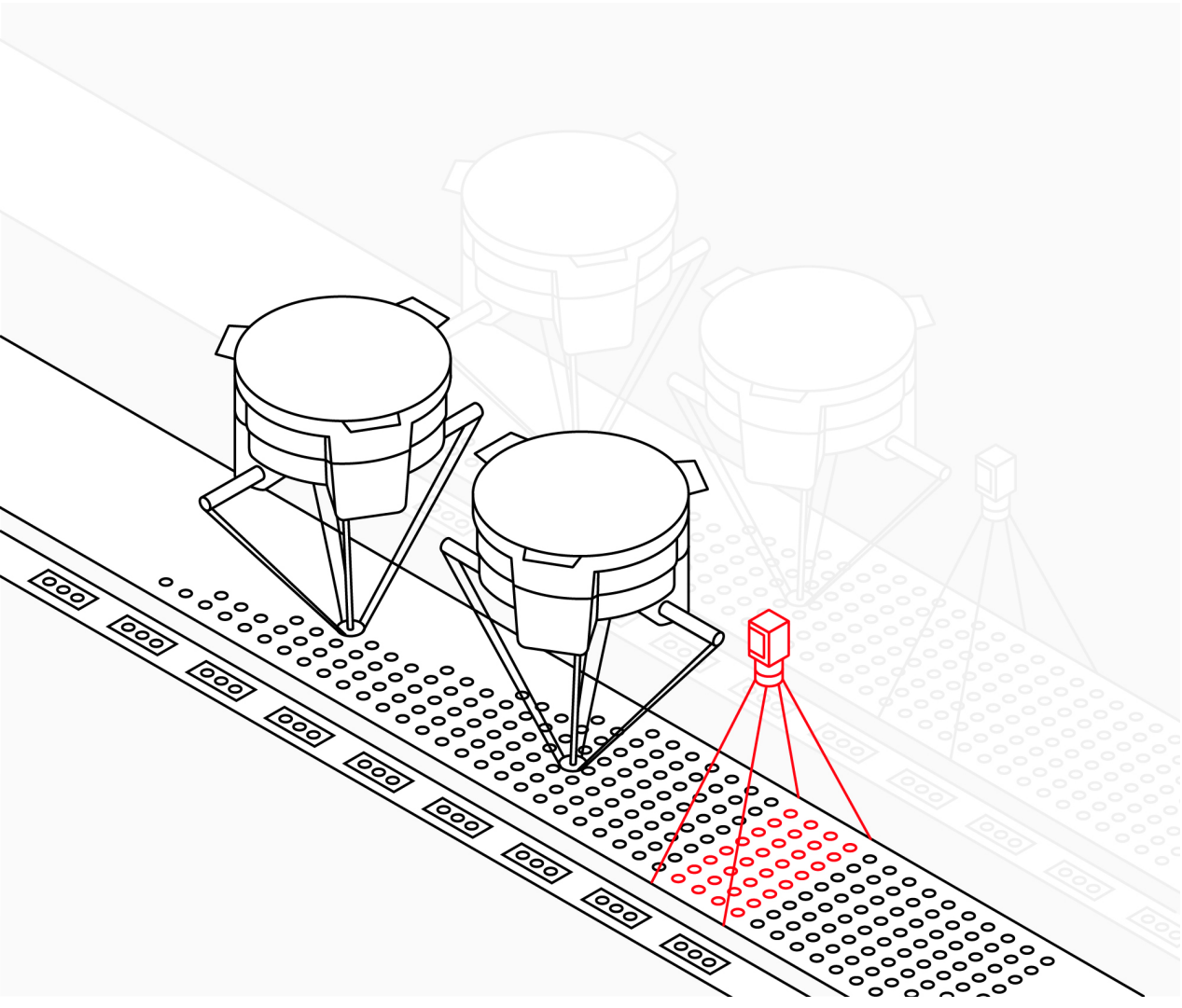
ROBOTICS

アプリケーションマニュアル

PickMaster® Twin - PowerPac



Trace back information:
Workspace Main version a676
Checked in 2025-06-08
Skribenta version 5.6.019

アプリケーションマニュアル

PickMaster® Twin PowerPac Release 3.0.1

OmniCore and IRC5

文書ID: 3HAC092761-012
改訂: B

© Copyright 2025 ABB. All rights reserved.
仕様は事前の通知なく変更されることがあります。

本書に記載されている情報は事前の通知なく変更されることがあり、ABBは通知の義務を負いません。ABBは、本書における間違いや誤記に対して一切の責任を負いません。

本マニュアルのいずれかに明示的に記載されている場合を除き、本書のいかなる内容も、ABBによる損失、個人または財産への損害、特定の目的への適合性などに対するいかなる種類の保証または担保責任としても解釈されないものとします。

いかなる状況においても、本書とここに記載されている製品の使用によって発生した偶発的または必然的損害に対してABBは一切責任を負いません。

本書およびそのすべての部分は、ABBの書面による許可なしに複写または複製することはできません。

今後の参考のために保管しておいてください。

本書の追加部数はABBから入手できます。

オリジナルの説明の翻訳。

© Copyright 2025 ABB. All rights reserved.
仕様は事前の通知なく変更されることがあります。

目次

本書の概要 9
安全性について 13

1 PickMaster PowerPacへようこそ 15

1.1 概要 15
1.2 PickMaster PowerPac用語 17

2 設置 19

2.1 PickMaster パッケージ 19
2.2 PickMaster® Twin ハードウェア接続図 21
2.3 システム要件 22
2.3.1 システム要件 22
2.3.2 Ethernetスイッチ 24
2.3.3 ビジョンシステム 25
2.3.4 カメラの要件 28
2.4 PickMaster PowerPac ライセンス 31
2.5 自己署名証明書 35
2.5.1 証明書の取り扱い 35
2.5.2 PickMaster® Runtimeデフォルトの証明書を自己署名証明書に変換 36
2.5.3 OpenSSLによる自己署名証明書の生成 38
2.5.4 自己署名証明書のインストール 40
2.6 PickMaster時刻同期化サービス 42
2.7 ソフトウェアのインストール 47
2.7.1 RobotStudio のインストール 48
2.7.2 PickMaster Twin Clientのインストールとアンインストール 49
2.8 電気接続 51
2.8.1 ネットワークの設定 51
2.8.2 カメラの接続 59
2.8.3 I/O信号の接続 62
2.8.4 ロボット コントローラの設定 64
2.8.5 オプションのロボットとプロセス設定 65
2.8.6 6軸ロボット構成 66

3 PickMaster PowerPac のナビゲート 67

3.1 メインウィンドウ 67
3.2 リボンタブ 69
3.3 ツリー ビュー ブラウザ 77
3.3.1 レイアウト 77
3.3.2 Process 87
3.4 ステーションビュー 89
3.5 ログビュー 90
3.6 ステータスビュー 91

4 PickMaster PowerPacを使用する作業 93

4.1 概要 93
4.2 フレーム関係 95
4.2.1 座標系とは何ですか? 95
4.2.2 PickMaster® Twin のフレーム関係 102
4.3 VirtualRuntime（VRT）でのレイアウトおよびプロセスでソリューションを設定 115
4.3.1 ソリューション 115
4.3.2 レイアウト 116
4.3.2.1 コントローラ 116
4.3.2.2 グリッパー 119
4.3.2.3 コンベヤ 122
4.3.2.4 カメラ 124

© Copyright 2025 ABB. All rights reserved.

4.3.2.5 I/Oセンサー 126
4.3.2.6 External sensor（外部センサー） 128
4.3.2.7 作業範囲 130
4.3.2.8 Indexed work area（インデックスワークエリア） 133
4.3.2.9 ポジションジェネレータ 136
4.3.2.10 キャリブレーション 141
4.3.3 プロセス 142
4.3.3.1 Item（項目） 142
4.3.3.2 Container（コンテナ） 149
4.3.3.3 フロー 157
4.3.3.4 レシピ 160
4.3.4 操作 173
4.3.4.1 シミュレーション 173
4.4 Real Runtime（RRT）の設定 179
4.4.1 Real Runtimeへの切り替え 179
4.4.2 カメラの設定 190
4.4.3 キャリブレーションロボット 194
4.4.4 リニア コンベアのキャリブレーション 195
4.4.4.1 DSQC 2000 を使ったリニアコンベアのキャリブレーション 196
4.4.4.2 DSQC 377 を使ったリニアコンベアのキャリブレーション 219
4.4.5 円形コンベアのキャリブレーション 226
4.4.5.1 DSQC 2000 (CTM)による円形コンベアのキャリブレーション 228
4.4.5.2 DSQC 377での円形コンベアのキャリブレーション 257
4.4.5.3 円形コンベアのタイプ設定 263
4.4.6 インデックスワークエリアのキャリブレーション 265
4.4.7 コンベアのキャリブレーションの検証 280
4.4.8 カメラのキャリブレーション 282
4.4.8.1 ライブ画像の表示 289
4.4.8.2 詳細なビジョン情報 290
4.4.8.3 イメージウィンドウ 292
4.4.9 ビジョンモデルの追加 293
4.4.9.1 ビジョンモデル化 293
4.4.9.2 Geometricモデルの設定 296
4.4.9.3 Blobモデルの構成 305
4.4.9.4 検査モデルの構成 312
4.4.9.5 カラービジョンの使用 324
4.4.10 生産開始 333
4.4.11 生産中のロボットの管理 335

5 高度な機能 337

5.1 UserScripts 337
5.2 外部センサー 357
5.3 さまざまな高さの製品の扱い（2.5D ビジョン） 371
5.4 フローを使用した生産 (ゴースト ピッキング) 379
5.5 複数ブランドのカメラの設定 383

6 RAPIDリファレンス 387

6.1 命令 387
6.1.1 AckItmTgtーアイテムターゲットの承認 387
6.1.2 FlushItmSrcーアイテムソースのフラッシュ 389
6.1.3 GetItmTgt - 次のアイテムターゲットを取得する 390
6.1.4 NextItmTgtType - 次のアイテムターゲットのタイプを取得 396
6.1.5 QStartItmSrc - アイテムソースでキューを開始 398
6.1.6 QStopItmSrc - アイテムソースでキューを停止 399
6.1.7 ResetFlowCount - フローカウンタをリセット 400
6.1.8 ResetMaxUsageTime - 最大測定使用時間をリセットします。 401
6.1.9 UseReachableTargets - 達せられるターゲットを使用する 402
6.2 関数 405
6.2.1 GetMaxUsageTime - 最大測定使用時間を受ける 405

© Copyright 2025 ABB. All rights reserved.

6.2.2 GetQueueLevel - キューレベルを取得 406
6.2.3 GetQueueTopLevel - キューのトップレベルを取得 408
6.2.4 GetFlowCount - 限界を超えたアイテム数の取得 409
6.3 データタイプ 410
6.3.1 itmtgt - アイテムターゲットデータ 410
6.3.2 selectiondataー選択データ 413
6.3.3 sortdataー並べ替えデータ 416
6.4 RAPIDプログラム 417
6.4.1 RAPID プログラム 417
6.4.2 変数 422
6.4.3 ルーチン 425
6.5 プログラムの例 429
6.5.1 例：1つのピックワークエリアと2つのプレイスワークエリアの組み合わせ 429
6.5.2 例：2つのピックワークエリアと1つのプレイスワークエリアの組み合わせ 430
6.5.3 例：1つのピックと1つのプレイスワークエリアの組み合わせ 431
6.5.4 例：ダブルピック・シングルプレイス 432
6.5.5 例：事前定義パターンをインデックスワークエリアにプレイス 435
6.5.6 例：クリアランスゾーンに基づくアイテムの選択 437
6.5.7 例：負の Y 方向の並べ替え 440
6.5.7.1 リニアコンベアの負の y 方向の並べ替え 440
6.5.7.2 円形コンベアの負の半径方向の並べ替え 443
6.5.8 例：インデックスワークエリアおよび事前定義ポジション 446
6.5.9 例：インデックスワークエリアへの新規ポジションの自動生成 447
6.5.10 例：アイテム バッファ 448

7 トラブルシューティング 453

7.1 トラブルシューティング入門 453
7.2 トラブルシューティング時の安全 454
7.3 PickMaster Twin Powerpac のログの管理 455
7.4 PickMaster Twin Operator のログの管理 456
7.5 PickMaster Twin Runtime のログの管理 457
7.6 Runtimeエラーコード 459
7.7 故障の徴候またはエラー 475
7.7.1 警告 4326 ~ 4329 475
7.7.2 カメラが画像を取得しない 479
7.7.3 ロボットが動かない 480
7.7.4 ポジション精度が低い、または変動する 481
7.7.5 ポジションが2度使われる 482
7.7.6 PickMaster のカメラの解像度の問題 483
7.7.7 画像ダイアログを表示できません 484
7.7.8 円形コンベアでカメラを使用すると、ロボットがアイテムを握らない 487
7.7.9 開始ボタンを、インデックス作成の作業範囲で`Arm check point limit`エラー発生後にクリックすると、ロボットが起動しない。 489
7.7.10 アーカイブ 「OE」 データを読み取れませんでした 490

8 予備部品 491

8.1 カメラ部品 492
8.2 USBドングル部品 495
8.3 GigE ネットワーク カードの部品 496

9 回路図 497

9.1 回路図 497

索引 499

© Copyright 2025 ABB. All rights reserved.

このページは意図的に空白のまま残しています

本書の概要

本書について

本マニュアルには、 PickMaster PowerPacのインストール、設定、および操作の手順が記載されています。

> **注記**
>
> コントローラを操作するためのすべての安全情報は、コントローラの製品マニュアルに記載されています。

> **ヒント**
>
> ソフトウェアのバージョンにより機能説明が異なる場合があります。
> ソフトウェアのバージョンに合わせて、対応するマニュアルのバージョンを必ずお読みください。

使用法

本マニュアルは、PickMaster Twin システムのインストール、設定、およびメンテナンス中に使用する必要があります。

PickMaster PowerPacは、ポータブル ラップトップ PC のエンジニアリング ツールとして、オフラインで使用したり、コミッショニング目的でインストールしてホスト コンピュータにオンライン接続したりすることを目的としています。PickMaster PowerPacは、実稼働の際にホストコンピューターとして使用されることは意図されていません。

対象ユーザー

本書は、次のユーザーを対象としています。

- 設備導入担当者
- プログラマ
- インテグレータ
- Operator

事前知識

ABB ロボットで作業するメンテナンス/修理設備導入担当者は、ABB によるトレーニングを受け、機械的および電気的なインストール/修理/メンテナンス作業に必要な知識を持っている必要があります。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

続き

サイバーセキュリティ

本製品は、ネットワークインターフェースを介して情報やデータに接続し、やり取りするように設計されています。製品とユーザーのネットワークまたは他のネットワークとの安全な接続を継続的に保証するのはユーザーの責任です。

お客様は、製品、ネットワーク、そのシステム、あらゆる種類のセキュリティ違反、不正アクセス、干渉、侵入、漏洩、および/またはデータや情報の盗難に対するインターフェースを保護するための適切な手段（ファイアウォールのインストール、認証手段の適用、データの暗号化、ウイルス対策プログラムのインストールなど）を確立および維持するものとします。ABB Ltdおよびその事業体は、このようなセキュリティ違反、不正アクセス、干渉、侵入、漏洩、および/またはデータや情報の盗難に関連する損害および/または損失に対して責任を負いません。

注記

資格のある担当者のみがスクリプト ファイルを作成または変更してください。

スクリプト ファイルを使用して実行するときにセルが安全であることを確認するのは作成者の責任です。

アプリケーション ポートとプロトコルのタイプ

次の表は、PickMaster PowerPacと他のコンポーネント間の使用ポート番号、通信プロトコルのタイプ、および使用法を示しています。

ポート	プロトコルのタイプ	使用法
50000	TCP	PickMaster Runtime の RIS2 コマンド
6001	TCP	PickMaster Runtime のエミュレーション
3	TCP	ビジョン クライアント
5	TCP	ビジョン サーバー
9000	TCP	Zenon イベント ポート
319-320	UDP	時刻同期サービス
502	TCP	Modbus
34964	UDP	Profinet
44818	TCP	EtherNet/IP
2222	UDP	EtherNet/IP

参考

ヒント

すべてのドキュメントは、myABBビジネスポータル（*www.abb.com/myABB*）から入手できます。

OmniCore

参考資料	文書ID
アプリケーション マニュアル - *PickMaster Twin Operator*	*3HAC092762-001*

次のページに続く

© Copyright 2025 ABB. All rights reserved.

参考資料	文書ID
操作マニュアル - *PickMaster Twin Recipe Manager*	*3HAC092763-001*
製品仕様 - *PickMaster® Twin*	*3HAC092765-001*
Circuit diagram - PickMaster Twin	*3HAC024480-020*
ロボット用安全マニュアル - マニピュレータおよび*IRC5*または*OmniCore*コントローラ [i]	*3HAC031045-012*
Application manual - Conveyor tracking	*3HAC066561-001*
製品マニュアル - *OmniCore C30*	*3HAC060860-012*
製品マニュアル - *OmniCore C90XT*	*3HAC073706-012*
Operating manual - OmniCore	*3HAC065036-001*
Operating manual - Integrator's guide OmniCore	*3HAC065037-001*
Application manual - Controller software OmniCore	*3HAC066554-001*
テクニカルリファレンスマニュアル - *RobotWare 7* のイベントログ	*3HAC042927-001*
Technical reference manual - Lubrication in gearboxes	*3HAC042927-001*
Technical reference manual - System parameters	*3HAC065041-001*

[i] このマニュアルには、マニピュレータとコントローラの製品マニュアルから引用したすべての安全に関する説明が含まれています。

IRC5

参考資料	文書ID
アプリケーション マニュアル - *PickMaster Twin Operator*	*3HAC069977-012*
操作マニュアル - *PickMaster Twin Recipe Manager*	*3HAC092763-001*
製品仕様 - *PickMaster® Twin*	*3HAC073650-012*
Circuit diagram - PickMaster Twin	*3HAC024480-020*
ユーザーマニュアル - *RobotStudio*	*3HAC032104-012*
Application manual - Conveyor tracking	*3HAC050991-001*
製品マニュアル - *IRC5*	*3HAC047136-012*
Product manual - IRC5 Panel Mounted Controller	*3HAC027707-001*
ユーザーマニュアル - *FlexPendant*を搭載した*IRC5*	*3HAC050941-012*
Operating manual - IRC5 Integrator's guide	*3HAC050940-001*
ユーザーマニュアル - *IRC5* のトラブルシューティング	*3HAC020738-012*
Technical reference manual - RAPID Instructions, Functions and Data types	*3HAC050917-001*
Technical reference manual - RAPID Overview	*3HAC050947-001*
Technical reference manual - System parameters	*3HAC050948-001*

外部参考資料

参考資料	説明
Cognex Ethernet Camera Tool	カメラネットワークの構成用。
Gigabit Ethernet Performance Driver	カメラ通信用。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

続き

参考資料	説明
aca1440-73gc	Basler Ace ギガビット イーサネット カメラとギガビットイーサネットカメラ用スイッチに関する情報。 **注記** このカメラは、ABB から購入する必要がある ABB カスタマイズ ファームウェアを使用します。
sca1300-32gc	Basler Scout ギガビット イーサネット カメラとギガビットイーサネットカメラ用スイッチに関する情報。 **注記** このカメラは、ABB から購入する必要がある ABB カスタマイズ ファームウェアを使用します。
CognexPCConfigGuide	ビジョンシステム用PCの詳細な要件情報。

改訂

改訂	説明
A	第1版
B	PickMaster® Twin 3.0.1でリリースされました。 • 一部訂正 • 多言語ドキュメントをサポートしました。

© Copyright 2025 ABB. All rights reserved.

安全性について

作業者の安全

ロボットは重量があり、速度に関係なく非常に強力な機械です。 一時停止あるいは長時間停止の後、急に危険性の高い早い動きをする可能性があります。 動きのパターンが予測されている場合でも、外部信号により操作への変更をき、っかけに予想できない動きをする可能性もあります。

したがって、安全保護区域に立ち入る場合は安全基準を遵守することが重要です。

安全基準

ロボットでの作業を開始する前に、『ロボット用安全マニュアル - マニピュレータおよび*IRC5*または*OmniCore*コントローラユーザーマニュアル』に記載されている安全基準をよくお読みください。

PickMaster® Twin を使用する場合

- PickMaster® Twinと一緒に使用する場合、関連する規格と安全指令を順守するのはユーザーの責任です。 また、適切な使用のためのアプリケーション マニュアルを遵守する必要があります。
- PickMaster® Twinを使用できるのは、適切なトレーニングを受け、必要な知識を備えた担当者のみです。
- PickMaster® Twin を導入するインテグレータ(統括者)は、安全に関する責任を負います。
- 自動モードの操作は、可能な限り、安全保護区域に人がいない状態で行う必要があります。
- 非常停止機能が有効であることを確認するために、非常停止機能が使用できる必要があります。
- PickMaster® Twin は、操作停止 (プログラム停止) のみを提供します。 インテグレータは、適切な通常停止 (機械停止) がシステムで正しく設定されていることを確認する必要があります。
- 非常停止状態の原因となった危険な状況が存在しないことを確認してください。 非常停止状態を解除するには、非常停止ボタンを手動で戻します。
- 機械の停止はインテグレータの責任であり、現地の法律に従って対処する必要があります。
- インテグレータは、最終アプリケーションのリスク評価を実施する責任があります。
- 眼や喉頭などの敏感な体の部位は、個人保護具（PPE）で保護する必要があります。
- PickMaster® Twinを使用する際は、保護対策をなされていることが前提条件となります。 PickMaster® Twin は、ロボットのターゲットが常にセーフ ゾーンにあることを保証するものではありません。 Safe-move機能の使用や適切なロボット作業範囲の設定など、保護対策を講じることはインテグレータの責任です。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

- 安全関連のステータスと操作は、コントローラとセーフティシステムによって処理されます。 PickMaster® Twin のステータス情報は、安全関連の情報および操作の入力として使用してはなりません。
- カメラなどのハードウェア部品を取り付け/調整/交換する際には、保護対策をなされていることが前提条件となります。
- PickMaster® Twin の停止機能は、A-stop/E-stop またはその他の安全関連の停止の代わりに使用することはできません。

注記

PickMaster Twinがロボットコントローラからアクティブな AS/ESのステータスを取得した場合、キュー内のアイテムはクリアされません。一旦AS/ESが非アクティブ化されると、**Start**/**Unhold**/**Unsuspend** のボタンをクリックすることで生産を再開することができます。

© Copyright 2025 ABB. All rights reserved.

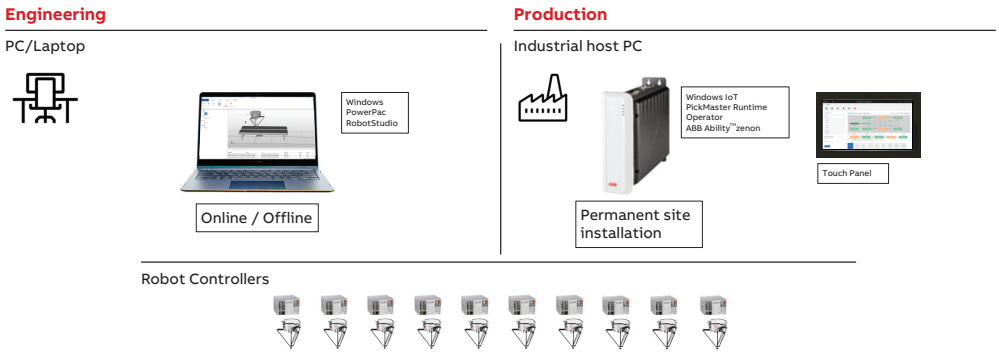
1 PickMaster PowerPacへようこそ

1.1 概要

PickMaster® Twinについて

PickMaster® Twin は、画像認識に基づいたランダム フロー製品の高速ピッキング用に設計されたアプリケーション製品です。 PickMaster® Twin は、多数のロボット、カメラ、コンベア、およびインデックスワークエリアで構成された小規模または大規模なライン レイアウトのさまざまなバリエーションの使いやすい設定、シミュレーション、および操作をサポートします。 ピッキング設備の提案、エンジニアリング、試運転、運用、メンテナンス・サポートまでのライフサイクルを網羅した生産システムです。

PickMaster® Twin は、ロボット コントローラを介してピッキング アプリケーションで ABB ロボットを制御するためのモジュール製品です。アイテムのピック アンド プレース操作を実行するように構成できます。ビジョン システムは、コンベア ベルト上にランダムに配置されたアイテムを見つけたり、静的ワークエリアにインデックスを付けたりするために使用されます。PickMaster PowerPac は、仮想システムを使用したオフライン シミュレーションと、実際のインストールに直接接続されたオンライン モードでアプリケーションを設定および検証することを目的としたエンジニアリング ソフトウェアです。包括的なグラフィカル インターフェイスを使用して強力なアプリケーションを設定し、複数のロボットを制御して、センサーで検出されたアイテムをさまざまなコンベア ベルト上でピックアップして配置することができます。PickMaster Operator は、生産環境で PickMaster Twin を統合および操作するためのソフトウェアです。



xx2100001619

PickMaster PowerPacは、次の特別なニーズの一部に合わせてカスタマイズできます。

- 統合されたビジョンシステムにより、連続的に移動するコンベアでの完全なランダム操作や、フィーダまたはトレイでの絶対的な正確なポジション決めに使用できます。
- ビジョン認識を使用しなくても、複数のコンベヤ上での生産数を表示し、効率的な生産を支援するツールとして使用できます。
- 位置検出により、効率的な品質検査や製品の分類（いずれかもしくは両方）に対応。

PickMaster® Twin は、次のモジュールで構成されています。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

PickMaster® PowerPac

RobotStudio™を搭載した視覚的な3D環境でのオフラインおよびオンライン設定と試運転のための使いやすいソフトウェア。

PickMaster® Operator

ABB Ability™ zenon データ管理ソフトウェア上に構築された、現場で PickMaster を操作および統合するための最先端のユーザー インターフェイス。PickMaster Operator を使用すると、ユーザーはすべてのレシピをホスト上で直接管理することもできます。

PickMaster® Runtime

正確なロボット ガイダンスと品質検査のための統合ビジョン ソフトウェアを含む、多数のロボットとコンベアのパッケージング プロセスを調整するための効率的な Runtime操作ソフトウェア。

- Virtual Runtime：仮想ロボット コントローラに接続されたクライアント システム上のシミュレーションされた仮想環境で PickMaster プロセスを実行します。
- Real Runtime: 実際のロボットコントローラに接続された Host コンピュータにインストールした実生産環境で PickMaster プロセスを実行します。

注記

PickMaster® Twin は、さまざまなハードウェア構成で提供されます。 詳細については、製品仕様 - *PickMaster® Twin* を参照してください。

PickMaster PowerPacについて

本マニュアルでは、2つのモードのエンジニアリング ソフトウェアとしてPickMaster PowerPacをインストールして使用する方法について説明します。

- 仮想システムを使用したオフライン シミュレーションでのアプリケーションの設定と検証。
- 実際の設備に直接接続されたオンラインモードでの試運転。

本マニュアルでは、実際のシステムのコンポーネント、それらのインストール、設定、およびキャリブレーションについても説明しています。

© Copyright 2025 ABB. All rights reserved.

1.2 PickMaster PowerPac用語

これらの用語について

本マニュアルでは、いくつかの用語が通常とは異なる意味で使用されています。本マニュアルでのこれらの用語の定義を以下に示します。一部の用語はピック & プレイスする工程を記述する際に使用されます。

用語リスト

定義欄内のイタリック体の用語は用語リストに含まれており、それぞれの定義が説明してあります。

用語	定義
PickMaster PowerPac	Virtual RuntimeとRealRuntimeを使用してピッキングラインをシミュレーションおよび試運転するために使用されるPickMaster PC エンジニアリング ソフトウェアの製品用語。
PickMaster Operator	PickMasterアプリケーションを生産で実行するために使用される、 PickMasterプロダクションOperatorーインターフェースソフトウェアの製品用語。PickMaster Operatorは、 PickMaster PowerPacによって生成されたソリューションを読み取り、書き込み可能。Real Runtimeにアクセス可能。
PickMaster Recipe Manager	工場出荷時のインストールでホスト コンピュータ上でアイテム、コンテナ、レシピを作成、変更するためのソフトウェアのマーケット名。Recipe Manager は PickMaster Operator から起動します。
PickMaster VirtualRuntime（VRT）	シミュレーションでの仮想ピック & プレイスする操作のすべての計算を調整るコア エンジン。
PickMaster RealRuntime（RRT）	実際の製品におけるピック & プレース操作のすべての計算を調整するコア エンジン。Runtimeはカメラおよびロボット コントローラと通信します。 Runtimeとも呼ばれる。
PickMaster Twin Client	それは、PickMaster PowerPacソリューションの設定、シミュレーション、および試運転を行うためのクライアントコンピュータです。PickMaster Twin Clientインストール パッケージは、クライアント コンピュータにインストールされます。これには、PickMaster PowerPac、PickMaster 仮想Runtime、およびPickMaster Real Runtimeが含まれています。
PickMaster Twin Host	これは、生産でPickMaster Twinを操作および管理するためのホスト コンピュータです。PickMaster Twin Hostインストール パッケージはホスト コンピュータにインストールされます。これには、PickMaster OperatorとPickMaster real Runtimeが含まれています。
ソリューション	PickMaster Twin構成結果の保存用フォーマット。
レシピ	ステーションで実行する処理を格納するための、ピック & プレイスする処理に関するパラメータの集まりとフォーマット。
レイアウト	PickMasterインストールのロボット、ワークエリアなどの静的オブジェクトです。
プロセス	PickMasterの拾うプロセスとすべてのアイテム、コンテナ、フロー、およびレシピの説明。
ワークエリア	ロボット用のピック & プレイスする定義されたエリア。
アイテム	PickMaster PowerPacアプリケーションにおいて選択または配置する特定のオブジェクトを示す総称。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

1.2 PickMaster PowerPac用語
続き

用語	定義
コンテナ	特定のパターンを設定できる形状と、パターン内の各ポジションに使用するアイテムを定義します。
ポジションジェネレータ	コンベアとインデックスワークエリアのセンサー設定を定義します。
エミュレーション	実際のセルやラインの挙動を模倣し、画面上に表示する活動。
ゴーストピッキング	生産が記録された仮想アイテムを使用して拾うため、拾う実際のアイテムがない場合の一種の予行演習。
オフラインシミュレーション	仮想ロボットに接続したときのシミュレーション プロセス。

© Copyright 2025 ABB. All rights reserved.

2 設置

2.1 PickMaster パッケージ

PickMaster Twin使用コンセプト

PickMaster PowerPacは、試運転、新しいレシピの導入、メンテナンス、およびサービスの目的で複数の物理的な設備に接続できるさまざまな設備のソリューションをホストできるラップトップ コンピュータに取り付けるように設計されており、それぞれには独自の永続的なホスト コンピュータがあります。

ソフトウェアのインストールパッケージは2種類：PickMaster Twin Clientはポータブルエンジニアリングシステム用、 PickMaster Twin Hostは常設工場システム用。

Client設置には、物理的機器は不要。物理的コンポーネントの設置、設定、校正はすべて、Host システム上で実施。

PickMaster Twin Client

PickMaster Twin のオフライン設定、シミュレーション、およびテスト用のインストール パッケージは、PickMaster Twin Clientと呼ばれます。次のソフトウェアがインストールされます。

- PickMaster PowerPac
- PickMaster Virtual Runtime
- PickMaster リアル Runtime

> **注記**
>
> 本パッケージはエンジニアリング用であり、最終的な工場設置用ではない。同一コンピューター上でRealRuntimeに切り替え機能は、あくまで検査目的で、オフラインでビジョンモデルを作成および編集する目的で使用可能。ビジョンデモドングルは本目的に使用可能。

ソフトウェア インストール パッケージ

ABB に登録済みのお客様は、PickMaster Twin Clientの最新バージョンとPickMaster PowerPacのユーザー ドキュメントを ABB ダウンロード センターからダウンロードできます。

> **ヒント**
>
> ダウンロード センターのアドレスは https://new.abb.com/products/robotics/application-software/pickmaster です。

> **注記**
>
> PickMaster ソフトウェアは 64 ビット バージョンで利用できます。

PickMaster Twin Client が正常にインストールされると、インストール フォルダ *Documentation* にPickMaster PowerPacのユーザー ドキュメントとキャリブレーション ペーパーが用意されます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

2.1 PickMaster パッケージ
続き

注記

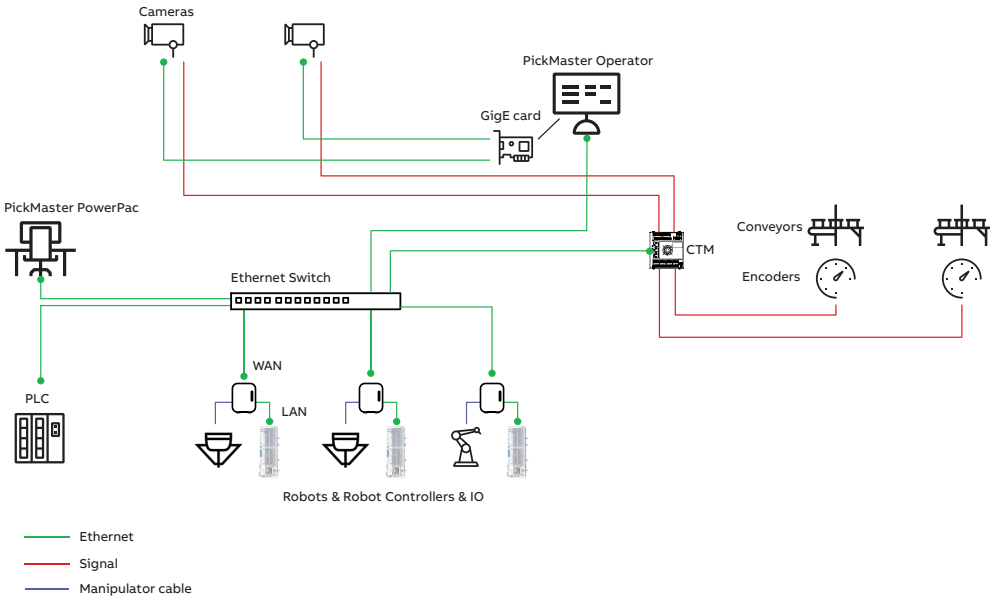
PickMaster PowerPacの新しいバージョンの をインストールする前に、PickMaster PowerPacの古いバージョンの をアンインストールする必要があります。

© Copyright 2025 ABB. All rights reserved.

2.2 PickMaster® Twin ハードウェア接続図

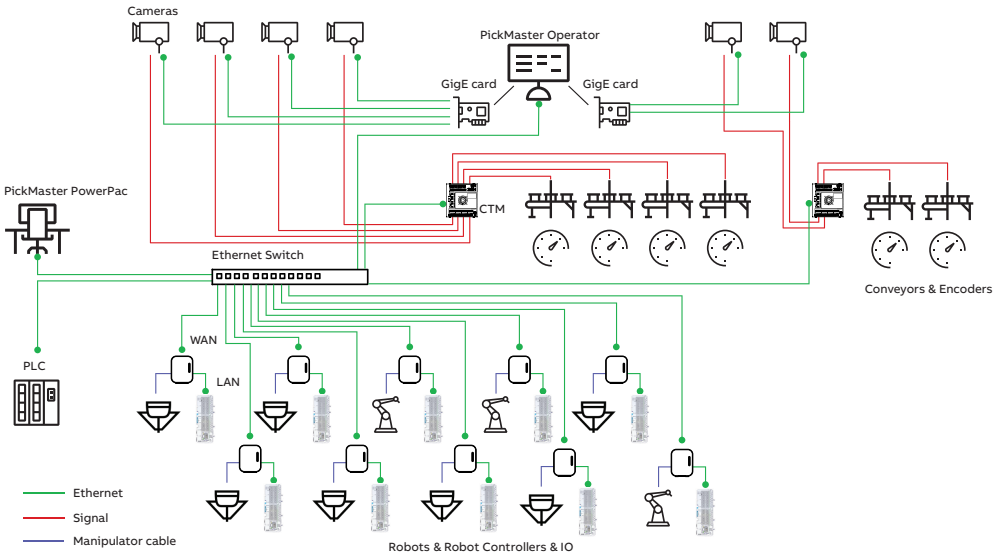
例

次の図は、ロボット 3 台、カメラ 2 台、コンベア 2 台の設置例を示しています。



xx1900001746

次の図は、ロボット 10 台、カメラ 6 台、コンベア 6 台の設置例を示しています。



xx2400000560

© Copyright 2025 ABB. All rights reserved.

2.3 システム要件

2.3.1 システム要件

ハードウェアとソフトウェアの要件

ハードウェアの要件

パーツ	PickMaster Twin Client	PickMaster Twin Host
CPU	2.0GHz 以上のプロセッサ、複数のコアを推奨します。	
メモリ	8 GB の RAM が最小要件です。 ビジョンや重い CAD モデルを扱う場合は、16 GB 以上を推奨します。	
ディスク	500 GB のソリッド ステート ドライブ (SSD) が推奨されます。	
グラフィックカード	大手ベンダー製の高性能でDirectX 11対応、ゲーミンググラフィックカード。高度な発光モードにはDirect3D機能レベル10_1以上が必要です。	
ディスプレイ設定	23 インチ、1920 x 1080 ピクセル以上の解像度を推奨します。 PickMaster Twin Host には、マルチタッチ スクリーンを推奨します。	
マウス	3 ボタン マウス	
3Dマウス［オプション］	3DConnexion製の任意の3Dマウス。 http://www.3dconnexion.comを参照してください。	
USB スロット	USB スロットが最低 2 つ。	
Ethernet ポート	Ethernet ポートが少なくとも 1 つ。	
PCI Express スロット	NA(適用不可)	PCI Express スロットPickMaster Twin Host 用の 145 mm x 110 mm サイズ PCIE カード用に、少なくとも 1 つの空き PCI Express x4 スロット。

注記

ソフトウェアを実行するときは、多くのメモリを消費する他のソフトウェアを閉じてください。そうしないと、ソフトウェアの通常の使用に影響します。

ソフトウェア要件

パーツ	PickMaster Twin Client	PickMaster Twin Host
システム	Windows 10/11 (64 ビット)	
RobotStudio	RobotStudio 2024.3.1	
RobotWare	• RobotWare 7.16を搭載したOmnicore • RobotWare 6.15 を搭載した IRC5	
インストール パッケージ	PickMaster Twin Client (PickMaster PowerPac、PickMaster Runtime を含む)。	PickMaster Twin Host (PickMaster Operator、PickMaster Runtime、PickMaster Recipe Manager を含む)。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

パーツ	PickMaster Twin Client	PickMaster Twin Host
ライセンス	• アクティブなRobotStudio ライセンス • アクティブなPickMaster PowerPacライセンス。	• ホスト PC の場合: PickMaster Runtime ライセンス。 • ロボット コントローラの場合: PickMaster Cell ready または PickMaster Robot Ready (PickMaster Vision Ready あり/なし)

詳細については、製品仕様 - *RobotStudio* を参照してください。

© Copyright 2025 ABB. All rights reserved.

2.3.2 Ethernetスイッチ

概要

イーサネットスイッチは、PC と複数のロボットコントローラを接続するために使用されます。通信速度1000Mbit/s以上の産業用スイッチの使用を推奨します。

© Copyright 2025 ABB. All rights reserved.

2.3.3 ビジョンシステム

概要

PickMaster PowerPac はイーサネット経由で通信するカメラを使用してイメージを取得し、ターゲットを生成することができます。カメラと RuntimePC との間の通信には、イーサネットネットワーク（ネットワークインターフェースカード、ケーブル、スイッチ）を使用します。トリガー/ストロボと電源は、カメラの筐体にある Hirose12 ピン/6 ピンコネクタに接続します。イーサネットカメラには、ロボットコントローラから独立している外部電源を使用することをお勧めします。

ビジョンシステムの要件

イーサネットのカメラ通信用にサポートされているネットワークカードは、GigE network card DSQC1083 (3HAC078753-001)です。その他のネットワーク インターフェース カードは機能する可能性がありますが、検証されていません。

Gigabit イーサネット ビジョン システムには Cognex USB ライセンスが必要です。Runtimeの実行中は、USB スティックを接続する必要があります。

使用できるカメラの最大数は 10 台です。

ネットワークカードは空いている PCI-expressスロットに挿入します（PCI-express x4、x8、またはx16）。

色覚

色覚は標準機能として利用でき、次の機能があります。

- カラー カメラの接続性
- ホワイトバランスのキャリブレーション
- カラー フィルター設定

> **注記**
>
> ここでは、*Geometric*とBlobの前処理として実行されるカラー フィルタを定義できます。フィルタは、スタンドアロン、アライメント、およびサブ検査モデルで使用できます。

検証済みのカメラについては、検証済みカメラ ページ *59* を参照してください。

トリガーストロボケーブル

使用するケーブルは4種類あります。それらのトリガーストロボの接続は以下の通りです。

ケーブル	ID	説明
新たな取り替えケーブル(10 m)	2000034085	パワー-I/O PLC+ケーブルHRS 12p/開放、10m
古いスカウトケーブル(10 m)	2000026632	電源-I/Oケーブル、HRS 12p、オープン、10m
古いスカウトケーブル(10 m)	2000022909	パワー-I/O ケーブル、HRS 12p/開放、10m
エースカメラケーブル	2000034084	パワー-I/O PLC+ ケーブル 6p/開放、10m

次のページに続く

© Copyright 2025 ABB. All rights reserved.

2.3.3 ビジョンシステム
続き

2000034085 (Basler Scout カメラ) のトリガー ストロボ接続

次の表はBasler Scoutカメラへのトリガー/ストローブ/電源接続のための物理インタフェースを説明しています。

カメラの接続方法の詳細については、回路図を参照してください。

配線ペア	ピン番号	電線色	スカウトGigE	機能
1	1	白	カメラ グランド	0V(CamPower-)[i]
1	2	緑	カメラ グランド	0V(CamPower-)
2	3	桃	Opto in 1	トリガ[ii]
2	(5)[iii]	灰色	Opto in 1 グランド	0V (Cam I/O-)[iv]
3	4	赤	Opto in 2	未使用
3	(5)	青	Opto in 2 グランド	未使用
4	6	紫	Opto out 1	ストロボ[v]
4	(10)	黒	Opto out 1 VCC	24V (Cam I/O +)
5	7	赤色/青色	Opto out 2	未使用
5	(10)	灰色/桃色	Opto out 2 VCC	未使用
6	8	茶	Camera VCC	24V (CamPower +)
6	9	黄	Camera VCC	24V (CamPower +)
7	(10)	白色/緑色	Opto Out 3 VCC	未使用
7	11	茶色/緑色	Opto Out 3	未使用
8	(10)	白色/黄色	Opto Out 4 VCC	未使用
8	12	黄色/茶色	Opto Out 4	未使用

i カメラ用0/24V電源。ロボットコントローラがシャットダウンした場合にも電力を供給し続ける電源を使用することを強くお勧めします。
ii カメラに画像の取り込みを命令する入力信号。
iii 括弧内のピン番号 ”(X)” は、配線が内部的にピン ”X” に接続されることを示しています。
iv カメラのI/Oシステム用0/24V電源
v カメラが画像を取り込んだことを返す出力信号。

2000026632 および 2000022909 (Basler Ace および Cognex CIC-A カメラ) の古いトリガー ストロボ接続

ピン番号	電線色	スカウト GigE	機能
1	白	カメラ電源接地	0V(CamPower-)
2	緑	カメラ電源接地	0V(CamPower-)
3	青	I/O 入力1	トリガ
4	赤	I/O 入力2	未使用
5	灰	I/O 入力グランド	0V (Cam I/O-)
6	黒	I/O 出力1	ストロボ
7	紫	I/O 出力2	未使用
8	茶	カメラ電源VCC	24V (CamPower +)
9	黄	カメラ電源VCC	24V (CamPower +)
10	桃	I/O 出力VCC	24V (Cam I/O +)

次のページに続く

© Copyright 2025 ABB. All rights reserved.

ピン番号	電線色	スカウト GigE	機能
11	灰/桃	I/O 出力3	未使用
12	赤色/青色	I/O 出力4	未使用

2000034084 (Ace および Cognex CIC-A カメラ) のトリガー ストロボ接続

次の表はエースカメラへのトリガー/ストローブ/電源接続のための物理インタフェースを説明しています。

電源 I/O ケーブル HRS 6p/オープン、ツイスト、10 m - IO/電源ケーブル Basler ace または Cognex CIC-A GigE カメラの光結合 I/O の電源およびトリガー用ケーブル (長さ 10 メートル)。

ケーブルはカメラ側にHRS 6ピンコネクタが付いています。 個々の要件に合わせてケーブルを短くできるように、もう一方の端は開いています。

配線情報：

ピン番号	電線色	GigEg	機能
1	茶	カメラの電源	24V (CamPower +)
2	桃	光絶縁 IN (Line1)	トリガ
3	緑	未接続	0V (Cam I/O-)
4	黄	光絶縁 OUT (Out1)	未使用
5	灰	光絶縁 I/O グランド	0V (CamPower-)
6	白	カメラ電源接地	0V (CamPower-)

注記

ace または Cognex CIC-A カメラにはストロボ出力がないため、TrigOut と SYNCIN の間にジャンパーが必要です。

© Copyright 2025 ABB. All rights reserved.

2.3.4 カメラの要件

取り付け

振動やその他の動的な動きを避けるために、カメラは非常に安定した方法で取り付ける必要があります。 カメラは、画像エリアに対して任意の方向に取り付けることができます。

照明

確実な結果を得るために、画像領域へ均等に照射することが非常に重要です。

カメラのその他の要件

PickMaster カメラは、移動するベルト上の物体の画像を記録するために使用されるため、プログレッシブ スキャン (ノンインターレース) である必要があります。

PickMaster カメラは、電子シャッター制御もサポートする必要があります。 この機能を使用すると、PickMaster PowerPac から露出を設定できます。それ以外の場合は、露出時間をカメラで手動で設定する必要があります。

カメラの構成

一部のカメラの場合、上記の要件を満たすように構成する必要があります。カメラの設定の詳細については、*Cognex*マニュアルおよび*PickMaster Release Notes*を参照してください。

Basler または Cognex CIC-A ギガビット イーサネット カメラに関する具体的な情報については、参考 ページ *10*を参照してください。

レンズに対する推奨事項

ビジョンを使ったシステムを設計する際には、適切な視野（FOV）を得るために、最適なカメラレンズを選択することが重要です。

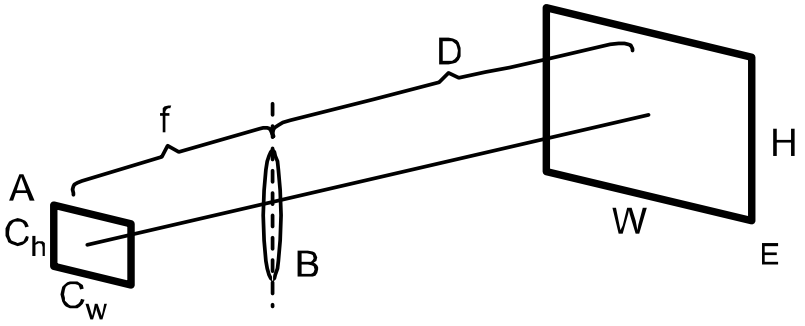
カメラの視野は次の3つの要素によって決まります：

- カメラと対象物の距離.
- レンズの焦点距離。
- カメラのセンサーチップのサイズ（通常はチップの対角距離で表され、インチで表示されます）。

光学的な設定の配置を下図に示します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.



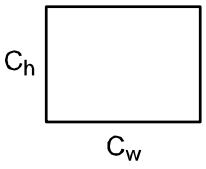
xx0900000550

A	センサーチップ
B	レンズ
Ch	チップ高さ（mm）
Cw	チップ幅（mm）
D	レンズから対象物までの距離（mm）
E	対象物
f	カメラの焦点距離（mm）
H	対象物高さ（mm）
W	対象物幅（mm）

適切なレンズを選択するには、カメラとアイテム（D）間の距離と画像領域のサイズ（W*H）を測定します。

レンズの適切な焦点距離を計算するには：

- 画像領域の高さが最も重要な場合には：`f = (D/W)*Cw`
- 画像領域の長さが最も重要な場合には：`f = (D/W)*Cw`



xx0900000565

下表に、一般的なセンサーチップのサイズの高さと幅をミリメートルで示します。

センサーチップのサイズ(インチ)	C_h (mm)	C_W (mm)
1/4"	2.4	3.2
1/3"	3.6	4.8
1/2"	4.8	6.4
2/3"	6.6	8.8

焦点距離が短いほど視野は広くなります。つまり、特定のW （幅）とH（高さ） を得るには、計算値よりも焦点距離の短いレンズを選定します。

例：レンズの計算

この例は、1/2 インチのセンサー チップに基づいています。

- 視野幅500mmのコンベヤベルトをカバーしなければなりません。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

2.3.4 カメラの要件
続き

- 視野の最低高さは制限されていません。
- カメラとコンベヤの距離は800mmです.
- カメラは底をロボットに向けて取り付けられています（PickMasterのデフォルト設定）.

最小視野はコンベヤの最小幅によって決まるので、必要な焦点距離は下記の式によって与えられます：

```
f = (D/W)*Cw
```

既知のデータを入力します。C_W は6.4mmです（上図参照）。

```
f = (800/500)*6.4 = 10.24 mm
```

得られる視野の高さは次のように計算されます：

```
H = D*CH/f = 800*4.8/10.24 = 375 mm
```

高さを増した場合の代替値

センサーチップの高さ寸法（4.8mm）がコンベヤの幅寸法と一致するようにカメラを90度回転させることによって、視野の高さ（H）を増すことができます。幅寸法（6.4mm）はコンベヤの x 軸と一致します。

```
f = (800/500)*4.8 = 7.68 mm
```

得られる視野の高さは次のようになります：

```
H = 800*6.4/7.68 = 666 mm
```

つまり、レンズは複数の標準焦点距離で使用できるのです。対象物全体を確実に取得するために、計算した値よりも短い焦点距離のレンズを選択してください。

© Copyright 2025 ABB. All rights reserved.

2.4 PickMaster PowerPac ライセンス

ライセンスについて

PickMaster PowerPacを実行するには、ABB が提供するライセンス アクティベーション キーをインストールしてアクティベートする必要があります。

PickMaster PowerPacは、RobotStudio のアクティベーションに依存します。PickMaster PowerPacオプションを含むライセンスでRSをアクティベートした場合のみ、PickMaster PowerPacを通常通りご利用いただけます。 PickMaster PowerPacとは別にアクティブ化することもできますが、それでも RS アクティブ化手順が呼び出されます。

さまざまな PickMaster PowerPac ライセンス オプションでサポートされる機能

PickMaster PowerPac には、Free、Basic、Premium の 3 つのライセンス オプションがあります。

ユーザーは Free オプションを無料で入手し、限られた機能で作業できます。Free オプションでは、既存のソリューションの調整とシミュレーション、プロセスでコンポーネントの編集、新しいレシピの追加のみが可能です。

Robotstudio ライセンスが利用可能な場合、ユーザーは追加料金なしで Basic オプションを入手できます。Basic オプションでは、既存のソリューションの調整とシミュレーション、プロセスですべてのコンポーネントの編集と追加が可能です。

Premiumオプションでは、既存のソリューションの編集、新しいソリューションの作成、プロセスとレイアウトのすべてのコンポーネントの編集と追加など、プロのインテグレータやコミッショナ向けの機能がさらに提供されます。

ライセンスオプション間の機能比較

次の表に、主なアプリケーション シナリオと 3 つのライセンス オプションの違いを示します。

機能		Premium		Basic		Free	
		Robotstudio ライセンスがアクティベートされます	PickMaster PowerPac ライセンスがアクティベートされます	Robotstudio ライセンスがアクティベートされます	PickMaster PowerPac ライセンスがアクティベートされません	Robotstudio ライセンスがアクティベートされません	PickMaster PowerPac ライセンスがアクティベートされません
オープンなソリューション		x		x		x	
レイアウト編集 (コントローラ、グリッパー、コンベア...)		x					
プロセス編集 (アイテム/コンテナ/フロー/レシピ)		x		x		x	
新しいアイテムの作成		x		x		x	
新しいコンテナの作成		x		x		x	
新しいフローの作成		x		x			
新しいレシピの作成		x		x		x	
Pack&Go		x		x		x	

次のページに続く

© Copyright 2025 ABB. All rights reserved.

2.4 PickMaster PowerPac ライセンス
続き

機能		Premium		Basic		Free	
		Robotstudio ライセンスがアクティベートされます	PickMaster PowerPac ライセンスがアクティベートされます	Robotstudio ライセンスがアクティベートされます	PickMaster PowerPac ライセンスがアクティベートされません	Robotstudio ライセンスがアクティベートされません	PickMaster PowerPac ライセンスがアクティベートされません
ポジションジェネレータ		x		x		x	
カメラ設定/キャリブレーション/ライブビデオ		x		x		x	
テンプレートとして圧縮		x		x		x	
保存		x		x		x	
Save as		x		x		x	
運転スタート (生産/シミュレーション/エミュレーション)	生産	x		x		x	
	シミュレーション	x		x			
	エミュレーション	x		x		x[i]	
運転停止		x		x		x	
調整		x		x		x	
ソリューションの作成		x					
解凍		x		x		x	
キャリブレーション		x		x		x	
リセット		x		x		x	
録音		x		x		x	
コントロール スタート (生産/シミュレーション/エミュレーション)	生産	x		x		x	
	シミュレーション	x		x			
	エミュレーション	x		x		x[i]	
RTに接続する		x		x		x	
言語切り替え		x		x		x	
手動		x		x		x	

[i] ロボットのみが移動します。

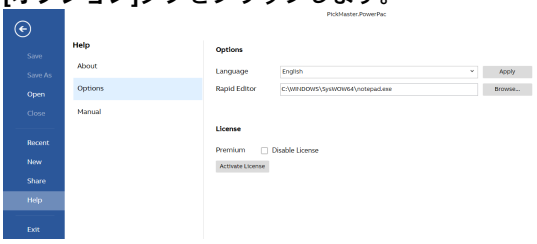
現在のライセンス情報

現在のライセンスに関する情報を取得するには、次の手順を使用します。

	操作	注記
1	PickMaster PowerPacを開始します。	
2	**[ファイル]**タブをクリックします。	
3	**Help**をクリックします。	

次のページに続く

© Copyright 2025 ABB. All rights reserved.

	操作	注記
4	[オプション]タブをクリックします。  xx2300001444	

ライセンスキーのアクティブ化

インターネットを使用したライセンスキーの自動アクティブ化

インターネットを使用してライセンスキーを自動的にアクティブ化するには、次の手順に従ってください。

	操作
1	ライセンス アプリケーションを起動するには、次のいずれかの方法に従ってください。 • PickMaster PowerPac の **オプション** メニューで、**ライセンスのアクティブ化** をクリックします。
2	**スタンドアロン ライセンス** で、**スタンドアロン ライセンス キーをアクティブにしたい** を選択し、**次へ** をクリックします。
3	**自動アクティベーション** で、**RobotStudio をインターネット経由でアクティベート** を選択し、**次へ** をクリックします。
4	25 文字の有効化キー（xxxxx-xxxxx-xxxxx-xxxxx-xxxxx）を入力し、**[Next（次へ）]** をクリックします。 有効化リクエストはインターネット経由で ABB に送信されます。 有効なアクチベーションキーを使用している場合には、PickMaster PowerPac ライセンスが即座にアクティブ化され、次回起動時にPickMaster PowerPac が使用できるようになります。

ライセンス キーを手動でアクティブ化します

PickMaster PowerPac をインストールしたコンピュータにインターネット接続機能がない場合には、ライセンスを手動でアクティブ化しなければなりません。この手順は次の3 つのステップで行います：

1 ライセンスリクエストファイルを作成する（`*.licreqx`）。
2 インターネットに接続しているコンピュータを使用して、ライセンスファイル（`*.bin`）をダウンロードします。
3 ライセンスファイル（`*.bin`）をインストールします。

PickMaster PowerPac ライセンスキーを手動でアクティブ化するには、次の手順に従ってください。

	操作
1	ライセンス アプリケーションを起動するには、次のいずれかの方法に従ってください。 • PickMaster PowerPacの **オプション** メニューで、**ライセンスの確認** をクリックします。
2	ライセンスアプリケーション内で、**PickMaster ライセンス アクティベーション ウィザード....**をクリックします。
3	［Automatic Activation］の下で［Step 1：**Create a license request file**］を選択し、［Next］をクリックします。.

次のページに続く

© Copyright 2025 ABB. All rights reserved.

2.4 PickMaster PowerPac ライセンス
続き

	操作
4	25 文字の有効化キー（xxxxx-xxxxx-xxxxx-xxxxx-xxxxx）を入力し、**[Next（次へ）]** をクリックします。
5	［Save Request］をクリックします。.
6	ライセンスリクエストファイル（*.licreq）の名前を入力し、適切なフォルダに行き、［保存］をクリックします。
7	［完了］をクリックします。
8	USBデバイスなどのリムーバブル・メディアを使用して、インターネットに接続されたコンピュータにライセンスリクエストファイルを転送します。
9	インターネットに接続されたコンピュータ上で、インターネットブラウザを起動して［http://www.manualactivation.e.abb.com/］のリンクに行き、指示に従ってライセンスを手動でアクティブ化します。保存したライセンスリクエストファイルを参照するように指示が出ます。 次にライセンスファイル（*.bin）を任意の場所に保存します。
10	ライセンスファイルをPickMaster PowerPac のPCに転送します。
11	PickMaster PowerPac コンピュータで、ライセンス アプリケーションを起動します。
12	［Automatic Activation］の下で［Step 3：**Install a license file (*.bin)**］を選択し、［Next］をクリックします。.
13	ウィザードの指示に従います。 PickMaster ライセンスがPickMaster PowerPac とRuntimeに対してアクティブ化され、PickMaster インストールが使用できるようになります。

© Copyright 2025 ABB. All rights reserved.

2.5 自己署名証明書

2.5.1 証明書の取り扱い

デフォルトの自己署名証明書

PickMaster® Twin製品は、ネットワーク上での安全な通信のために、 X.509証明書の使用をサポートします。 PickMaster® Runtimeは、デフォルトで自己署名X.509証明書をPickMaster® PowerPacおよびPickMaster® Operatorに対して生成します。生成された自己署名証明書は、キー長2048ビットのRSAキーペアを有します。

証明書の交換

システムのセキュリティを強化し、データが安全な接続を介して送信されていることを保証するために、PickMaster® Runtimeの自己署名証明書を独自の証明書への変換が推奨される。これにより、セキュリティが強化され、独自の信頼できる証明書チェーンが使用可能になる。

自己署名証明書を置き換えるには、必要な証明書とプライベートキーをエクスポートし、PickMaster® Runtimeで証明書を変換します。シームレスで安全な通信を確保するためには、証明書の変換を適切な手順で行うことが重要である。

© Copyright 2025 ABB. All rights reserved.

2.5.2 PickMaster® Runtimeデフォルトの証明書を自己署名証明書に変換

プロシージャ

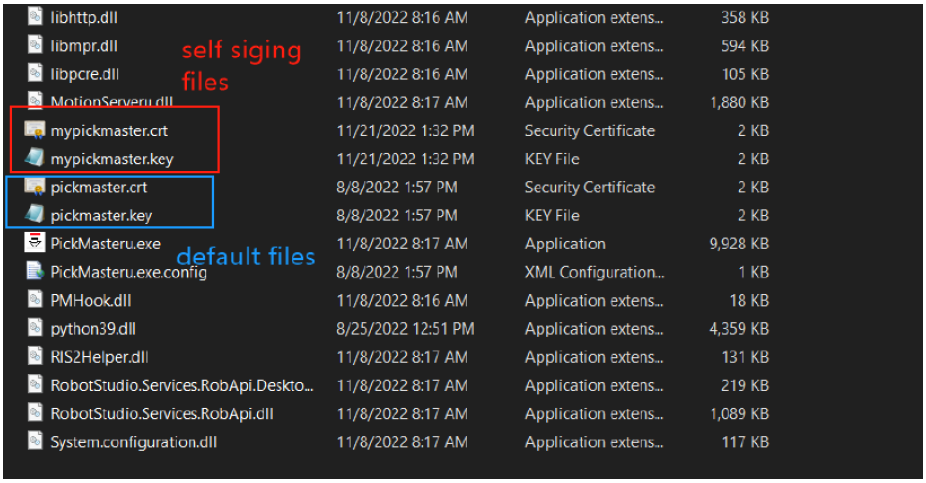
以下の手順を使用して、PickMaster® Runtimeデフォルトの証明書を自己署名証明書に変換します。

1 インストールパスに移動PickMaster® Runtime。

仮想Runtimeの場合: `C:\Program Files (x86)\ABB\PickMaster Twin 3\PickMaster Twin Runtime 3\PickMaster VirtualRuntime`

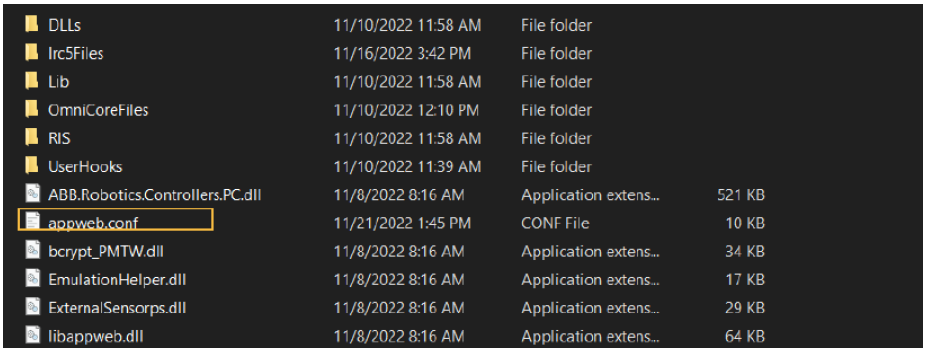
実際のRuntimeの場合: `C:\Program Files (x86)\ABB\PickMaster Twin 3\PickMaster Twin Runtime 3\PickMaster Runtime`

2 自己署名証明書`.key`および`.crt`ファイルをインストールパスに追加。たとえば、赤色表示のファイル名は自己署名証明書ファイルである。



xx2300000747

3 ダブルクリックして`appweb.conf`ファイルを開く。



xx2300000748

4 `SSLCertificateFile`および`SSLCertificateKeyFile`を自己署名証明書ファイルに変更。

次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2300000749

5 管理者として変更を保存。

© Copyright 2025 ABB. All rights reserved.

2.5.3 OpenSSLによる自己署名証明書の生成

概要

ユーザーが自己署名証明書ファイルを生成する必要がある場合、OpenSSLで生成すが推奨される。

プロシージャ

以下の手順で、OpenSSL を使って自己署名証明書を生成。

1 プライベートキーを生成。

```
C:\Program Files\OpenSSL-Win64\bin>openssl genrsa -des3 -out mypickmasterwithpassword.key 2048
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
```

xx2300000740

2 .keyファイルに対してパスワードを入力し、確認。

```
C:\Program Files\OpenSSL-Win64\bin>openssl genrsa -des3 -out mypickmasterwithpassword.key 2048
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
```

xx2300000741

3 証明書署名要求（CSR）を生成。

> **ヒント**
>
> 生成されたプライベートキーを呼び出す鍵の名前を、.csrファイルに入力。

```
C:\Program Files\OpenSSL-Win64\bin>openssl req -new -key mypickmasterwithpassword.key -out mypickmaster.csr
Enter pass phrase for mypickmasterwithpassword.key:
```

xx2300000742

4 必須およびオプション情報を入力。

```
C:\Program Files\OpenSSL-Win64\bin>openssl req -new -key mypickmasterwithpassword.key -out mypickmaster.csr
Enter pass phrase for mypickmasterwithpassword.key:
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:CN
State or Province Name (full name) [Some-State]:Shanghai
Locality Name (eg, city) []:Shanghai
Organization Name (eg, company) [Internet Widgits Pty Ltd]:ABB
Organizational Unit Name (eg, section) []:Info Technology
Common Name (e.g. server FQDN or YOUR name) []:PickMasterTwin
Email Address []:Default@cn.abb.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
```

xx2300000743

> **ヒント**
>
> ユーザーは自分の条件に基づいて情報を入力可能。

5 生成されたプライベートキーファイルを、パスワードなしの新しいプライベートキーファイルに変換。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

```
C:\Program Files\OpenSSL-Win64\bin>openssl rsa -in mypickmasterwithpassword.key -out mypickmaster2.key
Enter pass phrase for mypickmasterwithpassword.key:
writing RSA key
```

xx2300000744

注記

新しい.keyファイル用に生成されたプライベートキーを呼び出す鍵の名前を入力。

```
C:\Program Files\OpenSSL-Win64\bin>openssl rsa -in mypickmasterwithpassword.key -out mypickmaster2.key
Enter pass phrase for mypickmasterwithpassword.key:
writing RSA key
```

xx2300000745

6 自己署名証明書を生成。

```
C:\Program Files\OpenSSL-Win64\bin>openssl x509 -req -days 365 -in mypickmaster.csr -signkey mypickmaster2.key -out mypickmaster.crt
```

xx2300000746

パスワードなしの新しいプライベートキーファイルと、生成された自己署名証明書.crtファイルが最終出力である。

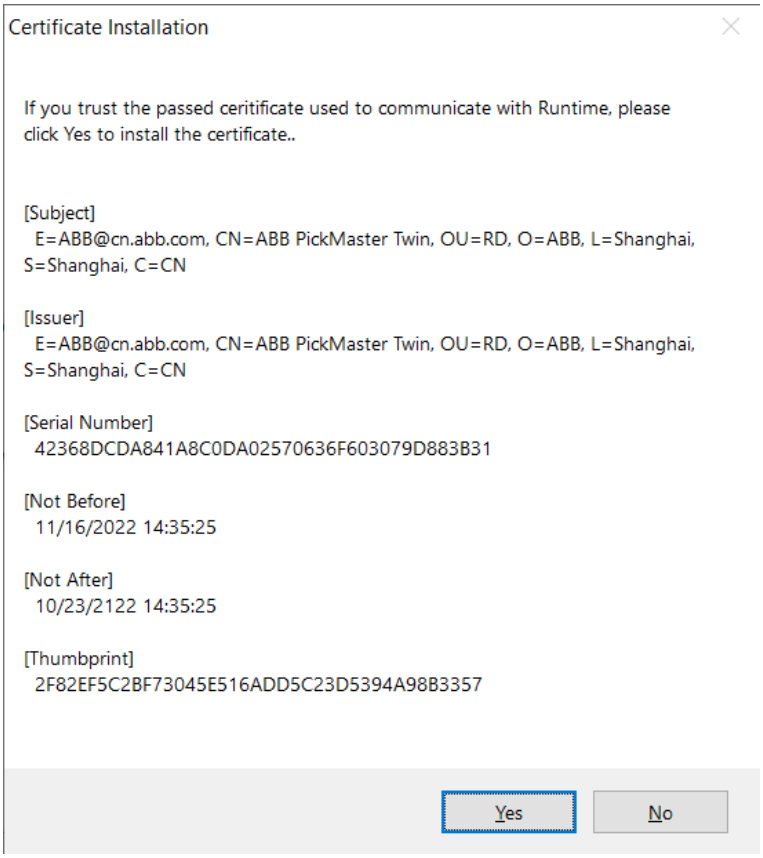
© Copyright 2025 ABB. All rights reserved.

2.5.4 自己署名証明書のインストール

プロシージャ

以下の手順で、自己署名証明書をインストールします。

1 PickMaster® Twin製品を初めて使用する場合、以下のダイアログボックスがポップアップ表示される。



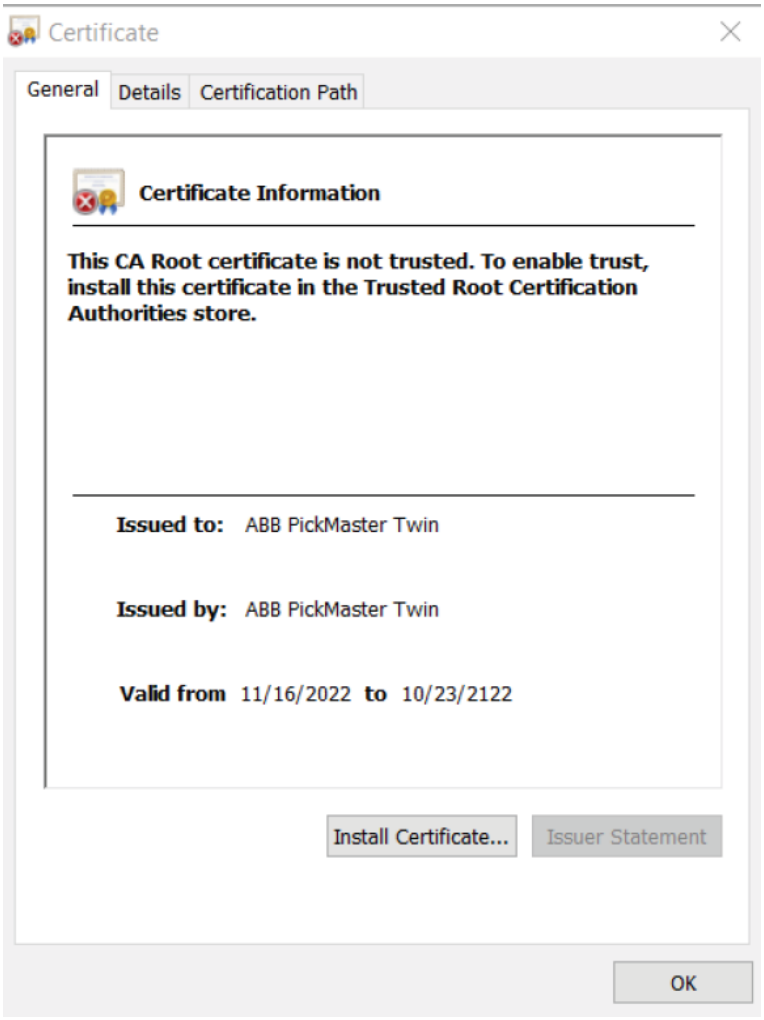
xx2300000751

2 自己署名証明書をインストールするには、 はい をクリックします。

3 OKをクリックしてインストールを終了します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2300000752

© Copyright 2025 ABB. All rights reserved.

2.6 PickMaster時刻同期化サービス

時間同期サービス

PickMaster Twinは、ロボットコントローラおよび HostPickMaster 稼動 PC 間の時刻を同期するために、時刻同期サービスを使用している。同期は、PickMaster Runtimeおよびロボットコントローラ間の通信に使用されるのと同一ネットワーク上で実行される。

IRC5(RobotWare 6)コントローラにはPickMaster Twin PTP v1を使用しています。

OmniCore(RobotWare 7)コントローラにはPickMaster Twin PTP v2を使用しています。

注記

グランドマスターが変更されるたびに、Runtimeがローカル エリア ネットワークの時刻を再同期するのに数秒または数分かかります。この間、ロボットが停止する場合があります。

注記

時刻同期サービスを有効にするには、ユーザーは、PickMaster Twin Client のインストール中に、実際のコントローラに接続されているローカル IP アドレスを選択する必要があります。

コンピュータがまだ実際のコントローラに接続されていない場合は、インストール後に IP アドレスを設定することもできます。 詳細については、*PickMaster Runtimeでのローカル IP アドレスの設定 ページ 179* を参照してください。

ヒント

現在のローカル エリア ネットワーク内のすべてのデバイスの PTP ステータスの概要を把握するには、PTPTrackHound を使用することをお勧めします。

PTPTrackHound ツールをダウンロードするには、*www.ptptrackhound.com* を参照してください。

設定

同期サービスは、精密時間プロトコル (PTP) に基づいており、IEEE 1588-2002 PTP v1 (RobotWare 6 の場合)/1588-2008 PTP v2 (RobotWare 7 の場合) 標準を実装しています。 このプロトコルは、UDP/IP を介したマルチキャスト メッセージを使用し、UDP ポート 319 および 320 が (受信トラフィックと送信トラフィックの両方で) 利用可能である必要があります。 したがって、ファイアウォールがこれらのポートをブロックしていないことが必要です。 システム管理者に連絡して、適切な設定が実行されていることを確認してください。

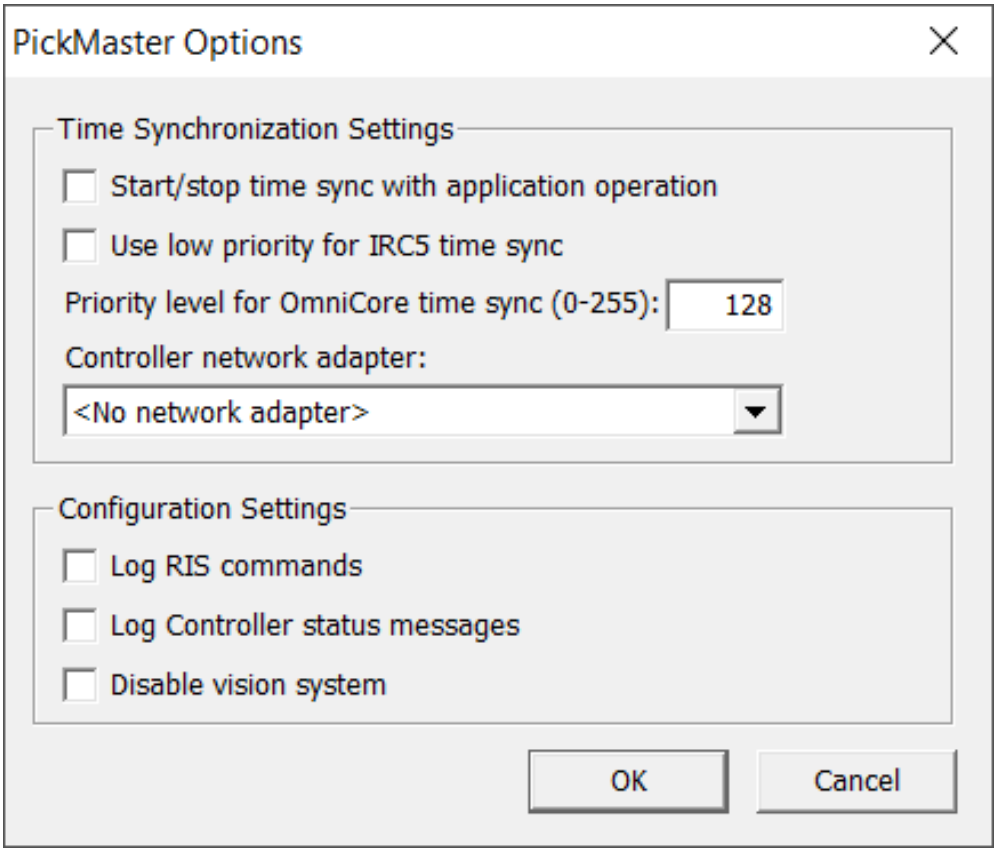
PTP はもともと IEEE 1588-2002 標準で定義され、正式には「ネットワーク化された測定および制御システム用の高精度クロック同期プロトコルの標準」と題され、2002 年に公開されました。2008 年に、IEEE 1588-2008 が改訂標準としてリリースされました。 PTP バージョン 2 とも呼ばれ、精度、精度、堅牢性が向上していますが、元の 2002 バージョンとの後方互換性はありません。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

時刻同期サービスは、正しい PC ネットワーク インターフェイス ポート、つまりロボット コントローラと通信するネットワーク ポートで動作するように設定する必要があります。

時刻同期設定インターフェイス



xx2300001442

No.	名前	値
1	開始/停止時刻をアプリ操作と同期	チェックボックスが次の場合: • チェックあり：Runtimeの起動/停止に連動して、PTPサーバーも自動的に起動/停止します。 ヒント 自動的に再起動される PTP サーバーは、最後に使用されたものです。 IRC5(RobotWare 6)とOmniCore(RobotWare 7)のPTPサーバーは独立して存在します。 • チェックなし：Runtimeの起動/停止に連動して、PTP サーバーは自動的に起動/停止しません。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

No.	名前	値
2	**IRC5 時刻同期に低い優先度を使用**	IRC5(RobotWare 6)コントローラにはPickMaster Twin PTPv1を使用しています。 PTPv1 戦略には、高優先度と低優先度の 2 つの優先順位があります。 時刻同期サーバーは、優先度の高いデバイスをグランドマスター デバイスとして使用します。 ローカル エリア ネットワーク内に優先度の高いデバイスが存在しない場合、または優先度の高いデバイスが複数存在する場合、より優れたハードウェアを備えたデバイスが自動的にグランドマスターとして設定されます。 チェックボックスが次の場合: • チェックあり：現在のデバイスは低優先度に設定されています。 **ヒント** 自動的に再起動される時刻同期サーバーは、最後に使用されたサーバーです。 • チェックなし：現在のデバイスは高優先度として設定されます。 優先順位の値はレジストリに保存されます。
3	**OmniCore 時刻同期の優先レベル (0 ~ 255)**	OmniCore(RobotWare 7)コントローラにはPickMaster Twin PTPv2を使用しています。 PTPv2 の priority1 値を使用して、時刻同期サービスのデバイスの優先順位を制御します。 範囲は 0 ~ 255 です。値が小さいほど優先度が高くなります。ローカル エリア ネットワーク内で最も優先度の高いデバイスはグランドマスター デバイスになります。 デフォルト値は128です。 priority1 の値がローカル エリア ネットワーク内の複数のデバイスで同じ値に設定されている場合、より優れたハードウェアを備えたデバイスが自動的にグランドマスターとして設定されます。 優先順位の値はレジストリに保存されます。
4	**コントローラ ネットワーク アダプター**	この IP は、コントローラに接続するデバイスの IP アドレスとして設定する必要があります。

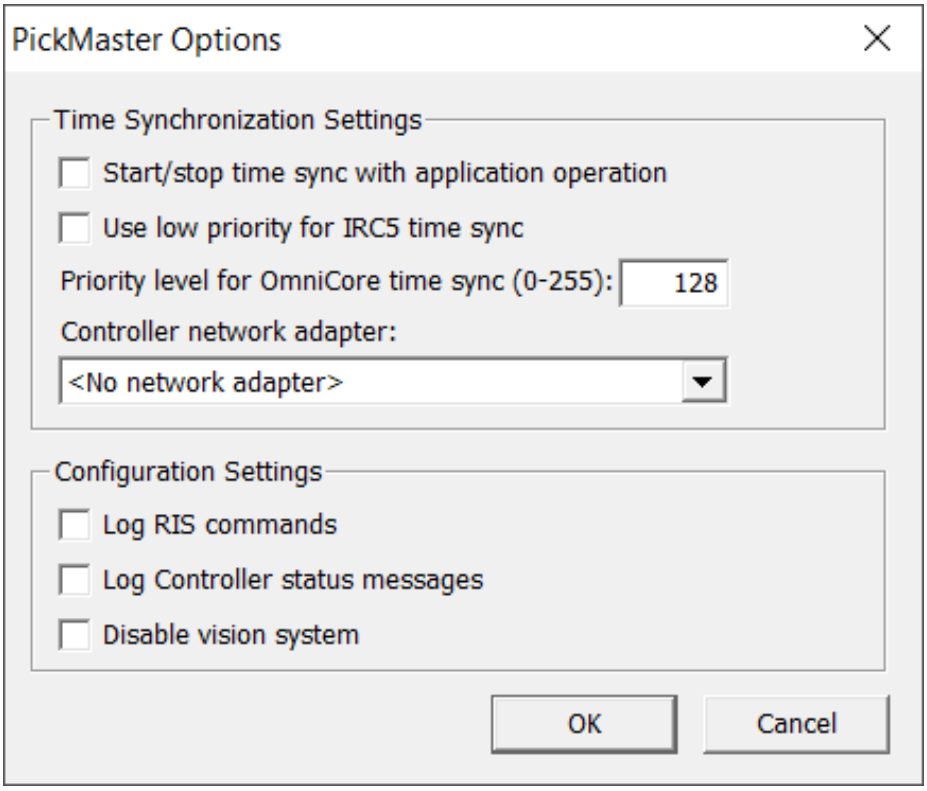
PickMasterのオプション設定

下記の手順に従ってPickMasterのオプションを設定してください.

1 [File（ファイル）] メニューで [Option（オプション）] を選択します。
[PickMasterオプション] ダイアログが開きます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2300001442

2 **アプリケーション操作と同期する開始/停止時刻** ボックスで、Runtimeの開始/停止に連動して PTP サーバーを自動的に開始/停止するかを選択します。

3 PTP v1 サーバーの現在のデバイスの優先順位を変更するには、**IRC5 時刻同期に低い優先順位を使用** ボックスから選択します。

4 **[OmniCore 時間同期の優先レベル (0 ~ 255)]** テキスト ボックスで、PTP v2 サーバーの現在のデバイスの priority1 の値を定義します。

> **ヒント**
>
> 値が小さいほど優先度が高くなります。

5 ［**コントローラネットワークアダプタ**］から、ロボットコントローラとの通信を行うネットワークアダプタのIPアドレスを選択します。

6 必要に応じて［**ログRISコマンド**］を 選択して、ログエリアにすべてのRISコマンドを表示します。

7 必要に応じて［**ログ IRC5 状態メッセージ**］を選択して、 ログエリアに状態メッセージを表示します。

デフォルトでは、ログエリアには警告とエラーメッセージのみが表示されます.

8 PickMaster の内部ビジョン システムを使用しない場合は、**ビジョン システムを無効にする**を選択します。その後、PickMaster は接続されているカメラに接続しなくなります。これは、Basler カメラが外部センサー インターフェイスを通じて使用される場合の競合を回避するのに役立ちます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

9 [OK] をクリックします。

© Copyright 2025 ABB. All rights reserved.

2.7 ソフトウェアのインストール



注記

ABB ロボットの設置作業を行う人は、ABB によるトレーニングを受け、機械的および電気的インストール作業に必要な知識を持っている必要があります。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

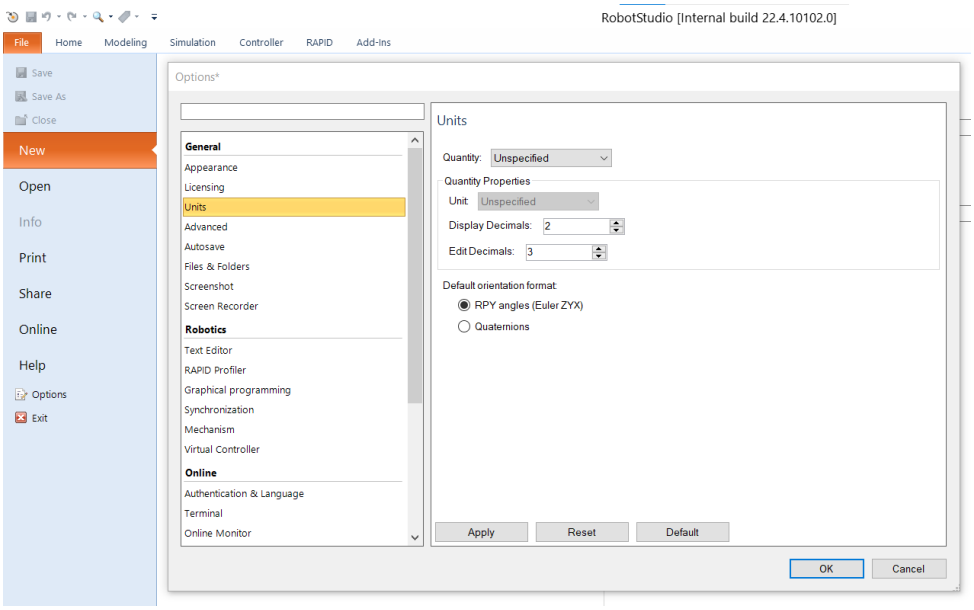
2.7.1 RobotStudio のインストール

命令

詳細な RobotStudio インストール手順については、以下を参照してください。
ユーザーマニュアル - *RobotStudio*、*3HAC032104-012*。

注記

RobotStudio でユニットを設定する場合は、Unspecified をお勧めします。
他のユニットは、PickMasterに不確定なエラーを引き起こす可能性があります。



xx2200002058

© Copyright 2025 ABB. All rights reserved.

2.7.2 PickMaster Twin Clientのインストールとアンインストール

概要

このセクションでは、PickMaster Twin Clientのインストール プロセスについて説明します。

> **注記**
>
> PickMaster 3 と PickMaster Twin Client を同じ PC にインストールすることはお勧めしません。
> それらは互いに影響を与える可能性があります。

> **注記**
>
> PickMaster Twin Client と PickMaster Twin Host を同じ PC にインストールすることはお勧めしません。

> **注記**
>
> PickMaster Twin Client をインストールする前に、コンピュータに RobotStudio がインストールされていることを確認してください。 RobotStudio のインストール手順については、ユーザーマニュアル - *RobotStudio* を参照してください。

PickMaster Twin Clientのインストール

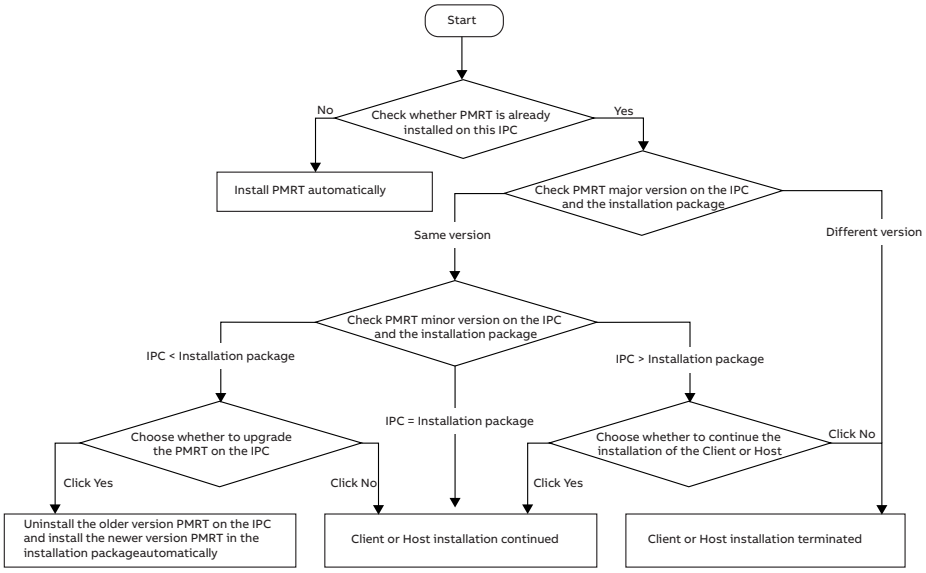
次の手順を使用して、PickMaster Twin Client をインストールします。

1. PickMaster Twin Client インストール パッケージを参照し、`Setup.exe` をダブルクリックします。
   インストールが開始されます。
2. ［次へ］をクリックします。
3. 使用許諾契約を読み、条件に同意します。
4. ［次へ］をクリックします。
5. PickMaster Twin Clientは、**Congnex ビジョン ドライバー**がこのコンピュータに既にインストールされているかどうかを自動的に確認します。
   インストールされてない場合は、**Congnex ビジョン ドライバー**が自動的にインストールされます。
6. PickMaster Twin Clientは、次のフローチャートに従ってPickMaster Runtimeのインストールを確認します。

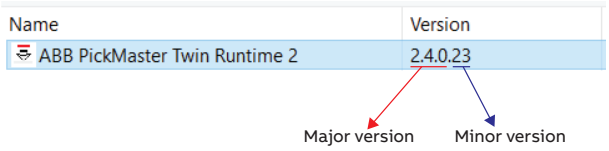
次のページに続く

© Copyright 2025 ABB. All rights reserved.

2.7.2 PickMaster Twin Clientのインストールとアンインストール
続き



xx2400000341



xx2400000342

7 インストールを続行する場合は、次へをクリックします。
8 次へ をクリックしてインストールを開始します。
9 インストールが完了したら、コンピュータを今すぐ再起動するか後で再起動するかを選択し、完了 をクリックします。

PickMaster Twin Clientのアンインストール

PickMaster Twin Clientをアンインストールするには、次の手順を使用します。

1 右クリックして、コントロール パネルでPickMaster Twin Clientでアンインストールを選択します。
2 必要に応じて、Cognex ドライバーのアンインストールチェックボックスを選択して、コンピュータから cognex ドライバーをアンインストールします。
3 ［次へ］をクリックします。
4 はいをクリックしてアンインストールを開始します。

> **注記**
>
> PickMaster Twin Clientのみが存在する場合、PickMaster Runtimeは同時に自動的にアンインストールされます。
>
> PickMaster Twin ClientとPickMaster Twin Hostの両方が IPC 上に存在する場合、PickMaster Runtimeはアンインストールされません。

5 アンインストールが完了したら、完了 をクリックします。

© Copyright 2025 ABB. All rights reserved.

2.8 電気接続

2.8.1 ネットワークの設定

コントローラネットワークについて

PickMaster PowerPacとロボットコントローラは、イーサネットにより通信します。ネットワークに接続できない場合は、ローカルネットワーク管理者に連絡してください。

> **注記**
>
> PickMaster PowerPacは、コントローラの WAN ポートに接続する必要があります。 サービス ポートは使用しないでください。

コントローラネットワークの構成

PickMaster PowerPac を使用するために新しくローカルエリアネットワーク（LAN）を設定する場合は、次の設定を使用することができます。

- コンピュータとロボットコントローラの両方に対して、異なるアドレスを採用して、静的 IP ナンバリングを使用する.
- IPアドレス：192.168.1.X（Xは 1 ~ 253の数字）
- サブネットマスク： 255.255.255.0
- ゲートウェイ： 192.168.1.254
- DNS：適用なし
- Wins：適用なし

> **注記**
>
> ロボットコントローラには、IP アドレス（192.168.125.1）構成済みのサービスイーサネットカードが付属しています。したがって、標準 LAN イーサネットボードに同じサブネット（192.168.125.X）を使用することはできません。
>
> 詳細については、*Windows* のドキュメンテーションとロボットコントローラの製品マニュアルで IP 構成の設定を参照してください。

> **注記**
>
> 他の機能に割り当てられている次の IP アドレスは使用できません。
>
> - 192.168.127.0 - 255
>
> IP アドレスは、上記の予約済みIPアドレスのいずれかと重複するサブネット上には存在できません。クラスB範囲のサブネットマスクを使用する必要がある場合、クラスBのプライベートアドレスを使用して、重複を回避しなければなりません。ネットワークの重複については、ローカルネットワーク管理者にお問い合わせください。
>
> *Technical reference manual - System parameters*のセクション*Communication* を参照してください。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

2.8.1 ネットワークの設定
続き

ビジョンネットワークの前提条件

ビジョンネットワークの設定には、ロボットコントローラ ネットワークの設定に類似した構成を適用する必要があります。

ビジョンシステムには別のネットワークを使用します。つまりコントローラとカメラを PC の同じネットワークポートに接続することはできません。

PC で使用可能なイーサネット ポートの数よりも多くのカメラを使用するには、追加の GigE カードを 1 つまたは 2 つ使用します。

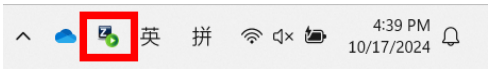
単体の PC に追加できるカメラの最大数は 10 です。PC のビジョンネットワーク専用ポートに繋がれます。。GigE カードおよびカメラの間は、固定ネジを有する付属ケーブルを使用します。ネットワークアーキテクチャの事例については、適切なネットワークアーキテクチャの例 ページ 57を参照。

概要

この章では、インターネットの設定手順について説明します。インターネットの設定をしないと、PackML 機能が正常に動作しません。

> **注意**
>
> ネットワーク アダプタの名前が正しく変更されていないと、PickMaster PowerPac は正常に動作しません。

この手順を使用して、PickMaster PowerPacの次のネットワークを設定します。

> **注記**
>
> zenon のステータスインジケータが赤の場合は、ネットワーク名が「ProfinetIOAdapter」に変更されているかどうかを確認します。
>
> そうでない場合は、名前を変更して問題を解決してください。
>
> 
>
>
> xx2400001789

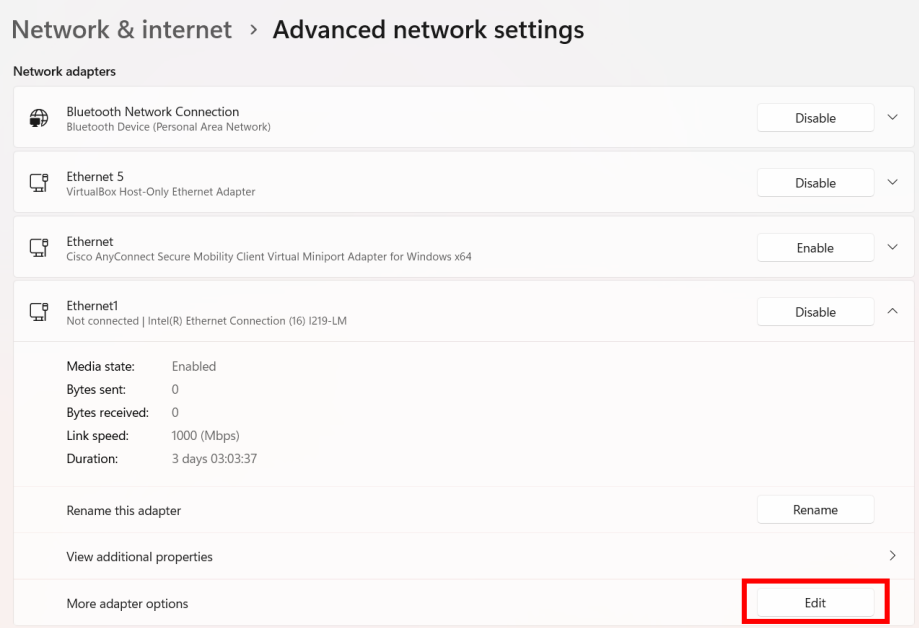
PickMaster Powerpac の IPC ネットワークの設定

カメラが接続されているコンピュータのネットワーク設定を変更するには、次の手順をお勧めします。

1. IPC で **[ネットワークとインターネット]** -> **[詳細ネットワーク設定]** ページを開きます。
2. PickMaster PowerPacの接続に使用するネットワークを開きます。
3. **[編集]** をクリックして、プロパティ ウィンドウを開きます。

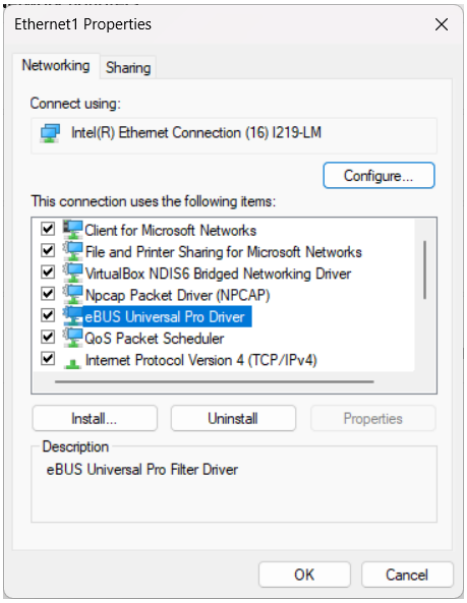
次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2400001792

4 次のプロトコルまたはドライバが選択されていることを確認します:
- **eBUS ユニバーサル プロ ドライバー**
- **インターネット プロトコル バージョン4(TCP/IPv4)**



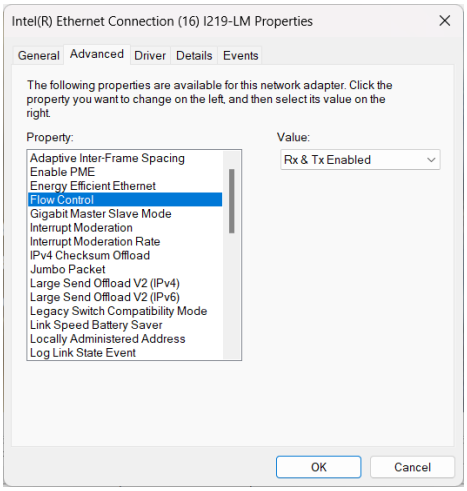
xx2400001793

5 **設定** をクリックし、**詳細設定** タブを選択します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

2.8.1 ネットワークの設定
続き



xx2200002067

6 必要に応じて次のプロパティを変更します。
   - **Jumbo packet** プロパティを選択し、ダイアログ ボックスで可能な限り高い値を選択します。

7 さらに、Cognex では、このネットワーク接続の次のプロパティを変更することをお勧めします。これらのプロパティは、以前のプロパティと一緒にグループ化されている場合とグループ化されていない場合があります。
   - **受信バッファ** プロパティを変更し、**値** リストで可能な限り高い値を選択します。
   - 値リストで、**割り込み加減率** プロパティを **極度** に変更します。
   - **送信バッファ** プロパティを変更し、その値リストで可能な限り高い値を選択します。
   - **フロー制御**プロパティをその値リストで **Rx&Tx** 有効に変更します。

8 ［OK］をクリックします。

必要に応じて変更する必要があるシステム プロパティの詳細については、Gig ビジョン設定ツールに組み込まれている質問と回答を参照してください。

ビジョンネットワークの構成

下記の手順に従ってビジョンネットワークを構築してください。

1 各カメラには独自の IP アドレスを割り当てる必要があります。他のイーサネットと同じ規則が適用されます。たとえば、各カメラとビジョンネットワーク カードは独自の IP アドレスを持ち、同じサブネット上に設定されます。ビジョンネットワークとコントローラネットワークは、異なったサブネットに設定する必要があります。適切なネットワークアーキテクチャの例 ページ 57を参照してください。

2 **Cognex GigE ビジョン設定**を使用して、カメラの IP アドレスを設定します。カメラとネットワーク インターフェース カードの両方の IP アドレスを設定するために使用できます。

3 すべてのカメラの構成が完了したら、各ポートにGigabitイーサネットビジョン用の*Performance Driver*をインストールします。4-6項を参照します。

次のページに続く

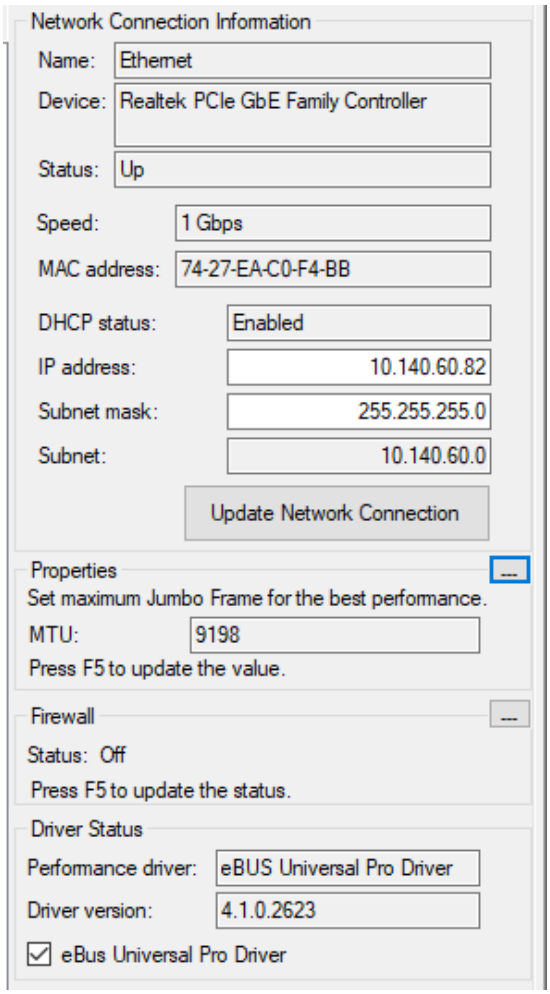
© Copyright 2025 ABB. All rights reserved.

4 イーサネットカメラツールでツリービューの各ビジョンネットワークポートについて、以下の設定を行います。

a プロパティセクションでMTUの値を9000前後に設定します。MTUの値が1500前後の場合は、Jumboフレームが設定されていないことを意味します。

Jumboフレームを設定する場合：

I クリック...



xx2200001607

イーサネットプロパティウィンドウが表示されます。

II ネットワークタブをクリックします。

III [構成] をクリックします。

プロパティウィンドウが表示されます。

IV 詳細設定タブをクリックします。

V プロパティリストからジャンボフレームを選択します。

VI 値ドロップダウンリストから、なるべく高い値を選択します。

VII イーサネットカメラ設定ツールに戻るまで、OK/適用 をクリックします。

VIII F5キーを押すと、ウィンドウ内の値が更新されます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

IX MTU値が約 9000 であることを確認します。

b **eBus Universal Pro Driver**チェックボックスを選択します。署名されていないソフトウェアのインストールに関する警告が表示されます。

c ［OK］をクリックします。

5 すべてのビジョンポートのインストールが完了したら、PCを再起動してください。

6 **イーサネットカメラツール**を起動し、各ビジョンネットワークポートにパフォーマンスドライバが正常にインストールされたことを確認します。また、Jumbo frames MTUの値が約9000に設定されていることを確認します。

注記

撮影中に問題が発生した場合は、カメラが接続されているイーサネットのネットワーク設定を以下のように変更してください。

- ツリービューのビジョンネットワークポートの**イーサネットカメラツール**で以下をクリックします。

  イーサネットプロパティウィンドウが表示されます。

  - **ネットワーク**タブで、**この接続は、eBUS Universal Pro ドライバーとインターネット プロトコル バージョン 4 (TCP/Ipv4)**以外の次の項目を使用しますの下にリストされているすべてのチェック ボックスをオフにします。
  - **設定** をクリックし、**詳細設定** タブを選択します。
    - # **受信バッファ** プロパティを選択し、**値** リストで可能な限り高い値を選択します。
    - # **インタラプトモデレーション率** プロパティを選択し、値を **エクストリーム**として選択します。

注記

イーサネット カメラツール と Runtime を同時に実行すると、予期しない動作が生じることがあります。これを回避するには、一度に 1 つのプログラムのみ使用します。

注意

同じネットワーク上でカメラトラフィックとコントローラトラフィックを実行すると、深刻な通信エラーが発生する可能性があります。

注記

サードパーティのビジョン ソフトウェアがインストールされている場合は、ビジョンネットワークの構成 ページ *54*.に従ってビジョン ネットワーク設定を確認してください。

サードパーティのソフトウェアを使用すると、意図しない設定変更が発生し、カメラが誤動作する可能性があります。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

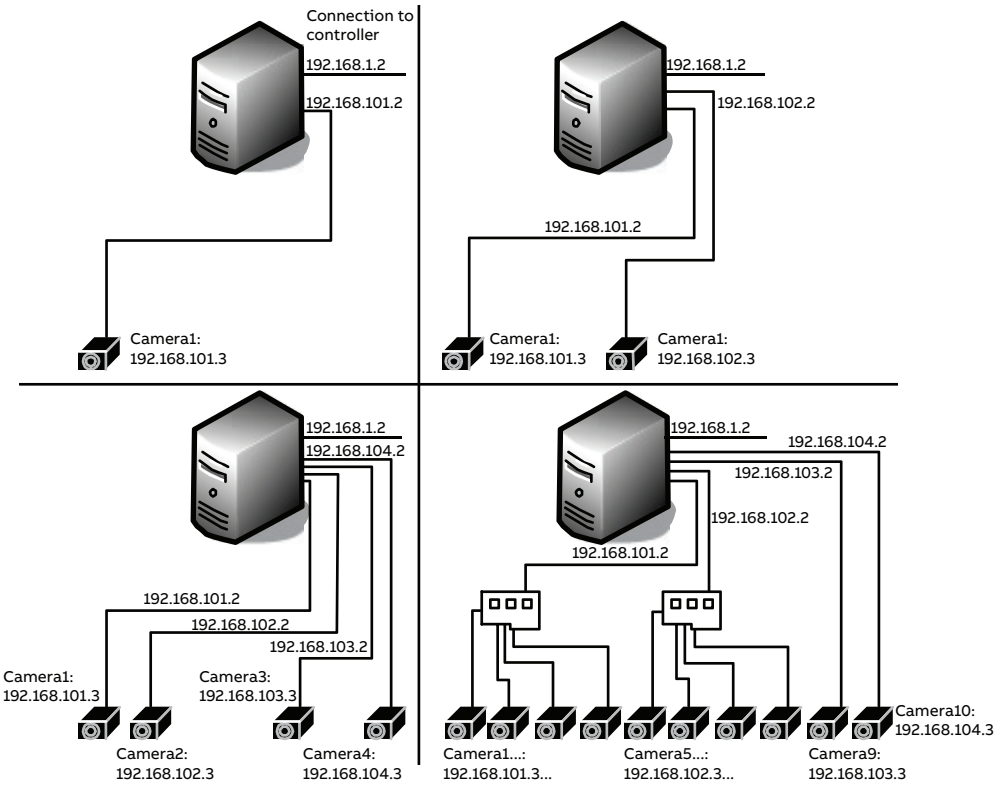
コントローラネットワークの設定

Runtime専用の新しいローカル エリア ネットワーク (LAN) を作成する場合は、次の設定を使用できます。

- PickMaster PowerPacとロボットコントローラに対して、異なるアドレスを採用して、静的 IP ナンバリングを使用します。
- IPアドレス：192.168.1.X（Xは 1 ~ 253の数字）

適切なネットワークアーキテクチャの例

- コンピュータとカメラの両方に対して、異なるアドレスを採用して、静的IPアドレスを使用する.
- ポート#1 のIP アドレスとポート#1 に接続されるカメラ：192.168.101.X（X は 1 ~ 253 の数字）
- ポート#2 のIP アドレスとポート#2 に接続されるカメラ：192.168.102.X（X は 1 ~ 253 の数字）
- サブネットマスク： 255.255.255.0
- ゲートウェイ：必要なし。
- DNS：適用なし
- Wins：適用なし



xx2300001025

次のページに続く

© Copyright 2025 ABB. All rights reserved.

2.8.1 ネットワークの設定
続き

注記

Runtime外でカメラ設定に加えられた変更は、Runtimeが再起動されるまで適用されません。 これは、カメラが再起動された場合 (電源のオン/オフ)、またはカメラの IP アドレスが変更された場合、適切に機能するためにRuntimeを再起動する必要があることを意味します。 したがって、予期しない動作を避けるために、Runtimeとイーサネット カメラ ツール プログラムを同時に実行しないでください。 代わりに、変更を加える前にRuntimeをシャットダウンし、変更を保存した後にRuntimeを開始します。

© Copyright 2025 ABB. All rights reserved.

2.8.2 カメラの接続

カメラの接続について

カメラは、イーサネット ケーブルからは電圧を受け取りません。別個に接続される電源/トリガー/ストロボケーブルにより、電力と I/O 機能が提供されます。

Gigabit イーサネット カメラには外部電源を使用することをお勧めします。 これにより、ロボット コントローラの電源が入っているかどうかに関係なく、電源が供給されます。 カメラにロボット コントローラから直接電源が供給されている場合、コントローラの電源がオフになるとカメラはシャットダウンします。 Runtimeは、シャットダウンして再起動したカメラに再接続できません。 これは、カメラの電源として機能するコントローラがシャットダウンされたときにRuntimeが実行されている場合、コントローラの電源を再度オンにした後でRuntimeを再起動する必要があることを意味します。 この問題は、外部電源を使用することで回避できます。

Gigabit イーサネット カメラには、GigE Ready オプションに含まれる 4 ポート Gigabit イーサネット ボードを使用する必要があります。 また、カメラはコントローラと同じネットワーク カードを使用できません。そうしないと、キャプチャされた画像が影響を受けます。

カメラで使用する場合は、ネットワークカードのジャンボパケット機能を有効にする必要があります。

カメラからのトリガー ストロボと電源ワイヤをロボット コントローラの I/O ボードに接続する方法の回路図は、回路図で確認できます。*Circuit diagram - PickMaster Twin、3HAC024480-020* を参照してください。EMI/ESD 問題の回避に関する詳細情報は、*Avoid_EMI_ESD_in_camera_installations* に記載されています。参考 ページ 10 を参照してください。

> **注記**
>
> コントローラを操作するためのすべての安全情報は、コントローラの製品マニュアルに記載されています。

事前知識

カメラを接続する前に、すべての電源がオフになっていることを確認してください。

検証済みカメラ

PickMaster® Twin は次のカメラをサポートしています。

- Basler Ace acA1440-73gc
- Basler Scout scA1300-32gc
- Basler Scout scA1390-17gm
- Cognex CIC-A02-CG-60-G
- Cognex CIC-A05-CG-24-G

次のページに続く

© Copyright 2025 ABB. All rights reserved.

2.8.2 カメラの接続
続き

注意

短絡の場合の人身傷害の危険性とカメラの損傷の危険性。

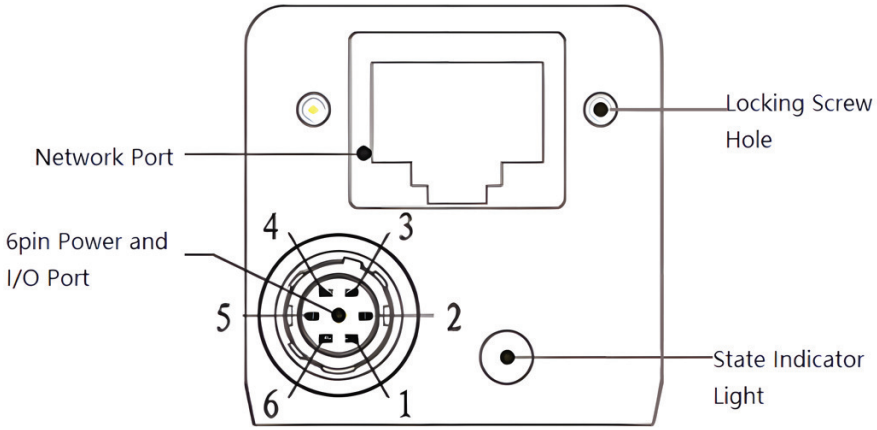
短絡により、カメラのハウジングの温度が極端に上昇する場合があります。 これにより、カメラが損傷する可能性があり、火傷などの人身事故につながる可能性もあります。 最悪の場合、過熱による発火の恐れがあります。

それを防ぐために、短絡時に個々の配線を流れる電流を制限します。 許容される最大電流は2 A です。ヒューズを使用するか、限られた電源を使用してください。

カメラの接続

下記の手順に従ってカメラを接続してください。

1 スクリューコネクタを使用して、イーサネットケーブルをカメラに接続します。



xx2400001601

2 反対側のコネクタをPCまたはスイッチングハブ（使用している場合）に接続します。

3 スイッチングハブを使用している場合には、イーサネットケーブルを使用してPCと接続します。

4 電源/トリガー/ストロボケーブルの電源線を外部電源に接続します。

外部電源を使用しない場合には、コントローラに接続します。

5 電源/トリガ/ストロボケーブルのトリガ/ストロボワイヤをロボットコントローラに接続します。

注記

Runtime をシャットダウンして、素早く再起動した場合、Gigabit イーサネット カメラの機種によっては Gigabit イーサネット パフォーマンス ドライバが正しくロードされないことがあります。ドライバがロードされていないカメラでは、システムに負荷がかかった場合に画像が取得できないという問題が時々生じます。これは、シャットダウンして、15秒待ってから再起動することによって回避できます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

関連情報

Circuit diagram - PickMaster Twin、*3HAC024480-020*.

© Copyright 2025 ABB. All rights reserved.

2.8.3 I/O信号の接続

I/Oの接続について

Runtimeのコンセプトは、物理的に接続する必要がある多数の I/O コンポーネントで構成されています。

ロボットコントローラのI/Oボード

ロボットコントローラには1枚以上のI/Oボードが必要です。コンベヤのトラッキングにはエンコーダーボードが必要です。

エンコーダーボードはDI/DOボードと異なったアドレスで設定されています。このアドレスは変更できます。

エンコーダーボードのアドレスの確認方法は、コントローラの製品マニュアルに記載されています。参考 ページ *10*を参照してください。

事前知識

すべての電源がオフになっていることを確認してください。

I/O信号の接続

下記の手順に従ってI/O信号を接続してください。

1. コンベアを使用する場合は、各コンベア コントローラを標準の DI/DO ボードに接続して、Runtimeから制御します。

   Circuit diagram - PickMaster Twin、*3HAC024480-020* の図面では、ACS 301-1P6-3 をコンベア コントローラとして使用していますが、他のコンベア コントローラを使用することもできます。
2. カメラからの電源/トリガ/ストロボケーブルのトリガ/ストロボワイヤをロボットコントローラに接続します。
3. ロボットハンドのI/Oケーブルをロボットコントローラに接続します。
4. センサーなど、その他の外部デバイスからのI/Oケーブルをロボットコントローラに接続します。

I/Oの接続

トリガ、ストロボ間の処理は、ロボットコントローラと取得画像を精密に同期します。Gigabit イーサネットカメラのI/Oが、この処理を行います。

エンコーダーボード上の入力番号 9（StartSig）で複数の接続を使用できるようにするには、ダイオードの使用が推奨されます。使用できるダイオードの例として HER105/Taw ダイオード1A 400V DO41 （このダイオードはABBによってサポートされていません）があります。このダイオードの使用により、逆方向電流が発生しなくなります。

1 台のカメラを複数のロボットコントローラに接続する場合には、カメラへ電源を供給しているコントローラがオフになった時にシステムがどのように働くのかを考慮することが重要です。カメラには24Vの外部電源を使用することをお勧めします。これにより、コントローラがオフであるかどうかに拘わらず、カメラは電源とI/Oの両方に接続されることになります。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

関連情報

Circuit diagram - PickMaster Twin、3HAC024480-020

I/O 信号 ページ 185。

コンベア作業エリアのデフォルト I/O シグナル ページ 187。

© Copyright 2025 ABB. All rights reserved.

2.8.4 ロボット コントローラの設定

注意

外部制御パネルからロボットの動きを起動できる場合は、緊急停止も利用できる必要があります。

注記

PickMaster TwinがロボットコントローラからアクティブなAS/ESのステータスを取得した場合、キュー内のアイテムはクリアされません。一旦AS/ESが非アクティブ化されると、**Start/Unhold/Unsuspend** のボタンをクリックすることで生産を再開することができます。

RobotWare :

PickMaster PowerPac サポート OmniCoreとIRC5ロボットコントローラ。
RobotWare は、ロボット コントローラにインストールされます。 Runtimeを実行するには、オプション *PickMaster Ready* が必要です。 オプションの詳細については、*PickMaster Twin* 製品仕様を参照してください。

詳細については、コントローラの製品マニュアルを参照してください。参考 ページ *10*.

システムパラメータ

コンベヤの数はシステムパラメータで指定する必要があります。コンベヤに対して、モーション、プロセス、I/O 等、その他のパラメータを定義する必要があります。

システムパラメータはFlexPendantとRobotStudioのどちらでも変更できます。

I/O信号

I/O信号およびボードの構成方法については、「*I/O 信号 ページ 185*」のセクションを参照してください。

事前定義のI/O信号については、「コンベア作業エリアのデフォルト *I/O シグナル ページ 187*」のセクションを参照してください。

関連情報

コントローラの製品マニュアルについては、参考 ページ *10*を参照してください.

『*Technical reference manual - System parameters*』。

*6*軸ロボット構成 ページ *66*。

© Copyright 2025 ABB. All rights reserved.

2.8.5 オプションのロボットとプロセス設定

コンベアプロセスの変更

システム パラメータを変更できます。

トピック Process

トピック「*Process*」では、次のパラメータを変更可能です。これはタイプ *Conveyor systems* に属しています。

パラメータ	説明
maximum distance	コンベヤ ワーク オブジェクトを新しいワーク オブジェクトに切り替えるまでの標準トラッキング距離を規定します。デフォルトでは、20000 mm に設定されています。ワーク オブジェクトの切り替えは自動的に行われ、短時間で終了しますが、高速ピッキング アプリケーションの処理時間がいくらか奪われる可能性があります。値を上げることで、サイクル時間をわずかに改善することができます。

© Copyright 2025 ABB. All rights reserved.

2.8.6 6軸ロボット構成

6軸ロボットの改造

PickMaster を 6 軸ロボットで使用する場合、コンベア トラッキング プロセスでロボットの動作を最適化するために、システム パラメータでいくつかの変更を行う必要があります。

トピックProcess

次の3つのパラグラフは、トピック *Process*で変更できます。これはタイプ *Conveyor systems*に属しています。

パラメータ	説明
Start ramp	これは、動作中のコンベアに同期する際に使用する開始補正フィルタランプです。デフォルトでは、5（ステップ）に設定されています。 より高い精度が求められる場合は、このパラメータを調整します。値を低く設定すると精度が高くなりますが、移動中のオブジェクトに接続する際にマニピュレータが揺れる場合があります。
Stop ramp	これは、動作中のコンベアを非同期する際に使用する停止補正フィルタランプです。デフォルトでは、10（ステップ）に設定されています。 より高い精度が求められる場合は、このパラメータを調整します。 値を低く設定すると精度が高くなりますが、コンベアを非同期する際にマニピュレータが揺れる場合があります。
Adjustment speed	ロボットがコンベヤに追いつく速度（mm / s）。 一般的な推奨値は、コンベヤ速度の130%です。 少なくとも、値は100%を超え、ある程度のマージンが必要です。 ロボットの速度がコンベヤの速度に比べて非常に速い場合は、値をさらに増やす必要があることがよくあります。 設定値が低すぎると、ロボットの動きがぎくしゃくしたり、コンベヤのトラッキング精度が低下したりする場合があります。 一方、設定値が高すぎると、ドライブシステムが過負荷になり、モーション監視エラーが発生する場合があります。 通常、推奨される最大値は200%です。 ロボットの速度が速いアプリケーションのIRB360の場合、推奨される最大値は500%です。

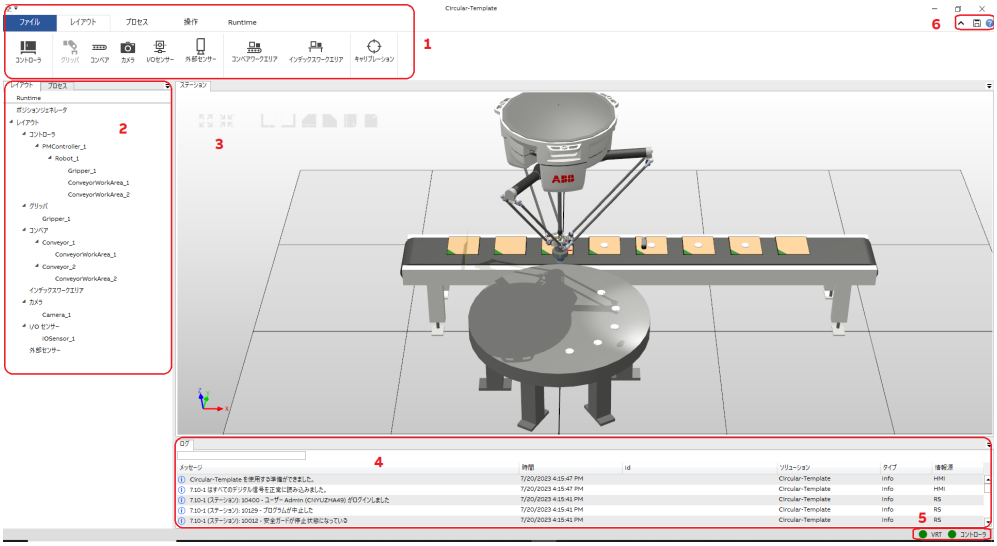
© Copyright 2025 ABB. All rights reserved.

3 PickMaster PowerPac のナビゲート

3.1 メインウィンドウ

概要

この章では、PickMaster PowerPac のユーザ インタフェースについて説明します。次の図と表は、ユーザーインターフェイスの主要な要素に関する情報を示しています。



xx2100000855

		説明
1	リボンタブ	PickMaster PowerPacの一般的な機能が含まれています。新しいソリューションを作成する場合、ワークフローは通常、左から右になります。 詳細については、セクションリボンタブ ページ69を参照してください。
2	ツリー ビュー ブラウザ	ピッキング アプリケーションのプログラム可能な物体(ロボット、センサー、コンベアなど)をツリー構造で編成します。**レイアウト** タブと**プロセス** タブに分かれています。 詳細については、セクションツリー ビュー ブラウザ ページ 77を参照してください。
3	ステーションビュー	ピッキング アプリケーションのリアルな 3D 表示。 ツリー ビュー ブラウザを使用して選択または編集すると、ステーション ビューの物体が強調表示されます。 詳細については、セクションステーションビュー ページ89を参照してください。
4	ログビュー	現在のステーションで発生したすべてのイベントを表示します。 **ヒント** 特定のイベントの検索ボックスにキーワードを入力して検索できます。
5	ステータスビュー	コントローラとシステムの現在の状態を表示します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

続き

		説明
6	追加操作ビュー	保存ボタンとヘルプ ボタンを表示します。 ヘルプ：PickMaster PowerPacアプリケーション マニュアルを開きます。  xx2100000867



ヒント

すべてのウィンドウは、自由に分散してフローティングできます。

© Copyright 2025 ABB. All rights reserved.

3.2 リボンタブ

概要

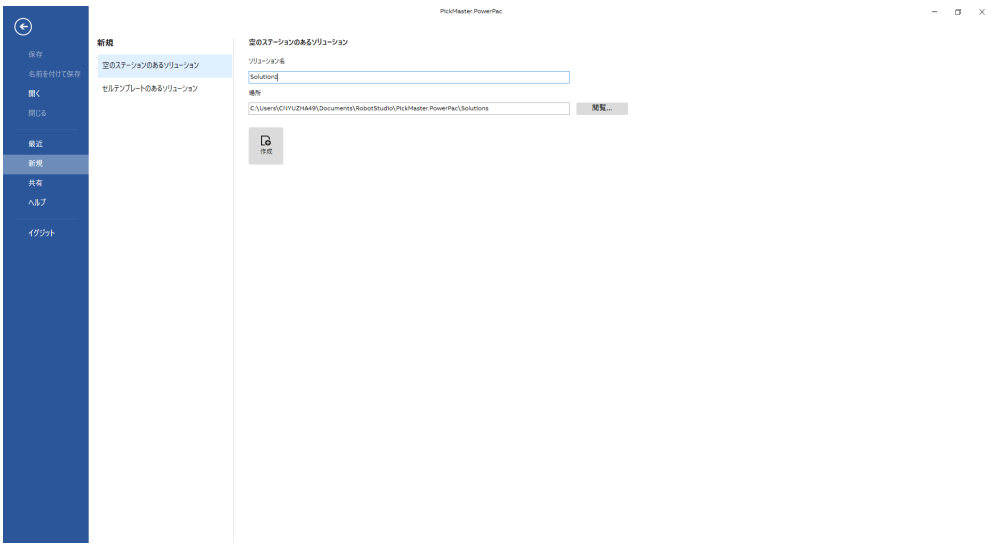
PickMaster PowerPac リボンには、さまざまなグループに配置された要素が含まれています。 次の図と表は、PickMaster PowerPac リボンの要素に関する詳細情報を提供します。

以下は、リボン タブに保存された物体トと設定です。



xx2100000856

ファイル



xx2100000854

ボタン	説明
 xx2100000857	メインウィンドウに戻ります。
保存する	現在、ソリューションの変更を保存します。 **注記** ソリューションが PickMaster Operator で使用される場合、PickMaster PowerPac で同じ設定を持つ実際のコントローラに接続されている必要があります。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

続き

ボタン		説明
名前を付けて保存		現在のソリューションを新しいソリューションとして目的の場所に保存します。 **注記** ソリューションが PickMaster Operator で使用される場合、PickMaster PowerPac で同じ設定を持つ実際のコントローラに接続されている必要があります。
開く		他のソリューション、またはローカル フォルダに保存されているソリューションを開きます。 **ヒント** PickMaster PowerPac 2.0 以降で作成されたソリューションまたは共有ファイルのみを開くことができます。
終了		現在のソリューションを閉じます。
最近		以前に開いたソリューションを開きます。
新規	空のステーションを使用したソリューション	新しい空のソリューションを作成します。 **注記** ソリューションの作成時にファイル名に使用できるのは、a-z, A-Z, 0-9,_ ,-のみです。
	セル テンプレートを使用したソリューション	テンプレートを使用して新しいソリューションを作成します。
情報		開いているソリューションの基本情報を表示します。 **ヒント** このページは、ソリューションを開いた場合にのみ表示されます。

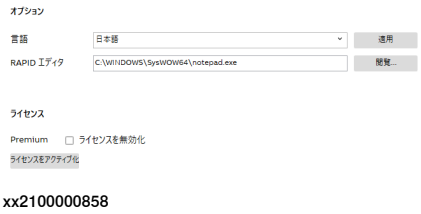
次のページに続く

© Copyright 2025 ABB. All rights reserved.

ボタン		説明
共有	他の人とデータを共有する	**Pack&Go** 現在のソリューション、ソリューションで使用されているコントローラ、および3Dモデルのすべての情報をファイルにまとめて、ユーザー間でファイルを共有できるようにします。 **注記** パックされたファイルの名前変更は不可。そうしないと、開梱に問題が生じる可能性がある。 **注記** Pythonスクリプトファイルは Pack& Go ファイルには含まれない。Pythonスクリプトファイルを目的の保存先にコピー。
		開梱して作業する ソリューション、ソリューションで使用されるコントローラ、および3Dモデルのすべての情報を含む共有ファイルを解凍します。
		テンプレートとしてパック 現在のソリューションをテンプレートとしてローカル フォルダにパックします。
	Runtime ファイル転送	**Runtimeにアップロードする** 必要な.rspagファイルを、接続されているHostコンピュータにアップロードします。「PP」と日付と時刻を含む名前にサフィックスが追加されます。 **ヒント** アップロードする前に、Host コンピュータ上のRealRuntimeに接続します。
		Runtimeからダウンロード 接続されているHostコンピュータから必要な.rspagファイルをダウンロードします。 **ヒント** ダウンロードする前に、Host コンピュータ上のRealRuntimeに接続します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

ボタン		説明
ヘルプ	バージョン情報	基本的なバージョン情報を表示します。
	オプション	言語：適用する言語を選択します。 次の 8 つの言語がサポートされています。 • 英語 • 簡体字中国語 • ドイツ語 • イタリア語 • スペイン語 • 日本語 • フランス語 • 韓国語 ラピッド エディタ：ラピッドを開くエディタを指定します。 ライセンス：現在のライセンスの種類を表示します。 ライセンスを無効にする チェック ボックス：チェックすると、プレミアム ライセンスが無効になります。 ライセンスを有効にする アイコン：プレミアム ライセンスを有効にします。  xx2100000858 **注記** PickMaster PowerPacでの作業中にユーザーが言語を変更した場合、選択した言語はPickMaster PowerPac の再起動後に有効になります。
	マニュアル	PickMaster PowerPac アプリケーションマニュアルを開きます。
終了		PickMaster PowerPacを閉じて終了します。

ヒント

PickMaster PowerPacで新しいソリューションを開くか作成すると、Virtual Runtimeが開始され、自動的に接続されます。

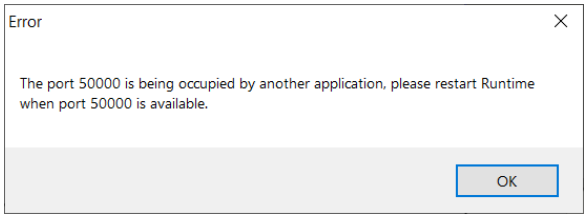
次のページに続く

© Copyright 2025 ABB. All rights reserved.

ヒント

PickMaster® Runtime (VRT および RRT) は、50000 ポートを使用するように定義されています。 50000 ポートが他のプログラムによって占有されている場合、この警告が表示され、Runtimeに接続できなくなります。

50000 ポートを解放し、PickMaster® Runtimeを再起動します。



xx2100000868

この手順を使用して、50000 ポートを解放します。

1. コマンド `netstat -aon|findstr "50000"` を CMD ウィンドウに入力します。
2. ポート 50000 を占有するプロセスがウィンドウに一覧表示されます。 プロセスの PID コードを取得します。
3. タスク マネージャでこの PID に対応するプロセスを見つけて閉じます (このコンピュータでこのプロセスを閉じることが許可されていることを確認してください)。
4. PickMaster® Runtime を再起動して接続します。

レイアウト



xx2100000856

ボタン	説明
コントローラ	ステーション ビューにロボット システムを含むコントローラを追加します。 コントローラの作成についての詳細は、 セクションコントローラ *ページ 116* を参照してください。
グリッパー	グリッパーを追加します。 グリッパーの作成の詳細については、セクショングリッパー *ページ 119*を参照してください。
コンベア	コンベアを追加します。 コンベアの作成に関する詳細は、 セクションコンベヤ *ページ 122* にあります。
カメラ	カメラを追加します。 カメラの作成の詳細については、セクションカメラ *ページ 124*を参照してください。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

続き

ボタン	説明
I/Oセンサー	I/O センサーを追加します。 I/O センサーの作成の詳細については、セクション*I/O センサーの追加 ページ 126*を参照してください。
外部センサー	外部センサーを追加します。 外部センサーの作成の詳細についてはセクション外部センサーの追加 *ページ 128*を参照。
コンベア作業エリア	コンベア作業エリアを追加します。 コンベア作業エリアの作成に関する詳細については、セクション 作業範囲 *ページ 130*を参照してください。
インデックスワークエリア	インデックスワークエリアを追加します。 インデックスワークエリアの作成に関する詳細は、セクション インデックスワークエリアの追加 *ページ 133*を参照してください。
キャリブレーション	PickMaster PowerPacで作成したソリューションを調整します。 作成されたソリューションの調整に関する詳細は、セクション キャリブレーション *ページ 141*で利用できます。

プロセス



xx2100000859

ボタン	説明
アイテム	アイテムを追加します。 アイテムの作成の詳細については、セクション*Item（項目） ページ 142*を参照してください。
コンテナ	コンテナを追加します。 コンテナの作成の詳細については、セクション*Container（コンテナ） ページ 149*を参照してください。
フロー	シミュレーションでアイテムとコンテナを生成する方法を定義します。 フローの作成の詳細については、セクションフロー *ページ157*を参照してください。
レシピ	レシピを作成します。 レシピの作成の詳細については、セクションレシピ *ページ160*を参照してください。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

操作



xx2100000860

ボタン	説明
Start（起動）	シミュレーションを開始します。 下矢印をクリックすると、**開始と記録** が表示される。 **開始と記録**：シミュレーションを開始し、.exeファイルとして記録します。
Stop（停止）	シミュレーションを停止します。 **ヒント** **停止** はソリューションを停止し、ロボットを原点に戻します。
リセット	以前に実行したシミュレーションで一時的に作成された物体からステーション ビューをリセットします。 **ヒント** **リセット** は、コンベア上のアイテムとコンテナをクリーニングします。
コントロール	生産を開始します。 生産を行う方法の詳細については、セクション*シミュレーション ページ 173* と *エミュレーション ページ 333*を参照してください。
記録開始	カーソルとマウス クリックを含むシミュレーションを ..mp4 ファイルとして記録します。
記録停止	カーソルとマウス クリックを含むシミュレーションの記録を停止します。

Runtime



xx2100000862

次のページに続く

© Copyright 2025 ABB. All rights reserved.

ボタン	説明
ローカル RRT を開始	コンピュータでRuntimeを開始します。  注記 ローカル RRT は、PickMaster PowerPacとともにインストールされるRuntimeを指す。検査用としても使用可能。
RRTに接続	Real Runtimeに接続します。
VRTの起動および接続	コンピューター上でバーチャル Runtimeを起動して接続。

© Copyright 2025 ABB. All rights reserved.

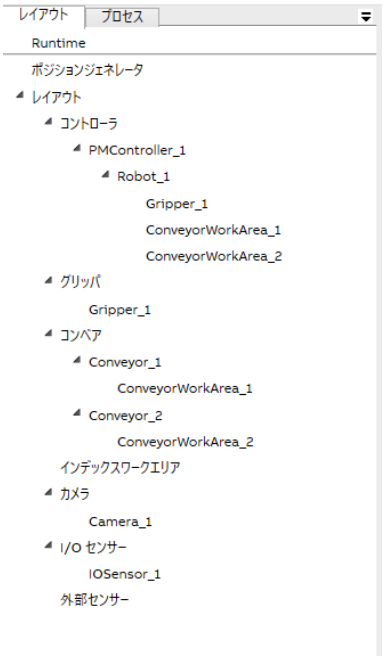
3.3 ツリー ビュー ブラウザ

3.3.1 レイアウト

概要

レイアウト タブには、Runtimeと、ロボット、カメラ、コンベアなどのアプリケーション ハードウェア物体が表示されます。

以下は、**レイアウト** タブに保存される物体と設定です。



xx2100000863

- Runtime
- **Position generator（ポジションジェネレータ）**
- **レイアウト**
  - **コントローラ**
  - **グリッパー**
  - **コンベア**
  - **インデックスワークエリア**
  - **カメラ**
  - **I/O センサー**
  - **外部センサー**

Runtime

Runtime を右クリックして、シミュレーションモードでバーチャル Runtime（VRT）、あるいはエミュレーションモードでコンピHostュータ上で実際のロボットを操作するReal Runtime（RRT）への接続を設定します。

次のページに続く

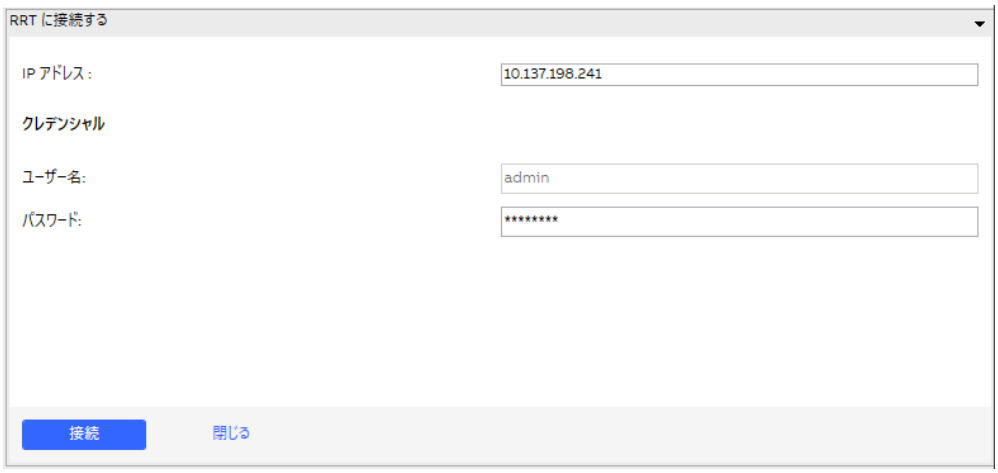
© Copyright 2025 ABB. All rights reserved.

3.3.1 レイアウト
続き

ヒント

RRTへの接続前に、PickMaster RuntimeをHostコンピュータ上で起動。

RRT に接続を選択すると、サインインウィンドウが表示される。



xx2100000872

	説明
IPアドレス	Runtime コンピュータの IP アドレスを入力。 **ヒント** PickMaster Runtimeがインストールされているコンピュータの IPv4 アドレスを確認します。
資格情報	
ユーザー名	デフォルトのユーザー名はadmin です。 変更はできません。
パスワード	Runtimeでアカウントのパスワードを入力します。

注記

ローカル RRT は、PickMaster PowerPacとともにインストールされるRuntimeを指す。検査用としても使用可能。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

Position generator（ポジションジェネレータ）

ポジションジェネレータ を右クリックし、関係およびコンベアのソースタイプを定義。



xx2100000874

	説明
利用可能なコンベアおよびインデックスワークエリアリスト	関連する関係を設定するために、コンベアまたはインデックスワークエリアを選択。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

3.3.1 レイアウト
続き

	説明
ソースタイプ	入力シグナル ソース タイプを選択します。 • ビジョン：ソースタイプがビジョンに設定されている場合、カメラおよびビジョンモデルを使用して物体のポジションが検索されます。ビジョンモデルはビジョンモデルの追加 ページ *293*に説明されています。  ヒント ソース タイプがビジョンに設定されている場合、使用可能なすべてのカメラと関連アイテムが使用可能なカメラに一覧表示されます。 • 事前定義：ソース タイプが事前定義に設定されている場合、ポジションソースによって生成されるポジションは静的に定義され、カメラは使用されません。 • 外部：ソースタイプが外部 に設定されている場合は、ソリューション内の外部センサーおよび外部ポジションジェネレータを使用して、アイテムのポジションを定義します。  ヒント インデックスワークエリアが使用されている場合、外部センサー機能は無効になる。
トリガー設定	トリガー タイプを選択して、新しいアイテムのポジションをいつ生成するかを定義します。  注記 トリガー タイプが距離 に設定されている場合、レシピ の下の操作 設定の トリガー距離 ボックスでトリガー距離を定義する必要があります。 間隔トリガは、コンベヤワークエリアでのみ使用できます。入力された値は、連続するトリガ同士の間で、コンベヤが移動する距離です.  注意 レシピで事前定義 センサーと IO センサーが選択されている場合は、オフセットを構成するアイテムの半径距離のチューニングでピックポジションを調整します。  ヒント インデックスワークエリアが使用されている場合、トリガー設定は使用できません。
ベースフレーム調整	選択したコンベアまたはインデックスワークエリアのベース フレームを調整します。 詳しくは「ベースフレームの調整 ページ *138*」を参照してください。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

コントローラ

コントローラの管理

ツリービューで**コントローラ**を右クリックし、コントローラを選択して定義。

	説明
コントローラの編集	選択したコントローラの設定を変更します。 コントローラを右クリックして**コントローラの編集**を選択すると、**コントローラの編集** ウィンドウが表示されます。 選択したコントローラの管理の詳細については、次のセクションを参照してください。
削除	選択したコントローラを削除します。
名前の変更	選択したコントローラの名前を変更します。

次の表は、**コントローラ編集**ウィンドウの詳細を示す。

	説明
システム名	システムの名前を表示します。
IPアドレス	選択したコントローラの IP アドレスを表示します。
バージョン	システムのバージョンを表示します。
システムID	システムの ID を表示します。
システム パス	仮想コントローラ システムのパスを表示します。
仮想コントローラーの選択 アイコン	選択した仮想コントローラを起動します。
リアルコントローラの選択 アイコン	生産を行うときは、リアルコントローラを選択します。

ロボットの管理

ツリービューで**ロボット**を右クリックして、ロボットを管理。

	説明
ホームにジャンプ	ロボットをホームポジションに移動します。
ポジションの設定	選択したロボットのポジションを設定します。 ロボットを右クリックして**ポジションの設定** を選択すると、**ロボットポーズの設定** ウィンドウが表示されます。 選択したロボットのポジションを管理する方法の詳細については、次のセクションを参照してください。
検査	ステーション ビューでロボットを検査します。
名前の変更	選択したロボットの名前を変更します。
移動	ステーション ビューでコンポーネントを直接移動し、**[位置の設定]** ウィンドウを開くことができます。
非表示/表示	コンポーネントを非表示にし、ツリー ビューでコンポーネント名の前に **(非表示)** とマークします。

ポジションの設定

次の表は、**ポジション設定設定**ウィンドウの詳細を示す。

	説明
参考資料	座標系を選択します。
ポジション X、Y、Z (mm)	選択したロボットの新しいポジションを設定します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

3.3.1 レイアウト
続き

	説明
方位 (度)	選択したロボットの新しい方位を設定します。

グリッパー

グリッパーの管理

グリッパーの管理は、ツリービューでグリッパーを 右クリックします。

	説明
設定	選択したグリッパーの設定を管理します。 設定 を選択すると ロボットグリッパーの設定 ウィンドウが表示されます。 選択したグリッパーの管理の詳細については、セクショングリッパー ページ *119*を参照してください。
削除	選択したグリッパーを削除します。
名前の変更	選択したグリッパーの名前を変更します。
検査	ステーション ビューで選択したグリッパーを検査します。
非表示/表示	コンポーネントを非表示にし、ツリー ビューでコンポーネント名の前に (非表示) とマークします。

コンベア

コンベアの管理

ツリービューでコンベアを右クリックして、コンベアを管理。

	説明
設定	選択したコンベアの設定を管理します。 設定を選択すると、コンベア設定 画面が表示されます。 選択したコンベアの管理についての詳細は、セクションコンベヤ ページ *122*にあります。
削除	選択したコンベアを削除します。
名前の変更	選択したコンベアの名前を変更します。
ホットスポット	ホットスポットを管理します。 ホットスポット を選択すると、コンベア ホットスポットの設定 ウィンドウが表示されます。コンベア ホットスポットの設定 ウィンドウの詳細については、次のセクションを参照してください。 **注記** ホットスポットは、コンベア上の保存された場所です。ホットスポットは、コンベア上のフローが生成される場所を定義します。 コンベアの始点には常にデフォルトのホットスポット、ホットスポット0があります。 フローが間違った場所に表示される場合は、ホットスポットの場所を変更して調整します。
検査	ステーションビューで選択したコンベアを調べます。
移動	ステーション ビューでコンポーネントを直接移動し、その設定ウィンドウを開くことができます。
非表示/表示	コンポーネントを非表示にし、ツリー ビューでコンポーネント名の前に (非表示) とマークします。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

作業エリアの管理

ツリービューのコンベア WAを右クリックして、作業範囲を管理。

	説明
設定	選択した作業エリアの設定を管理します。 コンベアの作業エリアを右クリックして設定を選択すると、コンベア WA 設定ウィンドウが表示されます。 コンベア作業エリアの管理の詳細については、セクション作業範囲 ページ *130*を参照してください。 インデックスワークエリアを右クリックして設定を選択すると、インデックス付き WA 設定 ウィンドウが表示されます。 インデックスワークエリアの管理の詳細については、セクションインデックスワークエリアの追加 ページ *133*を参照してください。
削除	選択したコンベア作業エリアを削除します。
名前の変更	選択したコンベア作業エリアの名前を変更します。

インデックスワークエリア

インデックスワークエリアの管理

ツリービューでインデックスワークエリアを右クリックして、インデックスワークエリアを管理。

	説明
設定	選択したインデックスワークエリアの設定を管理します。 設定 を選択すると、インデックスワークエリアの設定 ウィンドウが表示されます。 選択したコンベアの管理についての詳細は、セクション*Indexed work area（インデックスワークエリア） ページ 133*にあります。
削除	選択したインデックスワークエリアを削除します。
名前の変更	選択したインデックスワークエリアの名前を変更します。
ホットスポット	ホットスポットを管理します。 ホットスポット を選択すると、インデックスワークエリアのホットスポットの設定 ウィンドウが表示されます。 インデックスワークエリアのホットスポットの設定 ウィンドウの詳細については、次のセクションを参照してください。 注記 ホットスポットは、インデックスワークエリアに保存された場所です。 ホットスポットは、インデックスワークエリアでフローが生成される場所を定義します。 インデックスワークエリアの先頭には、デフォルトのホットスポットである Hotspot0 が常に存在します。 フローが間違った場所に表示される場合は、ホットスポットの場所を変更して調整します。
検査	ステーション ビューで、選択したインデックス付きの作業エリアを調べます。
移動	ステーション ビューでコンポーネントを直接移動し、その設定ウィンドウを開くことができます。
非表示/表示	コンポーネントを非表示にし、ツリー ビューでコンポーネント名の前に (非表示) とマークします。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

3.3.1 レイアウト
続き

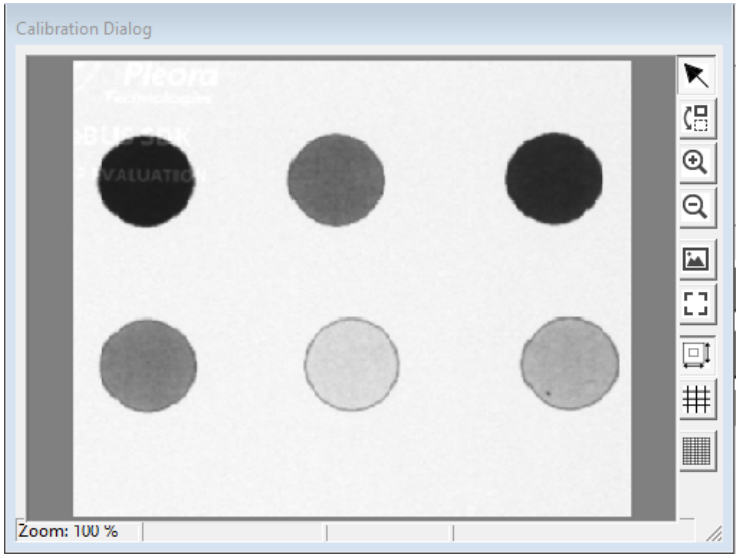
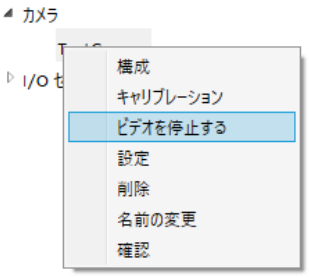
カメラ

ツリービューでカメラを右クリックして、カメラを管理。

	説明
構成	選択したカメラを設定します。 カメラを右クリックして設定 を選択すると、カメラ設定 ウィンドウが表示されます。 カメラの管理の詳細については、セクションカメラの設定 ページ *190*を参照してください。
キャリブレーション	選択したカメラをキャリブレーションします。 カメラを右クリックしてキャリブレーション を選択すると、カメラのキャリブレーション ウィンドウが表示されます。 カメラの管理の詳細については、セクションカメラのキャリブレーション ページ *282* を参照してください。  ヒント キャリブレーション、ライブ ビデオ、および設定 は、実際のカメラに対してのみ有効です。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

	説明
ライブビデオ	生産前のリアルカメラのシーンを表示します。  xx2100001521 ライブ ビデオを開始したら、センサーを右クリックして ビデオを停止 をクリックすることで停止できます。  xx2100001809 **ヒント** キャリブレーション、ライブ ビデオ、および設定 は、実際のカメラに対してのみ有効です。
設定	選択したカメラの設定を管理します。 カメラを右クリックして設定を選択すると、カメラ設定ウィンドウが表示されます。 カメラの管理の詳細については、セクションカメラ ページ 124を参照してください。 **ヒント** キャリブレーション、ライブ ビデオ、および設定 は、実際のカメラに対してのみ有効です。
削除	選択したカメラを削除します。
名前の変更	選択したカメラの名前を変更します。
検査	ステーション ビューで選択したカメラを調べます。
移動	ステーション ビューでコンポーネントを直接移動し、その設定ウィンドウを開くことができます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

3.3.1 レイアウト
続き

	説明
非表示/表示	コンポーネントを非表示にし、ツリー ビューでコンポーネント名の前に (非表示) とマークします。

I/O センサー

I/O センサーの管理は、ツリービューで**I/O センサー**を 右クリックします。

	説明
設定	選択した I/O センサーの設定を管理します。 I/O センサーを右クリックして**設定** を選択すると、**I/O センサー設定** ウィンドウが表示されます。 I/O センサーの管理の詳細については、セクション*I/O センサーの追加 ページ 126*を参照してください。
削除	選択した I/O センサーを削除します。
名前の変更	選択した I/O センサーの名前を変更します。
検査	ステーション ビューで選択した I/O センサーを調べます。
移動	ステーション ビューでコンポーネントを直接移動し、その設定ウィンドウを開くことができます。
非表示/表示	コンポーネントを非表示にし、ツリー ビューでコンポーネント名の前に (非表示) とマークします。

外部センサー

外部センサーの管理は、ツリービューで**外部センサー**を 右クリックします。

	説明
設定	選択した外部センサーの設定を管理。 外部センサーを右クリックして**設定**を選択すると、**外部センサー設定** ウィンドウが表示される。カメラ管理の詳細については、セクション外部センサーの追加 ページ *128*を参照。
構成	選択した外部センサーを設定。 外部センサーを右クリックして**設定**を選択すると、**外部センサー設定**ウィンドウが表示されます。カメラ管理の詳細については、セクション外部センサー ページ *357*を参照。
削除	選択した外部センサーを削除。
名前の変更	選択した外部センサーの名前を変更。
検査	ステーション ビューで選択した外部センサーを調査。
移動	ステーション ビューでコンポーネントを直接移動し、その設定ウィンドウを開くことができます。
非表示/表示	コンポーネントを非表示にし、ツリー ビューでコンポーネント名の前に (非表示) とマークします。

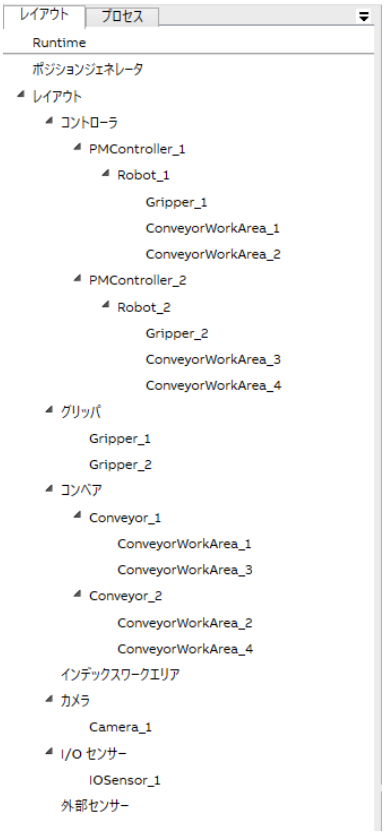
© Copyright 2025 ABB. All rights reserved.

3.3.2 Process

概要

プロセス タブには、設定ファイルと、アイテム、コンテナ、フロー、レシピなどのアプリケーション ハードウェア 物体が表示されます。

以下は、**プロセス** タブに保存される物体と設定です。



xx2100000864

- **アイテム**
- **コンテナ**
- **フロー**
- **レシピ**

アイテム

管理アイテム

ツリービューで**アイテム**を右クリックすると、そのアイテムを管理できます。

	説明
設定	選択したアイテムの設定を管理します。 **設定**を選択すると**アイテム設定** ウィンドウが表示されます。 選択したアイテムの管理の詳細については、セクションアイテムの追加 ページ 142を参照してください。
削除	選択したアイテムを削除します。
名前の変更	選択したアイテムの名前を変更します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

3.3.2 Process
続き

	説明
コピー	選択したアイテムのコピーをすべての設定で作成します。

コンテナ

コンテナの管理

ツリービューでコンテナを 右クリックして、コンテナを管理。

	説明
設定	選択したコンテナの設定を管理します。 設定を選択すると、コンテナ設定 ウィンドウが表示されます。 選択したコンテナの管理の詳細については、セクションコンテナの追加 ページ *149*を参照してください。
削除	選択したコンテナを削除します。
名前の変更	選択したコンテナの名前を変更します。
コピー	選択したコンテナのコピーをすべての設定で作成します。

フロー

フローの管理

ツリービューでフローを右クリックして、フローを管理。

	説明
設定	選択したフローの設定を管理します。 フローを右クリックして設定 を選択すると、フロー設定 ウィンドウが表示されます。 フローの管理の詳細については、セクションフロー ページ *157*を参照してください。
削除	選択したフローを削除します。
名前の変更	選択したフローの名前を変更します。
コピー	選択したフローのコピーをすべての設定で作成します。

レシピ

レシピの管理

ツリービューでレシピを右クリックし、レシピを管理。

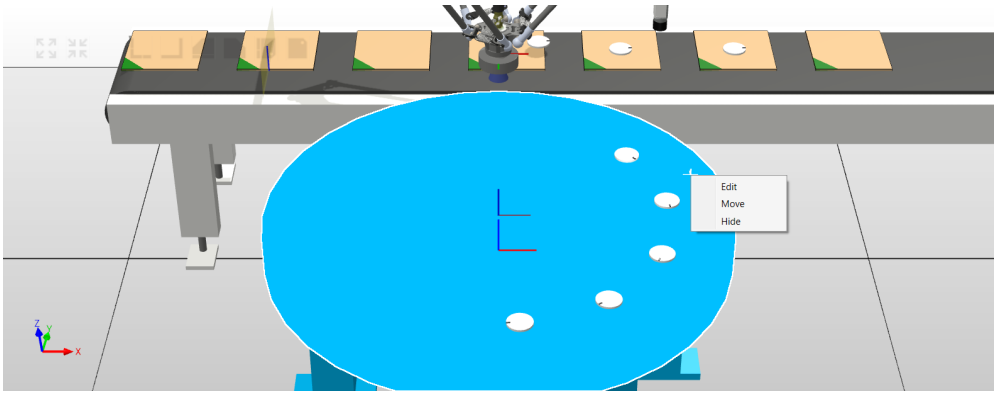
	説明
設定	選択したレシピの設定を管理します。 設定を選択すると、レシピ設定ウィンドウが表示されます。 選択したレシピの管理の詳細については、セクションレシピ ページ *160*を参照してください。
削除	選択したレシピを削除します。
名前の変更	選択したレシピの名前を変更します。
コピー	選択したレシピのコピー ファイルをすべての設定で作成します。

© Copyright 2025 ABB. All rights reserved.

3.4 ステーションビュー

ステーションビュー

レイアウト コンポーネントを右クリックすると、3D 機能がポップアップ表示されます。ステーション ビューから直接 3D 操作を使用する場合、コンポーネントごとに使用できる機能が異なります。



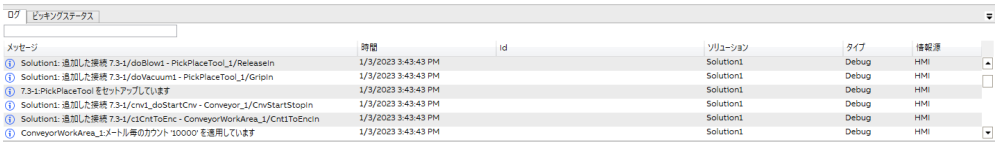
xx2400001316

		説明	コントローラ (ロボット)	グリッパー	コンベヤ	インデックス付きワークエリア	カメラ	I/Oセンサー	外部センサー
1	編集	コンポーネント設定ウィンドウを開きます。	x	x	x	x	x	-	x
2	移動	ドラッグすると、ワールド フレームに基づいてステーションビューでコンポーネントを直接移動し、同時にコンポーネント設定ウィンドウを開きます。 **ヒント** 基準座標が手動で**ローカル**に設定されている場合、ステーションビューでコンポーネントを直接ドラッグして更新すると、自動的に**ワールド**に変更されます。	x	-	x	x	x	-	x
3	非表示	コンポーネントを非表示にし、ツリー ビューでコンポーネント名の前に (非表示) とマークします。 コンポーネントが非表示になっている場合は、ツリー ビューの [表示] ボタンからのみ再表示できます。	x	x	x	x	x	-	x

© Copyright 2025 ABB. All rights reserved.

3.5 ログビュー

ログ



xx2100001518

	説明
ログ	すべてのログを表示します。 **注記** 1 つのログ メッセージを右クリックすると、**ログを保存** と **すべてクリア** が利用できます。
フィルタボックス	特定のログをキーワードでフィルタリングします。
コンテキストメニュー	現在のログのエクスポートやクリーンアップなど、ログに対する操作をさらに拡張します。
ピッキング状況	ピッキング状況の概要を要約または詳細に表示します。

© Copyright 2025 ABB. All rights reserved.

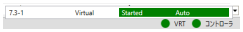
3.6 ステータスビュー

状態

システムが起動すると、コントローラとRuntimeのステータスが図のように右上隅に表示されます。



xx2100000865

	説明	注記
Runtime	**グレー**：ソリューションが開かれていません。	
VRT	**赤**：Virtual Runtimeへの接続に失敗しました。 **緑**：Virtual Runtimeへの接続に成功しています。 **黄色**：Virtual Runtimeへの接続が進行中です。	
RRT	**赤**：Real Runtimeへの接続に失敗しました。 **緑**：Real Runtimeへの接続に成功しています。 **黄色**：Real Runtimeへの接続が進行中です。	
コントローラ	**赤**：少なくとも 1 つのコントローラが停止しています。 **緑**：すべてのコントローラが起動し、自動実行中です。 **黄色**：少なくとも 1 つのコントローラが開始され、手動で制御されているか、接続されたばかりです。 **グレー**：既存のソリューションにコントローラが追加されていません。	
コントローラ	**赤**：コントローラが停止しています **緑**：コントローラが起動し、自動実行中です。 **黄色**：コントローラが起動され、手動で制御されているか、接続されたばかりです。	**コントローラ** ボタンをクリックすると、各コントローラの詳細なステータスが表示されます。  xx2100001517

© Copyright 2025 ABB. All rights reserved.

このページは意図的に空白のまま残しています

4 PickMaster PowerPacを使用する作業

4.1 概要

概要

仮想RuntimeでのPickMaster PowerPacを使用する作業は、視覚的な状態でシミュレーション機能を実現することです。

RealRuntimeでのPickMaster PowerPacを使用する作業は、実際のロボットとコントローラを使用して、実際のステーションでエミュレーションと生産機能を実現することです。

シミュレーションは、実際のステーションを作成する際のコストと時間を節約するための事前のデバッグ手順です。

PickMaster PowerPacを使用する場合の推奨フローは次のとおりです。作業フローを完了したら、これらのタスクを任意の順序で実行できます。

注記

コントローラ (少なくとも 1 つのロボット システムを含む) は、PickMaster PowerPac でコントローラをソリューションに追加する前に、RobotStudio または PickMaster PowerPac でセットアップする必要があります。

ソリューションで複数のコントローラが必要な場合は、事前に複数のコントローラを作成する必要があります。 PickMaster PowerPacでは、同じコントローラを同じソリューションに繰り返しインポートすることはできません。

注記

ファイアウォールまたはウイルス対策ソフトウェアがインストールされている場合は、`pickmasteru.exe` および `visionclient.exe` をホワイト リストに追加します。

そうしないと、PickMaster PowerPac はRuntime に接続できず、ビジョン機能が正常に動作しません。

PickMaster PowerPacの作業フロー

PickMaster PowerPac を使って作業するには、次の手順を使用します。

		タスク	説明
VRT	1	空のソリューションを作成します。	詳細については、ソリューション ページ*115*を参照してください。
	2	コントローラを追加します。	詳細については、コントローラ ページ*116*を参照してください。
	3	グリッパーを追加します。	詳細については、グリッパー ページ*119*を参照してください。
	4	コンベアを追加します。	詳細については、コンベヤ ページ*122*を参照してください。
	5	カメラを追加します。	詳細については、カメラ ページ*124*を参照してください。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

		タスク	説明
	6	I/O センサーを追加します。	詳細については、*I/O センサーの追加 ページ 126*を参照してください。
	7	外部センサーを追加します。	詳細については、*External sensor（外部センサー） ページ 128*を参照してください。
	8	作業エリアを追加します。	詳細については、作業範囲 ページ*130*を参照してください。
	9	インデックスワークエリアを追加します。	詳細については、*Indexed work area（インデックスワークエリア） ページ 133*を参照してください。
	10	ポジションジェネレータを設定します。	詳細については、ポジションジェネレータ ページ*136*を参照してください。
	11	ソリューションを調整します。	詳細については、キャリブレーション ページ*141*を参照してください。
	12	アイテムを追加します。	詳細については、*Item（項目） ページ 142*を参照してください。
	13	コンテナを追加します。	詳細については、*Container（コンテナ） ページ 149*を参照してください。
	14	レシピを追加します。	詳細については、レシピ ページ*160*を参照してください。
	15	シミュレーションを行います。	詳細については、シミュレーション ページ*173*を参照してください。
RRT	16	ロボットをキャリブレーションします。	詳細については、キャリブレーションロボット ページ*194*を参照してください。
	17	Real **Runtime**に切り替えます。	詳細については、*Real Runtime*への切り替え ページ*179*を参照してください。
	18	カメラを設定します。	詳細については、カメラの設定 ページ*190*を参照してください。
	19	外部センサーの設定。	詳細については、外部センサー ページ*357*を参照してください。
	20	カメラをキャリブレーションします。	詳細については、カメラのキャリブレーション ページ*282*を参照してください。
	21	コンベアまたはインデックスワークエリアをキャリブレーションします。	詳細については、リニア コンベアのキャリブレーション ページ*195*、円形コンベアのキャリブレーション ページ*226*、インデックスワークエリアのキャリブレーション ページ*265*を参照してください。
	22	キャリブレーションを確認します。	詳細については、コンベアのキャリブレーションの検証 ページ*280*を参照してください。
	23	ビジョン モデルを追加します。	詳細については、ビジョンモデルの追加 ページ*293*を参照してください。
	24	生産を開始します。	詳細については、生産開始 ページ*333*を参照してください。

© Copyright 2025 ABB. All rights reserved.

4.2 フレーム関係

4.2.1 座標系とは何ですか?

概要

座標系は、起点とよばれる固定ポイントから軸を使って面または領域を定義します。ロボットターゲットとポジションは、座標系の軸を基にした測定を使って配置されます.

ロボットは、ジョグやプログラミングのタイプに合わせて複数の座標系を使用します。

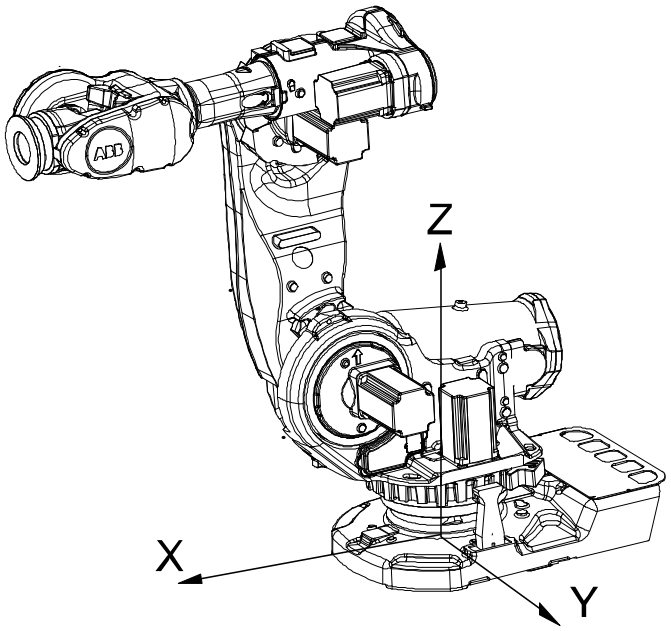
- ベース座標系は、ロボットのベースにあります。 これは、ロボットをあるポジションから別のポジションへ移動させるのに最も簡単な座標系です。 詳しくは、ベース座標系 ページ 96 を参照してください。
- ロボット セルを定義するワールド座標系。他のすべての座標系は、直接的または間接的にワールド座標系に関連付けられています。これは、ジョギング、一般的な移動、および複数のロボットまたは追加の軸によって移動されるロボットを含むステーションとセルの処理に役立ちます。詳細については、世界座標系 ページ 97を参照してください。
- ユーザー座標系は、作業物体座標系など他の座標系を保持している機器を示すのに便利です。詳しくは、ユーザー座標系 ページ98を参照してください。
- 作業オブジェクト座標系は、作業オブジェクトに基づいており、ロボットのプログラムに適しています。 詳しくは、『作業物体座標系 ページ 99』を参照してください.
- ツール座標系は、ロボットがプログラムされたターゲットへ移動する際に使用するツールのポジションを定義します。 詳しくは、ツール座標系 ページ100を参照してください。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.2.1 座標系とは何ですか?
続き

ベース座標系



xx0300000495

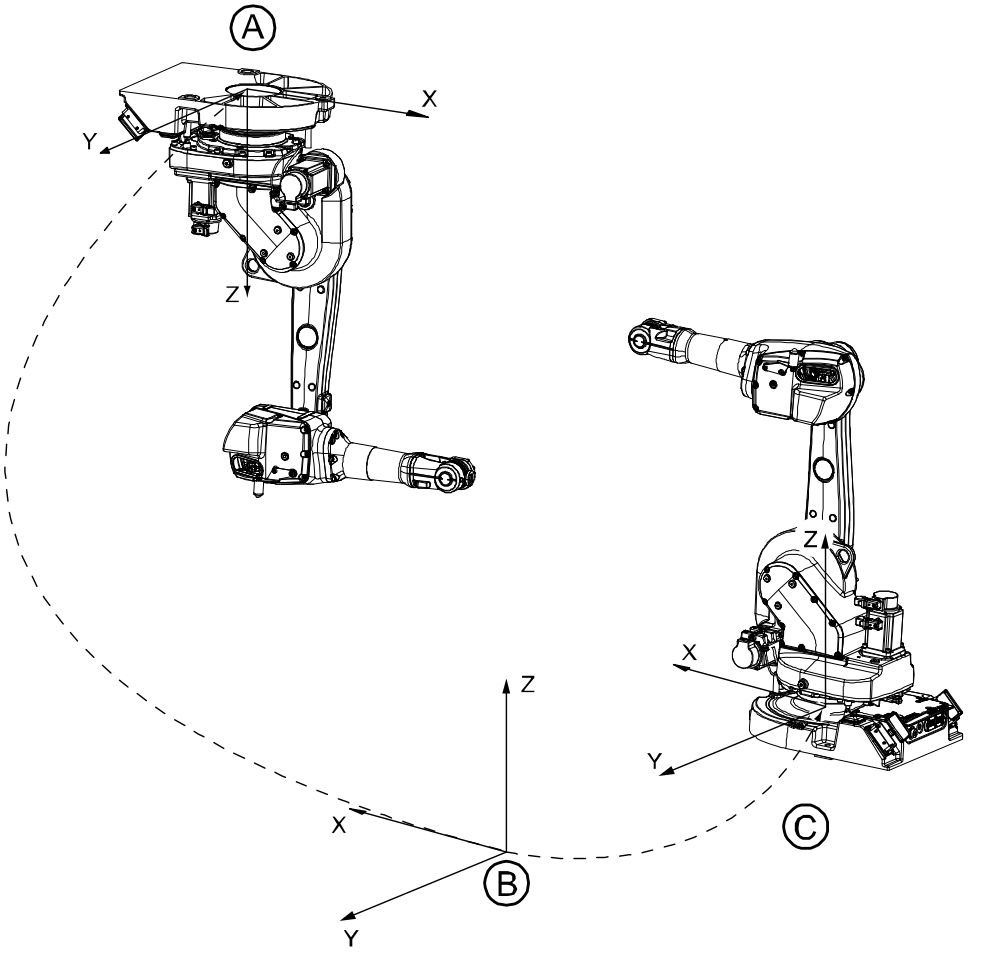
ツール座標系ではロボットのベースにゼロポイントがあり、固定装備するロボットの動きを予測可能にします。そのため、ロボットをあるポジションから別のポジションへジョグさせるのに便利です。ロボットをプログラムする場合は、作業オブジェクト座標系など他の座標系がより適しています。詳しくは、『作業物体座標系 ページ 99』を参照してください.

通常のロボットシステムで設定されているロボットの前面に立ってベース座標系でジョグさせると、ジョイスティックを手前へ引くとロボットはY軸に沿って動き、ジョイスティックを横へ動かすとロボットはX軸に沿って動きます。ジョイスティックを旋回させると、ロボットはZ軸に沿って動きます.

次のページに続く

© Copyright 2025 ABB. All rights reserved.

世界座標系



en0300000496

A	ロボット1のベース座標系
B	ワールド座標
C	ロボット2のベース座標系

ワールド座標系では、ゼロ ポイントはセルまたはステーション内の固定位置にあります。これにより、複数のロボットや追加の軸で移動するロボットを処理する場合に便利です。

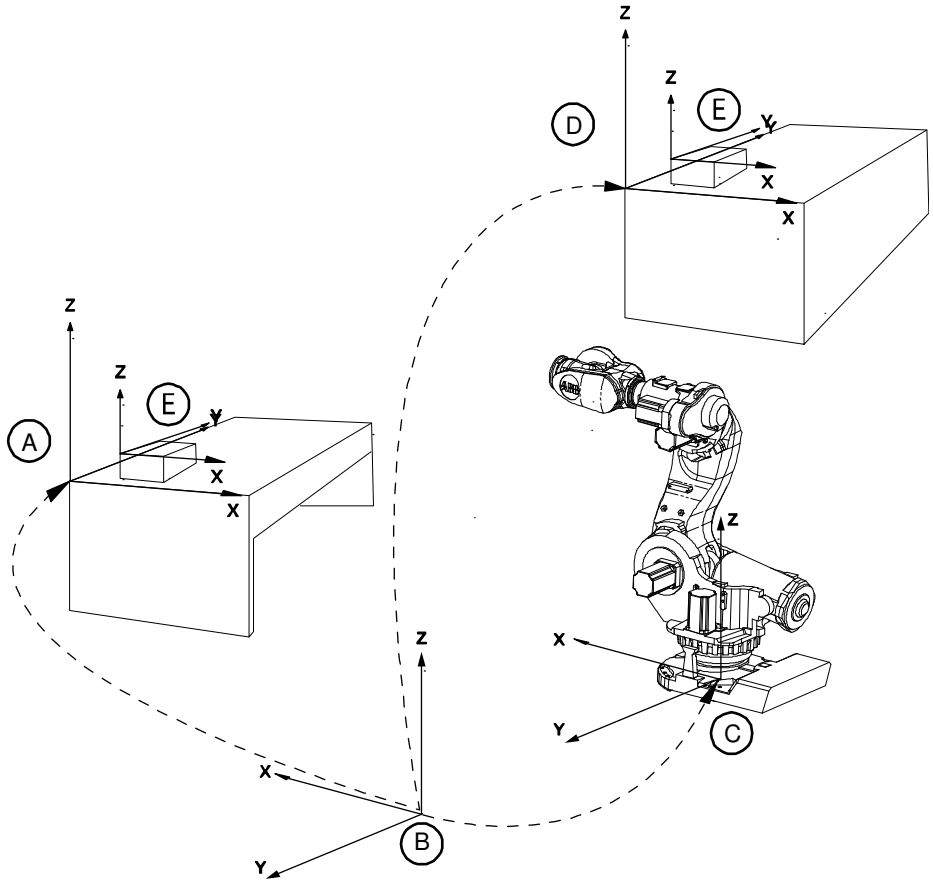
デフォルトでは、ワールド座標系はベース座標系と一致しています.

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.2.1 座標系とは何ですか?
続き

ユーザー座標系



en0400001225

A	ユーザー座標系
B	ワールド座標系
C	ベース 物体座標系
D	移動したユーザー座標系
E	ユーザー座標系で移動させた作業オブジェクト座標系

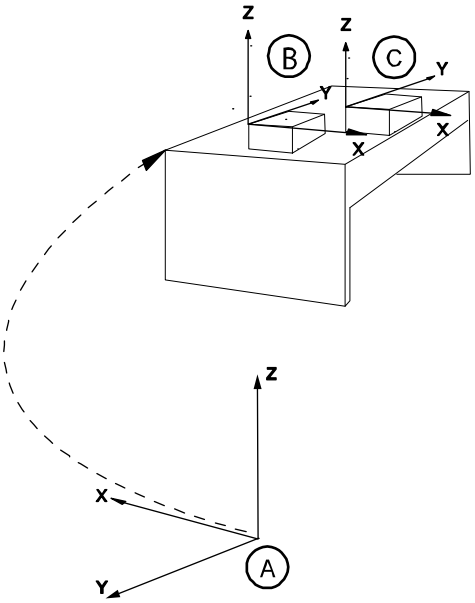
ユーザー座標系は、固定具やワークベンチなどの機器を示すことができます。そのため、関連する座標系の連鎖に追加レベルを提供し、作業オブジェクト座標系やその他の座標系を使用している機器の操作に便利です.

ユーザー座標系を定義する方法については、*Technical reference manual - RAPID Instructions, Functions and Data types* のデータ型 `wobjdata` に関する情報を参照してください。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

作業物体座標系



xx0600002738

A	ワールド座標系
B	作業オブジェクト座標系1
C	作業オブジェクト座標系2

作業オブジェクト座標系は、作業オブジェクトに対応しています。ワールド座標系（または他の座標系）に関連付けて、作業オブジェクトのポジションを定義します.

一機のロボットで複数の作業オブジェクト座標系を使用でき、異なる作業オブジェクトや同じ作業オブジェクトの複製を別のポジションで使用する場合などに便利です.

ロボットをプログラムする際は、ターゲットとパスを作業オブジェクト座標系で作成します。この座標系にはさまざまな利点があります:

- ステーションで作業オブジェクトを示す場合、作業オブジェクト座標系のポジションを変更するだけですべてのパスが一度に更新されます.
- ワークオブジェクト全体をそのパスとともに移動できるため、追加の軸またはコンベア トラックによって移動されるワーク ピースでの作業が可能になります。

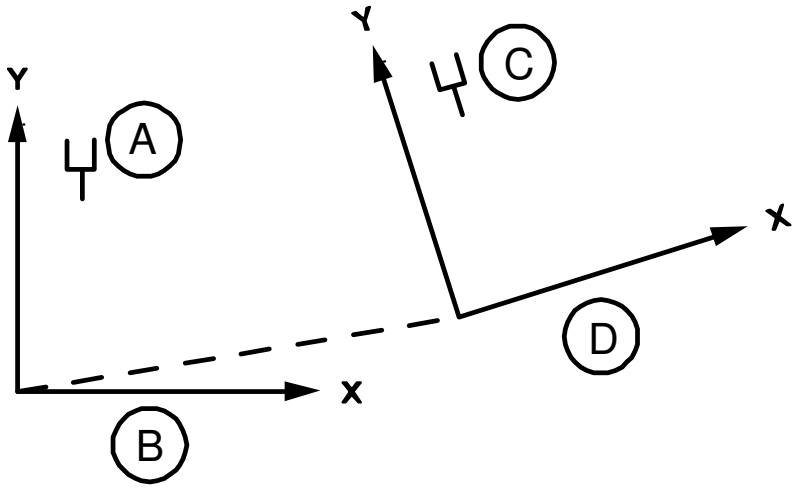
作業オブジェクトの座標系を定義する方法については、*Technical reference manual - RAPID Instructions, Functions and Data types* のデータ型 `wobjdata` に関する情報を参照してください。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.2.1 座標系とは何ですか?
続き

変位座標系



en0400001227

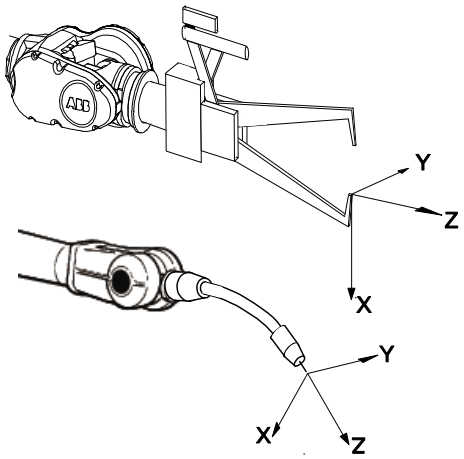
A	元のポジション
B	オブジェクト座標系
C	新しいポジション
D	変位座標系

場合により、同じオブジェクトの複数ポジションで、または互いに隣接する複数の作業オブジェクトで、同じパスを実行する場合があります。そのような場合に、すべてのポジションをプログラムし直さなくてもいいように、変位座標系を定義することができます。

変位座標系は、個々の部品のポジションの違いを補正するために、検索と組み合わせて使用することもできます。

変位座標系は、作業物体座標系に基づいて定義します。

ツール座標系



en0300000497

次のページに続く

© Copyright 2025 ABB. All rights reserved.

ツール座標系では、ツールのセンターポイントがゼロポジションになります。そこを基点に、ツールのポジションと方向を定義します。ツール座標系は、TCP（Tool Center Point Frame）またはTCPF（Tool Center Point）と呼ばれます.

TCPは、プログラムを実行するとロボットがプログラムされたターゲットポジションへジョグします。そのため、ツール（およびツール座標系）を変更すると、ロボットの動きも変更され、新しいTCPがターゲットまで移動します.

すべてのロボットには `tool0` という名前であらかじめ定義されているツール座標系があり、ロボットのリストにあります。次に、1つまたは多数の新しいツール座標系を `tool0` からオフセットとして定義できます.

ロボットをジョグさせる場合、のこぎりの刃を曲げずに動かすなど、稼動中にツールのポジションを変更したくない場合にツール座標系は便利です.

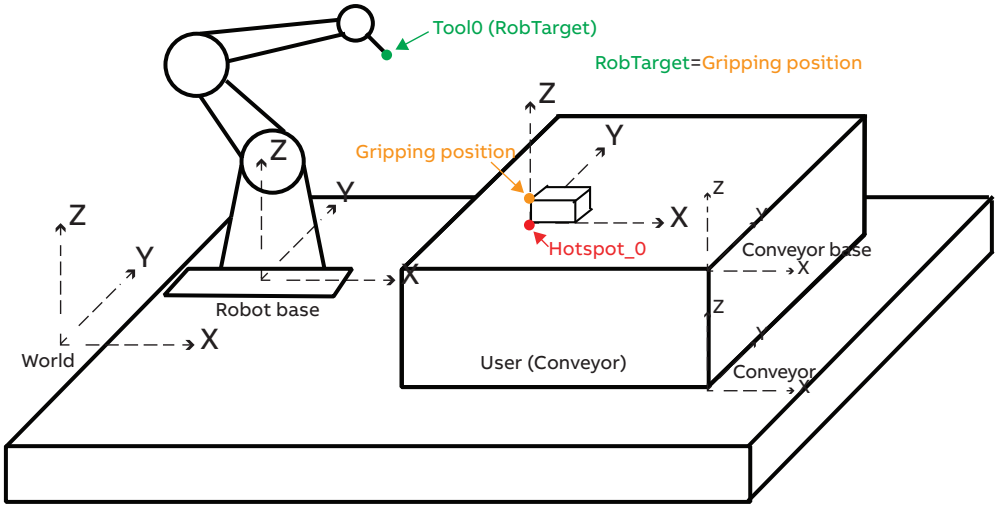
ツール座標系を定義する方法については、*Technical reference manual - RAPID Instructions, Functions and Data types* のデータ型 `tooldata` に関する情報を参照してください。

© Copyright 2025 ABB. All rights reserved.

4.2.2 PickMaster® Twin のフレーム関係

概要

このセクションでは、PickMaster PowerPacソリューション内のコンベアに関する座標系の定義について説明します。



xx2100002290

ワールドフレーム

ワールド フレームは、PickMaster PowerPac ソリューションの基本的なフレームです。ロボット、コンベアなどの他のすべてのコンポーネントのポジションは、このフレームで表現されます。

ローカル フレーム

すべての物体には、ローカル座標系と呼ばれる独自の座標系があります。

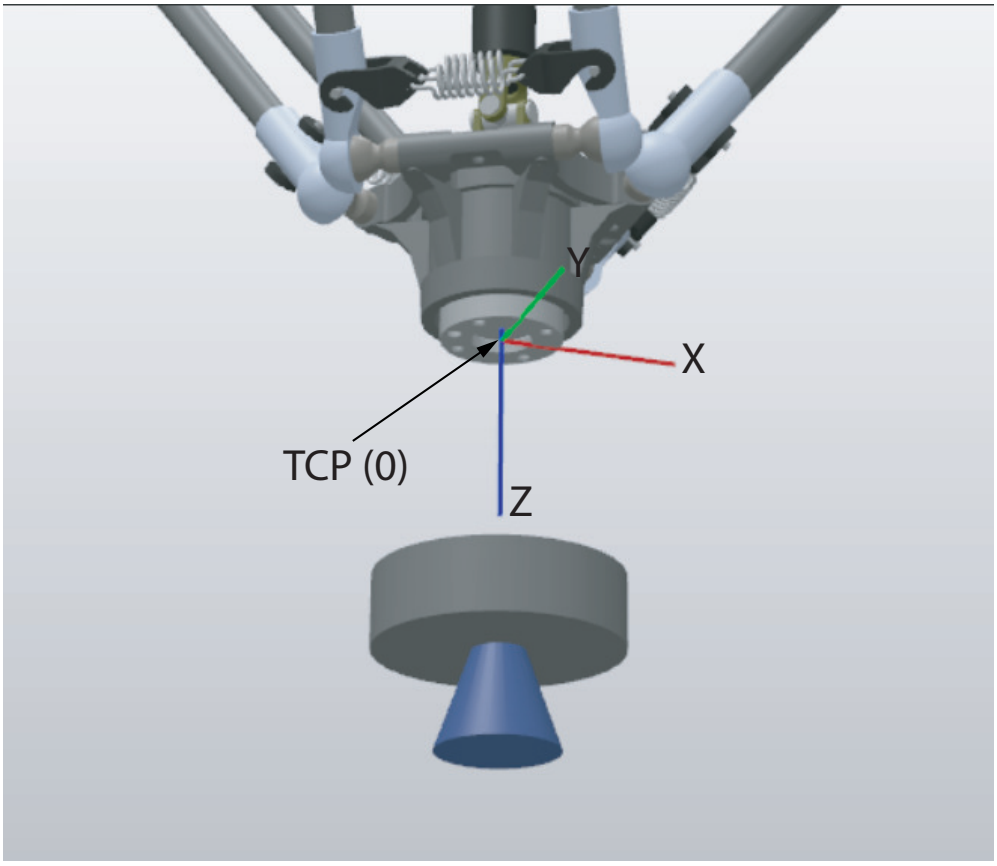
物体の寸法はこの座標系に対して定義されます。WCSなど、他の座標系から物体のポジションが参照される場合、基準点として物体のローカル原点が使用されます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

TCP(0)

ツール中心ポジション(0)は、リスト座標系(ツール0)で表したツール座標系の原点ポジションです。



xx2200001148

次のページに続く

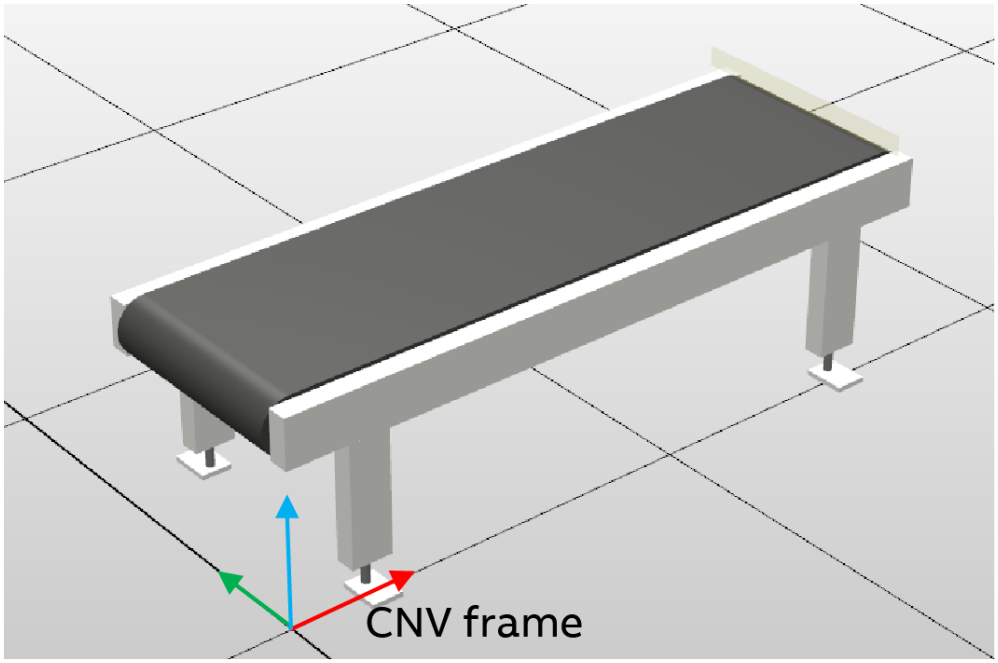
© Copyright 2025 ABB. All rights reserved.

4.2.2 PickMaster® Twin のフレーム関係
続き

コンベアフレーム

リニアコンベア用

コンベアフレームとしてリニアコンベアの下隅に配置されるフレーム。 このフレームは、コンベアに対して固定されています。 コンベアのポジションは、コンベア フレームとワールド フレームで表されるワールド フレーム間の距離 (3 次元) として定義されます。 コンベアの向きは、コンベア フレームとワールド フレームで表されるワールド フレームの間の角度として定義されます。 コンベア フレームは、コンベアがPickMaster PowerPacソリューション内のどこにあるかを定義するために使用されますが、ロボット コントローラー システムでは直接使用されません。



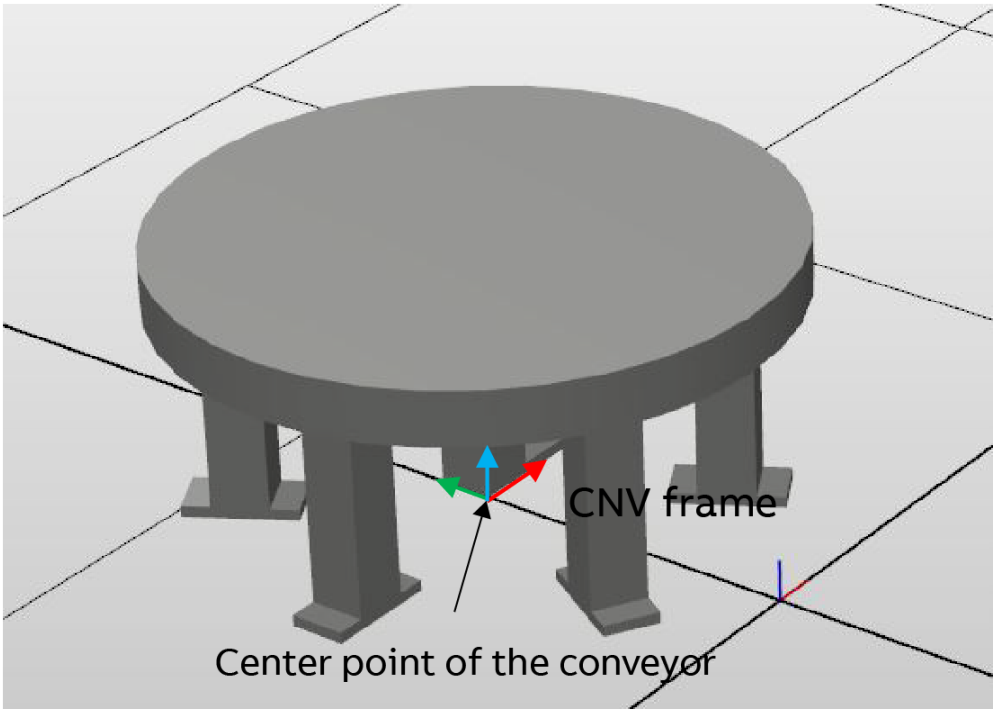
xx2400000394

次のページに続く

© Copyright 2025 ABB. All rights reserved.

循環コンベア用

円形コンベアの下部中央にコンベアフレームとして取り付けるフレームです。 このフレームは、コンベアに対して固定されています。 コンベアのポジションは、コンベア フレームとワールド フレームで表されるワールド フレーム間の距離 (3 次元) として定義されます。 コンベアの向きは、コンベア フレームとワールド フレームで表されるワールド フレームの間の角度として定義されます。 コンベア フレームは、コンベアがPickMaster PowerPacソリューション内のどこにあるかを定義するために使用されますが、ロボット コントローラ システムでは直接使用されません。



xx2400000395

ホットスポットフレーム

ホットスポットはコンベアに取り付けられたフレームですが、ユーザーが設定でき、コンベア フレーム内で表現されます。

アイテムまたはコンテナがソリューションで生成されるか、エミュレーションで出てくる場所です。

x、y、z および角度 Z の定義済みの値は、アイテムまたはコンテナがホットスポット フレームに出てくる場所を示します。

各コンベアのツリー ビューからホットスポットの場所を定義します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

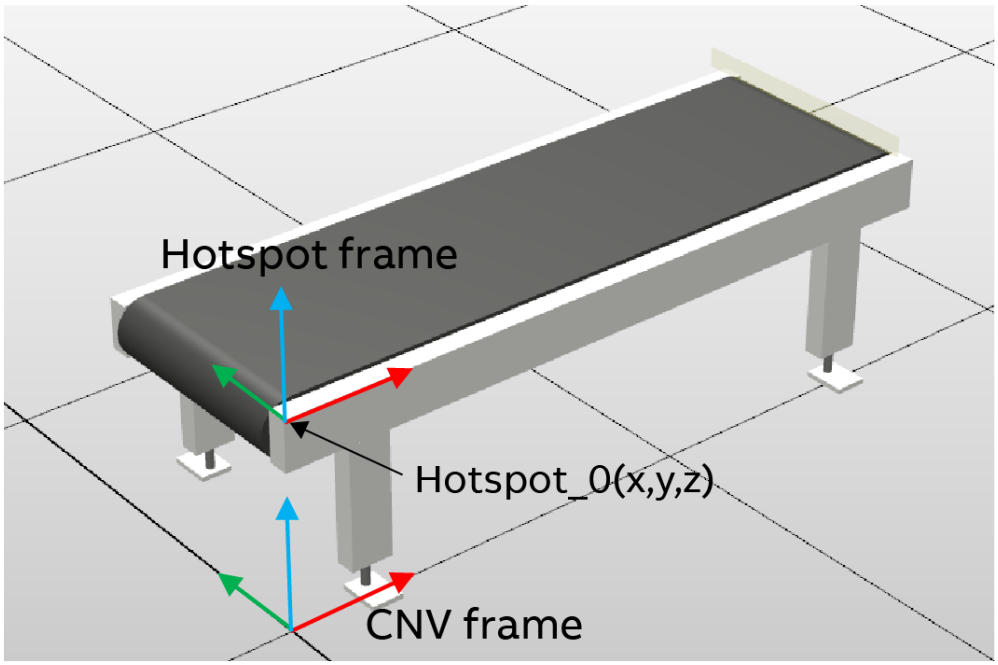
4.2.2 PickMaster® Twin のフレーム関係
続き



xx2400000398

リニアコンベア用

ホットスポット フレームの軸は常にコンベア フレームの軸と平行で、場所を設定できます。 ホットスポット フレームの向きは設定できません。



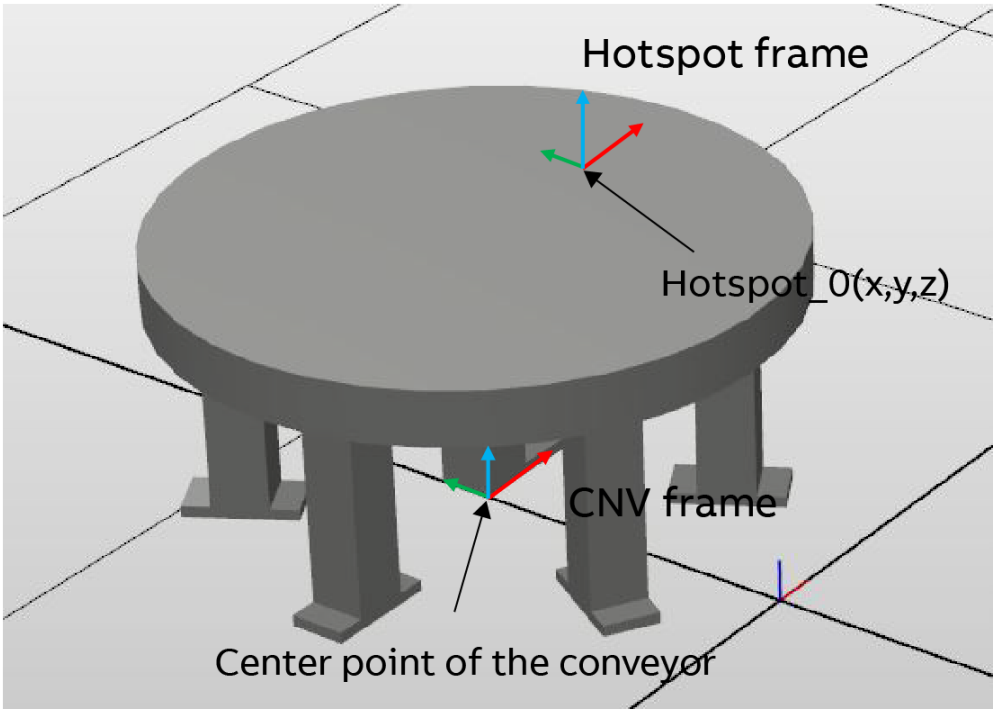
xx2400000396

次のページに続く

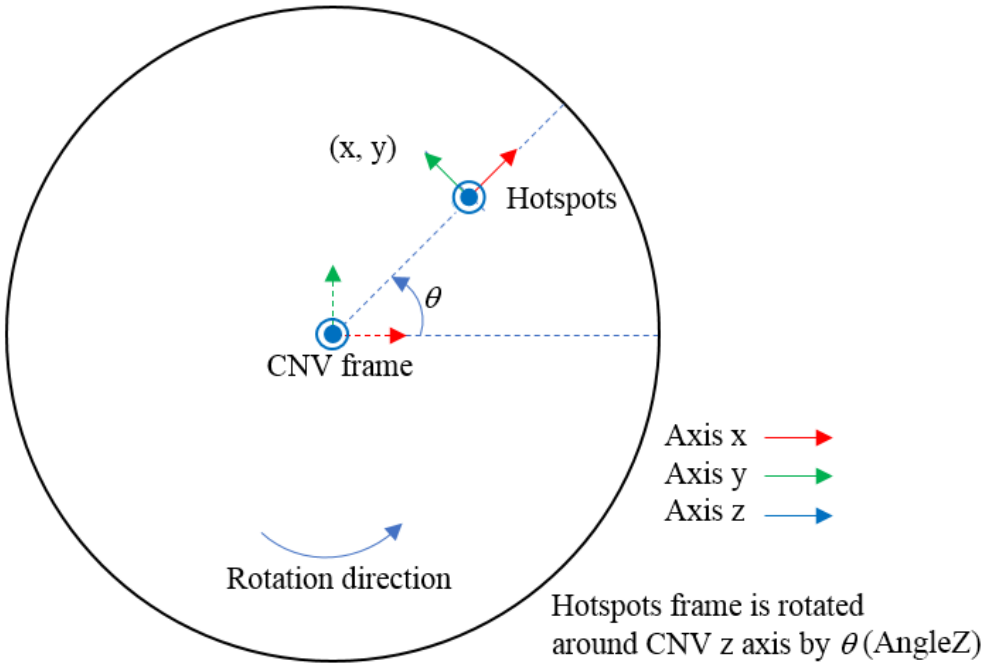
© Copyright 2025 ABB. All rights reserved.

循環コンベア用

ホットスポット フレームの X 軸は常に円形コンベアの半径に沿っており、外側を向いています。ホットスポット フレームの Z 軸は、コンベア フレームの Z 軸と平行です。



xx2400000397



xx2100002385

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.2.2 PickMaster® Twin のフレーム関係
続き

ホットスポット内の生成ポジションと角度の定義

ユーザーは、アイテムまたはコンテナを定義するときに、ホットスポット フレーム内のアイテムまたはコンテナの平面上の生成ポジションと角度を定義できます。



xx2100002389

注記

円形コンベアのトリガー タイプとして I/O センサーを使用する場合、ホットスポット フレームでアイテムまたはコンテナを生成する角度 Z を変更すると、誤トリガーが発生する可能性があります。これがピッキングミスの原因となります。

コンベア ベース フレーム

コンベア ベース フレームは、ロボットのベース フレームに対するコンベアの場所と向きを定義します。 コンセプトは、ABB のコンベア トラッキング製品から来ています。

このフレームは、コンベアがどこにあるかをロボットに伝えるためのものであり、コンベア上のすべてのアイテムを表現するために使用されます。 ロボットにアイテムの場所を「認識」させるには、まずコンベア ベース フレームを定義する必要があります。次に、アイテムの場所と向きを特定のセンサーで検出し、コンベア ベース フレームで表現する必要があります。 コンベア ベース フレームは、アイテムの場所と向きを計算するために直接使用されますが、PickMaster PowerPacソリューションでは明示的に使用されません。 シミュレーションでは、PickMaster PowerPac のキャリブレーション ボタンをクリックして、コンベア ベース フレームを決定します。 実際のシステムでは、コンベアのベース フレームは、リアル世界での特定の測定値によって決定されます。

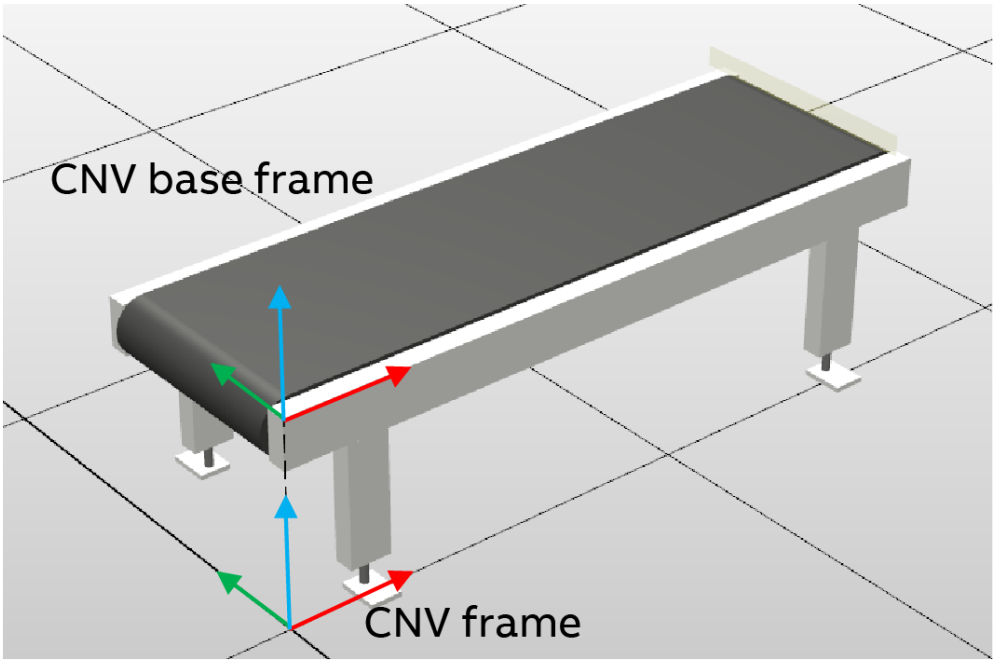
リニアコンベア用

リニアコンベアの X、Y、Z 軸は、それぞれコンベアフレームの X、Y、Z 軸と常に平行です。

コンベアベースフレームのポジションは、ソースの種類とコンベアのトリガー設定によって異なります。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

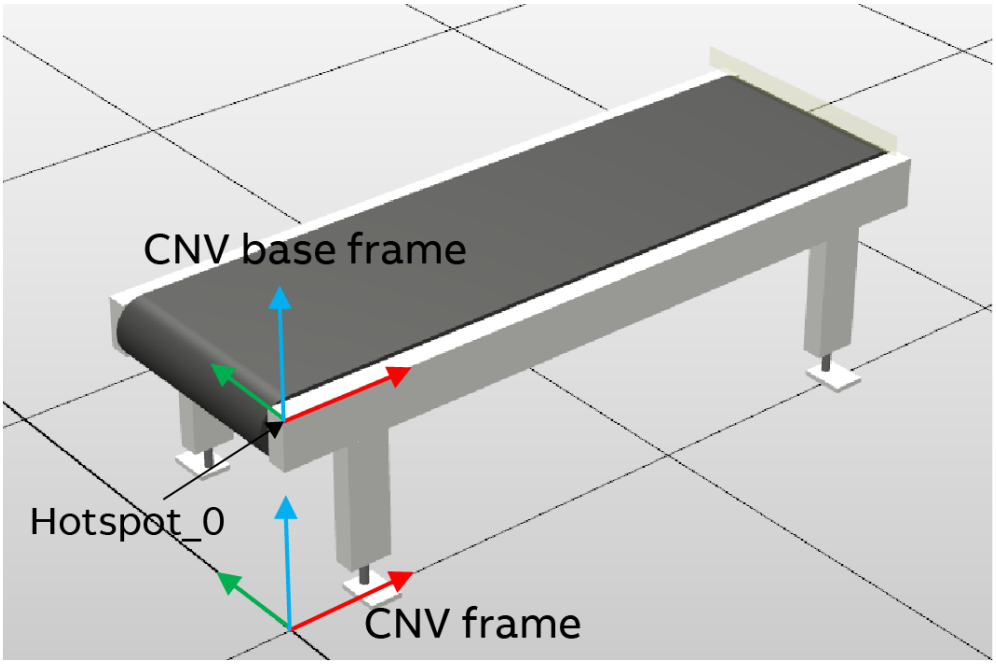


xx2100002388

ソースタイプは事前定義済み、トリガー設定は距離 (カメラと I/O センサーなし)

X 軸、Y 軸、Z 軸は、それぞれコンベア フレームの軸と平行です。

コンベア ベース フレームのポジションは Hotspot_0 です。



xx2100002469

ソースタイプはビジョン、トリガー設定は距離または I/O (カメラが使用されます)

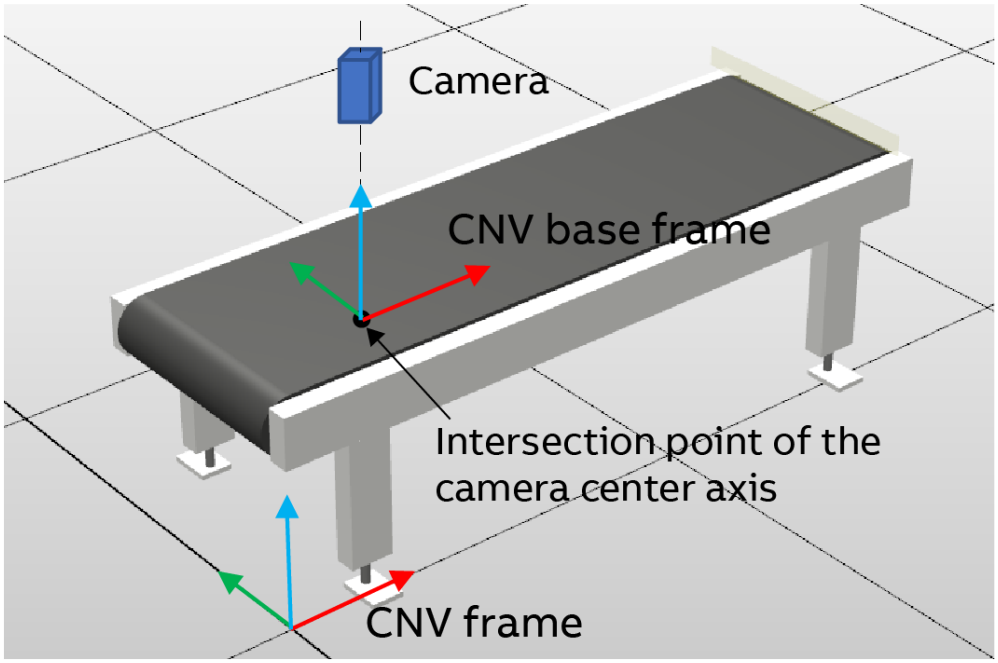
X 軸、Y 軸、Z 軸は、それぞれコンベア フレームの軸と平行です。

コンベア ベース フレームの場所は、カメラ中心軸とコンベア上面の交点です。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.2.2 PickMaster® Twin のフレーム関係
続き



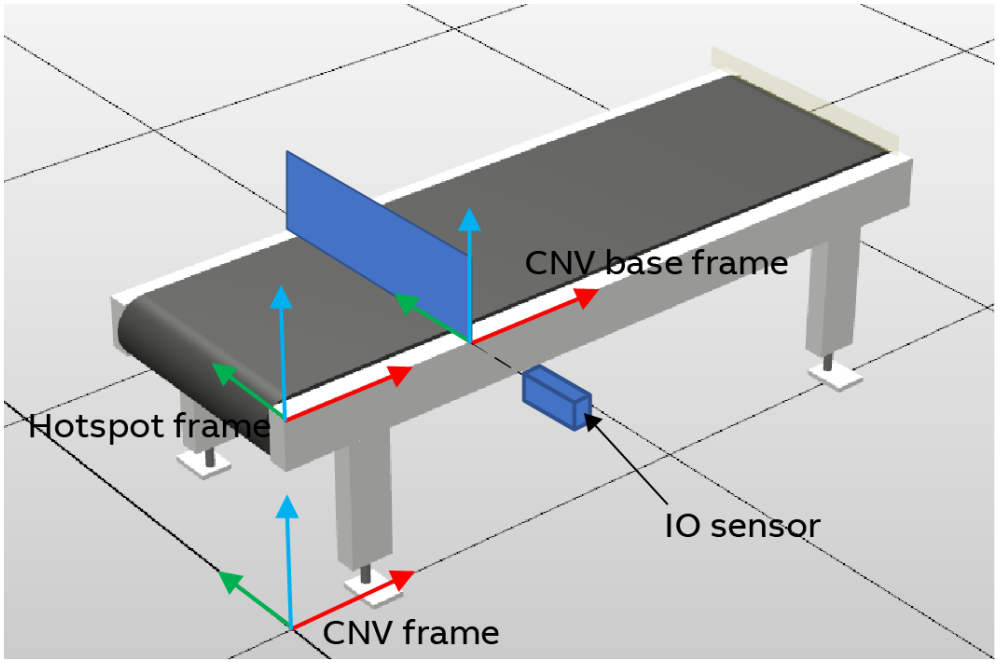
xx2100002470

ソース タイプを事前定義、トリガー設定を I/O (I/O センサーと事前定義されたソース タイプ)

X 軸、Y 軸、Z 軸は、それぞれコンベア フレームの軸と平行です。

コンベア フレームの X に沿った場所は、I/O センサーによって決定されます。

コンベア フレームの Y および Z に沿ったポジションは、hotspot_0 によって決定されます。

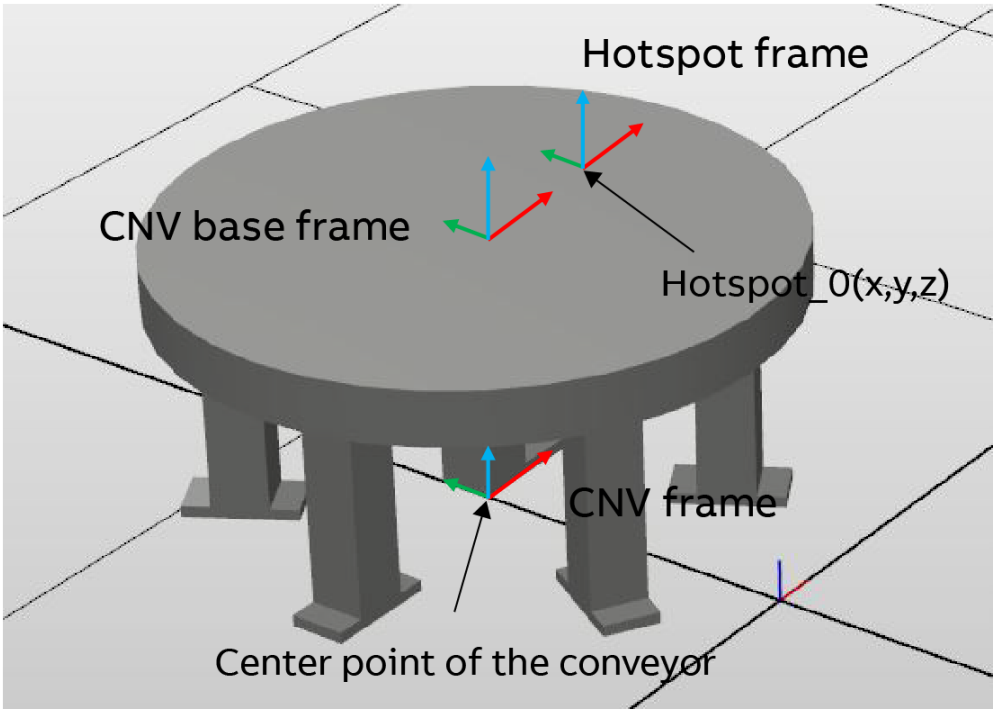


xx2100002471

次のページに続く

© Copyright 2025 ABB. All rights reserved.

反時計回りの円形コンベアの場合



xx2100002386

注記

コンベア ベース フレームのZ軸は、右手の法則を使用して正の回転方向を定義します。

ヒント

円形 CNV の I/O センサーは常に半径上にあり、内側を向いています。

ソースタイプは事前定義済み、トリガー設定は距離 (カメラと I/O センサーなし)

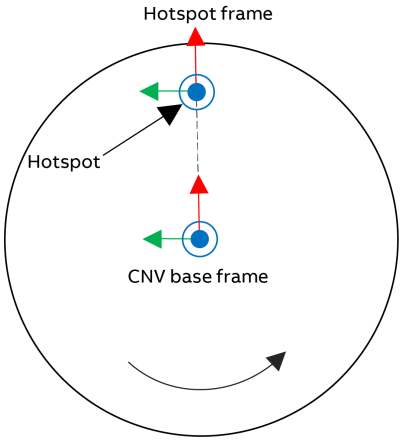
ベースフレーム X はhotspot_0を指します。

コンベアフレームがhotspot_0によって決定される場合、Z に沿ったポジション。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.2.2 PickMaster® Twin のフレーム関係
続き

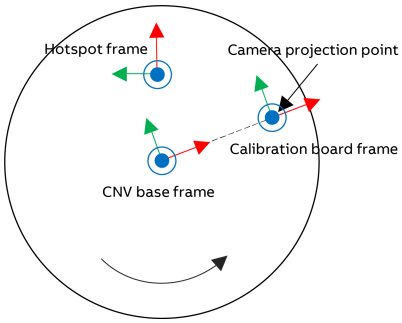


xx2100002466

ソースタイプとしてビジョン、トリガー設定として距離 (カメラを使用)

ベース フレーム X は、カメラの中心軸とコンベアの上面の交点を指します。

場所はコンベア上面です。

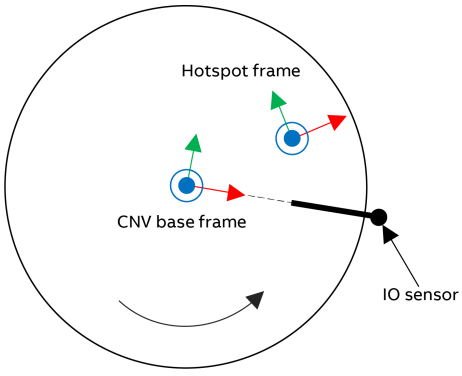


xx2100002467

ソース タイプを事前定義、トリガー設定を I/O (I/O センサーと事前定義されたソース タイプ)

ベース フレーム X は I/O センサーの方向を指します。

コンベアフレームの Z に沿ったポジションは、hotspot_0によって決定されます。



xx2100002468

次のページに続く

© Copyright 2025 ABB. All rights reserved.

時計回りの円形コンベアの場合

注記

コンベア ベース フレームのZ軸は、右手の法則を使用して正の回転方向を定義します。

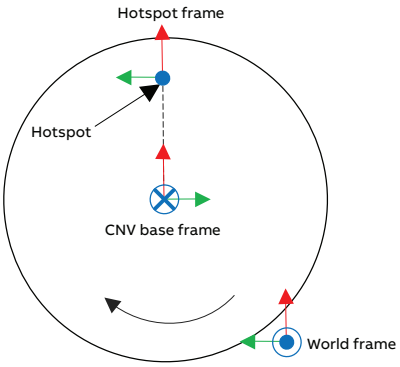
ヒント

円形 CNV の I/O センサーは常に半径上にあり、内側を向いています。

ソースタイプは事前定義済み、トリガー設定は距離 (カメラと I/O センサーなし)

ベースフレーム X はhotspot_0を指します。

コンベアフレームがhotspot_0によって決定される場合、Z に沿ったポジション。

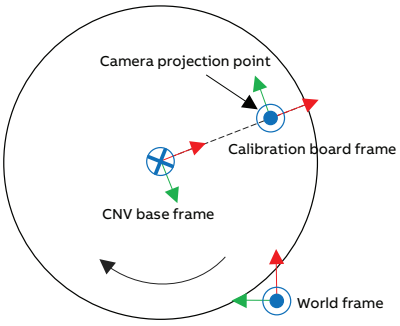


xx2400000402

ソースタイプとしてビジョン、トリガー設定として距離 (カメラを使用)

ベース フレーム X は、カメラの中心軸とコンベアの上面の交点を指します。

場所はコンベア上面です。



xx2400000403

ソース タイプを事前定義、トリガー設定を I/O (I/O センサーと事前定義されたソース タイプ)

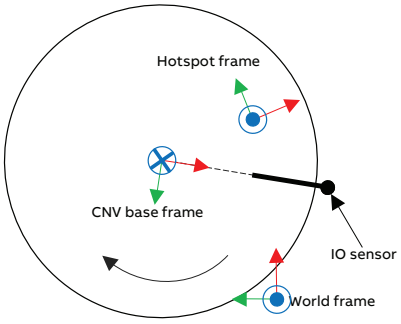
ベース フレーム X は I/O センサーの方向を指します。

コンベアフレームの Z に沿ったポジションは、hotspot_0によって決定されます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.2.2 PickMaster® Twin のフレーム関係
続き



xx2400000404

ベースフレームの調整の詳細については、ポジションジェネレータ ページ*136*を参照してください。

© Copyright 2025 ABB. All rights reserved.

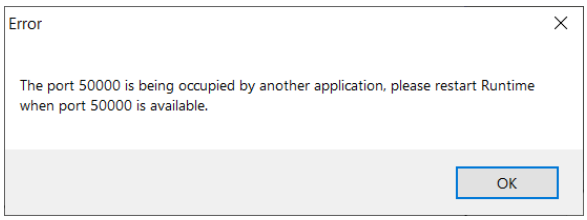
4.3 VirtualRuntime（VRT）でのレイアウトおよびプロセスでソリューションを設定

4.3.1 ソリューション

ヒント

PickMaster® Runtime (VRT および RRT) は、50000 ポートを使用するように定義されています。 50000 ポートが他のプログラムによって占有されている場合、この警告が表示され、Runtimeに接続できなくなります。

50000 ポートを解放し、PickMaster® Runtimeを再起動します。



xx2100000868

この手順を使用して、50000 ポートを解放します。

1. コマンド `netstat -aon|findstr "50000"` を CMD ウィンドウに入力します。
2. ポート 50000 を占有するプロセスがウィンドウに一覧表示されます。 プロセスの PID コードを取得します。
3. タスク マネージャでこの PID に対応するプロセスを見つけて閉じます (このコンピュータでこのプロセスを閉じることが許可されていることを確認してください)。
4. PickMaster® Runtime を再起動して接続します。

注記

PickMaster PowerPacの最初の操作でSSLダイアログボックスがポップアップ表示されたら、はいをクリックします。

そうしないと、PickMaster PowerPac は正常に動作しません。

注記

モデルが不足しているダイアログ ボックスが表示されたら、はいをクリックしてモデルをダウンロードします。そうしないと、ソリューションは正常に機能しません。

モデルのダウンロードには数分かかる場合があります。

ソリューションを開く

ユーザーは、ファイルリボンのタブから新しいソリューションを作成したり、既存のソリューションを開いたりすることができる。

詳細については、ファイル ページ *69*を参照してください。

© Copyright 2025 ABB. All rights reserved.

4.3.2 レイアウト

4.3.2.1 コントローラ

概要

このセクションでは、コントローラを追加および変更する方法について説明します。

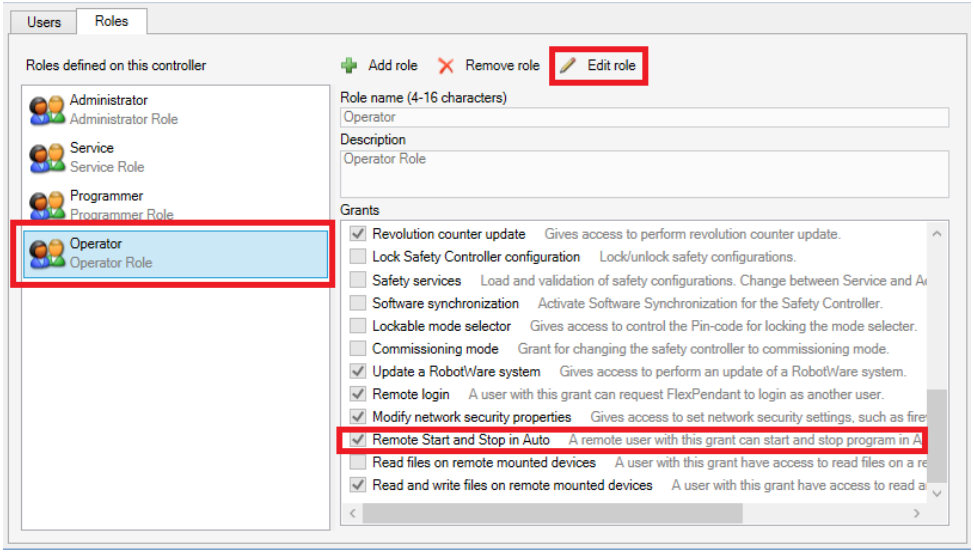
コントローラを作成する

コントローラの作成方法の詳細については、ユーザーマニュアル - *RobotStudio* を参照してください。

警告

OmniCore コントローラを作成するときは、自動でのリモート開始と停止を選択する必要があります。

そうしないと、RobotWare 権限の制限により、コントローラーを PMPP および PMOP を介してリモートで開始することができません。 この場合、シミュレーションと本番は正常に動作しません。



xx2100000051

注記

コントローラは、ソリューションに追加する前に作成する必要があります。

コントローラの追加

ソリューションにコントローラを追加するには、リボンのコントローラをクリックします。

以下の表に、コントローラ追加ダイアログボックスの詳細を示す。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

注記

このページが開かれる前に作成されたコントローラ のみが仮想コントローラ に表示されます。

新しいコントローラ が作成された場合、ユーザーはコントローラ ダイアログ ボックスを更新する必要があります。

注記

シミュレーションの前に、仮想コントローラが他のソリューションで使用されている場合は、ソリューションをキャリブレーションすることをお勧めします。

異なるソリューションが同じ仮想コントローラを使用している場合、1 つのソリューションのコントローラを変更すると、他のソリューションに影響します。これにより、他のソリューションの予期しない誤解を招く動作が発生します。

	説明
ポジション	場所 は、必要なコントローラ システムが保存されている PC の場所とフォルダを指定します。
管理...	ロボットシステムを管理します。
仮想コントローラ	選択したシステム フォルダで見つかったシステムを一覧表示します。
システムのリセット(I-start)	これを選択すると、コントローラがリセットされます。 **注記** すべてのパラメータと設定が工場出荷時の値に復元されます。
新規ライブラリのインポート	事前定義済みのロボットをPickMaster PowerPac に追加します。
既存のステーション ライブラリを使用	RobotStudio から既存のロボットを開きます。
RAPID プログラムをステーションに同期	RAPID プログラムをソリューションに同期します。

プロシージャ

PickMaster PowerPac リボン タブで、**レイアウト** をクリックします。

次の手順を使用してコントローラを追加します。

注記

ソリューションに追加できるのは、システム内の既存コントローラに限られる。コントローラ作成の詳細については、コントローラを作成する ページ*116*を参照。

1 リボン タブで、**コントローラ** をクリックします。

コントローラ追加ダイアログ ボックスが開く。

2 場所 リストにフォルダを追加するには、**...**ボタンをクリックし、追加するフォルダを参照して選択します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.3.2.1 コントローラ
続き

3 **仮想コントローラ** には、選択したシステム フォルダにある仮想コントローラ システムが一覧表示されます。 システムをクリックして、新しいソリューションを選択します。

4 **オプション** で必要なチェック ボックスをオンにします。

> **注記**
>
> システムビルダーの**システムの変更**を使用して変更された仮想コントローラ システムは、変更を適用するには、**システムのリセットオプション**で再起動する必要があります。

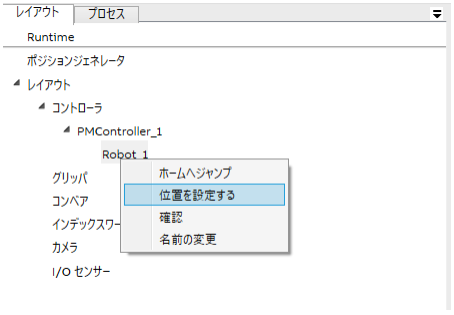
5 ダイアログ ボックスで **OK** をクリックして、選択したコントローラをソリューションに追加します。

選択したコントローラがソリューションに追加されます。 新しく追加されたコントローラが、**レイアウト** ウィンドウの**コントローラ** リストに表示されます。

6 **ロボット** ウィンドウで、必要に応じて **ポジション X Y Z (mm)** テキスト ボックスと **向き (度)** テキスト ボックスに数字を入力します。

7 ［**OK**］をクリックします。

> **注記**
>
> 他のパーツを追加することでポジションを調整可能。
>
> **コントローラーリスト**のロボット1を右クリックします。



xx2100001532

© Copyright 2025 ABB. All rights reserved.

4.3.2.2 グリッパー

概要

このセクションでは、グリッパーを追加する方法について説明します。

グリッパーの追加

ソリューションにグリッパーを追加するには、リボンのグリッパーをクリックします。

次の表は、グリッパー設定ダイアログボックスの詳細を示す。

	説明
インポート	ライブラリから事前定義済みのグリッパーをインポートするか、ユーザー定義のグリッパーをライブラリにアップロードします。 注記 ユーザー定義のグリッパーをアップロードするには、**ライブラリに追加** をクリックし、ローカル フォルダーを参照して `*.rslib` ファイルを選択します。グリッパーが自動的にライブラリに追加されます。
グリッパー名	グリッパーの名前を設定。
使用可能なコントローラ	[コントローラ] リストからコントローラを選択します。
使用可能なロボット	[ロボット] リストからロボットを選択します。
基準座標	グリッパーの基準座標を選択します。
位置 (XYZ)[mm]	グリッパーのポジションを設定します。
方向 [度]	グリッパーの向きを設定します。
モデルの不透明度	グリッパーの不透明度を設定します。 注記 **モデルの不透明度**の設定を変更すると、設定ウィンドウの **[OK]** ボタンをクリックしなくても自動的に保存されます。 注記 不透明度を0% に設定すると、コンポーネントはステーション ビューで非表示になり、ツリー ビューに (非表示) マークは表示されません。
質量設定	詳細については、次のセクションを参照してください。
アクティベータ設定	詳細については、次のセクションを参照してください。

質量設定

質量設定	説明
デフォルトを使用	質量設定はデフォルトの設定を使用します。
質量	質量 (kg) フィールドにツールの質量を入力します。
重心	重心の座標を入力します。
慣性	慣性 (kgm^2) に慣性の値を入力します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.3.2.2 グリッパー
続き

アクティベータ設定

アクティベータ設定	説明
アクティベータの使用	使用するアクティベータを選択します。
追加 ボタン	新しいアクティベータを追加します。 注記 複数のピックを行う必要がある場合は、ピックごとに十分な数のアクティベータを追加する必要があります。 たとえば、4 つのアイテムを選択して配置する必要がある場合は、デフォルトのアクティベータに加えて、さらに 3 つのアクティベータを追加する必要があります。 複数ピックを行うには、インストール パッケージ内の複数ピック ラピッド ファイルを、必要なロボットのレシピにインポートする必要があります。
削除 ボタン	選択したアクティベータを削除します。
Rapid ツール データ	ツールデータからRAPID ツールデータを選択します。 選択したツールデータは、このアクティベータでピッキングするときに RAPID プログラムによって使用されます。 注記 複数のアクティベータを使用する場合は、RAPID プログラムを更新する必要があります。 詳細については、例：ダブルピック・シングルプレイス ページ 432 を参照してください。
TCP ポジション	ツールの中心点の座標を入力します。ツールの中心点は、アイテムが取り付けられるツール上のポジションを定義します。 注記 座標は、シミュレーション中に選択したツールデータに適用されます。
TCP 向き	ツールの中心点の向きを入力します。TCP 向きは、アイテムをピックアップする際のツールの望ましい向きを定義します。 向きは、オイラー XYZ 角度 (度) として指定する必要があります。 注記 向きは、シミュレーション中に選択したツールデータに適用されます。
アクティベータ シグナル タイプ	信号のデフォルト設定またはカスタマイズ設定を選択。 注記 PickMaster PowerPacのアクティベータ シグナル設定は、接続されたコントローラのシグナル設定とまったく同じでなければなりません。 つまりPickMaster PowerPacで、グリッパーはアイテムを選択または配置しない。
デフォルトの設定	信号の詳細なデフォルト設定を表示。
カスタマイズされた設定	信号の詳細なカスタマイズ設定を表示し、ユーザーによる信号の変更を可能にします。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

プロシージャ

次の手順を使用してグリッパーを追加します。

PickMaster PowerPac リボン タブで、**レイアウト** をクリックします。

1 リボン タブで、**グリッパー** をクリックします。
グリッパーウィンドウが開きます。
2 **グリッパーウィンドウで、グリッパー名テキスト ボックスに名前を入力するか、デフォルトの名前を使用します。**
3 **グリッパーウィンドウで、質量設定とアクティベータ設定にデフォルトを使用します。**
4 **［OK］をクリックします。**

© Copyright 2025 ABB. All rights reserved.

4.3.2.3 コンベヤ

概要

このセクションでは、コンベアを追加する方法について説明します。

コンベアの追加

ソリューションにコンベアを追加するには、リボンのコンベアをクリックします。
次の表は、コンベア設定ダイアログボックスの詳細を示す。

	説明
コンベア名	コンベアの名前を設定。 ヒント 名前が現在のタスクで一意であることを確認してください。
コンベア タイプ	ライナーコンベアまたは円形コンベアを選択します。
サイズ(x,y,z)[mm][i]/RHサイズ[mm mm][ii]	コンベアのサイズを定義します。 • リニアコンベアの場合、値は次の範囲内である必要があります。 長さ：1,700 mm - 1,000,000 mm 幅：110 mm - 1,000,000 mm 高さ：210 mm - 1,000,000 mm • 円形コンベアの場合、値は次の範囲内である必要があります。 半径：251 mm - 1,000,000 mm 高さ：210 mm - 1,000,000 mm
方向[ii]	円形コンベアの回転方向を選択します。
基準座標	コンベアの基準座標を選択します。
位置 (XYZ)[mm]	コンベアのポジションを設定します。
方向 [度]	コンベアの向きを設定します。
モデルの不透明度	コンベアの不透明度を設定します。 注記 モデルの不透明度の設定を変更すると、設定ウィンドウの [OK] ボタンをクリックしなくても自動的に保存されます。 注記 不透明度を0% に設定すると、コンポーネントはステーションビューで非表示になり、ツリー ビューに (非表示) マークは表示されません。

注記

円形コンベアとカメラまたは I/O センサーを同時に使用する場合、カメラまたは I/O センサーは、コンベアのホットスポットと回転方向の最初のロボットの間に設定する必要があります。
つまり、ロボットがアイテムを認識しない可能性がある。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

プロシージャ

PickMaster PowerPac リボン タブで、**レイアウト** をクリックします。

次の手順を使用してコンベアを追加します。

1. リボン タブで、**コンベア** をクリックします。
   コンベアウィンドウが開きます。
2. **コンベアウィンドウで、コンベア名テキスト** ボックスに名前を入力するか、デフォルトの名前を使用します。
3. **コンベアウィンドウで、コンベア タイプドロップダウン** リストからタイプをライナーまたは円形として選択します。
4. ライナー コンベヤを選択した場合は、**コンベアウィンドウで、サイズ (x,y,z)[mm]テキスト** ボックスに数値を入力して、要件に従ってコンベアのサイズを定義します。
5. 円形コンベヤを選択した場合は、**コンベアウィンドウで、RH サイズ(mm)テキスト** ボックスに数値を入力して、要件に従ってコンベアのサイズを定義します。
6. **コンベアウィンドウで、ポジション XYZ(mm)テキスト** ボックスと **向き XYZ(度)テキスト** ボックスに数値を入力して、要件に従ってコンベアのポジションを定義します。
7. ［**OK**］をクリックします。

© Copyright 2025 ABB. All rights reserved.

4.3.2.4 カメラ

概要

このセクションでは、カメラを追加する方法について説明します。

カメラの追加

ソリューションにカメラを追加するには、リボンの**カメラ** をクリックします。

以下の表は、**カメラ**設定ダイアログボックスの詳細を示す。

	説明
名前	カメラの名前を設定。  ヒント 名前が現在のタスクで一意であることを確認してください。
コンベア/インデックスに装着	センサーをコンベアに取り付ける場合は、コンベアを選択します。
入口[mm]	コンベアに沿ったカメラの下の可視エリアの入口制限を入力します。可視エリアがカメラの場所から上流で始まる場合は、負の数値が使用されます。
出口[mm]	コンベアに沿ったカメラの下の可視エリアの出口制限を入力します。可視エリアがカメラの場所の下流で終了する場合は、正の数値が使用されます。
ビジョン幅の有効化	可視エリアの幅制限を有効にするには、これを選択します。  注記 **ビジョン幅の有効化** チェックボックスが選択されている場合にのみ、**左 (mm)** と**右 (mm)** の値が設定に入ります。  注記 カメラが円形コンベヤに取り付けられている場合、**ビジョン幅の有効化**チェックボックスはデフォルトで選択されており、無効にすることはできません。
左[mm]	可視エリアの左側の制限値を入力します。可視エリアがカメラ場所の左側で終了する場合 (上流の視点から)、負の値が使用されます。  注記 **左 (mm)** と**右 (mm)** が正しく設定されていないと、ロボットが空気をキャッチしたり、一部のアイテムを見逃したりする可能性があります。
右[mm]	可視エリアの右側の制限値を入力します。可視エリアがカメラ場所の右側で終了する場合 (上流の視点から)、正の値が使用されます。  注記 **左 (mm)** と**右 (mm)** が正しく設定されていないと、ロボットが空気をキャッチしたり、一部のアイテムを見逃したりする可能性があります。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

	説明
基準座標	カメラの基準座標を選択します。
ポジション (X、Y、Z) [mm]	カメラのポジションを設定します。
方向 [度]	カメラの向きを設定します。
モデルの不透明度	コンベアの不透明度を設定します。 注記 モデルの不透明度の設定を変更すると、設定ウィンドウの [OK] ボタンをクリックしなくても自動的に保存されます。 注記 不透明度を0% に設定すると、コンポーネントはステーションビューで非表示になり、ツリー ビューに (非表示) マークは表示されません。

注記

カメラがインデックスワークエリアで使用されている場合、可視エリアは制限されません。

注記

カメラは、他のコンベアまたはインデックスワークエリアで作成または配置された物体を検出しません。

注記

円形コンベアとカメラまたは I/O センサーを同時に使用する場合、カメラまたは I/O センサーは、コンベアのホットスポットと回転方向の最初のロボットの間に設定する必要があります。

つまり、ロボットがアイテムを認識しない可能性がある。

プロシージャ

PickMaster PowerPac リボン タブで、**レイアウト** をクリックします。

カメラを追加するには、次の手順に従います。

1. リボン タブで、**カメラ**をクリックします。
   カメラウィンドウが開きます。
2. **カメラ**ウィンドウで、**カメラ名**テキスト ボックスに名前を入力するか、デフォルトの名前を使用します。
3. **カメラ**ウィンドウで、**コンベア/インデックス**に装着ボックスでコンベヤを選択し、要件に応じて新しいカメラを取り付けるコンベアを定義します。
4. **カメラ**ウィンドウで、その他の設定にはデフォルトを使用します。
5. [OK] をクリックします。

© Copyright 2025 ABB. All rights reserved.

4.3.2.5 I/Oセンサー

概要

このセクションでは、I/O センサーを追加する方法について説明します。

I/O センサーの追加

ソリューションに I/O センサーを追加するには、リボンの**I/O センサー**をクリックします。

以下の表は、**I/Oセンサー設定**ダイアログボックスの詳細を示す。

	説明
名前	I/O センサーの名前を設定します。 **ヒント** 名前が現在のタスクで一意であることを確認してください。
LHサイズ[mm]	新しい I/O センサーの高さと長さ。
コンベア/インデックスワークエリア	センサーをコンベアに取り付ける場合は、コンベアを選択してください。
基準座標	I/O センサーの基準座標を選択します。
ポジション (X、Y、Z) [mm]	I/O センサーのポジションを設定します。
方向 [度]	I/O センサーの向きを設定します。

注記

正しく機能するためには、I/O センサーが他の静止物体 (コンベアなど) と接触していてはなりません。

注記

円形コンベアとカメラまたは I/O センサーを同時に使用する場合、カメラまたは I/O センサーは、コンベアのホットスポットと回転方向の最初のロボットの間に設定する必要があります。

つまり、ロボットがアイテムを認識しない可能性がある。

プロシージャ

PickMaster PowerPac リボン タブで、**レイアウト** をクリックします。

本手順を用いて I/O センサーを追加する：

1. リボン タブで、**I/O センサー**をクリックします。
   I/O センサーウィンドウが開きます。
2. **I/O センサーウィンドウ**で、**I/O センサー名テキスト** ボックスに名前を入力するか、デフォルトの名前を使用します。
3. **I/O センサーウィンドウ**で、**I/O センサーの高さテキスト** ボックスに数値を入力して、要件に従って I/O センサーの高さを定義するか、デフォルト設定を使用します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4 I/O センサーウィンドウで、I/O センサーの長さテキスト ボックスに数値を入力して、要件に従ってI/O センサーの長さを定義するか、デフォルト設定を使用します。
5 I/O センサーウィンドウで、コンベア/インデックスに取り付けボックスでコンベヤを選択し、要件に応じて新しいカメラをどのコンベアに取り付けるかを定義します。
6 ［OK］をクリックします。

© Copyright 2025 ABB. All rights reserved.

4.3.2.6 External sensor（外部センサー）

概要

外部センサーは、ユーザーがアイテムポジションの生成を完全にコントロールできる機能である。

本セクションでは、あらゆる種類の検知デバイスまたは純粋なバーチャルソフトウェアセンサーを使用して外部センサーを追加する方法について説明します。

外部センサーの追加

ソリューションに外部センサーを追加するには、リボンの**外部センサー**をクリックします。

以下の表は、**外部センサー設定**ダイアログボックスの詳細を示す。

	説明
名前	外部センサーの名前を設定。  **ヒント** 名前が現在のタスクで一意であることを確認してください。
コンベア/インデックスワークエリア	センサーをコンベアに取り付ける場合は、コンベアを選択してください。
基準座標	外部センサーの基準座標を選択。
ポジション　(X、Y、Z) [mm]	外部センサーのポジションを設定。
方向 [度]	外部センサーの方向を設定。
モデルの不透明度	コンベアの不透明度を設定します。  **注記** **モデルの不透明度**の設定を変更すると、設定ウィンドウの **[OK]** ボタンをクリックしなくても自動的に保存されます。  **注記** 不透明度を0% に設定すると、コンポーネントはステーションビューで非表示になり、ツリー ビューに **(非表示)** マークは表示されません。

プロシージャ

PickMaster PowerPac リボン タブで、**レイアウト** をクリックします。

本手順で外部センサーを追加。

1 リボンタブで、**外部センサー**をクリックします。

外部センサーウィンドウが開く。

2 **外部センサーのウィンドウで、外部センサーの名前のテキスト** ボックスに名前を入力するか、デフォルトの名前を使用します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

3 外部センサーのウィンドウで、コンベア/インデックスに取り付けボックスでコンベヤを選択し、要件に応じて新しいカメラをどのコンベアに取り付けるかを定義します。
4 ［OK］をクリックします。

Advanced function - External sensor

外部センサーはプログラミングユーザー向けの高度な機能です。詳細については、外部センサー ページ *357*を参照してください。

© Copyright 2025 ABB. All rights reserved.

4.3.2.7 作業範囲

概要

このセクションでは、作業エリアを追加する方法について説明します。

作業エリアを追加します。

コンベアの作業範囲は、ロボットがアイテムをピッキングまたは配置するコンベア上の範囲を指す。各作業範囲に1台のコンベア boardが必要となる。ロボットは通常、関連する各コンベア上に1つのコンベア作業範囲しか有さないが、制限はない。

ソリューションに作業範囲を追加するには、リボン上の **コンベア作業範囲**をクリックします。

次の表は、**コンベア作業範囲**の設定ダイアログボックスの詳細を示す。

注記

"*" を伴う全信号は必須信号である。

	説明
作業エリア名	コンベア作業範囲の名前を設定。
コントローラ	リストからコントローラを選択します。
ロボット	リストからロボットを選択します。
コンベアボード	リストからコンベア ボードを選択します。
コンベア	リストからコンベアを選択します。
作業エリアタイプ	使用可能なオプションから作業エリアのタイプを選択します。 • **拾う**：作業エリアがピッキングエリアである場合、これを選択します。 • **配置**：作業エリアが配置エリアの場合に選択します。
選択インデックス	インデックスを選択して、選択したロボットで複数のピック作業エリアと1つの配置作業エリアを使用する場合に、RAPIDプログラムでピックまたは配置の順序を指定します
シグナル タイプ	安全信号の設定。 デフォルトのシグナル設定を使用するには、**デフォルト設定** チェック ボックスを**オン**にします。 バーチャルRuntimeで作業する場合は、デフォルト設定の使用が推奨される。 **カスタマイズ設定**オプションを使用して信号を管理。 RealRuntimeで使用する場合は、信号を**カスタマイズ設定**にする必要がある。詳しくは「*I/Oの設定 ページ185*」を参照してください。

注記

コントローラ、**ロボット**、**コンベア**のいずれかの作業エリア設定を変更した場合、レシピ設定ページを再度開いて変更を有効にする必要があります。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

コンベア作業エリア シグナル

	説明
Conveyor start/stop（コンベヤの起動/停止)	デジタル出力シグナル。このシグナルは、コンベアの動きを作業エリアで制御することによってオーバーフローを回避する必要がある場合に使用されます。 オーバーフローを避けるために、コンベアが動き始めるとシグナルが高くなり、コンベアが停止するときにシグナルが低くなります。
Queue idle (キューのアイドル状態)	デジタル出力シグナル。この作業エリアのキューが空の場合、このシグナルはHighになります。 最後のアイテムがキューから取得されると、シグナルが高くなります。
Position available (使用可能なポジション)	デジタル出力シグナル。このシグナルは、作業エリアの開始制限と終了制限の間に 1 つ以上のアイテムがある場合にHighになります。
Position generator（ポジションジェネレータ)	画像認識やセンサを使って新しくポジションを生成するためのデジタル入力信号です。この信号は、間隔トリガコンベヤが使用されている場合は無視されます。
Trig（トリガ)	ビジョンを使用している場合、このデジタル出力シグナルはカメラの I/O ポートのトリガー入力に接続されなくてはなりません。事前定義済ポジションが使用されている場合、この出力シグナルはコンベアエンコーダボードの開始入力に直接接続されている必要があります。これは、*doManSyncX* シグナルを使用すると最適な状態で実行できます。事前定義ポジションがこの作業エリアのみに分配されている場合（例えば、ロボットが 1 台の Runtime）、エンコーダ シグナル *cXSoftSyncSig* を *doManSyncX* の代わりに使用できます。つまり、エンコーダボードの開始入力にシグナルを送る必要がないということです。
Strobe（ストロボ)	これは、ストロボ信号の入力信号名で、コンベヤのエンコーダーボードの開始信号です。信号名は*cXNewObjStrobe*に設定されています。ビジョンが使用されている場合、信号をカメラのストロボ出力と接続する必要があります。事前定義のポジションが使用されている場合、ストロボは*doManSyncX*信号から直接生成できます。この信号は、エンコーダーボードの開始信号に直接接続されています.

注記

DSQC2000、カメラまたは事前定義したソースで距離トリガーしたポジションソースを使用し、`cxTrigVis` を **トリガー** 信号として構成します。RW6.10 以降、ストロボ信号は自動構成されるため、作動領域信号構成では省かれます。

プロシージャ

PickMaster PowerPac リボン タブで、**レイアウト** をクリックします。

作業エリアに対して次の手順を使用します。

1. リボン タブで、**コンベア作業エリア** をクリックします。

   コンベア作業エリアウィンドウが開きます。
2. **コンベア作業エリア**ウィンドウで、**作業エリア名テキスト** ボックスに名前を入力するか、デフォルトの名前を使用します。
3. 複数のコントローラを作成した場合は、**コントローラ**で必要なコントローラを選択します。
4. 複数のロボットを作成した場合は、**ロボット**で必要なロボットを選択します。
5. **コンベア ボード**で必要なコンベア ボードを選択します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.3.2.7 作業範囲
続き

6 目的のコンベアを選択し、必要な作業エリアのタイプを設定して、設定を構成します。
7 ［OK］をクリックします。

© Copyright 2025 ABB. All rights reserved.

4.3.2.8 Indexed work area（インデックスワークエリア）

概要

このセクションでは、インデックスワークエリアを追加する方法について説明します。

インデックスワークエリアの追加

インデックス作業範囲は、ロボットがコンベアなしでアイテムをピックまたは配置する固定範囲です。

ソリューションにインデックス作業範囲を追加するには、リボン上のインデックス作業範囲をクリックします。

以下の表は、インデックスワークエリアの設定ダイアログボックスについての詳細を示す。

注記

"*" を伴う全信号は必須信号である。

	説明
作業エリア名	インデックスワークエリアの方向を設定。
サイズ	インデックスワークエリアのゾーンを定義します。
作業エリアタイプ	使用可能なオプションから作業エリアのタイプを選択します。 • 拾う：インデックスワークエリアがピッキング エリアである場合は、これを選択します。 • 配置：インデックスワークエリアが配置エリアである場合、これを選択します。
コントローラ	リストからコントローラを選択します。
ロボット	リストからロボットを選択します。
作業オブジェクト	RAPID 作業物体 データ (wobjdata) を選択します。 関連する wobjdata は、インデックスワークエリアで自動的に使用されます。 **注記** 作業物体のキャリブレーションは必要ありません。 選択した wobjdata は、シミュレーションが開始されると自動的に更新されます。
選択インデックス	インデックスを選択して、選択したロボットで複数のピック作業エリアと1つの配置作業エリアを使用する場合に、RAPID プログラムでピックまたは配置の順序を指定します。
基準座標	インデックスワークエリアの基準座標を選択します。
位置 (x,y,z)[mm]	インデックスワークエリアのポジションを設定します。
方向 [度]	インデックスワークエリアの向きを設定します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.3.2.8 Indexed work area（インデックスワークエリア）
続き

	説明
モデルの不透明度	コンベアの不透明度を設定します。 **注記** **モデルの不透明度**の設定を変更すると、設定ウィンドウの **[OK]** ボタンをクリックしなくても自動的に保存されます。 **注記** 不透明度を0% に設定すると、コンポーネントはステーション ビューで非表示になり、ツリー ビューに (非表示) マークは表示されません。
シグナル タイプ	信号を構成します。**カスタマイズ設定** オプションを使用してシグナルを管理します。インデックスワークエリア シグナルの詳細については、次のセクションを参照してください。 デフォルトのシグナル設定を使用するには、**デフォルト設定** チェック ボックスをオンにします。

インデックスワークエリア シグナル

注記

プロダクションでは、**キュー アイドル**シグナルとインデックスワークエリアの**ストロボ**シグナルを同じものとして設定し、その他のシグナルをブランクとして設定することをお勧めします。

信号	説明
Robot execution（ロボットの実行)	このオプションのデジタル入力 I/O シグナルは、ロボットが RAPID プログラムで項目ターゲットを実行できることを示すために使用されます。 シグナルがHighになると実行が開始され、シグナルがLowになると実行が停止します。 シグナルが低くなると、現在実行中のシーンに残っているすべてのアイテムがドロップされるため、シグナルが再び高くなると、次のシーンのアイテム ターゲットが実行されます。 また、シグナルは、1 つのシーンが終了した後に Low になり、次に再び High になって、次のシーンのアイテム ターゲットの実行を開始する必要があります。ロボットがモーションを繰り返す必要がある場合、このシグナルはストロボの信号と同じである必要があります。
Queue idle (キューのアイドル状態)	デジタル出力シグナル。 この作業エリアのキューが空の場合、このシグナルはHighになります。 最後のアイテムがキューから取得されると、シグナルが高くなります。 **注記** ロボットがモーションを繰り返す必要がある場合、この信号は**ストロボ** の信号と同じである必要があります。
Position available (使用可能なポジション)	この出力信号は、作業範囲の`Robot execution`信号が高い場合、ひとつか複数のアイテムがある場合に高くなる。`Robot Execution`信号が使われない場合`Position Available`、キューにアイテムがあるとすぐに信号が高くなる。
Position generator（ポジションジェネレータ)	画像認識やセンサを使って新しくポジションを生成するためのデジタル入力信号です。この信号は、間隔トリガコンベヤが使用されている場合は無視されます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

信号	説明
Trig（トリガ）	ビジョンを使用する場合、このデジタル出力シグナルをカメラの I/O ポートのトリガー入力に接続する必要があります。
Strobe（ストロボ）	ストロボシグナルの入力シグナル名です。 ビジョンを使用する場合、カメラの I/O ポートのストロボ出力からシグナルを生成する必要があります。 事前定義済みのポジションが使用されている場合、ストロボはトリガー出力によって直接生成されます。これは、トリガーシグナル用のシミュレートされた出力シグナルと、シミュレートされたストロボ入力シグナルへのロジック クロス接続を使用して行うのが最適です。

プロシージャ

PickMaster PowerPac リボン タブで、**レイアウト** をクリックします。

次の手順を使用して、インデックスワークエリアを追加します。

1 リボン タブで、**インデックスワークエリア** をクリックします。
インデックスワークエリアウィンドウが開きます。
2 **インデックスワークエリアウィンドウで、作業エリア名テキスト** ボックスに名前を入力するか、デフォルトの名前を使用します。
3 必要な作業範囲のタイプを選択。
4 ［OK］をクリックします。

© Copyright 2025 ABB. All rights reserved.

4.3.2.9 ポジションジェネレータ

概要

このセクションでは、作成したソリューションのポジションジェネレータを設定する方法について説明します。

ポジションジェネレータの設定

ソリューション内でポジションを生成する場所および方法を定義するには、**ポジションジェネレータを** クリックします。ステーションの校正前に、**ポジションジェネレータを** 正しく定義する必要がある。

次の表は、**ポジションジェネレータ**設定ダイアログボックスの詳細を示す。

	説明
利用可能なコンベアおよびインデックスワークエリアリスト	関連する関係を設定するために、コンベアまたはインデックスワークエリアを選択。
ソースタイプ	入力シグナル ソース タイプを選択します。 • ビジョン：ソースタイプがビジョンに設定されている場合、カメラおよびビジョンモデルを使用して物体のポジションが検索されます。ビジョンモデルはビジョンモデルの追加 ページ *293*に説明されています。  ヒント ソース タイプがビジョンに設定されている場合、使用可能なすべてのカメラと関連アイテムが使用可能なカメラに一覧表示されます。 • 事前定義：ソース タイプが事前定義に設定されている場合、ポジションソースによって生成されるポジションは静的に定義され、カメラは使用されません。 • 外部：ソースタイプが外部 に設定されている場合は、ソリューション内の外部センサーおよび外部ポジションジェネレータを使用して、アイテムのポジションを定義します。  ヒント インデックスワークエリアが使用されている場合、外部センサー機能は無効になる。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

	説明
トリガー設定	トリガー タイプを選択して、新しいアイテムのポジションをいつ生成するかを定義します。 **注記** トリガー タイプが距離 に設定されている場合、レシピ の下の操作 設定の トリガー距離 ボックスでトリガー距離を定義する必要があります。 間隔トリガは、コンベヤワークエリアでのみ使用できます。入力された値は、連続するトリガ同士の間で、コンベヤが移動する距離です. **注意** レシピで事前定義 センサーと IO センサーが選択されている場合は、オフセットを構成するアイテムの半径距離のチューニングでピックポジションを調整します。 **ヒント** インデックスワークエリアが使用されている場合、トリガー設定は使用できません。
ベースフレーム調整	選択したコンベアまたはインデックスワークエリアのベース フレームを調整します。 詳しくは「ベースフレームの調整 ページ 138」を参照してください。

プロシージャ

注記

ソース タイプまたはトリガー設定を変更すると、新しいキャリブレーションが必要になります。

次の手順を使用してポジションジェネレータを設定します。

1. PickMaster PowerPac ツリー ビューのレイアウトで、ポジションジェネレータ を右クリックします。
2. クリックしてコンベアを 1 つ選択します。
3. ソースのタイプとトリガー設定を設定します。
4. 必要に応じて、仮想ベース フレーム データを適宜設定します。
5. クリックして他のコンベアを選択し、設定します。
6. ［OK］をクリックします。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.3.2.9 ポジションジェネレータ
続き

ベースフレームの調整

デフォルトの仮想ベース フレームが実際のステーションの実際のベース フレームと一致しない場合は、ベース フレームを調整して、生産環境でのピック & プレイスする精度を確保します。

以下の表は、ベースフレーム調整設定ダイアログボックスの詳細を示す。

ベースフレーム調整	説明
コントローラ	必要なコンベアまたはインデックスワークエリアを選択して、そのベース フレームを調整します。
仮想ベース フレーム	現在の仮想ベース フレーム データを表示し、ユーザーが仮想ベース フレーム データを編集できるようにします。 ヒント バーチャルステーションの校正時に、バーチャルベースフレームのデータは自動的に更新される。また、実際のステーションからコピーしたり、手動で編集することも可能。
リアルベースフレーム	リアルコントローラから取得した現在のリアルベース フレーム データを表示します。 ヒント リアル ベース フレーム データはPickMaster PowerPac から変更できません。
 xx2200001993	リアル ベース フレーム データを仮想ベース フレーム データに同期します。
ベースフレームを表示	選択すると、ステーション ビューにベース フレームが表示されます。
適用	編集した仮想ベース フレーム データを保存し、仮想コントローラに適用します。
取得	リアルコントローラからリアルベースフレームデータを取得します。 ヒント リアルベース フレーム データは、リアルRuntimeが接続されている場合にのみ取得できます。 リアルRuntimeへの接続の詳細については、*Real Runtimeへの切り替え ページ 179* を参照してください。

プロシージャ

下記の手順に従って、仮想ベース フレームを調整 します。

1 リアルRuntimeに切り替えます。
詳しくは「*Real Runtimeへの切り替え ページ 179*」を参照してください。

次のページに続く

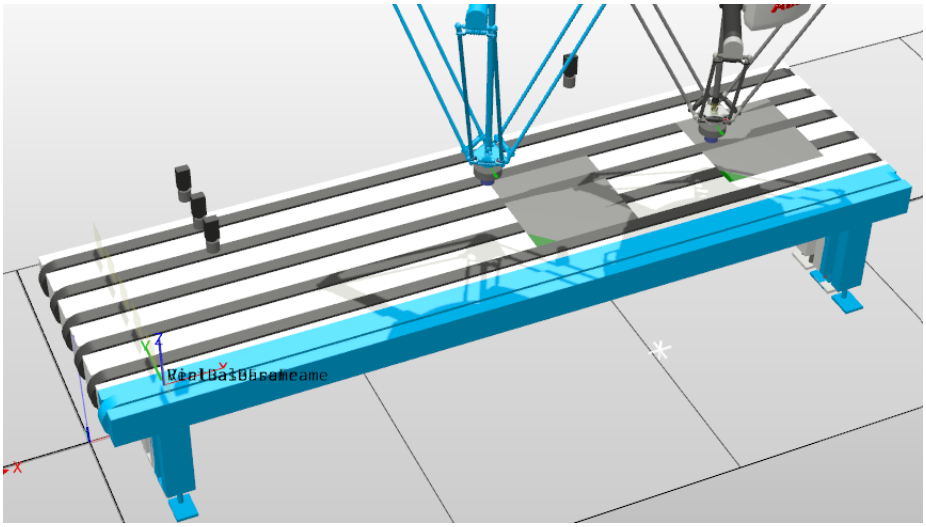
© Copyright 2025 ABB. All rights reserved.

2 PickMaster PowerPac ツリー ビューの**レイアウト**で、**ポジションジェネレータ** を右クリックします。
3 クリックして、目的のコンベアまたはインデックスワークエリアを選択します。
4 **コントローラ** ドロップダウン リストをクリックして、目的のコントローラを選択します。
5 **取得** をクリックして、リアルコントローラからリアルベース フレーム データを取得します。
6 **同期** ボタンをクリックして、リアル ベース フレームから仮想ベース フレームにデータを同期します。
7 ［はい］をクリックします。
8 ［適用］をクリックします。
9 ポップアップメッセージボックスで**はい**をクリックし、バーチャルベースフレームを保存。

> **注記**
>
> このステップでユーザーが**いいえ** をクリックすると、仮想ベース フレーム データは保存されません。

10 必要に応じて、**はい** をクリックし、ステーションビュー内のステーション コンポーネントのポジションを調整。

はい をクリックすると、仮想ベース フレームとリアルベース フレームがステーション ビューで一致します。



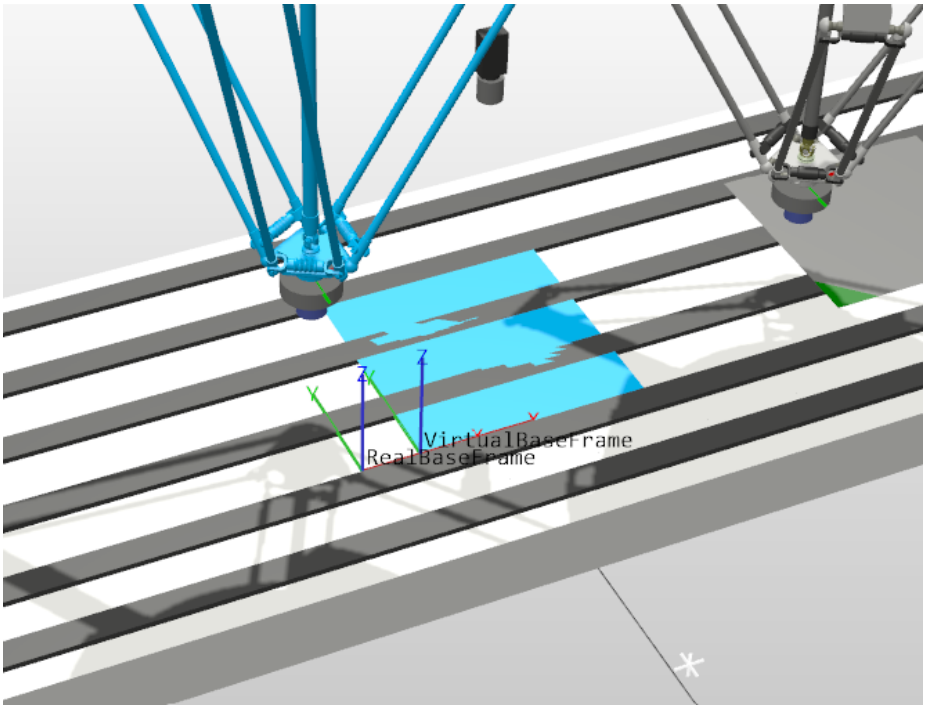
xx2200002000

ユーザーがこのステップで**いいえ** をクリックすると、ステーション コンポーネントはそれに応じて移動しません。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.3.2.9 ポジションジェネレータ
続き



xx2200001999

11 クリックして他のコンベアを選択し、設定します。
12 ［OK］をクリックします。

© Copyright 2025 ABB. All rights reserved.

4.3.2.10 キャリブレーション

概要

このセクションでは、作成したソリューションをキャリブレーションする方法について説明します。

PickMaster PowerPacでのキャリブレーションは、シミュレーションを実行するための前提条件です。 キャリブレーションは、実際のハードウェア (カメラ、コンベア、IO センサーなど) のキャリブレーションとは異なります。 このキャリブレーションを実行しても、実際のハードウェア キャリブレーションが完了したことにはなりません。

PickMaster PowerPacのキャリブレーションは、コンベア ベース フレームと仮想コントローラのロボット ベース座標系との間の相対的な関係を確立するために使用されます。

ソリューション内のリニア コンベアにカメラが使用されている場合、キャリブレーション後、コンベアのベース フレームはカメラの真下にあります (x は順方向)。 IO センサーがリニア コンベアに使用される場合、コンベアのベース フレームは IO センサーのポジションにあります。 リニア コンベアのデフォルト設定で事前定義を使用すると、コンベアのベース フレームがホットスポット 0 にオーバーラップします。 フレームの詳細については、フレーム関係 ページ *95*を参照してください。

割り出しされた作業エリアのキャリブレーションは、リニア コンベアのキャリブレーションと一致しています。

円形コンベア ベルトのキャリブレーションされたベース座標系は、コンベア ベルトの中心にあり、x 方向はカメラの真下または IO センサーに沿った方向を指します。 円形コンベアが事前定義済みのポイントを使用する場合、x 方向は事前定義済みの座標ポイント (ホットスポット) を指します。

キャリブレーション

PickMaster PowerPac リボン タブで、**レイアウト** をクリックします。

この手順を使用して、キャリブレーションします。

1 リボン タブの **キャリブレーション** をクリックします。 その後、作成されたソリューションの自動キャリブレーションが開始されます。

キャリブレーションが自動的に実行されます。

> **注記**
>
> カメラのポジションやロボットのポジションを変更するなど、ソリューション内のレイアウトが変更された場合は、キャリブレーションをやり直してください。

フレームの詳細については、フレーム関係 ページ *95* を参照してください。

© Copyright 2025 ABB. All rights reserved.

4.3.3 プロセス

4.3.3.1 Item（項目）

概要

アイテムは、ロボットによりピックアンドプレイスされるオブジェクトです。ピック / プレイスの両方に対して、1つのアイテムを使用するのが一般的ですが、作成するアイテムの数はいくつでも構いません。

アイテムのグリップのポジションは、アイテムのポジションに相対するピック / プレイスポジションを定義します。

このセクションでは、アイテムを追加する方法について説明します。

アイテムの追加

ソリューションにアイテムを追加するには、リボンの**アイテム**をクリックします。

以下の表は、**アイテム設定**ダイアログボックスの詳細を示す。

アイテムプロパティタブ

アイテムのプロパティ

	説明
名前	名前を変更します。
タイプ	アイテムの形状を変更します。 • 円柱 • ボックス • カスタマイズした：事前定義されたモデルをインポートします。  ヒント `*.rslib` と `*.rsgfx` のみを PickMaster PowerPac にインポートできます。 これら 2 つのタイプのファイルは Robotstudio からエクスポートされます。
サイズ（x、y、z）[mm]	アイテムのサイズを設定します。

Rapid特性

	説明
許可タイプ	許可アイテムタイプの値を定義します。 許可アイテム タイプの値が RAPID プログラムに送信され、アイテム ターゲットとともに提供されます。 詳細については、*GetItmTgt - 次のアイテムターゲットを取得する ページ 390* を参照してください。
拒否タイプ	拒否アイテムタイプの値を定義します。 拒否アイテム タイプの値は RAPID プログラムに送信され、アイテム ターゲットとともに提供されます。 詳細については、*GetItmTgt - 次のアイテムターゲットを取得する ページ 390* を参照してください。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

> **注記**
>
> 1 つのソリューション内の異なるアイテムの許可タイプまたは拒否タイプが同じ値に設定されている場合、ピッキング ステータスに影響します。

外観特性

	説明
テンプレート	デフォルト設定 タブ：プリセット テンプレートの 1 つを選択します。 デフォルト名テキスト ボックス：新しいテンプレートの名前を入力します。 保存アイコン：新しいテンプレートを保存します。 削除アイコン：テンプレートを削除します。 **ヒント** テンプレート テキスト ボックスに新しいテンプレート名を入力すると、名前が変更される代わりに、新しいテンプレートが作成されます。 **注記** 新しいテンプレートを作成する代わりに、デフォルト テンプレートの外観を直接変更すると、デフォルト テンプレートのデフォルト値が変更されます。 また、デフォルトのテンプレートで作成されたすべてのアイテムも変更されます。
Color	新しいアイテムの色を変更します。
テクスチャを使用	アイテムにテクスチャイメージ ファイルを使用します。
ラベルの場所	アイテムのラベルの場所を設定します。
ラベル写真	ラベル写真の画像ファイルを選択します。
輪郭を表示	輪郭を表示するかどうかを選択します。
向きマーカーを表示	向きマーカーを表示するかどうかを選択します。
ブラウズ	カスタマイズされたモデルを選択してインポートします。
オフセット量 (mm)	インポートされたカスタマイズモデルのオフセット量を設定します。
向き [度]	インポートされたカスタマイズ モデルの向きを設定します。

アイテムソースタブ

> **注記**
>
> ユーザーがアイテムのソース タイプを変更した場合、ユーザーはそれに応じて、関連するレシピ設定で選択をやり直す必要があります。

	説明
ビジョン	ソースタイプが［Vision（ビジョン）］に設定されている場合、カメラおよびビジョンモデルを使用してオブジェクトポジションが検索されます。ビジョンモデルはビジョンモデルの追加 ページ293に説明されています。 ビジョン モデルの詳細については、次のセクションを参照してください。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

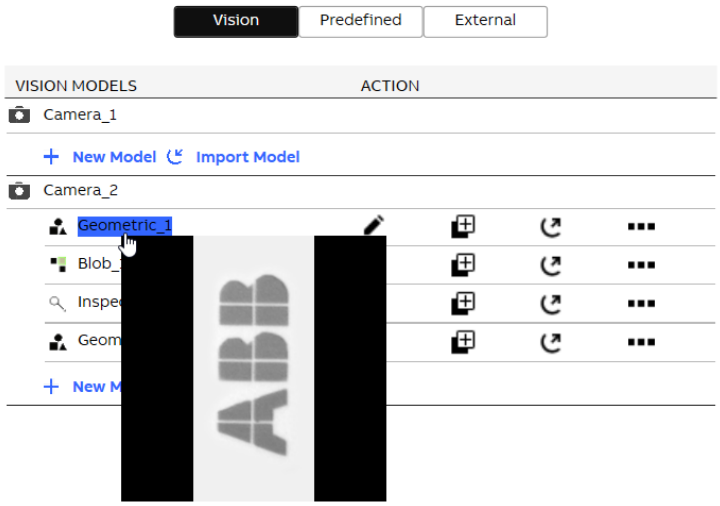
4.3.3.1 Item（項目）
続き

	説明
事前定義済み	ソースタイプが［Predefined（事前定義）］に設定されている場合には、ポジションソースで生成されたポジションは静的に定義され、カメラは使用されません。
外部の	ソースタイプが外部 に設定されている場合は、ソリューション内の外部センサーおよび外部ポジションジェネレータを使用して、アイテムのポジションを定義します。 詳しくは「外部センサー ページ *357*」を参照してください。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

ビジョン

	説明
新モデル	新バージョンモデルを追加。 • 図形：図形バージョンモデルを追加。 ジオメトリック サブ検査モデルは、*Geometric* モデルと同じ方法で設定されます。*Geometricモデルの設定 ページ296*を参照してください。さらに、見つかったアイテムの相対位置と対応するアライメント ヒットをトレーニングする必要があります。 • 輪郭：輪郭ビジョンモデルを追加。 輪郭サブ検査モデルは、Blobモデルと同じように構成されます。*Blobモデルの構成 ページ305*を参照します。必要とされるヒット数も構成する必要があります. • 検査：検査ビジョン モデルを追加。 ビジョン モデル名の上に 1 秒間マウスを移動すると、トレーニングされたモデルがプレビュー画像として表示されます。 **注記** プレビューできるのは、幾何学モデルまたは幾何学アライメントモデルを備えた検査モデルのみです。 **注記** PickMaster Powerpac 2.3.1以前のバージョンで作成されたビジョンモデルはすべて、直接プレビューすることができません。 PickMaster Powerpac 2.3.1以前のバージョンで作成されたビジョンモデルを処理する場合、編集タブを開き、**OK**ボタンをクリックしてプレビュー画像を生成します。  xx2400000635
インポートモデル	既存バージョンモデルのインポート。
編集	選択したビジョンモデルを編集。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.3.3.1 Item（項目）
続き

	説明
コピー	• コピー：選択したビジョン モデルを同じタイプのモデルにコピーします。 • 検査モデルとしてコピー：選択したビジョン モデルをコピーし、選択したビジョン モデルをアライメント モデルとして使用して検査モデルとして保存します。  注記 形状モデルの場合、検査モデルとしてコピーできるのは、アイテム高さが設定されている形状モデルのみです。 ビジョンの高さと外部の高さは、幾何学モデルでのみ使用できます。幾何学的なアライメントモデルを備えた検査モデルには使用できません。この場合、ビジョン高さまたは外部高さが設定されている形状モデルを検査モデルにコピーすることはできません。 アイテムの高さ、ビジョンの高さ、および外部の高さの詳細については、高さ設定の構成 ページ372を参照してください。
エクスポート	選択したビジョン モデルをエクスポート。
さらに	• 削除：選択したビジョンモデルを削除します。 • 名前の変更：選択したビジョンモデルの名前を変更します。

事前定義済み

	説明
ポジション （X、Y、Z） [mm]	定義済みモデルのポジションを設定。
角度 Z [度]	定義済モデルの Z 軸の角度を設定。

外部の

 ヒント

アイテム/コンテナの外部設定は、RealRuntimeが接続時にのみ実装可能。

	説明
新しいポジションジェネレータ	外部ポジションジェネレータを追加。 ユーザーが本センサーのポジションジェネレータを事前に作成していない場合、最初に新規ポジションジェネレータのボタンをクリックする必要がある。すると、 `def configurePosGen(self, posGenId)`の python インターフェースが自動的に呼び出され、それは次の操作「設定」と同一である。この操作の前提条件は、対応する外部センサーがセクション3.2に従ってすでに構成されていること、そうしない場合、メッセージボックスが以下に表示される。「現在のセンサーは未設定。ポジションジェネレータの作成前に、センサーを設定すること。」 外部ポジションジェネレーターの設定についての追加情報は、外部センサー ページ357を参照。
同期時間 [MS]	RT受信ストローブ信号の時刻は、現在のシステム時刻（StrobeTime）からデータ処理時間（iTimeSinceStrobe）を引いたものである。しかし、コントローラのトリガストローブ信号からRT受信ストローブ信号までの時間は計算不可能。そのため、同期時間の値はこの値を補正するために使用される。この値は、ハードウェア上での信号伝送と機能呼び出しに費やされる時間を補正するために、ユーザーが設定します。外部センサーが異なる場合、この値は異なって設定される場合がある。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

	説明
設定	一旦ポジションジェネレータが作成および設定されると、ユーザーは再度設定を行うために設定ボタンをクリックできる。本操作は、`def configurePosGen(self, posGenId)`の Pythonインターフェースを参照。ユーザーは、本インターフェースのポジションジェネレータの設定作業を、自身のPythonクラスで自己定義する必要がある。PMPP UI ではポジションジェネレータはひとつしか作成できないが、このインターフェースではより多くのポジション生成メソッドをユーザーが実装可能で、ひとつまたは複数のメソッドに基づいてポジションを生成できる。 センサー設定と同様に、ポジションジェネレータの設定情報は文字列に直列化される必要があり、PMPP ソリューションは本文字列を取得して保存可能。 このボタンは、ボタンの状態が有効である限り、クリックできる。現在の文字列が無効状態の場合、対応するポジションジェネレータは、設定 - 有効状態になるまで設定不可。
削除	選択したポジションジェネレータを削除。
保存する	保存 - 有効の状態では、ユーザーは「保存」ボタンをクリックし、Pythonプログラムから設定文字列を取得し、PMPP で更新可能。本ボタンはPythonインターフェース「def savePosGen(self, posGenId)」を参照し、本インターフェースは、PMTWの開発者がExternalSensorInterface.py ファイルで提供しているもので、ユーザーは本インターフェースの内容を変更する必要はない。内容には構成文字列の返しのみが含まれているため、ユーザーは、全設定情報が「configurePosGen」インターフェースの本文字列に含まれていることを確認する必要がある。 「保存」ボタンをクリックすると、すべての文字列が有効状態になる。
OK	「OK」ボタンはアイテム/コンテナ表示に対応します。本ボタンをクリックすると、すべてのデータが保存され、アイテム/コンテナビューが閉じる。外部センサーポジションジェネレーターが保存 - 有効状態の場合、ビューを閉じる前に、「savePosGen」Pythonインターフェースがはじめに呼び出される。
`Cancel`	「取消」ボタンはアイテム/コンテナビューに対応します。本ボタンがクリックされると、変更データはすべて破棄され、アイテム/コンテナビューは閉じられる。

プロシージャ

PickMaster PowerPacリボン タブで、**プロセス** をクリックします。

以下の手順に従ってアイテムを追加します。

1. リボン タブで、**アイテム** をクリックします。

   アイテム ウィンドウが開きます。
2. **RH サイズ** 部分で、アイテムのサイズを定義します。

   アイテムの高さは、ピックの高さを定義し、ビジョンモデルまたはあらかじめ定義されたポジションソースで定義されたポジションによって検出されたアイテムに常に追加されます。
3. 必要に応じて、許可または拒否するアイテムタイプのレベルを定義します。

   検査プログラムを使用すると、検出されたアイテムは許可または拒否のいずれかにマークされます。許可および拒否のアイテムタイプの値が［Item Configuration（アイテムの構成）］ダイアログボックスに表示され、［RAPID］プログラムに送られ、そこで処理されます。検査モデルの構成 ページ *312*を参照します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.3.3.1 Item（項目）
続き

4 ［OK］をクリックします。

関連情報

検査モデルの構成 ページ *312*。

© Copyright 2025 ABB. All rights reserved.

4.3.3.2 Container（コンテナ）

概要

コンテナは、使用するパターンと、パターン内の各ポジションに使用するアイテムを定義します。このように、異なるコンテナで同じパターンを使用できますが、アイテムは異なります。

このセクションでは、コンテナを追加する方法について説明します。

事前知識

コンテナを設定する前に、ソリューションで少なくとも 1 つのアイテムを定義する必要があります。

コンテナの追加

ソリューションにコンテナを追加するには、リボンの**コンテナ**をクリックします。

以下の表は、**コンテナ設定**ダイアログボックスの詳細を示す。

コンテナのプロパティタブ

コンテナのプロパティ

	説明
コンテナ名	名前を変更します。
LWH サイズ (mm)	コンテナのサイズを設定します。
タイプ	コンテナのタイプを定義します。 • **ボックス** • **カスタマイズした**：事前定義されたモデルをインポートします。  **ヒント** `*.rslib` と `*.rsgfx` のみを PickMaster PowerPac にインポートできます。 これら 2 つのタイプのファイルは Robotstudio からエクスポートされます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.3.3.2 Container（コンテナ）
続き

外観特性

	説明
テンプレート	デフォルト設定 タブ：プリセット テンプレートの 1 つを選択します。 デフォルト名テキスト ボックス：新しいテンプレートの名前を入力します。 保存アイコン：新しいテンプレートを保存します。 削除アイコン：テンプレートを削除します。  ヒント テンプレート テキスト ボックスに新しいテンプレート名を入力すると、名前が変更される代わりに、新しいテンプレートが作成されます。  注記 新しいテンプレートを作成する代わりに、デフォルト テンプレートの外観を直接変更すると、デフォルト テンプレートのデフォルト値が変更されます。 また、デフォルト テンプレートで作成されたすべてのコンテナも変更されます。
Color	コンテナの色を変更。
テクスチャを使用	コンテナにテクスチャイメージ ファイルを使用します。
ラベルの場所	コンテナのラベルのポジションを設定。
ラベル写真	ラベル写真の画像ファイルを選択します。
輪郭を表示	輪郭を表示するかどうかを選択します。
向きマーカーを表示	向きマーカーを表示するかどうかを選択します。
ブラウズ	カスタマイズされたモデルを選択してインポートします。
オフセット量 (mm)	インポートされたカスタマイズモデルのオフセット量を設定します。
向き [度]	インポートされたカスタマイズ モデルの向きを設定します。

コンテナパターンタブ

パターンは、ポジションのコレクションを定義します。 たとえば、特定の物体のポジションが事前定義されたボックスです。 使用可能なオプションを使用して、選択したレイヤーの順序を変更、削除、または再配置できます。オフセット (mm) を変更して、各レイヤーの垂直ポジションを調整できます。 ソート方法も管理できます。 ソート方法 セクションでは、コンテナ パターン内のアイテムがロボットによって処理される順序を定義します。

	説明
レイヤーの追加	新しいレイヤーを追加します。 レイヤーの追加の詳細については、次のセクションを参照してください。
レイヤーの編集	選択したレイヤーを編集します。
コピー	選択したレイヤーをコピーします。
レイヤーの削除	選択したレイヤーを削除します。
アップ	選択したレイヤーを上の階層に移動します。
ダウン	選択したレイヤーを下の階層に移動します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

	説明
すべて削除	既存のレイヤーをすべて削除します。
総重量	すべてのアイテムの総重量を表示。
全高	すべてのアイテムの合計の高さを表示。
総数	すべてのアイテムの合計数を表示。

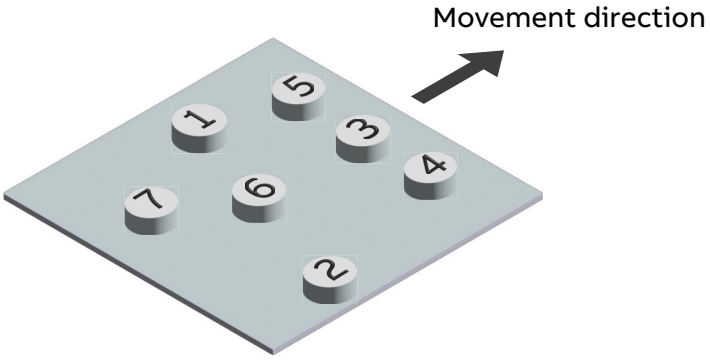
レイヤーの追加

	説明
利用可能なアイテム	作成済みの利用可能なアイテムをひとつ選択。 追加アイコン：選択したアイテムをレイヤーに追加します。 削除アイコン：選択したアイテムを削除します。 すべて選択アイコン：レイヤー内のすべてのアイテムを選択します。
整列スタイル	レイヤーに複数のアイテムがある場合は、整列スタイルを定義します。 左揃えアイコン：このレイヤー内のすべてのアイテムを左から揃えます。 中央揃えアイコン：このレイヤー内のすべてのアイテムを中央から揃えます。 右揃えアイコン：このレイヤー内のすべてのアイテムを右から揃えます。 上揃えアイコン：このレイヤー内のすべてのアイテムを上から揃えます。 上下中央揃えアイコン：このレイヤー内のすべてのアイテムを上下中央から揃えます。 下揃えアイコン：このレイヤー内のすべてのアイテムを下から揃えます。
配布スタイル	レイヤーに複数のアイテムがある場合の配信スタイルを定義します。 横方向アイコン：すべてのアイテムを横方向に配置します。 縦方向アイコン：すべてのアイテムを縦方向に配置します。
Else 機能	回転アイコン：選択したアイテムを回転します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.3.3.2 Container（コンテナ）
続き

	説明
ソート方法	シグナルを設定します。 カスタマイズ設定オプションを使用してシグナルを管理します。  xx2400000741 オプションなし：レイヤー内のアイテムは、各レイヤーのレイアウトで定義されているのと同じ順序でアクセスされますが、次のアイテムに到達できない場合は、その次のアイテムが使用されます。図のワークエリアの並び順は、1 -> 2 -> 3 -> 4 -> 5 -> 6 ->7となります。 移動方向オプション：アイテムは、各レイヤーの移動方向に沿って、つまりコンベアに沿って移動する順序でアクセスされます。図のワークエリアの並び順は、3 -> 4 -> 5 -> 1 -> 6 -> 2 ->7となります。 厳密なオプション：作業エリアは、各レイヤーのレイアウトで定義されているのと同じ順序で使用されます。ロボットがレイヤー内の次のアイテムポジションにアクセスできない場合、そのロボットはコンテナ パターン内のそれ以上のアイテムポジションを使用しません。たとえば、このオプションが選択されているロボットが作業エリア5にアクセスできない場合、作業エリア5, 6, 7は選択されず、図のワークエリアの並べ替え順序は1 -> 2 -> 3 -> 4になります。
順序	レイヤーの順序を定義します。
ポジション X Y Z [mm]	レイヤー内のアイテムのポジションを定義します。
角度 X Y Z [度]	レイヤー内のアイテムの角度を定義します。
アイテム名の表示	アイテムの名前を表示します。
アイテム順序の表示	アイテムの追加順序を表示します。

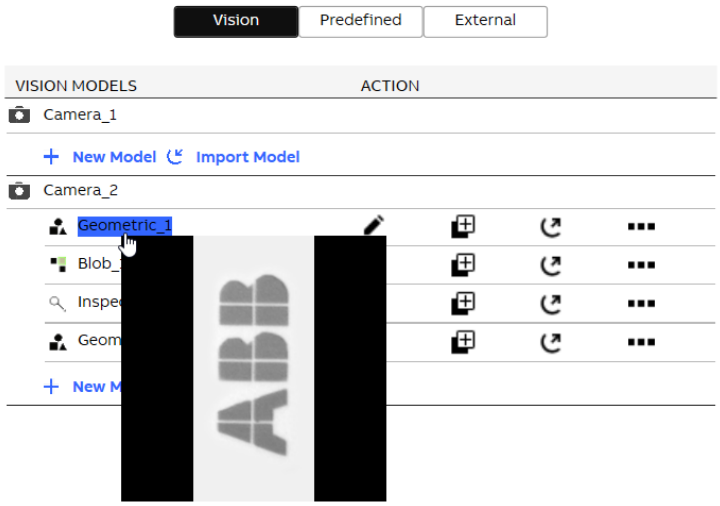
コンテナソースタブ

	説明
ビジョン	ソースタイプが［Vision（ビジョン）］に設定されている場合、カメラおよびビジョンモデルを使用してオブジェクトポジションが検索されます。ビジョンモデルはビジョンモデルの追加 ページ*293*に説明されています。 ビジョン モデルの詳細については、次のセクションを参照してください。
事前定義済み	ソースタイプが［Predefined（事前定義）］に設定されている場合には、ポジションソースで生成されたポジションは静的に定義され、カメラは使用されません。
外部の	ソースタイプが外部 に設定されている場合は、ソリューション内の外部センサーおよび外部ポジションジェネレータを使用して、コンテナのポジションを定義します。 詳しくは「外部センサー ページ *357*」を参照してください。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

ビジョン

	説明
新モデル	新バージョンモデルを追加。 • 図形：図形バージョンモデルを追加。 ジオメトリック サブ検査モデルは、*Geometric* モデルと同じ方法で設定されます。*Geometricモデルの設定 ページ296*を参照してください。さらに、見つかったアイテムの相対位置と対応するアライメント ヒットをトレーニングする必要があります。 • 輪郭：輪郭ビジョンモデルを追加。 輪郭サブ検査モデルは、Blobモデルと同じように構成されます。*Blobモデルの構成 ページ305*を参照します。必要とされるヒット数も構成する必要があります. • 検査：検査ビジョン モデルを追加。 ビジョン モデル名の上に 1 秒間マウスを移動すると、トレーニングされたモデルがプレビュー画像として表示されます。 **注記** プレビューできるのは、幾何学モデルまたは幾何学アライメントモデルを備えた検査モデルのみです。 **注記** PickMaster Powerpac 2.3.1以前のバージョンで作成されたビジョンモデルはすべて、直接プレビューすることができません。 PickMaster Powerpac 2.3.1以前のバージョンで作成されたビジョンモデルを処理する場合、編集タブを開き、**OK**ボタンをクリックしてプレビュー画像を生成します。  xx2400000635
インポートモデル	既存バージョンモデルのインポート。
編集	選択したビジョンモデルを編集。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.3.3.2 Container（コンテナ）
続き

	説明
コピー	• コピー：選択したビジョン モデルを同じタイプのモデルにコピーします。 • 検査モデルとしてコピー：選択したビジョン モデルをコピーし、選択したビジョン モデルをアライメント モデルとして使用して検査モデルとして保存します。 注記 形状モデルの場合、検査モデルとしてコピーできるのは、アイテム高さが設定されている形状モデルのみです。 ビジョンの高さと外部の高さは、幾何学モデルでのみ使用できます。幾何学的なアライメントモデルを備えた検査モデルには使用できません。この場合、ビジョン高さまたは外部高さが設定されている形状モデルを検査モデルにコピーすることはできません。 アイテムの高さ、ビジョンの高さ、および外部の高さの詳細については、高さ設定の構成 ページ372を参照してください。
エクスポート	選択したビジョン モデルをエクスポート。
さらに	• 削除：選択したビジョンモデルを削除します。 • 名前の変更：選択したビジョンモデルの名前を変更します。

事前定義済み

	説明
ポジション （X、Y、Z） [mm]	定義済みモデルのポジションを設定。
角度 Z [度]	定義済モデルの Z 軸の角度を設定。

外部の

ヒント

アイテム/コンテナの外部設定は、RealRuntimeが接続時にのみ実装可能。

	説明
新しいポジションジェネレータ	外部ポジションジェネレータを追加。 ユーザーが本センサーのポジションジェネレータを事前に作成していない場合、最初に新規ポジションジェネレータのボタンをクリックする必要がある。すると、 `def configurePosGen(self, posGenId)`の python インターフェースが自動的に呼び出され、それは次の操作「設定」と同一である。この操作の前提条件は、対応する外部センサーがセクション3.2に従ってすでに構成されていること、そうしない場合、メッセージボックスが以下に表示される。「現在のセンサーは未設定。ポジションジェネレータの作成前に、センサーを設定すること。」 外部ポジションジェネレーターの設定についての追加情報は、外部センサー ページ357を参照。
同期時間 [MS]	RT受信ストローブ信号の時刻は、現在のシステム時刻（StrobeTime）からデータ処理時間（iTimeSinceStrobe）を引いたものである。しかし、コントローラのトリガストローブ信号からRT受信ストローブ信号までの時間は計算不可能。そのため、同期時間の値はこの値を補正するために使用される。この値は、ハードウェア上での信号伝送と機能呼び出しに費やされる時間を補正するために、ユーザーが設定します。外部センサーが異なる場合、この値は異なって設定される場合がある。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

	説明
設定	一旦ポジションジェネレータが作成および設定されると、ユーザーは再度設定を行うために設定ボタンをクリックできる。本操作は、`def configurePosGen(self, posGenId)`の Pythonインターフェースを参照。ユーザーは、本インターフェースのポジションジェネレータの設定作業を、自身のPythonクラスで自己定義する必要がある。PMPP UI ではポジションジェネレータはひとつしか作成できないが、このインターフェースではより多くのポジション生成メソッドをユーザーが実装可能で、ひとつまたは複数のメソッドに基づいてポジションを生成できる。 センサー設定と同様に、ポジションジェネレータの設定情報は文字列に直列化される必要があり、PMPP ソリューションは本文字列を取得して保存可能。 このボタンは、ボタンの状態が有効である限り、クリックできる。現在の文字列が無効状態の場合、対応するポジションジェネレータは、設定 - 有効状態になるまで設定不可。
削除	選択したポジションジェネレータを削除。
保存する	保存 - 有効の状態では、ユーザーは「保存」ボタンをクリックし、Pythonプログラムから設定文字列を取得し、PMPP で更新可能。本ボタンはPythonインターフェース「def savePosGen(self, posGenId)」を参照し、本インターフェースは、PMTWの開発者がExternalSensorInterface.py ファイルで提供しているもので、ユーザーは本インターフェースの内容を変更する必要はない。内容には構成文字列の返しのみが含まれているため、ユーザーは、全設定情報が「configurePosGen」インターフェースの本文字列に含まれていることを確認する必要がある。 「保存」ボタンをクリックすると、すべての文字列が有効状態になる。
OK	「OK」ボタンはアイテム/コンテナ表示に対応します。本ボタンをクリックすると、すべてのデータが保存され、アイテム/コンテナビューが閉じる。外部センサーポジションジェネレーターが保存 - 有効状態の場合、ビューを閉じる前に、「savePosGen」Pythonインターフェースがはじめに呼び出される。
`Cancel`	「取消」ボタンはアイテム/コンテナビューに対応します。本ボタンがクリックされると、変更データはすべて破棄され、アイテム/コンテナビューは閉じられる。

プロシージャ

PickMaster PowerPacリボン タブで、**プロセス** をクリックします。

次の手順でコンテナを追加します。

1. リボン タブで、**コンテナ** をクリックします。
   コンテナウィンドウが開きます。
2. **コンテナのプロパティ** タブで、要件を使用してコンテナを定義します。
3. **コンテナ パターン** タブで、要件を使用してコンテナ パターンを定義します。
4. **コンテナ パターン** タブで、**新規レイヤー** をクリックしてコンテナ内のレイヤーを定義します。
5. 必要に応じて、レイヤー上のアイテムのレイアウトを調整します。
   A レイヤー上のすべてのアイテムを選択します。
   B 「Ctrl」を押しながらベースアイテムを選択します。
   C **左** をクリックして、すべてのアイテムをベース アイテムに従って左端に揃えます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.3.3.2 Container（コンテナ）
続き

右 をクリックして、ベース アイテムに従ってすべてのアイテムを右端に揃えます。

上下中央 をクリックして、すべてのアイテムをベース アイテムに従って縦方向に上下中央線に揃えます。

中央 をクリックして、すべてのアイテムをベース アイテムに従って横方向に中央線に揃えます。

上 をクリックして、ベース アイテムに従ってすべてのアイテムを上端に揃えます。

下 をクリックして、ベース アイテムに従ってすべてのアイテムを下端に揃えます。

D 横方向 をクリックして、すべてのアイテムを横方向に接するように設定します。

縦方向 をクリックして、すべてのアイテムを縦方向に接するように設定します。

6 保存をクリックします。

レイヤーレイアウトが保存されます。

7 ［OK］をクリックします。

コンテナが保存され、ウィンドウが閉じられます。

© Copyright 2025 ABB. All rights reserved.

4.3.3.3 フロー

概要

フローは、シミュレーションでアイテムとコンテナがどのように生成されるかを定義するために使用されます。フローを使用すると、実際にランダムかつ不規則に流入する材料の流れをシミュレートできます。フローはコンベア上のホットスポットに接続されます。フローをアタッチすると、ホットスポットは、フロー設定に従ってシミュレーションにアイテムとコンテナが表示されるソースになります。フローには次の 2 種類があります。

- レイアウト：レイアウト フローは、ホットスポットで定期的に再生成される事前定義済みのレイアウトです。 レイアウトには、アイテムまたはコンテナの場所とそれらの入手可能性に関して、ランダムな変動が生じる場合があります。 レイアウトは、さまざまなアイテムまたはコンテナ パターンで構成されている場合があります。
- 記録：記録されたフローは、シミュレーションまたは生産からのセンサーの記録です。 記録は、ある期間中に検出されたすべてのアイテムとコンテナの情報を含む xml ファイルとしてPickMaster PowerPacからエクスポートされます。 ファイルがインポートされると、検出されたアイテムが設定済みアイテムおよびコンテナ パターンにマップされます。

> **注記**
>
> 作業範囲のソースタイプがビジョンに設定されている場合のみ、フローを使用できる。

フローの追加

ソリューションにフローを追加するには、リボンのフローをクリックします。

以下の表は、フロー設定ダイアログボックスの詳細を示す。

	説明
レイアウト	フローのレイアウトを定義します。
記録済み	アイテムやコンテナの記録されたポジションに合わせてフローを作ります。

レイアウト

	説明
名前	フローの名前を変更します。
フロータイプ	フローのタイプをライナーまたは円形から選択します。
LWサイズ[mm]	レイアウトのサイズを編集します。 レイアウトの編集：選択したフローを編集します。 レイアウトの編集の詳細については、次のセクションを参照してください。
安定性[%]	100% に設定すると、レイアウト内のすべてのアイテムが、すべてのトリガーで損失なく生成されます。 低い値は、レイアウト内のアイテムが生成される確率を定義します。 たとえば、安定性を 50% に設定すると、各アイテムが生成されない確率が半分になります。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.3.3.3 フロー
続き

	説明
位置安定性[%]	100% に設定すると、生成されたアイテムは常に正しいポジションになります。 値が小さいほど、アイテムが正しいポジションに配置される確率が定義されます。
X ポジション偏差最小/最大 [mm]	X ポジションの正しい値からの最小および最大偏差を定義します。
Y ポジション偏差最小/最大 [mm]	Y ポジションの正しい値からの最小および最大偏差を定義します。
方向安定性[%]	100% に設定すると、生成されたアイテムは常に正しい向きになります。 値が小さいほど、アイテムが正しい向きになる確率を定義します。
Z 回転偏差最小/最大[度]	正しい値からの Z 回転の最小および最大偏差を定義します。
拒否率[%]	アイテムがカメラによって拒否される確率を定義します。 0% に設定すると、レイアウトのアイテム設定「拒否」によって、アイテムが拒否されるかどうかが決定されます。

レイアウトの編集

注記

環状フローの場合、青いセクター ビューは、ホットスポット フレーム内にアイテムとコンテナを生成するエリアです。

直感的な視覚効果を得るには、このコンベアのホットスポットの x と y の値を 0 と 0 に設定します。これにより、ホットスポットの原点が円形コンベアの中心と同じポジションに設定されます。 そして、青いセクタービューは、円形コンベア上のクォータービューでもあります。

	説明
利用可能な物体	本システム用に作成したアイテムまたはコンテナを 1 つ選択します。 追加アイコン：選択したアイテムまたはコンテナをレイヤーに追加します。 削除アイコン：選択したアイテムを削除します。 すべて選択アイコン：レイヤー内のすべてのアイテムを選択します。
整列スタイル	レイヤーに複数のアイテムがある場合は、整列スタイルを定義します。 左揃えアイコン：このレイヤー内のすべてのアイテムを左から揃えます。 中央揃えアイコン：このレイヤー内のすべてのアイテムを中央から揃えます。 右揃えアイコン：このレイヤー内のすべてのアイテムを右から揃えます。 上揃えアイコン：このレイヤー内のすべてのアイテムを上から揃えます。 上下中央揃えアイコン：このレイヤー内のすべてのアイテムを上下中央から揃えます。 下揃えアイコン：このレイヤー内のすべてのアイテムを下から揃えます。
配布スタイル	レイヤーに複数のアイテムがある場合の配信スタイルを定義します。 横方向アイコン：すべてのアイテムを横方向に配置します。 縦方向アイコン：すべてのアイテムを縦方向に配置します。
Else 機能	回転アイコン：選択したアイテムを回転します。
順序	レイヤーの順序を定義します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

	説明
ポジション[mm]	レイヤー内のアイテムのポジションを定義します。
角度 [度]	レイヤー内のアイテムの角度を定義します。
アイテム名の表示	アイテムの名前を表示します。
アイテム順序の表示	アイテムの追加順序を表示します。

プロシージャ

PickMaster PowerPacリボン タブで、**プロセス** をクリックします。

次の手順で、長方形フローを追加します。

1 リボン タブで、**フロー** をクリックします。

フロー ウィンドウが開きます。

2 **フロー タイプ** でフローのタイプを選択します。

3 **レイアウトの編集** アイコンをクリックして、ダイアログを開きます。

4 **レイアウトの編集** ダイアログの**追加** アイコンをクリックして、アイテムを追加します。

A レイヤー上のすべてのアイテムを選択します。

B 「Ctrl」を押しながらベースアイテムを選択します。

C **左** をクリックして、すべてのアイテムをベース アイテムに従って左端に揃えます。

右 をクリックして、ベース アイテムに従ってすべてのアイテムを右端に揃えます。

上下中央 をクリックして、すべてのアイテムをベース アイテムに従って縦方向に上下中央線に揃えます。

中央 をクリックして、すべてのアイテムをベース アイテムに従って横方向に中央線に揃えます。

上 をクリックして、ベース アイテムに従ってすべてのアイテムを上端に揃えます。

下 をクリックして、ベース アイテムに従ってすべてのアイテムを下端に揃えます。

D **横方向** をクリックして、すべてのアイテムを横方向に接するように設定します。

縦方向 をクリックして、すべてのアイテムを縦方向に接するように設定します。

5 **OK**をクリックして設定を適用します。

6 **OK** をクリックして **フロー** ダイアログを閉じます。

© Copyright 2025 ABB. All rights reserved.

4.3.3.4 レシピ

概要

このセクションでは、レシピを追加する方法について説明します。

1 つのソリューションで、複数のレシピを作成できます。 このソリューションのすべての要素 (ロボット、センサーなど) は、制限なく任意のレシピに追加できます。

レシピの追加

ソリューションにレシピを追加するには、リボンのレシピをクリックします。

以下の表は、レシピ設定ダイアログボックスの詳細を示す。

プロパティ

	説明
利用可能なデバイス	ロボットやコンベアなど、利用可能なデバイスを定義します。 同じソリューション内のすべてのロボットとコンベアがすべてのレシピにリストされますが、異なるレシピでは異なる属性設定を持つことができます。 たとえば、同じロボットの速度がレシピによって異なる場合があります。 利用可能なデバイスの詳細については、次のセクションを参照してください。
利用可能な作業エリア	利用可能な作業エリアを定義します。 同じソリューション内のすべての作業エリアがすべてのレシピにリストされますが、異なるレシピでは異なる属性設定を持つことができます。 利用可能な作業エリアの詳細については、次のセクションを参照してください。

利用可能なデバイス

	説明
ロボットの設定	**注記** 本システムに複数のロボットがある場合、すべてのロボットが定義された名前でここにリストされます。 速度：ロボットの速度を変更します。 緊急：ロボットの緊急プログラムをインポート/エクスポート/編集。 ラピッド エディタ：ラピッドを開くエディタを指定します。 **注記** デフォルトのRAPIDモジュールをIRB 360用に作成。 さまざまなロボット タイプ カテゴリおよびダブル ピッキング用の代替 RAPID テンプレート モジュールは、インストール フォルダ `C:\Program Files (x86)\ABB\PickMaster Twin 3\Samples\RAPIDs` からインポートできます。
コンベア設定	速度：コンベアの速度を変更します。 加速度：コンベアの加速度を変更します。 減速度：コンベアの減速度を変更します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

	説明
カメラ パラメータ設定	すべてのカメラを同期: カメラ設定からレシピカメラパラメータ設定にすべてのカメラのパラメータを同期します。 同期: カメラ設定からレシピカメラパラメータ設定にカメラのパラメータを同期します。 リセット: パラメータを最後に保存したデータにリセットします。 露出: カメラの露出。 明るさ: カメラの明るさ。 コントラスト: カメラのコントラスト。 ライブ/停止: 現在の設定でライブビデオ画像ダイアログを表示/停止します。 適用: パラメータを適用し、ライブビデオ画像ダイアログで有効にします。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

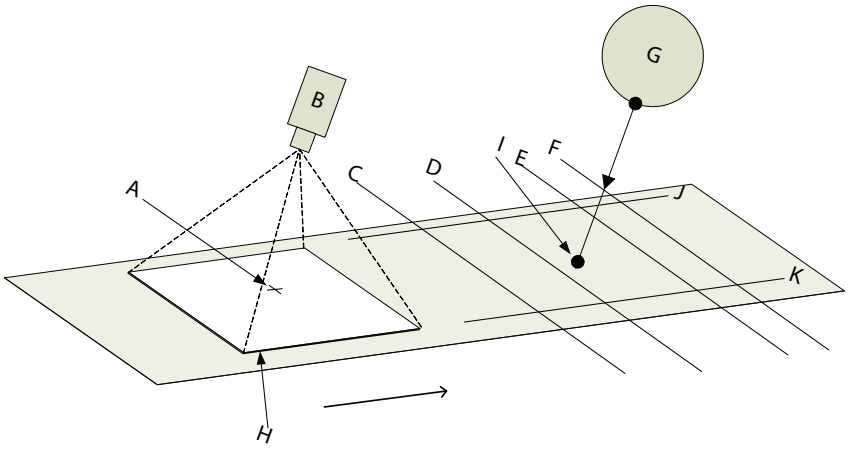
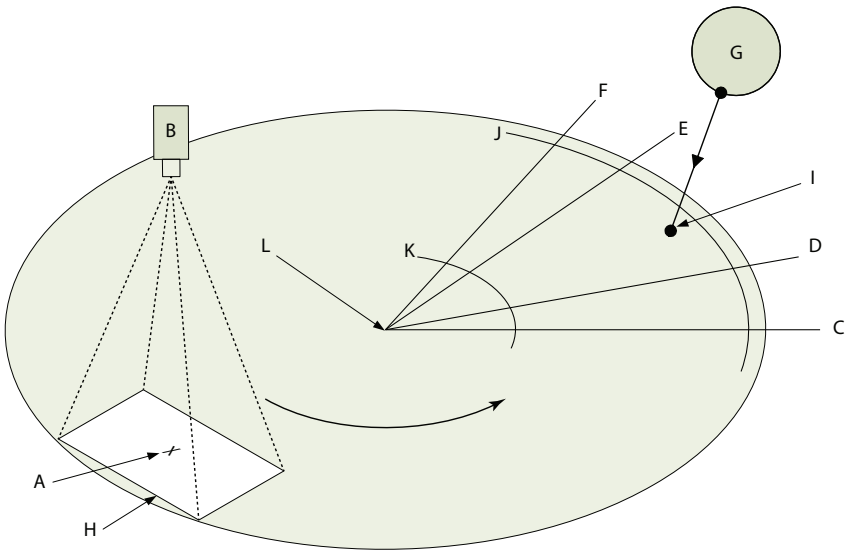
4.3.3.4 レシピ
続き

利用可能な作業エリア

	説明	
ピック(拾う)の設定	ピック/プレースの高さ	ロボットがワークエリアターゲットに近づく場所からの、ツールを基準とした負の Z 方向の距離。
	ピック/プレースの時間[秒]	ロボットがピック/プレースポジションにある時間。ピック/プレース時間中にコンベアが移動している場合、ロボットはコンベアに沿ってトラッキングし、移動するコンベア上の相対ポジションを維持します。
	バキューム アクティベーション[秒]	バキューム I/O を設定する必要がある、接近ポジションのコーナー パスの中央までの時間 (秒単位)。 負の値を入力すると、バキュームI/Oはコーナー パスの中央以降の時間に設定されます。この値は、タイプがPickのワークエリアに対してのみ有効です。 **注記** シミュレーションでは、吸着のアクティベーションはアイテムのピッキングには影響を及ぼしません。ピックルーチンでは、アイテムは例えばSimAttachイベントを使用してピッキングツールに装着されます。
	バキューム リバージョン[s]	ブロー I/O を設定する必要がある、定ポジションでの半定ポジション時間前の秒単位の時間。 負の値を入力すると、ブロー I/O は、プレースポジションでのプレース時間の半分後の時間を設定します。この値は、タイプがプレースのワークエリアに対してのみ有効です。 **注記** シミュレーションでは、吸着の開放はアイテムのプレイシングには影響を及ぼしません。プレイスルーチンでは、アイテムは例えばSimDetachイベントを使用してピッキングツールから外されます。
	バキューム オフ[秒]	所定ポジションでの半プレース時間の経過後、ブロー I/O をリセットする必要がある時間 (秒単位)。負の値を入力すると、ブロー I/O は、プレースポジションでのプレース時間の半分前の時間にリセットされます。この値は、タイプがプレースのワークエリアに対してのみ有効です。 **注記** シミュレーションでは、吸着オフはアイテムのプレイシングには影響を及ぼしません。プレイスルーチンでは、アイテムは例えばSimDetachイベントを使用してピッキングツールから外されます。
	ロード時間[秒]	インデックス付きワークエリア内のオブジェクトの生成間隔時間。この値は、インデックス付きワークエリアに対してのみ有効です。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

	説明
エリア設定	Start X と呼ばれる作業エリアで開始エントリを定義した後、Start X の垂直方向に Start Y と呼ばれる同じ開始エントリを定義できます。  xx1800001747  xx2400000631

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.3.3.4 レシピ
続き

	説明

A	リニアコンベアのカメラとベースフレームの原点 円形コンベア用カメラ原点
B	カメラ
C	エンター
D	Start（起動）
E	Stop（停止）
F	終了
G	ロボット
H	イメージ フレーム
I	ロボットの中心
J	Y 最大値/半径 最大値
K	Y 最小値/半径 最小値
L	円形コンベアのベースフレーム原点

注記

エンター、終了、起動、停止 の基準原点は **I** (ロボットの中心) です。 **Y Max** と **Y Max** の基準ベースは、コンベア ベース フレームです。

入力[mm][i]/[度][ii]	**エンター** は、ロボットが作業エリアでアイテムターゲットへの作業実行を開始する限界の地点です。距離は、ロボットの中心からミリ単位で計算されます。限界が、フィーダの移動方向に相対してロボット中心よりも後ろの場合は、正数の範囲になります。エンターの限界がロボットの届く範囲であることを確認してください。
開始[mm][i]/[度][ii]	**［Start（起動）］** の場合、コンベヤ上で作業実行される次のアイテムがこの限界を超えていると、コンベヤが起動します。距離は、ロボットの中心からミリ単位で計算されます。限界が、フィーダーの移動方向に相対してロボット中心よりも後ろの場合は、正数の範囲になります。
停止[mm][i]/[度][ii]	**［Stop（停止）］** の場合、コンベヤ上のアイテムがこの限界に達すると、コンベヤは停止します。距離は、ロボットの中心からミリ単位で計算されます。限界が、フィーダーの移動方向に相対してロボット中心よりも後ろの場合は、正数の範囲になります。
終了[mm][i]/[度][ii]	**終了** は、ロボットが作業エリアでアイテムターゲットを失ったものと見なす限界地点です。距離は、ロボットの中心からミリ単位で計算されます。限界が、フィーダの移動方向に相対してロボット中心よりも後ろの場合は、正数の範囲になります。トラッキングされたアイテムがこの限界を超えると、このアイテムはドロップされます。この限界は、ロボットの最大リーチ範囲内に十分入るように選択する必要があります。ロボットは、このポジションにリーチできなくなる前に、ロボットの作業エリアのあらゆるポジションからこのポジションにリーチできなくてはなりませ ん。
Y 最大[mm][i]/半径 最大[mm][ii]	**Y 最大[mm]/半径 最大[mm]**は、ロボットがアイテム ターゲットをエンド Y のワークエリアで紛失したとみなす限界です。距離はロボットの中心からミリメートル単位で計算されます。コンベアの移動垂直方向に対して、制限がロボットの中心を超えている場合、範囲は正になります。 ロボットが**Y 最大/半径 最大**に到達できることを確認してください。アイテムのポジションの Y 座標値が**Y 最大/半径 最大**より大きい場合、ロボットはアイテムを掴みません。したがって、トラッキングされたアイテムがこの制限を超えると、ドロップされます。この制限は、ロボットの最大到達範囲内で十分に選択する必要があります。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

	説明	
	Y 最小[[mm]*i*/ 半径 最小[mm]*ii*	Y 最小[mm]/半径 最小[mm]は、ロボットが開始 Y の作業エリアでアイテム ターゲットの実行を開始する限界です。距離はロボットの中心からミリメートル単位で計算されます。コンベアの移動垂直方向に対して、制限がロボットの中心を超えている場合、範囲は正になります。
	開始/停止を使用する	作業エリアで開始および停止の制限を監視する必要がある場合は、このチェックボックスをオンにします。 注記 起動と停止の値は、エンターと終了の制限の境界内にある必要があります。 エンターの値は、起動の値より小さくなければなりません。 停止の値は、終了の値より小さくなければなりません。 つまり、シミュレーション中に複数のエラーが発生します。 注記 起動/停止の使用 チェックボックスが選択されている場合、停止 と 終了 の間の距離は、コンテナのサイズ (x 方向) よりも大きくする必要があります。 これは、コンベヤ起動/停止信号により処理されます。「作業範囲 ページ 130」を参照してください。
	生産で開始	生産の開始時にワークエリアがコンベアと連動し、生産の停止時に停止するようにする場合は、このチェックボックスをオンにします。
	Y 最大値/Y 最小値を使用*i*/ 半径 最大値/半径 最小値を使用*ii*	作業エリアで上限と下限を監視する必要がある場合は、このチェックボックスをオンにします。
記録設定	シミュレーションと生産におけるアイテムとコンテナのポジションを記録します。 注記 記録シーンが選択された場合、どの作業範囲でも保存されると、以下のメッセージがポップアップ表示される。 `Scenes recording is activated for: {0}` その後、シミュレーションや生産が開始されると、自動的に録画が開始される。 同じ操作のすべてのワークエリアのシーンが 1 つの `.xml` ファイルに保存されます。	

操作

操作には、ピック(拾う)作業とプレース(配置)作業があります。

	説明
主な設定	操作名、フロー、ソース タイプなど、操作のいくつかの基本設定を定義します。 メイン設定の詳細については、主な設定 ページ 166を参照してください。
フィルタ設定	操作のフィルタ設定を定義します。 フィルタ設定の詳細については、フィルタ設定 ページ 168を参照してください。
UserScripts	操作のUserScripts機能を定義する場合に選択します。 ユーザー スクリプトの詳細については、*UserScripts ページ 337* を参照してください。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.3.3.4 レシピ
続き

	説明
配信設定	操作の配信設定を定義します。 納品設定の詳細については、配信設定 ページ 168を参照してください。

主な設定

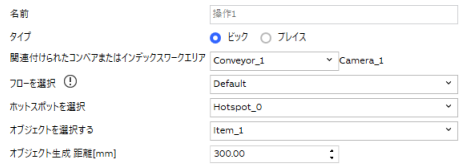
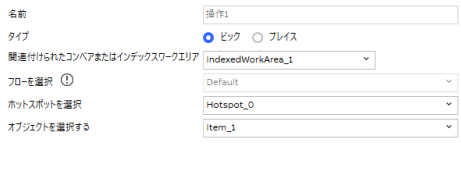
	説明
操作名	操作の名前を変更します。
操作タイプ	操作のタイプを設定します。
関連付けられたコンベアまたはインデックス付き WA	関連するコンベアまたはインデックス付き WA を選択します。
フローの選択	定義したフローを選択します。 フローを追加する方法の詳細については、フロー ページ 157 を参照してください。 **ヒント** コンベアで外部センサーが使用されている場合、フロー機能は無効になる。
ホットスポットの選択	定義したホットスポットを選択します。
物体の選択	定義した使用可能なアイテムまたはコンテナを選択します。
物体生成距離[mm]/[度]	オブジェクト生成距離値を定義します。 **ヒント** インデックスワークエリアを使用する場合、**物体生成距離[mm]/[度]**は使用できません。 詳細については、次の表を参照してください。
トリガー距離[mm]/[度]	**トリガー設定** が **距離** として設定されている場合のトリガー距離の値を定義します。 **注記** **ソースタイプ** が **事前定義済み** に設定され、**トリガー設定** が **距離** に設定されている場合、トリガー距離の値は **物体生成距離[mm]/[度]** の値から取得されます。 詳細については、次の表を参照してください。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

オブジェクト生成およびトリガー距離の使用条件の違い

物体生成距離[mm]/[度] と トリガー距離[mm]/[度] はさまざまな条件で有効であるため、すべての条件とそれぞれのオプションを以下に示します。

	ソースタイプ	トリガー設定	物体生成距離[mm]/[度]	トリガー距離[mm]/[度]	主な設定 ビュー
コンベヤ	ビジョン/外部センサー	Distance	利用可	利用可	 xx2200002001
コンベヤ	ビジョン/外部センサー	入出力	利用可	使用できません	 xx2200002002
コンベヤ	事前定義	入出力			
コンベヤ	事前定義	Distance	利用可	無効	 xx2200002003
Indexed work area（インデックスワークエリア）	ビジョン/外部センサー	Distance	使用できません		 xx2200002004
Indexed work area（インデックスワークエリア）	ビジョン/外部センサー	Distance			
Indexed work area（インデックスワークエリア）	事前定義	入出力			
Indexed work area（インデックスワークエリア）	事前定義	入出力			

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.3.3.4 レシピ
続き

フィルタ設定

	説明
ポジションフィルタ距離	ポジションフィルタは、カメラまたは外部センサーによって検出されたアイテムポジション間の最小許容距離を決めます。 たとえば、同じ物体を識別するために 2 つ以上のモデルが使用されている場合、モデルごとに1つのヒットがほぼ同じポジションにある可能性があります。 同じアイテムの 2 つのポジションが、定義されたアイテムの最小距離よりも x 方向または y 方向に近い場合、ソート値が最も高いポジションのみがロボット コントローラに送信されます。 ソート値は、ビジョン モデルごとに設定できます。ビジョンモデルの追加 ページ *293* を参照してください。 **「同じレベルのみ」** をチェックした場合、フィルタリングは検査レベルが同じアイテムポジション間でのみ行われます。 **注記** 事前定義済みのポジションが使用されている間は、ポジションフィルタは使用されません。
オーバーラップ フィルタ距離	リニアコンベアの場合、アイテムは、オーバーラップにより、2つの連続するフレームで認識される場合があります。モデルは、これらのフレーム間のピック / 配置ポジションでいくつかのバリエーションを持つことができます。2つの連続するフレームに収まり、これらのフレーム間のピック/配置ポジションがオーバーラップフィルタの間隔を超えて変化しないアイテムは、1つのアイテムとして見なされます。最初に特定されたヒットがロボットに送信され、その後のヒットはフィルタで排除されます。
オーバーラップ フィルタ角度	リニアコンベアの場合、アイテムは、オーバーラップにより、2つの連続するフレームで認識される場合があります。モデルは、これらのフレーム間のピック / 配置ポジションでいくつかのバリエーションを持つことができます。2つの連続するフレームに収まり、これらのフレーム間のピック配置ポジションがオーバーラップフィルタ角度を超えて変化しないアイテムは、1つのアイテムとして見なされます。最初に特定されたヒットがロボットに送信され、その後のヒットはフィルタで排除されます。 **注記** 円形コンベアの場合、**オーバーラップ フィルタ距離**e と **オーバーラップ フィルタ角度** の両方が有効です。 どちらが機能するかは、どちらのフィルタリング条件がより厳しいかによって異なります。

高度な機能 - UserScripts

UserScriptsは、プログラミングユーザーのための高度な機能です。詳細については、*UserScripts ページ 337*を参照してください。

配信設定

デフォルトでは、すべてのポジションが同じワークエリアに送られます。負荷を複数のロボット間で分散するか、すべてのポジションが確実にアクセスされるように、アイテムポジションを複数のワークエリアに配置することが可能です。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

特定のアイテムタイプに対するポジションはすべて、1 つのアイテムディストリビュータによってロボットに分配されます。アイテムディストリビュータには 4 つのタイプがあります。

- ワークエリア：アイテムポジションが、1 つのコンベヤまたはインデックスワークエリアにより処理されます。
- バイパス：アイテムポジションが破棄される、すなわちどのワークエリアでも処理されません。アイテムタイプに対してディストリビュータが選択されない場合、それはバイパスであるとみなされます。
- LB グループ:アイテムポジションが、負荷分散グループに含まれるワークエリアでは処理されません。負荷分散グループは、ワークエリア、バイパス、および ATC グループディストリビュータの集まりです。アイテムポジションは、同一ワークエリアに隣り合う 2 つのポジションが送られることがないように、複数ワークエリアの間で最適な方法で分配されます。
- ATC グループ:ポジションが、適応型タスク完了（ATC）グループに含まれるワークエリアでは処理されません。ATC グループは、同一アイテムポジションを受け取る、順序付けされたワークエリアの集まりです。最初のロボットは、可能な限り多くのポジションにアクセスします。ATC グループ内のその他のロボットは、初回でアクセスしそこなったポジションにアクセスします。グループ内の最後のワークエリアが、開始 / 停止を行うコンベヤワークエリアである場合、すべてのポジションが必ずアクセスされることが保証されます。

負荷分散または ATC のいずれかを使用するためには、ワークエリアがポジションソース（例としてカメラまたはセンサー）の後に発生する順番で順序付けされる必要があります。

ポジションソースをトリガーする作業エリアは、自動的に設定されます。生産の開始時、初め作動準備完了になるロボットの作業エリアが、トリガー作業エリアとして 設定されます。トリガー作業エリアのロボットが停止した場合、作動中の別のロボットの作業エリアがポジションソー スをトリガーする作業エリアになります。

アイテムの分配ツリーコントロールには、ポジションが生成されるアイテムが表示されます。許可されたアイテムと拒否されたアイテムは異なる方法で分配できます。

配信

	説明
アイテム配分	この操作で利用可能な全アイテムに対して、配分戦略を受入または拒否に設定します。 **注記** 利用可能なすべてのアイテムについて、**アイテム分配の 許可** または **拒否**の中に少なくとも 1 グループの有効な分配設定があることを確認してください。 そうでなければ、このレシピを選択してシミュレーションや生産の実施時にエラーがポップアップ表示される。 `{0} lacked valid distribution. Please check settings in Recipe -> Operation.`
利用可能なディストリビューター	この操作で利用可能なディストリビューターを表示します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.3.3.4 レシピ
続き

負荷バランス

負荷分散グループにより分配されたアイテムポジションは、そのグループ内のディストリビュータの間で分散されます。負荷分散グループに含まれるアイテムのディストリビュータの数は、いくつでも構いません。また、1 つのディストリビュータが数回現れる場合もあります。1つのディストリビュータが追加される回数と、ディストリビュータの合計数の比は、特定のディストリビュータにより送られるアイテムポジションの比を定義します。同一ワークエリアに隣り合うポジションが送られることがないように、アイテムポジションはグループ内のディストリビュータへ最適な方法で分配されます。

適応型タスク完了が選択されている場合、定義されたATCグループが、使用可能なディストリビュータに一覧されるようになります。またATCグループは、負荷分散グループにも追加されます。ただし、タスクを完了するには、負荷分散グループに含まれるグループが ATC グループのみである必要があります。

	説明
負荷バランスグループ	作成した負荷バランスグループを表示します。
利用可能なディストリビューター	この操作で利用可能なディストリビューターを表示します。
新しい LB グループ	負荷バランスグループを作成します。
グループの削除	負荷バランスグループを削除します。
再バランス戦略チェックボックス	現在のレシピの戦略を設定します。 詳しくは「ロボットが停止した場合の再分散方法 ページ *170*」を参照してください。

ロボットが停止した場合の再分散方法

ロボットが停止した場合にアイテムポジションを再分散する方法は 3 つです。アイテムポジションは、作動中のロボットに自動的に送ることができます。ただし、作動していないロボットに引き続きアイテムポジションを送る方が便利な場合があります。

一時停止した(例えばモータがオフ状態)ロボットコントローラに送られたアイテムポジションは、対象のロボットを通過するまでは失われません。作動を再開したロボットは、直ちにピックを開始できます。

停止されたロボットは、再起動されるまでアイテムポジションを一切受け取ることができません。そのロボットに分配済みのアイテムはすべて失われます。再起動されたロボットは、新しいポジションが到達するまで待たなければなりません。

再分散の戦略：:

- 再バランス戦略チェックボックスが無効: アイテムの位置は常に、配布ツリーで定義されているとおりに配布されます。ロボットがダウンすると、一部のアイテムの位置が失われます。
- 実行中のロボットと一時停止中のロボット間の再バランス (デフォルト設定): アイテムの位置は、実行中または一時停止中のロボットがあるワークエリアにのみ送信されます。
- 実行中のロボット間の再調整: アイテムの位置は、実行中のロボットがあるワークエリアにのみ送信されます。

異なるロボット状態の詳細は、*Robot states（ロボットの状態） ページ335*を参照してください。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

再分散方法の選択は、負荷分散を生産量低減の最小化などに用いる際に重要です。

ATC

適応型タスク完了では、アイテムポジションがATCグループ内の全てのロボットによりアクセスされることが保証されます。ATCグループには順序付けされたワークエリアが含まれ、また1つのワークエリアはグループ内で1度しか存在することができません。ATCグループに分配されたアイテムポジションはいずれも、グループ内のすべてのワークエリアに送られ、初回のワークエリアによりアクセスされなかったポジションは、他のいずれかのワークエリアによりアクセスされます。最後のワークエリアが開始 / 停止を行うコンベヤにある場合、すべてのアイテムポジションがATCグループ内のいずれかのロボットにより必ずアクセスされることが保証されます。

	説明
適応型タスク完了グループ	作成された適応型タスク完了グループ:を表示します。
利用可能なディストリビューター	この操作で利用可能なディストリビューターを表示します。
新しいATCグループ	適応型タスク完了グループ:を作成します。
グループの削除	適応型タスク完了グループ:を削除します。

プロシージャ

PickMaster PowerPacリボン タブで、**プロセス** をクリックします。

次の手順でレシピを追加します。

1 リボン タブで、**レシピ** をクリックします。

レシピウィンドウが開きます。

2 **操作の追加** をクリックして、新しい操作を追加します。

3 **操作 1** をクリックして、操作の設定ウィンドウを開きます。

4 操作タイプを **ピック** または **配置** として選択します。

5 必要に応じて、**フローの選択** で該当するフローをクリックして選択します。

6 **使用可能な物体** でアイテムをクリックして選択します。

7 **利用可能な作業エリア** で作業エリアをクリックして選択します。

8 **トリガー/フィルタ設定** タブで、要件に従ってトリガーまたはフィルタ設定を定義します。

9 必要に応じて、要件に応じて**ユーザー スクリプト** をクリックして選択し、設定します。

10 **配信** タブで、**ディストリビューターを使用可能なディストリビューター** リストから **配信 スト** にドラッグします。

各アイテムタイプに対してディストリビュータは1つだけです。ディストリビュータがないアイテムタイプはバイパスと見なされます。

11 負荷分散を使用する場合には、［**負荷分散**］タブで、分布を［**使用可能なディストリビュータ**］リストから［**負荷分散グループ**］にドラッグします。

新しい負荷分散グループを作成するには、<［**Available distributors（使用可能なディストリビュータ）**］**リスト内の**> ［<New LbGroup>（新規Lbグループ）］**をダブルクリック**> します。

再分散の戦略を選択します.

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.3.3.4 レシピ
続き

12 ATCタブで［Adaptive Task Completion（適応型タスク完了）を使用するには、 ワークエリアを［Available work areas（使用可能なワークエリア）］リストから［Adaptive Task Completion group（適応型タスク完了グループ）］にドラッグします。
13 ［OK］をクリックします。
ウィンドウは閉じています。

1 つのロボットから下流のロボットへの再分配

すでに配信されたアイテムポジションは、ロボットの作業エリアに入った際に変更することが可能です。そのロボットを制御するRapidプログラムが、現在のフロー条件に基づいてアイテムポジションのスキップとタイプの変更を決定します。その結果、選択されたアイテムタイプの設定済み配信計画に従って、PickMaster PowerPac がそのアイテムポジションを下流のロボットに再配信します。

© Copyright 2025 ABB. All rights reserved.

4.3.4 操作

4.3.4.1 シミュレーション

概要

このセクションでは、作成したソリューションを使用してシミュレーションを実行する方法について説明します。

コントロール

シミュレーション生産のすべての操作はステーション ビューに反映され、すべてのデータはソリューションから取得されます。

ソリューションのコントロールダイアログボックスを開くには、ツリービューからレシピをひとつ選択し、リボンのコントロールをクリックします。

以下の表は コントロールダイアログボックスの詳細を示す。

	説明
レシピ	現在のレシピのステータスを管理し、生産データの概要を把握します。 Recipeの詳細については、次のセクションを参照してください。
チューニング	アイテム、作業エリア、ロボットのパラメータを調整します。 チューニングの詳細については、次のセクションを参照してください。
フロー制御	コンベアの速度を調整します。 フロー制御の詳細については、次のセクションを参照してください。

レシピ

	説明
レシピステータス	生産状況を管理します。
ピッキング状況	ピッキング状況の概要を要約または詳細に表示します。

チューニング

時として、ピックアンドプレイスの精密なポジションは、期待された場所とは多少異なることがあります。これは、カメラまたはワークエリアのキャリブレーションの小さな誤差が原因である可能性があります。プロジェクトの実行中に、ポジションを調節することが可能です。これは調整と呼ばれます。

ヒント

アイテム チューニングの場合、調整値は新しく生成されたアイテム ターゲットにのみ影響します。 調整値は、キュー内で認識されたアイテム ターゲットは効果がありません。

ワークエリアとロボットのチューニングの場合、チューニング値はすぐに有効になります。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.3.4.1 シミュレーション
続き

アイテムのチューニング

	説明
オフセットX	X方向のピッキングとプレースの動作時のグリッパーの場所を設定します。
オフセットY	Y方向のピッキングとプレースの動作時のグリッパーの場所を設定します。
オフセットZ	Z方向のピッキングとプレースの動作時のグリッパーの場所を設定します。
回転RX	X方向のピッキングとプレースの動作時のグリッパーの角度を設定します。 注記 角度が物理的限界から外れることはない。そうでなければ、ロボットは正常に動作しない。 たとえば、IRB 360 ロボットのグリッパーを X または Y 方向に回転しようとすると、エラーが発生します。エラーが発生した後にシミュレーションをやり直してください。
回転RY	Y方向のピッキングとプレースの動作時のグリッパーの角度を設定します。 注記 角度が物理的限界から外れることはない。そうでなければ、ロボットは正常に動作しない。 たとえば、IRB 360 ロボットのグリッパーを X または Y 方向に回転しようとすると、エラーが発生します。エラーが発生した後にシミュレーションをやり直してください。
回転RZ	Z方向のピッキングとプレースの動作時のグリッパーの角度を設定します。 注記 角度が物理的限界から外れることはない。そうでなければ、ロボットは正常に動作しない。

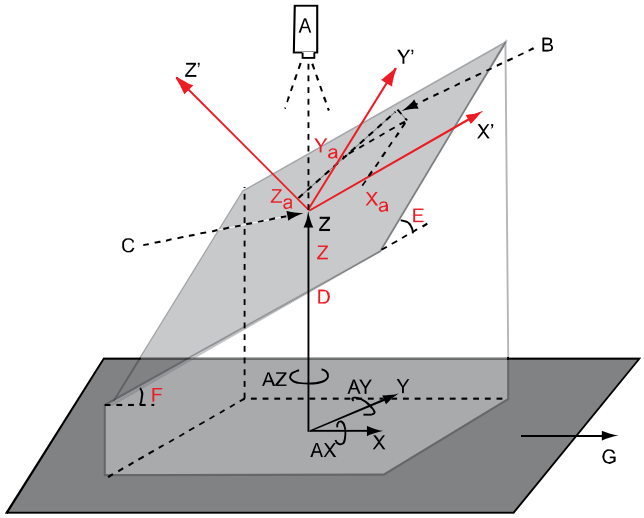
グリップのポジションの構成

下記の手順に従って、アイテムのグリップのポジションを構成してください。

1 **アイテム**として**タイプ**を選択し、必要なアイテムを選択します。
2 X´、Y´、およびZ´座標で指定されたアイテムのグリップポジション（ミリメートル）を定義します。ポジションは、ティーチモデル（ビジョンモデルグリップポイント）の原点に相対します。次の図を参照してください。

次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx0900000522

A	カメラ
B	調整済みグリップポイント
C	ビジョンモデルポイント
D	アイテムの高さ
E	アングルX
F	アングルY
G	コンベヤの方向

3 アイテム上のグリップ方向のオイラー方向を度数で定義します。

4 軸ロボットは Z 軸を中心にのみ回転できるため、回転RZのみを使用できます。

6軸ロボットは、オイラー方向回転RX、回転RY、およびアイテムの高さを定義することにより、3D アイテムをピック/プレースできます。グリップの向きは、教示モデルの原点 (ビジョン モデルのグリップ ポイント) に対する向きを持ちます。アイテムの高さは、アイテム設定ダイアログで、ベース フレームからアイテムの原点 (ビジョン モデルのグリップ ポイント) までの距離として指定する必要があります。

コンベヤのベースフレームのキャリブレーションを行う場合、アイテムグリップポイント（プレイス / ピック）に相対する向きが正確になるように、正しいキャリブレーションツールを定義することが重要です。また、アイテムのグリップポイント、つまりビジョンモデルグリップポイントと同じ高さでカメラのキャリブレーションを実行することも重要です。

作業エリアのチューニング

注記

チューニング作業エリアのパラメータは、レシピのパラメータと同期されます。1 つの場所を変更すると、別の場所のパラメータも変更されます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.3.4.1 シミュレーション
続き

	説明
オフセットX[mm]	シミュレーションまたは生産の実行時に、X方向に沿って作業エリアのポジションを調整します。 X 方向に沿って作業エリアのポジションを調整することは、X 方向に沿ってコンベア ベース フレームをオフセットすることと同じです。
オフセットY[mm]	シミュレーションまたは生産の実行時に、Y方向に沿って作業エリアのポジションを調整します。 Y 方向に沿って作業エリアのポジションを調整することは、Y 方向に沿ってコンベア ベース フレームをオフセットすることと同じです。
オフセットZ[mm]	シミュレーションまたは生産を実行するときに、Z 方向に沿って作業エリアのポジションを調整します。 Z 方向に沿って作業エリアのポジションを調整することは、Z 方向に沿ってコンベア ベース フレームをオフセットすることと同じです。
入力[mm][i]/[度][ii]	**［Enter（エンター）］** は、ロボットがワークエリアでアイテムターゲットへの作業実行を開始する限界の地点です。距離は、ロボットの中心からミリ単位で計算されます。限界が、フィーダーの移動方向に相対してロボット中心よりも後ろの場合は、正数の範囲になります。エンターの限界がロボットの届く範囲であることを確認してください。 詳細は、利用可能な作業エリア ページ *162*を参照してください。
終了[mm][i]/[度][ii]	**［Exit（終了）］** は、ロボットがワークエリアでアイテムターゲットを失ったものと見なす限界地点です。距離は、ロボットの中心からミリ単位で計算されます。限界が、フィーダーの移動方向に相対してロボット中心よりも後ろの場合は、正数の範囲になります。トラッキングされたアイテムがこの限界を超えると、このアイテムはスルーされます。この限界は、ロボットの最大リーチ範囲内に十分入るように選択する必要があります。ロボットは、このポジションにリーチできなくなる前に、ロボットのワークエリアのあらゆるポジションからこのポジションにリーチできなくてはなりませ ん。 詳細は、利用可能な作業エリア ページ *162*を参照してください。
高さ[mm]	**高さ**は、ロボットがターゲットに近づくポジションから、ツールを基準にして負の Z 方向の距離です。
フォロータイム[秒]/ドエルタイム[秒]	**フォロータイム/ドエルタイム**は、ロボットがピック/プレースポジションにある時間です。ピック/プレース時間中にコンベアが移動している場合、ロボットはコンベアに沿ってトラッキングし、移動するコンベア上の相対ポジションを維持します。
バキューム アクティベーション[秒]	**バキューム アクティベーション**は、バキューム I/O を設定する必要がある、接近ポジションのコーナー パスの中央までの時間 (秒単位) です。負の値を入力すると、バキュームI/Oはコーナー パスの中央以降の時間に設定されます。この値は、不要**ピック**のワークエリアに対してのみ有効です。 **注記** シミュレーションでは、吸着のアクティベーションはアイテムのピッキングには影響を及ぼしません。ピックルーチンでは、アイテムは例えば`SimAttach`イベントを使用してピッキングツールに装着されます。
バキューム リバージョン[s]	**バキューム リバージョン**は、ブロー I/O を設定する必要がある、配置ポジションでの配置時間の半分前の時間 (秒単位) です。 負の値を入力すると、ブロー I/O は、プレースポジションでのプレース時間の半分後の時間を設定します。この値は、不要**プレース**のワークエリアに対してのみ有効です。 **注記** シミュレーションでは、吸着の開放はアイテムのプレイシングには影響を及ぼしません。プレイスルーチンでは、アイテムは例えば`SimDetach`イベントを使用してピッキングツールから外されます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

	説明
バキューム オフ [秒]	バキューム オフは、ブロー I/O をリセットする必要がある、配置ポジションでの半分の配置時間の後の秒単位の時間です。負の値を入力すると、ブロー I/O は、プレースポジションでのプレース時間の半分前の時間にリセットされます。この値は、不要プレースのワークエリアに対してのみ有効です。 **注記** シミュレーションでは、吸着オフはアイテムのプレイシングには影響を及ぼしません。プレイスルーチンでは、アイテムは例えばSimDetachイベントを使用してピッキングツールから外されます。
Y 最大値[i]/半径 最大値[ii]	**注記** この機能を有効にするには、レシピ設定ページでこの機能の 起動/停止を使用 チェックボックスを選択する必要があります。 **注記** 調整ウィンドウのY 最大値/半径 最大値機能にはわずかな遅延があります。この値に更新がある場合は、結果が表示されるまでしばらく待つ必要があります。 Y 最大値/半径 最大値は、ロボットがアイテム ターゲットをエンド Y のワークエリアで紛失したとみなす限界です。距離はロボットの中心からミリメートル単位で計算されます。コンベアの移動垂直方向に対して、制限がロボットの中心を超えている場合、範囲は正になります。 ロボットがY 最大値/半径 最大値に到達できることを確認してください。アイテムのポジションの Y 座標値がY 最大値/半径 最大値より大きい場合、ロボットはアイテムを掴みません。したがって、トラッキングされたアイテムがこの制限を超えると、ドロップされます。この制限は、ロボットの最大到達範囲内で十分に選択する必要があります。 詳細は、利用可能な作業エリア ページ 162を参照してください。
Y 最小値[i]/半径 最小値[ii]	**注記** この機能を有効にするには、レシピ設定ページでこの機能の 起動/停止を使用 チェックボックスを選択する必要があります。 **注記** 調整ウィンドウのY 最小値/半径 最小値機能にはわずかな遅延があります。この値に更新がある場合は、結果が表示されるまでしばらく待つ必要があります。 Y 最小値/半径 最小値は、ロボットが開始 Y のワークエリアでアイテム ターゲットの実行を開始する限界です。距離はロボットの中心からミリメートル単位で計算されます。コンベアの移動垂直方向に対して、制限がロボットの中心を超えている場合、範囲は正になります。 詳細は、利用可能な作業エリア ページ 162を参照してください。
ベースフレームを表示する	ステーション ビューでコンベア ベースフレームを表示する場合は、このチェック ボックスをオンにします。

i コンベアがリニアコンベアの場合のみ使用可能です。
ii コンベアが円形コンベアの場合のみ使用可能です。

ロボットのチューニング

ロボットの設定は、**ロボットのチューニング** ウィンドウを使用して、生産の実行中にチューニングできます。

制限事項

ロボットのチューニング、アイテムのチューニング、作業エリアのチューニングを含むすべてのチューニングは、シミュレーションまたは生産の実行中にのみ有効です。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.3.4.1 シミュレーション
続き

フロー制御

	説明
コンベア	速度[mm/秒 または rad/秒] コンベアの速度を調整します。
インデックスワークエリア	ロード時間[秒]：インデックス付きワークエリア内のオブジェクトの生成時間間隔。この値は、インデックス付きワークエリアに対してのみ有効です。

シミュレーション

注記

シミュレーションの前に、仮想コントローラが他のソリューションで使用されている場合は、ソリューションをキャリブレーションすることをお勧めします。

異なるソリューションが同じ仮想コントローラを使用している場合、1 つのソリューションのコントローラを変更すると、他のソリューションに影響します。これにより、他のソリューションの予期しない誤解を招く動作が発生します。

次の手順を使用してシミュレーションを実行します。

1. PickMaster PowerPacリボン タブで、操作 をクリックします。
2. クリックして、ツリー ビュー ブラウザから 1 つのレシピを選択します。
3. リボン タブの起動 をクリックします。 次に、作成したソリューションのシミュレーションを開始します。

   シミュレーションは自動的に実行されます。
4. リボン タブの停止 をクリックします。それで、シミュレーションを停止します。

© Copyright 2025 ABB. All rights reserved.

4.4 Real Runtime（RRT）の設定

4.4.1 Real Runtimeへの切り替え

PickMaster RuntimeでのローカルIPアドレスの設定

> **注記**
>
> Runtimeで設定されるネットワーク インターフェイスは、WAN インターフェイスを使用してコントローラに接続されているローカル コンピュータの IP アドレスである必要があります。

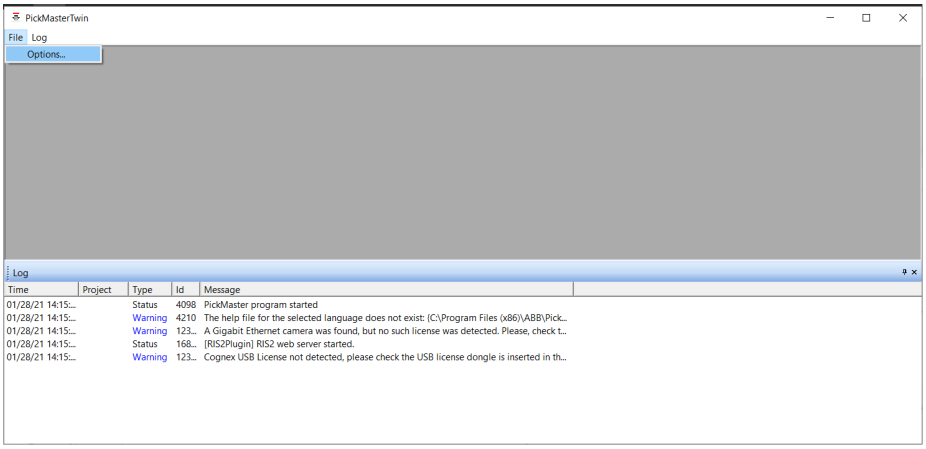
次の場合には、ローカル IP アドレスをPickMaster Runtime (RRT) で設定する必要があります。

- PickMaster Runtime時刻同期サービスの IP は以前に設定されていません。
- 実際のコントローラの接続に現在使用されているネットワーク インターフェイスが変更されました。

> **注記**
>
> Runtimeで設定されるネットワーク インターフェイスは、WAN インターフェイスを使用してコントローラに接続されているローカル コンピュータの IP アドレスである必要があります。

次の手順を使用して、PickMaster Runtime (RRT) でローカル IP アドレスを設定します。

1. Runtimeを開始します。
2. **ファイル - オプション** をクリックして、ポップアップ ダイアログを開きます。



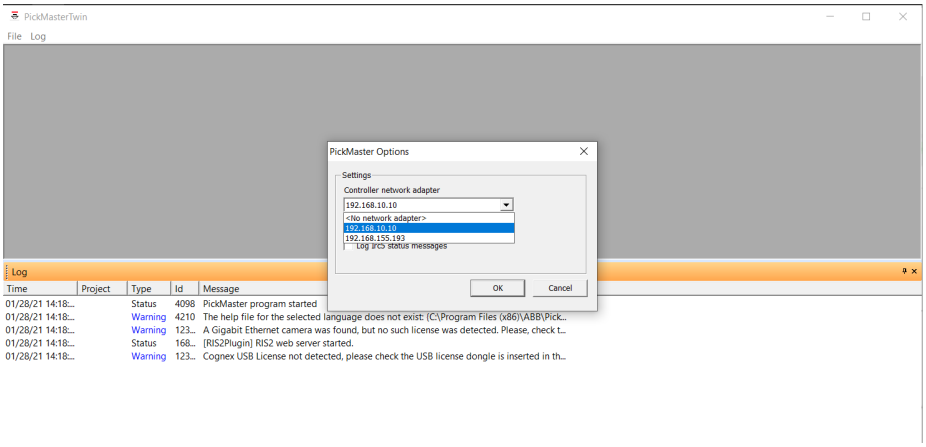
xx2100000346

3. リスト ボックスで対応する IP アドレスを選択し、**OK** をクリックします。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.1 Real Runtimeへの切り替え
続き



xx2100000347

切り替えRuntime

注記

推奨に従って、PickMaster Twin ClientおよびPickMaster Twin Hostを別の PC にインストールすると、 2 種のReal Runtimeが利用できるが、コントローラまたはカメラに接続されている方だけを使用すること。

Host PC および クライアント PC にあるReal Runtimeは同じだが、Hostのものが生産に対応します。ロボットコントローラやカメラもこれに接続します。

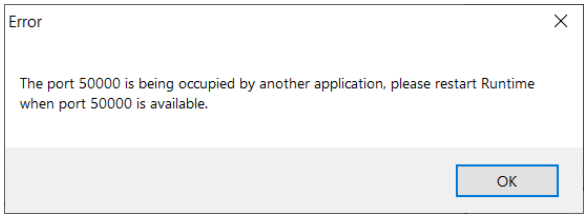
次のページに続く

© Copyright 2025 ABB. All rights reserved.

ヒント

PickMaster® Runtime (VRT および RRT) は、50000 ポートを使用するように定義されています。 50000 ポートが他のプログラムによって占有されている場合、この警告が表示され、Runtimeに接続できなくなります。

50000 ポートを解放し、PickMaster® Runtimeを再起動します。



xx2100000868

この手順を使用して、50000 ポートを解放します。

1. コマンド `netstat -aon|findstr "50000"` を CMD ウィンドウに入力します。
2. ポート 50000 を占有するプロセスがウィンドウに一覧表示されます。 プロセスの PID コードを取得します。
3. タスク マネージャでこの PID に対応するプロセスを見つけて閉じます (このコンピュータでこのプロセスを閉じることが許可されていることを確認してください)。
4. PickMaster® Runtime を再起動して接続します。

Runtime を右クリックして、シミュレーションモードでバーチャル Runtime（VRT）、あるいはエミュレーションモードでコンピHostュータ上で実際のロボットを操作するReal Runtime（RRT）への接続を設定します。

ヒント

RRTへの接続前に、PickMaster RuntimeをHostコンピュータ上で起動。

RRT に接続を選択すると、**サインインウィンドウ**が表示される。

以下の表は、**RRT** への接続ダイアログボックスの詳細を示す。

	説明
IPアドレス	Runtime コンピュータの IP アドレスを見つけます。 **ヒント** PickMaster Runtimeがインストールされているコンピュータの IPv4 アドレスを確認します。
資格情報	
ユーザー名	デフォルトのユーザー名はadmin です。 変更はできません。
パスワード	Runtimeでアカウントのパスワードを入力します。

役割ごとにデフォルトのユーザー、パスワードが作成されています。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.1 Real Runtimeへの切り替え
続き

管理者ユーザー名：admin 、パスワード：password

> **注記**
>
> ソリューションが PickMaster Operator で使用される場合、PickMaster PowerPac で同じ設定を持つ実際のコントローラに接続されている必要があります。

プロシージャ

Runtimeに接続します。

1. ツリー ビューの**レイアウト**で**Runtime**を右クリックし、**ローカル RRT の開始**を選択します。
2. ツリー ビュー **Layout**で**Runtime**を右クリックし**RRT に接続** を選択します。
   ConnectToRRT ウィンドウが開きます。
3. **サインイン** ダイアログで、正しい情報を入力します。
4. ［**OK**］をクリックします。

> **ヒント**
>
> 切り替えに失敗すると、ステータス ウィンドウが表示される。
> メッセージ：Failed to connect to RRT, please make sure the RRT starts or login information is correct.

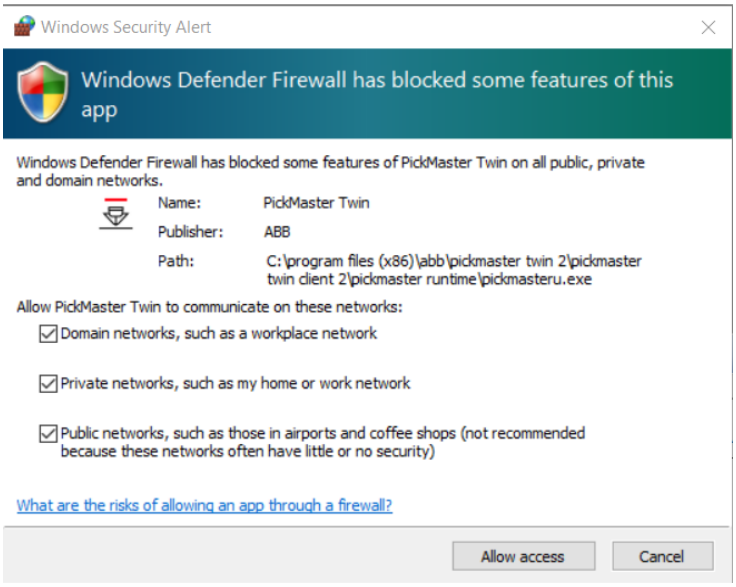
次のページに続く

© Copyright 2025 ABB. All rights reserved.

注記

PickMaster PowerPacとリアルRuntimeの間の接続を構築する際に問題が発生した場合は、次の考えられる理由を確認してください。

1 管理者ではないホスト アカウントの使用、
2 ファイアウォールのブロック、
3 VPN 干渉、
4 Host IP アドレスが正しくないか、クライアント ポートと同じ IP セグメントにない。
5 PickMaster PowerPacがすべてのネットワークで通信できるわけではありません。



xx2100001954

リアルコントローラを選択します

注記

生産を実行する必要があるコントローラに、少なくとも 1 つの実際のコントローラが選択されていることを確認してください。

そうしないと、このレシピを選択して生産する場合に、エラーがポップアップ表示される。

```
{0} lacked real controller setting. Please connect to a real controller first.
```

次の手順を使用して、リアルコントローラを選択します。

1 ツリー ビューの**レイアウト**で**コントローラ**を右クリックし、**コントローラの編集** を選択します。

コントローラの編集 ダイアログが開きます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.1 Real Runtimeへの切り替え
続き

2 リアルコントローラの選択 アイコンをクリックして、リアルコントローラの選択 ダイアログを開きます。

注記

PickMaster PowerPacでリアルOmniCore コントローラを選択する前に、ユーザーはファイアウォール設定を変更する必要があります。

WAN ポートの場合、設定/通信/ファイアウォール マネージャ で、次の機能を有効にする必要があります。

"RobICI" -EnableOnPublicNet

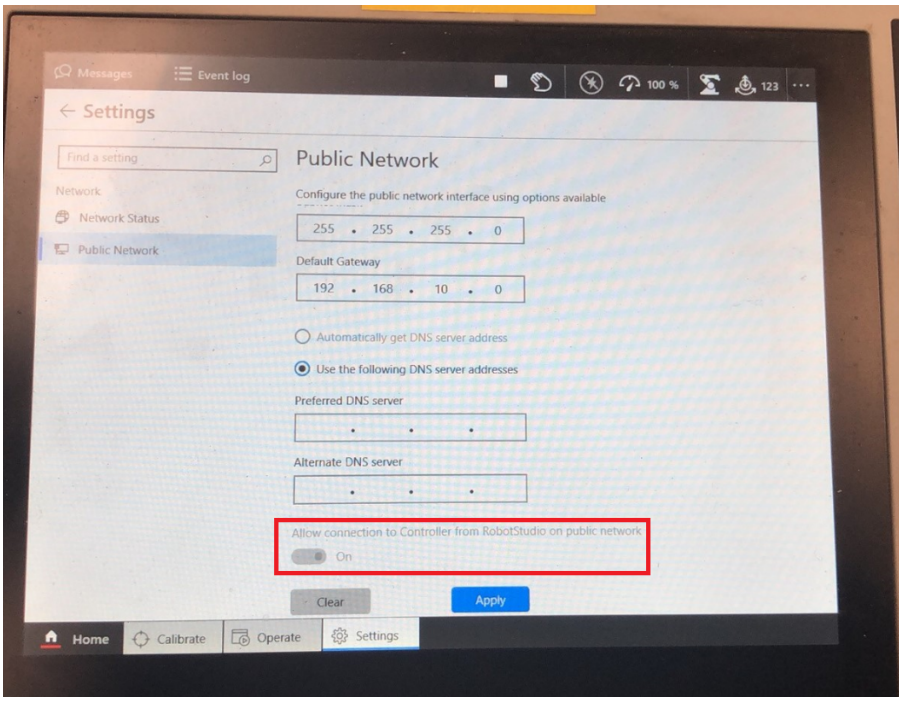
"IEEE1588" -EnableOnPublicNet

"Netscan" -EnableOnPublicNet

"RobAPI" -EnableOnPublicNet

注記

[パブリック ネットワーク上の RobotStudio からコントローラへの接続を許可する] 設定が有効になっていることを確認します。



xx2100000506

リアルコントローラの選択 ダイアログが開きます。

3 ダイアログ ボックスで、接続するリアルコントローラを選択します。

4 OKをクリックして設定を適用します。

5 閉じる をクリックして、コントローラの編集 ダイアログを閉じます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

作業範囲で I/O 信号を変更

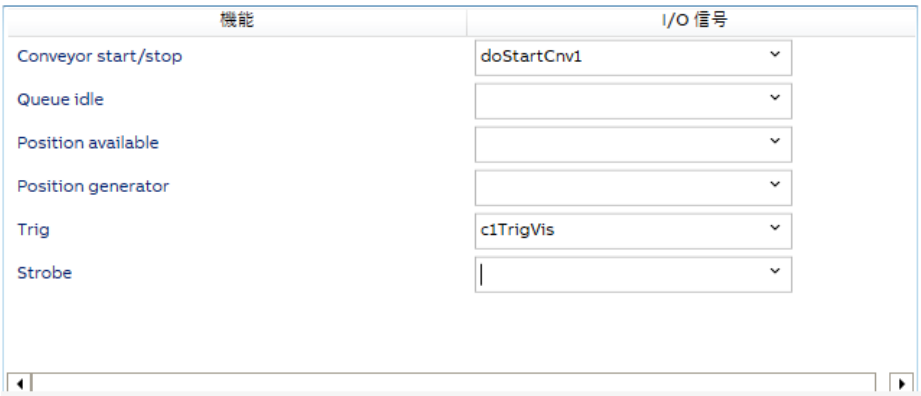
> **注記**
>
> 生産を実行する必要がある作業エリアの **デフォルト** 信号タイプとして設定されていないことを確認してください。
>
> そうでない場合は、本レシピを選択して生産を行う際に警告がポップアップ表示される。
>
> `{0} used default signal type and lacked customized signal type setting. Please check the signal configuration in work area.`

本手順を使用して、作業範囲の I/O 信号を変更。

1. ツリー ビューの **レイアウト** で **コンベア作業エリア 1** を右クリックし、**設定** を選択します。

   コンベア作業エリアの設定ウィンドウが開きます。
2. **シグナルタイプ** タブで **カスタマイズの設定** を選択します。
3. **I/O シグナル**設定表に必要なデータを入力します。詳しくは「*I/O の設定 ページ 185*」を参照してください。

   例：



xx2100001628

4. **OK** をクリックして、**レシピ設定**ウィンドウを閉じます。
5. もう一方の**コンベア WA** に対して、手順 1 ~ 4 を繰り返します。

I/O の設定

I/O 信号

I/O シグナルは、RobotStudio または FlexPendant を使用して設定されます。

信号はすべて事前定義されています。必要に応じて信号を追加してください。

> **注記**
>
> PickMaster の作業エリア信号の最大文字数は 15 文字です。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

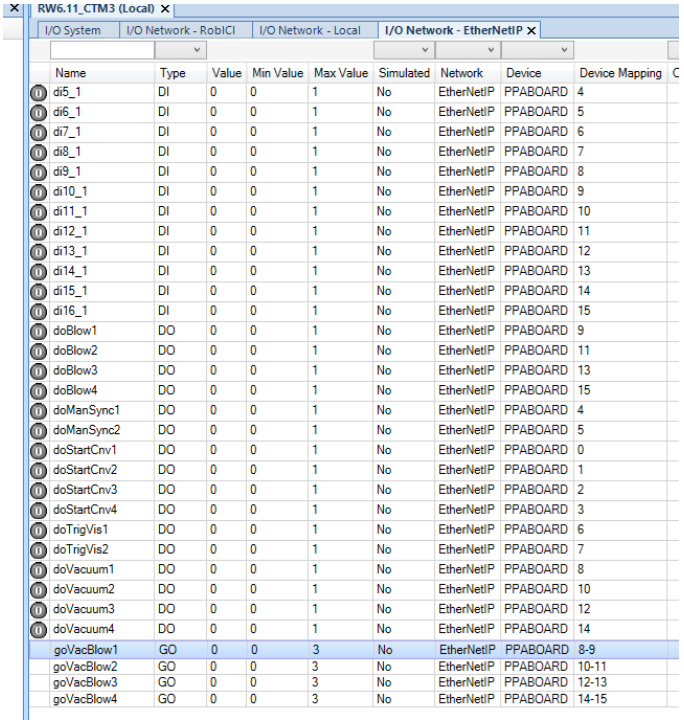
4.4.1 Real Runtimeへの切り替え
続き

PickMaster PowerPacでは、次の I/O シグナルが使用されます。 それらのいくつかは、ソリューションを設定するときに使用または参照されます。 エンコーダシグナルについては、 *Application manual - Conveyor tracking*で説明します。

I/O信号名	説明
di*X*_1	汎用的に使用するためのデジタル入力信号。ポジションジェネレーター、グリッパ圧力スイッチのチェックなど。
doStartCnv*X*[i]	コンベヤの起動/停止用のデジタル出力。
doTrigVis*X*/c*X*TrigVis	画像取得をトリガーするためのデジタル出力。 この信号は、Runtimeがカメラに画像を取得するよう命令するために使用されます。 DSQC 377 の場合、この出力は、対応するエンコーダ ボードの doTrigVis*X* に接続する必要があります。 DSQC 2000 の場合、この出力は c*X*TrigVis に接続する必要があります。 詳細については、回路図を参照してください。
doManSync*X*	コンベアの作業エリアで事前定義されたポジションをトリガーするために使用されるデジタル出力。 DSQC 377 の場合、この出力は、対応するエンコーダ ボードの StartSig (入力 9) に接続する必要があります。 DSQC 2000 の場合、この出力は c*X*TrigVis に接続する必要があります。 詳細については、回路図を参照してください。
doVacuum*X*	吸着に使用されるデジタル出力。例えば、製品をグリップする場合です。出力信号は、アイテムがツールに装着されるべきときに設定されます。 **注記** 信号はRAPIDプログラムから制御されます。シミュレーションでは、信号が設定されているときにRAPID triggdata SimAttachXが制御を行います。実際のロボットでは、信号が設定されているときにRAPID triggdata VacuumActXが制御を行います。
doBlow*X*	通気に使用されるデジタル出力。例えば、ロボットがグリップしている製品をリリースする場合です。出力信号は、アイテムがツールから外されるべきときに設定されます。 **注記** リリース信号はRAPIDプログラムから制御されます。シミュレーションでは、信号が設定されているときにRAPID triggdata SimDetachXが制御を行います。実際のロボットでは、信号が設定されている/発信されたときにRAPID triggdatas VacuumRevXおよびVacuumOffXが制御を行います。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

I/O信号名	説明
goVacBlow*X*	*doVacuumX* と *goVacBlowX* を含むデジタル I/O グループ。 **注記** このシグナルグループが同じボードで定義されていない場合、対応する IO ボードで 4 つの goVacBlow*X* シグナルグループを定義する必要があります。  xx2100001673

RW6.11_CTM3 (Local) ×

I/O System | I/O Network - RobICI | I/O Network - Local | I/O Network - EtherNetIP ×

Name	Type	Value	Min Value	Max Value	Simulated	Network	Device	Device Mapping
di5_1	DI	0	0	1	No	EtherNetIP	PPABOARD	4
di6_1	DI	0	0	1	No	EtherNetIP	PPABOARD	5
di7_1	DI	0	0	1	No	EtherNetIP	PPABOARD	6
di8_1	DI	0	0	1	No	EtherNetIP	PPABOARD	7
di9_1	DI	0	0	1	No	EtherNetIP	PPABOARD	8
di10_1	DI	0	0	1	No	EtherNetIP	PPABOARD	9
di11_1	DI	0	0	1	No	EtherNetIP	PPABOARD	10
di12_1	DI	0	0	1	No	EtherNetIP	PPABOARD	11
di13_1	DI	0	0	1	No	EtherNetIP	PPABOARD	12
di14_1	DI	0	0	1	No	EtherNetIP	PPABOARD	13
di15_1	DI	0	0	1	No	EtherNetIP	PPABOARD	14
di16_1	DI	0	0	1	No	EtherNetIP	PPABOARD	15
doBlow1	DO	0	0	1	No	EtherNetIP	PPABOARD	9
doBlow2	DO	0	0	1	No	EtherNetIP	PPABOARD	11
doBlow3	DO	0	0	1	No	EtherNetIP	PPABOARD	13
doBlow4	DO	0	0	1	No	EtherNetIP	PPABOARD	15
doManSync1	DO	0	0	1	No	EtherNetIP	PPABOARD	4
doManSync2	DO	0	0	1	No	EtherNetIP	PPABOARD	5
doStartCnv1	DO	0	0	1	No	EtherNetIP	PPABOARD	0
doStartCnv2	DO	0	0	1	No	EtherNetIP	PPABOARD	1
doStartCnv3	DO	0	0	1	No	EtherNetIP	PPABOARD	2
doStartCnv4	DO	0	0	1	No	EtherNetIP	PPABOARD	3
doTrigVis1	DO	0	0	1	No	EtherNetIP	PPABOARD	6
doTrigVis2	DO	0	0	1	No	EtherNetIP	PPABOARD	7
doVacuum1	DO	0	0	1	No	EtherNetIP	PPABOARD	8
doVacuum2	DO	0	0	1	No	EtherNetIP	PPABOARD	10
doVacuum3	DO	0	0	1	No	EtherNetIP	PPABOARD	12
doVacuum4	DO	0	0	1	No	EtherNetIP	PPABOARD	14
goVacBlow1	GO	0	0	3	No	EtherNetIP	PPABOARD	8-9
goVacBlow2	GO	0	0	3	No	EtherNetIP	PPABOARD	10-11
goVacBlow3	GO	0	0	3	No	EtherNetIP	PPABOARD	12-13
goVacBlow4	GO	0	0	3	No	EtherNetIP	PPABOARD	14-15

i DSQC 2000 の場合、このシグナル用の定義済みポートはありません。 ボード上の実際に接続されているポートをシグナル名として定義します。

コンベア作業エリアのデフォルト I/O シグナル

シミュレーションにはデフォルトの I/O シグナルが使用されます。

Item（項目）	DSQC 377	DSQC 2000
Conveyor start/stop（コンベヤの起動/停止）	cnv*X*_doStartCnv	cnv*X*_doStartCnv
Queue idle (キューのアイドル状態)	cnv*X*_doQIdle	cnv*X*_doQIdle
Position available (使用可能なポジション)	cnv*X*_doPAvail	cnv*X*_doPAvail
Position generator（ポジションジェネレータ）	cnv*X*_diPosGen	cnv*X*_diPosGen
Trig（トリガ）	doTrigVis*X*	c*X*TrigVis
Strobe（ストロボ）	c*X*NewObjStrobe	c*X*NewObjStrobe

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.1 Real Runtimeへの切り替え
続き

コンベア作業エリアのカスタマイズされた I/O シグナル

カスタマイズされた I/O シグナルは生産に使用されます。

Item（項目）	DSQC 377	DSQC 2000
Conveyor start/stop（コンベヤの起動/停止）	doStartCnv*X* **注記** コンベアが稼働中の場合、このシグナルは空のままにすることができます。	doStartCnv*X* **注記** コンベアが稼働中の場合、このシグナルは空のままにすることができます。
Queue idle (キューのアイドル状態)		
Position available (使用可能なポジション)		
Position generator（ポジションジェネレータ）		
Trig（トリガ）	doTrigVis*X*	c*X*TrigVis
Strobe（ストロボ）	c*X*NewObjStrobe	

インデックスワークエリアのデフォルト I/O シグナル

シミュレーションにはデフォルトの I/O シグナルが使用されます。

Item（項目）	DSQC 377	DSQC 2000
Conveyor start/stop（コンベヤの起動/停止）		
Queue idle (キューのアイドル状態)	ind*X*_doQIdle	ind*X*_doQIdle
Position available (使用可能なポジション)	ind*X*_doPAvail	ind*X*_doPAvail
Position generator（ポジションジェネレータ）	ind*X*_diPosGen	ind*X*_diPosGen
Trig（トリガ）	ind*X*_doTrigVis	ind*X*_doTrigVis
Strobe（ストロボ）	ind*X*_diStrobe	ind*X*_diStrobe

インデックスワークエリアのカスタマイズされた I/O シグナル

カスタマイズされた I/O シグナルは生産に使用されます。

Item（項目）	DSQC 377	DSQC 2000
Conveyor start/stop（コンベヤの起動/停止）		

次のページに続く

© Copyright 2025 ABB. All rights reserved.

Item（項目）	DSQC 377	DSQC 2000
Queue idle (キューのアイドル状態)	doTrigVis*X*[i] **注記** **キュー アイドル シグナルと Strobe シグナルは同じでなければなりません。**	cXTrigVis[i] **注記** **キュー アイドル シグナルと Strobe シグナルは同じでなければなりません。**
Position available (使用可能なポジション)		
Position generator（ポジションジェネレータ)		
Trig（トリガ)		
Strobe（ストロボ)	doTrigVis*X*	cXTrigVis

[i] 使用可能な任意の do シグナルを使用できます。

注記

グリッパーの起動シグナル設定が接続されたコントローラと完全に一致していることを確認してください。

つまりPickMaster PowerPacで、グリッパーはアイテムを選択または配置しない。

© Copyright 2025 ABB. All rights reserved.

4.4.2 カメラの設定

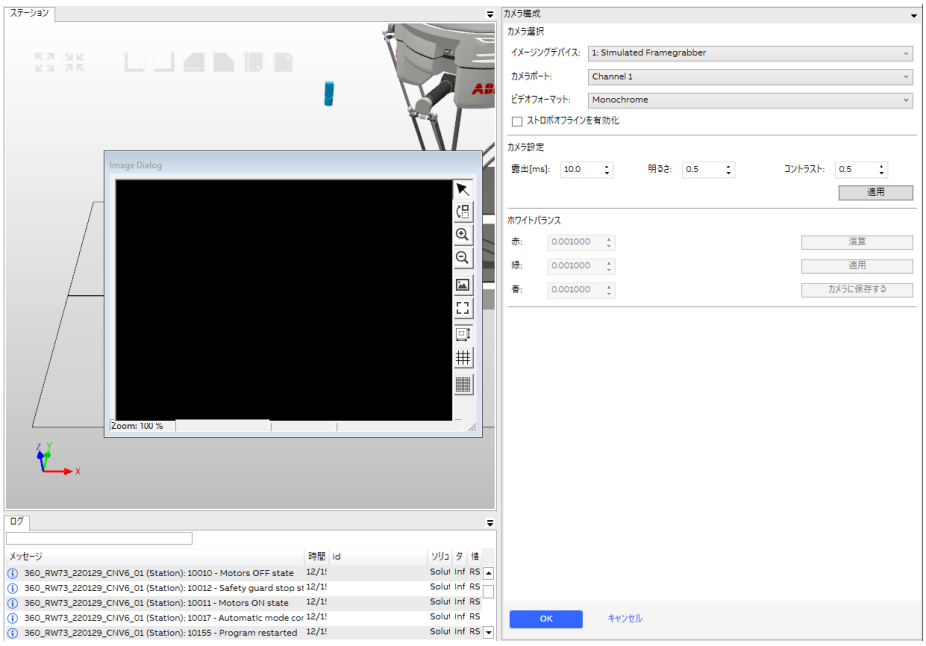
概要

> **注記**
>
> ファイアウォールまたはウイルス対策ソフトウェアがインストールされている場合は、pickmasteru.exe および visionclient.exe をホワイト リストに追加します。
>
> そうしないと、PickMaster PowerPac はRuntime に接続できず、ビジョン機能が正常に動作しません。

視覚ありモデルのカメラは、カメラ視野で物体を見つけるるために使用します。ツリー ビューでカメラを作成した場合、物理的にカメラと接続されません。カメラへの接続は設定ウィンドウで手動にて行う必要があります。ツリー ビュー内のカメラは、1 台のカメラを使用するように設定されます。 また、カメラは最適な画像を得ることができるように設定する必要があります。

カメラを設定るには、

1 ツリー ビューの**レイアウト**でカメラを右クリックし、**設定** を選択します。
カメラの設定 ダイアログと **画像ダイアログ**が開きます。



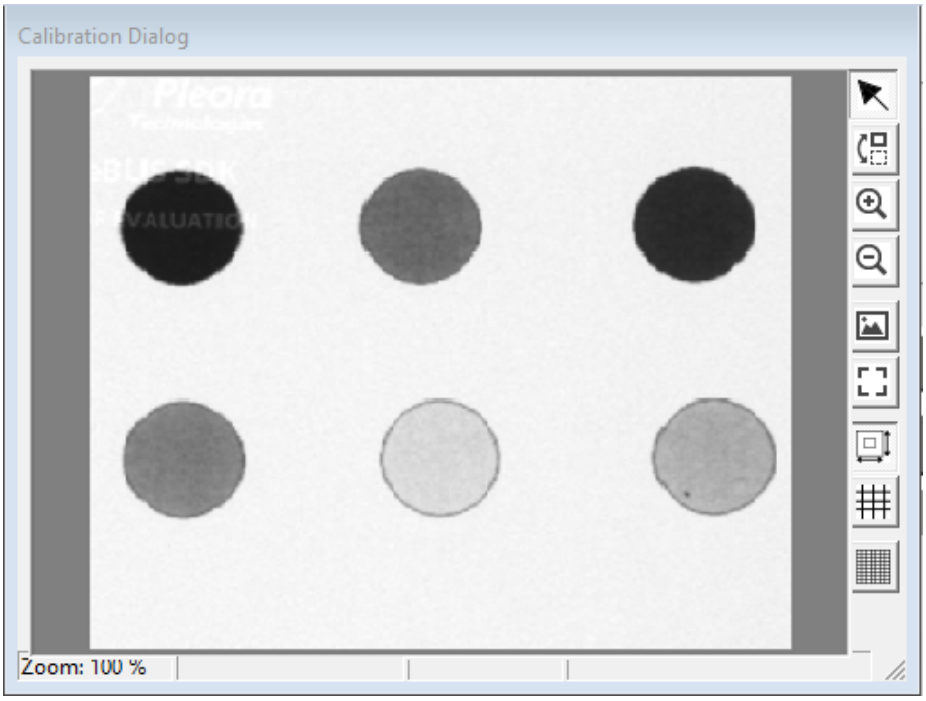
xx2100001622

2 ［フレーム取り込み］リスト内で、カメラが接続されているGigabit Ethernet カメラを選択します。

3 ［ビデオ形式］リスト内で、接続されているカメラのタイプを選択します。
画像ダイアログに画像が表示されます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2100001521

4 **設定ファイル** タップでは、ABB 検証済みカメラの設定ファイルが自動的に読み込まれます。

ABB 検証済みでないカメラの設定については、複数ブランドのカメラの設定 ページ 383を参照してください。

5 生産モード以外でカメラにストロボを実行させる場合には、［ストロボをオフラインで有効化］チェックボックスを選択します。これは、カメラをストロボライトと共に使用する場合などに必要となります。この設定は Gigabit Ethernet カメラのみに適用されます。

6 必要に応じて、［露出］、［明るさ］、および［コントラスト］ を選択し、［カメラの設定］欄の［適用］をクリックします。

露出を調節して、可能な限り最適な画像を取得します。レンズの絞りとカメラの露出は被写界深度とピントボケを調整します。これらの2つのパラメータは、探そうとしているアイテムとコンベヤの速度に従って、適切に調節しなければなりません。

最良の画像を取得するために、明るさとコントラストも変更できます。中には、明るさとコントラストのパラメータと共に照明を調節した方が容易に検出するアイテムもあります。

これらのパラメータ値の変更の効果は［適用］をクリックするまでわかりません。

7 選択したカメラがカラーカメラであり、カラービデオフォーマットと共に使用される場合には、下記の手順でカメラのホワイトバランスのキャリブレーションを行う必要があります。

a カメラの下に白紙を置きます。この紙は視野全体をカバーする必要があります。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

b 画像が灰色（白と黒のちょうど中間）に見えるように照明を調節します。レンズの絞りまたはカメラの露出時間で調節します。

c [ホワイトバランス]欄で[計算]をクリックします。これにより、ホワイトバランスのキャリブレーションパラメータが計算されます。

d [適用]をクリックします。これにより、カメラの内部設定が修正されます。

e [カメラに保存]をクリックします。これにより、カメラに設定が保存されます。

カラービジョンの詳細については、「カラービジョンの使用 ページ324」を参照してください。

8 [OK] をクリックします。

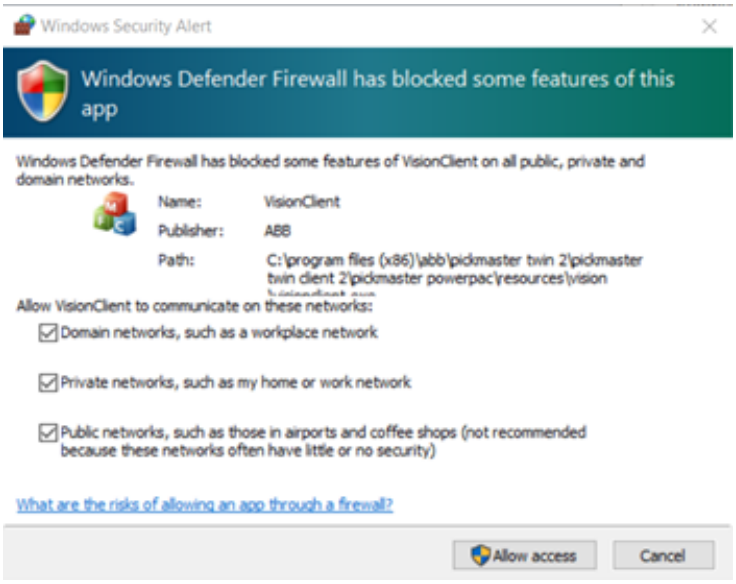
9 必要に応じて、カメラ パラメータの露出、明るさ、コントラストをレシピ カメラ パラメータ設定ページに同期します。

注記

VisionClient を開くときに問題が発生した場合は、次の考えられる理由を確認してください。

1 ファイアウォールのブロック、

2 **VisionClient** は、すべてのネットワークで通信できるわけではありません。



xx2200001061

シミュレーションカメラの設定

PickMasterにビジョン機能は、カメラの実機を接続しなくても使用できます。この目的は、任意のラップトップまたはPC上でビジョンドングルのみを接続した状態でのビジョンモデリングおよび評価を可能にすることです。カメラの実機から画像を取得する代わりに、ファイルから画像を読み込みます。この機能はオフラインでの使用のみを目的としており、生産モードではサポートされません。

ドングルには、標準カメラドングルとシミュレーションドングルの2種類があります。プログラムの起動時にカメラがないと、標準ドングルを接続したときに

次のページに続く

© Copyright 2025 ABB. All rights reserved.

PickMasterは自動的にシミュレーションモードになります。シミュレーションドングルではカメラからの画像取得が無効になるため、すべての画像をファイルから読み込む必要があります。

シミュレーションカメラを設定するには、下記の手順に従ってください。

1 ツリー ビューのセルでカメラを右クリックし、設定 を選択します。
［カメラの構成］ウィンドウが開きます。
2 [画像デバイス（Imaging device）］リストで[シミュレーションフレームグラバ（Simulated framegrabber）]を選択します。

> **注記**
>
> 8100D フレームグラバは、PickMaster 3.41 以降との互換性はありません。

3 ポートを設定します。
4 [ビデオフォーマット（Video format）]を、カラー表示またはモノクロ画像に設定します。
5 画像を読み込みます。各種ビジョンダイアログで画像を読み込むには、2つの方法があります。
- 各種ビジョンダイアログで[インポート（Import）]ボタンを使用して任意のフォルダから画像を読み込む。
- 登録された画像フォルダから画像ファイルを読み込む。各カメラには、モデリング用のデフォルト画像とキャリブレーション用の画像セットがあり、キャリブレーションダイアログで[取得（Acquire）]を押して切り替えることができます。これを使用するためには、下で説明するように、登録された画像をインストールするための追加設定が必要です。

6 ［OK］をクリックします。

関連情報

カラービジョンの使用 ページ *324*。
カメラのキャリブレーション ページ *282*。

© Copyright 2025 ABB. All rights reserved.

4.4.3 キャリブレーションロボット

命令

ロボットのキャリブレーション方法の詳細については、ロボットの製品マニュアルに記載されています。

© Copyright 2025 ABB. All rights reserved.

4.4.4 リニア コンベアのキャリブレーション

概要

> **注記**
>
> 生産およびエミュレーションを実行する場合、次のキャリブレーション プロセスが必要です。 PickMaster PowerPac の [シミュレーション] タブでのキャリブレーションでは、次のキャリブレーション プロセスは完了しません。

コンベアに必要なキャリブレーションは、カメラと作業エリアのキャリブレーションです。作業エリアのキャリブレーションは、コンベア作業エリアではベースフレームのキャリブレーション、インデックスワークエリアでは作業物体の定義になります。ここで重要なのは、カメラ座標とロボットベースフレームまたは作業物体を同じ起点で定義することです。

カメラのキャリブレーションはカメラ毎に実行しなければなりません。また、ベースフレームのキャリブレーションはコンベヤ毎に実行する必要があります。

カメラのキャリブレーションはソリューションに保存されるため、そのソリューション内のすべてのレシピで同じキャリブレーションを共有できます。 カメラを再キャリブレーションする必要がある場合、ソリューション内のすべてのレシピが新しいキャリブレーションで更新されます。

カメラのキャリブレーションとベースフレームのキャリブレーションは別々に実行することができます。ただし、ワークエリアのキャリブレーション後に新しく正確なカメラのキャリブレーションを行うことは非常に困難です。

ワークエリアのキャリブレーションはロボットコントローラに保存されます。

リニア コンベアをキャリブレーションするには:

1. エンコーダからDSQC 2000へのケーブル接続については、エンコーダからリニアコンベアのDSQC 2000までのケーブル接続 ページ 196を参照してください。
2. パラメータ *Counts Per Meter* を定義します (コンベアのみ)。パラメータ「カウント/メートル」の定義 ページ 197、パラメータ「カウント/メートル」の定義 ページ 219 を参照してください。
3. カメラのキャリブレーションを行います。ベース フレームの定義 ページ 199, ベース フレームの定義 ページ 221を参照してください。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.4.1 DSQC 2000 を使ったリニアコンベアのキャリブレーション

4.4.4.1.1 エンコーダからリニアコンベアのDSQC 2000までのケーブル接続

概要

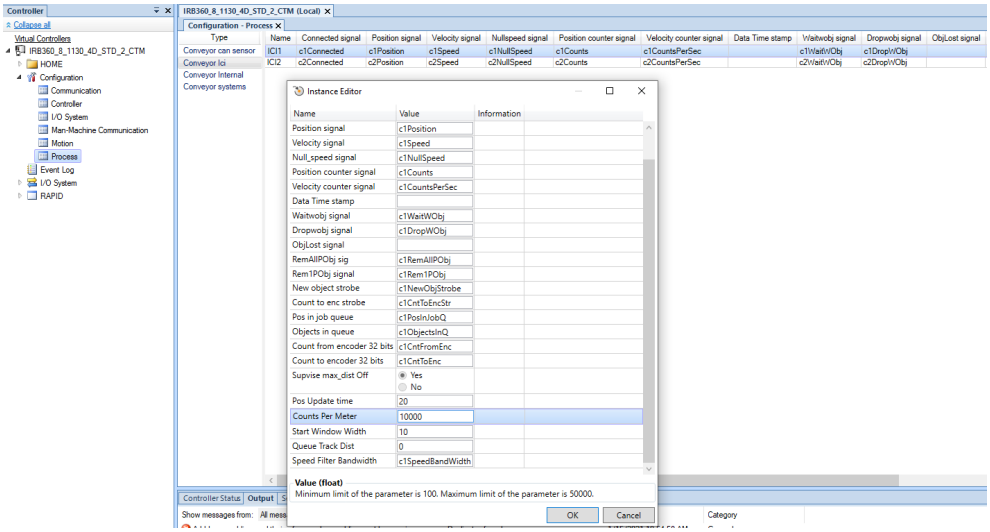
リニアコンベアのDSQC 2000までのエンコーダの接続については、*Application manual - Conveyor tracking*を参照してください。

© Copyright 2025 ABB. All rights reserved.

4.4.4.1.2 パラメータ「カウント/メートル」の定義

概要

*Counts Per Meter*システムパラメータは、コンベアエンコーダのキャリブレーションに使用します。*Counts Per Meter*システムパラメータは、トピック*Process*のタイプ*Conveyor Ici*に属します。



xx2100000042

Counts Per Meterの計算

Counts Per Meter システム パラメータの値は、次のように計算されます。

```
counts value/measured_meters
```

値	説明
`counts value` (`position1-position0`)	DSQC 2000については、FlexPendant またはRobotStudioの事前定義済みの I/O シグナルから読み取ります。 たとえば、CNV1,ではシグナル名は c1countsです。
`measured_meters`(m)	コンベアが移動した手動測定距離 (メートル単位)。

Counts Per Meterの定義

以下の手順に従って、コンベヤエンコーダーの*Counts Per Meter*を定義します。

1. コンベヤベルトにマークを付けます。たとえば、線を引いたり、テープを貼り付けたりします。コンベアの固定側の同じポジションにもマークを付けます。
2. OmniCore の場合、FlexPendant **I/O**-> **RobICI**-> **CTM** ウィンドウで、コンベアの c1countsを読み取ります。これは`position0` です。

   IRC5 の場合、FlexPendant **入力および出力**-> **I/O デバイス**-> **RobICI**-> **CTM** ウィンドウで、コンベアの c1countsを読み取ります。これは`position0` です。
3. コンベヤベルトを約1メートル移動させます。
4. FlexPendant ジョギング ウィンドウで、コンベアの c1countsを読み取ります。これは`position1` です。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.4.1.2 パラメータ「カウント/メートル」の定義
続き

5 2つのマーク間の物理的な距離を測定します。これがmeasured_metersの値です。
6 counts value= position1- position0 を計算します。
7 counts valueと測定値を使用して、*Counts Per Meter*を計算します。
例: 20200/1.005 = 20099
8 RobotStudioで、［構成］を開き、トピックProcessを選択し、Conveyor Iciを入力します。
9 ユニット*ICIx*（x はコンベア番号）を選択し、パラメータ*Counts Per Meter*の値を編集します。
10 ［OK］をタッチします。
11 コントローラを再起動します。

関連情報

『*Technical reference manual - System parameters*』。

© Copyright 2025 ABB. All rights reserved.

4.4.4.1.3 ベース フレームの定義

概要

コンベヤのコンベヤワークエリアごとに、コンベヤベースフレームのキャリブレーションを行います。ベースフレームのキャリブレーションを行うことで、ワークエリアでセンサー検出オブジェクトのピックまたはプレイスを行うときのロボットの基準点を求められます。

準備

- 各コンベア作業エリアのCounts Per Meter(メーターごとのカウント) システム パラメータを定義します。 詳細については、パラメータ「カウント/メートル」の定義 ページ 219, パラメータ「カウント/メートル」の定義 ページ 197を参照してください。
- ロボットに一時的に取り付けるキャリブレーションツールを準備します。このキャリブレーションツールにはポイントが付いたTCPがあるため、TCPオフセット値を正確に測定します。
- 各ロボットについて、RAPIDプログラムでキャリブレーションツールのツールデータを作成します。TCPオフセットを測定値で更新します。FlexPendant ジョグウィンドウで、ロボットのツールデータを選択します。
- カメラを使用する場合は、カメラをキャリブレーションします。カメラのキャリブレーション ページ 282を参照してください。カメラのキャリブレーション後、カメラキャリブレーションパターンをコンベアに添付します。

OmniCore の手順

次の手順を使用して、OmniCore コントローラを備えたラインのコンベアのすべてのベース フレームをキャリブレーションします。

1 コンベヤベルトにキャリブレーションの基準点が正確にマークされていることを確認してください。
   - カメラを使用する場合、基準点はカメラビューのローカル起点です。カメラのキャリブレーションが終了すると、コンベヤに添付されたカメラキャリブレーションパターンの起点によって基準点がすでにマークされています。
   - I/Oセンサーを使用して事前定義のポジションを生成する場合は、コンベヤ上で、センサーがオブジェクトを検出するポイントに基準点をマーク付けします。このポイントは、検出されたアイテムまたはコンテナのローカル起点になります。
2 コンベヤ（エンコーダーボード）ポジションをリセットします。

> **注記**
>
> このステップが完全に終了するまで、コンベヤを動かさないでください。

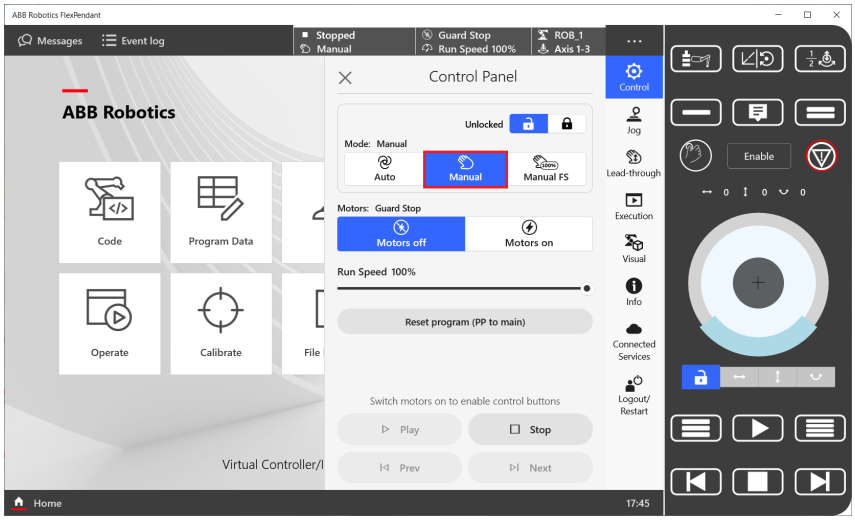
コンベヤに沿ってキャリブレートする必要のある、ワークエリアを持ったすべてのロボットについて、以下を行います。

- コントローラを手動モードに設定します。

次のページに続く

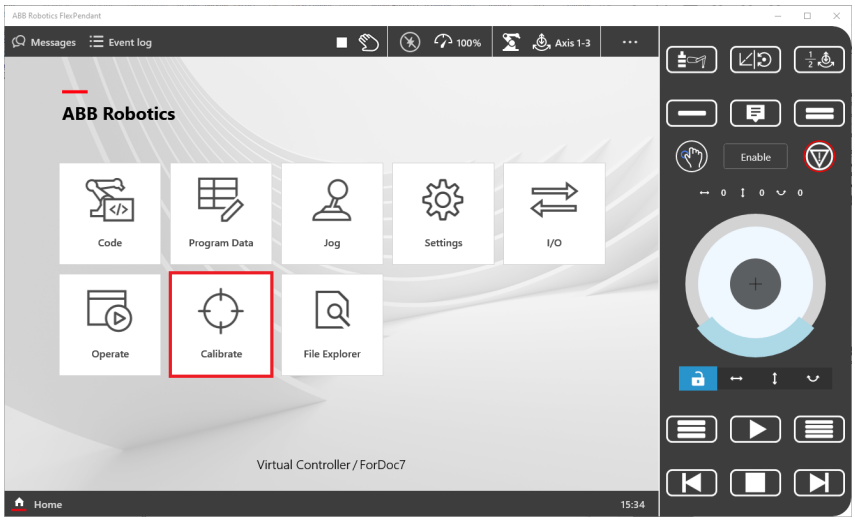
© Copyright 2025 ABB. All rights reserved.

4.4.4.1.3 ベース フレームの定義
続き



xx2100000368

- FlexPendant で、**キャリブレート** をクリックします。

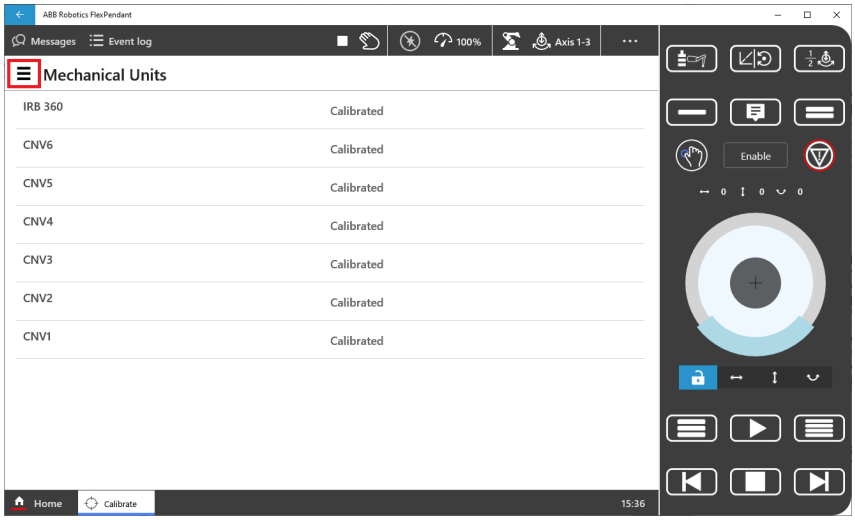


xx2100000362

- 左上隅の **オプション タブ** をクリックします。

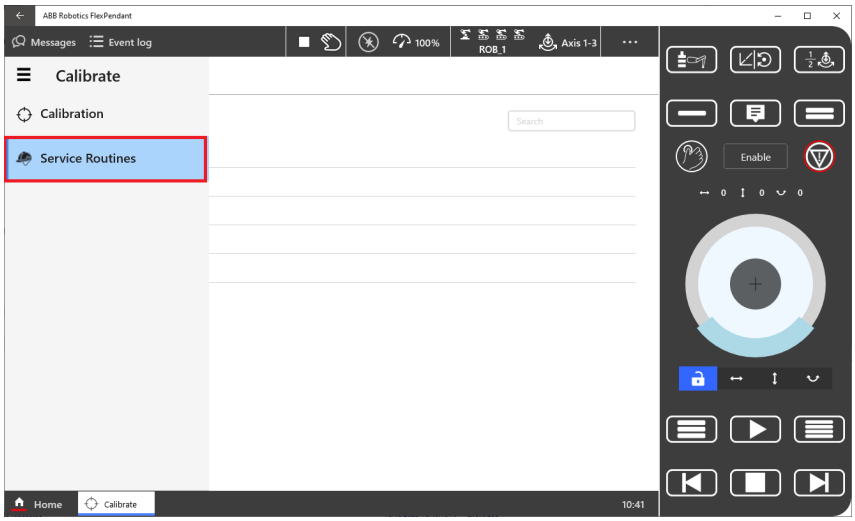
次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2100000363

- **サービス ルーチン** をクリックします。



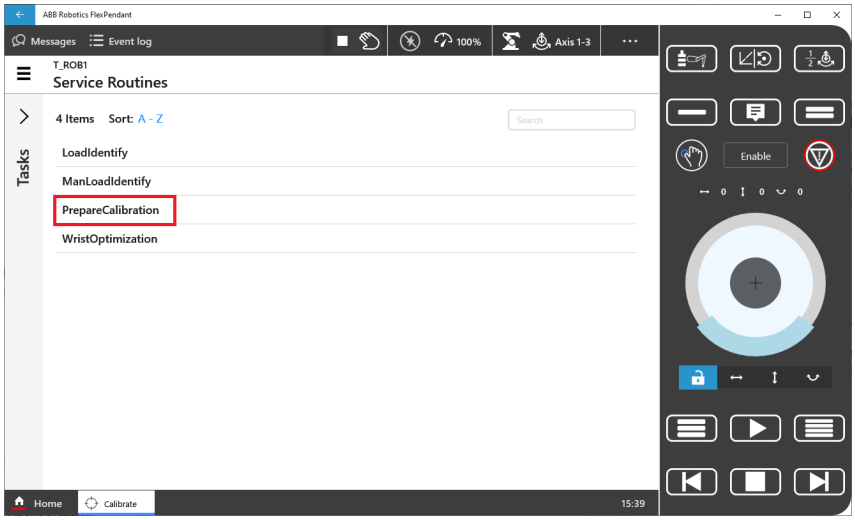
xx2100000364

- **キャリブレーションの準備** をクリックします。

次のページに続く

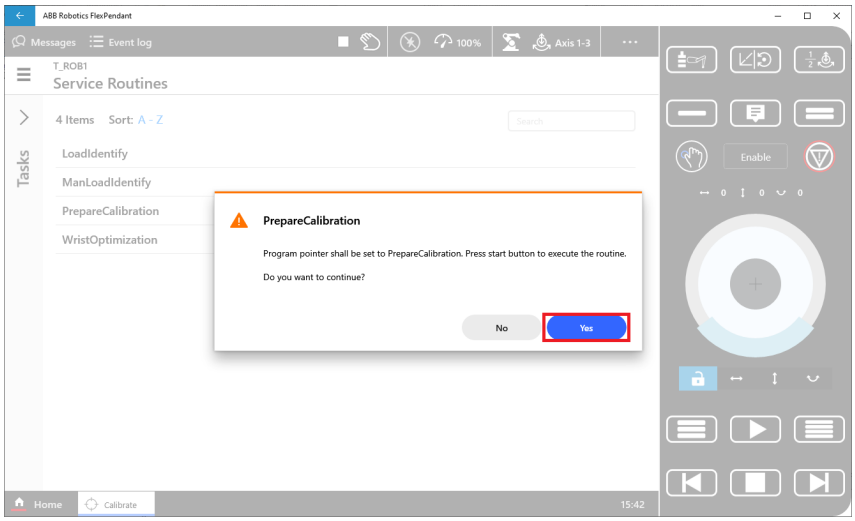
© Copyright 2025 ABB. All rights reserved.

4.4.4.1.3 ベース フレームの定義
続き



xx2100000365

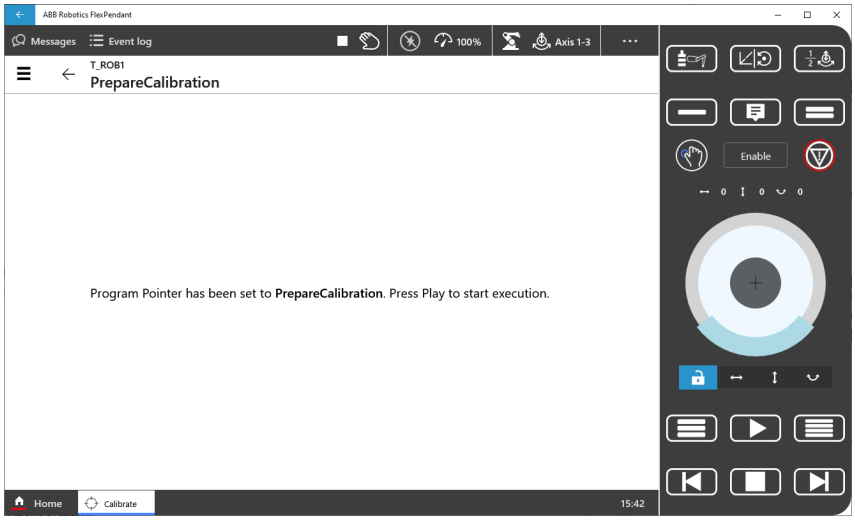
- 表示されたダイアログで はい をクリックします。



xx2100000366

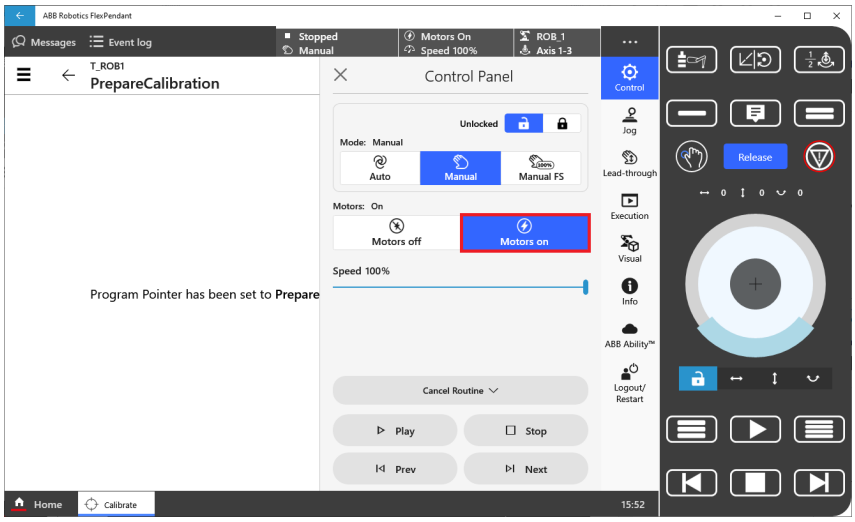
次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2100000367

- コントローラのモーターへの有効化デバイス ボタンを有効にします。



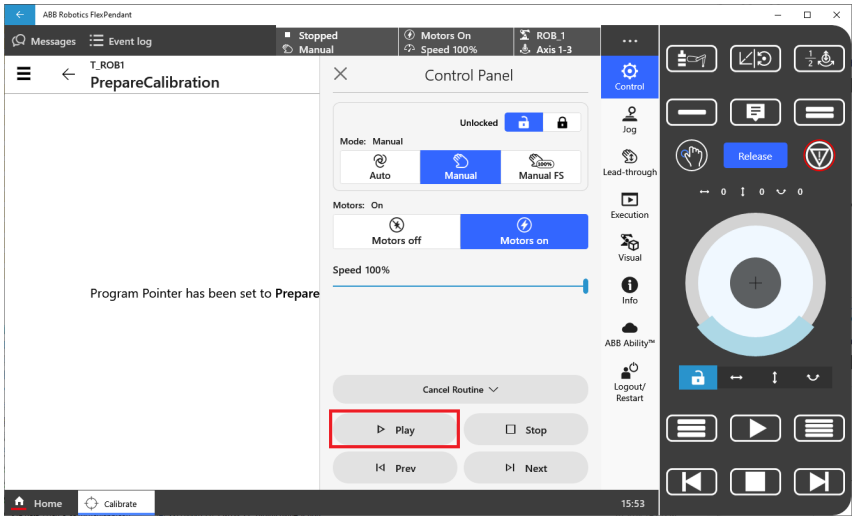
xx2100000369

- 再生 をクリックします。

次のページに続く

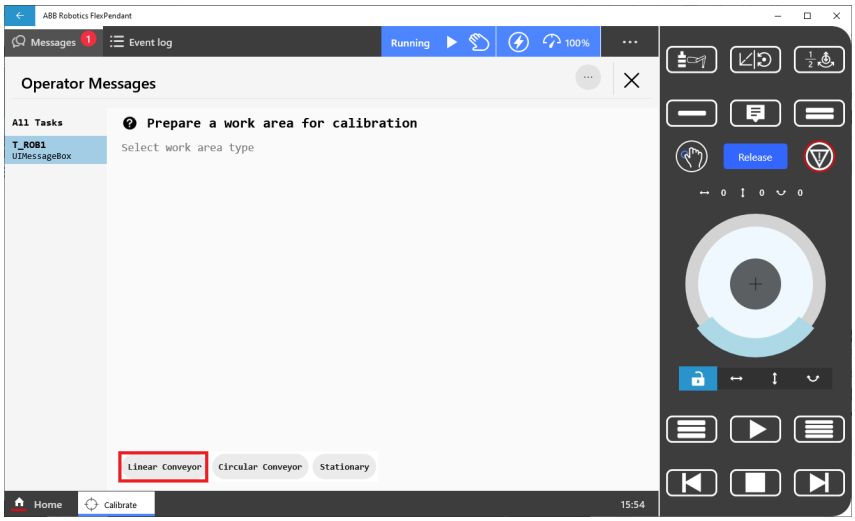
© Copyright 2025 ABB. All rights reserved.

4.4.4.1.3 ベース フレームの定義
続き



xx2100000370

- 作業エリアのタイプとしてリニア コンベアを選択します。

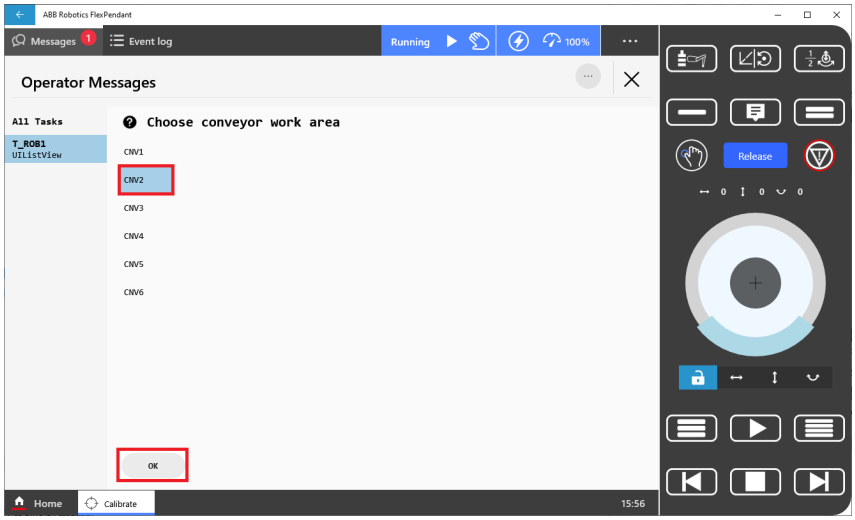


xx2100000371

- コンベアを選択します:例えば、CNV2。次に、**OK**をクリックします。

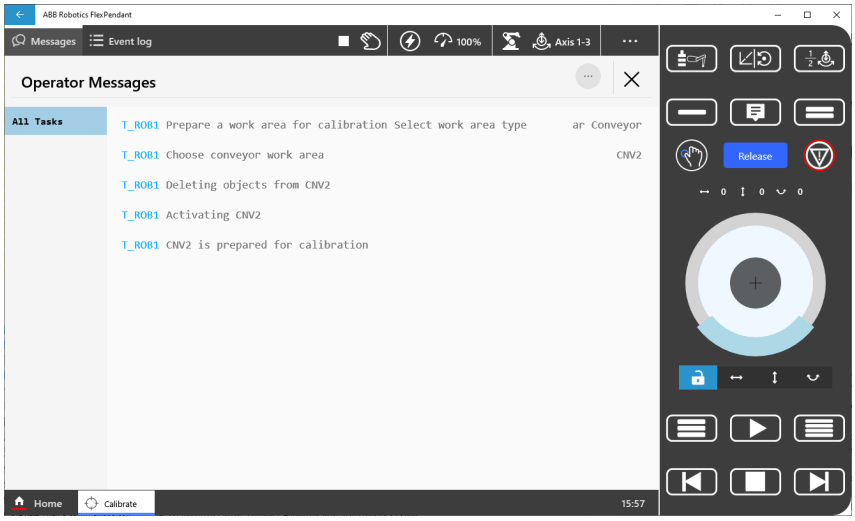
次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2100000372

- メッセージ...キャリブレーションの準備ができています を待ちます。CNV2 のジョギング ウィンドウのコンベアポジションは、「0」mm として表示されるはずです。



xx2100000395

3 コンベヤベルトを前方に動かして、基準点が次のキャリブレーション対象ロボットの動作範囲に入るようにします。

ジョグウィンドウで、すべてのコンベヤワークエリアのコンベヤポジションは、基準点について同じ総移動距離を示すはずです。カメラまたはセンサーに最も近いロボットが最初にキャリブレートされ、次に2番目に近いロボット、といった順序で、コンベヤ沿いのすべてのロボットがキャリブレートされます。

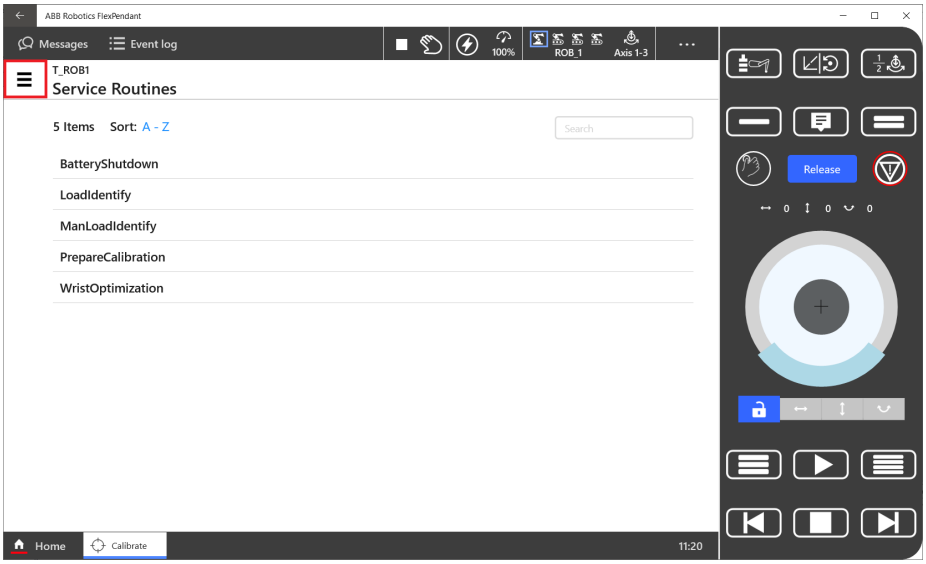
4 キャリブレーションツールをロボットに取り付けます。

5 FlexPendant で、左上隅の[オプション] タブをクリックします。

次のページに続く

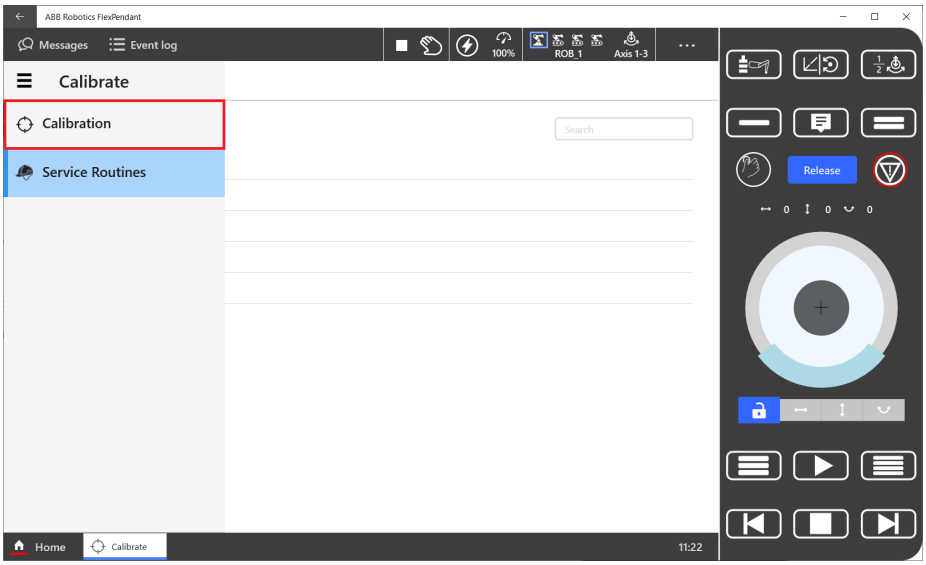
© Copyright 2025 ABB. All rights reserved.

4.4.4.1.3 ベース フレームの定義
続き



xx2100000373

6 [キャリブレーション] をクリックします。

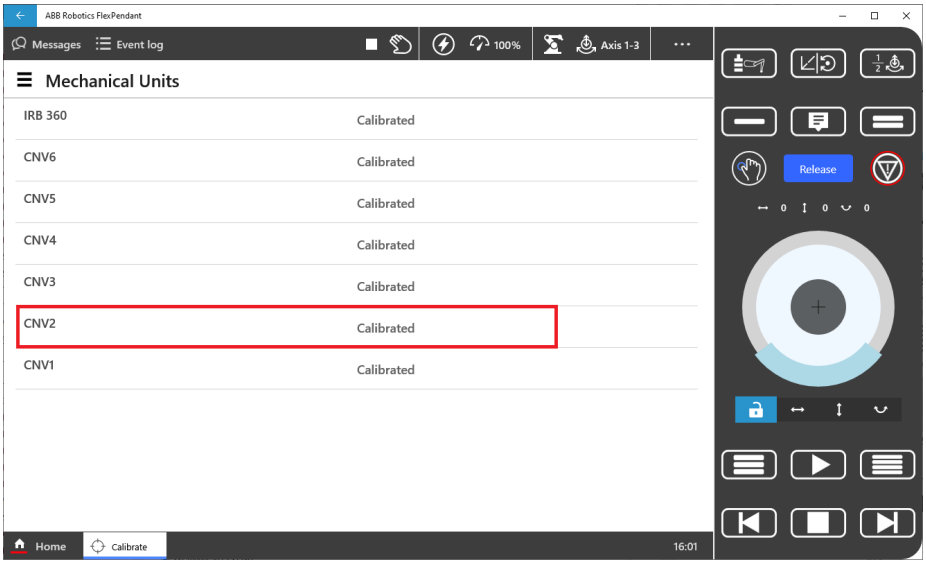


xx2400001469

7 コンベアを選択します。例：CNV2

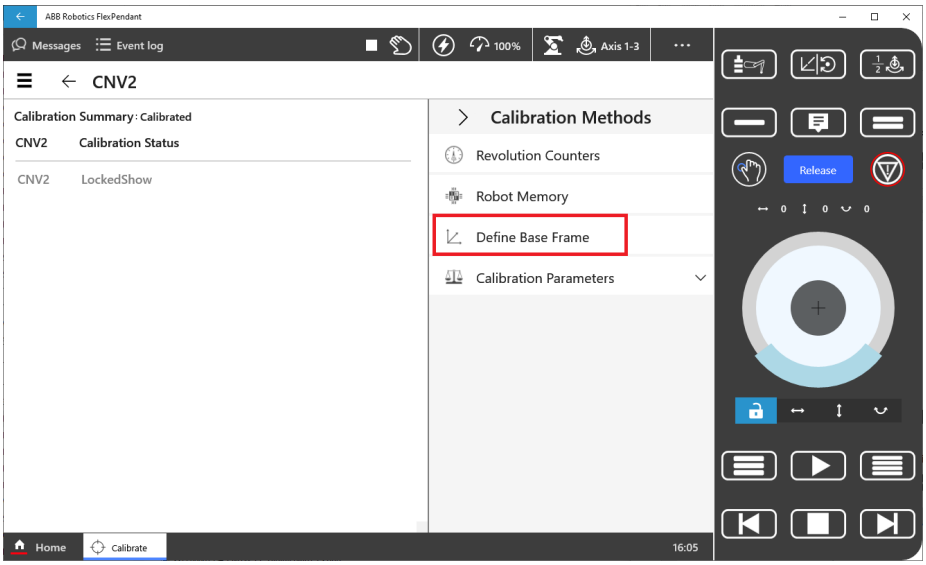
次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2100000374

8 **[キャリブレーション方法] -> [ベース フレームの定義] をタップします。**



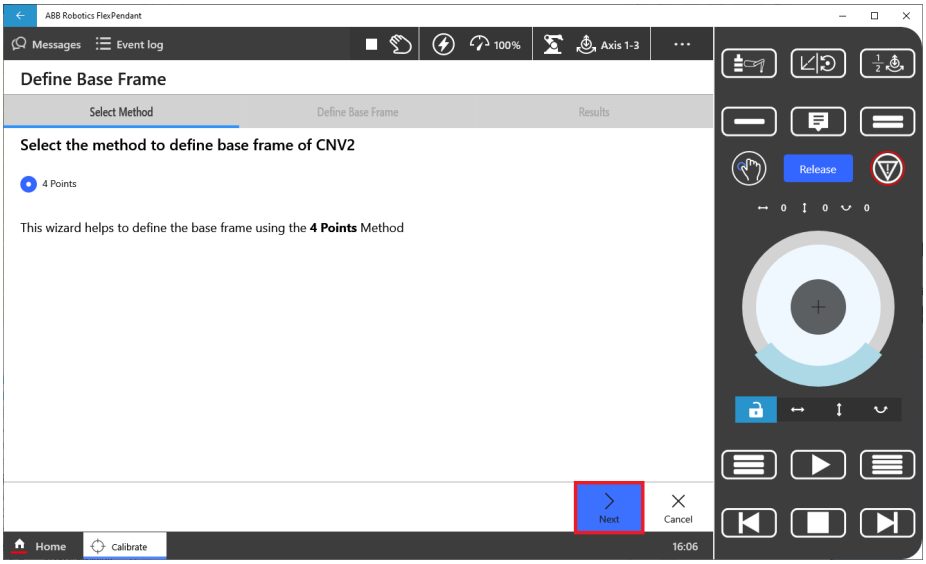
xx2100000375

9 **4 ポイント をタップし、次へ をクリックします。**

次のページに続く

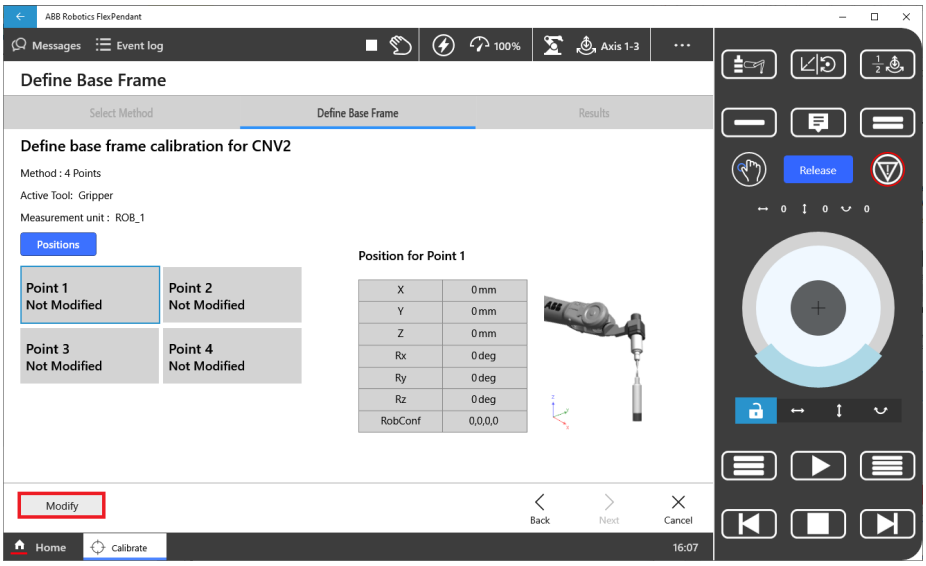
© Copyright 2025 ABB. All rights reserved.

4.4.4.1.3 ベース フレームの定義
続き



xx2100000376

10 最初のポイント [ポイント 1 を選択します。
11 ロボットを手動でジョグまたは移動します。コンベヤ上の基準点をキャリブレーションツールTCPで正確にポイントします。
12 ポジションの変更 ファンクションキーをタップして、選択したポイント (ポイント 1) を変更します。



xx2100000377

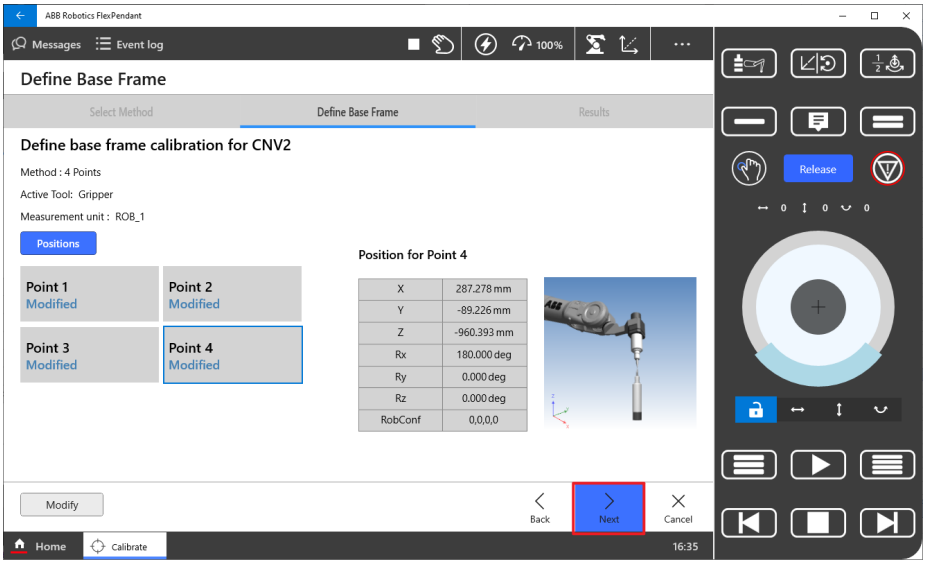
13 ロボットが基準点に到達できる範囲の距離だけ、コンベヤベルトを前方に動かします。

4つのキャリブレーションポイント間（Point 1 ~ 4）の距離を長く均等に空けることをお勧めします。そうすることで、キャリブレーションの精度が向上するからです。

14 ポイントPoint 2、Point 3、Point 4についても、ステップ10 ~ 13を繰り返します。
15 次へ をタップして、ベース フレームを計算します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

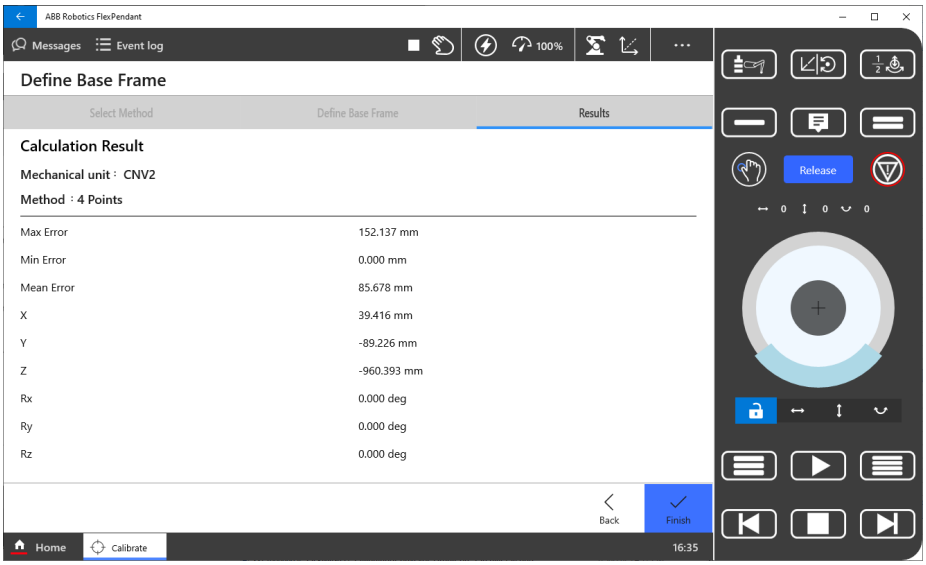


xx2100000378

16 ベース フレーム計算の表示された平均誤差と最大誤差が許容範囲内かどうかを確認します。 推定誤差が許容できる場合は、完了 をタップして確認し、新しいベース フレームを保存します。

注記

ほとんどの場合、1 mm未満の平均誤差は許容されます。



xx2100000379

推定誤差に問題がある場合、このベースフレームは再補正する必要があります。

- コンベヤベルトを後方に動かし、基準点がロボットの作業範囲に入るようにします。すべてのポイントPoint1、Point 2、Point 3、Point 4についてもステップ10 ~ 13を繰り返します。
- コンベヤを後方に動かせない場合は、手順1からやり直してください。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.4.1.3 ベース フレームの定義
続き

17 コンベヤに他にもキャリブレーション対象のロボットがある場合は、ステップ3から繰り返します。
18 次のベースフレームをアクティブにするには、コントローラを再起動します。

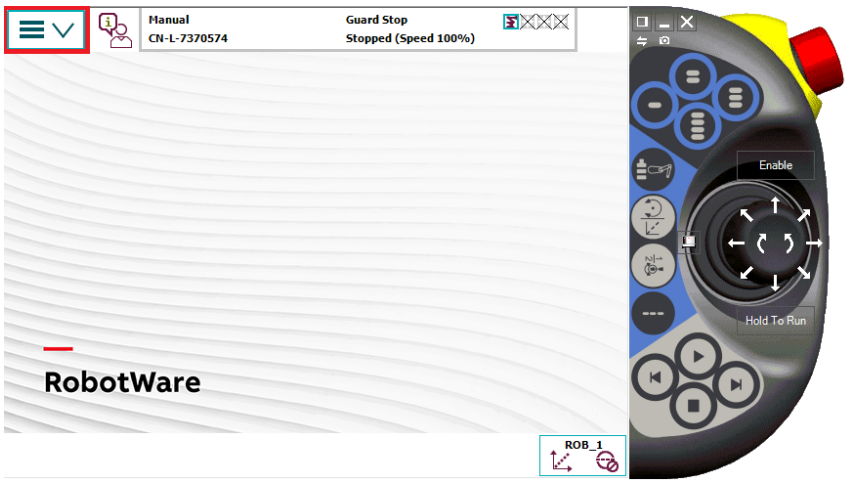
IRC5 の手順

以下の手順を使用して、IRC5 コントローラを備えた円形コンベアのすべてのベースフレームをキャリブレーションします。

1 コンベヤベルトにキャリブレーションの基準点が正確にマークされていることを確認してください。
   - カメラを使用する場合、基準点はカメラビューのローカル起点です。カメラのキャリブレーションが終了すると、コンベヤに添付されたカメラキャリブレーションパターンの起点によって基準点がすでにマークされています。
   - I/Oセンサーを使用して事前定義のポジションを生成する場合は、コンベヤ上で、センサーがオブジェクトを検出するポイントに基準点をマーク付けします。このポイントは、検出されたアイテムまたはコンテナのローカル起点になります。
2 コンベヤ（エンコーダーボード）ポジションをリセットします。

> **注記**
> このステップが完全に終了するまで、コンベヤを動かさないでください。

コンベヤに沿ってキャリブレートする必要のある、ワークエリアを持ったすべてのロボットについて、以下を行います。
- コントローラを手動モードに設定します。
- FlexPendant で、**メニュー** をクリックしてドロップダウン リストを開きます。

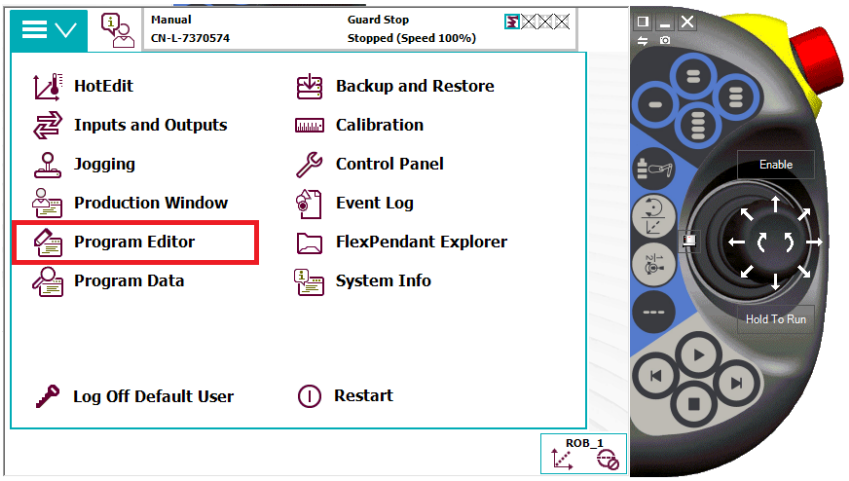


xx2200001925

- ドロップダウン リストで **プログラム エディタ** をクリックします。

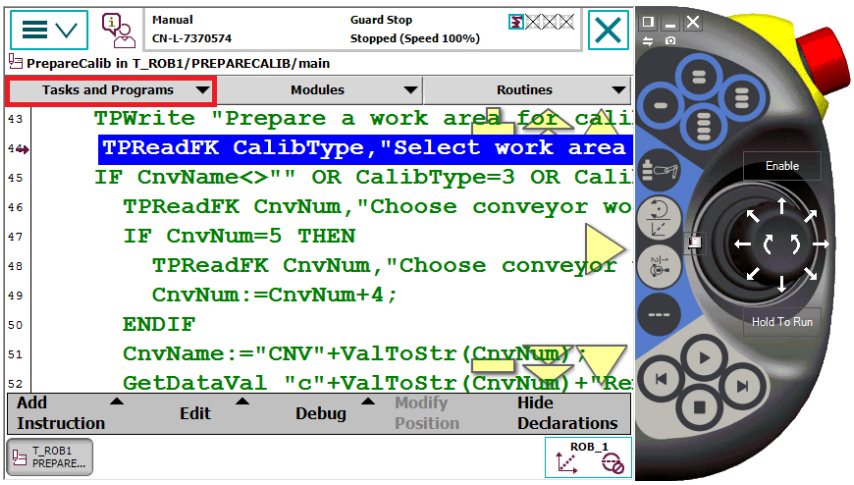
次のページに続く

© Copyright 2025 ABB. All rights reserved.



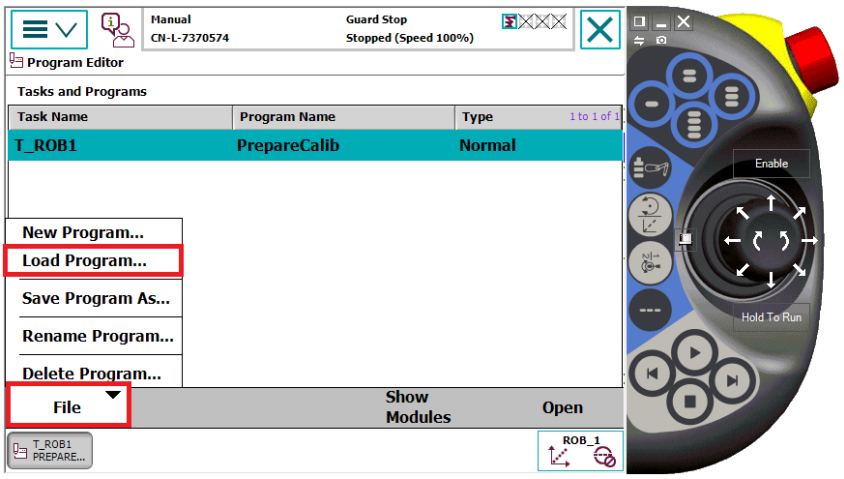
xx2200001926

- タスクとプログラム をクリックします。



xx2200001927

- ファイル とプログラムの読み込み をクリックします。



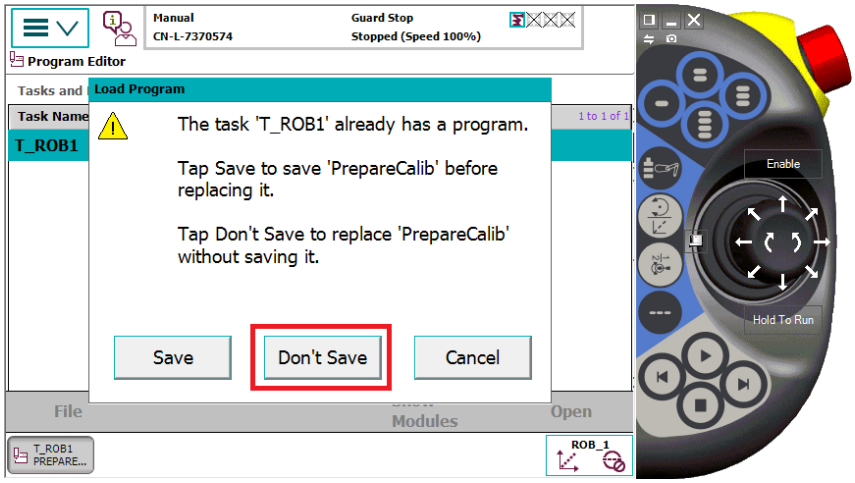
xx2200001928

- 表示されたダイアログで 保存しない をクリックします。

次のページに続く

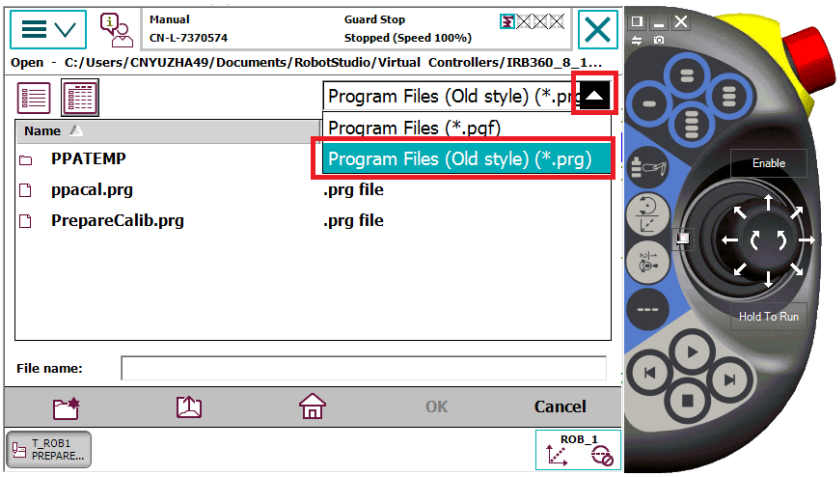
© Copyright 2025 ABB. All rights reserved.

4.4.4.1.3 ベース フレームの定義
続き



xx2200001929

- 右上隅のドロップダウン リストで **プログラムファイル (古いスタイル)(.prg)** をクリックします。

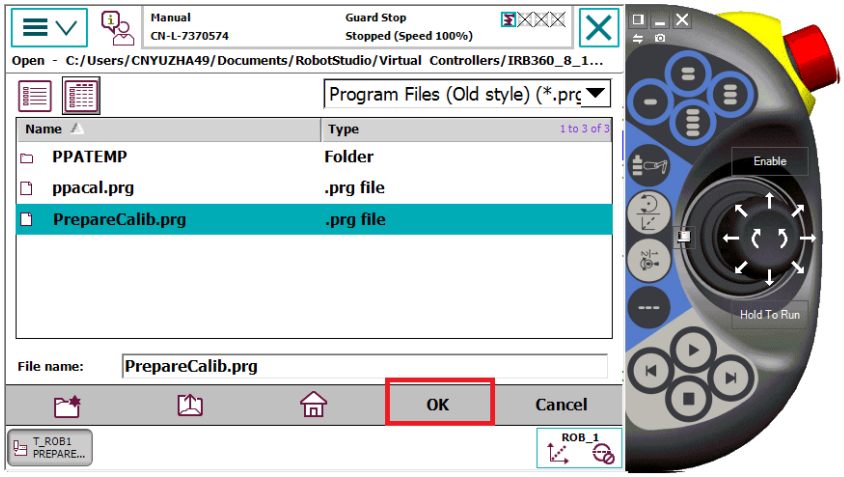


xx2200001930

- `PrepareCalib.prg` を選択し、**[OK]** をクリックします。

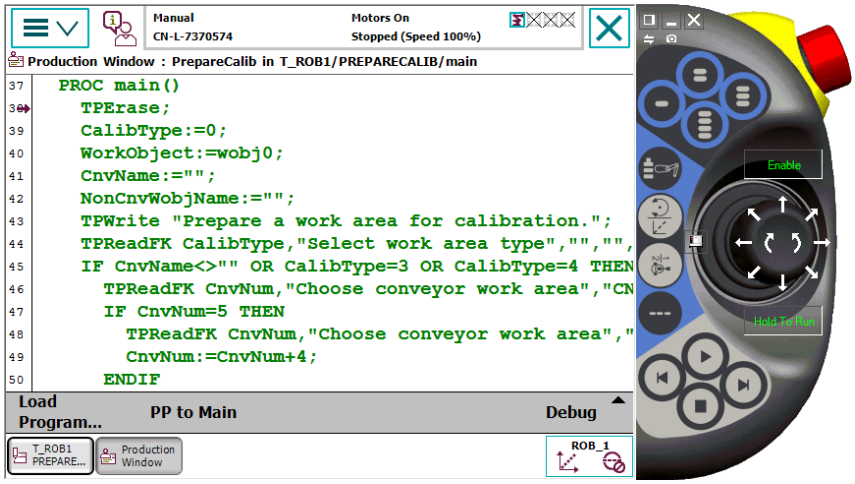
次のページに続く

© Copyright 2025 ABB. All rights reserved.



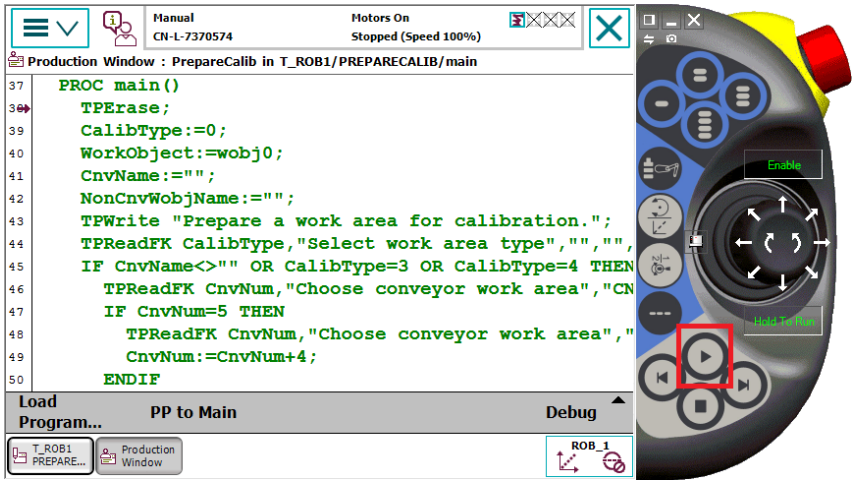
xx2200001931

- コントローラのモーターへの有効化デバイス ボタンを有効にします。



xx2200001933

- 再生 をクリックします。



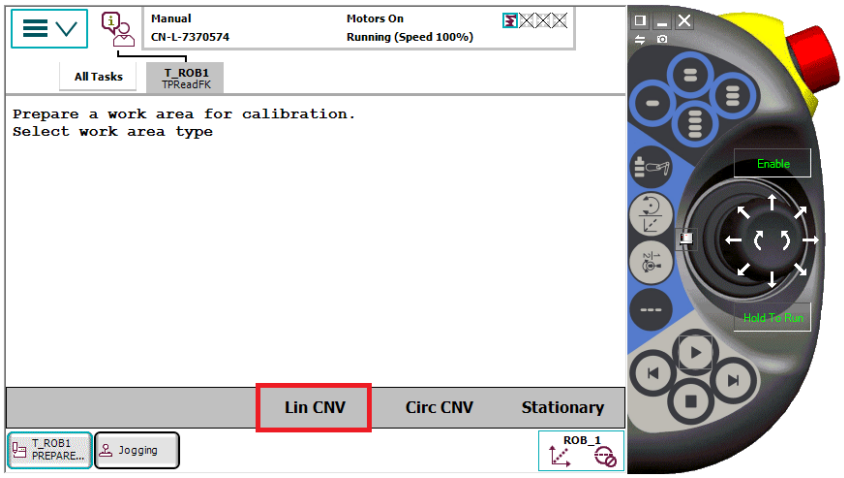
xx2200001934

- 作業エリアタイプ Lin CNV を選択します。

次のページに続く

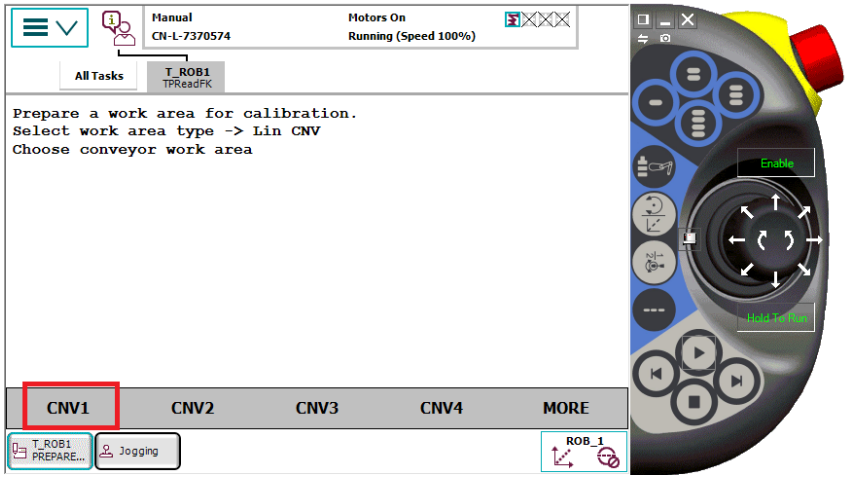
© Copyright 2025 ABB. All rights reserved.

4.4.4.1.3 ベース フレームの定義
続き



xx2200001943

- コンベアを選択します。例えば、CNV1。

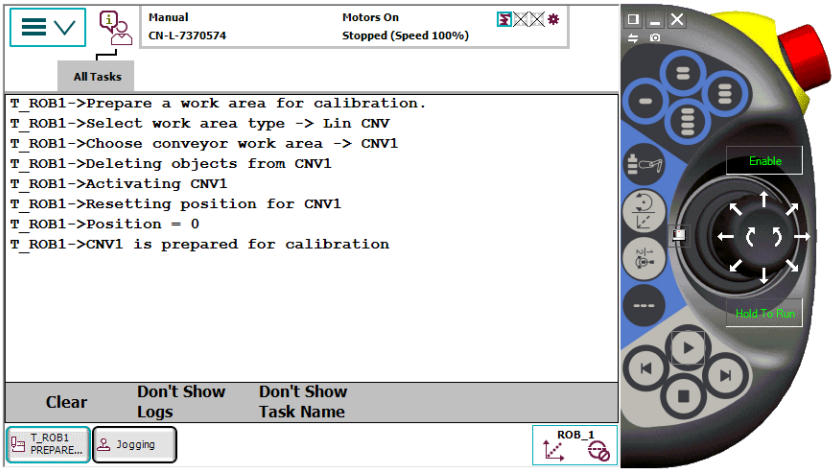


xx2200001944

- メッセージ...キャリブレーションの準備ができています を待ちます。CNV1 のジョギング ウィンドウのコンベアポジションは、「0」mm として表示されるはずです。

次のページに続く

© Copyright 2025 ABB. All rights reserved.



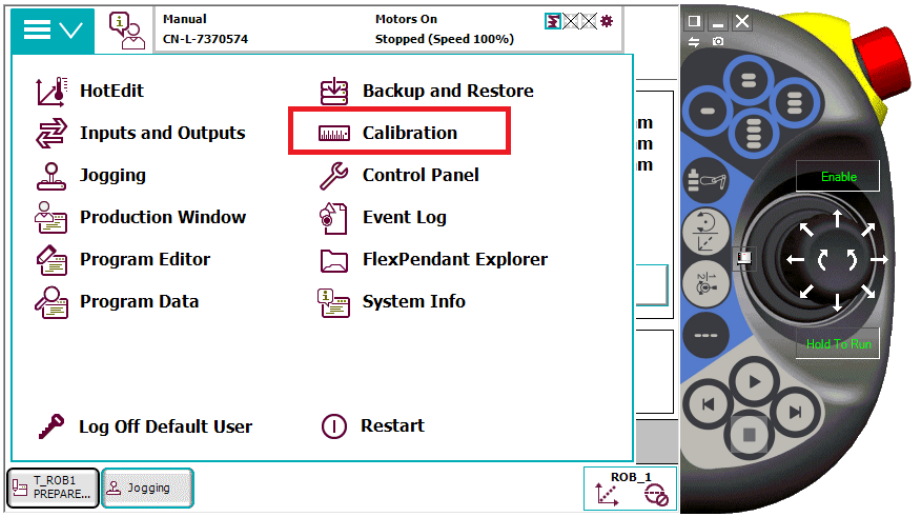
xx2200001945

3 コンベヤベルトを前方に動かして、基準点が次のキャリブレーション対象ロボットの動作範囲に入るようにします。

ジョグウィンドウで、すべてのコンベヤワークエリアのコンベヤポジションは、基準点について同じ総移動距離を示すはずです。カメラまたはセンサーに最も近いロボットが最初にキャリブレートされ、次に2番目に近いロボット、といった順序で、コンベヤ沿いのすべてのロボットがキャリブレートされます。

4 キャリブレーションツールをロボットに取り付けます。

5 FlexPendantでキャリブレーション ウィンドウを開きます。



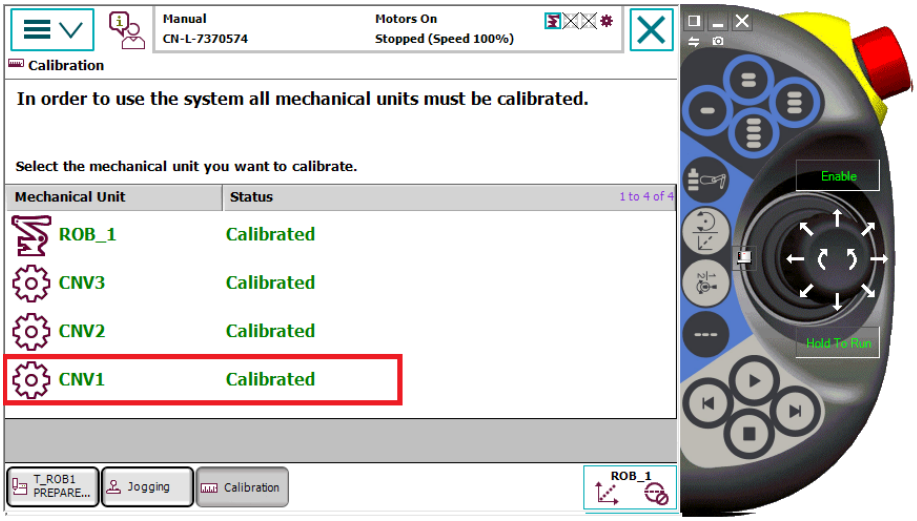
xx2200001946

6 コンベアを選択します。例えば、CNV1。

次のページに続く

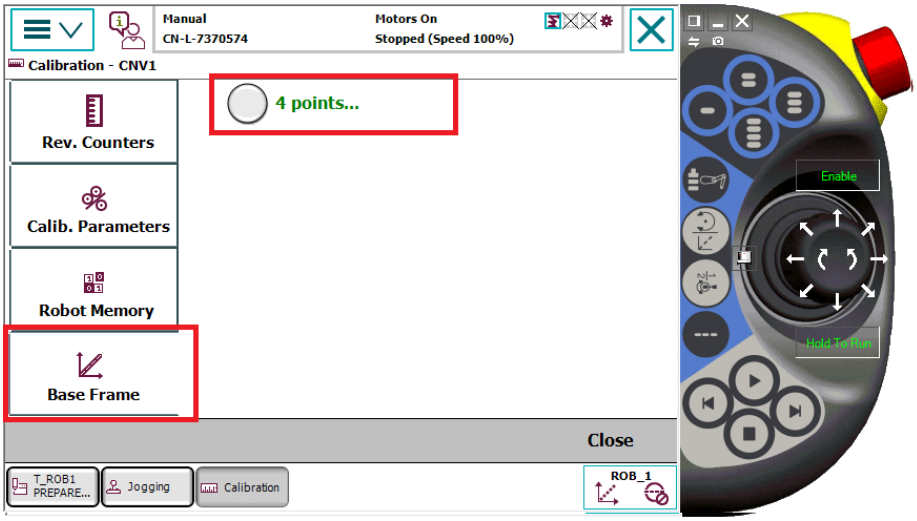
© Copyright 2025 ABB. All rights reserved.

4.4.4.1.3 ベース フレームの定義
続き



xx2200001947

7 ベース フレーム をタップし、4 ポイント を選択します。

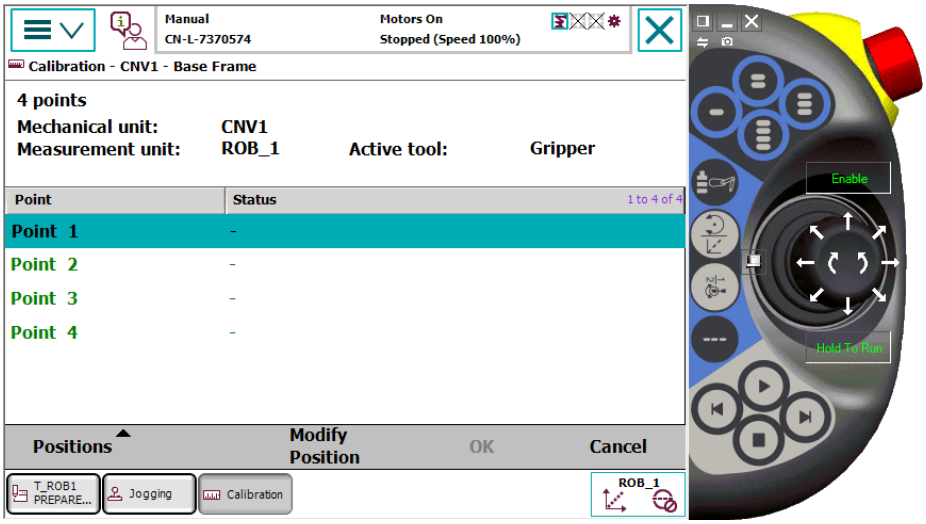


xx2200001948

8 最初のポイント[Point 1]を選択します。

次のページに続く

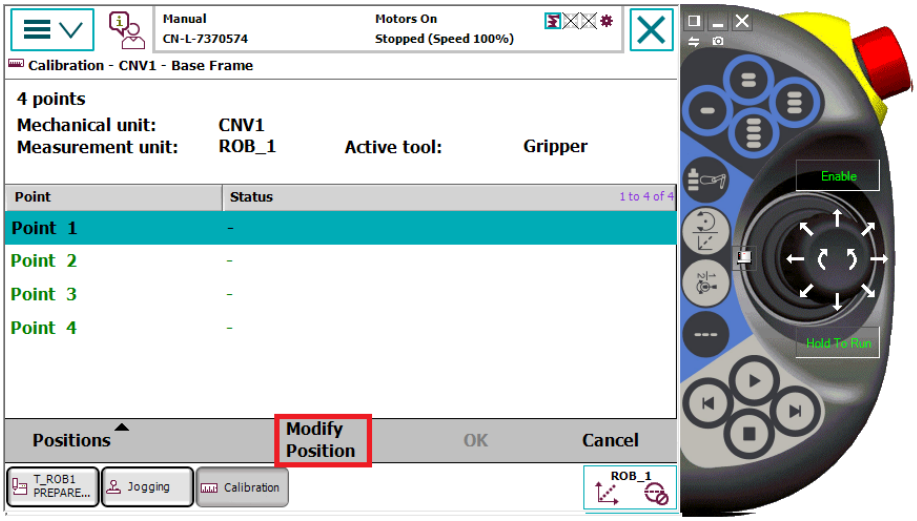
© Copyright 2025 ABB. All rights reserved.



xx2200001949

9 ロボットを手動でジョグまたは移動します。コンベヤ上の基準点をキャリブレーションツールTCPで正確にポイントします。

10 ポジションの変更 ファンクションキーをタップして、選択したポイント (ポイント 1) を変更します。



xx2200001950

11 ロボットが基準点に到達できる範囲の距離だけ、コンベヤベルトを前方に動かします。

4つのキャリブレーションポイント間（Point 1 ~ 4）の距離を長く均等に空けることをお勧めします。そうすることで、キャリブレーションの精度が向上するからです。

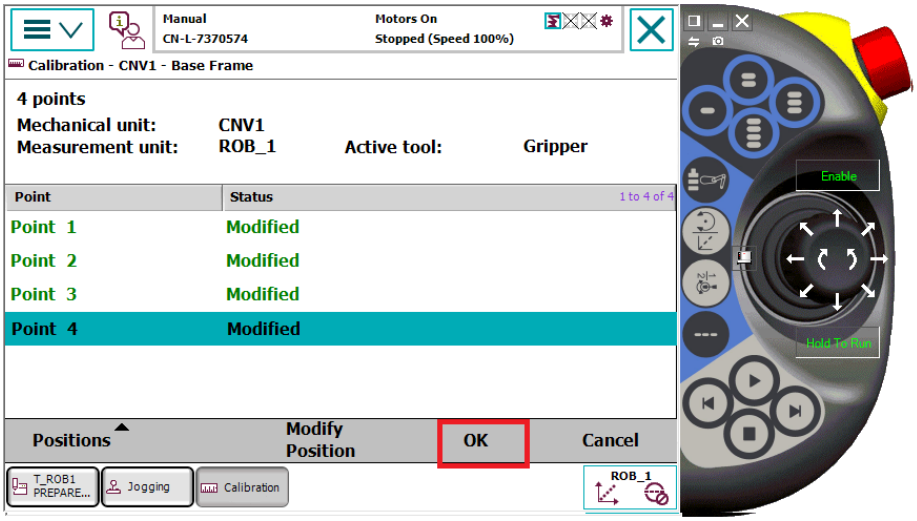
12 ポイントPoint 2、Point 3、Point 4についても、ステップ10 ~ 13を繰り返します。

13 [OK]をタップして、ベースフレームを計算します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.4.1.3 ベース フレームの定義
続き



xx2200001951

14 表示されたベースフレーム計算の平均誤差と最大誤差が許容範囲内であるかどうかを確認します。推定誤差が許容範囲内である場合は、[OK]をタップして、次のユーザーフレームの確認と保存を行います。

> **注記**
>
> ほとんどの場合、1 mm未満の平均誤差は許容されます。

推定誤差に問題がある場合、このベースフレームは再補正する必要があります。

- コンベヤベルトを後方に動かし、基準点がロボットの作業範囲に入るようにします。すべてのポイントPoint1、Point 2、Point 3、Point 4についてもステップ10 ~ 13を繰り返します。
- コンベヤを後方に動かせない場合は、手順1からやり直してください。

15 コンベヤに他にもキャリブレーション対象のロボットがある場合は、ステップ3から繰り返します。

16 次のベースフレームをアクティブにするには、コントローラを再起動します。

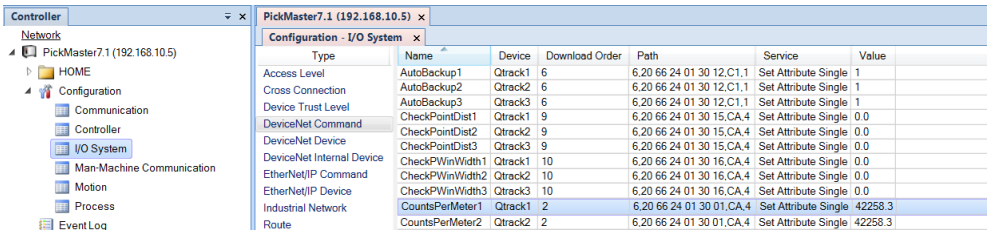
© Copyright 2025 ABB. All rights reserved.

4.4.4.2 DSQC 377 を使ったリニアコンベアのキャリブレーション

4.4.4.2.1 パラメータ「カウント/メートル」の定義

概要

*Counts Per Meter*システムパラメータは、コンベアエンコーダのキャリブレーションに使用します。*Counts Per Meter*システムパラメータは、トピック*I/O System*のタイプ*DeviceNet Command*に属します。



xx1900000551

Counts Per Meterの計算

Counts Per Meter システム パラメータの値は、次のように計算されます。

`position1/measured_meters`

値	説明
`position1`	FlexPendant または RobotStudioの事前定義された I/O シグナルから読み取ります。 たとえば、CNV1、シグナル名はc1counts です。
`measured_meters`(m)	コンベアが移動した手動測定距離 (メートル単位)。

Counts Per Meterの定義

以下の手順に従って、コンベヤエンコーダーの*Counts Per Meter*を定義します。

1. コンベヤベルトにマークを付けます。たとえば、線を引いたり、テープを貼り付けたりします。コンベアの固定側の同じポジションにもマークを付けます。
2. FlexPendant **プログラム エディタ** で、プログラム `ppacal.prg` をロードして実行します。

   これにより、コンベアの現在の位置がゼロに設定されます。値は、FlexPendant **入力と出力**-> **I/O デバイス**-> **RobICI**-> **DeviceNet** ウィンドウの位置部分に **CNV** 値として表示されます。
3. コンベヤベルトを約1メートル移動させます。
4. FlexPendant **入力と出力**-> **I/O デバイス**-> **RobICI**-> **DeviceNet** ウィンドウで、コンベアの c1counts を読み取ります。これは `position1` です。
5. 2つのマーク間の物理的な距離を測定します。これが`measured_meters`の値です。
6. 読み込んだ値と測定した値を使用して、*Counts Per Meter*を計算します。

   例: `20200/1.005 = 20099`
7. RobotStudioで、 [構成] を開き、トピック**I/O System**を選択し、**DeviceNet Command**を入力します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.4.2.1 パラメータ「カウント/メートル」の定義
続き

8 ユニット*Qtrackx*（x はコンベヤ番号）を選択し、パラメータ*Counts Per Meter*の値を更新します。
9 ［OK］をタッチします。
10 コントローラを再起動します。

関連情報

Application manual - Conveyor tracking。
『*Technical reference manual - System parameters*』。

© Copyright 2025 ABB. All rights reserved.

4.4.4.2.2 ベース フレームの定義

概要

コンベヤのコンベヤワークエリアごとに、コンベヤベースフレームのキャリブレーションを行います。ベースフレームのキャリブレーションを行うことで、ワークエリアでセンサー検出オブジェクトのピックまたはプレイスを行うときのロボットの基準点を求められます。

準備

- 各コンベア作業エリアのCounts Per Meter(メーターごとのカウント) システム パラメータを定義します。 詳細については、パラメータ「カウント/メートル」の定義 ページ 219, パラメータ「カウント/メートル」の定義 ページ 197を参照してください。
- ロボットに一時的に取り付けるキャリブレーションツールを準備します。このキャリブレーションツールにはポイントが付いたTCPがあるため、TCPオフセット値を正確に測定します。
- 各ロボットについて、RAPIDプログラムでキャリブレーションツールのツールデータを作成します。TCPオフセットを測定値で更新します。FlexPendant ジョグウィンドウで、ロボットのツールデータを選択します。
- カメラを使用する場合は、カメラをキャリブレーションします。カメラのキャリブレーション ページ 282を参照してください。カメラのキャリブレーション後、カメラキャリブレーションパターンをコンベアに添付します。

プロシージャ

次の手順を使用して、IRC5 コントローラを備えたラインのコンベアのすべてのベース フレームをキャリブレーションします。

1 コンベヤベルトにキャリブレーションの基準点が正確にマークされていることを確認してください。
   - カメラを使用する場合、基準点はカメラビューのローカル起点です。カメラのキャリブレーションが終了すると、コンベヤに添付されたカメラキャリブレーションパターンの起点によって基準点がすでにマークされています。
   - I/Oセンサーを使用して事前定義のポジションを生成する場合は、コンベヤ上で、センサーがオブジェクトを検出するポイントに基準点をマーク付けします。このポイントは、検出されたアイテムまたはコンテナのローカル起点になります。
2 コンベヤ（エンコーダーボード）ポジションをリセットします。

> **注記**
>
> このステップが完全に終了するまで、コンベヤを動かさないでください。

コンベヤに沿ってキャリブレートする必要のある、ワークエリアを持ったすべてのロボットについて、以下を行います。

- FlexPendant **プログラム エディタ** で、プログラム ppacal.prg をロードします。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.4.2.2 ベース フレームの定義
続き

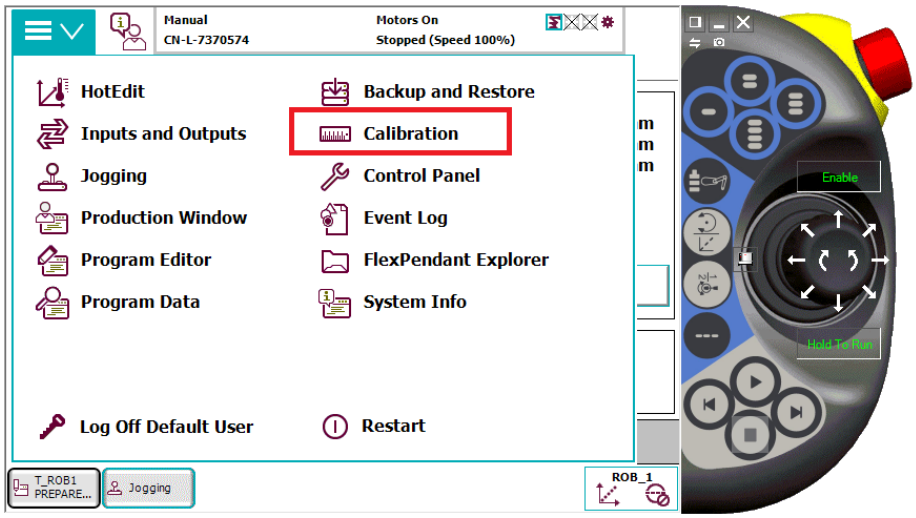
- ロードしたRAPIDを起動します
  - キャリブレーションタイプ、Conveyorを選択します。
  - コンベヤを選択します。例：CNV1
  - メッセージ**READY FOR CALIB**の表示を待ちます。ここで、CNV1のジョグウィンドウにコンベヤポジションが「0」mmと表示されます。

3 コンベヤベルトを前方に動かして、基準点が次のキャリブレーション対象ロボットの動作範囲に入るようにします。

ジョグウィンドウで、すべてのコンベヤワークエリアのコンベヤポジションは、基準点について同じ総移動距離を示すはずです。カメラまたはセンサーに最も近いロボットが最初にキャリブレートされ、次に2番目に近いロボット、といった順序で、コンベヤ沿いのすべてのロボットがキャリブレートされます。

4 キャリブレーションツールをロボットに取り付けます。

5 FlexPendantで**キャリブレーション** ウィンドウを開きます。

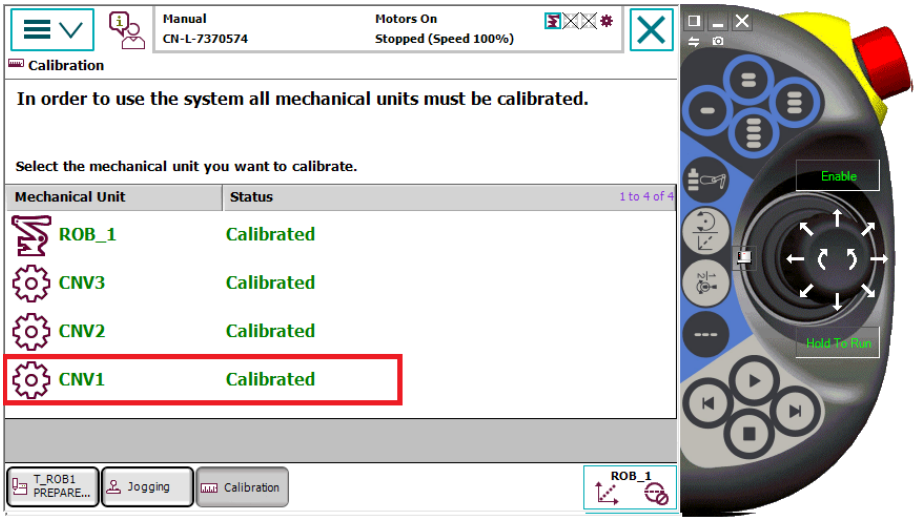


xx2200001946

6 コンベアを選択します。例えば、**CNV1**。

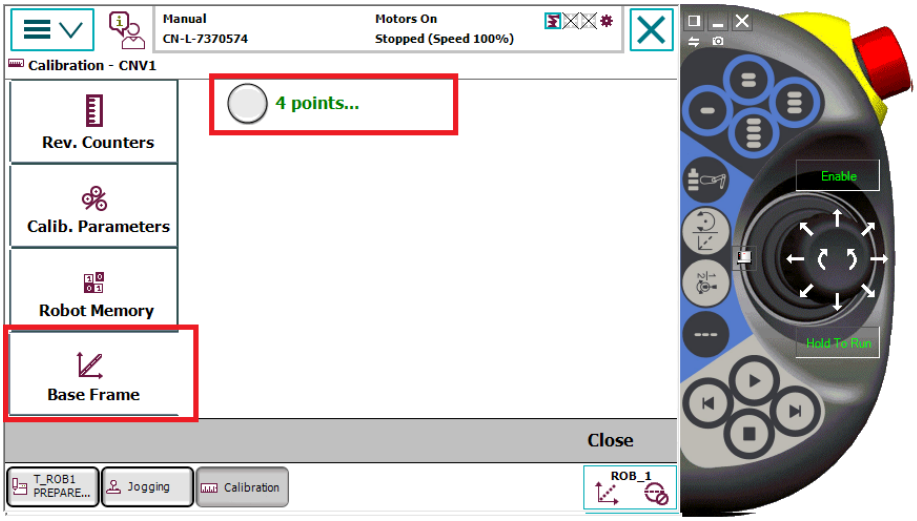
次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2200001947

7 ベース フレーム をタップし、4 ポイント を選択します。



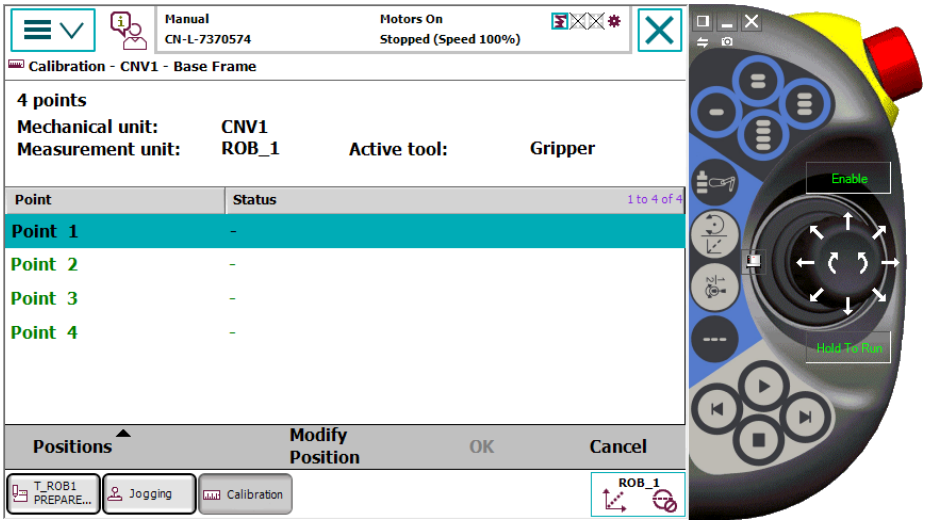
xx2200001948

8 最初のポイント[Point 1]を選択します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

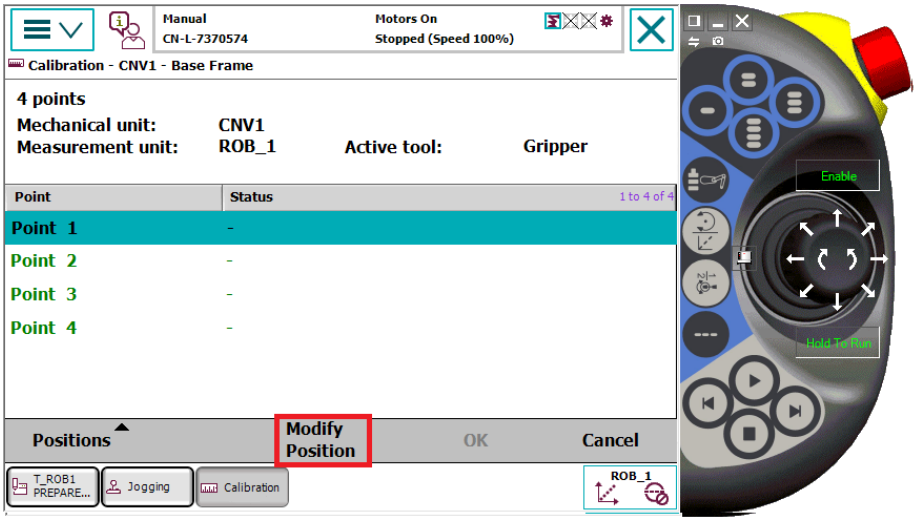
4.4.4.2.2 ベース フレームの定義
続き



xx2200001949

9 ロボットを手動でジョグまたは移動します。コンベヤ上の基準点をキャリブレーションツールTCPで正確にポイントします。

10 ポジションの変更 ファンクションキーをタップして、選択したポイント (ポイント 1) を変更します。



xx2200001950

11 ロボットが基準点に到達できる範囲の距離だけ、コンベヤベルトを前方に動かします。

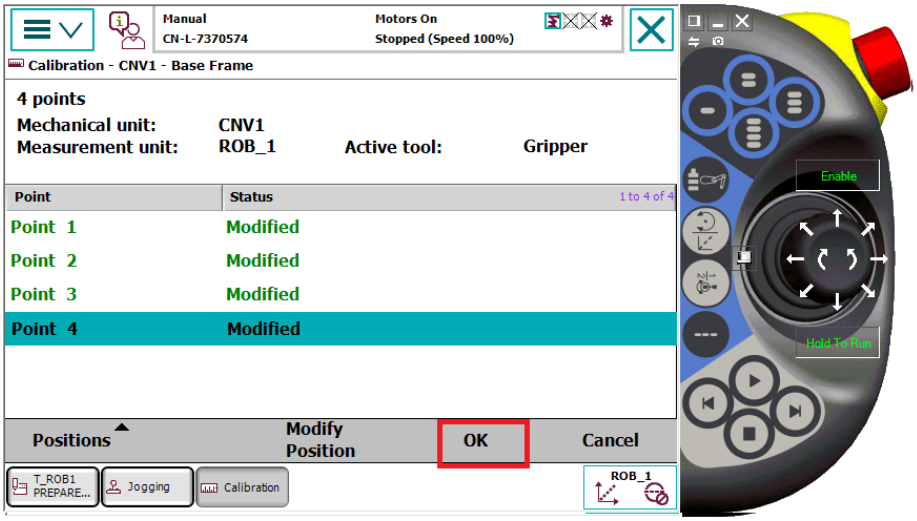
4つのキャリブレーションポイント間（Point 1 ~ 4）の距離を長く均等に空けることをお勧めします。そうすることで、キャリブレーションの精度が向上するからです。

12 ポイントPoint 2、Point 3、Point 4についても、ステップ10 ~ 13を繰り返します。

13 [OK]をタップして、ベースフレームを計算します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2200001951

14 表示されたベースフレーム計算の平均誤差と最大誤差が許容範囲内であるかどうかを確認します。推定誤差が許容範囲内である場合は、[OK]をタップして、次のユーザーフレームの確認と保存を行います。

> **注記**
>
> ほとんどの場合、1 mm未満の平均誤差は許容されます。

推定誤差に問題がある場合、このベースフレームは再補正する必要があります。

- コンベヤベルトを後方に動かし、基準点がロボットの作業範囲に入るようにします。すべてのポイントPoint1、Point 2、Point 3、Point 4についてもステップ10 ~ 13を繰り返します。
- コンベヤを後方に動かせない場合は、手順1からやり直してください。

15 コンベヤに他にもキャリブレーション対象のロボットがある場合は、ステップ3から繰り返します。

16 次のベースフレームをアクティブにするには、コントローラを再起動します。

© Copyright 2025 ABB. All rights reserved.

4.4.5 円形コンベアのキャリブレーション

概要

> **注記**
>
> 生産およびエミュレーションを実行する場合、次のキャリブレーション プロセスが必要です。 PickMaster PowerPac の [シミュレーション] タブでのキャリブレーションでは、次のキャリブレーション プロセスは完了しません。

円形コンベアに必要なキャリブレーションは、カメラと作業エリアのキャリブレーションです。作業エリアのキャリブレーションは、コンベア作業エリアではベースフレームのキャリブレーション、インデックスワークエリアでは作業物体の定義になります。ここで重要なのは、カメラ座標とロボットベースフレームまたは作業物体を同じ起点で定義することです。

カメラのキャリブレーションはカメラ毎に実行しなければなりません。また、ベースフレームのキャリブレーションはコンベヤ毎に実行する必要があります。

カメラのキャリブレーションはソリューションに保存されるため、そのソリューション内のすべてのレシピで同じキャリブレーションを共有できます。 カメラを再キャリブレーションする必要がある場合、ソリューション内のすべてのレシピが新しいキャリブレーションで更新されます。

カメラのキャリブレーションとベースフレームのキャリブレーションは別々に実行することができます。ただし、ワークエリアのキャリブレーション後に新しく正確なカメラのキャリブレーションを行うことは非常に困難です。

ワークエリアのキャリブレーションはロボットコントローラに保存されます。

円形コンベアをキャリブレーションするには、

1 必要に応じて、円形コンベアが時計回りに移動するようにエンコーダからDSQC 2000までのケーブル接続を変更します。エンコーダからさまざまなコンベア回転方向のDSQC 2000へのケーブル接続 ページ 228を参照してください。

> **注記**
>
> DSQC 2000へのデフォルトのハードウェア接続およびソフトウェア設定では、PickMaster PowerPacでの円形コンベアのデフォルトの回転方向として反時計回りに設定されています。
>
> ユーザーが円形コンベアをデフォルトの方向ではない時計回りに回転させたい場合は、ハードウェア接続を変更することが前提条件となります。

> **ヒント**
>
> 円形コンベアの時計回り回転は、DSQC 377の検証はされていません。

2 パラメータ*Counts Per Meter*を定義します（コンベヤのみを使用する場合）。パラメータ「カウント/メートル」の定義 ページ 230を参照します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

注記

円形コンベアでは、パラメータ *Counts Per Meter* はラジアンあたりのカウントを示します。

3 ベースフレームを定義します。ベース フレームの定義 ページ *232*、ベース フレームの定義 ページ *259*を参照してください。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.5.1 DSQC 2000 (CTM)による円形コンベアのキャリブレーション

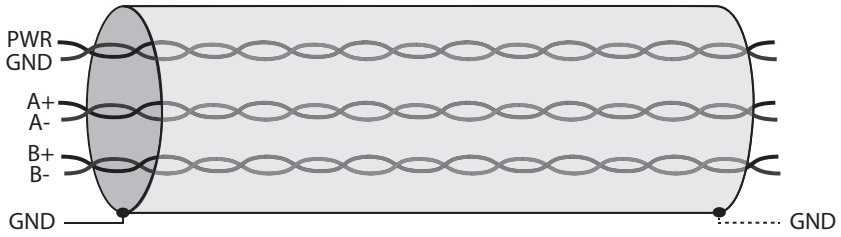
4.4.5.1.1 エンコーダからさまざまなコンベア回転方向のDSQC 2000へのケーブル接続

概要

PickMaster PowerPacでは、反時計回りの回転は、デフォルトのハードウェア接続とソフトウェア設定での円形コンベアのデフォルトの移動方向に設定されます。詳細については、*Application manual - Conveyor tracking*を参照してください。

ユーザーが円形コンベアをデフォルトの方向ではない時計回りに回転するように設定する必要がある場合は、ハードウェア接続を変更することが前提条件となります。

エンコーダは、DSQC 2000のエンコーダ インターフェイスに接続されます。ケーブルの図は以下のとおりです。

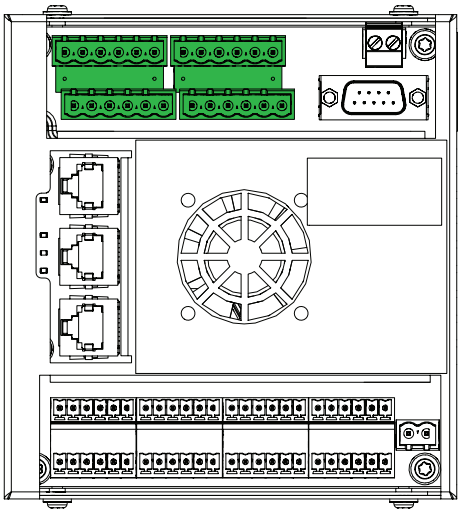


xx1800001539

PWR	電力
GND	アース
A+, A-	エンコーダ信号 A (0°)
B+, B-	エンコーダ信号 B (90°)

CTMのエンコーダ接続

CTM には、PNP、プッシュプル、および NPN エンコーダをサポートする 4 つの独立したエンコーダ インターフェイスがあります。

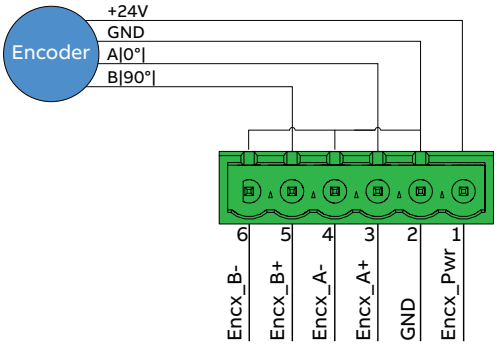
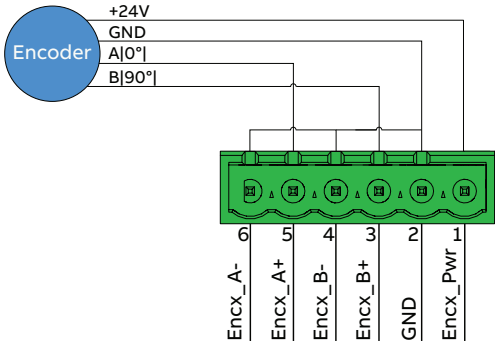
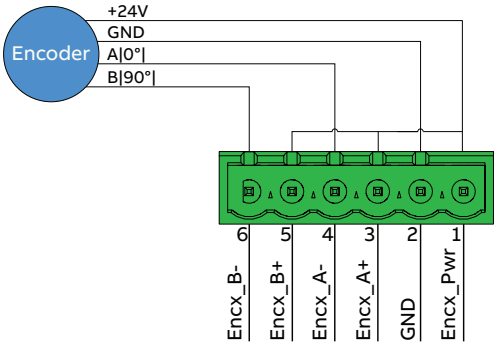
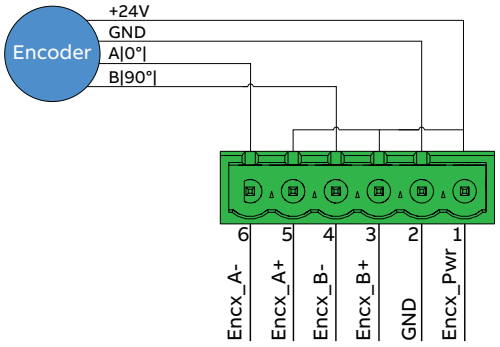
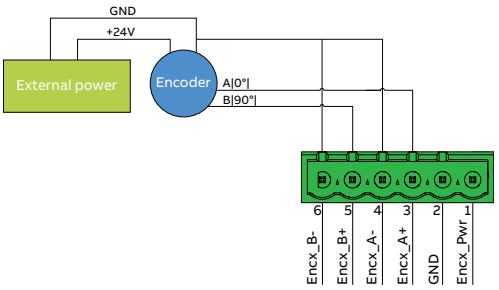
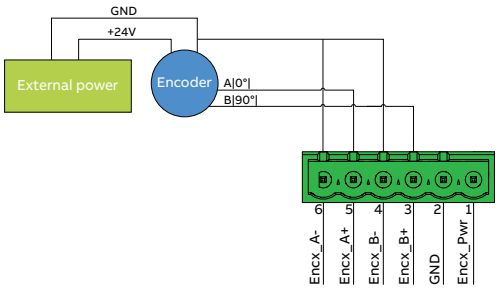


xx2400000415

次のページに続く

© Copyright 2025 ABB. All rights reserved.

エンコーダから CTM までのケーブル接続は、コンベアの回転方向に応じて下表のように異なります。

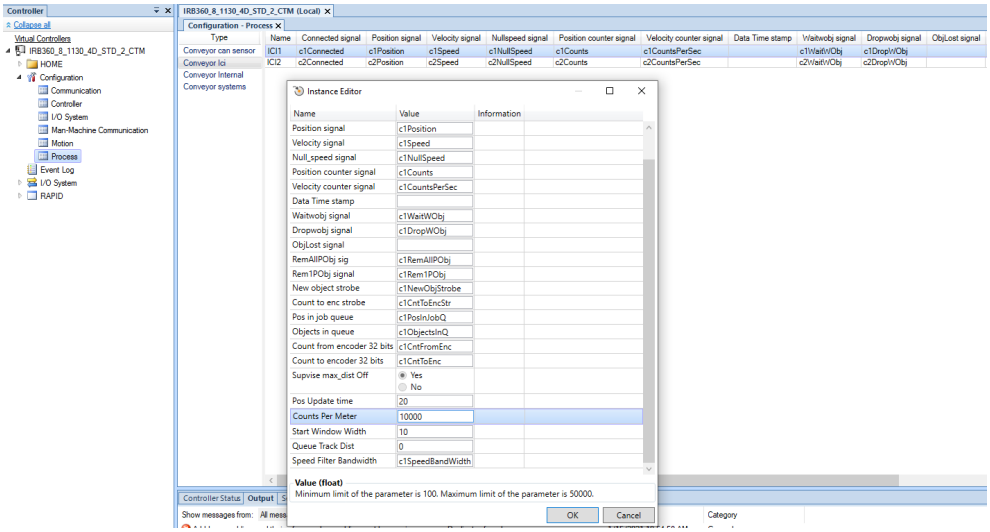
	反時計回りの回転	時計回りの回転
PNP & プッシュ/プル型エンコーダ	 xx2400000416	 xx2400000417
NPN型エンコーダ	 xx2400000418	 xx2400000419
エンコーダは外部から電源を供給されます	 xx2400000420	 xx2400000421

© Copyright 2025 ABB. All rights reserved.

4.4.5.1.2 パラメータ「カウント/メートル」の定義

概要

*Counts Per Meter*システムパラメータは、コンベアエンコーダのキャリブレーションに使用します。*Counts Per Meter*システムパラメータは、トピック*Processのタイプ*Conveyor Iciに属します。



xx2100000042

> **ヒント**
>
> 円形コンベアの場合、パラメータ*Counts Per Meter*は 1 度あたりのカウントを示します。

Counts Per Meterの計算

Counts Per Meter の値 システム パラメータは次のように計算されます。

```
counts value/measured_radians
```

値	説明
`counts value` (`position1-position0`)	FlexPendant または RobotStudioの事前定義された I/O シグナルから読み取ります。 たとえば、CNV1、シグナル名はc1counts です。
`measured_radians`(ラジアン)	コンベアが移動したことを手動で測定したラジアン。

Counts Per Meterの定義

以下の手順に従って、コンベヤエンコーダーの*Counts Per Meter*を定義します。

1. コンベアベルトに線を引く、テープを貼るなどのマークを付け、コンベアの外側の同じ場所にマークを付けます。
2. OmniCore の場合、FlexPendant **I/O-> RobICI-> CTM** ウィンドウで、コンベアの c1countsを読み取ります。これは`position0` です。

この変数が円形コンベアに適用される場合、実際の意味はラジアンあたりのカウントです。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

IRC5 の場合、FlexPendant 入力および出力-> I/O デバイス-> RobICI-> CTM ウィンドウで、コンベアの c1countsを読み取ります。これはposition0 です。

3 コンベアベルトを約360度回転させます。

4 FlexPendant ジョギング ウィンドウで、コンベアの c1countsを読み取ります。これはposition1 です。

5 コンベアの物理ラジアンを測定します。これは、measured_radiansの値です。

6 counts value= position1- position0 を計算します。

7 counts valueと測定値を使用して、*Counts Per Meter*を計算します。

例: 60000/1 = 60000

8 RobotStudioで、［構成］ を開き、トピックProcessを選択し、Conveyor Ici を入力します。

9 ユニット*ICIx*（x はコンベア番号）を選択し、パラメータ*Counts Per Meter*の値を編集します。

10 ［OK］をタッチします。

11 コントローラを再起動します。

関連情報

Application manual - Conveyor tracking。

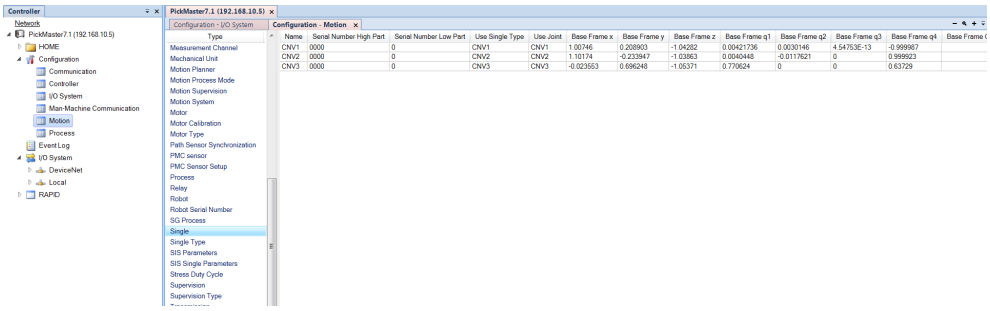
『*Technical reference manual - System parameters*』。

© Copyright 2025 ABB. All rights reserved.

4.4.5.1.3 ベース フレームの定義

概要

円形コンベアのコンベア作業エリアごとに、コンベアベースフレームのキャリブレーションを行います。ベースフレームのキャリブレーションを行うことで、作業エリアでセンサー検出物体のピックまたは配置を行うときのロボットの基準点を求められます。



xx1900000592

準備

- 各コンベア作業エリアのCounts Per Meter(メーターごとのカウント) システム パラメータを定義します。 詳細については、パラメータ「カウント/メートル」の定義 ページ 257, パラメータ「カウント/メートル」の定義 ページ 230を参照してください。
- ロボットに一時的に取り付けるキャリブレーションツールを準備します。このキャリブレーションツールにはポイントが付いたTCPがあるため、TCPオフセット値を正確に測定します。
- 各ロボットについて、RAPIDプログラムでキャリブレーションツールのツールデータを作成します。TCPオフセットを測定値で更新します。FlexPendant ジョグウィンドウで、ロボットのツールデータを選択します。
- カメラを使用する場合は、カメラをキャリブレーションします。カメラのキャリブレーション ページ 282 を参照してください。カメラのキャリブレーション後、カメラキャリブレーションパターンをコンベアに添付します。

Recommendation

このセクションでは、TCP 測定と RAPID プログラムを使用して、円形コンベアのコンベア ベース フレームポジションと四元数を計算する方法について説明します。

この方法では、円形コンベア上の 3 つの測定ポイントを使用して、回転の中心を計算します。 3 つのポイントは、周囲でできるだけ離して配置する必要があります。

OmniCore の手順

以下の手順を使用して、OmniCore コントローラを備えた円形コンベアのすべてのベース フレームをキャリブレーションします。

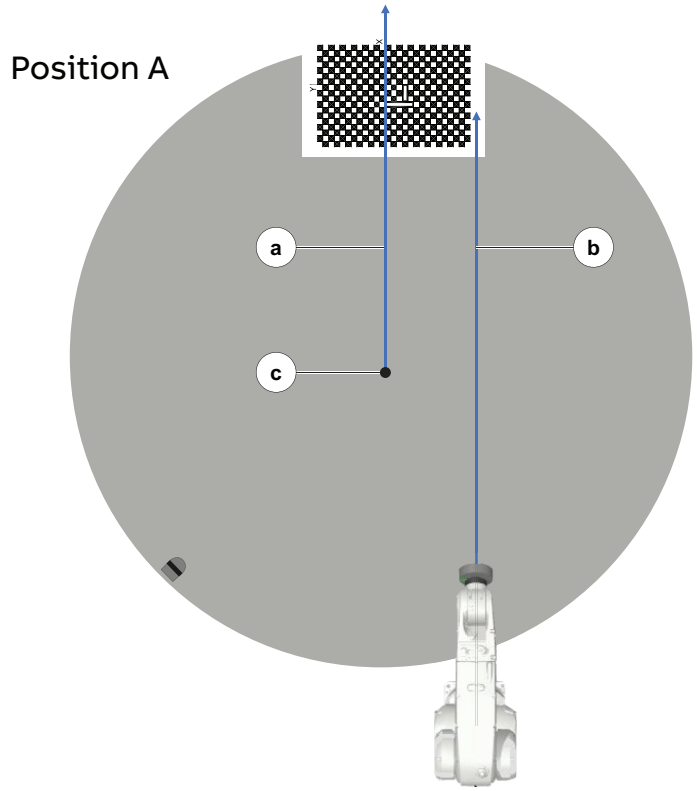
1 キャリブレーションツールをロボットに取り付けます。
2 キャリブレーション グリッドを中心線 (a) に X 揃えで配置します。
グリッド X が円形コンベアの外側を向いていることを確認してください。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

3 ベルトをポジション A まで回転させ、中心線がキャリブレーションロボットの X 軸 (b) と平行になります。

中心線は、円形コンベアの中心ポイント(c)と目盛方眼紙のX軸を結ぶ線です。



xx2200002007

4 コンベア（エンコーダボード）のポジションをAのポジションにリセットします。

注記

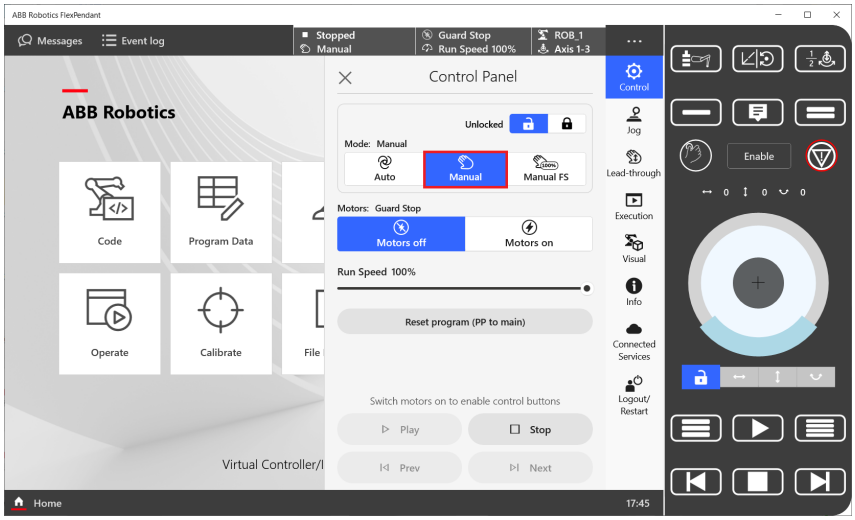
このステップが完全に終了するまで、コンベヤを動かさないでください。

A コントローラを手動モードに設定します。

次のページに続く

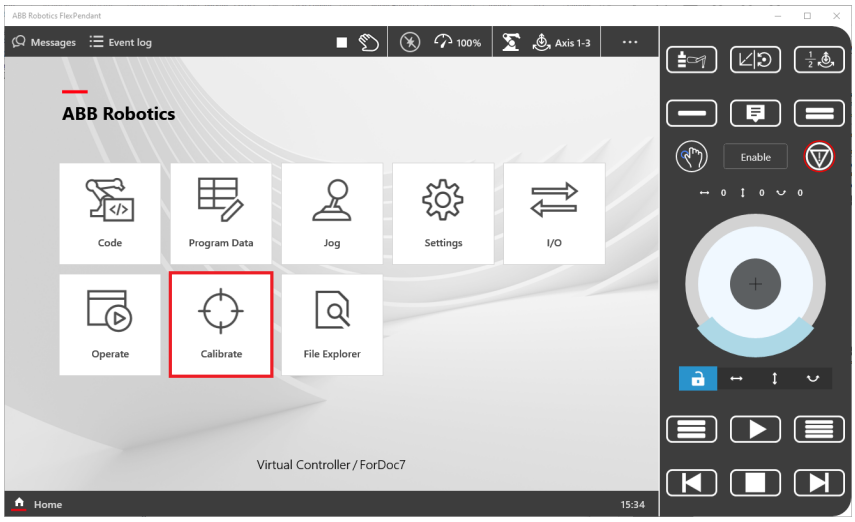
© Copyright 2025 ABB. All rights reserved.

4.4.5.1.3 ベース フレームの定義
続き



xx2100000368

B FlexPendant で、**キャリブレート** をクリックします。

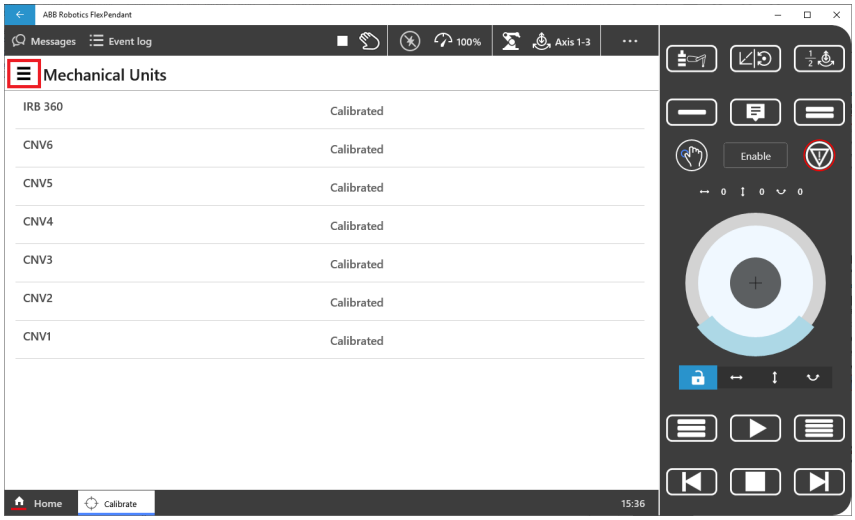


xx2100000362

C 左上隅の **オプション タブ** をクリックします。

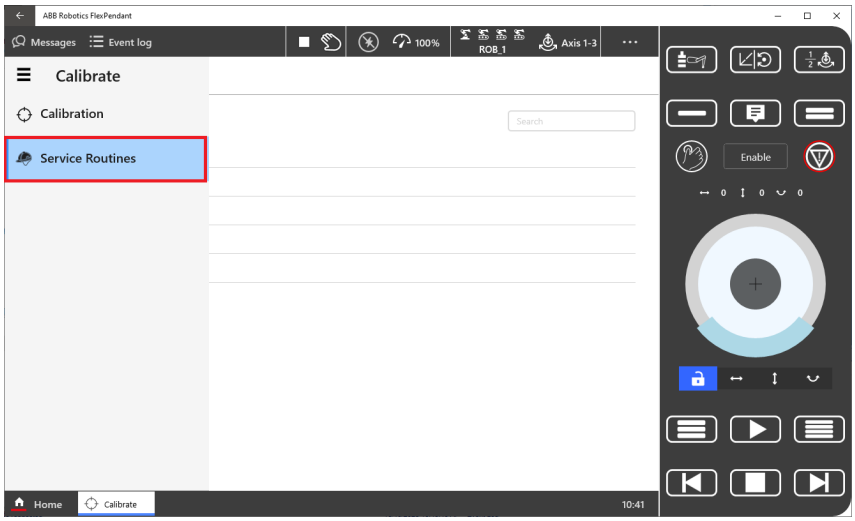
次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2100000363

D サービス ルーチン をクリックします。



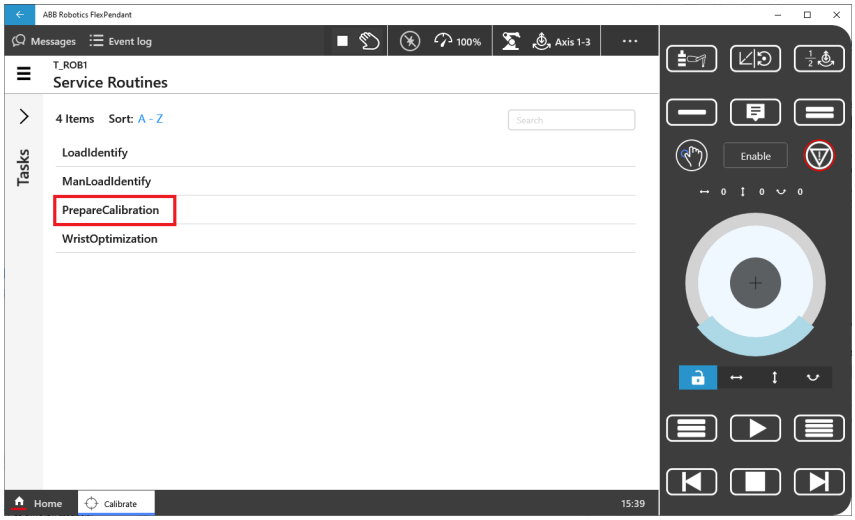
xx2100000364

E キャリブレーションの準備 をクリックします。

次のページに続く

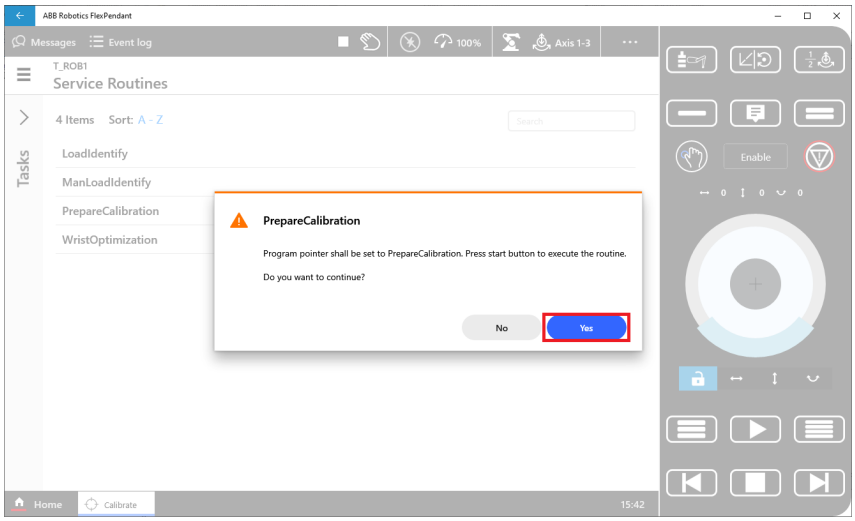
© Copyright 2025 ABB. All rights reserved.

4.4.5.1.3 ベース フレームの定義
続き



xx2100000365

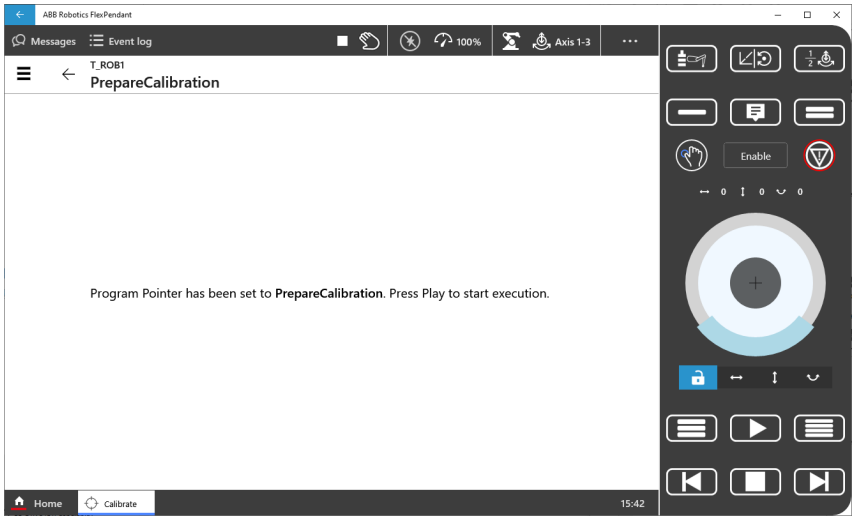
F 表示されたダイアログで はい をクリックします。



xx2100000366

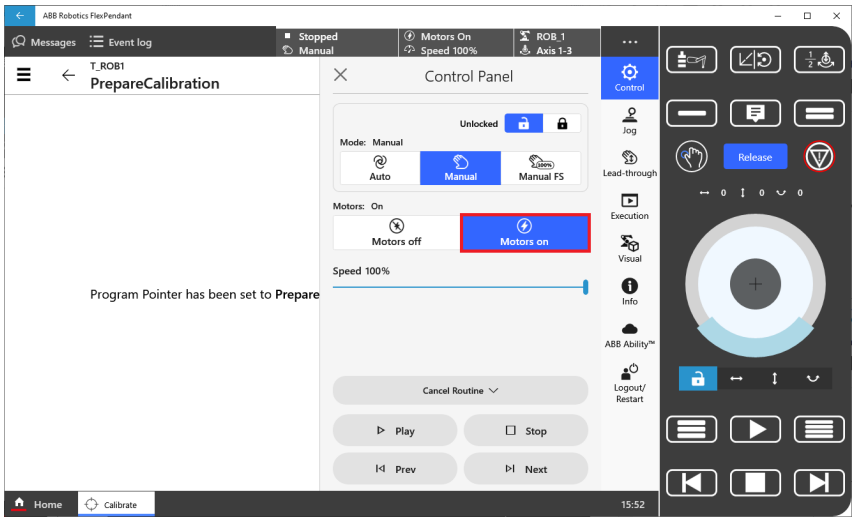
次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2100000367

G コントローラのモーターへの有効化デバイス ボタンを有効にします。



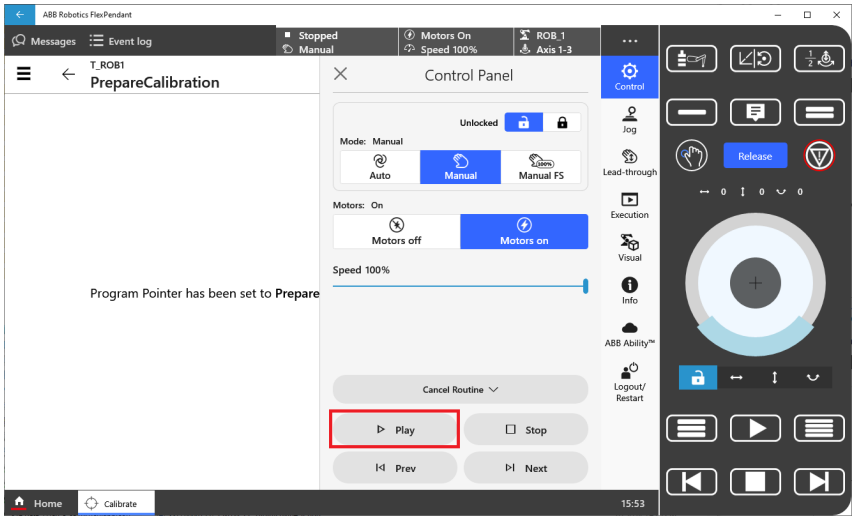
xx2100000369

H 再生 をクリックします。

次のページに続く

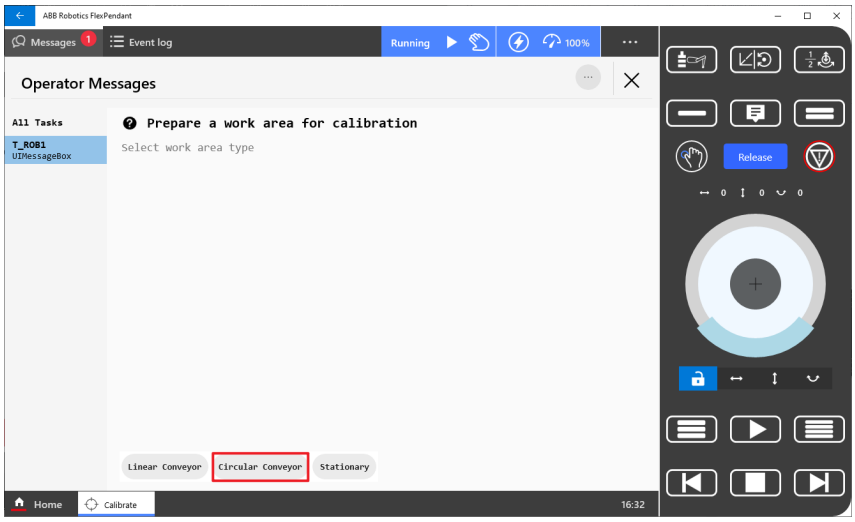
© Copyright 2025 ABB. All rights reserved.

4.4.5.1.3 ベース フレームの定義
続き



xx2100000370

I 作業エリアの種類として円形コンベアを選択します。

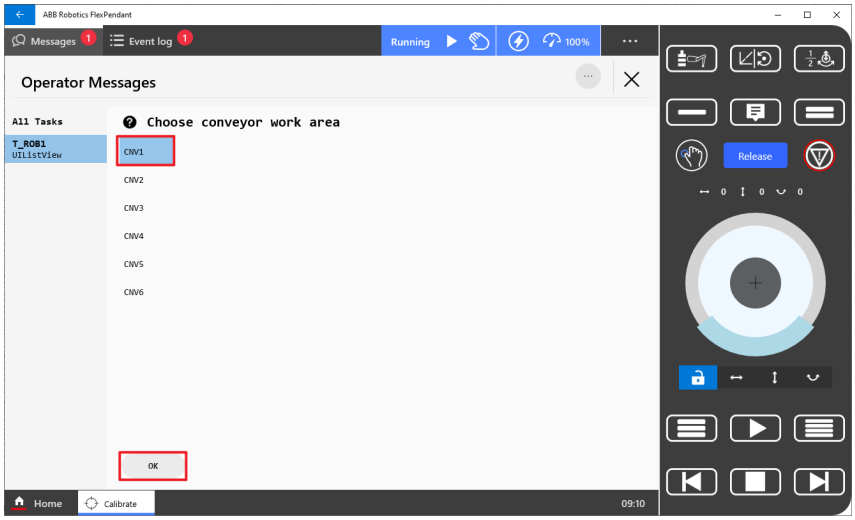


xx2100000690

J コンベアを選択します:例えば、CNV1。次に、**OK**をクリックします。

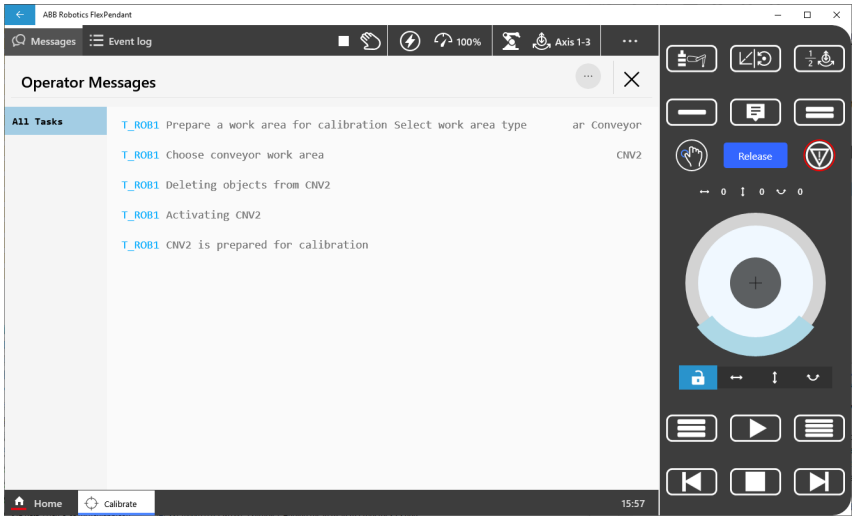
次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2100000691

K メッセージ...キャリブレーションの準備ができています を待ちます。CNV1 のジョギング ウィンドウのコンベアポジションは、「0」mm として表示されるはずです。



xx2100000395

5 コンベアに沿って調整する必要があるワークエリアを持つロボットに対して、次の手順を実行します。

A 回転を選択してくださいというメッセージが表示されたら、クリックしてコンベアの移動方向を選択します。

反時計回りの移動方向の場合は、反時計回りを選択します。

次のページに続く

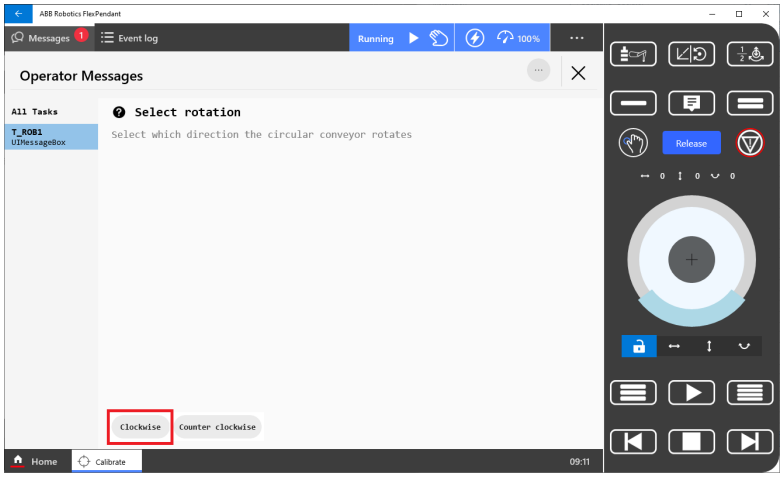
© Copyright 2025 ABB. All rights reserved.

4.4.5.1.3 ベース フレームの定義
続き



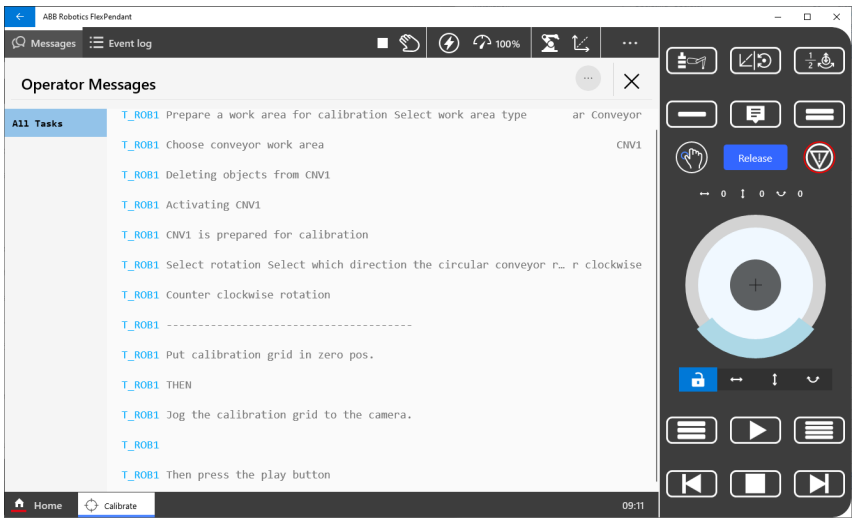
xx2100000692

時計回りの移動方向には、時計回りを選択します。



xx2400000430

B プログラムは自動的に続行されます。



xx2100000693

次のページに続く

© Copyright 2025 ABB. All rights reserved.

C ベルトをポジション B (基準ポイント) まで回転させ、カメラ (d)の下にキャリブレーション グリッドを作成します。

反時計回りを選択した場合、ベルトを反時計回りに回転させます。

時計回りを選択した場合、ベルトを時計回りに回転させます。

ヒント

コンベアベルトを手で基準ポイントまで回転させます。

- ソース タイプが [事前定義] で、トリガー タイプが [距離] の場合、基準ポイントはHotspotの原点になります。
- ソース タイプが [事前定義] で、[I/O センサー、Hotspotとしてトリガー] が事前定義されたアイテムまたはコンテナの生成に使用される場合、基準ポイントはオブジェクトがセンサーによって検出されるポイントにある必要があります。このポイントが、検出されたアイテムまたはコンテナのローカル原点になります。
- ソース タイプがビジョンの場合、基準ポイントはカメラ ビューのローカル原点になります。

  カメラを使用する場合は、同時にカメラを調整します。カメラのキャリブレーション ページ 282を参照してください。

  カメラが調整されたばかりの場合、基準ポイントはコンベアに取り付けられたカメラ調整パターンの原点によってすでにマークされています。

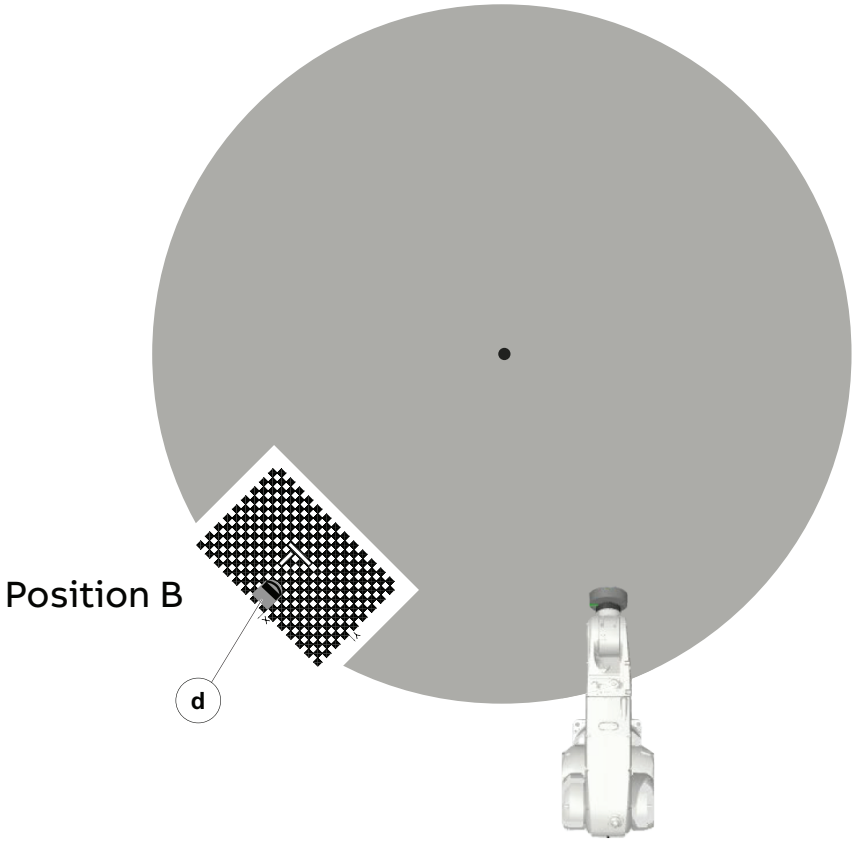
注記

ベルトを前後に回転させないでください。これにより、不正確な調整データが生成されます。

次のページに続く

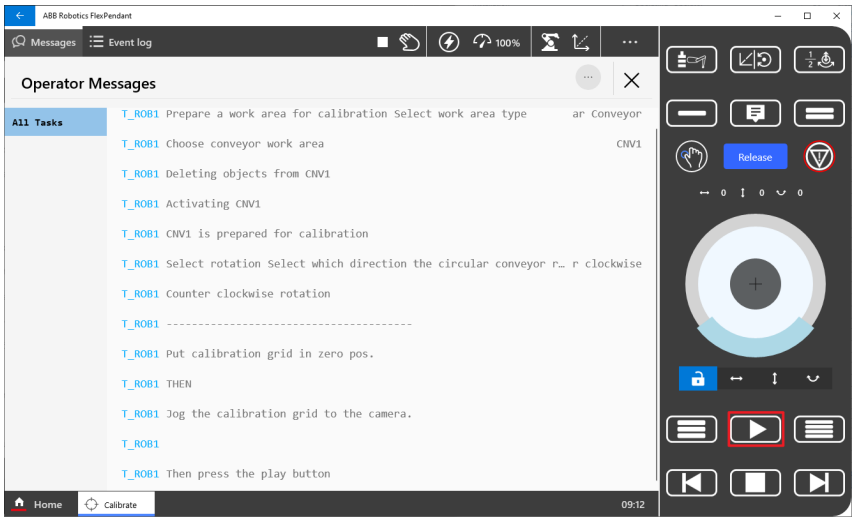
© Copyright 2025 ABB. All rights reserved.

4.4.5.1.3 ベース フレームの定義
続き



xx2200002008

D リセットされたロボットの FlexPendant で再生 をクリックします。



xx2100000694

E ロボットが基準点に到達できる範囲の距離だけ、コンベヤベルトを前方に動かします。

3つのキャリブレーションポイント間（Point 1 ~ 3）の距離を長く均等に空けることをお勧めします。そうすることで、キャリブレーションの精度が向上するからです。

反時計回りを選択した場合、ベルトを反時計回りに回転させます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

時計回りを選択した場合、ベルトを時計回りに回転させます。

> **注記**
>
> ベルトを前後に回転させないでください。これにより、不正確な調整データが生成されます。

F ロボットを手動でジョグまたは移動します。コンベヤ上の基準点をキャリブレーションツールTCPで正確にポイントします。

G 再生 をタップしてポイント (ポジション 1) を変更します。

H ポジション 2とポジション 3に対して F から H までの手順を繰り返します。

I ベース フレーム計算の表示された平均誤差と最大誤差が許容範囲内かどうかを確認します。 推定誤差が許容できる場合は、システムを再起動して、新しいベース フレームを確認して保存します。

> **注記**
>
> ほとんどの場合、1 mm未満の平均誤差は許容されます。



xx2100000695

推定誤差に問題がある場合、このベースフレームは再補正する必要があります。

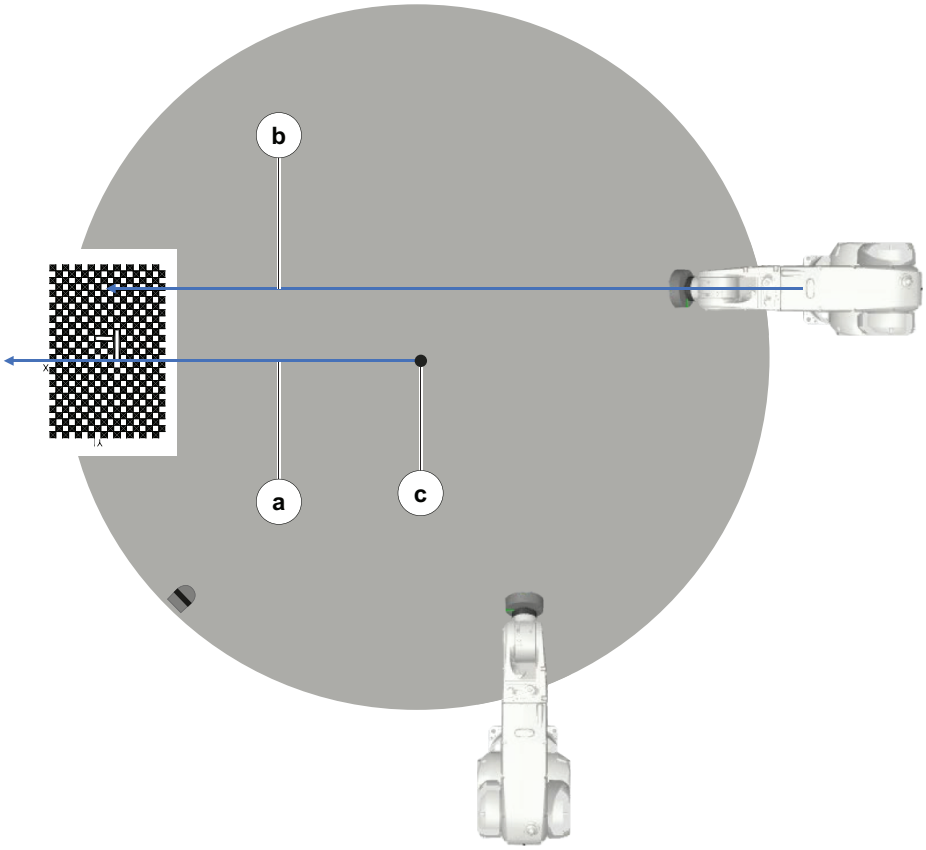
> **ヒント**
>
> **カメラ センターからのオフセット**の値を読み取ります。 この値は、円形コンベアのタイプ設定 ページ 263の`Sensor offset`の入力として使用されます。

6 キャリブレーションが必要なロボットが他にもある場合は、ロボットごとに手順 3 から手順 5 を繰り返します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.5.1.3 ベース フレームの定義
続き



xx2400000405

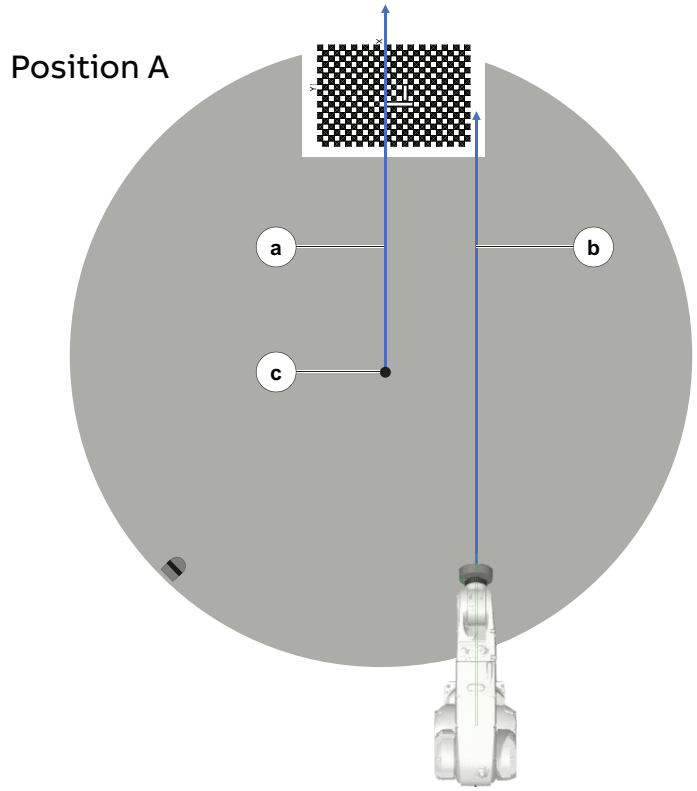
IRC5 の手順

以下の手順を使用して、IRC5 コントローラを備えた円形コンベアのすべてのベースフレームをキャリブレーションします。

1. キャリブレーションツールをロボットに取り付けます。
2. キャリブレーション グリッドを中心線 (a) に X 揃えで配置します。
   グリッド X が円形コンベアの外側を向いていることを確認してください。
3. ベルトをポジション A まで回転させ、中心線がキャリブレーションロボットの X 軸 (b) と平行になります。
   中心線は、円形コンベアの中心ポイント(c)と目盛方眼紙のX軸を結ぶ線です。

次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2200002007

4 コンベア（エンコーダボード）のポジションをAのポジションにリセットします。

注記

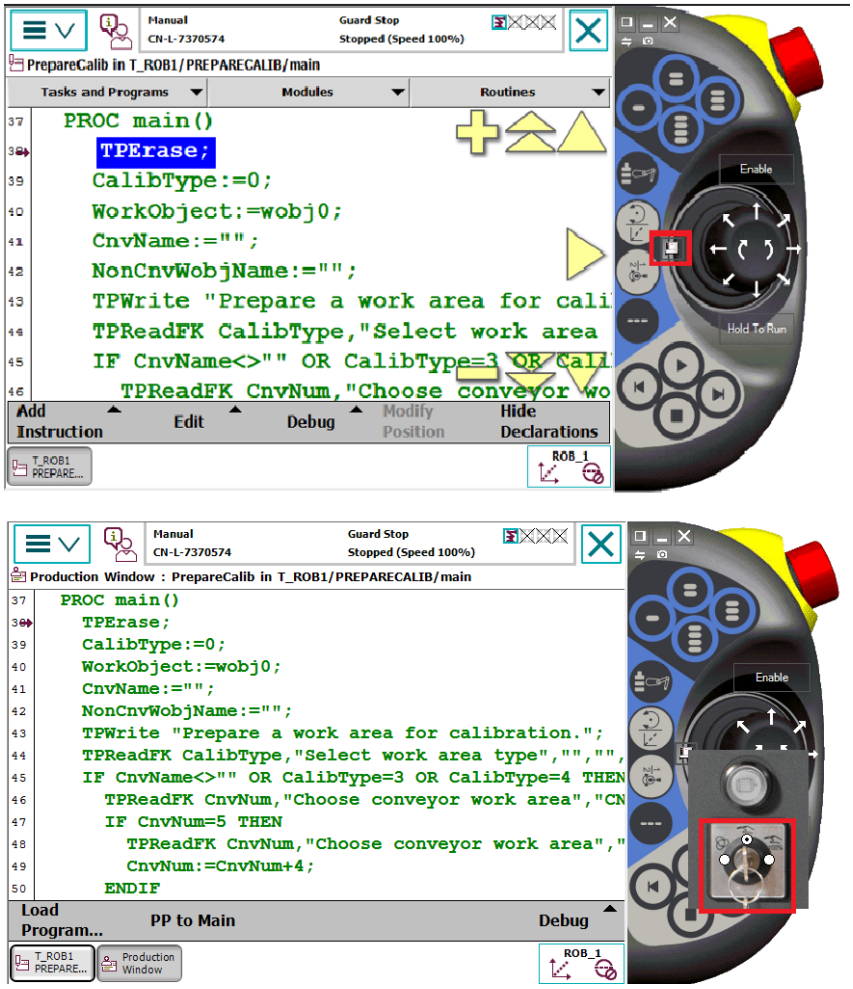
このステップが完全に終了するまで、コンベヤを動かさないでください。

A コントローラを手動モードに設定します。

次のページに続く

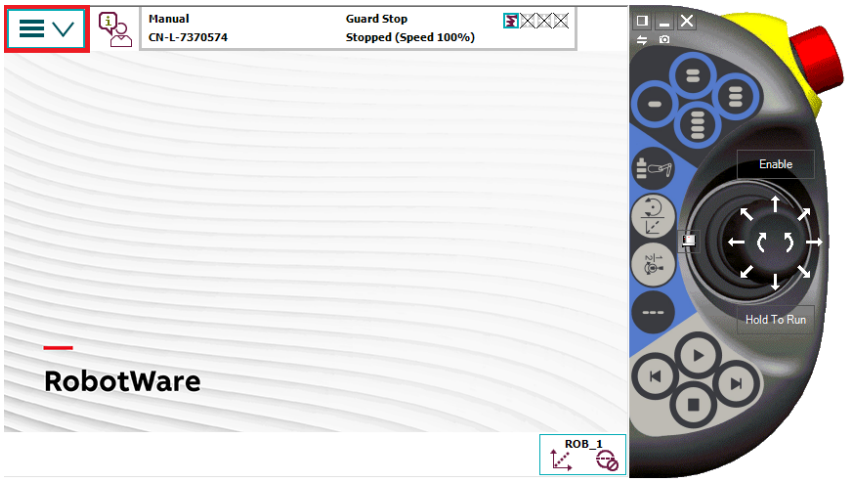
© Copyright 2025 ABB. All rights reserved.

4.4.5.1.3 ベース フレームの定義
続き



xx2200001932

B FlexPendant で、**メニュー** をクリックしてドロップダウン リストを開きます。

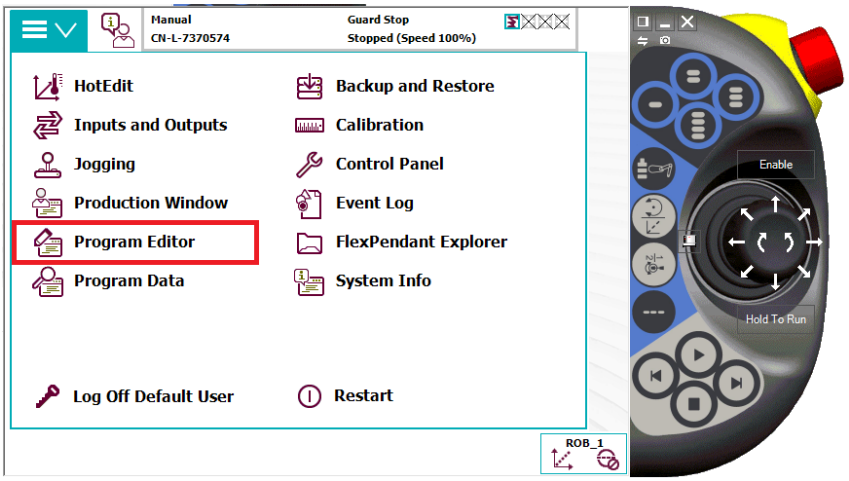


xx2200001925

C ドロップダウン リストで **プログラム エディタ** をクリックします。

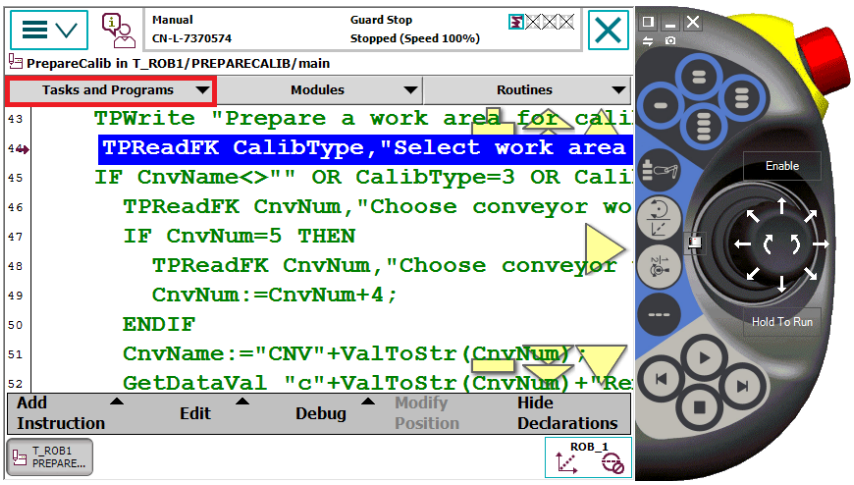
次のページに続く

© Copyright 2025 ABB. All rights reserved.



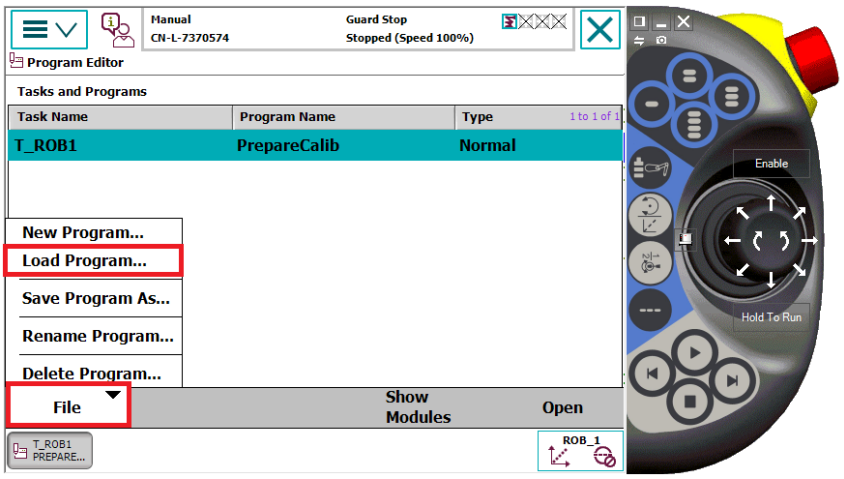
xx2200001926

D タスクとプログラム をクリックします。



xx2200001927

E ファイル とプログラムの読み込み をクリックします。



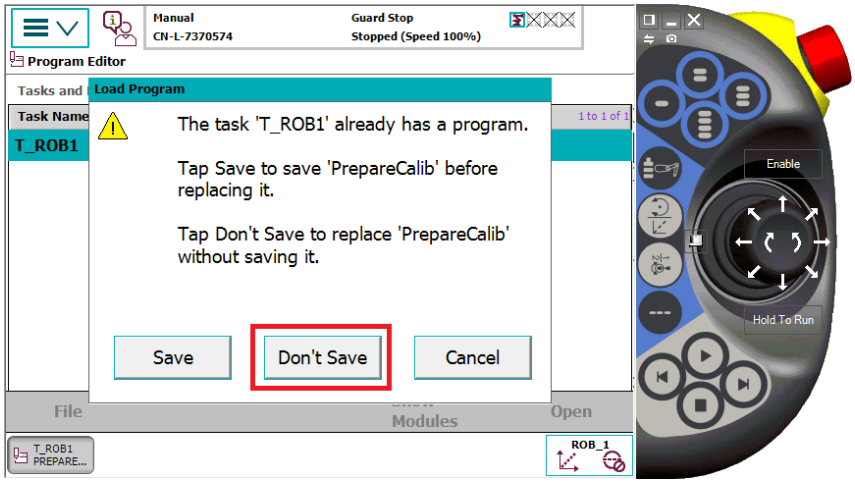
xx2200001928

F 表示されたダイアログで 保存しない をクリックします。

次のページに続く

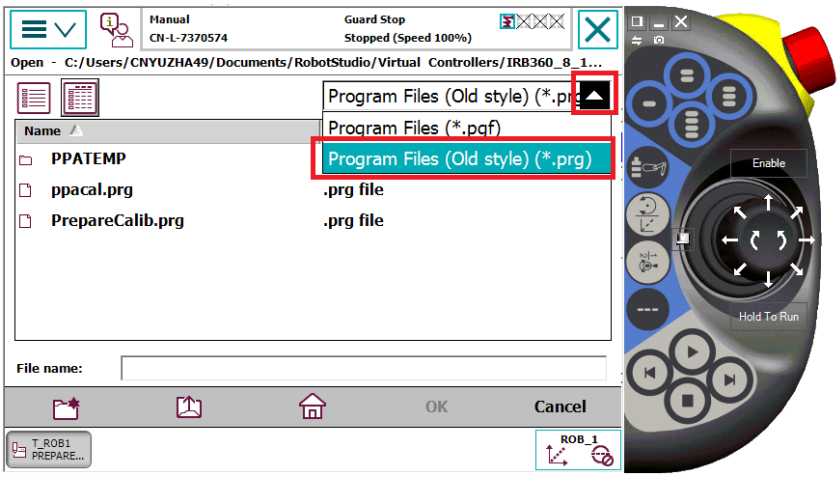
© Copyright 2025 ABB. All rights reserved.

4.4.5.1.3 ベース フレームの定義
続き



xx2200001929

G 右上隅のドロップダウン リストで **プログラムファイル (古いスタイル)(.prg)** をクリックします。

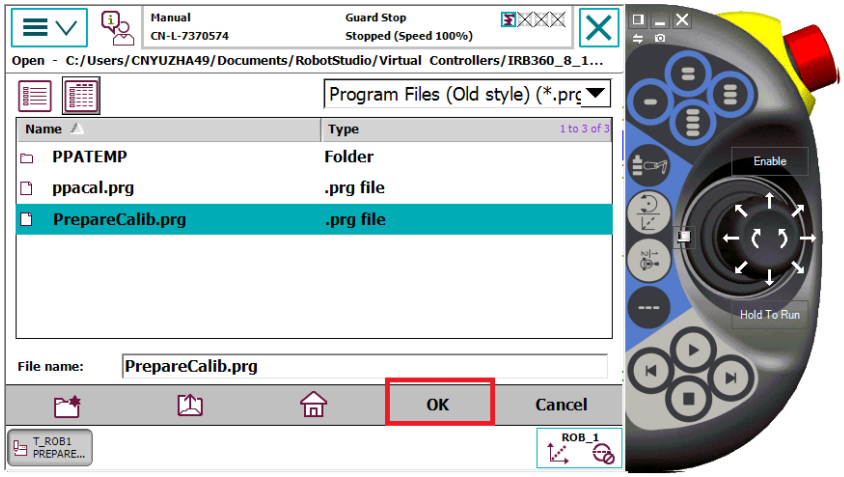


xx2200001930

H `PrepareCalib.prg` を選択し、**[OK]** をクリックします。

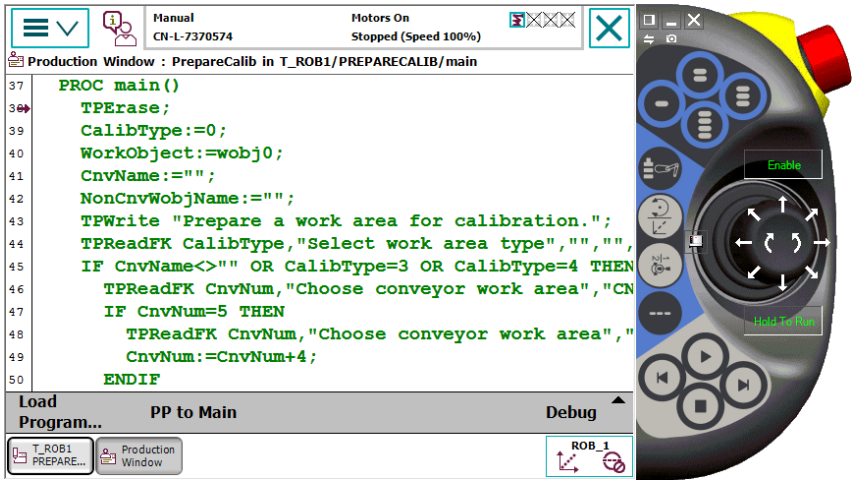
次のページに続く

© Copyright 2025 ABB. All rights reserved.



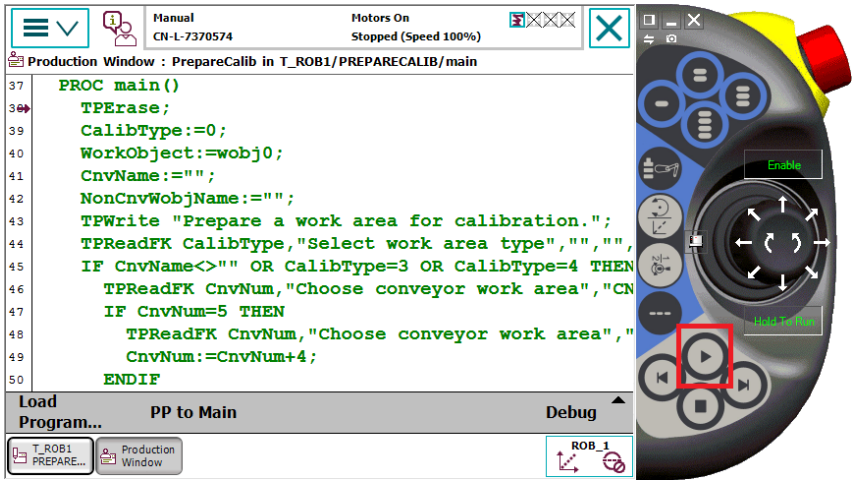
xx2200001931

I コントローラのモーターへの有効化デバイス ボタンを有効にします。



xx2200001933

J 再生 をクリックします。



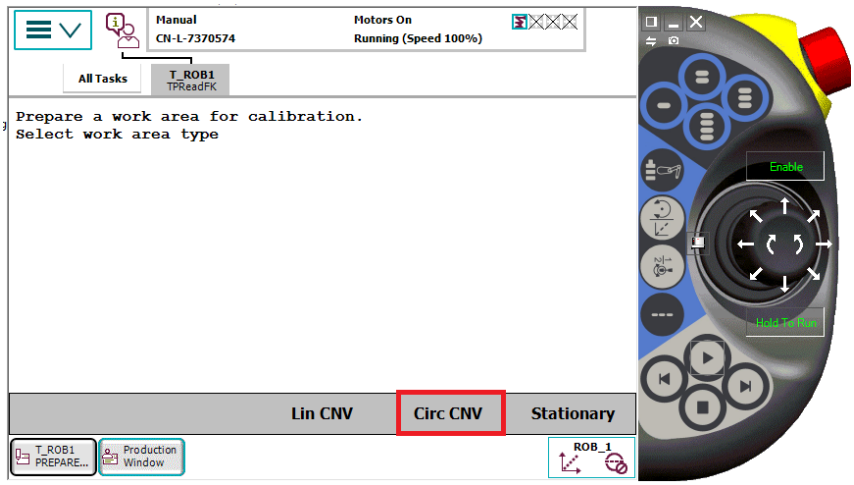
xx2200001934

K 作業エリア タイプ Circ CNV を選択します。

次のページに続く

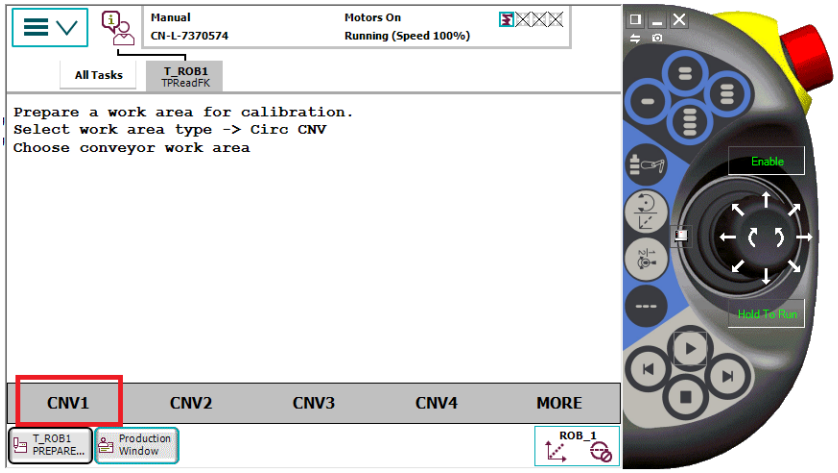
© Copyright 2025 ABB. All rights reserved.

4.4.5.1.3 ベース フレームの定義
続き



xx2200001935

L コンベアを選択します。例えば、**CNV1**。

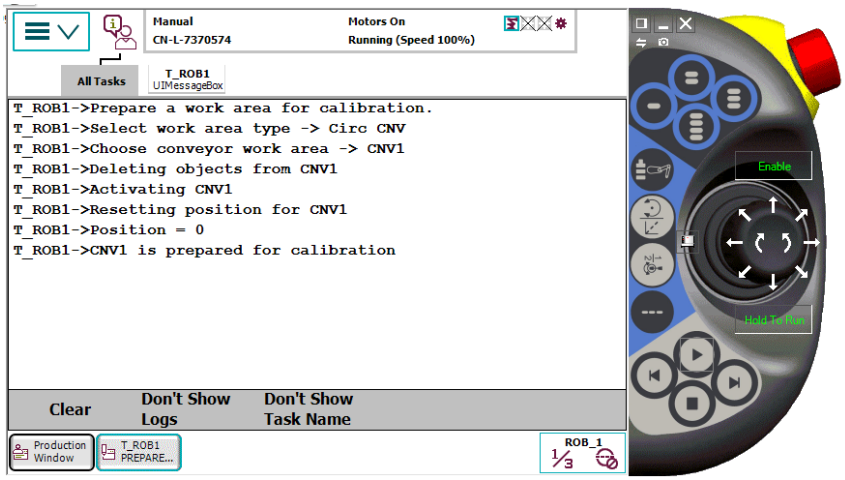


xx2200001936

M メッセージ...キャリブレーションの準備ができています を待ちます。CNV1 のジョギング ウィンドウのコンベアポジションは、「0」mm として表示されるはずです。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

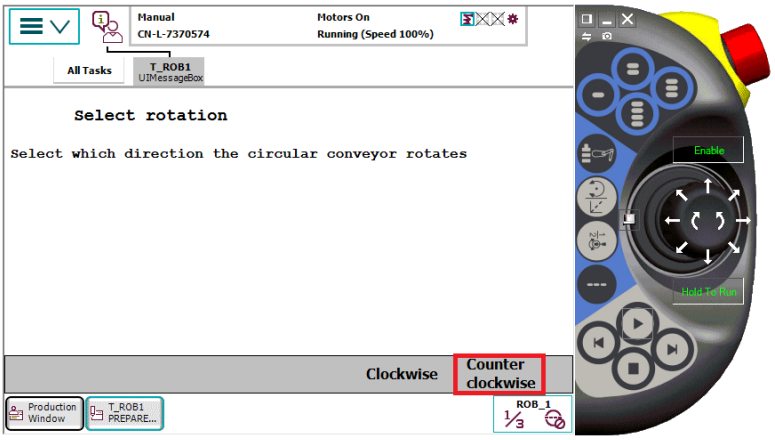


xx2200001937

5 コンベアに沿ってキャリブレーションが必要な作業エリアを持つロボットに対して、次の手順を実行します。

A 回転を選択のメッセージが表示されるまで待ち、コンベアの方向を選択します。

- 反時計回りの移動方向の場合は、反時計回りを選択します。



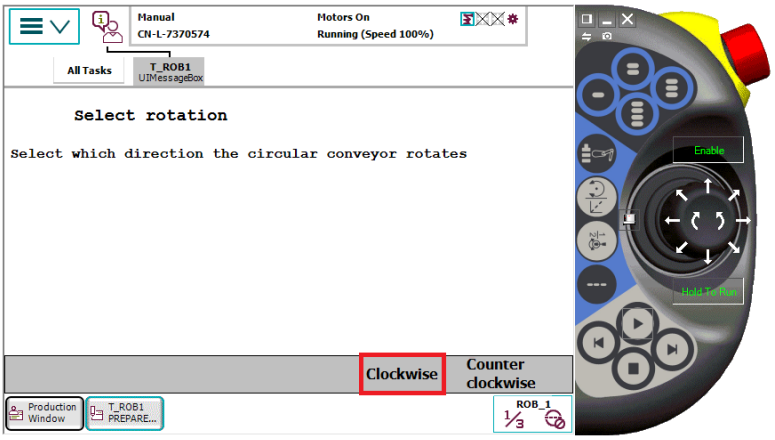
xx2200001938

- 時計回りの移動方向には、時計回りを選択します。

次のページに続く

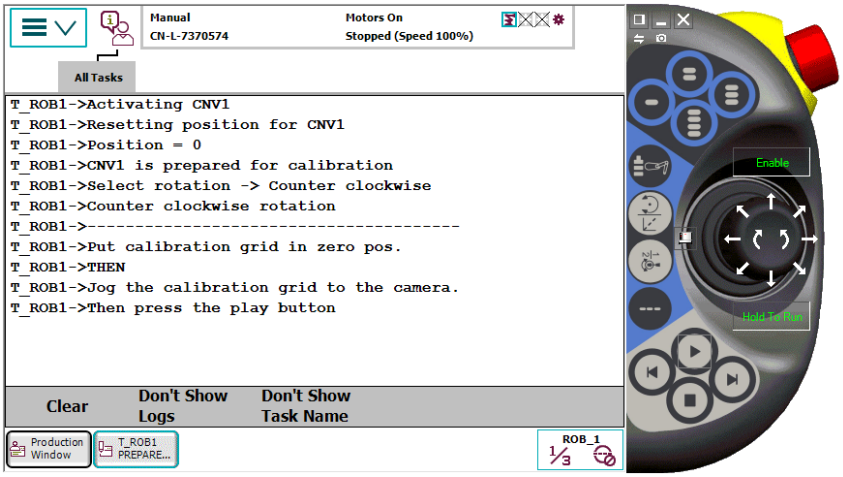
© Copyright 2025 ABB. All rights reserved.

4.4.5.1.3 ベース フレームの定義
続き



xx2400000743

B プログラムは自動的に続行されます。



xx2200001939

C ベルトをポジション B (基準ポイント) まで回転させ、カメラ (d) (ゼロポジション) の下にキャリブレーション グリッドを作成します。

反時計回りを選択した場合、ベルトを反時計回りに回転させます。

時計回りを選択した場合、ベルトを時計回りに回転させます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

ヒント

コンベアベルトを手で基準ポイントまで回転させます。

- ソース タイプが [事前定義] で、トリガー タイプが [距離] の場合、基準ポイントはHotspotの原点になります。
- ソース タイプが [事前定義] で、[I/O センサー、Hotspotとしてトリガー] が事前定義されたアイテムまたはコンテナの生成に使用される場合、基準ポイントはオブジェクトがセンサーによって検出されるポイントにある必要があります。このポイントが、検出されたアイテムまたはコンテナのローカル原点になります。
- ソース タイプがビジョンの場合、基準ポイントはカメラ ビューのローカル原点になります。

  カメラを使用する場合は、同時にカメラを調整します。カメラのキャリブレーション ページ *282*を参照してください。

  カメラが調整されたばかりの場合、基準ポイントはコンベアに取り付けられたカメラ調整パターンの原点によってすでにマークされています。

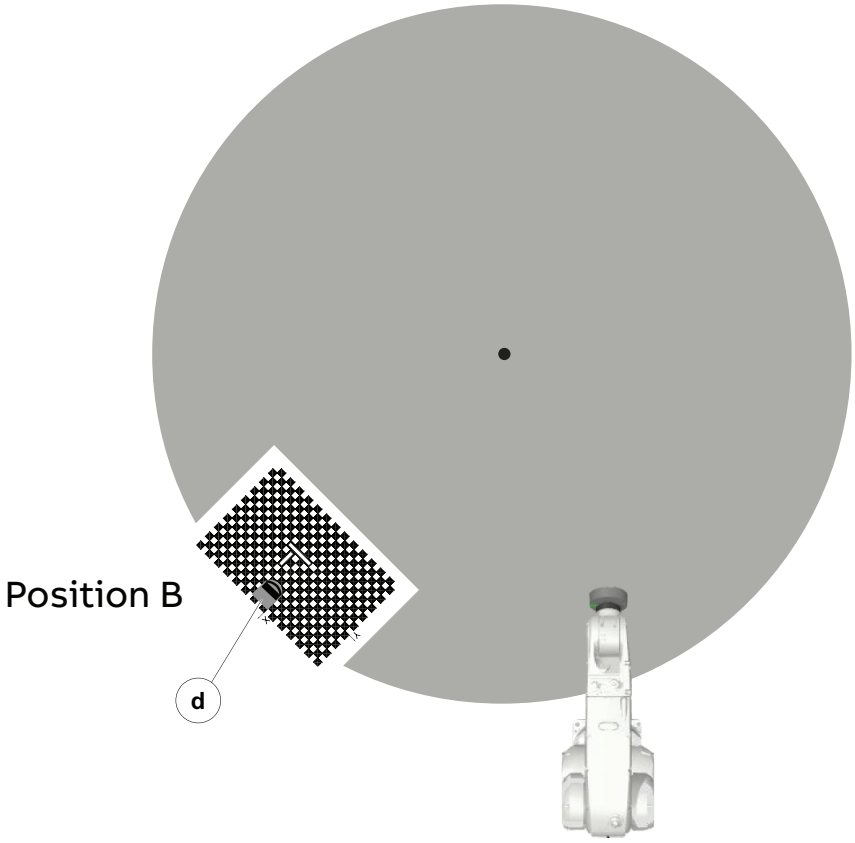
注記

ベルトを前後に回転させないでください。これにより、不正確な調整データが生成されます。

次のページに続く

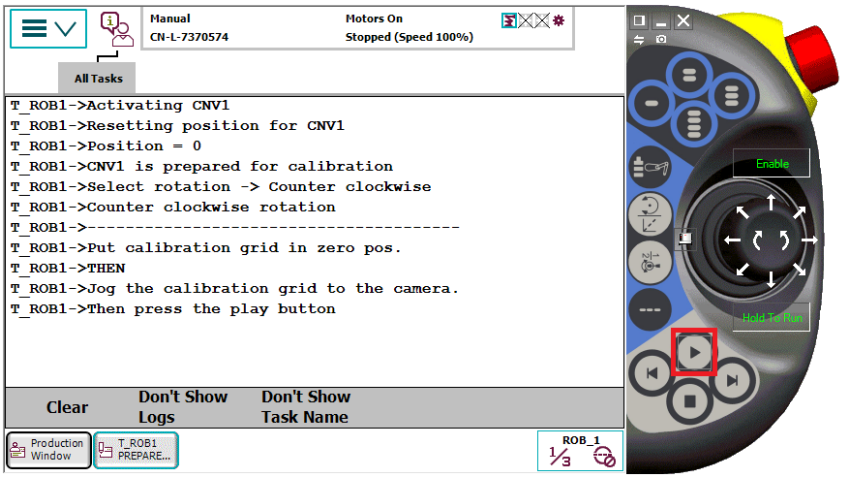
© Copyright 2025 ABB. All rights reserved.

4.4.5.1.3 ベース フレームの定義
続き



xx2200002008

D リセットされたロボットの FlexPendant で再生 をクリックします。



xx2200001940

E ロボットが基準点に到達できる範囲の距離だけ、コンベヤベルトを前方に動かします。

3つのキャリブレーションポイント間（Point 1 ~ 3）の距離を長く均等に空けることをお勧めします。そうすることで、キャリブレーションの精度が向上するからです。

反時計回りを選択した場合、ベルトを反時計回りに回転させます。

時計回りを選択した場合、ベルトを時計回りに回転させます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

注記

ベルトを前後に回転させないでください。これにより、不正確な調整データが生成されます。

F ロボットを手動でジョグまたは移動します。コンベヤ上の基準点をキャリブレーションツールTCPで正確にポイントします。

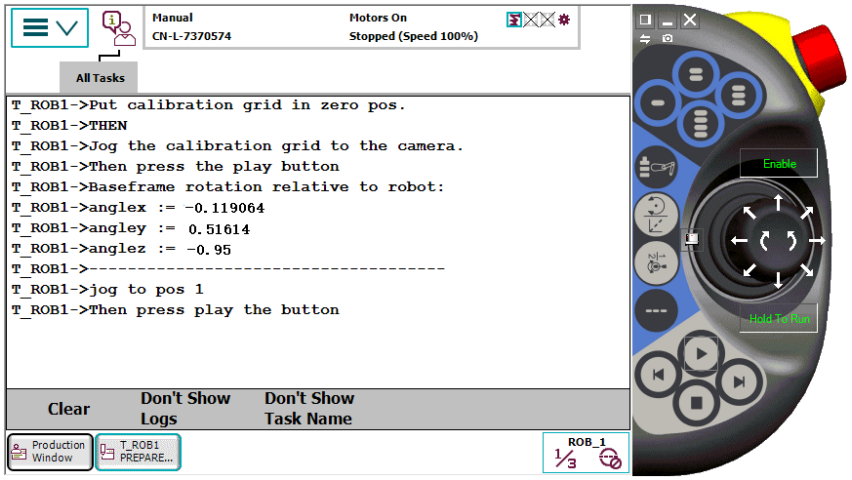
G 再生 をタップしてポイント (ポジション 1) を変更します。

H ポイント ポジション 2 と ポジション 3 に対して手順を繰り返します。

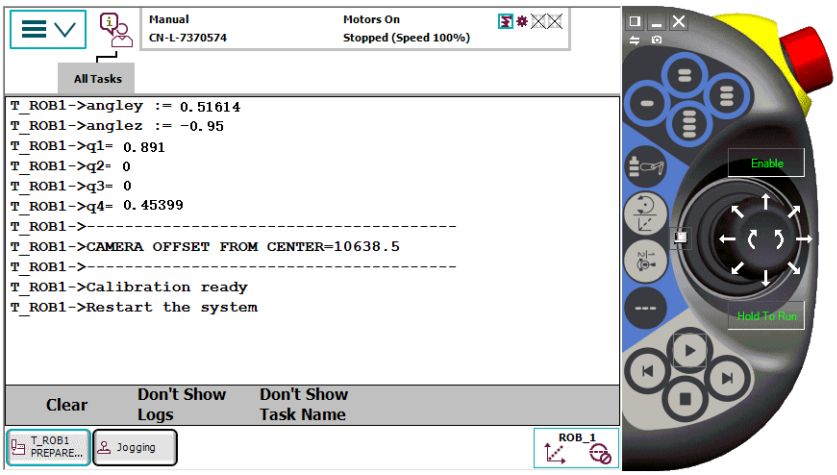
I ベース フレーム計算の表示された平均誤差と最大誤差が許容範囲内かどうかを確認します。 推定誤差が許容できる場合は、システムを再起動して、新しいベース フレームを確認して保存します。

注記

ほとんどの場合、1 mm未満の平均誤差は許容されます。



xx2200001941



xx2200001942

次のページに続く

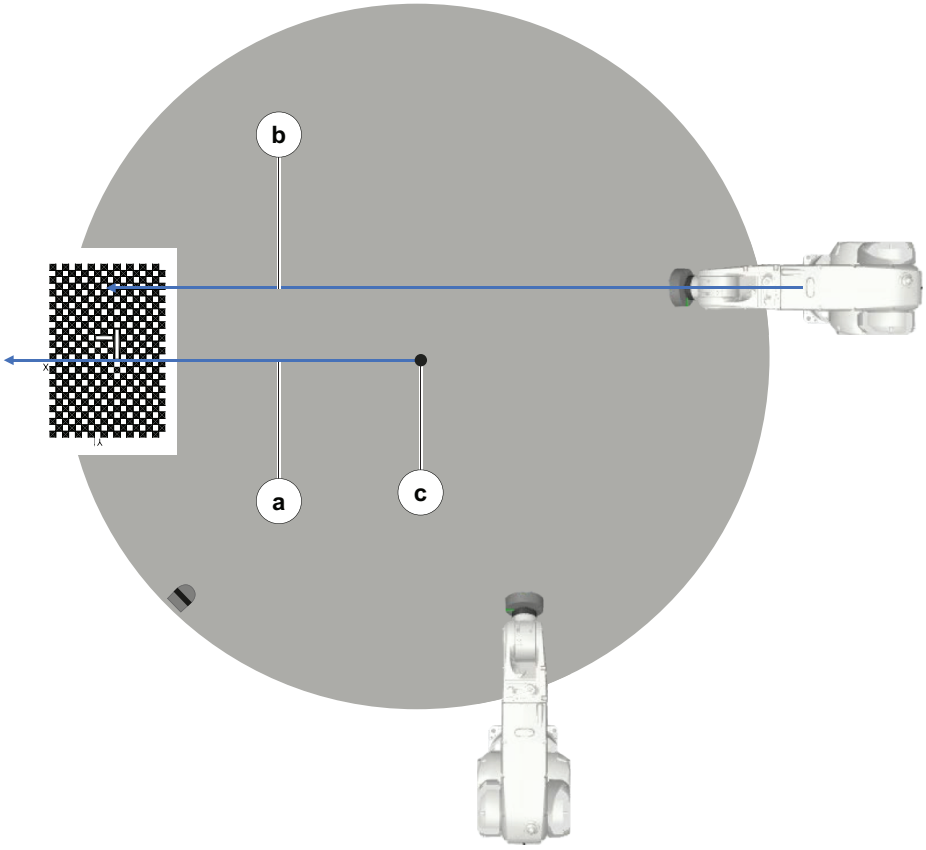
© Copyright 2025 ABB. All rights reserved.

4.4.5.1.3 ベース フレームの定義
続き

推定誤差に問題がある場合、このベースフレームは再補正する必要があります。

> **ヒント**
>
> カメラ センターからのオフセットの値を読み取ります。 この値は、円形コンベアのタイプ設定 ページ 263のSensor offsetの入力として使用されます。

6 キャリブレーションが必要なロボットが他にもある場合は、ロボットごとに手順 3 から手順 5 を繰り返します。



xx2400000405

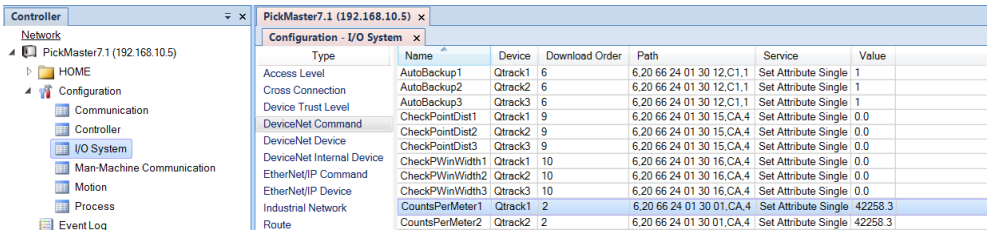
© Copyright 2025 ABB. All rights reserved.

4.4.5.2 DSQC 377での円形コンベアのキャリブレーション

4.4.5.2.1 パラメータ「カウント/メートル」の定義

概要

*Counts Per Meter*システムパラメータは、コンベアエンコーダのキャリブレーションに使用します。*Counts Per Meter*システムパラメータは、トピック*I/O System*のタイプ*DeviceNet Command*に属します。



xx1900000551

Counts Per Meterの計算

Counts Per Meter の値 システム パラメータは次のように計算されます。

`position1/measured_radians`

値	説明
`position1`	FlexPendant または RobotStudioの事前定義された I/O シグナルから読み取ります。 たとえば、CNV1、シグナル名はc1counts です。
`measured_radians`(ラジアン)	コンベアが移動したことを手動で測定したラジアン。

Counts Per Meterの定義

以下の手順に従って、コンベヤエンコーダーの*Counts Per Meter*を定義します。

1. コンベヤベルトにマークを付けます。たとえば、線を引いたり、テープを貼り付けたりします。コンベアの固定側の同じポジションにもマークを付けます。
2. FlexPendant **プログラム エディタ** で、プログラム `ppacal.prg` をロードして実行します。

   これにより、コンベアの現在の位置がゼロに設定されます。値は、FlexPendant **入力と出力**-> **I/O デバイス**-> **RobICI**-> **DeviceNet** ウィンドウの位置部分に **CNV** 値として表示されます。
3. コンベア ベルトを約 180 度回転させます。
4. FlexPendant **入力と出力**-> **I/O デバイス**-> **RobICI**-> **DeviceNet** ウィンドウで、コンベアの c1counts を読み取ります。これは `position1` です。
5. 2 つのマークの間の物理ラジアンを測定します。 これは、`measured_radians`(測定されたラジアン)の値です。
6. 読み込んだ値と測定した値を使用して、*Counts Per Meter*を計算します。

   例えば、 `30000/0.5 = 60000`

この変数が円形コンベアに適用される場合、実際の意味はラジアンあたりのカウントです。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.5.2.1 パラメータ「カウント/メートル」の定義
続き

7 RobotStudioで、［構成］ を開き、トピックI/O Systemを選択し、DeviceNet Commandを入力します。
8 ユニット*Qtrackx*（x はコンベヤ番号）を選択し、パラメータ*Counts Per Meter*の値を更新します。
9 ［OK］をタッチします。
10 コントローラを再起動します。

関連情報

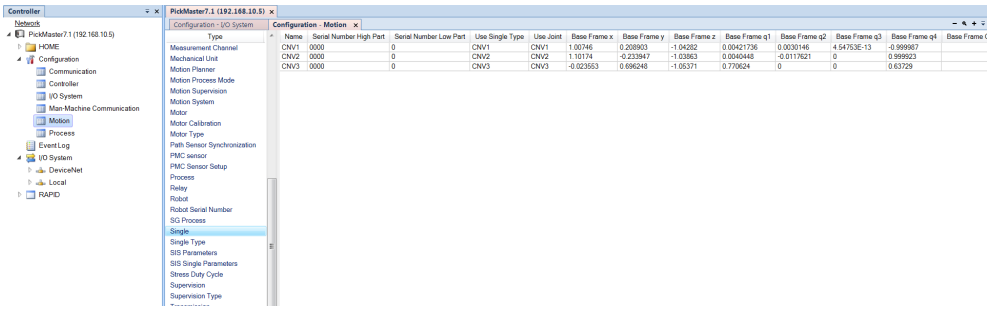
Application manual - Conveyor tracking。
『*Technical reference manual - System parameters*』。

© Copyright 2025 ABB. All rights reserved.

4.4.5.2.2 ベース フレームの定義

概要

円形コンベアのコンベア作業エリアごとに、コンベアベースフレームのキャリブレーションを行います。ベースフレームのキャリブレーションを行うことで、作業エリアでセンサー検出物体のピックまたは配置を行うときのロボットの基準点を求められます。



xx1900000592

準備

- 各コンベア作業エリアのCounts Per Meter(メーターごとのカウント) システム パラメータを定義します。 詳細については、パラメータ「カウント/メートル」の定義 ページ 257, パラメータ「カウント/メートル」の定義 ページ 230を参照してください。
- ロボットに一時的に取り付けるキャリブレーションツールを準備します。このキャリブレーションツールにはポイントが付いたTCPがあるため、TCPオフセット値を正確に測定します。
- 各ロボットについて、RAPIDプログラムでキャリブレーションツールのツールデータを作成します。TCPオフセットを測定値で更新します。FlexPendant ジョグウィンドウで、ロボットのツールデータを選択します。
- カメラを使用する場合は、カメラをキャリブレーションします。カメラのキャリブレーション ページ 282 を参照してください。カメラのキャリブレーション後、カメラキャリブレーションパターンをコンベアに添付します。

Recommendation

このセクションでは、TCP 測定と RAPID プログラムを使用して、円形コンベアのコンベア ベース フレームポジションと四元数を計算する方法について説明します。

この方法では、円形コンベア上の 3 つの測定ポイントを使用して、回転の中心を計算します。 3 つのポイントは、周囲でできるだけ離して配置する必要があります。

ベース フレームの向きを定義し、ウィンドウの起動キャリブレーションを開始する

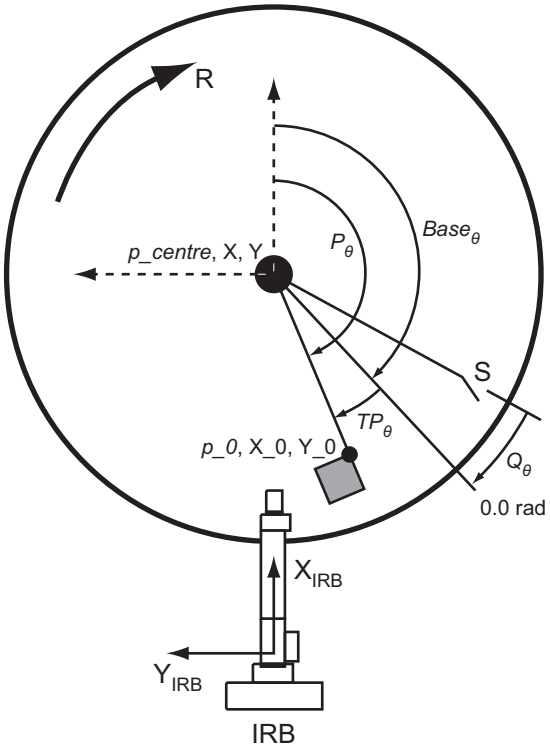
ベース フレーム四元数は、ロボット モーションの 0.0 ラジアン ポイントがどこにあるかを定義します。

次の図は、円形コンベアのベース フレームの方向を定義するときに使用される角度の例を示しています。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.5.2.2 ベース フレームの定義
続き



xx1200001103

R	回転方向
S	同期スイッチ
Q_θ	キュートラッキング距離角度
TP_θ	FlexPendant に表示される角度
P_θ	p_0 のポジションから計算された角度
$Base_\theta$	四元数に変換されるベース フレーム角度

ベース フレームの x ポジションと y ポジションの計算

この手順を使用して、ベース フレームの x ポジションと y ポジションを計算します。

1. FlexPendant で `Wobj0` を使用します。 円形コンベア上の基準ポイントを選び、TCP をこのポイントまでジョグし、*p_0* を記録します。
2. コンベアを別のポジションに移動します。 TCP を基準ポイントにジョグし、*p_1* を記録します。
3. コンベアを 3 番目のポジションまで動かし、TCP を基準点までジョグして、*p_2* を記録します。
4. 関数 `CNVUTL_cirCntr` をポイント *p_0*、*p_1*、および *p_2* で使用して、円の中心 *p_centre* を計算します。

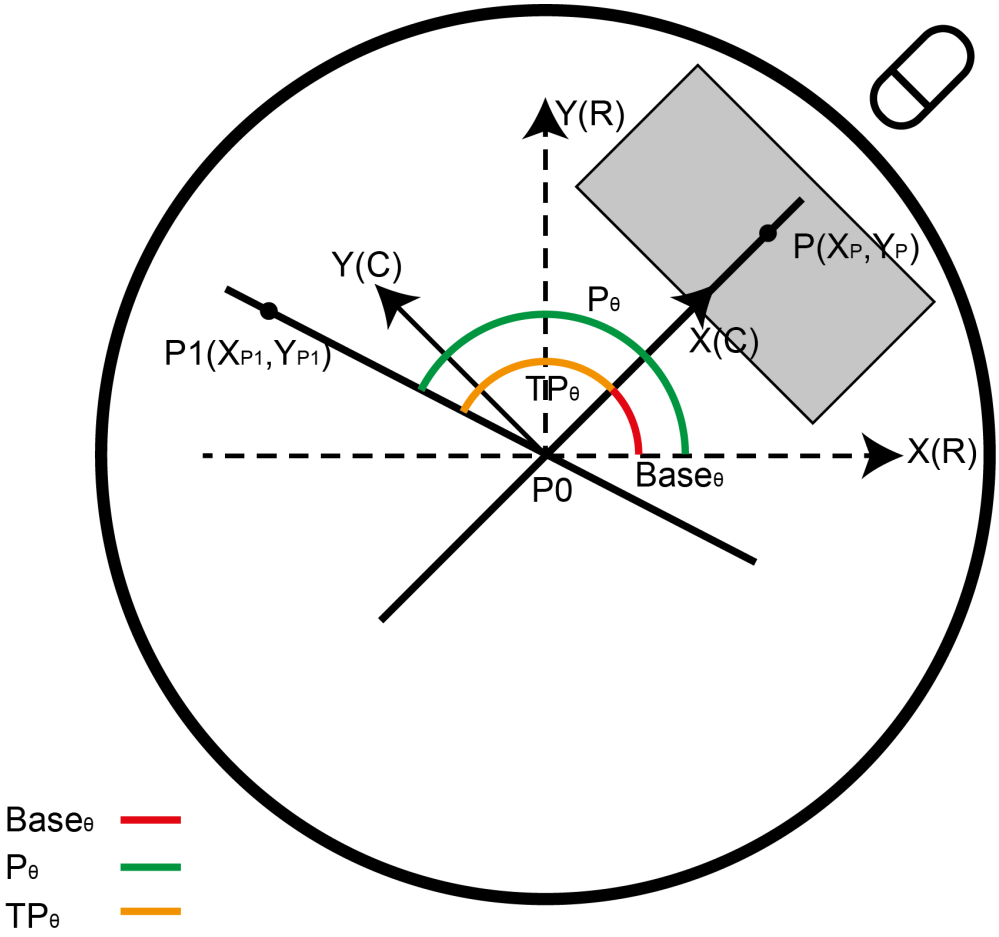
   システム モジュール `cnv_utl.sys` は Robotware にあります。
5. *p_centre* から x と y の値を取得し、それらをコンベアのベース フレーム値に入力して、メートルに変換します。*Application manual - Conveyor tracking* を参照してください。 作業物体のゼロポジションが選択されると、z 値は後で入力されます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

四元数の計算

この手順を使用して、ベース フレームの向きの四元数を計算します。



xx1900000678

1 ベース フレームの x ポジションと y ポジションを計算するときに、手順 5 で記録した角度を使用します。 これは角度TP_θです。ベース フレームの向きを定義し、ウィンドウの起動キャリブレーションを開始する ページ *259* の測定ポイントの例を参照してください

2 *P0* の *XP1* および *YP1* 座標と atan 関数から Pθ を計算します。
ポイントが第 1 四分位または第 4 四分位にある場合: Pθ = arctan(YP1/XP1)
ポイントが第 2 四分位または第 3 四分位にある場合: Pθ = π + arctan(YP1/XP1)

> **ヒント**
>
> 計算ツールが arctan2 関数を提供する場合、四分位数を判断して Pθ = arctan2 (XP1, YP1) を直接使用する必要はありません。

3 Baseの値を計算します。

$$Base_\theta = P_\theta - TP_\theta$$

4 回転方向を考慮して、ベース フレームの四元数を計算します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.5.2.2 ベース フレームの定義
続き

反時計回りの回転：

```
q1 = cos(Base θ /2)
q2 = 0.0
q3 = 0.0

q4 = sin(Base θ /2)
```

時計回りの回転：

```
q1 = 0.0

q2 = cos(Base θ /2)

q3 = –sin(Base θ /2)
q4 = 0.0
```

5 *p_0* からの z の値 (メートル単位) と、四元数 *q1*、*q2*、*q3*, および *q4* の値を、コンベアのベース フレームに入力します。

© Copyright 2025 ABB. All rights reserved.

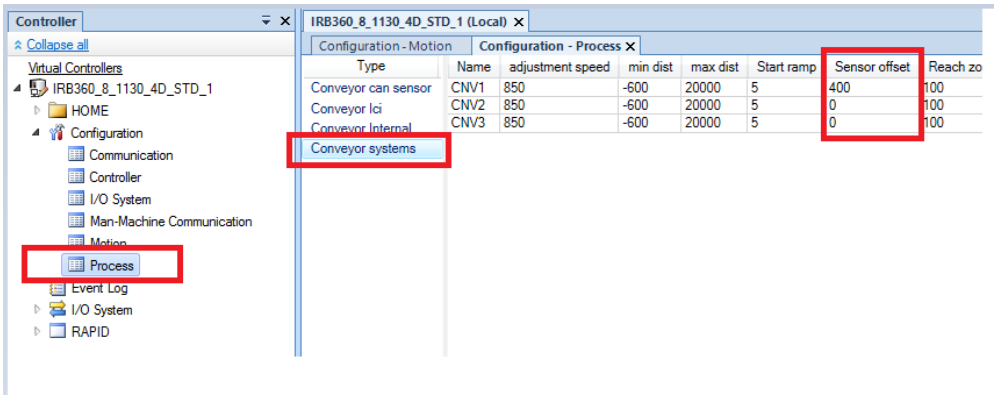
4.4.5.3 円形コンベアのタイプ設定

概要

円形コンベア上のコンベアワークエリアごとに、タイプ パラメータ、Sensor offset、Reach zone accuracy、Mechanics、Rotating Moveを設定する必要があります。

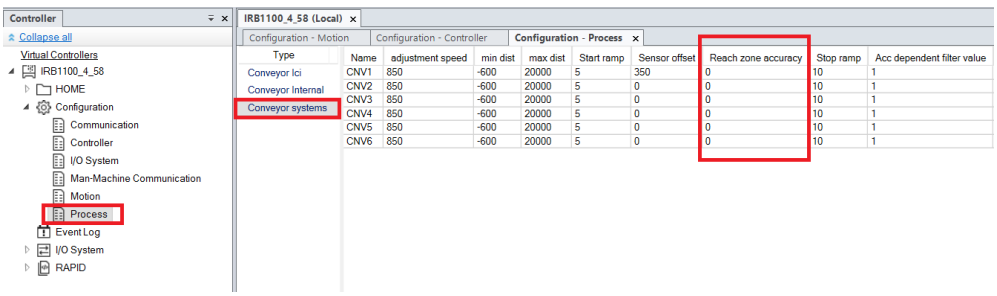
Sensor offsetは、センサーとコンベア ベース フレームの原点との間の距離を定義します。 たとえば、カメラを使用する場合、このパラメータは、円の中心からコンベア ベルト上のカメラの投影ポイントまでの距離を表します。

> **注記**
>
> DSQC 377の場合、Sensor offsetの距離は手動で測定されます。
>
> DSQC 2000の場合、Sensor offsetの距離はプログラム結果から読み取られます。 IRC5 ページ 255 と OmniCore ページ 243 の値を参照してください。



xx2100000066

Reach zone accuracyは、tool_0 とロボット ターゲット ポイントの間の正確な角度精度を定義します。値は0-100の間で設定できます。値が大きい場合 (たとえば 100) は、ロボットのグリップ ポーズが計算されたアイテムのターゲット ポイントと 100 パーセント一致する必要があることを意味します。一致しない場合、ロボットはアイテムをグリップできません。したがって、円形コンベアを使用し、時計回りに回転させる場合は、コンベア ベース フレームの反転の影響を無視するように0としてこのパラメータを設定する必要があります。



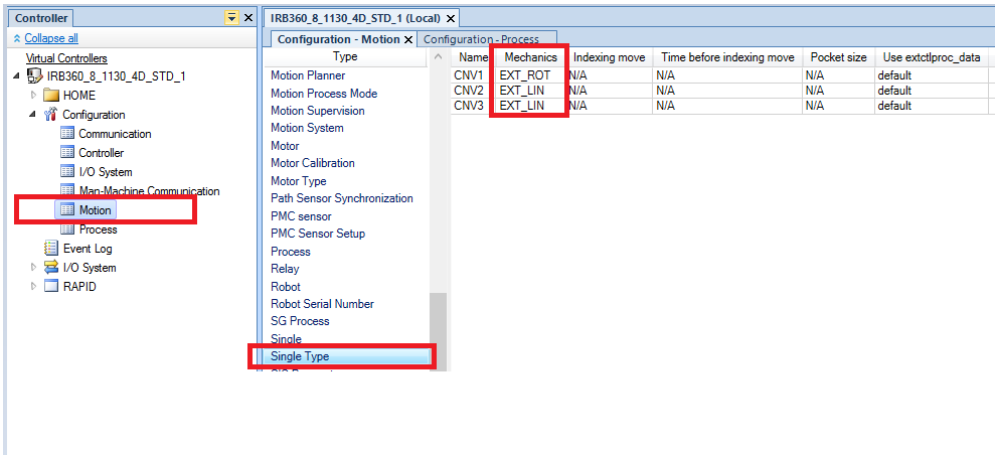
xx2400000407

次のページに続く

© Copyright 2025 ABB. All rights reserved.

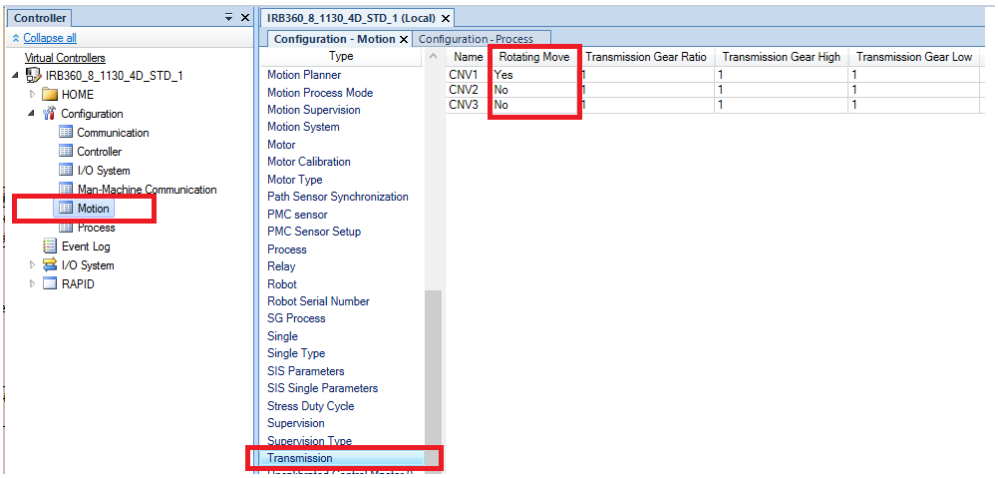
4.4.5.3 円形コンベアのタイプ設定
続き

Mechanicsは、コンベアの移動軌跡を定義します。 デフォルト値は EXT_LIN (リニア コンベア) です。 そのため、円形コンベアを使用する場合、このパラメータを EXT_ROT として設定する必要があります。



xx2100000067

Rotating Moveは、コンベアの回転状態を定義します。 デフォルト値は No (リニア コンベア) です。 そのため、円形コンベアを使用する場合は、このパラメータを Yes に設定する必要があります。



xx2100000068

© Copyright 2025 ABB. All rights reserved.

4.4.6 インデックスワークエリアのキャリブレーション

概要

インデックスワークエリアでは、ワークオブジェクトのキャリブレーションを行います。ワークオブジェクトのキャリブレーションによって、ワークエリアでセンサー検出オブジェクトのピックまたはプレイスを行うときのロボットの基準点を求められます。

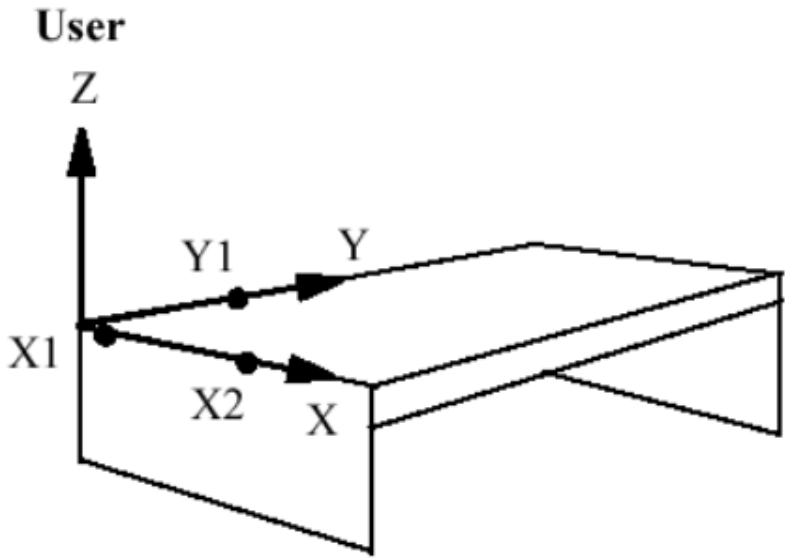
インデックスワークエリアのキャリブレーションの準備

- ロボットに一時的に取り付けるキャリブレーションツールを準備します。このキャリブレーションツールにはポイントが付いたTCPがあるため、TCPオフセット値を正確に測定します。
- 各ロボットについて、RAPIDプログラムでキャリブレーションツールのツールデータを作成します。TCPオフセットを測定値で更新します。FlexPendantジョグウィンドウで、ロボットのツールデータを選択します。
- カメラをキャリブレーションしてください。カメラのキャリブレーション ページ 282 を参照してください。 カメラのキャリブレーションが完了したら、カメラのキャリブレーション パターンをコンベアに取り付けたままにしておきます。
- ワークオブジェクトのキャリブレーションの基準x軸およびy軸がインデックスワークエリアで正確にマーク付けされていることを確認します。キャリブレーションには3つの基準点（x軸に2ポイント、y軸に1ポイント）が必要です。
  - カメラを使用する場合、カメラビューのローカル起点として基準x軸とy軸をマーク付けします。カメラのキャリブレーションが終了すると、インデックスワークエリアに添付されたカメラキャリブレーションパターンによってローカル起点がマークされています。
  - ポジションジェネレータI/O信号を使用して、事前定義のポジションを生成する場合は、アイテムまたはコンテナが生成されるローカル起点として好ましい場所に、基準x軸およびy軸をマーク付けします。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.6 インデックスワークエリアのキャリブレーション
続き



xx1400002201

操作方法(OmniCore)

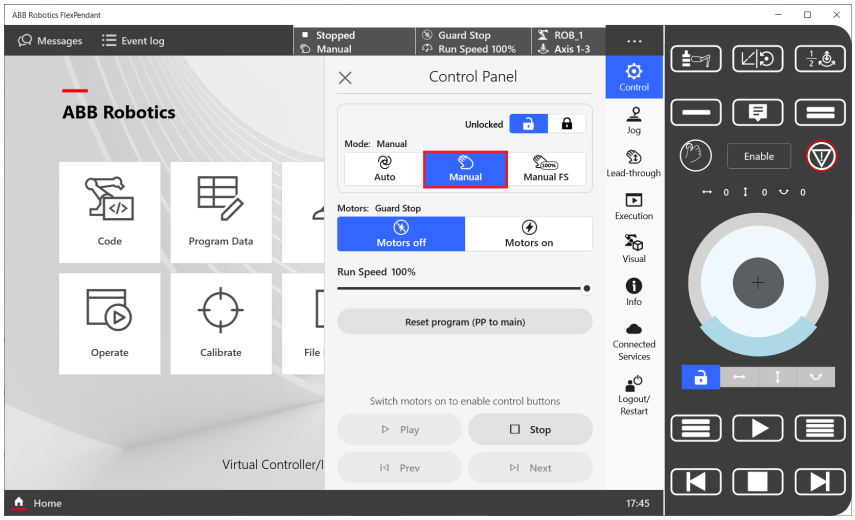
1 コンベヤベルトにキャリブレーションの基準点が正確にマークされていることを確認してください。
   - カメラを使用する場合、基準点はカメラビューのローカル起点です。カメラのキャリブレーションが終了すると、コンベヤに添付されたカメラキャリブレーションパターンの起点によって基準点がすでにマークされています。
   - I/Oセンサーを使用して事前定義のポジションを生成する場合は、コンベヤ上で、センサーがオブジェクトを検出するポイントに基準点をマーク付けします。このポイントは、検出されたアイテムまたはコンテナのローカル起点になります。
2 コンベヤ（エンコーダーボード）ポジションをリセットします。

> **注記**
>
> このステップが完全に終了するまで、コンベヤを動かさないでください。

コンベヤに沿ってキャリブレートする必要のある、ワークエリアを持ったすべてのロボットについて、以下を行います。
   - コントローラを手動モードに設定します。

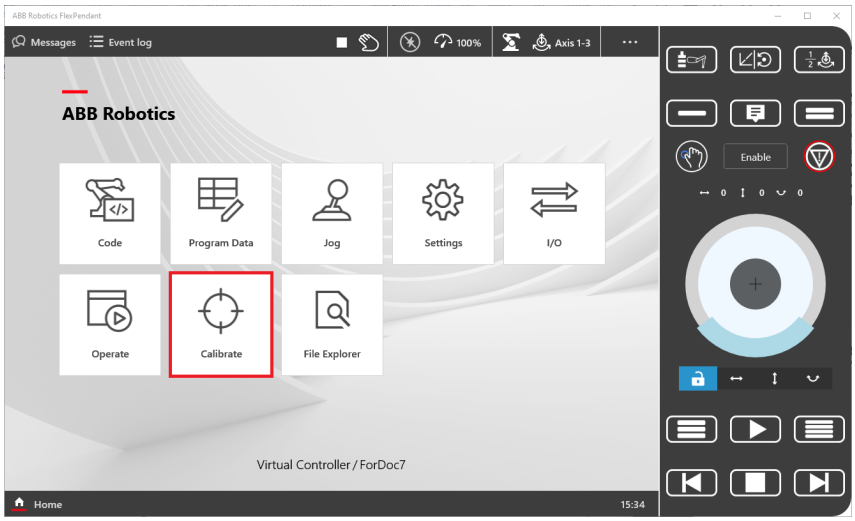
次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2100000368

- FlexPendant で、**キャリブレート** をクリックします。



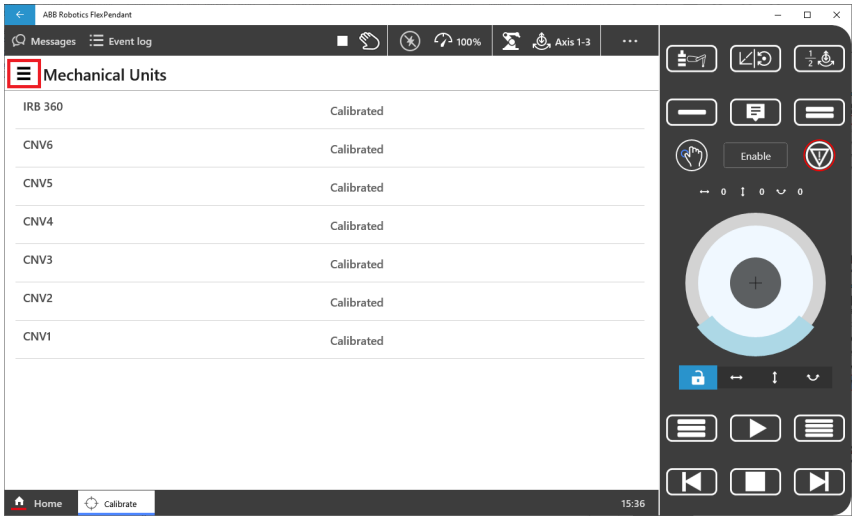
xx2100000362

- 左上隅の **オプション タブ** をクリックします。

次のページに続く

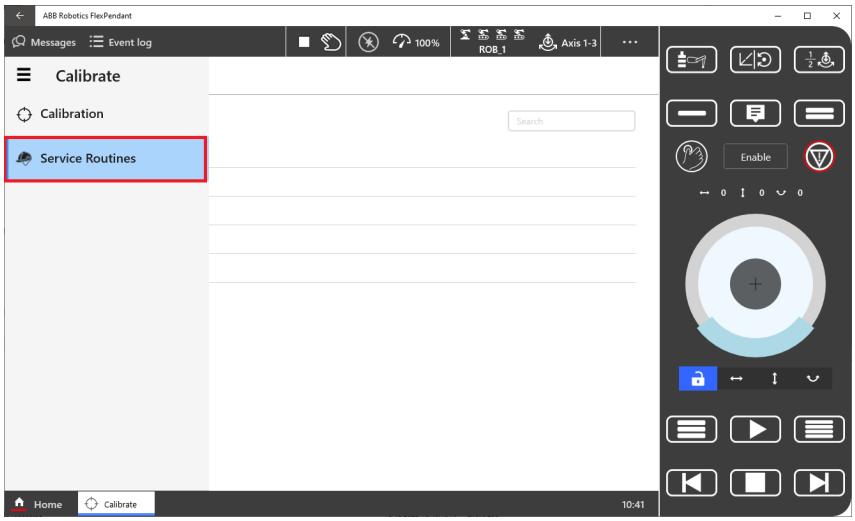
© Copyright 2025 ABB. All rights reserved.

4.4.6 インデックスワークエリアのキャリブレーション
続き



xx2100000363

- **サービス ルーチン** をクリックします。

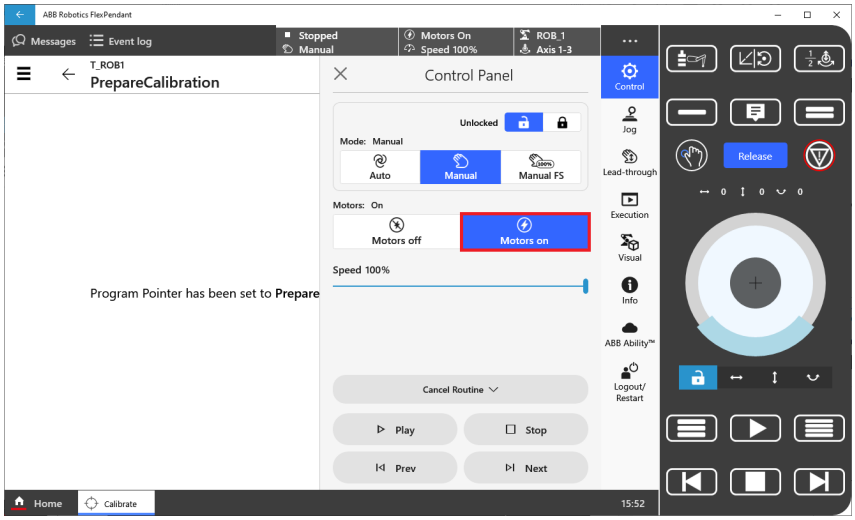


xx2100000364

- コントローラのモーターへの有効化デバイス ボタンを有効にします。

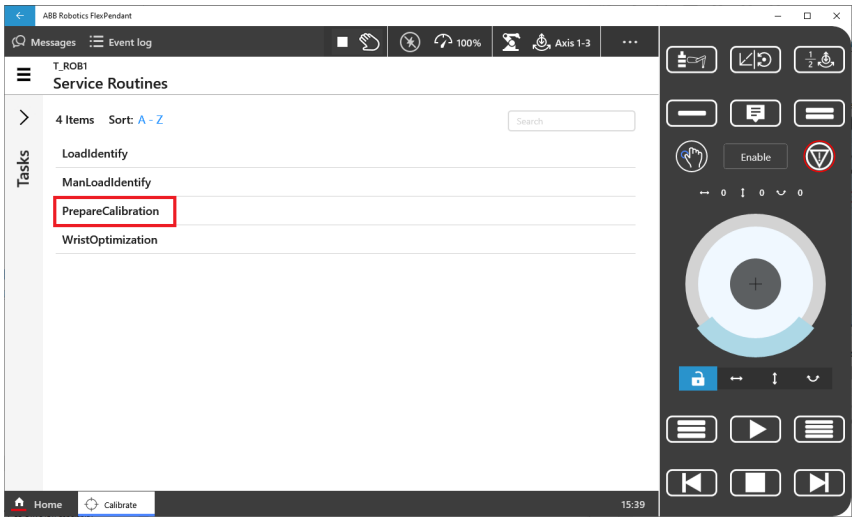
次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2100000369

- **キャリブレーションの準備** をクリックします。



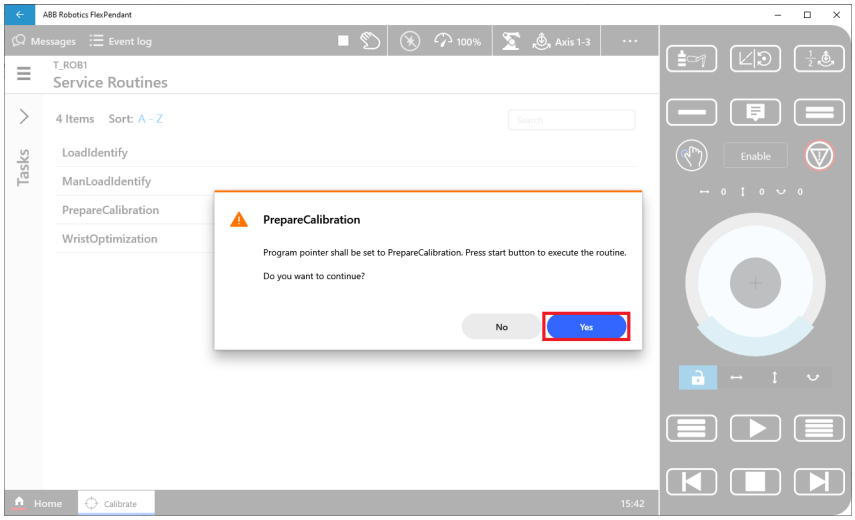
xx2100000365

- 表示されたダイアログで **はい** をクリックします。

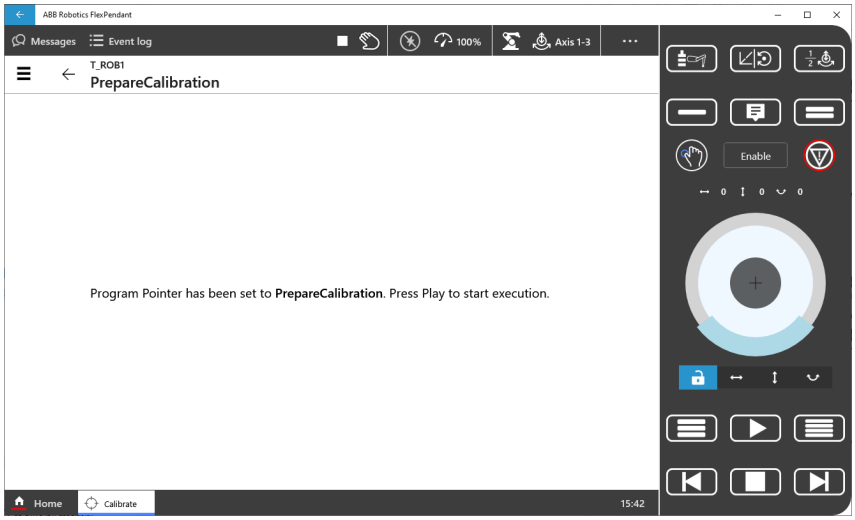
次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.6 インデックスワークエリアのキャリブレーション
続き



xx2100000366

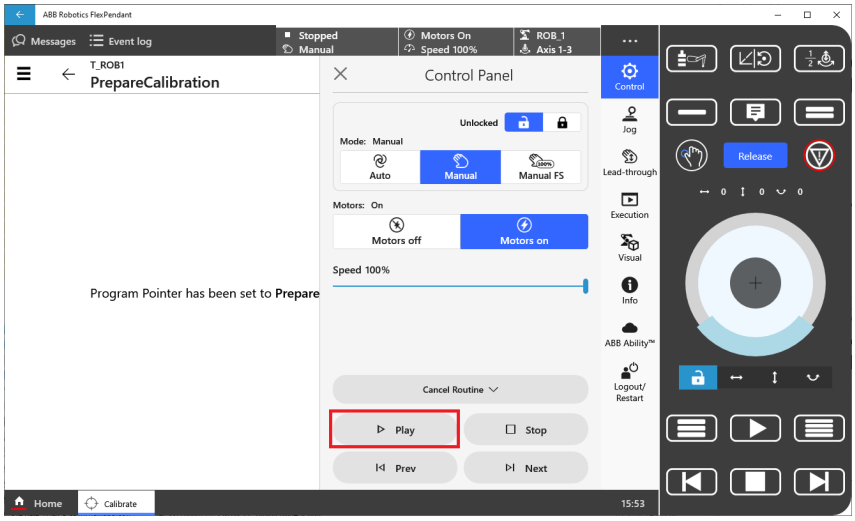


xx2100000367

- **再生** をクリックします。

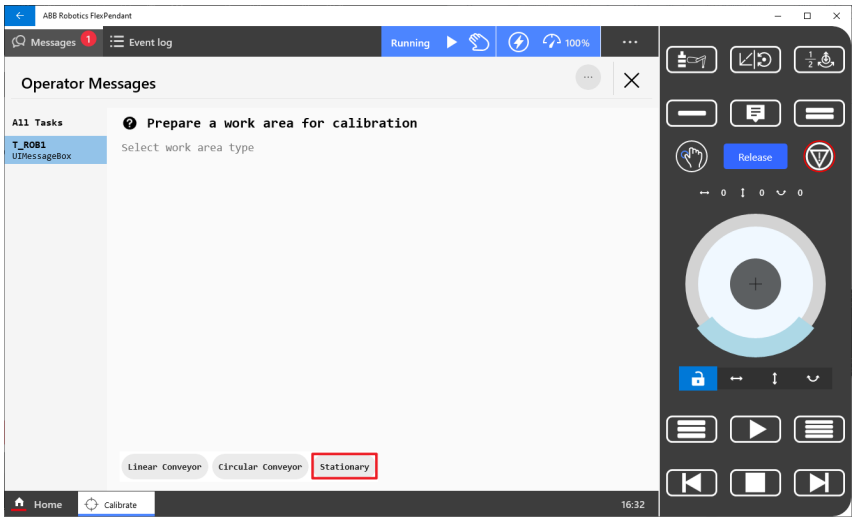
次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2100000370

- 作業エリアタイプ固定を選択します。



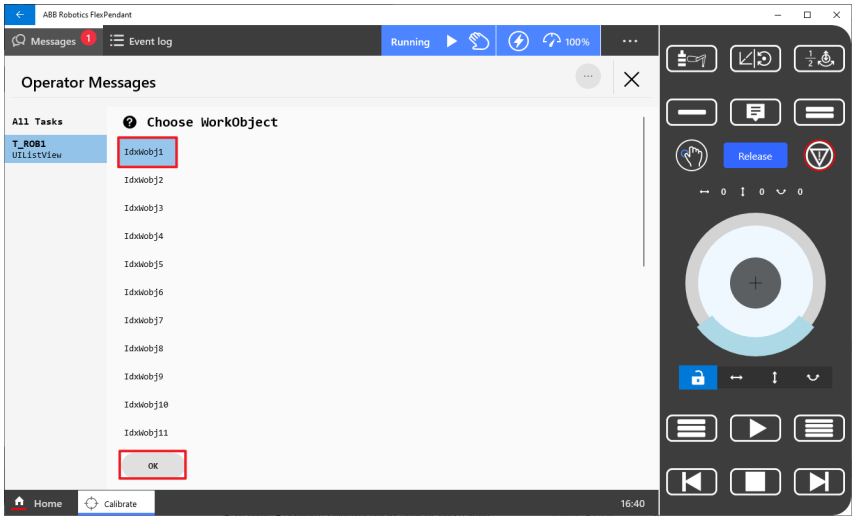
xx2100000396

- コンベアを選択します:例えば、Idxwobj1。 次に、OK をクリックします。

次のページに続く

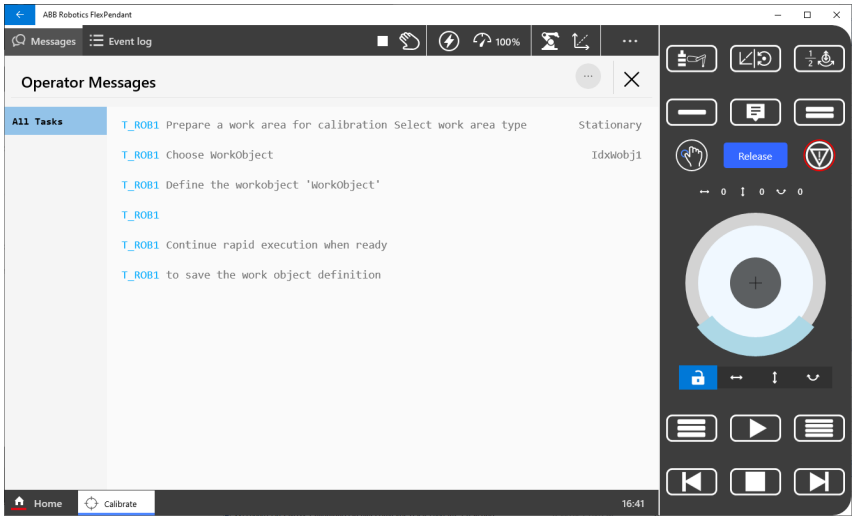
© Copyright 2025 ABB. All rights reserved.

4.4.6 インデックスワークエリアのキャリブレーション
続き



xx2100000397

- メッセージ...キャリブレーションの準備ができています を待ちます。CNV1 のジョギング ウィンドウのコンベアポジションは、「0」mm として表示されるはずです。

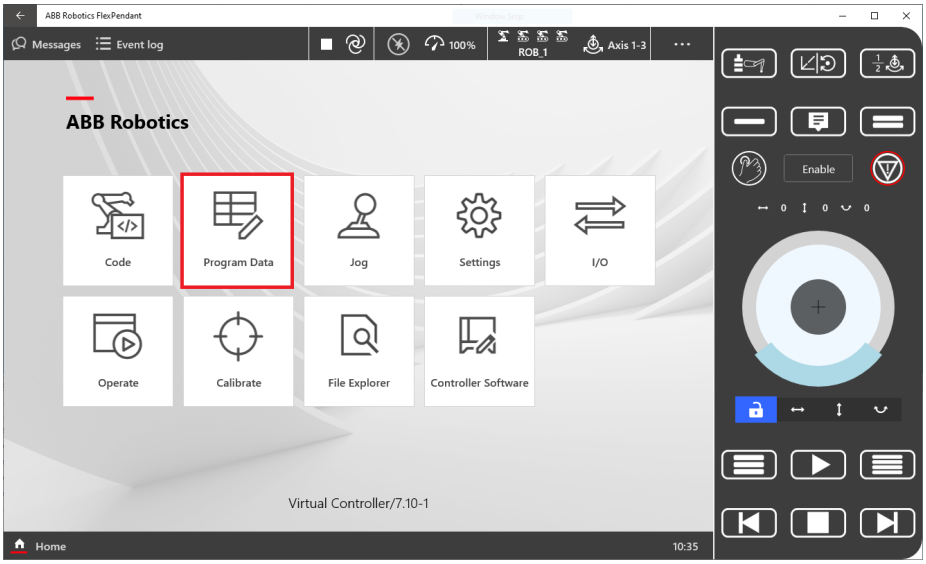


xx2100000398

3 ホームページに戻り、プログラム データを選択します。

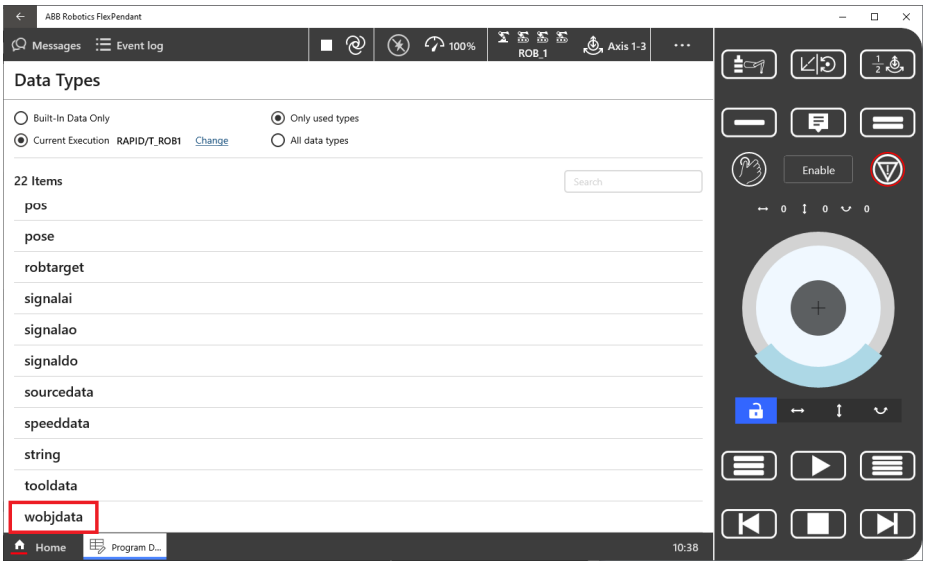
次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2300001587

4 データタイプで作業オブジェクトを選択します。



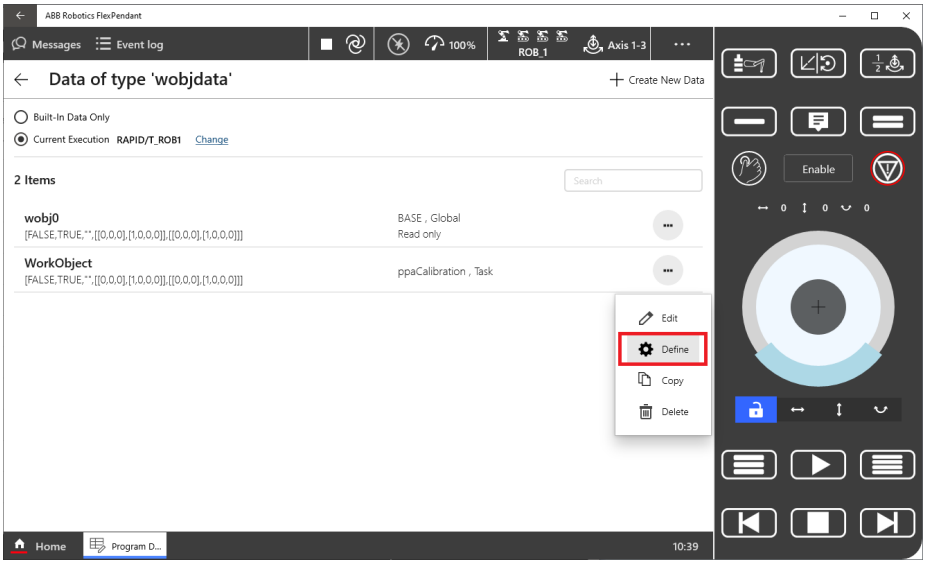
xx2100000380

5 作業オブジェクトで ... をタップして 定義 を選択します。

次のページに続く

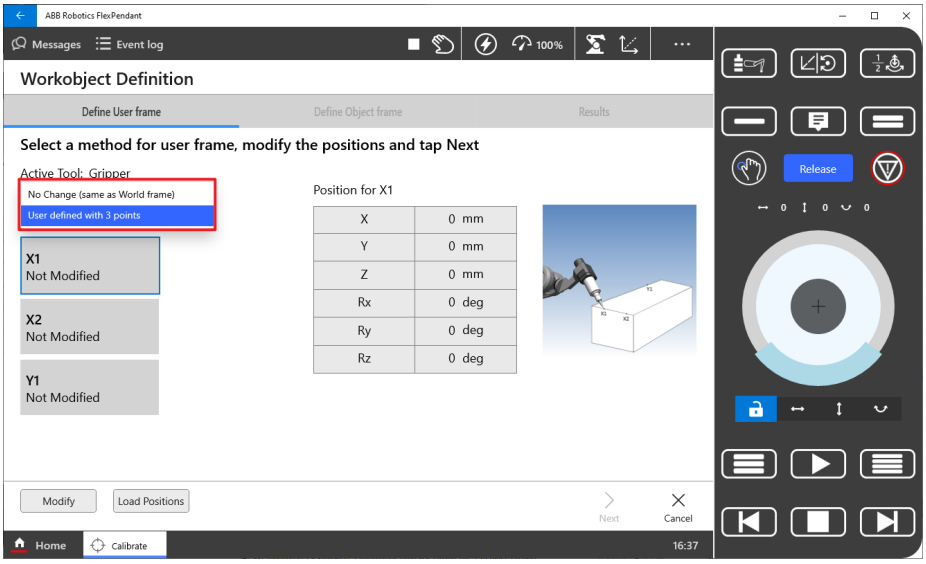
© Copyright 2025 ABB. All rights reserved.

4.4.6 インデックスワークエリアのキャリブレーション
続き



xx2100000381

6 **ユーザー フレームの定義 ウィンドウで、ユーザー メソッド を** `User defined with 3 points`**[3 点でユーザー定義] に設定します。**

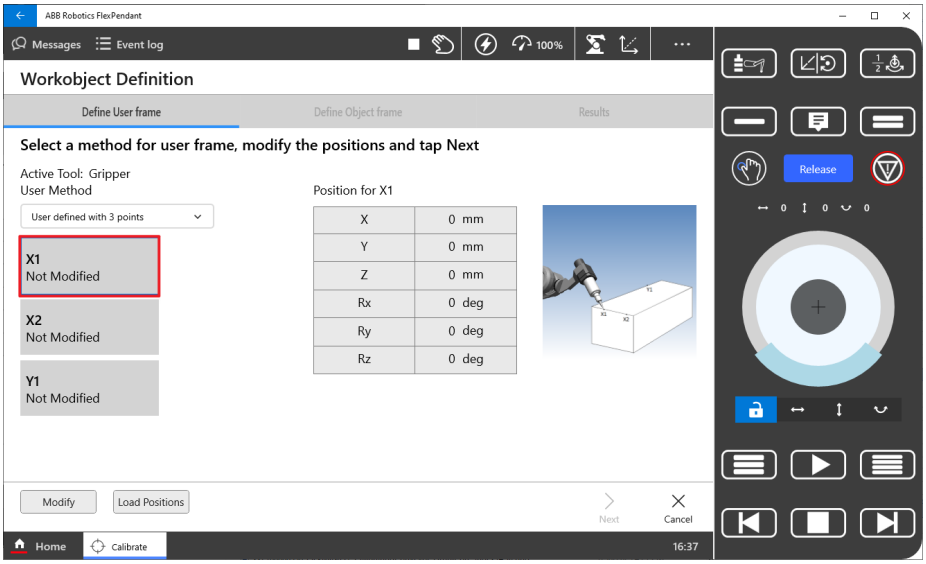


xx2100000382

7 **X1 を選択します。 ロボットの TCP で原点に近い x 軸上の場所を指摘します。**

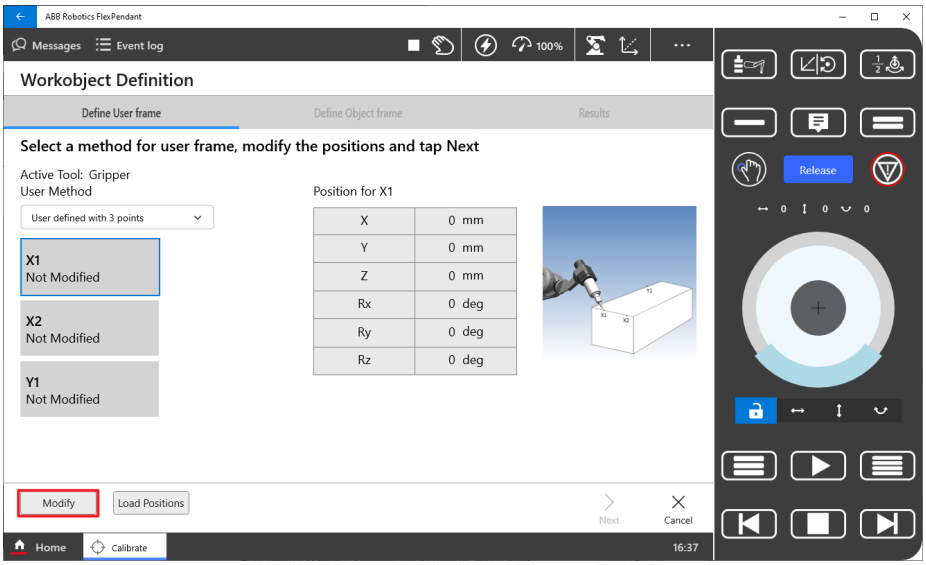
次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2100000383

8 変更を押します。



xx2100000384

9 **X2** を選択します。 TCP を x 軸方向に移動します。 ロボットの TCP で x 軸上の場所を示します。

10 変更を押します。

11 **Y1** を選択します。ロボットのTCPで、y軸上の正のポジションにあるポイントをポイントします。[ポジションの変更]を押します。

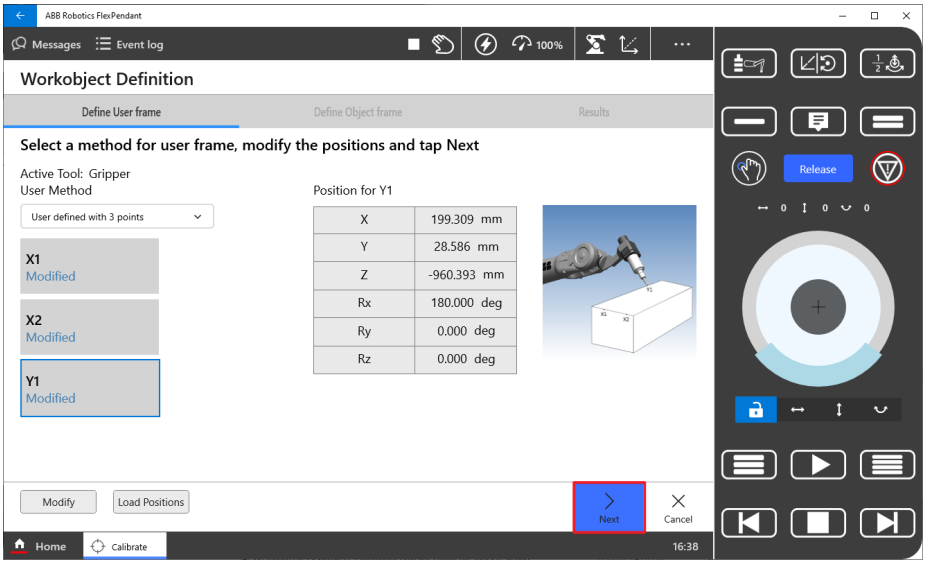
12 変更を押します。

13 **Next**[次へ]をタップします。

次のページに続く

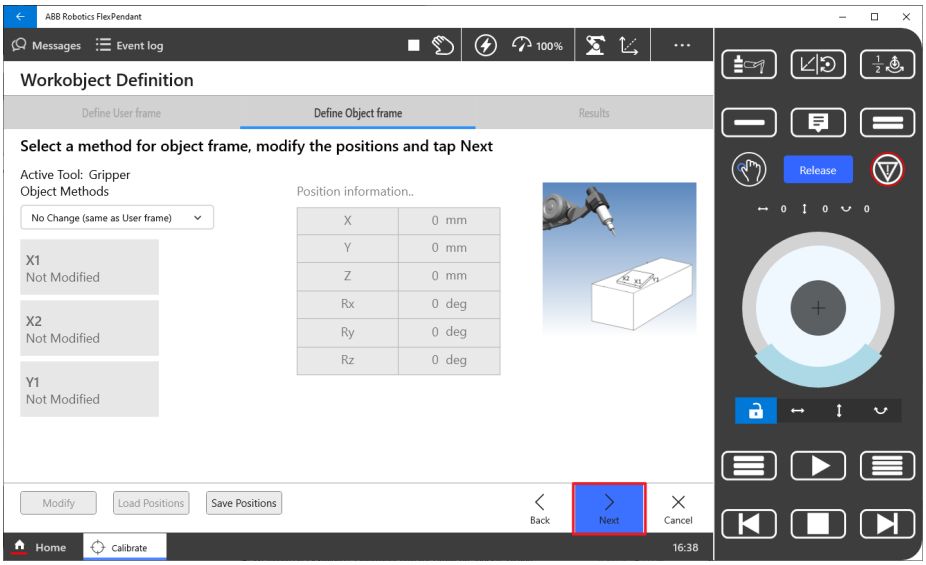
© Copyright 2025 ABB. All rights reserved.

4.4.6 インデックスワークエリアのキャリブレーション
続き



xx2100000385

14 物体フレームの定義 ウィンドウで、次へ をタップします。

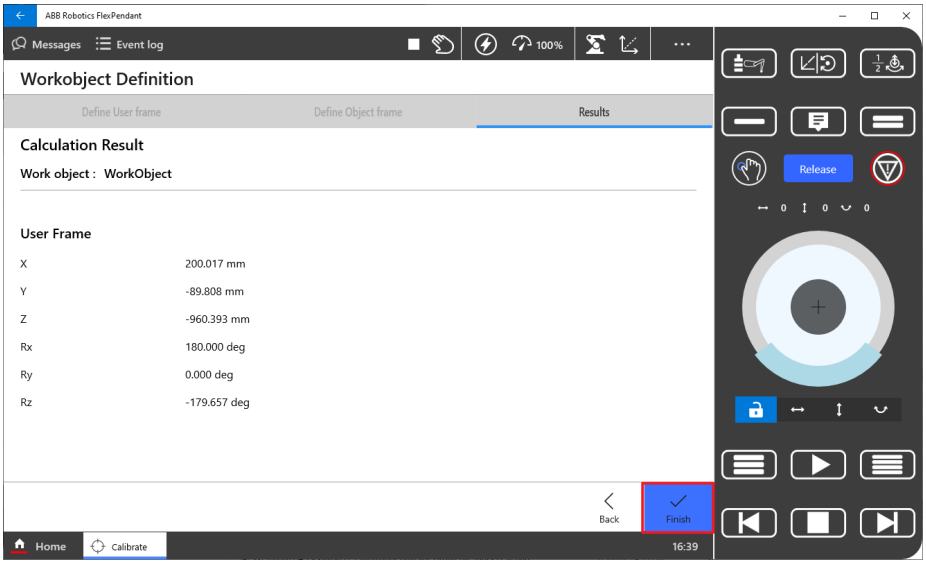


xx2100000386

15 ユーザー フレーム計算の表示された平均誤差と最大誤差が許容範囲内かどうかを確認します。 推定誤差が許容できる場合は、完了 をタップして確認し、新しいユーザー フレームを保存します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

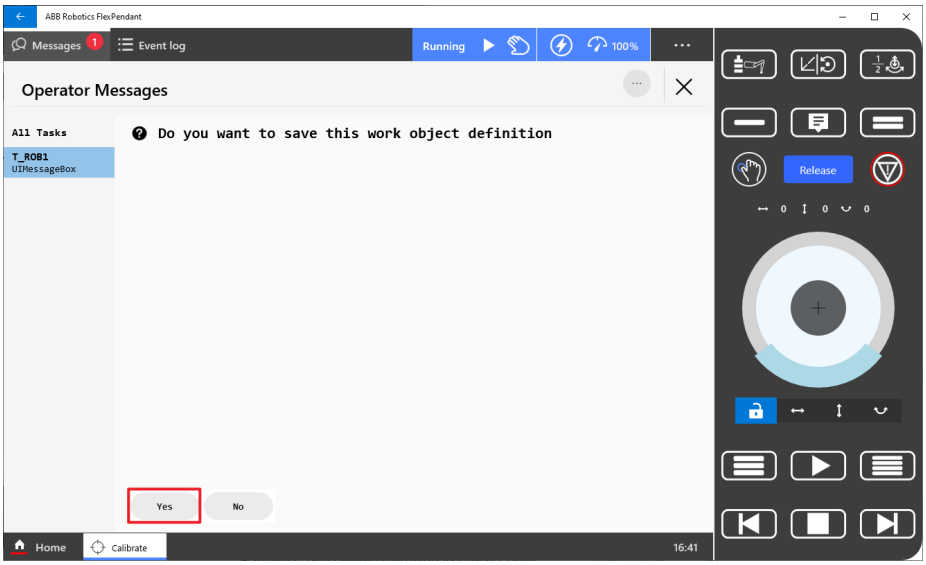


xx2100000391

16 コントローラのモーターへの有効化デバイス ボタンを有効にします。

17 再生 をクリックします。

18 Do you want to save this work object definition[「この作業物体定義を保存しますか?]という質問に対してはいをクリックします。

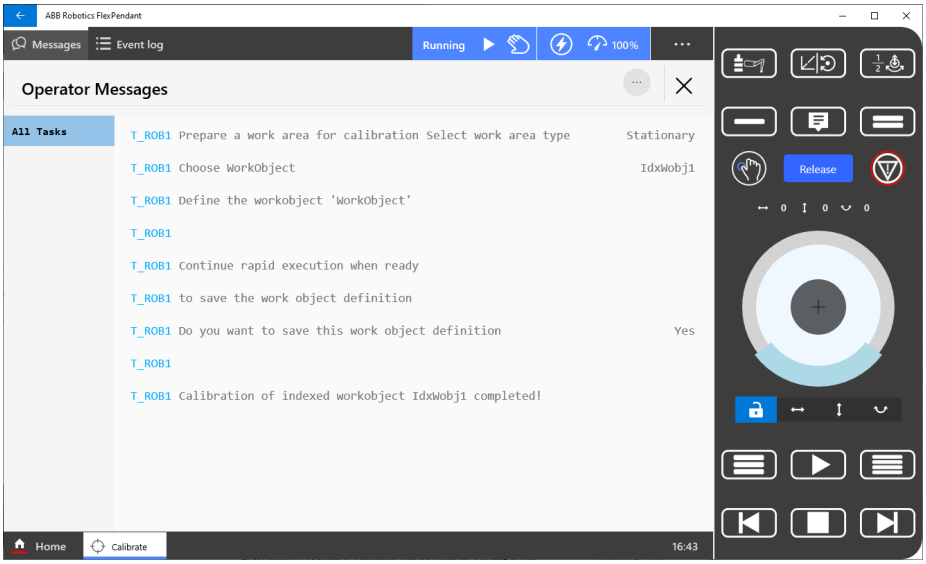


xx2100000393

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.6 インデックスワークエリアのキャリブレーション
続き



xx2100000394

19 定義は、ppaUserシステムモジュールにあるRAPIDデータアレイNonCnvWOData に保存されます。

操作方法(IRC5)

1 キャリブレートするワークオブジェクトを選択します。
   - FlexPendant プログラム エディタ で、プログラム ppacal.prg(DSQC 377)/ PrepareCalib.prg(DSQC 2000) をロードします。
   - ロードしたRAPIDを起動します
     - キャリブレーションタイプ、Fixed/Indexedを選択します。
     - ワークオブジェクトを選択します。例：IdxWobj1
     - メッセージDEFINE CURRENT WORKOBJECTを待ちます。

> **注記**
>
> キャリブレーションが完了するまで、プログラムポインタを動かしません。動かすと、キャリブレーションが正しく保存されません。

2 FlexPendant [ジョグ]ウィンドウで、[ワークオブジェクト]をタップして選択します。次に、[編集]をタップして[定義]を選択します。

3 オブジェクト方法の[変更なし]を選択します。ユーザー方法、[3ポイント]を選択します。

4 ユーザー ポイント X 1 を選択します。ロボットの TCP を使用して、起点に近い x 軸上にあるポイントをポイントします。[ポジションの変更] を押します。

5 [ユーザーポイントX 2]を選択します。TCPをx軸方向に距離を空けて動かします。ロボットのTCPで、x軸上にあるポイントをポイントします。[Modify Position（ポジションの変更）]を押します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

6 [ユーザーポイントY 1]を選択します。ロボットのTCPで、y軸上の正のポジションにあるポイントをポイントします。[Modify Position（ポジションの変更）]を押します。
7 ［OK］をタッチします。
8 RAPIDプログラムを再起動して（PPを動かさずに）、選択したワークオブジェクト定義を保存します。

   定義は、ppaUserシステムモジュールにあるRAPIDデータアレイNonCnvWODataに保存されます。

© Copyright 2025 ABB. All rights reserved.

4.4.7 コンベアのキャリブレーションの検証

概要

キャリブレーションの検証は検証用紙を使用します。用紙には、ターゲットとして使用できるモデルが含まれています。ここでは、ベースフレームのキャリブレーションに使用するものと同じツールを使用します。

キャリブレーション検証用紙のファイルは、PickMaster インストールフォルダと CD の両方にあります。

良好なキャリブレーションを達成するには、カメラのキャリブレーション調整とベースフレームのキャリブレーション調整を複数回行います。この作業を繰り返し実行すると、良好な精度が得られます。

> **注記**
>
> キャリブレーション調整は、微調整に使用してください。誤差が大きい場合は、ラインのキャリブレーションそのものを再実行する必要があります。

カメラとベースフレームのキャリブレーションの調整

下記の手順に従って、カメラとベースフレームのキャリブレーションを調整してください.

1. キャリブレーション検証用紙を、カメラの下のコンベヤ上に置きます。検証用紙の中心をカメラ視野の中央にセットします。用紙をできるだけ正確にコンベヤの進行方向に合わせます。
2. キャリブレーション検証用紙上のオブジェクトの1つをモデルとして使用します。カメラのキャリブレーション ページ *282*を参照してください。
3. グリップポジションをモデルの中心にセットします。
4. ロボットがホールをどのように配置しているかを調べる[3] カメラのキャリブレーションまたはベース フレームのキャリブレーションで発生する可能性のあるエラーを調整します。



xx0900000649

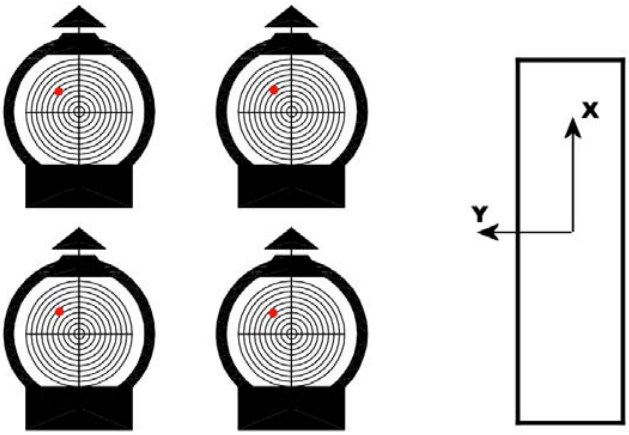
ホールが回転している場合[4] 物体の中心と比較して大きすぎると、把握の精度に影響します。次に、カメラを再調整します。

3 用紙上のポイント。

4 ホールから中心までの角度と X 軸の間の角度。

次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx0900000650

ホールが物体の中心からずれすぎて、把握の精度に影響する場合は、コンベアのベース フレームを再調整します。

© Copyright 2025 ABB. All rights reserved.

4.4.8 カメラのキャリブレーション

概要

概要

カメラのキャリブレーションでは、カメラ座標とロボットベースフレームまたはインデックスワークエリアの関連付けを定義します。カメラを使用する場合、カメラのキャリブレーションはベースフレームキャリブレーションより先に実行してください。これは、カメラのキャリブレーション起点が両方のキャリブレーションの共通の起点となるためです。カメラのキャリブレーションが完了すると、ベースフレームキャリブレーションが実行される時に起点が保存され、ユーザーはこの起点をグラフィック表示できます.

> **注記**
>
> ファイアウォールまたはウイルス対策ソフトウェアがインストールされている場合は、`pickmasteru.exe`、`sshd.exe`、および `visionclient.exe` をホワイト リストに追加します。
>
> そうしないと、PickMaster PowerPac はRuntime に接続できず、ビジョン機能が正常に動作しません。

チェッカーボードキャリブレーション

カメラキャリブレーション法はチェッカーボードキャリブレーションと呼ばれています。このキャリブレーションは2 つのステップで実行されます。まず画像全体を分析して歪み補正し、次に結果画像の領域を定義します。

アルゴリズムでは画像の中心のスケールを使用します。これにより、すべてのマスがもとの画像の中心のマスと同じサイズになります。

マルチビューキャリブレーション

カメラは1つまたは複数の画像を使用してキャリブレーションされます。複 数の画像を使用した場合は、カメラのポジションが計算されます。この情報は、製品高さを決定する必要のある 2.5Dアプリケーションと、2D アプリケーションの視差エラー補正の両方に使用されます。さまざまな高さの製品の扱い（2.5D ビジョン） ページ 371を参照してください.

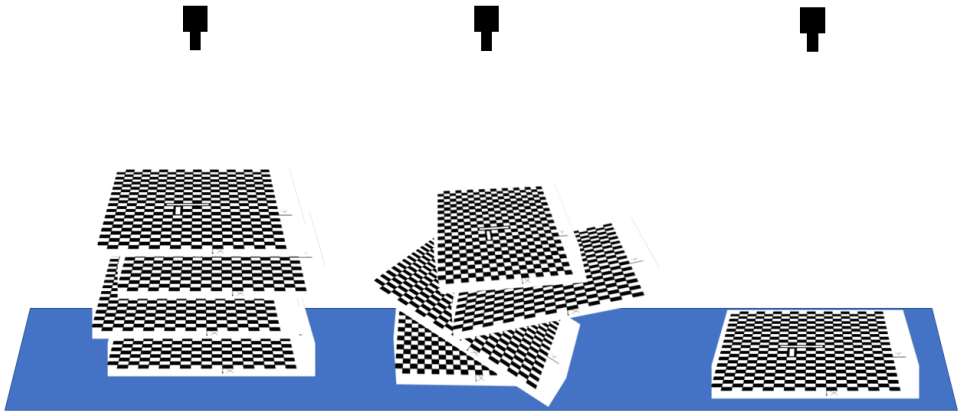
マルチビューキャリブレーションの精度は、入力画像の枚数が増えるほど高くなります。

次の仕様の画像を少なくとも 10 ~ 15 枚使用してください。

- キャリブレーションパターンがカメラの下で平らになっている高さの異なる画像のセット（3 ~ 5枚）。
- キャリブレーションパターンの傾きや高さが異なる画像のセットです(最低3枚ですが、より多くの画像でより良い結果が得られます)。
- キャリブレーションパターンをコンベア面に下向きに置きます。これが原点画像となるはずです。

次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2300000276

注記

キャリブレーションプレートの画像を平行に複数枚使用しても、精度は上がりません。

前提条件

カメラのキャリブレーションは、印刷する必要のあるキャリブレーション ペーパーを使用して行われます。キャリブレーション ペーパーは、`C:\Program Files (x86)\ABB\PickMaster Twin 3\Help\CallibrationPapers` にあります。

印刷イメージはハイコントラストで、反射性（高光沢）を有さない。校正紙はカメラの全視野の対象であることを確認します。正方形が比例していることを定規で確認します。正方形の幅および長さを正しく入力します。精度は最低でもmmの 10 分の 1 である必要がある。正確な値を得るには、校正シートの各方向の全長を測定し、マス目の数で割ること。

キャリブレーション用紙に適切な照明があたっており、影がないことを確認してください。

コンベアを使用する場合、校正紙の X 軸をコンベヤの進行方向に合わせて配置すること。

カメラのキャリブレーション

カメラキャリブレーションは、**カメラノキャリブレーションウィンドウ**が使用できます。これにより、キャリブレーションを作成、編集、インポート、およびエクスポートすることができます。

下記の手順に従ってカメラのキャリブレーションを行ってください.

1. ツリー ビューの**カメラ**でカメラを右クリックし、**キャリブレーション** を選択します。

   カメラのキャリブレーション ウィンドウが開きます。
2. カメラの下のコンベアに検証用紙を置きます。 用紙をコンベアにできるだけ正確に合わせます。

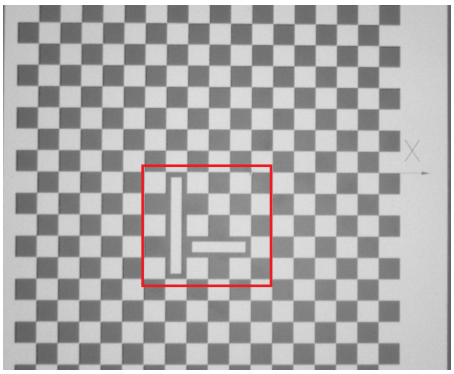
次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.8 カメラのキャリブレーション
続き

ヒント

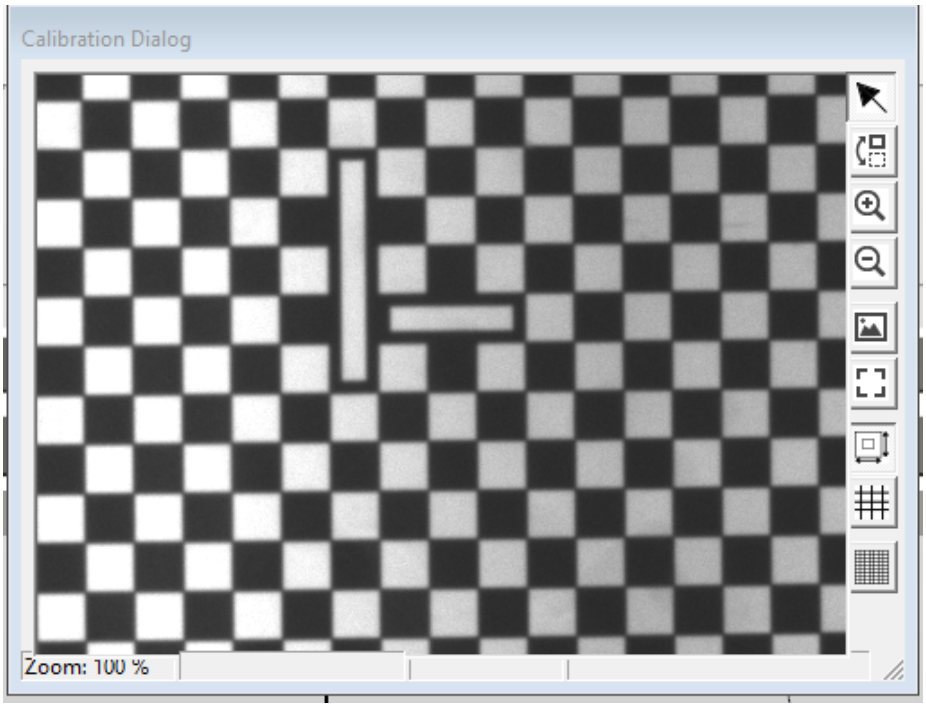
キャリブレーションに使用するすべての画像は、キャリブレーション ボードの中央部分が常に画像内に収まるように撮影する必要があります。



xx2300001516

3 リストからデフォルトキャリブレーションを選択し、［編集］をクリックします。

カメラ キャリブレーション機能 と キャリブレーション ダイアログが開きます。



xx1800001469

次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2100001642

4 ［イメージ]で［ライブ］をクリックし、 動画として取得します。または[取得］をクリックして、現在の静止画像を取得します。ファイル内の画像を使用する場合、または現在の画像を保存する場合には、[インポート］ または［エクスポート］をクリックします。

5 シングルビューキャリブレーションの場合：キャリブレーションプレートが定ポジションにある時に、イメージを取得し、 ［キャリブレーション画像]の［原点設定］をクリック します。これにより、画像が保存され、その画像に起点のマークを付けます（このイメージの起点はカメラ座標系の原点となります）。

6 マルチビューキャリブレーションの場合：複数のイメージでカメラのキャリブレーションを行うとき、カメラのキャリブレーションの終了後にも起点イメージが所定のポジションにあることが重要です。起点イメージはロボットの座標系の定義に使用するからです。

これは 2 つ方法があります。1 つ目の方法では、まず異なる角度と高さの画像を 3 つ以上取得し、（[取得］と［画像の追加］をクリックします）、次に起点イメージを取得します（［取得］ と［原点設定］をクリックします）。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

この時、キャリブレーション用紙をワークオブジェクト/ ベースフレームのキャリブレーションのために残しておきます。

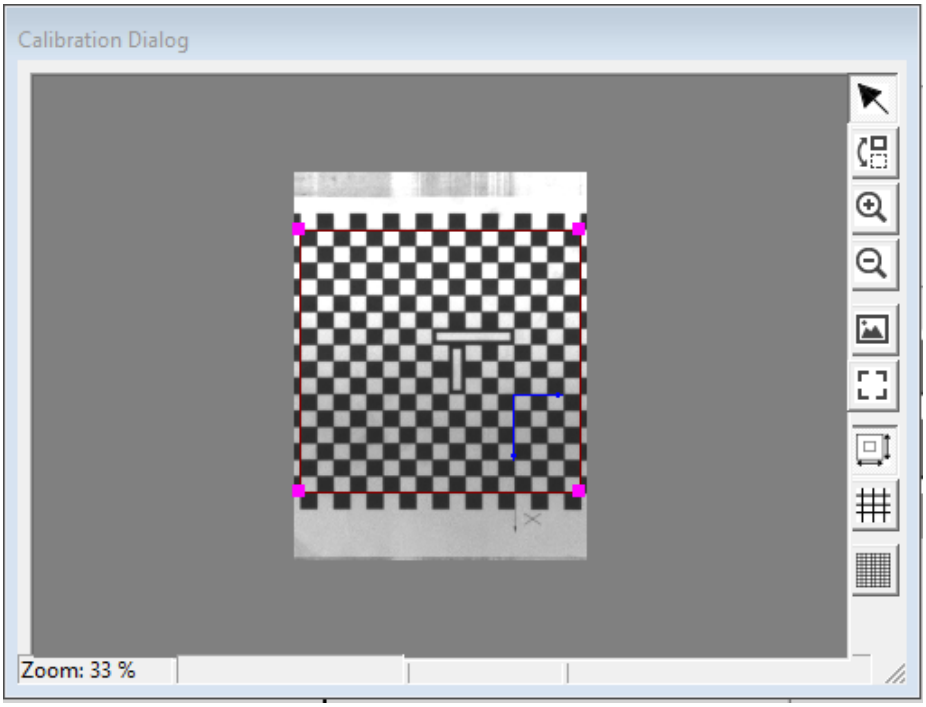
2つ目の方法では、まったく同一のグリッドピッチを持つ2枚のキャリブレーション用紙を使用します。1枚のキャリブレーション用紙を、カメラの原点にするポジションに置きます。画像を 1 つ取得し、［原点設定］をクリックします。この 1 枚目のキャリブレーション用紙を残したまま、異なる角度と高さの 2 番目のキャリブレーション用紙を使って異なる角度と高さの画像を 3 つ以上取得し、［追加］をクリックしてリストに保存します。

ヒント

マルチビューキャリブレーションでは、空間情報が自動計算されます。

7 ［キャリブレーション］で［キャリブレート］をクリックしてキャリブレーションを開始します。

画像が分析され、指定のパラメータでキャリブレーションが実行されます。正しい画像が、最終的なイメージエリアの定義に使用される枠と共に表示されます。キャリブレーションは領域の設定が実行されるまで完了しません。



xx1800001703

8 枠を任意の範囲に調節し、［領域の設定］をクリックして得られた画像サイズを定義します。

これでキャリブレーションが完了し、その結果が［キャリブレーションの結果］に表示されます。キャリブレーション結果 ページ 287を参照してください。

9 必要に応じて下記の項目を行います：

- ［キャリブレーション画像］をクリックしてカメラのキャリブレーションに使用した元の画像を表示します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

- ［ワープライブ］をクリックして 取得および修正した動画として表示します。
- ［ワープイメージ］をクリックして 現在の静止画像として表示します。

10 必要に応じて下記の項目を行います：

- ［交点の表示］をクリックしてキャリブレーション実行中に使用されたチェッカーボードの交点を表示します。これはキャリブレーション画像の中だけで表示されます。
- ［原点の表示］をクリックして 得られた座標系の原点を表示します。

11 ［OK］をクリックします。

ヒント

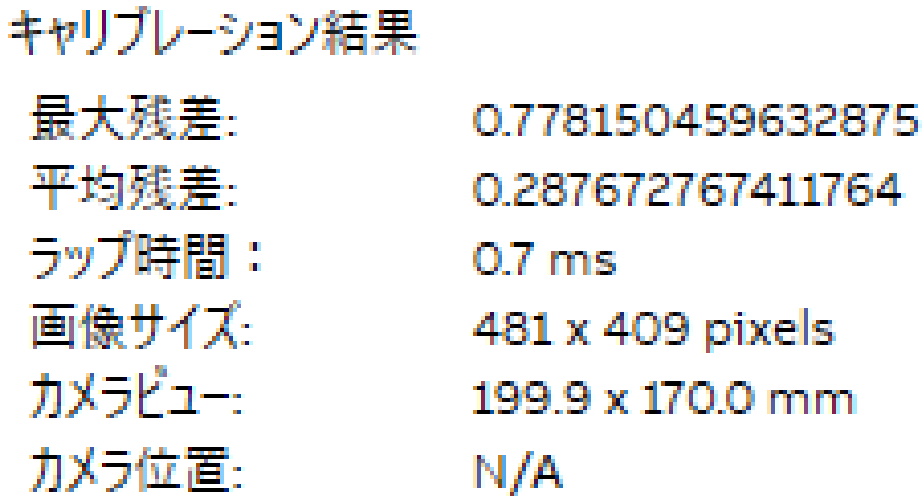
コンベヤの場合、ベースフレームのキャリブレーションが完了するまで、キャリブレーション用紙をそのまま残して置きます。

注記

カメラ キャリブレーションはエクスポートまたはインポートすることができます。エクスポートされたファイルは .pmcalib 形式で保存されます。また、特定のキャリブレーションに使用されたイメージを保存するために、カメラ キャリブレーション ウィンドウからイメージをエクスポートすることも可能です。

キャリブレーション結果

シングルビュー キャリブレーション：



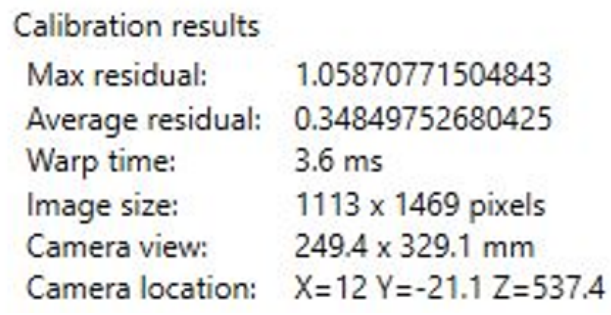
xx2100001646

マルチビュー キャリブレーション: :

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.8 カメラのキャリブレーション
続き



xx2300001517

結果	説明
Max residual（最大誤差）	キャリブレーションの最大誤差です.
Average residual（平均誤差）	キャリブレーションの平均誤差です.
Warp time（補正時間）	画像の修正に必要な時間です。画像の分析に必要な合計時間を計算する際に、この時間が考慮されます。
Image size（イメージサイズ）	修正された画像のサイズがピクセルで表示されます。
Camera view（カメラビュー）	新しいキャリブレーションで計算されたカメラ視野のサイズです。
Camera location（カメラポジション）	起点キャリブレーション用紙の原点に対するカメラのポジションです。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.8.1 ライブ画像の表示

ライブ画像

生産の実行中に、各カメラからの画像を表示することができます。

> **注記**
>
> Runtime 画像の表示には多くの処理能力が必要なため、複雑なビジョン モデルを使用する場合は長時間使用しないでください。

ライブ画像の表示

画像を表示するには、**コントロール** をクリックします。 カメラ画像は **ビジョン** タブに表示されます。

検出されたアイテムは、ビジョンモデルとの比較により許可（緑）あるいは拒否（青）のどちらかで表示されます。ビジョンモデル化 ページ *293*を参照してください.

© Copyright 2025 ABB. All rights reserved.

4.4.8.2 詳細なビジョン情報

詳細なビジョン情報

［詳細なビジョン情報］ダイアログボックスでは、ライブ画像の情報よりもさらに詳細な情報を表示することができます。 このダイアログボックスは、画像のバッファ、およびビジョンモデルの認識情報を保持します。

画像の順序はバッファに記録され、その後個々に分析することができます.

記録する際、画像は先入れ、先出し方式でバッファに保存され、最後の画像がダイアログに表示されます。

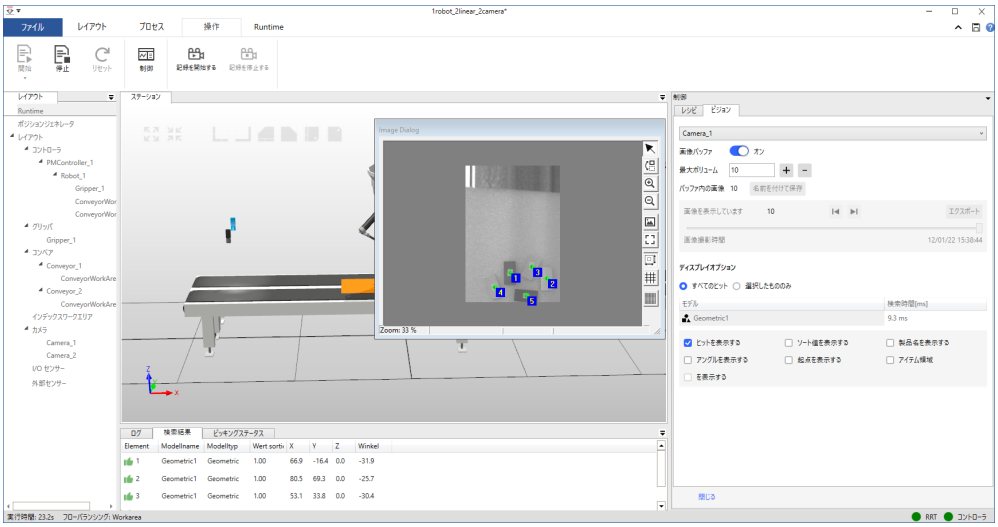
画像バッファ機能をオフにすると、画像は追加されなくなり、バッファ内の画像を分析できます。 現在のバッファ内の画像をファイルに保存して、ビジョン アナライザ プログラムで後で分析できるようにします。*Vision Analyzer ページ 291* を参照してください。

ドロップダウン リストから別のカメラに切り替えることができます。

バッファの最大サイズは、コンピュータの RAM メモリによって異なります。

図

コントロールの下の ビジョンタブをクリックして、ダイアログを開きます。 デフォルトでは、録画状態が有効になっており、バッファの最大ボリュームは 10 画像に設定されています。



xx2100001638

イメージ バッファ	「記録」と「一時停止」を切り替えるために使用します。名前を付けて保存するをクリックして、バッファ内のすべての画像を .pmv ファイルに保存します. 記録が一時停止しているときに画像バッファをステップスルーします。 LEFT または RIGHT ARROW ボタンもステップに使用できます。エクスポート をクリックして、現在の画像をファイル (.bmp 形式) に保存します。
オプションの表示	表示するビジョンモデルを、すべてまたは個々に選択します。さらに、イメージ内に表示するその他の項目を設定します。この設定は記録と一時停止の両方に有効です。
検索結果	一番下にあるリストビューには、すべての検出されたアイテムの情報が表示されます。個々のモデルを選択すると、選択したモデルのタイプによって表示される内容が変化します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

画像ダイアログ	［パン］および［ズーム］ボタンを使用して、画像をより詳しく分析することもできます.

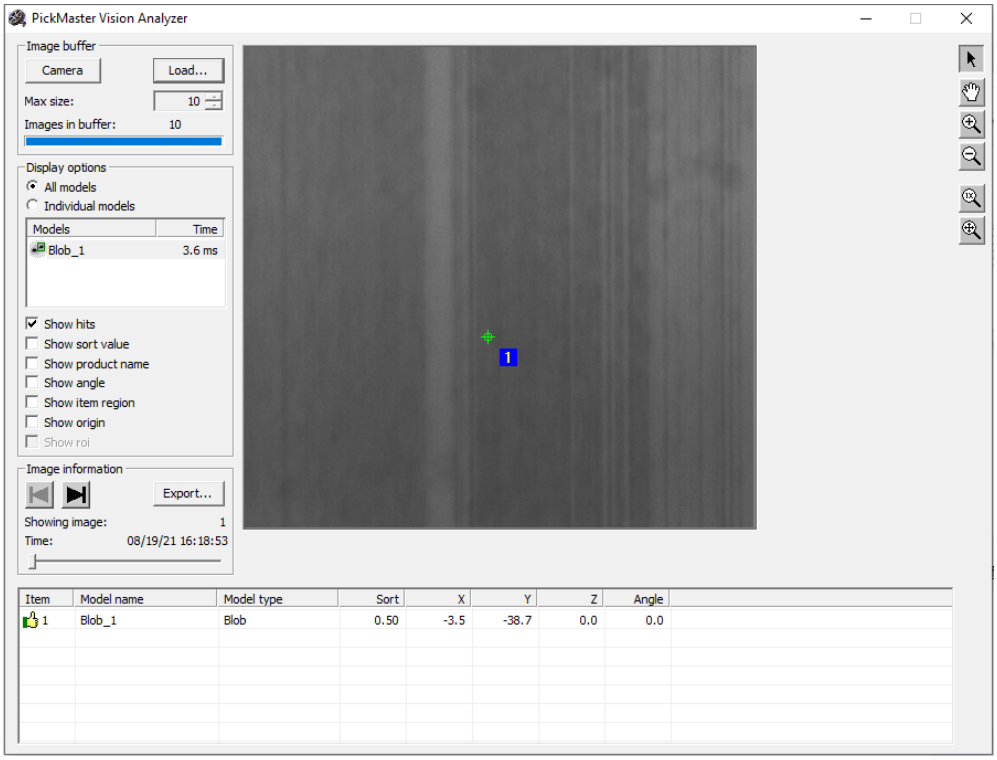
注記

同じイメージ内にあるオーバーラップアイテム領域だけがオーバーラップとマークされ、連続するイメージのアイテム領域とオーバーラップしている領域のアイテムにロボットはアクセスしません。

Vision Analyzer

［詳細なビジョン情報］ウィンドウに記録された画像バッファは、 .pmv ファイルとして保存できます。これらのファイルは、PickMaster Vision Analizer という名前のプログラムで表示できます。

PickMaster Twin Client のインストール フォルダまたは Windows の [スタート] メニューから、ビジョン アナライザ を起動します。



xx2100001831

［Load（ロード）］をクリックして、 .pmvファイルを開きます。

［Camera（カメラ）］をクリックして、 画像を撮影したカメラの詳細な情報を表示します。

ビジョンのその他の設定は、詳細なビジョン情報の設定と同じです。

© Copyright 2025 ABB. All rights reserved.

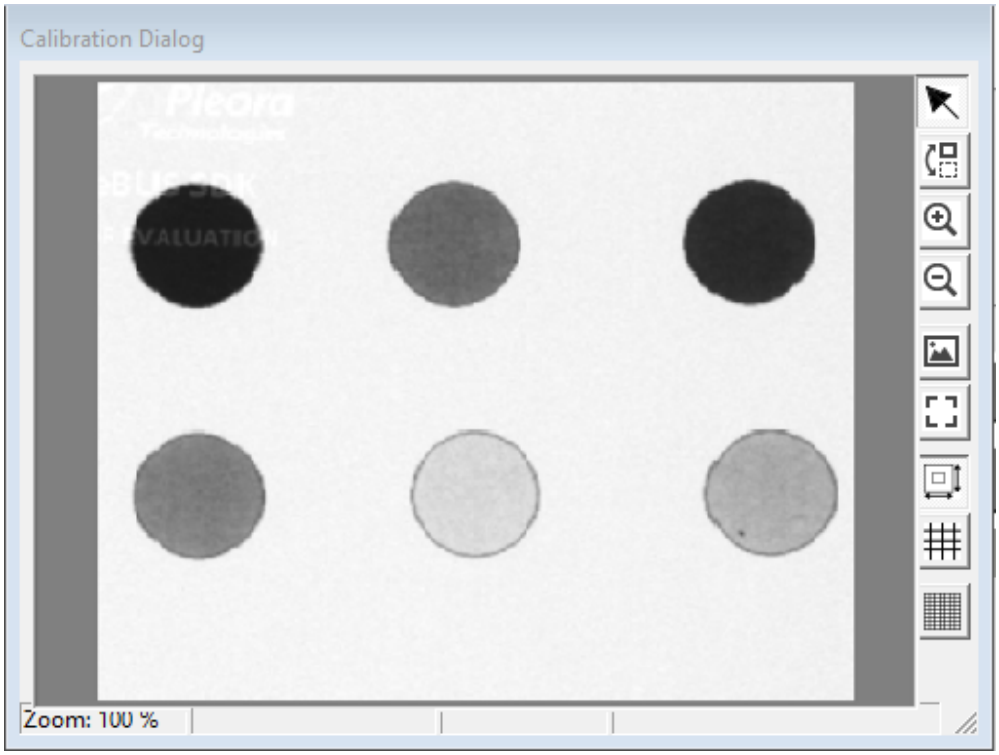
4.4.8.3 イメージウィンドウ

イメージウィンドウ

カメラやビジョンモデルを設定する際、カメラ画像は別のウィンドウに表示されます。画像ウィンドウはサイズ変更が可能で、表示されている画像を簡単にズームしたりパンできるツールがあります。ツールによっては、マウスポインタの形状が変化します.

キーボードとマウスを使用してズーム操作を行うには、マウスのポインタを画像上に置いてCtrl ボタンを押し、マウスホイールを回転させます.

現在のズームレベルとマウスポインタのワールド座標は、イメージの下にあるステータスバーに表示されます。ライブイメージが表示されている場合は、ステータスバーに現在のフレームレートも表示されます.



xx2100001521

© Copyright 2025 ABB. All rights reserved.

4.4.9 ビジョンモデルの追加

4.4.9.1 ビジョンモデル化

ビジョンモデル化について

ソリューションでモデルを生成するために使用できる 3 つの異なるツールがあります。 3 つのツールは次の通りです。

- *Geometric*はパターン認識ツールです。*Geometric*モデルの設定 ページ *296* を参照してください。
- *Blob*（輪郭）：イメージ内の二次元形状を検知します。「*Blob*モデルの構成 ページ *305*」を参照してください。
- 検査ツール (検査 II) は、*Geometric*、Blob、ヒストグラム、キャリパーを組み合わせてモデルを生成することを可能にします。検査モデルの構成 ページ *312* を参照してください。

注記

ビジョン モデリングは、ソフトウェアがReal Runtimeに接続されている場合にのみ作成または編集できます。

注記

PickMaster 3 ソリューションおよびその他のPickMaster PowerPacソリューションからビジョン モデルをインポートできます。

既存のビジョン モデルのインポート

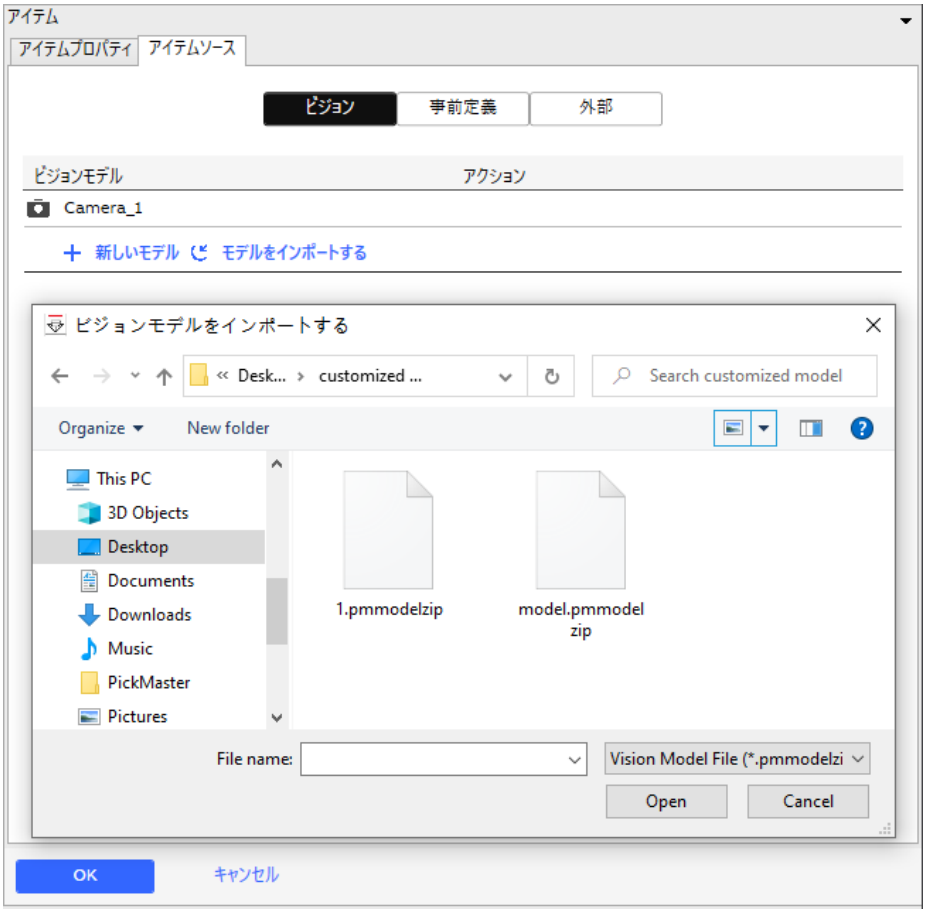
この手順を使用して、既存のビジョン モデルをインポートします。

1. ツリー ビューの**プロセス**で 1 つの**アイテム**を右クリックし、**設定** を選択します。
   アイテム設定ウィンドウが開きます。
2. **アイテム ソース** タブをクリックして選択します。
3. **アイテム ソース** ダイアログで、必要なカメラの下にある **モデルのインポート** をクリックします。
   ビジョン モデルのインポート ウィンドウが開きます。
4. 有効なビジョン モデル (`.pmmodel` または `.pmmodelzip`) を選択し、**開く** をクリックします。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.9.1 ビジョンモデル化
続き



xx2200000479

5 ［OK］をクリックします。

アイテムの分類

ビジョン モデルによって識別されたアイテムは、承認済みまたは拒否済みに分類できます。これら2つのタイプは、異なるワークエリアに分散され、RAPID プログラムからアクセス可能な異なるアイテム タイプ値が割り当てられます。アイテムの分類は、*Geometric*、*Blob*、および検査ツールによって行うことができます。

アイテムターゲットのビジョンモデルパラメータ

ビジョン モデルによって識別されるアイテム ターゲットは、コンポーネント `Val1`、`Val2`、`Val3`、`Val4`、および `Val5` に最大 5 つのビジョン モデル パラメータの選択を格納できます。 これらのパラメータは、RAPID プログラムでアクセスできます。

「検査モデル」によって識別されるアイテムターゲットは、アライメントモデルと補助検査モデルからパラメータを選択して保存できます。

一部のビジョンパラメータでは、各ビジョンモデルに「ターゲットストレージ」を選択できます。

外部ビジョンモデル

この機能は、次のバージョンのために予約されています。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

関連情報

*Geometric*モデルの設定 ページ *296*。
*Blob*モデルの構成 ページ *305*。
検査モデルの構成 ページ *312*。

© Copyright 2025 ABB. All rights reserved.

4.4.9.2 Geometricモデルの設定

Geometricモデルの概要

*Geometric*はパターン位置検索テクノロジーです。このツールは以下を測定します。

- パターンのポジション。
- 本来教示されたパターンに相対するサイズ。
- 本来教示されたパターンに相対するアングル

Geometric は、効率的かつ正確に回転または拡大縮小できないピクセル グリッド表現に基づいていないため、他のパターン位置特定テクノロジーとは異なります。代わりに、*Geometric* は、パターン マッチングのために迅速かつ正確に変換できるフィーチャ ベースの表現を使用します。

パターンを作成する際は、次の点を考慮しなくてはなりませ ん。

- 一貫性のあるフィーチャーを持つ、代表的なパターンを選択します。不必要なフィーチャーおよびイメージノイズを削減します。重要なフィーチャーのみを教示します。必要に応じて、イメージをエクスポートし、外部プログラムを使用してノイズを消します。
- パターンが大きいと、実行時に計算される境界ポイントもより多く含まれるため、精度も高くなります。
- パターンの外側エッジでは、頻繁に現れるフィーチャー がより重要になります。

モデルの分類は、使用可能な機能*Inspection I*（検査 *I*）を使用して実行できます。モデルは許可あるいは拒否のいずれかとして定義されます。アイテムプロパティタブ ページ *142*を参照してください。

パーツのグレースケールトーンが同程度の画像のコントラストを高くするには、カラーフィルタを使用します。カラービジョンの使用 ページ*324*を参照してください。

効率的なモデルを作成するために調整できるパラメータがいくつかあります。設定は **図形モデル** タブ ビューで行い、結果は **検索結果** ウィンドウと **画像ダイアログ** に表示されます。

アルゴリズム

PickMaster PowerPacGeometric モデルは、2 つのパターン位置特定アルゴリズムをサポートします。

- PatMax
- PatQuick

PatMax は PatQuick よりも高い精度を提供しますが、PatMax の実行にはより長い時間がかかります。PatMax アルゴリズムは、追加のスコア情報を返すこともできます。一方、PatQuick は、詳細な変動に対してより寛容です。

使用可能な設定パラメータは、PatMax と PatQuick で異なります。

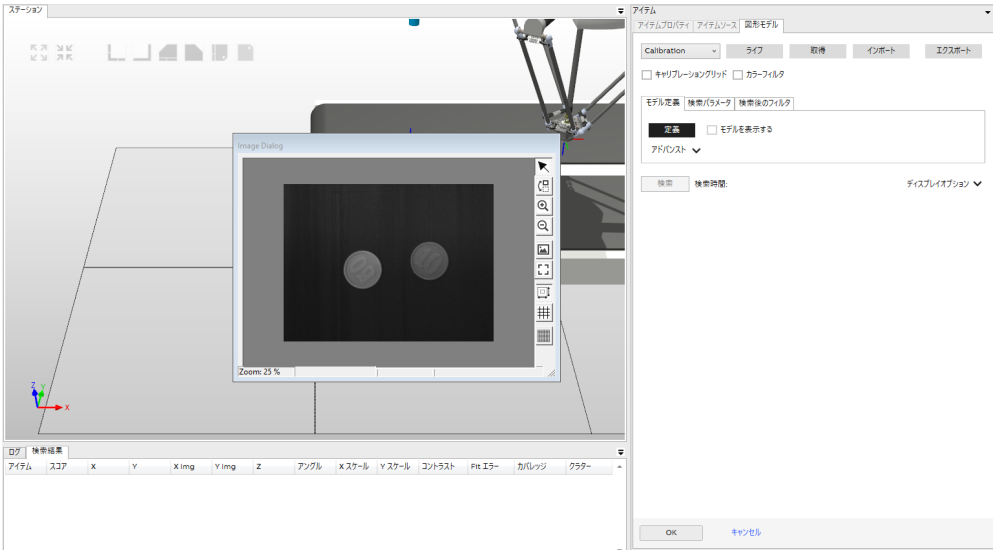
アルゴリズム	使用可能な設定パラメータ
PatMax	すべての設定パラメータ

次のページに続く

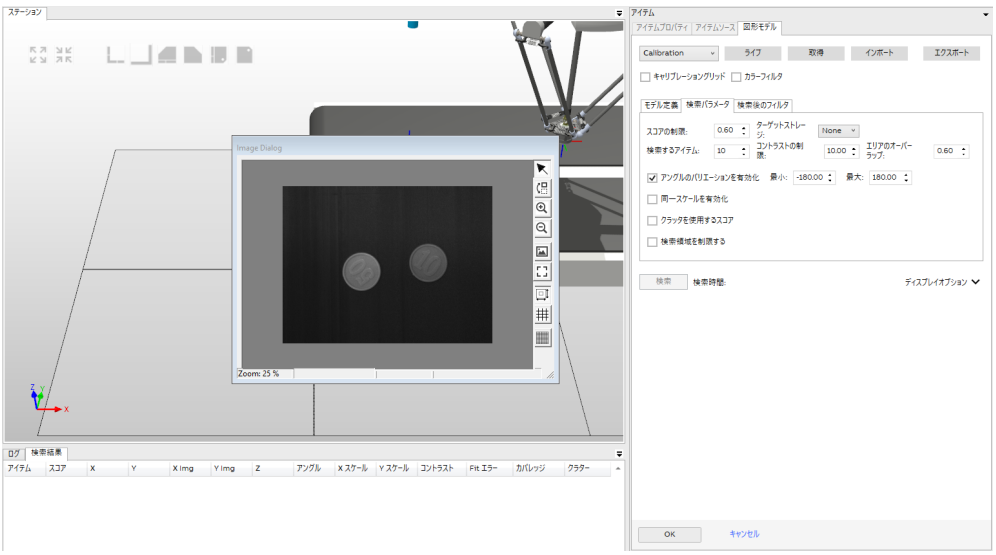
© Copyright 2025 ABB. All rights reserved.

アルゴリズム	使用可能な設定パラメータ
PatQuick	次の設定パラメータを除く: • 検索パラメータ -> クラッターを使用したスコア • ポスト検索フィルタ -> フィット エラー • ポスト検索フィルタ -> カバレッジ • ポスト検索フィルタ -> クラッター • ポスト検索フィルタ -> ターゲット ストレージ • 表示オプション -> 一致情報 PatQuick アルゴリズムを選択すると、使用できないパラメータはデフォルト値としてブロックされます。

イラスト図形モデル設定



xx2100001647

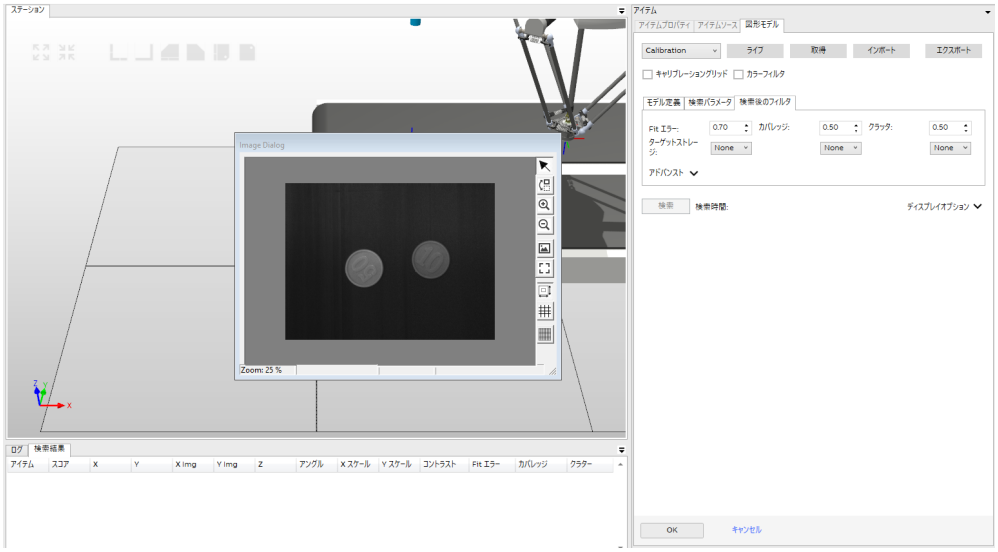


xx2100001648

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.9.2 Geometricモデルの設定
続き



xx2100001649

Geometricモデルの設定

この手順を使用して、Geometricモデルを設定します。

1 ツリー ビューのプロセスで 1 つのアイテムを右クリックし、設定 を選択します。

アイテム設定ウィンドウが開きます。

2 アイテム ソース タブをクリックして選択します。

3 アイテム ソース ダイアログで、必要なカメラの下にある 新しいモデル をクリックし、図形 を選択します。

画像ダイアログと図形ダイアログが開きます。

4 Geometric モデル で、ライブ、取得、またはインポートをクリックして画像を取得します。キャリブレーション リストから、カメラ キャリブレーションで設定したキャリブレーションを選択します。

Geometric アルゴリズムで、このモデルに優先するアルゴリズムを選択します。使用可能なアルゴリズムの詳細については、アルゴリズム ページ296を参照してください。

座標系の補助線を表示するには、キャリブレーション グリッド チェックボックスをオンにします。補助線はマウスで移動でき、パターンのトレーニングを容易にします。

5 カラー フィルタリングを使用する必要がある場合は、カラー フィルタ チェックボックスを選択してフィルターを有効にします。 カラー フィルタ タブでフィルタ パラメータを設定します。カラービジョンの使用 ページ 324 を参照してください。

6 ［Model definition（モデルの定義）］パートで、カメラの前のイメージ またはインポートしたイメージを使用して、パターン用のモデルを定義します。選択したキャリブレーションが使用されます。

次のページに続く

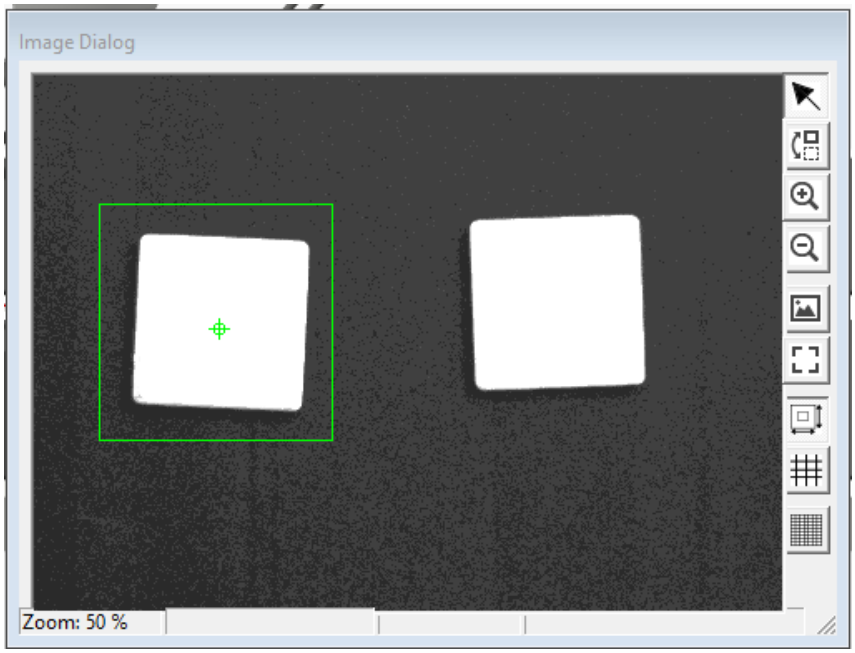
© Copyright 2025 ABB. All rights reserved.

注記

ビジョンモデルをインポートする場合は、モデル構成を入力し、キャリブレーション ドロップダウンメニューで、どのキャリブレーションを使用するかを再度選択する必要があります。これは、定義済みのキャリブレーションが 1 つのみである場合でも必要です。これを行わないと、その後の操作で「No valid calibration for the *Geometric* model」エラーが発生することがあります。

a アイテムの高さを定義する場合は、アイテムをトレーニングする前に適切な計算方法を選択します。**モデルの高さ**は、トレーニングされたアイテムの基本高さとして使用されます。**Pick Offsett** は、この計算方法でピッキング ポイントのずれを補うために使用されます。 詳細については、さまざまな高さの製品の扱い（2.5D ビジョン） ページ371を参照してください。

b [Define（定義）] をクリックして モデルを定義します。パターンをカバーするように長方形をドラッグし、十字を希望のピック / プレイスポジションへ移動させます。最も高い精度を維持するには、ピック / プレイスポジションが、教示パターンの中心近くに来るように配置します。



xx1800001708

c [Train（教示）] をクリックして パターンを教示します.

d [Show Model（モデルの表示）] を選択して検索イメージにあるティーチモデルのフィーチャーを表示します。

e 必要に応じて、 [Advanced（アドバンス）] をクリックしてさらに多くのモデル設定にアクセスします。

f [粒度の調整] をクリックして、[細かい] および [粗い] ボックスのレベルを定義します。粒度は、パターン内の特徴の検出を決定する影響の半径

次のページに続く

© Copyright 2025 ABB. All rights reserved.

(ピクセル単位) です。*Geometric* は、最初に大きな特徴のみを検索して、検索イメージ内のパターンを見つけます。1 つ以上のパターン インスタンスを見つけた後、より小さな特徴を使用して、トレーニングされたパターンと検索イメージ内のパターン間の正確な変換を決定します。*Geometric* は、パターンをトレーニングするときに計算される粒度の範囲と同じ範囲を使用して、検索イメージ内の特徴を検出します。パターンをトレーニングするときに、粒度パラメータの 細かい と 粗い が自動的に選択され、多くの場合、これらの値が最適です。これらは手動で設定することもできます。下限は 1、上限は 25.5 です。

g フィーチャーが明るい背景上で暗いか、暗い背景上で明るい場合には、[極性を無視] を選択して極性を無視します。

> **注記**
>
> *Geometric* は、製品が暗い背景で明るいか、明るい背景で暗いかを気にしません。これは、背景がグリッドなどの場合に便利です。

h 有機製品などの非線形ゆがみを許容するには、 [Elasticity（弾力性）] の値を大きくします。この値は教示したフィーチャーとマッチしたフィーチャーの間の最大距離を表します（ピクセル単位）。下限は 0 で上限は 25 です。この設定は、形状に差異がある製品の場合に便利です。

7 [Search parameters（検索パラメータ）] パートで、検索手順と分析時間を制限するためのパラメータを設定します。

[Socre Limit（スコアの限界）] は 検出アイテムがティーチモデルに対して、どのくらい合致するかを示します。1のスコアは、完全合致を、0はパターンがまったく合致しないことを示します。定義されたスコアしきい値が高いほど、*Geometric*は検索をより速く実行することができます。

ターゲット ストレージは、Rapid の変数を示します。詳細については、*GetItmTgt - 次のアイテムターゲットを取得する ページ 390*を参照してください。

[Items to Find（検索アイテム数）] は イメージに表示されるアイテムの数です。イメージにこれ以上のアイテムがあっても、*Geometric*はレポートしません。

[Contrast Limit（コントラストの限界）] は イメージで検出される各アイテムの最低のイメージコントラストを定義します。コントラストは、*Geometric* がティーチモデルと検索イメージの検出アイテム間で合致した、すべての境界ポイントに対するグレーレベル値の平均差異です。*Geometric*は、コントラストの限界を超えるコントラスト値を持つアイテムのみを考慮に入れます。

エリア オーバーラップは、画像内の複数のパターンが互いにどの程度大きく重なることができるかを定義します。*Geometric* は、これらのパターンが実際には画像内の同じアイテムを表していると想定します。2 つのパターンがエリア オーバーラップしきい値を超える割合で重なる場合、それらは 1 つのパターンとして扱われます。

角度の変化を有効にする は、アイテムの回転許容範囲を定義します。アイテムが有効範囲を超えて回転している場合、これはビジョンシステムにより破棄されます。デフォルトは +/- 180 度です。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

同一スケールを有効にする は、学習したビジョン モデルとは異なるサイズのヒットを受け入れるしきい値です。 スケール値 1 は、見つかった項目と教えられたビジョンモデルの間に違いがないことを示します。 値が 1 より小さい場合は、モデルが小さいことを示します。

クラッターを使用したスコア[1] は、トレーニング済み視覚モデルに存在しない特徴が検出されたアイテムに含まれる程度の尺度を定義します。デフォルトでは、*Geometric*分析はスコアリング時にクラッターを無視します。つまり、パターンは追加機能の有無にかかわらず同じスコアを受け取ります。このチェックボックスをオンにすると、スコアの計算にクラッターが含まれます。アプリケーションが背景が変化しないアライメント アプリケーションである場合は、**クラッターを使用したスコア** を選択する必要があります。

Limit Search Region（検索エリアの制限）は*Geometric*分析の検索エリアを制限します。このエリア内のオブジェク トのみが検索対象となります。検索エリアが小さいほど、検索時間は短くなります。

> **注記**
>
> **細かい/粗い粒度と比率保持** を組み合わせると、設計時と実行時のスコアにわずかな違いが現れることがあります。 したがって、実行時にモデルをテストして、アイテムが期待どおりに識別されていることを確認する必要があります。

8 **［Post search filters（検索語のフィルタ）］パート内で、** 検索イメージで検出された各パターンに対してスコア値を定義します。

フィットエラーは、トレーニングされたパターンの形状と検索画像で見つかったパターンの形状の差異の尺度です。検索画像で見つかったパターンがトレーニングされたパターンに完全に適合する場合、フィット エラーは 0 になります。これは、PatMax アルゴリズムのモデルにのみ有効です。

カバレッジは、トレーニングされたパターンのすべての部分が検索画像にも存在する程度の尺度です。トレーニングされたパターン全体が検索画像にも存在する場合、カバレッジ スコアは 1 になります。カバレッジ スコアが低いほど、パターンの出現が少ないことを示します。このパラメータは、欠落している特徴を検出するために使用できます。これは、PatMax アルゴリズムのモデルにのみ有効です。

クラッターは、見つかったパターンにトレーニングされたパターンに存在しない特徴がどの程度含まれるかの尺度です。クラッターが 0 の場合、見つかったパターンには余分な特徴が含まれていないことを示します。クラッター スコアが 1 の場合、トレーニングされたパターンのすべての特徴に対して、見つかったパターンに余分な特徴が 1 つあることを示します。クラッターは 1.0 を超えることがあります。これは、PatMax アルゴリズムのモデルにのみ有効です。

ターゲット ストレージは、Rapid の変数を示します。詳細については、*GetItmTgt - 次のアイテムターゲットを取得する ページ 390*を参照してください。これは、PatMax アルゴリズムを使用するモデルにのみ有効です。

さらに多くの設定が必要な場合には、 **［Advanced（アドバンス）］** をクリックして **［Advanced Search Settings（アドバンス検索設定）］** ダイアログを開きます。そこに次の設定が表示されます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.9.2 Geometricモデルの設定
続き



xx2100002333

検査レベル - 検査 I の使用、この検査は、PickMaster PowerPac では検査 *I* とも呼ばれます。 この機能を使用すると、見つかったモデルを 2 つのカテゴリに分類できます。 モデルは、許可されるか拒否されるかのいずれかに分類できます。 許可されたモデルは、拒否されたモデルよりも優れた検索結果を示します。アイテムタイプ番号は、**アイテム** ダイアログで承認モデルと拒否モデルに対して定義されます。*アイテムプロパティタブ ページ 142* を参照してください。 アイテムタイプは RAPID コードで読み取ることができます。*RAPID プログラム ページ 417* を参照してください。

[検査パラメータ] セクションでは、検索パラメータとポスト フィルタに指定された条件を満たすすべてのモデルが分類されます。**検査レベルを使用** を選択して、見つかったアイテムを 2 つのカテゴリに分類するパラメータを定義します。 **検査レベルを使用** が選択されていない場合、検出されたすべてのモデルは許可済みモデルとして分類されます。

［Score（スコア）］、 **[Contrast（コントラスト）]**, **[Coverage（適用範囲）]** では［Inspection Parameter（検査パラメータ）］ボックスで定義された値を超える値を持つアイテムは、許可されたものとして 定義されます。

［Angle（アングル）］および ［Uniform Scale（同一スケール）］では［Inspection Parameter（検査パラメータ）］内の2つの値の間に収まる値は、許可されたものとして定義されます。

［Fit Error（適合エラー）］および ［Clutter（クラッター）］では定義された値に満たない値が、許可されたものとして分類され ます。

[Limit Position Region（ポジション領域の制限）]は*Geometric*分析がイメージ全体に対して行われたかどうかを定義します。この領域内のオブジェクトは、標準として処理されます。この領域外のオブジェクトは破棄されます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

アイテム領域を定義するには、**アイテム領域を使用** チェックボックスを選択して、**領域を定義** をクリックします。 頂点を使用して、見つかった物体の周囲に表示されるポリゴンを調整します。 次に、**教える** をクリックします。 ポリゴンには 2 ~ 16 個の頂点を含めることができます。

9 ［Display options（オプションの表示）］パートで、グラフィックで表示する情報のタイプを選択します。

一致情報には、検索画像内の一致した境界ポイントの品質が表示されます。これは、PatMax アルゴリズムを使用するモデルにのみ有効です。境界ポイントの描画:

- 合致率が低い場合は赤。
- 合致率が中程度の場合は黄。
- 合致率が高い場合は緑。

アイテムのスコア は、画像ウィンドウに、選択されたアイテムのスコアを表示します。 項目領域は、画像ウィンドウに領域を表示します。 赤い領域は重複を示し、対応するヒットは破棄されたと見なされます。

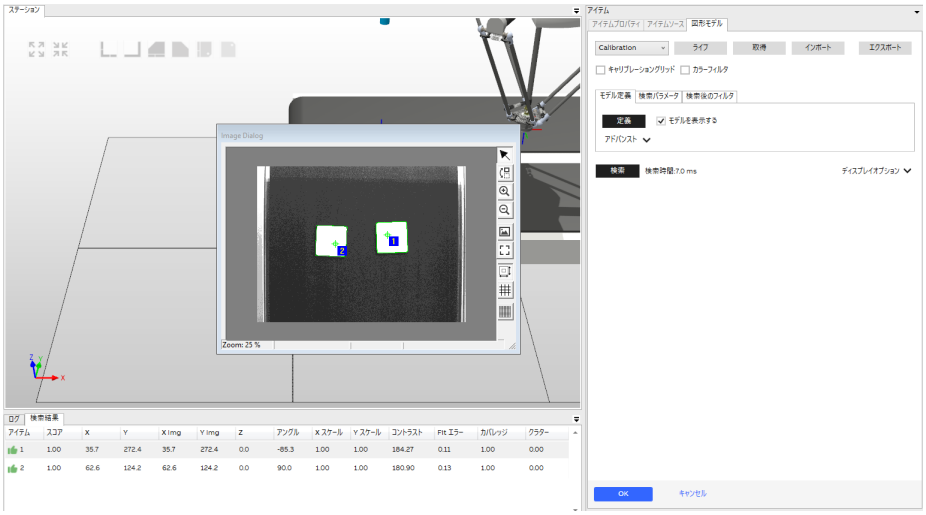
アイテム領域 は、画像ウィンドウ内の領域を表示します。赤い領域はオーバーラップを示し、それに対応するヒットは破棄と見なされます。

アイテムの角度 は、ロボットに送られる、アイテムの角度を表示します。この角度は、ティーチモデルに関連しています。

同じアイテムに複数のヒットがある場合、**ソート値**が使用されます。最も高いソート値を持つヒットのみがロボット コントローラに送信されます。ソート値はすべてのモデルに対して個別に設定できます。または、**ソート値**としてスコアを選択して*Geometric* スコアを使用することもできます。

10 **検索** をクリックして画像を分析します。必要に応じて、ソート値を定義します。

結果は、**画像** ダイアログに番号付きのヒットを含む画像として表示され、対応する結果リストが表示されます。

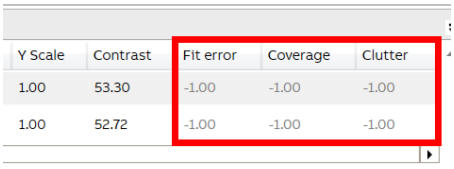


xx2100001636

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.9.2 Geometricモデルの設定
続き

> **注記**
>
> アルゴリズムとしてPatQuickを選択した場合、結果リストに使用できないパラメーター (**フィット エラー**、**カバレッジ**、**クラッター**) が以下のように表示されます。
>
> 
>
>
> xx2400001696

モデルのヒットは、通常そのまま分類されます。検査が使用された場合、ヒットは許可あるいは拒否として分類されます。アイテムプロパティタブ ページ142を参照してください。要件をすべて満たしていないヒットや、領域にオーバーラップしているヒットにはロボットからアクセスできず、［破棄］として分類さ れます。結果リストに表示されているヒットは、その分類内容を示すアイコンで表示されます。許可されなかったヒットについては、失敗したパラメータが結果リストに赤または青で表示されます。

検索時間 は、画像の分析にかかる時間をmsで表示します。

11 ［**OK**］をクリックします。

I PatMax アルゴリズムを備えたモデルでのみ使用できます。

> **注記**
>
> *Geometric*設定ウィンドウでの検索操作後に見つかったアイテムは、実際には別のパラメータ (フィット エラー、クラッターなど) によって拒否されている場合でも、アイテム領域の重複により破棄されたものとして表示されます。これは、アイテム領域がアクティブ化され、アイテム領域が実行時に互いに重複している場合にのみ発生します。ただし、破棄されたアイテムは、アイテム領域を適用する前に削除されます。

アイテム ターゲット内の*Geometric* パラメータ

ターゲット ストレージには、*Geometric* パラメータ `Score`、`fit error`[I]、`coverage`[I]、および`clutter`[I]を選択できます。

関連情報

アイテムプロパティタブ ページ 142。
カラービジョンの使用 ページ 324。
RAPID プログラム ページ 417。

© Copyright 2025 ABB. All rights reserved.

4.4.9.3 Blobモデルの構成

Blobモデルについて

機械ビジョンに使用できる最もシンプルなイメージは、二次元形状または輪郭です。Blob（輪郭）分析は、イメージ内の二次元形状を検知します。さらに、事前定義のグレースケール範囲に収まるピクセルの集まりを認識することにより、オブジェクトを検出します。

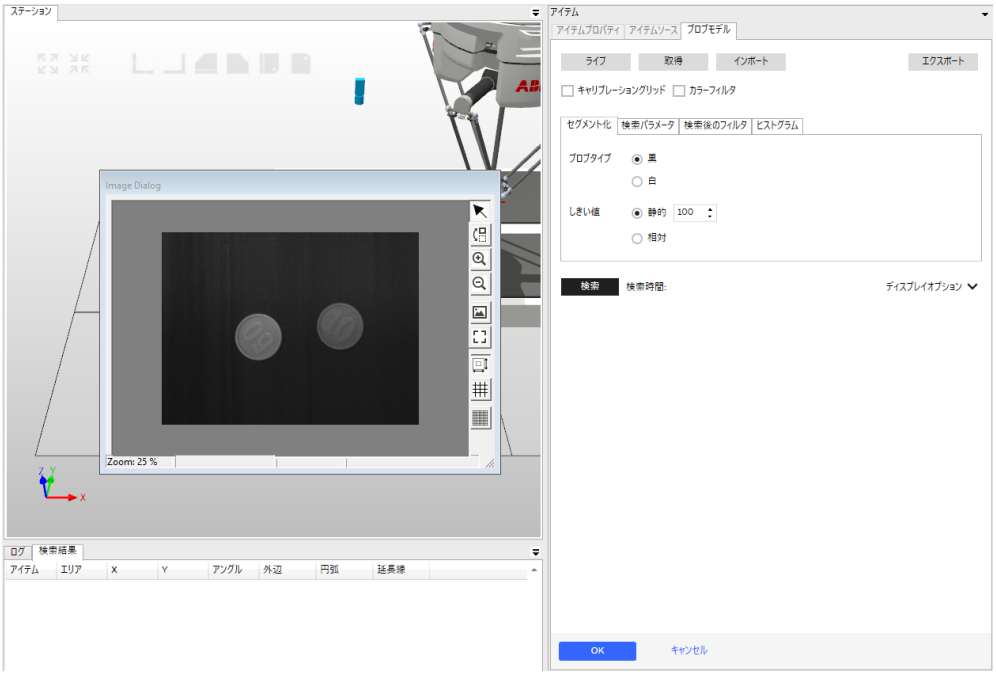
このような分析は、次のような場合のアプリケーションに適しています：

- オブジェクトのサイズ、形状、または方向に大きなばらつきがある場合。
- オブジェクトのグレー色調が、背景にない特有の色である場合。

Blob（輪郭）分析は、前景と背景のピクセルに容易に区分できるイメージにおいて、最適に機能します。通常、強い照明を使用した現場で不透明オブジェクトが使用されると、このように分析に適したイメージが作り出されます。

パーツのグレースケールトーンが同程度の画像のコントラストを高くするには、カラーフィルタを使用します。カラービジョンの使用 ページ*324*を参照してください。

[ブロブ (輪郭) の構成]の図

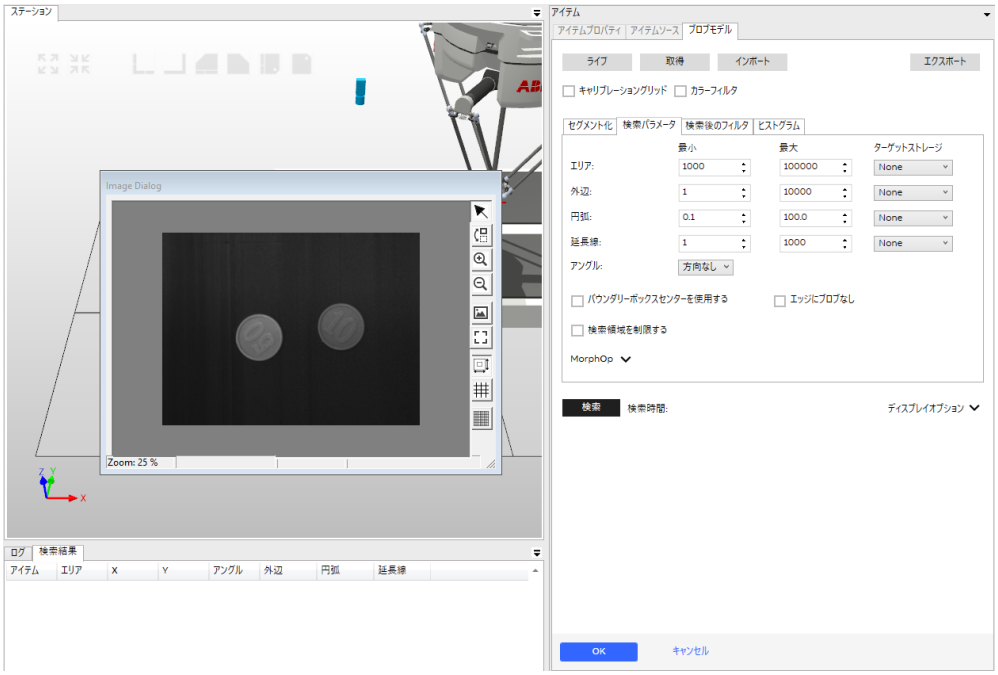


xx2100001661

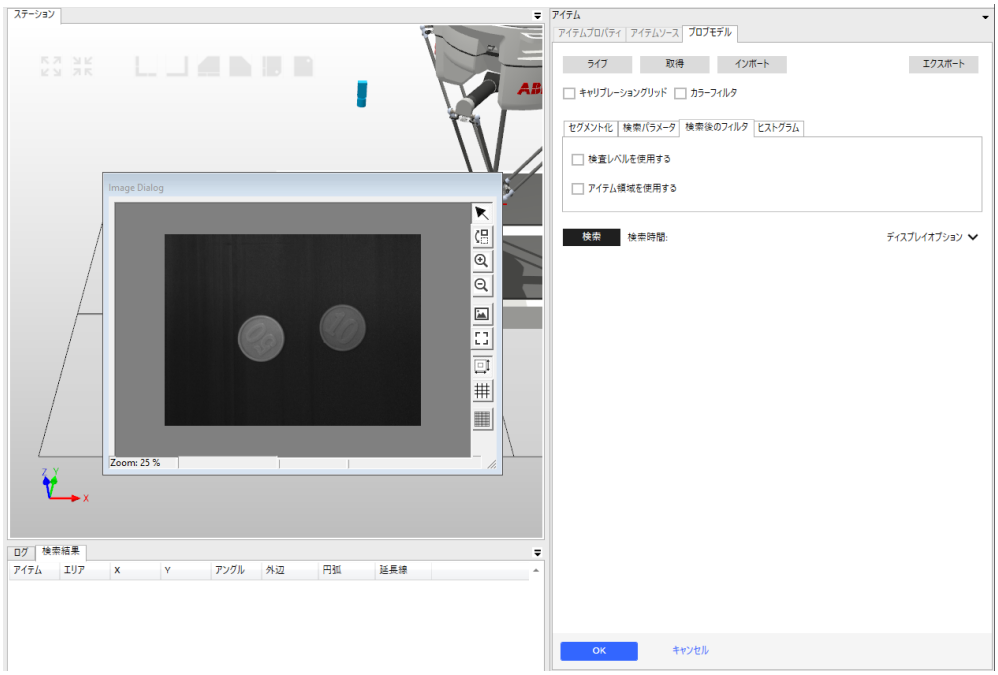
次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.9.3 Blobモデルの構成
続き



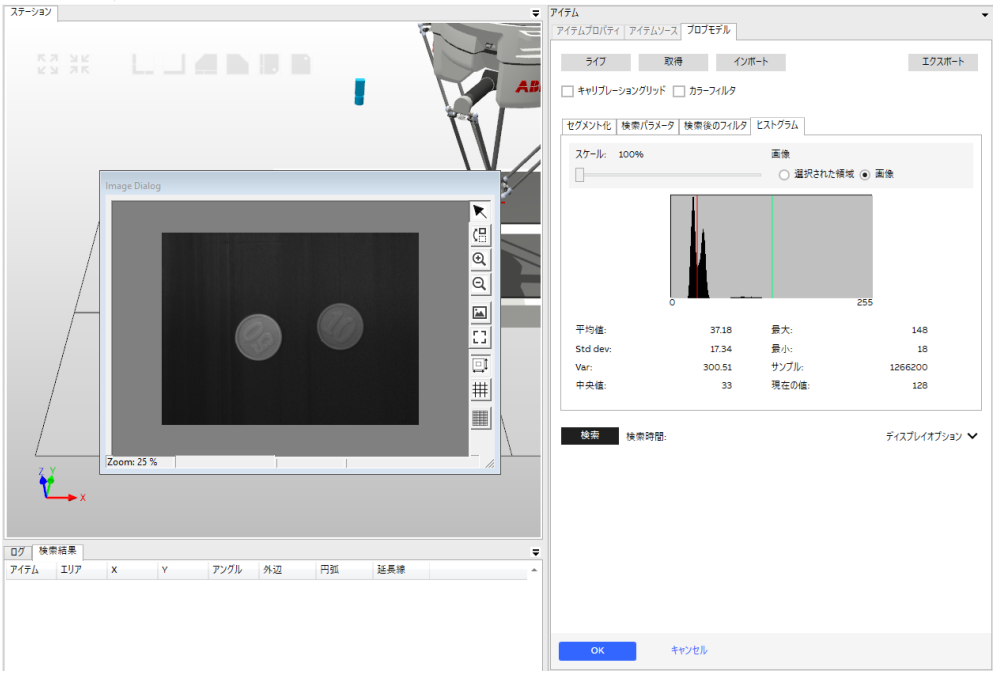
xx2100001662



xx2100001663

次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2100001664

輪郭ビジョンモデルの構成

下記の手順に従って、輪郭ビジョンモデルを構成してください。

1. ツリー ビューの**プロセス**で 1 つの**アイテム**を右クリックし、**設定** を選択します。

   アイテム設定ウィンドウが開きます。
2. **アイテム ソース** タブをクリックして選択します。
3. **アイテム ソース** ダイアログで、**新しいモデル** をクリックし、**ブロブ** を選択します。
4. [Image（イメージ）] パートで [Live（ライブ）]、**[Acquire（取得）]**, または [Import（インポート）] をクリックして イメージを取得します。

   [Calibration origin（キャリブレーション起点）] チェックボックスを 選択して、座標システムのヘルプラインを表示します。 [Histogram（ヒストグラム）] をクリックして、 取得したイメージ内のピクセル分布をグラフで表示します。

   カラー フィルタリングを使用する必要がある場合は、**カラー フィルタ** チェックボックスを選択してフィルターを有効にし、**カラー フィルタ** タブでフィルタ パラメータを設定します。カラービジョンの使用 ページ 324 を参照してください。
5. **モデル定義** の下の**セグメンテーション**で 白 をクリックして選択します。

   [Segmentation（セグメント化）] パートで セグメント化方法と輪郭タイプを選択します。

   セグメント化は、イメージ内のピクセルをオブジェクトピク ルと背景ピクセルに分ける作業です。通常、オブジェクト には1の値、背景ピクセルには0の値が割り当てられています。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

[Static（静的）] 方法では、輪郭ピクセルと背景ピクセルの区分けは、グレー値により決定されます。しきい値に満たないグレースケール値を持つピクセルは、すべてオブジェクトピクセルとして割り当てられ、しきい値を超える値を持つピクセルはすべて背景ピクセルとして割り当てられます。

[Relative（相対）] 方法では、相対的しきい値を左端と右端の間の合計ピクセルのパーセンテージとして表すことによって、輪郭ピクセルと背景ピクセルを区分けします。両端は、ヒストグラムの極点（最低値と最高値）に存在するノイズレベルピクセルを表します。

静的セグメント化は、パーセンテージに対応するグレーレベルを計算する必要がないため、相対的セグメント化よりも早く実行できます。静的セグメント化は、現場のフィーチャーの有無をテストできます。一方、相対的セグメント化は、常に現場の輪郭を検出します。

6 要件に応じて、検索パラメータ のパラメータを調整します。

[Search Parameters（検索パラメータ）] パートで、 フィーチャーの値を定義します。

[Area（エリア）] はmm^2で表示されます。

[Perimeter（外辺）] は mm で表示されます。

[Circularity（円弧）] は 円弧を定義します。1 の値は、円形が完璧で、完全に塗りつぶされていること（穴がない）を意味します。

[Elongation（延長線）] は 第二主軸の周りのフィーチャーの第二慣性モーメントと、最初の主軸の周りのフィーチャーの第二慣性モーメントの比率 です。

[Angle（アングル）] は検出アイテムがコントローラにどのように送られるかを定義します。

- [No Orientation（方向なし）] は、 検出アイテムがアングル=0 （ゼロ）の値でコントローラに送らるということを意味します。
- [First Principal Axis（最初の主軸）] は、 検出アイテムが最初の主軸の周りのアングルと共に送られということを意味します。アングルは、x 軸に相対し、±90 度の範囲です。

[Use boundary box center（境界ボックスの中心の使用）] は、 輪郭のポジションが、その質量の中心ではなく、その境界ボックスの中心にあるかどうかを定義します。

[No Blob On Edge（エッジに輪郭なし）] は、検索領域のエッジに触れる輪郭を報告するかどうかを定義します。

検査レベルの使用は、見つかったモデルを分類する必要があるかどうかを定義します。アイテムプロパティタブ ページ142を参照してください。 アイテム タイプは RAPID コードで読み取ることができます。RAPID プログラム ページ417を参照してください。検査レベルの使用 を選択して、[検査パラメータ] 部分を開きます。

[Use Inspection levels（検査レベルの使用）] が選択されないと、 すべての検出モデルが許可として分類されます。 [Search Parameters（検索パラメータ）] に指定された条件を満たす、 すべてのモデルが分類されます。

[Limit Search Region（検索領域の制限）] は輪郭分析の検索エリアを制限します。このエリア内のオブジェク トのみが検索対象となります。

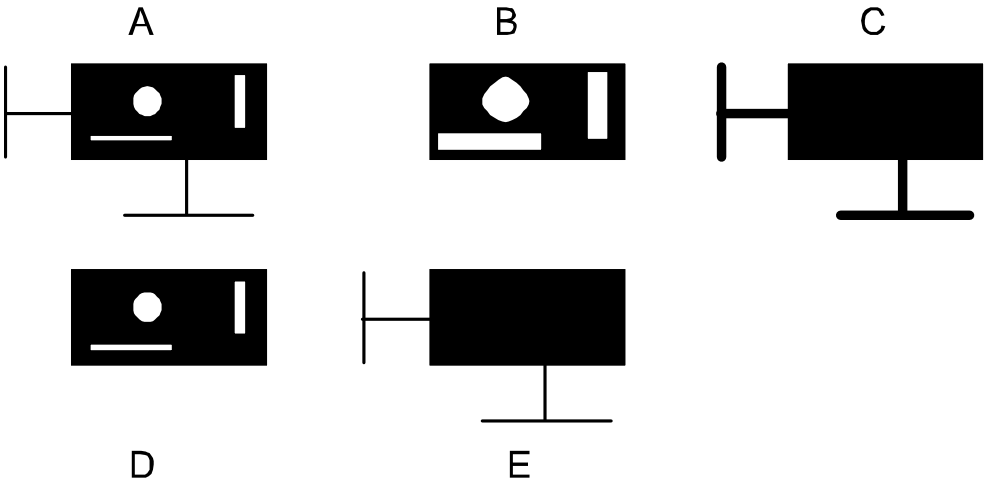
次のページに続く

© Copyright 2025 ABB. All rights reserved.

注記

検索 を押して輪郭ツールを調整すると、輪郭アルゴリズムによってすべての輪郭が一覧表示されます。サイズのしきい値制限を調整し、大きすぎる輪郭や小さすぎる輪郭をフィルタで除外します。必要に応じて他のパラメータを調整します。

7 必要に応じて、MorphOp欄で［Morphological（形態構造）］および/または［Clean Up（クリーンアップ）］チェックボックスを開いて設定を定義します。



xx0900000542

A	オリジナル
B	縮小
C	拡張
D	開く
E	閉じる

形態構造の設定：

- ［Erode（縮小）］はオブジェクトフィーチャーを削減または削除したり、オブジェクト内にある穴の厚みを増加します。この操作は、イメージ内の各ピクセルの値を、垂直方向と水平方向に隣接する8つのピクセルの最高値で置き換えます。
- ［Dilation（拡張）］はオブジェクト内の穴を削減または削除したり、オブジェクトのフィーチャーの厚みを増加します。この操作は、イメージ内の各ピクセルの値を、垂直方向と水平方向に隣接する8つのピクセルの最小値で置き換えます。
- ［Closing（閉じる）］は穴を削除します。小さなフィーチャーを保存します。イメージに対して、縮小操作、拡張操作が順に適用されます。
- ［Opening（開く）］は穴を保存します。小さなオブジェクトフィーチャーを削除します。イメージに対して、拡張操作、縮小操作が順に適用されます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

クリーンアップの設定：

- ［Prune（簡素化）］は、指定サイズ（連続性サイズ）を下回るフィーチャーを削除せずに無視します。イメージで指定サイズを下回るフィーチャーがすべて無視されると、簡素化されたフィーチャーを含む輪郭に対して返ってくる測定値が、簡素化されたフィーチャーがまだ存在するかのように計算されます。ただ し、簡素化されたフィーチャーは、実際には計算には入れられません。
- ［Fill（塗りつぶし）］は、簡素化されたフィーチャーを、左側に隣接するピクセルのグレー値を使用して塗りつぶします。フィーチャーの塗りつぶしに使用されるピクセル値は、塗りつぶされるフィーチャーのすぐ左にポジションするピクセルの値です。フィーチャーに含まれるピクセルの各列が塗りつぶされる際、その列のすぐ左側にあるピクセル値が、その列の塗りつぶし値として使用されます。
- ［Connectivity（結合性）］は、輪郭が使用されるために必要な最小サイズ（ピクセル単位）を定義します。簡素化あるいは塗りつぶしで使用されます。

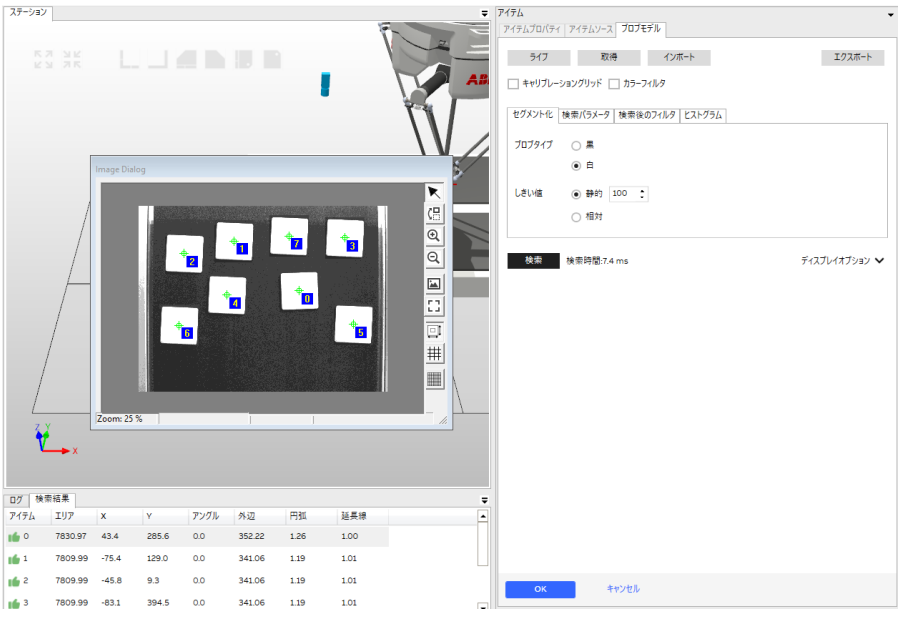
8 **アイテム領域部分で、アイテム領域を使用** チェックボックスを選択し、**領域を定義** をクリックします。 頂点を使用して、見つかった物体の周囲に表示されるポリゴンを調整します。 次に、**教える** をクリックします。

多角形には2個から16個の頂点があります。

9 **表示オプション** で**検索** をクリックします。

ヒント

検索結果が**イメージ ダイアログ** と一致する場合、設定は成功です。



xx2100001637

次のページに続く

© Copyright 2025 ABB. All rights reserved.

表示オプション の部分でセグメント化イメージ を選択して処理済みイメージを表示します。結果がどのように表示されるかを選択します。

- アイテム範囲 は画像ウィンドウに、輪郭の範囲を表示します。
- 境界ボックス は輪郭全体を含む最小の横長方形を表示します。
- ［Item region（アイテム領域）]はイメージウィンドウ内の領域を表示します。赤い領域はオーバーラップを示し、それに対応するヒットは破棄と見なされます
- ブロブ角度 はロボットに送られる、アイテムの角度を表示します。
- スコア値は、画像ウィンドウで選択したアイテムのスコアを表示します。

10 ［OK］をクリックします。

アイテムターゲットのBlobパラメータ

ターゲットストレージにはblob パラメータ、`Area`, `perimeter`, `circularity`, `elongation` を選択できます。

関連情報

アイテムプロパティタブ ページ *142*。
カラービジョンの使用 ページ *324*。
RAPID プログラム ページ *417*。

© Copyright 2025 ABB. All rights reserved.

4.4.9.4 検査モデルの構成

検査モデルについて

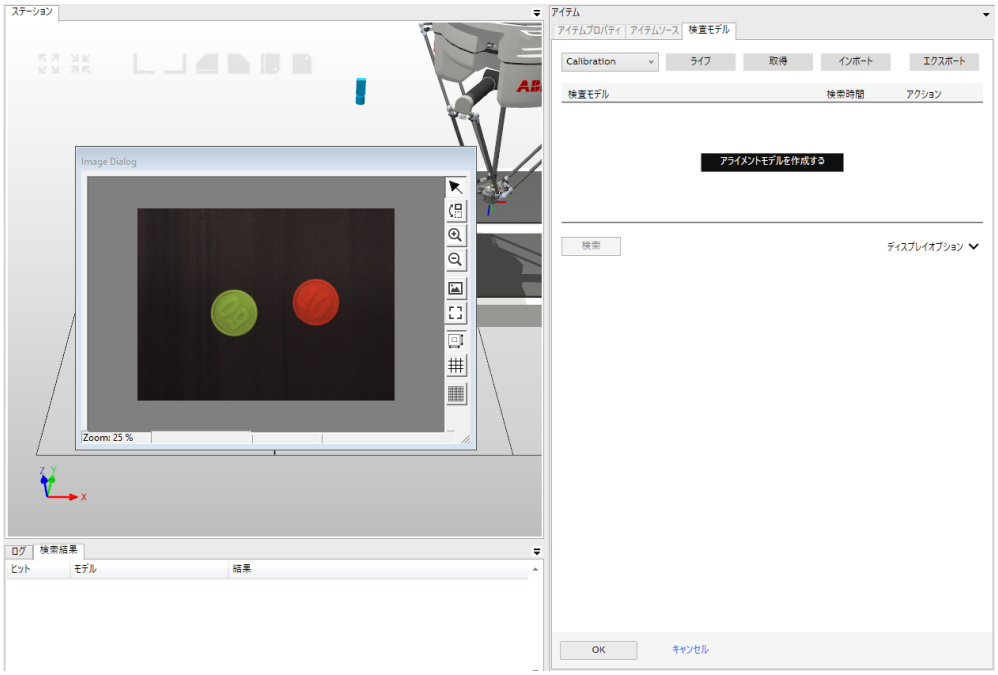
検査モデルでは、*Geometric*、Blob、ヒストグラム、および*Caliper*の複数のモデルを組み合わせることができます。これは、検査 *II* と呼ばれることもあります。

検査モデルは常にアライメント モデルで構成されます。アライメント モデルは、*Geometric*またはBlobのいずれかで、検査モデルの参照として機能します。アイテムのピック/プレース位置と回転は、このモデルの位置と回転です。

検査エリアはアライメント モデルを基準として定義され、Blob、ヒストグラム、*Caliper*、または*Geometric*のいずれかをこれらの各エリア内で実行できます。見つかったアイテムの数やアライメント モデルを基準とした位置などの条件を設定できます。

許可と分類されるアイテムの場合、すべての検査エリアと整列モデルも許可として分類されている必要があります。いずれかの検査エリアがこのような条件を満たさない場合、対応するアイテムが拒否されたものとして分類されます。

検査の構成の図



xx2100001668

検査モデルの構成

下記の手順に従って検査モデルを構成してください。

1. ツリー ビューのプロセスで 1 つのアイテムを右クリックし、設定 を選択します。

   アイテム設定ウィンドウが開きます。
2. アイテム ソース タブをクリックして選択します。
3. アイテム ソース ダイアログで、新しいモデル をクリックし、検査 を選択します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4 ［Image（イメージ）］欄で ［Live（ライブ）］、 ［Acquire（取得）］または［Import（インポート）］をクリックしてイメージを取得します。

5 ［Inspection model（検査モデル）］欄で、 整列モデルとその対応検索エリア間の関係を定義します。

作成されたモデルはツリー ビューで表示されます。

整列モデル は、アイテムのポジションと向きを定義します。整列モデルの構成ダイアログの詳細については、ビジョンモデル化 ページ293を参照してください。

サブ検査モデル は、整列モデルに検査エリアを追加します。サブ検査モデル ページ 315を参照してください。

［編集］は選択したモデルの構成ダイアログを開きます。既存の整列モデルに変更が加えられる場合、検査エリアへの関係は再教示されなくてはなりません。

［削除］は、選択されたモデルとそれに対応する検査エリアを消去するために使用します。

エリアの編集 は、すべての検査エリアを表示します。選択したサブ検査モデルのエリアを再配置できます。 編集エリアには、現在のモデルのエリアが表示されます。 選択したサブ検査モデルの領域を再配置できます。

6 **整列モデルの作成** をクリックして、**モデル タイプの選択** ドロップダウン リストを開きます。

7 整列モデルを作成するには、ドロップダウン リストで **図形** または **ブロブ** を選択します。 図形 モデルまたはブロブ モデルを作成する方法の詳細な手順については、*Geometricモデルの設定 ページ 296* または*Blobモデルの構成 ページ 305* を参照してください。

8 **+ サブ検査モデル** をクリックして、**モデル タイプの選択** ドロップダウン リストを開きます。

9 サブモデルを作成するには、ドロップダウン リストで **図形**、**ブロブ**、**ヒストグラム** または **カリパス** を選択します。

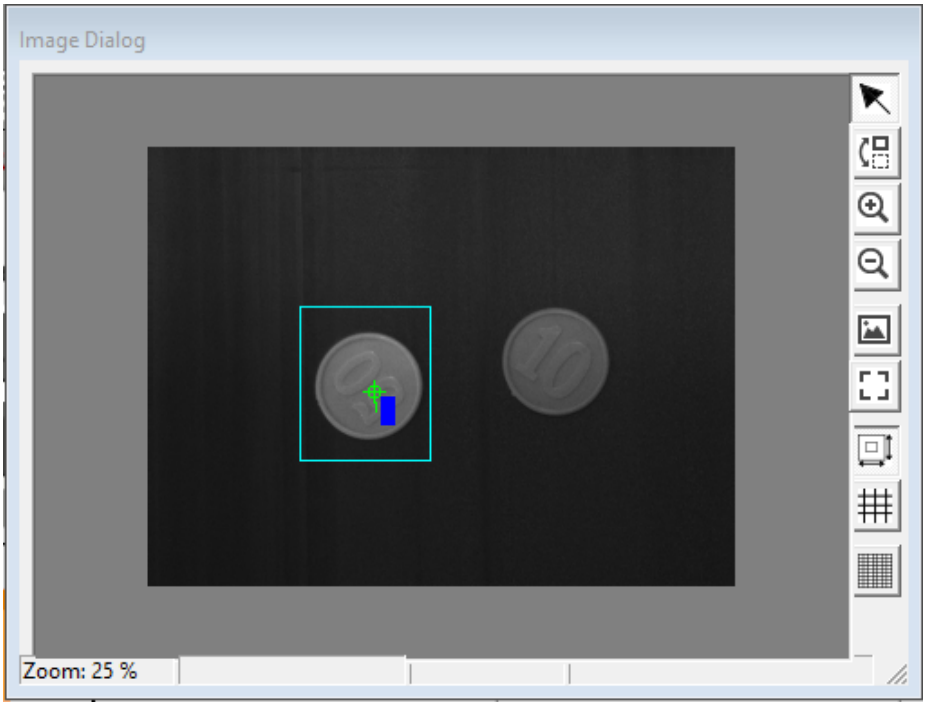
10 表示されたダイアログで **OK** をクリックしてエリアを編集します。

11 パターンを覆うように長方形をドラッグします。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.9.4 検査モデルの構成
続き



xx2100002275

12 **Edit** ボタンをクリックして、対応するモデル作成ウィンドウを開きます。 図形モデル、ブロブ モデル、ヒストグラム モデル、またはカリパス モデルを作成する方法の詳細な手順については、*Geometricモデルの設定 ページ296* *Blobモデルの構成 ページ305*、*ヒストグラム ページ316* および *Caliper ページ319* を参照してください。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

注記

図形サブモデルの場合、モデルを定義してトレーニングをした後、別のトレーニングを行う必要があります。



13 検索 をクリックします。

結果は、画像ダイアログに番号付きヒットの画像として表示され、対応する詳細リストが検索結果ウィンドウに表示されます。

ヒント

検索結果がイメージ ダイアログ と一致する場合、設定は成功です。

14 [OK] をクリックします。

サブ検査モデル

概要

サブ検査モデルは整列モデルに検査エリアを追加するために使用されます。各エリアは指定されたサブ検査モデルを使用します。検査エリアは、サブモデルを分析する場所を整列モデルに相対するポジションとして定義します。エリアは画像内に表示され、分析するエリアをカバーするために、移動され、サイズを変更されます。

サブ検査モデルは、それら自身のダイアログ内で構成されます。サブ検査モデルをテストする際には、整列ヒットは対応する検査エリアと共にイメージウィンドウ内に表示されます。サブ検査モデルは、それ自身の検査エリアによって定義されるイメージの部分のみを分析します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.9.4 検査モデルの構成
続き

図形

ジオメトリック サブ検査モデルは、*Geometric* モデルと同じ方法で設定されます。*Geometricモデルの設定 ページ296*を参照してください。さらに、見つかったアイテムの相対位置と対応するアライメント ヒットをトレーニングする必要があります。

必要とされるヒット数は許可されたと見なされる結果に必要な、検査エリア内のヒット数とそのサブ検査モデルを定義します。

［**Deviation limits（標準偏差の限界）**］は教示ポジションからの許容偏差を定義します。

検索を実行して検査エリア内で対象アイテムが検出されたら、それらのポジションを教示します。相対ポジションは**xDiff**、**yDiff**および**AngleDiff**として表示されます。

［**Train（教示）**］をクリックして、整列ヒットに対する検出アイテムのポジションを保存します。

アイテムターゲットの幾何補助検査パラメータ

ターゲットストレージにはパラメータ`Number of hits` を選択できます。

輪郭

輪郭サブ検査モデルは、Blobモデルと同じように構成されます。*Blobモデルの構成 ページ 305*を参照します。必要とされるヒット数も構成する必要があります.

必要とされるヒット数は許可されたと見なされる結果に必要な、検査エリア内のヒット数とそのサブ検査モデルを定義します。

アイテムターゲットのBlob補助検査パラメータ

ターゲットストレージにはパラメータ`Number of hits` を選択できます。

ヒストグラム

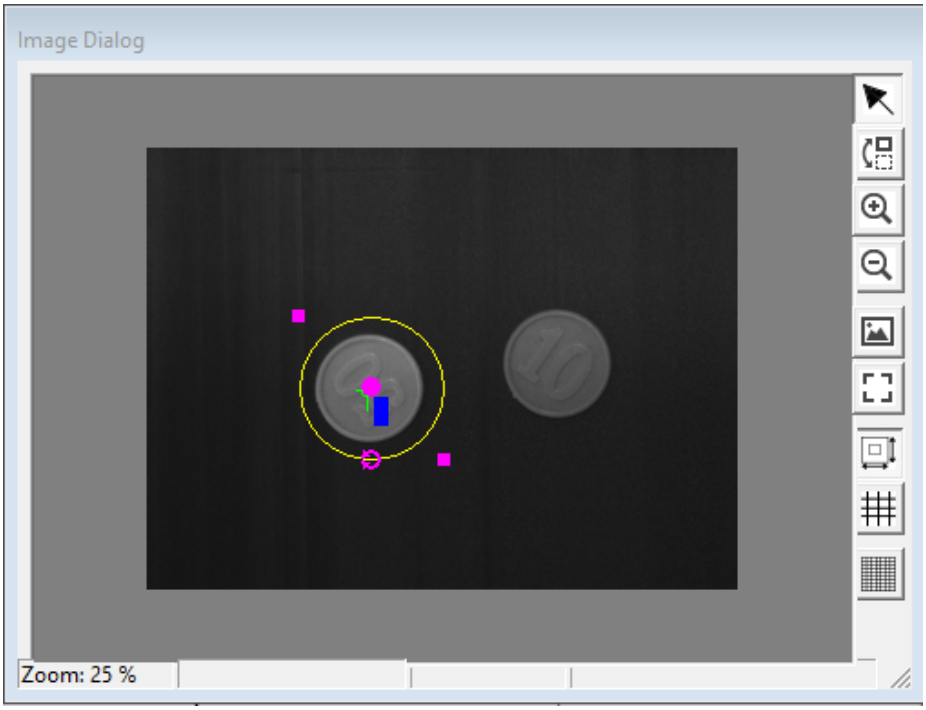
ヒストグラムツールは、特定領域の色またはグレースケールレベルを計測します。モノクロームカメラを使用した場合、ヒストグラムツールでは領域内のグレースケールレベルを計測します。同様に、カラーカメラを使用した場合は、赤、緑、青のカラーチャンネルそれぞれが個別に計測されます。ヒストグラムツールは、認識・分類対象のオブジェクトが類似の形状であり、かつ異なる色の場合に効果的です。

ヒストグラムサブ検査モデルの検査エリアは、常に円形としてグラフィック表示されますが、ヒスグラム分析で実際に使用されているエリアは、サブ検査エリア内の四角形であり、この四角形部分がイメージに整列されています。

1 **+ サブ検査モデル** をクリックして、**モデル タイプの選択** ドロップダウン リストを開きます。
2 サブモデルを作成するには、ドロップダウン リストで **ヒストグラム** を選択します。
3 表示されたダイアログで **OK** をクリックしてエリアを編集します。
4 パターンを覆うように円をドラッグします。

次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2200001129

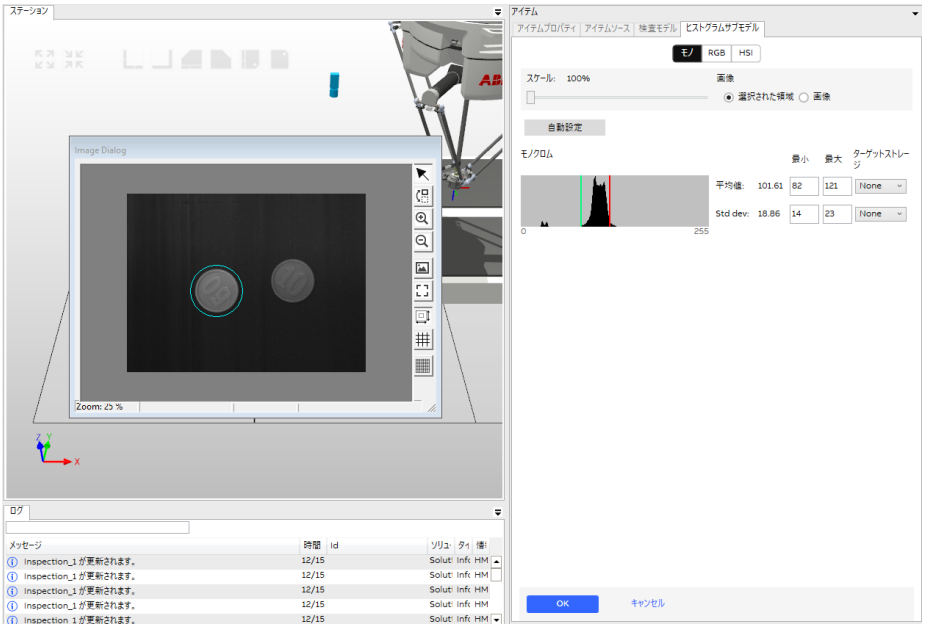
5 アクションの下の 編集 アイコンをクリックして、ヒストグラムモデルの編集ウィンドウを開く。

6 ［自動設定］を押すと、適切なヒストグラム用の範囲 (最小値と最大値) が自動設定されます。また、代わりに最小値と最大値は、赤と緑のバーをヒストグラム上でスライドするか、テキストボックスに直接値を入力して手動設定できます。適切な結果を得るには、標準偏差と平均値が指定した制限値以内である必要があります。カラービジョンのヒストグラムを使用する場合、全チャンネルのヒストグラムが制限値以内である必要があります。

次のページに続く

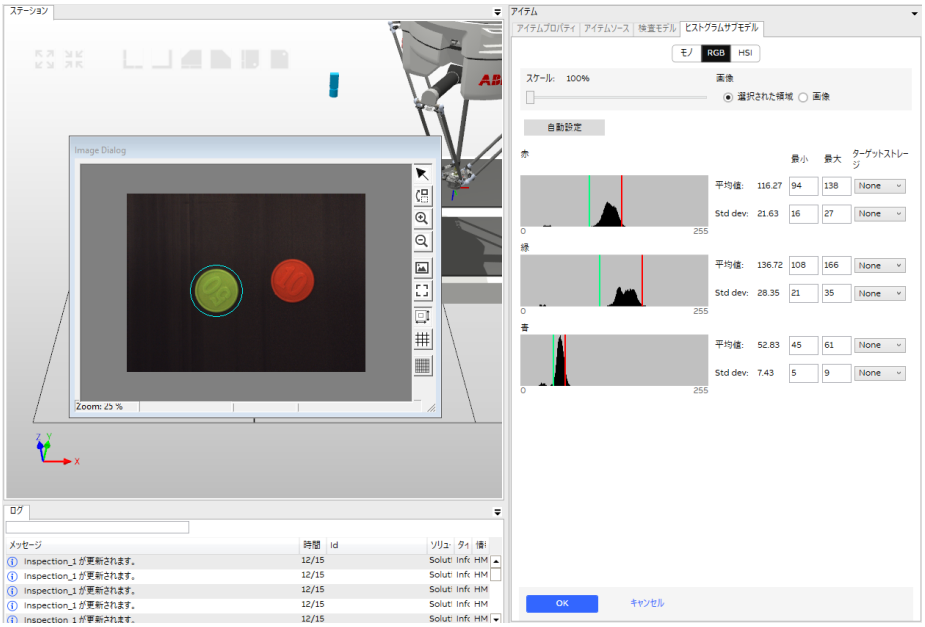
© Copyright 2025 ABB. All rights reserved.

4.4.9.4 検査モデルの構成
続き



xx2200001126

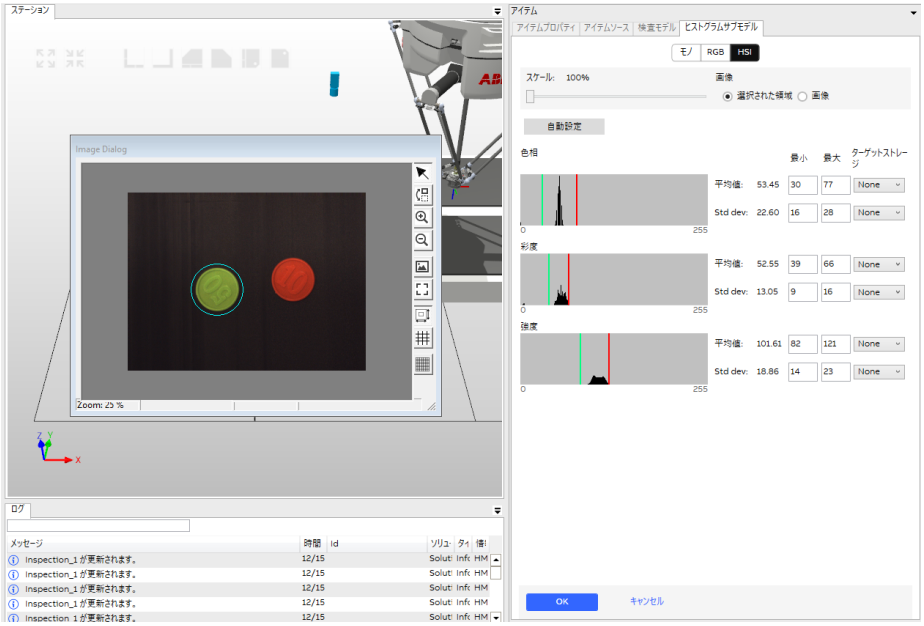
7 タブ **RGB** または **HSI** に変更すると、色のウィンドウが表示されます。



xx2200001127

次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2200001128

8 ［OK］をクリックします。

分析エリアを採用と却下に分類するため、ヒストグラムツールでは、指定エリアについて次の2項目の程度が評価されます:

［Mean（平均）］は検査モデルの最小値と最大値を定義します。検査エリアの平均が最小値に満たない場合、あるいは最大値を超える場合、検査エリアは拒否されたものとして分類されます。

[Std dev]はピクセルの値が平均値周辺にどのように集積しているかを計測する値です。均等な色調の場合、標準偏差の低い、狭い領域に分布するヒストグラムが形成され、スペックルパターンでは標準偏差 Std dev の高い、広範囲におよぶヒストグラムが形成されます。

アイテムターゲットのヒストグラム補助検査パラメータ

ターゲットストレージにはパラメータ、`Meanstandard deviation` を選択できます。

Caliper

*Caliper*ツールは、外縁部分を識別し、その間の距離を計測します。この分析は対応する分析エリア内のみに対して実施されます。各パーツのグレースケールが類似している場合、イメージのコントラストを上げるためにカラーフィルターを適用可能です。詳細はカラービジョンの使用 ページ *324*を参照してください。

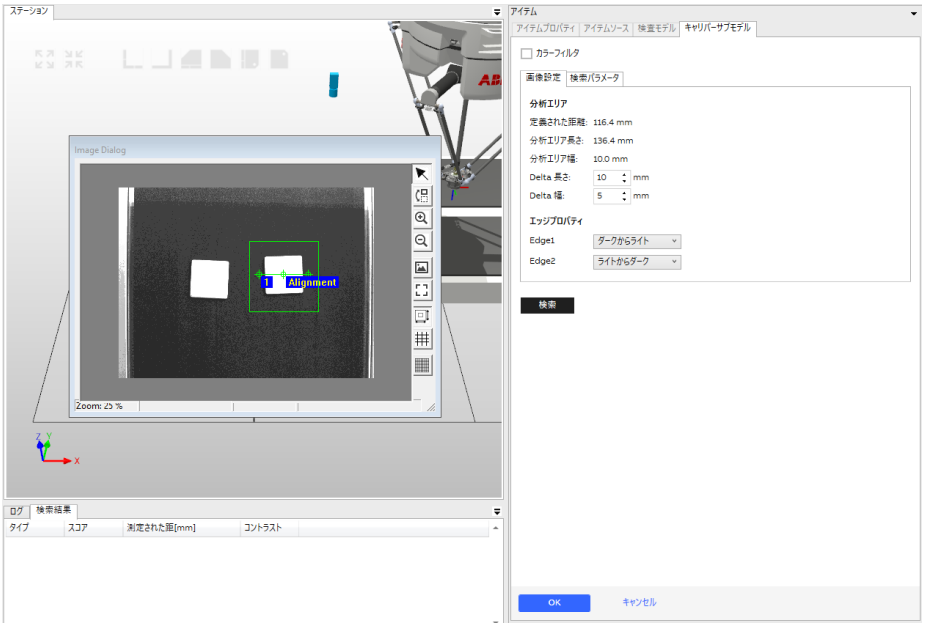
1 + サブ検査モデル をクリックして、モデル タイプの選択 ドロップダウン リストを開きます。
2 サブモデルを作成するには、ドロップダウン リストで カリパス を選択します。
3 表示されたダイアログで OK をクリックしてエリアを編集します。
4 パターンを覆うように長方形をドラッグします。
5 アクション の下の 編集 アイコンをクリックして、キャリパーモデル編集ウィンドウを開く。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

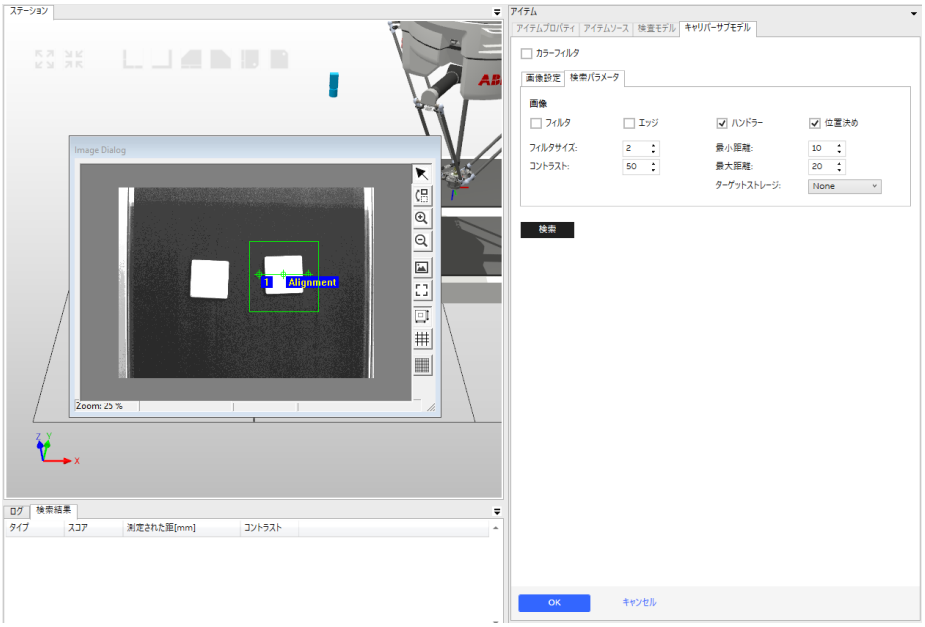
4.4.9.4 検査モデルの構成
続き

6 端点が画像設定の下のエリアの端に配置されるように線を移動します。



xx2200001119

7 分析エリアの定義された距離に従って、検索パラメータのパラメータを調整します。

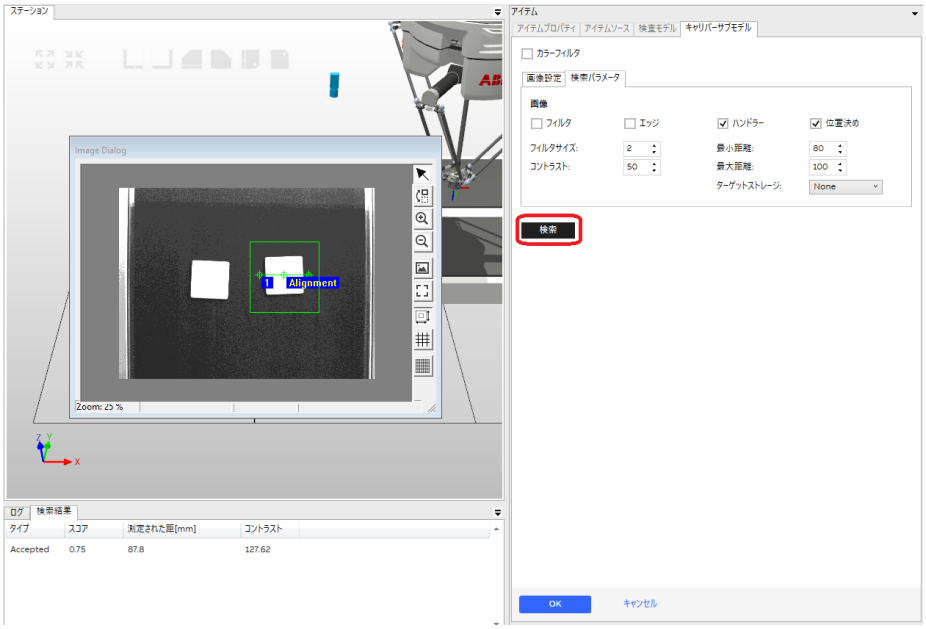


xx2200001120

8 検索 をクリックします。

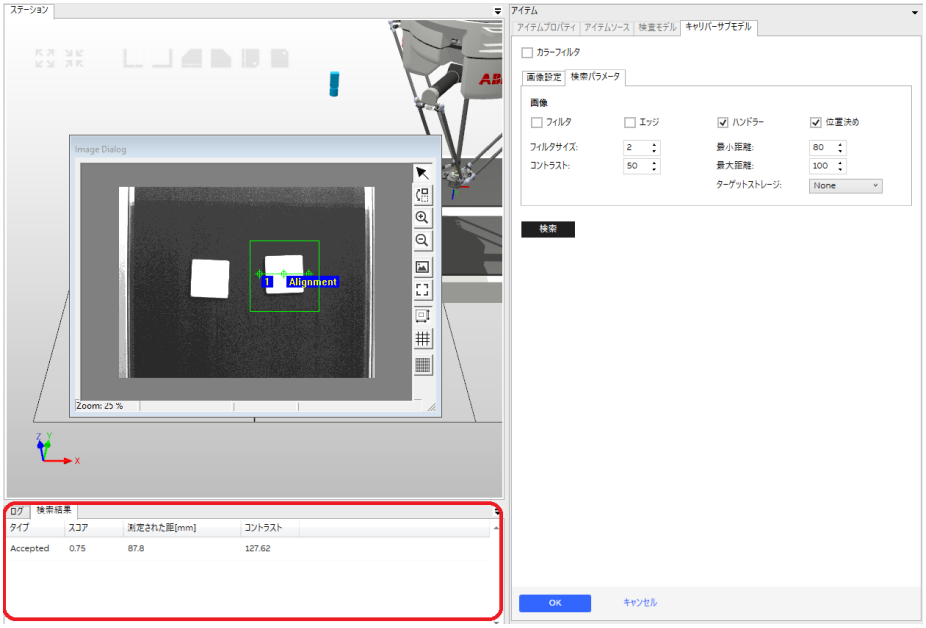
次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2200001121

検索結果タブに結果が表示されます。



xx2200001122

9 [OK] をクリックします。

*Caliper*分析を実行するため、検索ラインを囲む長方形が定義されます。

定義された距離は、画像ダイアログにある緑色の線の終点間の距離です。 端点がエリアの端に配置されるように線を移動します。

分析エリアの長さは、カリパス分析が実行される長方形の長さです。 分析エリアの長さを増やすには、デルタの長さの値を増やすか、定義された距離ラインのサイズを変更します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.9.4 検査モデルの構成
続き

分析エリアの幅は、カリパス分析が実行される長方形の幅です。**分析エリアの幅**を増やすには、**デルタ幅**の値を増やします。

デルタの長さは、**分析エリアの長さ**を取得するために、**定義された距離**に追加する追加の mm を定義します。

```
Analyze area length=2*Delta length + Defined distance
```

デルタ幅は分析エリアの幅を定義します。

```
Analyze area width=2*Delta width
```

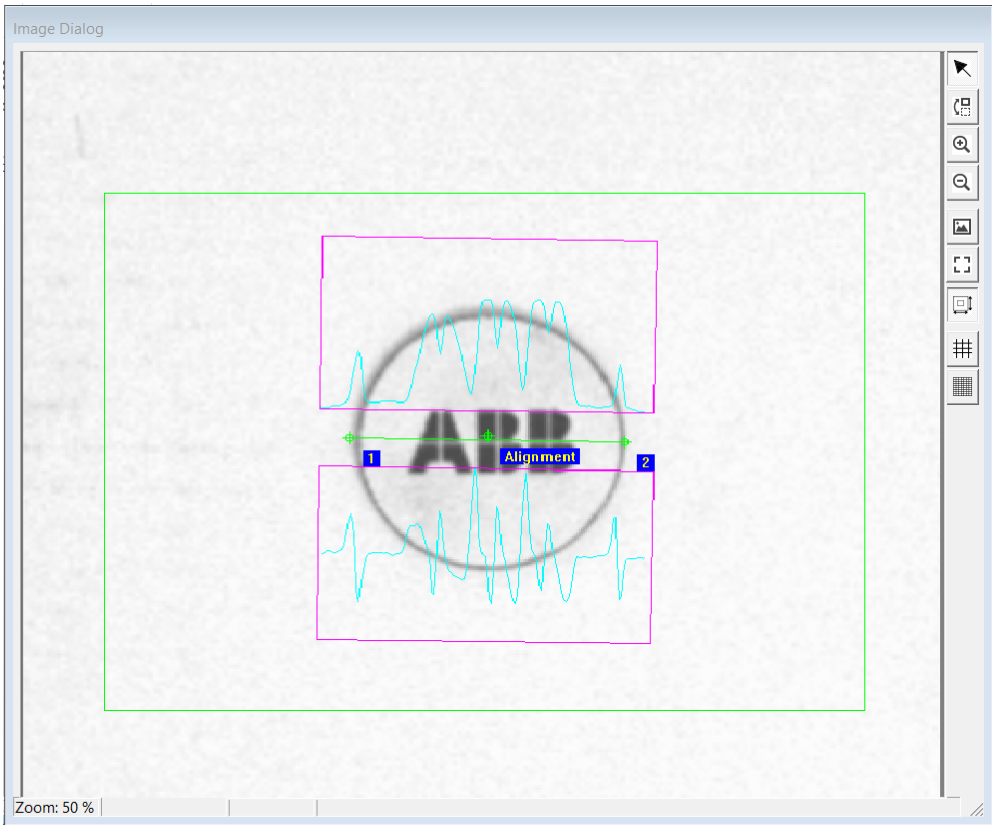
分析エリアから生産画像が作成されます。この操作では、分析エリア内のすべての情報が合計され、**分析エリアの幅**に平行なエッジの強度が強調され、ノイズの影響が軽減されます。

エッジのプロパティはエッジの極性を定義します。極性は、**エッジ 1**から**エッジ 2** までの測定値として定義されます。

検索パラメータは、ガウス曲線を使用して フィルタを定義します。フィルタは、*Caliper*ツールがどのようにノイズ低減させるか、画像のピーク、コントラスト、および間隔を強調するかをコントロールします。

検索は、指定された距離 (**定義された距離**) と定義された極性を持つ 2 つのエッジを検索するために使用されます。

検索パラメータのチェックボックスは、**画像ダイアログ**に表示する結果を定義します。



xx2200001123

タームターゲットのカリパス補助検査パラメータ

ターゲットストレージには`Distance` パラメータを選択できます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

外部モデル

この機能は、次のバージョンのために予約されています。

© Copyright 2025 ABB. All rights reserved.

4.4.9.5 カラービジョンの使用

カラービジョンについて

PickMaster PowerPacにはモノクロ、カラーのどちらのカメラも使用できます。両者の違いは、モノクロ（グレースケール）イメージでは各ピクセルを表示するのに、1つの8ビット値しか使用しないのに、カラーカメラで取得したイメージは3つの8ビット値（10進法で0～255）を使用するということです。モノクロイメージでは白から黒までのグレーレベルを8ビット値で表示しますが、カラーイメージでは3つの別々のカラーチャネルの内容を3つの値で表します。これらの3つのチャネルは赤、緑、および青（カラースペースRGB）、または色相、彩度、および明度（カラースペースHSI）を表します。どちらのカラースペースで作業をするかはイメージの内容によります。

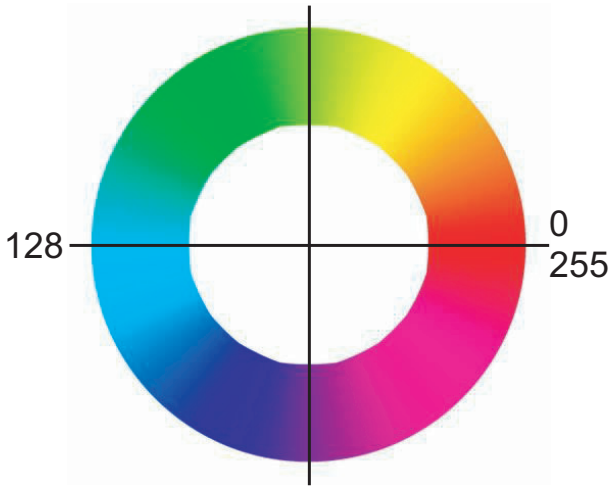
カラースペース

RGBで作業する場合には、各ピクセルの色は、その赤、緑、および青の含有量で表されます。数値的な表現は三原色の場合、赤は（255、0、0）、緑は（0、255、0）、青は（0、0、255）となるので、わかりやすいです。しかし、三色の混合した色を理解するのは困難です。

HSIは、人間の色の知覚にとって、より理解しやすいカラースペースです。

- 色相：電磁スペクトル上の色のポジション。下図を参照してください。
- 彩度：色の純度。
- 明度：色の明るさ。

色相スペクトルは一周すると元に戻るので（0と255の両方が赤を表します）、1つの輪として表示するのが適当です。



xx2100002336

カラーフィルタリングを使用すると、似ていない色の識別がより簡単になります。類似の程度は、カラースペース内の色の間の距離で表すことができます。違いの明瞭さは使用するカラースペースによって異なる場合があり、これが両方のカラースペースのフィルタを使用する方が良いという理由です。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

照明

カラーシステムはイメージの色の内容情報をより多く供給できるので、照明の状態に対しても、より敏感です。時間が経っても変化しない、均一な照明を備えることが重要です。

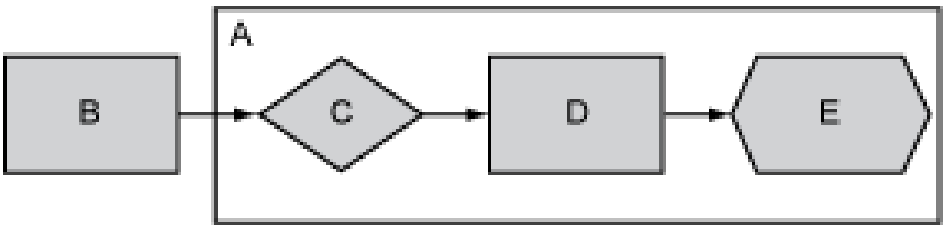
コンピュータの性能

カラービジョンは非常にリソースを消費する機能であり、画像の取得、修正、フィルタリングのすべてに時間が掛かります。カメラの台数とフレームレートをほどほどにしておくことが重要です。この機能はビジョンタスクとコンピュータリソースとの組み合わせで実現するので、性能の限度は大きく変動します。

PickMaster PowerPac のカラービジョン

PickMaster PowerPacは、フィルタの形式で色覚を提供します。このフィルタは、スタンドアロン、アライメント、およびサブ検査モデルの両方で、*Geometric*、Blob、およびCaliper設定ダイアログからアクセスできます。フィルタは、オブジェクトの認識または測定の前に行われる前処理ステップです。各モデルには、独自のフィルタ設定があります。

以下の図のように、カメラがカラーイメージを取得し、そのイメージはカラーフィルタを通すことによってグレースケールイメージに変換されます。



xx0900000445

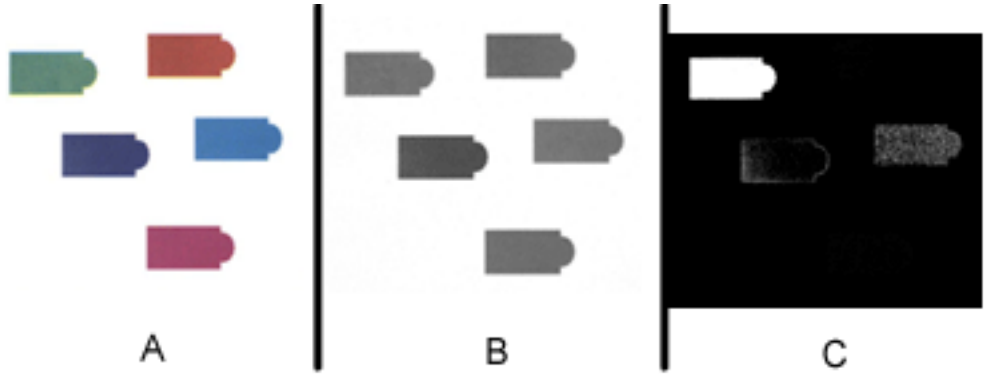
A	ビジョンモデル
B	カラーイメージ
C	カラーフィルタ
D	グレースケールイメージ
E	オブジェクトの認識

カラー フィルタの結果は、フィルタ設定に従って特定の色が強調または減衰されたグレースケール画像です。オブジェクト認識ツール (*Blob*/*Geometric*) は、このグレースケール画像に対して動作します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.9.5 カラービジョンの使用
続き



xx0900000446

A	カラーカメラが撮影したイメージ。
B	モノクロカメラが撮影した同一のシーン。
C	緑を抽出するように設定されたフィルタを通過した後のカラー画像。これが、*Geometric*/*Blob* で使用される画像です。

前提条件

カメラはカラーカメラでなければなりません.

カラービデオフォーマットをカメラ用に構成しなければなりません.

Cognesビジョンライセンスにはカラーツールオプションが含まれていなければなりません.

カメラのホワイトバランスのキャリブレーション

カメラはデフォルト設定をされて納入されます。これらの設定には、カメラのホワイトバランスを表す3つのパラメータが含まれています。光源によっては、イメージのカラートーンが好ましくない結果になることがあります。異なる光源は異なる温度（色内容）の光を放射し、カメラはこの光を補償するために、カラーキャリブレーションを行う必要があります。

基本的なコンセプトは、赤、緑、および青を同等に含んでいるグレーシーンをカメラに撮影させることです。最も正確な方法は、白紙を1枚撮影し、そのシーンがグレーになるように、カメラの照明設定を調節することです。

下記の手順に従って、カメラのホワイトバランスをキャリブレートしてください.

1 ツリービュー内でカメラを右クリックし、設定を選択します。

［ **Camera Configuration（カメラの構成）** ］ダイアログが開きます。

2 カメラの下に白紙を置きます。この紙は視野全体をカバーする必要があります。

3 シーンが中程度のグレーに見えるように照明設定（口径または露出時間）を調節します。飽和ピクセル（完全な黒または白）の数を最少に維持します。

4 ［Calculate（計算）］を押します。これにより、ホワイトバランスのキャリブレーションパラメータが計算されます。

5 ［適用］をクリックします。

カメラの内部設定が修正されます。キャリブレーションが正常に行われると、白紙のカラーイメージとグレースケールイメージが同一（グレー）になります。

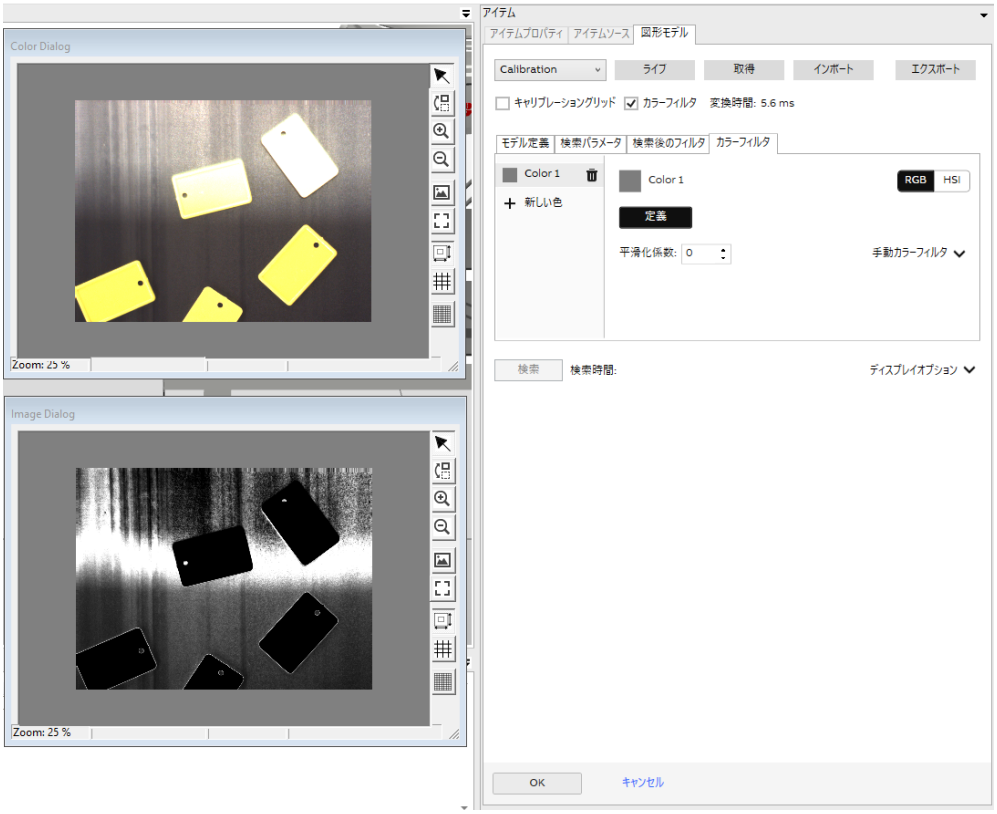
次のページに続く

© Copyright 2025 ABB. All rights reserved.

6 [OK] をクリックします。

設定はカメラに保存されます。 パラメータが保存されていない場合、PickMaster PowerPacの再起動時にカメラのキャリブレーションが失われます。

[カラーフィルタ設定] の図



xx2100002268

カラービジョンの構成

*Geometric*および*Blob*設定ダイアログには、カラー フィルタリングを有効にするチェックボックス (**カラー フィルタ**) と、フィルタ設定を表示するタブ ページが含まれています。

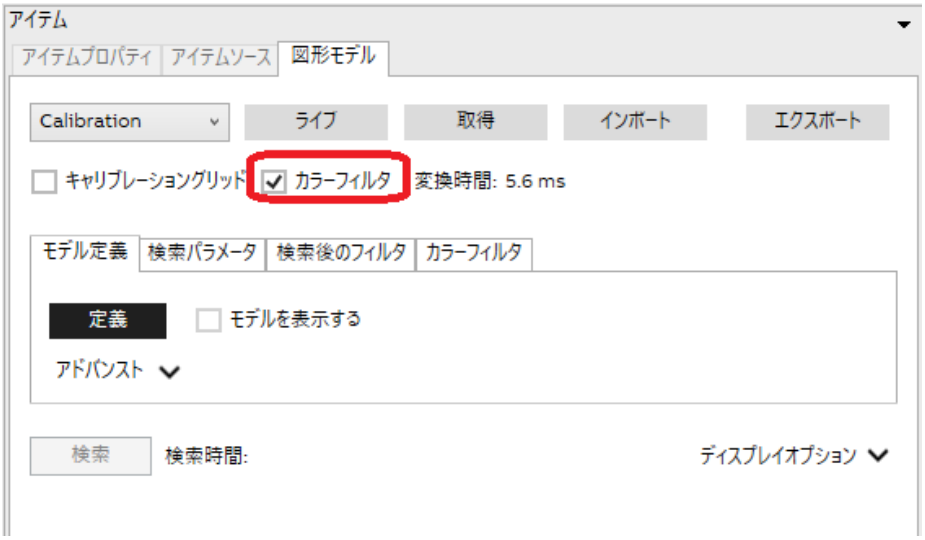
下記の手順に従ってカラービジョンを構成してください。

1 *Geometric*または*Blob*設定ダイアログで、**カラー フィルタ**を選択します。これにより、フィルタが有効になります。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.9.5 カラービジョンの使用
続き



xx2100002266

カラー フィルター設定 タブが開き、カラー画像を表示する 2 つ目のビデオウィンドウが表示されます。

2 **カラー フィルタ** タブで、**RGB** または **HSI** を選択。

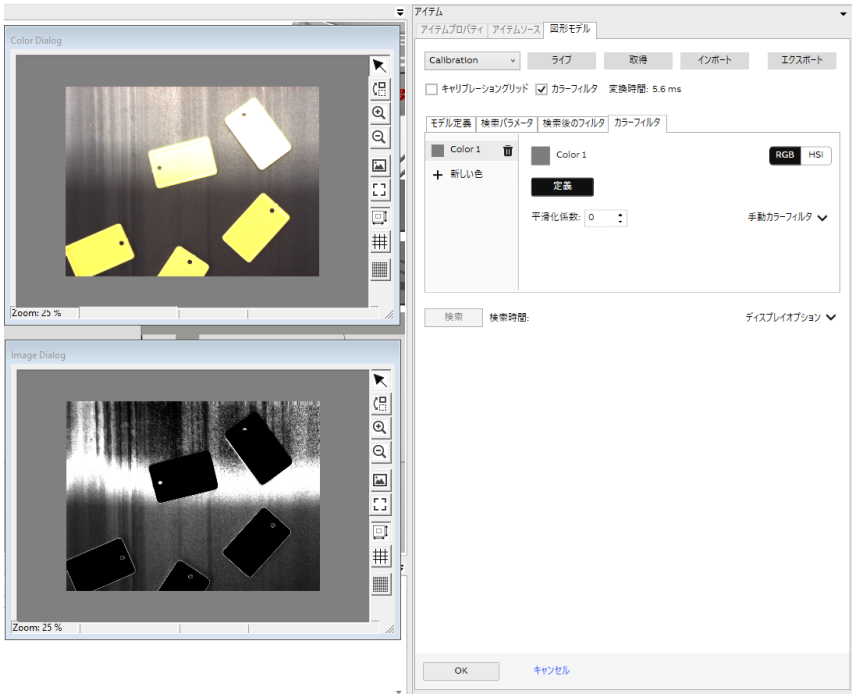
3 色彩定義タブで、画面から色彩サンプルを収集して、どの色を強調する必要があるかを示す。

a **定義** をクリックします。 調整可能な長方形がカラー ダイアログに表示されます。

b 長方形を移動 / サイズ変更することにより、フィルタを通す色を表示します。表示されたカラーレンジが、出力グレースケールイメージの白に変換されます。指定した色と似ていない色は黒に変換されます。

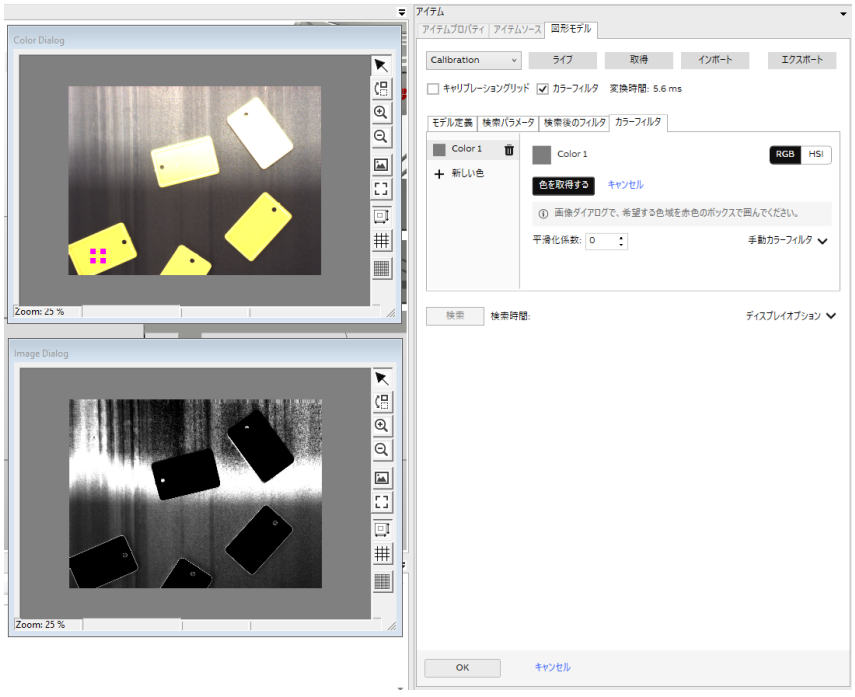
次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2100002268

c ［Get color（色を取得）］をクリックして、このカラーレンジを保存します。



xx2100002269

4 手動カラーフィルタタブで、必要に応じて各カラーチャネルを調節し、結果を改善します。

- 低い は、出力イメージで白いピクセルに変換される色範囲の下限を指定します。最小は 0、最大は 255 です。

次のページに続く

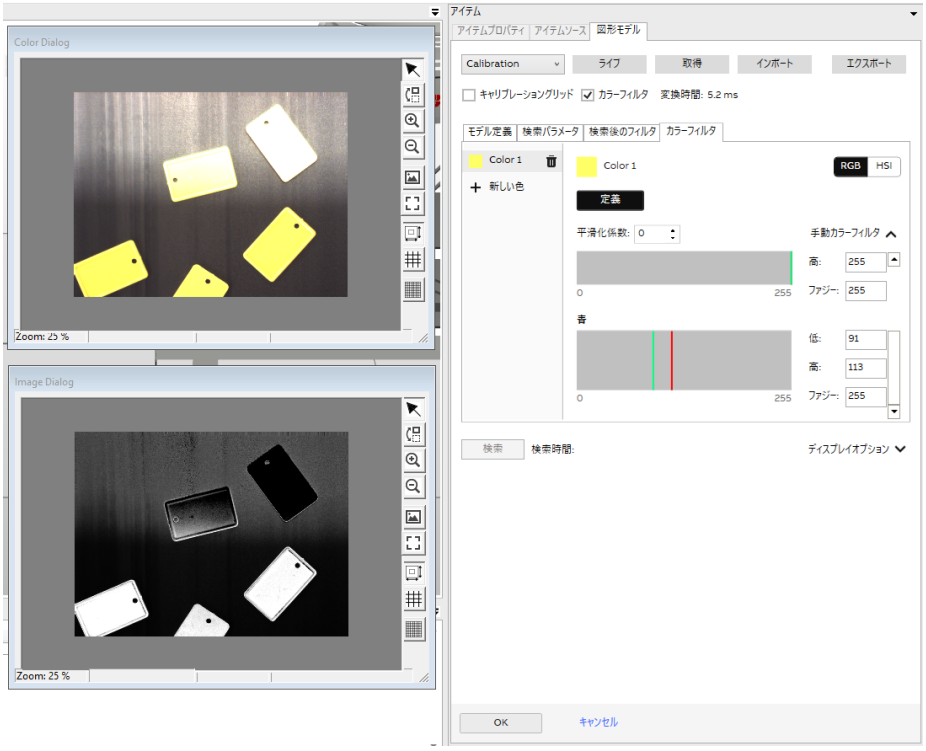
© Copyright 2025 ABB. All rights reserved.

- **高い** は、出力イメージで白いピクセルに変換される色範囲の上限を指定します。最小は 0、最大は 255 です。
- ［**Fuzzy（あいまい）**］は最低および最高しきい値の範囲外の色を、出力グレースケールイメージ用に、どのようにフィルタリングするかを指定します。値 0 は、Low（低）とHigh（高）の範囲外の色をフィルタによって完全に除去する事を示します。この場合、結果は黒と白のイメージになります。0 以外の値は、Low（低） / High（高）の範囲外の色を出力イメージ内で重み付けすることを意味しています。値が高いほど、滑らかなグレースケールイメージが得られます。最低は 0 で最高は 255 です。

5 必要に応じて、新しいカラーレンジを［**Color（色）**］セクションに付け加えます。

出力イメージの各ピクセルは、すべての個々のカラーレンジフィルタの、対応する最高出力ピクセルとして計算されます。

6 必要に応じて、滑らかさの要素を調節して、得られるグレースケールイメージのノイズを低減させます。



xx2100002270

7 オブジェクト認識モデルを定義します。

ヒント

フィルタレンジは、出力イメージのコントラストが高くなるように、なるべく低く設定します。イメージの質の観点からは、小さくて均質のカラーサンプルを選択し、色のリストに複数の範囲を付加した方が良い場合が多いです。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

ヒント

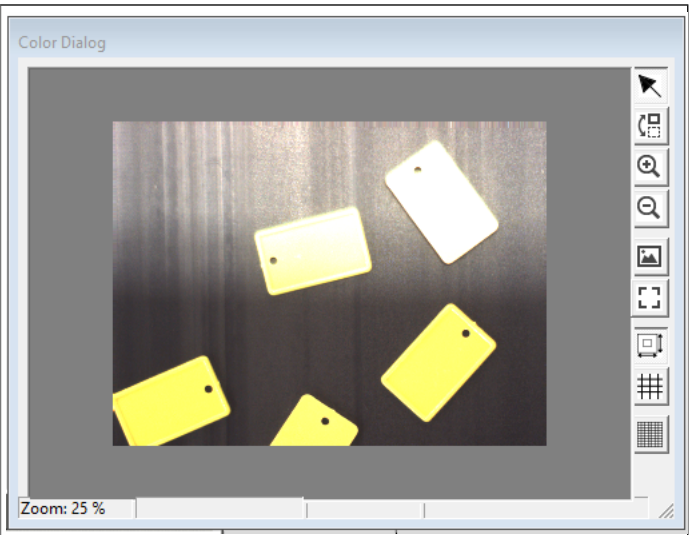
RGBとHSIの両方のフィルタを使用してみてください。時として、どちらかの一方が他方よりずっと良い場合があります。

例1

この例では、*Geometric*を使用してパーツを見つけ、*Blob*を使用して色を検査する方法について説明します。

1 検査モデルを作成します。検査モデルの構成 ページ *312*を参照します.
2 *Geometric* アライメント モデルを作成します。コントラストを上げる必要がある場合は色フィルタを使用し、コントラストが十分にある場合はフィルタされていないモノクロ画像を使用します。
3 *Blob*（輪郭）サブ検査モデルを付加します.
   a **カラーフィルタ**チェックボックスを選択します。 **カラー フィルター設定** タブが開きます。
   b ［**Define color（色の定義）**］をクリックして、検査する色を抽出します。これにより、希望する色が、Blob（輪郭）イメージウィンドウ上では白にフィルタリングされます。
   c 他のタブに切り替えて、さらに設定を行います。
   d Blob（輪郭）の設定を調節して、白の輪郭を探します。
   e 必要に応じて、カラーフィルタとBlob（輪郭）分析の設定を調節します.
4 ［Inspection Configuration（検査の構成）ダイアログで結果をチェックします.

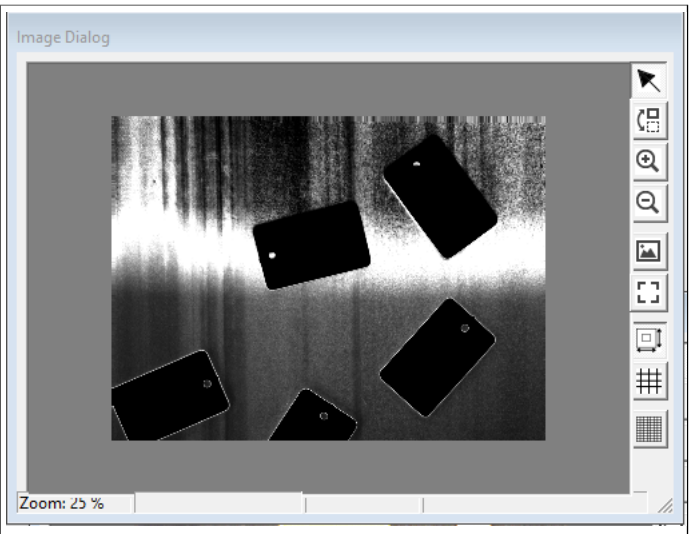
例2

フィルタ前のカラー画像	 xx1900000946

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.9.5 カラービジョンの使用
続き

フィルタ後の白黒画像	 xx1900000947

© Copyright 2025 ABB. All rights reserved.

4.4.10 生産開始

生産

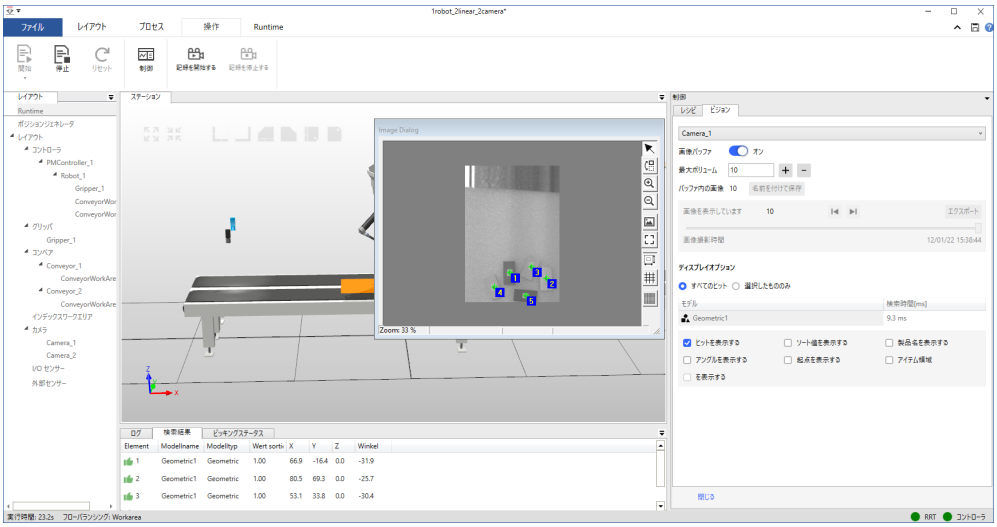
リアルコントローラとリアルRuntimeに切り替えた後、生産環境でのすべての操作はリアルセルに反映され、すべてのデータはリアルシステムから取得されます。

ソリューションのコントロールダイアログボックスを開くには、ツリービューからレシピをひとつ選択し、リボンのコントロールをクリックします。

以下の表は コントロールダイアログボックスの詳細を示す。

	説明
レシピ	生産のステータスを管理し、生産データの概要を把握します。 スタティックの詳細については、レシピ ページ 173 を参照してください。
チューニング	アイテム、作業エリア、ロボットのパラメータを調整します。 チューニングの詳細については、チューニング ページ 173 を参照してください。
フロー制御	コンベアの速度を調整します。 フロー制御の詳細については、フロー制御 ページ 178 を参照してください。
ビジョン	カメラのライブ映像をご覧ください。 ビジョンの詳細については、ビジョン ページ 333 を参照してください。

ビジョン



xx2100001638

詳しくは「詳細なビジョン情報 ページ 290」を参照してください。

エミュレーション

プロダクション実行時、PickMaster PowerPacの 3D モデルの動きを「エミュレート」と呼ぶ。

次の手順を使用してエミュレーションを行います。

1 PickMaster PowerPacリボン タブで、操作 をクリックします。
2 操作 リボン タブで、コントロール をクリックします。
コントロールダイアログが開きます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.10 生産開始
続き

3 開始 をクリックして、生産を開始します。
エミュレーションが実行を開始します。
4 停止 をクリックしてエミュレーションを停止します。

注記

生産を実行するとき、PickMaster PowerPacの3D モデルの動きは実際のシステムに従います。 ただし、PickMaster PowerPacの3D モデルの寸法はリアルセルと完全には一致しないためです。 エミュレーションでのコンベア、カメラ、I/O センサー、およびロボットのレイアウトは、エミュレーションを可能な限り実際のシステムに近づけるために、実際の寸法に応じて調整する必要がある場合があります。

エミュレーション中にアイテムが欠落している場合は、PickMaster PowerPacステーションのサイズがリアルステーションと正確に一致していないことが原因である可能性があります。 アイテムはコンベア モデルで非表示になります。

コンベア モデルの高さを調整して、アイテムが正常に表示されるようにします。

© Copyright 2025 ABB. All rights reserved.

4.4.11 生産中のロボットの管理

生産の開始

コントロールメニューから生産を開始および停止します。

生産時、ロボットは作業範囲エリアのコントロールタブからアクセスされる。さらに詳細は生産 ページ 333を参照。

事前知識

生産を開始するには、ソリューションを設定する必要があります。

レシピが開いていて有効になっている必要があります。

ピックレート

ピックレートはロボットが作動している場合、［生産］タブにアイコンで表示されます。

以下の値が表示されます：:

- **過去1分間に実行されたピック数.**
- **生産開始からの総ピック数。**

Robot states（ロボットの状態）

ロボットにはいくつかの状態があります。

状態	色	説明
Running（実行中）	緑	ロボットはアイテムのピックアンドプレイスを実行できます.
Paused（一時停止中）	赤	ロボットが、モーターオフ状態で一時停止したか、あるいはRAPIDプログラムが停止しました。
非常状態	赤	ロボットが非常停止状態にあります.
Stopped（停止済み）	赤	ロボットは停止しています。つまり、ロボットは何もアイテムを扱っていません。また、ロボットにアイテムも分配されていません。

ロボットの停止と再スタート

Runtime中にロボットを停止することができます。

［生産］タブのロボットアイコンをクリックし、ポップアップメニューからアクションを選択します。

非常停止

非常の場合：

1. **ロボットコントローラまたはFlexPendantの非常停止ボタンを押して、ロボットを即座に停止します.**

   これにより、コントローラが非常状態に設定され、警告が FlexPendantに、PickMaster PowerPacおよびRuntimeに表示されます。
2. **問題を修正します.**
3. **非常停止ボタンを解放します.**
4. **ロボットを再起動する前に、リセットボタン、またはポップアップメニューで非常状態のリセットを行います.**

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4.4.11 生産中のロボットの管理
続き

> **注意**
>
> 非常停止はロボットに不必要な負荷を与えるので、通常のプログラム停止には使用しないでください。

© Copyright 2025 ABB. All rights reserved.

5 高度な機能

5.1 UserScripts

概要

ユーザー スクリプトは、ユーザーがカスタム機能を統合するためにPickMaster Twinによって提供されるソフトウェア コンポーネントです。

この機能を使用すると、ユーザーは独自の要件に従ってアイテムのポジションの生成、調整、フィルタ、または配布をカスタマイズし、ユーザー定義のアイテムのピッキングと配置を実現できます。 たとえば、事前定義されたポジションを使用する代わりに、**ユーザー スクリプト**でポジションをクエリできます。**ユーザー スクリプト**オブジェクトで、PickMaster PowerPacのビジョン モデルによって生成されたワークエリアのポジションを調整することもできます。ワークエリアのポジションには、ユーザー スクリプトで設定できるいくつかの自由な使用パラメータが含まれています。 これらのパラメータには、後でポジションを処理するロボットから RAPID でアクセスできます。

注記

資格のある担当者のみがスクリプト ファイルを作成または変更してください。

スクリプト ファイルを使用して実行するときにセルが安全であることを確認するのは作成者の責任です。

注記

PickMaster® Twin 製品では、Python 3.12 のみがサポートされています。

PickMaster Twin 製品をインストールすると、Python 3.12 がコンピュータにインストールされ、Python 3.12 のパスがシステム環境変数に自動的に追加されます。

ヒント

構文エラーは、スクリプトファイルの実行に失敗する原因となります。

以下の方法で、シンタックスエラーを回避できます。

1 同じスクリプトファイルには、同じエディタを使用するようにしてください。
2 スクリプトファイルの編集には、PyCharmないしNotepad++の使用が、Pythonファイルの構文チェック機能を有するので、推奨される。

注意

ローカルプレゼンスが正しい方法で設定されるように実装するのは、インテグレータの責任です。

単一の制御ポイントが正しい方法で設定されるように実装するのは、インテグレータの責任です。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

5.1 UserScripts
続き

危険

スクリプトを生産環境で使用する場合は、慎重に保護してください。

スクリプトにアクセスできる人は誰でも、スクリプトを直接変更できます。 これは重大な危険を引き起こす可能性があります。

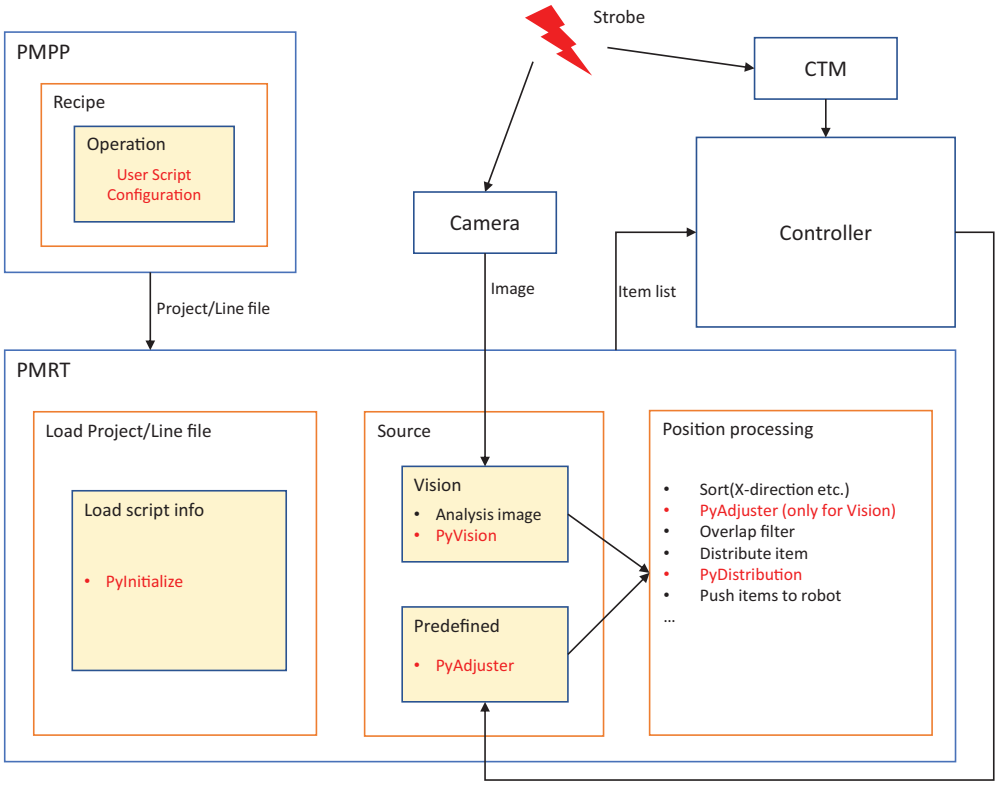
注記

ひとつのレシピでUserScriptsと外部センサーを同時に使用することはできない。

注記

Pythonスクリプトファイルは Pack& Go ファイルには含まれない。Pythonスクリプトファイルを目的の保存先にコピー。

フローチャート



xx2400000705

アプリケーション シナリオ

ユーザー スクリプトは、PMTW によってユーザーに提供される高度な機能であり、次のシナリオで使用できます。

1 アイテムの生成と削除

ユーザーは、お客様の実際のニーズを満たすために、要件に従ってスクリプト内のワークエリアの生成と配置をカスタマイズできます。

2 アイテムのピッキングとプレースのポジションを調整する

次のページに続く

© Copyright 2025 ABB. All rights reserved.

ユーザーは、要件に応じてスクリプト内でワークエリアのポジションを静的またはリアルタイムに調整でき、リアルタイムの調整と素材のポジションの高精度に対するユーザーの要件を満たします。

3 アイテムのフィルタリングと並べ替え

ユーザーは、アイテムのスクリーニングとキャプチャ順序に対するユーザーの要件を満たすために、要件に従ってスクリプト内の現在のアイテムをフィルタリングおよび並べ替えることができます。

4 アイテム配布戦略の調整

現在の配布戦略は、LoadBalance とATCです。ユーザーは、ユーザー定義の配布の要件を満たすように現在の配布戦略を調整できます。

5 アイテムの識別

ビジョン インターフェイスを使用すると、ユーザーはカメラで撮影した写真をさらに処理して、新しいアイテム情報を識別できます。

6 バインド追加情報

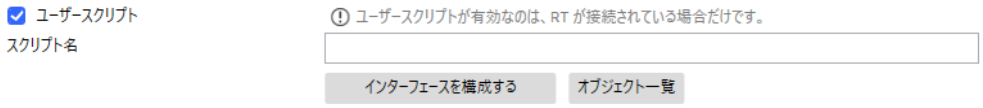
インターフェイス パラメータには 5 つのオプションのパラメータが提供されます。ユーザーはオプションのパラメータを設定して追加情報をマテリアルにバインドし、ソフトウェアを通じてロボットに送信して、アイテムコードのバインドやアイテムトラッキングなどの特別な機能を実現できます。

ユーザーの価値

この機能により、ソフトウェアの活用シーンが広がります。 ユーザーは、独自のニーズに応じてソフトウェアの標準機能をカスタマイズできます。これにより、オブジェクトのカスタム生成、ピッキング、プレース、並べ替え、フィルタリング、および配布の機能を実現でき、さまざまな特定のアプリケーションシナリオに対するユーザーのニーズを満たすことができ、改善されます。ピッキング精度と生産効率を向上させ、ユーザーにとってより多くの価値を創造します。

設定の概要

ユーザー スクリプトのチェックボックスを選択すると、**ユーザー スクリプト**が表示される。



xx2200001779

	説明
スクリプト名	事前定義済みのスクリプト ファイル名を .py で入力します。 **ヒント** スクリプト機能を使用する前に、定義済みのスクリプト ファイルを C:\Users\xxxx\Documents\PickMaster\PMScripts フォルダに配置する必要があります。
インターフェイスの設定	使用するユーザー スクリプト インターフェイスを選択します。 4 種類のユーザー スクリプト インターフェースがPickMaster Twinでサポートされる。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

	説明
物体一覧	現在の操作で使用可能なすべてのオブジェクト (名前と ID) を表示します。

サポートされているユーザー スクリプト インターフェイスの種類の概要

PickMaster Twinは、4 種のユーザー スクリプトをサポートします。

ユーザー スクリプト インターフェイス	説明
インターフェイスの初期化	このインターフェイスは、パラメータの初期化など、ユーザー スクリプト プログラムを初期化するためにユーザーに提供するために使用されます。  ヒント インターフェイスの初期化は、起動 をクリックしたときに 1 回だけ実行されます。 他の 3 つのインターフェイスは、DSQC 2000 またはDSQC 377 シグナルがトリガーされたときに実行されます。 詳細は、インターフェイスの初期化*PyInitialize*：データの初期化 ページ *342*を参照してください。
アジャスタ インターフェイス	このインターフェースは、カスタマイズされたアイテムのポジションの生成と調整を実現するためにユーザーに提供するために使用されます。 モデルがポジションを生成するたびに、ポジションの配列がユーザー スクリプト の物体に送信されます。 その後、ユーザー スクリプト の物体は、任意の方法でポジションを制御できます。ポジションは、変更、削除、または追加できます。 詳細は、アジャスタ インターフェース *PyAdjuster*：ポジションの変更 ページ *343*を参照してください。
ビジョン インターフェース	このインターフェースは、ビジョン結果によるカスタマイズされたアイテムポジションフィルタと調整を実現するためにユーザーに提供するために使用されます。 このインターフェイスは、Runtimeが本番環境でアイテム認識セクションを実行するときに呼び出されます。  ヒント ビジョン インターフェース は生産環境でのみ使用できます。 他の 3 つのインターフェイスは、生産およびシミュレーションで使用できます。 詳細は、ビジョン インターフェース *PyVision:* 画像を再解析してアイテムを認識する ページ *344*を参照してください。
配信インターフェース	このインターフェースは、カスタマイズされた配信機能を実現するためのユーザーに提供するために使用されます。 このインターフェースは、アイテム配信が実行されるときに呼び出されます。 詳細は、配信インターフェース *PyDistribution:* 配信後、ロボットにプッシュする前に対象アイテム情報を調整する ページ *348*を参照してください。

UserScripts機能の設定

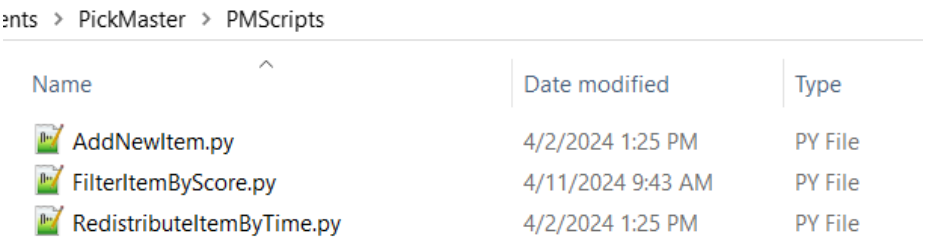
次の手順に従って、ユーザー スクリプト機能を設定します。

1 事前定義されたスクリプト ファイルを宛先フォルダに配置します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

> **ヒント**
>
> スクリプト機能を使用する前に、定義済みのスクリプト ファイルを `C:\Users\xxxx\Documents\PickMaster\PMScripts` フォルダに配置する必要があります。



xx2200001784

2 PickMaster Powerpac **Recipe設定**の**ユーザー スクリプト**チェックボックスを選択して、設定ページを開きます。
3 事前定義済みのスクリプト ファイル名を**スクリプト名**テキスト ボックスに入力します。
4 **インターフェイスの設定** をクリックして、インターフェイスのタイプのページを開きます。
5 ポップアップ ページで、目的のインターフェイス タイプを選択します。

> **ヒント**
>
> 4タイプ同時に使用できます。

6 **完了**をクリックして、PickMaster Powerpacでのユーザー スクリプト機能の設定を終了します。
7 Runtime設定ファイル*PickMasteru.exe.config*でタイムアウト値を設定します。

タイムアウト設定の詳細については、ユーザー スクリプトのタイムアウト設定 ページ *342*を参照してください。

> **ヒント**
>
> Runtime設定ファイル*PickMasteru.exe.config*の宛先フォルダ:
>
> - **VRT:** `C:\Program Files (x86)\ABB\PickMaster Twin 3\PickMaster Twin Runtime 3\PickMaster VirtualRuntime`
> - **RRT:** `C:\Program Files (x86)\ABB\PickMaster Twin 3\PickMaster Twin Runtime 3\PickMaster Runtime`

次のページに続く

© Copyright 2025 ABB. All rights reserved.

ユーザー スクリプトのタイムアウト設定

ユーザー スクリプトの実行時間制限を設定して、ユーザー スクリプトの過剰な実行時間によって引き起こされるPickMaster Twin製品の例外を回避します。

テンプレート	キー	値 (テンプレート内)	説明
`<add key="PyInitializeTimeout" value ="1500"/>`	`PyInitializeTimeout`	**x (1500)**	PyInitializeインターフェイスのタイムアウトはx (1500)ミリ秒です。 実行時間が設定値を超えると、ログビューに警告が表示されます。
`<add key="PyAdjusterTimeout" value ="1500"/>`	`PyAdjusterTimeout`	**x (1500)**	PyAdjusterインターフェイスのタイムアウトはx (1500)ミリ秒です。
`<add key="PyVisionTimeout" value ="1500"/>`	`PyVisionTimeout`	**x (1500)**	PyVisionインターフェイスのタイムアウトはx (1500)ミリ秒です。
`<add key="PyDistributionTimeout" value ="1500"/>`	`PyDistributionTimeout`	**x (1500)**	PyDistributionインターフェイスのタイムアウトはx (1500)ミリ秒です。
`<add key="MaxTimeoutCount" value ="5"/>`	`MaxTimeoutCount`	**x (5)**	各インターフェースの最大連続タイムアウトはx (5)回です。 連続タイムアウト数が設定された最大値を超えると、Runtimeはインターフェイス機能の呼び出しを停止し、すべてのオブジェクトをクリアしてエラー ログを表示し、ステーションを停止してスクリプトを確認するようにユーザーに通知します。

ユーザー スクリプト インターフェイス

インターフェイスの初期化PyInitialize：データの初期化

このインターフェイスは、スクリプトを初期化し、現在のRT情報、アイテム情報、コンテナ情報、およびワークエリア情報をユーザー スクリプトに転送するために使用されます。ユーザー スクリプトは、新しいアイテムの作成など、ユーザーが処理できます。同時に、ユーザーは、ステーションの起動と同時に開始できる外部プログラムの起動など、このインターフェイスにユーザー プログラムの初期化操作を追加できます。

引数	説明	説明	例では：
タイプ	Runtime タイプ	• 0:VRT • 1:RRT	

アイテム情報	**アイテム情報**
	これには `{Key}:{Name:{} Id:{}}`が含まれます
	例：
	`itemInfo=`
	`{`
	`'0':{'Name':Item_1,`
	`'Id':3[illegible]},`
	`};`
		`{Key}`
		キー：一意のインデックス
			`'0'`

		`Name:{}` **名前：アイテムの名前**	`'Name':Item_1`
		`Id:{}` **Id：アイテムの ID**	`'Id':3[illegible]`

詳細については、*AddNewItem.py ページ 350*と*RedistributeItemByTime.py ページ 354*を参照してください。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

アジャスタ インターフェース PyAdjuster：ポジションの変更

引数	説明	説明	例では：

アイテム	アイテム情報は、
	Time:{} {Key}:{X:{} Y:{} Z:{} RX:{} RY:{} RZ:{} Tag:{} Val1:{} Val2:{} Val3:{} Val4:{} Val5:{} Level:{} Id:{}}を有します。
	例：
	items =
	{
	'Time': 1666849507.969,
	'0': {'X': 0.0,
	'Y': 150.0,
	'Z': 0.0,
	'RX': 0.0,
	'RY': 0.0,
	'RZ': 0.0,
	'Tag': 0,
	'Val1': 0.0,
	'Val2': 0.0,
	'Val3': 0.0,
	'Val4': 0.0,
	'Val5': 0.0,
	'Level': 2,
	'Id': '[illegible]'
	}
	}
		Time:{}
		時間: タイムスタンプ、1970 年 1 月 1 日からのミリ秒数を取得
			'Time': 1666849507.969,

		{Key} キー：一意のインデックス	'0'
		X:{} X: X 方向のアイテムのポジションの値	'X': 0.0
		Y:{} Y: Y 方向のアイテムのポジションの値	'Y': 150.0
		Z:{} Z: Z 方向のアイテムのポジションの値	'Z': 0.0
		RX:{} RX: X 方向のアイテムの回転角の値	'RX': 0.0
		RY:{} RY: Y 方向のアイテムの回転角の値	'RY': 0.0
		RZ:{} RZ: Z 方向のアイテムの回転角の値	'RZ': 0.0
		Tag:{} タグ: Rapidで使用	'Tag': 0
		Val1:{} Val1、Val2、Val3、Val4、Val5: オプションの値、Rapidで使用	'Val1': 0.0
		Val2:{} Val2: オプションの値、Rapid で使用されます。	'Val2': 0.0
		Val3:{} Val3: オプションの値、Rapid で使用されます。	'Val3': 0.0
		Val4:{} Val4: オプションの値、Rapid で使用されます。	'Val4': 0.0
		Val5:{} Val5: オプションの値、Rapid で使用されます。	'Val5': 0.0
		Level:{} レベル: 検査レベル • 0: 破棄 • 1: 拒否 • 2: 許可	'Level': 2

次のページに続く

© Copyright 2025 ABB. All rights reserved.

引数	説明	説明	例では：
		Id:{} Id：アイテムの ID	'Id': '[illegible]' }

詳細な例については、*AddNewItem.py ページ 350*を参照してください。

ビジョン インターフェース PyVision: 画像を再解析してアイテムを認識する

引数	説明	説明	

画像データ	画像データはWidth:{} Height:{} IsColor:{} Grey:{} Blue:{} Green:{} Red:{}を有します。
	例：
	Grey image
	imageData =
	{
	'Width': 481,
	'Height': 409,
	'IsColor': 0,
	'Grey': [56,…,67]
	}
	Colorful image
	imageData =
	{
	'Width': 481,
	'Height': 409,
	'IsColor': 1,
	'Blue': [56,…,67],
	'Green': [56,…,67],
	'Red': [56,…,67]
	}
		Width:{}
		• 幅: 画像の幅 (ピクセル単位)
			'Width': 481,

		Height:{} • 高さ: 画像の高さ (ピクセル単位)	'Height': 409,
		IsColor:{} • Grey:{} • Blue:{} Green:{} Red:{}. • Is色: - 0：グレーの画像 - 1：カラフルな画像 • グレー: 0 から 255 まで有効なグレーのデータ • 青: 青のデータ、0 から 255 まで有効 • 緑: 緑のデータ、0 から 255 まで有効 • 赤: 赤のデータ、0 から 255 まで有効	Grey image用 'IsColor': 0, 'Grey': [56,…,67] Colorful image用 'IsColor': 1, 'Blue': [56,…,67], 'Green': [56,…,67], 'Red': [56,…,67]

次のページに続く

© Copyright 2025 ABB. All rights reserved.

引数	説明	説明	

calibData	校正データは
	UpperLeftX:{}
	UpperLeftY:{}
	LowerRightX:{}
	LowerRightY:{}
	XScale:{} YScale:{}
	を有します。
	例：
	calibData =
	{
	'UpperLeftX': -313,
	'UpperLeftY': -265,
	'LowerRightX': 168,
	'LowerRightY': 144,
	'XScale': 0.415,
	'YScale': 0.415
	}
		UpperLeftX:{}
		• 左上X: 座標系の X 方向の左上点 (ピクセル単位)
			'UpperLeftX': -313,

		UpperLeftY:{} • 左上Y: 座標系の Y 方向の左上点 (ピクセル単位)	'UpperLeftY': -265,
		LowerRightX:{} • 右下X: 座標系の X 方向の右下の点 (ピクセル単位)	'LowerRightX': 168,
		LowerRightY:{} • 右下Y: 座標系の Y 方向の右下点 (ピクセル単位)	'LowerRightY': 144,
		XScale:{} • Xスケール: 実際のアイテムと画像の X 軸スケール (ピクセル単位)	'XScale': 0.415,
		YScale:{} • Yスケール: 実際のアイテムと画像の Y 軸スケール (ピクセル単位)。	'YScale': 0.415

次のページに続く

© Copyright 2025 ABB. All rights reserved.

続き

引数	説明	説明	
アイテム	アイテム情報は以下を有する：Time:{}そして • 幾何学図形： {Key}:{X:{} Y:{} Z:{} RZ:{} SortValue:{} ZValid:{} XImgPos:{} YImgPos:{} Val1:{} Val2:{} Val3:{} Val4:{} Val5:{} Level:{} Id:{} ModelType:{} Score:{} XScale:{} YScale:{} Contrast:{} FitError:{} Coverage:{} Clutter:{}} • 輪郭： {Key}:{X:{} Y:{} Z:{} RZ:{} SortValue:{} ZValid:{} XImgPos:{} YImgPos:{} Val1:{} Val2:{} Val3:{} Val4:{} Val5:{} Level:{} Id:{} ModelType:{} Area:{} Perimeter:{} Elongation:{} Circularity:{}} • 検査： {Key}:{X:{} Y:{} Z:{} RZ:{} SortValue:{} ZValid:{} XImgPos:{} YImgPos:{} Val1:{} Val2:{} Val3:{} Val4:{} Val5:{} Level:{} Id:{} ModelType:{}} 例： 幾何学模様	Time:{} • 時間: タイムスタンプ、1970 年 1 月 1 日からのミリ秒数を取得	'Time': 1666849507.969,
	{Key} キー：一意のインデックス	'0’
	X:{} • X: X 方向のアイテムのポジションの値	'X’: -80.1,
	Y:{} • Y: Y 方向のアイテムのポジションの値	'Y’: -77.2,
	Z:{} • Z: Z 方向のアイテムのポジションの値	'Z': 0.0,
	RZ:{} • RZ:Z方向のアイテムの回転角の値	‘RZ’: -7.22,
	SortValue:{} • SortValue: ソート値	‘SortValue’: 0.976,
	ZValid:{} • Z有効: - 1: 有効 - 0: 無効	‘ZValid': 0,
	XImgPos:{} • XImgPos: X 方向の画像内のアイテムのポジション •	‘XImgPos': -80.1,
	YImgPos:{} • YImgPos: Y 方向の画像内のアイテムのポジション	‘YImgPos': -77.2,
	Val1:{} Val2:{} Val3:{} Val4:{} Val5:{} • Val1、Val2、Val3、Val4、Val5: オプションの値、Rapidで使用	'Val1': 0.0, 'Val2': 0.0, 'Val3': 0.0, 'Val4': 0.0, 'Val5': 0.0,
	Level:{} • レベル: 検査レベル - 0: 破棄 - 1: 拒否 - 2: 許可	'Level':2,
	Id:{} • Id：アイテムの ID	'Id': '5[illegible]',

次のページに続く

© Copyright 2025 ABB. All rights reserved.

引数	説明	説明	

	resResult = {
	'Time':
	1666849507.969,
	'0':{'X': -80.1,
	'Y': -77.2,
	'Z': 0.0,
	'RZ': -7.22,
	'SortValue': 0.976,
	'ZValid': 0,
	'XImgPos': -80.1,
	'YImgPos': -77.2,
	'Val1': 0.0,
	'Val2': 0.0,
	'Val3': 0.0,
	'Val4': 0.0,
	'Val5': 0.0,
	'Level':2,
	'Id':
	'3519a6c-5a8-437d-b80-7f4e4bf6ff',
	'ModelType':1,
	'Score':0.747174859046936,
	'XScale':0.9995959997177124,
	'YScale':0.9995959997177124,
	'Contrast':12.289325714111328,
	'FitError':0.3699681468573914,
	'Coverage':0.747174859046936,
	'Clutter':0.10466811060905457
	}
	}
	ブロブ
	resResult = {
	'Time':
	1666849507.969,
	'0':{'X': -80.1,
	'Y': -77.2,
	'Z': 0.0,
	'RZ': -7.22,
	'SortValue': 0.976,
	'ZValid': 0,
	'XImgPos': -80.1,
	'YImgPos': -77.2,
	'Val1': 0.0,
	'Val2': 0.0,
	'Val3': 0.0,
	'Val4': 0.0,
	'Val5': 0.0,
	'Level':2,
	'Id':
	'3519a6c-5a8-437d-b80-7f4e4bf6ff',
	'ModelType':2,
		ModelType:{}
		• モデルタイプ:
		- 1: 幾何学模様
		- 2: ブロブ
		- 3: 検査
			ジオマトリック(幾何学模様)の場合：
			'ModelType':1,
			BLOB の場合：
			'ModelType':2,
			検査用：
			'ModelType':3 }

		詳しくは「*Geometricモデルの設定 ページ296*」を参照してください。 Score:{} • スコア: 見つかったアイテムがトレーニング済みのモデルとどの程度一致しているか。	'Score':0.747174859046936,
		XScale:{} • Xスケール: 実際のアイテムと画像の X 軸スケール (ピクセル単位)	'XScale':0.9995959997177124,
		YScale:{} • Yスケール: 実際のアイテムと画像の Y 軸スケール (ピクセル単位)。	'YScale':0.9995959997177124,
		Contrast:{} • コントラスト: 画像内にある各アイテムの画像コントラスト。	'Contrast':12.289325714111328,
		FitError:{} • FitError: トレーニングされたパターンの形状と検索画像で見つかったパターンの形状との間のバリエーションの尺度。	'FitError':0.3699681468573914,
		Coverage:{} • カバレッジ: トレーニングされたパターンのすべての部分が検索画像にも存在する程度の尺度。	'Coverage':0.747174859046936,
		Clutter:{} • クラッター: 見つかったパターンに、トレーニングされたパターンには存在しない特徴が含まれている程度の尺度。	'Clutter':0.10466811060905457 }
			'Area':0,

次のページに続く

© Copyright 2025 ABB. All rights reserved.

引数	説明	説明	

	'Area':0,
	'Perimeter':0,
	'Elongation':0,
	'Circularity':0 }
	}
	検査
	resResult = {
	'Time':
	1666849507.969,
	'0':{'X': -80.1,
	'Y': -77.2,
	'Z': 0.0,
	'RZ': -7.22,
	'SortValue': 0.976,
	'ZValid': 0,
	'XImgPos': -80.1,
	'YImgPos': -77.2,
	'Val1': 0.0,
	'Val2': 0.0,
	'Val3': 0.0,
	'Val4': 0.0,
	'Val5': 0.0,
	'Level':2,
	'Id':
	'[illegible]',
	'ModelType':3 }
	}
		詳しくは「*Blobモデルの構成 ページ305*」を参照してください。
		Area:{}
		• 面積：mm²で表される	

		Perimeter:{} • 周長：mmで表されます	'Perimeter':0,
		Elongation:{} • 伸び: フィーチャーの第 2 主軸周りの二次慣性モーメントと、フィーチャーの第 1 主軸周りの二次慣性モーメントの比率。	'Elongation':0,
		Circularity:{} • 円形度: 円形度を定義します。 値 1 は、完全な円形で完全に塗りつぶされている (穴がない) ことを意味します。	'Circularity':0 }

詳細な例については、*FilterItemByScore.py ページ352*を参照してください。

配信インターフェース PyDistribution: 配信後、ロボットにプッシュする前に対象アイテム情報を調整する

引数	説明		説明
WaId	作業範囲 IDはWaId:{}を有します。 例： WaId = ([illegible])	WaId:{} • WaId: 作業エリア ID	WaId = ([illegible])

次のページに続く

© Copyright 2025 ABB. All rights reserved.

引数	説明		説明

アイテム	アイテム情報は、Time:{} {Key}: {X:{} Y:{} Z:{} q1:{} q2:{} q3:{} q4:{} Val1:{} Val2:{} Val3:{} Val4:{} Val5:{} Type:{} Tag:{} Index:{} State:{} Container:{} Layer:{} Group:{} Id:{}}を有します。
	例：
	items =
	{
	'Time': 1666849507.969,
	'0':{'X': 0.0,
	'Y': 150.0,
	'Z': 0.0,
	'q1': 0.0,
	'q2': 1.0,
	'q3': 0.0,
	'q4': 0.0,
	'Val1': 0.0,
	'Val2': 0.0,
	'Val3': 0.0,
	'Val4': 0.0,
	'Val5': 0.0,
	'Type': 2,
	'Tag': 0,
	'Index': 2,
	'State': 0,
	'Container': 1,
	'Layer': 1,
	'Group': 0,
	'Id': '[illegible]'
	}
	}
		Time:{}
		• 時間: タイムスタンプ、1970 年 1 月 1 日からのミリ秒数を取得
			'Time': 1666849507.969,

		{Key} キー：一意のインデックス	'0'
		X:{} • X: X 方向のアイテムのポジションの値	{'X': 0.0,
		Y:{} • Y: Y 方向のアイテムのポジションの値	'Y': 150.0,
		Z:{} • Z: Z 方向のアイテムのポジションの値	'Z': 0.0,
		q1:{} • q1、q2、q3、q4: アイテムの四元数の値	'q1': 0.0,
		q2:{}	'q2': 1.0,
		q3:{}	'q3': 0.0,
		q4:{}	'q4': 0.0,
		Val1:{} Val2:{} Val3:{} Val4:{} Val5:{} • Val1、Val2、Val3、Val4、Val5: オプションの値、Rapidで使用	'Val1': 0.0, 'Val2': 0.0, 'Val3': 0.0, 'Val4': 0.0, 'Val5': 0.0,
		Type:{} • インデックス：承認されたタイプまたは拒否されたタイプのインデックス番号	'Type': 2,
		Tag:{} • タグ：Rapid で使用	'Tag': 0,
		Index:{} • インデックス：現在のアイテムのシーケンス番号。コンベア上のアイテムの生成やコンテナ内の事前定義されたレイアウトに応じて増加します。	'Index': 2,
		State:{}	'State': 0,

次のページに続く

© Copyright 2025 ABB. All rights reserved.

引数	説明		説明
		• 状態：アイテムの状態 - 0: 使用 - 1: バイパス - 2: 使用済み	
		Container:{} • コンテナ：コンテナ番号 - 0：アイテムです - 1-n：コンテナです	'Container': 1,
		Layer:{} • レイヤー：レイヤー番号 - 0：アイテムです - 1-n：コンテナ内のレイヤーです	'Layer': 1,
		Group:{} • グループ: ソート方法 - 0：なしまたは移動方向 - 1: 厳密	'Group': 0,
		Id:{} • Id：アイテムの ID	'Id': '[illegible]' }

詳細な例については、*RedistributeItemByTime.py* ページ *354*を参照してください。

テンプレート

すべてのユーザー スクリプト テンプレートは、PickMaster クライアントがインストールされている場合、フォルダC:\Program Files (x86)\ABB\PickMaster Twin 3\Samples\UserScriptTemplatesにも提供されます。

AddNewItem.py

```
# PMTW user script demo -- AddNewItem
# Add a new item in the default item list.

# Global definition
RTType = 1
item_1 = r''
container_1 = r''
newObject = {'X': 100.0, 'Y': 50.0, 'Z': 5.0, 'RX': 0.0, 'RY': 0.0,
     'RZ': 0.0, 'Tag': -1, 'Val1': 0.0, 'Val2': 0.0, 'Val3': 0.0,
     'Val4': 0.0, 'Val5': 0.0, 'Level': 2, 'Id':
     '5FAD0398-74F8-4786-BF2A-1225924A8A41'}
# This path need to be created by users.
logPath = r'C:\PMScriptsLog\PickInfo.txt'
```

次のページに続く

© Copyright 2025 ABB. All rights reserved.

```
# PyInitialize interface
def PyInitialize(type, itemInfo):
  global RTType
  global item_1
  global container_1
  RTType = type
  f = open(logPath,'a')
  f.write("PyInitialize\n")
  # RT type
  strLine = "RTType:{}\n".format(str(RTType))
  f.write(strLine)
  # Item information
  keys = itemInfo.keys();
  for key in keys:
    strLine = "{} Name:{} Id:{}\n".format(str(key),
         str(itemInfo[key]['Name']), str(itemInfo[key]['Id']))
    if itemInfo[key]['Name'] == 'Item_1':
      item_1 = itemInfo[key]['Id']
    elif itemInfo[key]['Name'] == 'Container_1':
      container_1 = itemInfo[key]['Id']
    f.write(strLine)
  f.close()
# PyAdjuster interface
def PyAdjuster(items):
  global RTType
  global item_1
  global container_1
  global newObject
  f = open(logPath, 'a')
  f.write("PyAdjuster\n")
  # Modify Id
  newObject['Id'] = item_1
  #newObject['Id'] = container_1

  # Add new item
  iSize = len(items)
  newKey = str(iSize - 1)
  items[newKey] = newObject

  # Item information
  keys = items.keys()
  for key in keys:
    if key == 'Time':
      # Time stamp(s), get the number of milliseconds since 1 Jan
           1970.
      strLine = "Time:{}\n".format(str(items[key]))
      f.write(strLine)
    else:
      # Print
```

次のページに続く

© Copyright 2025 ABB. All rights reserved.

続き

```
                    strLine = "{} X:{} Y:{} Z:{} RX:{} RY:{} RZ:{} Tag:{} Val1:{}
                         Val2:{} Val3:{} Val4:{} Val5:{} Level:{}
                         Id:{}\n".format(str(key), str(items[key]['X']),
                         str(items[key]['Y']), str(items[key]['Z']),
                         str(items[key]['RX']), str(items[key]['RY']),
                         str(items[key]['RZ']), str(items[key]['Tag']),
                         str(items[key]['Val1']), str(items[key]['Val2']),
                         str(items[key]['Val3']), str(items[key]['Val4']),
                         str(items[key]['Val5']), str(items[key]['Level']),
                         items[key]['Id'])
                    f.write(strLine)
                f.close()
                return items;
```

FilterItemByScore.py

```
# PMTW user script demo -- FilterItemByScore
# Filter item according to score value.

# Global definition
# This path need to be created by users.
logPath = r'C:\PMScriptsLog\PlaceInfo.txt'

# PyVision interface
def PyVision(imageData,calibData,items):
  f = open(logPath, 'a')
  f.write("PyVision\n")
  # Image data
  f.write("ImageData:\n")
  strLine = "Width:{} Height:{}
       IsColor:{}\n".format(str(imageData['Width']),
       str(imageData['Height']), str(imageData['IsColor']))
  f.write(strLine)
  if imageData['IsColor'] == 0:
    strLine = "Grey:{}\n".format(str(imageData['Grey']))
  else:
    strLine =
         "Blue:{}\nGreen:{}\nRed:{}\n".format(str(imageData['Blue']),
         str(imageData['Green']), str(imageData['Red']))
  f.write(strLine)
  # Calibration data
  f.write("CalibrationData:\n")
  strLine = "UpperLeftX:{} UpperLeftY:{} LowerRightX:{}
       LowerRightY:{} XScale:{}
       YScale:{}\n".format(str(calibData['UpperLeftX']),
       str(calibData['UpperLeftY']), str(calibData['LowerRightX']),
       str(calibData['LowerRightY']), str(calibData['XScale']),
       str(calibData['YScale']))
  f.write(strLine)
  # Item information
  f.write("Items:\n")
  keys = items.keys();
  for key in keys:
```

次のページに続く

© Copyright 2025 ABB. All rights reserved.

```
if key == 'Time':
  # Time stamp(s), get the number of milliseconds since 1 Jan
      1970.
  strLine = "Time:{}\n".format(str(items[key]))
  f.write(strLine)
else:
  if items[key]['ModelType'] == 1:
    # Geomatric
    strLine = "{} X:{} Y:{} Z:{} RZ:{} SortValue:{} ZValid:{}
         XImgPos:{} YImgPos:{} Val1:{} Val2:{} Val3:{} Val4:{}
         Val5:{} Level:{} Id:{} ModelType:{} Score:{} XScale:{}
         YScale:{} Contrast:{} FitError:{} Coverage:{}
         Clutter:{}\n".format(str(key), str(items[key]['X']),
         str(items[key]['Y']), str(items[key]['Z']),
         str(items[key]['RZ']), str(items[key]['SortValue']),
         str(items[key]['ZValid']), str(items[key]['XImgPos']),
         str(items[key]['YImgPos']), str(items[key]['Val1']),
         str(items[key]['Val2']), str(items[key]['Val3']),
         str(items[key]['Val4']), str(items[key]['Val5']),
         str(items[key]['Level']), items[key]['Id'],
         str(items[key]['ModelType']),
         str(items[key]['Score']), str(items[key]['XScale']),
         str(items[key]['YScale']),
         str(items[key]['Contrast']),
         str(items[key]['FitError']),
         str(items[key]['Coverage']),
         str(items[key]['Clutter']))
    # Filter
    if items[key]['Score'] < 0.8:
      items[key]['Level'] = 0
  elif items[key]['ModelType'] == 2:
    # Blob
    strLine = "{} X:{} Y:{} Z:{} RZ:{} SortValue:{} ZValid:{}
         XImgPos:{} YImgPos:{} Val1:{} Val2:{} Val3:{} Val4:{}
         Val5:{} Level:{} Id:{} ModelType:{} Area:{}
         Perimeter:{} Elongation:{}
         Circularity:{}\n".format(str(key),
         str(items[key]['X']), str(items[key]['Y']),
         str(items[key]['Z']), str(items[key]['RZ']),
         str(items[key]['SortValue']),
         str(items[key]['ZValid']), str(items[key]['XImgPos']),
         str(items[key]['YImgPos']), str(items[key]['Val1']),
         str(items[key]['Val2']), str(items[key]['Val3']),
         str(items[key]['Val4']), str(items[key]['Val5']),
         str(items[key]['Level']), items[key]['Id'],
         str(items[key]['ModelType']), str(items[key]['Area']),
         str(items[key]['Perimeter']),
         str(items[key]['Elongation']),
         str(items[key]['Circularity']))
    # Filter
    if items[key]['Score'] < 0.8:
      items[key]['Level'] = 0
  else:
    # Inspection
```

次のページに続く

© Copyright 2025 ABB. All rights reserved.

続き

```
                    strLine = "{} X:{} Y:{} Z:{} RZ:{} SortValue:{} ZValid:{}
                         XImgPos:{} YImgPos:{} Val1:{} Val2:{} Val3:{} Val4:{}
                         Val5:{} Level:{} Id:{}
                         ModelType:{}\n".format(str(key), str(items[key]['X']),
                         str(items[key]['Y']), str(items[key]['Z']),
                         str(items[key]['RZ']), str(items[key]['SortValue']),
                         str(items[key]['ZValid']), str(items[key]['XImgPos']),
                         str(items[key]['YImgPos']), str(items[key]['Val1']),
                         str(items[key]['Val2']), str(items[key]['Val3']),
                         str(items[key]['Val4']), str(items[key]['Val5']),
                         str(items[key]['Level']), items[key]['Id'],
                         str(items[key]['ModelType']))
                    f.write(strLine)
                f.close()
                return items;
```

RedistributeItemByTime.py

```
# PMTW user script demo -- RedistributeItemByTime
# Every minute a robot is exchanged to pick and place items.

import time

# Global definition
RTType = 1
item_1 = r''
item_2 = r''
workarea_2 = r''
workarea_4 = r''
# This path need to be created by users.
logPath = r'C:\PMScriptsLog\PlaceInfo.txt'

# PyInitialize interface
def PyInitialize(type, itemInfo):
  global RTType
  global item_1
  global item_2
  global workarea_2
  global workarea_4
  RTType = type
  f = open(logPath,'a')
  f.write("PyInitialize\n")
  # RT type
  strLine = "RTType:{}\n".format(str(RTType))
  f.write(strLine)
  # Item information
  keys = itemInfo.keys()
  for key in keys:
    strLine = "{} Name:{} Id:{}\n".format(str(key),
         str(itemInfo[key]['Name']), str(itemInfo[key]['Id']))
    if itemInfo[key]['Name'] == 'Item_1':
      item_1 = itemInfo[key]['Id']
```

次のページに続く

© Copyright 2025 ABB. All rights reserved.

```
    elif itemInfo[key]['Name'] == 'Item_2':
      item_2 = itemInfo[key]['Id']
    elif itemInfo[key]['Name'] == 'ConveyorWorkArea_2':
      workarea_2 = itemInfo[key]['Id']
    elif itemInfo[key]['Name'] == 'ConveyorWorkArea_4':
      workarea_4 = itemInfo[key]['Id']
    f.write(strLine)
  f.close()
# PyDistribution interface
def PyDistribution(WaId, items):
  global RTType
  global item_1
  global item_2
  global workarea_2
  global workarea_4
  f = open(logPath, 'a')
  f.write("PyDistribution\n")
  # Workarea Id
  strLine = "WaId:{}\n".format(WaId)
  f.write(strLine)
  # Item information
  keys = items.keys()
  for key in keys:
    if key == 'Time':
      # Time stamp(s), get the number of milliseconds since 1 Jan
          1970.
      strLine = "Time:{}\n".format(str(items[key]))
      f.write(strLine)
    else:
      # Modify before
      strLine = "{} X:{} Y:{} Z:{} q1:{} q2:{} q3:{} q4:{} Tag:{}
          Val1:{} Val2:{} Val3:{} Val4:{} Val5:{} Type:{} Index:{}
          State:{} Container:{} Layer:{} Group:{}
          Id:{}\n".format(str(key), str(items[key]['X']),
          str(items[key]['Y']), str(items[key]['Z']),
          str(items[key]['q1']), str(items[key]['q2']),
          str(items[key]['q3']), str(items[key]['q4']),
          str(items[key]['Tag']), str(items[key]['Val1']),
          str(items[key]['Val2']), str(items[key]['Val3']),
          str(items[key]['Val4']), str(items[key]['Val5']),
          str(items[key]['Type']), str(items[key]['Index']),
          str(items[key]['State']), str(items[key]['Container']),
          str(items[key]['Layer']), str(items[key]['Group']),
          items[key]['Id'])
      f.write(strLine)
      # Modify
      if (divmod(time.localtime().tm_min, 2))[1] == 0:
        if WaId == workarea_2:
          items[key]['State'] = 1
        elif WaId == workarea_4:
          items[key]['State'] = 0
      else:
```

次のページに続く

© Copyright 2025 ABB. All rights reserved.

続き

```
            if WaId == workarea_2:
              items[key]['State'] = 0
            elif WaId == workarea_4:
              items[key]['State'] = 1
          # Modify after
          strLine = "{} X:{} Y:{} Z:{} q1:{} q2:{} q3:{} q4:{} Tag:{}
               Val1:{} Val2:{} Val3:{} Val4:{} Val5:{} Type:{} Index:{}
               State:{} Container:{} Layer:{} Group:{}
               Id:{}\n".format(str(key), str(items[key]['X']),
               str(items[key]['Y']), str(items[key]['Z']),
               str(items[key]['q1']), str(items[key]['q2']),
               str(items[key]['q3']), str(items[key]['q4']),
               str(items[key]['Tag']), str(items[key]['Val1']),
               str(items[key]['Val2']), str(items[key]['Val3']),
               str(items[key]['Val4']), str(items[key]['Val5']),
               str(items[key]['Type']), str(items[key]['Index']),
               str(items[key]['State']), str(items[key]['Container']),
               str(items[key]['Layer']), str(items[key]['Group']),
               items[key]['Id'])
          f.write(strLine)
      f.close()
      return items;
```


© Copyright 2025 ABB. All rights reserved.

5.2 外部センサー

External sensor（外部センサー）

外部センサーは、アイテムポジションがどのように生成されるかをユーザーが完全にコントロールできるソフトウェアコンポーネントである。外部センサーは、バーコードリーダー、カメラ、フォトセンサーの組み合わせなど、あらゆるタイプのアイテム検出を使用してアイテムのポジションを生成可能。カメラを使用する場合、視覚ハードウェアや画像検索アルゴリズムは何でも使用可能。PMTWはPythonを使って外部センサープログラムの実装をサポートします。

注記

資格のある担当者のみがスクリプト ファイルを作成または変更してください。

スクリプト ファイルを使用して実行するときにセルが安全であることを確認するのは作成者の責任です。

注記

PickMaster® Twin 製品では、Python 3.12 のみがサポートされています。

PickMaster Twin 製品をインストールすると、Python 3.12 がコンピュータにインストールされ、Python 3.12 のパスがシステム環境変数に自動的に追加されます。

ヒント

構文エラーは、スクリプトファイルの実行に失敗する原因となります。

以下の方法で、シンタックスエラーを回避できます。

1 同じスクリプトファイルには、同じエディタを使用するようにしてください。
2 スクリプトファイルの編集には、PyCharmないしNotepad++の使用が、Python ファイルの構文チェック機能を有するので、推奨される。

注意

ローカルプレゼンスが正しい方法で設定されるように実装するのは、インテグレータの責任です。

単一の制御ポイントが正しい方法で設定されるように実装するのは、インテグレータの責任です。

危険

スクリプトを生産環境で使用する場合は、慎重に保護してください。

スクリプトにアクセスできる人は誰でも、スクリプトを直接変更できます。 これは重大な危険を引き起こす可能性があります。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

ヒント

コンベアで外部センサーが使用されている場合、フロー機能は無効になる。

注記

ひとつのレシピでUserScriptsと外部センサーを同時に使用することはできない。

注記

Pythonスクリプトファイルは Pack& Go ファイルには含まれない。Pythonスクリプトファイルを目的の保存先にコピー。

ヒント

インデックスワークエリアが使用されている場合、外部センサー機能は無効になる。

ツリービューで **外部センサー**を 右クリックし、**設定**を選択して外部センサーを設定。

以下の表は、 **外部センサー設定** ダイアログボックスの詳細を示す。

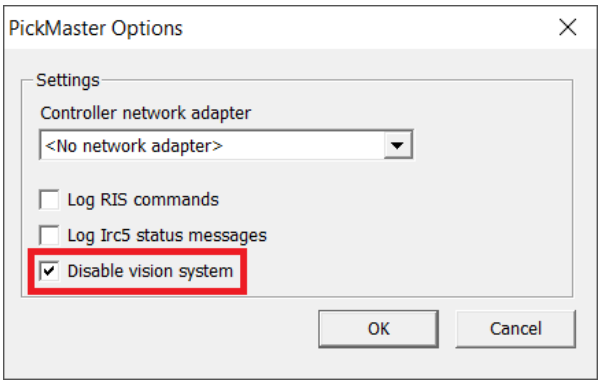
	説明
スクリプト名	事前定義済スクリプト ファイル名を.pyで入力します。テンプレートとして、名前ExternalSensors.pyの使用が推奨される。 **ヒント** スクリプト機能を使用する前に、定義済みのスクリプト ファイルを C:\Users\xxxx\Documents\PickMaster\PMScripts フォルダに配置する必要があります。
ユーザープログラムでの設定	本ボタンをクリックすると、Pythonインターフェースdef configureSensor(self, sensorId)を参照。ユーザーは、本インターフェースの設定作業を、自身のPythonクラスで自己定義する必要がある。ユーザーが本ボタンをクリックすると、インターフェースが呼び出される。
センサー情報の取得	本ボタンをクリックすると、 ファイルが提供するPythonインターフェースdef getSensorInfo(self)が、PMTWによってExternalSensorInterface.pyファイルで参照される。本インターフェイスでは、名前、作者、バージョン、説明などのユーザープログラム情報は、 **外部センサー設定** ダイアログに返送および表示される。ユーザープログラムの情報は、Pythonコードまたは configureSensorインターフェイスのファイル ExternalSensors.pyで直接変更できる。 **注記** ファイル ExternalSensorInterface.pyは変更できない。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

Basler カメラが外部センサー インターフェイスを通じて使用されている場合は、両方のシステムが同じカメラに接続しようとするのを避けるために、PickMaster'の内部ビジョン システムをオフにする必要があります。

RealRuntime内部ビジョンシステムをオフにします。



xx2300000904

外部センサーのメインファイルの新規作成

注記

ファイル`ExternalSensor.py`のすべてのコンテンツは、ファイル`ExternalSensorInterface.py`の定義済ベーシッククラスを継承する必要がある。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

定義済 外部センサーインターフェースクラス

PickMaster Twin外部センサー用に 4 種のクラスを`ExternalSensorInterface.py`ファイルで事前に定義。

これら 4 種のクラス以外に、ツールクラス`StoppableThread`が用意され、`threading.Thread`のクラスをネイティブのPythonから継承し、スレッドを停止する機能を有します。

クラス	説明

`SensorInfo`	本インターフェースは、Pythonプログラム初期化に使用される。たとえば、全基本情報および設定を初期化。
	• `def registerLogCallback(self, logCallBackFunc)`：ユーザーは`logCallBackFunc`を使用して、PMTWで次の形式で Python ログを表示できます。
	`log = {'LogLevel': 0,'Log': 'python log string'}`
	`self.fLogCallback.ShowPythonLog(log)`
	ログレベルには次の 4 種類があります。
	- 0-Status
	- 1-Warning
	- 2-Error
	- 3-Debug
	  ヒント
	ステータス、警告、エラー情報がRuntimeログ ビューに表示されます。
	デバッグ情報は、「デバッグ」ビューでのみ表示できます。

`SensorConfig`	本インターフェースは、センサー操作を実現するため、ユーザーへの提供に使用される。 • `def configureSensor(self, sensorId)`：内容は、ユーザーが自身のクラスで実装する必要がある。 • `def getSensorInfo(self)`：PMTWによって、ユーザープログラムの名前、作者、バージョン、説明の情報を取得。センサー情報取得ボタンがクリックされると、本インターフェースが呼び出される。 • `def loadSensor(self, sensorId, configurationInfo)`：PMTWによって、Python辞書データ`sensorConfigurationDict[sensorId]`に、以前に保存した設定情報文字列をロードし、ユーザープログラムの最後の設定を更新します。このインターフェースは、Python環境が初期化中で、センサーが以前に設定済みである場合に呼び出される。 • `def saveSensor(self, sensorId)`：PMTWによって、`configureSensor`で変更の可能性を有する最新の設定情報の文字列を取得します。センサー設定ダイアログでOKボタンがクリックされるか、保存ボタンがPositionGeneratorダイアログでクリックされると、このインターフェースが呼び出される。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

クラス	説明
PositionGenerator	このインターフェースは、カスタマイズされたセンサーポジションの生成で、ユーザーに提供するために使用される。 • def initializePosGenRelatedMap(self, sensorId, posGenId, objectName)：PMTWによって、センサー、ポジションジェネレーター、オブジェクト（アイテム/コンテナ）の関係を初期化します。ユーザーは、objectName (item/container name)をPythonのロジックで使用し、異なるポジションジェネレータを区別可能。本インターフェースは、Python環境が初期化中で、ひとつのポジションジェネレータが設定 - 有効状態から保存 - 有効状態に変更時に呼び出される。 • def configurePosGen(self, posGenId)：内容は、ユーザーが自身のクラスで実装する必要がある。 • def loadPosGen(self, posGenId, positionGeneratorInfo)：PMTWによって、以前に保存したポジションジェネレーター構成情報文字列をユーザープログラムのPython辞書データposGenConfigurationDict[posGenId]にロードして、最後設定を更新します。本インターフェースは、ひとつの既存のポジションジェネレーターが設定 - 有効状態から保存 - 有効状態に変更時に呼び出される。 • def savePosGen(self, posGenId)：PMTWによって、configurePosGen で変更された可能性を含む、最新のポジションジェネレーター設定情報文字列を取得。本インターフェイスは、「保存」ボタンがクリックされたとき、または「OK」ボタンがクリックされ、アイテム/コンテナソースタイプのビューでひとつのポジションジェネレータが保存可能状態になったときに呼び出される。
SensorRuntime	本インターフェースは、カスタマイズされたセンサー操作の実現で、ユーザーに提供するために使用される。 startSensorインターフェースの仕上げロジックを処理するために、フラグメカニズムが提供される。 • def startSensor(self, callBackFunc)：不必要なクラッシュを避けるために、フラグメカニズムに基づいて、ユーザーが独自のクラスでコンテンツを実装する必要がある。 • def stopSensor(self)：内容は、ユーザーが自身のクラスで実装する必要がある。 • def monitorRecipeStatus(self, callBackFunc)、def checkRecipeStatus(self, callBackFunc)、def waitForRecipeStop(self): Runtimeでレシピのステータスを監視するために PMTW によって提供されるフラグ メカニズム。ユーザーは、テンプレート フォルダ C:\Program Files (x86)\ABB\PickMaster Twin 3\Samples\ExternalSensorTemplates 内のテンプレートを参照してこれを使用できます。

スクリプトテンプレートExternalSensor.pyは、ユーザがクラスフォーマットに従って独自のクラスを開発する事例である。このクラスは、ExternalSensorInterface.pyで定義された4種のベースクラスを継承し、クラス名はメインファイル名と同じである必要があり、そうでなければ、PMTWがPythonファイルをロードしようとしたときにエラーが発生します。

def configureSensor(self, sensorId)インターフェース

本インターフェースでは、設定データの文字列への直列化と文字列のPython辞書sensorConfigurationDict[sensorId]への保存を含める必要がある。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

本インターフェースは、外部センサー設定ビューでユーザープログラムでの設定ボタンがクリックされると呼び出される。

システム信号の詳細については、*Pythonプログラムで使用されるデータ構造 ページ 367*を参照してください。

引数	説明	注記
self	Python構文 クラス参照	
sensorId	にPMPPよって自動生成されるセンサー ID	

例：

```
def configureSensor(self, sensorId):# this interface must be
     implemented by users
  # Step 1 & Step 2:
  # analyze self.sensorConfigurationDict[sensorId] to get the
       settings from last configuration.
  # user-defined configuration logic.
  self.name = "ExternalSensorsDemo02"
  if self.sensorIdNameMapDict[sensorId] == 'ExternalSensor_1':
    inputTitle = "ExternalSensor_1 configuration"
  elif self.sensorIdNameMapDict[sensorId] == 'ExternalSensor_2':
    inputTitle = "ExternalSensor_2 configuration"
  if sensorId in self.sensorConfigurationDict.keys():
    configurationInfo =
         ExternalSensorsDemo.showSensorConfigDialog(self,
         inputTitle, self.sensorConfigurationDict[sensorId])
  else:
    configurationInfo =
         ExternalSensorsDemo.showSensorConfigDialog(self,
         inputTitle, '0')
  log = {'LogLevel': 0,'Log': configurationInfo}
  self.fLogCallback.ShowPythonLog(log)
  # Step 3:
  # parse the configuration data into one string and update it in
       self.sensorConfigurationDict[sensorId].
  self.sensorConfigurationDict[sensorId] = configurationInfo
```

def configurePosGen(self, posGenId)インターフェース

本インターフェースでは、ポジションジェネレータ設定データの文字列への直列化と文字列のPython辞書posGenConfigurationDict[posGenId]への保存を含める必要がある。

本インターフェイスは、アイテム/コンテナソースタイプのビューで設定ボタン、または新規ポジションジェネレータボタンがクリックされると呼び出される。

システム信号の詳細については、*Pythonプログラムで使用されるデータ構造 ページ 367*を参照してください。

引数	説明	注記
self	Python構文 クラス参照	

次のページに続く

© Copyright 2025 ABB. All rights reserved.

引数	説明	注記
posGenId	PMPPによって自動生成されるポジションジェネレータ ID	

例：

```
def configurePosGen(self, posGenId):# this interface must be
     implemented by users.
  # Step 1 & Step 2:
  # analyze self.posGenConfigurationDict[posGenId] to get the
       settings from last configuration.
  # user-defined configuration logic.
  if self.sensorIdNameMapDict[self.posGenSensorMapDict[posGenId]]
       == 'ExternalSensor_1':
    inputTitle = "ExternalSensor_1 PosGen configuration"
  elif self.sensorIdNameMapDict[self.posGenSensorMapDict[posGenId]]
       == 'ExternalSensor_2':
    inputTitle = "ExternalSensor_2 PosGen configuration"
  if posGenId in self.posGenConfigurationDict.keys():
    positionGeneratorInfo =
         ExternalSensorsDemo.showPosGenConfigDialog(self,
         inputTitle, self.posGenConfigurationDict[posGenId])
  else:
    positionGeneratorInfo =
         ExternalSensorsDemo.showPosGenConfigDialog(self,
         inputTitle, "0")
  log = {'LogLevel': 0,'Log': positionGeneratorInfo}
  self.fLogCallback.ShowPythonLog(log)
  # Step 3:
  # parse the configuration data into one string and update it in
       self.posGenConfigurationDict[posGenId].
  self.posGenConfigurationDict[posGenId] = positionGeneratorInfo
```

def startSensor(self, callBackFunc)インターフェース

本インターフェースでは、ユーザーはセンサーを作動させ、ポジションを生成するロジックを処理する必要がある。本インターフェイスは、実行レシピの開始時に呼び出される。

PMTW`callBackFunc`はインターフェース入力引数として与えられ、ふたつの具象コールバック関数、すなわち`GetStrobeTime()`および`NewPosition(pos)`を含む。

`GetStrobeTime()`は、センサーがトリガー信号を取得したときに呼び出され、ストローブ時間を取得します。ポジション生成時に、`NewPosition(pos)`が通信され、新しく生成されたポジションがストローブ時間とともにPMTWに送信される。

引数	説明	注記
self	Python構文 クラス参照	

次のページに続く

© Copyright 2025 ABB. All rights reserved.

引数	説明	注記
callBackFunc	GetStrobeTime()および NewPosition(pos)を有する	ポジションフォーマットは本構成に従う： newPos = {‘SensorId’: sensorId, # sensor id ‘Time’: strobeTime, # time stamp, get from PMTW by calling GetStrobeTime() callback function, unit is ms. ‘key’: {'X': 0.0, # key refers to the position index, start from 0. X refers to the location value of the item in X direction, unit is mm. 'Y': 100.0, # Y refers to the location value of the item in Y direction, unit is mm 'Z': 5.0, # Z refers to the location value of the item in Z direction, unit is mm 'RX': 0.0, # RX refers to the rotation angle value of the item in X direction, unit is degree 'RY': 0.0, # RY refers to the rotation angle value of the item in Y direction, unit is degree 'RZ': 0.0, # RZ refers to the rotation angle value of the item in Z direction, unit is degree 'Tag': 0, # used in rapid 'Score': 1.0, # refers to the score of different position generator methods, could be used to sort the results 'Val1': 0.0, # optional value, used in rapid 'Val2': 0.0, # optional value, used in rapid 'Val3': 0.0, # optional value, used in rapid 'Val4': 0.0, # optional value, used in rapid 'Val5': 0.0, # optional value, used in rapid 'Level': 2, # level, 0: Discarded, 1: Rejected, 2: Accepted 'PosGenId': posGenId}, # external sensor position generator id

次のページに続く

© Copyright 2025 ABB. All rights reserved.

引数	説明	注記
		次の事例は、ふたつのポジションを含むフォーマットである： newPos = {‘SensorId’: ‘11548258-b028-470a-b399-b780084acc59’, ‘Time’: 376910718, ‘0’: {'X': 0.0, 'Y': 100.0, 'Z': 5.0, 'RX': 0.0, 'RY': 0.0, 'RZ': 0.0, 'Tag': 0, 'Score': 1.0, 'Val1': 0.0, 'Val2': 0.0, 'Val3': 0.0, 'Val4': 0.0, 'Val5': 0.0, 'Level': 2, 'PosGenId': '17dec9b3-8624-45fa-b8cb-366b457e6024'}, ‘1’: {'X': 0.0, 'Y': 100.0, 'Z': 5.0, 'RX': 0.0, 'RY': 0.0, 'RZ': 0.0, 'Tag': 0, 'Score': 1.0, 'Val1': 0.0, 'Val2': 0.0, 'Val3': 0.0, 'Val4': 0.0, 'Val5': 0.0, 'Level': 2, 'PosGenId': '5413832d-dbbe-44b9-aab3-6bb64d8d30a1'}}

例：

```
def startSensor(self, callBackFunc):# this interface must be
     implemented by users.
  try:
    # Step 1: call classname.monitorRecipeStatus(self, callBackFunc)
         to monitor the recipe status running in PMTW.
    ExternalSensorsDemo.monitorRecipeStatus(self, callBackFunc)

    # Step 2: start logic defined by users. For each sensor, a
         StoppableThread must be created to generate positions and
         appended to self.allThreads.
    for posGenId in self.posGenSensorMapDict:
      sensorId = self.posGenSensorMapDict[posGenId]
```

次のページに続く

© Copyright 2025 ABB. All rights reserved.

```
            windowTitle = self.sensorIdNameMapDict[sensorId] + ' ' +
                 self.posGenObjectMapDict[posGenId]
            if self.sensorIdNameMapDict[sensorId] == 'ExternalSensor_1':
              thread_1 =
                   StoppableThread(target=ExternalSensorsDemo.showStartDialog,
                   args=(self, callBackFunc, sensorId, posGenId,
                   self.posGenConfigurationDict[posGenId], windowTitle))
              self.allThreads.append(thread_1)
            elif self.sensorIdNameMapDict[sensorId] == 'ExternalSensor_2':
              thread_2 =
                   StoppableThread(target=ExternalSensorsDemo.showStartDialog,
                   args=(self, callBackFunc, sensorId, posGenId,
                   self.posGenConfigurationDict[posGenId], windowTitle))
              self.allThreads.append(thread_2)
          # Step 3: start all threads in self.allThreads.
          for td in self.allThreads:
            td.start()

          # Step 4: call classname.waitForRecipeStop(self) to wait for
               the stop signal from PMTW.
          ExternalSensorsDemo.waitForRecipeStop(self)
          # Step 5: stop all threads in self.allThreads. Note that the
               stop behavior of startSensor interface should be handled
               by users here.
          for td in self.allThreads:
            td.stop()
          log = {'LogLevel': 0, 'Log': "StartSensor: stopped all
               threads."}
          self.fLogCallback.ShowPythonLog(log)
        except:
          log = {'LogLevel': 2,'Log': "Python Error: Failed to start
               sensor"}
          self.fLogCallback.ShowPythonLog(log)
```

startSensorをフラッグメカニズムで使用

ユーザーは、startSensorのコンテンツをPMTWが提供するフラグメカニズムに基づいて実装する必要がある。以下のコードはその事例である。

- インターフェースの初期に、PMTWによって提供されるmonitorRecipeStatusメソッドが最初に呼び出され、レシピが実行されているかを監視するスレッドを開始する必要がある。
- そして、ポジション生成のロジックについては、ユーザーが独自のロジックを実装できる。本事例では、ポジション生成シミュレーターを起動するスレッドが作成される。インターフェースのコンテンツが終了することは保証されているので、スレッドは停止できる。そのため、本事例ではPMTW開発者が提供する停止対応スレッドを使用している。
- 複数のポジションジェネレーターが存在する場合もある。ひとつのスレッドでひとつのポジションジェネレーターが実行される場合、複数のスレッドが存在することになる。したがって、次のステップはすべてのスレッドを起動し、すべてのスレッドの内容を実行することである。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

- 次のステップは、レシピ停止を示すフラグ信号の待機である。レシピが実行中の場合、フラグは1になる。レシピが停止すると、フラグは0になる。`waitForRecipeStop`のメソッドはフラグの値を取得するもので、すべてのスレッドが呼び出された後に呼び出される必要がある。
- フラグ信号がレシピの停止を示す場合、本インターフェースのすべてのスレッドが停止します。スレッドの直接停止前に、スレッド内の接続ポートを閉じるなど、必要に応じて、まずスレッド内のコンテンツを停止することに注意すること。

def stopSensor(self)インターフェース

このインターフェースでは、ユーザーが何らかのロジックを実装する必要がある場合、このインターフェースにコンテンツを追加できる。

このインターフェースは実行中のレシピが停止される時に呼び出される。

引数	説明	注記
self	Python構文 クラス参照	

例：

```
def stopSensor(self):# this interface must be implemented by users.
  log = {'LogLevel': 0,'Log': "Python info: stop sensor"}
  self.fLogCallback.ShowPythonLog(log)
```

Pythonプログラムで使用されるデータ構造

`ExternalSensorInterface.py`ファイルでは、センサー、ポジションジェネレーター、オブジェクト（アイテム/コンテナ）間の関係を保存するために、いくつかのデータ構造が定義され、ユーザーは、Pythonロジックを実現するために、独自のクラスでそれらを使用できる。

	説明
sensorIdNameMapDict	センサー ID と名前の関係を保存します。`SensorId`は、PMPPで自動生成され、ユーザーは変更できない。`SensorName`は外部センサーノードの名前で、例えば**ExternalSensor_1**を指す。ユーザーは、PMPPが提供するデフォルト名を使用するか、要件に応じて名前を変更できる。ほとんどのPythonインターフェースでは、`SensorId`が入力引数として与えられるので、ユーザーは直接、`SensorId`を使って辞書を検索し、対応するセンサー名を見つけることができる。ユーザーがロジックでセンサー名を使用し、PMPPでの名前を変更した場合、Pythonの文字列を更新し、PMPPと同じにする必要がある。 ほとんどの場合、ユーザーはこの辞書でキーと値を定義したり割り当てたりする必要はなく、`self.sensorIdNameMapDict[sensorId]`を使用して`SensorId`に基づいたセンサー名の取得に限られる。 事例：`sensorIdNameMapDict = {'11548258-b028-470a-b399-b780084acc59': 'ExternalSensor_1'}`

次のページに続く

© Copyright 2025 ABB. All rights reserved.

	説明
sensorConfigurationDict	センサー ID と設定情報の文字列の関係を保存します。これまでのセクションによると、設定情報の文字列はセンサー設定を反映し、データは文字列に直列化され、PMPPソリューションに保存される。 loadSensorが呼び出されると保存された文字列はこの辞書に更新され、本辞書は ExternalSensorInterface.pyで実装されている。 configureSensorの呼び出し時、センサーがすでに設定されている場合、ユーザーは最初に設定文字列を解析するか、直列化を解除する必要がある。SensorId（インターフェースの入力引数）を使って、まず文字列を見つけることができる：self.sensorConfigurationDict[sensorId] configureSensorの最後に、ユーザーは本インターフェイスの構成設定を変更する可能性があるため、最新の設定を反映した新しい文字列を生成し、辞書に保存する必要がある：self.sensorConfigurationDict[sensorId] = configInfoString saveSensorの呼び出し時に、設定の文字列は最新の設定を含む辞書で見つかり、PMPPに返され、これは ExternalSensorInterface.pyに実装される 。 事例：sensorConfigurationDict= {'11548258-b028-470a-b399-b780084acc59': ' SensorType:Camera;IP:192.169.10.10;Brightness:10'}
posGenConfigurationDict	ポジションジェネレータの ID および構成情報文字列の関係を保存します。センサー設定と同様に、ポジションジェネレーター設定情報の文字列は、ポジションジェネレーターの設定を反映し、データは文字列で直列化され、PMPPソリューションに保存される。 loadPosGenが呼び出されると、保存された文字列はこの辞書に更新され、本辞書は ExternalSensorInterface.pyで実装されている。 configurePosGenの呼び出し時に、ポジションジェネレーターがすでに設定されている場合、ユーザーは最初に設定文字列を解析するか、直列化解除の必要がある場合がある。posGenId（インターフェースの入力引数）を使って、まず文字列を見つけることができる：self.posGenConfigurationDict[posGenId] configurePosGenの最後に、ユーザーは本インターフェイスの構成設定を変更する可能性があるため、最新の設定を反映した新しい文字列を生成し、辞書に保存する必要がある：self.posGenConfigurationDict[posGenId] = posGenConfigInfoString savePosGenの呼び出し時に、設定の文字列は最新の設定を含む辞書で見つかり、PMPPに返され、これは ExternalSensorInterface.pyに実装される 。 事例：posGenConfigurationDict= {'17dec9b3-8624-45fa-b8cb-366b457e6024': 'PositionGeneratorType:Blob;Type:White;ThresholdValue:100'}

次のページに続く

© Copyright 2025 ABB. All rights reserved.

	説明
posGenSensorMapDict	ポジションジェネレーター ID およびセンサー ID の関係を保存。以前紹介したように、外部センサーポジションジェネレータは、異なるアイテム/コンテナ下の各センサーに対して作成できる。PMPPソリューションに複数の外部センサーがある場合、同一アイテム/コンテナの下で、**externalsensor_1**はそのポジションジェネレーターを保有可能で、**externalsensor_2**も独自のポジションジェネレーターを保有できる。これらふたつのポジションジェネレータは異なる外部センサーに属しているため、データの混乱を避けるためにその関係を明確にする必要がある。 それらの関係は、`initializePosGenRelatedMap`に実装されているインターフェース`ExternalSensorInterface.py`で初期化され、ユーザーは必要なときに利用できる：`self.posGenSensorMapDict[posGenId]` 事例：`posGenSensorMapDict= {'17dec9b3-8624-45fa-b8cb-366b457e6024': '11548258-b028-470a-b399-b780084acc59'}`
posGenObjectMapDict	ポジションジェネレーター ID およびオブジェクト名の関係を保存。ここでオブジェクトとは、アイテムやコンテナを指す。先に紹介したように、外部センサーポジションジェネレータは、異なるアイテム/コンテナ下にある各センサー用に作成できる。PMPPソリューションに複数のアイテムがある場合、同一外部センサーに対して、**item_1**はそのポジションジェネレーターを保有可能で、**item_2**も独自のポジションジェネレーターを保有できる。このふたつのポジションジェネレータは異なるアイテムに属しているため、データの混乱を避けるために、それらの関係を明確にする必要がある。 ユーザープログラムのロジックではオブジェクト名が必要になることがあるため、本マップ辞書ではアイテム ID の代わりにオブジェクト名が保存される。 それらの関係は、`ExternalSensorInterface.py`に実装されているインターフェース`initializePosGenRelatedMap`で初期化され、ユーザーは必要なときに利用できる：`self.posGenObjectMapDict [posGenId]` 事例：`posGenObjectMapDict= {'17dec9b3-8624-45fa-b8cb-366b457e6024': 'Item_1'}`

すべてのスクリプト サンプル ファイルは、PickMaster クライアントがインストールされている場合、フォルダ `C:\Program Files (x86)\ABB\PickMaster Twin 3\Samples\ExternalSensorTemplates` に用意されています。

ユーザーはまた、名前、説明、作者、バージョンの情報プロパティを上書きできる。これらのインターフェースとプロパティを除けば、ユーザーはこのクラスや他のPythonファイルにさらに多くの関数を実装できる。PMTWソフトウエアはこの4種のインターフェースを認識して呼び出すだけで、他のインターフェースは無視します。

すべての使用Pythonファイルは、`C:\Users\..\Documents\PickMaster\PMScripts`のHost PC で以下の同一パスにある。

外部センサーの構成

外部センサーがツリービューで作成された場合、物理的センサーには接続されない。これは、外部センサー設定ダイアログボックスで手動で行う必要がある。ツリービューの外部センサーは、特定の物理的外部センサーを使用するように設定されている。外部センサーも最適な画像が得られるように設定する必要がある。

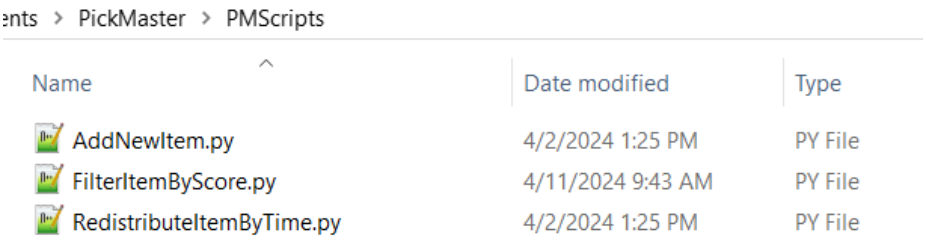
次のページに続く

© Copyright 2025 ABB. All rights reserved.

外部センサーの設定。

1 事前定義されたスクリプト ファイルを宛先フォルダに配置します。

> **ヒント**
>
> スクリプト機能を使用する前に、定義済みのスクリプト ファイルを `C:\Users\xxxx\Documents\PickMaster\PMScripts` フォルダに配置する必要があります。



xx2200001784

2 ツリー ビューのレイアウトでカメラを右クリックし、設定 を選択。
外部センサー構成ダイアログを開く。
3 スクリプト名に定義済のメインファイル名を入力。
4 外部センサーを設定するには、ユーザープログラムの設定をクリックします。
5 OKをクリックして、PMPPに設定を適用。
6 ツリー ビューのプロセスで必要なレシピを右クリックし、設定を選択。
7 アイテムソース/コンテナソースのページに 切り替える。
8 新規ポジションジェネレータをクリックし、外部タブで希望する外部センサー用のポジションジェネレーターを作成します。

> **注記**
>
> ポジションジェネレータを作成できるのは、外部センサー設定で 設定された外部センサーに限られる。

> **ヒント**
>
> 本ソリューションで作成されたすべての外部センサーは、外部タブに一覧表示される。

9 外部センサースクリプトファイルで定義されたユーザーに従ってポジションジェネレーターを設定。
10 保存をクリックして、ポジションジェネレータの設定データをPMPP に保存します。
11 [OK] をクリックします。

© Copyright 2025 ABB. All rights reserved.

5.3 さまざまな高さの製品の扱い（2.5D ビジョン）

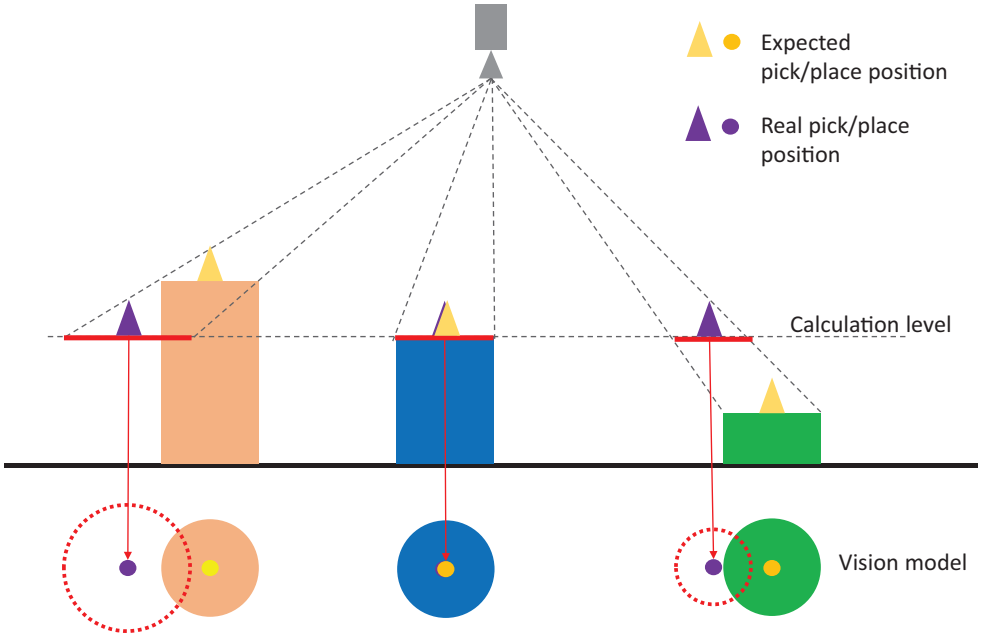
高さの設定について

通常、PickMaster PowerPacのビジョン ツールは、特定の高さで行われた計算に基づいて、2D 座標系の結果 (X 角度と Y 角度) を返します。 トレーニングされたモデルは、カメラ キャリブレーションの平面に配置されていると想定されます。

キャリブレーションされた平面の上または下にあるオブジェクトを操作すると、視差ポジションの問題が発生します。

定義された高さに基づいて計算すると、異なる高さのオブジェクトは、結果として生じる視差によってシフトされます。

カメラは、青いブロックの上面に基づいて計算するように教えられます。平行がない場合、カメラはこのサーフェスを下の緑色のブロックまたは上のオレンジ色のブロックに使用します。その結果、緑色のブロックとオレンジ色のブロックの x 座標とy座標が、誤用された計算平面に基づいて移動することになります。それはロボットを誤って間違ったポジションをターゲットにする可能性があります。



xx2300001744

2.5D キャリブレーションでは、完全な 3 次元空間がキャリブレーションされ、システムがオブジェクトの正確な高さを認識している場合、システムは視差エラーを補正できます。これにより、次のような疑問が生じます。

1. オブジェクトはどの高さにあるのか（z 座標）？
2. 真の x および y 座標はどこにあるのか？オブジェクト認識ツールは、オブジェクトがキャリブレーションの平面上にあるものと仮定しており、従ってこの平面から照射される座標を提供します。

真の X 座標と Y 座標を計算するには、マルチビュー キャリブレーションを実行することによって提供されるカメラのポジションに基づいて、キャリブレーション面上のカメラの高さと、キャリブレーション面までの製品の距離 (上/下) を知る必要があります。カメラのキャリブレーション ページ 282を参照してください。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

オブジェクトが配置されている高さの決定は、PickMaster PowerPacを使用して 3 つの方法で行うことができます。

1 手動入力
2 トレーニング済みオブジェクトに対するスケール変化に基づく自動計算。
3 外部入力

3 つの方法はすべて視差補正された x 座標と y 座標を返し、方法 2 と 3 は推定された Z 座標も返します。

実際には、本セクションで説明するツールで、パララックスエラーの補償（ x およびy座標の検出）および製品の高さの測定ができます。

事前知識

カメラは複数のイメージでキャリブレートしなければなりません（マルチビュー）。

高さ設定は、幾何学的なスタンドアロンモデルまたはメインの幾何学ベースの検査モデルと一緒に使用できます。

> **注記**
>
> メインの幾何学ベースの検査モデルは、ビジョン高さまたは外部高さと互換性がありません。

高さ設定の構成

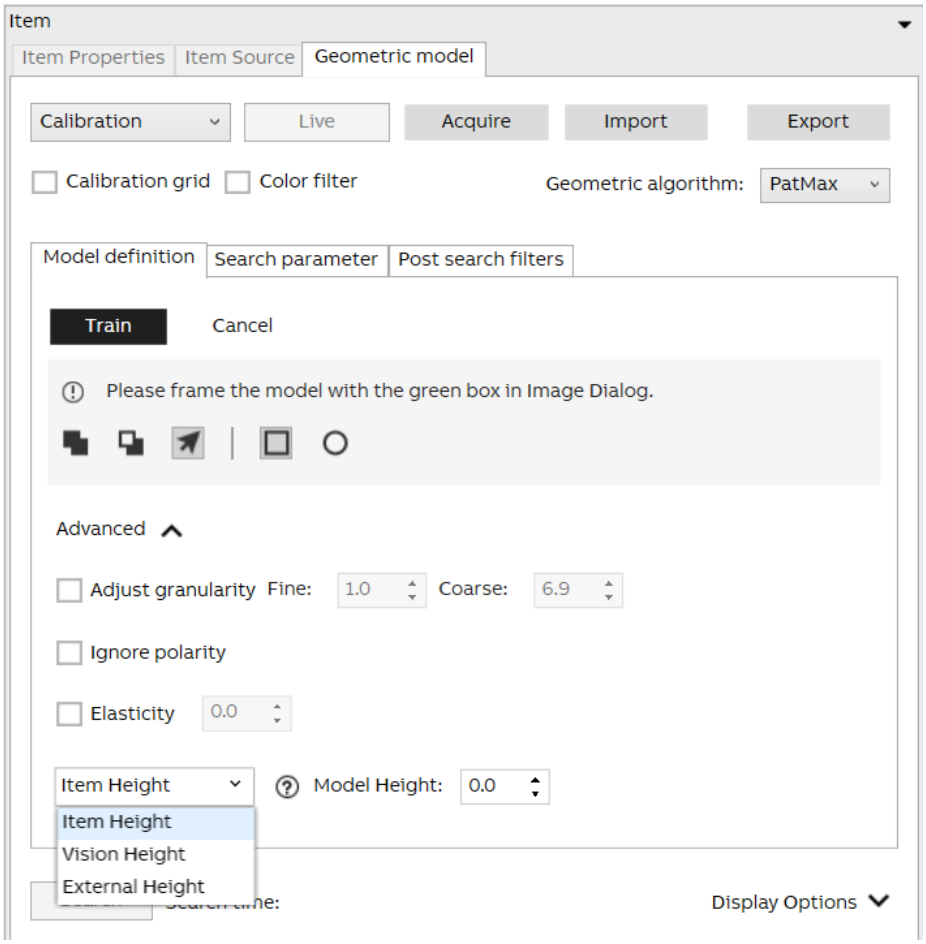
高さ設定は特定のモデルに属しており、幾何学的と一緒にのみ設定できます。

下記の手順に従って高さの設定を構成してください.

1 幾何学的モデルを作成します。
2 ［Image（イメージ）］欄で［Calibration（キャリブレーション）］リストからキャリブレーションを選択します。これはマルチビューキャリブレーションでなければなりません。
3 モデル定義部分で、詳細をクリックします。これにより、幾何学的詳細モデル設定ダイアログが開きます。
4 アイテムをトレーニングする前に、適切な計算方法を選択してください。

次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2300001532

5 計算方法を**アイテムの高さ**に設定した場合、
アイテムの高さ: ピッキング/プレース高さの値を手動で入力します。

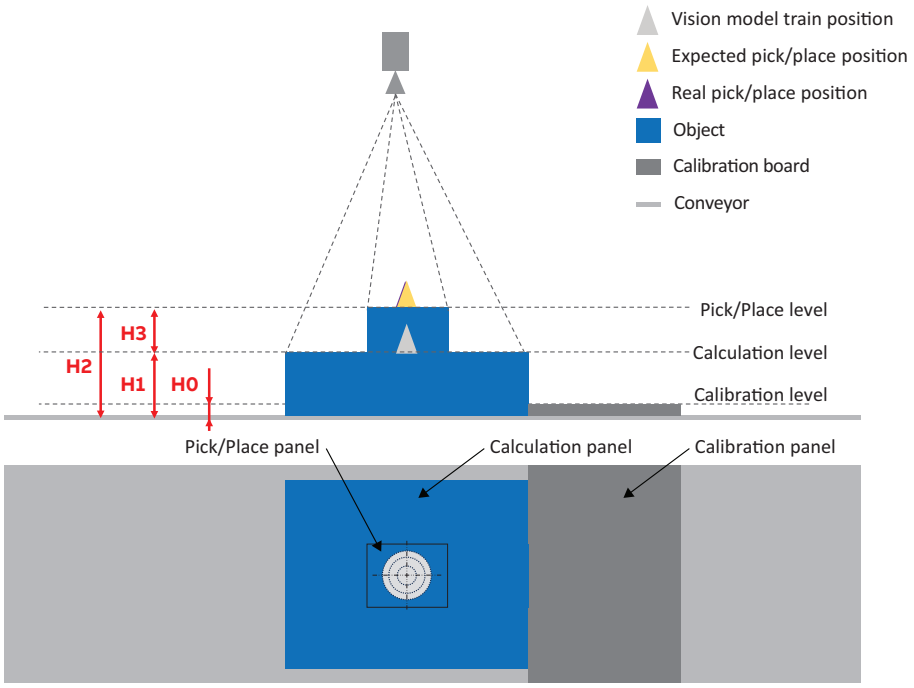


xx2300001533

この計算方法では 1 つのパラメータを満たす必要があります。**モデルの高さ**は文字通り、キャリブレーション パネルから計算平面までの高さを表すために使用されます。Z 座標は、オブジェクトの実際のピッキング/プレースの高さとして定義され、ロボット コントローラに送信されます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2300001745

a キャリブレーション ボードの高さ（H0）を測定します。

> **ヒント**
>
> キャリブレーション ボードの厚さが薄すぎて測定できない場合 (たとえば、普通紙など)、ユーザーはキャリブレーション ボードの高さを無視でき、H0は0です。

b コンベアからピッキング/プレース・オブジェクトの最大コンター パネルまでの高さ(H1)を測定します。

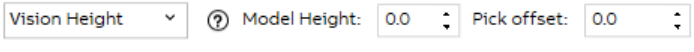
c ピッキング・プレース・オブジェクトのコンベアからピッキング・プレース パネルまでの高さ(H2)を測定します。

d **モデルの高さ**に値 (H1-H0) を入力します。

e **寸法(x,y,z)[mm]/RH 寸法[mm]**を設定するアイテムのZ座標に値(H2-H0)を入力します。*Item（項目） ページ 142*を参照してください。

6 計算方法を**ビジョンの高さ**に設定した場合、

ビジョンの高さ：キャリブレーション パネルから計算平面までの値は、見つかったオブジェクトのスケール変化 (トレーニング済みパターンに対する) から計算されます。



xx2300001534

この計算方法では 2 つのパラメータを満たす必要があります。**モデルの高さ**は、**アイテムの高さ**で定義されたのと同じ意味に従います。**ピック オフセッ**

次のページに続く

© Copyright 2025 ABB. All rights reserved.

トは、計算平面からピック/プレース パネルまでの偏差です。 計算平面は、識別されたオブジェクトの最大の等高線パネルとして定義されます。

> **注記**
>
> [均一スケールを有効にする] を有効にする必要があります。最大値と最小値には、十分なスケール変動を考慮する必要があります。

a キャリブレーション ボードの高さ（H0）を測定します。

b コンベアからピッキング/プレース・オブジェクトの最大コンター パネルまでの高さ(H1)を測定します。

c ピッキング・プレース・オブジェクトの最大コンター パネルからピッキング・プレース パネルまでの高さ(H3)を測定します。

> **ヒント**
>
> ピッキング・プレース パネルが計算パネルよりもZ方向に高い場合、H3 は正の数になります。
>
> 計算パネルがピッキング・プレース パネルよりもZ方向に高い場合、H3 は負の数になります。

d **モデルの高さ**に値 (H1-H0) を入力します。

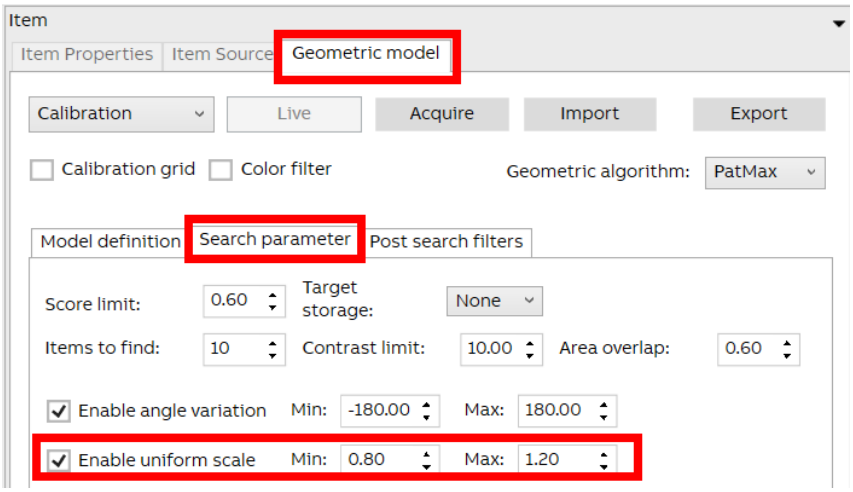
> **ヒント**
>
> この値が正の数である場合は、計算パネルが Z 方向でキャリブレーション パネルよりも高いことを意味します。
>
> この値が負の数の場合、キャリブレーション パネルが計算パネルよりも Z 方向に高いことを意味します。

e **ピック オフセット**に値 (H3) を入力します。

f **均一スケールを有効にする** を有効にして、スケーリングの適切な範囲を入力します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2300001537

7 計算方法を**外部の高さ**に設定した場合、

外部高さ：計算平面までの製品の距離 (上/下) は外部ソースによって計算されます。 これは、高さセンサー、またはセル PLC またはその他の外部デバイスからの情報である可能性があります。Z 座標は、UDP ポートを介して外部ソースからPickMaster Runtimeに送信されます。外部ソースの UDP リスニング ポートは、ビジョン モデルごとに一意である必要があります。現在のビジョン モデルのポジションメッセージのみが、このビジョン モデルのリスニング ポートを介して送信できます。



xx2300001535

この計算方法では 1 つのパラメータを満たす必要があります。

a **UDP リスニング ポート**に UDP ポートを入力します。次に、計算された Z 座標が、このポートを介して UDP メッセージとともにPickMaster Runtimeに送信されます。

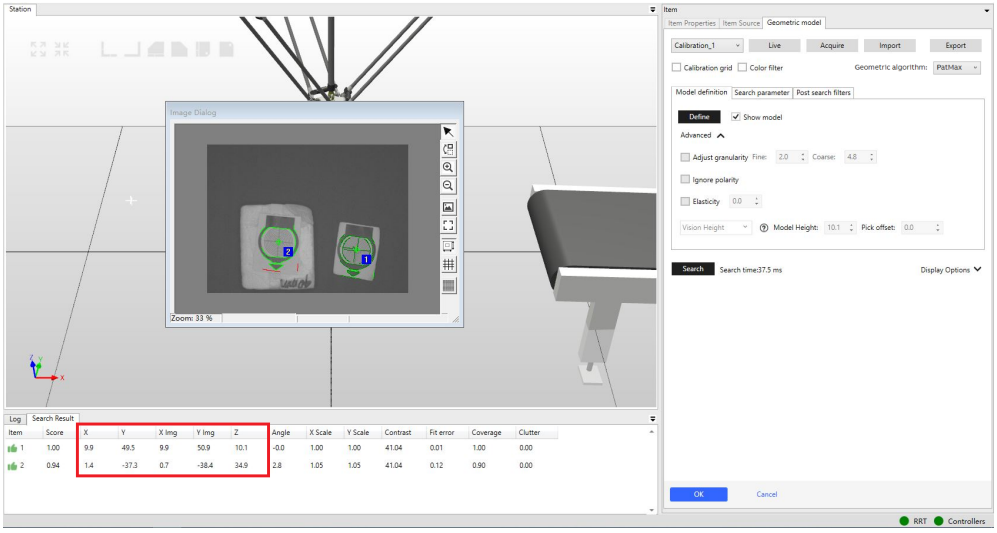
> **ヒント**
>
> この**UDP リスニング ポート**は、外部ソースの事前定義されたポートと一致する必要があります。

> **注記**
>
> 異なるビジョン モデルは、異なるポートを使用するように設定する必要があります。

モデルのトレーニング中に高さを設定すると、検索結果には、検索されたすべてのオブジェクトの空間情報が含まれます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2300001536

注記

ビジョン高さ法の精度は低い場合があります。精度は、カメラ、カメラのキャリブレーション、カメラの解像度、イメージに対するモデルの相対サイズなどの多くの要素によって異なります。従って、得られる精度は個々の適用に対して確認しなければなりません。

注記

高さ法としての［Model height（モデルの高さ）］の値の定義および［Item height（アイテムの高さ）］の選択は パララックスの補償となりますが、 z 座標はビジョンシステムによって計算されることはありません。

注記

オブジェクトタイプが1つしかなく、それが常に同一の高さにポジションしている場合には、**モデルの高さ**で補償する代わりに、その高さでカメラをキャリブレートした方が精度が高くなります。

ヒント

ビジョンの高さ方法を使用するときに誤った高さ情報を除外するには、幾何モデル ダイアログの **ポスト検索フィルタ** 部分で適切なスケール制限を設定します。

外部高さプロトコル

SetHeight

外部機器からモデルの高さの値を設定するために PickMaster Runtime に送信されるリクエストメッセージ。

メッセージ長: 5 バイト。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

リクエスト例: 高さ 110.11 [0x01, 0x42, 0xDC, 0x38, 0x52] の送信

ポジション	パラメータ	タイプ	変数属性
バイト 0	1	整数型	高さを設定するコマンド。 この例では、値 0x01 は外部ソースが高さの値を設定することを意味します。
バイト 1 ~ 4	y,y,y,y	浮動小数点	アイテムの高さ (mm)、ビッグ エンディアン。 この例では、バイト値 0x42, 0xDC, 0x38, 0x52 は、110.11として高さの値を示します。

PickMaster Runtime から受信した応答メッセージ。

メッセージ長: 5 バイト。

応答例: Runtimeは送信された高さ 110.11 [0x01, 0x42, 0xDC, 0x38, 0x52] を受信しました

ポジション	パラメータ	タイプ	変数属性
バイト 0	1	整数型	高さを設定するコマンド。 この例では、値 0x01 は外部ソースが高さの値を設定することを意味します。
バイト 1 ~ 4	y,y,y,y	浮動小数点	アイテムの高さ (mm)、ビッグ エンディアン。 この例では、バイト値 0x42, 0xDC, 0x38, 0x52 は、110.11として高さの値を示します。

ヒント

float型からバイトストリームに変換する場合は、大きい順にソートしてください

GetHeight

外部機器から現在の高さの値セットを取得するために PickMaster Runtime に送信されるリクエストメッセージ。

メッセージ長: 1 バイト。

リクエストの例: 高さ [0x02] が必要です

ポジション	パラメータ	タイプ	変数属性
バイト 0	2	整数型	高さを取得するためのコマンド。 この例では、値 0x02 は外部ソースが高さの値を取得することを意味します。

PickMaster Runtime から送信される応答メッセージ。

メッセージ長: 5 バイト。

応答の例: Runtime が高さ 110.11 [0x02, 0x42, 0xDC, 0x38, 0x52] を送信しました

ポジション	パラメータ	タイプ	変数属性
バイト 0	2	整数型	高さを取得するためのコマンド。 この例では、値 0x02 は外部ソースが高さの値を取得することを意味します。
バイト 1 ~ 4	y,y,y,y	浮動小数点	アイテムの高さ (mm)、ビッグ エンディアン。 この例では、バイト値 0x42, 0xDC, 0x38, 0x52 は、110.11として高さの値を示します。

© Copyright 2025 ABB. All rights reserved.

5.4 フローを使用した生産 (ゴースト ピッキング)

概要

Ghost picking flow(ゴースト ピッキング フロー)は、アプリケーション エンジニアが生産前にドライ サイクル機能を実行するために使用されます。 ユーザーは、ロボットがリアルワークステーションで空の物体を拾っているのを見ます。 この機能は、入力される素材が仮想であり、前のレコードによって生成されたフローによって提供されるという点で、生産とは異なります。

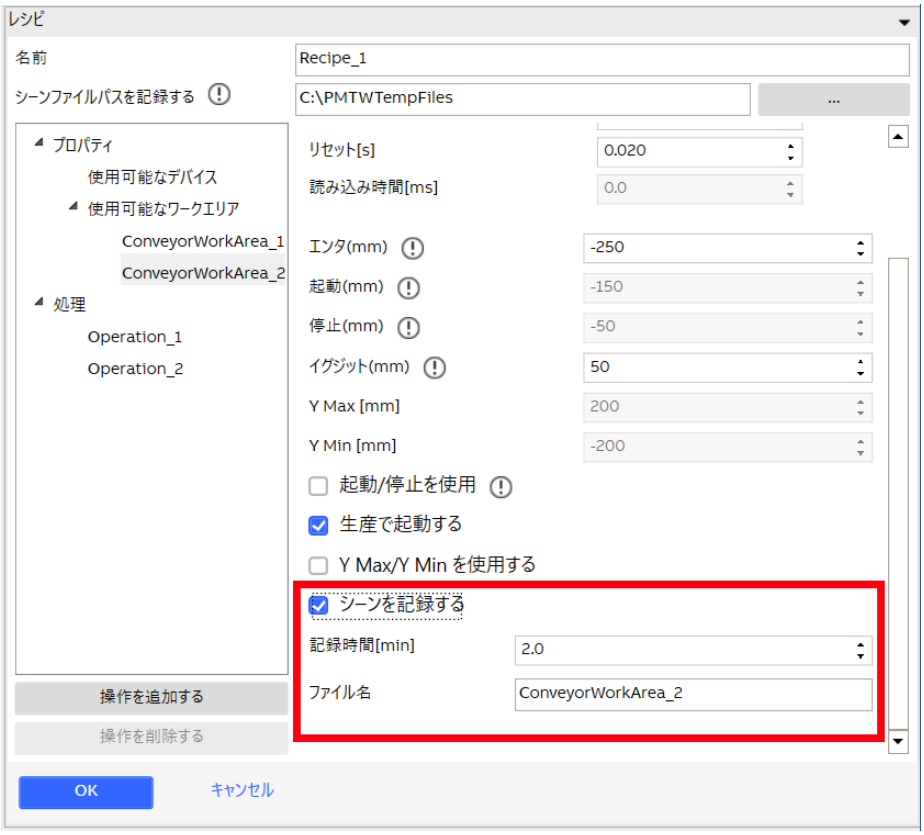
注記

ファイルの記録に推奨される最大時間は10分です。時間の値が長すぎると、記録されたファイルのサイズが大きくなりすぎて開けなくなります。

ゴースト ピッキング フローの作成

次の手順でゴースト ピッキング フローを作成します。

1. ソリューションを開きます。
2. ツリー ビュー **プロセス**で必要なレシピを右クリックし、**設定** を選択します。
   レシピ設定ウィンドウが開きます。
3. 記録する**使用可能なワークエリア**で**コンベア WA** をクリックします。
4. **記録設定**の**記録シーン**チェックボックスを選択します。



xx2100001677

次のページに続く

© Copyright 2025 ABB. All rights reserved.

> **注記**
>
> 記録シーンが選択された場合、どの作業範囲でも保存されると、以下のメッセージがポップアップ表示される。
>
> `Scenes recording is activated for: {0}`
>
> その後、シミュレーションや生産が開始されると、自動的に録画が開始される。
>
> 同じ操作のすべてのワークエリアのシーンが 1 つの `.xml` ファイルに保存されます。

5 必要に応じて記録時間を設定します。
6 **OK**をクリックして設定を適用します。
7 **OK** をクリックして **レシピ** ダイアログを閉じます。
8 生産を実行して、記録を開始します。

作成された `.xml` ファイルは、接続されたRuntime PC の `C:\ProgramData\ABB\PickMaster Twin\PickMaster Twin Runtime\RecordedScene` フォルダに保存されます。

> **ヒント**
>
> 記録シーンのファイル パスは、接続されたRuntime PC 上の既存のフォルダに変更できます。

ゴースト ピッキング フローのソリューションへの追加

この手順でゴースト ピッキング フローを追加します。

1 PickMaster PowerPacリボン タブで、**プロセス** をクリックします。
2 リボン タブで、**フロー** をクリックします。
フロー ウィンドウが開きます。
3 **記録済み** タブをクリックします。
4 **フロー タイプ** でフローのタイプを選択します。

> **注記**
>
> ここで選択したタイプは、インポートされたフロー タイプと同じである必要があります。そうでない場合、フローは正常に動作しません。

5 **フローのインポート** アイコンをクリックして、事前定義済みの作業エリア `.xml` ファイルをインポートします。
6 **開く** をクリックして設定を適用します。
7 **アイテム/コンテナパターン**のドロップダウン リストをクリックして、インポートされたアイテム/コンテナを現在のフローのアイテム/コンテナと一致させます。
8 **OK** をクリックして保存し、**フロー** ダイアログを閉じます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

ポジションジェネレータの変更

この手順を使用して、ゴースト ピッキング フローのポジションジェネレータを変更します。

1 ツリー ビューの**レイアウト**で1つの**ポジションジェネレータ**を右クリックし、**設定** を選択します。

ポジションジェネレータ ウィンドウが開きます。

2 記録されたフローを追加したコンベアをクリックします。

3 **ソース タイプ**で**ビジョン**が選択されていることを確認します。

4 ドロップダウン リストで、このコンベアに使用されているカメラが選択されていることを確認します。

5 **トリガー設定**で距離が選択されていることを確認します。

6 **OK** をクリックして、**ポジションジェネレータ** ウィンドウを閉じます。

ゴーストピッキングフローの選択 (レシピ変更)

この手順でゴースト ピッキング フローを選択します。

1 ツリー ビューの**プロセス** で 1 つの**レシピ** を右クリックし、**設定** を選択します。

レシピ設定ウィンドウが開きます。

2 ターゲット操作 (例: **操作_1**) をクリックして、操作の設定ウィンドウを開きます。

3 **フローの選択**で、ドロップダウン リストからターゲットフロー (例: **フロー1**) を選択します。

> **ヒント**
>
> コンベアのソース タイプが [事前定義済み] に設定されている場合、またはコンベアにカメラが追加されていない場合、このコンベアに対してフローを選択することはできません。

4 ターゲットコンベアワークエリア (例: **コンベア WA 1**) を、**距離の設定** タブの下の **受け入れ** にドラッグして追加します。

5 **OK** をクリックして、**レシピ設定**ウィンドウを閉じます。

ゴーストピッキング フロー

ゴースト ピッキング フローを実行するには、次の手順を使用します。

1 PickMaster PowerPacリボン タブで、**操作** をクリックします。

2 ツリービューで実行するレシピを選択し、**コントロール**を 、リボンタブの操作でクリックします。

コントロールダイアログが開きます。

3 **開始** をクリックして、生産を開始します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

注記

ゴースト ピッキングはデフォルトでループに設定されています。 停止アイコンがクリックされるまで、記録されたポジションデータをリアルコントローラに送信し続けます。

注記

ゴースト ピッキング フローを実行している場合、詳細なビジョンは適用されません。

エミュレーションが実行を開始します。

4 停止をクリックして、生産でのゴースト ピッキングを停止します。

© Copyright 2025 ABB. All rights reserved.

5.5 複数ブランドのカメラの設定

概要

ABB 検証済みカメラの他に、PickMaster Twin は、以下の仕様に従う CVL9 に基づく他の GigE カメラもサポートしています。

カメラの仕様	説明
インターフェース	GigE
標準	カメラの汎用インターフェイス (GenICam)
サポートされている機能	• グローバルシャッター • タイムスタンプ • IO トリガーとストロボ
ビデオフォーマット	モノクロモード

ABB 検証済みカメラについては、検証済みカメラ ページ 59を参照してください。

ABB 検証済みでないカメラの場合、カラーカメラはモノクロモードでのみ使用できます。サポートされているカメラの詳細については、*Cognex Vision Pro* を参照してください。ABB は、代替カメラの正常な動作について責任を負いません。

設定ファイルのパラメータ

ABB 検証済みカメラの場合、設定ファイルが提供され、自動的に読み込まれます。定義済みの設定ファイルは、`C:\Program Files (x86)\ABB\PickMaster Twin 3\Samples\CameraConfig` にあります。定義済みの設定ファイルは、他の ABB 検証済みでないカメラのテンプレートファイルとして使用できます。

注記

定義済みの設定ファイルに変更を加える場合は注意してください。

定義済みの ABB 検証済みカメラの設定ファイルに変更を加えると、設定ファイルを正常に使用できなくなる可能性があります。

ABB 検証済みでないカメラの場合、定義済みの設定ファイルは提供されません。ユーザーは、カメラを設定する際に、テンプレート ファイルに従ってカメラ設定ファイルを定義し、それを PickMaster Twin にロードする必要があります。ABB は、代替カメラの正しい機能については責任を負いません。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

カメラ設定ファイルに必要な基本パラメータは、以下のとおりです:

グループ	機能ノード名	機能タイプ	機能有効化モード	機能値
	(ABB 対応カメラで使用される例)	データタイプ : • 1: ブールまたは文字列 • 2: 整数 • 3: 倍精度または浮動小数点数 • 4: 列挙型 • 5: コマンド	ストロボをオフラインで有効化[i] • 0: 未定義 • 1: eStrobeOff、[ストロボをオフラインで有効化] チェックボックスがオフになっています。 • 2: eStrobeOn、[ストロボをオフラインで有効化] チェックボックスがオンになっています。このオプションをお勧めします。	(ABB 対応カメラで使用される例)
トリガー設定	LineSelector	1	0	Line1
	LineDebouncerTimeAbs	3	0	3000
画像取得設定	AcquisitionFrameRateEnable	1	0	1
	AcquisitionFrameRateAbs	3	0	16
タイマー設定	TimerSelector	4	0	Timer1
	TimerDurationRaw	2	1	0
			2	1000
ストロボ設定	LineSelector	4	0	Out1
	LineSource	4	0	Timer1Active

i [ストロボをオフラインで有効化] チェックボックスをオンにすることをお勧めします。

注記

機能の特定のノード名と値は、カメラのブランドやタイプによって異なる場合があります。特定のノード名を取得するには、特定のカメラのマニュアルまたはメーカーのカメラ設定アプリを参照してください。

注記

サードパーティのビジョン ソフトウェアがインストールされている場合は、ビジョンネットワークの構成 ページ 54.に従ってビジョン ネットワーク設定を確認してください。

サードパーティのソフトウェアを使用すると、意図しない設定変更が発生し、カメラが誤動作する可能性があります。

設定ファイルのアップロード

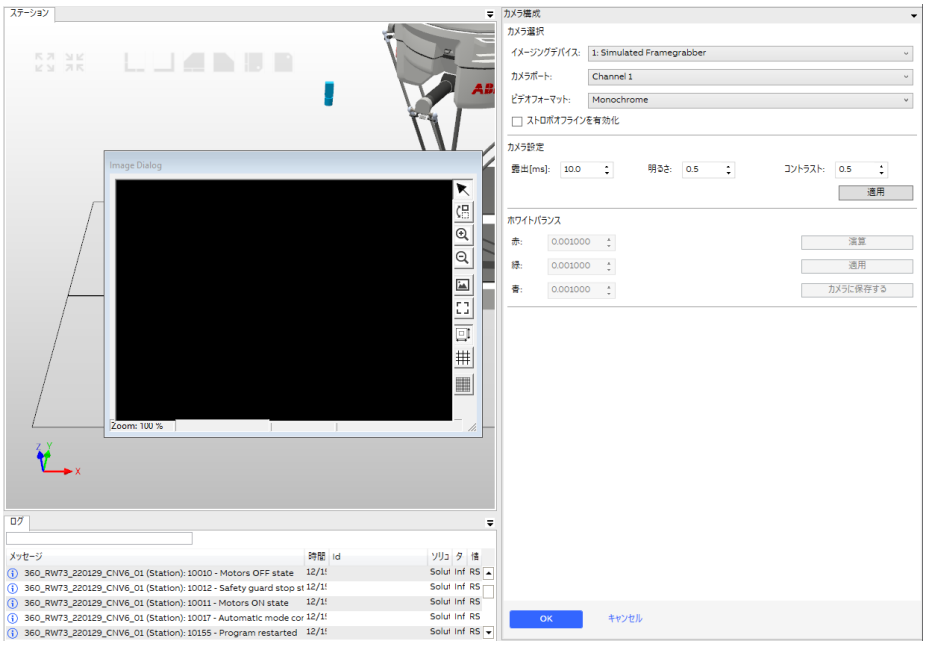
定義済みの設定ファイルをアップロードしてカメラを設定するには、次の手順に従います。

1 テンプレートと設定ファイルのパラメータに従って設定ファイルを準備します。

次のページに続く

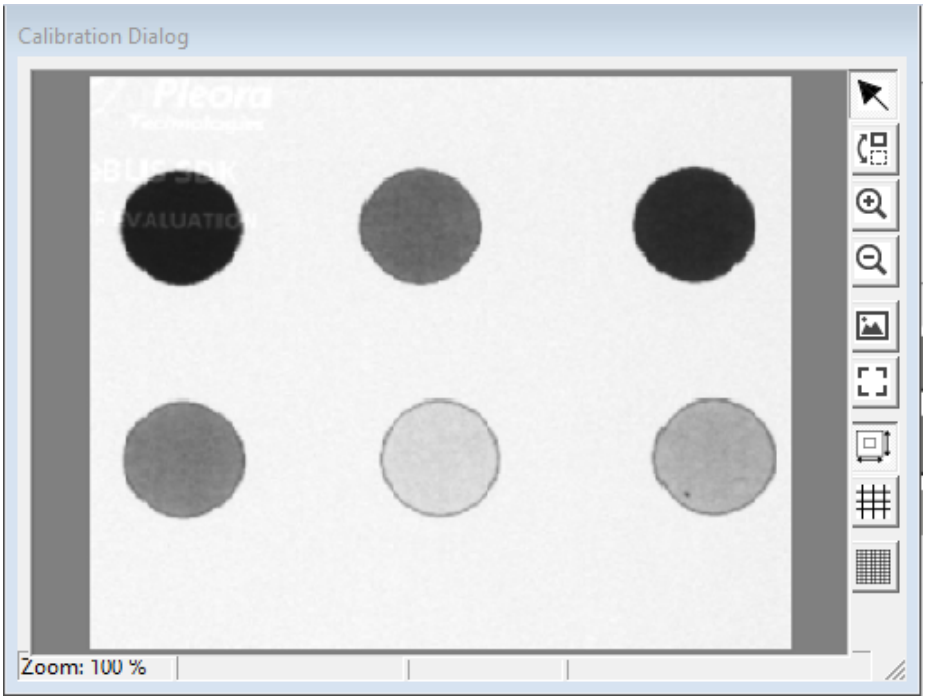
© Copyright 2025 ABB. All rights reserved.

2 PickMaster PowerPacの RRT に接続した後、ツリー ビュー **レイアウト**で目的のカメラを右クリックし、[設定] を選択します。
カメラの設定 ダイアログと **画像ダイアログ**が開きます。



xx2100001622

3 **イメージング デバイス** リストで、ターゲットのカメラを選択します。
4 **ビデオフォーマット**リストで、カメラのタイプを選択します。
画像ダイアログに画像が表示されます。



xx2100001521

5 **設定ファイル**タップで、準備した対応するカメラ設定ファイルを選択します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

続き

6 残りの部分を カメラの設定 ページ *190* として設定します。

© Copyright 2025 ABB. All rights reserved.

6 RAPIDリファレンス

6.1 命令

6.1.1 AckItmTgtーアイテムターゲットの承認

使用法

AckItmTgt は、GetItmTgt を用いてアイテムソースから取得した itmtgt が使用済みであること（例えば、ロボットにより処理されている、スキップされている、後で使用するためにキューに戻されているなど）を認知するために使用します。通常、認知はパス上の TriggL イベントとして設定します（sourcedata からの Ack または Nack トリガーデータを使用）。これは、ターゲットに関係する移動が完了する前に認知が行われないようにするためです。ただし、取得された itmtgt をスキップするか、後で使用するためにキューに戻す場合は、ターゲットに関係する移動は必要なくなることがあります。この場合は、代わりこの命令を使うと便利です。認知が行われて初めて、新規 itmtgt をアイテムソースから取り出すことが可能になります。

基本的な例

```
VAR itmtgt PlaceTarget;
GetItmTgt ItmSrcData{Index}.ItemSource, PlaceItem;
AckItmTgt ItmSrcData{Index}.ItemSource, PlaceItem, FALSE
     \Skip:=TRUE;
```

引数

```
AckItmTgt ItemSource ItemTarget Acknowledge [\Skip] [\Type]
```

ItemSource

データタイプ： itmsrc

GetItmTgt がアイテムターゲットを受け取るアイテムソースです。

ItemTarget

データタイプ： itmtgt

承認するアイテムターゲットです.

Acknowledge

データタイプ： bool

承認の状態。ロボットにより itmtgt が処理（ピックまたはプレイス）された場合は TRUE となります。それ以外の場合は FALSE となり、itmtgt がキューに戻されます。

Skip

データタイプ： bool

itmtgt をスキップする必要があるかどうかを示します。TRUE に設定されている場合、GetItmTgt を用いて itmtgt を再取得することはできません。Acknowledge = FALSE と組み合わせると、下流のロボットが処理できるように itmtgt が流されます。Acknowledge = TRUE と組み合わせると、スキップは無効になります。ス

次のページに続く

© Copyright 2025 ABB. All rights reserved.

6.1.1 AckItmTgtーアイテムターゲットの承認
続き

キップが FALSE に設定された場合、itmtgt はロボットによって処理されるものと判断する（Acknowledge = TRUE と組み合わされている場合）か、または後で使用するためにキューに戻されます（Acknowledge = FALSE と組み合わされている場合）。

Type

データタイプ： num

itmtgt のタイプを変更します。Acknowledge = FALSE および Skip = TRUE と組み合わせると、新規アイテムタイプの構成済み分配に従って、下流のロボットにアイテムが流されます。

Acknowledge = FALSE および Skip = FALSE と組み合わせると、アイテムが新規アイテムタイプでキューに戻され、GetItmTgt を用いて取得することが可能です。アイテムタイプはローカルでのみ変更可能です。アイテムタイプとアイテムの分配は下流のロボットでも変わりません。

Acknowledge = TRUE と組み合わせると、スキップは無効になります。

エラーの処理

次のような、復元可能なエラーが生成される場合があります。エラーは、エラーハンドラで対処できます。システム変数ERRNOを次のように設定します:

エラーコード	説明
ERR_ITMSRC_UNDEF	itmsrc 未定義。

制限事項

itmtgt は GetItmTgt 命令を用いて取得する必要があります。

構文

```
AckItmTgt
  [ItemSource ':=' ] <variable (VAR) of itmsrc>,
  [ItemTarget ':=' ] <var or pers (INOUT) of itmtgt>,
  [Acknowledge':=' ] <expression (IN) of bool>,
  [\Skip ':=' ] <expression (IN) of bool>,
  [\Type ':=' ] <expression (IN) of num>;
```

関連情報

下記の詳細は	次で説明されています
データタイプ itmtgt	*itmtgt - アイテムターゲットデータ ページ 410*。

© Copyright 2025 ABB. All rights reserved.

6.1.2 FlushItmSrcーアイテムソースのフラッシュ

使用法

FlushItmSrcは、アイテム ソースをフラッシュするために使用されます。命令は、アイテム ソース バッファをクリアし、シーン番号をリセットし、コンベア トラッキング ボードをリセットします。

基本的な例

```
FlushItmSrc PlaceSource;
```

以前作成されたアイテムソースオブジェクト*PlaceSource*をフラッシュします.

引数

```
FlushItmSrc ItemSource
```

`ItemSource`

データタイプ： itmsrc

作成されたアイテムソースです.

エラーの処理

次の復元可能なエラーが生成できます。エラーはエラーハンドラで処理できます。システム変数 ERRNO は次のように設定されます。

エラーコード	説明
ERR_ITMSRC_UNDEF	itmsrc 未定義

制限事項

問題の発生を避けるために、この命令は最後のアイテムターゲットが承認された場合のみ実行されます。

構文

```
FlushItmSrc
  [ItemSource ':=' ] <variable (VAR) of itmsrc>;
```


© Copyright 2025 ABB. All rights reserved.

6.1.3 GetItmTgt - 次のアイテムターゲットを取得する

使用法

GetItmTgt は、ワークエリアのエンタ限界と終了限界間でアイテムソースキューで使用可能な次の itmtgt を取得するために使用します。RAPID プログラムは、次のアイテムに到達可能になるか、タイムアウトするまでこの命令で待機します。

基本的な例

命令GetItmTgtの基本的な例を下記に示します.

例1

```
GetItmTgt PlaceSource, PlaceItem;
```

使用可能なプレイスアイテムが存在する場合は、PlaceSourceからこれを取り出します.

例2

```
...
VAR selectiondata neg_y_sort;

neg_y_sort.ShapeType:=BOX;
neg_y_sort.ConsiderType:=BitOr(ITEMS_TO_USE,ITEMS_BYPASS);
neg_y_sort.GeometricData.x:=60;
neg_y_sort.GeometricData.y:=500;
neg_y_sort.GeometricData.z:=10;
neg_y_sort.GeometricData.radius:=0;
neg_y_sort.Offset.OffsetRelation:=FRAME_COORD_DIR;
neg_y_sort.Offset.OffsetPose.trans.x:=0;
neg_y_sort.Offset.OffsetPose.trans.y:=-500;
neg_y_sort.Offset.OffsetPose.trans.z:=0;
neg_y_sort.Offset.OffsetPose.rot.q1:=1;
neg_y_sort.Offset.OffsetPose.rot.q2:=0;
neg_y_sort.Offset.OffsetPose.rot.q3:=0;
neg_y_sort.Offset.OffsetPose.rot.q4:=0;
IF pick_type = 2 THEN pick_type := 1; ELSE
  pick_type := 2
ENDIF

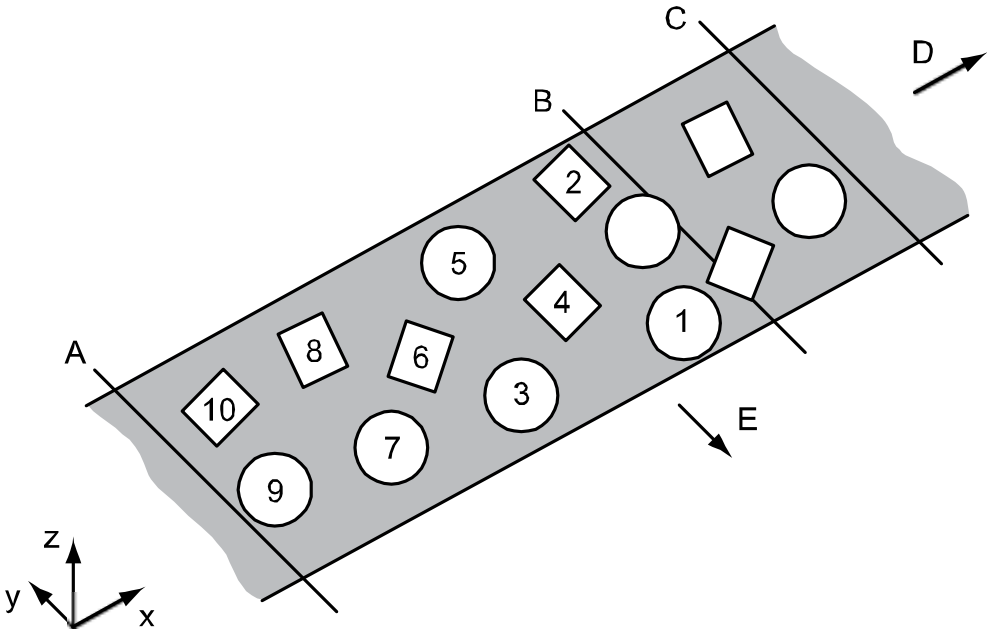
GetItmTgt PickSource, PickItem \ItemType:=pick_type \Limit:=100
     \Selection:=neg_y_sort;
```

マイナスYの並べ替え、およびタイプリクエストに基づいて、*PickSource* からピックアイテムを取り出します。タイプは、2 つのタイプが交互に切り替わります。Limit 引数は、検索がどこから開始されるかを指定します。

下図の例では、プラスのX方向、マイナスのY方向に並べ替え、2つの異なるオブジェクトタイプで作業を実行しています。2 つのオブジェクトタイプが、円形のワークから交互に選択されます。これにより、図で番号付けされたとおりの順番（1 ~ 10）が与えられます.

次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx0900000451

A	エンター
B	境界チェック
C	終了
D	製品フロー方向
E	並べ替え方向
1-10	並べ替え順序

引数

```
GetItmTgt ItemSource, ItemTarget [\MaxTime] [\TimeFlag] [\ItemType]
     [\Limit] [\EnterCheckLimit] [\ExitCheckLimit] [\SortData]
     [\Selection] [\Val1Min] [\Val1Max] [\Val2Min] [\Val2Max]
     [\Val3Min] [\Val3Max] [\Val4Min] [\Val4Max] [\Val5Min]
     [\Val5Max]
```

`ItemSource`

データタイプ： `itmsrc`

アイテムターゲットが取り出される、アイテムソースです.

`ItemTarget`

データタイプ： `itmtgt`

取り出されるアイテムターゲットです.

`[\MaxTime]`

データタイプ： `num`

許可された最大待機時間です（秒単位）。アイテムターゲットが取り出される前にこの時間を超過し、`TimeOut` フラグが指定されていない場合、エラーハンドラがエラーコード `ERR_PPA_TIMEOUT` で呼び出されます。エラーハンドラが存在しない場合は実行が停止されます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

6.1.3 GetItmTgt - 次のアイテムターゲットを取得する
続き

[\TimeFlag]

データタイプ： bool

アイテムターゲットが取り出される前に許可された最大待機時間を 超過した場合の、TRUE の値を含む出力パラメータです。このパラメータが命令に含まれている場合、最大時間を超過しても、エラーとはみなされません。この引数は、命令に MaxTime 引数が含まれていない場合は無視されます。

[\ItemType]

データタイプ： num

リクエストされるアイテムタイプ番号を指定します。命令は、リクエストされたタイプ番号を持つアイテムターゲットが実行可能になるまで待ちます.

[\Limit]

データタイプ： num

非推奨となり、[\ExitCheckLimit] に置き換えられました。

[\EnterCheckLimit]

データタイプ： num

キューからアイテムターゲットを取得できる距離を変更します。この命令は、次のターゲットを取得できる範囲を制限します。この引数を指定しない場合、命令はPickMasterアプリケーションによって指定されたEnter制限を超えた次のアイテムターゲットを返します。

ロボットの中心からの距離はミリメートル単位で指定します。正の値は、EnterCheckLimitがフィーダーの移動方向においてロボットの中心を超えていることを意味します。この引数はコンベアを使用する場合にのみ有効です。

この引数は、入力制限を一時的に制限する必要がある場合に役立ちます。

[\ExitCheckLimit]

データタイプ： num

アイテムターゲットを無視する距離を変更します。この距離を超えるアイテムターゲットは考慮されません。この引数を省略した場合、PickMasterアプリケーションによって指定されたExit制限が考慮されます。

ロボットの中心からの距離はミリメートル単位で指定します。正の値は、ExitCheckLimitがフィーダーの移動方向においてロボットの中心を超えることを意味します。この引数はコンベアを使用する場合にのみ有効です。

この引数は、Exit 制限を一時的に制限する必要がある場合に役立ちます。

[\SortData]

データタイプ： sortdata

このデータ構造は、アイテムの並べ替え方法を定義します.

[\Selection]

データタイプ： selectiondata

このデータ構造は、アイテムの選択方法を定義します.

[\Val1Min]

データタイプ： num

次のページに続く

© Copyright 2025 ABB. All rights reserved.

itmtgt パラメータ Val1 の最小値を指定します。この条件を満たすアイテムターゲットが実行可能になるまで待機します。

[\Val1Max]

データタイプ： num

itmtgt パラメータ Val1 の最大値を指定します。この条件を満たすアイテムターゲットが実行可能になるまで待機します。

[\Val2Min]

データタイプ： num

itmtgt パラメータ Val2 の最小値を指定します。この条件を満たすアイテムターゲットが実行可能になるまで待機します。

[\Val2Max]

データタイプ： num

itmtgt パラメータ Val2 の最大値を指定します。この条件を満たすアイテムターゲットが実行可能になるまで待機します。

[\Val3Min]

データタイプ： num

itmtgt パラメータ Val3 の最小値を指定します。この条件を満たすアイテムターゲットが実行可能になるまで待機します。

[\Val3Max]

データタイプ： num

itmtgt パラメータ Val3 の最大値を指定します。この条件を満たすアイテムターゲットが実行可能になるまで待機します。

[\Val4Min]

データタイプ： num

itmtgt パラメータ Val4 の最小値を指定します。この条件を満たすアイテムターゲットが実行可能になるまで待機します。

[\Val4Max]

データタイプ： num

itmtgt パラメータ Val4 の最大値を指定します。この条件を満たすアイテムターゲットが実行可能になるまで待機します。

[\Val5Min]

データタイプ： num

itmtgt パラメータ Val5 の最小値を指定します。この条件を満たすアイテムターゲットが実行可能になるまで待機します。

[\Val5Max]

データタイプ： num

itmtgt パラメータ Val5 の最大値を指定します。この条件を満たすアイテムターゲットが実行可能になるまで待機します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

プログラムの実行

バッファにアイテムターゲットが存在しない場合、またはワークエリアで使用可能なアイテムターゲットが存在しない場合、プログラム実行はアイテムがワークエリア内であると見なされるようになるまで、この命令で待機します.

MaxTime 引数が指定された場合は 待機時間が監視されます。その待機時間が MaxTime の値を超過し、TimeFlag 引数が使用されている場合、プログラムはそのまま続行します。TimeFlag が使用されない場合は、 エラーが発生します。TimeFlag が指定され、待機時間を超過した場合は TRUE に設定され、そうでない場合は FALSE に設定されます。

Limit 引数は、アイテムターゲットの抜き取りが始まる、開始ポイントを指定します。

SortData 引数が指定された場合、命令は X 方向の終了限界に最も近いアイテムターゲットを返し、並べ替え方向にその他のオブジェクトが存在すれば並べ替え方向の最初のオブジェクトが選択されます。CheckBoundry 間隔は、オブジェクト周辺に要求されるクリアランス間隔を定義します。並べ替えは、生産フローの上方向および下方向において、その他のアイテムターゲットの有無をチェックします。この引数が Limit 引数と組み合わせて使用されている場合、並べ替えアルゴリズムは、最も近いオブジェクトの安全間隔をチェックする際、チェック限界と終了限界の間にあるすべてのオブジェクトを対象とします。負荷分散や ATC などの共有ポジションソースシステムで複数のロボットが使用されている場合、並べ替えでは SortData がバイパスするアイテムを取得しないため、 Selection 引数を使用してください。

Selection 引数を指定する場合、命令は X 方向の終了限界に最も近く、指定形状内に他のアイテムターゲットを持たないアイテムターゲットを返します。この引数が Limit 引数と組み合わせて 使用されている場合、最も近いオブジェクトの間隔をチェックする際に、並べ替えアルゴリズムはチェック限界と終了限界の間にあるすべてのオブジェクトを対象とします。

オプションの引数 Va1Xmin または Va1Xmax に対して値が指定されている場合は、Va1X に対して必要な最大値と最小値を満足するアイテムターゲットを命令で返します。

エラーの処理

次のような、復元可能なエラーが生成される場合があります。エラーは、エラーハンドラで対処できます。システム変数ERRNOを次のように 設定します:

エラーコード	説明
ERR_ITMSRC_UNDEF	itmsrc 未定義。
ERR_PPA_TIMEOUT	エラーフラグなしでタイムアウトになりました。

構文

```
GetItmTgt
  [ItemSource ':=' ] <variable (VAR) of itmsrc>,
  [ItemTarget ':=' ] <var or pers (INOUT) of itmtgt>
  [\MaxTime ':=' ] <expression (IN) of num>
  [\TimeFlag ':=' ] <var or pers (INOUT) of bool>
  [\ItemType ':=' ] <expression (IN) of num>
```

次のページに続く

© Copyright 2025 ABB. All rights reserved.

```
[\Limit ':=' ] <expression (IN) of num>
[\SortData ':=' ] <expression (IN) of sortdata>
[\Selection ':=' ] <expression (IN) of selectiondata>
[\Val1Min ':=' ] <expression (IN) of num>
[\Val1Max ':=' ] <expression (IN) of num>
[\Val2Min ':=' ] <expression (IN) of num>
[\Val2Max ':=' ] <expression (IN) of num>
[\Val3Min ':=' ] <expression (IN) of num>
[\Val3Max ':=' ] <expression (IN) of num>
[\Val4Min ':=' ] <expression (IN) of num>
[\Val4Max ':=' ] <expression (IN) of num>
[\Val5Min ':=' ] <expression (IN) of num>
[\Val5Max ':=' ] <expression (IN) of num>;
```

関連情報

下記の詳細は	次で説明されています
データタイプ itmtgt	*itmtgt - アイテムターゲットデータ ページ 410*。
データタイプ selectiondata	*selectiondataー選択データ ページ 413*。
データタイプ sortdata	*sortdataー並べ替えデータ ページ 416*。

© Copyright 2025 ABB. All rights reserved.

6.1.4 NextItmTgtType - 次のアイテムターゲットのタイプを取得

使用法

NextItmTgtType は、アイテムソースバッファに含まれる次のアイテムターゲット（itmtgt）のタイプを取得するために使用します。Limit 間隔パラメータが 指定されている場合、命令は、限界より上の次のアイテムターゲットのタイプを返します。RAPID プログラムは、このキューにアイテムが存在するようになるまで、この命令で待機します。

基本的な例

```
NextItmTgtType PlaceSource, PlaceType
```

PlaceSource に含まれる、次の itmtgt タイプを取り出します。

引数

```
NextItmTgtType ItemSource ItemType [\Limit] [\MaxTime] [\TimeFlag]
```

ItemSource

データタイプ： itmsrc

アイテムターゲットタイプが取り出される、アイテムソースです.

ItemType

データタイプ： num

取り出されるアイテムターゲットタイプです.

[\Limit]

データタイプ： num

これは、タイプが取り出される限界地点です。命令は、この限界以前の、次のアイテムターゲットタイプを返します。この引数が除外されると、命令は、終了限界以前の次のアイテムターゲットタイプを返します.

距離は、ロボットの中心からミリ単位で計算されます。限界が、コンベヤの移動方向にロボット中心よりも後ろの場合は、正の数になります.

この引数は、コンベヤが使用されている場合に限り有効です.

[\MaxTime]

データタイプ： num

許可された最大待機時間です（秒単位）。アイテムターゲットが取り出される前にこの時間を超過し、TimeOut フラグが指定されていない場合、エラーハンドラがエラーコード ERR_PPA_TIMEOUT で呼び出されます。エラーハンドラが存在しない場合、実行は停止されます。

[\TimeFlag]

データタイプ： bool

アイテムターゲットが取り出される前に許可された最大待機時間を超過した場合の、TRUE の値を含む出力パラメータです。このパラメータが命令に含まれている場合、最大時間を超過しても、エラーとはみなされません。

この引数は、MaxTime 引数が使用される場合のみ使用されます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

プログラムの実行

バッファ内にアイテムターゲットが存在しない場合、またはLimitより上にアイテムターゲットが存在しない場合、プログラム実行はバッファにアイテムが存在するようになるまでこの命令で待機します。

MaxTime 引数が指定された場合は 待機時間が監視されます。その待機時間がMaxTimeの値を超過し、TimeFlag引数が使用されている場合、プログラムはそのまま続行します。TimeFlag が使用されない場合は、 エラーが発生します。TimeFlag が指定され、待機時間を超過した場合は TRUE に設定され、そうでない場合は FALSE に設定されます。

エラーの処理

次の復元可能なエラーが生成できます。エラーはエラーハンドラで処理できます。システム変数 ERRNO は次のように設定されます。

エラーコード	説明
ERR_ITMSRC_UNDEF	itmsrc 未定義。
ERR_PPA_TIMEOUT	エラーフラグなしでタイムアウトになりました

構文

```
NextItmTgtType
  [ItemSource ':=' ] <variable (VAR) of itmsrc>,
  [ItemType ':=' ] <var or pers (INOUT) of num>
  [\Limit ':=' ] <expression (IN) of num>
  [\MaxTime ':=' ] <expression (IN) of num>
  [\TimeFlag ':=' ] <var or pers (INOUT) of bool>;
```

関連情報

下記の詳細は	次で説明されています
データタイプ itmtgt	*itmtgt - アイテムターゲットデータ ページ 410*。

© Copyright 2025 ABB. All rights reserved.

6.1.5 QStartItmSrc - アイテムソースでキューを開始

使用法

QStartItmSrcは、アイテムソースでキューを開始するために使用します。この命令は、新しいプログラムを開始する際、またはフラッシュの後で使用します.

基本的な例

```
QStartItmSrc PlaceSource;
```

アイテムソース*PlaceSource*に含まれるオブジェクトのキューが開始されます.

引数

```
QStartItmSrc ItemSource
```

ItemSource

データタイプ： itmsrc

開始されたアイテムソースです.

エラーの処理

次の復元可能なエラーが生成できます。エラーはエラーハンドラで処理できます。システム変数 ERRNO は次のように設定されます。

エラーコード	説明
ERR_ITMSRC_UNDEF	itmsrc 未定義

構文

```
QStartItmSrc
  [ItemSource ':=' ] <variable (VAR) of itmsrc>;
```

関連情報

下記の詳細は	次で説明されています
命令 QStopItmSrc	*QStopItmSrc - アイテムソースでキューを停止 ページ 399*。

© Copyright 2025 ABB. All rights reserved.

6.1.6 QStopItmSrc - アイテムソースでキューを停止

使用法

QStopItmSrcは、アイテムソースでキューを停止するために使用します.

基本的な例

```
QStopItmSrc PlaceSource;
```

アイテムソース*PlaceSource*に含まれるオブジェクトのキューが停止されます.

引数

```
QStopItmSrc ItemSource
```

ItemSource

データタイプ： itmsrc

停止されたアイテムソースです.

エラーの処理

次の復元可能なエラーが生成できます。エラーはエラーハンドラで処理できます。
システム変数 ERRNO は次のように設定されます。

エラーコード	説明
ERR_ITMSRC_UNDEF	itmsrc 未定義

構文

```
QStopItmSrc
  [ItemSource ':=' ] <variable (VAR) of itmsrc>;
```

関連情報

下記の詳細は	次で説明されています
命令 QStartItmSrc	*QStartItmSrc - アイテムソースでキューを開始 ページ 398*。

© Copyright 2025 ABB. All rights reserved.

6.1.7 ResetFlowCount - フローカウンタをリセット

使用法

ResetFlowCount はフローカウントのリセットに使用します。フローカウンタは、前回のリセット後にコンベヤワークエリアの終了限界を超えたオブジェクト数を表示します。フローカウンタの値は、GetFlowCount 関数を用いて取得できます。

基本的な例

```
ResetFlowCount PlaceSource;
```

あるアイテムソースのフローカウンタをリセットします。

引数

```
ResetFlowCount ItemSource
```

ItemSource

データタイプ： itmsrc

アイテム ソース。

エラーの処理

次のような、復元可能なエラーが生成される場合があります。このエラーにはエラーハンドラで対処できます。システム変数 ERRNO は次のように設定します。

エラーコード	説明
ERR_ITMSRC_UNDEF	itmsrc 未定義

構文

```
ResetFlowCount[ItemSource ':=' ] <variable (VAR) of itmsrc>;
```

関連情報

下記の詳細は	次で説明されています
機能 GetFlowCount	*GetFlowCount - 限界を超えたアイテム数の取得 ページ 409*。

© Copyright 2025 ABB. All rights reserved.

6.1.8 ResetMaxUsageTime - 最大測定使用時間をリセットします。

説明

ResetMaxUsageTime(最大使用時間のリセット) は、以前に処理された物体の測定された最大使用時間をリセットするために使用されます。 これは、*GetItmTgt* でターゲットを受信してから、ロボットが物体を処理するまでの時間 (確認時間) です。 *ResetMaxUsageTime* は、*PickMaster Ready* でのみ利用できます。

例

```
ResetMaxUsageTime ItmSrcData{PickWorkArea{1}}.ItemSource;
```

あるアイテムソースの最大測定使用時間をリセットします。

引数

```
ResetMaxUsageTime ItemSource
```

`Item Source`

ItemSource

データタイプ： `itmsrc`

アイテム ソース。

エラーの処理

次の復元可能なエラーが生成できます。エラーはエラーハンドラで処理できます。システム変数 `ERRNO` は次のように設定されます。

`ERR_ITMSRC_UNDEF`	**itmsrcは未定義です。**

構文

```
ResetMaxUsageTime[ItemSource ':=' ] <variable (VAR) of itmsrc>;
```


© Copyright 2025 ABB. All rights reserved.

6.1.9 UseReachableTargets - 達せられるターゲットを使用する

説明

UseReachableTargets は、ロボットが物体処理のために到達可能なターゲットのみを受け取る機能モードをアクティブにするために使用されます。

アクティブにすると、*GetItmTgt* を使用したターゲット リクエストに対して、到達不能なターゲットが除外されます。

UseReachableTargets は、可変サイズで、最適ターゲットリリースゾーンを設定します。リリースゾーンのサイズは、ロボットの対応可能範囲とコンベヤのリアルタイム速度によって異なります。コンベヤ速度が上がると、リリースゾーンのサイズが小さくなるため、使用可能なターゲットの量が少なくなります。コンベヤ速度が速すぎると、リリースゾーンが完全に消え、速度を落とすまで、ターゲットは捕捉されなくなります。

UseReachableTargets(リーチ可能なターゲットの使用) は、*PickMaster Ready* オプションでのみ使用できます。

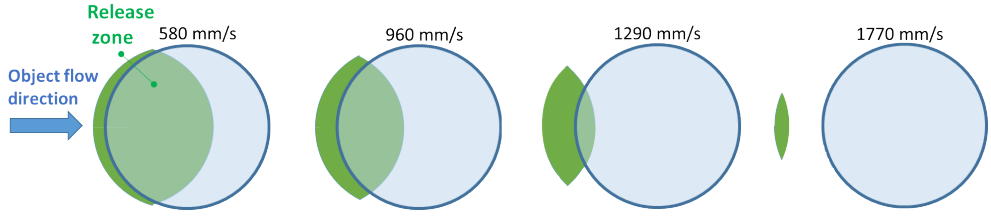
警告

ターゲットリリースゾーンが進入/退出限定のセクションによることです。アプリケーションマニュアル - *PickMaster 3*を参照してください。結果とするターゲットリリースゾーンが最適なターゲットリリースゾーンと進入/退出の領域の交差点です。

最適ターゲットで衝撃をさけるための推薦された退出/進入の値が次のように：

- 進入= -10000 mm（これは、ロボットは上流の方向にいい距離という意味である）
- 退出= 10000 mm（これは、ロボットは下流の方向にいい距離という意味である）

下図は、4つの異なる速度での IRB 360 (上記のとおり) のターゲットリリースゾーンを示しています。ライトブルーの領域は、ロボットの作業レンジ、グリーンの領域はターゲットリリースゾーンです。



xx1800000383

例

```
UseReachableTargets ItmSrcData{PlaceWorkArea{1}}.ItemSource, TRUE,
     0.7 \ReleaseTime:=0.1;
GetItmTgt PlaceSource, PlaceItem;
```

UseReachableTargets をリニアコンベヤの作業領域で有効かします。使用中のターゲットは、 *GetItmTgt* で捕捉された後、最大 0.7 秒以内に配置されます。ターゲッ

次のページに続く

© Copyright 2025 ABB. All rights reserved.

トは、ロボット対応範囲に入る0.1秒前に利用できるようになります。続いて、ターゲットは、リリースゾーンを離れるまで、使用可能のままです。

引数

```
UseReachableTargets ItemSource, Enable, UsageTime [\ReleaseTime]
```

`Item Source`

ItemSource

データタイプ： `itmsrc`

UseReachableTargets **を有効化するアイテムソース。**

`Enable`

Enable

データタイプ： `bool`

これで ***UseReachableTargets*** **を有効化/無効化します。**

`Usage Time`

UsageTime

データタイプ： `num`

ターゲットの予想使用時間が*ItmTgt*w p 得るでターゲットを受けるときと対象をロボットで処理した後（承認時刻）の間です。実際の使用時間が連続的に測定します。最大の測定した使用時間が 最大使用時間を得るで受けることができます到着エラーを避けるため、使用時間値が最大の測定した使用時間とマージンの和として定義されます。例えば、使用時間=最大使用時間 + 0.1秒を設定します。大きな安全マージンの欠点は必要ではないターゲットリリースゾーン、選択率を下げることが可能です。

`[\ReleaseTime]`

Release Time

データタイプ： `num`

ReleaseTime**はターゲットがリリースゾーンを入る時間を定義します。ロボットリーチを入る前に、値が負であれば、ターゲットがロボットリーチに入った後でリリースゾーンに入ります。0.1またはもっと小さい値がリーチエラーを避けるために推薦されます。もっと高い値が高速導電体をコントロールするために使用します。低速の場合に、もっと高い値の欠点は上流の方向リーチエラーのリスクを増加します。**

> **注記**
>
> **使用時間 とリリース時間がいつも変更できます。例えば、リーチを避けるために、臨時的なロボット速度の減少が長い使用時間が要求します。**

構文

```
UseReachableTargets
  [ItemSource ':=' ] <variable (VAR) of itmsrc>,
  [Enable ':=' ] <var or pers (IN) of bool>
  [UsageTime ':=' ] <var or pers (IN) of num>
  [\ReleaseTime ':=' ] <expression (IN) of num>;
```

次のページに続く

© Copyright 2025 ABB. All rights reserved.

6.1.9 UseReachableTargets - 達せられるターゲットを使用する
RobotWare - OS
続き

制限事項

ロボット作業領域でモーション構成に制限がある場合、コンベヤ可動方向に垂直な作業領域の最大 20 ㎜ 外側のターゲットが `GetItmTgt` 指示により捕捉される可能性があります。

外部延長エラーを避ける対策として、その方向に移動させる前に、`itemtarget` の Y 値を別途チェックします。

© Copyright 2025 ABB. All rights reserved.

6.2 関数

6.2.1 GetMaxUsageTime - 最大測定使用時間を受ける

説明

GetMaxUsageTime(最大使用時間の取得) は、以前に処理されたオブジェクトの測定された最大使用時間を取得するために使用されます。 *GetItmTgt* でターゲットを受信してから、ロボットがオブジェクトを処理するまでの時間 (確認時間) です。 実際の使用時間は、取り扱うオブジェクトごとに継続的に測定されます。
GetMaxUsageTime は、*PickMaster Ready* でのみ使用できます

例

```
VAR num usetime;
usetime := GetMaxUsageTime(ItmSrcData{PickWorkArea{1}}.ItemSource);
```

使用時間は生産開始後や最大使用時間をリセットするの実行中の最大測定時間です。

返り値

データタイプ： num

生産開始後や最大使用時間をリセットするの実行中の最大測定時間です。

引数

```
GetMaxUsageTime (ItemSource)
```

Item Source

ItemSource

データタイプ： itmsrc

アイテム ソース。

エラーの処理

次の復元可能なエラーが生成できます。これらはエラーハンドラで処理できます。
システム変数 ERRNO は次のように設定されます。

ERR_ITMSRC_UNDEF	*itmsrc*は未定義です。

構文

```
GetMaxUsageTime '('[ItemSource ':=' ] <variable (VAR) of itmsrc>
    ')';
```

この機能がデータの値をnumタイプで返します。

© Copyright 2025 ABB. All rights reserved.

6.2.2 GetQueueLevel - キューレベルを取得

使用法

GetQueueLevelは、特定の条件に合致するアイテムソースにあるアイテムターゲットの現在の数を取得します。

基本的な例

```
reg1 := GetQueueLevel(PlaceSource);
```

reg1 には、アイテムソース PlaceSource にあるアイテムターゲットの現在の数が割り当てられます。

返り値

データタイプ： num

アイテムソースにあるアイテムターゲットの現在の数です.

引数

```
GetQueueLevel (ItemSource [\ItmType] [\MinLimit] [\MaxLimit])
```

ItemSource

データタイプ： itmsrc

アイテムターゲットの現在の数が取得されるアイテムソースです.

\ItmType

データタイプ： num

指定されたタイプ番号のアイテムのみがカウントされます。

\MinLimit

データタイプ： num

アイテムのカウントの開始点となるロボットの中心からの最小距離を定義します。値が負の場合は、限界がロボットの上流にあります。値が正の場合は、限界が下流にあります。パラメータはインデックスワークエリアに影響しません。

\MaxLimit

データタイプ： num

アイテムのカウントの開始点となるロボットの中心からの最大距離を定義します。値が負の場合は、限界がロボットの上流にあります。値が正の場合は、限界が下流にあります。パラメータはインデックスワークエリアに影響しません。

エラーの処理

次の復元可能なエラーが生成できます。エラーはエラーハンドラで処理できます。システム変数 ERRNO は次のように設定されます。

エラーコード	説明
ERR_ITMSRC_UNDEF	itmsrc 未定義

構文

```
GetQueueLevel '('
  [ItemSource ':=' ] <variable (VAR) of itmsrc> ')'
```

次のページに続く

© Copyright 2025 ABB. All rights reserved.

```
[\ItmType ':=' ] <expression (IN) of num>
[\MinLimit ':=' ] <expression (IN) of num>
[\MaxLimit ':=' ] <expression (IN) of num>;
```

データタイプがnumの返り値を持つ関数です.

© Copyright 2025 ABB. All rights reserved.

6.2.3 GetQueueTopLevel - キューのトップレベルを取得

使用法

GetQueueTopLevelは、アイテムソースにあるアイテムターゲットの最も多い数を取得します.

基本的な例

```
reg1 := GetQueueTopLevel(PlaceSource);
```

*reg1*には、アイテムソース*PlaceSource*にあるアイテムターゲットの最も多い数が割り当てられます.

返り値

データタイプ： num

アイテムソースにあった今までのアイテムターゲットの最も多い数です.

引数

```
GetQueueTopLevel (ItemSource)
```

ItemSource

データタイプ： itmsrc

アイテムターゲットの現在の数が取得されるアイテムソースです.

エラーの処理

次のような、復元可能なエラーが生成される場合があります。エラーは、エラーハンドラで対処できます。システム変数ERRNOを次のように 設定します:

エラーコード	説明
ERR_ITMSRC_UNDEF	itmsrc 未定義

構文

```
GetQueueTopLevel '('
  [ItemSource ':=' ] <variable (VAR) of itmsrc> ')';
```

データタイプがnumの返り値を持つ関数です.

© Copyright 2025 ABB. All rights reserved.

6.2.4 GetFlowCount - 限界を超えたアイテム数の取得

使用法

GetFlowCount は、ResetFlowCount の実行後にコンベヤワークエリアの終了限界を超えたアイテム数の取得に使用します。ロボットが処理するアイテムはカウントされません（ピック / プレイスが行われる前に終了限界を超える場合でも）。

基本的な例

```
VAR num counter;
ResetFlowcount PlaceSource;
WaitTime 10;
counter := GetFlowCount(PlaceSource);
```

counter には、終了限界を超えた PlaceSource からのアイテム数が割り当てられます。

返り値

データタイプ： num

ResetFlowCount の実行後に終了限界を超えたアイテム数。

引数

GetFlowCount (ItemSource)

ItemSource

データタイプ： itmsrc

アイテムソース。

エラーの処理

次のような、復元可能なエラーが生成される場合があります。このエラーにはエラーハンドラで対処できます。システム変数 ERRNO は次のように設定します。

エラーコード	説明
ERR_ITMSRC_UNDEF	itmsrc 未定義

構文

```
GetFlowCount '('[ItemSource ':=' ] <variable (VAR) of itmsrc> ')';
```

データタイプが num の値を返す関数。

関連情報

下記の詳細は	次で説明されています
ResetFlowCount 命令	*ResetFlowCount - フローカウンタをリセット ページ 400*。

© Copyright 2025 ABB. All rights reserved.

6.3 データタイプ

6.3.1 itmtgt - アイテムターゲットデータ

使用法

itmtgtは、ピックやプレイスアイテムを説明するために使用します.

説明

Itmtgt は、ピックまたはプレイスするアイテムを指定します。ポジションとその他のデータが含まれます。

コンポーネント

tag

データタイプ： num

アイテムを識別する連番です。ユーザーフックで変更して自由に使用できます。整数値に限定されます。

type

データタイプ： num

アイテムのタイプです.

scene

データタイプ： num

例えば、ビジョンシステムが取得した画像に対応するシーンなど、シーンを識別する連続番号です.

robtgt

データタイプ： robtgt

ピックやプレイスポジションです.

val1

データタイプ： num

これは任意です。ユーザーフックからの情報などのアイテム固有の情報の追加に使用できます。データタイプは浮動小数点です。

val2

データタイプ： num

これは任意です。ユーザーフックからの情報などのアイテム固有の情報の追加に使用できます。データタイプは浮動小数点です。

val3

データタイプ： num

これは任意です。ユーザーフックからの情報などのアイテム固有の情報の追加に使用できます。データタイプは浮動小数点です。

val4

データタイプ： num

次のページに続く

© Copyright 2025 ABB. All rights reserved.

これは任意です。ユーザーフックからの情報などのアイテム固有の情報の追加に使用できます。データタイプは浮動小数点です。

val5

データタイプ： num

これは任意です。ユーザーフックからの情報などのアイテム固有の情報の追加に使用できます。データタイプは浮動小数点です。

例

例1

```
CONST itmtgt pickpos :=
     [1,2,1,0,0,0,0,0,[[20,40,8],[1,0,0,0],[0,0,0,0],
     [9E+9,9E+9,9E+9,9E+9,0,0]]];
```

ピックポジションが定義されています。使用コンベヤに関連する外部軸は0に設定する必要があるため、（9E+9を設定して）未使用としてマークすることはありません。例：コンベヤが2つ存在する場合、最後の2つの外部軸ポジションを0に設定します.

構造

```
<dataobject of itmtgt>
  <tag of num>
  <type of num>
  <scene of num>
  <val1 of num>
  <val2 of num>
  <val3 of num>
  <val4 of num>
  <val5 of num>
  <dataobject of robtarget>
    <trans of pos>
      <x of num>
      <y of num>
      <z of num>
    <rot of orient>
      <q1 of num>
      <q2 of num>
      <q3 of num>
      <q4 of num>
    <robconf of confdata>
      <cf1 of num>
      <cf4 of num>
      <cf6 of num>
      <cfx of num>
    <extax of extjoint>
      <eax_a of num>
      <eax_b of num>
      <eax_c of num>
      <eax_d of num>
      <eax_e of num>
      <eax_f of num>
```

次のページに続く

© Copyright 2025 ABB. All rights reserved.

6.3.1 itmtgt - アイテムターゲットデータ
続き

関連情報

下記の詳細は	次で説明されています
ポジション決め方法	*Technical reference manual - RAPID Overview*
座標系	*Technical reference manual - RAPID Overview*
構成データの処理	*Technical reference manual - RAPID Overview*
外部軸の構成	*Technical reference manual - System parameters*
四元数について	*Technical reference manual - RAPID Overview*

© Copyright 2025 ABB. All rights reserved.

6.3.2 selectiondataー選択データ

使用法

selectiondataは、選択条件の記述に使用します。また、アイテムの並べ替えの記述にも使用します.

説明

selectiondataは、アイテムソースからアイテムターゲットを取り出す際に、並べ替えの条件とクリアランス範囲を設定するために使用します。

コンポーネント

ShapeType

データタイプ： shapetype

使用するクリアランス範囲の形状を指定します.

- SHAPE_UNDEFINEDは、 何も選択されていないことを示します。
- BOXは、 アイテムターゲットの回りに他のアイテムターゲットが存在せず、クリアなボックス形状が存在する必要があることを示しています。
- CYLINDERは、アイテムターゲットの回りに他のアイテムターゲットが存在せず、クリアな円筒形状が存在する必要があることを示しています。
- SPHEREは、 アイテムターゲットの回りに他のアイテムターゲットが存在せず、クリアな球形状が存在する必要があることを示しています。

ConsiderType

データタイプ： aconsidertype

選択を行う際、キューに含まれるどのアイテムを対象とするかを示しています.

- ITEMS_TO_USEは、 このキューが使用対象としてマークしたアイテムだけを選択対象とすることを示しています。
- ITEMS_BYPASSは、 このキューがパスの対象としてマークしたアイテムだけを選択対象とすることを示しています。
- ITEMS_PICKEDは、 このキュー、またはライン上の以前のキューがピック済みとしてマークしたアイテムだけを選択対象とすることを示しています。
- ITEMS_PLACEDは、 このキュー、またはライン上の以前のキューがプレイス済みとしてマークしたアイテムだけを選択対象とすることを示しています。

アイテムの選択時に異なるタイプのマークを持つ複数のアイテムが対象となる場合、その対象タイプで bit-or 操作を実行できます（RAPID 関数 BitOr(<byte>,<byte>)。）

GeometricData

データタイプ： geodata

図形の寸法（x、y、z、半径）を定義するデータです.

- BOX　　形状は、 x、y、および z の値で定義します.
- CYLINDER　　形状は、 半径の値で定義し、高さはz 値で定義します.
- SPHERE　　形状は、 半径の値で定義します.

次のページに続く

© Copyright 2025 ABB. All rights reserved.

形状の座標システムの向きは、オフセットデータコンポーネントで定義します。デフォルトでは、形状の座標システムはワークオブジェクトやコンベヤフレームに合わせて配置されています。すべての形状起点は形状の中心に配置され、値はすべての平面上における正と負の距離です。これは、ボックスがx：10、y：15、z：20 として定義されている場合、そのボックスは x 方向に 20 mm、y 方向に 30 mm、z 方向に 40 mm のサイズになることを意味します。オフセットが使用されていない場合、範囲内のすべてのアイテムの 10 mm 前、10 mm 後ろ、15 mm 左、15 mm 右、20 mm 上、20 mm 下で、別のアイテムのチェックが実行されます.

`Offset`

データタイプ： `offsetdata`

オフセットは `OffsetRelation`（`offsetreltype`）**および** `OffsetPose`（`pose`）**から成ります。**

`OffsetRelation` **は、 2 つの異なるタイプのいずれかになります。**

- `FRAME_COORD_DIR` **は、** `OffsetPose` **の回転が ワークオブジェクトまたはコンベヤフレームの座標システムに相対していることを示しています。**
- `ITEM_COORD_DIR` **は、** `OffsetPose` **の回転が チェックするアイテムのアイテム座標システムに相対していることを示しています。**

`OffsetPose` **は、形状の 中心をアイテムポジションから離す場合、例えばアイテムのグリップポジションが実際にピックするオブジェクトの中心ではない場合に使用します。**

例

```
VAR selectiondata clear_rect:= [BOX,ITEMS_TO_USE,[22,15,5,0],
     [FRAME_COORD_DIR,[[0,7,0],[1,0,0,0]]]];
```

制限事項

方向は標準化する必要があります。平方根の合計は 1 でなければなりません.

$q1^2 + q2^2 + q3^2 + q4^2 = 1$

構造

```
<dataobject of selectiondata>
  <ShapeType of shapetype>
  <ConsiderType of considertype>
  <GeometricData of geodata>
    <x of num>
    <y of num>
    <z of num>
    <radius of num>
  <Offset of offsetdata>
    <OffsetRelation of offsetreltype>
    <OffsetPose of pose>
      <trans of pos>
        <x of num>
        <y of num>
        <z of num>
      <rot of orient>
        <q1 of num>
```

次のページに続く

© Copyright 2025 ABB. All rights reserved.

```
<q2 of num>
<q3 of num>
<q4 of num>
```

関連情報

下記の詳細は	次で説明されています
データタイプ pose	*Technical reference manual - RAPID Instructions, Functions and Data types*。
関数 BitOr	*Technical reference manual - RAPID Instructions, Functions and Data types*。
四元数について	*Technical reference manual - RAPID Overview*。
selectiondata の使用例	例：クリアランスゾーンに基づくアイテムの選択 ページ *437*。

© Copyright 2025 ABB. All rights reserved.

6.3.3 sortdataー並べ替えデータ

使用法

sortdataは、並べ替え条件の説明で使用します.

説明

sortdataは、アイテムソースからのアイテムターゲットで使用する並べ替え条件の設定で使用します。

コンポーネント

SortType

データタイプ： sorttype

使用する並べ替えのタイプです.

- UNSORT_TYPE は、 並べ替えが使用されないことを示しています。
- POS_Y_SORT_TYPE は、 ワークエリアの正のＹ方向から並べ替えを実行する必要があることを示します。
- NEG_Y_SORT_TYPE は、 ワークエリアの負のＹ方向から並べ替えを実行する必要があることを示します。

CheckBoundary

データタイプ： num

並べ替えのクリアランス間隔です（ミリ単位）。間隔は、並べ替え方向における次のアイテムとの最小間隔として定義します.

SortDirOffset

データタイプ： num

並べ替え方向におけるアイテムターゲットの後ろのオフセット間隔です。これは、他のアイテムターゲットを配置できないコリドーの内側限界の定義で使用します.

例

```
VAR sortdata y_sort:=[NEG_Y_SORT_TYPE ,78, 52];
```

構造

```
<dataobject of sortdata>
  <SortType of sorttype>
  <CheckBoundary of num>
  <SortDirOffset of num>
```


© Copyright 2025 ABB. All rights reserved.

6.4 RAPIDプログラム

6.4.1 RAPID プログラム

概要

概要

各ロボットにデフォルトのRAPIDプログラムがあり、このプログラムは通常のテキストエディターでジョブダイアログのロボット設定から編集できます。ジョブが始まると、ピッキングコントローラのPickMasterによってプログラムがダウンロードされます。プログラムには、プログラムの実行が始まるメインルーチンがあります。

> **注記**
>
> このプログラムは、ダウンロードの手順により、ロボットシステムで直接変更することはできません。

インストールには次のプログラムテンプレートファイルが含まれます。

テンプレート	カスタマイズの対象
`PMppa360.mod`	4軸FlexPicker IRB360
`PMppaDelta.mod`	5 軸の FlexPicker IRB 365 および IRB 390。
`PMppa6Axes.mod`	SCARAロボット、事例として、IRB 910 関節式アームタイプ IRB 460 および IRB 660 の 4 軸ロボット。 IRB 120 などのアーティキュレートアームの 6 軸ロボット。 IRB 14050 などの、関節式アームタイプの 7 軸ロボット。

プログラムの実行 - 一般

新しいジョブが開始されると、 Main ルーチンから、 PickMaster によってRAPIDプログラムがロードされ起動します。

デフォルトの RAPID プログラムは、どのサイクルでも次を実行します。

- ピックワークエリアでのピック。
- プレイスワークエリアでのプレイス。

ロボットのピックワークエリアが2つ以上ある場合、設定されたワークエリアインデックスが最小のエリアを使用します。ロボットのプレイスワークエリアが2つ以上ある場合、設定されたワークエリアインデックスが最小のエリアを使用します。RAPIDは、たとえば1つのプレイスでダブルピックするなど、別のシーケンスを実施するように変更できます。

プログラムの実行 - ワークエリア

RAPIDでは、ワークエリアは常にアイテムソースオブジェクトに関連づけられます。アイテムソースはキューとも呼ばれます。アイテムソースは、このワークエリアに関係するターゲットポジションをすべて保持します。ターゲットポジションはアイテムソースに連続的に受信され、その間、関連するフローハンドラーセンサーで検出されます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

6.4.1 RAPID プログラム
続き

プログラムの実行 - ターゲットポジション

ピックごとに、ピックアイテムソースからピックターゲットがフェッチされます。ターゲットポジションは、次にピックされるアイテムの場所になります。

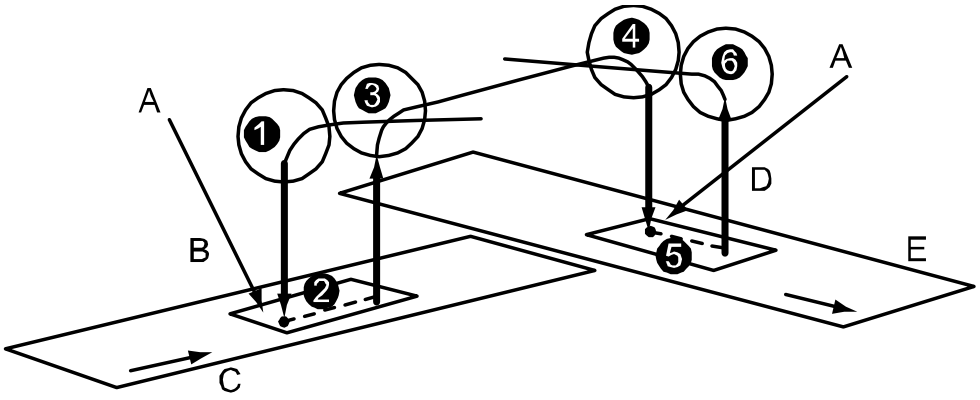
ピックごとに、プレイスアイテムソースからプレイスターゲットがフェッチされます。ターゲットポジションは、次にアイテムを置く空きプレイスポジションになります。

動作

RAPIDプログラムは、6つの異なる動きで設計されています.

6 軸ロボットの場合、以下の 2 つの中間ポイントを使用する必要があります。

- ポジション3とポジション4の間.
- ポジション6と次のループのポジション1の間.



xx0800000326

下記の6つの動きが含まれています.

	説明
1	ピックターゲット上のアプローチポジション. プレイスターゲット上の距離は、 **ワークエリアのプロパティ** ダイアログに表示されるピックエレベーション値（ツールの z のマイナス方向）です。ターゲットはコーナーパスタイプで、真空アクティベーション時はコーナーパスの中央より前の時間として計算されます。この時間は **ワークエリアのプロパティ** ダイアログに入力します。
2	これはピックターゲットです. ロボットTCPは、 **ワークエリアのプロパティ** ダイアログに入力されたピック時間に、コンベヤとの関係を基準に調整されます。TCPはピック時間にピックターゲットをフォローします。
3	ピックシーケンスの最後のポジションです. ピックターゲット上の距離は、アプローチポジションと同じ方法で計算されます. コーナーパスの中心に到達するまでコンベヤに追従します。したがって、アイテムソースがピックワークエリアバッファの次のアイテムのトラッキングを開始できるように、使用されたアイテムターゲットが認知されなくてはなりません。ターゲットはfineポイントにはできません.
4	プレイスターゲット上のアプローチポジション. プレイスターゲット上の距離は、 **ワークエリアのプロパティ** ダイアログに表示されるピックまたはプレイスのエレベーション値（ツールの z のマイナス方向）です。
5	これはプレイスターゲットです. ロボットのTCPは、プレイスの間、コンベヤに相対してコーディネートされます。開放イベントのタイミングは、1/2プレイス時間を過ぎる前の時間として計算されます。吸着オフのタイミングは、1/2プレイス時間を過ぎた時間として計算されます.

次のページに続く

© Copyright 2025 ABB. All rights reserved.

	説明
6	ピックシーケンスの最後のポジションです. このポジションは、TCP がコーナーパスの中心を過ぎるまで、あるいはfineポイントに到達するまで、コンベヤに追従します。したがって、アイテムソースがピックワークエリアバッファの次のアイテムのトラッキングを開始できるように、使用されたアイテムターゲットが認知されなくてはなりません。ターゲットは細かいポイントにはできません.

注記

動いているコンベヤの上でピックアンドプレイスサイクルを実行する際は、RAPIDプログラムポインターがあらかじめ起動し、ターゲットが使用されるときよりかなり前にターゲットを取り出します。ロボットがターゲットを使用するときには、ターゲットはすでに終了限界を過ぎている可能性があります。RAPID は約 100ms 前にプログラムポインターを動かします。調整された微細点では、ロボットがコンベヤの上でロックすると、微細点移動の開始時に「事前実行」がトリガーされます。PickTime が長い（例えば、50ms）場合は、ロボットがそこに物理的に移動するかなり前（50ms）に、次のターゲットがキューから取り出されます。コンベヤ速度が高速の 50ms である場合、ピックするターゲットはすでに終了限界を超えていることを意味することがあります。それでもロボットはターゲットを取り出そうと試みます。

プログラムモジュール

デフォルトのRAPID プログラムには3 つのプログラムモジュールがあります.

モジュール	説明
PMTWMAIN	メインプログラムの起動と実行シーケンスを行います。このモジュールを編集してカスタマイズしないでください.

システムモジュール

RobotWare オプション *PickMaster Ready* を備えた ABB ロボット コントローラには、ロードされたシステム モジュール *ppaBase* (暗号化) および *ppaUser* (オープン) が常に含まれます。

モジュール	説明
ppaBase	PickMasterとの通信に使用される変数、イベントルーチン、アイテムソースの作成、開始、削除に使用されるルーチンが含まれています.
ppaUser	パブリックデータタイプの宣言だけでなく、インデックスワークエリアのワークオブジェクトデータも含まれています。また、デフォルトのツールデータ、例えば`PickAct1`や`PickAct2`などの宣言も含まれています。

パブリックデータタイプ

概要

システムモジュール*ppaUser*には、*sourcedata*と*noncnvwobjdata*の2つのレコード定義が含まれています.

次のページに続く

© Copyright 2025 ABB. All rights reserved.

sourcedata

*sourcedata*は変数アレイ*ItmSrcData*で使用されます。このアレイには、各アイテムソースに関するデータが含まれています.

レコードは、他の目的のために新しいコンポーネントを追加することはできますが、その構造内の既存コンポーネントを1つでも変更したり削除することはできません.

名前	エイリアス	説明
Used	bool	アレイインデックスが使用されていることを示すフラグです.
ItemSource	itmsrc	アイテムソースへの記述子です.
SourceType	itmsrctype	ソースのタイプ PICK_TYPE、PLACE_TYPE または UNDEFINED_TYPE。
Ack	triggdata	アイテムターゲットを認知するためのトリガーデータです。
Nack	triggdata	アイテムターゲットを拒否するためのトリガーデータです。
SimAttach1	triggdata	シミュレーションで近くにあるアイテムをアクティベータ 1 に付加するためのトリガーデータです。
SimAttach2	triggdata	シミュレーションで近くにあるアイテムをアクティベータ 2 に付加するためのトリガーデータです。
SimDetach1	triggdata	シミュレーションでアクティベータ 1 により保持されているアイテムを分離するためのトリガーデータです。
SimDetach2	triggdata	シミュレーションでアクティベータ 2 により保持されているアイテムを分離するためのトリガーデータです。
VacuumAct1	triggdata	実ロボットでの吸着のためのトリガーデータです。
VacuumAct2	triggdata	実ロボットでの吸着のためのトリガーデータです。
VacuumRev1	triggdata	実ロボットでのブローのためのトリガーデータです。
VacuumRev2	triggdata	実ロボットでのブローのためのトリガーデータです。
VacuumOff1	triggdata	実ロボットでのリセットのためのトリガーデータです。
VacuumOff2	triggdata	実ロボットでのリセットのためのトリガーデータです。
Wobj	wobjdata	ソースのワークオブジェクトデータです
VacActDelay	num	吸着の遅延です
VacRevDelay	num	開放の遅延です
VacOffDelay	num	吸着オフの遅延です
TunePos	pos	ワークエリアのポジション調整です.
TrackPoint	stoppointdata	フォロータイムのデータです.
OffsZ	num	ピック / プレイスポジション上のオフセットポイントの高さです.

次のページに続く

© Copyright 2025 ABB. All rights reserved.

noncnvwobjdata

noncnvwobjdata がパーシステント変数 *NonCnvWOData*に使用されます。これが使われるのはインデックスワークエリアだけです。ワークオブジェクトデータはこのアレイに保存されます。このデータは次に、アイテムソースが作成された時に使われます。

レコードは、他の目的のために新しいコンポーネントを追加することはできますが、その構造内の既存コンポーネントを1つでも変更したり削除することはできません.

名前	エイリアス	説明
Used	bool	アレイインデックスが使用されていることを示すフラグです.
NonCnvWobjName	string	ワークエリアの名前です.
Wobj	wobjdata	保存されたワークオブジェクトデータです.

必ずパスを消去

パスの消去

移動する作業オブジェクトとの連携モーションで停止が発生した場合、再開前にロボットのパスが消去されます。消去されないと、連携モーションは保存されたパスで続行されますが、コンベヤ上のオブジェクトのポジションは、ロボットの到達範囲外になっている可能性があります。

無条件のパスの消去

`AlwaysClearPath`（常にブール）ルーチンは、入力パラメータ値がTRUEに設定されている場合、再開前にパスを無条件で消去します。

© Copyright 2025 ABB. All rights reserved.

6.4.2 変数

変数について

PickMaster ロボットコントローラには、多くの RAPID 変数が含まれています。これらの変数は、ppaBase と ppaUser の両方で宣言されます。これらの多くはカスタマイズされたプログラムでは使用できません。

ppaUser にあるパブリック変数

概要

ppaUser にある以下の変数が使用できます。

VAR sourcedata ItmSrcData{MaxNoSources}

これは、すべてのワークエリアについての情報を保管するアレイ変数です。ワークエリアの構成で提供されるインデックスは、ItmSrcDataアレイのインデックスです。

PERS noncnvwobjdata NonCnvWOData{MaxNoSources}:=[[…

このアレイの変数はインデックスワークエリアのワークオブジェクトフレームを保存します。ワークオブジェクトの補正を見つけるカギは名前にあり、これはワークエリア設定の名前と同じにする必要があります。

TASK PERS tooldata PickAct1:=[…]

このツールデータはピックとプレイスの操作に用いられます。

> **注記**
>
> ツールの方向は、キューから取得されるアイテムの方向と一致している必要があります。キューから取得されるアイテムのターゲットポジションは、定義された方向からX軸を中心に180度回転されます。

天井取り付け型 IRB360 と水平コンベヤに置かれたアイテムを使用する標準インストールでは、tool0 と同様にツールの Z 軸が先端からコンベヤに向いています。

ppaBase にあるパブリック変数

ppaBase にある以下の変数が使用できます。

TASK PERS num Vtcp:=1000

PickMasterから実行される速度調整は、この変数で設定します.

TASK PERS speeddata MaxSpeed:=[…]

これは、最高速の移動で使用する速度です.

TASK PERS speeddata LowSpeed:=[…]

これは、低速の移動で使用する速度です.

TASK PERS speeddata VeryLowSpeed:=[…]

これは、最低速の移動で使用する速度です.

次のページに続く

© Copyright 2025 ABB. All rights reserved.

PickMasterテンプレートプログラムにあるパブリック変数

PickMasterテンプレートプログラムにある使用可能なパブリック変数は次の通りです.

VAR num PickWorkArea{X}:=0

ロボットがどのワークエリアからアイテムをピックするかを指定するには、PickWorkArea アレイを使用します。ピックワークエリアは、選択インデックスを基準に並べられています。

`PickWorkArea{1}` はワークエリア選択インデックスが最も低いです。

`PickWorkArea{2}` はワークエリア選択インデックスが 2 番目に低いです。

VAR num PlaceWorkArea{X}:=0

ロボットがどのワークエリアにアイテムをプレイスするかを指定するには、PlaceWorkArea アレイを使用します。プレイスワークエリアは、選択インデックスを基準に並べられています。

`PlaceWorkArea{1}` はワークエリア選択インデックスが最も低いです。

`PlaceWorkArea{2}` はワークエリア選択インデックスが 2 番目に低いです。

VAR num OtherWorkArea{X}:=0

ユーザー定義の目的のためにロボットがどのワークエリアに移動するかを指定するには、OtherWorkArea アレイを使用します。他のワークエリアは、選択インデックスを基準に並べられています。

`OtherWorkArea{1}` はワークエリア選択インデックスが最も低いです。

`OtherWorkArea{2}` はワークエリア選択インデックスが 2 番目に低いです。

VAR itmtgt PickTarget:=[...]

この変数は、ピックアイテムソースからピックターゲットを取得するために使用されます.

VAR itmtgt PlaceTarget:=[...]

この変数は、ピックアイテムソースからプレイスターゲットを取得するために使用されます.

TASK PERS wobjdata WObjPick:=[…]

ワークエリアの wobjdata を保持します。モーション命令は `PERS` タイプとして wobjdata を持っている必要があるため、情報はピックルーチンで `ItmSrcData` から `WObjPick` に移動されます。

TASK PERS wobjdata WObjPlace:=[…]

ワークエリアの `wobjdata` を保持します。モーション命令は `PERS` タイプとして `wobjdata` を持っている必要があるため、情報はプレイスルーチンで `ItmSrcData` から `WObjPlace` に移動されます。

TASK PERS robtarget SafePos:=[…]

これは、ロボットの開始ポジションを定義します。robtargetは使用状況に応じて変更してください.

次のページに続く

© Copyright 2025 ABB. All rights reserved.

6.4.2 変数
続き

TASK PERS robtarget IntPosPickX:=[…]

これは、すべてのピックワークエリアロボットに対して中間ポジションを定義します。robtargetは各ワークエリアに応じて変更してください.

TASK PERS robtarget IntPosPlaceX:=[…]

これは、すべてのプレイスワークエリアロボットに対して中間ポジションを定義します。robtargetは各ワークエリアに応じて変更してください.

TASK PERS loaddata ItemLoad:=[…]

これはピックアンドプレイス操作で使用するロードデータ（loaddata）です。このloaddataはピックされるアイテムに適合するように編集します。異なるアイテムタイプが使用される場合には、各タイプに対して1つのロードデータを宣言します。ロボットが最高のパフォーマンスを実現できるように、正しいデータを使用することが重要です.

デフォルトのロードデータは、tooldataload0 と同じです。

© Copyright 2025 ABB. All rights reserved.

6.4.3 ルーチン

ルーチンについて

PickMaster Rapid モジュールには、多くのルーチンが含まれています。エンドユーザーにとって非常に便利なものもあれば、PickMasterプログラムが内部使用するだけのものもあります.

PickMasterテンプレートプログラムにあるパブリックルーチン

PickMasterテンプレートプログラムにある使用可能なパブリックルーチンは次の通りです.

PROC main()

これは、 Rapid プログラムの開始ルーチンです。プログラムは、常にこのルーチンから開始されます.

PROC InitSafeStop()

このルーチンは、SafeStop トラップを開始します。これは、PickMaster プロジェクトが一時停止または停止された場合にロボットの停止を正しく行うため、プログラム実行の初めに実行される必要があります.

PROC InitTriggs()

このルーチンは、プロジェクト開始時に、各ワークエリアインデックスに対して、吸着、開放、オフのトリガイベントを設定します。詳細は`SetTriggs`を参照してください.

PROC InitPickTune()

このルーチンは、PickTune トラップを開始します。これは、ワークエリアの調整を行うため、プロジェクトの開始時に実行されなければなりません.

PROC SetTriggs(num Index)

これは、吸着、オフのトリガーイベントを設定するルーチンです。デフォルトプログラムは、I/O グループ `goVacBlow1` の 1 つの吸着イジェクタに対するイベントのみを設定します。複数の吸着イジェクタが使用される場合、新しい吸着イジェクタ I/O グループを正しいワークエリアに設定する必要があります。また、デフォルトルーチンを編集して、各ワークエリアに対して正しい吸着イジェクタが取得されるようにしなくてはなりません。

PROC SetSimulatedTriggs(num Index)

PROC SetTriggs(num Index) のオフライン バージョン。

ツールに 1 ~ 2 個のアクティベータがある場合は、変更する必要はありません。

PROC SetSimulatedDummyTriggs(num Index)

シミュレーションされたモードに関連しない RAPID コードで使用されるすべてのトリガー イベントを設定します。

ツールに 1 ~ 4 個のアクティベータがある場合は、変更する必要はありません。

PROC SetDummyTriggs(num Index)

オンライン モードに関連しない、RAPID コードで使用されるすべてのトリガー イベントを設定します。

ツールに 1 ~ 4 個のアクティベータがある場合は、変更する必要はありません。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

6.4.3 ルーチン
続き

PROC InitSpeed()

このルーチンは、プログラムで使用されるロボットの速度を設定します。このルーチンでは、命令VelSetが 実行されます。これはロボットの最高許容速度を設定します。6軸ロボットを使用する場合、この限界速度を調整してモーションエラーを回避できます.

PROC PickPlace()

このルーチンはアイテムキューを開始し、最終設定を起動します。ピックアンドプレイスシーケンスは、このルーチンから呼び出されます。このルーチンには何も変更を加えないようにしてください。

このルーチンは、 ピックアンドプレイス操作の開始時に呼び出されます.

PROC SafeStop()

プロジェクトが停止または一時停止されると、このルーチンがSafeStopTrapルーチン、 あるいはPickPlaceルーチンのいずれかから 呼び出されます。安全なポジションへのスローモーションは、このルーチンから呼び出されます.

PROC GotoRestartPos()

安全ポジションへのスローモーションが実行され、全てのアイテムソースが拒否されます。これは実行割り込みされたことをソースに伝えるために必ず実行する必要があります.

PROC Home()

これは、ロボットを安全ポジションに移動するサービスルーチンです.

PROC WashDown()

これは、ウォッシュダウンサービスルーチンです.

PROC TestCycle()

これは、テストサービスルーチンです.

PROC Homepos()

これは、ロボットを同期化ポジションに移動するサービスルーチンです.

PROC EnumerateWorkAreas()

ピック、プレイス、およびその他についてワークエリアのアレイを設定します。

PROC PickPlaceSeq()

これはアプリケーションのシーケンスを指定するルーチンです。ロボットが、別のキューからピックアンドプレイスを実行する方法のロジックは、このルーチンに実装されます.

このルーチンは、ループごとに呼び出されます。これは、 [PickMaster production（PickMaster 生産）] タブに表示されるピックレート統計では、1「Pick」としてカウントされます.

PROC Pick(num Index)

1 ピックを実行します。指定された Index は、アイテムがピックされるワークエリアを定義します.

次のページに続く

© Copyright 2025 ABB. All rights reserved.

PROC Place(num Index)

1 プレイスを実行します。指定された Index は、アイテムがプレイスされるワークエリアを定義します.

TRAP SafeStopTrap

停止 I/O 信号をとらえるためのトラップルーチンです。PickPlace ルーチンから SafeStop が呼び出される前に停止 I/O 信号が設定される場合に実行されます。

TRAP PickTuneTrap

PickMaster から調整された値を、対応する変数に添付するトラップルーチンです.

---

ppaBase モジュールにある非表示ルーチン

概要

ppaBase モジュールの非表示ルーチンは次のとおりです。

PROC ResetEvent()

一部の変数をリセットします。このルーチンは、RESET システム イベントシェルフのみで実行されます。

PROC PowerOnEvent()

一部の変数をリセットします。このルーチンは、POWER_ON システムイベントシェルフのみで実行されます。

PROC StopEvent()

停止が発生し、ロボットが追従動作中の場合、ロボットパスをクリアします。このルーチンは、STOP システムイベントシェルフのみで実行されます。

PROC RestartEvent()

このルーチンは、RESTART システムイベント シェルフのみで実行されます。ロボットが追従動作中の場合、このルーチンは発生エラー PPA_RESTART のエラーハンドラを持つレベルからプログラムを強制的に再起動します。

PROC NewSource()

新しいアイテムソースを作成し、ItmSrcData 変数を起動します。PickMaster は、プロジェクトの開始時に各ワークエリアについてこのルーチンを呼び出します。

PROC ClearAll()

このルーチンは重要な変数をすべてリセットし、アイテムソースをすべて削除します。このルーチンはプロジェクトの停止時に呼び出されます.

PROC PickRateInit()

このルーチンはピックレートの計算を開始します.

PROC PickRateReset()

このルーチンはピックレートの計算をリセットします.

PROC CheckAx4Rev ()

このルーチンは、IRB340の4番目の軸をリセットする必要があるかどうかチェックします.

次のページに続く

© Copyright 2025 ABB. All rights reserved.

6.4.3 ルーチン
続き

PROC ResetAx4 (VAR mecunit MechUnit)

このルーチンは4番目の軸をリセットします.

PROC NotifyClearAll ()

このルーチンは、PickMasterに`ClearAll`が 実行されたことを通知します.

PROC NotifySafeStop ()

このルーチンは、PickMasterに`SafeStop` が 実行されたことを通知します.

PROC NotifyRunning ()

このルーチンは、PickMasterにプロセスが実行されていることを通知します.

PROC NotifyWaitForExe ()

このルーチンは、PickMasterにRAPIDプログラムが新しい命令を待っていることを通知します.

PROC WaitForExeOrder ()

PickMaster が次の実行命令を出すのを RAPID プログラムが待機する命令。 命令が与えられない場合、RAPID の実行は待機し、この命令でアイドル状態になります。

PROC IncrPicks ()

このルーチンはピックレートの計算を増分します.

PROC ppaDropWobj(PERS wobjdata Wobj)

このルーチンは、`DropWobj` 命令をカプセル化します。詳細については、*Application manual - Conveyor tracking*を参照してください。

PROC WalkTheData()

このルーチンは、アレイ変数*ItmSrcData* および*NonCnvWOData*のコンテンツをトラッキングするために使用され、エラーを検出する場合に役立ちます。これは、コントローラのシステムディレクトリにTheData.logファイルを書き込みます.

TRAP PickRateTrap

これはロボットの正しいピックレートを計算するトラップルーチンです.

PROC AlwaysClearPath(bool always)

詳細は、を参照してください。

必ずパスを消去 ページ 421

© Copyright 2025 ABB. All rights reserved.

6.5 プログラムの例

6.5.1 例：1つのピックワークエリアと2つのプレイスワークエリアの組み合わせ

例の説明

この例では、1つのピックワークエリアと2つの異なるタイプのアイテムが使用されています。アイテムは、そのタイプによって、2 つのワークエリアに配置されます.

1 アイテムをピックワークエリアからピックします
2 アイテムのタイプを定義します
3 ワークエリア外でのプレイス

コードの例

```
PROC PickPlaceSeq()
  Pick PickWorkArea{1};
  IF PickTarget.Type = 1 THEN
    Place PlaceWorkArea{1};
  ELSEIF PickTarget.Type = 2 THEN
    Place PlaceWorkArea{2};
  ENDIF
ENDPROC
```


© Copyright 2025 ABB. All rights reserved.

6.5.2 例：2つのピックワークエリアと1つのプレイスワークエリアの組み合わせ

例の説明

この例では、プレイスワークエリアがマスターで、パターンの塗りつぶしが必要なアイテムを判断し、どちらのピックワークエリアからピックするかを定義します.

1 次のアイテムターゲットタイプを確認します
2 どちらのワークエリアからピックするかを決定します
3 アイテムをピックワークエリアからピックします
4 ワークエリア外でのプレイス

コードの例

```
PROC PickPlaceSeq()
  VAR num PlaceType:=0;

  NextItmTgtType
    ItmSrcData{PlaceWorkArea{1}}.ItemSource,
    PlaceType;
  IF PlaceType = 1 THEN
    Pick PickWorkArea{1};
  ELSEIF PlaceType = 2 THEN
    Pick PickWorkArea{2};
  ENDIF
  Place PlaceWorkArea{1};
ENDPROC
```


© Copyright 2025 ABB. All rights reserved.

6.5.3 例：1つのピックと1つのプレイスワークエリアの組み合わせ

例の説明

この例では、プレイスワークエリアがマスターで、パターンの塗りつぶしが必要なアイテムを判断し、どちらのアイテムをピックするかを定義します.

1 次のアイテムターゲットタイプを確認します
2 アイテムをピックワークエリアからピックします
3 ワークエリア外でのプレイス

> **注記**
>
> **Available Work Areas**[使用可能な作業領域]設定で**Use Start/Stop**[開始/停止を使用]を使用することをお勧めします。

コードの例

```
PROC Pick(num Index)
  VAR num PickType:=0;
  VAR num PlaceType:=0;

  WObjPick:=ItmSrcData{Index}.Wobj;
  NextItmTgtType
    ItmSrcData{PlaceWorkArea{1}}.ItemSource,PlaceType;
  TEST PlaceType
  CASE 4:
    PickType:=1;
  CASE 5:
    PickType:=2;
  CASE 6:
    PickType:=3;
  ENDTEST
  GetItmTgt ItmSrcData{Index}.ItemSource, PickTarget
       \ItemType:=PickType;
  TriggL \Conc, RelTool(PickTarget.RobTgt, 0, 0,
       -ItmSrcData{Index}.OffsZ), MaxSpeed,
       ItmSrcData{Index}.VacuumAct1, z20, PickAct1 \WObj:=WObjPick;
  MoveL \Conc, PickTarget.RobTgt, LowSpeed, z5 \Inpos:=
       ItmSrcData{Index}.TrackPoint, PickAct1 \WObj:=WObjPick;
  GripLoad ItemLoad;
  TriggL RelTool(PickTarget.RobTgt, 0, 0, -ItmSrcData{Index}.OffsZ),
       LowSpeed, ItmSrcData{Index}.Ack, z20, PickAct1
       \WObj:=WObjPick;
ENDPROC
```


© Copyright 2025 ABB. All rights reserved.

6.5.4 例：ダブルピック・シングルプレイス

例の説明

ロボットは、インフィーダーコンベヤで2つのアイテムを1つずつピックし、両方をアウトフィードコンベヤに載せます。この操作には、吸着イジェクタを2つ備えたピッキングツールが必要です。

実装

開始点として、ロボット1台で簡単な作業設定を策定します。

RAPIDプログラムを修正する必要があります。RAPIDプログラムを編集するには、**レシピ設定**に移動してロボットを選択し、ドロップダウンメニューを表示して、**RAPIDプログラム**、**編集...**の順に選択します。

PickPlaceSeqルーチンは2つのピックルーチンコールを実行して第1、第2ピックを処理します。次に 1 つのプレイスルーチンを実行し、ピックしたアイテムの同時プレイシングを処理します。次のコード例を参照してください。

```
!
! Procedure PickPlaceSeq
!
! The Pick and Place sequence.
! Edit this routine to specify how the robot shall execute the
    movements.
!

!***********************************************************
PROC PickPlaceSeq()
  Pick PickWorkArea{1}, 1;
  Pick PickWorkArea{1}, 2;
  Place PlaceWorkArea{1};
ENDPROC
```

ピックルーチンについては、以下のコード例を参照してください。第2ピックのためのPickAct2とVacuumAct2の使用についてご留意ください。

```
!***********************************************************
!
! Procedure Pick
!
! Executes a pick movement.
! Edit this routine to modify how the robot shall
! execute the pick movements.
! Needs to be changed if more than one activator is used.
!
!***********************************************************
PROC Pick(num Index, num pickNo)
  IF Index > 0 THEN
    WObjPick:=ItmSrcData{Index}.Wobj;
    GetItmTgt ItmSrcData{Index}.ItemSource,PickTarget;
    IF pickNo = 1 THEN
      TriggL\Conc,RelTool(PickTarget.RobTgt,0,0,
      -ItmSrcData{Index}.OffsZ),
```

次のページに続く

© Copyright 2025 ABB. All rights reserved.

```
      MaxSpeed,ItmSrcData{Index}.VacuumAct1,z20,
      PickAct1\WObj:=WObjPick;
      TriggL\Conc,PickTarget.RobTgt,LowSpeed,ItmSrcData{Index}.SimAttach1,
      z5\Inpos:=ItmSrcData{Index}.TrackPoint,
      PickAct1\WObj:=WObjPick;
      GripLoad ItemLoad;
      TriggL
      RelTool(PickTarget.RobTgt,0,0,-ItmSrcData{Index}.OffsZ),
      LowSpeed,ItmSrcData{Index}.Ack,z20,PickAct1\WObj:=WObjPick;
    ELSEIF pickNo = 2 THEN
      TriggL\Conc,RelTool(PickTarget.RobTgt,0,0,-ItmSrcData{Index}.OffsZ),
      MaxSpeed,ItmSrcData{Index}.VacuumAct2,
      z20,PickAct2\WObj:=WObjPick;
      TriggL\Conc,PickTarget.RobTgt,LowSpeed,ItmSrcData{Index}.SimAttach2,
      z5\Inpos:=ItmSrcData{Index}.TrackPoint,
      PickAct2\WObj:=WObjPick;
      GripLoad ItemLoad;
      TriggL
      RelTool(PickTarget.RobTgt,0,0,-ItmSrcData{Index}.OffsZ),
      LowSpeed,ItmSrcData{Index}.Ack,z20,
      PickAct2\WObj:=WObjPick;
    ENDIF
  ELSE
    ErrWrite "Missing item distribution", "Cannot pick because no
         item distribution contains current work area."
    \RL2:="Please check configuration";
    SafeStop;
  ENDIF
ENDPROC
```

最初のピックではツールデータPickAct1が使用されます。ツールデータPickAct2は、第2ピックで使用されます。PickAct1とPickAct2（ppaUser.sysモジュールで定義）を更新：制御された真空イジェクタの中心でツール中心点を定義します。重量と重心も更新します。RAPIDプログラムの更新を保存し、エディタを終了して更新を適用します。

プレイスルーチンについては、以下の例を参照してください。両方の保持アイテムの同時プレイシングのためのVacuumOff1とVacuumOff2にご留意ください。

```
!*********************************************************
!
! Procedure Place
!
! Executes a place movement.
! Edit this routine to modify how the robot shall
! execute the place movements.
! Needs to be changed if more than one activator is used.
!
!*********************************************************
PROC Place(num Index)
  IF Index > 0 THEN
    WObjPlace:=ItmSrcData{Index}.Wobj;
    GetItmTgt ItmSrcData{Index}.ItemSource,PlaceTarget;
```

次のページに続く

© Copyright 2025 ABB. All rights reserved.

6.5.4 例：ダブルピック・シングルプレイス
続き

```
      MoveL\Conc,RelTool(PlaceTarget.RobTgt,0,0,-ItmSrcData{Index}.OffsZ),
      MaxSpeed,z20,PlaceAll\WObj:=WObjPlace;
      TriggL\Conc,PlaceTarget.RobTgt,LowSpeed,ItmSrcData{Index}.VacuumRev1\T2:=
      ItmSrcData{Index}.VacuumOff1\T3:=ItmSrcData{Index}.VacuumOff2\T4:=
      ItmSrcData{Index}.VacuumRev2\T5:=ItmSrcData{Index}.SimDetach1\T6:=
      ItmSrcData{Index}.SimDetach2,z5\Inpos:=
      ItmSrcData{Index}.TrackPoint,PlaceAll\WObj:=WObjPlace;
      GripLoad load0;
      TriggL RelTool(PlaceTarget.RobTgt,0,0,-ItmSrcData{Index}.OffsZ),
      LowSpeed,ItmSrcData{Index}.Ack,z20,PlaceAll\WObj:=WObjPlace;
    ELSE
      ErrWrite "Missing item distribution", "Cannot place because no
           item distribution contains current work area."
      \RL2:="Please check configuration";
      SafeStop;
    ENDIF
  ENDPROC
```

プレイスではツールデータPlaceAll（ppaUser.sys**モジュールで定義）が使用されます。**PlaceAll**を更新：制御された真空イジェクタの中心でツール中心点を定義します。重量と重心も更新します。RAPIDプログラムの更新を保存し、エディタを終了して更新を適用します。**

> **注記**
>
> **同じ方法を使用して、アクティベータが3つ以上あるツールを設定します。ただし、いくつかの追加ステップが必要となります。例えば、アクティベータが3 ~ 4個あるツールを使用するには、以下の追加ステップが必要となります。**
>
> 1 **コントローラオプションとして2つのI/Oボードを選択します。あるいは、追加的な信号**goVacBlow3**、**goVacBlow4**、**doVacuum3**、**doVacuum4**、**doBlow3**、および** doBlow4**を作成します。**goVacBlowX**の第1ビットは**doVacuumX**信号とオーバーラップするようにします。**goVacBlowX**の第2ビットは**doBlowX**信号とオーバーラップするようにします。**
> 2 SetTriggs**ルーチンを更新します。**TriggEquip**イベントの** VacuumAct3**、**VacuumOff3**、**VacuumAct4**、および** VacuumOff4** **のライン上のコメントを削除することにより、これらのイベントを有効にします。**

© Copyright 2025 ABB. All rights reserved.

6.5.5 例：事前定義パターンをインデックスワークエリアにプレイス

例の説明

この例の場合、定義済みパターンをインデックスワークエリアに置きます。RAPIDからポジションジェネレータ信号がトリガーされます。

新しい4つの信号を定義する必要があります.

1. RAPID から設定するポジションジェネレータ信号 doSIMPosGen。
2. コントローラからコンピュータへのイベントを生成するポジションジェネレータ信号 diSIMPosGen。
3. コンピュータシステムに対して、事前定義ポジションを送るよう指示するトリガー信号 doSIMTrig。
4. ポジションが送られたことをシステムに知らせるストロボ信号diSIMStrobe。

これらの信号は PPASIM ボードで定義できます。例：:

```
-Name "doSIMPosGen" -SignalType "DO" -Unit "PPASIM" -UnitMap "6"
     -Access "ALL"
-Name "doSIMTrig" -SignalType "DO" -Unit "PPASIM" -UnitMap "7"
     -Access "ALL"
-Name "diSIMPosGen" -SignalType "DI" -Unit "PPASIM" -UnitMap "6"
     -Access "ALL"
-Name "diSIMStrobe" -SignalType "DI" -Unit "PPASIM" -UnitMap "7"
     -Access "ALL"
```

トリガ/ストロボ信号と、ポジションジェネレータ信号のクロスコネクションを行います。例：:

```
EIO_CROSS
  -Res "diSIMPosGen" -Act1 "doSIMPosGen"
  -Res "diSIMStrobe" -Act1 "doSIMTrig"
```

RAPID コードでプレイスキューのコントロールを実行します。キューが空の場合（パターン内のすべてのポジションが使用された状態）、（RAPID コードで）信号 doSIMPosGen を High に設定します（RAPID コードで）。この信号は diSIMPosGen にクロスコネクションされ、コントローラからコンピュータへ新しいパターンをコントローラに送信するようリクエストするイベントが送られます。トリガー/ストロボ信号もクロスコネクションされ、diSIMStrobe を使用してシステムがストロボされます。

コードの例

```
PROC Place(num Index)
  VAR bool flagplace:=TRUE;

  WObjPlace:=ItmSrcData{Index}.Wobj;
  flagplace:=TRUE;

  WHILE flagplace=TRUE DO
    GetItmTgt ItmSrcData{Index}.ItemSource,
         PlaceTarget\MaxTime:=1\TimeFlag:=flagplace;
    IF flagplace=TRUE THEN
      PulseDO\PLength:=0.2,doSIMPosGen;
    ENDIF
```

次のページに続く

© Copyright 2025 ABB. All rights reserved.

6.5.5 例：事前定義パターンをインデックスワークエリアにプレイス
続き

```
  ENDWHILE
  MoveL\Conc, RelTool(PlaceTarget.RobTgt, 0, 0,
       ItmSrcData{Index}.OffsZ), MaxSpeed, z20,
       PickAct1\WObj:=WObjPlace;
  TriggL\Conc, PlaceTarget.RobTgt, LowSpeed,
       ItmSrcData{Index}.VacuumRev1
       \T2:=ItmSrcData{Index}.VacuumOff1, z5
       \Inpos:=ItmSrcData{Index}.TrackPoint,
       PickAct1\WObj:=WObjPlace;
  GripLoad load0;
  TriggL RelTool(PlaceTarget.RobTgt, 0, 0, ItmSrcData{Index}.OffsZ),
       LowSpeed, ItmSrcData{Index}.Ack, z20,
       PickAct1\WObj:=WObjPlace;
ENDPROC
```


© Copyright 2025 ABB. All rights reserved.

6.5.6 例：クリアランスゾーンに基づくアイテムの選択

例の説明

本例では、指定エリア内に他のアイテムターゲットがあるかどうかに基づいて、コンベヤベルトのアイテムを選択します。この方法はグリッパーが周囲のオブジェクトに触れることなく、アイテムをつかむ場合に役立ちます.

選択アルゴリズムは、x 方向の終了限界に最も近く、選択形状にロックオブジェクトを持たないオブジェクトを選択します.

`GetItmTgt` 命令のパラメータとして x 方向のチェック限界を使用することで、最初のアイテムがシャッフルされる開始ポイントを定義できます。命令は、チェック限界とエンター限界の間にある最初のオブジェクトを取り出そうと試みます。これにより、選択アルゴリズムは最も近いオブジェクトをチェックする際、チェック限界と終了限界の間にあるすべてのオブジェクトを対象とします。したがって、チェック限界と終了限界の間の距離は、最低でも最も大きなアイテムの直径になります。

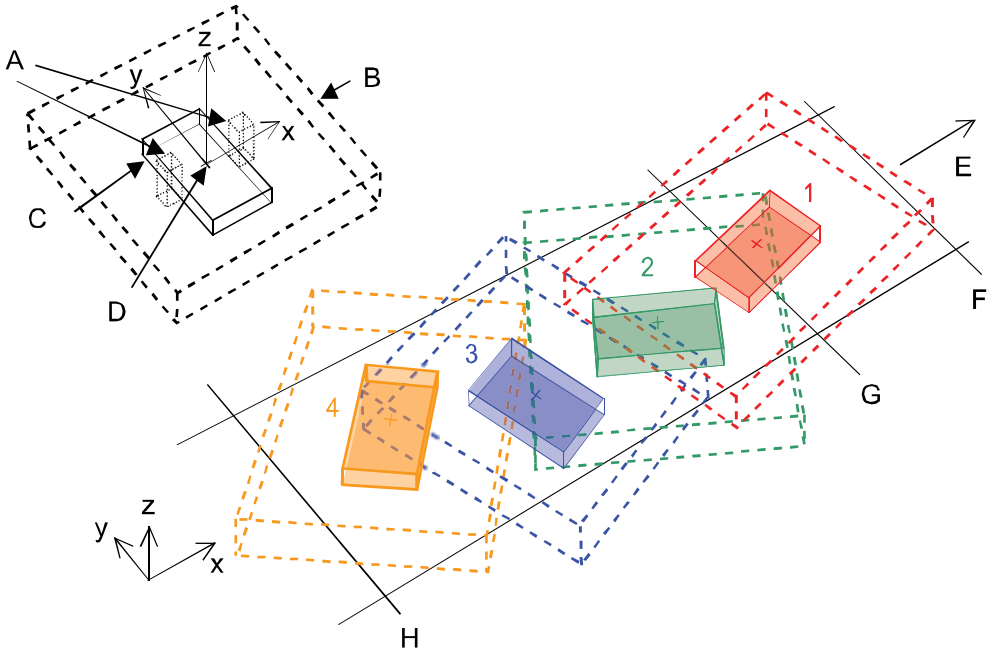
次の図は、アイテムがポジションと向きに基づいてどのように選択されるかを表しています。まずロボットは、アイテム4に続いてアイテム3をピックします。他の2つのアイテムはピックされません.

- アイテム1はチェック限界を超えているため、またアイテム2は選択形状内にあるため、それぞれピックできません.
- アイテム1と3がアイテム2の選択形状内にあるため、アイテム2はピックできません.
- アイテム 3 は、アイテム 4 が選択範囲内にあるためピックできません.
- アイテム 4 は選択範囲内に他のアイテムがないためピックできます.
- アイテム 3 は、アイテム 4 が存在しなくなるとピックされます.

次のページに続く

© Copyright 2025 ABB. All rights reserved.

6.5.6 例：クリアランスゾーンに基づくアイテムの選択
続き



xx0800000323

A	グリッパー
B	選択形状
C	Item（項目）
D	アイテムターゲットポジション
E	製品フロー方向
F	終了
G	境界チェック
H	エンター

*selectiondata－選択データ ページ 413*を参照してください。

コードの例

```
PROC Pick(num Index)
  VAR selectiondata sel_data;
  VAR robtarget draw_target;
  VAR num check_limit;

  sel_data.ShapeType:=BOX;
  sel_data.ConsiderType:=BitOr(ITEMS_TO_USE,ITEMS_BYPASS);
  sel_data.GeometricData.x:=60;
  sel_data.GeometricData.y:=70;
  sel_data.GeometricData.z:=10;sel_data.GeometricData.radius:=0;
  sel_data.Offset.OffsetRelation:=ITEM_COORD_DIR;
  sel_data.Offset.OffsetPose.trans.x:=0;
  sel_data.Offset.OffsetPose.trans.y:=0;
  sel_data.Offset.OffsetPose.trans.z:=0;
  sel_data.Offset.OffsetPose.rot.q1:=1;
  sel_data.Offset.OffsetPose.rot.q2:=0;
```

次のページに続く

© Copyright 2025 ABB. All rights reserved.

```
  sel_data.Offset.OffsetPose.rot.q3:=0;
  sel_data.Offset.OffsetPose.rot.q4:=0;
  check_limit:=150;

  WObjPick:=ItmSrcData{Index}.Wobj;
  GetItmTgt ItmSrcData{Index}.ItemSource,PickTarget
       \Limit:=check_limit\Selection:=sel_data;
  TriggL \Conc, RelTool(PickTarget.RobTgt, 0, 0,
       -ItmSrcData{Index}.OffsZ), MaxSpeed,
       ItmSrcData{Index}.VacuumAct1, z20, PickAct1\WObj:=WObjPick;
  MoveL \Conc, PickTarget.RobTgt, LowSpeed, z5 \Inpos:=
       ItmSrcData{Index}.TrackPoint, PickAct1\WObj:=WObjPick;
  GripLoad ItemLoad;
  TriggL RelTool(PickTarget.RobTgt, 0, 0, -ItmSrcData{Index}.OffsZ),
       LowSpeed, ItmSrcData{Index}.Ack, z20,
       PickAct1\WObj:=WObjPick;
ENDPROC
```


© Copyright 2025 ABB. All rights reserved.

6.5.7 例：負の Y 方向の並べ替え

6.5.7.1 リニアコンベアの負の y 方向の並べ替え

例の説明

本例では、周囲のオブジェクトに触れることなくコンベアベルトからアイテムをシャッフルします。シャッフル動作は、水平面でコンベヤの右側に垂直方向に実行され、マニピュレータのモーションはコンベヤのモーションに合わせて実行されます.

並べ替えアルゴリズムは、 x 方向の終了限界に最も近く、選択形状にロックオブジェクトを持たないアイテムを選択します.

選択形状は長いボックスとして定義されます。形状の x 値はコリドー幅の定義に使用されます。 y 値はコンベヤベルト幅の1/2以上でなくてはなりません。 z 値はすべてのアイテム間の高さにおける最も大きな差異よりも大きくなくてはなりません.

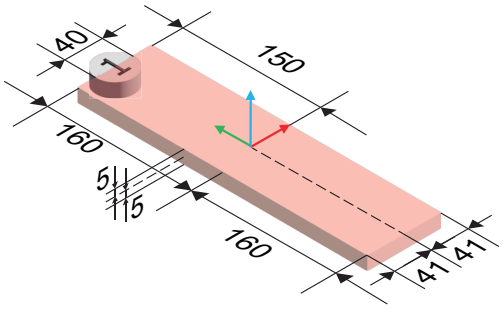
`OffsetData` の Y 値を形状のマイナス Y 値に設定すると、選択ボックスは右に取り出されます。

この結果、アイテムがシャッフルされる前に、すべてのアイテムの右側にクリアなコリドーが存在しなくてはなりません.

アルゴリズムは、生産フローの上下方向で、その他のアイテムの有無をチェックします.

`GetItmTgt` 命令のパラメータとして `x` 方向のチェック限界を使用することで、最初のアイテムがシャッフルされる開始ポイントを定義できます。命令は、チェック限界とエンター限界の間にある最初のアイテムをシャッフルしようと試みます。これにより、選択アルゴリズムは最も近いアイテムをチェックする際、チェック限界と終了限界の間にあるすべてのアイテムを対象とします。したがって、チェック限界と終了限界の間の距離は、最低でも最も大きなアイテムの直径にな ります。

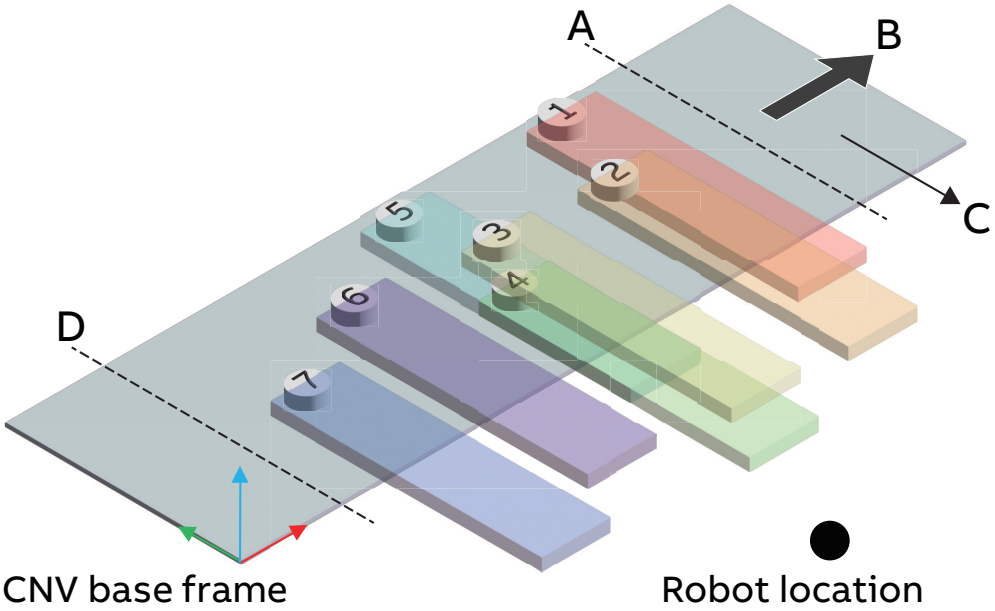
下の図では、すべてのアイテムがコンベア ベルトの右側にシャッフルされます。各アイテムにはクリア ゾーン、つまり*ShapeType*の形状が必要であるため、アイテムは、以下の図で番号が付けられているように 1 から 7 の順序でシャッフルされます。負の y 方向の並べ替えとして設定した場合のピッキング順序は、2->1->4->3->5->6->7になります。



xx2400000740

次のページに続く

© Copyright 2025 ABB. All rights reserved.



xx2400000737

A	終了
B	製品フロー方向
C	並べ替え方向
D	エンター

コードの例

```
PROC Pick(num Index)
  VAR selectiondata y_sort;
  VAR robtarget draw_target;
  VAR num check_limit;

  y_sort.ShapeType:=BOX;
  y_sort.ConsiderType:=BitOr(ITEMS_TO_USE,ITEMS_BYPASS);
  y_sort.GeometricData.x:=41;
  y_sort.GeometricData.y:=160;
  y_sort.GeometricData.z:=5;
  y_sort.GeometricData.radius:=0;
  y_sort.Offset.OffsetRelation:=FRAME_COORD_DIR;
  y_sort.Offset.OffsetPose.trans.x:=0;
  y_sort.Offset.OffsetPose.trans.y:=-150;
  y_sort.Offset.OffsetPose.trans.z:=0;
  y_sort.Offset.OffsetPose.rot.q1:=1;
  y_sort.Offset.OffsetPose.rot.q2:=0;
  y_sort.Offset.OffsetPose.rot.q3:=0;
  y_sort.Offset.OffsetPose.rot.q4:=0;
  check_limit:=150;

  WObjPick:=ItmSrcData{Index}.Wobj;
```

次のページに続く

© Copyright 2025 ABB. All rights reserved.

6.5.7.1 リニアコンベアの負の y 方向の並べ替え
続き

```
  GetItmTgt ItmSrcData{Index}.ItemSource,PickTarget
       \Limit:=check_limit\Selection:= y_sort;
  TriggL\Conc, RelTool(PickTarget.RobTgt, 0, 0,
       -ItmSrcData{Index}.OffsZ), MaxSpeed,
       ItmSrcData{Index}.VacuumAct1, z20, Gripper\WObj:=WObjPick;
  MoveL\Conc, PickTarget.RobTgt, LowSpeed, z5
       \Inpos:=ItmSrcData{Index}.TrackPoint, Gripper
       \WObj:=WObjPick;
  GripLoad ItemLoad;
  draw_target:=PickTarget.RobTgt;
  draw_target.trans.y:=-200;
  draw_target.rot:=[0,1,0,0];
  TriggL draw_target, LowSpeed, ItmSrcData{Index}.Ack, z20,
       Gripper\WObj:=WObjPick;
ENDPROC
```


© Copyright 2025 ABB. All rights reserved.

6.5.7.2 円形コンベアの負の半径方向の並べ替え

例の説明

本例では、周囲のオブジェクトに触れることなくコンベアベルトからアイテムをシャッフルします。シャッフル動作は、水平面でコンベヤの右側に垂直方向に実行され、マニピュレータのモーションはコンベヤのモーションに合わせて実行されます.

並べ替えアルゴリズムは、 x 方向の終了限界に最も近く、選択形状にロックオブジェクトを持たないアイテムを選択します.

選択形状は長いボックスとして定義されます。形状の y 値はコリドーの幅を定義するために使用されます。x値はコンベア ベルトの半径の半分より大きい必要があり、z 値はすべてのアイテムの最大の高さの差より大きくなければなりません。

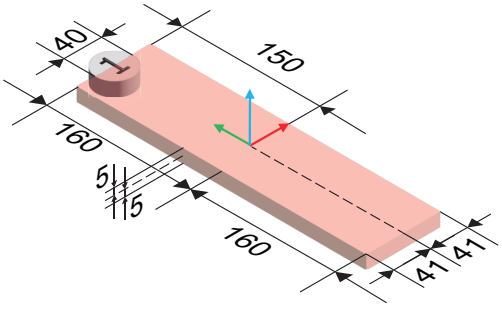
`OffsetData`の x 値をシェイプの負の x 値に設定すると、選択ボックスが右に移動します。

この結果、アイテムがシャッフルされる前に、すべてのアイテムの右側にクリアなコリドーが存在しなくてはなりません.

アルゴリズムは、生産フローの上下方向で、その他のアイテムの有無をチェックします.

y 方向のチェック制限を`GetItmTgt`命令のパラメータとして使用して、最初のワークエリアがシャッフルされる開始点を定義します。この命令は、チェック制限と入力制限の間で最初のワークエリアをシャッフルしようとします。これにより、選択アルゴリズムは、最も近いアイテムをチェックするときに、チェック リミットと終了リミットの間のすべてのアイテムを考慮するようになります。したがって、チェック リミットとエグジット リミットの間の距離は、少なくとも最大のアイテムの直径になります。

下の図では、すべてのアイテムがコンベア ベルトの右側にシャッフルされます。 各アイテムにはクリア ゾーン、つまり*ShapeType*の形状が必要であるため、アイテムは、下の図で番号が付けられているように 1 から 4 の順序でシャッフルされます。



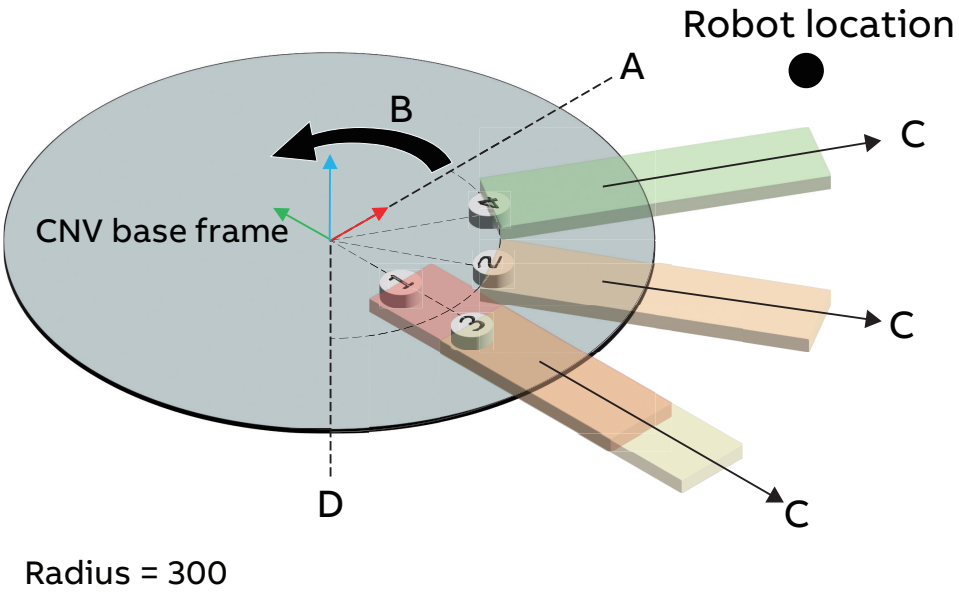
xx2400000740

反時計回りの移動方向コンベアの場合、負のy方向の並べ替えとして設定されたピッキング順序は4->2->3->1になります。

次のページに続く

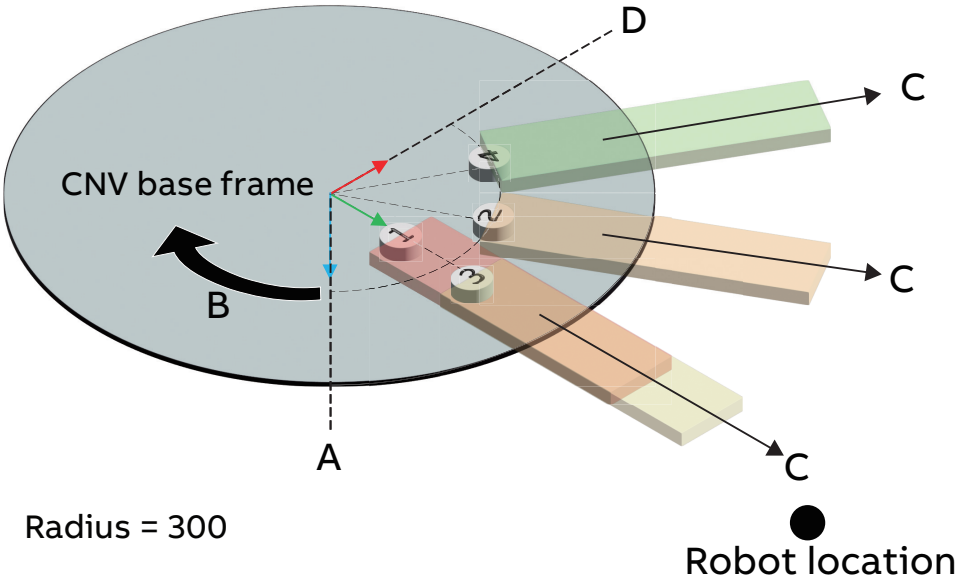
© Copyright 2025 ABB. All rights reserved.

6.5.7.2 円形コンベアの負の半径方向の並べ替え
続き



xx2400000738

時計回りの移動方向コンベアの場合、負の y 方向の並べ替えとして設定されたピッキング順序は3->1->2->4になります。



xx2400000739

A	終了
B	製品フロー方向
C	並べ替え方向
D	エンター

コードの例

```
PROC Pick(num Index)
  VAR selectiondata x_sort;
```

次のページに続く

© Copyright 2025 ABB. All rights reserved.

```
  VAR robtarget draw_target;
  VAR num check_limit;

  x_sort.ShapeType:=BOX;
  x_sort.ConsiderType:=BitOr(ITEMS_TO_USE,ITEMS_BYPASS);
  x_sort.GeometricData.x:=160;
  x_sort.GeometricData.y:=41;
  x_sort.GeometricData.z:=5;
  x_sort.GeometricData.radius:=0;
  x_sort.Offset.OffsetRelation:=FRAME_COORD_DIR;
  x_sort.Offset.OffsetPose.trans.x:=-150;
  x_sort.Offset.OffsetPose.trans.y:=0;
  x_sort.Offset.OffsetPose.trans.z:=0;
  x_sort.Offset.OffsetPose.rot.q1:=1;
  x_sort.Offset.OffsetPose.rot.q2:=0;
  x_sort.Offset.OffsetPose.rot.q3:=0;
  x_sort.Offset.OffsetPose.rot.q4:=0;
  check_limit:=150;

  WObjPick:=ItmSrcData{Index}.Wobj;
  GetItmTgt ItmSrcData{Index}.ItemSource,PickTarget
       \Limit:=check_limit\Selection:= x_sort;
  TriggL\Conc, RelTool(PickTarget.RobTgt, 0, 0,
       -ItmSrcData{Index}.OffsZ), MaxSpeed,
       ItmSrcData{Index}.VacuumAct1, z20, Gripper\WObj:=WObjPick;
  MoveL\Conc, PickTarget.RobTgt, LowSpeed, z5
       \Inpos:=ItmSrcData{Index}.TrackPoint, Gripper
       \WObj:=WObjPick;
  GripLoad ItemLoad;
  draw_target:=PickTarget.RobTgt;
  draw_target.trans.x:=-200;
  draw_target.rot:=[0,1,0,0];
  TriggL draw_target, LowSpeed, ItmSrcData{Index}.Ack, z20,
       Gripper\WObj:=WObjPick;
ENDPROC
```


© Copyright 2025 ABB. All rights reserved.

6.5.8 例：インデックスワークエリアおよび事前定義ポジション

例の説明

本例では、インデックスワークエリアと共に事前定義ポジションを使用します.

インデックスワークエリアとともに事前定義ポジションが使用する場合、EIO.cfg ファイルで構成を変更する必要があります。ポジションが事前定義されている場合、ストロボ信号を生成するシステムがありませんので、トリガとストロボ信号をクロスコネクションする必要があります。事前定義ポジションがない場合には、トリガ信号がビジョンシステムに送られてイメージを取得します。それから、イメージが取得されたことを知らせるため、ビジョンシステムからストロボが送り返されます.

これは、I/O信号によって外部からトリガされ、ポジションソースが事前定義ポジションタイプのラインにおける設定の例です。標準システムとかなり異なるシステムを設定する場合、固有の新しい信号名を定義する事が推奨されます.

このラインでは、2つの新しい信号が使用されています:

- トリガー信号 `doTrigSignal`。
- ストロボ信号 `diStrobeSignal`。

2つの信号を付加して信号の構成を変更します.

```
EIO_SIGNAL:
-Name "doTrigSignal" -SignalType "DO" -Unit "PPASIM" -UnitMap "6"
     -Access "ALL"
-Name "diStrobeSignal" - SignalType "DI" -Unit "PPASIM" -UnitMap
     "6" -Access "ALL"
```

ストロボ信号を送り返すビジョンシステムがないため、トリガとストロボ信号はクロスコネクションする必要があります.

```
EIO_CROSS
-Res "diStrobeSignal" -Act1 "doTrigSignal"
```

この場合のポジションジェネレータシグナルは、DSQC 328A:X3 ボードの接続 1 である `di1_1` です。*Circuit diagram - PickMaster Twin*, *3HAC024480-020* を参照してください。

（外部 I/O 信号によって）`di1_1` が High 状態になると、トリガー信号はパルスになります。トリガー信号とストロボ信号はクロスコネクションされているため、ストロボは即座に受信されます。次に、新しいアイテムポジションの受け入れ準備が完了したコントローラからコンピュータに対してイベントが送られ、事前定義ポジションコントローラに送られます。パターンが使用されている場合、各信号に対して複数のポジションが送られます。

この例では、ロボットの実行信号は使用されないため削除されています.

© Copyright 2025 ABB. All rights reserved.

6.5.9 例：インデックスワークエリアへの新規ポジションの自動生成

例の説明

本例では、空のキューに新しいポジションが自動入力される場合にインデックスワークエリアを構成する方法を説明します.

トリガとストロボ信号は例：インデックスワークエリアおよび事前定義ポジション ページ 446.

外部入力I/O信号を使用する代わりに、新しいシミュレート入力信号がポジションジェネレータ信号として使用されます。この信号は、クロスコネクションされたシミュレート出力信号で設定されます.

このラインでは、2つの新しい信号が使用されています:

- 出力ポジションジェネレータ信号 `doPosGenSignal`。
- 入力ポジションジェネレータ信号 `diPosGenSignal`。

2つの信号を付加して信号の構成を変更します.

```
EIO_SIGNAL:
-Name "doPosGenSignal" - SignalType "DO" -Unit "PPASIM" - UnitMap
     "7" -Access "ALL"
-Name "diPosGenSignal" - SignalType "DI" -Unit "PPASIM" - UnitMap
     "7" -Access "ALL"
```

The position generator signals are cross connected.

```
EIO_CROSS
-Res "diPosGenSignal" -Act1 "doPosGenSignal"
```

`diPosGenSignal` はラインでポジションジェネレータ信号として定義され 、`doPosGenSignal` はキューのアイドル状態信号として定義されます。

キューが空になると、キューのアイドル状態信号 `doPosGenSignal` が High になります。このクロスコネクションは `diPosGenSignal` を High にし、前述した原則に基づいて、新しいポジションがキューに押し出されます。

© Copyright 2025 ABB. All rights reserved.

6.5.10 例：アイテム バッファ

例の説明

この例では、アイテムバッファを使用します。アイテムは事前定義されたバッファーポジションに配置されます。

1 アイテムをピックワークエリアからピックします
2 バッファポジションに配置

> **注記**
>
> バッファポジションは、コンベア内の X 軸と Y 軸の範囲外でなければなりません。そうしないと、ロボットはアイテムをバッファポジションではなくコンベアに直接配置します。

コードの例

RAPID プログラム内のバッファポジションを定義します。次のコード例を参照してください。

```
!***********************************************************
! Global BUFFER Variables
! Robtarget BufferPos must be defined in wobj0
!***********************************************************
TASK PERS robtarget BufferPos{3}:=[
     [[-200,-10,-1084],[0,-1,0,0],[0,0,0,0],[0,0,0,0,0,0]] ,
     [[0,-10,-1084],[0,-1,0,0],[0,0,0,0],[0,0,0,0,0,0]] ,
     [[200,-10,-1084],[0,-1,0,0],[0,0,0,0],[0,0,0,0,0,0]] ];
VAR num BufferMax{3}:=[3,3,3]; !Number of items i a buffer
VAR num BufferPitch{3}:=[50,50,50]; !distance between buffer
     positions
VAR num DropAction:=0; !What to do if an item can not be used.
     0=Ack 1=Nack 2=Skip
VAR num BufferZ{3}:=[0,0,0]; !Buffer Z-adjustments

VAR num InFlowEnter:=-250; !Set to same as pickarea Enter limit
VAR num InFlowExit:=250; !Set to litte before pickarea Exit limit
VAR num OutFlowEnter:=-250; !Set to same as placearea Enter limit
VAR num OutFlowExit:=250; !Set to litte before placearea Exit limit

VAR num BufferX{3};
VAR num BufferY{3};
VAR num BufferIndex{3}:=[0,0,0];
VAR num IType:=0;
VAR num Picked:=0;
```

バッファポジションで Pick ルーチンと Place ルーチンを実行するように PickPlaceSeqルーチンを変更します。次のコード例を参照してください。

```
!***********************************************************
!
! Procedure PickPlaceSeq
!
! The Pick and Place sequence.
```

次のページに続く

© Copyright 2025 ABB. All rights reserved.

```
! Edit this routine to modify from which work areas to pick and
     place.
! Needs to be changed if more than one pick work area is used.
! Needs to be changed if more than one place work area is used.
!
!***********************************************************
PROC PickPlaceSeq()
VAR num GQL:=0;
Picked:=0;
WHILE Picked=0 DO
IType:=0;
!=== PickPosAvailable ? ===
GQL:=GetQueueLevel (ItmSrcData{PickWorkArea{1}}.ItemSource
     \MinLimit:=InFlowEnter \MaxLimit:=InFlowExit);
If GQL>0 THEN
NextItmTgtType ItmSrcData{PickWorkArea{1}}.ItemSource, IType;
!=== Matching Item on outfeeder ? ===
GQL:=GetQueueLevel (ItmSrcData{PlaceWorkArea{1}}.ItemSource
     \ItmType:=IType \MinLimit:=OutFlowEnter
     \MaxLimit:=OutFlowExit);
IF GQL>0 THEN
Picked:=1;
ELSE
!=== Empty pos in buffer? ===
IF BufferIndex{IType}<BufferMax{IType} THEN
Picked:=3;
ELSE
!=== What to do with the item on the infeeder? ===
GetItmTgt
     ItmSrcData{PickWorkArea{1}}.ItemSource,PickTarget\ItemType:=IType;
TEST DropAction
CASE 0:
AckItmTgt ItmSrcData{PickWorkArea{1}}.ItemSource,PickTarget,TRUE;
CASE 1:
AckItmTgt ItmSrcData{PickWorkArea{1}}.ItemSource,PickTarget,FALSE;
CASE 2:
AckItmTgt
     ItmSrcData{PickWorkArea{1}}.ItemSource,PickTarget,FALSE\Skip:=TRUE;
ENDTEST
Picked:=0;
ENDIF
ENDIF
ELSE
GQL:=GetQueueLevel (ItmSrcData{PlaceWorkArea{1}}.ItemSource
     \MinLimit:=OutFlowEnter \MaxLimit:=OutFlowExit);
If GQL>0 THEN
NextItmTgtType ItmSrcData{PlaceWorkArea{1}}.ItemSource, IType;
!=== Matching Item on Infeeder ? ===
GQL:=GetQueueLevel (ItmSrcData{PickWorkArea{1}}.ItemSource
     \ItmType:=IType \MinLimit:=InFlowEnter \MaxLimit:=InFlowExit);
IF GQL>0 THEN
```

次のページに続く

© Copyright 2025 ABB. All rights reserved.

6.5.10 例：アイテム バッファ
続き

```
Picked:=1;
ELSE
!=== Matching pos in buffer? ===
IF BufferIndex{IType}>0 THEN
Picked:=2;
ELSE
Picked:=0;
ENDIF
ENDIF
ELSE
Picked:=0;
ENDIF
ENDIF

TEST Picked
CASE 0:
!=== No pick, no place ===
WaitTime 0.1;
CASE 1:
!=== pick infeed, place outfeed ===
Pick PickWorkArea{1},IType;
Place PlaceWorkArea{1},IType;
CASE 2:
!=== pick buffer, place outfeed ===
PickBuffer PickWorkArea{1},IType;
Place PlaceWorkArea{1},IType;
CASE 3:
!=== pick infeed, place buffer ===
Pick PickWorkArea{1},IType;
PlaceBuffer PlaceWorkArea{1},IType;
ENDTEST
ENDWHILE
ENDPROC
```

このルーチンを編集して、ロボットがピック アンド プレイス動作を実行する方法を変更します。

```
!*********************************************************
!
! Procedure PickBuffer
!
! Executes a pick movement
!
!*********************************************************
PROC PickBuffer(num Index,num TypeNr)
VAR num zboffs;
BufferX{TypeNr}:=(BufferIndex{TypeNr}-1)*BufferPitch{TypeNr};
WObjPick:=Wobj0;
zboffs:=BufferZ{TypeNr};
TriggL\Conc,Offs(RelTool(BufferPos{TypeNr},0,0,-ItmSrcData{Index}.OffsZ),
BufferX{TypeNr},0,0),MaxSpeed,ItmSrcData{Index}.VacuumAct1,z20,
PickAct1\WObj:=WObjPick;
```

次のページに続く

© Copyright 2025 ABB. All rights reserved.

```
TriggL\Conc,Offs(BufferPos{TypeNr},BufferX{TypeNr},0,zboffs),
LowSpeed,ItmSrcData{Index}.SimAttach1,z5\Inpos:=ItmSrcData{Index}.TrackPoint,
PickAct1\WObj:=WObjPick;
GripLoad ItemLoad;
MoveL Offs(RelTool(BufferPos{TypeNr},0,0,-ItmSrcData{Index}.OffsZ),
BufferX{TypeNr}, 0,0), LowSpeed,z20,PickAct1\WObj:=WObjPick;
Decr BufferIndex{TypeNr};
ENDPROC


!*********************************************************
!
! Procedure PlaceBuffer
!
! Executes a place movement
!
!*********************************************************
PROC PlaceBuffer(num Index,num TypeNr)
VAR num zboffs;
Incr BufferIndex{TypeNr};
BufferX{TypeNr}:=(BufferIndex{TypeNr}-1)*BufferPitch{TypeNr};
WObjPlace:=Wobj0;
zboffs:=BufferZ{TypeNr};
MoveL\Conc,Offs(RelTool(BufferPos{TypeNr},0,0,-ItmSrcData{Index}.OffsZ),
BufferX{TypeNr},0,0), MaxSpeed,z20,PickAct1\WObj:=WObjPlace;
TriggL\Conc,Offs(BufferPos{TypeNr},BufferX{TypeNr},0,zboffs),LowSpeed,
ItmSrcData{Index}.VacuumRev1\T2:=ItmSrcData{Index}.VacuumOff1\T3:=
ItmSrcData{Index}.SimDetach1,z5\Inpos:=ItmSrcData{Index}.TrackPoint,
PickAct1\WObj:=WObjPlace;
GripLoad load0;
MoveL Offs(RelTool(BufferPos{TypeNr},0,0,-ItmSrcData{Index}.OffsZ),
BufferX{TypeNr},0,0),LowSpeed,z20,PickAct1\WObj:=WObjPlace;
ENDPROC
```


© Copyright 2025 ABB. All rights reserved.

このページは意図的に空白のまま残しています

7 トラブルシューティング

7.1 トラブルシューティング入門

トラブルシューティング

本章では、PickMaster PowerPac のインストール、設定、または実行時に知られている最も一般的な問題について説明します。

ロボットシステムのエラーはまず、以下のような兆候としてあらわれます:

- PickMaster Twin、FlexPendant、RobotStudio、または Windows イベント ビューアーを使用して表示できるイベント ログ メッセージ。
- システムが作動不良を起こす、または機械的な障害を示す。
- システムが起動しない、または起動時に不合理な動作を示す。
- LEDなどのハードウェアのインジケータの表示。
- その他の兆候。ロボットシステムは複雑で、多数の機能や機能の組み合わせが備えられています。

関連情報

ロボット システムの一般的なトラブルシューティングとすべてのエラー メッセージは、テクニカルリファレンスマニュアル - *RobotWare 7* のイベントログとユーザーマニュアル - *IRC5* のトラブルシューティングにリストされています。

© Copyright 2025 ABB. All rights reserved.

7.2 トラブルシューティング時の安全

一般

設置、保守点検、修理などの作業を行うときは、電気、エアー、油圧のスイッチをすべてオフにした状態で実施します。通常、あらゆるマニピュレータの動作はメカストップなどによって防止できます。

トラブルシューティングの作業はこれとは異なります。トラブルシューティングの際に、FlexPendant から手動で、そしてローカルに実行されているロボット プログラムやシステムが接続されている PLC により全て又は一部の電源が投入され、マニピュレーターの運動が制御される場合があります。

トラブルシューティング時の危険

これは、トラブルシューティングを行う場合、無条件で特別に考慮する必要があることを意味しています。

- あらゆる電気部品には電気が流れているとみなさなくてはなりません。
- マニピュレータは常にいかなる動作も行えるようにしておく必要があります。
- 通常は禁止されている機能を実行するために安全回路が切断されていたり、もしくは短絡されていたりする可能性があるので、システムは状況に応じて機能できなくてはなりません。



危険

電源投入時のコントローラのトラブルシューティングは、ABBによるトレーニングを受けた担当者か、ABBフィールドエンジニアが実行する必要があります。

© Copyright 2025 ABB. All rights reserved.

7.3 PickMaster Twin Powerpac のログの管理

ログメッセージの種類

イベント ログ メッセージには 3 つのタイプがあります。

タイプ	説明
情報	これらのメッセージはイベントログに情報を記録するためのものですが、これには特にユーザーが手を加える必要はありません。情報を伝えるメッセージはコントローラの表示装置に焦点を当てたものではありません。
警告する	これらのメッセージは、システムに不具合のある可能性をユーザーに伝えるものですが、動作は継続します。これらのメッセージはイベントログに置かれますが、表示装置に焦点を当てたものではありません。
Error（エラー）	これらのメッセージは、システムに重大な問題があることを示し、動作は停止します。ユーザーが直ちに処置を取る必要のある場合に表示されるメッセージです。

注記

メッセージのタイプは実際のメッセージテキストには含まれず、メッセージが生成された時にコード自体に含まれます。メッセージは1つのタイプとしてのみ表示され、そのタイプに関する情報を提供するために書かれています。

PickMaster Twin Powerpac でのログの管理

この手順を使用して、イベント ログを管理します。

1. ログビューでは、現在のステーションのすべてのイベントログがここに表示されます。
2. PickMaster Twin Powerpacを使用せずにイベント ログを表示する必要がある場合は、ログ領域を右クリックして、ログの保存を選択します。イベントログはcsvファイルとして保存されます。
3. 現在のステーションのイベント ログをクリアする必要がある場合は、ログ領域を右クリックしてすべてクリアを選択します。イベントログはすべてクリアされます。

© Copyright 2025 ABB. All rights reserved.

7.4 PickMaster Twin Operator のログの管理

ログメッセージの種類

イベント ログ メッセージには 3 つのタイプがあります。

タイプ	説明
PMOPInfo	これらのメッセージはイベントログに情報を記録するためのものですが、これには特にユーザーが手を加える必要はありません。情報を伝えるメッセージはコントローラの表示装置に焦点を当てたものではありません。
PMOPWarning	これらのメッセージは、システムに不具合のある可能性をユーザーに伝えるものですが、動作は継続します。これらのメッセージはイベントログに置かれますが、表示装置に焦点を当てたものではありません。
PMOPError	これらのメッセージは、システムに重大な問題があることを示し、動作は停止します。ユーザーが直ちに処置を取る必要のある場合に表示されるメッセージです。
RTInfo/RTStatus	これらのメッセージはイベントログに情報を記録するためのものですが、これには特にユーザーが手を加える必要はありません。情報を伝えるメッセージはコントローラの表示装置に焦点を当てたものではありません。
RTWarning	これらのメッセージは、システムに不具合のある可能性をユーザーに伝えるものですが、動作は継続します。これらのメッセージはイベントログに置かれますが、表示装置に焦点を当てたものではありません。
RTError	これらのメッセージは、システムに重大な問題があることを示し、動作は停止します。ユーザーが直ちに処置を取る必要のある場合に表示されるメッセージです。

注記

メッセージのタイプは実際のメッセージテキストには含まれず、メッセージが生成された時にコード自体に含まれます。メッセージは1つのタイプとしてのみ表示され、そのタイプに関する情報を提供するために書かれています。

PickMaster Twin Operator でのログの管理

この手順を使用して、イベント ログを管理します。

1 **ログイン分析グループ**をクリックします。イベントログが表示されます。
2 PickMaster Twin Operatorを使用せずにイベント ログを表示する必要がある場合は、**エクスポート**をクリックします。イベントログは`txt`ファイルとして保存されます。

© Copyright 2025 ABB. All rights reserved.

7.5 PickMaster Twin Runtime のログの管理

ログメッセージの種類

イベント ログ メッセージには 3 つのタイプがあります。

タイプ	説明
状態	これらのメッセージはイベントログに情報を記録するためのものですが、これには特にユーザーが手を加える必要はありません。情報を伝えるメッセージはコントローラの表示装置に焦点を当てたものではありません。
警告	これらのメッセージは、システムに不具合のある可能性をユーザーに伝えるものですが、動作は継続します。これらのメッセージはイベントログに置かれますが、表示装置に焦点を当てたものではありません。
Error（エラー）	これらのメッセージは、システムに重大な問題があることを示し、動作は停止します。ユーザーが直ちに処置を取る必要のある場合に表示されるメッセージです。

注記

メッセージのタイプは実際のメッセージテキストには含まれず、メッセージが生成された時にコード自体に含まれます。メッセージは1つのタイプとしてのみ表示され、そのタイプに関する情報を提供するために書かれています。

PickMaster Twin Runtime でのログの管理

この手順を使用して、イベント ログを管理します。

1. **ログビュー**では、すべてのイベント ログがここに表示されます。
2. PickMaster Twin Runtimeを使用せずにイベント ログを表示する必要がある場合は、**ログ**リボンをクリックするか、ログエリアを右クリックして**ログの保存**を選択します...イベント ログは`txt`ファイルとして保存されます。
3. 現在のステーションのイベント ログをクリアする必要がある場合は、ログ領域を右クリックして**すべてクリア**を選択します。イベントログはすべてクリアされます。

Windows イベント ビューアを使用したログの管理

PickMaster Runtimeのログエリアに表示されるイベント ログ メッセージは、Windows イベント ログにも保存されます。 メッセージは Windows イベント ビューアで表示できます。

下記の手順に従い、Windowsのイベントビューワを使用して、イベントログを管理してください.

1. ログエリアを右クリックし、［**イベントビューワ**］を選択します.

   ［Event Viewer（イベントビューワ）］はWindowsのControl Panel（コントロールパネル）からも起動できます.
2. **イベント ビューア**のツリー リストで、**Windows ログ -> アプリケーション**を選択します。
3. PickMaster メッセージのみを表示するには、**アプリケーション**を右クリックし、**現在のログをフィルタ**をクリックしてから...、イベント ソースとして**PickMaster**を選択します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

4 ログを保存するには、**アプリケーション**を右クリックし、**フィルターされたログ ファイルに名前を付けて保存**を選択します。

ログはネットワークを介して、他のコンピュータから確認できます。他のコンピュータからログを見るには、 **[Event Viewer（イベントビューワ）]** を右クリックし、 **[Connect to another computer（他のコンピュータへ接続）]** を選択し、 ネットワーク上のコンピュータのポジションを示します.

5 ログサイズの処理方法を設定するには、 **[Application（適用）]** を右クリックし、 [Propeties（プロパティ）] を選択します。. ログへの書き込みミスを、確実になくすには、 **[Overwrite events as needed（必要に応じてイベントを上書き）]** を選択します.

© Copyright 2025 ABB. All rights reserved.

7.6 Runtimeエラーコード

共通のエラー コード

エラーコード	タイプ	説明
4097	Error（エラー）	未定義エラー 理由：発生したエラーに対して正しいエラー ID が与えられていませんが、エラー メッセージにより理由が説明されているはずです.
4098	状態	情報のみ。
4099	Error（エラー）	コマンドライン オプション 理由：PickMasterで、スタートアップ時に /pなどの未知のコマンドラインが指定されました。
4100	Error（エラー）	説明:予期しないエラー 理由：PickMaster で予期しないエラーが発生しました。ログ メッセージで詳細情報を確認してください.
4101	Error（エラー）	XML 解析エラー 理由：pmline または pmproj ファイルで読み取りに問題が発生しました。どのファイルでエラーが発生したかは、ログ メッセージを確認してください。
4102	状態	ユーザー情報。
4197	Error（エラー）	プロジェクトが後のバージョンにアップグレードされ、ファイルが修正済みとしてマークされています。変更が永久的になるように、ファイルを保存する必要があります.
4198	Error（エラー）	ラインがより新しいバージョンにアップグレードされています。ラインそのものが開いている場合は修正済みとしてマークし、保存する必要があります。プロジェクトが開いている場合、作業を継続する前にラインを開いて保存する必要があります.
4199	Error（エラー）	プロジェクト ファイルの形式が無効です。ファイルがPickMasterのベータ バージョンで作成されたか、ファイルが壊れています.
4200	Error（エラー）	PickMaster は、その構成を書き込みの際、あるいは読み取りの際、Windows レジストリにアクセスできませんでした
4202	警告	現在のラインではプロジェクトがデザインされていません。プロジェクトを開こうとして、異なるラインに作成されたプロジェクトが既に開いています. 理由：一度に使用できるラインは 1 に限られています。 解決方法：開いているプロジェクトを閉じて、再度プロジェクトを開いてください.
4203	Error（エラー）	プロジェクトを開く際に、対応するラインをロードできませんでした。ラインファイルが壊れている可能性があります。
4204	Error（エラー）	ラインのロードに失敗しました。ファイルが壊れている可能性があります.
4205	Error（エラー）	インポートされたラインは再キャリブレートする必要があります 理由：インポートされたラインが他のカメラまたはレンズとともにデザインされている場合、カメラおよびロボットのベースフレームを再キャリブレートする必要があります.
4206	Error（エラー）	プログラム起動時に、選択された RIS プラグインをロードできませんでした。ファイルが壊れている可能性があります.

次のページに続く

© Copyright 2025 ABB. All rights reserved.

エラーコード	タイプ	説明
4207	Error（エラー）	選択された RIS プラグインが見つかりませんでした.
4208	Error（エラー）	以前使用可能だったラインの 1 つが別のラインで上書きされています。古いラインは使用可能なラインとして表示されず、そのラインでデザインされたプロジェクトも使用できません。
4209	Error（エラー）	ライン ファイルが無効なため、開くことができません.
4210	Error（エラー）	選択した言語に対してリソースをロードできませんでした。デフォルト言語（英語）が使用されます
4211	状態	プロジェクトが停止するまでにロボットが実行したピック合計数に関する通知です.
4212	Error（エラー）	ライン ファイルを削除できませんでした。ファイルは手動で削除する必要があります.
4213	警告	選択した言語の、html ヘルプ ファイルが見つかりませんでした。PickMaster フォルダに含まれる Documentation フォルダに、「Application manual xxx.chm」ファイルが存在することを確認してください.
4216	Error（エラー）	PickMaster が認識できないファイルを開こうとしました。
4217	Error（エラー）	時刻同期化サービスを使用できません. 理由: **PickMaster 時間同期サービス** が正常にインストールまたは起動してない可能性があります。 対処方法: サービスがインストールされているかを確認し、サービスを再起動してください。
4218	警告	同じサブネット上で2つ以上のネットワークが構成されています: x.x.x ユーザーガイドを参照し、推奨ネットワーク設定を確認してください。
4297	状態	既に実行されているプロジェクトを開始しようとしています.
4298	状態	開始されていないプロジェクトを停止しようとしています。
4300	Error（エラー）	カメラは現在、別のプロジェクトで使用中です。 理由：プロジェクトの開始時、ポジション ソースの 1 つが別のプロジェクトで使用しているカメラで構成されています。 解決方法：カメラは、生産において一度に 1 つのプロジェクトでしか使用できません。1 つのプロジェクトを再構成するか、これらを一度に 1 つずつ実行します。
4301	Error（エラー）	プロジェクト実行を開始できませんでした 理由：内部エラーが発生しました。メモリ不足が原因の可能性があります。 解決方法：PickMaster プログラムを再起動してみてください.
4302	Error（エラー）	プロジェクトの開始時に、ビジョン定義ポジション ソースにカメラが定義されていませんでした。 解決方法：ポジション ソースを削除するか、あるいは使用するカメラで構成してください。
4303	Error（エラー）	プロジェクトの開始時に、ポジション ソースにワーク エリアが定義されていませんでした。 解決方法：ポジション ソースを削除するか、あるいは使用するワーク エリアで構成してください。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

エラーコード	タイプ	説明
4304	警告	プロジェクトの開始時に、ビジョン定義ポジション ソースにビジョン モデルが定義されていませんでした。 解決方法：ポジション ソースを削除するか、あるいは使用するビジョン モデルを定義します。
4305	Error（エラー）	プロジェクトの開始時に、事前定義ポジション ソースにオブジェクトが定義されていませんでした。 解決方法：ポジション ソースを編集し、使用する事前定義オブジェクトを定義します.
4306	状態	モデルが作成されたカメラ以外のカメラで編集されました。 解決方法：ポジション ソースで正しいカメラが選択されていることを確認し、モデルを再教示してください.
4307	警告	ビジョン モデルがキャリブレートされていないカメラで作成されました。 解決方法：対応するラインを開いてカメラをキャリブレートしてから、モデルを教示します.
4308	Error（エラー）	プロジェクトの実行で、ビジョン モデルはオブジェクトを検出しましたが、参照するアイテムあるいはコンテナを検出できませんでした. 解決方法：プロジェクトを停止し、問題のビジョン モデルを削除して正しいアイテムに対して新しいビジョン モデルを作成します.
4309	警告	コンテナが正しく構成されていません。 解決方法：エラーメッセージで詳細情報を確認してください.
4310	状態	生産が正しく開始されませんでした。
4311	状態	生産が正しく停止されませんでした。
4312	警告	PickMaster は、限定された生産期間を持つデモ用ライセンスで実行されています。 理由：デモ用ライセンスしかインストールされていません。 解決方法：期間無制限でプロジェクトを実行するため、フル機能のライセンスをリクエストしてください。
4313	Error（エラー）	PickMaster はデモ用ライセンスで実行されており、許可された生産期間を超過しています. 解決方法：PickMaster プログラムを再起動し、プロジェクトを再開できるようにするため、フル機能のライセンスをリクエストしてください。
4314	Error（エラー）	不明な作業エリアからシーン情報を取得。
4315	状態	ポジションソースをトリガーするワーク エリアが変更されています。この変更は、プロジェクトのスタートアップ時、あるいは以前のトリガー ワーク エリアを持つロボット コントローラが停止した場合に発生します.
4319	警告	未知のワーク エリアからアイテム確認を受け取りました。
4320	警告	負荷分散を使用するプロジェクトがアップグレードされ、ワーク エリアの順序が生成されました。ワーク エリアの順序は、[ポジション ソースの構成] ダイアログボックスで検証する必要があります
4321	警告	ワーク エリアからアイテムの認知を受け取りましたが、対応するアイテム ポジションを検出できません。続くワーク エリアには、アイテム ポジションが既にアクセスされていることは通知されません.

次のページに続く

© Copyright 2025 ABB. All rights reserved.

続き

エラーコード	タイプ	説明
4326	警告	ストロボがないためにワークエリアでアイテムポジションが失われている。詳細については、「警告 4326 ~ 4329 ページ 475」を参照。
4327	警告	予期されるアイテムポジションがポジションソースにない。詳細については、「警告 4326 ~ 4329 ページ 475」を参照。
4328	警告	トリガー / ストロボ時間の不一致。ポジションソースからワークエリアへのアイテムポジションが失われている。詳細については、「警告 4326 ~ 4329 ページ 475」を参照。
4329	警告	トリガー / ストロボ時間の不一致。ワークエリアからのストロボが無視された。詳細については、「警告 4326 ~ 4329 ページ 475」を参照。
4396	Error（エラー）	外部センサーの使用中に COM エラーが発生しました。ログ メッセージで詳細情報を確認してください.
4397	Error（エラー）	外部センサー COM オブジェクトの関数を呼び出そうとしてエラーが発生しました。ログ メッセージで詳細情報を確認してください.
4398	Error（エラー）	外部ポジション ジェネレータでプロジェクトを開こうとして、対応するセンサーが使用されているラインで検出されませんでした.
4399	Error（エラー）	プロジェクトの開始時に外部センサーを起動できませんでした。ポジション ソースは生産中に使用できません.
4596	Error（エラー）	一般ユーザー フック エラーです。詳細については説明を参照してください.
4797	Error（エラー）	一般ライセンス エラーです。詳細については説明を参照してください.
4798	Error（エラー）	現在インストールされているライセンスで許可されているよりも多くのカメラが使用されています。 解決方法：カメラを削除するか、新しいライセンスをリクエストしてください。
4799	Error（エラー）	現在インストールされているライセンスで許可されているよりも多くのロボット コントローラが使用されています。 解決方法：ロボット コントローラを削除するか、新しいライセンスをリクエストしてください。
4800	Error（エラー）	現在インストールされているライセンスで許可されているよりも多くのカメラが、検査ビジョン モデルを使用しています. 解決方法：検査モデルを削除するか、新しいライセンスをリクエストしてください。
4804	Error（エラー）	現在インストールされているライセンスで許可されているよりも多くのロボット コントローラが、カメラ配置を使用しています。 解決方法：許可されている以上のカメラ ポジションが使用されないようにするか、新しいライセンスをリクエストしてください。
4805	Error（エラー）	該当するライセンスなしで、ATC を使用してプロジェクトを開始しようとしています。 解決方法：ATC オプションを含む新しいライセンスをリクエストするか、プロジェクトから ATC を削除します。
4806	警告	ライセンスの有効期限が 14 日以内で切れます. 解決方法：新しいライセンスをリクエストしてください。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

エラーコード	タイプ	説明
4807	Error（エラー）	現在インストールされているライセンスで許可されているよりも多くの外部センサーが使用されています. 解決方法：外部センサーを削除するか、新しいライセンスをリクエストしてください.
4808	Error（エラー）	該当するライセンスなしで、コンベヤを使用してプロジェクトを開始しようとしています。 解決方法：新しいライセンスをリクエストするか、あるいはプロジェクトからすべてのコンベヤを削除します.
4809	Error（エラー）	ネットワーク アダプタ（IP アドレス）が見つかりません. 解決方法：指定されたネットワーク カードが有効になっており、カードの IP アドレスが変更されていないことを確認してください.
4810	Error（エラー）	サービスへのアクセスが拒否されました. 理由：PickMaster が Windows サービスにアクセスできません.
4811	Error（エラー）	PickMaster 時刻同期化サービスにアクセスできません. 理由：PickMaster 時刻同期サービスがインストールされていません.
4812	Error（エラー）	PickMaster 時刻同期化サービスを停止できません.
4813	Error（エラー）	PickMaster 時刻同期化サービスを開始できません.
4814	Error（エラー）	PickMaster 時刻同期サービスの設定エラー。時刻同期サービスの設定パラメータを確認してください。
4815	警告	PickMaster 時刻同期サービス ネットワーク アダプタが選択されていません。
4896	Error（エラー）	説明：コントローラの作成時に予期しないエラーが発生しました。 理由：PickMaster で予期しないエラーが発生しました。ログ メッセージで詳細情報を確認してください.
4897	Error（エラー）	説明：ロボットの更新時に予期しないエラーが発生しました。 理由：PickMaster で予期しないエラーが発生しました。ログ メッセージで詳細情報を確認してください.
4898	Error（エラー）	説明：コントローラの設定時に予期しないエラーが発生しました。 理由：PickMaster で予期しないエラーが発生しました。ログ メッセージで詳細情報を確認してください.
4899	Error（エラー）	説明：ワークエリアを処理するときに予期しないエラーが発生しました。 理由：PickMaster で予期しないエラーが発生しました。ログ メッセージで詳細情報を確認してください.
4900	警告	ユーザー管理の動作が異常です。詳細については、ログ メッセージを参照してください。
4901	警告	外部センサーの異常。 外部センサーの設定を確認してください。
4902	状態	ビジョン情報
4903	状態	ユーザーフック情報
4904	状態	インデックスワークエリア情報
4905	状態	ラインファイル情報

次のページに続く

© Copyright 2025 ABB. All rights reserved.

エラーコード	タイプ	説明
4906	状態	ログ管理情報
4907	状態	ログイン情報
4908	状態	メインフォーム情報
4909	状態	カメラの機種情報
4910	警告	ネットワーク設定が異常です。ユーザーガイドの推奨ネットワーク設定を確認してください。
4911	警告	オプション設定が変更されました。詳細については、ログ メッセージを参照してください。
4912	警告	パターン動作が異常です。詳細については、ログ メッセージを参照してください。
4913	警告	説明：予期しないエラーが発生しました。 理由：PickMaster で予期しないエラーが発生しました。ログ メッセージで詳細情報を確認してください.
4914	警告	ポジションソース設定異常。詳細については、ログ メッセージを参照してください。
4915	警告	プロジェクトの設定が異常です。詳細については、ログ メッセージを参照してください。
4916	警告	RIS XML ファイルが異常です。詳細については、ログ メッセージを参照してください。
4996	状態	アイテムのポジション情報
4997	状態	アイテム重複フィルタ情報
4998	状態	アイテム配布情報
4999	状態	ワークエリアのアイテム情報
5000	状態	ワークエリアACK情報
16785	警告	プロジェクトのオープンが異常です。別のプロジェクトが実行されているときは、新しいプロジェクトを開くことはできません。
16885	状態	Ris2 クライアント接続情報
16886	状態	Ris2 クライアント切断情報

ロボットに関連するエラーコード

エラーコード	タイプ	説明
8193	状態	ロボットが実行中です.
8194	状態	ロボットが停止しています.
8195	状態	ロボットが一時停止中です
8196	警告	自動モードでロボットを設定してください. 理由：ロボットは起動されていますが、コントローラが自動モードに設定されていません. 解決方法：コントローラを自動モードに切り替えます.
8197	警告	自動モードを確認してください（FlexPendant）。 理由：ロボットが起動されて自動モードに設定されていますが、自動モードが確認されていません. 解決方法：FlexPendantで自動モードを確認してください.
8198	状態	ロボットが自動モードです.

次のページに続く

© Copyright 2025 ABB. All rights reserved.

エラーコード	タイプ	説明
8199	Error（エラー）	ロボット エラー X（iX はロボット エラー番号）. 解決方法：特定のエラーに対応するロボットのドキュメンテーションを参照してください.
8200	警告	ロボット警告 X（X はロボット警告番号）. 解決方法：特定の警告に対応するロボットのドキュメンテーションを参照してください.
8201	警告	ロボット プログラム コントローラが未知の状態です. 理由：ロボットは起動されましたが、プログラム コントローラが未知の状態にあります.
8202	警告	ガード停止 理由：ガードがアクティブ化されたため、ロボットが停止されました.
8203	警告	Emergency stop（非常停止） 理由：非常停止がアクティブ化されたため、ロボットが停止されました 解決方法：停止の理由を削除し、非常停止をリセットします。ロボットを再起動します（プロジェクトを停止せずに実行できます）.
8204	状態	RAPID プログラムは停止されています
8205	状態	RAPID プログラムが再起動されています。
8209	状態	ロボット コントローラでシステム障害が発生しています 理由：詳細についてはコントローラのイベント ログを参照してください
8211	Error（エラー）	失われた接続 理由：コンピュータは、コントローラとの接続を失いました。ネットワーク接続が機能していない可能性があります。コントローラが遮断されたか電源が切れている可能性があります. 解決方法：コントローラがオンになっており、電力供給が行われていることを確認します。また、ネットワーク接続が機能していることも確認します.
8212	警告	ロボット コントローラが他のプロジェクトで使用されています 理由：ロボット コントローラは、一度に 1 つのプロジェクトでしか使用できません
8213	警告	ロボット コントローラが使用されておらず、アクセスできません. 理由：アクセスしようとしたロボット コントローラは、プロジェクトで使用されるように構成されていません.
8214	警告	コントローラとの接続が回復します。
8215	警告	説明：予期しないエラーが発生しました。 理由：PickMaster で予期しないエラーが発生しました。ログ メッセージで詳細情報を確認してください.
8216	状態	コントローラの再接続情報
8217	状態	OmniCoreロボット接続情報
8218	状態	OmniCoreロボット切断情報
8219	状態	ロボット保有情報
8220	状態	ロボット休止情報

次のページに続く

© Copyright 2025 ABB. All rights reserved.

続き

エラーコード	タイプ	説明
8293	Error（エラー）	モーターオンを設定できませんでした。 理由：PickMaster はモーターオンを設定できませんでした。システムの状態によっては、PickMaster はモーターをオンに設定できない場合があります（例：非常停止、ガード停止など）.
8294	Error（エラー）	RAPID プログラムを起動できませんでした.
8295	Error（エラー）	RAPID プログラムを起動するための準備ができませんでした.
8297	Error（エラー）	RAPID 変数 RoutineName を「ClearAll」に設定できませんでした。 理由：変数 RoutineName が存在しないか、間違ったタイプです（ストリングタイプでなくてはなりません）。 解決方法：変数が存在し、ストリングタイプであることを確認します.
8298	Error（エラー）	ロボット コントローラの状態を取得できませんでした. 解決方法：コントローラの実行準備ができていることを確認します。できていなければ、コントローラを再起動してください.
8299	Error（エラー）	ロボット コントローラからイベントを取得できませんでした. 解決方法：コントローラの実行準備ができていることを確認します。できていなければ、コントローラを再起動してください。ライン中の特定のコントローラに対して、正しいネットワーク アダプタが使用されていることを確認します.
8300	Error（エラー）	RAPID 変数 StopProcess を「TRUE」に設定できませんでした。 解決方法：RAPID 変数 StopProcess が存在し、bool タイプであることを確認します.
8302	Error（エラー）	RAPID 変数 RoutineName を「PickPlace」に設定できませんでした。 理由：変数 RoutineName が存在しないか、間違ったタイプです（ストリングタイプでなくてはなりません）。 解決方法：変数が存在し、ストリングタイプであることを確認します.
8303	内部エラー	ワーク エリア ID が存在しないため、システムは新しいワーク エリア調整を適用できませんでした.
8304	内部エラー	ワーク エリア ID が存在しないため、システムは新しいワーク エリア設定を適用できませんでした.
8305	内部エラー	システムは新しいワーク エリア設定を適用できませんでした.
8306	Error（エラー）	DO 信号 を「doSafeStop」に設定できませんでした。 解決方法：この信号が存在し、正しく設定されていることを確認します.
8307	Error（エラー）	コントローラへ接続できませんでした. 解決方法：コントローラへのネットワーク アドレス（IP アドレス）が正しいことを確認します。コンピュータのネットワーク設定が正しいことを確認します。ライン上で、ロボット コントローラの接続に正しいネットワーク アダプタが使用されていることを確認します.
8308	Error（エラー）	コントローラへの IP アドレスを書き込みできませんでした. 解決方法：RAPID 変数 RemoteIPNode が存在し、正しいタイプ（ストリングタイプ）であることを確認します.

次のページに続く

© Copyright 2025 ABB. All rights reserved.

エラーコード	タイプ	説明
8309	Error（エラー）	ロボットコントローラからイベントを初期化できませんでした. 解決方法：ロボット コントローラの実行準備ができていることを確認します。できていなければ、コントローラを再起動してください.
8310	Error（エラー）	ロボット コントローラの状態を取得できませんでした. 解決方法：コントローラの実行準備ができていることを確認します。できていなければ、コントローラを再起動してください.
8313	Error（エラー）	IO 信号 ppaExe を設定できませんでした. 解決方法：信号 ppaExe が存在し、正しく設定されていることを確認します.
8314	Error（エラー）	RAPID 変数 RoutineName を「NewSource」に設定できませんでした. 理由：変数 RoutineName が存在しないか、間違ったタイプです（ストリングタイプでなくてはなりません）。 解決方法：変数が存在し、ストリングタイプであることを確認します.
8315	Error（エラー）	システムは新しいロボット速度を適用できませんでした.
8316	Error（エラー）	IO 信号 doTune を設定できませんでした. 解決方法：信号 doTune が存在し、正しく設定されていることを確認します.
8317	Error（エラー）	システムは新しいワーク エリア調整を適用できませんでした. 解決方法：次の RAPID 変数が存在することを確認します. Num SourceIndex Num TunePosX Num TunePosY Num TunePosZ
8318	Error（エラー）	RAPID プログラムをロードできませんでした. 解決方法：RAPID プログラムにエラーがないことを確認します（エラーがあるとロードできません）.
8319	Error（エラー）	RAPID プログラムをコントローラにダウンロードできませんでした.
8320	Error（エラー）	RAPID プログラムの実行を停止できませんでした.
8321	Error（エラー）	RAPID プログラムを削除できませんでした.
8322	Error（エラー）	非常停止をリセットできませんでした.
8323	Error（エラー）	RAPID プログラムを再起動できませんでした. 解決方法：プロジェクトを停止し、再起動します.
8324	Error（エラー）	ローカル IP アドレスを取得できませんでした。 理由：ネットワーク設定が正しくありません（例：間違った IP 設定、不完全なネットワーク アダプタ構成など）. 解決方法：コンピュータ上でローカル ネットワークの問題を解決します.

次のページに続く

© Copyright 2025 ABB. All rights reserved.

エラーコード	タイプ	説明
8325	Error（エラー）	キューを起動できませんでした. 理由：PickMaster がアイテム キューを起動できませんでした。キューは、RAPID 変数をいくつか設定することにより起動します。これらの変数は、削除または変更する必要があります。変数は次のとおりです: String ItmSrcName String CnvName String NonCnvWobjName Num SourceType Num SourceIndex Num TunePosX Num TunePosY Num TunePosZ Num FollowTime Num Vtcp Num OffsZ Num VacActDelay Num VacRevDelay 解決方法：RAPID プログラム、あるいは PAA システム モジュール（*ppasys.sys*）に全ての変数が存在し、正しいタイプ（string または num など）であることを確認します.
8326	Error（エラー）	ロボット コントローラと PickMaster コンピュータの時間を同期化できませんでした
8327	Error（エラー）	プロジェクトの開始時に、ロボット コントローラに定義されている RAPID プログラムがありません. 理由：ロボット コントローラに対してどの RAPID プログラムを使用するかを構成せずにプロジェクトを開始しようとしています. 解決方法：キューに存在するロボット コントローラに使用する RAPID プログラムを選択し、プロジェクトを再開します.
8337	Error（エラー）	アイテムソースキューの消去に失敗しました (ItmSrcCnvxx)。C0040403: コントローラーから応答がありません。 理由: 作業領域が大きすぎる大型ロボットの場合、CPU は `GetReachableTarget` 機能により、インデックスにより多くの時間を要します。 対処方法: 機能 UseReachableTargets と関連したリリースゾーン（インデックス作業レンジ）の精度は、新しいプロセスシステムパラメーター `Reach Zone Accuracy`（タイプ `Conveyor`中）で、0% ~ 100% まで調節可能です。デフォルト値は 100% です。CPU 負荷を少なくするには、この値をゼロまたは非常に低くします。UseReachableTargets 機能を使用しない場合は、`Reach Zone Accuracy` 値を 0 に設定して、オフにできます。
8338	Error（エラー）	コントローラに接続されていません. 理由：コントローラとの通信を完了できませんでした.
8339	Error（エラー）	ABB産業用ロボット通信Runtimeを使用してコントローラと通信するときに予期しないエラーが発生しました。 理由：エラー ログで詳細情報を確認してください.
8340	Error（エラー）	予期しないロボット エラー. 理由：エラー ログで詳細情報を確認してください.
8341	Error（エラー）	コントローラへの書き込みアクセスを取得できませんでした。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

エラーコード	タイプ	説明
8342	Error（エラー）	アイテムソースがポジションをコントローラに送れませんでした。コントローラから応答がありません。
8343	Error（エラー）	RobotWare バージョンは PC にインストールされている ABB Industrial Robot Communication Runtime 以降のものです。新しい ABB Industrial Robot Communication Runtime をインストールする必要があります. 解決方法：可能であれば、PickMaster を最新バージョンに更新します。これで問題が解決しない場合や、なんらかの理由で更新が不可能である場合、PC で ABB Industrial Robot Communication Runtime を更新してください。 システムは *RobotStudio オンラインコミュニティ*からダウンロードできます。*Tools and Utilities* パッケージにあります。
8345	Error（エラー）	自動でプログラムを開始できませんでした。 考えられる理由：RW ロール設定「自動でのリモート開始/停止プログラム」が選択されていません。
8393	Error（エラー）	モーション サーバーが既にインスタンスとして存在します（インスタンスは 1 つしか許可されません）。
8394	Error（エラー）	ロボット ID が既に存在します（ID は固有でなくてはなりません）.
8395	Error（エラー）	この ID を定義されたロボット コントローラがありません。
8396	Error（エラー）	ワーク エリアがまだ存在します。ワーク エリアを削除する前にコンベヤを削除することはできません。 解決方法：コンベヤのワーク エリアをすべて削除します.
8397	Error（エラー）	この ID を持つワーク エリアは既に存在します（すべての ID は固有でなくてはなりません）。
8398	Error（エラー）	この ID を持つワーク エリアは存在しません。操作が存在しないワーク エリアで実行されています。ワーク エリアが削除されている可能性があります.
8399	Error（エラー）	ワーク エリア ID が正しくないため、ワーク エリアを設定できませんでした.
8400	Error（エラー）	ワーク エリア ID が正しくないため、システムは新しいワーク エリア設定を適用できませんでした.
8401	Error（エラー）	ワーク エリア ID が存在しないため、システムは新しいワーク エリアを設定できませんでした.
8402	Error（エラー）	ワーク エリア ID が存在しないため、システムは新しいワーク エリア調整を適用できませんでした.
8403	Error（エラー）	ロボット ID が存在しないため、システムは新しいロボット設定を適用できませんでした.
8404	Error（エラー）	ロボット ID が存在しないため、システムは新しいロボット設定を設定できませんでした.
8406	Error（エラー）	ロボット ID が存在しないため、システムは新しいロボット速度を設定できませんでした.
8407	Error（エラー）	ワーク エリア タイプが正しくないため（インデックス ワーク エリア/コンベヤ ワーク エリア）、ワーク エリアを更新できませんでした.
8408	警告	ロボットに対してワーク エリアが定義されていません. 解決方法：プロジェクトが開始される前に、ロボットのワーク エリアを定義し、そのワーク エリアに対するポジション ソースを設定します

次のページに続く

© Copyright 2025 ABB. All rights reserved.

エラーコード	タイプ	説明
8418	状態	コントローラからエラー ログ ファイルをダウンロードしています. 理由：エラー ログ ファイルが生産開始時に検出されないと、自動的にダウンロードされます.
8419	Error（エラー）	アイテムターゲット情報を更新
8420	Error（エラー）	サポートされていないアイテムターゲット情報を更新
8421	状態	ロボット保持情報
8422	状態	ロボット停止情報

ビジョンに関連するエラーコード

エラーコード	タイプ	説明
12298	状態	フレーム取り込みや Gigabit Ethernet カメラがインストールされていません。
12299	内部エラー	問題のカメラがビジョン サーバーにありません.
12300	内部エラー	問題のビジョン モデルがビジョン サーバーにありません.
12301	内部エラー	カメラはロックされています.
12302	内部エラー	既に存在するカメラを作成またはロードしようとしています.
12305	Error（エラー）	現在のフレーム取り込みは、選択されたビデオ形式をサポートしません.
12306	内部エラー	カメラの作成に失敗しました.
12307	内部エラー	ビジョン サーバーは、実行時に取得したカメラを検出できません.
12308	警告	カメラのトリガー時期が早すぎます。 理由：最終イメージの分析が完了する前にカメラがトリガーされました。イメージの数が少ない場合は、イメージは失われません. 解決方法：分析時間がより速くなるよう、カメラのビジョン モデルを調整します。システム パフォーマンスそのものを改善する必要があるため、他のカメラのモデルも調整します。必要に応じて、コンベヤ速度を遅くして問題を解決することもできます.
12309	Error（エラー）	プロジェクトの実行時に、カメラからイメージを取得できませんでした. 理由：システムへの負荷が大きすぎるか、フレーム取り込みのトリガーが早すぎた可能性があります。 解決方法：システム負荷を検証し、ロボット コントローラが間違ったビジョン トリガーを送らないようにします.
12310	内部エラー	図形モデルを作成できませんでした. 理由：エラー メッセージで詳細情報を確認してください.
12312	内部エラー	存在しないフレーム取り込み上のカメラポートにアクセスしようとしています.
12313	内部エラー	フレーム取り込みに、そのカメラに指定されたカメラポートがありません. 解決方法：対応するラインを開き、カメラをカメラポートで構成します.

次のページに続く

© Copyright 2025 ABB. All rights reserved.

エラーコード	タイプ	説明
12315	Error（エラー）	プロジェクトの開始時に、カメラを起動できませんでした. 理由：システムのリソースが不足している可能性があります.
12316	Error（エラー）	外部モデルは画像の分析に失敗しました. 理由：ログ メッセージで詳細情報を確認してください
12317	Error（エラー）	プロジェクトの開始時に、外部モデルを起動できませんでした. 理由：ログ メッセージで詳細情報を確認してください
12318	Error（エラー）	画像を外部ビジョン モデルでサポートされる形式に変換できませんでした.
12319	Error（エラー）	外部モデルは画像の検査に失敗しました. 理由：ログ メッセージで詳細情報を確認してください
12321	Error（エラー）	ラインを開いたときに、複数のカメラが同じフレーム取り込みの同じポートを使用するよう定義されています。1つのカメラポートを使用するよう構成できるのは1つのカメラだけで、他のカメラはリセットされるため、構成をやり直す必要があります.
12322	Error（エラー）	ラインを開いたときに、カメラが使用不可能なフレーム取り込みで定義されています。カメラはリセットされ、構成をやり直す必要があります.
12323	Error（エラー）	カメラを初期化できませんでした。ログ メッセージで詳細情報を確認してください.
12324	Error（エラー）	カメラ構成の保存に失敗しましたログ メッセージで詳細情報を確認してください
12325	Error（エラー）	カメラ構成のロードに失敗しました。ログ メッセージで詳細情報を確認してください.
12326	Error（エラー）	ビジョン モデル構成のロードに失敗しました。ログ メッセージで詳細情報を確認してください
12329	警告	ギガビット イーサネット カメラとの通信に失敗しました. 理由：イーサネット接続が不適切であるか、イーサネット通信が過剰です.
12330	警告	画像のトリガー頻度が多すぎます。 解決方法：ビジョン モデルを調整して、時間の消費を少なくするか、トリガー頻度を少なくします.
12331	警告	カメラとの接続が失われました。再接続を試行中です。 理由：イーサネット ケーブルまたは電源ケーブルの接続が解除されました.
12332	警告	イメージ バッファがいっぱいです。ログ メッセージで詳細情報を確認してください.
12333	警告	ギガビット イーサネット カメラは見つかりましたが、このライセンスは検出されませんでした. 理由：ビジョン ライセンスが含まれる USB スティックが PC に挿入されていません.
12334	警告	ギガビット イーサネット ビジョンのライセンスは検出されましたが、このカメラは見つかりませんでした. 理由：カメラが接続されていないか、オンになっていないか、IP アドレスが無効です.

次のページに続く

© Copyright 2025 ABB. All rights reserved.

エラーコード	タイプ	説明
12337	警告	カメラの作成に失敗しました。 理由: 適切なCognex Driverがインストールされているか確認してください。問題が続く場合は、ネットワーク接続を調べてください。
12341	状態	Cognex USB ライセンス ドングルが取り付けられます。
12342	警告	Cognex USB ライセンス ドングルが取り外されます。
12343	状態	詳細ビジョン運用情報
12344	警告	PatMax に対して有効なキャリブレーションが選択されていません。カメラの高さを使用できません。
12345	警告	PatMax に対して有効なキャリブレーションが選択されていません。モデルやキャリブレーションを修正してください。
12346	警告	検査用に有効なキャリブレーションが選択されていません。モデルやキャリブレーションを修正してください。
16985	Error（エラー）	カメラからの画像の取得に失敗しました。パラメータの設定とハードウェアの接続を確認してください。
16986	Error（エラー）	説明：ビジョンの使用時に予期しないエラーが発生しました。 理由：PickMaster で予期しないエラーが発生しました。ログ メッセージで詳細情報を確認してください.
16987	Error（エラー）	説明：ビジョンの設定時に予期しないエラーが発生しました。 理由：PickMaster で予期しないエラーが発生しました。ログ メッセージで詳細情報を確認してください.
16988	Error（エラー）	カメラ設定エラー。この機能は実際のカメラの種類では利用できません。
16989	Error（エラー）	白値エラー。詳細については、ログ メッセージを参照してください。
16990	Error（エラー）	カメラのビデオ形式はサポートされていません。
16991	Error（エラー）	対象領域が無効です。画像全体が使用されます。
16992	Error（エラー）	チェッカーボード キャリブレーションはトレーニングされていません。
16993	Error（エラー）	説明：キャリブレーションに関する予期しないエラーが発生しました。 理由：PickMaster で予期しないエラーが発生しました。ログ メッセージで詳細情報を確認してください.
16994	Error（エラー）	キャリブレーション エラー。チェッカーボード画像内の頂点を識別できませんでした。設定を確認してください。
16995	Error（エラー）	キャリブレーション エラー。原点マーカーが見つかりません。
16996	Error（エラー）	キャリブレーション エラー。 チェッカーボード画像内に十分な頂点がありません。
16997	Error（エラー）	キャリブレーション エラー。 カメラをキャリブレーションするには、原点画像を設定する必要があります。
16998	Error（エラー）	キャリブレーション エラー。到達領域はソース画像の外側に定義されます。
16999	Error（エラー）	カラーフィルタの設定ミスです。すべての設定を適用できません。ご検討ください。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

エラーコード	タイプ	説明
17000	Error（エラー）	PatMax エラー。ビジョン ライセンスではカラー ツールが有効になっていません
17001	Error（エラー）	PatMax エラー。パラメータが不正です。チェックしてください。
17002	Error（エラー）	PatMax エラー。検索領域は画像領域の外側に定義されます。
17003	Error（エラー）	PatMax エラー。ポジション制限は画像領域の外側に定義されます。
17004	Error（エラー）	PatMax エラー。選択した領域をトレーニングできませんでした。
17005	Error（エラー）	PatMax モデルエラー。モデルはトレーニングされていません。
17006	Error（エラー）	PatMax エラー。検索タイムアウト。
17007	Error（エラー）	PatMax エラー。無効な検索領域です。
17008	Error（エラー）	PatMax エラー。選択した領域が画像内に完全に収まりません。
17009	Error（エラー）	ブロブエラー。検索領域は画像領域の外側に定義されます。
17010	Error（エラー）	ブロブエラー。領域を定義するには、トレーニングされたモデルに少なくとも 1 つのヒットが必要です。
17011	Error（エラー）	ブロブエラー。ヒストグラムの無効な検索領域。
17012	Error（エラー）	SubPatMax エラー。詳細については、ログ メッセージを参照してください。
17013	Error（エラー）	ラインファイルエラーです。 指定された行ファイル内に指定されたカメラが見つかりませんでした。
17014	Error（エラー）	説明：カメラの使用時に予期しないエラーが発生しました。 理由：PickMaster で予期しないエラーが発生しました。ログ メッセージで詳細情報を確認してください.

ユーザー スクリプトのエラー コード

エラーコード	タイプ	説明
41989	メッセージ	{%s} での {%s} の実行がタイムアウトしました。
41990	メッセージ	{%s} のロードで {%s} のモジュールが失敗しました。
41991	メッセージ	{%s} ロードの {%s} のインターフェースが失敗しました。
41992	メッセージ	{%s} の {%s} の戻り値が正しくありません。
41993	メッセージ	{%s} 内の {%s} の戻りデータ構造が正しくありません。
41995	メッセージ	{%s} の {%s} の戻りデータに {%s} の要素が見つかりませんでした。
41996	メッセージ	[{%.1f}, {%.1f}] のポジションは、物体 ID が不明なため破棄されました。 {%s} の {%s} からです。
41997	メッセージ	{%s} の {%s} の戻りデータで、{%s} の要素タイプが正しくありません。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

続き

エラーコード	タイプ	説明
42003	メッセージ	{%s} で {%s} が実行されたときに、ドキュメント フォルダ パスの取得に失敗しました。確認してください。

© Copyright 2025 ABB. All rights reserved.

7.7 故障の徴候またはエラー

7.7.1 警告 4326 ~ 4329

検証アクション

以下に示すのは、警告 4326、4327、4328、および 4329 に対する一般的検証アクションです。詳細については、「警告 *4326* ページ *476*」、「警告 *4327* ページ *477*」、「警告 *4328* および *4329* の同時受信 ページ *477*」、「警告 *4328* だけを受信（警告 *4329* なし） ページ *478*」、「警告 *4329* だけを受信（警告 *4328* なし） ページ *478*」を参照してください。

アクション 1

PickMaster ラインのワークエリア構成で選択されているトリガー信号とストロボ信号をチェックします。これらの信号の I/O 構成が配線に対応しているかどうかチェックします。

アクション 2

すべてのトリガー / ストロボ配線をチェックします。トリガーケーブルとストロボケーブルが混じっていないかチェックします。ケーブルがシールドされ、適切に接続・接地されていることを確認します。シールドに電流が流れていてはなりません。24 V の複数の電源が混じり合っていないことを確認します。コントローラシステムパラメータ *SyncSeparation*（トピック：I/O、タイプ：Fieldbus Command、名前：CNVX）を変更して、カメラまたはセンサーからのストロボ入力イベントをフィルタすることができます。

アクション 3

ロボットネットワーク上の LAN ケーブルすべてをチェックしてください。ケーブルが被覆され、適切に取り付けられているか確認してください。正しい IP アドレス、デフォルトゲートウェイ、およびサブネットマスクが（PC とロボットコントローラーの両方で）定義されているかチェックしてください。ネットワーク上にコンピューターが1台およびロボットコントローラーが1基のみでも、3つの値すべてが定義されている必要がありますので、注意してください。詳細は、ネットワークの設定 ページ *51* を参照してください。

アクション 4

ネットワークの設定 ページ *51*を参照してください。

アクション 5

「コントローラ ネットワーク アダプタ」フィールドの IP アドレス (**ファイル** に移動し、RRT の **オプション** をクリック) が、ロボット コントローラと通信する PC のネットワーク インターフェイス カードのアドレスであることを確認します。 時刻同期サービスがコントローラに接続できないかどうかを確認します。 サービスを 30 秒間停止してから、もう一度再開します。 時刻同期サービスに影響を与えるアクティブなファイアウォールがないことを確認してください。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

7.7.1 警告 4326 ~ 4329
続き

アクション 6

トリガー頻度を低くします。トリガー間隔が短すぎるために、対応できない頻度でシステムがトリガーされている場合があります。処理可能なトリガーの頻度は、システムで使用されているモデルの複雑さで決まります。トリガー / ストロボの誤配線または電気ノイズが原因で、トリガーが頻発する場合もあります。

アクション 7

一部のスイッチは、存在しなければならないデータをバッファリングしています。このバッファリング時間が長すぎる場合可能性があります。単純なハブに切り替えるか、このバッファ時間を短くしてみてください。また、ハブ / スイッチで最新のソフトウェアが実行されていることを確認します。ラピッドコードに無限ループがないことを確認します。無限ループがあるとロボットネットワーク通信に影響します。

アクション 8

外部センサーの実装をデバッグします。

アクション 9

外部センサーの場合、ストロボパルスとタイムスタンプの記録の間に常に短い遅れが生じることがあります（例えば、トリガー信号がストロボと相互接続されている場合）。ポジションソースパラメータ *Synchronization tune* を変更して、PickMaster に送信されるすべてのタイムスタンプを一定の時間値で修正します。

警告 4326

検証アクションについては、前のセクションを参照してください。

エラーの説明：

4326 ストロボがないためにアイテムポジションが %s で失われている。アプリケーションマニュアルを参照。

考えられる原因：

次の表に、警告 4326 の考えられる原因を示します。

考えられる原因	検証アクション
ワークエリアがコンベヤの場合：	
開始入力時にコンベヤボードがストロボパルスを受信していない。	*アクション 1 ページ 475*、*アクション 2 ページ 475*
ストロボ信号が cXNewObjStrobe として構成されていない。	*アクション 1 ページ 475*
PickMaster PC がロボット コントローラに接続されていません。	*アクション 3 ページ 475*
ワークエリアがインデックスの場合：	
構成されたストロボ信号がストロボパルスを受信していない。	*アクション 1 ページ 475*、*アクション 2 ページ 475*
PickMaster PC がロボット コントローラに接続されていません。	*アクション 3 ページ 475*

次のページに続く

© Copyright 2025 ABB. All rights reserved.

警告 4327

エラーの説明：

4327 予期されるアイテムポジションが %s にない。アプリケーションマニュアルを参照。

考えられる原因：

次の表に、警告 4327 の考えられる原因を示します。

考えられる原因	検証アクション
ソースタイプがカメラの場合：	
カメラがトリガーパルスを受信していない。	*アクション 1 ページ 475*、*アクション 2 ページ 475*
PickMaster PC はカメラに接続されていません。	*アクション 4 ページ 475*
ソースタイプが外部センサーの場合：	
外部センサーがトリガーパルスを受信していない。	*アクション 1 ページ 475*、*アクション 2 ページ 475*
外部センサーが PickMaster にポジションを送信していない。	*アクション 8 ページ 476*
ソースタイプが外部センサーの場合：	
外部センサーがトリガーパルスを受信していない。	*アクション 1 ページ 475*、*アクション 2 ページ 475*
外部センサーが PickMaster にポジションを送信していない。	*アクション 8 ページ 476*
ソースタイプが事前定義済みで、ワークエリアがコンベヤの場合：	
開始入力時にコンベヤボードがストロボパルスを受信していない。	*アクション 1 ページ 475*、*アクション 2 ページ 475*
ストロボ信号が cXNewObjStrobe として構成されていない。	*アクション 8 ページ 476*
ストロボ ケーブルが接続されているかどうかを確認するには、コンベア トラッキング ボードの同期 LED を確認します。	*アクション 3 ページ 475*
ソースタイプが事前定義済みで、ワークエリアがインデックスの場合：	
構成されたストロボ信号がストロボパルスを受信していない。	*アクション 1 ページ 475*、*アクション 2 ページ 475*
PickMaster PC がロボット コントローラに接続されていません。	*アクション 3 ページ 475*

警告 4328 および 4329 の同時受信

エラーの説明：

通常は、1 つ、複数、またはすべてのワークエリアに関係するトリガー / ストロボに対して 4328 と 4329 の 2 つが同時に受信されます。

4328 トリガー / ストロボ時間の不一致（%.1f s）。%s ~ %s のアイテムポジションが失われている。アプリケーションマニュアルを参照。

4329 トリガー / ストロボ時間の不一致（%.1f s）。%s からのストロボが無視された。アプリケーションマニュアルを参照。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

7.7.1 警告 4326 ~ 4329
続き

考えられる原因：

次の表に、警告 4328 および 4329 の考えられる原因を示します。

考えられる原因	検証アクション
可能性の高い順：	
コントローラと PickMaster 間の時間同期化が機能していない。	*アクション 6 ページ 476*
トリガー頻度の設定が高すぎる。	*アクション 5 ページ 475*
ロボットネットワークのパフォーマンスが低い	*アクション 7 ページ 476*
カメラネットワークのパフォーマンスが低い	*アクション 4 ページ 475*
外部センサーの場合の他の原因：	
タイムスタンプがストロボと充分に同期化されていない。	*アクション 9 ページ 476*
外部センサーがポジションとともに送信するタイムスタンプが正しくない。	*アクション 8 ページ 476*

警告 4328 だけを受信（警告 4329 なし）

エラーの説明：

4328 トリガー / ストロボ時間の不一致（%.1f s）。%s ~ %s のアイテムポジションが失われている。アプリケーションマニュアルを参照。

考えられる原因：

次の表に、警告 4328 および 4329 の考えられる原因を示します。

考えられる原因	検証アクション
トリガー信号が不安定。	*アクション 2 ページ 475*

警告 4329 だけを受信（警告 4328 なし）

エラーの説明：

4329 トリガー / ストロボ時間の不一致（%.1f s）。%s からのストロボが無視された。アプリケーションマニュアルを参照。

考えられる原因：

次の表に、警告 4328 および 4329 の考えられる原因を示します。

考えられる原因	検証アクション
ストロボ信号が不安定。	*アクション 2 ページ 475*

© Copyright 2025 ABB. All rights reserved.

7.7.2 カメラが画像を取得しない

エラーの説明

カメラが画像を取得しない.

考えられる原因

カメラが画像を取得しない場合には、いくつかの原因があります。すべての考えられる原因をチェックするには、下記を確認してください.

- トリガケーブルが適切に接続されているかチェックしてください。
- カメラケーブルが正しいポートに接続されていることを確認してください.

カメラが距離トリガされていると、下記の理由により、エンコーダーがコンベヤの動きを記録していない場合があります

- エンコーダーの接続が不良である、または
- ワークエリア内で間違ったコンベヤが選択されている.

カメラがI/Oトリガされていると、下記の理由により、フォトアイが部品を検知していない場合があります:

- 誤接続。
- 反射不良。

© Copyright 2025 ABB. All rights reserved.

7.7.3 ロボットが動かない

エラーの説明

カメラはオブジェクトを認識しているが、ロボットが動かない.

考えられる原因

カメラが写真を撮影しているのにロボットが動かない場合には、いくつかの原因があります。すべての考えられる原因をチェックするには、下記を確認してください.

- ストロボ ケーブルが接続されていることを確認するには、コンベア トラッキング ボードの Sync LED を確認します。
- ポジションソース内の分配をチェックしてください。
- I/OリストのAI*c*Speed*をチェックし、速度が検知されているかチェックしてください。チェックされていなければ、エンコーダーパルス信号を確認してください.
- I/OリストのAI*c*Position*をチェックし、ポジションがトラッキングされているかチェックしてください。トラッキングされていない場合には、ポジションソース内の配分をチェックしてください.
- DI*c*DirOfTravel*上の移動方向をチェックしてください。
- *Queue Idle*信号を監視し、キューがポジションを取得しているか確認してください。
- *Position Available*信号を監視し、製品が検知されているか確認してください.

© Copyright 2025 ABB. All rights reserved.

7.7.4 ポジション精度が低い、または変動する

エラーの説明

ポジション精度が低い、または変動します。

考えられる原因

ポジション精度が低い、または変動する場合には、いくつかの原因があります。すべての考えられる原因をチェックするには、下記を確認してください.

- *Counts Per Meter*のキャリブレーションが正確であることを確認してください。複数回確認してください。定期メンテナンスも考慮してください.
- ローラーとベルト間のスリップ率が変動するので、ドライブシャフトエンコーダーの使用は避けてください.
- カメラのキャリブレーションをチェックしてください。キャリブレーショングリッドの品質が悪いと、キャリブレーション結果の精度も悪くなります。
- キャリブレーション用紙の高さと製品の高さに違いがないかチェックしてください。
- 高さの高い製品を認識する場合に、焦点距離が正しいかチェックしてください。
- カメラがロボットフレーム上に搭載されて振動を受けていないことを確認してください.

© Copyright 2025 ABB. All rights reserved.

7.7.5 ポジションが2度使われる

エラーの説明

ロボットが各ポジションを2度使用します。

考えられる原因

ロボットが各ポジションを 2 度使用する場合には、いくつかの原因があります。すべての考えられる原因をチェックするには、下記を確認してください.

- I/Oトリガされた事前定義ポジションまたはコンテナを使用する場合には、*SyncSeparation*フィルタ間隔を設定して、ダブルおよびゴーストトリガを避けてください。
- ビジョンを使用する場合には、オーバーラップフィルタとポジションフィルタの値を増やしてください.
- ポジションソース内の［**Same level only（同一レベルのみ）**］チェックを外してください。

ATCグループのロボットの下流で、すでに使用されたアイテムが使用される場合には、ポジションソース内のワークエリアの順序が間違っています。

© Copyright 2025 ABB. All rights reserved.

7.7.6 PickMaster のカメラの解像度の問題

エラーの説明

カメラの画像サイズは、校正画像解像度と比較して解像度を低下させるために縮小されます。

考えられる原因

カメラの解像度を低下させる理由が幾つかあります。可能性のある原因すべてを確認するために、次のことを検証する必要があります。：

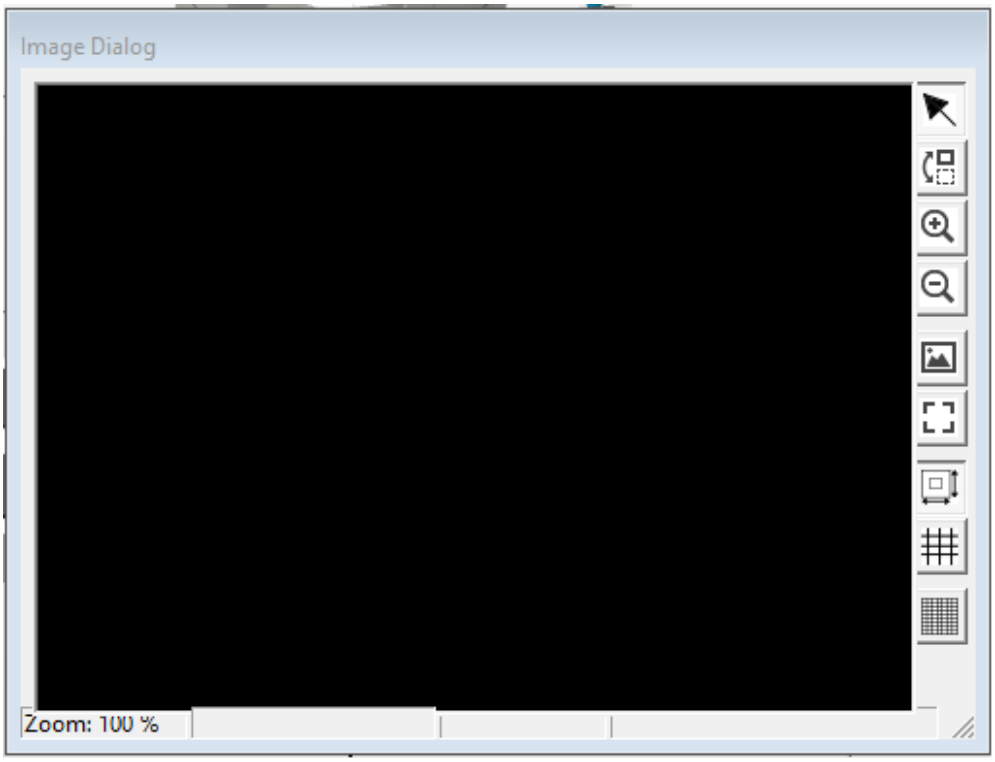
- 工場出荷時の既定の設定が有効になっています。
- カスタム構成が有効になっている場合があります。カスタム構成が低減された ROI (関心領域) を持っているかを確認します。

© Copyright 2025 ABB. All rights reserved.

7.7.7 画像ダイアログを表示できません

エラーの説明

ユーザーがカメラ関連の機能 (カメラ設定、カメラ キャリブレーション、図形モデル、ブロブ モデル、検査モデル、ライブ ビデオ、詳細ビジョン) を使用しようとすると、以下に示す特定の画像ダイアログが表示されません。「純粋仮想機能呼び出し」エラーが表示されることがあります。



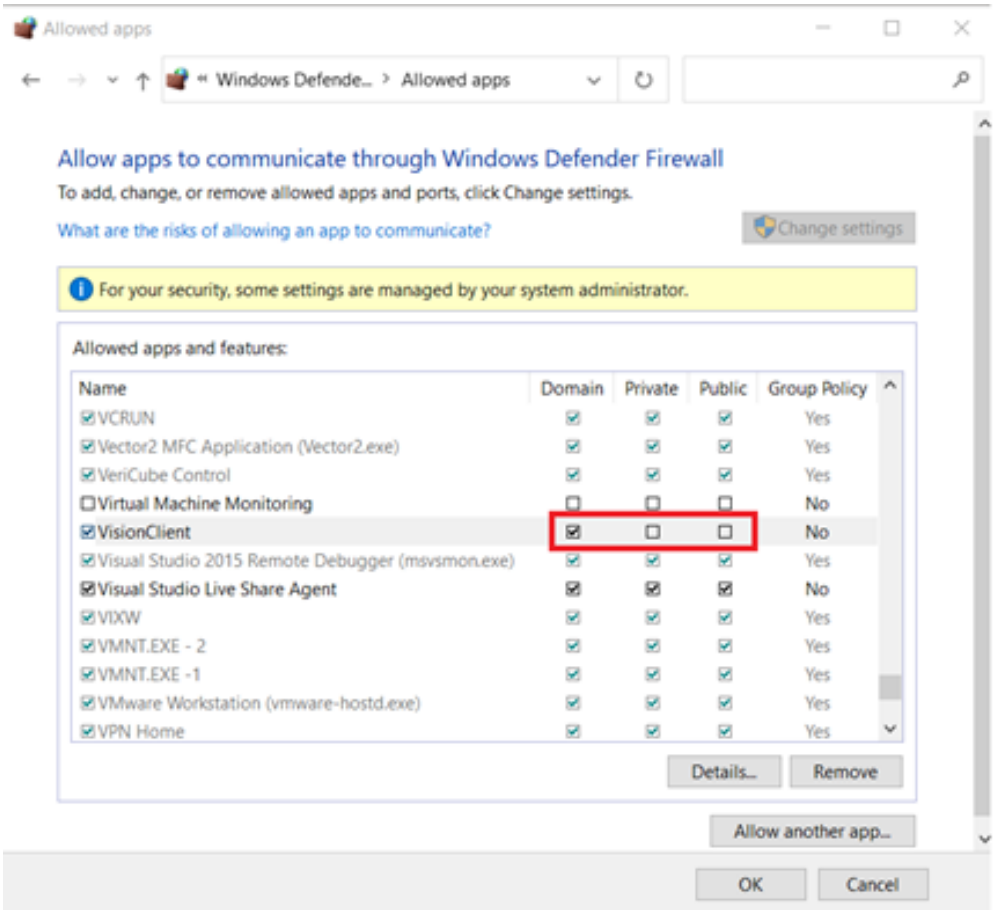
xx2200001057

次のページに続く

© Copyright 2025 ABB. All rights reserved.

考えられる原因

Windows ファイアウォールは、一部のネットワークで、カメラ機能関連のソフトウェア エンジンである **VisionClient.exe** をブロックします。 ユーザーは、許可されたアプリ ウィンドウで、コンピュータのネットワーク内で **VisionClient** が利用可能かどうかを確認する必要があります。**VisionClient** のすべてのネットワーク設定が選択されていない場合、この問題は次のような場合に発生する可能性があります。



xx2200001058

推奨される解決方法

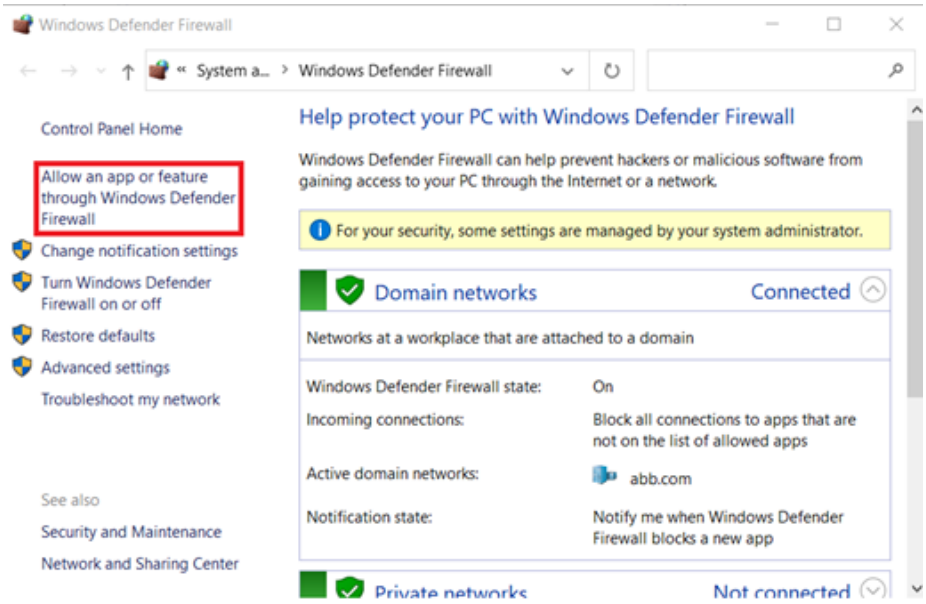
ファイアウォール設定を手動で変更するには、次の手順をお勧めします。

1. **Windows Defender ファイアウォール** を開きます。
2. **Windows Defender ファイアウォールを介した機能のアプリを許可する** をクリックして、**許可されたアプリ** ウィンドウを開きます。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

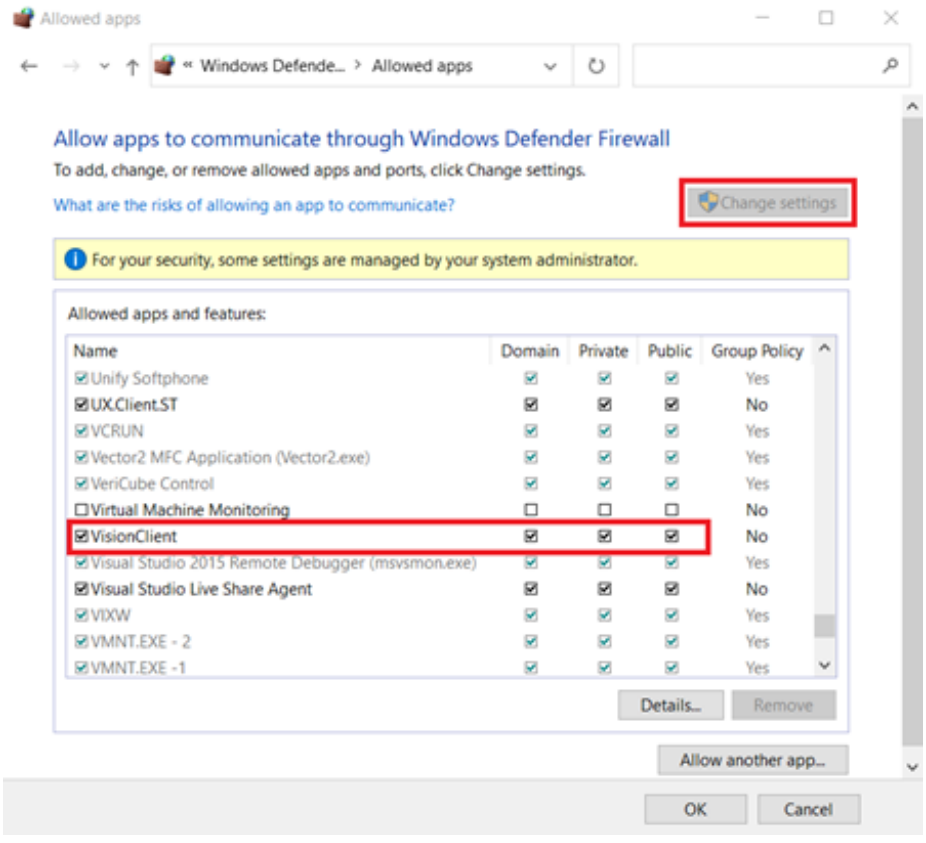
7.7.7 画像ダイアログを表示できません
続き



xx2200001059

3 **設定の変更** をクリックします。

4 リストで **VisionClient** を見つけ、すべての **VisionClient** または **visionclient.exe** アプリのすべてのネットワーク チェックボックスが選択されていることを確認します。



xx2200001060

5 **[OK]** をクリックします。

© Copyright 2025 ABB. All rights reserved.

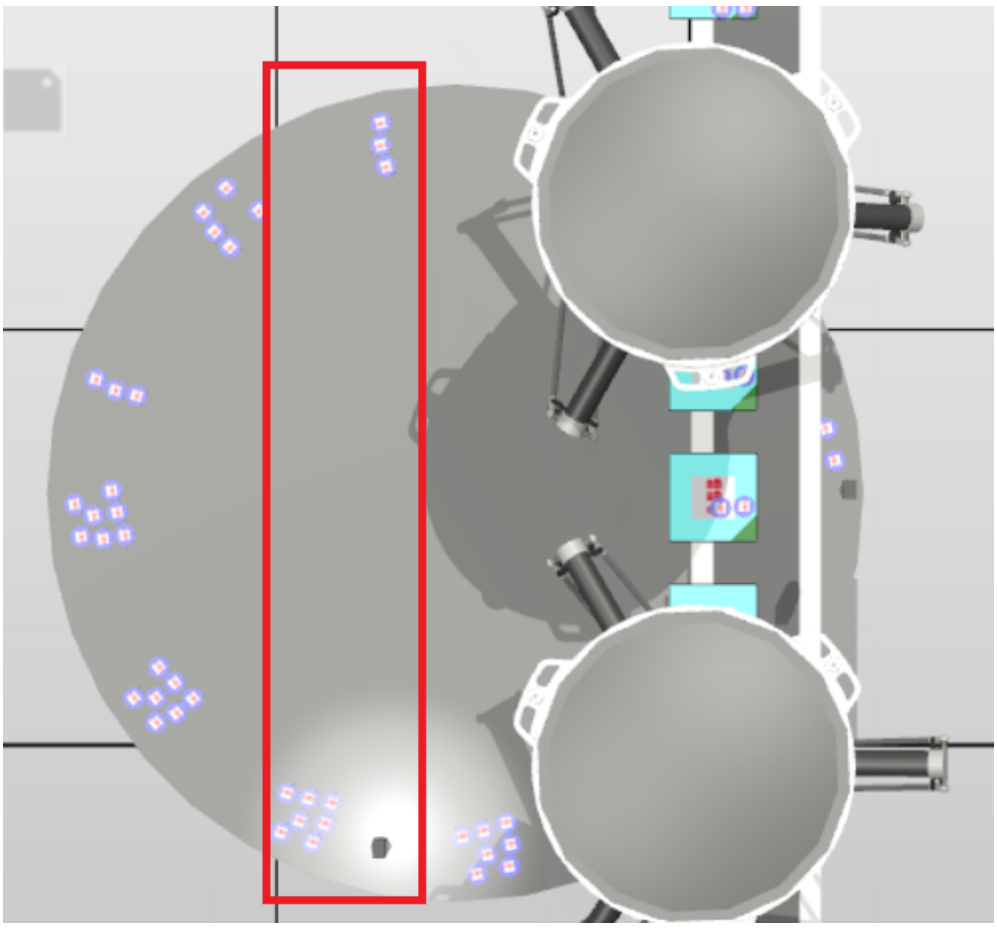
7.7.8 円形コンベアでカメラを使用すると、ロボットがアイテムを握らない

エラーの説明

円形コンベアでカメラを使用しているステーションで、ロボットがアイテムを握ることができず、「ツール 'PickPlaceTool_1' でアイテムを握ることができませんでした。無視しています... 」というエラー メッセージが表示されます。 生産を実行すると、ログに表示されます。

考えられる原因

ビジョン幅の有効化が有効でない場合、ビジョンスコープには以下の画像に示す赤丸表示のアイテムが含まれる。その場合、誤ったポジション情報がロボットに送られ、グリップエラーが発生します。



xx2200002019

円形コンベアでカメラを使用する場合は、ビジョン幅有効化でビジョンスコープを制限することが推奨される。そうでなければ、本ビジョンスコープは円形コンベアの反対側のアイテムを対象とします。

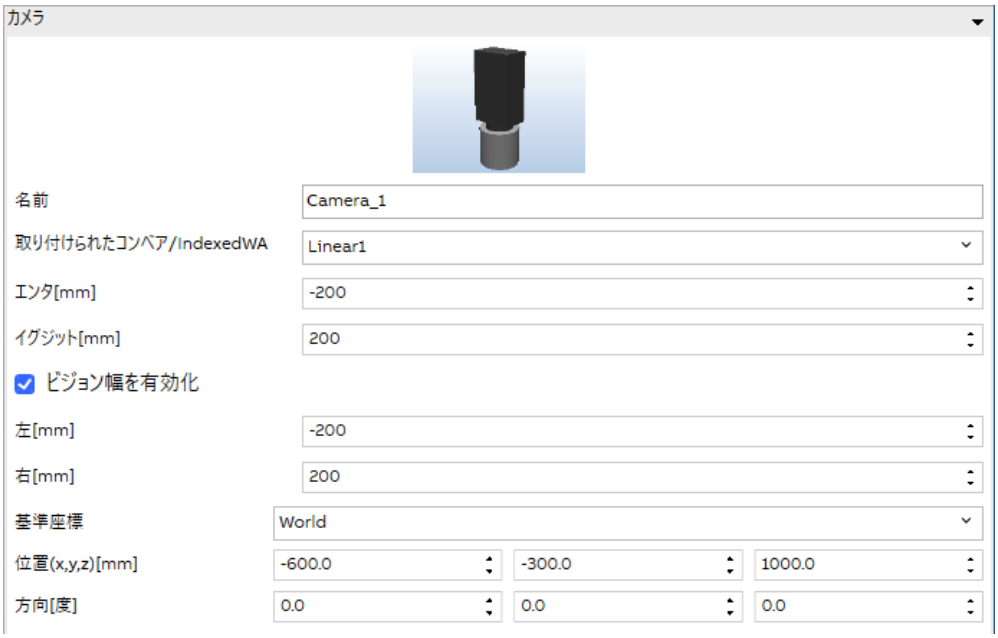
次のページに続く

© Copyright 2025 ABB. All rights reserved.

7.7.8 円形コンベアでカメラを使用すると、ロボットがアイテムを握らない
続き

推奨される解決方法

カメラが円形コンベアで使用されている場合は、カメラ設定ビューでビジョン幅を有効にするを有効にする選択をします。



xx2200002020

© Copyright 2025 ABB. All rights reserved.

7.7.9 開始ボタンを、インデックス作成の作業範囲でArm check point limitエラー発生後にクリックすると、ロボットが起動しない。

エラーの説明

インデックス作成の作業範囲を採用しているステーションで、ホールド解除ボタンをPackMLでクリックするか、開始ボタンをアームチェックポイント制限...のエラーメッセージがログに表示された後にクリックすると、ロボットの直接起動に失敗します。

考えられる原因

インデックスの作業範囲は本シナリオをサポートしない。

推奨される解決方法

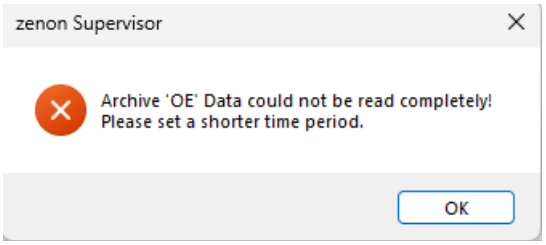
まずロボットを停止させ、ロボットの開始ボタンまたはホールド解除ボタンをPackMLでクリックします。

© Copyright 2025 ABB. All rights reserved.

7.7.10 アーカイブ 「OE」 データを読み取れませんでした

エラーの説明

「アーカイブ 「OE」 データを完全に読み取れませんでした。期間を短く設定してください」というエラー メッセージが表示されます。同時に、PMOP は拡張トレンド機能を使用してピッキング レートの履歴を正しく記録および表示できません。



xx2400001832

考えられる原因

コンピュータの使用可能なメモリ (RAM) が不足しています。コンピュータのメモリ使用量が限界に達しているか限界に近づいており、PMTW に十分なメモリがありません。

推奨される解決方法

推奨される対策は次のとおりです。

- 他のソフトウェアを閉じてメモリを解放します。
- コンピュータを再起動します。
- コンピュータに十分なメモリを追加します。

© Copyright 2025 ABB. All rights reserved.

8 予備部品

スペアパーツレベル

ABB スペアパーツは、L1、L2、L3 の 3 つのレベルに分類されます。スペアパーツのサービス作業を行う前に、必ずパーツ レベルを確認してください。

- L1スペアパーツ

  L1部品は現地で交換可能です。関連する製品マニュアルに記載されるメンテナンスおよび交換についての指示事項には、厳密に従ってください。なんらかの問題がある場合は、お近くのABBまでご連絡ください。

- L2スペアパーツ

  L2部品の交換には、特別なトレーニングが必要であり、特別なツールが必要になる場合があります。ABB現場サービス担当者、またはABBから訓練を受けた資格のある担当者のみがL2部品を交換できます。

- L3スペアパーツ

  L3 スペアパーツの交換または修理は、怪我や機器の損傷のリスクを軽減するため、アプリケーションの知識を持つ資格のある ABB サービス技術者のみが行うものとします。不適切な取り付けにより保証が無効になる場合があります。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

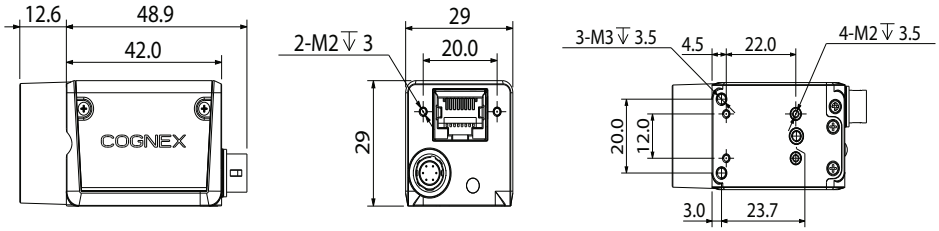
8.1 カメラ部品

スペアパーツ - PickMaster カメラ

スペアパーツ番号	説明	タイプ	スペアパーツレベル
3HAC092971-001	DSQC1106 PickMaster Twin 3 CAM	Cognex CIC-A02-CG-60-G	L1



xx2400001599



xx2400001600

Sony IMX430 CMOS センサーを搭載した Cognex CIC-A02-CG-60-G GigE カメラは、2.3 MP の解像度で毎秒 60 フレームを提供します。

次のページに続く

© Copyright 2025 ABB. All rights reserved.

スペアパーツ - PickMaster カム I/O ケーブル

スペアパーツ番号	説明	タイプ	スペアパーツレベル
3HAC072141-001	PickMaster カム I/O ケーブル	ケーブル	L1



xx2200000589

電源 I/O ケーブル HRS 6p/オープン、ツイスト、10 m - IO / 電源ケーブルは、Basler ace GigE カメラの光結合 I/O の電源供給とトリガー用に設計された 10 メートル長のケーブルです。

ケーブルはカメラ側にHRS 6ピンコネクタが付いています。 個々の要件に合わせてケーブルを短くできるように、もう一方の端は開いています。

配線情報：

ピン番号	電線色	Ace GigEg (GPIO なし)	Ace GigEg (GPIO 付き)	アビエイターCLランナー
1	茶	カメラの電源	カメラの電源	カメラの電源
2	桃	光絶縁 IN (Line1)	光絶縁 IN (Line1)	カメラの電源
3	緑	未接続	GPIO (Line3)	未接続
4	黄	光絶縁 OUT (Out1)	光絶縁 OUT	未接続
5	灰	光絶縁 I/O グランド	光絶縁 I/O グランド	カメラ電源 Groud?
6	白	カメラ電源接地	カメラの電源と GPIO 接地	カメラ電源 Groud?

次のページに続く

© Copyright 2025 ABB. All rights reserved.

スペアパーツ - PickMaster カムコムケーブル

スペアパーツ番号	説明	タイプ	スペアパーツレベル
3HAC072142-001	PickMaster カムコムケーブル	ケーブル	L1



xx2200000590

ケーブル GigE Cat 6、S/STP、1x ねじロック水平方向、DrC、20 m

カメラ側に水平固定ねじが付いた RJ-45 プラグ付き、長さ 20 メートルのデータ転送用 GigE ケーブル。

ツイストシールド ケーブルには、ホスト側に RJ-45 クリック ロック プラグがあり、ドラッグ チェーン アプリケーションに適しています。

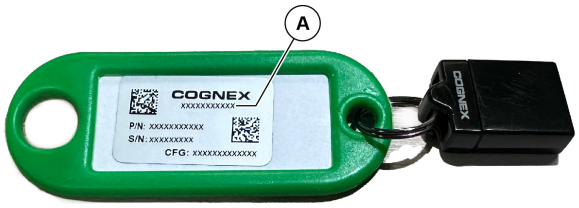
© Copyright 2025 ABB. All rights reserved.

8.2 USBドングル部品

スペアパーツ

スペアパーツ番号	説明	タイプ	スペアパーツレベル
3HAC092975-001	CVL9 Basic ドングル[i], DSQC1107	最大 2 台のカメラのビジョン ライセンス	L1
3HAC092982-001	CVL9 Advanced ドングル[i], DSQC1108	最大 10 台のカメラのビジョン ライセンス	L1

i ドングルは、ホスト コンピュータの任意の USB インターフェイスに接続できます。



xx2400001794

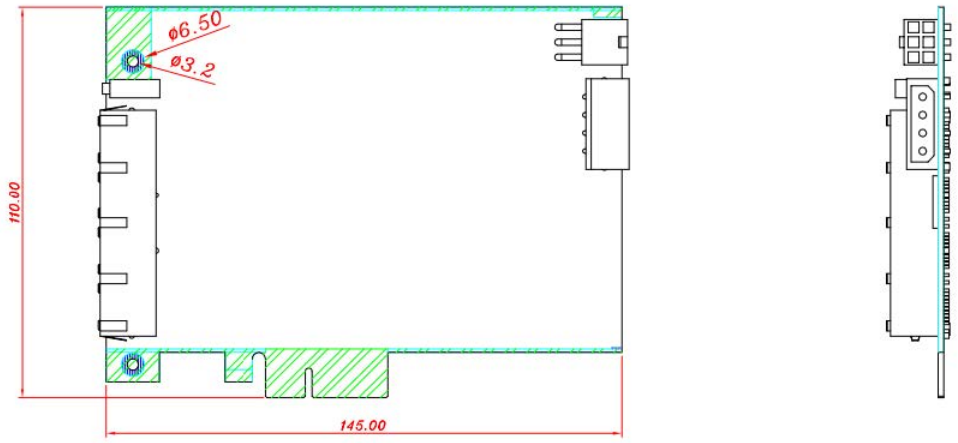
数字	タイプ番号	説明
A	CVL-MAX-WB	GigE Vision Ready Basic ドングル
	CVL-MAX-MULTI-WB	GigE Vision Ready Advanced ドングル
	CVL-MAX-SIM-WB	PickMaster Vision Sim. ドングル

© Copyright 2025 ABB. All rights reserved.

8.3 GigE ネットワーク カードの部品

スペアパーツ

スペアパーツ番号	説明	タイプ	スペアパーツレベル
3HAC093456-001	PMTW3 用 GigE カード	DSQC1109	L1



xx2400001751

仕様

パラメータ	説明
Ethernetポート	• Intel® Ethernet コントローラ I210-AT、9.5KB ジャンボ フレームと IEEE をサポート • すべての LAN は 10/100/1000Mbps をサポート
ポート コネクタ	• 8 ピン RJ-45
バス タイプ	• PCI Express x4
PoE 機能	• EEE 802.3at 準拠、各ポートは 54 VDC で最大 30W を供給
接続	• 4 ピン Molex および 6 ピン ATX 電源コネクタ
出力 PoE 電力	• 最大。 20W（PCIe スロット付き） • 最大 120W（4 ピン Molex または 6 ピン ATX 電源コネクタ付き）
寸法	• 145 mm x 110 mm
OS サポート	• Windows® 7/10/11（64 ビット）
環境	• 動作温度: 0°C ~ 60°C • 動作湿度 10% ~ 95% 結露なし • 保管温度: -20°C ~ 80°C

© Copyright 2025 ABB. All rights reserved.

9 回路図

9.1 回路図

概要

本書には回路図が含まれていませんが、登録ユーザーはmyABBビジネスポータル、*www.abb.com/myABB*で入手できます。

以下の表の品番を参照してください。

コントローラ

製品	回路図の品番
Circuit diagram - OmniCore C30、*Circuit diagram - OmniCore C30 for IRB 14050*、*Circuit diagram - OmniCore C30 for CRB 15000*	*3HAC059896-009*、*3HAC063898-009*、*3HAC072448-009*
Circuit diagram - OmniCore C90XT	*3HAC065464-009*
Circuit diagram - IRC5	*3HAC024480-011*

RobotWare オプション

製品	回路図の品番
Circuit diagram - PickMaster Twin	*3HAC024480-020*

© Copyright 2025 ABB. All rights reserved.

このページは意図的に空白のまま残しています

索引

6
6軸ロボット設定, 66

A
AckItmTgt, instruction, 387
Adjuster Interface,

C
CCD, 28
counts_per_meter
　システムパラメータ, 197, 219, 230, 257

D
data types
　itmtgt, 410
　selectiondata, 413
　sortdata, 416
dll
　ユーザーフック, 340
DSQC, 62

F
FlushItmSrc, instruction, 389
FOV, 28
functions
　GetQueueLevel, 406
　GetQueueTopLevel, 408

G
GetItmTgt, instruction, 390
GetQueueLevel, function, 406
GetQueueTopLevel, function, 408
Gigabit イーサネット
　説明, 25
　要件, 25

H
HSI：説明, 324

I
I/O
　構成, 185
　事前定義済み, 187–188
　取り付けられたボード, 62
　接続, 62
instructions
　AckItmTgt, 387
　FlushItmSrc, 389
　GetItmTgt, 390
　NextItmTgtType, 396
　QStartItmSrc, 398
　QStopItmSrc, 399
　ResetFlowCount ¶, 400
IPアドレス
　コントローラ, 51
itmtgt, data type, 410

M
max_dist, 65

N
NextItmTgtType, instruction, 396
noncnvwobjdata, 421

P
PatMax
　アドバンスモデル設定, 372
PickMaster Host
　定義, 17
PickMaster Operatorー
　定義, 17
PickMaster Recipe Manager
　定義, 17
PickMaster Runtime
　定義, 17
PickMaster パワーパック
　定義, 17
precision time protocol、PTP, 42
PROC.cfg、システムパラメータ, 66

Q
QStartItmSrc, instruction, 398
QStopItmSrc, instruction, 399

R
Rebalancing,
RIS
　ログ, 44
RobotWare, 64
RRG
　説明, 324

S
selectiondata, data type, 413
sortdata, data type, 416
sourcedata, 420
StartSig, 62
SyncSeparationフィルタ間隔, 482

U
UDP/IP, 42

ア
アイテム
　高さ, 371
　定義, 17
アジャスタ インターフェイス, 340, 342
アジャスタ インターフェース, 340, 343–344, 348

イ
イーサネットカード
　IPアドレス, 51
インターフェイスの初期化, 340

エ
エミュレーション
　定義, 18
エラーログ, 453

オ
オプション
　PickMaster, 44
オフラインシミュレーション
　定義, 18

カ
カメラ
　CCD, 28
　IP アドレス, 54
　センサーチップ, 28
　チェッカーボード, 282
　最大数, 25

© Copyright 2025 ABB. All rights reserved.

写真が撮られない, 479
接続, 59
要件, 28
カラースペース
構成, 327
説明, 324
カラービジョン
構成, 327
説明, 324
前提条件, 326
カラーフィルタリング説明, 324
カラーホイール, 324
カリパス
サブ検査モデル, 319

キ
キャリブレーション
キャリブレーション用紙, 283
チェッカーボード, 282
ラインコンベア, 195
円形コンベア, 226
結果, 287
用紙, 283
キャリブレーションホワイトバランス, 326
キューのアイドル信号, 131, 134, 187–189

ク
クワイエットシャットダウン, 44

グ
グリップのポジション
構成, 174
グレースケール, 324

コ
コンテナ
定義, 18
コントローラ
I/Oボード, 62
IPアドレス, 51
RobotWare, 64
設定, 64
コントローラネットワークアダプタ, 44
コンベヤの起動/停止信号, 131, 187–188

ゴ
ゴーストピッキング
定義, 18

サ
サブ検査モデル
カリパス, 319
ヒストグラム, 316
外部, 323
図形PatMax, 316
輪郭, 316

シ
シグナル
事前定義済み, 187–188
システムパラメータ
6軸ロボット設定, 66
counts_per_meter, 197, 219, 230, 257
I/Oの接続, 62
I/Oボード, 62
PROC.cfg, 66
について, 64
シミュレーションカメラ, 192

ス
スイッチ
ネットワーク, 52
ストロボ信号, 131, 135, 187–189

ズ
ズーム；イメージウィンドウ, 292

セ
センサーチップ, 28

ダ
ダイオード, 62

チ
チェッカーボードキャリブレーション, 282

ト
トラブルシューティング, 453
トリガ信号, 131, 135, 187–189

ネ
ネットワーク
イーサネットカード, 51
カメラ設定, 54
ケーブル, 52
コントローラ, 51
コントローラの構成, 51
コントローラの設定, 57
スイッチ, 52
ビジョンの構成, 54
ビジョンの前提条件, 52
構成, 51
通常の設定、コントローラ, 51
通常の設定、ビジョン, 57

ヒ
ヒストグラム
サブ検査モデル, 316

ビ
ビジョン インターフェース, 340
ビジョンカラー
カラービジョン, 324
ビジョンしすてむ
Gigabit イーサネット, 25
ビジョンシステム
カメラの最大数, 25
要件, 25
ビジョンモデル
コンセプト, 293
外部, 294
輪郭, 305
ビジョン モデル
Geometricモデル, 296
PatMax
PatQuick, 296
ビジョン高さ法, 372
ビュー フィールド, 28

ピ
ピックレート, 335

フ
ファイアウォール設定
UDP, 42

© Copyright 2025 ABB. All rights reserved.

プ
プロセス
　システムパラメータ, 66

ホ
ホワイトバランス
　キャリブレーション, 326

ポ
ポジション
　２度使用される, 482
　低い精度, 481

マ
マルチビュー, 372

モ
モノクローム, 324

ユ
ユーザー スクリプト
　インターフェイス, 342
ユーザーフック
　dll, 340

ラ
ライン
　キャリブレーション, 195

レ
レシピ
　定義, 17
レンズ
　計算, 29
　推奨事項, 28
　例, 29

ロ
ログ
　RIS, 44
　状態メッセージ, 44
ロボット
　起動, 335
　停止、一時停止, 335
　動かない, 480
ロボットコントローラ
　I/Oボード, 62
ロボット コントローラ
　設定, 64
ロボットの状態, 335

ワ
ワークエリア
　定義, 17

一
一時停止
　ロボット
　　再スタート, 335

円
円形
　キャリブレーション, 226

外
外部センサー
　設定, 129, 357
外部検査モデル, 323

関
関数
　GetFlowCount ¶, 409

警
警告
　4326, 476
　4327, 477
　4328, 477
　4329, 477

検
検査II, 312
検査モデル
　コンセプト, 312
　構成, 312

言
言語, 44

構
構成
　アイテムの高さ, 372
　カラースペース, 327
　カラービジョン, 327
　カリパスサブ検査モデル, 319
　グリップのポジション, 174
　コントローラネットワーク, 51
　システムパラメータ, 64
　ネットワーク, 51
　ビジョンネットワーク, 54
　ヒストグラムサブ検査モデル, 316
　外部サブ検査モデル, 323
　検査モデル, 312
　高さ設定, 372
　図形サブ検査モデル, 316
　製品の高さ, 372
　輪郭, 305
　輪郭サブ検査モデル, 316

高
高さの設定, 371
高さ法, 372

座
座標系
　ベース, 96
　ユーザー, 98
　意味, 95
　作業対象, 99
　世界, 97
　道具, 100
　変位, 100

再
再スタート, 335
　ロボット, 335
再バランス, 170
再起動
　非常停止, 335
再分散, 170

最
最大距離, 65

彩
彩度説明, 324

© Copyright 2025 ABB. All rights reserved.

使
使用可能なポジション信号, 131, 134, 187–189

時
時間同期サービス, 42

焦
焦点距離
　計算, 29
　説明, 28

状
状態メッセージ
　ログ, 44

色
色相説明, 324

信
信号
　キューのアイドル, 131, 134, 187–189
　コンベヤの起動/停止, 131, 187–188
　ストロボ, 131, 135, 187–189
　トリガ, 131, 135, 187–189
　構成, 185
　使用可能なポジション, 131, 134, 187–189
　接続, 62

製
製品
　高さ, 371

接
接続
　I/O, 62
　カメラ, 59

設
設定
　コントローラネットワーク, 57
　外部センサー, 129, 357

配
配信インターフェース, 340

非
非常停止, 335

標
標準
　IEEE 1588, 42

明
明度
　説明, 324

輪
輪郭
　コンセプト, 305
　サブ検査モデル, 316
　構成, 305

© Copyright 2025 ABB. All rights reserved.

ABB AB
Robotics & Discrete Automation
S-721 68 VÄSTERÅS, Sweden
Telephone +46 10-732 50 00

ABB AS
Robotics & Discrete Automation
Nordlysvegen 7, N-4340 BRYNE, Norway
Box 265, N-4349 BRYNE, Norway
Telephone: +47 22 87 2000

ABB Engineering (Shanghai) Ltd.
Robotics & Discrete Automation
No. 4528 Kangxin Highway
PuDong New District
SHANGHAI 201315, China
Telephone: +86 21 6105 6666

ABB Inc.
Robotics & Discrete Automation
1250 Brown Road
Auburn Hills, MI 48326
USA
Telephone: +1 248 391 9000

abb.com/robotics

3HAC092761-012, Rev B, ja

© Copyright 2025 ABB. All rights reserved.
仕様は事前の通知なく変更されることがあります。