

# バイナリデータの送受信例

ACCEL ti2k-190724 (LastUpdate Jul 29, 2019)



## ■ 送受信フォーマット

Byte No.	0	1	2	3	4
Character	STX (0x02)	NUL (0x00)	dataH	dataL	ETX (0x03)

## ■ プログラム

このサンプルでは配列に格納したバイナリデータをPOKEで文字列化、STR\_LEN でキャラクタ数を指定して送信します。  
受信時は CHR\_C でキャラクタ数を指定、PEEK で必要部分を取り出して数値化します。

```
CNFG# 2 "38400b8pns1NONE"
/* CNFG# RS485 2 "38400b8pns1NONE" /* RS-485の場合
TIME 500

DIM a(5)
QUIT_FORK 1 *main
END
*main

dat=&h1234 /*32767          /* data
a(0)=2          /* STX
a(1)=0          /* NUL
a(2)=(dat&&hFF00)>>8 /* dataH
a(3)=dat&&hFF    /* dataL
a(4)=3          /* ETX
send$=""        /* 送信文字列初期値 Max256byte
FOR ac=0 TO 4
  POKE a(ac) (send$+ac) /* 送信文字列作成
NEXT

INPUT# 2 CLR_BUF          /* 受信バッファクリア
PRINT# 2 STR_LEN|ac send$ /* 送信
PR "Send Data" HEX$(dat) dat

INPUT# 2 CHR_C|5 res$    /* 受信
res=(PEEK(res$+2)&&hFF)<<8 /* dataH
res=res+(PEEK(res$+3)&&hFF) /* dataL
PR "Receive Data" HEX$(res) res
```

## ■ 実行結果

・ dat=&h1234の時  
0x02 0x00 0x12 0x34 0x03 と送受信

RUN

```
# Send Data 00001234 4660
  Receive Data 00001234 4660
```

・ dat=32767の時  
0x02 0x00 0x7F 0xFF 0x03 と送受信

RUN

```
# Send Data 00007FFF 32767
  Receive Data 00007FFF 32767
```