

AIOT0-Q170

用户手册

Version 2.0

免责声明

本手册内容系本公司知识产权，版权归本公司所有。本产品的所有部分，包括配件与软件等其所有权都归本公司所有。未经本公司书面许可，不得以任何形式对此手册和其中所包含的任何内容进行仿制、拷贝、摘抄或转译为其它语言文字。

我们本着对用户负责的态度精心地编写该手册，但不保证本手册的内容完全准确无误。本手册为纯技术文档，无任何暗示及影射第三方之内容，且不承担排版错误导致的用户理解歧义。若有任何因本手册或其所提到之产品的所有资讯，所引起直接或间接的信息流失或事业终止，本公司及其所属员工恕不为其担负任何责任。

由于我们的产品一直在持续的改良及更新，故本公司保留对本手册内容进行修正而不另行通知之权利。

版权声明

本手册中所提及之商标，均属其合法注册公司所有。

本手册所涉及到的产品名称仅做识别之用，其所有权归其制造商或品牌所有人。

目录

第1章 综述	1
1.1 包装清单	1
1.2 主板规格	2
1.3 主板结构图	3
1.4 主板IO接口结构图	3
1.5 主板布局图	4
1.6 IO面板接口	5
第2章 硬件安装	6
2.1 安装I/O后置面板	6
2.2 安装主板到机箱	6
2.3 安装内存	7
2.4 连接外部设备	8
2.4.1 Serial ATA连接器	8
2.4.2 M2-KEY插槽	8
2.4.3 MPCIE和SIM插槽	9
第3章 跳线&接头安装与设置	10
3.1 各跳线设置说明	10
3.2 跳线设置	10
3.3 CFAN 插针接口	11
3.4 ATX和PWR12V 电源接口	11
3.5 JCOM 插针接口	12
3.6 LPT 插针接口	12
3.7 J_GPIO 插针接口	13
3.8 SFAN 插针接口	13
3.9 F_USB 插针接口 (USB2.0插针)	14
3.10 F_PANEL 插针接口	14
3.11 JLPC 插针接口	15
3.12 COM1 接口	15
第4章 BIOS 设置	16
4.1 BIOS解释说明	16
4.2 BIOS设定	16
4.2.1 进入BIOS设定程序	16
4.2.2 控制键位	16
4.3 Main	17

4.4 Advanced	17
4.5 Chipset	29
4.6 Security	31
4.7 Boot	32
4.8 Save & Exit	33
第5章 安装驱动	34
第6章 WDT编程指导	35
6.1 编程指导文档说明	35
6.1.1 涉及到的IO函数定义	35
6.2 系统WatchDog编程	36
6.2.1 WatchDog编程注意事项	36
6.2.2 编程示例	36
第7章 GPIO编程指导	38
7.1 编程指导文档说明	38
7.1.1 涉及到的函数定义（以BIOS中的定义为例）	38
7.2 GPIO编程示例	41
7.2.1 8个GPIO对应的内存地址说明	41
7.2.2 编程示例（以GPIO1为例）：	41
附录1 订购信息	42
附录2 产品有毒有害物质或元素标示表	43

第1章 综述

1.1 包装清单

感谢您选用我们的产品。

请确认您所购买的主板包装是否完整，如果有包装损坏或是有任何配件短缺的情形，请尽快与您的经销商联系。

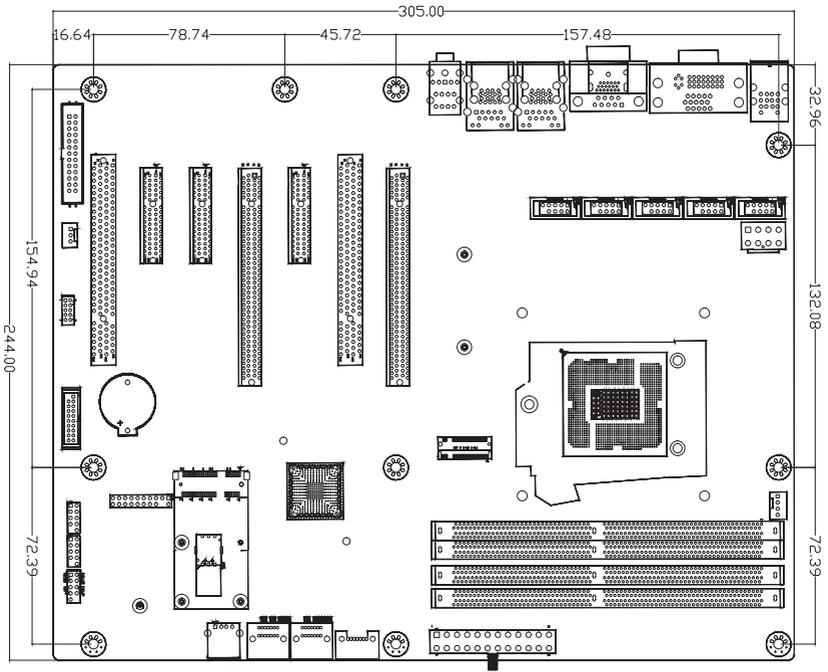
- ★ 主板 X 1
- ★ 驱动光盘 X 1(工业包装：1PCS/箱)
- ★ SATA硬盘转接线 X 1
- ★ COM转接线 X 5
- ★ 专用I/O挡片 X 1

上述附带配件规格仅供参考，实际规格以实物为准，本公司保留修改之权利。

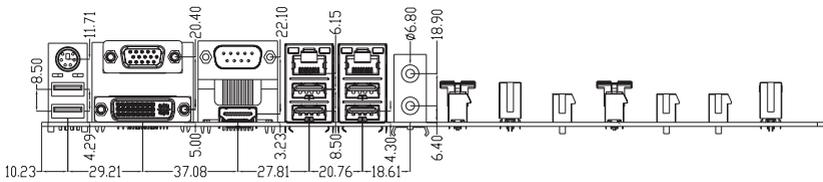
1.2 主板规格

处理器	- 支持Intel LGA1151封装的第 6/7/8/9代i3/i5/i7/i9/Pentium/Celeron CPU
芯片组	- Intel® Q170芯片组/可选C236支持E3 V5&V6
内存	- 4 x 288PIN DDR4 DIMM内存插槽, 单根16GB, 最大支持64GB
显示控制器	- Intel CPU集成显示控制器(依据所配CPU不同)
显示接口	- 三显示输出DVI-D+VGA+HDMI, 支持独立三显
存储	- 5个SATA3.0, 支持SATA RAID 0/1/5/10 - 1个M.2 2280插座, 支持SATA SSD
音频	- 支持 MIC-in, Speaker-out
网络	- 2个Intel GbE, LAN1: Intel I219LM, LAN2: Intel I211AT
USB	- 4个USB3.0 Type A与RJ45整合板边引出; 2个USB3.0插针; - 2个USB2.0 Type A与PS2整合, 1个USB2.0板内向后板边出为USB Dongle预设; - 4个USB2.0 插针(其中2个插针靠后板边预留装机客户机箱前面板扩展)
并口	- 1个并口
串口	- COM1支持 RS232/422/485, COM2-6支持 RS232
键盘鼠标接口	- 1个PS/2 二合一接口与两个USB采用3合1连接器
数字 I/O	- 8位数字I/O, 提供电源和地, +5V电平
LPC 扩展接口	- 提供1个LPC局部总线接口可扩展串口模块等
TPM 接口	- 预留LPC接口支持安全加密模块
电源	- ATX电源, 支持ATX/AT开关机模式
扩展总线	- 2个PCIe x16 (x8信号) - 2个PCIe x4 - 1个 PCIe x4(x1信号) - 2个 PCI (32bit) - 1个 Mini-PCIe (支持SIM卡座支持4G模块) - 1个M.2 2280插座支持SATA SSD
工作环境	- 工作温度和湿度要求:0~60摄氏度, 5%~95%无凝结 - 非工作温度和湿度要求:-40~80摄氏度, 5%~95%无凝结
Watch Dog	- 255级可编程秒/分, 支持超时中断或系统复位
BIOS	- AMI UEFI BIOS
操作系统	- Win7/Win10/WES7/Linux, 32位/64位
PCB 尺寸	- 305mm x 244mm

1.3 主板结构图

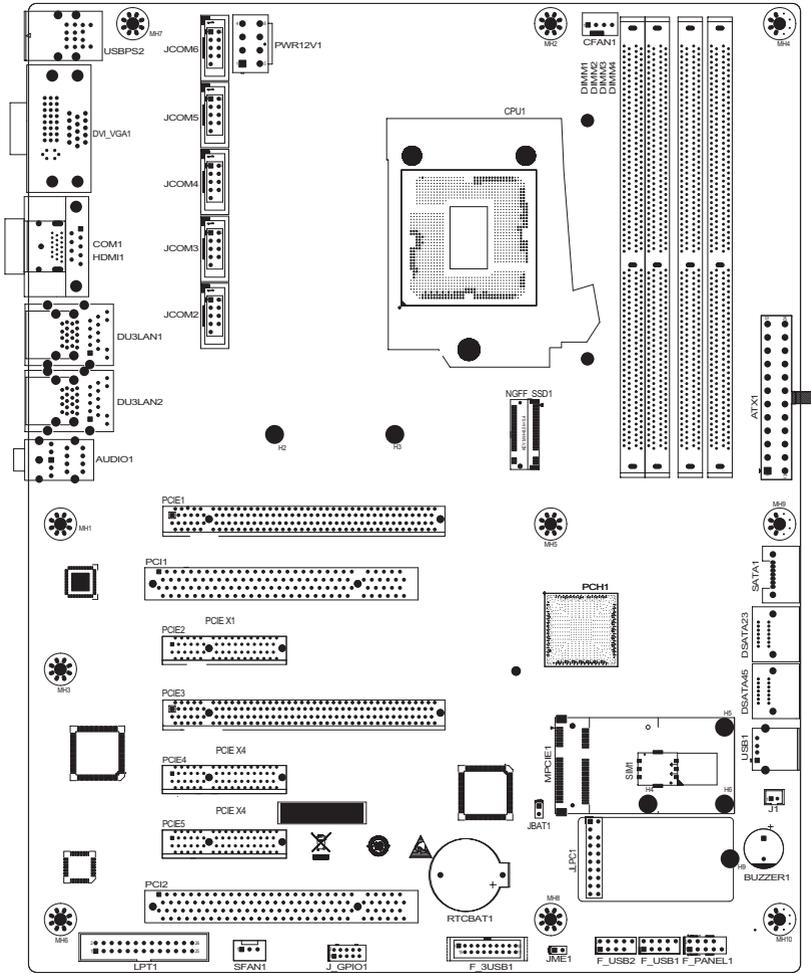


1.4 主板IO接口结构图



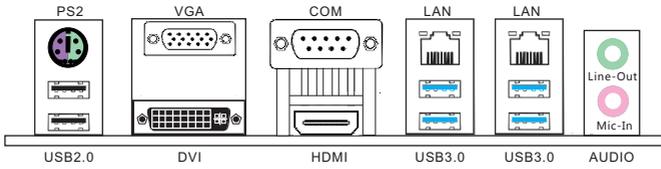
(此图片仅供参考，请以实物为准)

1.5 主板布局图



(此图片仅供参考，请以实物为准)

1.6 IO面板接口



(此图片仅供参考，请以实物为准)

- **PS2:** 鼠标键盘接口
- **USB2.0:** USB2.0接口
- **VGA:** VGA显示接口
- **DVI:** DVI显示接口
- **COM:** 串口
- **HDMI:** HDMI显示接口
- **LAN:** RJ45 以太网接口
- **USB3.0:** USB3.0接口
- **LINE_OUT:** 耳机插孔
- **MIC_IN:** 麦克风插孔

第2章 硬件安装

2.1 安装I/O后置面板

该主板提供了一块I/O后置面板。在机箱内安装此后置面板后，可阻挡无线电射频的传播，保护内部组件不受灰尘和异物侵害，并可促使机箱内的气流保持畅通。在向机箱中安装主板之前，应先安装 I/O 后置面板。首先将机箱本身附带的后置面板卸下来(来回晃动几次即可将其卸下)，然后安装主板附带的外置面板放入机箱，推压防护板，使其紧固到位。

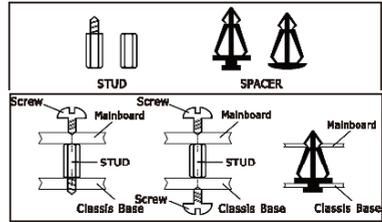
2.2 安装主板到机箱

大多数电脑机箱的主板托架上都会有多个固定孔位，可使主板确实固定并且不会短路，有两种方法可以固定主板到机箱的底座上：

- (1) 使用铜柱
- (2) 使用塑料卡扣

一般来说，最好的方式是使用铜柱来固定主板，只有在您无法使用铜柱时才使用塑料卡扣来固定主板。

小心找寻主板上便可发现许多固定孔位，将这些孔对准主板托架上的固定孔。如果孔能对准并且有螺丝孔，就表示可使用铜柱来固定主板。在安装主板之前，先将机箱提供的主板垫脚螺母安放到机箱主板托架的对应位置（有些机箱购买时就已经安装）。然后双手平行托住主板，将主板放入机箱中，确定机箱安放到位。（可以通过机箱背部的I/O后置面板来确定。）接着拧紧螺丝，固定好主板。（在装螺丝时，注意每颗螺丝不要一次性的就拧紧，等全部螺丝安装到位后，再将每颗螺丝拧紧，这样做的好处是随时可以对主板的位置进行微调。）如果孔对准但是只有凹槽，这表示只能使用塑料卡榫来固定主板。抓住塑料卡榫的尖端并将其底部滑入主板托架的凹槽内，在所有凹槽都装好了卡榫后，您便可将主板对准凹槽固定至定位。主板固定至定位后，且在您将外壳装上之前，请再次检查以确定所有安装都正确无误。



注意：如果有固定铜柱已经锁在主板托架上，且该铜柱与主板对应的地方没有固定孔，请将该铜柱移除，以避免导致主板上的电路短路。

2.3 安装内存

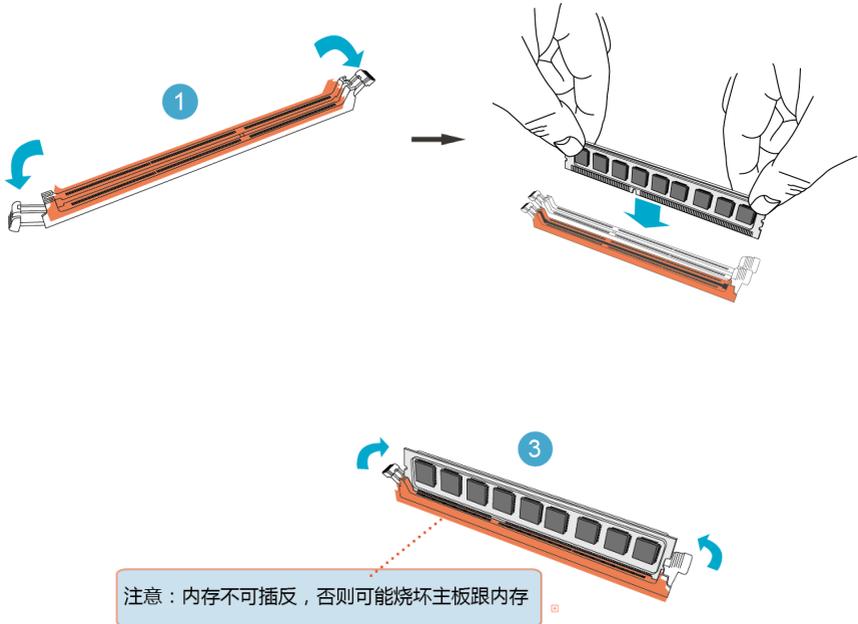
在开始安装内存前，请遵守以下的警告信息：

1. 请先确认您所购买的内存适用本主板所支持的规格。
2. 在安装或移除内存之前，请先确定电脑的电源已经关闭以免造成损毁。
3. 内存设计有防呆标示，若您插入方向错误，内存就无法插入，此时请立即更改插入方向。

安装内存：

1. 在安装或移除内存之前请先关掉电源，并且拨下AC电源线。
2. 推开主板内存卡扣。
3. 查看内存条的缺口，位置对应内存插槽。
4. 对好缺口位置之后，用力按压到插槽，安装成功会有很明显的卡的声音！（注意：将内存条下压的力度，不可过大，以免损坏内存）
5. 要移除内存条，请将DIMM插槽两端的卡扣同时向外推，然后拿出内存条。

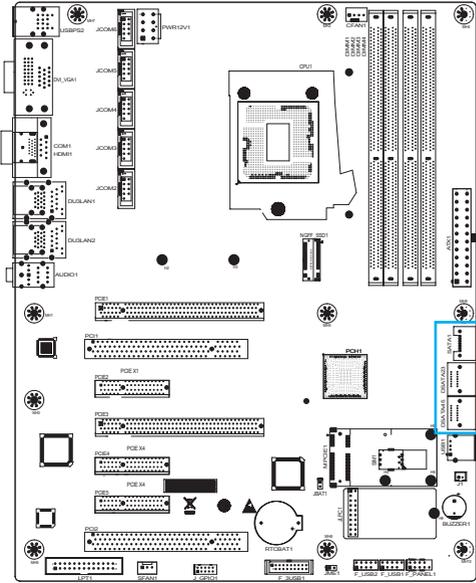
内存安装图示（仅供参考）：



 **注意：**静电会损害电脑或内存的电子元件，所以在进行以上步骤之前，请务必先短暂接触接地金属物体，以去除身上的静电。

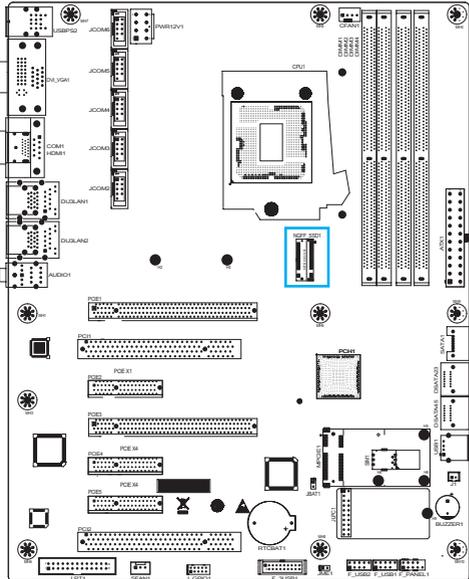
2.4 连接外部设备

2.4.1 Serial ATA连接器



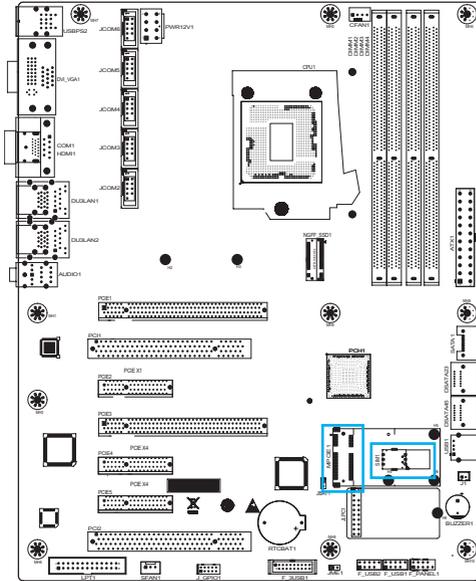
支持使用Serial ATA排线来连接Serial ATA硬盘或是其它符合Serial ATA规范的设备。

2.4.2 M2-KEY插槽



M.2插槽，支持SATA总线的M.2固态硬盘。
安装此卡时，请将卡斜30度插入，然后向下压至螺柱处，再用螺丝将其固定。

2.4.3 MPCIE和SIM插槽



MPCIE插槽，支持4G功能。
安装此卡时，请将卡斜30度插入，然后向下压至螺柱处，再用螺丝将其固定。

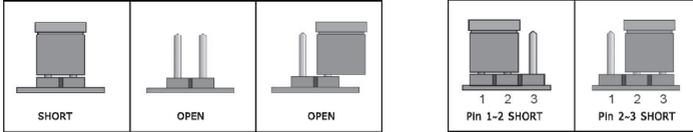
SIM卡槽支持4G卡。

第3章 跳线&接头安装与设置

3.1 各跳线设置说明

2针脚的接头：将跳线帽插入两个针脚将使其关闭（短路）。移除跳线帽或是插入其它针脚（为未来扩充预留）将会使其开启。

3针脚的接头：跳线帽可插入针脚1~2或针脚2~3使其关闭（短路）。



怎么辨认跳线的第1脚位置？

1. 请仔细查看主板，凡有标明“1”或是有白色粗线标记的接脚即为1脚位置。
2. 观看背板的焊盘，通常方型焊盘为第一脚。

3.2 跳线设置

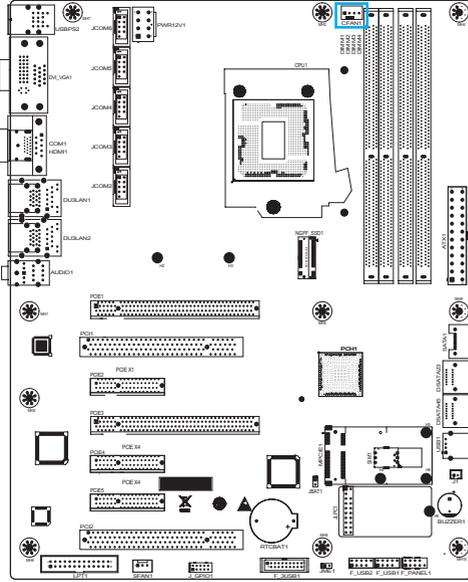
JBAT 跳线设置

接脚	定义
SHORT	CLEAR CMOS
OPEN	NORMAL

JME 跳线设置

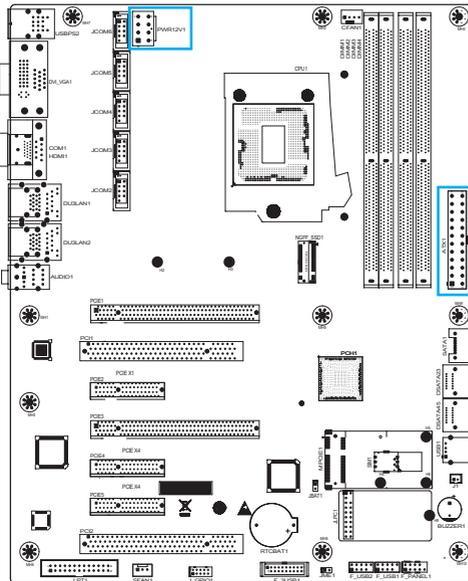
接脚	定义
OPEN	Enable ME
CLOSE	Disable ME

3.3 CFAN 插针接口



接脚	接脚定义
1	GND
2	+12V
3	FAN_SENSE
4	FAN_CTRL

3.4 ATX和PWR12V 电源接口



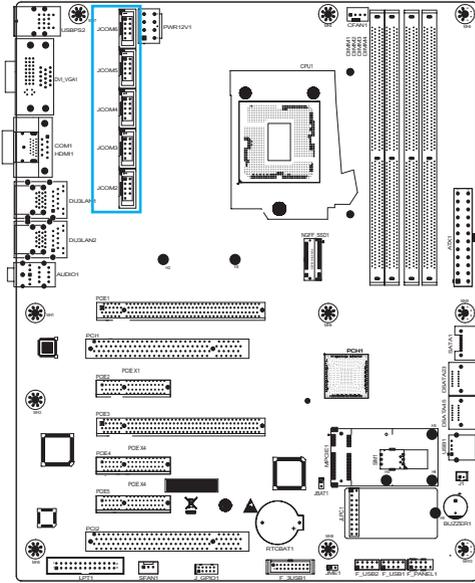
ATX电源插针定义

接脚	接脚定义	接脚	接脚定义
1	+3.3V	13	+3.3V
2	+3.3V	14	-12V
3	GND	15	GND
4	+5V	16	PSON#
5	GND	17	GND
6	+5V	18	GND
7	GND	19	GND
8	POK	20	NC
9	5VSB	21	+5V
10	+12V	22	+5V
11	+12V	23	+5V
12	+3.3V	24	GND

PWR12V电源插针定义

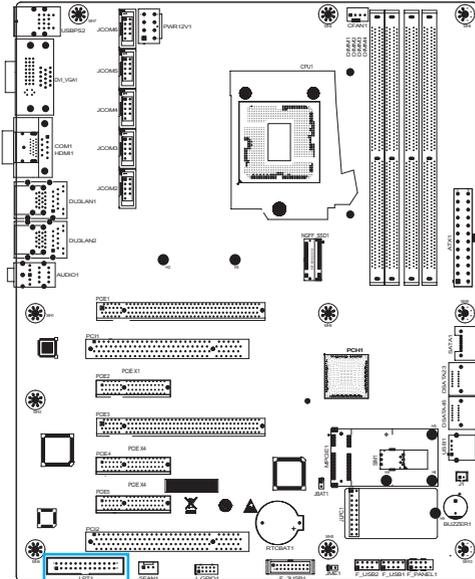
接脚	接脚定义	接脚	接脚定义
1	GND	5	+12V
2	GND	6	+12V
3	GND	7	+12V
4	GND	8	+12V

3.5 JCOM 插针接口



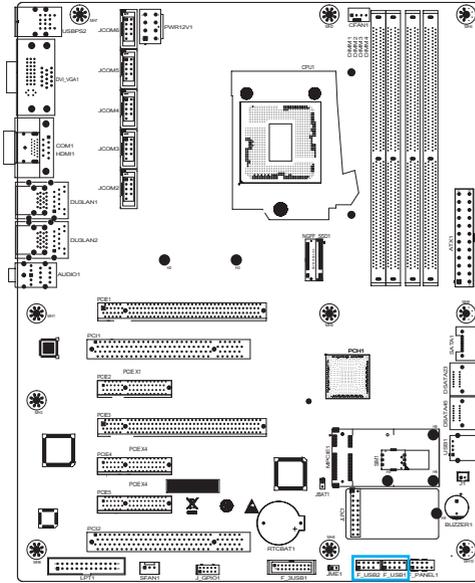
接脚	接脚定义	接脚	接脚定义
1	DCD	2	RXD
3	TXD	4	DTR
5	GND	6	DSR
7	RTS	8	CTS
9	RI		

3.6 LPT 插针接口



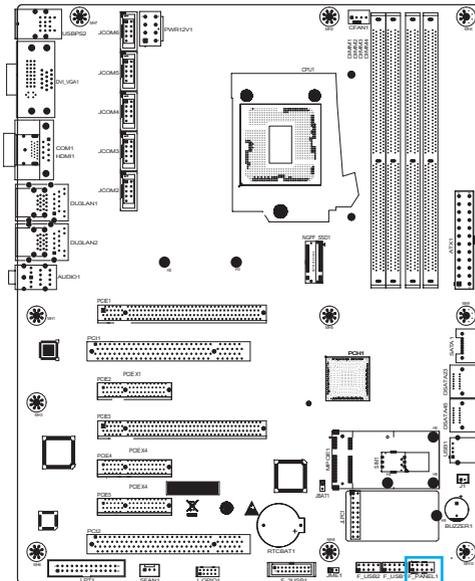
接脚	接脚定义	接脚	接脚定义
1	LPT_STB	2	LPT_AFD
3	LPT_PD0	4	LPT_ERR
5	LPT_PD1	6	LPT_INIT
7	LPT_PD2	8	LPT_SLIN
9	LPT_PD3	10	GND
11	LPT_PD4	12	GND
13	LPT_PD5	14	GND
15	LPT_PD6	16	GND
17	LPT_PD7	18	GND
19	LPT_ACK	20	GND
21	LPT_BUSY	22	GND
23	LPT_PE	24	GND
25	LPT_SLCT	26	NC

3.9 F_USB 插针接口 (USB2.0插针)



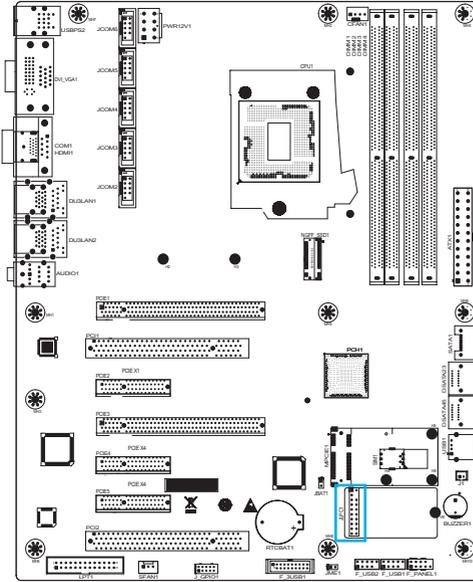
接脚	接脚定义	接脚	接脚定义
1	5V	2	5V
3	D-	4	D-
5	D+	6	D+
7	GND	8	GND
		10	NC

3.10 F_PANEL 插针接口



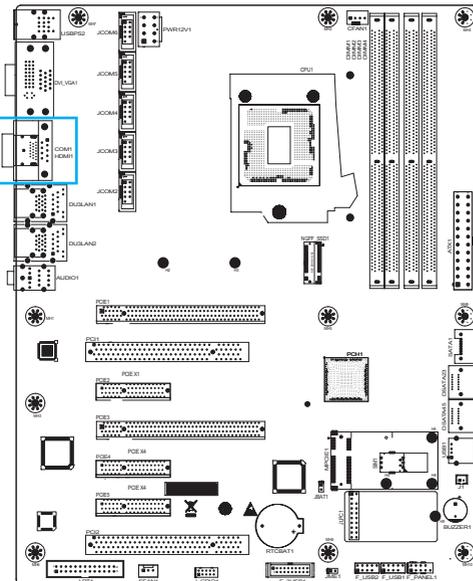
接脚	接脚定义	接脚	接脚定义
1	HDD_LED+	2	PWRLED+
3	HDD_LED-	4	PWRLED-
5	GND	6	PWR-SW
7	SYS-RST	8	GND
9	NC		

3.11 JLPC 插针接口



接脚	接脚定义	接脚	接脚定义
1	LPC_CLK_PCI	2	GND
3	LFRAME#	4	NC
5	LPC_RST	6	5V
7	LAD3	8	LAD2
9	3.3V	10	LAD1
11	LAD0	12	GND
13	SMCLK	14	SMDATA
15	3VSB	16	SERIRQ
17	GND	18	NC
19	NC	20	LPC_CLK

3.12 COM1 接口



接脚	RS232	RS422	RS485
	接脚定义	接脚定义	接脚定义
1	DCD#	TX-	Data-
2	RXD	TX+	Data+
3	TXD	RX+	NC
4	DTR#	RX-	NC
5	GND	GND	GND
6	DSR#	NC	NC
7	RTS#	NC	NC
8	CTS#	NC	NC
9	RI#	NC	NC

注：COM1的串口模式需要通过设置COM Port Mode Setting进行切换：Advanced→Super I/O Configuration→Serial Port 1 Configuration，BIOS默认为RS-232模式，用户可根据需要选择RS-422或RS-485模式。

第4章 BIOS 设置

4.1 BIOS解释说明

本主板使用AMI BIOS。BIOS全称为Basic Input Output System(基本输入输出系统)。它是存储在电脑主板上的一块ROM (Read-Only Memory) 芯片中。当您开启电脑时，BIOS是最先运行的程序，它主要有以下几项功能：

- a. 对您的电脑进行初始化和检测硬件，这个过程叫POST(Power On Self Test)。
- b. 加载并运行您的操作系统。
- c. 为您的电脑硬件提供最底层、最基本的控制。
- d. 通过SETUP管理您的电脑。

被修改的BIOS资料会被存在一个以电池维持的CMOS RAM中，在电源切断时所存的资料不会被丢失。一般情况下，系统运行正常时，无需修改BIOS。如果由于其他原因导致CMOS资料丢失时，须重新设定BIOS值

4.2 BIOS设定

本章提供了BIOS Setup程序的信息，让用户可以自己配置优化系统设置。BIOS中一些未做过多说明的项目，属于非常用项目，在未完全了解其功能之前建议保持默认设置，不要随意更改。

如下情形您需要运行SETUP程序：

- a. 系统自检时屏幕上出现错误信息，并要求进入SETUP程序；
- b. 您想根据客户特征更改出厂时的默认设置。

注意：由于主板的BIOS版本在不断的升级，所以，本手册中有关BIOS的描述仅供参考。我们不保证本说明书中的相关内容与您所获得的信息的一致性。

4.2.1 进入BIOS设定程序

打开电源或重新启动系统，在自检画面可看到如下信息，按键即可进入BIOS设定程序。

Press <Delete> to enter SETUP

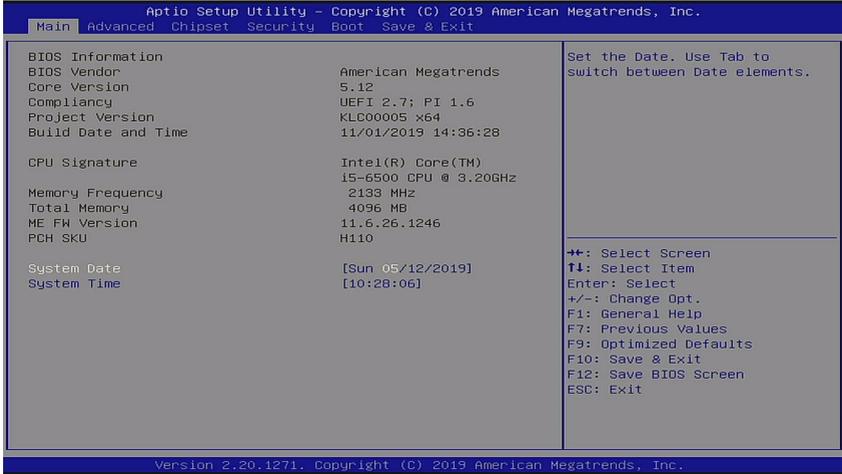
Press <F11> to enter Boot Menu

4.2.2 控制键位

您可以用箭头键移动高亮度选项，按 <Enter> 键进行选择，按 <F1> 键寻求帮助，按 <Esc> 键退出。下列表格将详细列出如何运用键盘来引导系统程序设定。

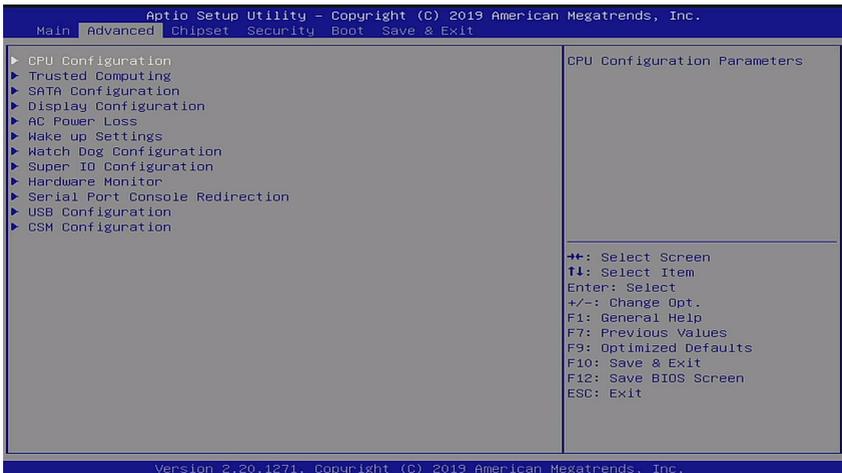
控制键	功能描述
← / →	移动左右箭头选择屏幕
↑ / ↓	移动上下箭头选择上下项目
+ / -	增加/减少数值或改变选择项
<Enter>	选定此选项，进入子菜单
<ESC>	返回主画面，或由主画面中结束CMOS SETUP程序
<F1>	显示相关辅助说明
<F7>	之前设定值
<F9>	载入最优化值的设定
<F10>	保存改变后的CMOS设定值并重启

4.3 Main

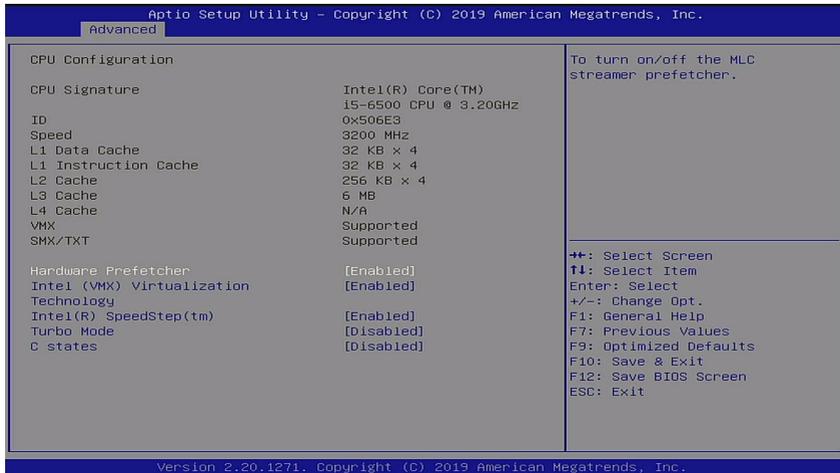


- **BIOS Information**(BIOS相关信息)
- **System Date** (系统日期设置)
设置电脑的日期, 格式为“星期, 月/日/年”。
- **System Time** (系统时间设置)
时间格式为<时><分><秒>。

4.4 Advanced



► **CPU Configuration** 按<Enter>键进入子菜单。



- **Hardware Prefetcher**

设置硬件预取器。

可选项: Enabled, Disabled.

- **Intel (VMX) Virtualization Technology**

设置虚拟化技术功能。

可选项: Enabled, Disabled.

- **Intel (R) SpeedStep (tm)**

CPU根据系统负荷, 自动调节CPU主频的功能。

可选项: Enabled, Disabled.

- **Turbo Mode**

设置处理器Turbo模式。

可选项: Enabled, Disabled.

- **C states**

设置CPU电源管理。

可选项: Enabled, Disabled.

- 按 <Esc> 键返回 “**Advanced**” 主菜单

► **Trusted Computing** 按<Enter>键进入子菜单。

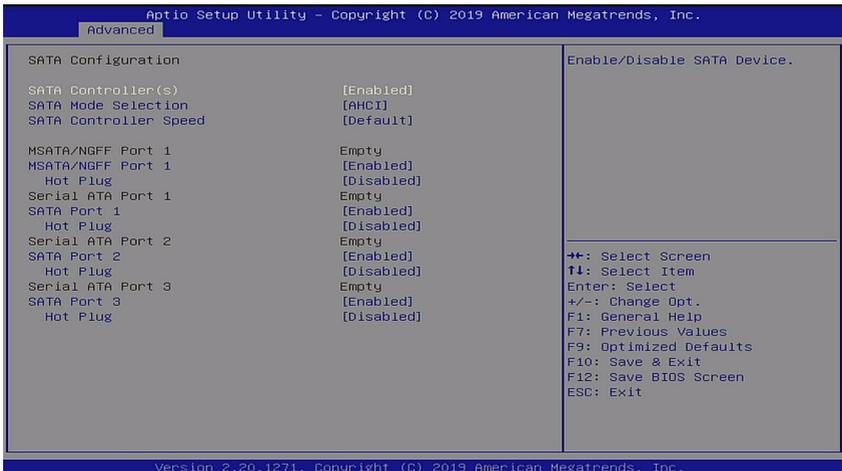


• **Security Device Support**

设置安全设备的BIOS支持。可选项：Enabled, Disabled.

- 按 <Esc> 键返回 “Advanced” 主菜单

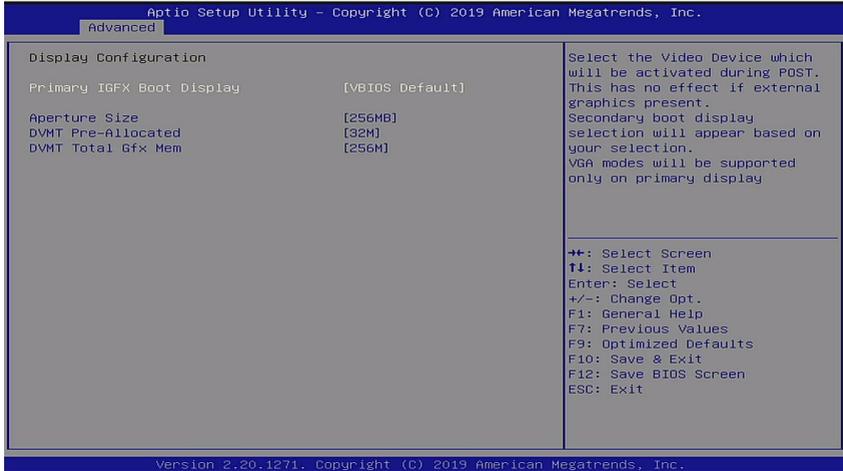
► **SATA Configuration** 按<Enter>键进入子菜单。



此项为设置SATA设备。

- 按 <Esc> 键返回 “Advanced” 主菜单

► **Display Configuration** 按<Enter>键进入子菜单。



- **Primary IGFX Boot Display**

设置视频设备。

可选项：VBIOS Default, DVI, HDMI, VGA.

- **Aperture Size**

设置AGP有效空间的大小，即划拨内存为显存的大小。

可选项：128MB, 256MB, 512MB, 1024MB, 2048MB.

- **DVMT Pre-Allocated**

此项为选择内部图形设备使用的DVMT 5.0预分配（固定）图形内存大小。

可选项：0M, 32M, 64M, 4M, 8M, 12M, 16M, 20M, 24M, 28M, 32M/F7, 36M, 40M, 44M, 48M, 52M, 56M, 60M.

- **DVMT Total Gfx Mem**

此项为选择内部图形设备使用的DVMT 5.0总图形内存大小。

可选项：256M, 128M, MAX.

- 按 <Esc> 键返回 “**Advanced**” 主菜单

► **AC Power Configuration** 按<Enter>键进入子菜单。



- **Power state after AC loss**

交流电断电后，再次通电时的系统状态。

可选项: Power Off, Power On, Last State.

- **ME State**

启用或禁用ME临时禁用模式。 可选项: Disabled, Enabled.

- **ME Unconfig on RTC Clear**

清除RTC时不重新配置ME。

可选项: Disabled, Enabled.

- **Soft-off by PWR-BTTN**

通过电源按钮进行软关机，选择延迟4秒或立即关机。

可选项: Delay 4 sec, Instant-off.

- 按 <Esc> 键返回 “**Advanced**” 主菜单

► **Wake up Settings** 按<Enter>键进入子菜单。



- **Wake system from s5**

RTC从S5唤醒系统功能。

可选项: Disabled, Fixed Time, Dynamic Time.

- **Wake on LAN**

网络唤醒功能。

可选项: Enabled, Disabled.

- 按 <Esc> 键返回 “Advanced” 主菜单

► **Watch Dog Configuration** 按<Enter>键进入子菜单。



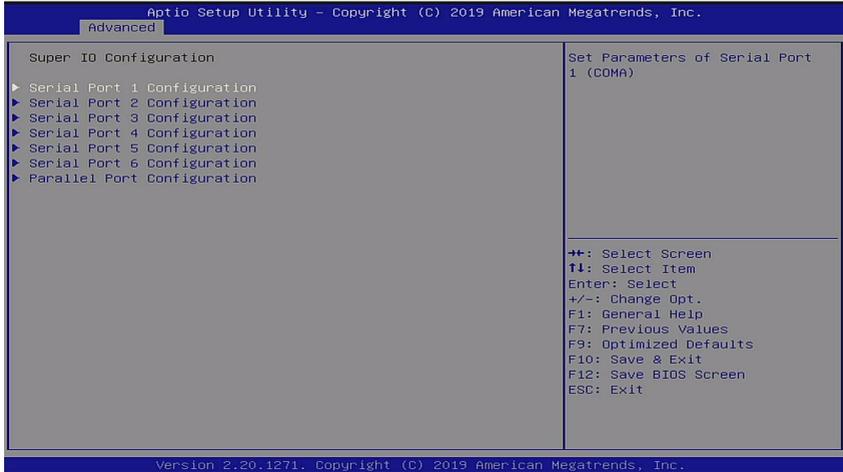
- **Watch Dog Control**

启用/禁用看门狗。

可选项: Enabled, Disabled.

- 按 <Esc> 键返回 “Advanced” 主菜单

- ▶ **Super IO Configuration** 按<Enter>键进入子菜单。



此项为串口设置。

- ▶ **Serial Port 1 Configuration**



- **Serial Port**

设置用串口。

可选项: Enabled, Disabled.

- **COM Port Mode Setting**

设置串口模式。

可选项: RS232, RS422, RS485 DE#/RE.

- **Change Settings**

设置串口的IO地址。

可选项: Auto,

IO=3F8h; IRQ=4;

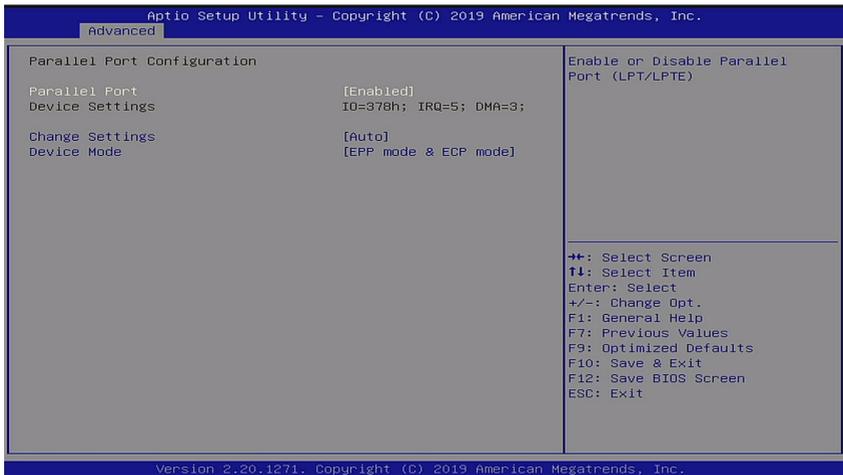
IO=3F8h; IRQ=3, 4, 5, 6, 7, 10, 11, 12;

IO=2F8h; IRQ=3, 4, 5, 6, 7, 10, 11, 12;

IO=3E8h; IRQ=3, 4, 5, 6, 7, 10, 11, 12;

IO=2E8h; IRQ=3, 4, 5, 6, 7, 10, 11, 12

▶ **Parallel Port Configuration**



- **Parallel Port**

设置并口。

可选项: Enabled, Disabled.

- **Change Settings**

设置并口的IO地址。

可选项: Auto,

IO=378h; IRQ=5; DMA=3;

IO=378h; IRQ=5, 6, 7, 10, 11, 12; DMA=1, 3;

IO=278h; IRQ=5, 6, 7, 10, 11, 12; DMA, 1, 3;

IO=3BCh; IRQ=5, 6, 7, 10, 11, 12; DMA=1, 3;

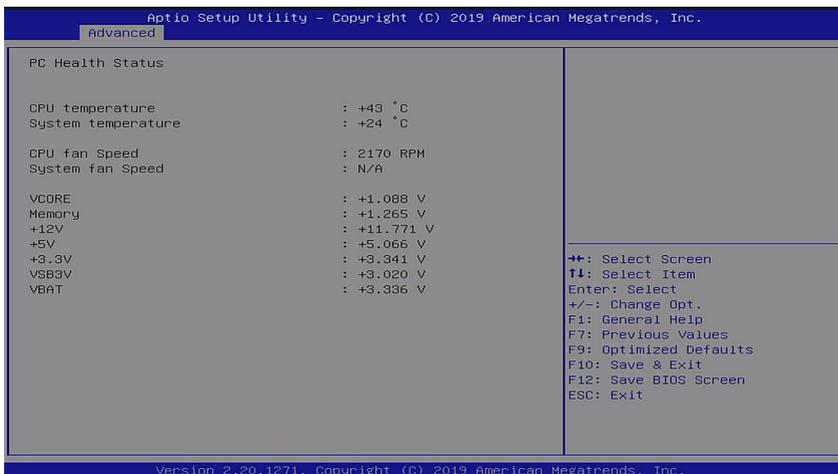
- **Device Mode**

设置打印机端口模式。

可选项: Standard Parallel Port Mode, EPP Mode, ECP Mode, EPP Mode & ECP Mode.

- 按 <Esc> 键返回 “Advanced” 主菜单

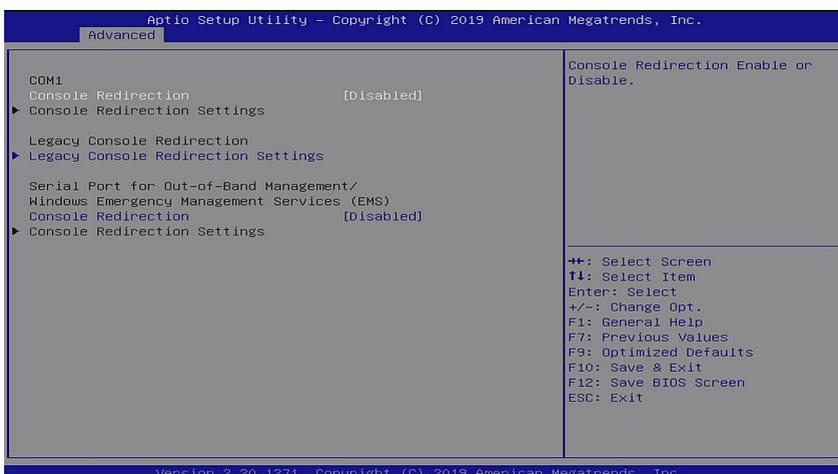
▶ **Hardware Monitor** 按<Enter>键进入子菜单。



此项为监控当前的硬件状态包括CPU温度、电压等系统状态。

- 按 <Esc> 键返回 “Advanced” 主菜单

▶ **Serial Port Console Redirection** 按<Enter>键进入子菜单。



- **Console Redirection**

设置控制台重定向。

可选项: Enabled, Disabled.

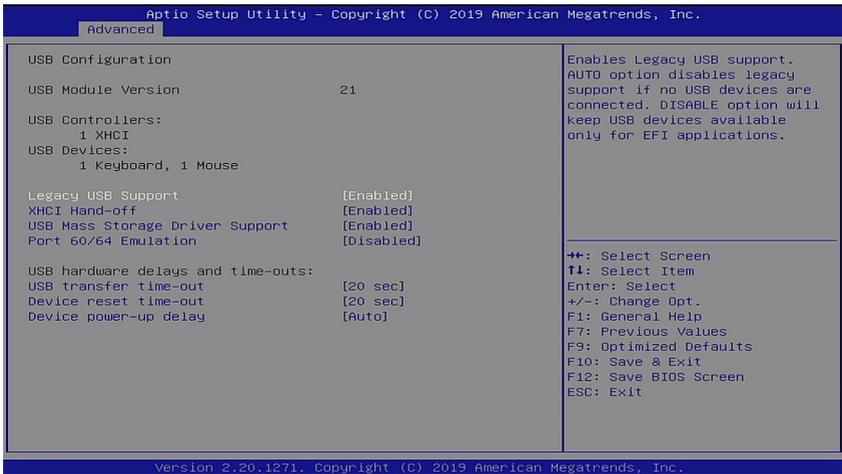
- ▶ **Legacy Console Redirection**

旧版控制台重定向设置。

可进入子菜单设置。

- 按 <Esc> 键返回“**Advanced**”主菜单

- ▶ **USB Configuration** 按<Enter>键进入子菜单。



- **Legacy USB Support**

设置旧版USB支持。

可选项: Enabled, Disabled, Auto.

- **XHCI Hand-off**

设置XHCI驱动程序。

可选项: Enabled, Disabled.

- **USB Mass Storage Driver Support**

设置USB海量存储驱动程序支持。

可选项: Enabled, Disabled.

- **Port 60/64 Emulation**

设置I/O端口60h/64h仿真支持。

可选项: Enabled, Disabled.

- **USB transfer time-out**

设置USB传输超时。

可选项: 1 sec, 5 sec, 10 sec, 20 sec.

- **Device reset time-out**

设备重置超时设置。

可选项: 10 sec, 20 sec, 30 sec, 40 sec.

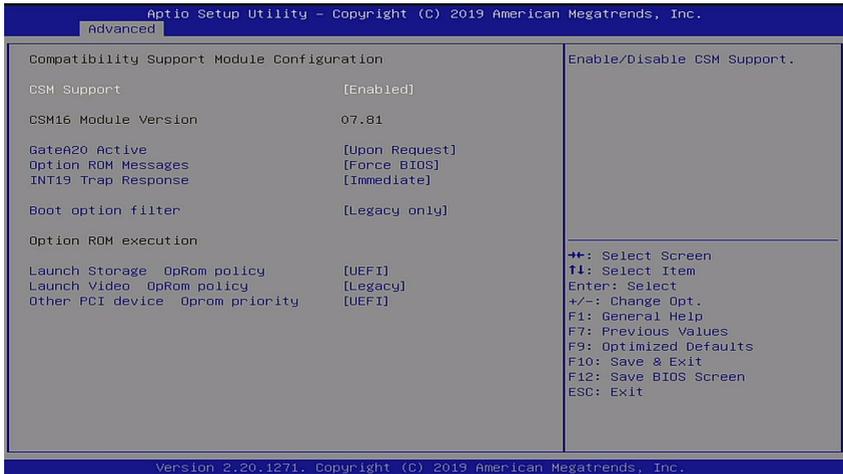
- **Device power -up delay**

设备上电延迟设置。

可选项: Auto, Manual.

- 按 <Esc> 键返回 “**Advanced**” 主菜单

▶ **CSM Configuration** 按<Enter>键进入子菜单。



- **CSM Support**

设置CSM支持。

可选项: Enabled, Disabled.

- **GateA20 Active**

设置BIOS服务禁用GA20。

可选项: Upon Request, Always.

- **Option ROM Messages**

设置Option ROM的显示模式。

可选项: Force BIOS, Keep Current.

- **INT19 Trap Response**

设置INT19陷阱响应。

可选项: Immediate, Postponed.

- **Boot option filter**

设置系统可引导到的设备。

可选项: UEFI and Legacy, Legacy only, UEFI only.

- **Launch Storage OpRom policy**

设置存储OpRom的执行。

可选项: Do not launch, UEFI, Legacy.

- **Launch Video OpRom policy**

设置视频OpRom的执行。。

可选项: Do not launch, UEFI, Legacy.

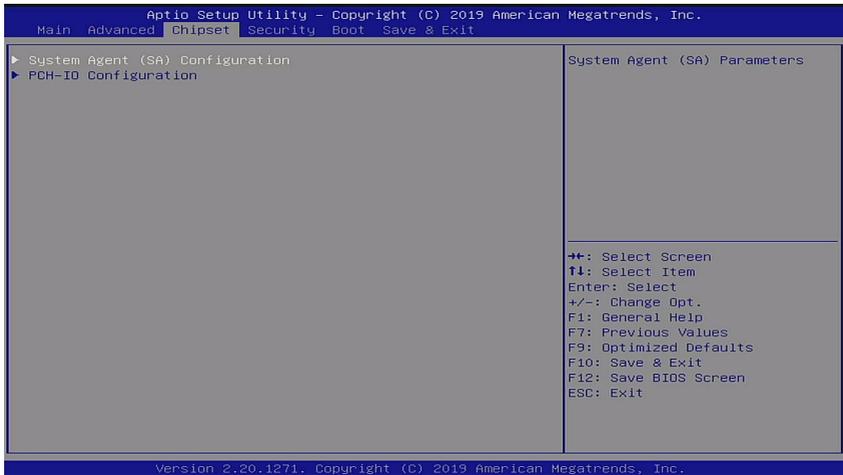
- **Other PCI device OpRom priority**

此项为大容量存储或视频启动设置。

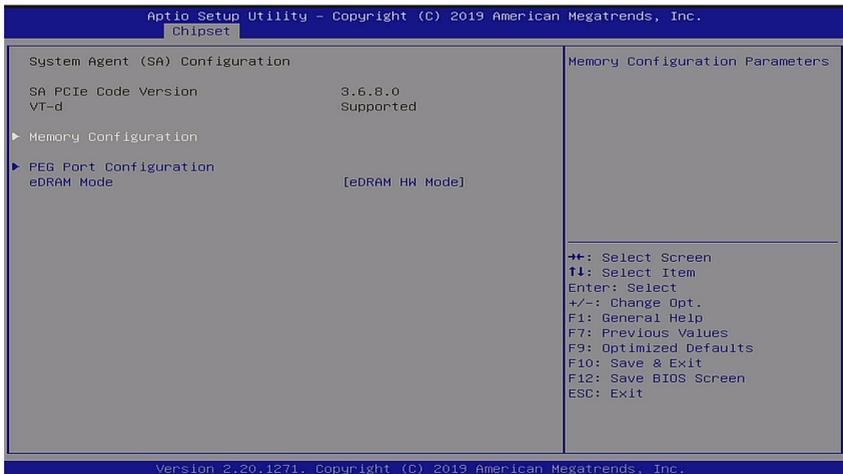
可选项: Do not launch, UEFI, Legacy.

- 按 <Esc> 键返回 “**Advanced**” 主菜单

4.5 Chipset



- ▶ **System Agent (SA) Configuration** 按<Enter>键进入子菜单。



- ▶ **Memory Configuration**

设置内存参数。

可进入子菜单设置。

- ▶ **PEG Port Configuration**

设置PEG端口。

可进入子菜单设置。

- 按 <Esc> 键返回 “Chipset” 主菜单

- ▶ **PCH-IO Configuration** 按<Enter>键进入子菜单。



- ▶ **PCI Express Configuration**

设置PCI Express配置。

可进入子菜单设置。

- ▶ **LAN Configuration**

局域网配置。

可进入子菜单设置。

- ▶ **USB Configuration**

USB配置设置。

可进入子菜单设置。

- ▶ **HD Audio Configuration**

HD Audio配置设置。

可进入子菜单设置。

- 按 <Esc> 键返回 “Chipset” 主菜单

4.6 Security

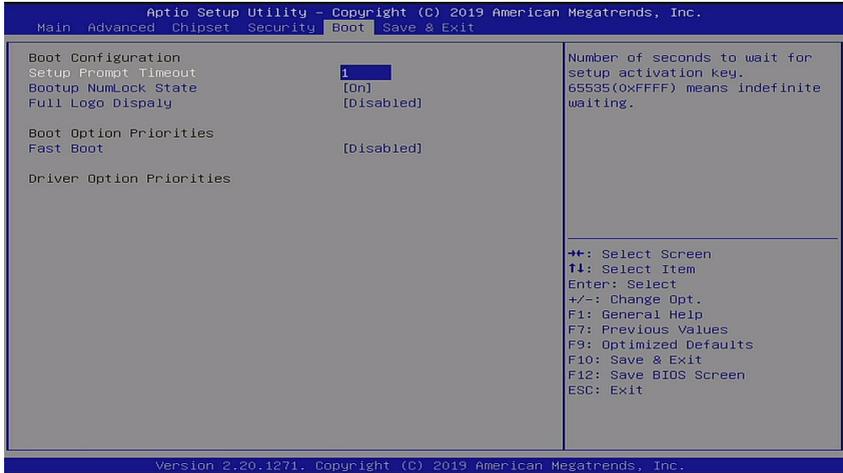


• Administrator Password

设该选项被用来设置系统管理员密码，有以下这些步骤：

1. 选择Administrator Password设置项，按<Enter>键。
2. 在“Create New Password”对话框中输入3~20位要设置的字符或数字密码，输入完成按<Enter>键后，出现“Confirm Password”对话框，再一次输入密码以确认密码正确。若提示“Invalid Password!”，表示两次输入密码不匹配，请重新再输入一次。若要清除系统管理员密码，请选择“Administrator Password”，出现“Enter Current Password”对话框时，输入旧密码后出现“Create New Password”<Enter>密码即清除。

4.7 Boot



- **Setup Prompt Timeout**

设置开机界面停留的时间。

- **Bootup NumLock State**

设置系统启动后，Numlock的状态。当设定为On时，系统启动后将打开NumLock，小键盘的数字键有效。当设定为Off的时候，系统启动后Numlock关闭，小键盘方向键有效。

可选项：On, Off.

- **Full Logo Display**

启用或禁用图标，

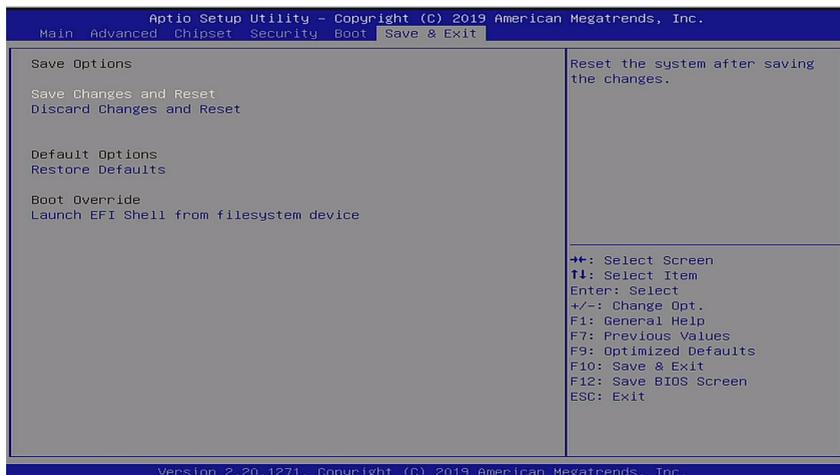
可选项：Enabled, Disabled.

- **Fast Boot**

设置初始化启动活动引导选项所需的最小设备。

可选项：Enabled, Disabled.

4.8 Save & Exit



- **Save Changes and Reset**
保存更改并重启系统。
- **Discard Changes and Reset**
不保存更改并重启系统。
- **Restore Defaults**
还原加载所有选项默认值。
- **Launch EFI Shell from filesystem device**
尝试从文件系统中加载运行Shellx64.efi。

第5章 安装驱动

请将主板驱动光盘放入光驱，光盘将自动运行，并弹出如下图所示界面。如果此界面没有出现，请双击运行X:\AUTORUN.EXE（假设光驱盘符为X:）。



（此图片仅供参考，请以实物为准）

请在以上界面依次点击您需要安装的驱动，根据提示进行安装即可。

第6章 WDT编程指导

6.1 编程指导文档说明

本文档包含了AIoT0-Q170主板的WatchDog二次开发说明和软件样例代码。
说明：本编程指导适用于WDT出自ITE8786 Super IO的相应信号的项目。

6.1.1 涉及到的IO函数定义

Linux系统下，需要包含的几个头文件：

```
#include <stdio.h>
#include <errno.h>
#include <sys/io.h>
```

//从IO地址读1个Byte

```
/*
   Name:IoRead8
   Input:   GpioAddress      - GPIO base address
   Output:  GPIO port read value
   Description:
*/
unsigned char IoRead8(unsigned short GpioAddress) {
    unsigned char ret;
    ret = iopl(3);
    if (ret)
    {
        printf("ret = %d\n", ret);
        printf("errno = %d\n", errno);
        printf("error for iopl\n");
        return 1;
    }
    return inb(GpioAddress);
}
```

```
//向IO地址写1个Byte
```

```
unsigned char IoWrite8(unsigned short GpioAddress, unsigned char Data) {  
    unsigned char ret;  
    ret = iopl(3);  
    if (ret)  
    {  
        printf("ret = %d\n", ret);  
        printf("errno = %d\n", errno);  
        printf("error for iopl\n");  
        return 1;  
    }  
    outb(Data, GpioAddress);  
    return 0;  
}
```

6.2 系统WatchDog编程

6.2.1 WatchDog编程注意事项

1、WatchDog超时时间建议设置不低于5秒，以免软件系统在CPU负荷高时，不能及时复位Watchdog，引起系统不必要的重启。

2、在关闭操作系统和应用软件时，建议先关闭WatchDog或设置较长时间的Watchdog超时时间，以免操作系统未完全关闭时，出现Watchdog复位。

6.2.2 编程示例

说明：本编程指导适用于WDT出自IT8786 Super IO的相应信号的项目。

WDT编程示例

```
#define SuperIO_Index_Port    0x2E  
#define SuperIO_Data_Port    0x2F  
#define WDT_LDN_DEVICE        0x07
```

//a. 进入Super IO的Configuration Mode

```
IoWrite8(SuperIO_Index_Portt, 0x87);  
IoWrite8(SuperIO_Index_Port, 0x01);  
IoWrite8(SuperIO_Index_Port, 0x55);  
IoWrite8(SuperIO_Index_Port, 0x55);
```

//b. 选择逻辑设备GPIO, LDN 7

```
IoWrite8(SuperIO_Index_Port, 0x07);  
IoWrite8(SuperIO_Data_Port , WDT_LDN_DEVICE);
```

//c. Clear WDT状态寄存器

```
IoWrite8(SuperIO_Index_Port, 0x71);  
Data8 =IoWrite8(SuperIO_Data_Port , ((IoRead8(SuperIO_Data_Port) & ~0x10) | 0x01));
```

//d. 设置WDT的工作模式，以秒为单位或分为单位进行计时

```
IoWrite8(SuperIO_Index_Port, 0x72);  
Data8 = IoRead8(SuperIO_Data_Port );  
Data8 = (Data8&0xc0) |0xc0; //WDT计数模式以秒为单位  
// Data8 = (Data8&0x40) |0x40; /*WDT计数模式以分为单位*/  
IoWrite8(SuperIO_Data_Port, Data8 );
```

**//e. 启动WDT，WDT范围为0~255，如果为0，则WDT处于停止模式，非0
//的数值为WDT计数器数值，这个时间之后系统将自动重启。**

UINT8 Timer;

```
IoWrite8(SuperIO_Index_Port,0x73);
```

IoWrite8(SuperIO_Data_Port,Timer); //低字节Timer为用户自定义的1到255之间的任意值。

```
IoWrite8(SuperIO_Index_Port,0x74);
```

IoWrite8(SuperIO_Data_Port,0x00); //高字节Timer一般设置为0x00，因为有低字节的Timer配置已经足够用了。

//f. 停用WDT功能

```
IoWrite8(SuperIO_Index_Port,0x73);  
IoWrite8(SuperIO_Data_Por, 0x00);
```

//g. 重置（喂狗）WDT Timer

```
IoWrite8(SuperIO_Index_Port,0x73);  
IoWrite8(SuperIO_Data_Port,Timer); //Timer为用户自定义的1到255之间的任意值。
```

//h. 退出Super IO的Configuration Mode

```
IoWrite8(SuperIO_Index_Port, 0x02);  
IoWrite8(SuperIO_Index_Port, 0x02); //退出Super IO Configuration Mode后，  
//SuperIO的各个逻辑设备将不能再被访问，除非再次进入Configuration Mode。
```

第7章 GPIO编程指导

7.1 编程指导文档说明

本文档包含了AIoT0-Q170主板的GPIO二次开发说明和软件样例伪代码。

说明：本编程指导适用于GPIO出自Intel PCH的相应信号的项目，不同板卡使用的具体GPIO信号可能不同。另外，请注意访问GPIO的内存地址为物理内存地址。

7.1.1 涉及到的函数定义（以BIOS中的定义为例）

```
/**
    Reads a 32-bit MMIO register.

    Reads the 32-bit MMIO register specified by Address. The 32-bit read value is
    returned. This function must guarantee that all MMIO read and write
    operations are serialized.

    If Address is not aligned on a 32-bit boundary, then ASSERT().

    If 32-bit MMIO register operations are not supported, then ASSERT().

    @param Address The MMIO register to read.

    @return The value read.

**/
UINT32
EFIAPI
MmioRead32 (
    IN      UINTN          Address
)
{
    //
    // Make sure Address is aligned on a 32-bit boundary.
    //
    ASSERT ((Address & 3) == 0);
    return (UINT32)MmioReadWorker (Address, EfiCpuIoWidthUint32);
}

/**
```

Writes a 32-bit MMIO register.

Writes the 32-bit MMIO register specified by Address with the value specified by Value and returns Value. This function must guarantee that all MMIO read and write operations are serialized.

If Address is not aligned on a 32-bit boundary, then ASSERT().

If 32-bit MMIO register operations are not supported, then ASSERT().

@param Address The MMIO register to write.

@param Value The value to write to the MMIO register.

```
/**/  
UINT32  
EFIAPI  
MmioWrite32 (  
    IN     UINTN           Address,  
    IN     UINT32         Value  
)  
{  
    //  
    // Make sure Address is aligned on a 32-bit boundary.  
    //  
    ASSERT ((Address & 3) == 0);  
    return (UINT32)MmioWriteWorker (Address, EfiCpuIoWidthUint32, Value);  
}  
/**
```

Reads memory-mapped registers in the EFI system memory space.

Reads the MMIO registers specified by Address with registers width specified by Width.

The read value is returned. If such operations are not supported, then ASSERT().

This function must guarantee that all MMIO read and write operations are serialized.

@param Address The MMIO register to read.

The caller is responsible for aligning the Address if required.

@param Width The width of the I/O operation.

@return Data read from registers in the EFI system memory space.

```

**/
UINT64
EFIAPI
MmioReadWorker (
    IN        UINTN                Address,
    IN        EFI_CPU_IO_PROTOCOL_WIDTH Width
)
{
    EFI_STATUS Status;
    UINT64     Data;

    Status = mCpuIo->Mem.Read (mCpuIo, Width, Address, 1, &Data);
    ASSERT_EFI_ERROR (Status);

    return Data;
}

```

/**

Writes memory-mapped registers in the EFI system memory space.

Writes the MMIO registers specified by Address with registers width and value specified by Width

and Data respectively. Data is returned. If such operations are not supported, then ASSERT().

This function must guarantee that all MMIO read and write operations are serialized.

@param Address The MMIO register to read.
 The caller is responsible for aligning the Address if required.
@param Width The width of the I/O operation.
@param Data The value to write to the I/O port.

@return Data read from registers in the EFI system memory space.

```

**/
UINT64
EFIAPI
MmioWriteWorker (
    IN        UINTN                Address,
    IN        EFI_CPU_IO_PROTOCOL_WIDTH Width,
    IN        UINT64                Data
)
{
    EFI_STATUS Status;

```

```

Status = mCpuIo->Mem.Write (mCpuIo, Width, Address, 1, &Data);
ASSERT_EFI_ERROR (Status);

return Data;
}

```

7.2 GPIO编程示例

7.2.1 8个GPIO对应的内存地址说明

```

GPIO1: 0xFD AE0638
GPIO2: 0xFD AE0588
GPIO3: 0xFD AE0590
GPIO4: 0xFD AE05E8
GPIO5: 0xFD AE05F0
GPIO6: 0xFD AE05FC
GPIO7: 0xFD AE0600
GPIO8: 0xFD AE0608

```

7.2.2 编程示例（以GPIO1为例）：

```

UINT Data32;

```

GPIO输入的配置和读取

//a. 设置GPIO1为输入

```

Data32 = MmioRead32 (0xFD AE0638);
MmioWrite32 (0xFD AE0638, Data32 &0xFFFFFDFE |0x00000100);

```

//b. 读取GPIO1作为输入时的电平状态

```

Data32 = (MmioRead32 (0xFD AE0638)&0x02)>>1;

```

GPIO输出的配置和读取

//a. 设置GPIO1为输出

```

Data32 = MmioRead32 (0xFD AE0638);
MmioWrite32 (0xFD AE0638, Data32 &0xFFFFFEFF |0x00000200);

```

//b. 设置GPIO1为输出时的电平状态

```

Data32 = MmioRead32 (0xFD AE0638);
MmioWrite32 (0xFD AE0638, Data32 &0xFFFFFFF0); //设置为低电平
或

```

```

MmioWrite32 (0xFD AE0638, Data32 |0x01); //设置为高电平

```

//c. 读取GPIO1为输出时的电平状态

```

Data32 = (MmioRead32 (0xFD AE0638)&0x01);

```

其他的GPIO访问方式类似，只需更换成对应的GPIO地址即可。

订购信息

产品型号	芯片组	内存	显示	存储	USB3	USB2	COM	LAN	PCI	PCIe
AIoT0-Q170	Q170	4DIMM	三显	5SATA	4	10	6	2	2PCI	5PCIe
AIoT0-C236	C236	4DIMM	三显	5SATA	4	10	6	2	2PCI	5PCIe



根据中华人民共和国信息产业部发布的《电子信息产品污染控制管理办法》所展开的 SJ/T11364-2006 标准要求，本产品污染控制标识以及有毒有害物质或元素标识说明如下：

产品有毒有害物质或元素标识：

产品中有毒有害物质或元素的名称及含量

部件名称	有毒有害物质或元素					
	铅(Pb)	汞(Hg)	镉(Cd)	六价铬(Cr(VI))	多溴联苯(PBB)	多溴二苯醚(PBDE)
PCB 板	X	0	0	0	0	0
结构件	0	0	0	0	0	0
芯片	0	0	0	0	0	0
连接器	0	0	0	0	0	0
被动电子元器件	X	0	0	0	0	0
焊接金属	X	0	0	0	0	0
线材	0	0	0	0	0	0
助焊剂, 散热膏, 标签以及其他耗材	0	0	0	0	0	0

0：表示该有毒有害物质在该部件所有均质材料中的含量均在 SJ/T11363-2006 标准规定的限量要求以下。
X：表示该有毒有害物质至少在该部件的某一均质材料中的含量超出 SJ/T11363-2006 标准规定的限量要求。
备注： X 位置的铅的含量超出 SJ/T11363-2006 标准规定的限量要求，但符合欧盟 RoHS 指令的豁免条款。