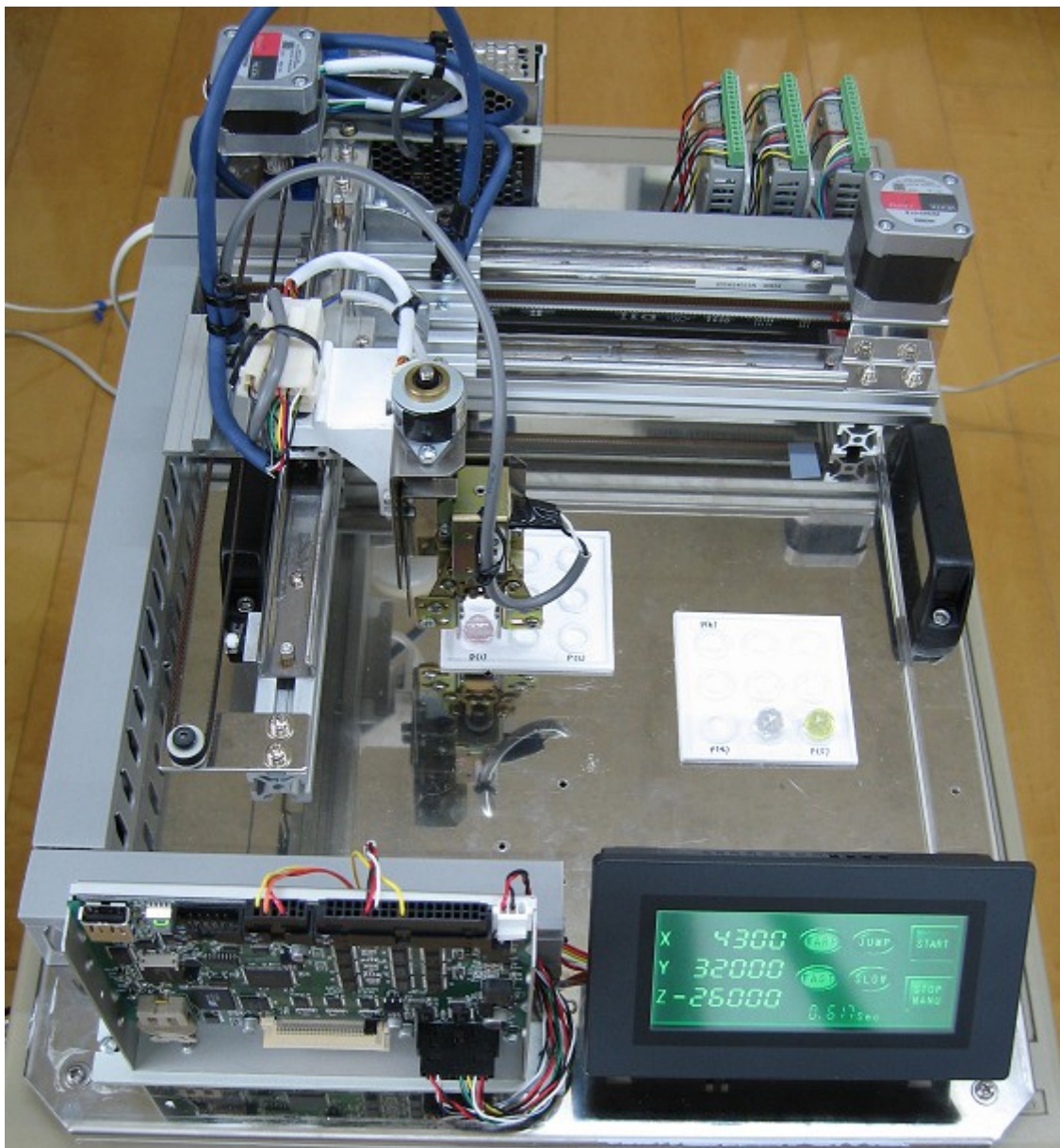


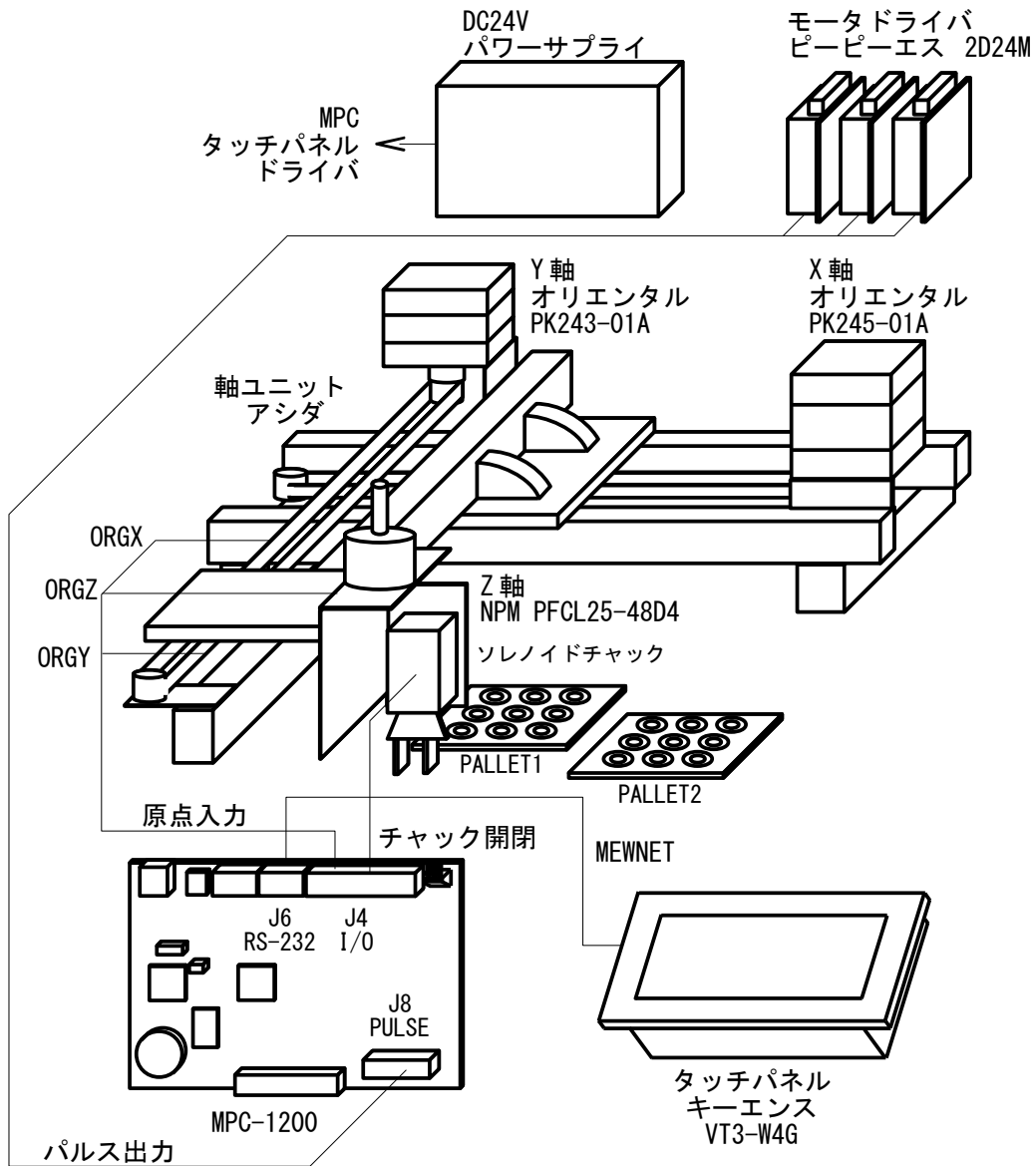
Application Note		Ref No: an2k-047	Last Modify 140704
テーマ	新型 XYZ デモ機 (XY-14)		
使用機器	MPC-1200		

概要

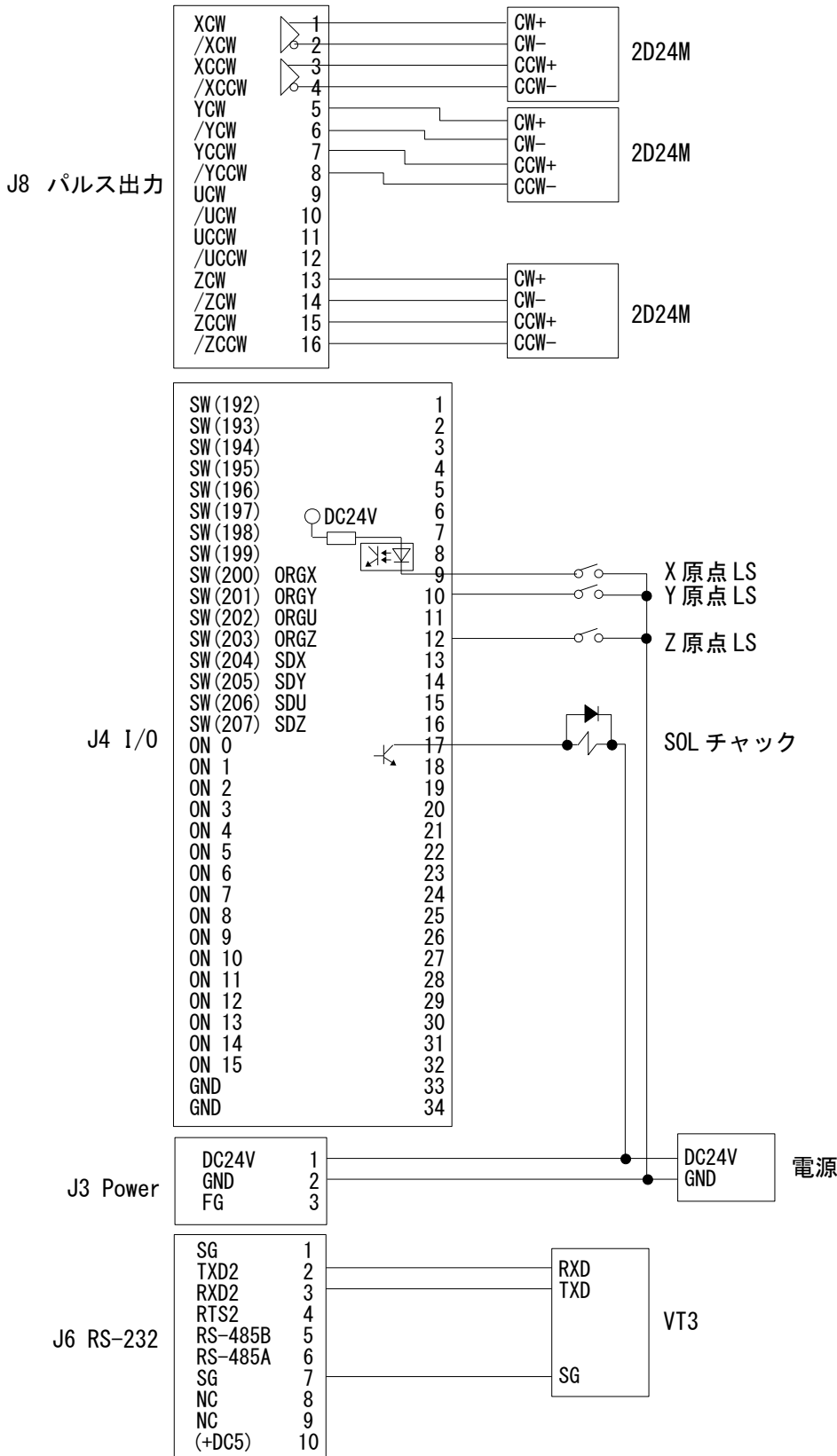
- MPC-1200 単体で制御する XYZ ロボットです。
- WARP モーションでピックアンドプレースを高速化します。
- マニュアルモードでタッチパネルでティーチングも行えます。



機器構成



MPC-1200 結線



MPC プログラム

```
QUIT_FORK 1 *Main
END /* TASK0 END

*Main
MEWNET 38400 2
MBK(4)=200 /* Touch Panel Page Control
PG 17

MBK(101)=100 /* Default JOG Distance
MBK(102)=1 /* Default JUMP Point No
MBK(103)=1 /* Default TEACH Point No
MBK(104)=0 /* Default Error Number

DO
OFF 70000 /* Start SW

page=MBK(4) /* タッチパネルのページで動作を切り替える
SELECT_CASE page
CASE 200 /* AUTO Page
QUIT_FORK 2 *Auto /* 自動モード
QUIT_FORK 3 *Coordinates /* 座標表示
QUIT_FORK 4 *SetWarp /* WARP/JUMP 切替
WAIT MBK(4) <>page
CASE 201 /* JOG Page
CASE 202 /* JUMP Page
CASE 203 /* TEACH Page
QUIT_FORK 2 *Manu /* マニュアルモード
WAIT MBK(4) <>page
CASE 204 /* ERROR Page
WAIT MBK(4) <>page
CASE_ELSE
END_SELECT

QUIT 2 3 4
STOP ALL_A STP_D
WAIT RR(ALL_A)==0

LOOP

*Manu /* マニュアル
PG 17
GOSUB *SetAccel
FEED X_A|Y_A 10
FEED Z_A 40
S_MBK " " 210 10

DO
SELECT_CASE VOID

CASE SW(70200) /* X+
RMVS X_A MBK(101)
WAIT RR(ALL_A)==0
WAIT SW(70200)==0

CASE SW(70201) /* X-
RMVS X_A MBK(101)*-1
WAIT RR(ALL_A)==0
WAIT SW(70201)==0

CASE SW(70202) /* Y+
RMVS Y_A MBK(101)
WAIT RR(ALL_A)==0
WAIT SW(70202)==0

CASE SW(70203) /* Y-
```

```

RMVS Y_A MBK(101)*-1
WAIT RR(ALL_A)==0
WAIT SW(70203)==0

CASE SW(70204)          /* Z+
  RMVS Z_A MBK(101)
  WAIT RR(ALL_A)==0
  WAIT SW(70204)==0

CASE SW(70205)          /* Z-
  RMVS Z_A MBK(101)*-1
  WAIT RR(ALL_A)==0
  WAIT SW(70205)==0

CASE SW(70206)          /* HOME
  GOSUB *HOME
  WAIT SW(70200)==0

CASE SW(70207)          /* JUMP OK (Jump Page)
  LIMZ 0
  S_MBK "Jumping to" 210 10
  JUMP P(MBK(102))
  WAIT RR(ALL_A)==0
  S_MBK " " " 210 10
  MBK(103)=MBK(102)
  MBK(4)=201

CASE SW(70208)          /* TEACH OK (Teach Page)
  SETP MBK(103) P(0)
  FOR d=0 TO 1
    S_MBK "Completed " 210 10
    TIME 250
    S_MBK " " " 210 10
    TIME 250
  NEXT
  MBK(4)=201

CASE SW(70209)          /* Chuck
  IF SW(0) THEN
    OFF 0
  ELSE
    ON 0
  END_IF
  WAIT SW(70209)==0

CASE_ELSE
END_SELECT
SWAP

```

LOOP

```
*Auto          /* 自動運転
PG 17
```

```

/*      PALLET1          PALLET2
/*
/*      7      8      9          7      8      9
/*      ⊙      ○      ○          ⊙      ○      ○
/*      P(3)          P(6)
/*      4      5      6          4      5      6
/*      ○      ○      ○          ○      ○      ○
/*
/*      pln=1      2      3      pln=1      2      3
/*      ⊙      ○      ⊙          ⊙      ○      ⊙
/*      P(1)          P(2)      P(4)          P(5)

```

```

PALLET 1 P(1) P(2) P(3) 3 3
PALLET 2 P(4) P(5) P(6) 3 3

```

```
ChuckDelay0=10          /* Chuck Open Delay
```

```

ChuckDelayC=50          /* Chuck Close Delay
ChuckDelayW=200        /* Chuck Open/Close Delay
ON 70001                /* WARP SW
ON 70002                /* FAST SW
OFF 0                   /* Chuck

```

```
DO
```

```

WAIT SW(70000)==1      /* Start SW
GOSUB *HOME
GOSUB *SetAcceI

```

```
DO
```

```
WAIT SW(70000)==1
```

```

MOVS X_A|Y_A P(2)
WAIT RR(ALL_A)==0
Iz=Z(1)+15000
LIMZ Iz

```

```

GOSUB *Pallet1to2 3 1
GOSUB *Pallet2to1 1 3 3

```

```

GOSUB *Pallet1to2 6 4
GOSUB *Pallet2to1 4 6 3

```

```

GOSUB *Pallet1to2 9 7
GOSUB *Pallet2to1 7 9 -6

```

```

MOVS Z_A 0
WAIT RR(ALL_A)==0

```

```
GOSUB *Z0rgCheck      /* Z軸ずれチェック
```

```

FEED ALL_A fed
MOVS 50000 0 0 0
WAIT RR(ALL_A)==0
MOVS 50000 50000 0 0
WAIT RR(ALL_A)==0
MOVS 0 50000 0 0
WAIT RR(ALL_A)==0
MOVS 0 1000 0 0
WAIT RR(ALL_A)==0
MOVS 0 50000 0 0
WAIT RR(ALL_A)==0
MOVS 50000 50000 0 0
WAIT RR(ALL_A)==0
MOVS 50000 0 0 0
WAIT RR(ALL_A)==0
MOVS 0 0 0 0
WAIT RR(ALL_A)==0
TIME 100

```

```
LOOP
```

```
LOOP
```

```

*Pallet1to2           /* PALLET1 -> PALLET2
_VAR sp_ ep_

```

```
FOR pln=sp_ TO ep_ STEP -1
```

```

IF (pln<>sp_)|(sp_==3) THEN /* 3 は一番最初のPickup point
FEED ALL_A fed
WARP up_z1 PL(1;pln) dwn_z1
WAIT RR(ALL_A)==0
ELSE
TIME ChuckDelayW
END_IF

```

```

WAIT SW(70000)==1

ON 0
TIME ChuckDelayC
stclk=SYSCLK

FEED ALL_A fed
WARP up_z2 PL(2;pln) dwn_Z2
WAIT RR(ALL_A)==0

OFF 0
MBK(300)=SYSCLK-stclk
TIME ChuckDelay0

WAIT SW(70000)==1

NEXT

RETURN

*Pallet2to1                      /* PALLET2 -> PALLET1
  _VAR sp_ ep_ of_

FOR pln=sp_ TO ep_

  IF pln<>sp_ THEN
    FEED ALL_A fed
    WARP up_z1 PL(2;pln) dwn_Z1
    WAIT RR(ALL_A)==0
  ELSE
    TIME ChuckDelayW
  END_IF
  WAIT SW(70000)==1

  ON 0
  TIME ChuckDelayC
  stclk=SYSCLK

  FEED ALL_A fed
  WARP up_z2 PL(1;pln+of_) dwn_Z2
  WAIT RR(ALL_A)==0

  OFF 0
  MBK(300)=SYSCLK-stclk
  TIME ChuckDelay0

  WAIT SW(70000)==1

NEXT

RETURN

*HOME                               /* 原点復帰
ACCEL X_A|Y_A 10000 1000 500
ACCEL Z_A 8000 500 100

IF SW(203)==1 THEN
  RMVS Z_A -1000
  WAIT RR(Z_A)==0
END_IF

HOME Z_A POS_L
WAIT RR(ALL_A)==0

IF SW(200)==1 THEN
  RMVS X_A 5000
END_IF
IF SW(201)==1 THEN
  RMVS Y_A 5000
END_IF

```

```

WAIT RR(ALL_A)==0
TIME 500

HOME X_A|Y_A NEG_L
WAIT RR(ALL_A)==0

RMVS X_A|Y_A 1000          /* OFF SET
WAIT RR(ALL_A)==0

CLRPOS
TIME 500

RETURN

*SetAccel                /* 加減速、最高速設定
ACCEL X_A|Y_A 12000 12000 500
ACCEL Z_A 60000 3000 100
RETURN

*SetWarp                 /* WARP/JUMP 切替
PG 17
ON 70002                 /* FAST/SLOW

DO

  IF SW(70001) THEN
    up_z1=10000          /* 取りに行く時の上昇 置いたワークの上まで上がる
    dwn_Z1=10000        /* 取りに行く時の下降 取るワークの上から入る
    up_z2=4000          /* 置きに行く時の上昇
    dwn_Z2=4000        /* 置きに行く時の下降
    yoko_z=10000
  ELSE
    up_z1=0
    up_z2=0
    dwn_Z1=0
    dwn_Z2=0
    yoko_z=0
  END_IF

  IF SW(70002) THEN
    fed=100
  ELSE
    fed=50
  END_IF
  SWAP

LOOP

*ZOrgCheck              /* Z軸ずれチェック
MOVS Z_A -600
WAIT RR(ALL_A)==0
TIME 100
IF SW(203)==1 THEN
  MBK(104)=1           /* Error Number
  MBK(4)=204
END_IF
MOVS Z_A 500
WAIT RR(ALL_A)==0
TIME 100
IF SW(203)==0 THEN
  MBK(104)=2           /* Error Number
  MBK(4)=204
END_IF
TIME 100

RETURN

*Coordinates            /* 座標表示
PG 17

```


DO
 MBK (200~Lng)=X (0)
 MBK (202~Lng)=Y (0)
 MBK (204~Lng)=Z (0)
 SWAP
 LOOP

タッチパネル画面

• Auto (画面 No200)

[部品]	[MPC 表記]	[パネル表記]
START	SW (70000)	R0000
STOP/MANU		ページ切替 No201
WARP, JUMP	SW (70001)	R0001
FAST, SLOW	SW (70002)	R0002
X 軸座標	MBK (200~Lng)	DT00200
Y 軸座標	MBK (202~Lng)	DT00202
Z 軸座標	MBK (204~Lng)	DT00204
動作時間	MBK (300)	DT00300



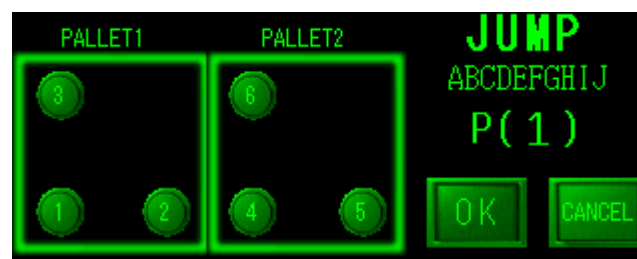
• JOG (画面 No201)

X+	SW (70200)	R0020
X-	SW (70201)	R0021
Y+	SW (70202)	R0022
Y-	SW (70203)	R0023
Z+	SW (70204)	R0024
Z-	SW (70205)	R0025
HOME	SW (70206)	R0026
Chuck	SW (70209)	R0029
JUMP		ページ切替 No202
TEACH		ページ切替 No203
AUTO		ページ切替 No200
10, 100, 1000	MBK (101)	DT101



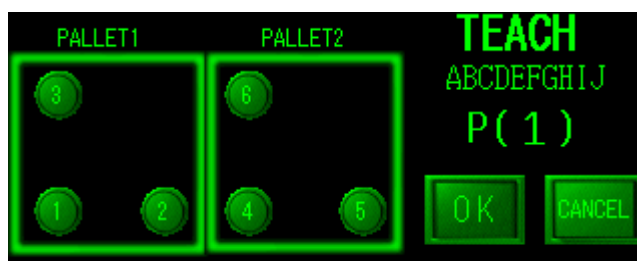
• JUMP (画面 No202)

OK	SW (70207)	R0027
CANCEL		ページ切替 No201
P (1) ~ (6)	MBK (102)	DT (102)
文字表示	MBK (210)	DT (210) ~ (215)



• TEACH (画面 No203)

OK	SW (70208)	R0028
CANCEL		ページ切替 No201
P (1) ~ (6)	MBK (103)	DT (103)
文字表示	MBK (210)	DT (210) ~ (215)



WARP モーション

WARP は高速ゲートモーションです。

Z 軸上昇中に XY 軸が移動開始、XY 軸が移動中に Z 軸が下降開始します。

WARP の第一、第三パラメータに 0 を与えると JUMP コマンドと同じ動作になります(例 WARP 0 P(2) 0)
 本機の場合、1 回の P&P(ワークを取ってから置くまで)が、WARP で 0.617sec、JUMP では 1.005sec でした。

- WARP 動作例 (MPG-2314)

```

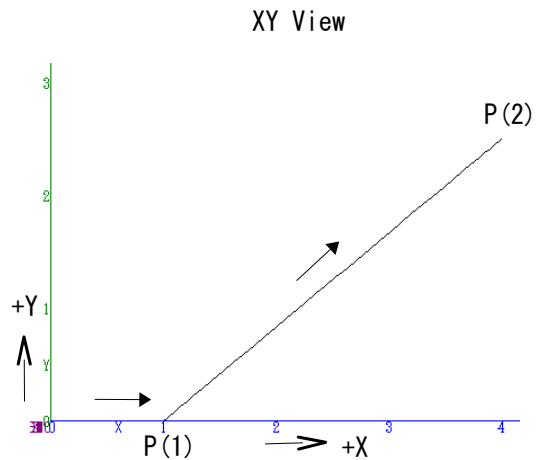
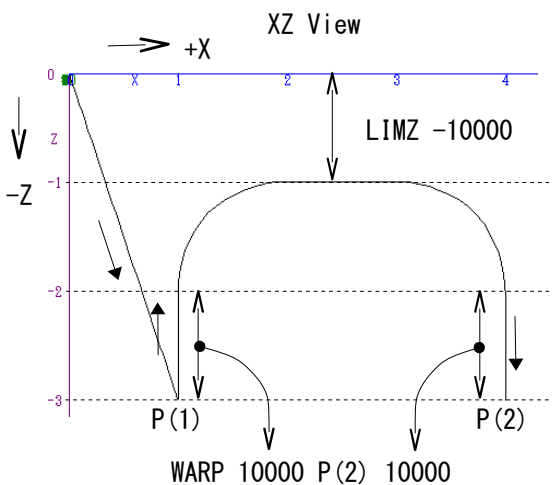
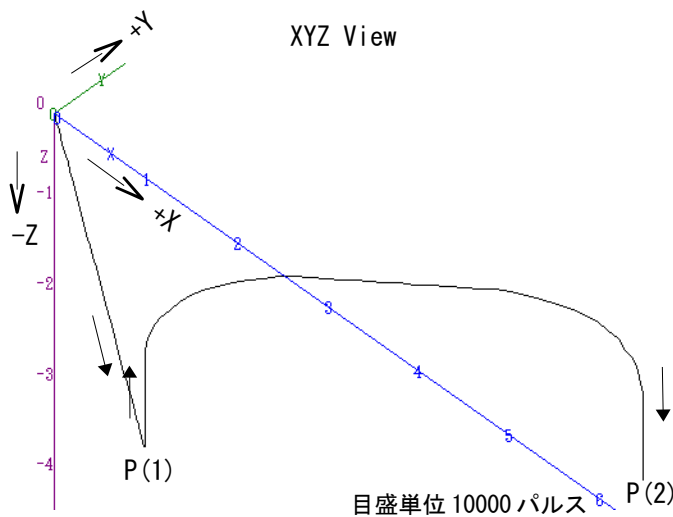
PG 0 /* PG ボードは MPG-2314
ACCEL ALL_A 50000 5000 500 /* 加減速
FEED ALL_A 100
SETP 1 10000 0 0 -30000 /* P(1) Z 軸はマイナス方向
SETP 2 40000 25000 0 -30000 /* P(2) Z 軸はマイナス方向
LIMZ -10000 /* Z 上昇制限
CLRPOS /* 現在位置クリア (原点復帰の代わり)
OFF 0 1
    
```

```

MOVL P(1) /* P(1) へ移動
WAIT RR(ALL_A)==0
    
```

```

WARP 10000 P(2) 10000 /* P(2) へ WARP 移動
    
```



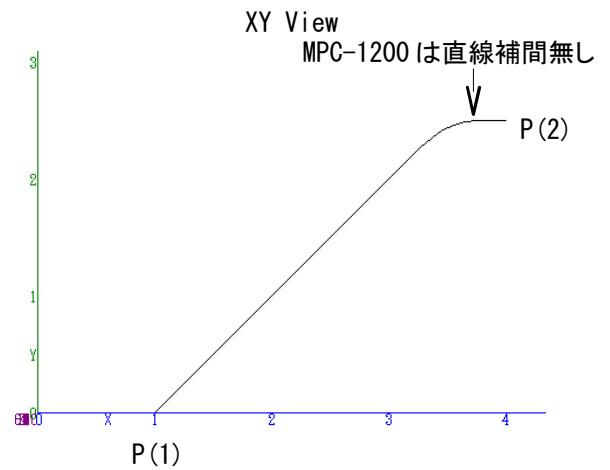
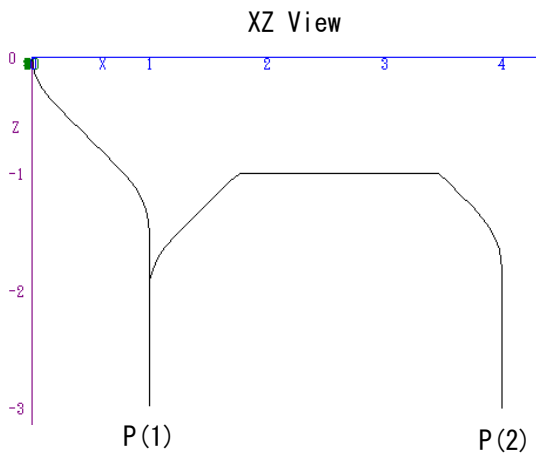
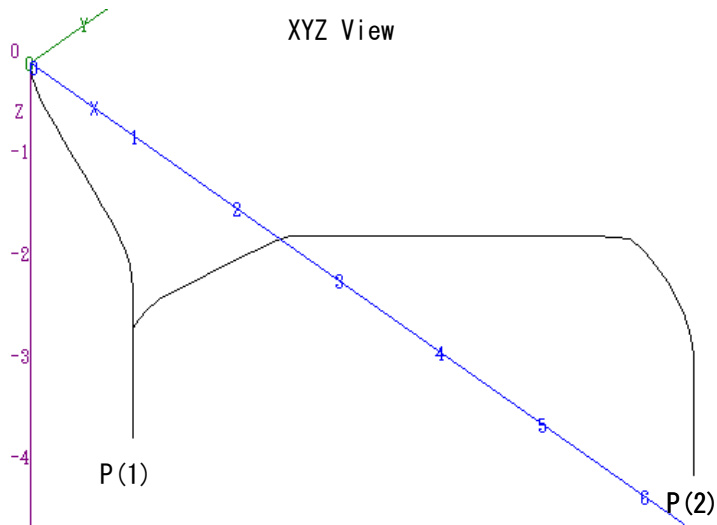
- WARP 動作例 (MPC-1200)

```
PG 17
ACCEL X_A|Y_A 50000 5000 500
ACCEL Z_A 50000 500 500
FEED ALL_A 100
```

```
SETP 1 10000 0 0 -30000
SETP 2 40000 25000 0 -30000
LIMZ -10000
CLRPOS
```

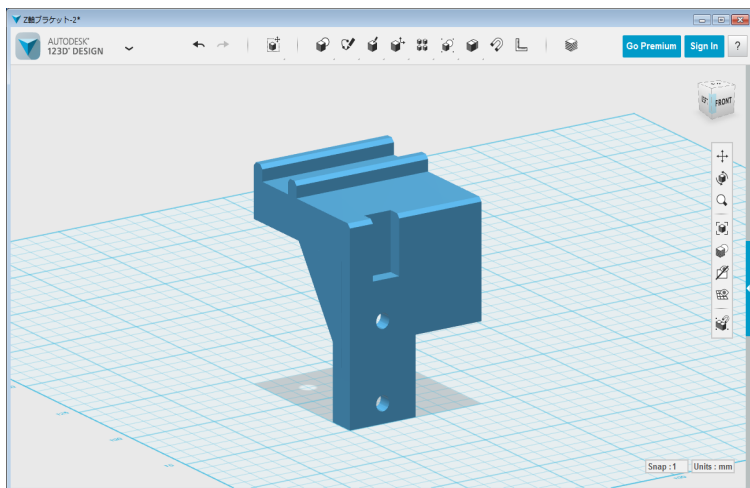
```
MOVS X(1) Y(1) VOID Z(1)
WAIT RR(ALL_A)==0
```

WARP 10000 P(2) 10000

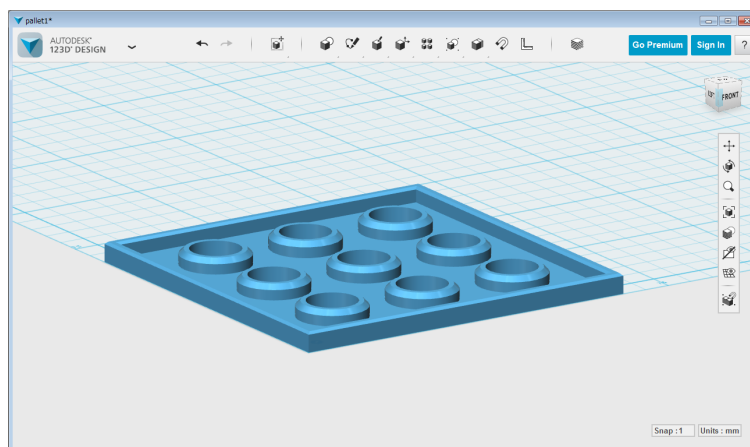
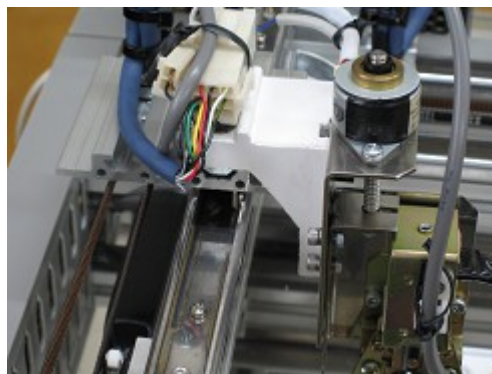


3D 部品

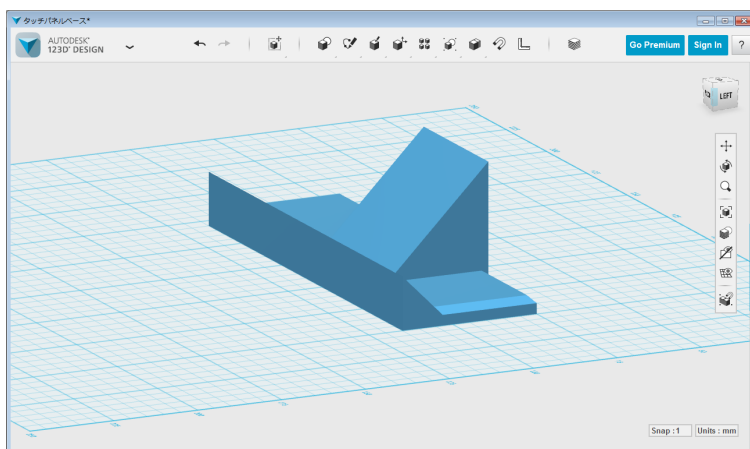
MPCの機能とは関係ありませんが、一部の部品を3Dプリンタ(XYZ printing社 ダヴィンチ1.0)で作ってみました。(出来映えはお値段なりってところですか...)



Z軸ブラケット



パレット



タッチパネルベース

