

RCX3 シリーズ

RCX320

2 軸

RCX340/RCX341

3 軸～4 軸

[多軸ロボットコントローラ]



RCX320

対応軸数	2軸
運転方法	プログラム リモートコマンド オンライン命令
ポイント数	30000ポイント
入力電源	制御電源 単相AC200V～230V±10%以内 主電源 単相AC200V～230V±10%以内
原点復帰方式	アブソリュート インクリメンタル セミアブソ



RCX340/RCX341

対応軸数	3, 4軸
運転方法	プログラム リモートコマンド オンライン命令
ポイント数	30000ポイント
入力電源	制御電源 単相AC200V～230V±10%以内 主電源 単相AC200V～230V±10%以内
原点復帰方式	アブソリュート インクリメンタル セミアブソ

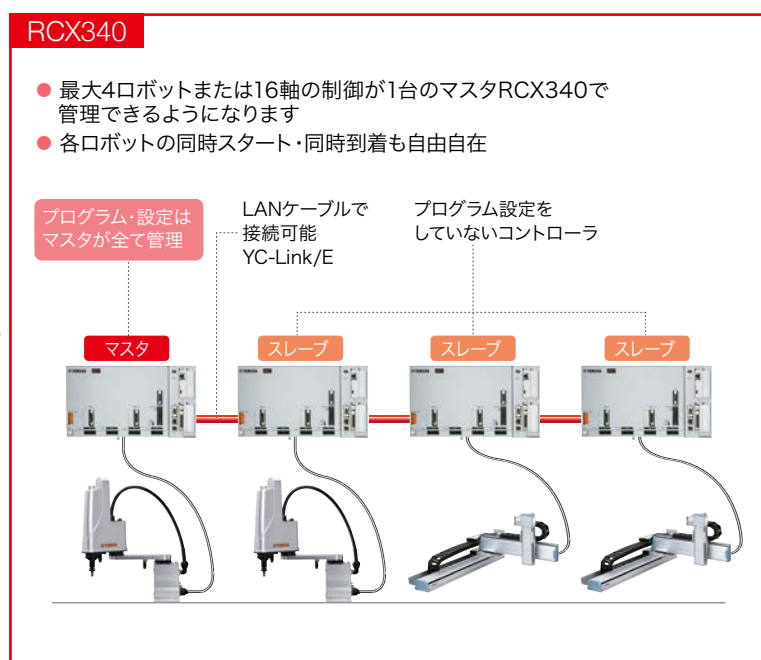
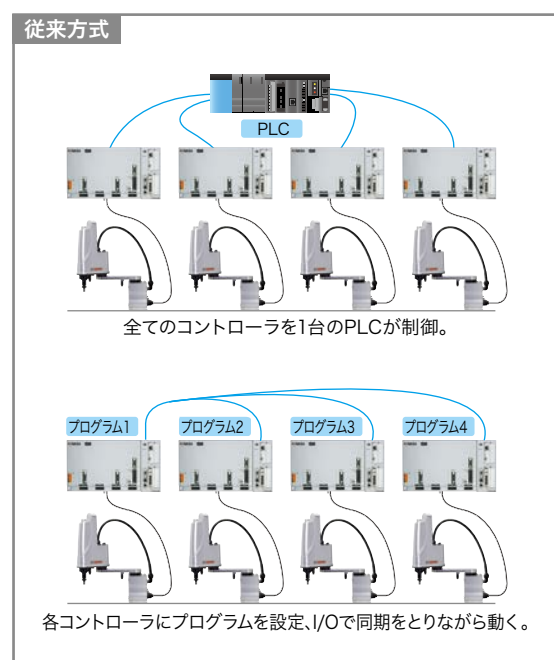
※ RCX341 は大型スカラロボット「YK1200XG」専用です。

ハイレベルな設備構築が実現できる高度な機能性

高速通信により複数台ロボットの同期動作が可能になりました。コントローラ間リンクによりプログラムは一台のコントローラに記載するだけで OK。新開発のアルゴリズムにより、位置決め時間の短縮や軌跡精度の向上も達成しました。

■ 複数台のロボットの制御を一台のマスタコントローラで管理可能

RCX340 コントローラは、コントローラ間的高速通信が可能となりました。マスタコントローラから各スレーブのコントローラに動作指令を出せるようになったため、プログラムやポイントは上位のマスタコントローラのみを管理するだけで OK です。また、マルチタスクにも柔軟に対応しているため、PLC を使うやり取りも簡素化できるため、システムをより簡単に低コストにて構築可能です。

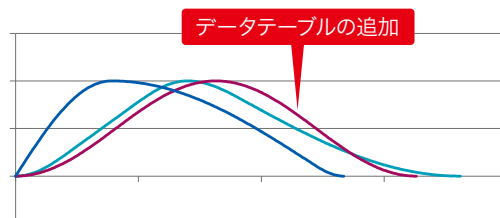


■ モーションの最適化

ロボットの性能を最大限に引き出すため、動作パターンに合わせたモーションの最適化をさらに強化しました。動作タイムの短縮や停止時の振動抑制など、より質の高いロボット動作を実現します。

■ 最適な加減速モーション

振動を抑えつつ高速な動作ができる加減速モーションを生成します。



■ モーション機能の大幅改善によるなめらかな動きの実現

新サーボ・モーションエンジン搭載により、さまざまな動作の連結が可能になりました。新開発のアルゴリズムにより、位置決め時間の短縮や軌跡精度の向上を達成しました。

■ CONTオプションの機能拡張

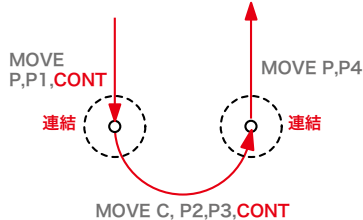
PTPと補間動作、コンペイトラッキングなど、異種動作の連結により高速化が図れます。

■ 連続動作の向上

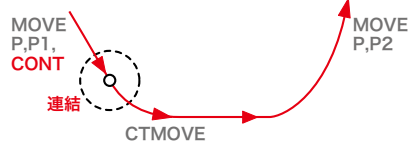
障害物の待避などを目的に動作途中で中継ポイントを経由する場合などCONTオプションを使用することで、各動作ごとに減速停止をせず滑らかに連結動作をすることが可能です。
動作の種類(PTP、補間動作)に関わらず連結が可能です。

異なる動作の連結が可能

【例:PTP↔CP】

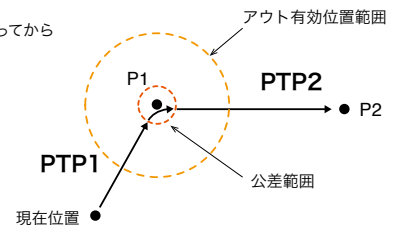


【例:PTP↔コンペイトラッキング】



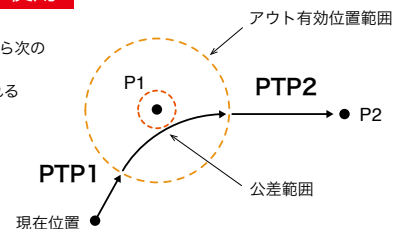
通常の移動

PTP1で減速し終わってから次の動作が始まる



CONTオプション使用

減速している途中から次の動作が始まるため、移動時間が短縮される



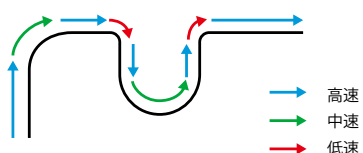
■ 用途による使い分け

連結動作では移動時間を優先し、直線部分と円弧補間部分で動作速度を変えて移動します。

PATHは事前に経路を登録することで、複雑な経路でも一定速度で動作させることが可能となり、さらに軌跡精度も向上します。シーリングなどの用途に最適です。

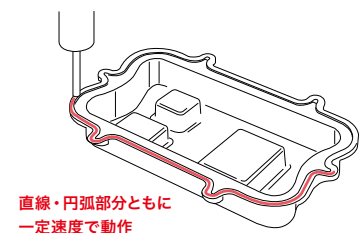
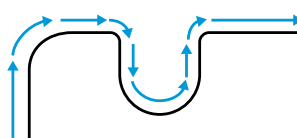
連結動作

直線部分と円弧部分で動作速度が異なる



PATH

全ての経路で速度を一定に保って移動する



直線・円弧部分ともに一定速度で動作

■ USB バックアップ可能な PBX 対応

プログラミングボックス「PBX」に対応しています。機能追加・修正作業が簡単で、プログラミング知識が無い方でも操作可能です。USB メモリにコントローラデータを保存する機能も搭載しています。操作メニューは日本語、英語、中国語に対応。



■ エラー状況がすぐにわかる

コントローラ前面にある「7 セグ LED ディスプレイ」に運転状況が表示されます。異常が発生するとエラーメッセージが表示されるため、プログラミングボックスを接続しなくてもエラー状況の把握がすぐに行えます。



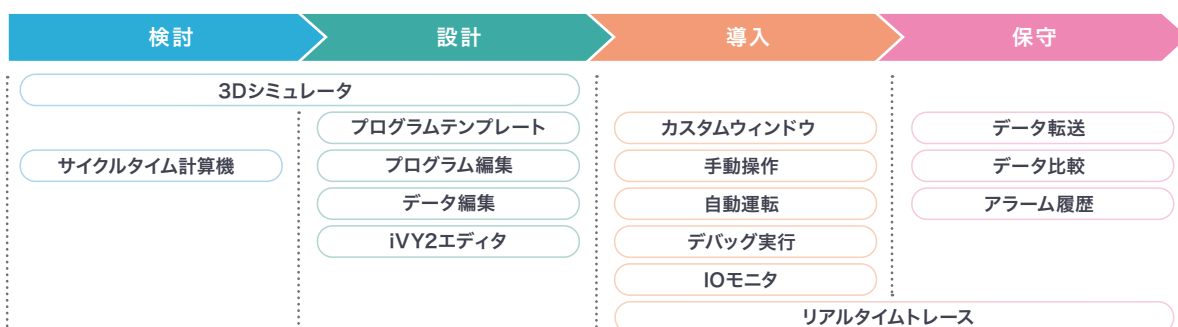
▲ 7 セグ LED ディスプレイ

■ 回生ユニットが内蔵 RCX340

回生抵抗 (RGU3 相当) が内蔵されているため既存のロボットと接続する場合、追加の回生ユニットは不要です。

サポートソフト「RCX-Studio 2020」対応

従来製品「RCX-Studio Pro」に 3D シミュレータ機能やプログラミングテンプレート（プログラム雛形自動生成機能）などの新機能を搭載し、よりユーザビリティを向上させました。

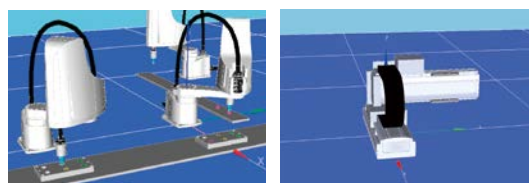


3Dシミュレータ機能

**実際のロボットが無くても
事前のレイアウト検証が可能**

ロボットと周辺機器を3Dで表示しパソコン上でロボットの動作をシミュレーションします（スカルロボットと直交ロボットに対応）。

- ▶ ロボットの配置検討やティーチング、デバッグなどが可能
- ▶ 設備稼働前にロボットと周辺機器の干渉チェックが可能



プログラムテンプレート機能（プログラム雛形自動生成機能）

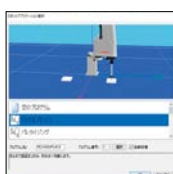
**プログラム作成時間の
大幅な短縮が可能**

10種類のアプリケーションのプログラムテンプレートを搭載しています。手順に従い操作していただくだけでプログラムの雛形が自動生成されます。

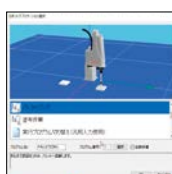


対応アプリケーション

- ピック&プレイス
- パレタイジング
- 塗布作業
- 実行プログラム切り替え
- コンベアトラッキング
- ビジョンによるパレットピッキング
- ビジョンによる塗布作業
- ビジョンによる掴みずれ補正
- ビジョンによる掴みずれ&搭載位置補正
- ビジョンによる掴みずれ&搭載位置補正（マスタなし）



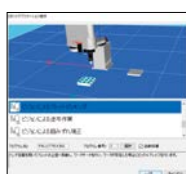
ピック&プレイス



パレタイジング



コンベアトラッキング



ビジョンによるパレットピッキング



ビジョンによる掴みずれ補正

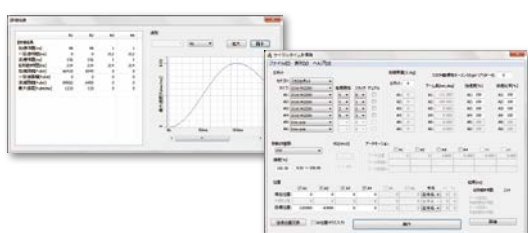


実行プログラム切り替え

プログラム自動変換機能

RCX240以前のコントローラプログラムをRCX3シリーズ用に変換

その他の機能



従来製品「RCX-Studio Pro」から継承された豊富な機能により、立ち上げから保守までヤマハロボットの運用をサポートします。

サイクルタイム
計算機能

リアルタイム
トレース機能

データ
比較機能

カスタムウィンドウ
作成機能

一段と充実した拡張性

RS-232C と Ethernet ポートを標準で装備。オプションで CC-Link、DeviceNet™に加え、EtherNet/IP™、EtherCAT といった高速・大容量の幅広いフィールドネットワークに対応します。汎用サーボアンプとの連結や、他社 VISION との連結も容易で、RCX320、RCX340 はまさに“つながるコントローラ”と呼べるでしょう。

コントローラ間通信

YC-Link/E

**RCX320、RCX340を
最大4台（最大制御軸数16軸）まで
接続可能**

より柔軟なロボット構成

プログラム作成がラク

複数台のロボットを集中管理

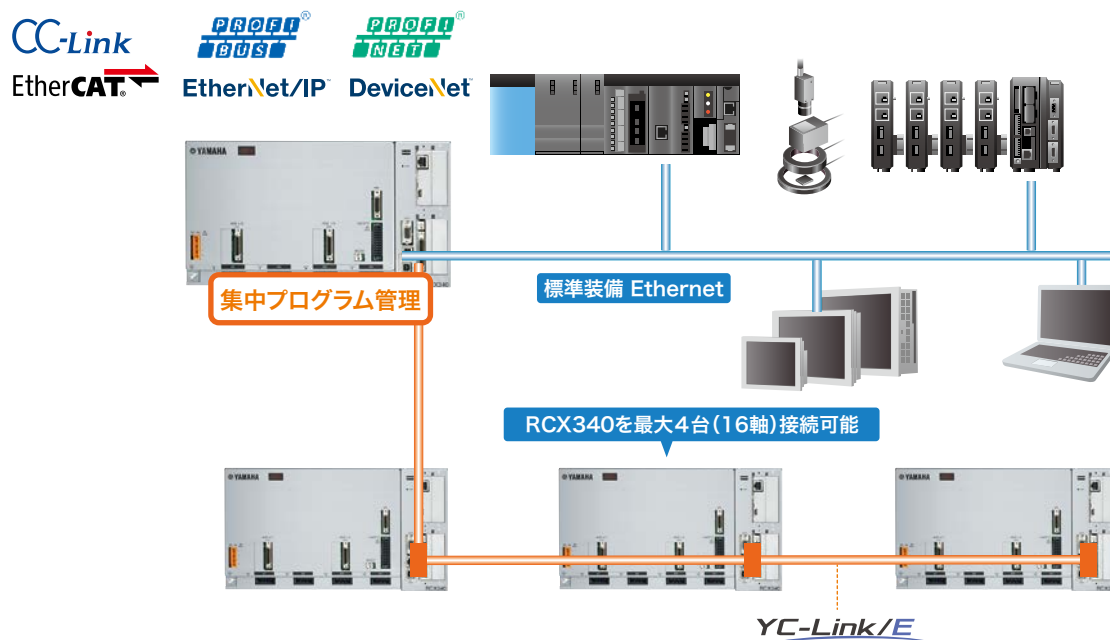
コスト削減

■ 多彩なフィールドバスに対応 / 最大4台接続でロボットを集中管理

RS-232C、Ethernet ポートを標準で装備。そのほか CC-Link、EtherNet/IP™、DeviceNet™、PROFIBUS、PROFINET※¹、EtherCAT など充実したフィールドバスに対応可能で、多種多様なデバイスとの接続、制御を行うことができます。5 軸以上の場合、YC-Link/E を使うことで RCX340 コントローラを最大 4 台接続できるので、複数台のロボットを集中管理できます。また YC-Link/E※²を使用すると、複数台のロボットをあたかも 1 つのコントローラで動かしているように扱うことが可能なため、ロボットのプログラム作成や管理が非常にラクに行えます。そのためセットアップに費やす人件費などの見えないコストの削減に貢献します。

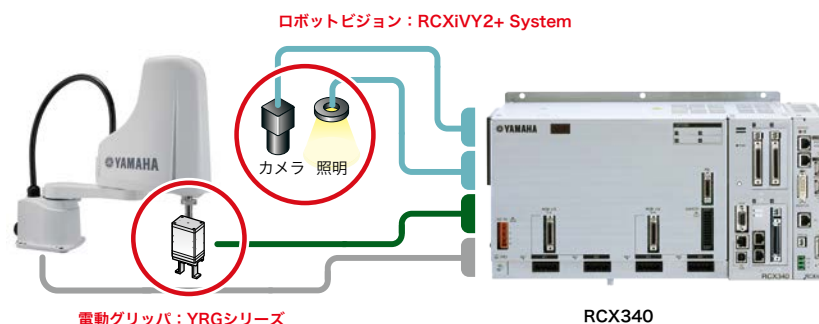
※ 1. PROFINET Ver.2.2 に対応

※ 2. YC-Link/E をご注文の際はどのロボットを何台目のコントローラに接続するかをご指定ください。



■ ロボットビジョン及び電動グリッパに対応

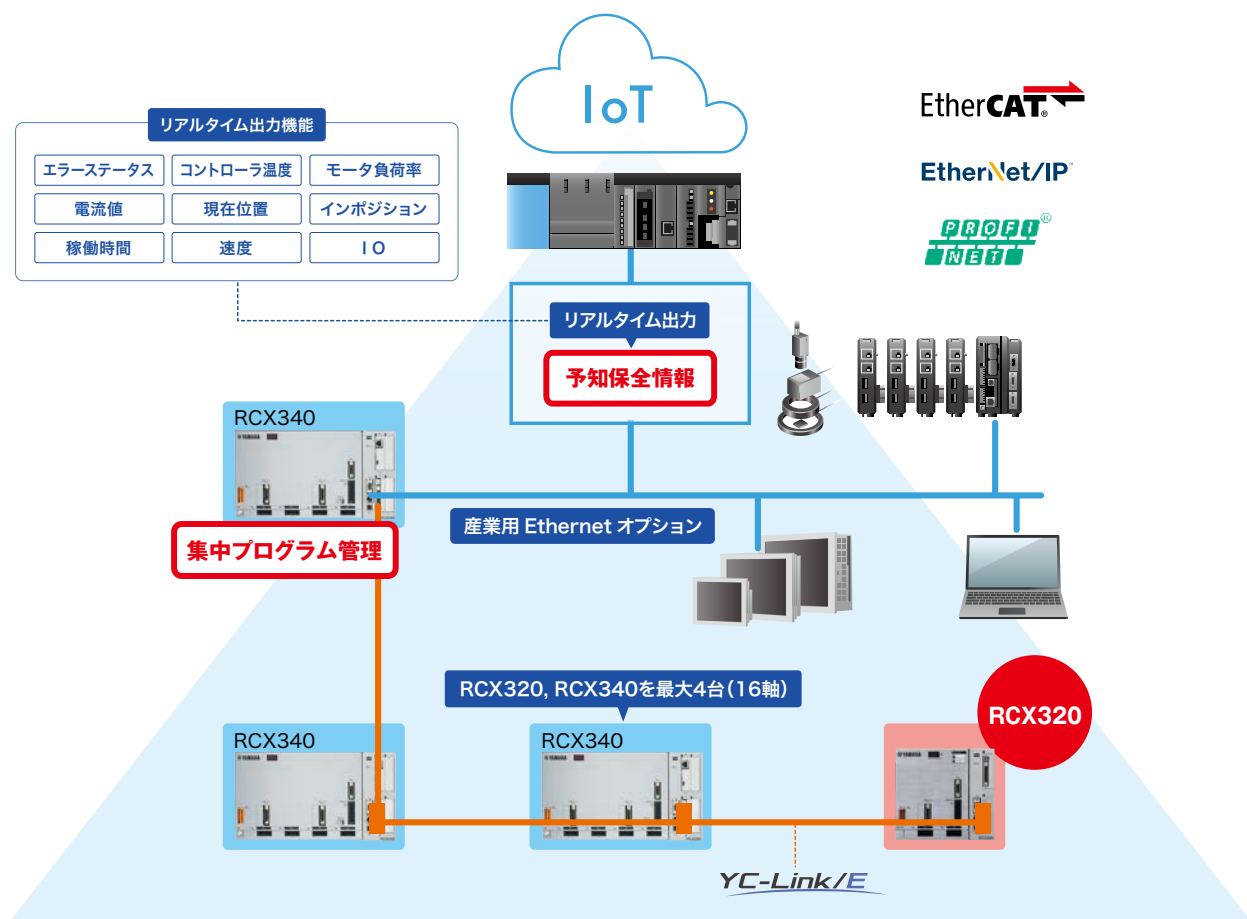
ロボット一体型ビジョン「RCXiVY2+」及び電動グリッパ「YRG シリーズ」に対応しています。制御は全てロボットコントローラ 1 台で可能です。PLC など上位装置とのやり取りが不要のため、セットアップや立ち上げが圧倒的に容易です。



予知保全情報のリアルタイム出力機能搭載

産業用 Ethernet オプション リアルタイム出力機能

産業用 Ethernet オプション（EtherNet/IP, EtherCAT, Profinet）を選択した場合、エラーステータス、現在位置、電流値、モータ負荷率、稼働時間など、予知保全に必要な情報をリアルタイムに出力し、“止まらない生産ライン”の実現に貢献します。



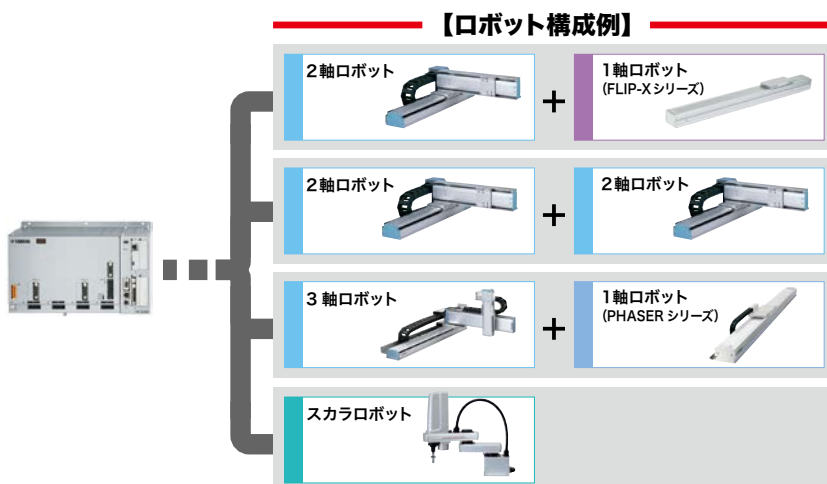
※ CC-Link、DeviceNet、PROFIBUS はリアルタイム出力機能非対応

RCX340 は単軸・直交・スカラ・P&P 全てに対応可能※

4 軸コントローラの RCX340 は、単軸・直交・スカラ・ピック&プレイスの全機種に対応しています。

ボールネジタイプの FLIP-X、リニアモータタイプの PHASER の混在制御も可能ですので、用途に合わせた自由な組み合わせが可能です。また、複数台のロボットの保守用としてご用意いただく場合も 1 台で OK。設定変更するだけでどの機種にもお使いいただけます。

※ 24V 仕様モデル、Robonity series を除く。



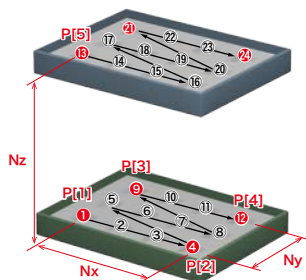
RCX コントローラの主な特長と機能

パレタイズをしたい

機能：パレタイズ

パレット上の4隅の座標値を入力し、縦方向・横方向の個数を指定することで各ポイントの座標値を自動で生成します。高さ方向の座標値・個数を指定することで立体的なパレットにも対応します。

定義できるパレットは最大40個ですが、4隅の座標値や各方向の個数はプログラムで変更することが可能なため、実質的にいくつでも対応できます。



- ・同時に使用できるパレット数：40
- ・2次元 / 3次元パレットに対応

サンプルプログラム

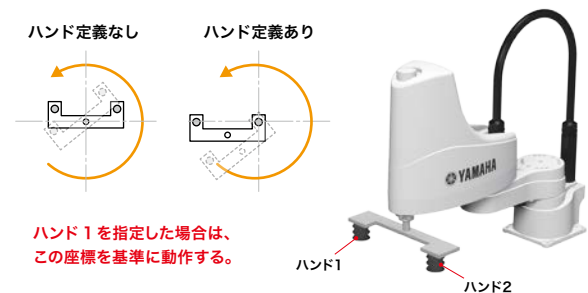
PDEF(1)=3,4,2,P3991 ... パレット定義 1 を P3991 ~ P3995 を使って、Nx : 4、Ny : 3、Nz : 2 に定義する

PMOVE(1,16),S=50 ... ロボットをパレット番号 1 の位置番号 16 のポイントに速度 50% で移動

ロボット先端からツールをオフセットさせたい

機能：ハンド定義

ロボットの先端軸にオフセットした状態でツールを取り付けたとき、オフセットしたツール先端の座標を基準にロボットを動作させるための機能です。特に複数のハンドがある場合や、スカラロボットや回転軸を含むロボットでツールを中心に回転させる場合に有効です。



- ・登録できるハンド：32
- ・R 軸がある場合の指定方法：1) +X 方向を基準にした場合の角度
2) ハンドの長さ
3) Z 軸のオフセット量

サンプルプログラム

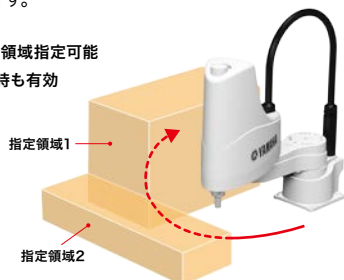
HAND H1= 0.000 150.000 0.000 R
HAND H2= -90.000 100.000 0.000 R
P1= 150.000 300.000 0.000 0.000 0.000 0.000
CHANGE H1 ... ロボット 1 のハンドデータをハンド 1 に変更
MOVE P,P1 ... ロボット 1 のハンド 1 の先端が P1 へ移動
CHANGE H2 ... ロボット 1 のハンドデータをハンド 2 に変更
MOVE P,P1 ... ロボット 1 のハンド 2 の先端が P1 へ移動
HALT

周辺装置との干渉を防止したい

機能：領域判定出力

あらかじめ登録した範囲内にロボットが入ると指定したポートに信号を出力します。装置内に干渉物がある場合ロボットの動作範囲を制限したい場合や、複数台のロボットがお互いに干渉するレイアウトで使用する場合などに便利な機能です。自動・手動の運転モードに関わらず機能しますので、ティーチング時のジョグ操作にも有効です。

- ・最大：32 領域指定可能
- ・手動運転時にも有効



- ・登録できる領域数：32
- ・自動運転中はもちろん、手動運転中も機能する

ワークを軽く押込みしたい

機能：トルク制限 (PUSH)

ワークの圧入などでモータのトルクや移動速度を制限して動作させることが可能です。指定した押付時間を経過しても目的位置まで移動が完了しなかった場合は動作を停止します。



- ・指定は軸単位
- ・押付力指定：定格推力に対する % で指定
- ・押付時間値：1 ~ 32767msec
- ・押付速度指定：1 ~ 100%
- ・STOPON 条件指定：条件成立で移動停止

サンプルプログラム

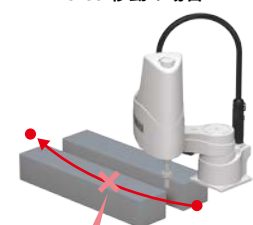
PUSH(3,P1),F=20,TIM=5000,S=10
... 第3軸をP0で指定された位置へ下記の条件で移動
押付力：定格推力の20%、押付時間：5秒、速度：10%
*押付力が20%に達した状態が5秒以上経過すると命令を終了する

指定した経路で動かしたい

機能：直線補間・円弧補間 (2D・3D)

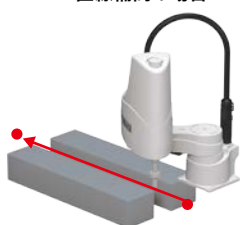
2次元・3次元の直線および円弧補間制御が可能です。シーリング作業の他、障害物の回避などで経路を指定して動作させたい場合などに有効です。

PTP移動の場合



PTPでは障害物にあたってしまう

直線補間の場合



- ・直線補間と円弧補間に対応
＜オプション＞
- ・SPEED：相対速度指定
- ・DSPEED：絶対速度指定
- ・VEL：直線速度指定 (mm/s で指定)
- ・STOPON 条件指定：条件成立で減速停止
- ・CONT 指定：次の移動命令と連結
- ・加速度、減速度指定
- ・ポート出力指定：指定距離移動後に信号を出力

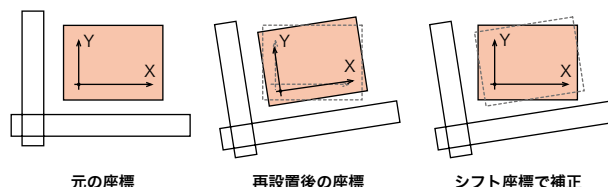
サンプルプログラム

```
MOVE L,P20      ... 現在位置からP20へ直線補間移動
MOVE C,P21,P22,P23,P20 ... P21, P22, P23, P20で構成する円弧補間移動
MOVE L,P24      ... P24に直線補間移動
```

ロボットを取り外したいが、再ティーチングはやりたくない

機能：シフト座標

メンテナンスなどでロボットの再設置や交換を行った場合、座標系にズレが生じることがあります。そのような場合はシフト座標機能を使用することで座標系の補正を行うことができますので、ポイントデータをそのまま利用可能です。再度ティーチングをする必要はありません。



- ・定義できるシフトの数：40

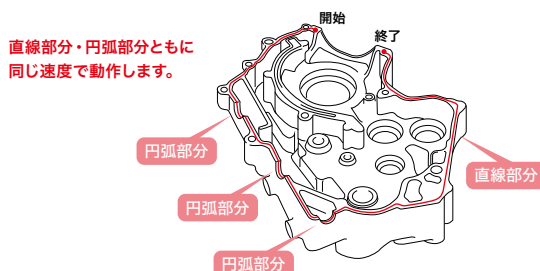
サンプルプログラム

```
S0= 0.000 0.000 0.000 0.000 ... S0のシフト座標を定義
S1= 100.000 200.000 50.000 90.000 ... S1のシフト座標を定義
P3= 100.000 ... P3のポイントデータを定義
SHIFT S0 ... シフト座標をS0に変更
MOVE P,P3 ... P3にPTP 移動
SHIFT S1 ... シフト座標をS1に変更
MOVE P,P3 ... P3にPTP 移動
HALT
```

一定速度でシーリングを行いたい

機能：PATH 文

シーリングでは経路の正確性と共に移動速度が一定であることが求められます。PATHは直線と円弧で構成された経路上を指定した速度で移動する機能で、移動中の速度変動が少ないためシーリング用途に最適です。経路の一部のみ速度を変更したり、移動中の任意の区間で指定ポートに信号を出力することができます。



直線部分・円弧部分ともに
同じ速度で動作します。

- ・指定した経路を「一定速度」で移動する
- ・あらかじめ「PATH SET、PATH、PATH END」で経路を指定しておいてから、「PATH START」で移動を開始する
- ・最大 1000 ポイントまで指定可能

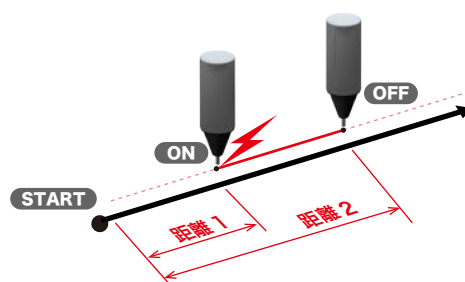
サンプルプログラム

```
PATH SET
PATH L,P1,DO(20)=1@10 ... 在位置からP1へ直線補間で移動中、
                        開始位置から半径10mmの位置でDO(20)に
                        1を出力するよう設定
PATH L,P2,DO(21)=1@12.5 ... P2へ直線補間で移動中、P1から半径12.5mm
                        の位置で DO(21)に1を出力するよう設定
PATH END
PATH START
```

シーリングの移動中に信号を出力したい

機能：通過点出力

シーリング時に吐出のON/OFFをさせる用途など、補間動作時に軸動作を止めることなく指定位置で汎出力のON/OFF制御が可能です。MOVEコマンド、PATHコマンドのどちらでも使用できます。



- ・小数点以下 3 桁まで指定可能 (mm)
- ・1 つの MOVE 文で 2 回まで指定可能

サンプルプログラム

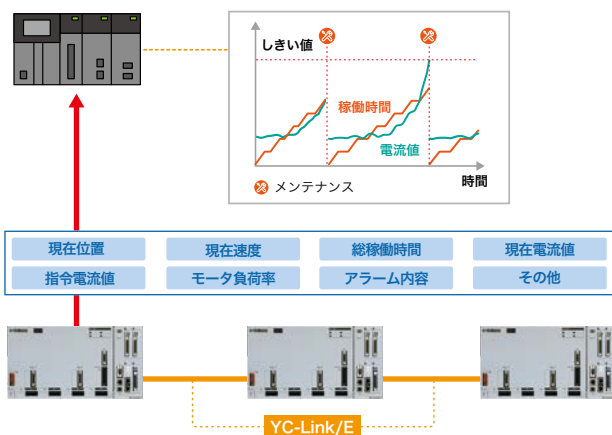
```
A1=10
B1=20
MOVE L,P1,DO(20)=1@A1,DO(20)=0@B1... P1へ移動開始後、10mm離れる
                        タイミングで DO(20) をオンし、
                        20mm離れるタイミングでDO(20)
                        をオフする
```

予知保全に必要な情報を出力したい

機能：リアルタイム出力

エラーステータス、現在位置、電流値、モータ負荷率、稼働時間など、予知保全に必要な情報をリアルタイムに出力可能です。

※産業用Ethernetオプション (EtherNet/IP、EtherCAT、Profinet) に対応。

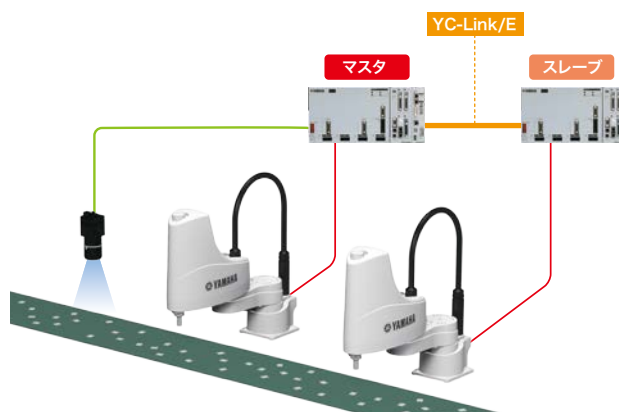


2台のロボットを効率的に動作させたい

機能：マルチタスク

ロボットと周辺機器など複数のタスク（最大16タスク）を同時に並行して実行することが可能です。タスクごとに優先順位をつけることも可能で、タスク起動中に優先順位の変更もできます。

コンベアトラッキング時にビジョンとロボットの動作を別タスクで同時に実行し、ロボット動作時も常にワークを監視するなどの用途に有効です。



- ・同時に実行できるタスク：16
- ・優先順位：1～64（高い～低い）

サンプルプログラム

プログラム名 <TRACK_MAIN>

START<CONV_SCAN>,T2

*CONVEYOR:

WHILE CCHKEQUE(1)=-1

CRMVQUE(1)

WEND

IF CCHKQUE(1)>0 THEN

(ロボットの動作ルーチン)

ENDIF

GOTO *CONVEYOR

プログラム名 <CONV_SUB>

CTVISION ON(1)

*SCAN:

VSEARCH 1,2,0

IF VGENCNT>0 THEN

FOR I%=0 TO VGETCNT-1

CADDQUEV 1,VGETPOS(I%),TG=I%

NEXT I

ENDIF

GOTO *SCAN

… サーチタスクの開始

… 作業エリアを通過したワークがなくなるまで繰り返す

… エリアを通過したワークの要素を削除

… 作業エリアに入ったら作業を開始

… ルーチンを繰り返す

… コンベア 1 でビジョン使用に切り替え

… サーチを行う

… ワークが検出された場合の処理

… サーチ結果を位置監視配列に追加

… 位置監視キューに追加

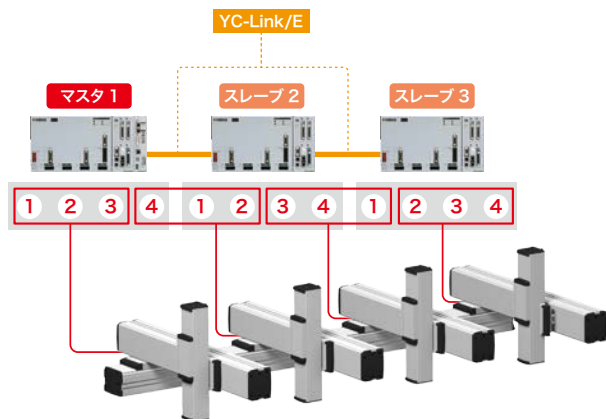
… サーチを繰り返す

複数のロボットで1台のコントローラで制御したい

機能：YC-Link/E

複数のRCXコントローラをリンクして1台のマスターコントローラで制御可能。単軸・直交・スカルロボットを混在させることも可能で、ネットワークボードやビジョンユニットなどは全てマスターコントローラの上に搭載します。

そのため1台のカメラ情報を複数のロボットで共有することも可能です。

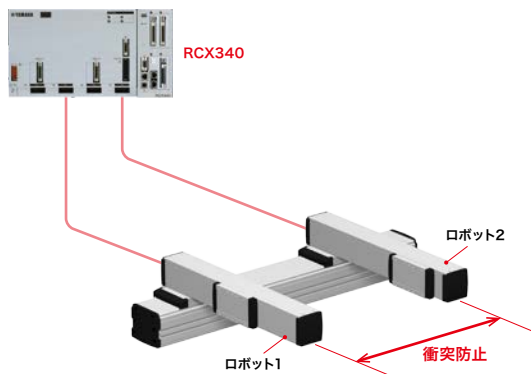


- ・最大でコントローラ 4 台まで接続可能
- ・RCX340 を使用した場合、最大 16 軸まで対応

1 台のコントローラで複数台のロボットを制御したい

機能：複数台ロボット設定

1台のコントローラの各軸を複数台のロボットに振り分けて設定可能です。RCX320は2軸、RCX340は4軸まで対応。さらに、複数台のコントローラをYC-LINK/Eで接続することで、最大でロボット4台・16軸まで設定できます。

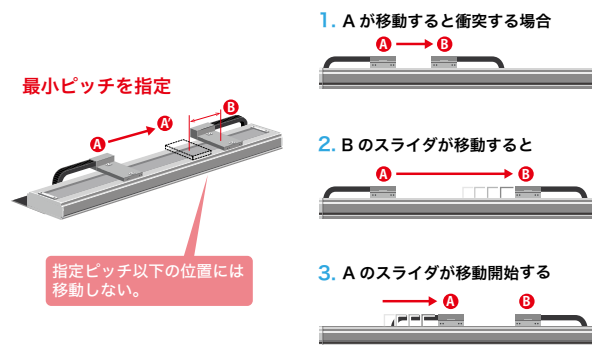


- ・MOVE[1]～MOVE[4] 命令で各ロボットごとに動作が可能。
- ・マルチタスクを使用することで、各ロボットのスムーズな連携も可能。

ダブルキャリアのロボットでパレットの干渉を防ぎたい

機能：衝突防止機能

ダブルキャリアのロボットで、両キャリアの衝突をコントローラ内の制御で防止します。ゾーン制御や外部センサーの設置などは不要です。キャリアよりも大きなパレットを搭載している場合は、キャリア間の最小距離をパラメータで設定可能です。



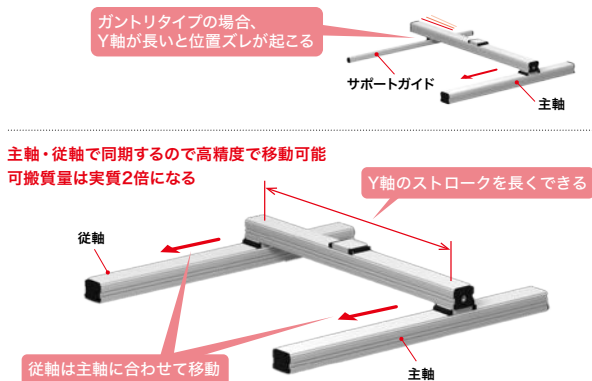
- ・RCX でダブルキャリアのロボットを動作させる場合に対応 (N15、N18、PHASER シリーズ)。

直交ロボットの Y 軸ストロークを長くしたい

機能：デュアルドライブ

同じタイプのロボット2台を同期制御する機能。主軸を動作させると従軸がそれに合わせて追従して動作します。

重量物の搬送や直交ロボットのY軸ロングストローク対応などに有効です。また、リニアモータなどダブルキャリアのロボットで2つのスライダを同期させることもできます。



- ・リジッドデュアル：主軸と従軸が高い剛性で連結されている
- ・フレキシブルデュアル：主軸と従軸が力干渉しない or 連結されていない
- ・タンデムデュアル：同じ軸上の2つのスライダを同期させる

動くものに追従しながらワークを取りたい

機能：コンベアトラッキング

コンベア上で動いているワークの動きに追従してピックすることができます。直線及び円弧のトラッキングに対応しており、エンコーダの入力信号を元に追従動作を行いますので、コンベア速度が変動した場合も追従可能です。ロボットビジョンによりサーチしたワークだけでなく、センサー信号入力によるトラッキングにも対応しています。



- ・ビジョントラッキング及びセンサトラッキングに対応
- ・エンコーダ接続可能台数：2 台
- ・対象エンコーダ：26LS31/26C31 相当ラインドライバ
- ・最高応答周波数：2MHz

タクトを上げたい

機能：搬送質量設定、アーチモーション、アウト有効位置

ワークのピック&ブレイスなどのタクトアップにはアーチモーションが有効です。Z軸の上昇・下降時の直線移動距離を指定することで、最適な移動パターンで動作させることができます。

また、アウト有効位置の値を大きくすると次の動作を実行するタイミングが早まり、動作時間の短縮に効果があります。

*ロボットは搬送質量を設定すると自動的に最適な加速度に設定されます（スカラロボットの場合は慣性モーメントの設定も可能）。



アウト有効位置：
軸先端がこの範囲に入ると次の動作を開始します。障害物回避などで中継ポイントを経由する際などは、この値を大きくすることで動作時間短縮が可能。
*値はプログラムで変更可能です。

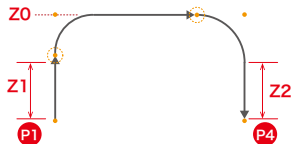
▶ 通常の移動



通常は、P1～P4を指定する。各動作はアウト有効位置の範囲に入ると、次の動作を開始する。

MOVE P,P2,CONT ... 現在位置からP2に移動
MOVE P,P3,CONT ... アウト有効位置に入ると停止せずにP3に移動
MOVE P,P4 ... アウト有効位置に入ると停止せずにP4に移動

▶ アーチモーション使用



アーチモーション動作時は、
・P1とP4のみ指定。
・移動中のZ軸高さを指定（Z0）
・上昇時、下降時の直線移動距離をそれぞれ指定（Z1、Z2）

A%=OUTPOS(3) ... アウト有効位置のパラメータをA%に代入
OUTPOS(3)=2000 ... アウト有効位置のパラメータを2000に変更
MOVE P,P4,A3=0.00{50.00,70.00}
... P4に移動する際、A3軸が0.00mmまで上昇。
A3軸は上昇時：50.00mm、下降時：70.00mm
直線移動する。
OUTPOS(3)=A ... アウト有効位置パラメータを元の値に戻す

精度を改善したい

機能：WAIT ARM、公差設定、加速度設定

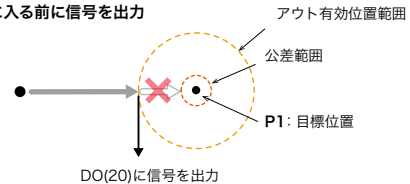
通常の移動命令ではアウト有効位置に入ると次の命令を実行します。動作時の位置決め精度が求められる場合は「WAIT ARM」を使用すると、公差範囲に入るのを待ってから命令を実行します。

また、公差範囲はプログラムで変更可能なので、移動命令ごとに異なる公差で移動させることもできます。

- ・ WAIT ARM
公差範囲に入ってから次の命令を実行する
- ・ TOLE
公差パラメータを設定/取得する

▶ 通常の移動

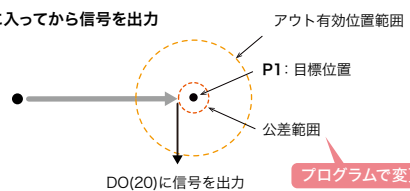
公差範囲に入る前に信号を出力



MOVE P,P1 ... P1に移動
DO(20)=1 ... アウト有効位置に入るとDO20に「1」を出力

▶ WAIT ARM使用

公差範囲に入ってから信号を出力



MOVE P,P1 ... P1に移動
WAIT ARM ... 公差に入るまで移動を継続
DO(20)=1 ... 公差範囲に入るとDO20に「1」を出力

プログラムで変更可能

回避ポイントで停止せずに動作させたい

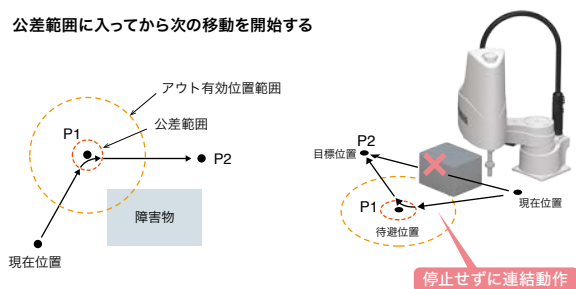
機能：CONT オプション

ロボットの移動経路上に障害物があり、それを回避するために待避ポイントを設定する場合は、移動命令にCONTオプションを使用することでよりスムーズな動作が可能になります。

通常のMOVEコマンドは各ポイントで位置決めを行います、CONTオプションを使用すると各動作が連結することで途中で停止することなく移動が継続されます。

▶ 通常の移動

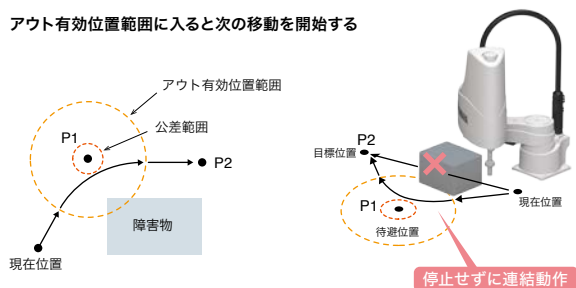
公差範囲に入ってから次の移動を開始する



```
MOVE P,P1 ... P1に移動し、移動軸が公差範囲に入ったら
MOVE P,P2 ... P2への移動を開始する
```

▶ CONTオプション使用

アウト有効位置範囲に入ると次の移動を開始する



アウト有効位置の場合

```
OUTPOS 10000 ... 全軸のOUTPOSパラメータを10000に変更
MOVE P,P1,CONT ... P1に移動し、移動軸がアウト有効位置範囲に入ったらP2への移動を開始する
MOVE P,P2
```

ロボットを2台使用してタクトアップしたい

機能：領域判定出力、内部出力変数

タクトアップを目的にロボットを2台使用してワークの搬送を行う場合など、領域判定出力を使用することで、お互いのロボットが干渉しないように動作させることが可能です。その際、内部出力変数 (MI, MO) を使用すると上位のPLCを介さずに高速で信号のやり取りをすることが可能です。

▶ 領域判定出力の設定



```
MO(20) ... ロボット1が領域に入ったらON
MO(40) ... ロボット2が領域に入ったらON
```

▶ プログラム例

プログラム名<ROB1_MAIN>

```
START <ROB2_SUB>,T2 ... サブタスクを起動
MOVE[1] P,P1,A3=0.00 ... 待機位置に移動
*LOOP1:
WAIT MO(50,40)=&B10 ... ロボット2が領域外に移動するまで待つ
MO(30)=0 ... 動作中フラグOFF
MOVE[1] P,P3 ... プレイス位置に移動
WAIT ARM[1]
MO(30)=1 ... 動作中フラグON
MOVE[1] P,P2 ... ピック位置に移動
WAIT ARM[1]
GOTO *LOOP1
```

プログラム名<ROB2_SUB>

```
MOVE[2] P,P11,A3=0.00 ... ロボット2を待機位置に移動
*LOOP2:
MO(50)=1 ... 動作中フラグON
MOVE[2] P,P12 ... ピック位置に移動
WAIT ARM[2]
WAIT MO(30,20)=&B10 ... ロボット1が領域外に移動するまで待つ
MO(50)=0 ... 動作中フラグOFF
MOVE[2] P,P13 ... プレイス位置に移動
WAIT ARM[2]
GOTO *LOOP2
```