

## SIMOTION

### モーションコントロール

## Getting Started (SIMOTION SCOUT TIA 用)

### 入門書

バージョン 5.1 以降で有効

### はじめに

基本的な安全に関する指示

1

Getting Started (SIMOTION SCOUT TIA 用)

2

コンフィグレーションの準備

3

プロジェクトを作成

4

SIMOTION デバイスの作成およびオンライン通信のコンフィグレーション

5

SIMOTION SCOUT TIA を開始

6

ターゲットシステムへのプロジェクトのダウンロード

7

ドライブのコンフィグレーション

8

電源装置のコンフィグレーション

9

軸のコンフィグレーション

10

軸コントロールパネルでの軸の試験

11

デジタル出力をコンフィグレーション

12

SIMOTION アプリケーションのプログラミング

13

実行システムのコンフィグレーション

14

システムの開始および停止

15

アプリケーションの監視

16

ESD に関する指示

A

## 法律上の注意

### 警告事項

本書には、ユーザーの安全性を確保し製品の損傷を防止するうえ守るべき注意事項が記載されています。ユーザーの安全性に関する注意事項は、安全警告サインで強調表示されています。このサインは、物的損傷に関する注意事項には表示されません。以下に表示された注意事項は、危険度によって等級分けされています。

#### 危険

回避しなければ、直接的な死または重傷に至る危険状態を示します。

#### 警告

回避しなければ、死または重傷に至るおそれのある危険な状況を示します。

#### 注意

回避しなければ、軽度または中度の人身傷害を引き起こすおそれのある危険な状況を示します。

#### 通知

回避しなければ、物的損傷を引き起こすおそれのある危険な状況を示します。

複数の危険レベルに相当する場合は、通常、最も危険度の高い事項が表示されることになっています。安全警告サイン付きの人身傷害に関する注意事項があれば、物的損傷に関する警告が付加されます。

### 有資格者

本書が対象とする製品 / システムは必ず有資格者が取り扱うものとし、各操作内容に関連するドキュメント、特に安全上の注意及び警告が遵守されなければなりません。有資格者とは、訓練内容及び経験に基づきながら当該製品 / システムの取り扱いに伴う危険性を認識し、発生し得る危害を事前に回避できる者をいいます。

### シーメンス製品を正しくお使いいただくために

以下の事項に注意してください。

#### 警告

シーメンス製品は、カタログおよび付属の技術説明書の指示に従ってお使いください。他社の製品または部品との併用は、弊社の推奨もしくは許可がある場合に限りです。製品を正しく安全にご使用いただくには、適切な運搬、保管、組み立て、据え付け、配線、始動、操作、保守を行ってください。ご使用になる場所は、許容された範囲を必ず守ってください。付属の技術説明書に記述されている指示を遵守してください。

### 商標

®マークのついた称号はすべて **Siemens AG** の商標です。本書に記載するその他の称号は商標であり、第三者が自己の目的において使用した場合、所有者の権利を侵害することになります。

### 免責事項

本書のハードウェアおよびソフトウェアに関する記述と、実際の製品内容との一致については検証済みです。しかしなお、本書の記述が実際の製品内容と異なる可能性もあり、完全な一致が保証されているわけではありません。記載内容については定期的に検証し、訂正が必要な場合は次の版で更新いたします。

# はじめに

## 本書の対象範囲および関連する標準

この文書 / 資料は、**Engineering System Handling documentation package** の一部です。

## 適用範囲

本書は SIMOTION SCOUT、製品バージョン V5.1 用です。

## SIMOTION 資料 / 文書

SIMOTION 資料は、"SIMOTION Documentation Overview document" で確認できます。

本書は、電子媒体のマニュアルとして SIMOTION SCOUT TIA の納入範囲に含まれ、10 種類の文書 / 資料パッケージで構成されます。

以下の文書 / 資料パッケージは、SIMOTION 製品バージョン V5.1 の場合に利用可能です：

- SIMOTION エンジニアリングシステムの取り扱い
- SIMOTION システムおよび機能の説明
- SIMOTION サービスおよび診断
- SIMOTION IT
- SIMOTION のプログラミング
- SIMOTION プログラミング - 参照先
- SIMOTION C
- SIMOTION P
- SIMOTION D
- SIMOTION 補足資料

## ホットラインおよびインターネットアドレス

### SIMOTION の概要

シーメンスは、ワンクリックで開くことができる、よく質問される項目についての最も重要な情報を含む **SIMOTION** に関する情報範囲から概要ページをコンパイルしました。

熟練度にかかわらず、**SIMOTION** ユーザは最も重要なダウンロード、マニュアル、説明書、FAQ、アプリケーション例を以下から入手できます：

<https://support.industry.siemens.com/cs/ww/de/view/109480700>

### 関連情報

以下のトピックに関する情報を入手するには次のリンクをクリックしてください：

- 本書の概要
- 資料をダウンロードするための他のリンク
- オンラインでの説明書の利用 (マニュアル / 情報の検索)

<https://support.industry.siemens.com/cs/ww/en/view/109479653>

### My Documentation Manager

シーメンスのコンテンツに基づいて個別にマニュアルをコンパイルする方法およびそれぞれの機械に関する資料の目的に合わせて調整する方法については、次のリンクをクリックしてください：

<https://support.industry.siemens.com/My/ww/en/documentation>

### トレーニング

SITRAIN - シーメンスのオートメーション製品、システムおよびソリューションのためのトレーニングコース - に関する情報は以下のリンクをクリックしてください：

<http://www.siemens.com/sitrain>

### FAQ (よくある質問)

FAQ は、SIMOTION SCOUT TIA に同梱される **SIMOTION Utilities & Applications** および、Service&Support ページの **"Product Support"** を参照してください：

<https://support.industry.siemens.com/cs/de/en/ps/14505/faq>

## テクニカルサポート

国別のテクニカルサポートの電話番号は、以下のインターネットサイトの **[Contact]** (コンタクト先) に記されています：

<https://support.industry.siemens.com/cs/ww/en/sc/2090>



# 目次

	はじめに.....	3
1	基本的な安全に関する指示.....	11
1.1	一般的な安全に関する指示.....	11
1.2	産業セキュリティ.....	12
1.3	脱着可能な記憶媒体を使用する際の、ソフトウェアの (不正な) 操作による生命の危険性.....	13
2	Getting Started (SIMOTION SCOUT TIA 用).....	15
2.1	Getting Started の目的.....	15
2.2	サンプルプロジェクト.....	15
2.3	要件.....	16
2.4	一般情報.....	19
2.4.1	TIA ポータル オンラインヘルプ.....	19
2.4.2	SIMOTION SCOUT TIA オンラインヘルプ.....	19
2.4.3	利用可能な資料 / 文書.....	20
2.4.4	ユーティリティ & アプリケーション.....	22
2.4.5	特別な安全情報.....	23
3	コンフィグレーションの準備.....	25
3.1	出荷時設定の復元.....	25
3.2	オンライン通信のための要件.....	27
3.3	サンプルプロジェクトでの結果.....	28
4	プロジェクトを作成.....	29
4.1	概要.....	29
4.2	新しいプロジェクトを作成.....	29
4.3	サンプルプロジェクトでの結果.....	31
5	SIMOTION デバイスの作成およびオンライン通信のコンフィグレーション.....	33
5.1	概要.....	33
5.2	SIMOTION デバイス (1) を作成.....	34
5.3	Ethernet インターフェース (2) をコンフィグレーション.....	36
5.4	PG/PC インターフェース (3) を設定.....	38
5.5	サンプルプロジェクトでの結果.....	47

<b>6</b>	<b>SIMOTION SCOUT TIA を開始.....</b>	<b>49</b>
<b>7</b>	<b>ターゲットシステムへのプロジェクトのダウンロード.....</b>	<b>51</b>
7.1	概要.....	51
7.2	プロジェクトの保存およびコンパイル.....	51
7.3	選択されたターゲットデバイスに接続 – "Go online".....	52
7.4	ターゲットシステムへのプロジェクトのダウンロード.....	56
<b>8</b>	<b>ドライブのコンフィグレーション.....</b>	<b>59</b>
8.1	概要.....	59
8.2	ドライブの自動コンフィグレーション.....	60
8.3	サンプルプロジェクトでの結果.....	63
<b>9</b>	<b>電源装置のコンフィグレーション.....</b>	<b>65</b>
9.1	概要.....	65
9.2	DRIVE-CLiQ インターフェースがない電源装置のコンフィグレーション.....	66
<b>10</b>	<b>軸のコンフィグレーション.....</b>	<b>69</b>
10.1	概要.....	69
10.2	軸の作成.....	70
10.3	軸のコンフィグレーションをターゲットシステムにダウンロード.....	73
<b>11</b>	<b>軸コントロールパネルでの軸の試験.....</b>	<b>75</b>
11.1	概要.....	75
11.2	軸コントロールパネルを使った作業.....	76
11.3	サンプルプロジェクトでの結果.....	79
<b>12</b>	<b>デジタル出力をコンフィグレーション.....</b>	<b>81</b>
<b>13</b>	<b>SIMOTION アプリケーションのプログラミング.....</b>	<b>83</b>
13.1	概要.....	83
13.2	変数.....	84
13.2.1	変数タイプ.....	84
13.2.2	サンプルプロジェクトの変数.....	84
13.2.3	グローバルデバイス変数の作成.....	85
13.2.4	I/O 変数の作成.....	86
13.2.5	このコンフィグレーションをバックアップ.....	91
13.3	プログラミング.....	91
13.3.1	サンプルプロジェクトのプログラミング言語.....	91
13.3.2	MCC モーションコントロールチャート.....	91
13.3.2.1	プログラミング言語 MCC.....	91



13.3.2.2	MCC ユニットの作成.....	92
13.3.2.3	MCC チャートの作成.....	93
13.3.2.4	MCC チャートへコマンドブロックの挿入.....	93
13.4	MCC サンプルプログラムの作成：ベーシックフレームワーク.....	96
13.4.1	概要.....	96
13.4.2	プログラムのフロー.....	96
13.4.3	変数の割り付け g_bo_ready:=false / g_bo_start:=true.....	97
13.4.4	変数の割り付け g_bo_start:=true.....	100
13.4.5	軸イネーブルへの切り替え.....	102
13.4.6	軸の原点復帰.....	103
13.4.7	軸のターゲット位置への位置決め.....	105
13.4.8	軸の開始位置への位置決め.....	107
13.4.9	軸の無効化.....	109
13.4.10	変数の割り付け g_bo_start:=false / g_bo_ready:=true.....	110
13.4.11	変数の割り付け g_bo_ready:=true.....	110
13.5	MCC サンプルプログラムの拡張：電源装置制御.....	111
13.5.1	プログラムのフロー.....	111
13.5.2	システムファンクションの呼び出し _LineModule_control[FB].....	113
13.5.3	変数の割り付け LineModule_STW:=myFB_LineControl.periOut.....	116
13.5.4	UNTIL ループを作成.....	117
13.5.5	ブロックのコピー.....	118
13.5.6	システムファンクションの呼び出し _linemodule_control の適用.....	119
13.6	サンプルプロジェクトのための追加 MCC プログラムの作成.....	119
13.7	MCC サンプルプログラムのバックアップ.....	120
13.8	LAD/FBD ラダーロジック / ファンクションブロックダイアグラム.....	121
13.8.1	LAD および FBD プログラミング言語.....	121
13.8.2	LAD/FBD ユニットの作成.....	122
13.8.3	LAD/FBD プログラムの作成.....	123
13.8.4	LAD サンプルプログラムの作成.....	124
13.8.5	LAD/FBD サンプルプログラムのバックアップ.....	128
13.9	その他のプログラミングオプション.....	129
13.10	サンプルプロジェクトでの結果.....	130
14	実行システムのコンフィグレーション.....	131
14.1	概要.....	131
14.2	プログラムをタスクに割り付け.....	131
14.3	コンフィグレーションされた実行システムのターゲットシステムへのダウンロード...135	
14.4	サンプルプロジェクトでの結果.....	136
15	システムの開始および停止.....	137
15.1	概要.....	137

15.2	RUN および STOP 動作モード.....	137
15.3	ソフトウェア側とハードウェア側のモードセクタスイッチ.....	138
15.4	サンプルプロジェクトのプログラム制御の開始.....	140
<b>16</b>	<b>アプリケーションの監視.....</b>	<b>141</b>
16.1	概要.....	141
16.2	プログラム実行の監視.....	141
16.3	変数の監視.....	143
16.4	トレース機能での信号の記録.....	145
16.4.1	トレース機能.....	145
16.4.2	トレース機能を使った作業.....	146
16.5	サンプルプロジェクトでの結果.....	150
<b>A</b>	<b>ESD に関する指示.....</b>	<b>151</b>
A.1	ESD の定義.....	151
A.2	人体の静電気蓄積.....	151
A.3	静電放電に対する基本的保護措置.....	152
	索引.....	153

## 基本的な安全に関する指示

### 1.1 一般的な安全に関する指示

#### 警告

##### 安全に関する指示および残存リスクを遵守しない場合の生命の危険性

関連するハードウェアの取扱説明書に記載された安全に関する指示および残存リスクを遵守しない場合、重傷または死亡に至る場合があります。

- ハードウェアの取扱説明書に記載された安全に関する指示を遵守してください。
- リスク評価では残存リスクを考慮してください。

#### 警告

##### 不正なパラメータ設定または変更されたパラメータ設定に起因する機械の誤動作による生命の危険性

不正なパラメータ設定または変更されたパラメータ設定は、傷害または死亡に至る機械の誤動作の原因となる場合があります。

- 承認されないアクセスに対するパラメータ設定変更 (パラメータ割り付け) を保護してください。
- 適切な対策を講じることで、考えられる誤動作に対応します (例 : EMERGENCY STOP または EMERGENCY OFF)。

## 1.2 産業セキュリティ

### 注記

#### Industrial security (産業 (サイバー) セキュリティ)

シーメンスでは、プラント、システム、機械装置およびネットワークの安全な運転をサポートする産業 (サイバー) セキュリティ機能を備えた製品およびソリューションを提供しています。

サイバー攻撃に対して、プラント、システム、機械装置およびネットワークを保護するために、総合的で最新の産業 (サイバー) セキュリティコンセプトを実装し、継続的に維持することが必要です。シーメンスの製品およびソリューションは、このようなコンセプトの一要素のみを構成するものです。

お客様には、プラント、システム、機械装置およびネットワークへの不正なアクセスを防止する責任があります。システム、機械装置およびコンポーネントは、必要な場合のみ、必要に応じて、十分なセキュリティ対策を講じて (例：ファイアウォールとネットワークの細分化)、企業ネットワークまたはインターネットに接続してください。

加えて、適切なセキュリティ対策に関するシーメンスのガイドラインを考慮してください。産業 (サイバー) セキュリティに関する詳細は、以下のサイトを参照してください：

<http://www.siemens.com/industrialsecurity>

シーメンスの製品およびソリューションは、更にセキュリティレベルを高めるために、継続的な開発が行われています。シーメンスは、可能になった時点でできるだけ早く更新を適用し、常に最新の製品バージョンを使用することを強く推奨します。もはやサポートされない製品バージョンの使用、最新版への更新適用失敗は、お客様へのサイバー攻撃の危険性を高めることがあります。

製品の更新情報を継続して受け取るには、<http://www.siemens.com/industrialsecurity> で Siemens Industrial Security RSS Feed を購読してください。



### 警告

#### ソフトウェアの誤操作に起因する危険な運転状態による死亡の危険性

ソフトウェアの誤動作、例えば、ウィルス、トロイの木馬、マルウェアまたはウォームは、死亡、重傷や物的損傷に至る恐れがあるシステムの安全ではない運転状態の原因となる場合があります。

- 最新のソフトウェアを使用してください。
- オートメーションおよびドライブコンポーネントを、設置された機器または機械装置に対する最先端の統合産業保全コンセプトに組み込んでください。
- 設置された製品すべてを統合産業 (サイバー) セキュリティコンセプトに組み込まれていることを確認してください。
- 適切な保護対策、例えば、ウィルススキャンを実行して、着脱可能な保存メディア上のファイルをマルウェアに対して保護してください。

1.3 脱着可能な記憶媒体を使用する際の、ソフトウェアの (不正な) 操作による生命の危険性

## 1.3 脱着可能な記憶媒体を使用する際の、ソフトウェアの (不正な) 操作による生命の危険性



### 警告

脱着可能な記憶媒体を使用する際の、ソフトウェアの (不正な) 操作による生命の危険性

脱着可能な記憶媒体へのファイルの保存により、例えば、ウィルスまたはマルウェアの感染率が高くなります。不正なパラメータ割り付けは、傷害や死亡に至る機械の誤動作の原因となる場合があります。

- 例えば、ウィルススキャンで有害なソフトウェアから、ウィルス・スキャンなどの適切な保護対策により脱着可能な記憶媒体を保護してください。

### 1.3 脱着可能な記憶媒体を使用する際の、ソフトウェアの (不正な) 操作による生命の危険性

## Getting Started (SIMOTION SCOUT TIA 用)

### 2.1 Getting Started の目的

Getting Started で、SIMOTION SCOUT TIA エンジニアリングシステムでの作業方法が紹介されます。単純なサンプルプロジェクトを作成します。こうして、デバイスやドライブ、軸のコンフィグレーションに含まれる代表的な手順を踏まえつつ、作業を行うことになります。SIMOTION SCOUT TIA がコンフィグレーション、プログラミングおよび診断のために提供する最も重要なツールに精通していただけます。

### 2.2 サンプルプロジェクト

Getting Started では、皆様が単純なサンプルプロジェクトを作成するための説明が記されています。

#### コンフィグレーション手順

##### コンフィグレーションの準備

- SIMOTION デバイスを出荷時設定に戻します。
- PG/PC および SIMOTION デバイス間のネットワーク通信用インターフェースをコンフィグレーションします。

##### プロジェクトの作成、SIMOTION デバイスおよびネットワークと PG/PC の通信コンフィグレーション

- プロジェクトを作成します。
- SIMOTION デバイスを作成し、PG/PC と SIMOTION デバイス間のネットワーク通信を設定します。

##### ドライブのコンフィグレーション

- ドライブの試運転を行ないます。

##### 電源装置のコンフィグレーション

- 電源装置とドライブを接続します。

## 2.3 要件

### 軸のコンフィグレーションおよび試験

- 軸を設定します。
- 軸とドライブを接続します。
- 軸を軸のコントロールパネルで試験します。

### 入 / 出力のコンフィグレーション

- サンプルプログラムでの使用のために I/O をコンフィグレーションします。

### SIMOTION のプログラム、設定および監視

- コンフィグレーションされた軸を制御する単純な SIMOTION ユーザプログラムを書きます。
  - プログラムで必要とされる変数を作成します。
  - グラフィックエディタでプログラムとその他の補助プログラムを作成します。
- 作成したプログラムを実行システムのタスクに割り付けます。
- SIMOTION ランタイムシステムでプログラムの実行を開始します。
- プログラムで制御された軸の動作を監視します。
  - プログラムの実行を監視します。
  - シンボルブラウザで、これらの値を監視します。
  - ウォッチテーブルの値をコンパイルします。
  - トレースで軸動作の軌道を記録します。

## 2.3 要件

Getting Started では、皆様が単純なサンプルプロジェクトを作成するための説明が記されています。



## トレーニングシステム

サンプルプロジェクトを作成するには、いくつかのコンポーネントを含むトレーニングシステムが必要です：

- 最新のファームウェアを備えた SIMOTION D4x5-2 デバイス
- フリーの Ethernet インターフェース付き PG/PC  
USB Ethernet アダプタは Ethernet 接続にも適しています。

---

### 注記

USB PROFIBUS アダプタ、例えば、PC アダプタ USB A2 (手配形式：6GK1571-0BA00-0AA0) は PROFIBUS 接続に使用できます。それ以前のバージョンが使用される場合、パフォーマンス上の問題が発生することがあります。

- コンポーネントの完全な DRIVE-CLiQ 配線；DRIVE-CLiQ インターフェース付きモータ、そして、自動エンコーダ定数測定 (SMI 内蔵センサモジュール)。
- TIA ポータルおよび SIMOTION SCOUT TIA エンジニアリングシステム

SIMOTION D435-2 DP/PN デバイスが、サンプルプロジェクトで使用されています。

SIMOTION デバイスの PROFINET インターフェースはサンプルプロジェクトでは使用されません。

---

### 注記

Getting Started では、自動ドライブコンフィグレーションが扱われますが、ドライブウィザードを使ったコンフィグレーションは扱われません。自動ドライブコンフィグレーションを完全に実行できるための必要条件は、該当するコンポーネントの完全な DRIVE-CLiQ 配線です。

## トレーニングシステムの準備

トレーニングシステムの TIA ポータルおよび SIMOTION SCOUT TIA でのコンフィグレーションの準備完了：

- ハードウェアは既に設置され、配線されています。
- 最新のファームウェアを含む CF カードが挿入されています。
- PG/PC および SIMOTION D435-2 は直接 Ethernet ケーブルで接続されています。  
SIMOTION D435-2 の Ethernet X127 インターフェースが接続に使用されています。
- TIA ポータル (V14 以降) が PG/PC にインストールされています。

## 2.3 要件

- SIMOTION SCOUT TIA (V4.5 以降) が PG/PC にインストールされ、正しいライセンスが与えられています。
- TIA ポータルを起動し、ポータルビューが PG/PC の画面上で確認されています。

## コンフィグレーション手順の概要

下表で、それぞれのコンフィグレーション手順の概要とそれらを実装する場所が提供されています。

コンフィグレーション手順	場所	章
① プロジェクトを作成	TIA ポータル	プロジェクトの作成 (ページ 29) を参照
② SIMOTION デバイスを作成し、オンライン通信をコンフィグレーション (ハードウェアコンフィグレーションのコンパイルおよび SIMOTION デバイスへのダウンロードを含む)	TIA ポータル	SIMOTION デバイスの作成およびオンライン通信のコンフィグレーション (ページ 33) を参照
③ SIMOTION SCOUT TIA を開始	TIA ポータル	SIMOTION SCOUT TIA の開始 (ページ 49) を参照
④ ターゲットシステムにプログラムをダウンロード (コンパイルを含む)	SIMOTION SCOUT TIA	プロジェクトをターゲットシステムにダウンロード (ページ 51) を参照
⑤ ドライブのコンフィグレーション	SIMOTION SCOUT TIA	ドライブのコンフィグレーション (ページ 59) を参照
⑥ 電源装置のコンフィグレーション	SIMOTION SCOUT TIA	電源装置のコンフィグレーション (ページ 65) を参照
⑦ テクノロジーオブジェクトのコンフィグレーション (TO 軸の例で)	SIMOTION SCOUT TIA	軸のコンフィグレーション (ページ 69) / 軸のコントロールパネルを使った軸の試験 (ページ 75) を参照
⑧ デジタル出力をコンフィグレーション	SIMOTION SCOUT TIA	デジタル出力のコンフィグレーション (ページ 81) を参照
⑨ ユーザプログラムの作成	SIMOTION SCOUT TIA	SIMOTION アプリケーションのプログラミング (ページ 83) を参照
⑩ 実行システムのコンフィグレーション	SIMOTION SCOUT TIA	実行システムのコンフィグレーション (ページ 131) を参照

コンフィグレーション手順	場所	章
⑪ システムの開始および停止	SIMOTION SCOUT TIA	システムの開始および停止 (ページ 137)を参照
⑫ アプリケーションの監視	SIMOTION SCOUT TIA	アプリケーションの監視 (ページ 141)を参照

## 2.4 一般情報

### 2.4.1 TIA ポータル オンラインヘルプ

TIA ポータルには広範なオンラインヘルプシステムがあります。

#### オンラインヘルプの開き方

- メニューで **Help > Show help** を選択、または、
- **F1** キーを押してください。

オンラインヘルプを呼び出した後、インフォメーションシステムに入ります。利用可能なヘルプおよび TIA ポータルでのヘルプの利用方法は、**Further support > Help on the information system** で説明されています。

### 2.4.2 SIMOTION SCOUT TIA オンラインヘルプ

SIMOTION SCOUT TIA には包括的なオンラインヘルプが備わっています。オプションで、オンラインヘルプの特別な項目の検索、または、作業内容からのヘルプを呼び出すことができます。

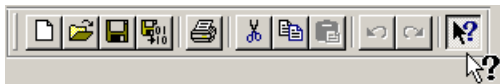
#### オンラインヘルプの開き方

- メニューの **Help > Help topics** を選択する、または、
- **F1** キーを押してください。

## 2.4 一般情報

### 内容に依存するオンラインヘルプの開き方

- ダイアログまたはウィンドウのヘルプボタンをクリックする、または、
- **Shift+F1** キーの同時押し、または、
- ツールバーの **Help** ボタンをクリックしてください。



(クエスチョンマークに変わった) マウスポインターを使って、ヘルプを必要とするパラメータまたはウィンドウをクリックしてください。

**SIMOTION SCOUT TIA > General > Use Help** のオンラインヘルプで、内容に依存したヘルプの使用に関する詳細情報を入手できます。

### オンラインヘルプの全文検索

**Basic > Use Online Help and Function Block Diagrams > Use Online Help > Full Text Search** の下のオンラインヘルプに、全文検索に関する詳細があります。

---

#### 注記

#### オンラインヘルプの全文検索

長い文字列の一部としての検索文字を含む検索結果を含めるために、検索文字を星印 \* ではさんでください。例えば、

- **variable** では、"variable" または "Variable" の単語が照会されます。
  - **\*variable\*** では "system variables"、"variable assignment" などが照会されます。
- 

## 2.4.3 利用可能な資料 / 文書

### 電子文書

本書は、SIMOTION SCOUT TIA (SIMOTION SCOUT DVD Documentation、Utilities & Applications) の納品範囲に電子媒体で含まれます。電子文書形式のすべての PDF 文書を索引で検索することができます (SIMOTION.pdx)。個別の文書 *Overview of the SIMOTION Documentation* に SIMOTION の PDF 文書の構造および目次一覧があります。

僅かに例外がありますが。文書 / 資料の内容は SIMOTION SCOUT TIA オンラインヘルプでも利用可能です。

オンラインヘルプで、**Overview of the SIMOTION Documentation > ... > Overview of the SIMOTION Documentation** の下に *Overview of the SIMOTION Documentation* があります。

---

#### 注記

*SIMOTION SCOUT TIA Configuration Manual* は SIMOTION SCOUT DVD に PDF 文書としてあります。

---

## SIMOTION の概要

オンラインヘルプの **Basic > Basic Functions > System Overview** に、SIMOTION の簡潔なシステムの概要があります。

本書には、特に **SIMOTION D** のコンフィグレーションおよび試運転に関する包括的な情報が含まれます。*SIMOTION D4x5-2* マニュアルおよび *SIMOTION D4x5-2 試運転* マニュアルおよびハードウェア設置マニュアルを参照してください。オンラインヘルプの **SIMOTION Devices > SIMOTION D** にも SIMOTION D マニュアルがあります。

## 関連情報

Industry Online Support に SIMOTION の関連情報および資料があります。ここで、SIMOTION の最も重要な技術情報とソリューション一覧が提供されます。

<http://support.automation.siemens.com/WW/view/de/10805436/130000>

TIA ポータルのハードウェアカタログ経由でもオンラインサポートにアクセスできます (該当するモジュールを選択し、ショートカットメニューを開いてください)。

## 2.4 一般情報

### 2.4.4 ユーティリティ & アプリケーション

無料の SIMOTION ユーティリティ & アプリケーションで、SIMOTION のすべての要素、つまり、ツール、特殊機能、ブロック、SIMOTION サンプルプロジェクト並びに例証またはプロジェクトでの使用のための既成の標準アプリケーションに関する多くの重要なバックグラウンド情報が提供されます。そこで、SIMOTION での作業を容易にする多くのサンプルスクリプトに関する情報も入手していただけます。

#### 1. FAQ

油圧軸の制御や通信問題などの興味深い FAQ

#### 2. スクリプト

例えば、多くのスクリプト、有効なヒントだけでなく、反復作業を容易にする包括的な解決方法

---

#### 注記

SIMOTION SCOUT TIA (TIA ポータル) の一部として、純粋な SIMOTION SCOUT/ SIMOTION SCOUT TIA データや機能 (性) に影響を及ぼす、または、それらを使用するスクリプトのみを使用することができます。

データのスクリプト / HWCN の機能性、および TIA ポータル (フレームワーク) のプロジェクト処理は不可能です。

以下は、明らかにスクリプトでは不可能です：

- TIA ポータル (例えば、すべての SIMOTION デバイス) で表示されるオブジェクトの作成、削除および名称変更。
- プロジェクトの作成、削除および名称変更
- ファイルシステムの TIA ポータル プロジェクトへのアクセス。
- TIA ポータル (フレームワーク) で提供される機能の使用、例えば、
- アーカイブ保存 / アーカイブ展開
- すべての TIA ポータル データの処理

SIMOTION easyProject プロジェクトジェネレータは SIMOTION SCOUT TIA のプロジェクトに使用できません。なぜなら、スクリプト機能が現時点で TIA ポータルで利用できないためです。

SIMOTION SCOUT TIA の外部スクリプトがサポートされないことにも注意してください。

---

#### 3. ツール & 文書 / 資料

多くのタスクのための操作性に優れたツールと詳細な文書 / 資料が提供されます。

#### 4. 例

初回ユーザのためのサンプルプロジェクト、例えば、"Getting Started" 並びに特殊な項目の例

#### 5. アプリケーション

SIMOTION には、すでに確立されている基本的なフレームワークを提供する多数のアプリケーションが用意されています。提供された資料を使って、このアプリケーションを皆様自身のアプリケーションとして使用し、それらを適用し、拡張することができます。更に、皆様のアプリケーション、例えば DPV1 を介したドライブ通信用の LDPV1 ライブラリ (SINAMICS パラメータの読み出しや書き込み、SINAMICS からのエラーや警告の読み出し、オブジェクトの無効化、立ち上がり条件など)、または SIMOTION や SIMATIC のための TCP/IP での通信用機能を含む LCom ライブラリの作成時に、一般的に役に立つ "Cross-Sector application" で利用可能です。

#### 6. SIMOTION IT

SIMOTION IT には、制御やドライブソリューションを少し IT 環境に近づける革新的な例やツールがありますこれらには、インターネットブラウザで実行されるトレース機能、SIMOTION IT ウェブサーバや OPC XML DA を使用するためのユーザ定義のウェブページが含まれます。

### 2.4.5 特別な安全情報

トレーニングシステムが完全に生産運転から切り離されていることを確認してください。

使用されるデバイスの説明書にある安全に関する指示を遵守してください。

## 2.4 一般情報



## コンフィグレーションの準備

### 3.1 出荷時設定の復元

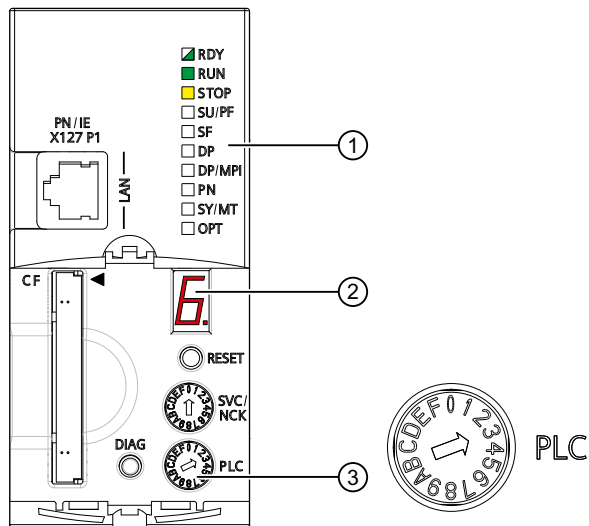
サンプルプロジェクトの場合、SIMOTION デバイスを出荷時設定にリセットすることが有用です。

こうして、デフォルトの通信パラメータに復元し、デバイスや CF カードにインストールされたユーザデータを以前のコンフィグレーションで削除します。ランタイムライセンスは保持されます。

### 3.1 出荷時設定の復元

以下の方法で、SIMOTION デバイスの出荷時設定に戻します。

1. SIMOTION デバイスの電源をオフにしてください。
2. SIMOTION デバイスのモードセクタスイッチ ③ を **3** の位置に設定してください。



- ① LED 表示
- ② 7 セグメント表示
- ③ 位置 3 のモードセクタスイッチ

図 3-1 SIMOTION D435-2 モジュールの正面図

#### 通知

##### 静電気放電による損傷

ロータリスイッチが静電気により破損するおそれがあります。  
ロータリスイッチは、絶縁されたドライバでのみ操作してください。  
ESD 規則を遵守してください。

3. SIMOTION デバイスの電源をオンにしてください。  
デフォルト設定がロードされます。SIMOTION D435-2 は STOP モードに切り替わります。  
この手順が終了するのを待機してください。モジュール正面の要素が完了を示します：
  - 7 セグメントディスプレイ ② はステータス番号 **6** を表示：SIMOTION D435-2 が起動しました。
  - LED ① :  
LED **RDY** が緑で点滅 (0.5 Hz)：ドライバは、試運転準備完了です。  
LED **STOP** が黄で点灯：SIMOTION デバイスは STOP モードです。  
他のすべての LED はオフです。
4. モードセクタスイッチ ③ を **0** の位置に回してください。
  - LED **RUN** が緑で点灯：SIMOTION デバイスは RUN モードです。

## 結果

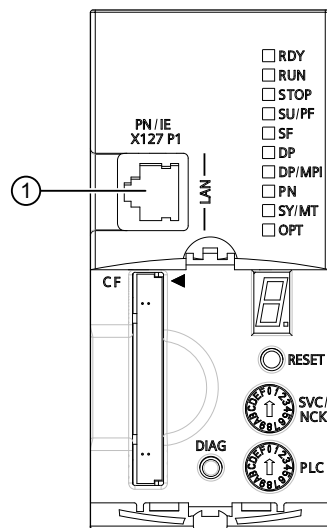
SIMOTION デバイス は出荷時設定に戻され、試運転の準備完了です。

## 3.2 オンライン通信のための要件

SIMOTION デバイスとの PG/PC のオンライン通信は PROFIBUS、PROFINET または産業用 Ethernet 経由で設定できます。サンプルプロジェクトは、最も頻繁なアプリケーションケースに制限されます：産業用 Ethernet での通信。

## 要件

- PG/PC と SIMOTION デバイスが Ethernet ケーブルで接続されています。
- SIMOTION デバイスの X127 PN/IE インターフェースが Ethernet 接続に使用されています。



① Ethernet インターフェース X127 PN/IE

図 3-2 SIMOTION D435-2 モジュールの正面図

- Ethernet インターフェース X127 PN/IE にはデフォルトのアドレスがあります：  
IP アドレス：169.254.11.22 - サブネット：255.255.0.0

## 下記も参照

Ethernet インターフェース (2) をコンフィグレーション (ページ 36)

### 3.3 サンプルプロジェクトでの結果

## 3.3 サンプルプロジェクトでの結果

SIMOTION デバイスの出荷時設定が復元されました。PG/PC と SIMOTION デバイスのオンライン通信の要件が満たされます。

## プロジェクトを作成

### 4.1 概要

#### Getting Started の目的

Getting Started の本項では、TIA ポータルでサンプルプロジェクト **Sample\_1** を作成します。後続するコンフィグレーション手順はすべてサンプルプロジェクトに言及するものです。

#### プロジェクト

プロジェクトには、機械装置とその機能を説明するすべての情報が含まれます：コンフィグレーションデータ、プログラム、モーションプロファイル、ドライブデータ。

プロジェクトに複数の **SIMOTION** デバイスを含めることができます。

### 4.2 新しいプロジェクトを作成

TIA ポータルを開くと、ポータルビューが表示されます。ポータルビューは、ツールのタスク別のビューを提供し、それぞれのタスク域のための基本機能を提供します。

## 4.2 新しいプロジェクトを作成

### 新規プロジェクトを以下の手順で作成

TIA ポータルで新規プロジェクトを作成することで、新しいコンフィグレーションを開始します。

1. ポータルビューのナビゲーションで **Start > Create new project** を選択してください。  
**Create new project** ダイアログ (ボックス) が現れます。

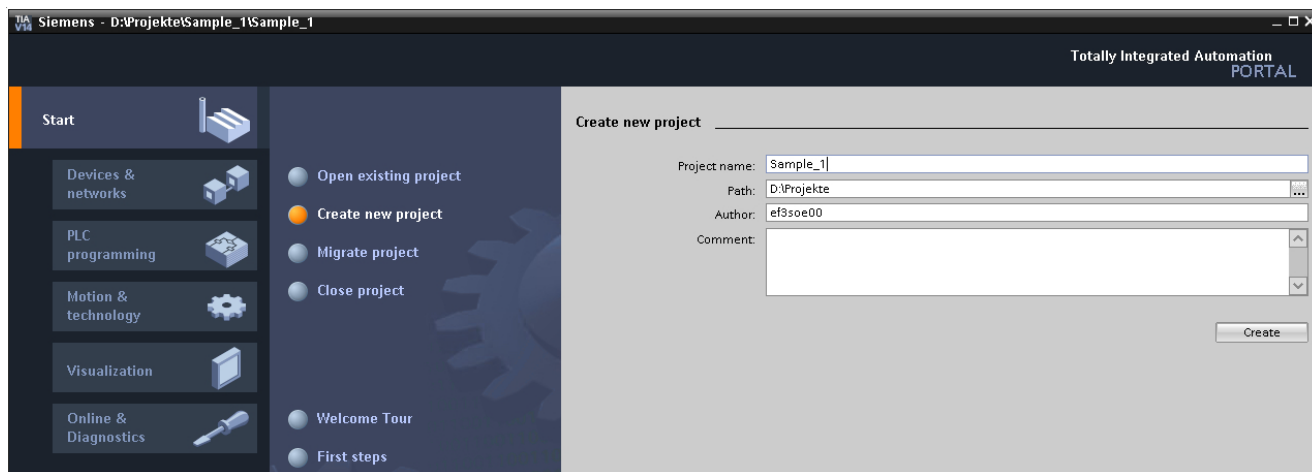


図 4-1 プロジェクトを作成

2. **Project name** にプロジェクト名を入力してください。例えば、**Sample\_1**。
3. **Path** に、プロジェクトの保存パスを入力してください。デフォルトのパスは既に設定されています。
4. **Create** で確定してください。

## 4.3 サンプルプロジェクトでの結果

Getting Started のサンプルプロジェクトが TIA ポータルで作成されました。

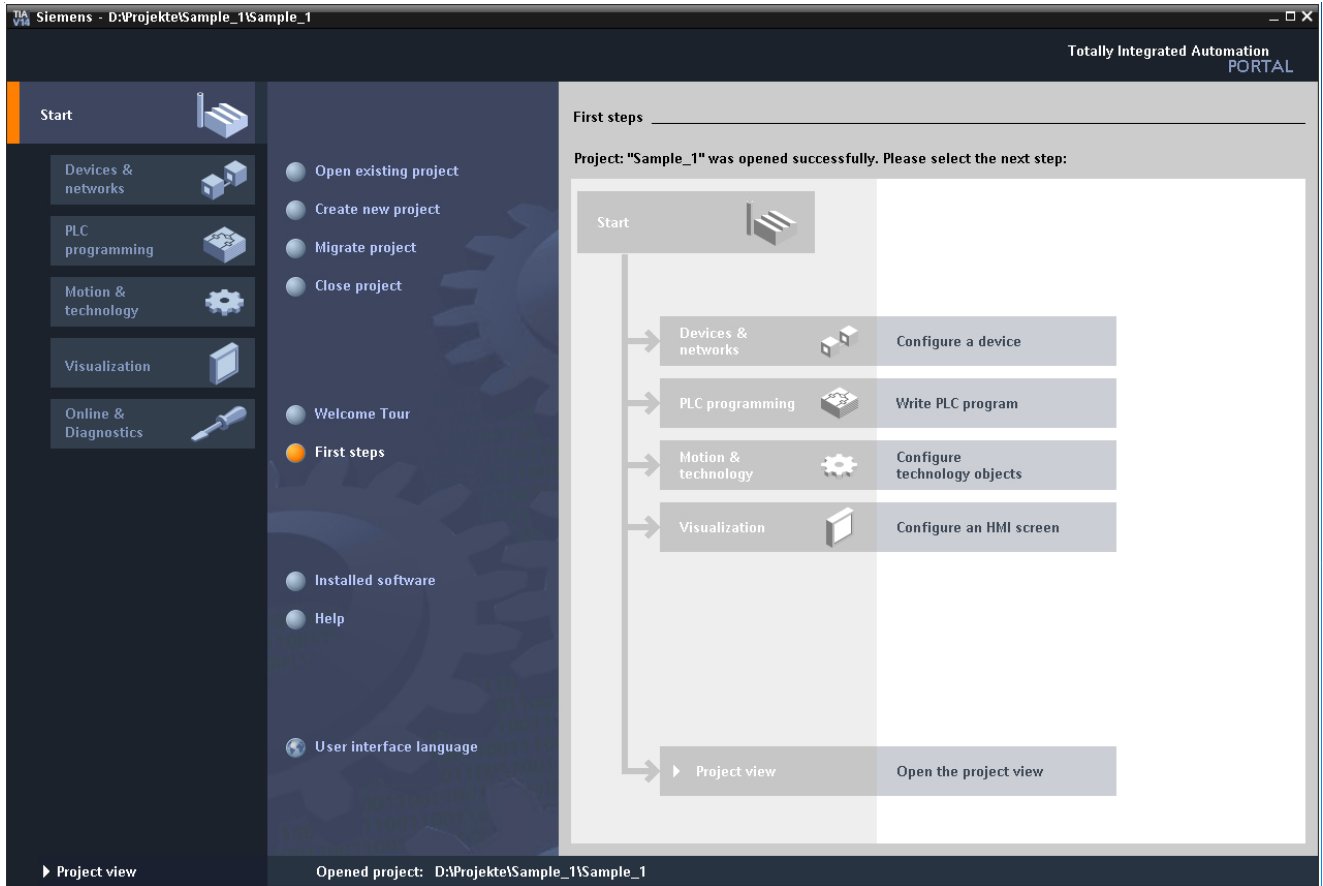


図 4-2 作成されたプロジェクト

#### 4.3 サンプルプロジェクトでの結果



# SIMOTION デバイスの作成およびオンライン通信の コンフィグレーション

# 5

## 5.1 概要

### Getting Started の目的

Getting Started の本項では、TIA ポータルでのプロジェクトで SIMOTION デバイスの作成方法、PG/PC と SIMOTION デバイス間での通信の設定方法を示します。

### 手順

プロジェクトで SIMOTION デバイスを作成するには、以下の手順に従ってください：

1. SIMOTION デバイス を作成してください。
2. Ethernet インターフェースをコンフィグレーションしてください。
3. PG/PC と SIMOTION デバイス間の通信を設定、プロジェクトをコンパイル、デバイスにダウンロード。

### SIMOTION D プラットフォーム

SIMOTION D は、SINAMICS 120 ドライブファミリをベースとする、SIMOTION モーションコントロールシステムのドライブベースバージョンです。SIMOTION D では、SIMOTION モーションコントロール機能と SINAMICS ドライブソフトウェアが、SINAMICS タイプの閉ループ制御ハードウェアデバイス上で動作します。SIMOTION D デバイスには以下の特性があります：

- ドライブに直接統合されたモーションコントロール機能および制御機能
- コンパクトかつ特に応答時間が短い
- 様々なアプリケーション用の異なるパフォーマンスバージョンでの 1 軸または多軸システムとして、最大限のスケラビリティおよびフレキシビリティ
- 高速アイソクロナスカップリングを含むモジュラーマシンコンセプトに特に適しています、多軸機械装置で、例えば、

SINAMICS S120 多軸ドライブシステムの閉ループ制御モジュールの SINAMICS の機能性は、SIMOTION D (SINAMICS Integrated) に統合されています。

## 5.2 SIMOTION デバイス (1) を作成

D435 コントロールユニット例を使った SIMOTION D の簡潔な説明がオンラインヘルプにあります。SIMOTION SCOUT のメニューバーの **Help > Tutorials > SIMOTION Drive-Based** をクリックしてください。

## 5.2 SIMOTION デバイス (1) を作成

以下の方法で、プロジェクトに SIMOTION コントローラを作成します。

1. ポータルビューのナビゲーションで、**Start > Devices & Networks > Add new device** を選択してください。  
ダイアログ (ボックス) **Add new device** が現れます。
2. **Controller** で、希望するデバイス変数を開いてください (例 : SIMOTION D - ドライブベース)。
3. 希望する SIMOTION device **D435-2 DP/PN** を選択してください。
4. そのデバイス固有の情報を表示するために、手配形式をクリックしてください。

5. **Version** で、設置された SIMOTION デバイスのファームウェアバージョンを選択してください。

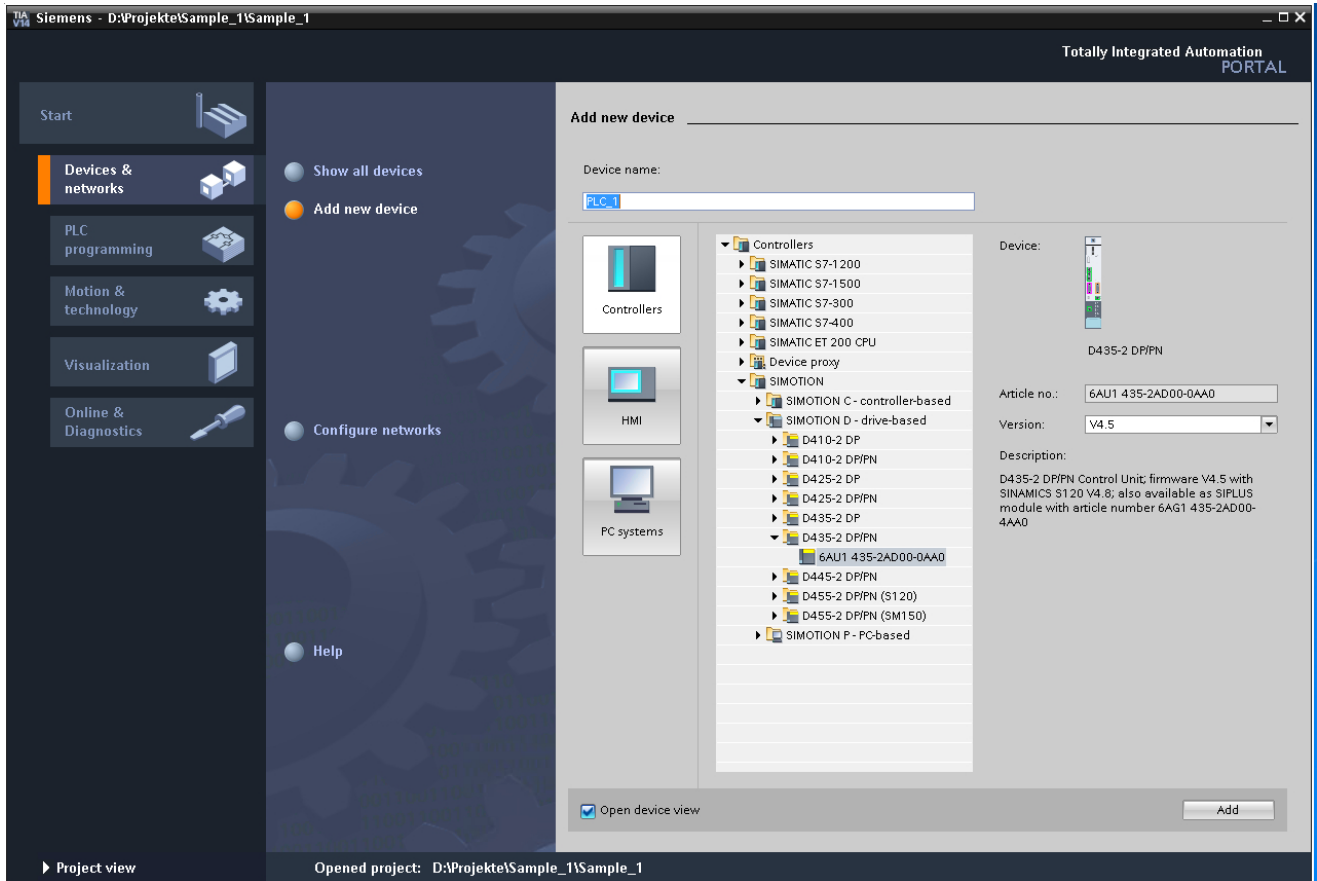


図 5-1 SIMOTION D435-2 DP/PN を選択

6. **OK** をクリックし、設定を適用し、SIMOTION デバイスを作成してください。

### 5.3 Ethernet インターフェース (2) をコンフィグレーション

#### 結果

SIMOTION デバイスはプロジェクトで作成されます。デバイス作成後、プロジェクトビューのデバイスビュータブが表示されます。

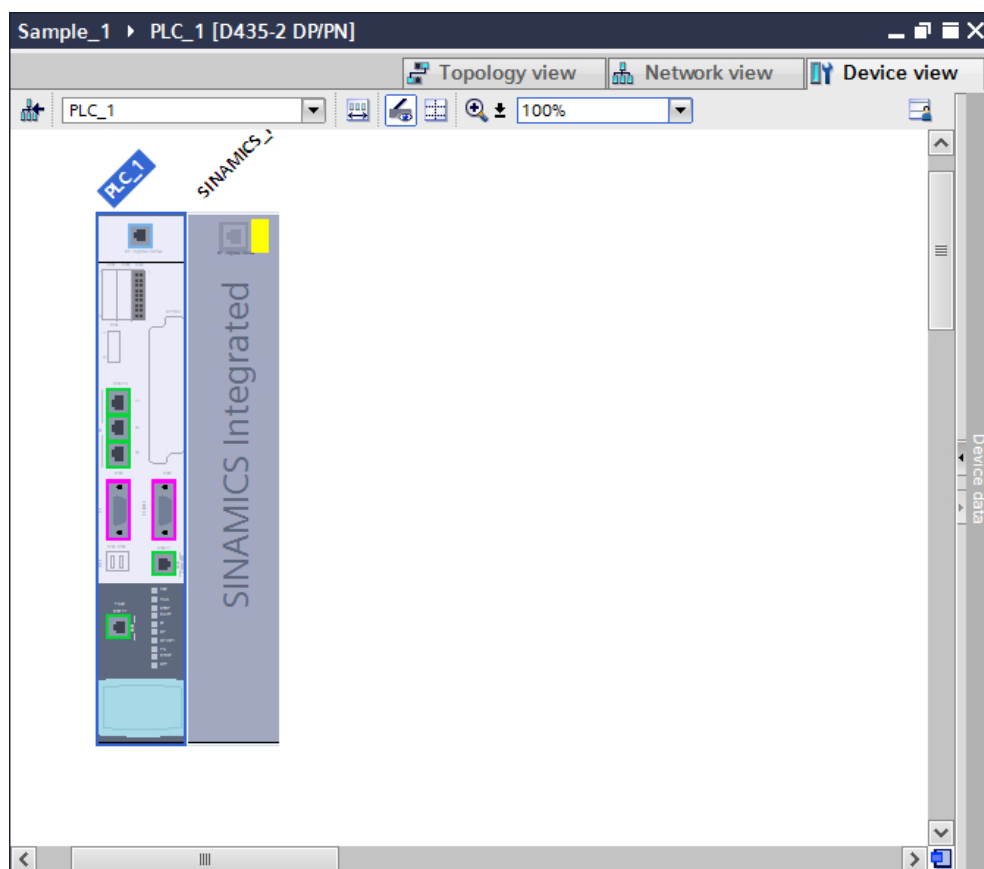


図 5-2 デバイスビューの SIMOTION D435-2 DP/PN

#### 注記

選択されたファームウェアバージョンは、デバイスのメモ리카ードのファームウェアと一致する必要があります。それ以外の場合、オンラインモードに切り替えると直ちに、エラーメッセージが現れます。

## 5.3 Ethernet インターフェース (2) をコンフィグレーション

PG/PC と SIMOTION デバイスのオンライン通信を PROFIBUS、PROFINET、または Industrial Ethernet 経由で設定できます。

最も一般的な使用は、産業用 Ethernet (PROFINET プロトコルでの通信) での通信です。

IO デバイスをコンフィグレーションする場合、サブネットを作成する必要があります。以下の例では、サブネットの作成方法が示されています。

#### 手順

サブネットを作成するには、以下の手順に従ってください：

1. これを行うには、まずプロジェクトビューでネットワークビューに切り替えてください。
2. マウスで、サブネットの作成を希望する SIMOTION デバイスの Ethernet インターフェース **X127** をクリックしてください。
3. ショートカットメニューから **サブネットの追加** を選択するために右クリックしてください。

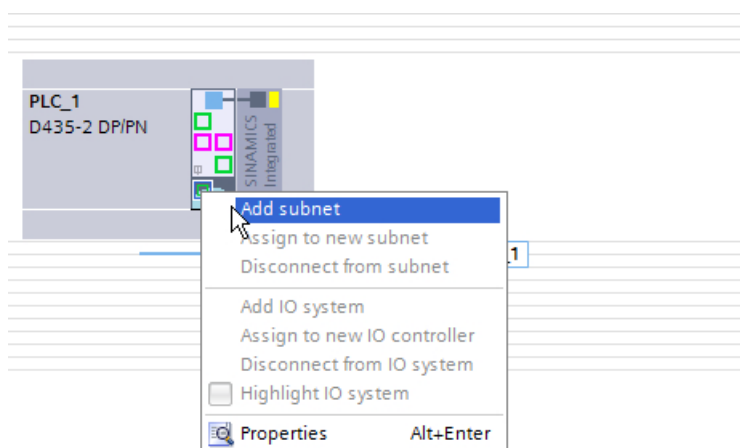


図 5-3 サブネットの追加

#### 結果

Ethernet サブネットは、ネットワークビューで示されます。

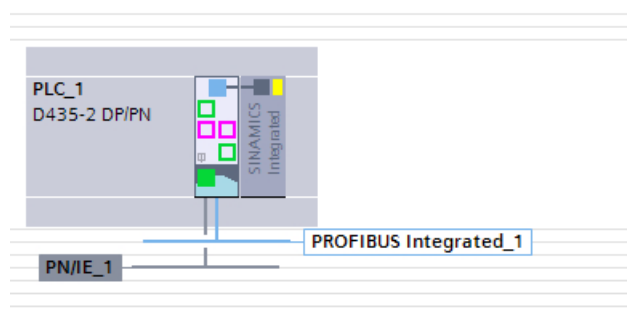


図 5-4 ネットワークビューの Ethernet サブネット

## 5.4 PG/PC インターフェース (3) を設定

---

### 注記

TIA ポータルで PG/PC インターフェースを割り付けます。

---

SIMOTION デバイスと PG/PC 間の通信にはコンディショナカード (PROFIBUS 用) または Ethernet インターフェースが必要です。PG/PC を使ってコンフィグレーション、パラメータ設定、プログラムおよび試験を行います。

PG/PC インターフェースを割り付けるために、以下の機能があります：

- **Connect online** (オンライン接続) 機能  
この機能は、PG/PC インターフェースが正常に設定されるまで利用可能です。
- **Online & diagnostics** (オンライン & 診断) 機能
- **Online access** (オンラインアクセス) 機能

### PG/PC インターフェースの割り付け

次の手順では、[オンラインアクセス] 機能を使用する Ethernet インターフェースタイプのプロセスについて記述します。

インターフェースを割り付けるには、以下の手順に従ってください：

1. プロジェクトナビゲータの **オンラインアクセス** の下で、関連するインターフェースに移動します。
2. ショートカットメニューで、**プロパティ** をクリックしてください。

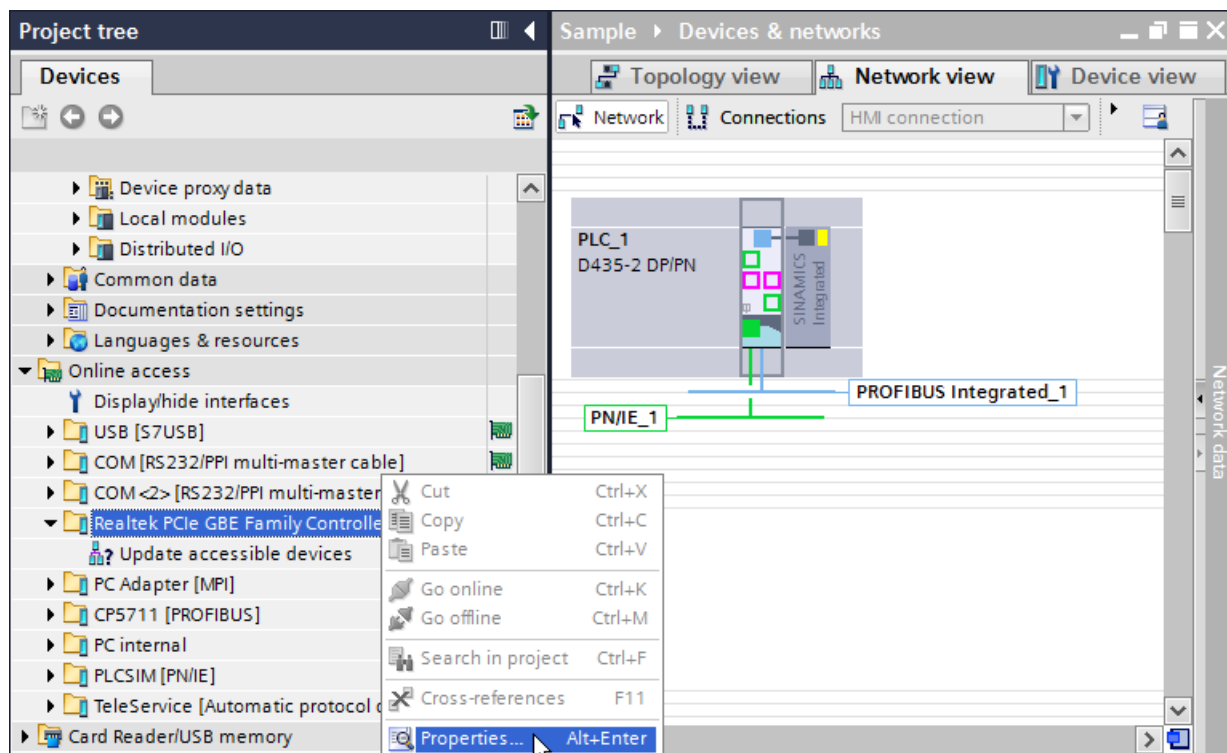


図 5-5 オンラインアクセスのプロパティ

3. 次の段階で、サブネットを選択し、この設定を **OK** で適用してください。

## 5.4 PG/PC インターフェース (3) を設定

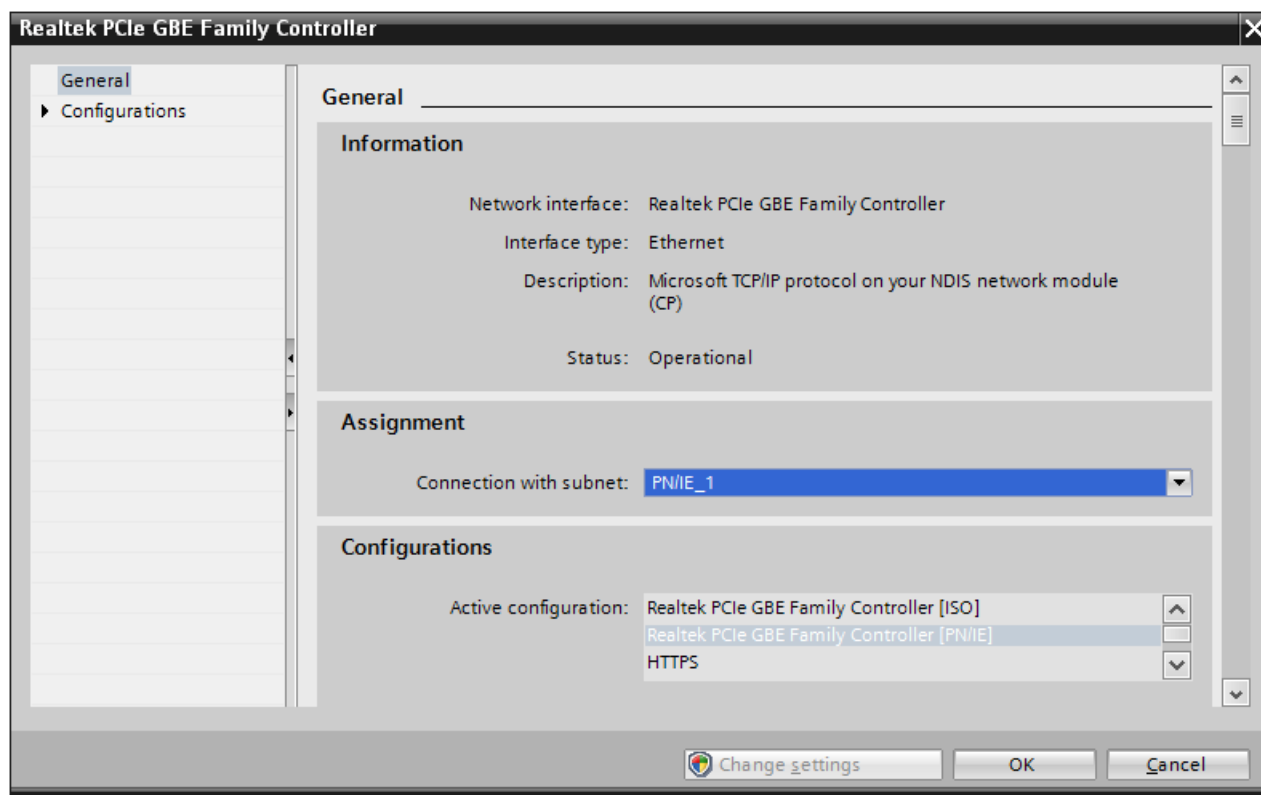



図 5-6 サブネットの割り付け

### サブネットに IP アドレスを追加

SIMOTION デバイスと PG/PC が同じサブネットにない場合、デバイスのサブネットからの IP アドレスを PG/PC に追加してください。



サブネットで自動的に IP アドレスを追加するには、以下の手順に従ってください：

1. ツールバーの  **Go online** をクリックしてください。  
**Connect online (オンライン接続)** ダイアログが開きます。

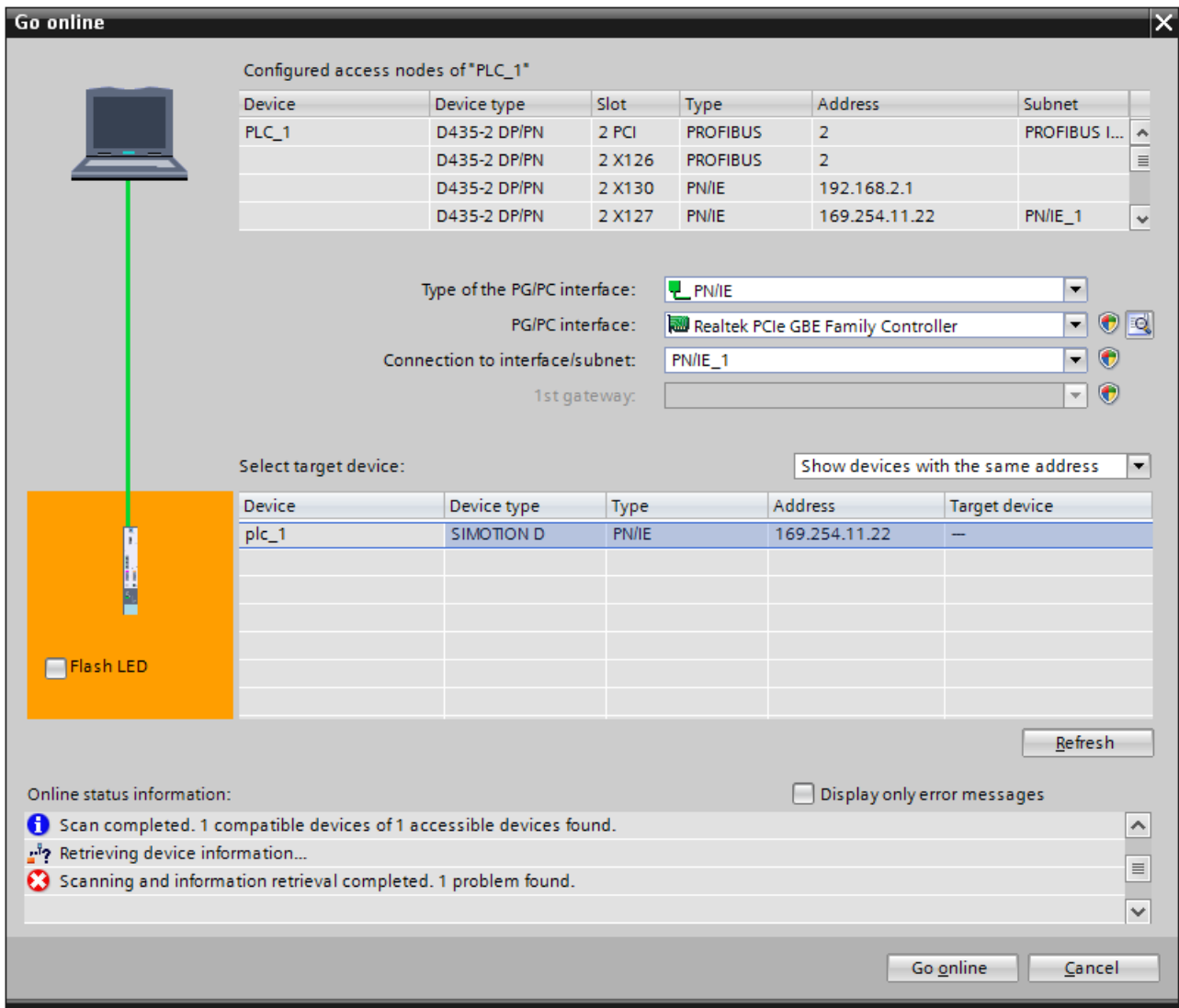


図 5-7 オンライン接続のためのデバイスの選択

2. この検索で確認されたデバイスとの接続を確立するために **Connect (オンライン接続)** をクリックしてください。

## 5.4 PG/PC インターフェース (3) を設定

3. SIMOTION デバイスと PG/PC が同じサブネットにない場合、デバイスのサブネットから一時的に適切な IP アドレスを割り付けるオプションを提供するメッセージが表示されます。

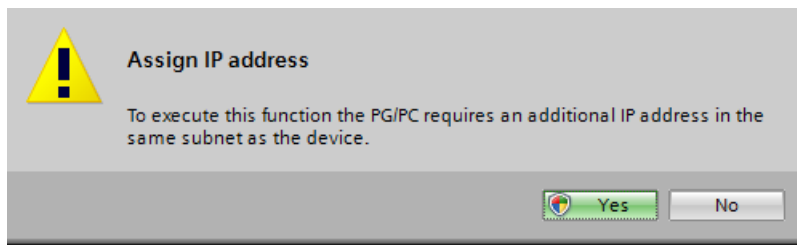


図 5-8 IP アドレスの割り付け

4. **Yes (はい)** をクリックし、確定してください。

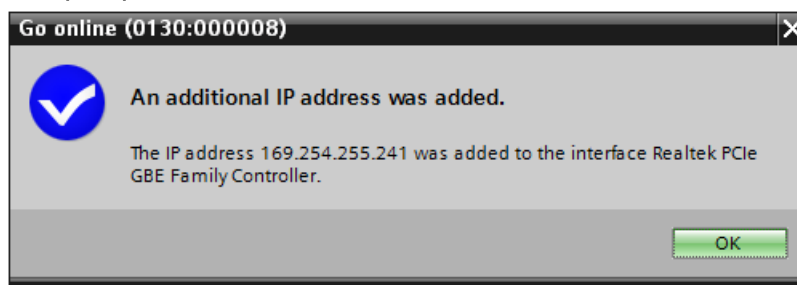


図 5-9 割り付けられた IP アドレス

## 結果

- PG/PC インターフェースを割り付けました。
- TIA ポータルは、プロジェクト内で、内部的に IP アドレスを割り付けました。
- オンライン接続が確立されました。  
プロジェクトナビゲーションのタイトルバーはこの時オレンジ色です。

## 暫定的な IP アドレスの表示および削除

すべての一時的に割り付けられたアドレスを表示し、それらを再び削除することができます。

1. プロジェクトナビゲータのオンラインアクセスの下で、関連するインターフェースに移動します。
2. ショートカットメニューで、**プロパティ** をクリックしてください。
3. **設定** で、項目 **IE-PG アクセス** を選択してください。

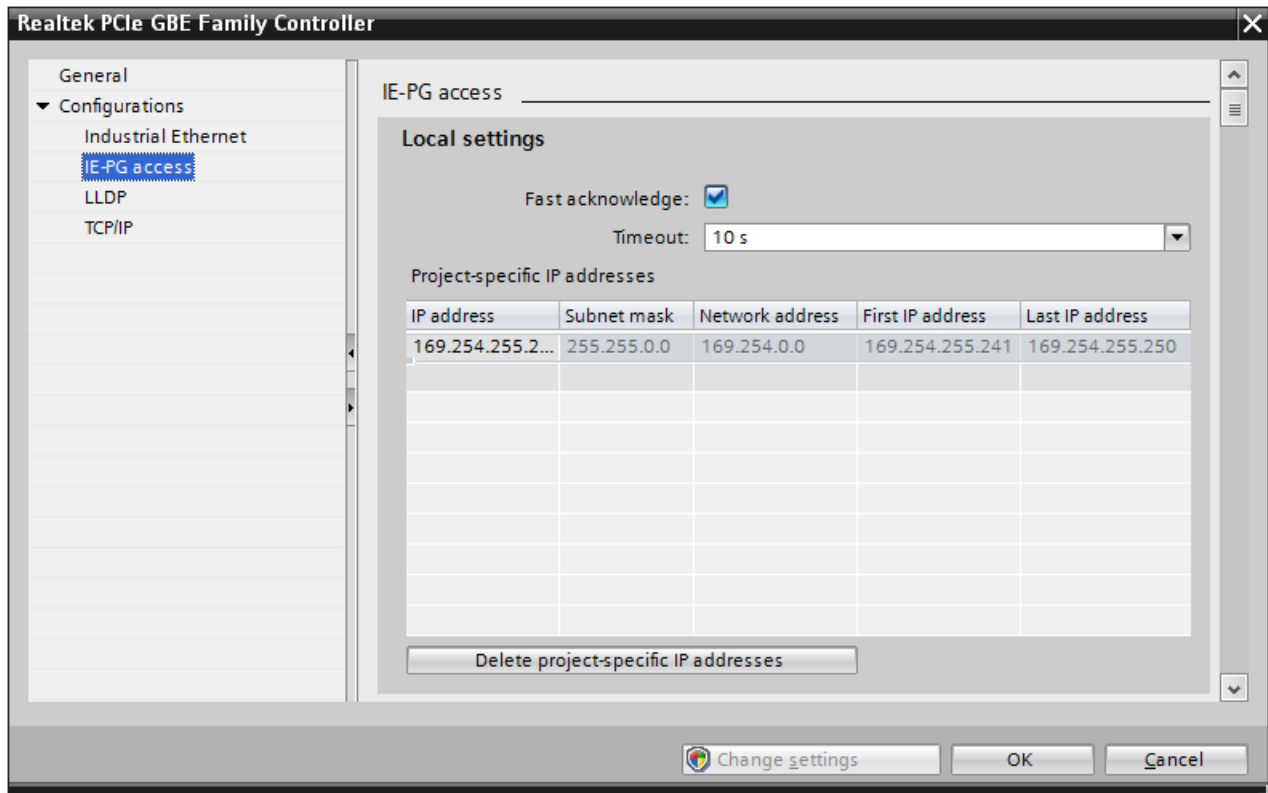


図 5-10 暫定的な IP アドレスの表示および削除

#### 5.4 PG/PC インターフェース (3) を設定

プロジェクトのコンパイルおよび **SIMOTION** デバイスへのハードウェアコンフィグレーションのダウンロード

---

##### 注記

SIMOTION SCOUT TIA でコンフィグレーションを続ける前に、TIA ポータルハードウェアコンフィグレーションを **SIMOTION** デバイスにダウンロードしてください。

例えば、IP アドレスのデバイス名を変更したい場合には、TIA ポータルから **SIMOTION** デバイスへのダウンロードが必要です。更に、TIA ポータルへのダウンロードで、布線情報がロードされます。

---

#### 要件

SIMOTION デバイスとのオンライン接続は解除されます。

## 手順

ハードウェアコンフィグレーションをコンパイルし、SIMOTION デバイスにダウンロードするには、以下の手順に従ってください：

1. SIMOTION デバイスを選択し、ショートカット経由で **Compile > Hardware (changes only)** を選択してください。

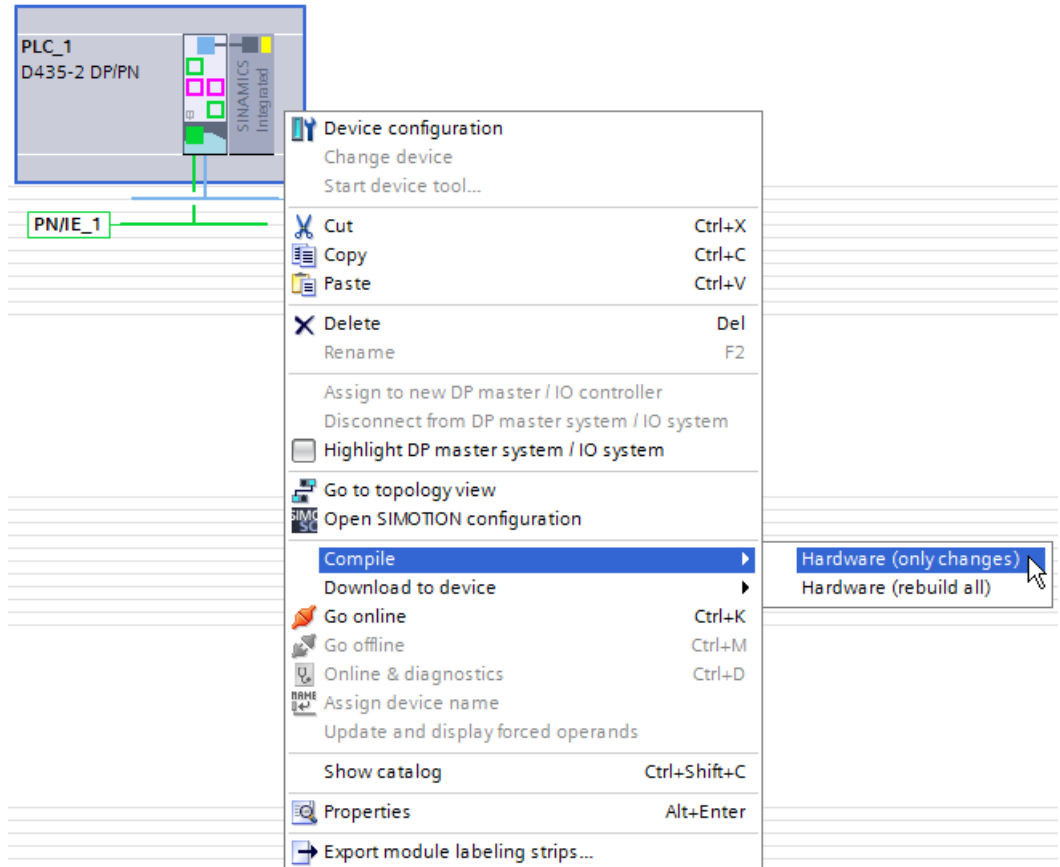


図 5-11 ハードウェアのコンパイル

2. SIMOTION デバイスを選択し、ショートカット経由で **Download to device > Hardware configuration** を選択してください。

## 5.4 PG/PC インターフェース (3) を設定

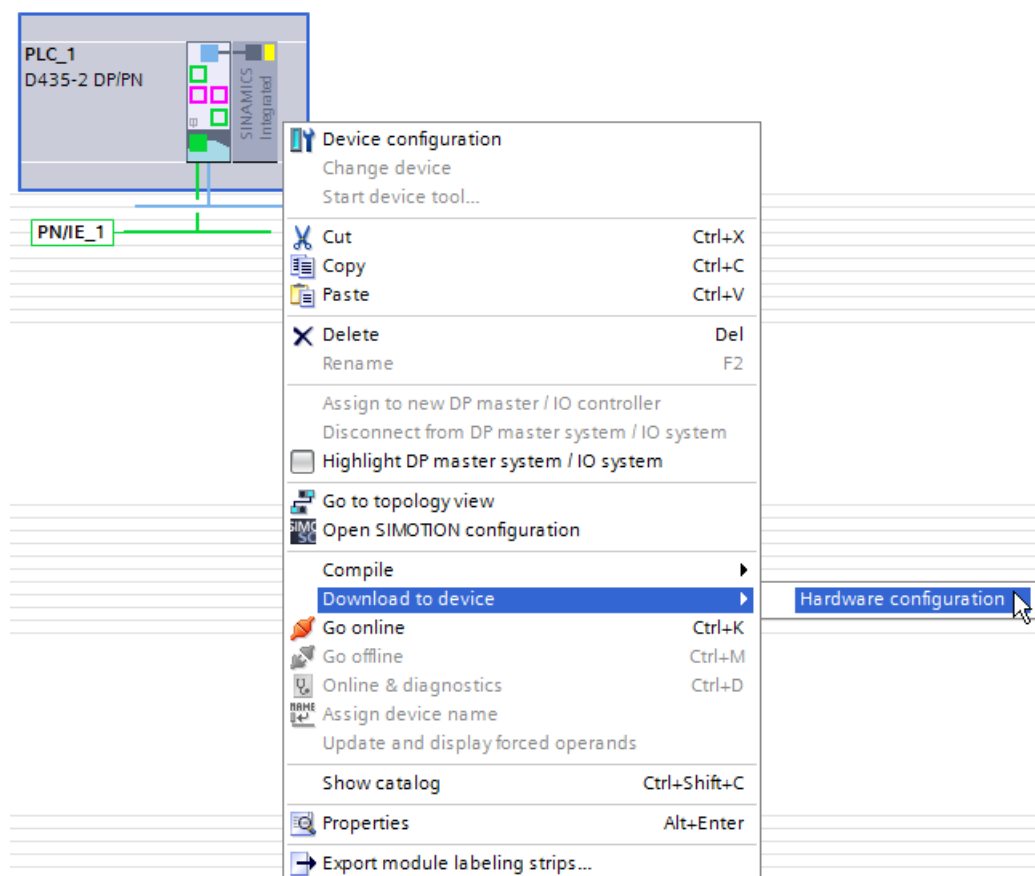


図 5-12 ハードウェアのロード

## 結果

現在のハードウェアコンフィグレーションをコンパイルし、それをターゲットデバイスにロードしました。

## 関連参照先

オンラインアクセスのコンフィグレーションに関する詳細は、TIA ポータル "Configuring online access" の情報システムを参照してください。

## 5.5 サンプルプロジェクトでの結果

### 新たに作成された SIMOTION デバイス

SIMOTION デバイスおよび PG/PC と SIMOTION デバイス間のネットワーク通信のコンフィグレーションは完了しました。

- 新たに作成された SIMOTION D435-2 デバイスは、プロジェクトビューのプロジェクトツリーに示されます。
- PG/PC は SIMOTION デバイスに Ethernet で接続されます。
- プロジェクトはコンパイルされ、ハードウェアコンフィグレーションが SIMOTION デバイス にロードされます。

サンプルプロジェクトでは、他のハードウェアコンフィグレーションは必要ありません。

## 5.5 サンプルプロジェクトでの結果



## SIMOTION SCOUT TIA を開始

---

### 注記

SIMOTION SCOUT TIA の起動は TIA ポータルからのみ実行できます。

---

TIA ポータルでプロジェクトを作成し、ハードウェアおよび通信のコンフィグレーション後、テクノロジーのコンフィグレーションのために **SIMOTION SCOUT TIA** を開始してください。

### 手順

以下の方法で、**SIMOTION SCOUT TIA** を開始することができます：

#### ポータルビューで

1. "モーション & テクノロジ" 機能をクリックしてください。
2. 2 番目のナビゲーションで [Open SIMOTION configuration] 入力域をクリックしてください。

#### プロジェクトビューで

1. **SIMOTION SCOUT TIA** を "プロジェクト (P)" > "Open SIMOTION configuration" メニューで開始してください。  
または、プロジェクトツリーの **SIMOTION** デバイスの "SIMOTION configuration" メニューエントリをダブルクリックしてください。

---

### 注記

#### TIA ポータルへのオンライン接続を解除

TIA ポータルと **SIMOTION SCOUT TIA** は同時にオンライン接続しないでください。

**SIMOTION SCOUT TIA** をオンライン接続する前に、TIA ポータルでのオンライン接続を必ず解除してください。プログラムを **SIMOTION SCOUT TIA** でコンパイルできるようにするためには、これを行う必要があります。

---

### 基本設定の実行

**SIMOTION SCOUT TIA** では、"Options > Settings..." メニューで様々な基本設定を行うことができます。

設定を行うには、以下の手順に従ってください：

1. "Options" メニューで、"Settings..." コマンドを選択してください。  
"Settings" ダイアログが開きます。
2. 機能のタブに切り替えてください。
3. 設定してください。
4. [OK] で確定してください。

## 関連参照先

SIMOTION SCOUT TIA オンラインヘルプに、それぞれのタブや設定に関する詳細があります。

# ターゲットシステムへのプロジェクトのダウンロード

# 7

## 7.1 概要

### Getting Started の目的

このコンフィグレーション手順で、ドライブのコンフィグレーションのための前提条件を作成します。

- オフラインでプロジェクトをハードディスクに保存します。
- プロジェクトを実行可能なコードにコンパイルします。
- SIMOTION デバイスとのオンライン通信を確立します。
- PG/PC から SIMOTION デバイスにプロジェクトをダウンロードします。

## 7.2 プロジェクトの保存およびコンパイル

SIMOTION SCOUT TIA で作成されたプロジェクトをターゲットシステムにダウンロードできるには、そのプロジェクトが実行可能なコードで保存される必要があります。

コマンド **Save project and compile changes** は、両方の手順を組み合わせます。プロジェクトは、ハードディスクにバックアップされます。SIMOTION SCOUT TIA は、プロジェクト全体で、変更を検索し、変更部のみをコンパイルします。

---

### 注記

一日毎の作業がこのましい場合、コマンド **Save project and compile changes** を使用してください。

"Save" コマンドの複数のタイプは、メニュータイトル **Project** の選択で利用可能です。**Save and compile** のオンラインヘルプにこれに関する情報があります。

---

プロジェクトの保存およびコンパイルは、以下の手順に従ってください：

メニューコマンド **Project > Save and compile changes** を選択、または、ツールバーの関連ボタンをクリックしてください。



### 7.3 選択されたターゲットデバイスに接続 – "Go online"

コンパイル実行は、ワークベンチの **Detail view** に統合されます。情報、警告およびコンパイルエラーはわかりやすいテキストで表示されます。

#### [detail view] の起動

**Detail view** はオフになる場合があります。ビューを有効化するために、メニュー項目 **View > Detail view** をクリックしてください。

## 7.3 選択されたターゲットデバイスに接続 – "Go online"

プロジェクトデータを SIMOTION SCOUT TIA プロジェクトからハードウェアにダウンロードする、または、逆にハードウェアから SIMOTION SCOUT TIA プロジェクトに転送するには、PG/PC と SIMOTION デバイス間の通信が有効化されていなければなりません。

ネットワークの通信状況はワークベンチのフッター部分に表示されます:

**Online mode**

ネットワーク通信がオンになりました。データは交換可能です。

**Offline mode**

ネットワーク接続はオフです。

以下の手順で、オンライン接続してください

1. メニューコマンド **Project > Connect with selected target devices** を選択、または、ツールバーの関連ボタンをクリックしてください。



コマンドが初めて呼び出される場合、SIMOTION SCOUT TIA は、**Target Device Selection** ダイアログボックスを開きます。ダイアログで、SIMOTION SCOUT TIA が接続されるデバイスの個別の選択がイネーブルされます。

#### 注記

速やかに SCOUT TIA で作業するために、現在必要なデバイスのみを選択することを推奨します (例：ドライブの試運転が完了している場合は SINAMICS ドライブの選択を解除)。

**Target system > Select target devices...** のメニューに設定があります。デバイスの選択 / 選択解除はオンラインで、

デバイス上のショートカットメニュー **Connect to target device** で実行できます。

2. ダイアログボックスで、コンフィグレーションされた SIMOTION デバイス **PLC\_1** および統合されたドライブ **SINAMICS\_Integrated\_1** を選択してください。

### 7.3 選択されたターゲットデバイスに接続 – "Go online"

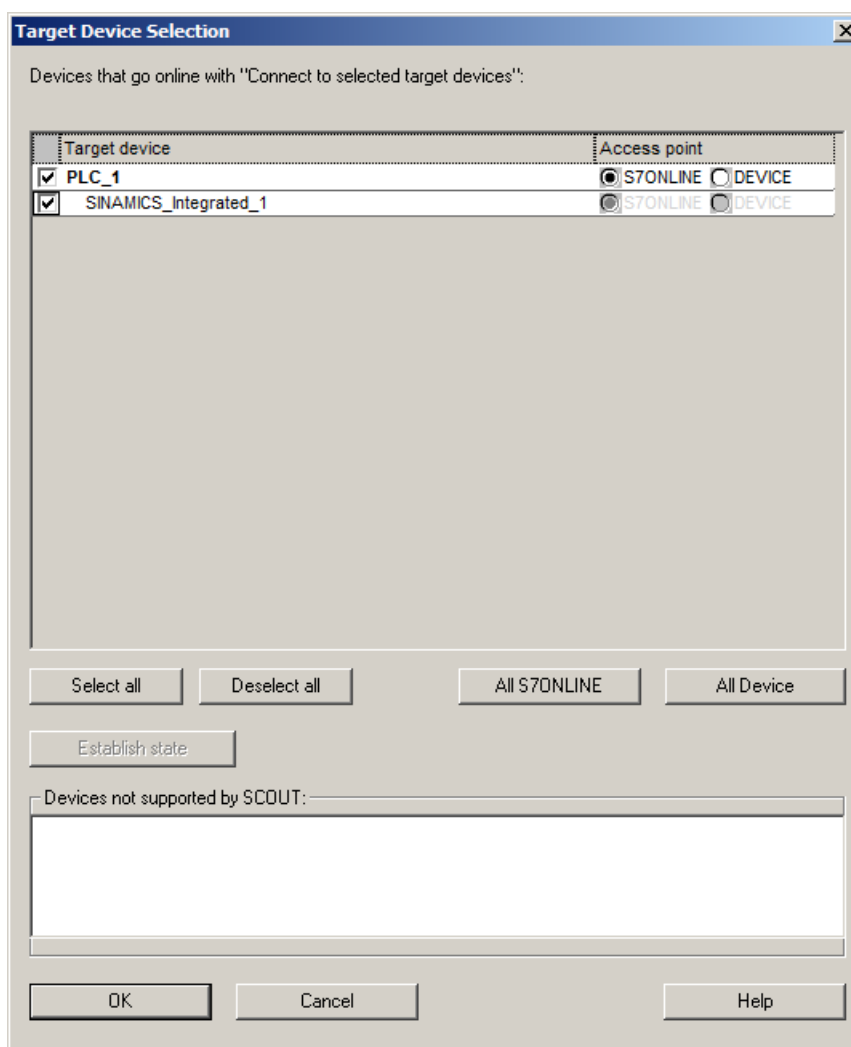


図 7-1 ターゲットデバイス選択ダイアログ

#### 注記

アクセスポイント **S7ONLINE** のプロパティは TIA ポータルで定義されます。このアクセスポイントで、TIA ポータルでコンフィグレーションしたインターフェース経由でオンライン接続します。

**DEVICE (STARTER)** のアクセスポイントで、並列、または、**S7ONLINE** に代わって、直接デバイスに、例えば **Ethernet** インターフェースを介して **SIMOTION SCOUT TIA** 接続するオプションが提供されます。**SIMOTION SCOUT TIA** でコンフィグレーションされたプロパティは、**DEVICE** アクセスポイントにのみ適用されます

### 7.3 選択されたターゲットデバイスに接続 – "Go online"

3. **OK** をクリックしてください。  
SIMOTION SCOUT TIA はオンライン接続を確立します。
4. 詳細域の **Target system output** タブで完了を確認するために一貫性チェックを実行してください。

#### 注記

ハードコンフィグレーションがまだロードされていない場合、SIMOTION SCOUT TIA は初回のオンライン接続時に、**SINAMICS Integrated** へのアクセスが不可能ということを示します。**SINAMICS Integrated** へのアクセスは、ハードウェアコンフィグレーションがロードされた場合にのみ可能です (AUTOHOTSPOT を参照。プロジェクトのコンパイルおよびハードウェアコンフィグレーションの **SIMOTION** デバイスへのダウンロード)。メッセージを閉じてください。

プロジェクトツリーで、**SIMOTION** デバイスおよび統合された **SINAMICS** ドライブのコネクタ記号が変わります。**SIMOTION** デバイスのコネクタ記号のみが部分的に緑になります。緑は、オンライン接続が存在していることを示します。

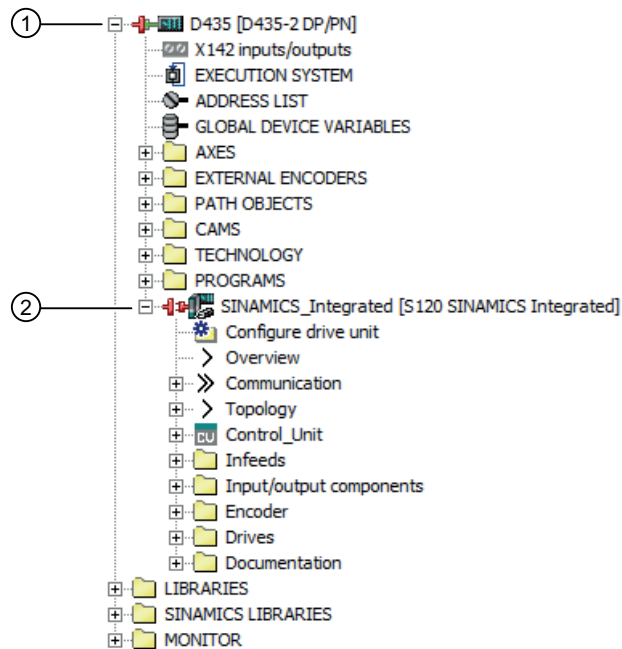


図 7-2 最初のオンライン化後のプロジェクトツリー

#### 関連参照先

*SIMOTION SCOUT TIA Configuration Manual* に、**DEVICE** アクセスポイントのコンフィグレーションに関する詳細があります。

## 7.4 ターゲットシステムへのプロジェクトのダウンロード

完全なプロジェクトのダウンロードは **STOP** モードでのみ可能です。必要に応じて、**SIMOTION SCOUT TIA** は、ダウンロード手順中に運転モードの変更ができます。

---

### 注記

プロジェクトのダウンロードは、**SIMOTION SCOUT TIA** でのコンフィグレーションやプログラミングの変更を行なう際に必要です。.

---



プロジェクトのダウンロードは、以下の手順に従ってください

1. メニューで **Project > Download to target system** を選択する、または、ツールバーの **Download project to target system** ボタンをクリックしてください。



ダイアログ **Download to target system** が現れます。

2. チェックボックス **After loading, copy RAM to ROM** を有効化してください。これは、SIMOTION デバイスの RAM を SIMOTION デバイスのメモ리카ード (ROM) に保存します。こうして、電源切 / 入後に、このコンフィグレーションは保持されます。
3. **Yes** でダウンロードを開始してください。  
プロジェクトデータとハードウェアコンフィグレーションのデータがターゲットシステムの RAM にダウンロードされます。  
CPU が STOP に切り替えられるかどうか質問される場合、**Yes** で確定してください。  
SIMOTION SCOUT TIA は、詳細域の **Target system output** タブに記録される数多くの確認を実行します。最後の入力域 **Download to target system completed successfully** が表示されます。

#### 注記

ターゲットシステムへの最初のダウンロードで、テクノロジパッケージのデータもダウンロードされます。この操作には数分かかる場合があります。

#### 注記

CF カードおよび SINAMICS コンポーネント (ラインモジュール、モータモジュール、増設 I/O モジュールなどの DRIVE-CLiQ コンポーネント) のファームウェアバージョンに依存し、コンポーネントのファームウェアは自動的にアップグレードまたはダウングレードされます。

アップデートには数分かかり、SIMOTION SCOUT TIA の出力ウィンドウの関連するメッセージで示されます。

DRIVE-CLiQ コンポーネントのファームウェアアップデートは、RDY LED の赤緑の点滅で発信されます：

- ファームウェア更新中：RDY LED はゆっくりと点滅します (0.5 Hz)
- ファームウェアの更新終了：RDY LED は速く点滅します (2 Hz)、POWER ON が必要です

これらの点滅パターンは、SIMOTION D の黄色い RDY LED で追加的に表示されます。これは SIMOTION D に接続されたコンポーネントがファームウェアの更新を実行している、または、すべてのコンポーネントでファームウェア更新が完了したことを記します。

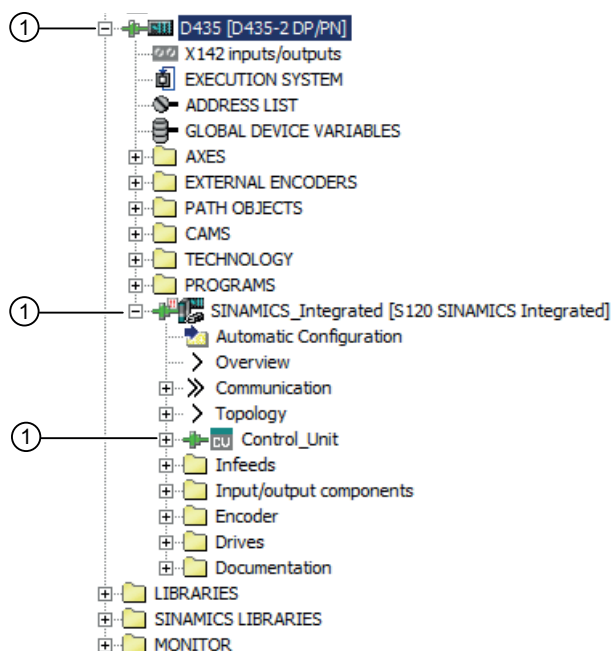
ファームウェアアップデートに続き POWER ON を要求するコンポーネントは、RDY LED の速い点滅でこれを示します。SIMOTION SCOUT TIA との接続をオフラインにし、該当するコンポーネントの 24 V 電源を切 / 入 (POWER ON) してください。

## 7.4 ターゲットシステムへのプロジェクトのダウンロード

### ドライブとの自動的に確立された接続

SIMOTION SCOUT TIA は自動的にプロジェクトダウンロードに続き、直ちにドライブへのオンライン接続を確立します。

プロジェクトツリーで、SIMOTION デバイス、統合された SINAMICS ドライブ、そして、SINAMICS コントロールユニットのコネクタ記号が変わります。コネクタ記号は完全に緑です。これは PG/PC のプロジェクトデータがターゲットシステムのプロジェクトデータと同一であることを示します。



- ① 緑のコネクタシンボル：要素はオンラインモードです。PG/PC のプロジェクトデータは、ターゲットシステムに保存されたプロジェクトデータと同じです。

図 7-3 プロジェクトツリー

この時点から、PG/PC でドライブにオンラインでアクセスできます。

## ドライブのコンフィグレーション

### 8.1 概要

#### Getting Started の目的

Getting Started の本項では、SIMOTION D435-2 デバイスの統合されたドライブをコンフィグレーションします。このために、自動コンフィグレーションを使用します。

#### 要件

- サンプルプロジェクトをターゲットシステムにダウンロード完了。ターゲットシステムへのプロジェクトのダウンロード (ページ 51) を参照。
- SIMOTION SCOUT TIA は、SIMOTION デバイスや統合されたドライブとオンライン接続されています (緑の接続記号)。
- SINAMICS ドライブコンポーネント、電源装置およびパワーユニット並びにモータおよびエンコーダは、DRIVE-CLiQ を介して、SIMOTION コントロールユニットに接続されています。自動コンフィグレーションのこの要件は、Getting Started の冒頭部分にあります。AUTOHOTSPOT を参照。

#### ドライブ

モータを制御するための速度および電流制御機能はドライブで実装されます。

#### 自動ドライブコンフィグレーション

SIMOTION SCOUT は、SINAMICS ドライブコンポーネントの電子銘板を DRIVE-CLiQ インターフェース経由で読み取り、このデータを使用してドライブを自動的にコンフィグレーションできます。従って、このデータは手動で入力する必要がありません。ここで示される SIMOTION D435-2 デバイスのサンプルコンフィグレーションでは完全な DRIVE-CLiQ 配線が必要とされます。

自動ドライブコンフィグレーションの選択肢として、D4x5-2 デバイスをオフラインでもコンフィグレーションできます。オンラインコンフィグレーションは、SIMOTION C および SIMOTION P の Getting Started のオンラインヘルプバージョンで説明されます。

## 8.2 ドライブの自動コンフィグレーション

自動コンフィグレーションを開けるには、以下の手順に従ってください

ドライブ SINAMICS\_Integrated のプロジェクトナビゲータで、**Automatic Configuration** をダブルクリックしてください。

**Automatic Configuration** ダイアログが開きます。

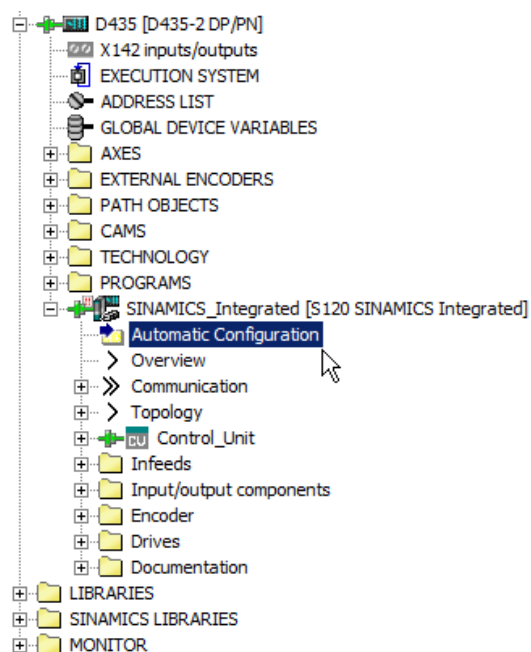


図 8-1 プロジェクトツリー 自動コンフィグレーションの呼び出し

### 注記

プロジェクトツリーで、要素 **Automatic Configuration** は、そのプロジェクトをターゲットシステムにダウンロードし、SIMOTION SCOUT TIA が SIMOTION デバイスと共にオンラインである場合にのみ表示されます。プロジェクトは一貫していなければなりません；ターゲットシステムへのプロジェクトのダウンロード (ページ 51) を参照してください。

以下の方法で、ドライブの自動コンフィグレーションを行います

1. **Automatic Configuration** ダイアログで **Configure** ボタンをクリックしてください。
2. 出荷時設定へのリセットに関するセキュリティチェックを **Yes** で確定してください。  
セキュリティチェックは、ドライブユニットが「最初の試運転」状態ではない場合に現れます。

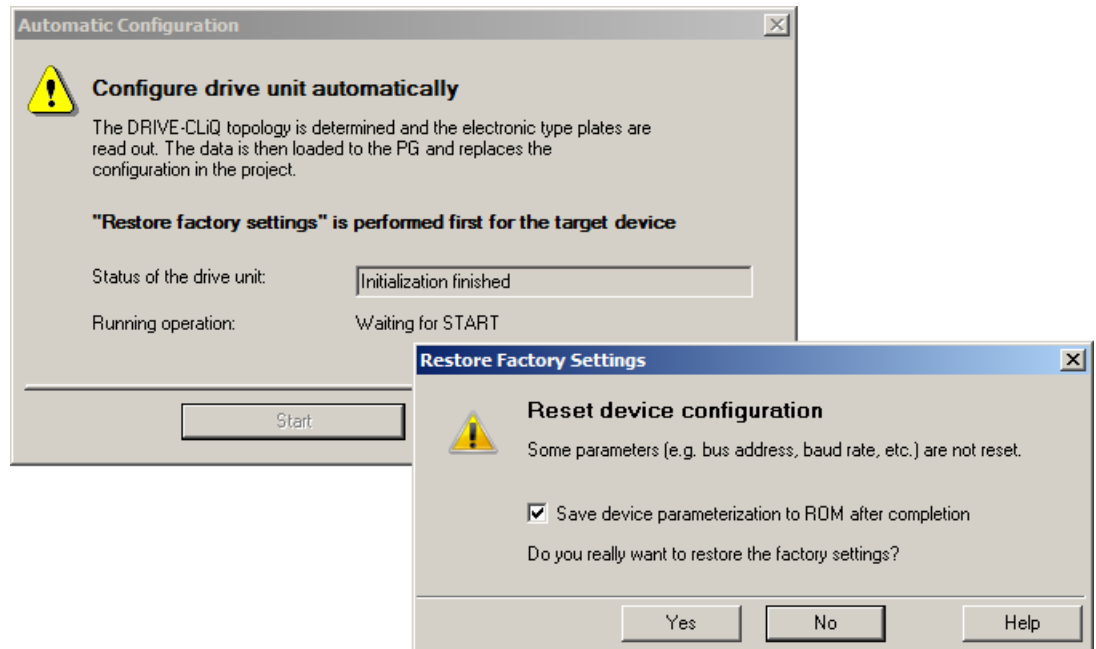


図 8-2 自動ドライブコンフィグレーションを開始し、出荷時設定にリセット

3. **Automatic Commissioning** ダイアログで、サーボまたはベクトルタイプのドライブオブジェクトを使用しているかどうかを指定することができます。  
**Servo** を選択してください。
4. **Automatic Commissioning** ダイアログで **Create** ボタンをクリックしてください  
自動コンフィグレーションが開始されました。

#### 注記

##### ファームウェアアップデート

DRIVE-CLiQ コンポーネントのファームウェアバージョンが CF カードのそれと異なる場合、ファームウェアのアップデートが自動でこの時実行されます。

この場合、以下の手順に従ってください:

- この手順が終了するのを待機してください。これには数分間かかる場合があります。
- オフラインにしてください。
- **SIMOTION** 電源切 / 入を実行してください。

5. ファームウェアのアップデートが実行されない場合、プロジェクトツリーで変更を確認できるようにオンラインを維持してください。

## 8.2 ドライブの自動コンフィグレーション

6. 統合されたドライブのプロジェクトツリーの **Infeeds** システムフォルダーを開いてください。
  - DRIVE-CLiQ インターフェース付き電源装置を使用している場合、自動コンフィグレーションがそこに電源装置を作成します。
  - DRIVE-CLiQ インターフェースなし電源装置を使用している場合、フォルダは空です。電源装置は、このプロジェクトでは判明していません。  
この方法に至る他のコンフィグレーション要件は次の項目 電源装置のコンフィグレーション (ページ 65) で扱われます。
7. 統合されたドライブのプロジェクトツリーの **Drives** システムフォルダーを開いてください。フォルダには、自動コンフィグレーションで検出されたドライブが含まれます。

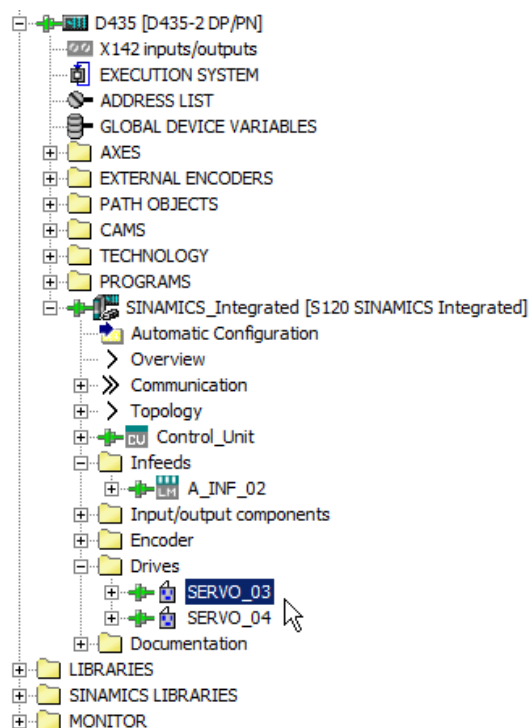


図 8-3 プロジェクトナビゲータ、コンフィグレーションされた電源装置およびドライブ

8. オフラインにしてください。  
ツールバーの **Disconnect from target system** ボタンをクリックしてください。



ワークベンチのフッター部分に **Offline mode** が表示されます。PG/PC と SIMOTION デバイス間のネットワーク通信が切断されました。

### ドライブ名の変更

自動コンフィグレーションでもドライブのオブジェクト名を割り付けます；この例では、SERVO\_03。

名称は後で変更することができます。変更はオフラインモードでのみ可能です。

以下の手順に従ってください：

1. 右クリックでドライブのショートカットメニューを開いてください。そこで **Rename** コマンドを選択してください。
2. 新しい名称を割り付けてください。この後、**OK** で確定してください。

オフラインのプロジェクトで実行されたすべての変更と同じく、名称変更がプロジェクトを一貫性のないものにします。"Project on the PG/PC" と "Project on the SIMOTION device" の一貫性をリセットするために、プロジェクトの再ダウンロードが必要です。この手順は ターゲットシステムへのプロジェクトのダウンロード (ページ 51) で説明されています。

## 8.3 サンプルプロジェクトでの結果

DRIVE-CLiQ インターフェース付き電源装置を使用している場合、ドライブは運転準備が完了しています。それを 1 軸と接続することができます。

### 8.3 サンプルプロジェクトでの結果



## 電源装置のコンフィグレーション

### 9.1 概要

#### Getting Started の目的

Getting Started の本項では、電源装置をコンフィグレーションします。

統合された SINAMICS コントロールユニットは、電源装置が準備完了の場合のみ、ドライブを始動します。プロジェクトは、そのため、電源装置の準備完了信号をドライブが受信するインターフェースを把握する必要があります。

ここでは、2 つのケースを区別する必要があります：

- **DRIVE-CLiQ インターフェース付き電源装置**  
DRIVE-CLiQ インターフェース付き電源装置が既に作成されている場合で、ドライブが挿入されている場合 (電源装置として同じドライブユニットに取り付けられたドライブにのみ適用) に、電源装置の準備完了信号 (r0863.0) は、自動的にドライブの "Infeed operation, p864" に接続されます。  
直ちに次のコンフィグレーション手順を続けることができます。軸のコンフィグレーション (ページ 69) を参照。
- **DRIVE-CLiQ インターフェースがない電源装置**  
DRIVE-CLiQ インターフェースがない電源装置、例えば、スマートラインモジュールを使用している場合、端子経由で電源装置の準備完了信号を接続する必要があります。以下の項目でこの手順が説明されます。

#### 要件

- SIMOTION D435-2 デバイスの統合されたドライブをコンフィグレーション完了。ドライブのコンフィグレーション (ページ 59) を参照。
- SIMOTION SCOUT TIA はオフラインモード。

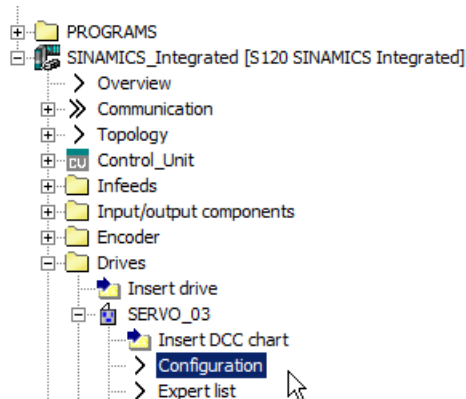
## 9.2 DRIVE-CLiQ インターフェースがない電源装置のコンフィグレーション

DRIVE-CLiQ インターフェースがない電源装置は、出力端子を介して、準備完了信号 (p0863.0) を提供します。プロジェクトで、統合された **SINAMICS** コントロールユニットの入力 (r0722) で、どの信号が有効であるのかを指定します。電源装置から給電されるドライブは、準備完了信号として、この信号を使用します (p0864)。

以下の方法で、電源装置の準備完了信号を接続します

サンプルプロジェクトで、電源装置の準備完了信号 (terminal "DO:Ready" of the infeed) は、D435-2 の DI 0 に配線されます。

1. ドライブのコンフィグレーションダイアログを開いてください。これを行うには、ドライブの下プロジェクトツリーの **Configuration** をダブルクリックしてください。



2. 作業領域で、黄色い背景の **Configure DDS** ボタンをクリックしてください。

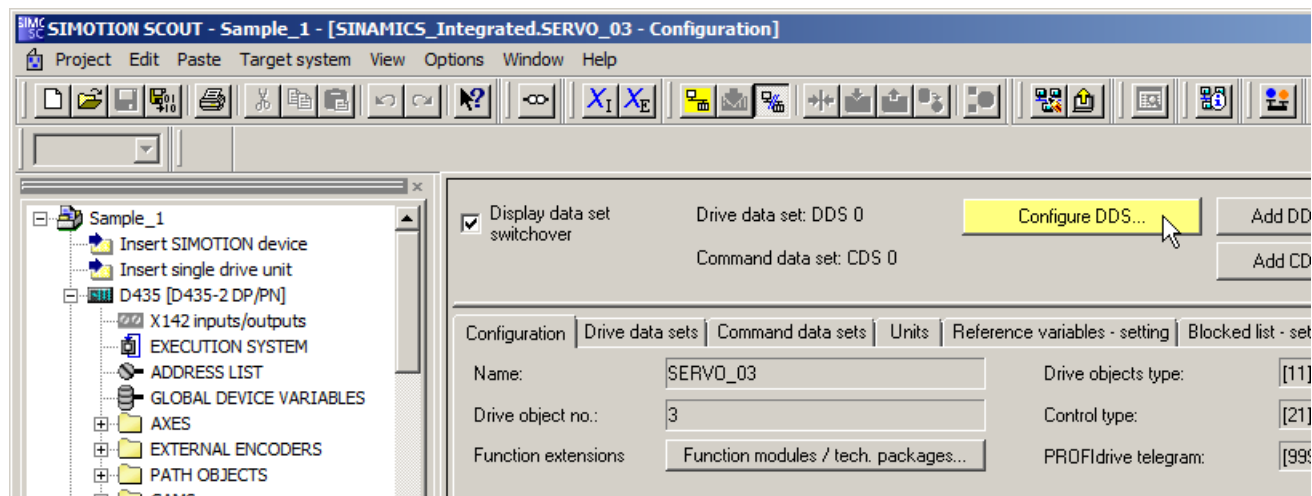


図 9-1 Configure DDS (DDS のコンフィグレーション)

ドライブウィザードが開きます。

## 9.2 DRIVE-CLiQ インターフェースがない電源装置のコンフィグレーション

3. ダイアログ **Configuration - SINAMICS\_Integrated - Power Unit BICO Technology** が現れるまで、ドライブウィザードの **Next** をクリックしてください。
4. **Power unit BICO** ダイアログで、**p0864** を入力し、電源装置の準備完了信号が配線されているデジタル入力を選択してください (例 : DI 0)。

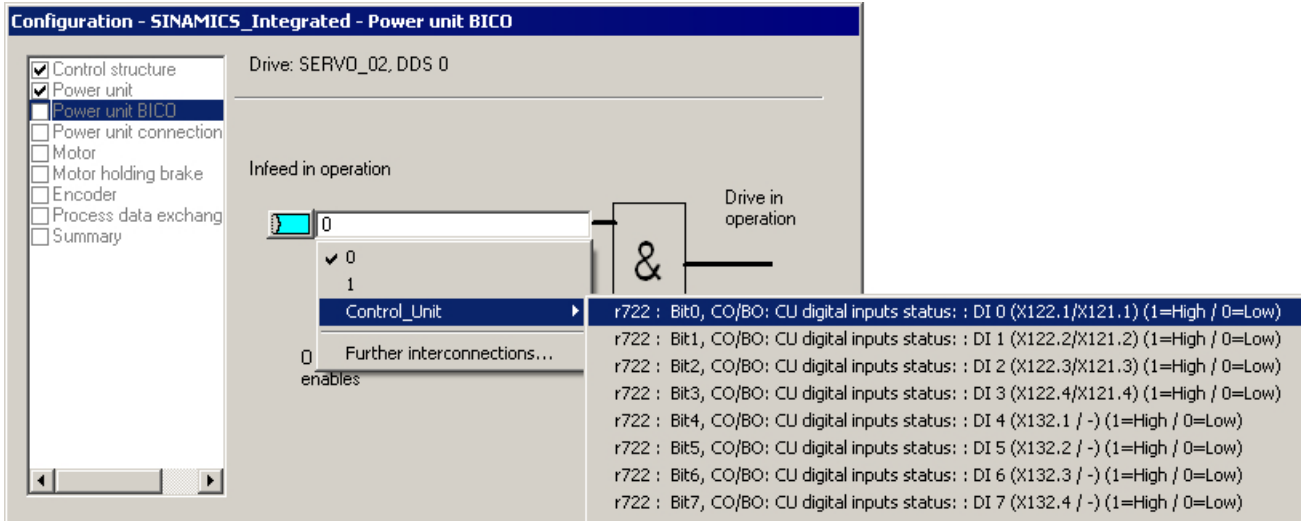


図 9-2 ドライブウィザードの BICO 接続

5. **Next** をクリックしてください。最後の **Summary** ダイアログまで、すべてのダイアログを変更せずに進んでください。
6. **Finish** をクリックしてください。  
これで、コンフィグレーションは完了です。
7. ドライブのコンフィグレーションダイアログはもはや必要とされません。ダイアログボックスの右下の **Close** をクリックしてください。



## 軸のコンフィグレーション

### 10.1 概要

#### Getting Started の目的

Getting Started の本項では、プロジェクトで軸の作成やコンフィグレーション方法、SIMOTION D デバイスの統合デバイスとの接続方法が示されています。SIMOTION SCOUT TIA は、この目的で軸ウィザードを提供します。

#### 要件

SIMOTION D デバイスの統合されたドライブのコンフィグレーション完了。ドライブのコンフィグレーション (ページ 59) を参照。

#### 軸のテクノロジーオブジェクト

テクノロジーオブジェクト (TO) は、コントローラ内の関連するリアルなオブジェクト (例えば、位置決め軸) を表します。

軸のテクノロジーオブジェクトは、ユーザにドライブやエンコーダ (アクチュエータおよびセンサ) の技術ビューを提供し、実際のハードウェアインターフェースを隠します。

軸のテクノロジーオブジェクトには、ドライブとの通信、実績値処理、位置制御、位置決め機能など、広範な機能が含まれています。これにより制御および動作コマンドが実行され、ステータスおよび実績値が表示されます。

軸およびエンコーダの技術的制限および機械的値 (例：送りねじピッチおよびギア) が軸で設定されます。これで、技術的変数だけを使用して作業することができます。

#### 軸ウィザード

SIMOTION SCOUT は、新しい軸の作成のために、軸ウィザードを提供します。ウィザードは、基本設定をスキャンし、TO 軸をドライブに接続します。


このウィザードは一度だけ実行できます。コンフィグレーションを後で変更する場合にも、TO の該当するダイアログを開けることができます。

## 10.2 軸の作成

サンプルプロジェクト用に、軸 **Axis\_1** を作成してください。項目 ドライブのコンフィグレーション (ページ 59) でコンフィグレーションしたドライブに軸を割り付けてください。

ウィザードを実行し、**Next** ですべての標準設定を確定してください。

プロジェクトでリアル軸を作成するには、以下の手順に従ってください

1. プロジェクトナビゲータの **AXES** フォルダを開いてください。
2.  **Insert axis** をダブルクリックしてください。Insert axis ダイアログボックスが現れます。
3. **Name** 域で軸に名称を付けてください。サンプルプロジェクトの軸に **Axis\_1** の名称を使用してください。
4. プリセットされた軸テクノロジーをデフォルト **Positioning** のままにしてください。

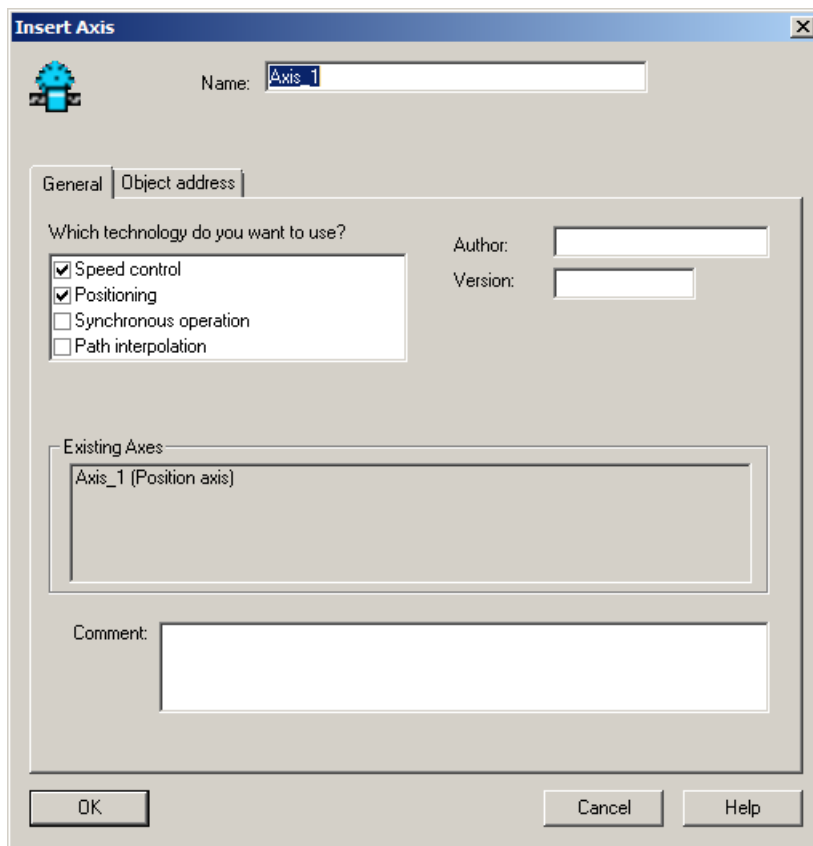


図 10-1 軸のウィザード、リアル軸の作成

5. **OK** をクリックしてください。  
軸のコンフィグレーションウィザードが現れます。

6. 軸タイプを定義してください。  
サンプルプロジェクトのために、電動ドライブに接続されているリニア軸をなるべく選択してください。

**linear** および **electrical** 域を有効化してください。

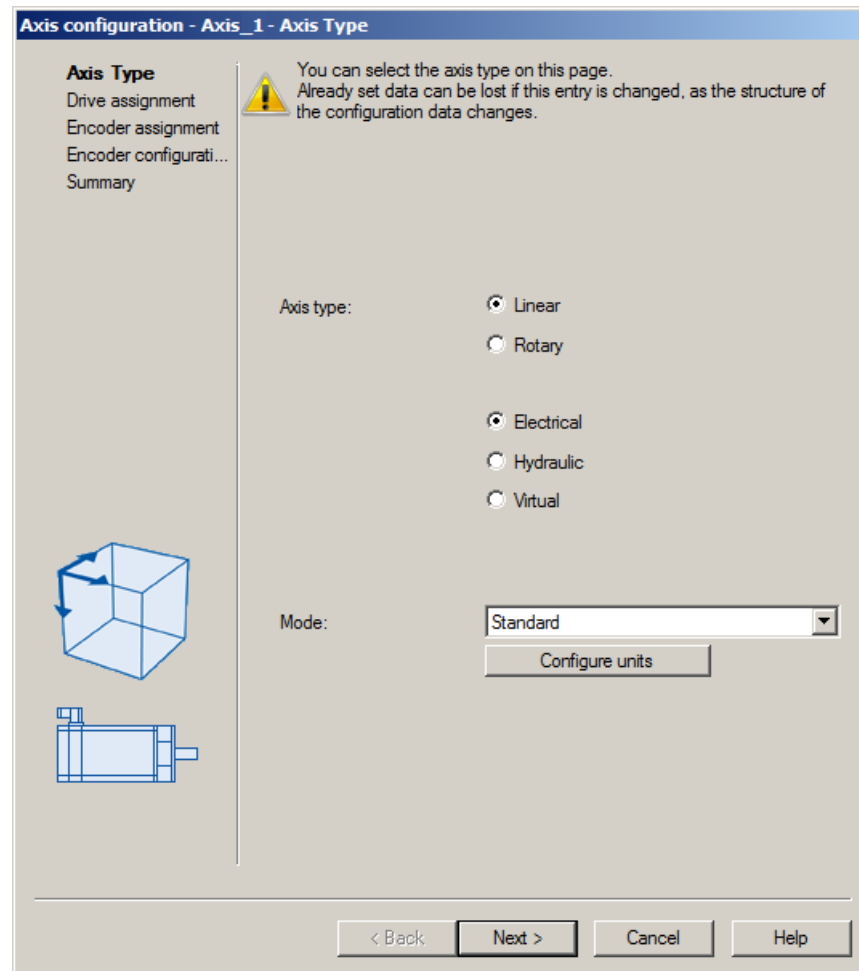


図 10-2 軸のウィザード、軸タイプの決定

7. **Next** をクリックしてください。  
**Drive assignment** ダイアログボックスが現れます。

## 10.2 軸の作成

8. 項目 **ドライブのコンフィグレーション** (ページ 59) でコンフィグレーションしたドライブに軸を割り付けてください。

これを行うには、**Assignment partner** 列のドライブユニットをクリックしてください。ドライブオブジェクトの接続を希望するドライブオブジェクトの下ツリーをクリックしてください。**Assign** は、**Assignment** 列に表示されます。軸 / ドライブの割り付けはこのように定義されます。

**Next** をクリックしてください。

## 注記

軸を既にコンフィグレーションされているドライブに割り付ける選択肢として、軸ウィザードが 2 つの他の選択オプションを提供します：

- 後での軸 / ドライブの割り付けを定義：  
軸が作成されますが、ドライブには割り付けられません。軸のプログラミングおよびシミュレーションもここで可能です。
- ドライブを作成：  
ドライブウィザードは軸ウィザードから呼び出すことができます (オフラインコンフィグレーション)。こうして、軸はドライブと共に 1 つの段階で作成し、ドライブに割り付けることができます。

これ以外の手順は、**Getting Started** では取り上げていません。

9. モータエンコーダを割り付けてください。

**Encoder assignment** ウィンドウで、標準設定として、ドライブに接続されたモータエンコーダを使用します。

**Next** をクリックしてください。

10. 軸ウィザードは、一覧でコンフィグレーションデータを集めます。こうして、プロジェクトでデータを受け付ける前に、それを確認し、変更する機会があります。

**Finish** でダイアログを閉じてください。

コンフィグレーションされた軸はプロジェクトナビゲータで表示されます。

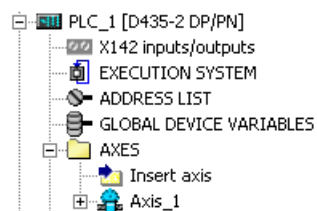


図 10-3 軸が作成されました

## エンジニアリングシステムの自動設定

エンジニアリングシステムは自動的に通信に必要な PROFIdrive 軸、並びに使用されるアドレスを定義します。

同様に、テレグラムは、選択された TO テクノロジー (例 : SINAMICS Safety Integrated) に応じて、ドライブで拡張され、接続が自動的に作成されます。

SINAMICS S120 のドライブおよびエンコーダデータ、並びに参照変数、最大変数、トルクリミットやトルク低減における細かさは、SIMOTION テクノロジーオブジェクト



## 10.3 軸のコンフィグレーションをターゲットシステムにダウンロード

"TO axis" および "TO external encoder" のコンフィグレーションのために、自動的に受け付けられます。このデータは SIMOTION に入力する必要がありません。

## 他の軸のコンフィグレーション

他の軸のコンフィグレーションは、軸の下のプロジェクトナビゲータでアクセス可能なダイアログで可能です。サンプルプロジェクトでは、他のコンフィグレーションは必要ありません。

## 軸のコンフィグレーションをターゲットシステムにダウンロード

- プロジェクトを保存し、コンパイルしてください。
- 次のコンフィグレーション手順で軸の機能を試験するために、ターゲットシステムに合わせた軸のコンフィグレーションを含むサンプルプロジェクトをダウンロードしてください。

## 10.3 軸のコンフィグレーションをターゲットシステムにダウンロード

次のコンフィグレーション手順で軸の機能を試験するために、ターゲットシステムに合わせた軸のコンフィグレーションを含むサンプルプロジェクトをダウンロードしてください。

## 手順

1. プロジェクトの保存および変更部のコンパイル。
2. オフラインの場合、オンライン接続の確立。
3. ターゲットシステムへプロジェクトをダウンロード。

## 変更の保存およびコンパイル

ツールバーの **Save project and compile changes** ボタンをクリックしてください。



## オンライン接続の確立

ツールバーの **Connect to selected target devices** ボタンをクリックしてください。



### 10.3 軸のコンフィグレーションをターゲットシステムにダウンロード

#### ターゲットシステムにダウンロード

ツールバーの **Download project to target system** ボタンをクリックしてください。



## 軸コントロールパネルでの軸の試験

### 11.1 概要

#### Getting Started の目的

"Getting Started" の本項では、コンフィグレーションされた軸を試験します。この目的のために、SIMOTION SCOUT TIA は軸コントロールパネルを提供します。

#### 要件

- 電源装置のコンフィグレーション完了。電源装置のコンフィグレーション (ページ 65) を参照。
- サンプルプロジェクトでの軸の作成およびコンフィグレーションを完了。軸のコンフィグレーション (ページ 69) を参照。
- 軸のコンフィグレーションを含むプロジェクトが、ターゲットシステムにダウンロード済。軸のコンフィグレーションをターゲットシステムにダウンロード (ページ 73) を参照。
- SIMOTION SCOUT TIA はオンラインモード。

#### 各軸の制御および監視

軸コントロールパネルは、ユーザープログラムの必要なしに個々の軸を制御および監視するために使用されます。ドライブで軸を移動させるために、それを使用することができます。

例えば、軸コントロールパネルを使用して以下のタスクを実行できます：

- プログラムでそれらを移動させる前に軸を試験してください。
- 故障発生時に、コントロールパネルを使って軸の移動が可能かどうかを試験してください。
- 軸のイネーブルを設定し、選択解除してください。

## 11.2 軸コントロールパネルを使った作業

サンプルプロジェクトで、軸の正しい機能を試験するために、JOG モードで設定された軸を移動します。

軸コントロールパネルを開くには、以下の手順に従ってください

プロジェクトナビゲータの **AXES** フォルダを開いてください。軸の下の **Control panel** をダブルクリックしてください。

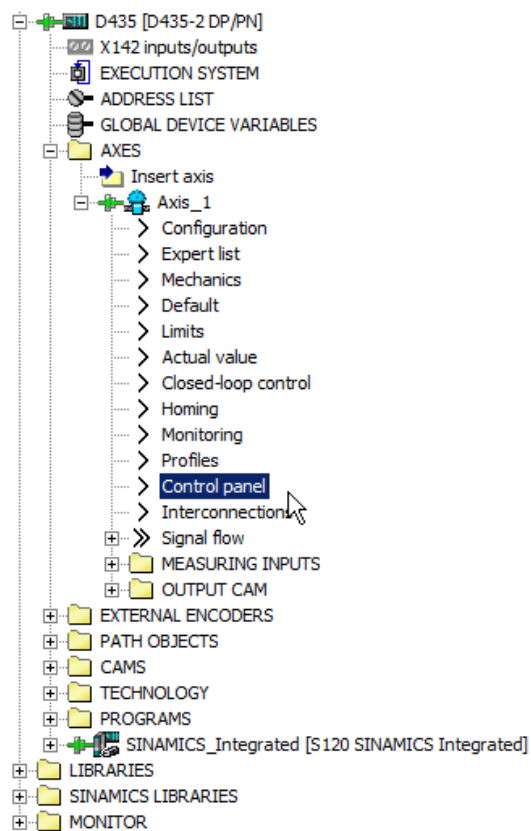
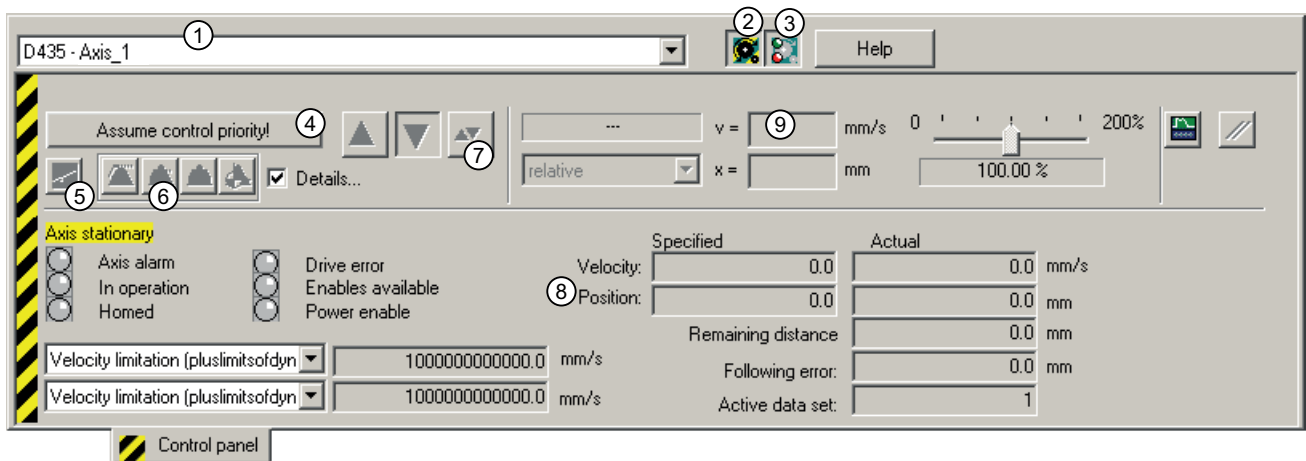


図 11-1 軸コントロールパネルの呼び出し

軸コントロールパネルは **Detail view** で開かれます。



① ... ⑨ 丸数字は以下のテキストで参照されています。

図 11-2 軸コントロールパネル

以下の方法で、軸コントロールパネルを使って軸を移動させます

1. コントロールパネルの左上 ① に現在選択されている軸が示されます。
2. コントロールパネルの表示域 :  
コントロールパネルは、制御域と診断域に区分されます。これらの領域は隠される場合があります。  
関係領域を表示するために、**Show/hide control area** ボタン ② および/または **Show/hide diagnostics area** ボタン ③ をクリックしてください。

## 11.2 軸コントロールパネルを使った作業

3. Assume control priority (制御優先度の想定) – セーフティ規則を遵守 :  
PG/PC から軸を移動させることができるように、PG/PC に制御優先度がある設定を行う必要があります。

**Assume Control Priority** ボタン ④ をクリックしてください。

**Assume Control Priority** ダイアログが開きます。

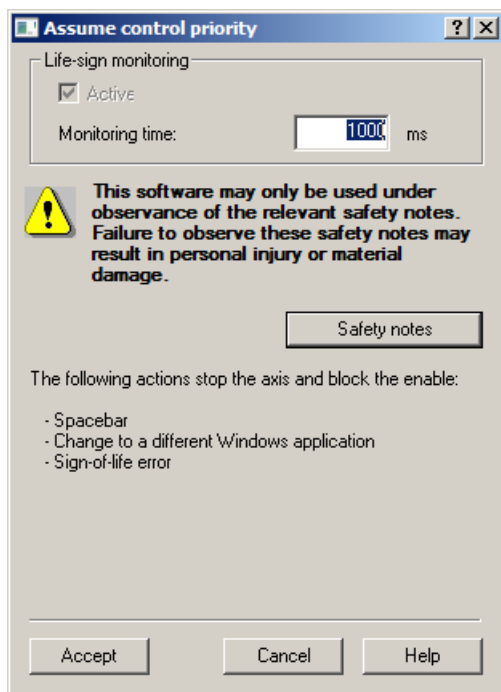


図 11-3 軸コントロールパネル、制御優先度の想定

セーフティ規則を遵守し、**Accept** で確定してください。  
この時 PG/PC には制御の優先順位があります。

### 注記

スペースバーを押すことで、いつでも軸を停止することができます。

4. 軸をイネーブル :  
軸をイネーブルするために、**Set/remove enables** ボタン ⑤ をクリックしてください。



現れる **Switch axis enable** ダイアログを **OK** で確定してください。  
スイッチ **Start motion**、**Stop motion** および **Jog**⑦ がイネーブルされます。

5. 軸の原点復帰 :  
**Home axis** ボタンをクリックしてください。  
ダイアログボックスで、**Homing type Set home position** を選択し、値 **0** を **Home position coordinates** 入力してください。  
**OK** でダイアログボックスを確定し、軸の原点復帰を開始してください。

6. **Position-controlled traversing of the axis** ボタン ⑥ をクリックしてください。



ダイアログ **Start axis position-controlled** が現れます。

軸の速度を指定してください。ダイアログを閉じるために **OK** をクリックしてください。

速度は、表示域 **v=⑨** に表示されます。

7. 軸の動作を開始 :

**Jog⑦** をクリック



そのスイッチを押している間、軸は移動します。**Velocity** および **Position** 域 ⑧ で、移動動作を監視することができます。

軸の試験はこのように実行されます。

8. 軸のイネーブルを無効化 :

**Set/remove enables⑤** をクリックしてください。

**Disable axis** ダイアログを **OK** で確定してください。

#### 注記

軸を再び移動させたい場合、[Alarms] ウィンドウのすべてのアラームを確認してください。

9. 制御の優先度を元に戻す :

**Give up Control Priority④** をクリックして、PG/PC からの軸の制御を無効にしてください。この運転状態では、もはや軸を PG/PC から制御できません。

10. オフラインを実行 :

メニューで **Project > Disconnect from target system** を選択してくださいまたは、ツールバーの **Disconnect from target system** ボタンをクリックしてください。



## 11.3 サンプルプロジェクトでの結果

サンプルプロジェクトの軸を軸コントロールパネルを使用して移動し、正しく機能していることを確認しました。これで、軸のコンフィグレーションは完了です。

### 11.3 サンプルプロジェクトでの結果



## デジタル出力をコンフィグレーション

### Getting Started の目的

Getting Started の本項では、SIMOTION D デバイスで出力として利用可能なデジタル入 / 出力の 2 つをコンフィグレーションします。

後続のコンフィグレーション手順で書くつもののプログラムのための出力が必要です。AUTOHOTPOT を参照。

### 要件

SIMOTION SCOUT TIA はオフラインモード。

### 端子 X142 の I/O チャンネル

サンプルプロジェクトのために、端子 X142 の I/O チャンネルを使用します。

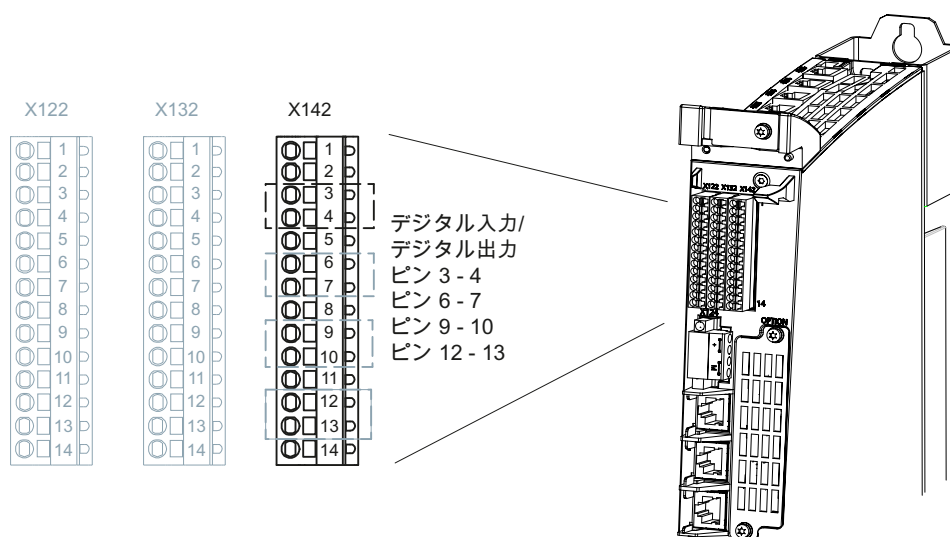


図 12-1 SIMOTION D445-2 DP/PN、デジタルインターフェース X142 の位置

端子 X142 の I/O は恒久的に SIMOTION D4x5-2 に割り付けられます。このように、端子は、SIMOTION デバイスのプロジェクトツリーに表示されます。

デジタル出力を定義するには、以下の手順に従ってください

1. SIMOTION デバイスのプロジェクトツリーで、要素 **Inputs/outputs X142** をダブルクリックしてください。  
**I/O properties** のダイアログボックスが開きます。
2. **Channels 0-7** タブに進んでください。  
端子 **X142** はタブで表示され、ピン番号の指定およびチャンネルの現在の使用状況を表示します。
3. ピン **3** および **4** のチャンネル **0** および **1** は、デジタル入力 **DI 0** および **DI 1** として出荷時設定されています。デジタル出力 **DO 0** および **DO 1** として 2 つのチャンネルを定義してください。それぞれの場合で、**Function** リスト域で、値 **DO** を選択してください。

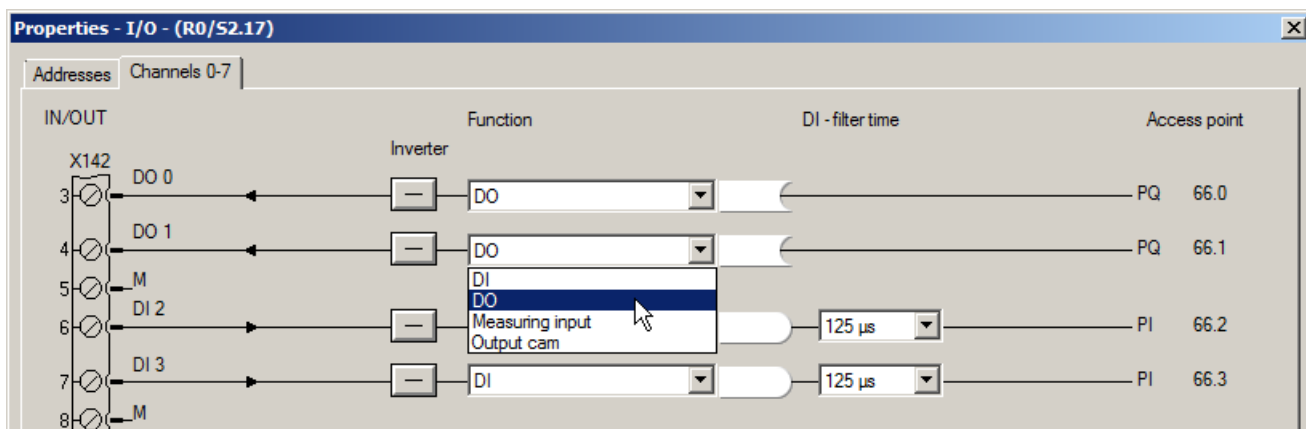


図 12-2 I/O プロパティ

4. **OK** で確定してください。  
デジタル出力のコンフィグレーションはこれで完了です。

# SIMOTION アプリケーションのプログラミング

## 13.1 概要

### Getting Started の目的

この項目では、サンプルプロジェクト用に単純な SIMOTION ユーザプログラムを書きます。

- プログラムで必要とされる変数を作成します。
- SIMOTION SCOUT TIA で提供されるグラフィック表示のエディタを使って、プログラムと追加の補助プログラムを作成します。

### このプログラムの目的

プログラムは、サンプルプロジェクトで設定された軸を制御します。

- 電源装置は、SINAMICS システムからのコールでオンします。
- 軸は、指定された位置に移動し、本来の位置に戻ります。
- このプログラムは "Program started" および "Program end reached" が I/O でそれらを評価するために、デジタル出力に適用されます。

### 要件

- 電源装置のコンフィグレーション完了。AUTOHOTSPOT を参照。
- 軸をコンフィグレーションし、その機能の試験済み。AUTOHOTSPOT および AUTOHOTSPOT を参照。
- プログラムで使用するために 2x デジタル出力を設定済み。AUTOHOTSPOT を参照。
- SIMOTION SCOUT TIA はオフラインモード。

## 13.2 変数

### 13.2.1 変数タイプ

#### 変数タイプ

複数の可変速タイプが SIMOTION で区別されます：

- システム変数

それぞれの SIMOTION デバイスおよびテクノロジーオブジェクトには特別なシステムタグがあります。すべてのプログラムから SIMOTION デバイス内のシステムタグにアクセスできます。

- I/O 変数

I/O は、SIMOTION デバイスまたはその I/O の I/O アドレスに割り付けられた記号的なタグ名です。このため、I/O への直接アクセスは可能です

I/O 変数は、すべてのデバイスで有効です。SIMOTION デバイスのすべてのプログラムで、それらにアクセスできます。

- グローバルデバイス変数、ユニット変数およびローカル変数は、有効性が制限されているユーザ定義の変数です：

SIMOTION デバイスのすべてのプログラムは、グローバルデバイス変数にアクセスできます。

すべてのプログラム、ファンクションブロックや同じユニット内で定義された機能、例えば、ST ソース、MCC ソース、LAD/FBD ソースは、ユニット変数にアクセスできます。キーワード **USES** で他のユニットの変数にアクセスすることもできます。ユニット (ソース) は皆様のプロジェクトで作成できる論理的ユニットで、プログラム、機能およびファンクションブロックを含みます。

ローカル変数はそれらが定義されているプログラム、機能またはファンクションブロック内で唯一アクセスできるものです。

**Variable model** のオンラインヘルプで変数に関する関連情報を入手いただけます。

### 13.2.2 サンプルプロジェクトの変数

Getting Started のサンプルプロジェクト用に以下の変数が必要です：

2x グローバルデバイス変数

- g\_bo\_start
- g\_bo\_ready

2x I/O 変数

- q\_bo\_output0
- q\_bo\_output1

加えて、DRIVE-CLiQ インターフェース付き電源装置の使用時：

2x I/O 変数

- LineModule\_STW
- LineModule\_ZSW

インスタンス

- myFB\_LineControl

### サンプルプロジェクトでの変数の使用

変数 **g\_bo\_start** および **g\_bo\_ready** は冗長にプログラム状態 "Program started" および "Program end reached" を記述します。

I/O 変数 **q\_bo\_output0** および **q\_bo\_output1** は、プログラム状態を、例えば、ディスプレイなどの SIMOTION デバイスのコンフィグレーションされたデジタル出力に適用します。

インスタンス **myFB\_LineControl** および I/O 変数 **LineModule\_STW** および **LineModule\_ZSW** は電源装置を制御します。

### 13.2.3 グローバルデバイス変数の作成

Detail view の **Symbol browser** タブでグローバルデバイス変数を作成します。

グローバルデバイス変数を作成するには、以下の手順に従ってください

サンプルプロジェクトのためにグローバルデバイス変数 **g\_bo\_start** および **g\_bo\_ready** を作成するには、以下の手順に従ってください。

変数の定義には以下が含まれます：

- 変数名称
- BOOL データタイプ

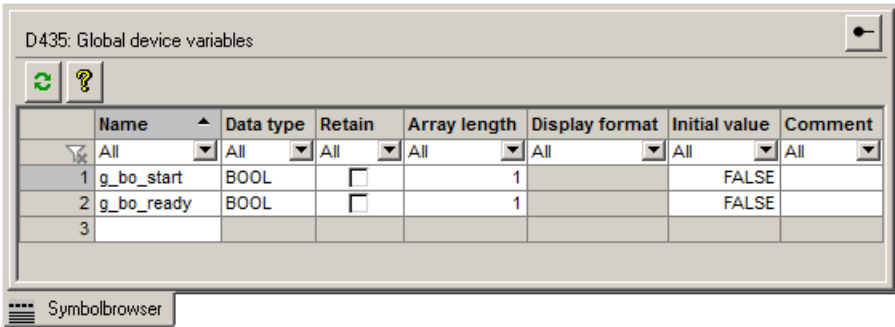
13.2 変数

注記

オフラインモードでのみ、グローバルデバイス変数が作成できます。

1. SIMOTION デバイスのプロジェクトナビゲータにある **GLOBAL DEVICE VARIABLES** 要素をクリックしてください。  
**Global device variables** 表はシンボルブラウザで表示されます。
2. 最初の空のセルの **Name** 列をクリックし、変数名 **g\_bo\_start** を入力してください。  
[RETURN] または [TAB] を押してください。入力カーソルは **Data type** 域にジャンプします。代わりに、その域をクリックして入力カーソルをそこに移動することもできます。
3. **Data type** 域にデータタイプ **BOOL** を入力してください。
4. RETURN を押し、確定してください。  
変数が作成され、プロジェクトで利用可能です。シンボルブラウザで、入力用の新しい空の行が開かれます。
5. 続けて、グローバルデバイス変数 **g\_bo\_ready** を作成してください。

以下の図は完全に定義されたことを示します。



	Name	Data type	Retain	Array length	Display format	Initial value	Comment
	All	All	All	All	All	All	All
1	g_bo_start	BOOL	<input type="checkbox"/>	1		FALSE	
2	g_bo_ready	BOOL	<input type="checkbox"/>	1		FALSE	
3							

図 13-1 サンプルプロジェクトのグローバルデバイス変数

13.2.4 I/O 変数の作成

I/O をコンフィグレーションする際、SIMOTION SCOUT TIA は SIMOTION/SINAMICS コンポーネント上の入出力の記号での割り付けをサポートします。ドライブレベル I/O に I/O 変数でアクセス可能にするには、I/O 変数を I/O チャンネルに割り付けるだけです。SIMOTION SCOUT TIA は自動的にテレグラム、BICO 接続およびアドレスをセットアップします。

コンフィグレーションされたデジタル出力用の I/O 変数を作成するには、以下の手順に従ってください

以下の手順で、サンプルプロジェクト用の I/O 変数 **q\_bo\_output0** および **q\_bo\_output1** を作成します。

変数宣言には以下が含まれます：

- 変数名称
- 出力変数 **OUT** として定義
- **BOOL** データタイプ
- デジタル出力への割り付け

---


#### 注記

I/O 変数はオフラインモードでのみ作成できます。

---

1. SIMOTION デバイスのプロジェクトナビゲータにある要素 **ADDRESS LIST** をダブルクリックしてください。  
**Address list** タブは、**Detail view** で開きます。
2. **Name** 列の最初の空のセルをクリックしてください。変数名称 **q\_bo\_output0** を入力してください。  
[RETURN] または [TAB] を押してください。入力カーソルは **I/O address** 域にジャンプします。代りに、その域をクリックして入力カーソルをそこに移動することもできます。
3. **I/O address** 域にキーワード **OUT** を入力してください。
4. **Data type** 域のデータタイプ **BOOL** を入力する、または、その域を空のままにしておいてください。

## 13.2 変数

5. コンフィグレーションされたデジタル出力を変数に割り付け：
  - **Assignment** セルのボタン  をクリックしてください。割り付けダイアログが開きます。
  - 割り付けダイアログで **D435** 要素を開いてください。
  - **D435** 要素の下でのデジタル出力 **DO\_0** をクリックしてください。Assign は、Assignment 列に表示されます。

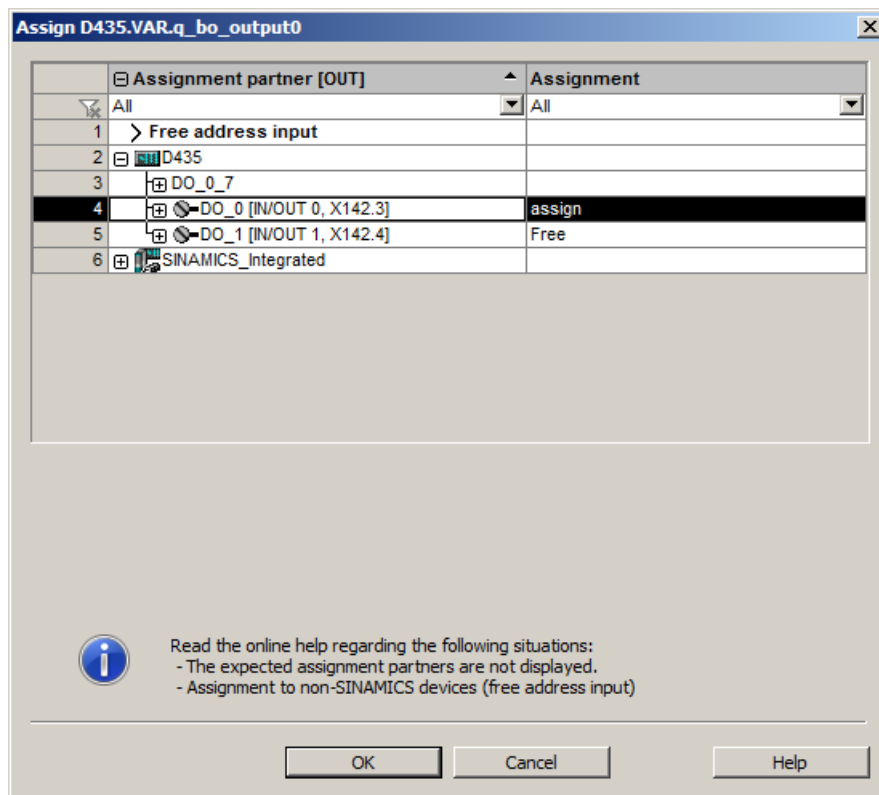
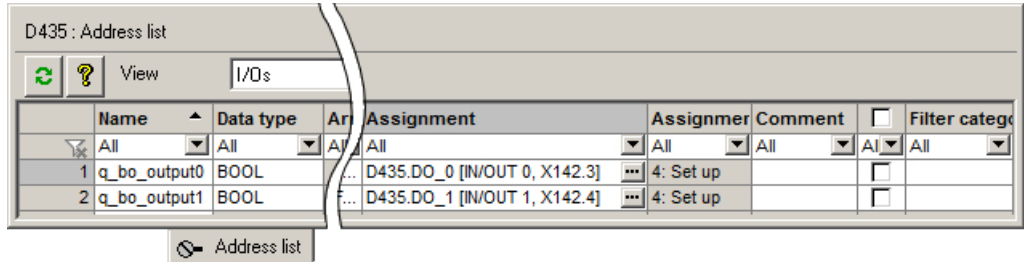


図 13-2 I/O 変数および I/O チャンネルの割り付け

6. **OK** で確定してください。  
 I/O 変数および I/O チャンネルの割り付けは完了しました。I/O 変数が作成され、プロジェクトで使用可能です。**Address list** 表で入力用の新しい空の行が開かれます。



7. 同じ方法で I/O 変数 **q\_bo\_output1** を作成してください。変数をコンフィグレーションされたデジタル出力 DO 1 に割り付けてください。  
下表はすべての宣言を示します：



	Name	Data type	Ar	Assignment	Assignmer	Comment		Filter category
	All	All	All	All	All	All	All	All
1	q_bo_output0	BOOL	...	D435.DO_0 [IN/OUT 0, X142.3]	4: Set up			
2	q_bo_output1	BOOL	...	D435.DO_1 [IN/OUT 1, X142.4]	4: Set up			

図 13-3 サンプルプロジェクトの I/O 変数

8. 他の I/O 変数を宣言するために **Address list** タブを開いたままにしてください。

電源装置の制御のための I/O 変数の作成は、以下の手順に従ってください

DRIVE-CLiQ インターフェース付き電源装置を使用している場合、I/O 変数 **LineModule\_STW** および **LineModule\_ZSW** も作成してください。

注記

I/O 変数はオフラインモードでのみ作成できます。

## 13.2 変数

I/O 変数 **q\_bo\_output0** および **q\_bo\_output1** の定義を以下の手順で行ってください。

1. 表示されない場合、詳細域の **Address list** タブを開いてください。
2. I/O 変数 **LineModule\_STW** を作成してください。
  - I/O アドレス : OUT
  - データタイプ : WORD
  - 割り付け : SINAMICS 電源装置のコントロールワード E\_STW1

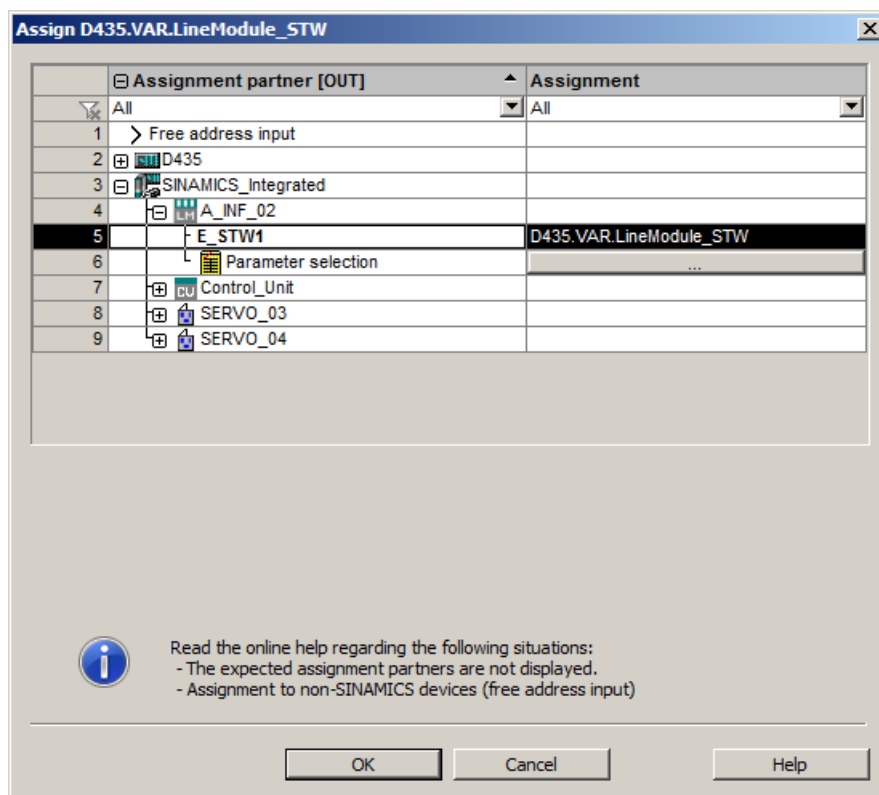


図 13-4 I/O 変数と電源装置のコントロールワードの割り付け

3. I/O 変数 **LineModule\_ZSW** を作成してください。
  - I/O アドレス : IN
  - データタイプ : WORD
  - 割り付け : SINAMICS 電源装置のステータスワード E\_ZSW1

下表はすべての宣言を示します：

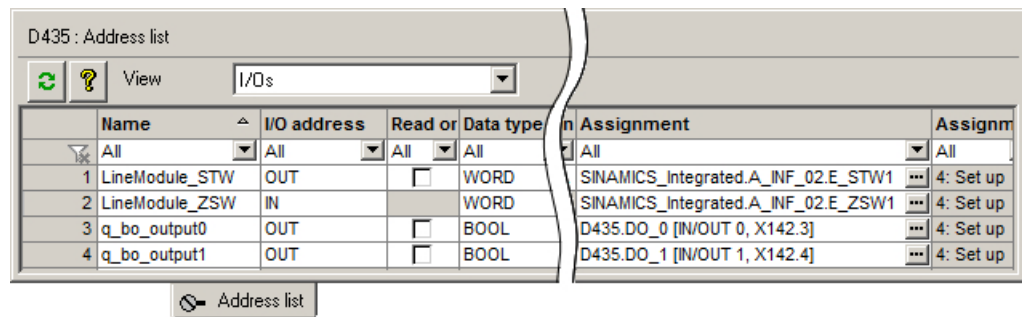
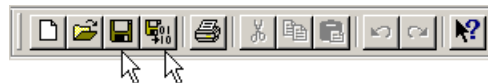


図 13-5 サンプルプロジェクトの I/O 変数

### 13.2.5 このコンフィグレーションをバックアップ

サンプルプロジェクトで作成された変数をバックアップしてください。

これを行うには、ツールバーの **Save project** または **Save project and compile changes** ボタンをクリックしてください。



## 13.3 プログラミング

### 13.3.1 サンプルプロジェクトのプログラミング言語

サンプルプロジェクトの範囲内で、プログラミング言語 **MCC** および **LAD/FBD** で単純なユーザプログラムを作成します。

このために、以前の項目 変数 (ページ 84) で作成した変数を使用します。

### 13.3.2 MCC モーションコントロールチャート

#### 13.3.2.1 プログラミング言語 MCC

MCC (Motion Control Chart) は、グラフィック表示のプログラミング言語です。

### 13.3 プログラミング

プログラムされたモーションシーケンス (マシンシーケンス) は、フローチャート (MCC チャート) として表示されます。フローチャートの構造は、機械装置の実際の操作シーケンスに基づいています。連続的なモーションシーケンスがプログラムされます。




#### SIMOTION SCOUT TIA のシステム

SIMOTION デバイスの **PROGRAMS** フォルダにはプロジェクトで作成された **MCC** ユニットが含まれます。

MCC ユニットには、SIMOTION デバイス で動作する **MCC** プログラムが含まれます。

"MCC プログラム" は、異なるプログラム組織のユニット (POE) のための総称です。

POE は、プログラム、機能またはファンクションブロックです。POE のタイプは、プロジェクトナビゲータのアイコンで表示されます：

-  プログラム
-  機能
-  ファンクションブロック

MCC チャートは、プログラム組織ユニット **POE** のグラフィック表示です。

MCC ユニットには複数の **MCC** チャートを含めることができます。

#### プログラム作成手順


MCC プログラムの作成は、以下の手順に従ってください：

1. MCC ユニットの作成。
2. MCC ユニットでの **MCC** チャートの作成。
3. MCC チャートへの **MCC** コマンドの挿入およびそれらのコマンドのパラメータ設定。

##### 13.3.2.2 MCC ユニットの作成

サンプルプロジェクトのための **MCC** ユニット **motion** の作成は、以下の手順に従ってください。

プロジェクトで **MCC** ユニットを作成するには、以下の手順に従ってください


1. プロジェクトナビゲータの **SIMOTION** デバイスにある **PROGRAMS** フォルダを開いてください。  **Insert MCC unit** をダブルクリックしてください。 **Insert MCC unit** ウィンドウが開きます。
2. 名称 **motion** を入力してください。

3. **Compiler** タブに進んでください。診断目的のために、オプション **Permit program status** および **Permit single step** を有効化してください。こうして、オンラインモードで後にプログラムの実行を監視することができます。
4. **OK** で確定してください。  
MCC ユニットが作成されます。
  - そのユニットは **PROGRAMS** 項目のプロジェクトナビゲータに表示されます。
  - ワークベンチの作業域で、そのユニットの宣言表が開きます。そこで宣言された変数は、MCC ユニット内で適合し、他のユニットにリンクすることができます。サンプルプロジェクトの場合、他の変数宣言は必要とされません。シンボルブラウザでのグローバルデバイスとして、必要な変数を既に作成しました。

### 13.3.2.3 MCC チャートの作成

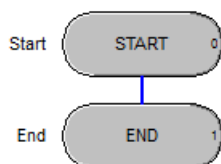
サンプルプロジェクト用の MCC チャート **pos\_axis** を作成するには、以下の手順に従ってください。MCC チャート **pos\_axis** はタイププログラムの POE です。

MCC チャートの作成は、以下の手順に従ってください

1. プロジェクトナビゲータの SIMOTION デバイスにある **PROGRAMS** フォルダを開いてください。
2. **PROGRAMS** フォルダの MCC ユニット **motion** を開いてください。
3.  **Insert MCC chart** をダブルクリックしてください。  
**Insert MCC Chart** ウィンドウが開きます。
4. 名称 **pos\_axis** を入力してください。
5. 作成タイプ **Program** を選択してください。
6. **OK** で確定してください。  
MCC チャートがプロジェクトで作成されます。
  - 作成された MCC チャート **pos\_axis** は、**motion** ユニットの **PROGRAMS** フォルダに表示されます。
  - MCC エディタがワークベンチの作業域で開きます。開始および最終ノードは既に事前に定義されています。MCC プログラミングを開始できます。

### 13.3.2.4 MCC チャートへコマンドブロックの挿入

すべての新たに作成された MCC チャートは既に開始ノートと最終ノードが含まれています。



## 13.3 プログラミング

図 13-6 MCC チャート、開始ブロック、終了ブロック

これらの間に **MCC コマンドブロック** を挿入します。これらのコマンドは、開始ノードから最終ノードへの方向順に処理されます。

MCC コマンドは以下を通じて利用可能：

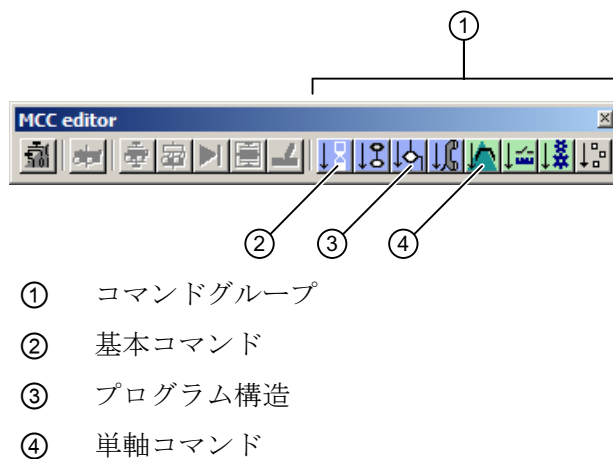
- **MCC エディタ ツールバー**
- **MCC Chart > Paste** メニューコマンド
- コマンドブロックのショートカットメニュー

MCC エディタツールバーでの作業は、以下の手順に従ってください

ツールバーを開く

**MCC エディタ** ツールバーは、**MCC チャート** を開くと直ちに表示されます。

これらのコマンドはコマンドグループに配置されます。サンプルプロジェクトは、コマンドグループ **Basic commands**、**Program structures**、および **Single axis commands** のコマンドを使用します。



- ① コマンドグループ
- ② 基本コマンド
- ③ プログラム構造
- ④ 単軸コマンド

図 13-7 MCC エディタツールバー

**注記**

ツールバーが見えない場合、ディスプレイの電源がオンになっているかどうか確認してください：メニュー **View > Toolbars** を開いてください。ツールバー ウィンドウで **MCC エディタ** のためのチェックボックスを有効化してください。

**コマンドグループを開く**

コマンドグループを表示するには、ツールバーの色つきのボタンの上にマウスを移動させてください。

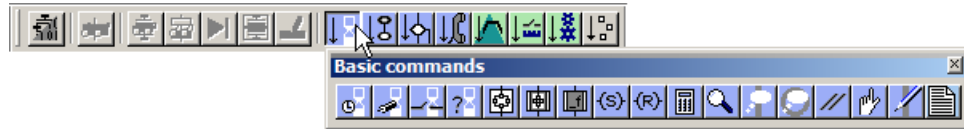


図 13-8 コマンドグループを開く

**コマンドグループを開いたままにする、または、常時閉じておく**

コマンドグループのウィンドウタイトルをクリックし、継続的にコマンドグループを開いていてください。

このコマンドグループを閉じるために、コマンドグループのショートカットメニューの **Hide** を選択してください。

**必要に応じたコマンドグループの設置**

ツールバー、または、ツールバーのコマンドグループをマウスで任意のワークベンチの場所にドラッグしてください。

**ツールバーのドッキング**

ツールバー、または、ツールバーのコマンドグループをマウスで任意のワークベンチの境界域にドラッグし、そこでそれらをドッキングしてください。

**コマンドのツールヒントの表示**

コマンドボタンの上で短時間マウスポインタを維持してください。コマンド名が表示されます。

**MCC エディタツールバーを使った MCC チャートへのコマンド挿入は、以下の手順に従ってください**

1. 有効な MCC チャートの 2 つのコマンド間の接続ラインをクリックする、または、新しいコマンドが挿入されるコマンドの後ろのコマンドをクリックしてください。  
接続ラインまたはコマンドボタンの境界が青く表示されます。マーキング部分が点滅します。
2. **MCC エディタ** ツールバーのコマンドグループを選択してください。コマンドグループの希望のコマンドをクリックしてください。  
コマンドがチャートに挿入されます。

新たに挿入されたコマンドは空です。そのパラメータ設定を行う必要があります；例えば、変数の割り付けの為、変数名および割り付けられた値は指定される必要があります。

以下の項目 **MCC サンプルプログラムの作成：ベーシックフレームワーク (ページ 96)** ではその手順が詳細に説明されます。

## 13.4 MCC サンプルプログラムの作成：ベーシックフレームワーク

### 13.4.1 概要

サンプルプロジェクトで、MCC プログラム **pos\_axis** を作成します。このプログラムでは、軸がターゲット位置に移動され、開始点に戻されます。

#### 電源装置の取り扱い

DRIVE-CLiQ インターフェースなし電源装置を使用している場合、この項目で扱われる軸のプログラム制御運転のためのベーシックフレームで十分です。ドライブには、電源装置の継続的な準備完了信号があります。なぜなら、パラメータが **p0864=1** に設定されているからです。項目 **AUTOHOTSPOT** を参照。

DRIVE-CLiQ インターフェース付き電源装置を使用している場合、プログラムが電源装置をオンにする必要があります。**Getting Started** は、この追加プログラムシーケンスを個別に **MCC サンプルプログラムの拡張：電源装置制御** (ページ 111) 項目で扱います。

### 13.4.2 プログラムのフロー

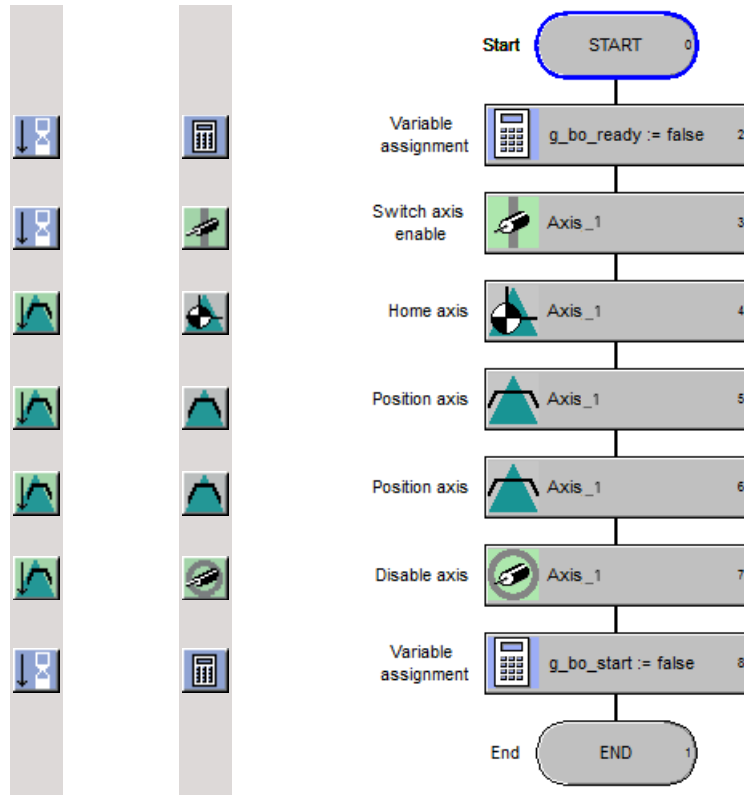
#### 完成した MCC チャート

以下の図は、サンプルプログラムで完成した MCC チャートを示します。

最初の 2 つのコマンドブロックは、変数 **g\_bo\_ready** および **g\_bo\_start** を初期化します。その後、軸はイネーブル、原点復帰、そしてターゲット位置へ移動します。位置到達後、軸は開始点に戻ります。イネーブルが取り消されます。最後に、変数 **g\_bo\_start** がリセットされ、変数 **g\_bo\_ready** が設定されます。



### コマンドグループおよび個別コマンドの MCC チャート



#### コマンドグループおよび個別コマンド

上の図で、コマンドグループおよびそのコマンドを **MCC** に挿入するために後でクリックしなければならない個別のコマンドのアイコンは、すべての **MCC** コマンド前に示されます。

以下の項目でこの手順が詳細に説明されます。

#### 13.4.3 変数の割り付け g\_bo\_ready:=false / g\_bo\_start:=true

変数 **g\_bo\_ready** は、その変数を定義された開始点に設定するために、**FALSE** に設定されます。

変数 **g\_bo\_start** は、軸の開始を示すために **TRUE** に設定されます。

MCC エディタで変数を割り付けるには、以下の手順に従ってください：

1. 開始および終了ノード間のラインをマークしてください。

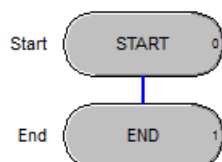


図 13-9 空の MCC チャート

2. MCC エディタ ツールバーの **Basic commands** コマンドグループを開いてください。コマンドグループの **Variable assignment** コマンドをクリックしてください。

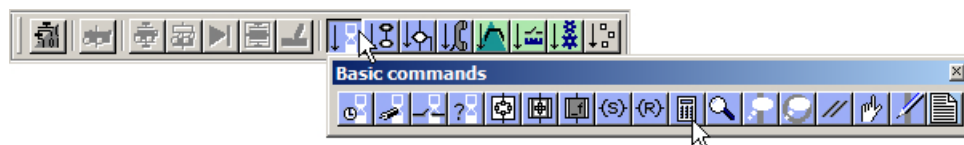


図 13-10 MCC エディタツールバー、変数の割り付けコマンド

**Variable assignment** コマンドブロックは、開始ノードの後の MCC チャートに挿入されます。

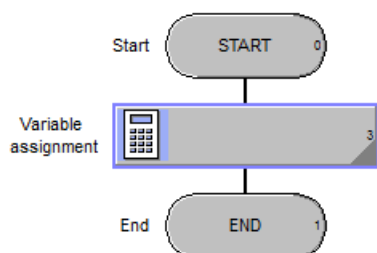


図 13-11 挿入された変数の割り付けコマンドブロックを含む MCC チャート

3. 値 FALSE を変数 **g\_bo\_ready** に割り付けてください。変数名をシンボルブラウザからコマンドブロックに転送します：
  - **Variable assignment** コマンドブロックをダブルクリックしてください。  
**Variable assignment** ウィンドウが開きます。
  - ドロップダウンリストで値 **Formula** を選択してください。  
**Variables** および **Expression** 列を含む表が開きます。
  - シンボルブラウザを開く。これを行うには、プロジェクトナビゲータの **SIMOTION** デバイスにある **GLOBAL DEVICE VARIABLES** をクリックしてください。
  - シンボルブラウザで変数 **g\_bo\_ready** を選択してください。これを行うには、シンボルブラウザの最初の列の先頭の番号をクリックしてください。クリックされた行は、黒い背景で表示されます。

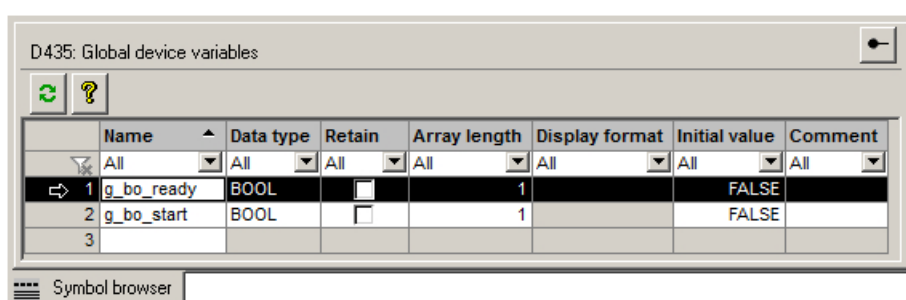


図 13-12 シンボルブラウザ

- マーキングされた行を **Variable assignment** ウィンドウの **Variable** 列にドラッグ & ドロップしてください。変数が挿入され、その名称が表示されます。
- **Expression** 列に、値 **false** を入力し、RETURN で確定してください。

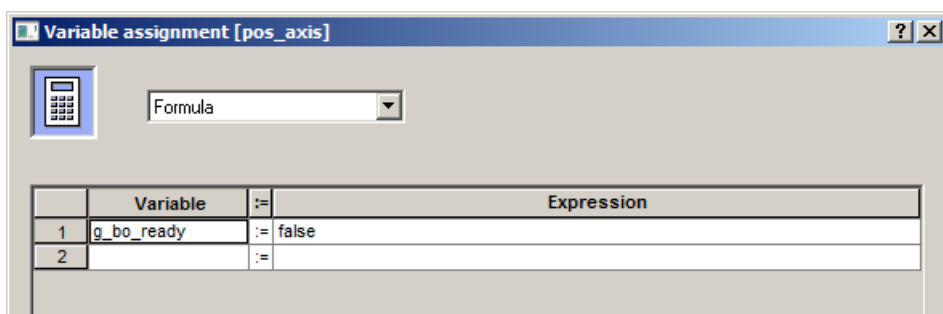


図 13-13 変数の割り付け

4. 値 **true** を変数 **g\_bo\_ready** に同じように割り付けてください。

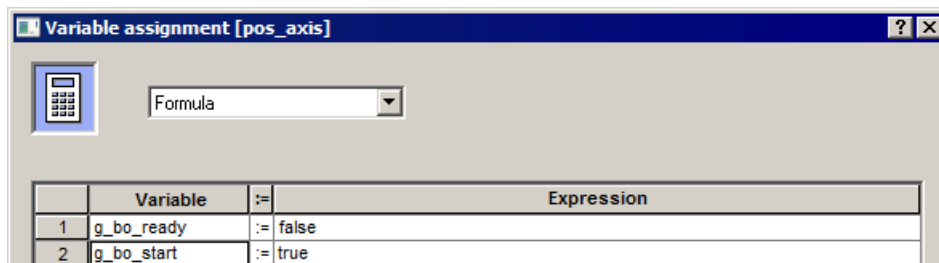


図 13-14 割り付けられた変数

5. **OK** をクリックし、ウィンドウを閉じてください。  
グローバルデバイス変数を値 **false** および **true** に設定しました。

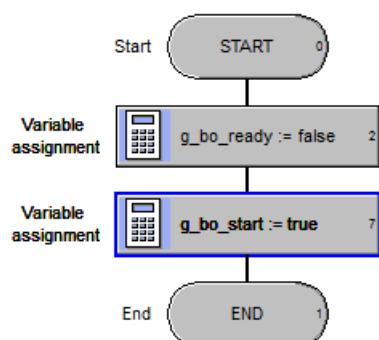


図 13-15 MCC チャート、実行された変数の割り付け

#### 13.4.4 変数の割り付け **g\_bo\_start:=true**

変数 **g\_bo\_start** は、軸の開始を示すために **TRUE** に設定されます。

値 TRUE を変数 **g\_bo\_start** に割り付けるには、以下の手順に従ってください

- 2 番目の **Variable assignment** ブロックを最終ノードの前の MCC チャートに挿入してください：
  - 新しいブロックが続くコマンドブロックをクリックしてください。ノードは青く表示されます。
  - MCC エディタ ツールバーで、**Basic commands > Variable assignment** コマンドを選択してください。  
**Variable assignment** コマンドブロックは、MCC チャートに挿入されます。
- 値 TRUE を変数 **g\_bo\_start** に割り付けてください。その変数を選択するために、オートコンプリート機能を使用してください：
  - MCC チャートの空の **Variable assignment** コマンドブロックをダブルクリックしてください。  
**Variable assignment** ウィンドウが開きます。
  - ドロップダウンリストで値 **Formula** を選択してください。
  - Variable** 列で、変数名 **g\_bo\_start** の最初の文字を入力してください。オートコンプリート機能のためにキー **Ctrl+ スペースキー** を押してください。リストから変数を選択してください。

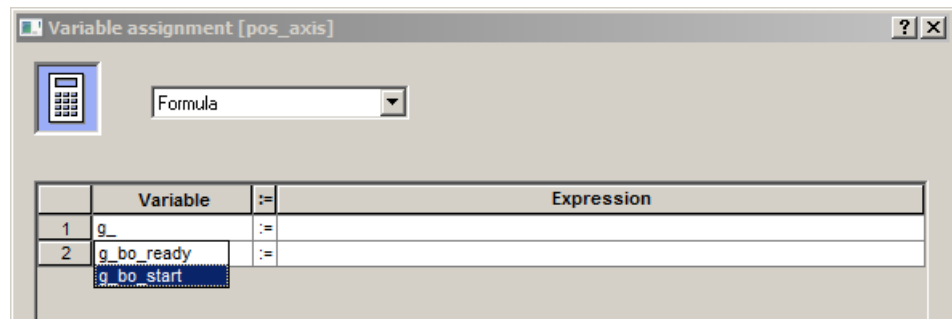


図 13-16 MCC チャート、オートコンプリート機能を使った変数の選択

- Expression** 列で、値 **true** を入力し、RETURN キーで確定してください。
- OK をクリックし、ウィンドウを閉じてください。  
 コマンドがパラメータ設定されました。

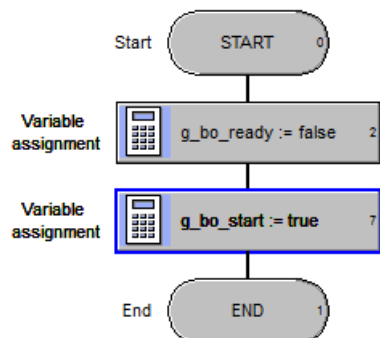


図 13-17 MCC チャート、実行された変数の割り付け

### 13.4.5 軸イネーブルへの切り替え

軸イネーブルへの切り替えは、以下の手順に従ってください

1. 最終ノードの前の MCC チャートにステートメントブロック **Switch axis enable** を挿入してください：
  - 最終ノード前の最後のブロックをクリックしてください。ノードは青く表示されます。
  - **MCC エディタ ツールバー**の **Single axis commands** コマンドグループを開いてください。コマンドグループの **Switch axis enable** コマンドをクリックしてください。



図 13-18 MCC エディタツールボックス、Switch axis enable コマンド

**Axis enable** コマンドブロックが、選択されたブロックの後に MCC チャートに挿入され、青く表示されます。

2. 軸イネーブルをパラメータ設定：
  - MCC チャートの挿入された **Switch axis enable** コマンドブロックをダブルクリックしてください。 **Switch axis enable** ウィンドウが開きます。
  - **Axis** 域で、プログラムにより制御される軸を選択してください。

- **Delay program execution** オプションがサンプルプロジェクトのために選択されていることを確認してください。このコマンドは、軸がイネーブルされる場合にのみ終了します。  
他のパラメータはデフォルト値のままにしておいてください。

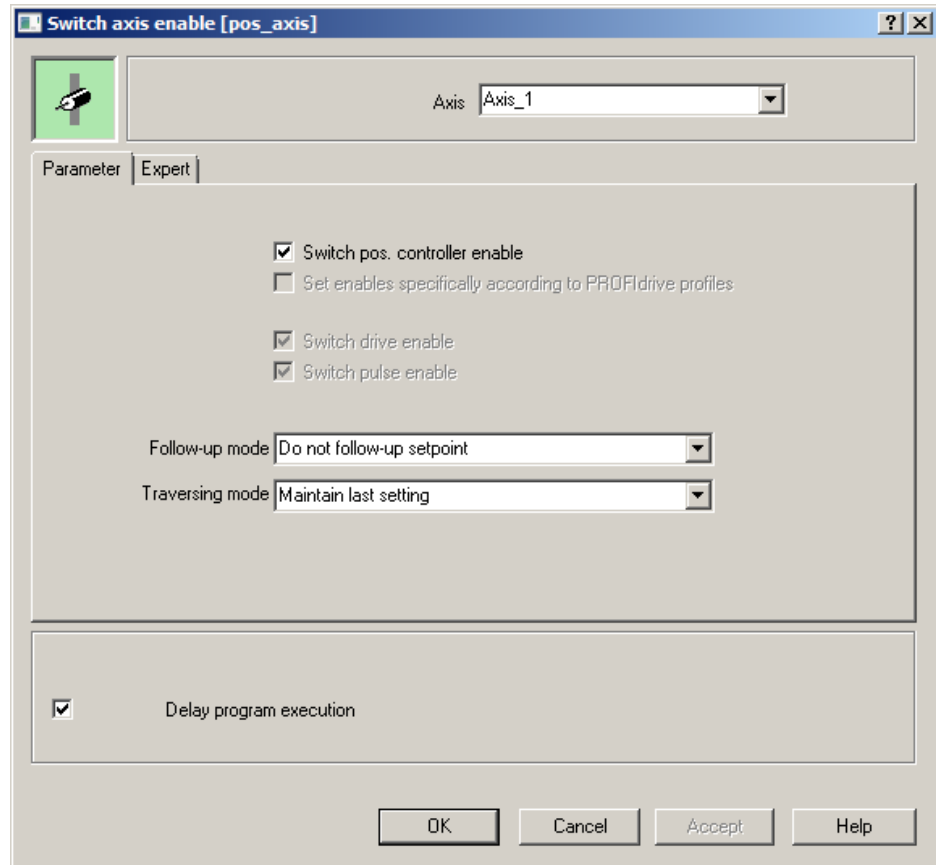


図 13-19 MCC チャート、軸イネーブルへの切り替え

3. **OK** をクリックし、ウィンドウを閉じてください。  
コマンドがパラメータ設定されました。

### 13.4.6 軸の原点復帰

軸の位置値は常に軸の座標系に関連しています。

原点復帰の手順は、座標系の原点位置を定義します。

軸の原点復帰の実装は、以下の手順に従ってください

1. 最終ノードの前の MCC チャートにステートメントブロック **Home axis** を挿入してください：
  - 最終ノード前の最後のブロックをクリックしてください。ノードは青く表示されます。
  - **MCC エディタ ツールバー**の **Single axis commands** コマンドグループを開いてください。コマンドグループのコマンド **Home axis** をクリックしてください。



図 13-20 MCC エディタツールバー、軸の原点復帰コマンド

**Home axis** コマンドブロックが、選択されたブロックの後に MCC チャートに挿入され、青く表示されます。

2. 軸の原点復帰をパラメータ設定：
  - MCC チャートの挿入された **Home axis** コマンドブロックをダブルクリックしてください。 **Home axis** ウィドウが開きます。
  - **Axis** 域で、プログラムにより制御される軸を選択してください。
  - **Parameters** タブで、原点復帰タイプ **Set home position** を選択してください。  
**Home position coordinates** 域に値 **0** を入力してください。  
原点復帰タイプで、基準点座標が軸の現在の位置に割り付けられます (実績値)。動作中の移動モーションはありません。



- 他のパラメータはデフォルト値のままにしておいてください。

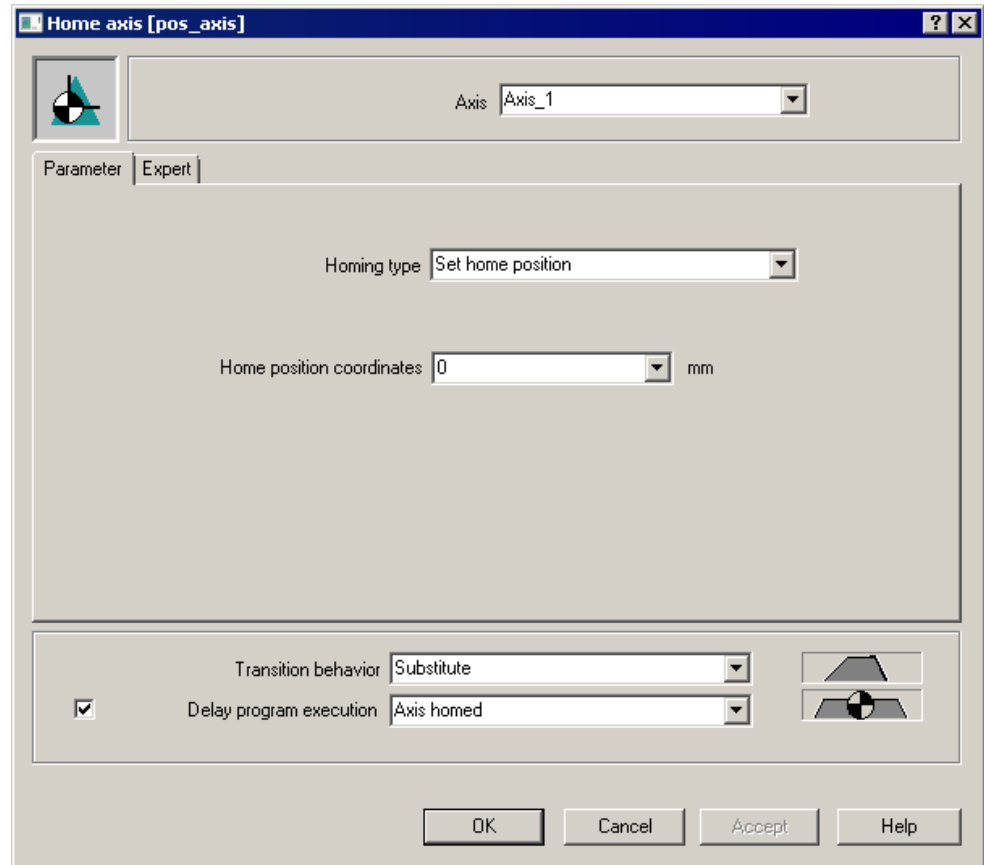


図 13-21 MCC チャート、軸の原点復帰

3. **OK** をクリックし、ウィンドウを閉じてください。  
コマンドがパラメータ設定されました。

### 13.4.7 軸のターゲット位置への位置決め

このプログラムは、速度 500 mm/s で位置 2000 mm に軸を移動させるものです。

軸をターゲット位置に移動させるには、以下の手順に従ってください

1. 最終ノードの前の MCC チャートにステートメントブロック **Position axis** を挿入してください：
  - 最終ノード前の最後のブロックをクリックしてください。ノードは青く表示されます。
  - **MCC エディタ ツールバー**の **Single axis commands** コマンドグループを開いてください。コマンドグループの **Position axis** コマンドをクリックしてください。

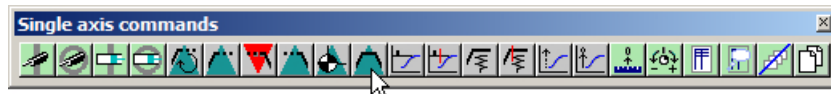


図 13-22 MCC エディタツールバー、軸の位置決めコマンド

**Position axis** コマンドブロックが、選択されたブロックの後に MCC チャートに挿入され、青く表示されます。

2. 軸動作をパラメータ設定：
  - MCC チャートの挿入された **Position axis** コマンドブロックをダブルクリックしてください。 **Position axis** ウィンドウが開きます。
  - **Axis** 域で、プログラムにより制御される軸を選択してください。
  - **Parameters** タブで位置 2000 を入力してください。速度を 500 mm/s に設定してください。

- 他のパラメータはデフォルト値のままにしておいてください。

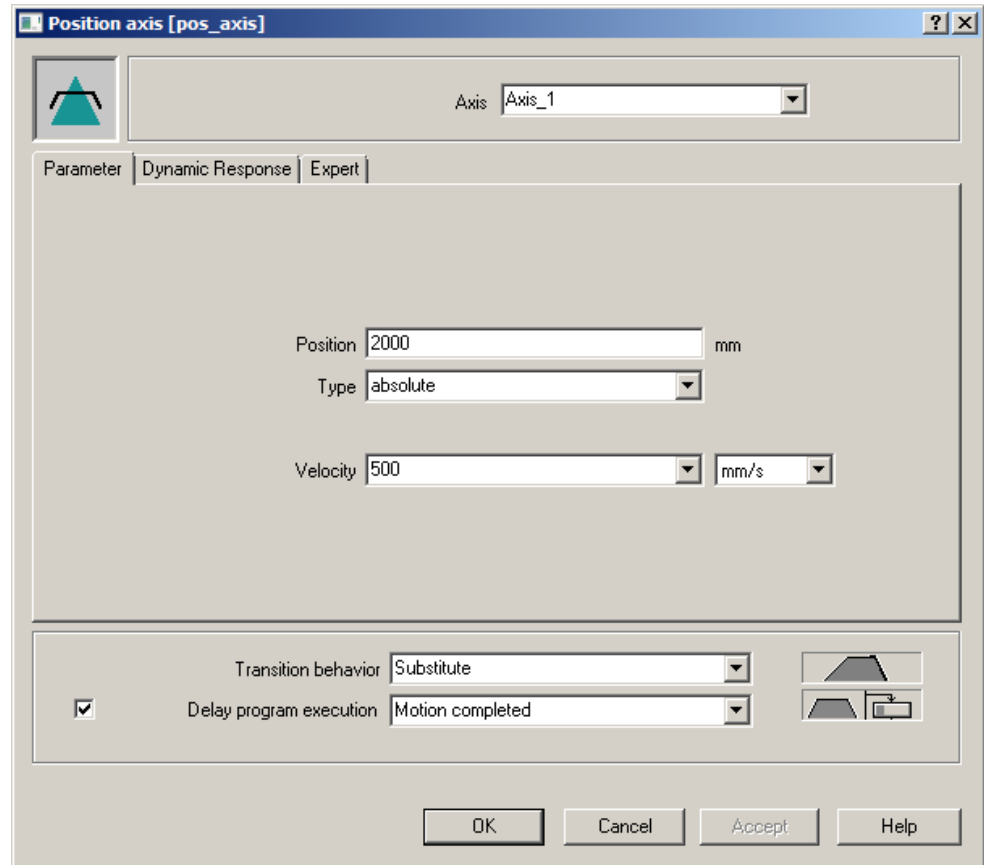


図 13-23 MCC チャート、軸のターゲット位置への位置決め

3. **OK** をクリックし、ウィンドウを閉じてください。  
コマンドがパラメータ設定されました。

### 13.4.8 軸の開始位置への位置決め

このプログラムは、速度 500 mm/s で開始位置 0 mm に軸を戻すものです。

軸を始動位置に戻すには、以下の手順に従ってください

軸のターゲット位置への位置決め (ページ 105) の手順と同じように行います。

1. 最終ノードの前の MCC チャートにステートメントブロック **Position axis** を挿入してください。
2. 軸動作をパラメータ設定：
  - **Parameters** タブで、位置 0 を入力してください。速度を 500 mm/s に設定してください。
  - 他のパラメータはデフォルト値のままにしておいてください。

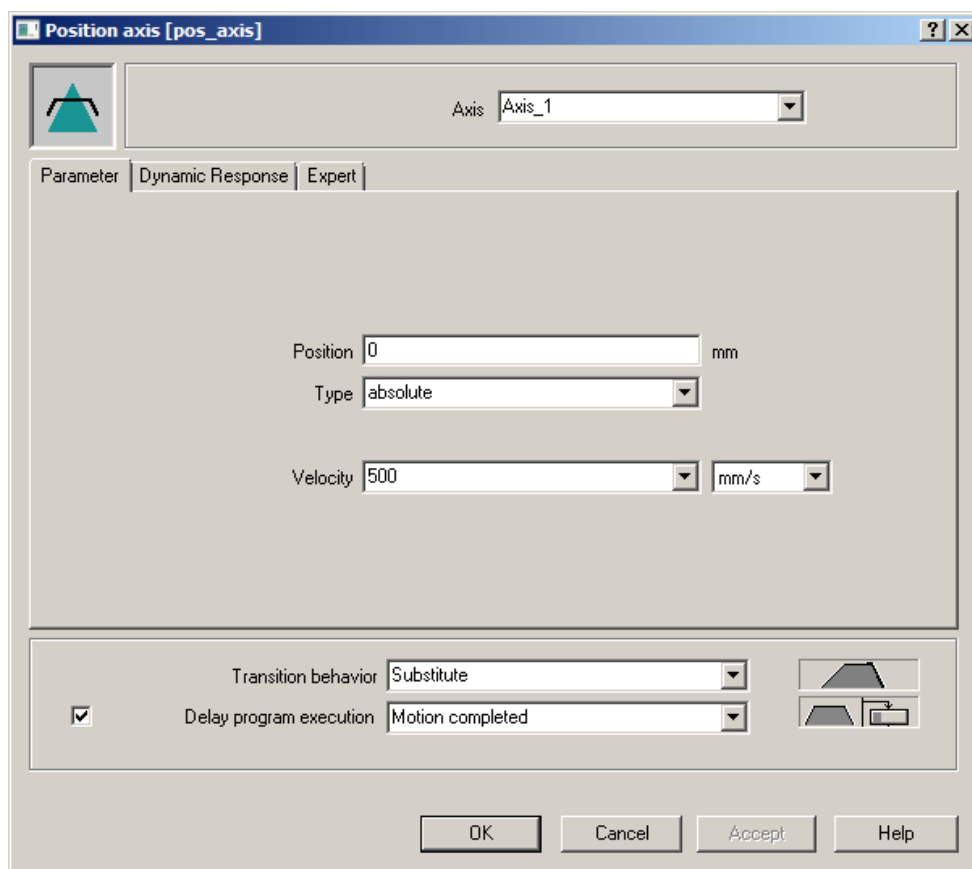


図 13-24 MCC チャート、軸の開始位置への位置決め

3. **OK** をクリックし、ウィンドウを閉じてください。  
コマンドがパラメータ設定されました。

### 13.4.9 軸の無効化

軸の無効化は、以下の手順に従ってください

1. 最終ノードの前の MCC チャートにステートメントブロック **Disable axis** を挿入してください：
  - 最終ノード前の最後のブロックをクリックしてください。ノードは青く表示されます。
  - **MCC エディタ ツールバー**の **Single axis commands** コマンドグループを開いてください。コマンドグループの **Disable axis** コマンドをクリックしてください。

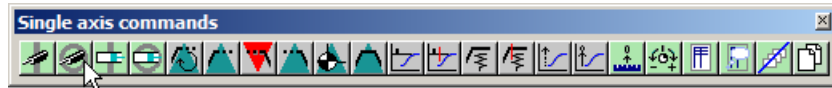


図 13-25 MCC エディタツールバー、軸の無効化コマンド

**Disable axis** コマンドブロックが、選択されたブロックの後に MCC チャートに挿入され、青く表示されます。

2. このコマンドをパラメータ設定：
  - MCC チャートの挿入された **Disable axis** コマンドブロックをダブルクリックしてください。 **Disable axis** ウィンドウが開きます。

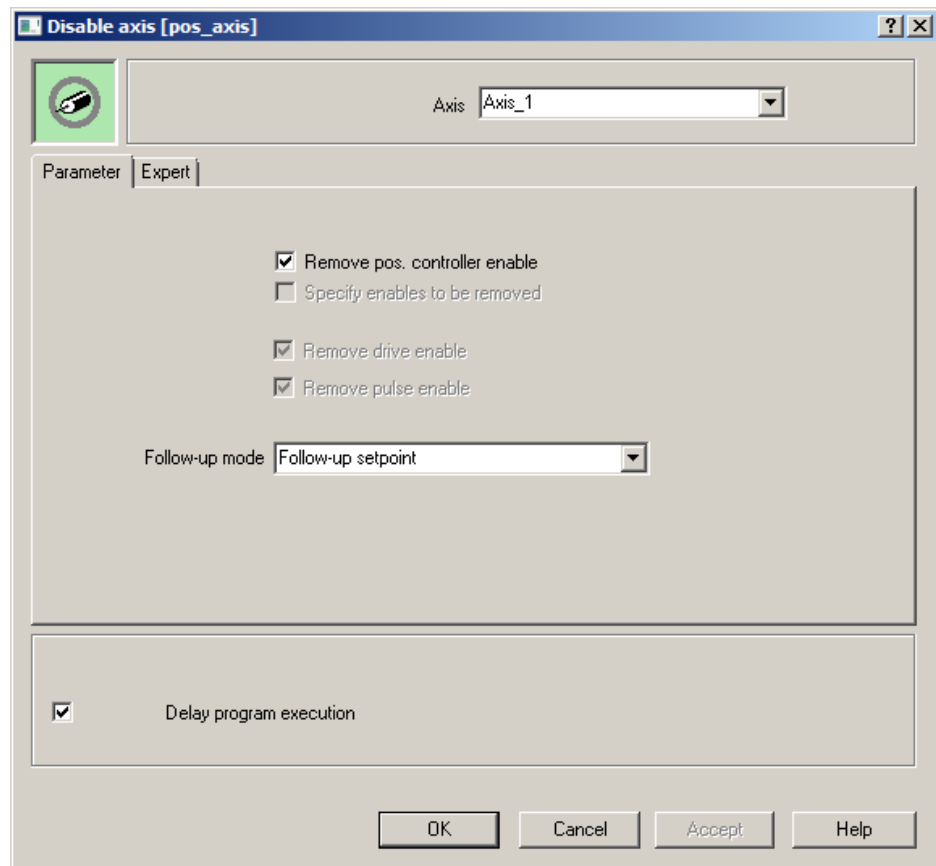


図 13-26 MCC チャート、軸の無効化

### 13.4 MCC サンプルプログラムの作成：ベーシックフレームワーク

- **Axis** 域で、プログラムにより制御される軸を選択してください。
  - 他のパラメータはデフォルト値のままにしておいてください。
3. **OK** をクリックし、ウィンドウを閉じてください。  
コマンドがパラメータ設定されました。

#### 13.4.10 変数の割り付け **g\_bo\_start:=false / g\_bo\_ready:=true**

軸の動作が完了しました。変数 **g\_bo\_start** が FALSE に設定されます。変数 **g\_bo\_ready** は TRUE に設定されます。

値 FALSE を変数 **g\_bo\_start** に割り付けるには、以下の手順に従ってください

変数の割り付け **g\_bo\_start:=true** (ページ 100) の手順と同じように行います。

1. 最終ノードの前の MCC チャートにステートメントブロック **Variable assignment** を挿入してください：
2. 値 **false** を変数 **g\_bo\_start** に、そして、値 **true** を変数 **g\_bo\_ready** に割り付けてください。
3. **OK** をクリックし、ウィンドウを閉じてください。  
コマンドがパラメータ設定されました。

#### 13.4.11 変数の割り付け **g\_bo\_ready:=true**

軸の動作が完了しました。変数 **g\_bo\_ready** は TRUE に設定されます。

値 TRUE を変数 **g\_bo\_ready** に割り付けるには、以下の手順に従ってください

変数の割り付け **g\_bo\_ready:=false / g\_bo\_start:=true** (ページ 97) の手順と同じように行います。

1. 最終ノードの前の MCC チャートにステートメントブロック **Variable assignment** を挿入してください：
2. 値 **true** を変数 **g\_bo\_ready** に割り付けてください。
3. **OK** をクリックし、ウィンドウを閉じてください。  
コマンドがパラメータ設定されました。

## 13.5 MCC サンプルプログラムの拡張：電源装置制御

### 13.5.1 プログラムのフロー

DRIVE-CLiQ インターフェース付き電源装置を使用している場合、プログラムされたコントローラは、軸のコマンドが発行できる前に、電源をオンにする必要があります。以下に示されるプログラムシーケンスでこのタスクを処理します。この時までには作成された **pos\_axis** プログラムはそれに応じて拡張される必要があります。

#### 完成した MCC チャート

軸制御 **Switch axis enable** のための最初のコマンドの前に、2 x 2 ブロック **\_linemodule\_control** システムファンクション呼び出しおよび **LineModule\_STW** 変数の割り付けが MCC チャート **pos\_axis** にあります。

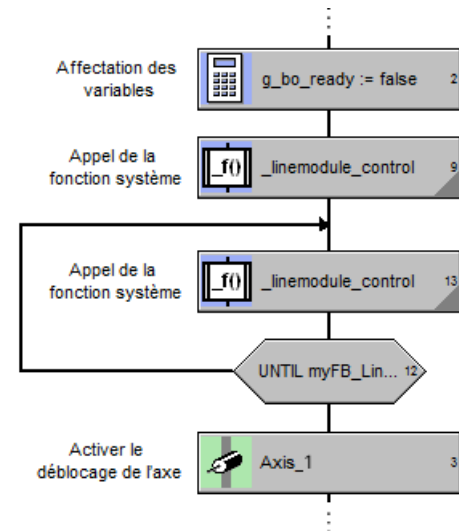
コマンドグループ



シングルコマンド



サンプルプログラムの MCC  
チャート、電源装置の制御



**UNTIL** ループ前の最初の 2 つのブロックは、電源装置の運転パラメータを開始値にリセットします。初期化なしには電源装置をオンにできない場合があります。更に、これは電源装置のイネーブルを妨げるアラームメッセージも確認します。

### 13.5 MCC サンプルプログラムの拡張：電源装置制御

**UNTIL** ループのブロックは、電源装置をオンにします。ループは、電源装置が運転準備完了になるまでスイッチオンコマンドを繰り返します。

---

#### 注記

無限ループで **RoundRobin** 実行レベルの性能に関する過度な要求 (バックグラウンドタスク、モーションタスク) を回避するために、イベントを能動的に待機するために、ループで `_waitTime(T#0ms)` の使用が推奨されます。

---



## 13.5.2 システムファンクションの呼び出し **\_LineModule\_control[FB]**

システムファンクションの呼び出し **\_LineModule\_control[FB]** の作成は、以下の手順に従ってください

1. ステートメントブロック **System function call** を MCC チャートに挿入してください：
  - コマンドブロック **g\_bo\_start := true** をクリックしてください。  
ノードは青く表示されます。
  - **MCC エディタ ツールバー**で、**Basic commands > System function call** コマンドを選択してください。



図 13-27 MCC エディタツールバー、システムファンクションの呼び出しコマンドブロックは、MCC チャートに挿入されます。

2. 機能の呼び出しをパラメータ設定：
  - MCC チャートの空の **System function call** コマンドブロックをダブルクリックしてください。  
**System function call [pos\_axis]** ダイアログが開きます。
  - コマンドライブラリを開いてくださいプロジェクトナビゲータの画面域に、個別タブとして、コマンドライブラリがあります。

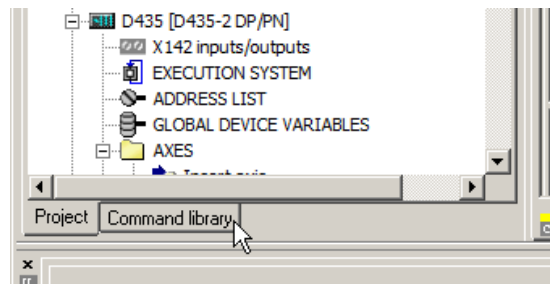
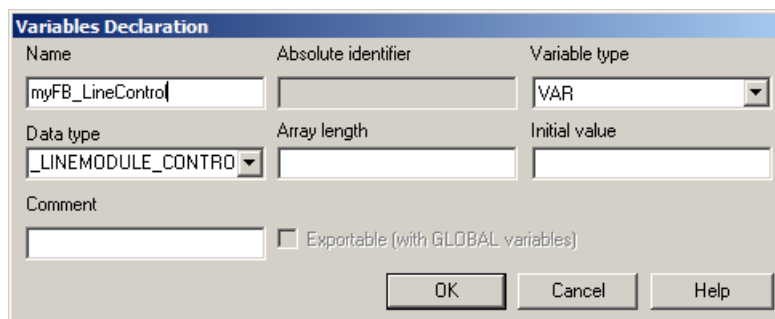


図 13-28 コマンドライブラリを開く

- コマンドライブラリの項目 **Drives > SINAMICS** に進んでください。
- コマンドライブラリから **\_LineModule\_control[FB]** ファンクションの呼び出しを **System function call[pos\_axis]** ウィンドウの **System function** 域にドラッグ & ドロップしてください

3. システムファンクションの呼び出しのためのインスタンスを定義してください：
- **System function call [pos\_axis]** ウィンドウの **Instance** 域に、インスタンス名 **myFB\_LineControl** を入力してください。  
例えば、他の要素をクリックしたり、TAB や RETURN キーを使って、この域を離れると直ちに、**Variables declaration** ダイアログが開きます。インスタンス変数 **myFB\_LineControl** は、変数タイプ **VAR** にデフォルトとして割り付けられます。変数はこのようにプログラム **pos\_axis** 内で有効です。



The image shows a 'Variables Declaration' dialog box. It has a title bar with the text 'Variables Declaration'. Inside, there are several input fields and a checkbox. The 'Name' field contains 'myFB\_LineControl'. The 'Absolute identifier' field is empty. The 'Variable type' dropdown menu is set to 'VAR'. The 'Data type' dropdown menu is set to '\_LINEMODULE\_CONTRO'. The 'Array length' field is empty. The 'Initial value' field is empty. There is a 'Comment' field which is empty. At the bottom right, there is a checkbox labeled 'Exportable (with GLOBAL variables)' which is unchecked. At the bottom, there are three buttons: 'OK', 'Cancel', and 'Help'.

図 13-29 インスタンス変数 myFB\_LineControl の変数宣言

デフォルトを受け付けてください。このダイアログを確定するために、**OK** をクリックしてください。

- インスタンスの入力変数 VAR\_INPUT を以下の値に設定してください。便利なオートコンプリートオペレータ機能を利用してください **Ctrl+space bar** :

変数名称	値	意味
reset	true	電源装置をリセット
periln	LineModule_ZSW	電源装置のステータスワード
typeLM	ACTIVE_LINE_MODULE SMART_LINE_MODULE BASIC_LINE_MODULE	使用される電源装置に応じて制御されるモジュールタイプ
periOut	LineModule_STW	コントロールワード、電源装置のシーケンスコントロール

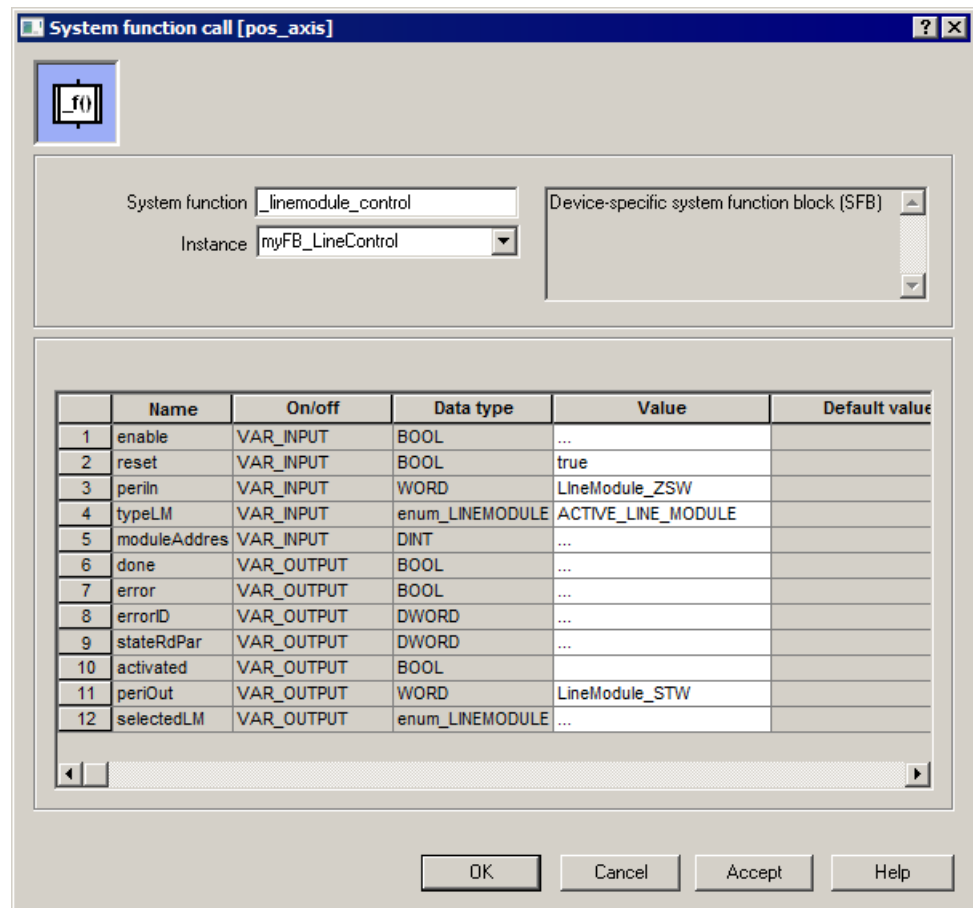


図 13-30 MCC チャート、システムファンクションの呼び出し \_linemodule\_control

4. **OK** で確定してください。コマンドがパラメータ設定されました。

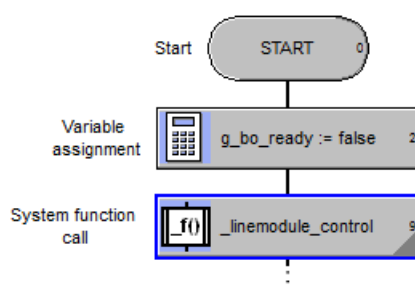


図 13-31 MCC チャート、実行された変数の割り付け \_linemodule\_control

### 13.5.3 変数の割り付け LineModule\_STW:=myFB\_LineControl.periOut

myFB\_LineControl.periOut 変数を LineModule\_STW 変数に割り付けるには、以下の手順に従ってください

1. **System function call** ブロックの後で、**Variable assignment** ブロックを作成してください。
2. myFB\_LineControl.periOut 変数を LineModule\_STW I/O 変数に割り付けてください。オートコンプリートを使用する、または **Variable** および **Expression** 域の両方でドラッグ & ドロップできます。

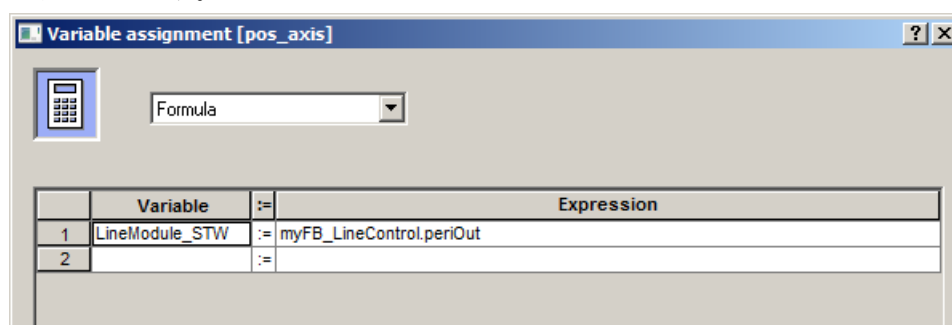


図 13-32 MCC チャート、変数の割り付け  
LineModule\_STW:=myFB\_LineControl.periOut

3. **OK** をクリックし、ウィンドウを閉じてください。  
コマンドがパラメータ設定されました。

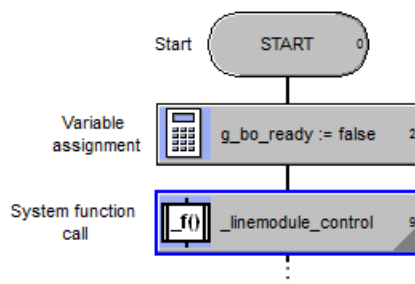


図 13-33 MCC チャート、システムファンクションの呼び出しおよび変数の割り付け

## 13.5.4 UNTIL ループを作成

UNTIL ループの作成は、以下の手順に従ってください

1. ブロック **System function call** および後続の軸コマンド **Switch axis enable** の間の接続ラインをクリックしてください。  
基準点は青く表示されます。
2. **MCC エディタ ツールバー**で、コマンドを選択 **Program structures > UNTIL:Loop with condition at end**

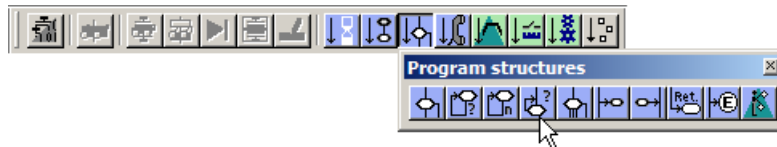


図 13-34 MCC エディタツールバー、UNTIL ループ  
ループを含む **UNTIL** ブロックが挿入されます。

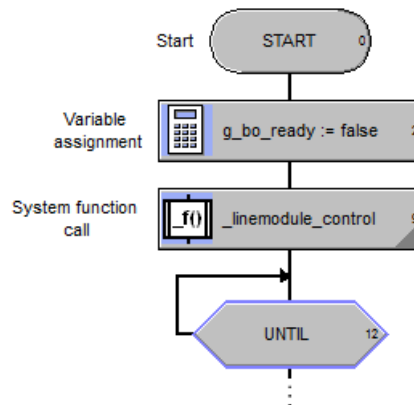


図 13-35 MCC エディタ、UNTIL ループ

3. ループ条件をパラメータ設定：  
ループは、電源装置の運転準備が完了するまで続きます。
  - UNTIL ブロックをダブルクリックしてください。ウィンドウ **UNTIL:Loop with condition at end[[pos\_axis]** が開きます。
  - 左上の入力域で値 **Formula** を選択してください。
  - **Until** 域にループ条件を入力：**myFB\_LineControl.activated=TRUE**。  
オートコンプリート機能 **Ctrl+space bar** をこの入力域でも使用することができます。
4. **OK** で確定してください。  
ループがプログラムで作成され、パラメータ設定されます。

### 13.5.5 ブロックのコピー

MCC チャートのステートメントブロックをコピーするには、以下の手順に従ってください

ステートメントブロック **System function call \_LineModule\_control[FB]** および **Variable assignment LineModule\_STW:=myFB\_LineControl.periOut** を **UNTIL** ループにコピーしてください。この目的のために、オペレータ機能 **Copy** および **Paste** を使用してください。 .

1. ブロックをマーク：
  - **Shift** キーを押し続けてください。
  - 次にブロックを順番にクリックしてください。両方のブロックの角が青く表示されます。
2. 選択されたブロックをクリップボードにコピー：
  - 右クリックでショートカットメニューを開いてください。
  - ショートカットメニューの **Copy** コマンドを選択してください。
  - または、**Ctrl+C** キー機能を使ってください。
3. MCC チャートでコピー操作のターゲットを選択：
  - **UNTIL** ループ内の接続ラインをクリックしてください。接続部は青く表示されます。

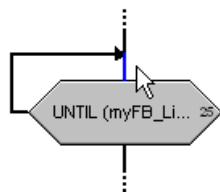


図 13-36 MCC チャート、UNTIL ループ内部域をマーク

4. ターゲット位置にコピーされたブロックを挿入してください：
  - 右クリックして、ショートカットメニューを再び開いてください。
  - **Paste** コマンドをショートカットメニューで選択してください。
  - または、**Ctrl+V** キー機能を使ってください。
  - ブロックはループ内に挿入されます。

### 13.5.6 システムファンクションの呼び出し `_linemodule_control` の適用

システムファンクションの呼び出し `_linemodule_control` の変更は、以下の手順に従ってください

**UNTIL** ループ内のシステムファンクションの呼び出し `_linemodule_control` で電源装置がオンになります。

1. ダブルクリックで、システムファンクションブロックを開いてください。
2. **myFB\_LineControl** インスタンスのパラメータ変数を以下のように調整してください：

変数名称	値	意味
<b>enable</b>	true	電源装置をオンにします
<b>reset</b>		空欄。変数はこのように <b>false</b> に設定されます
<b>periln</b>	LineModule_ZSW	電源装置のステータスワード
<b>typeLM</b>	ACTIVE_LINE_MODULE SMART_LINE_MODULE BASIC_LINE_MODULE	使用される電源装置に応じた制御されるモジュールタイプ

3. **OK** で確定してください。

## 13.6 サンプルプロジェクトのための追加 MCC プログラムの作成

### システムイベントの処理

後の項目 実行システムのコンフィグレーション (ページ 131) でセットアップする予定の実行システムは更に 2 つの MCC プログラムを必要とします：

- **technology\_fault**
- **peripheral\_fault**

これらのプログラムはシステムイベントの処理に使用されます。一般的に処理が必要です。ユーザプログラムで処理されないシステムイベントが発生する場合、CPU は STOP モードに移行します。

サンプルプロジェクトの範囲内で、システムイベントの特別な処理は必要ありません。プログラム **technology\_fault** および **peripheral\_fault** はそのため空のままです。

MCC プログラム **technology\_fault** および **peripheral\_fault** を作成するには、以下の手順に従ってください

1. MCC ユニット **fault** を作成してください。  
プロジェクトナビゲータの SIMOTION デバイスにある **PROGRAMS** フォルダを開いてください。📁 **Insert MCC unit** をダブルクリックしてください。名称 **fault** を MCC ユニットの割り付けてください。
2. MCC チャート **technology\_fault** を作成してください。  
**PROGRAMS** フォルダの MCC ユニット **fault** を開いてください。📁 **Insert MCC chart** をダブルクリックしてください。名称 **technology\_fault** を MCC チャートに割り付けてください。作成タイプ **Program** を選択してください。  
プログラムは空のままです。

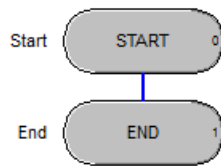


図 13-37 空のプログラム **technology\_fault**

3. MCC ユニット **fault** 内で MCC チャート **peripheral\_fault** を作成してください。  
このプログラムも空の状態です。

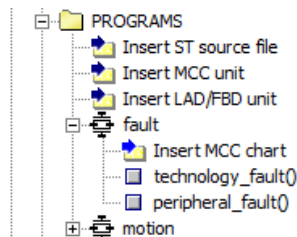
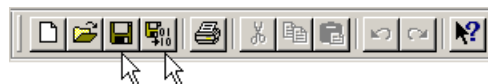


図 13-38 MCC ユニット **fault** 内のユニット MCC プログラム **technology\_fault** および **peripheral\_fault**

## 13.7 MCC サンプルプログラムのバックアップ

作成された MCC プログラムをバックアップしてください。

これを行うには、ツールバーの **Save project** または **Save project and compile changes** ボタンをクリックしてください。





## 関連情報

コマンド **Save project and compile changes** の他の選択肢として、MCC エディタ ツールバーに **Accept and compile** があります。



このコマンドは、現在選択されたプログラムおよび同じユニットの他のプログラムのすべてをコンパイルします。

しかしながら、このコマンドは変更を保存しません。

このように、プロジェクト全体を再び保存またはコンパイルすることなく、プログラムへの変更を受け付けるオプションがあります。

## 13.8 LAD/FBD ラダーロジック / ファンクションブロックダイアグラム

### 13.8.1 LAD および FBD プログラミング言語

#### LAD (ラダーロジック)

LAD は、ラダーロジックの略語です。LAD は、グラフィカルなプログラミング言語です。これらの指示のシンタックスは、回路図に類似しています。LAD を使用すると、入力、出力、および演算を通してコンダクタバー間の信号フローを簡単にトレースできます。LAD ステートメントは、ネットワークを構築するために、グラフィック表示で接続された要素やブロックで構成されます。LAD の演算は、ブール論理の規則に従って実行されます。

#### FBD (ファンクションブロックダイアグラム)

FBD は、ファンクションブロックダイアグラムの略語です。FBD は、グラフィック表示のプログラミング言語です。論理的な関係を示すために、ブーリアン演算からの類似したロジックボックスを使用します。更に、複雑な機能 (例：数学関数) を、論理ボックスに直接連動させて表示することができます。




## SIMOTION SCOUT のシステム

SIMOTION デバイスの **PROGRAMS** フォルダには、プロジェクトで作成された LAD/FBD ユニットが含まれます。

LAD/FBD ユニットには、SIMOTION デバイスで動作する LAD/FBD が含まれます。LAD/FBD ユニットには複数の LAD/FBD プログラムが含まれます。

"LAD/FBD program" は、異なるプログラム組織のユニット (POE) の総称です。

POE は、プログラム、機能またはファンクションブロックです。POE のタイプは、プロジェクトナビゲータのアイコンで表示されます：

-  プログラム
-  機能
-  ファンクションブロック

## プログラム作成手順

LAD/FBD プログラムには以下の手順が含まれます：

1. LAD/FBD ユニットの作成。
2. LAD/FBD ユニットでの LAD/FBD プログラムの作成。
3. LAD/FBD プログラムでの LAD/FBD コマンドの挿入およびコマンドのパラメータ設定。
4. LAD/FBD プログラムの保存およびコンパイル。

## プログラミング言語の切り替え

SIMOTION SCOUT TIA で、ラダーロジックとファンクションブロックダイアグラム間での容易な切り替えが可能になります。LAD/FBD エディタには、コマンド **LAD/FBD program > Switch to FBD** または **Switch to LAD** が含まれます。

### 13.8.2 LAD/FBD ユニットの作成

サンプルプロジェクト用の LAD/FBD ユニット **bg\_out** を作成するには、以下の手順に従ってください。

プロジェクトで LAD/FBD ユニットを作成するには、以下の手順に従ってください

1. プロジェクトナビゲータの SIMOTION デバイスにある **PROGRAMS** フォルダを開いてください。📁 **Insert LAD/FBD unit** をダブルクリックしてください。Insert LAD/FBD unit ウィンドウが開きます。
2. ユニット名 **bg\_out** を入力してください。
3. **Compiler** タブに進んでください。診断目的の場合、オプション **Permit program status** を有効化してください。こうして、オンラインモードで後にプログラムの実行を監視することができます。
4. **OK** で確定してください。  
LAD/FBD ユニットが作成されます。
  - LAD/FBD ユニット **bg\_out** が **PROGRAMS** フォルダ内に表示されます。

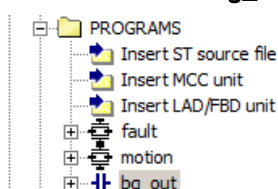


図 13-39 PROGRAMS フォルダ内の MCC および LAD/FBD ユニット

- ワークベンチの作業域で、そのユニットの宣言表が開きます。そこで宣言された変数は、LAD/FBD ユニット内で適用され、他のユニットでリンクできます。サンプルプロジェクトの場合、他の変数宣言は必要とされません。シンボルブラウザでのグローバルデバイスとして、必要な変数を既に作成しました。

### 13.8.3 LAD/FBD プログラムの作成

LAD/FBD ユニット **bg\_out** 内で LAD/FBD プログラム **LAD\_1** を作成するには、以下の手順に従ってください。

LAD/FBD で LAD/FBD プログラムを作成するには、以下の手順に従ってください

1. プロジェクトナビゲータの SIMOTION デバイスにある **PROGRAMS** フォルダを開いてください。
2. **PROGRAMS** フォルダで LAD/FBD ユニット **bg\_out** を開いてください。📁 **Insert LAD/FBD program** をダブルクリックしてください。Insert LAD/FBD program ウィンドウが開きます。
3. 名称 **LAD\_1** を入力してください。名称はプロジェクト全体で一意になっている必要があります。

4. 作成タイプ **Program** を選択してください。

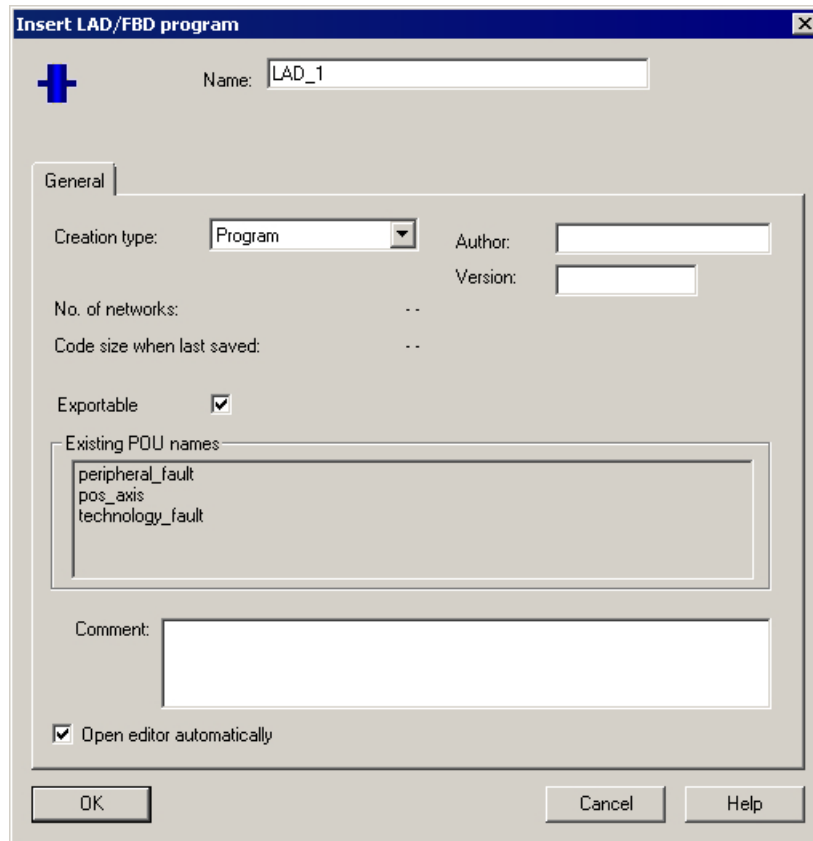


図 13-40 LAD/FBD プログラムの挿入

5. **OK** で確定してください。  
LAD/FBD プログラム **LAD\_1** がプロジェクトで作成されます。
- LAD/FBD プログラムは **PROGRAMS** フォルダに表示されます。

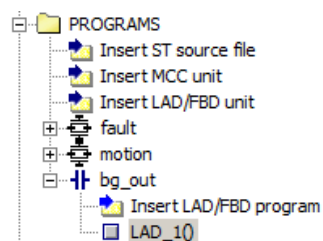


図 13-41 PROGRAMS フォルダの LAD プログラム LAD\_1

- LAD/FBD エディタはワーキングベンチの作業域で開きます。プログラミングを開始することができます

### 13.8.4 LAD サンプルプログラムの作成

サンプルプロジェクトのために LAD プログラム **LAD\_1** を書きます。

## 13.8 LAD/FBD ラダーロジック / ファンクションブロックダイアグラム

このプログラムでは、グローバルデバイス変数 **g\_bo\_start** および **g\_bo\_ready** は周期的に読み出され、I/O 変数 **q\_bo\_output0** および **q\_bo\_output1** がそれに従って設定されます。

プログラム状態のデジタル出力 DO 0 および DO 1 への転送は、MCC チャート **pos\_axis** で直接プログラムすることもできます。しかしながら、**Getting Started** では、LAD プログラムを紹介するために、自律 LAD プログラムでの割り付けを実装します。

サンプルプロジェクトのための LAD プログラム **LAD\_1** の作成は、以下の手順に従ってください。

1. それがワークベンチの作業域で見えない場合は、LAD/FBD プログラムを開いてください。この目的のために、プロジェクトナビゲータの **PROGRAMS** フォルダの LAD/FBS ユニットの下の LAD/FBD をダブルクリックしてください。  
サンプルプログラムで、空の LAD/FBD プログラム **LAD\_1** が既に開いています。

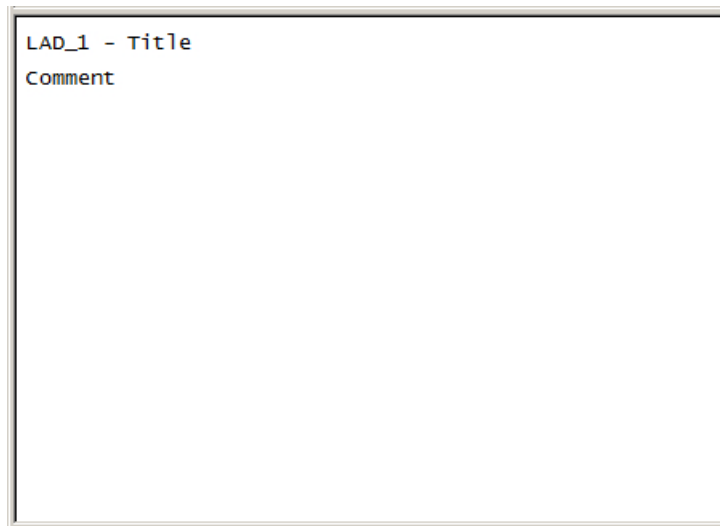


図 13-42 LAD/FBD ユニットの初回作成後の空の LAD プログラム

2. ネットワークを挿入：
  - LAD/FBD ツールバーの **Insert network** をクリックしてください。



コイル付き **001** ブロックが挿入されます。

## 13.8 LAD/FBD ラダーロジック / ファンクションブロックダイアグラム

3. コイル前に NO 接点を挿入してください :
  - コイルを選択してください。

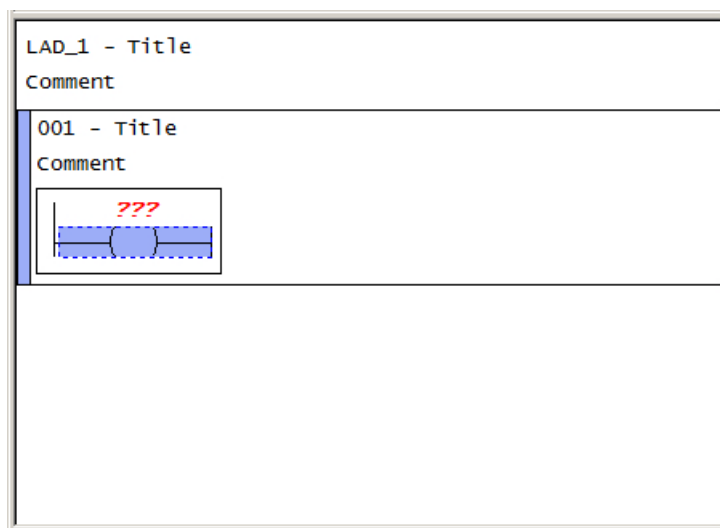


図 13-43 LAD プログラム、選択されたコイル

- ツールバーで **NO Contact** をクリックしてください。



NO 接点がコイル前に挿入されます。

4. **g\_bo\_start** 変数を NO 接点に、**q\_bo\_output0** 変数をコイルに割り付けてください！
- NO 接点およびコイル記号の上の **???** をクリックしてください。

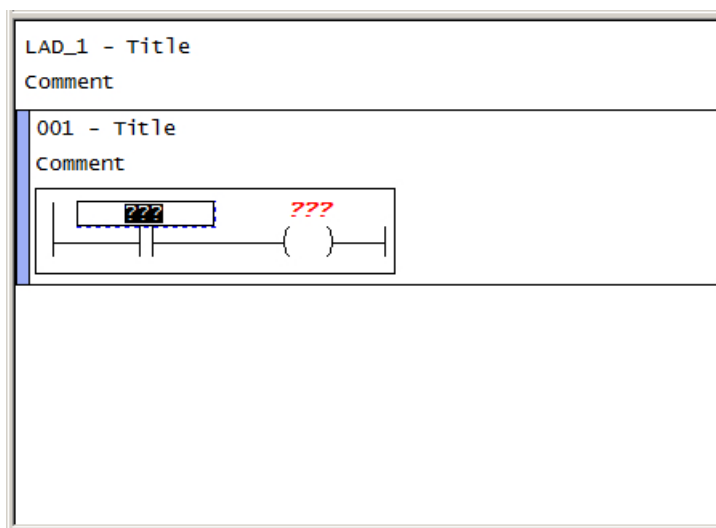


図 13-44 LAD プログラム、変数名を入力するために開かれた域

- シンボルブラウザまたは **ADDRESS LIST** から変数名をドラッグ & ドロップで転送してください。または、オートコンプリート機能を使用してください。RETURN を使用して、各入力を確認してください。

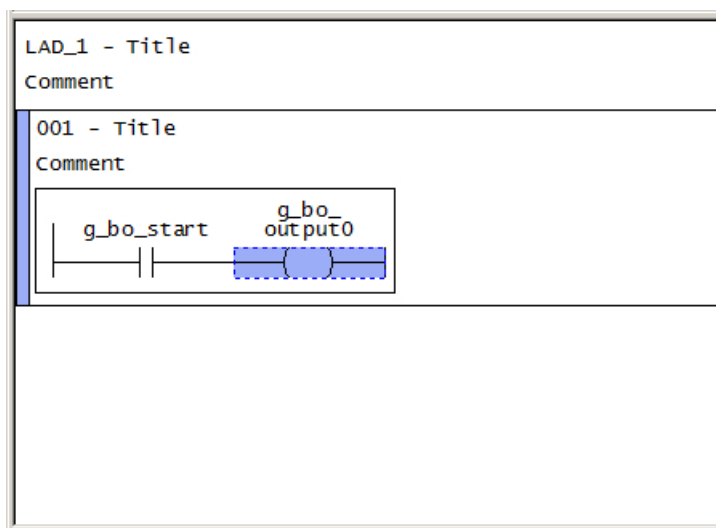


図 13-45 挿入された変数を含む LAD プログラム

5. ネットワーク **002** を挿入：
- このブロックを選択するために、ブロック **001** をクリックしてください。選択されたブロックは、左端が青くなります。
  - LAD/FBD ツールバーの **Insert network** をクリックしてください。ブロック **002** が、ブロック **001** の後に挿入されます。

6. ネットワーク **002** のコイル前に **NO** 接点を挿入してください。
7. **g\_bo\_ready** 変数を **NO** 接点に、**q\_bo\_output1** 変数をコイルに割り付けてください。  
LAD プログラムは完成しました。

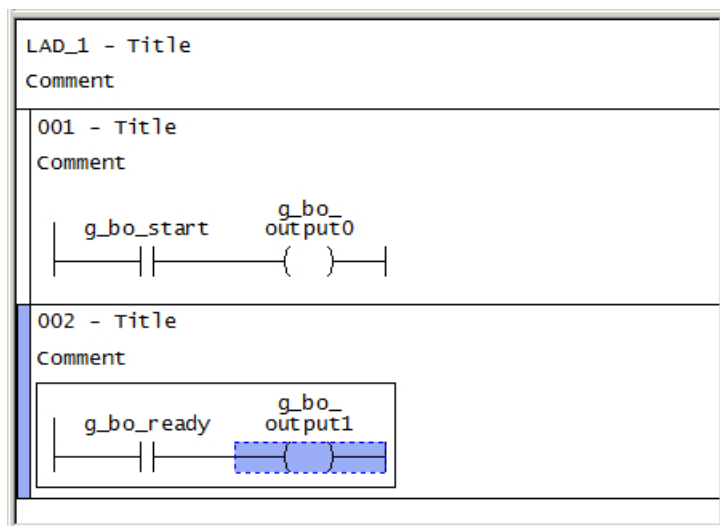
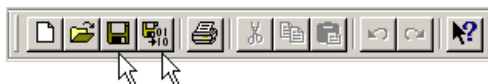


図 13-46 完成した LAD program LAD\_1

### 13.8.5 LAD/FBD サンプルプログラムのバックアップ

作成した LAD/FBD サンプルプログラムをバックアップします。

これを行うには、ツールバーの **Save project** または **Save project and compile changes** ボタンをクリックしてください。



#### 関連情報

コマンド **Save project and compile changes** の他の選択肢として、LAD/FBD ツールバーにコマンド **Accept and compile** があります。



このコマンドは、現在選択されたプログラムおよび同じユニットの他のプログラムのすべてをコンパイルします。

しかしながら、このコマンドは変更を保存しません。



このように、プロジェクト全体を再び保存またはコンパイルすることなく、プログラムへの変更を受け付けるオプションがあります。

## 13.9 その他のプログラミングオプション

SIMOTION SCOUT TIA で、MCC や LAD/FBD に加えて、他のプログラミング言語が提供されます。

- **ST (ストラクチャードテキスト)**

ST はテキストベースで、PASCAL ベースのプログラミング言語です。言語は、国際規格 IEC 61131-3 に基づくものです。この規格は、プログラマブルロジックコントローラ (PLC) のプログラミング言語の整合性をはかります。ST は、この規格の構造化テキスト部分に基づいています。IEC 61131-3 に準拠して標準化されたプログラミング言語に加えて、SIMOTION ST には技術コマンドが備わっています。

プログラムを作成するための使用しやすいテキストエディタも用意されています。ST コンパイラは編集されたプログラムを実行可能なコードにコンパイルし、構文エラーがあれば、プログラム行とエラーの原因を明確に示します。

プロジェクトナビゲータのコマンドライブラリタブで、プログラミングに必要なコマンドおよび機能がツリーで示されます。プログラミングの際には、すべてのプログラミング言語でこれらを使用することができます。

---

### 注記

SIMOTION SCOUT V4.5 以降では、ST は、オブジェクト指向のプログラミング用言語としても使用できます。他の情報は、SIMOTION SCOUT オンラインヘルプや SIMOTION ST Structured Text Programming および運転マニュアルに記されています。

---

- **ライブラリ**

ライブラリで、モジュラー構造のソフトウェア開発が可能になり、すべての SIMOTION デバイスから使用可能なユーザ定義のデータタイプ、機能やファンクションブロックが提供されます。

ライブラリはすべてのプログラミング言語で作成でき、すべてのプログラムソース (例えば、ST ユニット、MCC ユニット) で使用できます。

すべてのプログラミング言語で、アプリケーションが管理可能で再利用可能となるように、プログラム、機能やファンクションブロックを使ってアプリケーションを構造化することができます。

プログラムを試験するために、すべての言語でステータスプログラムとブレークポイントが利用可能な包括的な試験機能が利用いただけます。プログラムをオンラインで視覚化および試験することができます。

### 13.10 サンプルプロジェクトでの結果

プログラミング言語に関する詳細情報については、それぞれのマニュアルを参照してください。

## 13.10 サンプルプロジェクトでの結果

Getting Started のサンプルプロジェクトで軸のプログラム制御された移動に必要なプログラムを作成しました。

## 実行システムのコンフィグレーション

### 14.1 概要

#### Getting Started の目的

Getting Started の本項では、SIMOTION 実行システムを扱います。

サンプルプロジェクトプログラムを実行システムのタスクに割り付けます。こうして、プログラムを SIMOTION SCOUT TIA エンジニアリングシステムから SIMOTION ランタイムシステムに転送します。このプログラムは、SIMOTION デバイスの運転モードが RUN に切り替えられると直ちに、システムを制御します。

#### 要件

サンプルプロジェクト用に、プログラム `pos_axis`、`technology_fault`、`peripheral_fault`、および、`LAD_1` を作成済み。SIMOTION アプリケーションのプログラミング (ページ 83) を参照。

#### 実行レベルとタスク

実行レベルは、実行システム内におけるタスクの時系列シーケンスを定義します。1 レベルには複数のタスクが含まれます。タスクは、プログラム実行のためのフレームワークを提供します。タスクは、複数のプログラムで構成できます。作成したプログラムをタスクに割り付けることで、例えば、優先度、タイムフレーム、またはプログラムが実行される順序を定義できます。

### 14.2 プログラムをタスクに割り付け

以下で、サンプルプロジェクトプログラムを実行システムのタスクに割り付けます。


表 14-1 プログラムのタスクへの割り付け

サンプルプロジェクトのプログラム	タスク
<code>pos_axis</code>	<code>MotionTask_1</code>
<code>lad_1</code>	<code>BackgroundTask</code>

## 14.2 プログラムをタスクに割り付け

サンプルプロジェクトのプログラム	タスク
technology_fault	TechnologicalFaultTask
peripheral_fault	PeripheralFaultTask

実行システムのタスクにプログラムを割り付けるには、以下の手順に従ってください

1. SIMOTION デバイスの下のプロジェクトナビゲータにある  **EXECUTION SYSTEM** をダブルクリックしてください。  
**EXECUTION SYSTEM** ウィンドウが作業域で開きます。
2. MCC プログラム **motion.pos\_axis** をタスク **MotionTask\_1** に割り付けてください：
  - 実行システムのツリーで、項目 **ExecutionLevels > OperationLevels > MotionTasks > MotionTask\_1** を選択してください。

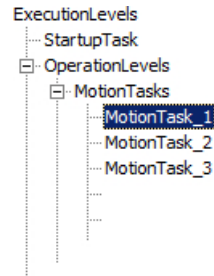


図 14-1 実行システム、実行レベルおよびタスクのツリービュー

**MotionTasks** ウィンドウは作業域の右側に表示されます。プログラミング **pos\_axis** および **LAD\_1** 並びに **fault** ユニットの補助プログラムは、**Programs** の **Program assignment** タブに表示されます。

## 14.2 プログラムをタスクに割り付け

- MCC プログラム **motion.pos\_axis** を選択し、>> ボタンをクリックしてください。そのプログラムは、**Programs used** に表示されます。こうして、タスク **MotionTask\_1** に割り付けられます。

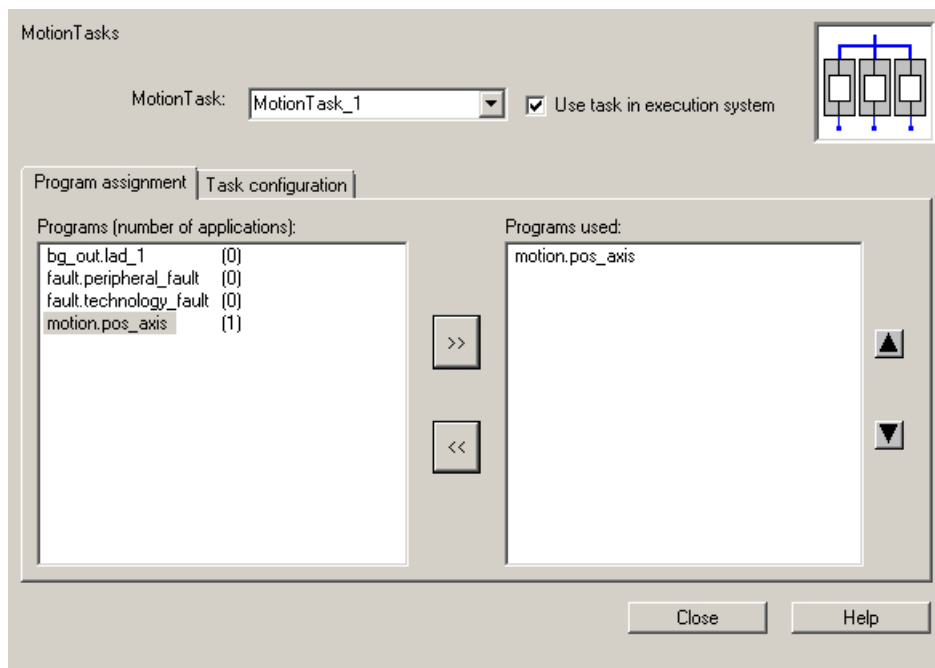


図 14-2 実行システム、MotionTasks ウィンドウ

この割り付けは実行システムのツリーに表示されます。プログラム **motion.pos\_axis** は、項目 **MotionTask\_1** の下に表示されます。

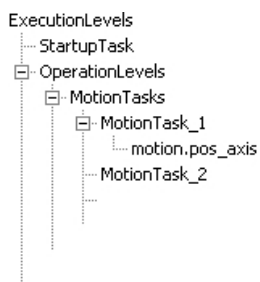


図 14-3 実行システム、割り付けられたプログラム pos\_axis を含む MotionTask\_1

- **Task configuration** タブのチェックボックス **Activation after StartupTask** を有効化してください。これは、SIMOTION デバイスの起動後直ちに MCC プログラムを実行します。このチェックボックスが有効でない場合、プログラムは明示的に呼び出され、**StartupTask** または別の有効なタスクに割り付けられた別のプログラムから起動する必要があります。
3. LAD プログラム **bg\_out.kop\_1** をタスク **BackgroundTask** に割り付けてください：
- 実行システムのツリーで、項目 **ExecutionLevels > OperationLevels > BackgroundTask** を選択してください。LAD プログラム **bg\_out.kop\_1** をこのタスクに割り付けてください。

## 14.3 コンフィグレーションされた実行システムのターゲットシステムへのダウンロード

4. エラー処理ルーチンを割り付けてください：
  - 実行システムのツリーで、項目 **ExecutionLevels > OperationLevels > SystemInterruptTasks > TechnologicalFaultTask** を選択してください。MCC プログラム **fault.technology\_fault** をこのタスクに割り付けてください。
  - 実行システムのツリーで、項目 **ExecutionLevels > OperationLevels > SystemInterruptTasks > PeripheralFaultTask** を選択してください。MCC プログラム **fault.peripheral\_fault** をこのタスクに割り付けてください。
5. **Close** をクリックしてください。保存するかどうか尋ねられる場合には **Yes** で確定してください。  
実行システムのコンフィグレーションが完了しました。

## コンフィグレーションされた実行システムのターゲットシステムへのロード

プロジェクトを保存し、コンパイルしてください。オンラインにしてください。コンフィグレーションされた実行システムを含むサンプルプロジェクトを、ターゲットシステムにダウンロードしてください。

## 14.3 コンフィグレーションされた実行システムのターゲットシステムへのダウンロード

コンフィグレーションされた実行システムを含むサンプルプロジェクトを、ターゲットシステムにダウンロードしてください。

## 手順

1. プロジェクトの保存および変更部のコンパイル
2. オンライン接続の確立
3. ターゲットシステムへのプロジェクトのダウンロード

## 変更の保存およびコンパイル

ツールバーの **Save project and compile changes** ボタンをクリックしてください。



## オンライン接続の確立

ツールバーの **Connect to selected target devices** ボタンをクリックしてください。



## 14.4 サンプルプロジェクトでの結果

### ターゲットシステムにダウンロード

ツールバーの **Download project to target system** ボタンをクリックしてください。



## 14.4 サンプルプロジェクトでの結果

軸制御のコンフィグレーションはこれで完了です：

- サンプルプロジェクトで軸をセットアップしました。
- 軸移動のためのプログラム、並びに運転に必要なプログラムを作成しました。
- これらのプログラムを **SIMOTION** ルーチンシステムのタスクに割り付けました。

以下のコンフィグレーション手順で、**SIMOTION** デバイスでの軸制御を開始し、プログラム制御された軸動作を監視することになります。



## システムの開始および停止

### 15.1 概要

#### Getting Started の目的

**pos\_axis** プログラムの実行を開始するために、サンプルプロジェクトの SIMOTION CPU を RUN モードに切り替えます。ハードウェア上で、軸が約 4 秒間に渡って 2 度回転するのを見ることができます。プログラムの実行後、STOP 状態に戻してください。

ダイアログ **Control Operating Mode** で運転モードを切り替えます。

#### 要件

- サンプルプロジェクト用にプログラム **pos\_axis**、**technology\_fault**、**peripheral\_fault** および **LAD\_1** を作成済み、および、それらを SIMOTION 実行システムのタスクに割り付け済み。SIMOTION アプリケーションのプログラミング (ページ 83) および 実行システムのコンフィグレーション (ページ 131) を参照。
- プロジェクトがコンパイルされ、ターゲットシステムにダウンロード済み。コンフィグレーションされた実行システムのターゲットシステムへのダウンロード (ページ 135) を参照。
- SIMOTION SCOUT TIA はオンラインモード。

### 15.2 RUN および STOP 動作モード

#### 動作モード

SIMOTION SCOUT TIA ダイアログ **Control Operating Mode** で、SIMOTION CPU を RUN、または STOP モードに切り替えることができます。

#### RUN モード

SIMOTION は、ユーザプログラムおよび関連システムサービスを実行します：

- プロセスイメージ入力を読み取り
- 実行システムに割り付けられるユーザプログラムを実行
- 書き込み処理イメージ出力

### 15.3 ソフトウェア側とハードウェア側のモードセレクトスイッチ

この状態で、テクノロジーパッケージは有効です。それらは、ユーザプログラムからのコマンドを実行することができます。

#### STOP モード

SIMOTION は任意のユーザプログラムを処理しません。

- ユーザプログラム全体をロードすることができます。
- すべてのシステムサービス (通信など) が有効です。
- I/O モジュールは安全な状態です (つまり、例えば、デジタル出力は "LOW level" でアナログ出力はゼロ電流 / 電圧状態です)
- テクノロジーパッケージは無効です。つまり、すべてのイネーブルが取り消されます。軸の動作は実行できません。

**SIMOTION device:Operating mode** のオンラインヘルプに運転状態の完全な説明があります。

## 15.3 ソフトウェア側とハードウェア側のモードセレクトスイッチ

#### [Control Operating Mode] ダイアログ

メニューコマンド **Target system > Control Operating Mode** または、ワークベンチのツールバーにあるボタン **Control Operating Mode** がダイアログ **Control Operating Mode** を開きます。



このダイアログはすべてのコンフィグレーションされた CPU をリストアップします。State 列には、現在の運転モードが表示されます。Control 列に割り付けられたスイッチは、CPU を RUN または STOP モードに切り替えます。CPU がオフラインモードの場合、スイッチは無効です。

## 15.3 ソフトウェア側とハードウェア側のモードセクタスイッチ

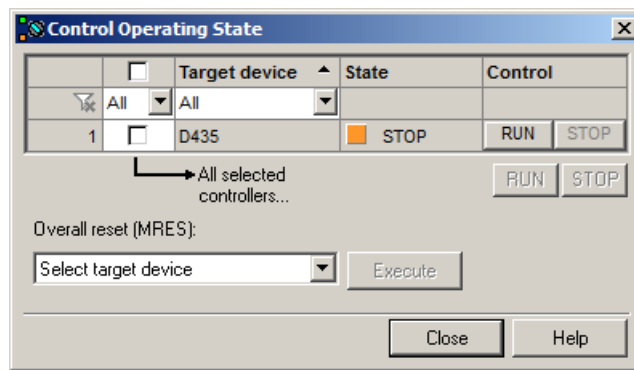


図 15-1 [Control Operating Mode] ダイアログ

## SIMOTION デバイスのモードセクタスイッチの優先度

SIMOTION デバイスのモードセクタスイッチ設定には優先度があります。SIMOTION SCOUT TIA は、デバイス上のモードセクタスイッチが **0 (RUN)** に設定されている場合にのみ、SIMOTION デバイスを RUN 運転モードに切り替えることができます。

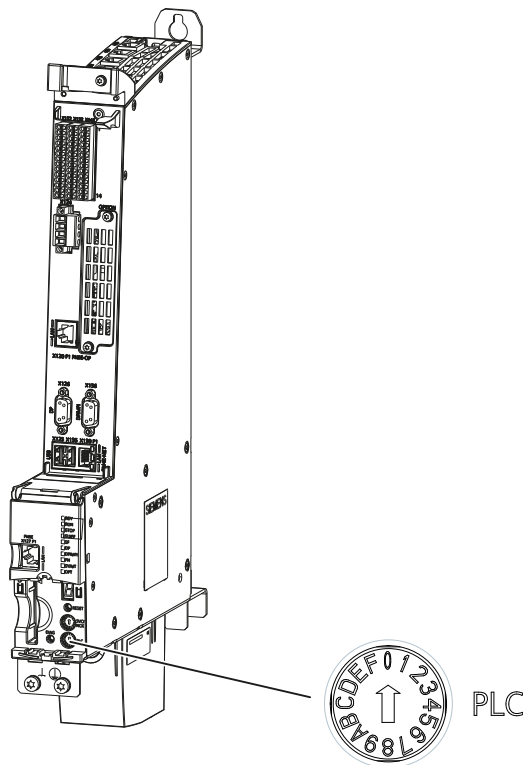


図 15-2 SIMOTION D435-2 のモードセクタスイッチ、スイッチ位置 0 (RUN)

SIMOTION D4x5-2 デバイスのモードセクタスイッチに関する詳細は、取扱説明書および試運転マニュアル、並びにオンラインヘルプにあります。

## 15.4 サンプルプロジェクトのプログラム制御の開始

SIMOTION デバイスのサンプルプロジェクトで軸の制御を開始するには、以下の手順に従ってください

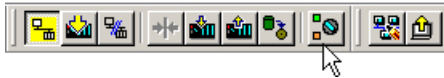


### 警告

不意の機械動作による死亡事故の危険性

これにより、オペレータ (人) または資産 (物) に危険がないことを確認してください。

1. **Control Operating Mode** ダイアログを開いてください。これを行うには、ツールバーの **Control Operating Mode** をクリックしてください。



2. サンプルプロジェクトの SIMOTION CPU を RUN モードに切り替えます。これを行うには、**Control Operating Mode** ダイアログの **Control** 列にある割り付けられた **RUN** スイッチをクリックしてください。  
軸は直ちに回転を開始します。軸は約 8 秒後に静止状態に戻ります。  
SIMOTION デバイスは、**pos\_axis** プログラムの実行後、RUN モードを維持します。
3. SIMOTION デバイスを STOP モードに切り替えてください。これを行うには **Control** 列の割り付けられた **STOP** スイッチをクリックしてください。

---

### 注記

軸を再び移動させたい場合、[Alarms] ウィンドウのすべてのアラームを確認してください。

---

## アプリケーションの監視

### 16.1 概要

#### Getting Started の目的

Getting Started の本項では、プログラム制御された軸動作を監視します。

- プログラムの実行を監視します。
- シンボルブラウザで、これらの値を監視します。
- ウォッチテーブルの一定の値をコンパイルします。
- トレースで軸動作の軌道を記録します。

#### 要件

- サンプルプロジェクト用にプログラム `pos_axis`、`technology_fault`、`peripheral_fault` および `LAD_1` を作成済み、そして、それらを **SIMOTION** 実行システムのタスクに割り付け済み。実行システムのコンフィグレーション (ページ 131) を参照。
- プロジェクトがコンパイルされ、ターゲットシステムにダウンロード済み。コンフィグレーションされた実行システムのターゲットシステムへのダウンロード (ページ 135) を参照。
- **SIMOTION SCOUT TIA** はオンラインモード。
- **SIMOTION** デバイスの運転モード：
  - サンプルプロジェクトの **SIMOTION** デバイスはソフトウェア側で **STOP** モードに切り替え済み。**SIMOTION SCOUT TIA** モードセクタスイッチ、**Control Operating Mode** ダイアログは **STOP** モードを表示します。
  - サンプルプロジェクトの **SIMOTION** デバイスはハードウェア側で **RUN** モードに切り替え済み。**SIMOTION** デバイスのモードセクタスイッチは、位置 **0 (RUN)** です。

### 16.2 プログラム実行の監視

**SIMOTION SCOUT TIA** には、プログラム実行を監視するための複数の機能があります。

## 16.2 プログラム実行の監視

サンプルプロジェクトで、機能 **Monitor** を使用します。この機能を使って、**MCC** チャートのコマンドブロックの実行をトラックできます。最近実行されたコマンドブロックは、そのチャートの黄色い背景に示されます。このマーキングは、実際のプログラム実行と同じ速度で動作します。このマーキングは従って、機械装置のコマンド実行を待機する "slow" コマンドの場合にのみ視認できます。

**MCC** チャート **pos\_axis** でのコマンド実行を監視するには、以下の手順に従ってください

1. **Control Operating Mode** ダイアログを開いてください。



SIMOTION デバイスは、**STOP** モードでなければなりません。

2. **MCC** チャート **pos\_axis** を開いてください。
3. 監視機能をオンにしてください：
  - これを行うには、ワークベンチのメニューバーにあるコマンド **MCC chart > Monitor** を選択してください。この機能は、メニュー項目のとなりのチェックマークが表示されている場合、オンになります。
  - または、**Monitor** の **MCC** エディタ ツールバーをクリックしてください。



4. プログラムコントロールを開始： **Control Operating Mode** ダイアログボックスで、SIMOTION デバイスを **RUN** モードに切り替えてください。  
**pos\_axis** プログラムが一度実行されます。最近実行されたコマンドブロックが黄色い背景に示されます。

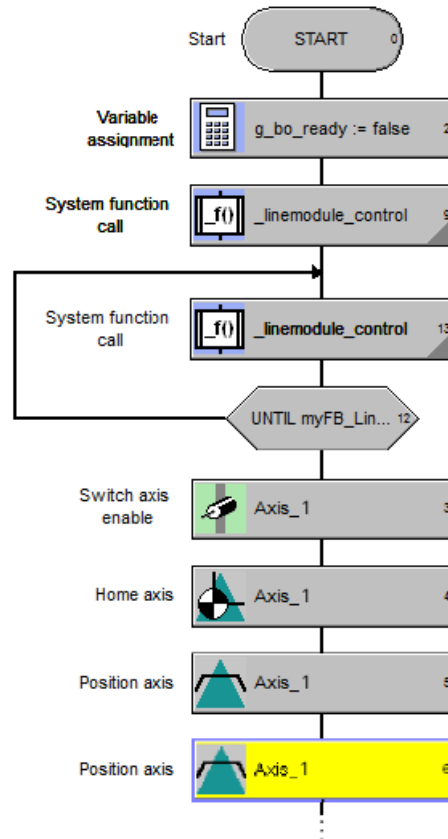


図 16-1 MCC チャート pos\_axis のプログラムの実行を監視

5. SIMOTION デバイスを **Control Operating State** ダイアログボックスで **STOP** モードに切り替えてください。

## 関連情報

プログラム監視に関する詳細は、**Monitoring program execution** のオンラインヘルプにあります。

## 16.3 変数の監視

シンボルブラウザでは、変数 (状態値の読み出し) の監視、または、それらへの値の割り付け (制御値の割り付け) を実行することができます。

サンプルプロジェクトでは、プログラムの実行中に、軸の現在の位置を監視します。

## 16.3 変数の監視

## シンボルブラウザの変数を監視

1. プロジェクトナビゲータの **AXES** フォルダで、サンプルプロジェクトで作成された軸を選択してください。  
軸のシステム変数およびコンフィグレーションデータは、Detail view の **Symbol browser** タブに表示されます。
2. シンボルブラウザのシステム変数 **positioningstate.actualposition** (軸の実際の位置) を開いてください。  
変数を確認するには、以下の手順に従ってください：
  - このリストをフィルタリング：システムブラウザのフィルタラインでフィルタ基準を指定できます。最後の 5 基準は保存され、再利用のために選択することができます。  
適切なフィルタのためのワード、例えば、**positioningstate** をフィルタラインに入力してください。**RETURN** を押し、確定してください。
  - 変数検索：このフィルタ機能の代わりに、変数を検索することもできます。メニューコマンド **Edit > Find** を選択してください。**Find** ダイアログボックスに、例えば、適切な検索ワード **positioningstate** を入力してください。**Find next** をクリックしてください。現在の軸の位置は、シンボルブラウザの **Status value** 列に表示されます。
3. ダイアログボックス **Control Operating Mode** で、プログラム制御を開始してください。プログラムが一度実行されます。この時間中、実際の軸位置の値の変更は、シンボルブラウザに表示されます。  
プログラムがこの動作を完了した後、SIMOTION デバイスは RUN 状態です。
4. SIMOTION デバイスを **Control Operating State** ダイアログボックスで **STOP** モードに切り替えてください。

## ウォッチテーブルでの変数の監視

複数のデバイスの異なる変数でさえ、同時にそれらを監視するために、1 つの表に組み合わせることができます。

変数をウォッチテーブルに追加するには、最初にウォッチテーブルを作成する必要があります。

以下の手順に従ってください：

1. **MONITOR** の下のプロジェクトナビゲータで、エントリ **Insert watch table** をダブルクリックし、コンフィグレーションを確定してください。
2. システムブラウザで、ウォッチテーブルに追加したい変数を右クリックしてください。ショートカットメニューの **Add to watch table** コマンドを選択してください。  
選択された変数はその表に表示されます。こうして、他の変数をウォッチテーブルに追加し、それらを監視することができます。



ウォッチテーブルを開くには、以下の手順に従ってください

プロジェクトナビゲータの **MONITOR** フォルダに、作成されたすべてのウォッチテーブルがあります。それを開くために、ウォッチテーブルをダブルクリックしてください。

この項目に関する詳細は、**Working with the SCOUT Workbench > Working with lists > Watch table** のオンラインヘルプにあります。

## 16.4 トレース機能での信号の記録

### 16.4.1 トレース機能

トレースを使用すれば、信号特性と変数値を記録して保存できます。例えば、記録したデータは、機械運動シーケンスの診断、およびユーザプログラムでのトラブルシューティングに使用できます。

サンプルプログラムの場合、時間と共に軸の位置実績値を記録し、表でそれを表示します。

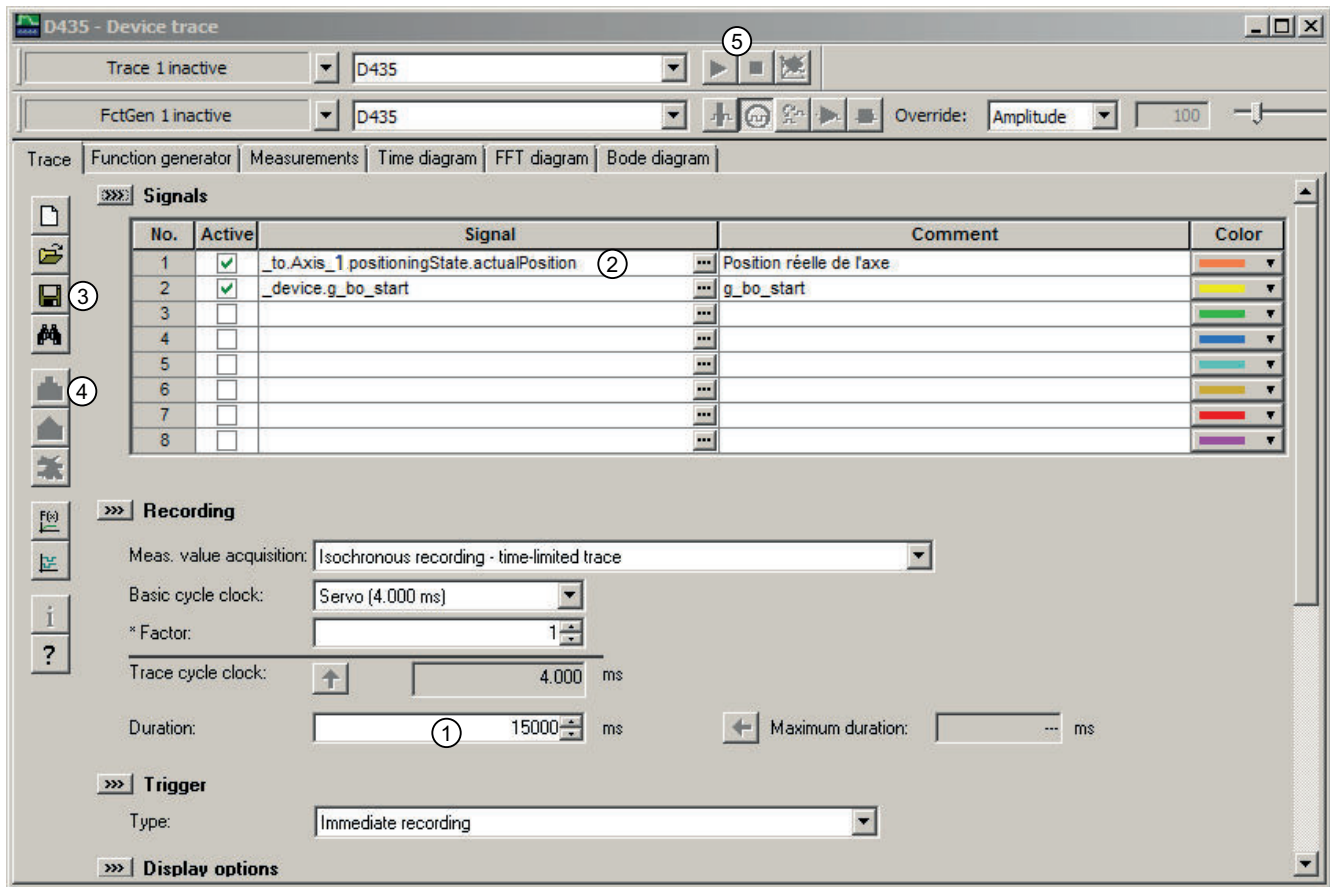
## 16.4.2 トレース機能を使った作業

以下の手順で、トレースを行ってください

1. プロジェクトナビゲータで、SIMOTION デバイスをハイライト表示してください。
2. メニューコマンド **Target system > Device trace** を選択、または **Trace ツールバー** で **Device trace function generator** ボタンをクリックしてください。



**Device trace** ウィンドウが作業域に表示されます。



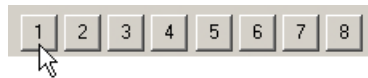
① ... ⑤ 丸数字は以下のテキストで参照されています。

図 16-2 デバイストレースウィンドウ

記録用のトレース機能のパラメータ設定は、以下の手順に従ってください

1. **Trace** タブの **Duration** 域 ① に、記録時間 15000 ms を入力してください。
2. **Signals** テーブルのライン **No 1** で、ボタン ... ② をクリックしてください。  
**Trace Signal Selection** ウィンドウが開きます。

- このツリーで、項目 **Sample\_1** > (SIMOTION デバイス、例えば **D435**) > **TO** > **Axis\_1** > **positioningstate** を選択してください。
- 変数表のシステム変数 **actualPosition** を選択してください。
- チャンネル 1 にシステム変数を受け付けるためにボタン **1** をクリックしてください。



変数が **Signal name** に表示されます。

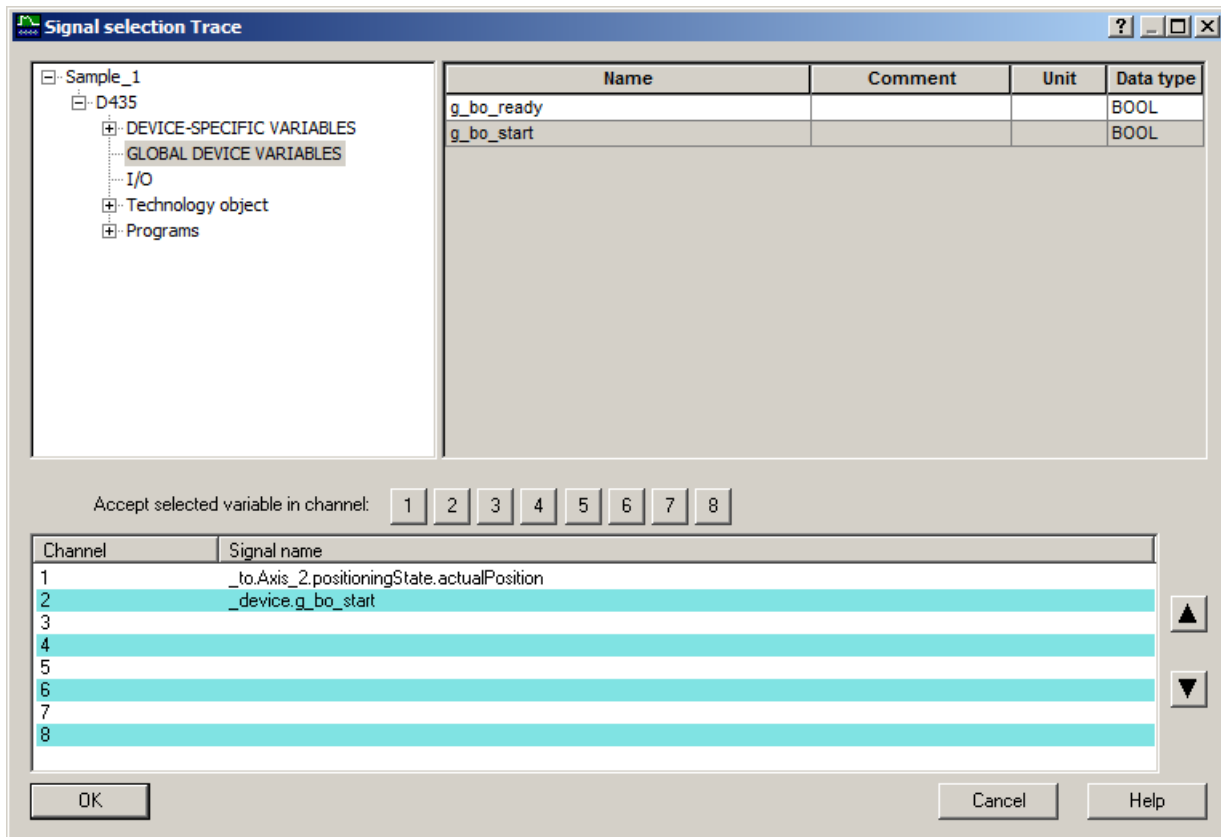


図 16-3 トレース、信号選択

- 変数 **g\_bo\_start** 用のチャンネルの割り付けを繰り返してください：  
項目 (プロジェクト) > (SIMOTION デバイス) > **GLOBAL DEVICE VARIABLES** を選択してください。変数表のシステム変数 **g\_bo\_start** を選択してください。ボタン **2** をクリックしてください。

#### 注記

シンボルブラウザまたはウォッチテーブルから変数をトレースダイアログの信号域にドラッグ & ドロップすることができます。ドラッグ & ドロップに関しては、変数の割り付け **g\_bo\_ready:=false / g\_bo\_start:=true** (ページ 97) を参照してください。

- OK** で確定してください。  
**Trace Signal Selection** ウィンドウが閉じます。  
これで、トレース機能は記録用にパラメータ設定されました。

以下の手順で、トレース機能のパラメータ設定を保存します

トレース機能のパラメータ設定はプロジェクトデータには保存されません。このプロジェクトを閉じると、トレース機能のパラメータ設定は削除されます。

パラメータを保存するには、**Trace** タブの **Accept in catalog** ボタン ③ をクリックしてください。**Catalog entry** 域に、トレースのカタログに保存される設定 / パラメータ設定名を入力してください。

トレースを使っでの記録は、以下の手順に従ってください

1. オンラインにしてください。
2. ターゲットシステムにトレースのパラメータ設定をダウンロード：
  - **Trace** タブの **Download parameterization** ボタン ④ をクリックしてください。



- ダウンロードが正常に完了後に現れるダイアログを承認するために **OK** をクリックしてください。

3. **Control Operating Mode** ダイアログを開いてください。



SIMOTION デバイスは、**STOP** モードでなければなりません。

4. **Device trace** ウィンドウで **Time diagram** タブを開いてください。

## 5. トレースの記録を開始：

- **Device trace** ウィンドウで、**Trace start**⑤ をクリックしてください。



- その後、直ちに、SIMOTION デバイスを **Control Operating Mode** ダイアログボックスで **RUN** モードに切り替えてください。

プログラムが開始されます。軸の位置実績値は、タイミング図に記録され、表示されます。記録時間の終了後、位置実績値の信号プロファイルが表示されます。

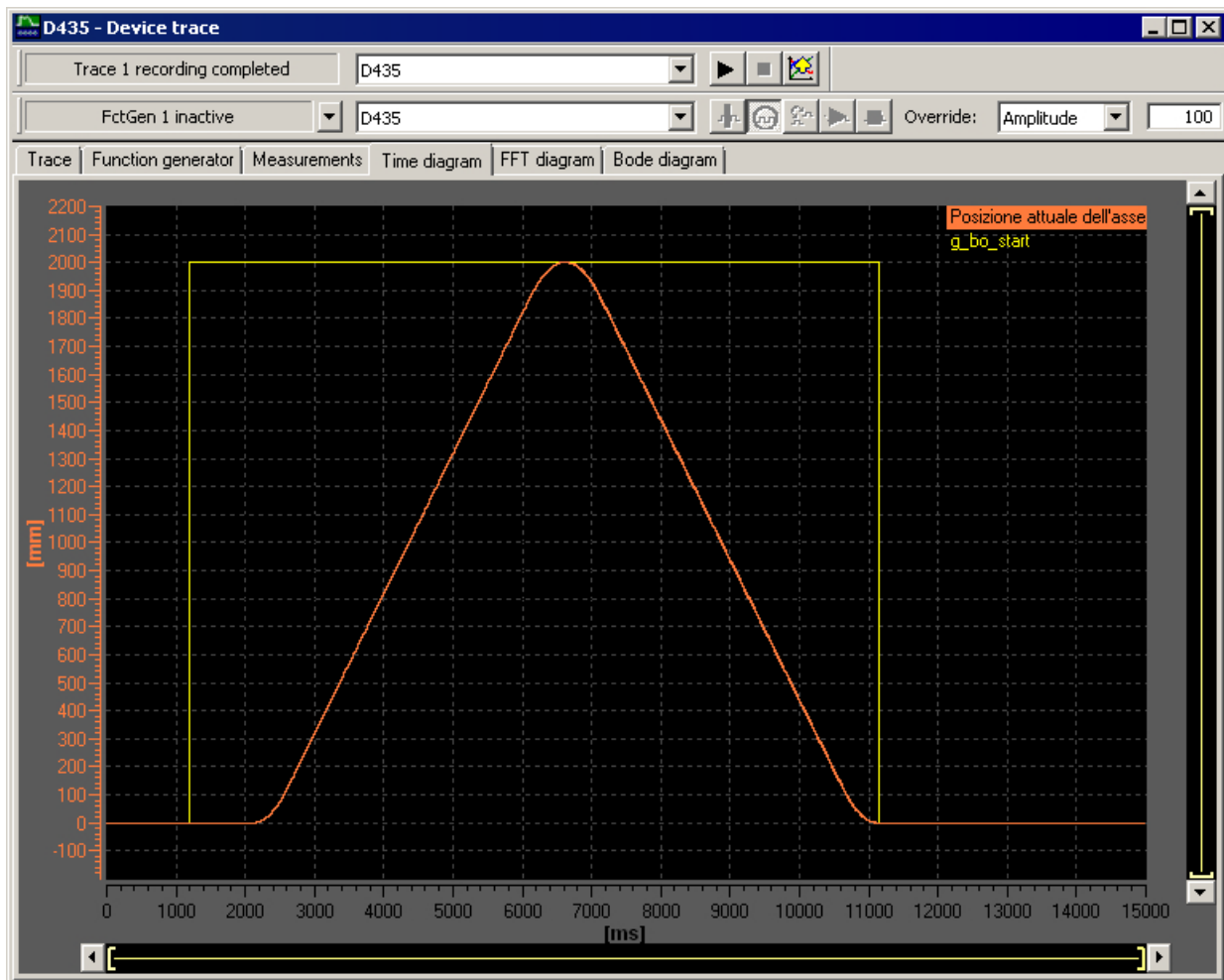


図 16-4 軸のタイミング図

6. 記録された軌道が部分的にしか表示されない場合、タイミング図のショートカットメニューでメニュー項目 **Auto-scaling** を選択してください。
7. **Control Operating Mode** ダイアログで、SIMOTION デバイスを **STOP** モードに切り替えてください。

## 16.5 サンプルプロジェクトでの結果

### 複数の測定の実行

複数の測定を実行することができます。それらは、**Measurements** タブに示され、図にも表示させることができます。

記録された測定は、文書での使用目的で保存および再度開くことができます。

## 16.5 サンプルプロジェクトでの結果

トレースのタイミング図の作成に関して、**Getting Started** での説明はこれで終わりです。

## ESD に関する指示

### A.1 ESD の定義

#### ESD の意味

静電気の影響を受けやすいデバイス (ESD) とは、静電磁界または静電気放電で破損する可能性のあるコンポーネント、IC 回路、モジュールまたはデバイスのことです。



#### 通知

##### 電磁界または静電気放電による破損

電磁界または静電気放電は、個々のパーツ、IC、モジュールまたは機器の破損の結果として誤作動に至ることがあります。

- 電気コンポーネント、モジュールまたは機器は、オリジナルの梱包材または他の適切な素材、例えば、導電性気泡ゴムまたはアルミ箔で梱包、保管、輸送および搬送してください。
- 以下の対策のいずれかを講じて接地されている場合にのみ、コンポーネント、モジュールおよび機器に触れてください:
  - － 帯電防止リストストラップの使用
  - － 導電性床材の ESD 保護区域での静電靴または ESD 接地ストラップの使用
- 導電性表面にのみ電気コンポーネント、モジュールまたは機器を置いてください (ESD 表面の作業台、導電性 ESD 気泡ゴム、ESD 梱包、ESD 搬送コンテナ)。

### A.2 人体の静電気蓄積

周囲の電位に接続していなければ、誰でも静電荷が蓄積する可能性があります。

この図は、特定の材料に触れた場合に、操作要員の身体に蓄積されうる最大静電量を示したものです。これらの数値は、IEC 801-2 の仕様に従っています。

## A.3 静電放電に対する基本的保護措置

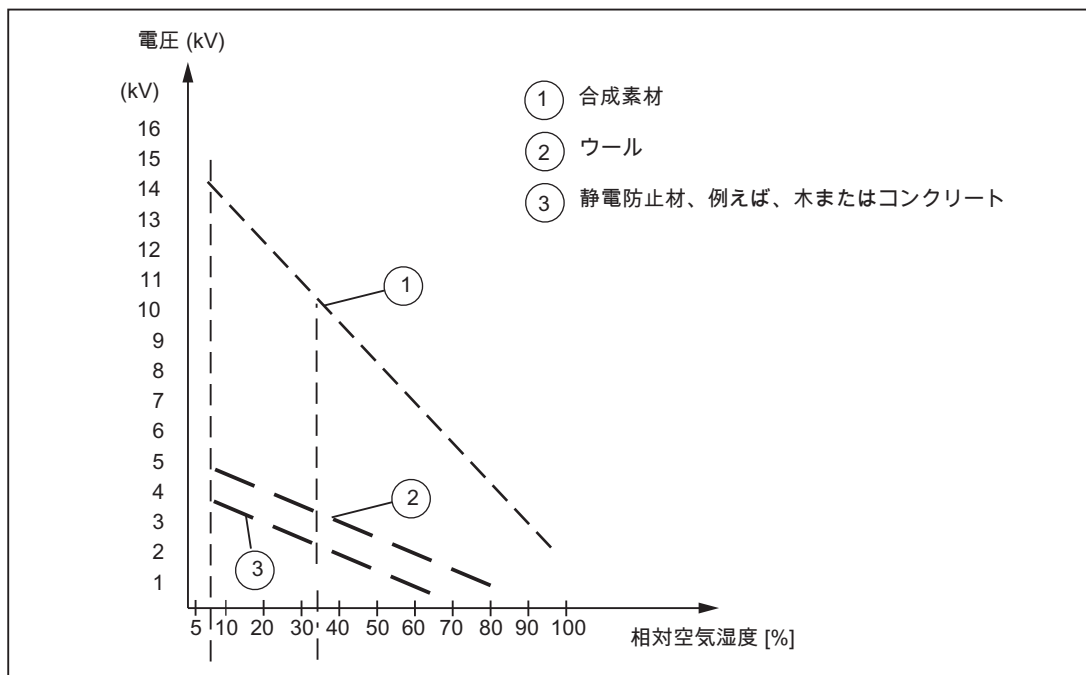


図 A-1 操作要員に蓄積する可能性がある静電電圧

## A.3 静電放電に対する基本的保護措置

## 十分な接地

静電放電により破損するおそれのあるデバイスを取り扱う場合は、作業員、ワークステーション、および梱包が正しく接地されていることを確認します。これにより、静電気の蓄積を防ぐことができます。

## 直接の接触の回避

やむを得ない場合を除き、ESD コンポーネントに触れないようにします(例えば、保守作業)。モジュールに触れる際は、モジュールのピンまたは印刷された導体に触れないでください。これらの指示に従えば、静電放電が敏感なコンポーネントに達したり、損傷を与えたりすることはありません。

モジュールの測定が必要な場合は、身体に蓄積した静電気を最初に必ず放電してください。これを行うには、接地した金属製の物体に触れます。接地された測定器だけを使用してください。



# 索引

## E

ESD ガイドライン, 151

## F

FBD, 121

## G

Getting Started (SCOUT TIA 用)

PG/PC 通信を設定, 33

プロジェクトで SIMOTION デバイスを作成, 33

Getting Started (SIMOTION SCOUT TIA 用)

I/O 変数の作成, 87

LAD サンプルプログラムの作成, 124

LAD/FBD プログラムの作成, 123

LAD/FBD ユニットの作成, 123

MCC サンプルプログラム、ベーシックフレームワーク, 96

MCC サンプルプログラム、電源装置, 111

MCC チャートの作成, 93

MCC ユニットの作成, 92

SIMOTION アプリケーションのプログラミング, 83

アプリケーションの監視, 141

グローバルデバイス変数の作成, 85

コンフィグレーションされた実行システムのターゲットシステムへのダウンロード, 135

サンプルプロジェクト、コンフィグレーション手順, 15

サンプルプロジェクトのプログラミング言語, 91

サンプルプロジェクトの変数, 84

システムの開始および停止, 137

システムファンクションの呼び出し

\_LineModule\_control[FB], 113

シンボルブラウザの変数を監視, 144

ターゲットシステムへのプロジェクトのダウンロード, 56, 73, 135

デジタル出力をコンフィグレーション, 81

ドライブのコンフィグレーション, 59

トレース機能での信号の記録, 145

プログラムをタスクに割り付け, 131

プログラム制御で電源装置をオン, 111

プロジェクトの保存および変更部のコンパイル, 51

概要, 15

軸コントロールパネルでの軸の試験, 75

軸のコンフィグレーション, 69

軸のコンフィグレーションをターゲットシステムにダウンロード, 73

軸の作成, 70

実行システムのコンフィグレーション, 131

新しいプロジェクトを作成, 29

選択されたターゲットデバイスに接続, 52

電源装置のコンフィグレーション, 65

変数タイプ, 84

## I

I/O 変数の作成, 87

## L

LAD, 121

## M

MCC, 92

システムイベントの処理, 119

軸イネーブルへの切り替え, 102

軸のターゲット位置への位置決め, 105

軸の開始位置への位置決め, 107

軸の原点復帰, 104

軸の無効化, 109

MCC エディタツールバー, 94

MCC チャートの作成, 93

MCC ユニットの作成, 92

## P

peripheral\_fault, 120

PG/PC インターフェース  
設定, 38

## S

SIMOTION D, 33

SIMOTION SCOUT TIA

開始, 49

設定, 49

SIMOTION SCOUT TIA オンラインヘルプ, 19

Getting Started (SIMOTION SCOUT TIA 用)

入門書

## T

technology\_fault, 120

## X

X142, 81

## オ

オートコンプリート, 101, 116

オンライン

接続, 38

オンライン & 診断, 38

オンラインアクセス, 38

## カ

ガイドライン

ESD, 151

## ク

グローバルデバイス変数, 85

## コ

コンフィグレーション手順

SIMOTION SCOUT TIA, 18, 19

TIA ポータル, 18, 19

## サ

サンプルプロジェクト

SIMOTION SCOUT TIA" での Getting started"; "プ

ロジェクト例、コンフィグレーション手順, 16

要件, 17

## シ

システムイベントの処理, 119

シンボルブラウザ, 85

## タ

ターゲットデバイスの選択, 53

タスク, 131

## テ

デジタル出力, 81

## ト

ドライブ

自動コンフィグレーション, 60

トレース機能, 145

## フ

プログラミング言語

FBD, 121

LAD, 121

MCC, 91

ST, 129

プロジェクトのダウンロード, 57

プロジェクトの保存およびコンパイル, 51

プロジェクトの保存および変更部のコンパイル, 120

## モ

モーションコントロールチャート, 92

モードセレクトスイッチ

SIMOTION SCOUT TIA で, 138

SIMOTION ハードウェア, 139

## ユ

ユーティリティ & アプリケーション, 22

ユニット

LAD/FBD, 122

MCC, 92

## 安

安全に関する指示, 23

## 運

運転モードの制御, 138

## 参

参照先, 3

g\_bo\_start:=true, 97, 100  
LineModule\_STW:=myFB\_LineControl.periOut,  
116

## 軸

軸イネーブルへの切り替え, 102  
軸ウィザード, 69  
軸コントロールパネル, 75  
軸のテクノロジーオブジェクト, 69  
軸の作成, 70  
軸の無効化, 109  
軸をドライブに割り付け, 72

## 実

実行システム, 131  
実行レベル, 131

## 新

新しいプロジェクトを作成, 30

## 設

設定  
SIMOTION SCOUT TIA, 49

## 選

選択されたターゲットデバイスに接続, 53

## 端

端子 X142 の I/O チャンネル, 81

## 電

電源装置, 65

## 動

動作モード, 137

## 変

変数タイプ, 84  
変数の割り付け  
g\_bo\_ready:=false, 97  
g\_bo\_ready:=true, 110  
g\_bo\_start:=false, 110

