

FSC-1813L2NA

PICMG 1.3 全长主板带 VGA/LAN

版本：A2

## 声明

除列明随产品配置的配件外，本手册包含的内容并不代表本公司的承诺，本公司保留对此手册更改的权利，且不另行通知。对于任何因安装、使用不当而导致的直接、间接、有意或无意的损坏及隐患概不负责。

订购产品前，请向经销商详细了解产品性能是否符合您的需求。

EVOC是研祥智能科技股份有限公司的注册商标。本手册所涉及到的其他商标，其所有权为相应的产品厂家所拥有。

本手册内容受版权保护，版权所有。未经许可，不得以机械的、电子的或其它任何方式进行复制。

# 安全使用小常识

---

1. 产品使用前，务必仔细阅读产品说明书；
2. 对未准备安装的板卡，应将其保存在防静电保护袋中；
3. 在从防静电保护袋中拿出板卡前，应将手先置于接地金属物体上一会儿（比如 10 秒钟），以释放身体及手中的静电；
4. 在拿板卡时，需佩戴静电保护手套，并且应该养成只触及边缘部分的习惯；
5. 为避免人体被电击或产品被损坏，在每次对主板、板卡进行拔插或重新配置时，须先关闭交流电源或将交流电源线从电源插座中拔掉；
6. 在需对板卡或整机进行搬动前，务必先将交流电源线从电源插座中拔掉；
7. 对整机产品，需增加 / 减少板卡时，务必先拔掉交流电源；
8. 当您需连接或拔除任何设备前，须确定所有的电源线事先已被拔掉；
9. 为避免频繁开关机对产品造成不必要的损伤，关机后，应至少等待 30 秒后再开机。

# 目 录

第一章 产品介绍.....	1
简介.....	1
订购信息.....	1
环境与机械尺寸.....	2
微处理器.....	2
芯片组.....	2
系统存储器.....	2
IDE 功能.....	2
USB 功能.....	2
显示功能.....	3
扩充插槽.....	3
SATA 功能.....	3
网络功能.....	3
外部 I/O 接口功能.....	3
Super I/O 看门狗定时器.....	3
硬件监测.....	3
第二章 安装说明.....	4
产品外形尺寸图.....	4
位置接口示意图.....	5
跳线功能设置.....	6
系统内存安装.....	6
USB 功能.....	6
IDE 接口.....	7

并口与串口.....	8
键盘与鼠标接口.....	9
音频接口.....	9
显示接口.....	9
软驱接口.....	10
网络接口.....	11
状态指示灯控制接口.....	11
电源接口.....	12
风扇接口.....	12
数字 IO 接口 .....	13
数字 IO 编程指引 .....	13
第三章 BIOS 功能简介.....	18
附录.....	19
Watchdog 编程指引.....	19
I/O 口地址映射表 .....	22
IRQ 中断分配表.....	23

## 第一章

### 产品介绍

#### 简介

FSC-1813L2NA是一款同时拥有高性能图形能力，高速运算能力和高速I/O处理能力的最新工业标准PICMG 1.3全长主板。

FSC-1813L2NA采用Intel最新LGA775微处理器，不仅支持新一代总线PCI Express，同时也继续支持PCI 2.3总线。FSC-1813L2NA前端系统总线最高可支持1.06GHz，再加上双通道DDR2 667/800内存技术，使得系统吞吐量得到大大提高。FSC-1813L2NA的图形控制器集成了最新的Intel GMA3100显示核心。南北桥之间使用DMI总线连接，其带宽高达2GB/s（原HUB Link只有266MB/s），大大提高了I/O吞吐量。FSC-1813L2NA配置了两个1Gb以太网接口，八位数字I/O接口。

配合最新的PICMG 1.3标准底板，FSC-1813L2NA可扩充4个32位标准PCI插槽，4个PCI Express x1插槽，1个PCI Express ×16插槽，4个串行ATA接口，8个USB 2.0接口，提供了强大的系统I/O扩充能力。

#### 订购信息

型号	描述
FSC-1813L2NA	PICMG 1.3全长主板带VGA/双千兆LAN接口

## 环境与机械尺寸

- 工作环境：  
温度：0~60°C  
湿度：5%~95%（非凝结状态）
- 贮存温度：  
温度：-25°C~75°C  
湿度：5%~95%（非凝结状态）
- 尺寸：338.58mm×126.39mm

## 微处理器

LGA775 封装 Intel® Core™2 Duo/Quad CPU，Pentium D, Pentium 4 and Celeron 系列，要求前端总线频率 800MHz/1066MHz。

## 芯片组

Intel G31 + ICH7R，800/1066MHz 前端总线。

## 系统存储器

两条 240-pin DDR2 内存条，可支持内存容量最大 4GB，支持内存频率 667MHz 和 800MHz。

## IDE 功能

一个 ATA100/66/33 标准 IDE 扩展接口，支持两个 IDE 设备。

## USB 功能

提供八个 USB2.0 高速接口扩展连接（通过 PICMG 1.3 底板扩充四个）。

## 显示功能

最新的 Intel GMA3100 显示核心，支持 DirectX9.0C。

## 扩充插槽

四个 32 位 PCI 插槽，1 个 PCI Express ×16 插槽，四个 PCI Express ×1 插槽（通过 PICMG 1.3 底板扩充）

## SATA 功能

本主板带有四个 SATA 接口，可用来连接大容量存储设备。支持 RAID0, 1, 5, 10。

## 网络功能

主板集成了两个 10/100/1000Mb 以太网控制器，为您提供高速稳定的网络平台选择。

## 外部 I/O 接口功能

- 提供 COM1 和 COM2 两组 COM 口扩展连接；
- 提供一个并口扩展连接；
- 提供一个标准 PS/2 键盘鼠标接口扩展连接。

## Super I/O 看门狗定时器

- 1-255 秒或 1-255 分，510 级；
- 可编程时间到中断；
- 超时事件复位系统。

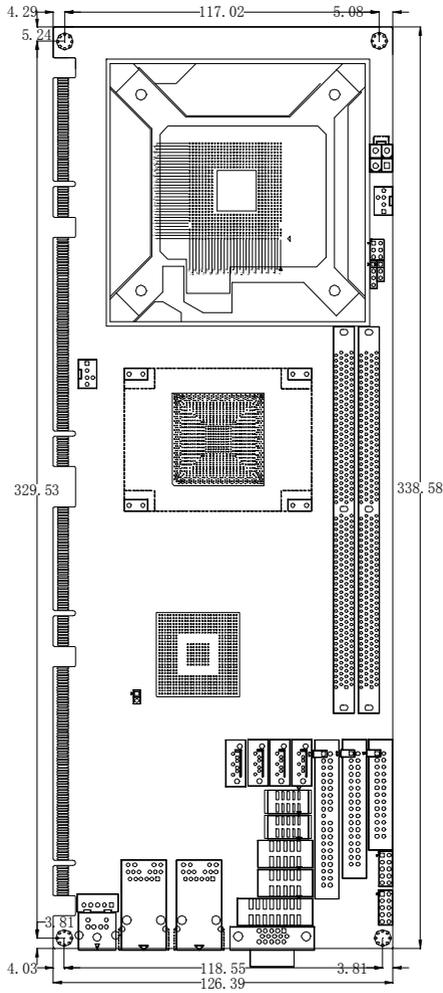
## 硬件监测

对工作电压，CPU 和系统机箱温度检测。

## 第二章

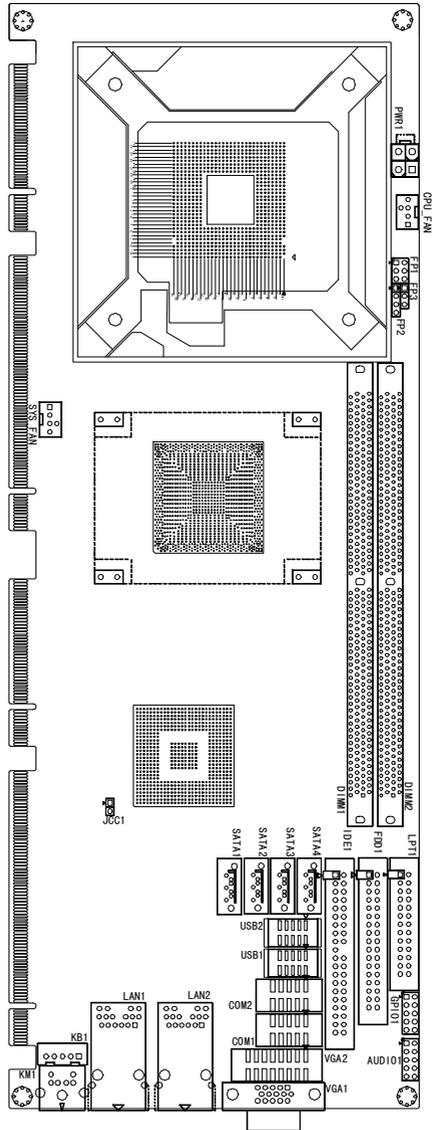
## 安装说明

### 产品外形尺寸图



单位：mm

位置接口示意图



## 跳线功能设置

### 1) JCC1:CMOS内容清除/保持设置

CMOS由板上钮扣电池供电。清CMOS会导致永久性消除以前系统配置并将其设为原始（工厂设置）系统设置。其步骤：(1)关计算机，断开电源；(2)瞬间短接JP1插针；(3)开计算机；(4)启动时按住DEL键进入BIOS设置，重载最优缺省值；(5)保存并退出设置。



JCC1

设置	功能
开路	[ 1-2 ] (正常工作状态, 默认设置)
瞬间短路	[1-2] (清除 CMOS 内容, 所有 BIOS 设置恢复成出厂值)

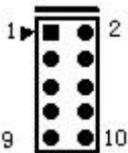
## 系统内存安装

主板配有二条DDR II (Double Data Rate) DIMM (Dual Inline Memory Modules) 240pin内存插槽(图示标识为DIMM1及DIMM2)。

安装内存条时，要注意以下几点：

- 安装时，先对准内存 DIMM 条的缺口和 DIMM 插槽的缺口后再用力插到位。
- 可使用符合 Intel 1.8V DDR II 667/800 规格的 DDR II 内存，单槽最大内存容量达 2GB。
- 最好选择带 SPD（内存自动识别功能）的 DIMM 内存条，以保证内存条工作稳定。

## USB 功能



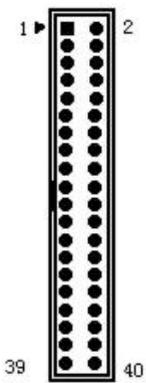
USB1/USB2

管脚	信号名称	管脚	信号名称
1	+5V	2	+5V
3	USB Data-	4	USB Data-
5	USB Data+	6	USB Data+
7	GND	8	GND
9	空	10	Shielded GND

## IDE 接口

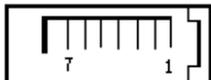
本单板电脑提供一个40针IDE接口（IDE1），安装IDE设备时，需注意：

- IDE 接口可以连接两台 IDE 设备：一个为主设备（Master），一个为从设备（Slave）。设备的连接方法是：主设备接在电缆的末端，从设备接在电缆的中间。
- 连接使用 Ultra66/100 的硬盘时，必须使用 80 线的专用扁平电缆（IDE 电缆有红色标示的为第一脚）。



管脚	信号名称	管脚	信号名称
1	RESET#	2	GND
3	D7	4	D8
5	D6	6	D9
7	D5	8	D10
9	D4	10	D11
11	D3	12	D12
13	D2	14	D13
15	D1	16	D14
17	D0	18	D15
19	GND	20	Key
21	DREQ	22	GND
23	IOW#	24	GND
25	IOR#	26	GND
27	IORDY	28	GND
29	DACK#	30	GND
31	IRQ	32	NC
33	DA1	34	ATA66_DET
35	DA0	36	DA2
37	CS1#	38	CS3#
39	LED#	40	GND

另外本CPU卡还提供4个串行ATA接口

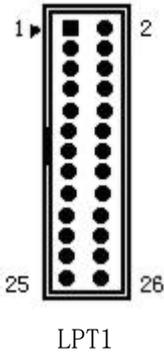


管脚	信号名称	管脚	信号名称
1	GND	5	RX-
2	TX+	6	RX+
3	TX-	7	GND
4	GND		

## 并口与串口

### (1) 并口

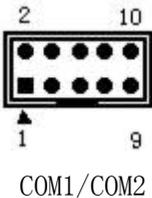
标准的26针并行接口，可依据您的需求用来连接并行接口外设。



管脚	信号名称	管脚	信号名称
1	STB#	2	AFD#
3	PD0	4	ERR#
5	PD1	6	INIT#
7	PD2	8	SLIN#
9	PD3	10	GND
11	PD4	12	GND
13	PD5	14	GND
15	PD6	16	GND
17	PD7	18	GND
19	ACK#	20	GND
21	BUSY	22	GND
23	PE	24	GND
25	SLCT	26	NC

### 串口

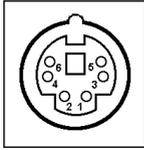
COM1, COM2可以连接具有RS-232标准接口的鼠标、调制解调器、数码相机等设备。



管脚	信号名称
1	DCD, 数据载波检测
2	RXD, 接收数据
3	TXD, 传输数据
4	DTR, 数据终端准备好
5	GND, 地
6	DSR, 数据设备准备好
7	RTS, 请求发送
8	CTS, 清发送
9	RI, 振铃指示
10	NC

## 键盘与鼠标接口

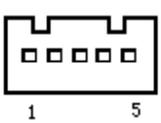
### 1) 键盘与鼠标接口



KM1

管脚	信号名称
1	Keyboard data
2	Mouse data
3	GND
4	+5V
5	Keyboard clock
6	Mouse clock

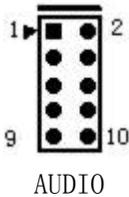
### 2) 键盘接口



KB1

管脚	信号名称	管脚	信号名称
1	键盘时钟	4	地
2	键盘数据	5	+5V
3	无连接		

## 音频接口

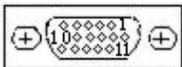


AUDIO

管脚	信号名称	管脚	信号名称
1	LOUT_R	2	LOUT_L
3	GND_AUDIO	4	GND_AUDIO
5	LIN_R	6	LIN_L
7	GND_AUDIO	8	GND_AUDIO
9	MIC_IN	10	MIC_P

## 显示接口

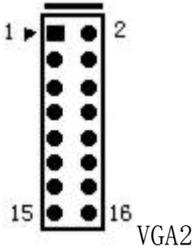
15芯D型VGA显示器插座 VGA1。



VGA1

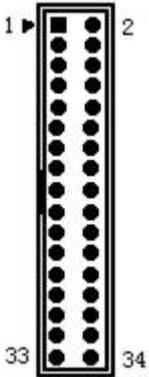
管脚	信号名称	管脚	信号名称
1	Red	2	Green
3	Blue	4	NC
5	GND	6	GND
7	GND	8	GND
9	+5V	10	GND
11	NC	12	DDCDATA
13	HSYNC	14	VSYNC
15	DDCCLK		

VGA1附加一pin header VGA2可将VGA通过排线外拉



管脚	信号名称	管脚	信号名称
1	Red	2	GND
3	NC	4	Green
5	GND	6	DCCDATA
7	Blue	8	GND
9	HSYNC	10	NC
11	+5V	12	VSYNC
13	GND	14	GND
15	DDCCLK	16	NC

### 软驱接口



管脚	信号名称	管脚	信号名称
1	GND	2	DENSEL
3	GND	4	NC
5	Key	6	DRATE
7	GND	8	INDEX#
9	GND	10	MOA#
11	GND	12	DSB#
13	GND	14	DSA#
15	GND	16	MOB#
17	GND	18	DIR#
19	GND	20	STEP#
21	GND	22	WD#
23	GND	24	WE#
25	GND	26	TRACKO#
27	GND	28	WP#
29	NC	30	RDATA#
31	GND	32	HDSEL#
33	NC	34	DSKCHG#

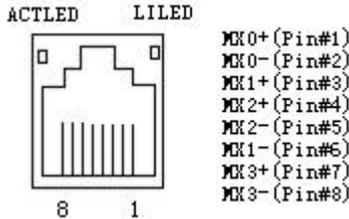
## 网络接口

主板上有两个10/100/1000Mbps以太网接口(LAN1, LAN2), LILED和ACTLED是以太网接口两边的LED指示灯, 它们显示着LAN的活动和传输状态。请参考以下每一个LED的状态描述:

MX[0: 3]+, MX[0: 3]-: 正/负 数据信号。

ACTLED: 网络活动状态灯。

LILED: 网络链路状态灯。

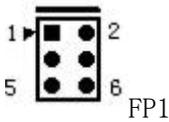


ACTLED (单色: 绿色灯)	网络活动 指示状态	LILED (双色: 黄/绿灯)	网络速度 指示状态
闪烁	正在收发 数据	绿色	1000Mbps
		黄色	100Mbps
		灭	10Mbps

## 状态指示灯控制接口

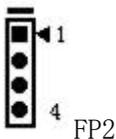
FP1, FP2, FP3用于连接至机箱前面板上所设的功能按钮或指示灯。

### ATX电源开关及硬盘指示灯接口



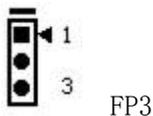
管脚	信号名称	管脚	信号名称
1	Power Button+	2	GND
3	GND	4	RESET
5	IDE LED -	6	IDE LED +

### 扬声器输出接口



管脚	信号名称
1	Speaker out
2	NC
3	GND
4	+5V

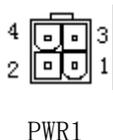
## 电源指示灯接口



管脚	信号名称
1	Power LED +
2	NC
3	GND

## 电源接口

### 1) +12V电源接口

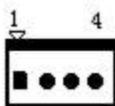


管脚	信号名称
1	GND
2	GND
3	+12V
4	+12V

## 风扇接口

本CPU卡提供两组标准风扇插座, 使用风扇插座时要注意以下三点:

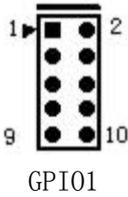
- 风扇电流不大于 350 毫安 (4.2 瓦, 12 伏特)。
- 请确认风扇接线和本插座的接线是否相符。电源线 (通常为红色) 在中间位置。另外就是地线 (通常为黑色) 和风扇转速输出脉冲信号线 (其它颜色)。有些风扇没有转速检测, 但该引线却有高达 12V 的输出, 会损坏 CPU 卡, 这是非标准接线。建议使用带转速检测风扇。
- 将风扇气流调整成能将热量排出的方向。



CPU\_FAN/ SYS\_FAN

管脚	信号名称
1	GND
2	VCC12
3	风扇转速脉冲
4	风扇控制

## 数字 IO 接口



管脚	信号名称	管脚	信号名称
1	DIO_IN0	2	DIO_OUT0
3	DIO_IN1	4	DIO_OUT1
5	DIO_IN2	6	DIO_OUT2
7	DIO_IN3	8	DIO_OUT3
9	GND	10	GND

## 数字 IO 编程指引

### 四位数字量输入/输出的说明

### 四位数字量输入/输出的使用

为了给用户更大的灵活性这四位数字输入/输出信号是可编程的因此在使用数字输入/输出之前必须选择定义：

- 数字量I/O的基地址；
- 定义哪一位为输入, 哪一位为输出。

```
#include <stdio.h>
#include <stdlib.h>
#include <bios.h>
#include <conio.h>
#include <dos.h>
#include <io.h>
```

```
unsigned long _pascal in_32(unsigned int io_port)
{
    unsigned long ReturnValue;
    _asm{
        mov dx, io_port
```

```
                in  eax, dx
                mov  ReturnValue, eax
            }
    return ReturnValue;

}

void _pascal out_32(unsigned int io_port, unsigned long
io_data)
{
    _asm{
        mov dx, io_port
        mov eax, io_data
        out dx,    eax
    }
}

long read_pci(long addr)
{
    long  pci_data;
    out_32( 0xCF8, addr );
    pci_data = in_32( 0xCFC );
    return(pci_data);
}

long Write_pci(long addr, long data)
{
    long  pci_data;
    out_32( 0xCF8, addr );
    out_32( 0xcfc, data );
}
```

```
        return(pci_data);
    }

    int main()
    {
        long addr_pci= 0x8000F848;
        long io_base=read_pci(addr_pci) & 0x0000ffff;//
取基地址

        unsigned int  addr_statue=io_base+0x1a;
        outportb(addr_statue, 0x00);
        addr_statue=io_base+0x1b;
        outportb(addr_statue, 0x00);

//BLINK

        //select as GPIO

        unsigned int  addr_select=io_base+0x02;
        unsigned int  get_1x=inportb(addr_select)|0x05;
        outportb(addr_select, get_1x);          //16, 18
set 1

        addr_select=io_base+0x03;
        unsigned int  get_2x=inportb(addr_select)|0x1c;
        outportb(addr_select, get_2x);          //26,
27, 28 set 1

        addr_select=io_base+0x30;
        unsigned int  get_3x=inportb(addr_select)|0xc2;
```

```
        outportb(addr_select, get_3x);
//33, 38, 39 set 1

//select as INPUT or OUTPUT

        unsigned int  addr_select2=io_base+0x06;
        unsigned int  get_0=inportb(addr_select2)|0x04;
        get_0=get_0&0xfe      ;
        outportb(addr_select2, get_0);          //18
input, 16 output

        addr_select2=io_base+0x07;
        unsigned int  get_2=inportb(addr_select2)&0xe3;
        outportb(addr_select2, get_2);          //26, 27, 28
output

        addr_select2=io_base+0x34;
        unsigned int  get_3=inportb(addr_select2)|0xc2;
        outportb(addr_select2, get_3);          //33, 38, 39
input

// OUTPUT 0 or 1

        unsigned int  addr_o1=io_base+0x0e;
        unsigned int  addr_o2=io_base+0x0f;
        unsigned int  addr_i1=io_base+0x0e;
        unsigned int  addr_i3=io_base+0x38;
```

```
unsigned int num_1=inportb(addr_o1)&0xfe;
outportb(addr_o1, num_1);
unsigned int num_2=inportb(addr_o2)&0xe3;
outportb(addr_o2, num_2);
sleep(2);
unsigned int i_0=inportb(addr_i1)&0x04;
unsigned int i_3=inportb(addr_i3)&0xc2;

    num_1=inportb(addr_o1)|0x01;
outportb(addr_o1, num_1);
    num_2=inportb(addr_o2)|0x1c;
outportb(addr_o2, num_2);
sleep(2);
unsigned int i_1=inportb(addr_i1)&0x04;
unsigned int i_2=inportb(addr_i3)&0xc2;

    if
((i_0==0x00)&&(i_3==0x00)&&(i_1==0x04)&&(i_2==0xc2))
        {printf("ok\n");}
    else{printf("fail\n");}
return 0;
}
```

## 第三章

### BIOS 功能简介

---

FSC-1813L2NA主板BIOS相关功能简介请参照我公司的《AMI BIOS 设置指南》。

---

## 附录

---

### Watchdog 编程指引

主板提供一个可按分或按秒计时的，最长达255级的可编程看门狗定时器(以下简称WDT)。通过编程，WDT超时事件可用来将系统复位或者产生一个可屏蔽中断。以下用C语言形式描述了WDT的编程。必须注意：在对WDT进行操作之前，需先进入WDT编程模式；在结束对WDT的操作之后，退出WDT。对WDT的编程需遵循以下步骤：

- 进入WDT编程模式
- 设置WDT工作方式/启动WDT/关闭WDT
- 退出WDT编程模式

需要include 以下几个文件：

```
#include <stdio.h>
```

```
#include <dos.h>
```

```
#include <bios.h>
```

```
#include <stdlib.h>
```

```
#include <string.h>
```

#### (1) 进入WDT编程模式

```
outportb(0x2e, 0x87); //进入WDT编程模式，在INDEX PORT 写2次87
```

```
outportb(0x2e, 0x87); //INDEX PORT [2E], DATA PORT[2F]
```

```
outportb(0x2e, 0x07); //reg0X07, 用来选择logic device
```

```
outportb(0x2f, 0x08); //选择logic device8,
```

```
outportb(0x2e, 0x30); //reg0x30, 设备使能寄存器, 0=disable,  
1=enable
```

```
outportb(0x2f, 0x01); //enable device8
```

(2) 设置WDT工作方式, 复位或中断方式, 选择一种:

```
outportb(0x2e, 0x2D);
```

```
char oldval = inportb(DATAP);
```

a. 配置WDT成复位工作方式

```
oldval &= 0xfe;
```

```
/*reg0x2d.bit0, 复位方式此位置0
```

```
1=GPI050
```

```
0=WDTO
```

```
*/
```

```
outportb(0x2f, oldval);
```

b. 配置WDT成中断工作方式

```
oldval |= 0x01;
```

```
/* reg0x2d.bit0, 中断方式此位置1
```

```
1=GPI050
```

```
0=WDTO
```

```
*/
```

```
outportb(0x2f, oldval);
```

```
outportb(0x2e, 0xf7); //选择WDT中断号
```

```
outportb(0x2f, IRQ_RESOURCE);
```

其中, IRQ\_RESOURCE =0: 禁止使用任何中断

```
/*
```

```
register 0xF7
```

```
bit[3:0]WDT interrupt Mapping
```

```
1111=IRQ 15
```

```
.....
```

---

---

```
0011=IRQ3
0010=IRQ2
0001=IRQ1
0000=Disable
*/
```

(3) **选择WDT按分或按秒计时，选择一种：**

a. **选择WDT按分计时用以下语句：**

;假定已处于WDT编程状态

```
outportb(0x2e, 0xf5); //选择按分计时, register 0xf5.bit3=1。
```

```
outportb(0x2f, 0x08);
```

b. **选择WDT按秒计时以下语句：**

;假定已处于WDT编程状态

```
outportb(0x2e, 0xf5); //选择按秒计时, register 0xf5.bit3=1。
```

```
outportb(0x2f, 0);
```

(4) **启动/禁止WDT**

;假定已处于WDT编程状态

```
outportb(0x2e, 0xf6); //写入预设的时间TIME-OUT-VALUE
```

```
outportb(0x2f, TIME-OUT-VALUE);
```

注意：TIME-OUT-VALUE的取值范围从1到255，计时单位为“分”或“秒”。如果TIME-OUT-VALUE为零，则禁止WDT。

TIME-OUT-VALUE为任何非零值都将启动WDT。

(5) **退出WDT编程模式**

```
outportb(0x2e, 0xaa);
```

## I/O 口地址映射表

系统I/O地址空间总共有64K，每一外围设备都会占用一段I/O地址空间。下表给出了本CPU卡部分设备的I/O 地址分配，由于PCI设备（如PCI网卡）的地址是由软件配置的，表中没有列出。

地址	设备描述
000h - 00Fh	DMA 控制器#1
020h - 021h	可编程中断控制器#1
040h - 043h	系统计时器
060h - 064h	标准 101/102 键盘控制器
070h - 071h	实时时钟, NMI
080h - 09Fh	DMA 页寄存器
0A0h - 0A1h	可编程中断控制器#2
0C0h - 0DEh	DMA 控制器#2
0F0h - 0FFh	数据数值处理器
170h - 177h	从 IDE
1F0h - 1F7h	主 IDE
295h - 296h	硬件监测器
2F8h - 2FFh	串行端口 #2 (COM2)
376h	从 IDE (dual FIFO)
378h - 37Fh	并行端口#1 (LPT1)
3B0h - 3DFh	Intel G31 内置图形控制器
3F0h - 3F5h	标准软磁盘控制器
3F6h	主 IDE (dual FIFO)
3F8h - 3FFh	串行端口#1 (COM1)

## IRQ 中断分配表

系统共有15个中断源，有些已被系统设备独占。只有未被独占的中断才可分配给其它设备使用。ISA设备要求独占使用中断；只有即插即用ISA设备才可由BIOS或操作系统分配中断。而多个PCI设备可共享同一中断，并由BIOS或操作系统分配。下表给出了本CPU卡部分设备的中断分配情况，但没有给出PCI设备所占用的中断资源。

级别	功能
IRQ0	系统计时器
IRQ1	标准 101/102 键或 Microsoft 键盘
IRQ2	可编程的中断控制器
IRQ3	串口#2
IRQ4	串口#1
IRQ5	保留
IRQ6	标准软磁盘控制器
IRQ7	并口#1
IRQ8	系统 CMOS/实时时钟
IRQ9	软件改道到 Int 0Ah
IRQ10	保留
IRQ11	保留
IRQ12	PS/2 鼠标
IRQ13	数据数值处理器
IRQ14	主 IDE
IRQ15	从 IDE

欲获取更多信息请访问研祥网站：<http://www.evoc.com>。