ユーザーズマニュアル

USB-RS232C I/O Controller USBPIO-24





株式会社 インターネット

Copyright©2008~2010 Internet Co., Ltd. All Rights Reserved

ユーザーズマニュアル履歴

Rev.	改訂日付	内容
1.0	2008/12/26	初版リリース
1.01	2009/2/11	字句修正
1.02	2009/11/23	コネクターCN-1 の電源ピン間違い修正
1.03	2009/12/ 1	複数デバイスの接続について 追加 mpusbapi.dll 関数一覧 追加
1.10	2010/ 7/17	Windows7 対応、コネクタ品番追加
1.11	2010/ 7/30	MPUSBWrite, MPUSBRead 引数の間違い
1.12	2010/ 9/ 2	CPU のスペック記述間違い訂正

☆本マニュアルの最新版は当社ホームページからダウンロードいただけます。

1.	はじめに5
2.	概要
	2.1 特長 10
	2.2 ボード仕様 11
3.	ハードウェア
	3.1 ブロック図 12
	3.2 外観・接続図 13
	3.3 各部名称と機能 15
	3.4 コネクター・ピン配置 17
	3.5 電気的特性19
	3.6 電源について20
4.	ソフトウェア
	4.1 ドライバーのインストール 21
	4.2 コマンド一覧 33
	4.3 サンプルプログラムの実行45
	4.4 サンプルプログラム解説54
	4.5 mpusbapi.dll の関数一覧63
	4.6 PIC18F4553 開発環境66

5. その他

5.1	オプションボード	67
5.2	特注仕様	68

1.はじめに

このたびはサイバーメロン・マイコンボードシリーズ USBPIO-24 をお買い あげいただきまして、誠に有り難うございます。

USBPIO-24 は USB または RS232C からパラレル I/O を制御したり A/D コンバ ータから読み込んだデータを USB または RS232C 経由で取り込むことの できるマイコンボードです。

ご使用に当たりましては本書を良くお読みいただき正しい取り扱い方法を ご理解の上ご使用いただきますようお願い致します。



- 1. 本製品の仕様、および本書の内容に関して事前の予告なく変更する ことがありますのでご了承ください。
- 本製品または付属プログラムの使用によるお客様の損害、および第三者からのいかなる請求につきましても当社はその責任を負いかねますので予めご 了承ください。
- 本製品に付属のソフトウェア・ライブラリおよびサンプルプログラムは その動作を完全保証するものではありません。製品に組み込んで使用される 場合にはユーザ様にて十分なテストと検証をお願いします。 ソフトウェアの最新版はユーザー登録後、当社ホームページからダウン ロードしていただけます。
- 4. 本製品および本書に関し、営利目的での複製、引用、配布は禁止されています。





- 2. ご使用になる前に下記の安全についての注意を必ずお読みください。
- 3. 通電する前に、本製品の使い方を十分ご確認いただき、正しい接続と 設定をご確認ください。



- 本製品を医療機器など人命に関わる装置や高度な信頼性・安全性を要求される装置へ搭載することはご遠慮ください。
 その他の装置に搭載する場合でもユーザー様にて十分な信頼性試験・評価をおこなった上で搭載してください。
 また非常停止や緊急時の制御は外部の独立した回路にておこなってください。
- 2. 本製品の改造使用は発熱、火災などの原因となり危険ですのでご遠慮ください。
- 3. 本製品のマニュアル記載環境以外でのご使用は故障、動作不良などの原因に なりますのでご遠慮ください。
- 本製品は導電部分が露出しておりますので、金属パーツなどショートの可能性のあるもの、液体のこぼれる可能性のある場所の近くでの使用はお控えください。

また装置に組み込む場合も絶縁に関しては十分な注意を払ってください。 5. 電源は必ず本製品専用(指定)のものをお使いください。 電圧、極性、プラグ形状など異なるものをご使用になりますと故障の原因と

- なるばかりでなく、火災など重大事故に繋がる危険性があります。
- 6. 本製品に触れる前には体から静電気を除去してください。
- 7. 本製品には落下など強い衝撃を与えないでください。

使用環境



- 以下の環境でのご使用はお控えください。
 - ・強い電磁界や静電気などのある環境
 - ・直射日光の当たる場所、高温になる場所
 - ・氷結や結露のある場所、湿度の異常に高い場所
 - ・薬品や油、塩分などのかかる場所
 - ・可燃性の気体、液体などに触れる場所
 - ・振動の多い場所、本製品が静止できない場所
 - ・基板のショートを引き起こす可能性のある場所

規格取得など



■ 本製品は UL CSA 規格、CCC 認証などいっさい取得しておりません。 装置に組み込む場合は各安全規格への適合性をユーザー様で ご確認いただき、対応して頂きますようお願いします。

■ また本製品は RoHS(特定有害物質の使用制限指令)に対応しておりません。

製品保証と修理



- ・本製品の保証は商品到着後10日以内の初期不良のみ無償交換 とさせていただきます。
 本マニュアルに記載するテスト手順にて正しく動作しない場合は ただちに電源を切って、当社ホームページのサポートからご連絡 ください。 折り返し交換手順をご案内いたします。
- ・保証期間中であってもユーザー様の責となる故障(落下や電源の 誤接続など)は有料修理になります。
- ・その他の故障やクレームにつきましても当社ホームページ
- <u>http://www.cyber-melon.com</u> お問い合わせコーナーよりご連絡ください。



梱包内容をご確認ください



※ RS-232C ケーブルは付属しません。 必要な場合は9ピン ストレート オス・メスタイプをお買い求めください。



動作チェック

・本ボードの動作および動作チェックには以下の環境が必要です。

Windows パソコン OS: WindowsXP 以上 USB の空きポート

- ・付属の5V電源アダプターをUSBPI0-24 本体に接続して通電してください。
 ボード上の赤い LED が点灯して電源が入ったことを示します。
 電源は必ず付属の5V DC 電源をお使いください。
- ・ USB ケーブルをパソコンと接続してください。
 . 4章-3の手順にしたがって USB ドライバーをインストールし、接続テストをおこなってください。

もし動作異常が認められた場合は電源をはずして当社ホームページ <u>http://www.cyber-melon.com</u>のサポートから症状をご連絡ください。 対処方法をメールまたは電話でご連絡いたします。

2. 概要

USBPIO-24 は Microchip 社の PIC マイコン PIC18F4553 を用いた USB または RS232C で I/O を制御する組み込みボードです。 ファームウェア書き込み済みで、パソコンなどと USB または RS232C で接続して

簡単なコマンドを送ることで下記の機能を使うことができます。

2.1 特長

USBPIO-24 には3つの8ビット I/O ポート(ポート A, B, C 合計 24ビット) があります。*1)

USBPI0-24 は以下の3つの機能を有しています。

1. PIO モード

USB/RS232C で 24bit のパラレル I/O を制御(入出力)できます。 1bit ごとに IN/OUT の方向を設定可能です。

2. ADC モード

12bit A/D コンバータ(最大8チャンネル) からの入力を USB/RS232C で 取り込みます。A/D コンバータ入力ピンはポートA と共通です。*2) データロガーとして利用可能です。

3. STN モード

USB/RS232C で STN 4行x20文字キャラクター液晶表示器 *3)
 に文字を出力します。
 簡易キャラクター端末として使える専用コネクター搭載。

- *1) A, B, C の名前は実際の PIC18F4553 のポート名(RA, RB, RC..) とは直接関係ありません。
 (一部共通です)
- *2) ADC モード、STN モードでもポート B, C は PIO モードで使うことができます。
- *3) STN 液晶表示器は付属しておりません。ご利用の際は別途ご購入ください。
 推奨品は SC2004C (秋月電子通商様扱い) です。
 尚、USBPI0-24 では MIL フラットケーブルコネクタを液晶表示器の裏側から取り付ける事を
 想定して、コネクターの奇数・偶数列を入れ替えていますので、くれぐれもご注意ください。
 液晶表示器の中には電源と GND が逆(1:Vdd, 2:Vss)のものがありますので特にご注意ください。
 液晶表示器側の表示で1が Vss(GND) のものをお使いいただく必要がございます。

2.2 ボード仕様

項目	仕 様			
CPU	PIC18F4553-I/P(または PIC18LF4553-I/P)			
	Flash ROM 容量 32KB,内蔵 RAM 容量 2KB			
動作クロック	20MHz			
電源電圧	DC5V 入力(CPU 周辺動作も5V)			
入出力	RS232C コネクター			
	USB コネクタ(標準B タイプ)			
	ICSP (インサーキット・シリアルプログラミング) 用ヘッダー			
	DIP スイッチ (3極)			
	リセットスイッチ			
	汎用 I/0 コネクター2(20 ピン)			
	汎用 I/0 コネクター1(16 ピン) 含アナログ系			
	STN キャラクター表示器用コネクター(14 ピン)			
基板寸法	99.4(W) x 73.6(D)x 14.8(H) mm (足なし基板のみ)			
消費電流	50mA (外部負荷なし)			

■ 出力用フラットケーブルコネクターの品番

ヒロセ HIF3BA-xxD-2.54R (xx はコンタクトピン数)

を推奨しておりますが、キー(誤挿入防止の突起)の位置が中央一カ所 で足が 2.54mm ピッチ(線間 1.27mm)であれば他のメーカーの相当品でも 問題ないと思います。

■ 取り付け穴の位置について
 取り付け穴位置: 基板4隅からX,Y各5.6mm
 USBコネクタのセンター位置: 基板端からY19.7mm

3. ハードウェア

3.1 ブロック図

USBPIO-24 ボード・ブロックダイアグラム



3.2 外観·接続図

USBPI0-24 の外観







3.3 各部名称と機能



- 電源ジャック
 2 電源選択ジャンパー
 5V DC 電源(外部電源)を接続します。
 ボードの電源を USB にするか 外部電源にするかを選
- 沢します。工場出荷時は外部電源(電源ジャック側)

 いなっています。

 ③ USB コネクター

 ④ 拡張コネクター2

 ⑤ RS232C コネクター

 ⑥ ICSP ヘッダー

 PICKIT-2 などのプログラマを接続します。

 ICD-2, Real ICE などを使う場合にはヘッダーピン
 - への変換コネクタが必要です。
- ⑦ 輝度調整ボリューム STN キャラクター表示器を繋いだ場合のコントラスト 調整をします。
- ⑧ リセットボタン CPU をリセットします。
- ⑨ バックライト用電源 バックライト電流を供給します。

 ① STN 表示器コネクター STN キャラクター表示器を接続します。
 ① バックライト用電流制限抵抗 バックライト電流に応じて値を決めます。 工場出荷時は R = 5.1 Ωです。バックライト電流は Ib1 = (5.0 - VLED) / R (VLED は バックライト LED の電圧降下)
 ② 拡張コネクター1 ポート A 入出力です。 詳細は 3.4 参照。 コマンドで設定読み出すことができ、ボードの番号な どに利用できます。

3.4 コネクター・ピン配置

PIN	機能	接続先	PIN	機能	接続先
1	PAO	IC1-2	2	+5V	
3	PA1	IC1-3	4	+5V	
5	PA2	IC1-4	6	+5V	
7	PA3	IC1-5	8	+5V	
9	PA4	IC1-7	10	GND	
11	PA5	IC1-8	12	GND	
13	PA6	IC1-9	14	GND	
15	PA7	IC1-10	16	GND	

CN-1 拡張コネクター1

7	PA3	IC1-5	8	+5V	
9	PA4	IC1-7	10	GND	
11	PA5	IC1-8	12	GND	
13	PA6	IC1-9	14	GND	
15	PA7	IC1-10	16	GND	

CN-2 拡張コネクター2

PIN	機能	接続先	PIN	機能	接続先
1	GND		2	GND	
3	PCO	IC1-19	4	PC1	IC1-20
5	PC2	IC1-21	6	PC3	IC1-22
7	PC4	IC1-27	8	PC5	IC1-28
9	RC6	IC1-29	10	PC7	IC1-30
11	PBO	IC1-33	12	PB1	IC1-34
13	PB2	IC1-35	14	PB3	IC1-36
15	PB4	IC1-37	16	PB5	IC1-38
17	PB6	IC1-39	18	PB7	IC1-40
19	+5V		20	+5V	

CN-4 ICSP ヘッダー

PIN	機能	接続先	PIN	機能	接続先
1	MCLR-/Vpp	IC1-1	2	VDD	+5V
3	GND		4	PGD	IC1-40
5	PGC	IC1-39	6	未使用	

CN-5 STN 液晶表示器 コネクター



!注意 このコネクターは STN 液晶表示器の裏側に MIL フラッ トケーブルコネクタを挿入する前提で偶数列と奇数列を入れ替えてあります。 偶数列と奇数列を反対にして丁度合う液晶表示器をお使いください。
中には電源と GND ピンが逆の液晶表示器もありますのでご注意ください。

PIN	機能	接続先	PIN	機能	接続先
1	Vdd	+5V	2	Vss	GND
3	RS	IC1-6	4	VLC	VR1
5	E	IC1-7	6	R/W	GND
7	DB1	未接続	8	DBO	未接続
9	DB3	未接続	10	DB2	未接続
11	DB5	IC1-3	12	DB4	IC1-2
13	DB7	IC1-5	14	DB6	IC1-4

※ STN 液晶表示器は付属しておりません。ご利用の際は別途ご購入ください。 推奨品は SC2004C (秋月電子通商様扱い)です。

STN 液晶表示器側

PIN	機能	PIN	機能
1	Vss(GND)	2	Vdd (+5V)
3	VLC (コントラスト)	4	RS
5	R/W	6	Е
7	DBO	8	DB1
9	DB2	10	DB3
11	DB4	12	DB5
13	DB6	14	DB7

PIN	機能	接続先	PIN	機能	接続先
1	DCD (IN)	4pin	2	TxD(OUT)	IC2-14
3	RxD(INT)	IC2-13	4	DSR(IN)	6pin
5	GND	GND	6	DTR (OUT)	4pin
7	CTS(IN)	8pin	8	RTS (OUT)	7pin
9	RI	未接続			

CN-6 RS232C コネクター

注) CTS, RTS を使ったハードウェアによるハンドシェークは使えません。 内部で DTR - DSR, RTS - CTS を結線しております。 パソコンとの接続は 9 ピン オス・メス ストレートケーブルをお使いください。

3.5 電気的特性

以下にポート入出力(コネクタ CN1, CN2)の電気的特性を示します。

絶対最大定格

項目	値	条件
最大出力シンク電流	25mA	
最大出力ソース電流	25mA	

注)上記の値は CMOS 入力電圧レベルを保証できる電流値ではありません。

USBPI0-24 ボードはコントローラに PIC18F4553 (または PIC18LF4553)を使用して おります。

詳しくは付属 CD-ROM の Microchip フォルダーにある PIC18F4553 のデータシート (英語版)を参照してください。

3.6 電源について

 ・USBPI0-24 ボードの電源は付属の5Vスイッチング電源を使うことも、また USBを使用する場合、USBから供給することも可能です。
 出荷時の初期設定は5V外部電源です。
 以下に設定箇所を示します。 USBから供給する場合はジャンパープラグを図に従って差し替えてください。



■ 外部負荷のない場合の5V電源の消費電流は約 50mA です。

<u>4.ソフトウェア</u>

4.1 ドライバーのインストール

USBPI0-24 を USB で動作させるためにはパソコンに本ボード用のUSBドライバーを インストールする必要があります。(RS232C で動作させる場合は不要です) 以下にその手順を説明します。

パソコンの OS が WindowsXP の場合は 4.1.1、 Vista の場合は 4.1.2 を参照して ください。

64bit OS 対応については当社ホームページの USBPIO-24 の「64bit OS 対応」 のコーナーからドライバーをダウンロードしてインストールしてください。

4.1.1 WindowsXP の場合

PCとUSB ケーブルを接続し、USBPI0-24 の電源を入れます。

下記のダイアログが表示されますので赤丸の項目を選択して「次へ」をクリックします。



「一覧または特定の場所からインストールする」を選択して「次へ」をクリックします。



付属の CD-ROM を挿入し、赤丸の項目を選択して「次へ」をクリックします。

新しいハードウェアの検出ウィザード
検索とインストールのオブションを選んでください。
○ 次の場所で最適のドライバを検索する(S)
下のチェックボックスを使って、リムーバブル メディアやローカル パスから検索できます。検索された最適のドラ イバがインストールされます。
▼ リムーバブル メディア (フロッピー、CD-ROM など)を検索(M)
□ 次の場所を含める(2)
D¥Driver 参照(Q)
○ 検索しないで、インストールするドライバを選択する(D)
一覧からドライバを選択するには、このオブションを選びます。選択されたドライバは、ハードウェアに最適のもの とは取りません。
< 戻る(B) (次へ(N) > キャンセル

自動的にインストールが始まります。もし始まらない場合は「次の場所を含める」 にチェックを入れて CD-ROM の "Driver"フォルダーを「参照」ボタンで選択します。

新しいハードウェブ	アの検出ウィザード
ソフトウェアを	インストールしています。お待ちください
E	USBPIO-24 Driver
	mchpusb.sys コピー先: C*WINDOWS¥System32¥Drivers
	< 戻る(日) 次へ(4) > キャンセル



USBPIO-24 ドライバーのインストールが完了しました。

■ 正常にインストールできていることを確認します。

スタートメニューからコントロールパネルを開き、

システム → ハードウェア タブ→デバイスマネージャ を開きます。



USBPI0-24 を接続した状態で上のように **IO Drivers** の項目に "USBPI0-24 Driver" が表示されていれば正しくボードが認識されています。

4.1.2 Windows Vista の場合

PCとUSB ケーブルを接続し、USBPI0-24 の電源を入れます。

下記のダイアログが表示されますので赤丸の項目を選択します。

➡ 新しいいードウェアが見つかりました	23
USBPIO-24 のドライバ ソフトウェアをインストールする	る必要があります
ドライバ ソフトウェアを検索してインストール このデバイスのドライバ ソフトウェアをインストー 内します。	します (推奨)(<u>L</u>) -ルする手順をご案
→ 後で再確認します(A) 次回デバイスをプラグインするときまたはデバイン きに、再度確認メッセージが表示されます。	スにログオンすると
⑦ このデバイスについて再確認は不要です(D) このデバイスは、ドライバ ソフトウェアをインスト 作しません。	ールするまでは動
	キャンセル

「オンラインで検索しません」を選択します。



下の画面が出たら付属の CD-ROM を挿入します。



下の警告が出たら「このドライバソフトウェアをインストールします」を選択します。





インストールが開始されます。

インストールが終了しました。



■ 正常にインストールできていることを確認します。

スタートメニューからコントロールパネルを開き、 システム → デバイスマネージャ を開きます。



USBPI0-24 を接続した状態で上のように **IO Drivers** の項目に "USBPI0-24 Driver" が表示されていれば正しくボードが認識されています。

4.1.3 Windows7 の場合

PCとUSB ケーブルを接続し、USBPI0-24 の電源を入れます。 一旦このようなエラーが出ます。「閉じる」で終了します。



コントロールパネルから「デバイスとプリンターの表示」を選択します。



USBPI0-24 のアイコンを右クリックして「ハードウェア」タブの「プロパティー」 を開きます。 右下の「プロパティー」ボタンをクリックします。

USBP	D-24	
✓ デバイスの機能:		
名前	種類	
🍌 USBPIO-24	ほかのデバイス	
デバイスの機能の 製造元:	現要 不明	
デバイスの機能の 製造元: 場所:	既要 	

「設定の変更」をクリックします。

全般	ドライバー 詳細	
1	USBPIO-24	
	デバイスの種類	ほかのデバイス
	製造元:	不明
	場所:	Port_#0001.Hub_#0004
ちの デバ この さい	デバイスのドライバーがイ イス情報セットまたは要 デバイス用のドライバーす 。	インストールされていません。(コード 28) 素に2選択されたドライバーがありません。 を検索するには、[ドライバーの更新]をクリックしてくだ
	😚 設定の変更	ドライバーの更新(U)
-		

「ドライバーの更新」をクリックします。

USBPIO-	24		
ニック デバイスの	種類: ほかの	デバイス	
製造元:	不明		
場所:	Port_#	0001.Hub_#0004	
デバイスの状態			
このデバイスのドライ	イバーがインストール	されていません。 (コー	۲ 28)
デバイス情報セット	または要素に選択さ	れたドライバーがありま	せん。
このデバイス用のド さい。	ライバーを検索する	こは、[ドライバーの更筆	府] をクリックしてくだ
		(15-	(バーの更新(U)…
		-	

「コンピュータを参照してドライバーソフトウェアを検索します」をクリック

● ■ ドライバー ソフトウェアの更新 - USBPIO-24	
どのような方法でドライバー ソフトウェアを検索しますか?	
→ ドライバー ソフトウェアの最新版を自動検索します(S) このデバイス用の最新のドライバー ソフトウェアをコンピューターとインター ネットから検索します。ただし、デバイスのインストール設定でこの機能を無効 にするよう設定した場合は、検索は行われません。	
→ コンピューターを参照してドライバー ソフトウェアを検索します(R) ドライバー ソフトウェアを手動で検索してインストールします。	
+v>	216

参照ボタンで Driver フォルダーを選択します。

•

	×
コンピューター上のドライバー ソフトウェアを参照します。	
次の場所でドライバー ソフトウェアを検索します: D:¥Driver マサブフォルダーも検索する(I)	
コンピューター上のデバイス ドライバーの一覧から選択します(L) この一覧には、デバイスと互換性があるインストールされたドライバー ソフトウェア と、デバイスと同じカテゴリにあるすべてのドライバー ソフトウェアが表示されま す。	
<u>次へ(N)</u> キャン	rtil)

「このドライバーソフトウェアをインストールします」をクリック。

The second sec	D更新 - USBPIO-24 ンストールしています
	Windows セキュリティ Fライバー ソフトウェアの発行元を検証できません
	→ このドライバー ソフトウェアをインストールしない(N) お使いのデバイス用の、更新されたドライバー ソフトウェアが存在するか どうか製造元の Web サイトで確認してください。
	 このドライバー ソフトウェアをインストールします(I) 製造元の Web サイトまたはディスクから取得したドライバー ソフトウェアのみインストールしてください。その他のソースから取得した署名のない ソフトウェアは、コンピューターに危害を及ぼしたり、情報を盗んだりする 可能性があります。 詳細の表示(D)

X

完了画面が出たら「閉じる」をクリックします。

◯ ■ ドライバー ソフトウェアの更新 - USBPIO-24 Driver	
ドライバー ソフトウェアが正常に更新されました。	
このデバイスのドライバー ソフトウェアのインストールを終了しました:	
USBPIO-24 Driver	
(開じる(C)

先ほど開いたデバイス画面で黄色い△マークが消えていることを確認します。



4.2 コマンド一覧

USBPI0-24 は USB または RS232C より簡単な文字列のコマンドを受け取って A, B, C 3つのポート(各8ビット)を制御します。

4.2.1 コマンドの形式

コマンドはすべて ASCII 文字列で数値はすべてヘキサ(16進数)にて送ります。 最初の2文字がコマンドで、あとポートの番号(A:00 B:01 C:02)、制御データ などのパラメータが続きます。

モード設定で「チェックサムを有効にする」指定をした場合はコマンドの後に チェックサム(それまでの文字をヘキサ値で合計した数値の下8ビットを2桁 ヘキサ文字にしたもの)を追加します。

チェックサムの計算例は 4.3 章のサンプルプログラムを参照してください。 コマンド、パラメータ(、チェックサム)の後にターミネータとして CR (キャリッジ・リターンコード = 0x0d)を付加してください。

USBPIO-24 は CR を受け取った時点でコマンドを解析して実行します。

コマンド実行に成功すれば"OK"失敗すれば "NGxx" の文字列を返します。

(xx はエラー番号)応答のターミネータも CR です。

ただしモード設定で「応答なし」を指定した場合、USBPIO-24 はコマンドに対する 応答を返しません。(ADC モード、ポート入力モードの場合を除く)

各コマンドは USB と RS232C で共通であり、USBPI0-24 は常に両方の入力を 監視しておりますので、USB/RS232C どちらを使うかの設定は不要です。

コマンドは以下の種類があります。

■ <u>モード設定コマンド</u> 形式: MDmm

最初の2文字"MD" がコマンド、続く2文字でモードを指定します。 2章の繰り返しになりますが、下記の3つのモードのいずれかを指定します。 USBPI0-24 は以下の3つの機能を有しています。

- PIO モード(パラメータm:00)
 USB/RS232C で 24bit のパラレル I/0 を制御(入出力)できます。
 1bit ごとに IN/OUT の方向を設定可能。

 ADC モード(パラメータm:01)
 - 12bit A/D コンバータ(最大8チャンネル)からの入力を USB/RS232C で 取り込みます。A/D コンバータ入力ピンはポートA と共通です。 データロガーとして利用可能。
- STN モード(パラメータmm:02)
 USB/RS232C で STN 4行x20文字キャラクター液晶表示器に文字を 出力します。

上記パラメータ mm で更に bit-7 (0x80) が 1 のときチェックサム有効となり、 USBPIO-24 はサムチェックをおこないます。 この場合次から送るコマンドには XXppcc の形式 (XX はコマンド, pp はパラメータ、 cc はチェックサムコード) になります。 USBPIO-24 で cc をチェックして正しくなければエラーを返してコマンド

は実行されません。

bit-6(0x40)が1のときは「応答なし」モードとなり、コマンドに対して 応答を返しません。(ADC コマンド、ポート入力コマンドを除く)

bit-5(0x20)が1のときは 外部電源モードとなり、ポートA のビット3 は Vref+ (外部基準電源)、ポートA のビット2は Vref-(外部基準電源 GND)に なります。

この設定は ADC モードの時のみ有効で、このモードを設定する場合は PA2, 3 に外部基準電源を接続する必要があります。

コマンド	MDmm		
機能	モードを設定する		
引数			
mm	00	PIOモード	
	01	ADC モード	
	02 STN モード		
	更に bit-7 (0x80)が1のとき:チェックサム有効		
	bit-6 (0x40)が 1のとき:応答なし		
	bit-5(0x20)が1のとき: 外部電源モード		
応答	"OK": 成功、"NGxx": 失敗		
	MD81 ADC モードでチェックサムを有効にする		
使用例	使用例 MDC0 PIO モードでチェックサム有効+応答な		
	MD42	STN モードで応答なし	
備考 USBPI0-24 を使用する前に必ず発行してください 途中でモードを変更することも可能です。		24 を使用する前に必ず発行してください。	
		っでモードを変更することも可能です。	

・チェックサムの計算法

MDコマンドで「チェックサム有効」を指定した場合は、コマンド文字列の末尾 (ターミネータ'CR'の前)にチェックサムを追加します。 チェックサムはコマンド文字列の各文字のアスキー値を合計したものの下 8ビットを2桁のヘキサで表します。 例えばボードに送信するコマンドが ポート B の読み込みの場合、コマンド文字列は "PI01" ですので、各文字の ASCII 値を合計すると P I 0 1 0x50 + 0x49 + 0x30 + 0x31 = 0xFA となるので送信する文字列は チェクサムの 2文字を追加して "PI01FA{CR}" となります。({CR} はターミネータの 0x0d)

・サンプルプロジェクトでは USBPI024Test の CUSBPI024TestDlg::SendCommand
 関数のところでチェックサム計算をおこなっています。

・ご注意

モード設定コマンドのみはボード側で受け取ったとき、チェックサムの検査を しないので、チェックサムエラーは発生しません。

■ <u>ポート方向設定コマンド</u> 形式: PDppvv

最初の2文字"PD" がコマンド、続く2文字でポートを指定し、最後の 2文字でそのポートの方向を指定します。 モード設定で「チェックサム有効」を指定したときは PDppvvcc (cc はチェックサム) の形式になります。 pp 00:ポートA、 01:ポートB、 02:ポートC

vv 1のビットが入力、0のビットが出力に設定されます。

コマンド	PDppvv		
機能		各ポートの入出力方向を設定する	
引数	値	内容	
pp	00	ボートA	
	01 ポートB		
	02 ポート C		
VV	00FF 1のビットが入力、0のビットが出力		
応答	"OK":成功、"NGxx":失敗		
	PD01FF ポートBを入力に設定		
使用例	PD02C0 ポートCのbit-6,7を入力、他のビットを出力に		
	PD005F	ポート A の bit-5,7 を出力、他のビットを入力に	
/## ##	ポート A は ADC モードでは入力、STN モードでは出力に		
価考 固定されますので PIO モードでのみポート A		で PIO モードでのみポート A 設定は有効です。	

■ <u>データ出力コマンド</u> 形式: P0ppvv

最初の2文字"P0" がコマンド、続く2文字でポートを指定し、最後の 2文字でそのポートの方向を指定します。

注) "PO "の2文字目は「ゼロ」ではなく OUT の「オー」です。

モード設定で「チェックサム有効」を指定したときは **POppvvcc**(cc はチェックサム) の形式になります。

pp 00:ポートA、 01:ポートB、 02:ポートC

vv 00~FF のヘキサ値をポートに設定します。

コマンド	POppvv		
機能	ポートにデータをラッチ(出力)する		
引数	值 内容		
pp	00	ボート A	
	01 ポートB		
	02 ポート C		
VV	00FF 出力値		
応答	"OK":成功、"NGxx":失敗		
	P001FF ポートBをすべて1に		
使用例	P002C0	ポート C の bit-6,7 を 1 に、他のビットを 0 に	
	P0005F	ポート A の bit-5,7を1に、他のビットを0に	
	ポート方向設定コマンドで「出力」に設定されたビットに対して		
伽有	のみ有効です。		

■ <u>データ入力コマンド</u> 形式: PIpp

最初の2文字"PI"がコマンド、続く2文字でポートを指定します。

モード設定で「チェックサム有効」を指定したときは **PIppcc**(cc はチェックサム)の形式になります。

pp 00:ポートA、 01:ポートB、 02:ポートC

コマンド	PIpp		
機能	ポートからデータを入力する		
引数	値	内容	
pp	00	ボートA	
	01 ポート B		
	02 ポート C		
応答	PIppvv	vv はポート入力値(00FF)	
使用例	PI01 ポートBを読み込む		
備考	ポート方向設定 のみ有効です。	コマンドで「入力」に設定されたビットに対して 他のビットは取り込み時にマスクしてください。	

■ <u>A/D 変換コマンド</u> 形式: ADhh

最初の2文字"AD" がコマンド、続く2文字でADC のチャンネルを指定します。 モード設定で「チェックサム有効」を指定したときは ADhhcc (cc はチェックサム) の形式になります。 A/D 変換の精度は12ビットですので 000~FFF の値を返します。

hh $00 \sim 07$

コマンド	ADhh			
機能	A/D 変換値を取得する			
引数	值内容			
hh	0007 A/D コンバータのチャンネル			
応答	ADhhvvvv	vvvv は A/D 変換値(00000FFF)		
使用例	ADO2 ADC チャンネル2を A/D 変換する			
備考	モード設定コマ 他のビットは取り	ンドで「ADC モード」に設定された場合のみ有効です。 込み時にマスクしてください。		

■ <u>STN 表示コマンド</u> 形式:LDrrmmnnvvv...vv

STN液晶キャラクターディスプレイに文字列を表示します。

最初の2文字"LD" がコマンド、続く2文字rrで表示する行を指定し、続く2文字 mm で桁を指定し、続く2文字 nn が表示する文字数、最後が表示する文字列(可変長) になります。 液晶の左上が座標原点となります。

モード設定で「チェックサム有効」を指定したときは LDrrmmnnvvv...vvcc (cc はチェックサム)の形式になります。

4桁x20文字のSTNキャラクターディスプレイ(10ページを参照)を対象と します。STNキャラクターディスプレイは専用コネクタ(CN5)でポートAに 接続されています。

コマンド	LDrrmmnnvvvvv		
機能	STN液晶キャラクターディスプレイに文字列を表示する		
引数	値 内容		
rr	0003	表示行番号(縦方向)	
mm	0013 表示桁位置(横方向)		
nn	0014 文字列の長さ(ヘキサ値)		
vvvvvv	可変長 nn で指定した長さの文字列		
応答	"OK":成功、"NGxx":失敗		
体田園	Image: Ideal and the second systemLD010005Hello2 行目の頭から"Hello" の5 文字を表示 3 行目の5 文字目から"ZigZag" の6 文字		
使用例			
備考	モード設定コマンドで「STN モード」に設定された場合のみ有効です。		

■ <u>STN 表示クリアコマンド</u> 形式:LRrr

STN液晶キャラクターディスプレイの文字列を消去します。 最初の2文字"LR"がコマンド、続く2文字rrで表示する行を指定します。 モード設定で「チェックサム有効」を指定したときはLRrrcc (ccはチェックサム)の形式になります。

コマンド	LRrr		
機能	STN液晶キャラクターディスプレイの文字列を消去する		
引数	値	内容	
rr	0003	消去する行番号 4以上のときは	
mm	0013	表示桁位置	
nn	0014 文字列の長さ(ヘキサ値)		
vvvvvv	可変長 nn で指定した長さの文字列		
応答	"OK":成功、"NGxx":失敗		
	LR02	3 行目を消去する	
使用例	LR04	LR04 STN 全体を消去する	
備考	モード設定コマンドで「STNモード」に設定された場合のみ有効です。		

■ <u>STN カーソル設定コマンド</u> 形式:LCrrmm

STN液晶キャラクターディスプレイの文字列を消去します。 最初の2文字"LC"がコマンド、続く2文字rrでカーソルを表示する行、 続く2文字mmでカーソルを表示する桁位置を指定します。 モード設定で「チェックサム有効」を指定したときはLCrrmmcc (cc はチェックサム)の形式になります。

コマンド	LCrrmm		
機能	STN液晶キャラクターディスプレイにカーソル表示/消去する		
引数	值	内 容	
1010	00 02	カーソルの行番号	
L.L.	0003	4以上のときはカーソル非表示	
mm	0013	カーソル表示桁位置	
応答	"OK":成功、"NGxx":失敗		
体田庙	LC0203	3行目の4文字目にカーソル表示	
使用例	LC0400	カーソルを非表示	
備考	モード設定コマンドで「STN モード」に設定された場合のみ有効です。		

■ <u>DIP スイッチ読み込みコマンド</u>形式: CH

DIP スイッチの設定を読み込みます。
ボードの番号区別などに利用することができます。
最初の2文字"CH" がコマンドです。
モード設定で「チェックサム有効」を指定したときは CHcc (cc はチェックサム)の形式になります。

応答は ON のビット が 1、OFF のビット が 0 になります。 DIP スイッチ上の番号とビットの対応は

- 1 bit-0
- 2 bit-1
- 3 bit-2
- 4 (未接続)

となります。

コマンド	СН			
機能	DIP スイッチの設定値を読み込みます。		DIP スイッチの設定値を読み込みます。	
引数	值 内容			
応答	CHvv	vv が DIP スイッチ読み込み値(0007)		
使用例	СН			
備考	DIP スイッチの bit-3 は未接続のため無効です。			

4.2.2 エラーコード一覧

コマンドにエラーがあった場合(モード設定で「応答なし」を設定いなければ) "NGxx" (xx は下記のエラー番号)を返します。

エラー名	エラー番号	内容
NOERROR	00	エラーなし (成功)
ERR_ILLEGAL_COMMAND	01	存在しないコマンド
ERR_ILLEGAL_PORT	02	不正なポート番号指定
ERR_ILLEGAL_VALUE	03	不正なパラメータ
ERR_ILLEGAL_MODE	04	不正なモード指定
ERR_ILLEGAL_FORMAT	05	不正なフォーマット
ERR_CHECKSUM	06	チェックサムエラー

4.3 サンプルプログラムの実行

・サンプルプログラムは付属 CD-ROM の"Sample Programs" フォルダーに以下の 3本が入っています。

1. USBPIOTest

USBPIO-24 を USB で制御する VC++ のサンプルです。

2. USBPI0Test2

USBPIO-24 を RS232C (COM ポート) で制御する VB(Visual Basic) の サンプルです。

3. USBPI0Test3

USBPIO-24 を USB で制御する VB(Visual Basic) のサンプルです。

4. USBPIOTest4

複数台の USBPIO-24 を USB で制御する VB(Visual Basic) のサンプルです。

- ・プログラムの実行には.NET Framework 2.0 以上の環境が必要です。
 OS が Vista の場合はそのままお使いいただけますが、Windows XP で
 .NET Framework がインストールされていない場合は 付属 CD-ROM の
 redist フォルダーにある vcredist_x86.exe を実行して Visual C++ 2005
 再頒布可能パッケージ (x86)を予めインストールしてください。
- ・もしサンプルプログラムを改造する場合はコンパイルに VisualStudio 2005 以上の開発環境が必要です。
- ・CD-ROM の "Sample Programs" フォルダーをどこかハードディスクの適当な 場所にコピーしてください。

4.3.1 USBPIOTest の実行

USBPIO-24 ボードをパソコンと USB ケーブルで接続し、電源を入れて Sample Programs の下の bin フォルダーの USBPIOTest.exe を実行します。

🏭 USBPIO24 Test for Windows	🖁 USBPIO24 Test for Windows				
	🗖 チェックサム	Γι	スポンス不要		
モード設定	ポートの方	句設定(0:出力)	1:入力)———		
● PIO モード	ポート A	00	設定		
○ ADC モード	ポートB	FF	設定		
O STN €-ド	ポートC	FF	設定		
	∟				
(Hex)	- 単力 - 1	2.77	00		
#-FR 00	/ 	<u></u>	55		
#	/				
-ro [00]			1.00		
チャンネル 🛛 💌	A/D コンバータ入ナ 	1	(Decimal)		
	— STN 液晶表示 ·				
開始アドレス(10進) 0			送信		
送信文字列	Welcome to l	JSBPIO-24			
אראב MD00		応答 01	(
			終了		

- モード設定で PIO モードを選択します。 ラジオボタンをクリックした時点で USBPIO-24 にコマンドが送られます。 送ったコマンド内容は画面下の「コマンド」 のところにそのまま表示されます。 それに対する USBPIO-24 から受け取った応答が 「応答」のところに表示されます。
- ・画面上の「チェックサム」にマークをつけると次からのコマンドはチェックサム 付きで送られて、USBPI0-24 でもサムチェックをおこないます。
- ・ポートの方向を2桁へキサで入力し、設定ボタンを押してコマンドを送ります。
- ・設定する値をポート A, B, C に入力し「出力」ボタンをクリックすると
 その値が各ポートに設定されます。
 「入力」ボタンをクリックすると入力に設定したビットを読み込んでヘキサで
 表示します。

・次にモード設定を ADC モードに変更します。

How Windows USBPIO24 Test for Windows	:	
	Γ チェックサム	□ レスポンス不要
	ポートの方信]設定(0: 出力 1: 入力)———
○ PIO モード	ポート A	FF 設定
ADC モード	ポート B	FF 設定
○ STN モード	ポートC	FF 設定
(Hex)		入力 00
-rc [00		
	A/D コンバータ入力	(Decimal)
チャンネル 3 💌		入力 [4083]
	——STN 液晶表示—	
開始アドレス(10進) 0		送信
送信文字列	Welcome to U	SBPIO-24
אעדב <mark>AD03</mark>		応答 AD030FF3
		終了

- ・チャンネル番号を選んで「入力」ボタンをクリックすると A/D 変換した値を 0~4095 (*注)の十進数で表示します。
 - *注) A/D変換の誤差により下限、上限は 正確に 0,4095 付近の値にならない 場合があります。
- ・USBPI0-24 ボードに送っているコマンドの詳細については 4.2 を参照して ください。

・次にモード設定を STN モードにします。

STN キャラクターディスプレイ(製品には付属しておりません)をCN5 コネクターに接続します。

🖁 USBPIO24 Test for Windows			
	🗖 ቻェックサム	🗖 レスポンス	不要
モード設定	ポートの方г	句設定(0: 出力 1: 入力)
○ PIO モード	ポート A	00 設定	
○ ADC モード	ポート B	FF 設定	
⊙ STN モード	ポートC	FF 設定	
(Hex)	Ψ π		_
	<u>ш</u> т (-
ポートC [00	出刀]
	A/D コンバータ入ナ		
チャンネル 🛛 💌		(Dec 入力 0	imal)
開始アドレス(10進) 0		送信	
送信文字列	Welcome to l	JSBPIO-24	=
,			
コマンド LD000014Welcome t	D USBPIO-24	応答	
			\$7

- ・送信文字列のところに文字列を入力して「送信」ボタンをクリックすると USBPIO-24 に接続した STN キャラクターディスプレイに文字列が 表示されます。
- ・開始アドレスのところには rr x 20 + mm (rr: 行番号、mm: 桁位置)の
 数値を入力してください。 たとえば2行め(rr = 1)の5文字目(mm = 4)
 から表示したい場合、アドレスは 1 x 20 + 4 = 24 になります。

・ご注意

STN モードに切り替えたあとは**キャラクタディスプレイの初期化に時間が** かかるため1秒程度待ってから次のコマンドを送ってください。

4.3.2 USBPIOTest2 の実行

USBPI0-24 ボードとパソコンを RS232C ケーブル (本製品には付属しません) で接続し、電源を入れます。

Sample Programs の下の bin フォルダーの USBPIOTest2.exe を実行します。 USBPIOTest2 ではUSB のかわりに RS232C (COM ポート) からコマンド を送ります。 4.3.1の USBPIOTest.exe に比べて簡易な仕様になっています。 9 ピン オス-メス ストレートケーブルでパソコンと USBPIO-24 を接続して ください。

MODE ボタンをクリックすると

・ポートA の全ビットを ADC 入力に

ポートBの全ビットをデジタル入力に

・ポートCの全ビットをデジタル出力に設定します。

モード設定に成功するとボタンをイネーブルします。

🖶 USBPIO-24 (RS232C)	_ D X
ポート選択 COM1	モード設定
(00.FF) (Hex)	ポートB 入力
(00.FF) (Hex)	ポートC出力
(0.4095) (Dec)	ADC 入力
Command PD0200	Reply OK

「ポートB入力」ボタンをクリックするとポートBからの入力値をヘキサで表示します。

🔜 USBPIO-24 (RS232C)	
ポート選択 COM1 ▼	モード設定
(OO.FF) FF (Hex)	ポートB 入力
(00.FF) (Hex)	ポートC 出力
(0.4095) (Dec)	ADC 入力
Command PID1	Reply PI01FF

🔜 USBPIO-24 (RS232C)	
ポート選択 COM1	モード設定
(00.FF) FF (Hex)	ポートB 入力
(00.FF) 59 (Hex)	ポートC 出力
(0.4095) (Dec)	ADC 入力
Command PO0259	Reply OK

左に数値をヘキサで入力して「ポートC出力」ボタンをクリックすると ポートCに値を設定します。

ADC入力ボタンをクリックするとチャンネルOのアナログ値を A/D 変換して 1 O 進数(0~4095)で表示します。

🛃 USBPIO-24 (RS232C)	
ポート選択 COM1 🗨	モード設定
(00.FF) (Hex)	#−トB 入力
(00.FF) (Hex)	ポートC 出力
(0.4095) 1384 (Dec)	ADC 入力
Command AD00	Reply AD000568

・USBPI0-24 ボードに送っているコマンドの詳細については 4.2 を参照して ください。

4.3.3 USBPIOTest3 の実行

Sample Programs の下の bin フォルダーの USBPIOTest3.exe を実行します。 USBPIOTest3 は USB を使った VB による I/O 制御サンプルプログラムで内容は 4.3.2 の RS232C 版とほぼ同等の仕様です。 USB なので COM ポート設定がない点だけが違います。 MODE ボタンをクリックすると USBPIO24Test2 と同様 ・ポートA の全ビットを ADC 入力に ・ポートB の全ビットをデジタル入力に ・ポートC の全ビットをデジタル出力に設定します。 モード設定に成功するとボタンをイネーブルします。 以下、ポート B, C, A の設定、読み込みも USBPIO24Test2 と同様におこなえます。

4.3.4 USBPIOTest4 の実行

2台の USBPIO-24 を USB に接続して Sample Programs の下の bin フォルダーの SBPIOTest4.exe を実行します。

USBPIO-24 Test4
Device-0
ポートB(出力値) 5A 設定
Device-1
ポートB(出力値) 6B 設定
終了

複数のボードを USB に接続すると先に接続したボードから順にデバイス ID が 0, 1, 2 ・・が与えられます。

このサンプルプログラムでは Device-0 の[接続]ボタンをクリックすると デバイス0 に対して USB 接続をオープンし、ポートBを出力に設定するコマンドを 送ったあとデバイス0の DIP スイッチを読み込んで表示します。 [設定]ボタンをクリックするとそのボードのポートBにヘキサで入力した値を出力 します。 デバイス1 も同様です。

接続順に依存しないでボードを識別するためにはボードのDIP スイッチをユーザー プログラムで読み込んでデバイス ID とボードを関連付ける必要があります。

具体的にはデバイス ID とチャンネル番号(DIP スイッチ)の対応表を作っておいて アクセスしたいボードのチャンネル番号からそのデバイス ID を取得し、そのデバイス ID を引数に USB の読み書きをおこないます。

このサンプルプログラムでは DIP スイッチを読み込むところまでを実装しています。

4.3.5 ボードの初期状態

パワーオン直後のボードの各ポートは PIO モードですべて入力に設定されます。

4.3.6 USB の速度などについて

USB 自体の転送速度は Full Speed 対応で最大 12Mbps ということになって いますが、コマンドを送ってからポートに反映されるまでの時間はパソコンの ドライバーとプログラムの条件などによって異なりますのでユーザー様にて ご確認願います。

ハンドシェークなしで連続転送した場合、PC側が速すぎるとオーバーフローする 可能性があります。

4.4 サンプルプログラム解説

4.4.1 USB による I/O 制御(VC++版)

付属 CD-ROM の Sample Programs フォルダーの USBPI024Test は VisualStudio2005 VC++ で作成した USB による I/O 制御サンプルです。

使い方は 4.3 に書いた通りです。 ここではプログラムの USB 制御コアについて 説明します。

プロジェクトの起動はUSBPI024Test フォルダーのUSBPI024Test.vcproj を ダブルクリックしておこなえます。

以下にPC側のプログラム階層図を示します。





関数名	機能
SendCommand	コマンド文字列を USB に送出する
SetDirection	ポートの方向(入出力)を設定する
SetMode	モードを設定する
SetPort	ポートの出力を設定する
GetPort	ポートの入力を読み込む
OnBnClickedXxxx	各ボタンの応答関数(Xxxx はボタン名)

ユーザーインターフェースと主な制御は USBPI024TestDlg.cpp にあります。

SendCommand 関数によって USB ドライバーにコマンドが送られて USBPIO-24 ボードが これを受け取り I/O 制御がおこなわれます。

USB 制御のコアは mchusb.cpp ファイルに CMchUSB クラスとしてカプセル化 されています。

このクラスはマイクロチップ社提供のDLL mcusbapi.dll を経由して USB

ドライバーを呼び出します。以下はこのクラスの主な関数です。

CMchUSB	ク	ラス	の概要
---------	---	----	-----

関数名	機能
Initialize	USB の初期化 プログラムの開始時に一度だけ呼んでください
Open	入出力のパイプをオープンする
Close	入出力のパイプをクローズする
Write	出力パイプにデータを書き込む
Read	入力パイプからデータを読み出す

USB の利用手順は

- CMchUSB クラスのインスタンスを作成する サンプルプログラムでは mchusb. cpp の 31 行目あたりにある CMchUSB gMchUSB;
 でグローバルインスタンスを定義しています。
- 2. Initialize で USB の初期化をおこなう。
- **3.** Open 関数でパイプをオープンする。
- 4. Read/Write 関数でパイプに対して読み書きをおこなう
- 5. 使用が終われば Close でパイプを閉じる

となります。

以下は CMchUSB クラスの関数仕様です。

CMchUSB クラスの関数仕様

関数名	Initialize	
機能	USB の初期化	
引数	タイプ	内容
なし		
戻り値	BOOL	TRUE: 成功、FALSE:失敗
備考		

関数名	Open	
機能	パイプのオープン	
引数	タイプ 内容	
iDevice	int	複数のボードを扱う場合のデバイス番号(07)
戻り値	BOOL	TRUE: 成功、FALSE:失敗
備考		

関数名	Close	
機能	パイプのクローズ	
引数	タイプ	内 容
iDevice	int	複数のボードを扱う場合のデバイス番号(07)
戻り値	BOOL	TRUE: 成功、FALSE:失敗
備考		

関数名	Write	
機能	パイプへの書き込み	
引数	タイプ	内容
iDevice	int	複数のボードを扱う場合のデバイス番号(07)
pData	BYTE*	データバッファへのポインタ
nLength	int	データのバイト数
戻り値	int	-1: エラー、> 0: 書き込んだバイト数
備考	USBPI0-24 側:	エンドポイント(チップ内バッファ)のサイズは
	64バイト	

関数名	Read	
機能	パイプの読み出し	
引数	タイプ	内容
iDevice	int	複数のボードを扱う場合のデバイス番号(07)
pData	BYTE*	データバッファへのポインタ
nLength	int	データのバイト数
nTimeout	int	タイムアウト (msec)
戻り値	int	-1: エラー、>=0: 読み込んだバイト数
備考		

4.4.2 RS232C による I/O 制御(VB版)

付属 CD-ROM の Sample Programs フォルダーの USBPI024Test2 は VisualStudio2005 VB(Visual Basic)で作成した RS232C による I/O 制御サンプルです。 使い方は 4.3 に書いた通りです。 ここではプログラムの制御コアについてのみ 説明します。

プロジェクトの起動はUSBPI024Test2 フォルダーのUSBPI024Test2.vcproj を ダブルクリックしておこなえます。

関数名	機能
ButtonMode_Click	モード設定ボタンのクリック処理
ButtonPortB_Click	ポートB入力ボタンのクリック処理
ButtonPortC_Click	ポートC出力ボタンのクリック処理
ButtonADC_Click	ADC 入力ボタンのクリック処理
GetReply	USBPI0-24 ボードからの応答文字列取得
IsHex	文字列が有効なヘキサ数字かチェック

「モード設定」ボタンのクリックにより SerialPort1 をオープンし、モード 設定コマンドを USBPIO-24 ボードに送ります。 SerialPort1 の設定は Visual Studio の「プロパティー」ダイアログで baud: 19200, 8N1 No Parity に設定しています。 ボーレートはこの値で固定になります。

Form1.vb の関数概要

4.4.3 USB による I/O 制御 (VB版)

付属 CD-ROM の Sample Programs フォルダーの USBPI024Test3 、USBPI024Test4 は VisualStudio2005 VB(Visual Basic) で作成した USB による I/O 制御サンプルです。 使い方は 4.3 に書いた通りです。 ここではプログラムの USB 制御コアについて 説明します。

プロジェクトの起動はUSBPI024Test3 フォルダーのUSBPI024Test3.vcproj を ダブルクリックしておこなえます。

以下にPC側のプログラム階層図を示します。

PCプログラム階層図

アプリケーション



図のように USB制御のコアは USBPI024.vb の USBPI024 クラスにカプセル化 されています。

アプリケーションからの USB 制御はこのクラスを使うことで mpusbapi.dll 経由で ドライバーを呼び出すことができます。

Form1.vb の関数概要

関数名	機能
ButtonMode_Click	モード設定ボタンのクリック処理
ButtonPortB_Click	ポートB入力ボタンのクリック処理
ButtonPortC_Click	ポートC出力ボタンのクリック処理
ButtonADC_Click	ADC 入力ボタンのクリック処理
GetReply	USBPI0-24 ボードからの応答文字列取得
IsHex	文字列が有効なヘキサ数字かチェック

Form1.vb ファイルのForm1_Load 関数でこのクラスのインスタンス usbpio を 生成します。

「モード設定」ボタンのクリックにより USB をオープンし、モード設定コマンドを USBPIO-24 ボードに送ります。

以下、 USBPI024 クラスの関数一覧を掲載します。

USBPI024 クラスの関数仕様

関数名	New	
機能	コンストラクタ (初期化)	
引数	タイプ	内 容
なし		
戻り値		
備考	インスタンスの生成で自動的にコールされます。	

関数名	Open	
機能	パイプのオープン	
引数	タイプ	内容
iDevice	int	複数のボードを扱う場合のデバイス番号(07)
戻り値	BOOL	TRUE: 成功、FALSE:失敗
備考		

関数名	Close	
機能	パイプのクローズ	
引数	タイプ 内容	
iDevice	int	複数のボードを扱う場合のデバイス番号(07)
戻り値	BOOL	TRUE:成功、FALSE:失敗
備考		

関数名	Write	
機能	パイプへの書き込み	
引数	タイプ	内容
iDevice	int	複数のボードを扱う場合のデバイス番号(07)
pData	BYTE*	データバッファへのポインタ
nLength	int	データのバイト数
戻り値	int	-1: エラー、> 0: 書き込んだバイト数
備考	USBPI0-24 側:	エンドポイント(チップ内バッファ)のサイズは
	64バイト	

関数名	Read	
機能	パイプの読み出し	
引数	タイプ	内容
iDevice	int	複数のボードを扱う場合のデバイス番号(07)
pData	BYTE*	データバッファへのポインタ
nLength	int	データのバイト数
nTimeout	int	タイムアウト (msec)
戻り値	int	-1: エラー、>=0: 読み込んだバイト数
備考		

関数名	Is0pen	
機能	指定デバイスがオープンされているかの問い合わせ	
引数	タイプ	内 容
iDevice	int	複数のボードを扱う場合のデバイス番号(07)
戻り値	BOOL	TRUE: オープンされている、FALSE:いない
備考		

4.4.4 おことわり

USB コア使用方法以外のアプリ全般のプログラミング技術や Visual Studio の使い方に 関するお問い合わせはサポート外とさせていただきますのでご了承ください。 USB 部分についてのご不明な点は当社ホームページの「お問い合わせ」より ご質問ください。

4.5 mpusbapi.dll の関数一覧

DLL を直接コールする場合の関数一覧です。

使い方は 4.4.1 4.4.3 のサンプルソースを参照してください。

関数名	MPUSBGetDeviceCount	
機能	接続されているデバイスの数を返す	
引数	タイプ	内容
pVID_PID	char*	ベンダーID, プロダクト ID "vid_04d8&pid_ff33"
戻り値	int	接続されているデバイスの数
備考		

関数名	MPUSBOpen	
機能	パイプのオープン	
引数	タイプ	内容
iDevice	int	デバイス ID
pVID_PID	char*	ベンダーID, プロダクト ID "vid_04d8&pid_ff33"
pEP	char*	エンドポイントの名前 "¥MCHP_EP1"
iDir	int	Write = 0, Read = 1
iReserved	int	未使用
戻り値	int	パイプのハンドル
備考		

関数名	MPUSBRead	
機能	パイプのリード	
引数	タイプ	内容
iHandle	int	パイプのハンドル
pData	char*	データバッファへのポインタ
iLen	int	データバッファ長
pLen	int*	読み込んだデータ長を返す変数へのポインタ
iTimeout	int	タイムアウト時間の指定(msec)
戻り値	int	1: 成功、0:失敗
備考		

関数名	MPUSBWrite	
機能	パイプのリード	
引数	タイプ	内 容
iHandle	int	パイプのハンドル
pData	char*	データバッファへのポインタ
iLen	int	データバッファ長
pLen	int*	読み込んだデータ長を返す変数へのポインタ
iTimeout	int	タイムアウト時間の指定(msec)
戻り値	int	1: 成功、0:失敗
備考		

関数名	MPUSBClose	
機能	パイプのクローズ	
引数	タイプ	内容
iPipe	int	パイプのハンドル
戻り値	int	1: 成功、0:失敗
備考		

4.6 PIC18F4553 開発環境

ユーザー様におかれまして PIC マイコンのプログラムを開発される場合は書き込みを コネクター CN4 (6ピンの ICSP)から PicKit-2 などでおこなうことができます。 ICD-2 などモジュラープラグの書き込み器を使用する場合は変換コネクターが必要です。 統合開発環境 (MPLAB IDE)をマイクロチップ社のホームページ

http://www.microchip.com

から無料でダウンロードすることができます。 最新版はトップページから左の Design タブの"MPLAB IDE" をクリックして MPLAB Integrated Development Environment のページに行き IDE downloads をクリックしてダウンロードし、IDE をインストールしてください。 OS は Windows のみの対応となります。

プログラムの開発にはこの他に C18 コンパイラが必要です。 C18 コンパイラは 無料の Student Edition が評価用に使えます。 マイクロチップ社のトップページから左の Design タブの "Development Tools" をクリックし、飛んだページの Software "MPLAB C Compilers "をクリック してコンパイラーのダウンロードページを表示します。

ページの下の方にある "MPLAB C Compiler for PIC18 v3.22 - Student Edition / " をクリックしてダウンロード、インストールしてください。(Vx. xx は最新のバージョン) Student Edition のダウンロードにはその前に簡単なユーザー登録が必要です。 Student Edition は一定期間を過ぎるとコンパイラの最適化など一部の機能が 使えなくなりますが、コンパイルは継続して無料で可能です。

ご注意

ユーザー様にてファームウェアを消去された場合、再書き込みサービスは 有料となりますのでご注意ください。

(ファームウェアのオブジェクトファイルは CD-ROM に付属しておりません。) 詳細は当社ホームページの「お問い合わせ」コーナーよりお問い合わせください。

5. その他

5.1 オプション製品

本ボードと接続してフォトカプラと小型リレーで絶縁入出力をおこなう **RYPC-8** をオプションで用意しておりますので、ご利用ください。 8ビットの入力(フォトカプラ)と8ビットの出力(リレー) を持ち USBPIO-24 とは MIL フラットケーブルコネクターで接続します。 当社ホームページ <u>http://www.cyber-melon.com</u> の <u>リレー・フォトカプラ絶縁入出力ボード</u>よりご購入いただけます。 詳細資料も上記からダウンロードしてください。

接続例1



接続例2



5.2 特注仕様

- ・数量によりましてはプリント基板のみ、ファームウェア書き込み済み CPU のみのパーツ販売も致します。
- ・本ボードを利用した特注ソフトウェアの開発請負も致します。

詳細は当社ホームページ <u>http://www.cyber-melon.com</u> お問い合わせコーナー よりご連絡ください。

本書の改訂版は当社ホームページの該当製品コーナーよりダウンロードしてください。

Cyber MELON

株式会社インターネット 〒665-0841 兵庫県宝塚市御殿山 2-25-39 <u>http://www.cyber-melon.com</u> e-mail: <u>info#cyber-melon.com</u> (# を @ に置き換えてください)