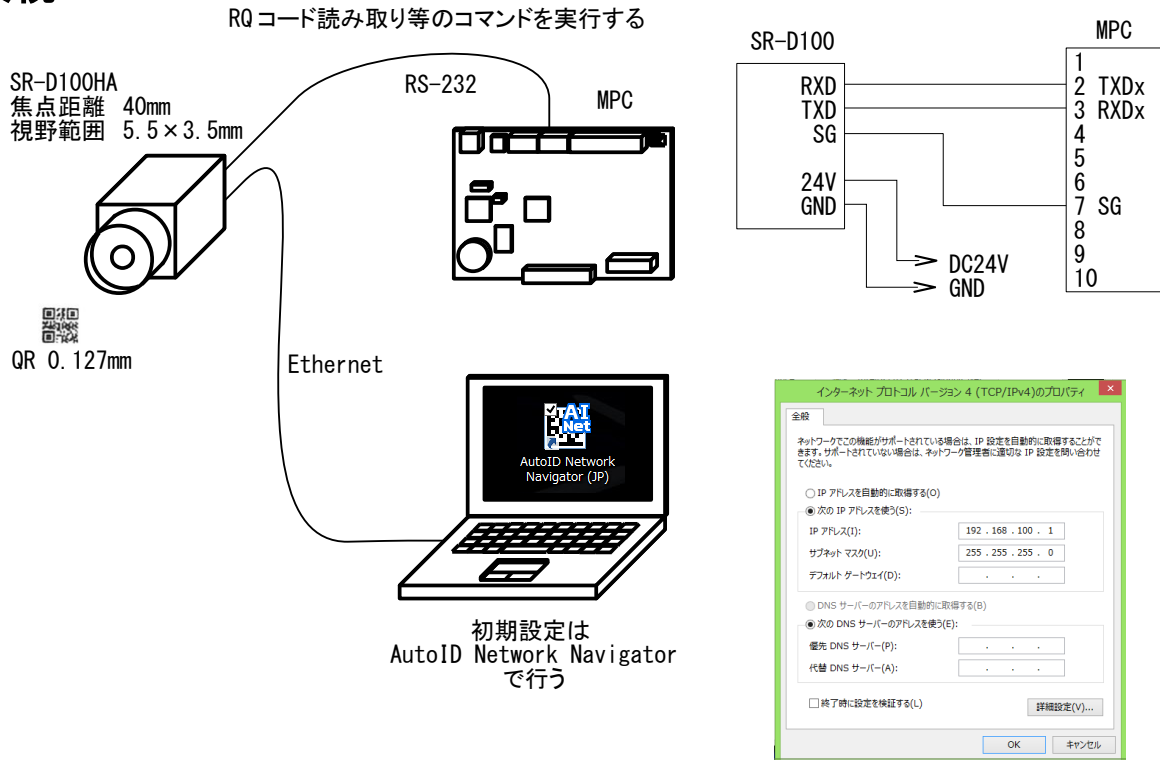


接続



MPC プログラム

- SW(192) をオンすると入力(SW(196) ~ (199)) に接続した DSW でコマンドを選択して実行します。
- 結果は PRINT で出力、タッチパネルにも表示します。

```

/* 「返信」のフォーマット
/*   OK, 送信コマンド, a, b, c, ---<cr>
/*   ER, 送信コマンド, エラーコード, <cr>
/*
/* 送信コマンドと<cr>間のデータは a() 配列に格納する。
/* 例) NUM の場合 "OK, 送信コマンド, aaaaa, bbbbb, ccccc, ddddd, eeeee<cr>"
/*
/*           a(0) a(1) a(2) a(3) a(4)
/*
/* LON, TUNE, TEST1 はターミネータの後(改行して)データが来る。
/* 例) LON, 01<cr>           /* コマンド送信           PRINT#
/*   OK, LON<cr>           /* 「返信」受信           INPUT#(1)
/*   12345678<cr>         /* 読み取りデータ受信 INPUT#(2)
/* INPUT#(2) の TMOU 設定時間内にデータが来ない場合はリトライする。

```

```
TIME 1000           /* SD-D100 起動待ち
```

```
QUIT_FORK 1 *main
END
```

```

*main
DIM a(20)           /* 「返信」の数値データ格納用
rsch=8
CNFG# rsch "38400b8pes1NONE"
MEWNET 38400 5

```

```

read_cnt=0
read_err=0
DO

```

WAIT SW(192)==1

job=IN(24)>>4 /* IN(24)の上位ニブルにDSWをつなぐ
SELECT_CASE job

```
CASE 0
PRINT "リセット"
S_MBK " リセット " 100 16
GOSUB *CLR_DISP
commnd$="RESET" /* RESET 後、再起動までに時間がかかる(本体LED参照)
tmot=0 /* 「返信」の後の「結果」が無い
GOSUB *SRD100
```

```
CASE 1
PRINT "コード読み取り"
S_MBK " コード読み取り " 100 16
commnd$="LON,01" /* タイミングON
tmot=2 /* 「返信」の後の「結果」が有る。TMOUT 1secだとギリ
GOSUB *SRD100
PRINT code$ /* エラーなら空
IF code$=="12345678" THEN
read_cnt=read_cnt+1
tmp$=STR$(read_cnt)+" "+code$
S_MBK tmp$ 110 16
ELSE
S_MBK "Error " 110 16
read_err=read_err+1
END_IF
tmp$="Error "+STR$(read_err)
S_MBK tmp$ 120 16
```

```
CASE 2
PRINT "ポインタ ON"
S_MBK " ポインタ ON " 100 16
GOSUB *CLR_DISP
commnd$="AMON" /* ポインタ ON
tmot=0 /* 「返信」の後の「結果」が無い
GOSUB *SRD100
```

```
CASE 3
PRINT "ポインタ OFF"
S_MBK " ポインタ OFF " 100 16
GOSUB *CLR_DISP
commnd$="AMOFF" /* ポインタ OFF
tmot=0 /* 「返信」の後の「結果」が無い
GOSUB *SRD100
```

```
CASE 4
PRINT "チューニング"
S_MBK " チューニング " 100 16
GOSUB *CLR_DISP
commnd$="TUNE,01" /* 「返信」の後の「結果」が有る、TUNE は「結果」が来るのが遅い
tmot=5
GOSUB *SRD100
PRINT code$
S_MBK code$ 110 16
STRCPY code$ tmp$ 17
S_MBK tmp$ 120 16
```

```
CASE 5
PRINT "読み取り成功、エラー回数履歴"
S_MBK " 成功、エラー履歴 " 100 16
GOSUB *CLR_DISP
commnd$="NUM" /* 読み取り成功、エラー回数履歴
tmot=0 /* NUM は「結果」が無いが「返信」にデータがくっ付いている
GOSUB *SRD100
tmp$=""
FOR i=0 TO 4
PR "a(" i ")=" a(i) /* 「返信」の aaaaa=a(0)~eeeeee=a(4)
tmp$=tmp$+STR$(a(i))+""
```

```
NEXT
S_MBK tmp$ 110 16
```

CASE 6

```
PRINT "バンク使用回数履歴"
S_MBK "バンク使用回数履歴" 100 16
/*S_MBK " Bank Log " 100 16
GOSUB *CLR_DISP
cmdnd$="NUMB" /* バンク使用回数履歴
tmot=0 /* NUMBは「結果」が無いが「返信」にデータがくっ付いている
GOSUB *SRD100
FOR i=0 TO 16
PR "a(" i ")=" a(i) /* 「返信」の BANK1=a(0)~TIMING=a(16)
NEXT
tmp$=STR$(a(0))+ " "+STR$(a(1))+ " "+STR$(a(2))+ " "+STR$(a(3))
S_MBK tmp$ 110 16
tmp$=STR$(a(16))
S_MBK tmp$ 120 16
```

CASE 7

```
PRINT "読み取り率測定"
GOSUB *CLR_DISP
S_MBK "読み取り率測定" 100 16
cmdnd$="TEST1,01" /* 読み取り率測定テストモード
tmot=4 /* TEST1は「結果」が連続で来る
GOSUB *SRD100
PRINT code$
STRCPY code$ tmp$ 0 12
S_MBK tmp$ 110 16
STRCPY code$ tmp$ 12
S_MBK tmp$ 120 16

TIME 1000
cmdnd$="QUIT"
tmot=0 /* 「返信」の後の「結果」が無い
GOSUB *SRD100
```

CASE 8

```
PRINT "MACアドレス"
GOSUB *CLR_DISP
S_MBK " MACアドレス " 100 16
cmdnd$="EMAC"
tmot=0 /*
GOSUB *SRD100
PRINT res2$
S_MBK res2$ 110 16
```

CASE 9

```
PRINT "時計"
GOSUB *CLR_DISP
S_MBK " 時計 " 100 16
cmdnd$="TMGET"
tmot=0 /*
GOSUB *SRD100
PRINT res2$
S_MBK res2$ 110 16
```

```
CASE_ELSE
PR "??"
```

```
END_SELECT
```

```
IF job<>1 THEN
TIME 500
END_IF
```

```
LOOP
```

```
*CLR_DISP
S_MBK " " 110 16
```

```
S_MBK " " 120 16
RETURN
```

```
/* SR-D100HA と通信
*SRD100
```

```
try=0
DO
  res0$=""
  res1$=""
  res2$=""
  res3$=""
  code$=""
  cc=0 /* comma count
  FILL a(0) 20

  INPUT# rsch CLR_BUF
  PRINT# rsch commnd$ "¥r"
  INPUT# rsch TMOUT|2 res$
  IF rse_!=0 THEN /* コマンドを受け付けてくれない。カメラ動作してない、断線 等
    try=-3
    BREAK /* exit from DO~LOOP
  END_IF
```

```
/* 受信した文字列を1キャラずつ調べて、でセパレートする
ptr_=res$ /* res$ 先頭位置
FOR i=1 TO LEN(res$)
```

```
  a$=PTR$(1) /* a$ <- 1文字
  ptr_=ptr_+1
```

```
  IF a$="," THEN
    cc=cc+1
  ELSE
    SELECT_CASE cc
      CASE 0
        res0$=res0$+a$ /* OK または ER
      CASE 1
        res1$=res1$+a$ /* 送信コマンド
      CASE 2
        res2$=res2$+a$ /* エラーコード 等
      CASE_ELSE
        res3$=res3$+a$
    END_SELECT
  END_IF
```

```
NEXT
```

```
GET_VAL res$ a(0) /* 「返信」の数字部分を配列に入れる。NUM, NUMB コマンド
```

```
/* PR res$ "/" res0$ res1$ res2$ res3$
```

```
SELECT_CASE res0$
CASE "OK"
```

```
  IF tmot==0 THEN /* 「結果」が無いコマンド
    BREAK /* exit from DO~LOOP
  END_IF
```

```
  INPUT# rsch TMOUT|tmot code$ /* 結果の文字列取得
  IF rse_==0 THEN /* 正常に結果が取得できた
    try=0
    BREAK /* exit from DO~LOOP
  ELSE /* タイムアウトしたらリトライ
    PRINT# rsch "LOFF¥r"
    try=try+1
    TIME 1000
    IF try>2 THEN /* リトライしても読めなかった
      BREAK /* exit from DO~LOOP
    END_IF
```

```

        END_IF
    CASE "ER"
        PRINT# rsch "LOFF¥r"          /* need?
        try=-1
        TIME 1000
        BREAK                          /* exit from DO~LOOP

    CASE_ELSE
        try=-2
        BREAK                          /* exit from DO~LOOP

END_SELECT

LOOP

SELECT_CASE try
CASE -3
    PRINT "[Command Send Timeout]"
CASE -2
    PRINT "[Unexpected error]"
CASE -1
    PRINT "[ER] Error Code=" res2$
CASE 0
    PRINT "[OK]"
CASE_ELSE
    PRINT "[Failed] Try Count=" try
END_SELECT

RETURN

```

実行結果

```

コード読み取り
[OK]
12345678
ポインタ ON
[OK]
ポインタ OFF
[OK]
チューニング
[OK]
Tuning SUCCEEDED, 1889ms, 00000000
読み取り成功、エラー回数履歴
[OK]
a( 0 )= 25
a( 1 )= 0
a( 2 )= 0
a( 3 )= 0
a( 4 )= 25
バンク使用回数履歴
[OK]
a( 0 )= 25
a( 1 )= 0
a( 2 )= 0
a( 3 )= 0
a( 4 )= 0
a( 5 )= 0
a( 6 )= 0
a( 7 )= 0
a( 8 )= 0
a( 9 )= 0
a( 10 )= 0
a( 11 )= 0
a( 12 )= 0
a( 13 )= 0
a( 14 )= 0
a( 15 )= 0

```

a(16)= 25
読み取り率測定
[OK]
12345678:01:100%:90
[OK]
MAC アドレス
[OK]
0001FC111DDD
時計
[OK]
20141127154701