読み取り/書き込み
その他の変数	7701-7799	**Float**、Long、	読み取り/書き

32 ビット IEEE 浮動小数点データ	範囲	データ型	アクセス
		DWord、LBCD	込み
メーターステーションデータ	7801-7899	**Float**、Long、DWord、LBCD	読み取り/書き込み
プローバーデータ	7901-7918	**Float**、Long、DWord、LBCD	読み取り/書き込み
プローバーの設定データ	7919-7958	**Float**、Long、DWord、LBCD	読み取り/書き込み
最終較正データ	7959-7966	**Float**、Long、DWord、LBCD	読み取り/書き込み
較正中に却下されたデータ	7967-7990	**Float**、Long、DWord、LBCD	読み取り/書き込み
較正管データ	7991-8050	**Float**、Long、DWord、LBCD	読み取り/書き込み
較正平均データ	8051-8079	**Float**、Long、DWord、LBCD	読み取り/書き込み
較正管 - クライアントメーターデータ	8080-8199	**Float**、Long、DWord、LBCD	読み取り/書き込み
較正中のシリーズデータ	8200-8223	**Float**、Long、DWord、LBCD	読み取り/書き込み
較正中のメーターのデータ	8224-8230	**Float**、Long、DWord、LBCD	読み取り/書き込み
マス較正データ	8231-8500	**Float**、Long、DWord、LBCD	読み取り/書き込み
その他の計測管 #1	8501-8599	**Float**、Long、DWord、LBCD	読み取り/書き込み
その他の計測管 #2	8601-8699	**Float**、Long、DWord、LBCD	読み取り/書き込み
その他の計測管 #3	8701-8799	**Float**、Long、DWord、LBCD	読み取り/書き込み
その他の計測管 #4	8801-8899	**Float**、Long、DWord、LBCD	読み取り/書き込み
ステーションの以前のバッチの平均データ	8901-8999	**Float**、Long、DWord、LBCD	読み取り/書き込み

16 ビット整数の設定データ	範囲	データ型	アクセス
計測管 #1	13001-13013	**Short**、Word、BCD	読み取り/書き込み
計測管 #2	13014-13026	**Short**、Word、BCD	読み取り/書き込み
計測管 #3	13027-13039	**Short**、Word、BCD	読み取り/書き込み
計測管 #4	13040-13052	**Short**、Word、BCD	読み取り/書き込み

16 ビット整数の設定データ	範囲	データ型	アクセス
プローバー構成	13053-13073	**Short**、Word、BCD	読み取り/書き込み
一般フロー構成	13074-13084	**Short**、Word、BCD	読み取り/書き込み
シリアルポート構成	13085-13128	**Short**、Word、BCD	読み取り/書き込み
PID 構成	13129-13160	**Short**、Word、BCD	読み取り/書き込み
PLC データ	13161-13299	**Short**、Word、BCD	読み取り/書き込み
ピアツーピア設定	13300-13499	**Short**、Word、BCD	読み取り/書き込み
生データアーカイブ	13500-13999	**Short**、Word、BCD	読み取り/書き込み

16 文字 ASCII 文字列データ	範囲	データ型	アクセス
フローコンピュータ構成	14001-14499	**String**	読み取り/書き込み

32 ビット整数データ	範囲	データ型	アクセス
フローコンピュータ構成	15001-16999	**Long**、DWord、LBCD、Float	読み取り/書き込み

32 ビット IEEE 浮動小数点データ	範囲	データ型	アクセス
フローコンピュータ構成	17001-18999	**Float**、Long、DWord、LBCD	読み取り/書き込み

サポートされる拡張 Omni タイプ

カスタムパケット
生データアーカイブ
テキストレポート
テキストアーカイブ

Omni カスタムパケット

Omni フローコンピュータでは、さまざまな範囲のメモリを単一のデータ構造にマッピングして、非常に効率の高い読み取りコマンド 1 つで読み取ることができます。これらのデータ構造をカスタムパケットと呼びます。

パケットの構成

各カスタムパケットにデータポイントのグループを 20 個まで含めることができます。各グループはその開始インデックスとデータポイント数によって定義されます。カスタムパケットの合計サイズが 250 バイトを超えてはなりません。カスタムパケットの定義に使用されるアドレスを以下に示します。

カスタムパケット 1 (アドレス 1)

3001 グループ 1 - 開始インデックス
3002 グループ 1 - ポイント数
から
3039 グループ 20 - 開始インデックス
3040 グループ 20 - ポイント数

カスタムパケット 2 (アドレス 201)
3041 グループ 1 - 開始インデックス
3042 グループ 1 - ポイント数
から
3055 グループ 20 - 開始インデックス
3056 グループ 20 - ポイント数

カスタムパケット 3 (アドレス 401)
3057 グループ 1 - 開始インデックス
3058 グループ 1 - ポイント数
から
3095 グループ 20 - 開始インデックス
3096 グループ 20 - ポイント数

注記: データはデバイスから 16 ビットレジスタとして返されます。デジタル I/O は 16 ビットのブロックでマッピングされる必要があります。

カスタムパケットのアドレス構文

カスタムパケット内の指定したオフセットにあるデータにアクセスするためにタグを作成できます。アドレス構文は次のとおりです。デフォルトのデータ型を**太字**で示しています。

アドレス	範囲	データ型	アクセス
CPn_o	n = パケット番号 (1-3) o = Word オフセット (0-125)	Word、**Short**、BCD、DWord、Long、LBCD、Float、String	読み取り専用
CPn_o.b	n = パケット番号 (1-3) o = Word オフセット (0-125) b = ビット番号 (0/1-15/16)*	**Boolean**	読み取り専用

詳細については、*設定*の「ゼロベースのビットアドレス指定」を参照してください。

注記:

1. 8 文字 ASCII 文字列データのみがサポートされます。
2. グループ構成に 16 文字 ASCII 文字列データアドレスが含まれている場合、データを 2 つの 8 文字 ASCII 文字列データアイテムとして読み取ることができます。

例

カスタムパケット #1 のマッピングを次のように定義します。

- 16 ビットのデジタル I/O (1001-1016)。
- 計測管 1 バッチデータの 15 個の 32 ビット整数 (5101 -5115)。

- アナログ出力の 12 個の 32 ビット単精度実数 (7001-7012)。
- 計測管の 4 つの 8 文字 ASCII 文字列 (4101-4104)。
- メーターステーションの 6 つの 8 文字 ASCII 文字列 (4808-4813)。
- フロー構成データの 2 つの 16 文字 ASCII 文字列 (14001-14002)。

注記: 合計で 222 バイトになります。カスタムパケット構成レジスタは次の値を持ちます。

3001 = 1001
3002 = 16
3003 = 5101
3004 = 15
3005 = 7001
3006 = 12
3007 = 4101
3008 = 4
3009 = 4808
3010 = 6
3011 = 14001
3012 = 2

デジタル I/O データにアクセスするタグは次のアドレスを持ちます (16 個の値すべてが Word 0 に含まれています):
CP1_0.0 (カスタムパケット 1 の Word 0、ビット 0 - マッピング先 1009)
CP1_0.1 (カスタムパケット 1 の Word 0、ビット 1 - マッピング先 1010)
...
CP1_0.6 (カスタムパケット 1 の Word 0、ビット 6 - マッピング先 1015)
CP1_0.7 (カスタムパケット 1 の Word 0、ビット 7 - マッピング先 1016)
CP1_0.8 (カスタムパケット 1 の Word 0、ビット 8 - マッピング先 1001)
CP1_0.9 (カスタムパケット 1 の Word 0、ビット 9 - マッピング先 1002)
...
CP1_0.14 (カスタムパケット 1 の Word 0、ビット 14 - マッピング先 1007)
CP1_0.15 (カスタムパケット 1 の Word 0、ビット 15 - マッピング先 1008)

計測管 1 バッチデータにアクセスするタグは次のアドレスを持ちます (32 ビット値それぞれが 2 Word を使用します):
CP1_1 (カスタムパケット 1 の Word 1 - マッピング先 5101)
CP1_3 (カスタムパケット 1 の Word 3 - マッピング先 5102)
...
CP1_29 (カスタムパケット 1 の Word 29 - マッピング先 5115)

アナログ出力データにアクセスするタグは次のアドレスを持ちます (32 ビット値それぞれが 2 Word を使用します):
CP1_31 (カスタムパケット 1 の Word 31 - マッピング先 7001)
CP1_33 (カスタムパケット 1 の Word 33 - マッピング先 7002)
...
CP1_53 (カスタムパケット 1 の Word 53 - マッピング先 7012)

計測管 8 文字 ASCII 文字列データにアクセスするタグは次のアドレスを持ちます (文字列値それぞれが 4 Word を使用します):
CP1_55 (カスタムパケット 1 の Word 55 - マッピング先 4101)
...
CP1_67 (カスタムパケット 1 の Word 67 - マッピング先 4104)

メーターステーション 8 文字 ASCII 文字列データにアクセスするタグは次のアドレスを持ちます (文字列値それぞれが 4 Word を使用します):
CP1_71 (カスタムパケット 1 の Word 71 - マッピング先 4808)

...
CP1_91 (カスタムパケット 1 の Word 91 - マッピング先 4813)

フロー構成 16 文字 ASCII 文字列データにアクセスするタグは次のアドレスを持ちます (文字列値それぞれが 4 Word を使用します):
CP1_95 (カスタムパケット 1 の Word 95 - マッピング先 14001 文字 1-8)
CP1_99 (カスタムパケット 1 の Word 99 - マッピング先 14001 文字 9-16)
CP1_103 (カスタムパケット 1 の Word 103 - マッピング先 14002 文字 1-8)
CP1_107 (カスタムパケット 1 の Word 107 - マッピング先 14002 文字 9-16)

Omni 生データアーカイブ

Omni フローコンピュータは、さまざまな範囲のメモリを単一のデータ構造にマッピングし、トリガされた場合にその構造をアーカイブに格納するよう設定できます。ユーザーは最大 10 個のアーカイブを設定できます。アラームデータと監査データ用に固定フォーマットのアーカイブがさらに 2 つあります。各アーカイブは循環バッファであり、新しいレコードが作成されるたびに最も古いレコードと入れ替わります。

レコードの構成と取得

ユーザーは生データアーカイブ 1 から 10 のレコード構造を設定できます。アーカイブ 11 と 12 は固定フォーマットであり、それぞれアラームデータと監査データが格納されます。
生データアーカイブの詳細については、Omni Technical Bulletin 96073 を参照してください。

各レコードにデータポイントのグループを 16 個まで含めることができます。各グループはその開始インデックスとデータポイント数によって定義されます。アーカイブレコードの定義に使用されるアドレスを以下に示します。レコードの合計サイズが 250 バイトを超えてはなりません。このデバイスでは日時スタンプデータに最初の 6 バイトが使用され、244 バイトは生データ用に残されます。各レコードには独自のブールトリガがあります。このトリガが低から高に変化するとデータが格納されます。

グループの開始インデックス、グループ内のポイント数、または生データアーカイブのトリガを変更する前に、アーカイブを停止する必要があります。デバイスで「**Allow Archive Configuration Flag**」が設定されている必要があります。これを行うとデバイス内のデータアーカイブ (すべての生データアーカイブやテキストアーカイブを含む) が再初期化される可能性があるので注意してください。

13920 アーカイブ実行 - 0 = 停止、1 = 開始
13921 アーカイブの再設定 - 0 = 設定変更を許可しない、1 = 設定変更を許可

このドライバーを使用して、生データアーカイブのレコードを一度に 1 つずつ読み取ることができます。レコードを読み取るには、最初に必要なレコードインデックスを "要求されたレコード" レジスタに書き込みます。この値が設定されると、ユーザーは "RA" タグを使用してそのレコードを読み取れるようになります。指定したレコードインデックスがそのアーカイブで許可されるレコードの最大数を超えないようにする必要があります。"更新された最後のレコード" の値がゼロの場合、最後に初期化されてからそのアーカイブにはレコードが保存されていません。

生データアーカイブ 1 (アドレス 701)
13500 グループ 1 - 開始インデックス
13501 グループ 1 - ポイント数
から
13530 グループ 16 - 開始インデックス
13531 グループ 16 - ポイント数

13900 ブールトリガ

3701 最大レコード数
3702 更新された最後のレコード

3703 要求されたレコード

生データアーカイブ 2 (アドレス 702)
13540 グループ 1 - 開始インデックス
13541 グループ 1 - ポイント数
から
13570 グループ 16 - 開始インデックス
13571 グループ 16 - ポイント数

13901 ブールトリガ

3704 最大レコード数
3705 更新された最後のレコード
3706 要求されたレコード

生データアーカイブ 3 (アドレス 703)
13580 グループ 1 - 開始インデックス
13581 グループ 1 - ポイント数
から
13610 グループ 16 - 開始インデックス
13611 グループ 16 - ポイント数

13902 ブールトリガ

3707 最大レコード数
3708 更新された最後のレコード
3709 要求されたレコード

生データアーカイブ 4 (アドレス 704)
13620 グループ 1 - 開始インデックス
13621 グループ 1 - ポイント数
から
13650 グループ 16 - 開始インデックス
13651 グループ 16 - ポイント数

13903 ブールトリガ

3710 最大レコード数
3711 更新された最後のレコード
3712 要求されたレコード

生データアーカイブ 5 (アドレス 705)
13660 グループ 1 - 開始インデックス
13661 グループ 1 - ポイント数
から
13690 グループ 16 - 開始インデックス
13691 グループ 16 - ポイント数

13904 ブールトリガ

3713 最大レコード数
3714 更新された最後のレコード
3715 要求されたレコード

生データアーカイブ 6 (アドレス 706)
13700 グループ 1 - 開始インデックス
13701 グループ 1 - ポイント数
から
13730 グループ 16 - 開始インデックス
13731 グループ 16 - ポイント数

13905 ブールトリガ

3716 最大レコード数
3717 更新された最後のレコード
3718 要求されたレコード

生データアーカイブ 7 (アドレス 707)
13740 グループ 1 - 開始インデックス
13741 グループ 1 - ポイント数
から
13770 グループ 16 - 開始インデックス
13771 グループ 16 - ポイント数

13906 ブールトリガ

3719 最大レコード数
3720 更新された最後のレコード
3721 要求されたレコード

生データアーカイブ 8 (アドレス 708)
13780 グループ 1 - 開始インデックス
13781 グループ 1 - ポイント数
から
13810 グループ 16 - 開始インデックス
13811 グループ 16 - ポイント数

13907 ブールトリガ

3722 最大レコード数
3723 更新された最後のレコード
3724 要求されたレコード

生データアーカイブ 9 (アドレス 709)
13820 グループ 1 - 開始インデックス
13821 グループ 1 - ポイント数
から
13850 グループ 16 - 開始インデックス
13851 グループ 16 - ポイント数

13908 ブールトリガ

3725 最大レコード数
3726 更新された最後のレコード
3727 要求されたレコード

生データアーカイブ 10 (アドレス 710)
13860 グループ 1 - 開始インデックス

13861 グループ 1 - ポイント数
から
13890 グループ 16 - 開始インデックス
13891 グループ 16 - ポイント数

13909 ブールトリガ

3728 最大レコード数
3729 更新された最後のレコード
3730 要求されたレコード

生データアーカイブ 11 (アドレス 711)
設定不可

3731 最大レコード数
3732 更新された最後のレコード
3733 要求されたレコード

生データアーカイブ 12 (アドレス 712)
設定不可

3734 最大レコード数
3735 更新された最後のレコード
3736 要求されたレコード

注記: データはデバイスから 16 ビットレジスタとして返されます。デジタル I/O は 16 ビットのブロックでマッピングされる必要があります。

生データアーカイブのアドレス構文

生データアーカイブレコード内の指定したオフセットにあるデータにアクセスするためにタグを作成できます。アドレス構文は次のとおりです。デフォルトのデータ型を**太字**で示しています。

アドレス	範囲	データ型	アクセス
RAn_o	n = アーカイブ番号 (1-12) o = Word オフセット (0-125)	Word、**Short**、BCD、DWord、Long、LBCD、Float、String	読み取り専用
RAn_o.b	n = アーカイブ番号 (1-12) o = Word オフセット (0-125) b = ビット番号 (0/1-15/16)*	**Boolean**	読み取り専用

詳細については、設定の「ゼロベースのビットアドレス指定」を参照してください。

注記:

1. 8 文字 ASCII 文字列データのみがサポートされます。
2. グループ構成に 16 文字 ASCII 文字列データアドレスが含まれている場合、データを 2 つの 8 文字 ASCII 文字列データアイテムとして読み取ることができます。

タイムスタンプのフォーマット

各レコードの最初の 6 バイトには、そのレコードがアーカイブに配置された日時が含まれています。

バイト	説明
1	月 (1-12)* 日 (1-31)
2	日 (1-31)* 月 (1-12)
3	年 (0-99)
4	時 (0-23)
5	分 (0-59)
6	秒 (0-59)

*日付フォーマットはレジスタ 3842 (0=dd/mm/yy、1= mm/dd/yy) によって設定します。

アラーム/イベントログレコードの構造 (アドレス 711)

フィールド	データ型	説明
1	3 バイトの日付	dd/mm/yy または mm/dd/yy。
2	3 バイトの時刻	hh/mm/ss。
3	16 ビット整数	アラームまたはイベントの Modbus インデックス番号。
4	1 バイト	アラームのタイプ。
5	1 バイト	0 = OK、1 = アラーム。
6	IEEE 単精度実数	アラームまたはイベント発生時のトランスデューサ変数の値。
7	32 ビット整数	アラームまたはイベント発生時の容積積算計。
8	32 ビット整数	アラームまたはイベント発生時の質量積算計。

アラームのタイプ

タイプ	説明
0	フィールド番号 3 によって示されるビットでエッジ変化があったことをイベントログに記録し、ビープ音で警告し、LCD に表示します。
1	フィールド番号 3 によって示されるビットで立ち上がりエッジ変化があったことをイベントログに記録し、ビープ音で警告し、LCD に表示します。
2	フィールド番号 3 によって示されるビットでエッジ変化があったことをイベントログに記録します。ビープ音とLCD 表示はありません。
3	フィールド番号 3 によって示されるビットで立ち上がりエッジ変化があったことをイベントログに記録します。ビープ音とLCD 表示はありません。

監査イベントログレコードの構造 (アドレス 712)

フィールド	データ型	説明
1	3 バイトの日付	dd/mm/yy または mm/dd/yy。
2	3 バイトの時刻	hh/mm/ss。
3	16 ビット整数	イベント番号、イベントごとに増分、65535 で最初に戻る。
4	16 ビット整数	変更された変数の Modbus インデックス。

5	IEEE 単精度実数	数値変数値の変更前の古い値。
6	IEEE 単精度実数	数値変数値の変更後の新しい値。
7	16 文字 ASCII	文字列変数値の変更前の古い値。
8	16 文字 ASCII	文字列変数値の変更後の新しい値。
9	32 ビット整数	変更時の容積積算計。
10	32 ビット整数	変更時の質量積算計。

注記: 変更された変数のタイプが文字列である場合、フィールド 5 と6 は 0.0 に設定されます。変更された変数のタイプが文字列でない場合、フィールド 7 と8 には Null 文字が含まれます。フィールド 7 と8 に8 文字の文字列が含まれている場合、残りの 8 文字は Null でパディングされます。

Omni テキストレポート

Omni フローコンピュータはさまざまなタイプのテキストレポートを生成できます。このドライバーによってこれらの各レポートを読み取って、OPC クライアントに文字列値として送信できます。

テキストレポートのタイプ

Omni フローコンピュータからは数多くのタイプのレポートを読み込むことができます。これらは "TR" タグを使用して読み取ることができます。レポートのタイプは次のとおりです。

カスタムレポートテンプレート
9001 レポートテンプレート - スナップショット/一定間隔
9002 レポートテンプレート - バッチ
9003 レポートテンプレート - 日次
9004 レポートテンプレート - 較正

以前のバッチレポート
9101 バッチレポート - 最後
9102 バッチレポート - 最後から 2 番目
...
9108 バッチレポート - 最後から 8 番目

以前の較正レポート
9201 較正レポート - 最後
9202 較正レポート - 最後から 2 番目
...
9208 較正レポート - 最後から 8 番目

以前の日次レポート
9301 以前の日次レポート - 最後
9302 以前の日次レポート - 最後から 2 番目
...
9308 以前の日次レポート - 最後から 8 番目

最終スナップショットレポート
9401 最終スナップショット/一定間隔レポート

その他のレポート用バッファ
9402 その他のレポート用バッファ

月次レポートのプレビュー

9501 前月のレポート - 最後
9502 前月のレポート - 最後から 2 番目
...
9508 前月のレポート - 最後から 8 番目

テキストレポートのアドレス構文

アドレス	範囲	データ型	アクセス
TRn TRn T (トリガによる読み取り)	n = レポートのアドレス (9001-9508)	String	読み取り/書き込み

例

スナップショットレポートテンプレート (アドレス 9001) との間で読み書きを行うため、アドレスが "TR9001" であるタグを作成します。

注記: テキストレポートの読み取りには数秒かかるので、"TR" タグを OPC クライアントで非アクティブなままにする必要があります。あるいは、トリガによる読み取りを代わりに使用できます。ドライバーがテキストレポートを読み書きしている間はチャネル上でその他のタグを読み書きできません。

トリガによるテキストレポートの読み取り

前述のように、必ずしも可能なわけではありませんが、テキストレポートタグは非アクティブなままにすることをお勧めします。テキストレポートタグをアクティブなままにするための代替手段として、「トリガによる読み取り」機能が追加されました。この機能は、補助トリガタグで実際のデバイス読み取りを行うタイミングを制御します。

トリガによる読み取りは、テキストレポートタグの更新サイクルでいつトリガが設定されるかによるため、ただちに開始しないことがあります。読み取りが完了すると、ドライバーはトリガの状態をクリアします。テキストレポートタグに、最後にトリガされた読み取りの結果である値とデータの品質が示されます。

テキストレポートの読み取りトリガのアドレス構文

アドレス	範囲	データ型	アクセス
TRIG_TRn	n = レポートのアドレス (9001-9508)	Boolean	読み取り/書き込み

例

トリガによって最終バッチレポート (アドレス 9101) を読み取るため、2 つのタグを作成します。1 つ目はアドレスが "TR9101 T" であるテキストレポートタグ、2 つ目はアドレスが "TRIG_TR9101" であるテキストレポート読み取りトリガタグです。

注記: テキストレポートタグのアドレスは、通常のテキストレポートのアドレスに、スペースと、"トリガによる読み取り" を表す "T" という文字を付けたものになります。トリガによる読み取りが機能するためには、アドレスにこの "T" が含まれている必要があります。

読み取りをトリガするには、トリガタグの値を true (ゼロ以外の値) に設定します。読み取りが完了すると、ドライバーはトリガの値を false (0) に設定します。読み取りに成功した場合、テキストレポートタグのデータ品質は良好になります。読み取りに失敗した場合、テキストレポートタグのデータ品質は不良になり、その値は最後に正常に読み取られた値になります。

ディスクへのテキストレポートデータの保存

ドライバーはテキストレポートデータをディスクに保存できます。この機能はテキストレポートパスタグを使用して有効にします。これらのタグを使用してファイルパス文字列をドライバーのメモリに書き込みます。各タイプのレポートには独自のパスバッファがあります。テキストレポートの読み取りに成功した後、ドライバーは関連するパスバッファをチェックします。そこに有効なパスが格納されている場合、ドライバーはそのファイルにレポートデータを ASCII テキストとして保存します。必要に応じてファイルが作成されます。以降のテキストレポート読み取りではこのファイルが上書きされます。

パスバッファはサーバーの起動時に空の文字列に初期化されます。関連するパスバッファに有効なパスが保存されるまで、ドライバーはテキストレポートデータをファイルに書き込みません。パスデータは永続的なものではありません。サーバーが再起動するたびにパス文字列を再書き込みする必要があります。パスの値はいつでも変更できるので、必要な場合には、読み取りのたびにデータを別のファイルに保存できます。

パス文字列の長さは最大 255 文字です。

テキストレポートパスのアドレス構文

アドレス	範囲	データ型	アクセス
PATH_TRn	n = レポートのアドレス (9001-9508)	String	読み取り/書き込み

例

最終バッチレポート (アドレス 9101) を読み取って結果をディスクに保存するため、2 つのタグを作成します。1 つ目はアドレスが "TR9101" であるテキストレポートタグ、2 つ目はアドレスが "PATH_TR9101" であるパスタグです。

(フォルダ "C:\OmniData\BatchReports" に作成される) "LastBatch.txt" という名前のファイルにレポートデータを保存するため、最初に "C:\OmniData\BatchReports\LastBatch.txt" をこのパスタグに書き込むようクライアントを設定します。これが完了した後、テキストレポートタグを読み取ります。テキストレポートの 1 回目の読み取りの前にパスが設定されていない場合、ドライバーはデータをディスクに保存できません。

注記: この機能を無効にするには、パスタグに空の文字列を書き込みます。

Omni テキストアーカイブ

Omni フローコンピュータはアーカイブにレポートも格納できます。このドライバーはアーカイブから一定範囲のレポートを読み取って、これを OPC クライアントに文字列値として送信できます。

テキストアーカイブの読み取り

テキストアーカイブを読み取る前に、デバイスでアーカイブ開始日と読み込み日数の 2 つの設定を行う必要があります。これらの 32 ビット整数値はそれぞれアドレス 15128 と 15127 にあります。日付フォーマットはアドレス 3842 にある値 (0 = dd/mm/yy、1 = mm/dd/yy) を使用して指定できます。日数を設定するとただちに、デバイスはデータの準備を開始します。データが読み取り可能になると、日数の値が負になります。テキストアーカイブは日数を設定した後でいつでも読み取ることができます。ドライバーはこの値が負になるのを待ちます。

テキストアーカイブのアドレス構文

アドレス	範囲	データ型	アクセス
TA TA T (トリガによる読み取り)	なし	String	読み取り専用

注記: テキストアーカイブの読み取りには数分かかるので、"TA" タグを OPC クライアントで非アクティブなままにする必要があります。あるいは、トリガによる読み取りを代わりに使用できます。サーバーで同期読み取りの最大タイムアウトを一般的なテキストアーカイブ要求を読み取るのに十分な値まで増やすことはできないので、このタグは必ず非同期読み取りを使用して読み取る必要があります。テキストアーカイブの読み取り中はチャネル上でその他のタグを読み書きできません。

テキストアーカイブの読み取りが途中で失敗した場合、ユーザーは日数レジスタ (15127) に 999 を書き込むことでデバイスの読み取りバッファをリセットしてから、通常のテキストアーカイブ読み取りプロシージャを繰り返す必要があります。これを行わない場合、ドライバーはアーカイブの要求された範囲の最初の部分を取得できません。

トリガによるテキストアーカイブの読み取り

必ずしも可能なわけではありませんが、テキストアーカイブタグは非アクティブなままにすることをお勧めします。テキストアーカイブタグをアクティブなままにするための代替手段として、「トリガによる読み取り」機能が追加されました。この機能は、補助トリガタグで実際のデバイス読み取りを行うタイミングを制御します。トリガ値はドライバーのメモリに格納され、以下で説明するアドレス構文のタグを使用して読み取りと設定を行うことができます。

トリガによる読み取りは、テキストアーカイブタグの更新サイクルでいつトリガが設定されるかによるため、ただちに開始しないことがあります。読み取りが完了すると、ドライバーはトリガの状態をクリアします。テキストアーカイブタグに、最後にトリガされた読み取りの結果である値とデータの品質が示されます。

テキストアーカイブの読み取りトリガのアドレス構文

アドレス	範囲	データ型	アクセス
TRIG_TA	なし	Boolean	読み取り/書き込み

例

トリガによってテキストアーカイブを読み取るため、2 つのタグを作成します。1 つ目はアドレスが "TA T" であるテキストアーカイブタグ、2 つ目はアドレスが "TRIG_TA" であるテキストアーカイブ読み取りトリガタグです。ユーザーは開始日タグと日数タグを作成する必要があります。

注記: テキストアーカイブタグのアドレスは、通常のテキストアーカイブのアドレスに、スペースと、"トリガによる読み取り" を表す "T" という文字を付けたものになります。トリガによる読み取りが機能するためには、アドレスにこの "T" が含まれている必要があります。

読み取りをトリガするには、トリガタグの値を true (ゼロ以外の値) に設定します。読み取りが完了すると、ドライバーはトリガの値を false (0) に設定します。読み取りに成功した場合、テキストアーカイブタグのデータ品質は良好になります。読み取りに失敗した場合、テキストアーカイブタグのデータ品質は不良になり、その値は最後に正常に読み取られた値になります。

ディスクへのテキストアーカイブデータの保存

ドライバーはテキストアーカイブデータをディスクに保存できます。この機能はテキストアーカイブパスタグを使用して有効にします。このタグを使用してファイルパス文字列をドライバーのメモリに書き込みます。テキストアーカイブの読み取りに成功した後、ドライバーは関連するパスバッファをチェックします。そこに有効なパスが格納されている場合、ドライバーはそのファイルにテキストアーカイブデータを ASCII テキストとして保存します。必要に応じてファイルが作成されます。以降のテキストアーカイブ読み取りではこのファイルが上書きされます。

パスバッファはサーバーの起動時に空の文字列に初期化されます。関連するパスバッファに有効なパスが保存されるまで、ドライバーはテキストアーカイブデータをファイルに書き込みません。パスデータは永続的なものではありません。ユーザーはサーバーが再起動するたびにパス文字列を再書き込みする必要があります。パスの値はいつでも変更できるので、(必要な場合には) 読み取りのたびにデータを別のファイルに保存できます。

パス文字列の長さは最大 255 文字です。ディレクトリは、ルートドライブ (C:\\TextArchive.txt)、Windows ディレクトリ内、またはサーバーのインストール場所以外の Program Files ディレクトリ内にすることはできません。ファイル拡張子には .txt または .log を使用する必要があります。

テキストアーカイブパスのアドレス構文

アドレス	範囲	データ型	アクセス
PATH_TA	なし	String	読み取り/書き込み

例

テキストアーカイブを読み取って結果をディスクに保存するため、2 つのタグを作成します。1 つ目はアドレスが "TA" であるテキストアーカイブタグ、2 つ目はアドレスが "PATH_TA" であるパスタグです。ユーザーは上記の説明に従って開始日タグと日数タグを作成する必要があります。

(フォルダ "C:\OmniData\ArchiveData" に作成される) "TextArchive.txt" という名前のファイルにテキストアーカイブデータを保存するため、最初に "C:\OmniData\ArchiveData\TextArchive.txt" をこのパスタグに書き込むようクライアントを設定します。これが完了した後、テキストアーカイブタグを読み取ります。テキストアーカイブの 1 回目の読み取りの前にパスが設定されていない場合、ドライバーはデータをディスクに保存できません。

注記: この機能を無効にするには、パスタグに空の文字列を書き込みます。

ファンクションコードの説明

Modbus アドレス指定モデル

10 進	16 進	説明
01	0x01	コイルのステータスを読み取り
02	0x02	入力ステータスを読み取り
03	0x03	保持レジスタを読み取り
04	0x04	内部レジスタを読み取り
05	0x05	単一コイルを適用
06	0x06	単一レジスタをプリセット
15	0x0F	複数コイルを適用
16	0x10	複数レジスタをプリセット
22	0x16	レジスタへのマスク書き込み

Daniels S500 フローコンピュータアドレス指定モデル

10 進	16 進	説明
01	0x01	コイルのステータスを読み取り
02	0x02	入力コイルを読み取り
03	0x03	保持レジスタを読み取り
05	0x05	単一コイルを適用
16	0x10	複数レジスタをプリセット

Configuration API Service - チャネルのプロパティ

以下のプロパティは、Configuration API Service を使用してチャネルを定義します。

一般プロパティ

common.ALLTYPES_NAME * 必須パラメータ

注記: このプロパティを変更すると、API エンドポイント URL が変更されます。

common.ALLTYPES_DESCRIPTION

サーバーメイン。MULTIPLE_TYPES_DEVICE_DRIVER * 必須パラメータ

サーバーメイン。CHANNEL_DIAGNOSTICS_CAPTURE

イーサネット通信プロパティ

サーバーメイン。CHANNEL_ETHERNET_COMMUNICATIONS_NETWORK_ADAPTER_STRING

詳細プロパティ

servermain.CHANNEL_NON_NORMALIZED_FLOATING_POINT_HANDLING

書き込み最適化

サーバーメイン。CHANNEL_WRITE_OPTIMIZATIONS_METHOD

サーバーメイン。CHANNEL_WRITE_OPTIMIZATIONS_DUTY_CYCLE

関連項目: サーバーヘルプシステムの Configuration API Service のセクション

Configuration API Services - デバイスのプロパティ

以下のプロパティは Configuration API Service を使用してチャネルを定義します。

一般プロパティ

common.ALLTYPES_NAME * 必須パラメータ

common.ALLTYPES_DESCRIPTION

servermain.DEVICE_CHANNEL_ASSIGNMENT

servermain.MULTIPLE_TYPES_DEVICE_DRIVER * 必須パラメータ

servermain.DEVICE_MODEL * 必須パラメータではありませんが、デフォルト値で問題ないかどうかを確認してください

servermain.DEVICE_ID_STRING * 必須パラメータ

servermain.DEVICE_DATA_COLLECTION

servermain.DEVICE_SIMULATED

スキャンモード

```
servermain.DEVICE_SCAN_MODE

servermain.DEVICE_SCAN_MODE_RATE_MS

servermain.DEVICE_SCAN_MODE_RATE_MS

servermain.DEVICE_SCAN_MODE_PROVIDE_INITIAL_UPDATES_FROM_CACHE
```

自動格下げ

```
servermain.DEVICE_AUTO_DEMOTION_ENABLE_ON_COMMUNICATIONS_FAILURES

servermain.DEVICE_AUTO_DEMOTION_DEMOTE_AFTER_SUCCESSIVE_TIMEOUTS

servermain.DEVICE_AUTO_DEMOTION_PERIOD_MS

servermain.DEVICE_AUTO_DEMOTION_DISCARD_WRITES
```

タグ生成

```
servermain.DEVICE_TAG_GENERATION_ON_STARTUP

servermain.DEVICE_TAG_GENERATION_DUPLICATE_HANDLING

servermain.DEVICE_TAG_GENERATION_GROUP

servermain.DEVICE_TAG_GENERATION_ALLOW_SUB_GROUPS
```

ヒント: 自動タグ生成を起動するには、空のボディを持つ PUT をデバイス上の taggeneration サービスエンドポイントに送信します。
関連項目: 詳細については、サービスのヘルプを参照してください。

タイミング

```
servermain.DEVICE_CONNECTION_TIMEOUT_SECONDS

servermain.DEVICE_REQUEST_TIMEOUT_MILLISECONDS

servermain.DEVICE_RETRY_ATTEMPTS

servermain.DEVICE_INTER_REQUEST_DELAY_MILLISECONDS
```

関連項目: サーバーヘルプシステムの Configuration API Service のセクション

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。

ヒント: データソース (データベースをはじめとするサードパーティ製ソフトウェアなど) から生成されたメッセージは、イベントログ経由で表示されます。トラブルシューティングを実行するには、オンラインとベンダーのドキュメントでこれらのメッセージを調べる必要があります。

ブロック範囲に不良アドレスがあります。| アドレス範囲 = <開始> ～ <終了>。

エラータイプ:

エラー

考えられる原因:

1. 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ました。
2. プロトコルによって許可されている数よりも多くのレジスタを読み込もうとしました。

解決策:

1. デバイスの指定された範囲のアドレスに割り当てられたタグを確認し、無効な位置を参照するタグを削除してください。
2. レジスタのブロックサイズの値を 125 に減らします。

関連項目:

1. エラー処理
2. ブロックサイズ

不良配列。| 配列範囲 = <開始> ～ <終了>。

エラータイプ:

エラー

考えられる原因:

アドレス空間の末端を超えてアドレスの配列が定義されています。

解決策:

1. デバイスのメモリ空間のサイズを確認し、配列長を適切に再定義してください。
2. デバイスの指定された範囲のアドレスに割り当てられたタグを確認し、無効な位置を参照するタグを削除してください。
3. 配列サイズの値を 125 に減らします。

関連項目:

1. エラー処理
2. ブロックサイズ

ブロックアドレスが例外コードを返しました。| アドレス範囲 = <開始> ～ <終了>、例外コード = <コード>。

エラータイプ:
警告

考えられる原因:
例外コードの説明については、「Modbus 例外コード」を参照してください。

解決策:
「Modbus 例外コード」を参照してください。

アドレスに書き込めません。デバイスは例外コードを返しました。| アドレス = '<アドレス>'、例外コード = <コード>。

エラータイプ:
警告

考えられる原因:
例外コードの説明については、「Modbus 例外コード」を参照してください。

解決策:
「Modbus 例外コード」を参照してください。

アドレスから読み取れません。デバイスは例外コードを返しました。| アドレス = '<アドレス>'、例外コード = <コード>。

エラータイプ:
警告

考えられる原因:
例外コードの説明については、「Modbus 例外コード」を参照してください。

解決策:
「Modbus 例外コード」を参照してください。

メモリリソース量の低下によりタグインポートが失敗しました。

エラータイプ:
警告

考えられる原因:
ドライバーは変数インポートファイルの処理に必要なメモリを割り当てることができません。

解決策:
不要なアプリケーションをすべて終了してから、再試行してください。

タグのインポート中にファイル例外が発生しました。

エラータイプ:
警告

考えられる原因:
変数インポートファイルを読み取れませんでした。

解決策:
変数インポートファイルを再生成してください。

インポートファイルのレコードの解析でエラーが発生しました。| レコード番号 = <数値>、フィールド = <名前>。

エラータイプ:
警告

考えられる原因:
変数インポートファイルの指定されたフィールドが無効であるか予想より長いため、解析できませんでした。

解決策:
変数インポートファイルを編集して、問題のあるフィールドを変更してください。

インポートファイルのレコードの説明が切り詰められました。| レコード番号 = <数値>。

エラータイプ:
警告

考えられる原因:
指定されたレコードのタグの説明が長すぎます。

解決策:
ドライバーは必要に応じて説明を切り詰めます。このエラーを防止するには、変数インポートファイルを編集して、説明を短くしてください。

インポートされたタグ名が無効のため変更されました。| タグ名 = '<タグ>'、変更後のタグ名 = '<タグ>'。

エラータイプ:
警告

考えられる原因:
変数インポートファイル内のタグ名に無効な文字が含まれていました。

解決策:
ドライバーは変数インポートファイルに基づいて有効な名前を構築します。このエラーを防止し、名前の一貫性を維持するには、エクスポートされた変数の名前を変更してください。

データ型がサポートされていないため、タグをインポートできませんでした。| タグ名 = '<タグ>'、サポートされていないデータ型 = '<タイプ>'。

エラータイプ:
警告

考えられる原因:
変数インポートファイルで指定されているデータ型はこのドライバーによってサポートされていません。

解決策:
変数インポートファイルで指定されているデータ型を、サポートされているいずれかの型に変更してください。構造体の変数である場合、ファイルを手動で編集して構造体に必要な各タグを定義するか、サーバーで必要なタグを手動で設定してください。

関連項目:
Concept からの変数のエクスポート

メモリ割り当ての問題により、Omni テキストバッファを読み取れませんでした。

エラータイプ:
警告

考えられる原因:
ドライバーは Omni テキストレコードまたはテキストアーカイブの読み取り操作に必要なメモリを割り当てることができません。

解決策:
不要なアプリケーションをすべて終了してから、再試行してください。

指定された日付範囲では使用可能な Omni テキストアーカイブデータがありません。

エラータイプ:
警告

考えられる原因:
開始日レジスタ (15128) と日数レジスタ (15127) によって指定されている日付範囲のテキストアーカイブにはデータがありません。

解決策:
これは必ずしもエラーではありません。指定された範囲で使用可能なデータがないことを確認してください。

Omni テキストレポートへの書き込みが切り詰められました。| レポート番号 = <数値>。

エラータイプ:
警告

考えられる原因:
テキストレポートに 8192 バイトを超えて書き込もうとしました。これはプロトコルによって課されている制限です。

解決策:

8192 バイトの制限を越える文字列を書き込まないでください。文字列がこれより長い場合、最初の 8192 文字だけがデバイスに書き込まれます。

パケット数の制限により、Omni テキストレポートを読み取れませんでした。| レポート番号 = <数値>。

エラータイプ:
警告

考えられる原因:
テキストレポートは 8192 バイト以下でなければなりません。これはプロトコルによって課せられている制限です。ドライバーは予想されるファイル終端文字が出現する前に 8192 バイトを読み取りました。

解決策:
デバイスによって使用されるレポートテンプレートでは 8192 バイト以下のレポートが生成されることを確認してください。

書き込みに失敗しました。パスの最大長を超えています。| タグアドレス = '<アドレス>'、最大長 = <数値>。

エラータイプ:
警告

考えられる原因:
パスの長さは示されている文字数に制限されています。

解決策:
短いパスを使用してください。

Omni テキストデータをファイルに書き込んでいるときにエラーが発生しました。| タグアドレス = '<アドレス>'、理由 = '<理由>'。

エラータイプ:
警告

考えられる原因:
示された理由により、ドライバーは Omni テキストデータをディスクに書き込めませんでした。

解決策:
示された理由に適した対処法については、オペレーティングシステムのドキュメントを参照してください。

Omni テキスト出力ファイルを開くことができませんでした。| タグアドレス = '<アドレス>'、理由 = '<理由>'。

エラータイプ:
警告

考えられる原因:
Omni テキストパスタグで指定されているファイルを作成するか開くことができませんでした。

解決策:

示された理由に適した対処法については、オペレーティングシステムのドキュメントを参照してください。パスが無効であることが最も考えられる原因です。

関連項目:

1. Omni テキストレポート
2. Omni テキストアーカイブ

アドレスに書き込めません。予期しない文字が応答に含まれています。| タグアドレス = '<アドレス>'。

エラータイプ:
警告

アドレスから読み取れません。予期しない文字が応答に含まれています。| タグアドレス = '<アドレス>'。

エラータイプ:
警告

ブロックのアドレスを読み取れません。予期しない文字が応答に含まれています。| アドレス範囲 = <開始> ～ <終了>。

エラータイプ:
警告

Omni テキスト出力ファイルを変更できませんでした。 | タグアドレス = '<アドレス>', 理由 = 指定したパスは許可されません。

エラータイプ:
警告

考えられる原因:

Omni テキストパスタグで指定されているディレクトリは許可されません。

解決策:

ドライバーヘルプの内容を参照し、安全な場所へのパスを指定してください。

関連項目:

1. Omni テキストレポート
2. Omni テキストアーカイブ

Omni テキスト出力ファイルを変更できませんでした。 | タグアドレス = '<アドレス>', 理由 = 指定したファイルの拡張子は '.txt' または '.log' でなければなりません。

エラータイプ:
警告

考えられる原因:

Omni テキストパスタグで指定されている拡張子は許可されません。.txt または .log でなければなりません。

解決策:

有効な拡張子を指定してください。

関連項目:

1. Omni テキストレポート
2. Omni テキストアーカイブ

ファイルからタグデータベースをインポートしています。| ファイル名 = '<名前>'。

エラータイプ:

情報

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

Modbus 例外コード

以下のデータは Modbus Application Protocol Specifications ドキュメントからのものです。

コード 10 進/16 進	名前	意味
01/0x01	ILLEGAL FUNCTION (不正なファンクション)	クエリーで受信したファンクションコードを、サーバーに対する操作として実行することはできません。このファンクションコードは新しいデバイスにだけ適用できるか、選択したユニットに実装されていないことが原因である可能性があります。また、サーバーが、このタイプの要求を処理する状態になっていない可能性もあります (レジスタ値を返す必要があるにもかかわらず、サーバーがそのように設定されていない場合など)。
02/0x02	ILLEGAL DATA ADDRESS (不正なデータアドレス)	クエリーで受信したデータアドレスを、サーバーに対するアドレスとして使用することはできません。具体的には、参照番号と転送長さの組み合わせが無効です。レジスタが 100 個あるコントローラの場合、オフセット 96 と長さ 4 の要求では成功します。オフセット 96 と長さ 5 の要求では例外 02 が生成されます。
03/0x03	ILLEGAL DATA VALUE (不正なデータ値)	クエリーデータフィールドに含まれている値を、サーバーに対する値として使用することはできません。これは、示された長さが正しくないなど、複合型要求の残りの構造体に誤りがあることを示しています。Modbus プロトコルでは個々のレジスタのそれぞれの値の有意性は認識されないので、これはレジスタのストレージにサブミットされたデータアイテムの値がアプリケーションプログラムでの予想の範囲外であることを必ずしも意味しません。
04/0x04	SERVER DEVICE FAILURE (サーバーデバイスの障害)	サーバーが要求された操作を実行しようとしたときに回復不可能なエラーが発生しました。
05/0x05	ACKNOWLEDGE	サーバーは要求を受け入れて処理していますが、処理が完了するまで時間がかかります。クライアントでタイムアウトエラーが発生しないようにするために、この応答が返されます。クライアントは次にプログラム完了ポーリングメッセージを送信して、処理が完了したかどうかを判断します。
06/0x06	SERVER DEVICE BUSY (サーバーデバイスがビジー状態)	サーバーは、時間がかかるプログラムコマンドを処理しています。クライアントは、サーバーによる処理の完了後にメッセージを再送信する必要があります。
07/0x07	NEGATIVE ACKNOWLEDGE (否定応答)	サーバーは、クエリーで受信したプログラムファンクションを実行できません。このコードはファンクションコード 13 または 14 (10 進) を使用したプログラミング要求が成功しなかった場合に返されます。クライアントは、サーバーに対して診断情報またはエラー情報を要求する必要があります。
08/0x08	MEMORY PARITY ERROR (メモリパリティエラー)	サーバーが拡張メモリを読み取ろうとしたときに、メモリ内でパリティエラーが検出されました。クライアントはこの要求を再試行できますが、サーバーデバイス上でサービスが必要になる場合があります。
10/0x0A	GATEWAY PATH UNAVAILABLE (ゲートウェイパスを使用できません)	ゲートウェイが使用されている場合、ゲートウェイが要求を処理するために入力ポートから出力ポートへの内部通信パスを割り当てることができなかったことを示します。これは通常、ゲートウェイの設定に誤りがあるかオーバーロードされていることを意味します。
11/0x0B	GATEWAY TARGET	ゲートウェイが使用されている場合、ターゲットデバイスから応答がな

コード 10 進/16 進	名前	意味
	DEVICE FAILED TO RESPOND (ゲートウェイのターゲットデバイスが応答しませんでした)	かったことを示します。これは通常、デバイスがネットワーク上に存在しないことを意味します。

注記: このドライバーでは、「サーバー」と「非送信請求」という用語は同じ意味になります。

索引

B

BCD 32
Boolean 32

C

COM ID 10
COM ポート 10

D

Daniels S500 フローコンピュータ 6
Daniels S500 フローコンピュータのアドレス指定 38
Double 32
DTR 11
DWord 32

E

Elliott フローコンピュータ 6
Elliott フローコンピュータのアドレス指定 37

F

Float 32

I

I/O エラー 66
ID 13
IP アドレス 15

L

LBCD 32
Long 32
Long を 10 進数として扱う 21

M

Magnetek GPD 515 ドライブ 6
Magnetek GPD 515 ドライブのアドレス指定 36
Modbus のアドレス指定 33
Modbus バイトオーダー 20
Modbus 関数 05 20
Modbus 関数 06 20
Modbus 対応デバイス 6
Modbus 例外コード 67
Modicon ビットオーダー 21

O

Omni カスタムパケット 45
Omni テキストアーカイブ 55
Omni テキストデータをファイルに書き込んでいるときにエラーが発生しました。| タグアドレス = '<アドレス>'、理由 = '<理由>'。 64
Omni テキストレポート 53
Omni テキストレポートへの書き込みが切り詰められました。| レポート番号 = <数値>。 63
Omni テキスト出力ファイルを開くことができませんでした。| タグアドレス = '<アドレス>'、理由 = '<理由>'。 64
Omni テキスト出力ファイルを変更できませんでした。| タグアドレス = '<アドレス>', 理由 = 指定したパスは許可されません。 65
Omni テキスト出力ファイルを変更できませんでした。| タグアドレス = '<アドレス>', 理由 = 指定したファイルの拡張子は '.txt' または '.log' でなければなりません。 65
Omni フローコンピュータ 6
Omni フローコンピュータのアドレス指定 40
Omni 生データアーカイブ 48

R

RS-485 11
RTS 11

RX バッファオーバーラン 66

S

Short 32
String 32

T

TSXCUSBMBP USB アダプタ 6
TX バッファフル 66

W

Word 32

あ

アイドル接続を閉じる 11
アドレスから読み取れません。デバイスは例外コードを返しました。| アドレス = '<アドレス>'、例外コード = <コード>。 61
アドレスから読み取れません。予期しない文字が応答に含まれています。| タグアドレス = '<アドレス>'。 65
アドレスに書き込めません。デバイスは例外コードを返しました。| アドレス = '<アドレス>'、例外コード = <コード>。 61
アドレスに書き込めません。予期しない文字が応答に含まれています。| タグアドレス = '<アドレス>'。 65

い

イーサネットカプセル化 15
イベントログメッセージ 60
インポートされたタグ名が無効のため変更されました。| タグ名 = '<タグ>'、変更後のタグ名 = '<タグ>'。 62
インポートファイルのレコードの解析でエラーが発生しました。| レコード番号 = <数値>、フィールド = <名前>。 62
インポートファイルのレコードの説明が切り詰められました。| レコード番号 = <数値>。 62

え

エラーマスクの定義 66
エラー時に格下げ 17

エラー処理 26

お

オーバーラン 66

き

キャッシュからの初期更新 15

く

クローズするまでのアイドル時間 11
グローバル設定 8

さ

サイクルあたりのトランザクション数 8
サブグループを許可 19
サポートされるデバイス 6

し

シミュレーション 14
シリアルポートの設定 10
シリアル通信 9

す

スキャンしない、要求ポールのみ 15
スキャンモード 14
ストップビット 6, 10
すべてのタグのすべての値を書き込み 9
すべてのタグの最新の値のみを書き込み 9

せ

ゼロで置換 12
ゼロベースアドレス指定 19
ゼロベースのビットアドレス指定 20

た

ダイナミックフルードメーター (DFM) SFC3 6
ダイナミックフルードメーターのアドレス指定 38
タイムアウト前の試行回数 16
タグに指定のスキャン速度を適用 15
タグのインポート中にファイル例外が発生しました。 62
タグ数 7, 14
タグ生成 17

ち

チャネルのプロパティ - シリアル通信 9
チャネルのプロパティ - 一般 6
チャネルのプロパティ - 書き込み最適化 9
チャネルのプロパティ - 詳細 12
チャネルのプロパティ - 通信シリアル化 7
チャネルレベルの設定 8
チャネル割り当て 13

て

データエンコーディング 20
データコレクション 13
データビット 6, 10
データ型がサポートされていないため、タグをインポートできませんでした。| タグ名 = '<タグ>'、サポートされていないデータ型 = '<タイプ>'。 63
データ型の説明 32
デバイスのプロパティ - イーサネットカプセル化 15
デバイスのプロパティ - タグ生成 17
デバイスのプロパティ - 一般 12
デバイスのプロパティ - 自動格下げ 17

デバイスのプロパティ - 冗長 27
デバイス間遅延 12
デバイス起動時 18
デューティサイクル 9

と

ドライバー 13

な

なし 10

ね

ネットワーク 1 - ネットワーク 500 8
ネットワークモード 8

は

ハードウェアの破損 66
パケット数の制限により、Omni テキストレポートを読み取れませんでした。| レポート番号 = <数値>。 64
パリティ 6, 10, 66

ふ

ファイルからタグデータベースをインポートしています。| ファイル名 = '<名前>'。 66
ファンクションコードの説明 57
フレーミング 25, 66
フロー制御 10
ブロックアドレスが例外コードを返しました。| アドレス範囲 = <開始> ～ <終了>、例外コード = <コード>。 61
ブロックサイズ 23
ブロックのアドレスを読み取れません。予期しない文字が応答に含まれています。| アドレス範囲 = <開始> ～ <終了>。 65
ブロック範囲に不良アドレスがあります。| アドレス範囲 = <開始> ～ <終了>。 60
プロトコル 15
プロパティ変更時 18

ほ

ポート 15
ポーリング遅延 11
ボーレート 6, 10

め

メモリリソース量の低下によりタグインポートが失敗しました。 61
メモリ割り当ての問題により、Omni テキストバッファを読み取れませんでした。 63

も

モデム 10-11
モデム設定 11
モデル 13

れ

レジスタ、コイル、アドレスの説明 33

漢字

一般 12
仮想ネットワーク 8
概要 5
格下げまでのタイムアウト回数 17
格下げ期間 17
格下げ時に要求を破棄 17
共有 10
後続 26
最初の DWord を下位とする 21
最初の Word を下位とする 20
最適化方法 9
作成 19
削除 18
指定された日付範囲では使用可能な Omni テキストアーカイブデータがありません。 63
事前オン 11

自動ダイヤル 11
自動タグデータベース生成 28
自動格下げ 17
識別 6, 13
実行動作 11
重複タグ 18
出力コイル 23
書き込みに失敗しました。パスの最大長を超えています。| タグアドレス = '<アドレス>'、最大長 = <数値>。 64
上書き 18
冗長 27
親グループ 19
診断 7
生成 18
接続タイプ 10
接続のタイムアウト 11
設定 6, 19
説明を含める 25
先頭 26
遅延オフ 11
通信エラーを報告 11
通信タイムアウト 16
通信なしの動作 11
通信プロトコル 6
統計アイテム 29
動作モード 13
読み取り処理 12
内部レジスタ 23
入力コイル 23
反復メッセージの却下 27
非 Boolean タグの最新の値のみを書き込み 9
非正規化浮動小数点処理 12
不正なアドレスでタグを無効化 27
不良配列。| 配列範囲 = <開始> ～ <終了>。 60
負荷分散 8
物理メディア 10
文字列のブロック読み取り 24
変数のインポートファイル 25
変数のインポート設定 25
保持レジスタ 23

保持レジスタのビット書き込み 20
未修正 12
名前 13
優先順位 8
要求のタイムアウト 16
要求間遅延 16