

Q/LS

Q/SW-LS 0001-2023

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 架构嵌入式系列)

V1.0

2023-11-08 发布

2023-11-16 实施

龙芯中科技术股份有限公司 批准

版权声明

本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	V1.0
	创建人	研发部
版本历史		
序号	版本号	更新内容
1	V1.0	1、发布文档初始版本 V1.0 版

1 序言

龙芯 CPU 统一系统架构规范是龙芯产品在嵌入式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。

2 发布列表

序号	规范名称	版本号
1	龙芯 CPU 嵌入式系统固件与内核接口规范	V1.0
2	龙芯 2K 系列 SoC 集显嵌入式软硬件设计规范	V1.0

龙芯 CPU 嵌入式系统固件与内核接口规范

V1.0

版本信息

文档信息	文档名	龙芯 CPU 嵌入式系统固件与内核接口规范
	版本号	V1.0
	创建人	系统研发部
版本历史		
序号	版本号	更新内容
1	V1.0	发布文档初始版本 V1.0 版。

目 录

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 固件与内核接口传参规范	2
4.1 传参约定	2
4.2 命令行传递的约定	3
4.2.1 initrd 表	3
4.3 UEFI system table 传递的约定	3
4.3.1 memory map 表	3
4.3.2 initrd 表	3
4.3.3 FDT 表	3
4.4 内核集成 FDT 约定	4
4.5 集显的传参约定	4
4.6 其他约定	4
5 设备树节点配置规范	4
5.1 CPU	4
5.1.1 cpu-map	4
5.1.2 cpu	5
5.2 内存	5
5.2.1 memory	5
5.2.2 reserved-memory	5
5.2.3 memalloc	6
5.3 温度传感器	6
5.4 中断控制器	7
5.4.1 CPU 中断控制器	7
5.4.2 传统 IO 中断控制器	7
5.4.3 扩展 IO 中断控制器	8
5.4.4 桥片 IO 中断控制器	8
5.4.5 MSI 中断控制器	9
5.5 SPI 控制器	9
5.6 GPIO	9
5.7 UART 控制器	10
5.8 CAN	10
5.9 I2C 控制器	11
5.10 PWM 控制器	11
5.11 NAND 控制器	12
5.12 电源管理模块	12
5.13 RTC	13
5.14 HPET 控制器	13

5.15 SDIO 控制器	13
5.16 GMAC 控制器	14
5.16.1 配置为 PLATFORM 设备	14
5.16.2 配置为 PCI 设备	15
5.17 DWC2 控制器	15
5.18 DWC3 控制器	16
5.19 OHCI 控制器	16
5.19.1 配置为 PLATFORM 设备	16
5.19.2 配置为 PCI 设备	16
5.20 EHCI 控制器	17
5.20.1 配置为 PLATFORM 设备	17
5.20.2 配置为 PCI 设备	17
5.21 XHCI 控制器	18
5.21.1 配置为 PCI 设备	18
5.22 HDA 控制器	18
5.22.1 配置为 PLATFORM 设备	18
5.22.2 配置为 PCI 设备	18
5.23 SATA 控制器	19
5.23.1 配置为 PLATFORM 设备	19
5.23.2 配置为 PCI 设备	19
5.24 PCIE	20
5.24.1 PCIE Host Bridge	20
5.24.2 PCIE Bridge	20
5.24.3 PCIE Device	21
5.25 RapidIO 控制器	21
5.25.1 配置为 PCI 设备	21
5.26 DMA 控制器	21
5.27 APB 总线控制器	22
5.27.1 配置为 PCI 设备	22
5.28 REBOOT	22
5.29 POWEROFF	22
5.30 CHIPID	23
5.31 PINCTRL	23
5.32 根节点	23
5.33 其他配置	23
6 总结	24
附录 A 规范适用处理器型号列表	25
附录 B 集显传参方式适用处理器型号列表	26
附录 C 2K0500 设备树源码参考配置	27
附录 D 2K1000LA 设备树源码参考配置	41
附录 E 2K2000 设备树源码参考配置	56

1 范围

本规范规定了固件与内核接口传参实现及设备树节点配置的要求，适用于龙芯 LoongArch 架构的 2 号系列 CPU，具体 CPU 型号请参考附录 A。建议其它系统厂商遵循此规范开发相关产品。

本规范为通用规范，描述一般性的约定。针对不同的芯片，请参阅对应的芯片手册。

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：统一的可扩展固定接口，Unified Extensible Firmware Interface，是 Intel 为全新类型的 PC 固件的体系结构、接口和服务提出的建议标准。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，被看做是有近 20 多年历史的 PC BIOS 的继任者。

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。

U-Boot：一个主要用于嵌入式系统并且支持多种体系结构的引导加载程序。

FDT(Flattened Device Tree)：即设备树，是一种描述硬件资源的数据结构，以便操作系统内核可以使用和管理这些组件，可以通过 bootloader 将它传给内核。扁平设备树主要由三大部分组成：头 (header)、结构块 (struct block)、字符串块 (strings block)。

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。

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

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 2 号 CPU 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。

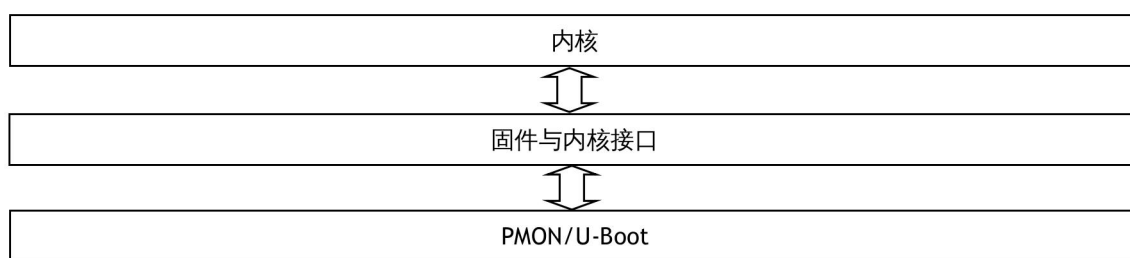


图 3-1 接口与内核和固件之间的关系

4 固件与内核接口传参规范

固件与内核之间按照内核参数约定传递参数。

4.1 传参约定

固件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 4-1 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, 固件可通过命令行的 noefi 禁止内核的运行时功能
a1	u64	64 位命令行物理地址
a2	u64	64 位 UEFI system table 物理地址

4.2 命令行传递的约定

4.2.1 initrd 表

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。

4.3 UEFI system table 传递的约定

4.3.1 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。

4.3.2 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-ca555231cc68})获取 initrd 表地址。

4.3.3 FDT 表

FDT 表的地址存储在 UEFI Configuration table 中，内核在 UEFI Configuration table 中查找 DEVICE_TREE_GUID({b1b621d5-f19c-41a5-830b-d9152c69aae0})获取 FDT 地址。

4.4 内核集成 FDT 约定

内核提供配置选项 CONFIG_BUILTIN_DTB 及 CONFIG_BUILTIN_DTB_NAME 来指定特定的 FDT，用于集成到内核。当内核集成 FDT 时，内核解析并使用该 FDT，不再使用固件提供的 FDT。

4.5 集显的传参约定

集显的传参方式分为设备树传参和 VBIOS 传参两种。不同 CPU 型号适用不同的传参方式，请参考附录 B。对于采用 VBIOS 传参方式的内置显卡，固件需要将 VBIOS 存放到内置显卡显存 BAR 空间的最后 1MB 内。

4.6 其他约定

UEFI Configuration table 包含了 memory map 表，initrd 表，FDT 表等，详细描述参考章节 4.3.1 memory map 表，4.3.2 initrd 表，4.3.3 FDT 表。其中，传参涉及到 memory map 表、initrd 表，FDT 表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（分别是命令行方式和 systemtable 方式，参见 4.2.1 和 4.3.2），兼容 UEFI 规范的固件推荐使用 UEFI system table 传递。

5 设备树节点配置规范

本节规定了龙芯 2 号系列平台设备树节点配置的要求。设备树具体配置请参考附录。

对于既可以配置为 PLATFORM 设备，也可以配置为 PCI 设备的节点，如 GMAC 控制器，USB 控制器（EHCI/OHCI/XHCI/OTG），HDA 控制器，SATA 控制器，DC 控制器，GPU 控制器，VPU 控制器等，本节对两者的配置规范都进行了描述。

对于硬件上以固定地址形式存在的设备，其设备节点配置为 PLATFORM 设备；对于硬件上以 PCI 形式存在的设备，部分配置为 PCI 设备节点，部分配置为 PLATFORM 设备节点。

对于硬件上以 PCI 形式存在，其设备节点配置为 PLATFORM 形式的设备，同时还需要配置其 PCI 形式的节点，节点中配置属性 status 值为"disabled"，避免内核 PCI 子系统对设备重新初始化。

设备的具体形态请参考对应的处理器用户手册。

5.1 CPU

5.1.1 cpu-map

必须配置子节点及属性：

(1) cluster

包含指定数量的 core。

(2) core

值：cpu = <&cpuN>。cpuN 为对应的 cpu 节点。

5.1.2 cpu

必须配置属性：

(1) device_type

值："cpu"。

(2) compatible

值："loongarch"。

(3) reg

CPU 的核号。

(4) clock-frequency

CPU 频率，单位 HZ。

5.2 内存

5.2.1 memory

必须配置属性：

(1) device_type

值："memory"。

(2) reg

可用内存的起始地址及大小。

5.2.2 reserved-memory

必须配置属性：

(1) #address-cells

如果根节点定义了#address-cells，必须与根节点#address-cells 保持一致。

如果根节点没有定义#address-cells，与内核定义的 OF_ROOT_NODE_ADDR_CELLS_DEFAULT 保持一致。

(2) #size-cells

如果根节点定义了#size-cells，必须与根节点#size-cells 保持一致。

如果根节点没有定义#size-cells，与内核定义的 OF_ROOT_NODE_SIZE_CELLS_DEFAULT 保持一致。

(3) ranges

该属性为空值。

linux,cma 子节点必须配置属性：

(1) compatible

值："shared-dma-pool"。

(2) reusable

空值。

(3) size

定义区域的大小。

(4) linux,cma-default

空值。标识该区域作为 cma 分配的缺省区域。

5.2.3 memalloc

必须配置属性：

(1) compatible

值："loongson,ls-memalloc"。

(2) reg

分配内存区域的起始地址及大小。

5.3 温度传感器

必须配置属性：

(1) compatible

值："loongson,ls2k-tsensors"。

(2) reg

设备处于父级总线地址空间的基地址及大小。

(3) interrupt-parent

路由到的中断控制器节点句柄。

(4) interrupts

在 interrupt-parent 对应的中断控制器中，使用的中断。

(5) id

温度传感器编号。

(6) #thermal-sensor-cells

定义描述该节点属性的单位个数。

5.4 中断控制器

5.4.1 CPU 中断控制器

必须配置属性：

(1) compatible

值："loongson,cpu-interrupt-controller"。

(2) interrupt-controller

空值，标识为中断控制器。

(3) #interrupt-cells

定义解析一个中断属性所需要的单位个数。

5.4.2 传统 IO 中断控制器

必须配置属性：

(1) compatible

值："loongson,lioinctc-2.0"。

(2) interrupt-controller

空值，标识此节点为中断控制器。

(3) reg

设备处于父级总线地址空间的基地址及大小。结合 reg-names 配置使用。

(4) reg-names

reg 中依次对应的寄存器空间名称，依次为"main"，"isr0"，"isr1"，... ，"isrN"。其中，"main"标识为中断控制器寄存器基地址；"isr0"~"isrN"依次标识为路由至 CPU 核 0~N 的中断状态寄存器，故按照 CPU 核数进行配置。

(5) #interrupt-cells

定义解析一个中断属性所需要的单位个数。

(6) interrupt-parent

路由到的中断控制器节点句柄。

(7) interrupts

在 interrupt-parent 对应的中断控制器中，使用的中断号。值 2，3，4，5 分别对应下面的 HWI0~HWI3，且只能配置一个值。

(8) interrupt-names

依次标识 interrupts 中所配置的中断名称，为"int0"，"int1"，"int2"，"int3"的子集。

(9) loongson,parent_int_map

包含 4 个 32 位数的元组，依次表示路由至 CPU 核中断引脚 HWI0~HWI3 的中断。HWI0~HWI3 对应的 interrupt-names 分别为"int0"，"int1"，"int2"，"int3"。

5.4.3 扩展 IO 中断控制器

必须配置属性：

(1) compatible

支持以下几种 compatible 属性：

"loongson,ls2k0500-eointc";

"loongson,ls2k2000-eointc";

不同平台使用的 compatible 不同，请参考平台相关的 dts 文件。

(2) interrupt-controller

空值，标识此节点为中断控制器。

(3) #interrupt-cells

定义解析一个中断属性所需要的单位个数。

(4) interrupt-parent

路由到的中断控制器节点句柄。

(5) interrupts

在 interrupt-parent 对应的中断控制器中，使用的中断。

(6) interrupt-names

表示 interrupts 所配置的中断名称。

使用扩展 IO 中断，需要使能扩展 IO 中断使能位，由固件负责使能，缺省开启状态。

5.4.4 桥片 IO 中断控制器

必须配置属性：

(1) compatible

值："loongson,pch-pic-1.0"。

(2) reg

设备处于父级总线地址空间的基地址及大小。

(3) interrupt-controller

空值，标识此节点为中断控制器。

(4) #interrupt-cells

定义解析一个中断属性所需要的单位个数。

(5) loongson,pic-base-vec

定义该中断控制器描述的起始中断向量号。

(6) interrupt-parent

路由到的中断控制器节点句柄。

5.4.5 MSI 中断控制器

(1) compatible

值："loongson,pch-msi-1.0"。

(2) reg

设备处于父级总线地址空间的基地址及大小。

(3) interrupt-controller

空值，标识此节点为中断控制器。

(4) loongson,msi-base-vec

msi 中断控制器支持的起始中断向量号。

(5) loongson,msi-num-vecs

msi 中断控制器支持的中断向量数量。

(6) interrupt-parent

路由到的中断控制器节点句柄。

5.5 SPI 控制器

必须配置属性：

(1) compatible

值："loongson,ls-spi"

(2) reg

设备处于父级总线地址空间的基地址及大小。

5.6 GPIO

必须配置属性：

(1) compatible

值："loongson,ls2k-gpio"和"loongson,loongson3-gpio"。

不同平台使用的 compatible 不同，具体参考平台相关的 dts 文件。

(2) reg

设备处于父级总线地址空间的基地址及大小。

(3) ngpios

gpio 的个数。

(4) gpio-controller

空值，标识为 gpio 控制器。

(5) #gpio-cells

描述节点属性使用的单位数量。

(6) conf_offset

gpio 方向控制寄存器偏移地址。

(7) out_offset

gpio 输出设置寄存器偏移地址。

(8) in_offset

gpio 输入采样寄存器偏移地址。

(9) inten_offset

gpio 中断使能寄存器偏移地址。

(10) in_start_bit

该属性用于兼容 3A3000 处理器型号，在 2K 系列平台，值定义为 0。

(11) gpio_base

定义节点描述的 gpio base 号。gpio 节点支持最多描述 64 个 gpio，如果处理器支持多于 64 个 gpio，需要定义 2 个 gpio 节点来描述，第 2 个 gpio 节点的 gpio_base 从 64 开始。

(12) support_irq

空值，标识 gpio 支持中断。

(13) interrupt-parent

路由到的中断控制器节点句柄。

(14) interrupts

在 interrupt-parent 对应的中断控制器中，使用的中断。

(15) interrupts-extended

该属性的作用与 interrupt-parent 和 interrupts 的组合使用是一致的。二者选其一即可。

5.7 UART 控制器

必须配置属性：

(1) compatible

值："ns16550a"。

(2) reg

设备处于父级总线地址空间的基地址及大小。

(3) clock-frequency

定义 uart 时钟频率。

(4) interrupt-parent

路由到的中断控制器节点句柄。

(5) interrupts

在 interrupt-parent 对应的中断控制器中，使用的中断。

5.8 CAN

必须配置属性：

(1) compatible

值："nxp,sja1000"。

(2) reg

设备处于父级总线地址空间的基地址及大小。

(3) interrupt-parent

路由到的中断控制器节点句柄。

(4) interrupts

在 interrupt-parent 对应的中断控制器中，使用的中断。

(5) nxp,external-clock-frequency

定义外部时钟频率。

5.9 I2C 控制器

必须配置属性：

(1) compatible

值："loongson,ls2k-i2c"。

(2) reg

设备处于父级总线地址空间的基地址及大小。

(3) interrupt-parent

路由到的中断控制器节点句柄。

(4) interrupts

在 interrupt-parent 对应的中断控制器中，使用的中断。

5.10 PWM 控制器

必须配置属性：

(1) compatible

值："loongson,ls2k-pwm"。

(2) reg

设备处于父级总线地址空间的基地址及大小。

(3) clock-frequency

输入时钟频率。

(4) interrupt-parent

路由到的中断控制器节点句柄。

(5) interrupts

在 interrupt-parent 对应的中断控制器中，使用的中断。

(6) #pwm-cells

定义描述该节点属性的单位个数。

5.11 NAND 控制器

必须配置属性：

(1) compatible

值："loongson,ls-nand"。

(2) reg

设备处于父级总线地址空间的基地址及大小。

(3) interrupt-parent

路由到的中断控制器节点句柄。

(4) interrupts

在 interrupt-parent 对应的中断控制器中，使用的中断。

(5) interrupt-names

定义中断名称。

(6) dmas

使用的 DMA 控制器节点句柄。

(7) dma-names

值："nand_rw"。

(8) dma-mask

dma bit 位掩码。

(9) nand-cs

定义片选信号。

(10) number-of-parts

nand 的分区数量。

partition 子节点配置请参考相应规范和示例。

5.12 电源管理模块

必须配置属性：

(1) compatible

支持以下几种 compatible 属性：

"loongson,ls2k0500-pmc", "syscon";

"loongson,ls2k1000-pmc", "syscon";

"loongson,ls2k2000-pmc", "syscon";

不同平台使用的 compatible 不同，请参考平台相关的 dts 文件。

(2) reg

设备处于父级总线地址空间的基地址及大小。

(3) interrupt-parent

路由到的中断控制器节点句柄。

(4) interrupts

在 interrupt-parent 对应的中断控制器中，使用的中断。

如果固件支持休眠唤醒，可以增加如下属性配置：

(5) loongson,suspend-address

休眠时，固件程序入口地址。

5.13 RTC

必须配置属性：

(1) compatible

支持以下几种 compatible 属性：

"loongson,ls2k0500-rtc", "loongson,ls7a-rtc";

"loongson,ls2k1000-rtc", "loongson,ls7a-rtc";

"loongson,ls2k2000-rtc", "loongson,ls7a-rtc";

不同平台使用的 compatible 不同，请参考平台相关的 dts 文件。

(2) reg

设备处于父级总线地址空间的基地址及大小。

(4) interrupt-parent

路由到的中断控制器节点句柄。

(5) interrupts

在 interrupt-parent 对应的中断控制器中，使用的中断。

5.14 HPET 控制器

必须配置属性：

(1) compatible

值："loongson,loongson2-hpet"。

(2) reg

设备处于父级总线地址空间的基地址及大小。

(3) clock-frequency

输入时钟频率。

(4) interrupt-parent

路由到的中断控制器节点句柄。

(5) interrupts

在 interrupt-parent 对应的中断控制器中，使用的中断。

5.15 SDIO 控制器

必须配置属性：

(1) compatible

值："loongson,ls2k_sdio"

(2) reg

设备处于父级总线地址空间的基地址及大小。

(3) interrupt-parent

路由到的中断控制器节点句柄。

(4) interrupts

在 interrupt-parent 对应的中断控制器中，使用的中断。

(5) interrupt-names

中断名称

(6) dmas

sdio 控制器使用的 dma 控制器

(7) dma-names

定义 dma 通道名称。

(8) dma-mask

设置 dma 控制器的 bit 位掩码

(9) clock-frequency

输入时钟频率

(10) cd-gpio

sdio detect 引脚使用的 gpio。通过 gpio 中断方式检测热插拔。

(11) non-removable

可选配置属性。该属性标识不支持热插拔。

(12) broken-cd

可选配置属性。该属性标识轮询方式检测热插拔。

5.16 GMAC 控制器

5.16.1 配置为 PLATFORM 设备

必须配置属性：

(1) compatible

值："snps,dwmac-3.70a"。

(2) reg

设备处于父级总线地址空间的基地址及大小。

(3) interrupt-parent

路由到的中断控制器节点句柄。

(4) interrupts

在 interrupt-parent 对应的中断控制器中，使用的中断。

(5) interrupt-names

依次标识 interrupts 中所配置的中断名称。

(6) phy-mode

phy 接口操作模式。

(7) bus_id

mii 总线号。

(8) mac-address

非必须属性。如果需要，可以通过该属性设置 mac 地址。

(9) dma-mask

定义 dma 掩码 bit 位数。

(10) snps,multicast-filter-bins

定义 MAC 地址 Hash 过滤组播报文的 Hash 表位数。

5.16.2 配置为 PCI 设备

必须配置属性：

(1) reg

5-cell PCI resource。第 1 个 cell 的 bit8-bit15 为 devfn 值。

(2) interrupts

在 interrupt-parent 对应的中断控制器中，使用的中断。

(3) interrupt-parent

路由到的中断控制器节点句柄。

可选配置属性：

(4) status

值："disabled"。如果设备与其它设备功能（管脚）复用，且选择其它设备功能，可以配置该属性，禁用该设备。

5.17 DWC2 控制器

必须配置属性：

(1) compatible

值："loongson,loongson2-dwc2"。

(2) reg

设备处于父级总线地址空间的基地址及大小。

(3) interrupt-parent

路由到的中断控制器节点句柄。

(4) interrupts

在 interrupt-parent 对应的中断控制器中，使用的中断。

(5) dma-mask

设置 dma bit 位掩码。

(6) dr_mode

可设置工作模式为："host", "peripheral"和"otg"。

5.18 DWC3 控制器

必须配置属性：

(1) compatible

值："synopsys,dwc3"。

(2) reg

设备处于父级总线地址空间的基地址及大小。

(3) interrupt-parent

路由到的中断控制器节点句柄。

(4) interrupts

在 interrupt-parent 对应的中断控制器中，使用的中断。

(5) usb-phy

设置 usb_phy，指向节点 usb2_phy 和 usb3_phy。

(6) dr_mode

可设置工作模式为："host"，"peripheral"和"otg"。

5.19 OHCI 控制器

5.19.1 配置为 PLATFORM 设备

必须配置属性：

(1) compatible

值："generic-ohci"。

(2) reg

设备处于父级总线地址空间的基地址及大小。

(3) interrupt-parent

路由到的中断控制器节点句柄。

(4) interrupts

在 interrupt-parent 对应的中断控制器中，使用的中断。

5.19.2 配置为 PCI 设备

必须配置属性：

(1) reg

5-cell PCI resource。第 1 个 cell 的 bit8-bit15 为 devfn 值。

(2) interrupts

在 interrupt-parent 对应的中断控制器中，使用的中断。

(3) interrupt-parent

路由到的中断控制器节点句柄。

可选配置属性：

(4) status

值："disabled"。如果设备与其它设备功能（管脚）复用，且选择其它设备功能，可以配置该属性，禁用该设备。

5.20 EHCI 控制器

5.20.1 配置为 PLATFORM 设备

必须配置属性：

(1) compatible

值："generic-ehci"。

(2) reg

设备处于父级总线地址空间的基地址及大小。

(3) interrupt-parent

路由到的中断控制器节点句柄。

(4) interrupts

在 interrupt-parent 对应的中断控制器中，使用的中断。

5.20.2 配置为 PCI 设备

必须配置属性：

(1) reg

5-cell PCI resource。第 1 个 cell 的 bit8-bit15 为 devfn 值。

(2) interrupts

在 interrupt-parent 对应的中断控制器中，使用的中断。

(3) interrupt-parent

路由到的中断控制器节点句柄。

可选配置属性：

(4) status

值："disabled"。如果设备与其它设备功能（管脚）复用，且选择其它设备功能，可以配置该属性，禁用该设备。

5.21 XHCI 控制器

5.21.1 配置为 PCI 设备

必须配置属性：

(1) reg

5-cell PCI resource。第 1 个 cell 的 bit8-bit15 为 devfn 值。

(2) interrupts

在 interrupt-parent 对应的中断控制器中，使用的中断。

(3) interrupt-parent

路由到的中断控制器节点句柄。

可选配置属性：

(4) status

值："disabled"。如果设备与其它设备功能（管脚）复用，且选择其它设备功能，可以配置该属性，禁用该设备。

5.22 HDA 控制器

5.22.1 配置为 PLATFORM 设备

必须配置属性：

(1) compatible

值："loongson,ls2k-audio"。

(2) reg

设备处于父级总线地址空间的基地址及大小。

(3) interrupt-parent

路由到的中断控制器节点句柄。

(4) interrupts

在 interrupt-parent 对应的中断控制器中，使用的中断。

5.22.2 配置为 PCI 设备

必须配置属性：

(1) reg

5-cell PCI resource。第 1 个 cell 的 bit8-bit15 为 devfn 值。

(2) interrupts

在 interrupt-parent 对应的中断控制器中，使用的中断。

(3) interrupt-parent

路由到的中断控制器节点句柄。

可选配置属性：

(4) status

值："disabled"。如果设备与其它设备功能（管脚）复用，且选择其它设备功能，可以配置该属性，禁用该设备。

5.23 SATA 控制器

5.23.1 配置为 PLATFORM 设备

必须配置属性：

(1) compatible

值："snps,spear-ahci"。

(2) reg

设备处于父级总线地址空间的基地址及大小。

(3) interrupt-parent

路由到的中断控制器节点句柄。

(4) interrupts

在 interrupt-parent 对应的中断控制器中，使用的中断。

5.23.2 配置为 PCI 设备

必须配置属性：

(1) reg

5-cell PCI resource。第 1 个 cell 的 bit8-bit15 为 devfn 值。

(2) interrupts

在 interrupt-parent 对应的中断控制器中，使用的中断。

(3) interrupt-parent

路由到的中断控制器节点句柄。

可选配置属性：

(4) status

值："disabled"。如果设备与其它设备功能（管脚）复用，且选择其它设备功能，可以配置该属性，禁用该设备。

5.24 PCIE

5.24.1 PCIE Host Bridge

必须配置属性：

(1) compatible

值："loongson,ls2k-pci"。

(2) reg

配置空间基地址及大小。

(3) ranges

MEM 地址空间以及 IO 地址空间。

(4) device_type

值："pci"。

(5) bus-range

定义总线号范围。

(6) #size-cells

解析子节点 size 域使用的单元个数。

(7) #address-cells

解析子节点 address 域使用的单元个数。

(8) msi-parent

如果使用 msi 中断，配置该属性指向 msi 中断控制器。

5.24.2 PCIE Bridge

必须配置属性：

(1) reg

5-cell PCI resource。第 1 个 cell 的 bit8-bit15 为 devfn 值。

(2) interrupts

在 interrupt-parent 对应的中断控制器中，使用的中断。

(3) interrupt-parent

路由到的中断控制器节点句柄。

(4) #interrupt-cells

定义解析一个中断属性所需要的单位个数。

(5) interrupt-map

该属性描述了 PCIE Bridge 路由到的中断控制器节点及中断号。

(6) interrupt-map-mask

该属性用于设置 interrupt-map 属性的 mask，结合 interrupt-map 使用。

5.24.3 PCIE Device

参考规范中其他章节关于设备“配置为 PCI 设备”的描述。

5.25 RapidIO 控制器

5.25.1 配置为 PCI 设备

必须配置属性：

(1) reg

5-cell PCI resource。第 1 个 cell 的 bit8-bit15 为 devfn 值。

(2) interrupts

在 interrupt-parent 对应的中断控制器中，使用的中断。

(3) interrupt-parent

路由到的中断控制器节点句柄。

可选配置属性：

(4) status

值："disabled"。如果设备与其它设备功能（管脚）复用，且选择其它设备功能，可以配置该属性，禁用该设备。

5.26 DMA 控制器

必须配置属性：

(1) compatible

值："loongson,loongson2-apbdma"。

(2) reg

设备处于父级总线地址空间的基地址及大小。

(3) loongson,apbdma-conf

指向 APBDMA 配置寄存器，配置 DMA 控制器应用于具体设备。

(4) #dma-cells

定义解析 dma 属性所需要的单位个数。

(5) dma-channels

DMA 通道数量。

(6) interrupt-parent

路由到的中断控制器节点句柄。

(7) interrupts

在 interrupt-parent 对应的中断控制器中，使用的中断。

5.27 APB 总线控制器

5.27.1 配置为 PCI 设备

必须配置属性：

(1) reg

5-cell PCI resource。第 1 个 cell 的 bit8-bit15 为 devfn 值。

(2) status

值："disabled"。在某些型号的处理器的 APB 总线控制器是 PCI 设备形式，但 APB 下挂载的 UART、I2C、PWM、RTC、NAND 等多种控制器类型不是 PCI 设备形式，这些控制器均配置为 PLATFORM 形式的设备节点，分配固定地址。APB 总线控制器不能再被内核 PCI 子系统重新初始化，所以需要建立 APB 总线控制器的 PCI 设备节点，且配置属性 status="disabled"。

5.28 REBOOT

必须配置属性：

(1) compatible

值："syscon-reboot"

(2) regmap

指向 pmc 设备节点。

(3) offset

电源管理模块的 reboot 相关控制器寄存器偏移。

(4) mask

访问寄存器的 bit 位掩码。

5.29 POWEROFF

必须配置属性：

(1) compatible

值："syscon-poweroff"

(2) regmap

关联电源管理设备节点。

(3) offset

电源管理模块 poweroff 相关控制器寄存器偏移。

(4) mask

访问寄存器的 bit 位掩码。

(5) value

设置寄存器的值。

5.30 CHIPID

必须配置属性：

(1) compatible

值："loongson,ls2k-chipid"

(2) reg

设备处于父级总线地址空间的基地址及大小。

(3) little-endian

小端模式

5.31 PINCTRL

必须配置属性：

(1) compatible

值："loongson,ls2k-pinctrl"

(2) reg

设备处于父级总线地址空间的基地址及大小。

具体设备的管脚复用配置及 pinctrl 设备节点的引用方式，请参考附录相应平台的设备树源码参考配置。

处理器管脚复用由固件根据板卡配置，PINCTRL 不作为管脚复用配置规范，用户可根据需要选择使用。

5.32 根节点

必须配置属性：

(1) model

格式：“处理器名称+空格+板卡描述”。其中，处理器名称值固定不可更改，对应 cpuinfo 中的 model name。

不同平台的处理器名称不一样，描述如下：

2K0500 平台： "Loongson-2K0500 Reference Board";

2K1000LA 平台："Loongson-2K1000 Reference Board";

2K2000 平台： "Loongson-2K2000 Reference Board";

5.33 其他配置

设备树中还配置了 spidev，eeprom，audio codec，thermal-zone，nand partition，i2c-gpio 等设备节点及属性，不属于处理器相关设备，用户可根据需求选择使用，配置方法可参考附录，也可查询相关约定。

6 总结

本规范的提出和制定旨在规范龙芯 CPU 嵌入式系统的固件和内核接口,重点规定了固件与内核接口传参实现及设备树节点配置等方面,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 规范适用处理器型号列表

2K0500, 2K1000LA, 2K2000 处理器。

附录 B 集显传参方式适用处理器型号列表

处理器型号	集显传参方式
2K0500	设备树
2K1000LA	设备树
2K2000	VBIOS

附录 C 2K0500 设备树源码参考配置

```
#include <dt-bindings/interrupt-controller/irq.h>
#include <dt-bindings/gpio/gpio.h>

/dts-v1/;
/ {
    model = "Loongson-2K0500 Reference Board";
    compatible = "loongson,loongson3";
    #address-cells = <2>;
    #size-cells = <2>;

    aliases {
        ethernet0 = &gmac0;
        ethernet1 = &gmac1;
        serial0 = &cpu_uart0;
        spi0 = &spi0;
        i2c0 = &i2c0;
        i2c1 = &i2c1;
        i2c2 = &i2c2;
        i2c3 = &i2c3;
        i2c4 = &i2c4;
        i2c5 = &i2c5;
    };

    chosen {
        stdout-path = "serial0:115200n8";
        bootargs = "earlycon";
    };

    memory {
        device_type = "memory";
        reg = <0 0x00200000 0 0x0ee00000
            0 0x90000000 0 0x60000000>;
    };

    reserved-memory {
        #address-cells = <2>;
        #size-cells = <2>;
        ranges;

        linux,cma {
            compatible = "shared-dma-pool";
            reusable;
            size = <0x0 0x2000000>;
            linux,cma-default;
        };
    };

    cpus {
        #address-cells = <1>;
        #size-cells = <0>;
        cpu-map {
            cluster0 {
```

```

        core0 {
            cpu = <&cpu0>;
        };
    };
};

cpu0: cpu@10000 {
    device_type = "cpu";
    compatible = "loongarch";
    clock-frequency = <500000000>;
    reg=<0>;
};

};

cpuic: interrupt-controller {
    compatible = "loongson,cpu-interrupt-controller";
    interrupt-controller;
    #interrupt-cells = <1>;
};

liointc0: interrupt-controller@1fe11400 {
    compatible = "loongson,liointc-2.0";
    reg = <0 0x1fe11400 0 0x40>,
        <0 0x1fe11040 0 0x8>;
    reg-names = "main", "isr0";

    interrupt-controller;
    #interrupt-cells = <2>;
    interrupt-parent = <&cpuic>;
    interrupts = <2>;
    interrupt-names = "int0";

    loongson,parent_int_map = <0xffffffff>, /* int0 */
        <0x00000000>, /* int1 */
        <0x00000000>, /* int2 */
        <0x00000000>; /* int3 */
};

liointc1: interrupt-controller@1fe11440 {
    compatible = "loongson,liointc-2.0";
    reg = <0 0x1fe11440 0 0x40>,
        <0 0x1fe11048 0 0x8>;
    reg-names = "main", "isr0";

    interrupt-controller;
    #interrupt-cells = <2>;

    interrupt-parent = <&cpuic>;
    interrupts = <4>;
    interrupt-names = "int2";

    loongson,parent_int_map = <0x00000000>, /* int0 */
        <0x00000000>, /* int1 */
        <0xffffffff>, /* int2 */
        <0x00000000>; /* int3 */
};

```

```

};

extioic: interrupt-controller@0x1fe11600 {
    compatible = "loongson,ls2k0500-eiointc";
    interrupt-controller;
    #interrupt-cells = <1>;
    interrupt-parent = <&cpuic>;
    interrupts = <3>;
};

soc {
    compatible = "ls,nbus", "simple-bus";
    #address-cells = <2>;
    #size-cells = <2>;
    ranges = <0 0x10000000 0 0x10000000 0 0x10000000
        0 0x2000000 0 0x2000000 0 0x2000000
        0 0x20000000 0 0x20000000 0 0x10000000
        0 0x40000000 0 0x40000000 0 0x40000000
        0xfe 0x00000000 0xfe 0x00000000 0 0x40000000>;

    isa@16400000 {
        compatible = "isa";
        #size-cells = <1>;
        #address-cells = <2>;
        ranges = <1 0 0 0x16400000 0x4000>;
    };

    generic_conf0: generic_conf@1fe10100 {
        compatible = "syscon";
        reg = <0 0x1fe10100 0 0x0004>;
    };

    cpu_uart0: serial@0x1ff40800 {
        compatible = "ns16550a";
        reg = <0 0x1ff40800 0 0x10>;
        clock-frequency = <100000000>;
        interrupt-parent = <&extioic>;
        interrupts = <2>;
        no-loopback-test;
    };

    pctrl: pinctrl@1fe00420 {
        compatible = "loongson,ls2k-pinctrl";
        reg = <0 0x1fe00420 0 0x18>;
        hda_default:hda {
            mux1 {
                groups = "hda";
                function = "hda";
            };
        };
    };

    dc@0x400c0000 {
        compatible = "loongson,la2k0500-dc", "loongson,display-subsystem";
        reg = <0 0x1f010000 0 0x10000>;
        interrupt-parent = <&liointc0>;
        interrupts = <28 IRQ_TYPE_LEVEL_HIGH>;
        dma-mask = <0x00000000 0xffffffff>;
    };
};

```

```
output-ports = <&dvo0 &vga>;

#address-cells = <1>;
#size-cells = <0>;

dc_identify {
    model = "loongson,2k500";
    compatible = "loongson,ls2k";
};

dvo0: dvo@0 {
    /* 0 for connector 0 (DVO0) */
    reg = <0>;
    ddc-i2c-bus = <&i2c4>;
    connector = "dvi-connector";
    status = "ok";
};

vga: vga@1 {
    /* 1 for connector 1 (DVO1) */
    reg = <1>;
    ddc-i2c-bus = <&i2c5>;
    connector = "vga-connector";
    status = "ok";
};

ahci@0x1f040000 {
    compatible = "snps,spear-ahci";
    reg = <0 0x1f040000 0 0x10000>;
    interrupt-parent = <&extioic>;
    interrupts = <75>;
};

pmc: acpi@0x1ff6c000 {
    compatible = "loongson,ls2k0500-pmc", "syscon";
    reg = <0x0 0x1ff6c000 0x0 0x58>;
    interrupt-parent = <&extioic>;
    interrupts = <56>;
    loongson,suspend-address = <0x0 0x1c000500>;
};

reboot {
    compatible = "syscon-reboot";
    regmap = <&pmc>;
    offset = <0x30>;
    mask = <0x1>;
};

poweroff {
    compatible = "syscon-poweroff";
    regmap = <&pmc>;
    offset = <0x14>;
    mask = <0x3c00>;
    value = <0x3c00>;
};
```

```

dwc2@0x1f080000 {
    compatible = "loongson,loongson2-dwc2";
    reg = <0 0x1f080000 0 0x40000>;
    interrupt-parent = <&extioic>;
    interrupts = <73>;
    dma-mask = <0x0 0xffffffff>;
    dr_mode = "otg";
    /* Optional for that dr_mode = "host" or dr_mode = "peripheral" */
};

gmac0: ethernet@0x1f020000 {
    compatible = "snps,dwmac-3.70a";
    reg = <0 0x1f020000 0 0x10000>;
    interrupt-parent = <&liointc0>;
    interrupts = <12 IRQ_TYPE_LEVEL_HIGH>;
    interrupt-names = "macirq";
    mac-address = [ 64 48 48 48 48 60 ];/* [>mac 64:48:48:48:48:60 <]*/
    phy-mode = "rgmii";
    bus_id = <0x0>;
    dma-mask = <0xffffffff 0xffffffff>;
};

gmac1: ethernet@0x1f030000 {
    compatible = "snps,dwmac-3.70a";
    reg = <0 0x1f030000 0 0x10000>;
    interrupt-parent = <&liointc0>;
    interrupts = <14 IRQ_TYPE_LEVEL_HIGH>;
    interrupt-names = "macirq";
    mac-address = [ 64 48 48 48 48 61 ];/* [>mac 64:48:48:48:48:61 <]*/
    phy-mode = "rgmii";
    bus_id = <0x1>;
    dma-mask = <0xffffffff 0xffffffff>;
};

ohci@0x1f058000 {
    compatible = "generic-ohci";
    reg = <0 0x1f058000 0 0x8000>;
    interrupt-parent = <&extioic>;
    interrupts = <72>;
};

ehci@0x1f050000 {
    compatible = "generic-ehci";
    reg = <0 0x1f050000 0 0x8000>;
    interrupt-parent = <&extioic>;
    interrupts = <71>;
};

usb2_phy: usb2phy@dwc3 {
    compatible = "usb-dummy-phy";
};

usb3_phy: usb3phy@dwc3 {
    compatible = "usb-dummy-phy";
};

```

```

dwc3@0x1f060000 {
    compatible = "synopsys,dwc3";
    reg = <0 0x1f060000 0 0x10000>;
    interrupt-parent = <&extioic>;
    interrupts = <74>;
    usb-phy = <&usb2_phy>, <&usb3_phy>;
    dr_mode = "host";
};

pci@0x16000000 {
    compatible = "loongson,ls2k-pci";
    device_type = "pci";
    bus-range = <0x1 0x6>;
    #size-cells = <2>;
    #address-cells = <3>;

    reg = <0xfe 0x00000000 0 0x20000000>;
    ranges = <0x02000000 0 0x40000000 0 0x40000000 0 0x40000000
        0x01000000 0 0x00004000 0 0x16404000 0x0 0x4000>;

    pci_bridge@0,0 {
        compatible = "pciclass060400",
            "pciclass0604";

        reg = <0x0000 0x0 0x0 0x0 0x0>;
        interrupts = <0 IRQ_TYPE_LEVEL_HIGH>;
        interrupt-parent = <&liointc1>;

        #interrupt-cells = <1>;
        interrupt-map-mask = <0 0 0 0>;
        interrupt-map = <0 0 0 0 &liointc1 0 IRQ_TYPE_LEVEL_HIGH>;
    };
    pci_bridge@1,0 {
        compatible = "pciclass060400",
            "pciclass0604";

        reg = <0x0800 0x0 0x0 0x0 0x0>;
        interrupts = <1 IRQ_TYPE_LEVEL_HIGH>;
        interrupt-parent = <&liointc1>;

        #interrupt-cells = <1>;
        interrupt-map-mask = <0 0 0 0>;
        interrupt-map = <0 0 0 0 &liointc1 1 IRQ_TYPE_LEVEL_HIGH>;
    };
};

#ifdef LS2K500_HAVE_PCI
ls2k500pci@0x17100000 {
    compatible = "loongson,ls2k500-pci";
    #interrupt-cells = <1>;
    bus-range = <0x10 0x14>;
    #size-cells = <2>;
    #address-cells = <3>;
    linux,pci-domain = <2>;

    #if 0
pci-gpios = <&pioA 0 0>, <&pioA 1 0>;
    #endif
};

```



```

IRQ_TYPE_LEVEL_HIGH>,<&liointc1      26      IRQ_TYPE_LEVEL_HIGH>,<&liointc1      26
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      26      IRQ_TYPE_LEVEL_HIGH>,<&liointc1      26
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      26      IRQ_TYPE_LEVEL_HIGH>,<&liointc1      26
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27      IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27      IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27      IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27      IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27      IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27      IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27      IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27      IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27      IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27      IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27      IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27      IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27      IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27      IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27      IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27      IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27      IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27      IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27      IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27      IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27      IRQ_TYPE_LEVEL_HIGH>,<&liointc1      27
};

pioB:gpio@0x1fe10450 {
    compatible = "loongson,loongson3-gpio";
    reg = <0 0x1fe10450 0 0x20>;
    gpio-controller;
    #gpio-cells = <2>;
    ngpios = <64>;
    conf_offset = <0>;
    out_offset = <0x10>;
    in_offset = <0x8>;
    in_start_bit = <0>;
    inten_offset = <0x98>;
    gpio_base = <64>;
    support_irq;
    interrupts-extended =
        /*
         * Every 32 gpios share a interrupt line. We need to disable
         * unnecessary GPIO interrupts in the firmware.
         */
        <&liointc1      28      IRQ_TYPE_LEVEL_HIGH>,<&liointc1      28
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      28      IRQ_TYPE_LEVEL_HIGH>,<&liointc1      28
        <&liointc1      28      IRQ_TYPE_LEVEL_HIGH>,<&liointc1      28
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      28      IRQ_TYPE_LEVEL_HIGH>,<&liointc1      28

```

```

IRQ_TYPE_LEVEL_HIGH>,
    <&liointc1          28          IRQ_TYPE_LEVEL_HIGH>,<&liointc1          28
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      28          IRQ_TYPE_LEVEL_HIGH>,<&liointc1      28
IRQ_TYPE_LEVEL_HIGH>,
    <&liointc1          28          IRQ_TYPE_LEVEL_HIGH>,<&liointc1          28
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      28          IRQ_TYPE_LEVEL_HIGH>,<&liointc1      28
IRQ_TYPE_LEVEL_HIGH>,
    <&liointc1          28          IRQ_TYPE_LEVEL_HIGH>,<&liointc1          28
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      28          IRQ_TYPE_LEVEL_HIGH>,<&liointc1      28
IRQ_TYPE_LEVEL_HIGH>,
    <&liointc1          28          IRQ_TYPE_LEVEL_HIGH>,<&liointc1          28
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      28          IRQ_TYPE_LEVEL_HIGH>,<&liointc1      28
IRQ_TYPE_LEVEL_HIGH>,
    <&liointc1          28          IRQ_TYPE_LEVEL_HIGH>,<&liointc1          28
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      28          IRQ_TYPE_LEVEL_HIGH>,<&liointc1      28
IRQ_TYPE_LEVEL_HIGH>,
    <&liointc1          29          IRQ_TYPE_LEVEL_HIGH>,<&liointc1          29
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      29          IRQ_TYPE_LEVEL_HIGH>,<&liointc1      29
IRQ_TYPE_LEVEL_HIGH>,
    <&liointc1          29          IRQ_TYPE_LEVEL_HIGH>,<&liointc1          29
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      29          IRQ_TYPE_LEVEL_HIGH>,<&liointc1      29
IRQ_TYPE_LEVEL_HIGH>,
    <&liointc1          29          IRQ_TYPE_LEVEL_HIGH>,<&liointc1          29
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      29          IRQ_TYPE_LEVEL_HIGH>,<&liointc1      29
IRQ_TYPE_LEVEL_HIGH>,
    <&liointc1          29          IRQ_TYPE_LEVEL_HIGH>,<&liointc1          29
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      29          IRQ_TYPE_LEVEL_HIGH>,<&liointc1      29
IRQ_TYPE_LEVEL_HIGH>,
    <&liointc1          29          IRQ_TYPE_LEVEL_HIGH>,<&liointc1          29
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      29          IRQ_TYPE_LEVEL_HIGH>,<&liointc1      29
IRQ_TYPE_LEVEL_HIGH>,
    <&liointc1          29          IRQ_TYPE_LEVEL_HIGH>,<&liointc1          29
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      29          IRQ_TYPE_LEVEL_HIGH>,<&liointc1      29
IRQ_TYPE_LEVEL_HIGH>,
    <&liointc1          29          IRQ_TYPE_LEVEL_HIGH>,<&liointc1          29
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      29          IRQ_TYPE_LEVEL_HIGH>,<&liointc1      29
IRQ_TYPE_LEVEL_HIGH>,<&liointc1      29          IRQ_TYPE_LEVEL_HIGH>,<&liointc1      29
};

```

```

can0: can@1ff44000 {
    compatible = "nxp,sga1000";
    reg = <0 0x1ff44000 0 0x1000>;
    nxp,external-clock-frequency = <100000000>;
    interrupt-parent = <&extioic>;
    interrupts = <10>;
};

```

```

can1: can@1ff45000 {
    compatible = "nxp,sga1000";
    reg = <0 0x1ff45000 0 0x1000>;
    nxp,external-clock-frequency = <100000000>;
};

```

```
    interrupt-parent = <&extioic>;
    interrupts = <11>;
};

spi0: spi@0x1fd00000 {
    compatible = "loongson,ls-spi";
    #address-cells = <1>;
    #size-cells = <0>;
    reg = <0 0x1fd00000 0 0x10>;
    spidev@0 {
        compatible = "rohm,dh2228fv";
        spi-max-frequency = <100000000>;
        reg = <0>;
    };
};

spi1: spi@0x1fd40000 {
    compatible = "loongson,ls-spi";
    reg = <0 0x1fd40000 0 0x10>;
    status = "disabled";
};

/* SPI2~5 has only one CS, which is set by SPCS */
spi2: spi@0x1ff50000 {
    compatible = "loongson,ls-spi";
    reg = <0 0x1ff50000 0 0x10>;
    spi-nocs;
    status = "disabled";
};

spi3: spi@0x1ff51000 {
    compatible = "loongson,ls-spi";
    reg = <0 0x1ff51000 0 0x10>;
    spi-nocs;
    status = "disabled";
};

spi4: spi@0x1ff52000 {
    compatible = "loongson,ls-spi";
    reg = <0 0x1ff52000 0 0x10>;
    spi-nocs;
    status = "disabled";
};

spi5: spi@0x1ff53000 {
    compatible = "loongson,ls-spi";
    reg = <0 0x1ff53000 0 0x10>;
    spi-nocs;
    status = "disabled";
};

i2c0: i2c@0x1ff48000 {
    compatible = "loongson,ls2k-i2c";
    #address-cells = <1>;
    #size-cells = <0>;
    reg = <0 0x1ff48000 0 0x0800>;
    interrupt-parent = <&extioic>;
```

```
interrupts = <14>;
eeprom@57 {
    compatible = "atmel,24c16";
    reg = <0x57>;
    pagesize = <16>;
};

i2c1: i2c@0x1ff48800 {
    compatible = "loongson,ls2k-i2c";
    reg = <0 0x1ff48800 0 0x0800>;
    interrupt-parent = <&extioic>;
    interrupts = <15>;
    status = "disabled";
};

i2c2: i2c@0x1ff49000 {
    compatible = "loongson,ls2k-i2c";
    reg = <0 0x1ff49000 0 0x0800>;
    interrupt-parent = <&extioic>;
    interrupts = <16>;
    status = "disabled";
};

i2c3: i2c@0x1ff49800 {
    compatible = "loongson,ls2k-i2c";
    reg = <0 0x1ff49800 0 0x0800>;
    interrupt-parent = <&extioic>;
    interrupts = <17>;
    status = "disabled";
};

i2c4: pxi2c@0x1ff4a000 {
    compatible = "loongson,ls2k-i2c";
    reg = <0 0x1ff4a000 0 0x0800>;
    interrupt-parent = <&extioic>;
    interrupts = <18>;
};

i2c5: pxi2c@0x1ff4a800 {
    compatible = "loongson,ls2k-i2c";
    reg = <0 0x1ff4a800 0 0x0800>;
    interrupt-parent = <&extioic>;
    interrupts = <19>;
};

dma0: dma@0x1fe10c00 {
    compatible = "loongson,loongson2-apbdma";
    reg = <0 0x1fe10c00 0 0x8>;
    loongson,apbdma-conf = <&generic_conf0 0x0 0x0>;
    #dma-cells = <0>;
    dma-channels = <1>;
    interrupt-parent = <&extioic>;
    interrupts = <67>;
};

dma1: dma@0x1fe10c10 {
```

```

compatible = "loongson,loongson2-apbdma";
reg = <0 0x1fe10c10 0 0x8>;
loongson,apbdma-conf = <&generic_conf0 0x0 0x0>;
#dma-cells = <0>;
dma-channels = <1>;
interrupt-parent = <&extioiic>;
interrupts = <68>;
};

dma2: dma@0x1fe10c20 {
compatible = "loongson,loongson2-apbdma";
reg = <0 0x1fe10c20 0 0x8>;
loongson,apbdma-conf = <&generic_conf0 0xc000 0xc000>; /* 0xc000 for sdio1 */
#dma-cells = <0>;
dma-channels = <1>;
interrupt-parent = <&extioiic>;
interrupts = <69>;
};

dma3: dma@0x1fe10c30 {
compatible = "loongson,loongson2-apbdma";
reg = <0 0x1fe10c30 0 0x8>;
loongson,apbdma-conf = <&generic_conf0 0x0 0x0>;
#dma-cells = <0>;
dma-channels = <1>;
interrupt-parent = <&extioiic>;
interrupts = <70>;
};

sdio0@0x1ff64000 {
#address-cells = <2>;
compatible = "loongson,ls2k_sdio";
reg = <0 0x1ff64000 0 0x1000>;
interrupt-parent = <&liointc0>;
interrupts = <31 IRQ_TYPE_LEVEL_HIGH>;
interrupt-names = "ls2k_mci_irq";
dmas = <&dma3>;
dma-names = "sdio_rw";
dma-mask = <0xffffffff 0xffffffff>;
clock-frequency = <0 125000000>;
bus-width = <4>;
cd-debounce-delay-ms = <0>;

/* cd-gpios = <&pioA 44 GPIO_ACTIVE_LOW>; */
/* Use this pin for irq hotplug detect SD card that */
/* need rework hardware and remove non-removable */
non-removable;
/* broken-cd; */
/* broken-cd for polling mode to hotplug detect SD */
/* card but consumes lots of CPU resources */
};

sdio1@0x1ff66000 {
#address-cells = <2>;
compatible = "loongson,ls2k_sdio";
reg = <0 0x1ff66000 0 0x1000>;
interrupt-parent = <&extioiic>;

```

```

interrupts = <58>;
interrupt-names = "ls2k_mci_irq";
dmas = <&dma2>;
dma-names = "sdio_rw";
dma-mask = <0xffffffff 0xffffffff>;
clock-frequency = <0 125000000>;
bus-width = <4>;
cd-debounce-delay-ms = <0>;

cd-gpios = <&pioB 22 GPIO_ACTIVE_LOW>;
status = "disabled";
};

hpet0: hpet@1ff68000 {
    compatible = "loongson,loongson2-hpet";
    reg = <0 0x1ff68000 0 0x1000>;
    clock-frequency = <125000000>;
    interrupt-parent = <&liointc0>;
    interrupts = <21 IRQ_TYPE_LEVEL_HIGH>;
};

rtc0: rtc@0x1ff6c100 {
    compatible = "loongson,ls2k0500-rtc", "loongson,ls7a-rtc";
    reg = <0 0x1ff6c100 0 0x1000>;
    interrupt-parent = <&liointc0>;
    interrupts = <8 IRQ_TYPE_LEVEL_HIGH>;
};

hda@0x1f070000 {
    compatible = "loongson,ls2k-audio";
    reg = <0 0x1f070000 0 0xffff>;
    interrupt-parent = <&liointc0>;
    interrupts = <4 IRQ_TYPE_LEVEL_HIGH>;
    pinctrl-0 = <&hda_default>;
    pinctrl-names = "default";
};

nand@0x1ff58040 {
    compatible = "loongson,ls-nand";
    reg = <0 0x1ff58040 0 0x0
        0 0x1ff58000 0 0x20>;
    interrupt-parent = <&extioic>;
    interrupts = <31>;
    interrupt-names = "nand_irq";
    dmas = <&dma0>;
    dma-names = "nand_rw";
    dma-mask = <0xffffffff 0xffffffff>;
    #address-cells = <2>;
    #size-cells = <2>;
    nand-cs = <0x0>;
    status = "disabled";

    number-of-parts = <0x2>;
    partition@0 {
        label = "kernel_partition";
        reg = <0 0x0000000 0 0x01e00000>;
    };
};

```

```
partition@0x01e00000 {  
    label = "os_partition";  
    reg = <0 0x01e00000 0 0x0>;  
};  
};  
};
```


附录 D 2K1000LA 设备树源码参考配置

```
#include <dt-bindings/thermal/thermal.h>
#include <dt-bindings/interrupt-controller/irq.h>
#include <dt-bindings/gpio/gpio.h>

/dts-v1/;
/ {
    model = "Loongson-2K1000 Reference Board";
    compatible = "loongson,ls2k";
    #address-cells = <2>;
    #size-cells = <2>;

    aliases {
        ethernet0 = &gmac0;
        ethernet1 = &gmac1;
        serial0 = &cpu_uart0;
        serial1 = &uart1;
        serial2 = &uart2;
        serial3 = &uart3;
        i2c0 = &i2c0;
        i2c1 = &i2c1;
        i2c2 = &i2c2;
        i2c3 = &i2c3;
    };

    chosen {
        stdout-path = "serial0:115200n8";
        bootargs = "earlycon";
    };

    memory {
        device_type = "memory";
        reg = <0 0x00200000 0 0x06e00000
            0 0x08000000 0 0x07000000
            0 0x90000000 1 0xe0000000>;
    };

    reserved-memory {
        #address-cells = <2>;
        #size-cells = <2>;
        ranges;

        linux,cma {
            compatible = "shared-dma-pool";
            reusable;
            size = <0x0 0x2000000>;
            linux,cma-default;
        };
    };

    memalloc@0x90000000 {
        compatible = "loongson,ls-memalloc";
    };
};
```

```
reg = <0 0x90000000 0 0x20000000>;
};

cpus {
#address-cells = <1>;
#size-cells = <0>;
cpu-map {
cluster0 {
core0 {
cpu = <&cpu0>;
};
core1 {
cpu = <&cpu1>;
};
};
};

cpu0: cpu@0 {
device_type = "cpu";
compatible = "loongarch";
clock-frequency = <1000000000>;
reg=<0>;
/* Node:kHz DVFS:vid[0:5] */
/* loongson,dvfs-vid-points = <
333333 0x11
400000 0x10
500000 0x01
666666 0x00
1000000 0x00
>; */
};

cpu1: cpu@1 {
device_type = "cpu";
compatible = "loongarch";
clock-frequency = <1000000000>;
reg=<1>;
/* Node:kHz DVFS:vid[0:5] */
/* loongson,dvfs-vid-points = <
333333 0x11
400000 0x10
500000 0x01
666666 0x00
1000000 0x00
>; */
};
};

cpuic: interrupt-controller {
compatible = "loongson,cpu-interrupt-controller";
interrupt-controller;
#interrupt-cells = <1>;
};

liointc0: interrupt-controller@1fe11400 {
compatible = "loongson,liointc-2.0";
reg = <0 0x1fe01400 0 0x40>;
};
```

```

        <0 0x1fe01040 0 0x8>,
        <0 0x1fe01140 0 0x8>;
reg-names = "main", "isr0", "isr1";

interrupt-controller;
#interrupt-cells = <2>;
interrupt-parent = <&cpuic>;
interrupts = <2>;
interrupt-names = "int0";

loongson,parent_int_map = <0xffffffff>, /* int0 */
        <0x00000000>, /* int1 */
        <0x00000000>, /* int2 */
        <0x00000000>; /* int3 */
};

liointc1: interrupt-controller@1fe11440 {
    compatible = "loongson,liointc-2.0";
    reg = <0 0x1fe01440 0 0x40>,
        <0 0x1fe01048 0 0x8>,
        <0 0x1fe01148 0 0x8>;
    reg-names = "main", "isr0", "isr1";

    interrupt-controller;
    #interrupt-cells = <2>;

    interrupt-parent = <&cpuic>;
    interrupts = <3>;
    interrupt-names = "int1";

    loongson,parent_int_map = <0x00000000>, /* int0 */
        <0xffffffff>, /* int1 */
        <0x00000000>, /* int2 */
        <0x00000000>; /* int3 */
};

soc {
    compatible = "ls,nbus", "simple-bus";
    #address-cells = <2>;
    #size-cells = <2>;
    ranges = <0 0x10000000 0 0x10000000 0 0x10000000
        0 0x20000000 0 0x20000000 0 0x20000000
        0 0x20000000 0 0x20000000 0 0x10000000
        0 0x40000000 0 0x40000000 0 0x40000000
        0xfe 0x00000000 0xfe 0x00000000 0 0x40000000>;

    dma-coherent;

    scfg: scfg@1fe00000 {
        compatible = "loongson,ls2k-chipid";
        reg = <0 0x1fe00000 0 0x3ffc>;
        little-endian;
    };

    cpu_uart0: serial@0x1fe20000 {
        compatible = "ns16550a";
        reg = <0 0x1fe20000 0 0x10>;
    };
};

```

```
clock-frequency = <125000000>;
interrupt-parent = <&liointc0>;
interrupts = <0 IRQ_TYPE_LEVEL_HIGH>;
no-loopback-test;
};

uart1: serial@0x1fe20100 {
    compatible = "ns16550a";
    reg = <0 0x1fe20100 0 0x10>;
    clock-frequency = <125000000>;
    interrupt-parent = <&liointc0>;
    interrupts = <0 IRQ_TYPE_LEVEL_HIGH>;
    no-loopback-test;
};

uart2: serial@0x1fe20200 {
    compatible = "ns16550a";
    reg = <0 0x1fe20200 0 0x10>;
    clock-frequency = <125000000>;
    interrupt-parent = <&liointc0>;
    interrupts = <0 IRQ_TYPE_LEVEL_HIGH>;
    no-loopback-test;
};

uart3: serial@0x1fe20300 {
    compatible = "ns16550a";
    reg = <0 0x1fe20300 0 0x10>;
    clock-frequency = <125000000>;
    interrupt-parent = <&liointc0>;
    interrupts = <0 IRQ_TYPE_LEVEL_HIGH>;
    no-loopback-test;
};

pctrl:pinctrl@1fe00420 {
    compatible = "loongson,ls2k-pinctrl";
    reg = <0 0x1fe00420 0 0x18>;

    sdio_default:sdio {
        sdio_mux {
            groups = "sdio";
            function = "sdio";
        };

        sdio_det_mux {
            groups = "pwm2";
            function = "gpio";
        };
    };

    pwm1_default:pwm1 {
        mux {
            groups = "pwm1";
            function = "pwm1";
        };
    };

    pwm0_default:pwm0 {
```

```
        mux {
            groups = "pwm0";
            function = "pwm0";
        };
};

i2c1_default:i2c1 {
    mux {
        groups = "i2c1";
        function = "i2c1";
    };
};

i2c0_default:i2c0 {
    mux {
        groups = "i2c0";
        function = "i2c0";
    };
};

nand_default:nand {
    mux {
        groups = "nand";
        function = "nand";
    };
};
};

pioA:gpio@0x1fe00500 {
    compatible = "loongson,ls2k-gpio";
    reg = <0 0x1fe00500 0 0x38>;
    ngpios = <64>;
    gpio-controller;
    #gpio-cells = <2>;
    interrupt-parent = <&liointc1>;
    interrupts =
        <28 IRQ_TYPE_LEVEL_HIGH>,
        <29 IRQ_TYPE_LEVEL_HIGH>,
        <30 IRQ_TYPE_LEVEL_HIGH>,
        <30 IRQ_TYPE_LEVEL_HIGH>,
        <26 IRQ_TYPE_LEVEL_HIGH>,
        <26 IRQ_TYPE_LEVEL_HIGH>,
        <26 IRQ_TYPE_LEVEL_HIGH>,
        <26 IRQ_TYPE_LEVEL_HIGH>,
        <26 IRQ_TYPE_LEVEL_HIGH>,
        <26 IRQ_TYPE_LEVEL_HIGH>,
        <26 IRQ_TYPE_LEVEL_HIGH>,
        <26 IRQ_TYPE_LEVEL_HIGH>,
        <26 IRQ_TYPE_LEVEL_HIGH>,
        <26 IRQ_TYPE_LEVEL_HIGH>,
        <26 IRQ_TYPE_LEVEL_HIGH>,
        <26 IRQ_TYPE_LEVEL_HIGH>,
        <0 IRQ_TYPE_NONE>,
        <26 IRQ_TYPE_LEVEL_HIGH>,
        <26 IRQ_TYPE_LEVEL_HIGH>,
        <26 IRQ_TYPE_LEVEL_HIGH>,
        <26 IRQ_TYPE_LEVEL_HIGH>,
        <26 IRQ_TYPE_LEVEL_HIGH>,
};
```



```

    mask = <0x1>;
};

poweroff {
    compatible = "syscon-poweroff";
    regmap = <&pmc>;
    offset = <0x14>;
    mask = <0x3c00>;
    value = <0x3c00>;
};

dwc2@0x40000000 {
    compatible = "loongson,loongson2-dwc2";
    reg = <0 0x40000000 0 0x40000>;
    interrupt-parent = <&liointc1>;
    interrupts = <17 IRQ_TYPE_LEVEL_HIGH>;
    dma-mask = <0x0 0xffffffff>;
    dr_mode = "otg";
    /* Optional for that dr_mode = "host" or dr_mode = "peripheral" */
};

ohci@0x40070000 {
    compatible = "generic-ohci";
    reg = <0 0x40070000 0 0x8000>;
    interrupt-parent = <&liointc1>;
    interrupts = <19 IRQ_TYPE_LEVEL_HIGH>;
    /* Need disable or remove this node when ohci can be scanned by pci */
};

ehci@0x40060000 {
    compatible = "generic-ehci";
    reg = <0 0x40060000 0 0x8000>;
    interrupt-parent = <&liointc1>;
    interrupts = <18 IRQ_TYPE_LEVEL_HIGH>;
    /* Need disable or remove this node when ehci can be scanned by pci */
};

i2c2: i2c@1fe21000 {
    compatible = "loongson,ls2k-i2c";
    reg = <0 0x1fe21000 0 0x8>;
    interrupt-parent = <&liointc0>;
    interrupts = <22 IRQ_TYPE_LEVEL_HIGH>;
    #address-cells = <1>;
    #size-cells = <0>;
    pinctrl-0 = <&i2c0_default>;
    pinctrl-names = "default";

    eeprom@57 {
        compatible = "atmel,24c16";
        reg = <0x57>;
        pagesize = <16>;
    };
};

i2c3: i2c@1fe21800 {
    #address-cells = <1>;
    #size-cells = <0>;
};

```

```
compatible = "loongson,ls2k-i2c";
reg = <0 0x1fe21800 0 0x8>;
interrupt-parent = <&liointc0>;
interrupts = <23 IRQ_TYPE_LEVEL_HIGH>;
pinctrl-0 = <&i2c1_default>;
pinctrl-names = "default";

codec@1a {
    compatible = "codec_uda1342";
    reg = <0x1a>;
};

i2c0: i2c-gpio@0 {
    compatible = "i2c-gpio";
    gpios = <&pioA 1 0 /* sda */
        &pioA 0 0 /* scl */
        >;
    /*i2c-gpio,sda-open-drain;*/
    /*i2c-gpio,scl-open-drain;*/
    i2c-gpio,delay-us = <5>;/* ~100 kHz */
    #address-cells = <1>;
    #size-cells = <0>;
    eeprom@50 {
        compatible = "dvi-eeprom-edid";
        reg = <0x50>;
    };
};

i2c1: i2c-gpio@1 {
    compatible = "i2c-gpio";
    gpios = <&pioA 32 0 /* sda */
        &pioA 33 0 /* scl */
        >;
    /*i2c-gpio,sda-open-drain;*/
    /*i2c-gpio,scl-open-drain;*/
    i2c-gpio,delay-us = <5>;/* ~100 kHz */
    #address-cells = <1>;
    #size-cells = <0>;
    eeprom@50 {
        compatible = "eeprom-edid";
        reg = <0x50>;
    };
};

dc@0x400c0000 {
    compatible = "loongson,display-subsystem";
    reg = <0 0x400c0000 0 0x00010000>;
    interrupt-parent = <&liointc0>;
    interrupts = <28 IRQ_TYPE_LEVEL_HIGH>;
    dma-mask = <0x00000000 0xffffffff>;
};

gpu@0x40080000 {
    compatible = "vivante,gc";
    reg = <0 0x40080000 0 0x00040000>;
};
```



```
interrupt-parent = <&liointc0>;
interrupts = <29 IRQ_TYPE_LEVEL_HIGH>;
dma-mask = <0x00000000 0xffffffff>;
};

rtc0: rtc@1fe27800 {
    compatible = "loongson,ls2k1000-rtc", "loongson,ls7a-rtc";
    reg = <0 0x1fe27800 0 0x100>;
    interrupt-parent = <&liointc1>;
    interrupts = <8 IRQ_TYPE_LEVEL_HIGH>;
};

pwm0: pwm@1fe22000 {
    compatible = "loongson,ls2k-pwm";
    reg = <0 0x1fe22000 0 0x10>;
    clock-frequency = <125000000>;
    interrupt-parent = <&liointc0>;
    interrupts = <24 IRQ_TYPE_LEVEL_HIGH>;
    #pwm-cells = <2>;
    pinctrl-0 = <&pwm0_default>;
    pinctrl-names = "default";
};

pwm1: pwm@1fe22010 {
    compatible = "loongson,ls2k-pwm";
    reg = <0 0x1fe22010 0 0x10>;
    clock-frequency = <125000000>;
    interrupt-parent = <&liointc0>;
    interrupts = <25 IRQ_TYPE_LEVEL_HIGH>;
    #pwm-cells = <2>;
    pinctrl-0 = <&pwm1_default>;
    pinctrl-names = "default";
};

gmac0: ethernet@0x40040000 {
    compatible = "snps,dwmac-3.70a";
    reg = <0 0x40040000 0 0x8000>;
    interrupt-parent = <&liointc0>;
    interrupts = <12 IRQ_TYPE_LEVEL_HIGH>,
        <13 IRQ_TYPE_LEVEL_HIGH>;
    interrupt-names = "macirq", "eth_wake_irq";
    mac-address = [ 64 48 48 48 48 60 ];/* [>mac 64:48:48:48:48:60 <]*/
    phy-mode = "rgmii";
    bus_id = <0x0>;
    dma-mask = <0xffffffff 0xffffffff>;
    snps,multicast-filter-bins = <256>;
};

gmac1: ethernet@0x40050000 {
    compatible = "snps,dwmac-3.70a";
    reg = <0 0x40050000 0 0x8000>;
    interrupt-parent = <&liointc0>;
    interrupts = <14 IRQ_TYPE_LEVEL_HIGH>,
        <15 IRQ_TYPE_LEVEL_HIGH>;
    interrupt-names = "macirq", "eth_wake_irq";
    mac-address = [ 64 48 48 48 48 61 ];/* [>mac 64:48:48:48:48:61 <]*/
    phy-mode = "rgmii";
};
```

```
bus_id = <0x1>;
dma-mask = <0xffffffff 0xffffffff>;
snps,multicast-filter-bins = <256>;
};

apbdma_conf: apbdma_conf@1fe00438 {
    compatible = "syscon";
    reg = <0 0x1fe00438 0 0x8>;
};

dma0: dma@1fe00c00 {
    compatible = "loongson,loongson2-apbdma";
    reg = <0 0x1fe00c00 0 0x8>;
    loongson,apbdma-conf = <&apbdma_conf 0x00000007 0x0>; /* configured for nand. */
    #dma-cells = <0>;
    dma-channels = <1>;
};

dma1: dma@1fe00c10 {
    compatible = "loongson,loongson2-apbdma";
    reg = <0 0x1fe00c10 0 0x8>;
    loongson,apbdma-conf = <&apbdma_conf 0x00038000 0x8000>; /* configured for sdio. */
    #dma-cells = <0>;
    dma-channels = <1>;
};

dma2: dma@1fe00c20 {
    compatible = "loongson,loongson2-apbdma";
    reg = <0 0x1fe00c20 0 0x8>;
    loongson,apbdma-conf = <&apbdma_conf 0x001c0000 0x80000>; /* configured for i2s. */
    #dma-cells = <0>;
    dma-channels = <1>;
};

dma3: dma@1fe00c30 {
    compatible = "loongson,loongson2-apbdma";
    reg = <0 0x1fe00c30 0 0x8>;
    loongson,apbdma-conf = <&apbdma_conf 0x00e00000 0x600000>; /* configured for i2s. */
    #dma-cells = <0>;
    dma-channels = <1>;
};

dma4: dma@1fe00c40 {
    compatible = "loongson,loongson2-apbdma";
    loongson,apbdma-conf = <&apbdma_conf 0x0 0x0>;
    reg = <0 0x1fe00c40 0 0x8>;
    #dma-cells = <0>;
    dma-channels = <1>;
};

sdio@0x1fe2c000 {
    #address-cells = <2>;
    compatible = "loongson,ls2k_sdio";
    reg = <0 0x1fe2c000 0 0x1000>;
    interrupt-parent = <&liointc0>;
    interrupts = <31 IRQ_TYPE_LEVEL_HIGH>;
    interrupt-names = "ls2k_mci_irq";
};
```

```

clock-frequency = <0 125000000>;
bus-width = <4>;
cd-debounce-delay-ms = <0>;

cd-gpio = <&pioA 22 GPIO_ACTIVE_LOW>;
dmas = <&dma1>;
dma-names = "sdio_rw";
dma-mask = <0xffffffff 0xffffffff>;
pinctrl-0 = <&sdio_default>;
pinctrl-names = "default";
};

spi0: spi@1fff0220 {
    compatible = "loongson,ls-spi";
    #address-cells = <1>;
    #size-cells = <0>;
    reg = <0 0x1fff0220 0 0x10>;
    spi-dev@0 {
        compatible = "rohm,dh2228fv";
        spi-max-frequency = <100000000>;
        reg = <0>;
    };
};

hpet0: hpet@1fe24000 {
    compatible = "loongson,loongson2-hpet";
    reg = <0 0x1fe24000 0 0x15f>;
    clock-frequency = <125000000>;
    interrupt-parent = <&liointc0>;
    interrupts = <21 IRQ_TYPE_LEVEL_HIGH>;
};

nand@0x1fe26040 {
    #address-cells = <2>;
    compatible = "loongson,ls-nand";
    reg = <0 0x1fe26040 0 0x0
        0 0x1fe26000 0 0x20>;
    interrupt-parent = <&liointc1>;
    interrupts = <12 IRQ_TYPE_LEVEL_HIGH>;
    interrupt-names = "nand_irq";
    pinctrl-0 = <&nand_default>;
    pinctrl-names = "default";
    nand-cs = <0x2>;
    dmas = <&dma0>;
    dma-names = "nand_rw";
    dma-mask = <0xffffffff 0xffffffff>;

    number-of-parts = <0x2>;
    partitions {
        compatible = "fixed-partitions";
        #address-cells = <2>;
        #size-cells = <2>;

        partition@0 {
            label = "kernel_partition";
            reg = <0 0x00000000 0 0x01400000>;
        };
    };
};

```

```

        partition@0x01400000 {
            label = "os_partition";
            reg = <0 0x01400000 0 0x0>;
        };
    };
};

/* CAN controller nodes:
 * If you want to use the "can" function,enable the "can"
 * controller by configure general configuration register 0.
 */
/*
can0: can@1fe20c00 {
    compatible = "nxp,sga1000";
    reg = <0 0x1fe20c00 0 0xff>;
    nxp,external-clock-frequency = <125000000>;
    interrupt-parent = <&liointc0>;
    interrupts = <16 IRQ_TYPE_LEVEL_HIGH>;
};
can1: can@1fe20d00 {
    compatible = "nxp,sga1000";
    reg = <0 0x1fe20d00 0 0xff>;
    nxp,external-clock-frequency = <125000000>;
    interrupt-parent = <&liointc0>;
    interrupts = <17 IRQ_TYPE_LEVEL_HIGH>;
};*/

tsensor: tsensor@1fe01500 {
    compatible = "loongson,ls2k-tsensor";
    reg = <0 0x1fe01500 0 0x30>;
    id = <0>;
    interrupt-parent = <&liointc0>;
    interrupts = <7 IRQ_TYPE_LEVEL_HIGH>;
    #thermal-sensor-cells = <1>;
};

fan0: pwm-fan {
    compatible = "pwm-fan";
    cooling-min-state = <0>;
    cooling-max-state = <3>;
    #cooling-cells = <2>;
    pwms = <&pwm1 0 100000 0>;
    cooling-levels = <255 153 85 25>;
    /* The duty cycle is defined in reverse in 2k1000la */
};

thermal-zones {
    cpu_thermal: cpu-thermal {
        polling-delay-passive = <1000>;
        polling-delay = <5000>;
        thermal-sensors = <&tsensor 0>;

        trips {
            cpu_alert: cpu-alert {
                temperature = <33000>;
                hysteresis = <2000>;
                type = "active";
            };
        };
    };
};

```

```

};

cpu_crit: cpu-crit {
    temperature = <85000>;
    hysteresis = <5000>;
    type = "critical";
};

};

cooling-maps {
    map0 {
        trip = <&cpu_alert>;
        cooling-device =
            <&fan0 THERMAL_NO_LIMIT
            THERMAL_NO_LIMIT>;
    };
};

};

pcie@0 {
    compatible = "loongson,ls2k-pci";
    device_type = "pci";
    bus-range = <0x1 0x16>;
    #size-cells = <2>;
    #address-cells = <3>;

    reg = < 0xfe 0x00000000 0 0x20000000>;
    ranges = <0x2000000 0x0 0x60000000 0 0x60000000 0x0 0x20000000 /* mem */
        0x01000000 0 0x00008000 0 0x18008000 0x0 0x8000>;

    ehci {
        reg = <0x2100 0x0 0x0 0x0 0x0>;
        interrupt-parent = <&liointc1>;
        interrupts = <18 IRQ_TYPE_LEVEL_HIGH>;
        status = "disabled";
    };

    ohci {
        reg = <0x2200 0x0 0x0 0x0 0x0>;
        interrupt-parent = <&liointc1>;
        interrupts = <19 IRQ_TYPE_LEVEL_HIGH>;
        status = "disabled";
    };

    dwc2@0,4,0 {
        reg = <0x2000 0x0 0x0 0x0 0x0>;
        status = "disabled";
    };

    apb@0,2,0 {
        reg = <0x1000 0x0 0x0 0x0 0x0>;
        status = "disabled";
    };

    gmac0@0,3,0 {
        reg = <0x1800 0x0 0x0 0x0 0x0>;
        interrupt-parent = <&liointc0>;

```

```
    interrupts = <12 IRQ_TYPE_LEVEL_HIGH>,<13 IRQ_TYPE_LEVEL_HIGH>;
    status = "disabled";
};

gmac1@0,3,1 {
    reg = <0x1900 0x0 0x0 0x0 0x0>;
    interrupt-parent = <&liointc0>;
    interrupts = <14 IRQ_TYPE_LEVEL_HIGH>,<15 IRQ_TYPE_LEVEL_HIGH>;
    status = "disabled";
};

gpu@0,5,0 {
    reg = <0x2800 0x0 0x0 0x0 0x0>;
    status = "disabled";
};

dc@0,6,0 {
    reg = <0x3000 0x0 0x0 0x0 0x0>;
    status = "disabled";
};

hda@0,7,0 {
    reg = <0x3800 0x0 0x0 0x0 0x0>;
    interrupt-parent = <&liointc0>;
    interrupts = <4 IRQ_TYPE_LEVEL_HIGH>;
};

ahci@0,8,0 {
    reg = <0x4000 0x0 0x0 0x0 0x0>;
    interrupt-parent = <&liointc0>;
    interrupts = <19 IRQ_TYPE_LEVEL_HIGH>;
};

pcie0_port0: pci_bridge@9,0 {
    compatible = "pciclass060400",
                "pciclass0604";
    reg = <0x4800 0x0 0x0 0x0 0x0>;
    interrupts = <0 IRQ_TYPE_LEVEL_HIGH>;
    interrupt-parent = <&liointc1>;

    #interrupt-cells = <1>;
    interrupt-map-mask = <0 0 0 0>;
    interrupt-map = <0 0 0 0 &liointc1 0 IRQ_TYPE_LEVEL_HIGH>;
};

pcie0_port1: pci_bridge@10,0 {
    compatible = "pciclass060400",
                "pciclass0604";
    reg = <0x5000 0x0 0x0 0x0 0x0>;
    interrupts = <1 IRQ_TYPE_LEVEL_HIGH>;
    interrupt-parent = <&liointc1>;

    #interrupt-cells = <1>;
    interrupt-map-mask = <0 0 0 0>;
    interrupt-map = <0 0 0 0 &liointc1 1 IRQ_TYPE_LEVEL_HIGH>;
};
```


附录 E 2K2000 设备树源码参考配置

```
#include <dt-bindings/interrupt-controller/irq.h>

/dts-v1/;
/ {
    model = "Loongson-2K2000 Reference Board";
    compatible = "loongson,ls2k2000";
    #address-cells = <2>;
    #size-cells = <2>;

    aliases {
        serial0 = &cpu_uart0;
    };

    chosen {
        stdout-path = "serial0:115200n8";
        bootargs = "earlycon";
    };

    memory {
        device_type = "memory";
        reg = <0 0x00200000 0 0x0ee00000
            0 0x90000000 1 0xe0000000>;
    };

    reserved-memory {
        #address-cells = <2>;
        #size-cells = <2>;
        ranges;

        linux,cma {
            compatible = "shared-dma-pool";
            reusable;
            size = <0x0 0x2000000>;
            linux,cma-default;
        };
    };

    cpus {
        #address-cells = <1>;
        #size-cells = <0>;
        cpu-map {
            cluster0 {
                core0 {
                    cpu = <&cpu0>;
                };
                core1 {
                    cpu = <&cpu1>;
                };
            };
        };
    };
};
```



```

cpu0: cpu@1 {
    device_type = "cpu";
    compatible = "loongarch";
    clock-frequency = <1400000000>;
    reg=<0>;
};
cpu1: cpu@2 {
    device_type = "cpu";
    compatible = "loongarch";
    clock-frequency = <1400000000>;
    reg=<1>;
};
};

cpuic: interrupt-controller {
    compatible = "loongson,cpu-interrupt-controller";
    interrupt-controller;
    #interrupt-cells = <1>;
};

iointc: interrupt-controller@1fe01400 {
    compatible = "loongson,liointc-2.0";
    reg = <0 0x1fe01400 0 0x40>,
        <0 0x1fe01440 0 0x8>,
        <0 0x1fe01448 0 0x8>;
    reg-names = "main", "isr0", "isr1";

    interrupt-controller;
    #interrupt-cells = <1>;
    interrupt-parent = <&cpuic>;
    interrupts = <2>;
    interrupt-names = "int0";

    loongson,parent_int_map = <0xffffffff>, /* int0 */
        <0x00000000>, /* int1 */
        <0x00000000>, /* int2 */
        <0x00000000>; /* int3 */
};

extioiic: interrupt-controller@0x1fe01600 {
    compatible = "loongson,ls2k2000-eiointc";
    interrupt-controller;
    #interrupt-cells = <1>;
    interrupt-parent = <&cpuic>;
    interrupts = <3>;
};

platic: interrupt-controller@0x10000040 {
    compatible = "loongson,pch-pic-1.0";
    reg = <0x0 0x10000000 0x0 0x400>;
    interrupt-controller;
    interrupt-parent = <&extioiic>;
    loongson,pic-base-vec = <0>;
    #interrupt-cells = <2>;
};

msi: interrupt-controller@0x1fe01140 {

```

```

compatible = "loongson,pch-msi-1.0";
reg = <0 0x1fe01140 0 0x8>;
interrupt-controller;
loongson,msi-base-vec = <64>;
loongson,msi-num-vecs = <192>;
interrupt-parent = <&extioiic>;
};

soc {
compatible = "ls,nbus", "simple-bus";
#address-cells = <2>;
#size-cells = <2>;
ranges = <0 0x10000000 0 0x10000000 0 0x10000000
0 0x2000000 0 0x2000000 0 0x2000000
0 0x40000000 0 0x40000000 0 0x40000000
0xfe 0x00000000 0xfe 0 0 0x40000000>;
pcie@0 {
compatible = "loongson,ls2k-pci";
device_type = "pci";
bus-range = <0x1 0x10>;
#size-cells = <2>;
#address-cells = <3>;
msi-parent = <&msi>;

reg = <0xfe 0x00000000 0 0x20000000>;
ranges = <0x02000000 0 0x40000000 0 0x40000000 0 0x40000000
0x01000000 0 0x00008000 0 0x18400000 0x0 0x8000>;

xhci0: usb@0,4,0 {
reg = <0x2000 0x0 0x0 0x0 0x0>;
interrupts = <48 IRQ_TYPE_LEVEL_HIGH>;
interrupt-parent = <&platic>;
};

gmac0@0,3,0 {
reg = <0x1800 0x0 0x0 0x0 0x0>;
interrupts = <12 IRQ_TYPE_LEVEL_HIGH>;
interrupt-parent = <&platic>;
};

gmac1@0,3,1 {
reg = <0x1900 0x0 0x0 0x0 0x0>;
interrupts = <14 IRQ_TYPE_LEVEL_HIGH>;
interrupt-parent = <&platic>;
};

gmac2@0,3,2 {
reg = <0x1a00 0x0 0x0 0x0 0x0>;
interrupts = <17 IRQ_TYPE_LEVEL_HIGH>;
interrupt-parent = <&platic>;
};

xhci1: usb@0,25,0 {
reg = <0xc800 0x0 0x0 0x0 0x0>;
interrupts = <22 IRQ_TYPE_LEVEL_HIGH>;
interrupt-parent = <&platic>;
};

```

```
dc@0,6,1 {
    reg = <0x3100 0x0 0x0 0x0 0x0>;
    interrupts = <28 IRQ_TYPE_LEVEL_HIGH>;
    interrupt-parent = <&platic>;
};

hda@0,7 {
    reg = <0x3800 0x0 0x0 0x0 0x0>;
    interrupts = <58 IRQ_TYPE_LEVEL_HIGH>;
    interrupt-parent = <&platic>;
};

ahci@0,8 {
    reg = <0x4000 0x0 0x0 0x0 0x0>;
    interrupts = <16 IRQ_TYPE_LEVEL_HIGH>;
    interrupt-parent = <&platic>;
};

pci_bridge@0,9 {
    compatible = "pciclass060400",
                "pciclass0604";
    reg = <0x4800 0x0 0x0 0x0 0x0>;
    interrupts = <32 IRQ_TYPE_LEVEL_HIGH>;
    interrupt-parent = <&platic>;

    #interrupt-cells = <1>;
    interrupt-map-mask = <0 0 0 0>;
    interrupt-map = <0 0 0 0 &platic 32 IRQ_TYPE_LEVEL_HIGH>;
};

pci_bridge@0,0xa {
    compatible = "pciclass060400",
                "pciclass0604";
    reg = <0x5000 0x0 0x0 0x0 0x0>;
    interrupts = <33 IRQ_TYPE_LEVEL_HIGH>;
    interrupt-parent = <&platic>;

    #interrupt-cells = <1>;
    interrupt-map-mask = <0 0 0 0>;
    interrupt-map = <0 0 0 0 &platic 33 IRQ_TYPE_LEVEL_HIGH>;
};

pci_bridge@0,0xb {
    compatible = "pciclass060400",
                "pciclass0604";
    reg = <0x5800 0x0 0x0 0x0 0x0>;
    interrupts = <34 IRQ_TYPE_LEVEL_HIGH>;
    interrupt-parent = <&platic>;

    #interrupt-cells = <1>;
    interrupt-map-mask = <0 0 0 0>;
    interrupt-map = <0 0 0 0 &platic 34 IRQ_TYPE_LEVEL_HIGH>;
};

pci_bridge@0,0xc {
    compatible = "pciclass060400",
```

```
        "pciclass0604";
        reg = <0x6000 0x0 0x0 0x0 0x0>;
        interrupts = <35 IRQ_TYPE_LEVEL_HIGH>;
        interrupt-parent = <&platic>;

        #interrupt-cells = <1>;
        interrupt-map-mask = <0 0 0 0>;
        interrupt-map = <0 0 0 0 &platic 35 IRQ_TYPE_LEVEL_HIGH>;
};

pci_bridge@0,13 {
    compatible = "pciclass060400",
                "pciclass0604";
    reg = <0x6800 0x0 0x0 0x0 0x0>;
    interrupts = <36 IRQ_TYPE_LEVEL_HIGH>;
    interrupt-parent = <&platic>;

    #interrupt-cells = <1>;
    interrupt-map-mask = <0 0 0 0>;
    interrupt-map = <0 0 0 0 &platic 36 IRQ_TYPE_LEVEL_HIGH>;
};

pci_bridge@0,14 {
    compatible = "pciclass060400",
                "pciclass0604";
    reg = <0x7000 0x0 0x0 0x0 0x0>;
    interrupts = <37 IRQ_TYPE_LEVEL_HIGH>;
    interrupt-parent = <&platic>;

    #interrupt-cells = <1>;
    interrupt-map-mask = <0 0 0 0>;
    interrupt-map = <0 0 0 0 &platic 37 IRQ_TYPE_LEVEL_HIGH>;
};

pci_bridge@0,0xf {
    compatible = "pciclass060400",
                "pciclass0604";
    reg = <0x7800 0x0 0x0 0x0 0x0>;
    interrupts = <40 IRQ_TYPE_LEVEL_HIGH>;
    interrupt-parent = <&platic>;

    #interrupt-cells = <1>;
    interrupt-map-mask = <0 0 0 0>;
    interrupt-map = <0 0 0 0 &platic 40 IRQ_TYPE_LEVEL_HIGH>;
};

pci_bridge@0,0x10 {
    compatible = "pciclass060400",
                "pciclass0604";
    reg = <0x8000 0x0 0x0 0x0 0x0>;
    interrupts = <30 IRQ_TYPE_LEVEL_HIGH>;
    interrupt-parent = <&platic>;

    #interrupt-cells = <1>;
    interrupt-map-mask = <0 0 0 0>;
    interrupt-map = <0 0 0 0 &platic 30 IRQ_TYPE_LEVEL_HIGH>;
};
```

```
rio0@0,0x18 {
    reg = <0xc000 0x0 0x0 0x0 0x0>;
    interrupts = <3 IRQ_TYPE_LEVEL_HIGH>;
    interrupt-parent = <&platic>;
};

rio1@0,0x1b {
    reg = <0xd800 0x0 0x0 0x0 0x0>;
    interrupts = <2 IRQ_TYPE_LEVEL_HIGH>;
    interrupt-parent = <&platic>;
};

};

pmc: acpi@0x100d0000 {
    compatible = "loongson,ls2k2000-pmc", "syscon";
    reg = <0x0 0x100d0000 0x0 0x58>;
    interrupt-parent = <&extioiic>;
    interrupts = <47>;
    loongson,suspend-address = <0x0 0x1c000500>;
};

rtc0: rtc@0x100d0100 {
    compatible = "loongson,ls2k2000-rtc", "loongson,ls7a-rtc";
    reg = <0x0 0x100d0100 0x0 0x100>;
    interrupts = <44>;
    interrupt-parent = <&extioiic>;
};

reboot {
    compatible = "syscon-reboot";
    regmap = <&pmc>;
    offset = <0x30>;
    mask = <0x1>;
};

poweroff {
    compatible = "syscon-poweroff";
    regmap = <&pmc>;
    offset = <0x14>;
    mask = <0x3c00>;
    value = <0x3c00>;
};

cpu_uart0: serial@0x1fe001e0 {
    #device_type = "serial";
    compatible = "ns16550a";
    reg = <0 0x1fe001e0 0 0x10>;
    clock-frequency = <100000000>;
    interrupt-parent = <&iointc>;
    interrupts = <10>;
    no-loopback-test;
};

};
```

龙芯 2K 系列 SoC 集显嵌入式软硬件设计规范

V1.0

版本信息

文档信息	文档名	龙芯 2K 系列 SoC 集显嵌入式软硬件设计规范
	版本号	V1.0
	创建人	系统研发部
版本历史		
序号	版本号	更新内容
1	V1.0	发布文档初始版本 V1.0 版。

目 录

1. 概述	1
1.1 适用范围	1
1.2 术语与定义	1
1.3 龙芯 2K 系列 SoC 集显设备树传参规范概述	1
2. 多屏显示方案	3
2.1 透明视频转换芯片方案	3
2.2 非透明视频转换芯片方案	4
2.3 液晶屏显示配置方案	4
2.4 LS2K0500 显示方案	5
2.5 多屏显示配置案例	6
3. 设备树传参规范 A	9
3.1 设备树传参规范 A	10
3.2 传递 EDID 的配置方法	12
3.3 传递显示时序的配置方法	12
3.4 关闭某一路显示管线的方法	16
3.5 硬件克隆显示设备树配置方法	17
3.6 背光配置方法	17
4. 设备树传参规范 B	20
4.1 设备树传参规范 B	20
4.2 LT8618 编码器设备树实现示例	22
4.3 SII9022 编码器设备树实现示例	23
4.4 透明类型编码器的实现示例	24
4.5 液晶屏显示设备实现示例	25
5. 参考文献	25

1. 概述

1.1 适用范围

本文档适用于 2k1000LA 和 2k0500 型号处理器。

1.2 术语与定义

本规范所涉及到的术语及其定义如下：

- 1) **DT**：设备树 (Device Tree) 的简称。
- 2) **BIOS**：Basic Input Output System 的简写，是一组固化到主板 ROM 芯片上的程序，也被称为固件。它保存着系统配置信息、计算机基本输入输出设置程序、开机上点后自检和内存训练等程序。对于龙芯嵌入式 SoC 来说，固件通常指的是 LOONGSON 公司维护的 PMON。
- 3) **Encoder/Transmitter/Transcoder/Display bridge**：都指显示编码器 (视频转换芯片)，用于将 DVO (RGB888) 信号转换为其它显示信号 (比如 LVDS、HDMI、DVI、VGA 和 eDP 等)。不同的厂商和芯片手册用不同的名字，本文中出现的视频转换芯片等术语也是这里的显示编码器的意思。
- 4) **Connector**：显示连接器，也被称为显示接口。常见的显示器为 VGA、DVI、HDMI 和 DP 等显示接口，而常用液晶屏多为 eDP，LVDS 或 DPI 等显示接口。
- 5) **EDID**：Extended display identification data，扩展显示器识别数据。是 VESA 制定的有关显示器识别数据的标准。EDID 通常存储在显示器的 EEPROM 中，EDID1.0 到 EDID1.3 规范中 EDID 的容量都是 128 字节。
- 6) **DDC**：Display Data Channel，显示数据通道。显示控制器驱动通过 DDC 探测显示器是否存在以及从显示器中获得 EDID 数据。
- 7) **DC**：Display Controller，显示控制器。
- 8) **LSDC**：龙芯 2K 系列 SoC 中显示控制器的 Linux 设备驱动程序，代码位于龙芯产品 Linux 内核 /drivers/gpu/drm/lsdc 目录下，它主要负责对龙芯 UMA 构架的芯片进行支持，比如 LS2K1000LA 和 LS2K0500。
- 9) **GPU**：图形处理单元，负责图形的渲染，本规范中如不加特殊说明指的是龙芯 2K 系列 SoC 中的 GC1000 GPU。
- 10) **显示管线**：一条显示管线包括一个 CRTC、一个 Encoder、一个 Connector、一个 I2C 设备等成分 (I2C 是可选的，必要时可以通过在设备树中提供显示时序参数的方式代替)。龙芯 2K 系列 SoC 具备两条独立的显示管线，支持双屏扩展和克隆。下文中出现的显示输出也是显示管线的简化说法。

1.3 龙芯 2K 系列 SoC 集显设备树传参规范概述

龙芯 2K 系列 SoC 中集成显示控制器 (DC) 和图形处理器 (GPU)，部分型号的 SoC 芯片内部中还集成了显示编码器。龙芯 2K 系列 SoC 芯片内部具备两组独立的显示管线，LS2K1000LA 中集成双路 DVO

显示输出，而 LS2K0500 集成了一路 DVO 输出接口和一路 VGA 输出接口。DVO 输出接口除了接 DPI (RGB888) 接口的液晶屏幕之外，一般需要和外部显示编码器搭配使用才能形成完整的显示方案。LS2K1000LA/LS2K0500 和不同的编码器或液晶屏幕组合，可以形成多种显示方案，本规范第二章将龙芯 4.19 产品内核已经支持的显示方案进行详尽说明，硬件工程师请参看本规范的第二章节。

用户可根据产品需求来进行硬件设计，包括外部显示编码器选型，是否需连接液晶屏幕以及是否需要双路显示输出等。由于 Linux 内核无法探测平台类型的硬件设备，所以需要使用设备树来描述设备属性，硬件设备之间的连接关系，与设备相匹配的驱动、设备驱动配置参数。并将这些信息传递给 Linux 内核。为了不限制嵌入式应用的灵活性，龙芯集显设备驱动 (LSDC) 支持了两种设备树传参方式，在本规范文档中称之为设备树传参方式 A 和设备树传参方式 B。

龙芯 4.19 内核中使用 LSDC 来作为嵌入式平台的显示驱动，LSDC 显示驱动支持 A 和 B 两种设备树传参方案。设备树传参方案 A 可用于透明编码器、DPI (RGB888) 接口的液晶屏的方案，支持通过设备树传 EDID、显示时序、使用硬件克隆模式等功能。设备树传参方式 B 支持透明类型和非透明类型的编码器的硬件方案，设备树传参方式 B 基于 DRM Bridge 框架来实现编码器驱动，每个编码器驱动都是独立的设备驱动模块，设备驱动程序能根据设备树所传递的拓扑信息在运行时绑定编码器驱动模块，实现与 LSDC 设备驱动解耦。是符合上游内核社区设备树传参规范的方式。

设备树传参方式 A 和设备树传参方式 B 不同之处在于它们描述显示编码器和显示连接器的方式的不同，对于龙芯 2K 系列 SoC 本身的显示控制器设备节点和 GPU 设备节点，二者在描述方式上并无差别。

<pre>dc@0x400c0000 { compatible = "loongson, display-subsystem"; output-ports = <&dvo0 &dvo1>; dvo0: dvo@0 { reg = <0>; /* 0 for display pipe 0 */ }; dvo1: dvo@1 { reg = <1>; /* 1 for display pipe 1 */ }; };</pre>	<pre>dc@0x400c0000 { compatible = "loongson, display-subsystem"; ports { port@0 { reg = <0>; /* 0 for display pipe 0 */ }; port@1 { reg = <1>; /* 0 for display pipe 0 */ }; }; };</pre>
--	--

图 1-1: 设备树传参方式 A 和设备树传参方式 B 的书写对比

龙芯 4.19 产品内核中的 LSDC 设备驱动虽然支持了两种设备树传参规范，但是初始化时只会选择一种传参规范来解析设备树，所以实际产品的设备树需要开发者根据需求选用其中一种规范来书写设备树文件。LSDC 设备驱动区分设备树类型的方法为：在设备树的显示控制器设备节点中实现 output-ports 及其相关属性，表示用设备树传参方式 A。在设备树的显示控制器设备节点中实现 ports 设备节点及其子节点，表示用设备树传参方式 B。如图 1-1 中的设备树所示，dc@0x400c0000 代表显示控制器的设备树节点，图 1-1 中左边文本框 dc@0x400c0000 设备节点中有 output-ports = <&dvo0 &dvo1>代码且没有 ports 属性，这种情况使用的就是设备树传参方式 A；图 1-1 中右半边文本框 dc@0x400c0000 设备节点中存在 ports 属性，代表使用的是设备树传参方式 B。

LSDC 设备驱动程序在初始化时，**优先检测**设备树显示控制器的设备节点中是否存在 ports 属性，如果

存在则用设备树传参方式 B 解析设备树；如果显示控制器的设备节点中没有 ports 属性则检测 output-ports 属性，如果存在就用设备树传参方式 A 去解析设备树。

本规范第三章着重介绍设备树传参规范 A 规范的实现方式。本规范第四章介绍设备树传参规范 B 的实现方式。龙芯 2K 系列 SoC 的开发者需要根据需求选择一种传参方式。如果用非透明类型的显示编码器，推荐用设备树传参方式 B 来传参。固件维护工程师、编码器驱动开发工程师、设备树编写者需要参看本规范的第三、四章节。

2. 多屏显示方案

龙芯 2K 系列 SoC 芯片内部具备两组独立的显示控制器，LS2K1000LA 集成双路 DVO 显示输出，而 LS2K0500 内部集成了一路 DVO 接口和一路 VGA 接口。DVO 输出接口除了接 RGB888 接口的液晶屏幕之外，一般需要和外部显示编码器搭配使用才能形成完整的显示方案。其中显示编码器分为透明类型的显示编码器和非透明类编码器，透明类的显示编码器指的是不需要程序配置的视频转换芯片。比如 ADV7123、ADV7125、GM7123C 等 DVO 转 VGA 视频转换芯片。非透明类编码器指的是需要写驱动程序配置的编码器芯片，比如 LT8618、SH9022、CH7033B 等芯片，对于这类芯片应该根据显示编码器芯片手册推荐的电路图选择最佳的显示配置方案。

龙芯 2K 系列 SoC 芯片由于具备 DVO 输出接口，可支持多种不同类型的显示编码器和硬件连接方案。本章 2.1、2.2、2.3、2.4 和 2.5 小节将介绍 4 种基础的多屏显示配置方案，实际的板卡基本上就是这四种显示方案的变种或组合。2.6 小节列举出几款由龙芯中科设计的板卡的显示配置方案框图，作为参考介绍给读者。

2.1 透明视频转换芯片方案

在选用透明输出类型的显示编码器转换芯片时，不需要在内核态为转换芯片实现驱动。由于透明输出类型的显示编码器转换芯片只具备视频信号转换的功能不具备 I2C 通信功能。电路布线时，某一路显示连接器的 I2C (SDA, SCL) 引脚直接与这路显示管线所占用的 I2C 设备相连接，两路显示输出均采用透明类型的编码器的显示配置方案如图 2-1 所示。

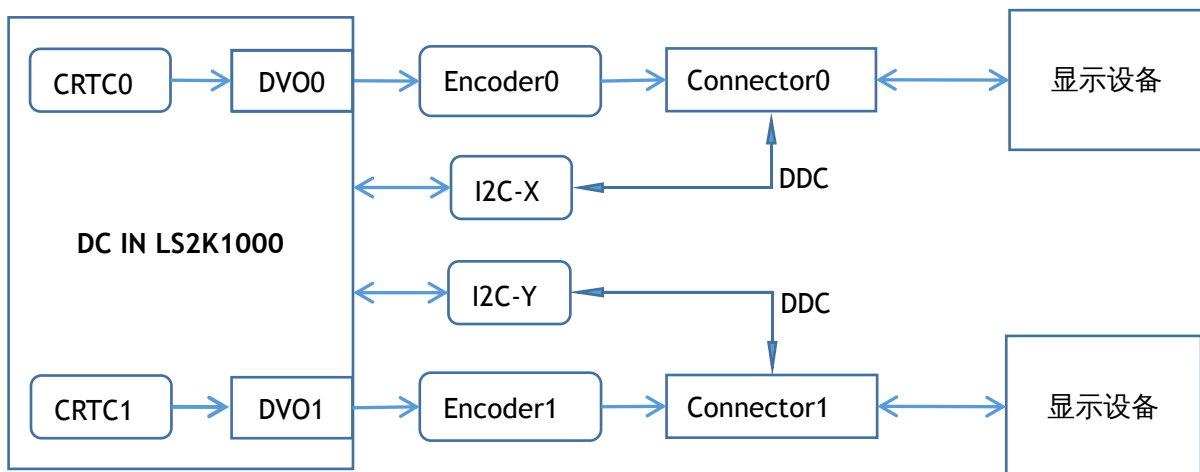


图 2-1：由透明输出类型的显示编码器所组成的显示连接方案

虽然是透明视频转换芯片，不代表芯片是完全不受控制的。比如 ADV7125 编码器具备功耗控制和硬件复位等功能管脚。在待机功耗和硬件复位功能有要求的场景下，可以连接到龙芯 2K 系列 SoC 的 GPIO 上，并且在设备树中描述连接信息，传递给设备驱动，软件层面可以对视频转换芯片进行控制。

2.2 非透明视频转换芯片方案

在非透明类型的显示编码转换芯片时，需要开发内核态编码器芯片驱动，芯片才能正常工作。绝大多数的非透明视频转换芯片都可以通过 I2C 接口来对编码器进行配置，两路输出均选用非透明的编码器的显示方案如图 2-2 所示。

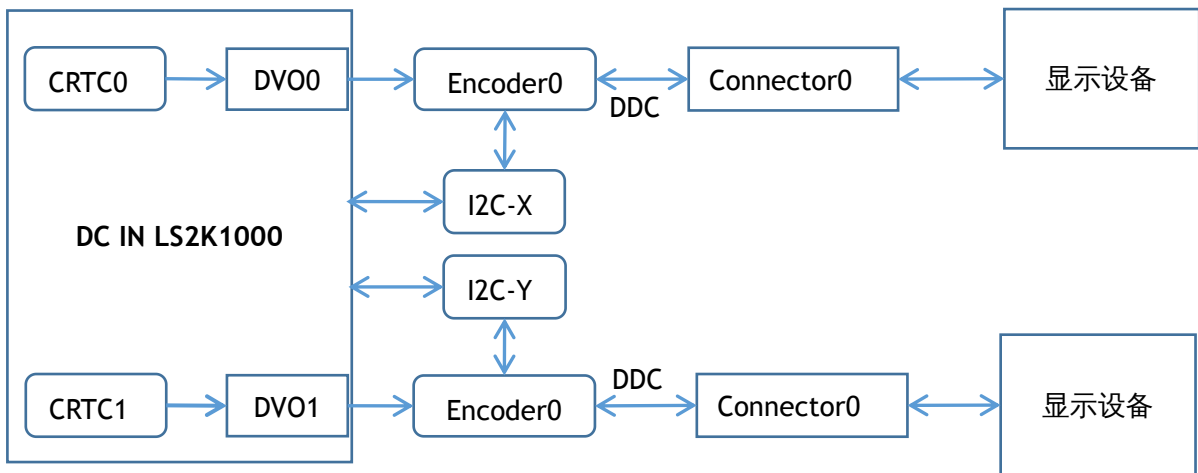


图 2-2: 非透明输出类型的显示编码器的显示连接方案

在主板设计时将视频转换芯片的 I2C Master 通信接口连接到显示连接器（VGA，DVI，HDMI 等接口）的 I2C 引脚，将 I2C Slave 通信接口连接到该条显示管线所占用的 I2C 设备（这里指龙芯 SoC 中的 I2C 设备，可以是硬件 I2C 或 GPIO 模拟 I2C）。显示编码转换芯片作为中间媒介将 EDID 数据从显示器读回，再把 EDID 送回给显示驱动。两条显示管线应该各自占用独立的 I2C 硬件功能单元，应尽量避免其他 I2C slave 设备与图形驱动共用硬件 I2C 的设计，如果 I2C 硬件功能单元不够用的情况，可以考虑使用通用 GPIO 资源来模拟 I2C。如果硬件设计无法避免共用硬件 I2C 的情况，同一个 I2C 总线下的 I2C 设备从地址应该不同，避免通信冲突。

非透明的显示编码器除具备 DDC 功能外，还具备热插拔中断转发的功能。需要把显示连接器的热插拔中断功能管脚连接到显示编码器的热插拔中断输入引脚上，再将显示编码器的热插拔中断输出引脚连接到 LS2K1000LA SoC 的 GPIO 上。

2.3 液晶屏显示配置方案

在使用 DPI 接口的液晶屏幕时，液晶屏幕接口可通过 FPC 排线接到 LS2K1000LA 的 DVO 输出接口上，无需视频编码转换芯片，适合低成本嵌入式类应用。这里以龙芯派的例子，给出了一种使用液晶屏幕显示方案的系统原理框图，如图 2-3 所示。

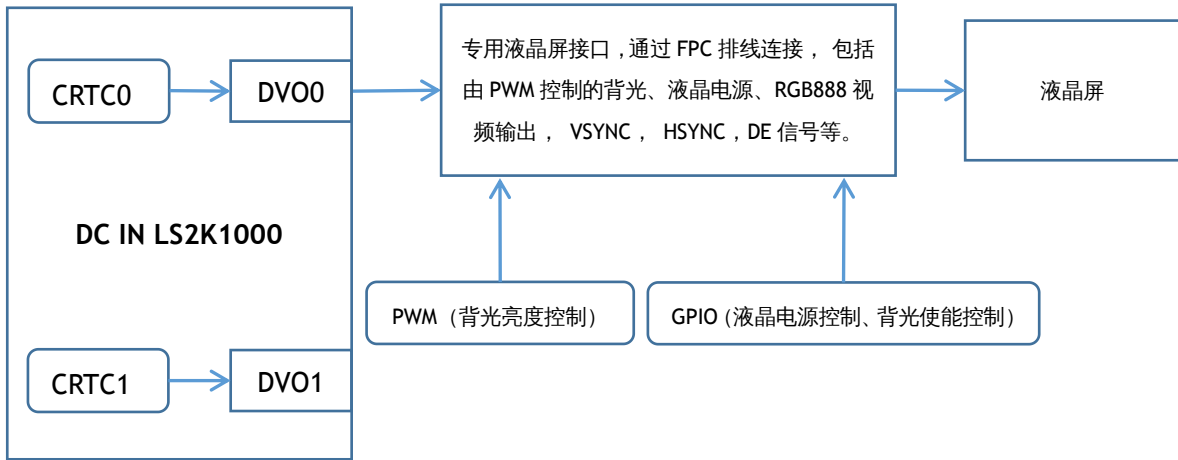


图 2-3: 液晶屏幕显示配置方案

由于 DPI 接口的液晶屏幕不具备 I2C 接口，无法通过 I2C 读取其显示屏幕的参数。因此需要通过设备树传递 EDID 或显示时序。此外液晶屏幕需要通过 PWM 控制器和 GPIO 来控制液晶屏幕的背光亮度和屏幕电源开关的。这些控制管脚需要连接到 LS2K SoC 的 PWM 控制器和 GPIO 上。具体使用哪个 PWM 或那个 GPIO 本规范不做限制，要求所用硬件资源不要和其它子系统功能模块重叠，避免硬件资源冲突，以免限制整个系统应用的灵活性。

扩展双屏应用时部分 LS2K1000LA 的显示控制器不推荐使用水平分辨率不能被 64 整除的屏幕（比如 800x480），如果一定要用这种显示屏幕，应将这种分辨率的屏幕连接到第二路显示管线上。单屏应用不存在这种问题。

2.4 LS2K0500 显示方案

LS2K0500 Soc 中集成两路显示，一路 DVO 显示，一路 VGA 显示。如果只需要一个 VGA 输出，那么可以直接使用 LS2K0500 的第二路显示管线；如果需要接其他接口(HDMI、DVI、LVDS、DPI)的显示设备，需要在第一路 DVO 显示输出上外接视频转换芯片，由于 LS2K0500 系统带宽有限，建议只使用一路显示输出。硬件连接方案如图 2-5、图 2-6、图 2-7 所示。

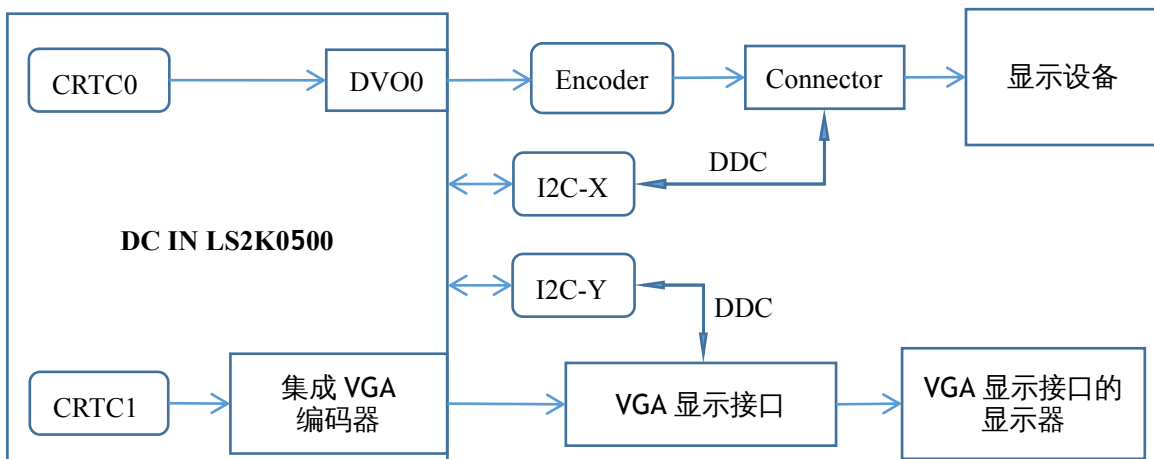


图 2-5: LS2K0500 第一路显示管线采用透明式编码器的显示连接方案示意图

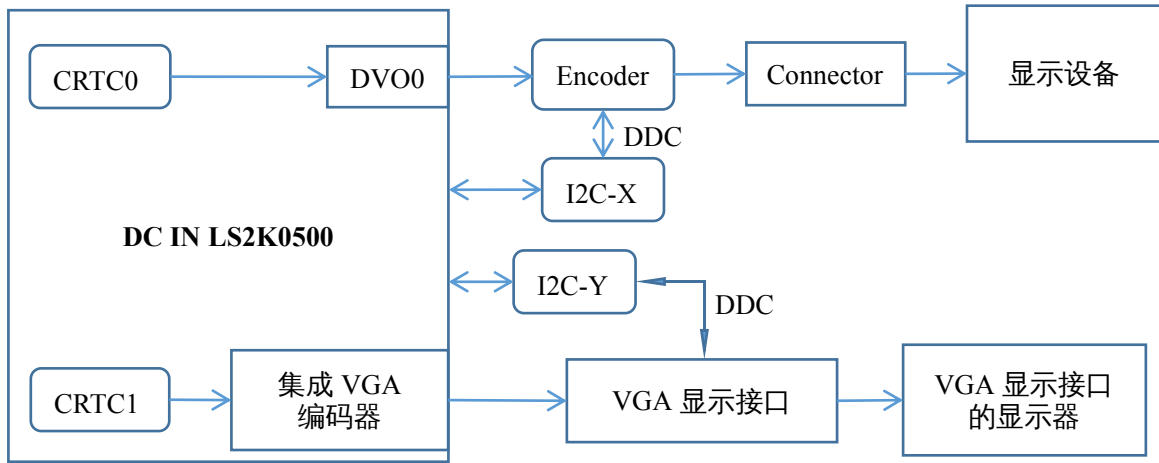


图 2-6 : LS2K0500 第一路显示管线采用非透明式编码器的显示连接方案示意图

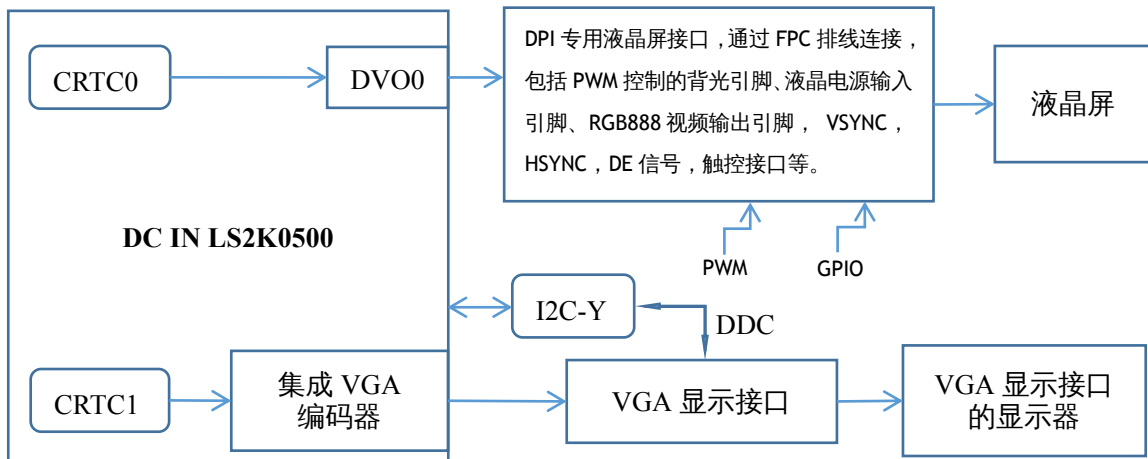


图 2-7 : LS2K0500 第一路显示管线采用液晶屏的显示连接方案示意图

2.5 多屏显示配置案例

推荐使用本章节中 2.1、2.2、2.3、2.4、2.5 小节所介绍的显示配置方案的组合，下面介绍几款有龙芯中科设计的板卡作为参考。

龙芯派 (LS2K1000_PA1_UDB_V1.5) 的显示配置方案如图 2-8 所示，主板的第一路显示配适的是飞凌嵌入式出品的 1024x600(或 800x480)的液晶屏幕。采用 LS2K1000LA SoC 中的 PWM0 控制背光亮度控制，GPIO3 控制背光电源使能和背光调节使能。主板的第二路显示管线选用 SII9022 DVO 转 HDMI 编码器，使用 LS2K1000LA SoC 中的硬件 I2C-1 来配置 SII9022 和读取 EDID，建议新设计的主板能够采用本章 2.2 章节所描述，采用非透明输出的方式使用 SII9022 编码器芯片。

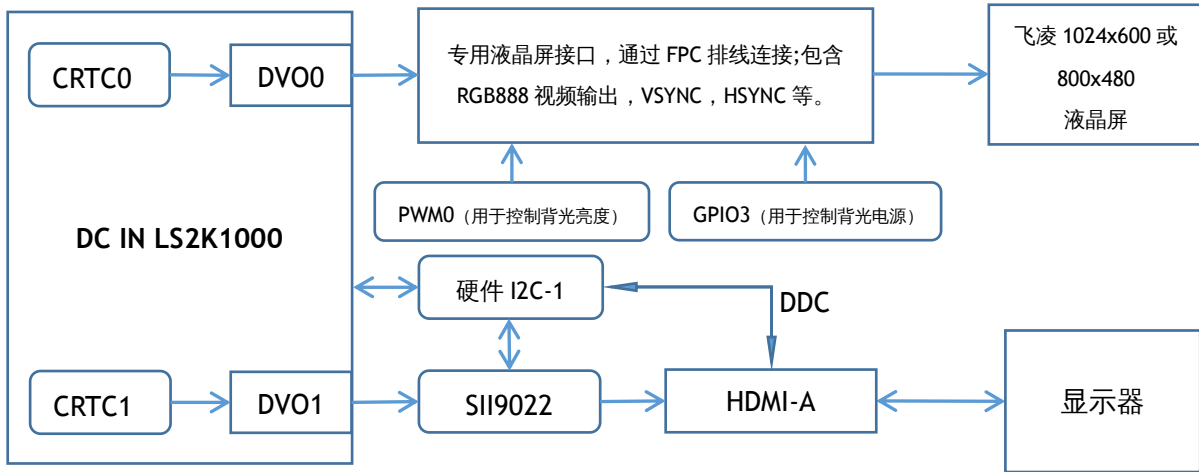


图 2-8: 龙芯派显示配置方案

龙芯教育派的显示配置方案如图 2-9 所示，龙芯教育派只使用第一路显示输出。它选取了 SII9022 作为其第一路显示管线的编码器芯片，对编码器的配置方法与上文中介绍的龙芯派的第二路显示管线相同。新设计的产品推荐采用非透明输出的方式使用 SII9022 编码器芯片。

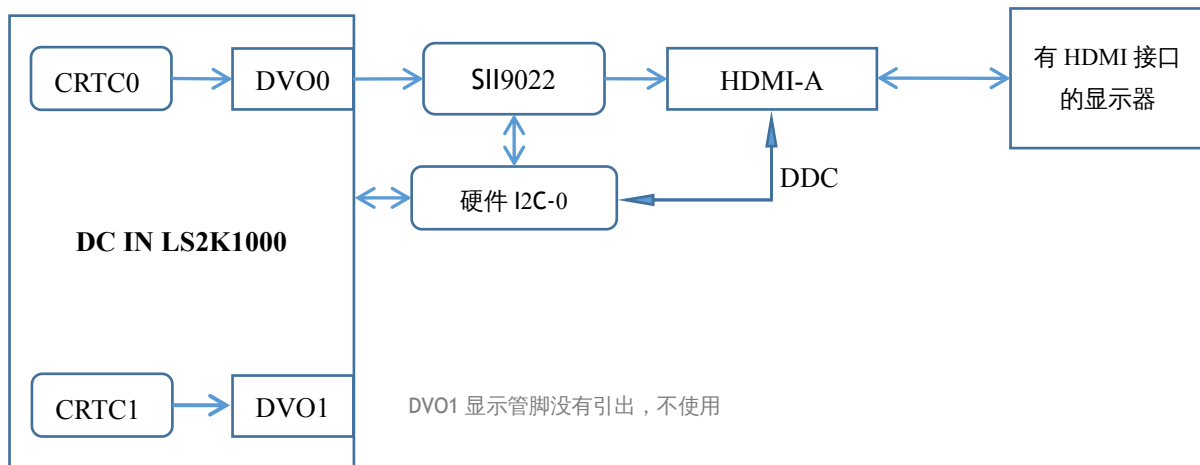


图 2-9: LS2K1000LA 教育派显示输出配置方案

LS2K1000LA 开发板 (LS2K1000_PC_EVB_V1.2) 非透明的显示编码器除具备 DDC 功能外，还具备热插拔中断转发的功能。需要把显示连接器的热插拔中断功能管脚连接到显示编码器的热插拔中断输入引脚上，再将显示编码器的热插拔中断输出引脚连接到 LS2K1000LA SoC 的 GPIO 上。

显示配置如图 2-10 所示，该板卡两路输出均采用透明类型的编码器，是本章中 2.1 小节所介绍的显示连接方案的一个实例。值得注意的是由于 LS2K1000LA 中只有两个硬件 I2C，I2C 资源不够用，所以这款主板用的是 GPIO 模拟 I2C 来充当 DDC 总线的功能。

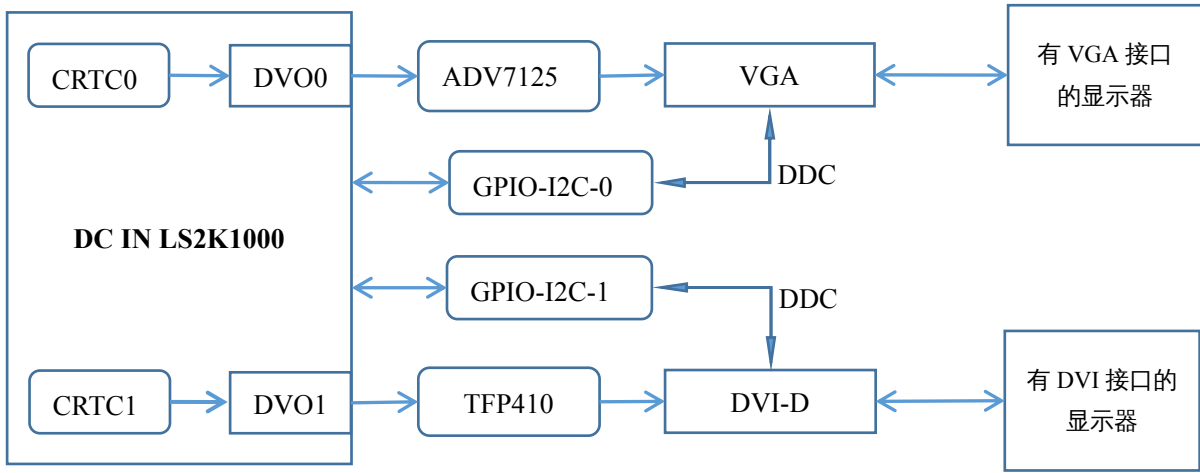


图 2-10 : LS2K1000LA 测试 (开发) 机器显示输出配置方案

3. 设备树传参规范 A

设备树传参方式 A 是一种简单易用集显设备树传参规范。适用于主板设计时采用透明视频转换芯片的场景，如果在固件启动阶段对非透明类型的显示编码器进行配置，使之不需要 LSDC 设备驱动的干预就能正常显示，这种情况下，设备树传参方式 A 也是适用的。设备树传参方式 A 支持从设备树传 EDID、支持从设备树传递多组显示时序、支持从设备树指定显示控制器工作在双屏硬件克隆工作模式等功能。除此之外，设备树传参规范 A 还承载着后向兼容性的历史责任。

本规范中为了节约篇幅本，章节中以片段的形式给出设备树示例。请读者结合龙芯 4.19 产品内核中 arch/loongarch/boot/dts/loongson/目录中查看完整的设备树文件。

3.1 设备树传参规范 A

龙芯 2K 系列 SoC 集显设备树的传参数基本实现方法如图 3-1 所示，图 3-1 中左边是显示控制器设备树节点，图 3-1 中右边是 I2C 设备树节点。固件工程师需要根据主板的设计来配置 I2C 应用于那一路显示管线。这里给出的样例表示的是第一路显示用的是 i2c0，第二路显示管线用的 i2c1。

<pre> dc@0x400c0000 { compatible = "loongson , display-subsystem"; reg = <0 0x400c0000 0 0x00010000>; interrupt-parent = <&icu>; interrupts = <28>; output-ports = <&dvo0 &dvo1>; #address-cells = <1>; #size-cells = <0>; dvo0: dvo@0 { reg = <0>; ddc-i2c-bus = <&i2c0>; connector = "vga-connector"; status = "okay"; }; </pre>	<pre> i2c0: i2c-gpio@0 { compatible = "i2c-gpio"; gpios = <&pioA 1 0 &pioA 0 0>; i2c-gpio , delay-us = <5>; /* ~100 kHz */ }; i2c1: i2c-gpio@1 { compatible = "i2c-gpio"; gpios = <&pioA 32 0 &pioA 33 0 >; i2c-gpio , delay-us = <5>; /* ~100 kHz */ }; i2c2: i2c@1fe21000 { compatible = "loongson , ls-i2c"; reg = <0 0x1fe21000 0 0x8>; interrupt-parent = <&icu>; interrupts = <22>; }; i2c3: i2c@1fe21800 { compatible = "loongson , ls-i2c"; reg = <0 0x1fe21800 0 0x8>; interrupt-parent = <&icu>; interrupts = <23>; }; </pre>
---	--

图 3-1: LS2K1000LA 集显设备树示例

LS2K1000LA 中的显示控制器有两路 dvo 显示输出，在设备树中用 output-ports = <&dvo0 &dvo1>列举出来。其中&dvo0 表示对 dvo@0 设备节点的引用，&dvo1 表示对 dvo@1 设备节点的引用。如果有 3 路显示输出则可以写为 output-ports = <&dvo0 &dvo1 &dvo2>，但目前龙芯所有的显示控制器都只有两路显示输出。

LSDC 驱动程序通过 ouput-ports 属性值找到 dvo@0 和 dvo@1 设备节点的，dvo@0 和 dvo@1 中能包涵子设备节点。dvo@0 设备节点表示第一路显示管线的相关属性，表征 LS2K1000LA 的 dvo0 输出接口及其下游配套的显示编码器、显示连接器等的一切属性的集合，用标号 dvo0 表示。dvo@1 设备节点表示第二路显示管线的相关属性，表征 LS2K1000LA 的 dvo1 输出接口及其下游的一切属性的集合，用标号 dvo1 表示。

dvo0 和 dvo1 中 status 属性的值为 "ok"、"okay" 或不设置都表示这路显示为可用的状态。如果设备树中实现 status 属性并且值不是 "ok" 或 "okay"，那么这条显示管线就会被驱动程序解析为不可用。如果两路

显示中只使用其中的一路显示管线，可以把需要关闭的显示管线的 status 属性值改为"disabled"。LSDC 设备驱动不会再占用这路显示管线原来所占用的显示输出接口以及 I2C 设备等硬件资源。

ddc-i2c-bus 属性用来描述路显示管线用的 i2c 设备，格式为 **ddc-i2c-bus = <&i2c0>**表示该路显示管线引用的是 i2c0，如果引用 i2c1 则写成 **ddc-i2c-bus = <&i2c1>**。设备树中没有 ddc-i2c-bus 属性时，默认 dvo0 显示管线 i2c0，dvo1 显示管线用 i2c1。推荐用户在设备树中通过 ddc-i2c-bus 设备节点提供该路显示管线所用的 I2C 设备。如果主板采用不支持 DDC 的液晶屏，那 ddc-i2c-bus 属性就不需要写，需要由设备树提供 EDID 或显示时序。如果通过设备树提供 EDID 或显示时序数据，那么这路显示管线就总被视为连接状态，LSDC 设备驱动按照给定的显示时序（或显示模式）强制输出。驱动通过 **ddc-i2c-bus** 属性所指定的 I2C 总线来探测显示器的插拔状态和读取 EDID 信息。

connector 属性用于指定连接器类型，需要根据主板的设计来填写。可选的值为："vga-connector"，"dvi-connector"，"hdmi-connector"，"dpi-connector"，"lvds-connector"。例如龙芯派配套电容屏的 connector 属性值为 "dpi-connector"。LS2K1000LA 开发板主机 (LS2K1000_PC_EVB_V1.2) 主板的 vga 接口的 connector 属性值为 "vga-connector"。dvi 显示接口的 connector 属性值为 "dvi-connector"。

reg 属性用于标识显示管线的硬件序号，从 0 开始，属于冗余设计。dvo@0 设备节点中就写 0，表示第一路显示输出；dvo@1 设备节点中就写 1，表示第二路显示输出。

compatible 属性用于指定的显示控制器所兼容的设备驱动，LSDC 设备驱动也通过 compatible 值来识别龙芯 2K 系列 SoC 的型号。支持"loongson,ls2k0500-dc"，"loongson,ls2k1000-dc"和"loongson,display-subsystem" 共计三个 compatible 属性值。ls2k0500 使用"loongson,ls2k0500-dc"，ls2k1000LA 使用"loongson,2k1000-dc"，如果 compatible 使用"loongson,display-subsystem"，设备驱动系统默认识别为 ls2k1000LA 中的显示控制器。

3.2 传递 EDID 的配置方法

图 3-2 给出通过设备树传递 EDID 数据的例子，需要注意的是 EDID 需要由液晶屏幕厂商提供，图 3-2 中左边的文本框给出了简要说明。

<p>假如主板的第二路显示管线 (dvo1) 连接了一个不支持 ddc 的液晶屏幕，设备树可以写成如右边文本框中的那样。</p> <p>用户需要自己根据液晶屏幕的分辨率生成相应的 EDID 数据，放在 edid=[]；目前支持的 edid 长度为 128 字节，每个字节与每个字节用之间用空格或 Tab 键或回车键分隔开就行。</p> <p>如果是第一路显示管线 dvo0 连接了一个不支持 ddc 的液晶屏幕，那就在 dvo@0 节点中填写 edid 属性。如果两路都接了不支持 ddc 的液晶屏，那两路都需要提供 edid。</p> <p>edid blob 和显示时序可以同时提供。</p> <p>由于历史原因，edid 比显示时序具有更高的优先级，设备树中提供了 edid，就优先用 edid。没有 edid 再检测设备树中有没有提供显示时序，如果时序也没有提供我们就去用 i2c 去读取显示器的 edid。</p>	<pre> dc@0x400c0000 { compatible = "loongson, display-subsystem"; reg = <0 0x400c0000 0 0x00010000>; interrupt-parent = <&icu>; interrupts = <28>; output-ports = <&dvo0 &dvo1>; #address-cells = <1>; #size-cells = <0>; dvo0: dvo@0 { reg = <0>; /* 0 for display pipe 0 */ ddc-i2c-bus = <&i2c0>; connector = "vga-connector"; status = "ok"; }; dvo1: dvo@1 { reg = <1>; /* 1 for display pipe 1 */ connector = "hdmi-connector"; /* 1920x1080@60Hz */ edid = [00 ff ff ff ff ff ff 00 1e 6d 54 5b 0b cc 04 00 02 1c 01 03 6c 30 1b 78 ea 31 35 a5 55 4e a1 26 0c 50 54 a5 4b 00 71 4f 81 80 95 00 b3 00 a9 c0 81 00 81 c0 90 40 02 3a 80 18 71 38 2d 40 58 2c 45 00 e0 0e 11 00 00 1e 00 00 00 fd 00 38 4b 1e 53 0f 00 0a 20 20 20 20 20 00 00 00 fc 00 4c 47 20 46 55 4c 4c 20 48 44 0a 20 20 00 00 00 ff 00 38 30 32 4e 54 43 5a 39 38 33 37 39 0a 00 35]; status = "ok"; }; }; </pre>
--	--

图 3-2：设备树传递 EDID 数据样例

3.3 传递显示时序的配置方法

嵌入式设备使用液晶屏幕通常没有 DDC 支持。从设备树传递显示屏的显示时序信息是常用的做法。图

3-3 所示的文本框给出通过设备树传递显示时序的例子。绝大多数液晶屏幕只支持一种显示分辨率，如果屏幕支持多个分辨率，设备树传参规范 A 也支持给定多组显示时序，详见图 3-4。

<p>右边 display-timing@0 {} 属性中包含的信息是某品牌的一款液晶屏幕的时序 (1024x600@60Hz)。用户需要根据自己的液晶屏幕参数填写正确的显示时序。</p> <p>display-timings 设备节点中的 native-mode 属性用于指定液晶屏所支持的最佳分辨率。开机进入桌面默认就会使用这个分辨率。</p> <p>如果是第二路显示管线连接了这种液晶屏幕 那就 dvo@1 节点中填写 display-timings 设备节点及其包含的 native-mode 属性和显示时序参数子节点。 如果两路显示都接了液晶屏， 那就两路显示输出属性设备节点中都需要提供相应的显示时序。</p> <p>注意： 提供了显示时序的显示管线会被驱动程序认为总是处于连接的状态。操作系统会往这块屏幕上更新画面，即使用户不接这块屏幕。</p>	<pre>dc@0x400c0000 { compatible = "loongson , display-subsystem"; output-ports = <&dvo0 &dvo1>; dvo0: dvo@0 { reg = <0>; status = "ok"; connector = "dpi-connector"; display-timings { native-mode = <&mode_1024x600_60>; mode_1024x600_60: display-timing@0 { clock-frequency = <51200000>; hactive = <1024>; hsync-len = <4>; hfront-porch = <160>; hback-porch = <156>; vactive = <600>; vfront-porch = <11>; vback-porch = <23>; vsync-len = <1>; }; }; }; dvo1: dvo@1 { reg = <1>; status = "ok"; connector = "hdmi-connector"; type = "hdmi-a"; ddc-i2c-bus = <&i2c1>; }; };</pre>
---	---

图 3-3： 设备树传递显示时序的示例

属于同一组的显示时序放于 display-timing@0 设备节点中，而 display-timing@0 又被包含在 display-timings {} 设备节点中。多数显示液晶显示屏只支持一种显示分辨率，用户只需提供一组时序参数。mode_1024x600_60 为 display-timing@0 设备节点的标号，它代表 display-timing@0 设备节点，起到被其它属性引用目的。display-timing@0 中的所有属性都是必须的，用户只需改变 <> 中的数字。具体的每个参数典型值需要从液晶屏厂商提供的数据手册中获取。

如果显示屏幕支持多个分辨率，在 **display-timings {}** 设备节点中依次罗列 **display-timing@1**，**display-timing@2**，...，**display-timing@n** 设备节点即可。每组显示时序信息的设备节点名字互不相同。其中每个 **display-timing@n** (n = 1,2,...,n) 设备节点中都包含一组显示时序参数。

如果主板可以配适多种不同的屏幕，那么一个设备树文件的 **display-timings {}** 设备节点中也可以包含多组不同的显示时序参数。更换显示屏时，只须改动 **native-mode** 属性所指向的显示时序即可。起到工程意义上的便捷、复用的目的。

如果用户实际的板卡是第二路显示输出连接的液晶屏幕，请把 **display-timings {}** 设备节点及其包含的子设备节点写到 **dvo@1** 设备节点中。这里为了节约篇幅，略去了对 **dvo@1** 设备节点的书写。

```
dc@0x400c0000 {
    compatible = "loongson, display-subsystem";
    output-ports = <&dvo0 &dvo1>;
    dvo0: dvo@0 {
        reg = <0>;
        status = "ok";
        display-timings {
            native-mode = <&mode_800x480_60>;

            mode_1024x600_60: display-timing@0 {
                clock-frequency = <51200000>;
                hactive = <1024>;
                hsync-len = <4>;
                hfront-porch = <160>;
                hback-porch = <156>;
                vactive = <600>;
                vfront-porch = <11>;
                vback-porch = <23>;
                vsync-len = <1>;
            };

            mode_800x480_60: display-timing@1 {
                clock-frequency = <30066000>;
                hactive = <800>;
                hfront-porch = <50>;
                hback-porch = <70>;
                hsync-len = <50>;
                vactive = <480>;
                vback-porch = <0>;
                vfront-porch = <0>;
                vsync-len = <50>;
            };
        };
    };
};
```

图 3-4: display-timings 设备节点中包含多组显示时序参数的设备树样例

3.3.1 显示时序中各个参数的意义

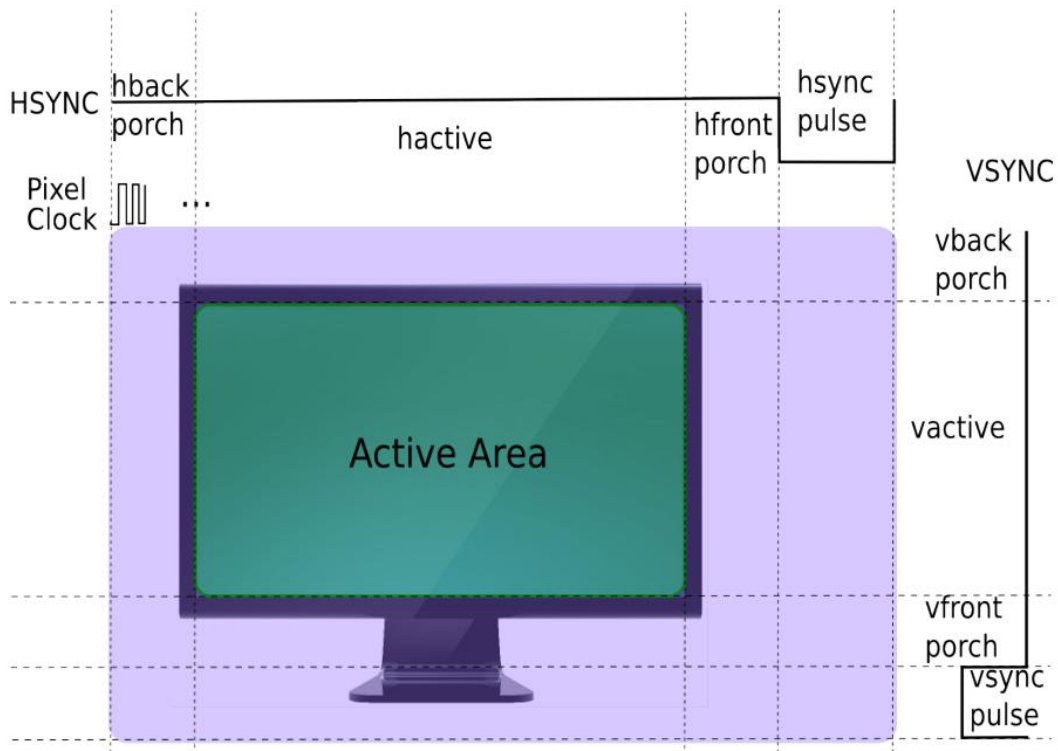


图 3-6: 液晶屏幕显示时序示意图

参数	意义
clock-frequency	单位为 Hz ，视频硬件在显示器上绘制像素的速率，一个时钟周期完成一个像素点的显示。这个参数表征的是视频硬件在一秒钟所绘制的显示个数。
hsync-len	水平同步信号的宽度，以像素个数为单位，用来 告诉显示硬件换行的 。hsync-len 就是图 3-6 中的 hsync pulse。
hactive	水平像素数目，单位为个。表示显示屏 一行 中可显示像素个数，就是显示屏的水平分辨率。
hfront-porch	水平同步信号的前肩，表示从一行的有效数据结束到下一个水平信号开始之间的像素时钟的个数。
hback-porch	水平同步信号的后肩，表示从水平同步信号开始到一行的有效数据开始之前的像素时钟的个数。
vactive	垂直像素数目，单位为个。表示显示屏 一列 中可显示像素个数，就是显示屏的垂直分辨率。
vsync-len	垂直同步信号的宽度，以行为单位，用来 告诉显示硬件这一帧画面显示完毕的，开始显示下一帧画面 。vsync-len 就是图 3-6 中的 vsync pulse。
vfront-porch	垂直同步信号的前肩，表示一帧图像结束后，垂直同步信号 以前 的无效行数
vback-porch	垂直同步信号的后肩，表示一帧图像开始时，垂直同步信号 以后 的无效行数

3.4 关闭某一路显示管线的方法

龙芯 2K 系列 SoC 在某些应用场合可能只需要单路显示输出，这时需要关闭将不使用的那路显示输出，在设备树中将对应的显示管线的 status 属性的值改为 disabled 即可。如图 3-7 设备树样例所示。

<p>如果主板只引出了第一路 dvo 显示，第二路显示接口没有被引出，则可以将第二路显示管线中的 status 属性改为 disabled。</p> <p>当某一条显示管线中的 status 属性被设置为“disabled”值，那么这条显示管线中的其他属性即使写了也不会被使用，驱动程序不会再为这条显示管线分配资源。这里为了清晰和节约篇幅删去了其余属性。</p>	<pre>dc@0x400c0000 { compatible = "loongson, display-subsystem"; reg = <0 0x400c0000 0 0x00010000>; interrupt-parent = <&icu>; interrupts = <28>; output-ports = <&dvo0 &dvo1>; #address-cells = <1>; #size-cells = <0>; dvo0: dvo@0 { reg = <0>; /* 0 for display pipe 0 */ ddc-i2c-bus = <&i2c1>; connector = "hdmi-connector"; status = "ok"; }; dvo1: dvo@1 { reg = <1>; /* 1 for display pipe 1 */ status = "disabled"; }; };</pre>
--	--

图 3-7: 禁止使用某一路显示管线的设备树书写样例

3.5 硬件克隆显示设备树配置方法

龙芯 LS2K 系列 SoC 中的显示控制器支持独特的硬件克隆工作模式，当开启时这种显示模式时，某一路显示管线会复制另一路显示管线的显示内容。从 LSDC 设备驱动角度来看，此时系统中只有一路可控的显示管线。使用硬件克隆方案可以降低显示控制对内存带宽的占用。图 3-8 给出了一个通过设备树来配置显示控制器工作在硬件克隆模式的例子。

<p>如果想实现第二路显示管线 (dvo1) 克隆第一路显示管线 (dvo0)，则可以在的 dvo1 设备节点中添加 clone 属性，如右面文本框。</p> <p>如果想第一路显示管线 (dvo0) 克隆第二路显示管线 (dvo1)，则可以在的 dvo@0 设备节点中添加 clone 属性。</p> <p>请勿在两条显示输出管线的设备结点中同时添加 clone 属性。</p>	<pre>dc@0x400c0000 { compatible = "loongson, display-subsystem"; reg = <0 0x400c0000 0 0x00010000>; interrupt-parent = <&icu>; interrupts = <28>; output-ports = <&dvo0 &dvo1>; #address-cells = <1>; #size-cells = <0>; dvo0: dvo@0 { reg = <0>; /* 0 for display pipe 0 */ ddc-i2c-bus = <&i2c0>; connector = "hdmi-connector"; status = "ok"; }; dvo1: dvo@1 { reg = <1>; /* 1 for display pipe 1 */ clone; }; }</pre>
---	--

图 3-8: 硬件克隆工作模式的设备树配置样例

3.6 背光配置方法

背光驱动是显示驱动的一部分，图 3-9 给出背光设备的设备树样例，左边背光设备节点的名字为 backlight，右边是背光所依赖的 pwm 控制器的设备树。其中 lcd_backlight 是背光设备节点的标号，用于被其他设备节点所引用。pwm0 是 pwm 控制器的设备树的标号，背光设备节点引用了 pwm0 设备节点。

<pre> lcd_backlight: backlight { compatible = "pwm-backlight"; pwms = <&pwm0 0 4000000>; brightness-levels = < 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 >; default-brightness-level = <80>; }; </pre>	<pre> pwm0: pwm@1fe22000{ compatible = "loongson,ls2k-pwm"; reg = <0 0x1fe22000 0 0x10>; clock-frequency = <125000000>; interrupt-parent = <&icu>; interrupts = <24>; #pwm-cells = <2>; }; pwm1: pwm@1fe22010{ compatible = "loongson,ls2k-pwm"; reg = <0 0x1fe22010 0 0x10>; clock-frequency = <125000000>; interrupt-parent = <&icu>; interrupts = <25>; #pwm-cells = <2>; }; </pre>
---	---

图 3-9：背光设备驱动节点及其属性（左）和 PWM（右）设备树示例

compatible 属性，属性的值为 **"pwm-backlight"**。表示基于 pwm 的背光设备驱动，其代码位于 linux 内核中的 drivers/video/backlight/pwm_bl.c 中。

pwms 属性，属性的值为 **<&pwm0 0 4000000>**，表示使用 **pwm0** 控制器来驱动液晶屏背光亮度，选用哪一个控制器是主板设计时决定的。如果用 **pwm1** 来做背光控制，那么需要修改成 **pwms = <&pwm1 0 4000000>**。

pwm 控制器的设备树节点中必须用 **#pwm-cells** 指定 pwm 控制器驱动接受参数的数量，这里 **#pwm-cells = <2>** 表示 pwm 控制器驱动接受两个参数。背光设备驱动节点中的 **pwms = <&pwm0 0 4000000>** 中的 **0** 和 **4000000** 就是这两个参数。第二个参数 **0** 表示这个 PWM 驱动所驱动的 PWM 设备的序号，当前产品内核中每个 PWM 驱动实例都只驱动一个硬件 PWM 设备，所以固定填写 0。第三个参数 **4000000** 则是用来设置背光驱动所用的 PWM 设备的工作周期，**单位是纳秒**。

brightness-levels 属性用来存放用来指示背光亮度级别，这里给定 100 个背光级别，0 表示亮度最暗状态；99 表示亮度最亮。

当 PWM 设备驱动的工作极性与正常期望相反时，给定 PWM 控制驱动的占空比越大，PWM 控制器输出的电平平均越低，显示屏亮度越暗。在这种情况下可以将 brightness-levels 中的数值反过来写，如图 3-10 中所示。

<pre> lcd_backlight: backlight { compatible = "pwm-backlight"; pwms = <&pwm0 0 4000000>; brightness-levels = < 99 98 97 96 95 94 93 92 91 90 89 88 87 86 85 84 83 82 81 80 79 78 77 76 75 74 73 72 71 70 69 68 67 66 65 64 63 62 61 60 59 58 57 56 55 54 53 52 51 50 49 48 47 46 45 44 43 42 41 40 39 38 37 36 35 34 33 32 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 >; default-brightness-level = <80>; }; </pre>	<pre> pwm0: pwm@1fe22000{ compatible = "loongson,ls2k-pwm"; reg = <0 0x1fe22000 0 0x10>; clock-frequency = <125000000>; interrupt-parent = <&icu>; interrupts = <24>; #pwm-cells = <2>; }; pwm1: pwm@1fe22010{ compatible = "loongson,ls2k-pwm"; reg = <0 0x1fe22010 0 0x10>; clock-frequency = <125000000>; interrupt-parent = <&icu>; interrupts = <25>; #pwm-cells = <2>; }; </pre>
---	---

图 3-10: 背光 (左) 和 PWM (右) 的设备树书写示例

default-brightness-level 属性用于设定背光设备的默认工作档位，取值范围根据需求，在 brightness-levels 属性的最小值和最大值之间取值即可。

亮度级别设置越多，背光亮度调节越细腻，如果希望少一些，下面给出了一种精简背光调级的设备树写法。

```

brightness-levels = < 90 80 70 60 50 40 30 20 10 0 >;
default-brightness-level = <8>;

```

这时背光驱动只有 10 个档位 (0~9)，档位 0 最亮，档位 9 最暗。

4. 设备树传参规范 B

本章节所介绍的设备树规范又被称为 OF GRAPH，读者可以结合 Linux 内核源码中的文档 Documentation/devicetree/bindings/graph.txt Documentation/devicetree/bindings/video-interface.txt 本章节主要以示例的方式介绍 DRM Bridge 类型显示编码器驱动及其设备树书写规则。

4.1 设备树传参规范 B

在显示控制器设备节点中实现 ports 节点。ports 节点是显示控制器节点 dc@0x400c0000 的子节点，用于囊括龙芯 2K 系列 SoC 的所有输出端口。龙芯 2K 系列 SoC 有两条显示管线，有两个输出端口，因此 ports 设备节点中需要实现两个子设备节点 port@0 和 port@1，每个 port 设备节点中只包含一个 endpoint 设备节点，简称端点，用来表示输出端口和外围设备之间的连接关系。port@0 代表第一路显示管线及其输出端口，简称为端口 0；port@1 代表的是第二路显示管线的及其输出端口，简称为端口 1。显示控制器设备节点的整体情况如图 4-1 所示。

```
dc@0x400c0000 {
    compatible = "loongson,display-subsystem";
    reg = <0x400c0000 0x00010000>;
    interrupt-parent = <&icu>;
    interrupts = <36>;
    #address-cells = <1>;
    #size-cells = <0>;

    ports {
        #address-cells = <1>;
        #size-cells = <0>;

        port@0 {
            reg = <0>;
            dvo0 : endpoint {
                remote-endpoint = <&vga_encoder_in>;
            };
        };

        port@1 {
            reg = <1>;
            dvo1 : endpoint {
                remote-endpoint = <&hdmi_encoder_in>;
            };
        };
    };
};
```

图 4-1: 设备树传参 B 规范下显示控制器设备节点的书写方法

endpoint 设备节点用于被其它设备节点所引用。每个端点必须包含一个 remote-endpoint 属性，用于表示这个端口连接到哪个远程设备。同样地，这个远程设备的 endpoint 属性中也必须包含一个 remote-endpoint 属性，这个 remote-endpoint 属性对显示控制器中 ports 设备节点中的 endpoint 节点进行引用，描述两个物理

设备之间的相互连接关系。

port、endpoint 和 remote-endpoint 属性只用来表示连接关系，不能用来表示连接的类型和数据流方向。图 4-1 中所示的显示控制器设备节，port@0 和 part@1 都是输出端口。图 4-1 中的 vga_encoder_in 和 hdmi_encoder_in 则是显示编码器设备节点的 endpoint 设备节点的标号，读者可以对照本章节中的图 4-2 和图 4-4 综合来看。用户需要根据板卡的硬件连接情况用设备树描述出来设备之间的连接信息。

设备树传参规范 B 中除 ports 设备节点之外的属性和设备树传参规范 A 规范要求相同，比如 **compatible** 属性用来指定龙芯 2K 系列 SoC 中的显示控制器所兼容的设备驱动，LSDC 设备驱动也通过 compatible 值来识别龙芯 2K 系列 SoC 的型号。支持 "loongson,2k0500-dc"，"loongson,2k1000-dc"，"loongson,display-subsystem" 三个 compatible 属性值。ls2k0500 使用"loongson,2k0500-dc"，ls2k1000LA 使用"loongson,2k1000-dc"，如果 compatible 值写成为"loongson,display-subsystem"，设备驱动系统将会把显示控制器识别为 ls2k1000LA 中的显示控制器。

本规范中为了节约篇幅本，章节中以片段的形式给出设备树示例。请读者结合龙芯 4.19 产品内核中 arch/loongarch/boot/dts/loongson/目录中查看完整的设备树文件。

4.2 LT8618 编码器设备树实现示例

龙讯 LT8618 转换芯片设备树实例如图 4-2 所示。

```
i2c1: i2c@1fe01800 {
    compatible = "loongson,ls-i2c";
    reg = <0x1fe01800 0x8>;
    #address-cells = <1>;
    #size-cells = <0>;
    lt8618 {
        compatible = "lontium,lt8618";
        reg = <0x39>;
        /* hpd-gpios = <&pioA 0 GPIO_ACTIVE_HIGH>; */
        /* reset-gpios = <&pioA 39 GPIO_ACTIVE_LOW>; */
        ports {
            #address-cells = <1>;
            #size-cells = <0>;
            port@0 {
                reg = <0>;
                hdmi_encoder_in: endpoint {
                    remote-endpoint = <&dvo1>;
                };
            };
            port@1 {
                reg = <1>;
                hdmi_encoder_out: endpoint {
                    remote-endpoint = <&hdmi_connector_in>;
                };
            };
        };
    };
};

hdmi-connector {
    type = "hdmi-a";
    status = "ok";
    port@0 {
        hdmi_connector_in: endpoint {
            remote-endpoint = <&hdmi_encoder_out>;
        };
    };
};
```

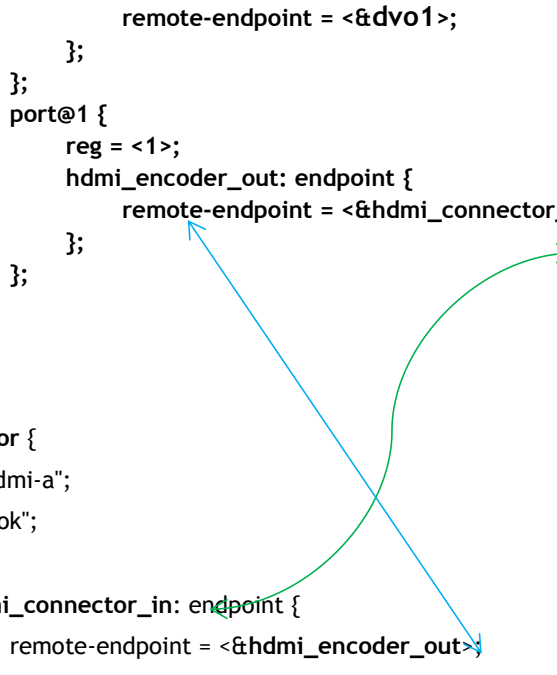


图 4-2：LT8618 DVO 转 HDMI 设备树节点书写示例

4.3 SII9022 编码器设备树实现示例

图 4-3 给出了 SII9022 编码器当作非透明编码器使用的设备树写法的示例。

```

i2c1: i2c@1fe01800 {
    compatible = "loongson,ls-i2c";
    reg = <0x1fe01800 0x8>;
    interrupt-parent = <&icu>;
    interrupts = <31>;

    #address-cells = <1>;
    #size-cells = <0>;

    sil9022acnu {
        compatible = "sil,sii9022";
        reg = <0x39>;

        hpd-gpios = <&pioA 0 GPIO_ACTIVE_HIGH>;
        interrupt-parent = <&pioA>;

        ports {
            #address-cells = <1>;
            #size-cells = <0>;
            port@0 {
                reg = <0>;
                hdmi_encoder_in: endpoint {
                    remote-endpoint = <&dvo1>;
                };
            };
            port@1 {
                reg = <1>;
                hdmi_encoder_out: endpoint {
                    remote-endpoint = <&hdmi_connector_in>;
                };
            };
        };
    };
};

hdmi-connector {
    type = "hdmi-a";
    status = "ok";
    port@0 {
        hdmi_connector_in: endpoint {
            remote-endpoint = <&hdmi_encoder_out>;
        };
    };
};
    
```

图 4-3 : SII9022 非透明类型编码器的设备树书写示例

4.4 透明类型编码器的实现示例

```
vga-encoder {
    compatible = "adi,adv7123", "dumb-vga-dac";

    ports {
        #address-cells = <1>;
        #size-cells = <0>;

        port@0 {
            reg = <0>;
            vga_encoder_in: endpoint {
                remote-endpoint = <&dvo0>;
            };
        };

        port@1 {
            reg = <1>;
            vga_encoder_out: endpoint {
                remote-endpoint = <&vga_connector_in>;
            };
        };
    };
};

vga-connector {
    ddc-i2c-bus = <&i2c0>;
    port {
        vga_connector_in: endpoint {
            remote-endpoint = <&vga_encoder_out>;
        };
    };
};
```

图 4-4: 透明设备树书写示例

4.5 液晶屏显示设备实现示例

图 4-5 给出了一个 DPI 接口的液晶显示屏的设备树实现示例。

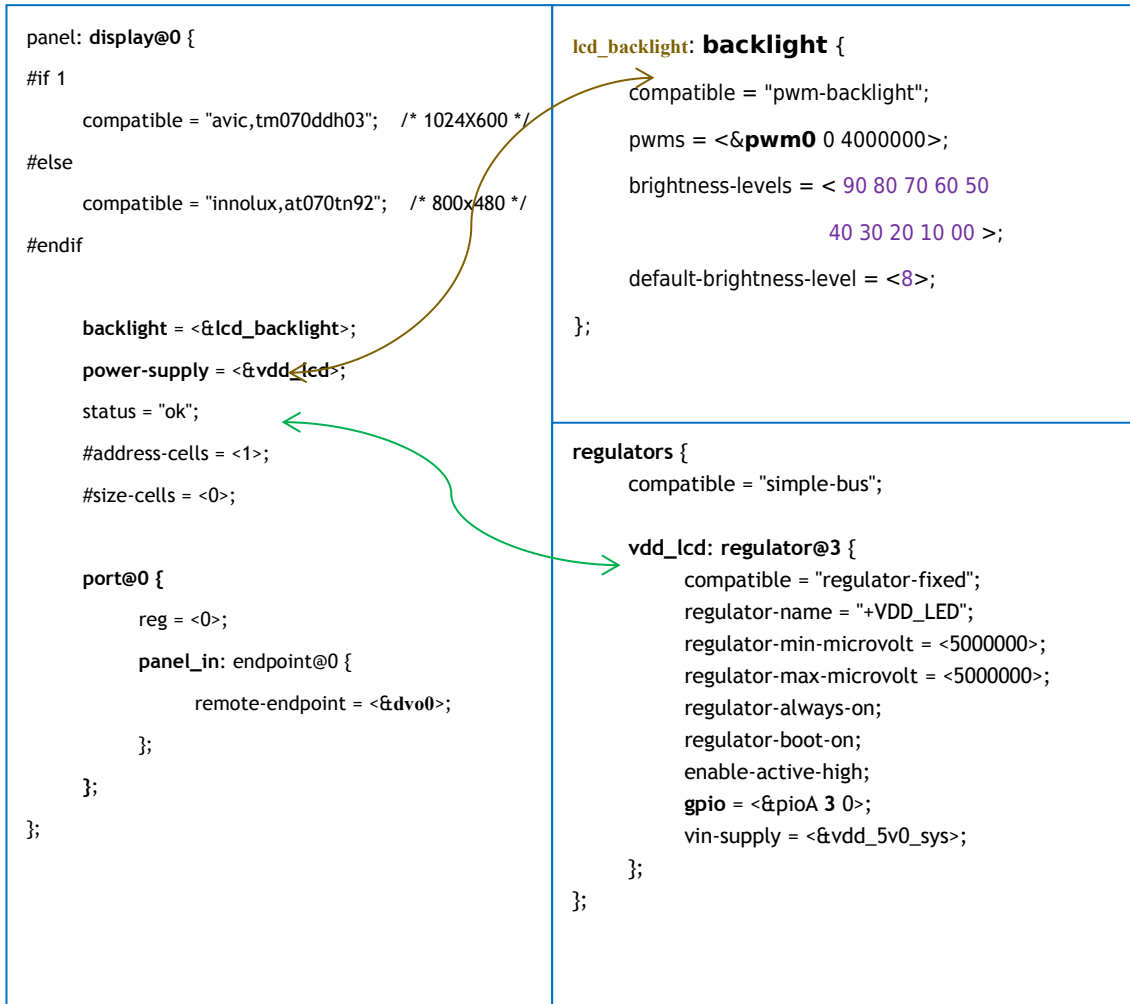


图 4-5: DPI 液晶显示屏示例

5. 参考文献

- [1] <https://www.kernel.org/doc/html/latest/devicetree/usage-model.html>
 [2] <https://www.devicetree.org/specifications/>