マシンオートメーションコントローラ

NJ シリーズ

ロボット統合システム

スタートアップガイド

形 NJ501-R□□□

形 AC1-152000

形 RL4-□□□□□□□

形 RX4-□□□□□□□

SBCE-493E

おことわり

(1) 本マニュアルの内容の一部または全部を無断で複写、複製、転載することを禁じます。

(2) 本マニュアルの内容に関しては、改良のため予告なしに仕様などを変更することがあります。
あらかじめご了承ください。

(3) 本マニュアルの内容に関しては、万全を期しておりますが、万一ご不審な点や誤りなど、お気づきのことがありましたら、お手数ですが、弊社支店または営業所までご連絡ください。
その際、巻末記載のMan.No.（マニュアルNo.）も併せてお知らせください。

商標

- Sysmacは、オムロン株式会社FA機器製品の日本およびその他の国における商標または登録商標です。
- Microsoft、Windows、Excel、Visual Basic、Microsoft Edgeは、米国Microsoft Corporationの米国およびその他の国における登録商標または商標です。
- EtherCAT® は、ドイツBeckhoff Automation GmbHによりライセンスされた特許取得済み技術であり登録商標です。
- ODVA、CIP、CompoNet、DeviceNet、EtherNet/IPはODVAの商標です。
- SD、SDHCロゴは、SD-3C, LLCの商標です。

その他、本文中に掲載している会社名および製品名は、それぞれ各社の商標または登録商標です。

著作権

- スクリーンショットはマイクロソフトの許諾を得て使用しています。
- inCAD LibraryのCADデータは株式会社ミスミの許可を得て使用しています。CADデータに含まれる一切の情報の著作権は株式会社ミスミまたは株式会社ミスミの取り扱いメーカーに帰属します。
株式会社ミスミは部品を組合わせての販売は実施しておりません。また、部品を組合わせた場合の品質・正確性・機能・安全性・信頼性等は一切保証いたしません。
- 本製品にはサードパーティ製のソフトウェアが組み込まれています。このソフトウェアに関連するライセンスと著作権については、http://www.fa.omron.co.jp/nj_info_j/ をご覧ください。

はじめに

このたびは、NJ シリーズ ロボット統合 CPU ユニットをお買い上げいただきまして、ありがとうございます。

本ガイドは、NJ シリーズ ロボット統合 CPU ユニットとオムロン製ロボットを組み合わせた“ロボット統合システム”の立ち上げ手順と Sysmac Studio の基本的な操作方法を、簡単な装置モデルの例で説明しています。本ガイドに沿って操作することで、ロボット統合システムの基礎を短時間で習得できます。

NJ シリーズ ロボット統合 CPU ユニット、IPC Application Controller、ロボットおよびその周辺機器の配線や動作設定については、必要に応じて「関連マニュアル (P.22)」を参照してください。

なお、本ガイドには安全に関する注意事項を含め、使用する際の詳細内容については記載していません。必ず、本ガイドで使用する各機器のマニュアルや取扱説明書を入手し、「安全上のご注意」「安全上の要点」「使用上の注意」などの安全に関する注意事項を含め、使用する際の詳細内容を確認したうえで使用してください。
また、ロボット安全に関する注意事項やロボットの取り扱いに関する詳細については、『ロボット セーフティガイド(SBCE-381)』などを確認したうえで使用してください。

対象となる読者の方々

本マニュアルは、電気の知識（電気工事士あるいは同等の知識）を有する、次の方を対象に記述しています。

- FA 機器の導入を担当される方
- FA システムを設計される方
- FA 機器を設置、接続される方
- FA 現場を管理される方

また、次の内容をご理解いただいている方を対象に記述しています。

- プログラミング言語に関して、国際標準規格 IEC 61131-3 もしくは国内標準規格 JIS B 3503 に規定される内容をご理解いただいている方*1
- NJ シリーズ CPU ユニットの標準仕様および取り扱い方法をご理解いただいている方*1
- Sysmac Studio の基本的な操作方法をご理解いただいている方*1
- 弊社が提供する産業用ロボットセミナを受講いただいた方、または同等の知識を習得された方
- ロボットの法定（労働安全衛生規則第 36 条 第 31 号、第 32 号）の特別教育を受講いただいた方

*1. IEC 61131-3 の概要、および NJ シリーズ CPU ユニットと Sysmac Studio の使い方の基礎については、「関連テクニカルガイド (P.25)」が参考になります。

対象となる製品

本マニュアルは以下の製品を対象にしています。

- NJ シリーズ ロボット統合 CPU ユニット

形 NJ501-R□□□

- SCARA ロボット eCobra 600/800

形 RL4-□□□□□□□

- IPC Application Controller

形 AC1-152000

分冊構成

本製品のマニュアルは、次表のとおりに分冊されています。目的に応じてお読みいただき、本製品を活用してください。
本製品の操作には、おもにオートメーションソフトウェア Sysmac Studio を使用します。Sysmac Studio については、『Sysmac Studio Version 1 オペレーションマニュアル (SBCA-470)』、および『Sysmac Studio ロボット統合システム構築機能 オペレーションマニュアル ロボット統合 CPU ユニット編 (SBCA-464)』を参照してください。

使用目的	マニュアル												
	基本情報												
	NJ シリーズ CPU ユニット ユーザーズマニュアル ハードウェア編	NJ/NX シリーズ CPU ユニット ユーザーズマニュアル ソフトウェア編	NJ/NX シリーズ コマンドリファレンスマニュアル 基本編	NJ/NX シリーズ CPU ユニット ユーザーズマニュアル モーション制御編	NJ/NX シリーズ コマンドリファレンスマニュアル モーション編	NJ/NX シリーズ CPU ユニット 内蔵 EtherCAT ポート ユーザーズマニュアル	NJ/NX シリーズ CPU ユニット 内蔵 EtherNet/IP ポート ユーザーズマニュアル	NJ/NX シリーズ データベース接続 CPU ユニット ユーザーズマニュアル	NJ シリーズ ロボット統合 CPU ユニット ユーザーズマニュアル	eV+3 ユーザーズマニュアル	eV+3 キーワードリファレンスマニュアル	NJ シリーズ NJ Robotics CPU ユニット ユーザーズマニュアル	NJ/NX シリーズ トラブルシューティングマニュアル
NJ シリーズの概要を知る	○												
装着、設置、ハードウェア設定を行う	○												
モーション制御を行う場合				○									
EtherCAT を使用する場合						○							
EtherNet/IP を使用する場合							○						
オムロン製ロボットの制御を行う場合									○				
ソフトウェア設定を行う		○											
モーション制御を行う場合				○									
EtherCAT を使用する場合						○							
EtherNet/IP を使用する場合							○						
データベース接続サービスを使用する場合								○					
オムロン製ロボットの制御を行う場合									○	○	○		
NJ Robotics 機能でロボット制御を行う場合												○	
ユーザプログラムを作成する		○	○										
モーション制御を行う場合				○	○								
EtherCAT を使用する場合						○							
EtherNet/IP を使用する場合							○						
データベース接続サービスを使用する場合								○					
オムロン製ロボットの制御を行う場合									○	○	○		
NJ Robotics 機能でロボット制御を行う場合												○	
異常処理を組み込む場合									○	○	○		○

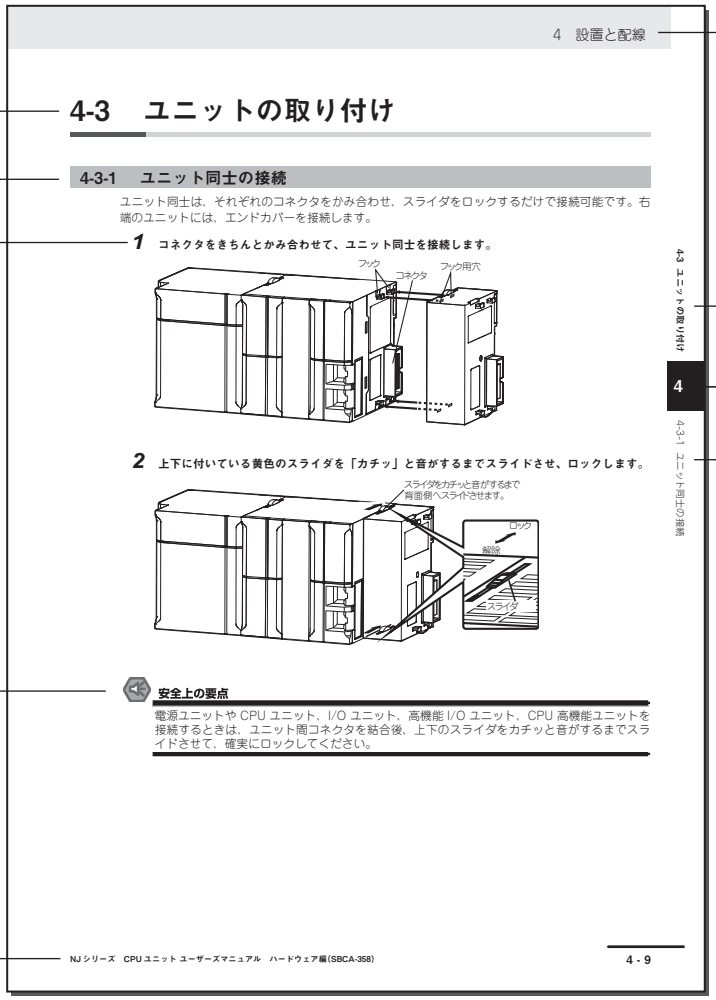
使用目的	マニュアル												
	基本情報												
	NJ シリーズ CPU ユニット ユーザーズマニュアル ハードウェア編	NJ/NX シリーズ CPU ユニット ユーザーズマニュアル ソフトウェア編	NJ/NX シリーズ コマンドリファレンスマニュアル 基本編	NJ/NX シリーズ CPU ユニット ユーザーズマニュアル モーション制御編	NJ/NX シリーズ コマンドリファレンスマニュアル モーション編	NJ/NX シリーズ CPU ユニット 内蔵 EtherCAT ポート ユーザーズマニュアル	NJ/NX シリーズ CPU ユニット 内蔵 EtherNet/IP ポート ユーザーズマニュアル	NJ/NX シリーズ データベース接続 CPU ユニット ユーザーズマニュアル	NJ シリーズ ロボット統合 CPU ユニット ユーザーズマニュアル	eV+3 ユーザーズマニュアル	eV+3 キーワードリファレンスマニュアル	NJ シリーズ NJ Robotics CPU ユニット ユーザーズマニュアル	NJ/NX シリーズ トラブルシューティングマニュアル
動作確認やデバッグを行う		○											
モーション制御を行う場合				○									
EtherCAT を使用する場合						○							
EtherNet/IP を使用する場合							○						
データベース接続サービスを使用する場合								○					
オムロン製ロボットの制御を行う場合									○	○	○		
NJ Robotics 機能でロボット制御を行う場合												○	
異常管理機能やトラブル発生時の対処方法を知る *1								△	△	△	△	△	○
保守作業について知る	○												
モーション制御を行う場合				○									
EtherCAT を使用する場合						○							
EtherNet/IP を使用する場合							○						

*1. 異常管理の考え方と異常項目については、『NJ/NX シリーズ トラブルシューティングマニュアル (SBCA-469)』を参照してください。ただし、△ マークがついているマニュアル製品に特有の異常の詳細については、△ マークがついているマニュアルを参照してください。

マニュアルの読み方

ページ構成

このマニュアルの各ページの構成は次のとおりです。



節タイトル

項タイトル

操作手順

操作手順を示します。

注記、補足、参照先

注記、補足、参照先などの各項目を、アイコンによって示しています。

マニュアル名称

章タイトル

節タイトル

項タイトル

現在のページの章／節／項タイトルをそれぞれ示します。

章番号

現在のページの章番号を示します。

このページは、説明用の見本です。実際の内容とは異なります。

アイコン

このマニュアルで使われているアイコンには、次のような意味があります。



安全上の要点

製品を安全に使用するために実施または回避すべきことを示します。



使用上の注意

製品の動作不能、誤動作、または性能・機能への悪影響を予防するために実施または回避すべきことを示します。



参考

必要に応じて読んでいただきたい項目です。
知っておくと便利な情報や、使用する上で参考となる内容について説明しています。



バージョン関連情報

コントローラや Sysmac Studio の、バージョンによる性能や機能の違いについて説明しています。

表記についての留意事項

本マニュアルでは、Sysmac Studio から実機コントローラにデータを転送することを“ダウンロード”、実機コントローラからデータを Sysmac Studio に転送することを“アップロード”と記述しています。
Sysmac Studio では、“ダウンロード”および“アップロード”は、いずれも“同期”機能を使用します。Sysmac Studio の“同期”機能は、パソコン上のデータと実機コントローラのデータを自動照合し、どちらの方向に転送するかをユーザが選択する機能です。

目次構成

1 概要

2 事前に準備すること

3 静的ピック & プレース装置の実現例

4 動的ピック & プレース装置の実現例

A 付録

目次

はじめに **1**
対象となる読者の方々 1
対象となる製品 1

分冊構成 **2**

マニュアルの読み方 **4**
ページ構成 4
アイコン 5
表記についての留意事項 5

目次構成 **7**

ご承諾事項 **11**

安全上の注意 **14**
安全に使用していただくための表示と意味 14
図記号の説明 14
警告 15
注意 15

安全上の要点 **16**

使用上の注意 **17**

法規と規格 **18**

バージョン **19**
バージョンの確認方法 19

関連マニュアル **22**

関連テクニカルガイド **25**

関連動画 **26**

用語説明 **27**

マニュアル改訂履歴 **28**

第 1 章　概要

1-1　ロボット統合システムの特長 **1-2**
1-1-1　NJ シリーズ ロボット統合 CPU ユニットの特長 1-2
1-1-2　EtherCAT 対応のオムロン製ロボットの特長 1-2
1-1-3　オートメーションソフトウェア Sysmac Studio の特長 1-2
1-2　本ガイドで構築するシステム構成 **1-3**
1-3　静的ピック & プレース装置のシステム構成 **1-4**
1-4　静的ピック & プレース装置の動作 **1-5**
1-4-1　動作仕様 1-5
1-4-2　装置の動作モード 1-5

1-5　動的ピック & プレース装置のシステム構成 1-6
1-6　動的ピック & プレース装置の動作 1-8
1-6-1　動作仕様 1-8
1-6-2　装置の動作モード 1-9
1-7　ハードウェア 1-10
1-8　ソフトウェア 1-11
1-8-1　Sysmac Studio 1-11
1-8-2　アプリケーションマネージャ 1-11
1-9　ユーザプログラム 1-13
1-9-1　ユーザプログラムの作成方法の種類 1-13
1-9-2　シーケンス制御プログラムと V+プログラムの使い分け 1-13
1-9-3　指令およびデータの流れ 1-14

第 2 章　事前に準備すること

2-1　Sysmac Studio をインストールする 2-2

第 3 章　静的ピック & プレース装置の実現例

3-1　静的ピック & プレース装置のプログラム仕様 3-2
3-1-1　シーケンス制御プログラムで動作を制御する場合 3-3
3-1-2　V+プログラムで動作を制御する場合 3-9
3-2　基本的な立ち上げ手順 3-15
3-3　プログラミングとシミュレーションの手順 3-16
3-3-1　プロジェクトファイルを新規作成する 3-16
3-3-2　EtherCAT ネットワーク構成を作成する 3-17
3-3-3　シーケンス制御プログラムを作成する 3-20
3-3-4　V+プログラムを作成する 3-26
3-3-5　3D 形状データを配置する 3-33
3-3-6　シミュレーションを開始する 3-59
3-3-7　ティーチングする 3-60
3-3-8　V+のプログラムと変数の電源 ON 時の自動読込を設定する 3-70
3-3-9　プログラムを実行して動作を確認する 3-74
3-4　設置、配線する 3-79
3-4-1　ロボット統合 CPU ユニットと EtherCAT デジタル I/O ターミナル間を配線する 3-79
3-4-2　EtherCAT デジタル I/O ターミナルの EtherCAT ノードアドレスを設定する 3-80
3-4-3　EtherCAT デジタル I/O ターミナルとロボット間を配線する 3-80
3-4-4　ロボットの EtherCAT ノードアドレスを設定する 3-81
3-4-5　ロボット統合 CPU ユニットとパソコン間を配線する 3-81
3-4-6　ロボットと T20 ペンダント、フロントパネル間を配線する 3-82
3-4-7　EtherCAT デジタル I/O ターミナルとソレノイドバルブ間を配線する 3-83
3-5　実機の立ち上げ手順 3-85
3-5-1　オンライン接続する 3-85
3-5-2　設定とプログラムを転送する 3-87
3-5-3　ティーチングする 3-89
3-5-4　プログラムを実行して動作を確認する 3-103

第 4 章　動的ピック & プレース装置の実現例

4-1　動的ピック & プレース装置のプログラム仕様 4-3
4-1-1　プログラム構成 4-3
4-2　基本的な立ち上げ手順 4-12
4-3　プログラミングとシミュレーションの手順 4-15
4-3-1　プロジェクトファイルを新規作成する 4-15

4-3-2 EtherCAT ネットワーク構成を作成する 4-16
4-3-3 V+プログラムを作成する 4-19
4-3-4 シーケンス制御プログラムを作成する 4-23
4-3-5 3D 形状データを配置する 4-30
4-3-6 Pack Manager サンプルを作成する 4-53
4-3-7 C#プログラムを作成する 4-82
4-3-8 V+のプログラムと変数の電源 ON 時の自動読込を設定する 4-84
4-3-9 プログラムを実行して動作を確認する 4-88

4-4 設置、配線する 4-94
4-4-1 ロボット統合 CPU ユニットと EtherCAT デジタル I/O ターミナル間を配線する 4-95
4-4-2 EtherCAT デジタル I/O ターミナルの EtherCAT ノードアドレスを設定する 4-95
4-4-3 EtherCAT デジタル I/O ターミナルとインバータの EtherCAT 通信ユニット間を配線する 4-96
4-4-4 インバータの EtherCAT 通信ユニットの EtherCAT ノードアドレスを設定する 4-96
4-4-5 インバータの EtherCAT 通信ユニットとロボット間を配線する 4-96
4-4-6 ロボットの EtherCAT ノードアドレスを設定する 4-97
4-4-7 ロボットと T20 ペンダント、フロントパネル間を配線する 4-97
4-4-8 ロボットとエンコーダ間を配線する 4-99
4-4-9 ロボット統合 CPU ユニットの EtherNet/IP ポートを配線する 4-100
4-4-10 IPC Application Controller とディスプレイ、カメラ間を配線する 4-101
4-4-11 カメラとロボット間を配線する 4-101
4-4-12 EtherCAT デジタル I/O ターミナルとソレノイドバルブ間を配線する 4-102

4-5 実機での動作確認 4-104
4-5-1 通信設定をする 4-104
4-5-2 オンライン接続する 4-110
4-5-3 設定とプログラムを転送する 4-112
4-5-4 カメラの設定をする 4-115
4-5-5 シーケンス制御プログラム、V+プログラムを実行する 4-126
4-5-6 ロボットのハイパワーを ON する 4-128
4-5-7 ベルトのキャリブレーションを行う 4-130
4-5-8 センサのキャリブレーションを行う 4-135
4-5-9 ロケータを設定する 4-146
4-5-10 アイドル位置、ピック位置、プレース位置をティーチングする 4-151
4-5-11 プログラムを実行して動作を確認する 4-154

付録

A-1 ピック & プレース装置の安全機能の設計例 A-2
A-1-1 ピック & プレース装置の構成 A-2
A-1-2 危険源 A-4
A-1-3 安全方策 A-4
A-1-4 安全機能 A-4
A-1-5 安全システム構成および使用機器 A-5
A-1-6 設置と配線 A-7
A-1-7 設定とプログラム A-13

A-2 3D 表示画面の操作方法 A-21
A-2-1 3D 表示画面をフローティング状態にする A-21
A-2-2 視点を回転する A-23
A-2-3 視点を平行移動する A-25
A-2-4 ズームインおよびズームアウト A-26

A-3 Sysmac Studio の設定項目と設定対象 A-28

A-4 トラブルシュート機能の操作方法 A-29

ご承諾事項

「当社商品」について特別の合意がない場合には、お客様のご購入先にかかわらず、本ご承諾事項記載の条件を適用いたします。

定義

本ご承諾事項中の用語の定義は次のとおりです。

- 「当社商品」:「当社」の FA システム機器、汎用制御機器、センシング機器、電子・機構部品
- 「カタログ等」:「当社商品」に関する、ベスト制御機器オムロン、電子・機構部品総合カタログ、その他のカタログ、仕様書、取扱説明書、マニュアル等であって電磁的方法で提供されるものも含みます。
- 「利用条件等」:「カタログ等」に記載の、「当社商品」の利用条件、定格、性能、動作環境、取り扱い方法、利用上の注意、禁止事項その他
- 「お客様用途」:「当社商品」のお客様におけるご利用方法であって、お客様が製造する部品、電子基板、機器、設備またはシステム等への「当社商品」の組み込み、または利用を含みます。
- 「適合性等」:「お客様用途」での「当社商品」の(a)適合性、(b)動作、(c)第三者の知的財産の非侵害、(d)法令の遵守および(e)各種規格の遵守

記載事項のご注意

「カタログ等」の記載内容については次の点をご理解ください。

- 定格値および性能値は、単独試験における各条件のもとで得られた値であり、各定格値および性能値の複合条件のもとで得られる値を保証するものではありません。
- 参考データはご参考として提供するもので、その範囲で常に正常に動作することを保証するものではありません。
- 利用事例はご参考ですので、「当社」は「適合性等」について保証いたしかねます。
- 「当社」は、改善や当社都合等により、「当社商品」の生産を中止し、または「当社商品」の仕様を変更することがあります。

ご利用にあたってのご注意

ご採用およびご利用に際しては次の点をご理解ください。

- 定格・性能ほか「利用条件等」を遵守しご利用ください。
- お客様ご自身にて「適合性等」をご確認いただき、「当社商品」のご利用の可否をご判断ください。「当社」は「適合性等」を一切保証いたしかねます。
- 「当社商品」がお客様のシステム全体の中で意図した用途に対して、適切に配電・設置されていることをお客様ご自身で、必ず事前に確認してください。
- 「当社商品」をご使用の際には、(i)定格および性能に対し余裕のある「当社商品」のご利用、冗長設計などの安全設計、(ii)「当社商品」が故障しても、「お客様用途」の危険を最小にする安全設計、(iii)利用者に危険を知らせるための、安全対策のシステム全体としての構築、(iv)「当社商品」および「お客様用途」の定期的な保守、の各事項を実施してください。
- 「当社」は DDoS 攻撃（分散型 DoS 攻撃）、コンピュータウイルスその他の技術的な有害プログラム、不正アクセスにより、「当社商品」、インストールされたソフトウェア、またはすべてのコンピュータ機器、コンピュータプログラム、ネットワーク、データベースが感染したとしても、そのことにより直接または間接的に生じた損失、損害その他の費用について一切責任を負わないものとします。

お客様ご自身にて、①アンチウイルス保護、②データ入出力、③紛失データの復元、④「当社商品」またはインストールされたソフトウェアに対するコンピュータウイルス感染防止、⑤「当社商品」に対する不正アクセス防止についての十分な措置を講じてください。

- 「当社商品」は、一般工業製品向けの汎用品として設計製造されています。従いまして、次に掲げる用途での使用は意図しておらず、お客様が「当社商品」をこれらの用途に使用される際には、「当社」は「当社商品」に対して一切保証をいたしません。ただし、次に掲げる用途であっても「当社」の意図した特別な商品用途の場合や特別の合意がある場合は除きます。
  a）高い安全性が必要とされる用途（例：原子力制御設備、燃焼設備、航空・宇宙設備、鉄道設備、昇降設備、娯楽設備、医用機器、安全装置、その他生命・身体に危険が及びうる用途）
  b）高い信頼性が必要な用途（例：ガス・水道・電気等の供給システム、24 時間連続運転システム、決済システムほか権利・財産を取扱う用途など）
  c）厳しい条件または環境での用途（例：屋外に設置する設備、化学的汚染を被る設備、電磁的妨害を被る設備、振動・衝撃を受ける設備など）
  d）「カタログ等」に記載のない条件や環境での用途
- 上記の(a)から(d)に記載されている他、「本カタログ等記載の商品」は自動車（二輪車含む。以下同じ）向けではありません。自動車に搭載する用途には利用しないで下さい。自動車搭載用商品については当社営業担当者にご相談ください。

保証条件

「当社商品」の保証条件は次のとおりです。

- 保証期間　ご購入後 1 年間といたします。
  （ただし「カタログ等」に別途記載がある場合を除きます。）
- 保証内容　故障した「当社商品」について、以下のいずれかを「当社」の任意の判断で実施します。
  a）当社保守サービス拠点における故障した「当社商品」の無償修理
  （ただし、電子・機構部品については、修理対応は行いません。）
  b）故障した「当社商品」と同数の代替品の無償提供
- 保証対象外　故障の原因が次のいずれかに該当する場合は、保証いたしません。
  a）「当社商品」本来の使い方以外のご利用
  b）「利用条件等」から外れたご利用
  c）本ご承諾事項「ご利用にあたってのご注意」に反するご利用
  d）「当社」以外による改造、修理による場合
  e）「当社」以外の者によるソフトウェアプログラムによる場合
  f）「当社」からの出荷時の科学・技術の水準では予見できなかった原因
  g）上記のほか「当社」または「当社商品」以外の原因（天災等の不可抗力を含む）

責任の制限

本ご承諾事項に記載の保証が、「当社商品」に関する保証のすべてです。
「当社商品」に関連して生じた損害について、「当社」および「当社商品」の販売店は責任を負いません。

輸出管理

「当社商品」または技術資料を、輸出または非居住者に提供する場合は、安全保障貿易管理に関する日本および関係各国の法令・規制を遵守ください。お客様が法令・規則に違反する場合には、「当社商品」または技術資料をご提供できない場合があります。

産業用ロボット

労働安全衛生規則第 36 条第 31 号および 32 号において、産業用ロボットの教示等を行う作業や検査・修理・調整等を行う作業は、労働安全衛生法で定める「危険又は有害な業務」に該当します。
労働安全衛生法第 59 条により、事業者は労働者に対し「安全又は衛生のための特別の教育」を行う義務があります。
お客様の責任において、リスクアセスメントの結果から適切な安全対策を実施してください。
ロボットセーフティガイドおよび弊社ロボットシステム商品関連資料の記載事項を全て遵守しても、産業用ロボットが引き起こす人身事故または機器の損傷を回避する保証はできません。

安全上の注意

安全に使用していただくための表示と意味

本マニュアルでは、NJ シリーズ ロボット統合 CPU ユニットを安全に使用していただくために、注意事項を次のような表示と図記号で示しています。
ここで示した注意事項は、安全に関する重大な内容を記載しています。必ず守ってください。
表示と意味は次のとおりです。

⚠ 警 告	正しい取り扱いをしなければ、この危険のために、軽傷・中程度の傷害を負ったり万一の場合には重傷や死亡に至ったりするおそれがあります。また、同様に重大な物的損害をもたらすおそれがあります。
⚠ 注 意	正しい取り扱いをしなければ、この危険のために、時に軽傷・中程度の傷害を負ったり、あるいは物的損害を受けたりするおそれがあります。

図記号の説明

	○に斜線の記号は、禁止を意味しています。 具体的な内容は、○の中と文章で示します。 左図の場合は、「分解禁止」を表します。
	△ 記号は、注意（警告を含む）を意味しています。 具体的な内容は、△ の中と文章で示します。 左図の場合は、「感電注意」を表します。
	△ 記号は、注意（警告を含む）を意味しています。 具体的な内容は、△ の中と文章で示します。 左図の場合は、「一般的な注意」を表します。
	● 記号は、強制を意味しています。 具体的な内容は、● の中と文章で示します。 左図の場合は、「一般的な強制事項」を表します。

警告

⚠ 警告

警告の内容については、以下のマニュアルを参照してください。
- 『NJ シリーズ CPU ユニット ユーザーズマニュアル ハードウェア編 (SBCA-466)』
- 『NJ シリーズ ロボット統合 CPU ユニット ユーザーズマニュアル (SBCE-436)』

注意

⚠ 注意

注意の内容については、以下のマニュアルを参照してください。
- 『NJ シリーズ CPU ユニット ユーザーズマニュアル ハードウェア編 (SBCA-466)』
- 『NJ シリーズ ロボット統合 CPU ユニット ユーザーズマニュアル (SBCE-436)』

安全上の要点

安全上の要点の内容については、以下のマニュアルを参照してください。
- 『NJ シリーズ CPU ユニット ユーザーズマニュアル ハードウェア編 (SBCA-466)』
- 『NJ シリーズ ロボット統合 CPU ユニット ユーザーズマニュアル (SBCE-436)』

使用上の注意

使用上の注意の内容については、以下のマニュアルを参照してください。
- 『NJ シリーズ CPU ユニット ユーザーズマニュアル ハードウェア編 (SBCA-466)』
- 『NJ シリーズ ロボット統合 CPU ユニット ユーザーズマニュアル (SBCE-436)』

法規と規格

法規と規格の内容については、以下のマニュアルを参照してください。
- 『NJ シリーズ CPU ユニット ユーザーズマニュアル ハードウェア編 (SBCA-466)』



参考

ロボット統合 CPU ユニットは、ISO 10218-1 で定められるロボット制御装置ではありません。そのため、ロボット統合 CPU ユニットとして、ロボット固有の法規と規格への適合はありません。
オムロン製ロボット本体に関しては、オムロン製ロボットのマニュアルを参照してください。

バージョン

NJ シリーズの各ユニットおよび各 EtherCAT スレーブは、ハードウェアやソフトウェアが、ハードウェアリビジョンやユニットバージョンという改変番号でバージョン管理されています。ハードウェアまたはソフトウェアに仕様変更があるごとに、ハードウェアリビジョンまたはユニットバージョンが更新されます。したがって、同じ形式のユニットや EtherCAT スレーブでも、ハードウェアリビジョンまたはユニットバージョンが異なると搭載機能や性能が異なります。

バージョンの確認方法

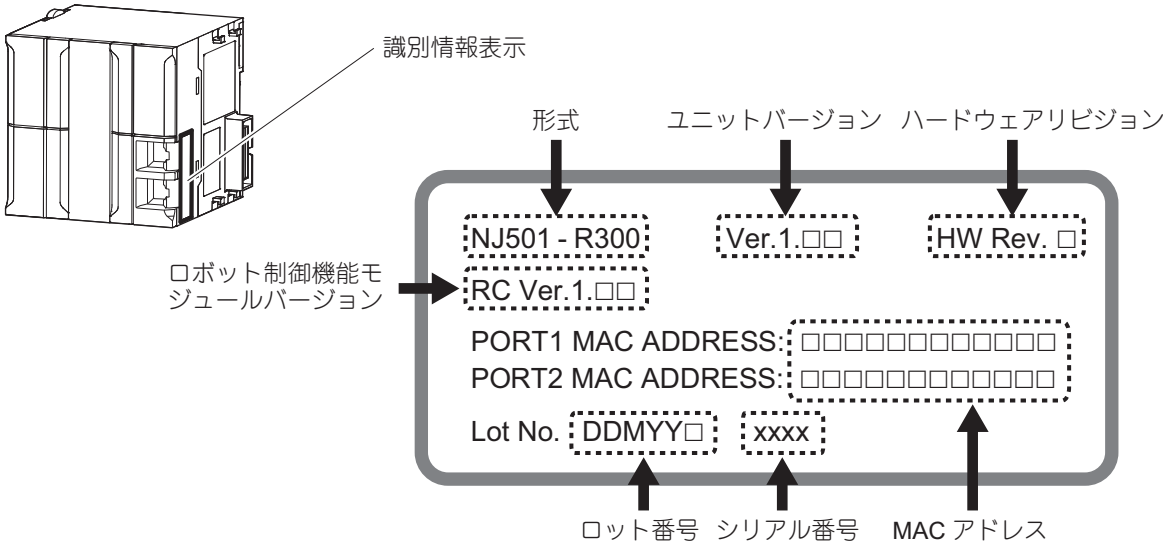
バージョンは、識別情報表示または Sysmac Studio で確認できます。

識別情報表示による確認

製品側面の識別情報表示で、バージョンが確認できます。

- **形 NJ501-R□00 の場合**

  NJ シリーズ CPU ユニット形 NJ501-R300 の識別情報表示は、下図のようになります。



（注）ハードウェアリビジョンが「なし」のユニットでは、ハードウェアリビジョンの表示はありません。

Sysmac Studio による確認方法

Sysmac Studio で、バージョンが確認できます。

● NJ シリーズ CPU ユニットでのバージョン確認方法

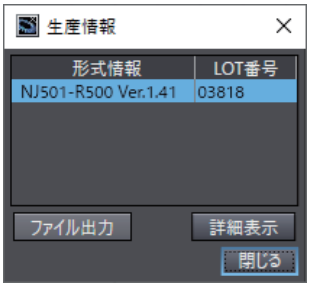
ユニットのバージョンは、オンライン状態の［生産情報］で確認できます。ただし、バージョンが確認できるユニットは、CPU ユニット、CJ シリーズの高機能 I/O ユニット、および CJ シリーズの CPU 高機能ユニットだけです。CJ ユニットの基本 I/O ユニットのバージョンは、Sysmac Studio では確認できません。

1 **マルチビューエクスプローラで［構成・設定］–［CPU・増設ラック］–［CPU ラック］をダブルクリックします。または、［構成・設定］–［CPU・増設ラック］–［CPU ラック］を右クリックして、［編集］を選択します。**
ユニットエディタが表示されます。

2 **ユニットエディタの空欄上で右クリックして［生産情報表示］を選択します。**
［生産情報］ダイアログボックスが表示されます。

● 生産情報の表示内容の切り替え

1 **［生産情報］ダイアログボックスの右下の、［簡易表示］または［詳細表示］を選択します。**
［生産情報］の簡易表示と詳細表示が、切り替わります。



簡易表示



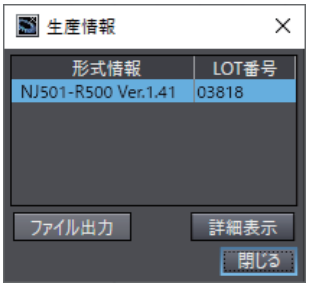
詳細表示

表示内容は、簡易表示と詳細表示とで異なります。詳細表示では、ユニットバージョン、ハードウェアリビジョン、および各種のバージョンが表示されます。簡易表示では、ユニットバージョンだけが表示されます。

（注）ハードウェアリビジョンは、ハードウェアバージョンの右端に「/」で区切られて表示されます。ハードウェアリビジョンが「なし」のユニットでは、ハードウェアリビジョンの表示はありません。

● 生産情報の表示内容の切り替え

1 **［生産情報］ダイアログボックスの右下の、［簡易表示］または［詳細表示］を選択します。**
［生産情報］の簡易表示と詳細表示が、切り替わります。


生産情報
形式情報
LOT番号
NJ501-R500 Ver.1.41
03818
ファイル出力
詳細表示
閉じる

簡易表示


生産情報
形式情報
LOT番号
NJ501-R500 Ver.1.41
03818
Robot Ver.1.04
RC Ver.1.00
ハードウェアバージョン : B C C C B /B
ソフトウェアバージョン
SYSTEM : 1.41.01 97851
Runtime : 1259
BOOT : 20180209
IOPFP : C-6-0
IOPFW : 1.49
Robot : 1.04.00
RC : 1.00.00
ファイル出力
簡易表示
閉じる

詳細表示

関連マニュアル

本マニュアルに関連するマニュアルは、下表のとおりです。あわせてご覧ください。

マニュアル名称	Man.No.	形式	用途	内容
NJ シリーズ CPU ユニット ユーザーズマニュアル ハードウェア編	SBCA-466	形 NJ501-□□□□ 形 NJ301-□□□□ 形 NJ101-□□□□	NJ シリーズ CPU ユニットの概要／設計／取付／保守などの基本的な仕様について知りたいとき。 おもにハードウェアに関する情報。	NJ シリーズのシステム全体概要、および CPU ユニットに関して、以下の内容を説明します。 • 特長やシステム構成 • 概要 • 各部の名称と機能 • 一般仕様 • 設置と配線 • 保守点検
NJ/NX シリーズ CPU ユニット ユーザーズマニュアル ソフトウェア編	SBCA-467	形 NX701-□□□□ 形 NX502-□□□□ 形 NX102-□□□□ 形 NX1P2-□□□□ 形 NJ501-□□□□ 形 NJ301-□□□□ 形 NJ101-□□□□	NJ/NX シリーズ CPU ユニットのプログラミング／システムの立ち上げについて知りたいとき。 おもにソフトウェアに関する情報。	NJ/NX シリーズ CPU ユニットに関して、以下の内容を説明します。 • CPU ユニットの動作 • CPU ユニットの機能 • 初期設定 • IEC 61131-3 ベースの言語仕様とプログラミング
NJ/NX シリーズ コマンドリファレンス マニュアル 基本編	SBCA-468	形 NX701-□□□□ 形 NX502-□□□□ 形 NX102-□□□□ 形 NX1P2-□□□□ 形 NJ501-□□□□ 形 NJ301-□□□□ 形 NJ101-□□□□	NJ/NX シリーズの基本命令仕様の詳細について知りたいとき。	各命令（IEC 61131-3 仕様）の詳細を説明します。
NJ/NX シリーズ CPU ユニット ユーザーズマニュアル モーション制御編	SBCE-433	形 NX701-□□□□ 形 NX502-□□□□ 形 NX102-□□□□ 形 NX1P2-□□□□ 形 NJ501-□□□□ 形 NJ301-□□□□ 形 NJ101-□□□□	モーション制御の設定やプログラミングの考え方について知りたいとき。	モーション制御のための CPU ユニットの設定や動作、プログラミングの考え方について説明します。
NJ/NX シリーズ コマンドリファレンス マニュアル モーション編	SBCE-434	形 NX701-□□□□ 形 NX502-□□□□ 形 NX102-□□□□ 形 NX1P2-□□□□ 形 NJ501-□□□□ 形 NJ301-□□□□ 形 NJ101-□□□□	モーション命令仕様の詳細について知りたいとき。	各モーション命令の詳細を説明します。
NJ/NX シリーズ CPU ユニット 内蔵 EtherCAT®ポート ユーザーズマニュアル	SBCD-376	形 NX701-□□□□ 形 NX502-□□□□ 形 NX102-□□□□ 形 NX1P2-□□□□ 形 NJ501-□□□□ 形 NJ301-□□□□ 形 NJ101-□□□□	NJ/NX シリーズ CPU ユニットの内蔵 EtherCAT ポートを使用するとき。	内蔵 EtherCAT ポートに関して説明します。 概要、構成、機能、セットアップについて記述しています。
NJ/NX シリーズ CPU ユニット 内蔵 EtherNet/IP™ポート ユーザーズマニュアル	SBCD-377	形 NX701-□□□□ 形 NX502-□□□□ 形 NX102-□□□□ 形 NX1P2-□□□□ 形 NJ501-□□□□ 形 NJ301-□□□□ 形 NJ101-□□□□	NJ/NX シリーズ CPU ユニットの内蔵 EtherNet/IP ポートを使用するとき。	内蔵 EtherNet/IP ポートに関して説明します。 基本設定、タグデータリンク、その他の機能について記述しています。
NJ シリーズ ロボット統合 CPU ユニット ユーザーズマニュアル	SBCE-436	形 NJ501-R□□□	NJ シリーズ ロボット統合 CPU ユニットを使用するとき。	オムロン製ロボットを制御するための CPU ユニットの設定や動作、プログラミングの考え方について説明します。

マニュアル名称	Man.No.	形式	用途	内容
Sysmac Studio ロボット統合システム構築機能 オペレーションマニュアル ロボット統合 CPU ユニット編	SBCA-464	形 SYSMAC-SE2□□□ 形 SYSMAC-SE200D-64	NJ シリーズ ロボット統合 CPU ユニットを使用してロボット統合システムを構築するとき。	Sysmac Studio で NJ シリーズ ロボット統合 CPU ユニットを使用するための機能の概要、操作方法について説明します。
Sysmac Studio ロボット統合システム構築機能 オペレーションマニュアル IPC Application Controller 編	SBCA-483	形 SYSMAC-SE2□□□ 形 SYSMAC-SE200D-64	IPC Application Controller を使用してロボット統合システムを構築するとき。	Sysmac Studio で IPC Application Controller を使用するための機能の概要、操作方法について説明します。
Sysmac Studio 3D シミュレーション機能 オペレーションマニュアル	SBCA-482	形 SYSMAC-SE2□□□ 形 SYSMAC-SA4□□□-64	Sysmac Studio の 3D シミュレーション機能の概要、使用方法について知りたいとき。	Sysmac Studio の 3D シミュレーション機能の概要、実行手順、操作方法について説明します。
eV+3 ユーザーズマニュアル	SBCE-486	形 NJ501-R□□□	V+プログラムでオムロン製ロボットを動作させるとき。	オムロン製ロボットを制御するための専用言語である V+言語について説明します。
eV+3 キーワードリファレンスマニュアル	SBCE-487	形 NJ501-R□□□	V+プログラムでオムロン製ロボットを動作させるとき。	V+言語で使用する V+キーワードについて説明します。
スカラロボット eCobra 600/800 EtherCAT タイプ ユーザーズマニュアル	SBCE-489	形 RL4-116□□□□ 形 RL4-216□□□□	eCobra を使用するとき。	eCobra について説明します。
スカラロボット i4L EtherCAT タイプ ユーザーズマニュアル	SBCE-496	形 RS4-2063□□□ 形 RS4-2064□□□ 形 RS4-2065□□□	i4L を使用するとき。	i4L について説明します。
スカラロボット i4H EtherCAT タイプ ユーザーズマニュアル	SBCE-497	形 RS4-2066□□□ 形 RS4-2067□□□ 形 RS4-2068□□□	i4H を使用するとき。	i4H について説明します。
垂直多関節ロボット Viper 650/850 EtherCAT タイプ ユーザーズマニュアル	SBCE-490	形 RL6-206□□□□	Viper を使用するとき。	Viper について説明します。
パラレルロボット iX3 565 EtherCAT タイプ ユーザーズマニュアル	SBCE-491	形 RX3-206□□□□	iX3 を使用するとき。	iX3 について説明します。
パラレルロボット iX4 650H/650HS/800H/800HS EtherCAT タイプ ユーザーズマニュアル	SBCE-492	形 RX4-216□□□□	iX4 を使用するとき。	iX4 について説明します。
ロボット セーフティガイド	SBCE-381	形 RL4-□□□□□□□ 形 RS4-□□□□□□□ 形 RL6-□□□□□□□ 形 RX3-□□□□□□□ 形 RX4-□□□□□□□	オムロン製ロボットの安全な使用方法について知りたいとき。	オムロン製ロボットの安全な使用方法について説明します。
ティーチングペンダント T20 ユーザーズマニュアル	SBCE-392	形 10046-010	オムロン製ロボットをティーチングペンダントで動作させるとき。	T20 ペンダントのセットアップ、操作、およびユーザ保守について説明します。
IPC アプリケーションコントローラ ユーザーズマニュアル	SBCE-454	形 AC1-152000	IPC Application Controller を使用するとき。	IPC Application Controller について説明します。
NJ/NX シリーズ データベース接続 CPU ユニット ユーザーズマニュアル	SBCA-411	形 NX701-□□20 形 NX502-□□□□ 形 NX102-□□20 形 NJ501-□□20 形 NJ101-□□20	NJ/NX シリーズでデータベース接続サービス機能を使用するとき。	データベース接続サービス機能について説明します。

マニュアル名称	Man.No.	形式	用途	内容
NJ シリーズ NJ Robotics CPU ユニット ユーザーズマニュアル	SBCA-421	形 NJ501-4□□□ 形 NJ501-R□□□	NJ シリーズで Robotics 機能を使用するとき。	NJ Robotics 機能について説明します。
NJ/NX シリーズ トラブルシューティング マニュアル	SBCA-469	形 NX701-□□□□ 形 NX502-□□□□ 形 NX102-□□□□ 形 NX1P2-□□□□ 形 NJ501-□□□□ 形 NJ301-□□□□ 形 NJ101-□□□□	NJ/NX シリーズで検出する異常の詳細について知りたいとき。	NJ/NX シリーズ システムにて検出する異常管理の考え方と各異常項目について説明します。
Sysmac Studio Version 1 オペレーションマニュアル	SBCA-470	形 SYSMAC -SE2□□□	Sysmac Studio の操作方法、機能について知りたいとき。	Sysmac Studio の操作方法について説明します。
NX シリーズ 位置インタフェースユニット ユーザーズマニュアル	SBCE-374	形 NX-EC0□□□ 形 NX-ECS□□□ 形 NX-PG0□□□	NX シリーズ位置インタフェースユニットの使用方法について知りたいとき。	NX シリーズインクリメンタルエンコーダ入力ユニット、SSI 入力ユニット、パルス出力ユニットのハードウェアや設定方法、機能について説明します。
AC サーボモータ／ドライバ 1S シリーズ EtherCAT®通信内蔵タイプ ユーザーズマニュアル	SBCE-377	形 R88M-1□ 形 R88D-1SN□-ECT	EtherCAT 通信内蔵タイプのサーボモータ／ドライバの使用方法について知りたいとき。	EtherCAT 通信内蔵タイプのサーボモータ／ドライバのハードウェアや設定方法、機能について説明します。
	SBCE-438	形 R88M-1AL□/ -1AM□ 形 R88D-1SAN□-ECT		
AC サーボモータ／ドライバ G5 シリーズ EtherCAT®通信内蔵タイプ ユーザーズマニュアル	SBCE-365	形 R88M-K□ 形 R88D-KN□-ECT	EtherCAT 通信内蔵タイプのサーボモータ／ドライバの使用方法について知りたいとき。	EtherCAT 通信内蔵タイプのサーボモータ／ドライバのハードウェアや設定方法、機能について説明します。 G5 シリーズには、リニアモータタイプや位置制御限定モデルがあります。
	SBCE-366	形 R88L-EC-□ 形 R88D-KN□-ECT-L		

関連テクニカルガイド

NJ シリーズ CPU ユニットの基礎を理解するためのテクニカルガイドは、下表のとおりです。あわせてご覧ください。

マニュアル名称	Man.No.	用途	内容
マシンオートメーションコントローラ スタートアップガイド（CPU ユニット編）	SBCA-404	NJ シリーズ CPU ユニットと Sysmac Studio の使い方の基礎を知りたいとき。	NJ シリーズ CPU ユニットを初めて使用する場合の立ち上げ手順と Sysmac Studio の基本的な操作方法を簡単なシーケンス制御の例で説明しています。
マシンオートメーションコントローラ スタートアップガイド（ モーション制御編）	SBCE-372	NJ シリーズのモーション制御機能の基礎を知りたいとき。	NJ シリーズのモーション制御機能を初めて使用する場合の立ち上げ手順と Sysmac Studio の操作方法を簡単な 1 軸位置決めや 2 軸直線補間の例で説明しています。
マシンオートメーションコントローラ 技術導入ガイド IEC プログラミング編	SBCA-405	IEC 61131-3 の概要とメリットを知りたいとき。	IEC 61131-3 プログラミングの背景と有用性、および IEC 61131-3 に沿ったプログラミングの効用について説明しています。

関連動画

NJ シリーズ ロボット統合システムでの Sysmac Studio による設定およびシミュレーションについては以下の動画もあわせてご覧ください。

- 動画でみる ロボット統合システムスタートアップガイド
  https://www.fa.omron.co.jp/product/robotics/lineup/integratedcontroller/featured-products/robotsystem-startupguide/

また、本ガイドでは関連動画がある箇所に動画の URL が記載されています。読み進めている内容に関連する動画をすぐに確認することができます。

用語説明

本マニュアルで使用する用語について説明します。

用語	説明
CP 動作	オムロン製ロボットの動きを止めることなく、連続する動作を滑らかに移行する動作のことです。
IEC 61131-3 言語	シーケンス制御を記述するためのプログラム言語です。
NJ Robotics 機能で制御できるロボット	NJ シリーズ CPU ユニットのモーション制御機能モジュールでロボット用演算処理を行うことにより、制御が可能なロボットを指します。 EtherCAT 通信内蔵タイプの 1S シリーズ、または G5 シリーズサーボモータ／ドライバとお客様で用意されたロボットアームで構成されます。
TCP	各オムロン製ロボットで定義されている先端(Tool Center Point)を表します。 この TCP を基準として、目標位置や軌跡を指定することができます。
TIO	オムロン製の SCARA ロボット(i4H, i4L)のアーム先端ツールを使用するためのデジタル入出力信号を指します。
V+	オムロン製ロボットを制御するオペレーティングシステムです。V +プログラムを実行します。
V+キーワード	V+プログラム、およびモニタコマンド中で使われる命令の総称です。
V+言語	オムロン製ロボット制御専用のプログラミング言語です。
V+タスク	V+プログラムを割り当てて実行することができるタスクです。
V+バージョン	ロボット統合 CPU ユニットやオムロン製ロボットの V+のバージョン。
V+プログラム	V+言語で記述された制御プログラムです。
オムロン製ロボット	ロボット統合 CPU ユニットから制御可能なオムロン製ロボットを指します。 ロボットアンプと、ロボットアンプに接続されたロボットアームから構成されます。
共有変数	シーケンス制御プログラムと V+プログラムとの間で共有することができる変数です。
シーケンス制御プログラム	IEC 61131-3 言語で記述された制御プログラムであり、モーション制御も含みます。
設定 V+バージョン	V+バージョン設定機能によって設定される V+バージョンです。 ロボット統合 CPU ユニットとオムロン製ロボットのそれぞれに設定されます。
ハードウェアサーボ	ロボットアンプに内蔵されているサーボシステムのことです。
ユーザプログラム	ラダー図言語、ST 言語、V+言語で作られたプログラムの集合体の総称です。
リモートエンコーダ	モーション制御軸をロボット制御用の外部エンコーダとして設定したエンコーダを指します。
レシピ	お客様のシステムにおける品種データのセットのことです。
レシピ変更	製品データや製造プロセスに関する情報（レシピ）を変更することを指します。 ロボット統合 CPU ユニットのレシピ対象は、変数の現在値や画像センサなどのプロパティです。
ローカルエンコーダ	オムロン製ロボットのエンコーダ入力ポートに接続したエンコーダを指します。
ロボット制御機能モジュール	ロボット統合 CPU ユニットに搭載されるロボット制御を行うソフトウェアです。
ロボット制御命令	シーケンス制御プログラム上に記述するオムロン製ロボットを制御するための FB 型命令です。 オムロン製ロボットを直接制御する命令、および V+タスクに割り当てられた V+プログラムを実行したり中断したりする命令があります。
ロボット統合 CPU ユニット	NJ シリーズ CPU ユニットにオムロン製ロボットの制御機能を搭載した CPU ユニットです。

マニュアル改訂履歴

マニュアル改訂記号は、表表紙・裏表紙に記載されている Man.No.の後尾に付記されます。



改訂記号	改訂年月	改訂理由・改訂ページ
A	2020 年 8 月	初版
B	2020 年 12 月	• 動的ピック&プレース装置についての説明を追加 • 誤記修正
C	2021 年 10 月	• 関連動画へのリンク情報を追加 • 誤記修正
D	2022 年 3 月	• 関連動画へのリンク情報を追加 • 誤記修正
E	2024 年 7 月	Sysmac Studio Ver.1.54 へのバージョンアップに伴う改訂

1

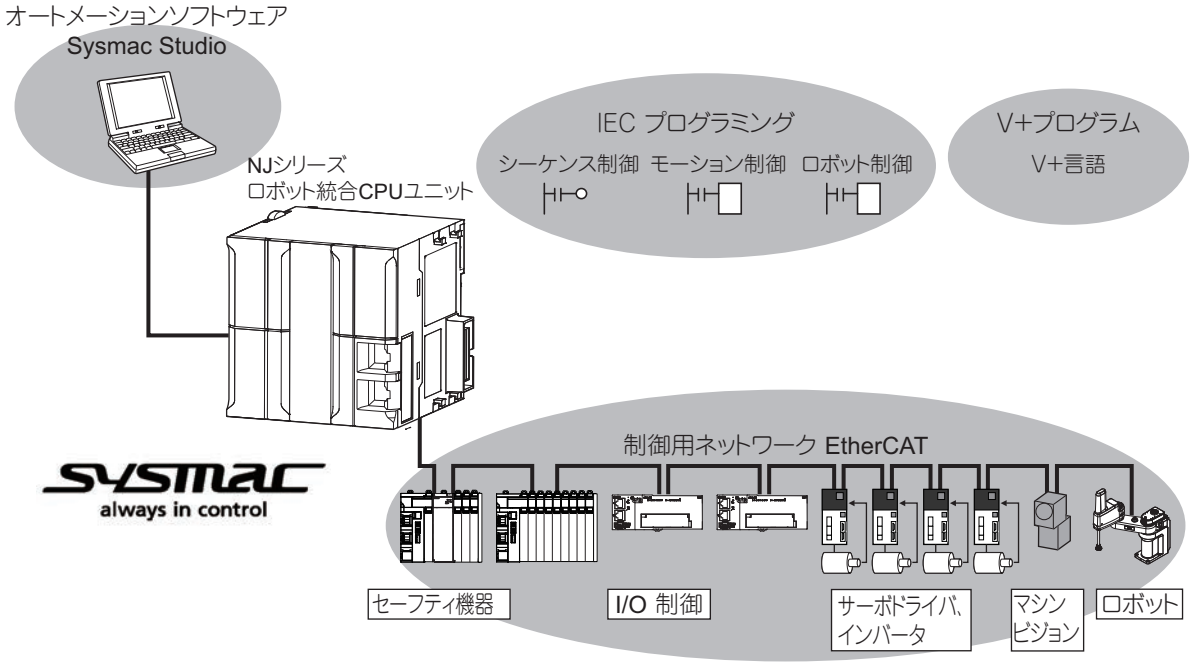
概要

この章では、ロボット統合システムの概要について説明します。

1-1　ロボット統合システムの特長 1-2
1-1-1　NJ シリーズ ロボット統合 CPU ユニットの特長 1-2
1-1-2　EtherCAT 対応のオムロン製ロボットの特長 1-2
1-1-3　オートメーションソフトウェア Sysmac Studio の特長 1-2
1-2　本ガイドで構築するシステム構成 1-3
1-3　静的ピック & プレース装置のシステム構成 1-4
1-4　静的ピック & プレース装置の動作 1-5
1-4-1　動作仕様 1-5
1-4-2　装置の動作モード 1-5
1-5　動的ピック & プレース装置のシステム構成 1-6
1-6　動的ピック & プレース装置の動作 1-8
1-6-1　動作仕様 1-8
1-6-2　装置の動作モード 1-9
1-7　ハードウェア 1-10
1-8　ソフトウェア 1-11
1-8-1　Sysmac Studio 1-11
1-8-2　アプリケーションマネージャ 1-11
1-9　ユーザプログラム 1-13
1-9-1　ユーザプログラムの作成方法の種類 1-13
1-9-2　シーケンス制御プログラムと V+プログラムの使い分け 1-13
1-9-3　指令およびデータの流れ 1-14

1-1 ロボット統合システムの特長

ロボット統合システムは、シーケンス制御とロボット制御をひとつのコントローラ、ネットワーク、およびソフトウェアに統合したシステムです。



ロボット統合システムは、NJ シリーズ ロボット統合 CPU ユニット、EtherCAT 対応のオムロン製ロボット、オートメーションソフトウェア Sysmac Studio で構成されます。

1-1-1 NJ シリーズ ロボット統合 CPU ユニットの特長

シーケンス制御とロボット制御をひとつの CPU ユニットで実行できます。これにより、シーケンス制御とロボット制御間のインタフェース設計が容易になり、設計や立ち上げ時の調整時間を短縮することができます。

1-1-2 EtherCAT 対応のオムロン製ロボットの特長

ロボットと周辺機器を同じ EtherCAT ネットワークに接続することができます。EtherCAT の同期機能により、ロボットと周辺機器の動作が同じ時間軸で扱えるようになり、非同期のための待ち時間やタイミングのばらつきが低減されるため、タクトタイムの短縮や、設備稼働中の異常要因の特定、周辺機器の動作を予測した制御ができるようになります。

1-1-3 オートメーションソフトウェア Sysmac Studio の特長

シーケンス制御とロボット制御をひとつのソフトウェアで開発できます。シーケンス制御とロボット制御が連携したシステムの動作を 3D モデルでシミュレーションできるため、机上での事前検証が容易になります。

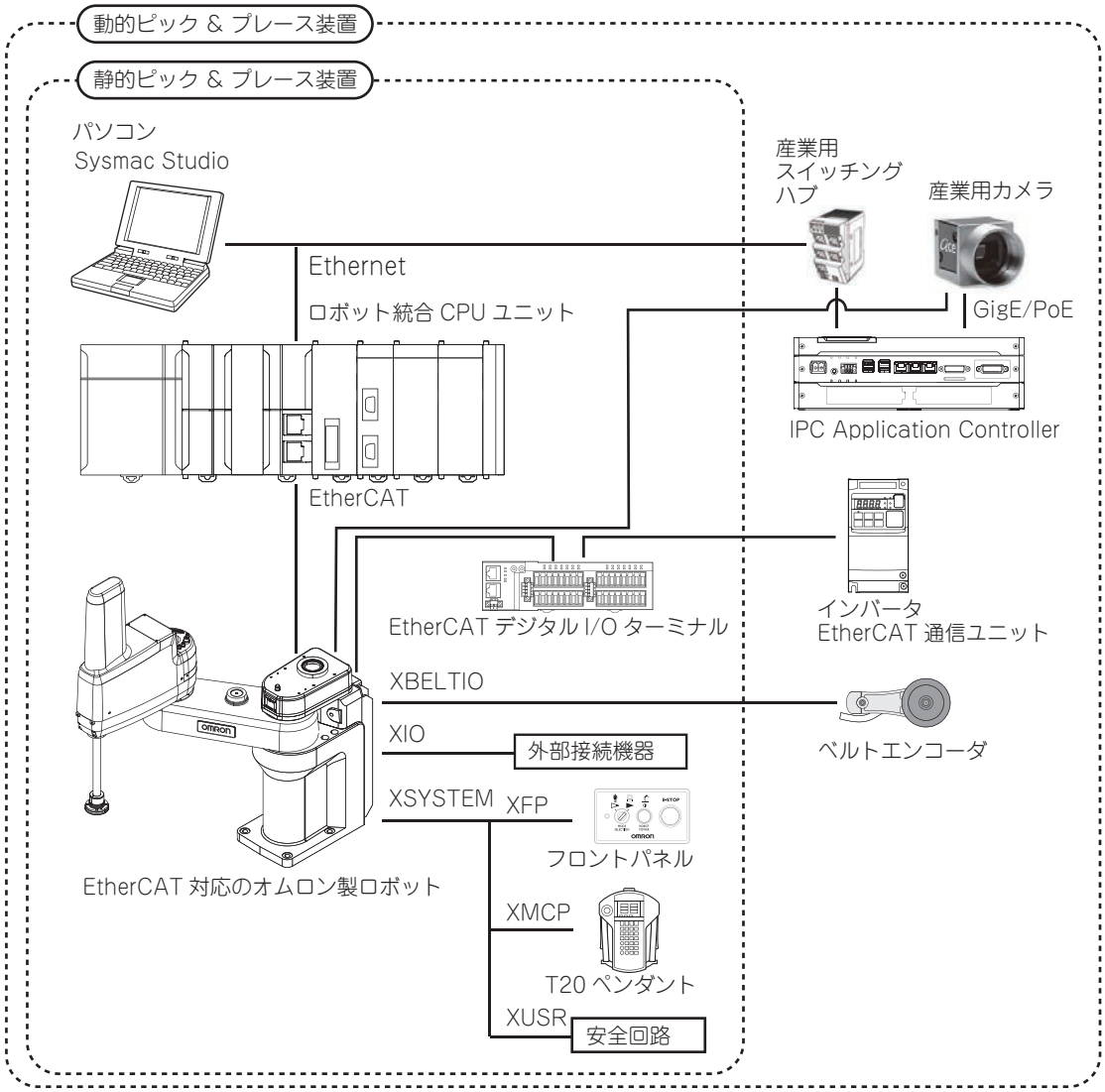
1-2 本ガイドで構築するシステム構成

『NJ シリーズ ロボット統合システム スタートアップガイド』（以降、本ガイドと省略する場合があります）では、静的ピック & プレース装置と動的ピック & プレース装置の実現例について説明しています。

本ガイドでは、決められた位置で静止しているワークを、決められた位置へ移動させる装置を静的ピック & プレース装置と呼んでいます。

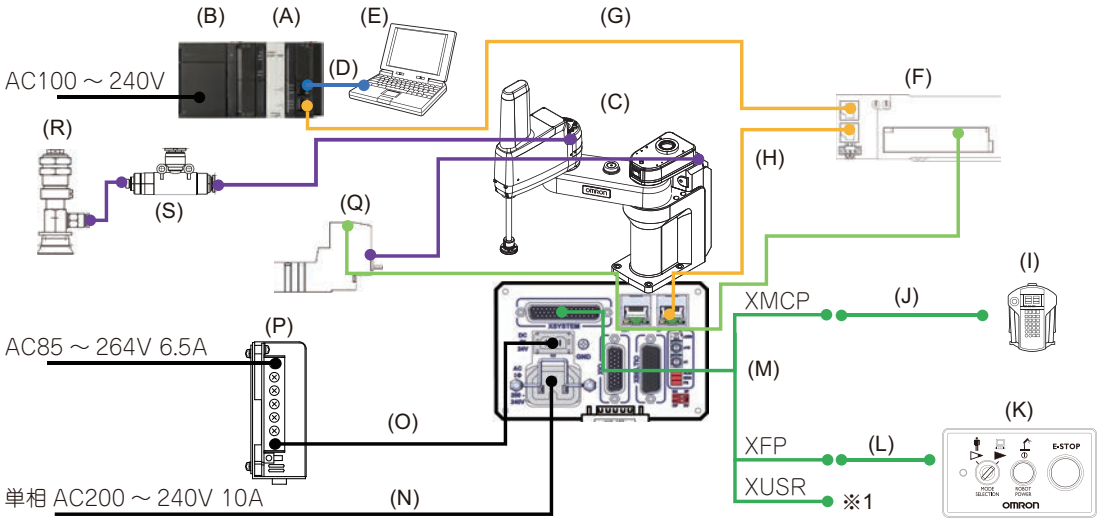
また、コンベアで搬送されているワークをピックし、指定した位置にプレースする装置を動的ピック & プレース装置と呼んでいます。カメラとエンコーダで検出した位置をもとに、ピック位置が自動的に計算されます。
複数のオムロン製ロボットやレシピを管理するための IPC Application Controller を Ethernet で接続します。IPC Application Controller、産業用カメラおよびベルトエンコーダは、「1-8-2 アプリケーションマネージャ (P.1-11)」の使用する機能に応じて使用の有無を決定してください。

それぞれの装置で使用する主な機器を以下に示します。



1-3 静的ピック & プレース装置のシステム構成

本ガイドで説明する静的ピック & プレース装置のシステム構成について記載します。
本ガイドでは、決められた位置で静止しているワークを、決められた位置へ移動させる装置を静的ピック & プレース装置と呼んでいます。
図中の“XUSR”は、安全回路への配線です。「A-1 ピック & プレース装置の安全機能の設計例 (P.A-2)」を参照し、実際の装置に必要な安全回路を設計してください。



次に、システム構成で使用する部品を以下に示します。

記号	名称	形式	バージョン
A	ロボット統合 CPU ユニット	形 NJ501-R300	Ver.1.41
B	電源ユニット	形 NJ-PA3001	---
C	SCARA ロボット eCobra 600 Pro	形 RL4-2166000	3.0 0-0 Edit A1
D	Ethernet ケーブル*1	-	---
E	パソコン(Sysmac Studio)	-	---
F	EtherCAT デジタル I/O ターミナル	形 GX-MD3218	---
G	Ethernet ケーブル*1	-	---
H	Ethernet ケーブル*1	-	---
I	T20 ペンダント	形 10046-010	3.0.0.1
J	T20 アダプタケーブル		---
K	フロントパネル	形 90356-10358	---
L	フロントパネル ケーブル		---
M	XSYSTEM ケーブルアセンブリ	形 RL4-2166000 に同梱	---
N	AC 電源ケーブル	形 04118-000	---
O	DC 電源ケーブル	形 04120-000	---
P	DC24V 電源 150W	形 S8FS-G15024CD	---
Q	ソレノイドバルブ	ワークに応じて選定	---
R	吸着パッド		---
S	真空エジェクタ		---

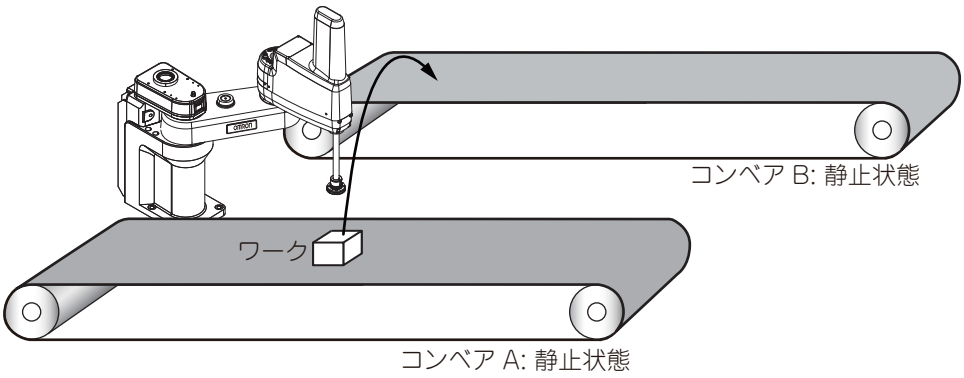
*1. Ethernet カテゴリ 5（100BASE-TX）以上のツイストペアケーブル（アルミテープと編組の二重遮へい）を使用してください。

1-4 静的ピック & プレース装置の動作

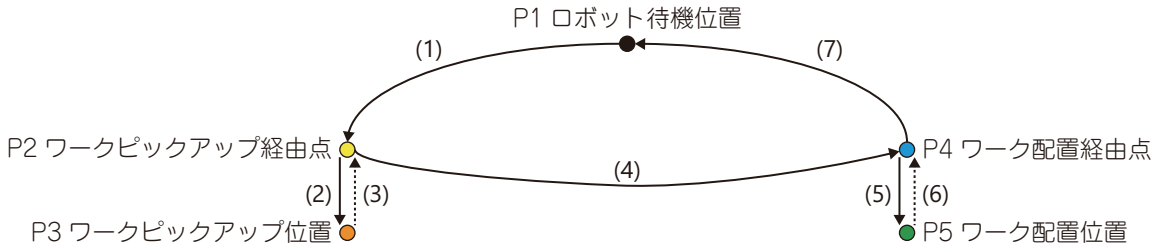
ここでは、静的ピック & プレース装置の動作仕様と操作手順について説明します。

1-4-1 動作仕様

本装置は、コンベア A 上で静止しているワークを取り上げ、コンベア B の決められた位置まで搬送します。



ロボットの具体的な動作を以下に示します。



番号	動作説明
(1)	ロボット待機位置(P1)からワークピックアップ経由点(P2)まで、関節補間動作で移動
(2)	ワークピックアップ経由点(P2)からワークピックアップ位置(P3)まで、直線補間動作で移動
(3)	ワークピックアップ位置(P3)からワークピックアップ経由点(P2)まで、直線補間動作で移動
(4)	ワークピックアップ経由点(P2)からワーク配置経由点(P4)まで、関節補間動作で移動
(5)	ワーク配置経由点(P4)からワーク配置位置(P5)まで、直線補間動作で移動
(6)	ワーク配置位置(P5)からワーク配置経由点(P4)まで、直線補間動作で移動
(7)	ワーク配置経由点(P4)からロボット待機位置(P1)まで、関節補間動作で移動

参考

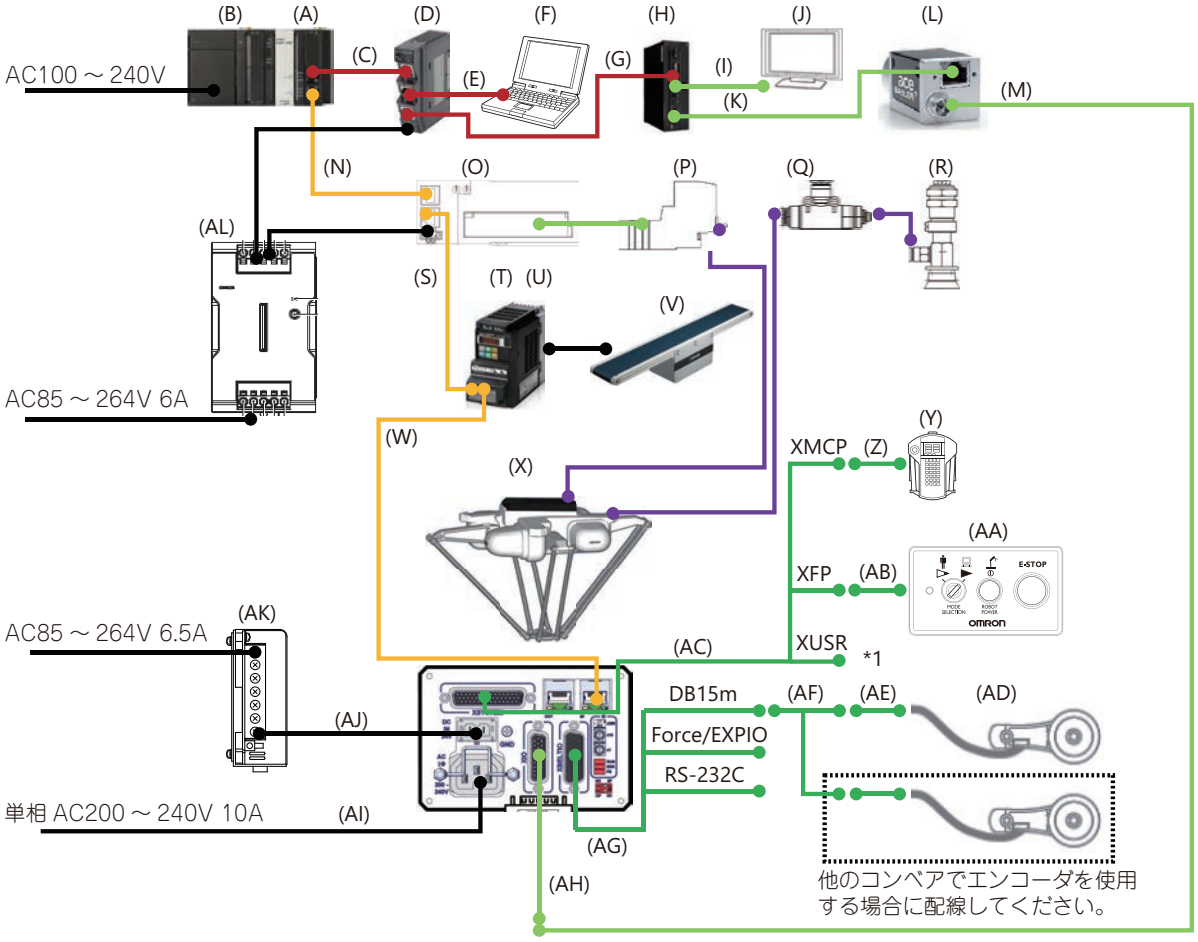
本ガイドでは、コンベア A およびコンベア B の動作について記載していません。実際の装置の動作仕様にあわせ、お客様でプログラミングしてください。

1-4-2 装置の動作モード

本装置では、手動運転と自動運転の２つの動作モードを持たせることとします。
手動運転でティーチングを行い、自動運転で実際のピック & プレース動作を実行します。

1-5 動的ピック & プレース装置のシステム構成

本ガイドで説明する動的ピック & プレース装置のシステム構成について記載します。
本ガイドでは、コンベアで搬送されているワークをピックし、指定した位置にプレースする装置を動的ピック & プレース装置と呼んでいます。カメラとエンコーダで検出した位置をもとに、ピック位置が自動的に計算されます。
図中の“XUSR”は、安全回路への配線です。実際の装置に必要な安全回路を設計してください。
図中の Y アダプタケーブルは、エンコーダへの配線です。本装置で使用するエンコーダは 1 台なので、エンコーダ延長ケーブルは片方にだけ接続します。もう片方には何も接続しません。



次に、システム構成で使用する部品を以下に示します。

記号	名称	形式	バージョン
A	ロボット統合 CPU ユニット	形 NJ501-R300	Ver.1.43
B	電源ユニット	形 NJ-PA3001	---
C	Ethernet ケーブル*1	---	---
D	産業用スイッチングハブ	形 W4S1-03B	---
E	Ethernet ケーブル*1	---	---
F	パソコン(Sysmac Studio)	---	---
G	Ethernet ケーブル*1	---	---
H	IPC Application Controller	形 AC1-152000	Ver.2.00
I	DVI ケーブル	---	---
J	ディスプレイ	---	---
K	カメラケーブル (GigE Cat.6) 10m	形 24114*2	---
L	カメラ		---
M	パワー I/O ケーブル		---
N	Ethernet ケーブル*1	---	---
O	EtherCAT デジタル I/O ターミナル	形 GX-MD3218	---
P	ソレノイドバルブ	ワークに適した真空圧、ベローズ形状のものを選定	---
Q	真空エジェクタ		---
R	吸着パッド		---
S	Ethernet ケーブル*1	---	---
T	EtherCAT 通信ユニット	形 3G3AX-MX2-ECT	---
U	インバータ*3	多機能型小型インバータ MX2 シリーズ　V1 タイプから選定	---
V	ベルトコンベア*3	---	---
W	Ethernet ケーブル*1	---	---
X	パラレルロボット iX4 650H, IP65, P30	形 RX4-2166020	4.0.C1
Y	T20 ペンダント	形 10046-010	3.0.0.1
Z	T20 アダプタケーブル		---
AA	フロントパネル	形 90356-10358	---
AB	フロントパネル ケーブル		---
AC	XSYSTEM ケーブルアセンブリ	形 RL4-2166000 に同梱	---
AD	エンコーダ IP65	形 09742-001	---
AE	エンコーダ延長ケーブル 5m		---
AF	Y-アダプタケーブル 3m	形 09443-000	---
AG	XBELTIO ケーブル 0.6m	形 13463-000	---
AH	XIO ブレイクアウトケーブル	形 04465-000	---
AI	AC 電源ケーブル	形 04118-000	---
AJ	DC 電源ケーブル	形 04120-000	---
AK	DC24V 電源 150W	形 S8FS-G15024CD	---
AL	DC24V 電源 120W	形 S8VK-S12024	---

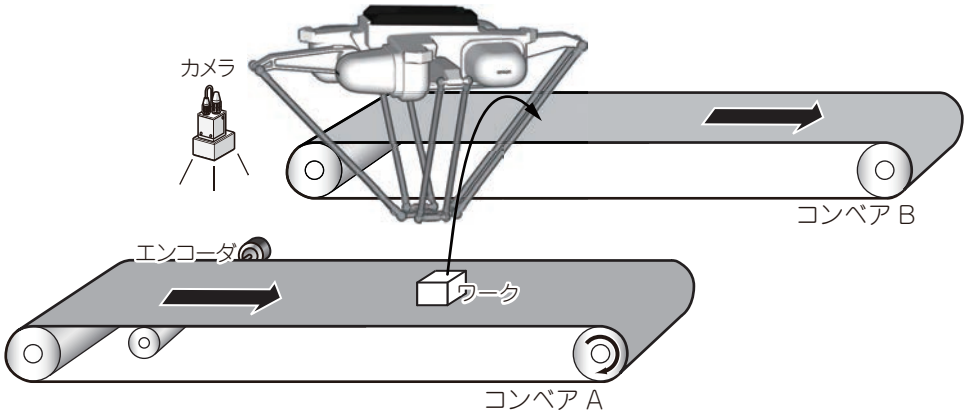
*1. Ethernet カテゴリ 5（100BASE-TX）以上のツイストペアケーブル（アルミテープと編組の二重遮へい）を使用してください。
*2. ベルト幅およびワークに適した撮像範囲と解像度のカメラを選定ください。
*3. インバータの周辺回路およびベルトコンベアとの接続については、『多機能型小型インバータ MX2 シリーズ V1 タイプ ユーザーズマニュアル(SBCE-376)』を参照ください。

1-6　動的ピック & プレース装置の動作

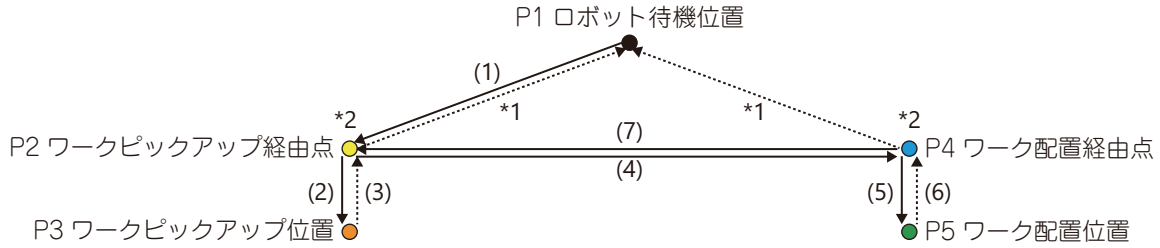
ここでは、動的ピック & プレース装置の動作仕様と操作手順について説明します。

1-6-1　動作仕様

本装置は、稼働中のコンベア A 上を搬送されているワークをカメラで検出した後にピックし、コンベア B の決められた位置でプレースします。



ロボットの具体的な動作を以下に示します。



*1: カメラで検出する際にピックアップ可能なワークが存在しない場合、一時的にワークピックアップ経由点(P2)およびワークピックアップ位置(P3)からロボット待機位置(P1)まで移動します。カメラでピックアップ可能なワークを検出している間は、(2)～(7)の動作を繰り返します。
*2: この動作図では、ワークにアプローチするときの経由点とワークからデパートするときの経由点を、同じ位置の“ワークピックアップ経由点(P2)”としています。Pack Manager の機能を使用することで、アプローチするときの経由点とデパートするときの経由点を異なる位置に設定することができます。“ワーク配置経由点(P4)”についても、同様に異なる位置を設定することができます。

番号	動作説明
(1)	ロボット待機位置(P1)からワークピックアップ経由点(P2)まで、直線補間動作で移動
(2)	ワークピックアップ経由点(P2)からワークピックアップ位置(P3)まで、直線補間動作で移動
(3)	ワークピックアップ位置(P3)からワークピックアップ経由点(P2)まで、直線補間動作で移動
(4)	ワークピックアップ経由点(P2)からワーク配置経由点(P4)まで、直線補間動作で移動
(5)	ワーク配置経由点(P4)からワーク配置位置(P5)まで、直線補間動作で移動
(6)	ワーク配置位置(P5)からワーク配置経由点(P4)まで、直線補間動作で移動
(7)	ワーク配置経由点(P4)からワークピックアップ経由点(P2)まで、直線補間動作で移動

“経由点”は、「4-3-6 Pack Manager サンプルを作成する (P.4-53)」の Pack Manager で使用する“アプローチ位置”および“デパート位置”と同じ意味です。



参考

本ガイドでは、コンベア B の動作について記載していません。実際の装置の動作仕様にあわせ、お客様でプログラミングしてください。

1-6-2 装置の動作モード

本装置では、手動運転と自動運転の２つの動作モードを持たせることとします。
手動運転でティーチングを行い、自動運転で実際のピック & プレース動作を実行します。

1-7 ハードウェア

システム構成における各ハードウェアの概要を以下に示します。
詳細については各製品のマニュアルを参照してください。

ハードウェア	形式	概要	静的	動的
ロボット統合 CPU ユニット	形 NJ501-R□□□	シーケンス制御とロボット制御を統合した CPU ユニットです。ロボット制御機能を使用するには、同梱の SD メモリカードを装着する必要があります。	○	○
EtherCAT 対応のオムロン製ロボット	形 RL□-□□□□□□□	EtherCAT に対応したオムロン製のスカラロボットです。	○	○
	形 RX□-□□□□□□□	EtherCAT に対応したオムロン製のパラレルロボットです。	○	○
パソコン	-	ロボット統合システムの設定およびデバッグを行うためのオートメーションソフトウェア Sysmac Studio がインストールされているパソコンです。*1	○	○
フロントパネル	形 90356-10358	ロボットの制御パネルです。モード変更、ハイパワーの有効化、非常停止のためのスイッチがあります。	○	○
T20 ペンダント	形 10046-010	ロボットのティーチングペンダントです。座標のティーチングに使用します。安全機能として非常停止ボタンとイネーブルスイッチを内蔵しています。	○	○
安全回路	-	安全入出力機器とセーフティコントローラなどで構成された安全回路です。詳細は「A-1 ピック & プレース装置の安全機能の設計例 (P.A-2)」を参照してください。	○	○
外部接続機器	-	ロボットのデジタル入出力端子に接続された入出力機器です。	○	○
EtherCAT スレーブ	-	デジタル I/O やサーボ、インバータなどの EtherCAT スレーブです。	○	○
産業用スイッチングハブ	形 W4S1-□□□	Ethernet ネットワークの分岐配線に使用します。	–	○
IPC Application Controller	形 AC1-152000	産業用パソコンです。プリインストールされたロボット制御システム用ソフトウェア群（アプリケーションマネージャ）を実行します。カメラ接続用インタフェース（PoE ポート）を持っています。	–	○
産業用カメラ	形 24114-□□□	画像センシング用カメラです。IPC Application Controller に接続することで、ワークの認識や検査をすることができます。	–	○
インバータ	3G3MX2-A□□□□-V1	ベルトコンベアを駆動するためのインバータです。多機能型小型インバータ MX2 シリーズ V1 タイプから選定します。	–	○
EtherCAT 通信ユニット	形 3G3AX-MX2-ECT	インバータに取り付けて EtherCAT 通信を行う通信ユニットです。	–	○
ベルトコンベア	SVKN-100-1000-25-T100-IM-9-H-A	ワークを搬送するための MISUMI 社製のベルトコンベアです。	–	○
エンコーダキット IP65	形 09742-001	ベルトコンベアに取り付けられたエンコーダです。	–	○

*1. 推奨パソコン動作環境については、『Sysmac Studio ロボット統合システム構築機能 オペレーションマニュアル ロボット統合 CPU ユニット編 (SBCA-464)』を参照してください。

1-8 ソフトウェア

システム構成における各ソフトウェアの概要を以下に示します。
ロボット統合システムの開発環境として Sysmac Studio を使用し、アプリケーションを実現するランタイムとしてアプリケーションマネージャを使用します。

1-8-1 Sysmac Studio

パソコン上で動作するロボット統合システムの開発環境です。

使用上の注意

ロボット統合システムでは、Sysmac Studio Ver.1.□□ 64bit 版 DVD(SYSMAC-SE200D-64)を使用してください。32bit 版 DVD(SYSMAC-SE200D)は使用できません。

Sysmac Studio は以下の機能を持っています。

機能	概要
ロボット制御機能	ロボット統合 CPU ユニットと IPC Application Controller を使ってロボットを制御するための設定およびプログラムを作成する機能です。本機能を使用するためには、スタンダードエディションのライセンス(SYSMAC-SE2□□L)が必要です。
3D シミュレーション機能	ロボットや周辺機器を含めた 3D シミュレーションを行う機能です。本機能を使用するためには、3D シミュレーションオプションのライセンス(SYSMAC-SA4□□L-64)が必要です。

Sysmac Studio のロボット制御機能については、『Sysmac Studio ロボット統合システム構築機能 オペレーションマニュアル ロボット統合 CPU ユニット編 (SBCA-464)』、3D シミュレーション機能については、『Sysmac Studio 3D シミュレーション機能 オペレーションマニュアル (SBCA-482)』を参照してください。

参考

Sysmac Studio のマルチビューエクスプローラ上の項目がどのハードウェアの設定に対応しているかについては、「A-3 Sysmac Studio の設定項目と設定対象 (P.A-28)」を参照してください。

1-8-2 アプリケーションマネージャ

IPC Application Controller にプリインストールされているソフトウェアである ACE の機能のひとつです。
アプリケーションマネージャは以下の機能を持っています。

機能	説明
プロセスマネージャ	ワークをどのロボットに処理させるかを自動で判断し、負荷分散を行うことができます。
レシピマネージャ	ワークの品種変更に伴うロボット制御パラメータの管理を行います。
Robot Vision Manager	カメラを用いた画像センシングにより、ワークの位置検出および外観検査をすることができます。本機能を有効にするには、IPC Application Controller に USB ドングル（20410-000 または 20433-000）を装着する必要があります。
Pack Manager	アプリケーションサンプルで作成した包装アプリケーションを実行します。本機能を有効にするには、IPC Application Controller に USB ドングル（20409-000 または 20433-000） を装着する必要があります。

使用上の注意

ACE は Smart Controller EX を用いたロボットシステムの開発環境としての機能も持ちますが、ロボット統合システムでは開発環境として使用することはできません。ロボット統合システムの開発には Sysmac Studio を使用してください。

1-9 ユーザプログラム

ロボット制御のためのユーザプログラムは Sysmac Studio を使って作成します。
ユーザプログラムの作成方法には複数の方法があります。複数の方法で作成したプログラムを連携させてアプリケーションを実現します。

1-9-1 ユーザプログラムの作成方法の種類

ユーザプログラムの作成方法の種類には以下があります。

方法	概要
シーケンス制御プログラム	ロボット統合 CPU ユニット上で動作するユーザプログラムを、IEC61131-3 で定義される言語でプログラミングすることができます。ロボット制御命令、ロボット制御システム変数を使用してロボットを制御します。 単純なロボット動作や、ロボットと周辺機器を連携させた制御を実現する場合に適しています。 本ガイドでは、ラダー図言語を使用します。
V+プログラム	ロボット統合 CPU ユニット上で動作するユーザプログラムを、オムロン製ロボット専用のロボット制御用プログラミング言語である V+言語でプログラミングすることができます。V+キーワードを使用してロボットを制御します。 また、ACE Sight V+キーワードや V+モジュール API を使用することで、アプリケーションマネージャの機能を呼び出して利用することができます。 複雑なロボット制御を実現する場合に適しています。
C#プログラム	IPC Application Controller 上で動作するユーザプログラムを、C#言語でプログラミングすることができます。アプリケーションマネージャの機能を呼び出して利用する場合に使用します。
アプリケーションサンプル	対話型の設定ウィザードで、アプリケーションサンプルを作成することができます。アプリケーションサンプルには IPC Application Controller およびロボット統合 CPU ユニットの設定とプログラムが含まれます。 本ガイドでは、Pack Manager サンプルを使用します。

1-9-2 シーケンス制御プログラムと V+プログラムの使い分け

シーケンス制御プログラムと V+プログラムの使い分けは以下のとおりです。

- シーケンス制御およびロボット制御を含むシステム全体の状態や異常、インタロックの管理は、シーケンス制御プログラムで行います。
- ロボット制御はシーケンス制御プログラムと V+プログラムを組み合わせて行います。
- ロボット以外の制御は、シーケンス制御プログラムで行います。

シーケンス制御プログラムと V+プログラムで可能なロボット制御の違いの代表的な例を以下に示します。
シーケンス制御プログラムでは、V+プログラムで可能な制御のうちの一部が可能です。

分類	項目	可否（可の場合は指定方法）	
		シーケンス制御プログラム	V+プログラム
ロボットの動作	関節補間動作	RC_MoveDirect *1	MOVE
	直線補間動作	RC_MoveLinear *1	MOVES
	円弧補間動作	不可	MOVEC
	ジョグ動作	不可	JOG
	ジョイント座標系動作	不可	JMOVE
	ツール座標系動作	不可	APPRO/DEPART/ALIGN

分類	項目	可否（可の場合は指定方法）	
		シーケンス制御プログラム	V+プログラム
ロボットの座標	フランジ面	_RC_RBT[*].TCPActPos *2	HERE
	ツール先端座標	_RC_RBT[*].TCPActPos *2 (RC_SetToolTransform *1)	HERE (TOOL)
	各軸ジョイント座標	_RC_RBT[*].JointActPos *2	#PHERE
その他の機能	ロボット内蔵 I/O	_RC_RBT_IO[*].XIO *2	SIGNAL/SIG
	ロボット拡張 I/O	_RC_RBT_IO[*].IOBlox1 *2 _RC_RBT_IO[*].IOBlox2 *2	SIGNAL/SIG
	ティーチング	不可	位置変数
	エンドエフェクタ操作	_RC_RBT_IO[*].RO *2	SIGNAL/SIG
	ロボット位置ラッチ	不可	LATCH
	ローカルエンコーダラッチ	不可	DEVICE
	アプリケーションマネージャ連携	不可	ACE Sight V+キーワード V+モジュール API

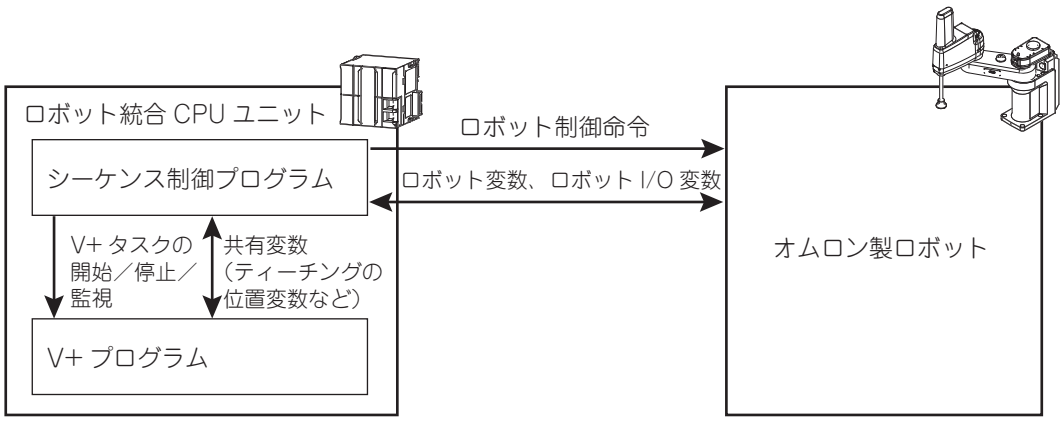
*1. ロボット制御命令です。
*2. ロボット制御システム変数です。

シーケンス制御プログラムでロボットを制御する場合の詳細については、『NJ シリーズ ロボット統合 CPU ユニット ユーザーズマニュアル(SBCE-436)』を参照してください。
V+プログラムでロボットを制御する場合の詳細については、『eV+3 ユーザーズマニュアル (SBCE-486)』および『eV+3 キーワードリファレンスマニュアル (SBCE-487)』を参照してください。

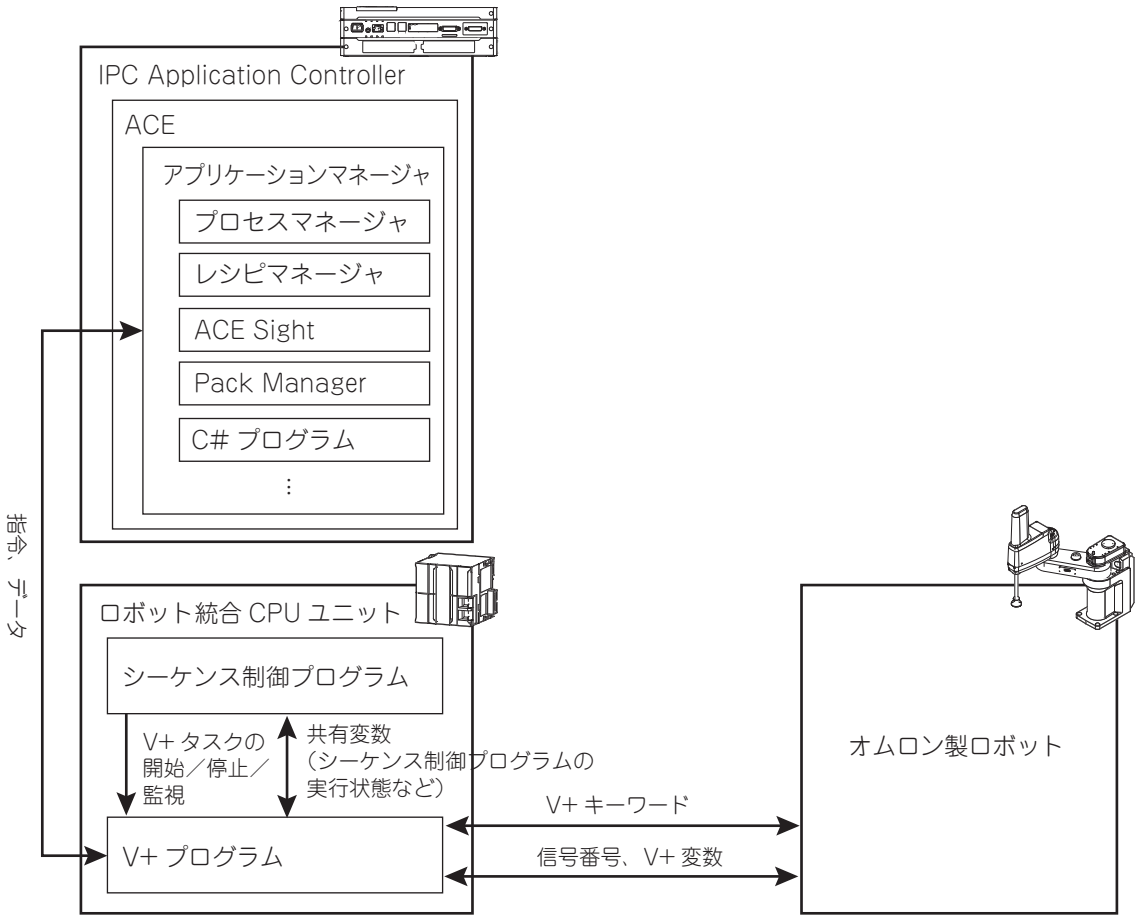
1-9-3 指令およびデータの流れ

シーケンス制御プログラム、V+プログラム、アプリケーションマネージャ、ロボット間の指令およびデータの流れを説明します。

シーケンス制御プログラムでは、ロボット制御命令、ロボット変数、ロボット I/O 変数を使ってロボットを制御することができます。
ティーチングした位置情報は V+プログラムの位置変数であるため、シーケンス制御プログラムで使用するためには、RC_ExecVpPrgTask で V+プログラムを呼び出し、共有変数に位置情報をコピーする必要があります。



V+プログラムでは、V+キーワード、信号番号、V+変数を使ってロボットを制御することができます。
また、ACE Sight V+キーワードおよび V+モジュール API により IPC Application Controller のアプリケーションマネージャの機能を実行することができます。
V+プログラムは、シーケンス制御プログラムから RC_ExecVpPrgTask で起動、RC_AbortVpPrgTask で停止します。



シーケンス制御プログラムでのロボット制御機能および共有変数の詳細については、『NJ シリーズ ロボット統合 CPU ユニット ユーザーズマニュアル(SBCE-436)』を参照してください。
V+プログラムでのロボット制御機能の詳細については、『eV+3 ユーザーズマニュアル (SBCE-486)』および『eV+3 キーワードリファレンスマニュアル (SBCE-487)』を参照してください。

2

事前に準備すること

この章では、Sysmac Studio のインストールについて説明します。

2-1 Sysmac Studio をインストールする.....2-2

2-1 Sysmac Studio をインストールする

ロボット統合システムでは、構成や設定の作成、プログラム作成、デバッグ、シミュレーションを行うサポートソフトとして Sysmac Studio Ver.1.□□ 64bit 版 DVD(SYSMAC-SE200D-64)を使用します。

1 **Sysmac Studio のインストールディスクを DVD-ROM ドライブにセットしてください。**
セットアッププログラムが自動的に起動され、［セットアップ言語の選択］ダイアログボックスが表示されます。

2 **インストールする言語を選択し、［OK］ボタンをクリックしてください。**
Sysmac Studio のセットアップウィザードが表示されます。

3 **セットアップウィザードに従ってインストールしてください。**

4 **インストールが完了したら、パソコンを再起動してください。**

参考

- Sysmac Studio をインストールするパソコンの動作環境については、『Sysmac Studio ロボット統合システム構築機能 オペレーションマニュアル ロボット統合 CPU ユニット編 (SBCA-464)』を参照してください。
- 上記の手順で Sysmac Studio がインストールできない場合は、『Sysmac Studio Version 1 オペレーションマニュアル (SBCA-470)』を参照してください。

3

静的ピック & プレース装置の実現例

この章では、静的ピック & プレース装置の実現例について説明します。
本ガイドでは、決められた位置で静止しているワークを、決められた位置へ移動させる装置を静的ピック & プレース装置と呼んでいます。
シーケンス制御プログラムおよび V+プログラムで実現することができます。

3-1 静的ピック & プレース装置のプログラム仕様 3-2
3-1-1 シーケンス制御プログラムで動作を制御する場合 3-3
3-1-2 V+プログラムで動作を制御する場合 3-9

3-2 基本的な立ち上げ手順 3-15

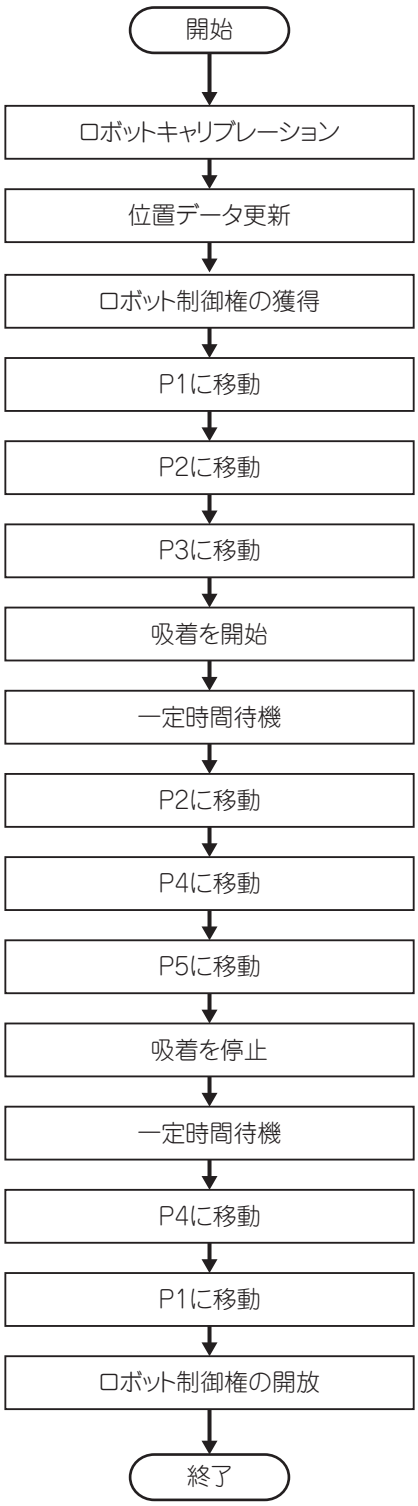
3-3 プログラミングとシミュレーションの手順 3-16
3-3-1 プロジェクトファイルを新規作成する 3-16
3-3-2 EtherCAT ネットワーク構成を作成する 3-17
3-3-3 シーケンス制御プログラムを作成する 3-20
3-3-4 V+プログラムを作成する 3-26
3-3-5 3D 形状データを配置する 3-33
3-3-6 シミュレーションを開始する 3-59
3-3-7 ティーチングする 3-60
3-3-8 V+のプログラムと変数の電源 ON 時の自動読込を設定する 3-70
3-3-9 プログラムを実行して動作を確認する 3-74

3-4 設置、配線する 3-79
3-4-1 ロボット統合 CPU ユニットと EtherCAT デジタル I/O ターミナル間を配線する 3-79
3-4-2 EtherCAT デジタル I/O ターミナルの EtherCAT ノードアドレスを設定する 3-80
3-4-3 EtherCAT デジタル I/O ターミナルとロボット間を配線する 3-80
3-4-4 ロボットの EtherCAT ノードアドレスを設定する 3-81
3-4-5 ロボット統合 CPU ユニットとパソコン間を配線する 3-81
3-4-6 ロボットと T20 ペンダント、フロントパネル間を配線する 3-82
3-4-7 EtherCAT デジタル I/O ターミナルとソレノイドバルブ間を配線する 3-83

3-5 実機の立ち上げ手順 3-85
3-5-1 オンライン接続する 3-85
3-5-2 設定とプログラムを転送する 3-87
3-5-3 ティーチングする 3-89
3-5-4 プログラムを実行して動作を確認する 3-103

3-1 静的ピック & プレース装置のプログラム仕様

静的ピック & プレース装置の動作を、シーケンス制御プログラムで制御する場合のプログラム仕様とV+プログラムで制御する場合のプログラム仕様について説明します。
以下にプログラムのフローチャートを記載します。フローチャート中の P1 から P5 が表す位置については、「1-4-1 動作仕様 (P.1-5)」を参照してください。



シーケンス制御プログラムで動作を制御する方法と、V+プログラムで動作を制御する方法があります。いずれかの方法でプログラムを作成します。
プログラムの作成方法については、「3-3 プログラミングとシミュレーションの手順 (P.3-16)」の中で説明します。

3-1-1 シーケンス制御プログラムで動作を制御する場合

静的ピック & プレース装置の動作を、シーケンス制御プログラムで制御する場合のプログラム仕様について説明します。

以下のプログラムで構成されます。

プログラム名	言語	説明
Main	シーケンス制御プログラム（ラダー図言語）	メインプログラム
Run	シーケンス制御プログラム（ラダー図言語）	動作プログラム
loccopy	V+プログラム	変数コピープログラム

共有変数

本プログラムで使用する共有変数の一覧を以下に記載します。

No.	変数名	データ型	説明
1	eLoc_Wait	ARRAY[0..5] OF LREAL	ロボットの待機位置を、V+プログラムからシーケンス制御プログラムへ引き渡すときに使用します。
2	eLoc_Place	ARRAY[0..5] OF LREAL	ロボットの配置位置を、V+プログラムからシーケンス制御プログラムへ引き渡すときに使用します。
3	eLoc_Pick	ARRAY[0..5] OF LREAL	ロボットのピックアップ位置を、V+プログラムからシーケンス制御プログラムへ引き渡すときに使用します。
4	eBool_Exe	BOOL	V+タスクが実行開始したことを、シーケンス制御プログラムで確認するときに使用します。

シーケンス制御プログラム

静的ピック & プレース装置を制御するためのシーケンス制御プログラムについて説明します。
シーケンス制御プログラムでは、グローバル変数を登録し、メインプログラム“Main”と動作プログラム“Run”の２つのプログラムを作成します。

- **グローバル変数**

  シーケンス制御プログラムで使用するグローバル変数を登録します。

名称	データ型	初期値	割付先	保持	コンスタント	ネットワーク公開	コメント
E001_In_Bit00	BOOL			☐	☐	非公開	
E001_In_Bit01	BOOL			☐	☐	非公開	
E001_In_Bit02	BOOL			☐	☐	非公開	
E001_In_Bit03	BOOL		ECAT://node#1/Read input 1st word/In Bit03	☐	☐	非公開	
E001_In_Bit04	BOOL		ECAT://node#1/Read input 1st word/In Bit04	☐	☐	非公開	
E001_In_Bit05	BOOL		ECAT://node#1/Read input 1st word/In Bit05	☐	☐	非公開	
E001_In_Bit06	BOOL		ECAT://node#1/Read input 1st word/In Bit06	☐	☐	非公開	
E001_In_Bit07	BOOL		ECAT://node#1/Read input 1st word/In Bit07	☐	☐	非公開	
E001_In_Bit08	BOOL		ECAT://node#1/Read input 1st word/In Bit08	☐	☐	非公開	
E001_In_Bit09	BOOL		ECAT://node#1/Read input 1st word/In Bit09	☐	☐	非公開	
E001_In_Bit10	BOOL		ECAT://node#1/Read input 1st word/In Bit10	☐	☐	非公開	
E001_In_Bit11	BOOL		ECAT://node#1/Read input 1st word/In Bit11	☐	☐	非公開	
E001_In_Bit12	BOOL		ECAT://node#1/Read input 1st word/In Bit12	☐	☐	非公開	
E001_In_Bit13	BOOL		ECAT://node#1/Read input 1st word/In Bit13	☐	☐	非公開	
E001_In_Bit14	BOOL		ECAT://node#1/Read input 1st word/In Bit14	☐	☐	非公開	
E001_In_Bit15	BOOL		ECAT://node#1/Read input 1st word/In Bit15	☐	☐	非公開	
E001_Minor_Fault	BOOL		ECAT://node#1/Sysmac Error Status/Minor Fault	☐	☐	非公開	
E001_Observation	BOOL		ECAT://node#1/Sysmac Error Status/Observation	☐	☐	非公開	
E001_Out_Bit00	BOOL			☐	☐	非公開	4001
E001_Out_Bit01	BOOL		ECAT://node#1/Write output 1st word/Out Bit01	☐	☐	非公開	
E001_Out_Bit02	BOOL		ECAT://node#1/Write output 1st word/Out Bit02	☐	☐	非公開	
E001_Out_Bit03	BOOL		ECAT://node#1/Write output 1st word/Out Bit03	☐	☐	非公開	
E001_Out_Bit04	BOOL		ECAT://node#1/Write output 1st word/Out Bit04	☐	☐	非公開	
E001_Out_Bit05	BOOL		ECAT://node#1/Write output 1st word/Out Bit05	☐	☐	非公開	
E001_Out_Bit06	BOOL		ECAT://node#1/Write output 1st word/Out Bit06	☐	☐	非公開	
E001_Out_Bit07	BOOL		ECAT://node#1/Write output 1st word/Out Bit07	☐	☐	非公開	
E001_Out_Bit08	BOOL		ECAT://node#1/Write output 1st word/Out Bit08	☐	☐	非公開	
E001_Out_Bit09	BOOL		ECAT://node#1/Write output 1st word/Out Bit09	☐	☐	非公開	
E001_Out_Bit10	BOOL		ECAT://node#1/Write output 1st word/Out Bit10	☐	☐	非公開	
E001_Out_Bit11	BOOL		ECAT://node#1/Write output 1st word/Out Bit11	☐	☐	非公開	
E001_Out_Bit12	BOOL		ECAT://node#1/Write output 1st word/Out Bit12	☐	☐	非公開	
E001_Out_Bit13	BOOL		ECAT://node#1/Write output 1st word/Out Bit13	☐	☐	非公開	
E001_Out_Bit14	BOOL		ECAT://node#1/Write output 1st word/Out Bit14	☐	☐	非公開	
E001_Out_Bit15	BOOL		ECAT://node#1/Write output 1st word/Out Bit15	☐	☐	非公開	
E001_Read_input_1st_word	WORD		ECAT://node#1/Read input 1st word	☐	☐	非公開	
E001_Sysmac_Error_Status	BYTE		ECAT://node#1/Sysmac Error Status	☐	☐	非公開	
E001_Write_output_1st_word	WORD		ECAT://node#1/Write output 1st word	☐	☐	非公開	
eBool_Exe	BOOL			☐	☐	非公開	V+プログラム実行開始
eLoc_Pick	ARRAY[0..5] OF LREAL			☐	☐	非公開	ピックアップ位置
eLoc_Place	ARRAY[0..5] OF LREAL			☐	☐	非公開	プレース位置
eLoc_Wait	ARRAY[0..5] OF LREAL			☐	☐	非公開	待機位置
gAutoOn	BOOL			☐	☐	非公開	自動運転
gAutoRun	BOOL			☐	☐	非公開	自動運転開始
gCalib	BOOL		ECAT://node#1/Read input 1st word/In Bit01	☐	☐	非公開	キャリブレーション実行ボタン
gPosDataOK	BOOL			☐	☐	非公開	位置データ更新完了
gRC_Err	BOOL			☐	☐	非公開	ロボット制御エラー
gReset	BOOL		ECAT://node#1/Read input 1st word/In Bit02	☐	☐	非公開	自動運転解除ボタン
gStart	BOOL		ECAT://node#1/Read input 1st word/In Bit00	☐	☐	非公開	自動運転開始ボタン
gSuctionOff	BOOL			☐	☐	非公開	吸着終了
gSuctionOn	BOOL			☐	☐	非公開	吸着開始
gSysOK	BOOL			☐	☐	非公開	運転可能
gTaskStatus	INT			☐	☐	非公開	タスクステータス
RC_Robot001	_sRC_RBT_REF		RC://_RC_RBT[0]	☐	☑	非公開	
RC_Robot001_IO	_sRC_RBT_IO_REF		RC://_RC_RBT_IO[0]	☐	☐	非公開	

● メインプログラム(Main)

メインプログラム“Main”では、以下のロボットおよびEtherCAT デジタル I/O ターミナルの状態を監視し、自動運転の開始可否を判断します。
各変数の詳細は、各機器のマニュアルを参照してください。

No.	変数名	説明
1	_EC_PDSlavTbl[1]	EtherCAT デジタル I/O ターミナルの通信が確立しているか確認します。
2	_RC_RBT[0].DrvStatus.RunMode	ロボット統合 CPU ユニットとロボットの通信が確立しているか確認します。
3	GetRCError	ロボット制御機能モジュールに起因するエラーが発生しているか確認します。
4	_RC_RBT[0].DrvStatus.ESTOP	ロボットが ESTOP 状態か確認します。

No.	変数名	説明
5	_RC_RBT[0].DrvStatus.Manual	ロボットの動作モードがマニュアルモードか確認します。
6	_RC_RBT[0].DrvStatus.PowerEnabled	ロボットのハイパワーが有効か確認します。

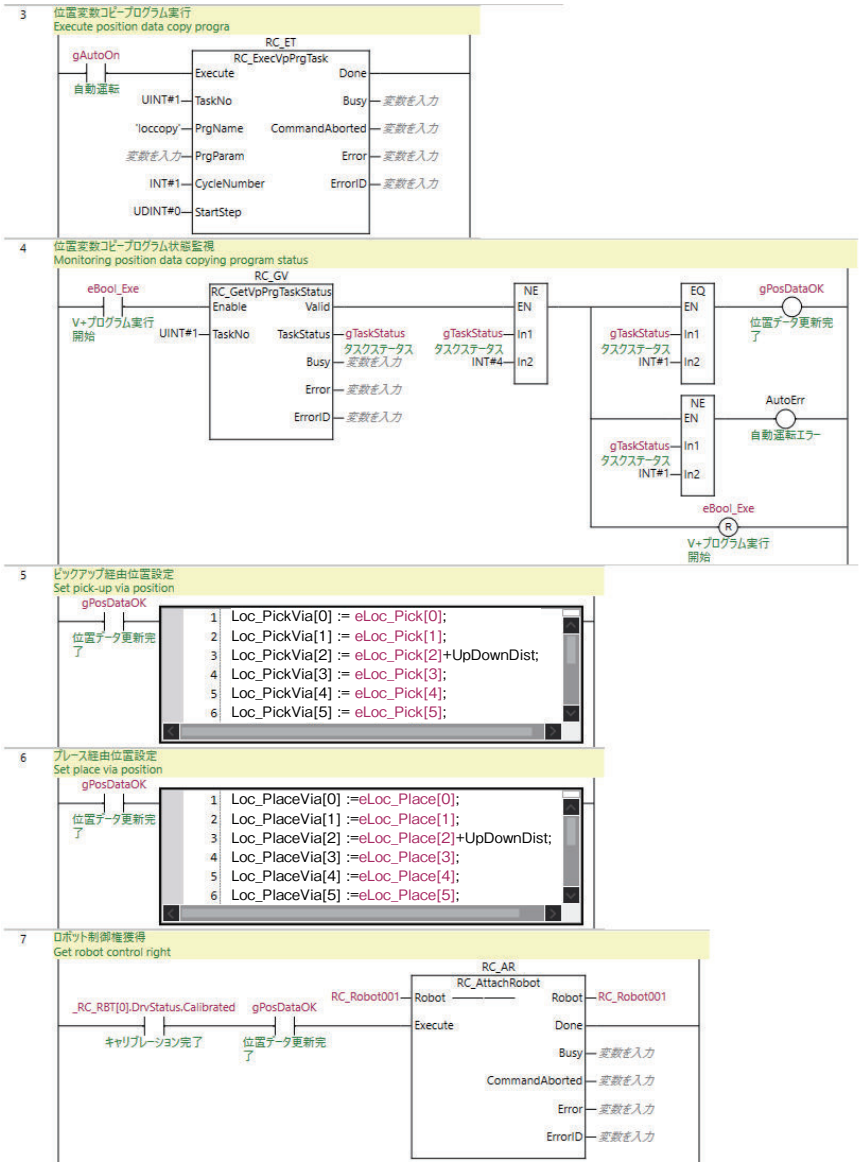
以下にプログラムを記載します。

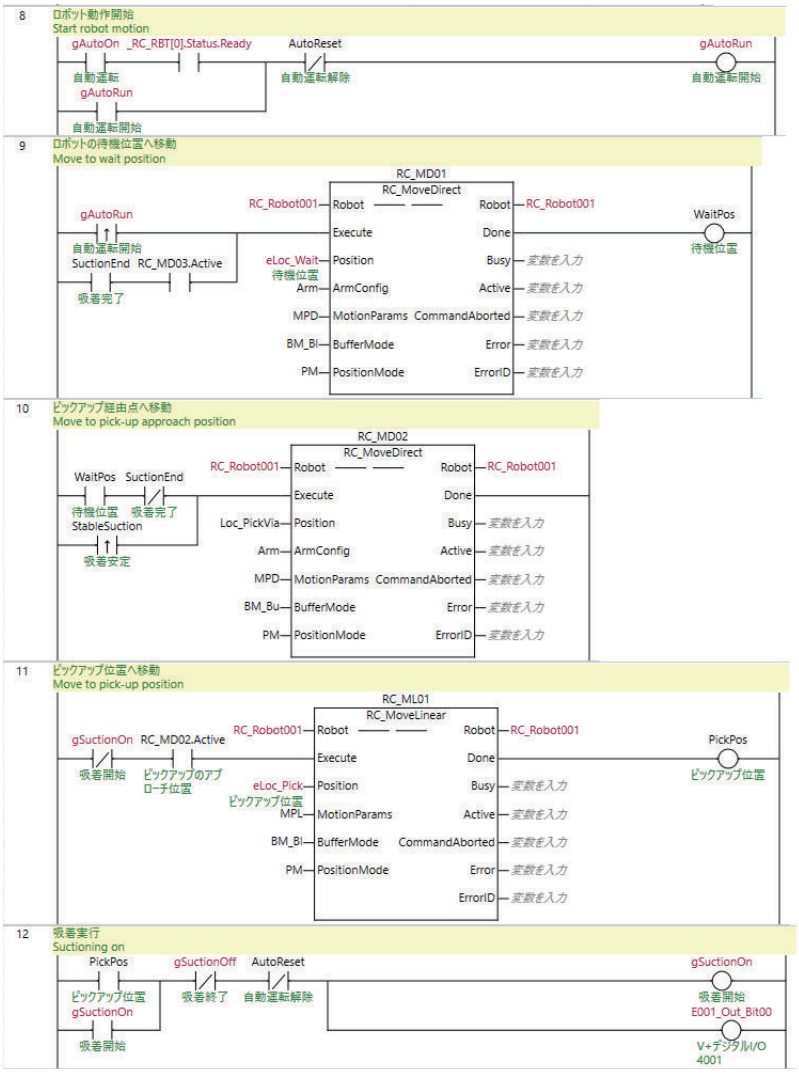


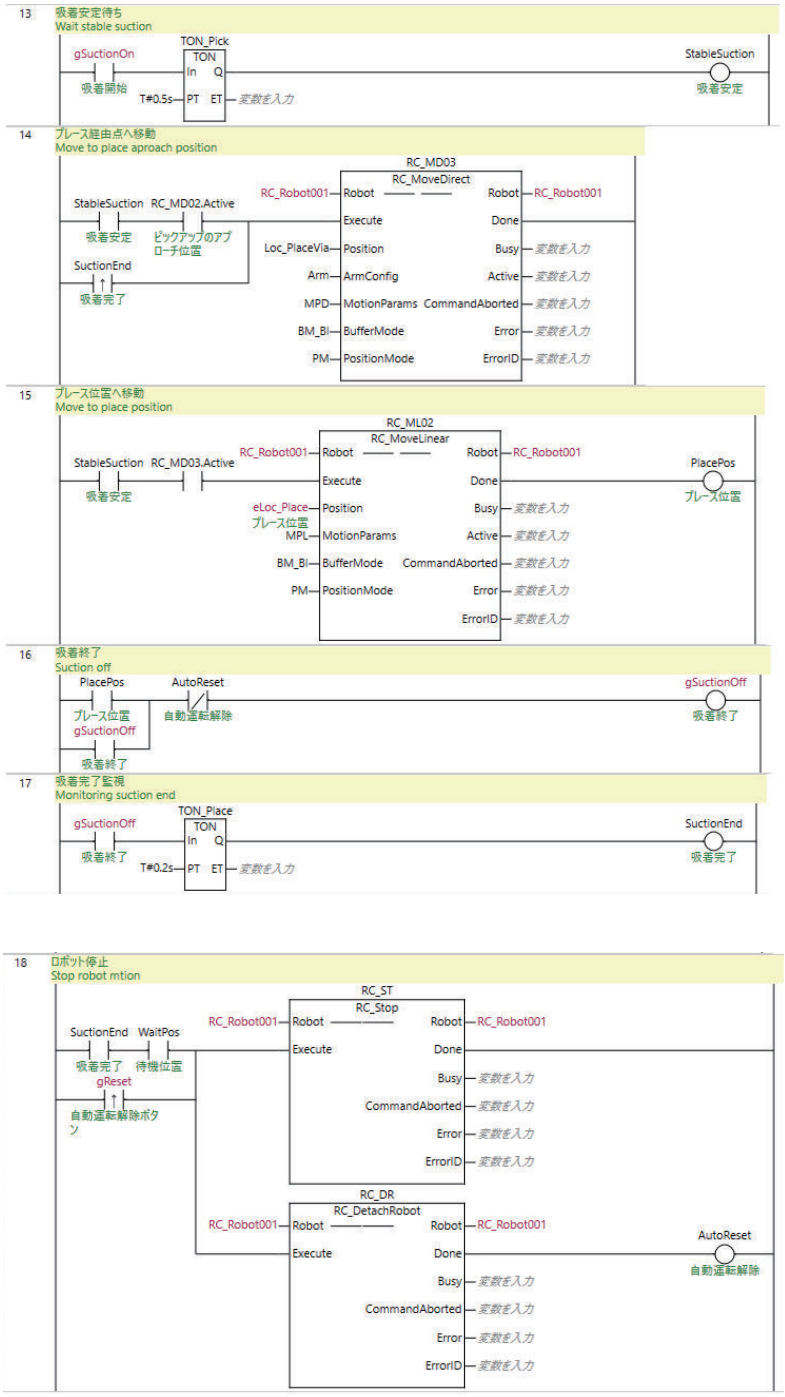
● 動作プログラム(Run)

動作プログラム“Run”では、ロボットを動作させます。
以下にプログラムを記載します。




3
位置変数コピープログラム実行
Execute position data copy progra
RC_ET
RC_ExecVpPrgTask
gAutoOn
自動運転
Execute
Done
UINT#1
TaskNo
Busy
変数を入力
'loccopy'
PrgName
CommandAborted
変数を入力
変数を入力
PrgParam
Error
変数を入力
INT#1
CycleNumber
ErrorID
変数を入力
UDINT#0
StartStep
4
位置変数コピープログラム状態監視
Monitoring position data copying program status
RC_GV
RC_GetVpPrgTaskStatus
eBool_Exe
V+プログラム実行開始
Enable
Valid
UINT#1
TaskNo
TaskStatus
gTaskStatus
タスクステータス
Busy
変数を入力
Error
変数を入力
ErrorID
変数を入力
NE
EN
In1
In2
INT#4
EQ
gPosDataOK
位置データ更新完了
INT#1
AutoErr
自動運転エラー
R
5
ピックアップ経由位置設定
Set pick-up via position
gPosDataOK
位置データ更新完了
1 Loc_PickVia[0] := eLoc_Pick[0];
2 Loc_PickVia[1] := eLoc_Pick[1];
3 Loc_PickVia[2] := eLoc_Pick[2]+UpDownDist;
4 Loc_PickVia[3] := eLoc_Pick[3];
5 Loc_PickVia[4] := eLoc_Pick[4];
6 Loc_PickVia[5] := eLoc_Pick[5];
6
プレース経由位置設定
Set place via position
gPosDataOK
位置データ更新完了
1 Loc_PlaceVia[0] :=eLoc_Place[0];
2 Loc_PlaceVia[1] :=eLoc_Place[1];
3 Loc_PlaceVia[2] :=eLoc_Place[2]+UpDownDist;
4 Loc_PlaceVia[3] :=eLoc_Place[3];
5 Loc_PlaceVia[4] :=eLoc_Place[4];
6 Loc_PlaceVia[5] :=eLoc_Place[5];
7
ロボット制御権獲得
Get robot control right
RC_AR
RC_AttachRobot
_RC_RBT[0].DrvStatus.Calibrated
キャリブレーション完了
gPosDataOK
位置データ更新完了
RC_Robot001
Robot
Execute
Done
Busy
CommandAborted
Error
ErrorID


8
ロボット動作開始
Start robot motion
gAutoOn
_RC_RBT[0].Status.Ready
AutoReset
gAutoRun
自動運転
自動運転解除
自動運転開始
gAutoRun
自動運転開始
9
ロボットの待機位置へ移動
Move to wait position
RC_MD01
RC_MoveDirect
RC_Robot001
Robot
Robot
RC_Robot001
gAutoRun
Execute
Done
WaitPos
自動運転開始
待機位置
SuctionEnd
RC_MD03.Active
eLoc_Wait
待機位置
Position
Busy
変数を入力
吸着完了
Arm
ArmConfig
Active
変数を入力
MPD
MotionParams
CommandAborted
変数を入力
BM_Bl
BufferMode
Error
変数を入力
PM
PositionMode
ErrorID
変数を入力
10
ピックアップ経由点へ移動
Move to pick-up approach position
RC_MD02
RC_MoveDirect
WaitPos
SuctionEnd
RC_Robot001
Robot
Robot
RC_Robot001
待機位置
吸着完了
Execute
Done
StableSuction
Loc_PickVia
Position
Busy
変数を入力
吸着安定
Arm
ArmConfig
Active
変数を入力
MPD
MotionParams
CommandAborted
変数を入力
BM_Bu
BufferMode
Error
変数を入力
PM
PositionMode
ErrorID
変数を入力
11
ピックアップ位置へ移動
Move to pick-up position
RC_ML01
RC_MoveLinear
gSuctionOn
RC_MD02.Active
RC_Robot001
Robot
Robot
RC_Robot001
PickPos
吸着開始
ピックアップのアプローチ位置
Execute
Done
ピックアップ位置
eLoc_Pick
Position
Busy
変数を入力
MPL
MotionParams
Active
変数を入力
BM_Bl
BufferMode
CommandAborted
変数を入力
PM
PositionMode
Error
変数を入力
ErrorID
変数を入力
12
吸着実行
Suctioning on
PickPos
gSuctionOff
AutoReset
gSuctionOn
ピックアップ位置
吸着終了
自動運転解除
吸着開始
gSuctionOn
吸着開始
E001_Out_Bit00
V+デジタルI/O
4001



V+プログラム

静的ピック & プレース装置の動作をシーケンス制御プログラムで制御するために必要な V+プログラムについて説明します。
V+プログラムでは、グローバル変数を登録し、変数コピープログラム“loccopy”と読み出しプログラム“auto”の２つのプログラムを作成します。

● グローバル変数

V+プログラムで使用するグローバル変数を登録します。

名前	種類	値	ロボット	表示モード	カテゴリ	説明
gl.pick	ロケーション	400.000 50.000 211.000 0.000 180.000 -180.000	R1Cobra600	表示しない		
gl.place	ロケーション	-285.000 500.000 211.000 0.000 180.000 180.000	R1Cobra600	表示しない		
gl.wait	ロケーション	400.000 0.000 330.000 0.000 180.000 -150.000	R1Cobra600	表示しない		

● 変数コピープログラム(loccopy)

変数コピープログラム“loccopy”では、V+プログラムのグローバル変数を、シーケンス制御プログラムとの共有変数にコピーします。
以下にプログラムを記載します。

```
.PROGRAM loccopy()
        GLOBAL gl.pick, gl.place, gl.wait
        EXTERNAL eloc_pick[]
        EXTERNAL eloc_place[]
        EXTERNAL eloc_wait[]
        EXTERNAL ebool_exe
        ; Copy location data to external variables
        DECOMPOSE eloc_pick[] = gl.pick
        DECOMPOSE eloc_place[] = gl.place
        DECOMPOSE eloc_wait[] = gl.wait
        ebool_exe = TRUE
.END
```

● 読み出しプログラム

読み出しプログラム“auto”では、V+プログラムおよびグローバル変数を、SD メモリカードからメインメモリに読み出します。
読み出しプログラム“auto”については、「3-3-8 V+のプログラムと変数の電源 ON 時の自動読込を設定する (P.3-70)」を参照してください。

3-1-2 V+プログラムで動作を制御する場合

静的ピック & プレース装置の動作を、V+プログラムで制御する場合のプログラム仕様について説明します。

以下のプログラムで構成されます。

プログラム名	言語	説明
Main	シーケンス制御プログラム（ラダー図言語）	メインプログラム
run	V+プログラム	ロボット指令発行プログラム
supervisor	V+プログラム	状態監視プログラム

使用上の注意

V+プログラムは、ロボット統合 CPU ユニットがプログラムモードになった場合でも継続して動作します。supervisor は、プログラムモードを検出し、ロボット動作および V+プログラムを停止させるプログラムです。

共有変数

シーケンス制御プログラムで使用する共有変数の一覧を以下に記載します。

No.	変数名	データ型	説明
1	eBool_ExeT1	BOOL	V+タスク 1 が実行開始したことを、シーケンス制御プログラムで確認するときに使用します。
2	eBool_ExeT2	BOOL	V+タスク 2 が実行開始したことを、シーケンス制御プログラムで確認するときに使用します。
3	eBool_Mode	BOOL	現在のロボット統合 CPU ユニットの動作モードを、V+プログラムで確認するときに使用します。
4	eBool_Reset	BOOL	自動運転時にリセットボタンが押されたことを、V+プログラムで確認するときに使用します。

シーケンス制御プログラム

静的ピック & プレース装置の動作を V+プログラムで制御するために必要なシーケンス制御プログラムについて説明します。
シーケンス制御プログラムでは、グローバル変数を登録し、周辺機器の状態を監視して自動運転開始可否を判断するメインプログラム“Main”を作成します。

● グローバル変数

シーケンス制御プログラムで使用するグローバル変数を登録します。

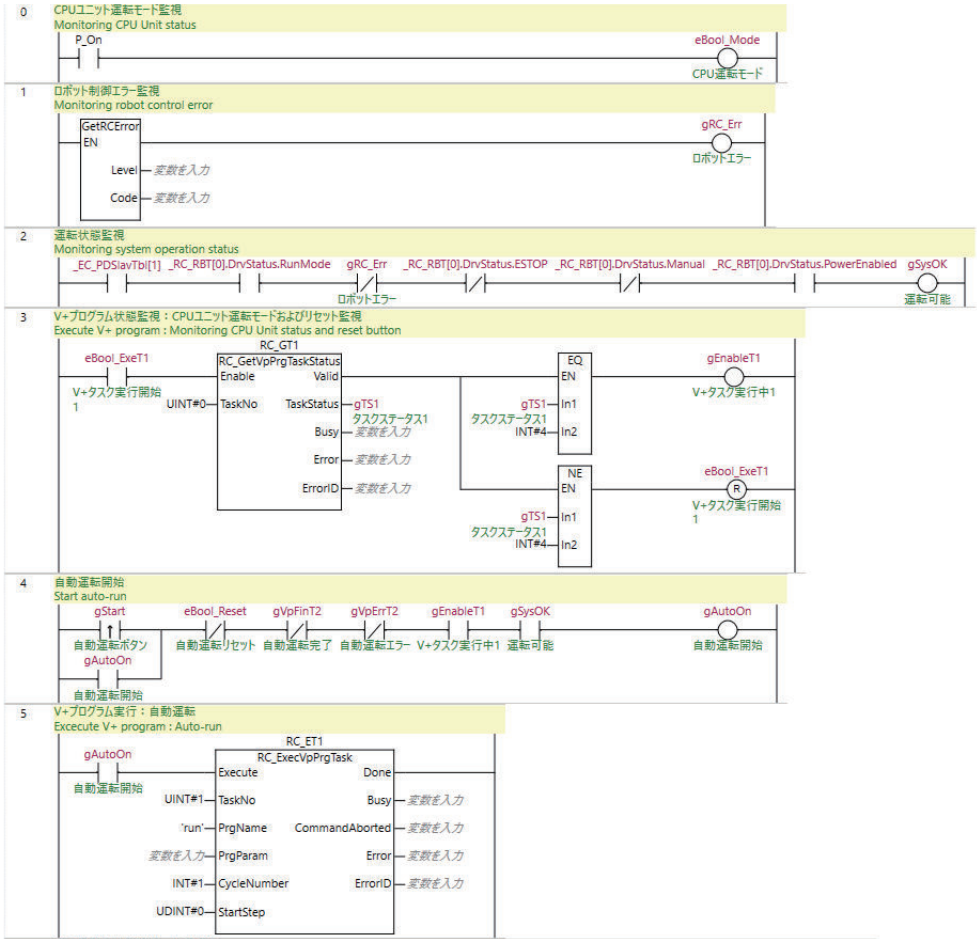
名称	データ型	初期値	割付先	保持	コンスタント	ネットワーク公開	コメント
E001_In_Bit00	BOOL			□	□	非公開	
E001_In_Bit01	BOOL		ECAT://node#1/Read input 1st word/In Bit01	□	□	非公開	
E001_In_Bit02	BOOL		ECAT://node#1/Read input 1st word/In Bit02	□	□	非公開	
E001_In_Bit03	BOOL		ECAT://node#1/Read input 1st word/In Bit03	□	□	非公開	
E001_In_Bit04	BOOL		ECAT://node#1/Read input 1st word/In Bit04	□	□	非公開	
E001_In_Bit05	BOOL		ECAT://node#1/Read input 1st word/In Bit05	□	□	非公開	
E001_In_Bit06	BOOL		ECAT://node#1/Read input 1st word/In Bit06	□	□	非公開	
E001_In_Bit07	BOOL		ECAT://node#1/Read input 1st word/In Bit07	□	□	非公開	
E001_In_Bit08	BOOL		ECAT://node#1/Read input 1st word/In Bit08	□	□	非公開	
E001_In_Bit09	BOOL		ECAT://node#1/Read input 1st word/In Bit09	□	□	非公開	
E001_In_Bit10	BOOL		ECAT://node#1/Read input 1st word/In Bit10	□	□	非公開	
E001_In_Bit11	BOOL		ECAT://node#1/Read input 1st word/In Bit11	□	□	非公開	
E001_In_Bit12	BOOL		ECAT://node#1/Read input 1st word/In Bit12	□	□	非公開	
E001_In_Bit13	BOOL		ECAT://node#1/Read input 1st word/In Bit13	□	□	非公開	
E001_In_Bit14	BOOL		ECAT://node#1/Read input 1st word/In Bit14	□	□	非公開	
E001_In_Bit15	BOOL		ECAT://node#1/Read input 1st word/In Bit15	□	□	非公開	
E001_Minor_Fault	BOOL		ECAT://node#1/Sysmac Error Status/Minor Fault	□	□	非公開	
E001_Observation	BOOL		ECAT://node#1/Sysmac Error Status/Observation	□	□	非公開	
E001_Out_Bit00	BOOL			□	□	非公開	
E001_Out_Bit01	BOOL		ECAT://node#1/Write output 1st word/Out Bit01	□	□	非公開	
E001_Out_Bit02	BOOL		ECAT://node#1/Write output 1st word/Out Bit02	□	□	非公開	
E001_Out_Bit03	BOOL		ECAT://node#1/Write output 1st word/Out Bit03	□	□	非公開	
E001_Out_Bit04	BOOL		ECAT://node#1/Write output 1st word/Out Bit04	□	□	非公開	
E001_Out_Bit05	BOOL		ECAT://node#1/Write output 1st word/Out Bit05	□	□	非公開	
E001_Out_Bit06	BOOL		ECAT://node#1/Write output 1st word/Out Bit06	□	□	非公開	
E001_Out_Bit07	BOOL		ECAT://node#1/Write output 1st word/Out Bit07	□	□	非公開	
E001_Out_Bit08	BOOL		ECAT://node#1/Write output 1st word/Out Bit08	□	□	非公開	
E001_Out_Bit09	BOOL		ECAT://node#1/Write output 1st word/Out Bit09	□	□	非公開	
E001_Out_Bit10	BOOL		ECAT://node#1/Write output 1st word/Out Bit10	□	□	非公開	
E001_Out_Bit11	BOOL		ECAT://node#1/Write output 1st word/Out Bit11	□	□	非公開	
E001_Out_Bit12	BOOL		ECAT://node#1/Write output 1st word/Out Bit12	□	□	非公開	
E001_Out_Bit13	BOOL		ECAT://node#1/Write output 1st word/Out Bit13	□	□	非公開	
E001_Out_Bit14	BOOL		ECAT://node#1/Write output 1st word/Out Bit14	□	□	非公開	
E001_Out_Bit15	BOOL		ECAT://node#1/Write output 1st word/Out Bit15	□	□	非公開	
E001_Read_input_1st_word	WORD		ECAT://node#1/Read input 1st word	□	□	非公開	
E001_Sysmac_Error_Status	BYTE		ECAT://node#1/Sysmac Error Status	□	□	非公開	
E001_Write_output_1st_word	WORD		ECAT://node#1/Write output 1st word	□	□	非公開	
eBool_ExeT1	BOOL			□	□	非公開	V+タスク実行開始1
eBool_ExeT2	BOOL			□	□	非公開	V+タスク実行開始2
eBool_Mode	BOOL			□	□	非公開	CPU運転モード
eBool_Reset	BOOL			□	□	非公開	自動運転リセット
gAutoOn	BOOL			□	□	非公開	自動運転開始
gEnableT1	BOOL			□	□	非公開	V+タスク実行中1
gRC_Err	BOOL			□	□	非公開	ロボットエラー
gStart	BOOL		ECAT://node#1/Read input 1st word/In Bit00	□	□	非公開	自動運転ボタン
gSysOK	BOOL			□	□	非公開	運転可能
gTS1	INT			□	□	非公開	タスクステータス1
gTS2	INT			□	□	非公開	タスクステータス2
gVpErrT2	BOOL			□	□	非公開	自動運転エラー
gVpFinT2	BOOL			□	□	非公開	自動運転完了
RC_Robot001	_sRC_RBT_REF		RC://_RC_RBT[0]	□	☑	非公開	
RC_Robot001_IO	_sRC_RBT_IO_REF		RC://_RC_RBT_IO[0]	□	□	非公開	

● メインプログラム

メインプログラム“Main”では、以下のロボットおよび EtherCAT デジタル I/O ターミナルの状態を監視し、自動運転の開始可否を判断します。
各変数の詳細は、各機器のマニュアルを参照してください。

No.	変数名	説明
1	_EC_PDSlavTbl[1]	EtherCAT デジタル I/O ターミナルの通信が確立しているか確認します。
2	_RC_RBT[0].DrvStatus.RunMode	ロボット統合 CPU ユニットとロボットの通信が確立しているか確認します。
3	GetRCError	ロボット制御機能モジュールに起因するエラーが発生しているか確認します。
4	_RC_RBT[0].DrvStatus.ESTOP	ロボットが ESTOP 状態か確認します。
5	_RC_RBT[0].DrvStatus.Manual	ロボットの動作モードがマニュアルモードか確認します。
6	_RC_RBT[0].DrvStatus.PowerEnabled	ロボットのハイパワーが有効か確認します。

以下にプログラムを記載します。



V+プログラム

静的ピック & プレース装置を制御するための V+プログラムについて説明します。

V+プログラムでは、グローバル変数を登録し、ロボット指令発行プログラム“run”、V+プログラム停止プログラム“run.checkmode”、および読み出しプログラム“auto”の３つのプログラムを作成します。

● グローバル変数

V+プログラムで使用するグローバル変数を登録します。

名前	種類	値	ロボット	表示モード	カテゴリ	説明
gl.pick	ロケーション	400.000 50.000 211.000 0.000 180.000 -180.000	R1Cobra600	表示しない		
gl.place	ロケーション	-285.000 500.000 211.000 0.000 180.000 180.000	R1Cobra600	表示しない		
gl.wait	ロケーション	400.000 0.000 330.000 0.000 180.000 -150.000	R1Cobra600	表示しない		

● ロボット指令発行プログラム(run)

ロボット指令発行プログラム“run”では、静的ピック & プレース動作を実行するためのロボット指令を発行します。
以下にプログラムを記載します。

```
.PROGRAM run()
; ABSTRACT:  Pick and place program
        AUTO ar.speed, ar.accel, ar.wait_on, ar.wait_off, ar.dist, ar.loc[5], al.safe
        EXTERNAL ebool_exet2
        ebool_exet2 = TRUE
; Set parameter
        ar.speed = 100 ; Speed ratio per full speed [%]
        ar.accel = 50 ;  Accel ratio per full speed [%]
        ar.wait_on = 0.5 ; Wait time for air-on [sec]
        ar.wait_off = 0.2 ; Wait time for air-off [sec]
        ar.dist = 40 ; Approach and deppart distance [mm]
; Calculate and set temporary destination
        DECOMPOSE ar.loc[] = HERE
        SET al.safe = TRANS(ar.loc[0],ar.loc[1],DZ(gl.wait),ar.loc[3],ar.loc[4],ar.loc[5])
;Prepare robot motion ready
        CALIBRATE
        ATTACH ()
        SPEED ar.speed, ar.speed ALWAYS
        ACCEL ar.accel, ar.accel
; Approach/Move to wait position
        MOVES al.safe
        MOVE gl.wait
        BREAK
; Approach/Move to pick position
        APPRO gl.pick, ar.dist
        MOVES gl.pick
        BREAK
;Suctioning on and waiting
        SIGNAL 4001
        WAIT.EVENT ar.wait_on
;Depart/Approach/MOVE to palce position
        DEPART ar.dist
        APPRO gl.place, ar.dist
        MOVES gl.place
```

```
        BREAK
; Suctioning off and waiting
        SIGNAL -4001
        WAIT.EVENT ar.wait_off
; Depart/Move to wait position
        DEPART ar.dist
        MOVE gl.wait
        BREAK
; Release robot control
        DETACH ()
.END
```

● 状態監視プログラム(supervisor)

V+プログラム停止プログラム“supervisor”では、シーケンス制御プログラムと V+プログラムとの共有変数を使用することで、ロボット統合 CPU ユニットがプログラムモードになったことを検出し、ロボット動作および V+プログラムを停止します。
以下にプログラムを記載します。

```
.PROGRAM supervisor()
; ABSTRACT:  Stop pic and place program
        EXTERNAL ebool_mode
        EXTERNAL ebool_exet1, ebool_reset
        ebool_exet1 = TRUE
        WHILE TRUE DO
; Check NJ mode and reset-button
            IF (ebool_mode == FALSE) OR (ebool_reset == TRUE) THEN
; Abort current robot motion
                BRAKE
; Terminate "run" program task
                ABORT 1
; Wait until "run" program task has stopped
                CYCLE.END 1
; Clear a program execution stack
                KILL 1
            END
            IF (ebool_mode == TRUE) THEN
                ebool_exet1 = TRUE
            END
; Wait next robot control period
            WAIT
        END
.END
```

● 読み出しプログラム

読み出しプログラム“auto”では、V+プログラムおよびグローバル変数を、SD メモリカードからメインメモリに読み出します。
読み出しプログラム“auto”については、「3-3-8 V+のプログラムと変数の電源 ON 時の自動読込を設定する (P.3-70)」を参照してください。

3-2 基本的な立ち上げ手順

静的ピック & プレース装置を実現するための基本的な立ち上げ手順を説明します。
装置の動作をプログラミングとシミュレーションで設定、確認した後に、実機で動作を確認し微調整します。

番号	手順		説明	参照
1	プログラミングとシミュレーションの手順	プロジェクト作成	Sysmac Studio でプロジェクトファイルを新規に作成します。	P.3-16
2		ネットワーク構成作成	EtherCAT ネットワーク構成を作成し、ロボットを追加します。	P.3-17
3		プログラミング	シーケンス制御プログラムおよび V+プログラムを作成します。	P.3-20 P.3-26
4		3D 形状データ配置	3D 形状データを配置し、ロボットの動作範囲を 3D 表示画面上で見えるようにします。	P.3-33
5		シミュレーション開始	シミュレーション機能を有効にし、シミュレーションを開始します。	P.3-59
6		ティーチング（シミュレーション）	Sysmac Studio のシミュレーションを使用し、ティーチングを行います。	P.3-60
7		プログラム実行（シミュレーション）	Sysmac Studio のシミュレーションを使用し、シーケンス制御プログラムを実行します。	P.3-74
8	設置、配線する	ロボット統合 CPU ユニットと EtherCAT デジタル I/O ターミナル間を配線する	ロボット統合 CPU ユニットと EtherCAT デジタル I/O ターミナル間を配線します。	P.3-79
9		EtherCAT デジタル I/O ターミナルの EtherCAT ノードアドレスを設定する	EtherCAT デジタル I/O ターミナルの EtherCAT ノードアドレスを設定します。	P.3-80
10		EtherCAT デジタル I/O ターミナルとロボット間を配線する	EtherCAT デジタル I/O ターミナルとロボット間を配線します。	P.3-80
11		ロボットの EtherCAT ノードアドレスを設定する	ロボットの EtherCAT ノードアドレスを設定します。	P.3-81
12		ロボット統合 CPU ユニットとパソコン間を配線する	ロボット統合 CPU ユニットとパソコン間を配線します。	P.3-81
13		ロボットと T20 ペンダント、フロントパネル間を配線する	ロボットと T20 ペンダント間、およびロボットとフロントパネル間を配線します。	P.3-82
14		EtherCAT デジタル I/O ターミナルとソレノイドバルブ間を配線する	EtherCAT デジタル I/O ターミナルとソレノイドバルブ間を配線します。	P.3-83
15	実機の立ち上げ手順	オンライン接続	ロボット統合 CPU ユニットと Sysmac Studio をオンライン接続します。	P.3-85
16		設定とプログラムの転送	Sysmac Studio のプロジェクトファイル内の設定およびプログラムをロボット統合 CPU ユニットに転送します。	P.3-87
17		ティーチング（実機）	実際にロボットを操作して、ティーチングデータを確認、修正します。	P.3-89
18		プログラム実行（実機）	プログラムに従って、ロボットを自動運転します。	P.3-103

3-3 プログラミングとシミュレーションの手順

プロジェクトファイルの作成、プログラミング、シミュレーションでの設定と動作確認の手順を説明します。
シミュレーションを利用することで、実機がなくても装置の動作を事前に確認することが可能です。

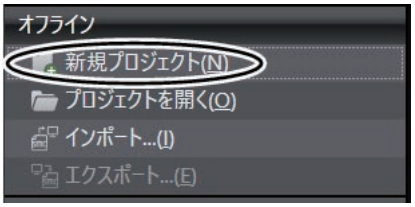
3-3-1 プロジェクトファイルを新規作成する

Sysmac Studio で新規プロジェクトを作成します。

参考

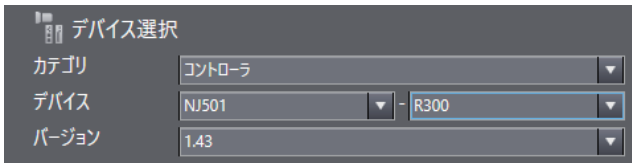
関連動画
- プロジェクトファイルの新規作成
  https://www.fa.omron.co.jp/video/detail/62645190180001/
  Sysmac Studio でロボットを EtherCAT 構成に含む新規プロジェクトを作成する手順を説明しています。

1 **Sysmac Studio を起動し、スタートページで［新規プロジェクト］をクリックします。**



［プロジェクトのプロパティ］画面が表示されます。

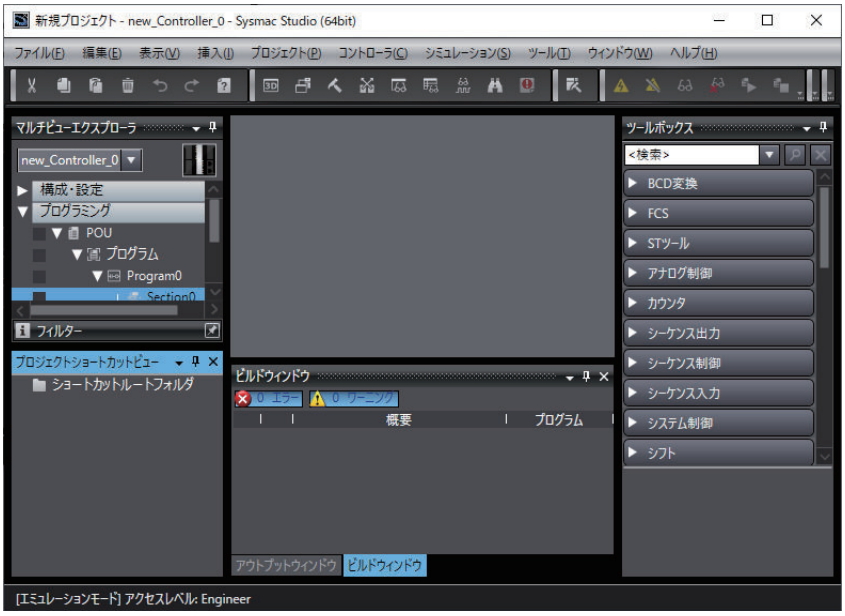
2 **［プロジェクトのプロパティ］画面で、［プロジェクト名］、［作成者］、［コメント］を入力し、デバイス選択の［カテゴリ］、［デバイス］、［バージョン］を以下の画面に示すとおりに選択します。**



3 **［エミュレーションモードで開く］チェックボックスにチェックを入れ、［作成］ボタンをクリックします。**



プロジェクトファイルが作成され、次の画面が表示されます。



3-3-2 EtherCAT ネットワーク構成を作成する

EtherCAT ネットワーク構成を作成し、ロボットを追加します。

参考

関連動画

- プロジェクトファイルの新規作成
  https://www.fa.omron.co.jp/video/detail/6264519018001/
  Sysmac Studio でロボットを EtherCAT 構成に含む新規プロジェクトを作成する手順を説明しています。

1 **マルチビューエクスプローラの［構成・設定］をクリックします。**



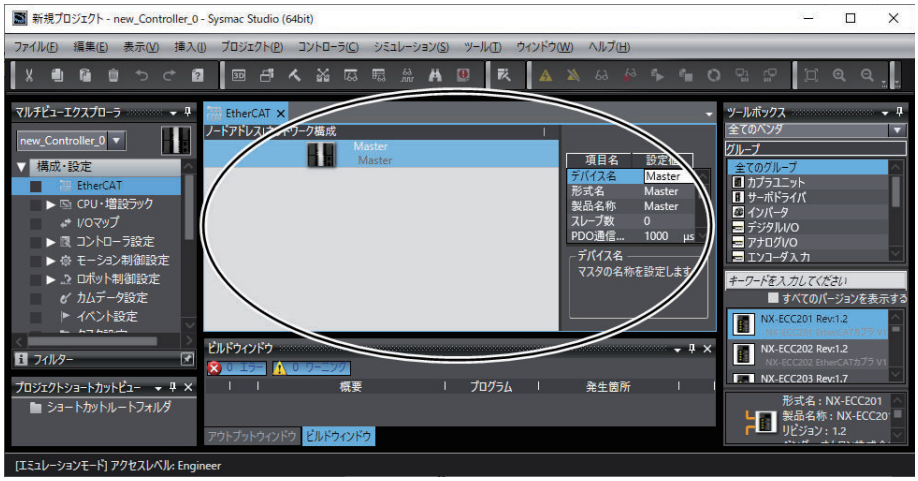
［構成・設定］の下にツリーが展開されます。



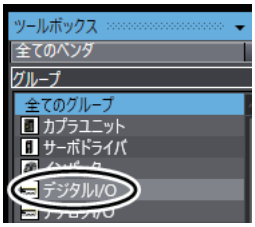
2 **マルチビューエクスプローラから［構成・設定］–［EtherCAT］をダブルクリックします。または［構成・設定］–［EtherCAT］を右クリックして［編集］を選択します。**



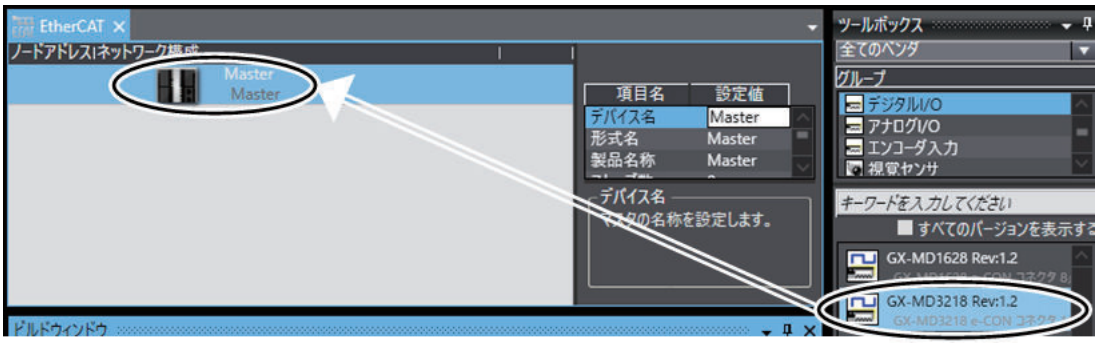
EtherCAT 構成編集画面に EtherCAT マスタが表示されます。



3 **［ツールボックス］のグループで［デジタル I/O］をクリックします。**



4 **使用するデジタル I/O をドラッグし、EtherCAT 構成編集画面のマスタにドロップします。この例では、［GX-MD3218］を使用します。**



マスタの下に［GX-MD3218］が追加されます。



5 **［ツールボックス］のグループで［ロボット］をクリックします。**



6 **使用するロボットをドラッグし、EtherCAT 構成編集画面の［GX-MD3218］にドロップします。この例では、［eCobra 600 Pro］を使用します。**



［GX-MD3218］の下に［eCobra 600 Pro］が追加されます。



3-3-3　シーケンス制御プログラムを作成する

シーケンス制御プログラムのプログラミング手順について説明します。
作成するプログラムについては、「3-1 静的ピック & プレース装置のプログラム仕様 (P.3-2)」を参照してください。

参考

関連動画

- シーケンス制御プログラムの作成
  https://www.fa.omron.co.jp/video/detail/6264519413001/
  Sysmac Studio でシーケンス制御プログラムを作成する手順を説明しています。

デバイス変数を作成する

ソレノイドバルブを制御するためのデバイス変数を作成します。
この設定は静的ピック & プレース装置の動作を、シーケンス制御プログラムで制御する場合に必要となります。
V+プログラムで動作を制御する場合、この設定は必要ありません。

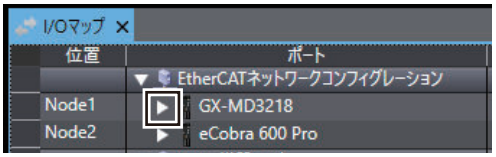
1　マルチビューエクスプローラで［構成・設定］–［I/O マップ］をダブルクリックします。



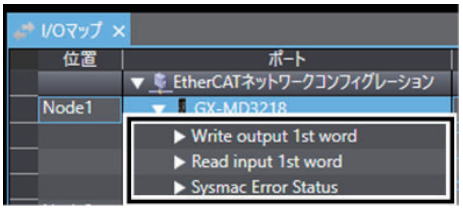
［I/O マップ］タブが表示されます。



2 **［GX-MD3218］の左側の▶アイコンをクリックします。**



［GX-MD3218］の I/O ポートが表示されます。



3 **［GX-MD3218］を右クリックし，［デバイス変数生成］を選択します。**



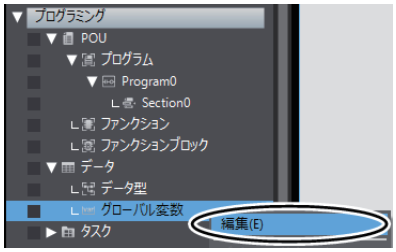
［GX-MD3218］の I/O ポートに対し、デバイス変数が自動的に割り付けられます。



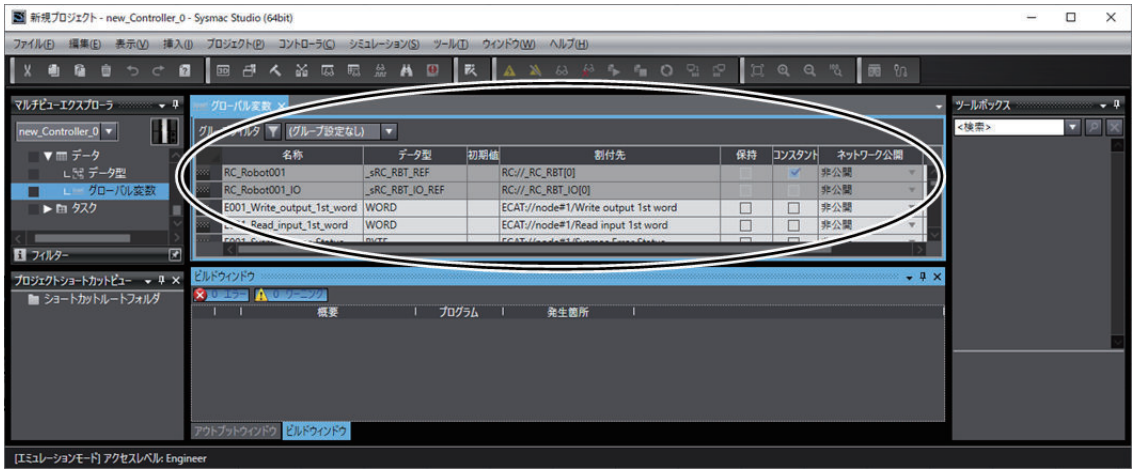
グローバル変数を定義する

シーケンス制御プログラムで位置データとして使用するグローバル変数を定義します。

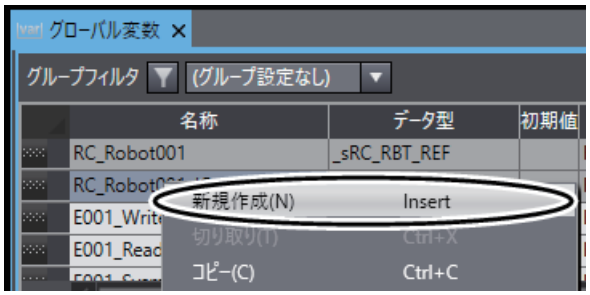
1 **マルチビューエクスプローラで［プログラミング］－［データ］－［グローバル変数］をダブルクリックします。または、［プログラミング］－［データ］－［グローバル変数］を右クリックし、メニューから［編集］を選択します。**



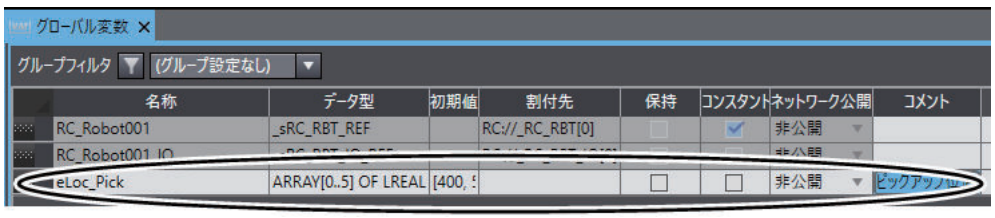
グローバル変数の登録テーブルが表示されます。



2 **グローバル変数テーブル内で［Insert］キーを押すか、または右クリックしてメニューから［新規作成］を選択します。**



3 **各項目を入力または選択し、［Enter］キーを押して確定します。**



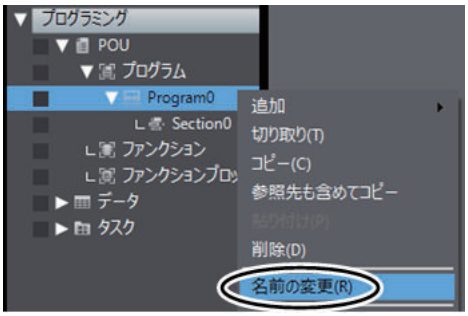
登録するグローバル変数については、シーケンス制御プログラムで動作を制御する場合は「シーケンス制御プログラム (P.3-3)」、V+プログラムで動作を制御する場合は「シーケンス制御プログラム (P.3-10)」を参照してください。

グローバル変数が登録されます。

ラダー図プログラムを作成する

ラダー図プログラムを作成する手順を説明します。

1 **マルチビューエクスプローラの［プログラミング］－［POU］－［プログラム］－［Program0］を右クリックし、［名前の変更］を選択します。**



シーケンス制御プログラム名を編集できる状態になります。



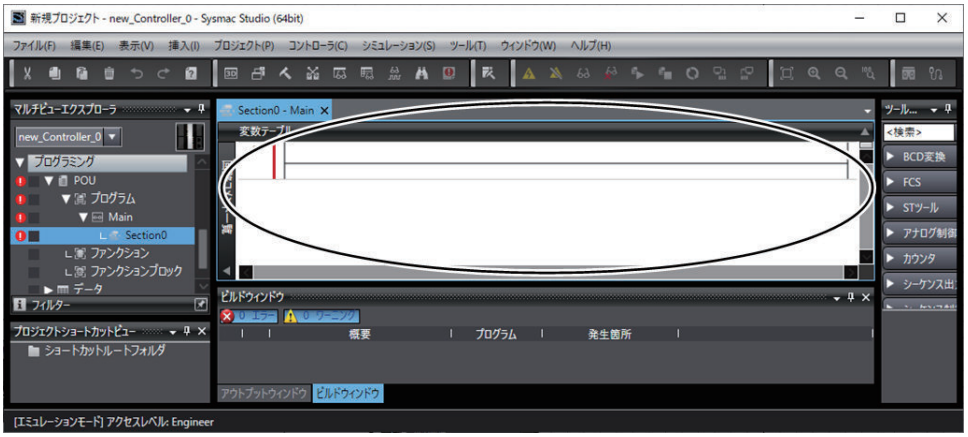
2 **シーケンス制御プログラム名を入力します。**
シーケンス制御プログラム名が変更されます。



3 **編集するセクションをダブルクリックします。**



エディットウィンドウに変数テーブルとラダーエディタが表示されます。



4 **ラダーエディタにプログラムを入力します。内部変数、外部変数はプログラム入力時に自動で登録されます。**

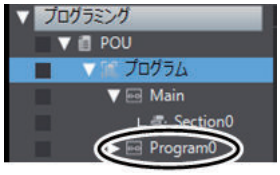
作成するプログラムについては、シーケンス制御プログラムで動作を制御する場合は「シーケンス制御プログラム (P.3-3)」、V+プログラムで動作を制御する場合は「シーケンス制御プログラム (P.3-10)」を参照してください。

シーケンス制御プログラムを複数作成する場合、以降の操作を行います。

5 **マルチビューエクスプローラの［プログラミング］－［POU］－［プログラム］を右クリックし、［追加］－［ラダー］を選択します。**



［プログラム］に［Program0］が追加されます。



6 **マルチビューエクスプローラの［プログラミング］－［POU］－［プログラム］－［Program0］の左に表示されたアイコンをクリックします。**



［Program0］の下に［Section0］が展開されます。



7 **操作 1.に戻り、同様に繰り返します。**

プログラムをタスクに割り付ける

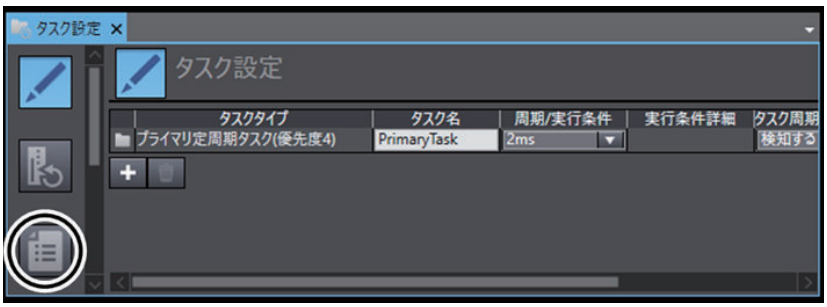
複数のラダー図プログラムをロボット統合 CPU ユニットのタスクに割り付ける手順を説明します。
本ガイドでは、静的ピック & プレース装置をシーケンス制御プログラムで制御する場合に複数のラダー図プログラムを使用しているため、プログラムをタスクに割り付ける操作が必要となります。
V+プログラムで制御する場合は、ひとつのラダー図プログラムだけを使用しているため、この操作は不要です。

1 **マルチビューエクスプローラで、[構成・設定] – [タスク設定] をダブルクリックします。**

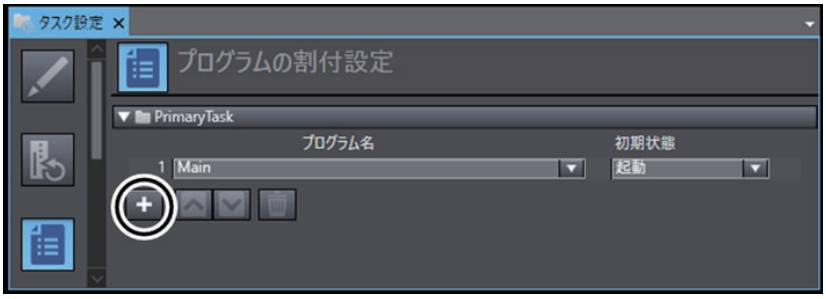


[タスク設定] タブが表示されます。

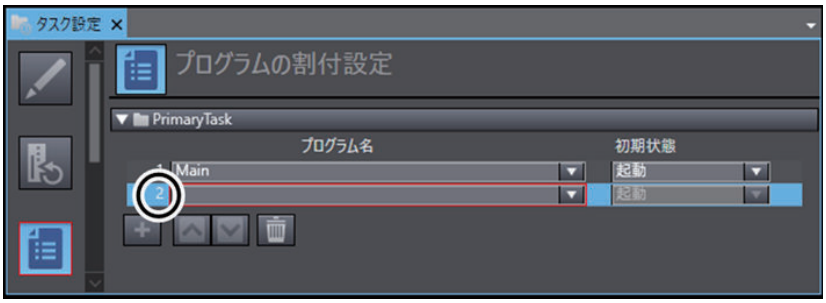
2 **エディットウィンドウに表示される編集画面で [プログラムの割付設定] ボタン（  ）をクリックします。**



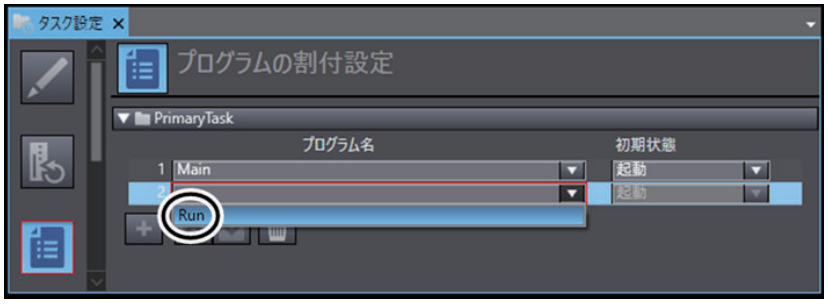
3 **[+] ボタンをクリックします。**



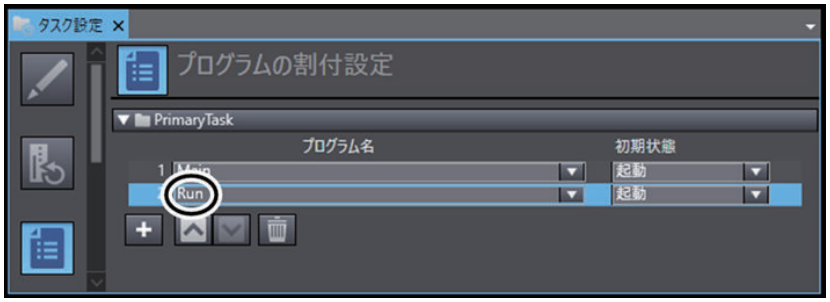
割り付けるプログラムの行が追加されます。



4 **［プログラム名］のプルダウンリストから使用するプログラム名を選択します。**



選択したプログラム名が表示されます。



3-3-4 V+プログラムを作成する

V+プログラムのプログラミング手順について説明します。
作成するプログラムについては、「3-1 静的ピック & プレース装置のプログラム仕様 (P.3-2)」を参照してください。

参考

関連動画
- V+プログラムの作成
  https://www.fa.omron.co.jp/video/detail/6294915156001/
  V+プログラムの作成手順を説明しています。

V+デジタル I/O 割付を設定する

V+プログラムで EtherCAT スレーブを制御するために、V+デジタル I/O 割付設定を行います。
この設定は静的ピック & プレース装置の動作を、V+プログラムで制御する場合に必要となります。
シーケンス制御プログラムで動作を制御する場合、この設定は必要ありません。

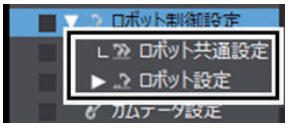
1 **マルチビューエクスプローラのデバイスで、［new_Controller_0］を選択します。**



2 **マルチビューエクスプローラの［ロボット制御設定］の左側の▶アイコンをクリックします。**



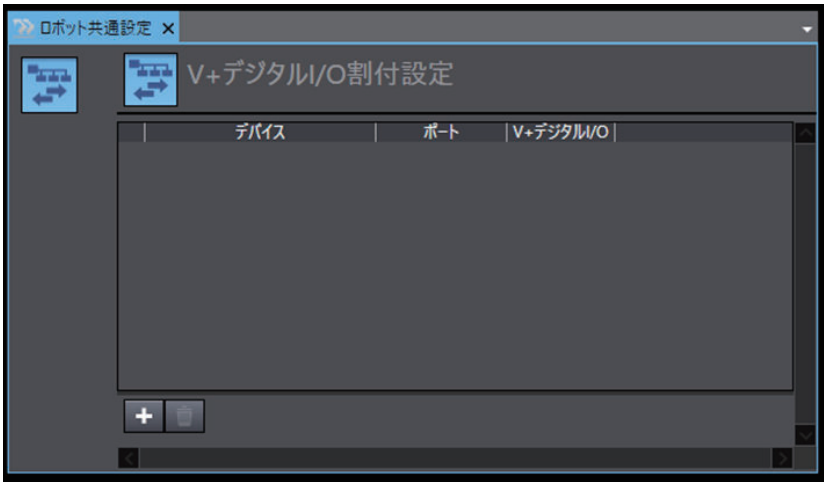
以下のツリーが表示されます。



3 **［ロボット共通設定］をダブルクリックします。**



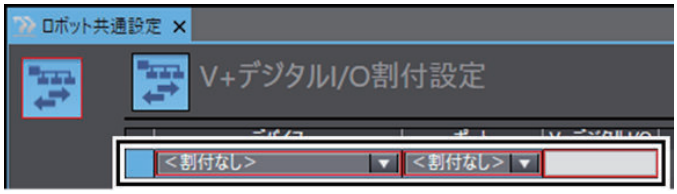
［ロボット共通設定］タブが表示されます。



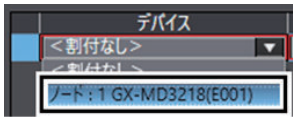
4 **［ロボット共通設定］タブの［+］ボタンをクリックします。**



新規設定のための行が追加されます。



5 **追加された行の［デバイス］欄のドロップダウンリストをクリックし、［ノード : 1 GX-MD3218(E001)］を選択します。**



6 **［ポート］欄のドロップダウンリストをクリックし，［Out Bit00］を選択します。**



7 **［V+デジタル I/O］欄に、「4001」と入力します。**



グローバル変数を定義する

V+のグローバル変数を定義します。
ここでは、「3-1-1 シーケンス制御プログラムで動作を制御する場合 (P.3-3)」の V+プログラムのグローバル変数である「gl.wait」を例に、変数を作成する手順を説明します。

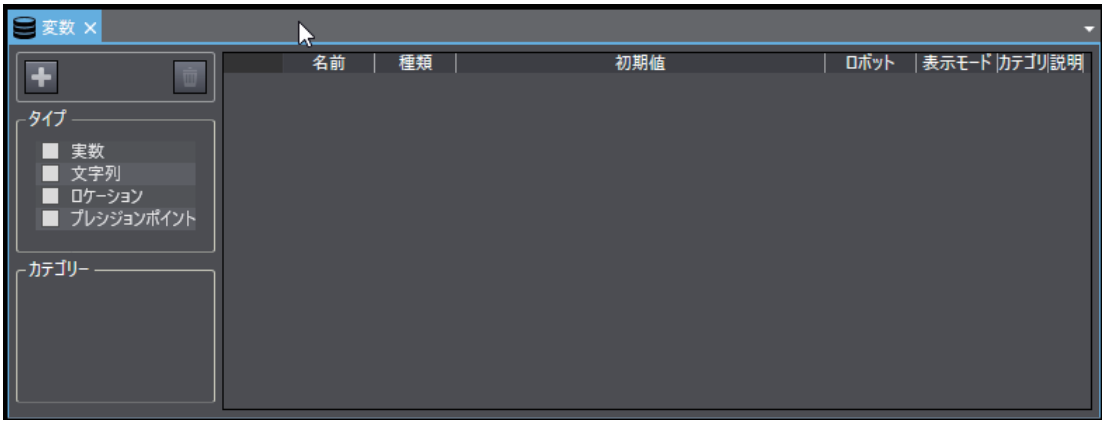
1 **マルチビューエクスプローラのデバイスで［RobotControlSettings］を選択します。**



2 **マルチビューエクスプローラの［プログラミング］–［変数］をダブルクリックします。**



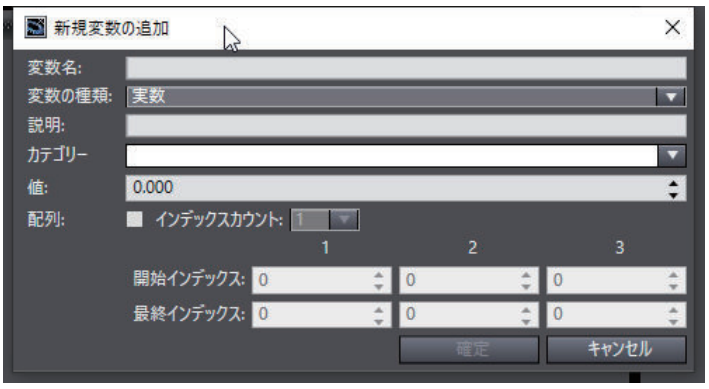
エディットウィンドウに変数編集画面が表示され、変数の作成を行うことができます。



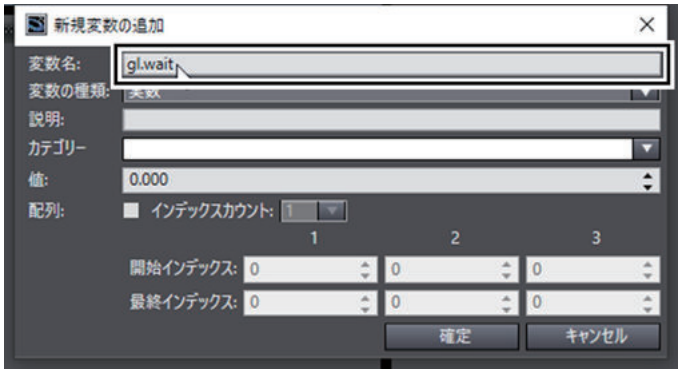
3 **[+] ボタンをクリックします。**



［新規変数の追加］画面が表示されます。

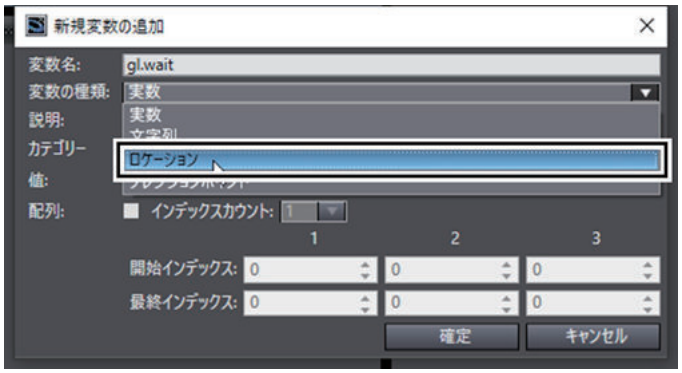


4 **変数名を入力します。**

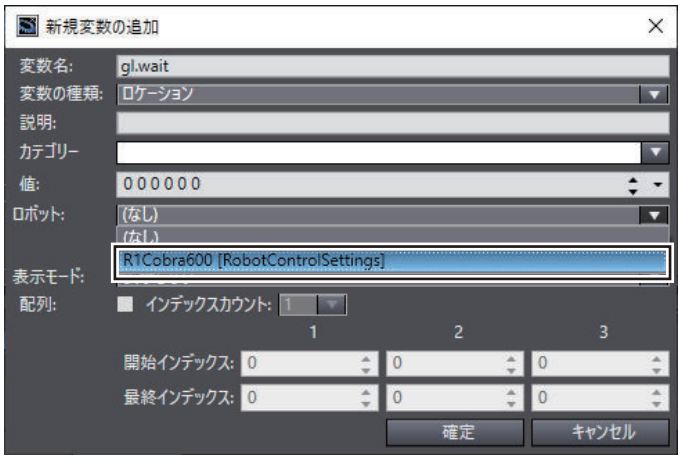


登録するグローバル変数については、シーケンス制御プログラムで動作を制御する場合は「V+プログラム (P.3-8)」、V+プログラムで動作を制御する場合は「V+プログラム (P.3-12)」を参照してください。

5 **変数の種類は、ドロップダウンリストから［ロケーション］を選択します。**



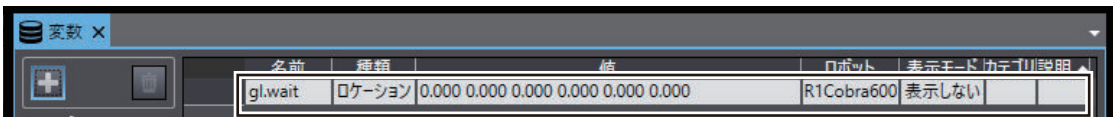
6 **ロボットは、ドロップダウンリストから［R1Cobra600］を選択します。**



7 **［確定］ボタンをクリックします。**



変数編集画面に［gl.wait］が新たに表示されます。



V+プログラムを作成する

V+プログラムを作成する手順を説明します。

1 **マルチビューエクスプローラのデバイスは、［RobotControlSettings］を選択します。**



2 **マルチビューエクスプローラの［プログラミング］－［V+モジュール］を右クリックし、［新規モジュールを追加］をクリックします。**



［V+モジュール］に［program0］－［program0］が追加され、V+プログラム名を編集できる状態になります。



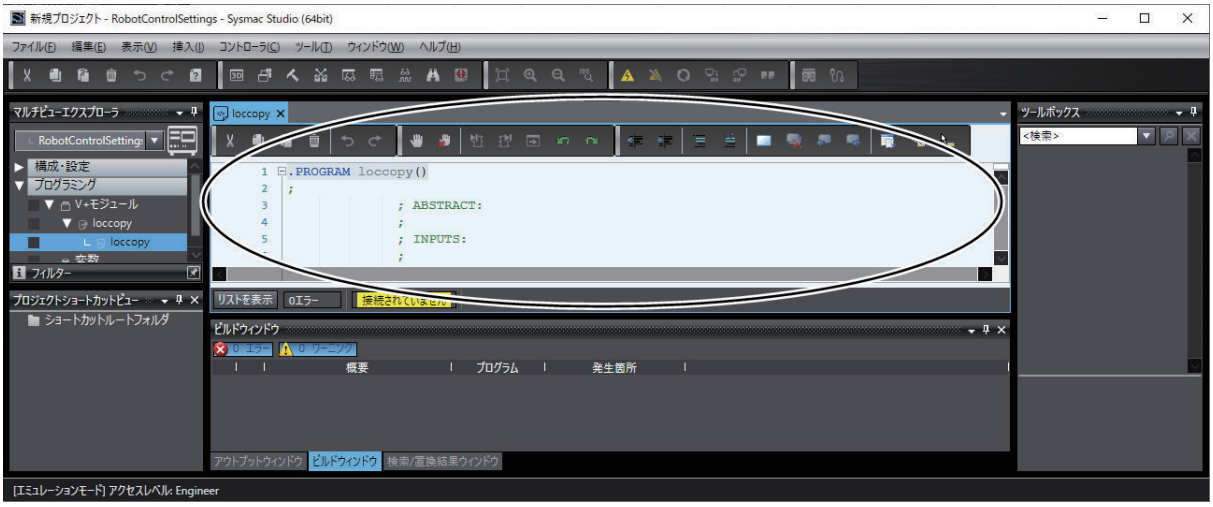
3 **V+プログラム名を入力します。**
V+プログラム名が変更されます。



4 **編集するプログラムをダブルクリックします。**



エディットウィンドウに編集画面が表示され、V+プログラムの作成を行うことができます。



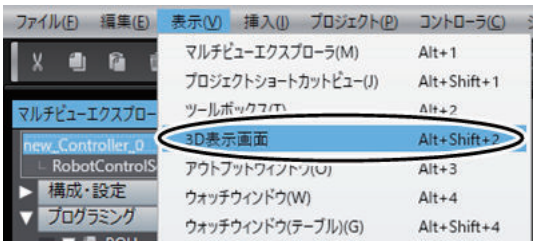
5 **編集画面で、プログラムを入力します。**
作成するプログラムについては、シーケンス制御プログラムで動作を制御する場合は「V+プログラム (P.3-8)」、V+プログラムで動作を制御する場合は「V+プログラム (P.3-12)」を参照してください。

3-3-5　3D 形状データを配置する

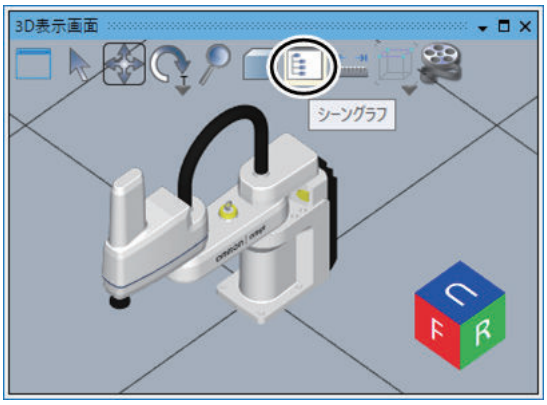
ロボットの動作範囲を表示する

ロボットの動作範囲を 3D 表示画面上で見えるようにします。動作範囲を見えるようにすることで、ロボットの位置の設定を行いやすくなります。

1 **Sysmac Studio のメインウィンドウのメニューバーから［表示］–［3D 表示画面］を選択します。**

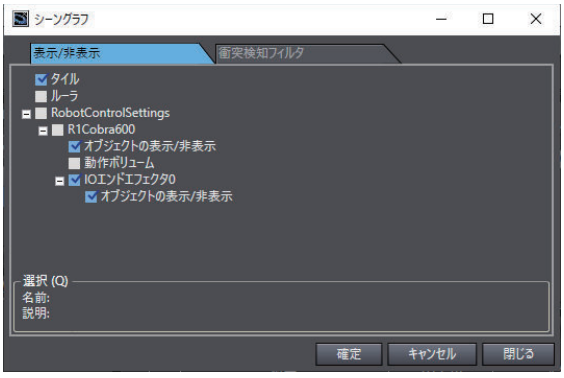


2 **3D 表示画面のシーングラフのアイコンをクリックします。**

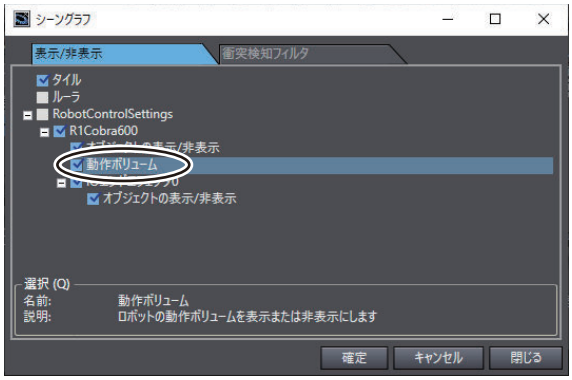


（注）3D 表示画面をフローティング状態にして説明しています。フローティング状態については、「A-2-1 3D 表示画面をフローティング状態にする (P.A-21)」を参照してください。

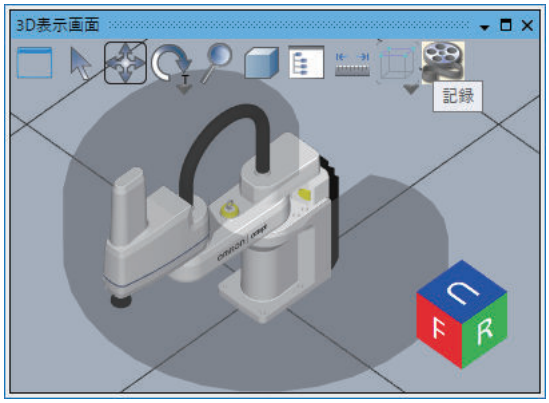
［シーングラフ］画面が表示されます。



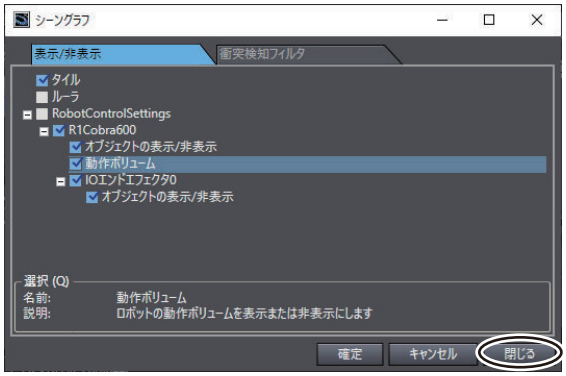
3 **［表示/非表示］タブ内の［RobotControlSettings］–［R1Cobra600］にある［動作ボリューム］のチェックボックスにチェックを入れます。**



3D 表示画面で、ロボットの動作範囲が灰色で表示されます。



4 **［シーングラフ］画面の［閉じる］ボタンをクリックします。**



3D CAD データをインポートする

Sysmac Studio にコンベアの 3D CAD データを 3D 形状データとしてインポートします。
ここでは、「1-4 静的ピック & プレース装置の動作 (P.1-5)」の装置モデルのコンベア A、コンベア B の 3D CAD データをインポートする手順を説明します。

参考

関連動画

- 3D CAD データのインポートと 3D 形状データの配置
  https://www.fa.omron.co.jp/video/detail/6264519020001/
  Sysmac Studio に 3D CAD データをインポートし、3D 表示画面上で適切な位置に配置する手順を説明しています。

1 **Sysmac Studio のメインウィンドウのメニューバーから［挿入］-［アプリケーションマネージャ］を選択します。**



［デバイスの追加］画面が開きます。



2 **［カテゴリ］、［デバイス］、および［バージョン］を以下のように選択し、［OK］ボタンをクリックします。**



マルチビューエクスプローラのウィンドウ上部にあるドロップダウンリストに［ApplicationManager0］が追加されます。



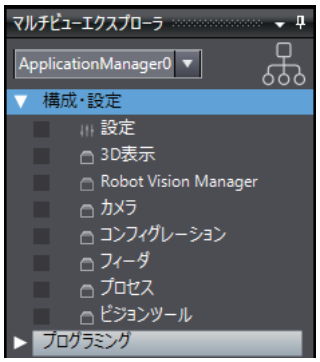
参考

本操作により追加したアプリケーションマネージャはシミュレーションでだけ使用します。実機では使用しません。

3 **マルチビューエクスプローラの［構成・設定］をクリックします。**



［構成・設定］の下にツリーが展開されます。



4 **［3D 表示］を右クリックします。**
コンテキストメニューが表示されます。

5 **コンテキストメニューから、［追加］-［CAD データ］を選択します。**

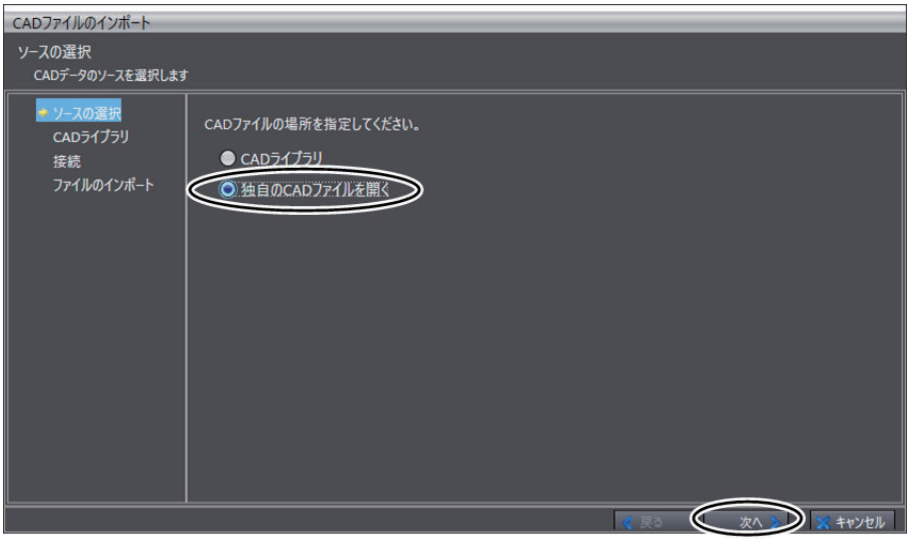


［CAD ファイルのインポート］画面が開きます。

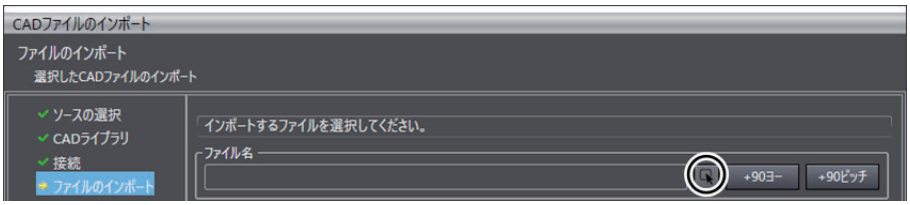


6 **［独自の CAD ファイルを開く］のラジオボタンをクリックします。**

7 **［次へ］ボタンをクリックします。**



8 **［ファイルを開く］ボタンをクリックします。**



［開く］ダイアログが開きます。



9 **対象の 3D CAD データを選択して［開く］ボタンをクリックします。**

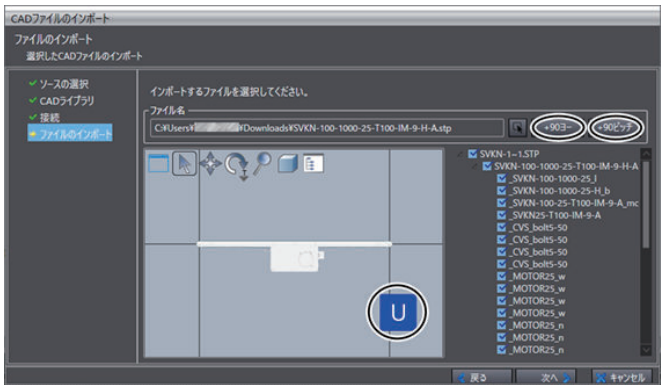
ここでは、コンベア A は MISUMI 社製 SVKN-100-1000-25-T100-IM-9-H-A とします。3D CAD データの拡張子は stp です。



［開く］ダイアログが閉じ、3D CAD データのインポートが始まります。



インポートが完了すると、［CAD ファイルのインポート］画面に、インポートした 3D CAD データのアセンブリ構成が表示されます。

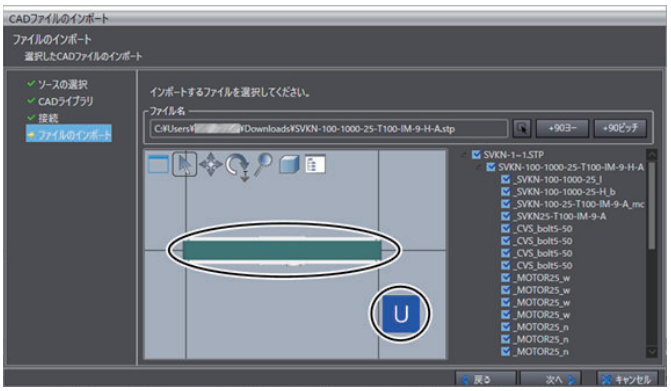


参考

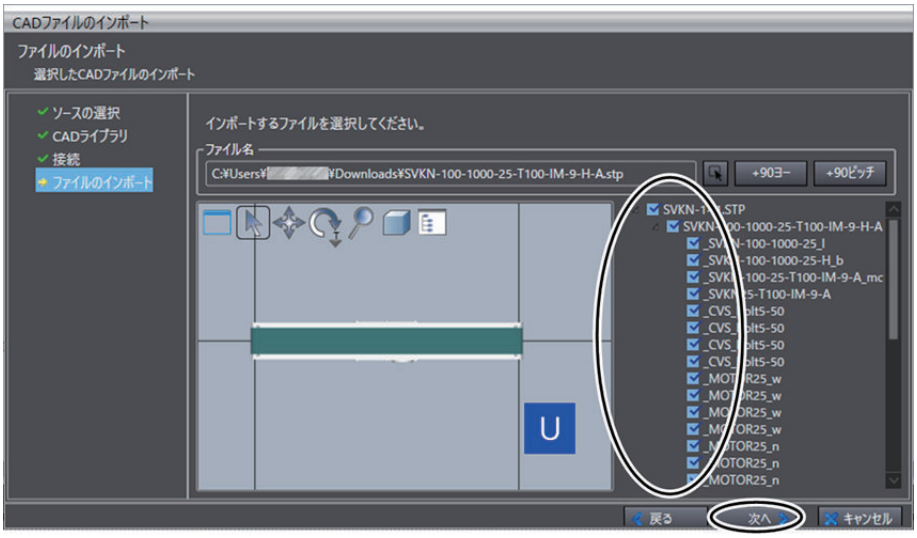
コンベア上面が U 面と一致していないため、一致させる必要があります。コンベアの向きを変えるには、［+90 ヨー］ボタン、および［+90 ピッチ］ボタンを使用します。
あらかじめ位置と向きを調整した CAD データをお使いいただくと、この操作は不要になります。

10 ［+90 ヨー］、［+90 ピッチ］、［+90 ヨー］、［+90 ピッチ］、［+90 ピッチ］の順にボタンをクリックします。

コンベア上面が U 面と一致します。



11 シミュレーションに追加するパーツにチェックを入れ、［次へ］ボタンをクリックします。



［CAD ファイルのインポート］画面が閉じ、メインウィンドウに戻ります。

12 マルチビューエクスプローラで［構成・設定］–［3D 表示］に［CAD データ 0］が追加されていることを確認します。



13 コンベア B についても同様に手順 4 以降を行います。

マルチビューエクスプローラの［構成・設定］–［3D 表示］に［CAD データ 1］が追加されます。



これでコンベア A、コンベア B の 3D CAD データのインポートができました。

3D 形状データを 3D 表示画面上に配置する

インポートした 3D 形状データを 3D 表示画面上で適切な位置に配置します。
3D 形状データを追加した時点では 3D 形状データの座標原点がロボット座標系と一致した状態ですので、3D 形状データを所定の位置に配置する必要があります。
ここでは、「1-4 静的ピック & プレース装置の動作 (P.1-5)」の装置モデルのコンベア A を例に配置する手順を説明します。

参考

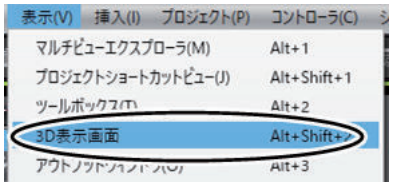
3D 表示画面の調整、および 3D 表示画面での視点を変更する方法については、「A-2 3D 表示画面の操作方法 (P.A-21)」を参照してください。

参考

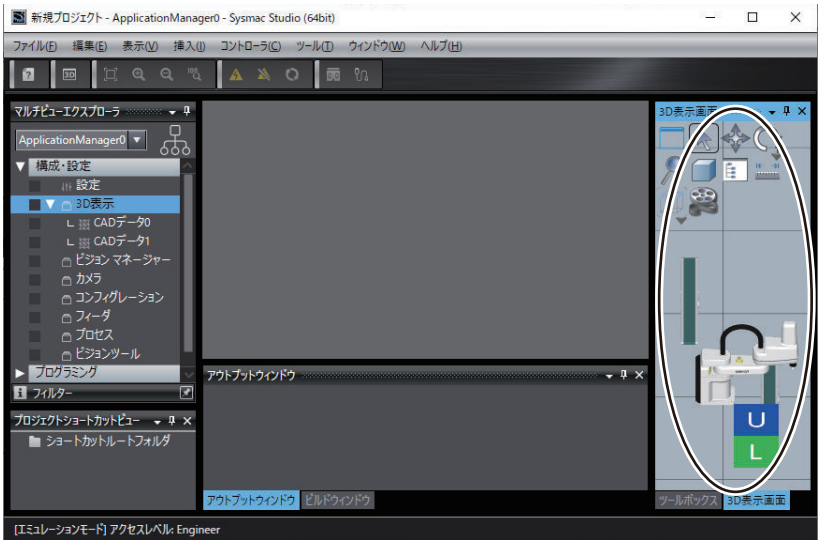
関連動画

- 3D CAD データのインポートと 3D 形状データの配置
  https://www.fa.omron.co.jp/video/detail/6264519020001/
  Sysmac Studio に 3D CAD データをインポートし、3D 表示画面上で適切な位置に配置する手順を説明しています。

1 Sysmac Studio のメインウィンドウのメニューバーから［表示］–［3D 表示画面］を選択します。



メインウィンドウ内の右側に 3D 表示画面が追加されます。

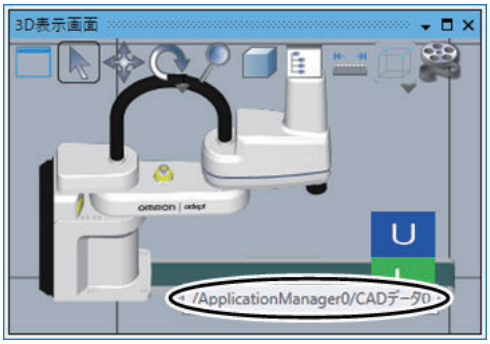


2 **3D 表示画面内で、インポートした 3D 形状データにマウスカーソルを合わせて右クリックします。**

コンテキストメニューが表示されます。

参考

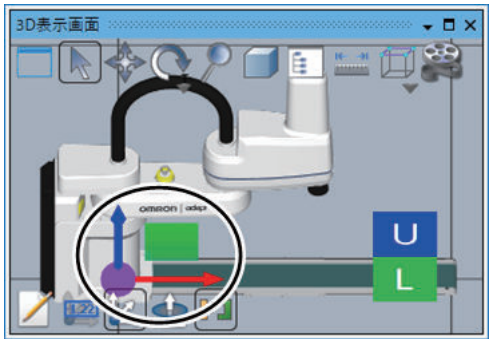
3D 表示画面の 3D 形状データにマウスカーソルを合わせると、3D 形状データの名称［/ApplicationManager0/CAD データ 0］が表示されます。



3 **コンテキストメニューから［ワークスペースの位置の編集］を選択します。**

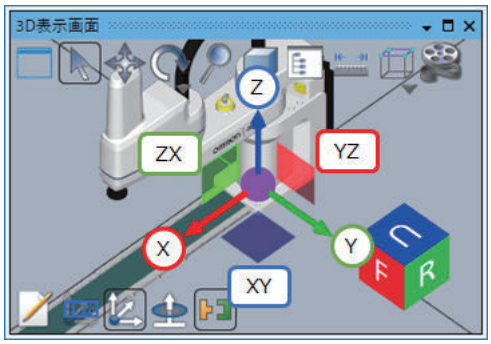


3D 形状データの原点に、色付き矢印と紫の丸、および面からなる移動アイコンが表示されます。

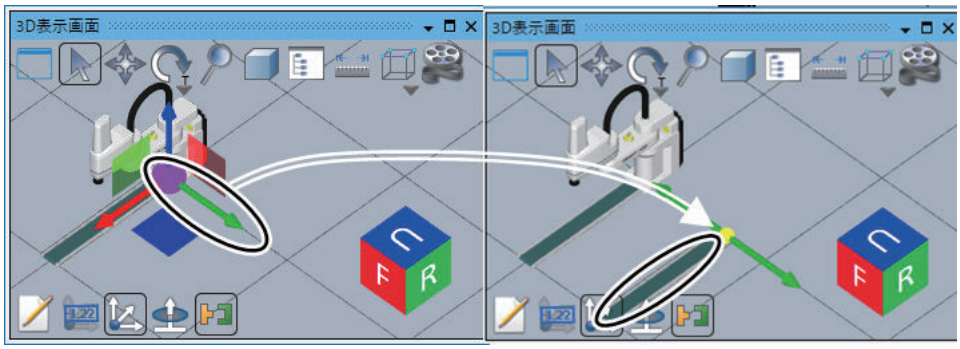


4 **矢印、面または中央の紫の丸をドラッグアンドドロップし、3D 形状データを所定の位置に配置します。**

矢印は、赤が X 軸、緑が Y 軸、青が Z 軸を示しています。また面は、赤が YZ 平面、緑が ZX 平面、青が XY 平面を示しています。

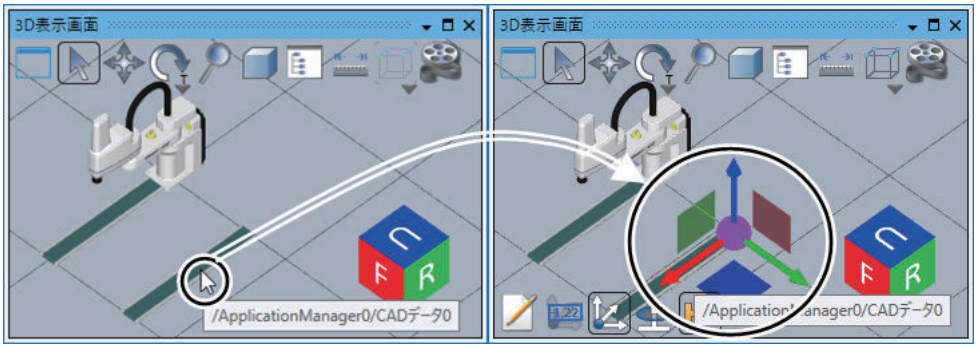


矢印をドラッグすると、対応する軸に沿って 3D 形状データを移動することができます。面をドラッグすると、対応する平面に沿って 3D 形状データを移動することができます。また、紫の丸をドラッグすると、軸方向とは無関係に 3D 形状データを移動することができます。



参考

移動アイコンが表示されなくなったときは、3D 形状データをクリックすれば表示されます。



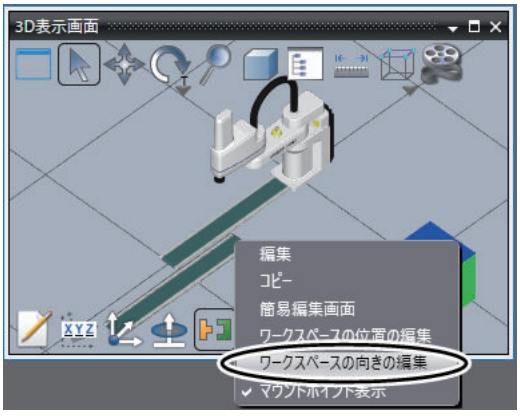
5 **3D 表示画面で、位置を調整していた 3D 形状データを右クリックします。**
コンテキストメニューが表示されます。

6 **コンテキストメニューから［ワークスペースの位置の編集］をクリックしてチェックを外します。**

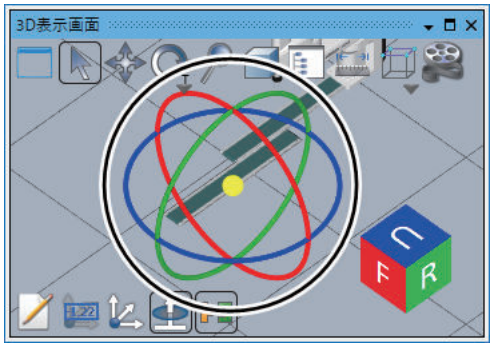


7 **3D 表示画面で 3D 形状データを右クリックします。**
コンテキストメニューが表示されます。

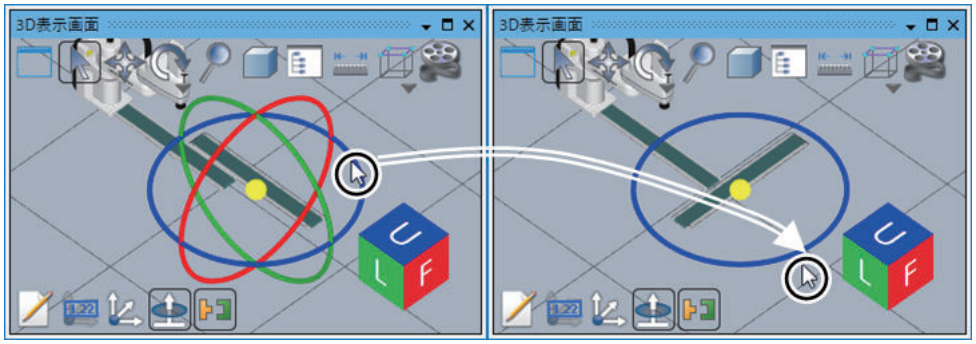
8 **コンテキストメニューから［ワークスペースの向きの編集］を選択します。**



回転アイコンが表示されます。



9 **回転アイコンのハンドルをドラッグアンドドロップし、所定の向きに調整します。**
ハンドルは、赤が X 軸、緑が Y 軸、青が Z 軸を中心とした回転を示しています。



10 **3D 表示画面で、向きを調整していた 3D 形状データを右クリックします。**
コンテキストメニューが表示されます。

11 **コンテキストメニューから［ワークスペースの向きの編集］をクリックしてチェックを外します。**

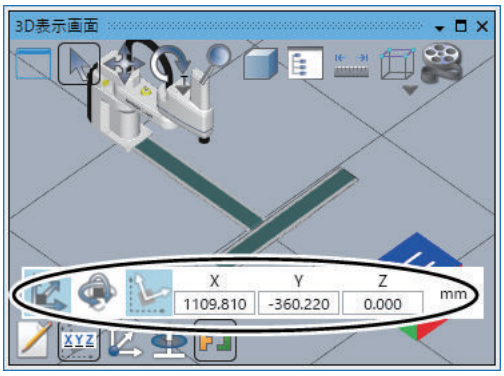


12 **3D 表示画面で、向きを調整していた 3D 形状データを右クリックします。**
コンテキストメニューが表示されます。

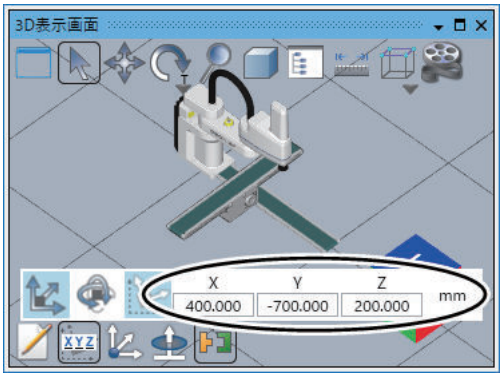
13 **コンテキストメニューから［簡易編集画面］をクリックします。**



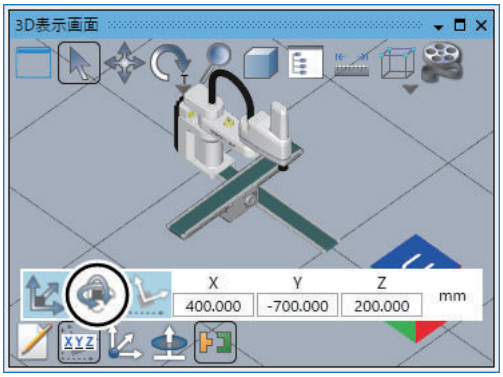
3D 形状データの位置を編集する入力フィールドが表示されます。



14 **3D 形状データが所定の位置になるように、X、Y、Z [mm]の入力フィールドを編集します。**
入力した値に従い、3D 形状データの位置が変化します。

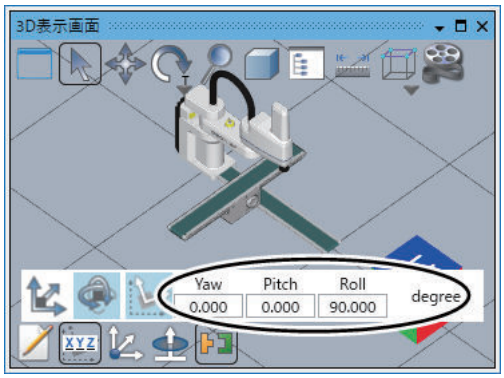


15 **[ワークスペースの向きの編集] アイコンをクリックします。**



3D 形状データの向きを編集する入力フィールドが表示されます。

16 **3D 形状データが所定の向きになるように、Yaw、Pitch、Roll [degree]の入力フィールドを編集します。**
入力した値に従い、3D 形状データの向きが変化します。



17 **3D 表示画面で、位置および向きを調整した 3D 形状データを右クリックします。**
コンテキストメニューが表示されます。

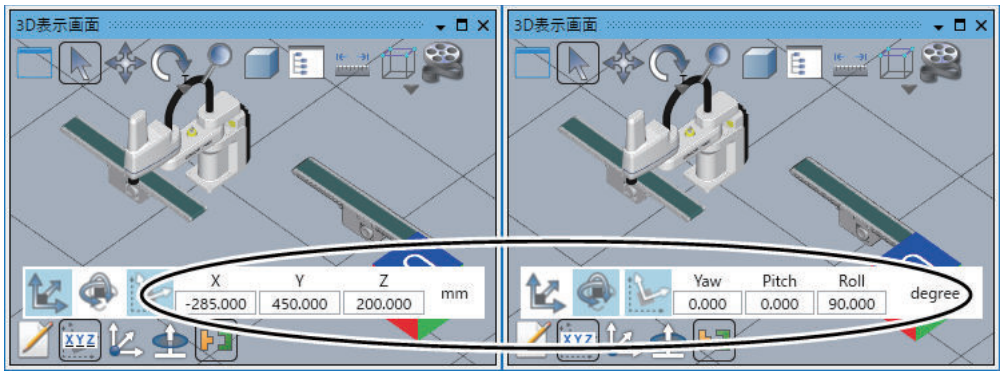
18 **コンテキストメニューから［簡易編集画面］をクリックしてチェックを外します。**



使用上の注意

3D シミュレーションにおける座標系の考え方については、『Sysmac Studio 3D シミュレーション機能 オペレーションマニュアル (SBCA-482)』を参照してください。

19 **コンベア B の CAD データ 1 についても同様に手順 2 以降を行い、位置と向きを設定します。**



これでコンベア A、コンベア B の 3D 形状データの 3D 表示画面上への配置ができました。

ロボットのエンドエフェクタの 3D 情報を設定する

ロボットのエンドエフェクタの 3D 情報を設定します。

参考

関連動画

- エンドエフェクタの 3D 情報の設定
  https://www.fa.omron.co.jp/video/detail/6294913719001/
  ロボットのエンドエフェクタの 3D 情報の設定手順を説明しています。

1 **マルチビューエクスプローラのデバイスは、［RobotControlSettings］を選択します。**



2 **マルチビューエクスプローラの［構成・設定］をクリックします。**



［構成・設定］の下にツリーが展開されます。



3 **［ロボット］をダブルクリックします。**



［ロボット］の下にツリーが展開されます。



4 **［R1Cobra600 ロボット］の左側の▶アイコンをクリックします。**



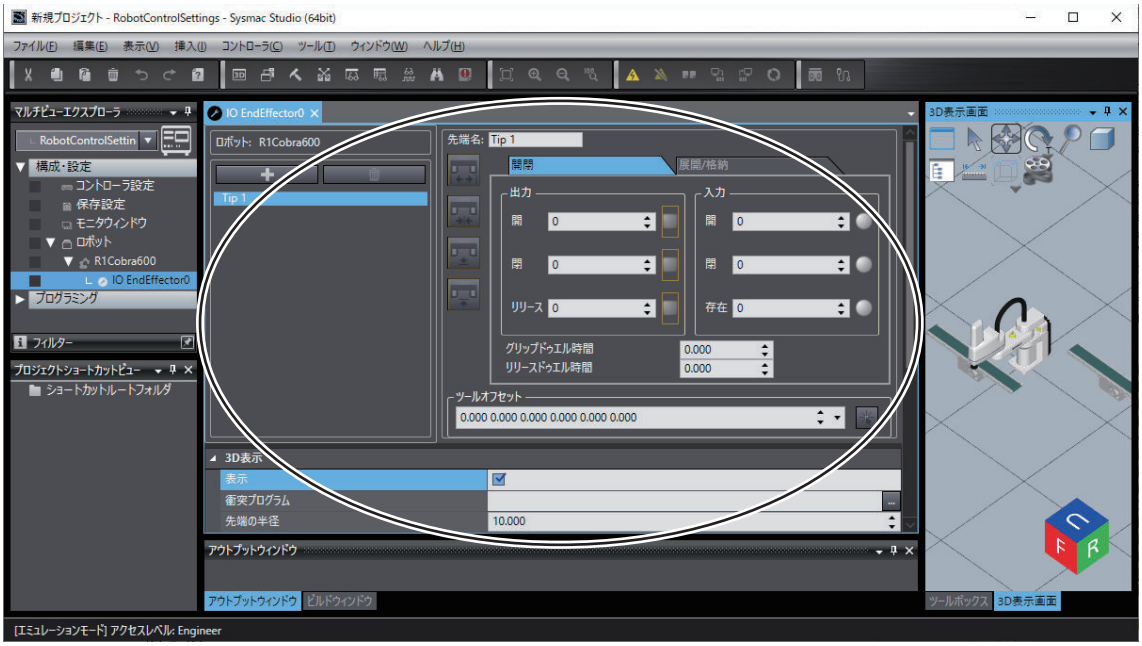
［R1Cobra600 ロボット］の下にツリーが展開されます。



5 **［IO EndEffector0］をダブルクリックします。**

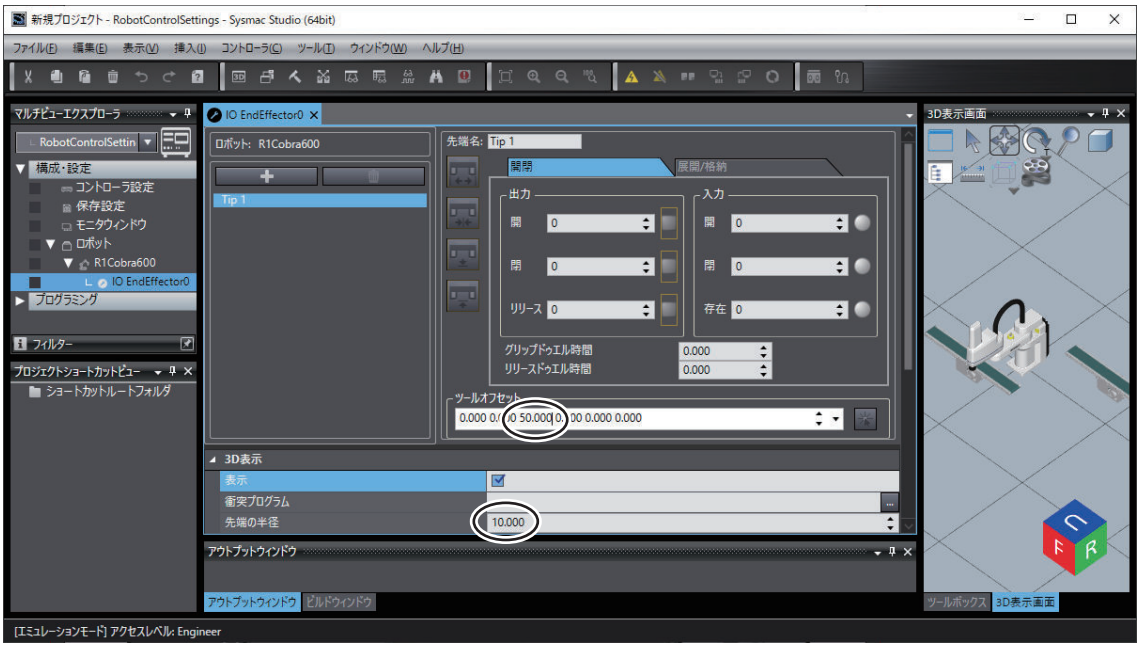


［IO EndEffector0］画面が表示されます。

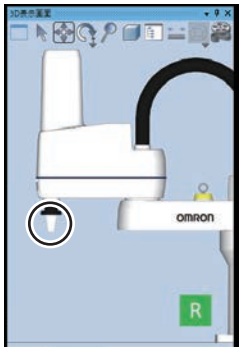


6 **［ツールオフセット］の Z 方向にツールの長さを設定します。**

ここでは 50mm であるとして設定します。
必要により［先端の半径］も設定します。



設定したサイズに応じてエンドエフェクタが 3D 表示画面に表示されます。



衝突検知機能を有効にする

ロボットが、インポートした 3D CAD データなどの他の 3D 形状データと干渉する場合に、シミュレーションで検出できるようにします。ロボットが他の 3D 形状データと干渉する場合には、ロボットや他の 3D 形状データの配置、またはロボットの動きを調整する必要があります。
ここでは、コンベア A、コンベア B とロボットの干渉を検出するための設定手順を説明します。干渉チェックはグループ間で 3D 形状データが干渉するかをチェックするため、ロボットを含むグループ 0 と、コンベア A、コンベア B を含むグループ 1 の 2 つのグループを設定します。

使用上の注意

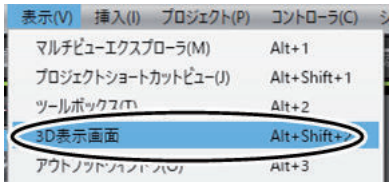
本ガイドではロボットを干渉検出の対象としています。ツール先端に 3D 形状データを配置した場合は、配置した 3D 形状データの干渉検出も必要です。ツール先端に配置した 3D 形状データを干渉検出の対象とする場合の操作方法については、『Sysmac Studio 3D シミュレーション機能 オペレーションマニュアル (SBCA-482)』を参照してください。

参考

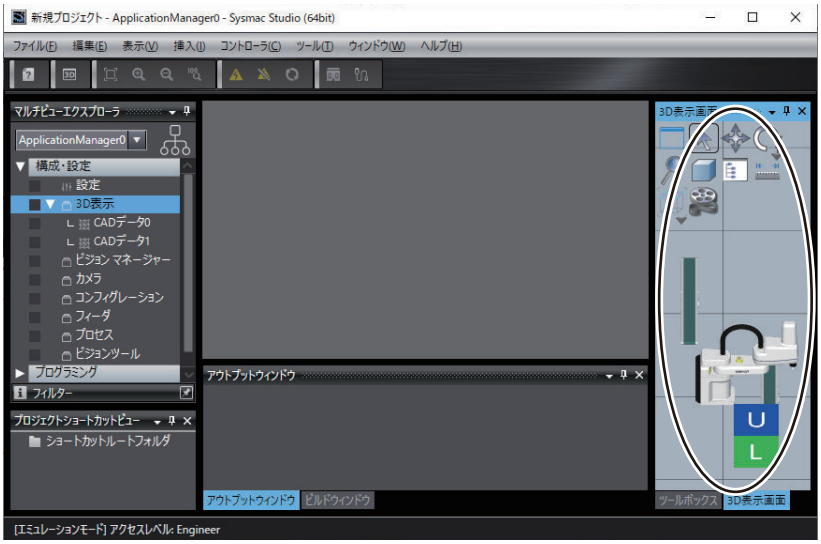
関連動画

- 衝突検知機能の設定
  https://www.fa.omron.co.jp/video/detail/6264519414001/
  Sysmac Studio で、ロボットが他の 3D 形状データと干渉する場合にシミュレーションで検出できるようにする手順を説明しています。

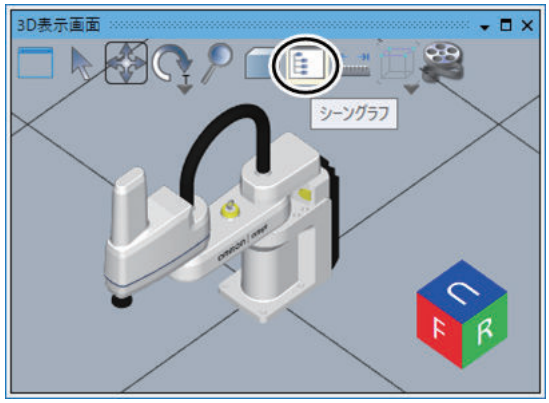
1 **Sysmac Studio のメインウィンドウのメニューバーから［表示］–［3D 表示画面］を選択します。**



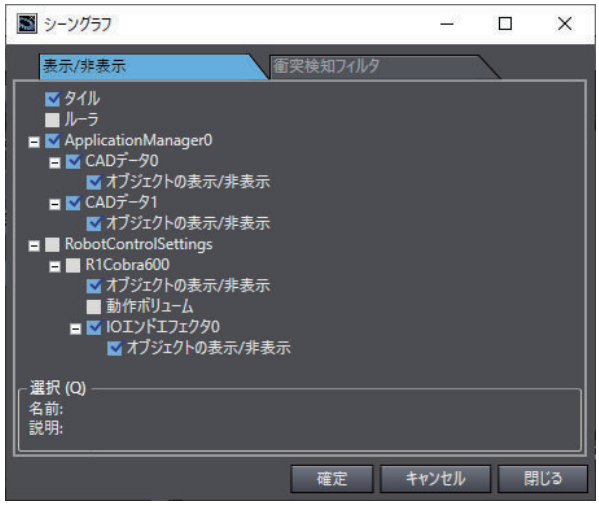
メインウィンドウ内の右側に 3D 表示画面が追加されます。



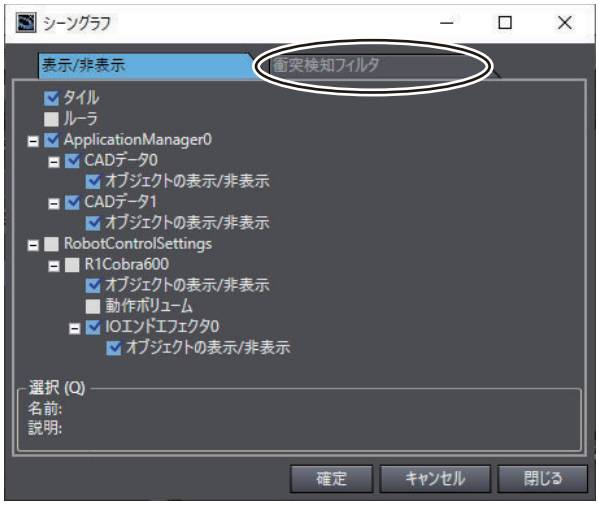
2 **3D 表示画面のアイコン［シーングラフ］をクリックします。**



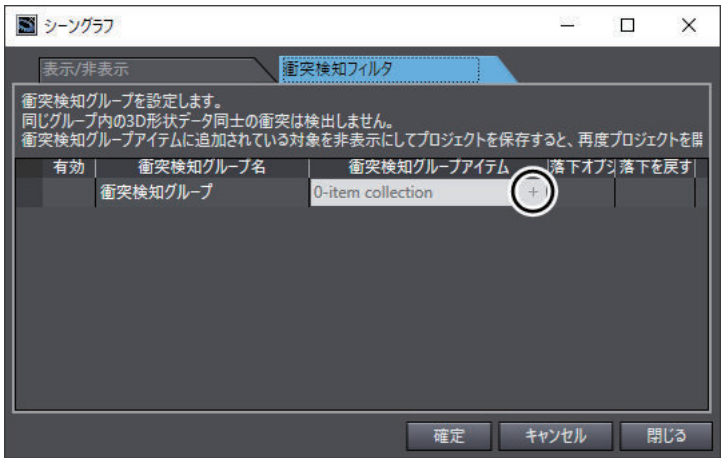
［シーングラフ］画面が開きます。



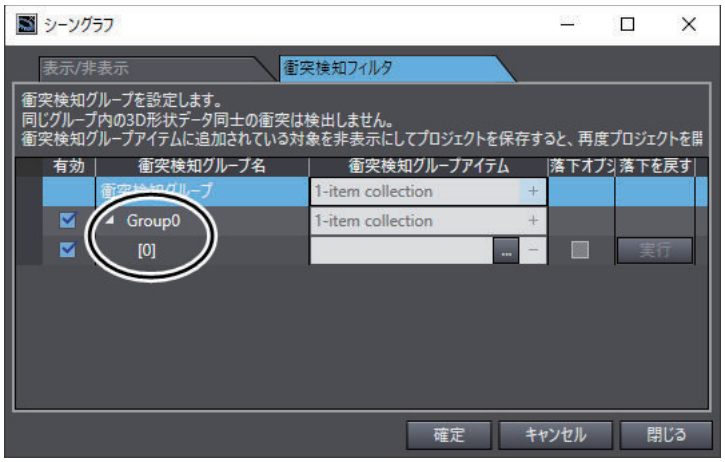
3 **［衝突検知フィルタ］タブをクリックします。**



4 **［衝突検知グループ］の［衝突検知グループアイテム］の［+］マークをクリックします。**

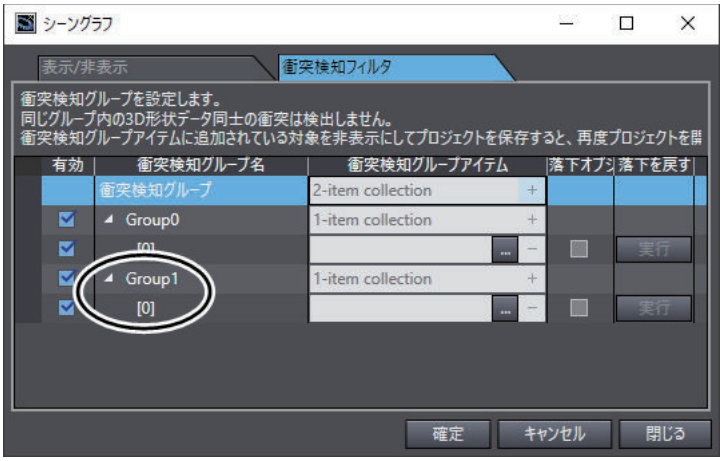


［シーングラフ］画面に、［衝突検知グループ名］が［Group 0］と設定された行が追加されます。

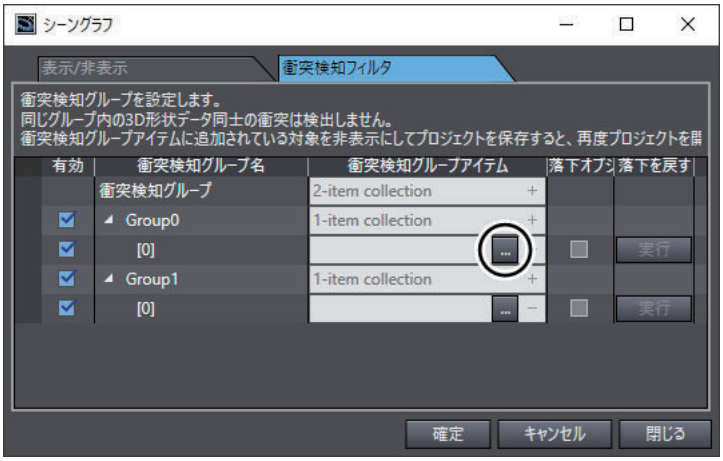


5 **手順 4 をもう一度実施します。**

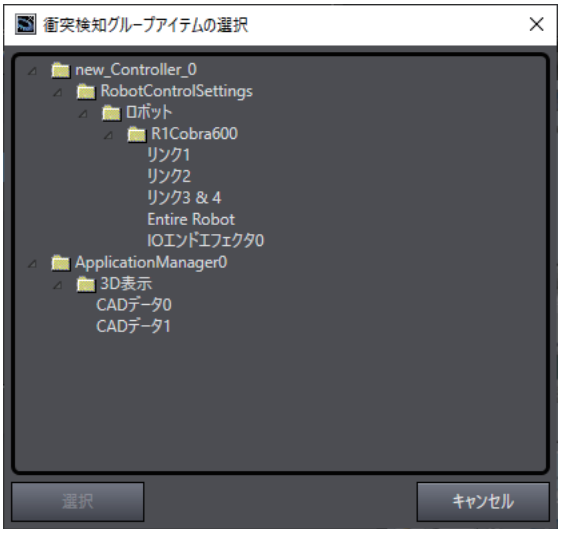
［シーングラフ］画面に、［衝突検知グループ名］が［Group 1］と設定された行が追加されます。



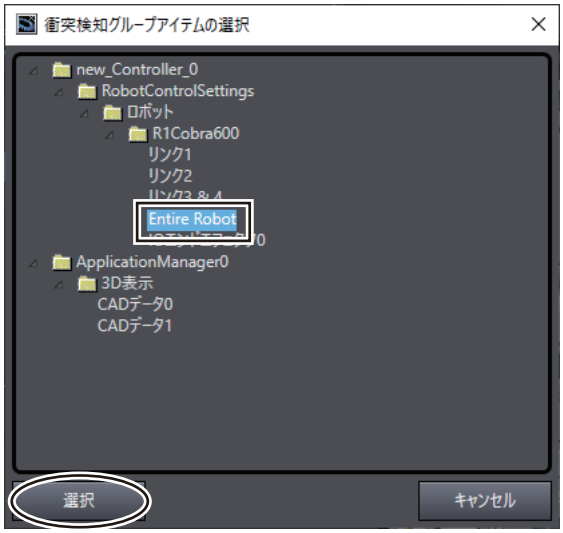
6 **［Group 0］の［[0]］の［衝突検知グループアイテム］の［...］ボタンをクリックします。**



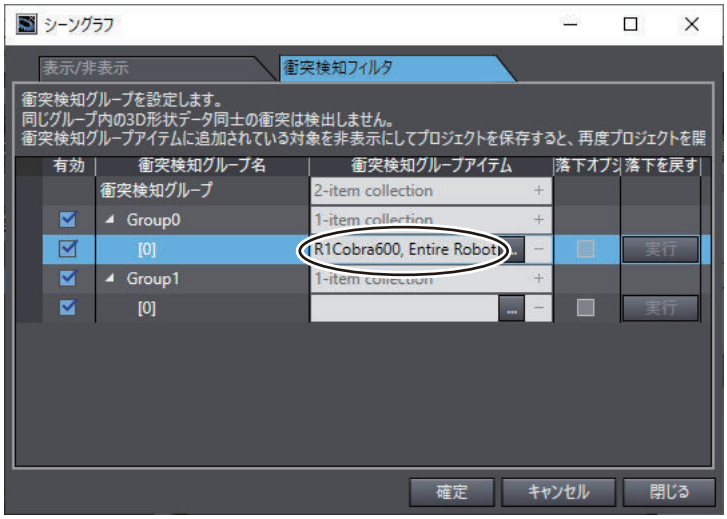
［衝突検知グループアイテムの選択］画面が開きます。



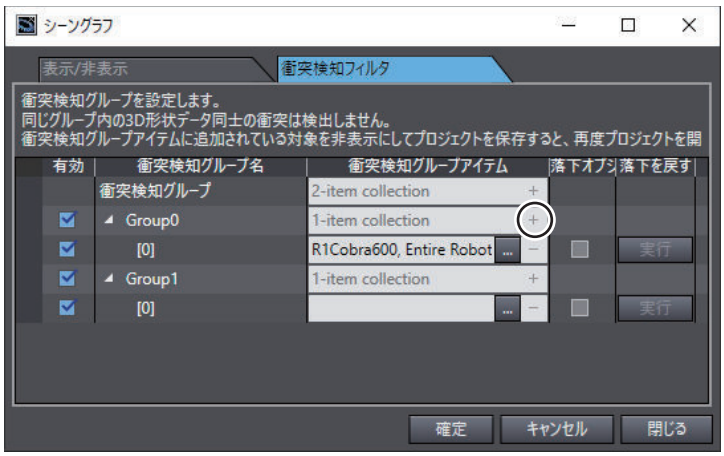
7 **ツリーから［new_Controller_0］–［RobotControlSettings］–［ロボット］–［R1Cobra600］–［Entire Robot］をクリックし、［選択］ボタンをクリックします。**



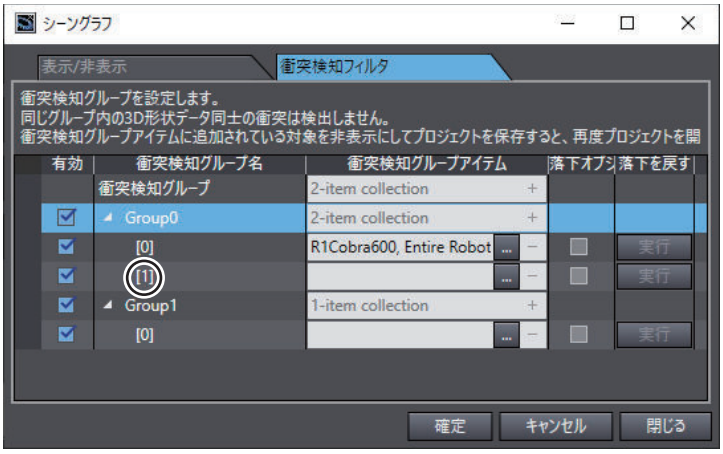
［衝突検知グループアイテムの選択］画面が閉じた後、［シーングラフ］画面に戻ります。［Group 0］－［[0]］の［衝突検知グループアイテム］に［R1Cobra600, Entire Robot］が表示されます。



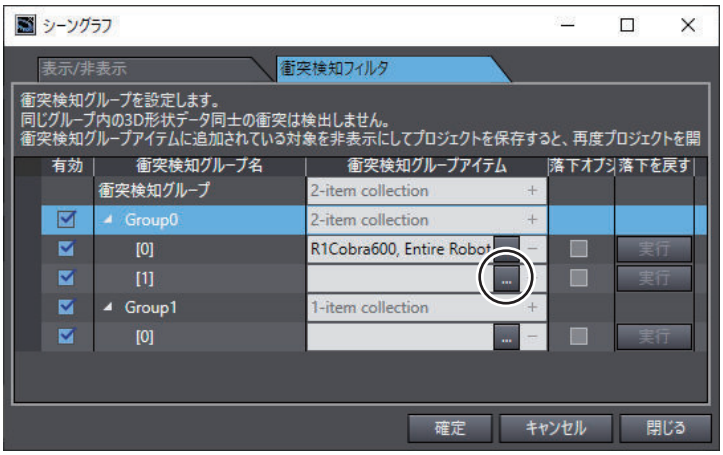
8 ［Group 0］の［衝突検知グループアイテム］の［+］マークをクリックします。



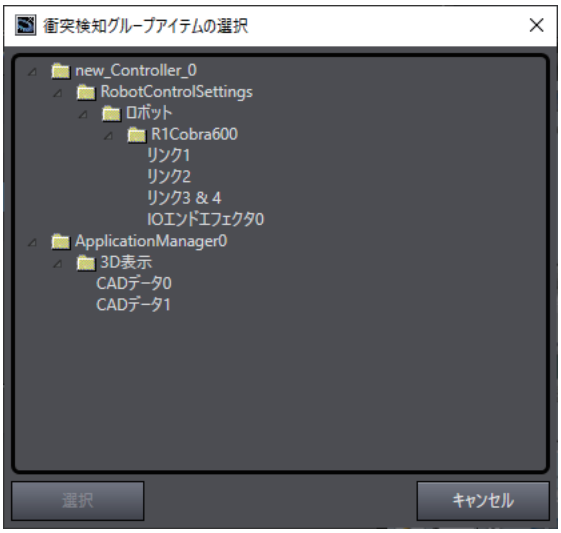
［Group 0］の下に［[1]］が追加されます。



9 **［Group 0］の［[1]］の［衝突検知グループアイテム］の［...］ボタンをクリックします。**



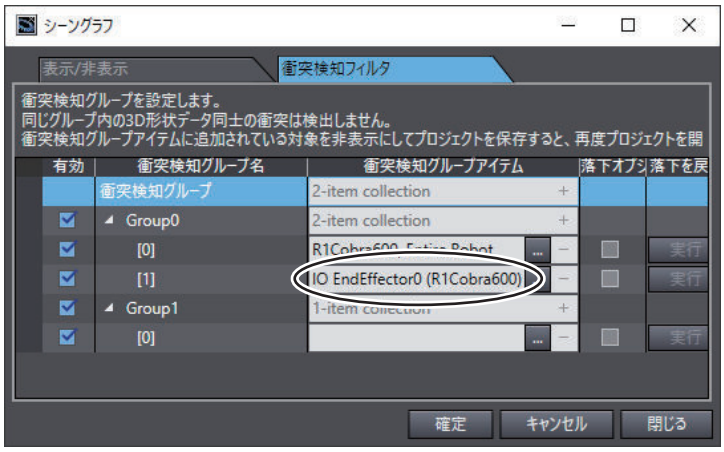
［衝突検知グループアイテムの選択］画面が開きます。



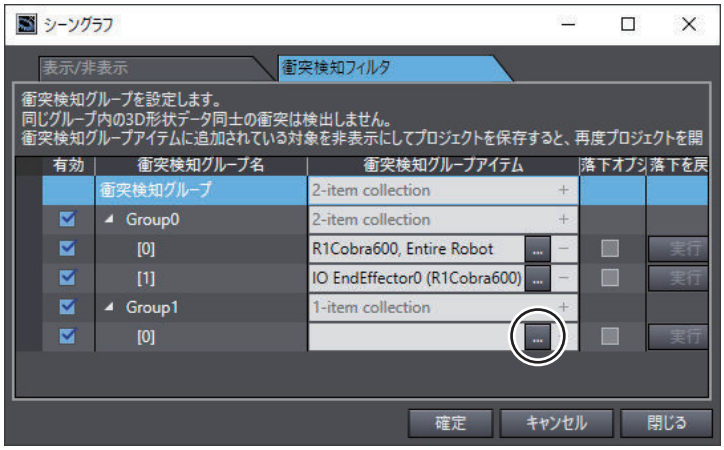
10 **ツリーから［new_Controller_0］－［RobotControlSettings］－［ロボット］－［R1Cobra600］－［IO エンドエフェクタ 0］をクリックし、［選択］ボタンをクリックします。**



［衝突検知グループアイテムの選択］画面が閉じた後、［シーングラフ］画面に戻ります。［Group 0］－［[1]］の［衝突検知グループアイテム］に［IO EndEffector0 (R1Cobra600)］が表示されます。



11 **［衝突検知グループ名］が［Group 1］－［[0]］の［衝突検知グループアイテム］の［...］ボタンをクリックします。**

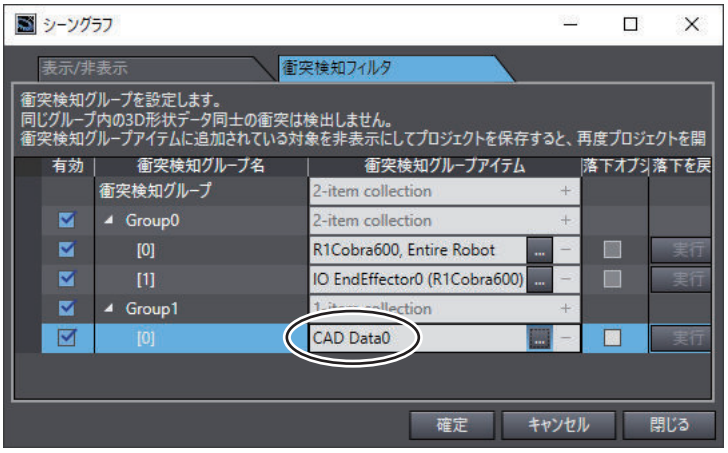


［衝突検知グループアイテムの選択］画面が開きます。

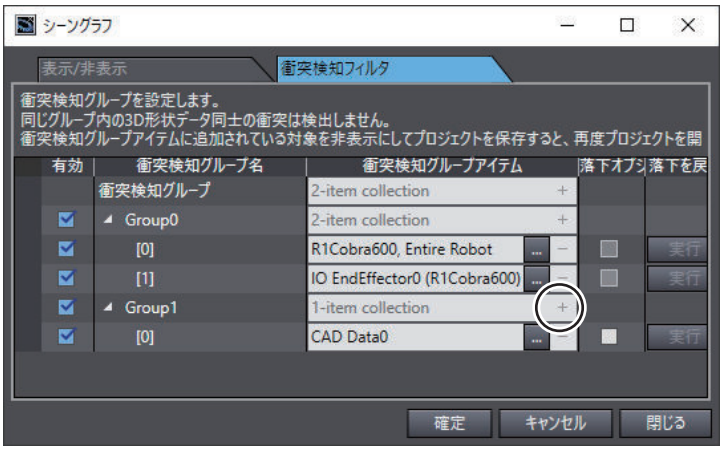
12 **ツリーから［ApplicationManager0］－［3D 表示］－［CAD データ 0］をクリックし、［選択］ボタンをクリックします。**



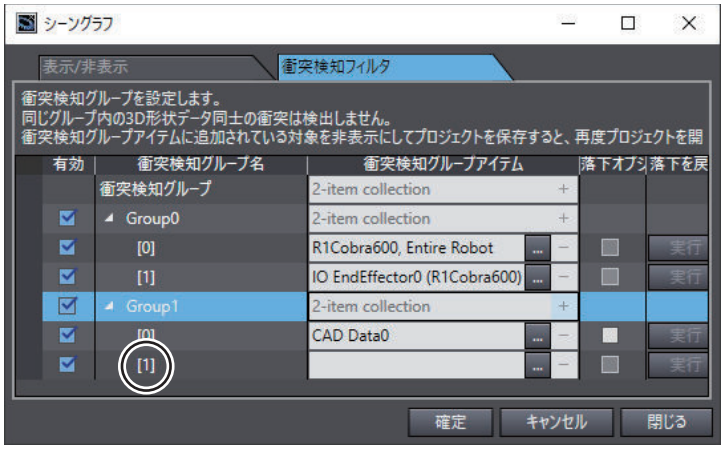
［衝突検知グループアイテムの選択］画面が閉じた後、［シーングラフ］画面に戻ります。［Group 1］－［[0]］の［衝突検知グループアイテム］に［CAD データ 0］が表示されます。



13 **［Group 1］の［衝突検知グループアイテム］の［+］マークをクリックします。**

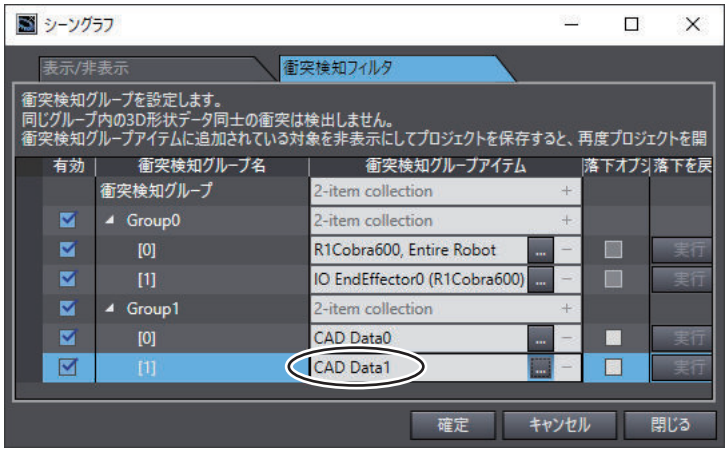


［Group 1］の下に［[1]］が追加されます。

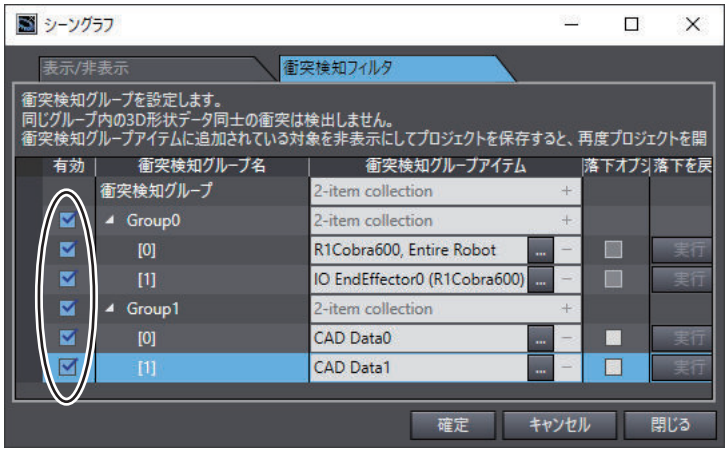


14 **[Group 1] – [[1]] についても同様に、手順 11 から手順 12 を行い［CAD データ 1］を追加します。**

［Group 1］–［[1]］–［衝突検知グループアイテム］に［CAD データ 1］が表示されます。



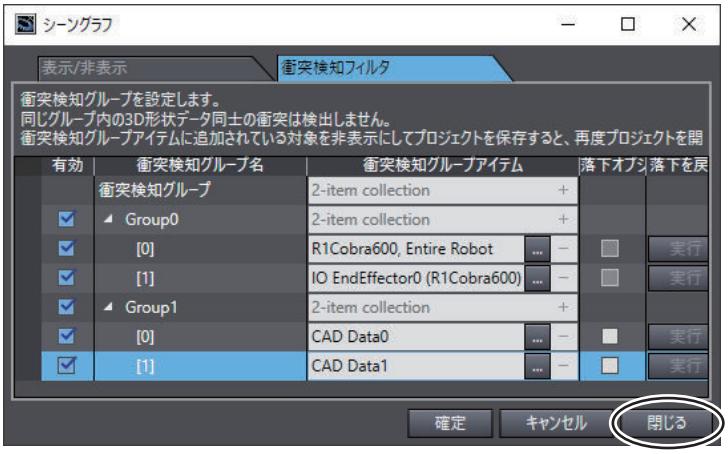
15 **[有効] のチェックボックスのすべてにチェックが入っていることを確認し、チェックが入っていない場合は、チェックボックスをクリックしてチェックを入れます。**



16 **[確定] ボタンをクリックします。**

衝突検知フィルタの設定が保存されます。

17 **［シーングラフ］画面の［閉じる］ボタンをクリックします。**



［シーングラフ］画面が閉じます。

これでロボットとコンベア A、コンベア B との干渉をチェックするための設定ができました。

3-3-6　シミュレーションを開始する

ロボットのシミュレーション機能を有効にする

Sysmac Studio のシミュレーション機能を有効にします。
ロボットのシミュレーションを行うためには、プロジェクトファイルをエミュレーションモードで開く必要があります。

Sysmac Studio でプロジェクトファイルを開いた状態でエミュレーションモードかどうかは、メインウィンドウのステータスバーの表示で確認してください。[［エミュレーションモード］アクセスレベル: Engineer] と表示されている場合は、プロジェクトはエミュレーションモードで開かれている状態です。



エミュレーションモードで開かれていない場合、ツールバーの［エミュレーションモードの有効化］アイコンを押し、エミュレーションモードとしてください。



1 **マルチビューエクスプローラのデバイスは、［new_Controller_0］を選択します。**



2 **メニューバーの［シミュレーション］-［実行］を選択します。**
シミュレータ起動中の画面が表示されます。

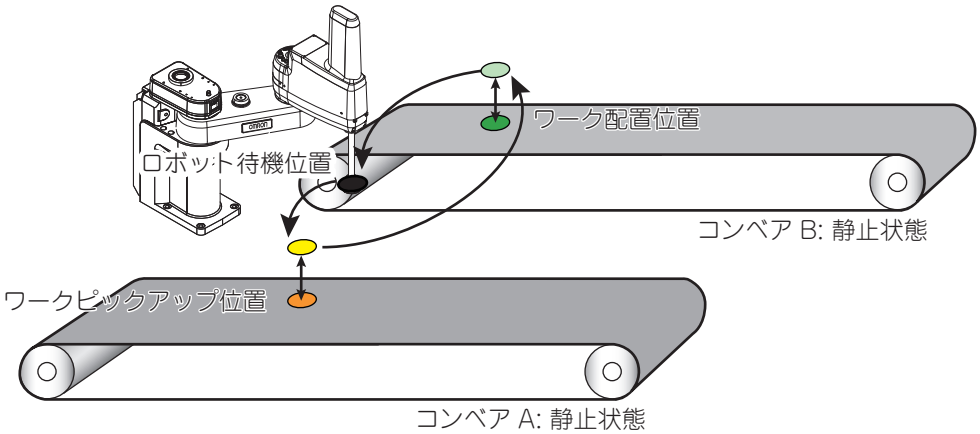


シミュレータ起動中の画面が閉じた後、メインウィンドウのツールバーの下部が黄緑色に変わります。



3-3-7 ティーチングする

Sysmac Studio のシミュレーションを用いてティーチングを行う操作について説明します。
本装置モデルでのティーチング対象は、ロボットの待機位置、ワークのピックアップ位置、ワークの配置位置です。



シミュレーション中にロボットを操作する方法は 2 種類あります。動作や動作に求める位置決め精度に応じて使い分けてください。

- 3D 表示画面でマウスを使ってロボットのツール先端を操作する。
- ［V+ジョグコントロール］画面でロボットの各軸を操作する。

参考

- おおよその位置にロボットを移動させる場合は、3D 表示画面でマウスを使ってロボットの先端を操作する方法が便利です。
- 正確な位置にロボットを移動させる場合は、［V+ジョグコントロール］画面でロボットの各軸を操作します。



ロボットにティーチングを行う場合、ロボットの位置データを V+の位置変数に保存する必要があります。ここでは、次の位置変数を作成します。

変数の名称	データ型	用途	対応させる IEC 変数
gl.pick	Location	ワークをピックアップする位置のデータを格納します。	eLoc_pick
gl.place	Location	ワークを配置する位置のデータを格納します。	eLoc_place
gl.wait	Location	ロボットの待機位置のデータを格納します。	eLoc_wait

参考

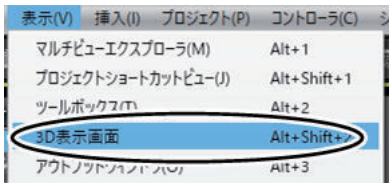
関連動画

- ロボット動作設定 静的ティーチング
  https://www.fa.omron.co.jp/video/detail/62645214760001/
  Sysmac Studio で、静的ピック&プレース装置の、ロボットの待機位置、ピック位置、プレース位置をティーチングする手順を説明しています。

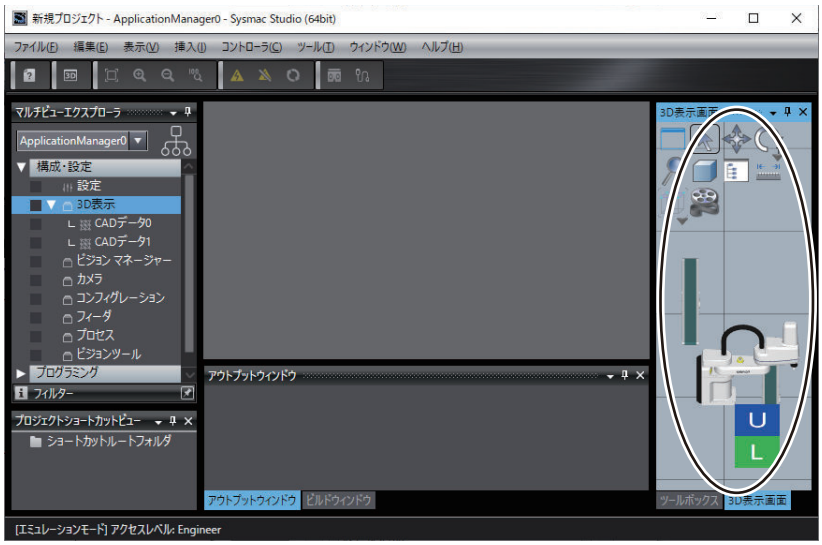
3D 表示画面でマウスを使ってロボットを操作する

3D 表示画面でロボットのツール先端を操作する手順を説明します。この操作ではロボットのツール先端がマウスカーソルに追従して動くため直感的なロボット操作が可能で、およその位置や姿勢を決めるのに向いています。

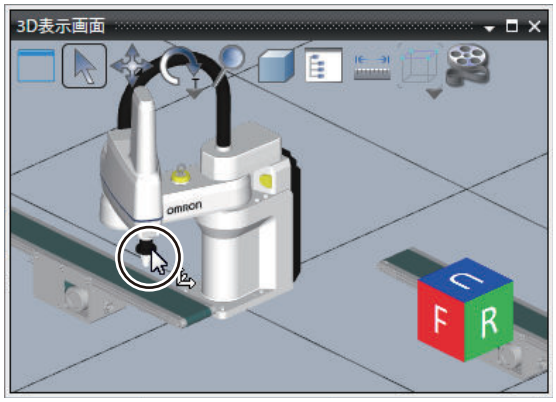
1 **Sysmac Studio のメインウィンドウのメニューバーから［表示］-［3D 表示画面］を選択します。**



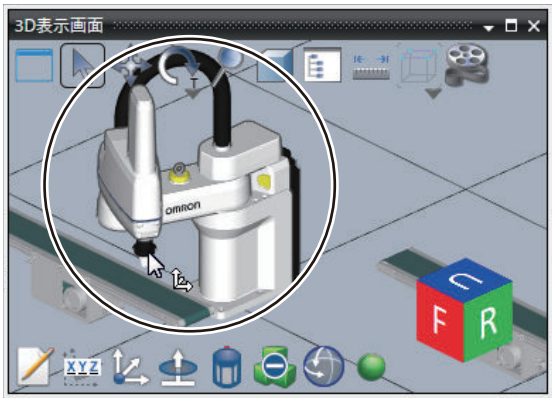
メインウィンドウ内の右側に 3D 表示画面が追加されます。



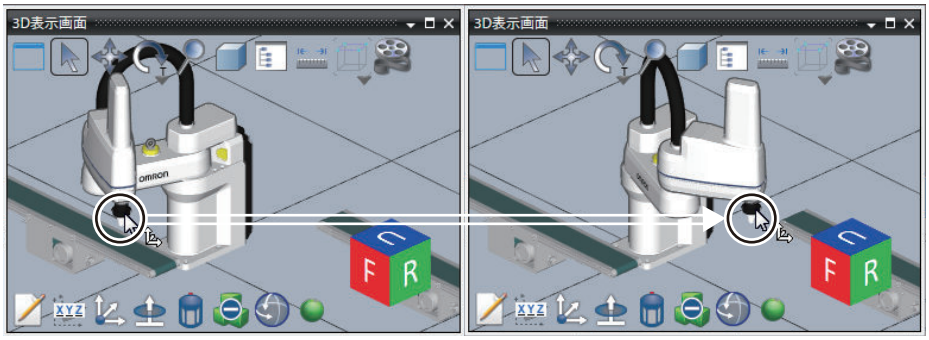
2 **3D 表示画面でロボットのツール先端にマウスカーソルを合わせてクリックします。**



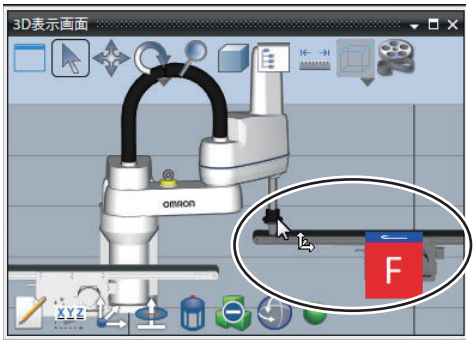
ロボットの輪郭が黒線で強調されます。



3 **そのままドラッグします。**
マウスカーソルに追従してツール先端が移動します。



4 **指定したい位置であるコンベアと重なる位置にツール先端を移動します。**
「衝突検知機能を有効にする (P.3-50)」で設定した衝突検知機能によりエンドエフェクタとコンベアの干渉が検知され、エンドエフェクタとコンベアが灰色になります。



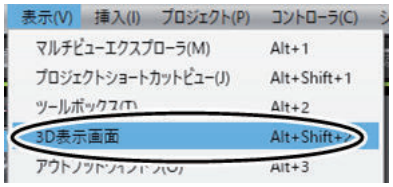
5 **エンドエフェクタとコンベアが灰色になった直後にドロップします。**
ドロップした位置でロボットが停止します。

これで、ロボットの位置が指定できました。現在のロボットの位置、姿勢でピックまたはプレースする場合は、「変数に現在位置を保存する (P.3-68)」の手順に進んでください。より精密な位置を指定したい場合や、ロボット座標系またはロボットの各ジョイントの軸に沿った位置を指定したい場合は、続けて「[V+ジョグコントロール] でロボットを操作する (P.3-64)」の手順に進んでください。

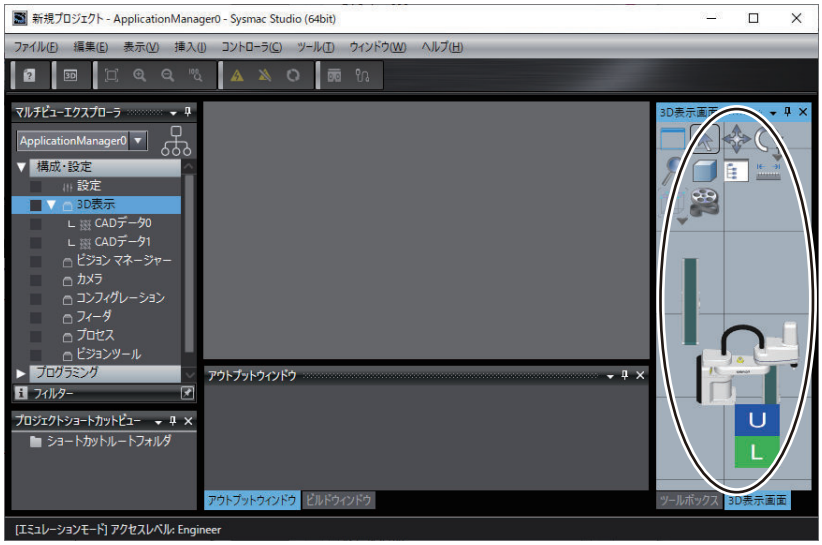
［V+ジョグコントロール］でロボットを操作する

［V+ジョグコントロール］画面でロボットの位置を操作する手順を説明します。この操作ではロボット座標系またはロボットの各ジョイントの軸に沿ったロボットの操作が可能です。

1 **Sysmac Studio のメインウィンドウのメニューバーから［表示］-［3D 表示画面］を選択します。**



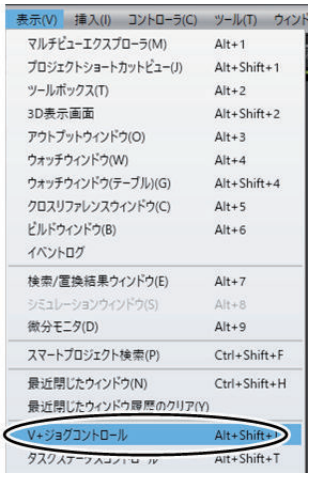
メインウィンドウ内の右側に 3D 表示画面が追加されます。



2 **マルチビューエクスプローラのデバイスは、［RobotControlSettings］を選択します。**



3 **メインウィンドウで、メニューバーから［表示］-［V+ジョグコントロール］を選択します。**



メインウィンドウの右に［V+ジョグコントロール］画面が追加されます。



4 **［V+ジョグコントロール］画面の［ジョグ操作］のドロップダウンリストから、［World］、［Joint］、［Tool］のいずれかの座標系を選択します。**

［World］はロボット座標系、［Joint］はジョイント座標系、［Tool］はロボットのツール先端の座標系でロボットの位置の指定が可能です。

ここでは、［World］を選択するとします。



5 **[V+ジョグコントロール] 画面の [ジョグ操作] のランプをクリックします。**

クリックしたランプが緑色に点灯します。点灯したランプの座標軸でロボットを操作できます。

ここでは、[Z] ランプを選択するとします。

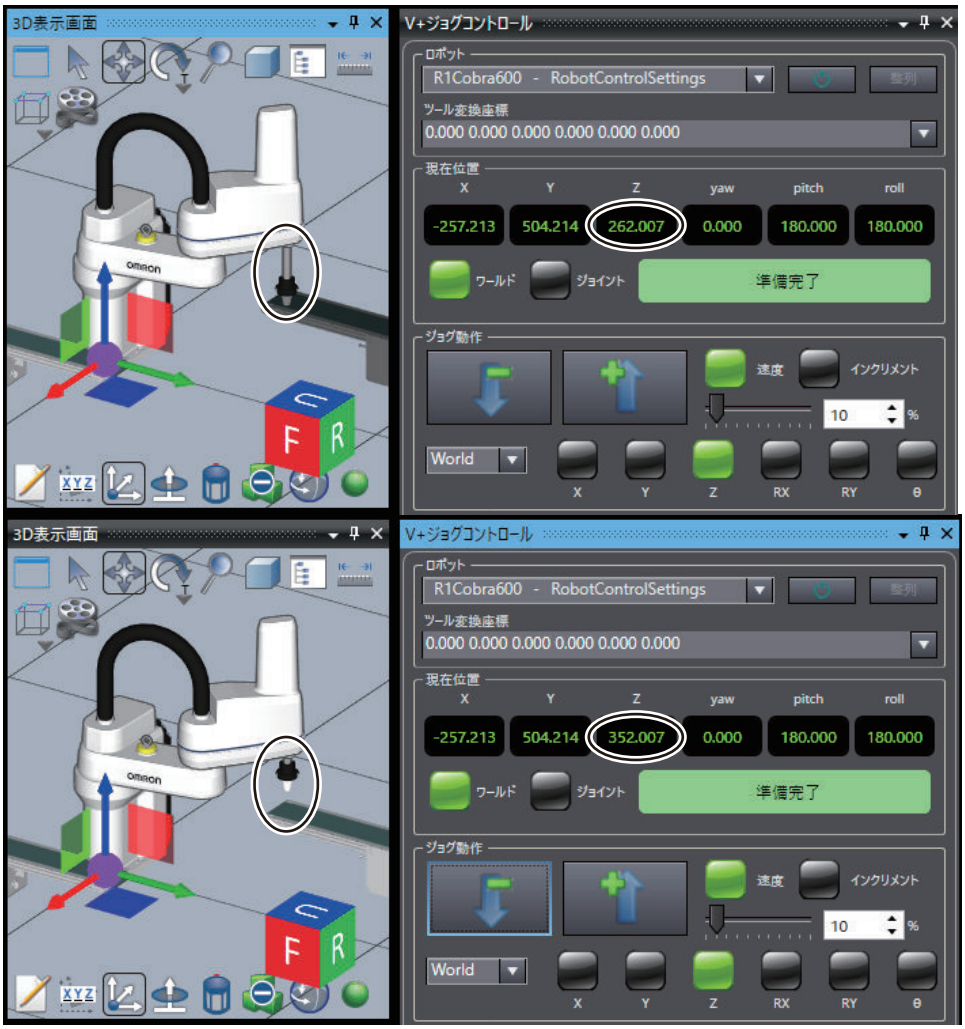


6 **[V+ジョグコントロール] 画面の [ジョグ操作] の [＋] または [－] が書かれた矢印ボタンを押し続けます。**

矢印ボタンの [＋]、[－] は指定した座標系の正負方向を表しています。たとえばロボット座標系では次の矢印方向が正方向です。



矢印ボタンを押し続けている間、指定した座標系の座標軸に沿ってロボットが移動します。



7 手順 4 から手順 6 を繰り返し、ロボットが指定したい位置や姿勢になるように操作します。

これで、ロボットの位置を指定できました。

変数に現在位置を保存する

ティーチングにより設定したロボットの現在位置は、V+の変数に保存する必要があります。
ティーチングにより設定したロボットの現在位置を、既存の V+の変数に保存する手順を説明します。
保存が必要な V+位置変数は以下のとおりです。

位置の名称（変数の名称）	説明
gl.wait	ロボットの待機位置
gl.pick	ワークをピックアップする位置
gl.place	ワークを配置する位置

ここでは例として、変数「gl.wait」に現在位置を保存します。そのほかの変数についても同様の操作を行ってください。

1 **［V+ジョグコントロール］画面の［ロケーション］をクリックします。**



位置変数を設定する画面が展開されます。



2 **ドロップダウンリストから、現在位置を保存したい変数または配列変数の要素を選択します。**



選択した変数名と、その変数に保存されている値が表示されます。



ここでは、変数「gl.wait」を選択するとします。

3 **[V+ジョグコントロール] 画面の [ロケーション] の [Here] ボタンをクリックします。**



選択している変数の値が現在位置に更新されます。



これで、既存の V+の変数にロボットの現在位置が保存されました。

参考

シーケンス制御プログラムで動作を制御する場合、変数コピープログラム“loccopy”により、V+プログラムの位置変数をシーケンス制御プログラムの共有変数にコピーします。

3-3-8 V+のプログラムと変数の電源 ON 時の自動読込を設定する

作成した V+プログラムおよび V+のグローバル変数は、プロジェクトの初期設定では、ロボット統合 CPU ユニットの電源 ON 時に自動的に読み込まれません。
ここでは、ロボット統合 CPU ユニットの電源 ON 時に、V+プログラムおよび V+のグローバル変数を自動的に読み込むための設定の手順を説明します。
シーケンス制御プログラムで動作を制御する場合は、手順 4 から手順 6 の実行は不要です。

参考

関連動画

- V+プログラムと変数の電源 ON 時の自動読み込みの設定
  https://www.fa.omron.co.jp/video/detail/6294914962001/
  電源 ON 時に V+プログラムとグローバル変数を自動的に読み込むための設定手順を説明しています。

1 **マルチビューエクスプローラのデバイスは、［RobotControlSettings］を選択します。**



2 **マルチビューエクスプローラの［構成・設定］をクリックします。**



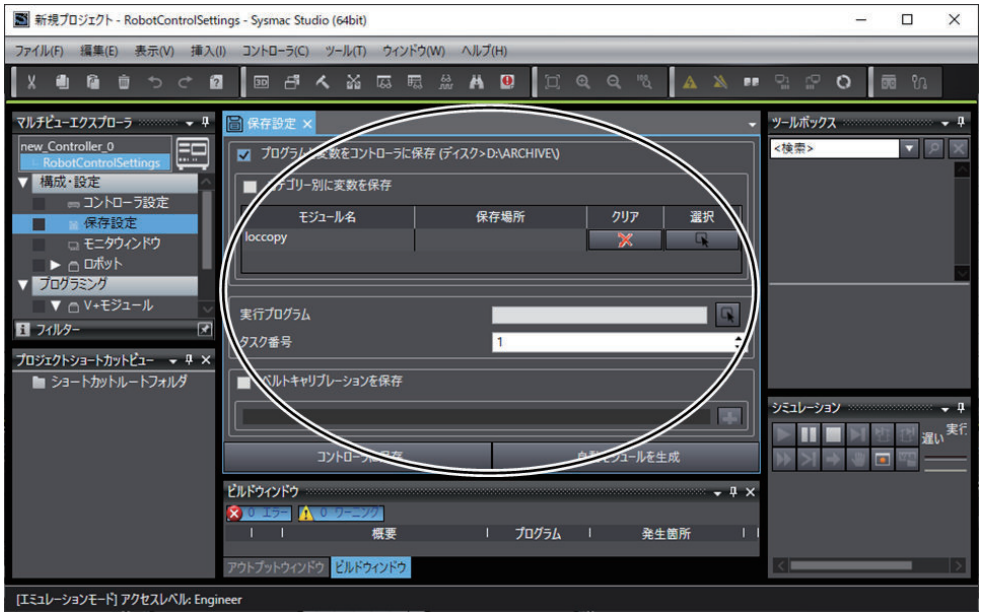
［構成・設定］の下にツリーが展開されます。



3 **［保存設定］をダブルクリックします。**



エディットウィンドウに［保存設定］タブが開きます。



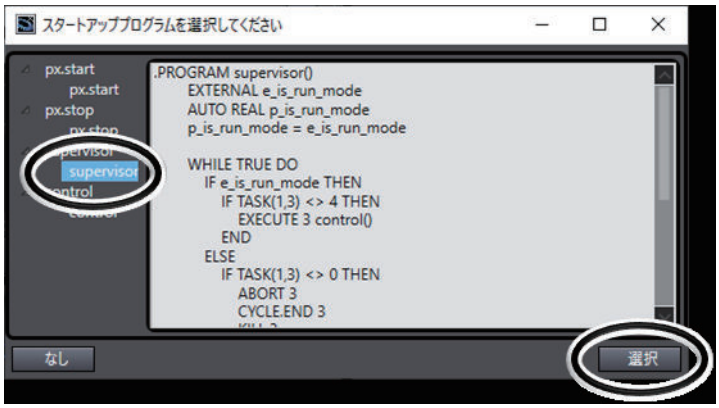
V+プログラムで動作を制御する場合は、手順 4 へ進んでください。
シーケンス制御プログラムで動作を制御する場合は、手順 7 へ進んでください。

4 **[実行プログラム] に「supervisor」が表示されていないことを確認し、右のボタンをクリックします。**



［スタートアッププログラムを選択してください］画面が表示されます。

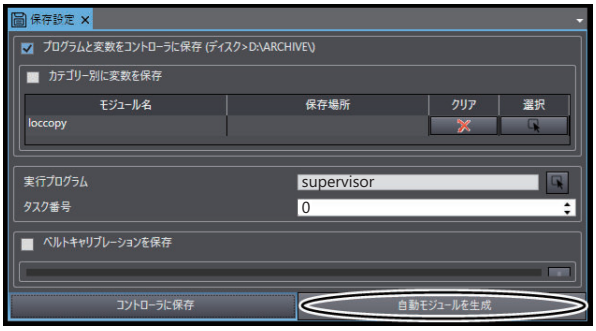
5 **[supervisor] – [supervisor] をクリックして [選択] ボタンをクリックします。**



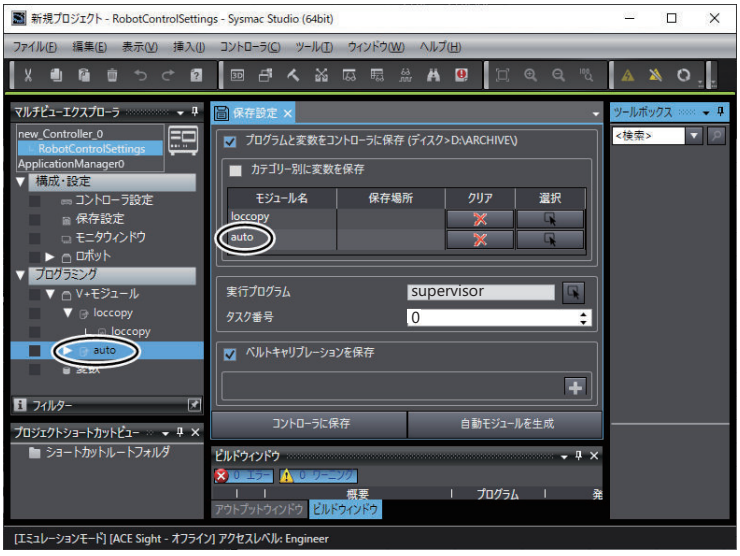
6 **[タスク番号] を「0」に設定します。**



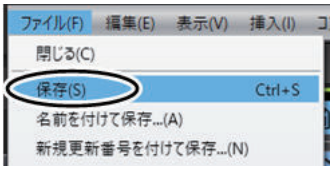
7 **［保存設定］の［自動モジュールを生成］ボタンをクリックします。**



［保存設定］タブのモジュールリストに［auto］が追加されます。また、マルチビューエクスプローラのツリーに［プログラミング］-［V+モジュール］-［auto］が追加されます。



8 **メニューバーの［ファイル］-［保存］をクリックします。**



これで、ロボット統合 CPU ユニットの電源 ON 時に、V+プログラムおよび V+のグローバル変数を自動的に読み込むための設定ができました。プログラム［auto］はロボット統合 CPU ユニットの電源 ON 時に自動的に実行され、保存された V+プログラムおよび V+のグローバル変数を読み込みます。

プログラム［auto］は通常の V+プログラムと異なるコマンドプログラムと呼ばれる形式で作成されます。コマンドプログラムに関する詳細については、『eV+3 ユーザーズマニュアル (SBCE-486)』を参照してください。

3-3-9　プログラムを実行して動作を確認する

「3-3-3 シーケンス制御プログラムを作成する (P.3-20)」で作成したシーケンス制御プログラムをシミュレーションで実行する手順を説明します。
ウォッチウィンドウでシーケンス制御プログラムのグローバル変数「gStart」を FALSE から TRUE にすることで、プログラムの実行を開始できます。
なお、V+プログラムは、本手順にて実行されたシーケンス制御プログラムから呼び出されて実行されます。

使用上の注意

本手順を行う場合は、エミュレーションモードでプロジェクトファイルを開いていることを確認してください。確認方法については、「ロボットのシミュレーション機能を有効にする (P.3-59)」を参照してください。

参考

プログラムのデバッグ機能については、『Sysmac Studio Version 1 オペレーションマニュアル (SBCA-470)』および『Sysmac Studio ロボット統合システム構築機能 オペレーションマニュアル ロボット統合 CPU ユニット編 (SBCA-464)』を参照してください。

参考

関連動画

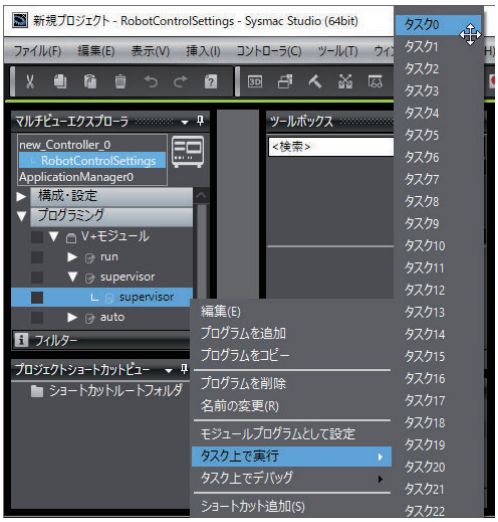
- シミュレーションでの動作確認
  https://www.fa.omron.co.jp/video/detail/6264519415001/
  Sysmac Studio のシミュレーション機能を使用してロボットの動作を確認する手順を説明しています。

V+プログラムで動作を制御する場合は、手順 1 から実行してください。
シーケンス制御プログラムで動作を制御する場合は、手順 3 から実行してください。

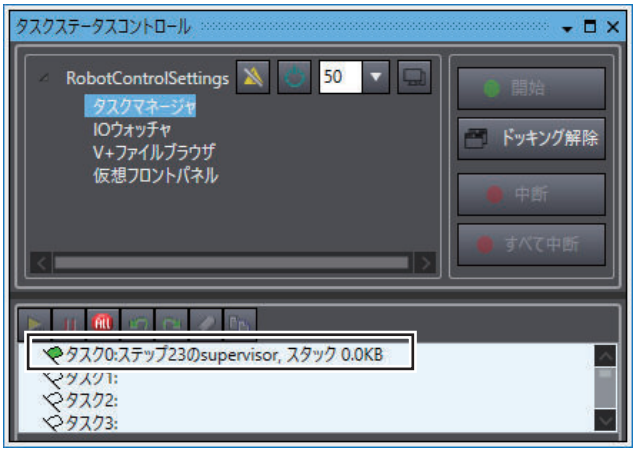
1 **マルチビューエクスプローラーのデバイスで、［RobotControlSettings］を選択します。**



2 **マルチビューエクスプローラで［プログラミング］－［V+モジュール］－［supervisor］－［supervisor］上で右クリックし、［タスク上で実行］－［タスク 0］を選択します。**



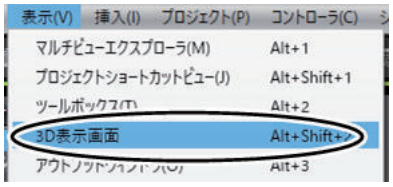
タスク 0 で V+プログラム supervisor が実行されます。



3 **マルチビューエクスプローラのデバイスで、［new_Controller_0］を選択します。**



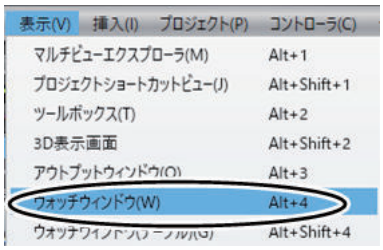
4 **Sysmac Studio のメインウィンドウのメニューバーから［表示］－［3D 表示画面］を選択します。**



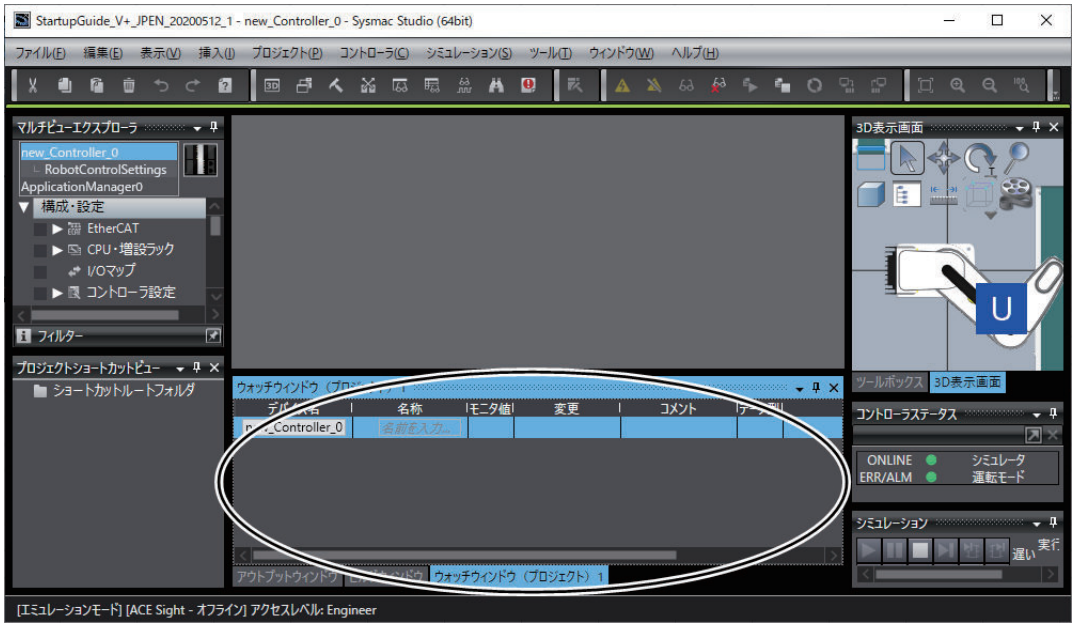
メインウィンドウ内の右側に 3D 表示画面が追加されます。



5 **メインウィンドウのメニューバーから［表示］–［ウォッチウィンドウ］を選択します。**



メインウィンドウの下に［ウォッチウィンドウ（プロジェクト）1］画面が表示されます。



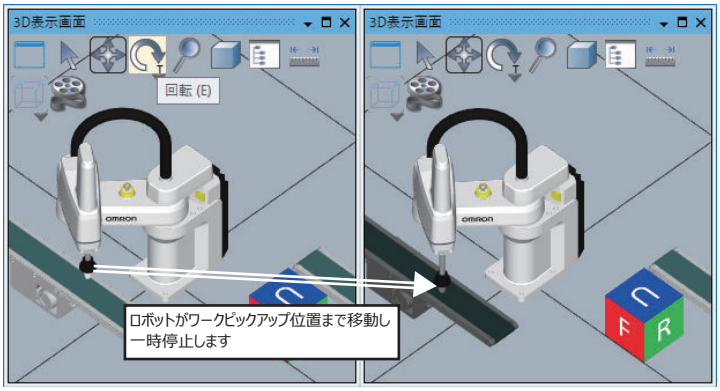
6 **［ウォッチウィンドウ（プロジェクト）1］画面の［名称］列に「gStart」と入力します。**
［モニタ値］列にプログラム［Program0］の内部変数［gStart］の現在値［False］が表示されます。



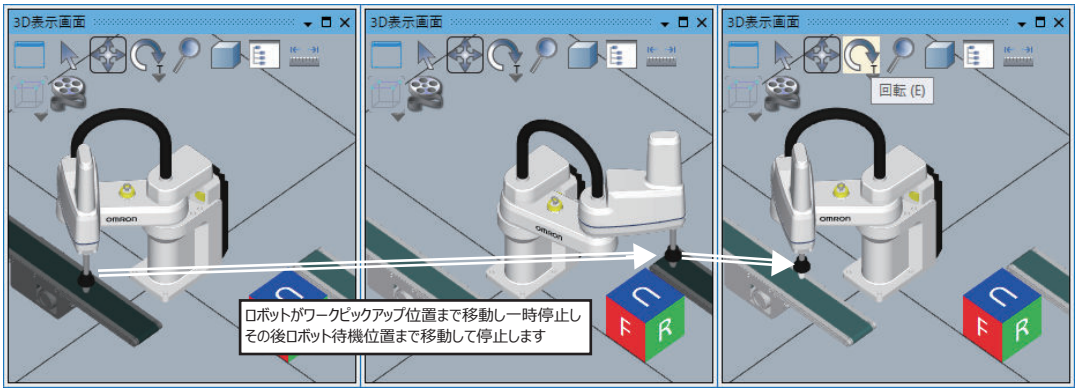
7 **［ウォッチウィンドウ（プロジェクト）1］画面で変数［gStart］の［変更］列の［TRUE］ボタンをクリックします。**



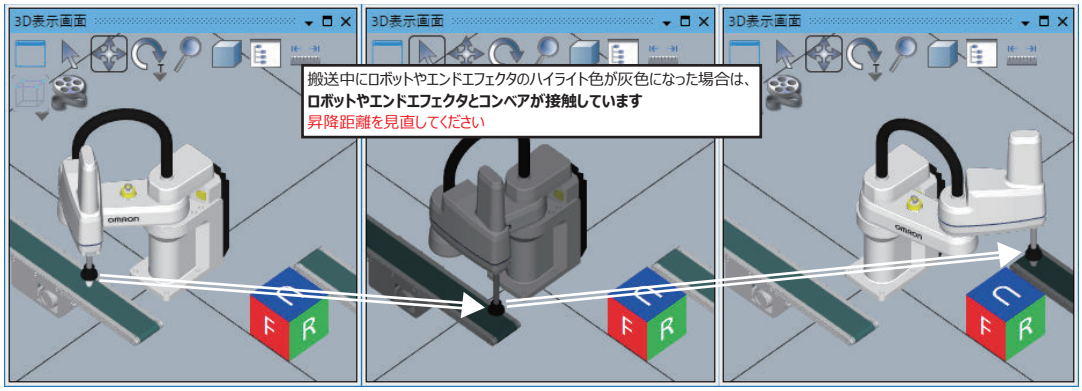
3D 表示画面でロボットが動作を開始し、ワークのピックアップ位置で一時停止します。



その後、3D 表示画面でロボットがワークのピックアップ位置から動作を再開し、ワークの配置位置を経由してロボット待機位置まで移動した後、停止します。



動作過程では、ワークのピックアップ位置およびワークの配置位置以外でロボットが灰色にならないかを確認してください。灰色になる場合はロボットとコンベアが干渉しています。「［V+ジョグコントロール］でロボットを操作する (P.3-64)」に記載されている操作でワークのピックアップ経由点またはワークの配置経由点をより高い位置に設定するなどしてティーチングを変更してください。

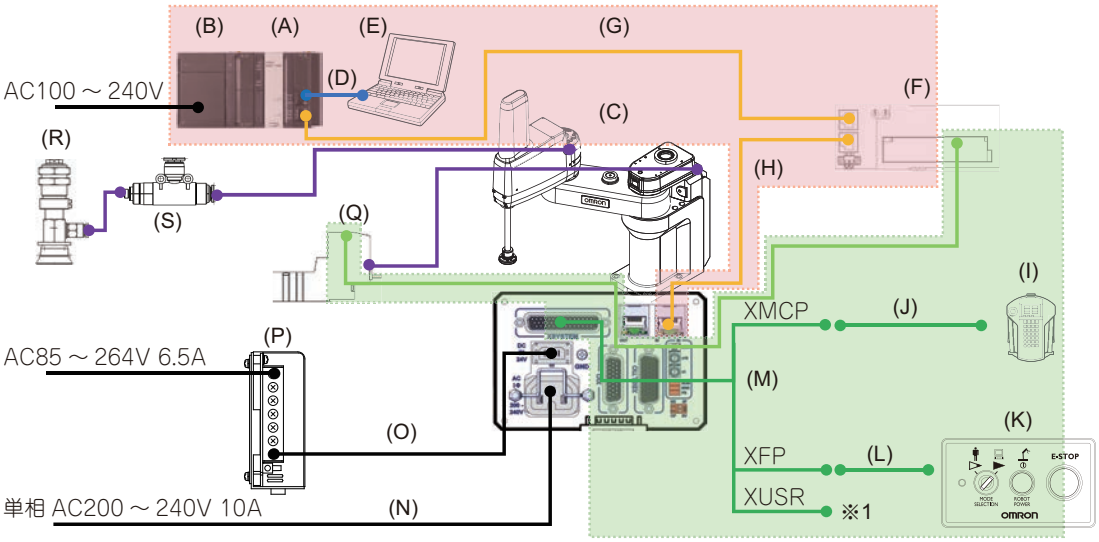


これで、シーケンス制御プログラムと V+プログラムの動作の確認が完了しました。

3-4 設置、配線する

システム構成において、図中の枠で囲われている以下の信号線の配線と、ロボットのノードアドレスの設定を行います。

- ロボット統合 CPU ユニット(A)と EtherCAT デジタル I/O ターミナル(F)間の配線
- EtherCAT デジタル I/O ターミナル(F)とロボット(C)間の配線
- ロボット統合 CPU ユニット(A)とパソコン(E)間の配線
- ロボット(C)と T20 ペンダント(I)間の配線
- ロボット(C)とフロントパネル(K)間の配線
- EtherCAT デジタル I/O ターミナル(F)とソレノイドバルブ(Q)間の配線

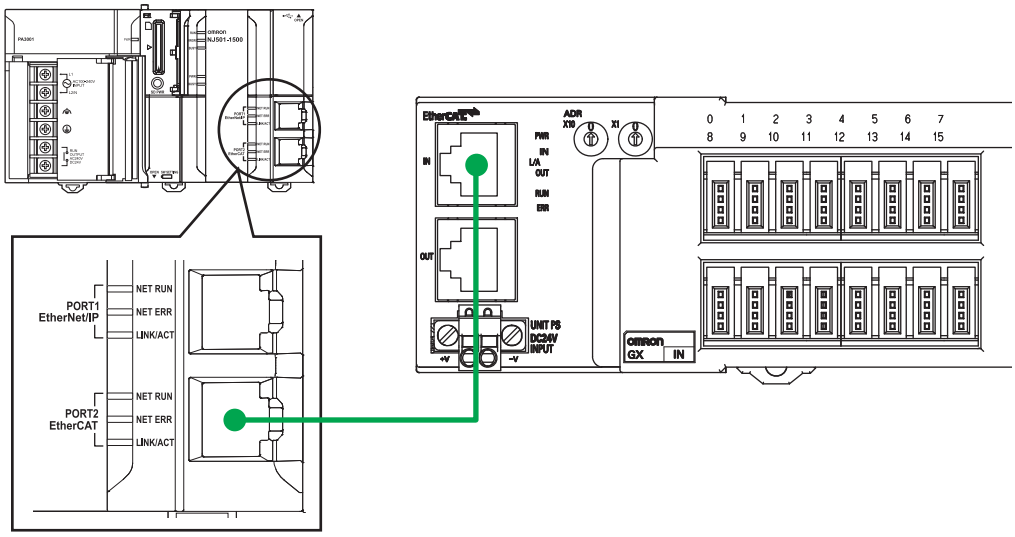


システム構成で使用する部品については、「1-3 静的ピック & プレース装置のシステム構成 (P.1-4)」を参照してください。
図中の枠で囲われていない電源線、およびロボットからの配管については、各製品のマニュアルを参照して配線、配管してください。

3-4-1 ロボット統合 CPU ユニットと EtherCAT デジタル I/O ターミナル間を配線する

ロボット統合 CPU ユニットと EtherCAT デジタル I/O ターミナル間を配線します。

1 **ロボット統合 CPU ユニットの EtherCAT ポートと、EtherCAT デジタル I/O ターミナルにある EtherCAT の IN ポートを Ethernet ケーブルで接続します。**

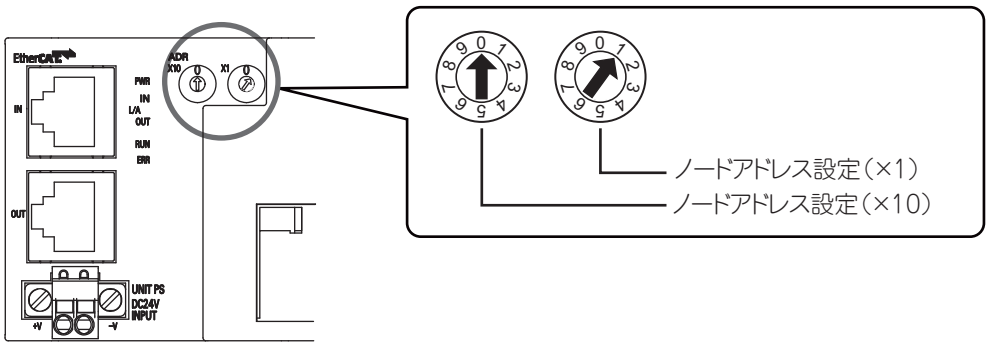


3-4-2 EtherCAT デジタル I/O ターミナルの EtherCAT ノードアドレスを設定する

EtherCAT デジタル I/O ターミナルの EtherCAT ノードアドレスを設定します。

1 **EtherCAT デジタル I/O ターミナルにあるスイッチで EtherCAT ノードアドレスを設定します。**

ノードアドレスを 1 に設定するため、×10 のスイッチを 0、×1 のスイッチを 1 に設定します。



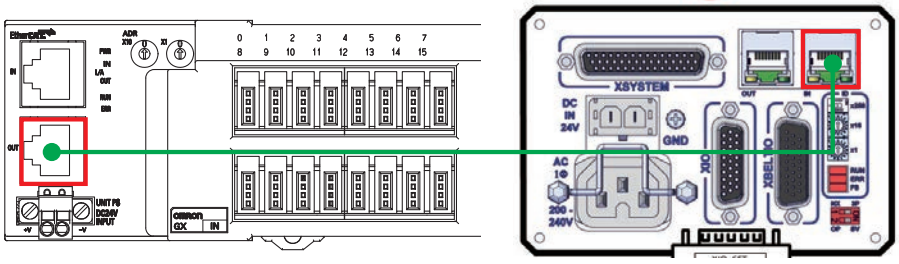
使用上の注意

デジタル I/O ターミナルの EtherCAT ノードアドレスは、「3-3-2 EtherCAT ネットワーク構成を作成する (P.3-17)」で設定したノードアドレスと一致するように設定します。

3-4-3 EtherCAT デジタル I/O ターミナルとロボット間を配線する

EtherCAT デジタル I/O ターミナルとロボット間を配線します。

1 **EtherCAT デジタル I/O ターミナルにある EtherCAT の OUT ポートと、ロボットのインタフェースパネルにある EtherCAT の IN ポートを Ethernet ケーブルで接続します。**



使用上の注意

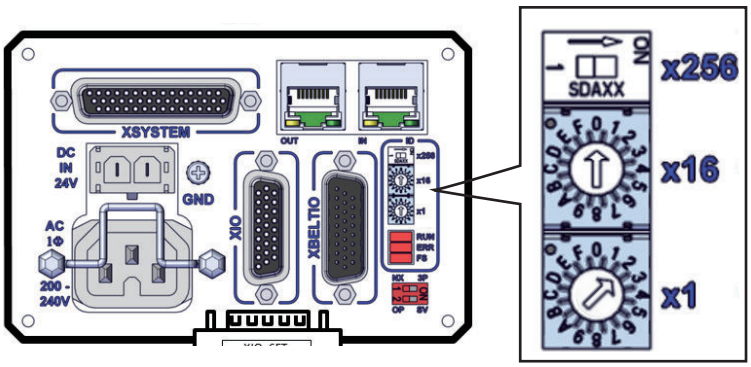
使用する Ethernet ケーブルの詳細については、『NJ/NX シリーズ CPU ユニット内蔵 EtherCAT ポート ユーザーズマニュアル (SBCD-376)』を参照してください。

3-4-4 ロボットの EtherCAT ノードアドレスを設定する

ロボットの EtherCAT ノードアドレスを設定します。

1 ロボットのインタフェースパネルにあるスイッチでロボットの EtherCAT ノードアドレスを設定します。

ノードアドレスを 2 に設定するため、×256 のスイッチを OFF、×16 のスイッチを 0、×1 のスイッチを 2 に設定します。



使用上の注意

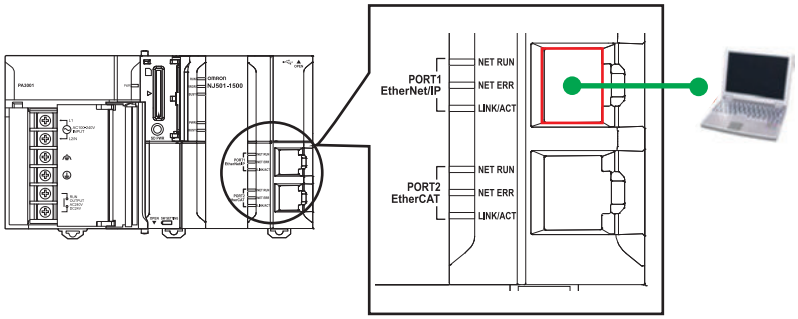
ロボットの EtherCAT ノードアドレスは、「3-3-2 EtherCAT ネットワーク構成を作成する (P.3-17)」で設定したノードアドレスと一致するように設定します。

ノードアドレスの設定方法については、ロボットのマニュアルを参照してください。

3-4-5 ロボット統合 CPU ユニットとパソコン間を配線する

ロボット統合 CPU ユニットとパソコン間を配線します。

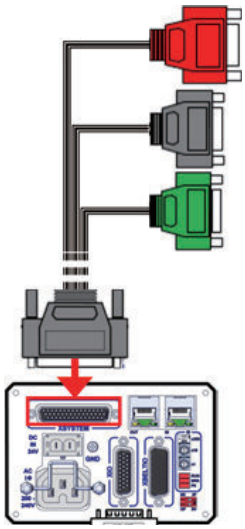
1 ロボット統合 CPU ユニットの内蔵 EtherNet/IP ポートと、パソコンを Ethernet ケーブルで接続します。



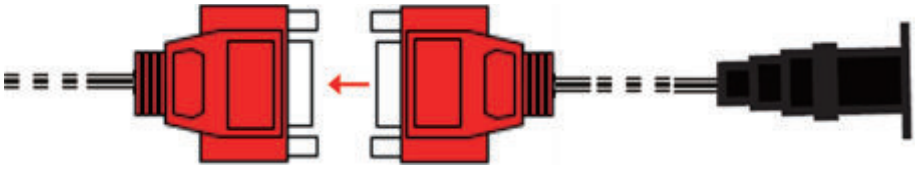
3-4-6 ロボットと T20 ペンダント、フロントパネル間を配線する

ロボットと T20 ペンダント間、およびロボットとフロントパネル間を配線します。配線には、XSYSTEM ケーブルアセンブリ、T20 アダプタケーブル、フロントパネルケーブルを使用します。

1 **ロボットと XSYSTEM ケーブルアセンブリを接続します。**



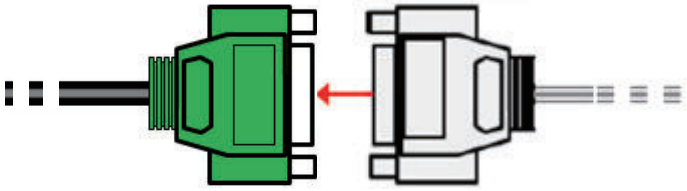
2 **XSYSTEM ケーブルアセンブリの XMCP コネクタ（赤）と T20 アダプタケーブルを接続します。**



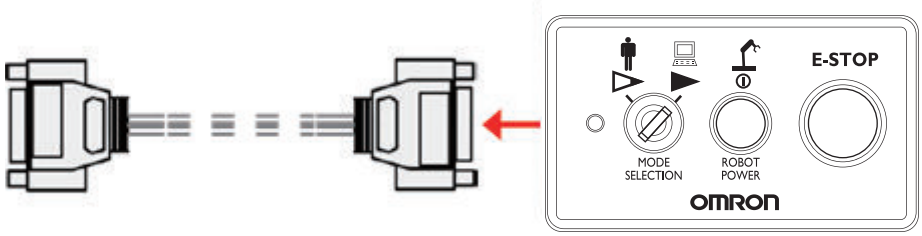
3 **T20 アダプタケーブルと T20 ペンダントを接続します。**



4 **XSYSTEM ケーブルアセンブリの XFP コネクタ（緑）とフロントパネルケーブルを接続します。**



5 **フロントパネルケーブルとフロントパネルを接続します。**



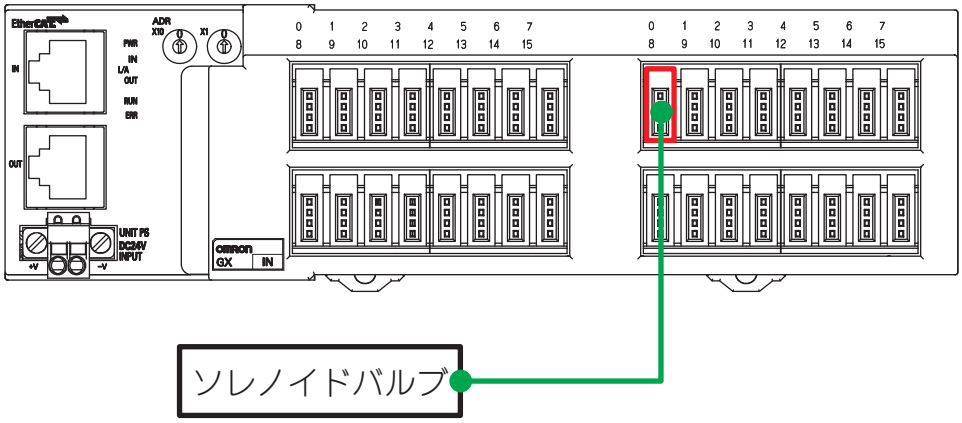
3-4-7 EtherCAT デジタル I/O ターミナルとソレノイドバルブ間を配線する

EtherCAT デジタル I/O ターミナルとソレノイドバルブ間を配線します。
配線するケーブルについては、EtherCAT デジタル I/O ターミナルおよびソレノイドバルブのマニュアルを参照し、配線図や仕様に応じて適切に制作してください。

1 **ソレノイドバルブが以下の挙動となるよう、ケーブルを制作します。**

デジタル I/O ターミナルの出力値	ソレノイドバルブの振る舞い
FALSE	配管が閉じる
TRUE	配管が開く

2 **EtherCAT デジタル I/O ターミナルとソレノイドバルブ間を配線します。**


EtherCAT
IN
OUT
PWR
IN
L/A
OUT
RUN
ERR
ADR
X10
X1
UNIT PS
DC24V
INPUT
+V
-V
OMRON
GX
IN
0 1 2 3 4 5 6 7
8 9 10 11 12 13 14 15
0 1 2 3 4 5 6 7
8 9 10 11 12 13 14 15
ソレノイドバルブ

3-5　実機の立ち上げ手順

実機での動作確認、調整の手順を説明します。
実機での動作確認の前に、シミュレーションで動作を確認しておいてください。

使用上の注意

T20 ペンダントによりロボットをマニュアルモードで操作する場合、iCS 試運転ジャンパを XBELTIO コネクタに接続し、ティーチング制限に関する設定を変更する必要があります。詳細は、『スカラロボット eCobra 600/800 EtherCAT タイプ ユーザーズマニュアル (SBCE-489)』を参照してください。

3-5-1　オンライン接続する

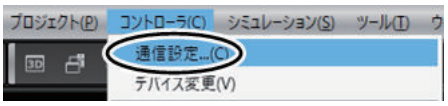
ロボット統合 CPU ユニットとパソコンをオンライン接続します。

1 **ロボット統合 CPU ユニットとロボットの電源を ON します。**

2 **Sysmac Studio のマルチビューエクスプローラのデバイスリストから［new_Controller_0］を選択します。**



3 **メインウィンドウのメニューバーから［コントローラ］-［通信設定］を選択します。**



［通信設定］画面が開きます。



4 **接続方法で［Ethernet-直接接続］ラジオボタンをクリックし、［OK］ボタンをクリックします。**

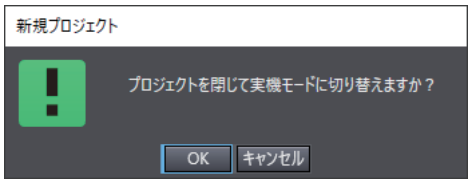


［通信設定］画面が閉じます。

5 **ツールバーの［エミュレーションモードの無効化］アイコンをクリックします。**

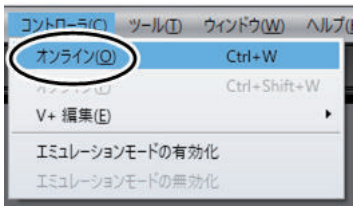


エミュレーションモードの無効化の確認ダイアログが表示されます。



6 **［OK］ボタンをクリックします。**
Sysmac Studio がメニュー画面に戻ったのち、メインウィンドウが再度表示されます。

7 **メニューバーから［コントローラ］-［オンライン］を選択します。**



使用上の注意

接続先のロボット統合 CPU ユニットの状態によっては、CPU ユニット名称の書き込みやシリアル ID の確認などのダイアログが表示されることがあります。詳細については、『Sysmac Studio Version 1 オペレーションマニュアル (SBCA-470)』を参照してください。

オンライン状態になると、エディットウィンドウの上部に黄線が表示されます。



3-5-2 設定とプログラムを転送する

Sysmac Studio のプロジェクトファイル内の設定およびプログラムをロボット統合 CPU ユニットに転送します。

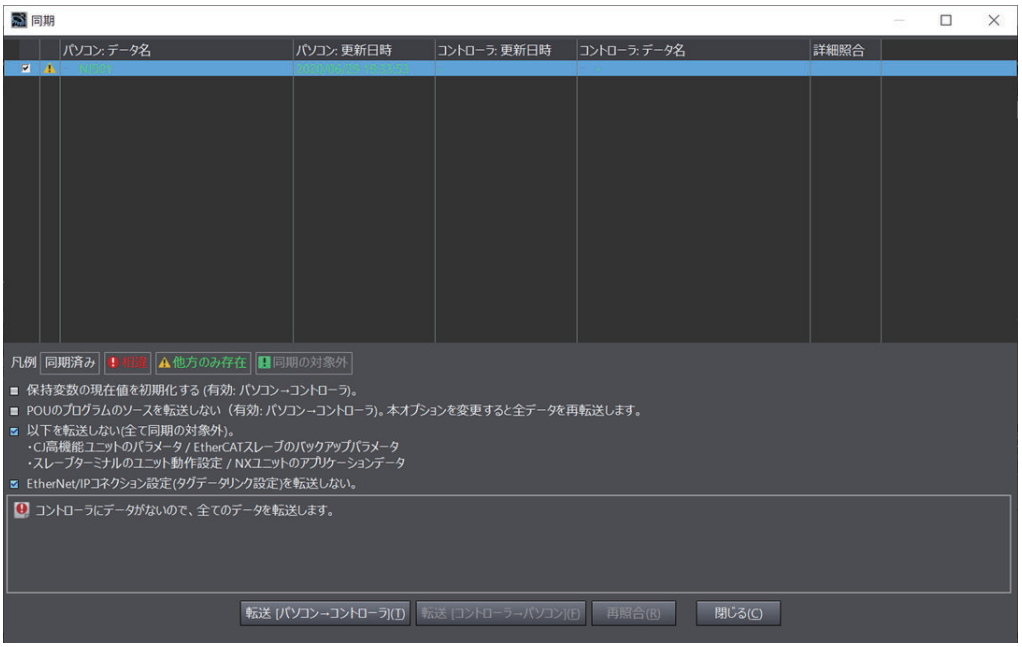
1 **Sysmac Studio のメニューバーから［コントローラ］-［同期］を選択します。**



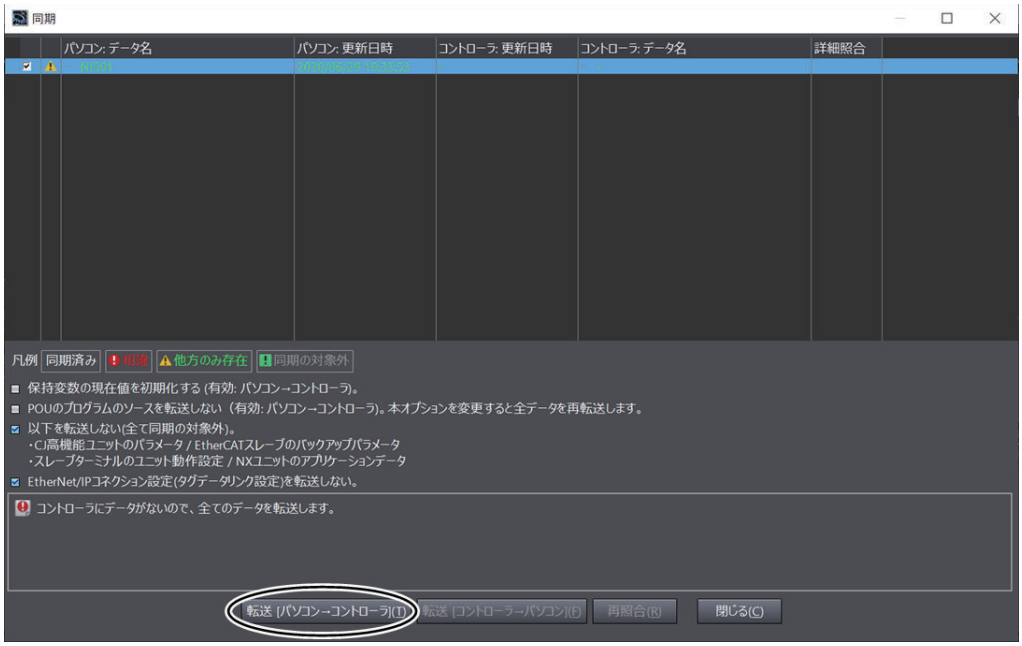
Sysmac Studio とコントローラ内のプログラムおよび設定データの照合が開始されます。



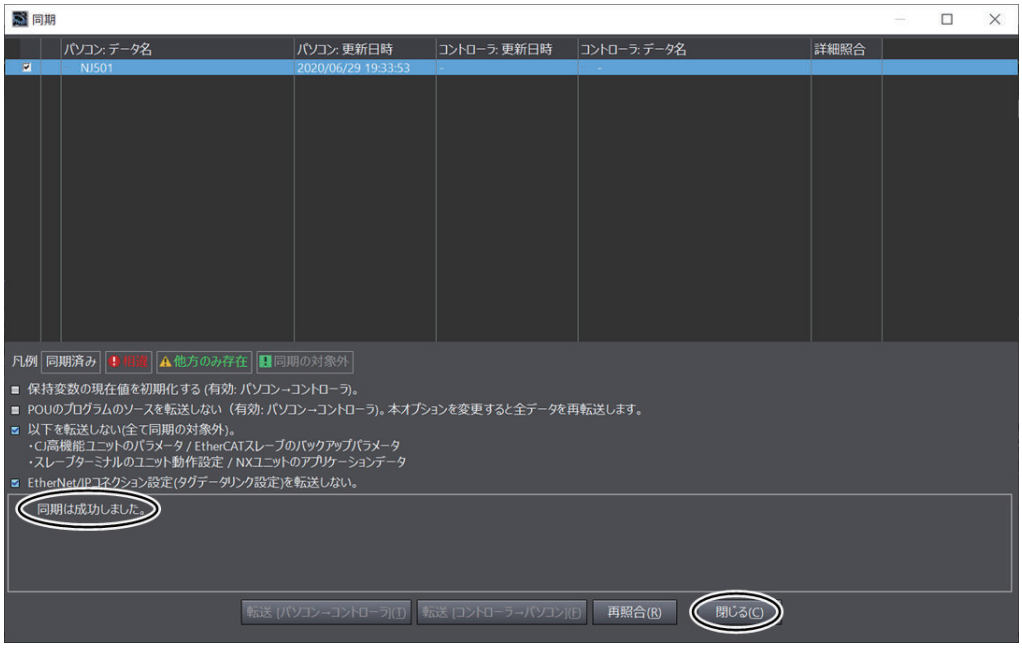
照合が完了すると、同期ウィンドウが表示されます。



2 **［転送（パソコン→コントローラ)］ボタンをクリックします。**



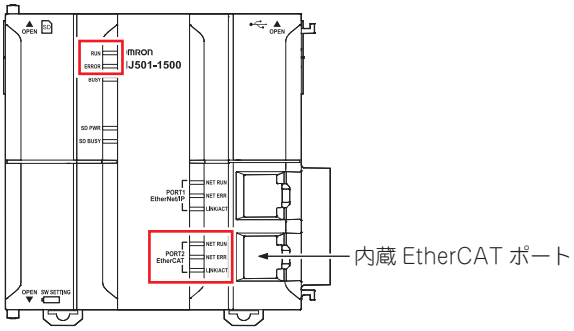
3 **“同期は成功しました。”と表示されたことを確認し、[閉じる] ボタンをクリックします。**



4 **ロボット統合 CPU ユニットが正常に起動したことを確認します。**
正常時の LED 状態は次のとおりです。

LED	状態
RUN LED	緑点灯
ERROR LED	消灯
NET RUN	緑点灯
NET ERR	消灯

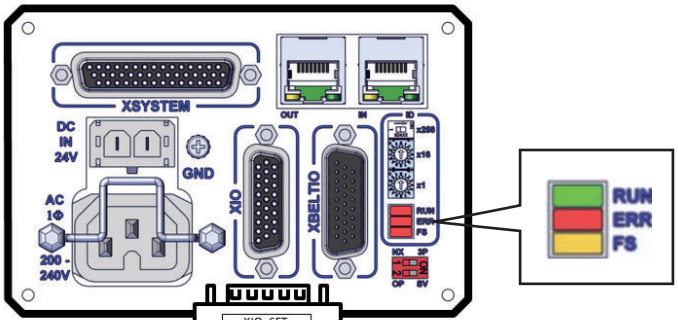
LED	状態
LINK/ACT	点滅



5 **ロボットが正常に起動したことを確認します。**

正常時の LED 状態は次のとおりです。

LED	状態
RUN LED	緑点灯
ERR LED	消灯



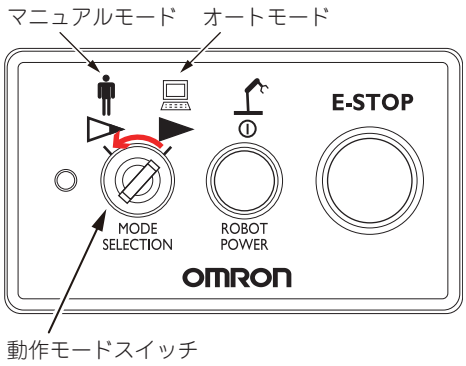
3-5-3 ティーチングする

実際にロボットを操作してティーチングデータを確認、修正します。
シミュレーションでティーチングした位置に実際にロボットを移動し、必要に応じて位置を調整します。ティーチングした位置データは、ロボット統合 CPU ユニットの電源を OFF にした後も保持されるよう SD メモリカードに保存します。
ティーチングには T20 ペンダントを使用します。T20 ペンダントの仕様は、『ティーチングペンダント T20 ユーザーズマニュアル (SBCE-392)』を参照してください。
また、ティーチングの操作の実施の前に、ワークをピック予定位置に設置してください。

ロボットのハイパワーを ON する

ロボットのハイパワーを ON します。

1 **フロントパネルの動作モードスイッチをマニュアルモードにします。**



2 **T20 ペンダントのイネーブルスイッチを中間位置（ポジション 2）まで押します。**



3 **イネーブルスイッチを中間位置（ポジション 2）にしたまま、ROBOT POWER ボタンを押します。**



フロントパネルの ROBOT POWER ボタンが点滅します。

4 **フロントパネルの ROBOT POWER ボタンを押します。**
ロボットのハイパワーが ON になり、ROBOT POWER ボタンが点灯します。

イネーブルスイッチを離したり（ポジション 1）、中間位置（ポジション 2）からさらに握り込んだり（ポジション 3）すると、ロボットのハイパワーは OFF になります。

使用上の注意

ロボットの移動が必要な場合は、上記の手順でハイパワーを ON にしてください。

シミュレーションでティーチングした位置を調整する

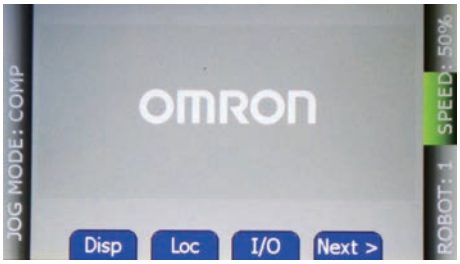
シミュレーションを使った立ち上げ手順でティーチングした位置を、実際の設置状態に合わせて調整します。ロボットをシミュレーションでティーチングした位置まで実際に移動し、期待する位置とのずれがある場合にはティーチングの位置を修正します。
位置の調整はロボットの移動経路に合わせて以下の順序で行います。待機位置および経由位置は複数回の調整が必要です。

位置の名称（変数の名称）	説明	調整の順序
gl.wait	ロボットの待機位置	1, 4
gl.pick	ワークをピックアップする位置	2
gl.place	ワークを配置する位置	3

使用上の注意

調整は上記の順序で行ってください。異なる順序で実行した場合、意図しない移動経路となり、コンベアなどの周辺設備とロボットが接触し機器が破損するおそれがあります。

1 **T20 ペンダントに［HOME1］画面が表示されていることを確認します。**



2 **F4 ボタン(Next>)を 2 回押します。**



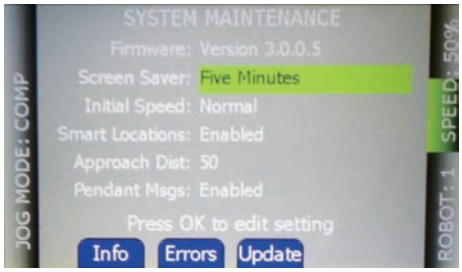
［HOME3］画面が表示されます。



3 **F1 ボタン(Maint)を押します。**

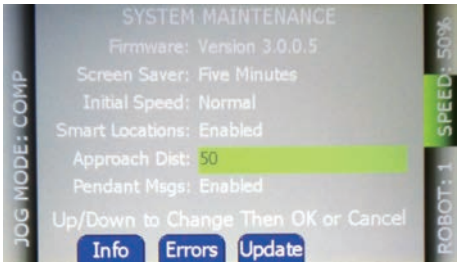


［SYSTEM MAINTENANCE］画面が表示されます。



4 **上／下矢印ボタンを押し、［Approach Dist］フィールドを選択します。**





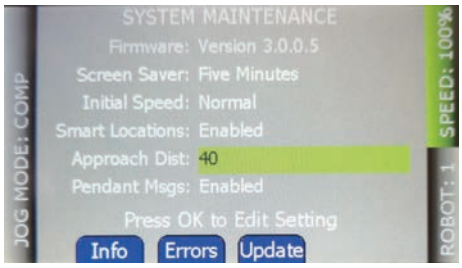
5 **OK ボタンを押します。**



6 **上／下矢印ボタンを押し［Approach Dist］フィールドの値を変更することで、昇降高さを 1～200mm の範囲で選択します。**



ここでは昇降高さの値を 40 にすることとします。



7 **OK ボタンを押します。**



8 **MENU ボタンを押し、［HOME1］画面に戻ります。**



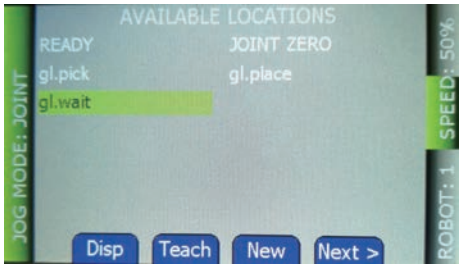
9 **F2 ボタン(Loc)を押します。**



［AVAILABLE LOCATIONS］画面が表示されます。



10 **［AVAILABLE LOCATIONS］画面で矢印ボタンを押し、位置のリストからティーチングする位置を選択します。**



11 **F4 ボタン(Next>)を押します。**



画面下のソフトキー表示が変わります。



12 **手順 10 で選択した変数が「gl.pick」または「gl.place」の場合は、F2 ボタン(Appro)を押し続けます。**



選択した位置の上部までロボットが直線的に移動します。

13 **F1 ボタン(JogTo)を押し続けます。**



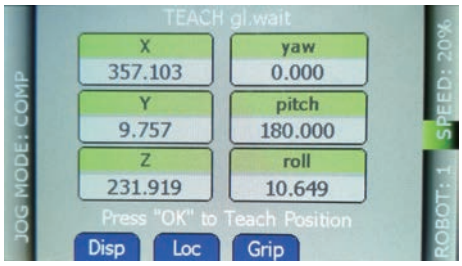
選択した位置までロボットが直線的に移動します。

14 **ロボットが、選択した位置まで移動したら、F1 ボタン(JogTo)を離します。**
移動した位置と姿勢で問題がなければ、手順 20 へ進みます。位置または姿勢を調整する場合は、引き続き手順 15 以降を実施します。

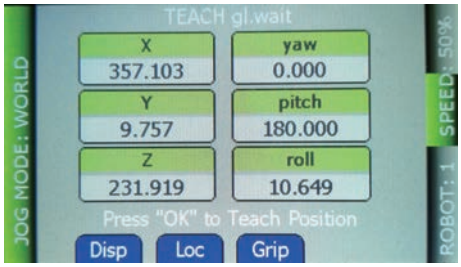
15 **F4 ボタン(Next)を押し、その後 F2 ボタン(Teach)を押します。**



選択している位置の［TEACH］画面が表示されます。



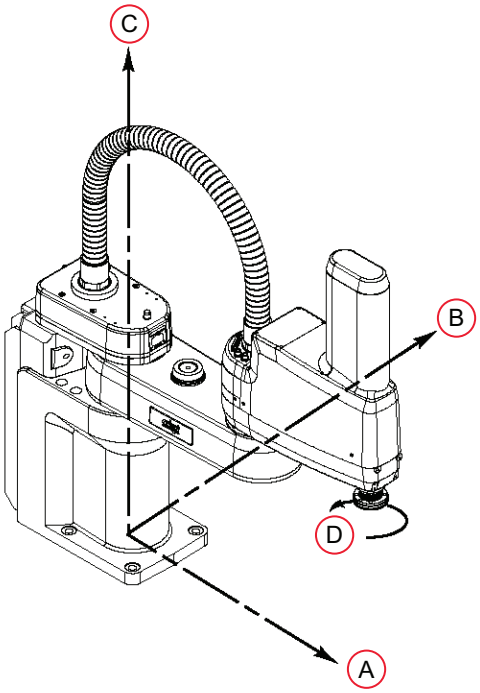
16 **ジョグモード表示が使用したいモードになるまで JOG MODE ボタンを押します。ここではワールドモードで位置の調整を行うとし、［WORLD］になるまでボタンを押します。**



17 **JOINT/AXIS CONTROL ボタンを押して、ロボットを所望の位置と姿勢になるように動かします。**

ワールドモードでの座標軸と JOINT/AXIS CONTROL ボタンの対応は次のとおりです。

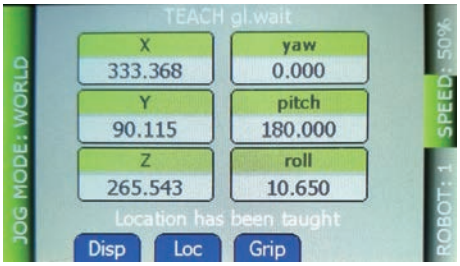
記号	説明	JOINT/AXIS CONTROL ボタン
A	+X（X 方向）	X
B	+Y（Y 方向）	Y
C	+Z（Z 方向）	Z
D	+RZ, CCW（RZ 回転）	RZ



18 **OK ボタンを押します。**



位置の値が、現在のロボットの位置に更新されます。



19 **F2(Loc)ボタンを押し、その後 F4(Next)ボタンを押します。**
［AVAILABLE LOCATIONS］画面に戻ります。



20 **手順 10 で選択した変数が「gl.pick」または「gl.place」の場合は、F2 ボタン(Appro)を押し続けます。**



選択した位置の上部までロボットが直線的に移動します。

21 **調整する位置がまだ残っている場合は手順 10 に戻ります。**

ティーチングした位置を SD メモリカードに保存する

ティーチングした位置のデータを、ロボット統合 CPU ユニットの電源を OFF にした後も保持されるよう SD メモリカードに保存します。

Sysmac Studio のバージョンによって、手順が異なります。Sysmac Studio のバージョンを確認の上、操作してください。

使用上の注意

位置データを SD メモリカードに保存せずに、ロボット統合 CPU ユニットの電源を OFF した場合、調整した位置データが失われ、次回の起動時にロボットが意図しない動作をすることがあります。実機でティーチングした場合は、本手順を実施してください。

● Sysmac Studio Ver.1.54 以降の場合

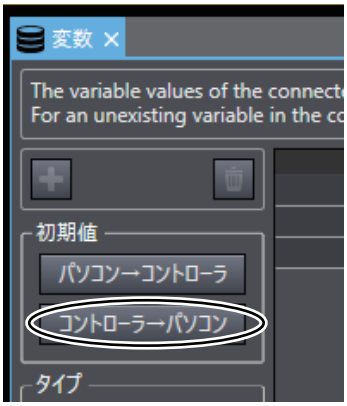
1 **Sysmac Studio のマルチビューエクスプローラのデバイスリストで、［RobotControlSettings］を選択します。**



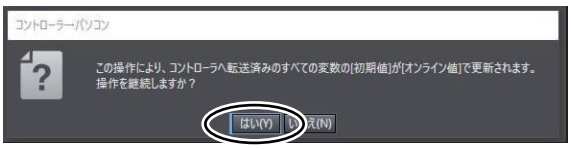
2 マルチビューエクスプローラの［プログラミング］－［V+モジュール］－［変数］を選択します。



3 ［変数］タブの［コントローラ→パソコン］ボタンをクリックします。



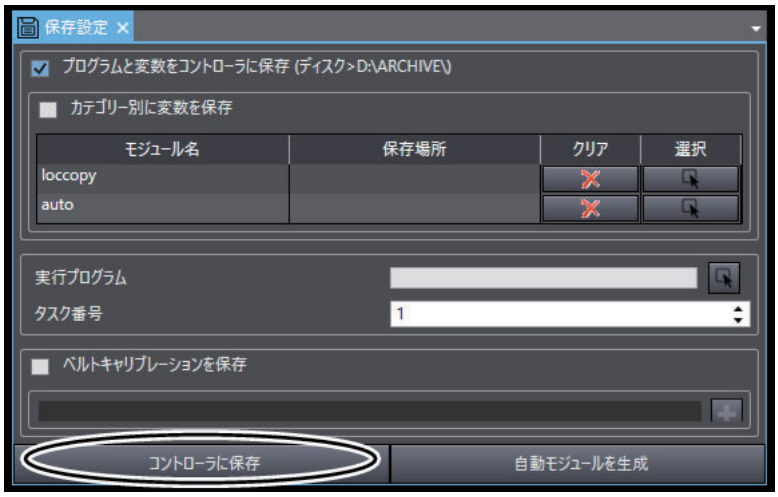
4 ［はい(Y)］ボタンをクリックします。



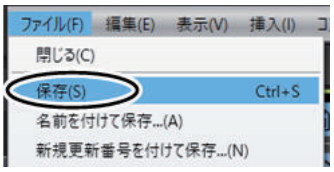
5 マルチビューエクスプローラの［構成・設定］－［保存設定］を選択します。



6 ［保存設定］タブの［コントローラに保存］ボタンをクリックします。



7 メニューバーの［ファイル］-［保存］をクリックします。



● Sysmac Studio Ver.1.53 以前の場合

1 Sysmac Studio のマルチビューエクスプローラのデバイスリストで、[RobotControlSettings] を選択します。



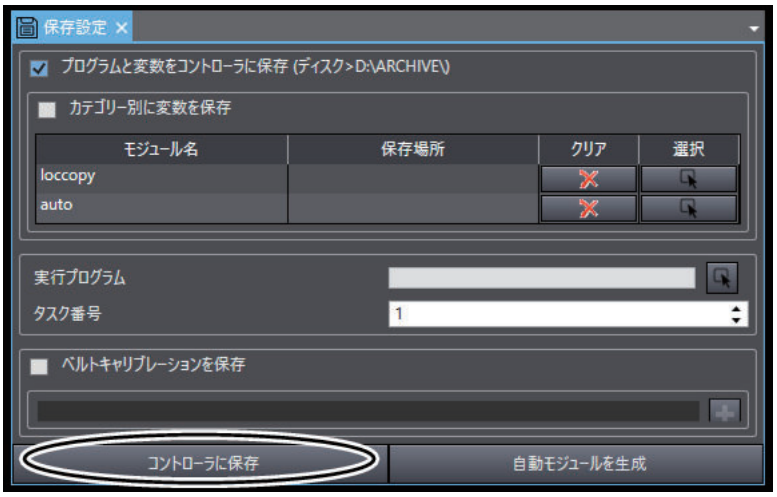
2 マルチビューエクスプローラのツールバーから［V+メモリを取得］アイコンをクリックします。



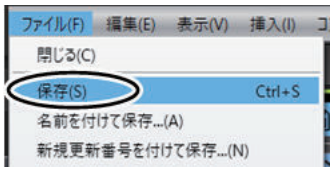
3 **マルチビューエクスプローラの［構成・設定］–［保存設定］を選択します。**



4 **［保存設定］タブの［コントローラに保存］ボタンをクリックします。**



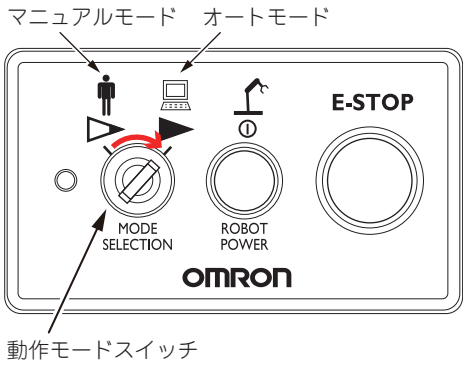
5 **メニューバーの［ファイル］–［保存］をクリックします。**



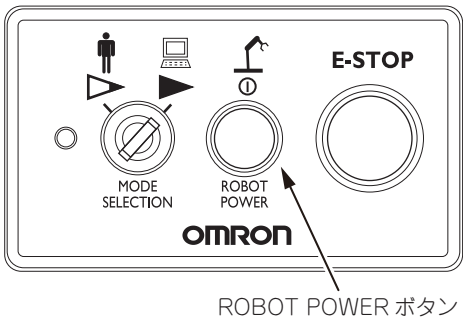
3-5-4 プログラムを実行して動作を確認する

ロボットをプログラムに従って自動運転前の動作の確認をします。

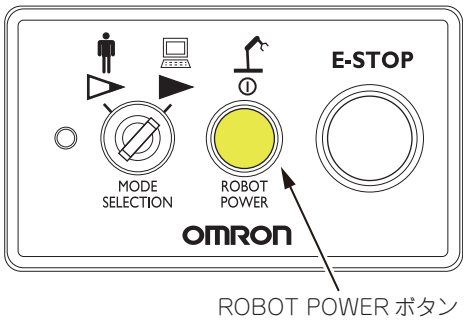
1 **フロントパネルの動作モードスイッチをオートモードにします。**



2 **フロントパネルの ROBOT POWER ボタンを押します。**



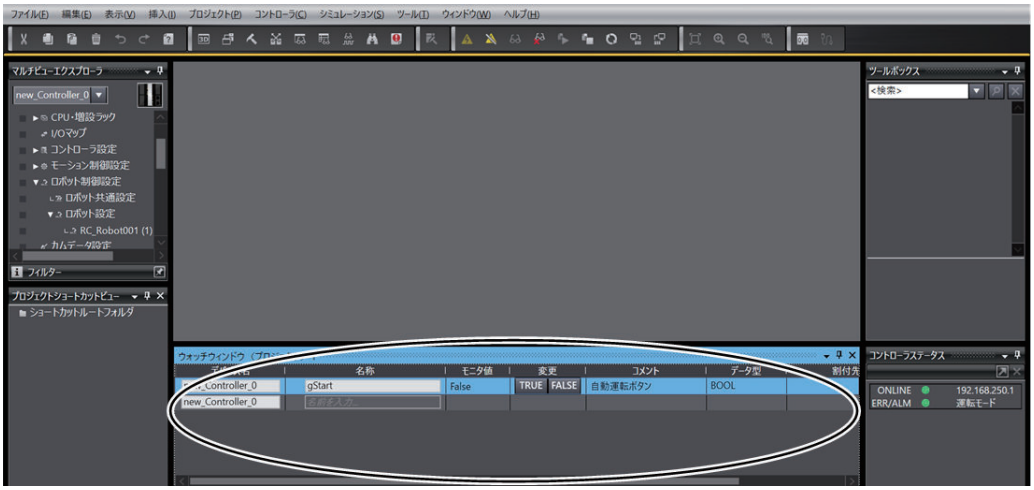
ROBOT POWER ボタンが点滅します。再度 ROBOT POWER ボタンを押すと ROBOT POWER ボタンが点灯し、ロボットのハイパワーが ON になります。



3 **パソコンの Sysmac Studio のメインウィンドウで、メニューバーから［表示］-［ウォッチウィンドウ］を選択します。**



メインウィンドウの下に［ウォッチウィンドウ］画面が表示されます。



4 **［gStart］の［変更］列の［TRUE］ボタンをクリックします。**



ロボットがプログラムに従ってピック&プレースの動作をします。

実際に本プログラムを使用する場合は、表示器から変数「gStart」を操作できるようにしてください。

使用上の注意

V+プログラムで動作を制御している際に、ロボット統合 CPU ユニットを運転モードからプログラムモードに切り替え、さらに運転モードに戻したときは、フロントパネルの ROBOT POWER ボタンを押して消灯を確認し、再度ボタンを押して点灯を確認してから上記の手順を実施してください。

4

動的ピック & プレース装置の実現例

この章では、動的ピック & プレース装置の実現例について説明します。

4-1　動的ピック & プレース装置のプログラム仕様 4-3
4-1-1　プログラム構成 4-3
4-2　基本的な立ち上げ手順 4-12
4-3　プログラミングとシミュレーションの手順 4-15
4-3-1　プロジェクトファイルを新規作成する 4-15
4-3-2　EtherCAT ネットワーク構成を作成する 4-16
4-3-3　V+プログラムを作成する 4-19
4-3-4　シーケンス制御プログラムを作成する 4-23
4-3-5　3D 形状データを配置する 4-30
4-3-6　Pack Manager サンプルを作成する 4-53
4-3-7　C#プログラムを作成する 4-82
4-3-8　V+のプログラムと変数の電源 ON 時の自動読込を設定する 4-84
4-3-9　プログラムを実行して動作を確認する 4-88
4-4　設置、配線する 4-94
4-4-1　ロボット統合 CPU ユニットと EtherCAT デジタル I/O ターミナル間を配線する 4-95
4-4-2　EtherCAT デジタル I/O ターミナルの EtherCAT ノードアドレスを設定する 4-95
4-4-3　EtherCAT デジタル I/O ターミナルとインバータの EtherCAT 通信ユニット間を配線する 4-96
4-4-4　インバータの EtherCAT 通信ユニットの EtherCAT ノードアドレスを設定する 4-96
4-4-5　インバータの EtherCAT 通信ユニットとロボット間を配線する 4-96
4-4-6　ロボットの EtherCAT ノードアドレスを設定する 4-97
4-4-7　ロボットと T20 ペンダント、フロントパネル間を配線する 4-97
4-4-8　ロボットとエンコーダ間を配線する 4-99
4-4-9　ロボット統合 CPU ユニットの EtherNet/IP ポートを配線する 4-100
4-4-10　IPC Application Controller とディスプレイ、カメラ間を配線する 4-101
4-4-11　カメラとロボット間を配線する 4-101
4-4-12　EtherCAT デジタル I/O ターミナルとソレノイドバルブ間を配線する 4-102
4-5　実機での動作確認 4-104
4-5-1　通信設定をする 4-104
4-5-2　オンライン接続する 4-110
4-5-3　設定とプログラムを転送する 4-112
4-5-4　カメラの設定をする 4-115
4-5-5　シーケンス制御プログラム、V+プログラムを実行する 4-126
4-5-6　ロボットのハイパワーを ON する 4-128
4-5-7　ベルトのキャリブレーションを行う 4-130

4-5-8　センサのキャリブレーションを行う 4-135
4-5-9　ロケータを設定する 4-146
4-5-10　アイドル位置、ピック位置、プレース位置をティーチングする 4-151
4-5-11　プログラムを実行して動作を確認する 4-154

4-1 動的ピック & プレース装置のプログラム仕様

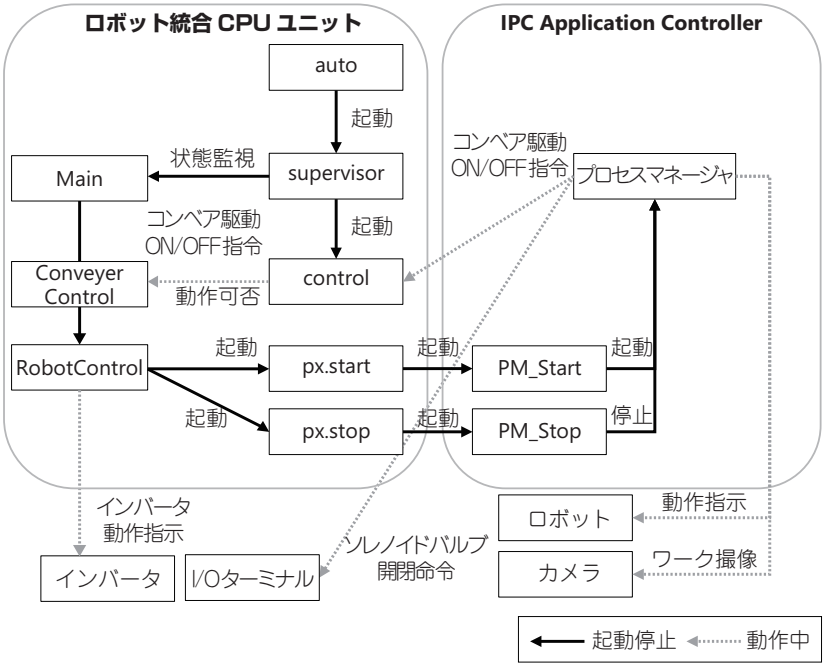
動的ピック & プレース装置のプログラム仕様について説明します。
プログラムの作成方法については、「4-3 プログラミングとシミュレーションの手順 (P.4-15)」の中で説明します。

4-1-1 プログラム構成

動的ピック & プレース装置のプログラム仕様について説明します。
以下のプログラムで構成されます。

プログラム名	言語	説明
Main	シーケンス制御プログラム（ラダー図言語）	メインプログラム
ConveyerControl	シーケンス制御プログラム（ラダー図言語）	コンベア動作プログラム
RobotControl	シーケンス制御プログラム（ラダー図言語）	ロボット自動運転実行プログラム
px.start	V+プログラム	開始用 C#プログラムの呼び出しプログラム
px.stop	V+プログラム	停止用 C#プログラムの呼び出しプログラム
supervisor	V+プログラム	ロボット動作の状態監視プログラム
control	V+プログラム	ロボット動作の定常処理プログラム
auto	V+プログラム	ロボット動作の起動処理プログラム
PM_Start	C#プログラム	プロセスマネージャの開始プログラム
PM_Stop	C#プログラム	プロセスマネージャの停止プログラム

各プログラム、および周辺機器の役割は次の通りです。



本装置では、Pack Manager サンプルを使用してピック & プレース動作を実現します。
Pack Manager サンプルを使用することで、IPC Application Controller のプロセスマネージャ機能により、ロボットを動作させることができます。周辺機器との連動処理が必要となる複雑なプログラムの作成は必要ありません。ただし、プロセスマネージャを開始したり、停止したりするための、V+プログラムおよび C#プログラムを作成する必要があります。

ロボット統合 CPU ユニットが実行する V+プログラムから、IPC Application Controller が実行する C#プログラムを呼び出すことで、プロセスマネージャを開始したり、停止したりすることができます。
プロセスマネージャのソレノイドバルブ開閉命令は、V+デジタル I/O 出力に書き込まれます。これは V+デジタル I/O 割付機能で EtherCAT スレーブの I/O ターミナルの接点に割り当てられ、その接点から出力されます。
プロセスマネージャのコンベア駆動 ON/OFF 命令は、V+デジタル I/O 出力に書き込まれます。V+プログラムの定常処理“control”がコンベア駆動 ON/OFF 命令の状態をポーリングし、共有変数に書き込みます。さらに、シーケンス制御プログラム“ConveyerControl”が共有変数をポーリングし、インバータへの動作指示（ON/OFF 指示、速度指示、および回転方向指示）に変換し、インバータへ出力します。

シーケンス制御プログラム

動的ピック & プレース装置を制御するためのシーケンス制御プログラムについて説明します。
シーケンス制御プログラムでは、グローバル変数を登録し、メインプログラム“Main”と動作プログラム“ConveyerControl”、“RobotControl”の 3 つのプログラムを作成します。

● グローバル変数

シーケンス制御プログラムで使用するグローバル変数を登録します。

次の変数は、I/O マップ画面で変数を自動生成します。

名称	データ型	コメント
E002_Command	WORD	インバータ動作指令
E002_Frequency_reference	UINT	インバータ目標周波数
E002_Status	WORD	インバータステータス
E002_Output_frequency_monitor	UINT	インバータ周波数モニタ
E002_Sysmac_Error_Status	BYTE	インバータ Sysmac 異常

次の変数は、手動で追加登録します。

名称	データ型	コメント
e_conveyor_dir	BOOL	コンベア逆転指示
e_conveyor_run	BOOL	コンベア運転指示
e_conveyor_spd	BOOL	コンベア高速指示
e_is_run_mode	BOOL	運転モード
gRC_Err	BOOL	ロボット制御エラー
gReset	BOOL	自動運転停止
gStart	BOOL	自動運転開始
gSysOK	BOOL	運転可能
eBool_exeT1	BOOL	supervisor 実行中
gEnableT1	BOOL	supervisor 実行可能

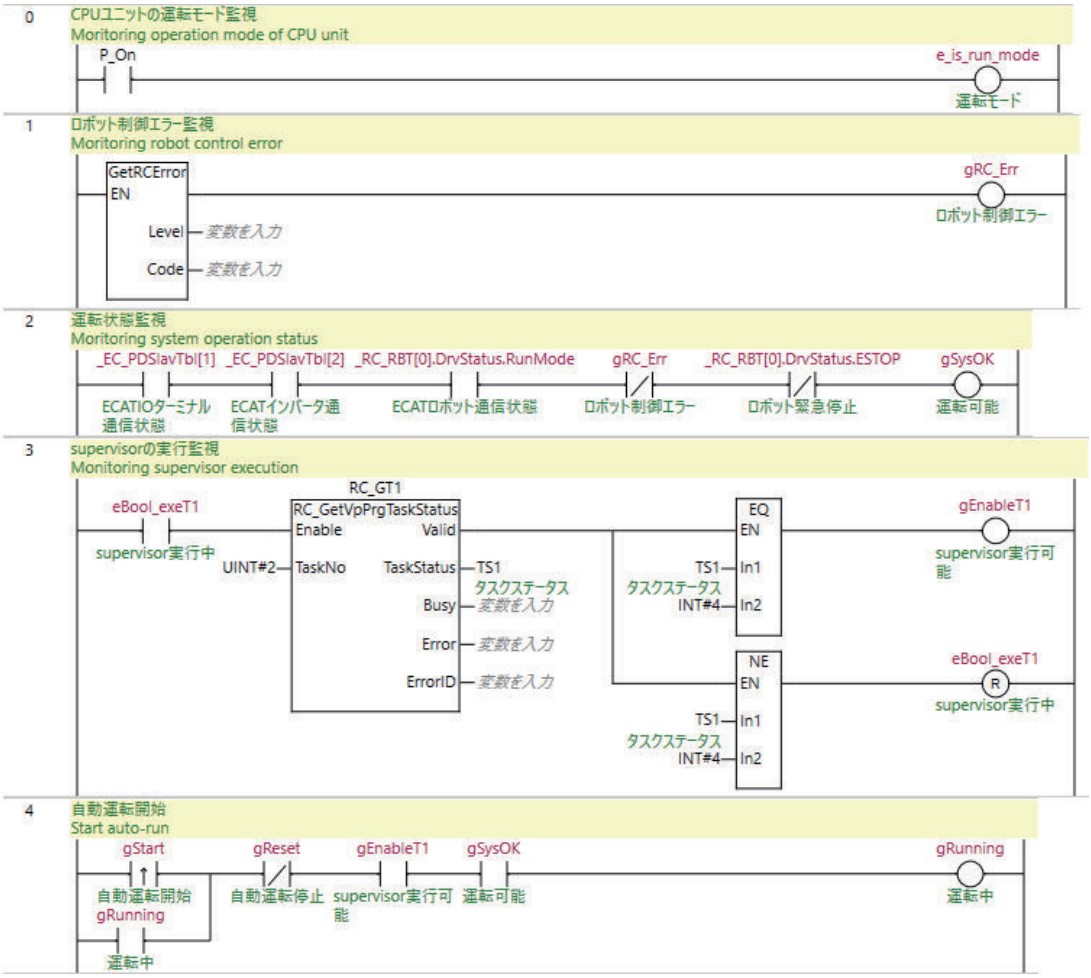
次の変数は、構成にロボットを追加すると自動的に生成されます。

名称	データ型	コメント
RC_Robot001	_sRC_RBT_REF	iX4 650H
RC_Robot001_IO	_sRC_RBT_IO_REF	iX4 650H I/O

● メインプログラム(Main)

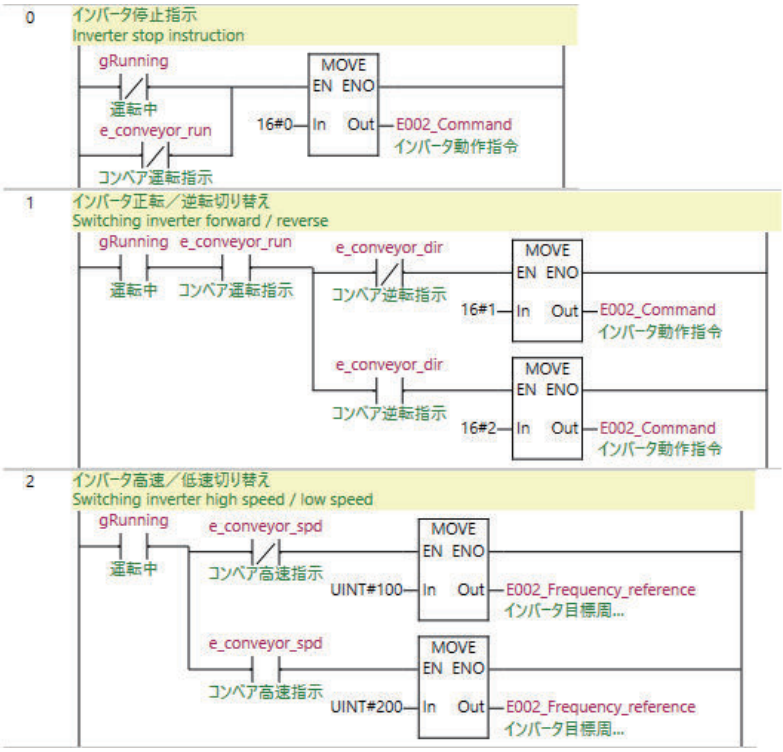
メインプログラム“Main”では、ロボットの状態を監視し、自動運転の開始可否を判断します。

以下にプログラムを記載します。



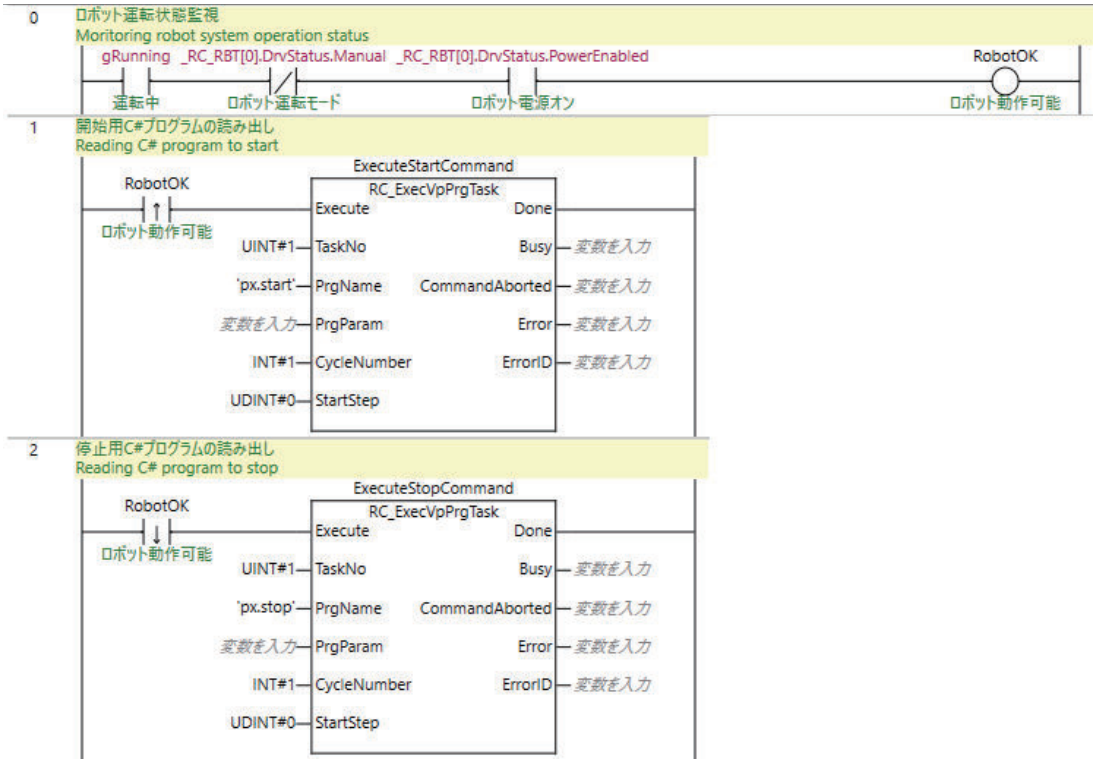
● 動作プログラム(ConveyerControl)

動作プログラム“ConveyerControl”では、指示を受けてインバータの運転を実行します。自動運転中は、プロセスマネージャからのベルトコンベア駆動 ON/OFF 指示に従い、インバータへの ON/OFF 命令を実行します。
以下にプログラムを記載します。



● 動作プログラム(RobotControl)

動作プログラム“RobotControl”では、指示を受けて自動運転を実行します。
以下にプログラムを記載します。



V+プログラム

動的ピック & プレース装置を制御するための V+プログラムについて説明します。
V+プログラムでは、プロセスマネージャ開始に使用する C#プログラムを呼び出すためのプログラム“px.start”、プロセスマネージャ停止に使用する C#プログラムを呼び出すためのプログラム“px.stop”、状態監視プログラム“supervisor”、定常処理プログラム“control”および起動処理プログラム“auto”の 5 つのプログラムを作成します。

● V+デジタル I/O

V+デジタル I/O を介し、複数の V+プログラム間あるいは V+プログラムと外部の EtherCAT デジタル I/O ターミナル間の通信を行います。
使用する I/O 番号を以下に示します。番号 4001 は、V+デジタル I/O 割付機能を使用して設定します。

範囲		番号	用途
1-96	出力	未使用	
1001-1096	入力	1001	ラッチ番号
2001-2999	ソフトウェアシグナル	2001	ベルトコンベア 運転
		2002	ベルトコンベア 高速・低速
		2003	ベルトコンベア 後退
3001-3004	ロボットシグナル	未使用	
4001-4032	外部	4001	ソレノイドバルブ 開 (GX-MD3218 OUT0)

● 開始用 C#プログラムの呼び出しプログラム(px.start)

プロセスマネージャを開始する C#プログラムを呼び出します。
以下にプログラムを示します。

参考

以下のプログラムは、Sysmac Studio Ver.1.42 では使用できません。Sysmac Studio Ver.1.43 以降で使用できます。
Sysmac Studio Ver.1.42 では、Sysmac Studio を IPC Application Controller にオンライン接続し、［タスクステータスコントロール］画面で、［ApplicationManager1］－［C#プログラム］－［PM_Start］を選択した後、［開始］ボタンをクリックしてください。




```
.PROGRAM px.start()
```

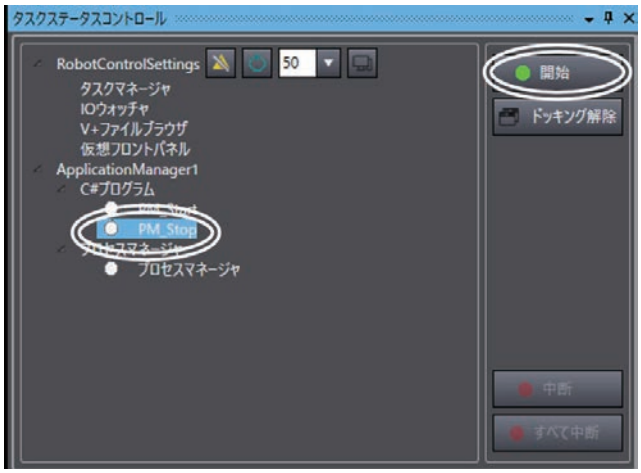
```
        GLOBAL $ip
        AUTO REAL status
        TYPE "Start PM_Start"
        CALL rm.execute3($ip, "/ApplicationManager1/PM_Start", "Execute", 0, $args[], 3, status)
        IF status < 0 THEN
            TYPE "Failed to execute: error code ", status
        END
.END
```

● 停止用 C#プログラムの呼び出しプログラム(px.stop)

プロセスマネージャを停止する C#プログラムを呼び出します。
以下にプログラムを示します。

参考

以下のプログラムは、Sysmac Studio Ver.1.42 では使用できません。Sysmac Studio Ver.1.43 以降で使用できます。
Sysmac Studio Ver.1.42 では、Sysmac Studio を IPC Application Controller にオンライン接続し、[タスクステータスコントロール] 画面で、[ApplicationManager1] - [C#プログラム] - [PM_Stop] を選択した後、[開始] ボタンをクリックしてください。




```
.PROGRAM px.stop()
        GLOBAL $ip
        AUTO REAL status
        TYPE "Start PM_Stop"
        CALL rm.execute3($ip, "/ApplicationManager1/PM_Stop", "Execute", 0, $args[], 3, status)
        IF status < 0 THEN
            TYPE "Failed to execute: error code ", status
        END
.END
```

● 状態監視プログラム(supervisor)

シーケンス制御が運転モードの間、定常処理“control”が異常停止しないよう監視します。また、シーケンス制御がプログラムモードになったことを検出し、定常処理“control”と Application Manager を停止します。
以下にプログラムを示します。

変数 ebool_exet1 は、V+プログラムが実行可能であることを示すフラグです。シーケンス制御プログラムでこの変数を監視し、V+プログラムが実行可能になっているときにシーケンス制御プログラムで自動運転を実行します。

```
.PROGRAM supervisor()
        EXTERNAL e_is_run_mode, ebool_exet1
        AUTO REAL p_is_run_mode
        GLOBAL $ip
        $ip = "127.0.0.1"
        p_is_run_mode = e_is_run_mode
        WHILE TRUE DO
            IF e_is_run_mode THEN
                IF TASK(1,3) <> 4 THEN
                    EXECUTE 3 control()
                END
                ebool_exet1 = TRUE
            ELSE
                IF TASK(1,3) <> 0 THEN
                    ABORT 3
                    CYCLE.END 3
                    KILL 3
                END
                IF p_is_run_mode THEN
                    BRAKE
                    CALL px.stop()
                END
            END
            p_is_run_mode = e_is_run_mode
            WAIT
        END
.END
```

● 定常処理プログラム(control)

シーケンス制御が運転モードの間、定常的に行う処理です。失敗時も自動的に再起動されます。定常的な処理とは、V+デジタル I/O 信号の値をシーケンス制御との共通変数に書き込む処理です。以下にプログラムを示します。

```
.PROGRAM control()
        EXTERNAL e_conveyor_run
        EXTERNAL e_conveyor_spd
        EXTERNAL e_conveyor_dir
        WHILE TRUE DO
            e_conveyor_run = SIG(2001)
            e_conveyor_spd = SIG(2002)
            e_conveyor_dir = SIG(2003)
            RELEASE
        END
.END
```

● 起動処理プログラム(auto)

V+プログラムおよびグローバル変数を SD メモリカードからメインメモリに読み出し、状態監視プログラムを起動します。
本プログラムは自動生成されるため、プログラムを作成する必要はありません。

C#プログラム

動的ピック & プレース装置を制御するための C#プログラムについて説明します。

● プロセスマネージャの開始プログラム(PM_Start)

プロセスマネージャを開始します。
以下にプログラムを示します。

```
using Ace.Server.Xpert.PackXpert;
using Ace.Services.NameLookup;
using Ace.Client.ApplicationManager.AutoConnect;
using Ace.Server;
using System;
using System.Collections.Generic;
using System.Diagnostics;

namespace Ace.Custom {
     public class Program {
          public INameLookupService ace;
          public void Main() {
               Trace.WriteLine("Script Starting");
               IProcessManager processManager = (IProcessManager) ace[
                   "/ApplicationManager1/プロセスマネージャ"];
               processManager.Start();
          }
     }
}
```

● プロセスマネージャの停止プログラム(PM_Stop)

プロセスマネージャを停止します。
以下にプログラムを示します。

```
using Ace.Server.Xpert.PackXpert;
using Ace.Services.NameLookup;
using Ace.Client.ApplicationManager.AutoConnect;
using Ace.Server;
using System;
using System.Collections.Generic;
using System.Diagnostics;

namespace Ace.Custom {
     public class Program {
          public INameLookupService ace;
          public void Main() {
               Trace.WriteLine("Script Starting");
```

```
            IProcessManager processManager = (IProcessManager) ace[
                "/ApplicationManager1/プロセスマネージャ"];
            processManager.Stop();
        }
    }
}
```

使用上の注意

各 C#プログラムに記載された文字列“/ApplicationManager1/プロセスマネージャ”は、プロジェクトファイルを作成したときの使用言語により、内容が異なります。[マルチビューエクスプローラ] ウィンドウで、[ApplicationManager1] の [構成・設定] - [プロセス] - [プロセスマネージャ] をドラッグし、C#エディタ上にドロップすることで適切な文字列が自動的に選択されます。

4-2 基本的な立ち上げ手順

動的ピック & プレース装置を実現するための基本的な立ち上げ手順を説明します。
装置の動作をプログラミングとシミュレーションで設定、確認した後に、実機で動作を確認し微調整します。

番号	手順		説明	参照
1	プログラミングとシミュレーションの手順	プロジェクト作成	Sysmac Studio でプロジェクトファイルを新規に作成します。	P.4-15
2		ネットワーク構成作成	EtherCAT ネットワーク構成を作成し、EtherCAT デジタル I/O ターミナル、インバータおよびロボットを追加します。	P.4-16
3		V+プログラミング	V+プログラムを作成します。	P.4-19
4		シーケンス制御プログラム	シーケンス制御プログラムを作成します。	P.4-23
5		3D 形状データ配置	3D 形状データを配置し、ロボットの動作範囲を 3D 表示画面上で見えるようにします。	P.4-30
6		Pack Manager サンプルの作成	シミュレーション機能を使用し、ロボット設定やカメラ設定、コンベア設定などを設定した上で、ティーチングを行います。	P.4-53
7		C#プログラミング	C#プログラムを作成します。	P.4-82
8		プログラム実行（シミュレーション）	Sysmac Studio のシミュレーションを使用し、シーケンス制御プログラムを実行します。	P.4-88

番号	手順		説明	参照
9	設置、配線する	ロボット統合 CPU ユニットと EtherCAT デジタル I/O ターミナル間を配線する	ロボット統合 CPU ユニットと EtherCAT デジタル I/O ターミナル間を配線します。	P.4-95
10		EtherCAT デジタル I/O ターミナルの EtherCAT ノードアドレスを設定する	EtherCAT デジタル I/O ターミナルの EtherCAT ノードアドレスを設定します。	P.4-95
11		EtherCAT デジタル I/O ターミナルとインバータの EtherCAT 通信ユニット間を配線する	EtherCAT デジタル I/O ターミナルとインバータの EtherCAT 通信ユニット間を配線します。	P.4-96
12		インバータの EtherCAT 通信ユニットの EtherCAT ノードアドレスを設定する	インバータの EtherCAT 通信ユニットの EtherCAT ノードアドレスを設定します。	P.4-96
13		インバータの EtherCAT 通信ユニットとロボット間を配線する	インバータの EtherCAT 通信ユニットとロボット間を配線します。	P.4-96
14		ロボットの EtherCAT ノードアドレスを設定する	ロボットの EtherCAT ノードアドレスを設定します。	P.4-97
15		ロボットと T20 ペンダント、フロントパネル間を配線する	ロボットと T20 ペンダント間、およびロボットとフロントパネル間を配線します。配線には、XSYSTEM ケーブルアセンブリ、T20 アダプタケーブル、フロントパネルケーブルを使用します。	P.4-97
16		ロボットとエンコーダ間を配線する	ロボットとエンコーダ間を配線します。	P.4-99
17		ロボット統合 CPU ユニットの EtherNet/IP ポートを配線する	ロボット統合 CPU ユニットと産業用スイッチングハブ間、産業用スイッチングハブとパソコン間、および産業用スイッチングハブと IPC Application Controller 間を配線します。	P.4-100
18		IPC Application Controller とディスプレイ、カメラ間を配線する	IPC Application Controller とディスプレイ間、および IPC Application Controller とカメラ間を配線します。	P.4-101
19		カメラとロボット間を配線する	カメラとロボット間を配線します。	P.4-101
20		EtherCAT デジタル I/O ターミナルとソレノイドバルブ間を配線する	EtherCAT デジタル I/O ターミナルとソレノイドバルブ間を配線します。	P.4-102

番号	手順		説明	参照
21	実機での動作確認	通信設定	パソコン上の Sysmac Studio と実機を接続するための設定をします。	P.4-104
22		オンライン接続	ロボット統合 CPU ユニットと Sysmac Studio をオンライン接続します。	P.4-110
23		設定とプログラムの転送	Sysmac Studio のプロジェクトファイル内の設定およびプログラムをロボット統合 CPU ユニットおよび IPC Application Controller に転送します。	P.4-112
24		カメラの設定	カメラの設定をします。	P.4-115
25		シーケンス制御プログラム、V+プログラムの実行	シーケンス制御プログラム、V+プログラムを実行します。	「4-5-5 シーケンス制御プログラム、V+プログラムを実行する (P.4-126)」
26		ロボットのハイパワー ON	ロボットのハイパワーを ON します。	「ロボットのハイパワーを ON する (P.3-89)」
27		ベルトのキャリブレーション	ベルトをキャリブレーションします。	「4-5-7 ベルトのキャリブレーションを行う (P.4-130)」
28		センサのキャリブレーション	センサをキャリブレーションします。	「4-5-8 センサのキャリブレーションを行う (P.4-135)」
29		ロケータの設定	ロケータを設定します。	「4-5-9 ロケータを設定する (P.4-146)」
30		アイドル位置、ピック位置、プレース位置のティーチング（実機）	実際にロボットを操作して、ティーチンデータを確認、修正します。	P.4-151
31		プログラム実行（実機）	システムをプログラムに従って自動運転前の動作の確認をします。	P.4-154

4-3 プログラミングとシミュレーションの手順

プロジェクトファイルの作成、プログラミング、シミュレーションでの設定と動作確認の手順を説明します。
シミュレーションを利用することで、実機がなくても装置の動作を事前に確認することが可能です。

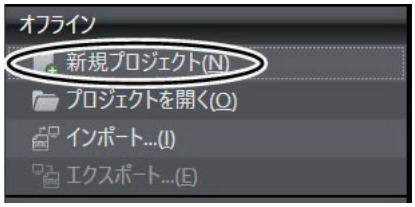
4-3-1 プロジェクトファイルを新規作成する

Sysmac Studio で新規プロジェクトを作成します。

参考

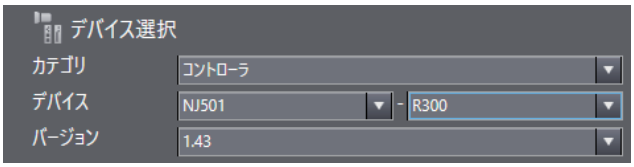
関連動画
- プロジェクトファイルの新規作成
  https://www.fa.omron.co.jp/video/detail/6264519018001/
  Sysmac Studio でロボットを EtherCAT 構成に含む新規プロジェクトを作成する手順を説明しています。

1 **Sysmac Studio を起動し、スタートページで［新規プロジェクト］をクリックします。**



［プロジェクトのプロパティ］画面が表示されます。

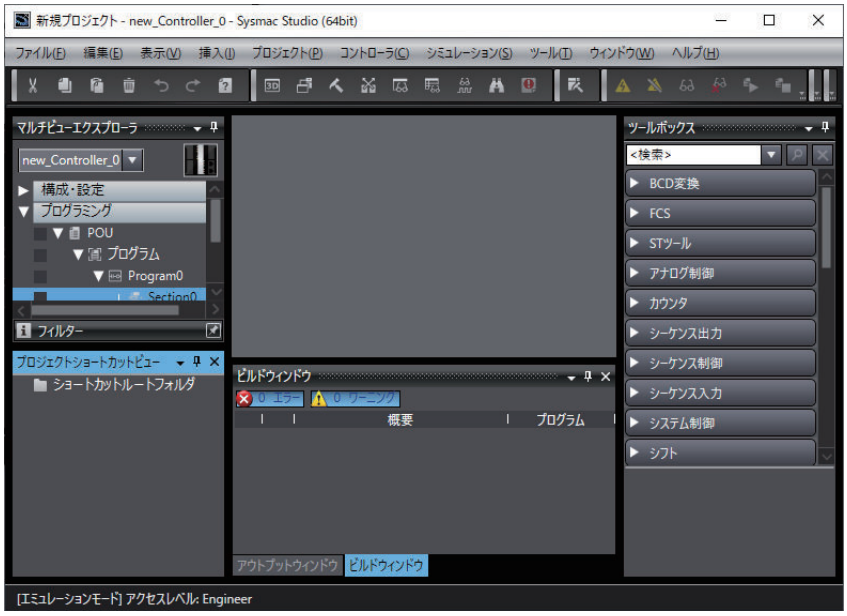
2 **［プロジェクトのプロパティ］画面で、［プロジェクト名］、［作成者］、［コメント］を入力し、デバイス選択の［カテゴリ］、［デバイス］、［バージョン］を以下の画面に示すとおりに選択します。**



3 **［エミュレーションモードで開く］チェックボックスにチェックを入れ、［作成］ボタンをクリックします。**



プロジェクトファイルが作成され、次の画面が表示されます。



4-3-2 EtherCAT ネットワーク構成を作成する

EtherCAT ネットワーク構成を作成し、EtherCAT デジタル I/O ターミナル、インバータおよびロボットを追加します。

参考

関連動画

- プロジェクトファイルの新規作成
  https://www.fa.omron.co.jp/video/detail/62645190180001/
  Sysmac Studio でロボットを EtherCAT 構成に含む新規プロジェクトを作成する手順を説明しています。

1 **マルチビューエクスプローラの［構成・設定］をクリックします。**



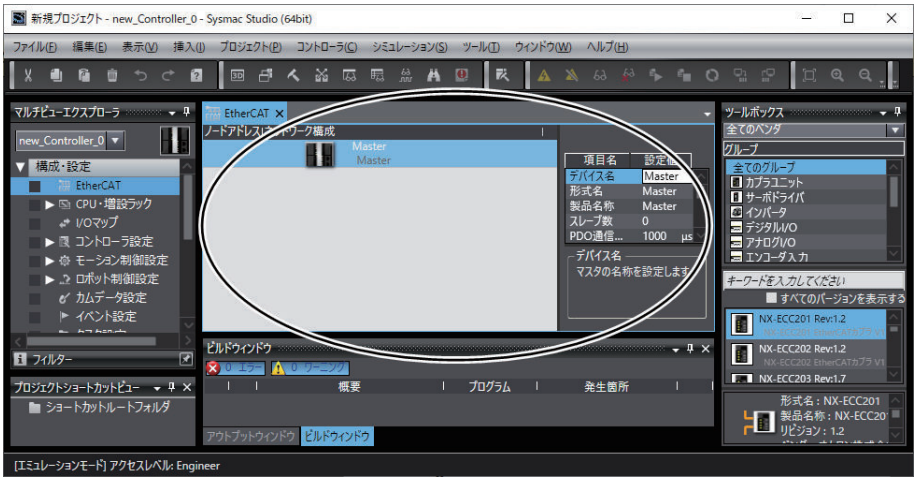
［構成・設定］の下にツリーが展開されます。



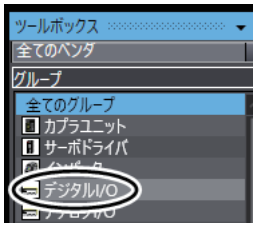
2 **マルチビューエクスプローラから［構成・設定］–［EtherCAT］をダブルクリックします。または［構成・設定］–［EtherCAT］を右クリックして［編集］を選択します。**



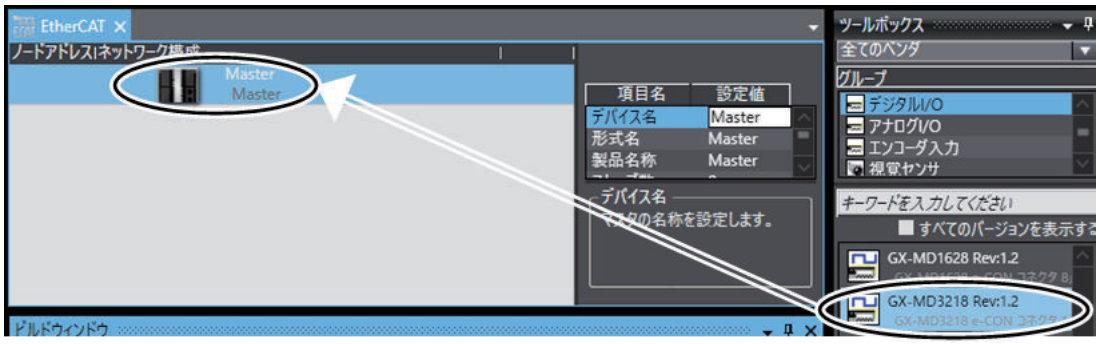
EtherCAT 構成編集画面に EtherCAT マスタが表示されます。



3 **［ツールボックス］のグループで［デジタル I/O］をクリックします。**



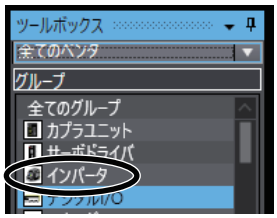
4 **使用するデジタル I/O をドラッグし、EtherCAT 構成編集画面のマスタにドロップします。この例では、［GX-MD3218］を使用します。**



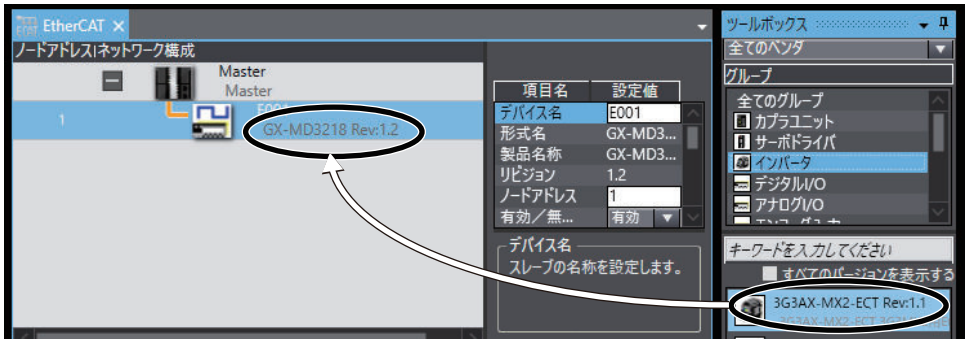
マスタの下に［GX-MD3218］が追加されます。



5 **［ツールボックス］のグループで［インバータ］をクリックします。**



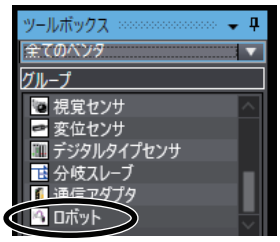
6 **使用するインバータをドラッグし、EtherCAT 構成編集画面の［GX-MD3218］にドロップします。この例では、［3G3AX-MX2-ECT Rev1.1］を使用します。**



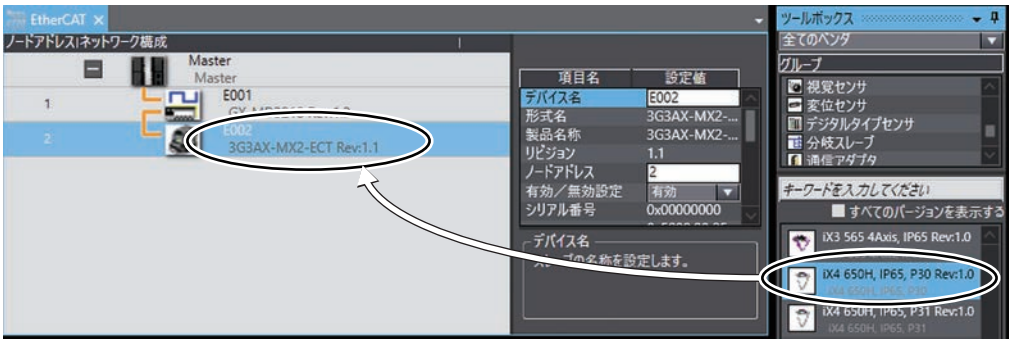
［GX-MD3218］の下に［3G3AX-MX2-ECT Rev1.1］が追加されます。



7 **［ツールボックス］のグループで［ロボット］をクリックします。**



8 **使用するロボットをドラッグし、EtherCAT 構成編集画面の［3G3AX-MX2-ECT Rev1.1］にドロップします。この例では、［iX4 650H, IP65, P30］を使用します。**



［EtherCAT］タブの［3G3AX-MX2-ECT Rev1.1］の下に［iX4 650H, IP65, P30］が追加されます。



4-3-3　V+プログラムを作成する

V+プログラムのプログラミング手順について説明します。
作成するプログラム、および V+デジタル I/O 割付については、「V+プログラム (P.4-7)」を参照してください。

参考

関連動画
- V+プログラムの作成
  https://www.fa.omron.co.jp/video/detail/6294915156001/
  V+プログラムの作成手順を説明しています。

V+デジタル I/O 割付を設定する

V+プログラムで EtherCAT スレーブを制御するために、V+デジタル I/O 割付設定を行います。

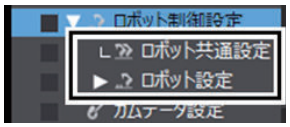
1 **マルチビューエクスプローラのデバイスで、[new_Controller_0] を選択します。**



2 **マルチビューエクスプローラの［ロボット制御設定］の左側の▶アイコンをクリックします。**



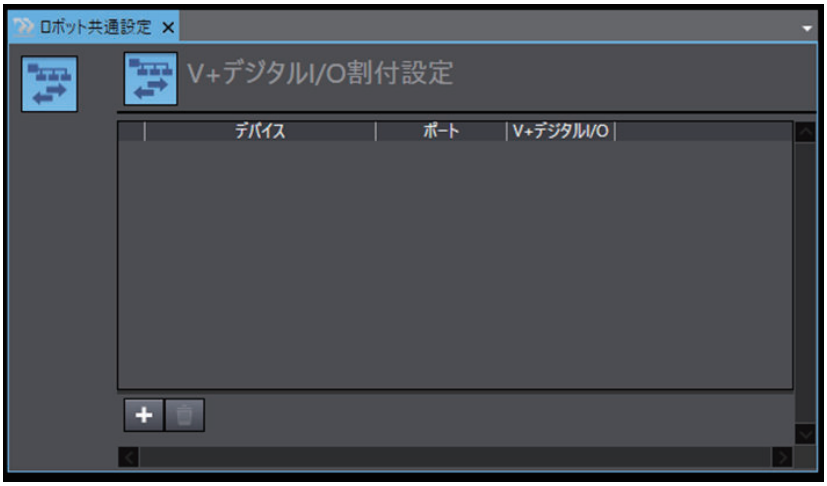
以下のツリーが表示されます。



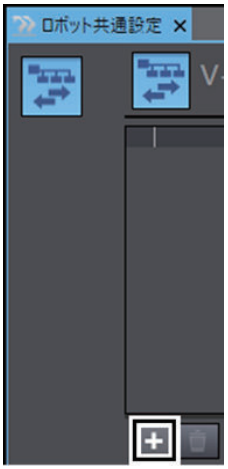
3 **[ロボット共通設定] をダブルクリックします。**



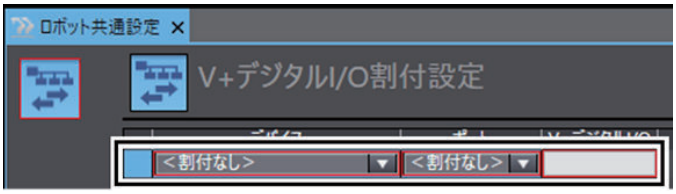
［ロボット共通設定］タブが表示されます。



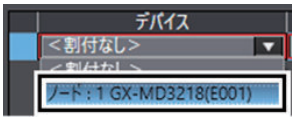
4 **［ロボット共通設定］タブの［+］ボタンをクリックします。**



新規設定のための行が追加されます。



5 **追加された行の［デバイス］欄のドロップダウンリストをクリックし、［ノード : 1 GX-MD3218(E001)］を選択します。**



6 **［ポート］欄のドロップダウンリストをクリックし，［Out Bit00］を選択します。**



7 **［V+デジタル I/O］欄に、「4001」と入力します。**



V+プログラムを作成する

V+プログラムを作成する手順を説明します。

1 **マルチビューエクスプローラのデバイスは、［RobotControlSettings］を選択します。**



2 **マルチビューエクスプローラの［プログラミング］－［V+モジュール］を右クリックし、［新規モジュールを追加］をクリックします。**



［V+モジュール］に［program0］－［program0］が追加され、V+プログラム名を編集できる状態になります。



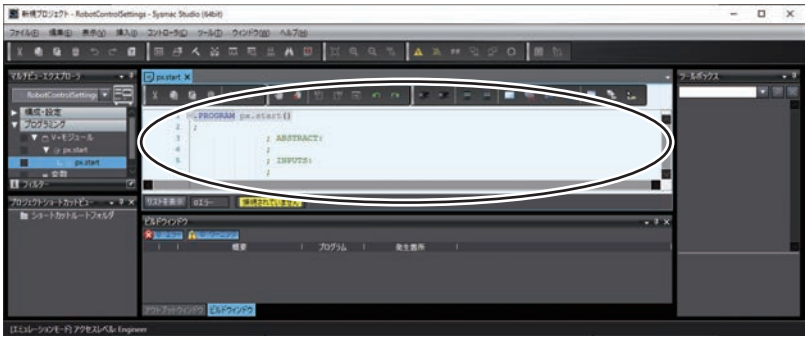
3 **V+プログラム名を入力します。**
V+プログラム名が変更されます。



4 **編集するプログラムをダブルクリックします。**



エディットウィンドウに編集画面が表示され、V+プログラムの作成を行うことができます。



5 **編集画面で、プログラムを入力します。**
作成するプログラムについては、「V+プログラム (P.4-7)」を参照してください。

6 **手順 2 に戻り、同様に繰り返してすべての V+プログラムを作成します。**

4-3-4 シーケンス制御プログラムを作成する

シーケンス制御プログラムのプログラミング手順について説明します。
作成するプログラム、および V+デジタル I/O 割付については、「シーケンス制御プログラム (P.4-4)」を参照してください。

参考

関連動画
- シーケンス制御プログラムの作成
  https://www.fa.omron.co.jp/video/detail/6264519413001/
  Sysmac Studio でシーケンス制御プログラムを作成する手順を説明しています。

デバイス変数を作成する

インバータを制御するためのデバイス変数を作成します。

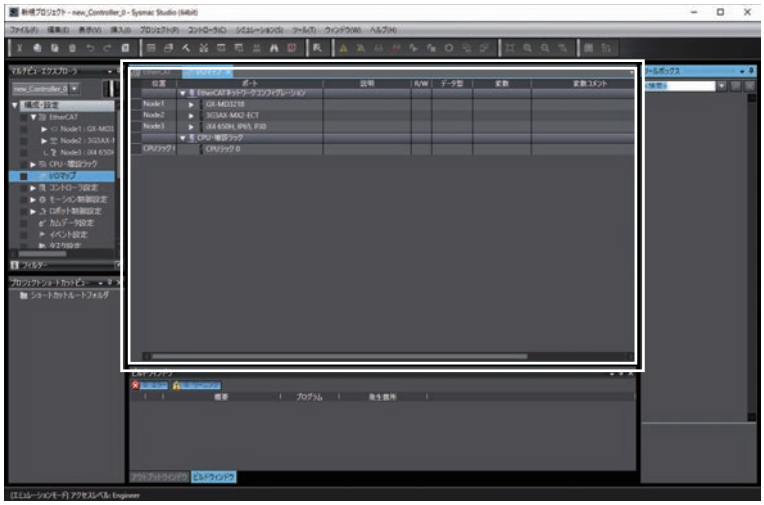
1 **マルチビューエクスプローラのデバイスは、[new_Controller_0] を選択します。**



2 **マルチビューエクスプローラで［構成・設定］–［I/O マップ］をダブルクリックします。**



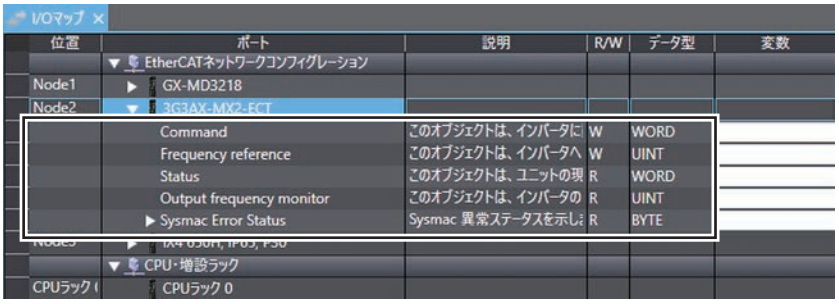
メインウィンドウ画面中央のエディットウィンドウに［I/O マップ］タブが表示されます。



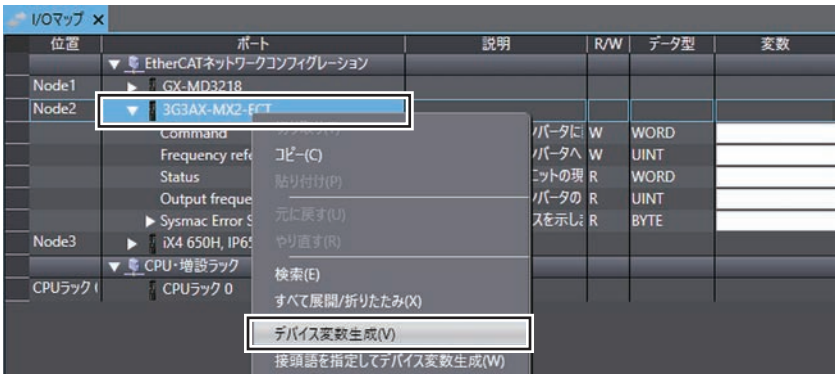
3 **［I/O マップ］タブの［ポート］行にある［3G3AX-MX2-ECT］の左側の▶アイコンをクリックします。**



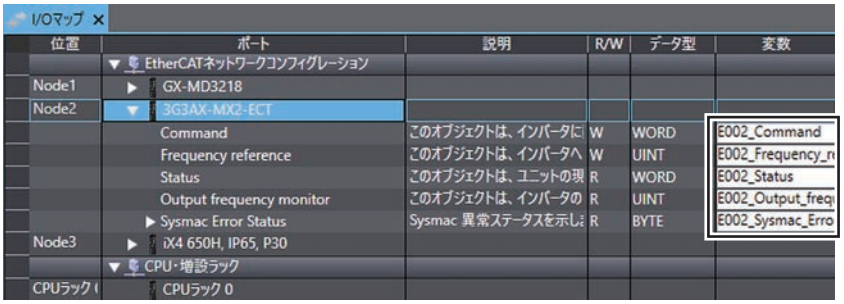
［3G3AX-MX2-ECT］以下のツリーが展開されます。



4 **[I/O マップ] タブの [ポート] 行にある [3G3AX-MX2-ECT] を右クリックし，[デバイス変数生成] を選択します。**



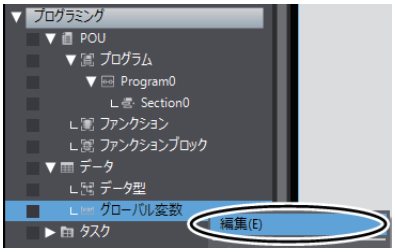
デバイス変数が生成されます。



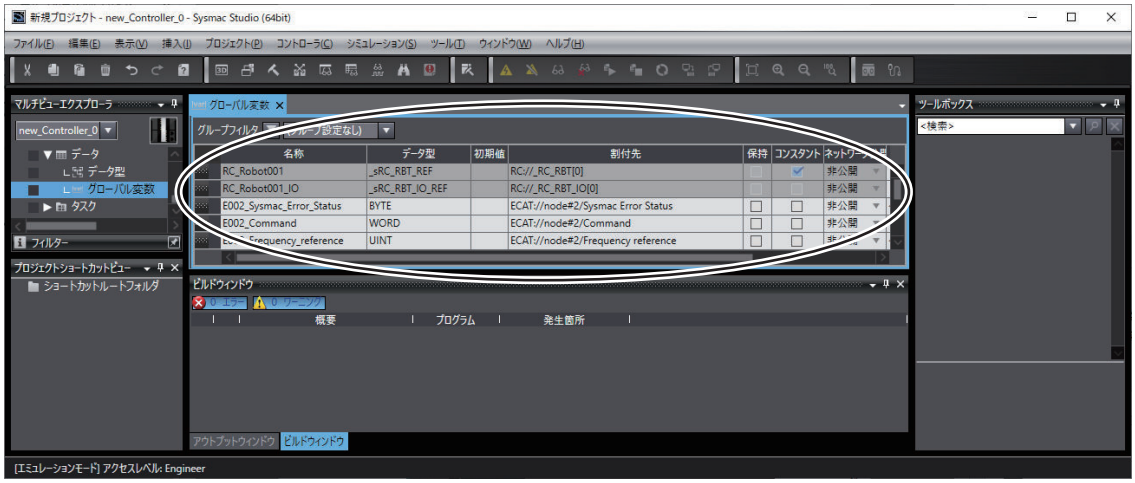
グローバル変数を定義する

シーケンス制御プログラムで位置データとして使用するグローバル変数を定義します。

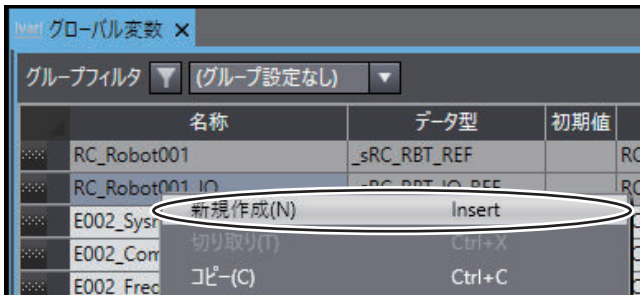
1 **マルチビューエクスプローラで [プログラミング] - [データ] - [グローバル変数] をダブルクリックします。または、[プログラミング] - [データ] - [グローバル変数] を右クリックし、メニューから [編集] を選択します。**



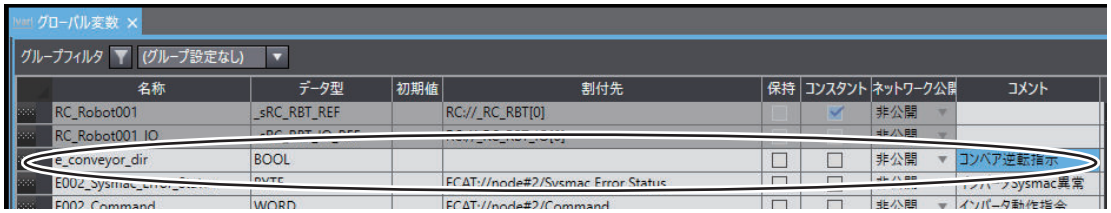
グローバル変数の登録テーブルが表示されます。



2 **グローバル変数テーブル内で［Insert］キーを押すか、または右クリックしてメニューから［新規作成］を選択します。**



3 **各項目を入力または選択し、［Enter］キーを押して確定します。**



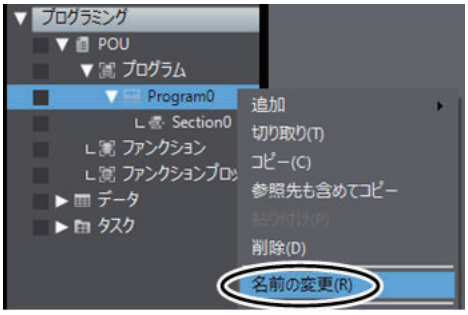
登録するグローバル変数については、「シーケンス制御プログラム (P.4-4)」を参照してください。

グローバル変数が登録されます。

ラダー図プログラムを作成する

ラダー図プログラムを作成する手順を説明します。

1 **マルチビューエクスプローラの［プログラミング］－［POU］－［プログラム］－［Program0］を右クリックし、［名前の変更］を選択します。**



シーケンス制御プログラム名を編集できる状態になります。



2 **シーケンス制御プログラム名を入力します。**

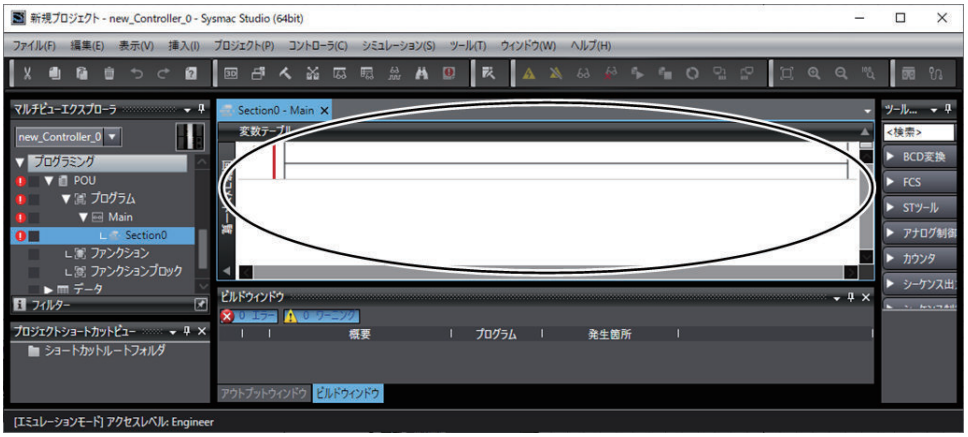
シーケンス制御プログラム名が変更されます。



3 **編集するセクションをダブルクリックします。**



エディットウィンドウに変数テーブルとラダーエディタが表示されます。



4 **ラダーエディタにプログラムを入力します。内部変数、外部変数はプログラム入力時に自動で登録されます。**

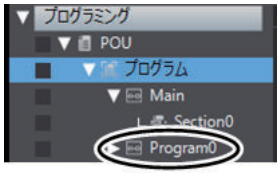
作成するプログラムについては、「シーケンス制御プログラム (P.4-4)」を参照してください。

シーケンス制御プログラムを複数作成する場合、以降の操作を行います。

5 **マルチビューエクスプローラの［プログラミング］－［POU］－［プログラム］を右クリックし、［追加］－［ラダー］を選択します。**



［プログラム］に［Program0］が追加されます。



6 **マルチビューエクスプローラの［プログラミング］－［POU］－［プログラム］－［Program0］の左に表示されたアイコンをクリックします。**



［Program0］の下に［Section0］が展開されます。



7 **操作 1.に戻り、同様に繰り返します。**

プログラムをタスクに割り付ける

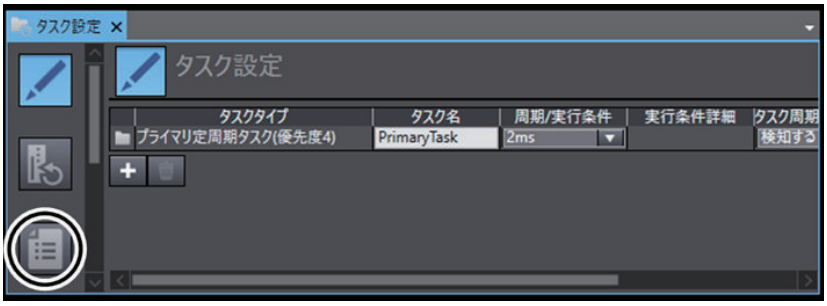
複数のラダー図プログラムをロボット統合 CPU ユニットのタスクに割り付ける手順を説明します。
本ガイドでは、動的ピック & プレース装置をシーケンス制御プログラムで制御する場合に複数のラダー図プログラムを使用しているため、プログラムをタスクに割り付ける操作が必要となります。
V+プログラムで制御する場合は、ひとつのラダー図プログラムだけを使用しているため、この操作は不要です。

1 **マルチビューエクスプローラで、［構成・設定］－［タスク設定］をダブルクリックします。**

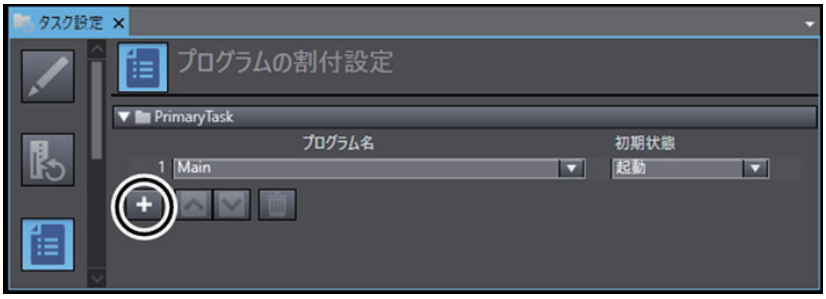


［タスク設定］タブが表示されます。

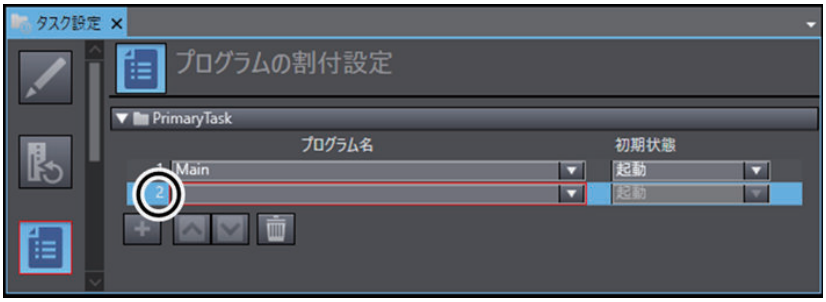
2 **エディットウィンドウに表示される編集画面で［プログラムの割付設定］ボタン（  ）をクリックします。**



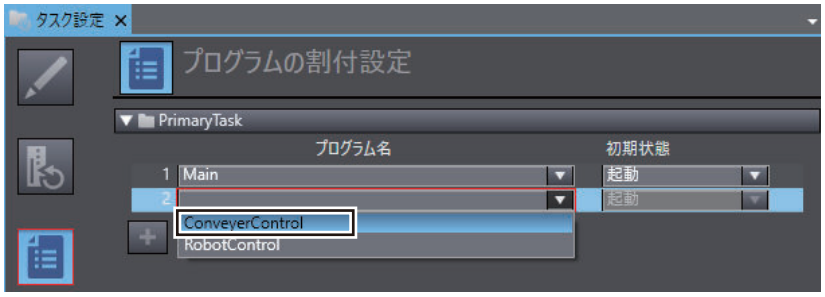
3 **［+］ボタンをクリックします。**



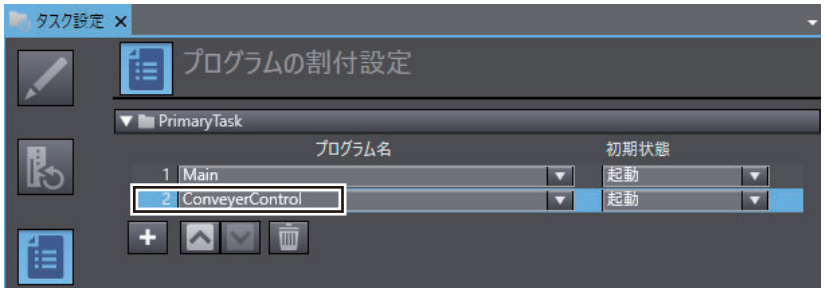
割り付けるプログラムの行が追加されます。



4 **［プログラム名］のプルダウンリストから使用するプログラム名を選択します。**



選択したプログラム名が表示されます。



5 **［RobotControl］についても同様に手順 3 以降を行います。**

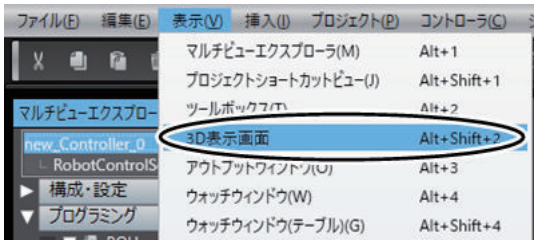


4-3-5 3D 形状データを配置する

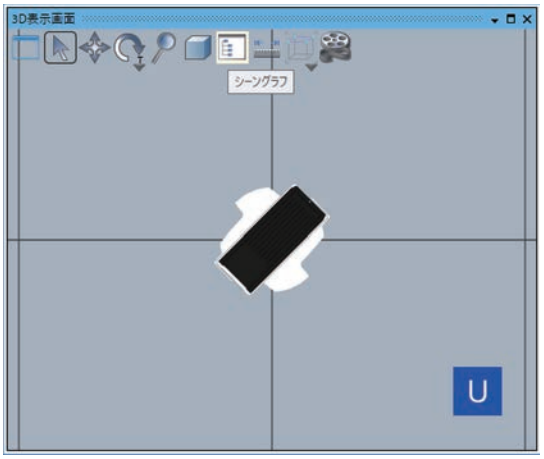
ロボットの動作範囲を表示する

ロボットの動作範囲を 3D 表示画面上で見えるようにします。動作範囲を見えるようにすることで、ロボットの位置の設定を行いやすくなります。

1 **Sysmac Studio のメインウィンドウのメニューバーから［表示］–［3D 表示画面］を選択します。**

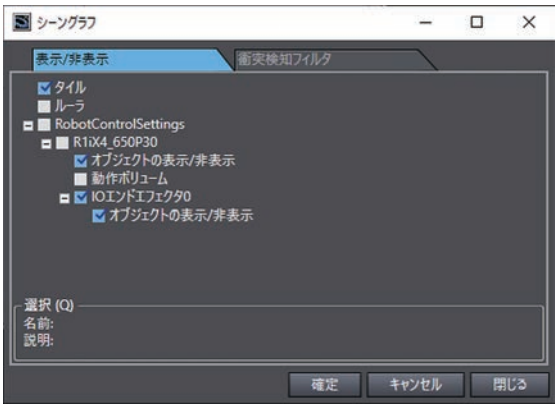


2 **3D 表示画面のシーングラフのアイコンをクリックします。**

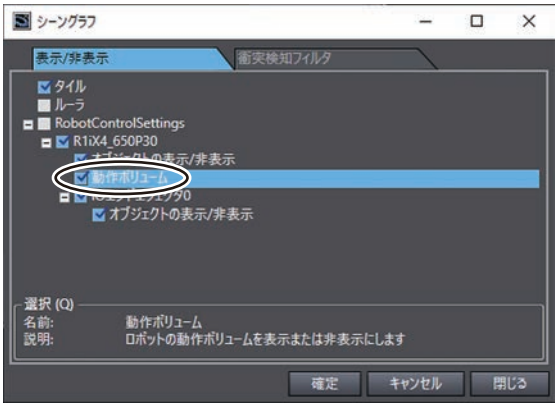


（注）3D 表示画面をフローティング状態にして説明しています。フローティング状態については、「A-2-1 3D 表示画面をフローティング状態にする (P.A-21)」を参照してください。

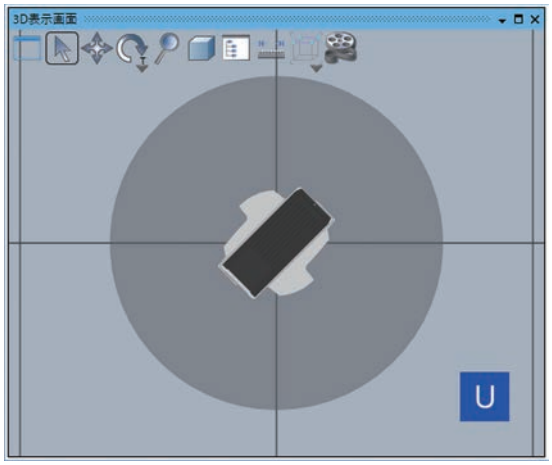
［シーングラフ］画面が表示されます。



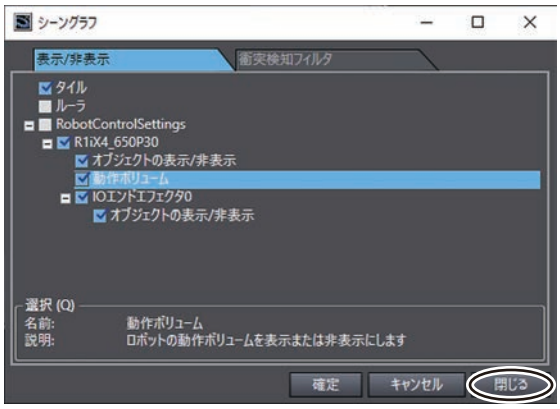
3 **［表示/非表示］タブ内の［RobotControlSettings］–［R1iX4_650P30］にある［動作ボリューム］のチェックボックスにチェックを入れます。**



3D 表示画面で、ロボットの動作範囲が灰色で表示されます。



4 **［シーングラフ］画面の［閉じる］ボタンをクリックします。**



ロボットを 3D 表示画面上に配置する

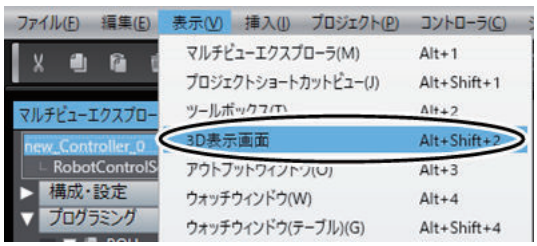
ロボットを 3D 表示画面上で適切な位置に配置します。

参考

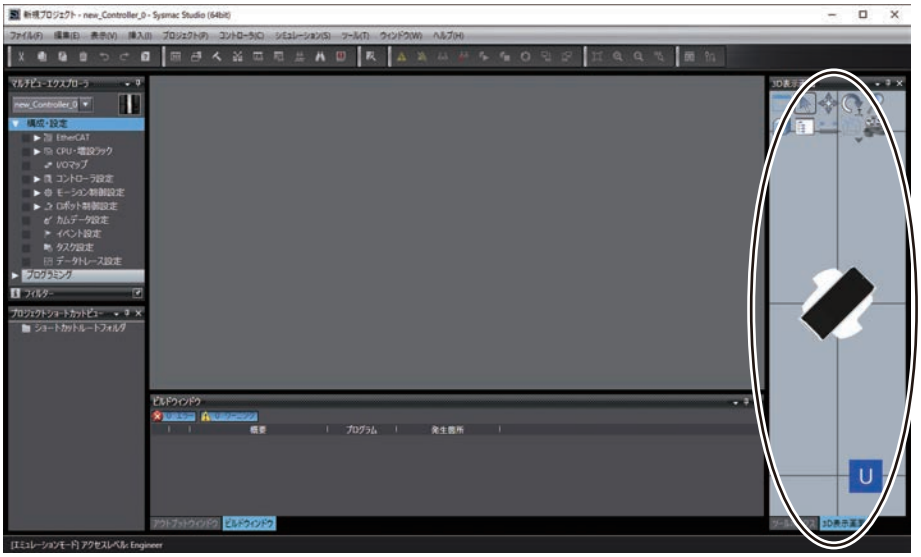
関連動画

- 3D 形状データの配置
  https://www.fa.omron.co.jp/video/detail/6264519599001/
  Sysmac Studio でロボットの 3D 形状データを配置する操作を説明しています。

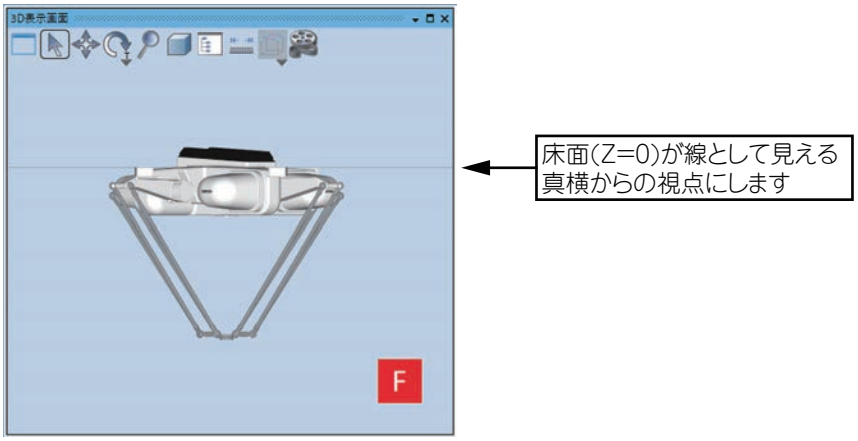
1 **Sysmac Studio のメインウィンドウのメニューバーから［表示］–［3D 表示画面］を選択します。**



メインウィンドウ内の右側に 3D 表示画面が追加されます。



2 **3D 表示画面をフローティング状態にし、視点を下記のように回転させます。**



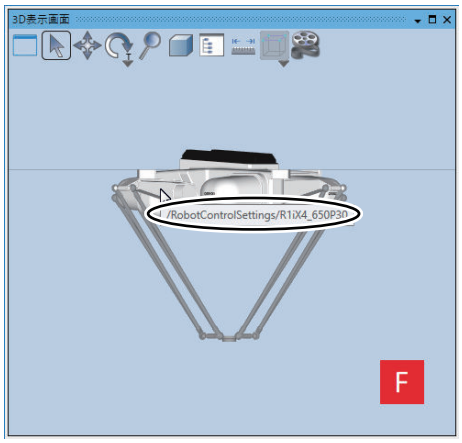
参考

3D 表示画面での視点を変更する方法については、「A-2 3D 表示画面の操作方法 (P.A-21)」を参照してください。

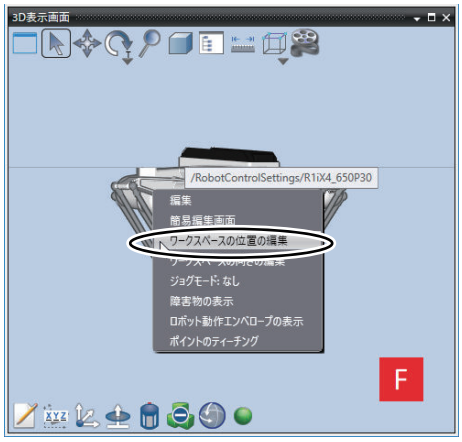
3 **3D 表示画面内で、ロボットにマウスカーソルを合わせて右クリックします。**
コンテキストメニューが表示されます。

参考

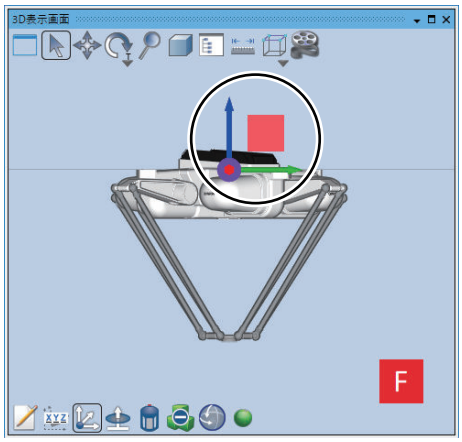
3D 表示画面のロボットにマウスカーソルを合わせると、3D 形状データの名称［/RobotControlSettings/R1iX4_650P30］が表示されます。



4 **コンテキストメニューから［ワークスペースの位置の編集］を選択します。**

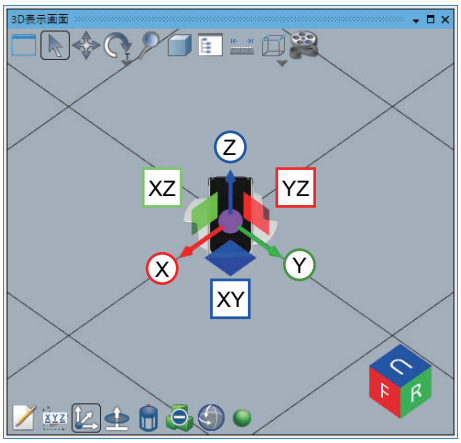


3D 形状データの原点に、色付き矢印と紫の丸、および面からなる移動アイコンが表示されます。



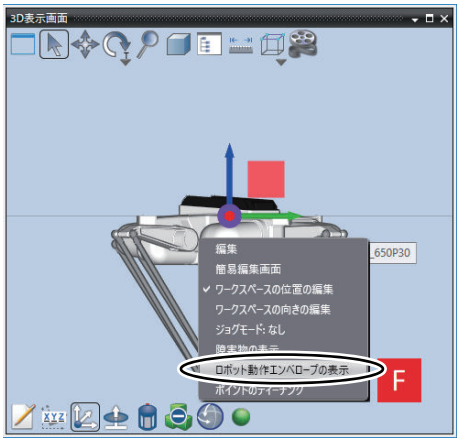
参考

矢印は、赤が X 軸、緑が Y 軸、青が Z 軸を示しています。また面は、赤が YZ 平面、緑が ZX 平面、青が XY 平面を示しています。

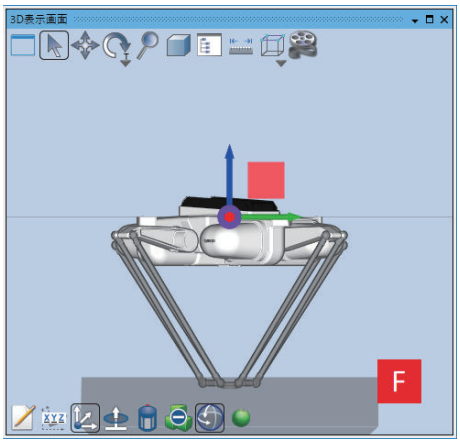


矢印をドラッグすると、対応する軸に沿って 3D 形状データを移動することができます。
面をドラッグすると、対応する平面に沿って 3D 形状データを移動することができます。また、紫の丸をドラッグすると、軸方向とは無関係に 3D 形状データを移動することができます。

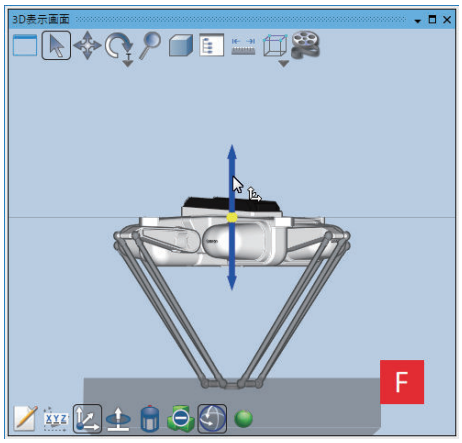
5 **3D 表示画面内で、ロボットにマウスカーソルを合わせて右クリックし、コンテキストメニューから［ロボット動作エンベロープの表示］を選択します。**



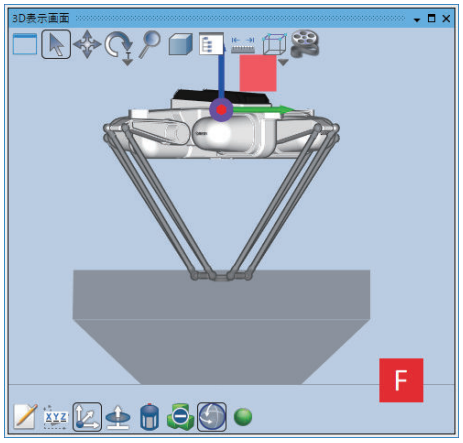
3D 表示画面で、ロボットの動作範囲が灰色で表示されます。



6 **青矢印をクリックし、上方向にドラッグすることにより、原点にいるロボットを Z 軸方向に引き上げます。**

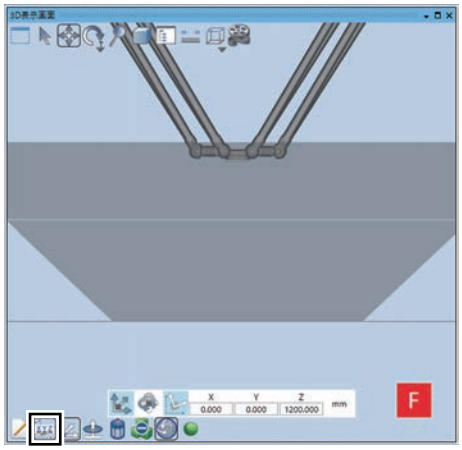


動作範囲が床面より上になるように移動させます。



参考

動作範囲をよりわかりやすく確認するために、3D 表示画面を拡大します。
また、［簡易編集画面］アイコンをクリックすることにより、X 軸、Y 軸、Z 軸の位置が数字で表記されます。

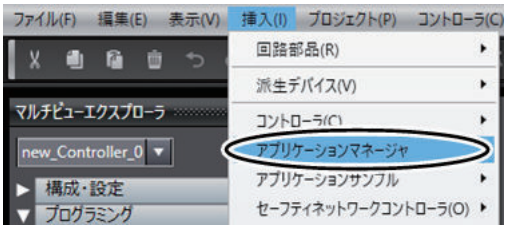


3D 表示画面を拡大や縮小については、「A-2-4 ズームインおよびズームアウト (P.A-26)」を参照してください。

3D CAD データをインポートおよび配置する

Sysmac Studio にコンベアの 3D CAD データを 3D 形状データとしてインポートします。
ここでは、装置モデルのコンベア A、コンベア B の 3D CAD データをインポートする手順、および配置について説明します。

1 **Sysmac Studio のメインウィンドウのメニューバーから［挿入］–［アプリケーションマネージャ］を選択します。**



［デバイスの追加］画面が開きます。



2 **［カテゴリ］、［デバイス］、および［バージョン］を以下のように選択し、［OK］ボタンをクリックします。**



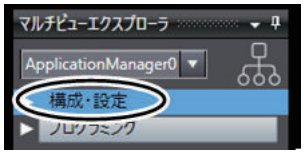
マルチビューエクスプローラのウィンドウ上部にあるドロップダウンリストに［ApplicationManager0］が追加されます。



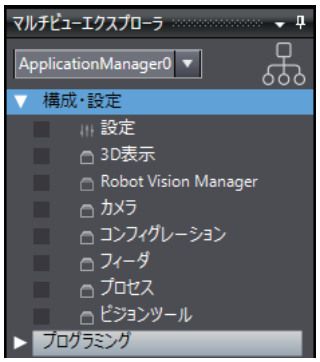
参考

本操作により追加したアプリケーションマネージャはシミュレーションでだけ使用します。実機では使用しません。

3 **マルチビューエクスプローラの［構成・設定］をクリックします。**



［構成・設定］の下にツリーが展開されます。

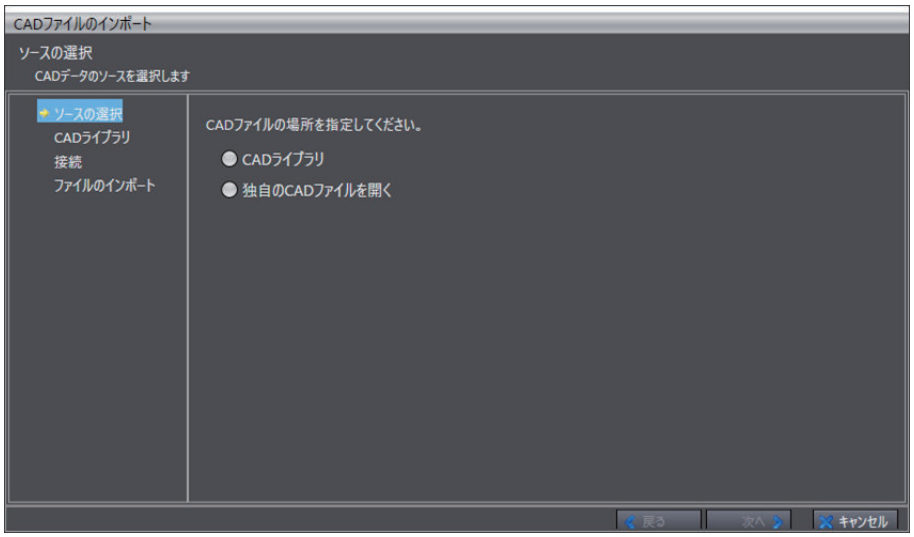


4 **［3D 表示］を右クリックします。**
コンテキストメニューが表示されます。

5 **コンテキストメニューから、［追加］－［CAD データ］を選択します。**

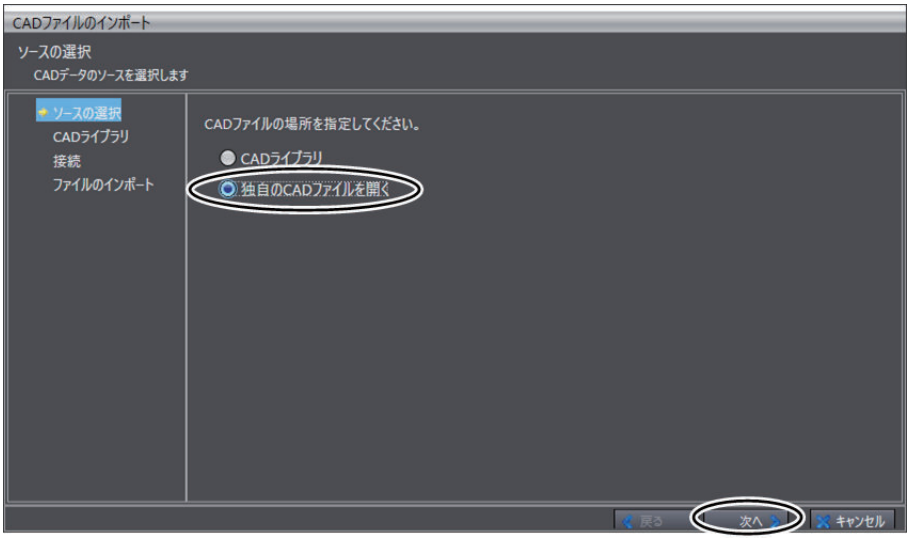


［CAD ファイルのインポート］画面が開きます。

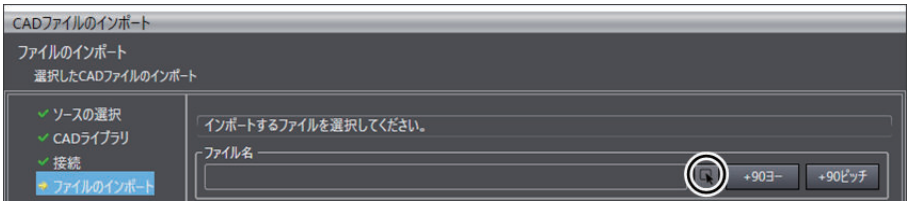


6 **［独自の CAD ファイルを開く］のラジオボタンをクリックします。**

7 **［次へ］ボタンをクリックします。**



8 **［ファイルを開く］ボタンをクリックします。**



［開く］ダイアログが開きます。



9 **対象の 3D CAD データを選択して［開く］ボタンをクリックします。**

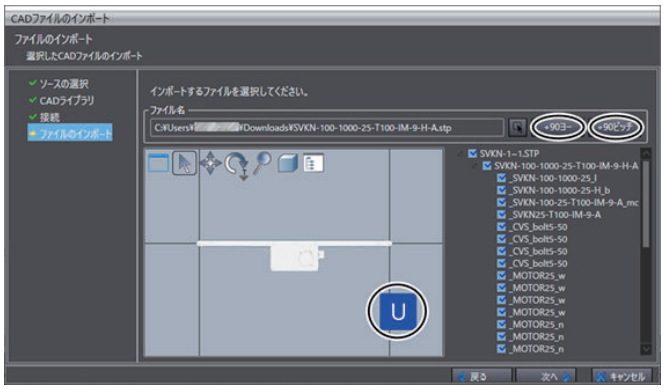
ここでは、コンベア A は MISUMI 社製 SVKN-100-1000-25-T100-IM-9-H-A とします。3D CAD データの拡張子は stp です。



［開く］ダイアログが閉じ、3D CAD データのインポートが始まります。



インポートが完了すると、［CAD ファイルのインポート］画面に、インポートした 3D CAD データのアセンブリ構成が表示されます。

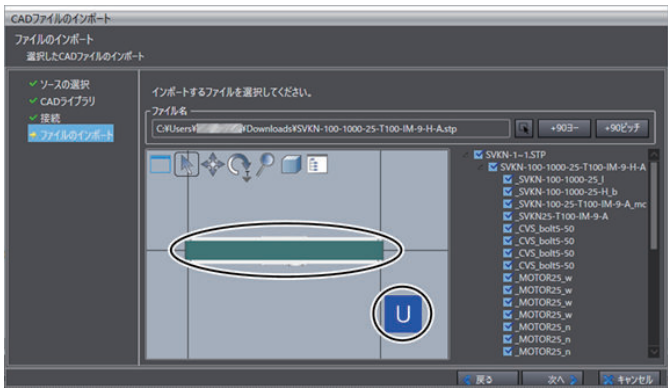


参考

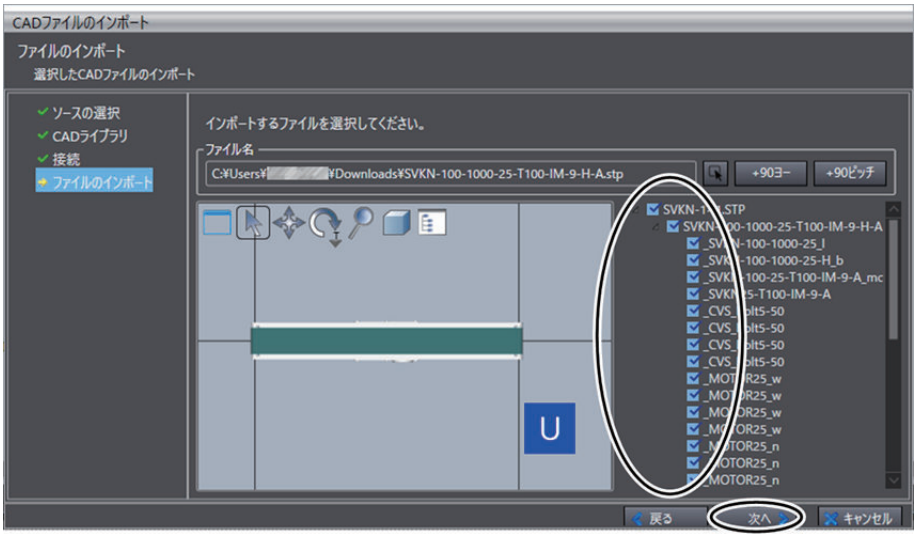
コンベア上面が U 面と一致していないため、一致させる必要があります。コンベアの向きを変えるには、［+90 ヨー］ボタン、および［+90 ピッチ］ボタンを使用します。
あらかじめ位置と向きを調整した CAD データをお使いいただくと、この操作は不要になります。

10 ［+90 ヨー］、［+90 ピッチ］、［+90 ヨー］、［+90 ピッチ］、［+90 ピッチ］の順にボタンをクリックします。

コンベア上面が U 面と一致します。



11 シミュレーションに追加するパーツにチェックを入れ、［次へ］ボタンをクリックします。

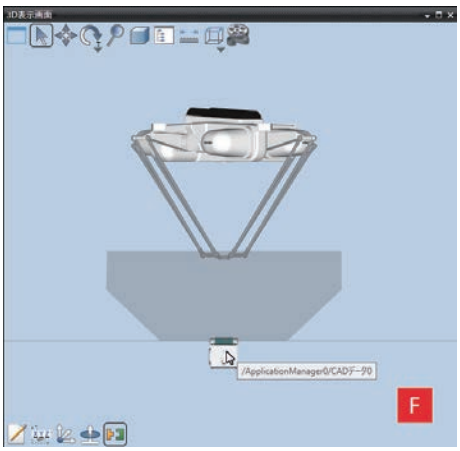


［CAD ファイルのインポート］画面が閉じ、メインウィンドウに戻ります。

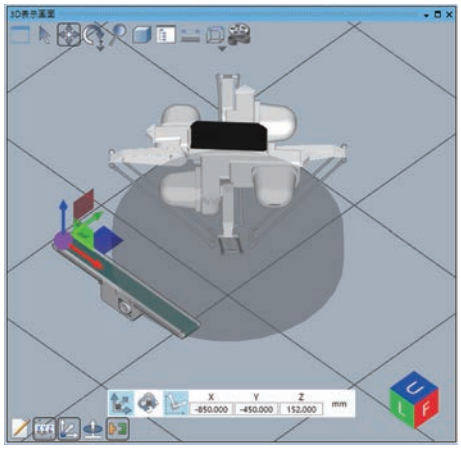
12 マルチビューエクスプローラで［構成・設定］–［3D 表示］に［CAD データ 0］が追加されていることを確認します。



13 コンベア A を「ロボットを 3D 表示画面上に配置する (P.4-32)」と同様に 3D 表示画面上に配置します。



コンベア A をロボットの動作範囲内でワークをピックできる位置に配置します。

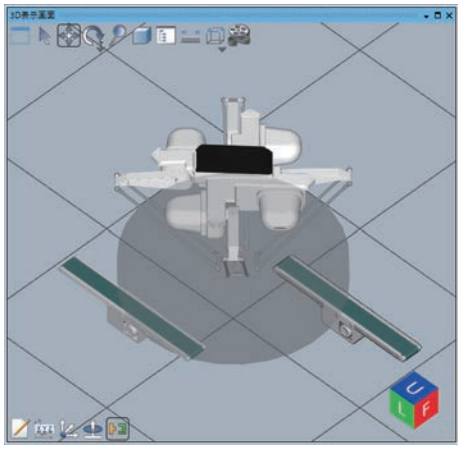


14 **コンベア B についても同様に手順 4 以降を行います。**

マルチビューエクスプローラの［構成・設定］-［3D 表示］に［CAD データ 1］が追加されます。



コンベア B をロボットの動作範囲内でワークをプレースできる位置に配置します。



これでコンベア A、コンベア B の 3D CAD データのインポート、および配置ができました。

衝突検知機能を有効にする

ロボットが、インポートした 3D CAD データなどの他の 3D 形状データと干渉する場合に、シミュレーションで検出できるようにします。ロボットが他の 3D 形状データと干渉する場合には、ロボットや他の 3D 形状データの配置、またはロボットの動きを調整する必要があります。
ここでは、コンベア A、コンベア B とロボットの干渉を検出するための設定手順を説明します。干渉チェックはグループ間で 3D 形状データが干渉するかをチェックするため、ロボットを含むグループ 0 と、コンベア A、コンベア B を含むグループ 1 の 2 つのグループを設定します。

使用上の注意

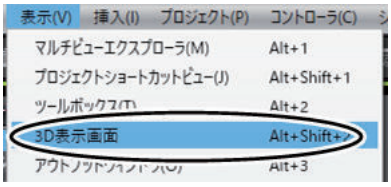
本ガイドではロボットを干渉検出の対象としています。ツール先端に 3D 形状データを配置した場合は、配置した 3D 形状データの干渉検出も必要です。ツール先端に配置した 3D 形状データを干渉検出の対象とする場合の操作方法については、『Sysmac Studio 3D シミュレーション機能 オペレーションマニュアル (SBCA-482)』を参照してください。

参考

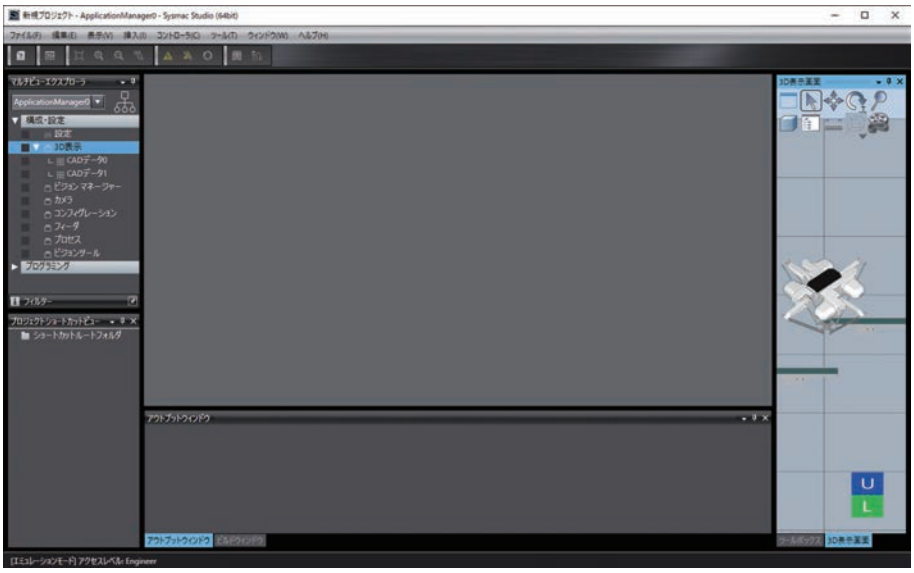
関連動画

- 衝突検知機能の設定
  https://www.fa.omron.co.jp/video/detail/6264519414001/
  Sysmac Studio で、ロボットが他の 3D 形状データと干渉する場合にシミュレーションで検出できるようにする手順を説明しています。

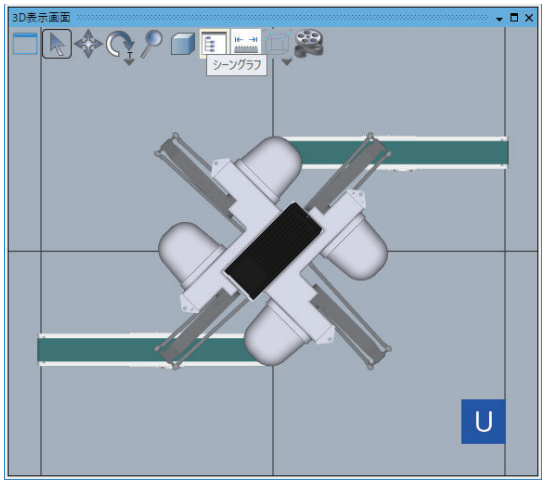
1 **Sysmac Studio のメインウィンドウのメニューバーから［表示］–［3D 表示画面］を選択します。**



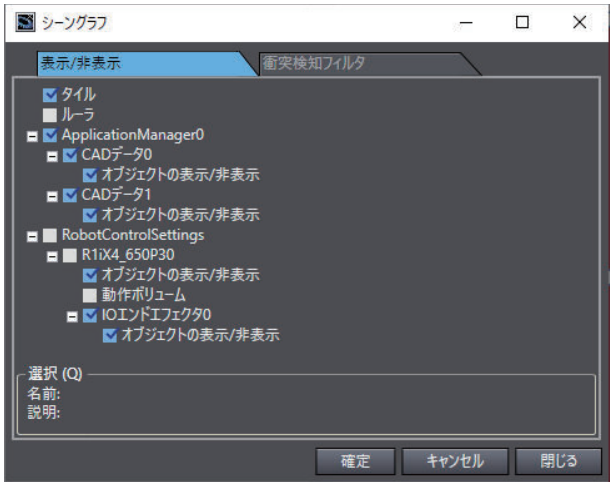
メインウィンドウ内の右側に 3D 表示画面が追加されます。



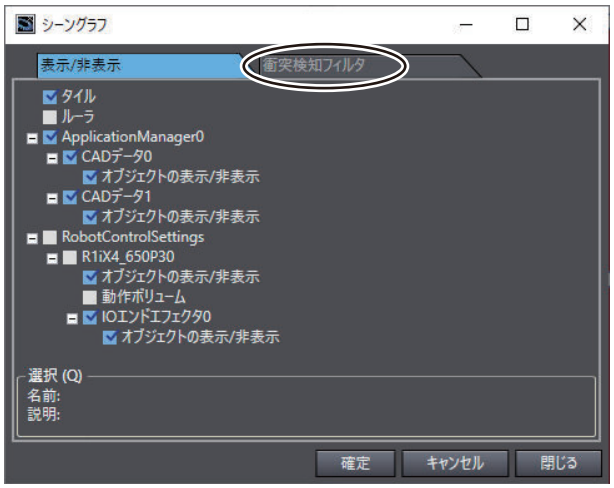
2 **3D 表示画面のアイコン［シーングラフ］をクリックします。**



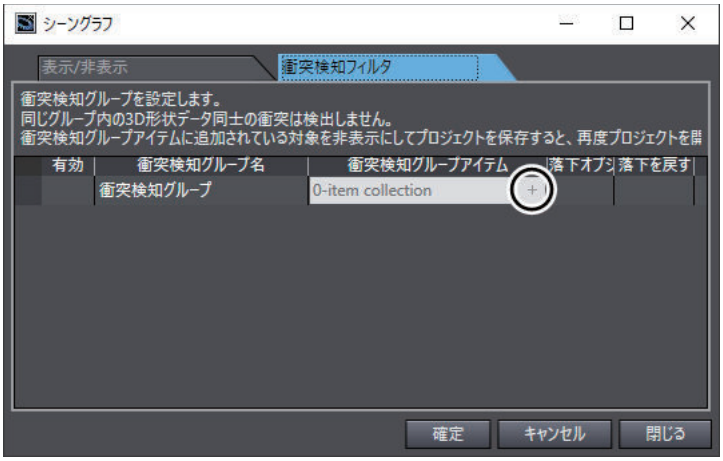
［シーングラフ］画面が開きます。



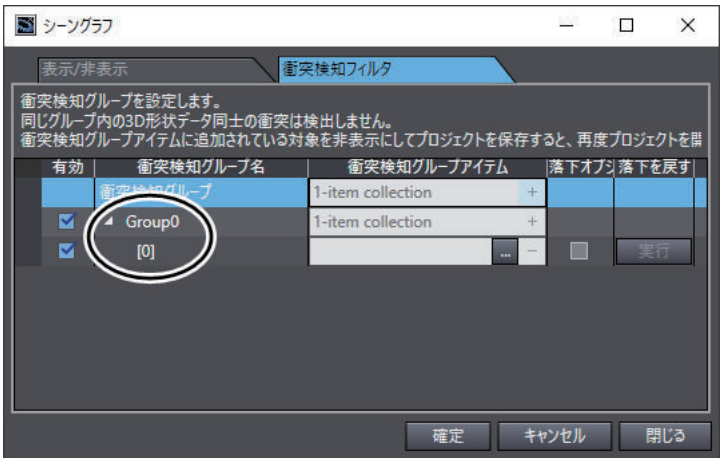
3 **［衝突検知フィルタ］タブをクリックします。**



4 **［衝突検知グループ］の［衝突検知グループアイテム］の［+］マークをクリックします。**

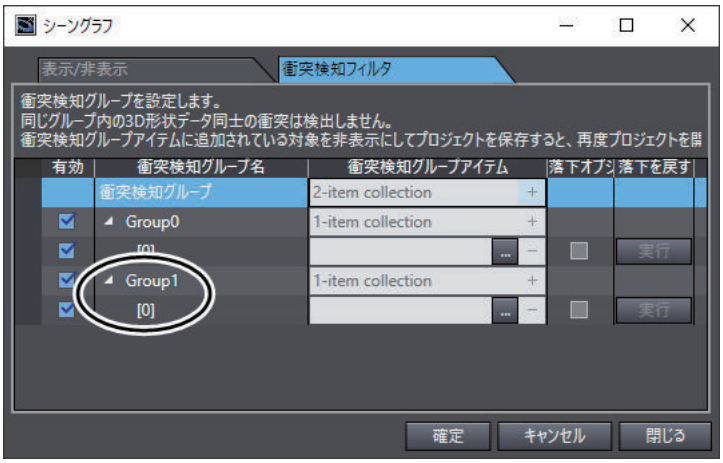


［シーングラフ］画面に、［衝突検知グループ名］が［Group 0］と設定された行が追加されます。

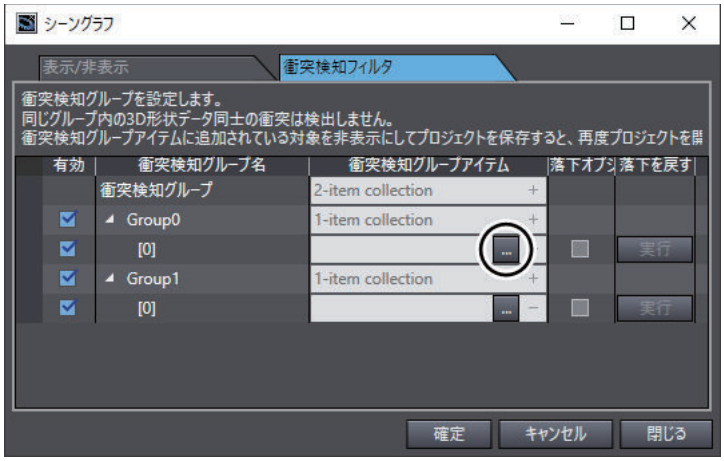


5 **手順 4 をもう一度実施します。**

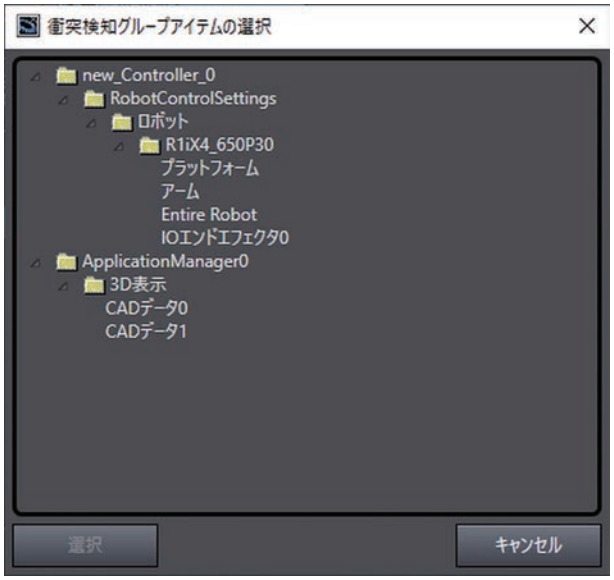
［シーングラフ］画面に、［衝突検知グループ名］が［Group 1］と設定された行が追加されます。



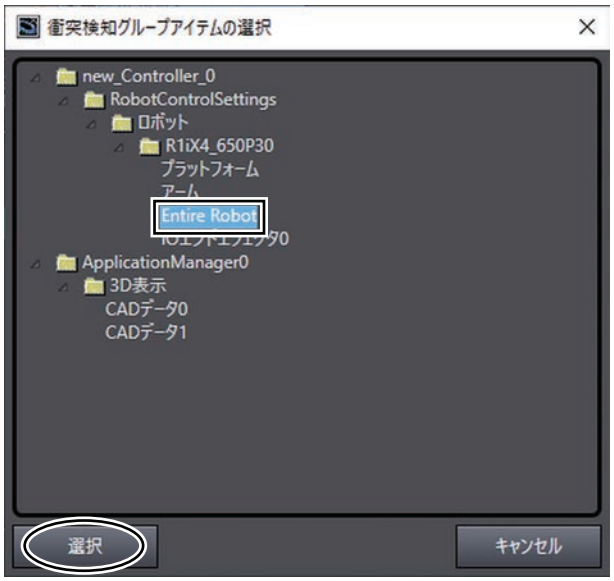
6 **［Group 0］の［[0]］の［衝突検知グループアイテム］の［...］ボタンをクリックします。**



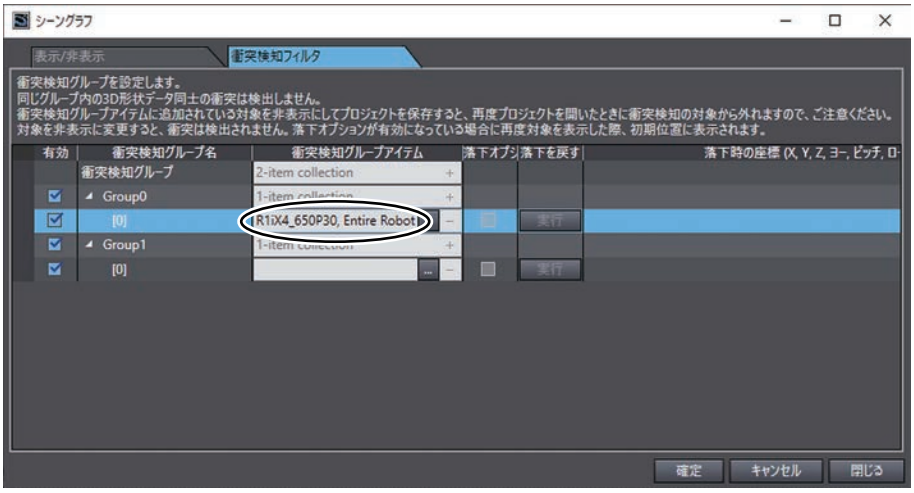
［衝突検知グループアイテムの選択］画面が開きます。



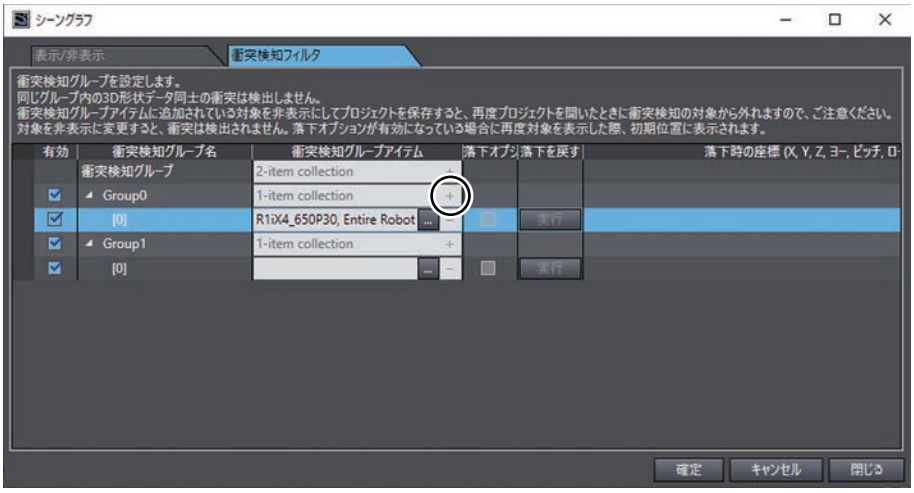
7 **ツリーから［new_Controller_0］–［RobotControlSettings］–［ロボット］–［R1iX4_650P30］–［Entire Robot］をクリックし、［選択］ボタンをクリックします。**



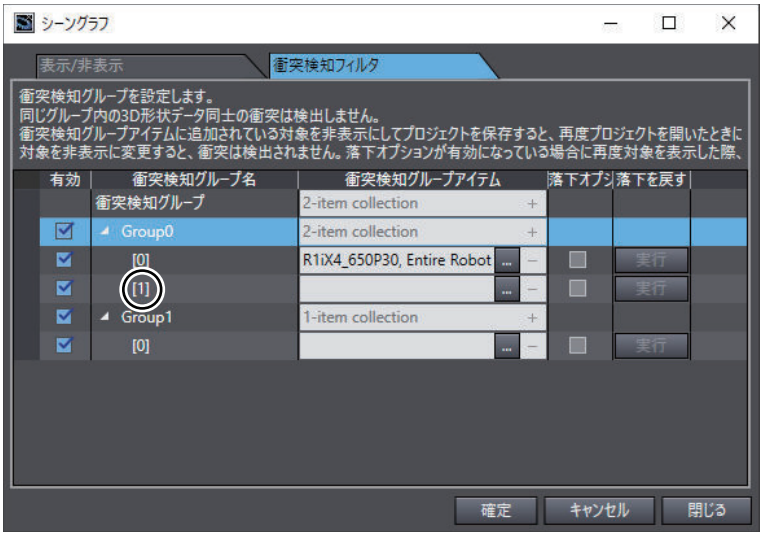
［衝突検知グループアイテムの選択］画面が閉じた後、［シーングラフ］画面に戻ります。［Group 0］－［[0]］の［衝突検知グループアイテム］に［R1iX4_650P30, Entire Robot］が表示されます。



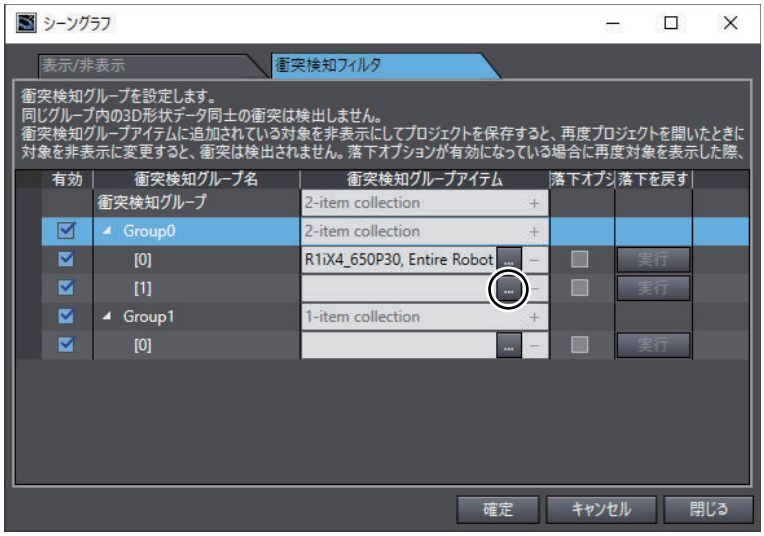
8 **［Group 0］の［衝突検知グループアイテム］の［+］マークをクリックします。**



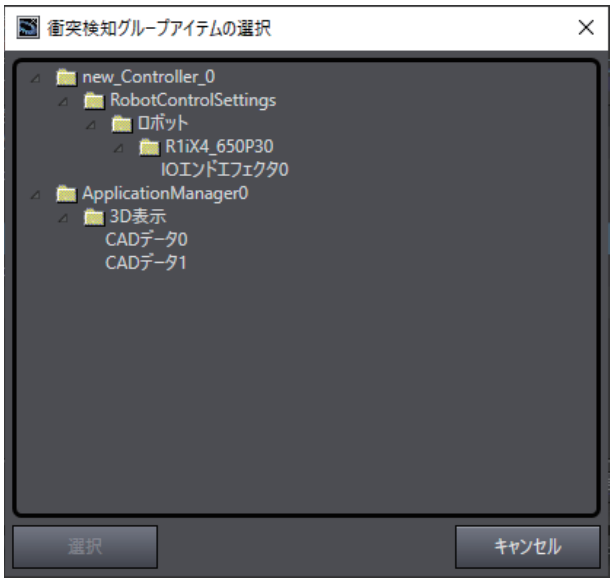
［Group 0］の下に［[1]］が追加されます。



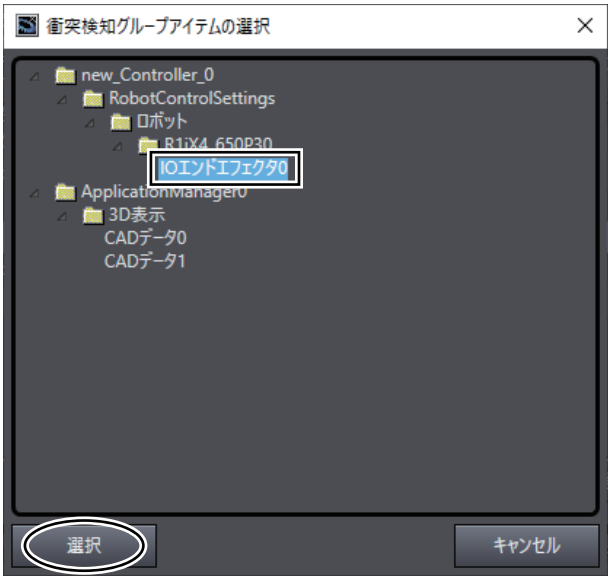
9 **［Group 0］の［[1]］の［衝突検知グループアイテム］の［...］ボタンをクリックします。**



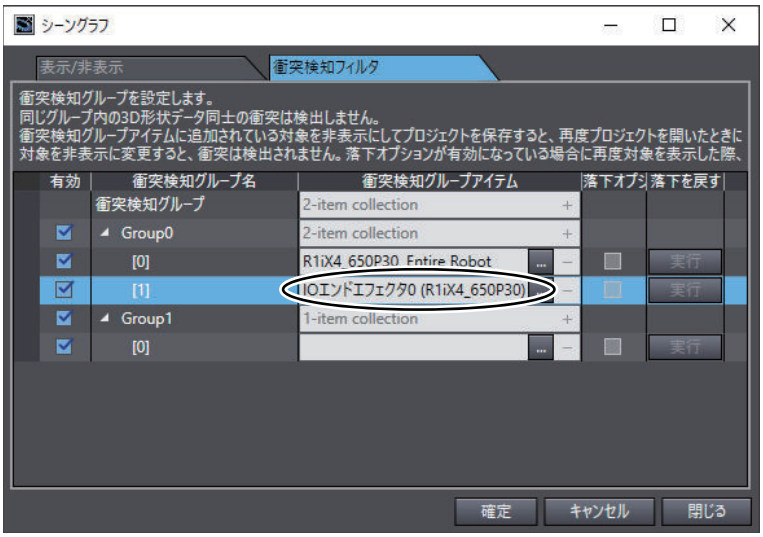
［衝突検知グループアイテムの選択］画面が開きます。



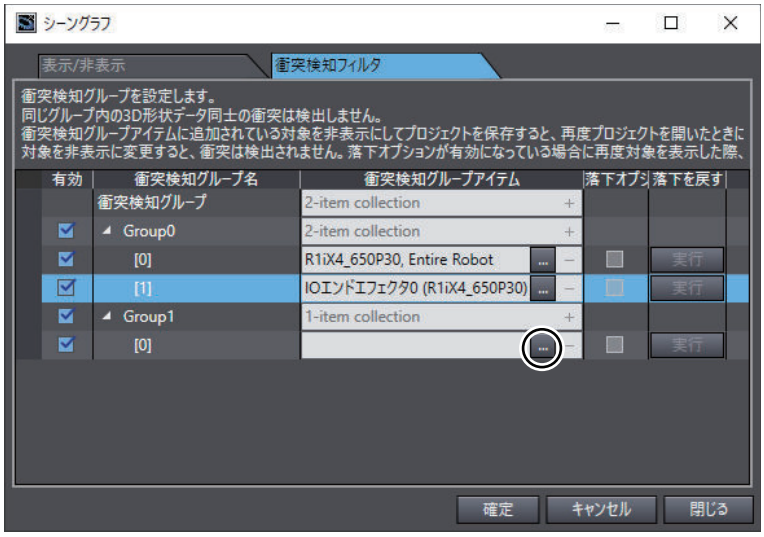
10 **ツリーから［new_Controller_0］–［RobotControlSettings］–［ロボット］–［R1iX4_650P30］–［IO エンドエフェクタ 0］をクリックし、［選択］ボタンをクリックします。**



［衝突検知グループアイテムの選択］画面が閉じた後、［シーングラフ］画面に戻ります。［Group 0］－［[1]］の［衝突検知グループアイテム］に［IO エンドエフェクタ 0 (R1iX4_650P30)］が表示されます。



11 **［衝突検知グループ名］が［Group 1］－［[0]］の［衝突検知グループアイテム］の［...］ボタンをクリックします。**

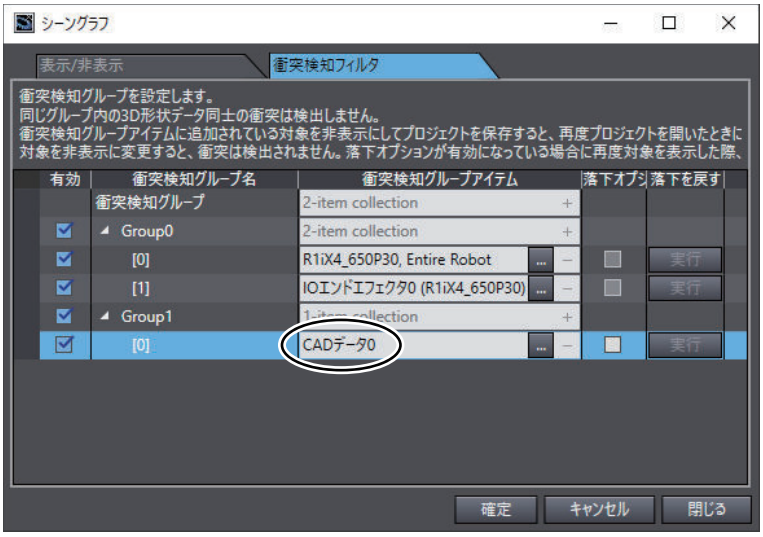


［衝突検知グループアイテムの選択］画面が開きます。

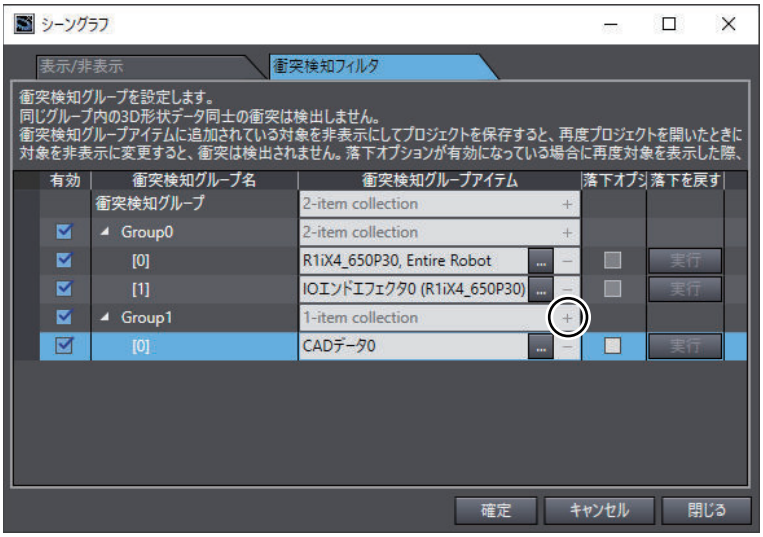
12 ツリーから［ApplicationManager0］－［3D 表示］－［CAD データ 0］をクリックし、［選択］ボタンをクリックします。



［衝突検知グループアイテムの選択］画面が閉じた後、［シーングラフ］画面に戻ります。［Group 1］－［[0]］の［衝突検知グループアイテム］に［CAD データ 0］が表示されます。



13 ［Group 1］の［衝突検知グループアイテム］の［+］マークをクリックします。

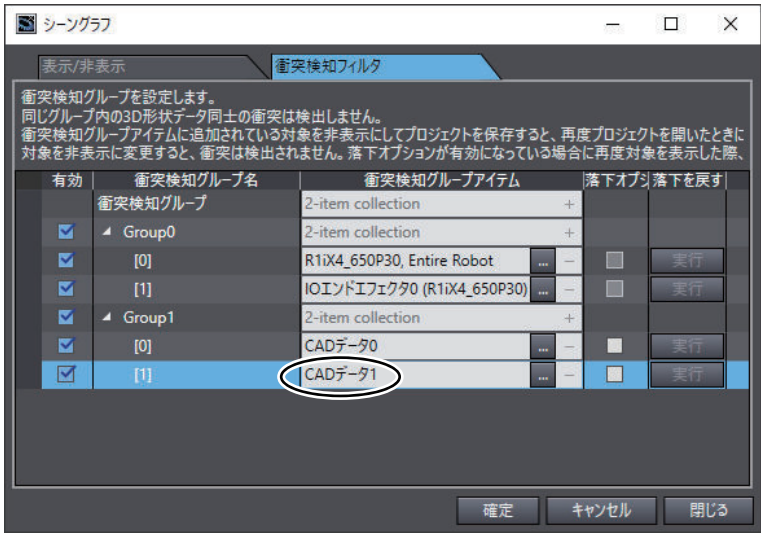


［Group 1］の下に［[1]］が追加されます。

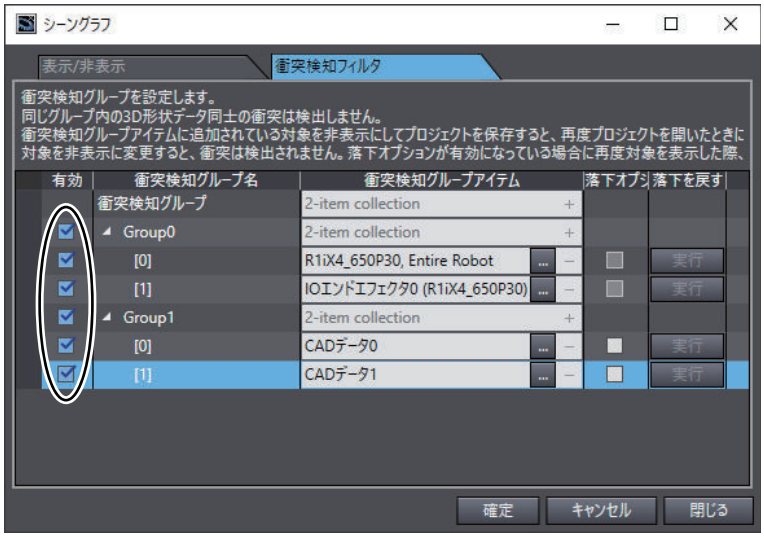


14 **[Group 1] – [[1]] についても同様に、手順 8 から手順 10 を行い［CAD データ 1］を追加します。**

［Group 1］–［[1]］–［衝突検知グループアイテム］に［CAD データ 1］が表示されます。



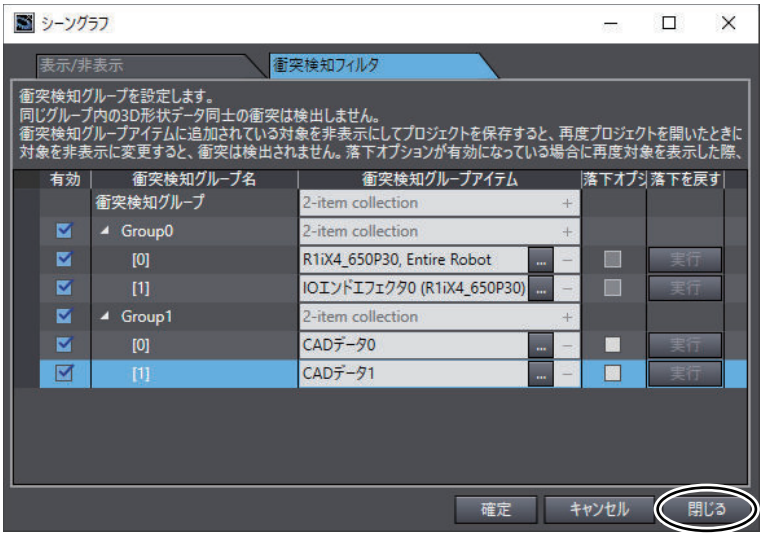
15 **[有効] のチェックボックスのすべてにチェックが入っていることを確認し、チェックが入っていない場合は、チェックボックスをクリックしてチェックを入れます。**



16 **[確定] ボタンをクリックします。**

衝突検知フィルタの設定が保存されます。

17 **[シーングラフ] 画面の［閉じる］ボタンをクリックします。**

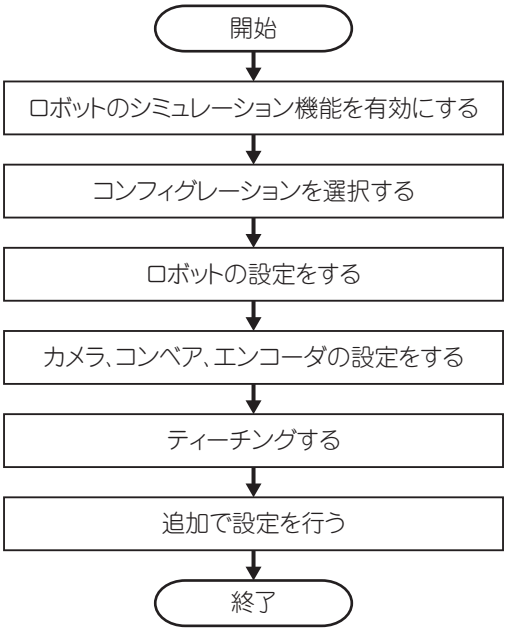


［シーングラフ］画面が閉じます。

これでロボットとコンベア A、コンベア B との干渉をチェックするための設定ができました。

4-3-6 Pack Manager サンプルを作成する

Pack Manager サンプル を作成する手順について説明します。
操作フローは、以下のようになります。



参考

以降で説明するフローの詳細な手順は動画でもご覧になれます。

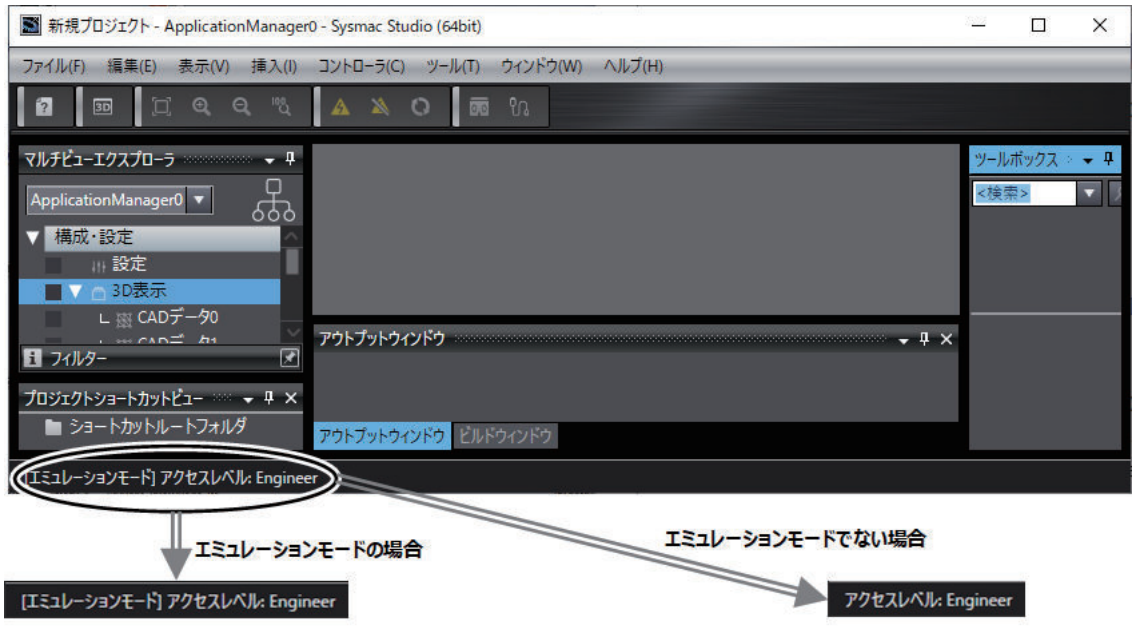
関連動画

- ロボット動作設定 Pack Manager サンプル作成
  https://www.fa.omron.co.jp/video/detail/6264519021001/
  Sysmac Studio の Pack Manager の機能を使用してシミュレーションで動的ピック＆プレース装置を設定する手順を説明しています。
- ロボット動作設定 Pack Manager でのティーチング
  https://www.fa.omron.co.jp/video/detail/6264519169001/
  Pack Manager の機能を使用してシミュレーションで動的ピック＆プレース装置のピック位置、プレース位置をティーチングする手順を説明しています。

ロボットのシミュレーション機能を有効にする

Sysmac Studio のシミュレーション機能を有効にします。
ロボットのシミュレーションを行うためには、プロジェクトファイルをエミュレーションモードで開く必要があります。

Sysmac Studio でプロジェクトファイルを開いた状態でエミュレーションモードかどうかは、メインウィンドウのステータスバーの表示で確認してください。[[エミュレーションモード] アクセスレベル: Engineer] と表示されている場合は、プロジェクトはエミュレーションモードで開かれている状態です。



エミュレーションモードで開かれていない場合、ツールバーの［エミュレーションモードの有効化］アイコンを押し、エミュレーションモードとしてください。



1 マルチビューエクスプローラのデバイスは、[new_Controller_0] を選択します。



2 **メニューバーの［シミュレーション］-［実行］を選択します。**
シミュレータ起動中の画面が表示されます。



シミュレータ起動中の画面が閉じた後、メインウィンドウのツールバーの下部が黄緑色に変わります。



コンフィグレーションを選択する

ロボットと周辺機器のコンフィグレーションを選択します。
ピック & プレース時にパレット機能、ベルトエンコーダおよびカメラを使用するかどうかを決めます。
以下に機能の詳細と選択方法を示します。
本ガイドでは、ピック時にベルトエンコーダおよびカメラを使用し、パレット機能は使用しません。

記号	動作	機能名	内容
A	ピック	パレット	ピックアップ位置をパレットとみなすか選択できます。 選択すると、パレットの各行／各列からワークをピックすることができます。
B	ピック	エンコーダ	ベルトエンコーダを使ってピックアップ位置を補正するか選択できます。 ベルトコンベア上を流れるワークをピックする場合に選択します。
C	ピック	カメラ	カメラを使ってピックアップ位置を補正するか選択できます。 コンベア上のワーク姿勢が固定されない場合に選択します。
D	搬送中	カメラ	搬送中に上向き姿勢のカメラを使って、プレース位置を補正するか選択できます。 搬出側コンベア上でワークを同一姿勢に整列してプレースする場合に選択します。
E	プレース	パレット	プレース位置をパレットとみなすか選択できます。 みなす場合は、パレットの各行／各列にワークをプレースすることができます。
F	プレース	エンコーダ	ベルトエンコーダを使ってプレース位置を補正するか選択できます。 搬出側コンベア上でワークを等間隔に整列してプレースする場合に選択します。
G	プレース	カメラ	カメラを使ってプレース位置を補正するか選択できます。 ワークを特定の型に合わせてプレースする場合に選択します。

ピック時のコンフィグレーションの一覧を、次に示します。
上表の記号 A, B, C がピック時の動作となります。

選択肢			選択するコンフィグレーション名
A	B	C	
あり	あり	あり	選択肢がありません。 作成後、『Sysmac Studio ロボット統合システム構築機能 オペレーションマニュアル ロボット統合 CPU ユニット編 (SBCA-464)』に従いプロセスマネージャ設定を変更してください。
あり	あり	なし	ベルトラッチセンサでパレットの位置を検出
あり	なし	あり	選択肢がありません。 作成後、『Sysmac Studio ロボット統合システム構築機能 オペレーションマニュアル ロボット統合 CPU ユニット編 (SBCA-464)』に従いプロセスマネージャ設定を変更してください。
あり	なし	なし	固定パレット
なし	あり	あり	カメラでベルト上の位置を検出
なし	あり	なし	固定マウントカメラを使用
なし	なし	あり	ベルトラッチセンサを使用
なし	なし	なし	固定位置

ピック後修正時のコンフィグレーションの一覧を、次に示します。
上表の記号 D がピック後の動作となります。

選択肢	選択するコンフィグレーション名
D	
あり	いいえ、グリッパ内でワークの位置を修正しません。
なし	はい、カメラを使用してグリッパ内でワークの位置を修正します。

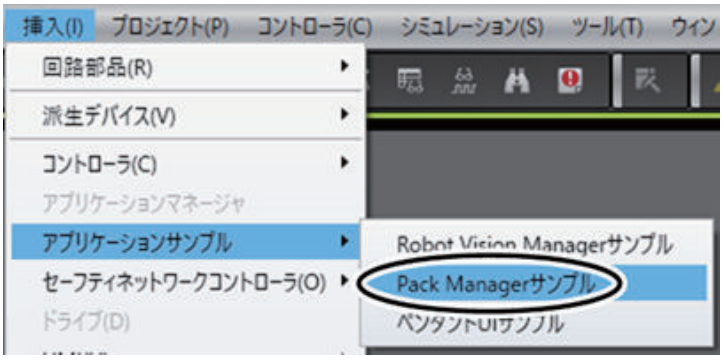
プレース時のコンフィグレーションの一覧を、次に示します。
上表の記号 E, F, G がプレース時の動作となります。

選択肢			選択するコンフィグレーション名
E	F	G	
あり	あり	あり	選択肢がありません。 作成後、『Sysmac Studio ロボット統合システム構築機能 オペレーションマニュアル ロボット統合 CPU ユニット編 (SBCA-464)』に従いプロセスマネージャ設定を変更してください。
あり	あり	なし	ベルトラッチセンサでパレットの位置を検出
あり	なし	あり	選択肢がありません。 作成後、『Sysmac Studio ロボット統合システム構築機能 オペレーションマニュアル ロボット統合 CPU ユニット編 (SBCA-464)』に従いプロセスマネージャ設定を変更してください。
あり	なし	なし	固定パレット
なし	あり	あり	カメラでベルト上の位置を検出
なし	あり	なし	固定マウントカメラを使用
なし	なし	あり	ベルトラッチセンサを使用
なし	なし	なし	固定位置

1 **Sysmac Studio のマルチビューエクスプローラのデバイスリストから［new_Controller_0］を選択します。**



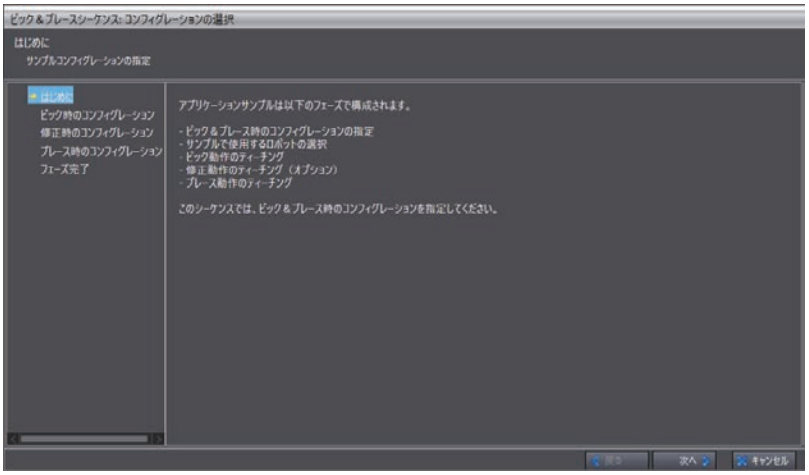
2 **Sysmac Studio のメインウィンドウのメニューバーから［挿入］–［アプリケーションサンプル］–［Pack Manager サンプル］を選択します。**



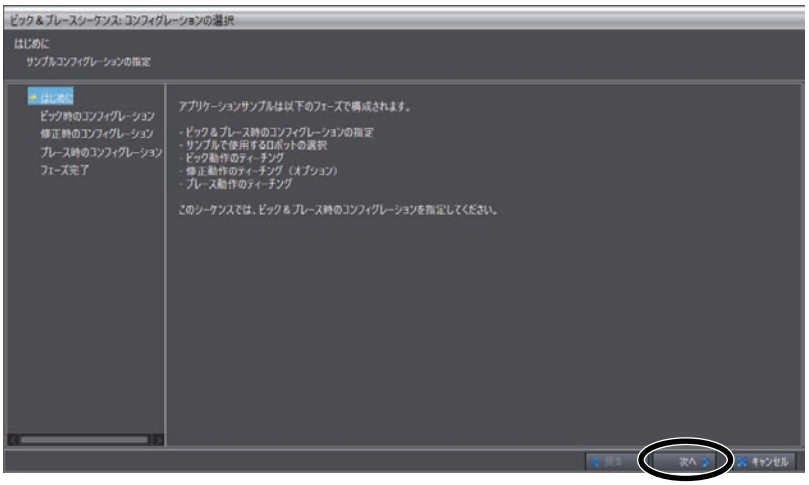
使用上の注意

[Pack Manager サンプル] での設定を最後のティーチングまで完了せずに終了またはキャンセルした場合は、[Pack Manager サンプル] の最初の手順からやり直す必要があります。なお、[Pack Manager サンプル] の実行でマルチビューエクスプローラのデバイスリストに [ApplicationManager1] が作成されてしまったときは、マルチビューエクスプローラのデバイスリストから[Delete]キーで一旦 [ApplicationManager1] を削除してからやり直してください。

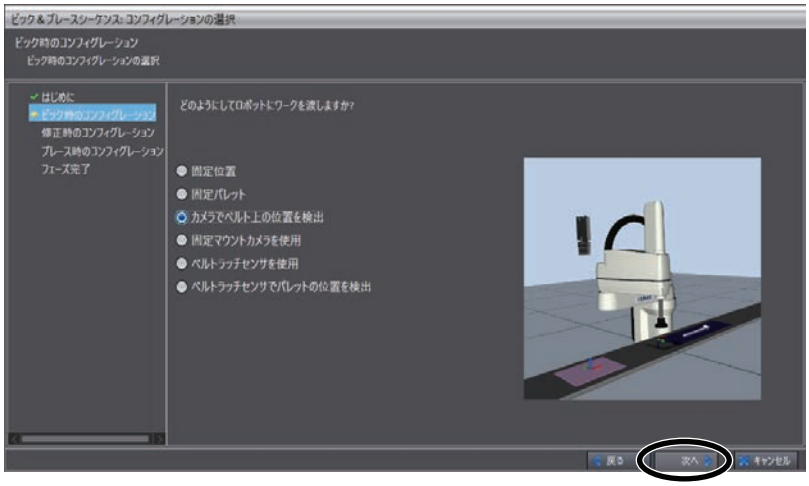
[ピック&プレースシーケンス: コンフィグレーションの選択] 画面が表示されます。



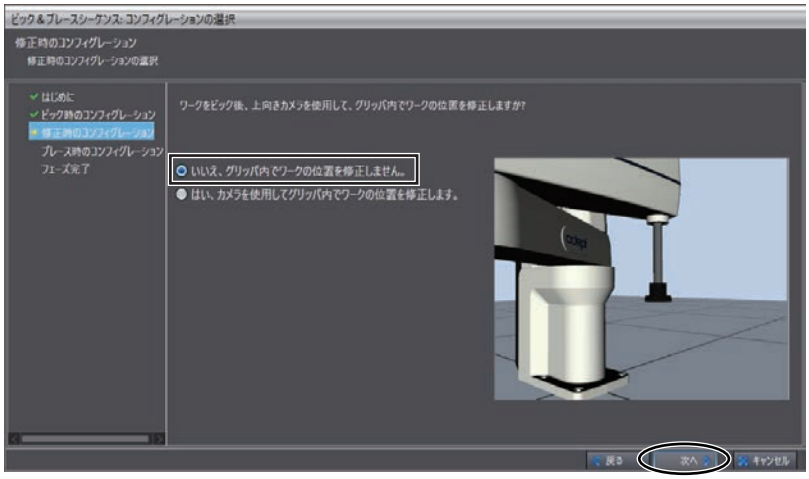
3 **画面右下の［次へ］ボタンをクリックします。**



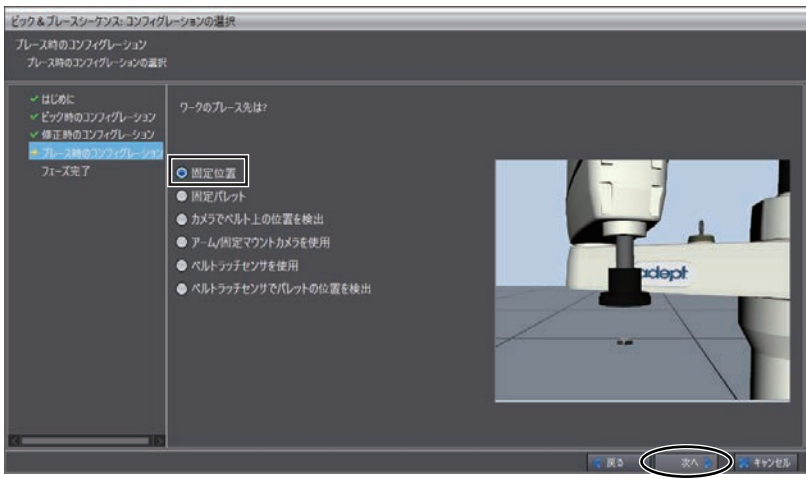
4 **[カメラでベルト上の位置を検出]のラジオボタンを選択し、[次へ]ボタンをクリックします。**



5 **[いいえ、グリッパ内でワークの位置を修正しません]のラジオボタンを選択し、[次へ]ボタンをクリックします。**



6 **[固定位置]のラジオボタンを選択し、[次へ]ボタンをクリックします。**



7 **[完了]ボタンをクリックします。**

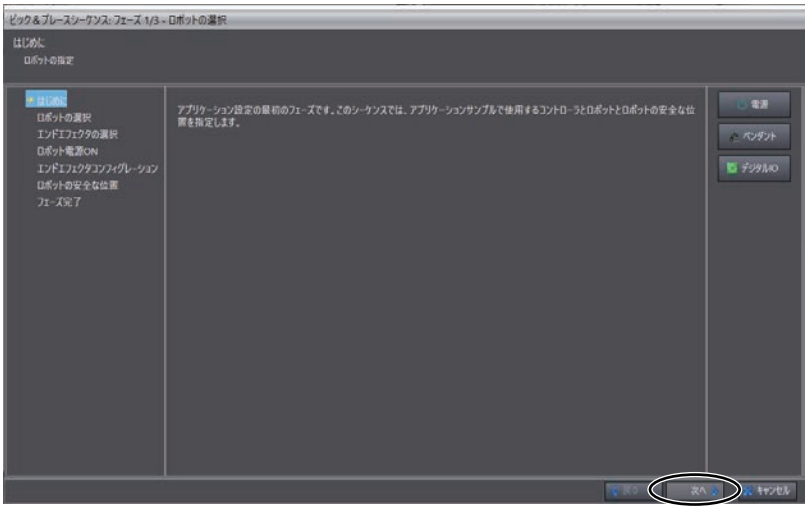


画面が閉じ、［ピック＆プレースシーケンス: フェーズ 1/3 – ロボットの選択］画面が表示されます。

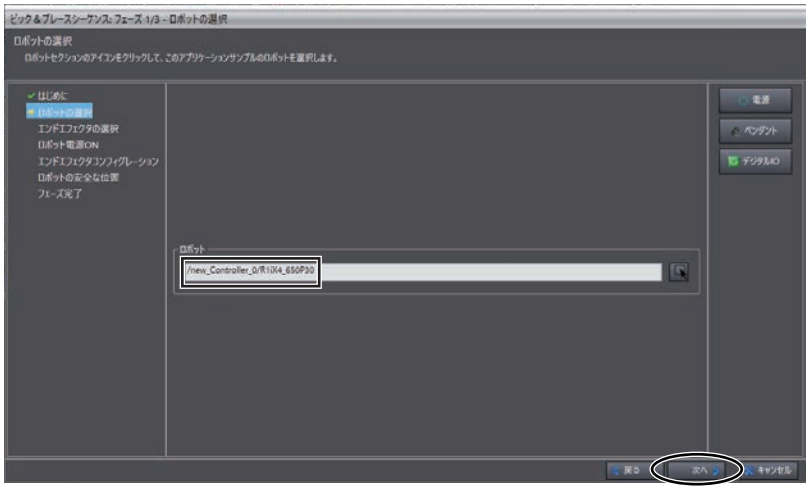
ロボットの設定をする

Pack Manager にて使用するロボットおよびエンドエフェクタを設定し、待機位置をティーチングします。

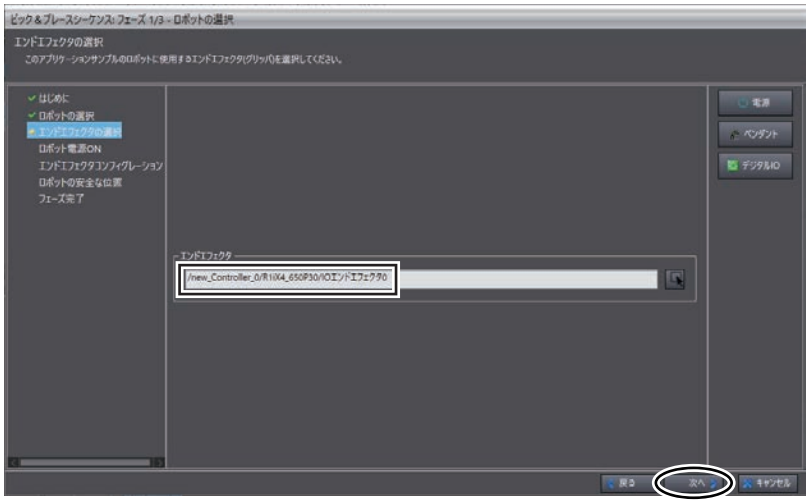
1 **［はじめに］の画面で、［次へ］ボタンをクリックします。**



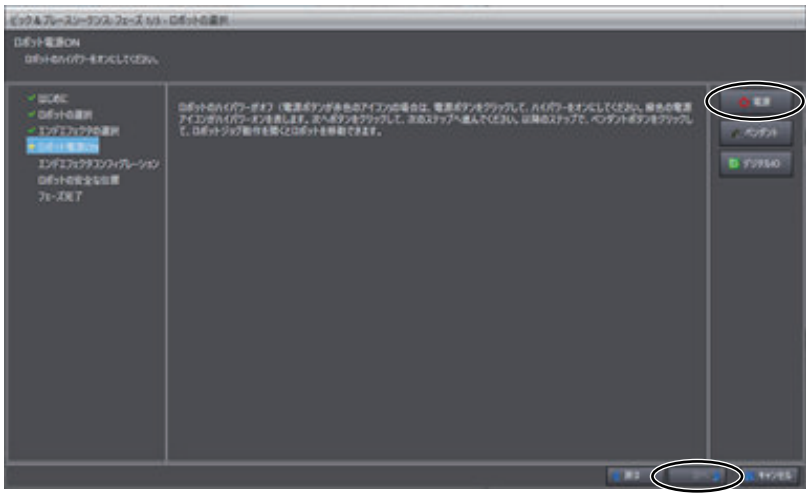
2 **［ロボットの選択］の画面で、［R1iX4_650P30］が選択されていることを確認し、［次へ］ボタンをクリックします。**



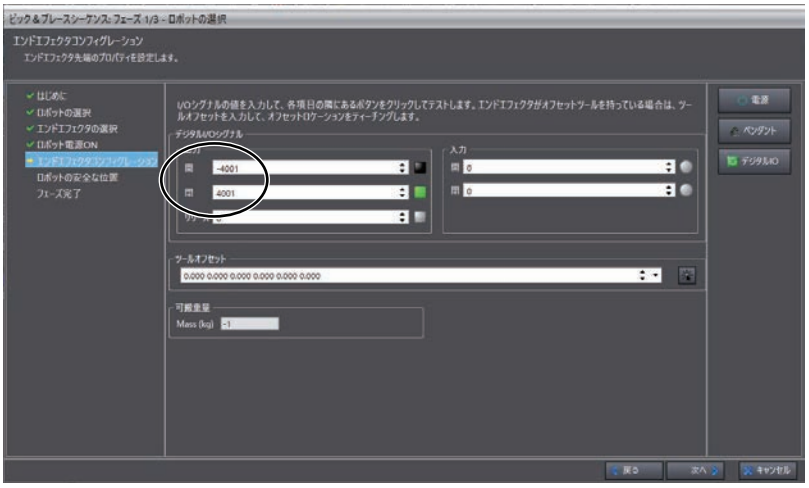
3 **[エンドエフェクタの選択]の画面で、[IO エンドエフェクタ 0]が選択されていることを確認し、[次へ]ボタンをクリックします。**



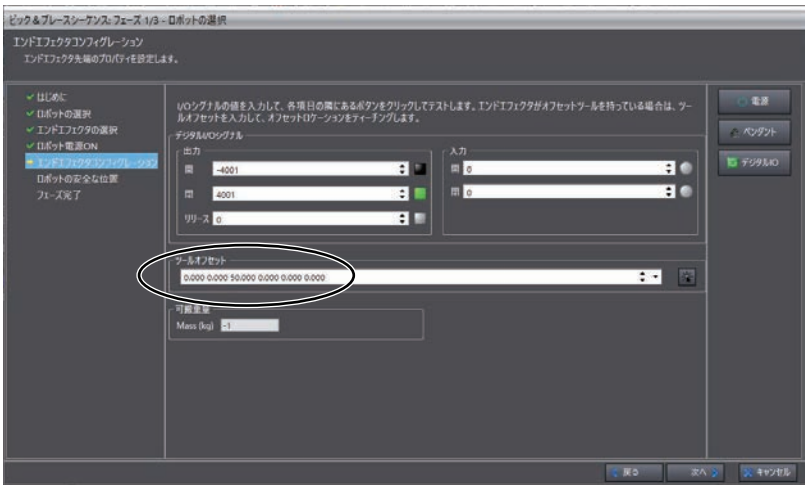
4 **[ロボット電源 ON]の画面が表示される場合があります。その場合、[電源]ボタンをクリックしてから、画面右下の[次へ]ボタンをクリックします。**



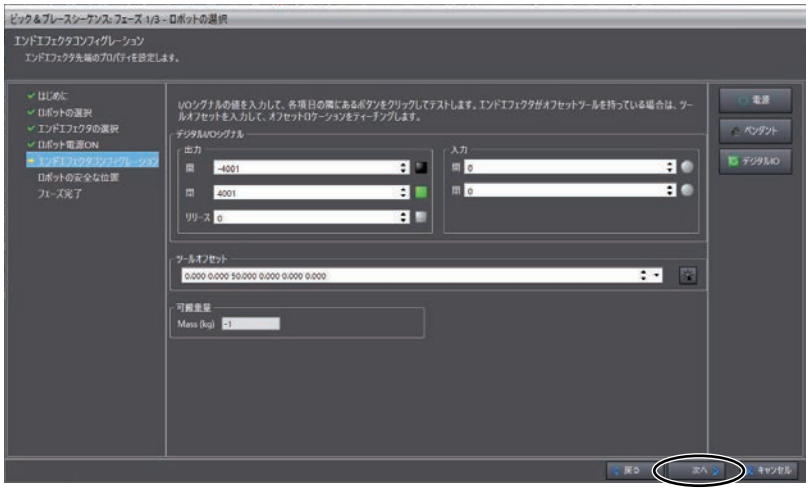
5 **吸着パッドを制御するためのロボット内蔵 I/O の信号番号の設定をします。画面中央にある［デジタル I/O シグナル］ボックス内の［出力］ボックスの［開］に「-4001」、［閉］に「4001」を入力します。これは、「V+デジタル I/O (P.4-7)」で割り当てたソレノイドバルブの開信号です。**



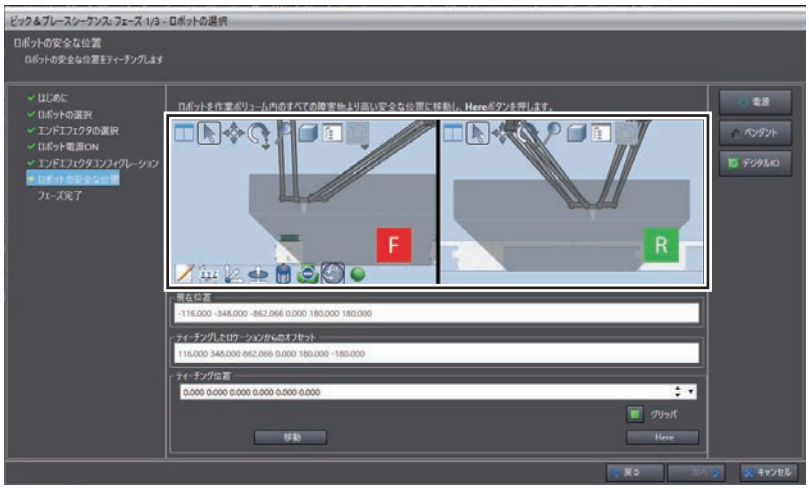
6 **TCP から吸着パッド先端までのツールオフセットとして Z 方向に 50mm を設定します。画面中央にある［ツールオフセット］ボックスに、「0.000 0.000 50.000 0.000 0.000 0.000」を入力します。**



入力が完了したら、画面右下の［次へ］ボタンをクリックします。



7 **待機位置を仮決めするため、画面中央に表示された［3D 表示画面］でロボットを操作します。待機位置は、ピックするコンベア上のおよそのピック位置の上方を指定します。**

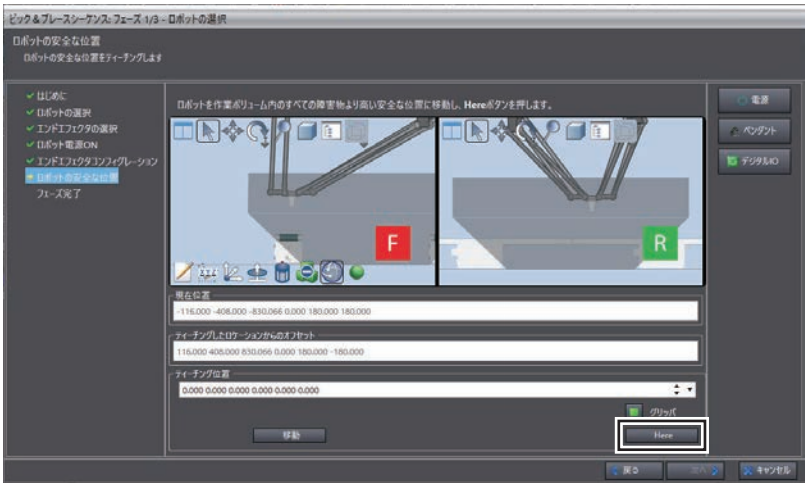


使用上の注意

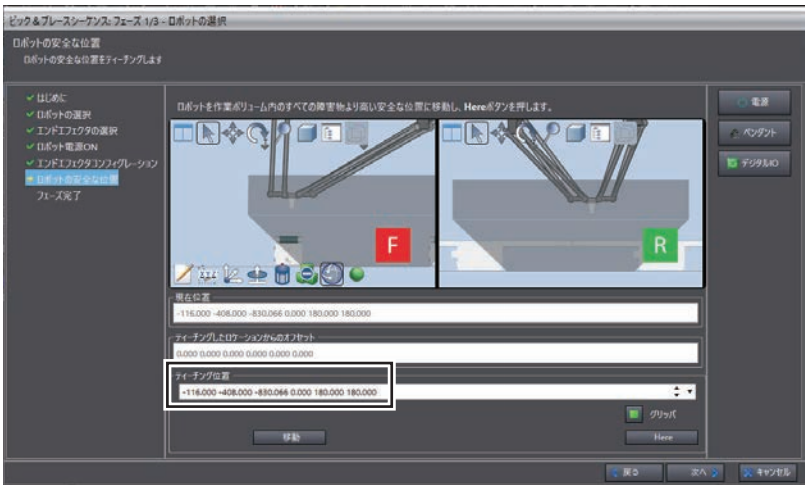
待機位置が「カメラ、コンベア、エンコーダの設定をする (P.4-64)」で設定するベルトウィンドウから外れていると、トラッキング時にベルトウィンドウ違反のエラーが発生しトラッキングできません。設定するベルトウィンドウのおよその位置をあらかじめ想定して、ベルトウィンドウ内に収まる位置の上方に待機位置を指定してください。

待機位置の仮決めが完了すれば、次の操作に進みます。

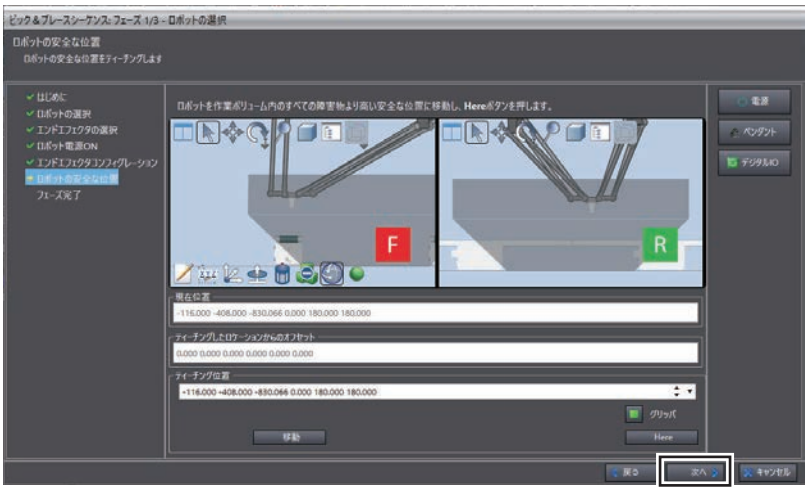
8 **画面右下の［Here］ボタンをクリックします。**



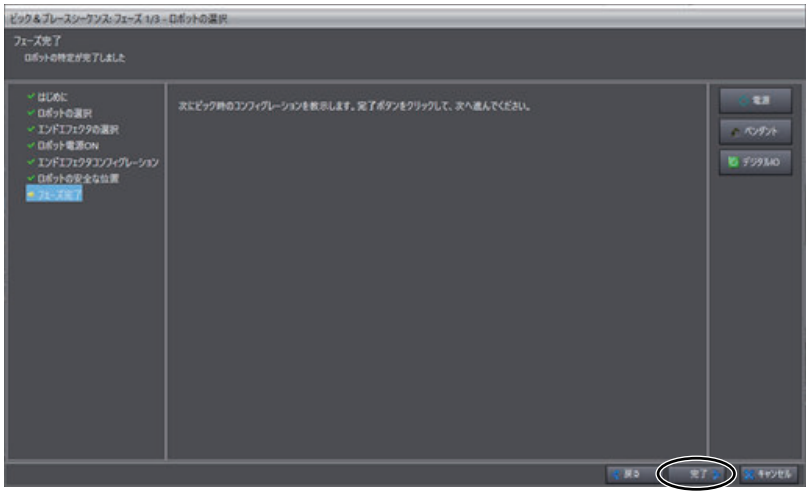
画面下の［ティーチング位置］ボックス内の値が、現在位置と同じ値に変更されます。



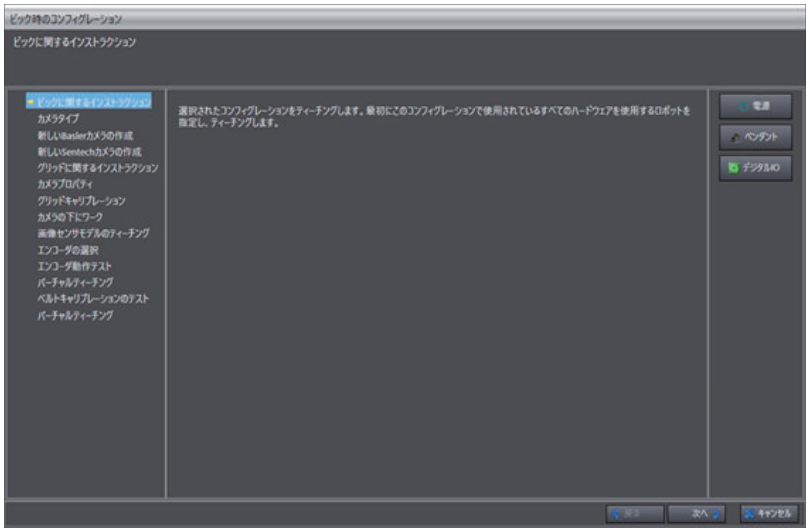
9 **画面右下の［次へ］ボタンをクリックします。**



10 **画面右下の［完了］ボタンをクリックします。**

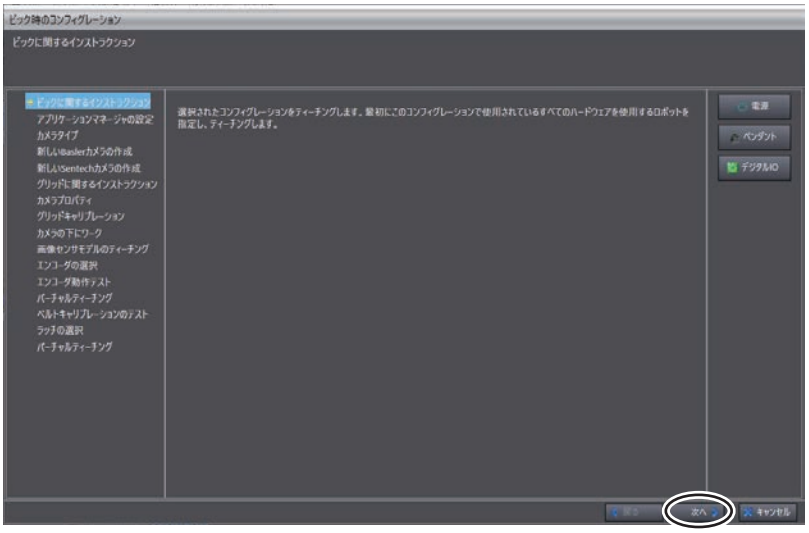


［ピック＆プレースシーケンス: フェーズ 1/3 – ロボットの選択］画面が閉じ、［ピック時のコンフィグレーション］画面がポップアップで表示されます。

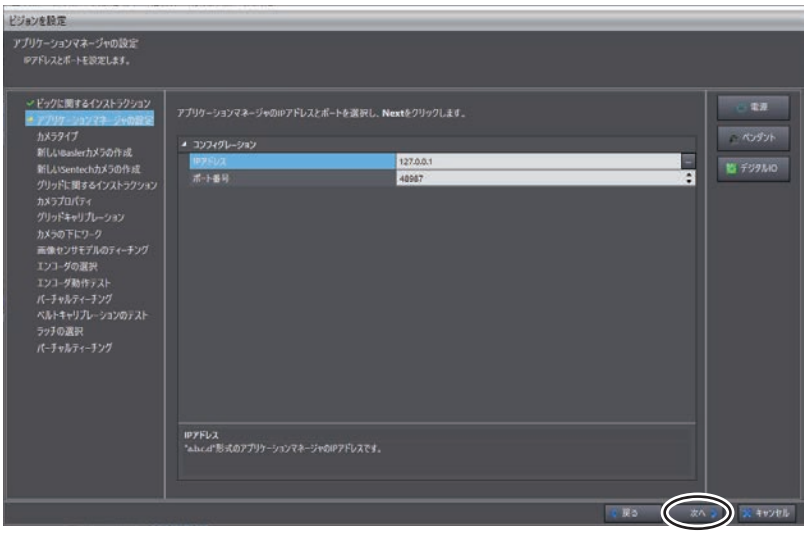


カメラ、コンベア、エンコーダの設定をする

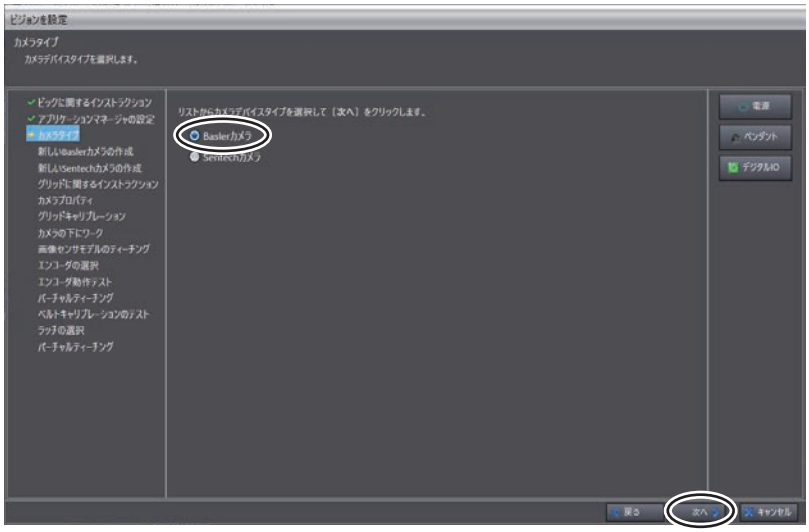
1 **画面右下の［次へ］ボタンをクリックします。**



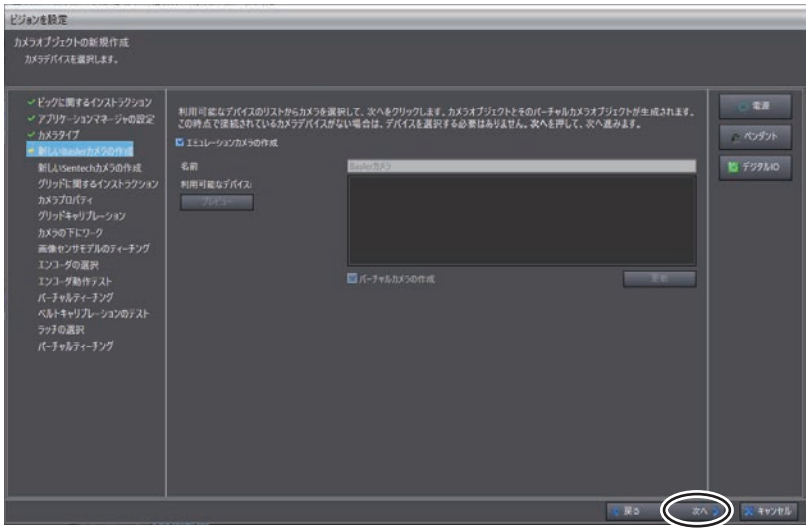
2 **［アプリケーションマネージャの設定］の画面で、［IP アドレス］が「127.0.0.1」になっていることを確認し、［次へ］ボタンをクリックします。**



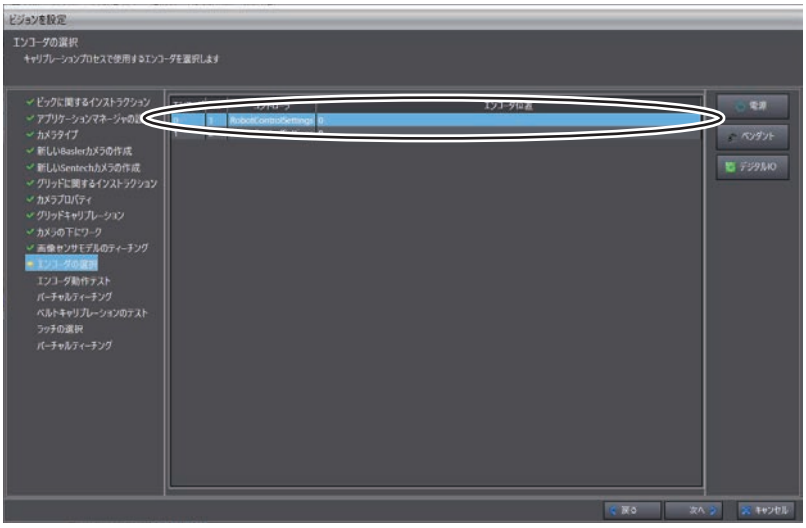
3 **本ガイドでは、システム構成にしたがって形 24114 Basler カメラを使用しますので、［Basler カメラ］が選択されていることを確認し、画面右下の［次へ］ボタンをクリックします。**



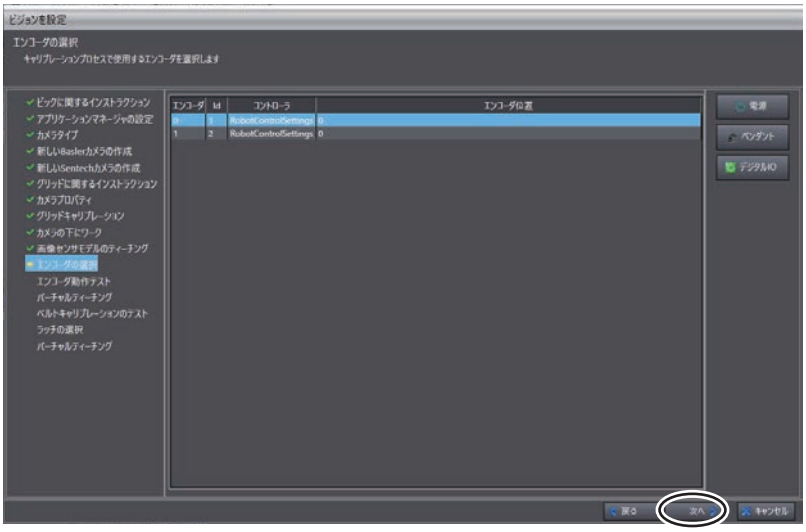
4 **画面右下の［次へ］ボタンをクリックします。**



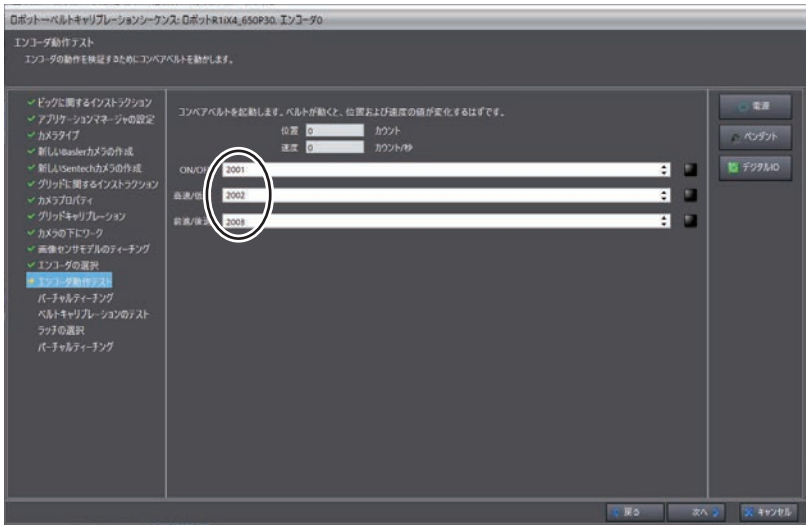
5 **本ガイドのエンコーダの配線に合わせて、画面中央のリストからエンコーダが［0］の行をクリックします。配線のコネクタのエンコーダ 1 番は、本画面の［0］に相当します。**
配線については、「4-4-8 ロボットとエンコーダ間を配線する (P.4-99)」を参照してください。



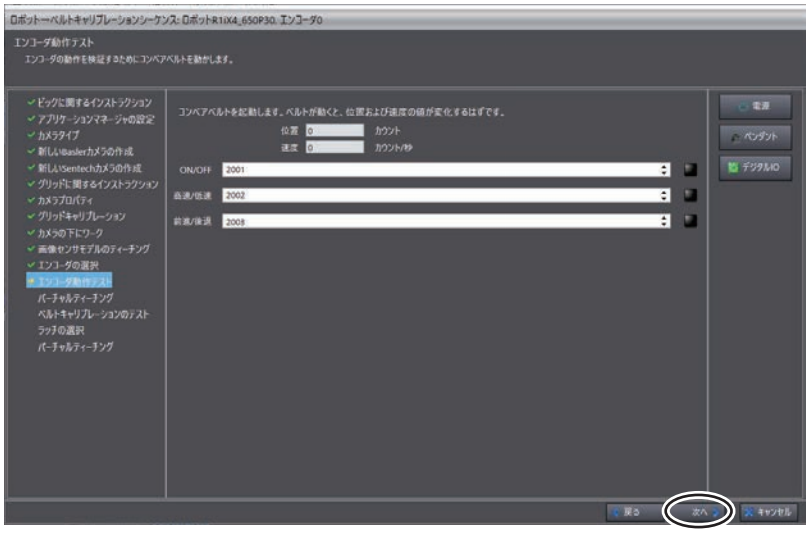
6 **画面右下の［次へ］ボタンをクリックします。**



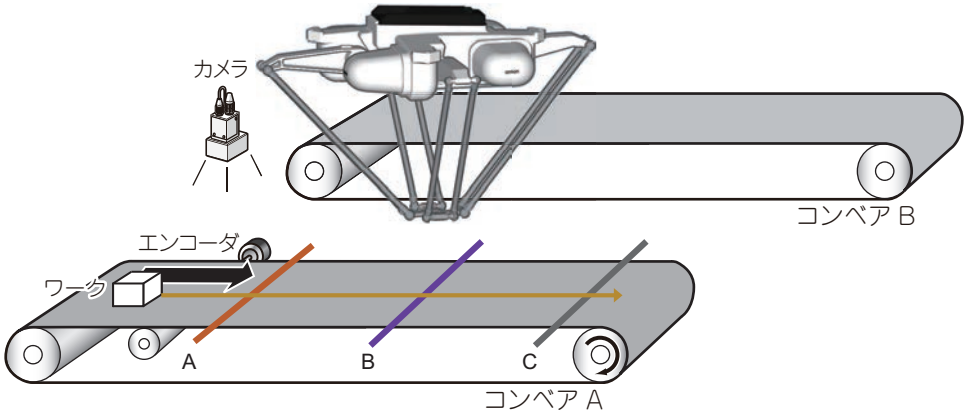
7 **本ガイドの V+デジタル I/O 割付で割り当てたコンベアの運転信号を設定します。**
画面中央の［ON/OFF］、［高速/低速］および［前進/後退］ボックスに値を入力します。
［ON/OFF］には「2001」を、［高速/低速］には「2002」を、［前進/後退］には「2003」を設定します。
V+デジタル I/O 割付については、「V+デジタル I/O (P.4-7)」を参照してください。



8 **画面右下の［次へ］ボタンをクリックします。**



9 **［コンベアの位置決め］、［上流側リミット］、［ピックリミット］、［下流側リミット］および［ベルト幅］の設定を行います。**

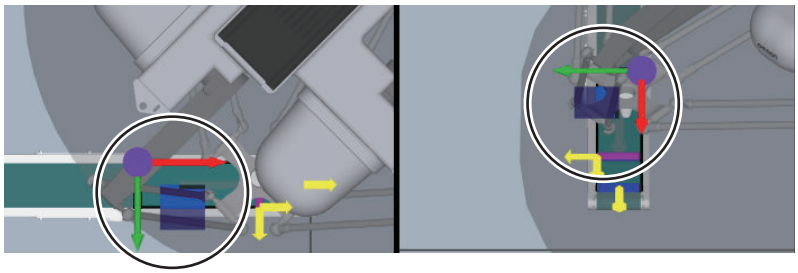


記号	項目名	内容
A	上流側リミット	コンベア上流のピック位置限界です。 ［上流側リミット］で指定した位置より上流のコンベア位置では、ロボットはピックを行いません。
B	ピックリミット	コンベア下流のピック位置準限界です。 ロボットはプレース完了時に、次にピックするワークの指定とピック位置計算を行います。その際、［ピックリミット］で指定した位置より下流にあるワークは、ピック対象から除外します。
C	下流側リミット	コンベア下流のピック位置限界です。 ［下流側リミット］で指定した位置より下流のコンベア位置では、ロボットはピックを行いません。

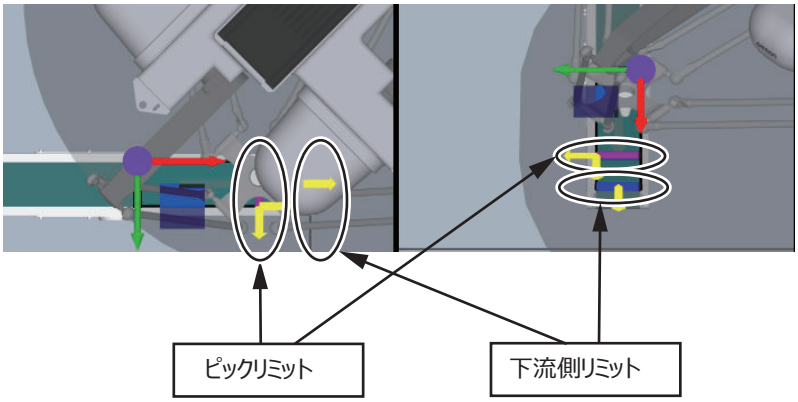


3D 表示画面でドラッグ&ドロップする方法、または［ベルト制御］ボックス内に数値を直接入力して設定する方法のいずれかで設定します。

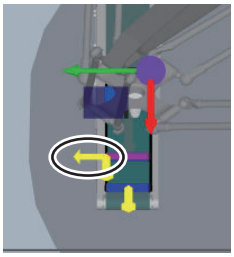
- 3D 表示画面でドラッグ&ドロップして設定する方法
  3D 表示画面の操作方法は、「A-2 3D 表示画面の操作方法 (P.A-21)」、および『Sysmac Studio ロボット統合システム構築機能 オペレーションマニュアル IPC Application Controller 編 (SBCA-483)』を参照してください。
  コンベアの位置は、緑、赤、青矢印をドラッグ&ドロップすることで操作可能です。



［上流側リミット］は、紫の点と緑の矢印（［ベルト座標変換］の原点および Y 軸）で示された位置となります。
また、［ピックリミット］は紫線、［下流側リミット］は青線で示されています。［ピックリミット］および［下流側リミット］は、黄矢印をドラッグ&ドロップすることで操作可能です。
いずれもロボットの動作範囲に収まる箇所に位置決めしてください。



ベルト幅は、下図の枠線で示した黄矢印で操作可能です。



・［ベルト制御］ボックス内に数値を直接入力して設定する方法

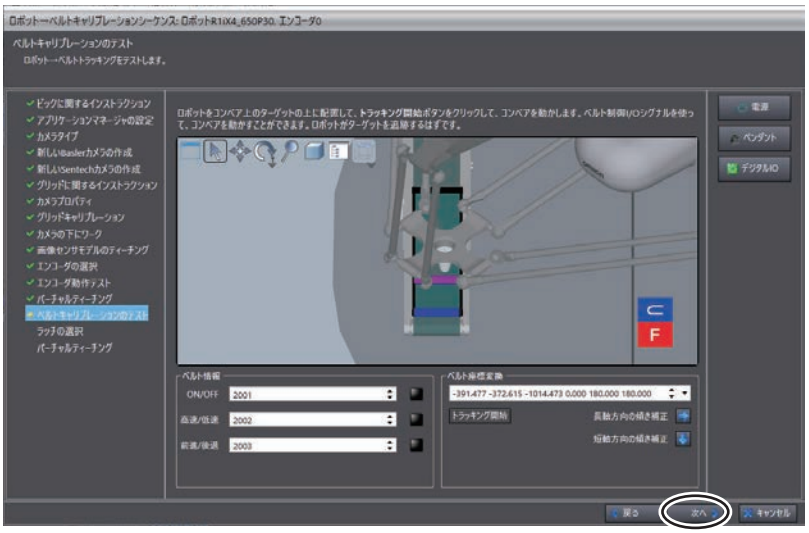


［上流側リミット］は、ベルト座標変換原点と一致します。そのため、［ベルト座標変換］数値ボックスでワールド座標からのオフセットとして入力します。
［ピックリミット］および［下流側リミット］は、各々の数値ボックスに［上流側リミット］からの相対距離(mm)で入力します。
［ベルト幅］は［ベルトウィンドウの幅］数値ボックスに mm 単位で入力します。
［エンコーダスケール］は、数値ボックスに(mm/エンコーダ)の値を入力します。

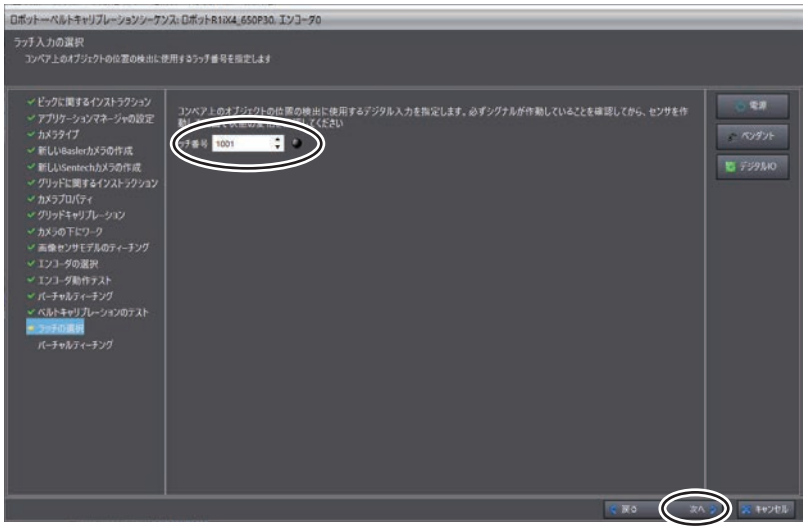
10 画面右下の［次へ］ボタンをクリックします。



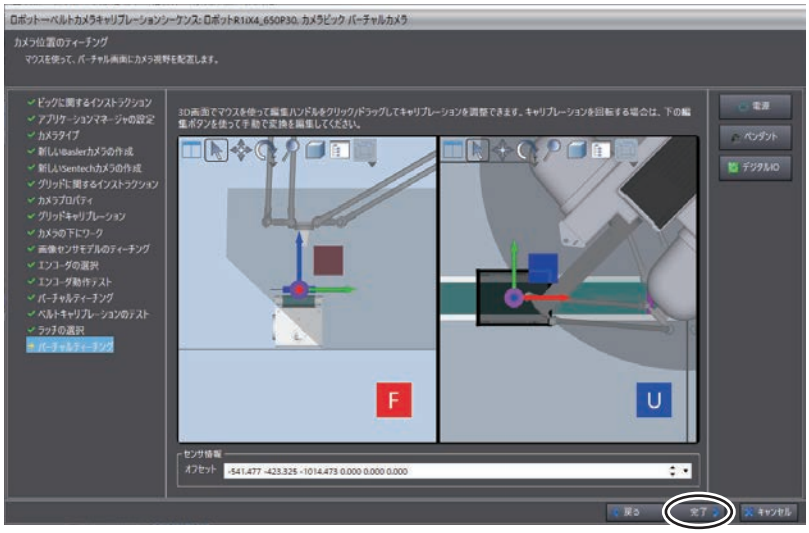
11 **[ON/OFF] ボタンをクリックして点灯させた後、[トラッキング開始] ボタンをクリックすることで，ロボットが正常にコンベア上をトラッキングできるか確認します。問題がなければ、画面右下の [次へ] ボタンをクリックします。**



12 **ラッチ番号の設定が正しいことを確認して [次へ] ボタンをクリックします。**



13 画面右下の［完了］ボタンをクリックします。



ティーチングする



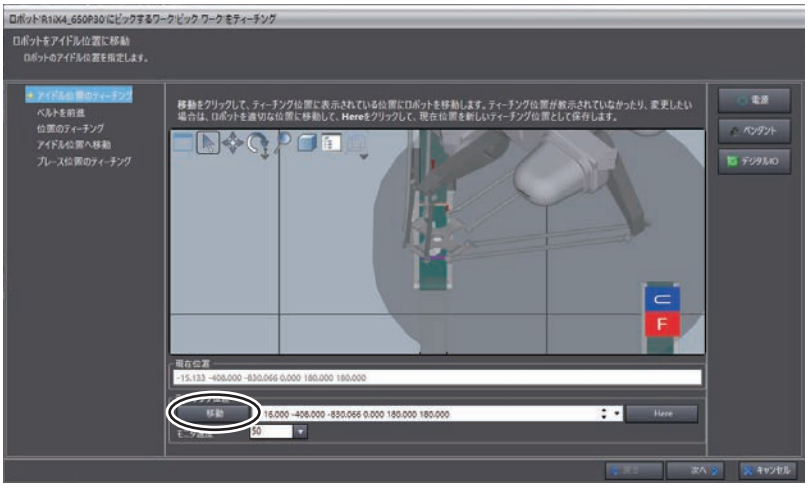
参考

関連動画

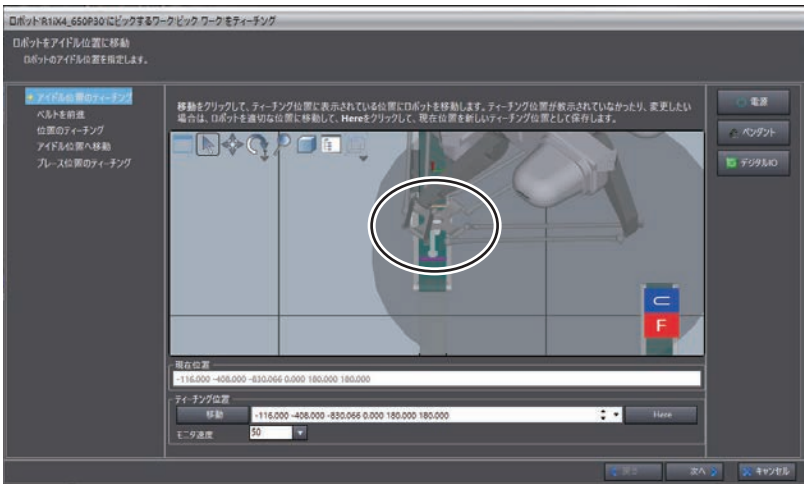
- ロボット動作設定 Pack Manager でのティーチング
  https://www.fa.omron.co.jp/video/detail/6264519169001/
  Pack Manager の機能を使用してシミュレーションで動的ピック＆プレース装置のピック位置、プレース位置をティーチングする手順を説明しています。

● ロボットアイドル位置のティーチング

1 画面下の［移動］ボタンをクリックします。

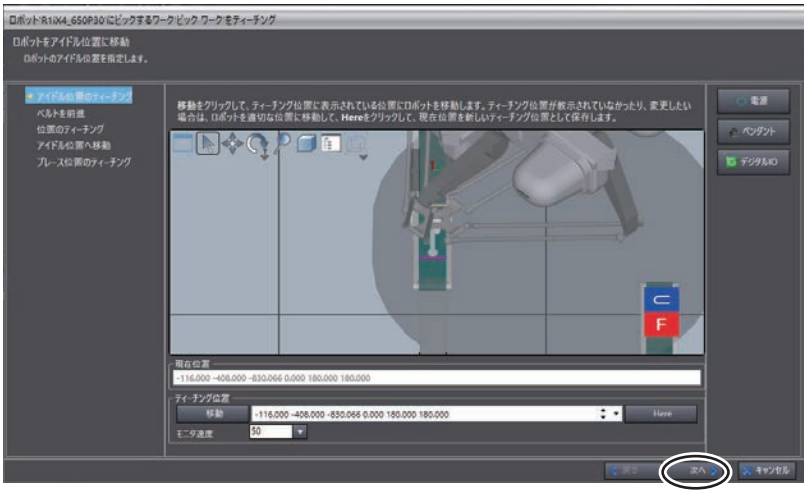


ロボットがアイドル位置まで移動します。

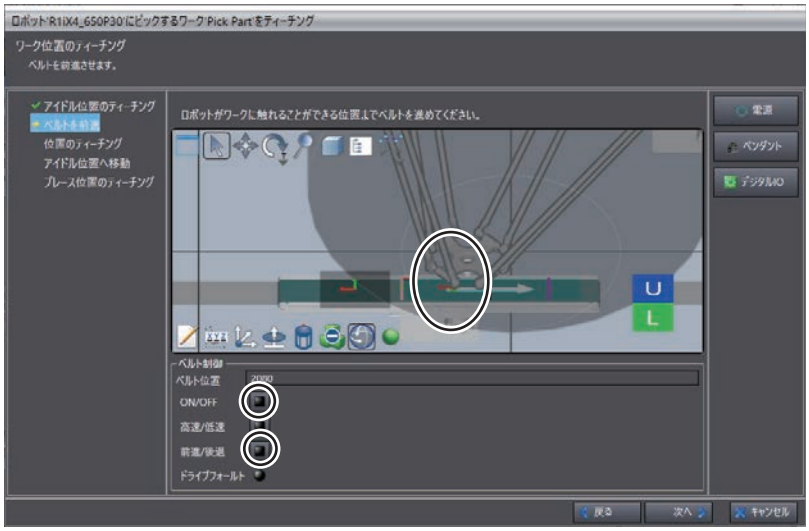


2 **ピック位置およびプレース位置に直線補間で移動可能な位置に、アイドル位置を調整します。完了したら［Here］ボタンをクリックします。**

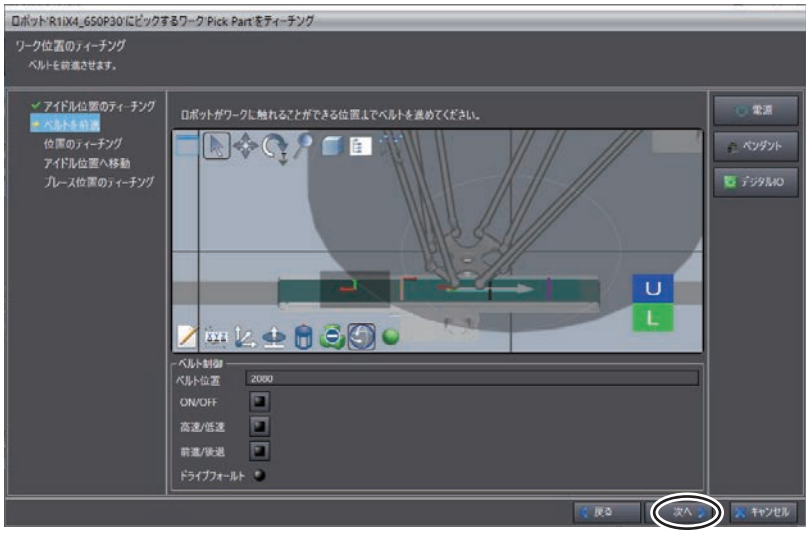
3 **画面右下の［次へ］ボタンをクリックします。**



4 **ロボットがワークに触れることができる位置までベルトを進めます。**

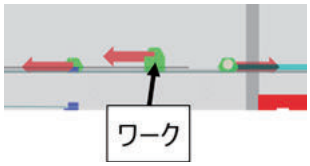


5 ［次へ］ボタンをクリックします。

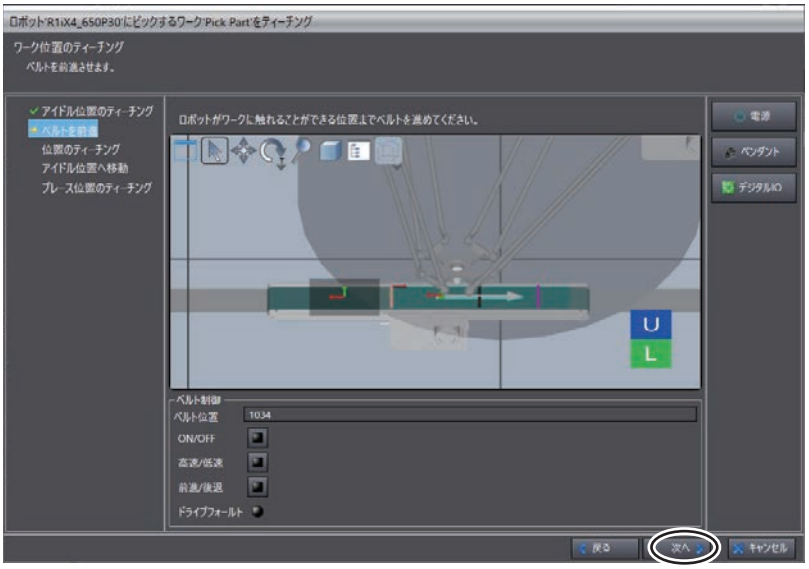


● ピック位置およびアプローチ高さ、デパート高さのティーチング

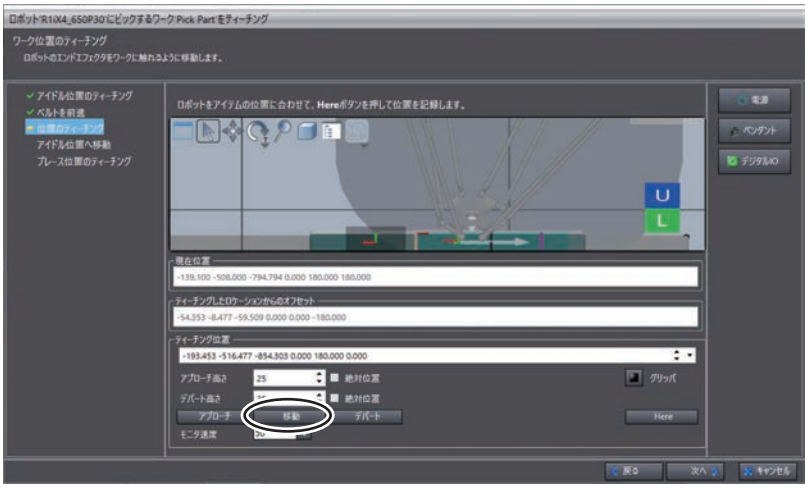
1 画面下の［ベルト制御］ボックス内の［ON/OFF］、［前進/後退］ボタンを押して下図に示す緑色のワークをコンベア上で移動させます。その後、ロボットの動作範囲内かつベルトウィンドウ内でワークを停止してください。



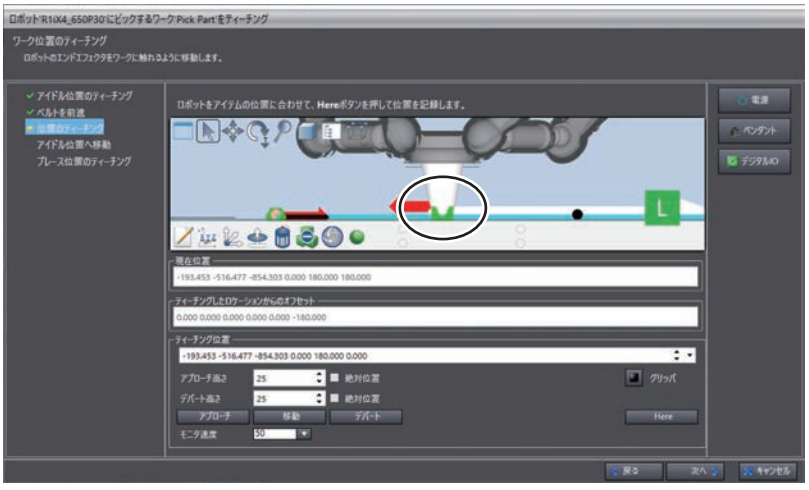
2 画面右下の［次へ］ボタンをクリックします。



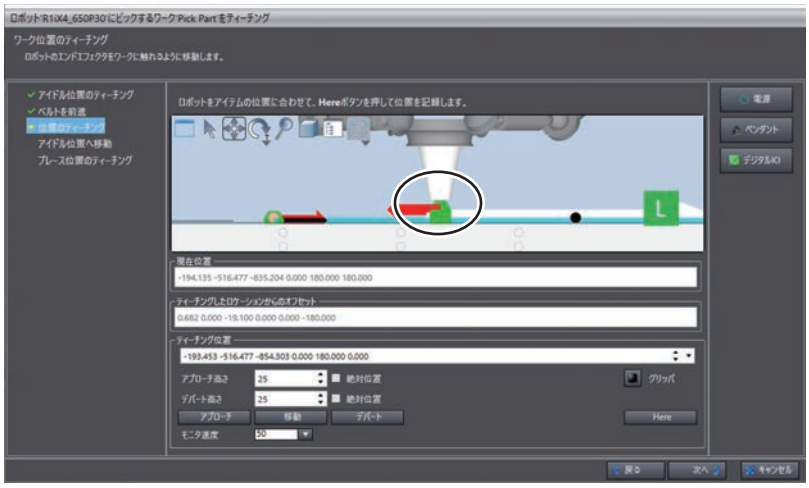
3 **画面下の［移動］ボタンをクリックします。**



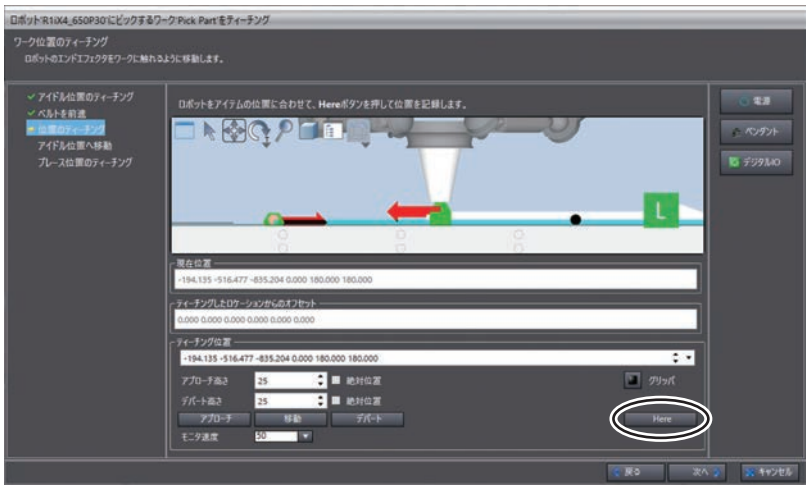
コンベア上のワーク底面にツール先端がくるように、ロボットが移動します。



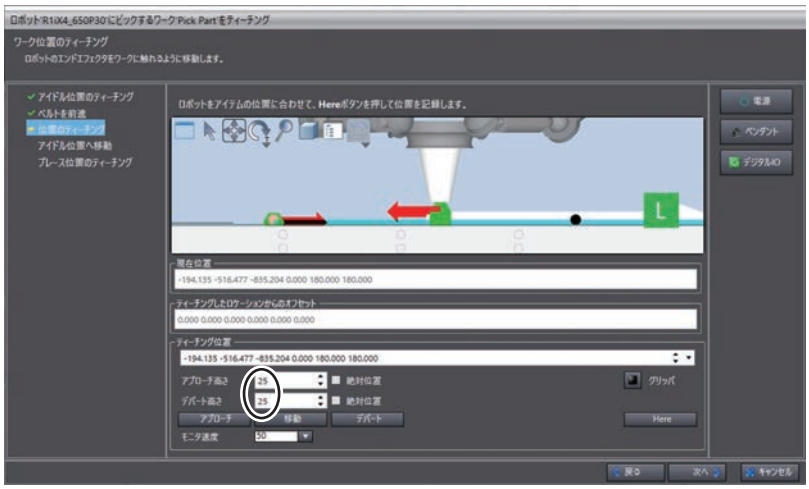
4 **ワーク上面にエンドエフェクタ先端がくるように、ロボットをＺ軸方向に動かします。**



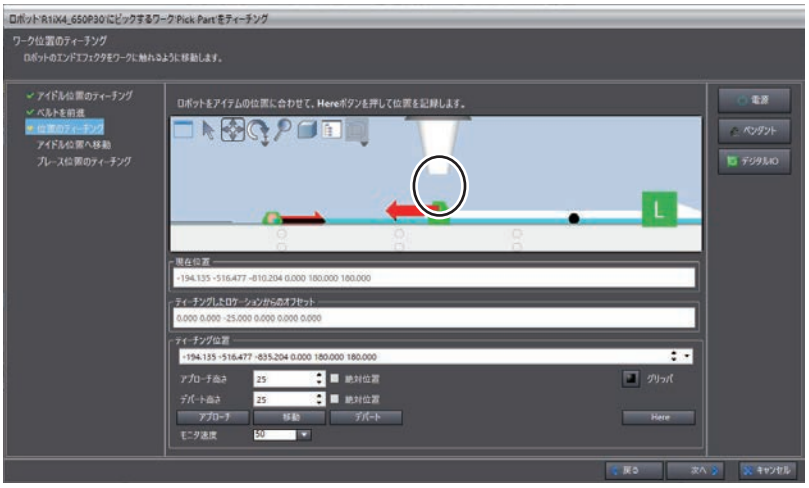
5 **[HERE] ボタンをクリックします。**



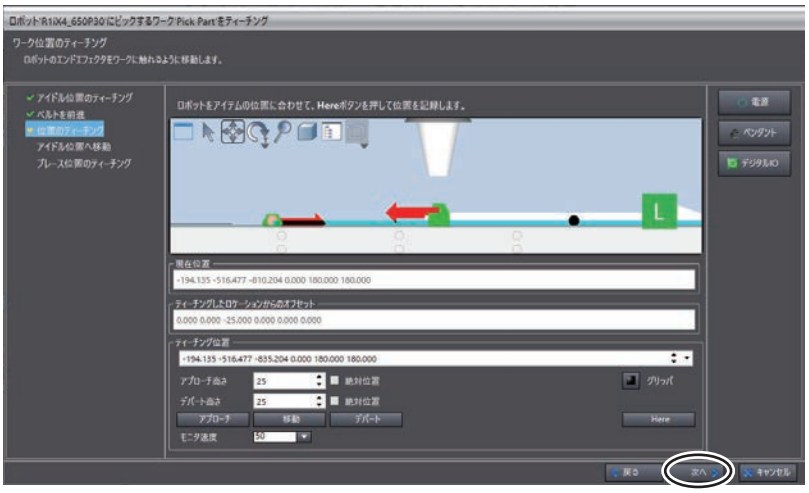
6 **画面下の [ティーチング位置] ボックス内の [アプローチ高さ] および [デパート高さ] 数値ボックスに、ワークピック位置への移動前後の昇降高さを入力します。**



[アプローチ] および [デパート] ボタンを押すことで、実際の高さまでロボットを動作させ、確認することができます。

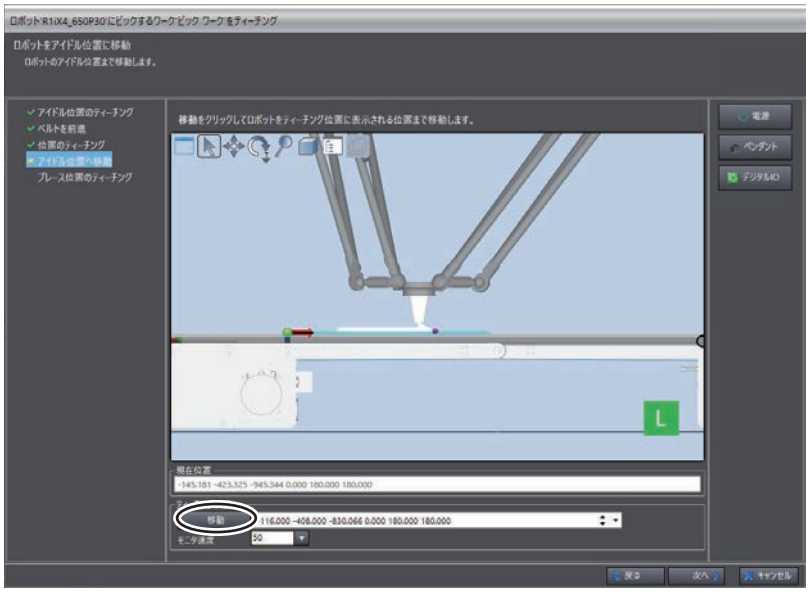


7 画面右下の［次へ］ボタンをクリックします。

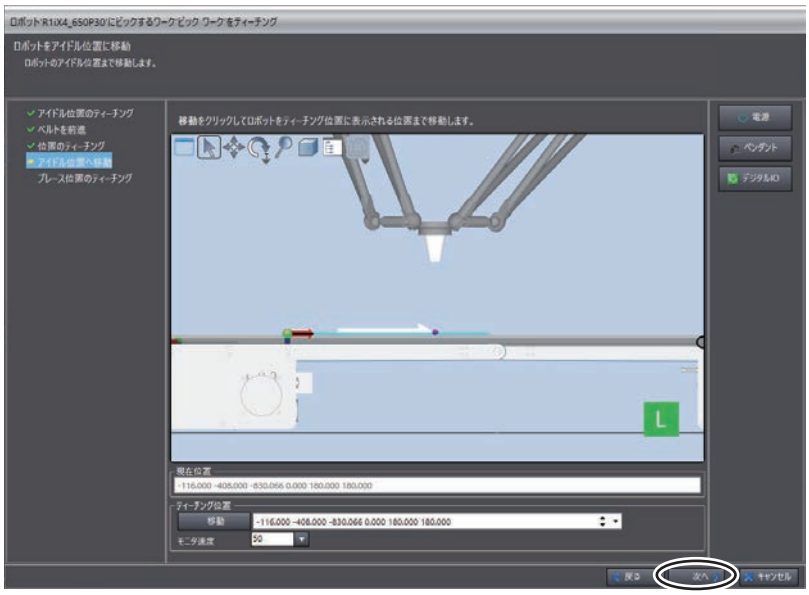


● アイドル位置への移動

1 画面下の［移動］ボタンをクリックします。



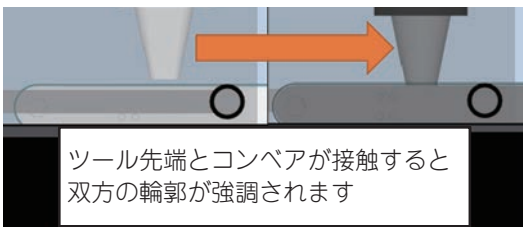
ロボットが待機位置まで移動完了したら、画面右下の［次へ］ボタンをクリックします。



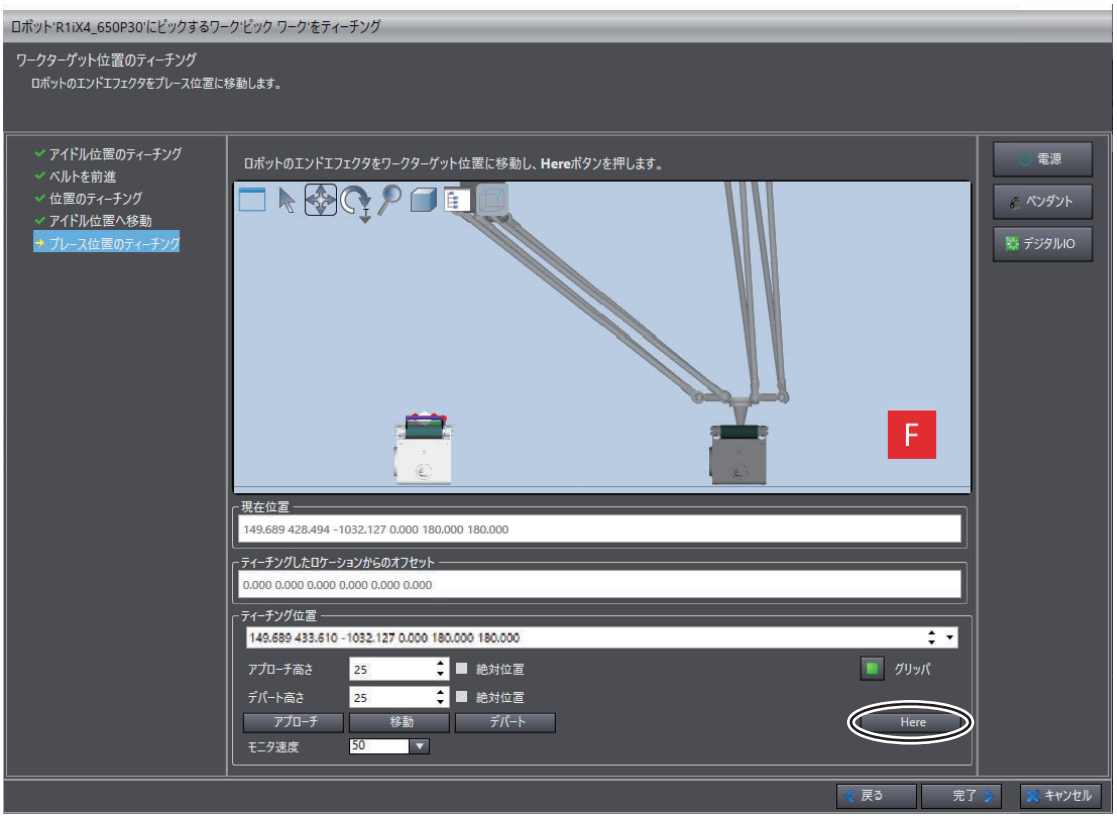
● プレース位置、およびアプローチ／デパート高さのティーチング

1 **画面中央の［3D 表示画面］でロボットを操作し、コンベア上のワーク配置位置まで移動します。**

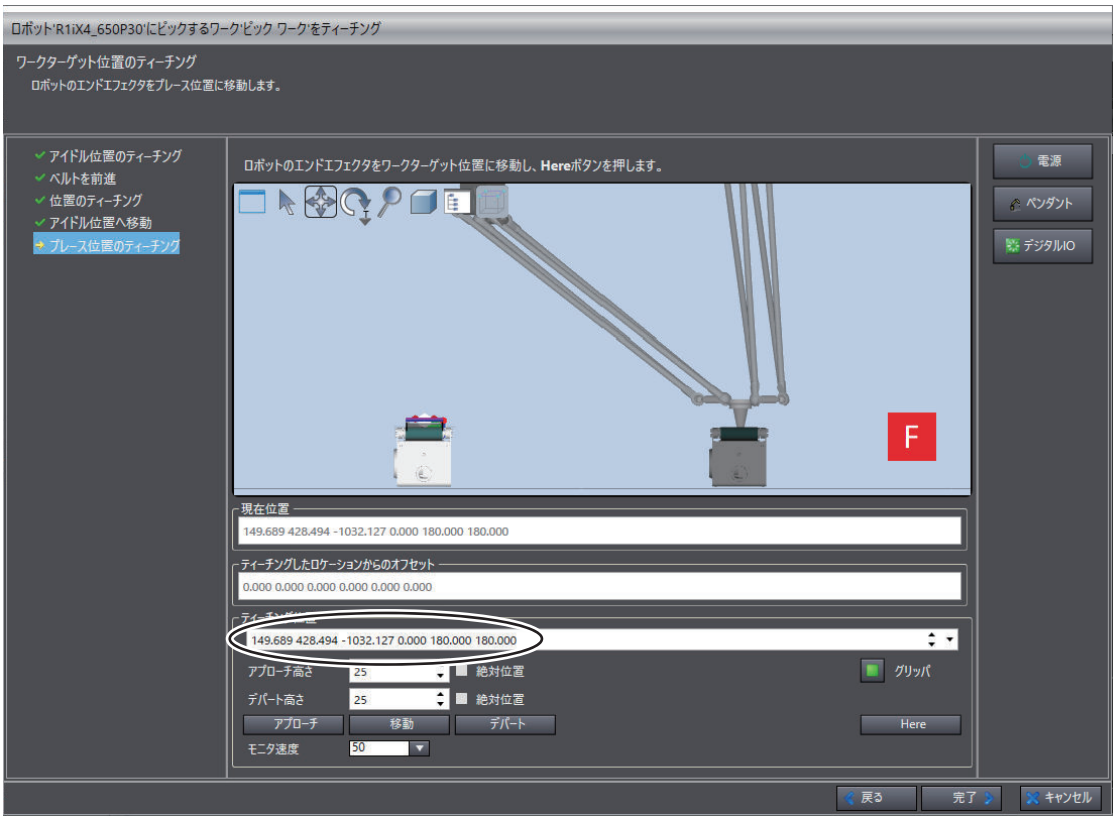
ツール先端とコンベアが接触した場合、ツール先端およびコンベアの輪郭が強調表示されます。強調表示を参考に配置位置を決めてください。



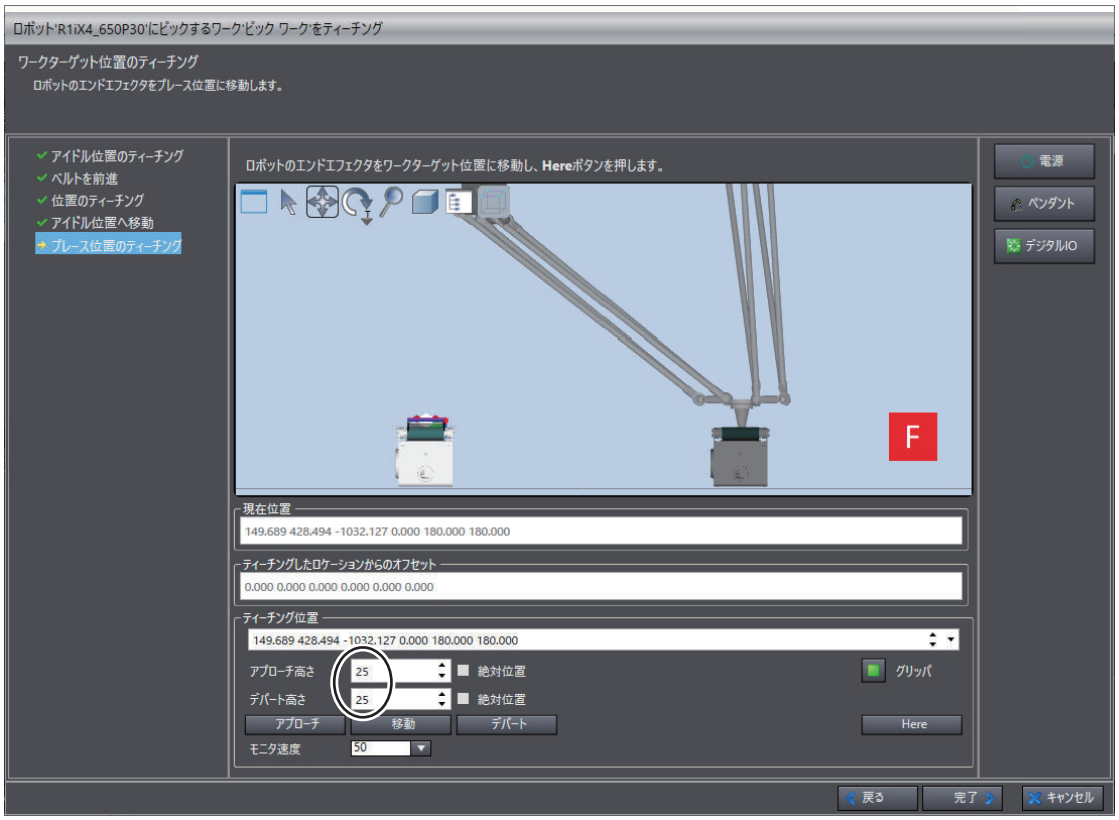
2 **画面右下の［Here］ボタンをクリックします。**



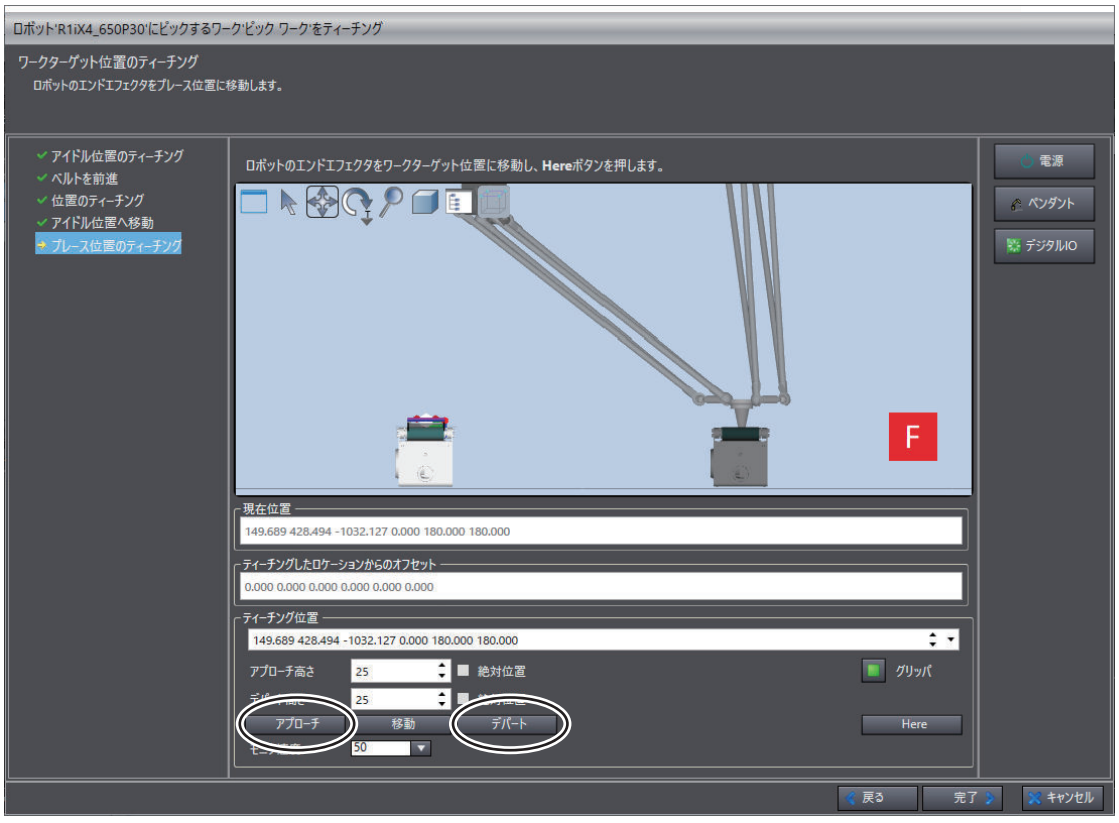
画面下の［ティーチング位置］の値が現在位置と同じ値に設定されます。ティーチング位置をより細かく調整する場合は、［ティーチング位置］の値を調整し、［移動］ボタンをクリックしてロボットを移動させてください。



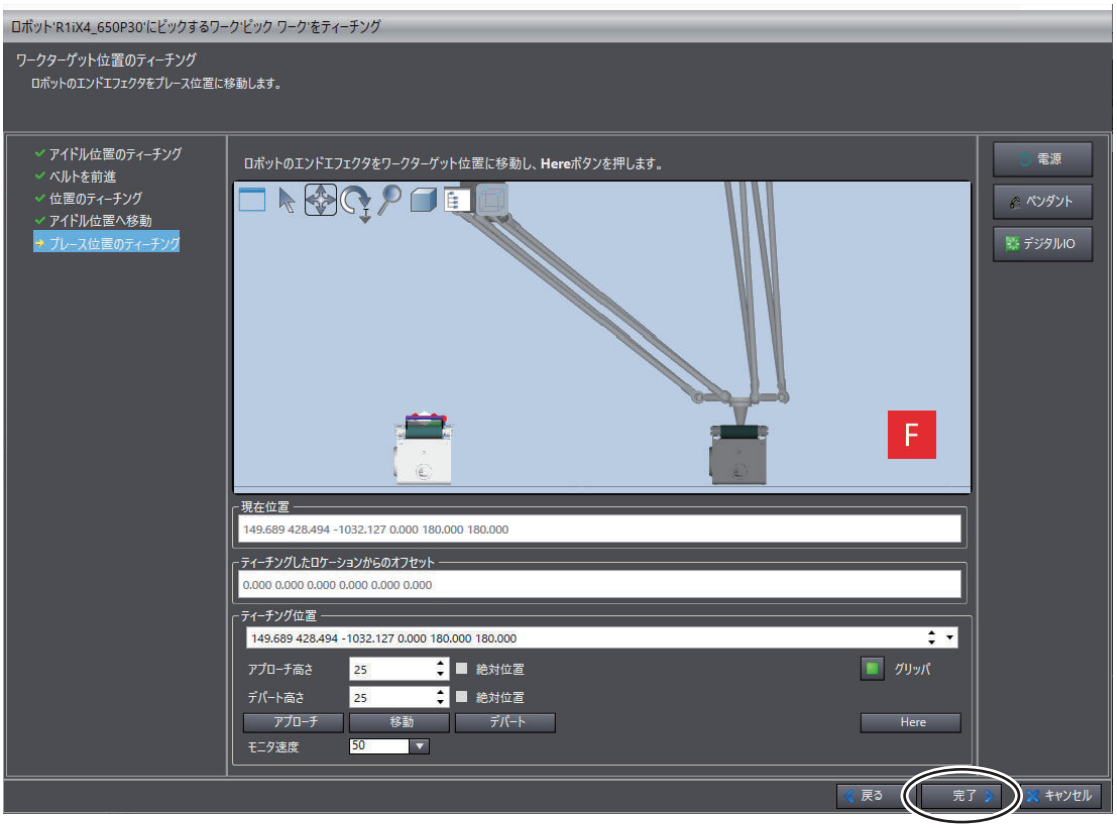
3 **画面下の［ティーチング位置］ボックス内の［アプローチ高さ］および［デパート高さ］に、ワーク配置経由点の昇降高さを入力します。**



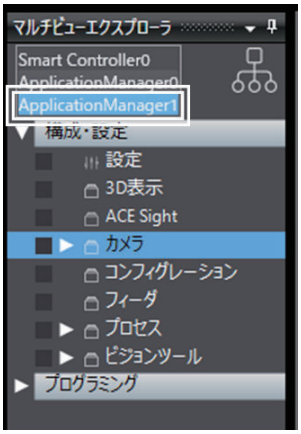
［アプローチ］および［デパート］ボタンを押すことで、設定した高さでのロボットの動作確認が可能です。



4 **画面右下の［完了］ボタンをクリックします。**



ティーチング画面が閉じ、メインウィンドウのマルチビューエクスプローラの上部にあるドロップダウンリストに［ApplicationManager1］が追加されます。



追加で設定を行う

● コンベアをアクティブ制御にする

Pack Manager からのコンベア制御を有効にするため、以下の操作を行います。

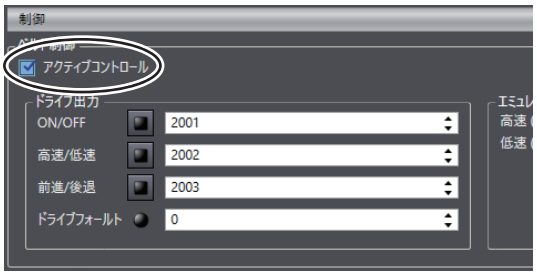
1 **マルチビューエクスプローラのデバイスは、［ApplicationManager1］を選択します。**



2 **マルチビューエクスプローラから［構成・設定］–［プロセス］–［ピックベルト］をダブルクリックしてピックベルト画面を開きます。**



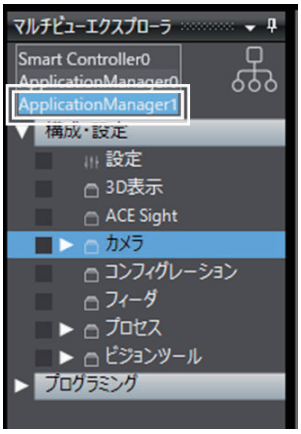
3 **［ベルト制御］の［アクティブコントロール］チェックボックスにチェックを入れます。**



4-3-7 C#プログラムを作成する

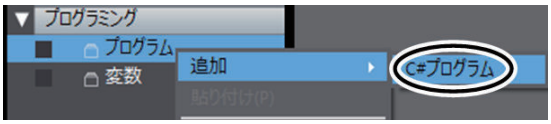
ここでは、C#プログラムの作成手順を説明します。作成するプログラムについては「C#プログラム (P.4-10)」を参照してください。

1 **マルチビューエクスプローラのデバイスで、［ApplicationManager1］を選択します。**

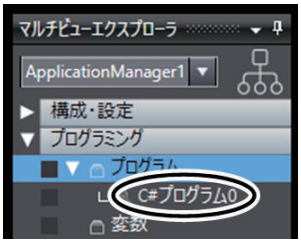


2 **マルチビューエクスプローラの［プログラミング］－［プログラム］を右クリックします。**
コンテキストメニューが表示されます。

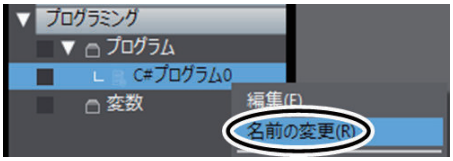
3 **コンテキストメニューから［追加］－［C#プログラム］を選択します。**



［プログラム］に［C#プログラム 0］が追加されます。



4 **［C#プログラム 0］を右クリックし、コンテキストメニューから［名前の変更］を選択します。**



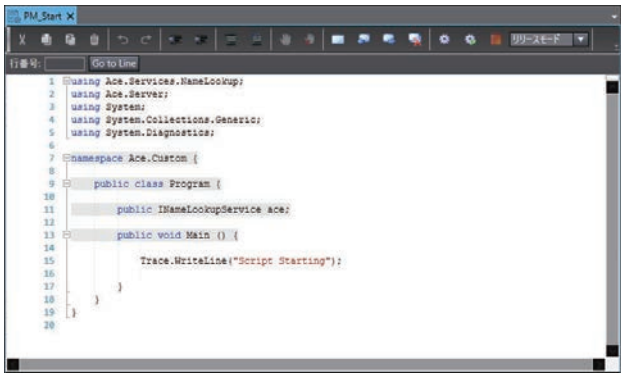
［C#プログラム 0］の名前を編集できる状態になります。



5 **「PM_Start」を入力し Enter キーを押します。**
プログラム名が“PM_Start”に変更されます。



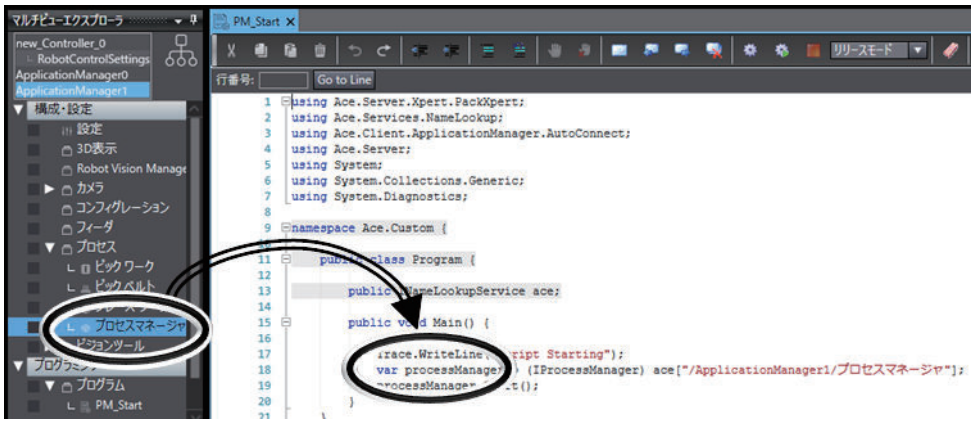
6 **追加したプログラムをダブルクリックします。**
エディットウィンドウに編集画面が表示されます。



7 **手順 3 から手順 6 までを繰り返し、「C#プログラム (P.4-10)」の内容で PM_Start と PM_Stop のプログラムを作成します。**

使用上の注意

C#プログラムで指定するプロセスマネージャの文字列は、プロジェクトファイルを作成したときの OS の言語により異なります。ここではマルチビューエクスプローラで［ApplicationManager1］の［構成・設定］–［プロセス］にある［プロセスマネージャ］をドラッグし、C#エディタ上にドロップすることで、適切な文字列が自動的に設定されます。



4-3-8 V+のプログラムと変数の電源 ON 時の自動読込を設定する

作成した V+プログラムおよび V+のグローバル変数は、プロジェクトの初期設定では、ロボット統合 CPU ユニットの電源 ON 時に自動的に読み込まれません。
ここでは、ロボット統合 CPU ユニットの電源 ON 時に、V+プログラムおよび V+のグローバル変数を自動的に読み込むための設定の手順を説明します。

参考

関連動画

- V+プログラムと変数の電源 ON 時の自動読み込みの設定
  https://www.fa.omron.co.jp/video/detail/6294914962001/
  電源 ON 時に V+プログラムとグローバル変数を自動的に読み込むための設定手順を説明しています。

1 **マルチビューエクスプローラのデバイスは、［RobotControlSettings］を選択します。**



2 **マルチビューエクスプローラの［構成・設定］をクリックします。**



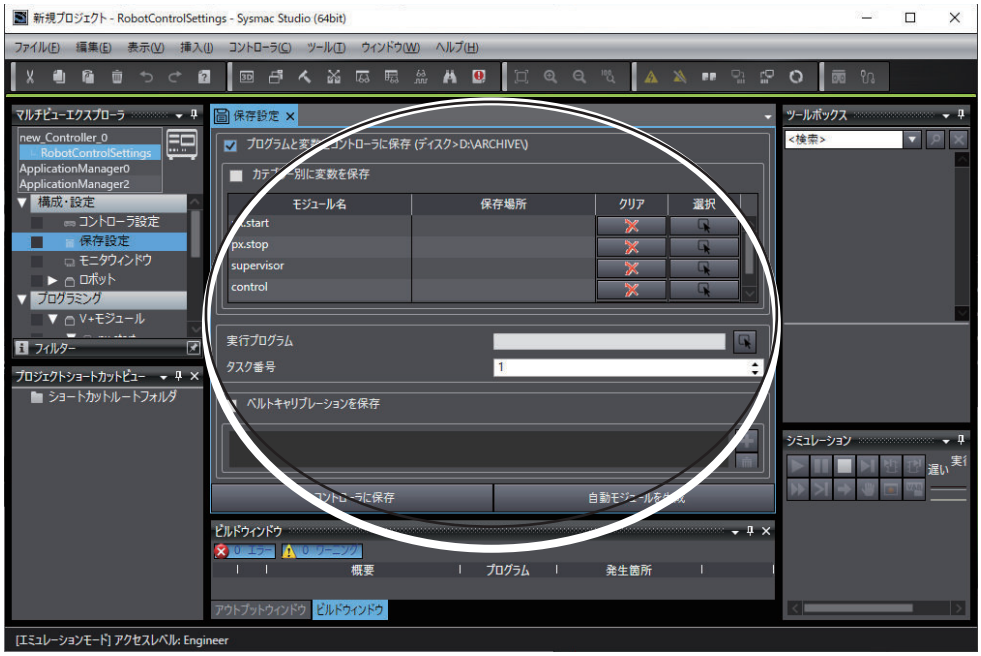
［構成・設定］の下にツリーが展開されます。



3 **［保存設定］をダブルクリックします。**



エディットウィンドウに［保存設定］タブが開きます。

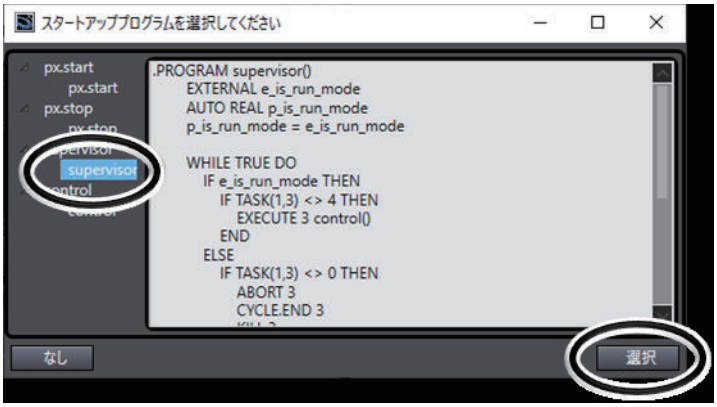


4 **［実行プログラム］に「supervisor」が表示されていないことを確認し、右のボタンをクリックします。**



［スタートアッププログラムを選択してください］画面が表示されます。

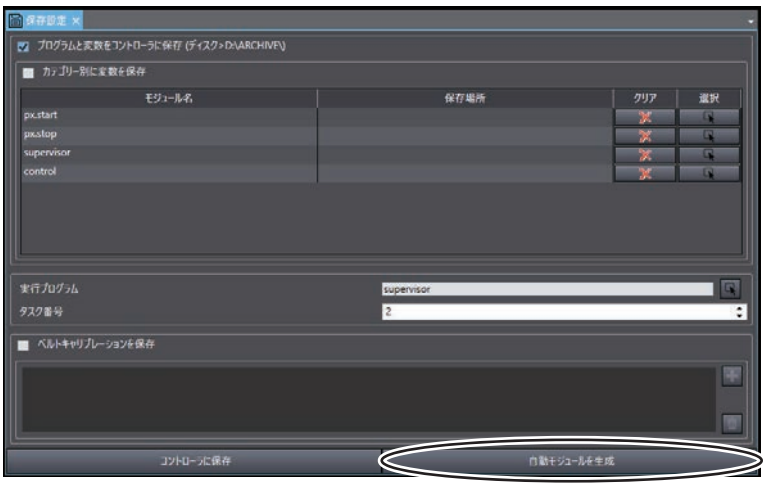
5 **［supervisor］-［supervisor］をクリックして［選択］ボタンをクリックします。**



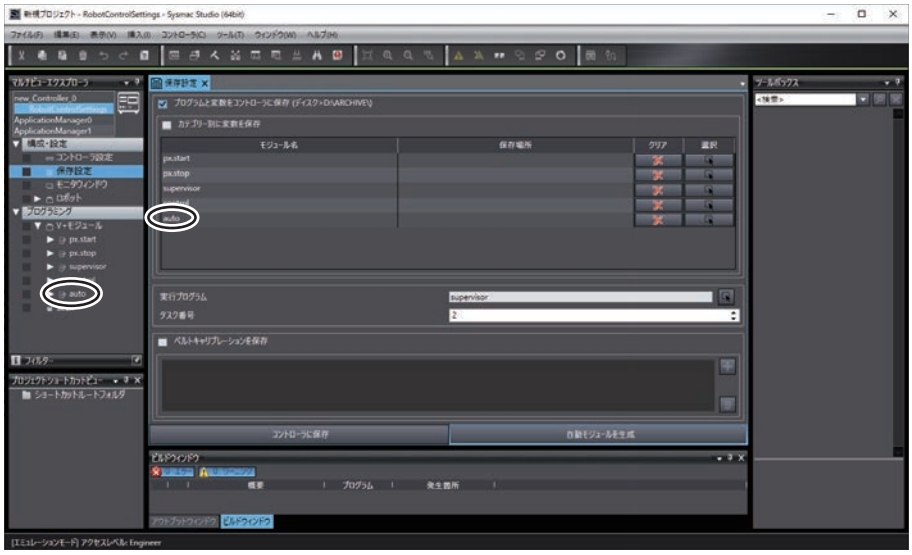
6 **［タスク番号］を「2」に設定します。**



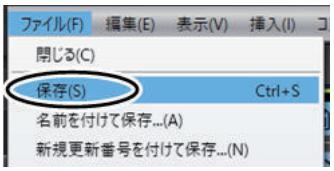
7 **［保存設定］の［自動モジュールを生成］ボタンをクリックします。**



［保存設定］タブのモジュールリストに［auto］が追加されます。また、マルチビューエクスプローラのツリーに［プログラミング］-［V+モジュール］-［auto］が追加されます。



8 **メニューバーの［ファイル］–［保存］をクリックします。**



これで、ロボット統合 CPU ユニットの電源 ON 時に、V+プログラムおよび V+のグローバル変数を自動的に読み込むための設定ができました。プログラム［auto］はロボット統合 CPU ユニットの電源 ON 時に自動的に実行され、保存された V+プログラムおよび V+のグローバル変数を読み込みます。

プログラム［auto］は通常の V+プログラムと異なるコマンドプログラムと呼ばれる形式で作成されます。コマンドプログラムに関する詳細については、『eV+3 ユーザーズマニュアル (SBCE-486)』を参照してください。

4-3-9 プログラムを実行して動作を確認する

「4-3-4 シーケンス制御プログラムを作成する (P.4-23)」で作成したシーケンス制御プログラムをシミュレーションで実行する手順を説明します。
ウォッチウィンドウでシーケンス制御プログラムのグローバル変数「gStart」を FALSE から TRUE にすることで、プログラムの実行を開始できます。
なお、V+プログラムは、本手順にて実行されたシーケンス制御プログラムから呼び出されて実行されます。

使用上の注意

本手順を行う場合は、エミュレーションモードでプロジェクトファイルを開いていることを確認してください。確認方法については、「ロボットのシミュレーション機能を有効にする (P.3-59)」を参照してください。

参考

プログラムのデバッグ機能については、『Sysmac Studio Version 1 オペレーションマニュアル (SBCA-470)』および『Sysmac Studio ロボット統合システム構築機能 オペレーションマニュアル ロボット統合 CPU ユニット編 (SBCA-464)』を参照してください。

参考

関連動画

- シミュレーションでの動作確認
  https://www.fa.omron.co.jp/video/detail/6264519415001/
  Sysmac Studio のシミュレーション機能を使用してロボットの動作を確認する手順を説明しています。

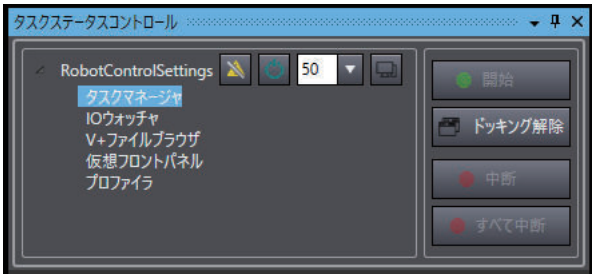
1 **マルチビューエクスプローラーのデバイスで、[RobotControlSettings] を選択します。**



2 **メインウィンドウで、メニューバーから [表示] - [タスクステータスコントロール] を選択します。**



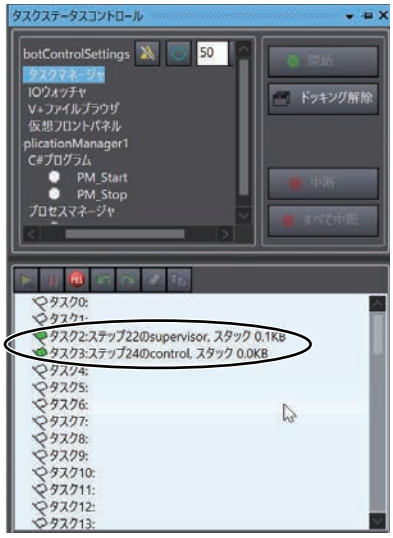
メインウィンドウの右に［タスクステータスコントロール］画面が追加されます。



3 **マルチビューエクスプローラの［プログラミング］–［V+モジュール］–［supervisor］–［supervisor］を右クリックし、［タスク上で実行］–［タスク 2］をクリックします。**



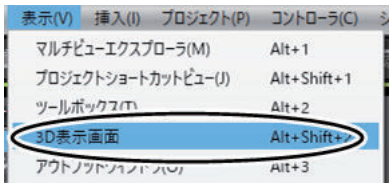
V+プログラムの supervisor がタスク 2、contorl がタスク 3 で実行されます。



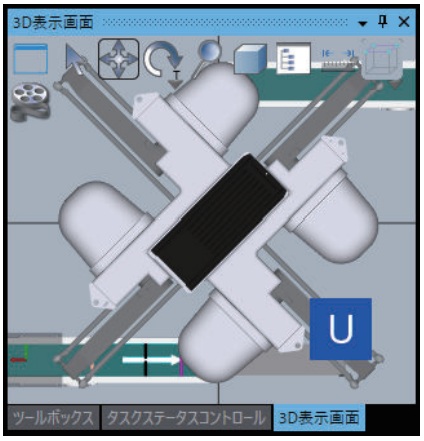
4 **マルチビューエクスプローラのデバイスで、［new_Controller_0］を選択します。**



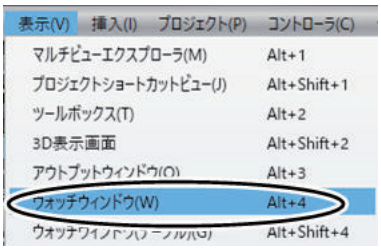
5 **Sysmac Studio のメインウィンドウのメニューバーから［表示］–［3D 表示画面］を選択します。**



メインウィンドウ内の右側に 3D 表示画面が追加されます。



6 **メインウィンドウのメニューバーから［表示］–［ウォッチウィンドウ］を選択します。**



メインウィンドウの下に［ウォッチウィンドウ（プロジェクト）1］画面が表示されます。



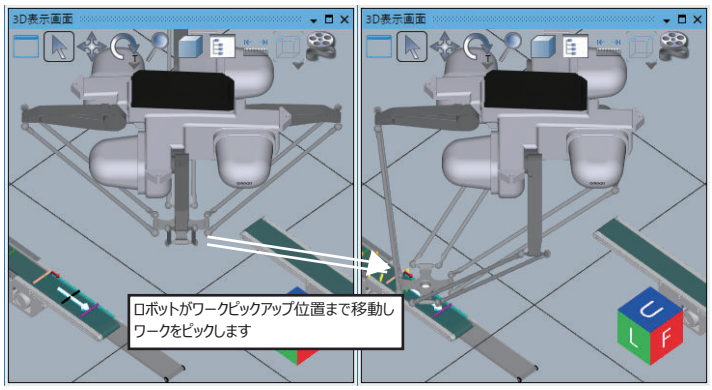
7 **［ウォッチウィンドウ（プロジェクト）1］画面の［名称］列に「gStart」と入力します。**
［モニタ値］列にプログラム［Program0］の内部変数［gStart］の現在値［False］が表示されます。



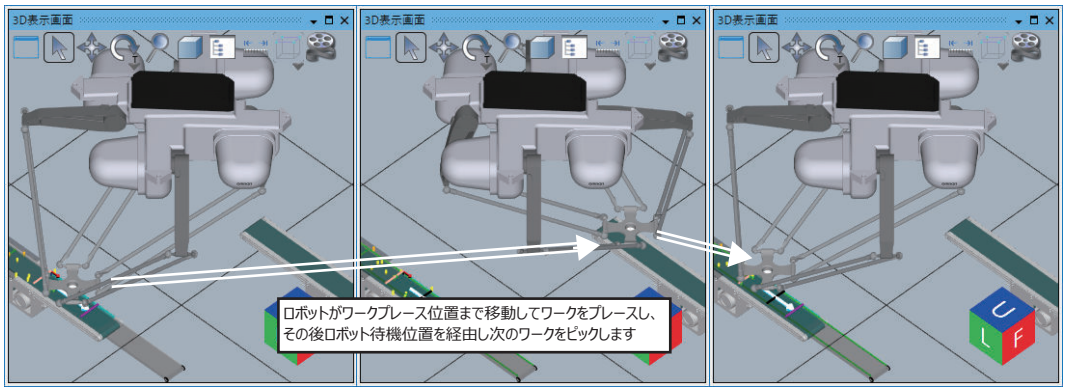
8 **［ウォッチウィンドウ（プロジェクト）1］画面で変数［gStart］の［変更］列の［TRUE］ボタンをクリックします。**



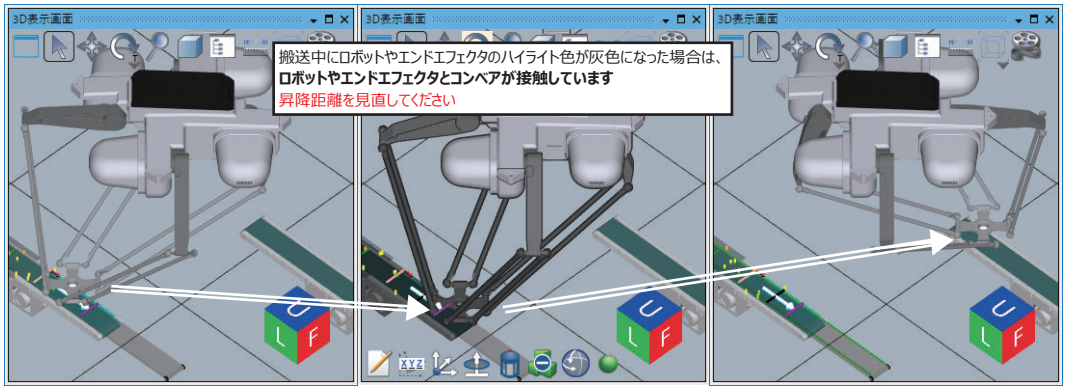
3D 表示画面でロボットが動作を開始し、ワークが流れてくるとワークをピックします。



その後、ワークの配置位置を経由してロボット待機位置まで移動した後、ワークのピック＆プレースを繰り返します。



動作過程では、ワークのピックアップ位置およびワークの配置位置以外でロボットが灰色にならないかを確認してください。灰色になる場合はロボットとコンベアが干渉しています。「ピック位置およびアプローチ高さ、デパート高さのティーチング (P.4-74)」に記載されている操作でワークのピックアップ経由点またはワークの配置経由点をより高い位置に設定するなどしてティーチングを変更してください。



これで、シーケンス制御プログラムの実行が完了しました。



参考

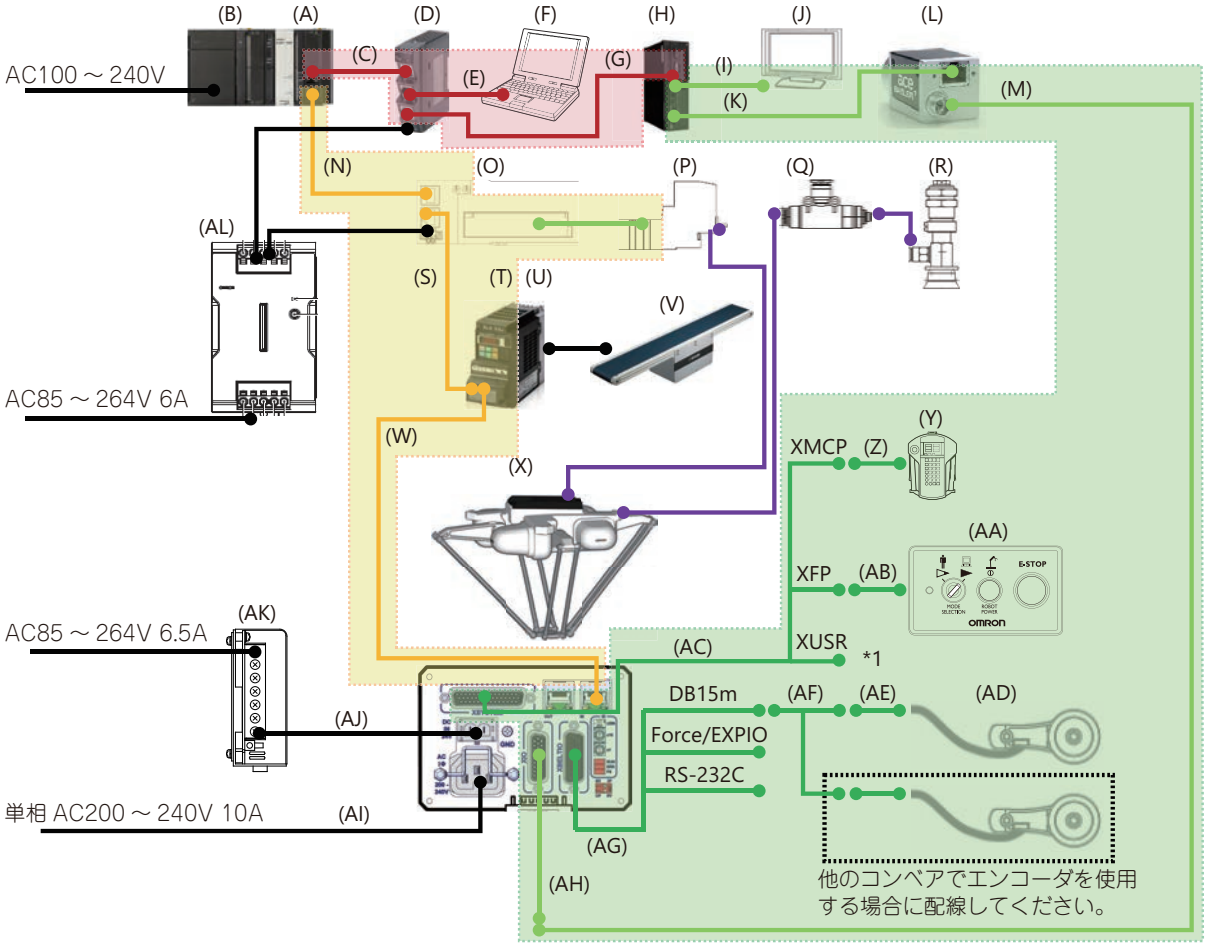
関連動画

- Process Manager とラダープログラムの連携
  https://www.fa.omron.co.jp/video/detail/6264521399001/
  シミュレーションで Process Manager とラダープログラムが連動して動作していることを確認できます。

4-4　設置、配線する

システム構成において、図中の枠で囲われている以下の信号線の配線と、ロボットのノードアドレスの設定を行います。

- ロボット統合 CPU ユニット(A)、EtherCAT デジタル I/O ターミナル(O)、インバータの EtherCAT 通信ユニット(T)、およびロボット(X)間の配線
- ロボット統合 CPU ユニット(A)、産業用スイッチングハブ(D)、パソコン(F)、および IPC Application Controller(H)間の配線
- ロボット(X)、T20 ペンダント(Y)、フロントパネル(AA)、およびエンコーダ(AD)間の配線
- IPC Application Controller(H)、ディスプレイ(J)、およびカメラ(L)間の配線
- EtherCAT デジタル I/O ターミナル(O)とロボット(C)間の配線
- カメラ(L)とロボット(X)間の配線
- EtherCAT デジタル I/O ターミナル(O)とソレノイドバルブ(P)間の配線

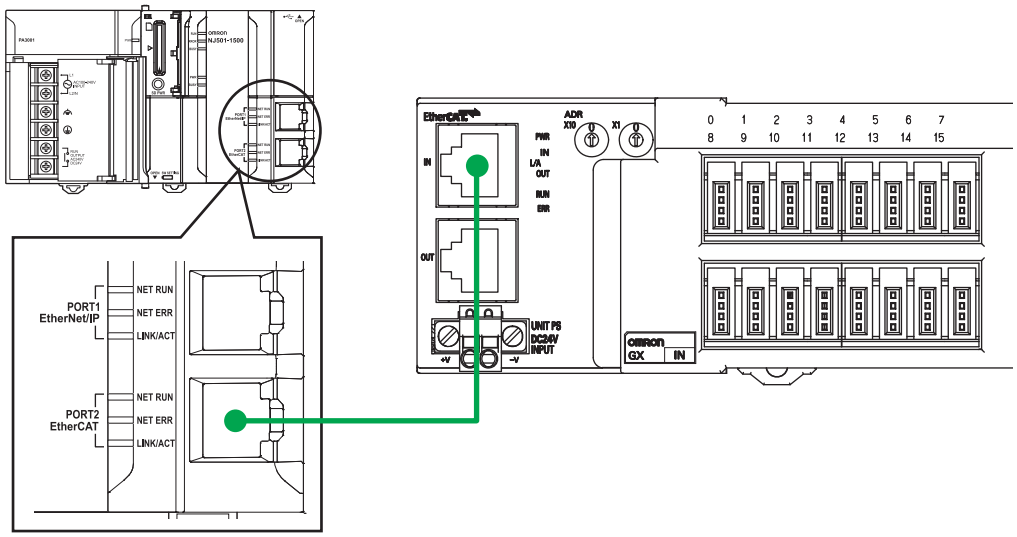


システム構成で使用する部品については、「1-5 動的ピック & プレース装置のシステム構成 (P.1-6)」を参照してください。
図中の枠で囲われていない電源線、およびロボットからの配管については、各製品のマニュアルを参照して配線、配管してください。

4-4-1 ロボット統合 CPU ユニットと EtherCAT デジタル I/O ターミナル間を配線する

ロボット統合 CPU ユニットと EtherCAT デジタル I/O ターミナル間を配線します。

1 **ロボット統合 CPU ユニットの EtherCAT ポートと、EtherCAT デジタル I/O ターミナルにある EtherCAT の IN ポートを Ethernet ケーブルで接続します。**

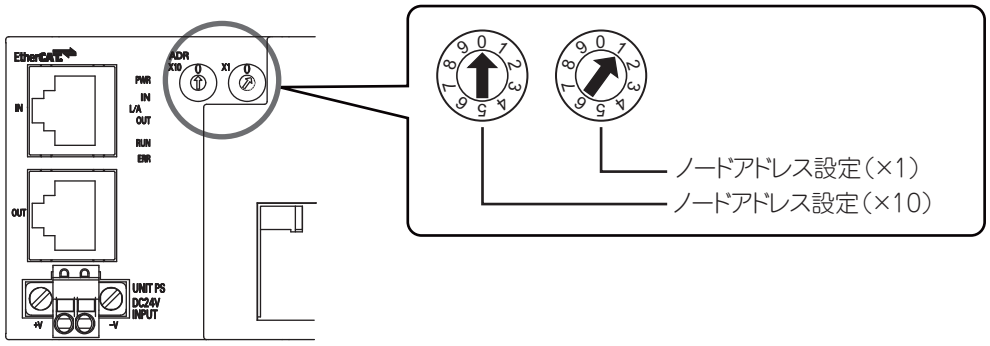


4-4-2 EtherCAT デジタル I/O ターミナルの EtherCAT ノードアドレスを設定する

EtherCAT デジタル I/O ターミナルの EtherCAT ノードアドレスを設定します。

1 **EtherCAT デジタル I/O ターミナルにあるスイッチで EtherCAT ノードアドレスを設定します。**

ノードアドレスを 1 に設定するため、×10 のスイッチを 0、×1 のスイッチを 1 に設定します。



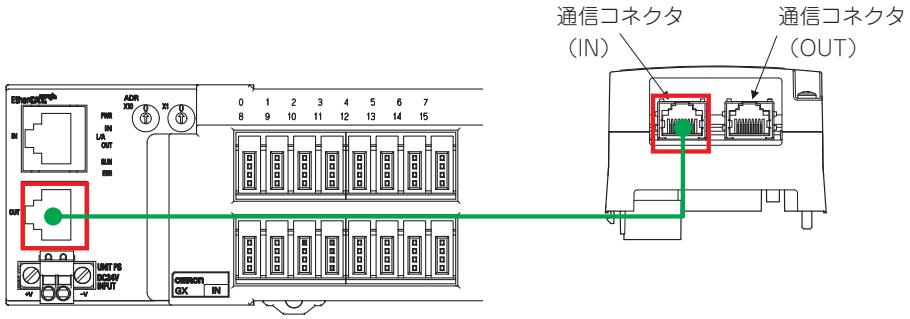
使用上の注意

デジタル I/O ターミナルの EtherCAT ノードアドレスは、「3-3-2 EtherCAT ネットワーク構成を作成する (P.3-17)」で設定したノードアドレスと一致するように設定します。

4-4-3 EtherCAT デジタル I/O ターミナルとインバータの EtherCAT 通信ユニット間を配線する

EtherCAT デジタル I/O ターミナルとインバータの EtherCAT 通信ユニット間を配線します。

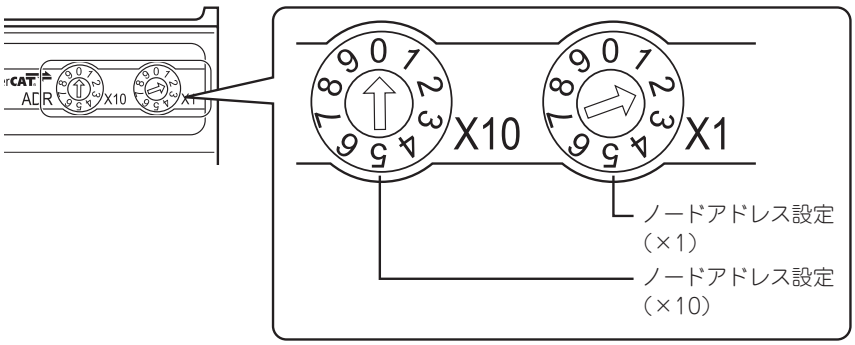
1 **EtherCAT デジタル I/O ターミナルにある EtherCAT の OUT ポートと、EtherCAT 通信ユニットにある EtherCAT の IN ポートを Ethernet ケーブルで接続します。**



4-4-4 インバータの EtherCAT 通信ユニットの EtherCAT ノードアドレスを設定する

インバータの EtherCAT 通信ユニットの EtherCAT ノードアドレスを設定します。

1 **インバータの EtherCAT 通信ユニットにあるスイッチで EtherCAT ノードアドレスを設定します。**
ノードアドレスを 2 に設定するため、×10 のスイッチを 0、×1 のスイッチを 2 に設定します。



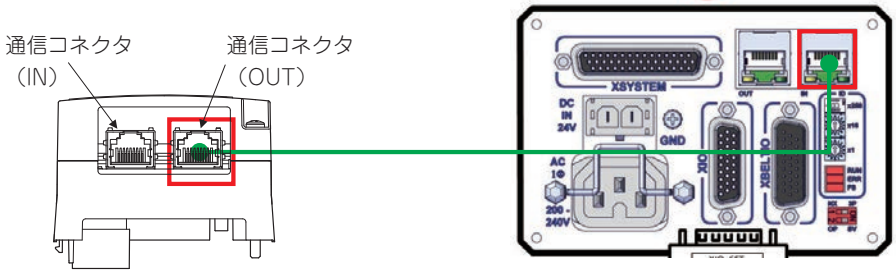
使用上の注意

インバータの EtherCAT ノードアドレスは、「3-3-2 EtherCAT ネットワーク構成を作成する (P.3-17)」で設定したノードアドレスと一致するように設定します。

4-4-5 インバータの EtherCAT 通信ユニットとロボット間を配線する

インバータの EtherCAT 通信ユニットとロボット間を配線します。

1 **インバータの EtherCAT 通信ユニットにある EtherCAT の OUT ポートと、ロボットのインタフェースパネルにある EtherCAT の IN ポートを Ethernet ケーブルで接続します。**



使用上の注意

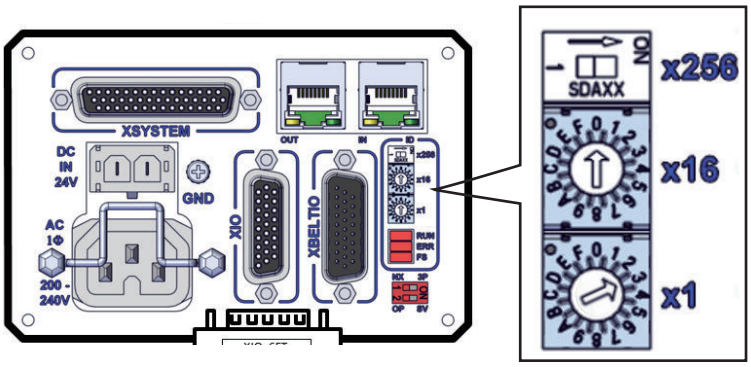
使用する Ethernet ケーブルの詳細については、『NJ/NX シリーズ CPU ユニット内蔵 EtherCAT ポート ユーザーズマニュアル (SBCD-376)』を参照してください。

4-4-6 ロボットの EtherCAT ノードアドレスを設定する

ロボットの EtherCAT ノードアドレスを設定します。

1 ロボットのインタフェースパネルにあるスイッチでロボットの EtherCAT ノードアドレスを設定します。

ノードアドレスを 3 に設定するため、×256 のスイッチを OFF、×16 のスイッチを 0、×1 のスイッチを 3 に設定します。



使用上の注意

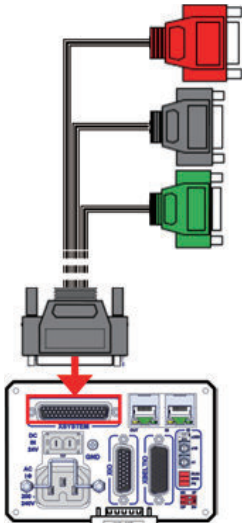
ロボットの EtherCAT ノードアドレスは、「4-3-2 EtherCAT ネットワーク構成を作成する (P.4-16)」で設定したノードアドレスと一致するように設定します。

ノードアドレスの設定方法については、ロボットのマニュアルを参照してください。

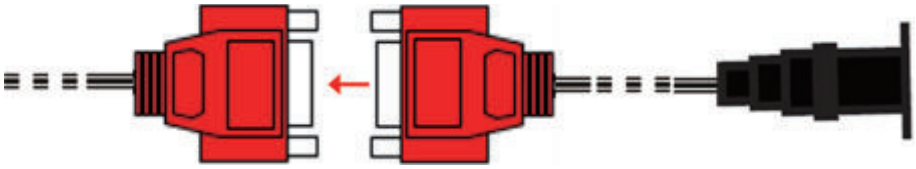
4-4-7 ロボットと T20 ペンダント、フロントパネル間を配線する

ロボットと T20 ペンダント間、およびロボットとフロントパネル間を配線します。配線には、XSYSTEM ケーブルアセンブリ、T20 アダプタケーブル、フロントパネルケーブルを使用します。

1 ロボットと XSYSTEM ケーブルアセンブリを接続します。



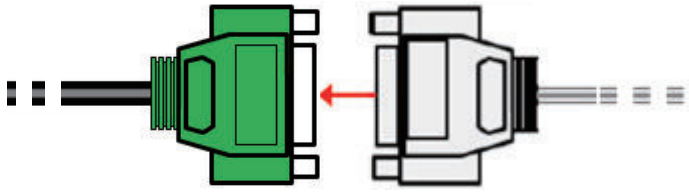
2 XSYSTEM ケーブルアセンブリの XMCP コネクタ（赤）と T20 アダプタケーブルを接続します。



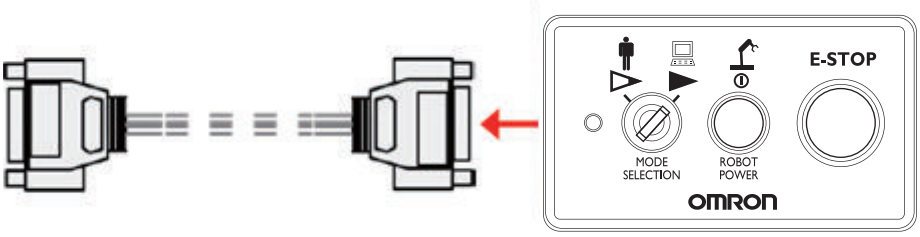
3 T20 アダプタケーブルと T20 ペンダントを接続します。



4 XSYSTEM ケーブルアセンブリの XFP コネクタ（緑）とフロントパネルケーブルを接続します。



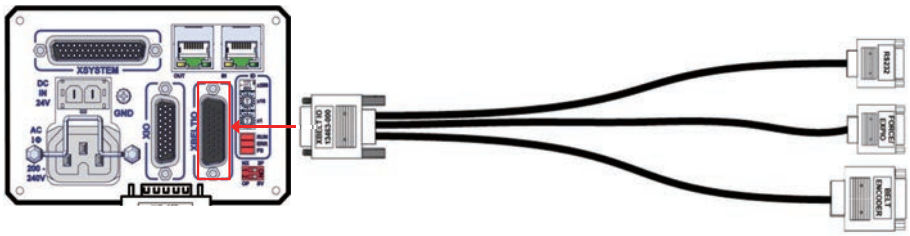
5 フロントパネルケーブルとフロントパネルを接続します。



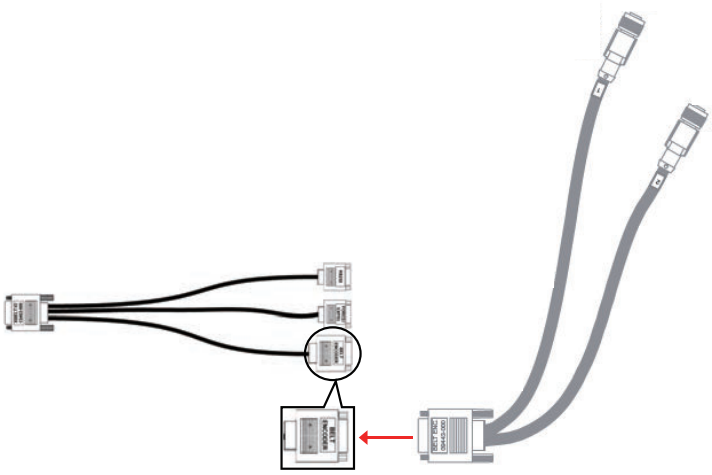
4-4-8 ロボットとエンコーダ間を配線する

ロボットとエンコーダ間を配線します。配線には、XBELTIO ケーブル、Y-アダプタケーブル、エンコーダ延長ケーブルを使用します。

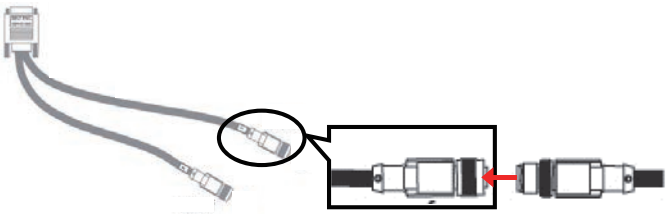
1 **ロボットと XBELTIO ケーブルを接続します。**



2 **XBELTIO ケーブルの DB15m コネクタと Y-アダプタケーブルを接続します。**



3 **Y-アダプタケーブルの[1]ラベルが貼られたコネクタとエンコーダ延長ケーブルを接続します。**



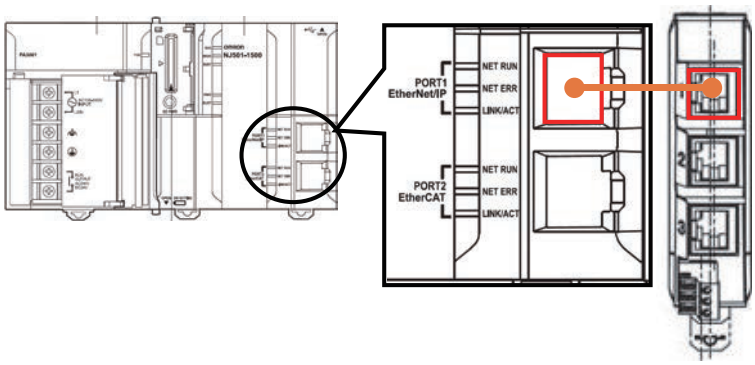
4 **エンコーダ延長ケーブルとエンコーダを接続します。**



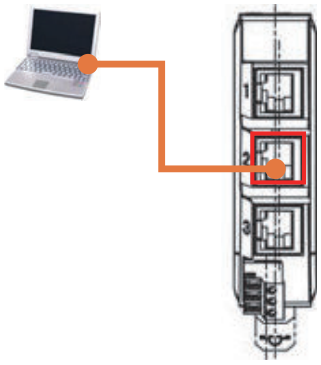
4-4-9 ロボット統合 CPU ユニットの EtherNet/IP ポートを配線する

ロボット統合 CPU ユニットと産業用スイッチングハブ間、産業用スイッチングハブとパソコン間、および産業用スイッチングハブと IPC Application Controller 間を配線します。配線には Ethernet ケーブルを使用します。

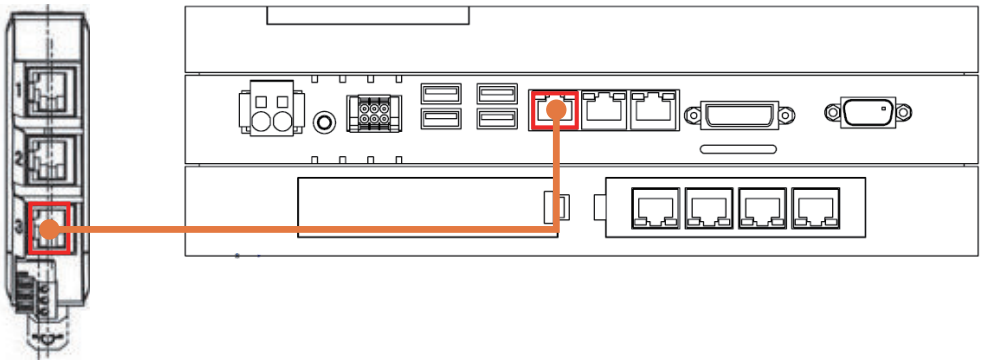
1 **ロボット統合 CPU ユニットと産業用スイッチングハブを接続します。**



2 **産業用スイッチングハブとパソコンを接続します。**



3 **産業用スイッチングハブと IPC Application Controller を接続します。**



4-4-10 IPC Application Controller とディスプレイ、カメラ間を配線する

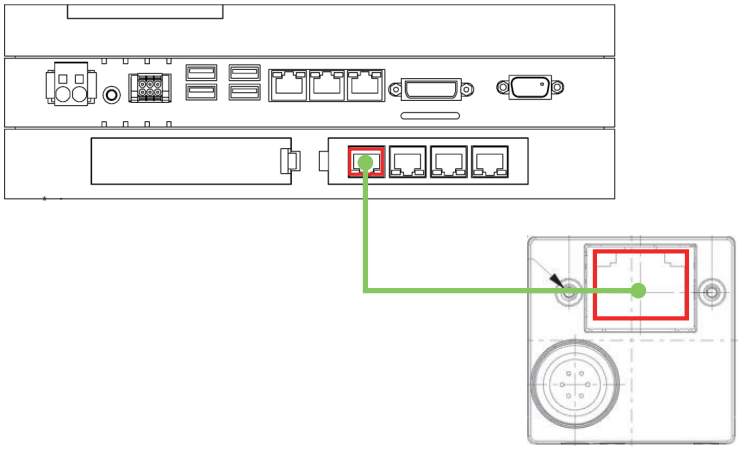
IPC Application Controller とディスプレイ間、および IPC Application Controller とカメラ間を配線します。配線には、DVI ケーブル、およびカメラケーブルを使用します。

1 **IPC Application Controller の DVI ポートと、ディスプレイの DVI ポートを DVI ケーブルで接続します。**

使用上の注意

使用する DVI ケーブルの詳細については、『IPC アプリケーションコントローラ ユーザーズマニュアル(SBCE-454)』を参照してください。

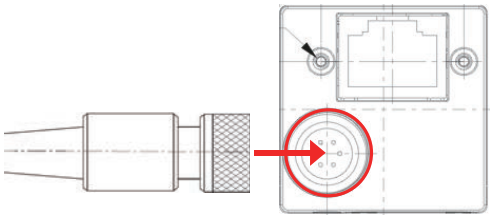
2 **IPC Application Controller の PCIe PoE ポートとカメラの PoE ポートをカメラケーブルで接続します。**



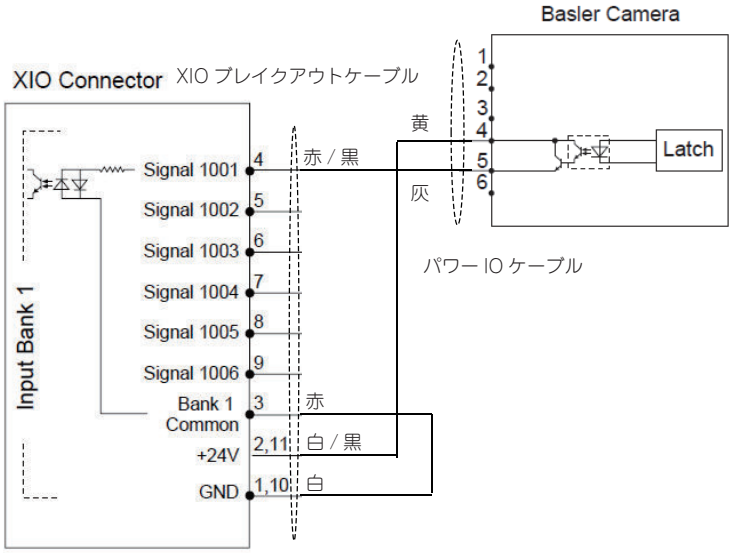
4-4-11 カメラとロボット間を配線する

カメラとロボット間を配線します。配線には、パワー I/O ケーブル、XIO ブレイクアウトケーブルを使用します。

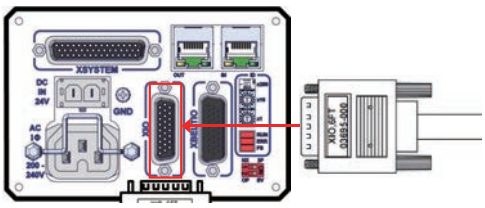
1 **カメラとパワー I/O ケーブルを接続します。**



2 **パワー I/O ケーブルと XIO ブレイクアウトケーブルを接続します。次の配線図のとおり、コネクタを作成するか端子台で中継してください。**



3 **XIO ブレイクアウトケーブルとロボットを接続します。**



使用上の注意

カメラの設置位置については、「カメラ、コンベア、エンコーダの設定をする (P.4-64)」を参照してください。ソフトウェアで設定するコンベアの上流側リミットからコンベアの入口までの間に設置する必要があります。

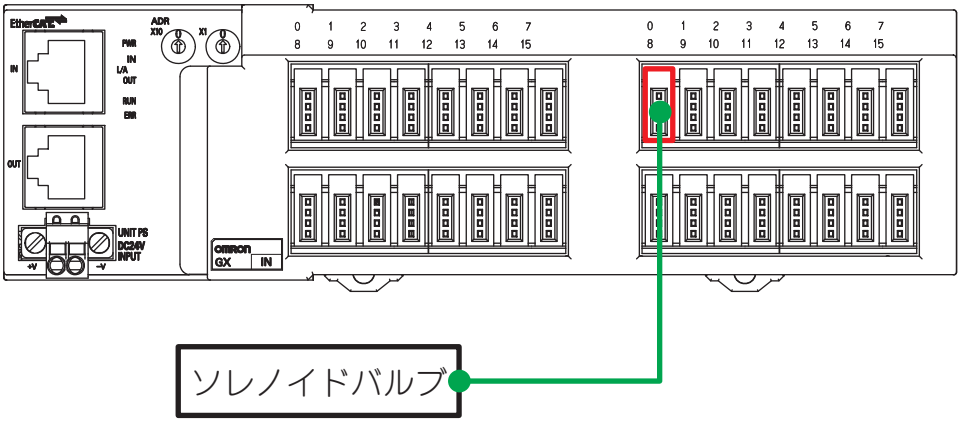
4-4-12 EtherCAT デジタル I/O ターミナルとソレノイドバルブ間を配線する

EtherCAT デジタル I/O ターミナルとソレノイドバルブ間を配線します。
配線するケーブルについては、EtherCAT デジタル I/O ターミナルおよびソレノイドバルブのマニュアルを参照し、配線図や仕様に応じて適切に制作してください。

1 **ソレノイドバルブが以下の挙動となるよう、ケーブルを制作します。**

デジタル I/O ターミナルの出力値	ソレノイドバルブの振る舞い
FALSE	配管が閉じる
TRUE	配管が開く

2 **EtherCAT デジタル I/O ターミナルとソレノイドバルブ間を配線します。**


EtherCAT
ADR
X10
X1
PWR
IN
L/A
OUT
RUN
ERR
IN
OUT
UNIT PS
DC24V
INPUT
OMRON
GX
IN
0 1 2 3 4 5 6 7
8 9 10 11 12 13 14 15
0 1 2 3 4 5 6 7
8 9 10 11 12 13 14 15
ソレノイドバルブ

4-5 実機での動作確認

実機での動作確認、調整の手順を説明します。
実機での動作確認の前に、シミュレーションで動作を確認しておいてください。

使用上の注意

T20 ペンダントによりロボットをマニュアルモードで操作する場合、iCS 試運転ジャンパを XBELTIO コネクタに接続し、ティーチング制限に関する設定を変更する必要があります。詳細は、『パラレルロボット iX4 650H/650HS/800H/800HS EtherCAT タイプ ユーザーズマニュアル(SBCE-492)』を参照してください。

4-5-1 通信設定をする

Sysmac Studio とロボット統合 CPU ユニットを接続するための設定をする

パソコン上の Sysmac Studio とロボット統合 CPU ユニットを接続するための設定をします。
本説明では、パソコンを接続するロボット統合 CPU ユニットの EtherNet/IP ポートのアドレスを 192.168.250.1 としています。
また、事前に IPC Application Controller にカメラの IP アドレスを設定してください。IPC Application Controller にカメラの IP アドレスを設定する方法については、『オートメーション コントロール エンバイロメント (ACE) Version 4 ユーザーズマニュアル(SBCE-456)』の付録を参照してください。

1 **Sysmac Studio を起動します。**

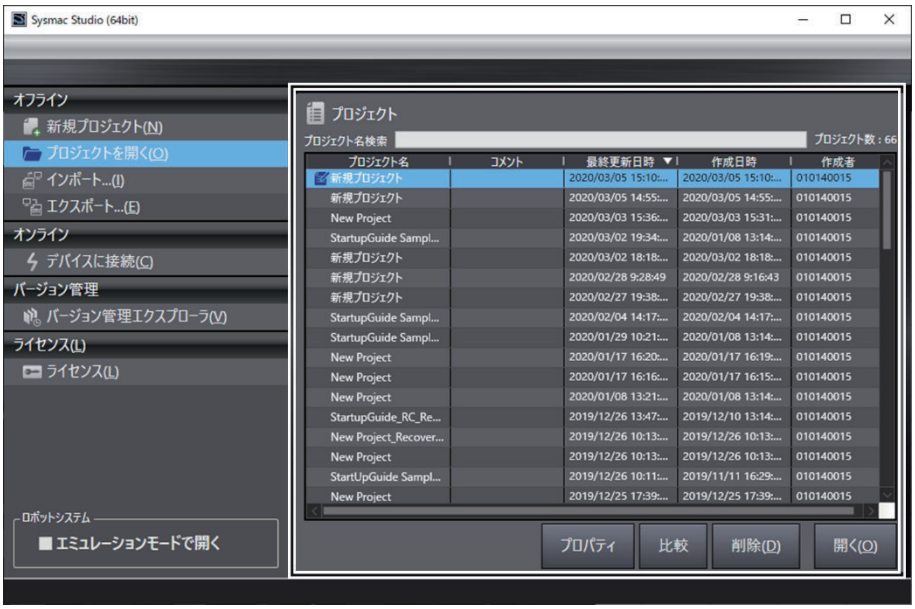
2 **スタートページ左下の［エミュレーションモードで開く］チェックボックスのチェックを外します。**



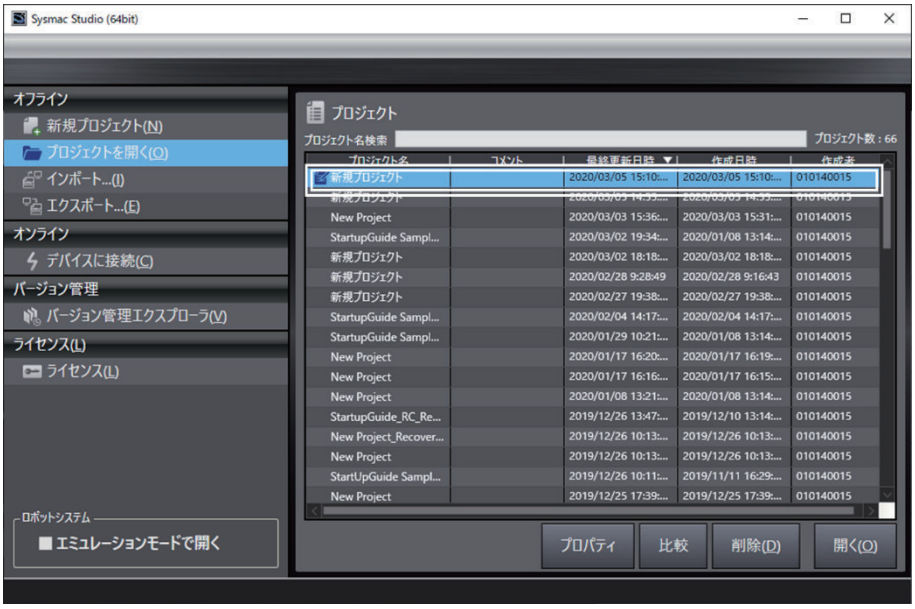
3 **［プロジェクトを開く］タブをクリックします。**



［プロジェクト］画面にこれまでに作成したプロジェクトの一覧が表示されます。

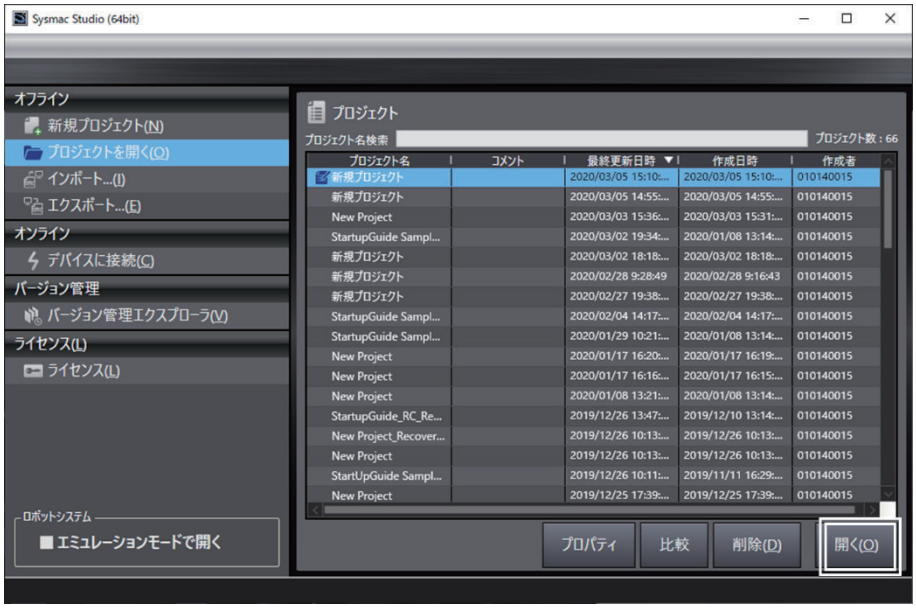


4 **プロジェクトの一覧から「4-3-1 プロジェクトファイルを新規作成する (P.4-15)」で作成した［新規プロジェクト］をクリックします。**

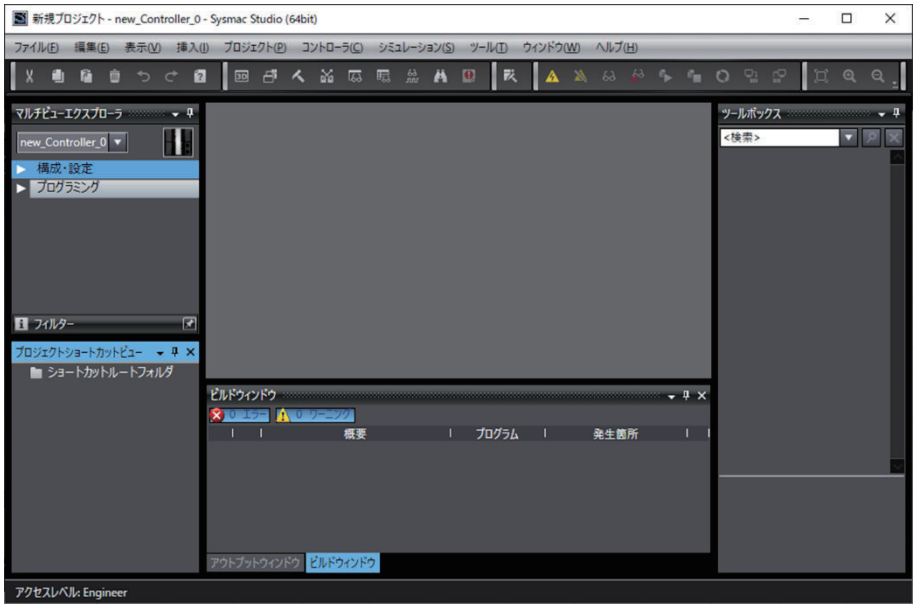


クリックした行がハイライトされます。

5 **［プロジェクト］画面の［開く］ボタンをクリックします。**



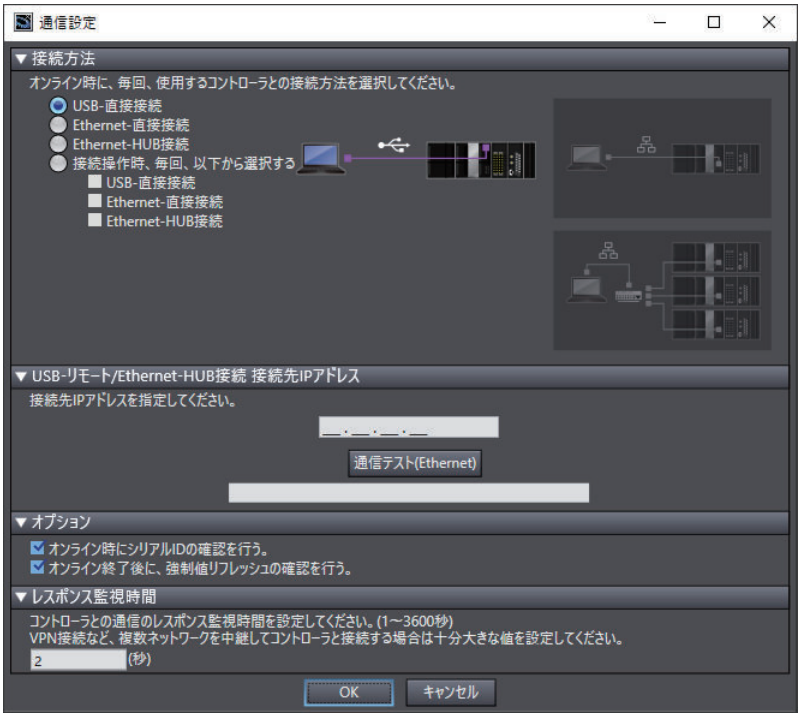
Sysmac Studio のメインウィンドウが表示されます。



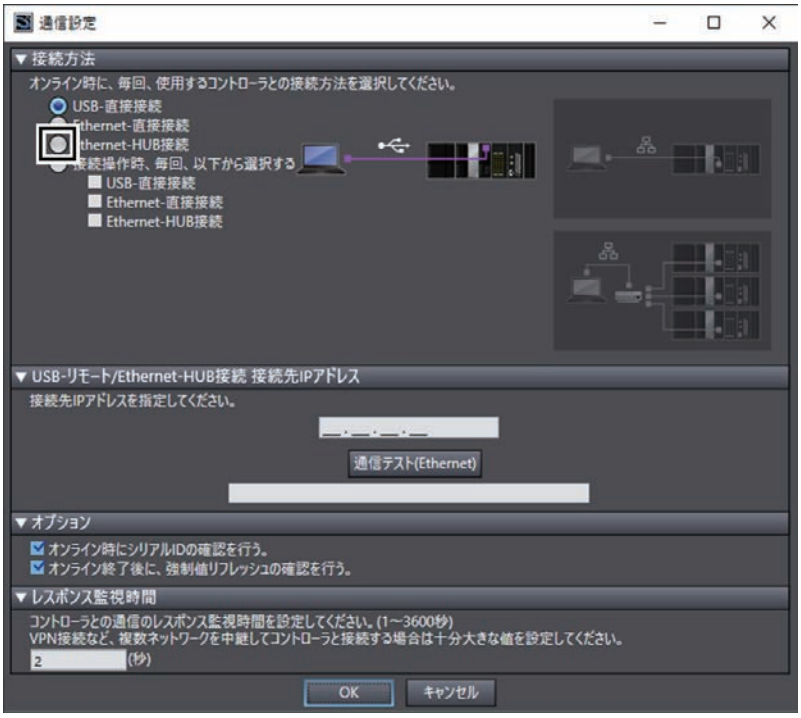
6 **メインウィンドウのメニューバーから［コントローラ］－［通信設定］をクリックします。**



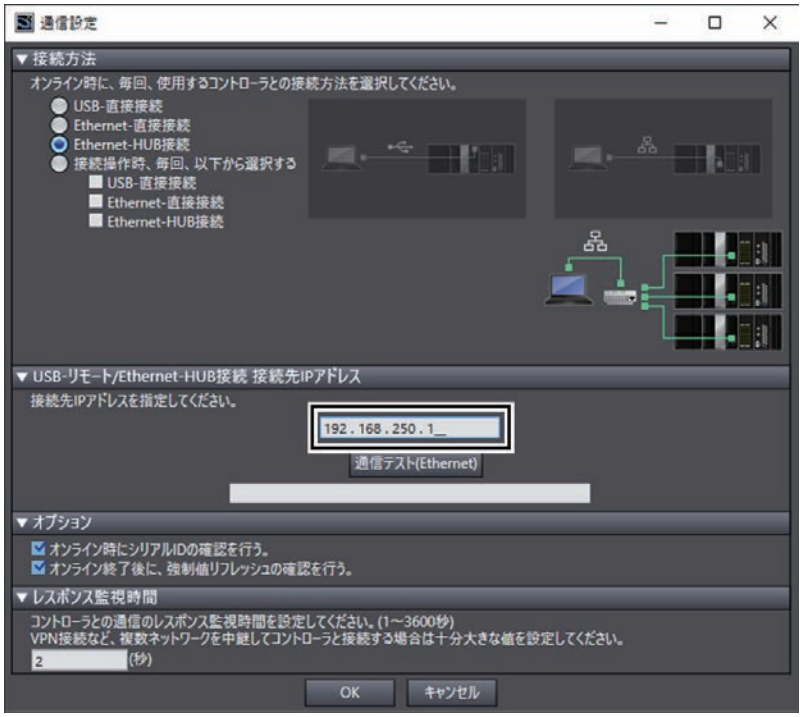
［通信設定］画面が表示されます。



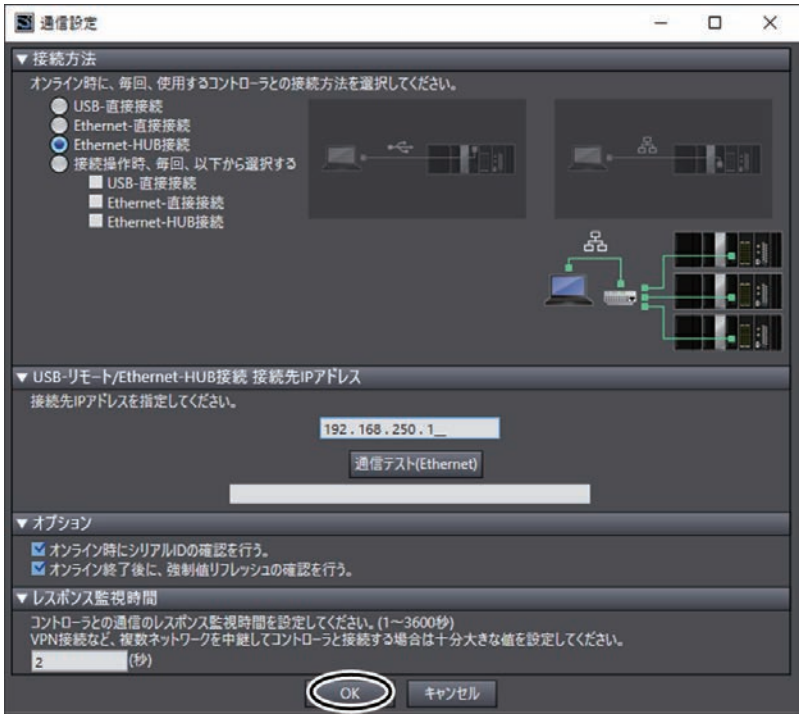
7 **接続方法で［Ethernet-HUB 接続］ラジオボタンをクリックします。**



8 **画面中央の［USB-リモート/Ethernet-HUB 接続 接続先 IP アドレス］にロボット統合 CPU ユニットの IP アドレス「192.168.250.1」を入力します。**



9 **画面下の［OK］ボタンをクリックします。**



［通信設定］画面が閉じます。

IPC Application Controller を選択する

使用する IPC Application Controller の IP アドレスを選択します。
本説明では IPC Application Controller の IP アドレスを 192.168.250.201 としています。事前に IPC Application Controller の IP アドレスを設定してから以下の手順を実行してください。

使用上の注意

IPC Application Controller で、ACE がサーバとしてあらかじめ起動している必要があります。ACE サーバインスタンスについては、『オートメーション コントロール エンバイロメント (ACE) Version 4 オペレーションマニュアル(SBCE-456)』を参照してください。

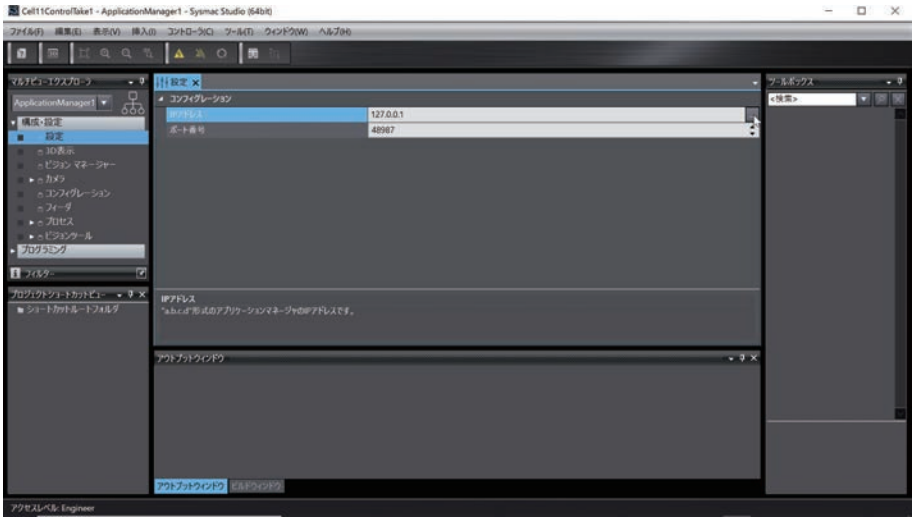
1 **マルチビューエクスプローラのデバイスで、［ApplicationManager1］を選択します。**



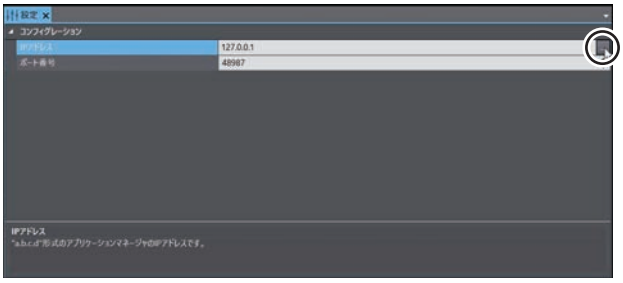
2 **マルチビューエクスプローラの［構成・設定］–［設定］をダブルクリックします。**



［設定］画面が表示されます。

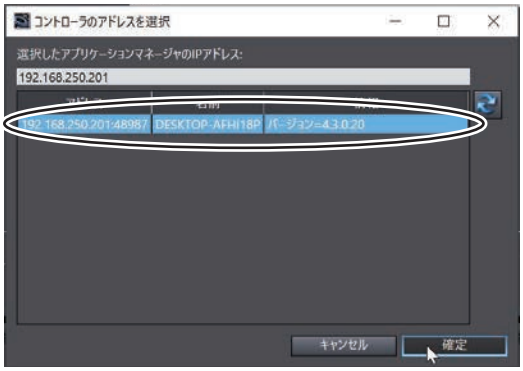


3 **［IP アドレス］の行の右側にある［...］をクリックします。**

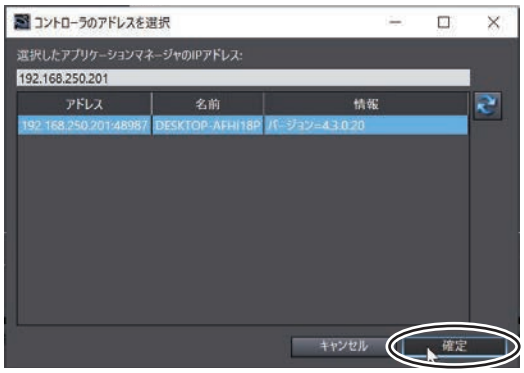


［コントローラのアドレスを選択］画面が表示されます。

4 **使用する IPC Application Controller の IP アドレスが表示されている行をクリックします。**



5 **［確定］ボタンをクリックします。**



［コントローラのアドレスを選択］画面が閉じます。

4-5-2 オンライン接続する

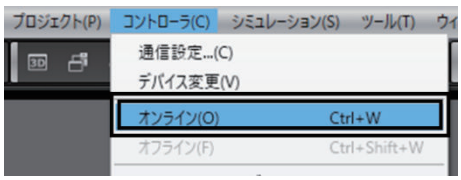
ロボット統合 CPU ユニットとオンライン接続する

Sysmac Studio とロボット統合 CPU ユニットをオンライン接続します。

1 **マルチビューエクスプローラのデバイスで、［new_Controller_0］を選択します。**



2 **メニューバーから［コントローラ］-［オンライン］をクリックします。**



参考

接続先のロボット統合 CPU ユニットの状態によっては、CPU ユニット名の書き込みやシリアル番号の確認などのダイアログが表示されることがあります。詳細は『Sysmac Studio Version 1 オペレーションマニュアル (SBCA-470)』を参照してください。

オンライン状態になるとエディットウィンドウの上部に黄線が表示されます。



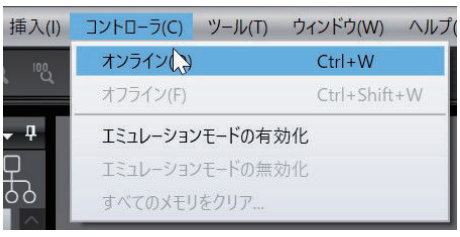
IPC Application Controller とオンライン接続する

Sysmac Studio と IPC Application Controller をオンライン接続します。

1 **マルチビューエクスプローラのデバイスで、[ApplicationManager1] を選択します。**



2 **メニューバーから [コントローラ] - [オンライン] をクリックします。**



オンライン状態になるとエディットウィンドウの上部に黄線が表示されます。



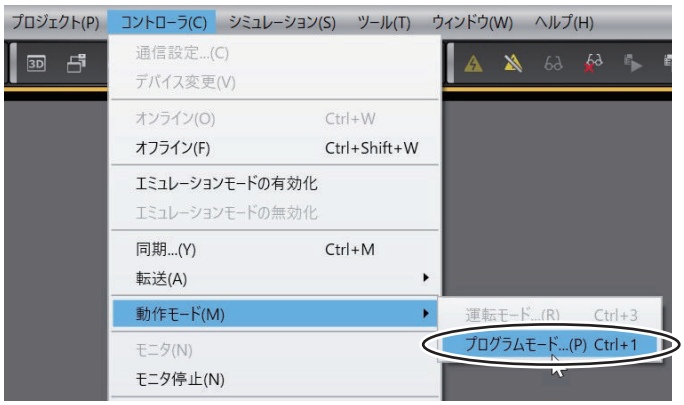
4-5-3 設定とプログラムを転送する

コンベアをプログラムで動作できるように、Sysmac Studio のプロジェクトファイル内の設定およびプログラムをロボット統合 CPU ユニットに転送します。

1 **マルチビューエクスプローラのデバイスで、[new_Controller_0] を選択します。**



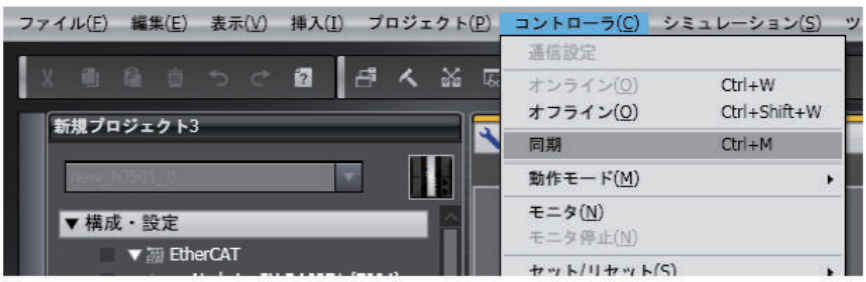
2 **メニューバーから [コントローラ] – [動作モード] – [プログラムモード] をクリックします。**



3 **確認画面の説明に従った対応を行い、[はい] ボタンをクリックします。**

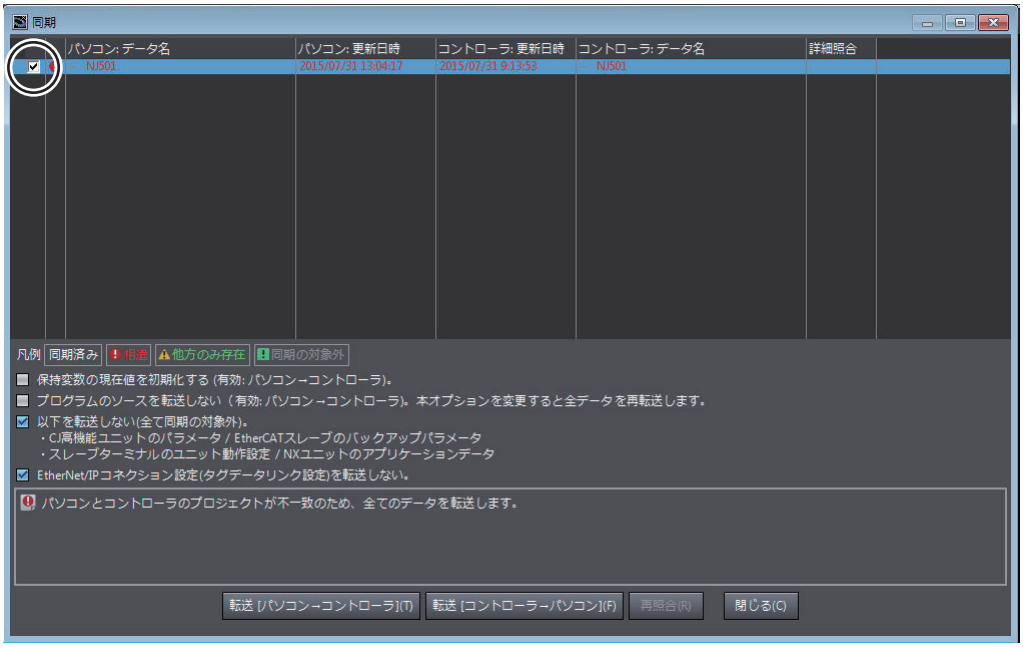


4 **Sysmac Studio のメニューバーから [コントローラ] – [同期] を選択します。**

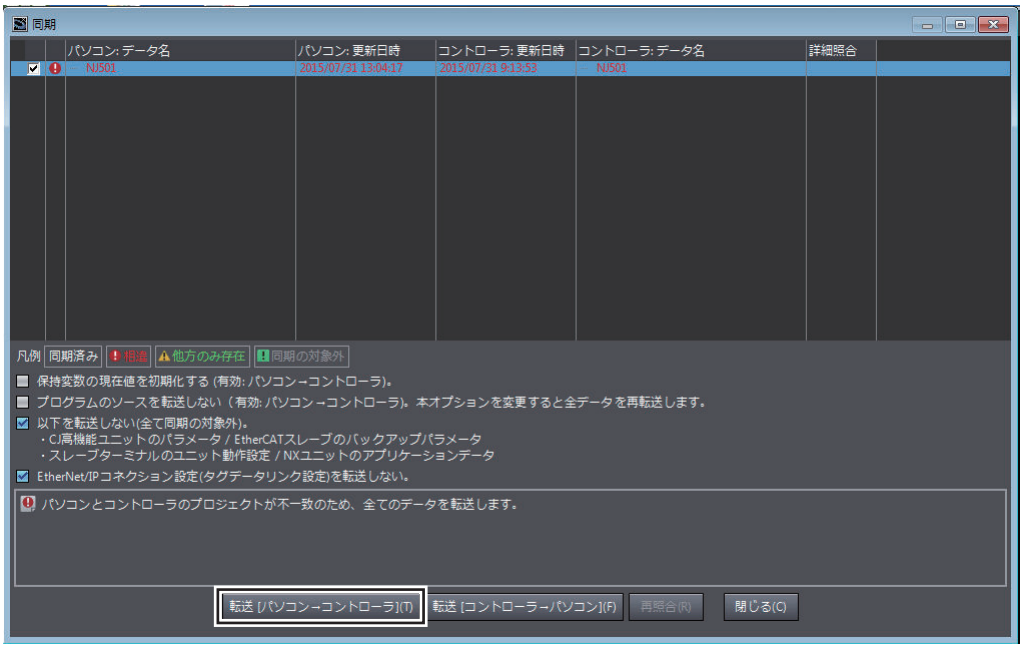


Sysmac Studio とコントローラ内のプログラムおよび設定データの照合が開始されます。照合が完了すると、同期ウィンドウが表示されます。

5 **[NJ501] にチェックを入れます。**



6 **[転送（パソコン→コントローラ)] ボタンをクリックします。**



7 **確認ダイアログが数回表示されますので、[はい] ボタンをクリックします。**

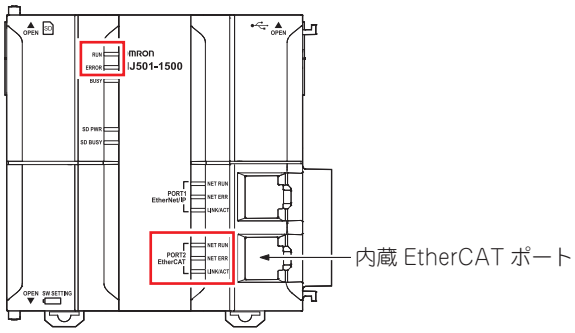
8 **“同期は成功しました。”と表示されたことを確認し、[閉じる] ボタンをクリックします。**



9 **ロボット統合 CPU ユニットが正常に起動したことを確認します。**

正常時の LED 状態は次のとおりです。

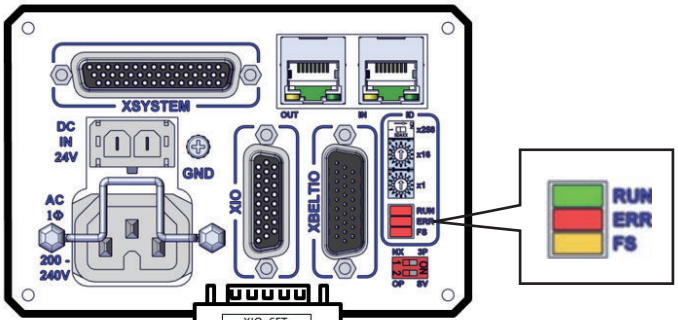
LED	状態
RUN LED	緑点灯
ERROR LED	消灯
NET RUN	緑点灯
NET ERR	消灯
LINK/ACT	点滅



10 **ロボットが正常に起動したことを確認します。**

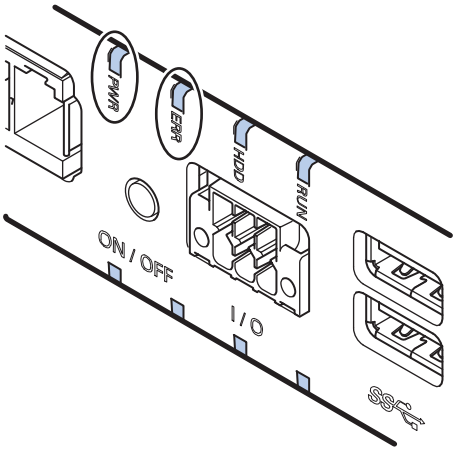
正常時の LED 状態は次のとおりです。

LED	状態
RUN LED	緑点灯
ERR LED	消灯



11 **IPC Application Controller が正常に起動したことを確認します。**
正常時の LED 状態は次のとおりです。

LED	状態
PWR LED	緑点灯
ERR LED	消灯



4-5-4 カメラの設定をする

使用するカメラを追加する

1 **マルチビューエクスプローラで［ApplicationManager1］を選択します。**

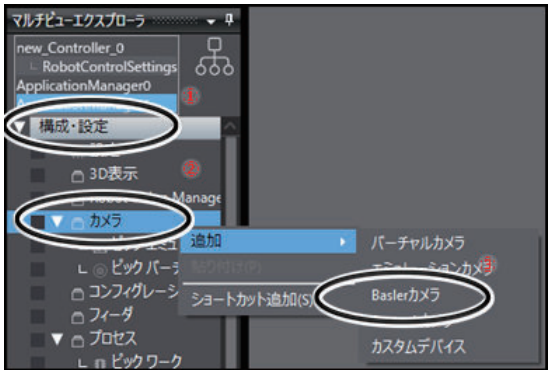


2 **カメラの種類を選択します。**

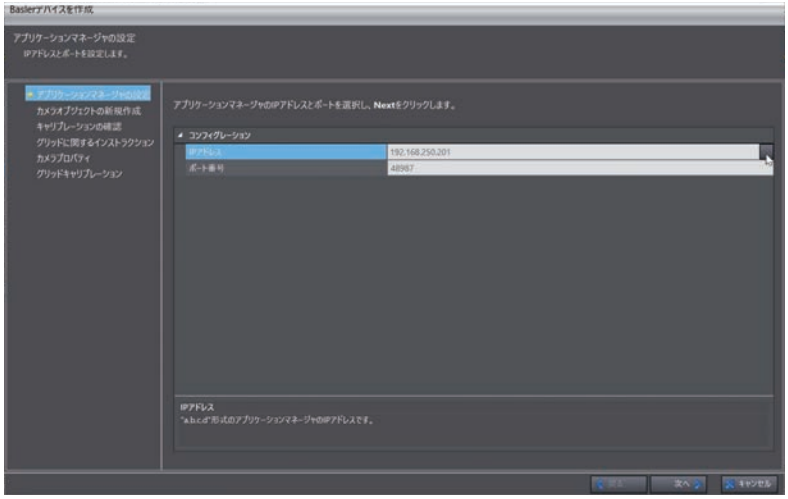
1) ［構成・設定］をクリックして展開します。

2) ［カメラ］を右クリックしてコンテキストメニューを表示します。

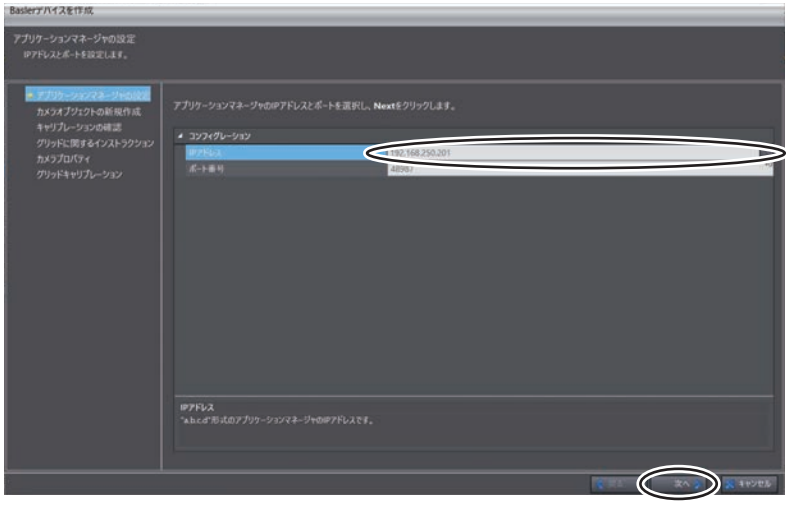
3) ［追加］–［Basler カメラ］を選択します。



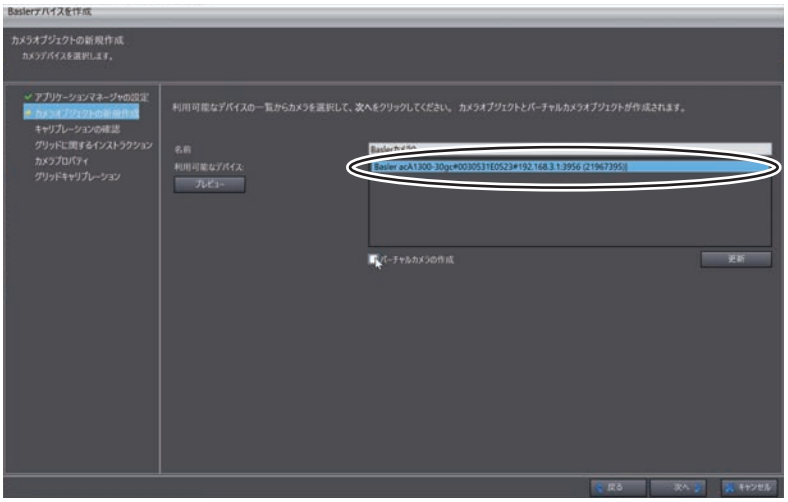
［Basler デバイスを作成］画面が表示されます。



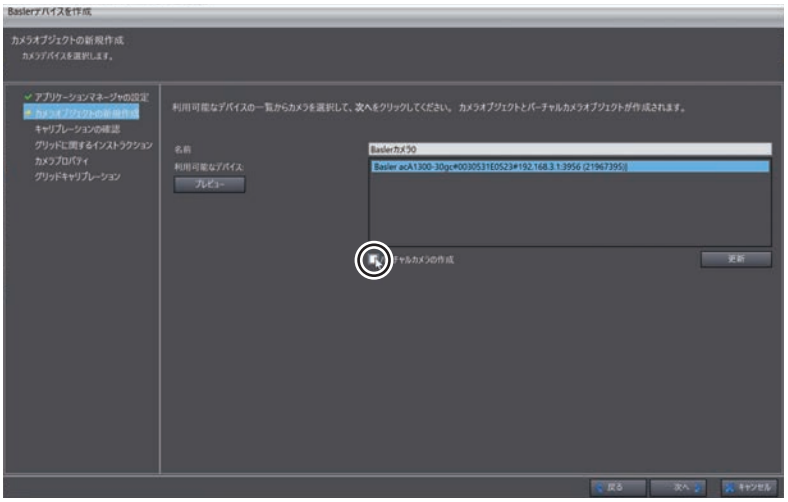
3 **［IP アドレス］に使用する IPC Application Controller の IP アドレスが設定されていることを確認して［次へ］ボタンをクリックします。**



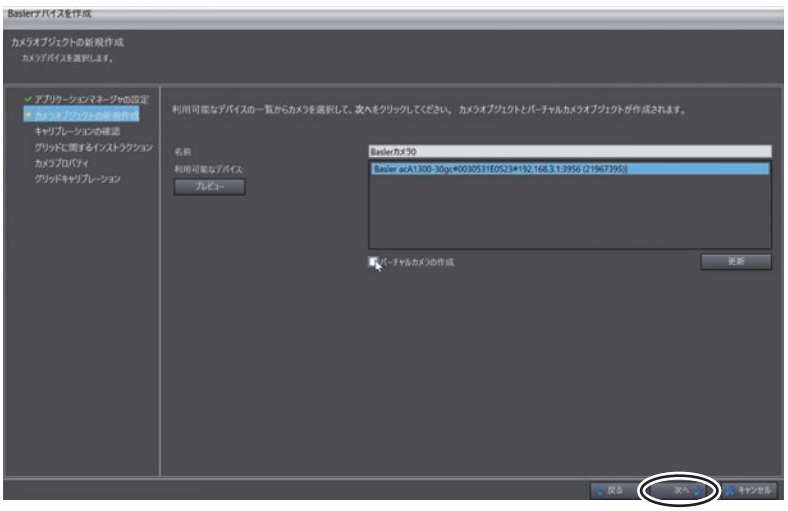
4 **使用しているカメラをデバイスの一覧から選択します。**



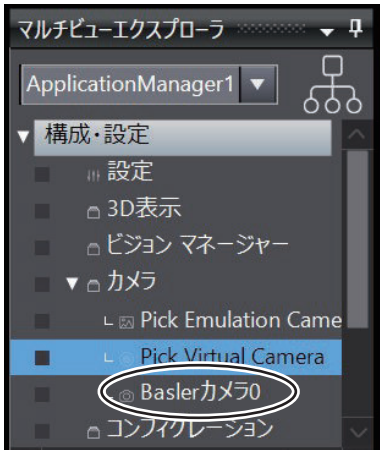
5 **［バーチャルカメラの作成］のチェックを外します。**



6 **［次へ］ボタンをクリックします。**

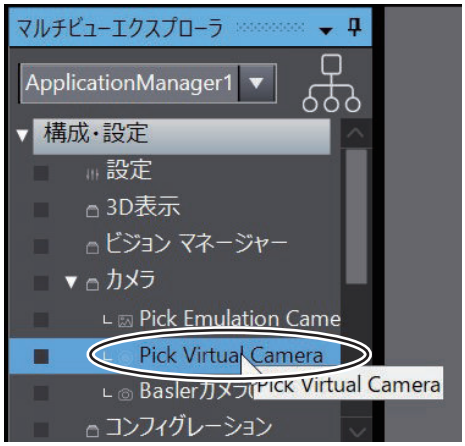


［Basler デバイスを作成］画面が閉じ、マルチビューエクスプローラの［構成・設定］–［カメラ］の下に［Basler カメラ 0］が追加されます。



バーチャルカメラのデフォルトデバイスを設定する

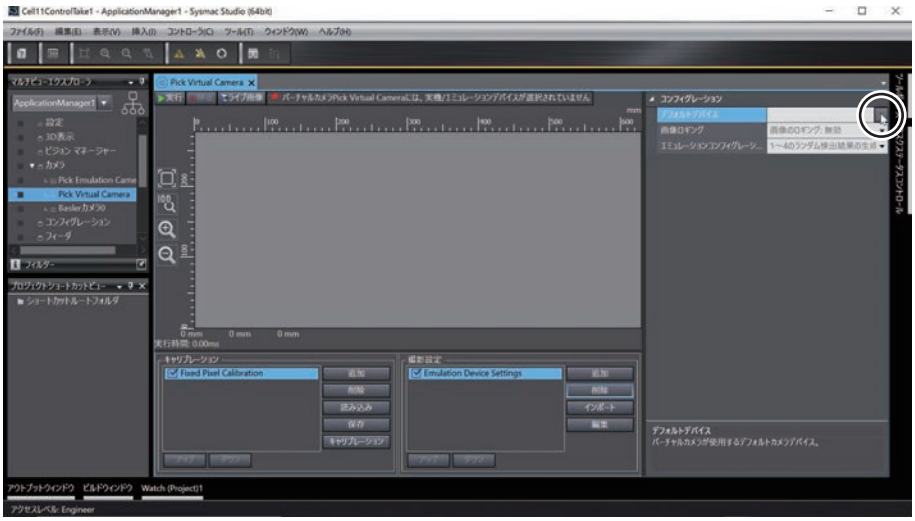
1 **マルチビューエクスプローラの［構成・設定］–［カメラ］–［Pick Virtual Camera］をダブルクリックします。**



［Pick Virtual Camera］画面が表示されます。



2 **［コンフィグレーション］–［デフォルトデバイス］の右側の［...］をクリックします。**



［基準を選択］画面が表示されます。



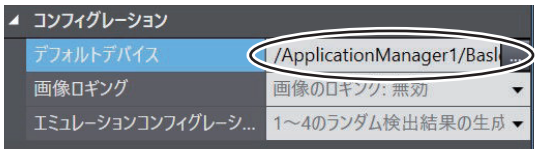
3 **［Basler カメラ 0］をクリックします。**



4 **［選択］ボタンをクリックします。**

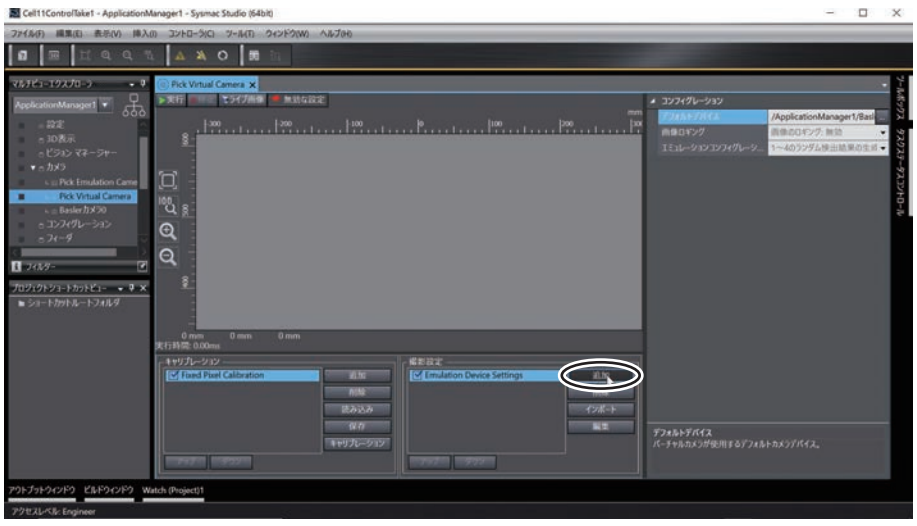


［基準を選択］画面が閉じ、［コンフィグレーション］-［デフォルトデバイス］に［Basler カメラ 0］が表示されます。



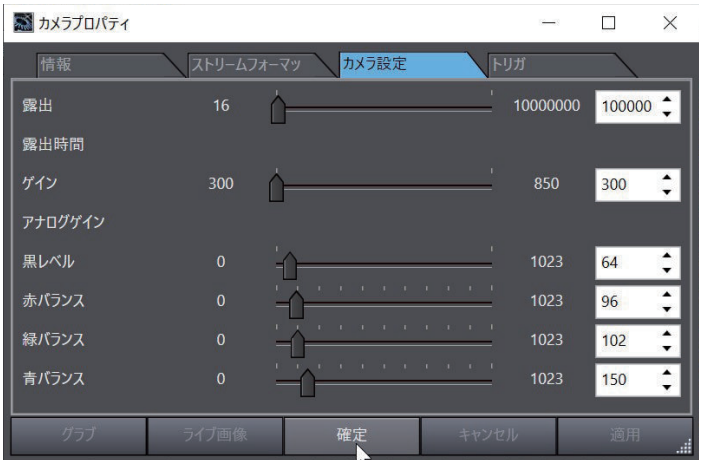
撮影設定を追加する

1 **［撮影設定］の［追加］ボタンをクリックします。**

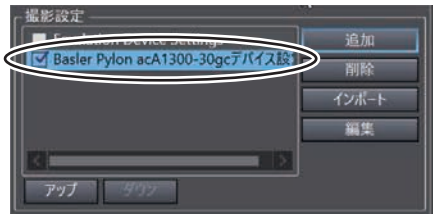


［カメラプロパティ］画面が表示されます

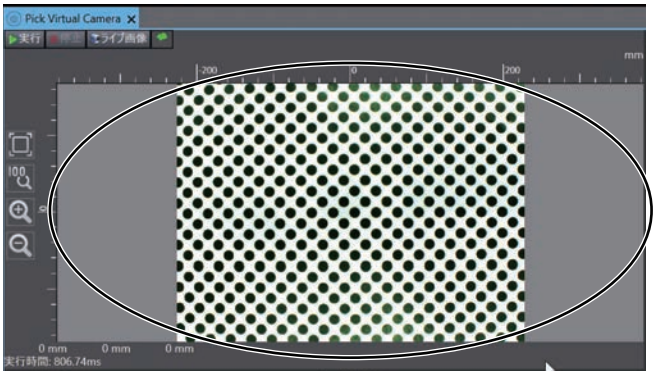
2 **［確定］ボタンをクリックします。**



［Basler Pylon acA1300-30gc デバイス設定］が追加されます。

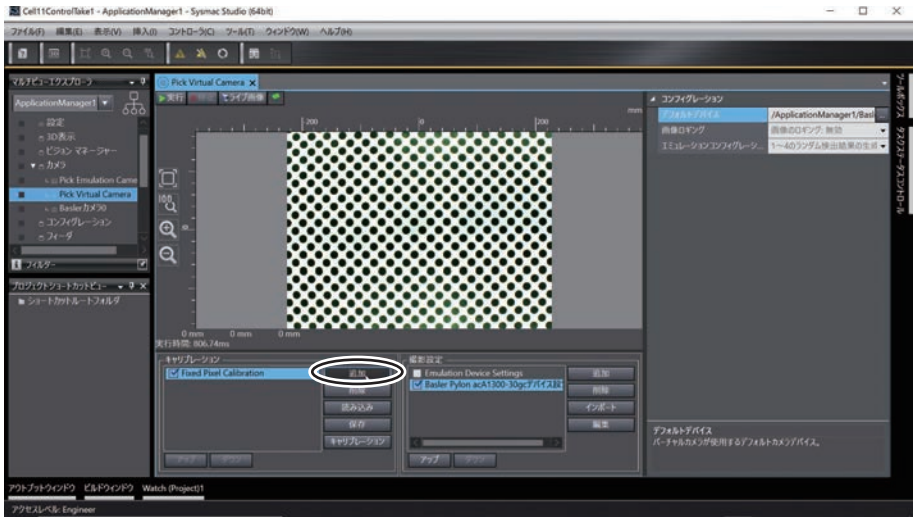


撮影画像が表示されるようになります。
たとえば、キャリブレーショングリッドを撮影範囲に置いてある場合には以下のように表示されます。

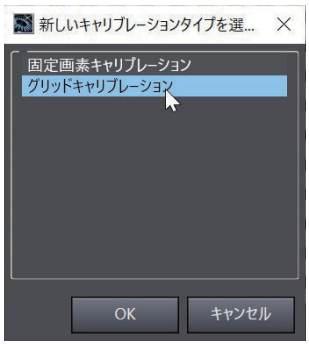


グリッドキャリブレーションを追加する

1 **［キャリブレーション］の［追加］ボタンをクリックします。**



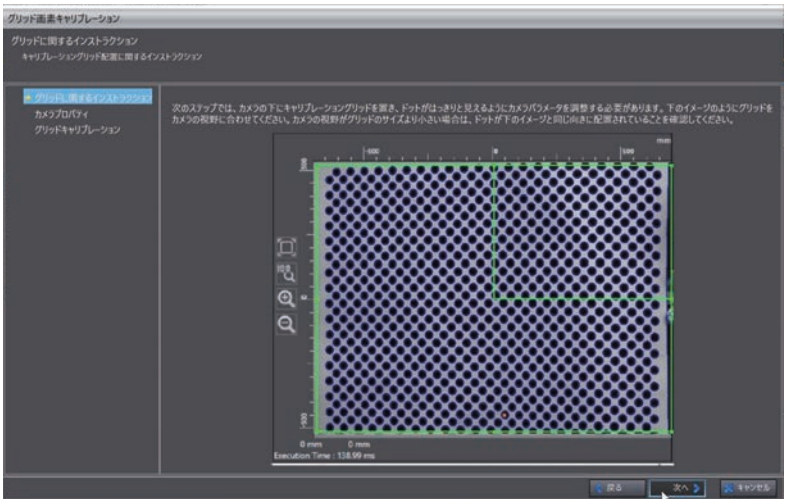
［新しいキャリブレーションタイプを選択］画面が表示されます。



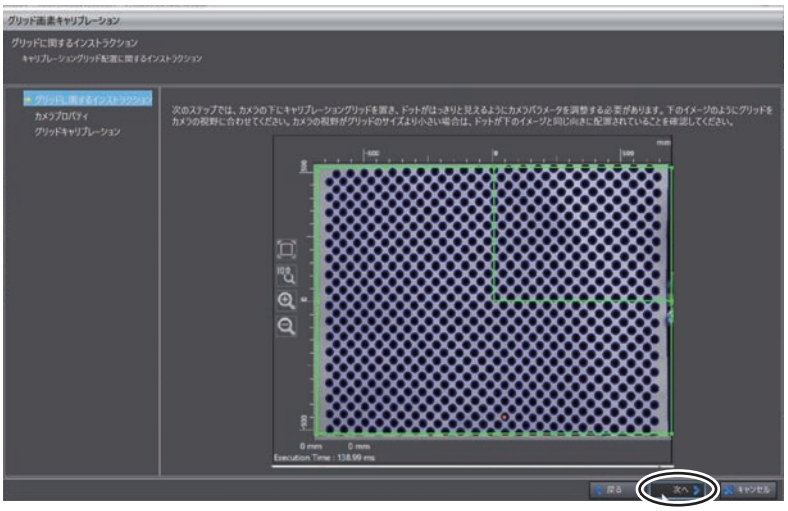
2 **［グリッドキャリブレーション］を選択し、［OK］ボタンをクリックします。**



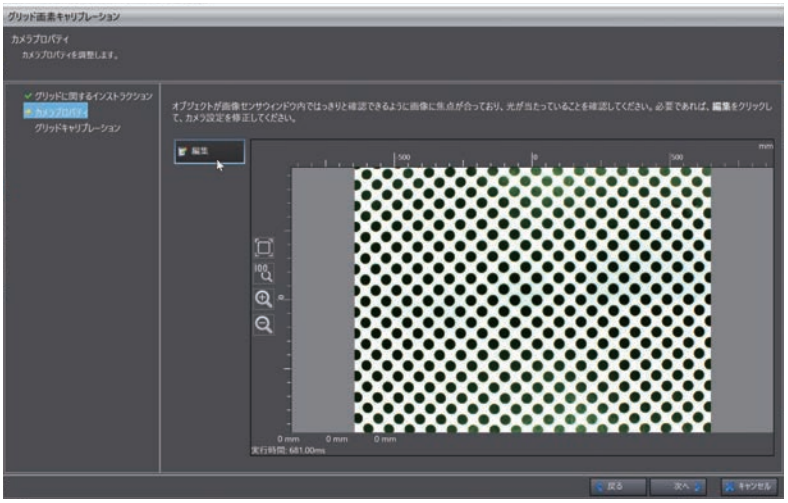
［グリッド画素キャリブレーション］画面が表示されます。



3 **[次へ] ボタンをクリックします。**

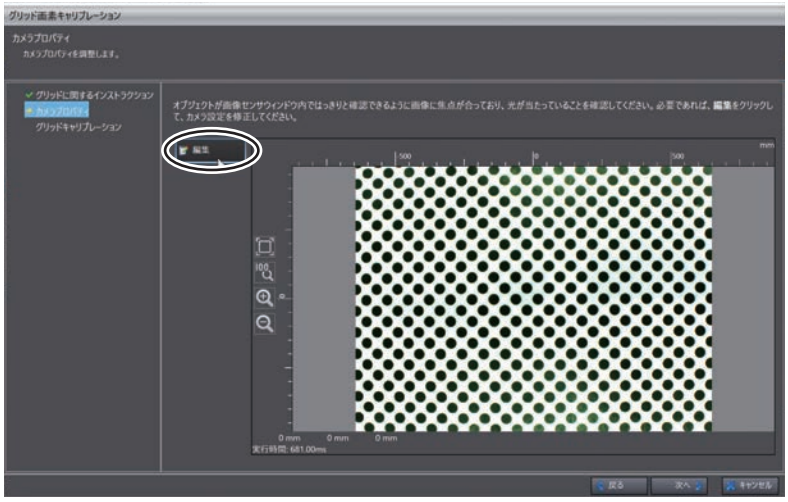


カメラで撮影されたキャリブレーショングリッドが表示されます。

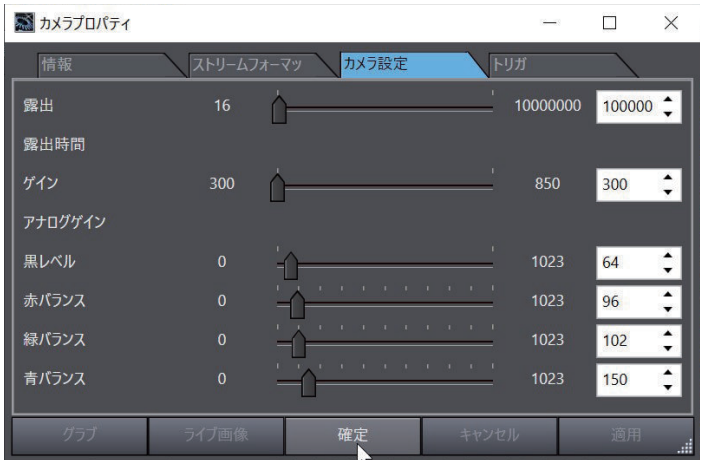


4 **キャリブレーショングリッドの位置、焦点、光が当たっていることを確認します。**
必要がある場合は、それぞれの調整を行います。

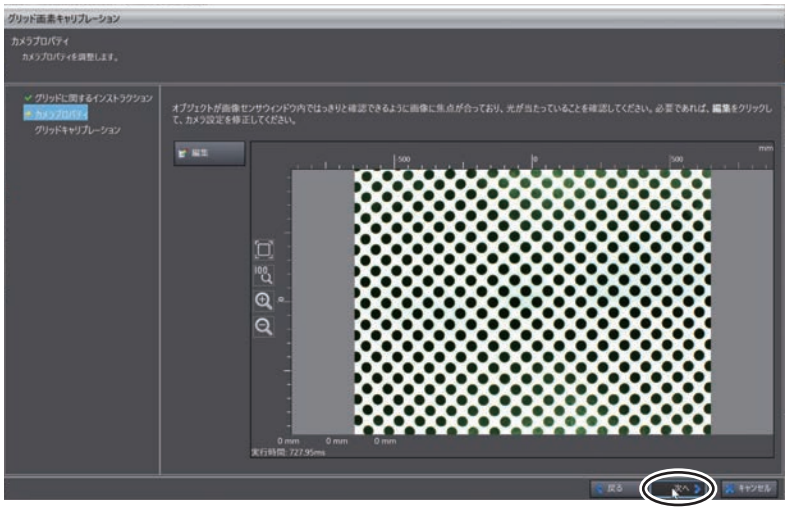
5 **ドットがはっきり見えない場合は、[編集] ボタンをクリックします。**



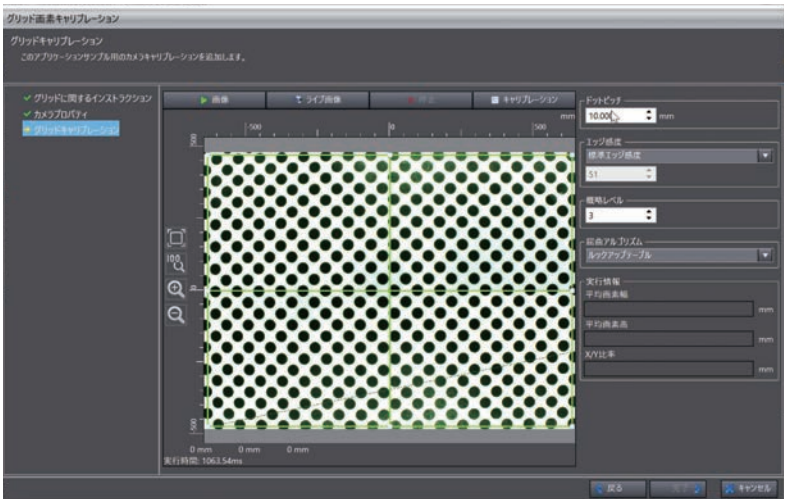
[カメラプロパティ] 画面が表示されますので、ドットがはっきりと見えるように調整し、[確定] ボタンをクリックして [カメラプロパティ] 画面を閉じます。



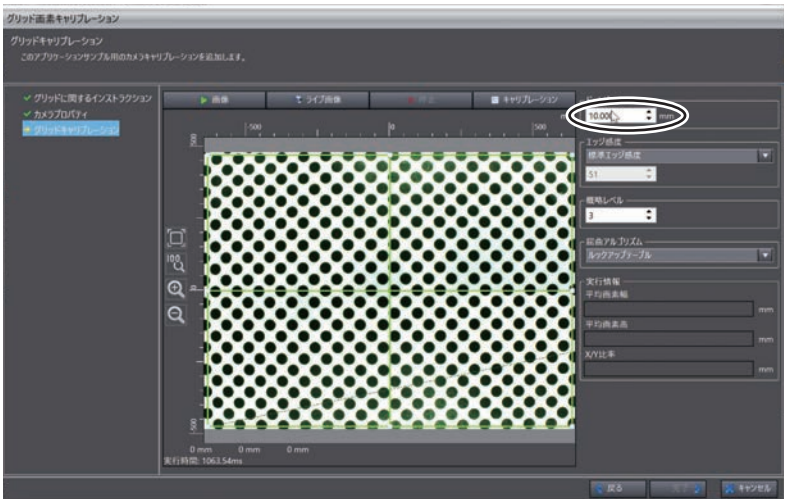
6 **[次へ] ボタンをクリックします。**



グリッドをキャリブレーションする画面が表示されます。

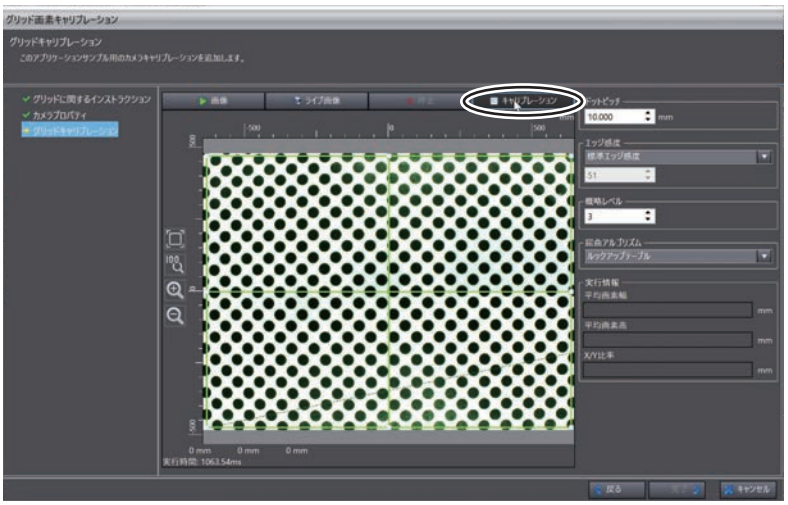


7 **［ドットピッチ］に使用しているキャリブレーショングリッドのドットピッチを入力します。**

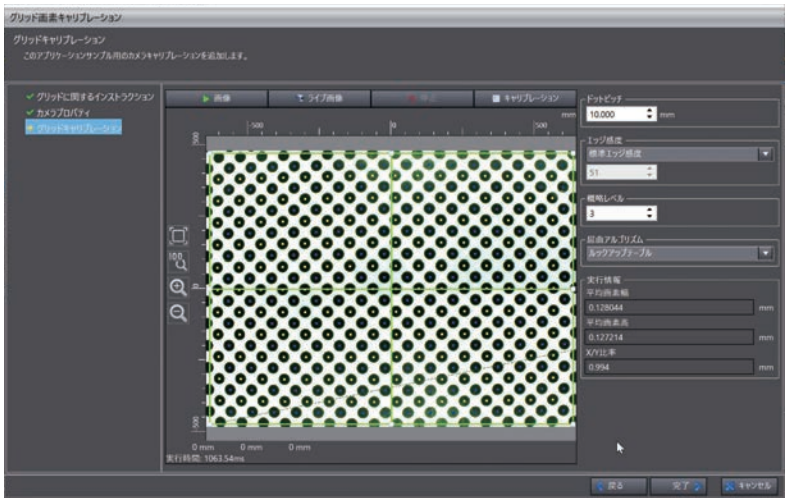


本説明では、ドットピッチが 10mm のキャリブレーショングリッドを使用しているため、「10.00」を入力しています。

8 **［キャリブレーション］ボタンをクリックします。**

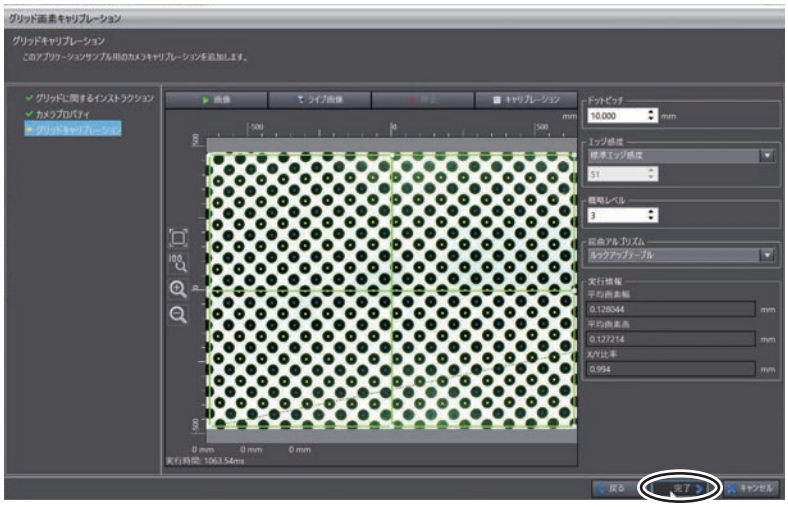


キャリブレーションが実行されます。



黒いドットの中に黄色い点が表示されれば正しくキャリブレーションされています。黄色い点が表示されない場合は、［戻る］ボタンで手順 4 の画面まで戻り、ドットがよりはっきり見えるように調整します。

9 **［完了］ボタンをクリックします。**



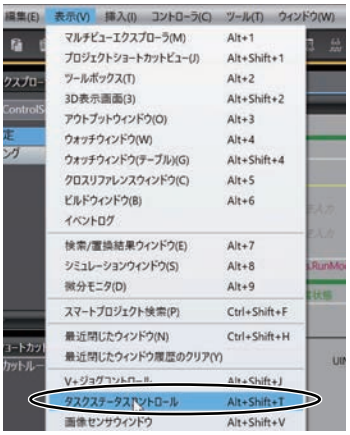
［グリッド画素キャリブレーション］画面が閉じます。

4-5-5 シーケンス制御プログラム、V+プログラムを実行する

キャリブレーションのためにコンベアを動作できるように、シーケンス制御プログラムおよび V+プログラムを実行します。

1 **メニューバーから［コントローラ］–［動作モード］–［運転モード］をクリックして運転モードにします。**

2 **メニューバーから［表示］–［タスクステータスコントロール］をクリックします。**



［タスクステータスコントロール］画面が表示されます。

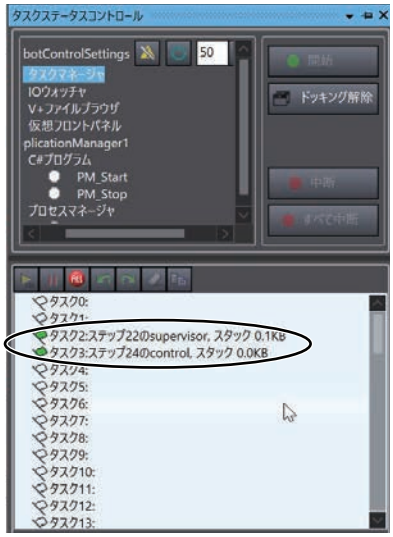
3 **マルチビューエクスプローラのデバイスで、［RobotControlSettings］を選択します。**



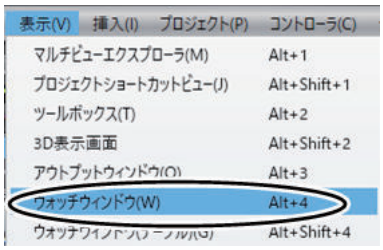
4 **マルチビューエクスプローラの［プログラミング］－［V+モジュール］－［supervisor］－［supervisor］を右クリックし、［タスク上で実行］－［タスク 2］をクリックします。**



V+プログラムの supervisor がタスク 2、contorl がタスク 3 で実行されます。



5 **メインウィンドウのメニューバーから［表示］–［ウォッチウィンドウ］を選択します。**



メインウィンドウの下に［ウォッチウィンドウ（プロジェクト）1］画面が表示されます。

6 **［ウォッチウィンドウ（プロジェクト）1］画面で変数［gStart］の［変更］列の［TRUE］ボタンをクリックします。**

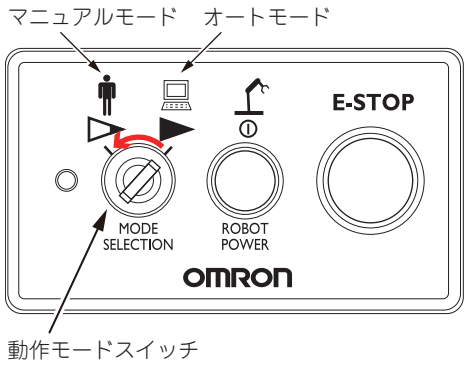


プログラムが実行されます。

4-5-6 ロボットのハイパワーを ON する

ロボットのハイパワーを ON します。

1 **フロントパネルの動作モードスイッチをマニュアルモードにします。**



2 **T20 ペンダントのイネーブルスイッチを中間位置（ポジション 2）まで押します。**



3 **イネーブルスイッチを中間位置（ポジション 2）にしたまま、ROBOT POWER ボタンを押します。**



フロントパネルの ROBOT POWER ボタンが点滅します。

4 **フロントパネルの ROBOT POWER ボタンを押します。**
ロボットのハイパワーが ON になり、ROBOT POWER ボタンが点灯します。

イネーブルスイッチを離したり（ポジション 1）、中間位置（ポジション 2）からさらに握り込んだり（ポジション 3）すると、ロボットのハイパワーは OFF になります。

使用上の注意

ロボットの移動が必要な場合は、上記の手順でハイパワーを ON にしてください。

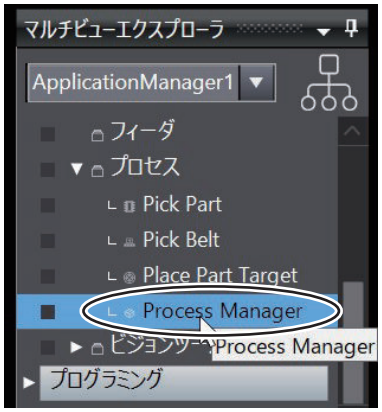
4-5-7 ベルトのキャリブレーションを行う

ベルトのキャリブレーションを行います。

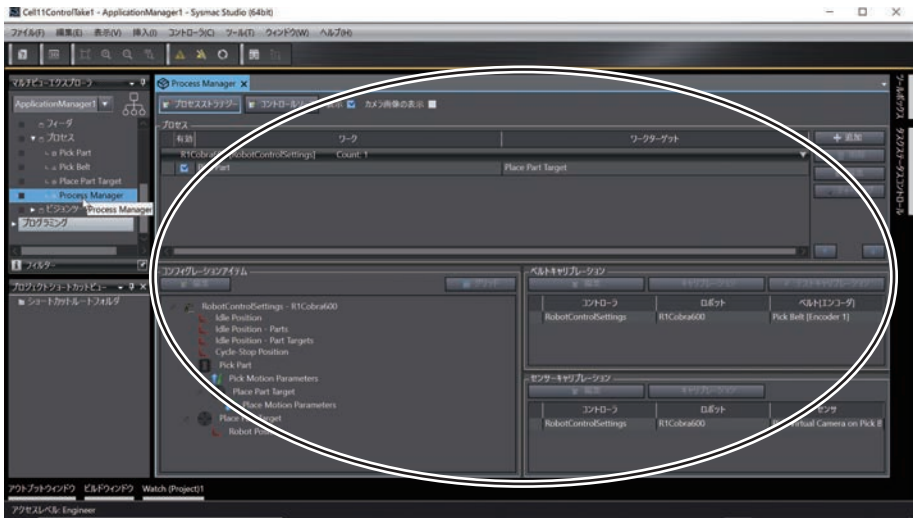
1 **マルチビューエクスプローラのデバイスリストから［ApplicationManager1］を選択します。**



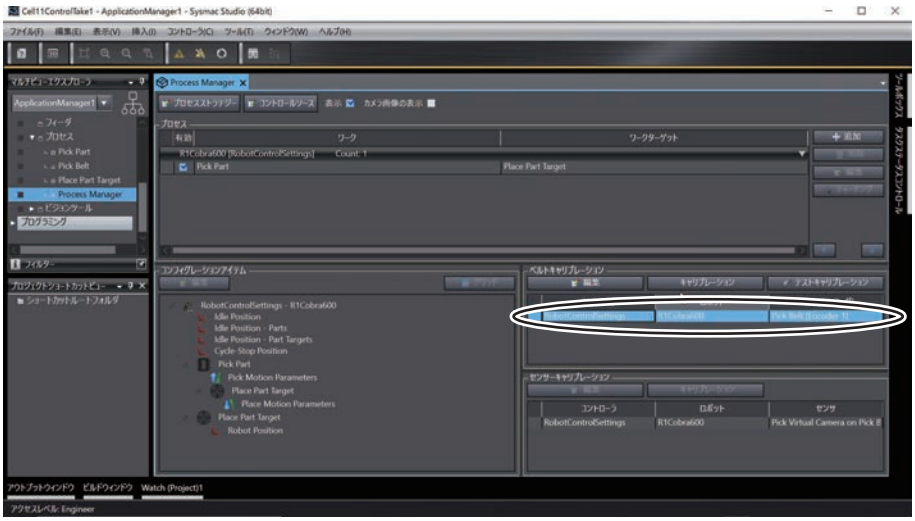
2 **マルチビューエクスプローラの［構成・設定］–［プロセス］–［Process Manager］をダブルクリックします。**



［Process Manager］画面が表示されます。



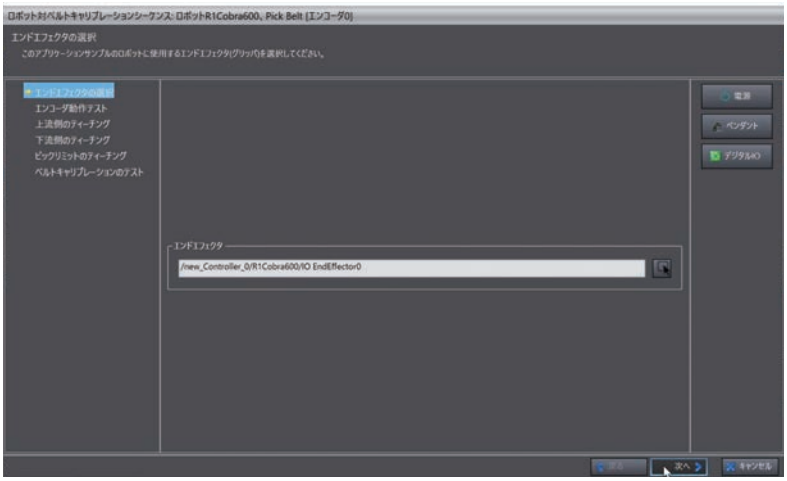
3 **［ベルトキャリブレーション］で、キャリブレーションするベルトを含む行をクリックします。**



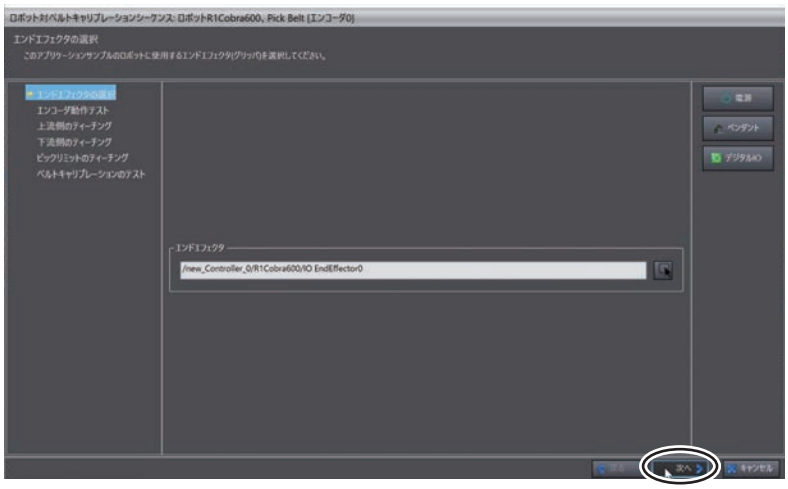
4 **［ベルトキャリブレーション］の［キャリブレーション］ボタンをクリックします。**



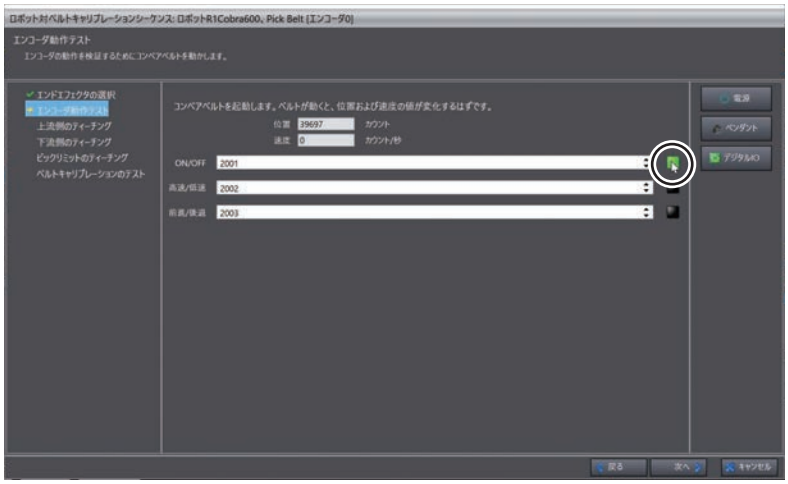
［ロボット対ベルトキャリブレーションシーケンス］画面が表示されます。



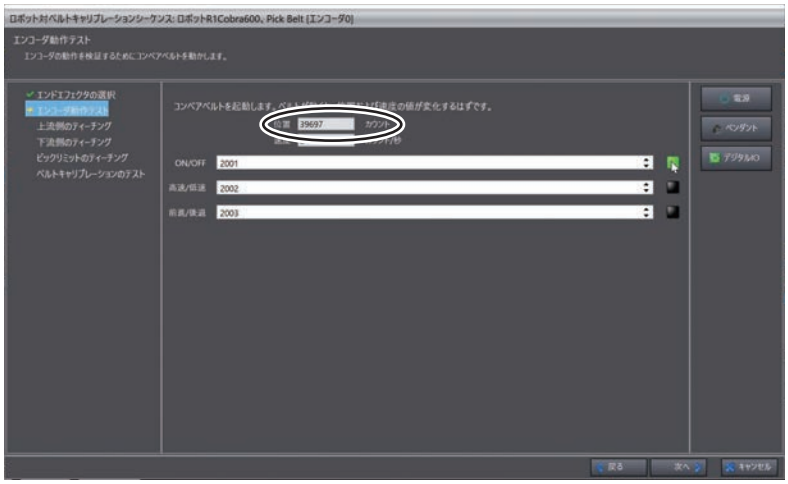
5 **エンドエフェクタが正しく選択されていることを確認し、［次へ］ボタンをクリックします。**



6 **[ON/OFF] のランプをクリックして点灯させます。**

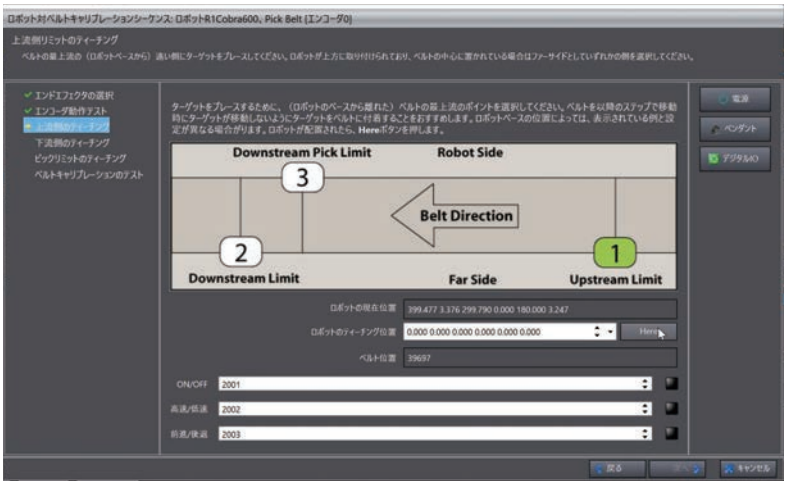


コンベアベルトが動作し、[位置] のカウントが変化することを確認します。



7 **[ON/OFF] のランプをクリックして消灯させます。**
コンベアベルトが停止し、[位置] のカウントの変化が停止します。

8 **[次へ] ボタンをクリックします。**
上流側リミットのティーチング画面が表示されます。



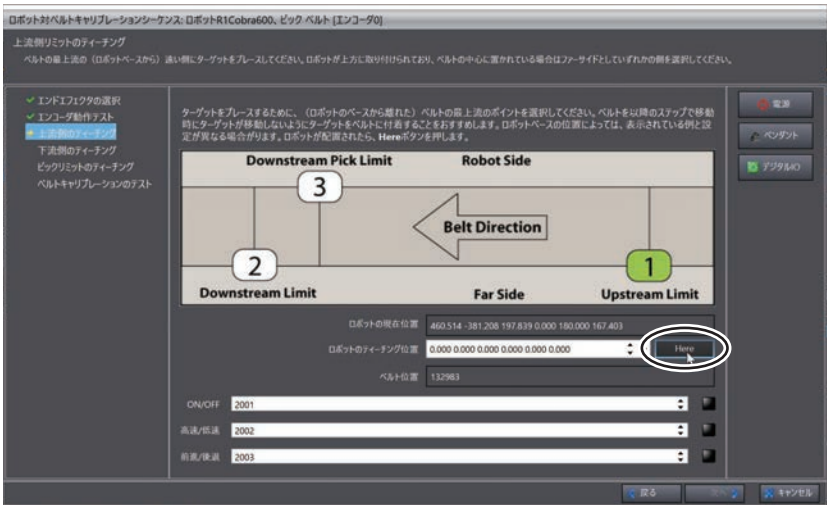
9 **ベルト上の上流側リミット位置にターゲットを置きます。**

10 **T20 ペンダントでロボットを操作し、上流側リミット位置まで移動させます。**

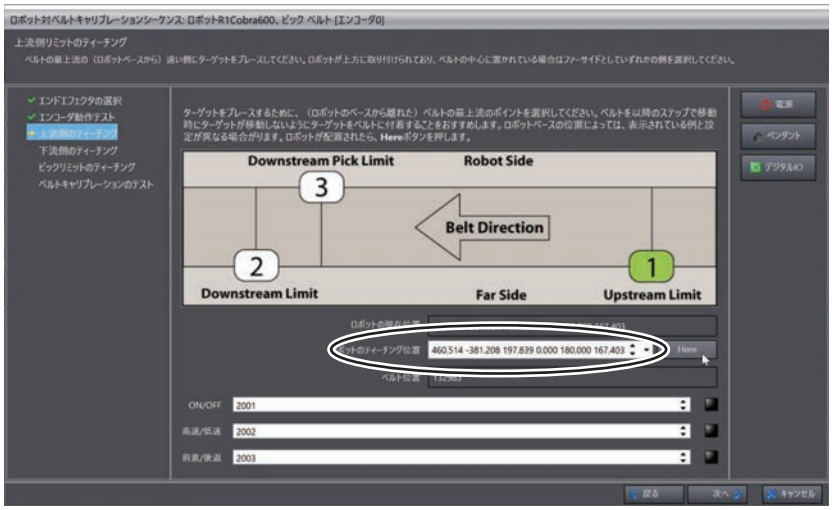
参考

［ペンダント］ボタンをクリックして表示される［ロボットジョグ動作］画面でロボットを移動することも可能です。

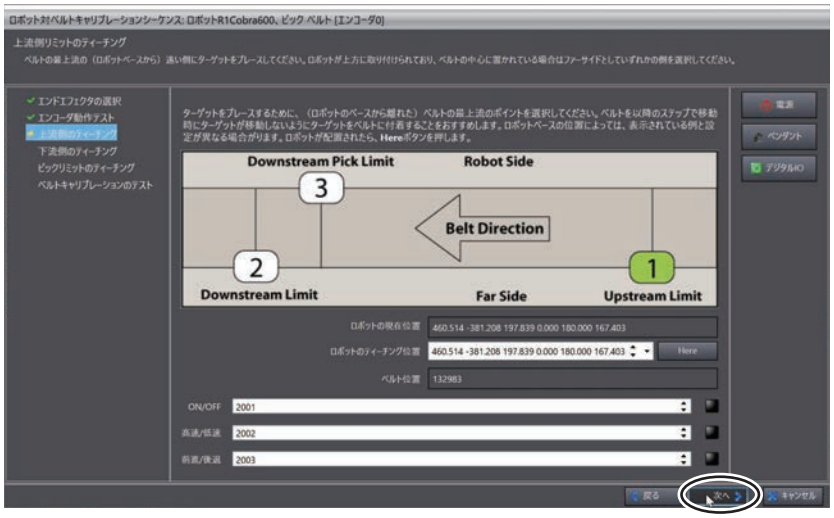
11 **［Here］ボタンをクリックします。**



［ロボットのティーチング位置］に現在のロボットの位置が設定されます。



12 [次へ] ボタンをクリックします。



13 [ON/OFF] ランプ、[前進/後退] ランプをクリックしてベルトを操作し、ターゲットが下流側リミットの位置にくるようにベルトを動かします。



14 手順 10～手順 12 と同様に下流側リミットをロボットにティーチングします。

15 [ON/OFF] ランプ、[前進/後退] ランプをクリックしてベルトを操作し、ターゲットがピックリミットの位置にくるようにベルトを動かします。

16 手順 10～手順 12 と同様にピックリミットをロボットにティーチングします。

17 **フロントパネルでマニュアルモードからオートモードに切り替えた後、画面上の［電源］ボタンをクリックします。フロントパネルの ROBOT POWER ボタンが点滅したら、ROBOT POWER ボタンを押してロボットのハイパワーを ON にします。**

18 **ロボットをベルトの上流側に移動してから、［ベルトキャリブレーションのテスト］画面の［ベルト座標変換］にある［トラッキング開始］ボタンをクリックします。**



ロボットがベルトの動きに追従して移動します。

19 **ロボットがベルトの動きに追従することを確認し、［トラッキング停止］ボタンをクリックします。**



トラッキングが停止します。

20 **［完了］ボタンをクリックします。**
［ロボット対ベルトキャリブレーションシーケンス］画面が閉じます。

4-5-8 センサのキャリブレーションを行う

センサのキャリブレーションを行います。

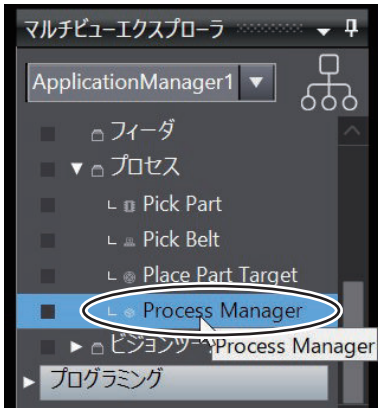
参考

- 本手順の中にターゲットを検出するための手順が含まれますが、センサのキャリブレーションに必要な位置検出のためです。実際のターゲットを検出するためのロケータモデルの設定は、「4-5-9 ロケータを設定する (P.4-146)」で行います。
- ターゲットが検出できない場合は、カメラの撮影環境や、カメラの撮影設定を調整してください。カメラの撮影設定についての詳細は、『オートメーション コントロール エンバイロメント(ACE) Version 4 オペレーションマニュアル(SBCE-456)』のバーチャルカメラの撮影設定に関する説明を参照してください。

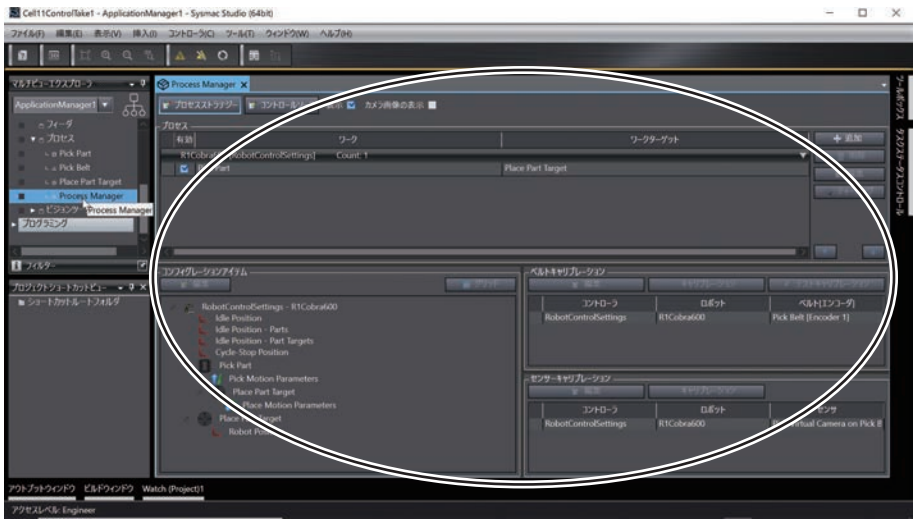
1 **マルチビューエクスプローラのデバイスリストから［ApplicationManager1］を選択します。**



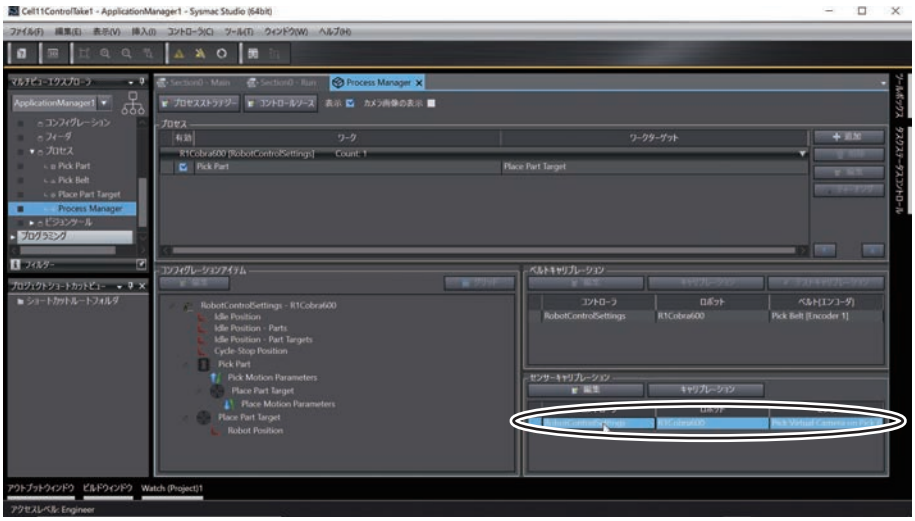
2 **マルチビューエクスプローラの［構成・設定］–［プロセス］–［Process Manager］をダブルクリックします。**



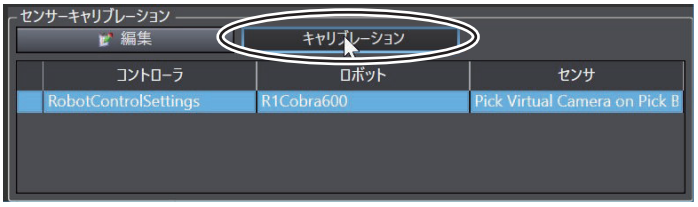
［Process Manager］画面が表示されます。



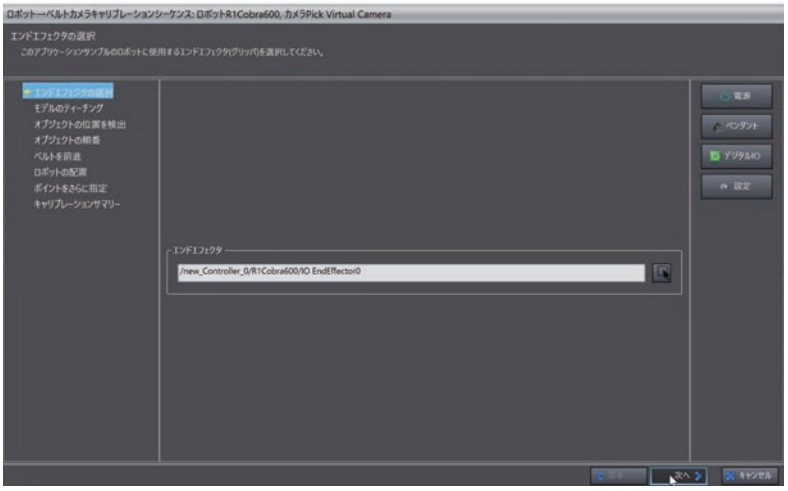
3 **［センサーキャリブレーション］で、キャリブレーションするセンサを含む行をクリックします。**



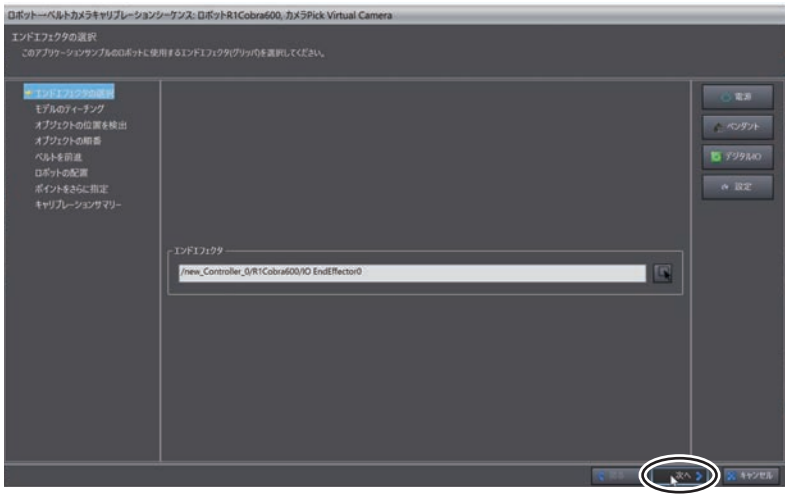
4 **［センサーキャリブレーション］の［キャリブレーション］ボタンをクリックします。**



［ロボット→ベルトカメラキャリブレーションシーケンス］画面が表示されます。



5 **エンドエフェクタが正しく選択されていることを確認し、［次へ］ボタンをクリックします。**



6 **[ライブ画像] ボタンを押して、表示される画像を見ながらカメラの視野にターゲットを配置します。**



7 **[停止] ボタンをクリックします。**



ターゲットを指定するための緑色の枠が表示されます。



8 **緑色の枠をマウスでドラッグ & ドロップして移動、拡大／縮小しターゲットの 1 つを囲みます。**



9 **［実行］ボタンをクリックします。**



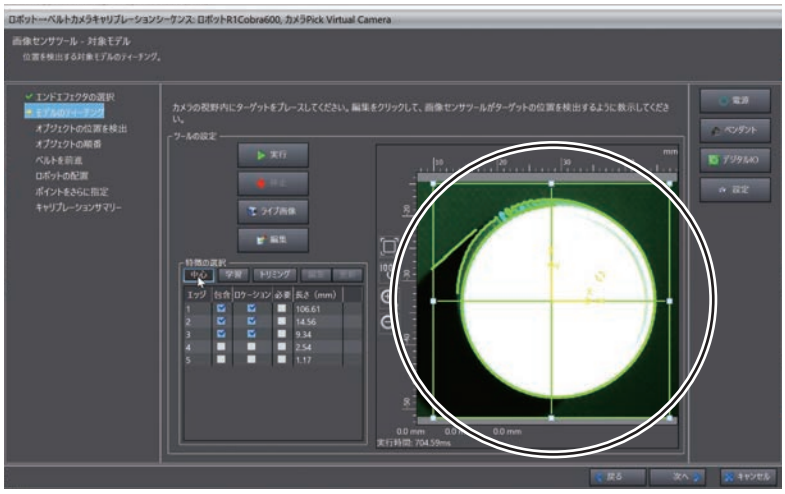
囲んだターゲットの特徴が検出されます。



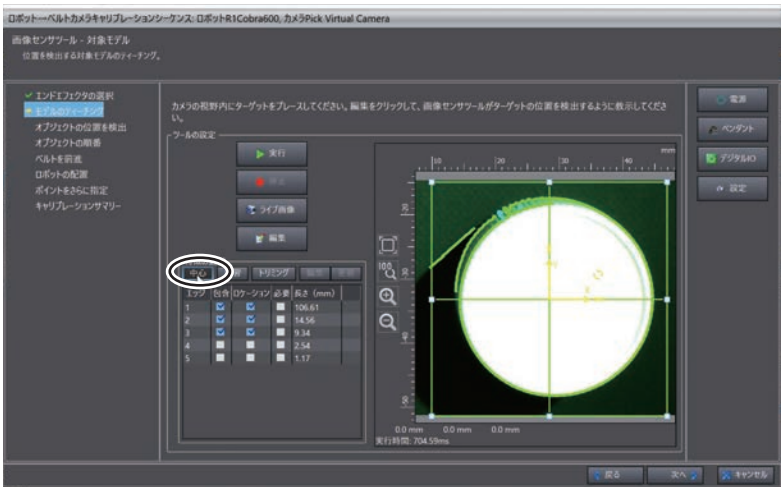
10 **[トリミング] ボタンを押します。**



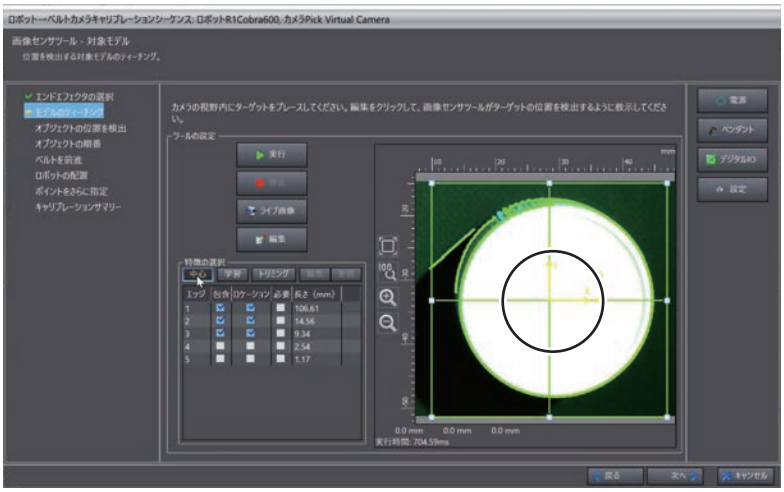
囲んだターゲットが拡大表示されます。



11 **[中心] ボタンをクリックします。**

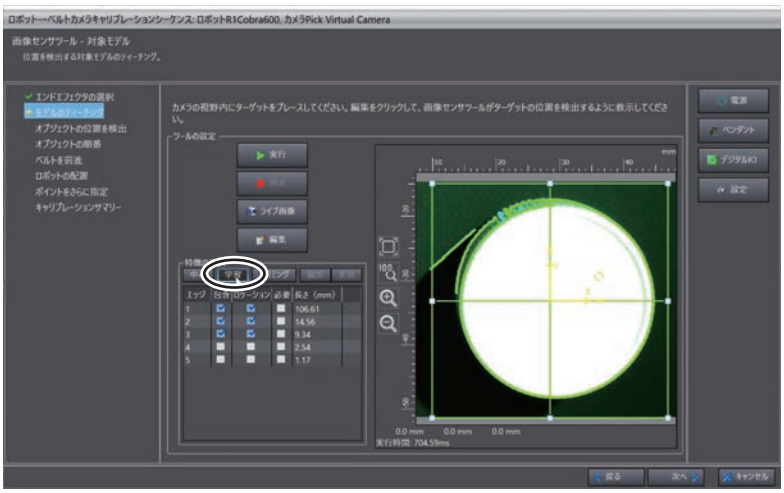


検出されたターゲットの中心に黄色のマークが表示されます。

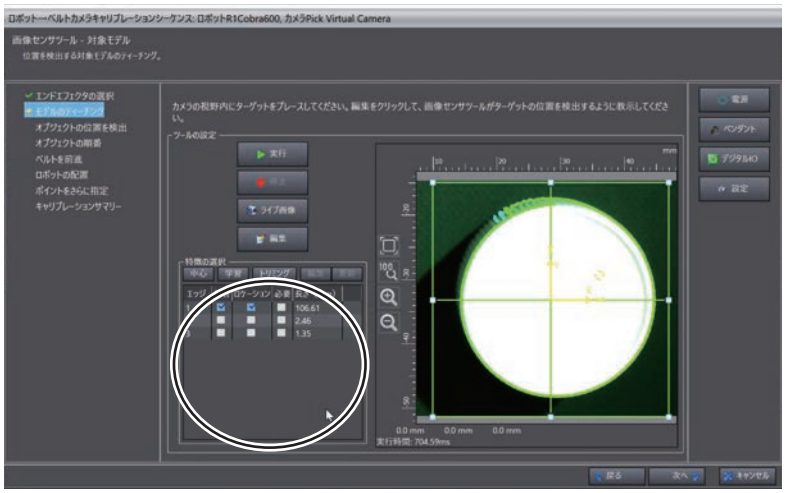


検出された中心位置が不適切な場合は、マークの矢印をドラッグ&ドロップして調整します。

12 **[学習] ボタンをクリックします。**

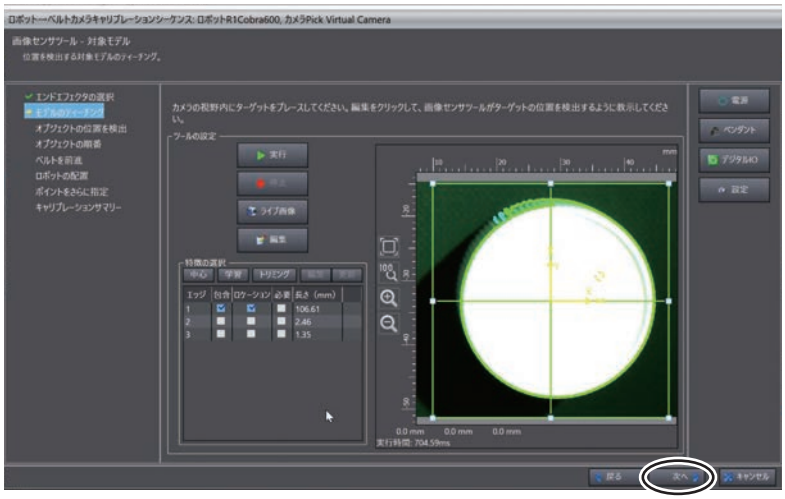


ターゲットの特徴が検出されます。

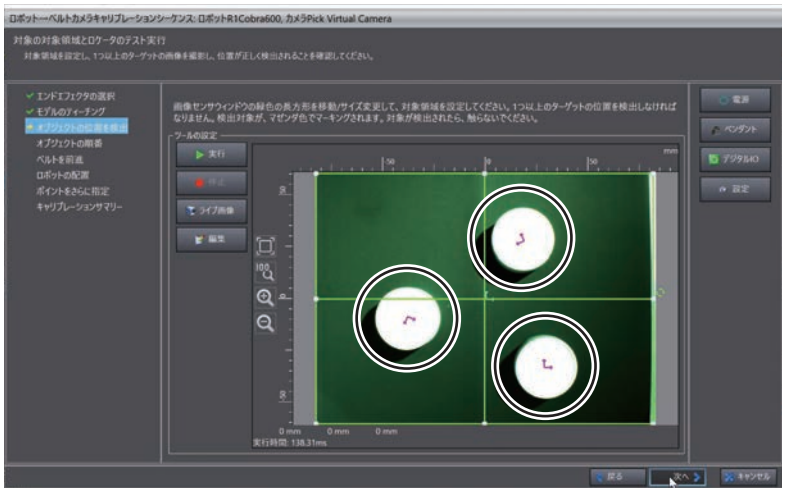


ターゲットの特徴として使用しない特徴がある場合は、使用しない特徴の［包含］のチェックを外します。［包含］のチェックを外したエッジの色は画像上で緑から赤に変わります。チェックを外した後、［更新］ボタンをクリックします。

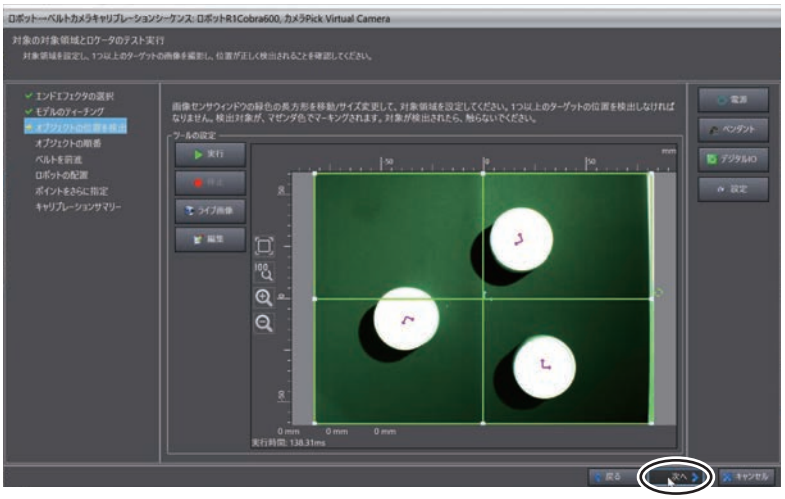
13 **［次へ］ボタンをクリックします。**



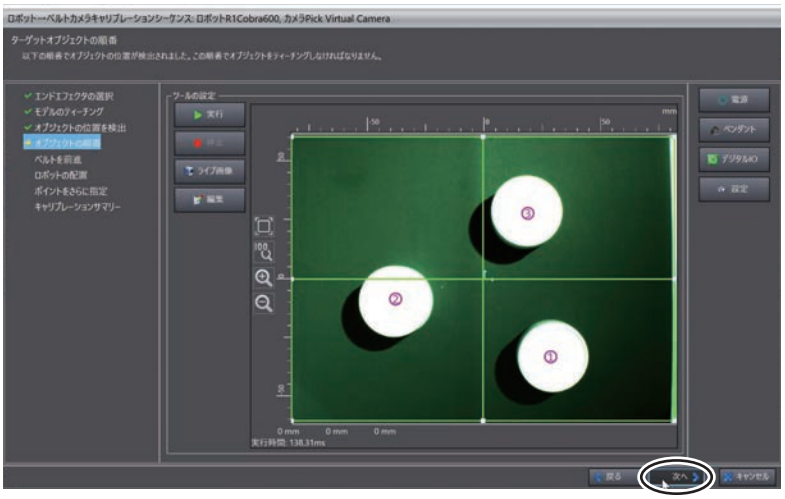
すべてのターゲットの位置が検出され、各ターゲットの中心付近にマゼンタ色のマークが付きます。



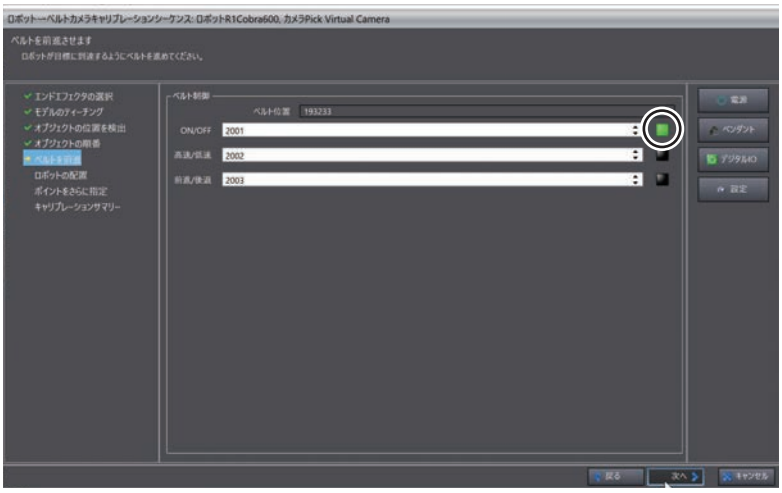
14 ターゲットが正しく検出されていることを確認し、［次へ］ボタンをクリックします。



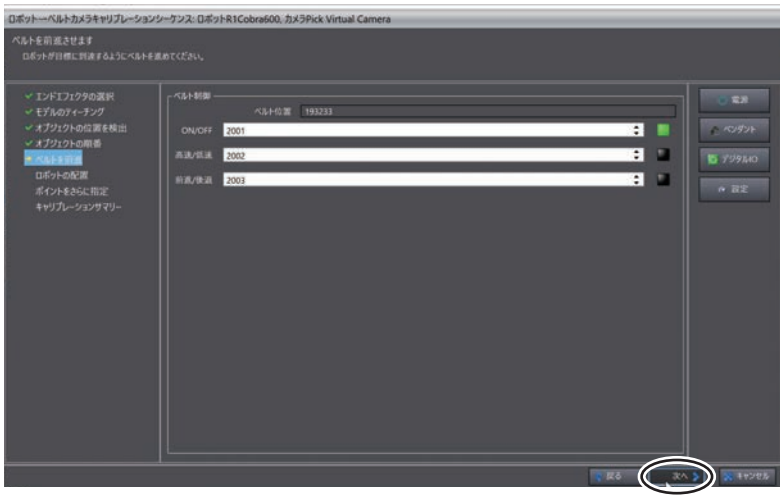
15 続いて表示されるターゲットオブジェクトの順番の画面で［次へ］ボタンをクリックします。



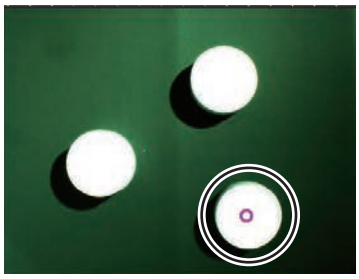
16 ベルト制御の［ON/OFF］でベルトを進めて、ロボットが到達できる位置までターゲットを移動します。



17 **[次へ] ボタンをクリックします。**



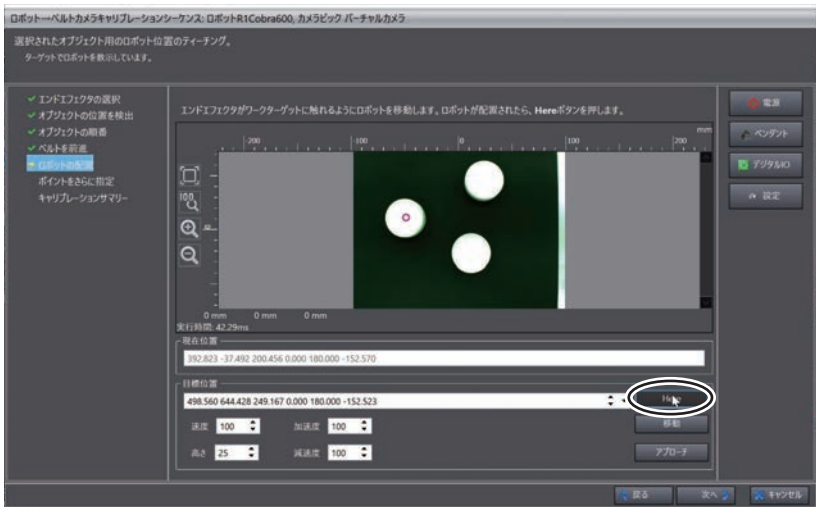
18 **T20 ペンダントでロボットを操作し、マークが表示されているターゲットにエンドエフェクタが触れるようにロボットを移動します。**



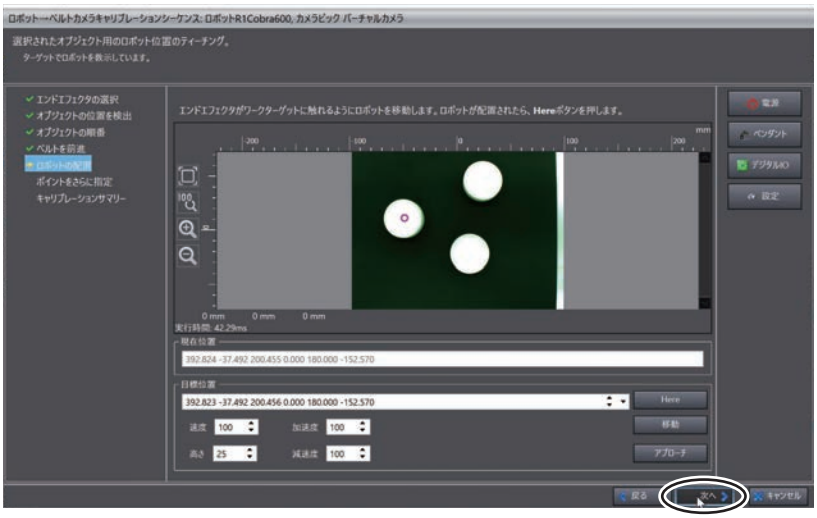
参考

［ペンダント］ボタンをクリックして表示される［ロボットジョグ動作］画面でロボットを移動することも可能です。

19 **[Here] ボタンをクリックします。**

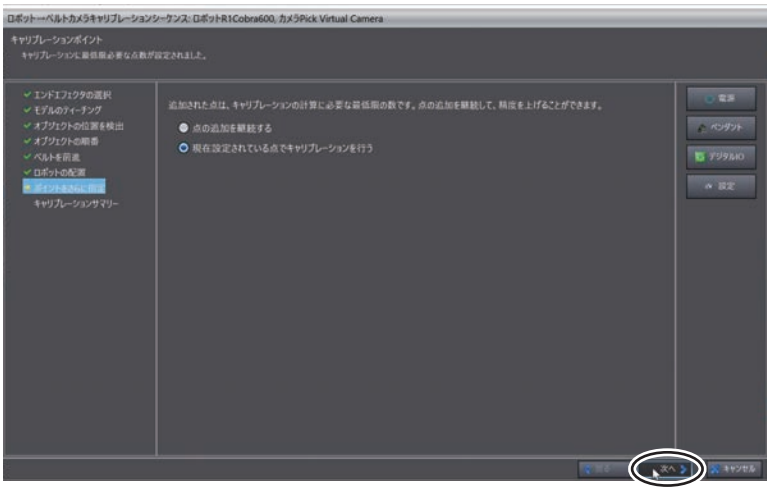


20 **[次へ] ボタンをクリックします。**



マークが次のターゲットに移動しますので、すべてのターゲットについて手順 19～手順 20 を同様に繰り返します。
なお、最低 4 つのターゲットについてティーチングが必要です。上記の例では 3 つのターゲットを置いてティーチングしましたので、あらためてカメラの視野にターゲットを置いて手順 14 から繰り返します。

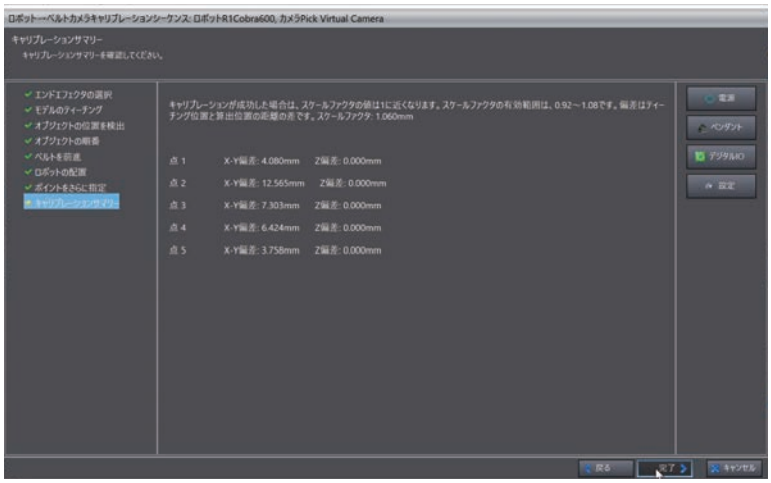
21 [次へ] ボタンをクリックします。



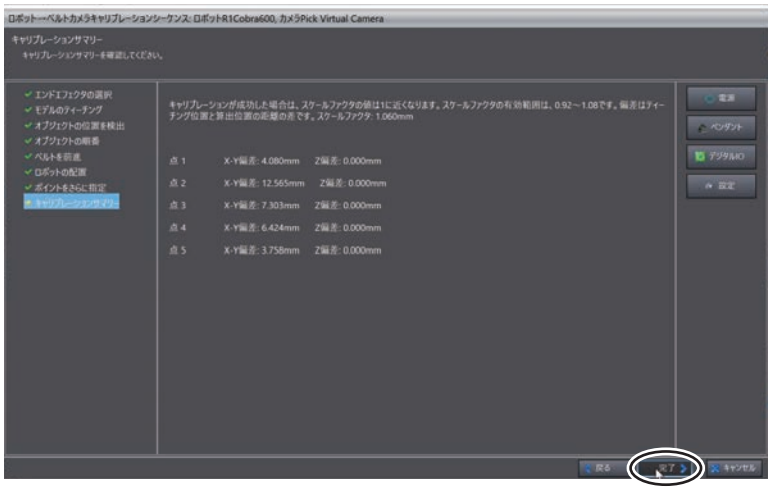
参考

ティーチングの精度を上げる場合は、[点の追加を継続する] を選択することで、追加のターゲットを使用してティーチングを続けることができます。

キャリブレーションサマリーの画面が表示されます。



22 **[次へ] ボタンをクリックします。**



［ロボット→ベルトカメラキャリブレーションシーケンス］画面が閉じます。

4-5-9 ロケータを設定する

ロケータを設定します。

参考

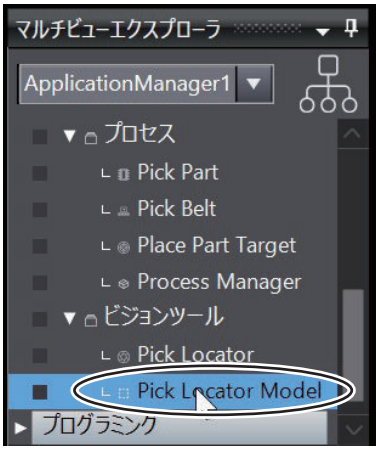
以下についての詳細な情報は、『オートメーション コントロール エンバイロメント(ACE) Version 4 オペレーションマニュアル(SBCE-456)』を参照してください。

項目	SBCE-456 参照先
ターゲットの形状や特徴を定義する、ロケータモデル設定項目（本節における Pick Locator Model）について	「画像センサツール」の「ロケータモデル設定項目」
ロケータ設定項目で定義される、検出したターゲットの各種プロパティの説明（本節における Pick Locator）について	「画像センサツール」の「ロケータ設定項目」

1 **マルチビューエクスプローラのデバイスリストから［ApplicationManager1］を選択します。**



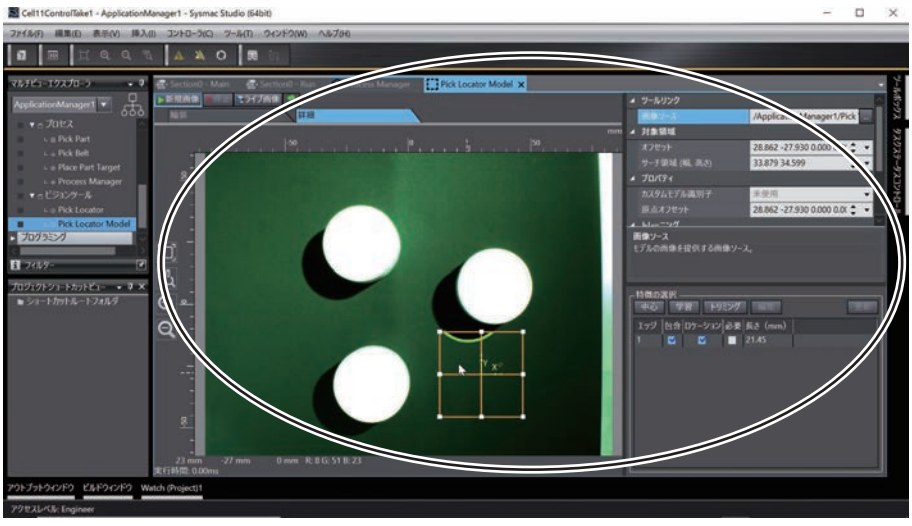
2 **マルチビューエクスプローラの［構成・設定］－［ビジョンツール］－［Pick Locator Model］をダブルクリックします。**



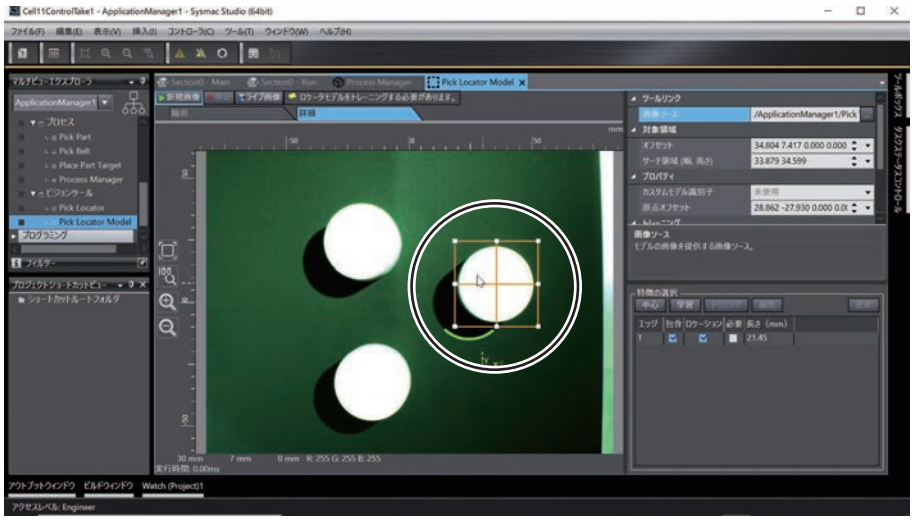
3 **［Pick Locator Model］画面が表示されますので、［ライブ画像］ボタンをクリックし、表示される画像を見ながらカメラの視野にターゲットを配置します。［停止］ボタンをクリックした後、［新規画像］ボタンをクリックします。**



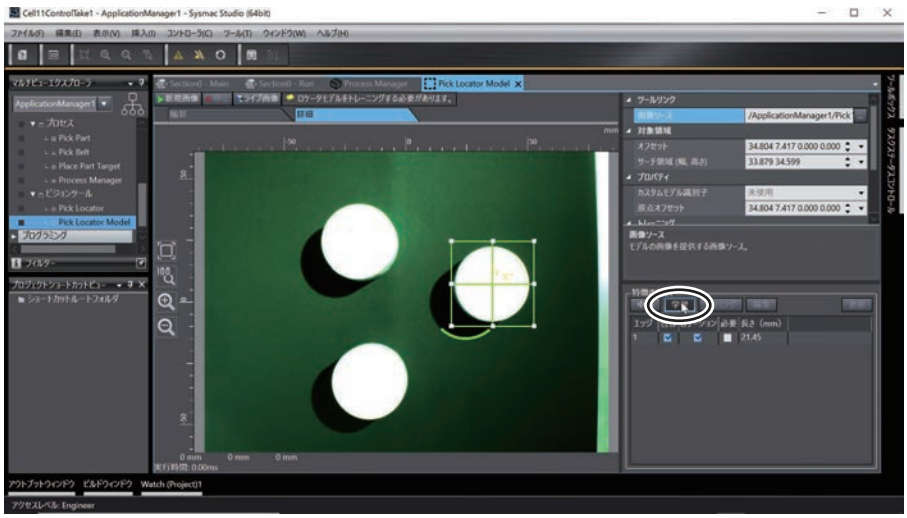
画像が固定され、緑色の枠が表示されます。



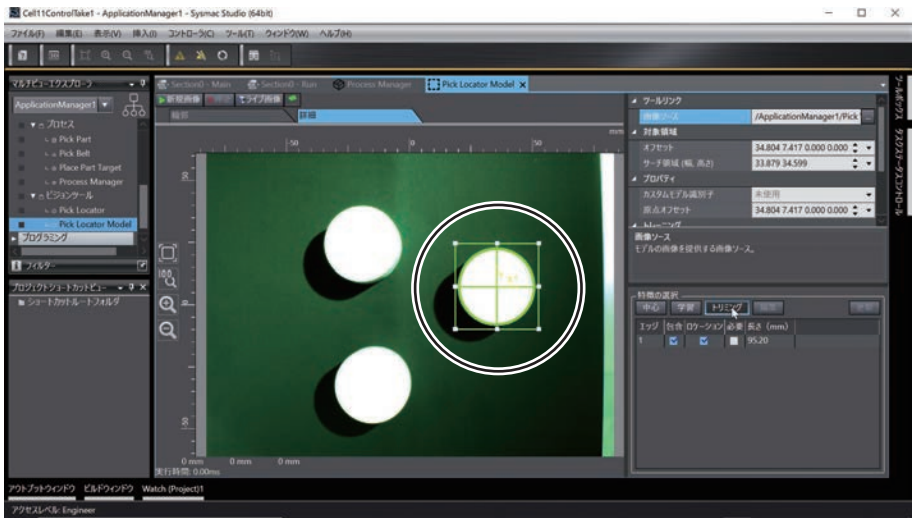
4 **緑色の枠をマウスでドラッグ＆ドロップして移動、拡大／縮小しターゲットの１つを囲みます。**



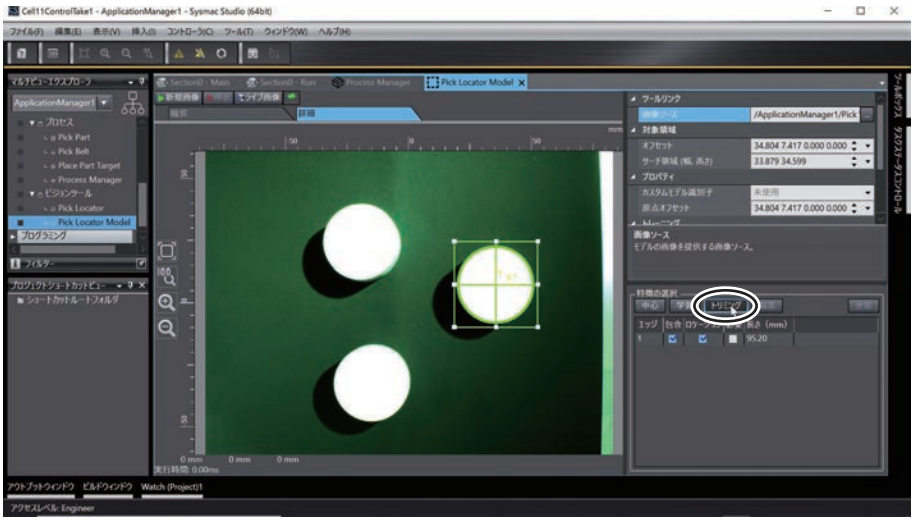
5 **[学習] ボタンをクリックします。**



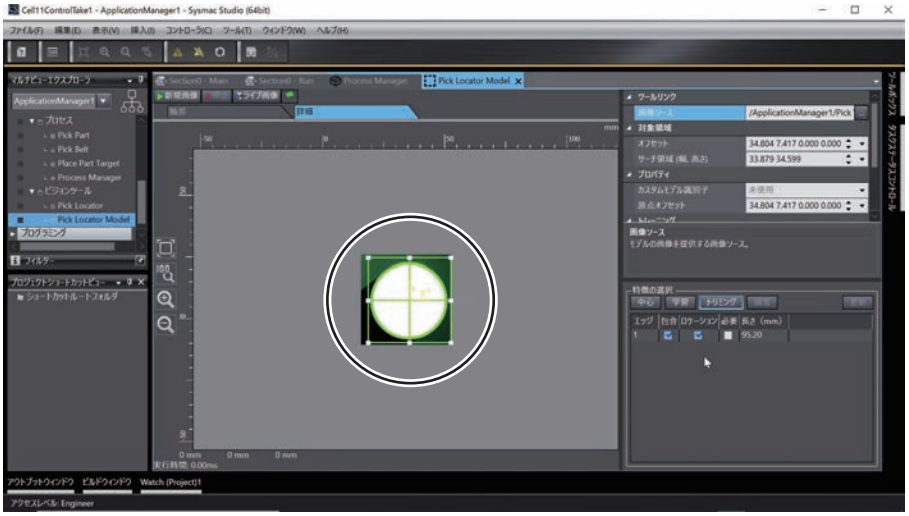
ターゲットの特徴が検出されて緑の線で表示されます。



6 **[トリミング] ボタンをクリックします。**

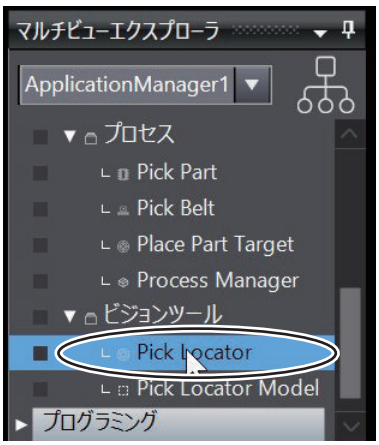


囲んだターゲットが拡大表示されます。

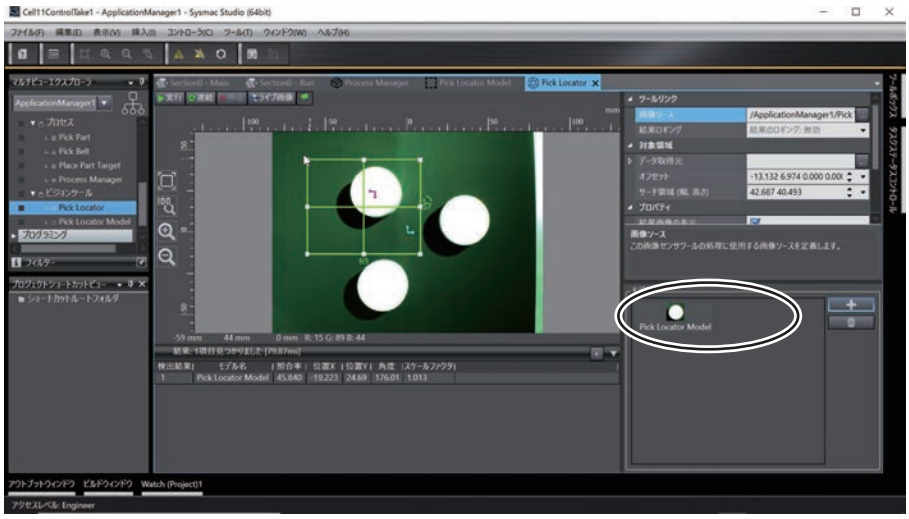


検出された位置や特徴が不適切な場合は、「4-5-8 センサのキャリブレーションを行う (P.4-135)」に記載されている方法と同様の方法で調整してください。

7 **マルチビューエクスプローラの［構成・設定］–［ビジョンツール］–［Pick Locator］をダブルクリックします。**



［モデル］に作成したモデルが登録されます。



［Pick Locator］画面で緑色の枠内のターゲットが検出され、検出されたターゲットの中心付近にマゼンタ色のマークが付くようになります。



8 緑色の枠をマウスでドラッグ＆ドロップし、撮影範囲全体まで広げます。



緑色の枠内のすべてのターゲットの位置が認識されます。



4-5-10　アイドル位置、ピック位置、プレース位置をティーチングする

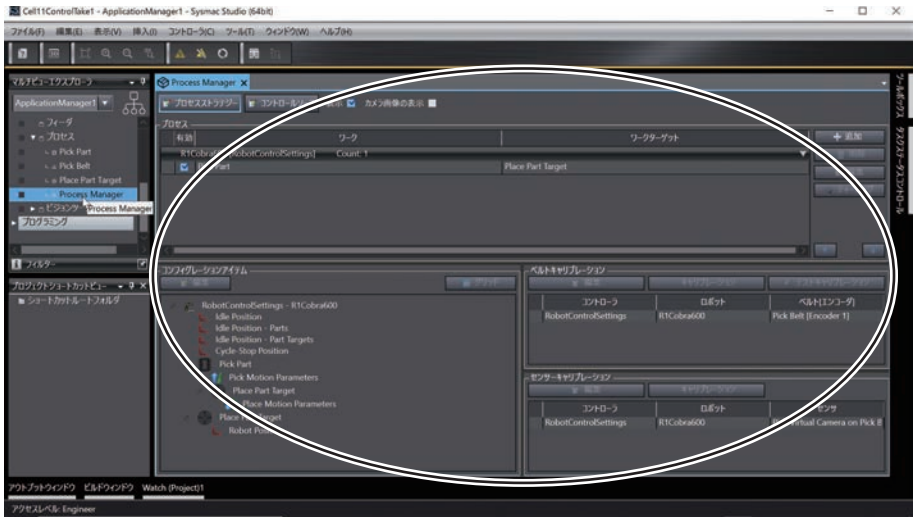
1　**マルチビューエクスプローラのデバイスで、［ApplicationManager1］を選択します。**



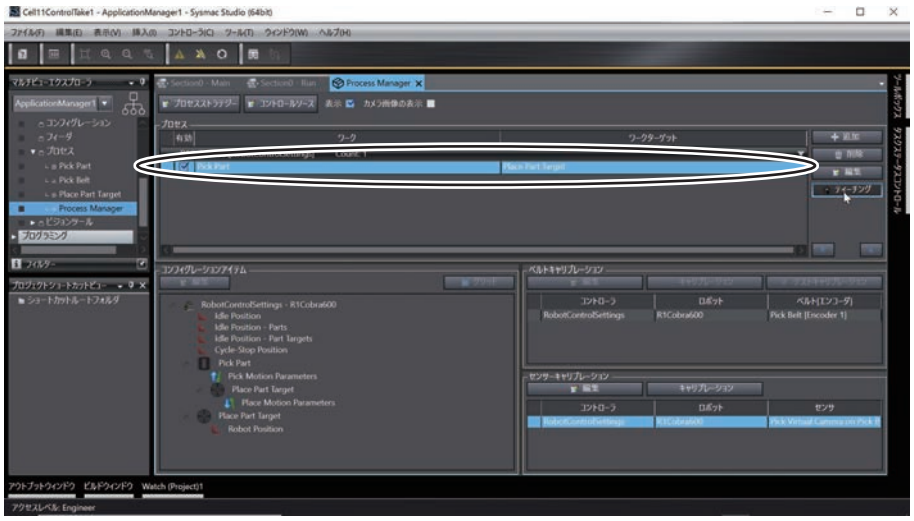
2　**［構成・設定］－［プロセス］－［Process Manager］をダブルクリックします。**



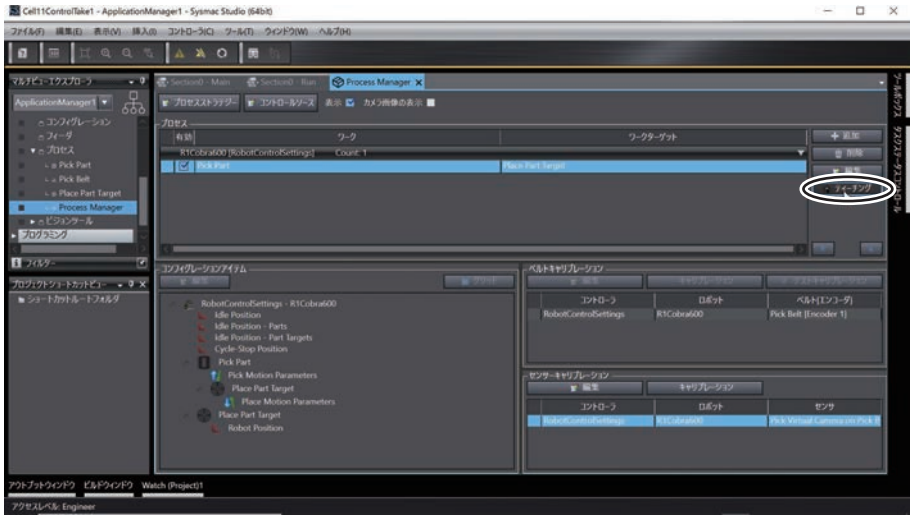
［Process Manager］画面が表示されます。



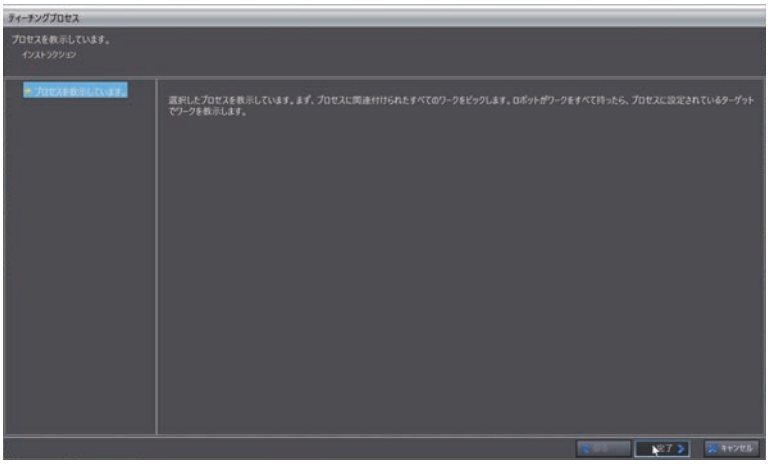
3 ［プロセス］リストからプロセスを選択します。



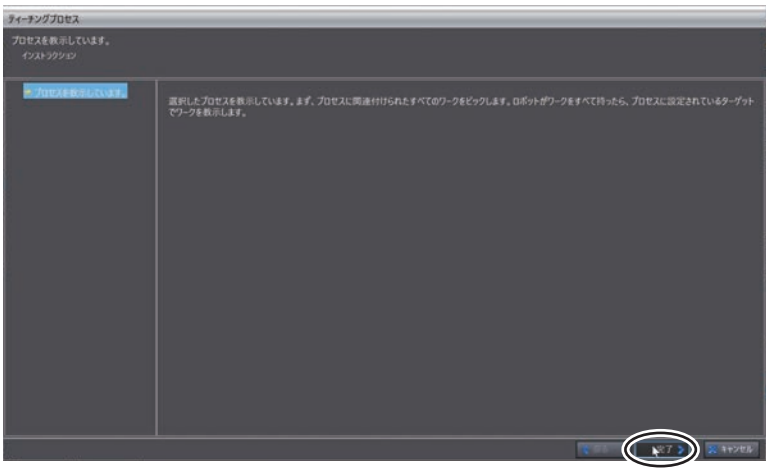
1) リスト右にある［ティーチング］ボタンをクリックします。



［ティーチングプロセス］画面が開きます。



4 **[完了] ボタンをクリックします。**

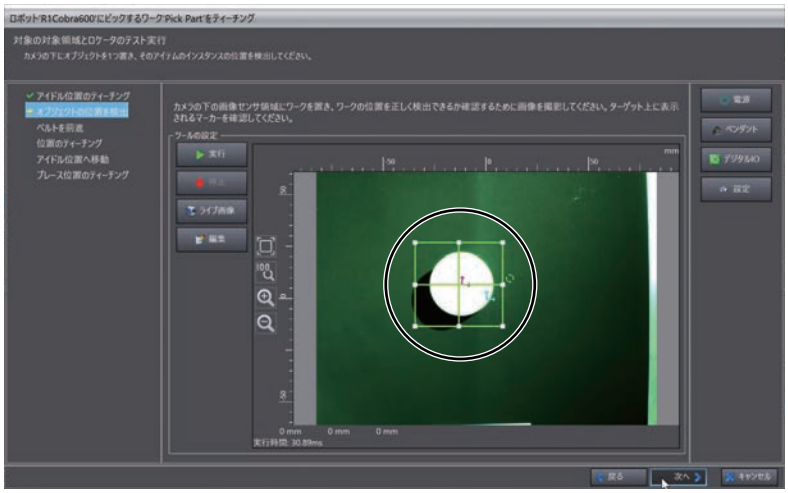


ティーチング画面が表示されます。

5 **以降、画面に従い各ティーチングの調整を行います。**

調整の方法はシミュレーションでの場合と同様です。
ロボットを移動する場合は、T20 ペンダントを使用して移動します。[ペンダント] ボタンをクリックし [ロボットジョグ動作] 画面で移動することもできます。

また、[オブジェクトの位置を検出] 画面では、ターゲットを 1 つだけ置いて位置を検出できることを確認します。



4-5-11　プログラムを実行して動作を確認する

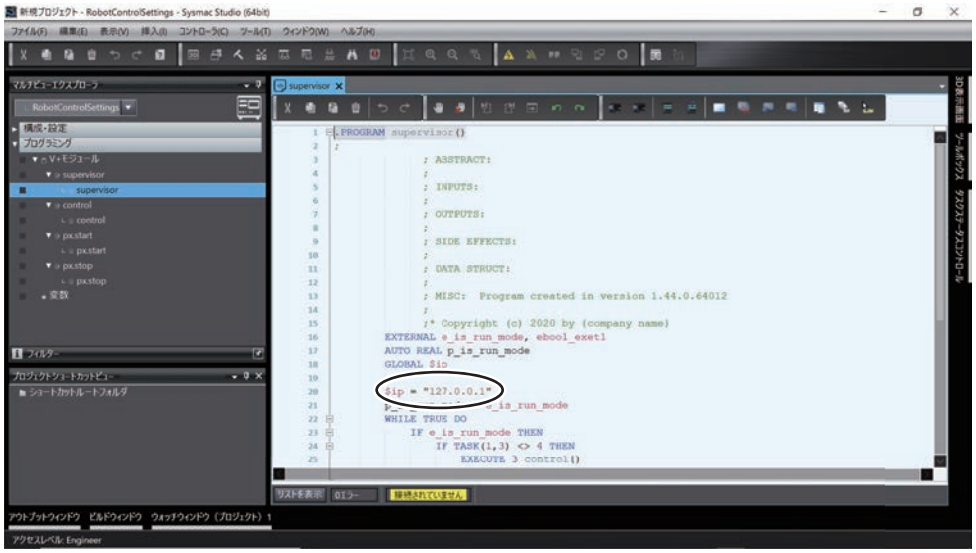
ロボット統合 CPU ユニットおよび IPC Application Controller の電源を ON にしたときに自動でプログラムが実行されることを確認します。

1　マルチビューエクスプローラのデバイスリストから［RobotControllerSettings］を選択します。

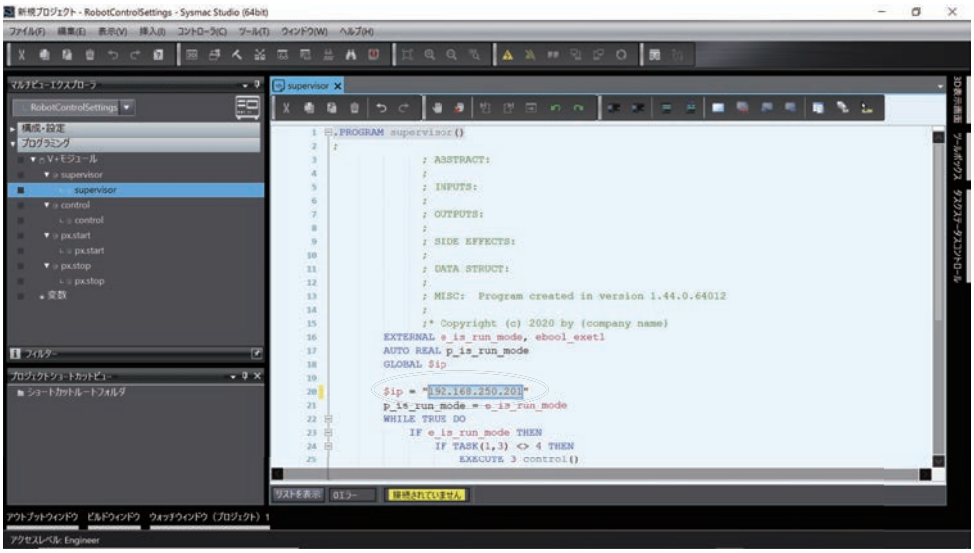


2　メニューバーから［コントローラ］–［オフライン］をクリックします。

3　V+プログラム supervisor の中で「$ip」に代入している IP アドレスの値を、IPC Application Controller の IP アドレスに書き換えます。



IPC Application Controller の IP アドレスが 192.168.250.201 の場合は次のように書き換えます。

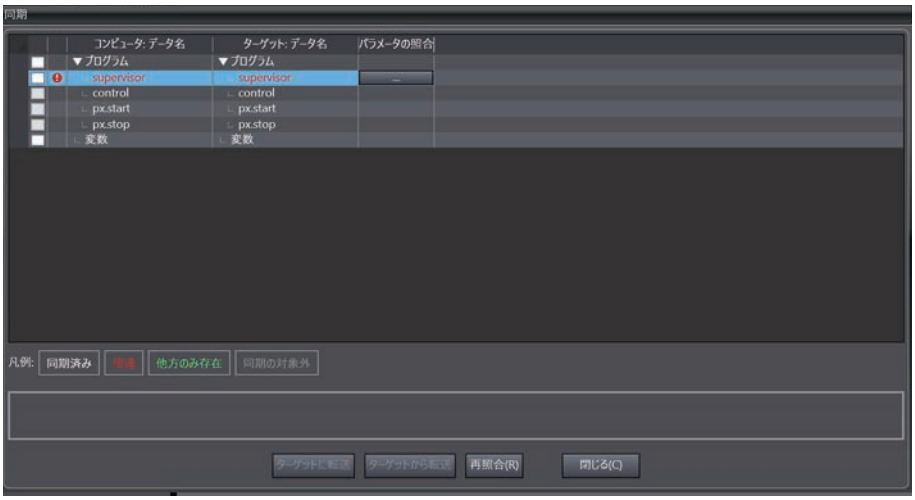


4 **メニューバーから［コントローラ］-［オンライン］をクリックします。**
オンライン状態になります。

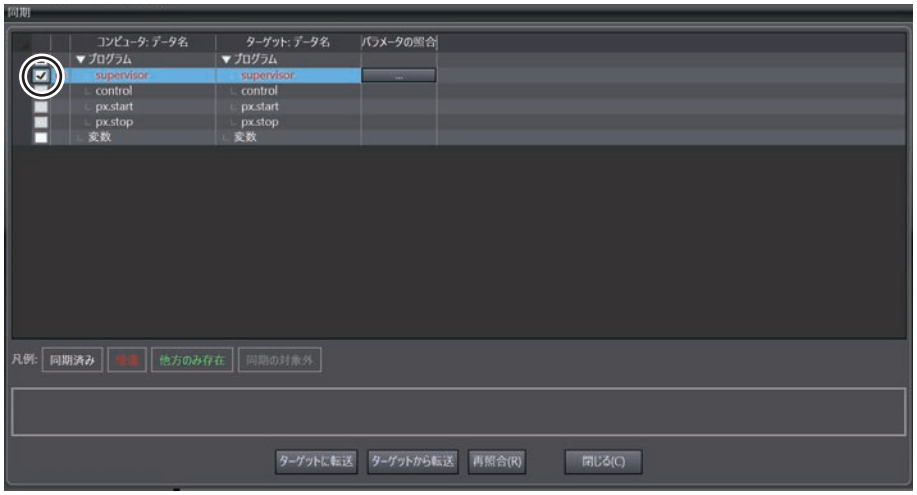
5 **ツールバーの同期のアイコンをクリックします。**



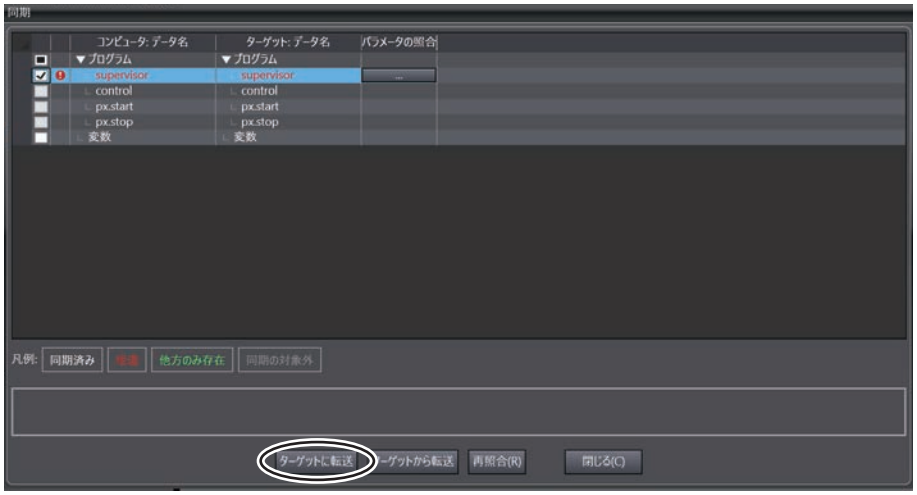
［同期］画面が表示されます。



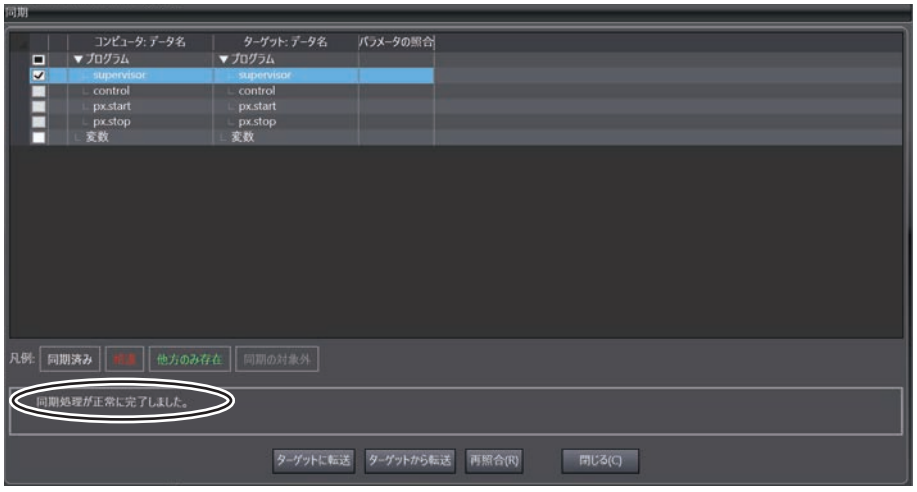
6 **［プログラム］-［supervisor］にチェックを入れます。**



7 **［ターゲットに転送］ボタンをクリックします。**



ロボット統合 CPU ユニットに supervisor のプログラムが転送されます。

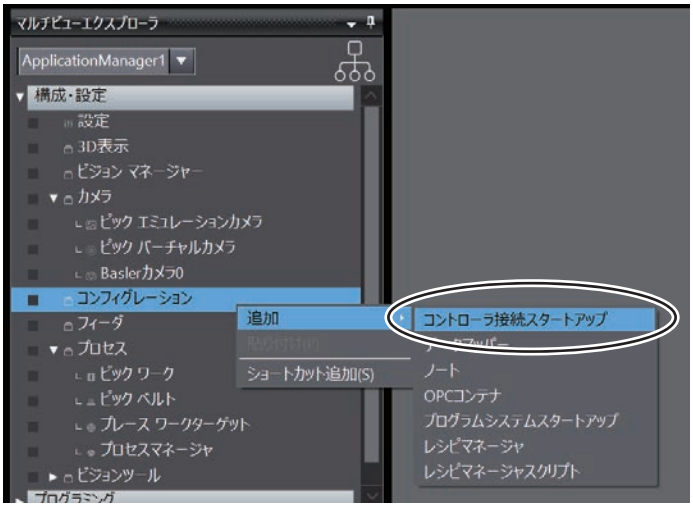


8 **［閉じる］ボタンをクリックします。**
［同期］画面が閉じます。

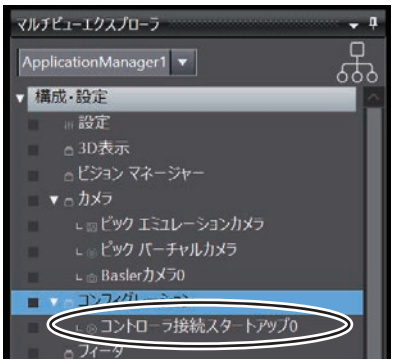
9 **マルチビューエクスプローラのデバイスで、［ApplicationManager1］を選択します。**



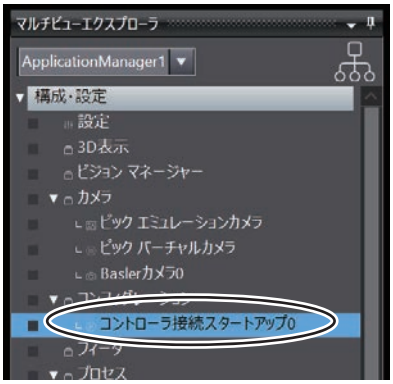
10 **［構成・設定］－［コンフィグレーション］－［追加］－［コントローラ接続スタートアップ］をクリックします。**



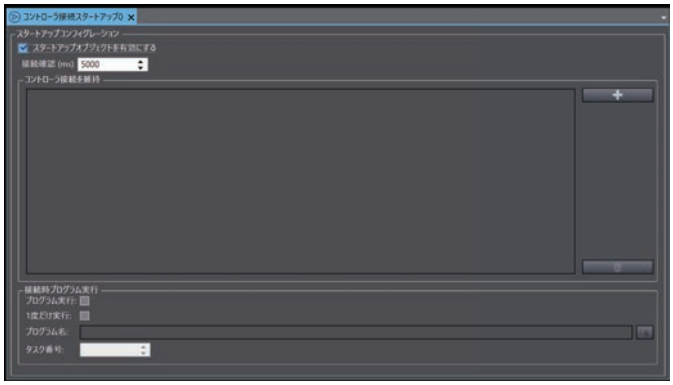
［コンフィグレーション］の下に、［コントローラ接続スタートアップ 0］が追加されます。



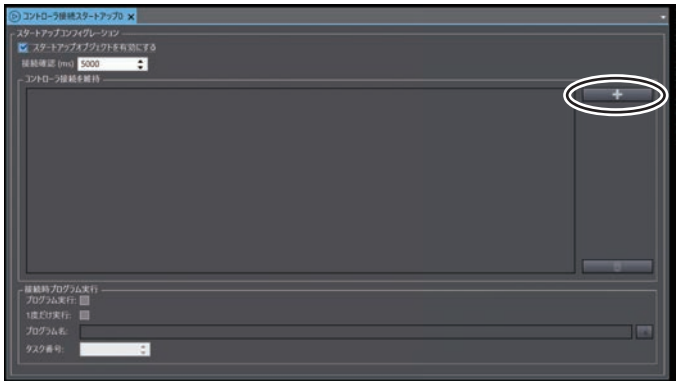
11 **［コントローラ接続スタートアップ 0］をダブルクリックします。**



［コントローラ接続スタートアップ 0］画面が表示されます。



12 **[+] ボタンをクリックします。**



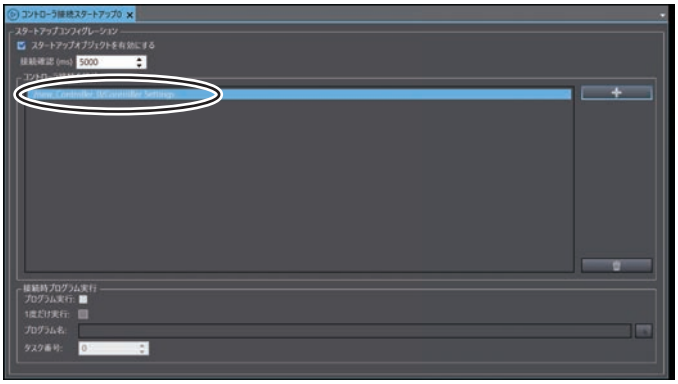
［基準を選択］画面が表示されます。



13 **［コントローラ設定］を選択して、［選択］ボタンをクリックします。**



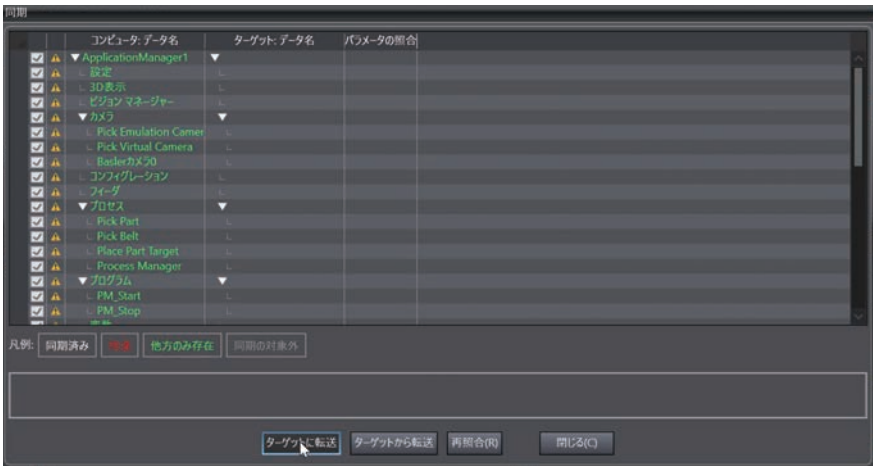
［コントローラ接続スタートアップ 0］画面に［/new_Controller_0/Controller Settings］が追加されます。



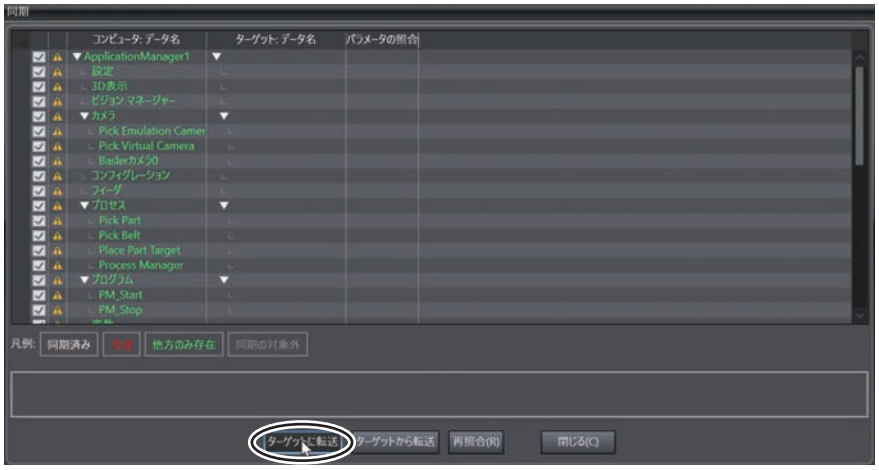
14 **ツールバーの同期のアイコンをクリックします。**



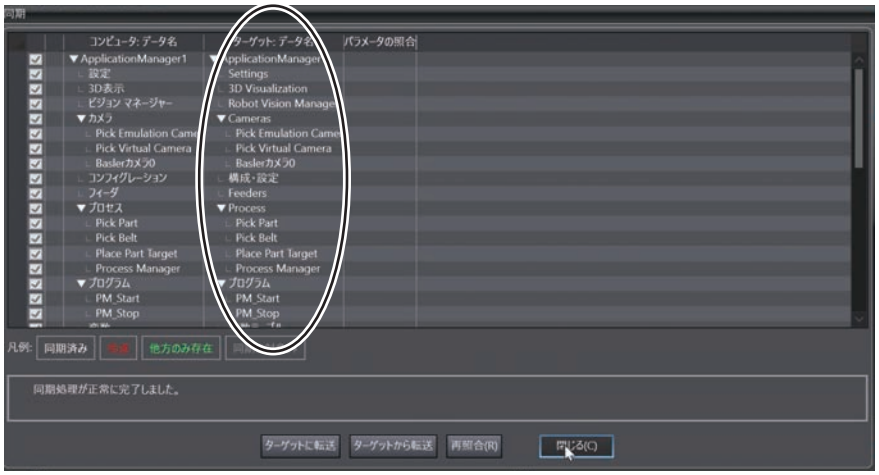
［同期］画面が表示されます。



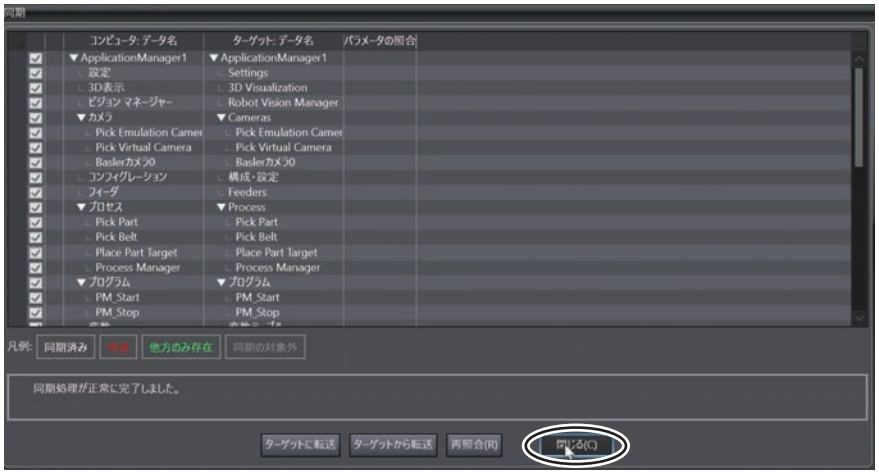
15 **［ターゲットに転送］ボタンをクリックします。**



転送先の IPC Application Controller に設定とプログラムが転送されます。



16 [閉じる] ボタンをクリックします。



［同期］画面が閉じます。

17 ロボット統合 CPU ユニットおよび IPC Application Controller とオフラインにします。

オフラインの方法については、「ロボット統合 CPU ユニットとオンライン接続する (P.4-110)」、および「IPC Application Controller とオンライン接続する (P.4-111)」の手順を参考に、［オンライン］の代わりに［オフライン］を選択してください。

18 **ロボット統合 CPU ユニットおよび IPC Application Controller の電源を OFF にします。**

19 **ロボット統合 CPU ユニットおよび IPC Application Controller の電源を ON にします。**

20 **IPC Application Controller で ACE をサーバとして起動します。**
ACE サーバインスタンスについては、『オートメーション コントロール エンバイロメント (ACE)Version 4 オペレーションマニュアル(SBCE-456)』を参照してください。

21 **Sysmac Studio でロボット統合 CPU ユニットにオンライン接続します。**
オンラインの方法については、「ロボット統合 CPU ユニットとオンライン接続する (P.4-110)」を参照してください。

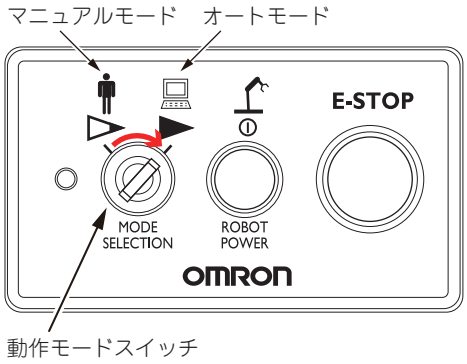
22 **[ウォッチウィンドウ（プロジェクト）1] 画面で「gStart」に TRUE を設定します。**



ウォッチウィンドウを開く手順については、「4-5-5 シーケンス制御プログラム、V+プログラムを実行する (P.4-126)」を参照してください。

23 **ロボットのフロントパネルでロボットのハイパワーを ON にします。**

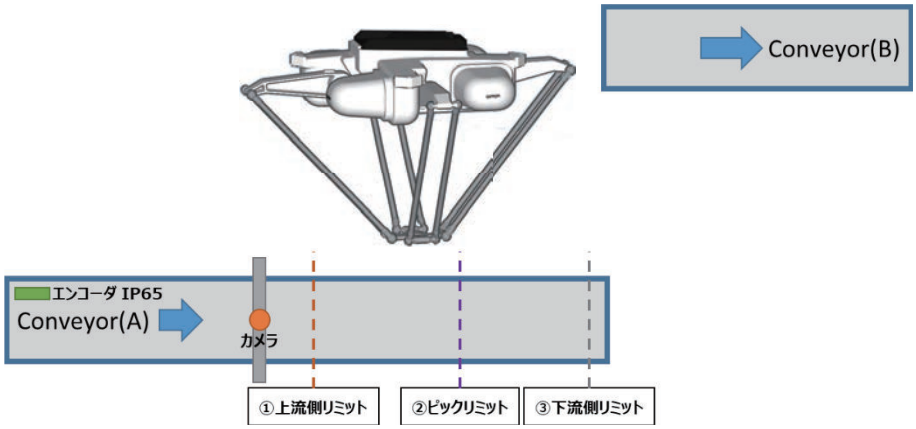
1) フロントパネルをオートモードにします。



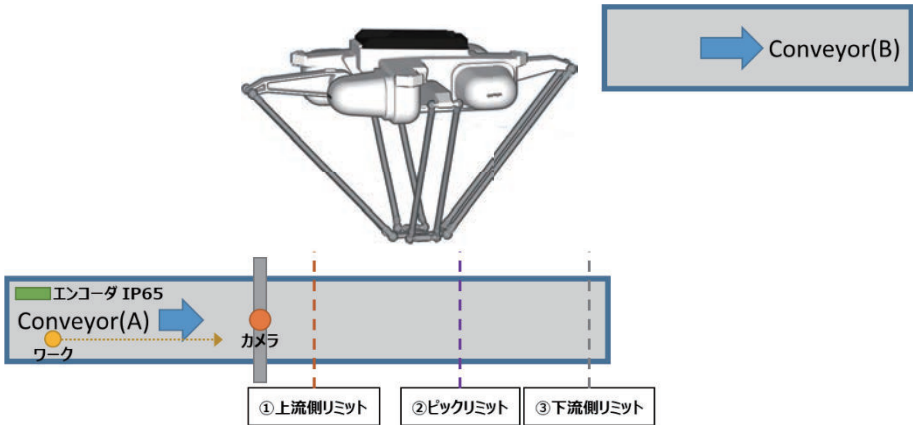
2) ROBOT POWER ボタンを押します。
ROBOT POWER ボタンが点滅します。

3) もう一度 ROBOT POWER ボタンを押します。

ロボットのハイパワーが ON になり、動的ピック&プレースのプログラムが自動で実行されます。
搬入用コンベア A が動作します。

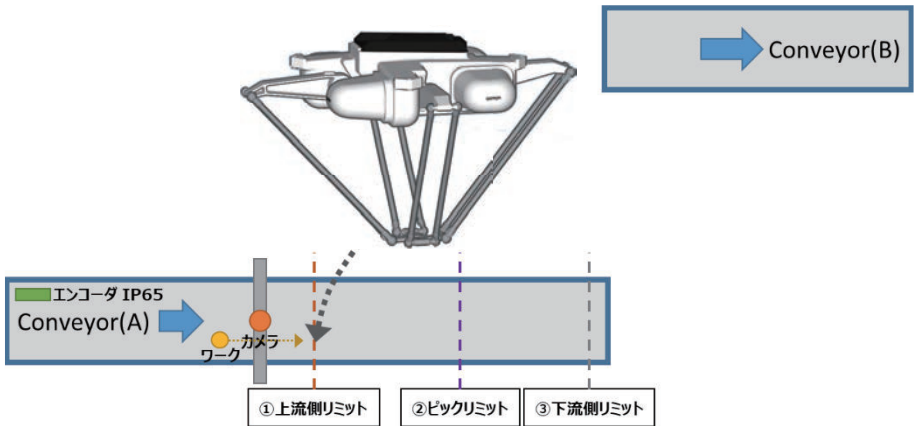


24 ワークをコンベア A の上流側からコンベア上に置きます。

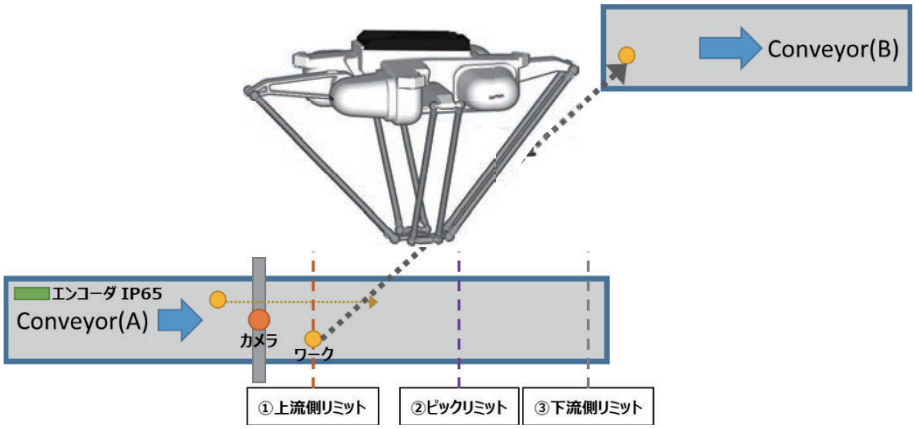


25 ロボットの動作が以下となるかを確認します。

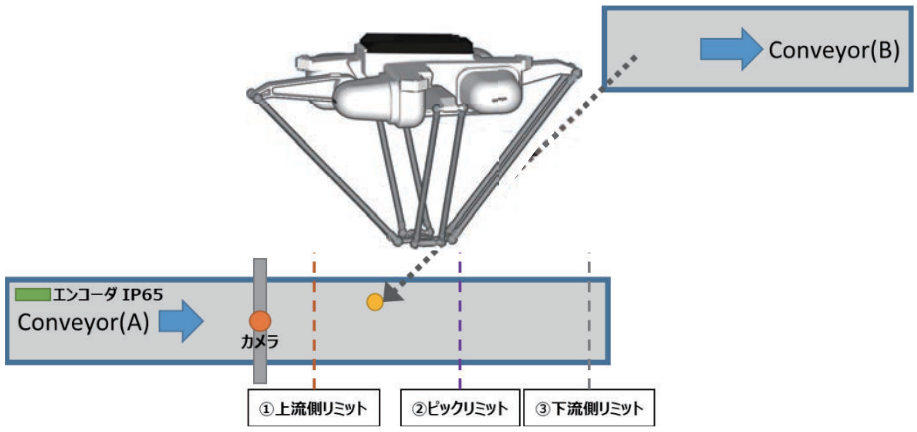
1）カメラがワークを検出したタイミングで、ロボットが上流リミット付近まで移動し、その後一時待機します。



2）ワークが上流側リミットまで搬送されたタイミングでロボットがワークをピックし、ピック&プレースの動作を行います。



26 **手順 24 から手順 25 を繰り返し、ロボットの動作が以下となるかを確認します。**
ロボットは、次に検出したワークをピック&プレースする動作を繰り返します。



A

付録

A-1　ピック & プレース装置の安全機能の設計例 A-2
A-1-1　ピック & プレース装置の構成 A-2
A-1-2　危険源 A-4
A-1-3　安全方策 A-4
A-1-4　安全機能 A-4
A-1-5　安全システム構成および使用機器 A-5
A-1-6　設置と配線 A-7
A-1-7　設定とプログラム A-13

A-2　3D 表示画面の操作方法 A-21
A-2-1　3D 表示画面をフローティング状態にする A-21
A-2-2　視点を回転する A-23
A-2-3　視点を平行移動する A-25
A-2-4　ズームインおよびズームアウト A-26

A-3　Sysmac Studio の設定項目と設定対象 A-28

A-4　トラブルシュート機能の操作方法 A-29

A-1　ピック & プレース装置の安全機能の設計例

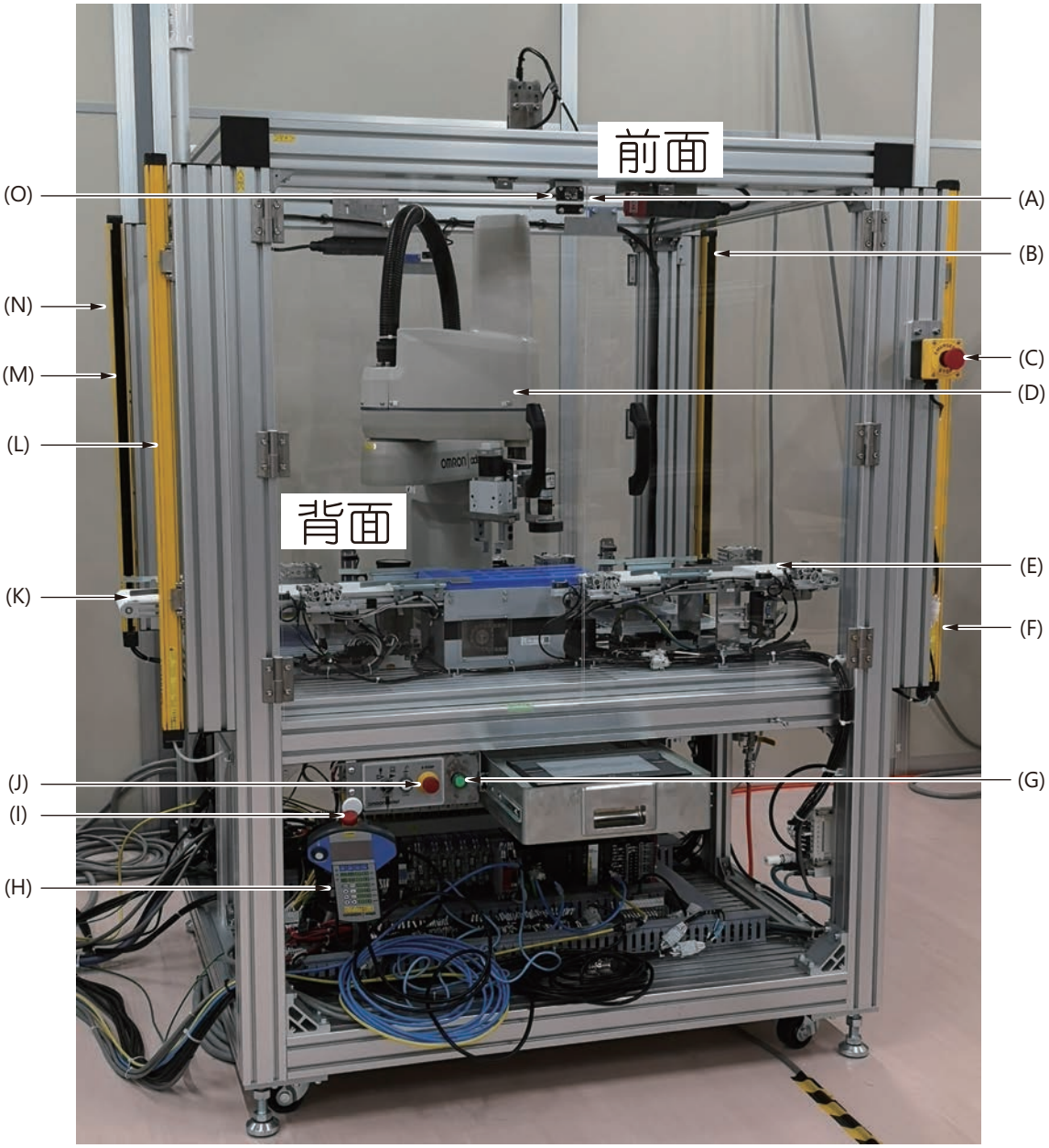
ここでは、ピック & プレース装置の安全機能の設計例について説明します。

使用上の注意

実際の安全対策は、『ロボット セーフティガイド(SBCE-381)』およびリスクアセスメントに基づいて実施してください。また、安全規格に適合する場合、パフォーマンスレベルや安全距離についても考慮してください。

A-1-1　ピック & プレース装置の構成

安全機器を含めたピック & プレース装置の構成を以下に示します。



記号	名称
A	セーフティドアスイッチ 2
B	セーフティライトカーテン 2（投光器）
C	非常停止用押しボタンスイッチ 1
D	ロボット
E	ベルトコンベア（搬出側）
F	セーフティライトカーテン 2（受光器）
G	リセットスイッチ
H	T20 ペンダントのイネーブルスイッチ
I	T20 ペンダントの E-STOP ボタン
J	フロントパネルの E-STOP ボタン
K	ベルトコンベア（搬入側）
L	セーフティライトカーテン 1（受光器）
M	セーフティライトカーテン 1（投光器）
N	非常停止用押しボタンスイッチ 2
O	セーフティドアスイッチ 1

A-1-2　危険源

ここで説明するシステム構成では、ロボットの可動部を機械的危険源とします。

A-1-3　安全方策

ピック & プレース装置の各面から危険源へ接近するケースを想定します。
それぞれのケースに対する安全防護を下記のように実施することとします。

装置の面	接近するケース	安全防護	安全機器
背面	---	固定式ガード	---
前面	調整／保守時	可動式ガード	セーフティドアスイッチ
搬入側	搬入時	光電式保護設備	セーフティライトカーテン
搬出側	搬出時	光電式保護設備	セーフティライトカーテン

また、安全防護に加え、付加保護方策として装置の各面からアクセス可能な位置に非常停止スイッチを設置します。

装置の面	付加保護方策	安全機器
背面	非常停止スイッチ	非常停止用押しボタンスイッチ
前面		非常停止用押しボタンスイッチ
		T20 ペンダントの E-STOP ボタン
		フロントパネルの E-STOP ボタン

A-1-4　安全機能

各安全機器が作動した場合の安全機能を決定します。
オムロン製ロボットの動作モードごとに決定する必要があります。

オートモード

オムロン製ロボットの動作モードがオートモードの場合の安全機能を以下のとおりとします。

安全機器	安全機能	復旧方法
非常停止用押しボタンスイッチ 1	押下時にロボットを停止	リセット後にハイパワーを ON
非常停止用押しボタンスイッチ 2	押下時にロボットを停止	リセット後にハイパワーを ON
セーフティライトカーテン 1	遮光時にロボットを停止	ハイパワーを ON
セーフティライトカーテン 2	遮光時にロボットを停止	ハイパワーを ON
セーフティドアスイッチ 1	ドア開時にロボットを停止	リセット後にハイパワーを ON
セーフティドアスイッチ 2	ドア開時にロボットを停止	リセット後にハイパワーを ON
フロントパネルの E-STOP ボタン	押下時にロボットを停止	リセット後にハイパワーを ON
T20 ペンダントの E-STOP ボタン	押下時にロボットを停止	リセット後にハイパワーを ON
T20 ペンダントのイネーブルスイッチ	---	---

マニュアルモード

オムロン製ロボットの動作モードがマニュアルモードの場合の安全機能を以下のとおりとします。

安全機器	安全機能	復旧方法
非常停止用押しボタンスイッチ 1	押下時にロボットを停止	リセット
非常停止用押しボタンスイッチ 2	押下時にロボットを停止	リセット
セーフティライトカーテン 1	---	---

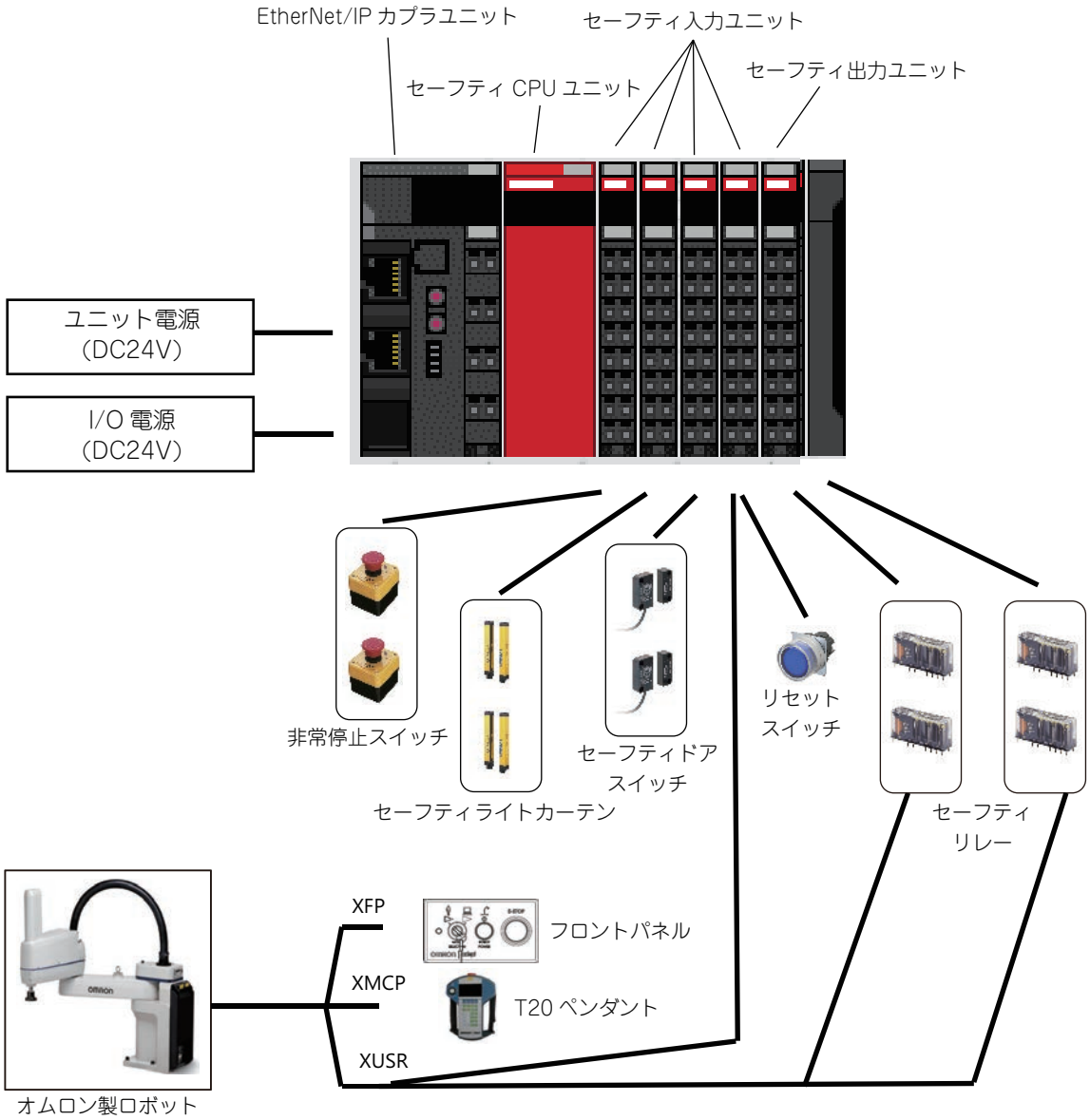
安全機器	安全機能	復旧方法
セーフティライトカーテン 2	---	---
セーフティドアスイッチ 1	---	---
セーフティドアスイッチ 2	---	---
フロントパネルの E-STOP ボタン	押下時にロボットを停止	リセット
T20 ペンダントの E-STOP ボタン	押下時にロボットを停止	リセット
T20 ペンダントのイネーブルスイッチ	非押下、または中間位置を過ぎて押下時にロボットを停止	ハイパワーを ON

A-1-5　安全システム構成および使用機器

安全機能を実現するための安全システム構成および使用機器を選定します。

安全システム構成

本ガイドでは、NX シリーズ EtherNet/IP カプラユニットおよび NX シリーズセーフティコントロールユニットを使用した安全システム構成とします。



使用機器

本ガイドの安全システム構成では、以下の機器を使用します。

名称	形式	メーカ	概要
EtherNet/IP カプラユニット	形 NX-EIC202	オムロン株式会社	EtherNet/IP 対応のカプラユニットです。 NX ユニットを接続することでスレーブターミナルを構成できます。
セーフティ CPU ユニット	形 NX-SL3300	オムロン株式会社	セーフティプログラムを実行するユニットです。
セーフティ入力ユニット	形 NX-SIH400	オムロン株式会社	セーフティ入力機器と接続するユニットです。
セーフティ出力ユニット	形 NX-SOD400	オムロン株式会社	セーフティ出力機器と接続するユニットです。
セーフティライトカーテン	形 F3SG	オムロン株式会社	光電式のセーフティセンサです。危険領域へ人が侵入したことを検知します。
セーフティドアスイッチ	形 D40A	オムロン株式会社	非接触式のセーフティドアスイッチです。危険領域に出入りするための扉の開閉を検知します。
リセットスイッチ	形 A22-H□-10M	オムロン株式会社	リセットスイッチです。安全機能が作動して停止した状態からの手動復旧に使用します。
非常停止用押しボタンスイッチ	形 A22E	オムロン株式会社	非常停止スイッチです。
セーフティリレー	形 G7SA	オムロン株式会社	セーフティリレーです。
ユニット電源	形 S8VK	オムロン株式会社	DC24V 電源です。EtherNet/IP カプラユニットおよび NX ユニットの制御電源です。
I/O 電源	形 S8VK	オムロン株式会社	DC24V 電源です。NX ユニットの入出力回路および外部接続機器に供給する電源です。

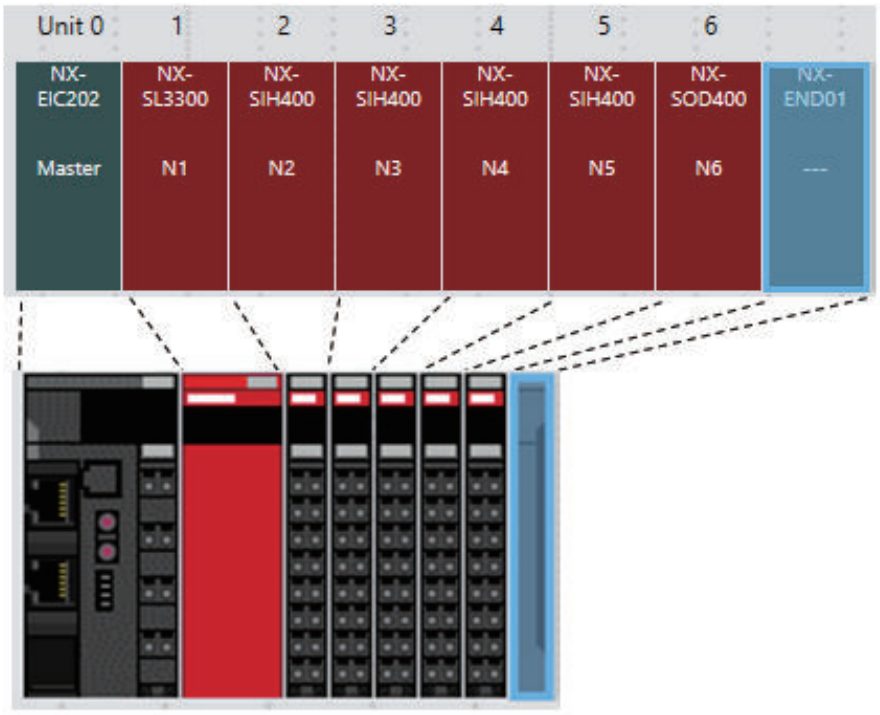
A-1-6　設置と配線

ピック & プレース装置の安全機能に関する、設置と配線について説明します。
本ガイドでは、EtherNet/IP スレーブターミナルの NX ユニット構成、および安全機器の配線図を説明しています。
実際の設置方法や配線方法については、下記のマニュアルを参照してください。

- NX シリーズ セーフティコントロールユニット ユーザーズマニュアル(SGFM-710)
- NX シリーズ EtherNet/IP カプラユニット ユーザーズマニュアル(SBCD-365)
- スカラロボット eCobra 600/800 EtherCAT タイプ ユーザーズマニュアル(SBCE-489)
- ティーチングペンダント T20 ユーザーズマニュアル(SBCE-392)

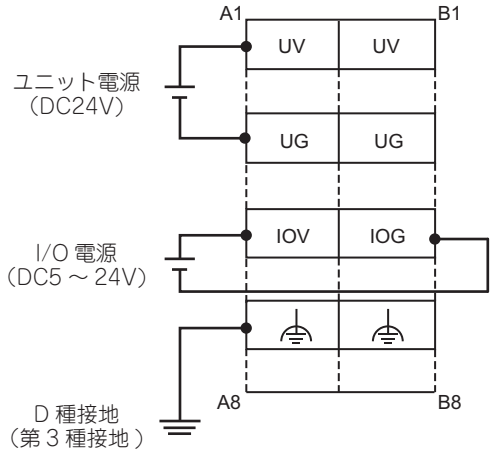
NX ユニット構成

下図に示す接続順序で、EtherNet/IP スレーブターミナルを構成します。



電源の配線

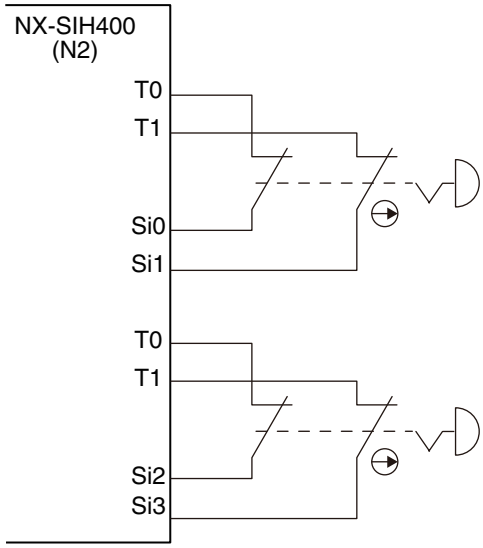
EtherNet/IP カプラユニットの電源、接地の配線を以下に示します。



- ユニット電源の＋極（DC24V）を A1 あるいは B1 のどちらか一方に接続します。
- ユニット電源の－極（DC0V）を A3 あるいは B3 のどちらか一方に接続します。
- I/O 電源の＋極（DC5 ～ 24V）を A5 に接続します。
- I/O 電源の－極（DC0V）を B5 に接続します。
- 接地線を、A7 あるいは B7 のいずれか一方に接続します。

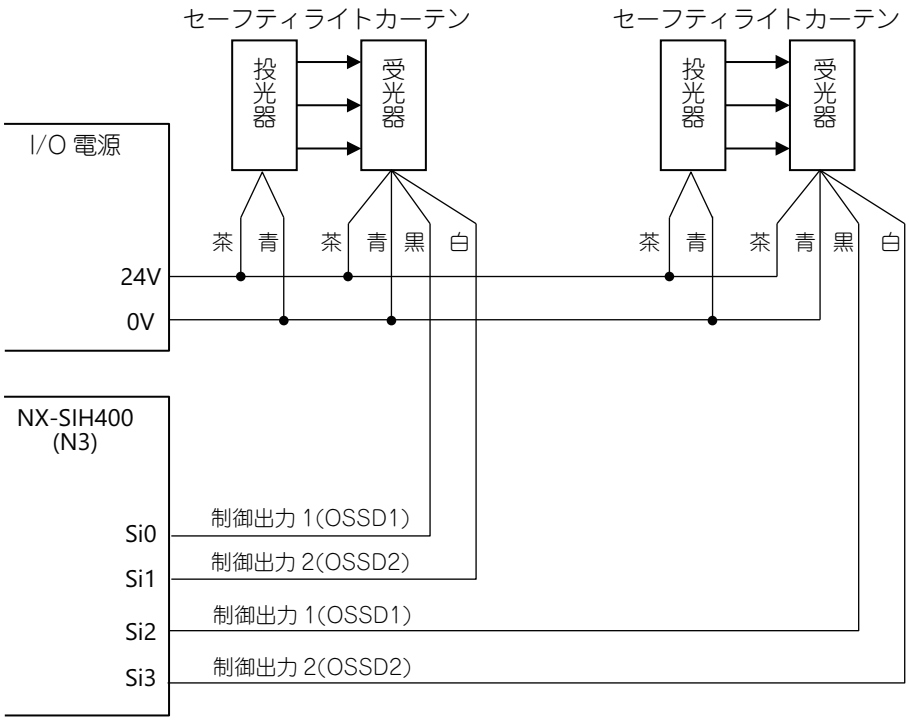
非常停止用押しボタンスイッチの配線

非常停止用押しボタンスイッチをセーフティ入力ユニット(N2)に接続します。



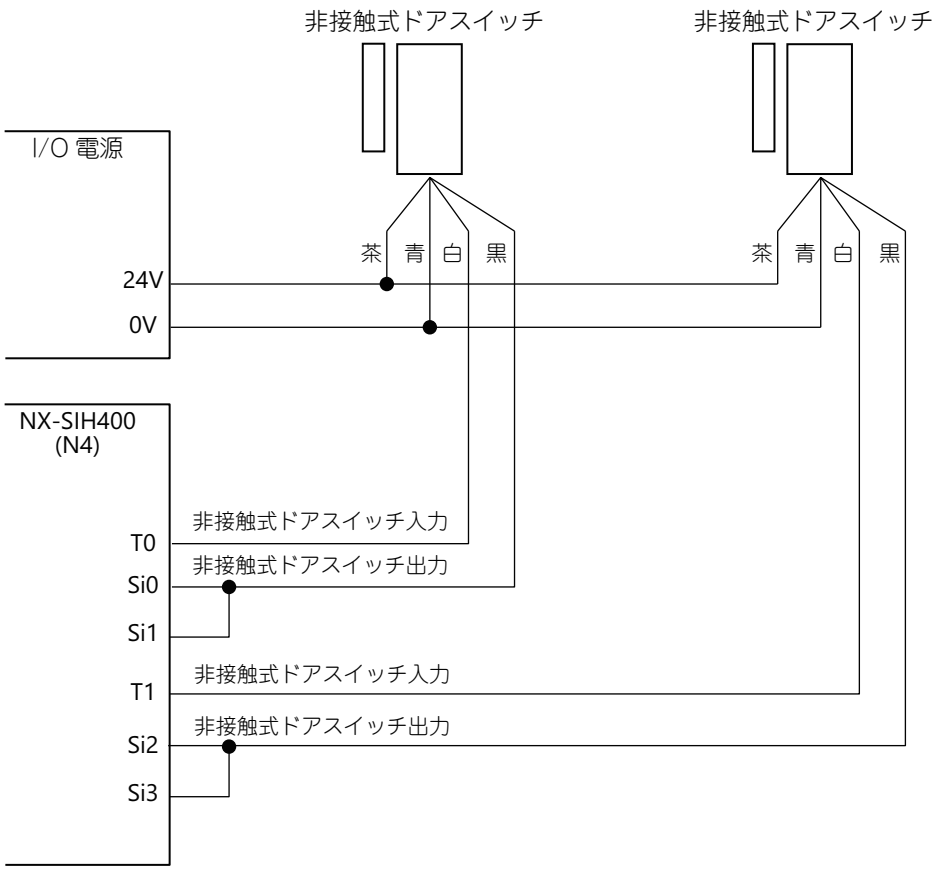
セーフティライトカーテンの配線

セーフティライトカーテンを I/O 電源とセーフティ入力ユニット(N3)に接続します。



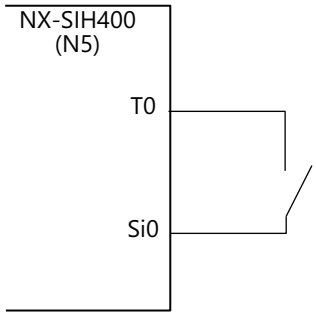
セーフティドアスイッチの配線

非接触式ドアスイッチを I/O 電源とセーフティ入力ユニット(N4)に接続します。



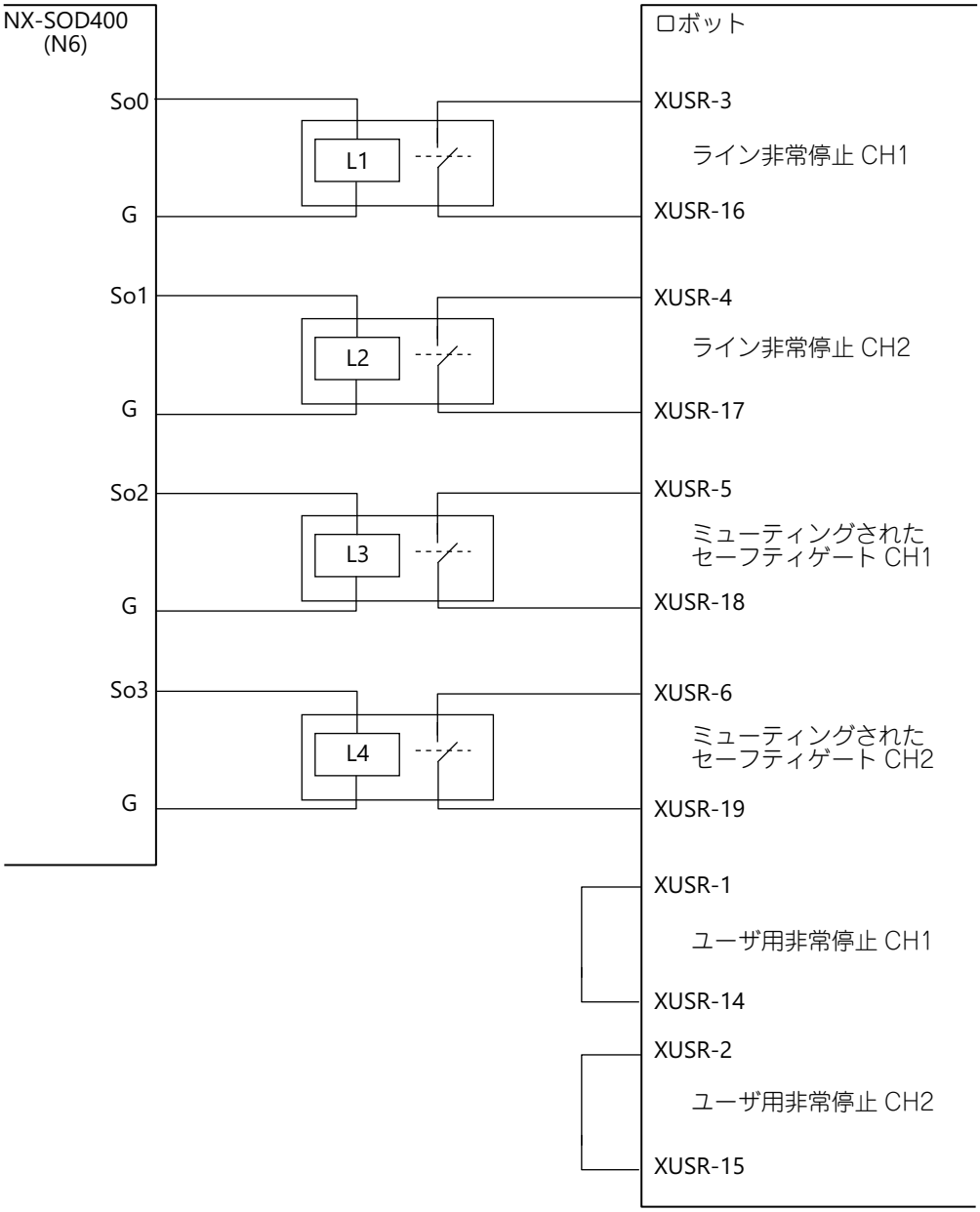
リセットスイッチの配線

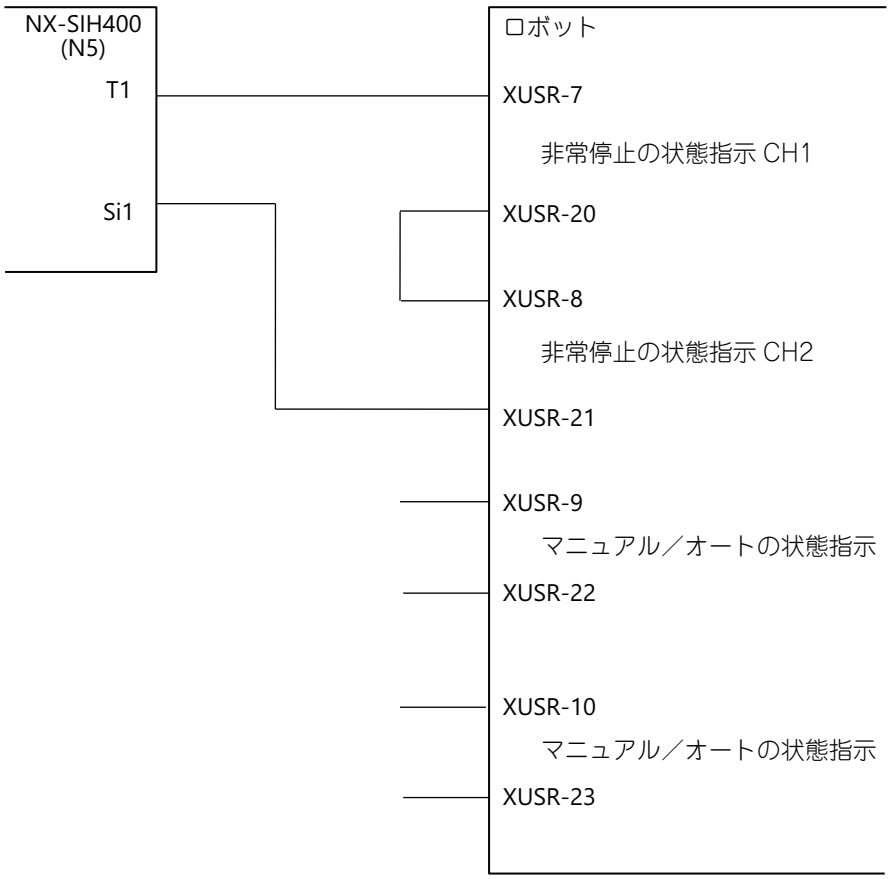
リセットスイッチをセーフティ入力ユニット(N5)に接続します。



ロボットの配線

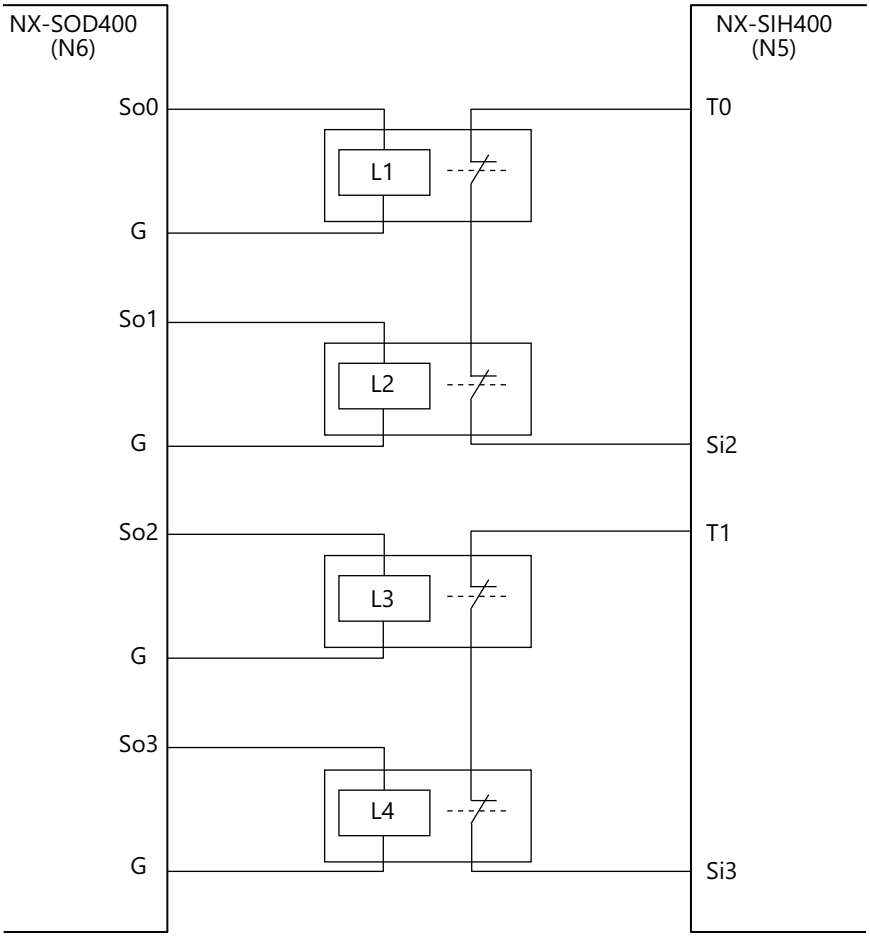
ロボットをセーフティ出力ユニット(N6)、セーフティ入力ユニット(N5)、およびセーフティリレーに接続します。
セーフティ出力ユニット(N6)の出力とロボットの入力は直接接続できないため、セーフティリレーを介して接続します。
セーフティリレー L1～L4 の a 接点側を使用します。


NX-SOD400
(N6)
So0
G
So1
G
So2
G
So3
G
L1
L2
L3
L4
ロボット
XUSR-3
ライン非常停止 CH1
XUSR-16
XUSR-4
ライン非常停止 CH2
XUSR-17
XUSR-5
ミューティングされた
セーフティゲート CH1
XUSR-18
XUSR-6
ミューティングされた
セーフティゲート CH2
XUSR-19
XUSR-1
ユーザ用非常停止 CH1
XUSR-14
XUSR-2
ユーザ用非常停止 CH2
XUSR-15



セーフティリレーの動作確認用配線

セーフティリレー L1～L4 の a 接点側の動作確認用として、b 接点側をセーフティ入力ユニット(N5)に接続します。



A-1-7　設定とプログラム

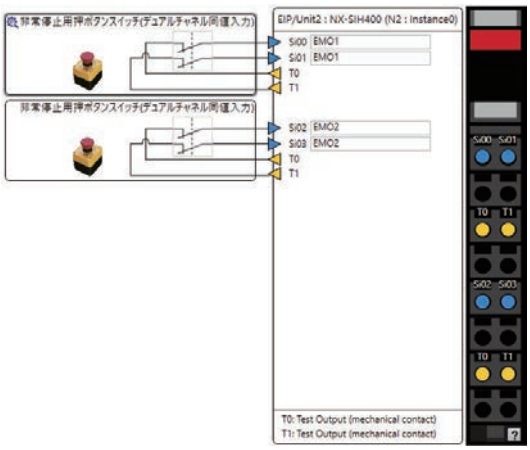
ここでは、安全機能に関する設定とプログラムについて説明します。

セーフティ CPU ユニットの命令については、『NX シリーズ セーフティコントロールユニット コマンドリファレンスマニュアル(SGFM-711)』を参照してください。
セーフティコントロールユニットの仕様や設定方法については、『NX シリーズ セーフティコントロールユニット ユーザーズマニュアル(SGFM-710)』を参照してください。

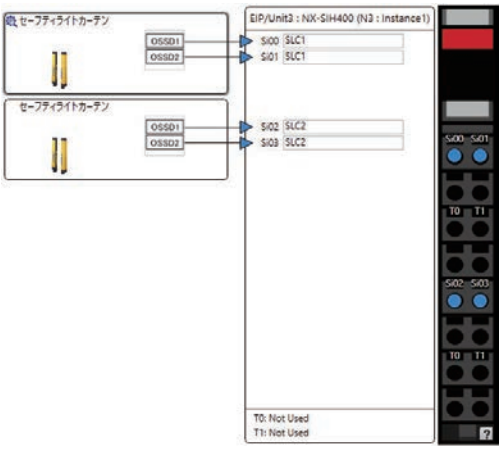
セーフティ入出力ユニットの設定

Sysmac Studio を使用して、セーフティ入出力端子のセーフティ機能を設定します。

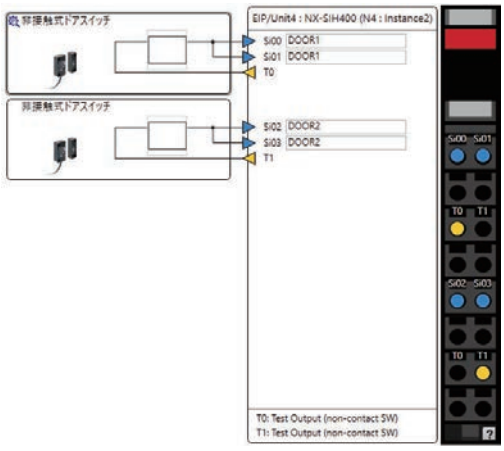
セーフティ入力ユニット(N2)を次のように設定します。



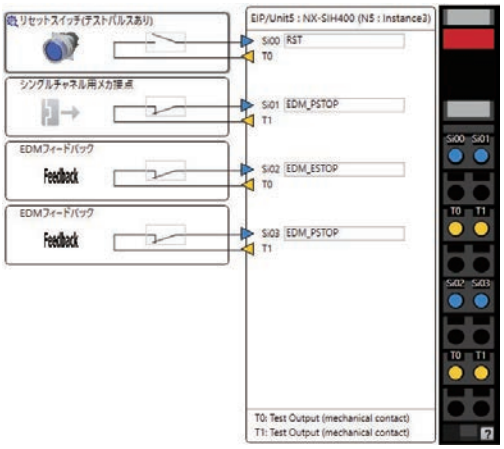
セーフティ入力ユニット(N3)を次のように設定します。



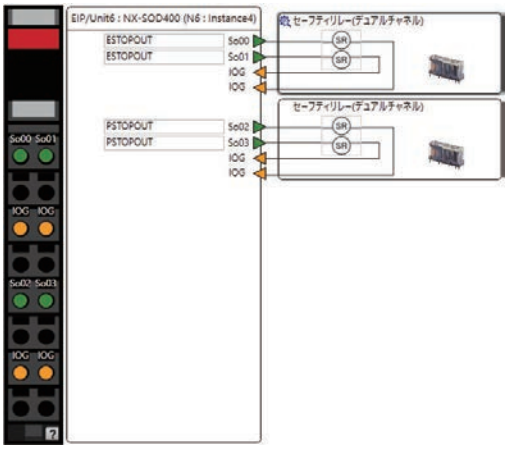
セーフティ入力ユニット(N4)を次のように設定します。



セーフティ入力ユニット(N5)を次のように設定します。



セーフティ出力ユニット(N6)を次のように設定します。



デバイス変数の登録

Sysmac Studio を使用して、セーフティ入出力ユニットのデバイス変数を次のように登録します。

位置	ポート	R/W	データ型	変数	変数コメント	変数種別
	▼ NXバス					
NXバスマスタ	マスタ					
EIP/Unit2	▼ NX-SIH400					
	▼ Safety Inputs and Status					
	Si00 Logical Value	R	SAFEBOOL	EMO1	EMO1	グローバル変数
	Si01 Logical Value	R	SAFEBOOL		EMO1	
	Si02 Logical Value	R	SAFEBOOL	EMO2	EMO2	グローバル変数
	Si03 Logical Value	R	SAFEBOOL		EMO2	
	Safety Connection Status	R	SAFEBOOL	N2_Safety_Connection_Status		グローバル変数
	Safety Input Terminal Status	R	SAFEBOOL			
EIP/Unit3	▼ NX-SIH400					
	▼ Safety Inputs and Status					
	Si00 Logical Value	R	SAFEBOOL	SLC1	SLC1	グローバル変数
	Si01 Logical Value	R	SAFEBOOL		SLC1	
	Si02 Logical Value	R	SAFEBOOL	SLC2	SLC2	グローバル変数
	Si03 Logical Value	R	SAFEBOOL		SLC2	
	Safety Connection Status	R	SAFEBOOL	N3_Safety_Connection_Status		グローバル変数
	Safety Input Terminal Status	R	SAFEBOOL			
EIP/Unit4	▼ NX-SIH400					
	▼ Safety Inputs and Status					
	Si00 Logical Value	R	SAFEBOOL	DOOR1	DOOR1	グローバル変数
	Si01 Logical Value	R	SAFEBOOL		DOOR1	
	Si02 Logical Value	R	SAFEBOOL	DOOR2	DOOR2	グローバル変数
	Si03 Logical Value	R	SAFEBOOL		DOOR2	
	Safety Connection Status	R	SAFEBOOL	N4_Safety_Connection_Status		グローバル変数
	Safety Input Terminal Status	R	SAFEBOOL			
EIP/Unit5	▼ NX-SIH400					
	▼ Safety Inputs and Status					
	Si00 Logical Value	R	SAFEBOOL	RST	RST	グローバル変数
	Si01 Logical Value	R	SAFEBOOL	LOCALSTOP	LOCALSTOP	グローバル変数
	Si02 Logical Value	R	SAFEBOOL	EDM_ESTOP	EDM_ESTOP	グローバル変数
	Si03 Logical Value	R	SAFEBOOL	EDM_PSTOP	EDM_PSTOP	グローバル変数
	Safety Connection Status	R	SAFEBOOL	N5_Safety_Connection_Status		グローバル変数
	Safety Input Terminal Status	R	SAFEBOOL			
EIP/Unit6	▼ NX-SOD400					
	▼ Status					
	Safety Connection Status	R	SAFEBOOL	N6_Safety_Connection_Status		グローバル変数
	Safety Output Terminal Status	R	SAFEBOOL			
	▼ Safety Outputs					
	So00 Output Value	W	SAFEBOOL	ESTOPOUT	ESTOPOUT	グローバル変数
	So01 Output Value	W	SAFEBOOL		ESTOPOUT	
	So02 Output Value	W	SAFEBOOL	PSTOPOUT	PSTOPOUT	グローバル変数
	So03 Output Value	W	SAFEBOOL		PSTOPOUT	

セーフティプログラム

Sysmac Studio を使用して、変数とプログラムを次のように作成します。

次のように変数を作成します。

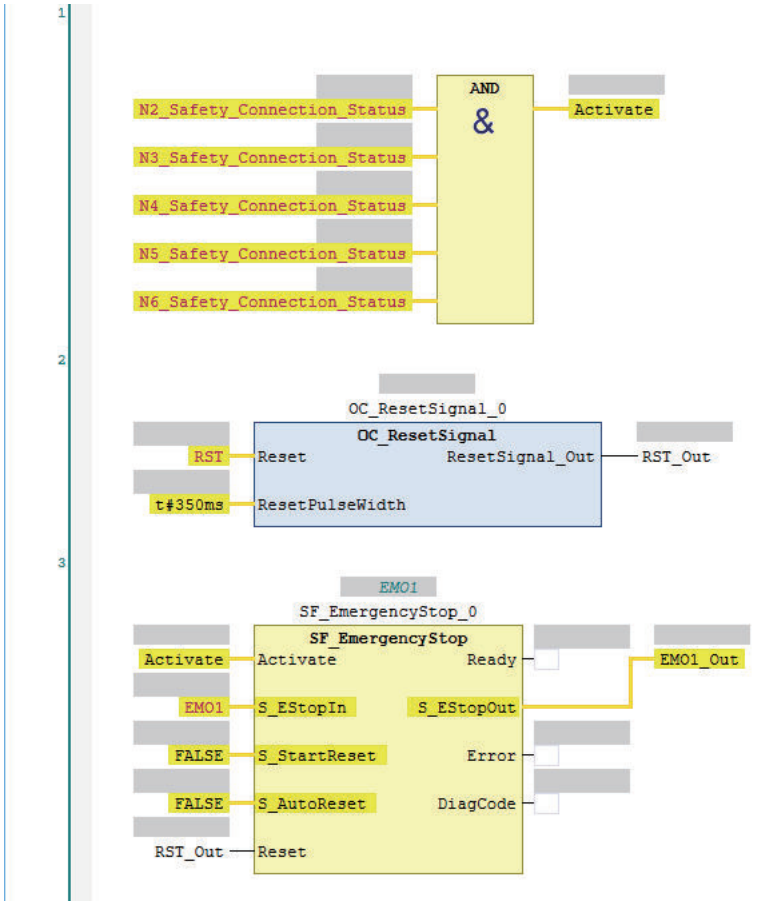
内部変数
外部変数

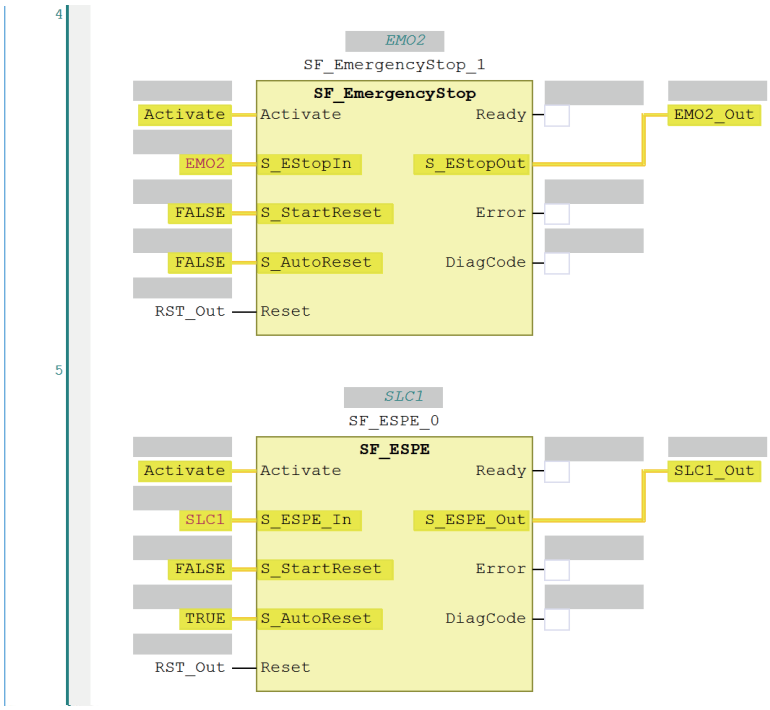
名称	データ型	初期値	コンスタント	コメント
OC_ResetSignal_0	OC_ResetSignal		☐	
RST_Out	BOOL	FALSE	☐	
SF_EmergencyStop_0	SF_EmergencyStop		☐	EMO1
EMO1_Out	SAFEBOOL	FALSE	☐	
SF_EmergencyStop_1	SF_EmergencyStop		☐	EMO2
EMO2_Out	SAFEBOOL	FALSE	☐	
SF_ESPE_0	SF_ESPE		☐	SLC1
SLC1_Out	SAFEBOOL	FALSE	☐	
SF_ESPE_1	SF_ESPE		☐	SLC2
SLC2_Out	SAFEBOOL	FALSE	☐	
SF_GuardMonitoring_0	SF_GuardMonitoring		☐	DOOR1
DOOR1_Out	SAFEBOOL	FALSE	☐	
SF_GuardMonitoring_1	SF_GuardMonitoring		☐	DOOR2
DOOR2_Out	SAFEBOOL	FALSE	☐	
SF_EmergencyStop_2	SF_EmergencyStop		☐	LOCALSTOP
LOCALSTOP_Out	SAFEBOOL	FALSE	☐	
ESTOPOUT_In	SAFEBOOL	FALSE	☐	
PSTOPOUT_In	SAFEBOOL	FALSE	☐	
SF_EDM_0	SF_EDM		☐	ESTOPOUT
SF_EDM_1	SF_EDM		☐	PSTOPOUT
Activate	SAFEBOOL	FALSE	☐	

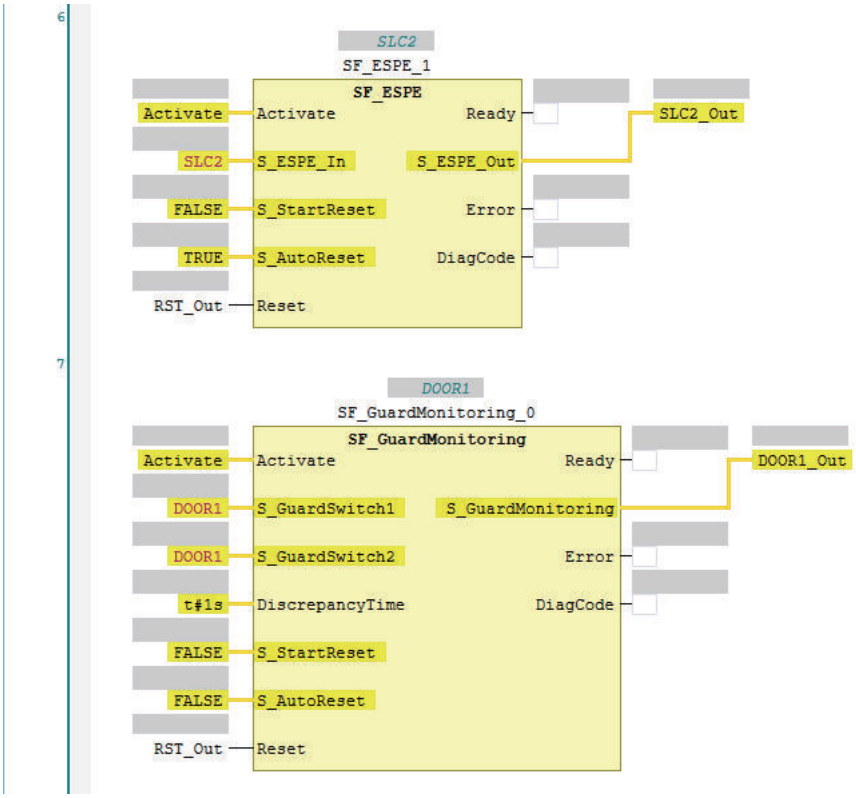
内部変数
外部変数

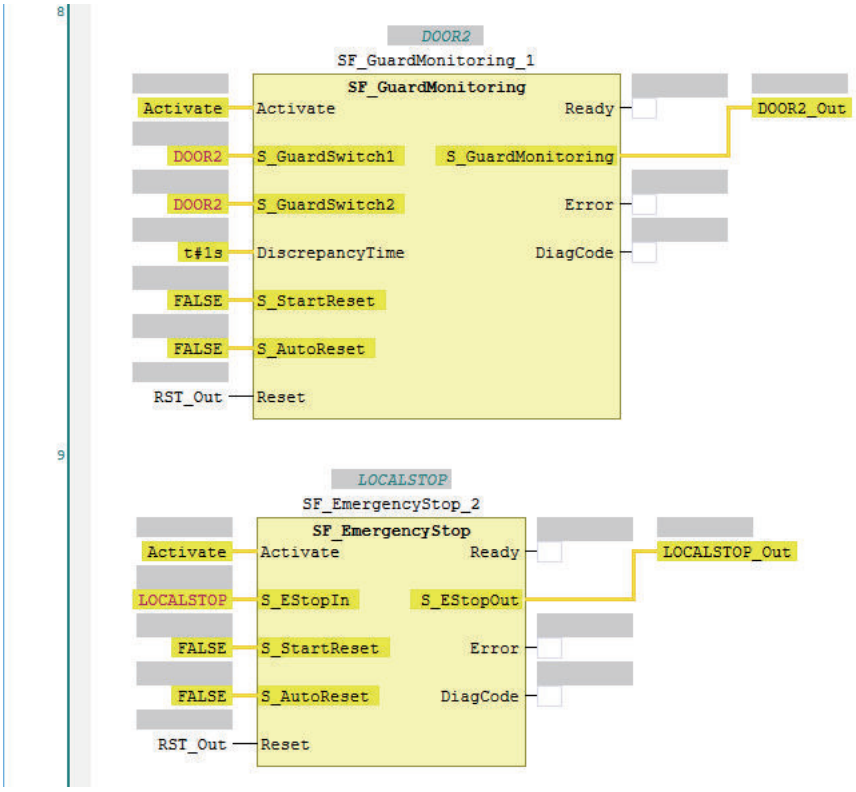
名称	データ型	コンスタント	コメント
RST	SAFEBOOL	☐	
EMO1	SAFEBOOL	☐	
EMO2	SAFEBOOL	☐	
SLC1	SAFEBOOL	☐	
SLC2	SAFEBOOL	☐	
DOOR1	SAFEBOOL	☐	
DOOR2	SAFEBOOL	☐	
LOCALSTOP	SAFEBOOL	☐	
ESTOPOUT	SAFEBOOL	☐	
PSTOPOUT	SAFEBOOL	☐	
N2_Safety_Connection_Status	SAFEBOOL	☐	
N3_Safety_Connection_Status	SAFEBOOL	☐	
N4_Safety_Connection_Status	SAFEBOOL	☐	
N5_Safety_Connection_Status	SAFEBOOL	☐	
N6_Safety_Connection_Status	SAFEBOOL	☐	
EDM_ESTOP	SAFEBOOL	☐	
EDM_PSTOP	SAFEBOOL	☐	

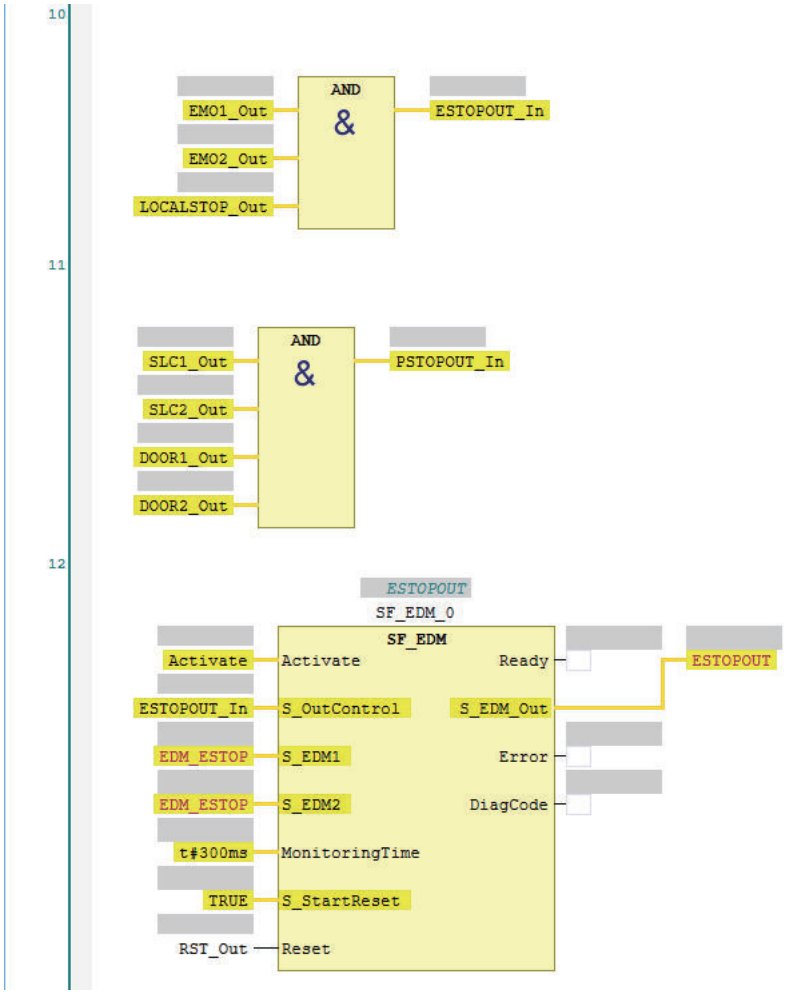
次のようにプログラムを作成します。

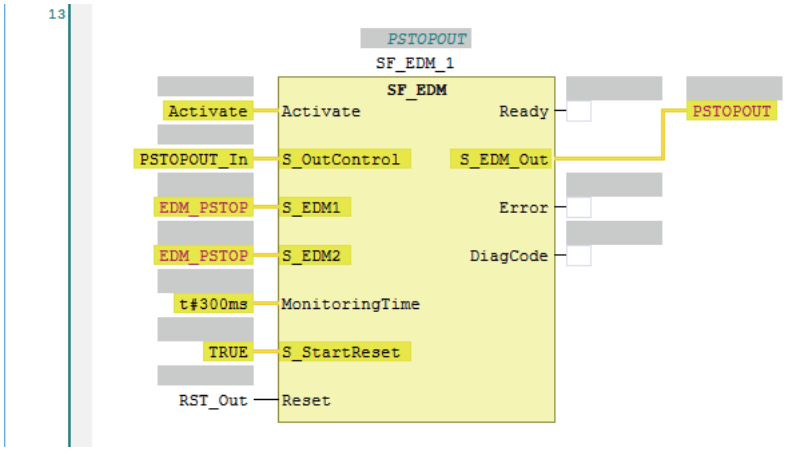

1
AND
&
N2_Safety_Connection_Status
N3_Safety_Connection_Status
N4_Safety_Connection_Status
N5_Safety_Connection_Status
N6_Safety_Connection_Status
Activate
2
OC_ResetSignal_0
OC_ResetSignal
RST
Reset
ResetSignal_Out
RST_Out
t#350ms
ResetPulseWidth
3
EMO1
SF_EmergencyStop_0
SF_EmergencyStop
Activate
Activate
Ready
EMO1_Out
EMO1
S_EStopIn
S_EStopOut
FALSE
S_StartReset
Error
FALSE
S_AutoReset
DiagCode
RST_Out
Reset


4
EMO2
SF_EmergencyStop_1
SF_EmergencyStop
Activate
Activate
Ready
EMO2_Out
EMO2
S_EStopIn
S_EStopOut
FALSE
S_StartReset
Error
FALSE
S_AutoReset
DiagCode
RST_Out
Reset
5
SLC1
SF_ESPE_0
SF_ESPE
Activate
Activate
Ready
SLC1_Out
SLC1
S_ESPE_In
S_ESPE_Out
FALSE
S_StartReset
Error
TRUE
S_AutoReset
DiagCode
RST_Out
Reset


6
SLC2
SF_ESPE_1
SF_ESPE
Activate
Activate
Ready
SLC2_Out
SLC2
S_ESPE_In
S_ESPE_Out
FALSE
S_StartReset
Error
TRUE
S_AutoReset
DiagCode
RST_Out
Reset
7
DOOR1
SF_GuardMonitoring_0
SF_GuardMonitoring
Activate
Activate
Ready
DOOR1_Out
DOOR1
S_GuardSwitch1
S_GuardMonitoring
DOOR1
S_GuardSwitch2
Error
t#1s
DiscrepancyTime
DiagCode
FALSE
S_StartReset
FALSE
S_AutoReset
RST_Out
Reset


8
DOOR2
SF_GuardMonitoring_1
SF_GuardMonitoring
Activate
Activate
Ready
DOOR2_Out
DOOR2
S_GuardSwitch1
S_GuardMonitoring
DOOR2
S_GuardSwitch2
Error
t#1s
DiscrepancyTime
DiagCode
FALSE
S_StartReset
FALSE
S_AutoReset
RST_Out
Reset
9
LOCALSTOP
SF_EmergencyStop_2
SF_EmergencyStop
Activate
Activate
Ready
LOCALSTOP_Out
LOCALSTOP
S_EStopIn
S_EStopOut
FALSE
S_StartReset
Error
FALSE
S_AutoReset
DiagCode
RST_Out
Reset


10
AND
&
EMO1_Out
EMO2_Out
LOCALSTOP_Out
ESTOPOUT_In
11
AND
&
SLC1_Out
SLC2_Out
DOOR1_Out
DOOR2_Out
PSTOPOUT_In
12
ESTOPOUT
SF_EDM_0
SF_EDM
Activate
Activate
ESTOPOUT_In
S_OutControl
EDM_ESTOP
S_EDM1
EDM_ESTOP
S_EDM2
t#300ms
MonitoringTime
TRUE
S_StartReset
RST_Out
Reset
Ready
S_EDM_Out
Error
DiagCode
ESTOPOUT


13
PSTOPOUT
SF_EDM_1
SF_EDM
Activate
Activate
PSTOPOUT_In
S_OutControl
EDM_PSTOP
S_EDM1
EDM_PSTOP
S_EDM2
t#300ms
MonitoringTime
TRUE
S_StartReset
RST_Out
Reset
Ready
S_EDM_Out
Error
DiagCode
PSTOPOUT

A-2　3D 表示画面の操作方法

ここでは、ロボットシミュレーション上で視点を操作するための各種操作方法について説明します。
Sysmac Studio の 3D 表示画面で、操作を行います。

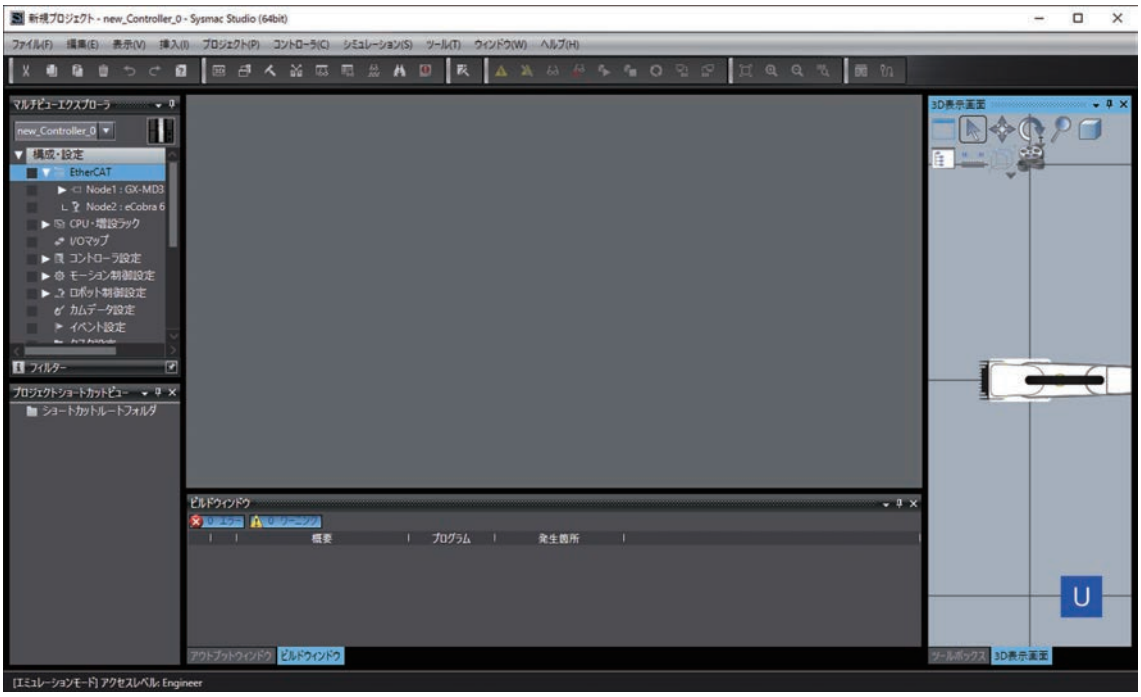
A-2-1　3D 表示画面をフローティング状態にする

参考

関連動画

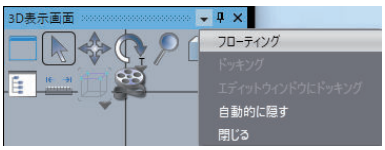
- 3D 表示画面のフローティング
  https://www.fa.omron.co.jp/video/detail/6294915157001/
  3D 表示画面をフローティング状態にする操作を説明しています。

Sysmac Studo の 3D 表示画面は、デフォルトではドック状態で表示されます。

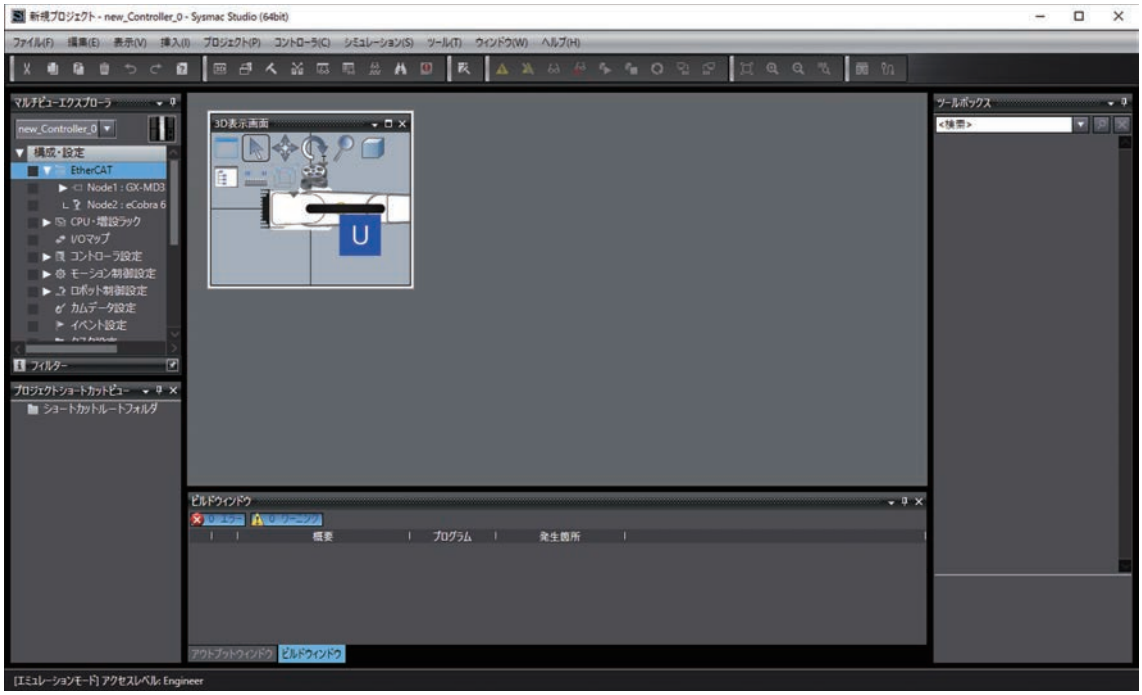


3D 表示画面を拡大表示したい場合、フローティング状態で表示し、画面サイズを調整します。

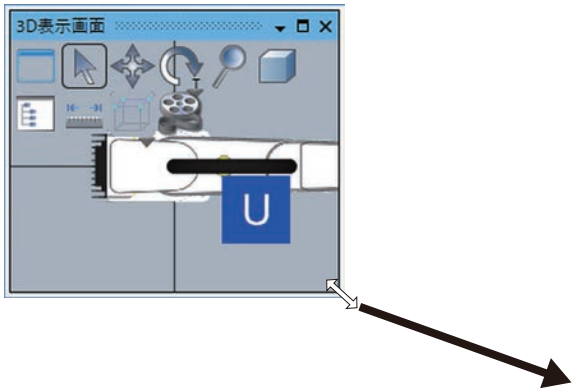
1　**3D 表示画面のタブを右クリックし、表示されたメニューから［フローティング］を選択します。**



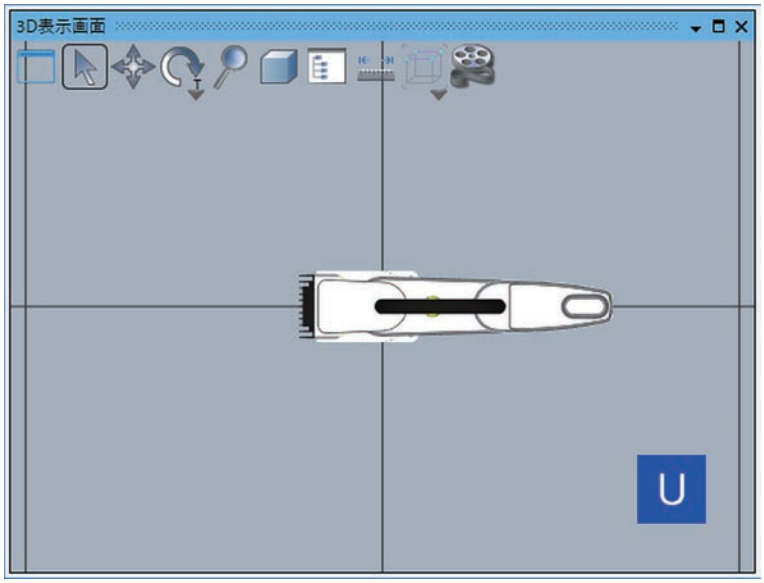
3D 表示画面がフローティング状態となります。



2 **3D 表示画面の枠をクリックしたまま、拡大したいサイズまでドラッグします。**



3D 表示画面が拡大されます。



A-2-2　視点を回転する

視点を回転させるには、以下の 2 とおりの方法があります。
- 3D ビュー切替ツールを使用する方法
- ［回転］アイコンを使用する方法

ここでは、3D 表示画面で視点を回転する 2 つの方法について説明します。

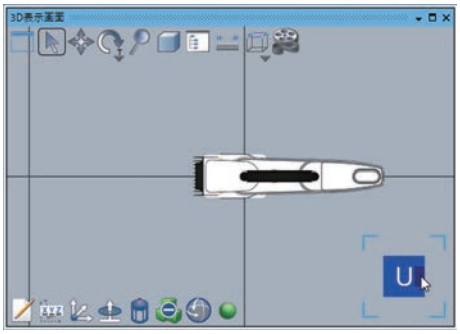
参考

関連動画
- 視点の回転
  https://www.fa.omron.co.jp/video/detail/6294915298001/
  3D 表示画面の視点を回転する操作を説明しています。

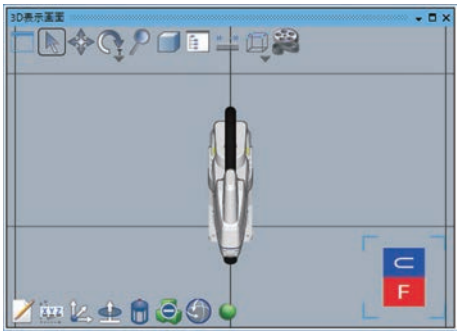
3D ビュー切替ツールを使用する方法

画面右下の 3D ビュー切替ツールを使用して視点を切り替えることがでできます。
3D ビュー切替ツールは、フェース、コーナ、エッジの 3 つの要素で構成されます。
以下に例を挙げて説明します。

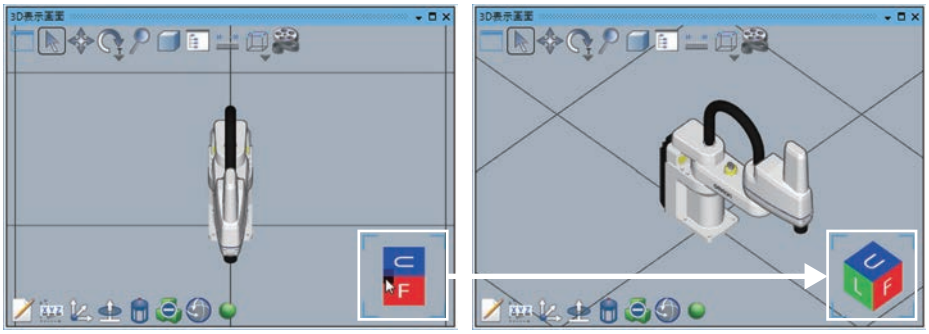
1　**［U］の右エッジにマウスポインタをあわせると黒く表示されます。**



2　**そのままクリックするとクリックした部分が正面になるように表示が切り替わります。**



同様に、左コーナをクリックすると以下のようになります。



［回転］アイコンを使用する方法

3D 表示画面の［回転］アイコンを使用して視点を回転する操作について説明します。

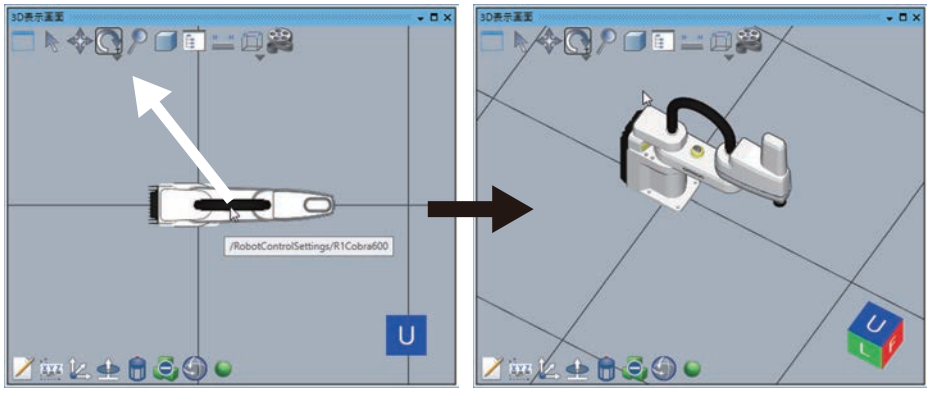
1 3D 表示画面の［回転］アイコンをクリックします。



ポインタが 3D 表示画面の視点の位置の回転を行う状態を示すアイコンになります。



2 マウスの左ボタンを押したまま回転させたい方向にドラッグします。



上方向のドラッグ: 視点の位置を下方向に移動
下方向のドラッグ: 視点の位置を上方向に移動
右方向のドラッグ: 視点の位置を左方向に移動
左方向のドラッグ: 視点の位置を右方向に移動

回転には、次の 2 つのモードがあります。

モード	説明
タンブラーモード	任意の角度から 3D 形状データを見ることができます。

モード	説明
ターンテーブルモード	ワールド座標系の Z 軸を中心に左右に回転します。また、上下は±90°の範囲で見ることができます。

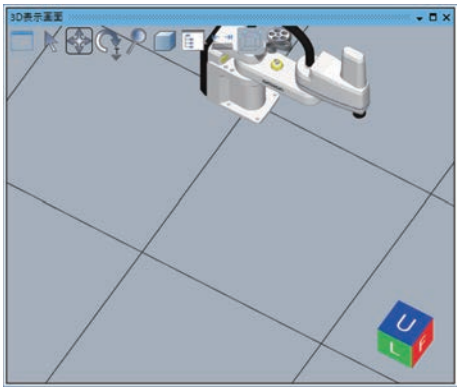
A-2-3　視点を平行移動する

3D 表示画面で視点を移動する操作について説明します。

参考

関連動画
- 視点の平行移動
  https://www.fa.omron.co.jp/video/detail/6294915158001/
  3D 表示画面の視点を平行移動する操作を説明しています。

ロボットを視野に入るように、下方向に移動させます。



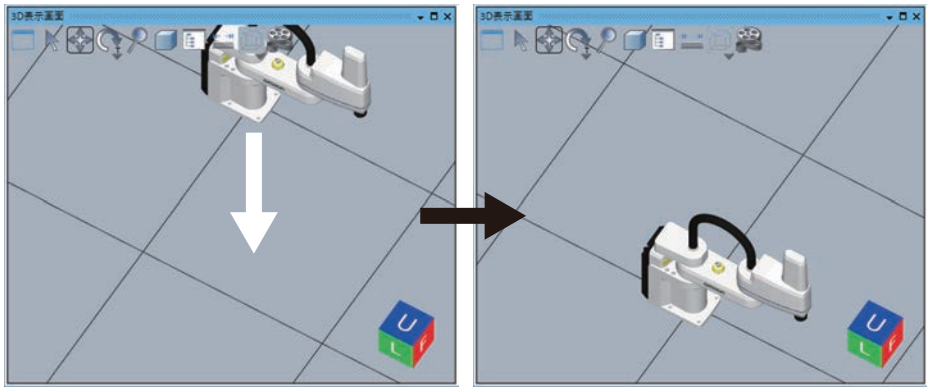
1　**3D 表示画面の［平行移動］アイコンをクリックします。**



ポインタが 3D 表示画面の平行移動を行う状態を示すアイコンになります。



2　**マウスの左ボタンを押したまま移動させたい方向にドラッグします。**



参考

上記の操作のほかに、マウスホイールを押したまま移動させたい方向にドラッグすることでも視点の平行移動ができます。

A-2-4　ズームインおよびズームアウト

3D 表示画面でズームインおよびズームアウトする操作について説明します。

参考

関連動画

- ズームインとズームアウト
  https://www.fa.omron.co.jp/video/detail/6294913720001/
  3D 表示画面をズームインまたはズームアウトする操作を説明しています。

1　**3D 表示画面の［ズーム］アイコンをクリックします。**

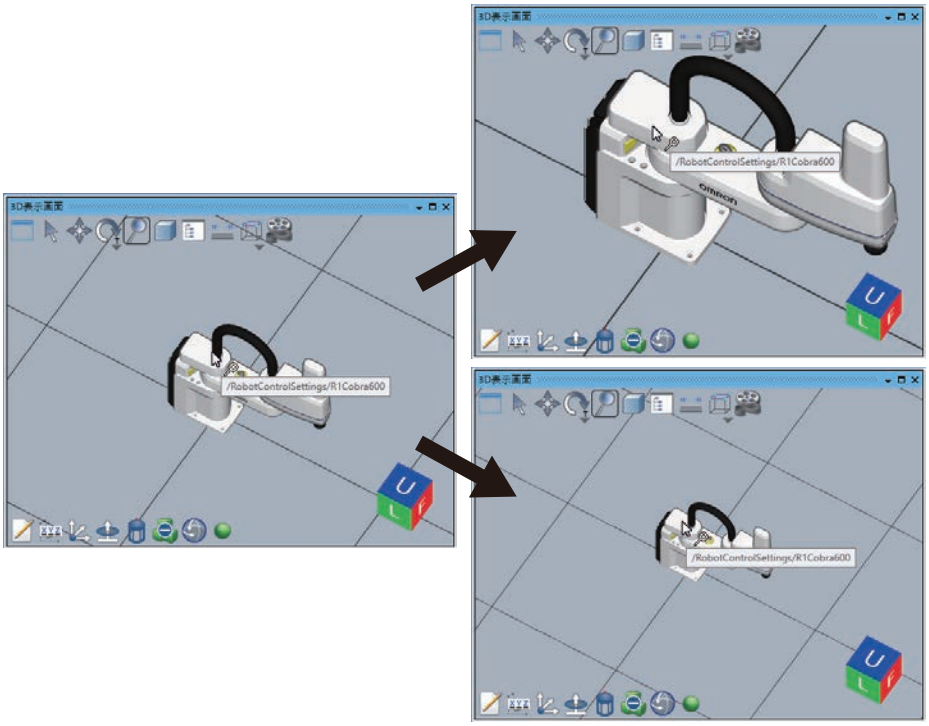


ポインタがズームを行う状態を示すアイコンになります。



2　**マウスの左ボタンを押したままドラッグします。**

右方向にドラッグ: 拡大（ズームイン）
左方向にドラッグ: 縮小（ズームアウト）



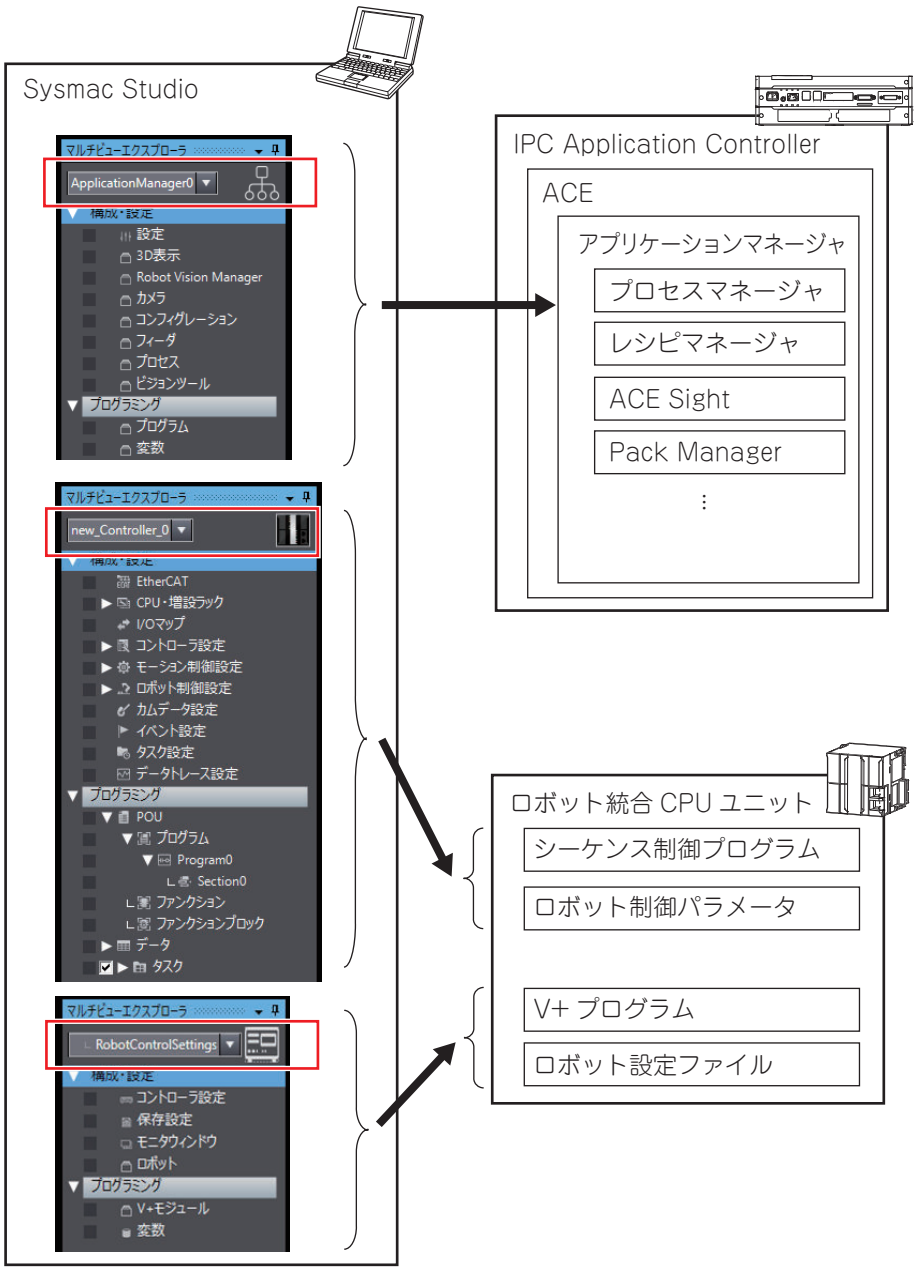
参考

マウスホイールを回転することで、ズームインおよびズームアウトすることもできます。
マウスポインタの位置がズームインおよびズームアウトの中心点となります。

A-3　Sysmac Studio の設定項目と設定対象

Sysmac Studio のマルチビューエクスプローラ上の項目と各ハードウェアの対応を以下に示します。
マルチビューエクスプローラのデバイスに［ApplicationManager0］を選択した場合は IPC Application Controller のアプリケーションマネージャの設定となります。［new_Controller_0］を選択した場合はロボット統合 CPU ユニットのシーケンス制御プログラムの設定、［RobotControlSettings］を選択した場合はロボット統合 CPU ユニットの V+プログラムの設定となります。
詳細については、各製品のマニュアルを参照してください。

マルチビューエクスプローラ上の設定項目　　設定対象



A-4　トラブルシュート機能の操作方法

Sysmac Studio でのトラブルシュート機能の操作方法については、『Sysmac Studio ロボット統合システム構築機能 オペレーションマニュアル ロボット統合 CPU ユニット編 (SBCA-464)』を参照してください。

オムロン株式会社 インダストリアルオートメーションビジネスカンパニー

製品に関するお問い合わせ先

お客様相談室

フリー通話 **0120-919-066**（クイック　オムロン）

携帯電話の場合、

055-982-5015（有料）をご利用ください。

受付時間：9:00～17:00（土・日・12/31～1/3を除く）

オムロンFAクイックチャット

www.fa.omron.co.jp/contact/tech/chat/

技術相談員にチャットでお問い合わせいただけます。（I-Webメンバーズ限定）

受付時間：平日9:00～12:00 / 13:00～17:00（土日祝日・年末年始・当社休業日を除く）

※受付時間、営業日は変更の可能性がございます。最新情報はリンク先をご確認ください。

その他のお問い合わせ：納期・価格・サンプル・仕様書は貴社のお取引先、または貴社担当オムロン販売員にご相談ください。オムロン制御機器販売店やオムロン販売拠点は、Webページでご案内しています。

オムロン制御機器の最新情報をご覧いただけます。緊急時のご購入にもご利用ください。 **www.fa.omron.co.jp**

●本誌に記載のない条件や環境での使用、および原子力制御・鉄道・航空・車両・燃焼装置・医療機器・娯楽機械・安全機器、その他人命や財産に大きな影響が予測されるなど、特に安全性が要求される用途に使用される際には、当社の意図した特別な商品用途の場合や特別の合意がある場合を除き、当社は当社商品に対して一切保証をいたしません。

●本製品の内、外国為替及び外国貿易法に定める輸出許可、承認対象貨物（又は技術）に該当するものを輸出（又は非居住者に提供）する場合は同法に基づく輸出許可、承認（又は役務取引許可）が必要です。

オムロン商品のご用命は

Man. No. SBCE-493E　2024年7月現在

©OMRON Corporation 2020 - 2024 All Rights Reserved.
お断りなく仕様などを変更することがありますのでご了承ください