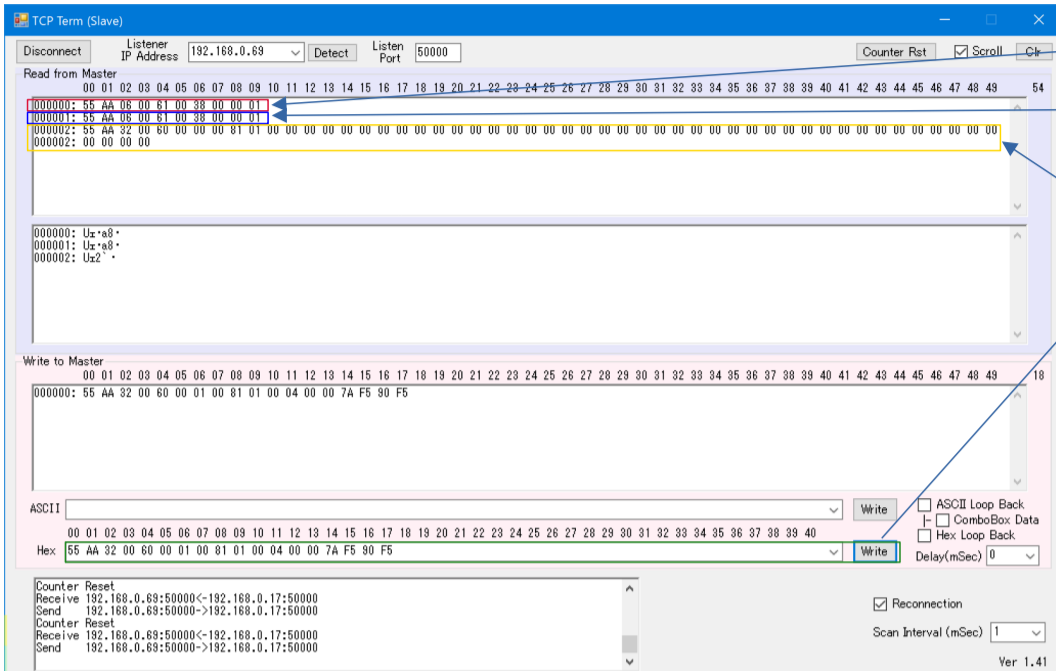


TcpTermS.exe (接続機器の代替)



MpcMpnitor.exe

```

LIST 0 0
10 SET_IP 192 168 0 17 255 255 255 0 192 168 0 248
20 SET_IP TCP Telnet
30 SET_IP TCP PACKET 50000 192 168 0 69
40 TCP_RO$=""
50 send$=CHR$(&H0000055)+CHR$(&H00AA)+CHR$(&H0006)+"¥0"+CHR$(&H0061)+"¥0"+CHR$(&H0038)+"¥0¥0"+CHR$(&H0001)
60 TCP_XO$=send$
70 WAIT TCP_XO$=""
80 send$="¥x55¥xAA¥x06¥0¥x61¥0¥x38¥0¥0¥x01"
90 TCP_XO$=send$
100 WAIT TCP_XO$=""
110 send$=CHR$(&H0055)+CHR$(&H00AA)+CHR$(&H0032)+"¥0"+CHR$(&H0060)+"¥0¥0¥0"+CHR$(&H0081)+CHR$(&H0001)
120 FOR i=0 TO 43
130 send$=send$+"¥0"
140 NEXT
150 TCP_XO$=send$
160 WAIT TCP_RO$<>""
170 len=IPC(TCP_RO$)
180 FOR i=0 TO len-1
190 PRINT i HEX$(PEEK(TCP_RO$+i)&&h0FF)
200 NEXT
#RUN

# 0 00000055
1 000000AA
2 00000032
3 00000000
4 00000060
5 00000000
6 00000001
7 00000000
8 00000081
9 00000001
10 00000000
11 00000004
12 00000000
13 00000000
14 0000007A
15 000000F5
16 00000090
17 000000F5
    
```

50行はCHR\$()を連結してありますがその数は10個程度です。
CHR\$()ではNULLが扱えないのでエスケープ文字を使います。
80行は50行と同じことをエスケープ文字でやっています。

エスケープ文字には次のものがあります。

- ¥r CR (&H0D)
- ¥n LF (&H0A)
- ¥t TAB (&H09)
- ¥0 NULL (&H00)
- ¥xhh ヘキサコードhh。必ず2桁で指定。¥rと¥x0Dは同じ。

60行のように予約文字列変数 TCP_XO\$ に文字列を入れるとTCP送信されます。

受信データにNULLやASCIIコード以外があると通常の文字列変数では処理できないので PEEK() で1キャラクタずつ取り込んでいます。

■ MPCプログラムソース

```

SET_IP 192 168 0 17 255 255 255 0 192 168 0 248
SET_IP TCP Telnet
SET_IP TCP PACKET 50000 192 168 0 69
    
```

```
TCP_RO$=""
```

```

send$=CHR$(&H55)+CHR$(&hAA)+CHR$(&h06)+"¥0"+CHR$(&h61)+"¥0"+CHR$(&h38)+"¥0¥0"+CHR$(&h01) /* 送信するデータを作成
TCP_XO$=send$ /* 送信
WAIT TCP_XO$="" /* 送信完了待ち
    
```

```

send$="¥x55¥xAA¥x06¥0¥x61¥0¥x38¥0¥0¥x01" /* エスケープ文字でデータを作成
TCP_XO$=send$
WAIT TCP_XO$=""
    
```

```

send$=CHR$(&h55)+CHR$(&hAA)+CHR$(&h32)+"¥0"+CHR$(&h60)+"¥0¥0¥0"+CHR$(&h81)+CHR$(&h01)
FOR i=0 TO 43
    send$=send$+"¥0"
NEXT
TCP_XO$=send$
WAIT TCP_XO$=""
    
```

```

WAIT TCP_RO$<>"" /* 受信待ち
len=IPC(TCP_RO$) /* 受信文字列長
FOR i=0 TO len-1
    PR i HEX$(PEEK(TCP_RO$+i)&&hFF) /* 1キャラクタずつ入力
NEXT
    
```