

## ピック&プレース動作応用プログラム例

【関連取扱説明書】: プログラミングマニュアル、第1章 プログラム例

ピック&プレース動作を応用した実際のプログラム例を紹介します。

### 目次

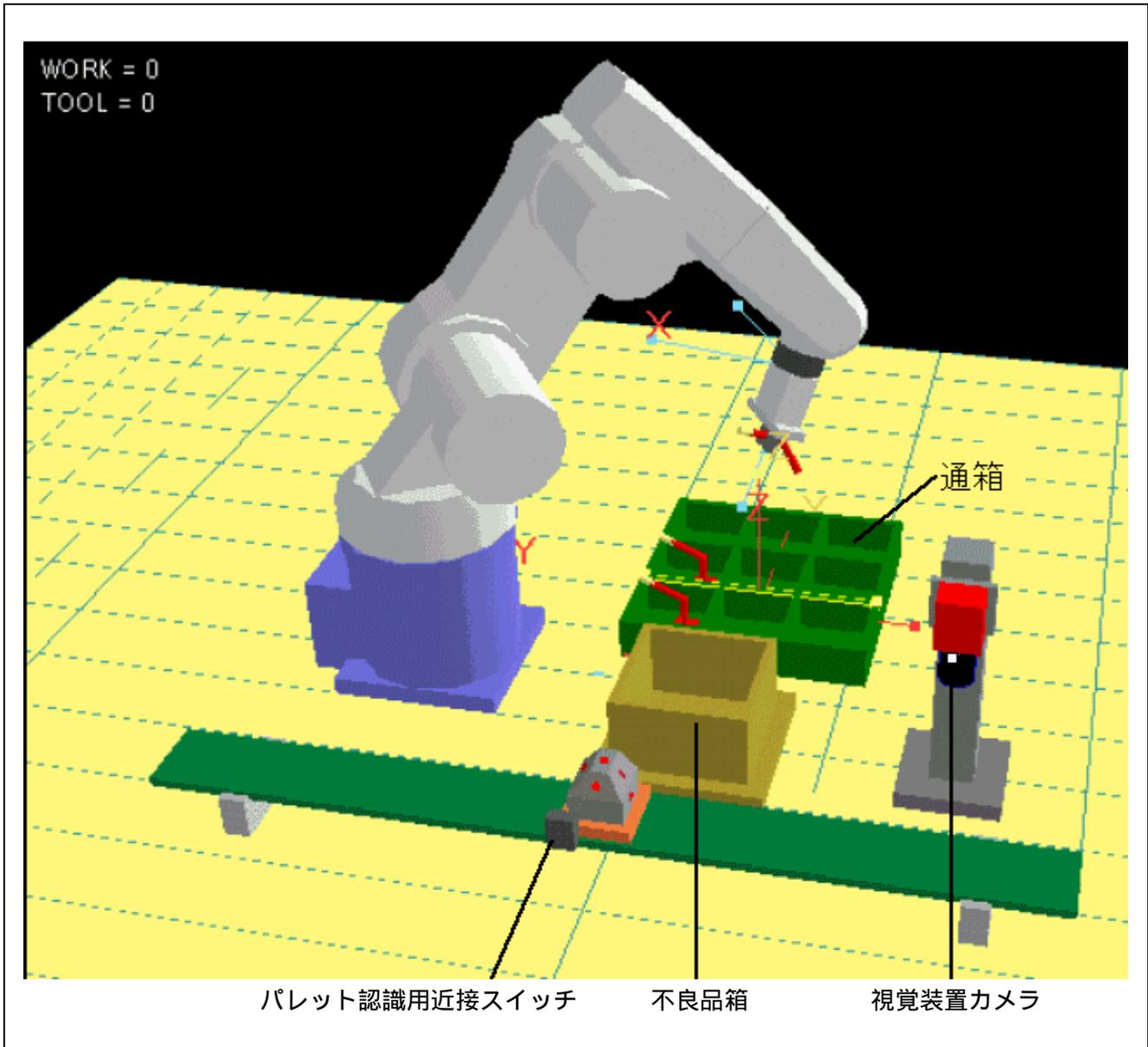
1. ピック&プレースの作業例.....	2
2. プログラムなどの定義.....	4
3. フローチャートとプログラム.....	6
3.1 メインプログラム (PRO11) のフローチャート.....	6
3.2 メインプログラム (PRO11) .....	7
3.3 チャック動作 (CHUCK) のフローチャートとプログラム.....	9
3.4 アンチャック動作 (UNCHUCK) のフローチャートとプログラム.....	9
3.5 原点復帰 (PRO2) のフローチャートとプログラム.....	9
3.6 パレタイジング (xdGetPalt) のフローチャート.....	9
3.6 パレタイジングプログラム (xdGetPalt) .....	10
3.7 品番挿入 (PRNUMIN) のフローチャート.....	12
3.8 品番挿入 (PRNUMIN) のプログラム.....	12
3.9 キャップ有無確認 (PRO1) のフローチャート.....	13
3.10 視覚プログラム: キャップ有無確認 (PRO1) .....	14
4. システム構成.....	16
4.1 システム構成図例と操作盤レイアウト例.....	16
4.2 ハード回路例.....	17
4.3 シーケンサ (三菱製 MELSEC-Q) のスロット割付例.....	21
4.5 シーケンサ (三菱製 MELSEC-Q) の I/O 割付の例.....	22
5. シーケンスプログラムの例.....	27

## ピック＆プレース動作応用プログラム例

## 1. ピック＆プレースの作業例

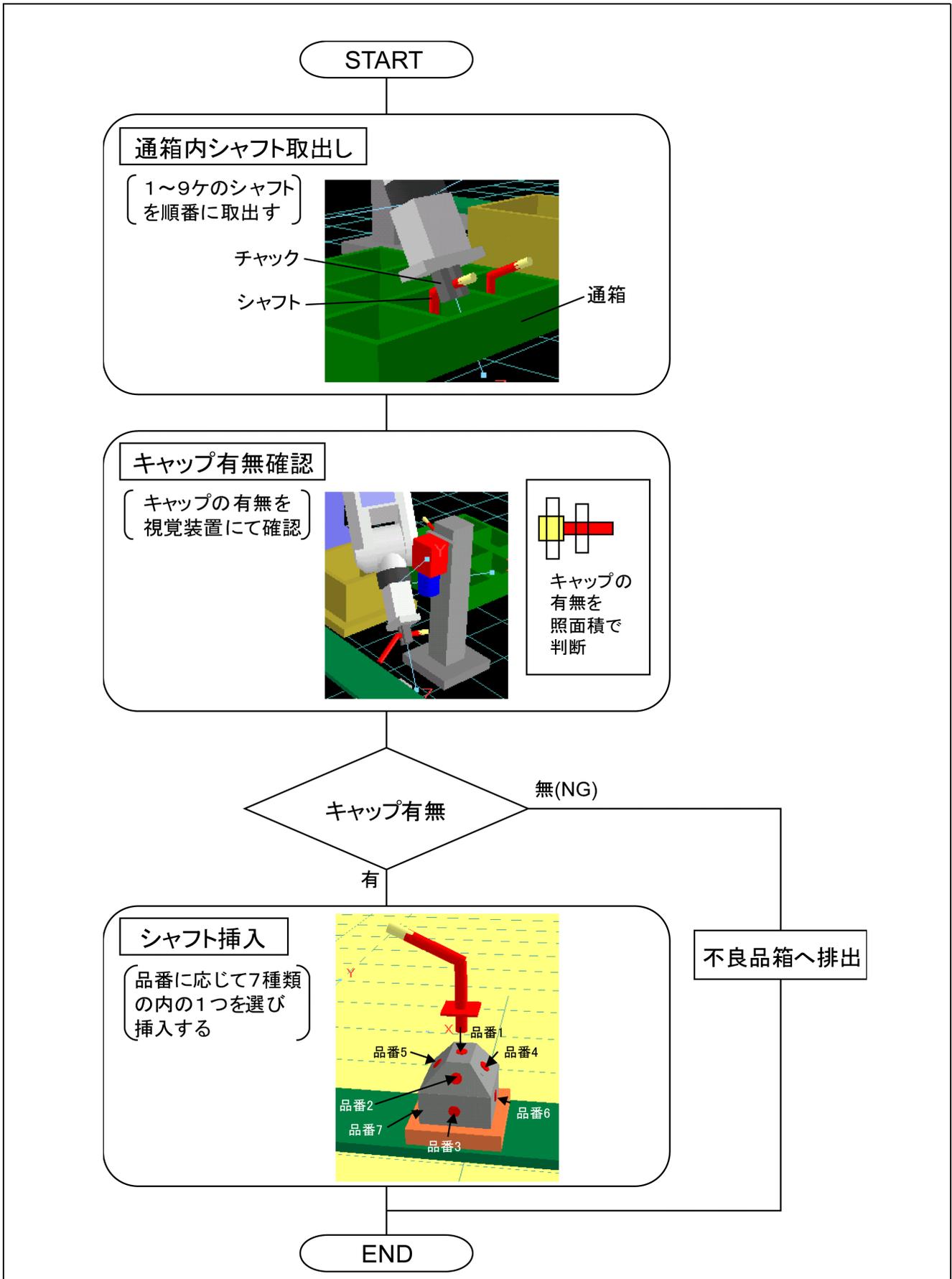
下図に示すように通箱内にあるシャフトを取り出し、視覚装置でキャップの有無を確認します。

キャップがない場合は不良品箱へ入れ、キャップがある場合はケースにシャフトを挿入します。



ピック＆プレース動作例

ピック & プレース動作応用プログラム例



ピック & プレース動作フロー

## ピック & プレース動作応用プログラム例

### 2. プログラムなどの定義

以下にプログラムなどの定義をします。

#### プログラムの定義

プログラム	工程および動作
PRO 2	原点復帰プログラム (作業原点に戻す)
PRO 11	メインプログラム
PRO 1	キャップ有無確認 (視覚 PRO)
CHUCK	チャック動作
UNCHUCK	アンチャック動作
PRNUMIN	品番挿入 (シャフトの挿入)
xdGetPalt	パレタイジングプログラム

#### 汎用出力の定義

汎用出力ポート	工程および動作
104	ロボット定位置
105	キャップ無し
106	通箱交換要求
107	品番要求
108	品番受取確認

#### 汎用入力 of 定義

汎用入力ポート	工程および動作
34	通箱交換完了
35	作業パレット有り
36	品番確認信号
37	品番OK
38	品番2 <sup>0</sup>
39	品番2 <sup>1</sup>
40	品番2 <sup>2</sup>
41	品番2 <sup>3</sup>
42	品番ストローク信号

#### ハンド出力の定義

ハンド出力ポート	工程および動作
64	チャックバルブ
65	アンチャックバルブ

#### 内部I/Oの定義

内部I/Oポート	工程および動作
231	画像入力完了
232	キャップ有り
233	キャップ無し
234	良否判定完了

# ピック & プレース動作応用プログラム例

## P型変数の定義

変数名	内容
P1	品番1シャフト挿入位置
P2	品番2シャフト挿入位置
P3	品番3シャフト挿入位置
P4	品番4シャフト挿入位置
P5	品番5シャフト挿入位置
P6	品番6シャフト挿入位置
P7	品番7シャフト挿入位置
P10	キャップ確認位置
P11	パレットの位置
P12	パレットの位置
P13	パレットの位置
P14	パレットの位置
P15	パレタイジング位置
P30	不良品排出位置

## I型変数の定義

変数名	内容
I1	内部スピード
I2	内部スピード
I3	内部スピード
I20	カウンタ
I50	品番

## J型変数の定義

変数名	内容
J1	作業原点

## ツール定義

TOOL 1

X=20

Y=0

Z=100

RX=0

RY=-45

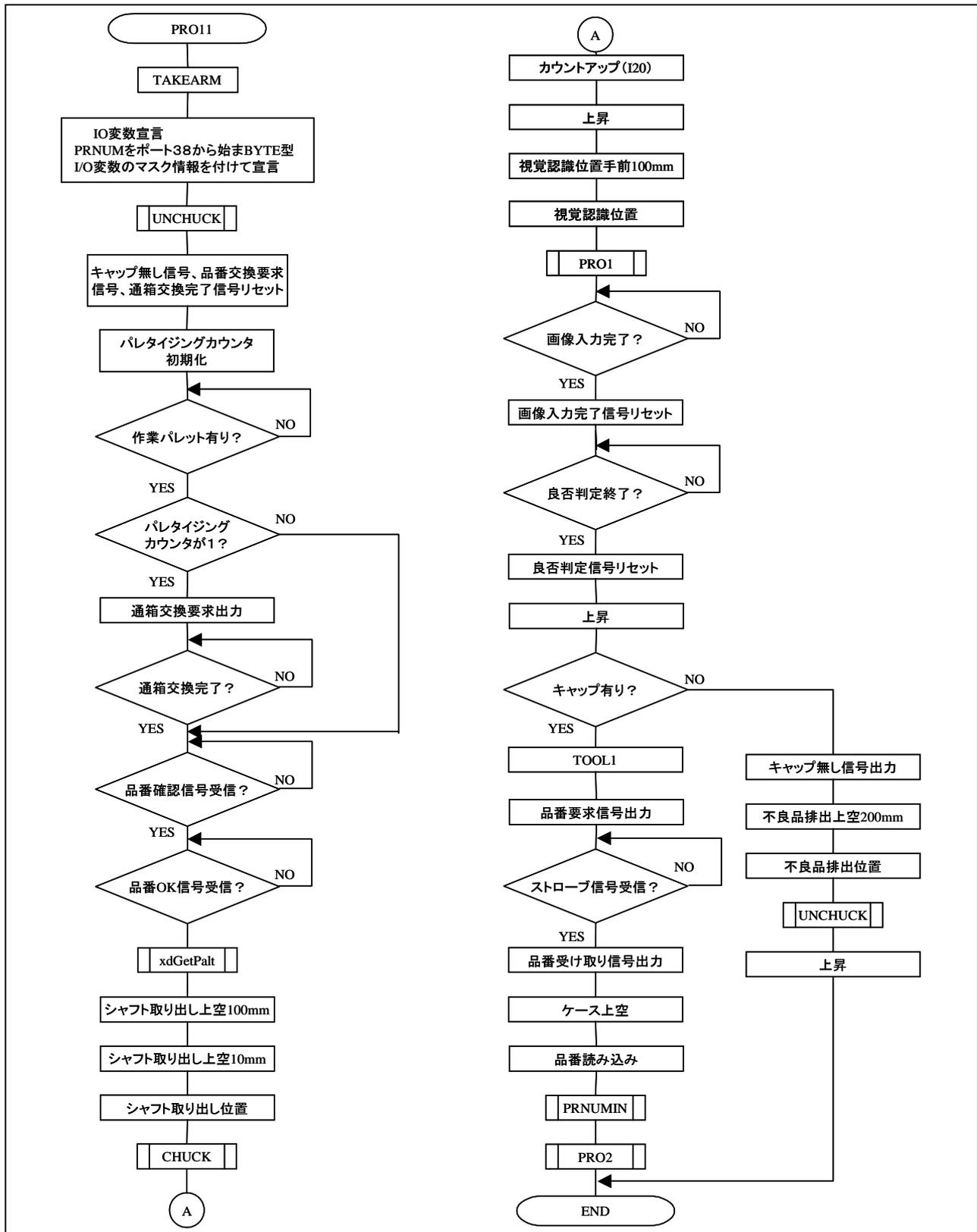
RZ=0

# ピック & プレース動作応用プログラム例

## 3. フローチャートとプログラム

以下にピックアップ&プレース動作例のフローチャートとプログラム例を示します。

### 3.1 メインプログラム (PRO11) のフローチャート



## ピック &amp; プレース動作応用プログラム例

## 3.2 メインプログラム (PRO11)

```

!TITLE "メインプログラム"
PROGRAM PRO11
TAKEARM
DEFIO PRNUM=BYTE,38,&B00001111          ' 品番

'----- 初期化 -----
CALL UNCHUCK                          'ハンド アンチャック
RESET IO[105]                          'キャップ無しポート OFF
RESET IO[107]                          '品番要求ポート OFF
RESET IO[108]                          '品番受取確認ポート OFF

IF I20 < 1 THEN                        'パレタイジング カマの初期化
    I20 = 1                             '
END IF                                  '
IF I20 > 9 THEN                          '
    I20 = 1                             '
END IF                                  '

'----- 作業パレット有無の確認 -----
WAIT IO[35] = ON

'----- 部品供給通箱の交換確認 -----
IF I20 = 1 THEN                        '通箱の交換 要
    SET IO[106]                        '通箱交換要求 出力
    WAIT IO[34]                        '通箱交換完了 入力
END IF

'----- 品番確認 -----
WAIT IO[36] = ON                      '品番確認信号 入力
IF IO[37] = ON THEN                  '品番OK

'==== ロボット動作 =====
'----- パレタイジング -----
CALL xdGetPalt(3,3,1,P11,P12,P13,P14,P15,I20,1)
APPROACH P,P15,@P 100,S=I1
APPROACH L,P15,10,S=I2
MOVE L,@E P15,S=I3
CALL CHUCK
    I20=I20+1
DRAW L,(0,0,100),S=I2

```

(次ページへ続く)

## ピック &amp; プレース動作応用プログラム例

(前ページから続く)

```

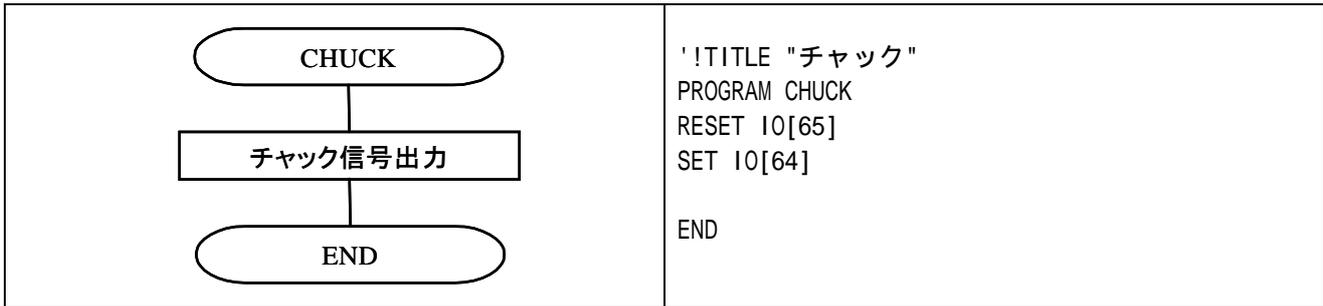
'----- キャップ確認 -----
APPROACH P,P10,100,S=I1
MOVE L,@E P10,S=I3
RUN PRO1
WAIT IO[231]=ON '画像入力完了待ち
RESET IO[231]
WAIT IO[234]=ON '良否判定終了待ち
RESET IO[234]
DEPART L,100,S=I1

'----- 良否プログラム -----
IF IO[232]=ON THEN 'キャップ有りの場合
  CHANGETOOL 1
'----- 品番確認 -----
SET IO[107] '品番要求
WAIT IO[42] 'ストローブ信号確認
IN I50=PRNUM '品番読み込み
SET IO[108] '品番受取確認
MOVE P,@P P20,S=I1 'ケース位置上空
CALL PRNUMIN '品番挿入プログラムへ
ELSE 'キャップ無しの場合
  IF IO[233]=ON THEN
    SET IO[105] 'キャップ無し
    APPROACH P,P30,@P 200,S=I1 '不良品排出
    MOVE P,P30,S=I2
    CALL UNCHUCK
    DEPART L,200,S=I1
  ENDIF
ENDIF
CALL PRO2 '原点復帰
END IF
END

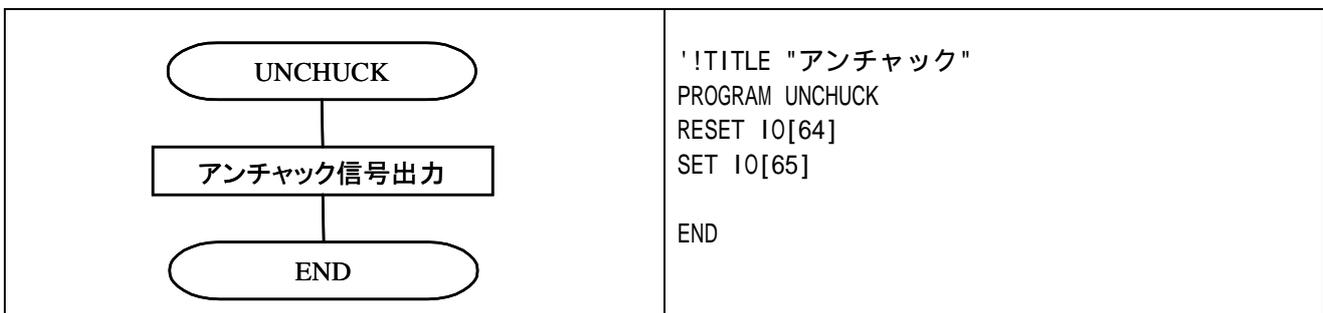
```

## ピック & プレース動作応用プログラム例

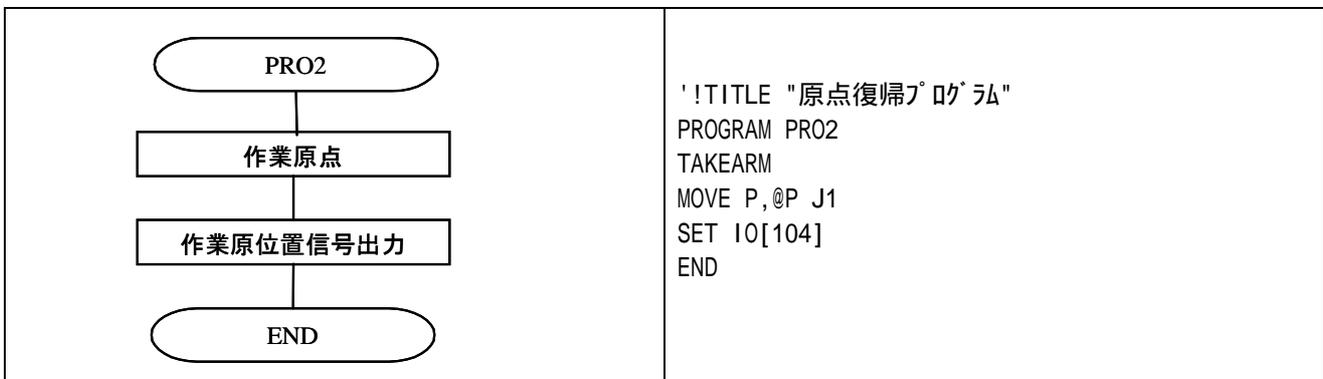
### 3.3 チャック動作 (CHUCK) のフローチャートとプログラム



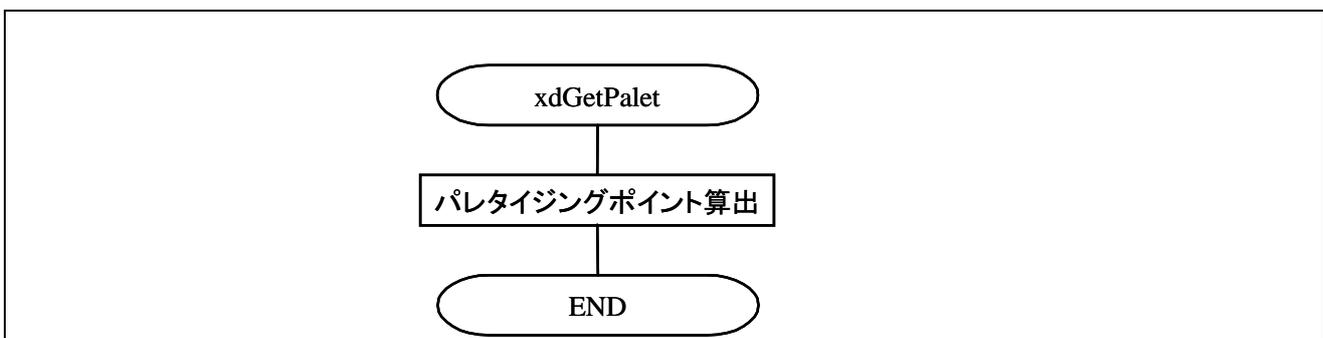
### 3.4 アンチャック動作 (UNCHUCK) のフローチャートとプログラム



### 3.5 原点復帰 (PRO2) のフローチャートとプログラム



### 3.6 パレタイジング (xdGetPalet) のフローチャート



## ピック &amp; プレース動作応用プログラム例

## 3.6 パレタイジングプログラム (xdGetPalt)

```

!TITLE   "簡易パレタイジング (4軸/6軸共用)"
!AUTHOR  "株式会社 デンソーウェーブ"
!ARG 1,  "横分割数N(I型)", 1, "I1"
!ARG 2,  "縦分割数M(I型)", 1, "I2"
!ARG 3,  "段積高さH(F型)", 1, "F1"
!ARG 4,  "パレット4隅P1(P型)", 1, "P1"
!ARG 5,  "パレット4隅P2(P型)", 1, "P2"
!ARG 6,  "パレット4隅P3(P型)", 1, "P3"
!ARG 7,  "パレット4隅P4(P型)", 1, "P4"
!ARG 8,  "動作ポイント(P型変数)", 1, "P5"
!ARG 9,  "パレタイジングカウンタ(I型)", 1, "I3"
!ARG 10, "段積カウンタ(I型)", 1, "I4"

#pragma optimize("ac", on)

PROGRAM xdGetPalt (n%,m%,h3!,pp1 AS POSITION,pp2 AS POSITION,pp3 AS POSITION,pp4 AS POSITION,nextpos
AS POSITION,count%,kcount%)
DIM m_vec1 AS VECTOR,m_vec2 AS VECTOR,m_pich1 AS VECTOR,m_pich2 AS VECTOR,mn_vec AS VECTOR

' 引数 N,M 異常チェック
if n <= 0 then
  printmsg "分割数nを1以上にしてください"+chr$(10)+" n値 "+str$(n),2,"パレタイジング 異常"
  stop
end if
if m <= 0 then
  printmsg "分割数mを1以上にしてください"+chr$(10)+" m値 "+str$(m),2,"パレタイジング 異常"
  stop
end if

' カウンタ値 異常チェック
if count <= 0 then
  printmsg "パレタイカウンタが0以下です"+chr$(10)+"カウンタ値 "+str$(count),2,"パレタイジング 異常"
  stop
end if
if count > (m*n) then
  printmsg "パレタイカウンタが範囲オーバーです(1~"+str$(m*n)+")"+chr$(10)+"カウンタ値 "+str$(count),2,"パレタイジ
ング 異常"
  stop
end if
if kcount <= 0 then
  printmsg "段積みカウンタが0以下です"+chr$(10)+"カウンタ値 "+str$(kcount),2,"パレタイジング 異常"
  stop
end if

```

(次ページへ続く)

## ピック &amp; プレース動作応用プログラム例

(前ページから続く)

```

m_vec1=PVEC(pp2)-PVEC(pp1)
m_vec2=PVEC(pp4)-PVEC(pp3)
if m >= 2 then                                ' 分割数mが一般的な値( >= 2)の場合
  m_pich1=m_vec1/(m%-1)
  m_pich2=m_vec2/(m%-1)
else                                           ' 分割数mが1の場合
  m_pich1=(0,0,0)
  m_pich2=(0,0,0)
end if

next_m%=(count%-1) MOD m%
next_n%=(count%-1) / m%

nextpos=pp1
mn_vec=((m_pich2*next_m%)+PVEC(pp3))-((m_pich1*next_m%)+PVEC(pp1))
if n >= 2 then                                ' 分割数nが一般的な値( >= 2)の場合
  LETP nextpos=((m_pich1*next_m%)+PVEC(pp1))+((mn_vec/(n%-1))*next_n%)
else                                           ' 分割数nが1の場合
  LETP nextpos=((m_pich1*next_m%)+PVEC(pp1))
end if

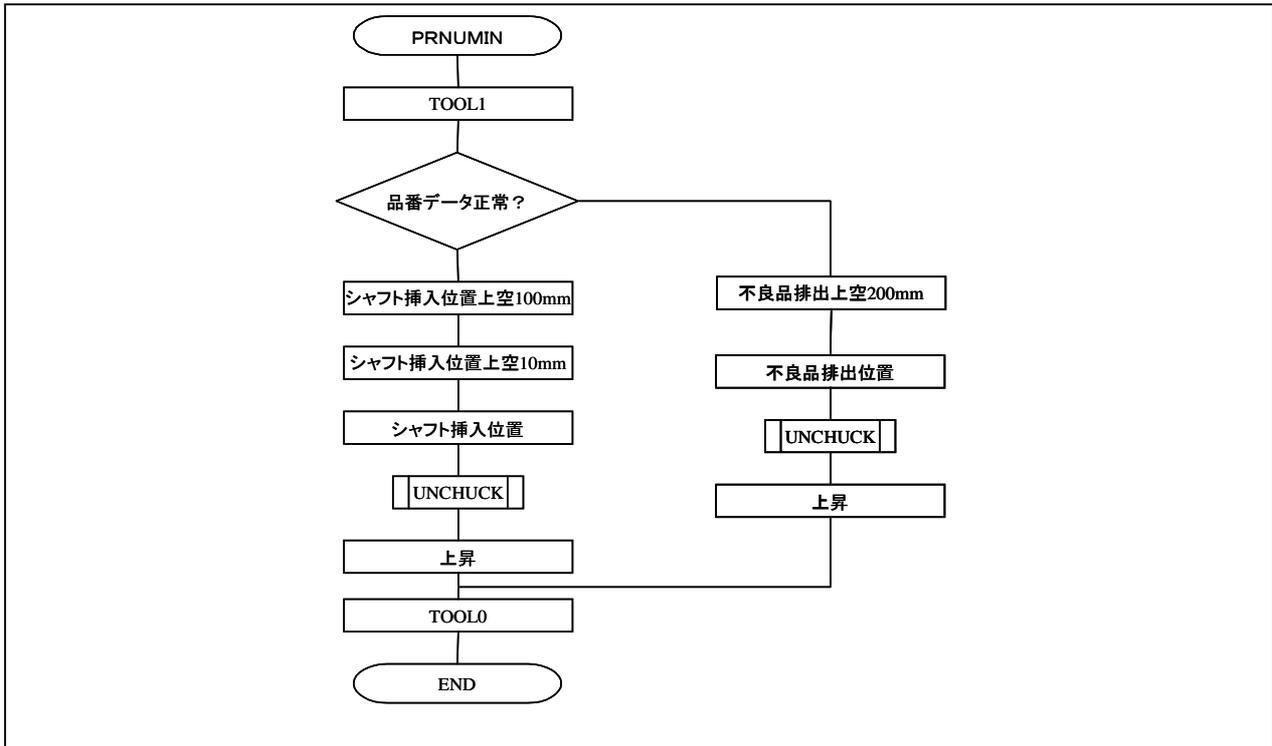
#ifdef __SCARA_ROBOT__                        ' SCARA ROBOT
' 4軸ロボット用
koffset!=h3!*(kcount%-1)
LETP nextpos=PVEC(nextpos)+(0,0,koffset!)
#else
' 6軸ロボット用
koffset!=(h3!*(kcount%-1))*-1
nextpos=T2P(P2T(nextpos)*(0,0,koffset!,0,1,0,0,0,1))
#endif

' FIG値をPP1と同じ値に設定
letf nextpos = fig(pp1)
END

```

## ピック & プレース動作応用プログラム例

### 3.7 品番挿入 (PRNUMIN) のフローチャート



### 3.8 品番挿入 (PRNUMIN) のプログラム

```

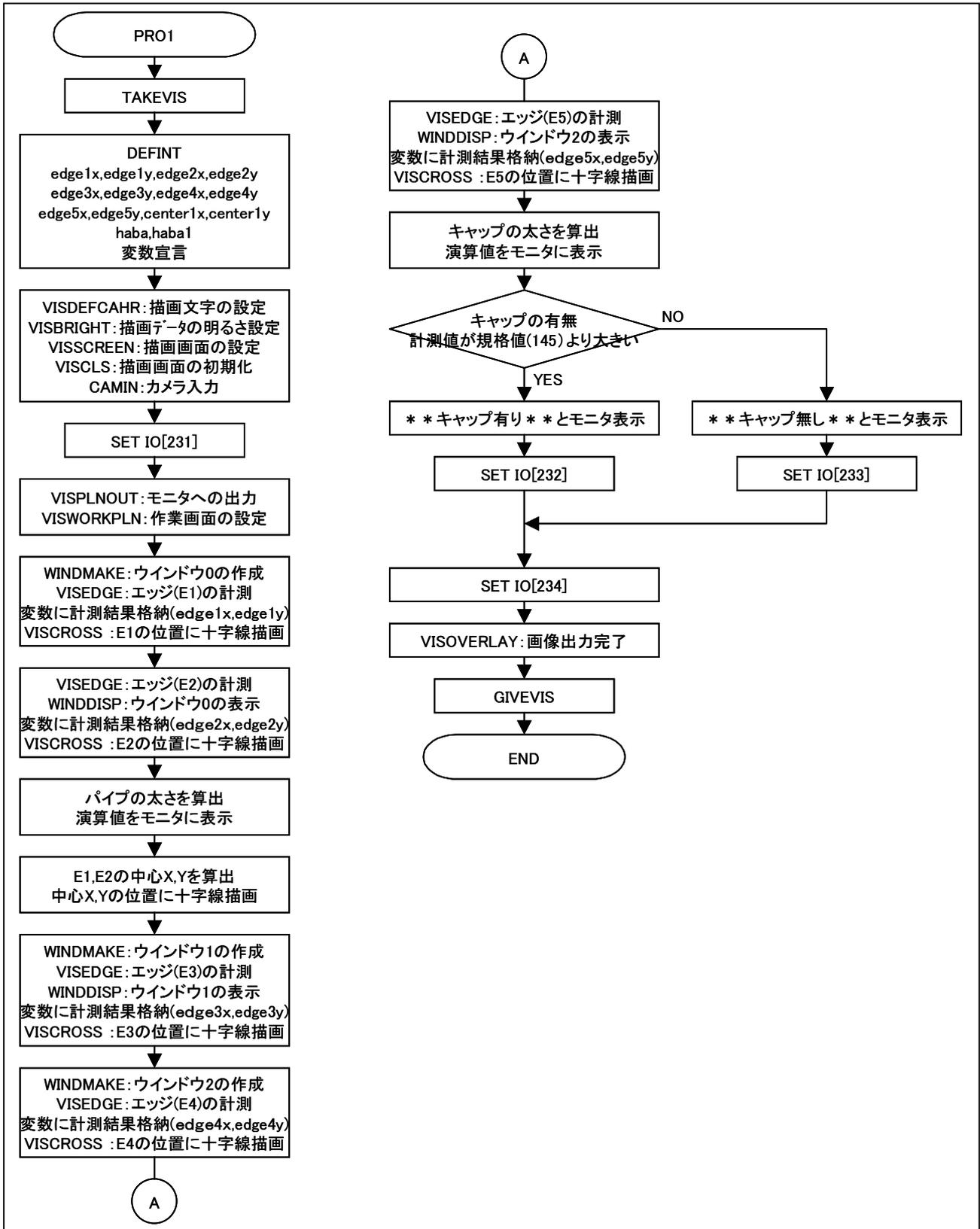
'!TITLE "品番挿入"
PROGRAM PRNUMIN
TAKEARM
CHANGETOOL 1
  IF I50<=7 THEN
    APPROACH P,P[I50],100,S=I1          '品番に応じシャフト挿入
    APPROACH L,P[I50],10,S=I2
    MOVE L,@E P[I50],S=I3
    CALL UNCHUCK
    DEPART L,100,S=I11

  ELSE
    APPROACH P,P30,@P 200,S=I1        '品番異常排出
    MOVE P,P30,S=I2
    CALL UNCHUCK
    DEPART L,200,S=I1

  END IF
CHANGETOOL 0
END
  
```

ピック & プレース動作応用プログラム例

3.9 キャップ有無確認 (PRO1) のフローチャート



## ピック &amp; プレース動作応用プログラム例

## 3.10 視覚プログラム：キャップ有無確認（PRO1）

```

'!TITLE "視覚プログラム"
PROGRAM PRO1
TAKEVIS                                     '視覚命令の許可

      DEFINT edge1x,edge1y,edge2x,edge2y     '値を格納する変数
      DEFINT edge3x,edge3y,edge4x,edge4y,edge5x,edge5y
      DEFINT center1x,center1y,haba,haba1

      VISDEFCHAR 2,2,0                       '描画文字の設定
      VISBRIGHT 255                          '描画データの明るさ設定

      VISSCREEN 1,0,0                        '描画面面の設定
      VISCLS 0                                '描画面面の初期化
      CAMIN 1,0,0                             'カメラ入力

      SET IO[231]                             '画像入力完了

      VISPLNOUT 0,1                           'モニタへの出力
      VISWORKPLN 0                            '作業画面の設定

      WINDMAKE R,0,30,300,0,1                'ウインドウ0の作成
      VISEGDE 0,350,100,1,1,180,0,185,255   'ウインドウ0内のエッジ(E1)の計測
      IF VISSTATUS(0)=0.0 THEN
          edge1x=VISGETNUM(0,1)               '変数edge1xにE1のX座標を格納
          edge1y=VISGETNUM(0,2)               '変数edge1yにE1のY座標を格納
          VISCROSS edge1x,edge1y,10,10        'E1の位置に十字線を描画
      ENDIF

      VISEGDE 0,350,100,-1,1,180,0,185,255   'ウインドウ0内のエッジ(E2)の計測
      WINDDISP 0                              'ウインドウ0の表示
      IF VISSTATUS(0)=0.0 THEN
          edge2x=VISGETNUM(0,1)               '変数edge2xにE2のX座標を格納
          edge2y=VISGETNUM(0,2)               '変数edge2yにE2のY座標を格納
          VISCROSS edge2x,edge2y,10,10        'E2の位置に十字線を描画
      ENDIF

      haba=edge2y-edge1y                       'パイプの太さ(幅)を算出

      VISLOC 10,2                              'パイプの太さ(幅)をモニタに表示
      VISPRINT "幅=";haba;

      center1x=(edge1x+edge2x)/2              'E1,E2の中心Xを算出
      center1y=(edge1y+edge2y)/2              'E1,E2の中心Yを算出
      VISCROSS center1x,center1y,5,5          'E1,E2の中心位置に十字線を描画

```

(次ページへ続く)

## ピック &amp; プレース動作応用プログラム例

(前ページから続く)

```

WINDMAKE R,1,200,30,0,0          'ウインドウ1の作成
VISEDGE 1,(center1x-250),(center1y-15),1,1,180,0,185,255 'ウインドウ1内のエッジ(E3)の計測
WINDDISP 1                        'ウインドウ1の表示
IF VISSTATUS(0)=0.0 THEN
    edge3x=VISGETNUM(0,1)          '変数edge3xにE3のX座標を格納
    edge3y=VISGETNUM(0,2)          '変数edge3yにE3のY座標を格納
    VISCROSS edge3x,edge3y,10,10  'E3の位置に十字線を描画
ENDIF

WINDMAKE R,2,30,240,0,1          'ウインドウ2の作成
VISEDGE 2,(edge3x+40),(edge3y-120),1,1,180,0,185,255 'ウインドウ2内のエッジ(E4)の計測
IF VISSTATUS(0)=0.0 THEN
    edge4x=VISGETNUM(0,1)          '変数edge4xにE4のX座標を格納
    edge4y=VISGETNUM(0,2)          '変数edge4yにE4のY座標を格納
    VISCROSS edge4x,edge4y,10,10  'E4の位置に十字線を描画
ENDIF

VISEDGE 2,(edge3x+40),(edge3y-120),-1,1,180,0,185,255 'ウインドウ2内のエッジ(E5)の計測
WINDDISP 2                        'ウインドウ2の表示
IF VISSTATUS(0)=0.0 THEN
    edge5x=VISGETNUM(0,1)          '変数edge5xにE5のX座標を格納
    edge5y=VISGETNUM(0,2)          '変数edge5yにE5のY座標を格納
    VISCROSS edge5x,edge5y,10,10  'E5の位置に十字線を描画
ENDIF

haba1=edge5y-edge4y              'キャップの幅を算出

VISLOC 10,4                       'キャップの幅をモニタに表示
VISPRINT "キャップ幅=";haba1;

IF haba1 > 145 THEN                '規格値と比べ計測値が大きければキャップ有り判定
    VISLOC 10,26                   ' **キャップ有り ** とモニタに表示
    VISPRINT " ** キャップ有り ** "
    SET IO[232]                     '認識OK信号をON
ELSE
    VISLOC 10,26                   '規格値と比べ計測値が小さければキャップ無し判定
    VISPRINT " ** キャップ無し ** " ' **キャップ無し ** とモニタに表示
    SET IO[233]                     '認識NG信号をON
ENDIF

SET IO[234]                        '画像出力完了

VISOVERLAY 1                       '描画の実行(一括表示)

GIVEVIS

```

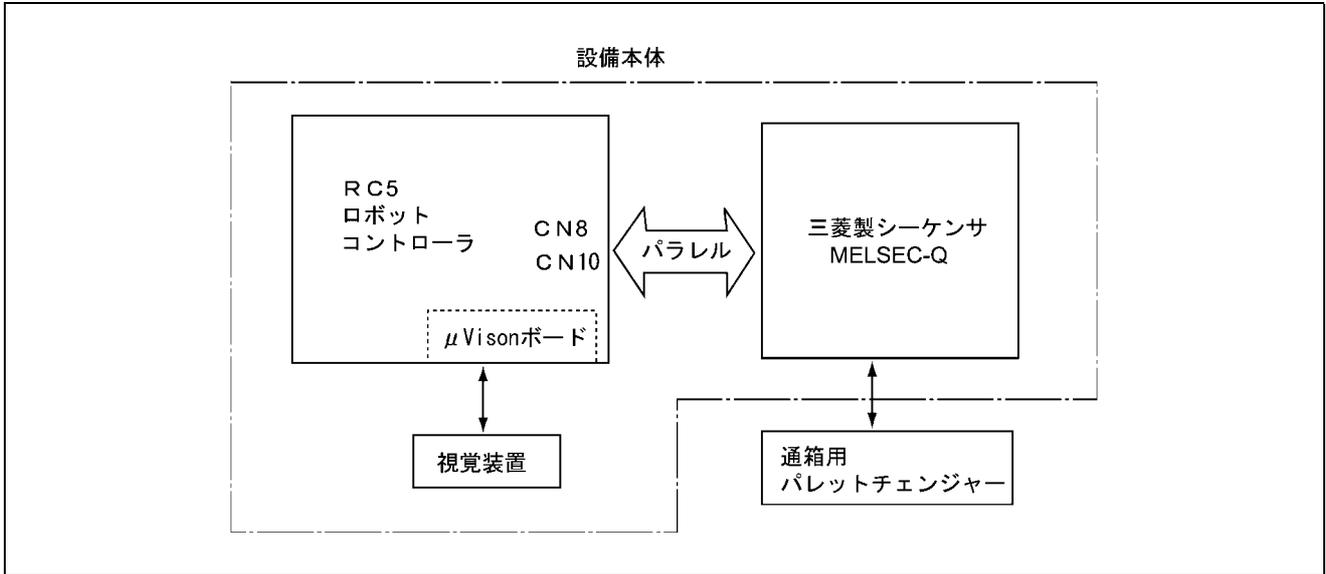
END

# ピック & プレース動作応用プログラム例

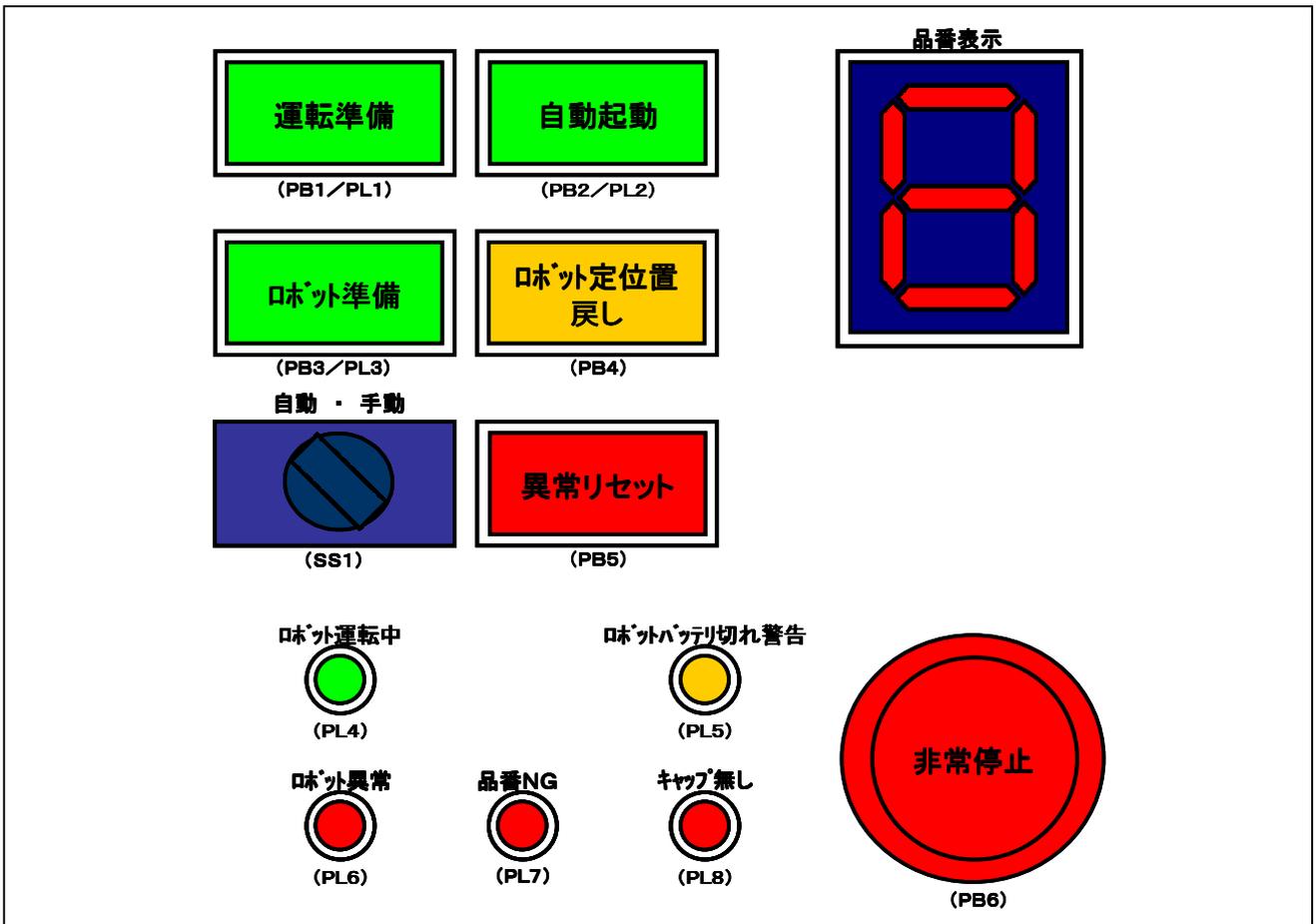
## 4. システム構成

ピック & プレース動作応用のシステム構成例を以下に示します。

### 4.1 システム構成図例と操作盤レイアウト例



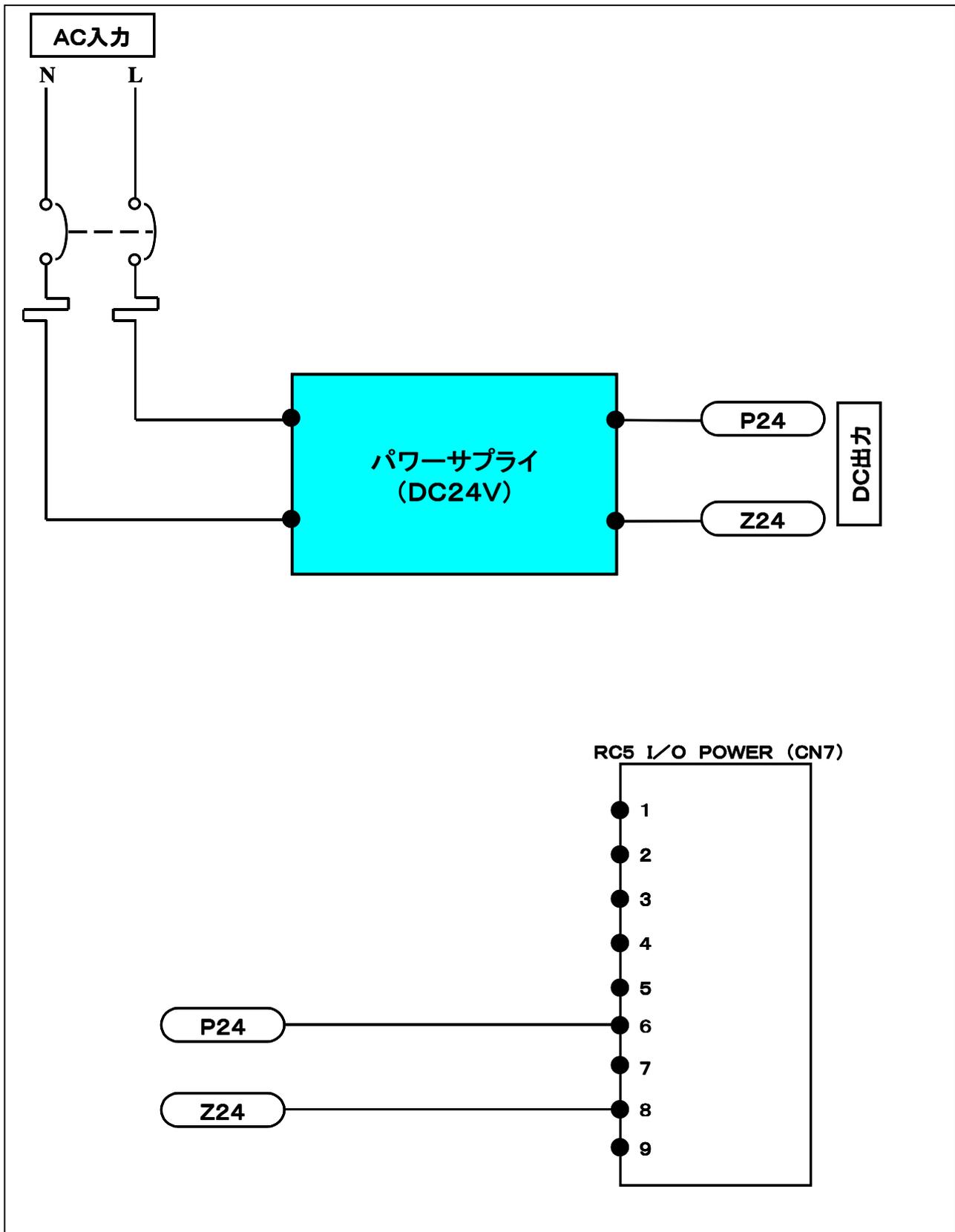
システム構成図の例



操作盤レイアウトの例

## ピック&amp;プレース動作応用プログラム例

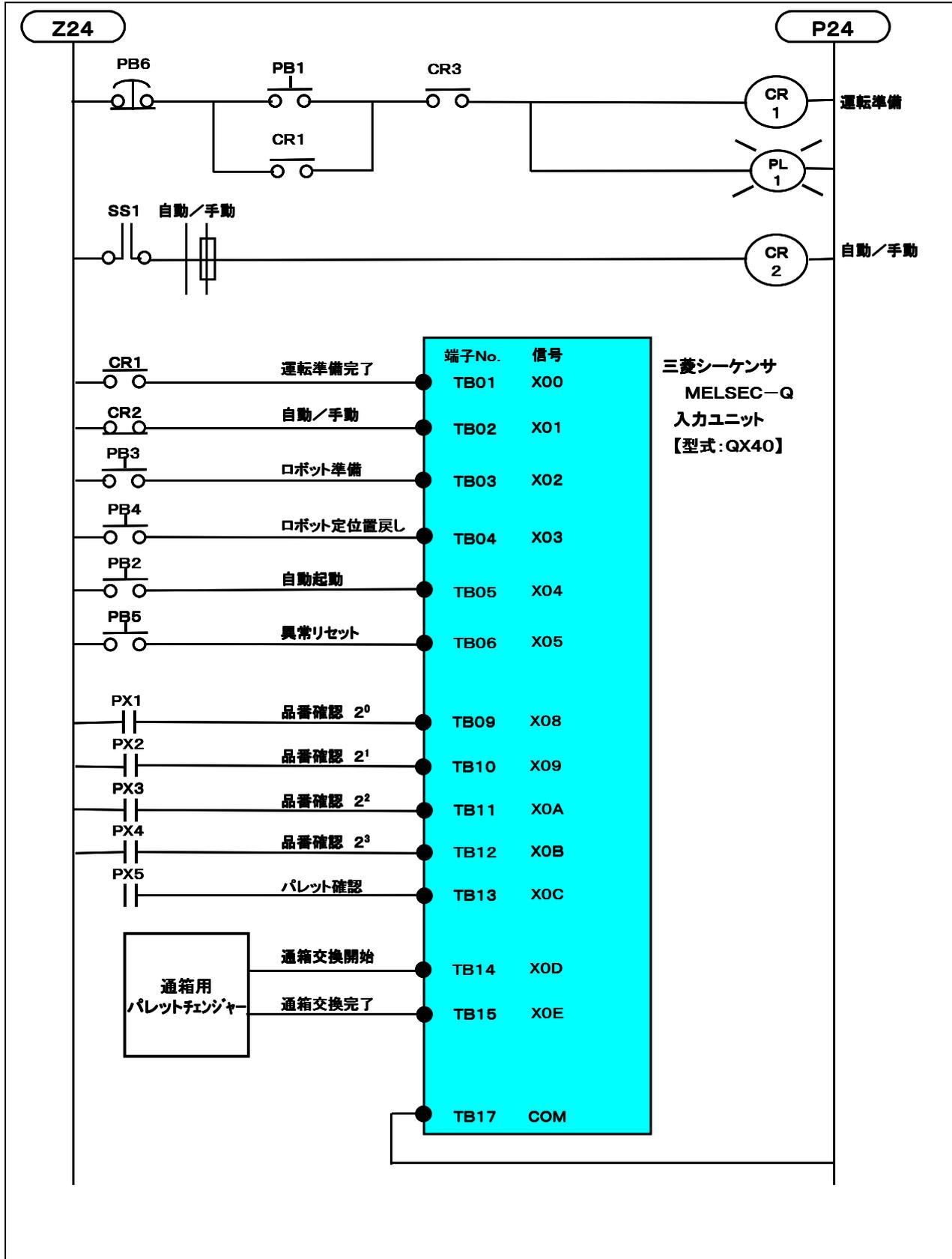
## 4.2 ハード回路例



ハード回路例 (1/4)

(次ページへ続く)

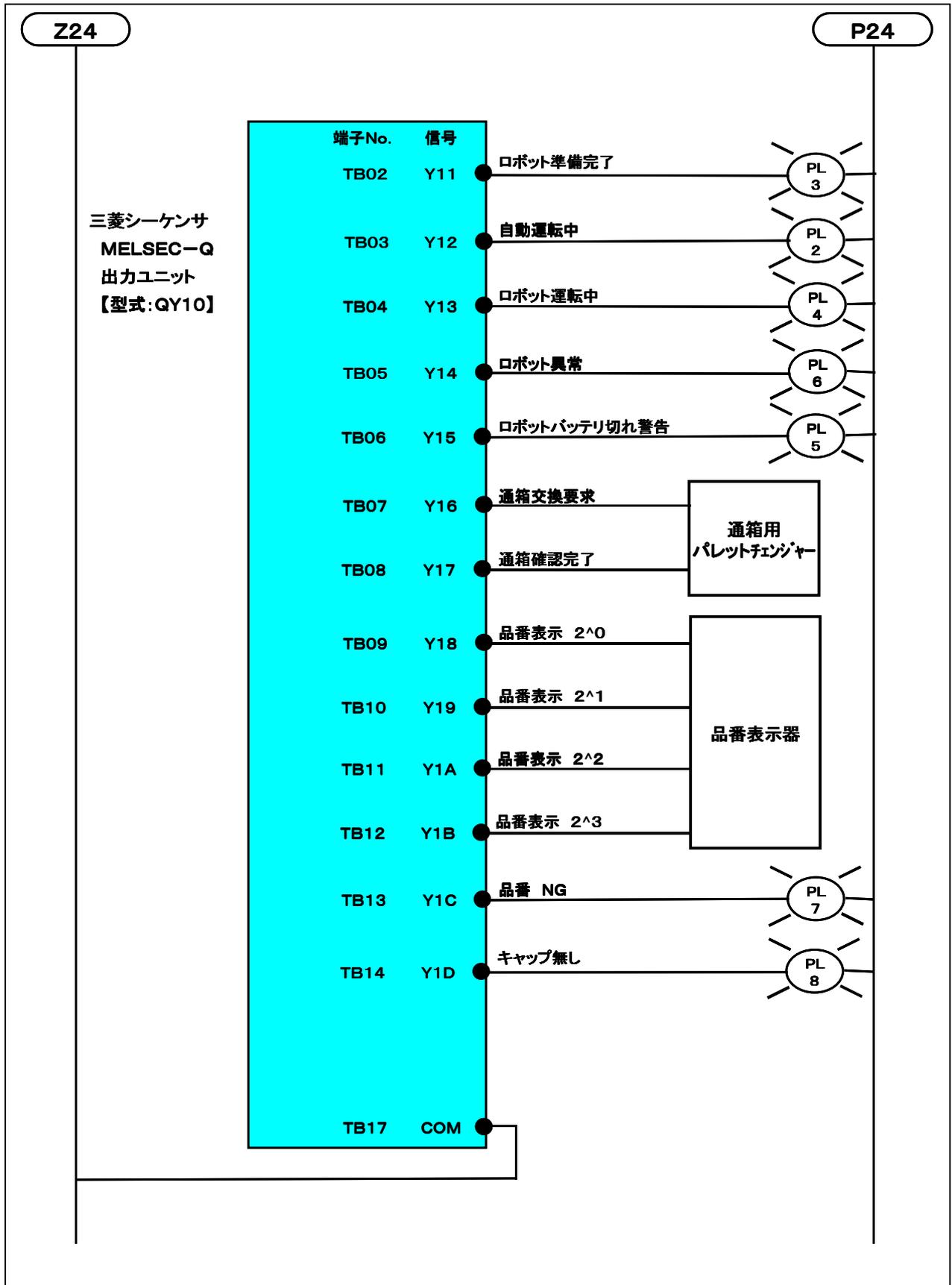
ピック & プレース動作応用プログラム例



ハード回路例 (2/4)

(次ページへ続く)

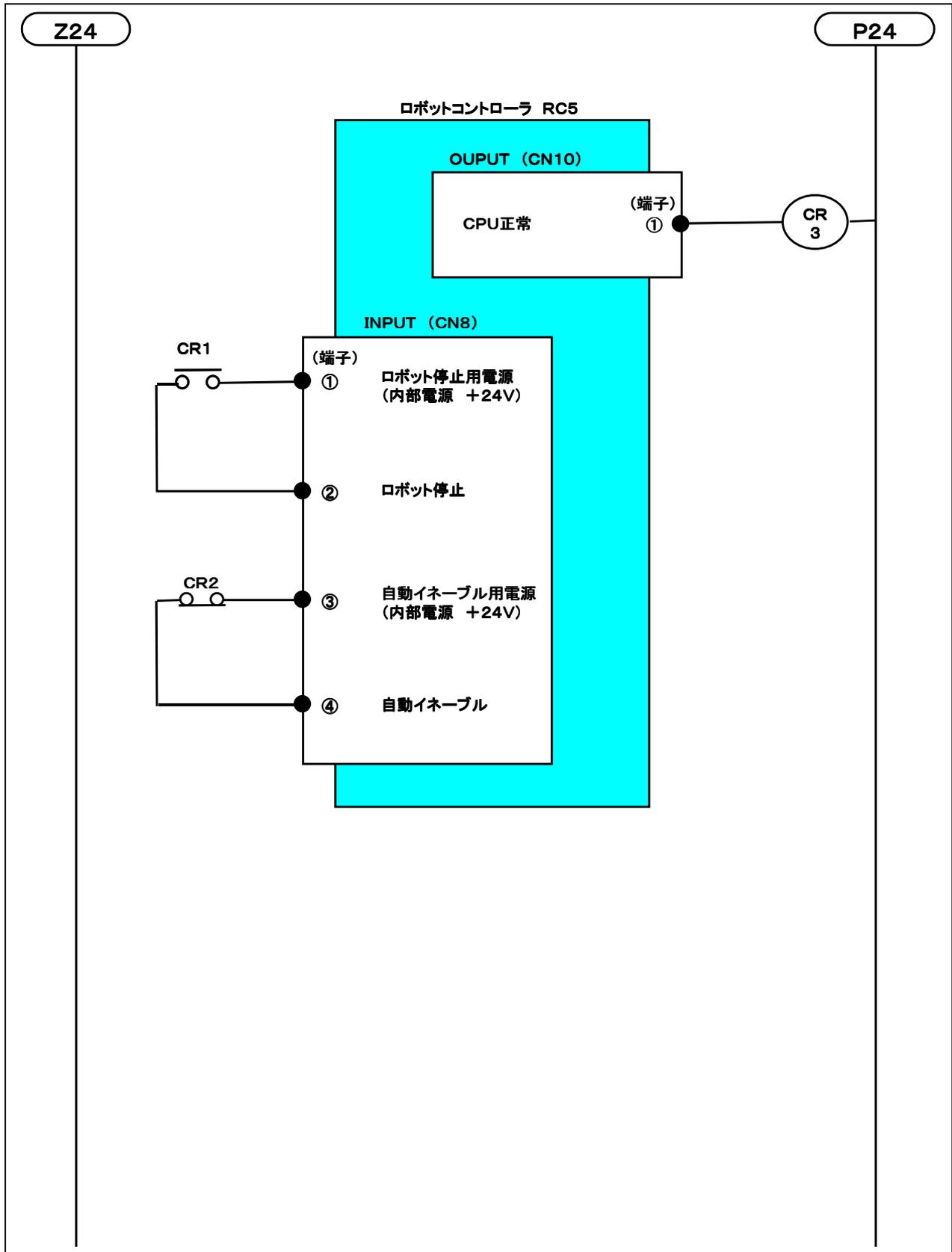
ピック & プレース動作応用プログラム例



ハード回路例 (3/4)

(次ページへ続く)

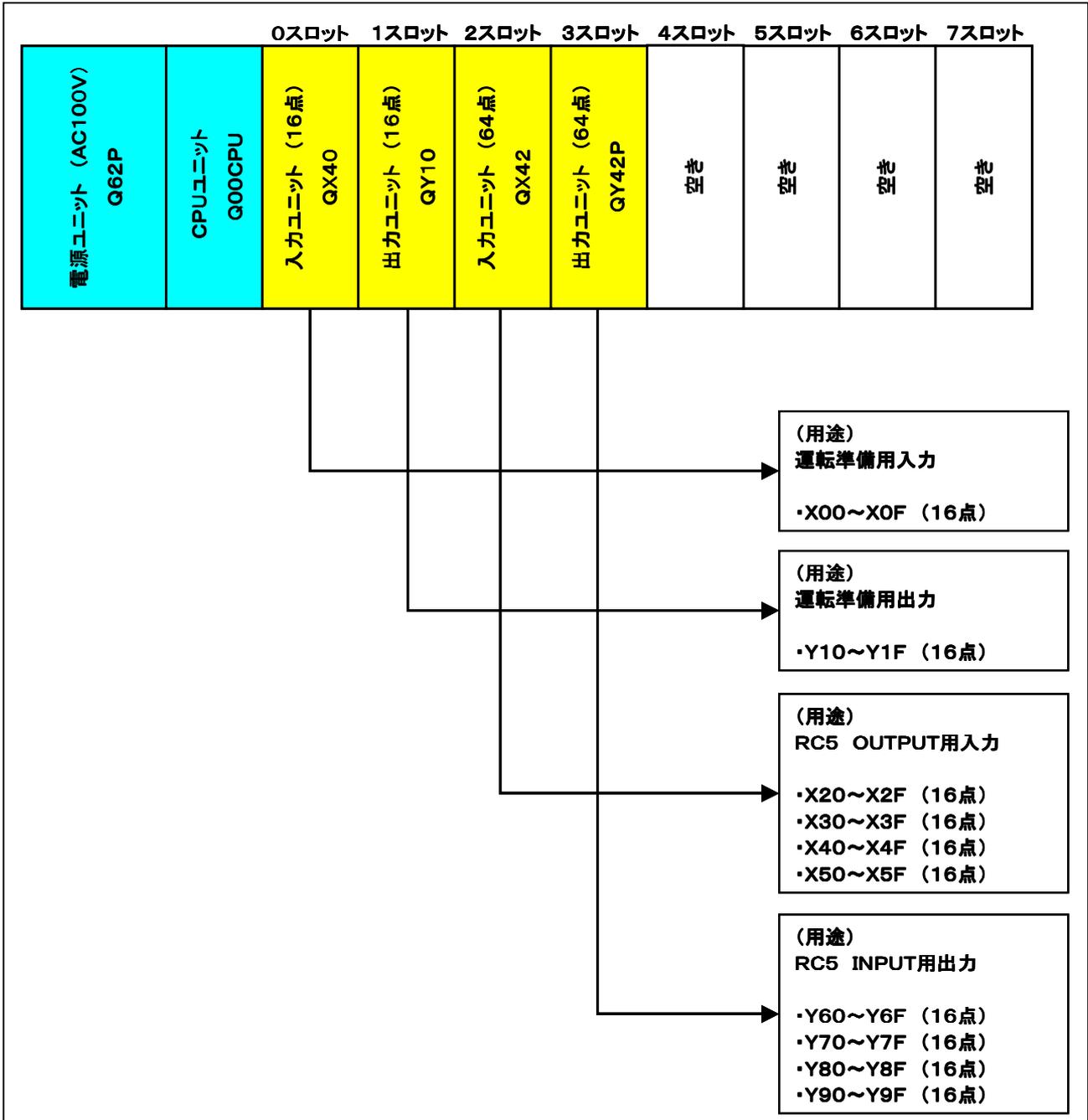
## ピック &amp; プレース動作応用プログラム例



ハード回路例 (4/4)

ピック & プレース動作応用プログラム例

4.3 シーケンサ (三菱製 MELSEC-Q) のスロット割付例



## ピック &amp; プレース動作応用プログラム例

## 4.5 シーケンサ (三菱製 MELSEC-Q) の I/O 割付の例

ワードアドレスと 使用ユニット	信号	名称	接続先	備考
000 入力ユニット QX40 (16点 / 16点)	X000	運転準備 (CR)	CR1	
	X001	自動/手動 (CR)	CR2	
	X002	ロボット準備 (PB)	PB3	
	X003	ロボット定位置戻し (PB)	PB4	
	X004	自動起動 (PB)	PB2	
	X005	異常リセット (PB)	PB5	
	X006			
	X007			
	X008	品番確認 2 <sup>0</sup>	PX1	
	X009	品番確認 2 <sup>1</sup>	PX2	
	X00A	品番確認 2 <sup>2</sup>	PX3	
	X00B	品番確認 2 <sup>3</sup>	PX4	
	X00C	パレット確認	PX5	
	X00D	通箱交換開始 (パレットチェンジャーから)	パレットチェンジャー	
	X00E	通箱交換完了 (パレットチェンジャーから)	パレットチェンジャー	
X00F				
001 出力ユニット QY10 (16点 / 16点)	Y010			
	Y011	ロボット準備完了 (PL)	PL3	
	Y012	自動運転中 (PL)	PL2	
	Y013	ロボット運転中 (PL)	PL4	
	Y014	ロボット異常 (PL)	PL6	
	Y015	バッテリー切れ警告 (PL)	PL5	
	Y016	通箱交換要求 (パレットチェンジャーへ)	パレットチェンジャー	
	Y017	通箱確認完了 (パレットチェンジャーへ)	パレットチェンジャー	
	Y018	品番表示 2 <sup>0</sup>	品番表示器	
	Y019	品番表示 2 <sup>1</sup>	品番表示器	
	Y01A	品番表示 2 <sup>2</sup>	品番表示器	
	Y01B	品番表示 2 <sup>3</sup>	品番表示器	
	Y01C	品番NG (PL)	PL7	
	Y01D	キャップ無し (PL)	PL8	
	Y01E			
Y01F				

シーケンサ (三菱製 MELSEC-Q) の I/O 割付の例 (1/5)

(次ページへ続く)

## ピック &amp; プレース動作応用プログラム例

ワードアドレスと 使用ユニット	信号	名称	接続先 (注1)	備考
002 入力ユニット QX42 (16点/64点)	X020			
	X021	ロボット運転中	RC5 CN10 端子2 【茶(桃)】	RC5ホト 73
	X022	ロボット異常	RC5 CN10 端子3 【赤(桃)】	RC5ホト 74
	X023	サーボON	RC5 CN10 端子4 【橙(桃)】	RC5ホト 75
	X024	ロボット初期化完了	RC5 CN10 端子5 【黄(桃)】	RC5ホト 76
	X025	自動モード	RC5 CN10 端子6 【黒(白)】	RC5ホト 77
	X026	外部モード	RC5 CN10 端子7 【茶(白)】	RC5ホト 78
	X027	バッテリー切れ警告	RC5 CN10 端子8 【赤(白)】	RC5ホト 79
	X028	ロボット警告	RC5 CN10 端子9 【橙(白)】	RC5ホト 80
	X029	コンティニュースタート許可	RC5 CN10 端子10 【黄(白)】	RC5ホト 81
	X02A	SSモード	RC5 CN10 端子11 【緑(白)】	RC5ホト 82
	X02B	予約	RC5 CN10 端子12 【青(白)】	RC5ホト 83
	X02C	予約	RC5 CN10 端子13 【紫(白)】	RC5ホト 84
	X02D	予約	RC5 CN10 端子14 【灰(白)】	RC5ホト 85
	X02E	コマンド処理完了	RC5 CN10 端子15 【桃(白)】	RC5ホト 86
	X02F	ステータス領域奇数パリティ	RC5 CN10 端子16 【黒(灰)】	RC5ホト 87
003 入力ユニット QX42 (32点/64点)	X030	ステータス領域第0ビット	RC5 CN10 端子17 【黒(紫)】	RC5ホト 88
	X031	ステータス領域第1ビット	RC5 CN10 端子18 【茶(紫)】	RC5ホト 89
	X032	ステータス領域第2ビット	RC5 CN10 端子19 【赤(紫)】	RC5ホト 90
	X033	ステータス領域第3ビット	RC5 CN10 端子20 【橙(紫)】	RC5ホト 91
	X034	ステータス領域第4ビット	RC5 CN10 端子21 【黄(紫)】	RC5ホト 92
	X035	ステータス領域第5ビット	RC5 CN10 端子22 【緑(紫)】	RC5ホト 93
	X036	ステータス領域第6ビット	RC5 CN10 端子23 【青(紫)】	RC5ホト 94
	X037	ステータス領域第7ビット	RC5 CN10 端子24 【灰(紫)】	RC5ホト 95
	X038	ステータス領域第8ビット	RC5 CN10 端子25 【桃(紫)】	RC5ホト 96
	X039	ステータス領域第9ビット	RC5 CN10 端子26 【茶(灰)】	RC5ホト 97
	X03A	ステータス領域第10ビット	RC5 CN10 端子27 【赤(灰)】	RC5ホト 98
	X03B	ステータス領域第11ビット	RC5 CN10 端子28 【橙(灰)】	RC5ホト 99
	X03C	ステータス領域第12ビット	RC5 CN10 端子29 【黄(灰)】	RC5ホト 100
	X03D	ステータス領域第13ビット	RC5 CN10 端子30 【緑(灰)】	RC5ホト 101
	X03E	ステータス領域第14ビット	RC5 CN10 端子31 【青(灰)】	RC5ホト 102
	X03F	ステータス領域第15ビット	RC5 CN10 端子32 【桃(灰)】	RC5ホト 103

注1： 接続先欄の【 】内に示す線色の見方。例えば、【茶(桃)】の場合、オプション設定品のI/Oケーブルを使用した時の茶色と桃色のツイストペア線の茶色線を示す。

シーケンサ(三菱製MELSEC-Q)のI/O割付の例(2/5)

(次ページへ続く)

## ピック &amp; プレース動作応用プログラム例

ワードアドレスと 使用ユニット	信号	名称	接続先 (注1)	備考
004 入力ユニット QX42 (48点 / 64点)	X040	ロボット定位置	RC5 CN10 端子33【黒(青)】	RC5ホト 104
	X041	キャップ無し	RC5 CN10 端子34【茶(青)】	RC5ホト 105
	X042	通箱交換要求	RC5 CN10 端子35【桃(黒)】	RC5ホト 106
	X043	品番要求	RC5 CN10 端子36【桃(茶)】	RC5ホト 107
	X044	品番受取確認	RC5 CN10 端子37【桃(赤)】	RC5ホト 108
	X045	(汎用出力)	RC5 CN10 端子38【桃(橙)】	RC5ホト 109
	X046	(汎用出力)	RC5 CN10 端子39【桃(黄)】	RC5ホト 110
	X047	(汎用出力)	RC5 CN10 端子40【白(黒)】	RC5ホト 111
	X048	(汎用出力)	RC5 CN10 端子41【白(茶)】	RC5ホト 112
	X049	(汎用出力)	RC5 CN10 端子42【白(赤)】	RC5ホト 113
	X04A	(汎用出力)	RC5 CN10 端子43【白(橙)】	RC5ホト 114
	X04B	(汎用出力)	RC5 CN10 端子44【白(黄)】	RC5ホト 115
	X04C	(汎用出力)	RC5 CN10 端子45【白(緑)】	RC5ホト 116
	X04D	(汎用出力)	RC5 CN10 端子46【白(青)】	RC5ホト 117
	X04E	(汎用出力)	RC5 CN10 端子47【白(紫)】	RC5ホト 118
	X04F	(汎用出力)	RC5 CN10 端子48【白(灰)】	RC5ホト 119
005 入力ユニット QX42 (64点 / 64点)	X050	(汎用出力)	RC5 CN10 端子49【白(桃)】	RC5ホト 120
	X051	(汎用出力)	RC5 CN10 端子50【灰(黒)】	RC5ホト 121
	X052	(汎用出力)	RC5 CN10 端子51【紫(黒)】	RC5ホト 122
	X053	(汎用出力)	RC5 CN10 端子52【紫(茶)】	RC5ホト 123
	X054	(汎用出力)	RC5 CN10 端子53【紫(赤)】	RC5ホト 124
	X055	(汎用出力)	RC5 CN10 端子54【紫(橙)】	RC5ホト 125
	X056	(汎用出力)	RC5 CN10 端子55【紫(黄)】	RC5ホト 126
	X057	(汎用出力)	RC5 CN10 端子56【紫(緑)】	RC5ホト 127
	X058			
	X059			
	X05A			
	X05B			
	X05C			
	X05D			
	X05E			
	X05F			

注1： 接続先欄の【 】内に示す線色の見方。例えば、【茶(桃)】の場合、オプション設定品のI/Oケーブルを使用した時の茶色と桃色のツイストペア線の茶色線を示す。

シーケンサ(三菱製MELSEC-Q)のI/O割付の例(3/5)

(次ページへ続く)

## ピック &amp; プレース動作応用プログラム例

ワードアドレスと 使用ユニット	信号	名称	接続先 (注1)	備考
006 出力ユニット QY42P (16点/64点)	Y060	ステップ停止 (全タスク)	RC5 CN8 端子5【茶(白)】	RC5ホト 0
	Y061			
	Y062	瞬時停止 (全タスク)	RC5 CN8 端子7【橙(白)】	RC5ホト 2
	Y063	ストローブ信号	RC5 CN8 端子8【黄(白)】	RC5ホト 3
	Y064	割り込みスキップ	RC5 CN8 端子9【緑(白)】	RC5ホト 4
	Y065			
	Y066			
	Y067	コマンド・データ奇数パリティ	RC5 CN8 端子10【青(白)】	RC5ホト 5
	Y068	データ領域1 第0 ビット	RC5 CN8 端子11【紫(白)】	RC5ホト 6
	Y069	データ領域1 第1 ビット	RC5 CN8 端子12【黒(灰)】	RC5ホト 7
	Y06A	データ領域1 第2 ビット	RC5 CN8 端子13【茶(灰)】	RC5ホト 8
	Y06B	データ領域1 第3 ビット	RC5 CN8 端子14【赤(灰)】	RC5ホト 9
	Y06C	データ領域1 第4 ビット	RC5 CN8 端子15【橙(灰)】	RC5ホト 10
	Y06D	データ領域1 第5 ビット	RC5 CN8 端子16【黄(灰)】	RC5ホト 11
	Y06E	データ領域1 第6 ビット	RC5 CN8 端子17【緑(灰)】	RC5ホト 12
Y06F	データ領域1 第7 ビット	RC5 CN8 端子18【青(灰)】	RC5ホト 13	
007 出力ユニット QY42P (32点/64点)	Y070	データ領域2 第0 ビット	RC5 CN8 端子19【紫(灰)】	RC5ホト 14
	Y071	データ領域2 第1 ビット	RC5 CN8 端子20【白(灰)】	RC5ホト 15
	Y072	データ領域2 第2 ビット	RC5 CN8 端子21【桃(灰)】	RC5ホト 16
	Y073	データ領域2 第3 ビット	RC5 CN8 端子22【黒(紫)】	RC5ホト 17
	Y074	データ領域2 第4 ビット	RC5 CN8 端子23【茶(紫)】	RC5ホト 18
	Y075	データ領域2 第5 ビット	RC5 CN8 端子24【赤(紫)】	RC5ホト 19
	Y076	データ領域2 第6 ビット	RC5 CN8 端子25【橙(紫)】	RC5ホト 20
	Y077	データ領域2 第7 ビット	RC5 CN8 端子26【桃(黒)】	RC5ホト 21
	Y078	データ領域2 第8 ビット	RC5 CN8 端子27【桃(茶)】	RC5ホト 22
	Y079	データ領域2 第9 ビット	RC5 CN8 端子28【桃(赤)】	RC5ホト 23
	Y07A	データ領域2 第10ビット	RC5 CN8 端子29【白(黒)】	RC5ホト 24
	Y07B	データ領域2 第11ビット	RC5 CN8 端子30【白(茶)】	RC5ホト 25
	Y07C	データ領域2 第12ビット	RC5 CN8 端子31【白(赤)】	RC5ホト 26
	Y07D	データ領域2 第13ビット	RC5 CN8 端子32【白(橙)】	RC5ホト 27
	Y07E	データ領域2 第14ビット	RC5 CN8 端子33【白(黄)】	RC5ホト 28
Y07F	データ領域2 第15ビット	RC5 CN8 端子34【白(緑)】	RC5ホト 29	

注1: 接続先欄の【 】内に示す線色の見方。例えば、【茶(桃)】の場合、オプション設定品のI/Oケーブルを使用した時の茶色と桃色のツイストペア線の茶色線を示す。

シーケンサ (三菱製MELSEC-Q) のI/O割付の例 (4/5)

(次ページへ続く)

## ピック &amp; プレース動作応用プログラム例

ワードアドレスと 使用ユニット	信号	名称	接続先 (注1)	備考
008 出力ユニット QY42P (48点/64点)	Y080	コマンド領域第0ビット	RC5 CN8 端子35【白(青)】	RC5ホト 30
	Y081	コマンド領域第1ビット	RC5 CN8 端子36【白(紫)】	RC5ホト 31
	Y082	コマンド領域第2ビット	RC5 CN8 端子37【灰(黒)】	RC5ホト 32
	Y083	コマンド領域第3ビット	RC5 CN8 端子38【灰(茶)】	RC5ホト 33
	Y084			
	Y085			
	Y086			
	Y087			
	Y088			
	Y089			
	Y08A			
	Y08B			
	Y08C			
	Y08D			
	Y08E			
Y08F				
009 出力ユニット QY42P (64点/64点)	Y090	通箱交換完了	RC5 CN8 端子39【灰(赤)】	RC5ホト 34
	Y091	作業パレット有り	RC5 CN8 端子40【灰(橙)】	RC5ホト 35
	Y982	品番確認信号	RC5 CN8 端子41【灰(黄)】	RC5ホト 36
	Y093	品番OK	RC5 CN8 端子42【灰(緑)】	RC5ホト 37
	Y094	品番 2 <sup>0</sup>	RC5 CN8 端子43【灰(青)】	RC5ホト 38
	Y095	品番 2 <sup>1</sup>	RC5 CN8 端子44【灰(紫)】	RC5ホト 39
	Y096	品番 2 <sup>2</sup>	RC5 CN8 端子45【灰(白)】	RC5ホト 40
	Y097	品番 2 <sup>3</sup>	RC5 CN8 端子46【灰(桃)】	RC5ホト 41
	Y098	品番ストロープ信号	RC5 CN8 端子47【紫(黒)】	RC5ホト 42
	Y099	(汎用入力)	RC5 CN8 端子48【紫(茶)】	RC5ホト 43
	Y09A	(汎用入力)	RC5 CN8 端子49【紫(赤)】	RC5ホト 44
	Y09B	(汎用入力)	RC5 CN8 端子50【紫(橙)】	RC5ホト 45
	Y09C			
	Y09D			
	Y09E			
Y09F				

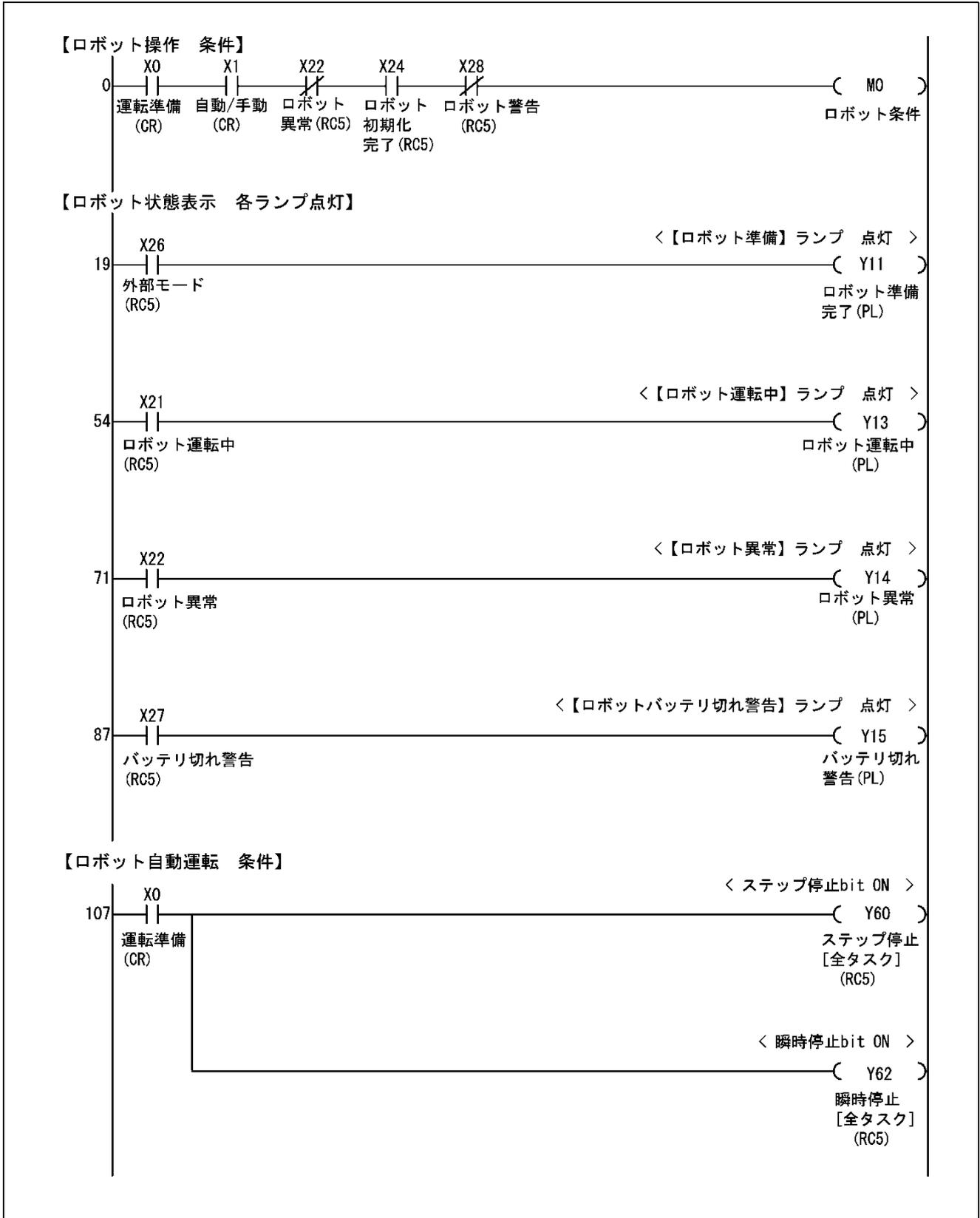
注1： 接続先欄の【 】内に示す線色の見方。例えば、【茶(桃)】の場合、オプション設定品のI/Oケーブルを使用した時の茶色と桃色のツイストペア線の茶色線を示す。

シーケンサ(三菱製MELSEC-Q)のI/O割付の例(5/5)

ピック & プレース動作応用プログラム例

5. シーケンスプログラムの例

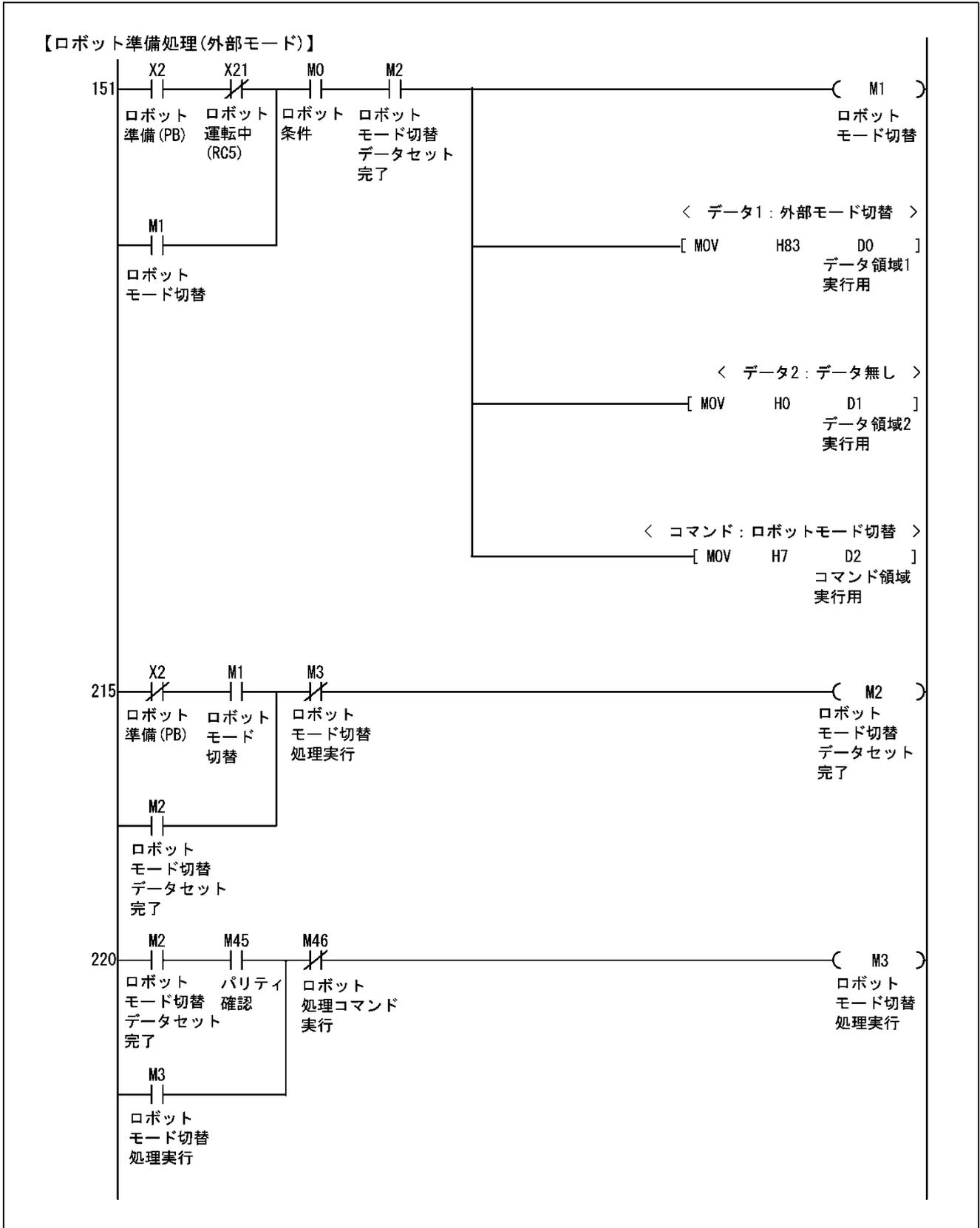
以下にシーケンスプログラムの例を示します。



シーケンスプログラムの例 (1/12)

(次ページへ続く)

ピック & プレース動作応用プログラム例

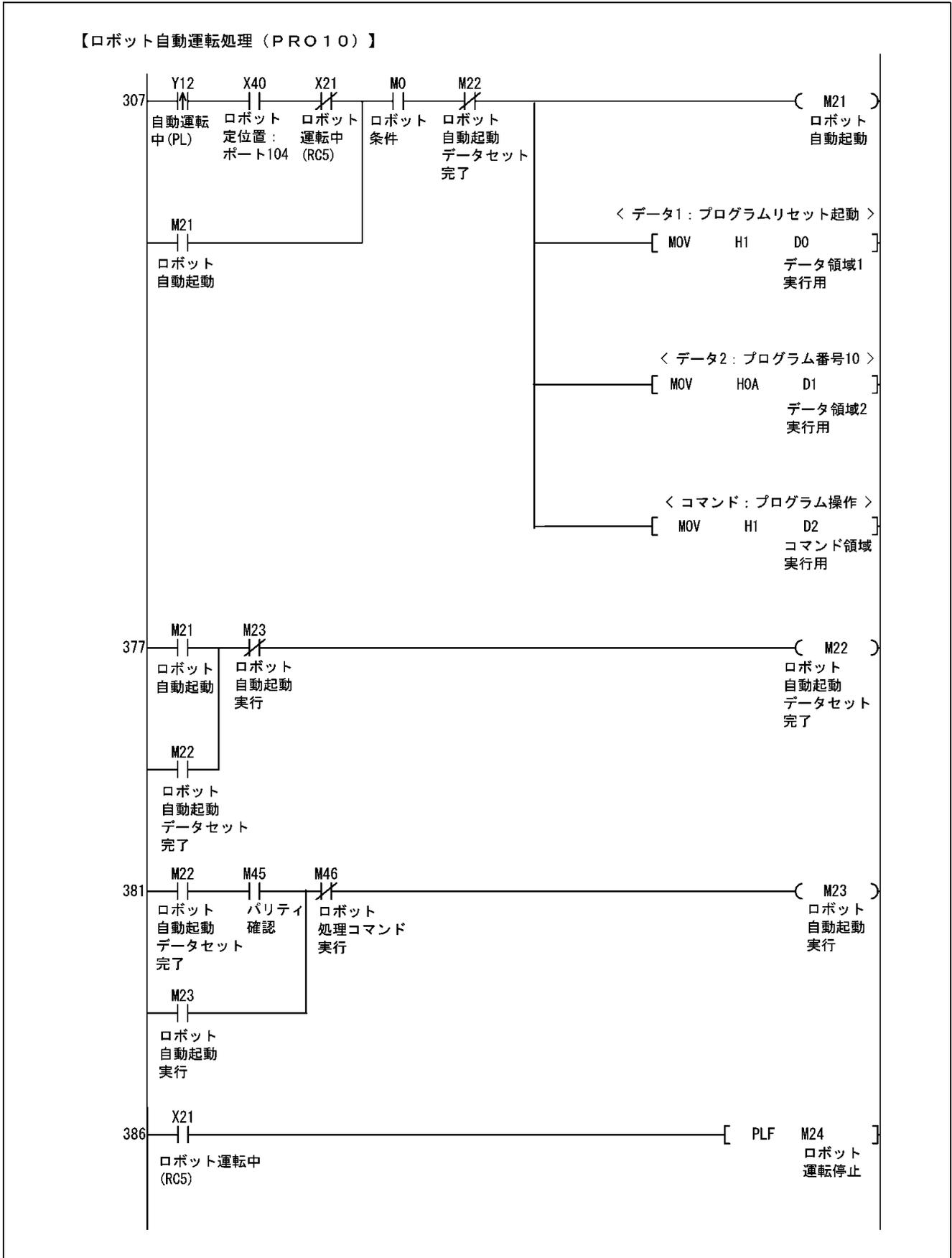


シーケンスプログラムの例 (2/12)

(次ページへ続く)



ピック & プレース動作応用プログラム例

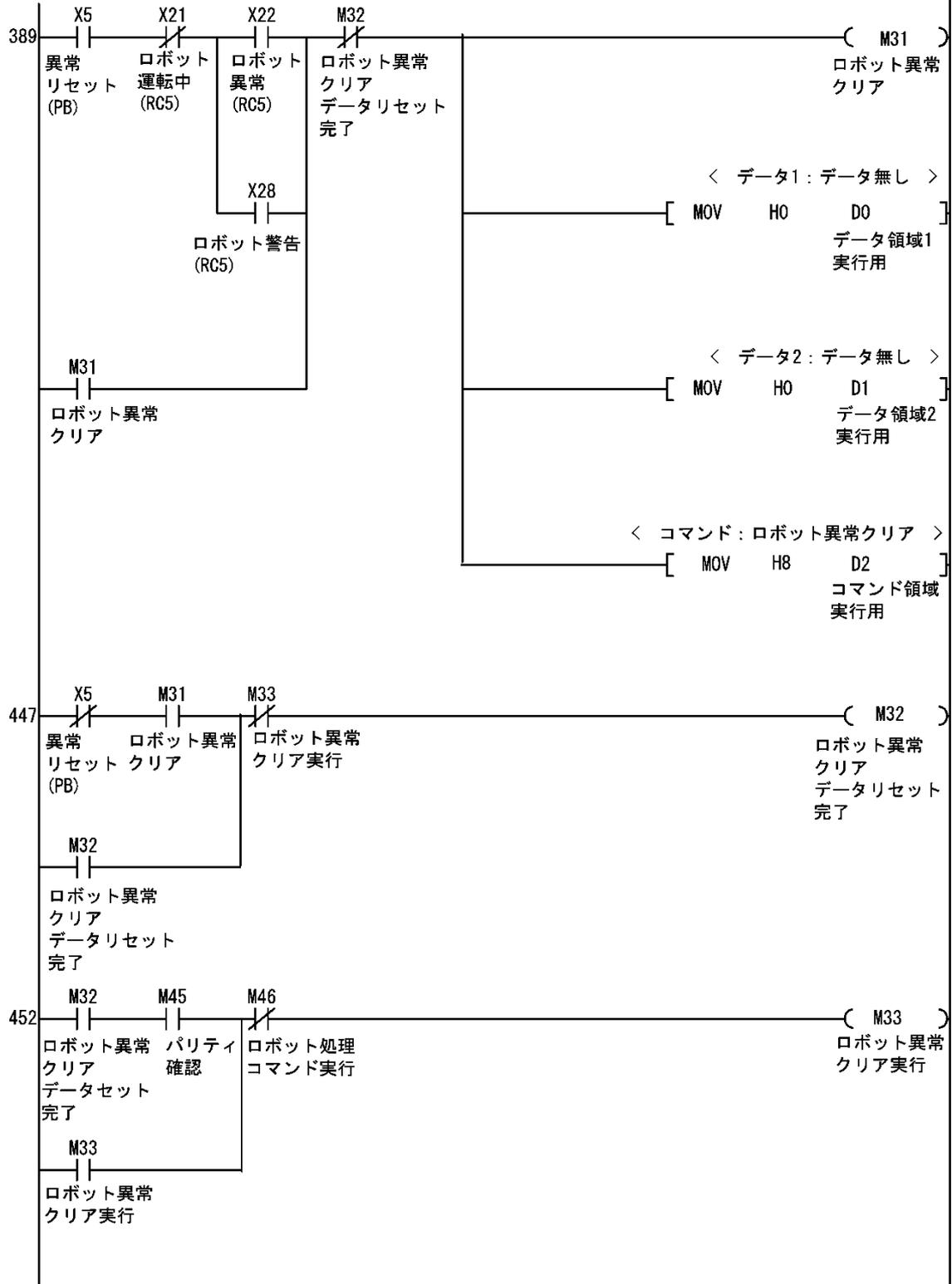


シーケンスプログラムの例 (4/12)

(次ページへ続く)

ピック & プレース動作応用プログラム例

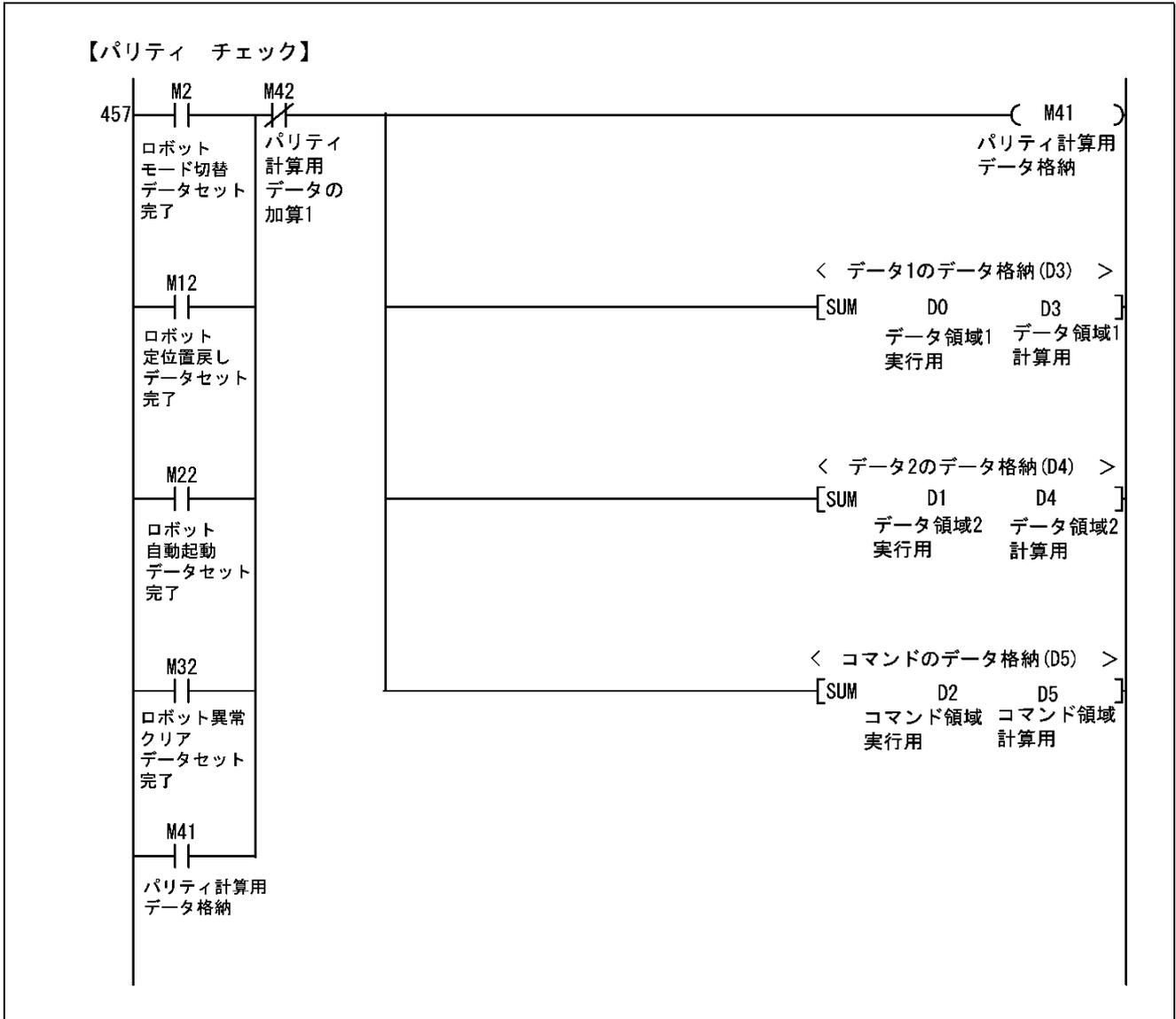
【ロボット異常処理】



シーケンスプログラムの例 ( 5/12 )

( 次ページへ続く )

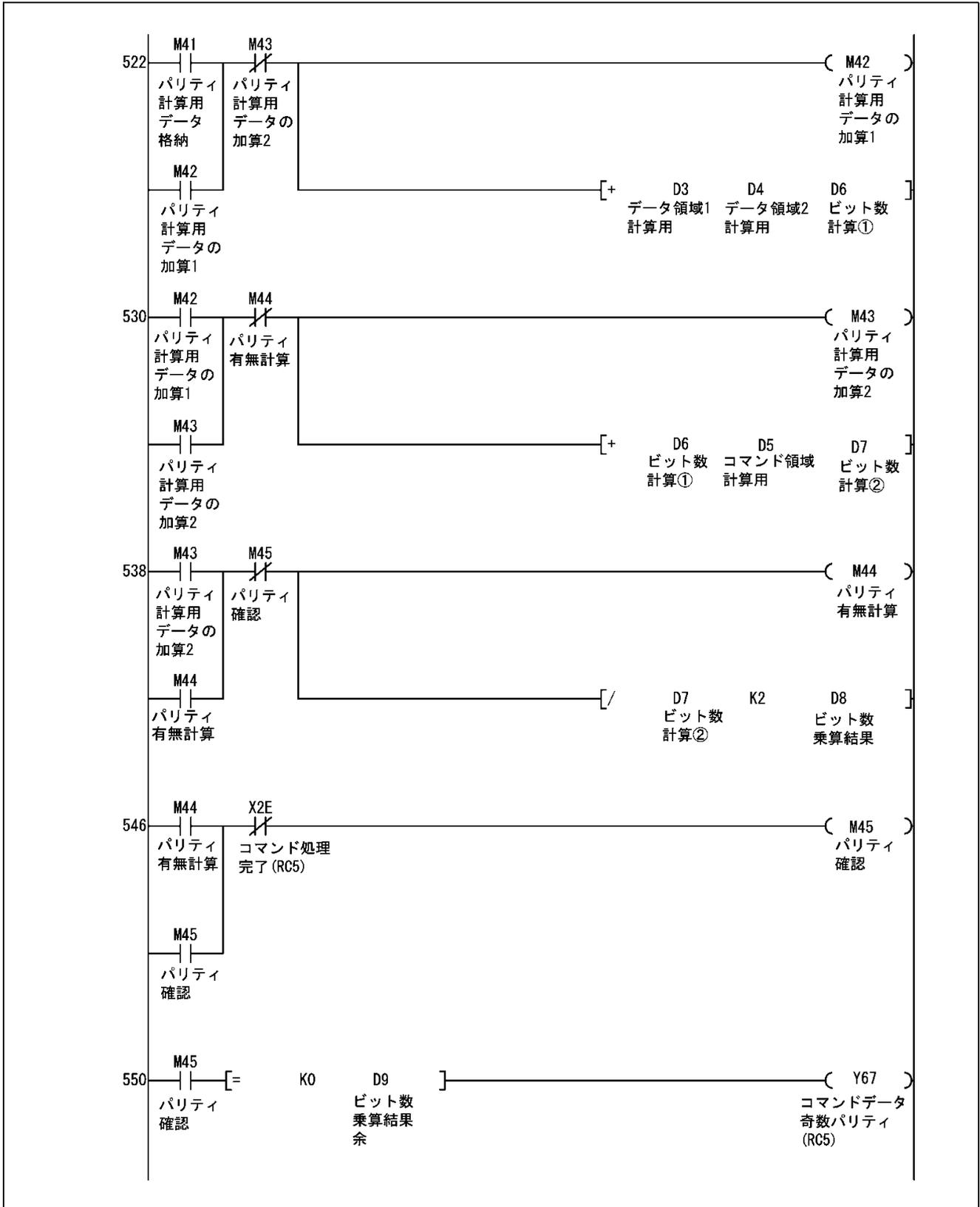
ピック & プレース動作応用プログラム例



シーケンスプログラムの例 ( 6/12 )

( 次ページへ続く )

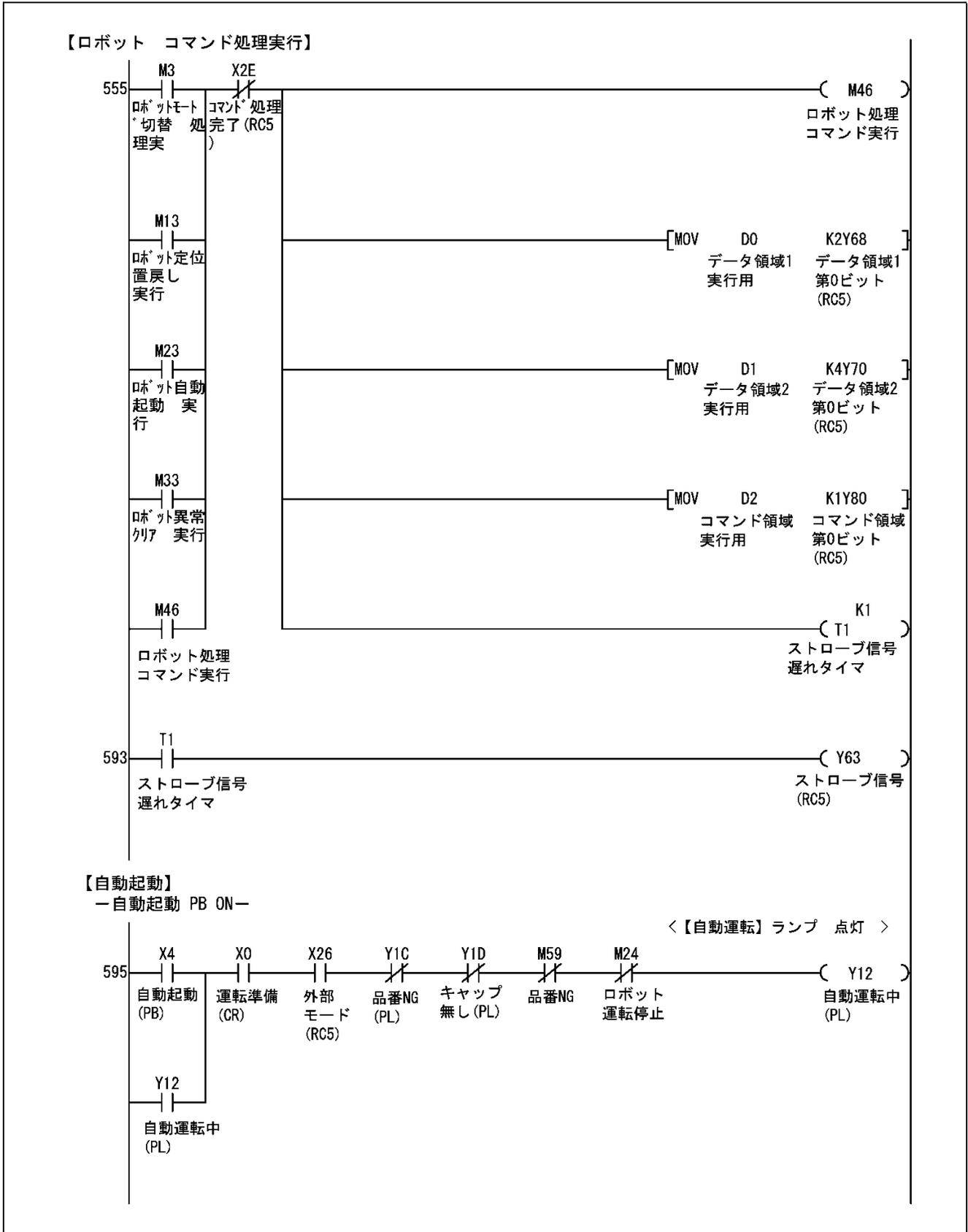
ピック & プレース動作応用プログラム例



シーケンスプログラムの例 (7/12)

(次ページへ続く)

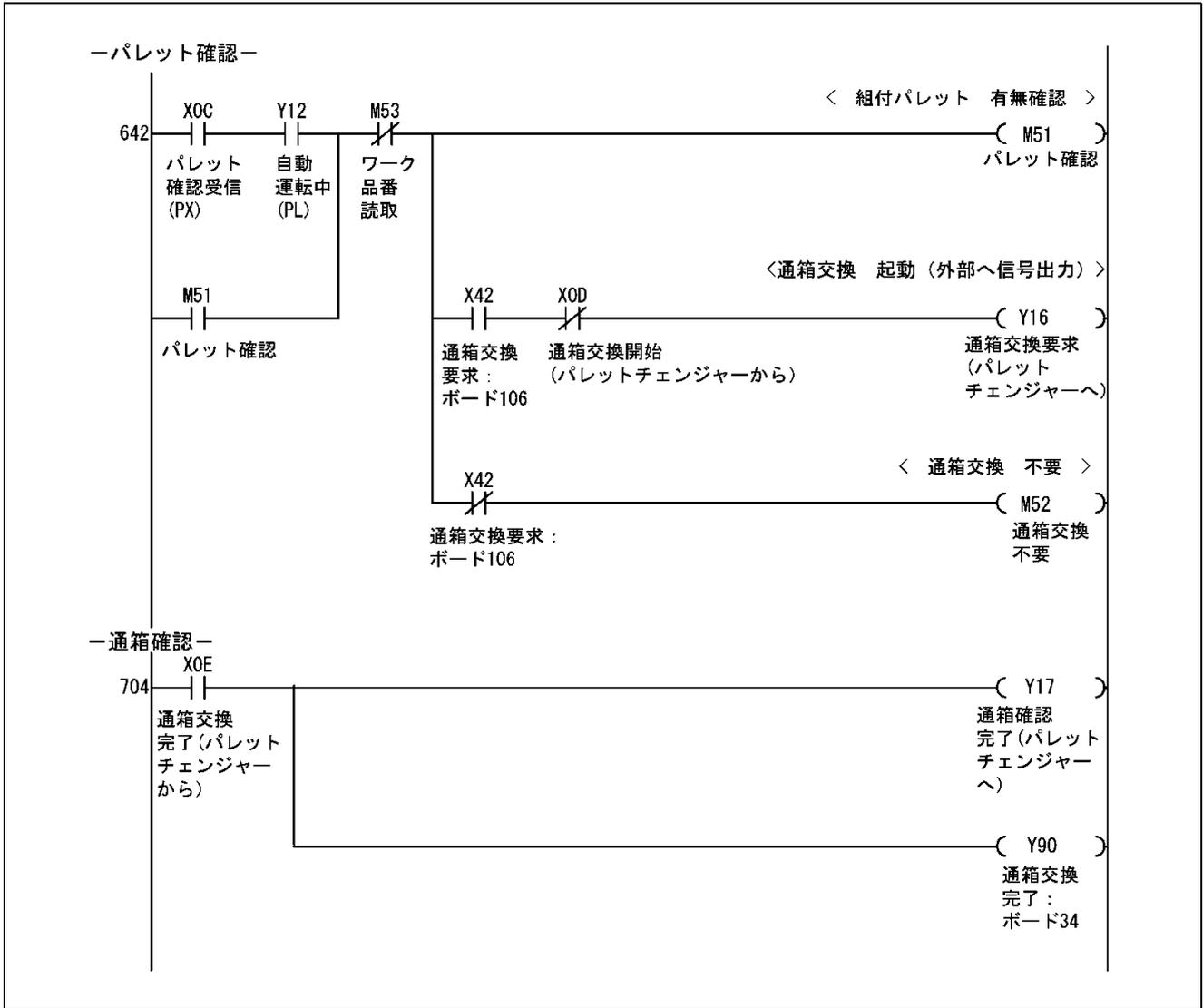
ピック & プレース動作応用プログラム例



シーケンスプログラムの例 ( 8/12 )

( 次ページへ続く )

ピック & プレス動作応用プログラム例

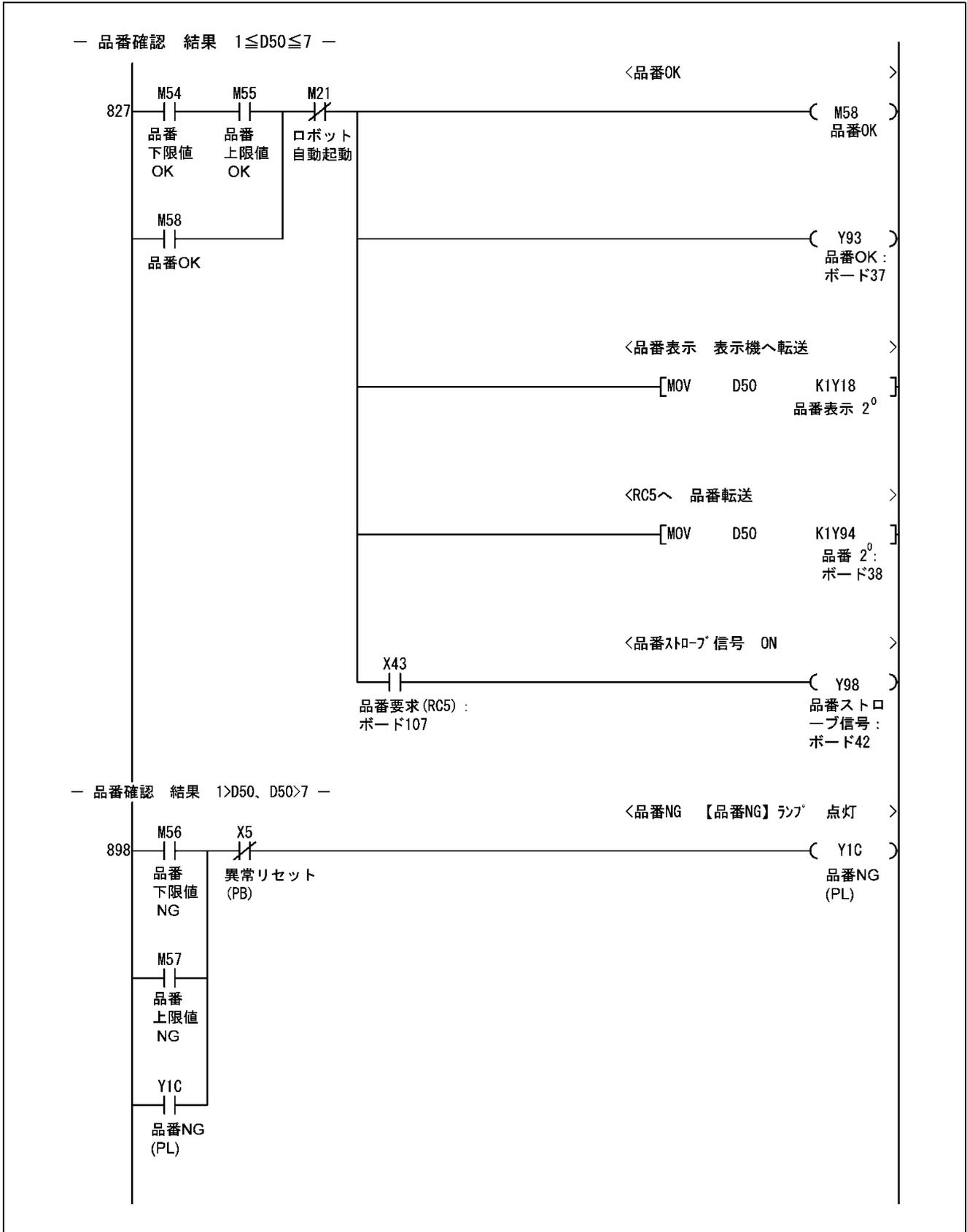


シーケンスプログラムの例 (9/12)

(次ページへ続く)



ピック & プレース動作応用プログラム例



シーケンスプログラムの例 (11/12)

(次ページへ続く)

