

**Q/LS**

**Q/LS 0046B-2021**

**龙芯中科技术股份有限公司企业标准**

**龙芯CPU统一系统架构规范**

**(适用于LA架构通用PC、服务器系列)**

**V4.3**

2026-01-15 发布

2026-01-15 实施

龙芯中科技术股份有限公司

批准

## 版权声明

本文档版权归龙芯中科技术股份有限公司所有，并保留一切权利。未经书面许可，任何公司和个人不得将此文档中的任何部分公开、转载或以其他方式散发给第三方。否则，必将追究其法律责任。

## 免责声明

本文档仅提供阶段性信息，所含内容可根据产品的实际情况随时更新，恕不另行通知。如因文档使用不当造成的直接或间接损失，本公司不承担任何责任。

### 龙芯中科技术股份有限公司

Loongson Technology Corporation Limited

地址：北京市海淀区中关村环保科技示范园龙芯产业园 2 号楼

Building No.2, Loongson Industrial Park,

Zhongguancun Environmental Protection Park, Haidian District, Beijing

电话(Tel): 010-62546668

传真(Fax): 010-62600826

## 版本信息

版本记录	文档名	龙芯CPU统一系统架构规范
	版本号	V4.3
	创建人	研发部
版本历史		
序号	版本号	更新内容
1	V4.2	<p>1、规范格式变更：使用形式统一的整体规范，具有唯一的版本号、目录及历史修改记录，原有子规范变为章节，取消原有子规范的版本号、目录、历史修改记录。V4.1及之前版本规范的历史修改记录可查阅历史规范版本；</p> <p>2、第一章6.4 其他约定：加入对支持UEFI固件的操作系统iso的约定；</p> <p>3、第一章新增11 多核启动传参约定；</p> <p>4、第七章，修正BMC、看门狗和I2C的描述，修改之前3C5000和3D5000看门狗GPIO引脚说明错误；增加7A2000 I2C0功能描述；</p> <p>5、第八章，增加裕泰I000E网卡、增加得一微DDR3颗粒、增加兆易创新和武汉新芯以及PUYA相关SPI Flash、增加得一微和景嘉微板载SSD；</p> <p>6、8.11 PPTT：更新ACPI Processor ID域的描述；</p> <p>7、第一章8.6.22 I2S音频配置：将codec匹配方式由codec-name修改为codec；</p> <p>8、第一章8.6.18节标题变更为LPC配置，增加_DEP属性约束；</p> <p>9、第一章8.6 DSDT新增内存ECC校验，龙芯SE设备及内部SDF设备配置描述；</p> <p>10、第二章1.5中断变更为按中断控制器模型分类和描述；</p> <p>11、第三章2.3.5 GPIO配置：修正南北桥GPIO的AddressMaximum和RangeLength；</p> <p>12、更新附录B 龙芯ACPI设备ID：新增内存ECC校验，龙芯SE设备及内部SDF设备的ACPI ID；</p> <p>13、更新第八章龙芯外围功能芯片支持列表。</p>
2	V4.3	<p>1、PCI总线枚举章节修正PCI IO/MEMORY地址空间的AddressGranularity为64K字节对齐；</p> <p>2、新增第九章 龙芯2K3000固件开发规范；</p> <p>3、修正几处描述不清晰和书写问题；</p> <p>4、龙芯ACPI设备ID：新增LOON0013，LOON0014和LOON0015；</p> <p>5、第一章8.6.22 I2S音频配置：修正Package参数错误；</p> <p>6、删除GPIO配置章节中对gpio_base属性的描述，gpio_base不再静态定义，通过内核动态分配；</p> <p>7、新增第十章 龙芯2K3000内置显卡软硬件设计规范；</p> <p>8、第七章龙芯LA架构硬件设计规范：新增龙芯3B6000，3C6000/S，3C6000/D，3C6000/Q；</p> <p>9、更新第八章龙芯外围功能芯片支持列表。</p>

## 前言

龙芯CPU统一系统架构规范是龙芯产品在桌面PC、服务器领域的开发指导规范，通过规范整机设计、固件开发、内核开发的技术要求，定义了整机、固件、系统的兼容性标准。

# 目录

前言	iv
第一章 龙芯CPU通用PC/服务器系统固件与内核接口规范	1
1 范围	1
2 术语与约定	1
2.1 术语	1
2.2 约定	2
3 架构关系	2
4 地址空间规范	2
4.1 地址空间	2
4.2 DMA地址映射规范	3
5 中断配置规范	3
5.1 配置方法	3
6 BootLoader与内核传参规范	3
6.1 传参约定	3
6.2 命令行传递的约定	3
6.2.1 initrd	3
6.3 UEFI system table传递的约定	4
6.3.1 memory map表	4
6.3.2 initrd表	4
6.4 其他约定	5
7 SMBIOS规范的实现约定	5
8 ACPI规范的实现约定	8
8.1 RSDP	8
8.2 XSDT	9
8.3 MADT	10
8.4 SRAT	13
8.5 FADT	15
8.6 DSDT	18
8.6.1 PCI总线枚举	18
8.6.2 PCI中断路由	20
8.6.3 设备电源管理	20
8.6.3.1 USB	20
8.6.3.2 GMAC	21
8.6.4 USB设备配置	21
8.6.5 电池配置	21
8.6.6 电源适配器配置	25
8.6.7 处理器配置和控制	26
8.6.8 系统休眠唤醒	26

8.6.9 热键配置 . . . . .	26
8.6.10 热区管理 . . . . .	31
8.6.11 串口配置 . . . . .	31
8.6.12 GPIO配置 . . . . .	32
8.6.13 I2C配置 . . . . .	35
8.6.14 GPIO模拟I2C配置 . . . . .	35
8.6.15 RTC配置 . . . . .	36
8.6.16 PWM配置 . . . . .	36
8.6.17 SE设备 . . . . .	36
8.6.18 LPC配置 . . . . .	36
8.6.18.1 EC设备 . . . . .	37
8.6.19 BMC设备 . . . . .	38
8.6.20 DMA配置 . . . . .	38
8.6.21 SDIO/MMC配置 . . . . .	39
8.6.22 I2S音频配置 . . . . .	39
8.6.23 SPI配置 . . . . .	42
8.6.24 内存ECC校验配置 . . . . .	42
8.6.25 龙芯SE设备配置 . . . . .	43
8.6.25.1 SDF设备配置 . . . . .	43
8.7 FACS . . . . .	45
8.8 MCFCG . . . . .	46
8.9 SLIT . . . . .	46
8.10 SPCR . . . . .	47
8.11 PPTT . . . . .	47
9 VBIOS的传递约定 . . . . .	48
10 screen_info传参约定 . . . . .	48
11 多核启动传参约定 . . . . .	50
12 总结 . . . . .	50
附录A LINUX操作系统键值表 . . . . .	50
附录B 龙芯ACPI设备ID . . . . .	66
第二章 龙芯7A芯片组固件开发规范 . . . . .	67
1 设计约束 . . . . .	67
1.1 Audio . . . . .	67
1.1.1 HDA . . . . .	67
1.1.2 I2S . . . . .	67
1.2 VBIOS . . . . .	67
1.3 GMAC . . . . .	67
1.4 DMA地址映射 . . . . .	68
1.5 中断 . . . . .	68
1.5.1 扩展IO中断控制器模型 . . . . .	68
1.5.2 高级向量中断控制器模型 . . . . .	72

1.6 地址空间	73
1.6.1 龙芯5000系列处理器+7A芯片组地址空间	73
2 参考配置	76
2.1 MADT配置	76
2.2 FADT配置	79
2.3 DSDT配置	82
2.3.1 PCI总线枚举	82
2.3.2 设备电源管理	90
2.3.2.1 USB	91
2.3.2.2 GMAC	91
2.3.3 系统休眠唤醒	91
2.3.4 串口配置	91
2.3.5 GPIO配置	93
2.3.6 I2C配置	95
2.3.7 RTC配置	96
2.3.8 PWM配置	98
2.3.9 热区管理	99
2.3.10 SE设备	102
2.3.11 EC配置	102
2.3.12 SPI配置	102
2.4 FACS配置	103
2.5 S3休眠地址	103
2.6 SRAT配置	103
2.7 MCFCG配置	105
2.8 SLIT配置	106
2.9 SPCR配置	107
2.10 PPTT配置	108
第三章 龙芯2K2000固件开发规范	113
1 设计约束	113
1.1 Audio	113
1.1.1 HDA	113
1.1.2 I2S	113
1.2 VBIOS	113
1.3 中断	113
1.4 地址空间	116
2 参考配置	117
2.1 MADT配置	117
2.2 FADT配置	120
2.3 DSDT配置	123
2.3.1 PCI总线枚举	123
2.3.2 设备电源管理	126

2.3.2.1 USB	126
2.3.3 系统休眠唤醒	127
2.3.4 串口配置	127
2.3.5 GPIO配置	128
2.3.6 I2C配置	130
2.3.7 RTC配置	131
2.3.8 PWM配置	132
2.3.9 热区管理	134
2.3.10 SE设备	135
2.3.11 EC配置	136
2.4 FACS配置	136
2.5 S3休眠地址	136
2.6 SRAT配置	136
2.7 MCFC配置	138
2.8 SPCR配置	139
第四章 龙芯7A1000内置显卡软硬件设计规范	141
1 范围	141
2 术语与定义	141
3 架构关系	141
4 显示类型和模式	142
4.1多屏显示方案	143
4.2 DDC通道硬件连接方案	144
4.3 获取EDID	145
4.4热插拔探测	145
4.5 背光控制	145
5 VBIOS存储约定	146
第五章 龙芯7A2000内置显卡软硬件设计规范	147
1 范围	147
2 术语与定义	147
3 架构关系	147
4 7A2000 桥片硬件显示管线设计规范	148
4.1 硬件显示管线简介	148
4.2显示器显示管线方案	149
4.2.1 显示器显示管线	149
4.2.2 获取EDID	150
4.2.3 热插拔探测	150
4.3 液晶屏显示管线方案	150
4.3.1 液晶屏显示管线	150
4.3.1 获取EDID	151
4.3.2屏幕背光	151
4.5 显存设计约束	152



4.6 信号转换芯片选型约束	152
4.7 多屏显示方案	152
4.7.1 单屏方案	152
4.7.2 双屏方案	152
4.7.3 三屏方案	153
5 7A2000 VBIOS设备属性标配	153
5.1显示控制器属性表	154
5.2内置编码器属性表	154
5.3 外接信号转换器属性表	154
5.4 连接器属性表	155
5.5 GPIO 设备属性表	155
5.6 PWM 设备属性表	155
5.7 GPU属性表	156
5.8显存属性表	156
6 efifb	156
7 附录	157
7.1 显示器方案配置示例	157
7.2 液晶屏方案配置示例	157
第六章 龙芯2K2000内置显卡软硬件设计规范	159
1 范围	159
2 术语与定义	159
3 架构关系	159
4 2K2000芯片硬件显示管线设计规范	160
4.1 硬件显示管线简介	160
4.2显示器显示管线方案	161
4.2.1 显示器显示管线	161
4.2.2 获取EDID	162
4.2.3 热插拔探测	162
4.3 液晶屏显示管线方案	162
4.3.1 液晶屏显示管线	162
4.3.1 获取EDID	163
4.3.2屏幕背光	163
4.5 显存设计约束	164
4.6 信号转换芯片选型约束	164
4.7 多屏显示方案	164
4.7.1 单屏方案	164
4.7.2 双屏方案	164
5 2K2000 VBIOS设备属性标配	164
5.1显示控制器属性表	165
5.2内置编码器属性表	165
5.3 外接信号转换器属性表	166

5.4 连接器属性表	166
5.5 GPIO 设备属性表	167
5.6 PWM 设备属性表	167
5.7 GPU属性表	167
5.8显存属性表	167
6 efifb	168
7 附录	168
7.1 显示器方案配置示例	168
7.2 液晶屏方案配置示例	169
第七章 龙芯LA架构硬件设计规范	171
1 龙芯3A5000	171
2 龙芯3A6000	171
3 龙芯3C5000	171
4 龙芯3D5000	172
5 龙芯7A1000	173
6 龙芯7A2000	174
7 龙芯2K2000	175
8 龙芯3B6000	175
9 龙芯3C6000/S	176
10 龙芯3C6000/D	176
11 龙芯3C6000/Q	178
第八章 龙芯外围功能芯片支持列表	181
第九章 龙芯2K3000固件开发规范	190
1 设计约束	190
1.1 Audio	190
1.1.1 HDA	190
1.1.2 I2S	190
1.2 VBIOS	190
1.3 中断	190
1.4 地址空间	193
2 参考配置	193
2.1 MADT配置	193
2.2 FADT配置	196
2.3 DSDT配置	200
2.3.1 PCI总线枚举	200
2.3.2 设备电源管理	202
2.3.2.1 USB	202
2.3.3 系统休眠唤醒	203
2.3.4 串口配置	203
2.3.5 GPIO配置	204
2.3.6 I2C配置	206

2.3.7 RTC配置	207
2.3.8 PWM配置	209
2.3.9 热区管理	210
2.3.10 龙芯SE设备配置	211
2.3.11 EC配置	212
2.4 FACS配置	212
2.5 S3休眠地址	212
2.6 SRAT配置	212
2.7 MCFCG配置	214
2.8 SPCR配置	215
第十章 龙芯2K3000内置显卡软硬件设计规范	217
1 范围	217
2 术语与定义	217
3 架构关系	217
4 2K3000芯片硬件显示管线设计规范	218
4.1 硬件显示管线简介	218
4.2 显示器显示管线方案	219
4.2.1 显示器显示管线	219
4.2.2 获取EDID	219
4.2.3 热插拔探测	219
4.3 液晶屏显示管线方案	220
4.3.1 液晶屏显示管线	220
4.3.2 获取EDID	220
4.4 屏幕背光	220
4.5 显存设计约束	221
4.6 多屏显示方案	221
4.6.1 单屏方案	221
4.6.2 双屏方案	221
4.6.3 三屏方案	221
5 2K3000 VBIOS设备属性标配	221
5.1 显示控制器属性表	222
5.2 内置编码器属性表	222
5.3 连接器属性表	223
5.4 GPIO 设备属性表	223
5.5 PWM 设备属性表	223
5.6 GPU属性表	224
5.7 显存属性表	224
5.8 PANEL属性表	224
6 efifb	225
7 附录	225
7.1 显示器方案配置示例	225

7.2 液晶屏方案配置示例 . . . . .	226
-------------------------	-----

# 第一章 龙芯CPU通用PC/服务器系统固件与内核接口规范

## 1 范围

本规范规定了龙芯CPU的地址空间、中断配置、固件与内核接口传参实现、SMBIOS、ACPI及VBIOS实现约定的要求。本规范适用于龙芯LoongArch架构系列CPU。建议其它系统厂商遵循此规范开发相关产品。

本规范正文及附录A、B为通用规范，通用规范描述一般性的约定。针对不同的芯片组，请参阅对应的固件开发规范。

## 2 术语与约定

### 2.1 术语

本规范所用术语定义如下：

**固件：**Firmware，写入ROM、EPROM等非易失存储器中的程序，负责控制和协调集成电路。

**BIOS：**基本输入输出系统，Basic Input Output System，一组固化到主板上一个ROM芯片上的程序，它保存着计算机基本输入输出程序、系统设置信息、开机后自检程序和系统自启动程序。BIOS与硬件系统集成在一起，也被称为固件，本规范中固件和BIOS不做区分。

**UEFI：**统一可扩展固件接口，Unified Extensible Firmware Interface，是UEFI Forum推广和发展的全新类型PC固件的体系结构、接口和服务的建议标准。主要目的是提供在OS加载之前在所有平台上一致、正确指定的启动服务。

**UEFI system table：**包含了指向UEFI runtime、boot services和Configuration Table等内容的指针，具体可参考UEFI规范。

**UEFI Configuration table：**对应UEFI system table中的ConfigurationTable字段，包含了一个表数组，每个数组成员由GUID和对应表的指针构成，具体可参考UEFI规范。

**BootLoader：**引导加载程序，又称引导加载器、启动加载器或启动引导器，是由计算机固件（BIOS或UEFI）启动的软件，它负责用想要的内核参数加载内核，并根据配置文件初始化RAM磁盘。常见的BootLoader有：ELILO, SYSLINUX, EFI stub loader, GRUB等

**PMON：**龙芯平台使用的一种兼有BIOS和BootLoader部分功能的开放源码软件。

**SMBIOS(System Management BIOS)：**是主板或系统制造者以标准格式显示产品管理信息所需遵循的统一规范。DMI(Desktop Management Interface)是帮助收集电脑系统信息的管理系统，DMI信息的收集必须在严格遵照SMBIOS规范的前提下进行。SMBIOS和DMI是由行业指导机构 Desktop Management Task Force (DMTF)起草的开放性的技术标准。

**HT(HyperTransport)：**是一种为主板上的集成电路互连而设计的端到端总线技术，目的是加快

芯片间的数据传输速度。HT通常指CPU到主板芯片（或北桥）之间的连接总线，即HT总线。类似于Intel平台中的前端总线（FSB），HT按技术规格分有HT1.0、HT2.0、HT3.0、HT3.1。

**PCI**(Peripheral Component Interconnect): 是连接电子计算机主板和外部设备的总线标准，用于定义局部总线的标准。此标准允许在计算机内安装多达10个遵从PCI标准的扩展卡。

**CPU**(central processing unit): 中央处理器，简称处理器。

**Core**: 处理器核，特指一个物理cpu，是一个独立的硬件执行单元，有独立的寄存器和计算单元。

**Node**: 节点，NUMA体系结构中的一个概念，一个NUMA node有一组core和内存，core访问自身node 内存（本地内存）的速度要快于访问其他node内存（远端内存）的速度，访问速度与 node的距离有关。在没有特殊说明情况下，统一系统架构规范中提到的节点或node，均为NUMA节点。

## 2.2 约定

(1) 本规范中的地址，未明确说明为虚拟地址时，均表示物理地址。

## 3 架构关系

龙芯PC产品的固件与内核接口在系统各软件之间所处的层次关系如图3-1所示：

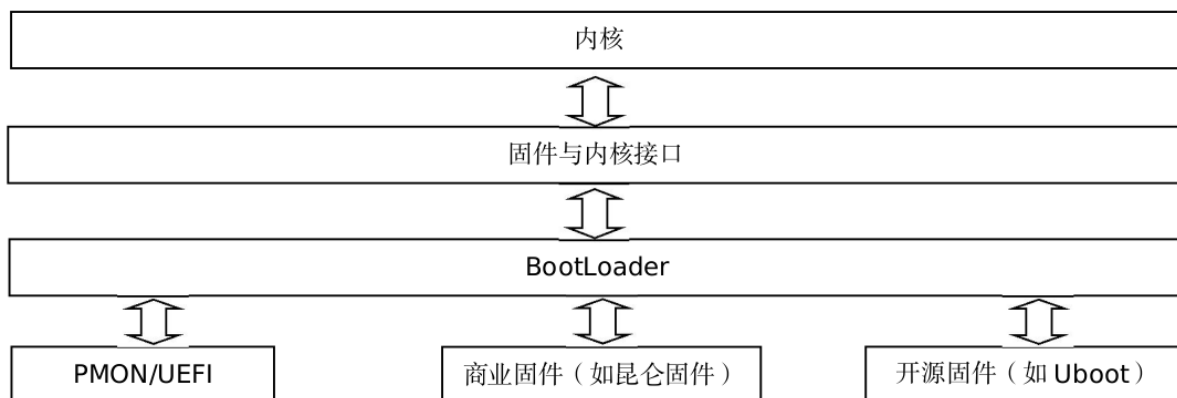


图3-1 接口与内核和固件之间的关系

## 4 地址空间规范

### 4.1 地址空间

龙芯CPU+芯片组的地址空间规定了内存空间、处理器及芯片组寄存器空间、PCI I/O空间、PCI MEM空间、PCI配置空间，每种地址空间的范围，请参阅对应处理器、芯片组的固件开发规

范。

## 4.2 DMA地址映射规范

DMA地址映射描述了系统中DMA地址与物理内存地址的转换关系，请参阅对应处理器、芯片组的固件开发规范。

## 5 中断配置规范

### 5.1 配置方法

根据ACPI规范，系统使用GSI（global system interrupts，全局系统中断）为系统中断源分配中断号，并通过ACPI表实现中断配置，系统GSI分配详见对应处理器、芯片组的固件开发规范。

## 6 BootLoader与内核传参规范

### 6.1 传参约定

BootLoader传递给内核的参数包括标志、命令行地址、UEFI system table地址，分别通过a0、a1、a2三个寄存器传递给内核。寄存器及对应的参数说明见下表：

表6-1 传参寄存器及对应参数说明

寄存器名	参数大小	参数说明
a0	u64	UEFI规范兼容标记，为1 1: 表示固件支持UEFI规范 0: 表示固件不支持UEFI规范 注：当前规范版本仅支持UEFI兼容的固件，固件不支持runtime时需将runtime指针置空，并通过命令行的noefi禁止内核的runtime功能
a1	u64	64位命令行物理地址
a2	u64	64位UEFI system table物理地址

### 6.2 命令行传递的约定

#### 6.2.1 initrd

当使用initrd时，用户可通过命令行向内核传递initrd信息，initrd信息的固定格式

为” initrd=initrd\_start,initrd\_size” , initrd\_start为initrd的起始64位物理地址, initrd\_size为initrd的64位大小。

## 6.3 UEFI system table传递的约定

### 6.3.1 memory map表

memory map表的数据结构定义如下:

```
struct efi_boot_memmap {  
    unsigned long map_size;  
    unsigned long desc_size;  
    u32 desc_ver;  
    unsigned long map_key;  
    unsigned long buff_size;  
    efi_memory_desc_t map[];  
};
```

其中, map\_size为UEFI memory map buffer大小; desc\_size为UEFI memory map描述符大小; desc\_ver为UEFI memory map 描述符版本; map\_key为UEFI memory map key; map为UEFI memory map起始地址; 具体请参考UEFI规范。buff\_size保留。

memory map表的地址存储在UEFI Configuration table中, 内核在UEFI Configuration table中查找memory map GUID({800f683f-d08b-423a-a293-965c3c6fe2b4})获取memory map地址。

### 6.3.2 initrd表

initrd表的数据结构定义如下:

```
struct efi_initrd {  
    unsigned long base;  
    unsigned long size;  
};
```

其中, base为initrd的起始64位物理地址, size为initrd的64位大小。

initrd表的地址存储在UEFI Configuration table中, 内核在UEFI Configuration table 中查找initrd GUID({5568e427-68fc-4f3d-ac74-ca55523lcc68})获取initrd表地址。



## 6.4 其他约定

UEFI Configuration table包含了SMBIOS表、ACPI表的入口地址、自定义screen\_info表、memory map表，详细描述参考章节7 SMBIOS规范的实现约定、章节8 ACPI的规范实现约定、章节10 screen\_info传参约定和章节6.3.1 memory map表。其中，传参涉及到的SMBIOS表、ACPI表、screen\_info表、memory map表、initrd的起始地址等地址均为64位物理地址且64K字节对齐。

initrd有两种传递方式（分别对应6.2.1和6.3.2章节），兼容UEFI规范的固件推荐使用UEFI system table传递。

支持UEFI固件的操作系统iso必须包含ESP（EFI system partition），ESP中的EFI文件路径为\EFI\BOOT\BOOTLOONGARCH64.EFI。

## 7 SMBIOS规范的实现约定

SMBIOS是主板或系统制造者以标准格式显示产品管理信息所需遵循的统一规范。DMI (Desktop Management Interface, DMI)就是帮助收集电脑系统信息的管理系统，DMI信息的收集必须在严格遵照SMBIOS规范的前提下进行。SMBIOS和DMI是由行业指导机构Desktop Management Task Force(DMTF)起草的开放性的技术标准，其中，DMI设计适用于任何的平台和操作系统。DMI充当了管理工具和系统层之间接口的角色。它建立了标准的可管理系统更加方便了电脑厂商和用户对系统的了解。DMI的主要组成部分是 Management Information Format(MIF)数据库。这个数据库包括了所有有关电脑系统和配件的信息。通过DMI，用户可以获取序列号、电脑厂商、串口信息以及其它系统配件信息。

SMBIOS表的地址存储在UEFI Configuration table中，内核在UEFI Configuration table中查找SMBIOS GUID({EB9D2D31-2D88-11D3-9A16-0090273FC14D})获取SMBIOS地址。

**龙芯固件平台必须实现的SMBIOS类别如下：**

BIOS 信息(Type 0)  
系统信息 (Type 1)  
产品信息 (Type 2)  
系统外围或底架 (Type3)  
处理器信息 (Type 4)  
高速缓存信息 (Type 7)  
系统插槽 (Type 9)  
物理存储阵列 (Type 16)  
存储设备(Type 17)  
存储阵列映射信息 (Type 19)  
表格结束指示 (Type 127)

其中：

**Type0**，描述固件信息。包括BIOS制造厂商、版本、ROM大小等，其中， BIOS Characteristics Extension Byte 2的Bit 3应与6.1章节a0一致。参考SMBIOS3系列协议实现。

**Type1**，描述系统信息。通常指的是品牌整机的信息，包括该机器的型号、版本、UUID等信息，参考SMBIOS3系列协议实现。

**Type2**，描述主板信息。指主板制造商和主板本身信息，包括主板生产厂家，主板生产名称、版本、串号等，参考SMBIOS3系列协议实现。

**Type3**，系统外围或底架。《System Management BIOS (SMBIOS) Reference Specification》7.4章节中规定了第六个字节的含义，里面包含了主机的类型。见表7-1。

表7-1 SMBIOS Type3第六字节信息含义

值	意义
01h	Other
02h	Unknown
03h	Desktop
04h	Low Profile Desktop
05h	Pizza Box
06h	Mini Tower
07h	Tower
08h	Portable
09h	Laptop
0Ah	Notebook
0Bh	Hand Held
0Ch	Docking Station
0Dh	All in One
0Eh	Sub Notebook
0Fh	Space-saving
10h	Lunch Box
11h	Main Server Chassis
12h	Expansion Chassis
13h	SubChassis
14h	Bus Expansion Chassis
15h	Peripheral Chassis

16h	RAID Chassis
17h	Rack Mount Chassis
18h	Sealed-case PC
19h	Multi-system chassis
1Ah	Compact PCI
1Bh	Advanced TCA
1Ch	Blade
1Dh	Blade Enclosure
1Eh	Tablet
1Fh	Convertible
20h	Detachable
21h	IoT Gateway
22h	Embedded PC
23h	Mini PC
24h	Stick PC

**Type4**，CPU信息；描述CPU表项信息，由龙芯实现，无需固件和主板厂商填写，龙芯固件针对不同节点数量实现相应的参考代码。

注：ProcessorVersion字段表示处理器名称；CurrentSpeed字段表示处理器运行频率；CoreCount字段表示一个封装内处理器核的数量。

**Type7**，Cache信息；描述Cache组织结构信息。由龙芯实现，无需固件和主板厂商填写。

**Type9**，系统插槽；描述主板的PCI、PCIE插槽的信息，该表项需要主板厂商根据各自设计自行完成；可参考龙芯固件代码实现。

**Type16**，物理存储阵列；描述内存的信息，如大小、DIMM槽数量、错误信息Handle等。该表项需要主板厂商根据各自设计自行完成，可参考龙芯固件代码实现。

**Type17**，存储设备；描述每个内存槽的信息，比如类型、大小、是否有ECC等。该表项需要主板厂商根据各自设计自行完成，可参考龙芯固件代码实现。

**Type19**，存储阵列映射信息；描述内存映射到物理地址的范围。需要根据二级交叉开关的映射关系以及主存实际大小填写相应的地址范围；该表项需要固件和主板厂商根据自己BIOS地址映射关系进行填写，可参考龙芯固件代码实现。

**Type127**，表格结束标识；标识SMBIOS表的结尾，无需固件和主板厂商填写，龙芯代码已经实现。

## 8 ACPI规范的实现约定

高级配置与电源接口（Advanced Configuration and Power Interface），简称ACPI，是独立于体系结构的电源管理和配置框架，此框架建立了一个硬件寄存器集来定义电源状态（睡眠、休眠、唤醒等），并在软件上通过ACPI表的方式描述硬件信息、特性和控制特性的方法。ACPI表列出了硬件板上无法使用硬件标准检测到或进行电源管理的设备，以及这些设备的功能。ACPI表还列出了休眠电源状态、可用电源平面描述等系统功能。

本接口规范支持表项如8-1:

表8-1支持表项

表	描述	是否强制
RSDP	Root System Description Pointer	是
XSDT	Extended System Description Table	是
MADT	Multiple APIC Description Table	是
SRAT	System Resource Affinity Table	是
FADT	Fixed ACPI Description Table	是
DSDT	Differentiated System Description Table	是
FACS	Firmware ACPI Control Structure	是
MCFG	PCI Express Memory-mapped Configuration Space base address description table	是
SLIT	System Locality Distance Information Table	否
SPCR	Serial Port Console Redirection Table	是
PPTT	Processor Properties Topology Table	否

注：建议同一厂家的各个表项的OEM ID保持相同

### 8.1 RSDP

RSDP（Root System Description Pointer）是整个ACPI表的第一个表，与其他表的关系可以用下图表示。

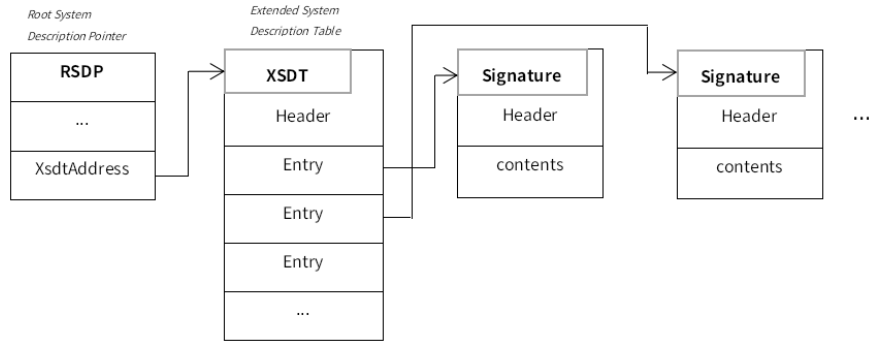


图8-1 RSDP示意图

RSDP结构包含XSDT（指向其他系统描述表的64位指针数组，见章节8.2）的地址。RSDP的地址存储UEFI Configuration table中，内核通过在UEFI Configuration table中查找ACPI2.0 GUID ({8868e871-e4f1-11d3-bc22-0080c73c8881})来获取RSDP的地址。

RSDP填写规范见表8-2：

表8-2 RSDP结构

域	长度（字节）	偏移（字节）	描述
Signature	8	0	“RSD PTR”
Checksum	1	8	ACPI 1.0规范中定义的字段的校验和。只包括该表的前20个字节，字节0到19，包括校验和字段。这些字节的和必须为零。（代码实际实现）
OEMID	6	9	OEM标识字符串
Revision	1	15	
RsdtdAddress	4	16	RSDT的32位地址
Length	4	20	表的长度，包括表头，从偏移量0开始。用于记录整个表的大小。
XsdtAddress	8	24	XSDT的64位地址
Extended Checksum	1	32	整个表的校验和，从偏移0开始到表结束，字节和为0
Reserved	3	33	保留

## 8.2 XSDT

XSDT（Extended System Description Table）包含其他系统描述表的64位指针数组，这些系统描述表向OS提供系统基本实现和配置的信息。部分定义如表8-3：

表8-3 XSDT约束

域	大小（字节）	偏移（字节）	描述
Header			
Signature	4	0	‘XSDT’
Revision	1	8	1

## 8.3 MADT

MADT（Multiple APIC Description Table）描述系统多核和中断信息。

表8-4，表列出了MADT部分约束条件，未列出的部分参考ACPI规范。

LoongArch架构支持的中断控制器模型包括CORE PIC、LIO PIC、HT PIC、EIO PIC、MSI PIC、BIO PIC、LPC PIC七种，具体结构定义参考对应表项。

表8-4 MADT约束

域	大小（字节）	偏移（字节）	描述
Header			
Signature	4	0	‘APIC’
Revision	1	8	1
Local Interrupt Controller Address	4	36	处理器中断控制器寄存器基地址
Flags	4	40	0
Interrupt Controller Structure[n]		44	中断控制器结构列表

MADT的CORE PIC结构参考如下配置：

表8-5 CORE PIC结构

域	大小（字节）	偏移（字节）	描述
Type	1	0	CORE PIC结构
Length	1	1	CORE PIC结构字节长度
Version	1	2	版本号
ACPI Processor ID	4	3	处理器核UID，与DSDT处理器对象中的_UID值相同

Physical Processor ID	4	7	CPU核物理ID
Flags	4	11	CORE PIC的标志，参考表8-6

表8-6 CORE PIC标志

CORE PIC Flags	大小（比特）	偏移（比特）	描述
Enabled	1	0	0: CPU不可用 1: CPU可用
Reserved	31	1	必须为0

MADT的LIO PIC结构参考如下配置：

表 8-7 LIO PIC结构

域	大小（字节）	偏移（字节）	描述
Type	1	0	LIO PIC结构类型
Length	1	1	LIO PIC结构字节长度
Version	1	2	版本号
Base Address	8	3	LIO PIC寄存器的基地址
Size	2	11	LIO PIC寄存器空间大小
Cascade vector	2	13	描述了LIO PIC路由到CORE PIC的向量信息,每个字节代表一个CORE PIC向量号。
Cascade vector mapping	8	15	描述了路由到CORE PIC的LIO PIC向量信息，CORE PIC向量由Cascade vector域指定，高4字节的LIO PIC向量对应于Cascade vector域高字节描述的CORE PIC向量，低4字节的LIO PIC向量对应于Cascade vector域低字节描述的CORE PIC向量。

MADT的HT PIC结构参考如下配置：

表 8-8 HT PIC结构

字段	大小（字节）	偏移（字节）	描述
Type	1	0	HT PIC结构类型
Length	1	1	HT PIC结构字节长度
Version	1	2	版本号

字段	大小（字节）	偏移（字节）	描述
Base Address	8	3	HT PIC寄存器的基地址
Size	2	11	HT PIC寄存器空间大小
Cascade Vector	8	13	第n字节表示 $32*n \sim 31*(n+1)$ 的HT PIC中断向量路由到LIO PIC的中断向量

MADT的EIO PIC结构参考如下配置：

表 8-9 EIO PIC结构

域	大小（字节）	偏移（字节）	描述
Type	1	0	EIO PIC结构类型
Length	1	1	EIO PIC结构字节长度
Version	1	2	版本号
Cascade vector	1	3	描述了EIO PIC路由到CORE PIC的CORE PIC向量号
Node	1	4	连接芯片组的处理器节点ID
Node map	8	5	EIO中断路由的处理器节点组，bit0-63分别表示0-63节点

注：EIO PIC结构中描述的节点概念与扩展IO中断中描述的EXT IOI node一致，详见相关处理器手册扩展IO中断章节。

MADT的MSI PIC结构参考如下配置：

表 8-10 MSI PIC结构

域	大小（字节）	偏移（字节）	描述
Type	1	0	MSI PIC结构类型
Length	1	1	MSI PIC结构字节长度
Version	1	2	版本号
Message Address	8	3	MSI消息的目标地址
Start	4	11	MSI在HT PIC或 EIO PIC中的起始向量
Count	4	15	MSI向量的个数

MADT的BIO PIC结构参考如下配置：

表 8-11 BIO PIC结构



域	大小（字节）	偏移（字节）	描述
Type	1	0	BIO PIC结构类型
Length	1	1	BIO PIC结构字节长度
Version	1	2	版本号
Base Address	8	3	BIO PIC寄存器的基地址
Size	2	11	BIO PIC寄存器空间的大小
Hardware ID	2	13	BIO PIC的硬件ID，即BIO PIC所在芯片组连接的处理器节点的节点号
GSI base	2	15	BIO PIC中断开始的GSI号，对于每个中断的GSI， $GSI = GSI\ base + BIO\ PIC\ 的中断向量$

MADT的LPC PIC结构参考如下配置：

表 8-12 LPC PIC结构

域	大小（字节）	偏移（字节）	描述
Type	1	0	LPC PIC结构类型
Length	1	1	LPC PIC结构字节长度
Version	1	2	版本号
Base Address	8	3	LPC PIC寄存器的基地址
Size	2	11	LPC PIC寄存器空间的大小
Cascade vector	2	13	描述了LPC PIC路由到BIO PIC的BIO PIC向量号

## 8.4 SRAT

SRAT（System Resource Affinity Table）为操作系统提供了处理器和内存范围的亲和性关系，在NUMA平台上，操作系统启动期间依据此表进行配置。表头的约束见表8-13。

龙芯平台需要实现两种亲和结构：

- （1）“Processor Local APIC/SAPIC Affinity Structure”，每个处理器核对应一个结构，见表8-14；
- （2）“Memory Affinity Structure”，每个Node的连续内存段对应一个结构，见表8-15。

表8-13 SRAT约束

域	大小（字节）	偏移（字节）	描述
Header			
Signature	4	0	‘SRAT’
Revision	1	8	2

表8-14 Processor Local APIC/SAPIC Affinity结构

域	大小（字节）	偏移（字节）	描述
Type	1	0	0，代表结构为Processor Local APIC/SAPIC Affinity Structure。
Length	1	1	16
Proximity Domain [7:0]	1	2	处理器亲和域[7:0]
APIC ID	1	3	处理器Local APIC ID，见MADT表
Flags	4	4	标志。0位： 0表示该Processor Local APIC/SAPIC Affinity Structure不可用； 1表示该Processor Local APIC/SAPIC Affinity Structure可用。 [31:1]位：必须为0
Local SAPIC EID	1	8	用于x86架构的SAPIC
Proximity Domain [31:8]	3	9	处理器亲和域[31:8]
Clock Domain	4	12	处理器时钟域

表8-15 Memory Affinity结构

域	大小（字节）	偏移（字节）	描述
Type	1	0	1，代表结构为Memory Affinity Structure
Length	1	1	40
Proximity Domain	4	2	内存亲和域
Reserved	2	6	保留
Base Address Low	4	8	内存范围地址的低32位
Base Address High	4	12	内存范围地址的高32位

Length Low	4	16	内存范围大小的低32位
Length High	4	20	内存范围大小的高32位
Reserved	4	24	保留
Flags	4	28	标志，指示内存区域是否已启用并可以热插拔，见表8-16
Reserved	8	32	保留

表 8-16内存热插拔标志

域	大小（ 比特）	偏移（ 比特）	描述
Enabled	1	0	0: 该Memory Affinity Structure不可用 1: 该Memory Affinity Structure可用
Hot Pluggable	1	1	是否支持内存热插拔
NonVolatile	1	2	是否为非易失内存
Reserved	29	3	0

## 8.5 FADT

FADT（Fixed ACPI Description Table）为操作系统提供了Fixed硬件ACPI描述信息。

表8-17列出了FADT部分约束条件，未列出的部分请参考ACPI规范。

表8-17 FADT约束

域	大小（ 字节）	偏移（ 字节）	描述
Header			
Signature	4	0	‘FACP’
Length	4	4	表的长度，包括表头，从偏移量0开始。用于记录整个表的大小
FADT Major Version	1	8	FADT版本号
FIRMWARE_CTRL	4	36	32位FACS 表地址
DSDT	4	40	32位DSDT表地址
Reserved	1	44	0
SCI_INT	2	46	SCI中断号
SMI_CMD	4	48	SMI命令端口地址，龙芯平台无效

ACPI_ENABLE	1	52	写入SMI_CMD端口的数值，用于使能SMI对ACPI硬件寄存器的控制权
ACPI_DISABLE	1	53	写入SMI_CMD端口的数值，用于释放SMI对ACPI硬件寄存器的控制权
S4BIOS_REQ	1	54	写入SMI_CMD端口的数值，用于进入S4BIOS模式
PSTATE_CNT	1	55	非0,则表示OSPM可向SMI_CMD写入该值，负责处理器性能控制
PMIa_EVT_BLK	4	56	PMIa Event Register Block的地址
PMIb_EVT_BLK	4	60	PMIb Event Register Block的地址
PMIa_CNT_BLK	4	64	PMIa Control Register Block的地址
PMIb_CNT_BLK	4	68	PMIb Control Register Block的地址
PM2_CNT_BLK	4	72	PM2 Control Register Block的地址
PM_TMR_BLK	4	76	Power Management Timer Control Register Block的地址
GPE0_BLK	4	80	General-Purpose Event 0 Register Block的地址
GPE1_BLK	4	84	General-Purpose Event 1 Register Block的地址
PMI_EVT_LEN	1	88	PMIa_EVT_BLK长度，单位为字节
PMI_CNT_LEN	1	89	PMIa_CNT_BLK长度，单位为字节
PM2_CNT_LEN	1	90	PM2_CNT_BLK长度，单位为字节
PM_TMR_LEN	1	91	PM_TMR_BLK长度，单位为字节
GPE0_BLK_LEN	1	92	GPE0_BLK长度，单位为字节
GPE1_BLK_LEN	1	93	PM2_CNT_BLK的长度
GPE1_BASE	1	94	ACPI GPE模型中GPE1的_偏移量
CST_CNT	1	95	非0,则OSPM可向SMI_CMD写入该值，表示支持_CST对象及相应的通知
P_LVL2_LAT	2	96	大于0x64表示不支持C2状态
P_LVL3_LAT	2	98	大于0x3e8表示不支持C3状态
FLUSH_SIZE	2	100	一次刷cache的大小，单位字节，仅当WBINVD为0时有效
FLUSH_STRIDE	2	102	cache line大小，单位字节，仅当WBINVD为0时有效
DUTY_OFFSET	1	104	P_CNT中处理器的空闲周期起始值
DUTY_WIDTH	1	105	P_CNT中处理器的空闲周期宽度
DAY_ALARM	1	106	RTC日期的偏移地址

MON_ALARM	1	107	RTC月份的偏移地址
CENTURY	1	108	RTC世纪的偏移地址
IAPC_BOOT_ARCH	2	109	IA-PC启动标志
Reserved	1	111	0
Flags	4	112	Fixed特征标志，支持下列标志： WBINVD、PROC_CI、SLP_BUTTON、RESET_REG_SUP、PCI_EXP_WAK 其中当RESET_REG_SUP为1，表示通过内核操作RESET_REG复位，为0表示通过固件操作RESET_REG复位
RESET_REG	12	116	见表8-18
RESET_VALUE	1	128	写入RESET_REG的值
ARM_BOOT_ARCH	2	129	ARM平台启动标志
FADT Minor Version	1	131	0
X_FIRMWARE_CTRL	8	132	FACS 64bit地址，固件动态生成
X_DSDT	8	140	DSDT 64bit地址，固件动态生成
X_PM1a_EVT_BLK	12	148	PM1a Event Register Block的64地址，见表8-18
X_PM1b_EVT_BLK	12	160	PM1b Event Register Block的64地址
X_PM1a_CNT_BLK	12	172	PM1a Control Register Block的64位地址，见表8-18
X_PM1b_CNT_BLK	12	184	PM1b Control Register Block的64位地址
X_PM2_CNT_BLK	12	196	PM2 Control Register Block的64位地址
X_PM_TMR_BLK	12	208	Power Management Timer Control Register Block的64位地址，见表8-18
X_GPE0_BLK	12	220	GPE0 block的64位地址，见表8-18
X_GPE1_BLK	12	232	GPE1 block的64位地址

注：UEFI固件动态生成X\_FIRMWARE\_CTRL与X\_DSDT之前，需将其初始化为0。

表 8-18 寄存器结构

域	大小（字节）	偏移（字节）	描述
Address Space Id	1	0	代表为System Memory
Reg Bit Width	1	1	寄存器位宽
Reg Bit Offset	1	2	寄存器偏移

Reserved	1	3	0
Address	8	4	寄存器地址

## 8.6 DSDT

DSDT (Differentiated System Description Table) 用来描述主板设计差异化信息。

本章节规定了龙芯平台ACPI设备ID和对DSDT配置的支持范围，支持范围之外的配置未验证，不保证其正确性。龙芯平台ACPI设备ID请参照附录B。

以下各小节列出了龙芯平台支持的对象及方法，并对其中部分对象及方法进行了约束，未做约束的请参考ACPI规范。

### 8.6.1 PCI总线枚举

支持的对象及方法：\_BBN、\_ADR、\_SEG、\_HID、\_CID、\_CRS、\_OSC。

(1) \_CRS (Current Resource Settings)

\_CRS方法支持总线范围、IO地址范围以及内存地址范围声明。

使用WordBusNumber()宏声明PCI总线范围，各个参数规定如表8-19：

表8-19 WordBusNumber()宏参数表

参数名	值
ResourceUsage	描述总线范围的使用者是设备本身，或是下级总线设备
IsMinFixed	描述总线范围的最小值是否固定
IsMaxFixed	描述总线范围的最大值是否固定
Decode	描述设备解码总线范围的方向
AddressGranularity	总线范围起始号对齐
AddressMinimum	总线范围的最小值
AddressMaximum	总线范围的最大值
AddressTranslation	相对于主总线范围的偏移
RangeLength	总线范围大小
ResourceSourceIndex	资源描述符索引
ResourceSource	资源描述符
DescriptorName	资源描述符名称

使用QWordIO()宏声明PCI IO地址范围，各个参数规定如表8-20：

表8-20 QWordIO()宏参数表

参数名	值
ResourceUsage	描述I/O范围的访问者
IsMinFixed	最小地址是否固定
IsMaxFixed	最大地址是否固定
Decode	描述I/O范围的访问方向
ISARanges	是否为ISA范围
AddressGranularity	地址对齐
AddressMinimum	起始地址
AddressMaximum	结束地址
AddressTranslation	相对于主总线的地址偏移
RangeLength	地址范围大小
ResourceSourceIndex	资源描述符索引
ResourceSource	资源描述符
DescriptorName	资源描述符名称
TranslationType	I/O转换类型，下级总线是否与本总线资源相同，则不需要转换，否则需要指定AddressTranslation
TranslationDensity	稀疏转换还是连续转换

使用QWordMemory()宏声明PCI MEMORY地址范围，各个参数规定如表8-21:

表8-21 QWordMemory参数表

参数名	值
ResourceUsage	描述内存范围的访问者
Decode	描述内存范围的访问方向
IsMinFixed	最小地址是否固定
IsMaxFixed	最大地址是否固定
Cacheable	是否支持cache
ReadAndWrite	是否可读写
AddressGranularity	地址对齐
AddressMinimum	起始地址
AddressMaximum	结束地址

参数名	值
AddressTranslation	相对于主总线的地址偏移
RangeLength	地址范围大小
ResourceSourceIndex	资源描述符索引
ResourceSource	资源描述符
DescriptorName	资源描述符名称
MemoryRangeType	内存访问类型
TranslationType	转换类型，下级总线是否与本总线资源相同，则不需要转换，否则需要指定AddressTranslation

## 8.6.2 PCI中断路由

支持的对象及方法：\_PRT，描述芯片组PCI主桥上PCI中断分配信息。固件必须提供此对象，规定如下：

- (1) \_PRT中描述的PCI路由信息仅限于芯片组集成PCI/PCIE设备，对于外扩PCI/PCIE设备，无需中断配置。
- (2) 表项中的中断号均使用GSI中断号，芯片组GSI分配详见芯片组固件开发规范。
- (3) 对于芯片组的PCIE PORT，\_PRT中需要实现4个表项，分别对应PCI规范中规定的A/B/C/D pin，4个表项使用相同的GSI中断号。
- (4) 对于芯片组的多功能设备，表项中的pin值为功能号。

## 8.6.3 设备电源管理

龙芯平台支持USB2.0和GMAC唤醒系统。

### 8.6.3.1 USB

支持的对象及方法：\_PR0、\_PRW。

- (1) \_PR0 (Power State 0)

必须提供这个控制方法。

- (2) \_PRW (Power Resources for Wake)

仅支持EventInfo和DeepestSleepState参数。



### 8.6.3.2 GMAC

支持的对象及方法：\_PR0、\_PRW。

(1) \_PR0 (Power State 0)

必须提供这个控制方法。

(2) \_PRW (Power Resources for Wake)

仅支持EventInfo和DeepestSleepState参数。

### 8.6.4 USB设备配置

支持的对象及方法：\_UPC及\_PLD。

### 8.6.5 电池配置

支持的对象及方法：\_BIF \_BST \_STA，龙芯平台仅支持Control Method Battery电池模型。

按照ACPI规范要求，\_BIF以及\_BST方法返回值为Package，格式分别如表8-22，8-23：

表8-22 \_BIF返回值package元素表

域	说明
Power Unit	电池信息使用单位
Design Capacity	电池设计容量
Last Full Charge Capacity	电池充满电时容量
Battery Technology	电池是否可充电
Design Voltage	电池设计电压
Design Capacity of Warning	电池告警电量
Design Capacity of Low	电池低电量
Battery Capacity Granularity 1	电池容量粒度1（低电量到告警电量之间）
Battery Capacity Granularity 2	电池容量粒度2（告警电量到满电量之间）
Model Number	电池型号
Serial Number	电池序列号
Battery Type	电池类型
OEM Information	OEM特定信息

表8-23 \_BST返回值package元素表

域	说明
Battery State	电池状态：充电状态，放电状态，是否处于紧急状态
Battery Present Rate	电池当前功率或电流
Battery Remaining Capacity	电池剩余电量
Battery Present Voltage	电池当前电压

通过EC（参见8.6.18.1节）管理电池时，\_BIF，\_BST，\_STA方法的实现可以参照如下示例：

```
Device (EC)
{
    Name (_HID, EisaId ("PNP0C09") /* Embedded Controller Device */) // _HID: Hardware
    ID
    Name (_UID, Zero) // _UID: Unique ID
    Name (_GPE, 0x04) // _GPE: General Purpose Events
    Name (_CRS, ResourceTemplate () // _CRS: Current Resource Settings
    {
        IO(Decode16, 0x62, 0x62, 0, 1,)
        IO(Decode16, 0x66, 0x66, 0, 1,)
    })
    OperationRegion (ECOR, EmbeddedControl, Zero, 0x0100)
    Field (ECOR, ByteAcc, NoLock, Preserve)
    {
        Offset (0x80),
        AST1,    1, //Battery Availability Status
        BCG1,    1, //Battery State
        LSTE,    1, //Lid Status
        RPWR,    1, //Power Source Status
        Offset (0x92),
        BVL,     8, //Battery Present Voltage Low 8 bits
        BVH,     8, //Battery Present Voltage High 8 bits
        BCL,     8, //Battery Present Rate Low 8 bits
        BCH,     8, //Battery Present Rate High 8 bits
        Offset (0x98),
        BRCL,    8, //Battery Remaining Capacity Low 8 bits
        BRCH,    8, //Battery Remaining Capacity High 8 bits
        BFCL,    8, //Battery Last Full Charge Capacity Low 8 bits
        BFCH,    8, //Battery Last Full Charge Capacity High 8 bits
        Offset (0xA2),
        BDVL,    8, //Battery Design Voltage Low 8 bits
```

```

    BDVH,    8, //Battery Design Voltage High 8 bits
    Offset (0xAE),
    BDCL,    8, //Battery Design Capacity Low 8 bits
    BDCH,    8, //Battery Design Capacity High 8 bits
    BSNL,    8, //Battery Serial Number Low 8 bits
    BSNH,    8  //Battery Serial Number High 8 bits
}
Device (BAT0)
{
    Name (_HID, EisaId ("PNP0C0A") /* Control Method Battery */) // _HID: Hardware
ID
    Name (_UID, Zero) // _UID: Unique ID
    Method (_STA, 0, NotSerialized) // _STA: Status
    {
        If (AST1)
        {
            Return (0x1F)
        }
        Else
        {
            Return (Zero)
        }
    }
}
Method (_BIF, 0, NotSerialized) // _BIF: Battery Information
{
    Name (BIFP, Package (0x0D))
    {
        One,
        0xFFFFFFFF,
        0xFFFFFFFF,
        One,
        0xFFFFFFFF,
        0xFFFFFFFF,
        0xFFFFFFFF,
        0xFFFFFFFF,
        0xFFFFFFFF,
        0xFFFFFFFF,
        "LoongsonM",
        "LoongsonS",
        "LI-ION",
        ""
    }
}
Locall = (BDVH * 0x0100)
Locall += BDVL

```

```

    BIFP [0x04] = Local1
    Local1 = (BDCH * 0x0100)
    Local1 += BDCL
    BIFP [One] = Local1
    Divide (Local1, 0x0A, Local2, BIFP [0x05])
    Divide (Local1, 0x14, Local2, BIFP [0x06])
    Divide (Local1, 0x64, Local2, BIFP [0x07])
    Divide (Local1, 0x64, Local2, BIFP [0x08])
    Local1 = (BFCH * 0x0100)
    Local1 += BFCL
    BIFP [0x02] = Local1
    Local1 = (BSNH * 0x0100)
    Local1 += BSNL
    BIFP [0x0A] = Local1
    Return (BIFP) /* \_SB_.PCI0.LPC_.EC___.BAT0._BIF.BIFP */
}
Method (_BST, 0, NotSerialized) // _BST: Battery Status
{
    Name (BSTP, Package (0x04)
    {
        0xFFFFFFFF,
        0xFFFFFFFF,
        0xFFFFFFFF,
        0xFFFFFFFF
    })
    If (RPWR)
    {
        If (BCG1)
        {
            BSTP [Zero] = 0x02
        }
        Else
        {
            BSTP [Zero] = Zero
        }
    }
    Else
    {
        BSTP [Zero] = One
    }
    Local1 = (BCH * 0x0100)
    BSTP [One] = (BCL + Local1)
    Local1 = (BRCH * 0x0100)

```

```

        BSTP [0x02] = (BRCL + Local1)
        Local1 = (BVH * 0x0100)
        BSTP [0x03] = (BVL + Local1)
        Return (BSTP) /* \_SB_.PCI0.LPC_.EC___.BAT0._BST.BSTP */
    }
}
}

```

## 8.6.6 电源适配器配置

支持的对象及方法:\_PSR。

通过EC（参见8.6.18.1节）管理电源适配器时，\_PSR方法的实现可以参照如下示例：

```

Device (EC)
{
    Name (_HID, EisaId ("PNP0C09") /* Embedded Controller Device */) // _HID: Hardware
ID
    Name (_UID, Zero) // _UID: Unique ID
    Name (_GPE, 0x04) // _GPE: General Purpose Events
    Name (_CRS, ResourceTemplate () // _CRS: Current Resource Settings
    {
        IO (Decode16, 0x0062, 0x0062, 0x01, 0x01,)
        IO (Decode16, 0x0066, 0x0066, 0x01, 0x01,)
    })
    OperationRegion (ECOR, EmbeddedControl, Zero, 0x0100)
    Field (ECOR, ByteAcc, NoLock, Preserve)
    {
        RPWR, 1 //Power Source Status
    }

Device (AC)
{
    Name (_HID, "ACPI0003" /* Power Source Device */) // _HID: Hardware ID
    Method (_PSR, 0, NotSerialized) // _PSR: Power Source
    {
        Return (RPWR) /* \_SB_.PCI0.LPC_.EC___.RPWR */
    }
}
}

```

## 8.6.7 处理器配置和控制

支持的对象及方法：\_HID、\_UID、\_PXM、\_STA、\_PPC、\_PCT、\_PSS。

(1) \_PCT(Performance Control)

定义ControlRegister和StatusRegister时，AddressSpaceKeyword必须为FFixedHW，其他域为任意值。

(2) \_PSS (Performance Supported States)

PSS中的Power、Bus master latency、Status为任意值。

Latency：大于20000

Control：

bits[31:9]：保留；

bit[8]：表示睿频标记，1表示睿频频率，0表示普频频率；

bits[7:0]：表示频率等级，范围3-10。

## 8.6.8 系统休眠唤醒

支持的对象及方法：\_S0、\_S3、\_S4、\_S5以及自定义的S3休眠地址。

S3休眠地址是系统S3休眠时进入固件执行休眠的入口地址，使用自定义对象声明，规定如下：

(1) 使用Name操作符定义的整数对象，对象名称：SADR。

(2) 该对象需要声明在\_SB下。

## 8.6.9 热键配置

龙芯平台提供了基于EC（参见8.6.18.1节）的统一热键驱动，主板厂商可通过热键映射实现热键差异化设计，规定如下：

(1) 在DSDT中定义热键设备，HID为LOON0000。

(2) 在热键设备中定义热键映射表，名称为“KMAP”，每个表项代表一个热键，表项格式约定如下：

表项类型为package，一个package由3个元素组成：按键类型、按键索引、按键键值，具体说明见表8-24：

表8-24 package元素表

域	说明
按键类型	1表示普通按键，操作系统键值表中前缀KEY_的按键为普通按键 2表示开关按键，操作系统键值表中前缀SW_的按键为开关按键
按键索引	按键标识，大于0的任意整数，用于唯一标识一个按键，不同按键不能重复

域	说明
按键键值	操作系统支持的键值码，参照附录A

(1) 处理EC热键事件产生时，需要实现EC热键事件号（EC执行QR\_EC命令后，EC\_DATA寄存器中存储的数值）对应的查询方法。实现查询方法时，需要通过Notify通知热键驱动，通知对象为定义的热键设备对象，通知数据为一个无符号16位数据，格式约定如表8-25：

表8-25 通知数据格式

域	说明
bits[15:12]	按键类型： 1表示普通按键 2表示开关按键
bits[11:0]	按键索引

(2) 对于开关类型的热键，需要声明名称为GSWS的方法，返回开关的状态。状态数据为32位数据，每一位表示一个SW类型开关对应的开关状态，SW类型开关的定义参考附录A。

其中，GSWS定义：

Method (GSWS, 0, NotSerialized)

作用：获取开盖状态

参数：无

返回值：32位正整数，开盖状态

通过EC（参见8.6.18.1节）管理开合盖时，可以参照如下示例：

```
Device (EC)
{
    Name (_HID, EisaId ("PNP0C09") /* Embedded Controller Device */) // _HID: Hardware
    ID
    Name (_UID, Zero) // _UID: Unique ID
    Name (_GPE, 0x04) // _GPE: General Purpose Events
    Name (_CRS, ResourceTemplate () // _CRS: Current Resource Settings
    {
        IO (Decode16, 0x0062, 0x0062, 0x01, 0x01,)
        IO (Decode16, 0x0066, 0x0066, 0x01, 0x01,)
    })
    OperationRegion (ECOR, EmbeddedControl, Zero, 0x0100)
    Field (ECOR, ByteAcc, NoLock, Preserve)
    {
        LSTE, 1 //Lid Status
    }
    Name (ELID, Zero)
```

```

Method (_Q21, 0, NotSerialized) // _Qxx: EC Query, xx=0x00-0xFF
{
    ELID = One
    Notify (HKEY, 0x2007)
}

Device (HKEY)
{
    Name (_HID, "LOON0000") // _HID: Hardware ID
    Name (KMAP, Package (One)
    {
        Package (0x03)
        {
            0x02,
            0x07,
            Zero//SW_LID
        }
    })

    Method (GSWS, 0, NotSerialized)
    {
        If (ELID)
        {
            ELID = Zero
            Return (NOr (LSTE, 0xFFFFFFFFFE))
        }
        Return (Zero)
    }
}
}

```

### (1) 背光控制

热键驱动支持三种背光控制方式：

当热键设备定义ECBS、ECBG、ECSL、ECLL、BLSW方法时，表示通过热键驱动定义背光设备控制背光；

当热键驱动检测到ACPI背光设备，则调用热键设备中的VCBL方法并传参，参数为false，表示由ACPI背光设备控制背光；

当热键驱动没有检测到ACPI背光设备，则由内置显卡背光设备控制背光。

其中，VCBL、ECBS、ECBG、ECSL、ECLL、BLSW定义如下：

VCBL:

Method (VCBL, 1, Serialized)



作用：设置背光事件通知对象

参数：8位正整数，0表示背光事件应有ACPI背光设备接收

返回值：无

ECBS:

Method (ECBS, 1, Serialized)

作用：设置背光亮度

参数：16位正整数，亮度值

返回值：无

ECBG:

Method (ECBG, 0, NotSerialized)

作用：获取背光亮度

参数：无

返回值：16位正整数，亮度值

ECSL:

Method (ECSL, 0, NotSerialized)

作用：获取支持的背光亮度最小值

参数：无

返回值：16位正整数，背光亮度最小值

ECLL:

Method (ECLL, 0, NotSerialized)

作用：获取支持的背光亮度最大值

参数：无

返回值：16位正整数，背光亮度最大值

BLSW:

Method (\_BLSW, 1, Serialized)

作用：控制系统背光开启/关闭

参数：16位正整数，0表示关，1表示开

返回值：无

通过EC（参见8.6.18.1节）管理屏幕背光时，可以参照如下示例：

```
Device (EC)
{
    Name (_HID, EisaId ("PNP0C09")) /* Embedded Controller Device */ // _HID: Hardware ID
    Name (_UID, Zero) // _UID: Unique ID
    Name (_GPE, 0x04) // _GPE: General Purpose Events
    Name (_CRS, ResourceTemplate () // _CRS: Current Resource Settings
    {
        IO (Decode16, 0x0062, 0x0062, 0x01, 0x01,)
```

```

    IO (Decode16, 0x0066, 0x0066, 0x01, 0x01,)
))
OperationRegion (ECOR, EmbeddedControl, Zero, 0x0100)
Field (ECOR, ByteAcc, NoLock, Preserve)
{
    BLAC,    8    //Backlight Adjust
}

Method (_Q38, 0, NotSerialized) // _Qxx: EC Query, xx=0x00-0xFF
{
    Notify (HKEY, 0x1005)
}

Method (_Q39, 0, NotSerialized) // _Qxx: EC Query, xx=0x00-0xFF
{
    Notify (HKEY, 0x1006)
}

Device (HKEY)
{
    Name (_HID, "LOON0000") // _HID: Hardware ID
    Name (KMAP, Package (0x02)
    {
        Package (0x03)
        {
            One,
            0x05,
            0xE0    //KEY_BRIGHTNESSDOWN
        },
        Package (0x03)
        {
            One,
            0x06,
            0xE1    //KEY_BRIGHTNESSUP
        }
    })
    Method (ECBS, 1, Serialized)
    {
        BLAC = Arg0
    }

    Method (ECBG, 0, NotSerialized)
    {

```

```

        Return (BLAC) /* \_SB_.PCI0.LPC_.EC_.BLAC */
    }

    Method (ECSL, 0, NotSerialized)
    {
        Return (0x05)
    }

    Method (ECLL, 0, NotSerialized)
    {
        Return (0x64)
    }
}
}

```

### 8.6.10 热区管理

支持的对象及方法：\_CRT、\_PSL、\_PSV、\_TC1、\_TC2、\_TMP、\_TSP、\_TZP。

### 8.6.11 串口配置

支持的对象及方法：\_HID, \_UID, \_DSD, \_CRS。

(1) \_HID

龙芯平台为PNP0501。

(2) \_UID

当使用处理器串口0时，\_UID的值必须为0，且此设备必须是DSDT中声明的第一个串口设备。

(3) \_CRS

\_CRS方法仅支持寄存器和中断资源声明。

寄存器资源使用QWordMemory()宏声明，传递寄存器地址信息，各个参数规定参考表8-21。中断号资源使用Interrupt()宏声明，各个参数的规定如表8-26：

表8-26 Interrupt()宏参数表

参数名	值
ResourceUsage	描述中断的使用者，该设备本身使用，或是子设备使用，为空，默认表示该设备本身使用
EdgeLevel	中断触发类型
ActiveLevel	中断触发极性
Shared	共享标志

参数名	值
ResourceSourceIndex	资源描述符索引
ResourceSource	资源描述符
DescriptorName	资源描述符名称
InterruptList	中断号

#### (4) \_DSD

\_DSD对象仅支持串口时钟频率声明：

UUID：值为"daffd814-6eba-4d8c-8a91-bc9bbf4aa301"。

支持属性如表8-27：

表8-27 支持属性表

属性名	值	说明
clock-frequency	串口时钟频率	表示实际外接的串口时钟频率，单位为Hz

## 8.6.12 GPIO配置

支持的对象及方法：\_HID，\_UID，\_DSD，\_CRS。

#### (1) \_HID

7A1000芯片组集成的普通GPIO为LOON0002，3A5000处理器集成的GPIO为LOON0007，2K2000处理器集成的GPIO为LOON000A，2K2000南北桥集成的GPIO为LOON000B，2K2000南北桥集成的ACPI GPIO为LOON000C，7A2000芯片组集成的普通GPIO为LOON000D，7A2000芯片组集成的ACPI GPIO为LOON000E，3A6000处理器集成的GPIO为LOON000F。

#### (2) \_CRS

\_CRS方法仅支持寄存器和中断资源声明。寄存器资源使用QWordMemory()宏声明，传递寄存器地址信息，各个参数规定如表8-21，中断号资源使用Interrupt()宏声明，各个参数的规定如表8-26。

#### (3) \_DSD

\_DSD对象仅支持GPIO如下属性：

UUID：值为"daffd814-6eba-4d8c-8a91-bc9bbf4aa301"。

支持属性如表8-28：

表8-28 支持属性如表

属性名	说明
ngpios	当前注册的gpio设备包含的gpio pin总数

注：具体配置约束见硬件设计规范章节

以7A2000 ACPI GPIO为例，DSDT表可参照如下配置：

```
Device (GPO2)
{
    Name (_HID, "LOON000E") // _HID: Hardware ID
    Name (_ADR, Zero) // _ADR: Address
    Name (_UID, One) // _UID: Unique ID
    Name (_CRS, ResourceTemplate () // _CRS: Current Resource Settings
    {
        QWordMemory (ResourceConsumer, PosDecode, MinFixed, MaxFixed, NonCacheable,
ReadWrite,
            0x0000000000000000, // Granularity
            0x00000E00100D0080, // Range Minimum
            0x00000E00100D009F, // Range Maximum
            0x0000000000000000, // Translation Offset
            0x0000000000000020, // Length
            ,, , AddressRangeMemory, TypeStatic)

        Interrupt (ResourceConsumer, Level, ActiveHigh, Exclusive, ,, )
        {
            0x0000006F,
        }

        Interrupt (ResourceConsumer, Level, ActiveHigh, Exclusive, ,, )
        {
            0x0000006F,
        }

        Interrupt (ResourceConsumer, Level, ActiveHigh, Exclusive, ,, )
        {
            0x0000006F,
        }

        Interrupt (ResourceConsumer, Level, ActiveHigh, Exclusive, ,, )
        {
            0x0000006F,
        }

        Interrupt (ResourceConsumer, Level, ActiveHigh, Exclusive, ,, )
        {
            0x0000006F,
        }

        Interrupt (ResourceConsumer, Level, ActiveHigh, Exclusive, ,, )
        {
            0x0000006F,
        }
    }
}
```



```
{
  ToUUID ("daffd814-6eba-4d8c-8a91-bc9bbf4aa301") /* Device Properties for _DSD */,
  Package (0x01)
  {
    Package (0x02)
    {
      "ngpios",
      16
    }
  }
})
}
```

### 8.6.13 I2C配置

支持的对象及方法：\_HID，\_UID，\_CRS。

(1) \_HID

龙芯平台为LOON0004。

(2) \_UID

必须为表示I2C总线号的任意正整数。

(3) \_CRS

\_CRS方法仅支持寄存器资源声明。寄存器资源使用QWordMemory()宏声明，传递寄存器地址信息，各个参数规定如表8-21。

注：具体配置约束见硬件设计规范章节。

### 8.6.14 GPIO模拟I2C配置

支持的对象及方法：\_HID，\_UID，\_DSD，\_CSR。

(1) \_HID

龙芯平台为LOON0005。

(2) \_UID

必须为表示I2C总线号的任意正整数。

(3) \_DSD

\_DSD对象仅支持如下属性：

UUID：值为"daffd814-6eba-4d8c-8a91-bc9bbf4aa301"。

支持属性如表8-29：

表8-29 支持属性表

属性名	说明
-----	----

sda-gpio	指定SDA信号线使用的GPIO管脚
scl-gpio	指定SCL信号线使用的GPIO管脚
delay-us	总线传输时钟周期，单位为us
timeout-ms	一次总线数据传输允许最大超时时间，单位为ms，此为可选参数

### 8.6.15 RTC配置

支持的对象及方法：\_HID，\_CRS。

(1) \_HID

龙芯平台为LOON0001。

(2) \_CRS

\_CRS方法仅支持寄存器和中断资源声明。寄存器资源使用QWordMemory()宏声明，传递寄存器地址信息，各个参数规定如表8-21，中断号资源使用Interrupt()宏声明，各个参数的规定如表8-26。

### 8.6.16 PWM配置

支持的对象及方法：\_HID，\_UID，\_CRS。

(1) \_HID

龙芯平台为LOON0006。

(2) \_CRS

\_CRS方法仅支持寄存器和中断资源声明。寄存器资源使用QWordMemory()宏声明，传递寄存器地址信息，各个参数规定如表8-21，中断号资源使用Interrupt()宏声明，各个参数的规定如表8-26。

### 8.6.17 SE设备

支持的对象及方法：\_HID、\_CRS。

(1) \_HID

龙芯平台为LOON0003。

(2) \_CRS

\_CRS方法仅支持中断资源声明。中断号资源使用Interrupt()宏声明，各个参数的规定如表8-26。

### 8.6.18 LPC配置

连接在LPC下的设备，需要通过\_DEP明确描述依赖的设备。



### 8.6.18.1 EC设备

支持的对象及方法：\_CRS, HID, GPE, \_Qxx, \_DEP。

EC (Embedded Controller) 设备需要描述在 LPC设备下, 并通过\_DEP明确描述依赖的设备, 可以通过EC管理电池, 热键, 键盘, 鼠标等设备。管理电池可以参照8.6.5节中示例, 热键管理及背光调节可以参照8.6.9节中示例, 管理键盘, 鼠标时可以参照如下示例:

```
Device (LPC)
{
    Name (_ADR, 0x00170000) // _ADR: Address
    Name (_S3D, 0x03) // _S3D: S3 Device State
    Name (RID, Zero)
    Device (EC)
    {
        Name (_HID, EisaId ("PNP0C09") /* Embedded Controller Device */) // _HID:
Hardware ID
        Name (_UID, Zero) // _UID: Unique ID
        Name (_DEP, Package() {\SB.PCI0})
        Name (_GPE, 0x04) // _GPE: General Purpose Events
        Name (_CRS, ResourceTemplate () // _CRS: Current Resource Settings
        {
            IO(Decode16, 0x62, 0x62, 0, 1,)
            IO(Decode16, 0x66, 0x66, 0, 1,)
        })

        Device (KBD)
        {
            Method (_HID, 0, NotSerialized) // _HID: Hardware ID
            {
                Return (0x0303D041)
            }

            Name (_CID, EisaId ("PNP0303") /* IBM Enhanced Keyboard (101/102-key, PS/2
Mouse) */) // _CID: Compatible ID
            Name (_CRS, ResourceTemplate () // _CRS: Current Resource Settings
            {
                IO(Decode16, 0x60, 0x60, 1, 1,)
                IO(Decode16, 0x64, 0x64, 1, 1,)
                IRQ (Level, ActiveHigh, Exclusive, )
                {1}
            })
        }
    }
}
```

```
Device (MOU)
{
    Name (_HID, EisaId ("LEN2020")) // _HID: Hardware ID
    Name (_CID, EisaId ("PNP0F13") /* PS/2 Mouse */) // _CID: Compatible ID
    Name (_CRS, ResourceTemplate () // _CRS: Current Resource Settings
    {
        IRQ (Level, ActiveHigh, Exclusive, )
            {12}
    })
}
}
```

### 8.6.19 BMC设备

支持的对象及方法：\_STR、\_CRS, HID, \_IFT, \_SRV。

BMC设备需要描述在 LPC 设备下。

### 8.6.20 DMA配置

龙芯平台默认物理地址与DMA地址相同，龙芯服务器为了兼容问题设备（能力寄存器标称支持64位DMA能力，但实际上不支持64位DMA寻址能力，比如仅支持38位DMA地址寻址能力），固件可通过\_DMA对象配置物理地址与DMA地址转换偏移。

以2节点为例子，0节点和1节点内存范围分别是0x20000 - 0x47ffffff, 0x100080000000 - 0x10047ffffff,若问题设备最大可支持38位地址，则需要将1位节点号放在DMA地址的bit37，上述物理地址范围对应的DMA地址范围分别是0x20000 - 0x47ffffff, 0x2080000000 - 0x247ffffff，0节点的节点号为0，所以物理地址与DMA地址相同，1节点物理地址于DMA地址的转换偏移为：0x100080000000-0x2080000000=0xfe00000000，相应的\_DMA对象描述如下：

```
Name (_DMA, ResourceTemplate() {
    QWordMemory (ResourceProducer,
        PosDecode,
        MinFixed,
        MaxFixed,
        NonCacheable,
        ReadWrite,
        0x0,
        0x0000200000,
        0x047ffffff,
```

```

        0x0000000000,
        0x47fe00000,
        ,
        ,
    )
    QWordMemory (ResourceProducer,
        PosDecode,
        MinFixed,
        MaxFixed,
        NonCacheable,
        ReadWrite,
        0x0,
        0x100080000000,
        0x10047ffffff,
        0xfe00000000,
        0x400000000,
        ,
        ,
    )
})

```

### 8.6.21 SDIO/MMC配置

支持的对象及方法: \_HID, \_UID, \_DSD, \_CSR。

(1)\_HID

龙芯平台为 PRP0001。

(2)\_DSD

\_DSD 对象仅支持如下属性: UUID:值为"daffd814-6eba-4d8c-8a91-bc9bbf4aa301"。

compatible: linux内核中SDIO驱动的匹配标识

dma-mask: 支持的dma位数

clock-frequency: 时钟频率

bus-width: 总线宽度

### 8.6.22 I2S音频配置

I2S声卡需要通过I2S控制器设备、Codec设备和Machine设备实现配置，下面以codec下挂在芯片组i2c1控制器举例，配置实例如下：

```

Device (I2S0)
{

```

```

Name (_ADR, 0x00070000) // _ADR: Address
Name (_CRS, ResourceTemplate () // _CRS: Current Resource Settings
{
    Interrupt (ResourceConsumer, Level, ActiveHigh, Shared, ,, )
    {
        0x00000078,
        0x00000079,
    }
})
Name (_DSD, Package (0x02) // _DSD: Device-Specific Data
{
    ToUUID ("daffd814-6eba-4d8c-8a91-bc9bbf4aa301") /* Device Properties for _DSD */,
    Package (0x02)
    {
        Package (0x02)
        {
            "clock-frequency",
            0x02FAF080
        },

        Package (0x02)
        {
            "interrupt-names",
            Package (0x02)
            {
                "tx",
                "rx"
            }
        }
    }
})
}

Device (I2C1)
{
    Name (_HID, "LOON0004" /* AT Real-Time Clock */)
    Name (_UID, 0x1) // _UID: Unique ID
    Name (_CRS, ResourceTemplate ()
    {
        QWordMemory (ResourceConsumer, PosDecode, MinFixed, MaxFixed, NonCacheable,
        ReadWrite,
            0x0000000000000000,
            0x0000000010090100,

```

```

        0x0000000010090107,
        0x0000000000000000,
        0x0000000000000008,
        ,, , AddressRangeMemory, TypeStatic)
Interrupt (ResourceConsumer, Level, ActiveHigh, Shared, ,, )
{
    73,
}
}))
Device(CODC)
{
    Name (_HID,"ESSX8323")
    Name (_CRS,ResourceTemplate()
    {
        I2cSerialBusV2 (0x0010, ControllerInitiated, 0x000186A0, AddressingMode7Bit,
        "\\_SB.I2CI", 0x00, ResourceConsumer, , Exclusive, )
    })
}
}

Device (PCA0)
{
    Name (_HID, "PRP0001") // _HID: Hardware ID
    Name (_UID, Zero) // _UID: Unique ID
    Name (_DSD, Package (0x02) // _DSD: Device-Specific Data
    {
        ToUUID ("daffd814-6eba-4d8c-8a91-bc9bbf4aa301") /* Device Properties for _DSD */,
        Package (0x06)
        {
            Package (0x02)
            {
                "compatible",
                "loongson,ls-audio-card"
            },

            Package (0x02)
            {
                "model",
                "Loongson-ASoC"
            },

            Package (0x02)
            {

```

```
        "mclk-fs",
        0x0100
    },

    Package (0x02)
    {
        "cpu",
        ^PCI0.I2S0
    },

    Package (0x02)
    {
        "codec",
        ^I2C1.CODC
    },

    Package (0x02)
    {
        "codec-dai-name",
        "ES8326 HiFi"
    }
}
}))
}
```

同时，固件需要根据芯片手册配置正确的I2S管脚复用，并禁止HDA设备,以避免内核中PCI扫描 阶段扫描到HDA设备。

### 8.6.23 SPI配置

(1)\_DSD

\_DSD 对象仅支持如下属性:

UUID:值为"daffd814-6eba-4d8c-8a91-bc9bbf4aa301"。

clock-frequency: 时钟频率

### 8.6.24 内存ECC校验配置

支持的对象及方法:\_HID,\_CSR。

(1)\_HID

龙芯平台为LOON0010。

(2)\_CRS

\_CRS方法仅支持寄存器资源声明。

寄存器资源使用QWordMemory()宏声明，传递寄存器地址信息，各个参数规定参考表8-21。

以3C6000为例，ECC可以参照如下配置：

```
Device (ECC)
{
    Name (_HID, "LOON0010") // _HID: Hardware ID
    Name (_CRS, ResourceTemplate () // _CRS: Current Resource Settings
    {
        QWordMemory (ResourceConsumer, PosDecode, MinFixed, MaxFixed, NonCacheable,
ReadWrite,
            0x0000000000000000, // Granularity
            0x000000001FE00600, // Range Minimum
            0x000000001FE006FF, // Range Maximum
            0x0000000000000000, // Translation Offset
            0x0000000000000100, // Length
            ,, , AddressRangeMemory, TypeStatic)
    })
}
```

## 8.6.25 龙芯SE设备配置

支持的对象及方法:\_HID,\_UID,\_CSR,\_DSD。

(1)\_HID

龙芯平台为LOON0011。

(2)\_CRS

\_CRS方法仅支持寄存器和中断资源声明。

寄存器资源使用QWordMemory()宏声明，传递寄存器地址信息，各个参数规定参考表8-21。

中断号资源使用Interrupt()宏声明，各个参数的规定如表8-26。

(3)\_DSD

\_DSD 对象支持如下属性：

UUID：值为"daffd814-6eba-4d8c-8a91-bc9bbf4aa301"。

dmam\_size：描述需要分配的DMA内存大小。

### 8.6.25.1 SDF设备配置

支持的对象及方法:\_HID,\_UID,\_DSD。

(1)\_HID

龙芯平台为LOON0012。

(2)\_DSD

\_DSD 对象支持如下属性:

UUID: 值为"daffd814-6eba-4d8c-8a91-bc9bbf4aa301"。

channel: 描述SDF设备连接到SE设备的通道。

以3C6000为例, SE和SDF可以参照如下配置:

```
Device (SE)
{
    Name (_HID, "LOON0011") // _HID: Hardware ID
    Name (_UID, Zero) // _UID: Unique ID
    Name (_CRS, ResourceTemplate () // _CRS: Current Resource Settings
    {
        QWordMemory (ResourceConsumer, PosDecode, MinFixed, MaxFixed, NonCacheable,
ReadWrite,
            0x0000000000000000, // Granularity
            0x00000C00E0000000, // Range Minimum
            0x00000C00E0000FFF, // Range Maximum
            0x0000000000000000, // Translation Offset
            0x0000000000001000, // Length
            ,, , AddressRangeMemory, TypeStatic)
        Interrupt (ResourceConsumer, Level, ActiveHigh, Shared, ,, )
        {
            0x00000020,
            0x00000021,
            0x00000022,
            0x00000023,
            0x00000024,
            0x00000025,
            0x00000026,
            0x00000027,
        }
    })
    Name (_DSD, Package (0x02) // _DSD: Device-Specific Data
    {
        ToUUID ("daffd814-6eba-4d8c-8a91-bc9bbf4aa301") /* Device Properties for _DSD
*/,
        Package (0x01)
        {
            Package (0x02)
            {
                "dmam_size",
                0x00800000
            }
        }
    }
}
```



```
    }  
  })  
  Device (SDF)  
  {  
    Name (_HID, "LOON0012") // _HID: Hardware ID  
    Name (_UID, One) // _UID: Unique ID  
    Name (_DSD, Package (0x02) // _DSD: Device-Specific Data  
    {  
      ToUUID ("daffd814-6eba-4d8c-8a91-bc9bbf4aa301") /* Device Properties for  
_DSD */,  
      Package (0x01)  
      {  
        Package (0x02)  
        {  
          "channel",  
          0x07  
        }  
      }  
    }  
  })  
}  
}
```

## 8.7 FACS

FACS (Firmware ACPI Control Structure) 主要包含了唤醒向量地址，此表必须实现，其中部分 约定如表8-30，其他字段保留为0。

表8-30 FACS表约束

域	大小 (字节)	偏移 (字节)	描述
Signature	4	0	‘FACS’
Firmware Waking Vector	4	12	32位唤醒向量
Global Lock	4	16	全局锁
Flags	4	20	固件控制标志
X Firmware Waking Vector	8	24	64位唤醒向量
Version	1	32	FACS的版本

## 8.8 MCFG

MCFG (PCI Express Memory-mapped Configuration Space base address description table) 主要包含了可访问到PCIE设备扩展配置空间基地址、PCI domain域ID及总线范围。表头部分约定如表8-31。PCIE配置信息部分约定如表8-32。

表 8-31 MCFG约束

域	大小 (字节)	偏移 (字节)	描述
Signature	4	0	‘MCFG’
Revision	1	8	0x1

表 8-32 Memory Mapped Enhanced Configuration Space Base Address Allocation Structure约束

域	大小 (字节)	偏移 (字节)	描述
Base Address	8	0	PCI域扩展配置空间基地址
PCI Segment Group Number	2	8	PCI域ID, 注: 此值需与DSDT中_SEG对象的值一致
Start Bus Number	1	10	主桥下总线号最小值
End Bus Number	1	11	主桥下总线号最大值

## 8.9 SLIT

SLIT (System Locality Distance Information Table) 主要描述了系统不同节点间相对距离的信息, 此表为可选表项。表部分约束如表3-33。

表 8-33 SLIT约束

域	大小 (字节)	偏移 (字节)	描述
Signature	4	0	‘SLIT’
Number of System Localities	8	36	节点数
Entry[x][y]	1		当x = y, 相同节点相对距离 当x != y, 不同节点相对距离 xy取值范围: Number of System Localities-1

## 8.10 SPCR

SPCR (Serial Port Console Redirection Table) 主要包含了串口配置，此表选择实现，其中部分约定如表8-34。

表8-34 SPCR表部分约束

域	大小 (字节)	偏移 (字节)	描述
Signature	4	0	‘SPCR’
Interface Type	1	36	接口类型
Space ID	1	40	地址空间类型
Encoded Access Width	1	43	访问带宽编码
Address	8	44	基地址
Baud Rate	1	58	波特率

## 8.11 PPTT

PPTT (Processor Properties Topology Table) 主要描述了处理器的拓扑结构，此表为可选表项，其中表头的约束见表8-35，Processor hierarchy node structure约束见表8-36。

表8-35 PPTT约束

域	大小 (字节)	偏移 (字节)	描述
Header			
Signature	4	0	‘PPTT’
Revision	1	8	3

表8-36 Processor hierarchy node structure约束

域	大小 (字节)	偏移 (字节)	描述
Type	1	0	0表示处理器结构
Length	1	1	本地处理器结构的长度，以字节为单位
Reserved	2	2	保留
Flags	4	4	处理器结构标志，见表8-37

Parent	4	8	父处理器节点结构引用，值为PPTT表的开始和父处理器结构项的开始之间的差值。当没有父节点时，值为0。
ACPI Processor ID	4	12	MADT表中的处理器核 UID。对于支持smt的处理器，如果该结构表示的processor是一个thread，该域和MADT表中的处理器核UID一致；如果该结构表示的processor是一个物理core，即a group of associated processor thread。该域中的值是物理core id。
Number of private resources	4	16	私有资源结构的个数
Private resources[N]	N*4	20	每个资源结构都表示处理器层次结构节点私有的资源

表8-37 处理器结构标志

域	大小（字节）	偏移（字节）	描述
Physical package	1	0	如果处理器拓扑节点表示物理封装边界，则设置为1； 如果处理器拓扑节点不表示物理封装边界，则设置为0
ACPI Processor ID valid	1	1	该标志表示ACPI Processor ID是否指向有效的条目
Processor is a Thread	1	2	对于叶子表项，如果表示该处理器的处理元素与兄弟节点共享功能单元，则必须设置为1； 对于非叶子表项，必须设置为0
Node is a Leaf	1	3	如果node是处理器层次结构中的叶子节点，则必须设置为1，否则必须设置为0
Identical Implementation	1	4	值为1表示所有子处理器共享相同的实现
Reserved	2	5	0

## 9 VBIOS的传递约定

对于内置显卡，固件需要将VBIOS存放到内置显卡BAR2空间的最后1MB内。

## 10 screen\_info传参约定

screen\_info为内核framebuffer设备提供信息，固件提供EFI\_GRAPHICS\_OUTPUT\_PROTOCOL\_GUID（简称GOP）时，BootLoader通过GOP获取screen\_info，

并注册自定义screen\_info表，传递给内核。screen\_info表GUID：

EFI\_GUID(0xe03fc20a, 0x85dc, 0x406e, 0xb9, 0x0e, 0x4a, 0xb5, 0x02, 0x37, 0x1d, 0x95)。

screen\_info定义了显存基地址、分辨率、颜色位宽等，和Linux内核struct screen\_info 保持一致，其数据结构定义如下表：

```
struct screen_info {
    UINT8  orig_x;           /* 0x00 */
    UINT8  orig_y;           /* 0x01 */
    UINT16 ext_mem_k;        /* 0x02 */
    UINT16 orig_video_page; /* 0x04 */
    UINT8  orig_video_mode; /* 0x06 */
    UINT8  orig_video_cols; /* 0x07 */
    UINT8  flags;            /* 0x08 */
    UINT8  unused2;          /* 0x09 */
    UINT16 orig_video_ega_bx; /* 0x0a */
    UINT16 unused3;          /* 0x0c */
    UINT8  orig_video_lines; /* 0x0e */
    UINT8  orig_video_isVGA; /* 0x0f */
    UINT16 orig_video_points; /* 0x10 */

    /* VESA graphic mode -- linear frame buffer */
    UINT16 lfb_width;        /* 0x12 */
    UINT16 lfb_height;       /* 0x14 */
    UINT16 lfb_depth;        /* 0x16 */
    UINT32 lfb_base;         /* 0x18 */
    UINT32 lfb_size;         /* 0x1c */
    UINT16 cl_magic, cl_offset; /* 0x20 */
    UINT16 lfb_linelength;    /* 0x24 */
    UINT8  red_size;         /* 0x26 */
    UINT8  red_pos;          /* 0x27 */
    UINT8  green_size;       /* 0x28 */
    UINT8  green_pos;        /* 0x29 */
    UINT8  blue_size;        /* 0x2a */
    UINT8  blue_pos;         /* 0x2b */
    UINT8  rsvd_size;        /* 0x2c */
    UINT8  rsvd_pos;         /* 0x2d */
    UINT16 vesapm_seg;       /* 0x2e */
    UINT16 vesapm_off;       /* 0x30 */
    UINT16 pages;            /* 0x32 */
    UINT16 vesa_attributes; /* 0x34 */
    UINT32 capabilities;     /* 0x36 */
    UINT8  _reserved[6];     /* 0x3a */
}
```

```
};
```

## 11 多核启动传参约定

主核唤醒从核，通过处理器核间中断和通信机制实现。约定使用被唤醒从核的MailBox0寄存器传递启动参数，且使用IPI中断向量号0向被唤醒从核发送中断。从核被唤醒后，跳转到需要执行代码前的状态约定为：最高特权等级（CRMD[1:0]清零）、关闭全局中断（CRMD[2]清零）。

## 12 总结

本传参规范的提出和制定旨在规范龙芯的固件和内核接口，重点规定了龙芯平台的地址空间划分、中断分配、传参数据结构、ACPI、VBIOS、SMBIOS约定等方面，使得内核具有更广泛的适应性及兼容性，能有效解决内核对具体板卡设备的依赖性，有利于龙芯系列产品基础软件的规范和统一。

## 附录A LINUX操作系统键值表

按键名称	键值
KEY_RESERVED	0
KEY_ESC	1
KEY_1	2
KEY_2	3
KEY_3	4
KEY_4	5
KEY_5	6
KEY_6	7
KEY_7	8
KEY_8	9
KEY_9	10
KEY_0	11
KEY_MINUS	12
KEY_EQUAL	13

KEY_BACKSPACE	14
KEY_TAB	15
KEY_Q	16
KEY_W	17
KEY_E	18
KEY_R	19
KEY_T	20
KEY_Y	21
KEY_U	22
KEY_I	23
KEY_O	24
KEY_P	25
KEY_LEFTBRACE	26
KEY_RIGHTBRACE	27
KEY_ENTER	28
KEY_LEFTCTRL	29
KEY_A	30
KEY_S	31
KEY_D	32
KEY_F	33
KEY_G	34
KEY_H	35
KEY_J	36
KEY_K	37
KEY_L	38
KEY_SEMICOLON	39
KEY_APOSTROPHE	40
KEY_GRAVE	41
KEY_LEFTSHIFT	42
KEY_BACKSLASH	43

KEY_Z	44
KEY_X	45
KEY_C	46
KEY_V	47
KEY_B	48
KEY_N	49
KEY_M	50
KEY_COMMA	51
KEY_DOT	52
KEY_SLASH	53
KEY_RIGHTSHIFT	54
KEY_KPASTERISK	55
KEY_LEFTALT	56
KEY_SPACE	57
KEY_CAPSLOCK	58
KEY_F1	59
KEY_F2	60
KEY_F3	61
KEY_F4	62
KEY_F5	63
KEY_F6	64
KEY_F7	65
KEY_F8	66
KEY_F9	67
KEY_F10	68
KEY_NUMLOCK	69
KEY_SCROLLLOCK	70
KEY_KP7	71
KEY_KP8	72
KEY_KP9	73



KEY_KPMINUS	74
KEY_KP4	75
KEY_KP5	76
KEY_KP6	77
KEY_KPPLUS	78
KEY_KP1	79
KEY_KP2	80
KEY_KP3	81
KEY_KP0	82
KEY_KPDOT	83
KEY_ZENKAKUHANKAKU	85
KEY_I02ND	86
KEY_F11	87
KEY_F12	88
KEY_RO	89
KEY_KATAKANA	90
KEY_HIRAGANA	91
KEY_HENKAN	92
KEY_KATAKANAHIRAGANA	93
KEY_MUHENKAN	94
KEY_KPJPCOMMA	95
KEY_KPENTER	96
KEY_RIGHTCTRL	97
KEY_KPSLASH	98
KEY_SYSRQ	99
KEY_RIGHTALT	100
KEY_LINEFEED	101
KEY_HOME	102
KEY_UP	103
KEY_PAGEUP	104

KEY_LEFT	105
KEY_RIGHT	106
KEY_END	107
KEY_DOWN	108
KEY_PAGEDOWN	109
KEY_INSERT	110
KEY_DELETE	111
KEY_MACRO	112
KEY_MUTE	113
KEY_VOLUMEDOWN	114
KEY_VOLUMEUP	115
KEY_POWER	116
KEY_KPEQUAL	117
KEY_KPPLUSMINUS	118
KEY_PAUSE	119
KEY_SCALE	120
KEY_KPCOMMA	121
KEY_HANGEUL	122
KEY_HANGUEL	KEY_HANGEUL
KEY_HANJA	123
KEY_YEN	124
KEY_LEFTMETA	125
KEY_RIGHTMETA	126
KEY_COMPOSE	127
KEY_STOP	128
KEY_AGAIN	129
KEY_PROPS	130
KEY_UNDO	131
KEY_FRONT	132
KEY_COPY	133

KEY_OPEN	134
KEY_PASTE	135
KEY_FIND	136
KEY_CUT	137
KEY_HELP	138
KEY_MENU	139
KEY_CALC	140
KEY_SETUP	141
KEY_SLEEP	142
KEY_WAKEUP	143
KEY_FILE	144
KEY_SENDFILE	145
KEY_DELETEFILE	146
KEY_XFER	147
KEY_PROG1	148
KEY_PROG2	149
KEY_WWW	150
KEY_MSDOS	151
KEY_COFFEE	152
KEY_SCREENLOCK	KEY_COFFEE
KEY_ROTATE_DISPLAY	153
KEY_DIRECTION	KEY_ROTATE_DISPLAY
KEY_CYCLEWINDOWS	154
KEY_MAIL	155
KEY_BOOKMARKS	156
KEY_COMPUTER	157
KEY_BACK	158
KEY_FORWARD	159
KEY_CLOSECD	160
KEY_EJECTCD	161

KEY_EJECTCLOSECD	162
KEY_NEXTSONG	163
KEY_PLAYPAUSE	164
KEY_PREVIOUSSONG	165
KEY_STOPCD	166
KEY_RECORD	167
KEY_REWIND	168
KEY_PHONE	169
KEY_ISO	170
KEY_CONFIG	171
KEY_HOMEPAGE	172
KEY_REFRESH	173
KEY_EXIT	174
KEY_MOVE	175
KEY_EDIT	176
KEY_SCROLLUP	177
KEY_SCROLLDOWN	178
KEY_KPLEFTPAREN	179
KEY_KPRIGHTPAREN	180
KEY_NEW	181
KEY_REDO	182
KEY_F13	183
KEY_F14	184
KEY_F15	185
KEY_F16	186
KEY_F17	187
KEY_F18	188
KEY_F19	189
KEY_F20	190
KEY_F21	191

KEY_F22	192
KEY_F23	193
KEY_F24	194
KEY_PLAYCD	200
KEY_PAUSECD	201
KEY_PROG3	202
KEY_PROG4	203
KEY_DASHBOARD	204
KEY_SUSPEND	205
KEY_CLOSE	206
KEY_PLAY	207
KEY_FASTFORWARD	208
KEY_BASSBOOST	209
KEY_PRINT	210
KEY_HP	211
KEY_CAMERA	212
KEY_SOUND	213
KEY_QUESTION	214
KEY_EMAIL	215
KEY_CHAT	216
KEY_SEARCH	217
KEY_CONNECT	218
KEY_FINANCE	219
KEY_SPORT	220
KEY_SHOP	221
KEY_ALTERASE	222
KEY_CANCEL	223
KEY_BRIGHTNESSDOWN	224
KEY_BRIGHTNESSUP	225
KEY_MEDIA	226

KEY_SWITCHVIDEOMODE	227
KEY_KBDILLUMTOGGLE	228
KEY_KBDILLUMDOWN	229
KEY_KBDILLUMUP	230
KEY_SEND	231
KEY_REPLY	232
KEY_FORWARDMAIL	233
KEY_SAVE	234
KEY_DOCUMENTS	235
KEY_BATTERY	236
KEY_BLUETOOTH	237
KEY_WLAN	238
KEY_UWB	239
KEY_UNKNOWN	240
KEY_VIDEO_NEXT	241
KEY_VIDEO_PREV	242
KEY_BRIGHTNESS_CYCLE	243
KEY_BRIGHTNESS_AUTO	244
KEY_BRIGHTNESS_ZERO	KEY_BRIGHTNESS_AUTO
KEY_DISPLAY_OFF	245
KEY_WWAN	246
KEY_WIMAX	KEY_WWAN
KEY_RFKILL	247
KEY_MICMUTE	248
KEY_OK	0x160
KEY_SELECT	0x161
KEY_GOTO	0x162
KEY_CLEAR	0x163
KEY_POWER2	0x164
KEY_OPTION	0x165

KEY_INFO	0x166
KEY_TIME	0x167
KEY_VENDOR	0x168
KEY_ARCHIVE	0x169
KEY_PROGRAM	0x16a
KEY_CHANNEL	0x16b
KEY_FAVORITES	0x16c
KEY_EPG	0x16d
KEY_PVR	0x16e
KEY_MHP	0x16f
KEY_LANGUAGE	0x170
KEY_TITLE	0x171
KEY_SUBTITLE	0x172
KEY_ANGLE	0x173
KEY_ZOOM	0x174
KEY_MODE	0x175
KEY_KEYBOARD	0x176
KEY_SCREEN	0x177
KEY_PC	0x178
KEY_TV	0x179
KEY_TV2	0x17a
KEY_VCR	0x17b
KEY_VCR2	0x17c
KEY_SAT	0x17d
KEY_SAT2	0x17e
KEY_CD	0x17f
KEY_TAPE	0x180
KEY_RADIO	0x181
KEY_TUNER	0x182
KEY_PLAYER	0x183

KEY_TEXT	0x184
KEY_DVD	0x185
KEY_AUX	0x186
KEY_MP3	0x187
KEY_AUDIO	0x188
KEY_VIDEO	0x189
KEY_DIRECTORY	0x18a
KEY_LIST	0x18b
KEY_MEMO	0x18c
KEY_CALENDAR	0x18d
KEY_RED	0x18e
KEY_GREEN	0x18f
KEY_YELLOW	0x190
KEY_BLUE	0x191
KEY_CHANNELUP	0x192
KEY_CHANNELDOWN	0x193
KEY_FIRST	0x194
KEY_LAST	0x195
KEY_AB	0x196
KEY_NEXT	0x197
KEY_RESTART	0x198
KEY_SLOW	0x199
KEY_SHUFFLE	0x19a
KEY_BREAK	0x19b
KEY_PREVIOUS	0x19c
KEY_DIGITS	0x19d
KEY_TEEN	0x19e
KEY_TWEN	0x19f
KEY_VIDEOPHONE	0x1a0
KEY_GAMES	0x1a1



KEY_ZOOMIN	0x1a2
KEY_ZOOMOUT	0x1a3
KEY_ZOOMRESET	0x1a4
KEY_WORDPROCESSOR	0x1a5
KEY_EDITOR	0x1a6
KEY_SPREADSHEET	0x1a7
KEY_GRAPHICSEDITOR	0x1a8
KEY_PRESENTATION	0x1a9
KEY_DATABASE	0x1aa
KEY_NEWS	0x1ab
KEY_VOICEMAIL	0x1ac
KEY_ADDRESSBOOK	0x1ad
KEY_MESSENGER	0x1ae
KEY_DISPLAYTOGGLE	0x1af
KEY_BRIGHTNESS_TOGGLE	KEY_DISPLAYTOGGLE
KEY_SPELLCHECK	0x1b0
KEY_LOGOFF	0x1b1
KEY_DOLLAR	0x1b2
KEY_EURO	0x1b3
KEY_FRAMEBACK	0x1b4
KEY_FRAMEFORWARD	0x1b5
KEY_CONTEXT_MENU	0x1b6
KEY_MEDIA_REPEAT	0x1b7
KEY_10CHANNELSUP	0x1b8
KEY_10CHANNELSDOWN	0x1b9
KEY_IMAGES	0x1ba
KEY_DEL_EOL	0x1c0
KEY_DEL_EOS	0x1c1
KEY_INS_LINE	0x1c2
KEY_DEL_LINE	0x1c3

KEY_FN	0xd0
KEY_FN_ESC	0xd1
KEY_FN_F1	0xd2
KEY_FN_F2	0xd3
KEY_FN_F3	0xd4
KEY_FN_F4	0xd5
KEY_FN_F5	0xd6
KEY_FN_F6	0xd7
KEY_FN_F7	0xd8
KEY_FN_F8	0xd9
KEY_FN_F9	0xda
KEY_FN_F10	0xdb
KEY_FN_F11	0xdc
KEY_FN_F12	0xdd
KEY_FN_1	0xde
KEY_FN_2	0xdf
KEY_FN_D	0xe0
KEY_FN_E	0xe1
KEY_FN_F	0xe2
KEY_FN_S	0xe3
KEY_FN_B	0xe4
KEY_BRL_DOT1	0xf1
KEY_BRL_DOT2	0xf2
KEY_BRL_DOT3	0xf3
KEY_BRL_DOT4	0xf4
KEY_BRL_DOT5	0xf5
KEY_BRL_DOT6	0xf6
KEY_BRL_DOT7	0xf7
KEY_BRL_DOT8	0xf8
KEY_BRL_DOT9	0xf9

KEY_BRL_DOT10	0x1fa
KEY_NUMERIC_0	0x200
KEY_NUMERIC_1	0x201
KEY_NUMERIC_2	0x202
KEY_NUMERIC_3	0x203
KEY_NUMERIC_4	0x204
KEY_NUMERIC_5	0x205
KEY_NUMERIC_6	0x206
KEY_NUMERIC_7	0x207
KEY_NUMERIC_8	0x208
KEY_NUMERIC_9	0x209
KEY_NUMERIC_STAR	0x20a
KEY_NUMERIC_POUND	0x20b
KEY_NUMERIC_A	0x20c
KEY_NUMERIC_B	0x20d
KEY_NUMERIC_C	0x20e
KEY_NUMERIC_D	0x20f
KEY_CAMERA_FOCUS	0x210
KEY_WPS_BUTTON	0x211
KEY_TOUCHPAD_TOGGLE	0x212
KEY_TOUCHPAD_ON	0x213
KEY_TOUCHPAD_OFF	0x214
KEY_CAMERA_ZOOMIN	0x215
KEY_CAMERA_ZOOMOUT	0x216
KEY_CAMERA_UP	0x217
KEY_CAMERA_DOWN	0x218
KEY_CAMERA_LEFT	0x219
KEY_CAMERA_RIGHT	0x21a
KEY_ATTENDANT_ON	0x21b
KEY_ATTENDANT_OFF	0x21c

KEY_ATTENDANT_TOGGLE	0x21d
KEY_LIGHTS_TOGGLE	0x21e
KEY_ALS_TOGGLE	0x230
KEY_ROTATE_LOCK_TOGGLE	0x231
KEY_BUTTONCONFIG	0x240
KEY_TASKMANAGER	0x241
KEY_JOURNAL	0x242
KEY_CONTROLPANEL	0x243
KEY_APPSELECT	0x244
KEY_SCREENSAVER	0x245
KEY_VOICECOMMAND	0x246
KEY_ASSISTANT	0x247
KEY_BRIGHTNESS_MIN	0x250
KEY_BRIGHTNESS_MAX	0x251
KEY_KBDINPUTASSIST_PREV	0x260
KEY_KBDINPUTASSIST_NEXT	0x261
KEY_KBDINPUTASSIST_PREVGROU P	0x262
KEY_KBDINPUTASSIST_NEXTGRO UP	0x263
KEY_KBDINPUTASSIST_ACCEPT	0x264
KEY_KBDINPUTASSIST_CANCEL	0x265
KEY_RIGHT_UP	0x266
KEY_RIGHT_DOWN	0x267
KEY_LEFT_UP	0x268
KEY_LEFT_DOWN	0x269
KEY_ROOT_MENU	0x26a
KEY_MEDIA_TOP_MENU	0x26b
KEY_NUMERIC_11	0x26c
KEY_NUMERIC_12	0x26d

KEY_AUDIO_DESC	0x26e
KEY_3D_MODE	0x26f
KEY_NEXT_FAVORITE	0x270
KEY_STOP_RECORD	0x271
KEY_PAUSE_RECORD	0x272
KEY_VOD	0x273
KEY_UNMUTE	0x274
KEY_FASTREVERSE	0x275
KEY_SLOWREVERSE	0x276
KEY_DATA	0x277
KEY_ONSCREEN_KEYBOARD	0x278
SW_LID	0x00
SW_TABLET_MODE	0x01
SW_HEADPHONE_INSERT	0x02
SW_RFKILL_ALL	0x03
SW_RADIO	SW_RFKILL_ALL
SW_MICROPHONE_INSERT	0x04
SW_DOCK	0x05
SW_LINEOUT_INSERT	0x06
SW_JACK_PHYSICAL_INSERT	0x07
SW_VIDEOOUT_INSERT	0x08
SW_CAMERA_LENS_COVER	0x09
SW_KEYPAD_SLIDE	0x0a
SW_FRONT_PROXIMITY	0x0b
SW_ROTATE_LOCK	0x0c
SW_LINEIN_INSERT	0x0d
SW_MUTE_DEVICE	0x0e
SW_PEN_INSERTED	0x0f
SW_MAX	0x0f

## 附录B 龙芯ACPI设备ID

设备ID	说明
LOON0000	热键设备
LOON0001	芯片组RTC模块
LOON0002	7A1000芯片组集成的普通GPIO
LOON0003	SE模块
LOON0004	芯片组I2C控制器
LOON0005	GPIO模拟I2C
LOON0006	芯片组PWM控制器
LOON0007	3A5000处理器集成的GPIO
LOON0008	处理器温度传感器
LOON0009	芯片组CAN控制器（7A2000，2K2000）
LOON000A	2K2000处理器集成的GPIO
LOON000B	2K2000南北桥集成的GPIO
LOON000C	2K2000南北桥集成的ACPI GPIO
LOON000D	7A2000芯片组集成的普通GPIO
LOON000E	7A2000芯片组集成的ACPI GPIO
LOON000F	3A6000处理器集成的GPIO
LOON0010	内存ECC校验
LOON0011	龙芯SE设备
LOON0012	龙芯SE设备内部的SDF设备
LOON0013	芯片组DMA控制器（2K2000）
LOON0014	芯片组DMA控制器（2K3000）
LOON0015	芯片组CAN控制器（2K3000）

## 第二章 龙芯7A芯片组固件开发规范

### 1 设计约束

#### 1.1 Audio

##### 1.1.1 HDA

固件通过Verb Table机制实现不同板卡中对声卡的差异化设计。固件开发人员需要根据具体主板的声卡设计正确填写Verb Table。具体参考龙芯代码中关于Realtek的Verb Table配置实现。

注意：

须确认芯片组HDA引脚工作在HDA模式（芯片组配置寄存器偏移0x0440寄存器的12:11为x1b。芯片组配置寄存器基址需要固件下进行配置，参考代码配置寄存器基址为：0x0e0010010000）。

##### 1.1.2 I2S

对于7A2000，固件需要根据芯片手册配置hda\_i2s\_sel引脚为I2S工作模式（芯片组配置寄存器偏移0x0440寄存器的12:11为10b，配置寄存器基址需要固件下进行配置，参考代码配置寄存器基址为：0x0e0010010000）。

#### 1.2 VBIOS

7A芯片组内部包含图形处理器（以下简称GPU），位于D6:F0；内核下GPU需要Video BIOS（以下简称VBIOS）的支撑才可以工作正常。

固件对VBIOS处理方式：将固件内默认VBIOS全部读取到内置显卡BAR2空间的最后1MB。

#### 1.3 GMAC

7A1000包含两个GMAC控制器，固件分别从7A1000 SPI flash偏移0x0和0x10处读取MAC0和 MAC1地址信息，并判断是否合法，如果合法，并写入相应GMAC的MAC地址寄存器中；如果非法，将随机数写入相应GMAC的MAC地址寄存器中。

7A2000包含一个GMAC控制器，固件从7A2000 SPI flash偏移0x0处读取MAC0地址信息，并判断是否合法，如果合法，写入相应GMAC的MAC地址寄存器中；如果非法，将随机数写入相应GMAC的MAC地址寄存器中。

MAC地址寄存器定义见表1-1和1-2：

表1-1 GMAC0/GMAC1 Address High Register(Offset 0x40)

位域	名称	默认值	描述
31	MO: Always 1	0x0	保留
30:16	Reserved	0x0	保留
15:0	MAC Address0[47:32] MAC地址高16位	0x0	存放用于接收地址过滤和传输流控帧的MAC地址。

表1-2 GMAC0/GMAC1 Address Low Register(Offset 0x44)

位域	名称	默认值	描述
31:0	MAC Address0[31:0] MAC地址低32位	0x0	存放用于接收地址过滤和传输流控帧的MAC地址。

## 1.4 DMA地址映射

对于不支持超过44位DMA地址的设备，固件需要调整7A芯片组的DMA路由配置以支持此类设备。参考对应7A芯片组用户手册，DMA路由配置寄存器（0x041C）的[12:8]位，具体实现参考龙芯代码。

寄存器定义见表1-3：

表1-3 DMA路由配置寄存器[12:8]位

位域	名称	访问	描述
12:8	dma_node_id_offset	RW	DMA访问中节点号所在的地址偏移（相对于bit36）

## 1.5 中断

### 1.5.1 扩展IO中断控制器模型

扩展IO中断控制器模型，芯片组中断控制器管理的中断可通过中断线或HT路由至处理器，MSI中断通过HT路由至处理器，以上中断最终路由至处理器核。见图1-1所示。

龙芯5000系列处理器，3A6000处理器采用该中断模型。



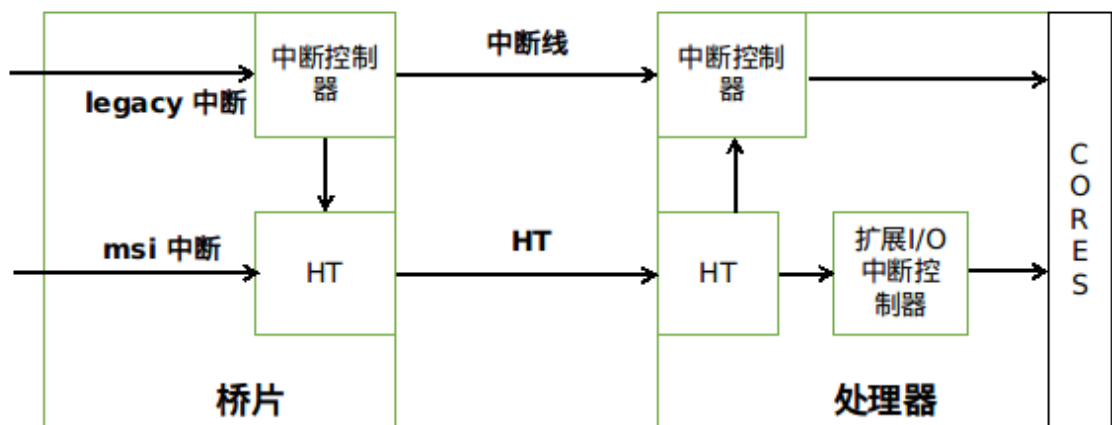


图1-1 扩展IO中断控制器模型中断路由示意图

7A芯片组系统依据ACPI规范使用GSI（global system interrupts，全局系统中断）为中断源分配虚拟化的中断号，一个GSI类型的虚拟中断号唯一标识了其所属的中断域，内核按照不同的中断域实现中断管理。各个中断域对应的GSI中断号范围约定如下：

0-15：LPC/ISA中断域，用于系统外扩LPC设备中断源，如表1-4所示，具体分配情况参考 LPC设备说明。

表1-4 LPC/ISA中断域GSI分配

GSI中断号	中断源	说明
0	HPET	HPET高精度定时器
1	I8042	键盘
2	级联	
3		
4		
5		
6		
7		
8	RTC	实时时钟
9		
10		
11		
12	I8042	鼠标

13		
14	ide0	硬盘
15	Idel	硬盘

16–47：处理器I/O中断控制器中断域，其GSI中断号与中断源向量号的映射关系： $GSI中断号 = 中断源向量号 + 16$  中断源向量号为处理器手册中某个中断源在中断控制寄存器中的索引，具体参考龙芯5000系列处理器手册。

48–63：保留。

64–127：芯片组I/O中断控制器中断域。其GSI中断号与中断源向量号的映射关系： $GSI中断号 = 中断源向量号 + GSI\ base$  中断源向量号为芯片组手册中某个中断源在中断控制寄存器中的索引，具体参考对应龙芯7A芯片组用户手册；GSI base为MADT表中当前芯片组BIO PIC结构的GSI base成员。其中单芯片组集成或外接的PCI兼容设备分配的中断号如表1-5、1-6：

表1-5 7A1000单芯片组I/O中断控制器中断域GSI分配

中断号	中断源	中断号	中断源
64		96	pcie_f0_0
65		97	pcie_f0_1
66		98	pcie_f0_2
67		99	pcie_f0_3
68		100	pcie_f1_0
69		101	pcie_f1_1
70		102	pcie_h_lo
71		103	pcie_h_hi
72	uart[3:0]	104	pcie_g0_hi
73	i2c[5:0]	105	pcie_g0_lo
74		106	pcie_g1_lo
75		107	pcie_g1_hi
76	gmac0_sbd	108	toy[0]
77	gmac0_pmt	109	toy[1]
78	gmac1_sbd	110	toy[2]
79	gmac1_pmt	111	acpi

80	sata[0]	112	usb_0_ehci
81	sata[1]	113	usb_0_ohci
82	sata[2]	114	usb_1_ehci
83	lpc	115	usb_1_ohci
84		116	rtc[0]
85		117	rtc[1]
86		118	rtc[2]
87		119	hpet_int
88	pwm[0]	120	ac97_dma[0]
89	pwm[1]	121	ac97_dma[1]
90	pwm[2]	122	ac97/hda
91	pwm[3]	123	gpio_hi
92	dc	124	gpio[0]
93	gpu	125	gpio[1]
94	gmem	126	gpio[2]
95	thsens	127	gpio[3]

表1-6 7A2000单芯片组I/O中断控制器中断域GSI分配

中断号	中断源	中断号	中断源
64		96	pcie_f0_0
65		97	pcie_f0_1
66		98	pcie_f0_2
67	rio	99	pcie_f0_3
68	prg	100	pcie_f1_0
69	sataphy	101	pcie_f1_1
70	usb3phy	102	pcie_h_lo
71	gmac0_phy	103	pcie_h_hi
72	uart[3:0]	104	pcie_g0_lo
73	i2c[5:0]	105	pcie_g0_hi
74	vpu	106	pcie_g1_lo

75	can	107	pcie_g1_hi
76	gmac0_sbd	108	toy[0]
77	gmac0_pmt	109	toy[1]
78	gmac1_sbd	110	toy[2]
79	gmac1_pmt	111	acpi
80	sata	112	usb_0_ehci
81		113	usb_0_ohci
82		114	usb_1_ehci
83	lpc	115	usb_1_ohci
84	hpet[1]	116	rtc[0]
85	hpet[2]	117	rtc[1]
86	usb3	118	rtc[2]
87	hda1	119	hpet[0]
88	pwm[0]	120	i2s_dma[0]
89	pwm[1]	121	i2s_dma[1]
90	pwm[2]	122	i2s/hda
91	pwm[3]	123	gpio_hi
92	dc	124	gpio[0]/gpio[13]
93	gpu	125	gpio[1]/gpio[14]
94	gmem	126	gpio[2]/gpio[15]
95	thsens	127	gpio[3]/gpio[50]

注意：

（ 1 ） 根据7A1000硬件规范，若使用EC时需要使用SCI中断，此中断需要硬件连接到7A1000 电源管理模块的LID管脚，信号约束参照硬件设计规范，对应GSI中断源为111号。

（ 2 ） 根据7A2000硬件规范，若使用EC时需要使用SCI中断，此中断需要硬件连接到7A2000 电源管理模块的ACPI GPIO管脚，信号约束参照硬件设计规范，对应GSI中断源为111号。

（ 3 ） 对于多芯片组系统,非PCI兼容设备的中断源仅在一个芯片组上可用。

## 1.5.2 高级向量中断控制器模型

高级向量中断控制器模型向前兼容扩展IO中断控制器模型，相关方案实现为：芯片组中断控

制器管理的中断可路由至处理器扩展IO中断控制器，芯片组MSI中断可路由至处理器高级向量中断控制器，处理器MSI中断可路由至处理器高级向量中断控制器，处理器非MSI中断可通过中断线路由至处理器传统IO中断控制器，以上中断最终路由至处理器核。见图1-2所示。

龙芯3C6000系列处理器采用该中断模型。

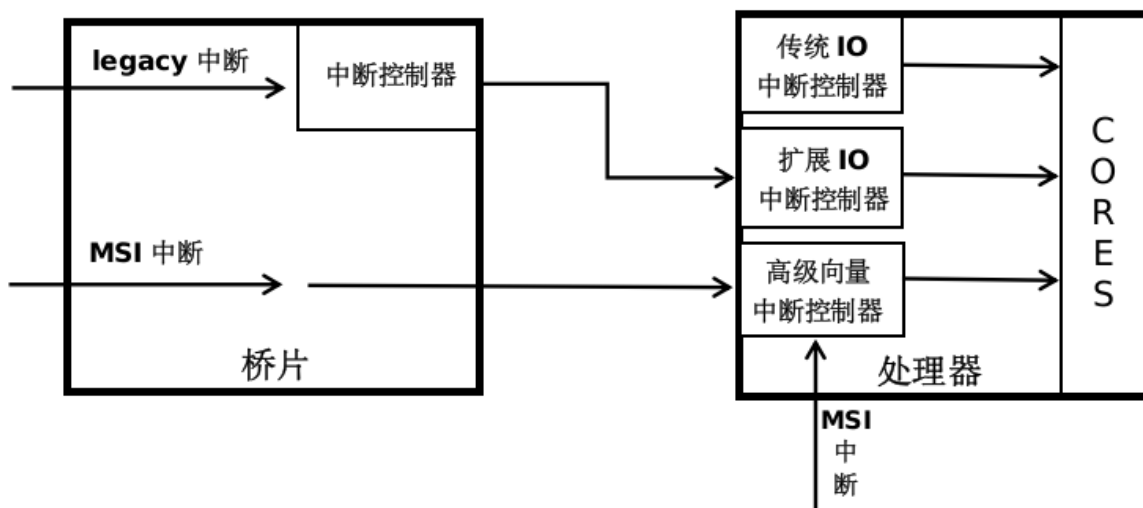


图1-2 高级向量中断控制器中断路由示意图

7A芯片组系统依据ACPI规范使用GSI (global system interrupts, 全局系统中断) 为中断源分配虚拟化的中断号，一个GSI类型的虚拟中断号唯一标识了其所属的中断域，内核按照不同的中断域实现中断管理。各个中断域对应的GSI中断号范围与1.5.1节约定一致。

## 1.6 地址空间

### 1.6.1 龙芯5000系列处理器+7A芯片组地址空间

龙芯5000系列处理器+7A芯片组地址空间如图1-3所示：

0xYF FF FFFF FFFF	Reserved
0xYF 00 0000 0000	PCI EXT CFG
0xYE FE 0000 0000	Reserved
0xYE FD 0000 0000	PCI EXT MEM
0xYE 80 0000 0000	7A HT1 Io
0xYE 00 0000 0000	Reserved
MEM_MAX_ADDR	System RAM
0xY0 00 9000 0000	System RAM/Reserved
0xY0 00 8000 0000	PCI MEM
0xY0 00 3000 0000	CPU Registers
0xY0 00 1C00 0000	PCI CFG
0xY0 00 1A00 0000	PCI I/O
0xY0 00 1800 0000	Chipset Registers
0xY0 00 1000 0000	System RAM/Reserved
0xY0 00 0000 0000	

图1-3 龙芯5000系列处理器+7A芯片组地址空间划分

地址中的Y 代表节点号（0-15），各个地址空间说明如下：

0xY0 00 0000 0000 – 0xY0 00 0FFF FFFF：对于0号节点，表示内存，对于非0号节点，表示保留空间

0xY0 00 1000 0000 – 0xY0 00 17FF FFFF：连接到处理器节点Y的芯片组寄存器

0xY0 00 1800 0000 – 0xY0 00 19FF FFFF：连接到处理器节点Y的PCI I/O空间

0xY0 00 1A00 0000 – 0xY0 00 1BFF FFFF：连接到处理器节点Y的PCI 配置空间

0xY0 00 1C00 0000 – 0xY0 00 1FFF FFFF：处理器节点Y的低速设备寄存器

0xY0 00 3000 0000 – 0xY0 00 7FFF FFFF：连接到处理器节点Y的32位PCI MEM空间

0xY0 00 8000 0000 – 0xY0 00 8FFF FFFF：对于非0号节点，表示内存，对于0号节点，表示保留空间

0xY0 00 9000 0000 – MEM\_MAX\_ADDR：内存，MEM\_MAX\_ADDR的取值见表1-7：

表1-7 MEM\_MAX\_ADDR的取值

系统中的节点数	MEM_MAX_ADDR
1	0xY0 80 0000 0000
2	0xY0 80 0000 0000
4	0xY0 40 0000 0000
8	0xY0 20 0000 0000
16	0xY0 10 0000 0000

0xY1 00 0000 0000 – 0xYD FF FFFF FFFF: 保留

0xYE 00 0000 0000 – 0xYE 00 7FFF FFFF: 7A HTI I/O 地址空间

0xYE 80 0000 0000 – 0xYE FC FFFF FFFF: 连接到处理器节点Y的扩展PCI MEM空间

0xYE FD 0000 0000 – 0xYE FD FFFF FFFF: 保留

0xYE FE 0000 0000 – 0xYE FF FFFF FFFF: 连接到处理器节点Y的扩展PCI 配置空间

0xYF 00 0000 0000 – 0xYF FF FFFF FFFF: 保留

以节点0地址为例，龙芯5000系列处理器+7A芯片组地址空间的详细分布如表1-8：

表1-8 龙芯5000系列处理器+7A芯片组地址空间分布

	起始地址	结束地址	说明
地址0	0x0000_0000_0000_0000	0x0000_0000_0FFF_FFFF	内存
地址1	0x0000_0000_1000_0000	0x0000_0000_17FF_FFFF	32位模式下7A 设备寄存器空间
地址2	0x0000_0000_1800_0000	0x0000_0000_1801_FFFF	32位模式下7A LPC的IO空间
地址3	0x0000_0000_1802_0000	0x0000_0000_19FF_FFFF	32位模式下7A PCI的IO空间
地址4	0x0000_0000_1A00_0000	0x0000_0000_1BFF_FFFF	32位模式下7A PCI的配置空间
地址5	0x0000_0000_1C00_0000	0x0000_0000_1DFF_FFFF	SPI Memory1
地址6	0x0000_0000_1FC0_0000	0x0000_0000_1FCF_FFFF	SPI Memory0
地址7	0x0000_0000_1FE0_01E0	0x0000_0000_1FE0_01E7	UART 0
地址8	0x0000_0000_1FE0_01E8	0x0000_0000_1FE0_01EF	UART 1
地址9	0x0000_0000_1FE0_01F0	0x0000_0000_1FE0_01FF	SPI Register
地址10	0x0000_0000_3000_0000	0x0000_0000_7FFF_FFFF	7A中PCI MEM空间
地址11	0x0000_0000_8000_0000	0x0000_0000_8FFF_FFFF	保留
地址12	0x0000_0000_9000_0000	MEM_MAX_ADDR	内存
地址13	0x0000_0E00_1000_0000	0x0000_0E00_1000_0FFF	7A中断控制器空间
地址14	0x0000_0E00_1000_1000	0x0000_0E00_1000_1FFF	7A中HPET寄存器空间
地址15	0x0000_0E00_1000_2000	0x0000_0E00_1000_2FFF	7A中LPC控制寄存器空间
地址16	0x0000_0E00_1001_0000	0x0000_0E00_1001_FFFF	7A中confbus空间
地址17	0x0000_0E00_1008_0000	0x0000_0E00_100F_FFFF	7A中misc设备寄存器空间
地址18	0x0000_0E00_1200_0000	0x0000_0E00_13FF_FFFF	7A中LPC MEM空间
地址19	0x0000_0E00_2000_0000	0x0000_0EFC_FFFF_FFFF	7A中PCI MEM空间
地址20	0x0000_0EFD_FC00_0000	0x0000_0EFD_FDFF_FFFF	HTI 的 I/O 空间

	起始地址	结束地址	说明
地址21	0x0000_0EFD_FE00_0000	0x0000_0EFD_FFFF_FFFF	HTI 的总线配置空间

## 2 参考配置

### 2.1 MADT配置

表2-1 MADT表约束

域	大小 (字节)	偏移 (字节)	描述
Header			
Signature	4	0	‘APIC’
Revision	1	8	1
Local Interrupt Controller Address	4	36	0x1fe01400

表2-2 CORE PIC结构

域	大小 (字节)	偏移 (字节)	描述
Type	1	0	0x11
Length	1	1	15
Version	1	2	1
ACPI Processor ID	4	3	处理器核UID，与DSDT处理器对象中的_UID值相同
Physical Processor ID	4	7	CPU核物理ID
Flags	4	11	CORE PIC的标志，参考表2-3

表2-3 CORE PIC标志

CORE PIC Flags	大小 (比特)	偏移 (比特)	描述
Enabled	1	0	0: CPU不可用 1: CPU可用
Reserved	31	1	必须为0



MADT表的LIO PIC结构参考如下表2-4:

表 2-4 LIO PIC结构

域	大小 (字节)	偏移 (字节)	描述
Type	1	0	0x12, LIO PIC结构
Length	1	1	23
Version	1	2	1
Base Address	8	3	0x1FE01400
Size	2	11	0x80
Cascade vector	2	13	0x0002
Cascade vector mapping	8	15	0x0000000000FFFFFF

MADT表的EIO PIC结构参考如下表2-5, 每个芯片组连接的处理器节点对应一个EIO PIC结构:

表 2-5 EIO PIC结构

域	大小 (字节)	偏移 (字节)	描述
Type	1	0	0x14, EIO PIC结构
Length	1	1	4
Version	1	2	1
Cascade vector	1	3	3+N, N为EIO PIC结构从0开始的序号
Node	1	4	连接芯片组的处理器节点ID
Node map	8	5	EIO中断路由的处理器节点组, bit0-63分别表示0-63节点

MADT表的MSI PIC结构参考如下表2-6, 每个芯片组对应一个相同的MSI PIC结构:

表 2-6 MSI PIC结构

域	大小 (字节)	偏移 (字节)	描述
Type	1	0	0x15, MSI PIC结构
Length	1	1	19
Version	1	2	1
Message Address	8	3	0x2FF00000

Start	4	11	0x40
Count	4	15	0xc0

MADT表的BIO PIC结构参考如下表2-7:

表 2-7 BIO PIC结构

域	大小 (字节)	偏移 (字节)	描述
Type	1	0	0x16, BIO PIC结构
Length	1	1	17
Version	1	2	1
Base Address	8	3	0xE0010000000
Size	2	11	0x1000
Hardware ID	2	13	0
GSI base	2	15	0x40

对于双桥片的主板（以桥0连接到0节点，桥1连接到N节点为例），需要再添加一组BIO PIC结构的描述，参考如下表2-8:

表 2-8 BIO PIC结构

域	大小 (字节)	偏移 (字节)	描述
Type	1	0	0x16, BIO PIC结构
Length	1	1	17
Version	1	2	1
Base Address	8	3	0xNE0010000000
Size	2	11	0x1000
Hardware ID	2	13	N
GSI base	2	15	0x80

MADT表的LPC PIC结构参考如下表2-9:

表 2-9 LPC PIC结构

域	大小 (字节)	偏移 (字节)	描述
---	------------	------------	----

Type	1	0	0x17, LPC PIC结构
Length	1	1	15
Version	1	2	1
Base Address	8	3	0xE0010002000
Size	2	11	0x1000
Cascade vector	2	13	0x13

## 2.2 FADT配置

此表为操作系统提供了Fixed硬件ACPI描述信息，相关约束见表2-10。

表2-10 FADT表约束

域	大小 (字节)	偏移 (字节)	描述
Header			
Signature	4	0	‘FACP’
Length	4	4	表的长度，包括表头，从偏移量0开始。用于记录整个表的大小
FADT Major Version	1	8	3
FIRMWARE_CTRL	4	36	0
DSDT	4	40	0
Reserved	1	44	0
SCL_INT	2	46	0x6f
SML_CMD	4	48	0
ACPI_ENABLE	1	52	0
ACPI_DISABLE	1	53	0
S4BIOS_REQ	1	54	0
PSTATE_CNT	1	55	0
PM1a_EVT_BLK	4	56	0
PM1b_EVT_BLK	4	60	0
PM1a_CNT_BLK	4	64	0
PM1b_CNT_BLK	4	68	0

PM2_CNT_BLK	4	72	0
PM_TMR_BLK	4	76	0
GPE0_BLK	4	80	0
GPE1_BLK	4	84	0
PM1_EVT_LEN	1	88	8
PM1_CNT_LEN	1	89	4
PM2_CNT_LEN	1	90	0
PM_TMR_LEN	1	91	4
GPE0_BLK_LEN	1	92	8
GPE1_BLK_LEN	1	93	0
GPE1_BASE	1	94	0
CST_CNT	1	95	0
P_LVL2_LAT	2	96	0x65
P_LVL3_LAT	2	98	0x3e9
FLUSH_SIZE	2	100	0
FLUSH_STRIDE	2	102	0
DUTY_OFFSET	1	104	0
DUTY_WIDTH	1	105	0
DAY_ALARM	1	106	0
MON_ALARM	1	107	0
CENTURY	1	108	0
IAPC_BOOT_ARCH	2	109	0
Reserved	1	111	0
Flags	4	112	默认值为0x425，包含如下标志： WBINVD、PROC_CI、SLP_BUTTON、RESET_REG_SUP
RESET_REG	12	116	见表2-11
RESET_VALUE	1	128	0x01
ARM_BOOT_ARCH	2	129	0
FADT Minor Version	1	131	0
X_FIRMWARE_CTRL	8	132	FACS 64bit地址，固件动态生成

X_DSDT	8	140	DSDT 64bit地址，固件动态生成
X_PM1a_EVT_BLK	12	148	见表2-12
X_PM1b_EVT_BLK	12	160	此结构中成员全为0
X_PM1a_CNT_BLK	12	172	见表2-13
X_PM1b_CNT_BLK	12	184	此结构中成员全为0
X_PM2_CNT_BLK	12	196	此结构中成员全为0
X_PM_TMR_BLK	12	208	见表2-14
X_GPE0_BLK	12	220	见表2-15
X_GPE1_BLK	12	232	此结构中成员全为0

注：UEFI固件动态生成X\_FIRMWARE\_CTRL与X\_DSDT之前，需将其初始化为0。

表 2-11 reset寄存器结构

域	大小 (字节)	偏移 (字节)	描述
Address Space Id	1	0	0
Reg Bit Width	1	1	0x20
Reg Bit Offset	1	2	0
Reserved	1	3	0
Address	8	4	0x00000e00100d0030

表 2-12 PM1a EVT寄存器结构

域	大小 (字节)	偏移 (字节)	描述
Address Space Id	1	0	0,
Reg Bit Width	1	1	0x40
Reg Bit Offset	1	2	0
Reserved	1	3	0
Address	8	4	0x00000e00100d000c

表 2-13 PM1a\_CNT寄存器结构

域	大小 (字节)	偏移 (字节)	描述
Address Space Id	1	0	0,
Reg Bit Width	1	1	0x20
Reg Bit Offset	1	2	0
Reserved	1	3	0
Address	8	4	0x00000e00100d0014

表 2-14 PM\_TMR寄存器结构

域	大小 (字节)	偏移 (字节)	描述
Address Space Id	1	0	0,
Reg Bit Width	1	1	0x20
Reg Bit Offset	1	2	0
Reserved	1	3	0
Address	8	4	0x00000e00100d0018

表 2-15 GPE0寄存器结构

域	大小 (字节)	偏移 (字节)	描述
Address Space Id	1	0	0,
Reg Bit Width	1	1	0x40
Reg Bit Offset	1	2	0
Reserved	1	3	0
Address	8	4	0x00000e00100d0028

## 2.3 DSDT配置

### 2.3.1 PCI总线枚举

\_SEG (Segment) 方法描述了PCI段号。

\_CRS (Current Resource Settings) 方法支持总线范围、IO地址范围以及内存地址范围声明。

\_OSC (Operating System Capabilities) 方法，当配置了\_OSC时，仅支持AER、PME和PCI Express

Capability Structure control。

(1) PCIE单主桥配置参考如下：

\_SEG为0

\_CRS

使用WordBusNumber()宏声明PCI总线号范围，如表2-16：

表2-16 PCI总线号范围

参数名	值
ResourceUsage	ResourceProducer
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Decode	PosDecode
AddressGranularity	0x0
AddressMinimum	0x0
AddressMaximum	0x00FF
AddressTranslation	0x0000
RangeLength	0x0100
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空

使用QWordIO()宏声明PCI I/O地址范围，如表2-17：

表2-17 PCI I/O地址范围

参数名	值
ResourceUsage	ResourceProducer
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Decode	PosDecode
ISARanges	EntireRange
AddressGranularity	0x00000000000010000
AddressMinimum	0x0000000000000000

AddressMaximum	0x000000000000FFFF
AddressTranslation	0x0000000018000000
RangeLength	0x0000000000010000
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
TranslationType	值为空
TranslationDensity	值为空

使用QWordMemory()宏声明32位PCI MEM地址范围如表2-18:

表2-18 32位PCI MEM地址范围

参数名	值
ResourceUsage	ResourceProducer
Decode	PosDecode
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Cacheable	Cacheable
ReadAndWrite	ReadWrite
AddressGranularity	0x0000000000010000
AddressMinimum	0x0000000030000000
AddressMaximum	0x000000007FFFFFFF
AddressTranslation	0x00000E0000000000
RangeLength	0x0000000050000000
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
MemoryRangeType	值为空
TranslationType	值为空

使用QWordMemory()宏声明64位PCI MEM地址范围如表2-19:

表2-19 64位PCI MEM地址范围



参数名	值
ResourceUsage	ResourceProducer
Decode	PosDecode
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Cacheable	Cacheable
ReadAndWrite	ReadWrite
AddressGranularity	0x00000000000010000
AddressMinimum	0x0000008000000000
AddressMaximum	0x000000FCFFFFFFF
AddressTranslation	0x00000E0000000000
RangeLength	0x0000007D00000000
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
MemoryRangeType	值为空
TranslationType	值为空

(2) PCIE双主桥配置参考如下:

主桥一的配置:

\_SEG为0

\_CRS

使用WordBusNumber()宏声明PCI总线号范围, 如表2-20:

表2-20 PCI总线号范围

参数名	值
ResourceUsage	ResourceProducer
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Decode	PosDecode
AddressGranularity	0x0

AddressMinimum	0x0
AddressMaximum	0x00FF
AddressTranslation	0x0000
RangeLength	0x0100
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空

使用QWordIO()宏声明PCI I/O地址范围，如表2-21：

表2-21 PCI I/O地址范围

参数名	值
ResourceUsage	ResourceProducer
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Decode	PosDecode
ISARanges	EntireRange
AddressGranularity	0x00000000000010000
AddressMinimum	0x0000000000000000
AddressMaximum	0x000000000000FFFF
AddressTranslation	0x0000000018000000
RangeLength	0x00000000000010000
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
TranslationType	值为空
TranslationDensity	值为空

使用QWordMemory()宏声明32位PCI MEM地址范围，如表2-22：

表2-22 32位PCI MEM地址范围

参数名	值
-----	---

ResourceUsage	ResourceProducer
Decode	PosDecode
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Cacheable	Cacheable
ReadAndWrite	ReadWrite
AddressGranularity	0x00000000000010000
AddressMinimum	0x0000000030000000
AddressMaximum	0x000000007FFFFFFF
AddressTranslation	0x00000E0000000000
RangeLength	0x0000000050000000
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
MemoryRangeType	值为空
TranslationType	值为空

使用QWordMemory()宏声明64位PCI MEM地址范围，如表2-23:

表2-23 64位PCI MEM地址范围

参数名	值
ResourceUsage	ResourceProducer
Decode	PosDecode
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Cacheable	Cacheable
ReadAndWrite	ReadWrite
AddressGranularity	0x00010000
AddressMinimum	0x0000008000000000
AddressMaximum	0x000000FCFFFFFFF
AddressTranslation	0x00000E0000000000

RangeLength	0x0000007D00000000
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
MemoryRangeType	值为空
TranslationType	值为空

主桥二的配置：

\_SEG为1

\_CRS

使用WordBusNumber()宏声明PCI总线号范围，如表2-24：

表2-24 PCI总线号范围

参数名	值
ResourceUsage	ResourceProducer
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Decode	PosDecode
AddressGranularity	0x0
AddressMinimum	0x0
AddressMaximum	0x00FF
AddressTranslation	0x0000
RangeLength	0x0100
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空

使用QWordIO()宏声明PCI I/O地址范围，如表2-25：

表2-25 PCI I/O地址范围

参数名	值
ResourceUsage	ResourceProducer

IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Decode	PosDecode
ISARanges	EntireRange
AddressGranularity	0x00000000000010000
AddressMinimum	0x0000000000000000
AddressMaximum	0x000000000000FFFF
AddressTranslation	0x00005EFDFC000000
RangeLength	0x00000000000010000
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
TranslationType	值为空
TranslationDensity	值为空

使用QWordMemory()宏声明32位PCI MEM地址范围，如表2-26：

表2-26 32位PCI MEM地址范围

参数名	值
ResourceUsage	ResourceProducer
Decode	PosDecode
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Cacheable	Cacheable
ReadAndWrite	ReadWrite
AddressGranularity	0x00000000000010000
AddressMinimum	0x0000000030000000
AddressMaximum	0x000000007FFFFFFF
AddressTranslation	0x0000YE0000000000，Y代表当前PCI主桥连接的处理器节点的节点号
RangeLength	0x0000000050000000
ResourceSourceIndex	值为空

参数名	值
ResourceSource	值为空
DescriptorName	值为空
MemoryRangeType	值为空
TranslationType	值为空

使用QWordMemory()宏声明64位PCI MEM地址范围，如表2-27：

表2-27 64位PCI MEM地址范围

参数名	值
ResourceUsage	ResourceProducer
Decode	PosDecode
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Cacheable	Cacheable
ReadAndWrite	ReadWrite
AddressGranularity	0x00000000000010000
AddressMinimum	0x0000008000000000
AddressMaximum	0x000000FCFFFFFFF
AddressTranslation	0x0000YE0000000000，Y代表当前PCI主桥连接的处理器节点的节点号
RangeLength	0x0000007D00000000
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
MemoryRangeType	值为空
TranslationType	值为空

## 2.3.2 设备电源管理

7A芯片组系统支持USB2.0和GMAC唤醒系统，需要配置相应的\_PRW。

### 2.3.2.1 USB

\_PRW (Power Resources for Wake)

表2-28 package对象参数表

参数名	值
EventInfo	设备号为D4:F1的USB控制器：值为0xa 设备号为D5:F1的USB控制器：值为0xd
DeepestSleepState	0x3

### 2.3.2.2 GMAC

\_PRW (Power Resources for Wake)

表2-29 package对象参数表

参数名	值
EventInfo	设备号为D3:F0的GMAC控制器：值为0x5 设备号为D3:F1的GMAC控制器：值为0x6
DeepestSleepState	0x3

## 2.3.3 系统休眠唤醒

表2-30 SLP\_TYP规定

芯片组	休眠状态	SLP_TYP
7A1000、 7A2000	S0	0
	S3	5
	S4	6
	S5	7

### 2.3.4 串口配置

\_CRS

表2-31串口参数表

参数名	值
ResourceUsage	ResourceConsumer
Decode	PosDecode
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Cacheable	NonCacheable
ReadAndWrite	ReadWrite
AddressGranularity	0x0
AddressMinimum	处理器串口0值为: 0x000000001FE001E0 芯片组串口0值为: 0x00000E0010080000
AddressMaximum	处理器串口0值为: 0x000000001FE001E7 芯片组串口0值为: 0x00000E00100800FF
AddressTranslation	0x0
RangeLength	处理器串口0值为: 0x0000000000000008 芯片组串口0值为: 0x0000000000000100
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
MemoryRangeType	AddressRangeMemory
TranslationType	TypeStatic

中断号资源使用Interrupt()宏声明，各个参数的规定如表2-32：

表2-32 Interrupt()宏参数表

参数名	值
ResourceUsage	ResourceConsumer
EdgeLevel	Level
ActiveLevel	ActiveHigh
Shared	Shared
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空



InterruptList	处理器串口0: 26 芯片组串口: 72
---------------	----------------------

\_DSD

表2-33 支持属性表

属性名	值	说明
clock-frequency	处理器串口值为: 100000000 芯片组串口值为: 50000000	表示实际外接的串口时钟频率, 单位为Hz

### 2.3.5 GPIO配置

\_HID

(1) 7A1000芯片组集成的普通GPIO为LOON0002, 3A5000处理器集成的GPIO为 LOON0007, 7A2000芯片组集成的普通GPIO为LOON000D, 7A2000芯片组集成的ACPI GPIO为LOON000E, 3A6000处理器集成的GPIO为LOON000F。

\_CRS

表2-34 QWordMemory()宏参数表

参数名	值
ResourceUsage	ResourceConsumer
Decode	PosDecode
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Cacheable	NonCacheable
ReadAndWrite	ReadWrite
AddressGranularity	0x0
AddressMinimum	7A芯片组集成的普通GPIO值为: 0x00000E00100E0000 7A芯片组集成的ACPI GPIO值为: 0x00000E00100D0080 3A处理器集成的GPIO值为: 0x000000001FE00500
AddressMaximum	7A芯片组集成的普通GPIO值为: 0x00000E00100E0BFF 7A芯片组集成的ACPI GPIO值为: 0x00000E00100D009F 3A处理器集成的GPIO值为: 0x000000001FE00517
AddressTranslation	0x0

RangeLength	7A芯片组集成的普通GPIO值为：0x0000000000000C00 7A芯片组集成的ACPI GPIO值为：0x0000000000000020 3A处理器集成的GPIO值为：0x0000000000000018
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
MemoryRangeType	AddressRangeMemory
TranslationType	TypeStatic

每个中断号资源使用一个Interrupt()宏声明，各个参数的规定如表2-35：

表2-35 Interrupt()宏参数表

参数名	值
ResourceUsage	ResourceConsumer
EdgeLevel	Level
ActiveLevel	ActiveHigh
Shared	Shared
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
InterruptList	7A芯片组集成的普通GPIO：123、124、125、126、127 7A芯片组集成的ACPI GPIO：111 3A处理器GPIO：16、17、18、19、20、21、22、23

\_DSD

表2-36 支持属性如表

属性名	值	说明
ngpios	7A1000芯片组集成的普通GPIO值为：57 7A2000芯片组集成的普通GPIO值为：58 7A2000芯片组集成的ACPI GPIO值为：16 3A处理器集成的GPIO值为：32	当前注册的gpio设备包含的gpio pin总数

### 2.3.6 I2C配置

\_HID

(1) LOON0004。

\_UID

各I2C控制器与\_UID对应关系规定如表2-37:

表2-37 I2C控制器与\_UID对应表

控制器	_UID
I2C控制器0	0
I2C控制器1	1
I2C控制器2	2
I2C控制器3	3
I2C控制器4	4
I2C控制器5	5

\_CRS

表2-38 QWordMemory()宏参数表

参数名	值
ResourceUsage	ResourceConsumer
Decode	PosDecode
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Cacheable	NonCacheable
ReadAndWrite	ReadWrite
AddressGranularity	0x0

AddressMinimum	I2C控制器值为: I2C 0: 0x00000E0010090000 I2C 1: 0x00000E0010090100 I2C 2: 0x00000E0010090200 I2C 3: 0x00000E0010090300 I2C 4: 0x00000E0010090400 I2C 5: 0x00000E0010090500
AddressMaximum	I2C控制器值为: I2C 0: 0x00000E0010090007 I2C 1: 0x00000E0010090107 I2C 2: 0x00000E0010090207 I2C 3: 0x00000E0010090307 I2C 4: 0x00000E0010090407 I2C 5: 0x00000E0010090507
AddressTranslation	0x0
RangeLength	I2C控制器值为: I2C 0: 0x0000000000000008 I2C 1: 0x0000000000000008 I2C 2: 0x0000000000000008 I2C 3: 0x0000000000000008 I2C 4: 0x0000000000000008 I2C 5: 0x0000000000000008
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
MemoryRangeType	AddressRangeMemory
TranslationType	TypeStatic

2.3.7 RTC配置

\_HID

LOON0001。

\_CRS

表2-39 QWordMemory()宏参数表

参数名	值
-----	---

ResourceUsage	ResourceConsumer
Decode	PosDecode
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Cacheable	NonCacheable
ReadAndWrite	ReadWrite
AddressGranularity	0x0
AddressMinimum	0x00000E00100d0100
AddressMaximum	0x00000E00100d01FF
AddressTranslation	0x0
RangeLength	0x0000000000000100
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
MemoryRangeType	AddressRangeMemory
TranslationType	TypeStatic

中断号资源使用Interrupt()宏声明，各个参数的规定如表2-40：

表2-40 Interrupt()宏参数表

参数名	值
ResourceUsage	ResourceConsumer
EdgeLevel	Level
ActiveLevel	ActiveHigh
Shared	Exclusive
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
InterruptList	116

## 2.3.8 PWM配置

\_HID

(1) LOON0006。

\_CRS

表2-41 QWordMemory()宏参数表

参数名	值
ResourceUsage	ResourceConsumer
Decode	PosDecode
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Cacheable	NonCacheable
ReadAndWrite	ReadWrite
AddressGranularity	0x0
AddressMinimum	PWM 0: 0x00000E00100A0000 PWM 1: 0x00000E00100A0100 PWM 2: 0x00000E00100A0200 PWM 3: 0x00000E00100A0300
AddressMaximum	PWM 0: 0x00000E00100A000F PWM 1: 0x00000E00100A010F PWM 2: 0x00000E00100A020F PWM 3: 0x00000E00100A030F
AddressTranslation	0x0
RangeLength	PWM 0: 0x0000000000000010 PWM 1: 0x0000000000000010 PWM 2: 0x0000000000000010 PWM 3: 0x0000000000000010
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
MemoryRangeType	AddressRangeMemory
TranslationType	TypeStatic

中断号资源使用Interrupt()宏声明，各个参数的规定如表2-42：

表2-42 Interrupt()宏参数表

参数名	值
ResourceUsage	ResourceConsumer
EdgeLevel	Level
ActiveLevel	ActiveHigh
Shared	Shared
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
InterruptList	PWM 0: 88 PWM 1: 89 PWM 2: 90 PWM 3: 91

### 2.3.9 热区管理

当配置龙芯5000系列处理器温度传感器时，可参考如下部分示例：

```
Scope (\_SB)
{
    Name (IDDR,0x1fe00000) //ChipReg

    OperationRegion (BASE, SystemMemory, IDDR, 0x1a4)
    Field (BASE, AnyAcc, NoLock, Preserve)
    {
        Offset (0x19C), //thsens
        THSE, 32
    }
    .....
}

Scope (\_TZ)
{
    ThermalZone (THM0)
    {
        Method (_TMP, 0, NotSerialized) // _TMP: Temperature
```

```

    {
        Store (\_SB.THSE, Local0)
        Store (CCNT (Local0), Local1)
        Return (C2K (Local1))
    }
.....
}

Method (CCNT, 1, NotSerialized)    //calculate cpu node temp, (reg & 0xffff) * 731 /
0x4000 - 273;
{
    Multiply (And (Arg0, 0xFFFF), 0x02DB, Local0)
    Divide (Local0, 0x4000, , Local1)
    Subtract (Local1, 273, Local2)

    Return (Local2)
}

Method (C2K, 1, NotSerialized)    //Celsius to Kelvin
{
    Add (Multiply (Arg0, 0x0A), 0x0AAC, Local0)
    If (LLessEqual (Local0, 0x0AAC))
    {
        Store (0x0BB8, Local0)
    }

    If (LGreater (Local0, 0x0FAC))
    {
        Store (0x0BB8, Local0)
    }

    Return (Local0)
}
}

```

当配置7A2000温度传感器时，可以参考如下部分示例：

```

Scope (\_SB)
{
    Device (PCI0)
    {
        Name (BTSA,0xe0010000400) //bridge temp sensor addr
    }
}

```



```

    OperationRegion (BASE, SystemMemory, BTSA, 0x18)
    Field (BASE, AnyAcc, NoLock, Preserve)
    {
        Offset (0x14), //thsens
        THS1, 32
    }
    .....
}

Scope (\_TZ)
{
    .....
    Method (CBST, 1, NotSerialized)    //calculate bridge sensor temp, (reg & 0xffff0000) >>
24;
    {
        ShiftRight (And (Arg0, 0xFFFFF0000), 24, Local0)
        Return (Local0)
    }

    ThermalZone (THM1)
    {
        Name(_TZP, 300)    //polling delay

        Method (_TMP, 0, NotSerialized)    // _TMP: Temperature
        {
            Store (\_SB.PCI0.THS1, Local0)
            Store (CBST (Local0), Local1)
            Return (C2K (Local1))
        }

        Method (_CRT, 0, NotSerialized)    // _CRT: Critical Temperature
        {
            Return (C2K (0x60))
        }
    }
    .....
}

```

### 2.3.10 SE设备

\_HID

(1) LOON0003。

\_CRS

中断号资源使用Interrupt()宏声明，各个参数的规定如表2-43：

表2-43 Interrupt()宏参数表

参数名	值
ResourceUsage	ResourceConsumer
EdgeLevel	Level
ActiveLevel	ActiveHigh
Shared	Exclusive
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
InterruptList	33、36

### 2.3.11 EC配置

\_GPE

- (1) 对于7A1000芯片组，SCI中断对应的ACPI LID管脚位
- (2) 对于7A2000芯片组，SCI中断对应的ACPI GPIO管脚位

### 2.3.12 SPI配置

当配置7A芯片组SPI控制器时，可以参考如下部分示例：

```
Device (SPI0)
{
    Name (_ADR, 0x00160000) // _ADR: Address
    Name (_DSD, Package (0x02) // _DSD: Device-Specific Data
    {
        ToUUID ("daffd814-6eba-4d8c-8a91-bc9bbf4aa301") /* Device Properties for _DSD */,
```

```
Package (0x01)
{
    Package (0x02)
    {
        "clock-frequency",
        100000000
    }
}
})
}
```

## 2.4 FACS配置

表2-44 FACS表约束

域	大小 (字节)	偏移 (字节)	描述
Signature	4	0	‘FACS’
Firmware Waking Vector	4	12	0
Global Lock	4	16	0
Flags	4	20	0
X Firmware Waking Vector	8	24	0
Version	1	32	1

## 2.5 S3休眠地址

SADR，是系统S3休眠时进入固件执行休眠的入口地址，该地址由固件决定，存放固件的S3处理程序。

## 2.6 SRAT配置

表2-45 SRAT表约束

域	大小 (字节)	偏移 (字节)	描述
Header			
Signature	4	0	‘SRAT’

Revision	1	8	2
----------	---	---	---

表2-46 Processor Local APIC/SAPIC Affinity结构

域	大小 (字节)	偏移 (字节)	描述
Type	1	0	0, 代表结构为Processor Local APIC/SAPIC Affinity Structure。
Length	1	1	16
Proximity Domain [7:0]	1	2	处理器节点号的0-7位
APIC ID	1	3	处理器Local APIC ID, 见MADT表
Flags	4	4	标志。  0位: 0表示该Processor Local APIC/SAPIC Affinity Structure不可用; 1表示该Processor Local APIC/SAPIC Affinity Structure可用。  [31:1]位: 必须为0
Local SAPIC EID	1	8	0
Proximity Domain [31:8]	3	9	处理器节点号的8-31位
Clock Domain	4	12	0

表2-47 Memory Affinity结构

域	大小 (字节)	偏移 (字节)	描述
Type	1	0	1, 代表结构为 Memory Affinity Structure
Length	1	1	40
Proximity Domain	4	2	内存范围所属的节点号
Reserved	2	6	保留
Base Address Low	4	8	内存范围地址的低32位
Base Address High	4	12	内存范围地址的高32位
Length Low	4	16	内存范围大小的低32位
Length High	4	20	内存范围大小的高32位

Reserved	4	24	保留
Flags	4	28	标志，指示内存区域是否已启用并可以热插拔，见表2-48
Reserved	8	32	保留

表 2-48内存热插拔标志

域	大小 (比特)	偏移 (比特)	描述
Enabled	1	0	0: 该Memory Affinity Structure不可用 1: 该Memory Affinity Structure可用
Hot Pluggable	1	1	0
NonVolatile	1	2	0
Reserved	29	3	0

## 2.7 MCFG配置

如为双桥服务器设备（以桥0连接到0节点，桥1连接到5节点为例），MCFG表的配置如表2-49、2-50、2-51所示。

表2-49 MCFG表约束

域	大小 (字节)	偏移 (字节)	描述
Header			
Signature	4	0	‘MCFG’
Revision	1	8	1
OEMID	6	10	‘LOONGS’
OEM Table ID	8	16	‘LOONGSON’
OEM Revision	4	24	1

表2-50 Memory Mapped Enhanced Configuration Space Base Address Allocation  
Structure[0]结构

域	大小 (字节)	偏移 (字节)	描述
Base Address	8	0	0x00000efe00000000

PCI Segment Group Number	2	8	0
Start Bus Number	1	10	0x00
End Bus Number	1	11	0xFF
Reserved	4	12	0

表2-51 Memory Mapped Enhanced Configuration Space Base Address Allocation Structure[1]结构

域	大小 (字节)	偏移 (字节)	描述
Base Address	8	0	0x00005efe00000000
PCI Segment Group Number	2	8	1
Start Bus Number	1	10	0x00
End Bus Number	1	11	0xFF
Reserved	4	12	0

## 2.8 SLIT配置

表2-52 SLIT表约束

域	大小 (字节)	偏移 (字节)	描述
Header			
Signature	4	0	‘SLIT’
Number of System Localities	8	36	节点数。 单路：0x1； 双路：0x2； 四路：0x4； 八路：0x8； 十六路：0x10
Entry[Number of System Localities- 1][Number of System Localities-1]	1		根据龙芯参考代码实现

## 2.9 SPCR配置

表2-53 SPCR表约束

域	大小 (字节)	偏移 (字节)	描述
Signature	4	0	‘SPCR’
Interface Type	1	36	0
Reserved	3	37	0
Space ID	1	40	0
Bit Width	1	41	0
Bit Offset	1	42	0
Encoded Access Width	1	43	1
Address	8	44	0x000000001FE001E0
Interrupt Type	1	52	0
PCAT-compatible IRQ	1	53	0
Interrupt	4	54	0
Baud Rate	1	58	0
Parity	1	59	0
Stop Bits	1	60	0
Flow Control	1	61	0
Terminal Type	1	62	0
Language	1	63	0
PCI Device ID	2	64	0xFFFF
PCI Vendor ID	2	66	0xFFFF
PCI Bus	1	68	0
PCI Device	1	69	0
PCI Function	1	70	0
PCI Flags	4	71	0
PCI Segment	1	75	0
Reserved	4	76	0

## 2.10 PPTT配置

表2-54 PPTT表约束

域	大小 (字节)	偏移 (字节)	描述
Header			
Signature	4	0	‘PPTT’
Revision	1	8	3

表2-55 Processor hierarchy node structure约束

域	大小 (字节)	偏移 (字节)	描述
Type	1	0	0
Length	1	1	14
Reserved	2	2	0
Flags	4	4	处理器结构标志
Parent	4	8	当有父处理器节点结构时，值为PPTT表的开始和父处理器结构项的开始之间的差值； 当没有父处理器节点结构时，值为0
ACPI Processor ID	4	12	MADT表中的处理器核 UID
Number of private resources	4	16	0

以3A6000+7A2000为例，Processor hierarchy node structure结构约束如下：

```

[024h 0036 001h]          Subtable Type : 00 [Processor Hierarchy Node]
[025h 0037 001h]          Length : 14
[026h 0038 002h]          Reserved : 0000
[028h 0040 004h]          Flags (decoded below) : 00000011
                        Physical package : 1
                        ACPI Processor ID valid : 0
                        Processor is a thread : 0
                        Node is a leaf : 0
                        Identical Implementation : 1
[02Ch 0044 004h]          Parent : 00000000
[030h 0048 004h]          ACPI Processor ID : 00000000

```



```

[034h 0052 004h] Private Resource Number : 00000000

[038h 0056 001h] Subtable Type : 00 [Processor Hierarchy Node]
[039h 0057 001h] Length : 14
[03Ah 0058 002h] Reserved : 0000
[03Ch 0060 004h] Flags (decoded below) : 00000012
                  Physical package : 0
                  ACPI Processor ID valid : 1
                  Processor is a thread : 0
                  Node is a leaf : 0
                  Identical Implementation : 1

[040h 0064 004h] Parent : 00000024
[044h 0068 004h] ACPI Processor ID : 00000000
[048h 0072 004h] Private Resource Number : 00000000

[04Ch 0076 001h] Subtable Type : 00 [Processor Hierarchy Node]
[04Dh 0077 001h] Length : 14
[04Eh 0078 002h] Reserved : 0000
[050h 0080 004h] Flags (decoded below) : 0000001E
                  Physical package : 0
                  ACPI Processor ID valid : 1
                  Processor is a thread : 1
                  Node is a leaf : 1
                  Identical Implementation : 1

[054h 0084 004h] Parent : 00000038
[058h 0088 004h] ACPI Processor ID : 00000001
[05Ch 0092 004h] Private Resource Number : 00000000

[060h 0096 001h] Subtable Type : 00 [Processor Hierarchy Node]
[061h 0097 001h] Length : 14
[062h 0098 002h] Reserved : 0000
[064h 0100 004h] Flags (decoded below) : 0000001E
                  Physical package : 0
                  ACPI Processor ID valid : 1
                  Processor is a thread : 1
                  Node is a leaf : 1
                  Identical Implementation : 1

[068h 0104 004h] Parent : 00000038
[06Ch 0108 004h] ACPI Processor ID : 00000002
[070h 0112 004h] Private Resource Number : 00000000

[074h 0116 001h] Subtable Type : 00 [Processor Hierarchy Node]
[075h 0117 001h] Length : 14

```

```

[076h 0118 002h]                               Reserved : 0000
[078h 0120 004h]       Flags (decoded below) : 00000012
                               Physical package : 0
                               ACPI Processor ID valid : 1
                               Processor is a thread : 0
                               Node is a leaf : 0
                               Identical Implementation : 1
[07Ch 0124 004h]                               Parent : 00000024
[080h 0128 004h]                               ACPI Processor ID : 00000001
[084h 0132 004h]       Private Resource Number : 00000000

[088h 0136 001h]                               Subtable Type : 00 [Processor Hierarchy Node]
[089h 0137 001h]                               Length : 14
[08Ah 0138 002h]                               Reserved : 0000
[08Ch 0140 004h]       Flags (decoded below) : 0000001E
                               Physical package : 0
                               ACPI Processor ID valid : 1
                               Processor is a thread : 1
                               Node is a leaf : 1
                               Identical Implementation : 1
[090h 0144 004h]                               Parent : 00000074
[094h 0148 004h]                               ACPI Processor ID : 00000003
[098h 0152 004h]       Private Resource Number : 00000000

[09Ch 0156 001h]                               Subtable Type : 00 [Processor Hierarchy Node]
[09Dh 0157 001h]                               Length : 14
[09Eh 0158 002h]                               Reserved : 0000
[0A0h 0160 004h]       Flags (decoded below) : 0000001E
                               Physical package : 0
                               ACPI Processor ID valid : 1
                               Processor is a thread : 1
                               Node is a leaf : 1
                               Identical Implementation : 1
[0A4h 0164 004h]                               Parent : 00000074
[0A8h 0168 004h]                               ACPI Processor ID : 00000004
[0ACh 0172 004h]       Private Resource Number : 00000000

[0B0h 0176 001h]                               Subtable Type : 00 [Processor Hierarchy Node]
[0B1h 0177 001h]                               Length : 14
[0B2h 0178 002h]                               Reserved : 0000
[0B4h 0180 004h]       Flags (decoded below) : 00000012
                               Physical package : 0
                               ACPI Processor ID valid : 1

```

```

Processor is a thread : 0
Node is a leaf : 0
Identical Implementation : 1
[0B8h 0184 004h] Parent : 00000024
[0BCh 0188 004h] ACPI Processor ID : 00000002
[0C0h 0192 004h] Private Resource Number : 00000000

[0C4h 0196 001h] Subtable Type : 00 [Processor Hierarchy Node]
[0C5h 0197 001h] Length : 14
[0C6h 0198 002h] Reserved : 0000
[0C8h 0200 004h] Flags (decoded below) : 0000001E
Physical package : 0
ACPI Processor ID valid : 1
Processor is a thread : 1
Node is a leaf : 1
Identical Implementation : 1
[0CCh 0204 004h] Parent : 000000B0
[0D0h 0208 004h] ACPI Processor ID : 00000005
[0D4h 0212 004h] Private Resource Number : 00000000

[0D8h 0216 001h] Subtable Type : 00 [Processor Hierarchy Node]
[0D9h 0217 001h] Length : 14
[0DAh 0218 002h] Reserved : 0000
[0DCh 0220 004h] Flags (decoded below) : 0000001E
Physical package : 0
ACPI Processor ID valid : 1
Processor is a thread : 1
Node is a leaf : 1
Identical Implementation : 1
[0E0h 0224 004h] Parent : 000000B0
[0E4h 0228 004h] ACPI Processor ID : 00000006
[0E8h 0232 004h] Private Resource Number : 00000000

[0ECh 0236 001h] Subtable Type : 00 [Processor Hierarchy Node]
[0EDh 0237 001h] Length : 14
[0EEh 0238 002h] Reserved : 0000
[0F0h 0240 004h] Flags (decoded below) : 00000012
Physical package : 0
ACPI Processor ID valid : 1
Processor is a thread : 0
Node is a leaf : 0
Identical Implementation : 1
[0F4h 0244 004h] Parent : 00000024

```

```

[0F8h 0248 004h]          ACPI Processor ID : 00000003
[0FCh 0252 004h]          Private Resource Number : 00000000

[100h 0256 001h]          Subtable Type : 00 [Processor Hierarchy Node]
[101h 0257 001h]          Length : 14
[102h 0258 002h]          Reserved : 0000
[104h 0260 004h]          Flags (decoded below) : 0000001E
                          Physical package : 0
                          ACPI Processor ID valid : 1
                          Processor is a thread : 1
                          Node is a leaf : 1
                          Identical Implementation : 1

[108h 0264 004h]          Parent : 000000EC
[10Ch 0268 004h]          ACPI Processor ID : 00000007
[110h 0272 004h]          Private Resource Number : 00000000

[114h 0276 001h]          Subtable Type : 00 [Processor Hierarchy Node]
[115h 0277 001h]          Length : 14
[116h 0278 002h]          Reserved : 0000
[118h 0280 004h]          Flags (decoded below) : 0000001E
                          Physical package : 0
                          ACPI Processor ID valid : 1
                          Processor is a thread : 1
                          Node is a leaf : 1
                          Identical Implementation : 1

[11Ch 0284 004h]          Parent : 000000EC
[120h 0288 004h]          ACPI Processor ID : 00000008
[124h 0292 004h]          Private Resource Number : 00000000

```

## 第三章 龙芯2K2000固件开发规范

### 1 设计约束

#### 1.1 Audio

##### 1.1.1 HDA

固件通过Verb Table机制实现不同板卡中对声卡的差异化设计。固件开发人员需要根据具体主板的声卡设计正确填写Verb Table。具体参考龙芯代码中关于Realtek的Verb Table配置实现。

固件需要根据芯片手册配置 hda\_i2s\_sel 引脚为HDA工作模式（芯片配置寄存器偏移0x0440寄存器的12:11为x1b，配置寄存器基址需要固件下进行配置，参考代码配置寄存器基址为：0x10010000）。

##### 1.1.2 I2S

固件需要根据芯片手册配置 hda\_i2s\_sel 引脚为I2S工作模式（芯片配置寄存器偏移0x0440寄存器的12:11为10b，配置寄存器基址需要固件下进行配置，参考代码配置寄存器基址为：0x10010000）。

#### 1.2 VBIOS

2K2000内部包含图形处理器（以下简称GPU），位于D6:F0；内核下GPU需要Video BIOS（以下简称VBIOS）的支撑才可以工作正常。

固件对VBIOS处理方式：将固件内默认VBIOS全部读取到内置显卡BAR2空间的最后1MB。

#### 1.3 中断

龙芯2K2000方案，南北桥中断控制器管理的中断可通过中断线或中断消息包路由至Node节点，MSI中断通过中断消息包路由至Node节点，以上中断最终路由至处理器核。见图1-1所示。

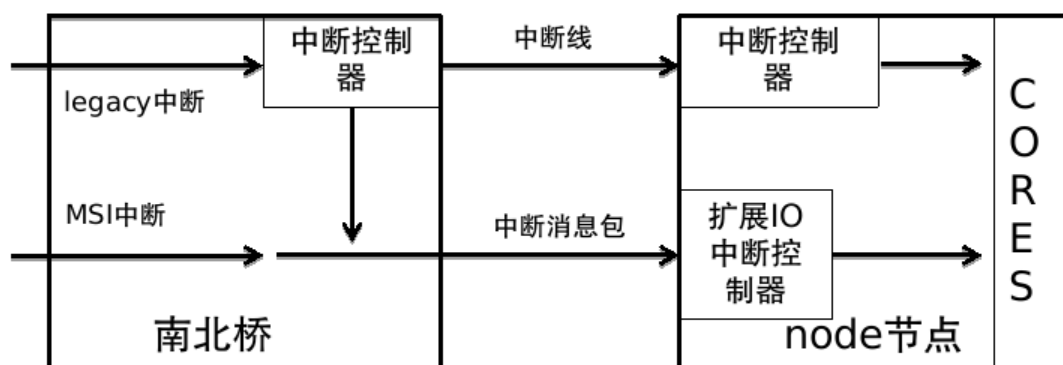


图1-1 龙芯2K2000处理器中断路由示意图

2K2000系统依据ACPI规范使用GSI（global system interrupts，全局系统中断）为中断源分配虚拟化的中断号，一个GSI类型的虚拟中断号唯一标识了其所属的中断域，内核按照不同的中断域实现中断管理。各个中断域对应的GSI中断号范围约定如下：

0-15：LPC/ISA中断域，用于系统外扩LPC设备中断源，如表1-1所示，具体分配情况参考 LPC设备说明。

表1-1 LPC/ISA中断域GSI分配

GSI中断号	中断源	说明
0	HPET	HPET高精度定时器
1	I8042	键盘
2	级联	
3		
4		
5		
6		
7		
8	RTC	实时时钟
9		
10		
11		
12	I8042	鼠标
13		

14	ide0	硬盘
15	Idel	硬盘

16-47: NODE I/O中断控制器中断域，其GSI中断号与中断源向量号的映射关系： $\text{GSI中断号} = \text{中断源向量号} + 16$  中断源向量号为处理器手册中某个中断源在Node中断控制寄存器中的索引，具体参考龙芯 2K2000处理器手册。

48-63: 保留。

64-127: 南北桥I/O中断控制器中断域。其GSI中断号与中断源向量号的映射关系： $\text{GSI中断号} = \text{中断源向量号} + \text{GSI base}$  中断源向量号为处理器手册中某个中断源在南北桥中断控制寄存器中的索引，具体参考龙芯2K2000用户手册；GSI base为MADT表中当前南北桥BIO PIC结构的GSI base成员。其中南北桥集成或外接的PCI兼容设备分配的中断号如表1-2:

表1-2 南北桥I/O中断控制器中断域GSI分配

中断号	中断源	中断号	中断源
64	eMMC_DMA	96	pcie_f0_0
65	SDIO_DMA	97	pcie_f0_1
66	rio1	98	pcie_f0_2
67	rio0	99	pcie_f0_3
68	NB_DMA	100	pcie_f1_0
69	MIPHY	101	pcie_f1_1
70	gmac1_phy	102	PWM4/SE_INT6
71	gmac0_phy	103	PWM5
72	UART/SE_INT5	104	PCIe2
73	I2C/SE_INT4	105	AES/SE_INT2
74	vpu	106	DES/SE_INT1
75	can	107	RSA/SE_INT0
76	gmac0_sbd	108	toy[0]
77	gmac0_pmt	109	toy[1]
78	gmac1_sbd	110	toy[2]
79	gmac1_pmt	111	acpi
80	sata	112	USB2

81	GMAC2_SBD	113	OTG
82	GMAC2_PMT	114	SDIO
83	lpc	115	eMMC
84	hpet[1]	116	rtc[0]
85	hpet[2]	117	rtc[1]
86	usb3	118	rtc[2]
87	hda1	119	hpet[0]
88	pwm[0]	120	i2s_dma[0]
89	pwm[1]	121	i2s_dma[1]
90	pwm[2]	122	i2s/hda
91	pwm[3]	123	gpio_hi
92	dc	124	gpio[0]/gpio[13]
93	gpu	125	gpio[1]/gpio[14]
94	gmem	126	gpio[2]/gpio[15]
95	thsens	127	gpio[3]/gpio[50]

注意：

(1) 若使用EC时需要使用SCI中断，请参考2K2000硬件规范，此中断需要硬件连接到 2K2000 电源管理模块的ACPI GPIO管脚，信号约束参照硬件设计规范，对应GSI中断源为111号。

## 1.4 地址空间

龙芯2K2000地址空间的分布如表1-3：

表1-3 龙芯2K2000地址空间分布

	起始地址	结束地址	说明
地址0	0x0000_0000_0000_0000	0x0000_0000_0FFF_FFFF	内存
地址1	0x0000_0000_1000_0000	0x0000_0000_17FF_FFFF	用于映射内部所使用的设备空间(BAR)
地址2	0x0000_0000_1800_0000	0x0000_0000_19FF_FFFF	用于映射 PCIE控制器对外的IO空间
地址3	0x0000_0000_1A00_0000	0x0000_0000_1AFF_FFFF	Type0 配置空间
地址4	0x0000_0000_1B00_0000	0x0000_0000_1BFF_FFFF	Type1 配置空间
地址5	0x0000_0000_1C00_0000	0x0000_0000_1CFF_FFFF	启动空间，可映射至SPI 和 LIO 上



	起始地址	结束地址	说明
地址6	0x0000_0000_1D00_0000	0x0000_0000_1DFF_FFFF	Local IO Memory
地址7	0x0000_0000_1FC0_0000	0x0000_0000_1FCF_FFFF	可映射至SPI、NAND、SDIO和LIO上
地址8	0x0000_0000_1FE0_0000	0x0000_0000_1FEF_FFFF	芯片配置寄存器空间
地址9	0x0000_0000_1FFF_0000	0x0000_0000_1FFF_FFFF	SPI 配置空间
地址10	0x0000_0000_4000_0000	0x0000_0000_7FFF_FFFF	PCIE MEM 空间
地址11	0x0000_0000_8000_0000	0x0000_0020_7FFF_FFFF	内存
地址12	0x0000_0040_0000_0000	0x0000_007F_FFFF_FFFF	PCIE MEM 空间
地址13	0x0000_00FD_FC00_0000	0x0000_00FD_FFFF_FFFF	PCIE I/O 空间
地址14	0x0000_00FE_0000_0000	0x0000_00FE_0FFF_FFFF	Type0配置空间
地址15	0x0000_00FE_1000_0000	0x0000_00FE_1FFF_FFFF	Type1配置空间
地址16	0x0000_00FE_2000_0000	0x0000_00FE_2FFF_FFFF	Type0配置空间
地址17	0x0000_00FE_3000_0000	0x0000_00FE_3FFF_FFFF	Type1配置空间

## 2 参考配置

### 2.1 MADT配置

表2-1 MADT表约束

域	大小 (字节)	偏移 (字节)	描述
Header			
Signature	4	0	‘APIC’
Revision	1	8	1
Local Interrupt Controller Address	4	36	0x1fe01400

表2-2 CORE PIC结构

域	大小 (字节)	偏移 (字节)	描述
Type	1	0	0x11

Length	1	1	15
Version	1	2	1
ACPI Processor ID	4	3	处理器核UID，与DSDT处理器对象中的_UID值相同
Physical Processor ID	4	7	CPU核物理ID
Flags	4	11	CORE PIC的标志，参考表2-3

表2-3 CORE PIC标志

CORE PIC	大小 (比特)	偏移 (比特)	描述
Flags			
Enabled	1	0	0: CPU不可用 1: CPU可用
Reserved	31	1	必须为0

MADT表的LIO PIC结构参考如下表2-4:

表 2-4 LIO PIC结构

域	大小 (字节)	偏移 (字节)	描述
Type	1	0	0x12, LIO PIC结构
Length	1	1	23
Version	1	2	1
Base Address	8	3	0x1FE01400
Size	2	11	0x80
Cascade vector	2	13	0x0002
Cascade vector mapping	8	15	0x000000000000FFFF

MADT表的EIO PIC结构参考如下表2-5:

表 2-5 EIO PIC结构

域	大小 (字节)	偏移 (字节)	描述
Type	1	0	0x14, EIO PIC结构
Length	1	1	4

Version	1	2	1
Cascade vector	1	3	3
Node	1	4	0
Node map	8	5	1

MADT表的MSI PIC结构参考如下表2-6:

表 2-6 MSI PIC结构

域	大小 (字节)	偏移 (字节)	描述
Type	1	0	0x15, MSI PIC结构
Length	1	1	19
Version	1	2	1
Message Address	8	3	0x1FE01140
Start	4	11	0x40
Count	4	15	0xc0

MADT表的BIO PIC结构参考如下表2-7:

表 2-7 BIO PIC结构

域	大小 (字节)	偏移 (字节)	描述
Type	1	0	0x16, BIO PIC结构
Length	1	1	17
Version	1	2	1
Base Address	8	3	0x10000000
Size	2	11	0x1000
Hardware ID	2	13	0
GSI base	2	15	0x40

MADT表的LPC PIC结构参考如下表2-8:

表 2-8 LPC PIC结构

域	大小 (字节)	偏移 (字节)	描述
---	------------	------------	----

Type	1	0	0x17, LPC PIC结构
Length	1	1	15
Version	1	2	1
Base Address	8	3	0x10002000
Size	2	11	0x1000
Cascade vector	2	13	0x13

## 2.2 FADT配置

此表为操作系统提供了Fixed硬件ACPI描述信息，相关约束见表2-9。

表2-9 FADT表约束

域	大小 (字节)	偏移 (字节)	描述
Header			
Signature	4	0	‘FACP’
Length	4	4	表的长度，包括表头，从偏移量0开始。用于记录整个表的大小
FADT Major Version	1	8	3
FIRMWARE_CTRL	4	36	0
DSDT	4	40	0
Reserved	1	44	0
SCL_INT	2	46	0x6f
SML_CMD	4	48	0
ACPI_ENABLE	1	52	0
ACPI_DISABLE	1	53	0
S4BIOS_REQ	1	54	0
PSTATE_CNT	1	55	0
PM1a_EVT_BLK	4	56	0
PM1b_EVT_BLK	4	60	0
PM1a_CNT_BLK	4	64	0
PM1b_CNT_BLK	4	68	0

PM2_CNT_BLK	4	72	0
PM_TMR_BLK	4	76	0
GPE0_BLK	4	80	0
GPE1_BLK	4	84	0
PM1_EVT_LEN	1	88	8
PM1_CNT_LEN	1	89	4
PM2_CNT_LEN	1	90	0
PM_TMR_LEN	1	91	4
GPE0_BLK_LEN	1	92	8
GPE1_BLK_LEN	1	93	0
GPE1_BASE	1	94	0
CST_CNT	1	95	0
P_LVL2_LAT	2	96	0x65
P_LVL3_LAT	2	98	0x3e9
FLUSH_SIZE	2	100	0
FLUSH_STRIDE	2	102	0
DUTY_OFFSET	1	104	0
DUTY_WIDTH	1	105	0
DAY_ALARM	1	106	0
MON_ALARM	1	107	0
CENTURY	1	108	0
IAPC_BOOT_ARCH	2	109	0
Reserved	1	111	0
Flags	4	112	默认值为0x425，包含如下标志： WBINVD、PROC_CI、SLP_BUTTON、RESET_REG_SUP
RESET_REG	12	116	见表2-10
RESET_VALUE	1	128	0x01
ARM_BOOT_ARCH	2	129	0
FADT Minor Version	1	131	0
X_FIRMWARE_CTRL	8	132	FACS 64bit地址，固件动态生成

X_DSDT	8	140	DSDT 64bit地址，固件动态生成
X_PM1a_EVT_BLK	12	148	见表2-11
X_PM1b_EVT_BLK	12	160	此结构中成员全为0
X_PM1a_CNT_BLK	12	172	见表2-12
X_PM1b_CNT_BLK	12	184	此结构中成员全为0
X_PM2_CNT_BLK	12	196	此结构中成员全为0
X_PM_TMR_BLK	12	208	见表2-13
X_GPE0_BLK	12	220	见表2-14
X_GPE1_BLK	12	232	此结构中成员全为0

注：UEFI固件动态生成X\_FIRMWARE\_CTRL与X\_DSDT之前，需将其初始化为0。

表 2-10 reset寄存器结构

域	大小 (字节)	偏移 (字节)	描述
Address Space Id	1	0	0
Reg Bit Width	1	1	0x20
Reg Bit Offset	1	2	0
Reserved	1	3	0
Address	8	4	0x00000000100d0030

表 2-11 PM1a EVT寄存器结构

域	大小 (字节)	偏移 (字节)	描述
Address Space Id	1	0	0
Reg Bit Width	1	1	0x40
Reg Bit Offset	1	2	0
Reserved	1	3	0
Address	8	4	0x00000000100d000c

表 2-12 PM1a\_CNT寄存器结构

域	大小 (字节)	偏移 (字节)	描述
Address Space Id	1	0	0
Reg Bit Width	1	1	0x20
Reg Bit Offset	1	2	0
Reserved	1	3	0
Address	8	4	0x00000000100d0014

表 2-13 PM\_TMR寄存器结构

域	大小 (字节)	偏移 (字节)	描述
Address Space Id	1	0	0
Reg Bit Width	1	1	0x20
Reg Bit Offset	1	2	0
Reserved	1	3	0
Address	8	4	0x00000000100d0018

表 2-14 GPE0寄存器结构

域	大小 (字节)	偏移 (字节)	描述
Address Space Id	1	0	0
Reg Bit Width	1	1	0x40
Reg Bit Offset	1	2	0
Reserved	1	3	0
Address	8	4	0x00000000100d0028

## 2.3 DSDT配置

### 2.3.1 PCI总线枚举

\_SEG (Segment) 方法描述了PCI段号。

\_CRS (Current Resource Settings) 方法支持总线范围、IO地址范围以及内存地址范围声明。

\_OSC (Operating System Capabilities) 方法，当配置了\_OSC时，仅支持AER、PME和PCI Express

Capability Structure control。

(1) 配置参考如下：

\_SEG为0

\_CRS

使用WordBusNumber()宏声明PCI总线号范围，如表2-15：

表2-15 PCI总线号范围

参数名	值
ResourceUsage	ResourceProducer
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Decode	PosDecode
AddressGranularity	0x0
AddressMinimum	0x0
AddressMaximum	0x00FF
AddressTranslation	0x0000
RangeLength	0x0100
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空

使用QWordIO()宏声明PCI I/O地址范围，如表2-16：

表2-16 PCI I/O地址范围

参数名	值
ResourceUsage	ResourceProducer
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Decode	PosDecode
ISARanges	EntireRange
AddressGranularity	0x00010000
AddressMinimum	0x0000000000004000



AddressMaximum	0x0000000000009FFF
AddressTranslation	0x000000FDFC000000
RangeLength	0x0000000000006000
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
TranslationType	值为空
TranslationDensity	值为空

使用DWordMemory()宏声明32位PCI MEM地址范围如表2-17:

表2-17 32位PCI MEM地址范围

参数名	值
ResourceUsage	ResourceProducer
Decode	PosDecode
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Cacheable	Cacheable
ReadAndWrite	ReadWrite
AddressGranularity	0x00010000
AddressMinimum	0x40000000
AddressMaximum	0x7FFFFFFF
AddressTranslation	0x00000000
RangeLength	0x40000000
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
MemoryRangeType	值为空
TranslationType	值为空

使用QWordMemory()宏声明64位PCI MEM地址范围如表2-18:

表2-18 64位PCI MEM地址范围

参数名	值
ResourceUsage	ResourceProducer
Decode	PosDecode
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Cacheable	Cacheable
ReadAndWrite	ReadWrite
AddressGranularity	0x00000000000010000
AddressMinimum	0x0000004000000000
AddressMaximum	0x0000007FFFFFFFFF
AddressTranslation	0x0000000000000000
RangeLength	0x0000004000000000
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
MemoryRangeType	值为空
TranslationType	值为空

## 2.3.2 设备电源管理

2K2000系统支持USB2.0/USB3.0唤醒系统，需要配置相应的\_PRW。

### 2.3.2.1 USB

\_PRW (Power Resources for Wake)

表2-19 package对象参数表

参数名	值
EventInfo	设备号为D4:F0的USB控制器：值为0xa 设备号为D25:F0的USB控制器：值为0xb
DeepestSleepState	0x3

## 2.3.3 系统休眠唤醒

表2-20 SLP\_TYP规定

处理器	休眠状态	SLP_TYP
2K2000	S0	0
	S3	5
	S4	6

## 2.3.4 串口配置

\_CRS

表2-21串口参数表

参数名	值
ResourceUsage	ResourceConsumer
Decode	PosDecode
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Cacheable	NonCacheable
ReadAndWrite	ReadWrite
AddressGranularity	0x0
AddressMinimum	Node节点串口0值为: 0x000000001FE001E0 南北桥串口0值为: 0x0000000010080000
AddressMaximum	Node节点串口0值为: 0x000000001FE001E7 南北桥串口0值为: 0x00000000100800FF
AddressTranslation	0x0
RangeLength	Node节点串口0值为: 0x0000000000000008 南北桥串口0值为: 0x00000000000000100
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
MemoryRangeType	AddressRangeMemory

TranslationType	TypeStatic
-----------------	------------

中断号资源使用Interrupt()宏声明，各个参数的规定如表2-22：

表2-22 Interrupt()宏参数表

参数名	值
ResourceUsage	ResourceConsumer
EdgeLevel	Level
ActiveLevel	ActiveHigh
Shared	Shared
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
InterruptList	Node节点串口0值为：26 南北桥串口值为：72

\_DSD

表2-23 支持属性表

属性名	值	说明
clock-frequency	Node节点串口值为：100000000 南北桥串口值为：50000000	表示实际外接的串口时钟频率，单位为Hz

### 2.3.5 GPIO配置

\_HID

(1) 2K2000处理器集成的GPIO为LOON000A，2K2000南北桥集成的GPIO为LOON000B，2K2000南北桥集成的ACPI GPIO为LOON000C。

\_CRS

表2-24 QWordMemory()宏参数表

参数名	值
ResourceUsage	ResourceConsumer

Decode	PosDecode
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Cacheable	NonCacheable
ReadAndWrite	ReadWrite
AddressGranularity	0x0
AddressMinimum	南北桥GPIO值为：0x00000000100E0000 南北桥集成的ACPI GPIO值为：0x00000000100D0080 Node节点GPIO值为：0x000000001FE00500
AddressMaximum	南北桥GPIO值为：0x00000000100E0EFF 南北桥集成的ACPI GPIO值为：0x00000000100D009F Node节点GPIO值为：0x000000001FE00517
AddressTranslation	0x0
RangeLength	南北桥GPIO值为：0x0000000000000F00 南北桥集成的ACPI GPIO值为：0x0000000000000020 Node节点GPIO值为：0x0000000000000018
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
MemoryRangeType	AddressRangeMemory
TranslationType	TypeStatic

每个中断号资源使用一个Interrupt()宏声明，各个参数的规定如表2-25：

表2-25 Interrupt()宏参数表

参数名	值
ResourceUsage	ResourceConsumer
EdgeLevel	Level
ActiveLevel	ActiveHigh
Shared	Shared
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空

InterruptList	南北桥GPIO: 123、124、125、126、127 南北桥集成的ACPI GPIO: 111 Node节点GPIO: 16、17、18、19、20、21、22、23
---------------	---

\_DSD

表2-26 支持属性如表

属性名	值	说明
ngpios	南北桥GPIO值为: 64 Node节点GPIO值为: 32 南北桥集成的ACPI GPIO: 16	当前注册的gpio设备包含的gpio pin总数

## 2.3.6 I2C配置

\_HID

(1) LOON0004。

\_UID

各I2C控制器与\_UID对应关系规定如表2-27:

表2-27 I2C控制器与\_UID对应表

控制器	_UID
I2C控制器0	0
I2C控制器1	1

\_CRS

表2-28 QWordMemory()宏参数表

参数名	值
ResourceUsage	ResourceConsumer
Decode	PosDecode
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Cacheable	NonCacheable
ReadAndWrite	ReadWrite

AddressGranularity	0x0
AddressMinimum	I2C控制器值为: I2C 0: 0x0000000010090000 I2C 1: 0x0000000010090100
AddressMaximum	I2C控制器值为: I2C 0: 0x0000000010090007 I2C 1: 0x0000000010090107
AddressTranslation	0x0
RangeLength	I2C控制器值为: I2C 0: 0x0000000000000008 I2C 1: 0x0000000000000008
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
MemoryRangeType	AddressRangeMemory
TranslationType	TypeStatic

## 2.3.7 RTC配置

\_HID

LOON0001。

\_CRS

表2-29 QWordMemory()宏参数表

参数名	值
ResourceUsage	ResourceConsumer
Decode	PosDecode
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Cacheable	NonCacheable
ReadAndWrite	ReadWrite
AddressGranularity	0x0

AddressMinimum	0x00000000100d0100
AddressMaximum	0x00000000100d01FF
AddressTranslation	0x0
RangeLength	0x0000000000000100
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
MemoryRangeType	AddressRangeMemory
TranslationType	TypeStatic

中断号资源使用Interrupt()宏声明，各个参数的规定如表2-30：

表2-30 Interrupt()宏参数表

参数名	值
ResourceUsage	ResourceConsumer
EdgeLevel	Level
ActiveLevel	ActiveHigh
Shared	Exclusive
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
InterruptList	116

## 2.3.8 PWM配置

\_HID

(1) LOON0006。

\_CRS

表2-31 QWordMemory()宏参数表

参数名	值
ResourceUsage	ResourceConsumer



Decode	PosDecode
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Cacheable	NonCacheable
ReadAndWrite	ReadWrite
AddressGranularity	0x0
AddressMinimum	PWM 0: 0x00000000100A0000 PWM 1: 0x00000000100A0100 PWM 2: 0x00000000100A0200 PWM 3: 0x00000000100A0300 PWM 4: 0x00000000100A0400 PWM 5: 0x00000000100A0500
AddressMaximum	PWM 0: 0x00000000100A000F PWM 1: 0x00000000100A010F PWM 2: 0x00000000100A020F PWM 3: 0x00000000100A030F PWM 4: 0x00000000100A040F PWM 5: 0x00000000100A050F
AddressTranslation	0x0
RangeLength	PWM 0: 0x0000000000000010 PWM 1: 0x0000000000000010 PWM 2: 0x0000000000000010 PWM 3: 0x0000000000000010 PWM 4: 0x0000000000000010 PWM 5: 0x0000000000000010
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
MemoryRangeType	AddressRangeMemory
TranslationType	TypeStatic

中断号资源使用Interrupt()宏声明，各个参数的规定如表2-32：

表2-32 Interrupt()宏参数表

参数名	值
ResourceUsage	ResourceConsumer

EdgeLevel	Level
ActiveLevel	ActiveHigh
Shared	Shared
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
InterruptList	PWM 0: 88 PWM 1: 89 PWM 2: 90 PWM 3: 91 PWM 4: 102 PWM 5: 103

### 2.3.9 热区管理

当配置龙芯2K2000处理器温度传感器时，可以参考如下部分示例：

```
Scope (\_SB)
{
    Name (IDDR,0x1fe00000) //ChipReg

    OperationRegion (BASE, SystemMemory, IDDR, 0x438)
    Field (BASE, AnyAcc, NoLock, Preserve)
    {
        Offset (0x428), //thsens
        THSE, 32
    }
    .....
}

Scope (\_TZ)
{
    ThermalZone (THM0)
    {
        Method (_TMP, 0, NotSerialized) // _TMP: Temperature
        {
            Store (\_SB.THSE, Local0)
            Return (C2K (Local0))
        }
    }
}
```

```
Method (_CRT, 0, NotSerialized) // _CRT: Critical Temperature
{
    Return (C2K (0x60))
}

.....
}

Method (C2K, 1, NotSerialized) //Celsius to Kelvin
{
    Add (Multiply (Arg0, 0x0A), 0x0AAC, Local0)
    If (LLessEqual (Local0, 0x0AAC))
    {
        Store (0x0BB8, Local0)
    }

    If (LGreater (Local0, 0x0FAC))
    {
        Store (0x0BB8, Local0)
    }

    Return (Local0)
}
}
```

## 2.3.10 SE设备

\_HID

(1) LOON0003。

\_CRS

中断号资源使用Interrupt()宏声明，各个参数的规定如表2-33：

表2-33 Interrupt()宏参数表

参数名	值
ResourceUsage	ResourceConsumer
EdgeLevel	Level
ActiveLevel	ActiveHigh
Shared	Exclusive

ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
InterruptList	107、106、105、95、73、72、102、92

## 2.3.11 EC配置

\_CPE

(1) SCI中断对应的ACPI GPIO管脚位

## 2.4 FACS配置

表2-34 FACS表约束

域	大小 (字节)	偏移 (字节)	描述
Signature	4	0	‘FACS’
Firmware Waking Vector	4	12	0
Global Lock	4	16	0
Flags	4	20	0
X Firmware Waking Vector	8	24	0
Version	1	32	1

## 2.5 S3休眠地址

SADR，是系统S3休眠时进入固件执行休眠的入口地址，该地址由固件决定，存放固件的S3处理程序。

## 2.6 SRAT配置

表2-35 SRAT表约束

域	大小 (字节)	偏移 (字节)	描述
Header			

Signature	4	0	‘SRAT’
Revision	1	8	2

表2-36 Processor Local APIC/SAPIC Affinity结构

域	大小 (字节)	偏移 (字节)	描述
Type	1	0	0, 代表结构为Processor Local APIC/SAPIC Affinity Structure。
Length	1	1	16
Proximity Domain [7:0]	1	2	0, 代表处理器节点号的0-7位
APIC ID	1	3	处理器Local APIC ID, 见MADT表
Flags	4	4	标志。  0位: 0表示该Processor Local APIC/SAPIC Affinity Structure不可用; 1表示该Processor Local APIC/SAPIC Affinity Structure可用。  [31:1]位: 必须为0
Local SAPIC EID	1	8	0
Proximity Domain [31:8]	3	9	0, 代表处理器节点号的8-31位
Clock Domain	4	12	0

表2-37 Memory Affinity结构

域	大小 (字节)	偏移 (字节)	描述
Type	1	0	1, 代表结构为 Memory Affinity Structure
Length	1	1	40
Proximity Domain	4	2	0
Reserved	2	6	保留
Base Address Low	4	8	内存范围地址的低32位
Base Address High	4	12	内存范围地址的高32位
Length Low	4	16	内存范围大小的低32位

Length High	4	20	内存范围大小的高32位
Reserved	4	24	保留
Flags	4	28	标志，指示内存区域是否已启用并可以热插拔，见表2-38
Reserved	8	32	保留

表 2-38内存热插拔标志

域	大小 ( 比特 )	偏移 ( 比特 )	描述
Enabled	1	0	0: 该Memory Affinity Structure不可用 1: 该Memory Affinity Structure可用
Hot Pluggable	1	1	0
NonVolatile	1	2	0
Reserved	29	3	0

## 2.7 MCFG配置

MCFG表的配置如表2-39、2-40所示。

表2-39 MCFG表约束

域	大小 ( 字节 )	偏移 ( 字节 )	描述
Header			
Signature	4	0	‘MCFG’
Revision	1	8	1
OEMID	6	10	‘LOONGS’
OEM Table ID	8	16	‘LOONGSON’
OEM Revision	4	24	1

表2-40 Memory Mapped Enhanced Configuration Space Base Address Allocation  
Structure[0]结构

域	大小 ( 字节 )	偏移 ( 字节 )	描述
Base Address	8	0	0x000000fe00000000

PCI Segment Group Number	2	8	0
Start Bus Number	1	10	0x00
End Bus Number	1	11	0xFF
Reserved	4	12	0

## 2.8 SPCR配置

表2-41 SPCR表约束

域	大小 (字节)	偏移 (字节)	描述
Signature	4	0	‘SPCR’
Interface Type	1	36	0
Reserved	3	37	0
Space ID	1	40	0
Bit Width	1	41	0
Bit Offset	1	42	0
Encoded Access Width	1	43	1
Address	8	44	0x000000001FE001E0
Interrupt Type	1	52	0
PCAT-compatible IRQ	1	53	0
Interrupt	4	54	0
Baud Rate	1	58	0
Parity	1	59	0
Stop Bits	1	60	0
Flow Control	1	61	0
Terminal Type	1	62	0
Language	1	63	0
PCI Device ID	2	64	0xFFFF
PCI Vendor ID	2	66	0xFFFF
PCI Bus	1	68	0

PCI Device	1	69	0
PCI Function	1	70	0
PCI Flags	4	71	0
PCI Segment	1	75	0
Reserved	4	76	0



## 第四章 龙芯7A1000内置显卡软硬件设计规范

### 1 范围

本规范规定龙芯7A1000内置显卡的硬件设计要求，VBIOS固件的存放位置等。本规范适用于龙芯7A1000桥片主板。建议其它系统厂商遵循此规范开发相关产品。

### 2 术语与定义

本规范所用术语定义如下：

**固件：**Firmware，写入ROM、EPROM等非易失存储器中的程序，负责控制和协调集成电路。

**BIOS：**基本输入输出系统，Basic Input Output System，一组固化到主板上一个ROM芯片上的程序，它保存着计算机基本输入输出程序、系统设置信息、开机后自检程序和系统自启动程序。BIOS与硬件系统集成在一起，也被称为固件，本规范中固件和BIOS不做区分。

**UEFI：**统一可扩展固件接口，Unified Extensible Firmware Interface，是UEFI Forum推广和发展的全新类型PC固件的体系结构、接口和服务的建议标准。主要目的是提供在OS加载之前在所有平台上一致、正确指定的启动服务。

**PMON：**一种兼有BIOS和boot loader部分功能的开放源码软件。

**VBIOS(Video BIOS)：**VBIOS是显卡的BIOS，VBIOS提供一些和显示相关的功能，并存放显示芯片与驱动程序之间的控制程序，另外还存放有显示卡型号、规格、生产厂家、出厂时间等信息。

**PCI (Peripheral Component Interconnect)：**是连接电子计算机主板和外部设备的总线标准，用于定义局部总线的标准。此标准允许在计算机内安装多达10个遵从PCI标准的扩展卡。

**Encoder：**信号编码器，用于将DVO信号转换为其他的显示信号，如模拟信号、LVTM或TMDS等显示信号。

**Crtc：**显示控制器，显示控制器把数据从显存中将要显示数据取出来,然后对这些数据做一定的处理送到显示屏。

**Connector：**显示接口连接器，指显示接口硬件，例如VGA连接器，HDMI连接器等。

### 3 架构关系

龙芯7A1000桥片中集成显示控制器和GPU，龙芯7A1000显示驱动包括内核驱动和VBIOS固件，由内核驱动和VBIOS固件共同控制龙芯显示控制器和GPU，实现显示，背光调节，渲染等功能。VBIOS固件需要使用VBIOS生成工具生成，并存放固定位置(详见第5章)，VBIOS生成工具的使用详见《龙芯VBIOS1.0生成工具使用手册》，龙芯7A1000内置显卡驱动初始化设备时，从VBIOS固件中解析硬件信息，配置，代码等内容。内置显卡内核驱动与VBIOS固件层次关系如图3-1所示：

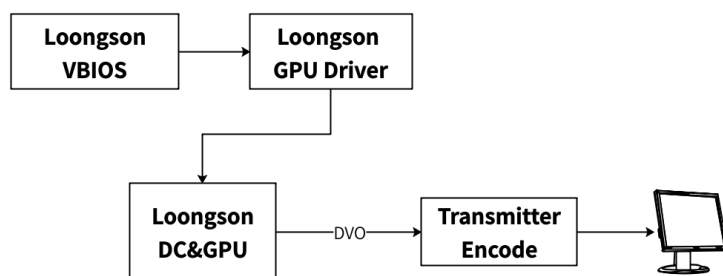


图3-1 显示驱动，固件与显示硬件的关系

## 4 显示类型和模式

龙芯7A1000桥片中集成的显示控制器包含两路DVO端口，通过连接外置的编码芯片(Encoder)可实现模拟或者数字信号输出。当前操作系统内核中已集成LT8618、IT66121、MS7210三款HDMI转换芯片的驱动，正常使用时需由VBIOS配合，实现点亮显示器和待机唤醒等功能。龙芯7A1000桥片内置显卡支持的显示模式，如表4-1所示。

表4-1 显示模式支持情况

显示类型	描述
CRT	支持模拟信号显示器(VGA)，通过DVO端口连接额外的转换芯片来支持。
显示面板	支持TMDS(DVI,HDMI)和LVDS兼容显示面板，通过DVO端口连接额外的转换芯片来支持。

龙芯7A1000显示控制器支持多种分辨率，每路显示最大支持1920x1080@60Hz，支持RGB565，ARGB8888两种色深，龙芯显示控制器要求水平分辨率满足256字节对齐，对齐计算方法：水平宽度 × 色深/8，计算后得到整数则表示对齐，否则表示不对齐，以1152x864分辨率16位色深为例， $1152 \times 16 / 8 / 256 = 9$ ，1152x864分辨率是可以正常显示的，在RGB565色深模式下，由于多种分辨率无法满足对齐要求，多屏模式下可能出现显示异常，常用分辨率支持情况见表4-2。

表4-2 分辨率支持表

分辨率	RGB565	ARGB8888
800x600	不支持	不支持
1024x768	支持	支持
1152x864	支持	支持
1680x1050	不支持	不支持
1280x1024	支持	支持

1600x1200	不支持	支持
1920x1080	支持	支持

## 4.1多屏显示方案

龙芯7A1000显示驱动支持单路和两路同时显示，其中两路显示支持几种不同的模式，支持的情况如表4-3所示。

表4-3 多屏支持情况

显示配置模式	描述	支持情况
Single	支持单独一个显示器	支持
Twin	支持两个显示器，以相同分辨率和时序，显示相同画面，	支持
Clone	支持两个显示器，以不同分辨率和时序，显示相同的画面	不支持
Extended	支持两个显示器，以左右或上下扩展的形式显示完整画面	支持

龙芯7A1000显示驱动支持多种硬件链接方案。

方案A: 硬件上选用一款多通道输出视频编解码转换芯片（Encoder）来实现多屏镜像(Twin)显示，如图4-1所示。注意：该方案只能支持镜像模式的双屏显示。

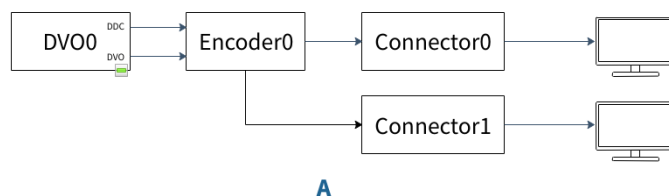


图4-1 镜像双屏显示硬件链接方案

方案B: 实现两路独立的双屏显示，需要同时使用两路显示控制器分别连接相应的视频编解码转换芯片(Encoder)芯片。如图4-2，实现多屏扩展(Extended)和多屏镜像(Twin)功能。支持单屏-多屏切换，多屏-单屏切换和多屏不同模式间的切换。

方案C: 两路独立的双屏显示，支持显示控制器(Crtc)和视频编解码转换芯片(Encoder)之间支持交叉连接，如图4-2中，采用这种方案时7A1000显示控制器在硬件设计时需要注意，同一个显示控制器的DVO显示通道和DDC通道必须同时连接到一个视频编解码转换芯片(Encoder)。

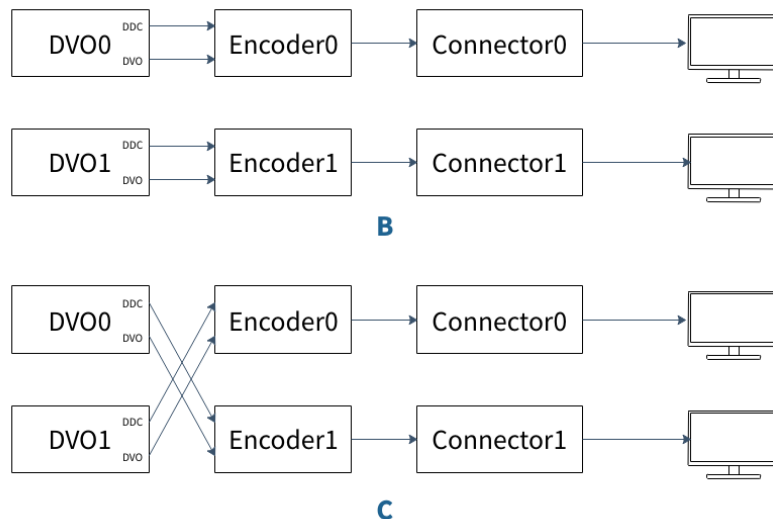


图4-2 独立双屏显示硬件连接方案

针对方案B和方案C，显示控制器(Crtc)和视频编解码转换芯片(Encoder)之间的连接关系，需要通过VBIOS生成工具进行相应的配置，仅在硬件布线正确连接，并且在VBIOS中正确配置连接关系后才能正常显示，同时显示接口能获取正确的EDID信息，才能实现正常的显示器的拔插检测功能。

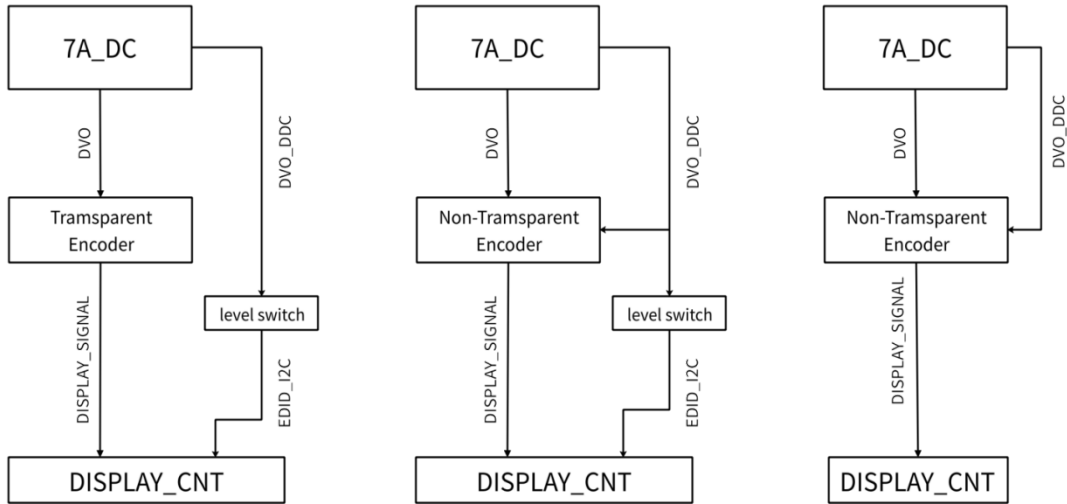
## 4.2 DDC通道硬件连接方案

龙芯7A1000桥片对每个显示通路的DDC连接方案做了具体规定。龙芯7A1000内置显卡支持三种DDC通道的连接方案。

方案A：使用不需要配置的Encoder芯片，则DVO0\_SCL和DVO0\_SDA经过电平转换后直接连接显示接口，硬件连接方案如图4-3所示；

方案B：使用需要配置(非透明)的Encoder芯片，则DVO0\_SCL和DVO0\_SDA需要同时作为Encoder芯片的配置通道和EDID读取通道使用，硬件连接方案如图4-4所示；

方案C：使用需要配置(非透明)的视频编解码转换芯片(Encoder)，且视频编解码转换芯片(Encoder)支持EDID的读取功能，则只需要将DVO0\_SCL和DVO0\_SDA作为视频编解码转换芯片(Encoder)的配置通道，通过向视频编解码转换芯片(Encoder)发送命令来读取EDID信息，如图4-5所示。



### 4.3 获取EDID

龙芯7A1000内置显卡提供两种EDID获取方案。

方案A：使用显示控制器的DDC通道读取外部EDID信息，驱动根据不同显示设备，读取不同的EDID信息；

方案B：在VBIOS中预存需要的EDID信息，适用于以下三种情况：

- 1.外部显示设备无法提供EDID信息；
- 2.外部显示设备仅能提供一个EDID信息，而实际需要使用多组分辨率。
- 3.无法采用标准的获取方式获取EDID。例如，无法通过i2c从设备0x50地址中读取到EDID。

### 4.4热插拔探测

龙芯7A1000内置显卡通过软件轮询是否能够获取EDID信息来判定是否正常连接显示器，实现热插拔探测，如果能正常读取EDID信息则表示显示器连接，否则表示显示器断开。

如需支持热插拔功能，需要按照4.2节所述设计硬件，保证驱动能够正确读取到EDID。如需不支持热插拔功能，需要在VBIOS中关闭热插拔探测，关闭热插拔探测情况下，显示接口一直输出显示信号，在这种模式下，如获取不到外部显示设备的EDID信息，仅能支持预存储在VBIOS或者系统中分辨率。

### 4.5 背光控制

使用7A1000内置显卡方案时，硬件设计约束为：显示屏背光开关控制必须使用7A桥片上的GPIO46和GPIO47，显示屏背光的亮度控制必须使用7A桥片上的PWM3(GPIO07)，如表4-4所示，否则背光功能可能无法正常工作。

表4-4 背光控制硬件链接关系

LCD_EN(LCD backlight enable)	CLKSEL0(GPIO46)默认下拉
------------------------------	---------------------

LCD_VDD_EN(LCD power enable)	CLKSEL1(GPIO47)默认下拉
LCD_PWM(LCD backlight PWM)	PWM3(GPIO07)

## 5 VBIOS存储约定

使用制作工具生成的VBIOS固件二进制文件存储在 BIOS 同一个ROM中。

## 第五章 龙芯7A2000内置显卡软硬件设计规范

### 1 范围

本规范详细介绍龙芯7A2000桥片内置显卡的硬件设计约束和VBIOS设备属性表。本规范适用于使用龙芯7A2000桥片的主板。请其它系统厂商遵循此规范开发相关产品。

### 2 术语与定义

本规范所用术语定义如下：

**固件：**Firmware，写入ROM、EPROM等非易失存储器中的程序，负责控制和协调集成电路。

**BIOS：**基本输入输出系统，Basic Input Output System，一组固化到主板上一个ROM芯片上的程序，它保存着计算机基本输入输出程序、系统设置信息、开机后自检程序和系统自启动程序。BIOS与硬件系统集成在一起，也被称为固件，本规范中固件和BIOS不做区分。

**UEFI：**统一可扩展固件接口，Unified Extensible Firmware Interface，是UEFI Forum推广和发展的全新类型PC固件的体系结构、接口和服务的建议标准。主要目的是提供在OS加载之前在所有平台上一致、正确指定的启动服务。

**PMON：**MIPS架构机器上使用的一种兼有BIOS和boot loader部分功能的开放源码软件。

**VBIOS(Video BIOS)：**VBIOS是显卡的BIOS，VBIOS提供一些和显示相关的功能，并存放显示芯片与驱动程序之间的控制程序，另外还存放有显示卡型号、规格、生产厂家、出厂时间等信息。

**PCI (Peripheral Component Interconnect)：**是连接电子计算机主板和外部设备的总线标准，用于定义局部总线的标准。此标准允许在计算机内安装多达10个遵从PCI标准的扩展卡。

**Encoder：**型号编码器，用于将DVO型号转换为其他的显示信号，如模拟型号、LVTM或TMDS等显示信号。

**CRTC：**显示控制器(DC)在驱动中的抽象，显示控制器把数据从显存中将要显示数据取出来,然后对这些数据做一定的处理送到显示屏。

**Connector：**显示接口连接器，指显示接口硬件，例如VGA连接器，HDMI连接器等。

**EDID：**显示器识别数据，存储在显示器中的DDC存储器中，当电脑主机与显示器连接后，电脑主机会通过DDC通道读取显示器中存储的EDID。

### 3 架构关系

龙芯7A2000桥片中集成显示控制器(DC)和图形处理器(GPU)，龙芯显示驱动包括内核驱动和VBIOS固件，GPU固件。由内核驱动和VBIOS固件，GPU固件共同控制龙芯显示控制器和GPU，实现显示、背光调节、二三维渲染加速等功能。VBIOS固件需要由专门的工具制作，龙芯7A2000桥片内置GPU驱动初始化设备时，从显存中获取VBIOS固件，解析VBIOS固件中包含的设备属性表，设

备配置，代码等内容。龙芯内置显卡驱动与VBIOS固件层次关系如图3-1所示：

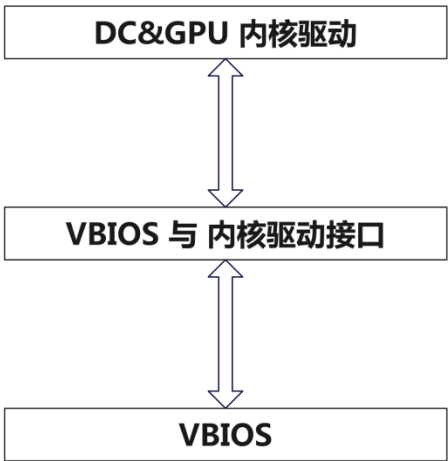


图3-1 显示驱动，固件与显示硬件的关系

## 4 7A2000 桥片硬件显示管线设计规范

### 4.1 硬件显示管线简介

显示管线是显卡中重要的组成部分，显示管线负责扫描显存中的数据并将这些数据送到终端显示，图形管线由显示控制器，编码器，信号转换器，连接器，I2C 控制器等多种设备组成，显示管线的结构如图 4-1 所示，每个设备的功能见表4-1所示。

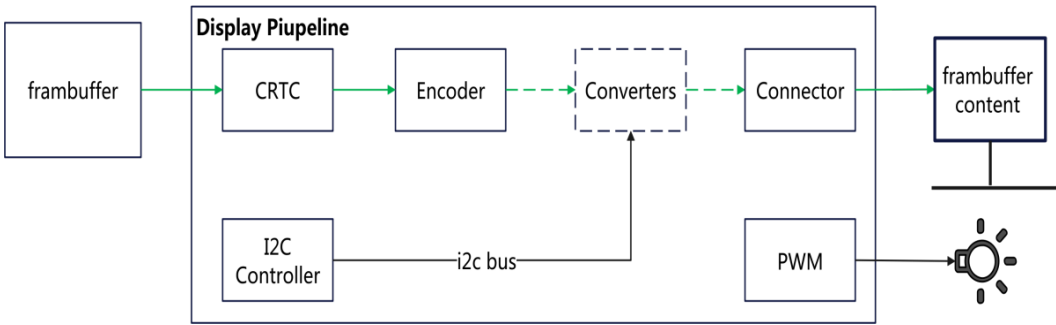


图4-1 显示管线

表4-1 设备名称功能

设备名称	功能描述
CRTC	显示控制器，对显示缓冲区进行扫描，并产生时序信号的硬件模块。
Encoder	编码器，将显示控制器产生的时序信号转换成外部设备所需要的信号。
Converter	信号转换器，将编码器产生的时序信号转换成外部设备所需要的信号，可选，可支持多个。



设备名称	功能描述
Connector	连接器，连接物理显示设备的连接器。
I2C	I2C 控制器，用于控制外部的信号转换器或者通过连接器读取EDID。
PWM	PWM 控制器，用于控制外部背光。
VRAM	显存

本规范约束搭载7A2000桥片主板的硬件显示管线设计。主板设计时需要符合本规范中的设计约束，能保证显示驱动软件的兼容性。

## 4.2显示器显示管线方案

### 4.2.1 显示器显示管线

本规范中定义了两种显示器方案，连接转换芯片和不连接转换芯片两种方案，在设计硬件显示管线方案时可根据具体的需求决定是否使用转换芯片，不同的硬件显示管线方案有不同的约束。

**方案一：**不需要连接信号转换芯片时，硬件设计约束示意图如图4-2，要求将7A2000桥片显示控制器的I2C直接连接到显示接口连接器上。用于实现读取EDID和使用VGA接口时的热插拔探测功能。

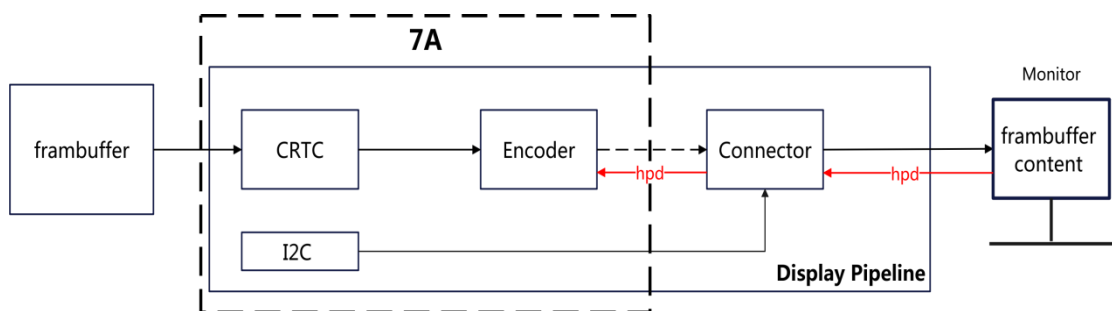


图4-2 无信号转换芯片的显示器显示管线

**方案二：**需要连接信号转换芯片时，硬件设计约束如图4-3，要求将7A2000桥片的I2C连接到转换芯片的 I2C 从接口上，将转换芯片的I2C主接口连接到对应的显示接口上。用于通过信号转换芯片实现读取EDID和热插拔探测功能。可以使用7A桥片上GPIO52来控制转换芯片的复位。在有低功耗需求的场景下对转换芯片的功耗进行控制（不强制）。

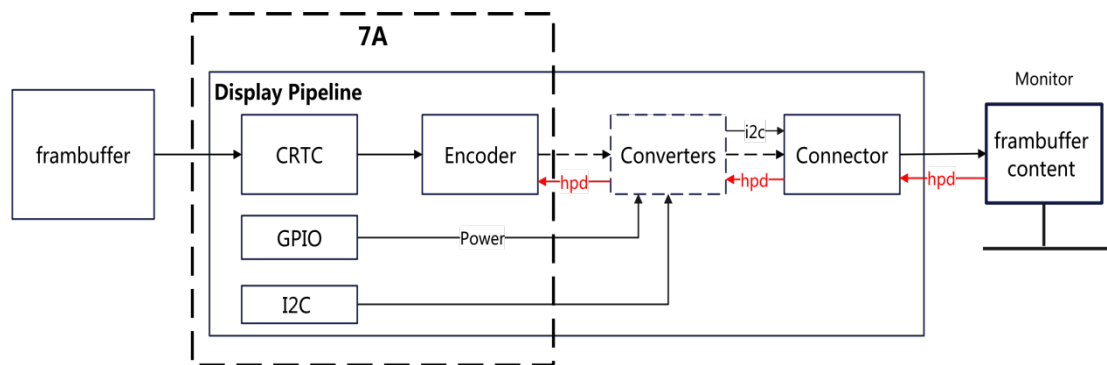


图4-3 有信号转换芯片的显示器显示管线

## 4.2.2 获取EDID

在显示器方案中，EDID由显示器提供。在不连接信号转换芯片时，通过显示控制器中I2C直接探测EDID，在连接信号转换芯片时，通过显示控制器的I2C向信号转换芯片发送探测指令，由显示转换芯片完成进一步的探测，并将EDID返回给驱动。驱动根据EDID中的信息设置显示控制器的输出时序。

## 4.2.3 热插拔探测

在设计显示器显示管线热插拔探测时，请按照如下的原则进行：

1. 直接使用内置VGA作为显示接口时，建议使用轮训方式进行热插拔探测，在硬件设计上务必保证具备探测到EDID，否则热插功能拔会失效。
2. 直接使用内置HDMI作为显示接口时，使用中断方式进行热插拔探测。
3. 使用外接转换芯片时，要求连接转换芯片的热插拔中断管脚。

## 4.3 液晶屏显示管线方案

### 4.3.1 液晶屏显示管线

本规范中约束了液晶屏硬件显示管线的实现方案，在液晶屏方案中，支持背光调节，不支持液晶屏热插拔功能。

**方案说明：**将桥片作为信号转换芯片的主控时，硬件参考设计如图4-4，将显示控制器的I2C连接到转换芯片。用于实现对信号转换芯片的控制和读取EDID功能。使用桥片的GPIO 管脚来控制转换视频转换芯片的电源，背光电源和液晶屏电源。

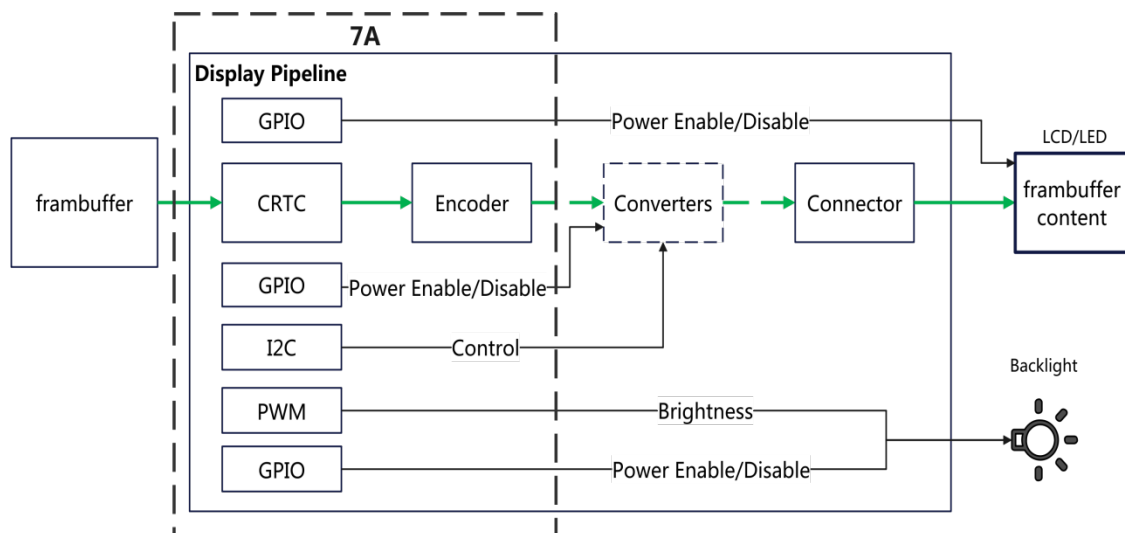


图4-4 液晶屏显示管线

### 4.3.1 获取EDID

在液晶屏方案中，EDID可由液晶屏或者显示转换芯片提供，在前两者无法提供的情况下也可由VBIOS提供。EDID用于提供显示控制器的输出时序参数。EDID由液晶屏或显示转换芯片提供时，通过7A2000桥片显示控制器中的I2C向信号转换芯片发送探测指令，这时可由显示转换芯片返回内部存储的EDID或者探测液晶屏中存储的EDID，并将EDID返回。如果EDID由VBIOS 提供，则由驱动解析VBIOS中的EDID数据段。

**需要特别注意的是：**EDID数据应该由液晶屏厂家提供，厂家在制作EDID时，EDID中的时序必须要符合液晶屏的时序要求，另外EDID中的features字段(第0x18字节)的第0位请勿写1，该位表示显示设备支持工作范围内的GTF时序，该位置1后驱动将根据EDID中提供的时序范围增加EDID中未列出的GTF时序，使用新增加的分辨率存在不可预测的结果不保证显示功能的正确性和稳定性。关于更多的EDID的信息，请参考《VESA ENHANCED EXTENDED DISPLAY IDENTIFICATION DATA STANDARD》

### 4.3.2 屏幕背光

本规范中约束了屏幕背光的实现方案，将桥片作为背光的主控时，由桥片上的GPIO和PWM分别控制背光电源和背光亮度。推荐使用桥片的GPIO 和 PWM 资源和状态，见表4-1。

表4-2 7A2000桥片GPIO, PWM 资源使用表

功能	GPIO 资源与状态
屏幕电源控制	7A2000 GPIO 47，下拉关闭，上拉使能，默认下拉。
屏幕背光控制	7A2000 GPIO 46，下拉关闭，上拉使能，默认下拉。
背光亮度调节	7A2000 PWM 3 (GPIO07)。

信号转换芯片复位控制	7A2000 GPIO 52。
------------	-----------------

## 4.5 显存设计约束

龙芯7A2000桥片支持外接32或16位宽的显存颗粒，出于对显示性能方面的考虑，推荐使用32bit位宽，容量不小于256 MB。在不接显存颗粒时，7A2000桥片内置GPU将无法正常工作，可能导致操作系统无法正常启动。

## 4.6 信号转换芯片选型约束

对于非透明信号转换芯片选型时需要满足如下的要求：

1. 具备两个I2C 控制器，作为从设备，用于与主控通信，作为主设备，用于热插拔探测，探测和获取EDID。
2. 在液晶屏方案下，如需支持多分辨率，要求信号转换芯片支持缩放功能，缩放后的显示质量由转换芯片决定。
3. 具备存储EDID 的功能(不强制)。
4. 具备低功耗模式。具备电源控制(不强制)。

## 4.7 多屏显示方案

龙芯7A2000桥片中有两个显示控制器，最多可支持三路显示，其中HDMI0和 VGA 复用相同的信号源，这两路显示内容相同，HDMI1 为独立的信号源，可与 HDMI0 或 VGA 组成组成扩展显示模式。

### 4.7.1 单屏方案

在单屏方案中，可选择单独使用HDMI0或者 VGA，也可单独使用HDMI1。如果使用VGA接口时使用轮训作为热插拔探测方案。使用HDMI时使用中断作为热插拔探测方案。

### 4.7.2 双屏方案

在双屏方案中，可使用HDMI0与HDMI1组成双屏方案，可使用VGA 与HDMI1 组成双屏方案，采用这两种方案时，**两个屏幕可显示不同的内容，实现扩展显示功能**；如使用HDMI0和VGA组成双屏方案时，**两个屏幕只能显示相同的内容，且必须使用相同品牌和型号的显示器，否则存在无法正确显示的风险。**

### 4.7.3 三屏方案

在三屏方案中，需要同时使用 VGA，HDMI0，HDMI1组成三屏方案，由于VGA和HDMI0是相同的信号源，这两路必须使用相同品牌，相同型号的显示器，否则存在无法正确显示的风险。出现不能正确显示的问题时，符合预期。在硬件设计时需要增加如下的开关电路保证VGA和HDMI0都能探测到EDID，在三屏方案时VGA 和 HDMI0 需要热插拔探测(HPD)对多路复用电路（MUX）开关进行切换，在VBIOS 配置方面，VGA和 HDMI0 的热插拔探测方法需要配置为轮询。

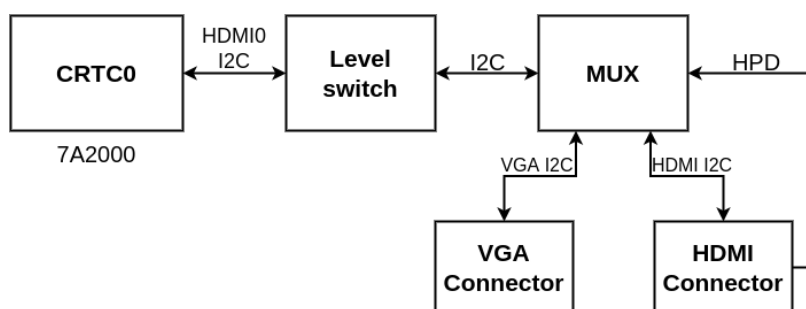


图4-5热插拔多路复用电路示意图

## 5 7A2000 VBIOS设备属性标配

龙芯VBIOS提供一种结构化的数据，用于描述无法被GPU驱动程序动态探测的设备属性和设备连接拓扑，如图5-1所示，通过固件传参的形式将这些设备属性传递给内核。这些设备可以在GPU内部或直接与GPU相连。关于VBIOS的更多信息，详见参考《龙芯VBIOS规范V2.0》文档，VBIOS生成工具的使用方法详见集成VBIOS 生成工具中的《龙芯VBIOS生成工具使用手册》。

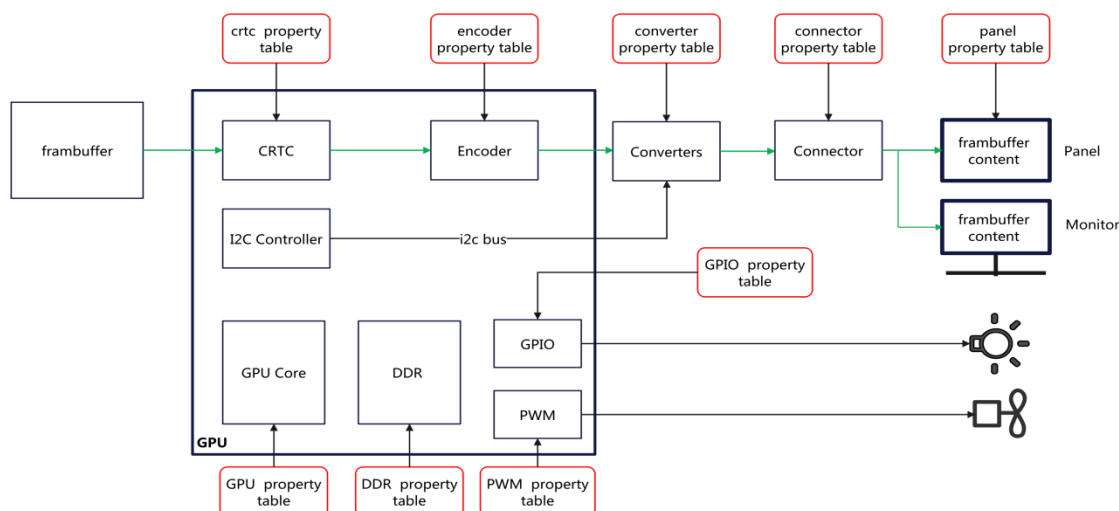


图 5-1 设备属性表示意图

## 5.1显示控制器属性表

表5-1 显示控制器属性表

属性名称	类型(大小)	功能说明
feature	u32(4B)	设备特性标记位。
crtc_id	u32(4B)	显示控制器硬件编号。编号从0开始。
encoder_id	u32(4B)	与该显示控制器相连的内置信号编码器编号。
max_freq	u32(4B)	控制器硬件支持的最大像素时钟频率，参考芯片手册填写。
max_width	u32(4B)	控制器硬件能支持的最大分辨率宽度，参考芯片手册填写。
max_height	u32(4B)	控制器硬件能支持的最大分辨率高度，参考芯片手册填写。
is_vb_timing	u8(4B)	是否由vbios提供显示控制器的时序，默认值为 0，已废弃，工具中不提供配置界面。

## 5.2内置编码器属性表

表5-2 信号转换器属性表

属性说明	类型(大小)	说明
feature	u32(4B)	设备特性标记位。
type	u32(4B)	内置编码器输出信号类型。
connector_id	u32(4B)	显示链路编号。

## 5.3 外接信号转换器属性表

表5-3 内置编码器属性表

属性名称	类型(大小)	功能说明
feature	u32(4B)	设备特性标记位。
i2c_id	u32(4B)	与转换芯片相连的i2c编号。默认值为 -1，已废弃。工具中不提供配置界面。
connector_id	u32(4B)	显示链路编号。
type	u32(4B)	外接编码器芯片输出信号的类型
config_type	u32(4B)	编码器芯片配置方式。

属性名称	类型(大小)	功能说明
chip_id	u32(4B)	外接转换芯片的型号。
chip_addr	u32(4B)	外接转换芯片i2c从设备地址。

## 5.4 连接器属性表

表5-4 接口连接器属性表

属性名称	类型(大小)	功能说明
feature	u32(4B)	设备特性标记位。
i2c_id	u32(4B)	与连接器相连i2c编号。默认值为 -1，已废弃，工具中不提供配置界面。
internal_edid	u32(256B)	存放由vbios提供的edid。
type	u32(4B)	连接器接口类型。
hotplug	u32(4B)	热插拔探测方式的类型。
edid_method	u32(4B)	获取EDID方式。
irq_gpio	u32(4B)	用于热插拔探测的GPIO号。7A2000 以后不再使用。工具中不提供配置界面。
gpio_placement	u32(4B)	使用GPIO的位置，，默认值为 7A，已废弃，工具中不提供配置界面。

## 5.5 GPIO 设备属性表

表5-5 GPIO 设备属性表

属性名称	类型(大小)	说明
feature	u32(4B)	设备特性标记位
gpio_id	u32(4B)	使用的 gpio 编号
usage	u32(4B)	gpio 用途

## 5.6 PWM 设备属性表

表5-6 PWM设备属性表

属性名称	类型(大小)	说明
------	--------	----

feature	u32(4B)	设备特性标记位。
pwm_id	u32(4B)	pwm 设备编号，参考主板进行配置。
polarity	u32(4B)	pwm 设备极性配置，参考主板进行配置。
peroid	u32(4B)	pwm 设备周期，单位Hz，参考芯片手册进行配置。
usage	u32(4B)	pwm 控制器的用途，参考主板进行配置。

## 5.7 GPU属性表

表5-7 GPU设备属性表

属性名称	类型(大小)	说明
feature	u32(4B)	设备特性标记位
gpu_freq_level	u32(4B)	GPU 频率调节档位，参考硬件配置。
gpu_freq	u32(4B)	GPU 最大频率，参考硬件配置。
num_gpu_shaders	u32(4B)	GPU 流处理器个数，参考硬件配置。

## 5.8显存属性表

表5-8 显存设备属性表

属性名称	类型(大小)	说明
feature	u32(4B)	设备特性标记位。
vram_type	u32(4B)	显存颗粒类型。参考主板设计配置。
vram_bit_width	u32(4B)	显存颗粒位宽。参考主板设计配置。
vram_capacity	u32(4B)	显存容量。参考主板设计配置。
vram_freq	u32(4B)	显存频率。参考主板设计配置。

## 6 efifb

efifb适用于符合UEFI规范的固件。efifb既支持带有图形输出协议（GOP）显示器的固件。efifb可以复用固件启动时的分辨率和帧缓冲区地址。因此可以实现在操作系统没有显示驱动的情况下支持基本的图形显示功能。

龙芯BIOS中已默认集成支持GOP驱动，在操作系统中没有集成显卡驱动或不满足显卡驱动运行条件时，可继续由efifb完成图形显示功能。**要求固件传递帧缓冲地址，分辨率，色深必须和固**



件使用相同，才能保证进入系统后正常显示。

系统使用efifb进入桌面后可正常操作应用程序，如开启浏览器、终端、文件管理等应用程序；efifb只能提供基础的图形显示功能，不支持画面旋转、切换分辨率、更改刷新率、显示器热插拔、显示器信息获取等功能。为使操作系统获得完整的使用体验，使用efifb启动后请尝试安装最新显卡驱动。

## 7 附录

### 7.1 显示器方案配置示例

如表7-1和表7-2所示，为使用内置HDMI直接输出显示器时vbios中设备信息的描述表。

表7-1 内置HDMI配置

设备	显示控制器	内置编码器	外接编码器	连接器
属性/配置	feature (0)	feature (0)	feature (0)	feature (0)
	crtc_id (0)	encoder_id (0)	i2c_id (0)	i2c_id (0)
	encoder_id (0)	connector_id (0)	connector_id (0)	internal_edid (不配置)
	max_freq (340000)		type (TMDS)	type (HDMI)
	max_width (4096)		config_type (driver)	hotplug (irq)
	max_height (4096)		chip_id (INTERNA_HDMI)	edid_method (I2C)
			chip_addr (0)	

表7-2 GPU配置

设备	GPU	显存
属性/配置	feature (0)	feature (0)
	gpu_freq_level (3)	vram_type (ddr4)
	gpu_freq (500MHz)	vram_bit_width (32 bit)
	num_gpu_shaders (64)	vram_capacity (1GB)
		vram_freq (2400MHz)

### 7.2 液晶屏方案配置示例

如表7-3和表7-4所示，为使用外接eDP转换芯片输出液晶屏时vbios对各设备信息的描述。

表7-3 外接eDP配置

设备	显示控制器	内置编码器	外接编码器	连接器	PWM
属性/配置	feature (0)	feature (0)	feature (0)	feature (0)	feature (0)
	crtc_id (0)	encoder_id (0)	i2c_id (0)	i2c_id (0)	pwm_id (3)
	encoder_id (0)	connector_id (0)	connector_id (0)	internal_edid (不配置)	polarity (1000000)
	max_freq (340000)		type (LVDS)	type (eDP)	peroid (1)
	max_width (4096)		config_type (driver)	hotplug (connected)	usage (0)
	max_height (4096)		chip_id (lt9721)	edid_method (I2C)	
			chip_addr (0x3a)		

表7-4 GPU配置

设备	GPU	显存
属性/配置	feature (0)	feature (0)
	gpu_freq_level (3)	vram_type (ddr4)
	gpu_freq (500MHz)	vram_bit_width (32 bit)
	num_gpu_shaders (64)	vram_capacity (1 GB)
		vram_freq (2400MHz)

## 第六章 龙芯2K2000内置显卡软硬件设计规范

### 1 范围

本规范详细介绍龙芯2K2000芯片内置显卡的硬件设计约束和VBIOS设备属性表。本规范适用于使用龙芯2K2000芯片的主板。请其它系统厂商遵循此规范开发相关产品。

### 2 术语与定义

本规范所用术语定义如下：

**固件：**Firmware，写入ROM、EPROM等非易失存储器中的程序，负责控制和协调集成电路。

**BIOS：**基本输入输出系统，Basic Input Output System，一组固化到主板上一个ROM芯片上的程序，它保存着计算机基本输入输出程序、系统设置信息、开机后自检程序和系统自启动程序。BIOS与硬件系统集成在一起，也被称为固件，本规范中固件和BIOS不做区分。

**UEFI：**统一可扩展固件接口，Unified Extensible Firmware Interface，是UEFI Forum推广和发展的全新类型PC固件的体系结构、接口和服务的建议标准。主要目的是提供在OS加载之前在所有平台上一致、正确指定的启动服务。

**PMON：**MIPS架构机器上使用的一种兼有BIOS和boot loader部分功能的开放源码软件。

**VBIOS(Video BIOS)：**VBIOS是显卡的BIOS，VBIOS提供一些和显示相关的功能，并存放显示芯片与驱动程序之间的控制程序，另外还存放有显示卡型号、规格、生产厂家、出厂时间等信息。

**PCI (Peripheral Component Interconnect)：**是连接电子计算机主板和外部设备的总线标准，用于定义局部总线的标准。此标准允许在计算机内安装多达10个遵从PCI标准的扩展卡。

**Encoder：**型号编码器，用于将DVO型号转换为其他的显示信号，如模拟型号、LVTM或TMDS等显示信号。

**CRTC：**显示控制器(DC)在驱动中的抽象，显示控制器把数据从显存中将要显示数据取出来，然后对这些数据做一定的处理送到显示屏。

**Connector：**显示接口连接器，指显示接口硬件，例如VGA连接器，HDMI连接器等。

**EDID：**显示器识别数据，存储在显示器中的DDC存储器中，当电脑主机与显示器连接后，电脑主机会通过DDC通道读取显示器中存储的EDID。

### 3 架构关系

龙芯2K2000芯片中集成显示控制器(DC)和图形处理器(GPU)，龙芯显示驱动包括内核驱动和VBIOS固件，GPU固件。由内核驱动和VBIOS固件，GPU固件共同控制龙芯显示控制器和GPU，实现显示、背光调节、二三维渲染加速等功能。VBIOS固件需要由专门的工具制作，龙芯2K2000芯片内置GPU驱动初始化设备时，从显存中获取VBIOS固件，解析VBIOS固件中包含的设备属性表，设备配置，代码等内容。龙芯内置显卡驱动与VBIOS固件层次关系如图3-1所示：

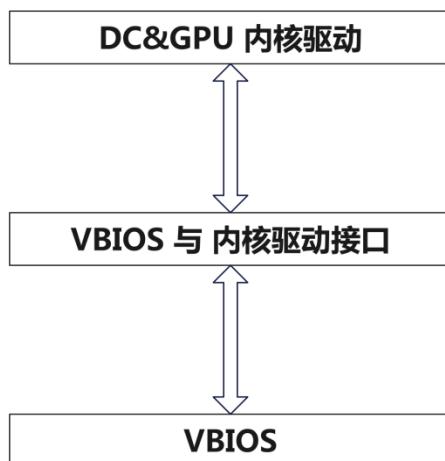


图3-1 显示驱动，固件与显示硬件的关系

## 4 2K2000芯片硬件显示管线设计规范

### 4.1 硬件显示管线简介

显示管线是显卡中重要的组成部分，显示管线负责扫描显存中的数据并将这些数据送到终端显示，图形管线由显示控制器，编码器，信号转换器，连接器，I2C 控制器等多种设备组成，显示管线的结构如图 4-1 所示，每个设备的功能见表4-1所示。

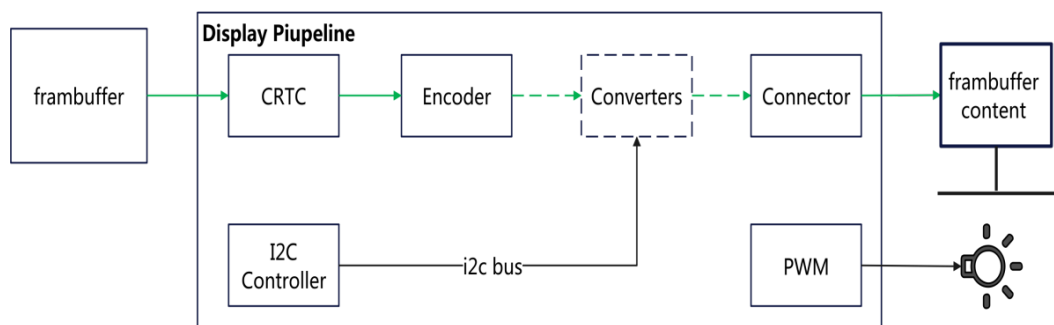


图4-1 显示管线

表4-1 设备名称功能

设备名称	功能描述
CRTC	显示控制器，对显示缓冲区进行扫描，并产生时序信号的硬件模块。
Encoder	编码器，将显示控制器产生的时序信号转换成外部设备所需要的信号。
Converter	信号转换器，将编码器产生的时序信号转换成外部设备所需要的信号，可选，可支持多个。
Connector	连接器，连接物理显示设备的连接器。



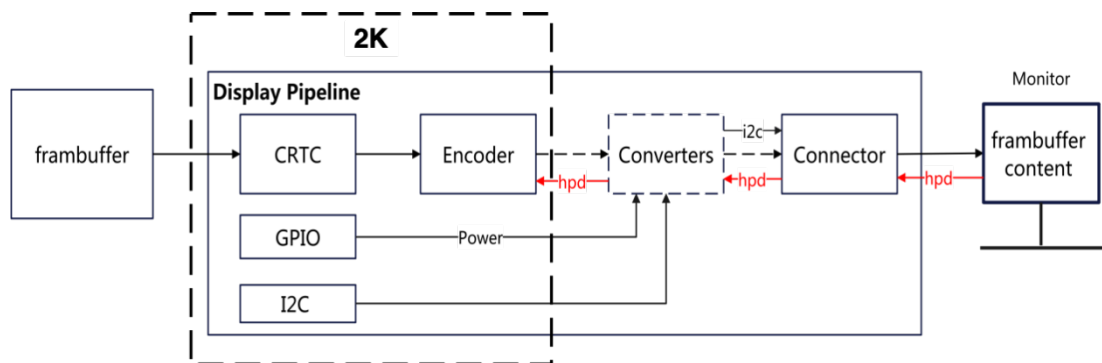


图4-3 有信号转换芯片的显示器显示管线

## 4.2.2 获取EDID

在显示器方案中，EDID由显示器提供。在不连接信号转换芯片时，通过显示控制器中I2C直接探测EDID，在连接信号转换芯片时，通过显示控制器的I2C向信号转换芯片发送探测指令，由显示转换芯片完成进一步的探测，并将EDID返回给驱动。驱动根据EDID中的信息设置显示控制器的输出时序。

## 4.2.3 热插拔探测

在设计显示器显示管线热插拔探测时，请按照如下的原则进行：

1. 直接使用内置VGA作为显示接口时，建议使用轮训方式进行热插拔探测，在硬件设计上务必保证具备探测到EDID，否则热插功能拔会失效。
2. 直接使用内置HDMI作为显示接口时，使用中断方式进行热插拔探测。
3. 使用外接转换芯片时，要求连接转换芯片的热插拔中断管脚。

## 4.3 液晶屏显示管线方案

### 4.3.1 液晶屏显示管线

本规范中约束了液晶屏硬件显示管线的实现方案，在液晶屏方案中，支持背光调节，不支持液晶屏热插拔功能。

**方案说明：**将2K2000作为信号转换芯片的主控时，硬件参考设计如图4-4，将显示控制器的I2C连接到转换芯片。用于实现对信号转换芯片的控制和读取EDID功能。使用芯片的GPIO管脚来控制转换视频转换芯片的电源，背光电源和液晶屏电源。

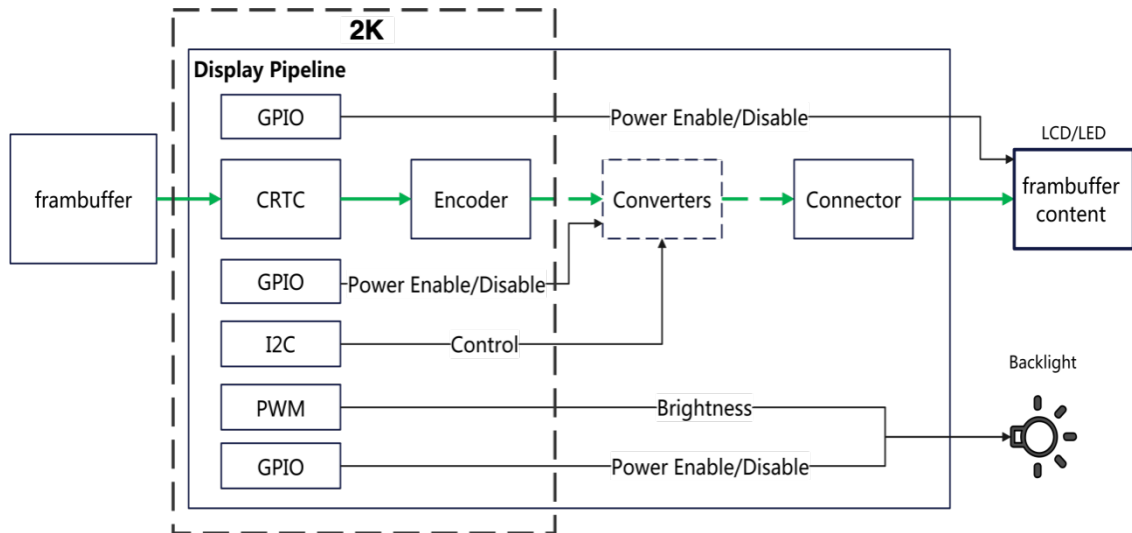


图4-4 液晶屏显示管线

### 4.3.1 获取EDID

在液晶屏方案中，EDID可由液晶屏或者显示转换芯片提供，在前两者无法提供的情况下也可由VBIOS提供。EDID用于提供显示控制器的输出时序参数。EDID由液晶屏或显示转换芯片提供时，通过2K2000芯片显示控制器中的I2C向信号转换芯片发送探测指令，这时可由显示转换芯片返回内部存储的EDID或者探测液晶屏中存储的EDID，并将EDID返回。如果EDID由VBIOS 提供，则由驱动解析VBIOS中的EDID数据段。

**需要特别注意的是：**EDID数据应该由液晶屏厂家提供，厂家在制作EDID时，EDID中的时序必须要符合液晶屏的时序要求，另外EDID中的features字段(第0x18字节)的第0位请勿写1，该位表示显示设备支持工作范围内的GTF时序，该位置1后驱动将根据EDID中提供的时序范围增加EDID中未列出的GTF时序，使用新增加的分辨率存在不可预测的结果不保证显示功能的正确性和稳定性。关于更多的EDID的信息，请参考《VESA ENHANCED EXTENDED DISPLAY IDENTIFICATION DATA STANDARD》

### 4.3.2 屏幕背光

本规范中约束了屏幕背光的实现方案，将芯片作为背光的主控时，由芯片上的GPIO和PWM分别控制背光电源和背光亮度。推荐使用芯片的GPIO 和 PWM 资源和状态，见表4-2。

表4-2 2K2000芯片GPIO ， PWM 资源使用表

功能	GPIO 资源与状态
屏幕电源控制	2K2000 PWM 05 (GPIO 09)，下拉关闭，上拉使能，默认下拉。
屏幕背光控制	2K2000 PWM 04 (GPIO 08)，下拉关闭，上拉使能，默认下拉。

背光亮度调节	2K2000 PWM 03 (GPIO07)。
信号转换芯片复位控制	2K2000 node GPIO 02。

## 4.5 显存设计约束

龙芯2K2000芯片支持使用系统内存作为显存，容量不小于256MB。

## 4.6 信号转换芯片选型约束

对于非透明信号转换芯片选型时需要满足如下的要求：

1. 具备两个I2C 控制器，作为从设备，用于与主控通信，作为主设备，用于热插拔探测，探测和获取EDID。
2. 在液晶屏方案下，如需支持多分辨率，要求信号转换芯片支持缩放功能，缩放后的显示质量由转换芯片决定。
3. 具备存储EDID 的功能(不强制)。
4. 具备低功耗模式。具备电源控制(不强制)。

## 4.7 多屏显示方案

龙芯2K2000芯片中有两个显示控制器，最多可支持双路显示，可由HDMI（显示通道0）和DVO（显示通道1）组成扩展显示模式。

### 4.7.1 单屏方案

在单屏方案中，可选择单独使用HDMI（显示通道0）或者 DVO（显示通道1）。

### 4.7.2 双屏方案

在双屏方案中，可使用HDMI与DVO组成双屏方案，两个屏幕可显示不同的内容，实现扩展显示功能。

## 5 2K2000 VBIOS设备属性标配

龙芯VBIOS提供一种结构化的数据，用于描述无法被GPU驱动程序动态探测的设备属性和设备连接拓扑，如图5-1所示，通过固件传参的形式将这些设备属性表传递给内核。这些设备可以在GPU内部或直接与GPU相连。关于VBIOS的更多信息，详见参考《龙芯VBIOS规范V2.0》文档，VBIOS生成工具的使用方法详见集成VBIOS 生成工具中的《龙芯VBIOS生成工具使用手



册》。

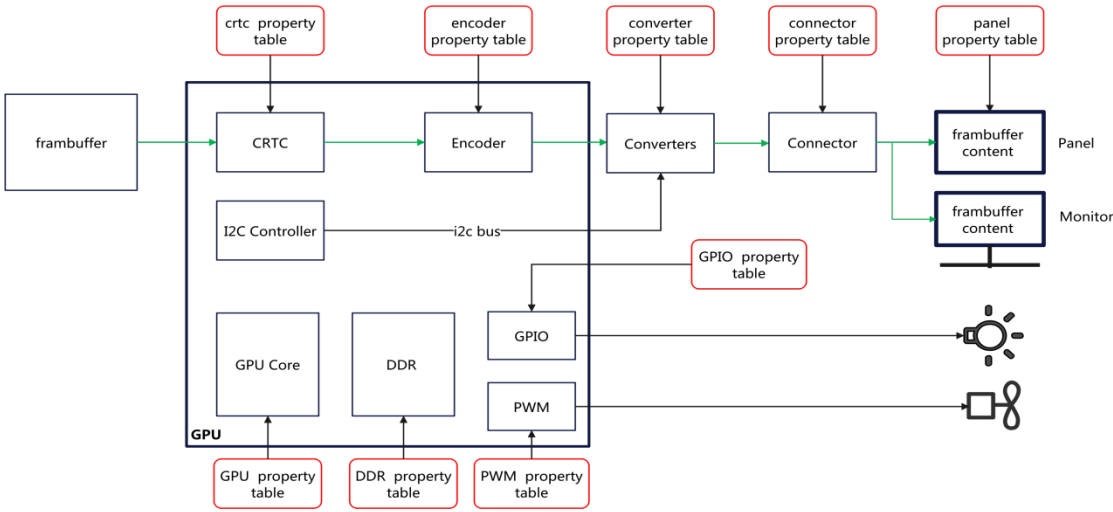


图 5-1 设备属性表示意图

5.1显示控制器属性表

表5-1 显示控制器属性表

属性名称	类型(大小)	功能说明
feature	u32(4B)	设备特性标记位。
crtc_id	u32(4B)	显示控制器硬件编号。编号从0开始。
encoder_id	u32(4B)	与该显示控制器相连的内置信号编码器编号。
max_freq	u32(4B)	控制器硬件支持的最大像素时钟频率，参考芯片手册填写。
max_width	u32(4B)	控制器硬件能支持的最大分辨率宽度，参考芯片手册填写。
max_height	u32(4B)	控制器硬件能支持的最大分辨率高度，参考芯片手册填写。
is_vb_timing	u8(4B)	是否由vbios提供显示控制器的时序，默认值为 0，已废弃，工具中不提供配置界面。

5.2内置编码器属性表

表5-2 信号转换器属性表

属性说明	类型(大小)	说明
feature	u32(4B)	设备特性标记位。
type	u32(4B)	内置编码器输出信号类型。

connector_id	u32(4B)	显示链路编号。
--------------	---------	---------

## 5.3 外接信号转换器属性表

表5-3 内置编码器属性表

属性名称	类型(大小)	功能说明
feature	u32(4B)	设备特性标记位。
i2c_id	u32(4B)	与转换芯片相连的i2c编号。默认值为 -1，已废弃。工具中不提供配置界面。
connector_id	u32(4B)	显示链路编号。
type	u32(4B)	外接编码器芯片输出信号的类型
config_type	u32(4B)	编码器芯片配置方式。
chip_id	u32(4B)	外接转换芯片的型号。
chip_addr	u32(4B)	外接转换芯片i2c从设备地址。

## 5.4 连接器属性表

表5-4 接口连接器属性表

属性名称	类型(大小)	功能说明
feature	u32(4B)	设备特性标记位。
i2c_id	u32(4B)	与连接器相连i2c编号。默认值为 -1，已废弃，工具中不提供配置界面。
internal_edid	u32(256B)	存放由vbios提供的edid。
type	u32(4B)	连接器接口类型。
hotplug	u32(4B)	热插拔探测方式的类型。
edid_method	u32(4B)	获取EDID方式。
irq_gpio	u32(4B)	用于热插拔探测的GPIO号。2K2000 以后不再使用。工具中不提供配置界面。
gpio_placement	u32(4B)	使用GPIO的位置，，默认值为 2K，已废弃，工具中不提供配置界面。

## 5.5 GPIO 设备属性表

表5-5 GPIO 设备属性表

属性名称	类型(大小)	说明
feature	u32(4B)	设备特性标记位
gpio_id	u32(4B)	使用的 gpio 编号
usage	u32(4B)	gpio 用途

## 5.6 PWM 设备属性表

表5-6 PWM设备属性表

属性名称	类型(大小)	说明
feature	u32(4B)	设备特性标记位。
pwm_id	u32(4B)	pwm 设备编号，参考主板进行配置。
polarity	u32(4B)	pwm 设备极性配置，参考主板进行配置。
peroid	u32(4B)	pwm 设备周期，单位Hz，参考芯片手册进行配置。
usage	u32(4B)	pwm 控制器的用途，参考主板进行配置。

## 5.7 GPU属性表

表5-7 GPU设备属性表

属性名称	类型(大小)	说明
feature	u32(4B)	设备特性标记位
gpu_freq_level	u32(4B)	GPU 频率调节档位，参考硬件配置。
gpu_freq	u32(4B)	GPU 最大频率，参考硬件配置。
num_gpu_shaders	u32(4B)	GPU 流处理器个数，参考硬件配置。

## 5.8显存属性表

表5-8 显存设备属性表

属性名称	类型(大小)	说明
------	--------	----

feature	u32(4B)	设备特性标记位。
vram_type	u32(4B)	显存颗粒类型。参考主板设计配置。
vram_bit_width	u32(4B)	显存颗粒位宽。参考主板设计配置。
vram_capacity	u32(4B)	显存容量。参考主板设计配置。
vram_freq	u32(4B)	显存频率。参考主板设计配置。

## 6 efifb

efifb适用于符合UEFI规范的固件。efifb既支持带有图形输出协议（GOP）显示器的固件。efifb可以复用固件启动时的分辨率和帧缓冲区地址。因此可以实现在操作系统没有显示驱动的情况下支持基本的图形显示功能。

龙芯BIOS中已默认集成支持GOP驱动，在操作系统中没有集成显卡驱动或不满足显卡驱动运行条件时，可继续由efifb完成图形显示功能。**要求固件传递帧缓冲地址，分辨率，色深必须和固件使用相同，才能保证进入系统后正常显示。**

系统使用efifb进入桌面后可正常操作应用程序，如开启浏览器、终端、文件管理等应用程序；efifb只能提供基础的图形显示功能，不支持画面旋转、切换分辨率、更改刷新率、显示器热插拔、显示器信息获取等功能。为使操作系统获得完整的使用体验，使用efifb启动后请尝试安装最新显卡驱动。

## 7 附录

### 7.1 显示器方案配置示例

如表7-1和表7-2所示，为使用内置HDMI直接输出显示器时vbios中设备信息的描述表。

表7-1 内置HDMI配置

设备	显示控制器	内置编码器	外接编码器	连接器
属性/配置	feature (0)	feature (0)	feature (0)	feature (0)
	crtc_id (0)	encoder_id (0)	i2c_id (0)	i2c_id (0)
	encoder_id (0)	connector_id (0)	connector_id (0)	internal_edid (不配置)
	max_freq (340000)		type (TMDS)	type (HDMI)
	max_width (4096)		config_type (driver)	hotplug (irq)
	max_height (4096)		chip_id (INTERNA_HDMI)	edid_method (I2C)

设备	显示控制器	内置编码器	外接编码器	连接器
			chip_addr (0)	

表7-2 GPU配置

设备	GPU	显存
属性/配置	feature (0)	feature (0)
	gpu_freq_level (3)	vram_type (ddr4)
	gpu_freq (500MHz)	vram_bit_width (32 bit)
	num_gpu_shaders (64)	vram_capacity (1GB)
		vram_freq (2400MHz)

## 7.2 液晶屏方案配置示例

如表7-3和表7-4所示，为使用外接eDP转换芯片输出液晶屏时vbios对各设备信息的描述。

表7-3 外接eDP配置

设备	显示控制器	内置编码器	外接编码器	连接器	PWM
属性/配置	feature (0)	feature (0)	feature (0)	feature (0)	feature (0)
	crtc_id (0)	encoder_id (0)	i2c_id (0)	i2c_id (0)	pwm_id (3)
	encoder_id (0)	connector_id (0)	connector_id (0)	internal_edid (不配置)	polarity (1000000)
	max_freq (340000)		type (eDP)	type (eDP)	peroid (1)
	max_width (4096)		config_type (driver)	hotplug (connected)	usage (0)
	max_height (4096)		chip_id (ncs8805)	edid_method (I2C)	
			chip_addr (0x3a)		

表7-4 GPU配置

设备	GPU	显存
属性/配置	feature (0)	feature (0)
	gpu_freq_level (3)	vram_type (ddr4)

	gpu_freq (500MHz)	vram_bit_width (32 bit)
	num_gpu_shaders (64)	vram_capacity (1 GB)
		vram_freq (2400MHz)

## 第七章 龙芯LA架构硬件设计规范

### 1 龙芯3A5000

为规范硬件设计和软件开发，龙芯3A5000相关板卡请遵循以下规则：

功能	设计规范	备注
UART	LS3A5000调试串口固定为UART0。无特殊情况，请不要随意更换。该串口亦可用作系统下普通串口功能。	
HT	LS3A5000芯片仅HT1可用于与桥片互联。	
I2C	固定I2C0作为调压以及读内存spd。	
看门狗	GPIO控制：定义GPIO10为看门狗使能，GPIO11为喂狗信号。	

### 2 龙芯3A6000

为规范硬件设计和软件开发，龙芯3A6000相关板卡请遵循以下规则：

功能	设计规范	备注
UART	LS3A6000调试串口固定为UART0。无特殊情况，请不要随意更换。该串口亦可用作系统下普通串口功能。	
看门狗	GPIO控制：定义GPIO10为看门狗使能，GPIO11为喂狗信号。	
I2C	固定I2C0作为调压以及读内存spd。若选用I2C作为调压芯片，约定设备地址为6BH。	
AVS	AVS调压地址：建议VDDN设置成 00h，VDDP 设置成01h 。	

### 3 龙芯3C5000

为规范硬件设计和软件开发，龙芯3C5000相关板卡请遵循以下规则：

功能	设计规范	备注
HT	LS3C5000芯片仅HT0可用于与桥片互联。	

功能	设计规范	备注
BMC	1、约定GPIO10作为固件post完成标记，高有效；GPIO15作为thermtrip功能。 针对使用2K0500 BMC的服务器，约定GPIO14作为监测BMC重启信号，低电平有效，需在2K0500 BMC 复位前约500ms（2K0500 BMC卡复位线路实测值）通知CPU。 2、约定UART0与BMC的UART0对接（作为SOL功能使用）。	
看门狗	GPIO控制：定义GPIO5为看门狗使能，高有效，GPIO6为喂狗信号，边沿触发。	
UART	LS3C5000调试串口固定为UART0。无特殊情况，请不要随意更换。该串口亦可用作系统下普通串口功能。	
I2C	固定I2C1作为电源调压，I2C0读内存spd。	3C5000 I2C说明见表1

表1 3C5000 I2C说明

3C5000内存通道	MC0	MC1	MC2	MC3
3C5000L内存通道	MC2	MC0	MC1	MC3
I2C通道	I2C0			
I2C地址	100, 101	000, 001	010, 011	110, 111

## 4 龙芯3D5000

为规范硬件设计和软件开发，龙芯3D5000相关板卡请遵循以下规则：

功能	设计规范	备注
HT	LS3D5000芯片仅HT0可用于与桥片互联。	
BMC	1、约定GPIO10作为固件post完成标记，高有效；GPIO15作为thermtrip功能。 针对使用2K0500 BMC的服务器，约定GPIO14作为监测BMC重启信号，低电平有效，需在2K0500 BMC 复位前约500ms（2K0500 BMC卡复位线路实测值）通知CPU。 2、约定UART0与BMC的UART0对接（作为SOL功能使用）。	
看门狗	GPIO控制：定义GPIO5为看门狗使能，高有效，GPIO6为喂狗信号，边沿触发。	
UART	LS3D5000调试串口固定为UART0。无特殊情况，请不要随意更换。该串口亦可用作系统下普通串口功能。	
I2C	固定DIE0-I2C0作为电源调压，DIE0-I2C1和DIE0-I2C2读内存spd。	3D5000 I2C说明见表2



表2 3D5000 I2C说明

内存通道	DIE0_MC0	DIE0_MC1	DIE1_MC0	DIE1_MC1	DIE0_MC2	DIE0_MC3	DIE1_MC2	DIE1_MC3
I2C通道	DIE0_I2C1				DIE0_I2C2			
I2C地址	000, 001	010, 011	100, 101	110, 111	000, 001	010, 011	100, 101	110, 111

## 5 龙芯7A1000

为规范硬件设计和软件开发，龙芯7A1000相关板卡请遵循以下规则：

功能	设计规范	备注
RTC	1、默认通用操作系统使用内部RTC功能模块。 2、外挂RTC需要选用《龙芯外围功能芯片支持列表》中的器件，目前仅支持计时功能，其他功能暂不支持。	
SPI	1、LS7A1000桥片的SPI总线推荐连接一个SPI flash，片选固定为CS0，读写方式必须完全兼容SST25VF010，该flash用于存储桥片集成GMAC的MAC地址、LS7A1000的GPU显示相关参数信息、主板串号等。 2、使用SPI的Tcm芯片需要连接到CS1。	
GPIO	1、约定使用GPIO0作为蜂鸣器控制，高电平有效。 2、使用SmartFan功能，约定PWM0作为处理器风扇控制，PWM1作为处理器风扇转速监测。	
显示	LS7A1000内置显卡的硬件设计要求、VBIOS固件的存放位置等，请参见《龙芯7A1000内置显卡软硬件设计规范》。	
I2C	如果使用I2C触摸板，约定GPIO50作为中断输入。	
BMC	连接BMC的PCIE，推荐优先使用主桥的F0_P1。	
PCIE	1、针对服务器双桥使用场景，副桥只支持PCIE。 2、针对服务器使用场景，接PCIe_Riser卡时，需要把控制器对应的PCIE复位控制信号接到相应的槽位上。以G0/G1/H控制器为例示意图见图1。	

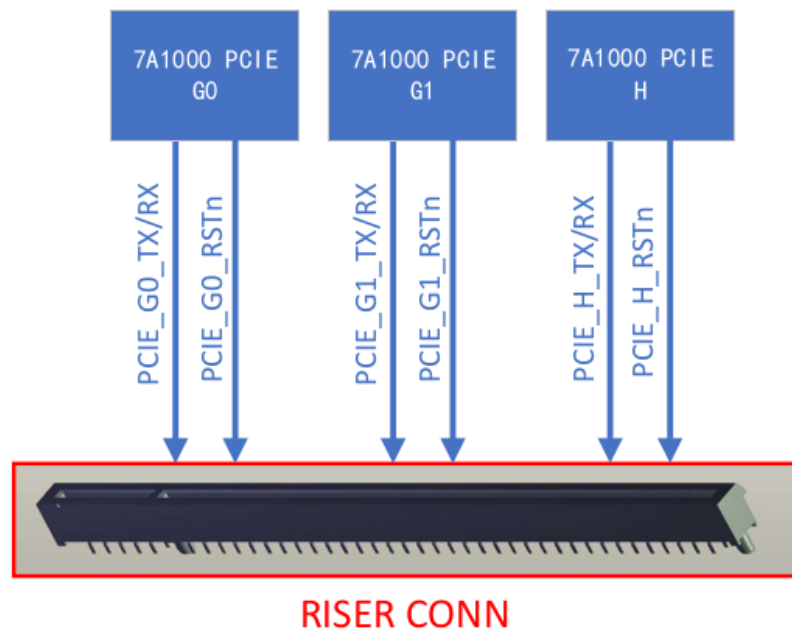


图1 LS7A1000 PCIe接Riser时连接示意图

## 6 龙芯7A2000

为规范硬件设计和软件开发，龙芯7A2000相关板卡请遵循以下规则：

功能	设计规范	备注
RTC	1、默认通用操作系统使用内部RTC功能模块。 2、外挂RTC需要选用《龙芯外围功能芯片支持列表》中的器件，目前仅支持计时功能，其他功能暂不支持。	
SPI	1、LS7A2000桥片的SPI总线推荐连接一个SPI flash，片选固定为CS0，读写方式必须完全兼容SST25VF010，该flash用于存储桥片集成GMAC的MAC地址、LS7A2000的GPU显示相关参数信息、主板串号等。 2、使用SPI的Tem芯片需要连接到CS1。	
GPIO	1、约定使用GPIO0作为蜂鸣器控制，低电平有效。 2、使用SmartFan功能，约定PWM0作为处理器风扇控制，PWM1作为处理器风扇转速监测。	
显示	LS7A2000内置显卡的硬件设计要求、VBIOS固件的存放位置等，请参见《龙芯7A2000内置显卡软硬件设计规范》。	
I2C	1、如果使用I2C触摸板，约定GPIO50作为中断输入。 2、I2C0：在服务器中I2C0作为SLAVE，用于BMC获取7A2000温度。	
BMC	连接BMC的PCIE，推荐使用主桥的F0_P1。对于2K0500 BMC，必须使用该端口。	

功能	设计规范	备注
PCIE	1、针对服务器双桥使用场景，副桥只支持PCIE，SATA，GNET，USB3。 2、针对服务器使用场景，接PCIE_Riser卡时，需要把控制器对应的PCIE复位控制信号接到相应的槽位上。以G0/H控制器为例示意图见图2。	

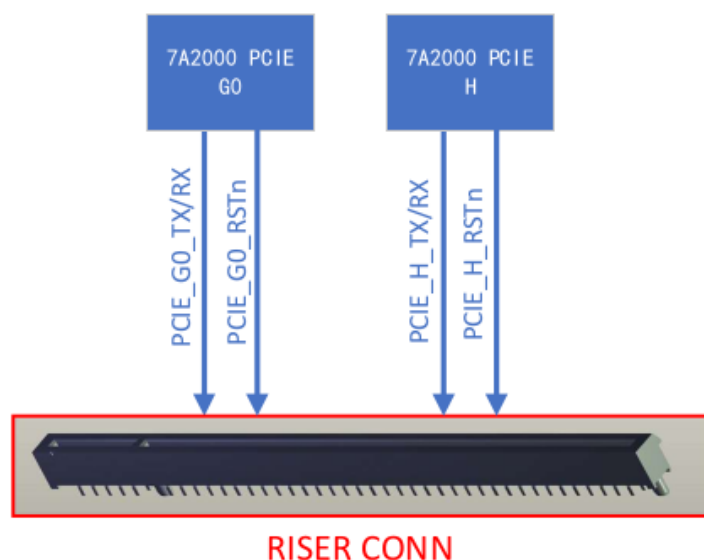


图2 LS7A2000 PCIE接Riser时连接示意图

## 7 龙芯2K2000

为规范硬件设计和软件开发，龙芯2K2000相关板卡请遵循以下规则：

功能	设计规范	备注
RTC	1、默认通用操作系统使用内部RTC功能模块。 2、外挂RTC需要选用《龙芯外围功能芯片支持列表》中的器件，目前仅支持计时功能，其他功能暂不支持。	
串口	约定ND_TXD0，ND_RXD0作为系统调试串口。	
显示	LS2K2000内置显卡的硬件设计要求、VBIOS固件的存放位置等，请参见《龙芯2K2000内置显卡软硬件设计规范》。	

## 8 龙芯3B6000

为规范硬件设计和软件开发，龙芯3B6000相关板卡请遵循以下规则：

功能	设计规范	备注
I2C	固定I2C0作为电源调压备份（调压默认使用AVS接口），I2C1读内存spd。	3B6000 I2C说明见下表

表3 3B6000 I2C说明

3B6000内存通道	MC2	MC3
I2C地址	010	011

## 9 龙芯3C6000/S

为规范硬件设计和软件开发，龙芯3C6000/S相关板卡请遵循以下规则：

功能	设计规范	备注
I2C	固定I2C0作为电源调压备份（调压默认使用AVS接口），I2C1读内存spd，I2C2配置成slave。	3C6000/S I2C说明见下表

表4 3C6000/S I2C说明

3C6000/S内存通道	MC0	MC1	MC2	MC3
I2C地址	000	001	010	011

## 10 龙芯3C6000/D

为规范硬件设计和软件开发，龙芯3C6000/D相关板卡请遵循以下规则：

功能	设计规范	备注
PCIe	CPU0 DIE0 PCIe0[7:0]用于与桥片互联。	

功能	设计规范	备注
内存	DIE0_MC0: DIMM_SPD_SA[2:0]=3' b000。	
	DIE0_MC1: DIMM_SPD_SA[2:0]=3' b001。	
	DIE0_MC2: DIMM_SPD_SA[2:0]=3' b010。	
	DIE0_MC3: DIMM_SPD_SA[2:0]=3' b011。	
	DIE1_MC0: DIMM_SPD_SA[2:0]=3' b000。	
	DIE1_MC1: DIMM_SPD_SA[2:0]=3' b001。	
	DIE1_MC2: DIMM_SPD_SA[2:0]=3' b010。	
	DIE1_MC3: DIMM_SPD_SA[2:0]=3' b011。	
SPI	CPU0 DIE0_SPI: 规定为BIOS启动接口。	
UART	CPU0 DIE0_UART0: 规定为系统调试串口，波特率115200，8N1。	
AVSBus	DIE0_AVSBus: 规定为VDDN/P电源调压通道。	
I2C	DIE0_I2C0: 规定为CPU VDDN/P电源PMBUS调压接口。	推荐优先使用AVSBus调压
	DIE0_I2C1: 规定为内存SPD I2C，归属DIE0_MC0/1，DIE1_MC2/3。	
	DIE0_I2C2: 规定为内存SPD I2C，归属DIE0_MC2/3，DIE1_MC0/1。	
	DIE1_I2C0: 默认为Slave模式，规定为CPU核温获取通道，用于BMC或者其它控制器获取CPU核温。I2C地址：0x1b(CPU0)，0x2b(CPU1)，0x1c(CPU2)，0x2c(CPU3)。	

功能	设计规范	备注
GPIO	DIE0_GPIO2: 复位同一个Group下的PCIe设备。一般推荐用来复位PCIe0/2。	PCIe0/2为同一个Group, PCIe1/3为同一个Group。
	DIE0_GPIO3: 复位同一个Group下的PCIe设备。一般推荐用来复位PCIe1/3。特例, 在双路设计中,用来复位CPU0 DIE0 PCIe2。	PCIe0/2为同一个Group, PCIe1/3为同一个Group。
	DIE0_GPIO10: 规定为BIOS Post Complete输出信号。	
	DIE0_GPIO11: 用于使能外部看门狗。	
	DIE0_GPIO14: 针对2K0500 BMC, 规定为监测BMC重启信号, 低电平有效, 需在2K0500 BMC复位前约500ms (2K0500 BMC卡复位线路实测值) 通知CPU。	仅针对CPU0
	DIE0_GPIO15: 复用为THERMTRIPn功能使用。	
	DIE1_GPIO2: 同DIE0。	
	DIE1_GPIO3: 同DIE0。	
	DIE1_GPIO12: 用于对外部看门狗喂狗信号。	
	DIE1_GPIO14: 复用为 PROCHOTn。	

## 11 龙芯3C6000/Q

为规范硬件设计和软件开发, 龙芯3C6000/Q相关板卡请遵循以下规则:

功能	设计规范	备注
PCIe	CPU0 DIE0 PCIe0[7:0]用于与桥片互联。	

功能	设计规范	备注
内存	DIE0_MC2: DIMM_SPD_SA[2:0]=3' b010。	
	DIE0_MC3: DIMM_SPD_SA[2:0]=3' b011。	
	DIE1_MC0: DIMM_SPD_SA[2:0]=3' b000。	
	DIE1_MC1: DIMM_SPD_SA[2:0]=3' b001。	
	DIE2_MC2: DIMM_SPD_SA[2:0]=3' b010。	
	DIE2_MC3: DIMM_SPD_SA[2:0]=3' b011。	
	DIE3_MC0: DIMM_SPD_SA[2:0]=3' b000。	
	DIE3_MC1: DIMM_SPD_SA[2:0]=3' b001。	
SPI	CPU0 DIE0_SPI: 规定为BIOS启动接口。	
UART	CPU0 DIE0_UART0: 规定为系统调试串口, 波特率115200, 8N1。	
AVSBus	DIE0_AVSBus: 规定为DIE0_3_VDDN/P电源调压通道。	
	DIE2_AVSBus: 规定为DIE1_2_VDDN/P电源调压通道。	
I2C	DIE0_I2C0: 规定为CPU VDDN/P电源PMBUS调压接口。	推荐优先使用AVSBus调压
	DIE0_I2C1: 规定为内存SPD I2C, 归属DIE0_MC2/3,DIE1_MC0/1。	
	DIE0_I2C2: 规定为内存SPD I2C, 归属DIE2_MC2/3,DIE3_MC0/1。	
	DIE1_I2C0: 规默认为Slave模式, 规定为CPU核温获取通道, 用于BMC或者其它控制器获取CPU核温。I2C地址: 0x1b(CPU0), 0x1c(CPU1) 。	

功能	设计规范	备注
GPIO	DIE0_GPIO2: 复位PCIe0下的设备。	
	DIE0_GPIO3: 复位PCIe3下的设备。	
	DIE0_GPIO10: 规定为BIOS Post Complete输出信号。	
	DIE0_GPIO11: 用于使能外部看门狗。	
	DIE0_GPIO14: 针对2K0500 BMC, 规定为监测BMC重启信号, 低电平有效, 需在2K0500 BMC复位前约500ms (2K0500 BMC卡复位线路实测值) 通知CPU。	仅针对CPU0
	DIE0_GPIO15: 复用为THERMTRIPn功能使用。	
	DIE1_GPIO2: 同DIE0。	
	DIE1_GPIO3: 同DIE0。	
	DIE2_GPIO2: 同DIE0。	
	DIE2_GPIO3: 同DIE0。	
	DIE3_GPIO2: 同DIE0。	
	DIE3_GPIO3: 同DIE0。	
	DIE3_GPIO12: 用于对外部看门狗喂狗信号。	
	DIE3_GPIO14: 复用为 PROCHOTn。	
	DIE3_GPIO15: 同DIE0。	



## 第八章 龙芯外围功能芯片支持列表

《龙芯外围功能芯片支持列表》

类型	厂商	型号	备注	已适配平台
千兆网卡	瑞昱 (Realtek)	RTL8111G/H		7A1000全系列、7A2000全系列
	网迅	WX1860AL-W		7A1000全系列、7A2000全系列
	沐创	沐创 N10L-X8		7A2000全系列
	Intel	82574	已停产	3A4000-7A1000产品
		I210/I211		7A1000全系列
		82580		7A1000全系列
		I350		7A1000全系列、7A2000全系列
	裕太微	YT6801/S		7A2000全系列
万兆网卡	网迅	WX1820AL		7A1000全系列、7A2000全系列
	沐创	N10G-X8		7A2000全系列
	Intel	82599		7A1000全系列、7A2000全系列
		X710		7A1000全系列、7A2000全系列
千兆网络PHY	Marvell	88E1510/88E1512		7A1000全系列、7A2000全系列
		88E1111-BAB11000		7A1000全系列
	瑞昱 (Realtek)	RTL8211E		7A1000全系列
		RTL8211FD		7A2000全系列
	裕太微	YT8521SC/H		7A1000全系列、7A2000全系列
		YT8531		7A1000全系列、7A2000全系列
		YT8511		7A1000全系列、7A2000全系列
	普维特电子	RPC101	无法关闭EEE模式	7A1000全系列
	Microchip	KSZ9031		7A1000全系列
	盛科	CTC2110II		7A2000全系列
	772s	B88E1111NYB		7A1000全系列

PCIe转USB3.0	瑞萨	UDP720201	四口	7A1000全系列、7A2000全系列
	钰创	EJ188H	四口	7A1000全系列、7A2000全系列
	祥硕	ASM1042	双口	7A1000全系列
PCIe转USB3.1	祥硕	ASM3142	双口	7A1000全系列
PCIe转SATA3.0	Mavell	9215		7A1000全系列
	祥硕	ASM1061		
	祥硕	ASM1064		
PCIe Switch	Broadcom	PEX8112		3A/B4000-7A1000产品
		PEX8619		
		PEX8632		
	祥硕	ASM8024		
PCIe转PCI Switch	Broadcom	PEX8112		3A/B4000-7A1000产品
	ITE	IT8893E/FX		7A2000全系列
PCIe ReDriver	万协通信	WSTR62		7A2000全系列
USB2.0 HUB	江苏沁恒	CH334U		3A5000-7A1000产品
	Terminus	FEL1s		7A1000全系列
SuperIO	Winbond	W83527	LPC接口	7A1000全系列
		W83795	I2C接口	7A1000全系列
	Fintek	F81866AD	LPC接口	7A1000全系列
	Nuvoton	NCT6106D	LPC接口	7A1000全系列、7A2000全系列
GPU	景嘉微	JM7201	PCIe x8	7A1000全系列
		JM9200	PCIe x8	7A1000全系列
	AMD	R5 340		7A1000全系列、7A2000全系列
		Radeon 520		7A1000全系列、7A2000全系列
		RX550		

DVO编码器	Chrontel	CH7055A	DVO转VGA	7A1000全系列
	Analog device	ADV7125	DVO转VGA	
	深圳晶格微电子	SDA7123	DVO转VGA	
	Analog device	ADV7513	DVO转HDMI	
	合肥宏晶微电子	MS7210	DVO转HDMI	
	Silicon image	SiI9022ACNU	DVO转HDMI	
	龙讯半导体	LT8618SXB	DVO转HDMI	
	TI	TFP410PAP	DVO转DVI	
	振芯科技	GM7510	DVO转DVI	
	新港海岸	NCS8805	DVO转EDP	
	振芯科技	GM8285C	DVO转LVDS	
	国腾	GM7123C	DVO转VGA	
	显示转换芯片	龙讯半导体	LT6711（不支持系统下修改分辨率）	
LT9721（不支持系统下修改分辨率）				
LT8619			HDMI转LVDS	
LT8612UX			HDMI转HDMI&VGA	
瑞昱（Reltek）		RTD2556T（支持系统下多分辨率）	HDMI转eDP，支持scalar	
MTK		MT9700FFFU（支持系统下多分辨率）		
HDA codec	前海深蕾	CX20632		7A1000全系列、7A2000全系列
		SN6140		7A1000全系列、7A2000全系列
	瑞昱（Reltek）	ALC897		7A1000全系列、7A2000全系列
		ALC269		7A1000全系列、7A2000全系列
		ALC662	已停产	7A1000全系列
		ALC663		7A2000全系列
I2S codec	顺芯	ES8388	只能有1个输入	7A2000全系列

CLock gen	IDT	6P4I505		7A1000全系列、7A2000全系列
	龙芯中科	8T4I505		7A1000全系列
	奥拉	Au5329		7A1000全系列、7A2000全系列
	IDT	5P49V6965A		3A5000-7A2000产品
	微宸半导体	MSCI42I		7A1000全系列
	IDT	9FGI08EFILFT		7A1000全系列
CLock buff	奥拉	Au54I1		7A1000全系列
	IDT	5PB1I04PGGI		3B4000、3B5000、3C5000L系列
	ON	NB3VI04CDTR2G		3B4000系列
UART	江苏沁恒	CH384L	PCIE转UART	7A1000全系列
		CH382		
		CH340/CH341/CH342/C H343/CH344	Usb转UART	
		CH9101/CH9102/CH9103		
		CH9344		
		CH348L		
RapidIO	IDT	TSI721	PCIE转Rapid io	7A1000全系列

DDR4颗粒	龙芯中科	LS6D4323-H/K	容量512MB, 位宽x8	3A4000全系列、3A5000全系列
		LS6D4423-H/K	容量512MB, 位宽x16	3A4000全系列、3A5000全系列
	Hynix	H5ANAG6NCJ-RXNC	容量2GB, 位宽x16	7A2000全系列
		H5AN4G6NBJR-VKC	容量512MB, 位宽x16	7A2000全系列
	Samsung	K4AAG165WA BCTD	容量2GB, 位宽x16	7A2000全系列
		K4A4G165WF BCTD	容量512MB, 位宽x16	7A2000全系列
		K4A8G165WC BCWE	容量1GB, 位宽x16	7A2000全系列
	紫光国芯	SCB12Q4G160AF-07Q		3A4000全系列
	兆易创新	GDQ2BFAA-CQ	容量512MB, 位宽x16	7A2000全系列
	合肥长鑫	CXDQ3BFAM-CG		3A4000全系列、7A2000全系列
		CXDQ3A8AM-CQ		3A4000全系列、3A5000全系列
		CXDQ3BFAM-IJ-A		3A5000全系列
DDR3颗粒	Samsung	K4B2G1646F-BYMA		7A1000全系列
		K4B2G1646Q-BCK0		
		K4BIG1646G-BCK0		
		K4BIG1646I-BCNB		
	晶豪科技 (ESMT)	M15T1G1664A		
	紫光国芯	HX115H4G160AF-I3K	已停产	
	得一微	SGA256M16A1BI-Z01		
		SGA128M16A1BI-Z02		
	龙芯中科	LS6D3323-H/K	容量512MB, 位宽x8	
		LS6D3423-H/K	容量512MB, 位宽x16	

232串口芯片	南京国博电子	WS3232ECN		7A1000全系列、7A2000全系列
		WS3243FCA		7A1000全系列、7A2000全系列
	英联	UM3232EEUE		3A4000、3A5000、3C5000L系列
	芯景科技	AT3232EUE+		3A5000系列
	Maxim	MAX3243EAI		3A4000、3A5000、3C5000L系列
		MAX3243CAI		
		MAX3232EUE+		3A5000系列
	Fintek	F81439		7A2000全系列

SPI Flash	兆易创新	GD25Q16ESIGR	16Mb 3.3V	7A1000系列、7A2000系列
		GD25Q64CSIG	64Mb 3.3V	3A4000系列、3A5000系列（需要加电平转换）
		GD25Q128CSIG	128Mb 3.3V	3A4000系列、3A5000系列（需要加电平转换）
		GD25LQ64ESI	64Mb 1.8V	3A5000系列、3A6000系列
		GD25LB64CWIGR	64Mb 1.8V	3A5000系列
	Winbond	W25Q64JVSIG		7A1000系列、7A2000系列
		W25Q32JVSIG	32Mb 3.3V	3A4000系列、3A5000系列（需要加电平转换）
		W25Q64JVSIG	64Mb 3.3V	3A4000系列、3A5000系列（需要加电平转换）
		W25Q64FVSIG	64Mb 3.3V	3A4000系列、3A5000系列（需要加电平转换）
	博雅	BY25Q32BSSIG	32Mb 3.3V	3A4000系列、3A5000系列（需要加电平转换）
		BY25Q64ASSIG	64Mb 3.3V	3A4000系列、3A5000系列（需要加电平转换）
		BY25Q128ASSIG	128Mb 3.3V	3A4000系列、3A5000系列（需要加电平转换）
		BY25Q32ALSIG	32Mb 1.8V	3A4000系列（IO电需要使用1.8V）
	武汉新芯	XM25QH32CHIG	32Mb 3.3V	3A4000系列、3A5000系列（需要加电平转换）
		XM25QH64CHIQ	64Mb 3.3V	3A4000系列、3A5000系列（需要加电平转换）
		XM25QH64AHIG	64Mb 3.3V	3A4000系列、3A5000系列（需要加电平转换）
		XM25QH128AHIG	128Mb 3.3V	3A4000系列、3A5000系列（需要加电平转换）
		XM25QH128CHIQ	128Mb 3.3V	3A4000系列、3A5000系列（需要加电平转换）
		XM25QU32CJIG	32Mb 1.8V	3A5000系列
		XM25QU64AHIG	64Mb 1.8V	3A5000系列

SPI Flash	武汉新芯	XM25QU64CHIQ	64Mb 1.8V	3A5000系列
		XM25LU64CHIQ	64Mb 1.8V	3A5000系列、3A6000系列
		XM25QU128CHIQ	128Mb 1.8V	3A5000系列
		XM25QU256CXIQ	256Mb 1.8V	3A5000系列、3A6000系列
	上海复旦微	FM25Q64	64Mb 3.3V	3A4000系列
		FM25W128	128Mb 1.8V/3.3V	3A6000系列
	合肥恒烁	ZB25VQ32ASIG	32Mb 3.3V	3A4000系列、3A5000系列（需要加电平转换）
		ZB25VQ64ASIG	64Mb 3.3V	3A4000系列、3A5000系列（需要加电平转换）
		ZB25VQ128ASIG	128Mb 3.3V	3A4000系列、3A5000系列（需要加电平转换）
		ZB25VQ64BSJG	64Mb 3.3V	3A5000系列（需要加电平转换）
		ZB25VQ128DSJG	128Mb 3.3V	3A5000系列（需要加电平转换）
		ZB25LQ64ASJG	64Mb 1.8V	3A5000系列
		ZB25LQ128CSJG	128Mb 1.8V	3A5000系列
	PUYA	PY25Q128LA-SUH-IR	128Mb 1.8V	3A5000系列、3A6000系列
		PY25Q256LC	256Mb 1.8V	3A6000系列
		PY25Q64LB-SUH-IT	64Mb 1.8V	3A6000系列
	芯天下	XT25Q64DSSIGT	64Mb 1.8V	3A5000系列
		XT25Q128DSSIGT	128Mb 1.8V	3A5000系列
板载SSD	安信物联	AXH-US064MTW		7A1000全系列
	三顿电子	SCUD128GMTWT		7A1000全系列
	鸿秦科技	HTUSMU064G-WM	64GB	7A1000全系列
	得一微	SGES060GB6PC04Q-I-PSG-AAR		7A1000全系列
	景嘉微	MWMMSS3-128CM-M		3A5000-7A1000产品
	安信达	AS619		7A1000全系列



测温芯片	申矽凌	CT75MR		7A1000全系列
	圣邦微	SGM452		7A1000全系列
	中科银河	GX21M15U		7A1000全系列
HT电平转换芯片	国微	SM0104E		3A/B5000/3C5000L/3C5000-7A1000产品
	润石科技	RS0104YQ		
	圣邦微	SGM4578YTQG20G/TR		
	英联	UM3208UK		
WIFI	瑞昱 ( Realtek )	RTL8821ce	PCIE设备	7A1000全系列、7A2000全系列
		RTL8822ce	PCIE设备	7A1000全系列、7A2000全系列
		RTL8192cu	USB设备	7A1000全系列、7A2000全系列
		RTL8188gu	USB设备	7A1000全系列、7A2000全系列
		RTL8812au	USB设备	7A1000全系列、7A2000全系列
		RTL8822bu	USB设备	7A1000全系列、7A2000全系列
		RTL8852BE-CG	PCIE设备	7A2000全系列
RTC外置芯片	武汉芯景	AT8339AT	I2C设备	7A1000全系列、7A2000全系列
	Maxim	DS1338Z	I2C设备	7A1000全系列、7A2000全系列
	上海贝岭	BL5372	I2C设备	7A1000全系列、7A2000全系列
	大普通信	INS5830B	I2C设备	7A1000全系列、7A2000全系列
	兴威帆电子	SD2068	I2C设备	7A1000全系列、7A2000全系列
	成都华微	HWD5372	I2C设备	7A1000全系列、7A2000全系列
HDMI redriver	Parade	PS8409A		7A2000全系列
CAN收发器	纳芯微	NSIP1042		7A2000全系列
EC	联阳 ( ITE )	IT8638E	LPC	7A2000全系列
		IT5570VG-I28/CX	LPC	
		IT8528	LPC	
LDO	龙芯中科	8P80103		7A1000全系列、7A2000全系列
		8P2050		
PMIC	龙芯中科	8P63602	四合一电源管理芯片	7A1000全系列、7A2000全系列

## 第九章 龙芯2K3000固件开发规范

### 1 设计约束

#### 1.1 Audio

##### 1.1.1 HDA

固件通过Verb Table机制实现不同板卡中对声卡的差异化设计。固件开发人员需要根据具体主板的声卡设计正确填写Verb Table。具体参考龙芯代码中关于Realtek的Verb Table配置实现。

固件需要根据芯片手册中引脚复用配置寄存器设置引脚为HDA工作模式（芯片配置寄存器偏移0x0440寄存器的24位为1，配置寄存器基址需要固件下进行配置，参考代码配置寄存器基址为：0x10010000）。

##### 1.1.2 I2S

固件需要根据芯片手册引脚复用配置寄存器设置引脚为I2S工作模式（芯片配置寄存器偏移0x0440寄存器的25位为1，配置寄存器基址需要固件下进行配置，参考代码配置寄存器基址为：0x10010000）。

#### 1.2 VBIOS

2K3000内部包含图形处理器（以下简称GPU），位于D6:F0；内核下GPU需要Video BIOS（以下简称VBIOS）的支撑才可以工作正常。

固件对VBIOS处理方式：将固件内默认VBIOS全部读取到内置显卡BAR2空间的最后1MB。

#### 1.3 中断

龙芯2K3000方案，IO设备中断控制器管理的中断可通过中断线或中断消息包路由至NODE节点，MSI中断可通过中断消息包路由至NODE节点，以上中断最终路由至处理器核。见图1-1所示。

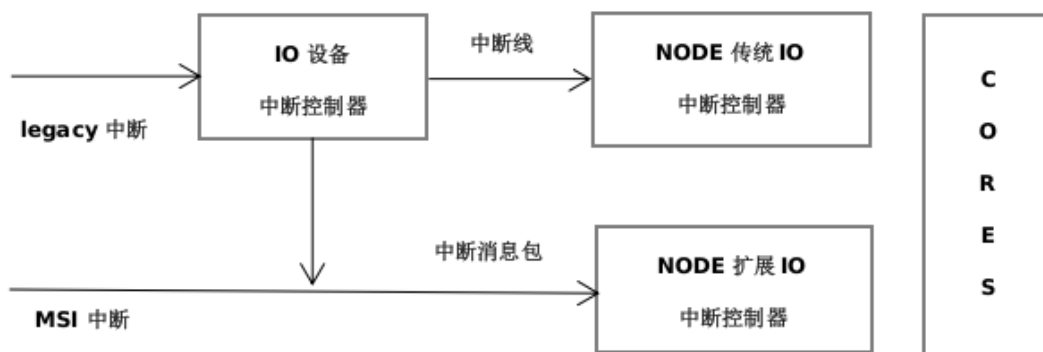


图1-1 龙芯2K3000处理器中断路由示意图

2K3000系统依据ACPI规范使用GSI（global system interrupts，全局系统中断）为中断源分配虚拟化的中断号，一个GSI类型的虚拟中断号唯一标识了其所属的中断域，内核按照不同的中断域实现中断管理。各个中断域对应的GSI中断号范围约定如下：

0-15：LPC/ISA中断域，用于系统外扩LPC设备中断源，如表1-1所示，具体分配情况参考 LPC设备说明。

表1-1 LPC/ISA中断域GSI分配

GSI中断号	中断源	说明
0	HPET	HPET高精度定时器
1	I8042	键盘
2	级联	
3		
4		
5		
6		
7		
8	RTC	实时时钟
9		
10		
11		
12	I8042	鼠标
13		
14	ide0	硬盘

15	Idle1	硬盘
----	-------	----

16-47: NODE传统I/O中断控制器中断域, 其GSI中断号与中断源向量号的映射关系: GSI中断号 = 中断源向量号+16 中断源向量号为处理器手册中某个中断源在NODE传统中断控制寄存器中的索引, 具体参考龙芯 2K3000处理器手册。

48-63: 保留。

64-127: I/O设备中断控制器中断域。其GSI中断号与中断源向量号的映射关系: GSI中断号 = 中断源向量号 + GSI base 中断源向量号为处理器手册中某个中断源在IO设备中断控制寄存器中的索引, 具体参考龙芯2K3000用户手册; GSI base为MADT表中当前BIO PIC结构的GSI base成员。其中IO设备集成或外接的PCI兼容设备分配的中断号如表1-2:

表1-2 I/O设备中断控制器中断域GSI分配

中断号	中断源	中断号	中断源
64	eMMC_DMA	96	pcie0_0
65	SDIO0_DMA	97	pcie0_1
66	SDIO1_DMA	98	pcie0_2
67	Resv	99	pcie0_3
68	RIO0	100	APB_DMA0
69	RIO1	101	APB_DMA1
70	VDEC	102	APB_DMA2
71	VENC	103	APB_DMA3
72	UART	104	PCM_DMA0
73	I2C	105	PCM_DMA1
74	Resv	106	PCM
75	CAN	107	Resv
76	gmac0_sbd	108	toy[0]
77	gmac0_pmt	109	toy[1]
78	gmac1_sbd	110	toy[2]
79	gmac1_pmt	111	acpi
80	sata	112	USB2
81	PCIe1_0	113	SDIO0
82	PCIe1_1	114	SDIO1

83	lpc	115	eMMC
84	hpet[1]	116	rte[0]
85	hpet[2]	117	rte[1]
86	usb3	118	rte[2]
87	hda1	119	hpet0
88	pwm[0]	120	i2s_dma[0]
89	pwm[1]	121	i2s_dma[1]
90	pwm[2]	122	i2s/hda0
91	pwm[3]	123	gpio_hi
92	dc	124	gpio0/gpio13
93	gpu	125	gpio1/gpio14
94	SPI1	126	gpio2/gpio15
95	thsens	127	gpio3/gpio50

注意：

(1) 若使用EC时需要使用SCI中断，请参考2K3000硬件规范，此中断需要硬件连接到 2K3000 电源管理模块的ACPI GPIO管脚，信号约束参照硬件设计规范，对应GSI中断源为111号。

## 1.4 地址空间

龙芯2K3000地址空间的分布参考处理器用户手册，表5-1芯片地址空间划分。

## 2 参考配置

### 2.1 MADT配置

表2-1 MADT表约束

域	大小 (字节)	偏移 (字节)	描述
Header			
Signature	4	0	‘APIC’
Revision	1	8	1

Local Interrupt Controller Address	4	36	0x1fe01400
------------------------------------	---	----	------------

表2-2 CORE PIC结构

域	大小 (字节)	偏移 (字节)	描述
Type	1	0	0x11
Length	1	1	15
Version	1	2	1
ACPI Processor ID	4	3	处理器核UID，与DSDT处理器对象中的_UID值相同
Physical Processor ID	4	7	CPU核物理ID
Flags	4	11	CORE PIC的标志，参考表2-3

表2-3 CORE PIC标志

CORE PIC Flags	大小 (比特)	偏移 (比特)	描述
Enabled	1	0	0: CPU不可用 1: CPU可用
Reserved	31	1	必须为0

MADT表的LIO PIC结构参考如下表2-4：

表 2-4 LIO PIC结构

域	大小 (字节)	偏移 (字节)	描述
Type	1	0	0x12, LIO PIC结构
Length	1	1	23
Version	1	2	1
Base Address	8	3	0x1FE01400
Size	2	11	0x80
Cascade vector	2	13	0x0002
Cascade vector mapping	8	15	0x0000000000FFFFFF

MADT表的EIO PIC结构参考如下表2-5:

表 2-5 EIO PIC结构

域	大小 (字节)	偏移 (字节)	描述
Type	1	0	0x14, EIO PIC结构
Length	1	1	4
Version	1	2	1
Cascade vector	1	3	3
Node	1	4	0
Node map	8	5	3

MADT表的MSI PIC结构参考如下表2-6:

表 2-6 MSI PIC结构

域	大小 (字节)	偏移 (字节)	描述
Type	1	0	0x15, MSI PIC结构
Length	1	1	19
Version	1	2	1
Message Address	8	3	0x2FF00000
Start	4	11	0x40
Count	4	15	0xc0

MADT表的BIO PIC结构参考如下表2-7:

表 2-7 BIO PIC结构

域	大小 (字节)	偏移 (字节)	描述
Type	1	0	0x16, BIO PIC结构
Length	1	1	17
Version	1	2	1
Base Address	8	3	0x00000E0010000000
Size	2	11	0x1000

Hardware ID	2	13	0
GSI base	2	15	0x40

MADT表的LPC PIC结构参考如下表2-8:

表 2-8 LPC PIC结构

域	大小 (字节)	偏移 (字节)	描述
Type	1	0	0x17, LPC PIC结构
Length	1	1	15
Version	1	2	1
Base Address	8	3	0x10002000
Size	2	11	0x1000
Cascade vector	2	13	0x13

## 2.2 FADT配置

此表为操作系统提供了Fixed硬件ACPI描述信息，相关约束见表2-9。

表2-9 FADT表约束

域	大小 (字节)	偏移 (字节)	描述
Header			
Signature	4	0	‘FACP’
Length	4	4	表的长度，包括表头，从偏移量0开始。用于记录整个表的大小
FADT Major Version	1	8	3
FIRMWARE_CTRL	4	36	0
DSDT	4	40	0
Reserved	1	44	0
SCL_INT	2	46	0x6f
SML_CMD	4	48	0
ACPI_ENABLE	1	52	0
ACPI_DISABLE	1	53	0



S4BIOS_REQ	1	54	0
PSTATE_CNT	1	55	0
PM1a_EVT_BLK	4	56	0
PM1b_EVT_BLK	4	60	0
PM1a_CNT_BLK	4	64	0
PM1b_CNT_BLK	4	68	0
PM2_CNT_BLK	4	72	0
PM_TMR_BLK	4	76	0
GPE0_BLK	4	80	0
GPE1_BLK	4	84	0
PM1_EVT_LEN	1	88	8
PM1_CNT_LEN	1	89	4
PM2_CNT_LEN	1	90	0
PM_TMR_LEN	1	91	4
GPE0_BLK_LEN	1	92	8
GPE1_BLK_LEN	1	93	0
GPE1_BASE	1	94	0
CST_CNT	1	95	0
P_LVL2_LAT	2	96	0x65
P_LVL3_LAT	2	98	0x3e9
FLUSH_SIZE	2	100	0
FLUSH_STRIDE	2	102	0
DUTY_OFFSET	1	104	0
DUTY_WIDTH	1	105	0
DAY_ALRM	1	106	0
MON_ALRM	1	107	0
CENTURY	1	108	0
IAPC_BOOT_ARCH	2	109	0
Reserved	1	111	0

Flags	4	112	默认值为0x25，包含如下标志： WBINVD、PROC_CI、SLP_BUTTON
RESET_REG	12	116	见表2-10
RESET_VALUE	1	128	0x01
ARM_BOOT_ARCH	2	129	0
FADT_Minor_Version	1	131	0
X_FIRMWARE_CTRL	8	132	FACS 64bit地址，固件动态生成
X_DSDT	8	140	DSDT 64bit地址，固件动态生成
X_PM1a_EVT_BLK	12	148	见表2-11
X_PM1b_EVT_BLK	12	160	此结构中成员全为0
X_PM1a_CNT_BLK	12	172	见表2-12
X_PM1b_CNT_BLK	12	184	此结构中成员全为0
X_PM2_CNT_BLK	12	196	此结构中成员全为0
X_PM_TMR_BLK	12	208	见表2-13
X_GPE0_BLK	12	220	见表2-14
X_GPE1_BLK	12	232	此结构中成员全为0

注：UEFI固件动态生成X\_FIRMWARE\_CTRL与X\_DSDT之前，需将其初始化为0。

表 2-10 reset寄存器结构

域	大小 (字节)	偏移 (字节)	描述
Address Space Id	1	0	0
Reg Bit Width	1	1	0x20
Reg Bit Offset	1	2	0
Reserved	1	3	0
Address	8	4	0x0000000010150030

表 2-11 PM1a EVT寄存器结构

域	大小 (字节)	偏移 (字节)	描述
Address Space Id	1	0	0

Reg Bit Width	1	1	0x40
Reg Bit Offset	1	2	0
Reserved	1	3	0
Address	8	4	0x000000001015000c

表 2-12 PM1a\_CNT寄存器结构

域	大小 (字节)	偏移 (字节)	描述
Address Space Id	1	0	0
Reg Bit Width	1	1	0x20
Reg Bit Offset	1	2	0
Reserved	1	3	0
Address	8	4	0x0000000010150014

表 2-13 PM\_TMR寄存器结构

域	大小 (字节)	偏移 (字节)	描述
Address Space Id	1	0	0
Reg Bit Width	1	1	0x20
Reg Bit Offset	1	2	0
Reserved	1	3	0
Address	8	4	0x0000000010150018

表 2-14 GPE0寄存器结构

域	大小 (字节)	偏移 (字节)	描述
Address Space Id	1	0	0
Reg Bit Width	1	1	0x40
Reg Bit Offset	1	2	0
Reserved	1	3	0
Address	8	4	0x0000000010150028

## 2.3 DSDT配置

### 2.3.1 PCI总线枚举

\_SEG（Segment）方法描述了PCI段号。

\_CRS（Current Resource Settings）方法支持总线范围、IO地址范围以及内存地址范围声明。

\_OSC（Operating System Capabilities）方法，当配置了\_OSC时，仅支持AER、PME和PCI Express Capability Structure control。

（1）配置参考如下：

\_SEG为0

\_CRS

使用WordBusNumber()宏声明PCI总线号范围，如表2-15：

表2-15 PCI总线号范围

参数名	值
ResourceUsage	ResourceProducer
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Decode	PosDecode
AddressGranularity	0x0
AddressMinimum	0x0
AddressMaximum	0x00FF
AddressTranslation	0x0000
RangeLength	0x0100
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空

使用QWordIO()宏声明PCI I/O地址范围，如表2-16：

表2-16 PCI I/O地址范围

参数名	值
ResourceUsage	ResourceProducer

IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Decode	PosDecode
ISARanges	EntireRange
AddressGranularity	0x00010000
AddressMinimum	0x0000000000000000
AddressMaximum	0x000000000000FFFF
AddressTranslation	0x0000000018000000
RangeLength	0x0000000000010000
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
TranslationType	值为空
TranslationDensity	值为空

使用DWordMemory()宏声明32位PCI MEM地址范围如表2-17:

表2-17 32位PCI MEM地址范围

参数名	值
ResourceUsage	ResourceProducer
Decode	PosDecode
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Cacheable	Cacheable
ReadAndWrite	ReadWrite
AddressGranularity	0x00010000
AddressMinimum	0x40000000
AddressMaximum	0x7FFFFFFF
AddressTranslation	0x00000000
RangeLength	0x40000000
ResourceSourceIndex	值为空

ResourceSource	值为空
DescriptorName	值为空
MemoryRangeType	值为空
TranslationType	值为空

使用QWordMemory()宏声明64位PCI MEM地址范围如表2-18:

表2-18 64位PCI MEM地址范围

参数名	值
ResourceUsage	ResourceProducer
Decode	PosDecode
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Cacheable	Cacheable
ReadAndWrite	ReadWrite
AddressGranularity	0x00000000000010000
AddressMinimum	0x0000004000000000
AddressMaximum	0x0000007FFFFFFFFF
AddressTranslation	0x0000000000000000
RangeLength	0x0000004000000000
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
MemoryRangeType	值为空
TranslationType	值为空

## 2.3.2 设备电源管理

2K3000系统支持USB2.0/USB3.0唤醒系统，需要配置相应的\_PRW。

### 2.3.2.1 USB

\_PRW (Power Resources for Wake)

表2-19 package对象参数表

参数名	值
EventInfo	设备号为D4:F0的USB控制器：值为0xa 设备号为D25:F0的USB控制器：值为0xb
DeepestSleepState	0x3

### 2.3.3 系统休眠唤醒

表2-20 SLP\_TYP规定

处理器	休眠状态	SLP_TYP
2K3000	S0	0
	S3	5
	S4	6

### 2.3.4 串口配置

\_CRS

表2-21串口参数表

参数名	值
ResourceUsage	ResourceConsumer
Decode	PosDecode
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Cacheable	NonCacheable
ReadAndWrite	ReadWrite
AddressGranularity	0x0
AddressMinimum	Node节点串口0值为：0x000000001FE001E0 南北桥串口0值为：0x0000000010100000
AddressMaximum	Node节点串口0值为：0x000000001FE001E7 南北桥串口0值为：0x00000000101000FF
AddressTranslation	0x0

RangeLength	Node节点串口0值为：0x0000000000000008 南北桥串口0值为：0x00000000000000100
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
MemoryRangeType	AddressRangeMemory
TranslationType	TypeStatic

中断号资源使用Interrupt()宏声明，各个参数的规定如表2-22：

表2-22 Interrupt()宏参数表

参数名	值
ResourceUsage	ResourceConsumer
EdgeLevel	Level
ActiveLevel	ActiveHigh
Shared	Shared
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
InterruptList	Node节点串口0值为：26 南北桥串口值为：72

\_DSD

表2-23 支持属性表

属性名	值	说明
clock-frequency	Node节点串口值为：100000000 南北桥串口值为：50000000	表示实际外接的串口时钟频率，单位为Hz

## 2.3.5 GPIO配置

\_HID

- (1) 2K3000南北桥集成的GPIO为LOON000B，2K3000南北桥集成的ACPI GPIO为LOON000C。



\_CRS

表2-24 QWordMemory()宏参数表

参数名	值
ResourceUsage	ResourceConsumer
Decode	PosDecode
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Cacheable	NonCacheable
ReadAndWrite	ReadWrite
AddressGranularity	0x0
AddressMinimum	南北桥GPIO值为: 0x0000000010160000 南北桥集成的ACPI GPIO值为: 0x0000000010150080
AddressMaximum	南北桥GPIO值为: 0x0000000010160EFF 南北桥集成的ACPI GPIO值为: 0x000000001015009F
AddressTranslation	0x0
RangeLength	南北桥GPIO值为: 0x0000000000000F00 南北桥集成的ACPI GPIO值为: 0x0000000000000020
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
MemoryRangeType	AddressRangeMemory
TranslationType	TypeStatic

每个中断号资源使用一个Interrupt()宏声明，各个参数的规定如表2-25:

表2-25 Interrupt()宏参数表

参数名	值
ResourceUsage	ResourceConsumer
EdgeLevel	Level
ActiveLevel	ActiveHigh
Shared	Shared

ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
InterruptList	南北桥GPIO: 123、124、125、126、127 南北桥集成的ACPI GPIO: 111

\_DSD

表2-26 支持属性如表

属性名	值	说明
ngpios	南北桥GPIO值为: 58 南北桥集成的ACPI GPIO: 16	当前注册的gpio设备包含的gpio pin总数

## 2.3.6 I2C配置

\_HID

(1) LOON0004。

\_UID

各I2C控制器与\_UID对应关系规定如表2-27:

表2-27 I2C控制器与\_UID对应表

控制器	_UID
I2C控制器0	0
I2C控制器1	1
I2C控制器2	2
I2C控制器3	3

\_CRS

表2-28 QWordMemory()宏参数表

参数名	值
ResourceUsage	ResourceConsumer
Decode	PosDecode

IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Cacheable	NonCacheable
ReadAndWrite	ReadWrite
AddressGranularity	0x0
AddressMinimum	I2C控制器值为: I2C 0: 0x000000001fe00120 I2C 1: 0x000000001fe00130 I2C 2: 0x0000000010110000 I2C 3: 0x0000000010110100
AddressMaximum	I2C控制器值为: I2C 0: 0x000000001fe00127 I2C 1: 0x000000001fe00137 I2C 2: 0x0000000010110007 I2C 3: 0x0000000010110107
AddressTranslation	0x0
RangeLength	I2C控制器值为: I2C 0: 0x0000000000000008 I2C 1: 0x0000000000000008 I2C 2: 0x0000000000000008 I2C 3: 0x0000000000000008
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
MemoryRangeType	AddressRangeMemory
TranslationType	TypeStatic

## 2.3.7 RTC配置

\_HID

LOON0001。

\_CRS

表2-29 QWordMemory()宏参数表

参数名	值
ResourceUsage	ResourceConsumer
Decode	PosDecode
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Cacheable	NonCacheable
ReadAndWrite	ReadWrite
AddressGranularity	0x0
AddressMinimum	0x0000000010150100
AddressMaximum	0x00000000101501FF
AddressTranslation	0x0
RangeLength	0x0000000000000100
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
MemoryRangeType	AddressRangeMemory
TranslationType	TypeStatic

中断号资源使用Interrupt()宏声明，各个参数的规定如表2-30：

表2-30 Interrupt()宏参数表

参数名	值
ResourceUsage	ResourceConsumer
EdgeLevel	Level
ActiveLevel	ActiveHigh
Shared	Exclusive
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
InterruptList	116

## 2.3.8 PWM配置

\_HID

(1) LOON0006。

\_CRS

表2-31 QWordMemory()宏参数表

参数名	值
ResourceUsage	ResourceConsumer
Decode	PosDecode
IsMinFixed	MinFixed
IsMaxFixed	MaxFixed
Cacheable	NonCacheable
ReadAndWrite	ReadWrite
AddressGranularity	0x0
AddressMinimum	PWM 0: 0x0000000010120000 PWM 1: 0x0000000010120100 PWM 2: 0x0000000010120200 PWM 3: 0x0000000010120300
AddressMaximum	PWM 0: 0x000000001012000F PWM 1: 0x000000001012010F PWM 2: 0x000000001012020F PWM 3: 0x000000001012030F
AddressTranslation	0x0
RangeLength	PWM 0: 0x0000000000000010 PWM 1: 0x0000000000000010 PWM 2: 0x0000000000000010 PWM 3: 0x0000000000000010
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
MemoryRangeType	AddressRangeMemory
TranslationType	TypeStatic

中断号资源使用Interrupt()宏声明，各个参数的规定如表2-32：

表2-32 Interrupt()宏参数表

参数名	值
ResourceUsage	ResourceConsumer
EdgeLevel	Level
ActiveLevel	ActiveHigh
Shared	Shared
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
InterruptList	PWM 0: 88 PWM 1: 89 PWM 2: 90 PWM 3: 91

## 2.3.9 热区管理

当配置龙芯2K3000处理器温度传感器时，可以参考如下部分示例：

```
Scope (\_SB)
{
    Name (IDDR,0x1fe00000) //ChipReg

    OperationRegion (BASE, SystemMemory, IDDR, 0x438)
    Field (BASE, AnyAcc, NoLock, Preserve)
    {
        Offset (0x428), //thsens
        THSE, 32
    }
    .....
}

Scope (\_TZ)
{
    ThermalZone (THM0)
    {
        Method (_TMP, 0, NotSerialized) // _TMP: Temperature
```

```
{
    Store (\_SB.THSE, Local0)
    Return (C2K (Local0))
}
Method (_CRT, 0, NotSerialized)  // _CRT: Critical Temperature
{
    Return (C2K (0x60))
}
.....
}

Method (C2K, 1, NotSerialized)  //Celsius to Kelvin
{
    Add (Multiply (Arg0, 0x0A), 0x0AAC, Local0)
    If (LLessEqual (Local0, 0x0AAC))
    {
        Store (0x0BB8, Local0)
    }

    If (LGreater (Local0, 0x0FAC))
    {
        Store (0x0BB8, Local0)
    }

    Return (Local0)
}
}
```

## 2.3.10 龙芯SE设备配置

\_HID

(1) LOON0011。

\_CRS

中断号资源使用Interrupt()宏声明，各个参数的规定如表2-33：

表2-33 Interrupt()宏参数表

参数名	值
ResourceUsage	ResourceConsumer

EdgeLevel	Level
ActiveLevel	ActiveHigh
Shared	Exclusive
ResourceSourceIndex	值为空
ResourceSource	值为空
DescriptorName	值为空
InterruptList	32、33、34、35、36、37、38、39

## 2.3.11 EC配置

\_GPE

(1) SCI中断对应的ACPI GPIO管脚位

## 2.4 FACS配置

表2-34 FACS表约束

域	大小 (字节)	偏移 (字节)	描述
Signature	4	0	‘FACS’
Firmware Waking Vector	4	12	0
Global Lock	4	16	0
Flags	4	20	0
X Firmware Waking Vector	8	24	0
Version	1	32	1

## 2.5 S3休眠地址

SADR，是系统S3休眠时进入固件执行休眠的入口地址，该地址由固件决定，存放固件的S3处理程序。

## 2.6 SRAT配置

表2-35 SRAT表约束



域	大小 (字节)	偏移 (字节)	描述
Header			
Signature	4	0	‘SRAT’
Revision	1	8	2

表2-36 Processor Local APIC/SAPIC Affinity结构

域	大小 (字节)	偏移 (字节)	描述
Type	1	0	0, 代表结构为Processor Local APIC/SAPIC Affinity Structure。
Length	1	1	16
Proximity Domain [7:0]	1	2	0, 代表处理器节点号的0-7位
APIC ID	1	3	处理器Local APIC ID, 见MADT表
Flags	4	4	标志。  0位: 0表示该Processor Local APIC/SAPIC Affinity Structure不可用; 1表示该Processor Local APIC/SAPIC Affinity Structure可用。  [31:1]位: 必须为0
Local SAPIC EID	1	8	0
Proximity Domain [31:8]	3	9	0, 代表处理器节点号的8-31位
Clock Domain	4	12	0

表2-37 Memory Affinity结构

域	大小 (字节)	偏移 (字节)	描述
Type	1	0	1, 代表结构为 Memory Affinity Structure
Length	1	1	40
Proximity Domain	4	2	0
Reserved	2	6	保留
Base Address Low	4	8	内存范围地址的低32位

Base Address High	4	12	内存范围地址的高32位
Length Low	4	16	内存范围大小的低32位
Length High	4	20	内存范围大小的高32位
Reserved	4	24	保留
Flags	4	28	标志，指示内存区域是否已启用并可以热插拔，见表2-38
Reserved	8	32	保留

表 2-38内存热插拔标志

域	大小 ( 比特 )	偏移 ( 比特 )	描述
Enabled	1	0	0: 该Memory Affinity Structure不可用 1: 该Memory Affinity Structure可用
Hot Pluggable	1	1	0
NonVolatile	1	2	0
Reserved	29	3	0

## 2.7 MCFG配置

MCFG表的配置如表2-39、2-40所示。

表2-39 MCFG表约束

域	大小 ( 字节 )	偏移 ( 字节 )	描述
Header			
Signature	4	0	‘MCFG’
Revision	1	8	1
OEMID	6	10	‘LOONGS’
OEM Table ID	8	16	‘LOONGSON’
OEM Revision	4	24	1

表2-40 Memory Mapped Enhanced Configuration Space Base Address Allocation  
Structure[0]结构

域	大小 (字节)	偏移 (字节)	描述
Base Address	8	0	0x000000fe00000000
PCI Segment Group Number	2	8	0
Start Bus Number	1	10	0x00
End Bus Number	1	11	0xFF
Reserved	4	12	0

## 2.8 SPCR配置

表2-41 SPCR表约束

域	大小 (字节)	偏移 (字节)	描述
Signature	4	0	‘SPCR’
Interface Type	1	36	0
Reserved	3	37	0
Space ID	1	40	0
Bit Width	1	41	0
Bit Offset	1	42	0
Encoded Access Width	1	43	1
Address	8	44	0x000000001FE001E0
Interrupt Type	1	52	0
PCAT-compatible IRQ	1	53	0
Interrupt	4	54	0
Baud Rate	1	58	0
Parity	1	59	0
Stop Bits	1	60	0
Flow Control	1	61	0
Terminal Type	1	62	0
Language	1	63	0

PCI Device ID	2	64	0xFFFF
PCI Vendor ID	2	66	0xFFFF
PCI Bus	1	68	0
PCI Device	1	69	0
PCI Function	1	70	0
PCI Flags	4	71	0
PCI Segment	1	75	0
Reserved	4	76	0

## 第十章 龙芯2K3000内置显卡软硬件设计规范

### 1 范围

本规范详细介绍龙芯2K3000芯片内置显卡的硬件设计约束和VBIOS设备属性表。本规范适用于使用龙芯2K3000芯片的主板。请其它系统厂商遵循此规范开发相关产品。

### 2 术语与定义

本规范所用术语定义如下：

**固件：**Firmware，写入ROM、EPROM等非易失存储器中的程序，负责控制和协调集成电路。

**BIOS：**基本输入输出系统，Basic Input Output System，一组固化到主板上一个ROM芯片上的程序，它保存着计算机基本输入输出程序、系统设置信息、开机后自检程序和系统自启动程序。BIOS与硬件系统集成在一起，也被称为固件，本规范中固件和BIOS不做区分。

**UEFI：**统一可扩展固件接口，Unified Extensible Firmware Interface，是UEFI Forum推广和发展的全新类型PC固件的体系结构、接口和服务的建议标准。主要目的是提供在OS加载之前在所有平台上一致、正确指定的启动服务。

**PMON：**MIPS架构机器上使用的一种兼有BIOS和boot loader部分功能的开放源码软件。

**VBIOS(Video BIOS)：**VBIOS是显卡的BIOS，VBIOS提供一些和显示相关的功能，并存放显示芯片与驱动程序之间的控制程序，另外还存放有显示卡型号、规格、生产厂家、出厂时间等信息。

**PCI (Peripheral Component Interconnect)：**是连接电子计算机主板和外部设备的总线标准，用于定义局部总线的标准。此标准允许在计算机内安装多达10个遵从PCI标准的扩展卡。

**Encoder：**型号编码器，用于将DVO型号转换为其他的显示信号，如模拟型号、LVTM或TMDS等显示信号。

**CRTC：**显示控制器(DC)在驱动中的抽象，显示控制器把数据从显存中将要显示数据取出来，然后对这些数据做一定的处理送到显示屏。

**Connector：**显示接口连接器，指显示接口硬件，例如DP连接器，HDMI连接器等。

**EDID：**显示器识别数据，存储在显示器中的DDC存储器中，当电脑主机与显示器连接后，电脑主机将通过DDC通道或AUX通道读取显示器中存储的EDID。

### 3 架构关系

龙芯2K3000芯片中集成显示控制器(DC)和图形处理器(GPU)，龙芯显示驱动包括内核驱动和VBIOS固件，GPU固件。由内核驱动和VBIOS固件，GPU固件共同控制龙芯显示控制器和GPU，实现显示、背光调节、二三维渲染加速等功能。VBIOS固件需要由专门的工具制作，龙芯2K3000芯片内置GPU驱动初始化设备时，从显存中获取VBIOS固件，解析VBIOS固件中包含的设备属性表，设备配置，代码等内容。龙芯内置显卡驱动与VBIOS固件层次关系如图3-1所示：

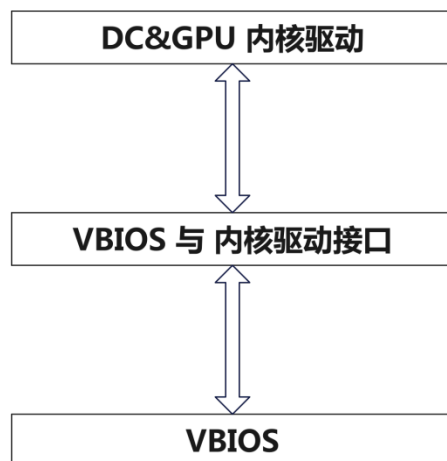


图3-1 显示驱动，固件与显示硬件的关系

## 4 2K3000芯片硬件显示管线设计规范

### 4.1 硬件显示管线简介

显示管线是显卡中重要的组成部分，显示管线负责扫描显存中的数据并将这些数据送到终端显示，显示管线由显示控制器，编码器，连接器，I2C 控制器等多种设备组成，显示管线的结构如图 4-1 所示， 每个设备的功能见表4-1所示。

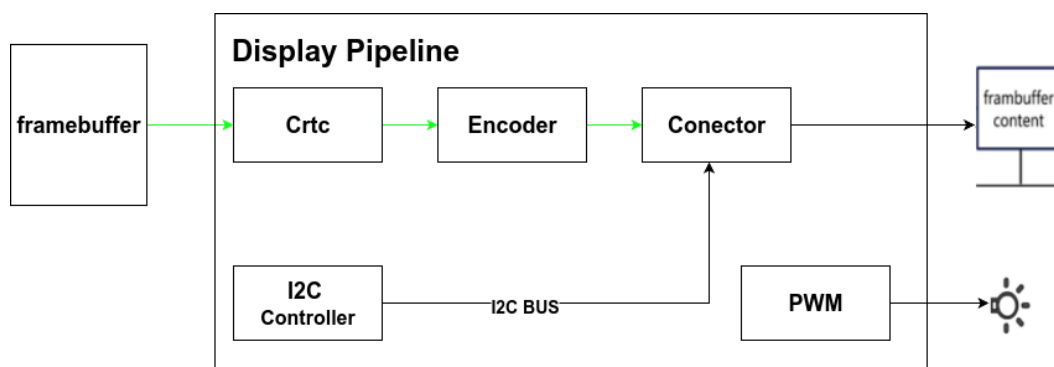


图4-1 显示管线

表4-1 设备名称功能

设备名称	功能描述
CRTC	显示控制器，对显示缓冲区进行扫描，并产生时序信号的硬件模块。
Encoder	编码器，将显示控制器产生的时序信号转换成外部设备所需要的信号。
Connector	连接器，连接物理显示设备的连接器。
I2C	I2C 控制器， 用于控制外部的信号转换器或者通过连接器读取EDID。

PWM	PWM 控制器，用于控制外部背光。
VRAM	显存

本规范约束搭载2K3000芯片主板的硬件显示管线设计。主板设计时需要符合本规范中的设计约束，保证显示驱动软件的兼容性。

## 4.2 显示器显示管线方案

### 4.2.1 显示器显示管线

本规范中定义了一种显示器方案，在设计硬件设计约束示意图如图4-2，要求将2K3000芯片显示控制器的I2C直接连接到显示接口连接器上，用于实现HDMI显示接口读取EDID。

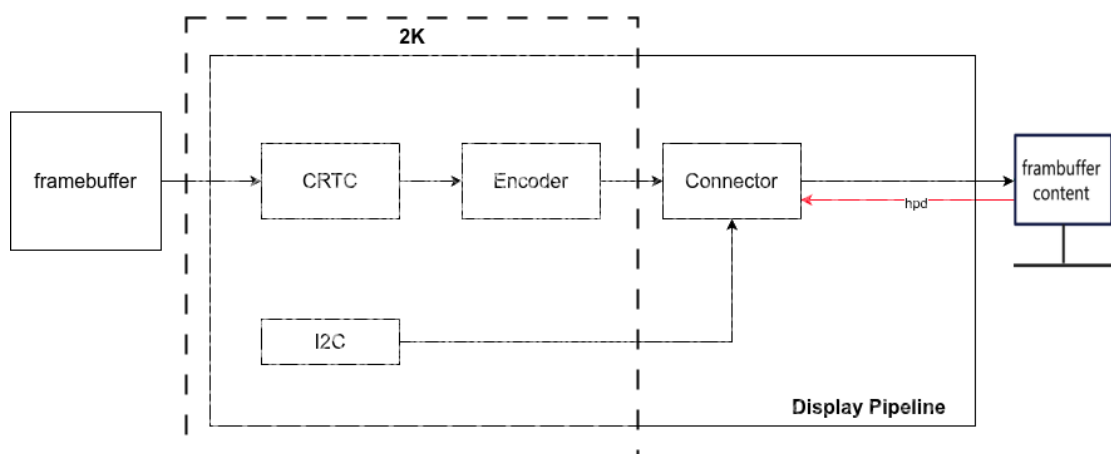


图4-2 显示器显示管线

### 4.2.2 获取EDID

在显示器方案中，EDID由显示器提供。当内置HDMI作为显示接口时，通过显示控制器中I2C直接探测EDID，当内置DP/EDP作为显示接口时，通过AUX通道读取EDID，并将EDID返回。驱动根据EDID中的信息设置显示控制器的输出时序。

### 4.2.3 热插拔探测

在设计显示器显示管线热插拔探测时，请按照如下的原则进行：

1. 直接使用内置HDMI作为显示接口时，使用中断方式进行热插拔探测。
2. 直接使用内置DP/EDP作为显示接口时，使用中断方式进行热插拔探测。

### 4.3 液晶屏显示管线方案

### 4.3.1 液晶屏显示管线

本规范中约束了液晶屏硬件显示管线的实现方案，在液晶屏方案中，支持背光调节，不支持液晶屏热插拔功能。

**方案说明:** 将2K3000作为主控时, 硬件参考设计如图4-3。EDP显示接口的AUX通道用于实现读取EDID功能, 使用芯片的GPIO管脚来控制背光电源和液晶屏电源。

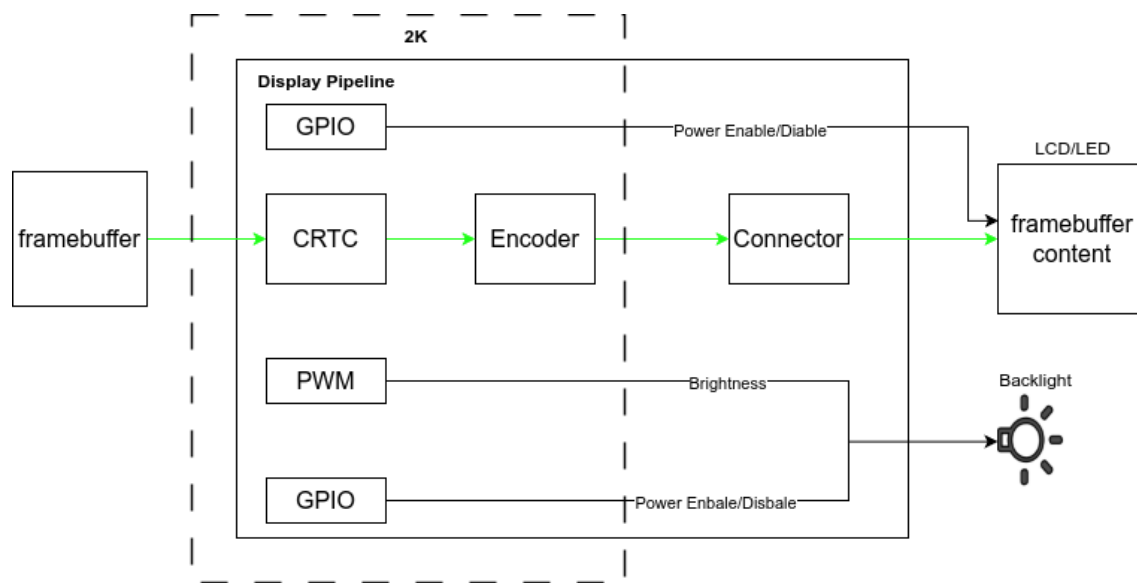


图4-3 液晶屏显示管线

### 4.3.2 获取EDID

在液晶屏方案中，EDID由液晶屏提供，EDID用于提供显示控制器的输出时序参数。通过2K3000芯片显示控制器中的AUX通道发送探测请求，AUX通道探测液晶屏中存储的EDID，并将EDID返回。

**需要特别注意的是：**EDID数据应该由液晶屏厂家提供，厂家在制作EDID时，EDID中的时序必须符合要求。液晶显示屏的时序要求，另外EDID中的features字段(第0x18字节)的第0位请勿写1，该位表示显示设备支持工作范围内的GTF时序，该位置1后驱动将根据EDID中提供的时序范围增加EDID中未列出的GTF时序，使用新增加的分辨率存在不可预测的结果不保证显示功能的正确性和稳定性。关于更多的EDID的信息，请参考《VESA ENHANCED EXTENDED DISPLAY IDENTIFICATION DATA STANDARD》。

## 4.4 屏幕背光

本规范中约束了屏幕背光的实现方案，将芯片作为背光的主控时，由芯片上的GPIO



和PWM分别控制背光电源和背光亮度。推荐使用芯片的GPIO和PWM资源和状态，见表4-2。

表4-2 2K3000芯片GPIO, PWM资源使用表

功能	GPIO和PWM资源与状态
屏幕电源控制	2K3000 GPIO 47, 下拉关闭, 上拉使能, 默认下拉。
屏幕背光控制	2K3000 GPIO 46, 下拉关闭, 上拉使能, 默认下拉。
背光亮度调节	2K3000 PWM 02。

## 4.5 显存设计约束

龙芯2K3000芯片支持使用系统内存作为显存，容量不小于512MB。

## 4.6 多屏显示方案

龙芯2K3000芯片中有两个显示控制器，最多可支持三路显示。一路HDMI接口显示，一路EDP接口显示，一路DP接口显示。可由EDP（显示通道0）和 HDMI/DP（显示通道1）组成扩展显示模式。

### 4.6.1 单屏方案

在单屏方案中，可选择单独使用EDP（显示通道0）或者 HDMI/DP（显示通道1）。

### 4.6.2 双屏方案

在双屏方案中，可使用HDMI与EDP组成双屏方案，也可以使用DP与EDP组成双屏方案。两个屏幕可显示不同的内容，实现扩展显示功能。

### 4.6.3 三屏方案

在三屏方案，可使用EDP（显示通道0），使用DP与HDMI作为（显示通道1）同源输出组成三屏方案。DP/EDP通过AUX通道读取EDID信息，HDMI通过I2C读取EDID信息。当显示通道1同时连接DP和HDMI显示器时，GPU驱动程序会对DP和HDMI的分辨率进行取交集处理，保证三屏方案正常显示。

## 5 2K3000 VBIOS设备属性标配

龙芯VBIOS提供一种结构化的数据，用于描述无法被GPU驱动程序动态探测的设备属性和设备连接拓扑，如图5-1所示，通过固件传参的形式将这些设备属性表传递给内核。这些设备可以

在GPU内部或直接与GPU相连。关于VBIOS的更多信息，详见参考《龙芯VBIOS规范V2.0》文档，VBIOS生成工具的使用方法详见集成VBIOS生成工具中的《龙芯VBIOS生成工具使用手册》。

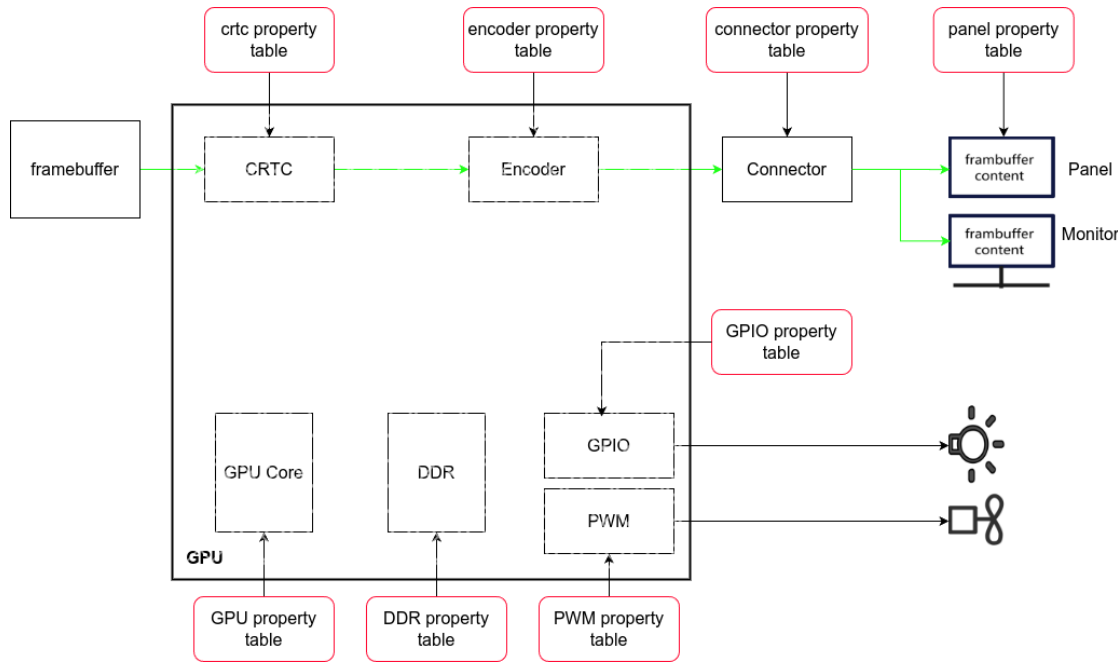


图 5-1 设备属性表示意图

## 5.1 显示控制器属性表

表5-1 显示控制器属性表

属性名称	类型(大小)	功能说明
feature	u32(4B)	设备特性标记位。
crtc_id	u32(4B)	显示控制器硬件编号。编号从0开始。
encoder_id	u32(4B)	与该显示控制器相连的内置信号编码器编号。
max_freq	u32(4B)	控制器硬件支持的最大像素时钟频率，参考芯片手册填写。
max_width	u32(4B)	控制器硬件能支持的最大分辨率宽度，参考芯片手册填写。
max_height	u32(4B)	控制器硬件能支持的最大分辨率高度，参考芯片手册填写。
is_vb_timing	u8(4B)	是否由vbios提供显示控制器的时序，默认值为 0，已废弃，工具中不提供配置界面。

## 5.2 内置编码器属性表

表5-2 信号转换器属性表

属性说明	类型(大小)	说明
feature	u32(4B)	设备特性标记位。
type	u32(4B)	内置编码器输出信号类型。
connector_id	u32(4B)	显示链路编号。

## 5.3 连接器属性表

表5-3 接口连接器属性表

属性名称	类型(大小)	功能说明
feature	u32(4B)	设备特性标记位。
i2c_id	u32(4B)	与连接器相连i2c编号。默认值为 -1，已废弃，工具中不提供配置界面。
internal_edid	u32(256B)	存放由vbios提供的edid。
type	u32(4B)	连接器接口类型。
hotplug	u32(4B)	热插拔探测方式的类型。
edid_method	u32(4B)	获取EDID方式。
irq_gpio	u32(4B)	用于热插拔探测的GPIO号。2K3000以后不再使用。工具中不提供配置界面。
gpio_placement	u32(4B)	使用GPIO的位置，默认值为 2K，已废弃，工具中不提供配置界面。

## 5.4 GPIO 设备属性表

表5-4 GPIO 设备属性表

属性名称	类型(大小)	说明
feature	u32(4B)	设备特性标记位。
gpio_id	u32(4B)	使用的GPIO编号。
usage	u32(4B)	GPIO用途。

## 5.5 PWM 设备属性表

表5-5 PWM设备属性表

属性名称	类型(大小)	说明
feature	u32(4B)	设备特性标记位。
pwm_id	u32(4B)	pwm 设备编号，参考主板进行配置。
polarity	u32(4B)	pwm 设备极性配置，参考主板进行配置。
peroid	u32(4B)	pwm 设备周期，单位Hz，参考芯片手册进行配置。
usage	u32(4B)	pwm 控制器的用途，参考主板进行配置。

## 5.6 GPU属性表

表5-6 GPU设备属性表

属性名称	类型(大小)	说明
feature	u32(4B)	设备特性标记位
gpu_freq_level	u32(4B)	GPU 频率调节档位，参考硬件配置。
gpu_freq	u32(4B)	GPU 最大频率，参考硬件配置。
num_gpu_shaders	u32(4B)	GPU 流处理器个数，参考硬件配置。

## 5.7 显存属性表

表5-7 显存设备属性表

属性名称	类型(大小)	说明
feature	u32(4B)	设备特性标记位。
vram_type	u32(4B)	显存颗粒类型。参考主板设计配置。
vram_bit_width	u32(4B)	显存颗粒位宽。参考主板设计配置。
vram_capacity	u32(4B)	显存容量。参考主板设计配置。
vram_freq	u32(4B)	显存频率。参考主板设计配置。

## 5.8 PANEL属性表

表5-8 PANEL属性表

属性名称	类型(大小)	说明
feature	u32(4B)	设备特性标记位。

属性名称	类型(大小)	说明
max_vrefresh	u32(4B)	主板支持的最大垂直刷新率，单位Hz，参考显示器参数。
max_hdisplay	u32(4B)	主板支持的最大水平像素数，参考显示器参数。
max_vdisplay	u32(4B)	主板支持的最大垂直像素数，参考显示器参数。
count	u32(4B)	主板不支持的分辨率数量，最大支持16组。
vrefresh	u32(4B)	主板不支持的垂直刷新率，单位Hz，参考显示器参数。
hdisplay	u32(4B)	主板不支持的水平像素数，参考显示器参数。
vdisplay	u32(4B)	主板不支持的垂直像素数，参考显示器参数。

## 6 efifb

efifb适用于符合UEFI规范的固件。efifb既支持带有图形输出协议（GOP）显示器的固件。efifb可以复用固件启动时的分辨率和帧缓冲区地址。因此可以实现在操作系统没有显示驱动的情况下支持基本的图形显示功能。

龙芯BIOS中已默认集成支持GOP驱动，在操作系统中没有集成显卡驱动或不满足显卡驱动运行条件时，可继续由efifb完成图形显示功能。**要求固件传递帧缓冲地址，分辨率，色深必须和固件使用相同，才能保证进入系统后正常显示。**

系统使用efifb进入桌面后可正常操作应用程序，如开启浏览器、终端、文件管理等应用程序；efifb只能提供基础的图形显示功能，不支持画面旋转、切换分辨率、更改刷新率、显示器热插拔、显示器信息获取等功能。为使操作系统获得完整的使用体验，使用efifb启动后请尝试安装最新显卡驱动。

## 7 附录

### 7.1 显示器方案配置示例

如表7-1，表7-2和7-3所示，为使用内置HDMI直接输出显示器时vbios中设备信息的描述表。

表7-1 内置HDMI配置

设备	显示控制器	内置编码器	外接编码器	连接器
属性/配置	feature (0)	feature (0)	feature (0)	feature (0)
	crtc_id (0)	encoder_id (0)	i2c_id (0)	i2c_id (0)
	encoder_id (0)	connector_id (0)	connector_id (0)	internal_edid (不配置)
	max_freq (340000)		type (TMDS)	type (HDMI)

设备	显示控制器	内置编码器	外接编码器	连接器
	max_width (4096)		config_type (driver)	hotplug (irq)
	max_height (4096)		chip_id (INTERNA_HDMI)	edid_method (vbios)
			chip_addr (0)	

表7-2 GPU配置

设备	GPU	显存
属性/配置	feature (0)	feature (0)
	gpu_freq_level (3)	vram_type (ddr4)
	gpu_freq (500MHz)	vram_bit_width (32 bit)
	num_gpu_shaders (64)	vram_capacity (1GB)
		vram_freq (2400MHz)

表7-3 PANEL配置

设备	PANEL	分辨率和刷新率
属性/配置	feature (0)	vrefresh(60Hz)
	max_vrefresh (30)	hdisplay(3840)
	max_hdisplay(3840)	vdisplay(2160)
	max_vdisplay(2160)	vrefresh(120Hz)
	count(2)	hdisplay(2880)
		vdisplay(1200)

## 7.2 液晶屏方案配置示例

如表7-4和表7-5所示，为使用eDP直接输出液晶屏时vbios对各设备信息的描述。

表7-4 外接eDP配置

设备	显示控制器	内置编码器	外接编码器	连接器	PWM
属性/配置	feature (0)	feature (0)	feature (0)	feature (0)	feature (0)
	crtc_id (0)	encoder_id (0)	i2c_id (0)	i2c_id (0)	pwm_id (2)

设备	显示控制器	内置编码器	外接编码器	连接器	PWM
	encoder_id (0)	connector_id (0)	connector_id (0)	internal_edid (不配置)	polarity (2000000)
	max_freq (340000)		type (LVDS)	type (eDP)	peroid (1)
	max_width (4096)		config_type (driver)	hotplug (connected)	usage (0)
	max_height (4096)		chip_id ((INTERNA_EDP)	edid_method (ENCODER)	
			chip_addr (0)		

表7-5 GPU配置

设备	GPU	显存
属性/配置	feature (0)	feature (0)
	gpu_freq_level (3)	vram_type (ddr4)
	gpu_freq (500MHz)	vram_bit_width (32 bit)
	num_gpu_shaders (64)	vram_capacity (1 GB)
		vram_freq (2400MHz)