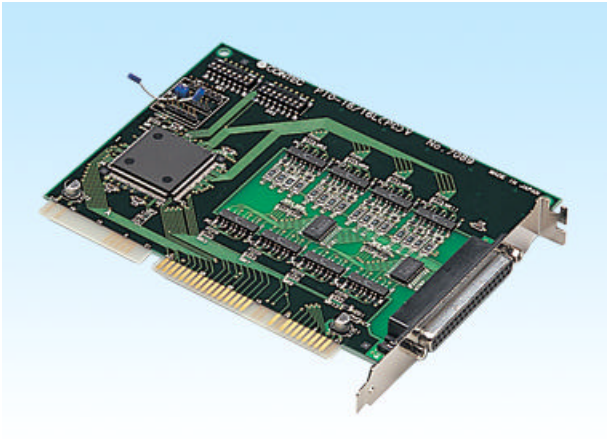
絶縁型デジタル入出力ボード

PIO-16/16L(PC)V



この製品は、絶縁型デジタル入出力を行うPC/ATおよびその互換機用のインターフェイスボードです。このボードは、パソコン本体の拡張スロットまたはI/O拡張ユニットに実装して使用し、これ1枚で最大16点入出力することができます。

特長

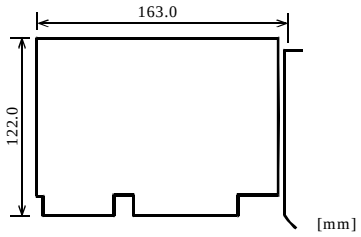
・フォトカプラによる絶縁入出力で耐ノイズ性が向上しています。
・8点を1グループとして2グループ、16点のデジタル信号が入力できます。
・8点を1グループとして2グループ、16点のデジタル信号が出力できます。
・入力信号16点のうち2点を、割り込み入力として最大2レベルの同時割り当てができます。
・出力定格は、1点当たり最大DC35V、100mAの大容量に設計されています。

機能

・入力
このボードは、外部から与えられた最大16点のデジタル信号を、8点単位で構成されるグループで読み出し、ボードを装着したパソコンに送出します。パソコンからのこのボードに対するアクセスは、任意に設定できる2つの入力ポートを介して行います。IN命令の実行によってこれらの入力ポートを読み出すと、その入力ポートに該当するバッファゲートが開かれ、外部装置から与えられているデジタル信号がグループ単位で取り込まれます。このとき、パソコンに送出される信号は負論理となります。また、入力16点のうち2点は、パソコンのユーザー解放割り込み入力として割り当てることができますので、この信号を割り込み要求信号として使用できます。
・出力
このボードは、最大16点のデジタル信号を、8点単位で構成されるグループで外部装置に書き出します。パソコンからのこのボードに対するアクセスは、任意に設定できる2つの出力ポートを介して行います。OUT命令の実行によって出力ポートにデータを書き出すと、その出力ポートに該当するラッチ回路にデータが保持されます。そして、デジタル信号はフォトカプラにより電気的に絶縁された後、トランジスタを通じて、接続されている外部装置にグループ単位で送出されます。このとき、外部装置に送出される信号は負論理となります。また、ラッチ回路上のデータは、再度OUT命令が実行されるまでその状態が保持されます。

仕様

項目			仕様
入力部	入力形式		フォトカプラ絶縁入力(シンク出力対応)
	入力抵抗		3kΩ
	入力ON電流		3.4mA以上
	入力OFF電流		0.16mA以下
	入力信号の点数		16点(2点は割り込みに使用可能) 注)コモンは16点単位で1コモン
	応答時間		1msec以内
出力部	出力形式		フォトカプラ絶縁による電流シンクタイプ出力(負論理)
	定格	出力耐圧	最大　DC35V
		出力電流	最大　100mA(1点当たり)
	出力信号の点数		16点　　注)コモンは16点単位で1コモン
	応答時間		1msec以内
共通部	I/Oアドレス		8ビット×2ポート占有
	割り込み		IRQ3～7、 9～12、14、15のうちいずれか (最大で同時2点まで) High→Lowのエッジで割り込み発生
	外部回路電源		DC12V～24V(±15%) 注)入力1点当たり4mA/12V～8mA/24V
	消費電流		DC5V　50mA(Max.)
	使用条件		0～50℃、20～90%(ただし、結露しないこと)
	信号延長可能距離		50m程度(配線環境による)
	外形寸法(mm)		163.0(L)×122.0(H)
	ボード本体の質量		120g



サポートソフトウェア

目的、開発環境に合わせて当社製サポートソフトウェアのご使用をおすすめします。

■サンプルソフトウェア(解説書にリスト掲載)

●GW-BASICプログラムによる使用方法例

1)0300H、0301Hポートからデータを入力し、ディスプレイに表示。また、そのデータを0300H、0301Hポートの出力します。

●GW-BASICと機械語のリンク例

1)割り込み信号が発生するたびに、割り込み処理プログラムで割り込み回数をカウントし、その回数をディスプレイに表示します。

■ドライバライブラリ API-PAC(W32) (無償ダウンロード)

当社ハードウェアへのコマンドをWindows標準のWin32API関数(DLL)形式で提供するライブラリソフトウェアです。Visual BasicやVisual C/C++などのWin32API関数をサポートしている各種プログラミング言語で、当社ハードウェアの特色を活かした高速なアプリケーションソフトウェアが作成できます。

また、インストールされた診断プログラムにより、ハードウェアの動作確認にも利用することができます。

最新ドライバおよび差分ファイルのダウンロードサービス(http://www.contec.co.jp/apipac/)も行っています。

詳細は、添付CD-ROM内のHelpまたは当社ホームページを参照してください。

＜動作環境＞

主な対応OS	Windows XP、2000、NT、Me、98など、
主な適応言語	Visual C/C++、Borland C++、Visual Basic、Delphi、Builderなど、
その他	ライブラリソフトウェアごとに50MBの空き領域を持つハードディスクが必要

■計測システム開発用ActiveXコンポーネント集 ACX-PAC(W32) (別売)

本製品は、200種類以上の当社計測制御用インターフェイスボード(カード)に対応した計測システム開発支援ツールです。計測用途に特化したソフトウェア部品集で画面表示(各種グラフ、スライダ 他)、解析・演算(FFT、フィルタ他)、ファイル操作(データ保存、読み込み)などのActiveXコンポーネントを満載しています。

アプリケーションプログラムの作成は、ソフトウェア部品を貼り付けて、関連をスクリプトで記述する開発スタイルで、効率よく短期間でできます。また、データロガーや波形解析ツールなどの実例集(アプリケーションプログラム)が収録されていますので、プログラム作成なしでパソコン計測がすぐに始められます。

「実例集」は、ソースコード(Visual Basic 他)付きですので、お客様によるカスタマイズも可能です。

詳細は、当社ホームページ(http://www.contec.co.jp/acxpac/)でご確認ください。

アクセサリ

■アクセサリ(別売)

・圧着端子用端子台(M3) :DTP-3(PC)
・導線用端子台 :DTP-4(PC)
・デジタル入出力用信号モニタアクセサリ :CM-32(PC)E *1
・デジタル入出力用信号モニタアクセサリ:CM-32(PC)
・中継端子台ターミナルユニット :EPD-37A *1
・中継端子台ターミナルユニット :EPD-37 *1

*1 オプションケーブルPCB37PまたはPCB37PSが別途必要

ケーブル・コネクタ

■ケーブル(別売)

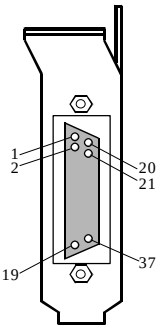
37ピンD-SUB用両端コネクタ付きフラットケーブル
: PCB37P-* (1.5m, 3m, 5m)
37ピンD-SUB用両端コネクタ付きシールドケーブル
: PCB37PS-*P (0.5m, 1.5m, 3m, 5m)
37ピンD-SUB用片端コネクタ付きフラットケーブル
: PCA37P-* (1.5m, 3m, 5m)
37ピンD-SUB用片端コネクタ付きシールドケーブル
: PCA37PS-*P (0.5m, 1.5m, 3m, 5m)

商品構成

・PIO-16/16L(PC)Vボード…1
・解説書…1
・登録カード＆保証書…1
・登録カード返送用封筒…1
・Question用紙…1

インターフェイスコネクタ

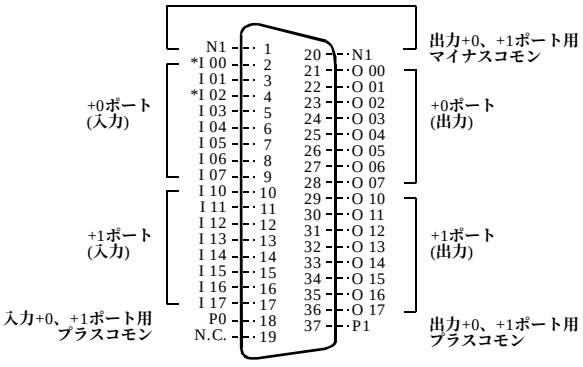
このボードと外部機器との接続は、ボード上のインターフェイスコネクタ(CN1)で行います。



・ 使用コネクタ
37ピンD-SUBコネクタ[F(雌)タイプ]
DCLC-J37SAF-20L9 [JAE製]相当品
ロックナット UNC #4-40 (インチネジ)

・ 適合コネクタ例
17JE-23370-02(D8C) [DDK製、M(雄)タイプ]
FDCD-37P [ヒロセ電機製、M(雄)タイプ]
DC-37P-N [JAE製、M(雄)タイプ]

インターフェイスコネクタの信号配置
本インターフェイスボードと外部装置の接続は、ボードに実装された37ピンのコネクタで行います。

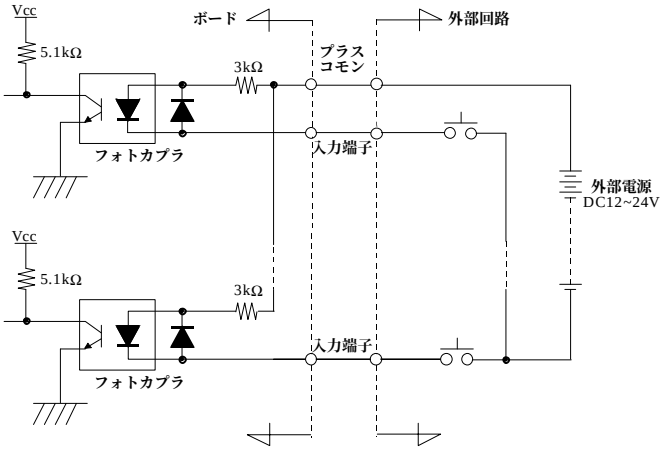


*I00、*I02：外部割り込み端子として使用可

外部入出力回路

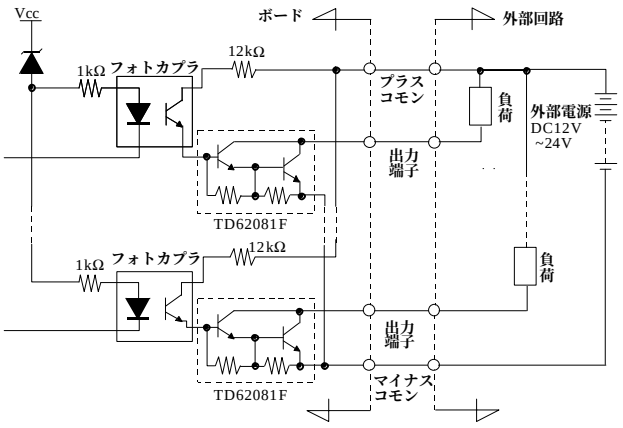
■入力部

このボードのインターフェイス部の入力回路は、図のとおりです。信号入力部は、フォトカプラ絶縁入力(シンク出力対応)になっています。したがって、このボードの入力部を駆動するためには外部電源が必要です。このとき必要となる電源容量は、DC24V時入力1点当たり約8mA(DC12V時には約4mA)です。なお、DC24V使用時に「ON」となる入力点数が多い場合、入力抵抗で若干の発熱を生じます。



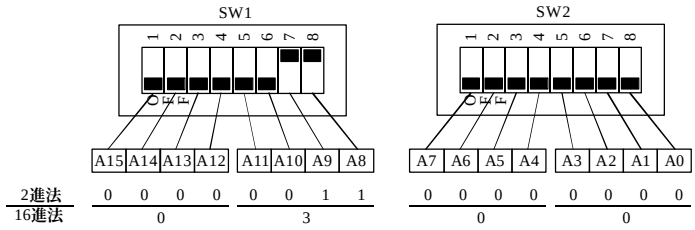
■出力部

このボードのインターフェイス部の出力回路は、図のとおりです。信号出力部はフォトカプラ絶縁による電流シンクタイプ方式になっています。したがって、このボードの出力を駆動するためには外部電源が必要です。出力電流の定格は1点当たり最大100mAです。また、このボードの出力トランジスタには、サージ電圧保護回路が付加されていません。したがって、このボードでリレーやランプなどの誘導負荷を駆動する場合には、負荷側でサージ電圧対策を行ってください。



I/Oアドレスの設定

先頭I/Oアドレスは、ボード上のディップスイッチ(SW1、SW2)で設定します。SW1とSW2の各ビットは先頭I/Oアドレスの上位15ビット(A15～A1)に対応しており、A0は常に「0(OFF)」に固定してください。
SW1とSW2の各ビットのON、OFFは先頭I/Oアドレスを2進数に変換した値を示し、ONが「1」に対応し、OFFが「0」に対応します。



図では先頭I/Oアドレスが0300Hに設定されており、0300H～0301HのI/Oアドレスがこのボードによって占有されます。

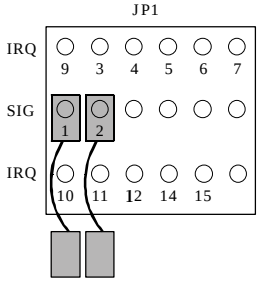
割り込みレベルの設定

このボードでは、入力信号16点のうち2点のデジタル信号を、割り込み要求信号として使用することができます。この信号によって、割り込み要求がパソコンへ出力されますので、パソコンの割り込み機能を利用することができます。割り込み設定はボード上のジャンパ(JP1)で行います。
割り込みを使用しないときは、入力信号を特定レベルに接続しないように、リード線付きショートコネクタで設定してください。
割り込みを使用するときは、ボード上のジャンパ(JP1)で割り込みレベルを設定します。このボードで設定できる割り込みレベルはIRQ3～7、9～12、14、15ですが、パソコン本体および他のボードで使用されていないレベルに設定してください。割り込み要求信号は、入力信号と一対一の対応で最大2つのレベルまで割り当てることができます。

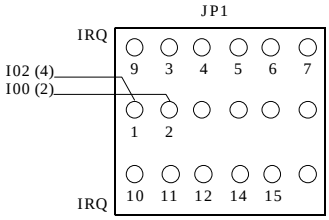
▼注意
- 割り込みを使用する場合、他で使用している割り込みレベルと重複しないように設定してください。
- 電源が入っているパソコン(またはI/O拡張ユニット)にこのボードが実装されているときは、JP1のショートコネクタは抜き差ししないでください。

割り込みレベルは、ボード上のジャンパ(JP1)で設定します。

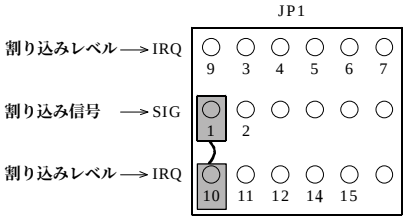
■割り込みを使用しないとき

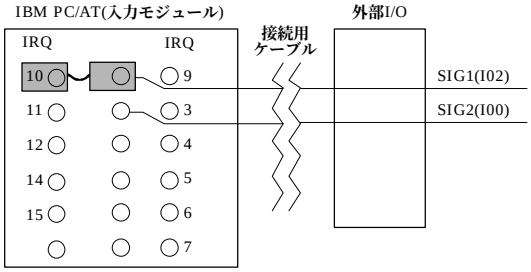


■割り込みを使用するとき
割り込みレベルをショートコネクタで接続してください。設定できる割り込みレベルはIRQ3～7、9～12、14、15です。ただし、XTバス仕様のパソコンではIRQ10～15は使用できません。



(例)インターフェイスボードで「SIG1(I02)」に割り込みをいかし、PC/ATのIRQ10に割り込みをつなぐ場合以下のようになります。





I/0ポートのビット割り付け

■入力ポート

外部装置(リレー接点や半導体スイッチなど)で、外部電源のマイナス側と入力端子を短絡(外部入力接点がON状態)している場合に入力命令が実行されると、内部論理「1」として取り込まれます。

先頭I/Oアドレス	D7	D6	D5	D4	D3	D2	D1	D0
+0H	入力グループ0 (+0ポート)							
	I07 [9]	I06 [8]	I05 [7]	I04 [6]	I03 [5]	I02 [4]	I01 [3]	I00 [2]
+1H	入力グループ1 (+1ポート)							
	I17 [17]	I16 [16]	I15 [15]	I14 [14]	I13 [13]	I12 [12]	I11 [11]	I10 [10]

表中のIxxは信号名を示し、[ ]内はコネクタの端子番号を示します。
また、I00、I02は割り込み信号としても使用できます。

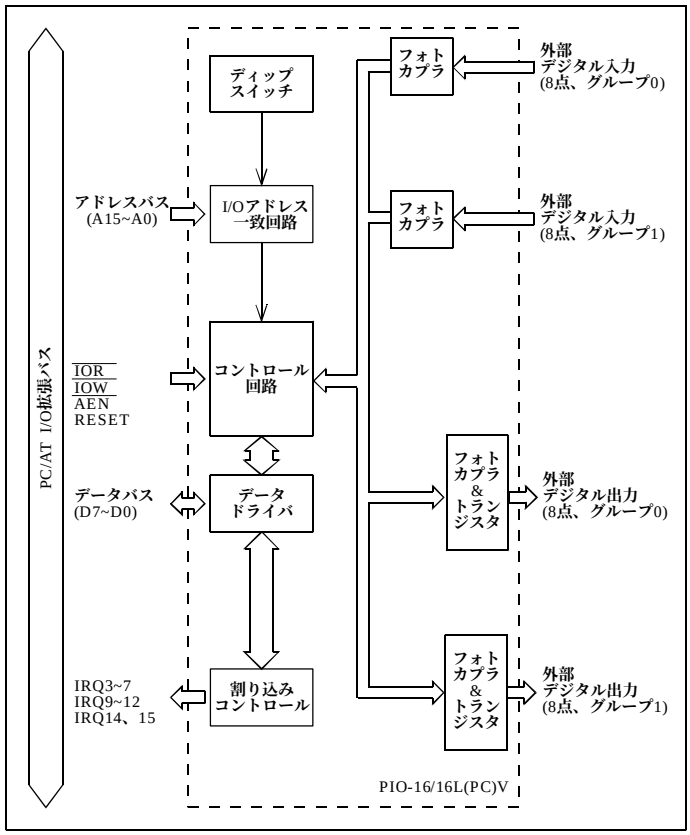
■出力ポート

内部論理「1」を出力(出力命令を実行)することによって、最終段のトランジスタがON状態になります。また、出力されたデータはラッチされ、再度出力命令が実行されるまでその状態を保持します。

先頭I/Oアドレス	D7	D6	D5	D4	D3	D2	D1	D0
+0H	出力グループ0 (+0ポート)							
	O07 [28]	O06 [27]	O05 [26]	O04 [25]	O03 [24]	O02 [23]	O01 [22]	O00 [21]
+1H	出力グループ1 (+1ポート)							
	O17 [36]	O16 [35]	O15 [34]	O14 [33]	O13 [32]	O12 [31]	O11 [30]	O10 [29]

表中のOxxは信号名を示し、[ ]内はコネクタの端子番号を示します。

ブロック図



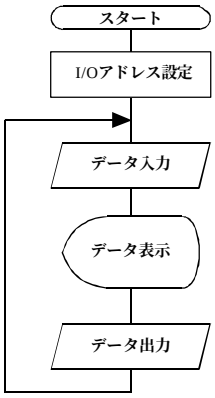
使用例

このボードの使用例として、I00入力端子に接続された外部スイッチのON/OFFに従ってO00出力端子に接続されたLEDを点灯/消灯させるプログラムを以下に示します。記述言語はMicrosoft Cです。スイッチON状態では画面に1が表示されLEDが点灯し、スイッチOFF状態では画面に0が表示されLEDは消灯します。

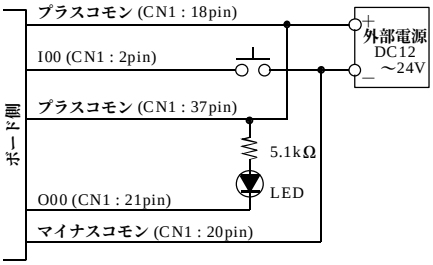
■設定条件

・I/Oアドレス　　：0300H
・割り込み　　：未使用(JP1のショートコネクタをすべて外してください。)

■フローチャート



■接続例



■プログラムリスト

```
#include<stdio.h>

void main(void)
{
  int port;
  int pidata;

  port = 0x300;                    /*I/Oアドレス設定*/

  do {

    pidata = inp ( port );         /*データ入力*/

    printf( "%x\n",pidata );       /*データ表示*/

    outp( port, pidata )           /*データ出力*/

  }while( !kbhit() );
}
```