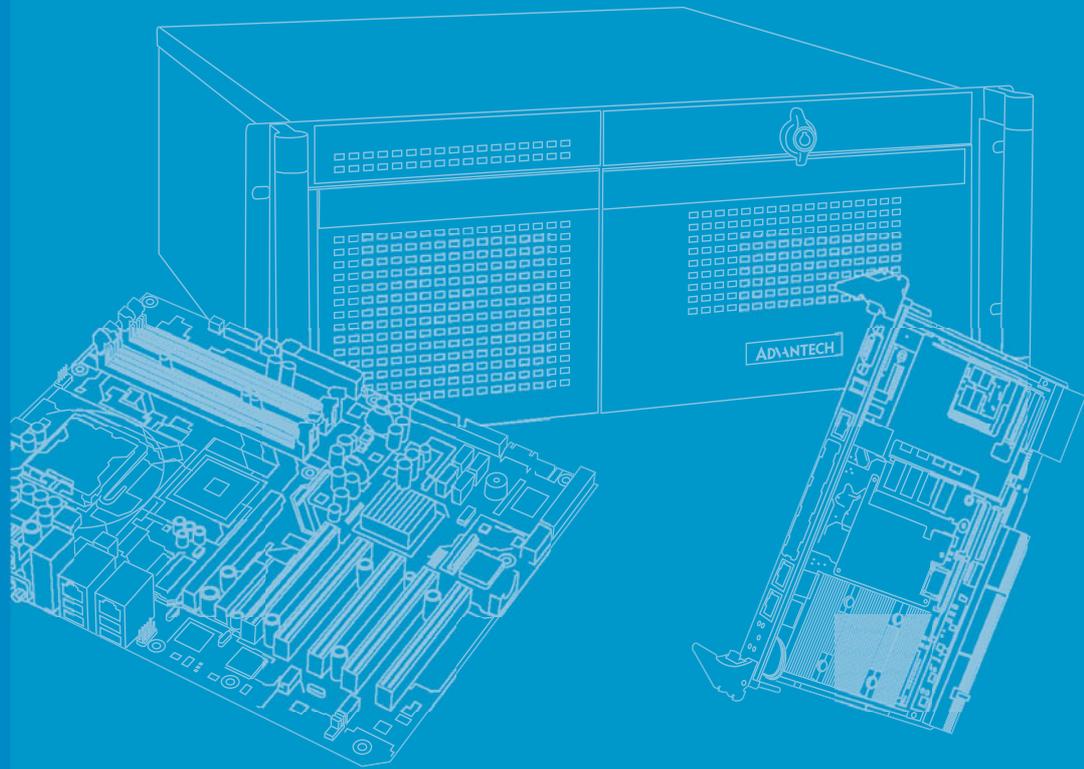


用户手册



## ITA-3630 系列

Intel® Gen2/Gen3 Core™ i7/i5/i3  
带宽电压输入的无风扇嵌入式  
双核紧凑型工业电脑

**ADVANTECH**

*Enabling an Intelligent Planet*

## 版权声明

随附本产品发行的文件为研华公司 2014 年版权所有，并保留相关权利。针对本手册中相关产品的说明，研华公司保留随时变更的权利，恕不另行通知。未经研华公司书面许可，本手册所有内容不得通过任何途径以任何形式复制、翻印、翻译或者传输。本手册以提供正确、可靠的信息为出发点。但是研华公司对于本手册的使用结果，或者因使用本手册而导致其它第三方的权益受损，概不负责。

## 认可声明

ITA-3630 系列为研华公司的商标。

所有其它产品名或商标均为各自所属方的财产。

## 产品质量保证（两年）

从购买之日起，研华为原购买商提供两年的产品质量保证。但对那些未经授权的维修人员维修过的产品不予提供质量保证。研华对于不正确的使用、灾难、错误安装产生的问题有免责权利。

如果研华产品出现故障，在质保期内我们提供免费维修或更换服务。对于出保产品，我们将会酌情收取材料费、人工服务费用。请联系相关销售人员了解详细情况。

如果您认为您购买的产品出现了故障，请遵循以下步骤：

1. 收集您所遇到的问题信息（例如，CPU 主频、使用的研华产品及其它软件、硬件等）。请注意屏幕上出现的任何不正常信息显示。
2. 打电话给您的供货商，描述故障问题。请借助手册，产品和任何有帮助的信息。
3. 如果您的产品被诊断发生故障，请从您的供货商那里获得 RMA (Return Material Authorization) 序列号。这可以让我们尽快地进行故障产品的回收。
4. 请仔细地包装故障产品，并在包装中附上完整的售后服务卡片和购买日期证明（如销售发票）。我们对无法提供购买日期证明的产品不提供质量保证服务。
5. 把相关的 RMA 序列号写在外包装上，并将其运送给销售人员。

料号：200K363020

中国印刷

第一版

2014 年 1 月

## 技术支持与服务

1. 有关该产品的最新信息，请访问研华公司的网站：  
<http://support.advantech.com.cn>
2. 用户若需技术支持，请与当地分销商、销售代表或研华客服中心联系。进行技术咨询前，用户须将下面各项产品信息收集完整：
  - 产品名称及序列号
  - 外围附加设备的描述
  - 用户软件的描述（操作系统、版本、应用软件等）
  - 产品所出现问题的完整描述
  - 每条错误信息的完整内容

## 致客户

### 研华为客户提供的服务

研华的每一款产品都是严格按照规格生产的。这样，产品的可靠性在恶劣粗糙的工业环境下也可以得到保证。无论您购买的研华产品属于实验室还是工厂层，请坚信它会为已知的研华产品提供可靠性和易于操作性。客户的满意是我们最关注的。下面是研华客户服务指南。为保证您从我们的服务中获得最大的利益，请谨慎遵循下面的操作指南。

### 技术支持

我们衷心希望您购买的产品能够发挥最大的性能。如果您遇到技术问题，我们随时准备为您提供帮助。对于常见问题，您可以在产品文档中找到满意答案。这些答案通常比我们可以在电话上给您提供的答案更为详细。请先参考本手册。如果仍找不到方案，请搜集和故障有关的所有信息和问题，汇同你手边的资料，给您的经销商打电话。我们的经销商都是接受过专业培训的。通过您提供的产品信息，他们会为您提供所需要的技术支持。事实上，多数问题都是很微小的，通过电话咨询即可解决。此外，在每个工作日，研华工程师都为客户提供免费的技术支持。关于研华任意一款产品安装和操作方面的应用需求或具体信息，我们都时刻准备着为您提供相关的建议。

## 初始检查

打开包装时，用户需确认包装中含有本设备以及下面所列各项：

- 1 x ITA-3630 系列工业电脑
- 1 x ITA-3630 附件盒
- 1 x 质保卡

如果其中任何一项缺失或者破损，请立即联系您的销售商或销售代表。装货前，我们已全面仔细检查过 ITA-3630 产品。因此您购买的产品应当是完好无损且运转正常的。在您打开 ITA-3630 产品的包装时，请检查是否有破损痕迹（例如，包装箱损坏、划痕、凹痕等）。如果产品有破损或者不符合规格，请立即联系我们的服务部门或您的销售商。同时也要通知搬运人员。请保留包装箱及包装材料以备搬运人员检查。检查之后，我们会给您提供维修或更换服务。

## 安全指示

1. 请仔细阅读此安全操作说明。
2. 请妥善保存此用户手册供日后参考。
3. 用湿抹布清洗设备前，请从插座拔下电源线。请不要使用液体或去污喷雾剂清洗设备。
4. 对于使用电源线的设备，设备周围必须有容易接触到的电源插座。
5. 请不要在潮湿环境中使用设备。
6. 请在安装前确保设备放置在可靠的平面上，意外跌落可能会导致设备损坏。
7. 当您连接设备到电源插座上前，请确认电源插座的电压是否符合要求。
8. 请将电源线布置在人们不易绊到的位置，并不要在电源线上覆盖任何杂物。
9. 请注意设备上的所有警告标识。
10. 如果长时间不使用设备，请将其同电源插座断开，避免设备被超标的电压波动损坏。
11. 请不要让任何液体流入通风口，以免引起火灾或者短路。
12. 请不要自行打开设备。为了确保您的安全，请由经过认证的工程师来打开设备。
13. 如遇下列情况，请由专业人员来维修：
  - 电源线或者插头损坏；
  - 设备内部有液体流入；
  - 设备曾暴露在过于潮湿的环境中使用；
  - 设备无法正常工作，或您无法通过用户手册来使其正常工作；
  - 设备跌落或者损坏；
  - 设备有明显的外观破损。
14. 请不要把设备放置在超出我们建议的温度范围的环境，即不要低于  $-25^{\circ}\text{C}$  ( $-13^{\circ}\text{F}$ ) 或高于  $60^{\circ}\text{C}$  ( $140^{\circ}\text{F}$ )，否则可能会损坏设备。
15. **注意：**计算机配置了由电池供电的实时时钟电路，如果电池更换不正确，将有爆炸的危险。因此，只可以使用制造商推荐的同一种或者同等型号的电池进行替换。请按照制造商的指示处理旧电池。

根据 IEC 704-1:1982 的规定，操作员所在位置的声压级不可高于 70dB(A)。

**免责声明：**该安全指示符合 IEC 704-1 的要求。研华公司对其内容的准确性不承担任何法律责任。

## 安全措施 – 静电防护

为了保护您和您的设备免受伤害或损坏，请遵照以下安全措施：

- 操作设备之前，请务必断开机箱电源，以防触电。
- 在更改任何配置之前请断开电源，以免在您连接跳线或安装卡时，瞬间电涌损坏敏感电子元件。
- 无论何时进行操作，请务必完全断开机箱电源。不可电源接通时进行设备连接，以避免瞬间电涌损坏敏感电子元件。只有专业技术人员才可以打开机箱。接触产品的母板、底板或附加卡前，请先确保您接地来移除身上附带的静电。由于现在的电子设备对静电十分敏感，为了安全起见，请使用接地腕环。请将所有电子元件放在无静电的表面或静电屏蔽袋中。

## 电池信息

电池、电池组和蓄电池不应作为未分类的生活垃圾处理。请使用公共收集系统返回和回收，或按照当地法规要求进行处理。





# 目录

<b>第 1 章</b>	<b>概述</b>	<b>1</b>
1.1	产品简介	2
1.2	产品规格	2
1.3	电源信息	2
	表 1.1: 电源	2
1.4	环境规格	3
	表 1.2: 环境规格	3
1.5	产品尺寸	3
	图 1.1: ITA-3630 产品尺寸	3
	图 1.2: ITA-3630 分解图	4
	表 1.3: 零件列表	4
<b>第 2 章</b>	<b>硬件安装</b>	<b>5</b>
2.1	简介	6
2.2	跳线和接口	6
2.2.1	跳线描述	6
2.2.2	跳线和接口位置图	7
	图 2.1: 主板接口及跳线位置图	7
	表 2.1: 跳线	7
	表 2.2: JLVDS1: LVDS 电压设定	7
	表 2.3: JCMOS1: 清除 JCMOS 设定	8
	表 2.4: PSON1: 开机模式设定	8
	表 2.5: VCCGPIO1: GPIO 电压设定	8
2.3	I/O 接口	9
	图 2.2: ITA-3630 后视图	9
	图 2.3: ITA-3630 前视图	9
2.3.1	COM 端口	10
	表 2.6: COM 接口针脚定义	10
2.3.2	HDMI 接口	10
	表 2.7: VGA 接口针脚定义	10
2.3.3	VGA 接口	11
	表 2.8: VGA 接口针脚定义	11
2.3.4	US2.0 接口	11
	表 2.9: USB2.0 针脚定义	11
2.3.5	USB3.0 接口	12
	表 2.10: USB3.0 针脚定义	12
2.3.6	DIO 接口	12
	表 2.11: DIO 接口针脚定义	12
2.3.7	Audio 接口	12
2.3.8	PS/2 接口	13
	表 2.12: PS/2 接口针脚定义	13
2.3.9	LAN 接口	13
	表 2.13: 以太网接口针脚定义	13
	表 2.14: 网口 LED 灯色状态	14
2.3.10	Phoenix 端子接口	14
	表 2.15: 电源接口	14
2.3.11	LPT 接口	14
	表 2.16: LPT 接口针脚定义	14
2.3.12	LVDS 接口	15
	表 2.17: LVDS 接口针脚定义	15
2.3.13	系统状态 LED 指示灯	15

## 第 3 章 系统安装 ..... 17

3.1	系统安装简介.....	18
3.1.1	安装主板 mini-PCIe 卡和 mini SATA 卡.....	18
	图 3.1: 安装 mSATA 和 mini-PCIe 示意图.....	18
3.1.2	安装 USB Dongle.....	18
	图 3.2: 安装 USB Dongle.....	18
3.1.3	安装硬盘模组.....	19
	图 3.3: 安装双 HDD 模组.....	19
	图 3.4: 安装 HDD+CF 模组.....	20
3.1.4	安装脚架.....	20
	图 3.5: 安装脚架示意图.....	20
3.1.5	安装注意事项.....	21
	图 3.6: 桌面式安装.....	21
	图 3.7: 悬挂式安装.....	21

## 第 4 章 BIOS 设置..... 23

4.1	简介.....	24
	图 4.1: 设置程序初始页面.....	24
4.2	进入设置界面.....	24
	图 4.2: Post 页面.....	24
4.3	主设置界面.....	25
	图 4.3: 主设置页面.....	25
4.3.1	System Time/System Date.....	25
4.4	高级 BIOS 特性设置.....	26
	图 4.4: 高级 BIOS 特性设置.....	26
4.4.1	ACPI 设置.....	26
	图 4.5: ACPI 设置.....	26
4.4.2	CPU Configuration.....	27
	图 4.6: CPU Configuration(1).....	27
	图 4.7: CPU Configuration(2).....	27
4.4.3	SATA Configuration.....	28
	图 4.8: SATA Configuration.....	28
4.4.4	USB Configuration.....	29
	图 4.9: USB Configuration.....	29
4.4.5	Super IO Configuration.....	30
	图 4.10: Super IO Configuration.....	30
	图 4.11: Serial Port Configuration.....	30
4.4.6	H/W Monitor.....	31
	图 4.12: 设置 H/W Monitor 页面.....	31
4.4.7	CPU PPM Configuration.....	31
	图 4.13: CPU PPM Configuration.....	31
4.4.8	APM Configuration.....	32
	图 4.14: APM Configuration.....	32
4.5	高级芯片组特性设置.....	33
	图 4.15: 高级芯片组特性设置.....	33
4.5.1	PCH-IO Configuration.....	33
	图 4.16: PCH-IO Configuration.....	33
	图 4.17: 设置 Chipset USB Configuration.....	34
	图 4.18: PCH Azalia Configuration.....	35
4.5.2	System Agent (SA) Configuration.....	35
	图 4.19: System Agent (SA) Configuration Menu.....	35
	图 4.20: 设置 Graphics Configuration Menu.....	36
	图 4.21: LCD Control.....	36
4.6	启动设置.....	37
	图 4.22: 启动设置.....	37
4.6.1	Boot Configuration.....	37
4.6.2	Boot Option Priorities.....	38

	图 4.23: Boot Option Priorities.....	38
4.6.3	CSM 参数 .....	38
	图 4.24: CSM 参数.....	38
4.7	安全设置.....	39
	图 4.25: Setup Security Menu.....	39
4.8	Save & Exit 设置.....	40
	图 4.26: Setup Save & Exit Menu.....	40

## 第 5 章 驱动安装 ..... 43

5.1	简介.....	44
5.2	驱动安装.....	44
5.2.1	芯片组 Windows 驱动安装.....	45
5.2.2	VGA Windows 驱动安装 .....	45
5.2.3	ME Windows 驱动安装 .....	46
5.2.4	LAN Windows 驱动安装 .....	46
5.2.5	USB 3.0 Windows 驱动安装 .....	47
5.2.6	AUDIO windows 驱动安装 .....	47

## 第 6 章 GPIO 编程指南..... 49

6.1	ITA-3630 数字 DIO 定义 (见 2.3.6).....	50
6.2	配置顺序.....	50
6.3	Function call for reference.....	54

## 附录 A 看门狗定时器编程 ..... 59

A.1	看门狗定时器编程.....	60
A.1.1	看门狗定时器概述 .....	60
A.1.2	编程看门狗定时器 .....	60
	表 A.1: 看门狗定时器寄存器.....	60
A.1.3	编程示例.....	61



# 第 1 章

## 概述

内容包括：

- 产品简介
- 产品规格
- 电源信息
- 环境规格
- 产品尺寸

## 1.1 产品简介

ITA-3630 是一款带 Core™ i7/i5/i3 双核处理器，及宽电压输入的无风扇嵌入式紧凑型工业电脑，是研华公司为智能交通 - 电子警察领域推出的专用机型。这款功能强大的计算平台能够 7 天 24 小时连续工作。

## 1.2 产品规格

- **芯片和芯片组：**英特尔 Core™ i7/i5/i3 处理器和英特尔 HM76 芯片组
- **BIOS：**AMI SPI 64 Mb Flash
- **内存：**板载 4GB DDR3 1333/1600MHz
- **显示：**集成显卡 HD4000，共享系统内存最高达 256MB SDRAM
- **显示模式：**
  - 单显分辨率可达 2048 x 1536 @ 60 Hz
  - 双显分辨率可达 1920 x 1200 @ 60 Hz
- **存储：**支持 2 个 2.5” HDD 槽位或者 1 个 CF 插槽和 1 个 2.5” HDD 槽位
- **扩展槽：**支援 1 个 Mini PCIe 插槽
- **以太网：**2 个 10/100/1000M 以太网（控制芯片：Intel i210-AT）
- **USB：**4 个 USB 2.0 和 2 个 USB 3.0
- **VGA：**1 个 VGA
- **HDMI：**1 个 HDMI
- **LVDS：**1 个 LVDS（主板预留 Pin header）
- **串口：**2 个 COM，支持 RS-232/422/485 切换
- **数字 I/O：**1 个 8-ch GPIO
- **声音：**1 个带 2 x 4w 放大器的 Speaker Out，1 个 Mic in.
- **预留区域（前面板，可选）：**
  - 1 个 LPT
  - 1 个 LVDS
  - 2 个 VGA（可选搭配显卡 PCA-5650 扩展 2 个 VGA 端口）
- **产品尺寸（W x H x D）：**无挂耳：213 x 82.7 x 240 mm  
有挂耳：253 x 86.6 x 240 mm
- **净重：**3.75Kg

## 1.3 电源信息

ITA-3630 采用直流电源 9 到 36V 的宽压输入。

表 1.1：电源

输入直流电压	9 V - 36 V
输入电流	7.2 A - 1.8 A
电源输入接口	2P 凤凰端子（欧规端子）

## 1.4 环境规格

表 1.2: 环境规格

工作温度	配工业级 HDD: 0 ~ 40° C 配 CF 卡: -25 ~ 60° C (带 0.7m/s 风流)
安规认证温度	-10 ~ 50° C (配工规 SSD/CF/mSATA)
存储温度	-40 ~ 85° C
湿度	95% @ 40° C, 非凝结
振动	带 Compact Flash: 2Grms @ 5 ~ 500 Hz, 随机, 1hr/axis 带 2.5-inch HDD: 1Grms @ 5 ~ 500 Hz, 随机, 1hr/axis IEC60068-2-6 Sine 2G @ 5 ~ 500Hz, 1hr/axis
冲击	带 Compact Flash: 20G, IEC-68-2-27, 半正弦波, 11ms 间隔 带 2.5-inch HDD: 10G, IEC-68-2-27, 半正弦波, 11ms 间隔
安全	符合 UL、CCC、BSMI、CE 和 FCC 标准

## 1.5 产品尺寸

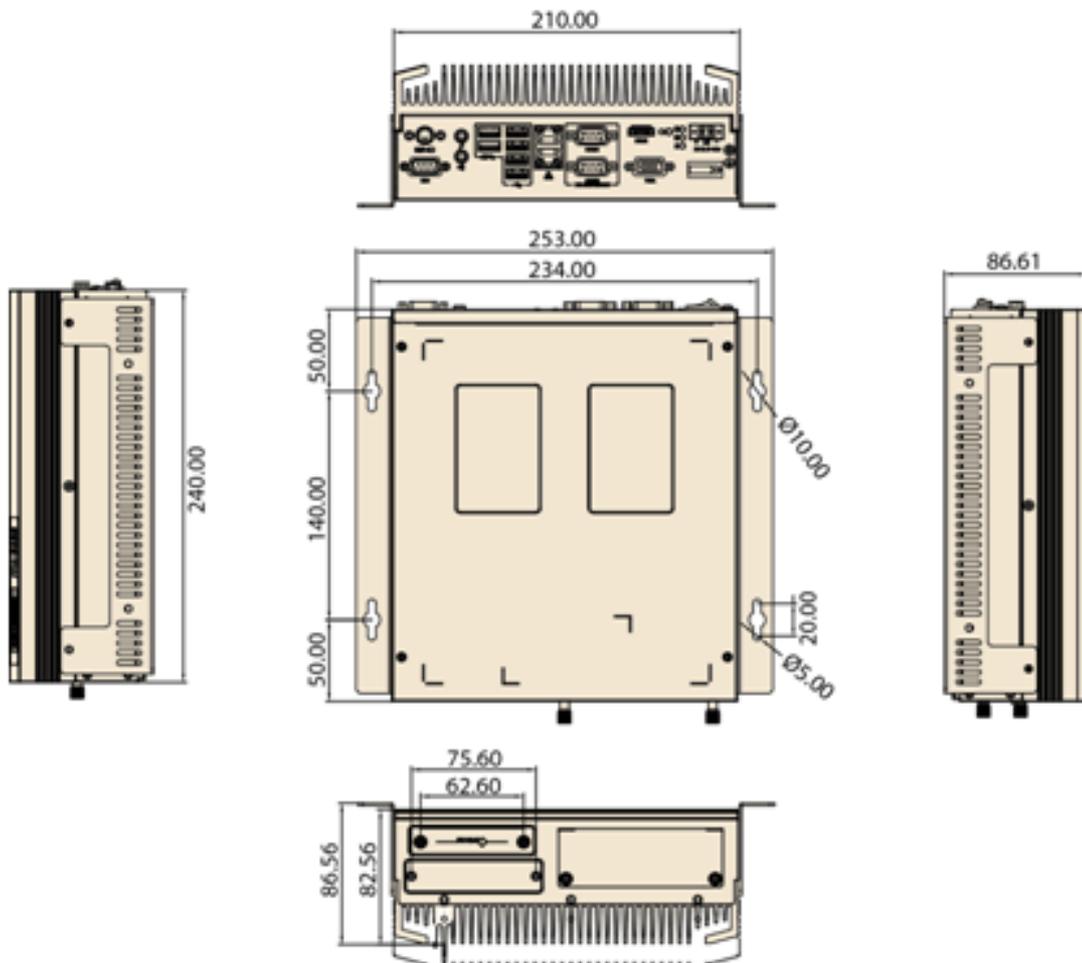


图 1.1: ITA-3630 产品尺寸

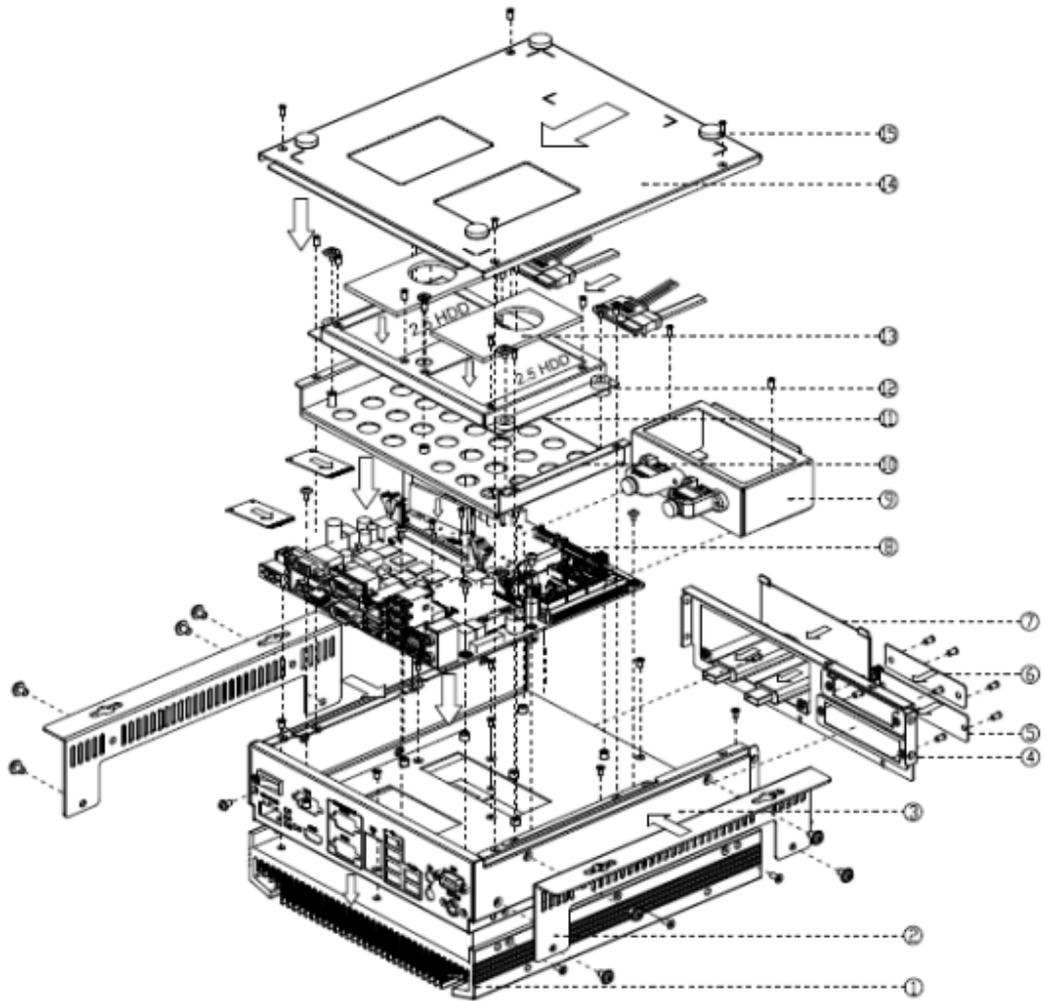


图 1.2: ITA-3630 分解图

表 1.3: 零件列表

1	散热片	9	USB 支架
2	机箱脚架	10	硬盘支架
3	机箱	11	2.5" 硬盘固定架
4	前面板	12	硬盘缓冲胶垫
5	空白挡片	13	散热硅胶
6	空白挡片 (CF)	14	机箱顶盖
7	USB 挡片	15	橡胶脚垫
8	主机板		

## 第 2 章

### 硬件安装

内容包括：

- 简介
- 跳线和接口
- I/O 接口

## 2.1 简介

以下章节介绍了内部跳线设置和外部接口针脚分配信息，用于实现应用集成。

## 2.2 跳线和接口

### 2.2.1 跳线描述

用户可根据需要通过设置跳线配置 ITA-3630。跳线是用来连通电路的金属桥。它包括 2 个金属针脚和一个跳线帽（里面是金属夹片，外部是起保护作用的塑料套）。跳线帽可套住针脚将其连成通路。移走跳线帽则会断开线路。有时，一个跳线具有 3 个针脚，分别为针 1、2、3。这种情况下，用户可以任意选择连接针脚 1、2 或者针脚 2、3。



设备的跳线设置如下图所示：



进行跳线设置时，使用针鼻钳可能会有所帮助。若用户对应用的最佳硬件配置产生任何疑问，请在进行更改前联系当地的分销商或销售代表。通常情况下，用户仅需要一根标准电缆进行大多数连接。

## 2.2.2 跳线和接口位置图

板卡带有一些接口和跳线，可供用户根据应用需要进行系统配置。每个接口和跳线的功能如下表所示。跳线和接口在电路板的位置参考图 2.1。

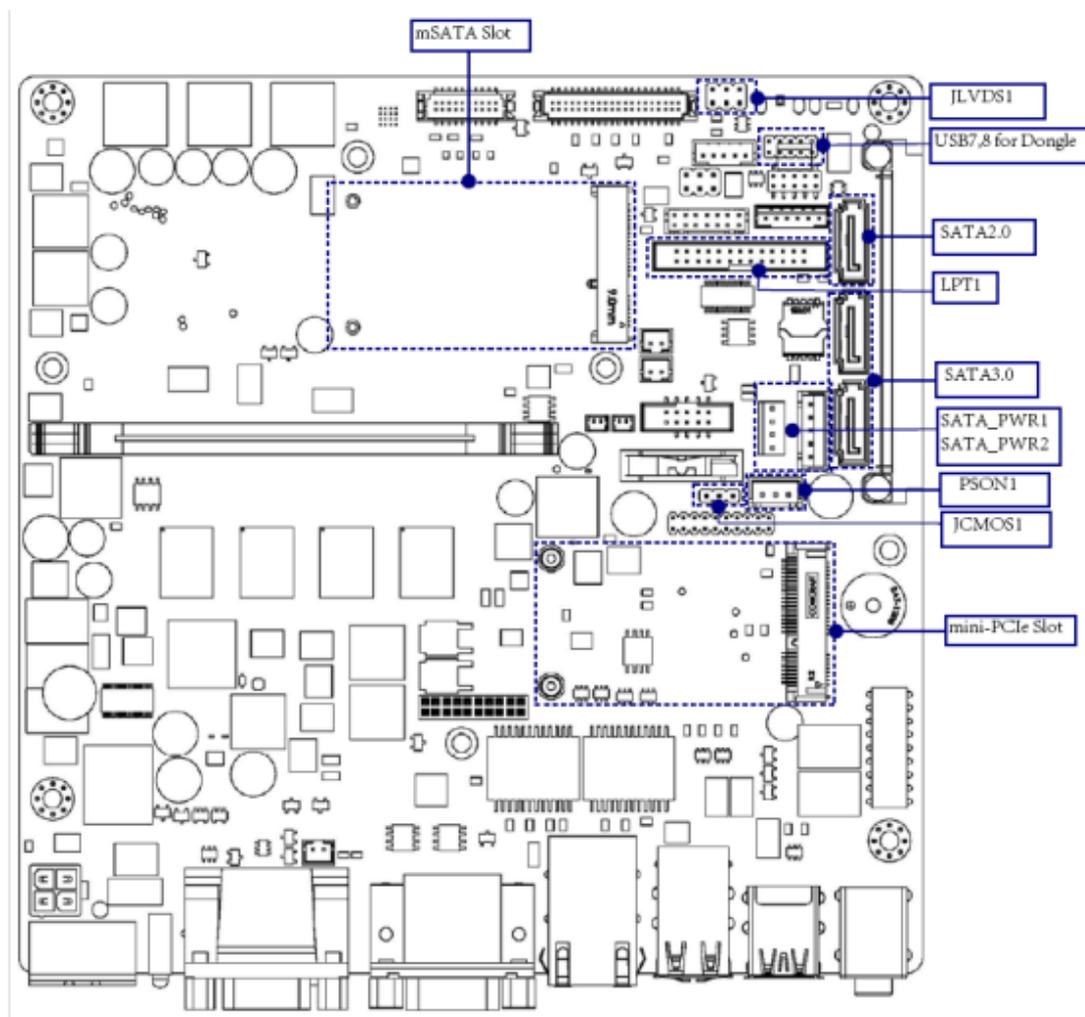


图 2.1: 主板接口及跳线位置图

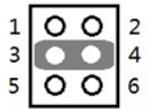
表 2.1: 跳线

名称	功能
JLVDS1	LVDS 电压设定
JCMOS1	清除 CMOS 设定
PSON1	开机模式设定
VCCGPIO1	GPIO 电压设定

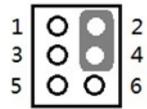
表 2.2: JLVDS1: LVDS 电压设定

闭合针脚	设置
3-4	+V12
2-4	+V5
4-6	正常 (+V3.3)*

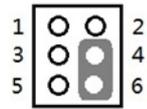
\* 代表默认设置



适用于 12V



适用于 5V

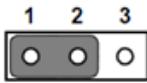


适用于 3.3V

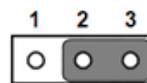
**表 2.3: JCMOS1: 清除 JCMOS 设定**

闭合引脚	设置
1-2	默认 *
2-3	清除 CMOS 设定

\* 代表默认设置



默认设置

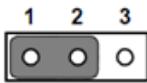


清除 CMOS 设定

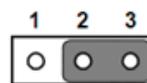
**表 2.4: PSON1: 开机模式设定**

闭合引脚	设置
1-2	AT 模式
2-3	ATX 模式 *

\* 代表默认设置



AT 模式

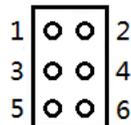


ATX 模式

**表 2.5: VCCGPIO1: GPIO 电压设定**

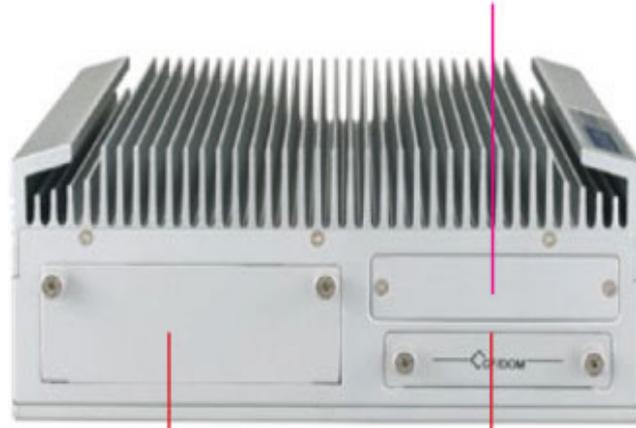
闭合引脚	设置
1-3	+V5_SB
2-4	正常 (+V3.3_SB)*
3-5	+V5
4-6	+V3.3

\* 代表默认设置



## 2.3 I/O 接口

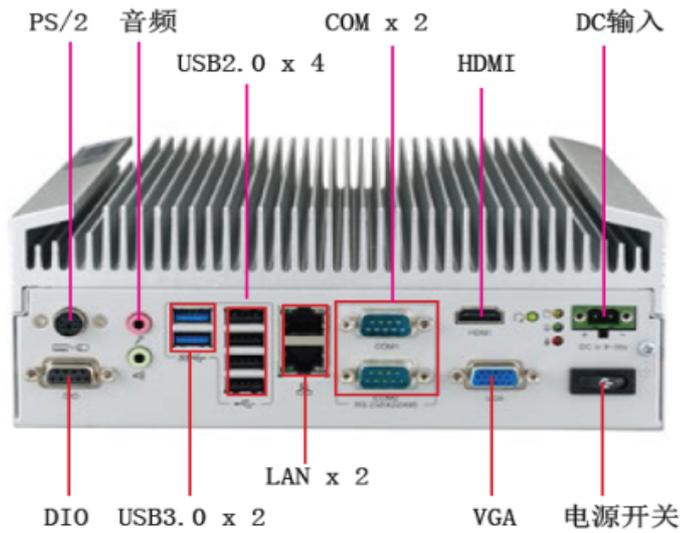
预留I/O接口，可直接通过电缆转接（如LPT/LVDS/COM等）



2 x USB2.0接口支架

易插拔CF/DO模块

图 2.2: ITA-3630 后视图



PS/2

音频

COM x 2

DC输入

USB2.0 x 4

HDMI

DIO

USB3.0 x 2

LAN x 2

VGA

电源开关

图 2.3: ITA-3630 前视图

### 2.3.1 COM 端口

ITA-3630 提供了一个集成了 2 个 D-sub 9 针 RS-232/RS-422/RS-485 接口，默认设置 RS232。

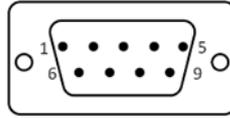


表 2.6: COM 接口针脚定义

针脚	RS-232	RS-422	RS-485
	信号名	信号名	信号名
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

### 2.3.2 HDMI 接口

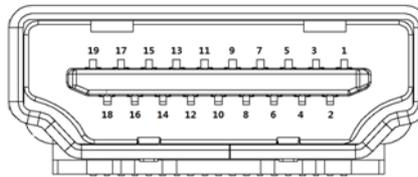


表 2.7: VGA 接口针脚定义

针脚	信号	针脚	信号
1	TMDS Data2+	11	TMDS Clock Shiled
2	TMDS Data2 Shiled	12	TMDS Clock-
3	TMDS Data2-	13	CEC
4	TMDS Data1+	14	预留
5	TMDS Data1 Shiled	15	SCL
6	TMDS Data1-	16	SDA
7	TMDS Data0+	17	DDC/CEC Ground
8	TMDS Data0 Shiled	18	+5V Power
9	TMDS Data0-	19	Hot Plug Detect
10	TMDS Clock+		

### 2.3.3 VGA 接口

ITA-3630 提供 1 个 D-SUB 15 针的母型接口，可支持最大分辨率 2048 x 1563。

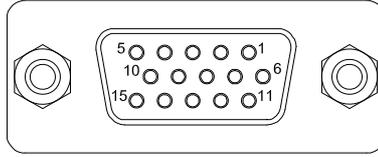


表 2.8: VGA 接口针脚定义

针脚	信号	针脚	信号
1	Red(红)	9	+5V
2	Green(绿)	10	GND
3	Blue(蓝)	11	NC
4	NC	12	DDC-DATA
5	GND	13	H-SYNC
6	GND	14	V-SYNC
7	GND	15	DDC-CLK
8	GND		

### 2.3.4 USB2.0 接口

ITA-3630 提供 1 组共 4 个 USB2.0 接口，其符合 USB UHCI Rev. 2.0 标准，可以通过 BIOS 来禁用。

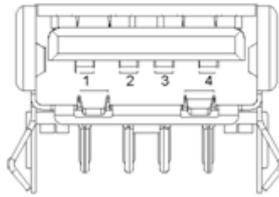


表 2.9: USB2.0 针脚定义

针脚	信号	针脚	信号
1	+V5 (VCC)	3	USB DATA+
2	USB DATA-	4	GND

### 2.3.5 USB3.0 接口

ITA-3630提供1组共2个 USB3.0接口，其符合USB UHCI Rev. 3.0 标准，可以通过BIOS来禁用。

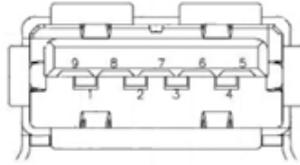


表 2.10: USB3.0 针脚定义

针脚	信号	针脚	信号
1	+V5 (VCC)	6	StdA_SSRX+
2	D-	7	GND_DRAIN
3	D+	8	StdA_SSTX-
4	GND	9	StdA_SSTX+
5	StdA_SSRX-		

### 2.3.6 DIO 接口

ITA-3630 提供一个 8 bit DI/O, D-sub 9 针的公头接口，无隔离保护。

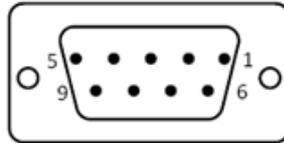
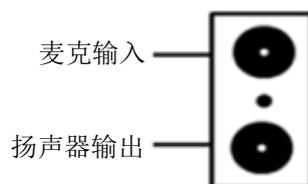


表 2.11: DIO 接口针脚定义

针脚	信号名	针脚	信号名
1	GPI00	6	GPI04
2	GPI01	7	GPI05
3	GPI02	8	GPI06
4	GPI03	9	GPI07
5	GND		

### 2.3.7 Audio 接口

ITA-3630 提供一个集成麦克风输入 / 扬声器输出的音频接口。



### 2.3.8 PS/2 接口

ITA-3630 提供一个 PS2 的键盘和鼠标接口。



表 2.12: PS/2 接口针脚定义

针脚	信号名	针脚	信号名
1	KB_Z_DAT	4	+V5_PS2
2	MS_Z_DAT	5	KB_Z_CLK
3	GND	6	MS_Z_CLK

### 2.3.9 LAN 接口

ITA-3630 提供有 2 个 RJ45 以太网连接器，带有 LED 灯显示网络状态。

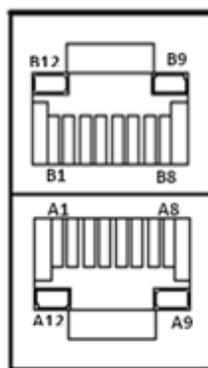


表 2.13: 以太网接口针脚定义

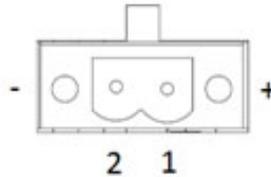
针脚	信号名	信号名
A1/B1	MDI00+	MDI00+
A2/B2	MDI00-	MDI00-
A3/B3	MDI01+	MDI01+
A4/B4	MDI02+	MDI02+
A5/B5	MDI02-	MDI02-
A6/B6	MDI01-	MDI01-
A7/B7	MDI03+	MDI03+
A8/B8	MDI03-	MDI03-
A9/B9	LED GREEN-	LED GREEN-
A10/B10	LED GREEN+	LED GREEN+
A11/B11	1000M LED	1000M LED
A12/B12	100m/10M LED	100m/10M LED

**表 2.14: 网口 LED 灯色状态**

千兆连接	绿色（左上方 LED）
百兆连接	橙色（左上方 LED）
连接状态	绿色闪烁（右上方 LED）

### 2.3.10 Phoenix 端子接口

ITA-3630 提供一个 2pin 的凤凰端子的电源接口，提供 9 V ~ 36V 的电源输入。

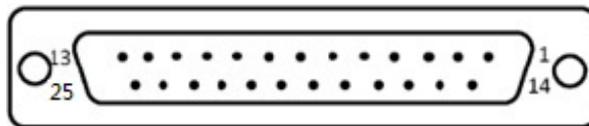


**表 2.15: 电源接口**

针脚	信号名	针脚	信号名
1	正极	2	GND

### 2.3.11 LPT 接口

ITA-3630 提供了一个并行接口（可选配置），可以通过线缆引出至前面板预留位置。



**表 2.16: LPT 接口针脚定义**

针脚	信号名	针脚	信号名
1	LPT_z_STB#	14	LPT_AFD#
2	LPT_z_PD0	15	LPT_ERR#
3	LPT_PD1	16	LPT_INIT#
4	LPT_PD2	17	LPT_SLIN#
5	LPT_PD3	18	GND
6	LPT_PD4	19	GND
7	LPT_PD5	20	GND
8	LPT_PD6	21	GND
9	LPT_PD7	22	GND
10	LPT_ACK#	23	GND
11	LPT_BUSY	24	GND
12	LPT_PE	25	GND
13	LPT_SLCT		

### 2.3.12 LVDS 接口

ITA-3630 提供一个 LVDS 接口（可选配置），可以通过线缆引出至前面板预留位置。

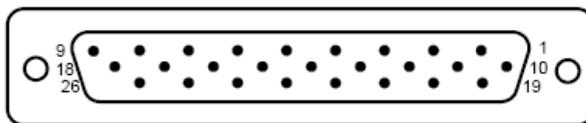


表 2.17: LVDS 接口针脚定义

针脚	信号名	针脚	信号名
1	TXL0+	2	TXL1+
3	TXL2+	4	TXL3+
5	TXU0+	6	TXU1+
7	TXU2+	8	TXU3+
9	TXU3-	10	TXLCLK-
11	TXL0-	12	TXL1-
13	TXL2-	14	TXL3-
15	TXU0-	16	TXU1-
17	TXU2-	18	TXUCLK+
19	TXLCLK+	20	LVCC
21	BLEENABLE	22	GND
23	GND	24	GND
25	GND	26	GND

### 2.3.13 系统状态 LED 指示灯

后面板右上 LED 用于指示系统健康和活跃状态。

LED 定义的详细信息请参考下表。

项目	LED	状态	色彩	描述
1	PWR	亮起	绿色	系统电源接通，系统安全
2	HDD	亮起	黄色	数据正在通过 SATA 进行接收 / 传输
		熄灭		没有数据正在通过 SATA 进行接收 / 传输



# 第 3 章

## 系统安装

内容包括：

- 安装主板 mini-PCIe 卡和 mini SATA 卡
- 安装 USB Dongle
- 安装硬盘模组
- 安装脚架
- 安装注意事项

## 3.1 系统安装简介

以下步骤将指导用户组装 ITA-3630 中的各个模块。

### 3.1.1 安装主板 mini-PCIe 卡和 mini SATA 卡

1. ITA-3630 主板有提供 mini-PCIe 和 mSATA 插槽各一个，插槽有 label 标记，使用时请注意区分；
2. 将 mini-PCIe 和 mSATA 卡分别插进插槽内用两颗螺丝固定即可。

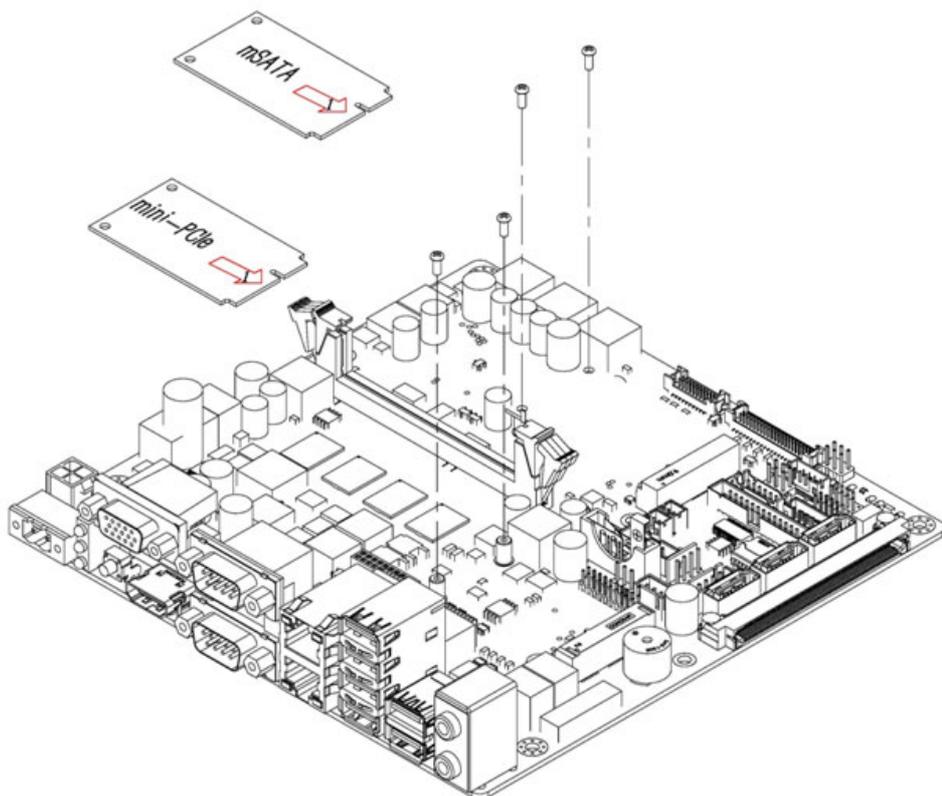


图 3.1：安装 mSATA 和 mini-PCIe 示意图

### 3.1.2 安装 USB Dongle

1. 将机器前面板 USB 档片处的手锁螺丝拧开，打开面板盖；
2. 将所用 USB Dongle 插进 USB 插槽中；
3. 将 USB 面板盖合上，并将手锁螺丝拧好。

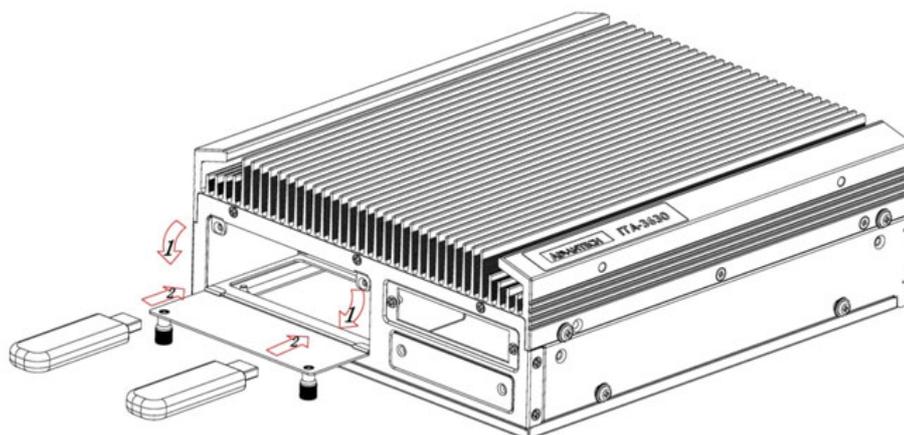


图 3.2：安装 USB Dongle

### 3.1.3 安装硬盘模组

ITA-3630 预留了安装双 HDD 模组和 HDD+CF 卡模组的空间。

请参考以下双 HDD 模组的指导安装：

1. 在 HDD 固定架上填入 4 颗 rubber ；
2. 将两块 HDD 用 8 颗螺丝所附在 HDD 固定架上，并贴上散热泥；
3. 将 HDD 固定架放入 HDD 支架，用 4 颗螺丝固定；
4. 将 HDD 支架安装在机箱上并用螺丝固定。

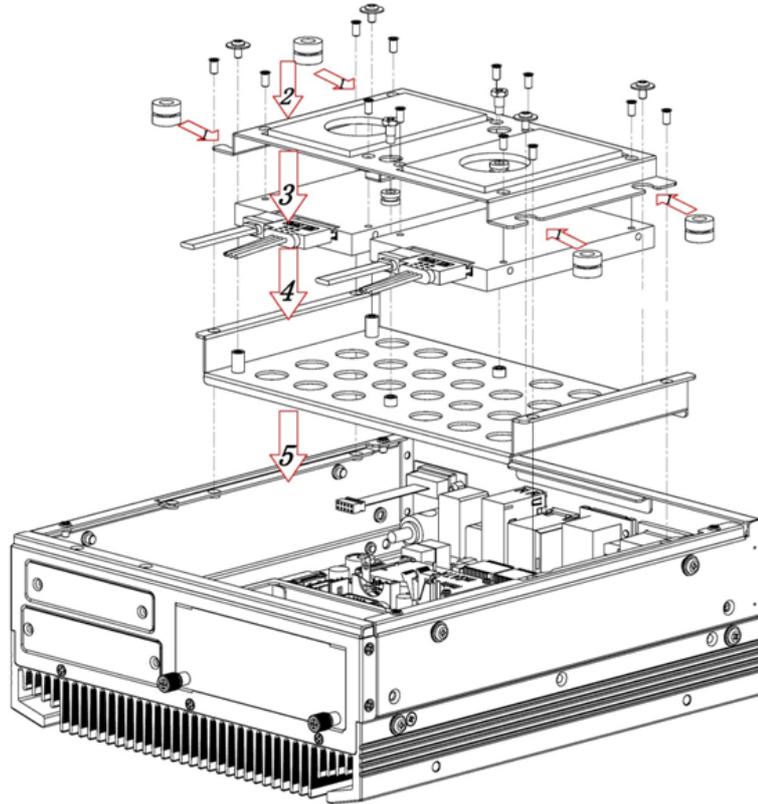


图 3.3： 安装双 HDD 模组

请参考以下 HDD+CF 模组的指导安装：

1. 先在 HDD 固定架上填入 rubber ；
2. 将 HDD 用 4 颗螺丝所附在固定架上，并贴上散热泥；
3. 将 HDD 固定架扣入 HDD 支架，用螺丝固定好；
4. 将 HDD 支架安装在机壳上；
5. 将 CF 模组小板锁在 CF 抽取支架上；
6. 将 CF 卡装入 CF 小板上并用 CF 固定塑料壳固定好 CF 卡；
7. 将 CF 抽取架插入 HDD 支架，拧好手动螺丝。

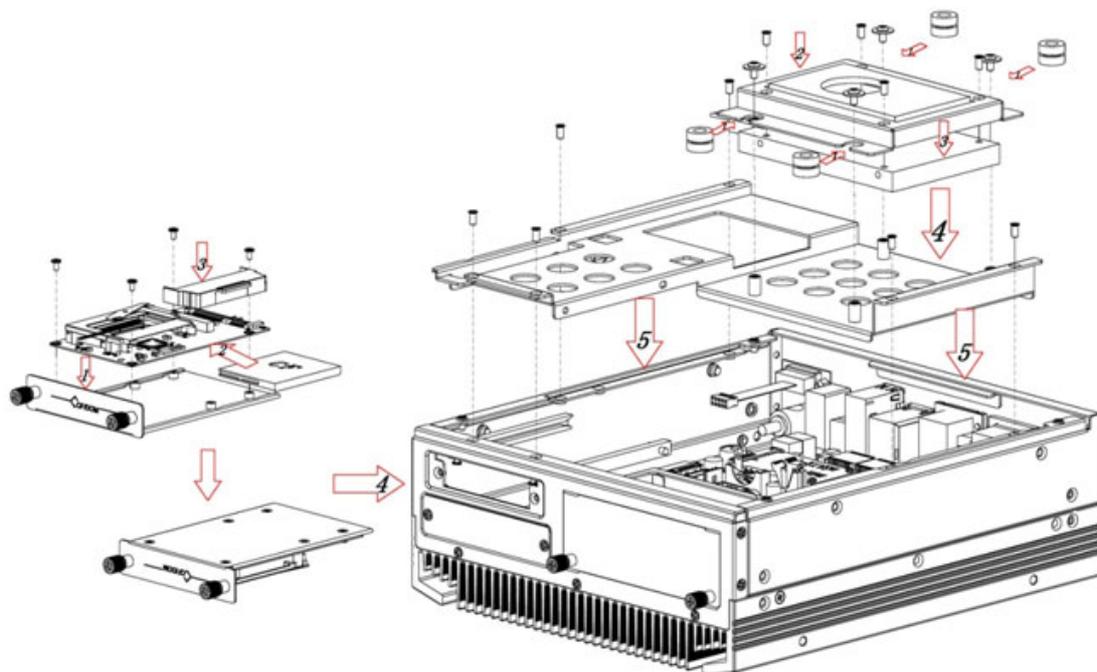


图 3.4： 安装 HDD+CF 模组

### 3.1.4 安装脚架

将 2 个脚架与机箱侧壁处的螺丝孔对好，用螺丝锁好即可。

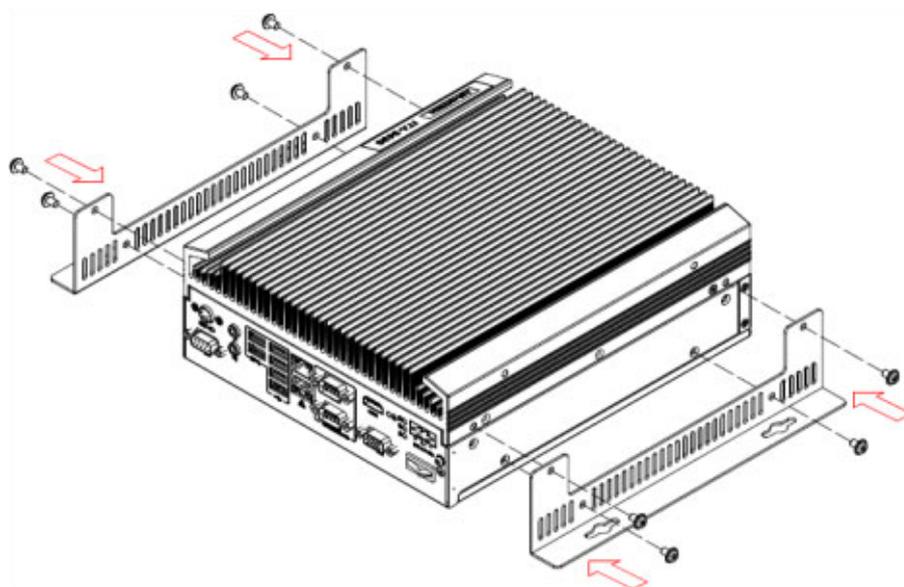


图 3.5： 安装脚架示意图

### 3.1.5 安装注意事项

设备运行时，散热片上方 30mm 内请尽量避免有覆盖物以防影响空气流动。

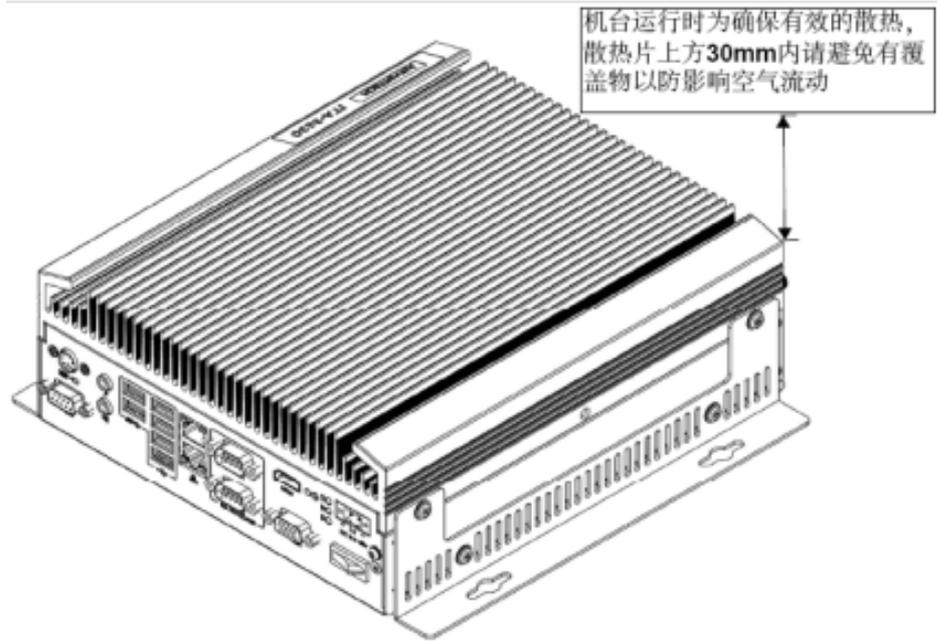


图 3.6：桌面式安装

悬挂式安装设备时，保持鳍片方向平行于空气对流方向可以有效提高散热效率。

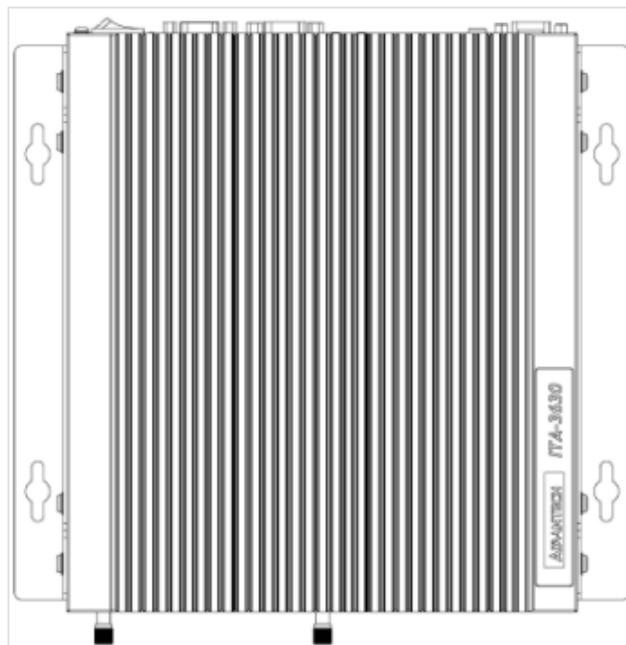


图 3.7：悬挂式安装



# 第 4 章

BIOS 设置

## 4.1 简介

AMI BIOS 已被集成到多种工业级和嵌入式主板中，在近年来非常受欢迎。本章介绍如何配置 BIOS，使之适用于 ITA-3630 系列产品。用户可在 AMI BIOS 设置实用程序中更改 BIOS 设置、实现对 ITA-3630 系列特殊性能的控制。设置程序内包含多个菜单，允许用户对计算机性能进行设置。本章介绍 ITA-3630 系列 BIOS 设置的基本知识。

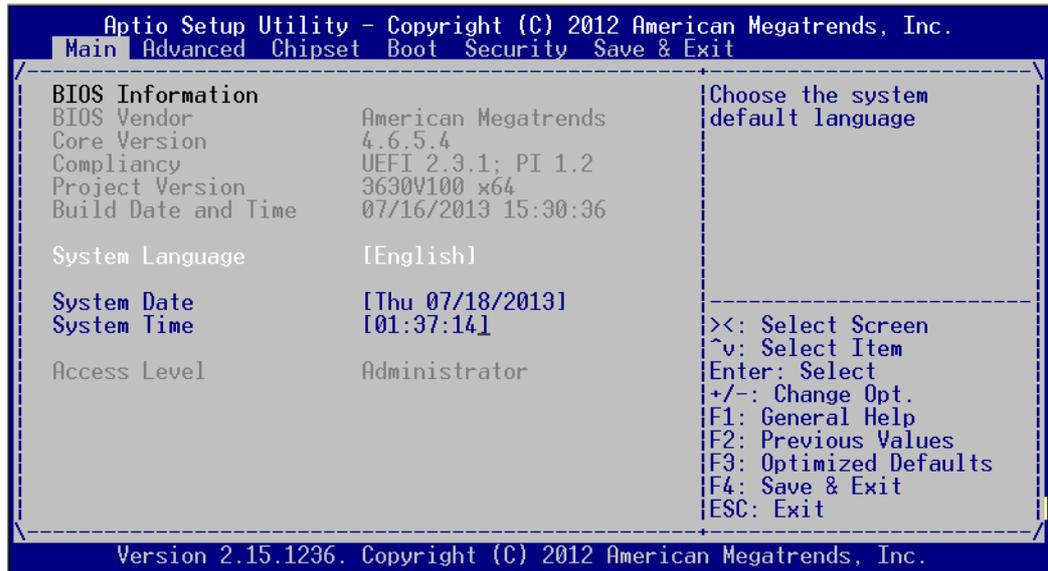


图 4.1：设置程序初始页面

BIOS ROM 带有内置的设置程序，允许用户修改基本系统配置信息。这些信息保存在 BIOS ROM，因此在电源关闭时仍不会丢失。

## 4.2 进入设置界面

开启计算机后，屏幕上将出现 POST（上电自检），显示 BIOS 和 CPU 信息。按 <Del> 键即可进入 BIOS 设置界面。

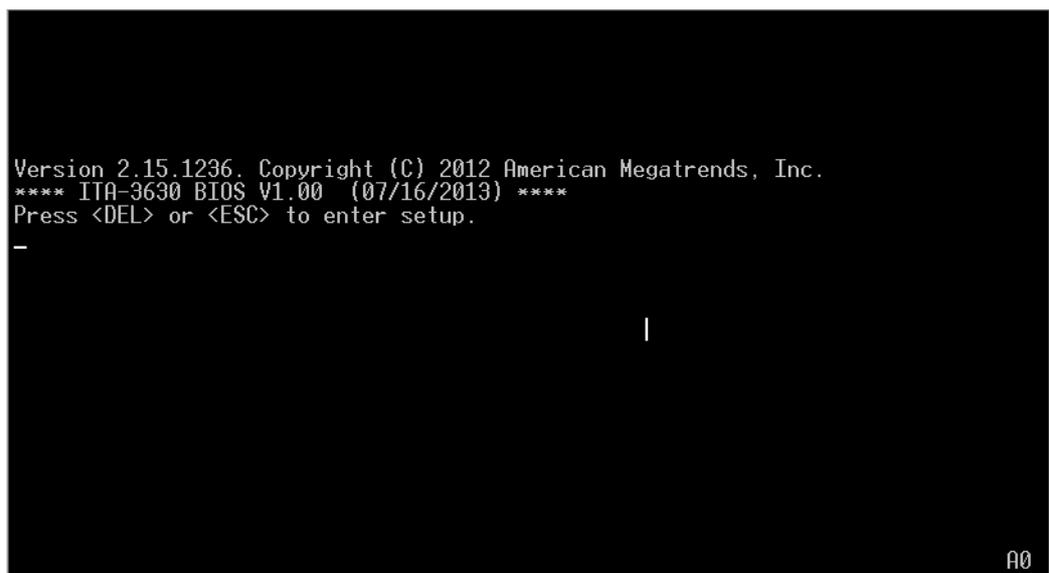


图 4.2：Post 页面

## 4.3 主设置界面

首次进入设置实用程序时，即会进入主设置页面。点击“Main”标签即可随时返回主菜单。本节共介绍共有 2 个主设置选项。BIOS 主设置页面如下图所示：

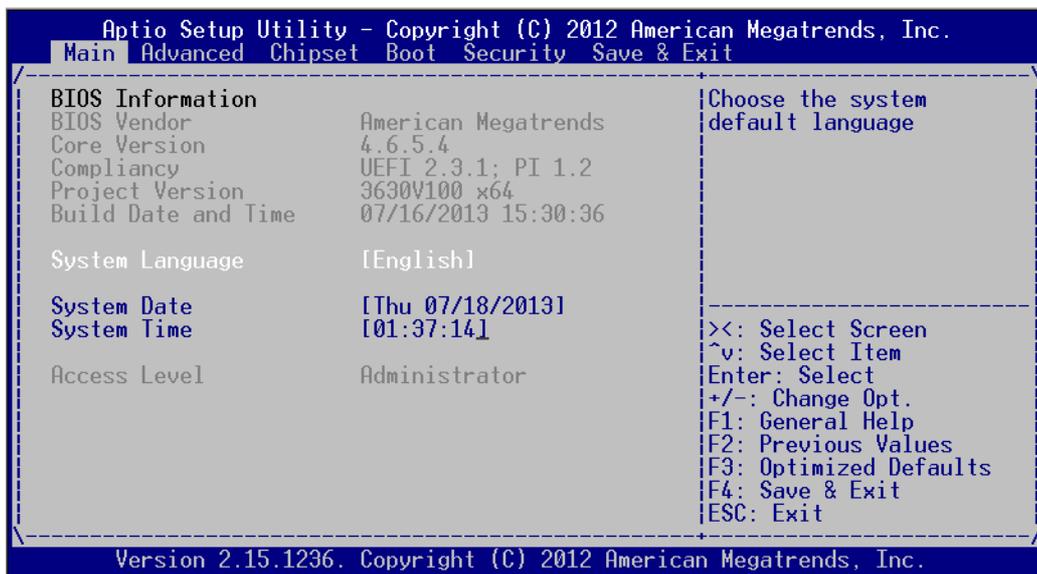


图 4.3：主设置页面

BIOS 设置主菜单由 2 部分构成。左栏显示的是可以配置的所有项。灰色的为不可配置的项，蓝色的则相反。右边栏为箭头图例。箭头上方的空白区域是为文字信息预留的。如果在左边栏选择了某项，该项将以加亮白色字体显示，且在预留的文字信息处显示简介文字。

### 4.3.1 System Time/System Date

此项可用于改变系统时间和日期。用户可使用方向键选中系统时间或系统日期，并可通过键盘输入新值。按 Tab 键或方向键可在各项间进行切换。日期的格式 MM/DD/YY，时间的格式为 HH: MM: SS。

## 4.4 高级 BIOS 特性设置

从 ITA-3630 系列的设置界面点击 “Advanced” 标签即可进入高级 BIOS 设置界面。用户可选择左边边框中的项进入相应项的子菜单，如 CPU Configuration。用户可使用方向键来选中相应的设置项。所有的高级 BIOS 设置选项都在此节中进行描述，如下图所示。子菜单将在后面进行介绍。

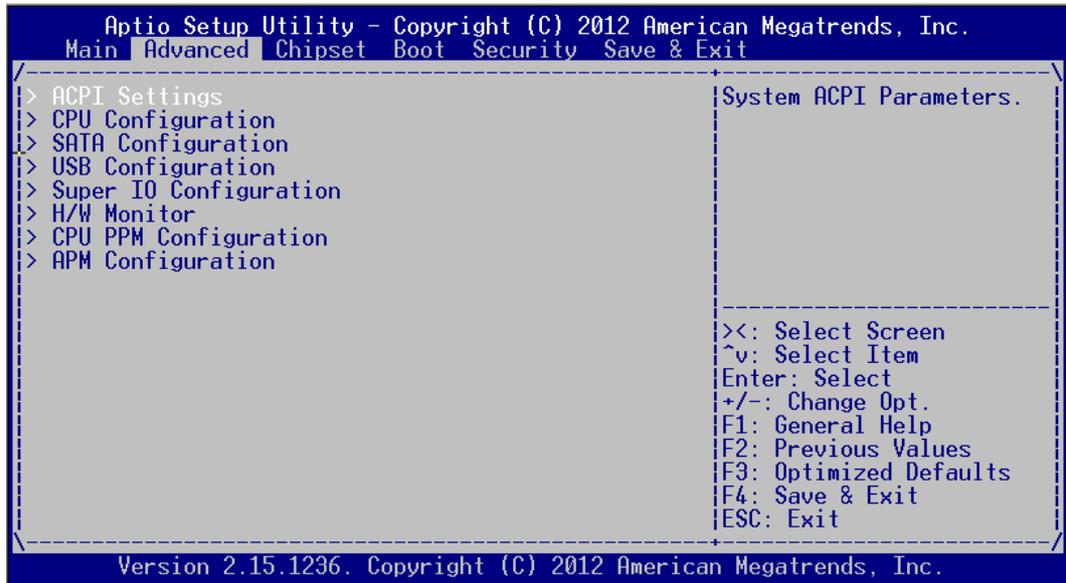


图 4.4：高级 BIOS 特性设置

### 4.4.1 ACPI 设置

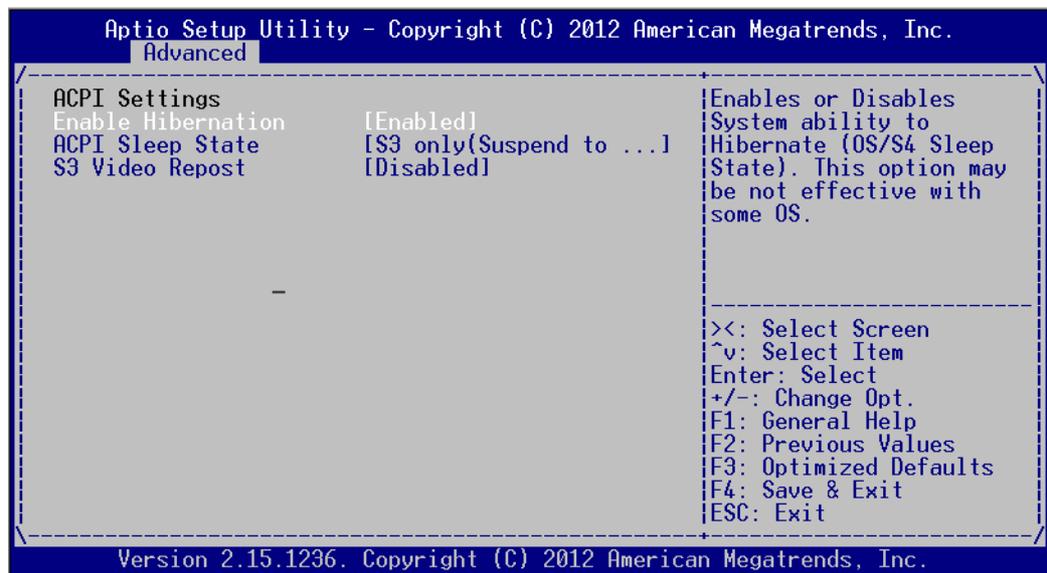


图 4.5：ACPI 设置

- **Enable Hibernation**  
此项允许使用使用者 Enabled 或者 Disabled 休眠状态。
- **ACPI Sleep State**  
为系统休眠选择 ACPI 状态。
- **S3 Video Repost**  
系统从 S3/SRT 恢复时是否调用 VBIOS。默认设置为 “Disabled”。

## 4.4.2 CPU Configuration

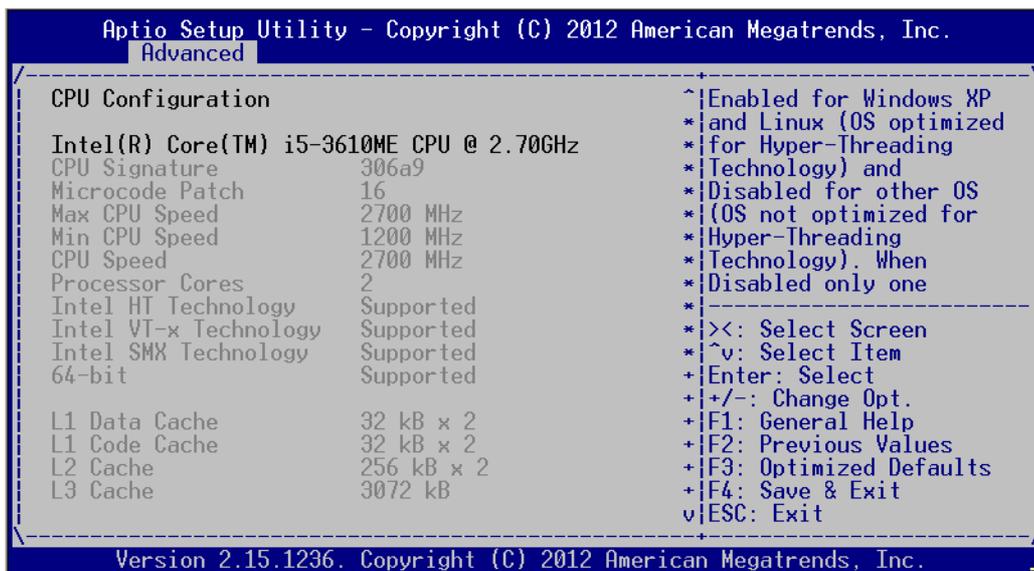


图 4.6: CPU Configuration(1)

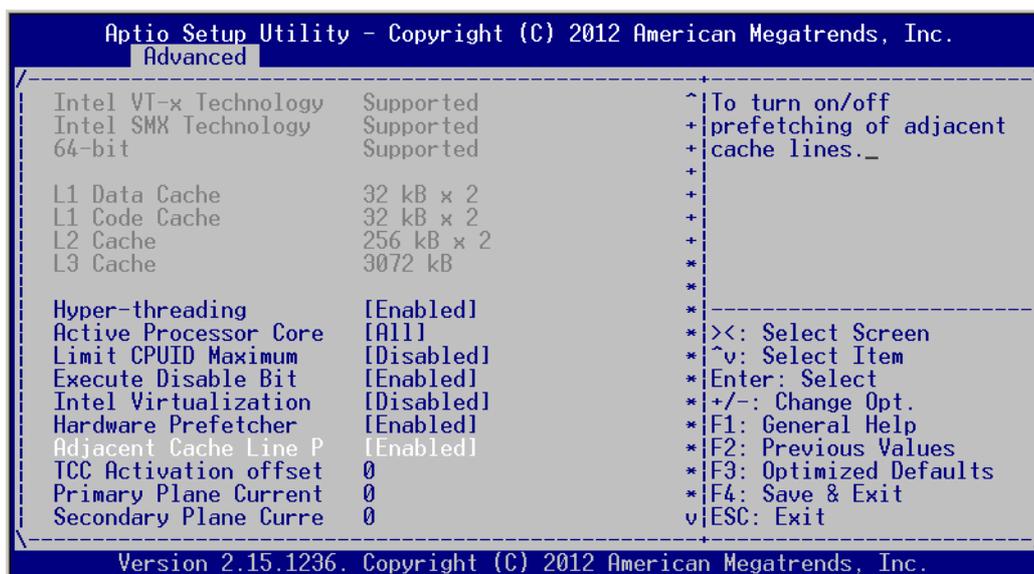


图 4.7: CPU Configuration(2)

- **Hyper-threading**  
此项允许用户启用或禁用 Intel 超线程技术，默认设置为“Enabled”。
- **Active Processor Core**  
此项允许用户设置多核 CPU 的核心数量，默认设置为“All”。
- **Limit CPUID Maximum**  
此项允许用户设置最大 CPUID 限制值，默认设置为“Disabled”。
- **Execute Disable Bit**  
项是 Intel 在新一代 CPU 中引入的一项硬件特性，它能帮助 CPU 在某些基于缓冲区溢出的恶意攻击下，实现自我保护，从而避免诸如“冲击波”之类病毒的恶意攻击。默认设置为“Enabled”。
- **Hardware Prefetcher**  
此项允许客户启用或禁用硬件预抓取技术。默认设置为“Enabled”。

- **Adjacent Cache Line Prefetch**  
此项允许客户启用或禁用顺序内存访问。默认设置为“Enabled”。
- **Intel Virtualization Technology**  
此项是 Intel 公司的 CPU 中采用的系统假想化技术。它使得在 1 台 PC 能够运行多个 OS，这种技术使得处理器具有虚拟化技术，使用 Vanderpool Technology 技术，允许一个平台运行多操作系统并且应用程序都可以在独立的空间内运行。默认设置为“Enabled”。

### 4.4.3 SATA Configuration

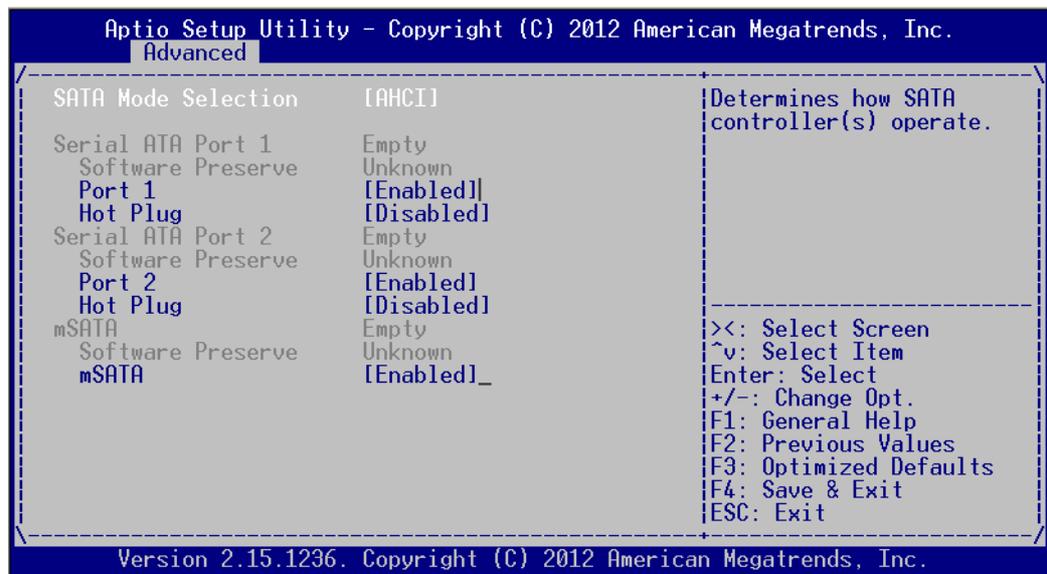


图 4.8: SATA Configuration

- **SATA Mode Selection**  
此项用于设置 SATA Controller(s) 配置模式，默认设置是 AHCI。

#### 4.4.4 USB Configuration

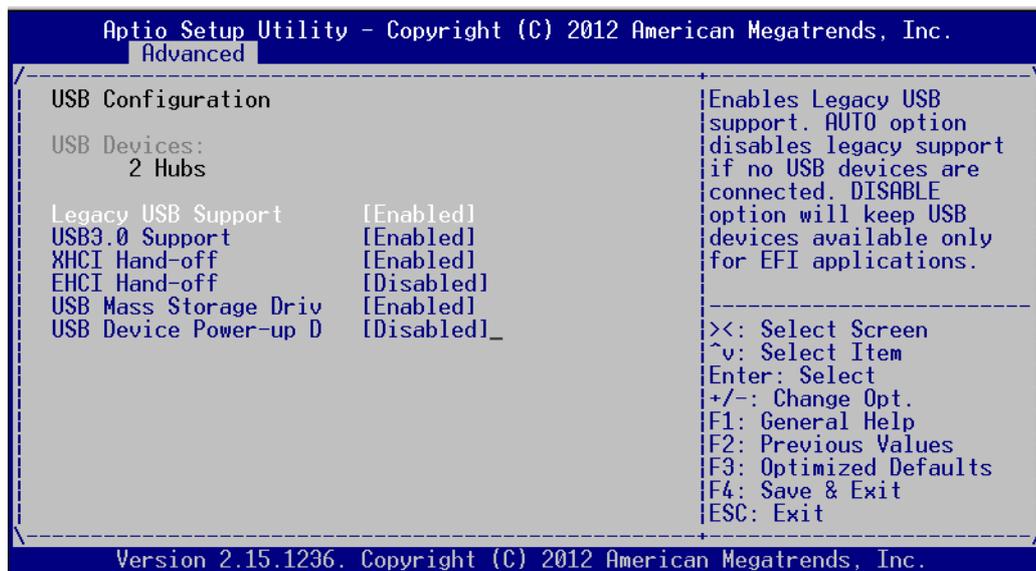


图 4.9: USB Configuration

- **Legacy USB Support**  
此项用于设置启用对传统 USB 的支持。如果未连接任何 USB 设备，则自动选项为禁用，默认设置为“Enabled”。
- **USB3.0 Support**  
允许客户设置启用或者禁用 USB3.0 (XHCI)，默认设置为“Enabled”。
- **XHCI Hand-off**  
启用允许无 XHCI Hand-off feature 用户操作系统，默认设置为“Enabled”。
- **EHCI Hand-off**  
启用允许无 EHCI Hand-off feature 用户操作系统，默认设置为“Disabled”。
- **USB Mass Storage Driver Support**  
此项用于设置所连接的 USB 设备的具体类型。
- **USB Device Power-up Delay**  
这是设置 USB 设备向主控制器报到的最大延迟时间，默认设置为“Disabled”。

## 4.4.5 Super IO Configuration

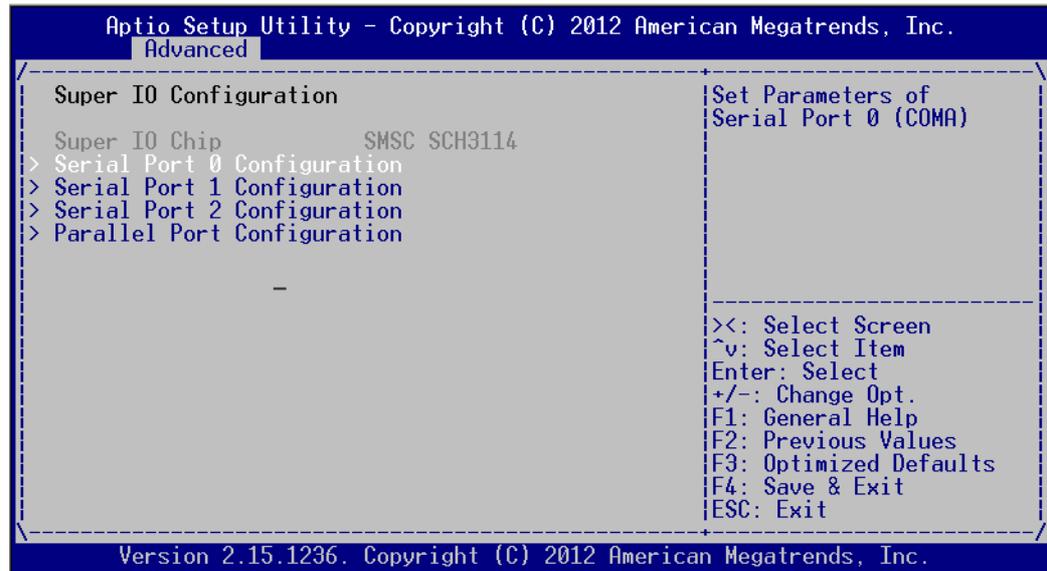


图 4.10: Super IO Configuration

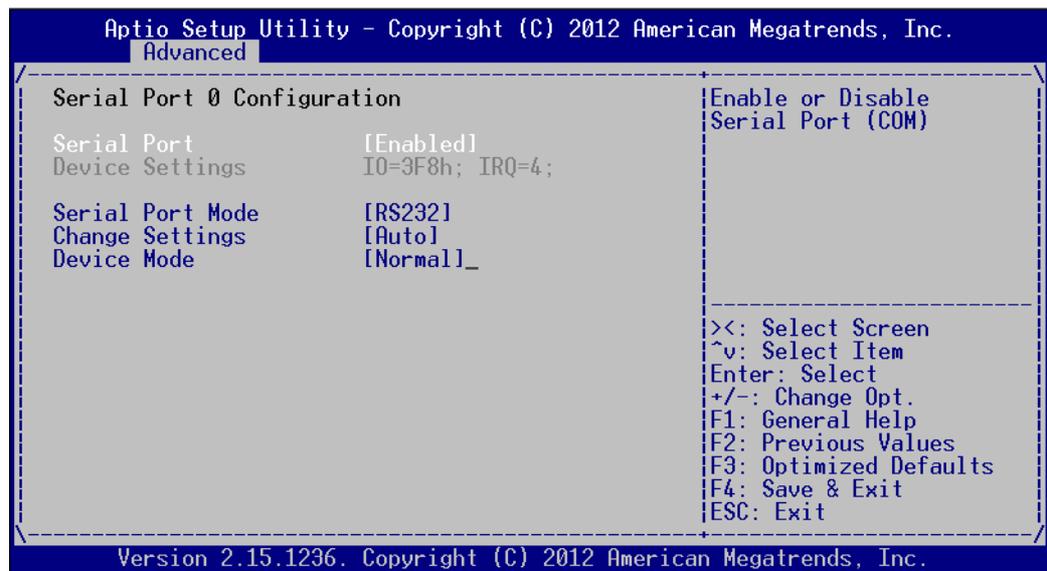


图 4.11: Serial Port Configuration

### Serial Port 0 Configuration

- **Serial Port**  
用于设置打开或关闭串行接口，默认设置为“Enabled”。
- **Serial Port Mode**  
用于设置 Serial Port，选项为 RS232/RS422/RS485，默认设置为“RS232”。
- **Change settings**  
IO=3F8h; IRQ=4;  
IO=3F8h; IRQ=3, 4, 5, 6, 7, 10, 11, 12;  
用于选择 IO 地址和 IRQ，改变串口设置。
- **Device mode**  
选择 High Speed 或者 Normal mode 模式，默认设置为“Normal mode”。

#### 4.4.6 H/W Monitor

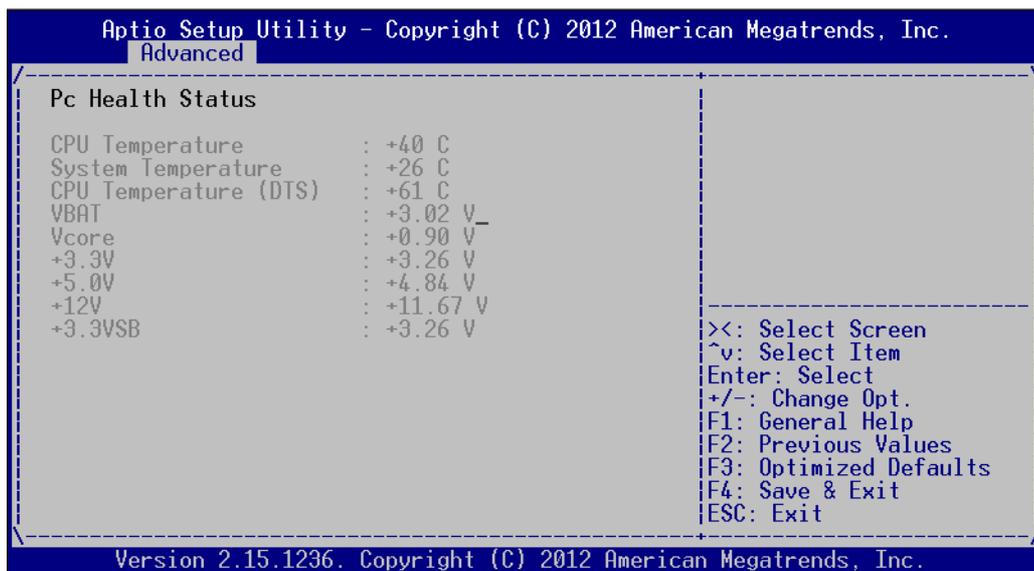


图 4.12: 设置 H/W Monitor 页面

##### ■ PC Health Status

此项用于做硬件安全侦测，BIOS 将显示当前的系统温度，CPU 温度，以及其他相关的电压值。以上参数都有一定的范围，系统不可以超过这些范围运行。

#### 4.4.7 CPU PPM Configuration

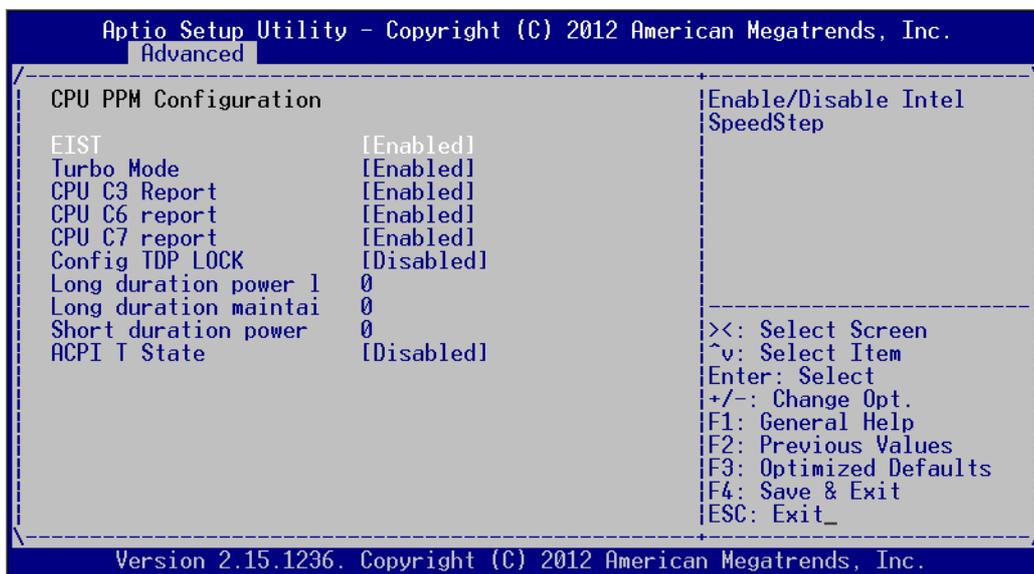


图 4.13: CPU PPM Configuration

- **EIST**  
用于设置 CPU SpeedStep 功能，默认设置 “Enabled”。
- **Turbo Mode**  
用于设置 CPU Turbo mode 功能，默认设置 “Enabled”。
- **CPU C3/C6/7 Report**  
用于设置 CPU C-State 功能，默认设置 “Enabled”。
- **Config TDP LOCK**  
用于设置 Config TDP LOCK 功能，默认设置 “Disabled”。
- **ACPI T State**  
用于设置 ACPI T State 功能，默认设置 “Disabled”。

#### 4.4.8 APM Configuration

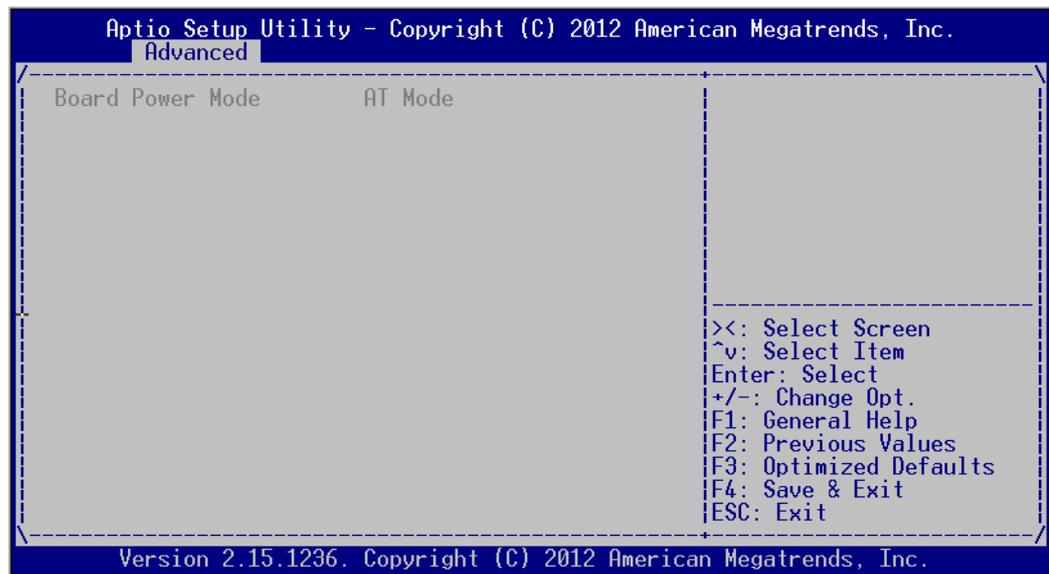


图 4.14: APM Configuration

Board Power Mode

用于设置开机模式：AT 或者 ATX mode

## 4.5 高级芯片组特性设置

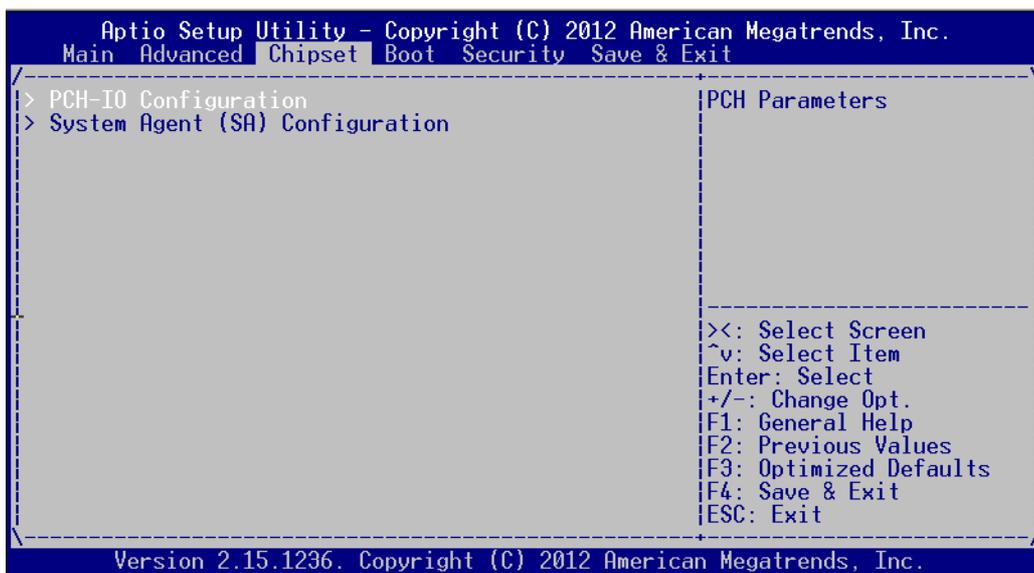


图 4.15: 高级芯片组特性设置

### 4.5.1 PCH-I/O Configuration

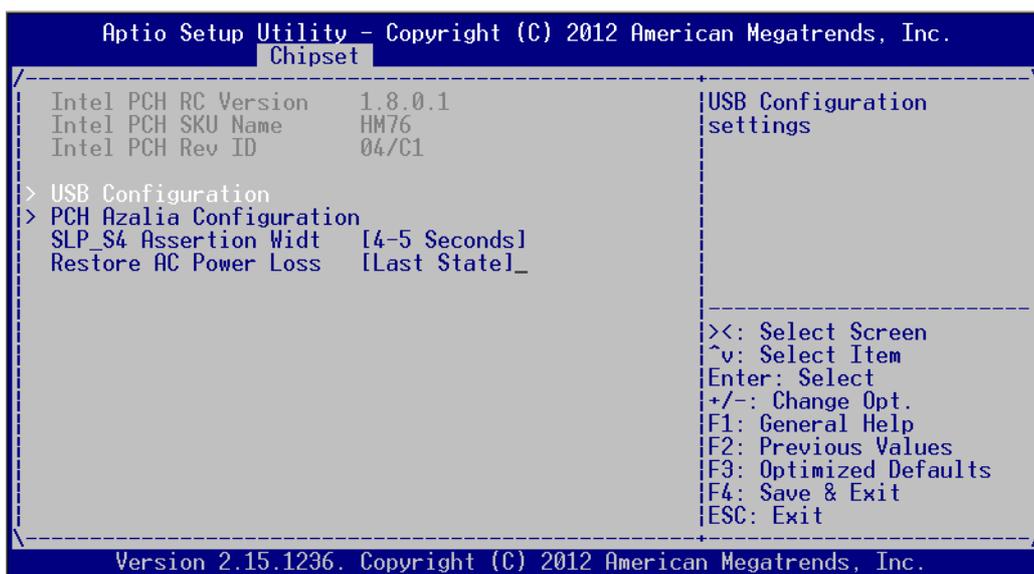


图 4.16: PCH-I/O Configuration

#### 4.5.1.1 SLP\_S4 Assertion Width

用于设置启动时 SLP\_S4# signal 最小延迟，默认设置是“4-5 Seconds”。

#### 4.5.1.2 Restore AC Power Loss

用于设置加电后的开机情况

Power Off: 加电后需按下电源键才能开机。

Power On: 加电后直接开机。

Last State: 加电后恢复到掉电前所在状态。

### 4.5.1.3 USB Configuration

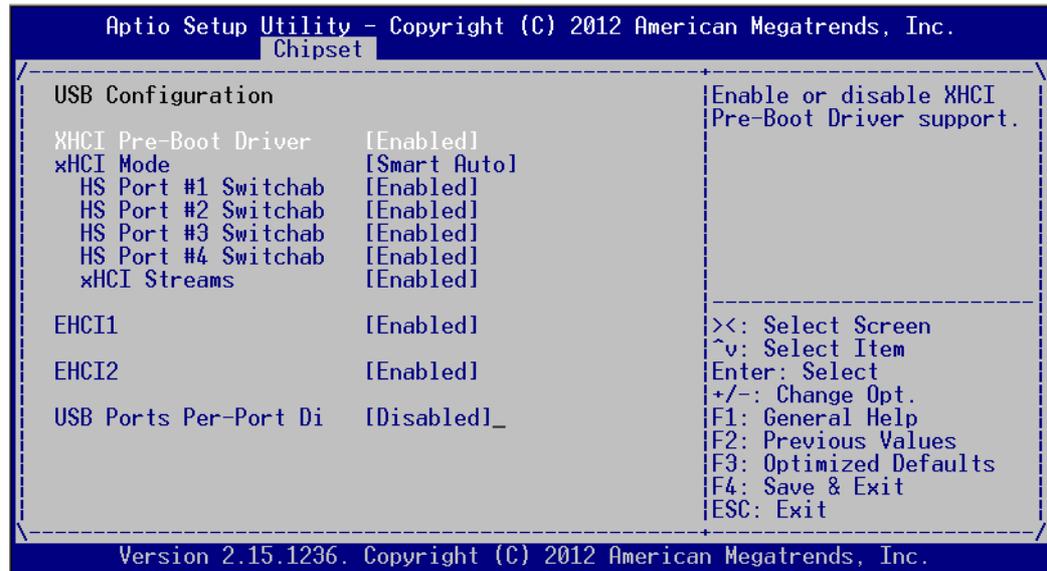


图 4.17: 设置 Chipset USB Configuration

- **XHCI Pre-Boot Driver**  
用于设置 Enable 或者 Disable XHCI Pre-Boot Driver，默认设置为“Enable”。
- **xHCI Mode**  
用于选择 xHCI 控制器模式，默认设置为“Smart Auto”。
- **HS Port #1/#2/#3/#4 Switchable**  
Enable 模式下 BIOS 将端口连接至 EHCI，Disable 模式下 BIOS 将端口连接至 xHCI。  
默认设置为“Enable”。
- **xHCI Streams**  
Enable 或者 disable xHCI Maximum Primary Stream Array Size.
- **EHCI1/EHCI2**  
Enable or disable EHCI #1/EHCI #2，默认设置为“Enable”。
- **USB Ports Per-Port Disable Control**  
用于设置每个 USB 端口的禁用控制。

#### 4.5.1.4 PCH Azalia Configuration

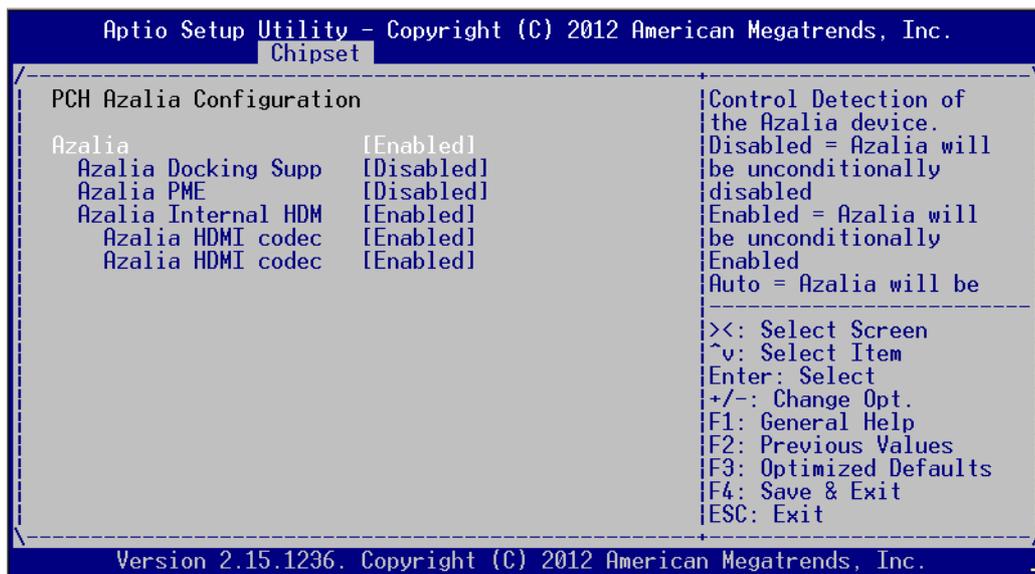


图 4.18: PCH Azalia Configuration

- Azalia

用于控制检测 Azalia device. 默认设置为“Enable”。

Disabled: Azalia will be unconditionally disabled.

Enabled: Azalia will be unconditionally Enabled.

Auto : Azalia will be enabled if present, disabled otherwise.

#### 4.5.2 System Agent (SA) Configuration

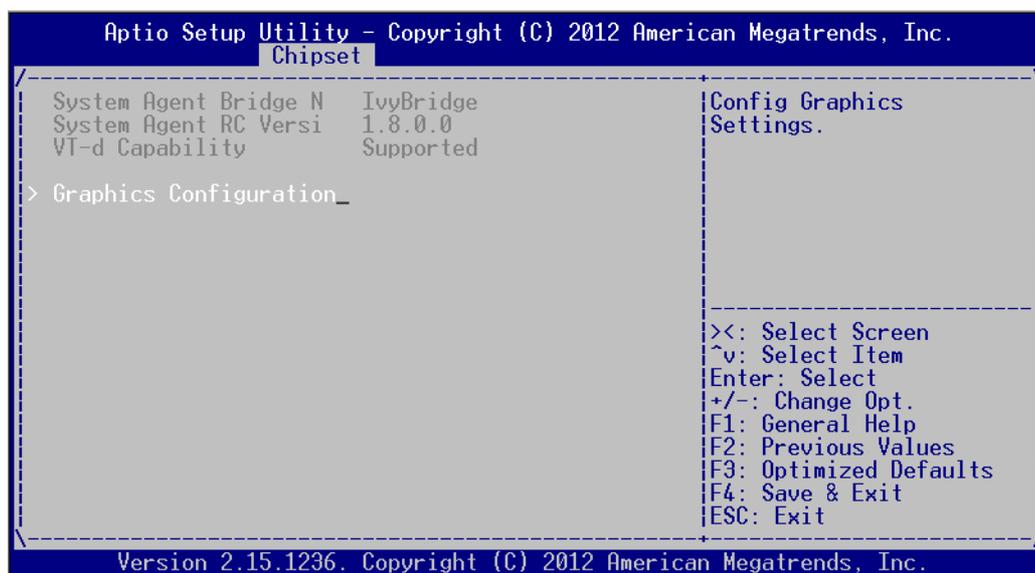


图 4.19: System Agent (SA) Configuration Menu

#### 4.5.2.1 Graphics Configuration

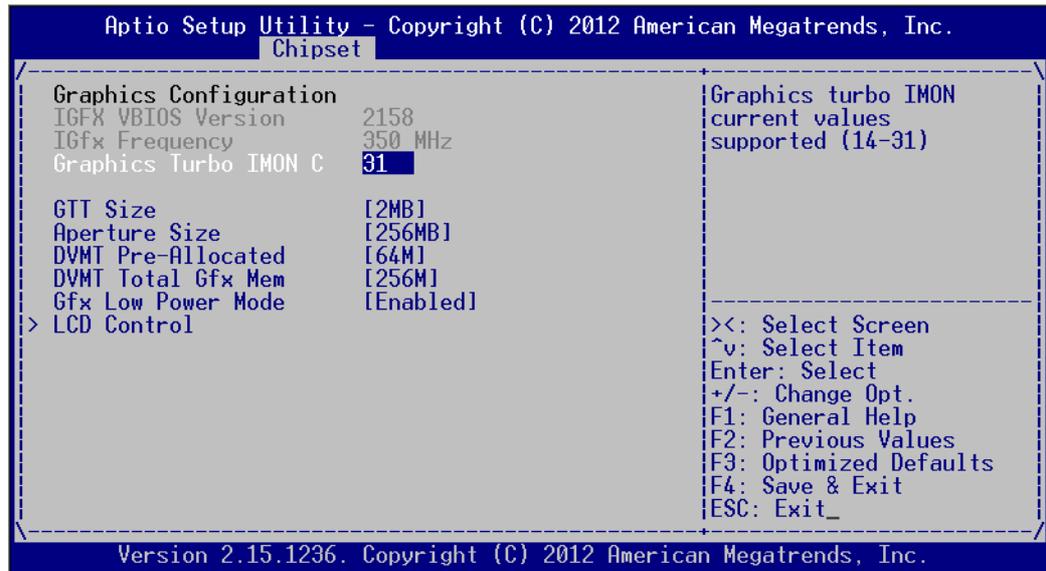


图 4.20: 设置 Graphics Configuration Menu

- **Graphics Turbo IMON Current**  
Graphics turbo IMON current values supported (14-31).
- **GTT Size**  
此项允许用户选择 GTT size。
- **DVMT Total Gfx Mem**  
此项允许用户选择 DVMT Total 内存容量。
- **Gfx Low Power Mode**  
此项允许用户 Enable 或者 Disable Low Power 模式。

#### 4.5.2.2 LCD Control

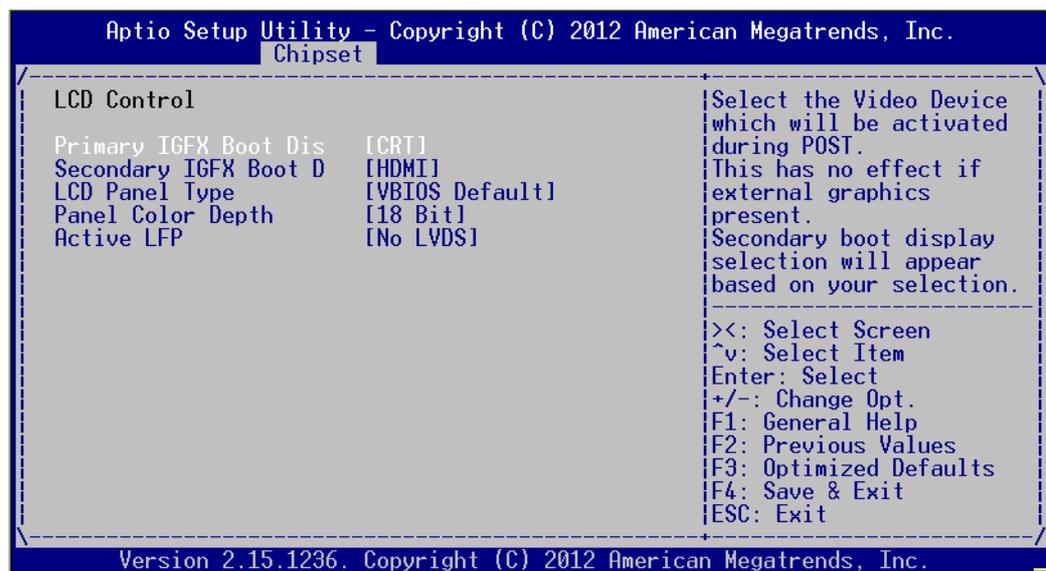


图 4.21: LCD Control

- **Primary IGFX Boot Display**  
设置在 POST 阶段启动的视频设备，默认设置为“CRT”。
- **Secondary IGFX Boot Display**  
设置 Secondary Display Device。
- **LCD Panel Type**  
设置 LCD 分辨率，默认设置为“1024x768(24bit)”。
- **Panel Color Depth**  
设置 LFP 面板颜色深度。
- **Active LFP**  
选择 Active LFP Configuration。

## 4.6 启动设置

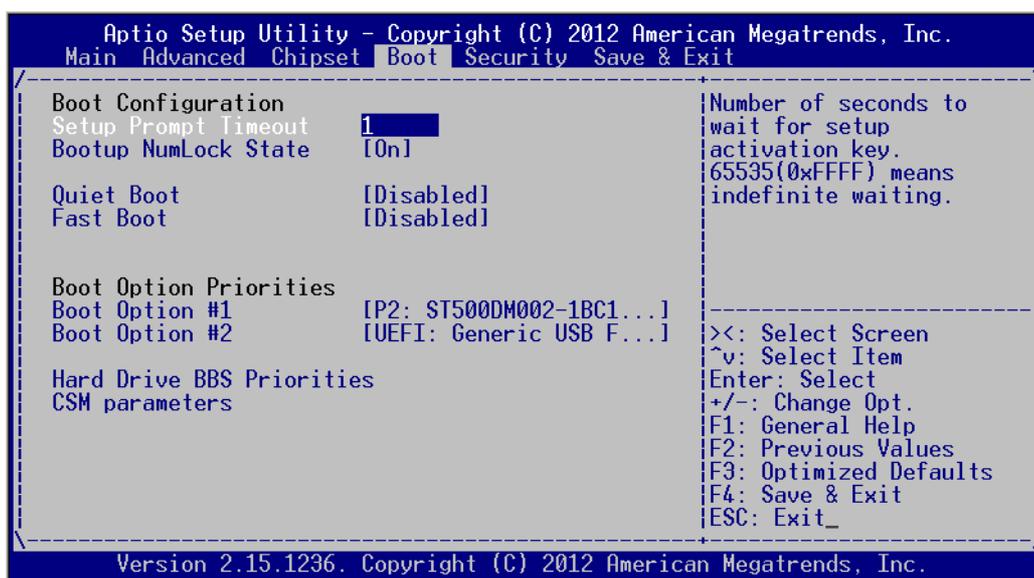


图 4.22: 启动设置

### 4.6.1 Boot Configuration

- **Setup Prompt Timeout**  
此项设置是按 Setup 快捷键的等待时间。如果在设置时间内没有按 Setup 快捷键就继续启动。
- **Bootup NumLock State**  
此项设置允许在系统上电至 DOS 系统后激活小键盘的数字锁功能。默认设置为“On”。  
On: 系统启动时处于数字锁开。  
Off: 启动时小键盘处于光标控制状态。
- **Quiet Boot**  
如果设置为“Disabled”则 BIOS 将显示正常的 POST 信息。如果设置为“Enabled”屏幕上将出现 OEM 图标，而非 POST 信息。
- **Fast Boot**  
此项允许 BIOS 在启动过程中跳过一些检测步骤，从而减少系统启动的时间。默认设置为“Disabled”。

## 4.6.2 Boot Option Priorities

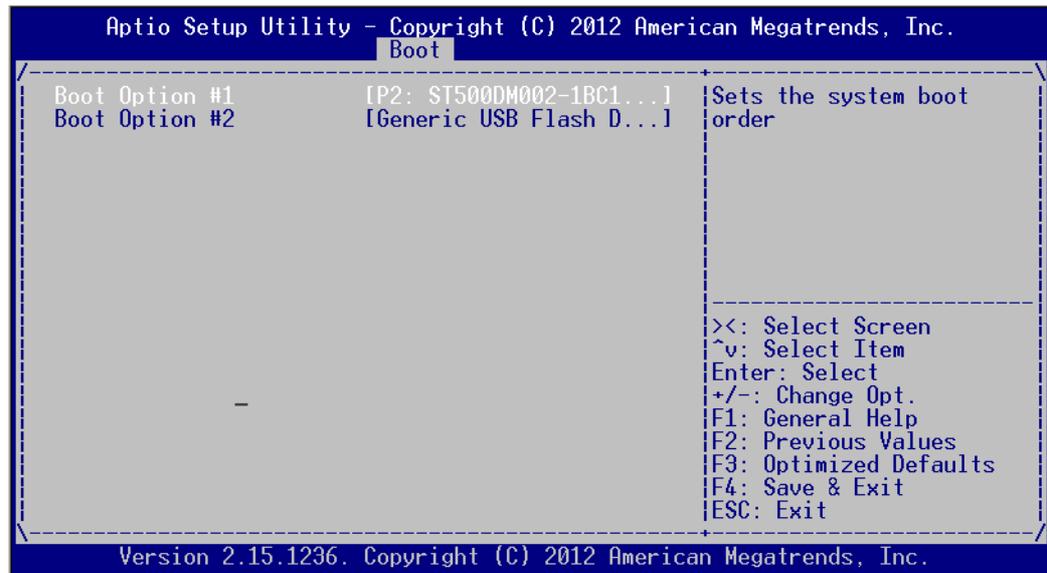


图 4.23: Boot Option Priorities

此项用于选择启动设备的启动顺序。

## 4.6.3 CSM 参数

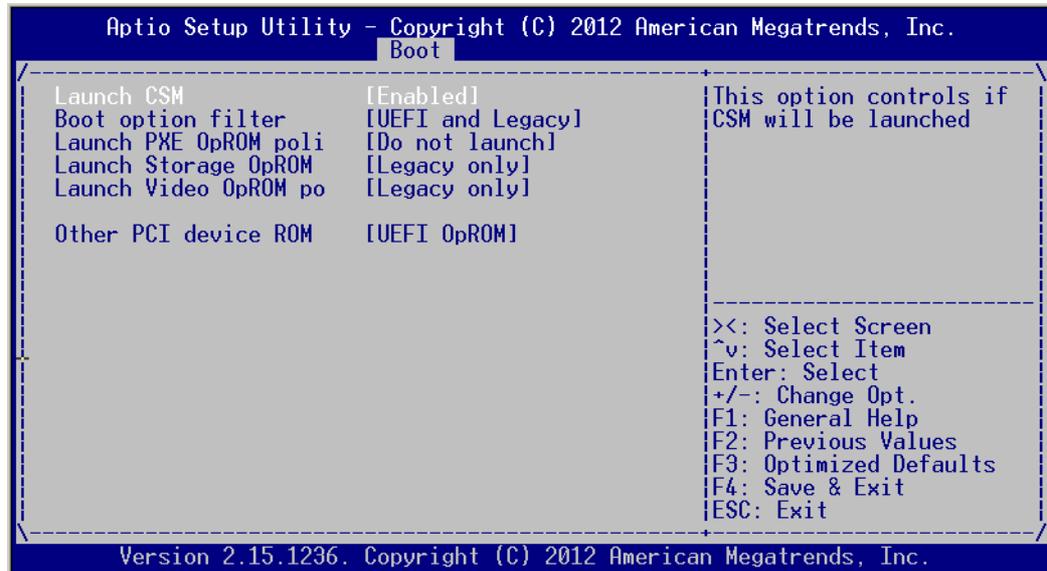


图 4.24: CSM 参数

- **Launch CSM**  
用于开启或禁用 CSM，默认设置为 “Enable”。
- **Boot option filter**  
用于控制启动设备系统，默认设置为 “UEFI and Legacy”。
- **Launch PXE OpROM policy**  
用于控制 UEFI 的执行和向下兼容 PXE OpROM，默认设置为 “Do not Lanch”。
- **Launch Storage OpROM policy**  
用于控制 UEFI 的执行和向下兼容储存 OpROM，默认设置为 “Legacy only”。
- **Launch Video OpROM policy**  
用于控制 UEFI 的执行和向下兼容视频 OpROM，默认设置为 “Legacy only”。
- **Other PCI device ROM priority**  
用于不同于网络的 PCI 设备，由 OpROM 启动的海量存储或视频定义。

## 4.7 安全设置



图 4.25: Setup Security Menu

- **Administrator Password**  
用于设置管理员密码。
- **User Password**  
用于设置用户密码。

## 4.8 Save & Exit 设置

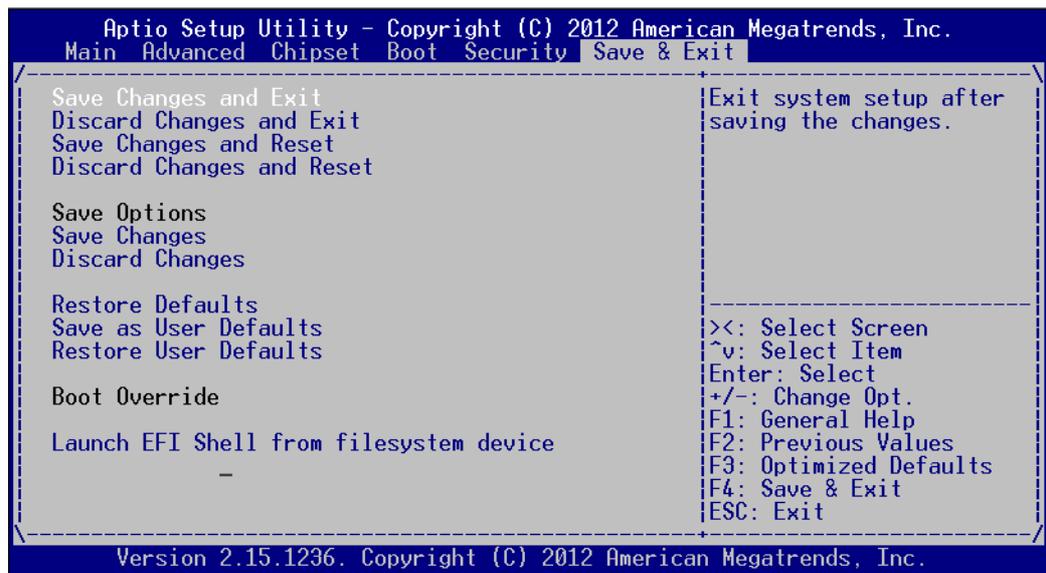


图 4.26: Setup Save & Exit Menu

### ■ Save Changes and Exit

如果已完成系统配置，请选择此项退出设置并重启计算机，则新的配置参数即可生效。

1. 请从退出菜单中选择“Save Changes and Exit”按后按下 <Enter> 键。屏幕上将出现以下信息：

Save Configuration Changes and Exit Now?

[Ok] [Cancel]

2. 选择“Ok”或“Cancel”。

### ■ Discard Changes and Exit

选择此项将退出设置且不保存对系统配置的任何修改。

1. 请从退出菜单中选择“Discard Changes and Exit”并按下 <Enter> 键。屏幕上将出现以下信息：

Discard Changes and Exit Setup Now?

[Ok] [Cancel]

2. 选择 Ok 将放弃修改并退出设置。

### ■ Save Changes and Reset

如果用户已经完成系统配置，请选择此项保存修改，退出 BIOS 设置菜单并重启后系统配置参数即可生效。

1. 选择保存修改并重置并按下 <Enter> 键。屏幕上将出现以下信息：

Save configuration and Reset? [Yes] [No]

2. 选择“Ok”或“Cancel”。

### ■ Discard Changes and Reset

选择此项将退出设置且不保存对系统配置的任何修改。

1. 请从退出菜单中选择“Discard Changes and Reset”并按下 <Enter> 键。屏幕上将出现以下信息：

Discard Changes and Reset Setup Now?

[Ok] [Cancel]

2. 选择 Ok 将放弃修改并退出设置。

### ■ Save Changes

允许保存用户所作的任何修改。

### ■ Discard Changes

允许放弃用作所作的任何修改。

- **Restore Default**  
用于恢复系统默认值。
- **Save as User Default**  
用于将所作的修改保存为用户默认值。
- **Restore User Default**  
用于恢复用户默认值。
- **Boot Override**  
用于设置开机的装置。
- **Launch EFI Shell from filesystem device**  
允许从系统文件设备启动 EFI shell。



## 第 5 章

### 驱动安装

内容包括：

- 芯片组 Windows 驱动安装
- VGA Windows 驱动安装
- ME Windows 驱动安装
- LAN Windows 驱动安装
- USB 3.0 Windows 驱动安装
- AUDIO Windows 驱动安装

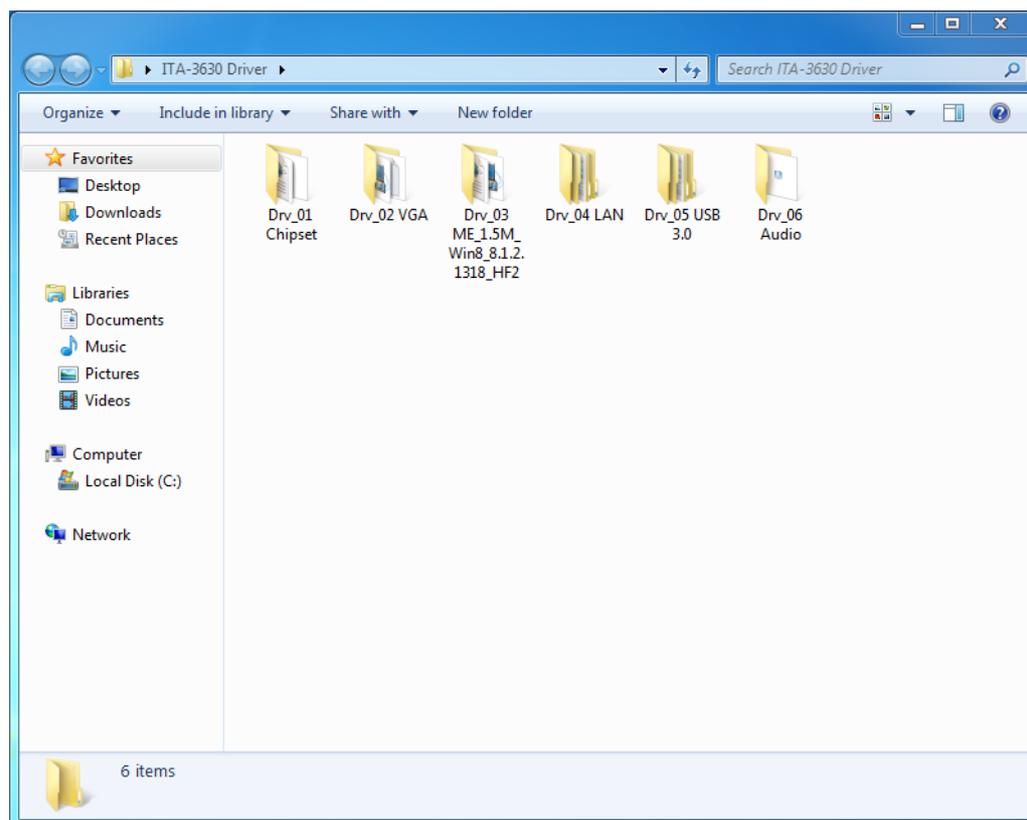
## 5.1 简介

研华为 Windows 程序开发人员提供了完整的设备驱动和软件。该设备驱动可应用于最通用的 Windows 编程工具中，如 Visual C++、Visual Basic、Borland C++ Builder 和 Borland Delphi。

以下驱动安装以 Windows 7 为例。

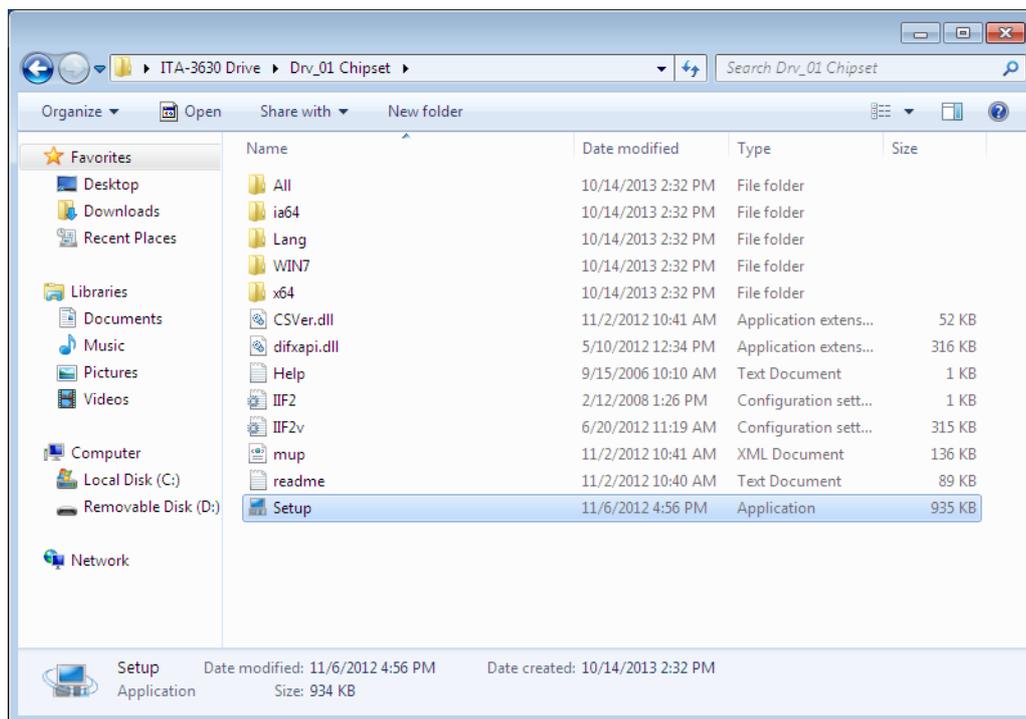
## 5.2 驱动安装

将驱动光盘插入系统 CD-ROM 驱动。用户即可看到 ITA-3630 系列驱动文件夹。



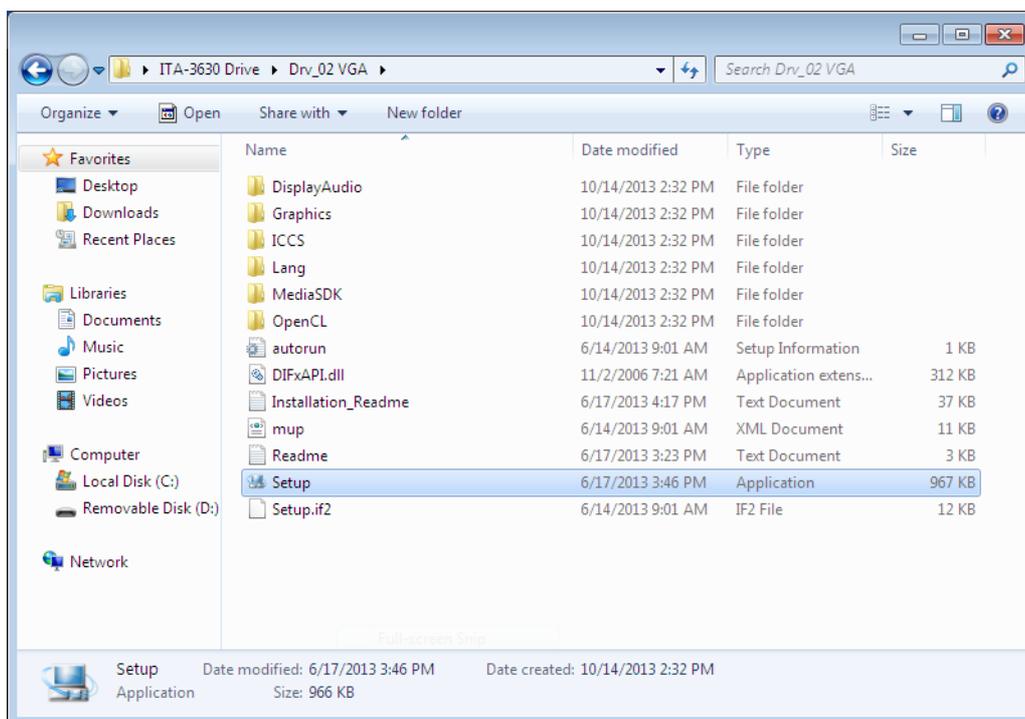
### 5.2.1 芯片组 Windows 驱动安装

将驱动光盘插入到系统 CD-ROM 驱动，用户即可看到 ITA-3630 系列驱动文件夹。找到“Drv\_01Chipset”文件夹然后双击“Setup”完成驱动安装。



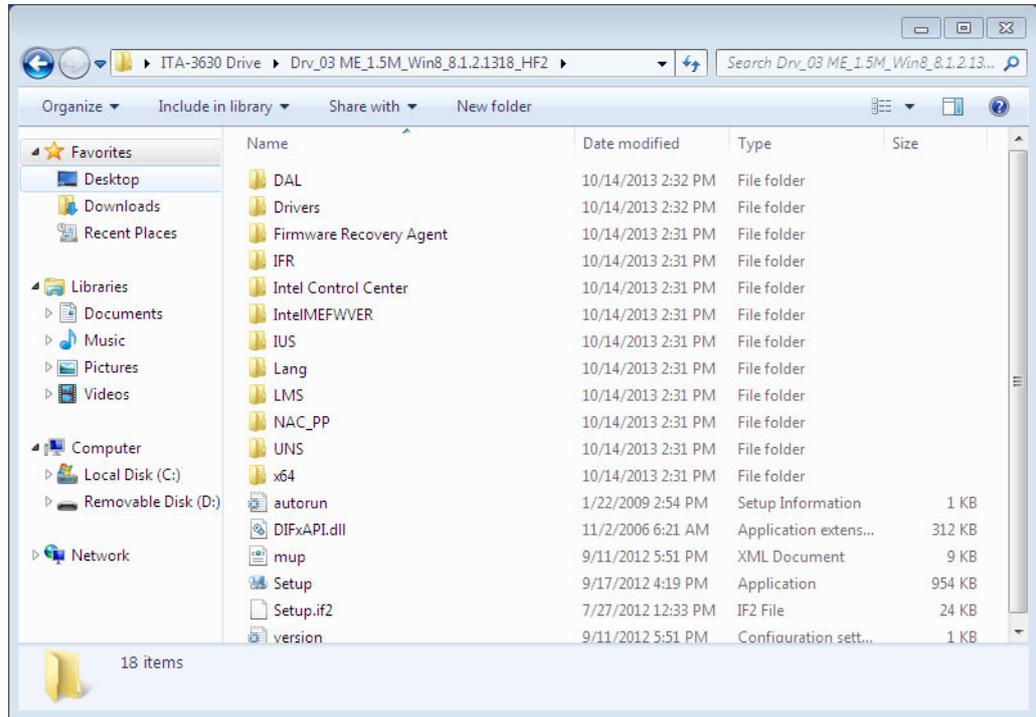
### 5.2.2 VGA Windows 驱动安装

将驱动光盘插入到系统 CD-ROM 驱动，用户即可看到 ITA-3630 系列驱动文件夹。找到“Drv\_02VGA”文件夹然后双击“Setup.exe”完成驱动安装。



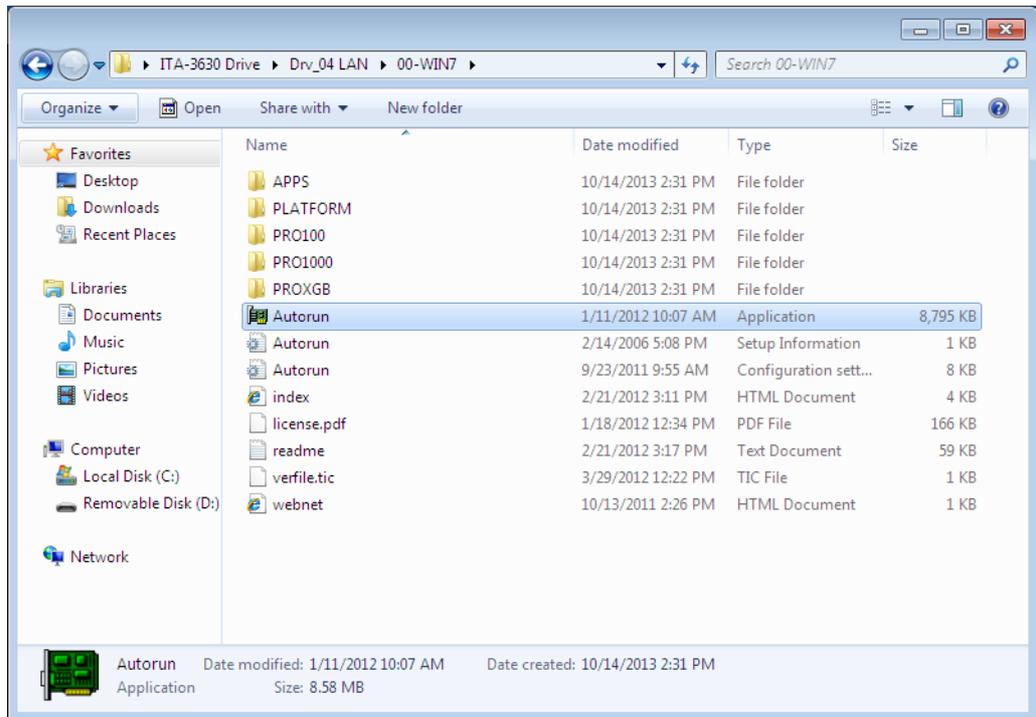
### 5.2.3 ME Windows 驱动安装

将驱动光盘插入到系统 CD-ROM 驱动，用户即可看到 ITA-3630 系列驱动文件夹。找到“Drv\_03ME”文件夹然后双击“Setup”完成驱动安装。



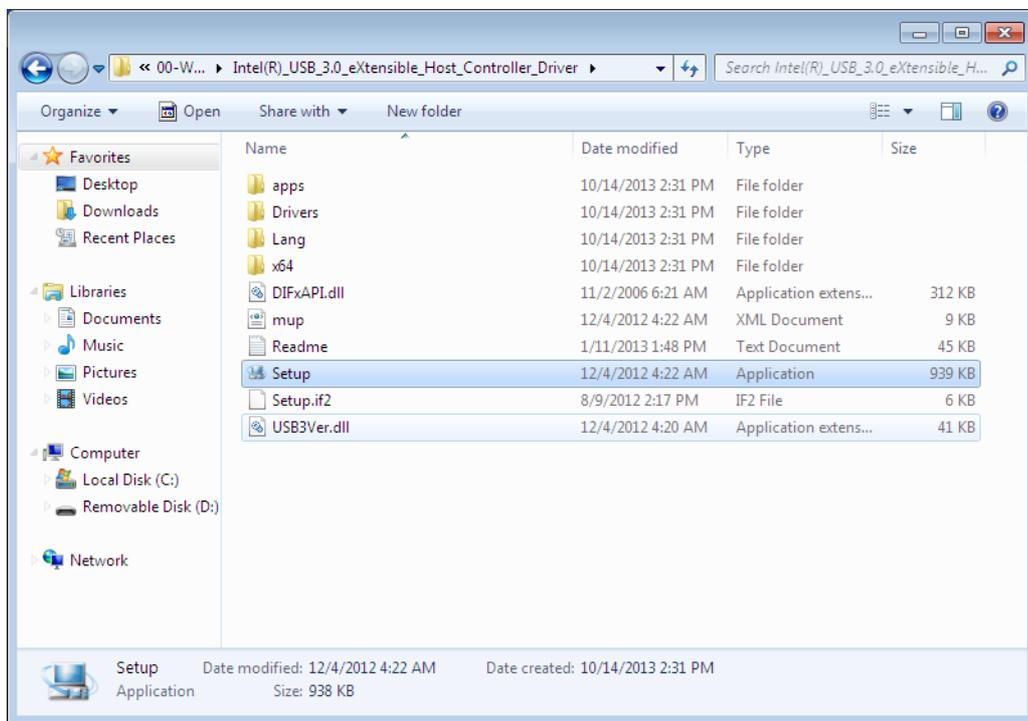
### 5.2.4 LAN Windows 驱动安装

将驱动光盘插入到系统 CD-ROM 驱动，用户即可看到 ITA-3630 系列驱动文件夹。找到“Drv\_04LAN”文件夹然后双击“Autorun”完成驱动安装。



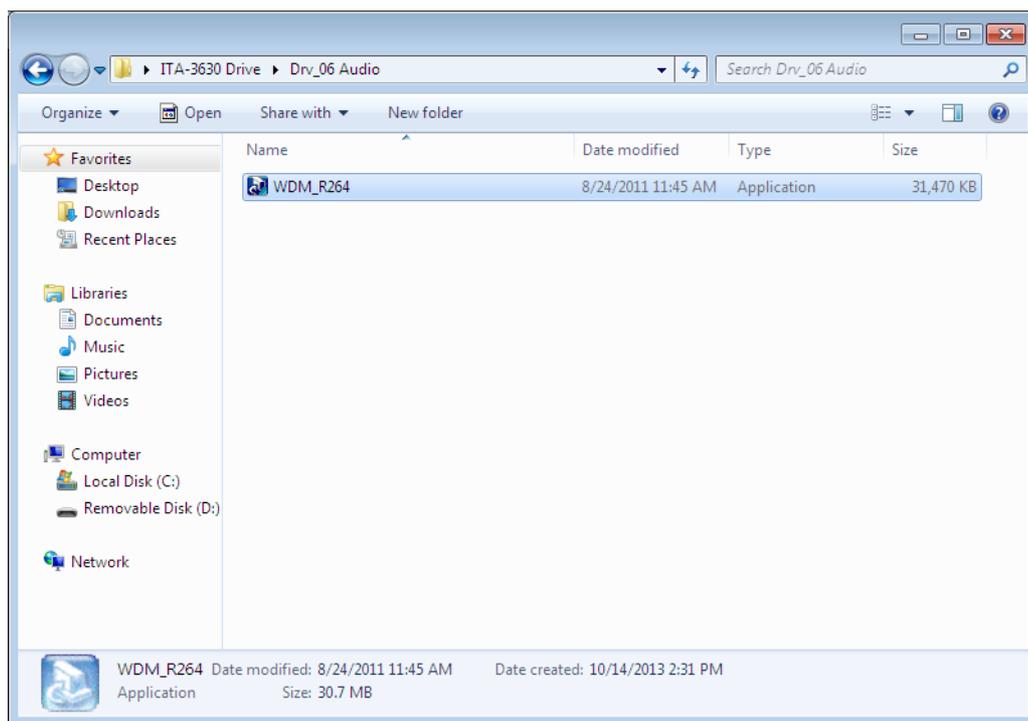
### 5.2.5 USB 3.0 Windows 驱动安装

将驱动光盘插入到系统 CD-ROM 驱动，用户即可看到 ITA-3630 系列驱动文件夹。找到“Drv\_05USB3.0”文件夹打开“Intel(R)\_USB\_3.0\_eXtensible\_Host\_Controller\_Driver”然后找到“Setup”双击完成驱动安装。



### 5.2.6 AUDIO windows 驱动安装

将驱动光盘插入到系统 CD-ROM 驱动，用户即可看到 ITA-3630 系列驱动文件夹。找到“Drv\_06AUDIO”文件夹然后双击“WDM\_R264.exe”完成驱动安装。





# 第 6 章

GPIO 编程指南

请仔细阅读并学习一下参考画面及蓝色粗体显示的源代码。  
请下载 NXP Semiconductors PCA9554 规格进行编程。

## 6.1 ITA-3630 数字 DIO 定义（见 2.3.6）

## 6.2 配置顺序

ITA-3630 的 GPIO 通过一个挂接在 ICH SMBUS 总线上的 PCA9554 GPIO IC 实现。  
因此对 GPIO IC 的设置和访问需要通过访问 ICH SMBUS controller 的 IO Space 来完成。  
下图是 ICH SMBUS IO Space 简图：

<b>SMB_BASE + Offset</b>	<b>Mnemonic</b>	<b>Register Name</b>	<b>Default</b>	<b>Type</b>
00h	HST_STS	Host Status	00h	R/WC, RO, R/WC (special)
02h	HST_CNT	Host Control	00h	R/W, WO
03h	HST_CMD	Host Command	00h	R/W
04h	XMIT_SLVA	Transmit Slave Address	00h	R/W
05h	HST_D0	Host Data 0	00h	R/W
06h	HST_D1	Host Data 1	00h	R/W

在 ITA-3630，上图中 **SMB\_BASE** 的 IO address 为 0xF40。  
具体的 SMBUS IO 控制访问的 code 在第 3 章节提供。

ITA-3630 上 GPIO 00 - GPIO 07 对应的 PCA9554 的 SMBUS slave address 为 0x40 (8bit address):

**GPIO 00 - GPIO 07 : PCA9554 0x40 (I00 - I07)**

下图是 PCA9554 的示意图:

**Table 2. Pin description**

Symbol	Pin			Description
	DIP16, SO16, SSOP16, TSSOP16	HVQFN16	SSOP20	
A0	1	15	6	address input 0
A1	2	16	7	address input 1
A2	3	1	9	address input 2
IO0	4	2	10	input/output 0
IO1	5	3	11	input/output 1
IO2	6	4	12	input/output 2
IO3	7	5	14	input/output 3
V <sub>SS</sub>	8	6 <sup>(1)</sup>	15	supply ground
IO4	9	7	16	input/output 4
IO5	10	8	17	input/output 5
IO6	11	9	19	input/output 6
IO7	12	10	20	input/output 7
INT	13	11	1	interrupt output (open-drain)

PCA9554 register 图解:

### 6.1.1 Command byte

**Table 3. Command byte**

Command	Protocol	Function
0	read byte	Input Port register
1	read/write byte	Output Port register
2	read/write byte	Polarity Inversion register
3	read/write byte	Configuration register

The command byte is the first byte to follow the address byte during a write transmission. It is used as a pointer to determine which of the following registers will be written or read.

PCA9554 共有 4 个 register 来对 GPIO 做控制。

PCA9554 register 0:

### 6.1.2 Register 0 - Input Port register

This register is a read-only port. It reflects the incoming logic levels of the pins, regardless of whether the pin is defined as an input or an output by Register 3. Writes to this register have no effect.

The default 'X' is determined by the externally applied logic level, normally '1' when no external signal externally applied because of the internal pull-up resistors.

**Table 4. Register 0 - Input Port register bit description**

Bit	Symbol	Access	Value	Description
7	I7	read only	X	determined by externally applied logic level
6	I6	read only	X	
5	I5	read only	X	
4	I4	read only	X	
3	I3	read only	X	
2	I2	read only	X	
1	I1	read only	X	
0	I0	read only	X	

如果某个GPIO Pin被设置为Input, 可以通过register 0的对应bit读取相应的Input值

PCA9554 register 1:

### 6.1.3 Register 1 - Output Port register

This register reflects the outgoing logic levels of the pins defined as outputs by Register 3. Bit values in this register have no effect on pins defined as inputs. Reads from this register return the value that is in the flip-flop controlling the output selection, **not** the actual pin value.

**Table 5. Register 1 - Output Port register bit description**

*Legend: \* default value.*

Bit	Symbol	Access	Value	Description
7	O7	R	1*	reflects outgoing logic levels of pins defined as outputs by Register 3
6	O6	R	1*	
5	O5	R	1*	
4	O4	R	1*	
3	O3	R	1*	
2	O2	R	1*	
1	O1	R	1*	
0	O0	R	1*	

如果某个GPIO Pin被设置为Output, 可以通过往register 1的对应bit来输出Output值

PCA9554 register 2:

### 6.1.4 Register 2 - Polarity Inversion register

This register allows the user to invert the polarity of the Input Port register data. If a bit in this register is set (written with '1'), the corresponding Input Port data is inverted. If a bit in this register is cleared (written with a '0'), the Input Port data polarity is retained.

**Table 6. Register 2 - Polarity Inversion register bit description**

Legend: \* default value.

Bit	Symbol	Access	Value	Description
7	N7	R/W	0*	inverts polarity of Input Port register data
6	N6	R/W	0*	0 = Input Port register data retained (default value)
5	N5	R/W	0*	1 = Input Port register data inverted
4	N4	R/W	0*	
3	N3	R/W	0*	
2	N2	R/W	0*	
1	N1	R/W	0*	
0	N0	R/W	0*	

如果某个 GPIO Pin 被设置成 Input，可以设置 register 2 的对应 bit 控制 Input pin 的 Polarity

PCA9554 register 3:

### 6.1.5 Register 3 - Configuration register

This register configures the directions of the I/O pins. If a bit in this register is set, the corresponding port pin is enabled as an input with high-impedance output driver. If a bit in this register is cleared, the corresponding port pin is enabled as an output. At reset, the I/Os are configured as inputs with a weak pull-up to  $V_{DD}$ .

**Table 7. Register 3 - Configuration register bit description**

Legend: \* default value.

Bit	Symbol	Access	Value	Description
7	C7	R/W	1*	configures the directions of the I/O pins
6	C6	R/W	1*	0 = corresponding port pin enabled as an output
5	C5	R/W	1*	1 = corresponding port pin configured as input (default value)
4	C4	R/W	1*	
3	C3	R/W	1*	
2	C2	R/W	1*	
1	C1	R/W	1*	
0	C0	R/W	1*	

Register 3 用来设置每个 GPIO 是 Input 还是 Output:

如果对应 bit 为 '0'，则对应 GPIO pin 设置为 Output；

如果对应 bit 为 '1'，则对应 GPIO pin 设置为 Input。

示例:

以 ITA-3630 为例, 假定 GPIO 00 设置成 Output, GPIO 07 设置成 Input, 两个 pin 互联, 如何设定相应 register?

GPIO 00 对应 PCA9554 0x40 I00, 而 GPIO 07 对应 PCA9554 0x40 I07。

设置 GPIO 00 为 Output:

1. 读取 SMBUS slave 0x40 register 3 byte 值 ;
2. 将 step1 读取值 bit 0 设为 0 并写入 SMBUS slave 0x40 register 3 ;
3. 读取 SMBUS slave 0x40 register 1 byte 值;
4. 根据输出值为 Low 或者 High, 设置 step 3 读取值 bit 0 为 0 或 1 并写回 SMBUS slave 0x40 register 1。

设置 GPIO 07 为 Input 值 :

1. 读取 SMBUS slave 0x40 register 3 byte 值 ;
2. 将 step1 读取值 bit 7 设为 1 并写入 SMBUS slave 0x40 register 3 ;
3. 读取 SMBUS slave 0x40 register 0 byte 值;
4. 通过 step 3 bit7 的值判断 Input 值为 Low 还是 High。

## 6.3 Function call for reference

ICH SMBUS 访问代码

(以下代码是模拟 BIOS 对 SMBUS 的访问而实现的, 使用 Borand C++ 3.1 编译通过, 并在 DOS 下验证通过, 未在其他 OS 下验证)

```
#define SMBUS_PORT 0xF040//SMB_BASE 为 0xF040
typedef unsigned char BYTE;

////////////////////////////////////
////////////////////////////////////
BYTE    smbus_read_byte(BYTE addr, BYTE offset)
// 读取 SMBUS Register byte 值, 一次读回一个 byte 值, 其中 addr 为 slave address
// 如 0x40; offset 为 register offset
{
    int    i;
    BYTE  data;

    outportb(SMBUS_PORT + 4, (addr | 1));// 往 SMB_BASE + 4 写入 slave
address (读取的时候需要设置 slave address bit 0 为 1, 所以这里有 addr|1)
    newiodelay();// 延时
    newiodelay();// 延时

    chk_smbus_ready();// 判断 SMBUS 总线是否 ready

    outportb(SMBUS_PORT + 3, offset);// 往 SMB_BASE + 3 写入 register
offset
    newiodelay();// 延时
    newiodelay();// 延时
```

```

        outportb(SMBUS_PORT + 2, 0x48); // 往 SMB_BASE + 2 写入 SMBUS 命令, 0x48
的意思是开始 Byte data 数据传输
        newiodelay(); // 延时
        newiodelay(); // 延时

        for (i = 0; i <= 0x100; i++)
        {
            newiodelay(); // 较长延时
        }

        chk_smbus_ready(); // 判断 SMBUS 总线是否 ready
        return(inportb(SMBUS_PORT + 5)); // 从 SMB_BASE + 5 读取读出的 byte 值
    }

////////////////////////////////////
////////////////////////////////////
void    smbus_write_byte(BYTE addr, BYTE offset, BYTE value)
// 写 SMBUS Register byte 值, 一次写一个 byte 值, 其中 addr 为 slave address 如
0x40; offset 为 register offset; value 为要写入的值
{
    int    i;

        outportb(SMBUS_PORT + 4, addr); // 往 SMB_BASE + 4 写入 slave address (
写入的时候需要设置 slave address bit 0 为 0)
        moredelay(); // 较长延时
        moredelay(); // 较长延时

        chk_smbus_ready(); // 判断 SMBUS 总线是否 ready

        outportb(SMBUS_PORT + 3, offset); // 往 SMB_BASE + 3 写入 register
offset
        moredelay(); // 较长延时
        moredelay(); // 较长延时

        outportb(SMBUS_PORT + 5, value); // 往 SMB_BASE + 5 写入数据 value
        moredelay(); // 较长延时
        moredelay(); // 较长延时

        outportb(SMBUS_PORT + 2, 0x48); // 往 SMB_BASE + 2 写入 SMBUS 命令, 0x48
的意思是开始 Byte data 数据传输
        moredelay(); // 较长延时
        moredelay(); // 较长延时

        for (i = 0; i <= 0x100; i++)
        {
            newiodelay(); // 较长延时

```

```

    }

    chk_smbus_ready(); // 判断 SMBUS 总线是否 ready
}

////////////////////////////////////
////////////////////////////////////
int    chk_smbus_ready()
// 判断 SMBUS 总线是否 ready 或者完成动作，基本上是做很长的等待，看 SMBUS 是否
// 传输完命令，因为出错情况基本不会出现，故 BIOS Code 在 SMBUS byte read 和 write
// 中并没有对本函数返回值做相应的判断
{
    int i, result = 1;
    BYTE data;

    for (i = 0; i <= 0x800; i++)
    {
        //SMB_BASE + 0 为 SMBUS status 值
        data = inportb(SMBUS_PORT); // 做一次读取 SMBUS status 值
        data = check_data(SMBUS_PORT); // 做多次读取 SMBUS status 值
        outportb(SMBUS_PORT, data); // 写回 SMBUS status 值，也就是清除
        status 值（在对应 bit 写入 1 代表清除 status）

        if (data & 0x02)
        {
            // 如果 bit 1 置位（代表命令完成），则 SMBUS 已经 ready
            result = 0; // SMBUS ready
            break;
        }

        if (!(data & 0xBF))
        {
            // 如果除 bit 2（代表 SMBUS 错误）以外的 bit 都为 0，则 SMBUS
            已经 ready
            result = 0; // SMBUS ready
            break;
        }

        if (data & 0x04)
        {
            // 如果 bit 2 置位（代表 SMBUS 错误），则 SMBUS 已经出现错误，
            这种情况很少出现
            result = 1; // SMBUS error
            break;
        }
    }

    return result;
}

```

```

/////////////////////////////////////////////////////////////////
/////////////////////////////////////////////////////////////////
BYTE    check_data(WORD addr)
{
    int i;
    BYTE data;

    for(i = 0; i <= 6; i++)
    {
        data = inportb(addr);
        if (data != 0)
            break;
    }

    returndata;
}

```

```

/////////////////////////////////////////////////////////////////
/////////////////////////////////////////////////////////////////

```

```

void    newiodelay()
// 短延时
{
    outportb(0xeb, 0); //IO port 0xeb 没有实际设备占用，通过往此 port 写值
    做延时功能，使用者可根据实际情况用其他方法代替
}

```

```

/////////////////////////////////////////////////////////////////
/////////////////////////////////////////////////////////////////

```

```

void    moredelay()
// 较长延时
{
    int i;
    for (i = 0; i < 20; i++)
    {
        outportb(0xeb, 0); //IO port 0xeb 没有实际设备占用，通过往此 port
        写值做延时功能，使用者可根据实际情况用其他方法代替
    }
}

```

\*\*\*\*\*

GPIO 模拟代码

(以第二章的 GPIO 00 和 GPIO 07 为例)

往 GPIO 00 输出 High:

```

data = smbus_read_byte(0x40, 0x03); // 读取 slave 0x40 register 3 byte
data &= 0xfe; // bit 0 设为 0

```

---

```
smbus_write_byte(0x40, 0x03, data)// 写回, GPIO 00 设为了 Output  
data = smbus_read_byte(0x40, 0x01)// 读取 slave 0x40 register 1  
data |= 0x01;//bit 0 设为 1, 代表 High  
smbus_write_byte(0x40, 0x01, data)// 写回, 输出 High 值
```

从 GPIO 07 读取 Input 值:

```
data = smbus_read_byte(0x40, 0x03);//读取 slave 0x40 register 3 byte  
data |= 0x80;//bit 7 设为 1  
smbus_write_byte(0x40, 0x03, data)// 写回, GPIO 07 设为了 Input  
data = smbus_read_byte(0x40, 0x00)// 读取 slave 0x40 register 0, 此时  
bit 7 的响应值就能得出 Input 是 Low 还是 High
```

# 附录 A

看门狗定时器编程

## A.1 看门狗定时器编程

ITA-3630 看门狗定时器可用于监控系统软件操作，并在编程过程中出现软件故障时采取适当措施。本章节介绍了看门狗定时器操作以及如何编程。

### A.1.1 看门狗定时器概述

看门狗定时器内置于高级 I/O 控制器 SMC SCH3114，提供了以下用户可编程功能：可通过用户编程启用或禁用。

定时器可设置为 1 ~ 255 秒或 1 ~ 255 分钟。

在软件复位定时器超时时产生中断或复位信号。

### A.1.2 编程看门狗定时器

看门狗定时器的 I/O 端口地址为 680h (hex)。

表 A.1: 看门狗定时器寄存器

地址: 680 (hex)

寄存器偏移	读 / 写	说明
65 (hex)	读 / 写	将定时器计时单位设置为秒或分钟： 向 bit 7 写入 0: 将计时单位设置为分钟。(默认) 向 bit 7 写入 1: 将计时单位设置为秒。
66 (hex)	读 / 写	0: 停止定时器 (默认)。 01~FF (hex): 计时值，单位为秒或分钟，取决于寄存器 65 (hex) 的设置值。 该值说明了看门狗定时器等待选通的时间达到何值时产生中断或复位信号。向该寄存器写入新值可以复位定时器，使其以新值开始计时。
67 (hex)	读 / 写	配置看门狗定时器 Bit 1: 写入 1 启用键盘复位定时器，写入 0 禁用 (默认)。 Bit 2 写入 1 启用鼠标复位定时器，写入 0 禁用 (默认)。 Bit 7~4: 设置看门狗定时器中断映射： 1111=IRQ15 ..... 0011=IRQ3 0010=IRQ2 0001=IRQ1 0000=Disable (默认)
68 (hex)	读 / 写	控制看门狗定时器 Bit 0: 读取看门狗状态，1= 定时器超时。 Bit 2: 写入 1 立即产生超时信号，并自动返回到 0 (只写)。 Bit 3: 写入 1 允许 P20 有效时触发定时器超时，写入 0 禁用 (默认)。

### A. 1.3 编程示例

```

;-----
1. 启用看门狗定时器，并将超时间隔设置为 10 秒。
;-----
Mov dx, 6E5h ; 选择寄存器 65h, 看门狗定时器的 I/O 端口地址 680h+ 寄存器偏移 65h
Mov al, 80h ; 将计时单位设置为秒
Out dx, al

Mov dx, 6E6h ; 选择寄存器 66h, 看门狗定时器的 I/O 端口地址 680h+ 寄存器偏移 66h
Mov al, 10 ; 将超时间隔设置为 10 秒然后开始计时
Out dx, al

;-----
2. 启用看门狗定时器功能并将超时间隔设置为 5 分钟。
;-----
Mov dx, 6E5h ; 选择寄存器 65h, 看门狗定时器的 I/O 端口地址 680h+ 寄存器偏移 65h
Mov al, 00h ; 将计时单位设置为分钟
Out dx, al

Mov dx, 6E6h ; 选择寄存器 66h, 看门狗定时器的 I/O 端口地址 680h+ 寄存器偏移 66h
Mov al, 5 ; 将超时间隔设置为 5 分钟然后开始计时
Out dx, al

;-----
3. 启用鼠标复位看门狗定时器功能。
;-----
Mov dx, 6E7h ; 选择寄存器 67h, 看门狗定时器的 I/O 端口地址 680h+ 寄存器偏移 67h
In al, dx
Or al, 4h ; 启用鼠标复位看门狗定时器
Out dx, al

;-----
4. 启用键盘复位看门狗定时器功能。
;-----
Mov dx, 6E7h ; 选择寄存器 67h, 看门狗定时器的 I/O 端口地址 680h+ 寄存器偏移 67h
In al, dx
Or al, 2h ; 启用键盘复位看门狗定时器
Out dx, al

;-----
5. 定时器不计时时，产生超时信号
;-----
Mov dx, 6E8h ; 选择寄存器 68h, 看门狗定时器的 I/O 端口地址 680h+ 寄存器偏移 68h
In al, dx
Or al, 4h ; 立即产生超时信号
Out dx, al

```

**ADVANTECH**

*Enabling an Intelligent Planet*

[www.advantech.com.cn](http://www.advantech.com.cn)

使用前请检查核实产品的规格。本手册仅作为参考。

产品规格如有变更，恕不另行通知。

未经研华公司书面许可，本手册中的所有内容不得通过任何途径以任何形式复制、翻印、翻译或者传输。

所有其他产品名或商标均为各自所属方的财产。

© 研华公司 2014