

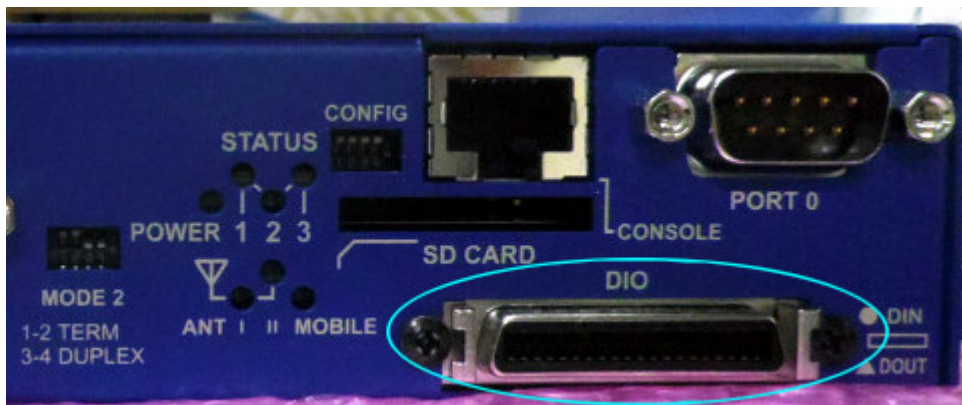
目次

DI/DO を利用する	3
ハードウェア仕様	3
電氣的仕様	3
DI (入力)	3
DO (出力)	3
ピンアサイン	4
等価回路	5
DI (入力)	5
DO (出力)	5
端子台への変換	5
ソフトウェア	6
DI (入力)	6
割込サポート	8
サンプルコード	10
DO (出力)	10

DI/DO を利用する

DI/DO コネクタが実装された下記機種では□DI/DO それぞれ 8ch/8ch が利用できます。

- [MA-E320/D-16](#)
- [MA-E350/FD-16](#)



DI/DO の利用方法を紹介します。

ハードウェア仕様

電氣的仕様

DI (入力)

項目	内容
接点入力	電圧接点入力
ポート数	8ch (DIN A0□A3, B0□B3)
コモン	4ch/コモン
入力電圧	DC12□24V±10% (DC10.8V□26.4V)
入力閾値	ON : DC10V以上, OFF: DC3V以下
入力電流	約2.5mA□5mA
入力インピーダンス	約6kΩ
絶縁方式	フォトカプラ絶縁
絶縁耐圧	DC500V1分間, 外部端子〜内部回路間

DO (出力)

項目	内容
接点出力	オープンコレクタ出力

項目	内容
ポート数	8ch (DOUT A0~A3, B0~B3)
コモン	4ch/コモン
負荷電圧	DC26.4V(最大)
負荷電流	DC50mA(最大)
ON電圧	DC1V以下
OFF時漏洩電流	0.1mA以下
保護機能	過電流保護
絶縁方式	フォトカプラ絶縁
絶縁耐圧	DC500V1分間, 外部端子~内部回路間

ピンアサイン

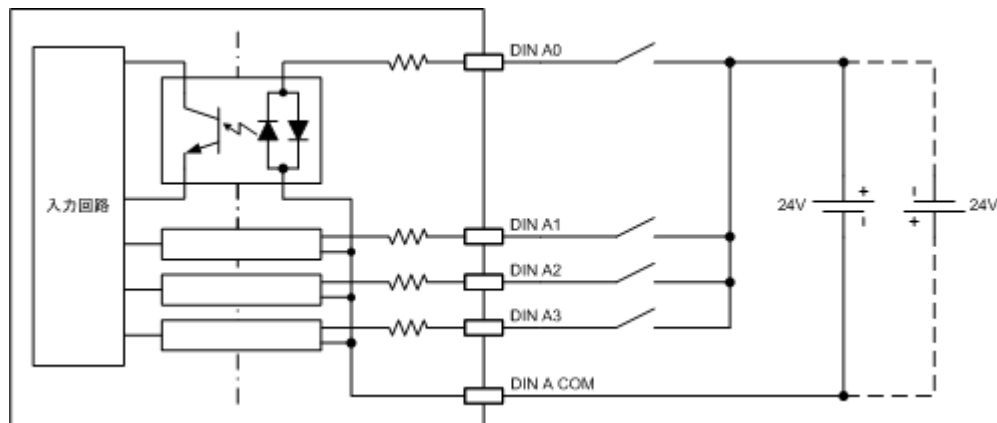


列No.	DO Group	ピン名称	機能	列No.	DI Group	ピン名称	機能	
1	A	NC	未使用	21	A	DIN A COM	コモン A	
2		DOUT A0	DO_0	22		DIN A0	port 0	
3		DOUT A1	DO_1	23		DIN A1	port 1	
4		DOUT A2	DO_2	24		DIN A2	port 2	
5		DOUT A3	DO_3	25	DIN A3	port 3		
6		DOUT A COM	コモン A	26	B	DIN B COM	コモン B	
7		NC	未使用	27		DIN B0	port 4	
8		NC	未使用	28		DIN B1	port 5	
9		NC	未使用	29		DIN B2	port 6	
10		NC	未使用	30		DIN B3	port 7	
11	B	NC	未使用	31		-	NC	未使用
12		DOUT B0	DO_4	32			NC	未使用
13		DOUT B1	DO_5	33	NC		未使用	
14		DOUT B2	DO_6	34	NC		未使用	
15		DOUT B3	DO_7	35	NC		未使用	
16		DOUT B COM	コモン B	36	NC		未使用	
17		NC	未使用	37	NC		未使用	
18		NC	未使用	38	NC		未使用	
19		NC	未使用	39	NC		未使用	
20		NC	未使用	40	NC		未使用	

等価回路

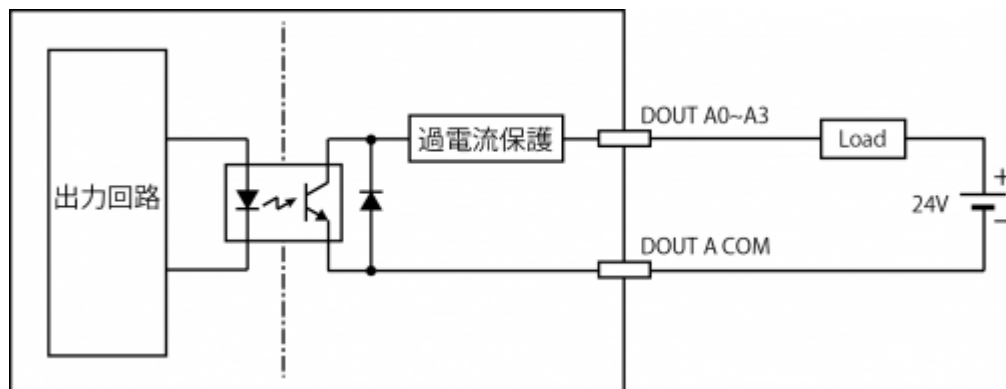
DI (入力)

DI部は、フォトカプラと電流制限抵抗により構成されています。
A,Bグループともに共通コモンとなっています。

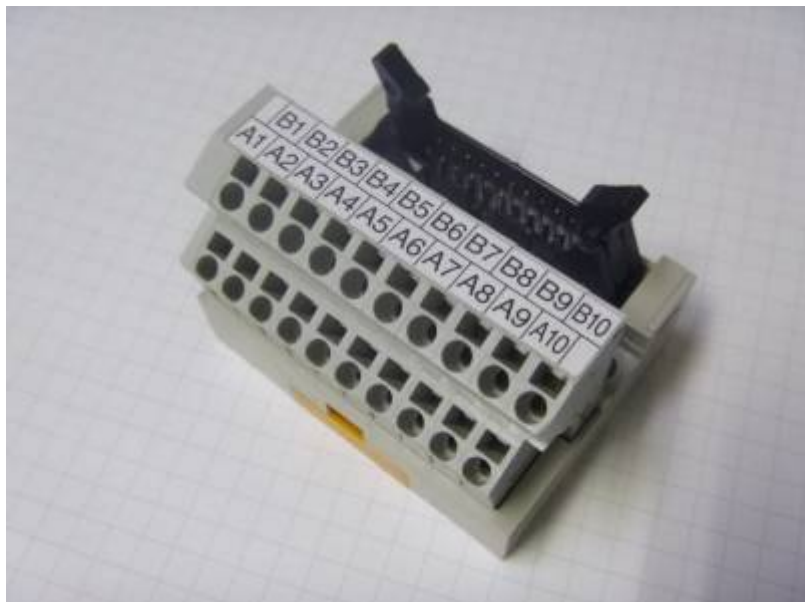


DO (出力)

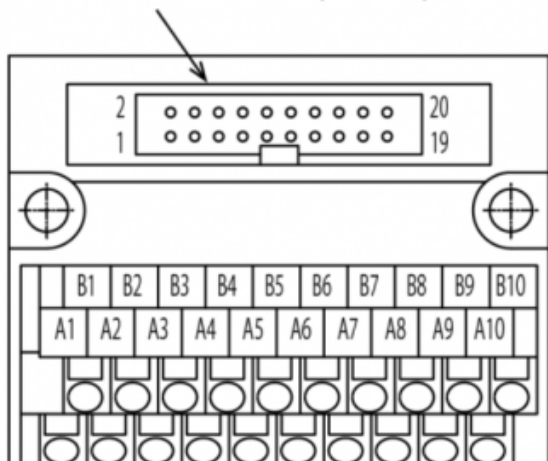
DO部は、フォトカプラと過電流保護素子により構成されています。
A,Bグループともに共通コモンとなっています。



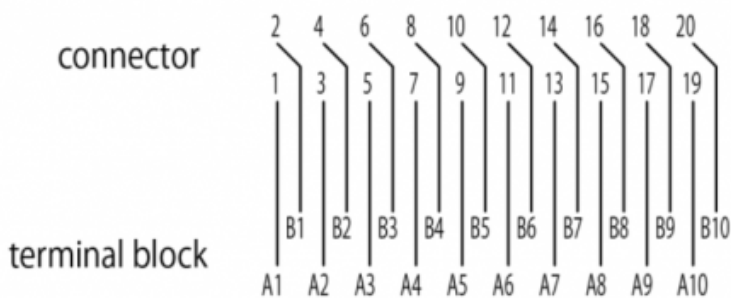
端子台への変換



HIF3BA-20PA-2.54DSA (HIROSE)



PCV5-1H202
Connecting Diagram
Top View



端子台変換器 [PCV5-1H202\(東洋技研製\)](#) を接続することによりDIO 及び AIN ポートをスプリング圧結線方式の端子へ変換することができます。端子台変換器はオプションの DIO ケーブル、もしくは AI ケーブルにより一括接続が可能です。

購入先: [ミスミ](#), [Amazon.co.jp](#)

ソフトウェア

DI (入力)

sysfs I/F でアクセスします。
 参考: [GPIO Sysfs Interface for Userspace](#)

/sys/devices/ocp.3/50000000.gpmc/11000040.gpio/gpio/ 以下の gpiochip でアクセスします。
ボード構成により gpiochip の番号がずれますので、あらかじめ全 DI ポートを export してあります。

DIポート一覧

```
root@plum:~# ls -l /tmp/DI/
total 0
lrwxrwxrwx 1 root root 23 Jun 19 11:40 0 -> /sys/class/gpio/gpio456
lrwxrwxrwx 1 root root 23 Jun 19 11:40 1 -> /sys/class/gpio/gpio457
lrwxrwxrwx 1 root root 23 Jun 19 11:40 2 -> /sys/class/gpio/gpio458
lrwxrwxrwx 1 root root 23 Jun 19 11:40 3 -> /sys/class/gpio/gpio459
lrwxrwxrwx 1 root root 23 Jun 19 11:40 4 -> /sys/class/gpio/gpio460
lrwxrwxrwx 1 root root 23 Jun 19 11:40 5 -> /sys/class/gpio/gpio461
lrwxrwxrwx 1 root root 23 Jun 19 11:40 6 -> /sys/class/gpio/gpio462
lrwxrwxrwx 1 root root 23 Jun 19 11:40 7 -> /sys/class/gpio/gpio463
```

v2.2.0 より GPIO に DeviceTree から命名する機能を入れましたので、このように見えるようになります。

```
user1@plum:~$ ls -l /sys/class/gpio/DI_0*
lrwxrwxrwx 1 root root 0 Aug 31 10:25 /sys/class/gpio/DI_00 ->
../../../../devices/platform/ocp/50000000.gpmc/11000040.gpio/gpiochip13/gpio/DI_00
lrwxrwxrwx 1 root root 0 Aug 31 10:25 /sys/class/gpio/DI_01 ->
../../../../devices/platform/ocp/50000000.gpmc/11000040.gpio/gpiochip13/gpio/DI_01
lrwxrwxrwx 1 root root 0 Aug 31 10:25 /sys/class/gpio/DI_02 ->
../../../../devices/platform/ocp/50000000.gpmc/11000040.gpio/gpiochip13/gpio/DI_02
lrwxrwxrwx 1 root root 0 Aug 31 10:25 /sys/class/gpio/DI_03 ->
../../../../devices/platform/ocp/50000000.gpmc/11000040.gpio/gpiochip13/gpio/DI_03
lrwxrwxrwx 1 root root 0 Aug 31 10:25 /sys/class/gpio/DI_04 ->
../../../../devices/platform/ocp/50000000.gpmc/11000040.gpio/gpiochip13/gpio/DI_04
lrwxrwxrwx 1 root root 0 Aug 31 10:25 /sys/class/gpio/DI_05 ->
../../../../devices/platform/ocp/50000000.gpmc/11000040.gpio/gpiochip13/gpio/DI_05
lrwxrwxrwx 1 root root 0 Aug 31 10:25 /sys/class/gpio/DI_06 ->
../../../../devices/platform/ocp/50000000.gpmc/11000040.gpio/gpiochip13/gpio/DI_06
lrwxrwxrwx 1 root root 0 Aug 31 10:25 /sys/class/gpio/DI_07 ->
../../../../devices/platform/ocp/50000000.gpmc/11000040.gpio/gpiochip13/gpio/DI_07
```

DIポート以下のノード

```
root@plum:~# ls -l /tmp/DI/0/
total 0
-rw-r--r-- 1 root root 4096 Jun 19 11:47 active_low
lrwxrwxrwx 1 root root 0 Jun 19 11:47 device -> ../../../../11000040.gpio
-rw-r--r-- 1 root root 4096 Jun 19 11:47 direction
drwxr-xr-x 2 root root 0 Jun 19 11:47 power
lrwxrwxrwx 1 root root 0 Jun 19 11:47 subsystem ->
../../../../../../../../class/gpio
-rw-r--r-- 1 root root 4096 Jun 19 11:40 uevent
-rw-r--r-- 1 root root 4096 Jun 19 11:47 value
```

value ノードを読むことでDIの値を読むことができます。

```
root@plum:~# cat /tmp/DI/1/value
0
```

割込サポート

v2.6.1α1 よりDI 回路の割込およびフィルタ機能をサポートしました。¹⁾
これにより、下記機能が利用できるようになります。

- チャタリング除去フィルタ (1ms / 5ms / 20ms / なし)
- カウンタ
- 変化待ち (poll() による)

sysfs のエントリが下記のとおり拡張されます。

```
root@plum:/sys/class/gpio/DI_00# ls -l /sys/class/gpio/DI_00/
total 0
-rw-r--r-- 1 root root 4096 Jun 26 15:10 active_low
-rw-r--r-- 1 root root 4096 Jun 29 09:42 counter
-rw-r--r-- 1 root root 4096 Jun 26 15:11 debounce
lrwxrwxrwx 1 root root 0 Jun 26 15:10 device -> ../../../../11000040.gpio
-rw-r--r-- 1 root root 4096 Jun 26 15:10 direction
-rw-r--r-- 1 root root 4096 Jun 29 09:20 edge
drwxr-xr-x 2 root root 0 Jun 26 15:10 power
lrwxrwxrwx 1 root root 0 Jun 26 15:10 subsystem ->
../../../../../../class/gpio
-rw-r--r-- 1 root root 4096 Jun 26 15:10 uevent
-rw-r--r-- 1 root root 4096 Jun 26 15:10 value
root@plum:/sys/class/gpio/DI_00#
```

追加されたエントリは下表のとおりです。

entry	function	note
counter	カウンタ値 R/W	任意の値 ²⁾ にセット可能
debounce	フィルタ設定 R/W	0 / 1 / 5 / 20
edge	割込極性 R/W	none / rising / falling / both

counter

カウンタ値の読み出しおよび設定(クリア)を行います。

読み出し

```
root@plum:~# cat /sys/class/gpio/DI_00/counter
2841
root@plum:~#
```

設定(クリア)

```
root@plum:~# echo 0 > /sys/class/gpio/DI_00/counter
root@plum:~# cat /sys/class/gpio/DI_00/counter
0
root@plum:~#
```

カウンタ利用上の注意

- 後述する “edge” 設定が、“rising” / “falling” / “both” の時に機能します。
- 内部では **32bit unsigned long** で値を保持しています。

debounce (チャタリング除去フィルタ設定)

チャタリング除去フィルタの設定値を読み書きします。

```
root@plum:~# cat /sys/class/gpio/DI_00/debounce
0 ms
root@plum:~# echo 5 > /sys/class/gpio/DI_00/debounce
root@plum:~# cat /sys/class/gpio/DI_00/debounce
5 ms
root@plum:~#
```

設定方法

- 0 / 1 / 5 / 20 を書き込むことで設定します。
- 他の値を書き込んだ場合、書き込んだ値より小さい値に設定されます。

```
root@plum:~# echo 11 > /sys/class/gpio/DI_00/debounce
root@plum:~# cat /sys/class/gpio/DI_00/debounce
5 ms
root@plum:~# echo 100 > /sys/class/gpio/DI_00/debounce
root@plum:~# cat /sys/class/gpio/DI_00/debounce
```

```
20 ms
root@plum:~#
```

edge (割込極性設定)

割込極性を設定します。

rising (立ち上がりトリガ) / falling (立ち下がりトリガ) / both (両エッジトリガ) / none (割込を使用しない) のいずれかを書き込みます。

```
root@plum:~# cat /sys/class/gpio/DI_00/edge
none
root@plum:~# echo rising > /sys/class/gpio/DI_00/edge
root@plum:~# cat /sys/class/gpio/DI_00/edge
rising
root@plum:~#
```

サンプルコード

割込を利用したプログラムのサンプルです。

- [DI 割込を使用したプログラミング](#)

DO (出力)

LEDクラスドライバにマッピングしてあります。

```
root@plum:~# ls -l /sys/class/leds/DO_*
lrwxrwxrwx 1 root root 0 Jun 19 11:41 /sys/class/leds/DO_0 ->
../..../devices/leds_do.7/leds/DO_0
lrwxrwxrwx 1 root root 0 Jun 19 11:41 /sys/class/leds/DO_1 ->
../..../devices/leds_do.7/leds/DO_1
lrwxrwxrwx 1 root root 0 Jun 19 11:41 /sys/class/leds/DO_2 ->
../..../devices/leds_do.7/leds/DO_2
lrwxrwxrwx 1 root root 0 Jun 19 11:41 /sys/class/leds/DO_3 ->
../..../devices/leds_do.7/leds/DO_3
lrwxrwxrwx 1 root root 0 Jun 19 11:41 /sys/class/leds/DO_4 ->
../..../devices/leds_do.7/leds/DO_4
lrwxrwxrwx 1 root root 0 Jun 19 11:41 /sys/class/leds/DO_5 ->
../..../devices/leds_do.7/leds/DO_5
lrwxrwxrwx 1 root root 0 Jun 19 11:41 /sys/class/leds/DO_6 ->
../..../devices/leds_do.7/leds/DO_6
lrwxrwxrwx 1 root root 0 Jun 19 11:41 /sys/class/leds/DO_7 ->
```

```
../../../../devices/leds_do.7/leds/D0_7
```

各 DO の下の構成はこのようなになっています。

```
root@plum:~# ls -l /sys/class/leds/D0_1/
total 0
-rw-r--r-- 1 root root 4096 Jun 19 11:50 brightness
lrwxrwxrwx 1 root root 0 Jun 19 11:50 device -> ../../../../leds_do.7
-r--r--r-- 1 root root 4096 Jun 19 11:50 max_brightness
drwxr-xr-x 2 root root 0 Jun 19 11:50 power
lrwxrwxrwx 1 root root 0 Jun 19 11:50 subsystem -> ../../../../class/leds
-rw-r--r-- 1 root root 4096 Jun 19 11:50 trigger
-rw-r--r-- 1 root root 4096 Jun 19 11:40 uevent
```

sysfs LED クラスドライバと同様に使用することができます。

1)

MA-E350/LAD-20 の 8bit ハードウェアカウンタ機能が実装された 前半 4channel 部の設定は [DI\(ハードウェアカウンタ機能付き\)を利用する](#) を参考。

2)

32bit unsigned long

From:

<https://ma-tech.centurysys.jp/> - MA-X/MA-S/MA-E/IP-K Developers' Wiki

Permanent link:

https://ma-tech.centurysys.jp/doku.php?id=mae3xx_ope:use_di_do:start

Last update: **2024/02/28 11:06**

