

ご自分でプログラミングされる方のための開発ツール!

GP-IB

MS-Excelと
測定器が直結!



Windows 2000/Xp/Vista

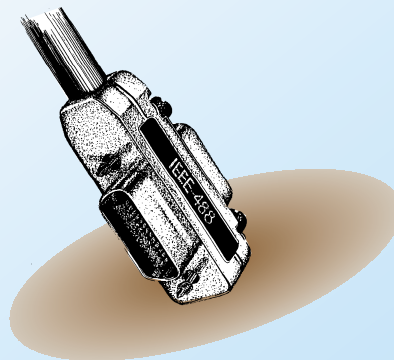
GP-IB (WIN) ライブラリ

「WindowsでGP-IBを簡単に!」を貫き、一切のGP-IBの機能を省略することなく、使いやすさと実用性を実現しました。

ちょっとした測定でしたらMS-Excelだけ用意いただければ、それで十分です。VBAマクロからどんな測定器も自由自在にコントロールでき、Excelシートへのデータの取得も可能です。もちろん、VisualBASICやC++等を使って本格的な自動計測プログラミングにも十二分の性能を発揮します。Win95/98では、デジタルI/Oボード・A/Dボード等の入出力関数が使用できます。

- ・ ActiveX コントロール (ocxカスタムコントロール)
- ・ ActiveX サーバ (OLEサーバ)
- ・ ダイナミックリンクライブラリ

上記3種類の形式をサポートしておりますから、使用される用途とプログラミングの習得レベルにあった形式を自由に選択いただけます。どの形式で使用される場合でも関数名は全て統一されています。



特徴

- 1 MS-Excelが自動計測ソフトに早変わり!**

もう VisualBASIC や C++ 等は必要ありません。MS-Excel 等に付属の VBA マクロから GP-IB の全機能が使用できますから、測定器のコントロールはもちろん、取得したデータを Excel 上で演算、帳票作成、グラフ化が自由自在です。
- 2 85個の豊富な関数群は測定器を選びません。**

あらゆる測定器に対応した GP-IB 入出力関数を用意しております。また、GP-IB 関数のエラー発生履歴を過去 50 回まで記憶していますからデバッグも容易です。自動計測では必須の 50 個の独立したタイマ関数も付属しております。
注) A/D ボードやデジタル I/O ボード用の入出力関数は、Win95/98 だけでご使用になれます。
- 3 様々なプログラム開発環境に対応しました。**

「ActiveX コントロール」「ActiveX サーバ」「ダイナミックリンクライブラリ」の 3 つの形式をサポートしておりますから、プログラム開発環境に合わせて使いやすい形式を選択できます。関数名はすべて統一されていますから言語間の移植は簡単です。
特に、Windows プログラミングで標準となった「ActiveX コントロール」は、GP-IB の使い勝手を飛躍的に向上させました。VisualBASIC, VBA, Delphi, C++ Builder 等のビジュアルなプログラム開発環境で最大の威力を発揮します。
- 4 使用例を取り入れた詳細なマニュアルが付属しております。**

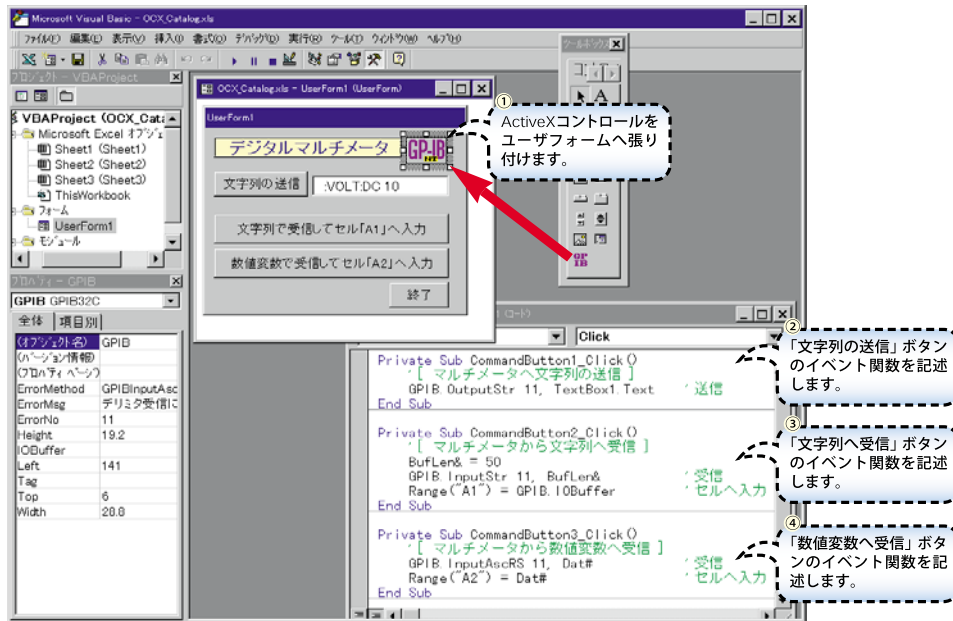
約 200 ページにおよぶマニュアルには、「GP-IB ボードのセットアップ」「具体的な使用例」「詳細な関数リファレンス」「うまくいかない場合の対処方法」「他言語からの移植方法」等が詳細に記載されております。
- 5 PC-98, DOS/V のどちらでも動作します。**

本ライブラリがインストールされているパソコンでしたら、PC-98, DOS/V のどちらでも動作します。
- 6 GP-IB PC カードによりノートパソコンで使用可能です。**

GP-IB PCMCIA カードを使用すれば、ノートパソコンによるポータブルな自動計測システムの構成が可能となります。

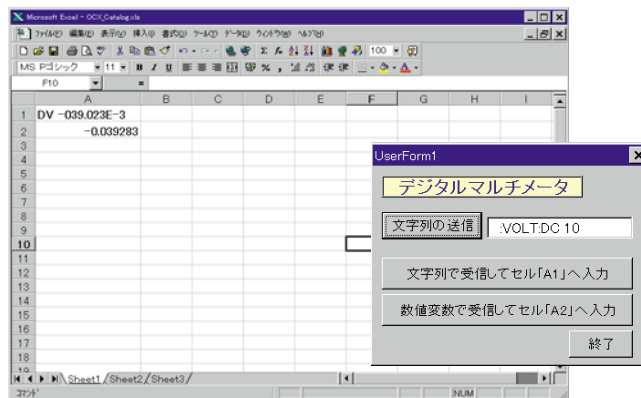
「MS-ExcelのVBA」でActiveXコントロールを使用する

Excel97付属の「Visual BASIC Editor」を起動し、ユーザフォームをデザインした後、下記、①②③④の手順でプログラムは完成です。

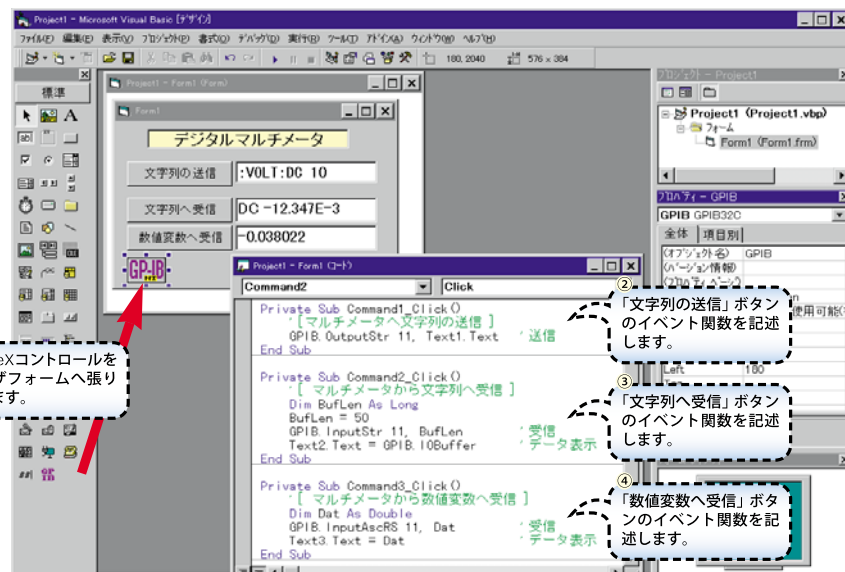


実行結果

MS-Excelのメニュー「ツール」を選択し、上記で作成したマクロを実行すると、右図の画面が表示されます。各ボタンを左クリックすると、マルチメータの測定値がExcelシートに入力されます。



「Visual BASIC 5.0/6.0」でActiveXコントロールを使用する



「Delphi 3.0/4.0」で ActiveX コントロールを使用する

① ActiveXコントロールをユーザフォームへ張り付けます。

② 「文字列の送信」ボタンのイベントハンドラを記述します。

③ 「文字列へ受信」ボタンのイベントハンドラを記述します。

④ 「数値変数へ受信」ボタンのイベントハンドラを記述します。

```

procedure TSDIAppForm.Button1Click(Sender: TObject);
begin
  // マルチメータへ文字列を送信
  GPIB.OutputStr(11, Edit1.Text);
end;

// マルチメータからのデータを文字列として受信
procedure TSDIAppForm.Button2Click(Sender: TObject);
var
  BufLen : integer;
begin
  BufLen := 50;
  GPIB.InputStr(11, BufLen);           (受信)
  Edit2.Text := GPIB.IOBuffer;        (データ表示)
end;

// マルチメータからのデータを数値変数へ受信
procedure TSDIAppForm.Button3Click(Sender: TObject);
var
  Dat : double;
begin
  GPIB.InputAscRS(11, Dat);           (受信)
  Edit3.Text := FloatToStr(Dat);     (データ表示)
end;
  
```

「C++ Builder 1.0/3.0」で ActiveX コントロールを使用する

① ActiveXコントロールをユーザフォームへ張り付けます。

② 「文字列の送信」ボタンのイベントハンドラを記述します。

③ 「文字列へ受信」ボタンのイベントハンドラを記述します。

④ 「数値変数へ受信」ボタンのイベントハンドラを記述します。

```

void __fastcall TSDIAppForm::Button1Click(TObject *Sender)
{
  // マルチメータへ文字列を送信
  GPIB->OutputStr(11, Edit1->Text);
}

void __fastcall TSDIAppForm::Button2Click(TObject *Sender)
{
  // マルチメータからのデータを文字列として受信
  long BufLen = 50;
  GPIB->InputStr(11, (long #)BufLen); // 受信
  Edit2->Text=GPIB->IOBuffer;        // データ表示
}

void __fastcall TSDIAppForm::Button3Click(TObject *Sender)
{
  // マルチメータからのデータを数値変数へ受信
  double Dat;
  char Buf[50];
  GPIB->InputAscRS(11, (double #)Dat); // 受信
  sprintf(Buf, "%f", Dat);          // データ表示
  Edit3->Text = Buf;
}
  
```

「Visual C++」でダイナミックリンク・ライブラリを使用する

マルチメータのファンクションやレンジの切り換えと、測定値の取り込み例

DCV 測定ファンクション
300mVレンジ
HOLDモードへ切り換え

積分時間をMID
SRQをOFFへ切り換え

トリガを送信

測定値を受信

測定値を表示

フリーラン
オートレンジへ切り換え

リモートを解除

```

// 【アドバンテスト R6551】
#include "Gpib32c.h" // GPIBのダイナミックリンクのヘッダファイル
void CMainFrame::OnDmmTest()
{
  long R6551=11; // マルチメータ GPIBアドレス
  long Er;
  DWORD BufLen;
  char Buf[100];

  Er = GPIBDeviceClear( R6551 ); // セレクト・デバイス・クリア コマンド送信
  Er = GPIBRemote( R6551 ); // R6551をリモート状態へ移行
  Er = GPIBOutputStr( R6551, "F1,R3,M1" ); // DC-Volt,300mV-Range,Hold-Mode設定
  Er = GPIBOutputStr( R6551, "PR2,S1" ); // MID SamplingRate,SRQ Off設定
  Er = GPIBTrigger( R6551 ); // R6551をトリガ
  BufLen = 50;
  Er = GPIBInputStr( R6551, &BufLen, Buf ); // 測定値を Buf へ読み込む
  {
    if ( Er<100 )
    {
      MessageBox( Buf, "R6551 受信データ" );
    }
  }
  Er = GPIBOutputStr( R6551, "MO,R0" ); // Free-Run,Auto-Range設定
  Er = GPIBGotoLocal( R6551 ); // R6551をローカルへ移行
}
  
```

GP-IB 関数 / メソッド一覧表

入力関連関数

GPiBQueryStr	クエリコマンド送信後、応答を数値データとして受信します。(旧名: Input482Z)
QueryNum	クエリコマンド送信後、応答を文字列データとして受信します。(旧名: Input488Z)
InputAscHdr	データブロック受信時、'#'に続く ASCII ヘッダだけを数値化して受信します。
InputAscLM	ASCII 型データを Long 型配列に受信します。
InputAscLS	ASCII 型データを Long 型数値変数に受信します。
InputAscRM	ASCII 型データを Double 型配列に受信します。
InputAscRS	ASCII 型データを Double 型数値変数に受信します。
InputStr	文字変数にデータを受信します。
LnumberBld	ASCII 文字列を指定された文字で分割し、Long 型配列へ入力します。
RnumberBld	ASCII 文字列を指定された文字で分割し、Double 型配列へ入力します。
StrSeparate	ASCII 文字列を指定された文字で分割し、文字列配列へ入力します。
InputBin	1 から 4 バイト構成のバイナリ型データを Long 型変数に受信します。
InputBinB	バイト型バイナリデータを Integer 型配列に受信します。
InputBinD	IEEE64 ビット型バイナリデータを Double 型配列に受信します。
InputBinS	IEEE32 ビット型バイナリデータを Single/float 型配列に受信します。
InputBinW	ワード型バイナリデータを Integer 型配列に受信します。
InputToFile	指定したファイルヘッダを受信します。
RByte	1 バイトを受信します。

出力関連関数

GPiBOutputAscLM	Long 型配列データを ASCII 型で送信します。
OutputAscLS	Long 型数値を ASCII 型で送信します。
OutputAscRM	Double 型配列データを ASCII 型で送信します。
OutputAscRS	Double 型数値を ASCII 型で送信します。
OutputBin	1 から 4 バイト構成の数値データをバイナリ型で送信します。
OutputBinB	Integer 型配列データをバイト型バイナリデータとして送信します。
OutputBinC	文字列変数データをバイナリデータとして送信します。
OutputBinD	Double 型配列データを IEEE64 ビット型バイナリデータとして送信します。
OutputBinS	Single/float 型配列データを IEEE32 ビット型バイナリデータとして送信します。
OutputBinW	Integer 型配列データをワード型バイナリデータとして送信します。
OutputFromFile	指定したファイルからデータを送信します。
OutputStr	文字列データを送信します。
Wbyte	1 バイトを送信します。

各種コマンド関連関数

GPiBDeviceClear	SDC または DC コマンドを送信します。
GotoLocal	GTL コマンドを送信します。または、REN ラインをインアクティブにします。
LocalLockout	LLO コマンドを送信します。
Remote	デバイスをリモート状態にします。または、REN ラインをアクティブにします。
SendCmd	各種コマンドを送信します。
Trigger	GET コマンドを送信します。
Unlisten	アンリッスンを送信します。
Untalk	アントークを送信します。
ListenCmdCreate	パソコンがリスナになるためのコマンド文字列を作成します。
TalkCmdCreate	パソコンがトーカーになるためのコマンド文字列を作成します。

タイマ関数

GPiBTimerRead	タイマ番号を指定してタイマスタートからの経過時間を読み出します。
TimerStart	タイマ番号を指定してタイマをスタートします。
WaitSec	指定した時間まで待ちます。

SRQ / ポーリング関連関数

GPiBPPoll	パラレルポーリングを実行します。
PPollConfig	パラレルポーリングの応答出力ラインを割当てます。
PPollResponse	スリープモードの時、パラレルポーリング応答モードを設定します。
PPollUnconfig	デバイスがパラレルポーリングに反応しないように指示します。
SendSRQ	スリープモードの時、SRQ を送信します。
Spoll	シリアルポーリングを実行します。
GetSRQ	SRQ の発生を調べます。
GetSRStatus	スリープ時、シリアルポーリングの受信を調べます。

情報取得関数

GPiBGet7210Reg	uPD7210 のレジスタ内容を調べます。
GetBussLine	GP-IB バスラインの状態を調べます。GP-IB (98) F、GP-IB (PC) F に限定
GetBussRequest	スリープモードの時、マスタからの要求を調べます。
GetConfig	GP-IB ボードの環境を調べます。
GetDelim	デリミタの設定値を調べます。
GetErrMsg	エラー / ワーニングの発生とその内容を調べます。
GetErrOccurred	過去のエラー発生の有無を調べます。
GetMasterMode	GP-IB のマスタ / スレーブを調べます。
GetMyAddress	GP-IB のマイアドレスを調べます。
GetOpenCount	GP-IB のオープン状態を調べます。
GetTimeout	タイムアウト設定時間を調べます。
GetTransMode	現在の転送モード (FIFO ON/OFF) を調べます。

その他の関数

GPiBAbort	GP-IB をリセットします。
Close	GP-IB をクローズします。
ClrBussRequest	スリープ時、マスタからのトーカー / リスナ指定を解除します。
ErrRecordClear	エラー発生履歴をクリアします。
Exit	パソコンの GP-IB を強制終了します。
MasterOpen	GP-IB をマスタモードでオープンします。
PassControl	指定したデバイスにアクティブコントロールの機能を渡します。
SetDelim	デリミタを設定します。
SetLSBFirst	バイナリ転送時、最初の送受信データを LSB か MSB に設定します。
SetDelay	出力実行後のデレイ時間を設定します。
SetTimeout	タイムアウト時間を設定します。
SetTransMode	転送モード (FIFO ON/OFF) を設定します。
SlaveOpen	GP-IB をスレーブモードでオープンします。
TransAbort	FIFO ON 時の転送を強制中断します。
WaitActive	指定した GP-IB のラインが有効になるのを待ちます。GP-IB (98) F、GP-IB (PC) F に限定

タイマ関数

GPiBTimerRead	タイマ番号を指定してタイマスタートからの経過時間を読み出します。
TimerStart	タイマ番号を指定してタイマをスタートします。
WaitSec	指定した時間まで待ちます。

パラレル I/O 入出力関数 (Windows NT/2000/XP/Vista では使用できません。)

GPiBInputPioB	A/D ボードやパラレル I/O ボードから 1 バイトを入力します。
InputPioW	A/D ボードやパラレル I/O ボードから 2 バイトを入力します。
InputPioDW	A/D ボードやパラレル I/O ボードから 4 バイトを入力します。
OutputPioB	A/D ボードやパラレル I/O ボードから 1 バイトを出力します。
OutputPioW	A/D ボードやパラレル I/O ボードから 2 バイトを出力します。
OutputPioDW	A/D ボードやパラレル I/O ボードから 4 バイトを出力します。

GP-IB 動作確認プログラム

本ライブラリには「GP-IB 動作試験プログラム」が含まれております。セットアップ完了後、簡単に GP-IB ホードの動作確認を行うことができます。また、その場で様々な測定器との通信テストを行うことも可能です。

測定器から ASCII 型データの受信試験を行います。

測定器にコマンドを送信し、測定器の動作を確認します。

バスライン・コマンド等の送信試験を行います。

マルチメータ等を接続し、測定値を Excel シートへ受信します。

GP-IB (WIN) ライブラリ価格表

■ 価格 (シングルライセンス)

コンテック用販売終了

製品名	対応GP-IBボード	型番	価格
GP-IB ライブラリ	コンテック社	GPW-32-C	75,000円
	NI社	GPW-32-N	75,000円

■ ライセンス商品価格

コンテック社用販売終了

ライセンス名	制限数	制限時間	型番	価格
サイトライセンス商品 同一事業所内で複数の検査装置や試験装置に組み込まれる場合は本ライセンス商品の購入をご検討ください	10個まで	無期限	GPW-32-C-S10 GPW-32-N-S10	190,000円
	30個まで	無期限	GPW-32-C-S30 GPW-32-N-S30	385,000円
	無制限	無期限	GPW-32-C-SU GPW-32-N-SU	550,000円
組込販売ライセンス商品 アプリケーションに組込販売	10個まで	1年間	GPW-32-C-E10 GPW-32-N-E10	190,000円
	無制限	1年間	GPW-32-C-EU GPW-32-N-EU	360,000円



■ 動作環境

動作パソコン/OS	動作するGP-IBボード	提供形式	使用できる言語環境
Windows NT 4.0 Windows 2000 Windows Xp Windows Vista	別紙 「GP-IBボード一覧」 参照	ActiveX コントロール (OSCX カスタムコントロール) ActiveX サーバ (OLE サーバ) ダイナミックリンク・ライブラリ	Visual BASIC 4.0/5.0/6.0~ VBA マクロ Office95/97~ Visual C++ 4.0~ Delphi 3.0/4.0~ C++Builder 1.0/3.0~ 等

GP-IB (WIN) ライブラリ