

IDAQ-3730

TTL 数字量I/O模块

用 户 手 册

一、概述

IDAQ-3730 是一款 32 路 TTL 数字量输入和 32 路 TTL 数字量输出的 PC/104 模块，标准的 PC/104 模块的安装规格，适于和标准的 PC/104 CPU 模块连接实现数字量信号的采集和控制。专为工业应用场合而设计，完全达到工业级工作温度的要求。

二、规格参数

☆输出部分

- 通道数：32 路
- TTL 兼容电平输出
- 输出低电平：最高 0.5V
- 输出高电平：最低 2.4V
- 高电平输出电流：最大—15mA(source)
- 低电平输出电流：最大 24mA(sink)
- 最高输出频率：300~400KHZ

☆输入部分

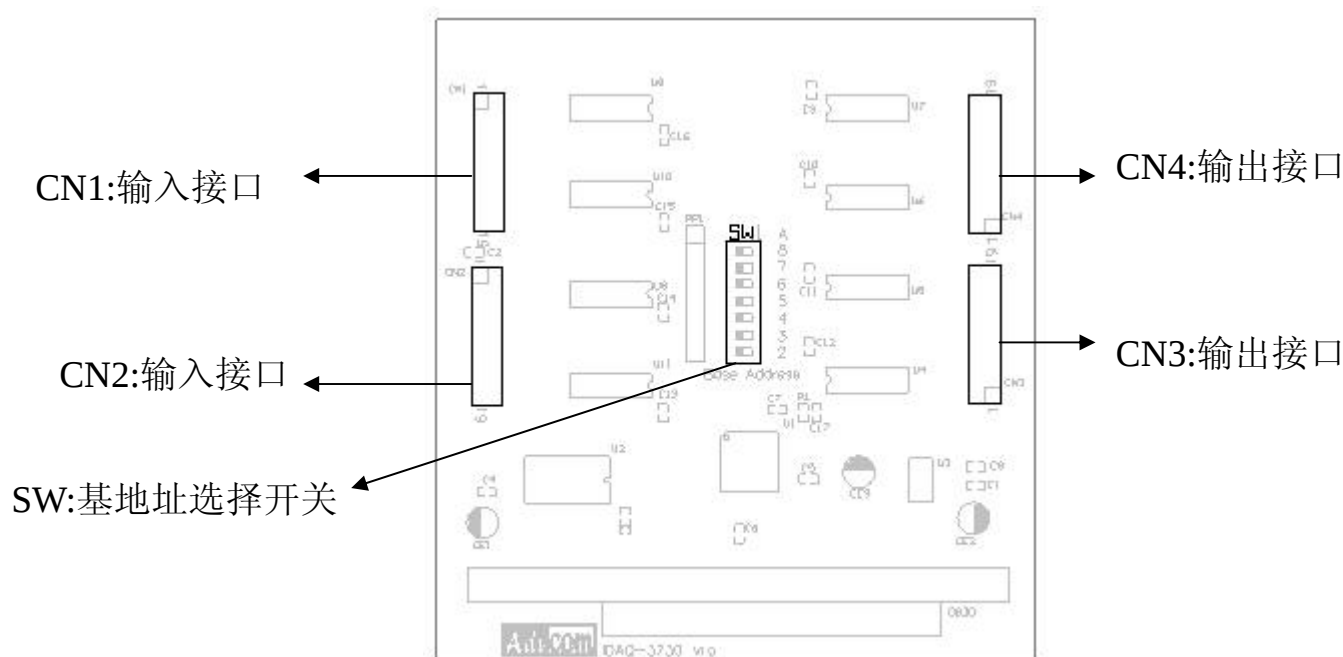
- 通道数：32 路
- TTL 兼容电平输入
- 逻辑高电平：2V~5V
- 逻辑低电平：0~0.8V
- 高电平输入电流：20uA
- 低电平输入电流：—0.2mA
- 最高输入频率：300~400KHZ

☆通用部分

- I/O 基址：4 个连续地址空间
- 储存温度：—25℃~80℃
- 工作温度：—25℃~80℃
- 湿度：5~95%，无凝结
- 电源功耗：+5V@450mA 典型值

●尺寸：90X191MM

三、硬件布局

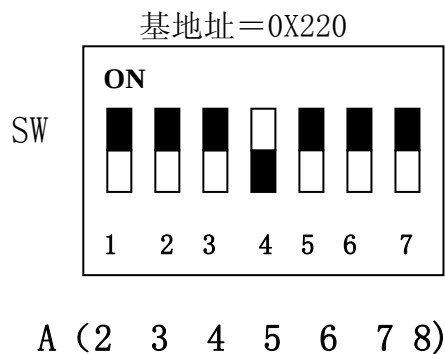


四、基地址设置

IDAQ-3730 要求 4 个连续 I/O 地址空间，且有以下条件限制。

1. 地址的范围必须要在 HEX200—HEX3FF 之间。
2. 地址不能与任何的嵌入式 CPU 模块预设的 I/O 地址矛盾，见附录。
3. 地址不能与嵌入式 CPU 模块的其他模块基地址矛盾。在安装前请检查确认。

IDAQ-3730 的基地址由一个 7 位的 DIP 开关 SW 选择（SW 的位置见硬件布局部分）。基地址的缺省设置为 HEX220。所有可能的基地址组合都在下表中给出。如果 HEX220 被占用，你可以改变预设的基地址。



缺省的基地址设置

可能的基地址组合表

I/O 口	1	2	3	4	5	6	7
地址 (HEX)	A2	A3	A4	A5	A6	A7	A8
200-203	ON	ON	ON	ON	ON	ON	ON
204-207	OFF	ON	ON	ON	ON	ON	ON
208-20B	ON	OFF	ON	ON	ON	ON	ON
...	...	...	...	...	...	...	...
220-223 (default)	ON 0	ON 0	ON 0	OFF 1	ON 0	ON 0	ON 0
...	...	...	...	...	...	...	...
300-303	ON	ON	ON	ON	ON	ON	OFF
304-307	OFF	ON	ON	ON	ON	ON	OFF
...	...	...	...	...	...	...	...
3FC-3FF	OFF	OFF	OFF	OFF	OFF	OFF	OFF

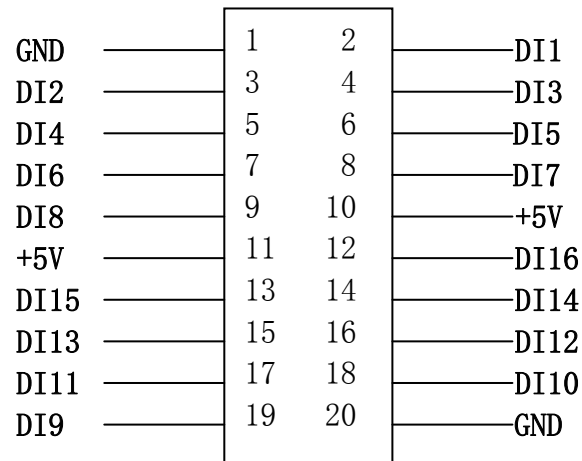
A0...A8 对应于 PC 地址线。

A9 固定为 1。

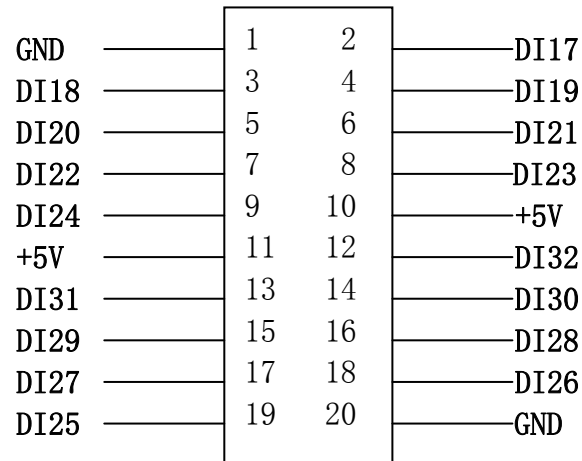
五、信号连接

IDAQ-3760 有 4 个用于数字量输入输出的 20PIN 针脚间距 2MM 的接口。

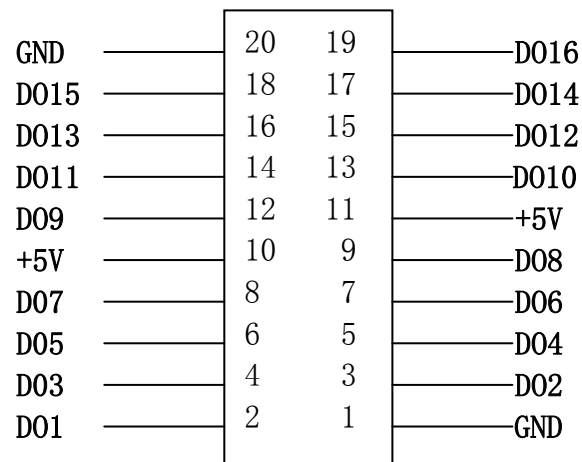
1. CN1 输入接口



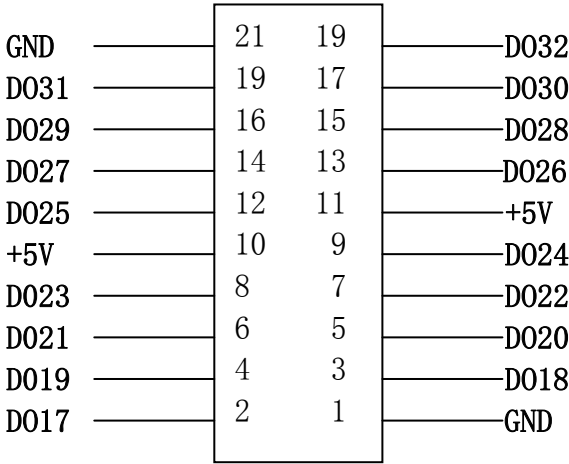
2. CN2 输入接口



3. CN3 输出接口



4. CN4 输出接口



说明：D0x:数字量信号输出通道 x。
DIx:数字量信号输入通道 x。
GND:数字量输入信号的公共端。
+5V:该板直流电源。

六、寄存器结构和格式

本节描述 IDAQ-3730 的寄存器格式和结构。这些信息对程序员利用低级程序语言处理板卡非常有用。

1. IDAQ-3730 要求的 I/O 地址空间中的 4 个连续的地址，下表给出了个每个寄存器相应于基地址的 I/O 地址，以及个寄存器的功能。

I/O 地址	读	写
Base+0	数字量输入通道 0-7	数字量输出通道 0-7
Base+1	数字量输入通道 8-15	数字量输出通道 8-15
Base+2	数字量输入通道 16-23	数字量输出通道 16-23
Base+3	数字量输入通道 24-31	数字量输出通道 24-31

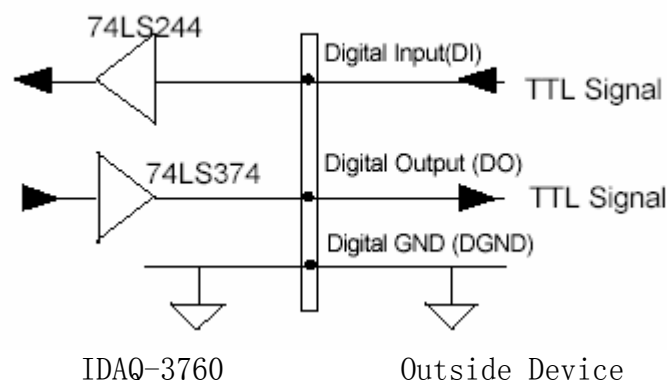
2. 读数据格式

位	7	6	5	4	3	2	1	0
Base+0	DI8	DI7	DI6	DI5	DI4	DI3	DI2	DI1
Base+1	DI16	DI15	DI14	DI13	DI12	DI11	DI10	DI9
Base+2	DI24	DI23	DI22	DI21	DI20	DI19	DI18	DI17
Base+3	DI32	DI31	DI30	DI29	DI28	DI27	DI26	DI25

3. 写数据格式

位	7	6	5	4	3	2	1	0
Base+0	D08	D07	D06	D05	D04	D03	D02	D01
Base+1	D016	D015	D014	D013	D012	D011	D010	D09
Base+2	D024	D023	D022	D021	D020	D019	D018	D017
Base+3	D032	D031	D030	D029	D028	D027	D026	D025

七、信号连接示意图



八、编程范例

选择哪种编程语言取决于用户的熟悉程度和系统的要求。不论哪种语言，用户必须理解访问的 I/O 卡的 I/O 指令语法。下面介绍了经常使用的编程语言的语法结构。每个都给出了写（输出）口指令和读（输入）口指令。例子中，I/O 卡的基地址被假设为 HEX300，要访问的寄存器口为基地址+2。

汇编

写一个输出口：

```
MOV DX, 302
MOV AL, 2F
OUT DX, AL
```

读一个输入口：

```
MOV DX, 222
IN AL, DX
```

BASIC 语言

写一个输出口：

```
10BASE=&H300
20 VALUE% = &H2F
30OUT (BASE+2), VALUE %
或
100UT (&H302), &H2F
```

读一个输入口

```
10BASE=&H300
20VALUE=INP (BASE+2)
```

或

```
IOVALUE=INP (&H302 )
```

C 语言 (Borland C++)

写一个输出口:

```
#define BASE0x300
unsigned int Value=0x2F;
outportb (BASE+2 , Value);
or
outportb (0x302, 0x2F);
```

读一个输入口:

```
#define BASE0x300
unsigned intValue;
Value = inpportb (BASE+2);
或
Value = inportb (0x302);
```

C 语言 (MicroSoft C)

写一个输出口:

```
#define BASE0x300
unsigned intValue=0x2F;
outp (BASE+2 , Value) ;
或
outp (0x302 , 0x2F);
```

读一个输入口:

```
#define BASE0x300
unsigned intValue;
Value = inp (BASE+2);
或
Value = inp (0x302);
```

附录：I/O 口地址映射

I/O 地址	设备
000-01F	DMA 控制器
020-03F	中断控制器
040-05F	计时器
060-06F	键盘
070-07F	实时时钟
080-09F	DMA 页面寄存器
0A0-0BF	中断控制器
0C0-0DF	DMA 控制器
0F0-0FF	数值数据处理
100-1EF	不可用
1F0-1F8	固定盘
200-207	游戏 I/O
278-27F	并行打印口 2 (LPT2)
2F8-2FF	串口 2 (COM2)
300-31F	原型卡
360-36F	保留
378-37F	并行打印口 1 (LPT1)
3B0-3BF	单色显示
3C0-3CF	保留
3D0-3DF	彩色图形显示
3F0-3F7	磁盘控制器
3F8-3FF	串口 1 (COM1)