

RS - 232C / GP - IB 活用ガイド

改版履歴

版数	改版日付	変更箇所	変更内容
3	00/11/09	動作確認済みボード そのほか	インターフェース PCI-4301 対応 ホームページアドレスなどを最新情報に更新
2	99/4/6	プログラム4および関する内容	アドバンテスト R 6552 対応

COPYRIGHT(C) FUJITSU LIMITED 1998,2000

Microsoftは、米国マイクロソフト社の米国及びその他の国における登録商標です。
その他記載されている会社及び商品名は各社の商標です。

はじめに

F-BASIC V6/6.3(以下、F-BASIC)を使った RS-232C や GP-IB インターフェースの制御について解説します。

パソコンと機器を RS-232C や GP-IB に経由して接続し、データを取得、表示、保存するには F-BASIC が最適です。たとえば、RS-232C とモデムを接続してパソコン通信を行ったり、GP-IB と計測機器を接続してデータを採取することができます。

ここでは、いくつかの例をつかって F-BASIC による RS-232C や GP-IB の使い方について、具体的に説明していきます。

なお、詳細については説明を一部省略しています。また、実際にプログラムを作成する場合は、F-BASIC に添付されているマニュアルも利用してください。

プログラムの入手方法

ここで紹介したプログラムは、すべてインターネットから入手することができます。

F-BASIC ホームページ: 製品情報、FAQ、技術情報などの最新情報

<http://www.fujitsu.co.jp/hypertext/fps/Products/fbasic/fbasic.html>

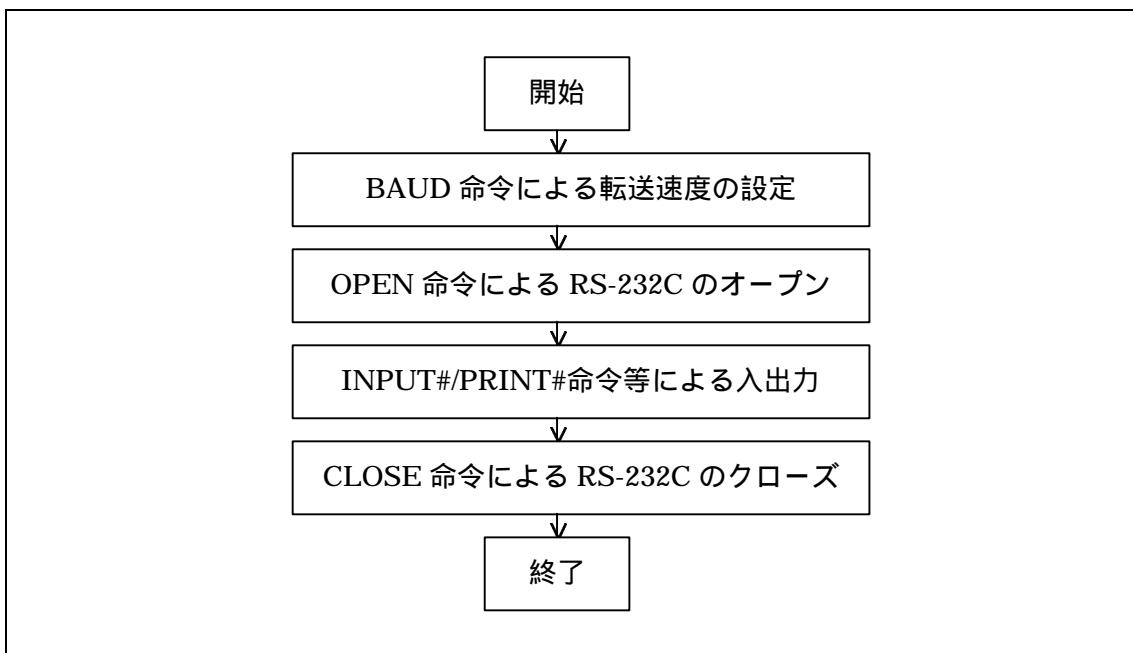
1.RS-232C

1- .RS-232Cについて

RS-232C は、パソコンでもっともポピュラーなインターフェースです。シリアルインターフェースと呼ばれることもあります。AT互換機では通常、COM1 COM2の2ポートが標準で用意されています。

1- .RS-232C 制御のながれ

RS-232C はファイルと同様に扱います。はじめにBAUD命令で転送速度を設定、OPEN命令でオープン、INPUT#/PRINT#命令で入出力、最後にCLOSE命令でクローズします。



【図 1】RS-232C 制御のながれ

注意

F-BASIC では、回線ポートが0から始まります。パソコンの COM1 ポートをプログラムで制御するためには、"COM0:"を使用します。

【表 1】F-BASIC のポート読みかえ

パソコンのポート	デバイス名	ポート番号
COM1	"COM0:"	0
COM2	"COM1:"	1

【プログラム 1】RS-232C をオープンする

例：パソコンの COM1 ポートを転送速度 9600 でオープンします。

```
BAUD 0, 9600  
OPEN "COM0:" AS #1  
...
```

1- モデムを制御する

【図1】RS-232C 制御のながれに従い、実際に機器を接続して制御します。RS-232C を利用して制御する機器のなかでもっとも身近なものにモデムがあります。モデムとは、電話回線を利用して、パソコンのデータを送受信するための機器です。インターネットやパソコン通信へ接続するためにおおく用いられます。

外付けのモデムは RS-232C ポートに接続します。モデム内臓のパソコンでは、通常 COM2 がモデムに割り付けられています。このとき、F-BASIC で注意しなければならない点に、デバイス番号があります（参照【表 1】F-BASIC のポート読みかえ）。

【プログラム 2】を実行するとカーソルが表示されます。モデムへのコマンド”AT”を入力すると、モデムからの応答”OK”が表示されます。

このプログラムはモデムだけでなく、RS-232C を利用するほとんどの機器で利用できます。キーボードから、接続機器に対応したコマンドを入力すると、その結果が、画面に出力されます。キーボードから ESC キーを入力するとプログラムが終了します。

【プログラム 2】モデムの入出力を行うプログラム

```
baud 1,9600  
open "COM1:(S8N2X8)" for output as #2  
open "COM1:(S8N2X8)" for input as #1  
  
locate ,1  
do  
    if not(eof(1)) then print input$(1,1);  
    a$=inkey$  
    if A$=chr$(&H1B) then exit do  
    if A$<>"" then print #2,A$;  
loop  
close  
end
```

【実行結果】

```
AT  
OK
```

注意

1. 下線部はキーボードから文字を入力すること表します。_は Enter キーを入力することを表します。
2. 実行例では接続機器がモデムのため、"AT"と入力します。そのほかの機器ではその機器を制御するコマンド（機器のマニュアル参照）を入力します。

入力したコマンドや結果がバケる、または、エラーが発生する場合は次の点に注意してください。

転送速度：BAUD 命令で変更します。接続している機器と同じ速度にしてください。

通信オプション：パリティやストップビット長などのことで OPEN 命令によって指定します。接続している機器と同じ設定にしてください。設定可能な条件には以下の項目があります。

データビット長、パリティ、ストップビット長、コード系
XON/OFF フロー制御、タイムアウト設定、送受信バッファサイズ
詳細は、F-BASIC のマニュアルを参照してください。

OPEN 命令でエラーが発生する場合は、ほかのプログラムが RS-232C ポートを使っている可能性があります。たとえば、FAX 受信のプログラムなどを起動させている場合にエラーになります。

1- RS-232C ケーブルについて

モデムなどの機器と接続する場合は、ストレートケーブルを使用します。コンピュータ同士を接続する場合は、クロスケーブルを使用します。適切なケーブルを使用しない場合は通信ができません。

1- パソコン通信をする

パソコン通信では、モデムに与えるコマンドや接続先のダイアルは毎回同じものを利用します。そのため次のプログラムのように、パソコン通信ホストへの接続用の処理をプログラムに記述して自動にしてしまいます。これらは、自動巡回（オートパイロット）や定期自動メール送受信のプログラムのプログラムへ応用できます。

【プログラム 3】パソコン通信(NIFTY-Serve)を行うプログラム

```
' 回線オープン
baud 0,9600
open "COM0:(S8N2X8)" for output as #2
open "COM0:(S8N2X8)" for input as #1

locate , ,1
'

' 接続処理
do
    read C$,R$:if C$="" then exit
    print #2,C$
    do
        input #1,A$
        print A$
    loop while instr(A$,R$)=0
loop
'

' コマンド入出力処理
do
    if not(eof(1)) then print input$(1,1);
    a$=inkey$
    if A$=chr$(&H1B) then exit do
    if A$<>"" then print #2,A$;
loop
close
end
'

' 接続用ATコマンドデータ
data "ATZ", "OK"                      ' モデム初期化
data "ATM1", "OK"                      ' スピーカON (OFFは"ATM0" )
data "ATX3", "OK"                      ' ビジー検出
data "ATDT03-5915-7023", "CONNECT"    ' NIFTY-Serve 東京
data "", ""
```

【実行結果】

```
ATZ
OK
ATMO
OK
ATX3
OK
ATDT03-5915-7023
CONNECT 9600
## Welcome to Network World ##
Login: NIFTY
Password: __
Enter Connection-ID --->SVC
Enter User-ID --->XXX01234
Enter Password --->xXxXxXxX
```

注意

- 1.下線部はキーボードから文字を入力すること表します。__は Enter キーを入力することを表します。
- 2.上の接続例で NIFTY-Serve の ID をお持ちでない方は接続のみできますが、適当な ID を入力してもログインできません。
- 3.接続する NIFTY-Serve のアクセスポイントは上の例では東京ですが、お近くのアクセスポイントの電話番号に変更することを勧めします。また、上の電話番号は 2000/11/01 日現在のものです。
- 4.内線電話をお使いの方は、電話番号の先頭に外線発信確立番号(通常0)を挿入する必要があります。

1- AT コマンドについて

ATコマンドはモデムを制御するためのコマンドです。代表的なコマンドには以下のようなものがあります。さらに詳しく知りたい場合は、お使いのモデムのマニュアルを参照してください。

【表 2】代表的な AT コマンド

コマンド	機能	説明
ATZ	初期化	モデムを初期化
ATXn	リザルトコード設定	ビジートーン表示 n=3 でビジー検出
ATMn	スピーカ制御	モデムスピーカの制御 ATM0 = OFF
ATDT	ダイアル	電話番号をダイアル
+++	コマンドモード	回線接続中からコマンドモードへ戻す

1- 計測機器からデータを取得する

RS-232C を利用して制御する例を示します。モデムの場合と大きな違いはありません。機器の初期化コマンド、データ要求コマンドは、接続された機器によって違いがあります。また、BAUD 命令で設定する転送速度、OPEN 命令で指定する通信オプションを必ず、計測機器の設定と同一にしてください。実際に使用した機器は、ADVANTEST DIGITAL MULTIMETER R6552 です。

【プログラム 4】計測機器からデータを取得するプログラム

```
' R6552 の設定: 9600bps,8 ビット長,パリティなし
'           : 1 ストップビット,XON/OFF フロー制御なし,8 ビットコード
'           : トーカンリ・モード
baud 1,9600
DMM=freefile
open "COM1:(S8N1N8)" as #DMM

' *** R6552 の設定 ***
' "Z" 初期化, "F1" 設定=直流電圧測定
' "R5" レンジ=30V, "PR2" サンプリング・レート=MID
print #DMM,"Z,F1,R5,PR2"

' *** 測定 ***
do
    if not(eof(#DMM)) then line input #DMM,D$:print D$
loop
close #DMM
end
```

2.GP-IB

2- GP-IBについて

GP-IBとは General Purpose Interface Bus の略で、国際的な計測用標準インターフェースとして定着しています。GP-IBはRS-232Cにくらべて、高速で、1度に14台までの装置を接続できる、という特徴があります。例えば、デジタルマルチメータ、デジタルオシロスコープといったデジタル計測機器とパソコンを接続するインターフェースとして利用されています。

今回は、ナショナルインスツルメンツ社製ボードとそのプログラムインターフェースで説明しています。そのほかのボードについては、【表 5】F-BASICによる動作確認済みボードを参照してください。

2- GP-IBの準備

GP-IBインターフェースは、RS-232Cインターフェースのように標準でパソコンに用意されていません。GP-IBのインターフェースボードを購入し、パソコンの拡張スロットに差し込む必要があります。

GP-IB ボードはいくつかの種類があり、どのボードを選ぶかを判断する必要があります。ボード選びの重要なポイントにバスのタイプがあります。GP-IB 自体は汎用のインターフェースですが、パソコンのスロットには ISA, PCI, PCMCIA など数種類のバスがあり、GP-IB ボードも使用するパソコンのバスに合うものを選ぶ必要があります。もうひとつ重要なポイントに、対応 OS があります。Windows95 で動作させるには Windows95 対応と書かれたボードを選ぶようにしてください。

GP-IB ボードを入手したら GP-IB ボードのパソコンへの挿入とソフトウェアのインストールを行います。ボードの設定について詳しくは、ボードメーカーのマニュアルを参照してください。

注意

- 1.GP-IB ボードは、バス、対応 OS によって選択してください。
- 2.Windows 対応が明記されていない古いボードについては、ボードメーカーにお問い合わせください。

2- F-BASICによるGP-IB制御の仕組み

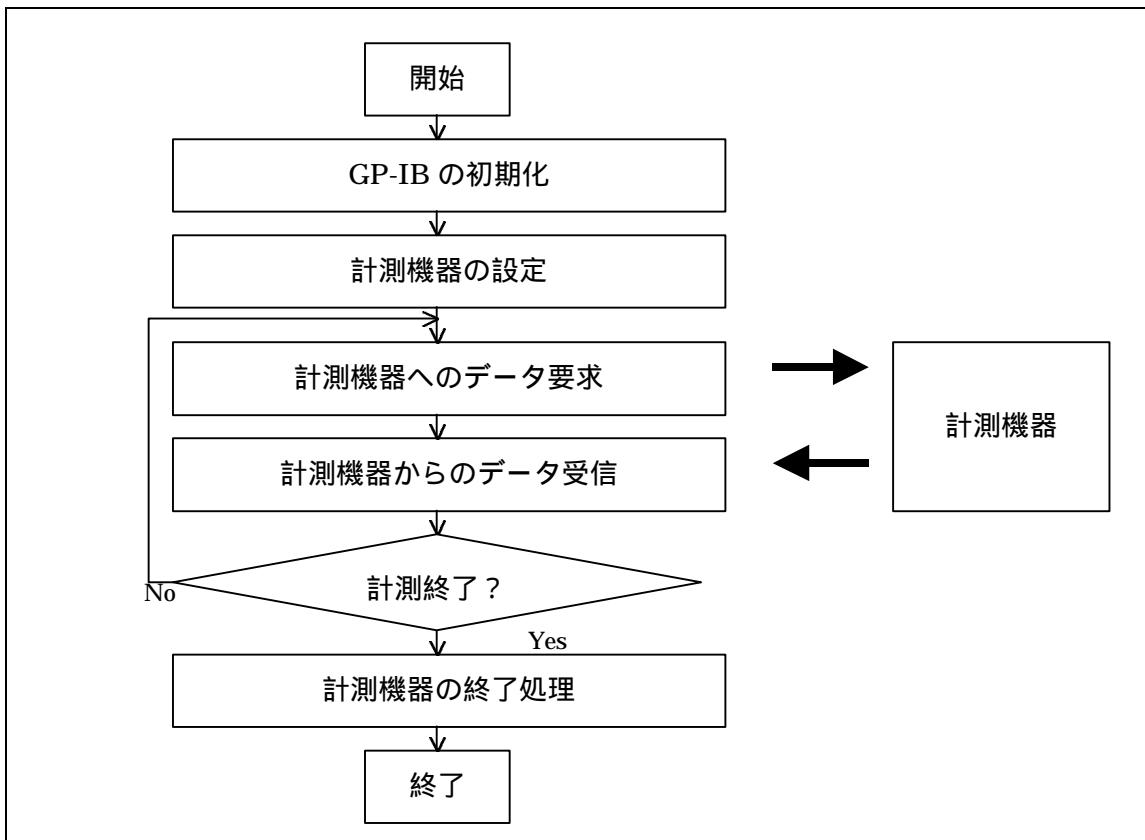
F-BASICではGP-IBボードに対する標準的なプログラムインターフェースを用意していません。しかし、ダイナミックリンクライブラリ(DLL)を呼び出す基本命令が用意されています。この命令と、ボードメーカーが提供する GP-IB ボード制御用ダイナミックリンクライブラリを使って、GP-IB を制御することができます。

一般的に、DLL ごとに用意されているアプリケーションインターフェースの名称やパラメタは、インクルードファイルの形で提供されます。以下の例でも、GP-IB 用の関数はすべてインクルードファイルで提供されていることが前提になっています。プログラムの先頭に、#include 命令があるのはこのためです。

インクルードファイルはサンプルプログラムとともに入手することができます(参照「プログラムの入手方法」)。

2-. GP-IB 制御のながれ

GP-IB を制御するには、【図 2】のように GP-IB インタフェースへ初期化をおこない、計測機器へのコマンド送信、計測機器からのデータ受信を行います。



【図 2】GP-IB 制御の流れ

2-. デジタルマルチメータを制御する

実際にデジタルマルチメータを制御する例を示します。【プログラム 5】は、ナショナルインスツルメンツ社製 GP-IB ボード用の NI-488.2Routines を使用して作られています。実際に使用した機器は、ADVANTEST DIGITAL MULTIMETER TR6845 です。

プログラムの流れは、SENDIFC 関数で IFC(インターフェースクリア)を設定して GP-IB を初期化、SEND 関数でデジタルマルチメータを初期化、RECEIVE 関数でデータを受信します。

この例の場合は、データ要求のためのコマンド送出しませんが、通常、SEND 関数で計測機器へデータ要求コマンドを送出したあとに、RECEIVE 関数でデータを受信します。

各関数の実行結果は THREADIBSTA 関数を使って戻り値を検査します。

この例では、電圧を測定しただけで終了していますが、測定を繰り返して、グラフ化するといったことも可能です。



【写真 1】デジタルマルチメータ

【プログラム 5】デジタルマルチメータを制御するプログラム

```
#include "gpib_def.bi"
#include "gpib.bi"

' デジタルマルチメータを設定する
GP_BOARDID& = 0
GP%=22           ' デジタルマルチメータのアドレス
SENDIFC GP_BOARDID&      ' GP-IBのリセット
DMY&=IBTMO(GP_BOARDID&,12&) ' タイムアウト設定
ADDRLIST_ANY&=NOADDR
ENABLEREMOTE GP_BOARDID&, ADDRLIST_ANY& '全ての機器のリモート設定
if (THREADIBSTA and &H8000)<>0 then
    print "装置をリモート化できませんでした。"
end if

' デジタルマルチメータの初期化
BUF$="S1,DL0,PH1,M0,F1,R0,PR2"+chr$(13,10)
SEND GP_BOARDID&, GP%, BUF$, Len(BUF$), DABEND
if (THREADIBSTA and &H8000)<>0 then
    print "装置を初期化できませんでした。"
end if

' 結果(電圧)を読み取る
MG$ = space$(20)
RECEIVE GP_BOARDID&, GP%, MG$, Len(MG$), &HOA
if (THREADIBSTA and &H8000)<>0 then
    print "装置からデータを読み取ることができませんでした。"
end if
print MG$

' 機器の終了処理
' (GPIBへuntalk,unlisten,goto localを設定する)
SENCMDS GP_BOARD&, chr$(&H3F,&H20+GP%,&H01,&H3F,&H5F), 5&
if (THREADIBSTA and &H8000)<>0 then
    print "装置を設定できませんでした。"
end if
stop
end
```

【実行結果】

```
DV 5.334E+0
```

1. 直流電圧を測定した結果です。デジタルマルチメータに接続された環境によって、表示される値が違います。
2. 表示されたデータは文字型です。数値形式に変更するには VAL 関数を使用してください。
例: print val(mid\$(MG\$,4))

注意 :

1. ボードメーカーによって関数名称やパラメタ形式に違いがありますが、制御のながれはほぼ同じです。
2. デリミタ設定が違うとデータの送受信ができません。CR,CRLF,EOI を計測機器やデータの形式に合わせて正しく選択してください。
3. リモート設定 / 解除が正しく行われないと、計測機器を操作できなくなります。

2- アドレスについて

GP-IBにおいては、データ通信する機器を識別するため、機器毎に固有のアドレスを設定します。このアドレスと、プログラム上の関数にパラメタとして引き渡す相手アドレスが一致しない場合、データの送受信ができません。機器のアドレスの設定は、機器のパネルなどを操作して設定します。詳しくは、機器のマニュアルを参照してください。

また、ボードによっては、自アドレスをプログラムの関数によって設定するタイプのものがあります。

【表 3】【プログラム 5】で使用した GP-IB 制御関数

関数	機能
SENDIFC	IFC を設定して GP-IB を初期化する
IBTMO	タイムアウトを設定する。
ENABLEREMOTE	接続機器をリモート設定する。
SEND	アドレス指定をしてから機器へデータを送る。
RECEIVE	アドレス指定をしてから機器のデータを受信する。
SENCMDS	GP-IB コマンドを送出する。
THREADIBSTA	各関数の実行結果を返す。

2- FFT アナライザを制御する

デジタルマルチメータ以外にもさまざまな計測機器があります。FFT アナライザからデータを取得する方法について説明します。

FFT アナライザのような計測機器では、1 度に受け取るデータが可変長で大規模になります。また、データ中に浮動小数点形式のバイナリ値が含まれます。そのようなデータを BASIC の配列変

数で受け取る処理が【プログラム 6】になります。

この例では、データ受信時に、まず、データ長さを調べてから、その分のデータを受信します。
実際に使用した機器は、Hewlett Packard DYNAMIC SIGNAL ANALYZER – 35670A です。



写真 2 FFT アナライザ

【プログラム 6】FFT アナライザ用データ受信ルーチン

```

...
' データフォーマット設定とデータ要求コマンドを送出する。
CMD$="form:data real,64"+chr$(13,10) :SEND GP_BOARDID&, GP%, CMD$, len(CMD$), DABEND
CMD$="CALC1:FEED"+chr$(13,10) :SEND GP_BOARDID&, GP%, CMD$, len(CMD$), DABEND
CMD$="TRAC:DATA D1,TRAC1"+chr$(13,10):SEND GP_BOARDID&, GP%, CMD$, len(CMD$), DABEND
CMD$="TRAC:DATA? D1"+chr$(13,10) :SEND GP_BOARDID&, GP%, CMD$, len(CMD$), DABEND
' データ受信を開始する。
RECEIVESETUP GP_BOARDID&, GP%
' データを受信し、つづく「データ長」を獲得する。
MG$=space$(2)
RCVRESPMSG GP_BOARDID&, MG$, len(MG$), STOPEND
L=val(right$(MG$,1)):MG$=space$(L)
RCVRESPMSG GP_BOARDID&, MG$, L, STOPEND
NUM_OF_BYTES&=val(MG$)
' データを「データ長」分 受信する。
MGD$=space$(NUM_OF_BYTE&)
RCVRESPMSG GP_BOARDID&, MGD$, NUM_OF_BYTES&, STOPEND
MG$=space$(10)
RCVRESPMSG GP_BOARDID&, MG$, len(MG$), &HOA
' 浮動小数点前進法の受信データを後退法に変換して配列へ代入する。
KK&=0:NUM_OF_DATA& = NUM_OF_BYTES& ¥ 8
for I=0 to NUM_OF_DATA&-1
    A(I,LY)=cvd(mid$(MGD$,KK&,8))*YSCT: KK&=KK&+8
next
...

```

注意

1. データ取得部分のみの処理でこれだけでは動作しません。
2. FFT アナライザによっては、データの浮動小数点形式が後退法のものか、両者を選択できるものがあります。その場合、GP-IB 関数のデータ受信関数の受信データアドレスパラメタに倍精度配列変数の先頭要素を引き渡します。前進法 -> 後退法の変換の必要はありません。

【表 4】【プログラム 6】で使用した GP-IB 制御関数

関数	機能
RECEIVESETUP	RCVRESPMSG 関数の準備（相手アドレス）をする。
RCVRESPMSG	アドレス指定された機器からデータを受信する。
SEND	アドレス指定された接続機器へ制御コマンドを送る。

2- .F-BASIC による動作確認済みボード

F-BASIC による動作確認済み GP-IB ボードは以下のようないわがあります。各ボード用のサンプルプログラムがホームページに用意されています（参照「プログラムの入手方法」）。

【表 5】F-BASIC による動作確認済みボード

GP-IB ボードメーカー	ボード名
日本ナショナルインスツルメンツ株式会社	AT-GPIB/TNT, PCMCIA-GPIB NI-488.2M Ver. 1.20
ラトックシステム株式会社	REX-5052 IEEE 488.1 PC CARD
株式会社エヌエフ回路設計プロック	KEITHLEY 製 KPC-488.2AT
株式会社インターフェース	PCI-4301

3. そのほか

3- .EXCEL へデータを移行する

Microsoft Excel へデータを移行するにはデータを CSV 形式のファイルに保存します。ファイルは Excel で読み込みグラフにします。

【プログラム 6】EXCEL で読み込み可能なCSV ファイルに保存するプログラム

```
do
    do
        WAIT 100
        loop while right$(time$,2)<>"00"
        RECEIVE GP_BOARDID&,GP%,A$,len(A$),&HOA
        DV$=mid$(A$,4,10)
        GOSUB *WRITE_DATA
    loop
    stop
    end
    ...
*WRITE_DATA
open "data.csv" for append as #1
write #1,time$,DV$
close #1
return
```

3- そのほかの言語とくらべて

RS-232C や GP-IB を制御する場合、BASIC で記述した場合と C 言語で記述した場合で比較すると、記述するコードの少なさに特徴があります。

機器を制御しながら実験や測定にあわせたデータの取得や表示に変更するには最適な言語といえます。

3- RS-232C/GP-IB 以外のインターフェースについて

RS-232C/GP-IB 以外のインターフェースがありますが、GP-IB のインターフェース同様、多くの場合、アプリケーションソフトウェアとのインターフェースはダイナミックリンクライブラリ(DLL)によって提供されています。特殊な I/O ボードにおいても、DLL が提供されている場合は F-BASIC によって制御することができます。

- 以上 -