

**LOONGSON**

龙芯 2P0500 处理器

用户手册

V1.0

2024 年 01 月

龙芯中科技术股份有限公司

自主决定命运, 创新成就未来

北京市海淀区温泉镇中关村环保科技示范园龙芯产业园2号楼 100095  
Loongson Industrial Park, building 2, Zhongguancun environmental protection park  
Haidian District, Beijing



[www.loongson.cn](http://www.loongson.cn)

## 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

## 阅读指南

《龙芯 2P0500 处理器用户手册》主要介绍龙芯 2P0500 的架构与寄存器描述，包括用户手册和片上设备使用指南两部分，软件编程指南介绍对 BIOS 和操作系统开发过程中的常见问题。

关于龙芯 2P0500 处理器所集成的 LA364 高性能处理器核的相关资料，请参阅《龙芯 LA364 处理器核用户手册》。

## 版本信息

文档更新记录	文档名	龙芯 2P0500 处理器用户手册
	版本号	V1.0
	创建人	芯片研发部
更新历史		
序号.	版本号	更新内容
1	V1.0	发布版本

手册信息反馈: [service@loongson.cn](mailto:service@loongson.cn)

## 目 录

1	概述.....	1
1.1	体系结构框图.....	1
1.2	芯片主要功能.....	2
1.2.1	处理器核.....	2
1.2.2	内存接口.....	2
1.2.3	USB 控制器.....	2
1.2.4	GMAC 控制器.....	3
1.2.5	SPI.....	3
1.2.6	UART.....	3
1.2.7	I <sup>2</sup> C.....	3
1.2.8	打印接口.....	4
1.2.9	扫描接口.....	4
1.2.10	DA.....	4
1.2.11	PMIO.....	4
1.2.12	PWM.....	4
1.2.13	SDIO.....	4
1.2.14	HPET.....	4
1.2.15	RTC.....	4
1.2.16	GPIO.....	5
1.2.17	Watchdog.....	5
1.2.18	温度传感器.....	5
1.2.19	中断控制器.....	5
2	引脚定义.....	6
2.1	约定.....	6
2.2	DDR3 接口.....	6
2.3	USB 接口.....	7
2.4	GMAC 接口.....	7
2.5	SPI 接口.....	8
2.6	UART 接口.....	8
2.7	I <sup>2</sup> C 接口.....	8
2.8	PRINTER 接口.....	8
2.9	SCANNER 接口.....	9
2.10	DA 接口.....	9
2.11	SDIO 接口.....	9
2.12	PMIO 接口.....	9
2.13	PWM 接口.....	9
2.14	电源地接口.....	10
2.15	测试接口.....	10
2.16	JTAG 接口.....	10
2.17	系统相关信号.....	11
2.18	上电配置信号.....	11
2.19	外设功能引脚复用.....	11
3	时钟结构.....	15
3.1	SYS PLL.....	15
3.2	DDR PLL.....	16
3.3	VID PLL.....	17
3.4	内部 PLL 配置方法.....	17
3.4.1	硬件配置.....	17



3.4.2	软件配置.....	18
3.5	USB 参考时钟.....	18
3.6	时钟信号说明.....	18
4	电源管理.....	20
4.1	电源管理模块介绍.....	20
4.2	电源级别.....	20
5	芯片配置与控制.....	21
5.1	芯片初始化信号.....	21
5.2	地址空间分配.....	21
5.3	时钟与复位控制.....	24
5.3.1	时钟配置概要.....	24
5.3.2	展频 PLL 配置.....	25
5.3.3	复位控制.....	26
5.4	芯片配置寄存器.....	26
5.4.1	主系统通用配置寄存器 0.....	32
5.4.2	主系统通用配置寄存器 1.....	33
5.4.3	主系统通用配置寄存器 2.....	35
5.4.4	主系统通用配置寄存器 3.....	36
5.4.5	主系统通用配置寄存器 4.....	37
5.4.6	主系统通用配置寄存器 5.....	38
5.4.7	主系统采样参数寄存器.....	38
5.4.8	主系统高精度计数器 0.....	39
5.4.9	主系统高精度计数器 1.....	39
5.4.10	主系统 GPIO0~15 复用配置寄存器.....	39
5.4.11	主系统 GPIO16~31 复用配置寄存器.....	40
5.4.12	主系统 GPIO32~43 复用配置寄存器.....	41
5.4.13	USB PHY 配置寄存器 0.....	42
5.4.14	USB PHY 配置寄存器 1.....	44
5.4.15	SYS PLL 时钟配置寄存器 0.....	46
5.4.16	SYSPLL 时钟配置寄存器 1.....	47
5.4.17	DDR PLL 时钟配置寄存器 0.....	47
5.4.18	DDR PLL 时钟配置寄存器 1.....	47
5.4.19	VID PLL 时钟配置寄存器 0.....	48
5.4.20	VID PLL 时钟配置寄存器 1.....	48
5.4.21	设备时钟分频配置寄存器 0.....	48
5.4.22	设备时钟分频配置寄存器 1.....	49
5.4.23	设备时钟分频配置寄存器 2.....	50
5.4.24	设备时钟输出使能配置寄存器.....	51
5.4.25	打印系统设备时钟门控寄存器.....	52
5.4.26	打印系统设备复位控制寄存器.....	52
5.4.27	打印系统 GPIO0~15 复用配置寄存器.....	53
5.4.28	打印系统 GPIO16~31 复用配置寄存器.....	54
5.4.29	打印系统 GPIO32~47 复用配置寄存器.....	55
5.4.30	打印系统 GPIO48~57 复用配置寄存器.....	56
5.4.31	打印系统通用配置寄存器 0.....	57
5.4.32	打印系统通用配置寄存器 1.....	57
5.4.33	打印系统采样参数寄存器.....	57
5.4.34	打印系统高精度计数器.....	58
5.4.35	打印系统设备 DMA 路由基址配置寄存器.....	58
5.4.36	打印系统共享设备通用配置寄存器 0.....	59
5.4.37	打印系统共享设备通用配置寄存器 1.....	59



5.4.38	打印系统共享设备通用配置寄存器 2	1
5.4.39	打印系统启动入口配置寄存器	1
5.4.40	打印系统 0 号 MAILBOX 中断状态寄存器	1
5.4.41	打印系统 0 号 MAILBOX 中断使能寄存器	1
5.4.42	打印系统 0 号 MAILBOX 中断触发寄存器	1
5.4.43	打印系统 0 号 MAILBOX 中断清除寄存器	1
5.4.44	打印系统 0 号 MAILBOX 消息寄存器 0~7	1
5.4.45	打印系统 1 号 MAILBOX 中断状态寄存器	1
5.4.46	打印系统 1 号 MAILBOX 中断使能寄存器	1
5.4.47	打印系统 1 号 MAILBOX 中断触发寄存器	1
5.4.48	打印系统 1 号 MAILBOX 中断清除寄存器	1
5.4.49	打印系统 1 号 MAILBOX 消息寄存器 0~7	1
5.4.50	打印系统 2 号 MAILBOX 中断状态寄存器	1
5.4.51	打印系统 2 号 MAILBOX 中断使能寄存器	1
5.4.52	打印系统 2 号 MAILBOX 中断触发寄存器	1
5.4.53	打印系统 2 号 MAILBOX 中断清除寄存器	1
5.4.54	打印系统 2 号 MAILBOX 消息寄存器 0~7	1
5.4.55	打印系统 MAILBOX 中断路由寄存器	1
5.4.56	扫描系统设备时钟门控寄存器	1
5.4.57	扫描系统设备复位控制寄存器	1
5.4.58	扫描系统 GPIO0~15 复用配置寄存器	1
5.4.59	扫描系统 GPIO16~31 复用配置寄存器	1
5.4.60	扫描系统 GPIO32~36 复用配置寄存器	1
5.4.61	扫描系统通用配置寄存器	1
5.4.62	扫描系统采样参数寄存器	1
5.4.63	扫描系统高精度计数器	1
5.4.64	扫描系统启动入口配置寄存器	1
5.4.65	扫描系统 0 号 MAILBOX 中断状态寄存器	1
5.4.66	扫描系统 0 号 MAILBOX 中断使能寄存器	1
5.4.67	扫描系统 0 号 MAILBOX 中断触发寄存器	1
5.4.68	扫描系统 0 号 MAILBOX 中断清除寄存器	1
5.4.69	扫描系统 0 号 MAILBOX 消息寄存器 0~7	1
5.4.70	扫描系统 1 号 MAILBOX 中断状态寄存器	1
5.4.71	扫描系统 1 号 MAILBOX 中断使能寄存器	1
5.4.72	扫描系统 1 号 MAILBOX 中断触发寄存器	1
5.4.73	扫描系统 1 号 MAILBOX 中断清除寄存器	1
5.4.74	扫描系统 1 号 MAILBOX 消息寄存器 0~7	1
5.4.75	扫描系统 2 号 MAILBOX 中断状态寄存器	1
5.4.76	扫描系统 2 号 MAILBOX 中断使能寄存器	1
5.4.77	扫描系统 2 号 MAILBOX 中断触发寄存器	1
5.4.78	扫描系统 2 号 MAILBOX 中断清除寄存器	1
5.4.79	扫描系统 2 号 MAILBOX 消息寄存器 0~7	1
5.4.80	扫描系统 MAILBOX 中断路由寄存器	1
5.5	中断配置及路由	1
5.5.1	各系统中断概述	1
5.5.2	中断触发类型	1
5.5.3	中断相关寄存器描述	1
5.5.4	中断路由寄存器描述	1
6	DDR3 控制器	1
6.1	访问地址	1
6.2	DDR3 控制器寄存器	1



7	GMAC 控制器.....	1
7.1	寄存器描述.....	1
7.2	软件编程向导:.....	1
8	USB 控制器.....	1
8.1	总体概述.....	1
8.2	控制器寄存器.....	1
9	OTG 控制器.....	1
9.1	概述.....	1
9.2	寄存器描述.....	1
10	SPI-FLASH 控制器.....	1
10.1	SPI 控制器结构.....	1
10.2	配置寄存器.....	1
10.2.1	控制寄存器 (SPCR) .....	1
10.2.2	状态寄存器 (SPSR) .....	1
10.2.3	数据寄存器 (TxFIFO/RxFIFO) .....	1
10.2.4	外部寄存器 (SPER) .....	1
10.2.5	参数控制寄存器 (SFC_PARAM) .....	1
10.2.6	片选控制寄存器 (SFC_SOFTCS) .....	1
10.2.7	时序控制寄存器 (SFC_TIMING) .....	1
10.3	接口时序.....	1
10.3.1	SPI 主控制器接口时序.....	1
10.3.2	SPI Flash 访问时序.....	1
10.4	使用指南.....	1
10.4.1	SPI 主控制器的读写操作.....	1
10.4.2	硬件 SPI Flash 读.....	1
10.4.3	混合访问 SPI Flash 和 SPI 主控制器.....	1
11	SPI-I/O 控制器.....	1
11.1	SPI 控制器结构.....	1
11.2	配置寄存器.....	1
11.2.1	控制寄存器 1 (CR1) .....	1
11.2.2	控制寄存器 2 (CR2) .....	1
11.2.3	控制寄存器 3 (CR3) .....	1
11.2.4	控制寄存器 4 (CR4) .....	1
11.2.5	中断寄存器 (IER) .....	1
11.2.6	状态寄存器 1 (SR1) .....	1
11.2.7	状态寄存器 2 (SR2) .....	1
11.2.8	配置寄存器 1 (CFG1) .....	1
11.2.9	配置寄存器 2 (CFG2) .....	1
11.2.10	配置寄存器 3 (CFG3) .....	1
11.2.11	CRC 寄存器 1 (CRC1) .....	1
11.2.12	CRC 寄存器 2 (CRC2) .....	1
11.2.13	数据寄存器 (DR) .....	1
11.3	DMA 寄存器.....	1
11.3.1	DMA 中断状态寄存器 (ISR) .....	1
11.3.2	DMA 中断状态清除寄存器 (IFCR) .....	1
11.3.3	DMA 控制寄存器 n (CCRn) .....	1
11.3.4	DMA 计数寄存器 (CNDTRn) .....	1
11.3.5	DMA 地址寄存器 (CMARn) .....	1
12	I2C 控制器.....	1
12.1	概述.....	1
12.2	I2C 控制器结构.....	1





12.3	I2C 控制器寄存器说明.....	1
12.3.1	分频锁存器低字节寄存器 (PRERlo) .....	1
12.3.2	分频锁存器高字节寄存器 (PRERhi) .....	1
12.3.3	控制寄存器 (CTR) .....	1
12.3.4	发送/接收数据寄存器 (TXD/RXD) .....	1
12.3.5	命令控制寄存器 (CR) .....	1
12.3.6	状态寄存器 (SR) .....	1
12.3.7	总线死锁时间寄存器 (BLTOP) .....	1
12.3.8	从设备地址寄存器 (SADDR) .....	1
13	打印接口.....	1
13.1	概述.....	1
13.2	配置寄存器.....	1
14	扫描接口.....	1
14.1	概述.....	1
14.2	配置寄存器.....	1
15	PMIO.....	1
15.1	概述.....	1
15.2	配置寄存器.....	1
16	JPEG 编码器.....	1
16.1	概述.....	1
16.2	寄存器描述.....	1
16.2.1	命令寄存器 (CMD) .....	1
16.2.2	状态寄存器 (STS) .....	1
16.2.3	量化表地址寄存器 (Q_TABLE_ADDR) .....	1
16.2.4	块数量寄存器 (BLOCK_OF_IMAGE) .....	1
16.2.5	输入地址寄存器 (RGB_IN_ADDR) .....	1
16.2.6	输出地址寄存器 (JPEG_OUT_ADDR) .....	1
16.2.7	编码字节数寄存器 (ENCODING_BYTE) .....	1
16.2.8	末尾字节位数寄存器 (ENF_BIT_CNT) .....	1
16.2.9	跟随字节寄存器 (TRAILING) .....	1
16.3	编程示例.....	1
17	JBIG85 解码器.....	1
17.1	概述.....	1
17.2	寄存器描述.....	1
17.3	编程说明.....	1
18	UART 控制器.....	1
18.1	概述.....	1
18.2	控制器结构.....	1
18.3	寄存器描述.....	1
18.3.1	数据寄存器 (DAT) .....	1
18.3.2	中断使能寄存器 (IER) .....	1
18.3.3	中断标识寄存器 (IIR) .....	1
18.3.4	FIFO 控制寄存器 (FCR) .....	1
18.3.5	线路控制寄存器 (LCR) .....	1
18.3.6	MODEM 控制寄存器 (MCR) .....	1
18.3.7	线路状态寄存器 (LSR) .....	1
18.3.8	MODEM 状态寄存器 (MSR) .....	1
18.3.9	分频锁存器.....	1
19	加解密.....	1
19.1	DES.....	1
19.1.1	DES 功能概述.....	1



19.1.2	DES 访问地址.....	1
19.1.3	DES 寄存器描述.....	1
19.2	AES.....	1
19.2.1	AES 功能概述.....	1
19.2.2	AES 访问地址.....	1
19.2.3	AES 寄存器描述.....	1
20	eMMC 控制器.....	1
20.1	功能概述.....	1
20.2	访问地址及引脚复用.....	1
20.3	寄存器描述.....	1
20.4	专用 DMA 控制器.....	1
20.4.1	结构描述.....	1
20.4.2	DMA 描述符.....	1
20.5	软件配置流程.....	1
20.5.1	eMMC 正常读写流程.....	1
20.5.2	eMMC 初始化流程.....	1
20.5.3	DDR 模式设置.....	1
21	SDIO 控制器.....	1
21.1	功能概述.....	1
21.2	访问地址及引脚复用.....	1
21.3	寄存器描述.....	1
21.4	软件编程指南.....	1
21.4.1	SD Memory 卡软件编程说明.....	1
21.4.2	SDIO 卡软件编程说明.....	1
21.4.3	DDR 模式设置.....	1
21.5	支持 SDIO 型号.....	1
22	PWM 控制器.....	1
22.1	概述.....	1
22.2	访问地址及引脚复用.....	1
22.3	寄存器描述.....	1
22.4	功能说明.....	1
22.4.1	脉宽调制功能.....	1
22.4.2	脉冲测量功能.....	1
22.4.3	防死区功能.....	1
23	HPET 控制器.....	1
23.1	概述.....	1
23.2	访问地址.....	1
23.3	寄存器描述.....	1
23.3.1	General Capabilities and ID Register.....	1
23.3.2	General Configuration Register.....	1
23.3.3	General Interrupt Status Register.....	1
23.3.4	Main Counter Value Register.....	1
23.3.5	Timer N Configuration and Capabilities Register.....	1
23.3.6	Timer N Comparator Value Register.....	1
24	DA 控制器.....	1
24.1	概述.....	1
24.2	寄存器描述.....	1
24.2.1	DAC 使能寄存器 (DACEN) .....	1
24.2.2	DAC 数值寄存器 (DACDATA) .....	1
25	电源管理模块.....	1
25.1	概述.....	1



25.2	动态电源管理.....	1
25.2.1	DFS 功能描述.....	1
25.2.2	DPM 功能描述.....	1
25.2.3	设备唤醒功能描述.....	1
25.3	寄存器描述.....	1
25.3.1	DPM 寄存器描述.....	1
25.3.2	DFS 寄存器描述.....	1
25.3.3	WDT 寄存器描述.....	1
26	RTC.....	1
26.1	概述.....	1
26.2	寄存器描述.....	1
26.2.1	寄存器地址列表.....	1
26.2.2	SYS_TOYWRITE0.....	1
26.2.3	SYS_TOYWRITE1.....	1
26.2.4	SYS_TOYREAD0.....	1
26.2.5	SYS_TOYREAD1.....	1
26.2.6	SYS_TOYMATCH0/1/2.....	1
26.2.7	SYS_RTCCTRL.....	1
26.2.8	SYS_RTCWRITE.....	1
26.2.9	SYS_RTCREAD.....	1
26.2.10	SYS_RTCMATCH0/1/2.....	1
27	GPIO.....	1
27.1	概述.....	1
27.2	寄存器描述.....	1
27.3	访问地址.....	1
27.4	控制寄存器.....	1
27.4.1	GPIO 方向控制.....	1
27.4.2	GPIO 输出值.....	1
27.4.3	GPIO 输入值.....	1
27.4.4	GPIO 中断使能.....	1
27.4.5	GPIO 中断极性.....	1
27.4.6	GPIO 中断边沿.....	1
27.4.7	GPIO 中断清除.....	1
27.4.8	GPIO 中断状态.....	1
27.4.9	GPIO 中断双沿模式.....	1



## 图目录

图 1- 1 龙芯 2P0500 芯片结构图.....	2
图 3- 1 时钟结构.....	15
图 3- 2 SYS PLL 结构图.....	16
图 3- 3 DDR PLL 时钟结构.....	17
图 3- 4 SOC PLL 时钟结构.....	17
图 5- 1 展频 PLL 概念性结构.....	25
图 5- 2 主系统传统 IO 中断路由示意图.....	76
图 5- 3 主系统扩展 IO 中断路由示意图.....	78
图 5- 4 打印系统 IO 中断路由示意图.....	80
图 5- 5 扫描系统 IO 中断路由示意图.....	82
图 10- 1 SPI 控制器结构.....	104
图 10- 2 SPI 主控制器接口时序.....	107
图 10- 3 SPI Flash 标准读时序.....	107
图 10- 4 SPI Flash 快速读时序.....	107
图 10- 5 SPI Flash 双向 I/O 读时序.....	108
图 12- 1 I2C 主控制器结构.....	119
图 17- 1 JBIG85 解码结构示意图.....	131
图 18- 1 UART 控制器结构.....	135
图 20- 1 eMMC 初始化流程图.....	159
图 22- 1 防死区功能.....	174
图 25- 1 芯片电源域划分示意图.....	183
图 25- 2 处理器核 DFS 控制结构框图.....	184
图 25- 3 处理器核 DFS 操作流程框图.....	185
图 25- 4 设备 DPM 控制结构框图.....	185
图 25- 5 功能模块 DPM 状态切换流程图.....	186
图 25- 6 芯片待机唤醒操作流程框图.....	187
图 25- 7 芯片休眠唤醒操作流程框图.....	188

## 表目录

表 2- 1 信号类型代码.....	6
表 2- 2 DDR3 SDRAM 控制器接口信号.....	6
表 2- 3 USB 接口信号.....	7
表 2- 4 GMAC 接口信号.....	7
表 2- 5 SPI 接口信号.....	8
表 2- 6 UART 接口信号.....	8
表 2- 7 I <sup>2</sup> C 接口信号.....	8
表 2- 8 Printer 接口信号.....	8
表 2- 9 Scanner 接口信号.....	9
表 2- 10 DA 接口信号.....	9
表 2- 11 SDIO 接口信号.....	9
表 2- 12 PMIO 接口信号.....	9
表 2- 13 PWM 接口信号.....	9
表 2- 14 电源地接口.....	10
表 2- 15 测试接口.....	10
表 2- 16 JTAG 接口.....	10
表 2- 17 系统相关信号.....	11
表 2- 18 上电配置信号.....	11
表 2- 19 主系统功能引脚复用关系表.....	11
表 2- 20 打印系统功能引脚复用关系表.....	12
表 2- 21 扫描系统功能引脚复用关系表.....	14
表 5- 1 初始化配置信号.....	21
表 5- 2 地址空间分配之主系统视角.....	21
表 5- 3 地址空间分配之打印系统视角.....	23
表 5- 4 地址空间分配之扫描系统视角.....	23
表 5- 5 地址空间分配之 DMA 视角.....	24
表 5- 6 PLL 相关配置信号说明表.....	25
表 5- 7 主系统芯片配置寄存器列表.....	26
表 5- 8 打印系统芯片配置寄存器列表.....	28
表 5- 9 扫描系统芯片配置寄存器列表.....	30
表 5- 10 通用配置寄存器 0.....	33
表 5- 11 通用配置寄存器 1.....	33
表 5- 12 通用配置寄存器 1.....	35

表 5- 13 通用配置寄存器 3.....	36
表 5- 14 通用配置寄存器 4.....	37
表 5- 15 通用配置寄存器 5.....	38
表 5- 16 主系统采样参数寄存器.....	38
表 5- 17 主系统高精度计数器 0.....	39
表 5- 18 主系统高精度计数器 1.....	39
表 5- 19 主系统 GPIO0~15 复用配置寄存器.....	39
表 5- 20 主系统 GPIO16~31 复用配置寄存器.....	40
表 5- 21 主系统 GPIO32~43 复用配置寄存器.....	41
表 5- 22 USB PHY 配置寄存器 0.....	42
表 5- 23 USB PHY 配置寄存器 1.....	44
表 5- 24 SYS-PLL 时钟配置寄存器 0.....	46
表 5- 25 SYS-PLL 时钟配置寄存器 1.....	47
表 5- 26 DDR-PLL 时钟配置寄存器 0.....	47
表 5- 27 DDR-PLL 时钟配置寄存器 1.....	47
表 5- 28 VID-PLL 时钟配置寄存器 0.....	48
表 5- 29 VID-PLL 时钟配置寄存器 1.....	48
表 5- 30 设备时钟分频配置寄存器.....	48
表 5- 31 设备时钟分频配置寄存器.....	49
表 5- 32 设备时钟分频配置寄存器.....	50
表 5- 33 设备时钟输出使能配置寄存器.....	51
表 5- 34 打印系统设备时钟门控寄存器.....	52
表 5- 35 打印系统设备复位控制寄存器.....	52
表 5- 36 打印系统 GPIO0~15 复用配置寄存器.....	53
表 5- 37 打印系统 GPIO16~31 复用配置寄存器.....	54
表 5- 38 打印系统 GPIO32~47 复用配置寄存器.....	55
表 5- 39 打印系统 GPIO48~57 复用配置寄存器.....	56
表 5- 40 通用配置寄存器 0.....	57
表 5- 41 通用配置寄存器 1.....	57
表 5- 42 芯片采样参数寄存器.....	58
表 5- 43 打印系统高精度计数器.....	58
表 5- 44 打印系统设备 DMA 路由基址配置寄存器.....	59
表 5- 45 打印系统共享设备通用配置寄存器 0.....	59
表 5- 46 打印系统共享设备通用配置寄存器 1.....	59



表 5- 47 打印系统共享设备通用配置寄存器 2.....	60
表 5- 48 打印系统启动入口配置寄存器.....	62
表 5- 49 打印系统 0 号 MAILBOX 中断状态寄存器.....	62
表 5- 50 打印系统 0 号 MAILBOX 中断使能寄存器.....	62
表 5- 51 打印系统 0 号 MAILBOX 中断触发寄存器.....	63
表 5- 52 打印系统 0 号 MAILBOX 中断清除寄存器.....	63
表 5- 53 打印系统 0 号 MAILBOX 消息寄存器 0~7.....	63
表 5- 54 打印系统 1 号 MAILBOX 中断状态寄存器.....	63
表 5- 55 打印系统 1 号 MAILBOX 中断使能寄存器.....	64
表 5- 56 打印系统 1 号 MAILBOX 中断触发寄存器.....	64
表 5- 57 打印系统 1 号 MAILBOX 中断清除寄存器.....	64
表 5- 58 打印系统 1 号 MAILBOX 消息寄存器 0~7.....	64
表 5- 59 打印系统 2 号 MAILBOX 中断状态寄存器.....	65
表 5- 60 打印系统 2 号 MAILBOX 中断使能寄存器.....	65
表 5- 61 打印系统 2 号 MAILBOX 中断触发寄存器.....	65
表 5- 62 打印系统 2 号 MAILBOX 中断清除寄存器.....	65
表 5- 63 打印系统 0 号 MAILBOX 消息寄存器 0~7.....	66
表 5- 64 打印系统 MAILBOX 中断路由寄存器.....	66
表 5- 65 扫描系统设备时钟门控寄存器.....	67
表 5- 66 扫描系统设备复位控制寄存器.....	67
表 5- 67 扫描系统 GPIO0~15 复用配置寄存器.....	67
表 5- 68 扫描系统 GPIO16~31 复用配置寄存器.....	68
表 5- 69 扫描系统 GPIO32~36 复用配置寄存器.....	69
表 5- 70 扫描系统通用配置寄存器 0.....	70
表 5- 71 扫描系统采样参数寄存器.....	70
表 5- 72 扫描系统高精度计数器.....	71
表 5- 73 扫描系统启动入口配置寄存器.....	71
表 5- 74 扫描系统 0 号 MAILBOX 中断状态寄存器.....	71
表 5- 75 扫描系统 0 号 MAILBOX 中断使能寄存器.....	72
表 5- 76 扫描系统 0 号 MAILBOX 中断触发寄存器.....	72
表 5- 77 扫描系统 0 号 MAILBOX 中断清除寄存器.....	72
表 5- 78 扫描系统 0 号 MAILBOX 消息寄存器 0~7.....	72
表 5- 79 扫描系统 1 号 MAILBOX 中断状态寄存器.....	73
表 5- 80 扫描系统 1 号 MAILBOX 中断使能寄存器.....	73





表 5- 81 扫描系统 1 号 MAILBOX 中断触发寄存器.....	73
表 5- 82 扫描系统 1 号 MAILBOX 中断清除寄存器.....	73
表 5- 83 扫描系统 1 号 MAILBOX 消息寄存器 0~7.....	73
表 5- 84 扫描系统 2 号 MAILBOX 中断状态寄存器.....	74
表 5- 85 扫描系统 2 号 MAILBOX 中断使能寄存器.....	74
表 5- 86 扫描系统 2 号 MAILBOX 中断触发寄存器.....	74
表 5- 87 扫描系统 2 号 MAILBOX 中断清除寄存器.....	75
表 5- 88 扫描系统 0 号 MAILBOX 消息寄存器 0~7.....	75
表 5- 89 打印系统 MAILBOX 中断路由寄存器.....	75
表 5- 90 主系统传统 IO 中断寄存器列表.....	77
表 5- 91 主系统 IO 扩展中断寄存器列表.....	79
表 5- 92 打印系统 IO 中断寄存器列表.....	81
表 5- 93 扫描系统 IO 中断寄存器列表.....	83
表 5- 94 主系统传统中断控制寄存器属性.....	83
表 5- 95 传统中断控制寄存器地址.....	85
表 5- 96 扩展中断控制寄存器属性.....	86
表 5- 97 打印系统 IO 中断控制寄存器属性.....	87
表 5- 98 打印系统中断控制寄存器描述.....	89
表 5- 99 扫描系统 IO 中断控制寄存器属性.....	90
表 5- 100 扫描系统中断控制寄存器描述.....	92
表 5- 101 传统中断路由寄存器的说明.....	93
表 5- 102 传统中断路由寄存器地址.....	93
表 5- 103 扩展中断路由寄存器的说明.....	94
表 5- 104 打印系统中断路由寄存器的说明.....	95
表 5- 105 扫描系统中断路由寄存器的说明.....	95
表 6- 1 内存控制器地址空间分配.....	96
表 6- 2 DDR3 SDRAM 配置参数寄存器.....	97
表 8- 1 USB 控制器地址空间分布.....	102
表 10- 1 SPI 控制器地址空间分布.....	104
表 10- 2 SPI0 配置寄存器列表.....	105
表 10- 3 SPI 控制寄存器 (SPCR).....	105
表 10- 4 SPI 状态寄存器 (SPSR).....	105
表 10- 5 SPI 数据寄存器 (TxFIFO/RxFIFO).....	105
表 10- 6 SPI 外部寄存器 (SPER).....	106





表 10- 7	SPI 分频系数.....	106
表 10- 8	SPI 参数控制寄存器 (SFC_PARAM) .....	106
表 10- 9	SPI 片选控制寄存器 (SFC_SOFTCS) .....	106
表 10- 10	SPI 时序控制寄存器 (SFC_TIMING) .....	106
表 11- 1	SPI 控制器地址空间分布.....	110
表 11- 2	SPI-IO 配置寄存器列表.....	110
表 11- 3	SPI 控制寄存器 1 (CR1) .....	110
表 11- 4	SPI 控制寄存器 2 (CR2) .....	111
表 11- 5	SPI 控制寄存器 3 (CR3) .....	111
表 11- 6	SPI 控制寄存器 4 (CR4) .....	112
表 11- 7	SPI 中断寄存器 (IER) .....	112
表 11- 8	SPI 状态寄存器 1 (SR1) .....	112
表 11- 9	SPI 状态寄存器 2 (SR2) .....	113
表 11- 10	SPI 配置寄存器 1 (CFG1) .....	114
表 11- 11	SPI 配置寄存器 2 (CFG2) .....	114
表 11- 12	SPI 配置寄存器 3 (CFG3) .....	114
表 11- 13	SPI CRC 寄存器 1 (CRC1) .....	115
表 11- 14	SPI CRC 寄存器 2 (CRC2) .....	115
表 11- 15	SPI 数据寄存器 (DR) .....	115
表 11- 16	SPI-IO DMA 寄存器列表.....	116
表 11- 17	SPI DMA 中断状态寄存器 (ISR) .....	116
表 11- 18	SPI DMA 中断状态清除寄存器 (IFCR) .....	116
表 11- 19	SPI DMA 控制寄存器 n (CCRn) .....	117
表 11- 20	SPI DMA 计数寄存器 (CNDTRn) .....	118
表 11- 21	SPI DMA 地址寄存器 (CMARn) .....	118
表 12- 1	I2C 配置寄存器列表.....	120
表 12- 2	分频锁存器低字节寄存器.....	120
表 12- 3	分频锁存器高字节寄存器.....	120
表 12- 4	控制寄存器.....	121
表 12- 5	发送/接收寄存器.....	121
表 12- 6	命令寄存器.....	122
表 12- 7	状态寄存器.....	122
表 12- 8	总线死锁时间寄存器.....	123
表 12- 9	从设备地址寄存器.....	123



表 13-1 打印控制器地址空间分布.....	124
表 14-1 SCAN 控制器地址空间分布.....	125
表 15-1 各系统 PMIO 控制器地址空间分布.....	126
表 16-1 JPEG 编码器寄存器列表.....	127
表 16-2 命令寄存器.....	127
表 16-3 状态寄存器.....	128
表 16-4 量化表地址寄存器.....	128
表 16-5 块数量寄存器.....	128
表 16-6 输入地址寄存器.....	129
表 16-7 输出地址寄存器.....	129
表 16-8 编码字节数寄存器.....	129
表 16-9 末尾字节位数寄存器.....	129
表 16-10 跟随字节寄存器.....	129
表 17- 1 JBIG 配置地址.....	132
表 17- 2 JBIG 配置寄存器.....	132
表 17- 3 jbig_top_cfg 寄存器.....	132
表 17- 4 error_code_reg 寄存器.....	133
表 18- 1 UART 配置寄存器列表.....	136
表 18- 2 数据传输寄存器.....	137
表 18- 3 中断使能寄存器.....	137
表 18- 4 中断源寄存器.....	137
表 18- 5 中断控制功能表.....	137
表 18- 6 FIFO 控制寄存器.....	138
表 18- 7 线路控制寄存器.....	138
表 18- 8 Modem 控制寄存器.....	139
表 18- 9 线路状态寄存器.....	139
表 18- 10 Modem 状态寄存器.....	140
表 18- 11 分频锁存器低 8 位寄存器.....	141
表 18- 12 分频锁存器高 8 位寄存器.....	141
表 18- 13 分频锁存器小数位寄存器.....	141
表 19- 1 DES 寄存器列表.....	142
表 19- 2 Comand 寄存器.....	143
表 19- 3 AES 寄存器列表.....	144
表 19- 4 Comand 寄存器.....	145

表 20- 1 eMMC 内部寄存器物理地址构成.....	147
表 20- 2 EMMC_CON 寄存器.....	147
表 20- 3 EMMC_CON 寄存器位域描述.....	147
表 20- 4 EMMC_PRE 寄存器.....	147
表 20- 5 EMMC_PRE 寄存器位域描述.....	148
表 20- 6 EMMC_CMD_ARG 寄存器.....	148
表 20- 7 EMMC_CMD_ARG 寄存器位域描述.....	148
表 20- 8 EMMC_CMD_CON 寄存器.....	148
表 20- 9 EMMC_CMD_CON 寄存器位域描述.....	148
表 20- 10 EMMC_CMD_STA 寄存器.....	149
表 20- 11 EMMC_CMD_STA 寄存器位域描述.....	149
表 20- 12 EMMC_RSP0 寄存器.....	149
表 20- 13 EMMC_RSP0 寄存器位域描述.....	149
表 20- 14 EMMC_RSP1 寄存器.....	149
表 20- 15 EMMC_RESP1 寄存器位域描述.....	149
表 20- 16 EMMC_RSP2 寄存器.....	150
表 20- 17 EMMC_RSP2 寄存器位域描述.....	150
表 20- 18 EMMC_RSP3 寄存器.....	150
表 20- 19 EMMC_RSP3 寄存器位域描述.....	150
表 20- 20 EMMC_DTIMER 寄存器.....	150
表 20- 21 EMMC_DTIMER 寄存器位域描述.....	150
表 20- 22 EMMC_BSIZE 寄存器.....	150
表 20- 23 EMMC_BSIZE 寄存器位域描述.....	150
表 20- 24 EMMC_DAT_CON 寄存器.....	150
表 20- 25 EMMC_DAT_CON 寄存器位域描述.....	151
表 20- 26 EMMC_DAT_CNT 寄存器.....	151
表 20- 27 EMMC_DAT_CNT 寄存器位域描述.....	151
表 20- 28 EMMC_DAT_STA 寄存器.....	151
表 20- 29 EMMC_DAT_STA 寄存器位域描述.....	152
表 20- 30 EMMC_FIFO_STA 寄存器.....	152
表 20- 31 EMMC_FIFO_STA 寄存器位域描述.....	152
表 20- 32 EMMC_INT_MASK 寄存器.....	152
表 20- 33 EMMC_INT_MASK 寄存器位域描述.....	153
表 20- 34 EMMC_DAT 寄存器.....	153



表 20- 35 EMMC_DAT 寄存器位域描述.....	153
表 20- 36 EMMC_INT_EN 寄存器.....	153
表 20- 37 EMMC_INT_EN 寄存器位域描述.....	153
表 20- 38 DLL_MASTER_VAL 寄存器.....	154
表 20- 39 DLL_MASTER_VAL 寄存器位域描述.....	154
表 20- 40 DLL_CON 寄存器.....	154
表 20- 41 DLL_CON 寄存器位域描述.....	154
表 20- 42 PARAM_DELAY 寄存器.....	154
表 20- 43 PARAM_DELAY 寄存器位域描述.....	154
表 20- 44 SDIO_EMMC_SEL 寄存器.....	155
表 20- 45 SDIO_EMMC_SEL 寄存器位域描述.....	155
表 20- 46 DMA_ORDER_ADDR_LOW 寄存器.....	155
表 20- 47 DMA_SADDR 寄存器.....	156
表 20- 48 DMA_DADDR 寄存器.....	156
表 20- 49 DMA_LENGTH 寄存器.....	156
表 20- 50 DMA_STEP_LENGTH 寄存器.....	156
表 20- 51 DMA_STEP_TIMERS 寄存器.....	157
表 20- 52 DMA_CMD 寄存器.....	157
表 20- 53 DMA 写状态描述.....	157
表 20- 54 DMA 读状态描述.....	158
表 20- 55 DMA_ADDR_HIGH 寄存器.....	158
表 20- 56 DMA_SADDR_HIGH 寄存器.....	158
表 21- 1 SDIO 内部寄存器物理地址构成.....	161
表 21- 2 SDI_CON 寄存器.....	161
表 21- 3 SDI_CON 寄存器位域描述.....	161
表 21- 4 SDI_PRE 寄存器.....	162
表 21- 5 SDI_PRE 寄存器位域描述.....	162
表 21- 6 SDI_CMD_ARG 寄存器.....	162
表 21- 7 SDI_CMD_ARG 寄存器位域描述.....	162
表 21- 8 SDI_CMD_CON 寄存器.....	162
表 21- 9 SDI_CMD_CON 寄存器位域描述.....	162
表 21- 10 SDI_CMD_STA 寄存器.....	163
表 21- 11 SDI_CMD_STA 寄存器位域描述.....	163
表 21- 12 SDI_RSPO 寄存器.....	163



表 21- 13 SDI_RSP0 寄存器位域描述.....	163
表 21- 14 SDI_RSP1 寄存器.....	163
表 21- 15 SDI_RSP1 寄存器位域描述.....	163
表 21- 16 SDI_RSP2 寄存器.....	164
表 21- 17 SDI_RSP2 寄存器位域描述.....	164
表 21- 18 SDI_RSP3 寄存器.....	164
表 21- 19 SDI_RSP3 寄存器位域描述.....	164
表 21- 20 SDI_DTIMER 寄存器.....	164
表 21- 21 SDI_DTIMER 寄存器位域描述.....	164
表 21- 22 SDI_BSIZE 寄存器.....	164
表 21- 23 SDI_BSIZE 寄存器位域描述.....	164
表 21- 24 SDI_DAT_CON 寄存器.....	164
表 21- 25 SDI_DAT_CON 寄存器位域描述.....	165
表 21- 26 SDI_DAT_CNT 寄存器.....	165
表 21- 27 SDI_DAT_CNT 寄存器位域描述.....	165
表 21- 28 SDI_DAT_STA 寄存器.....	165
表 21- 29 SDI_DAT_STA 寄存器位域描述.....	165
表 21- 30 SDI_FIFO_STA 寄存器.....	166
表 21- 31 SDI_FIFO_STA 寄存器位域描述.....	166
表 21- 32 SDI_INT_MASK 寄存器.....	166
表 21- 33 SDI_INT_MASK 寄存器位域描述.....	167
表 21- 34 SDI_DAT 寄存器.....	167
表 21- 35 SDI_DAT 寄存器位域描述.....	167
表 21- 36 SDI_INT_EN 寄存器.....	167
表 21- 37 SDI_INT_EN 寄存器位域描述.....	167
表 21- 38 dll_master_val 寄存器.....	168
表 21- 39 dll_master_val 寄存器位域描述.....	168
表 21- 40 dll_con 寄存器.....	168
表 21- 41 dll_con 寄存器位域描述.....	168
表 21- 42 param_delay 寄存器.....	168
表 21- 43 param_delay 寄存器位域描述.....	169
表 21- 44 sdio_emmc_sel 寄存器.....	169
表 21- 45 sdio_emmc_sel 寄存器位域描述.....	169
表 22- 1 PWM 寄存器地址.....	172



表 22- 2 PWM 寄存器列表.....	172
表 22- 3 PWM 控制寄存器设置.....	172
表 23- 1 HPET 寄存器地址.....	175
表 23- 2 HPET 寄存器.....	175
表 23- 3 General Capabilities ID 寄存器.....	176
表 23- 4 General Configuration 寄存器.....	176
表 23- 5 General Interrupt Status 寄存器.....	177
表 23- 6 Main Counter Value 寄存器.....	177
表 23- 7 Timer N Configuration Capabilities 寄存器.....	177
表 23- 8 Timer N Comparator Value 寄存器.....	178
表 24- 1 DACEN 寄存器.....	180
表 24- 2 DACDATA 寄存器.....	180
表 25- 1 芯片各电源域划分.....	182
表 25- 2 芯片各功能模块电源管理功能划分.....	183
表 25- 3 DPM_EN 寄存器描述.....	189
表 25- 4 PWRUP_SEL 寄存器描述.....	189
表 25- 5 DPM_TGT 寄存器描述.....	190
表 25- 6 DPM_STS 寄存器描述.....	191
表 25- 7 WAIT_TIME0 寄存器描述.....	191
表 25- 8 WAIT_TIME1 寄存器描述.....	192
表 25- 9 WAIT_TIME2 寄存器描述.....	192
表 25- 10 WAIT_TIME3 寄存器描述.....	192
表 25- 11 CG_CFG 寄存器描述.....	192
表 25- 12 NODE DVFS Control 寄存器描述.....	194
表 25- 13 LA132 DVFS Control 寄存器描述.....	194
表 25- 14 WDT 寄存器地址.....	195
表 25- 15 WD_EN 寄存器描述.....	195
表 25- 16 WD_SET 寄存器描述.....	195
表 25- 17 WD_TIMER 寄存器描述.....	196
表 25- 18 PLTRST_SET 寄存器描述.....	196
表 26- 1 RTC 寄存器列表.....	197
表 26- 2 TOY 计数器低 32 位写入寄存器.....	198
表 26- 3 TOY 计数器高 32 位写入寄存器.....	198
表 26- 4 TOY 计数器低 32 位读出寄存器.....	198



表 26- 5 TOY 计数器高 32 位读出寄存器.....	199
表 26- 6 TOY 计数器中断寄存器 0/1/2.....	199
表 26- 7 RTC 定时器中断寄存器 0/1/2.....	199
表 26- 8 RTC 计数器写入寄存器.....	200
表 26- 9 RTC 计数器读出寄存器.....	200
表 26- 10 RTC 定时器中断寄存器 0/1/2.....	200
表 27- 1 GPIO 寄存器地址.....	201
表 27- 2 GPIO 控制寄存器.....	201
表 27- 3 GPIO 模块内部寄存器物理地址.....	202
表 27- 4 按位控制 GPIO 配置寄存器地址.....	202
表 27- 5 按字节控制 GPIO 配置寄存器地址.....	202
表 27- 6 GPIO 方向按位控制寄存器.....	203
表 27- 7 GPIO 方向按字节控制寄存器.....	203
表 27- 8 GPIO 按位输出寄存器.....	203
表 27- 9 GPIO 按字节输出寄存器.....	203
表 27- 10 GPIO 按位输入寄存器.....	203
表 27- 11 GPIO 按字节输入寄存器.....	204
表 27- 12 GPIO 按位中断使能寄存器.....	204
表 27- 13 GPIO 按字节中断使能寄存器.....	204
表 27- 14 GPIO 按位中断极性寄存器.....	204
表 27- 15 GPIO 按字节中断极性寄存器.....	204
表 27- 16 GPIO 按位中断边沿寄存器.....	205
表 27- 17 GPIO 按字节中断边沿寄存器.....	205
表 27- 18 GPIO 按位中断清除寄存器.....	205
表 27- 19 GPIO 按字节中断清除寄存器.....	206
表 27- 20 GPIO 按位中断状态寄存器.....	206
表 27- 21 GPIO 按字节中断状态寄存器.....	206
表 27- 22 GPIO 按位中断双沿寄存器.....	206
表 27- 23 GPIO 按字节中断双沿寄存器.....	206





# 1 概述

龙芯 2P0500 芯片是一款适用于单/多功能打印机主控 SOC 芯片，支持打印、扫描、网络等常用政企办公需求，支持 A4 常用打印纸张幅面的图像处理。龙芯 2P0500 打印主控芯片是打印整机中核心控制部件，主要用于打印数据接收、解析和处理，打印引擎控制，扫描时序控制，扫描数据处理，图像处理及马达控制等。

龙芯 2P0500 具有以下关键特性：

- 集成一个 LA364 三发射龙芯处理器核，L1 Cache(I/D) 32KB，L2 Cache 512KB
- 集成两个 LA132 单发射龙芯处理器核
- 集成 1 个 32 位 DDR3 控制器
- 集成 2 个 10M/100M/1000M 自适应 GMAC，支持 RGMII/MI I
- 集成 3 个 USB2.0 HOST 接口，其中 1 个可配置为 OTG 接口
- 集成 4 个 SPI 控制器，1 路支持系统启动
- 集成 1 个打印控制器，支持 8 路 LSU 机芯控制
- 集成 1 个扫描控制器，支持 1 路 AFE/CIS 控制
- 集成 1 个 4 路 8 位 DA 接口
- 集成 4 路 I2C 控制器
- 集成 8 路 UART 控制器
- 集成 2 个 SDIO 控制器
- 集成 40 路 PWM 控制器
- 集成 3 组 PMIO 控制器
- 集成 139 路复用 GPIO
- 集成 1 个温度传感器
- 集成 RTC/HPET
- 集成看门狗电路
- 集成动态功耗控制模块，支持 DFS/DPM
- 集成中断控制器，支持灵活的中断设置

## 1.1 体系结构框图

龙芯 2P0500 内部采用多级总线结构。一级交叉开关连接一个处理器核、一个二级 Cache 以及 IO 子网络 (Cache 访问路径)。二级 Cache 及 IODMA、内存控制器、GMAC、USB、IMAGE、PRINT、SCAN 等设备共享系统互连网络。低速外设 (I2C、UART 等) 作为一个集合加在南桥总线上。



龙芯 2P0500 芯片结构图如图 1-1 所示：

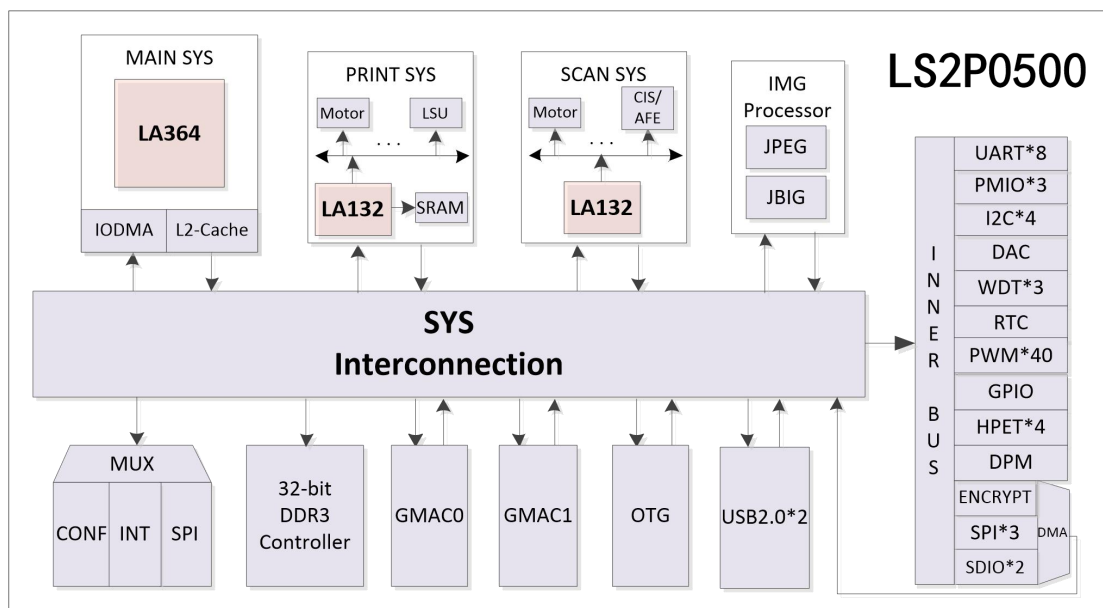


图1- 1 龙芯 2P0500 芯片结构图

## 1.2 芯片主要功能

### 1.2.1 处理器核

- LA364、LA132
- LoongArch 体系结构兼容
- 包括 1 个全流水的 64 位双精度浮点乘加部件
- 32KB 数据 Cache 和 32KB 的指令 Cache
- 512KB 共享二级 Cache
- 通过目录协议维护 I/O DMA 访问的 Cache 一致性
- JTAG 支持

### 1.2.2 内存接口

- 32 位 DDR3 控制器，典型工作频率 400MHz
- 不支持 ECC
- 可配置为 32/16 位模式
- 支持命令调度

### 1.2.3 USB 控制器

- 3 个独立的 USB2.0 的 HOST 端口
- 其中 1 端口可配置为 OTG 模式
- 兼容 USB1.1、USB2.0
- 内部 EHCI 控制和实现高速传输可达 480Mbps

- 内部 OHCI 控制和实现全速和低速传输
- 低功耗管理

#### 1.2.4 GMAC 控制器

- 两路 10/100/1000Mbps 自适应以太网 MAC
- 双网卡均兼容 IEEE 802.3
- 对外部 PHY 实现 RGMII/MII 接口
- 半双工/全双工自适应
- Timestamp 功能
- 半双工时，支持碰撞检测与重发（CSMA/CD）协议
- 支持 CRC 校验码的自动生成与校验，支持前置符生成与删除

#### 1.2.5 SPI

- 双缓冲接收器
- 极性和相位可编程的串行时钟
- 主模式支持
- 支持到 4 个的变长字节传输
- 支持系统启动(仅 SPI0 支持)
- 支持标准读、连续地址读、快速读、双路 I/O 等 SPI Flash 读模式

#### 1.2.6 UART

- 1 个全功能 UART 和流控 TXD,RXD,CTS, RTS, DSR,DTR,DCD, RI
- 最多 8 个 UART 接口
- 两路全双工异步数据接收/发送
- 可编程的数据格式
- 16 位可编程时钟计数器
- 支持接收超时检测
- 带仲裁的多中断系统

#### 1.2.7 I<sup>2</sup>C

- 履行双向同步串行协议
- 实现主/从设备操作
- 能够支持多主设备的总线
- 总线的时钟频率可编程
- 可以产生开始/停止/应答等操作
- 能够对总线的状态进行探测

- 支持低速和快速模式
- 支持 7 位寻址和 10 位寻址
- 支持时钟延伸和等待状态

### 1.2.8 打印接口

- 支持 JBIG85 解码
- 支持 8 路独立机芯控制
- 支持四色彩打功能

### 1.2.9 扫描接口

- 支持 JPEG 编码
- 支持 1 路 AFE、CIS 控制

### 1.2.10 DA

- 支持 4 路 8 位 DA 接口

### 1.2.11 PMIO

- 3 组 PMIO 控制器
- 支持共 80 路可编程多功能 IO

### 1.2.12 PWM

- 32 位计数器
- 支持脉冲生成及捕获
- 40 路控制器

### 1.2.13 SDIO

- 2 路独立 SDIO 控制器
- 1 路支持 SDIO 系统启动(SDIO0 支持启动, 可配置为 eMMC 模式)

### 1.2.14 HPET

- 32 位计数器
- 支持 1 个周期性中断
- 支持 2 个非周期性中断

### 1.2.15 RTC

- 计时精确到 0.1 秒
- 可产生 3 个计时中断

### 1.2.16 GPIO

- 139 位复用 GPIO 引脚
- 支持外部中断输入
- 与其他接口复用，使用各个接口电压域

### 1.2.17 Watchdog

- 32 比特计数器及初始化寄存器
- 低功耗模式暂停功能

### 1.2.18 温度传感器

- 温度观测，推荐使用温度范围：0~70°C(结温)
- 高低温中断

### 1.2.19 中断控制器

- 支持软件设置中断
- 支持电平与边沿触发
- 支持中断屏蔽与使能
- 支持多种中断分发模式

## 2 引脚定义

### 2.1 约定

龙芯 2P0500 的引脚进行了大量的功能复用。对于有复用关系的引脚，在介绍完其本身的功能之外会同时在下方给出其他复用功能的描述。

本章对龙芯 2P0500 引脚定义的说明使用以下约定：

- 信号名

信号名的选取以方便记忆和明确标识功能为原则。低有效信号以 N/n 结尾，高有效信号则不带 N/n。如无特别说明，以 RTC 开头的信号位于 RTC 域；其它信号位于 SOC 域。

- 类型

信号的输入输出类型由一个代码表示，见表 2-1。

表 2- 1 信号类型代码

代码	描述
A	模拟
DIFF I/O	双向差分
DIFF IN	差分输入
DIFF OUT	差分输出
I	输入
I/O	双向
O	输出
OD	开漏输出
P	电源
G	地

### 2.2 DDR3 接口

表 2- 2 DDR3 SDRAM 控制器接口信号

信号名称	类型	描述
DDR_DQ[31:0]	I/O	DDR3 SDRAM 数据总线信号
DDR_DQSP[3:0] DDR_DQSN[3:0]	DIFF I/O	DDR3 SDRAM 数据选通
DDR_DQM[3:0]	0	DDR3 SDRAM 数据屏蔽
DDR_A[15:0]	0	DDR3 SDRAM 地址总线信号
DDR_BA[2:0]	0	DDR3 SDRAM 逻辑 BANK 地址信号
DDR_WEN	0	DDR3 SDRAM 写使能信号
DDR_CASN	0	DDR3 SDRAM 列地址选择信号
DDR_RASN	0	DDR3 SDRAM 行地址选择信号
DDR_SCSNO	0	DDR3 SDRAM 片选信号
DDR_CKE0	0	DDR3 SDRAM 时钟使能信号

信号名称	类型	描述
DDR_CKPO DDR_CKNO	DIFF OUT	DDR3 SDRAM 差分时钟输出信号
DDR_ODTO	0	DDR3 SDRAM ODT 信号
DDR_RESETN	0	DDR3 SDRAM 复位控制信号

## 2.3 USB 接口

表 2- 3 USB 接口信号

信号名称	类型	描述
USB[1:0]_XI USB[1:0]_XO	I/O	USB0~1 参考时钟晶体
USB_TXRTUNE[1:0]	A	参考电阻，通过 200ohm/1%电阻连接到地
USB[1:0]_DP	I/O	USB D+
USB[1:0]_DM	I/O	USB D-
USB[1:0]_OC	I	USB0~1 过流检测，需注意该信号为高有效
OTG_DRVBUS	0	OTG 模式使用时，该信号为 DRVBUS 控制输出功能
OTG_DP	I/O	OTG D+
OTG_DM	I/O	OTG D-
OTG_TXRTUNE	A	参考电阻，通过 200ohm/1%电阻连接到地
OTG_XI OTG_XO	I/O	OTG 参考时钟晶体
OTG_ID	I	OTG ID 输入
OTG_VBUS	A	OTG VBUS 输入

## 2.4 GMAC 接口

表 2- 4 GMAC 接口信号

信号名称	类型	描述
GMAC[1:0]_TX_CLK_O	0	RGMII 发送时钟输出
GMAC[1:0]_TX_CLK_I	I	RGMII 发送时钟输入(备选，可不接)
GMAC_TX_CTL	0	RGMII 发送控制
GMAC_TXD[3:0]	0	RGMII 发送数据
GMAC[1:0]_RX_CLK_I	I	RGMII 接收时钟
GMAC_RX_CTL	I	RGMII 接收控制
GMAC_RXD[3:0]	I	RGMII 接收数据
GMAC_MDCK	0	SMA 接口时钟
GMAC_MDIO	I/O	SMA 接口数据

## 2.5 SPI 接口

表 2- 5 SPI 接口信号

信号名称	类型	描述
SPI[1:0]_CLK	0	SPI0~1 时钟输出
SPI[1:0]_CSN	0	SPI0~1 片选
SPI[1:0]_MOSI	0	SPI0~1 数据输出
SPI[1:0]_MISO	I	SPI0~1 数据输入
PRT_SPI_CLK	0	打印系统 SPI 时钟输出
PRT_SPI_CSN	0	打印系统 SPI 片选
PRT_SPI_MOSI	0	打印系统 SPI 数据输出
PRT_SPI_MISO	I	打印系统 SPI 数据输入
SCA_SPI_CLK	0	扫描系统 SPI 时钟输出
SCA_SPI_CSN	0	扫描系统 SPI 片选
SCA_SPI_MOSI	0	扫描系统 SPI 数据输出
SCA_SPI_MISO	I	扫描系统 SPI 数据输入

## 2.6 UART 接口

表 2- 6 UART 接口信号

信号名称	类型	描述
UART[1:0]_TXD	0	串口 0~1 数据输出
UART[1:0]_RXD	I	串口 0~1 数据输入
PRT_UART_TXD	0	打印系统串口数据输出
PRT_UART_RXD	I	打印系统串口数据输入
SCA_UART_TXD	0	扫描系统串口数据输出
SCA_UART_RXD	I	扫描系统串口数据输入

## 2.7 I<sup>2</sup>C 接口

表 2- 7 I<sup>2</sup>C 接口信号

信号名称	类型	描述
I2C[1:0]_SCL	0	I2C0~1 时钟
I2C[1:0]_SDA	I/O	I2C0~1 数据
PRT_I2C[1:0]_SCL	0	打印系统 I2C0~1 时钟
PRT_I2C[1:0]_SDA	I/O	打印系统 I2C0~1 数据

## 2.8 Printer 接口

表 2- 8 Printer 接口信号

信号名称	类型	描述
PRT_VIDOUT[15:0]	0	打印数据输出信号，支持 LVDS、TTL 两种电平信号模式

## 2.9 Scanner 接口

表 2- 9 Scanner 接口信号

信号名称	类型	描述
SCA_AFECLK	0	扫描 AFE 时钟输出信号
SCA_PLS	0	扫描 AFE PLS 脉冲输出信号
SCA_AFED[7:0]	I	扫描 AFE 输入数据信号
SCA_CISCLK[2:0]	0	扫描 CIS 时钟输出信号
SCA_CISPLS	0	扫描 CISPLS 脉冲输出信号
SCA_CISPWM	0	扫描 CISPWM 脉冲输出信号

## 2.10 DA 接口

表 2- 10 DA 接口信号

信号名称	类型	描述
DCHOUT[3:0]	0	4 路 DA 模拟输出信号

## 2.11 SDIO 接口

表 2- 11 SDIO 接口信号

信号名称	类型	描述
SDIO[1:0]_CLK	0	SDIO0~1 时钟输出
SDIO[1:0]_CMD	I/O	SDIO0~1 命令输入输出
SDIO[1:0]_DATA[3:0]	I/O	SDIO0~1 数据信号

## 2.12 PMIO 接口

表 2- 12 PMIO 接口信号

信号名称	类型	描述
PRT_PMIO[31:0]	I/O	打印系统 PMIO32 位信号输入输出
SCA_PMIO[15:0]	I/O	扫码系统 PMIO16 位信号输入输出
PMIO[15:0]	I/O	主系统 PMIO16 位信号输入输出

## 2.13 PWM 接口

表 2- 13 PWM 接口信号

信号名称	类型	描述
MAIN_PWM[15:0]	I/O	主系统 PWM 信号输入输出
PRT_PWM[23:0]	I/O	打印系统 PWM 信号输入输出



## 2.14 电源地接口

表 2- 14 电源地接口

信号名称	类型	描述
VDD_CORE	1.1V	CORE 域 1.1V 供电电源
VDD_1V1_USB	1.1V	USB PHY 接口电压域 1.1V 供电电源
VDD_NODE_PLL	1.2V	PLL 模拟电压 1.2V 供电电源
VDD_DDR_PLL	1.2V	PLL 模拟电压 1.2V 供电电源
VDD_SOC_PLL	1.2V	PLL 模拟电压 1.2V 供电电源
VDDIO_DDR	1.5V	DDR3 电压域 1.5V 供电电源
DDR_VREF	0.75V	DDR3 参考电源
VDD_3V3	3.3V	IO PAD 电压域 3.3V 供电电源
VDD_3V3_USB	3.3V	USB PHY 电压域 3.3V 供电电源
VDD_RTC	3V	RTC 电压域供电电源
VDD_3V3_DAC	3.3V	DA 接口 3.3V 供电电源
VDD_3V3_LVDS	3.3V	LVDS 接口 3.3V 供电电源
VDD_3V3_THSENS	3.3V	THSENS 3.3V 供电电源
VSS	0V	接地
VSS_RTC	0V	RTC 地
VSS_DAC	0V	DAC 模拟地
VSS_LVDS	0V	LVDS 模拟地
VSS_NODE_PLL	0V	NODE PLL 模拟地
VSS_DDR_PLL	0V	DDR PLL 模拟地
VSS_SOC_PLL	0V	SOC PLL 模拟地

## 2.15 测试接口

表 2- 15 测试接口

信号名称	类型	描述
DOTESTn	I	测试模式控制 (RTC 电压域) 0: 测试模式 1: 功能模式

## 2.16 JTAG 接口

表 2- 16 JTAG 接口

信号名称	类型	描述
JTAG_SEL	I	JTAG 选择 (0: 测试模式 JTAG, 1: 功能模式 JTAG)
JTAG_TCK	I	JTAG 时钟
JTAG_TDI	I	JTAG 数据输入
JTAG_TMS	I	JTAG 模式
JTAG_TRST	I	JTAG 复位
JTAG_TDO	O	JTAG 数据输出

## 2.17 系统相关信号

表 2- 17 系统相关信号

信号名称	类型	描述
SYSCLK	I	100MHz 系统参考时钟
SYSRESETN	I	系统复位输入信号
SYSPLTRSTN	O	系统平台复位输出信号
SYS_TESTCLK	I	测试时钟
RTC_XI RTC_XO	I/O	RTC 参考时钟 (32.768KHz)

## 2.18 上电配置信号

表 2- 18 上电配置信号

信号名称	类型	描述
PMOIO[0]	I	启动选择输入 0=SPI 1=SDIO
PMOIO[2:1]	I	PLL 时钟配置输入 00=低频模式 01=高频模式 10=软件模式 11=bypass 模式
PMOIO[3]	I	SDIO0 模式配置输入 0=SDIO 模式 1=EMMC 模式
PMOIO[4]	I	SDIO1 模式配置输入 0=SDIO 模式 1=EMMC 模式
PMOIO[6:5]	I	OTG 参考时钟模式输入 00=外部差分晶体输入 (XI/XO) 01/10=外部单端晶振输入 (XO) 11=内部参考时钟输入

## 2.19 外设功能引脚复用

模块层次的功能复用关系如下表所示：

表 2- 19 主系统功能引脚复用关系表

芯片主功能	第二复用	第一复用	默认功能 (除启动相关引脚)
main_uart0_rx	gmac1_col	sdio1_d[4]	MAIN_GPIO00
main_uart0_tx	gmac1_crs	sdio1_d[5]	MAIN_GPIO01
main_uart1_rx	gmac1_ptp_trig	sdio1_d[6]	MAIN_GPIO02
main_uart1_tx	gmac1_ptp_pps	sdio1_d[7]	MAIN_GPIO03
main_i2c0_scl	sdio0_d[4]	main_uart2_rx	MAIN_GPIO04
main_i2c0_sda	sdio0_d[5]	main_uart2_tx	MAIN_GPIO05

芯片主功能	第二复用	第一复用	默认功能 (除启动相关引脚)
main_i2c1_scl	sdio0_d[6]	main_uart3_rx	MAIN_GPIO06
main_i2c1_sda	sdio0_d[7]	main_uart3_tx	MAIN_GPIO07
main_spi0_clk	pmlio[0]	main_uart0_rts	MAIN_GPIO08
main_spi0_miso	pmlio[1]	main_uart0_cts	MAIN_GPIO09
main_spi0_mosi	pmlio[2]	main_uart0_dsr	MAIN_GPIO10
main_spi0_cs[0]	pmlio[3]	main_uart0_dtr	MAIN_GPIO11
main_spil_clk	pmlio[4]	main_uart0_dcd	MAIN_GPIO12
main_spil_miso	pmlio[5]	main_uart0_ri	MAIN_GPIO13
main_spil_mosi	pmlio[6]	main_uart1_rts	MAIN_GPIO14
main_spil_cs	pmlio[7]	main_uart1_cts	MAIN_GPIO15
sdio0_clk	pm0io[0]	-	MAIN_GPIO16
sdio0_cmd	pm0io[1]	main_spi0_cs[1]	MAIN_GPIO17
sdio0_d[0]	pm0io[2]	main_spi0_cs[2]	MAIN_GPIO18
sdio0_d[1]	pm0io[3]	main_spi0_cs[3]	MAIN_GPIO19
sdio0_d[2]	pm0io[4]	main_pwm[0]	MAIN_GPIO20
sdio0_d[3]	pm0io[5]	main_pwm[1]	MAIN_GPIO21
sdio1_clk	pm0io[6]	main_pwm[2]	MAIN_GPIO22
sdio1_cmd	pm0io[7]	main_pwm[3]	MAIN_GPIO23
sdio1_d[0]	pm0io[8]	main_pwm[4]	MAIN_GPIO24
sdio1_d[1]	pm0io[9]	main_pwm[5]	MAIN_GPIO25
sdio1_d[2]	pm0io[10]	main_pwm[6]	MAIN_GPIO26
sdio1_d[3]	pm0io[11]	main_pwm[7]	MAIN_GPIO27
pm2io[0]	pm0io[12]	main_pwm[0]	MAIN_GPIO28
pm2io[1]	pm0io[13]	main_pwm[1]	MAIN_GPIO29
pm2io[2]	pm0io[14]	main_pwm[2]	MAIN_GPIO30
pm2io[3]	pm0io[15]	main_pwm[3]	MAIN_GPIO31
pm2io[4]	gmac1_rx_ctl	main_pwm[4]	MAIN_GPIO32
pm2io[5]	gmac1_rx[0]	main_pwm[5]	MAIN_GPIO33
pm2io[6]	gmac1_rx[1]	main_pwm[6]	MAIN_GPIO34
pm2io[7]	gmac1_rx[2]	main_pwm[7]	MAIN_GPIO35
pm2io[8]	gmac1_rx[3]	main_pwm[8]	MAIN_GPIO36
pm2io[9]	gmac1_tx_ctl	main_pwm[9]	MAIN_GPIO37
pm2io[10]	gmac1_tx[0]	main_pwm[10]	MAIN_GPIO38
pm2io[11]	gmac1_tx[1]	main_pwm[11]	MAIN_GPIO39
pm2io[12]	gmac1_tx[2]	main_pwm[12]	MAIN_GPIO40
pm2io[13]	gmac1_tx[3]	main_pwm[13]	MAIN_GPIO41
pm2io[14]	gmac1_mdck	main_pwm[14]	MAIN_GPIO42
pm2io[15]	gmac1_mdio	main_pwm[15]	MAIN_GPIO43

表 2- 20 打印系统功能引脚复用关系表

芯片主功能	第二复用	第一复用	默认功能
prt_uart0_rx	pm0io[0]	prt_pwm[0]	PRT_GPIO00
prt_uart0_tx	pm0io[1]	prt_pwm[1]	PRT_GPIO01
prt_i2c0_scl	pm0io[2]	prt_pwm[2]	PRT_GPIO02
prt_i2c0_sda	pm0io[3]	prt_pwm[3]	PRT_GPIO03
prt_i2c1_scl	prt_uart1_rx	prt_pwm[4]	PRT_GPIO04
prt_i2c1_sda	prt_uart1_tx	prt_pwm[5]	PRT_GPIO05

芯片主功能	第二复用	第一复用	默认功能
prt_spi_clk	gmac0_col	prt_pwm[6]	PRT_GPIO06
prt_spi_miso	gmac0_crs	prt_pwm[7]	PRT_GPIO07
prt_spi_mosi	gmac0_ptp_trig	prt_pwm[8]	PRT_GPIO08
prt_spi_cs	gmac0_ptp_pps	prt_pwm[9]	PRT_GPIO09
pm0io[0]	prt_vid_out[16]	prt_pwm[10]	PRT_GPIO10
pm0io[1]	prt_vid_out[17]	prt_pwm[11]	PRT_GPIO11
pm0io[2]	prt_vid_out[18]	prt_pwm[12]	PRT_GPIO12
pm0io[3]	prt_vid_out[19]	prt_pwm[13]	PRT_GPIO13
pm0io[4]	prt_vid_out[20]	prt_pwm[14]	PRT_GPIO14
pm0io[5]	prt_vid_out[21]	prt_pwm[15]	PRT_GPIO15
pm0io[6]	prt_vid_out[22]	prt_pwm[16]	PRT_GPIO16
pm0io[7]	prt_vid_out[23]	prt_pwm[17]	PRT_GPIO17
pm0io[8]	prt_pwm[0]	vid_clk[0]	PRT_GPIO18
pm0io[9]	prt_pwm[1]	vid_clk[1]	PRT_GPIO19
pm0io[10]	prt_pwm[2]	vid_clk[2]	PRT_GPIO20
pm0io[11]	prt_pwm[3]	vid_clk[3]	PRT_GPIO21
pm0io[12]	prt_pwm[4]	prt_vid_en[0]	PRT_GPIO22
pm0io[13]	prt_pwm[5]	prt_vid_en[1]	PRT_GPIO23
pm0io[14]	prt_pwm[6]	prt_vid_en[2]	PRT_GPIO24
pm0io[15]	prt_pwm[7]	prt_vid_en[3]	PRT_GPIO25
pm0io[16]	prt_pwm[8]	prt_ovlin[0]	PRT_GPIO26
pm0io[17]	prt_pwm[9]	prt_ovlin[1]	PRT_GPIO27
pm0io[18]	prt_pwm[10]	prt_ovlin[2]	PRT_GPIO28
pm0io[19]	prt_pwm[11]	prt_ovlin[3]	PRT_GPIO29
pm0io[20]	prt_pwm[12]	prt_ovlin[4]	PRT_GPIO30
pm0io[21]	prt_pwm[13]	prt_ovlin[5]	PRT_GPIO31
pm0io[22]	prt_pwm[14]	prt_ovlin[6]	PRT_GPIO32
pm0io[23]	prt_pwm[15]	prt_ovlin[7]	PRT_GPIO33
pm0io[24]	prt_vid_out[24]	prt_ovlin[8]	PRT_GPIO34
pm0io[25]	prt_vid_out[25]	prt_ovlin[9]	PRT_GPIO35
pm0io[26]	prt_vid_out[26]	prt_ovlin[10]	PRT_GPIO36
pm0io[27]	prt_vid_out[27]	prt_ovlin[11]	PRT_GPIO37
pm0io[28]	prt_vid_out[28]	prt_ovlin[12]	PRT_GPIO38
pm0io[29]	prt_vid_out[29]	prt_ovlin[13]	PRT_GPIO39
pm0io[30]	prt_vid_out[30]	prt_ovlin[14]	PRT_GPIO40
pm0io[31]	prt_vid_out[31]	prt_ovlin[15]	PRT_GPIO41
prt_vid_out[0]	pm0io[8]	pm0io[16]	PRT_GPIO42
prt_vid_out[1]	pm0io[9]	pm0io[17]	PRT_GPIO43
prt_vid_out[2]	pm0io[10]	pm0io[18]	PRT_GPIO44
prt_vid_out[3]	pm0io[11]	pm0io[19]	PRT_GPIO45
prt_vid_out[4]	pm0io[12]	pm0io[20]	PRT_GPIO46
prt_vid_out[5]	pm0io[13]	pm0io[21]	PRT_GPIO47
prt_vid_out[6]	pm0io[14]	pm0io[22]	PRT_GPIO48
prt_vid_out[7]	pm0io[15]	pm0io[23]	PRT_GPIO49
prt_vid_out[8]	prt_pwm[16]	pm0io[24]	PRT_GPIO50
prt_vid_out[9]	prt_pwm[17]	pm0io[25]	PRT_GPIO51
prt_vid_out[10]	prt_pwm[18]	pm0io[26]	PRT_GPIO52
prt_vid_out[11]	prt_pwm[19]	pm0io[27]	PRT_GPIO53

芯片主功能	第二复用	第一复用	默认功能
prt_vid_out[12]	prt_pwm[20]	pm0io[28]	PRT_GPIO54
prt_vid_out[13]	prt_pwm[21]	pm0io[29]	PRT_GPIO55
prt_vid_out[14]	prt_pwm[22]	pm0io[30]	PRT_GPIO56
prt_vid_out[15]	prt_pwm[23]	pm0io[31]	PRT_GPIO57

表 2- 21 扫描系统功能引脚复用关系表

芯片主功能	第二复用	第一复用	默认功能
sca_uart0_rx	pmlio[16]	-	SCA_GPIO00
sca_uart0_tx	pmlio[17]	-	SCA_GPIO01
sca_spi0_clk	pmlio[18]	-	SCA_GPIO02
sca_spi0_cs	pmlio[19]	sca_uart1_rx	SCA_GPIO03
sca_spi0_miso	pmlio[20]	sca_uart1_tx	SCA_GPIO04
sca_spi0_mosi	pm0io[31]	pmlio[31]	SCA_GPIO05
pmlio[0]	pm0io[0]	sca_afesd[8]	SCA_GPIO06
pmlio[1]	pm0io[1]	sca_afesd[9]	SCA_GPIO07
pmlio[2]	pm0io[2]	sca_afesd[10]	SCA_GPIO08
pmlio[3]	pm0io[3]	sca_afesd[11]	SCA_GPIO09
pmlio[4]	pm0io[4]	sca_afesd[12]	SCA_GPIO10
pmlio[5]	pm0io[5]	sca_afesd[13]	SCA_GPIO11
pmlio[6]	pm0io[6]	sca_afesd[14]	SCA_GPIO12
pmlio[7]	pm0io[7]	sca_afesd[15]	SCA_GPIO13
pmlio[8]	pm0io[8]	sca_afepls[1]	SCA_GPIO14
pmlio[9]	pm0io[9]	sca_afepls[2]	SCA_GPIO15
pmlio[10]	pm0io[10]	sca_cispls[1]	SCA_GPIO16
pmlio[11]	pm0io[11]	sca_cisclk[3]	SCA_GPIO17
pmlio[12]	pm0io[12]	sca_cisclk[4]	SCA_GPIO18
pmlio[13]	pm0io[13]	sca_cisclk[5]	SCA_GPIO19
pmlio[14]	pm0io[14]	sca_cisclk[6]	SCA_GPIO20
pmlio[15]	pm0io[15]	-	SCA_GPIO21
sca_afeclk	pm0io[16]	pmlio[16]	SCA_GPIO22
sca_afepls[0]	pm0io[17]	pmlio[17]	SCA_GPIO23
sca_afesd[0]	pm0io[18]	pmlio[18]	SCA_GPIO24
sca_afesd[1]	pm0io[19]	pmlio[19]	SCA_GPIO25
sca_afesd[2]	pm0io[20]	pmlio[20]	SCA_GPIO26
sca_afesd[3]	pm0io[21]	pmlio[21]	SCA_GPIO27
sca_afesd[4]	pm0io[22]	pmlio[22]	SCA_GPIO28
sca_afesd[5]	pm0io[23]	pmlio[23]	SCA_GPIO29
sca_afesd[6]	pm0io[24]	pmlio[24]	SCA_GPIO30
sca_afesd[7]	pm0io[25]	pmlio[25]	SCA_GPIO31
sca_cisclk[0]	pm0io[26]	pmlio[26]	SCA_GPIO32
sca_cisclk[1]	pm0io[27]	pmlio[27]	SCA_GPIO33
sca_cisclk[2]	pm0io[28]	pmlio[28]	SCA_GPIO34
sca_cispls[0]	pm0io[29]	pmlio[29]	SCA_GPIO35
sca_cispwm	pm0io[30]	pmlio[30]	SCA_GPIO36

## 3 时钟结构

龙芯 2P0500 的时钟结构如图3- 1 所示，片内主要由一路外部时钟输入，100MHz 固定频率时钟作为系统参考时钟输入，供片内 PLL 使用。芯片内部共有 3 个独立的 PLL，其中每个 PLL 最多可以提供 3 组频率上相互依赖的时钟输出。这 3 个 PLL 的用途分别为：

一个 SYS PLL 用于产生 SYS 系统时钟，该时钟为芯片的主要时钟，分为三组频率时钟，一路供 CPU 核、二级 Cache、一二级交叉开关以及 IO 子网络使用，一路供打印、扫描核使用，一路供 USB、GMAC、APB 等设备控制器使用；

一个 DDR PLL 同时产生 DDR、NETWORK 以及 IMAGE 的时钟；

一个 VID PLL 同时产生 Printer、Scanner 控制器的内部时钟。

除了内部的 PLL 之外，对于 USB 采用 PHY 自身产生时钟的模块，也使用外部输入的参考时钟进行参考时钟通路设计。

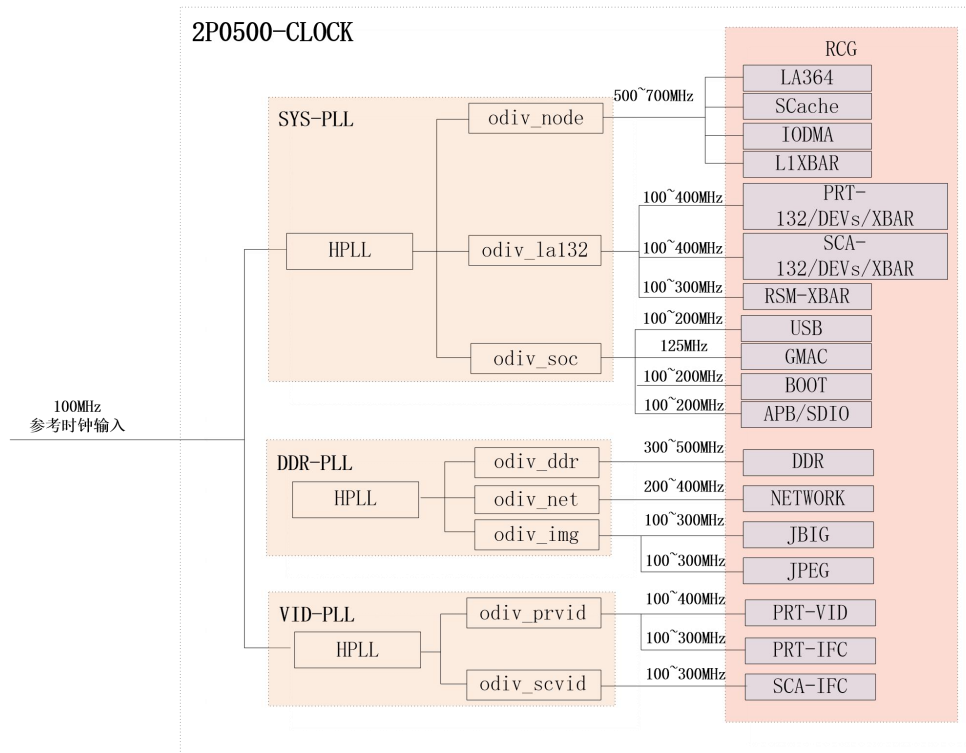


图3- 1 时钟结构

### 3.1 SYS PLL

sys clock 的产生结构图如 3-2 所示：

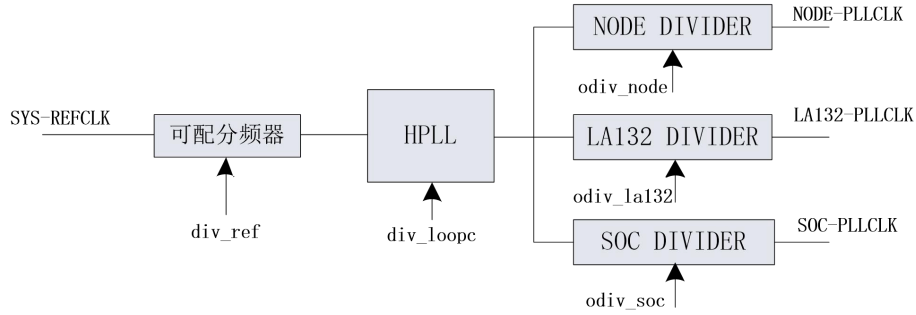


图3- 2 SYS PLL 结构图

SYS PLL 会输出三个时钟，分别为：

- ✓ node\_pllclk 用于 NODE 控制器，推荐频率范围 500-700MHz；
- ✓ la132\_pllclk 用于 LA132 核，推荐频率范围 100-400MHz；
- ✓ soc\_pllclk 用于 USB、GMAC 等模块，推荐频率范围 100-200MHz。

输出时钟频率的计算方式如下：

$$\text{node\_pllclk} = \text{refclk} / \text{div\_ref} * \text{div\_loopc} / \text{odiv\_node};$$

$$\text{la132\_pllclk} = \text{refclk} / \text{div\_ref} * \text{div\_loopc} / \text{odiv\_la132};$$

$$\text{soc\_pllclk} = \text{refclk} / \text{div\_ref} * \text{div\_loopc} / \text{odiv\_soc}.$$

node\_pllclk 的工作频率在 500M~700MHz 左右，其 PLL 的分频系数以及倍频系数可以任意配置，但是需要保证可配分频器的输出  $\text{refclk} / \text{div\_ref}$  在 20~40MHz 范围内，PLL 倍频值  $\text{refclk} / \text{div\_ref} * \text{div\_loopc}$  需要在 1.2GHz~3.2GHz。该限制对其他 2 个内部 PLL 也适用，所以下文不再赘述。

以上三个时钟共用一个 PLL，只是通过设置各自的 divout 值来实现不同的频率输出。所以在调整其中一个模块时钟时，如果对公用 PLL 的倍频系数进行了调整，那么需要注意对其他时钟的影响。

输出的时钟还可以经由 freq\_scale 模块进行细粒度分频控制。具体分频方法请参考 5.4.21 节。

## 3.2 DDR PLL

DDR PLL 结构与 SYS PLL 结构基本相同，输出三个时钟，分别为：

- ✓ ddr\_pllclk 用于内存控制器，推荐频率范围 300-500MHz；
- ✓ network\_pllclk 用于 NETWORK 模块，推荐频率范围 200-400MHz；
- ✓ img\_pllclk 用于 IMAGE 模块，推荐频率范围 100-300MHz。

以上三个时钟共用一个 PLL，通过设置各自的 divout 值来实现不同的频率输出。同样在调整其中一个模块时钟时，如果对公用 PLL 的倍频系数进行调整，需要注意对其他时钟

的影响。

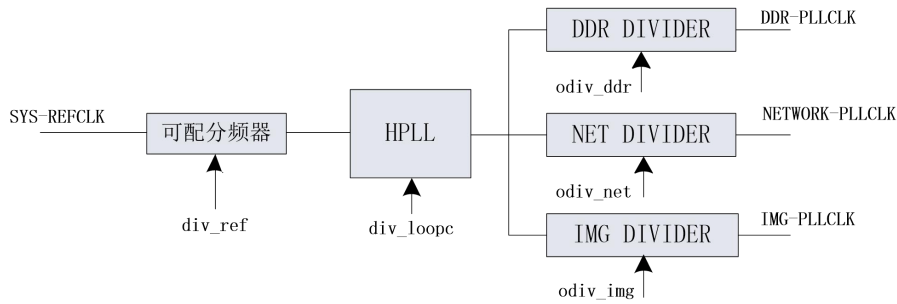


图3- 3 DDR PLL 时钟结构

### 3.3 VID PLL

VID PLL 结构与其他 PLL 结构类似，输出两个时钟，分别为：

- ✓ prvid\_pllclk 用于打印接口，推荐频率范围 100-400MHz；
- ✓ scvid\_pllclk 用于扫描接口，推荐频率范围 100-300MHz。

两个时钟共用一个 PLL，可通过设置各自的 divout 值来实现不同的频率输出。同样在调整其中一个模块时钟时，如果对公用 PLL 的倍频系数进行了调整，需要注意对其他时钟的影响。

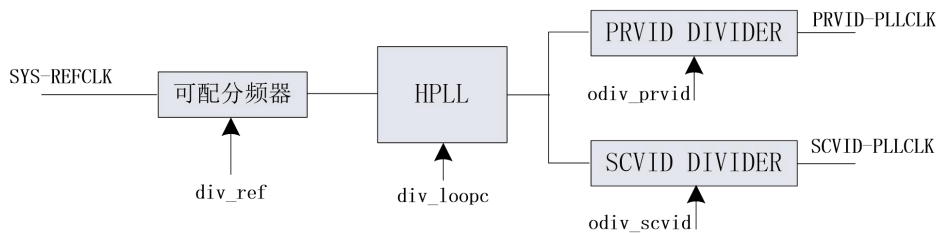


图3- 4 SOC PLL 时钟结构

### 3.4 内部 PLL 配置方法

以上 3 个内部 PLL 提供硬件配置和软件配置两种配置方法。这两种配置方法通过 SYS\_CLKSEL[1:0]的设置来区分。

#### 3.4.1 硬件配置

具体如下表所示。



表 3- 1 PLL 硬件配置

SYS_CLKSEL	00(硬件低频)	01(硬件高频)	10	11
NODE	500M	725M	软件配置	硬件 bypass 所有 PLL, 所有时钟频率与参考时钟相同 (100MHz)
LA132	200M	375M		
DDR	480M	600M		
NETWORK	320M	400M		
GMAC	不可硬件配置	不可硬件配置		
USB	125M	125M		
APB	125M	125M		
IMP	100M	200M		
PRVID	125M	250M		
SCVID	125M	250M		

### 3.4.2 软件配置

当 SYS\_CLKSEL 设置为 2'b10 时表示 PLL 频率通过软件配置。这种配置下，默认对应的时钟频率为外部参考时钟频率，即所有 PLL 输出都是 SYS\_REFCLK，需要在处理器启动过程中对时钟进行软件配置。各个时钟设置的过程应该按照以下方式：

1. 将对应的 PLL 的 PD 信号设置为 1；
2. 设置寄存器除了 sel\_pll\_\* 及 soft\_set\_pll 之外的其它寄存器，即这两个寄存器在设置的过程中写为 0；
3. 将对应的 PLL 的 PD 信号设置为 0；
4. 其他寄存器值不变，将 soft\_set\_pll 设置为 1；
5. 等待寄存器中的锁定信号 locked\_\* 为 1；
6. 设置 sel\_pll\_\* 为 1，此时对应的时钟频率将切换为软件设置的频率。

另外，芯片内部配有设备时钟分频配置参数 freqscale，可供软件对部分设备时钟进行再次分频选择。

具体的配置寄存器说明请参考第五章。

### 3.5 USB 参考时钟

USB PHY 为 3 个独立的单端口 PHY，参考时钟输入提供以下 2 种方式供选择：

- ✓ 使用 1 个 12MHz 晶振输入，其它使用 0 号 PHY 的时钟输出作为参考源
- ✓ 不使用单独的参考时钟输入，都使用 50MHz 的内部参考时钟(100MHz 的 SYSCLK 经二分频后)

### 3.6 时钟信号说明

表 3- 2 2P0500 时钟信号说明

统一介绍了龙芯 2P0500 的所有时钟引脚。

表 3- 2 2P0500 时钟信号说明

信号名称	推荐工作频率 (MHz)	类型	描述	
SYSCLK	100	I/O	外接系统参考时钟	
RTC_XI RTC_XO	32.768K	I/O	RTC 参考时钟	
JTAG_TCK	33	I	JTAG 时钟	
USB_XTAL	25	DIFF IN	USB 参考时钟输入 如果使用内部参考时钟，则可悬空。	
DDR_CKp[1:0] DDR_CKn[1:0]	400	DIFF OUT	DDR3 SDRAM 差分时钟输出	
内部时钟	NODE_CLOCK	500~700	-	主系统 NODE 模块时钟，供 LA364、SCACHE、IODMA、L1-XBAR 模块使用
	STABLE_CLOCK	100	-	LA364 核内 STABLE 计数时钟
	PRI32_CLOCK	100~400	-	打印系统 132 核时钟，供 132 处理器核内部使用
	PRXBAR_CLOCK	100~300	-	打印系统互联时钟
	PRDEV_CLOCK	100~200	-	打印系统内部设备、APB 时钟
	SC132_CLOCK	100~400	-	扫描系统 132 核时钟，供 132 处理器核内部使用
	SCXBAR_CLOCK	100~300	-	扫描系统互联时钟
	SCDEV_CLOCK	100~200	-	扫描系统内部设备、APB 时钟
	RSMXBAR_CLOCK	100~300	-	RESUME 单元内部互联时钟
	USB_CLOCK	100~200	-	USB 模块时钟
	GMAC_CLOCK	125	-	GMAC 模块时钟
	SB_CLOCK	100~200	-	SB 设备时钟 (BOOT/CONFBUS 等)
	APB_CLOCK	100~200	-	主系统 APB 设备时钟
	AES_CLOCK	100~150	-	主系统 AES 设备时钟
	SDIO_CLOCK	100~200	-	SDIO 模块时钟
	DDR_CLOCK	300~500	-	DDR 控制器时钟，供 DDR3 控制器使用
	NETWORK_CLOCK	200~400	-	供 NETWORK 互联结构使用
	JPEG_CLOCK	100~300	-	JPEG 功能单元时钟
	JBIG_CLOCK	100~300	-	JBIG 功能单元时钟
	PRVID_CLOCK	100~400	-	供 PRINTer 打印数据处理单元使用
PRIFC_CLOCK	100~300	-	供 PRINTer 打印模块 AXI 总线使用	
SCIFC_CLOCK	100~300	-	供 SCAN 扫描模块数据处理、AXI 总线使用	
THSENS_CLOCK	1	-	供 THSENS 设备使用	

# 4 电源管理

本章对电源管理的主体结构进行简要介绍。

## 4.1 电源管理模块介绍

- 龙芯 2P0500 电源管理模块(打印系统)提供系统功耗管理实现机制。
- 系统待机、休眠与唤醒。支持多种唤醒方式（OTG、网络、按键电源开关等）
- 支持 Dynamic Power Management（DPM），动态性能功耗控制，支持动态关闭 NODE（CORE+SCACHE）、USB2.0、GMAC 控制器等电源。
- 支持 Dynamic Frequency Scaling（DFS），处理器核 DFS 控制，由片内打印 LA132 处理器核独立控制。
- 系统时钟控制，模块时钟门控，多种方式调节频率。
- 提供温度管理控制功能。支持 3 级报警机制。

## 4.2 电源级别

表 4.1 显示了系统支持的电源管理状态及其相关说明。

表 4.1 电源管理状态说明

状态	描述
工作状态	全部工作，该模式下系统全部工作。
待机状态	主核及相关设备停止工作，可快速系统唤醒
休眠状态	除打印系统相关控制低速工作外，其他模块均掉电休眠，可通过 OTG、网络、按键唤醒

## 5 芯片配置与控制

### 5.1 芯片初始化信号

龙芯 2P0500 初始化信号：复用芯片功能引脚，通过在系统复位期间采样外部上下拉的值得到配置信息。部分配置信息编码成 bootcfg，供软件判定上电状态。

表 5- 1 初始化配置信号

信号名称	类型	描述
PM0IO[0]	I	启动选择输入 0=SPI 1=SDIO
PM0IO[2:1]	I	PLL 时钟配置输入 00=低频模式 01=高频模式 10=软件模式 11=bypass 模式
PM0IO[3]	I	SDIO0 模式配置输入 0=SDIO 模式 1=EMMC 模式
PM0IO[4]	I	SDIO1 模式配置输入 0=SDIO 模式 1=EMMC 模式
PM0IO[6:5]	I	OTG 参考时钟模式输入 00=外部差分晶体输入 (XI/XO) 01/10=外部单端晶振输入 (XO) 11=内部参考时钟输入

### 5.2 地址空间分配

龙芯 2P0500 的地址空间根据各个系统应用不同，分为主系统、打印系统、扫描系统和 DMA 四个视角地址空间：所有主系统处理器核可访问的设备编址在主系统地址空间上，谓之主系统视角；打印系统、扫描系统类似，可通过 DMA 直接访问系统内存的主设备所见到的空间为 DMA 视角。此外，各个系统之间存在可以共享访问设备地址空间，不同系统访问同一设备，其设备地址均相同，可通过不同系统之间共享设备操作。表 5-2~表 5-4 分别给出了这四个视角的具体定义。表格中未包含的地址空间均为系统保留，软件错误地访问保留空间将导致不可预知的后果。

表 5- 2 地址空间分配之主系统视角

地址空间 (unmapped)	大小	功能	拓扑
0x0000_0000 - 0x0fff_ffff	256MB	L-DDR	DDR
0x1000_0000 - 0x13ff_ffff	64MB	SPIO MEM	MEM

0x1400_0000 - 0x1401_ffff	128KB	SYS-CONFBUS	CONFBUS (主/打印/扫描系统可共享)
0x1402_0000 - 0x140f_ffff	1MB-128KB	DEVs (AXI-private)	DEVs (USB/GMAC1/IMGP)
0x1410_0000 - 0x141f_ffff	1MB	DEVs (AXI-resume)	LA364-DEVs (主/打印系统可共享)
0x1420_0000 - 0x142f_ffff	1MB	DEVs (APB)	LA364-DEVs (主/打印/扫描系统可共享)
		Reserved	
0x1500_0000 - 0x150f_ffff	1MB	PR-SRAM	PR-SRAM (主/打印/扫描系统可共享)
0x1510_0000 - 0x151f_ffff	1MB	PR-DEVs	PR-DEVs (主/打印/扫描系统可共享)
0x1520_0000 - 0x152f_ffff	1MB	SC-DEVs	SC-DEVs (主/打印/扫描系统可共享)
		Reserved	
0x1c00_0000 - 0x1c0f_ffff	1MB	BOOT	Boot (主/打印/扫描系统可共享)
		Reserved	
0x8000_0000 - 0xffff_ffff	2GB	H-DDR	DDR
地址空间 (AXI-private) [19:0] Base 0x1400_0000	大小 1MB	功能	拓扑
0x0_0000	128KB	CONF/INT	CLK/INT/IOMUX...
0x2_0000	64KB	GMAC1	
0x3_0000	64KB	USB1~2	
0x4_0000	64KB	JBIG*4	Jbig*4, 每路平均分配 16KB 空间
0x5_0000	64KB	JPEG	
地址空间 (AXI-Resume) [19:0] Base 0x1410_0000	大小 1MB	功能	拓扑
0x0_0000	256KB	OTG	(Resume 主/打印/系统可共享)
0x4_0000	64KB	GMACO	(Resume 主/打印/系统可共享)
地址空间 (APB) [19:0] Base 0x1420_0000	大小 1MB	功能	拓扑
0x0_0000	4KB	UARTx4	(主/打印/扫描系统可共享), 每路平均分配 1KB 空间
0x0_1000	4KB	I2Cx2	(主/打印/扫描系统可共享), 每路平均分配 2KB 空间
0x0_2000	4KB	GPIOx64	(主/打印/扫描系统可共享), 分为两种配置方式
0x0_3000	4KB	SPIx1	(主/打印/扫描系统可共享)
0x0_4000	4KB	HPETx2	(主/打印/扫描系统可共享), 每路平均分配 2KB 空间
0x0_5000	4KB	WDTx1	(主/打印/扫描系统可共享)
0x0_6000	4KB	PM2IOx1	(主/打印/扫描系统可共享)
0x0_7000	4KB	PWMx16	(主/打印/扫描系统可共享), 有效空间 256B, 每路 16B 空间
0x0_8000	512B	ENC-DMA0	(主/打印/扫描系统可共享)
0x0_8200	512B	ENC-DMA1	(主/打印/扫描系统可共享)
0x0_8400	1KB-16B	SM2x1	(主/打印/扫描系统可共享)
0x0_87f0	16B	RNGx1	(主/打印/扫描系统可共享)
0x0_8800	512B	AESx1	(主/打印/扫描系统可共享)
0x0_8a00	512B	DESx1	(主/打印/扫描系统可共享)
0x0_8c00	512B	SM3x1	(主/打印/扫描系统可共享)
0x0_8e00	512B	SM4x1	(主/打印/扫描系统可共享)
0x1_0000	32KB	SDIO0+DMA	(主/打印/扫描系统可共享)
0x1_8000	32KB	SDIO1+DMA	(主/打印/扫描系统可共享)

表 5- 3 地址空间分配之打印系统视角

地址空间 (unmapped)	大小	功能	拓扑
		Reserved	
0x1400_0000 - 0x1400_ffff	64KB	CONFBUS (AXI-CONF)	LA364-CONFBUS (主/打印/扫描系统可共享)
0x1410_0000 - 0x141f_ffff	1MB	DEVs (AXI-resume)	LA364-Resume-DEVs (主/打印系统可共享)
0x1420_0000 - 0x142f_ffff	1MB	DEVs (APB)	LA364-APB-DEVs (主/打印/扫描系统可共享)
		Reserved	
0x1500_0000 - 0x150f_ffff	1MB	SRAM (128K)	PR-BOOT/DATA (主/打印/扫描系统可共享)
0x1510_0000 - 0x151f_ffff	1MB	DEVs (APB)	PR-DEVs (主/打印/扫描系统可共享)
0x1520_0000 - 0x152f_ffff	1MB	SC-DEVs	SC-DEVs (主/打印/扫描系统可共享)
		Reserved	
0x1c00_0000 - 0x1c0f_ffff	1MB	BOOT	SYS-BOOT (主/打印系统可共享)
		Reserved	
0x8000_0000 - 0xffff_ffff	2GB	DDR	DDR/CCIO
地址空间 (AXI) [19:0] Base 0x1500_0000	大小 1MB	功能	拓扑
0x0_0000 - 0x1_ffff	128KB	SRAM (实际 128KB)	PR-BOOT/DATA (主/打印/扫描系统共享 PR-SRAM 存储)
地址空间 (APB) [19:0] Base 0x1510_0000	大小 1MB	功能	拓扑
0x0_0000	4KB	UARTx2	(主/打印/扫描系统可共享), 每路平均分配 2KB 空间
0x0_1000	4KB	I2Cx2	(主/打印/扫描系统可共享), 每路平均分配 2KB 空间
0x0_2000	4KB	GPIOx96	(主/打印/扫描系统可共享)
0x0_3000	4KB	CONF	MAILBOX/CLK/INT/CHIP/MUX... (主/打印/扫描系统可共享)
0x0_4000	4KB	DPM	(主/打印/扫描系统可共享)
0x0_5000	4KB	DAC	(主/打印/扫描系统可共享)
0x0_6000	4KB	Reserved	-
0x0_7000	4KB	PWMx24	(主/打印/扫描系统可共享) 有效空间 24*16B, 每路 16B 空间
0x0_8000	4KB	PMOIO	(主/打印/扫描系统可共享)
0x0_9000	4KB	SPIx1	(主/打印/扫描系统可共享)
0x0_a000	4KB	RTC	(主/打印/扫描系统可共享)
0x0_b000	4KB	HPET	(主/打印/扫描系统可共享)
0x0_c000	16KB	PR-VIDEO	(主/打印/扫描系统可共享)


表 5- 4 地址空间分配之扫描系统视角


地址空间 (unmapped)	大小	功能	拓扑
		Reserved	
0x1400_0000 - 0x1400_ffff	64KB	CONFBUS (AXI-CONF)	LA364-CONFBUS (主/打印/扫描系统可共享)
0x1410_0000 - 0x141f_ffff	1MB	DEVs (AXI-resume)	LA364-Resume-DEVs (主/打印系统可共享)
0x1420_0000 - 0x142f_ffff	1MB	DEVs (APB)	LA364-APB-DEVs (主/打印/扫描系统可共享)
		Reserved	
0x1500_0000 - 0x150f_ffff	1MB	SRAM (128K)	PR-BOOT/DATA (主/打印/扫描系统可共享)


0x1510_0000 - 0x151f_ffff	1MB	DEVs (APB)	PR-DEVs (主/打印/扫描系统可共享)
0x1520_0000 - 0x152f_ffff	1MB	SC-DEVs	SC-DEVs (主/打印/扫描系统可共享)
		Reserved	
0x1c00_0000 - 0x1c0f_ffff	1MB	BOOT	SYS-BOOT (主/扫描系统可共享)
		Reserved	
0x8000_0000 - 0xffff_ffff	2GB	DDR	DDR/CCIO
地址空间 (APB) [19:0] Base 0x1520_0000	大小 1MB	功能	拓扑
0x0_0000	4KB	UARTx2	(主/打印/扫描系统可共享) 每路平均分配 2KB 空间
0x0_1000	4KB	GPIOx64	(主/打印/扫描系统可共享)
0x0_2000	4KB	CONF	MAILBOX/INT... (设备可共享)
0x0_3000	4KB	SPIx1	(主/打印/扫描系统可共享)
0x0_4000	4KB	PMIIO	(主/打印/扫描系统可共享)
0x0_5000	4KB	HPETx1	(主/打印/扫描系统可共享)
0x0_6000	4KB	SCAN_VIDEO	(主/打印/扫描系统可共享)


**注：共享资源说明：**

(1) 主系统-设备可共享：


 Confbus(AXI)：主系统、打印系统、扫描系统；

 BOOT(AXI)：主系统、打印系统、扫描系统；


 DEVs(APB)：主系统、打印系统、扫描系统；

 OTG/GMAC(Resume)：主系统、打印系统；

(2) 打印系统设备共享：

 PRT 设备(PR-SRAM/DEVs)可共享：主系统、打印系统、扫描系统；

(3) 扫描系统设备共享：

 SCAN 设备(SC-DEVs)可共享：主系统、打印系统、扫描系统。

龙芯 2P0500 内部可发起 DMA 的主设备包括 GMAC、USB、IMG、SDIO 等，主设备的请求可以访问路由到表 5- 5 中的所有空间。

表 5- 5 地址空间分配之 DMA 视角

地址空间 (mapped)	大小	功能	说明
0x0000_0000 - 0x7fff_ffff	2G	DDR	同时设有 CACHE 访问配置位，可配置选择访问 DDR 或 CCIO，默认按照表中地址进行分配。
0x8000_0000 - 0xffff_ffff	2G	CCIO	
0x0000_0000 - 0x0001_ffff	128KB	SRAM	GMACO/OTG 可选数据存储

## 5.3 时钟与复位控制

### 5.3.1 时钟配置概要

龙芯 2P0500 的片上时钟有硬件、软件两种配置模式。硬件配置模式下时钟生成完全不需要软件参与，但频率选择非常有限。软件配置模式下所有时钟相关的参数都可以改变，非常灵活，但在操作时需谨慎。本小节主要描述软件配置的流程。

在系统复位结束后，所有 PLL 输出均被旁路为参考时钟，分频器设置初始化为默认分频数，整个系统以最低速度运行。这时引导程序应当对片上的每个 PLL 进行以下操作：

1. 将对应的 PLL 的 PD 信号设置为 1；
2. 设置寄存器除了 sel\_pll\_\*及 soft\_set\_pll 之外的其它寄存器，即这两个寄存器在设置的过程中写为 0；
3. 将对应的 PLL 的 PD 信号设置为 0；
4. 其他寄存器值不变，将 soft\_set\_pll 设置为 1；
5. 等待寄存器中的锁定信号 locked\_\*为 1；
6. 设置 sel\_pll\_\*为 1，此时对应的时钟频率将切换为软件设置的频率。

如果后期有需要修改 PLL 参数，则要先切换时钟为参考时钟，然后按上述步骤再配一遍。

### 5.3.2 展频 PLL 配置

展频 PLL 概念性结构如图5- 1 所示，输入时钟 Fin 经过输入分频器得到 Fref 送到倍频器，倍频器送出 Fvco，然后在输出前除以一个分频系数。展频 PLL 还带有一个展频控制器，能够用三角波对输出时钟的频率进行调制。

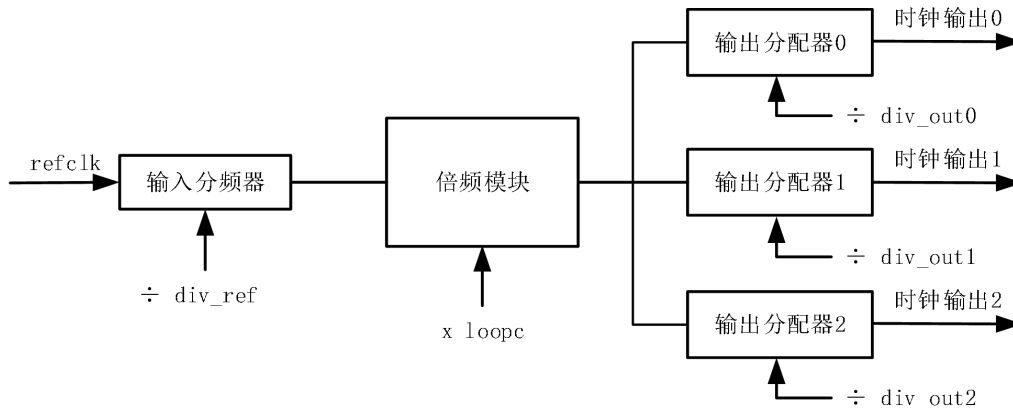


图5- 1 展频 PLL 概念性结构

输出时钟频率的计算方式如下： $clock\_out = refclk / div\_ref * loopc / divoutN$ ；

其中，2P0500 的 refclk 固定为 100MHz，此外需要保证输入分频器的输出 ( $refclk / div\_ref$ ) 在 20 ~ 40MHz 的范围内，倍频模块倍频后的频率 ( $refclk / div\_ref * loopc$ ) 在 1.2GHz ~ 3.2GHz 的范围内。

PLL 相关的配置信号及说明见表 5-6。

表 5- 6 PLL 相关配置信号说明表

信号	位数	方向	说明
pll_div_out0	7	R/W	PLL 输出时钟 0 分频数
pll_div_out1	7	R/W	PLL 输出时钟 1 分频数
pll_div_out2	7	R/W	PLL 输出时钟 2 分频数



信号	位数	方向	说明
pll_loopc	9	R/W	PLL 倍频乘数
pll_div_ref	7	R/W	PLL 输入分频数
pll_locked	1	RO	PLL 锁定
sel_pll_out0	1	R/W	选择 PLL 输出时钟 0
sel_pll_out1	1	R/W	选择 PLL 输出时钟 1
sel_pll_out2	1	R/W	选择 PLL 输出时钟 2
set_pll_param	1	R/W	设置 PLL 配置参数
pll_bypass	1	R/W	PLL 内部 bypass
pll_pd	1	R/W	PLL powerdown

### 5.3.3 复位控制

龙芯 2P0500 内部有多种复位机制，用户可根据需要进行选择：

- 模块级复位：只针对一个模块发起，如 USB、GMAC 等。一般只在配置时钟完成，还未开始工作前进行。在工作后复位有使系统死锁的危险。
- 系统热复位：除芯片配置寄存器和时钟配置寄存器保持不变外，全系统复位。
- 系统冷复位：由看门狗或者复位模块发起，详见相关章节。

## 5.4 芯片配置寄存器

龙芯 2P0500 有大量的配置寄存器，根据各个系统配置应用不同，分为主系统、打印系统、扫描系统配置空间,其余多数分布于各个功能模块中，本节介绍各个系统芯片级的配置寄存器。

表 5- 7 主系统芯片配置寄存器列表

地址	名称	描述
0x14000100	MAIN_CHIP_CTRL0	主系统通用配置寄存器 0
0x14000104	MAIN_CHIP_CTRL1	主系统通用配置寄存器 1
0x14000108	MAIN_CHIP_CTRL2	主系统通用配置寄存器 2
0x1400010c	MAIN_CHIP_CTRL3	主系统通用配置寄存器 3
0x14000110	MAIN_CHIP_CTRL4	主系统通用配置寄存器 4
0x14000114	MAIN_CHIP_CTRL5	主系统通用配置寄存器 5
0x14000120	MAIN_CHIP_SAMP0	主系统采样参数寄存器 0
0x14000124	MAIN_CHIP_SAMP1	主系统采样参数寄存器 1
0x14000130	MAIN_CHIP_HPT0	主系统高精度计数器低 32位
0x14000134	MAIN_CHIP_HPT1	主系统高精度计数器高 32位
0x14000490	MAIN_GPIO_CFG0	主 GPIO00~15 复用配置寄存器
0x14000494	MAIN_GPIO_CFG1	主 GPIO16~31 复用配置寄存器
0x14000498	MAIN_GPIO_CFG2	主 GPIO32~47 复用配置寄存器
0x14000500	USB_PHY0	USB 的 PHY 配置寄存器 0
0x14000504	USB_PHY1	USB 的 PHY 配置寄存器 1

地址	名称	描述
0x14001040	CORE_INTISRO	路由给 CORE的低 32位中断状态
0x14001044	INTISRO	低 32位中断状态寄存器
0x14001048	CORE_INTISR1	路由给 CORE的高 32位中断状态
0x1400104c	INTISR1	高 32位中断状态寄存器
0x14001148	EXTIOI_ACK	扩展中断设备反馈寄存器
0x14001400	ENTRY0_0	8 位中断路由寄存器[0--7]
0x14001408	ENTRY8_0	8 位中断路由寄存器[8--15]
0x14001410	ENTRY16_0	8 位中断路由寄存器[16--23]
0x14001418	ENTRY24_0	8 位中断路由寄存器[24--31]
0x14001420	INTISR_0	低 32位中断状态寄存器
0x14001424	INTIEN_0	低 32位中断使能状态寄存器
0x14001428	INTSET_0	低 32位中断设置使能寄存器
0x1400142c	INTCLR_0	低 32 位中断清除寄存器，清除使能寄存器和脉冲触发的中断
0x14001430	INTPOL_0	低 32 位极性设置寄存器(电平中断)
0x14001434	INTEDGE_0	低 32 位触发方式寄存器（1：脉冲触发；0：电平触发）
0x14001440	ENTRY0_1	8 位中断路由寄存器[32--39]
0x14001448	ENTRY8_1	8 位中断路由寄存器[40--47]
0x14001450	ENTRY16_1	8 位中断路由寄存器[48--55]
0x14001458	ENTRY24_1	8 位中断路由寄存器[56--63]
0x14001460	INTISR_1	高32位中断状态寄存器
0x14001464	INTIEN_1	高32位中断使能状态寄存器
0x14001468	INTSET_1	高32位设置使能寄存器
0x1400146c	INTCLR_1	高32位中断清除寄存器，清除使能寄存器和脉冲触发的中断
0x14001470	INTPOL_1	高32位极性设置寄存器(电平中断)
0x14001474	INTEDGE_1	高32位触发方式寄存器（1：脉冲触发；0：电平触发）
0x140014c0	EXTIOI_MAP	扩展中断设备路由寄存器
0x14001500	Thsens_int_ctrl_Hi0	温度传感器高温中断控制寄存器 0
0x14001504	Thsens_int_ctrl_Hi1	温度传感器高温中断控制寄存器 1
0x14001508	Thsens_int_ctrl_Lo0	温度传感器低温中断控制寄存器 0
0x1400150c	Thsens_int_ctrl_Lo1	温度传感器低温中断控制寄存器 1
0x14001510	Thsens_int_status/clr	温度传感器中断状态寄存器
0x14001514	Thsens_value	温度传感器测量值
0x14001520	Thsens_scale_hi0	温度传感器高阈值寄存器 0
0x14001524	Thsens_scale_hi1	温度传感器高阈值寄存器 1
0x14001600	EXTIOI_IEN0	扩展中断设备使能寄存器 0
0x14001604	EXTIOI_IEN1	扩展中断设备使能寄存器 1

地址	名称	描述
0x14001608	EXTIOI_IEN2	扩展中断设备使能寄存器 2
0x1400160c	EXTIOI_IEN3	扩展中断设备使能寄存器 3
0x14001640	EXTIOI_POLO	扩展中断电平配置寄存器 0
0x14001644	EXTIOI_POL1	扩展中断电平配置寄存器 1
0x14001648	EXTIOI_POL2	扩展中断电平配置寄存器 2
0x1400164c	EXTIOI_POL3	扩展中断电平配置寄存器 3
0x14001700	EXTIOI_ISR0	扩展中断状态寄存器 0
0x14001704	EXTIOI_ISR1	扩展中断状态寄存器 1
0x14001708	EXTIOI_ISR2	扩展中断状态寄存器 2
0x1400170c	EXTIOI_ISR3	扩展中断状态寄存器 3
0x14001800	EXTIOI_CORE_ISR0	路由至 CORE 扩展中断状态寄存器 0
0x14001804	EXTIOI_CORE_ISR1	路由至 CORE 扩展中断状态寄存器 1
0x14001808	EXTIOI_CORE_ISR2	路由至 CORE 扩展中断状态寄存器 2
0x1400180c	EXTIOI_CORE_ISR3	路由至 CORE 扩展中断状态寄存器 3
0x14003ff0	CHIP_CHIPID0	芯片标识号 0
0x14003ff8	CHIP_CHIPID1	芯片标识号 1

表 5- 8 打印系统芯片配置寄存器列表

地址	名称	描述
0x15103000	PLL_SYS_0	SYS-PLL 低 32 位配置
0x15103004	PLL_SYS_1	SYS-PLL 高 32 位配置
0x15103008	PLL_DDR_0	内存控制器 PLL 低 32 位配置
0x1510300c	PLL_DDR_1	内存控制器 PLL 高 32 位配置
0x15103010	PLL_VID_0	VID-PLL 低 32 位配置
0x15103014	PLL_VID_1	VID-PLL 高 32 位配置
0x15103020	FREQSCALE0	设备时钟分频配置 0
0x15103024	FREQSCALE1	设备时钟分频配置 1
0x15103028	FREQSCALE2	设备时钟分频配置 2
0x1510302c	PLLCLK_EN	设备 PLL 时钟使能配置
0x15103030	PRDEV_CLK_CTRL	打印设备时钟门控配置
0x15103034	PRDEV_RST_CTRL	打印设备复位配置
0x15103040	PRT_GPIO_CFG0	打印 GPIO00~15 复用配置寄存器
0x15103044	PRT_GPIO_CFG1	打印 GPIO16~31 复用配置寄存器
0x15103048	PRT_GPIO_CFG2	打印 GPIO32~47 复用配置寄存器
0x1510304c	PRT_GPIO_CFG3	打印 GPIO48~63 复用配置寄存器
0x15103060	PRT_CHIPID0	打印系统识别号 0
0x15103064	PRT_CHIPID1	打印系统识别号 1
0x15103068	PRT_CHIPID2	打印系统识别号 2

地址	名称	描述
0x1510306c	PRT_CHIPID3	打印系统识别号 3
0x15103070	PRT_CHIP_CTRL0	打印系统通用配置 0
0x15103074	PRT_CHIP_CTRL1	打印系统通用配置 1
0x15103078	PRT_CHIP_SAMP	打印系统采样状态
0x1510307c	PRT_CHIP_HPT	打印系统计数寄存器
0x151030e0	PRT_DEVS_BASE	打印设备基址路由配置寄存器
0x151030f0	PRT_SHARED_CTRL0	打印共享设备配置 0
0x151030f4	PRT_SHARED_CTRL1	打印共享设备配置 1
0x151030f8	PRT_SHARED_CTRL2	打印共享设备配置 2
0x15103100	PRT_BOOT_ENTRY	打印启动入口配置
0x15103400	PRT_INT_ISR0	打印设备中断状态 0
0x15103404	PRT_INT_IEN0	打印设备中断使能配置 0
0x1510340c	PRT_INT_CLR0	打印设备中断清除配置 0
0x15103410	PRT_INT_POLO	打印设备中断电平配置 0
0x15103414	PRT_INT_EDGE0	打印设备中断边沿配置 0
0x15103418	PRT_INT_DUAL0	打印设备中断双沿配置 0
0x15103420	PRT_INT_ISR1	打印设备中断状态 1
0x15103424	PRT_INT_IEN1	打印设备中断使能配置 1
0x1510342c	PRT_INT_CLR1	打印设备中断清除配置 1
0x15103430	PRT_INT_POL1	打印设备中断电平配置 1
0x15103434	PRT_INT_EDGE1	打印设备中断边沿配置 1
0x15103438	PRT_INT_DUAL1	打印设备中断双沿配置 1
0x15103440	PRT_INT_ISR2	打印设备中断状态 2
0x15103444	PRT_INT_IEN2	打印设备中断使能配置 2
0x1510344c	PRT_INT_CLR2	打印设备中断清除配置 2
0x15103450	PRT_INT_POL2	打印设备中断电平配置 2
0x15103454	PRT_INT_EDGE2	打印设备中断边沿配置 2
0x15103458	PRT_INT_DUAL2	打印设备中断双沿配置 2
0x15103460	PRT_INT_MAP0	打印设备中断路由配置 0
0x15103464	PRT_INT_MAP1	打印设备中断路由配置 1
0x15103468	PRT_INT_MAP2	打印设备中断路由配置 2
0x1510346c	PRT_INT_MAP3	打印设备中断路由配置 3
0x15103470	PRT_INT_MAP4	打印设备中断路由配置 4
0x15103474	PRT_INT_MAP5	打印设备中断路由配置 5
0x15103800	MLBO_IPI_Status	0 号 Mailbox 的 IPI_Status 寄存器
0x15103804	MLBO_IPI_Enable	0 号 Mailbox 的 IPI_Enalbe 寄存器
0x15103808	MLBO_IPI_Set	0 号 Mailbox 的 IPI_Set 寄存器
0x1510380c	MLBO_IPI_Clear	0 号 Mailbox 的 IPI_Clear 寄存器
0x15103810	MLBO_message0	0 号 Mailbox 的 Message0 寄存器

地址	名称	描述
0x15103814	MLB0_message1	0号Mailbox的Message1寄存器
0x15103818	MLB0_message2	0号Mailbox的Message2寄存器
0x1510381c	MLB0_message3	0号Mailbox的Message3寄存器
0x15103820	MLB0_message4	0号Mailbox的Message4寄存器
0x15103824	MLB0_message5	0号Mailbox的Message5寄存器
0x15103828	MLB0_message6	0号Mailbox的Message6寄存器
0x1510382c	MLB0_message7	0号Mailbox的Message7寄存器
0x15103830	MLB1_IPI_Status	1号Mailbox的IPI_Status寄存器
0x15103834	MLB1_IPI_Enable	1号Mailbox的IPI_Enalbe寄存器
0x15103838	MLB1_IPI_Set	1号Mailbox的IPI_Set寄存器
0x1510383c	MLB1_IPI_Clear	1号Mailbox的IPI_Clear寄存器
0x15103840	MLB1_message0	1号Mailbox的Message0寄存器
0x15103844	MLB1_message1	1号Mailbox的Message1寄存器
0x15103848	MLB1_message2	1号Mailbox的Message2寄存器
0x1510384c	MLB1_message3	1号Mailbox的Message3寄存器
0x15103850	MLB1_message4	1号Mailbox的Message4寄存器
0x15103854	MLB1_message5	1号Mailbox的Message5寄存器
0x15103858	MLB1_message6	1号Mailbox的Message6寄存器
0x1510385c	MLB1_message7	1号Mailbox的Message7寄存器
0x15103860	MLB2_IPI_Status	2号Mailbox的IPI_Status寄存器
0x15103864	MLB2_IPI_Enable	2号Mailbox的IPI_Enalbe寄存器
0x15103868	MLB2_IPI_Set	2号Mailbox的IPI_Set寄存器
0x1510386c	MLB2_IPI_Clear	2号Mailbox的IPI_Clear寄存器
0x15103870	MLB2_message0	2号Mailbox的Message0寄存器
0x15103874	MLB2_message1	2号Mailbox的Message1寄存器
0x15103878	MLB2_message2	2号Mailbox的Message2寄存器
0x1510387c	MLB2_message3	2号Mailbox的Message3寄存器
0x15103880	MLB2_message4	2号Mailbox的Message4寄存器
0x15103884	MLB2_message5	2号Mailbox的Message5寄存器
0x15103888	MLB2_message6	2号Mailbox的Message6寄存器
0x1510388c	MLB2_message7	2号Mailbox的Message7寄存器
0x15103890	PRT_MailBox_MAP	打印系统MailBox中断路由配置

表 5- 9 扫描系统芯片配置寄存器列表

地址	名称	描述
0x15202030	SCDEV_CLK_CTRL	扫描设备时钟门控配置
0x15202034	SCDEV_RST_CTRL	扫描设备复位配置
0x15202040	SCA_GPIO_CFG0	扫描GPIO00~15复用配置寄存器
0x15202044	SCA_GPIO_CFG1	扫描GPIO16~31复用配置寄存器

地址	名称	描述
0x15202048	SCA_GPIO_CFG2	扫描 GPIO32~47 复用配置寄存器
0x15202060	SCA_CHIPID	扫描系统识别号
0x15202070	SCA_CHIP_CTRL	扫描系统通用配置
0x15202078	SCA_CHIP_SAMP	扫描系统采样状态
0x1520207c	SCA_CHIP_HPT	扫描系统计数寄存器
0x15202100	SCA_BOOT_ENTRY	扫描启动入口配置
0x15202400	SCA_INT_ISR0	扫描设备中断状态 0
0x15202404	SCA_INT_IEN0	扫描设备中断使能配置 0
0x1520240c	SCA_INT_CLR0	扫描设备中断清除配置 0
0x15202410	SCA_INT_POLO	扫描设备中断电平配置 0
0x15202414	SCA_INT_EDGE0	扫描设备中断边沿配置 0
0x15202418	SCA_INT_DUAL0	扫描设备中断双沿配置 0
0x15202420	SCA_INT_ISR1	扫描设备中断状态 1
0x15202424	SCA_INT_IEN1	扫描设备中断使能配置 1
0x1520242c	SCA_INT_CLR1	扫描设备中断清除配置 1
0x15202430	SCA_INT_POL1	扫描设备中断电平配置 1
0x15202434	SCA_INT_EDGE1	扫描设备中断边沿配置 1
0x15202438	SCA_INT_DUAL1	扫描设备中断双沿配置 1
0x15202440	SCA_INT_ISR2	扫描设备中断状态 2
0x15202444	SCA_INT_IEN2	扫描设备中断使能配置 2
0x1520244c	SCA_INT_CLR2	扫描设备中断清除配置 2
0x15202450	SCA_INT_POL2	扫描设备中断电平配置 2
0x15202454	SCA_INT_EDGE2	扫描设备中断边沿配置 2
0x15202458	SCA_INT_DUAL2	扫描设备中断双沿配置 2
0x15202460	SCA_INT_MAP0	扫描设备中断路由配置 0
0x15202464	SCA_INT_MAP1	扫描设备中断路由配置 1
0x15202468	SCA_INT_MAP2	扫描设备中断路由配置 2
0x1520246c	SCA_INT_MAP3	扫描设备中断路由配置 3
0x15202470	SCA_INT_MAP4	扫描设备中断路由配置 4
0x15202474	SCA_INT_MAP5	扫描设备中断路由配置 5
0x15202800	MLBO_IPI_Status	0 号 Mailbox 的 IPI_Status 寄存器
0x15202804	MLBO_IPI_Enable	0 号 Mailbox 的 IPI_Enalbe 寄存器
0x15202808	MLBO_IPI_Set	0 号 Mailbox 的 IPI_Set 寄存器
0x1520280c	MLBO_IPI_Clear	0 号 Mailbox 的 IPI_Clear 寄存器
0x15202810	MLBO_message0	0 号 Mailbox 的 Message0 寄存器
0x15202814	MLBO_message1	0 号 Mailbox 的 Message1 寄存器
0x15202818	MLBO_message2	0 号 Mailbox 的 Message2 寄存器
0x1520281c	MLBO_message3	0 号 Mailbox 的 Message3 寄存器
0x15202820	MLBO_message4	0 号 Mailbox 的 Message4 寄存器

地址	名称	描述
0x15202824	MLB0_message5	0号Mailbox的Message5寄存器
0x15202828	MLB0_message6	0号Mailbox的Message6寄存器
0x1520282c	MLB0_message7	0号Mailbox的Message7寄存器
0x15202830	MLB1_IPI_Status	1号Mailbox的IPI_Status寄存器
0x15202834	MLB1_IPI_Enable	1号Mailbox的IPI_Enalbe寄存器
0x15202838	MLB1_IPI_Set	1号Mailbox的IPI_Set寄存器
0x1520283c	MLB1_IPI_Clear	1号Mailbox的IPI_Clear寄存器
0x15202840	MLB1_message0	1号Mailbox的Message0寄存器
0x15202844	MLB1_message1	1号Mailbox的Message1寄存器
0x15202848	MLB1_message2	1号Mailbox的Message2寄存器
0x1520284c	MLB1_message3	1号Mailbox的Message3寄存器
0x15202850	MLB1_message4	1号Mailbox的Message4寄存器
0x15202854	MLB1_message5	1号Mailbox的Message5寄存器
0x15202858	MLB1_message6	1号Mailbox的Message6寄存器
0x1520285c	MLB1_message7	1号Mailbox的Message7寄存器
0x15202860	MLB2_IPI_Status	2号Mailbox的IPI_Status寄存器
0x15202864	MLB2_IPI_Enable	2号Mailbox的IPI_Enalbe寄存器
0x15202868	MLB2_IPI_Set	2号Mailbox的IPI_Set寄存器
0x1520286c	MLB2_IPI_Clear	2号Mailbox的IPI_Clear寄存器
0x15202870	MLB2_message0	2号Mailbox的Message0寄存器
0x15202874	MLB2_message1	2号Mailbox的Message1寄存器
0x15202878	MLB2_message2	2号Mailbox的Message2寄存器
0x1520287c	MLB2_message3	2号Mailbox的Message3寄存器
0x15202880	MLB2_message4	2号Mailbox的Message4寄存器
0x15202884	MLB2_message5	2号Mailbox的Message5寄存器
0x15202888	MLB2_message6	2号Mailbox的Message6寄存器
0x1520288c	MLB2_message7	2号Mailbox的Message7寄存器
0x15202890	SCA_MailBox_MAP	扫描系统Mailbox中断路由配置

一、主系统通用配置寄存器，主要包括以下寄存器说明：

#### 5.4.1 主系统通用配置寄存器 0

主系统通用配置寄存器0，包括UART引脚复用模式、GMAC、USB、内存控制器的配置等。

寄存器地址：0x14000100。



表 5- 10 通用配置寄存器 0

位域	名称	访问	缺省值	描述
31:28	uart0_enable	RW	0x0	UART0对应的UART控制器模式及引脚复用关系(具体引脚复用分配参看芯片数据手册uart接口定义): Bit0: 对应uart0, 保留; Bit1: 为1对应uart1复用有效(若bit2为0则uart1复用为4线模式, 若bit2为1则复用为2线模式), 为0对应uart0的8线模式, uart1复用无效; Bit2: 为1对应uart2复用为2线模式, 为0对应uart0的8线模式, uart2复用无效; Bit3: 为1对应uart3复用为2线模式, 为0对应uart0的8线模式, uart3复用无效。
27:26	hpet_int_ctrl	RW	0x0	hpet0~1中断输出模式配置: 0: 单中断输出模式(内部3个计数器共享1个中断); 1: 三中中断输出模式(内部3个计数器独立3个中断)
25:24	Reserved	RO	0x0	-
23:20	conf_usb_flush_idle	RW	0xf	设置清空write buffer前空闲周期数
19	conf_usb_prefetch	RW	0x0	USB接口总线使能读预取
18	conf_usb_flush_wr	RW	0x0	USB接口总线设置写命令发出后是否清空read buffer
17	conf_usb_stop_waw	RW	0x0	USB接口总线是否允许在上一个写完成前发出写命令
16	conf_usb_stop_raw	RW	0x1	USB接口总线是否允许在上一个写完成前发出读命令
15	extioi_en	RW	0x0	扩展中断使能控制位: 1: 打印扩展中断(传统/扩展中断同时有效); 0: 关闭扩展中断(仅传统中断有效)
14	gmac1_mii_sel	RW	0x0	GMAC1接口MII模式选择: 1: MII接口模式; 0: RGMII接口模式
13:12	usbclk_mode	RW	0x3	2个usb接口参考时钟模式选择: 00: USB0选择外部晶体时钟输入(时钟频率12MHz), USB1选择由USB0输出时钟; 01: USB0~1全部选择外部晶振时钟分别由各自X0输入(时钟频率12MHz); 10: USB0选择外部晶振时钟X0输入(时钟频率12MHz), USB1选择由USB0输出时钟; 11: USB0选择内部PLL时钟产生(时钟频率50MHz), USB1选择由USB0输出时钟;
11:4	conf_iodma_spare_rd	RW	0x0	iodma读操作最大数设置
3:2	Reserved	RO	0x0	-
1	ddr3_regs_default	RW	0x0	窗口不命中处理 0: 关闭内存控制器的该功能 1: 当所有窗口不命中时, 由内存控制器给出响应, 防止CPU卡死。
0	ddr3_regs_disable	RW	0x0	DDR配置空间关闭, 高有效 DDR控制器在内存空间中开辟了一小段配置空间(1MB @0x0ff0, 0000), 在关闭后软件就可以使用这段空间。为避免意外访问, 建议在配置完成后及时关闭

## 5.4.2 主系统通用配置寄存器 1

主系统通用配置寄存器1, 包括对 USB、GMAC 等 Cache 一致性配置等。

寄存器地址: 0x14000104。

表 5- 11 通用配置寄存器 1

位域	名称	访问	缺省值	描述
31:28	Reserved	RO	0x0	-
27	scdev_coherent	RW	0x0	扫描系统设备DMA内部互联总线CACHE访问配置位(开启IO设备CACHE使能位后配置有效): 1: 开启CACHE加速访问; 0: 关闭CACHE加速访问



位域	名称	访问	缺省值	描述
26	prdev_coherent	RW	0x0	打印系统设备DMA内部互联总线CACHE访问配置位(开启IO设备CACHE使能位后配置有效): 1: 开启CACHE加速访问; 0: 关闭CACHE加速访问
25	apb_coherent	RW	0x0	APB设备DMA内部互联总线CACHE访问配置位(开启IO设备CACHE使能位后配置有效): 1: 开启CACHE加速访问; 0: 关闭CACHE加速访问
24	image_coherent	RW	0x0	IMAGE设备(JBIG+JPEG)内部互联总线CACHE访问配置位(开启IO设备CACHE使能位后配置有效): 1: 开启CACHE加速访问; 0: 关闭CACHE加速访问
23	rsm_coherent	RW	0x0	RESUME设备(GMAC0+OTG)内部互联总线CACHE访问配置位(开启IO设备CACHE使能位后配置有效): 1: 开启CACHE加速访问; 0: 关闭CACHE加速访问
22	gmactl_coherent	RW	0x0	GMAC1内部互联总线CACHE访问配置位(开启IO设备CACHE使能位后配置有效): 1: 开启CACHE加速访问; 0: 关闭CACHE加速访问
21	usb_coherent	RW	0x0	USB0~1内部互联总线CACHE访问配置位(开启IO设备CACHE使能位后配置有效): 1: 开启CACHE加速访问; 0: 关闭CACHE加速访问
20	Reserved	RO	0x0	-
19	scdev_coherent_enable	RW	0x0	扫描系统设备DMA内部互联CACHE访问使能位: 1: 使能设备CACHE访问配置有效, 配置对应设备coherent位开启CACHE加速访问; 0: 关闭设备CACHE访问配置, 对应设备coherent位配置无效, 此时设备可通过内部总线地址最高位(第32位)选择是否CACHE访问(1:开启, 0:关闭)
18	prdev_coherent_enable	RW	0x0	打印系统设备DMA内部互联CACHE访问使能位: 1: 使能设备CACHE访问配置有效, 配置对应设备coherent位开启CACHE加速访问; 0: 关闭设备CACHE访问配置, 对应设备coherent位配置无效, 此时设备可通过内部总线地址最高位(第32位)选择是否CACHE访问(1:开启, 0:关闭)
17	apb_coherent_enable	RW	0x0	APB设备DMA内部互联CACHE访问使能位: 1: 使能设备CACHE访问配置有效, 配置对应设备coherent位开启CACHE加速访问; 0: 关闭设备CACHE访问配置, 对应设备coherent位配置无效, 此时设备可通过内部总线地址最高位(第32位)选择是否CACHE访问(1:开启, 0:关闭)
16	image_coherent_enable	RW	0x0	IMAGE设备(JBIG+JPEG)内部互联CACHE访问使能位: 1: 使能设备CACHE访问配置有效, 配置对应设备coherent位开启CACHE加速访问; 0: 关闭设备CACHE访问配置, 对应设备coherent位配置无效, 此时设备可通过内部总线地址最高位(第32位)选择是否CACHE访问(1:开启, 0:关闭)
15	rsm_coherent_enable	RW	0x0	RESUME设备(GMAC0+OTG)内部互联CACHE访问使能位: 1: 使能设备CACHE访问配置有效, 配置对应设备coherent位开启CACHE加速访问; 0: 关闭设备CACHE访问配置, 对应设备coherent位配置无效, 此时设备可通过内部总线地址最高位(第32位)选择是否CACHE访问(1:开启, 0:关闭)
14	gmactl_coherent_enable	RW	0x0	GMAC1设备内部互联CACHE访问使能位: 1: 使能设备CACHE访问配置有效, 配置对应设备coherent位开启CACHE加速访问; 0: 关闭设备CACHE访问配置, 对应设备coherent位配置无效, 此时设备可通过内部总线地址最高位(第32位)选择是否CACHE访问(1:开启, 0:关闭)

位域	名称	访问	缺省值	描述
13	usb_coherent_enable	RW	0x0	USB0~1设备内部互联CACHE访问使能位： 1：使能设备CACHE访问配置有效，配置对应设备coherent位开启CACHE加速访问； 0：关闭设备CACHE访问配置，对应设备coherent位配置无效，此时设备可通过内部总线地址最高位(第32位)选择是否CACHE访问(1:开启, 0:关闭)
12	Reserved	RO	0x0	-
11	scdev_order_en	RW	0x0	扫描系统设备DMA内部互联读写请求按序执行使能位，高电平有效
10	prdev_order_en	RW	0x0	打印系统设备DMA内部互联读写请求按序执行使能位，高电平有效
9	apb_order_en	RW	0x0	APB设备DMA内部互联读写请求按序执行使能位，高电平有效
8	image_order_en	RW	0x0	IMAGE(JBIG+JPEG)内部互联读写请求按序执行使能位，高电平有效
7	resume_order_en	RW	0x0	RESUME(GMAC0+OTG)内部互联读写请求按序执行使能位，高电平有效
6	gmac1_order_en	RW	0x0	GMAC1内部互联读写请求按序执行使能位，高电平有效
5	usb_order_en	RW	0x0	USB0~1内部互联读写请求按序执行使能位，高电平有效
4	cpu_order_en	RW	0x0	CPU内部互联读写请求按序执行使能位，高电平有效
3:0	uart1_enable	RW	0x0	UART1对应的UART控制器模式及引脚复用关系(具体引脚复用分配参看芯片数据手册uart接口定义)： Bit0：对应uart1，保留； Bit1：保留； Bit2：保留； Bit3：为1对应uart3复用为2线模式，为0对应uart1的4线模式，uart3复用无效。

### 5.4.3 主系统通用配置寄存器 2

主系统通用配置寄存器2，包括对各功能模块 DMA 内部互联读写请求优先级的配置等。

寄存器地址：0x14000108。

表 5- 12 通用配置寄存器 2

位域	名称	访问	缺省值	描述
31:30	Reserved	RO	0x0	-
29:28	sm3_dma_map	RW	0x0	SM3模块数据DMA通道路由选择： 00：选择加解密DMA通道0；01：选择加解密DMA通道1；其他：无效
27:26	sm4_rdma_map	RW	0x1	SM4模块读数据DMA通道路由选择： 00：选择加解密DMA通道0；01：选择加解密DMA通道1；其他：无效
25:24	sm4_wdma_map	RW	0x0	SM4模块写数据DMA通道路由选择： 00：选择加解密DMA通道0；01：选择加解密DMA通道1；其他：无效
23:22	des_rdma_map	RW	0x1	DES模块读数据DMA通道路由选择： 00：选择加解密DMA通道0；01：选择加解密DMA通道1；其他：无效
21:20	des_wdma_map	RW	0x0	DES模块写数据DMA通道路由选择： 00：选择加解密DMA通道0；01：选择加解密DMA通道1；其他：无效
19:18	aes_rdma_map	RW	0x1	AES模块读数据DMA通道路由选择： 00：选择加解密DMA通道0；01：选择加解密DMA通道1；其他：无效

位域	名称	访问	缺省值	描述
17:16	aes_wdma_map	RW	0x0	AES模块写数据DMA通路由选择： 00：选择加解密DMA通道0；01：选择加解密DMA通道1；其他：无效
15	scdev_read_upgrade	RW	0x0	扫描系统设备DMA内部互联总线读请求优先级使能配置位（高电平有效，默认按序访问）
14	prdev_read_upgrade	RW	0x0	打印系统设备DMA内部互联总线读请求优先级使能配置位（高电平有效，默认按序访问）
13	apb_read_upgrade	RW	0x0	APB设备DMA内部互联总线读请求优先级使能配置位（高电平有效，默认按序访问）
12	image_read_upgrade	RW	0x0	IMAGE (JBIG+JPEG) 内部互联总线读请求优先级使能配置位（高电平有效，默认按序访问）
11	rsm_read_upgrade	RW	0x0	RESUME (GMACO+OTG) 内部互联总线读请求优先级使能配置位（高电平有效，默认按序访问）
10	gmac1_read_upgrade	RW	0x0	GMAC1内部互联总线读请求优先级使能配置位（高电平有效，默认按序访问）
9	usb_read_upgrade	RW	0x0	USB0~1内部互联总线读请求优先级使能配置位（高电平有效，默认按序访问）
8	Reserved	RO	0x0	-
7	scdev_write_upgrade	RW	0x0	扫描系统设备DMA内部互联总线写请求优先级使能配置位（高电平有效，默认按序访问）
6	prdev_write_upgrade	RW	0x0	打印系统设备DMA内部互联总线写请求优先级使能配置位（高电平有效，默认按序访问）
5	apb_write_upgrade	RW	0x0	APB设备DMA内部互联总线写请求优先级使能配置位（高电平有效，默认按序访问）
4	image_write_upgrade	RW	0x0	IMAGE (JBIG+JPEG) 内部互联总线写请求优先级使能配置位（高电平有效，默认按序访问）
3	rsm_write_upgrade	RW	0x0	RESUME (GMACO+OTG) 内部互联总线写请求优先级使能配置位（高电平有效，默认按序访问）
2	gmac1_write_upgrade	RW	0x0	GMAC1内部互联总线写请求优先级使能配置位（高电平有效，默认按序访问）
1	usb_write_upgrade	RW	0x0	USB0~1内部互联总线写请求优先级使能配置位（高电平有效，默认按序访问）
0	Reserved	RO	0x0	-

### 5.4.4 主系统通用配置寄存器 3

主系统2通用配置寄存器3，包括各功能模块时钟门控的控制。

寄存器地址：0x1400010c。

表 5- 13 通用配置寄存器 3

位域	名称	访问	缺省值	描述
31:24	Reserved	RO	0x1	-
23	jpeg_clk_ctrl	RW	0x1	JPEG模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
22	jbig_clk_ctrl	RW	0x1	JBIG模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
21	usb_clk_ctrl	RW	0x1	USB0~1模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
20	gmac1_clk_ctrl	RW	0x1	GMAC1模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
19	Reserved	RO	0x1	-
18	Reserved	RO	0x1	-
17	apbdev_clk_ctrl	RW	0x1	主系统APB各设备模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
16	sdio_clk_ctrl	RW	0x1	SDIO模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。

位域	名称	访问	缺省值	描述
15	Reserved	RO	0x1	-
14	sm4_clk_ctrl	RW	0x1	SM4模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
13	sm3_clk_ctrl	RW	0x1	SM3模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
12	des_clk_ctrl	RW	0x1	DES模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
11	aes_clk_ctrl	RW	0x1	AES模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
10	rng_clk_ctrl	RW	0x1	RNG模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
9	sm2_clk_ctrl	RW	0x1	SM2模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
8	encdma_clk_ctrl	RW	0x1	ENCRYPT-DMA模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
7	pwm_clk_ctrl	RW	0x1	PWM模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
6	pm2io_clk_ctrl	RW	0x1	PM2IO模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
5	wdt_clk_ctrl	RW	0x1	WDT模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
4	hpet_clk_ctrl	RW	0x1	HPET模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
3	spi_clk_ctrl	RW	0x1	SPI模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
2	gpio_clk_ctrl	RW	0x1	GPIO模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
1	i2c_clk_ctrl	RW	0x1	I2C模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
0	uart_clk_ctrl	RW	0x1	UART模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。

### 5.4.5 主系统通用配置寄存器 4

主系统通用配置寄存器4，包括各功能模块软件复位的控制。

寄存器地址：0x14000110。

表 5- 14 通用配置寄存器 4

位域	名称	访问	缺省值	描述
31:24	Reserved	RO	0x0	-
23	jpeg_rst_ctrl	RW	0x1	JPEG模块复位软件配置： 1：模块复位无效；0：模块复位有效。
22	jbig_rst_ctrl	RW	0x1	JBIG模块复位软件配置： 1：模块复位无效；0：模块复位有效。
21	usb_rst_ctrl	RW	0x1	USB <sup>0~1</sup> 模块复位软件配置： 1：模块复位无效；0：模块复位有效。
20	gmac1_rst_ctrl	RW	0x1	GMAC1模块复位软件配置： 1：模块复位无效；0：模块复位有效。
19	Reserved	RO	0x1	-
18	Reserved	RO	0x1	-
17	apbdev_rst_ctrl	RW	0x1	主系统所有APB设备模块复位软件配置： 1：模块复位无效；0：模块复位有效。
16	sdio_rst_ctrl	RW	0x1	SDIO模块复位软件配置： 1：模块复位无效；0：模块复位有效。
15	Reserved	RO	0x1	-

位域	名称	访问	缺省值	描述
14	sm4_rst_ctrl	RW	0x1	SM4模块复位软件配置： 1：模块复位无效；0：模块复位有效。
13	sm3_rst_ctrl	RW	0x1	SM3模块复位软件配置： 1：模块复位无效；0：模块复位有效。
12	des_rst_ctrl	RW	0x1	DES模块复位软件配置： 1：模块复位无效；0：模块复位有效。
11	aes_rst_ctrl	RW	0x1	AES模块复位软件配置： 1：模块复位无效；0：模块复位有效。
10	rng_rst_ctrl	RW	0x1	RNG模块复位软件配置： 1：模块复位无效；0：模块复位有效。
9	sm2_rst_ctrl	RW	0x1	SM2模块复位软件配置： 1：模块复位无效；0：模块复位有效。
8	encdma_rst_ctrl	RW	0x1	ENCRYPT-DMA模块复位软件配置： 1：模块复位无效；0：模块复位有效。
7	pwm_rst_ctrl	RW	0x1	PWM模块复位软件配置： 1：模块复位无效；0：模块复位有效。
6	pm2io_rst_ctrl	RW	0x1	PM2IO模块复位软件配置： 1：模块复位无效；0：模块复位有效。
5	wdt_rst_ctrl	RW	0x1	WDT模块复位软件配置： 1：模块复位无效；0：模块复位有效。
4	hpet_rst_ctrl	RW	0x1	HPET模块复位软件配置： 1：模块复位无效；0：模块复位有效。
3	spi_rst_ctrl	RW	0x1	SPI模块复位软件配置： 1：模块复位无效；0：模块复位有效。
2	gpio_rst_ctrl	RW	0x1	GPIO模块复位软件配置： 1：模块复位无效；0：模块复位有效。
1	i2c_rst_ctrl	RW	0x1	I2C模块复位软件配置： 1：模块复位无效；0：模块复位有效。
0	uart_rst_ctrl	RW	0x1	UART模块复位软件配置： 1：模块复位无效；0：模块复位有效。

### 5.4.6 主系统通用配置寄存器 5

主系统通用配置寄存器5。

寄存器地址：0x14000114。

表 5- 15 通用配置寄存器 5

位域	名称	访问	缺省值	描述
31:0	Reserved	RO	0x0	

### 5.4.7 主系统采样参数寄存器

主系统采样参数寄存器，包括芯片启动配置相关信息。

寄存器地址：0x14000120。

表 5- 16 主系统采样参数寄存器

位域	名称	访问	缺省值	描述
31:7	Reserved	RO	0x0	-
6:5	otg_clkmode	RO	0x0	OTG参考时钟模式输入 00=外部差分晶体输入(XI/XO) 01/10=外部单端晶振输入(XO) 11=内部参考时钟输入
4	sdiol_mode	RO	0x0	SDIO1模式配置输入 0=SDIO模式 1=EMMC模式

3	sdio0_mode	RO	0x0	SDIO0模式配置输入 0=SDIO模式 1=EMMC模式
2:1	clk_sel	RO	0x0	芯片内部PLL输出时钟上电配置选择： 00：硬件低频时钟配置模式，PLL按照低频配置参数输出时钟 (NODE:500M, DDR:480M, NET:320M)； 01：硬件高频时钟配置模式，PLL按照高频配置参数输出时钟 (NODE:725M, DDR:600M, NET:400M)； 10：软件配置模式，PLL按照软件配置选择输出时钟； 11：硬件bypass模式，PLL输出时钟全部使用外部输入系统时钟。
0	boot_sel	RO	0x0	芯片启动选择方式： 0：SPI0启动；1：SDIO0/eMMC0启动。

#### 5.4.8 主系统高精度计数器 0

主系统计数寄存器0，64位高精度时钟计数器0~31位，工作频率为内部总线时钟频率。  
寄存器地址：0x14000130。

表 5- 17 主系统高精度计数器 0

位域	名称	访问	缺省值	描述
31:0	CHIP_HPT[31:0]	RW	0x0	64位高精度时钟计数器低32位

#### 5.4.9 主系统高精度计数器 1

主系统计数寄存器1，64位高精度时钟计数器32~63位，工作频率为内部总线时钟频率。  
寄存器地址：0x14000134。

表 5- 18 主系统高精度计数器 1

位域	名称	访问	缺省值	描述
31:0	CHIP_HPT[63:32]	RW	0x0	64位高精度时钟计数器高32位

#### 5.4.10 主系统 GPIO0~15 复用配置寄存器

主系统 GPIO0~15复用配置寄存器。  
寄存器地址：0x14000490。

表 5- 19 主系统 GPIO0~15 复用配置寄存器

位域	名称	访问	缺省值	描述
31:30	GPIO15_MUX	RW	0x0	GPIO15引脚复用配置： 00：复用为GPIO15；01：第一复用； 10：第二复用；11：引脚主功能。
29:28	GPIO14_MUX	RW	0x0	GPIO14引脚复用配置： 00：复用为GPIO14；01：第一复用； 10：第二复用；11：引脚主功能。
27:26	GPIO13_MUX	RW	0x0	GPIO13引脚复用配置： 00：复用为GPIO13；01：第一复用； 10：第二复用；11：引脚主功能。
25:24	GPIO12_MUX	RW	0x0	GPIO12引脚复用配置： 00：复用为GPIO12；01：第一复用； 10：第二复用；11：引脚主功能。
23:22	GPIO11_MUX	RW	0x0	GPIO11引脚复用配置： 00：复用为GPIO11；01：第一复用； 10：第二复用；11：引脚主功能。



位域	名称	访问	缺省值	描述
21:20	GPI010_MUX	RW	0x0	GPI010引脚复用配置： 00：复用为GPI010；01：第一复用； 10：第二复用； 11：引脚主功能。
19:18	GPI009_MUX	RW	0x0	GPI009引脚复用配置： 00：复用为GPI009；01：第一复用； 10：第二复用； 11：引脚主功能。
17:16	GPI008_MUX	RW	0x0	GPI008引脚复用配置： 00：复用为GPI008；01：第一复用； 10：第二复用； 11：引脚主功能。
15:14	GPI007_MUX	RW	0x0	GPI007引脚复用配置： 00：复用为GPI007；01：第一复用； 10：第二复用； 11：引脚主功能。
13:12	GPI006_MUX	RW	0x0	GPI006引脚复用配置： 00：复用为GPI006；01：第一复用； 10：第二复用； 11：引脚主功能。
11:10	GPI005_MUX	RW	0x0	GPI005引脚复用配置： 00：复用为GPI005；01：第一复用； 10：第二复用； 11：引脚主功能。
9:8	GPI004_MUX	RW	0x0	GPI004引脚复用配置： 00：复用为GPI004；01：第一复用； 10：第二复用； 11：引脚主功能。
7:6	GPI003_MUX	RW	0x0	GPI003引脚复用配置： 00：复用为GPI003；01：第一复用； 10：第二复用； 11：引脚主功能。
5:4	GPI002_MUX	RW	0x0	GPI002引脚复用配置： 00：复用为GPI002；01：第一复用； 10：第二复用； 11：引脚主功能。
3:2	GPI001_MUX	RW	0x0	GPI001引脚复用配置： 00：复用为GPI001；01：第一复用； 10：第二复用； 11：引脚主功能。
1:0	GPI000_MUX	RW	0x0	GPI000引脚复用配置： 00：复用为GPI000；01：第一复用； 10：第二复用； 11：引脚主功能。

#### 5.4.11 主系统 GPI016~31 复用配置寄存器

主系统 GPI016~31复用配置寄存器。

寄存器地址：0x14000494。

表 5- 20 主系统 GPI016~31 复用配置寄存器

位域	名称	访问	缺省值	描述
31:30	GPI031_MUX	RW	0x0	GPI031引脚复用配置： 00：复用为GPI031；01：第一复用； 10：第二复用； 11：引脚主功能。
29:28	GPI030_MUX	RW	0x0	GPI030引脚复用配置： 00：复用为GPI030；01：第一复用； 10：第二复用； 11：引脚主功能。
27:26	GPI029_MUX	RW	0x0	GPI029引脚复用配置： 00：复用为GPI029；01：第一复用； 10：第二复用； 11：引脚主功能。
25:24	GPI028_MUX	RW	0x0	GPI028引脚复用配置： 00：复用为GPI028；01：第一复用； 10：第二复用； 11：引脚主功能。
23:22	GPI027_MUX	RW	0x0	GPI027引脚复用配置： 00：复用为GPI027；01：第一复用； 10：第二复用； 11：引脚主功能。

位域	名称	访问	缺省值	描述
21:20	GPI026_MUX	RW	0x0	GPI026引脚复用配置： 00：复用为GPI026；01：第一复用； 10：第二复用； 11：引脚主功能。
19:18	GPI025_MUX	RW	0x0	GPI025引脚复用配置： 00：复用为GPI025；01：第一复用； 10：第二复用； 11：引脚主功能。
17:16	GPI024_MUX	RW	0x0	GPI024引脚复用配置： 00：复用为GPI024；01：第一复用； 10：第二复用； 11：引脚主功能。
15:14	GPI023_MUX	RW	0x0	GPI023引脚复用配置： 00：复用为GPI023；01：第一复用； 10：第二复用； 11：引脚主功能。
13:12	GPI022_MUX	RW	0x0	GPI022引脚复用配置： 00：复用为GPI022；01：第一复用； 10：第二复用； 11：引脚主功能。
11:10	GPI021_MUX	RW	0x0	GPI021引脚复用配置： 00：复用为GPI021；01：第一复用； 10：第二复用； 11：引脚主功能。
9:8	GPI020_MUX	RW	0x0	GPI020引脚复用配置： 00：复用为GPI020；01：第一复用； 10：第二复用； 11：引脚主功能。
7:6	GPI019_MUX	RW	0x0	GPI019引脚复用配置： 00：复用为GPI019；01：第一复用； 10：第二复用； 11：引脚主功能。
5:4	GPI018_MUX	RW	0x0	GPI018引脚复用配置： 00：复用为GPI018；01：第一复用； 10：第二复用； 11：引脚主功能。
3:2	GPI017_MUX	RW	0x0	GPI017引脚复用配置： 00：复用为GPI017；01：第一复用； 10：第二复用； 11：引脚主功能。
1:0	GPI016_MUX	RW	0x0	GPI016引脚复用配置： 00：复用为GPI016；01：第一复用； 10：第二复用； 11：引脚主功能。

## 5.4.12 主系统 GPI032~43 复用配置寄存器

主系统 GPI032~43复用配置寄存器。

寄存器地址：0x14000498。

表 5- 21 主系统 GPI032~43 复用配置寄存器

位域	名称	访问	缺省值	描述
31:24	Reserved	RO	0x0	-
23:22	GPI043_MUX	RW	0x0	GPI043引脚复用配置： 00：复用为GPI043；01：第一复用； 10：第二复用； 11：引脚主功能。
21:20	GPI042_MUX	RW	0x0	GPI042引脚复用配置： 00：复用为GPI042；01：第一复用； 10：第二复用； 11：引脚主功能。
19:18	GPI041_MUX	RW	0x0	GPI041引脚复用配置： 00：复用为GPI041；01：第一复用； 10：第二复用； 11：引脚主功能。
17:16	GPI040_MUX	RW	0x0	GPI040引脚复用配置： 00：复用为GPI040；01：第一复用； 10：第二复用； 11：引脚主功能。



位域	名称	访问	缺省值	描述
15:14	GPI039_MUX	RW	0x0	GPI039引脚复用配置： 00：复用为GPI039；01：第一复用； 10：第二复用； 11：引脚主功能。
13:12	GPI038_MUX	RW	0x0	GPI038引脚复用配置： 00：复用为GPI038；01：第一复用； 10：第二复用； 11：引脚主功能。
11:10	GPI037_MUX	RW	0x0	GPI037引脚复用配置： 00：复用为GPI037；01：第一复用； 10：第二复用； 11：引脚主功能。
9:8	GPI036_MUX	RW	0x0	GPI036引脚复用配置： 00：复用为GPI036；01：第一复用； 10：第二复用； 11：引脚主功能。
7:6	GPI035_MUX	RW	0x0	GPI035引脚复用配置： 00：复用为GPI035；01：第一复用； 10：第二复用； 11：引脚主功能。
5:4	GPI034_MUX	RW	0x0	GPI034引脚复用配置： 00：复用为GPI034；01：第一复用； 10：第二复用； 11：引脚主功能。
3:2	GPI033_MUX	RW	0x0	GPI033引脚复用配置： 00：复用为GPI033；01：第一复用； 10：第二复用； 11：引脚主功能。
1:0	GPI032_MUX	RW	0x0	GPI032引脚复用配置： 00：复用为GPI032；01：第一复用； 10：第二复用； 11：引脚主功能。

### 5.4.13 USB PHY 配置寄存器 0

USB PHY 配置寄存器0，配置 USB 接口0的电气特性。

寄存器地址：0x14000500。

表 5- 22 USB PHY 配置寄存器 0

位域	名称	访问	缺省值	描述
31	Reserved	R/O	0	
30	cfg_fs_data_mod	R/W	0	低速模式数据有效使能控制 0：低速模式下，数据信号不受使能信号控制其有效性 1：低速模式下，数据信号受到使能信号控制其有效性
29	commononn0	R/W	0	通用模块关电模式 1：在suspend时，X0, Bias, PLL模块掉电；在睡眠时，Bias, PLL掉电 0：在suspend或睡眠时X0, Bias, PLL都有电
28	dmpulldown0	R/W	0	dm端口下拉电阻使能 1：D-使能 0：D-关闭
27	dppulldown0	R/W	0	dp端口下拉电阻使能 1：D+使能 0：D+关闭

位域	名称	访问	缺省值	描述
26:25	txrestune0	R/W	0	该信号调整了驱动电阻，用来补偿在发送器到电缆之间的寄生电阻 11: -4Ω 10: -2Ω 01: 默认 00: +1.5Ω
24	txpreempulsetune0	R/W	0	调整了在高速模式下预加强电流在dp, dm电缆上的持续时间，高速收发器预加强电流持续时间根据时间单位来定义 1: 1X, 短持续时间 0: 2X, 长持续时间
23:22	txpreempamtune0	R/W	0	调整了在高速模式从J到K或从K到J的跳变时提供的电流量，高速收发器预加强电流量根据电流量单位来定义 11: HS传输预加强3X电流 10: HS传输预加强2X电流 01: HS传输预加强1X电流 00: HS传输预加强电流关闭
21:20	txhsxvtune0	R/W	0	在高速模式下调整dp, dm信号交叉时的电压 11: default 10: +15mv 01: -15mv 00: reserved
19:18	txrisetune0	R/W	0	调整了高速波形上升、下降沿次数 11: -10% 10: default 01: +15% 00: +20%
17:14	txvrefltune0	R/W	0	调整高速直流电压 1111: +8.75% 1110: +7.5% 1101: +6.25% 1100: +5% 1011: +3.75% 1010: +2.5% 1001: +1.25% 1000: default 0111: -1.25% 0110: -2.5% 0101: -3.75% 0100: -5% 0011: -6.25% 0010: -7.5% 0001: -8.75% 0000: -10%
13:10	txfslstune0	R/W	0	调整了低、全速单端源阻抗 1111: -5% 0111: -2.5% 0011: default 0001: +2.5% 0000: +5%

位域	名称	访问	缺省值	描述
9:7	sqrxtune0	R/W	0	调整门限电压来检测有效的高速数据 111: -20% 110: -15% 101: -10% 100: -5% 011: default 010: +5% 001: +10% 000: +15%
6:4	compdistune0	R/W	0	调整门限电压用于检测在主控制器上的一个断开连接事件 111: +4.5% 110: +3% 101: +1.5% 100: default 011: -1.5% 010: -3% 001: -4.5% 000: -6%
3:1	otgtune0	R/W	0	调整Vbus Valid的门限电压 111: +9% 110: +6% 101: +3% 100: default 011: -3% 010: -6% 001: -9% 000: -12%
0	cfg_en0	R/W	0	配置使能 0: 使用硬件默认配置 1: 使用该寄存器的软件配置

#### 5.4.14 USB PHY 配置寄存器 1

USB PHY 配置寄存器1，配置 USB 接口1的电气特性。

寄存器地址：0x14000504。

表 5- 23 USB PHY 配置寄存器 1

位域	名称	访问	缺省值	描述
31:30	Reserved	R/O	0	
29	commononnl	R/W	0	通用模块关电模式 1: 在suspend时, XO, Bias, PLL模块掉电; 在睡眠时, Bias, PLL掉电 0: 在suspend或睡眠时XO, Bias, PLL都有电
28	dmpulldownl	R/W	0	dm端口下拉电阻使能 0: D-使能 1: D-关闭
27	dppulldownl	R/W	0	dp端口下拉电阻使能 0: D+使能 1: D+关闭

位域	名称	访问	缺省值	描述
26:25	txrestunel	R/W	0	该信号调整了驱动电阻，用来补偿在发送器到电缆之间的寄生电阻 11: -4Ω 10: -2Ω 01: 默认 00: +1.5Ω
24	txpreempulsetunel	R/W	0	调整了在高速模式下预加强电流在dp, dm电缆上的持续时间，高速收发器预加强电流持续时间根据时间单位来定义 1: 1X, 短持续时间 0: 2X, 长持续时间
23:22	txpreempamtunel	R/W	0	调整了在高速模式从J到K或从K到J的跳变时提供的电流量，高速收发器预加强电流量根据电流量单位来定义 11: HS传输预加强3X电流 10: HS传输预加强2X电流 01: HS传输预加强1X电流 00: HS传输预加强电流关闭
21:20	txhsxvtunel	R/W	0	在高速模式下调整dp, dm信号交叉时的电压 11: default 10: +15mv 01: -15mv 00: reserved
19:18	txrisetunel	R/W	0	调整了高速波形上升、下降沿次数 11: -10% 10: default 01: +15% 00: +20%
17:14	txvrefstunel	R/W	0	调整高速直流电压 1111: +8.75% 1110: +7.5% 1101: +6.25% 1100: +5% 1011: +3.75% 1010: +2.5% 1001: +1.25% 1000: default 0111: -1.25% 0110: -2.5% 0101: -3.75% 0100: -5% 0011: -6.25% 0010: -7.5% 0001: -8.75% 0000: -10%
13:10	txfslstunel	R/W	0	调整了低、全速单端源阻抗 1111: -5% 0111: -2.5% 0011: default 0001: +2.5% 0000: +5%

位域	名称	访问	缺省值	描述
9:7	sqrxtunel	R/W	0	调整门限电压来检测有效的高速数据 111: -20% 110: -15% 101: -10% 100: -5% 011: default 010: +5% 001: +10% 000: +15%
6:4	compdistunel	R/W	0	调整门限电压用于检测在主控制器上的一个断开连接事件 111: +4.5% 110: +3% 101: +1.5% 100: default 011: -1.5% 010: -3% 001: -4.5% 000: -6%
3:1	otgtunel	R/W	0	调整Vbus Valid的门限电压 111: +9% 110: +6% 101: +3% 100: default 011: -3% 010: -6% 001: -9% 000: -12%
0	cfg_en1	R/W	0	配置使能 0: 使用硬件默认配置 1: 使用该寄存器的软件配置

二、打印系统通用配置寄存器，主要包括以下寄存器说明：

### 5.4.15 SYS PLL 时钟配置寄存器 0

SYS PLL 时钟配置寄存器0，用于 SYS PLL 时钟参数配置。

寄存器地址：0x15103000。

表 5- 24 SYS-PLL 时钟配置寄存器 0

位域	名称	访问	缺省值	描述
31:30	Reserved	RO	0x0	
29:24	odiv_node	RW	0x0	NODE PLL分频系数配置：0~63
23:16	div_loopc	RW	0x0	PLL倍频系数：0~255
13:8	div_refc	RW	0x0	PLL参考时钟分频系数：0~63
7	pll_locked	RO	0x0	PLL锁定标志，1代表锁定
6	Reserved	RO	0x0	-
5	pd_pll	RW	0x0	PLL关电控制，1代表关电
4	bypass	RW	0x0	PLL时钟bypass控制，1代表bypass
3	pll_soft_set	RW	0x0	允许软件设置PLL，1代表允许软件配置
2	pll_sel_soc	RW	0x0	SOC选择PLL时钟输出配置，1代表选择PLL时钟输出

位域	名称	访问	缺省值	描述
1	pll_sel_la132	RW	0x0	LA132选择PLL时钟输出配置，1代表选择PLL时钟输出
0	pll_sel_node	RW	0x0	NODE选择PLL时钟输出配置，1代表选择PLL时钟输出

#### 5.4.16 SYSPLL 时钟配置寄存器 1

SYS PLL 时钟配置寄存器1。

寄存器地址：0x15103004。

表 5- 25 SYS-PLL 时钟配置寄存器 1

位域	名称	访问	缺省值	描述
31:14	Reserved	RO	0x0	-
13:8	odiv_soc	RW	0x0	SOC PLL分频系数配置：0~63
7:6	Reserved	RO	0x0	-
5:0	odiv_la132	RW	0x0	LA132 PLL分频系数配置：0~63

#### 5.4.17 DDR PLL 时钟配置寄存器 0

DDR PLL 时钟配置寄存器0，用于 DDR PLL 时钟参数配置。

寄存器地址：0x15103008。

表 5- 26 DDR-PLL 时钟配置寄存器 0

位域	名称	访问	缺省值	描述
31:30	Reserved	RO	0x0	-
29:24	odiv_ddr	RW	0x0	DDR PLL分频系数配置：0~63
23:16	div_loopc	RW	0x0	PLL倍频系数：0~255
13:8	div_refc	RW	0x0	PLL参考时钟分频系数：0~63
7	pll_locked	RO	0x0	PLL锁定标志，1代表锁定
6	Reserved	RO	0x0	-
5	pd_pll	RW	0x0	PLL关电控制，1代表关电
4	bypass	RW	0x0	PLL时钟bypass控制，1代表bypass
3	pll_soft_set	RW	0x0	允许软件设置PLL，1代表允许软件配置
2	pll_sel_img	RW	0x0	IMAGE选择PLL时钟输出配置，1代表选择PLL时钟输出
1	pll_sel_network	RW	0x0	NETWORK选择PLL时钟输出配置，1代表选择PLL时钟输出
0	pll_sel_ddr	RW	0x0	DDR选择PLL时钟输出配置，1代表选择PLL时钟输出

#### 5.4.18 DDR PLL 时钟配置寄存器 1

DDR PLL 时钟配置寄存器1，用于 DDR PLL 时钟参数配置。

寄存器地址：0x1510300c。

表 5- 27 DDR-PLL 时钟配置寄存器 1

位域	名称	访问	缺省值	描述
31:14	Reserved	RO	0x0	-
13:8	odiv_img	RW	0x0	IMAGE PLL分频系数配置：0~63
7:6	Reserved	RO	0x0	-
5:0	odiv_network	RW	0x0	NETWORK PLL分频系数配置：0~63

### 5.4.19 VID PLL 时钟配置寄存器 0

VID PLL 时钟配置寄存器0，用于 VID PLL 时钟参数配置。

寄存器地址：0x15103010。

表 5- 28 VID-PLL 时钟配置寄存器 0

位域	名称	访问	缺省值	描述
31:30	Reserved	RO	0x0	-
29:24	odiv_prvid	RW	0x0	PRVID PLL分频系数配置：0~63
23:16	div_loopc	RW	0x0	PLL倍频系数：0~255
13:8	div_refc	RW	0x0	PLL参考时钟分频系数：0~63
7	pll_locked	RO	0x0	PLL锁定标志，1代表锁定
6	Reserved	RO	0x0	-
5	pd_pll	RW	0x0	PLL关电控制，1代表关电
4	bypass	RW	0x0	PLL时钟bypass控制，1代表bypass
3	pll_soft_set	RW	0x0	允许软件设置PLL，1代表允许软件配置
2	pll_sel_gmacbp	RW	0x0	GMAC备份时钟选择PLL时钟输出配置，1代表选择PLL时钟输出
1	pll_sel_scvid	RW	0x0	扫描数据VIDEO选择PLL时钟输出配置，1代表选择PLL时钟输出
0	pll_sel_prvid	RW	0x0	打印数据VIDEO选择PLL时钟输出配置，1代表选择PLL时钟输出

### 5.4.20 VID PLL 时钟配置寄存器 1

SYS PLL 时钟配置寄存器1，用于 VID PLL 时钟参数配置。

寄存器地址：0x15103014。

表 5- 29 VID-PLL 时钟配置寄存器 1

位域	名称	访问	缺省值	描述
31:14	Reserved	RO	0x0	-
13:8	odiv_gmacbp	RW	0x0	GMAC-BACKUP PLL分频系数配置：0~63
7:6	Reserved	RO	0x0	-
5:0	odiv_scvid	RW	0x0	SCVID PLL分频系数配置：0~63

### 5.4.21 设备时钟分频配置寄存器 0

设备时钟分频配置寄存器0，按照分频模式分为两种分频配置：

(1) freq\_mode=0，设备时钟分频计算公式为： $f_{out}=f_{in}*(freqscale[2:0]+1)/8$ ；

(2) freq\_mode=1，设备时钟分频计算公式为： $f_{out}=f_{in}/(freqscale[2:0]+1)$ 。

寄存器地址：0x15103020。

表 5- 30 设备时钟分频配置寄存器

位域	名称	访问	缺省值	描述
31:28	jpeg_freqscale	RW	0x7	jpeg_freqscale[3]:JPEG时钟分频模式freq_mode； jpeg_freqscale[2:0]:JPEG时钟输出分频系数：0~7 注：对应时钟源为DDR PLL中IMAGE分频输出时钟，分频输出时钟用于JPEG编码模块时钟
27:24	jbig_freqscale	RW	0x7	jbig_freqscale[3]:JBIG时钟分频模式freq_mode； jbig_freqscale[2:0]:JBIG时钟输出分频系数：0~7 注：对应时钟源为DDR PLL中IMAGE分频输出时钟，分频输出时钟用于JBIG解码模块时钟

位域	名称	访问	缺省值	描述
23:20	sdio_freqscale	RW	0x7	sdio_freqscale[3]:SDIO时钟分频模式freq_mode; sdio_freqscale[2:0]:SDIO时钟输出分频系数: 0~7 注: 对应时钟源为SYS_PLL中SOC分频输出时钟, 分频输出时钟用于SDIO/eMMC模块时钟
19:16	apb_freqscale	RW	0x7	apb_freqscale[3]:APB时钟分频模式freq_mode; apb_freqscale[2:0]:APB时钟输出分频系数: 0~7 注: 对应时钟源为SYS_PLL中SOC分频输出时钟, 分频输出时钟用于主系统APB各功能模块时钟
15:12	boot_freqscale	RW	0x7	boot_freqscale[3]:BOOT时钟分频模式freq_mode; boot_freqscale[2:0]:BOOT时钟输出分频系数: 0~7 注: 对应时钟源为SYS_PLL中SOC分频输出时钟, 分频输出时钟用于主系统BOOT功能模块时钟 (INT/CONF/SPIO)
11:8	gmac_freqscale	RW	0x7	gmac_freqscale[3]:GMAC时钟分频模式freq_mode; gmac_freqscale[2:0]:GMAC时钟输出分频系数: 0~7 注: 对应时钟源为SYS_PLL中SOC分频输出时钟, 分频输出时钟用于GMAC模块时钟
7:4	usb_freqscale	RW	0x7	usb_freqscale[3]:USB时钟分频模式freq_mode; usb_freqscale[2:0]:USB时钟输出分频系数: 0~7 注: 对应时钟源为SYS_PLL中SOC分频输出时钟, 分频输出时钟用于USB/OTG模块时钟
3:0	node_freqscale	RW	0x7	node_freqscale[3]:NODE时钟分频模式freq_mode; node_freqscale[2:0]:NODE时钟输出分频系数: 0~7 注: 对应时钟源为SYS_PLL中NODE分频输出时钟, 分频输出时钟用于NODE模块(LA364/SCache/IODMA)

#### 5.4.22 设备时钟分频配置寄存器 1

设备时钟分频配置寄存器1, 按照分频模式分为两种分频配置:

- (1) freq\_mode=0, 设备时钟分频计算公式为:  $f_{out}=f_{in}*(freqscale[2:0]+1)/8$ ;
- (2) freq\_mode=1, 设备时钟分频计算公式为:  $f_{out}=f_{in}/(freqscale[2:0]+1)$ 。

寄存器地址: 0x15103024。

表 5- 31 设备时钟分频配置寄存器

位域	名称	访问	缺省值	描述
31:28	aes_freqscale	RW	0x7	aes_freqscale[3]:AES时钟分频模式freq_mode; aes_freqscale[2:0]:AES时钟输出分频系数: 0~7 注: 对应时钟源为SYS_PLL中SOC分频输出时钟, 分频输出时钟用于AES模块时钟
27:24	rsmxbar_freqscale	RW	0x7	rsmxbar_freqscale[3]:RSMXBAR时钟分频模式freq_mode; rsmxbar_freqscale[2:0]:RSMXBAR时钟输出分频系数: 0~7 注: 对应时钟源为SYS_PLL中LA132分频输出时钟, 分频输出时钟用于RESUME域功能模块(GMACO/OTG)
23:20	Reserved	RO	0x7	-
19:16	prvid_freqscale	RW	0x7	prvid_freqscale[3]:PRVID时钟分频模式freq_mode; prvid_freqscale[2:0]:PRVID时钟输出分频系数: 0~7 注: 对应时钟源为VID_PLL中PRVID分频输出时钟, 分频输出时钟用于PRINTER打印功能模块时钟



位域	名称	访问	缺省值	描述
15:12	prifc_freqscale	RW	0x7	prifc_freqscale[3]:PRIFC时钟分频模式 freq_mode; prifc_freqscale[2:0]:PRIFC时钟输出分频系数: 0~7 注: 对应时钟源为VID PLL中PRVID分频输出时钟, 分频输出时钟用于PRINTER打印模块内部总线时钟
11:8	prdev_freqscale	RW	0x7	prdev_freqscale[3]:PRDEV时钟分频模式 freq_mode; prdev_freqscale[2:0]:PRDEV时钟输出分频系数: 0~7 注: 对应时钟源为SYS PLL中LA132分频输出时钟, 分频输出时钟用于打印系统内各设备模块时钟
7:4	prxbar_freqscale	RW	0x7	prxbar_freqscale[3]:PRXBAR时钟分频模式 freq_mode; prxbar_freqscale[2:0]:PRXBAR时钟输出分频系 数: 0~7 注: 对应时钟源为SYS PLL中LA132分频输出时钟, 分频输出时钟用于打印系统内部互连模块时钟
3:0	pr132_freqscale	RW	0x7	pr132_freqscale[3]:PR132时钟分频模式 freq_mode; pr132_freqscale[2:0]:PR132时钟输出分频系数: 0~7 注: 对应时钟源为SYS PLL中LA132分频输出时钟, 分频输出时钟用于打印系统LA132核时钟

### 5.4.23 设备时钟分频配置寄存器 2

设备时钟分频配置寄存器2, 按照分频模式分为两种分频配置:

- (1) freq\_mode=0, 设备时钟分频计算公式为:  $f_{out}=f_{in}*(freqscale[2:0]+1)/8$ ;
- (2) freq\_mode=1, 设备时钟分频计算公式为:  $f_{out}=f_{in}/(freqscale[2:0]+1)$ 。

寄存器地址: 0x15103028。

表 5- 32 设备时钟分频配置寄存器

位域	名称	访问	缺省值	描述
31:16	Reserved	RO	0x77	-
15:12	scifc_freqscale	RW	0x7	scifc_freqscale[3]:SCIFC时钟分频模式 freq_mode; scifc_freqscale[2:0]:SCIFC时钟输出分频系数: 0~7 注: 对应时钟源为VID PLL中SCVID分频输出时钟, 分频输出时钟用于SCANNER扫描模块时钟
11:8	scdev_freqscale	RW	0x7	scdev_freqscale[3]:SCDEV时钟分频模式 freq_mode; scdev_freqscale[2:0]:SCDEV时钟输出分频系数: 0~7 注: 对应时钟源为SYS PLL中LA132分频输出时钟, 分频输出时钟用于扫描系统内各设备模块时钟
7:4	scxbar_freqscale	RW	0x7	scxbar_freqscale[3]:SCDEV时钟分频模式 freq_mode; scxbar_freqscale[2:0]:SCDEV时钟输出分频系数: 0~7 注: 对应时钟源为SYS PLL中LA132分频输出时钟, 分频输出时钟用于扫描系统内部互连模块时钟

位域	名称	访问	缺省值	描述
3:0	sc132_freqscale	RW	0x7	sc132_freqscale[3]:SC132时钟分频模式 freq_mode; sc132_freqscale[2:0]:SC132时钟输出分频系数: 0~7 注: 对应时钟源为SYS PLL中LA132分频输出时钟, 分频输出时钟用于扫描系统LA132核时钟

#### 5.4.24 设备时钟输出使能配置寄存器

设备时钟输出使能配置寄存器,

寄存器地址: 0x1510302c。

表 5- 33 设备时钟输出使能配置寄存器

位域	名称	访问	缺省值	描述
31	gmac_clkssel		0x1	gmac模块时钟源选择配置: 1: 选择SOC-PLL输出时钟; 0: 选择GMACBP-PLL备份 输出时钟
30:24	Reserved	RO	0x7f	-
23	scifc_clken	RW	0x1	scifc模块时钟输出使能配置: 1: 时钟输出使能; 0: 时钟输出关闭
22	scdev_clken	RW	0x1	scdev模块时钟输出使能配置: 1: 时钟输出使能; 0: 时钟输出关闭
21	scxbar_clken	RW	0x1	scxbar模块时钟输出使能配置: 1: 时钟输出使能; 0: 时钟输出关闭
20	sc132_clken	RW	0x1	sc132模块时钟输出使能配置: 1: 时钟输出使能; 0: 时钟输出关闭
19:17	Reserved	RO	0x7	-
16	prvid_clken	RW	0x1	prvid模块时钟输出使能配置: 1: 时钟输出使能; 0: 时钟输出关闭
15	prifc_clken	RW	0x1	prifc模块时钟输出使能配置: 1: 时钟输出使能; 0: 时钟输出关闭
14	prdev_clken	RW	0x1	prdev模块时钟输出使能配置: 1: 时钟输出使能; 0: 时钟输出关闭
13	prxbar_clken	RW	0x1	prxbar模块时钟输出使能配置: 1: 时钟输出使能; 0: 时钟输出关闭
12	pr132_clken	RW	0x1	pr132模块时钟输出使能配置: 1: 时钟输出使能; 0: 时钟输出关闭
11	aes_clken	RW	0x1	aes模块时钟输出使能配置: 1: 时钟输出使能; 0: 时钟输出关闭
10	rsmxbar_clken	RW	0x1	rsmxbar模块时钟输出使能配置: 1: 时钟输出使能; 0: 时钟输出关闭
9	jpeg_clken	RW	0x1	jpeg模块时钟输出使能配置: 1: 时钟输出使能; 0: 时钟输出关闭
8	jbig_clken	RW	0x1	jbig模块时钟输出使能配置: 1: 时钟输出使能; 0: 时钟输出关闭
7	sdio_clken	RW	0x1	sdio模块时钟输出使能配置: 1: 时钟输出使能; 0: 时钟输出关闭
6	apb_clken	RW	0x1	apb模块时钟输出使能配置: 1: 时钟输出使能; 0: 时钟输出关闭
5	boot_clken	RW	0x1	boot模块时钟输出使能配置: 1: 时钟输出使能; 0: 时钟输出关闭
4	gmac_clken	RW	0x1	gmac模块时钟输出使能配置: 1: 时钟输出使能; 0: 时钟输出关闭
3	usb_clken	RW	0x1	usb模块时钟输出使能配置: 1: 时钟输出使能; 0: 时钟输出关闭

位域	名称	访问	缺省值	描述
2	network_clken	RW	0x1	network模块时钟输出使能配置： 1：时钟输出使能；0：时钟输出关闭
1	ddr_clken	RW	0x1	ddr模块时钟输出使能配置： 1：时钟输出使能；0：时钟输出关闭
0	node_clken	RW	0x1	node模块时钟输出使能配置： 1：时钟输出使能；0：时钟输出关闭

#### 5.4.25 打印系统设备时钟门控寄存器

打印系统各设备时钟门控寄存器，包括打印系统各功能模块时钟门控的控制。

寄存器地址：0x15103030。

表 5- 34 打印系统设备时钟门控寄存器

位域	名称	访问	缺省值	描述
31:18	Reserved	RO	0x3fff	-
17	otg_clk_ctrl	RW	0x1	RESUME域OTG模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
16	gmac0_clk_ctrl	RW	0x1	RESUME域GMACO模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
15:10	Reserved	RO	0x3f	-
9	prt_clk_ctrl	RW	0x1	打印系统Printer模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
8	hpet_clk_ctrl	RW	0x1	打印系统HPET模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
7	spi_clk_ctrl	RW	0x1	打印系统SPI模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
6	pm0io_clk_ctrl	RW	0x1	打印系统PM0IO模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
5	pwm_clk_ctrl	RW	0x1	打印系统PWM模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
4	Reserved	RO	0x1	-
3	dpm_clk_ctrl	RW	0x1	打印系统DPM模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
2	gpio_clk_ctrl	RW	0x1	打印系统GPIO模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
1	i2c_clk_ctrl	RW	0x1	打印系统I2C模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
0	uart_clk_ctrl	RW	0x1	打印系统UART模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。

#### 5.4.26 打印系统设备复位控制寄存器

打印系统各设备复位配置寄存器，包括打印系统各功能模块软件复位的控制。

寄存器地址：0x15103034。

表 5- 35 打印系统设备复位控制寄存器

位域	名称	访问	缺省值	描述
31:20	Reserved	RO	0xffff	-
19	node_rst_ctrl	RW	0x1	NODE模块复位软件配置： 1：模块复位无效；0：模块复位有效。
18	ddr_rst_ctrl	RW	0x1	DDR模块复位软件配置： 1：模块复位无效；0：模块复位有效。
17	Reserved	RO	0x1	-
16	rsm_rst_ctrl	RW	0x1	RESUME域(GMAC+OTG)模块复位软件配置： 1：模块复位无效；0：模块复位有效。

位域	名称	访问	缺省值	描述
15:10	Reserved	RO	0x3f	-
9	prt_rst_ctrl	RW	0x1	打印系统Printer模块复位软件配置： 1：模块复位无效；0：模块复位有效。
8	hpet_rst_ctrl	RW	0x1	打印系统HPET模块复位软件配置： 1：模块复位无效；0：模块复位有效。
7	spi_rst_ctrl	RW	0x1	打印系统SPI模块复位软件配置： 1：模块复位无效；0：模块复位有效。
6	pm0io_rst_ctrl	RW	0x1	打印系统PM0IO模块复位软件配置： 1：模块复位无效；0：模块复位有效。
5	pwm_rst_ctrl	RW	0x1	打印系统PWM模块复位软件配置： 1：模块复位无效；0：模块复位有效。
4	Reserved	RO	0x1	-
3	dpm_rst_ctrl	RW	0x1	打印系统DPM模块复位软件配置： 1：模块复位无效；0：模块复位有效。
2	gpio_rst_ctrl	RW	0x1	打印系统GPIO模块复位软件配置： 1：模块复位无效；0：模块复位有效。
1	i2c_rst_ctrl	RW	0x1	打印系统I2C模块复位软件配置： 1：模块复位无效；0：模块复位有效。
0	uart_rst_ctrl	RW	0x1	打印系统UART模块复位软件配置： 1：模块复位无效；0：模块复位有效。

#### 5.4.27 打印系统 GPIO0~15 复用配置寄存器

打印系统 GPIO0~15复用配置寄存器。

寄存器地址：0x15103040。

表 5- 36 打印系统 GPIO0~15 复用配置寄存器

位域	名称	访问	缺省值	描述
31:30	GPIO15_MUX	RW	0x0	GPIO15引脚复用配置： 00：复用为GPIO15；01：第一复用； 10：第二复用； 11：引脚主功能。
29:28	GPIO14_MUX	RW	0x0	GPIO14引脚复用配置： 00：复用为GPIO14；01：第一复用； 10：第二复用； 11：引脚主功能。
27:26	GPIO13_MUX	RW	0x0	GPIO13引脚复用配置： 00：复用为GPIO13；01：第一复用； 10：第二复用； 11：引脚主功能。
25:24	GPIO12_MUX	RW	0x0	GPIO12引脚复用配置： 00：复用为GPIO12；01：第一复用； 10：第二复用； 11：引脚主功能。
23:22	GPIO11_MUX	RW	0x0	GPIO11引脚复用配置： 00：复用为GPIO12；01：第一复用； 10：第二复用； 11：引脚主功能。
21:20	GPIO10_MUX	RW	0x0	GPIO10引脚复用配置： 00：复用为GPIO10；01：第一复用； 10：第二复用； 11：引脚主功能。
19:18	GPIO09_MUX	RW	0x0	GPIO09引脚复用配置： 00：复用为GPIO09；01：第一复用； 10：第二复用； 11：引脚主功能。
17:16	GPIO08_MUX	RW	0x0	GPIO08引脚复用配置： 00：复用为GPIO08；01：第一复用； 10：第二复用； 11：引脚主功能。
15:14	GPIO07_MUX	RW	0x0	GPIO07引脚复用配置： 00：复用为GPIO07；01：第一复用； 10：第二复用； 11：引脚主功能。

位域	名称	访问	缺省值	描述
13:12	GPI006_MUX	RW	0x0	GPI006引脚复用配置： 00：复用为GPI006；01：第一复用； 10：第二复用； 11：引脚主功能。
11:10	GPI005_MUX	RW	0x0	GPI005引脚复用配置： 00：复用为GPI005；01：第一复用； 10：第二复用； 11：引脚主功能。
9:8	GPI004_MUX	RW	0x0	GPI004引脚复用配置： 00：复用为GPI004；01：第一复用； 10：第二复用； 11：引脚主功能。
7:6	GPI003_MUX	RW	0x0	GPI003引脚复用配置： 00：复用为GPI003；01：第一复用； 10：第二复用； 11：引脚主功能。
5:4	GPI002_MUX	RW	0x0	GPI002引脚复用配置： 00：复用为GPI002；01：第一复用； 10：第二复用； 11：引脚主功能。
3:2	GPI001_MUX	RW	0x0	GPI001引脚复用配置： 00：复用为GPI001；01：第一复用； 10：第二复用； 11：引脚主功能。
1:0	GPI000_MUX	RW	0x0	GPI000引脚复用配置： 00：复用为GPI000；01：第一复用； 10：第二复用； 11：引脚主功能。

#### 5.4.28 打印系统 GPI016~31 复用配置寄存器

打印系统 GPI016~31复用配置寄存器。

寄存器地址：0x15103044。

表 5-37 打印系统 GPI016~31 复用配置寄存器

位域	名称	访问	缺省值	描述
31:30	GPI031_MUX	RW	0x0	GPI031引脚复用配置： 00：复用为GPI031；01：第一复用； 10：第二复用； 11：引脚主功能。
29:28	GPI030_MUX	RW	0x0	GPI030引脚复用配置： 00：复用为GPI030；01：第一复用； 10：第二复用； 11：引脚主功能。
27:26	GPI029_MUX	RW	0x0	GPI029引脚复用配置： 00：复用为GPI029；01：第一复用； 10：第二复用； 11：引脚主功能。
25:24	GPI028_MUX	RW	0x0	GPI028引脚复用配置： 00：复用为GPI028；01：第一复用； 10：第二复用； 11：引脚主功能。
23:22	GPI027_MUX	RW	0x0	GPI027引脚复用配置： 00：复用为GPI027；01：第一复用； 10：第二复用； 11：引脚主功能。
21:20	GPI026_MUX	RW	0x0	GPI026引脚复用配置： 00：复用为GPI026；01：第一复用； 10：第二复用； 11：引脚主功能。
19:18	GPI025_MUX	RW	0x0	GPI025引脚复用配置： 00：复用为GPI025；01：第一复用； 10：第二复用； 11：引脚主功能。
17:16	GPI024_MUX	RW	0x0	GPI024引脚复用配置： 00：复用为GPI024；01：第一复用； 10：第二复用； 11：引脚主功能。
15:14	GPI023_MUX	RW	0x0	GPI023引脚复用配置： 00：复用为GPI023；01：第一复用； 10：第二复用； 11：引脚主功能。

位域	名称	访问	缺省值	描述
13:12	GPI022_MUX	RW	0x0	GPI022引脚复用配置： 00：复用为GPI022；01：第一复用； 10：第二复用； 11：引脚主功能。
11:10	GPI021_MUX	RW	0x0	GPI021引脚复用配置： 00：复用为GPI021；01：第一复用； 10：第二复用； 11：引脚主功能。
9:8	GPI020_MUX	RW	0x0	GPI020引脚复用配置： 00：复用为GPI020；01：第一复用； 10：第二复用； 11：引脚主功能。
7:6	GPI019_MUX	RW	0x0	GPI019引脚复用配置： 00：复用为GPI019；01：第一复用； 10：第二复用； 11：引脚主功能。
5:4	GPI018_MUX	RW	0x0	GPI018引脚复用配置： 00：复用为GPI018；01：第一复用； 10：第二复用； 11：引脚主功能。
3:2	GPI017_MUX	RW	0x0	GPI017引脚复用配置： 00：复用为GPI017；01：第一复用； 10：第二复用； 11：引脚主功能。
1:0	GPI016_MUX	RW	0x0	GPI016引脚复用配置： 00：复用为GPI016；01：第一复用； 10：第二复用； 11：引脚主功能。

#### 5.4.29 打印系统 GPI032~47 复用配置寄存器

打印系统 GPI032~47复用配置寄存器。

寄存器地址：0x15103048。

表 5-38 打印系统 GPI032~47 复用配置寄存器

位域	名称	访问	缺省值	描述
31:30	GPI047_MUX	RW	0x0	GPI047引脚复用配置： 00：复用为GPI047；01：第一复用； 10：第二复用； 11：引脚主功能。
29:28	GPI046_MUX	RW	0x0	GPI046引脚复用配置： 00：复用为GPI046；01：第一复用； 10：第二复用； 11：引脚主功能。
27:26	GPI045_MUX	RW	0x0	GPI045引脚复用配置： 00：复用为GPI045；01：第一复用； 10：第二复用； 11：引脚主功能。
25:24	GPI044_MUX	RW	0x0	GPI044引脚复用配置： 00：复用为GPI044；01：第一复用； 10：第二复用； 11：引脚主功能。
23:22	GPI043_MUX	RW	0x0	GPI043引脚复用配置： 00：复用为GPI043；01：第一复用； 10：第二复用； 11：引脚主功能。
21:20	GPI042_MUX	RW	0x0	GPI042引脚复用配置： 00：复用为GPI042；01：第一复用； 10：第二复用； 11：引脚主功能。
19:18	GPI041_MUX	RW	0x0	GPI041引脚复用配置： 00：复用为GPI041；01：第一复用； 10：第二复用； 11：引脚主功能。
17:16	GPI040_MUX	RW	0x0	GPI040引脚复用配置： 00：复用为GPI040；01：第一复用； 10：第二复用； 11：引脚主功能。
15:14	GPI039_MUX	RW	0x0	GPI039引脚复用配置： 00：复用为GPI039；01：第一复用； 10：第二复用； 11：引脚主功能。

位域	名称	访问	缺省值	描述
13:12	GPI038_MUX	RW	0x0	GPI038引脚复用配置： 00：复用为GPI038；01：第一复用； 10：第二复用； 11：引脚主功能。
11:10	GPI037_MUX	RW	0x0	GPI037引脚复用配置： 00：复用为GPI037；01：第一复用； 10：第二复用； 11：引脚主功能。
9:8	GPI036_MUX	RW	0x0	GPI036引脚复用配置： 00：复用为GPI036；01：第一复用； 10：第二复用； 11：引脚主功能。
7:6	GPI035_MUX	RW	0x0	GPI035引脚复用配置： 00：复用为GPI035；01：第一复用； 10：第二复用； 11：引脚主功能。
5:4	GPI034_MUX	RW	0x0	GPI034引脚复用配置： 00：复用为GPI034；01：第一复用； 10：第二复用； 11：引脚主功能。
3:2	GPI033_MUX	RW	0x0	GPI033引脚复用配置： 00：复用为GPI033；01：第一复用； 10：第二复用； 11：引脚主功能。
1:0	GPI032_MUX	RW	0x0	GPI032引脚复用配置： 00：复用为GPI032；01：第一复用； 10：第二复用； 11：引脚主功能。

### 5.4.30 打印系统 GPI048~57 复用配置寄存器

打印系统 GPI048~57复用配置寄存器。

寄存器地址：0x1510304c。

表 5-39 打印系统 GPI048~57 复用配置寄存器

位域	名称	访问	缺省值	描述
31:20	Reserved	RO	0x0	-
19:18	GPI057_MUX	RW	0x0	GPI057引脚复用配置： 00：复用为GPI057；01：第一复用； 10：第二复用； 11：引脚主功能。
17:16	GPI056_MUX	RW	0x0	GPI056引脚复用配置： 00：复用为GPI056；01：第一复用； 10：第二复用； 11：引脚主功能。
15:14	GPI055_MUX	RW	0x0	GPI055引脚复用配置： 00：复用为GPI055；01：第一复用； 10：第二复用； 11：引脚主功能。
13:12	GPI054_MUX	RW	0x0	GPI054引脚复用配置： 00：复用为GPI054；01：第一复用； 10：第二复用； 11：引脚主功能。
11:10	GPI053_MUX	RW	0x0	GPI053引脚复用配置： 00：复用为GPI053；01：第一复用； 10：第二复用； 11：引脚主功能。
9:8	GPI052_MUX	RW	0x0	GPI052引脚复用配置： 00：复用为GPI052；01：第一复用； 10：第二复用； 11：引脚主功能。
7:6	GPI051_MUX	RW	0x0	GPI051引脚复用配置： 00：复用为GPI051；01：第一复用； 10：第二复用； 11：引脚主功能。
5:4	GPI050_MUX	RW	0x0	GPI050引脚复用配置： 00：复用为GPI049；01：第一复用； 10：第二复用； 11：引脚主功能。



位域	名称	访问	缺省值	描述
3:2	GPI049_MUX	RW	0x0	GPI049引脚复用配置： 00：复用为GPI048；01：第一复用； 10：第二复用； 11：引脚主功能。
1:0	GPI048_MUX	RW	0x0	GPI048引脚复用配置： 00：复用为GPI047；01：第一复用； 10：第二复用； 11：引脚主功能。

### 5.4.31 打印系统通用配置寄存器 0

打印系统通用配置寄存器0，包括打印模块、PM0IO、RTC 等参数配置。

寄存器地址：0x15103070。

表 5- 40 通用配置寄存器 0

位域	名称	访问	缺省值	描述
31:10	Reserved	RO	0x0	-
9	pm0io_extin_sel	RW	0x0	PM0IO模块EXTIN[31:0]信号输入选择： 1：由PM0IO_INTCON[31:0]端口输入； 0：由PM0IO_INTCON[31:24]、VIDCLK_OUT[7:0]、 PRT_OVLIN[15:0]依次输入
8	printer_extttimin_sel	RW	0x0	打印模块16路外部TIMING信号输入选择： 1：由PM0IO[31:16]内部输出端口信号输入； 0：由PM0IO[15:00]内部输出端口信号输入
7:6	Reserved	RO	0x0	-
5	pr132_nmi	RW	0x0	打印系统处理器核NMI输入信号，高电平有效
4	hpet_int_ctrl	RW	0x0	hpet中断输出模式配置： 0：单中断输出模式(内部3个计数器共享1个中断)； 1：三中断输出模式(内部3个计数器独立3个中断)
3:1	Reserved	RO	0x0	-
0	conf_rtc_timer_hspeed	RW	0x0	RTC计数器值快速访问使能配置位： 1：开启快速访问；0：关闭快速访问。

### 5.4.32 打印系统通用配置寄存器 1

打印系统通用配置寄存器1，包括对 USB、GMAC 等 Cache 一致性配置等。

寄存器地址：0x15103074。

表 5- 41 通用配置寄存器 1

位域	名称	访问	缺省值	描述
31:16	Reserved	RO	0x0	-
15:8	prvidout_lvds_tm	RW	0x0	8路打印数据LVDS差分端口输出模式配置： 1：数据bypass输出模式； 0：数据normal输出模式(时钟采样后输出)
7:0	prvidout_lvds_pd	RW	0xff	8路打印数据LVDS差分端口PD配置： 1：powerdown，差分端口不工作，TTL模式输出； 0：poweron，差分端口正常工作，LVDS模式输出

### 5.4.33 打印系统采样参数寄存器

打印系统采样参数寄存器，包括打印系统启动配置及其他系统休眠相关信息。

寄存器地址：0x15103078。



表 5- 42 芯片采样参数寄存器

位域	名称	访问	缺省值	描述
31:12	Reserved	RO	0x0	-
11	la364_pwroff_sta	RO	0x0	主系统处理器核掉电状态监测： 0=POWERON状态 1=POWEROFF状态
10	la364_sleeping_sta	RO	0x0	主系统处理器核休眠状态监测： 0=WORKON状态 1=SLEEPING状态
9	sc132_sleeping_sta	RO	0x0	扫描系统处理器核休眠状态监测： 0=WORKON状态 1=SLEEPING状态
8	pr132_sleeping_sta	RO	0x0	打印系统处理器核休眠状态监测： 0=WORKON状态 1=SLEEPING状态
7	Reserved	RO	0x0	-
6:5	otg_clkmode	RO	0x0	OTG参考时钟模式输入 00=外部差分晶体输入 (XI/XO) 01/10=外部单端晶振输入 (XO) 11=内部参考时钟输入
4	sdio1_mode	RO	0x0	SDIO1模式配置输入 0=SDIO模式 1=EMMC模式
3	sdio0_mode	RO	0x0	SDIO0模式配置输入 0=SDIO模式 1=EMMC模式
2:1	clk_sel	RO	0x0	芯片内部PLL输出时钟上电配置选择： 00：硬件低频时钟配置模式，PLL按照低频配置参数输出时钟 (NODE:500M, DDR:480M, NET:320M)； 01：硬件高频时钟配置模式，PLL按照高频配置参数输出时钟 (NODE:725M, DDR:600M, NET:400M)； 10：软件配置模式，PLL按照软件配置选择输出时钟； 11：硬件bypass模式，PLL输出时钟全部使用外部输入系统时钟。
0	boot_sel	RO	0x0	芯片启动选择方式： 0：SPI0启动；1：SDIO0/eMMC0启动。

#### 5.4.34 打印系统高精度计数器

打印系统计数寄存器，32位高精度时钟计数器，工作频率为内部总线时钟频率。

寄存器地址：0x1510307c。

表 5- 43 打印系统高精度计数器

位域	名称	访问	缺省值	描述
31:0	PRCHIP_HPT	RW	0x0	32位高精度时钟计数器

#### 5.4.35 打印系统设备 DMA 路由基址配置寄存器

打印系统设备 DMA 路由基址配置寄存器，配置共享打印系统 SRAM、打印设备 DMA 基址路由。

寄存器地址：0x151030e0。

表 5- 44 打印系统设备 DMA 路由基址配置寄存器

位域	名称	访问	缺省值	描述
31:16	prdma_base_addr	RW	0x0	打印系统设备DMA访问共享存储基址配置： prdma_base_addr[0]：共享使能位： 0-不可共享打印系统SRAM(128KB)，仅可访问主系统内存； 1- 可共享打印系统SRAM(128KB)，通过地址路由访问主系统内存、打印系统SRAM。 prdma_base_addr[15:1]：共享打印系统SRAM存储基址，对应设备DMA共享访问地址[31:17]路由。
15:0	prsrsm_base_addr	RW	0x1500	其他系统共享访问打印系统SRAM访问基址配置，对应其他系统共享访问地址[31:16]路由

### 5.4.36 打印系统共享设备通用配置寄存器 0

打印系统共享设备通用配置寄存器0，包括 OTG、GMAC、JTAG 等参数配置。

寄存器地址：0x151030f0。

表 5- 45 打印系统共享设备通用配置寄存器 0

位域	名称	访问	缺省值	描述
31:13	Reserved	RO	0x0	-
12	la364_ram_cg	RW	0x0	主系统处理器核RAM时钟门控使能，高电平有效
11:10	jtag_mode_config	RW	0x0	JTAG功能调试接口复用模式配置： 00：LA364处理器JTAG调试接口； 01：PR132处理器JTAG调试接口； 10：SC132处理器JTAG调试接口； 11：LA364->PR132->SC132串行JTAG。
9	gmac_test_lpbk	RW	0x0	GMACO~1接口loopback环回测试模式使能： 1：使能loopback环回测试模式； 0：关闭loopback环回测试模式。
8	gmac0_mii_sel	RW	0x0	GMACO接口MII模式选择： 1：MII接口模式；0：RGMII接口模式
7:4	conf_otg_flush_idle	RW	0xf	设置OTG清空write buffer前空闲周期数
3	conf_otg_prefetch	RW	0x0	OTG接口总线使能读预取
2	conf_otg_flush_wr	RW	0x0	OTG接口总线设置写命令发出后是否清空read buffer
1	conf_otg_stop_waw	RW	0x0	OTG接口总线是否允许在上一个写完成前发出写命令
0	conf_otg_stop_raw	RW	0x1	OTG接口总线是否允许在上一个写完成前发出读命令

### 5.4.37 打印系统共享设备通用配置寄存器 1

打印系统共享设备通用配置寄存器1，配置 RESUME 中 GMAC、OTG 设备 DMA 基址路由。

寄存器地址：0x151030f4。

表 5- 46 打印系统共享设备通用配置寄存器 1

位域	名称	访问	缺省值	描述
31:16	gmacdma_base_addr	RW	0x0	GMACO设备DMA访问共享存储基址配置： gmacdma_base_addr[0]：共享使能位： 0-不可共享打印系统SRAM(128KB)，仅可访问主系统内存； 1- 可共享打印系统SRAM(128KB)，通过地址路由访问主系统内存、打印系统SRAM。 gmacdma_base_addr[15:1]：共享打印系统SRAM存储基址，对应设备DMA共享访问地址[31:17]路由。

位域	名称	访问	缺省值	描述
15:0	otgdma_base_addr	RW	0x1500	OTG设备DMA访问共享存储基址配置： otgdma_base_addr[0]：共享使能位： 0-不可共享打印系统SRAM(128KB)，仅可访问主系统内存； 2- 可共享打印系统SRAM(128KB)，通过地址路由访问主系统内存、打印系统SRAM。 otgdma_base_addr[15:1]：共享打印系统SRAM存储基址，对应设备DMA共享访问地址[31:17]路由。

#### 5.4.38 打印系统共享设备通用配置寄存器 2

打印系统共享设备通用配置寄存器2，配置 OTG 接口的电气特性。

寄存器地址：0x151030f8。

表 5- 47 打印系统共享设备通用配置寄存器 2

位域	名称	访问	缺省值	描述
31	Reserved	R/O	0	
30	cfg_fs_data_mod	R/W	0	低速模式数据有效使能控制 0: 低速模式下，数据信号不受使能信号控制其有效性 1: 低速模式下，数据信号受到使能信号控制其有效性
29	commononn0	R/W	0	通用模块关电模式 1: 在suspend时，X0, Bias, PLL模块掉电；在睡眠时，Bias, PLL掉电 0: 在suspend或睡眠时X0, Bias, PLL都有电
28	dmpulldown0	R/W	0	dm端口下拉电阻使能 1: D-使能 0: D-关闭
27	dppulldown0	R/W	0	dp端口下拉电阻使能 1: D+使能 0: D+关闭
26:25	txrestune0	R/W	0	该信号调整了驱动电阻，用来补偿在发送器到电缆之间的寄生电阻 11: -4Ω 10: -2Ω 01: 默认 00: +1.5Ω
24	txpreempulsetune0	R/W	0	调整了在高速模式下预加强电流在dp, dm电缆上的持续时间，高速收发器预加强电流持续时间根据时间单位来定义 1: 1X, 短持续时间 0: 2X, 长持续时间
23:22	txpreempamptune0	R/W	0	调整了在高速模式从J到K或从K到J的跳变时提供的电流量，高速收发器预加强电流量根据电流量单位来定义 11: HS传输预加强3X电流 10: HS传输预加强2X电流 01: HS传输预加强1X电流 00: HS传输预加强电流关闭

位域	名称	访问	缺省值	描述
21:20	txhsxvtune0	R/W	0	在高速模式下调整dp, dm信号交叉时的电压 11: default 10: +15mv 01: -15mv 00: reserved
19:18	txrisetune0	R/W	0	调整了高速波形上升、下降沿次数 11: -10% 10: default 01: +15% 00: +20%
17:14	txvref tune0	R/W	0	调整高速直流电压 1111: +8.75% 1110: +7.5% 1101: +6.25% 1100: +5% 1011: +3.75% 1010: +2.5% 1001: +1.25% 1000: default 0111: -1.25% 0110: -2.5% 0101: -3.75% 0100: -5% 0011: -6.25% 0010: -7.5% 0001: -8.75% 0000: -10%
13:10	txfslstune0	R/W	0	调整了低、全速单端源阻抗 1111: -5% 0111: -2.5% 0011: default 0001: +2.5% 0000: +5%
9:7	sqrxtune0	R/W	0	调整门限电压来检测有效的高速数据 111: -20% 110: -15% 101: -10% 100: -5% 011: default 010: +5% 001: +10% 000: +15%
6:4	compdistune0	R/W	0	调整门限电压用于检测在主控制器上的一个断开连接事件 111: +4.5% 110: +3% 101: +1.5% 100: default 011: -1.5% 010: -3% 001: -4.5% 000: -6%

位域	名称	访问	缺省值	描述
3:1	otgtune0	R/W	0	调整Vbus Valid的门限电压 111: +9% 110: +6% 101: +3% 100: default 011: -3% 010: -6% 001: -9% 000: -12%
0	cfg_en0	R/W	0	配置使能 0: 使用硬件默认配置 1: 使用该寄存器的软件配置

#### 5.4.39 打印系统启动入口配置寄存器

打印系统启动入口配置寄存器，配置打印系统处理器核启动地址。

寄存器地址：0x15103100。

表 5- 48 打印系统启动入口配置寄存器

位域	名称	访问	缺省值	描述
31:0	prl32_boot_entry	RW	0x1c000000	配置打印系统处理器核启动入口地址，bit[0]为地址配置有效使能位，高电平有效： 1-入口地址配置有效；0-入口地址配置无效，采用默认启动入口地址(0x1c000000)

#### 5.4.40 打印系统 0 号 MAILBOX 中断状态寄存器

打印系统0号 MAILBOX 中断状态寄存器，记录中断触发状态。

寄存器地址：0x15103800。

表 5- 49 打印系统 0 号 MAILBOX 中断状态寄存器

位域	名称	访问	缺省值	描述
31:0	mlb0_ipi_status	RO	0x0	0号Mailbox中断状态寄存器，记录被触发中断状态位，任何一位被置1且对应中断使能情况下，路由至对应处理器核中断位将被置位

#### 5.4.41 打印系统 0 号 MAILBOX 中断使能寄存器

打印系统0号 MAILBOX 中断使能寄存器，使能相应中断位。

寄存器地址：0x15103804。

表 5- 50 打印系统 0 号 MAILBOX 中断使能寄存器

位域	名称	访问	缺省值	描述
31:0	mlb0_ipi_enable	RW	0x0	0号Mailbox中断使能寄存器，使能对应中断位，高电平有效

#### 5.4.42 打印系统 0 号 MAILBOX 中断触发寄存器

打印系统0号 MAILBOX 中断触发寄存器，触发相应中断位。

寄存器地址：0x15103808。

表 5- 51 打印系统 0 号 MAILBOX 中断触发寄存器

位域	名称	访问	缺省值	描述
31:0	mlb0_ipi_set	W	-	0号Mailbox中断触发寄存器，触发对应中断位，高电平有效。当相应中断使能位有效时，该位置1将触发相应中断，并将路由至对应处理器核中断位置位

#### 5.4.43 打印系统 0 号 MAILBOX 中断清除寄存器

打印系统0号 MAILBOX 中断清除寄存器，清除相应中断状态位。

寄存器地址：0x1510380c。

表 5- 52 打印系统 0 号 MAILBOX 中断清除寄存器

位域	名称	访问	缺省值	描述
31:0	mlb0_ipi_clear	W	-	0号Mailbox中断清除寄存器，清除对应中断位，写1清除。对应位写1将清除相应中断状态位，取消相应中断触发，中断使能位不做清除

#### 5.4.44 打印系统 0 号 MAILBOX 消息寄存器 0~7

打印系统0号 MAILBOX 消息寄存器0~7，传递记录相应消息，可由用户自由定义。

寄存器地址：0x15103810~0x1510382c。

表 5- 53 打印系统 0 号 MAILBOX 消息寄存器 0~7

位域	名称	访问	缺省值	描述
31:0	mlb0_message7	RW	0x0	0号Mailbox消息寄存器7，记录相应消息定义，可进行不同系统间消息传递
31:0	mlb0_message6	RW	0x0	0号Mailbox消息寄存器6，记录相应消息定义，可进行不同系统间消息传递
31:0	mlb0_message5	RW	0x0	0号Mailbox消息寄存器5，记录相应消息定义，可进行不同系统间消息传递
31:0	mlb0_message4	RW	0x0	0号Mailbox消息寄存器4，记录相应消息定义，可进行不同系统间消息传递
31:0	mlb0_message3	RW	0x0	0号Mailbox消息寄存器3，记录相应消息定义，可进行不同系统间消息传递
31:0	mlb0_message2	RW	0x0	0号Mailbox消息寄存器2，记录相应消息定义，可进行不同系统间消息传递
31:0	mlb0_message1	RW	0x0	0号Mailbox消息寄存器1，记录相应消息定义，可进行不同系统间消息传递
31:0	mlb0_message0	RW	0x0	0号Mailbox消息寄存器0，记录相应消息定义，可进行不同系统间消息传递

#### 5.4.45 打印系统 1 号 MAILBOX 中断状态寄存器

打印系统1号 MAILBOX 中断状态寄存器，记录中断触发状态。

寄存器地址：0x15103830。

表 5- 54 打印系统 1 号 MAILBOX 中断状态寄存器

位域	名称	访问	缺省值	描述
31:0	mlb1_ipi_status	RO	0x0	1号Mailbox中断状态寄存器，记录被触发中断状态位，任何一位被置1且对应中断使能情况下，路由至对应处理器核中断位将被置位

#### 5.4.46 打印系统 1 号 MAILBOX 中断使能寄存器

打印系统1号 MAILBOX 中断使能寄存器，使能相应中断位。

寄存器地址：0x15103834。

表 5- 55 打印系统 1 号 MAILBOX 中断使能寄存器

位域	名称	访问	缺省值	描述
31:0	mlbl_ipi_enable	RW	0x0	1号Mailbox中断使能寄存器，使能对应中断位，高电平有效

#### 5.4.47 打印系统 1 号 MAILBOX 中断触发寄存器

打印系统1号 MAILBOX 中断触发寄存器，触发相应中断位。

寄存器地址：0x15103838。

表 5- 56 打印系统 1 号 MAILBOX 中断触发寄存器

位域	名称	访问	缺省值	描述
31:0	mlbl_ipi_set	W	-	1号Mailbox中断触发寄存器，触发对应中断位，高电平有效。当相应中断使能位有效时，该位置1将触发相应中断，并将路由至对应处理器核中断位置

#### 5.4.48 打印系统 1 号 MAILBOX 中断清除寄存器

打印系统1号 MAILBOX 中断清除寄存器，清除相应中断状态位。

寄存器地址：0x1510383c。

表 5- 57 打印系统 1 号 MAILBOX 中断清除寄存器

位域	名称	访问	缺省值	描述
31:0	mlbl_ipi_clear	W	-	1号Mailbox中断清除寄存器，清除对应中断位，写1清除。对应位写1将清除相应中断状态位，取消相应中断触发，中断使能位不做清除

#### 5.4.49 打印系统 1 号 MAILBOX 消息寄存器 0~7

打印系统1号 MAILBOX 消息寄存器0~7，传递记录相应消息，可由用户自由定义。

寄存器地址：0x15103840~0x1510385c。

表 5- 58 打印系统 1 号 MAILBOX 消息寄存器 0~7

位域	名称	访问	缺省值	描述
31:0	mlbl_message7	RW	0x0	1号Mailbox消息寄存器7，记录相应消息定义，可进行不同系统间消息传递
31:0	mlbl_message6	RW	0x0	1号Mailbox消息寄存器6，记录相应消息定义，可进行不同系统间消息传递
31:0	mlbl_message5	RW	0x0	1号Mailbox消息寄存器5，记录相应消息定义，可进行不同系统间消息传递
31:0	mlbl_message4	RW	0x0	1号Mailbox消息寄存器4，记录相应消息定义，可进行不同系统间消息传递
31:0	mlbl_message3	RW	0x0	1号Mailbox消息寄存器3，记录相应消息定义，可进行不同系统间消息传递
31:0	mlbl_message2	RW	0x0	1号Mailbox消息寄存器2，记录相应消息定义，可进行不同系统间消息传递



位域	名称	访问	缺省值	描述
31:0	mlb1_message1	RW	0x0	1号Mailbox消息寄存器1, 记录相应消息定义, 可进行不同系统间消息传递
31:0	mlb1_message0	RW	0x0	1号Mailbox消息寄存器0, 记录相应消息定义, 可进行不同系统间消息传递

#### 5.4.50 打印系统 2 号 MAILBOX 中断状态寄存器

打印系统2号 MAILBOX 中断状态寄存器, 记录中断触发状态。

寄存器地址: 0x15103860。

表 5- 59 打印系统 2 号 MAILBOX 中断状态寄存器

位域	名称	访问	缺省值	描述
31:0	mlb2_ipi_status	RO	0x0	2号Mailbox中断状态寄存器, 记录被触发中断状态位, 任何一位被置1且对应中断使能情况下, 路由至对应处理器核中断位将被置位

#### 5.4.51 打印系统 2 号 MAILBOX 中断使能寄存器

打印系统2号 MAILBOX 中断使能寄存器, 使能相应中断位。

寄存器地址: 0x15103864。

表 5- 60 打印系统 2 号 MAILBOX 中断使能寄存器

位域	名称	访问	缺省值	描述
31:0	mlb2_ipi_enable	RW	0x0	2号Mailbox中断使能寄存器, 使能对应中断位, 高电平有效

#### 5.4.52 打印系统 2 号 MAILBOX 中断触发寄存器

打印系统2号 MAILBOX 中断触发寄存器, 触发相应中断位。

寄存器地址: 0x15103868。

表 5- 61 打印系统 2 号 MAILBOX 中断触发寄存器

位域	名称	访问	缺省值	描述
31:0	mlb2_ipi_set	W	-	2号Mailbox中断触发寄存器, 触发对应中断位, 高电平有效。当相应中断使能位有效时, 该位置1将触发相应中断, 并将路由至对应处理器核中断位位置

#### 5.4.53 打印系统 2 号 MAILBOX 中断清除寄存器

打印系统2号 MAILBOX 中断清除寄存器, 清除相应中断状态位。

寄存器地址: 0x1510386c。

表 5- 62 打印系统 2 号 MAILBOX 中断清除寄存器

位域	名称	访问	缺省值	描述
31:0	mlb2_ipi_clear	W	-	2号Mailbox中断清除寄存器, 清除对应中断位, 写1清除。对应位写1将清除相应中断状态位, 取消相应中断触发, 中断使能位不做清除

#### 5.4.54 打印系统 2 号 MAILBOX 消息寄存器 0~7

打印系统2号 MAILBOX 消息寄存器0~7, 传递记录相应消息, 可由用户自由定义。



寄存器地址：0x15103870~0x1510388c。

表 5- 63 打印系统 0 号 MAILBOX 消息寄存器 0~7

位域	名称	访问	缺省值	描述
31:0	mlb2_message7	RW	0x0	2号Mailbox消息寄存器7, 记录相应消息定义, 可进行不同系统间消息传递
31:0	mlb2_message6	RW	0x0	2号Mailbox消息寄存器6, 记录相应消息定义, 可进行不同系统间消息传递
31:0	mlb2_message5	RW	0x0	2号Mailbox消息寄存器5, 记录相应消息定义, 可进行不同系统间消息传递
31:0	mlb2_message4	RW	0x0	2号Mailbox消息寄存器4, 记录相应消息定义, 可进行不同系统间消息传递
31:0	mlb2_message3	RW	0x0	2号Mailbox消息寄存器3, 记录相应消息定义, 可进行不同系统间消息传递
31:0	mlb2_message2	RW	0x0	2号Mailbox消息寄存器2, 记录相应消息定义, 可进行不同系统间消息传递
31:0	mlb2_message1	RW	0x0	2号Mailbox消息寄存器1, 记录相应消息定义, 可进行不同系统间消息传递
31:0	mlb2_message0	RW	0x0	2号Mailbox消息寄存器0, 记录相应消息定义, 可进行不同系统间消息传递

#### 5.4.55 打印系统 MAILBOX 中断路由寄存器

打印系统 MAILBOX 中断路由寄存器, 配置0~2号 MAILBOX 中断路由。

寄存器地址：0x15103890。

表 5- 64 打印系统 MAILBOX 中断路由寄存器

位域	名称	访问	缺省值	描述
31:11	Reserved	RO	0x0	-
10:8	mlb2_ipi_map	RW	0x0	2号Mailbox中断路由配置: 001:2号Mailbox触发中断路由至主系统中断; 010:2号Mailbox触发中断路由至打印系统中断; 100:2号Mailbox触发中断路由至扫系统中断
7	Reserved	RO	0x0	-
6:4	mlb1_ipi_map	RW	0x0	1号Mailbox中断路由配置: 001:1号Mailbox触发中断路由至主系统中断; 010:1号Mailbox触发中断路由至打印系统中断; 100:1号Mailbox触发中断路由至扫系统中断
3	Reserved	RO	0x0	-
2:0	mlb0_ipi_map	RW	0x0	0号Mailbox中断路由配置: 001:0号Mailbox触发中断路由至主系统中断; 010:0号Mailbox触发中断路由至打印系统中断; 100:0号Mailbox触发中断路由至扫系统中断

三、扫描系统通用配置寄存器, 主要包括以下寄存器说明:

#### 5.4.56 扫描系统设备时钟门控寄存器

扫描系统各设备时钟门控寄存器, 包括扫描系统各功能模块时钟门控的控制。

寄存器地址：0x15202030。

表 5- 65 扫描系统设备时钟门控寄存器

位域	名称	访问	缺省值	描述
31:9	Reserved	RO	0x7fffff	-
8	sc132_clk_ctrl	RW	0x1	扫描系统处理器核时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
7:6	Reserved	RO	0x3	-
5	scvid_clk_ctrl	RW	0x1	扫描系统Scanner模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
4	hpet_clk_ctrl	RW	0x1	扫描系统HPET模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
3	pmlio_clk_ctrl	RW	0x1	扫描系统PM1IO模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
2	spi_clk_ctrl	RW	0x1	扫描系统SPI模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
1	gpio_clk_ctrl	RW	0x1	扫描系统GPIO模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。
0	uart_clk_ctrl	RW	0x1	扫描系统UART模块时钟门控配置： 1：时钟门控打开；0：时钟门控关闭。

#### 5.4.57 扫描系统设备复位控制寄存器

扫描系统各设备复位配置寄存器，包括扫描系统各功能模块软件复位的控制。

寄存器地址：0x15202034。

表 5- 66 扫描系统设备复位控制寄存器

位域	名称	访问	缺省值	描述
31:9	Reserved	RO	0x7fffff	-
8	sc132_rst_ctrl	RW	0x1	扫描系统处理器核复位软件配置： 1：模块复位无效；0：模块复位有效。
7:6	Reserved	RO	0x3	-
5	scvid_rst_ctrl	RW	0x1	扫描系统Scanner模块复位软件配置： 1：模块复位无效；0：模块复位有效。
4	hpet_rst_ctrl	RW	0x1	扫描系统HPET模块复位软件配置： 1：模块复位无效；0：模块复位有效。
3	pmlio_rst_ctrl	RW	0x1	扫描系统PM0IO模块复位软件配置： 1：模块复位无效；0：模块复位有效。
2	spi_rst_ctrl	RW	0x1	扫描系统SPI模块复位软件配置： 1：模块复位无效；0：模块复位有效。
1	gpio_rst_ctrl	RW	0x1	扫描系统GPIO模块复位软件配置： 1：模块复位无效；0：模块复位有效。
0	uart_rst_ctrl	RW	0x1	扫描系统UART模块复位软件配置： 1：模块复位无效；0：模块复位有效。

#### 5.4.58 扫描系统 GPIO0~15 复用配置寄存器

扫描系统 GPIO0~15复用配置寄存器。

寄存器地址：0x15202040。

表 5- 67 扫描系统 GPIO0~15 复用配置寄存器

位域	名称	访问	缺省值	描述
31:30	GPIO15_MUX	RW	0x0	GPIO15引脚复用配置： 00：复用为GPIO15；01：第一复用； 10：第二复用；11：引脚主功能。
29:28	GPIO14_MUX	RW	0x0	GPIO14引脚复用配置： 00：复用为GPIO14；01：第一复用； 10：第二复用；11：引脚主功能。

位域	名称	访问	缺省值	描述
27:26	GPI013_MUX	RW	0x0	GPI013引脚复用配置： 00：复用为GPI013；01：第一复用； 10：第二复用； 11：引脚主功能。
25:24	GPI012_MUX	RW	0x0	GPI012引脚复用配置： 00：复用为GPI012；01：第一复用； 10：第二复用； 11：引脚主功能。
23:22	GPI011_MUX	RW	0x0	GPI011引脚复用配置： 00：复用为GPI012；01：第一复用； 10：第二复用； 11：引脚主功能。
21:20	GPI010_MUX	RW	0x0	GPI010引脚复用配置： 00：复用为GPI010；01：第一复用； 10：第二复用； 11：引脚主功能。
19:18	GPI009_MUX	RW	0x0	GPI009引脚复用配置： 00：复用为GPI009；01：第一复用； 10：第二复用； 11：引脚主功能。
17:16	GPI008_MUX	RW	0x0	GPI008引脚复用配置： 00：复用为GPI008；01：第一复用； 10：第二复用； 11：引脚主功能。
15:14	GPI007_MUX	RW	0x0	GPI007引脚复用配置： 00：复用为GPI007；01：第一复用； 10：第二复用； 11：引脚主功能。
13:12	GPI006_MUX	RW	0x0	GPI006引脚复用配置： 00：复用为GPI006；01：第一复用； 10：第二复用； 11：引脚主功能。
11:10	GPI005_MUX	RW	0x0	GPI005引脚复用配置： 00：复用为GPI005；01：第一复用； 10：第二复用； 11：引脚主功能。
9:8	GPI004_MUX	RW	0x0	GPI004引脚复用配置： 00：复用为GPI004；01：第一复用； 10：第二复用； 11：引脚主功能。
7:6	GPI003_MUX	RW	0x0	GPI003引脚复用配置： 00：复用为GPI003；01：第一复用； 10：第二复用； 11：引脚主功能。
5:4	GPI002_MUX	RW	0x0	GPI002引脚复用配置： 00：复用为GPI002；01：第一复用； 10：第二复用； 11：引脚主功能。
3:2	GPI001_MUX	RW	0x0	GPI001引脚复用配置： 00：复用为GPI001；01：第一复用； 10：第二复用； 11：引脚主功能。
1:0	GPI000_MUX	RW	0x0	GPI000引脚复用配置： 00：复用为GPI000；01：第一复用； 10：第二复用； 11：引脚主功能。

#### 5.4.59 扫描系统 GPI016~31 复用配置寄存器

扫描系统 GPI016~31复用配置寄存器。

寄存器地址：0x15202044。

表 5- 68 扫描系统 GPI016~31 复用配置寄存器

位域	名称	访问	缺省值	描述
31:30	GPI031_MUX	RW	0x0	GPI031引脚复用配置： 00：复用为GPI031；01：第一复用； 10：第二复用； 11：引脚主功能。
29:28	GPI030_MUX	RW	0x0	GPI030引脚复用配置： 00：复用为GPI030；01：第一复用； 10：第二复用； 11：引脚主功能。

位域	名称	访问	缺省值	描述
27:26	GPI029_MUX	RW	0x0	GPI029引脚复用配置： 00：复用为GPI029；01：第一复用； 10：第二复用； 11：引脚主功能。
25:24	GPI028_MUX	RW	0x0	GPI028引脚复用配置： 00：复用为GPI028；01：第一复用； 10：第二复用； 11：引脚主功能。
23:22	GPI027_MUX	RW	0x0	GPI027引脚复用配置： 00：复用为GPI027；01：第一复用； 10：第二复用； 11：引脚主功能。
21:20	GPI026_MUX	RW	0x0	GPI026引脚复用配置： 00：复用为GPI026；01：第一复用； 10：第二复用； 11：引脚主功能。
19:18	GPI025_MUX	RW	0x0	GPI025引脚复用配置： 00：复用为GPI025；01：第一复用； 10：第二复用； 11：引脚主功能。
17:16	GPI024_MUX	RW	0x0	GPI024引脚复用配置： 00：复用为GPI024；01：第一复用； 10：第二复用； 11：引脚主功能。
15:14	GPI023_MUX	RW	0x0	GPI023引脚复用配置： 00：复用为GPI023；01：第一复用； 10：第二复用； 11：引脚主功能。
13:12	GPI022_MUX	RW	0x0	GPI022引脚复用配置： 00：复用为GPI022；01：第一复用； 10：第二复用； 11：引脚主功能。
11:10	GPI021_MUX	RW	0x0	GPI021引脚复用配置： 00：复用为GPI021；01：第一复用； 10：第二复用； 11：引脚主功能。
9:8	GPI020_MUX	RW	0x0	GPI020引脚复用配置： 00：复用为GPI020；01：第一复用； 10：第二复用； 11：引脚主功能。
7:6	GPI019_MUX	RW	0x0	GPI019引脚复用配置： 00：复用为GPI019；01：第一复用； 10：第二复用； 11：引脚主功能。
5:4	GPI018_MUX	RW	0x0	GPI018引脚复用配置： 00：复用为GPI018；01：第一复用； 10：第二复用； 11：引脚主功能。
3:2	GPI017_MUX	RW	0x0	GPI017引脚复用配置： 00：复用为GPI017；01：第一复用； 10：第二复用； 11：引脚主功能。
1:0	GPI016_MUX	RW	0x0	GPI016引脚复用配置： 00：复用为GPI016；01：第一复用； 10：第二复用； 11：引脚主功能。

#### 5.4.60 扫描系统 GPI032~36 复用配置寄存器

扫描系统 GPI032~36复用配置寄存器。

寄存器地址：0x15202048。

表 5- 69 扫描系统 GPI032~36 复用配置寄存器

位域	名称	访问	缺省值	描述
31:10	Reserved	RO	0x0	-
9:8	GPI036_MUX	RW	0x0	GPI036引脚复用配置： 00：复用为GPI036；01：第一复用； 10：第二复用； 11：引脚主功能。

位域	名称	访问	缺省值	描述
7:6	GPI035_MUX	RW	0x0	GPI035引脚复用配置： 00：复用为GPI035；01：第一复用； 10：第二复用； 11：引脚主功能。
5:4	GPI034_MUX	RW	0x0	GPI034引脚复用配置： 00：复用为GPI034；01：第一复用； 10：第二复用； 11：引脚主功能。
3:2	GPI033_MUX	RW	0x0	GPI033引脚复用配置： 00：复用为GPI033；01：第一复用； 10：第二复用； 11：引脚主功能。
1:0	GPI032_MUX	RW	0x0	GPI032引脚复用配置： 00：复用为GPI032；01：第一复用； 10：第二复用； 11：引脚主功能。

### 5.4.61 扫描系统通用配置寄存器

扫描系统通用配置寄存器0，包括 UART 引脚复用模式、GMAC、USB、内存控制器的配置等。

寄存器地址：0x15202070。

表 5- 70 扫描系统通用配置寄存器 0

位域	名称	访问	缺省值	描述
31:9	Reserved	RO	0x0	-
8	pmlio_extin_sel	RW	0x0	PMIO模块EXTIN[31:0]信号输入选择： 1：由PMIO_INTCON[31:0]端口输入； 0：由PMIO_INTCON[31:8]、SCA_CISCLK[6:0]、 SCA_AFECLK依次输入
7:6	Reserved	RO	0x0	-
5	sc132_nmi	RW	0x0	扫描系统处理器核NMI输入信号，高电平有效
4	hpet_int_ctrl	RW	0x0	hpet中断输出模式配置： 0：单中断输出模式(内部3个计数器共享1个中断)； 1：三中断输出模式(内部3个计数器独立3个中断)
3:0	Reserved	RO	0x0	-

### 5.4.62 扫描系统采样参数寄存器

扫描系统采样参数寄存器，包括扫描系统启动配置及其他系统休眠相关信息。

寄存器地址：0x15202078。

表 5- 71 扫描系统采样参数寄存器

位域	名称	访问	缺省值	描述
31:9	Reserved	RO	0x0	-
8	sc132_sleeping_sta	RO	0x0	扫描系统处理器核休眠状态监测： 0=WORKON状态 1=SLEEPING状态
7	Reserved	RO	0x0	-
6:5	otg_clkmode	RO	0x0	OTG参考时钟模式输入 00=外部差分晶体输入(XI/XO) 01/10=外部单端晶振输入(XO) 11=内部参考时钟输入
4	sdio1_mode	RO	0x0	SDIO1模式配置输入 0=SDIO模式 1=EMMC模式

位域	名称	访问	缺省值	描述
3	sdio0_mode	RO	0x0	SDIO0模式配置输入 0=SDIO模式 1=EMMC模式
2:1	clk_sel	RO	0x0	芯片内部PLL输出时钟上电配置选择： 00：硬件低频时钟配置模式，PLL按照低频配置参数输出时钟 (NODE:500M, DDR:480M, NET:320M)； 01：硬件高频时钟配置模式，PLL按照高频配置参数输出时钟 (NODE:725M, DDR:600M, NET:400M)； 10：软件配置模式，PLL按照软件配置选择输出时钟； 11：硬件bypass模式，PLL输出时钟全部使用外部输入系统时钟。
0	boot_sel	RO	0x0	芯片启动选择方式： 0：SPI0启动；1：SDIO0/eMMC0启动。

### 5.4.63 扫描系统高精度计数器

扫描系统计数寄存器，32位高精度时钟计数器，工作频率为内部总线时钟频率。

寄存器地址：0x1520207c。

表 5- 72 扫描系统高精度计数器

位域	名称	访问	缺省值	描述
31:0	SCACHIP_HPT	RW	0x0	32位高精度时钟计数器

### 5.4.64 扫描系统启动入口配置寄存器

扫描系统启动入口配置寄存器，配置扫描系统处理器核启动地址。

寄存器地址：0x15202100。

表 5- 73 扫描系统启动入口配置寄存器

位域	名称	访问	缺省值	描述
31:0	sc132_boot_entry	RW	0x1c000000	配置扫描系统处理器核启动入口地址，bit[0]为地址配置有效使能位，高电平有效： 1-入口地址配置有效；0-入口地址配置无效，采用默认启动入口地址 (0x1c000000)

### 5.4.65 扫描系统 0 号 MAILBOX 中断状态寄存器

扫描系统0号 MAILBOX 中断状态寄存器，记录中断触发状态。

寄存器地址：0x15202800。

表 5- 74 扫描系统 0 号 MAILBOX 中断状态寄存器

位域	名称	访问	缺省值	描述
31:0	mlb0_ipi_status	RO	0x0	0号Mailbox中断状态寄存器，记录被触发中断状态位，任何一位被置1且对应中断使能情况下，路由至对应处理器核中断位将被置位

### 5.4.66 扫描系统 0 号 MAILBOX 中断使能寄存器

扫描系统0号 MAILBOX 中断使能寄存器，使能相应中断位。

寄存器地址：0x15202804。

表 5- 75 扫描系统 0 号 MAILBOX 中断使能寄存器

位域	名称	访问	缺省值	描述
31:0	mlb0_ipi_enable	RW	0x0	0号Mailbox中断使能寄存器，使能对应中断位，高电平有效

#### 5.4.67 扫描系统 0 号 MAILBOX 中断触发寄存器

扫描系统0号 MAILBOX 中断触发寄存器，触发相应中断位。

寄存器地址：0x15202808。

表 5- 76 扫描系统 0 号 MAILBOX 中断触发寄存器

位域	名称	访问	缺省值	描述
31:0	mlb0_ipi_set	W	-	0号Mailbox中断触发寄存器，触发对应中断位，高电平有效。当相应中断使能位有效时，该位置1将触发相应中断，并将路由至对应处理器核中断位置

#### 5.4.68 扫描系统 0 号 MAILBOX 中断清除寄存器

扫描系统0号 MAILBOX 中断清除寄存器，清除相应中断状态位。

寄存器地址：0x1520280c。

表 5- 77 扫描系统 0 号 MAILBOX 中断清除寄存器

位域	名称	访问	缺省值	描述
31:0	mlb0_ipi_clear	W	-	0号Mailbox中断清除寄存器，清除对应中断位，写1清除。对应位写1将清除相应中断状态位，取消相应中断触发，中断使能位不做清除

#### 5.4.69 扫描系统 0 号 MAILBOX 消息寄存器 0~7

扫描系统0号 MAILBOX 消息寄存器0~7，传递记录相应消息，可由用户自由定义。

寄存器地址：0x15202810~0x1520282c。

表 5- 78 扫描系统 0 号 MAILBOX 消息寄存器 0~7

位域	名称	访问	缺省值	描述
31:0	mlb0_message7	RW	0x0	0号Mailbox消息寄存器7，记录相应消息定义，可进行不同系统间消息传递
31:0	mlb0_message6	RW	0x0	0号Mailbox消息寄存器6，记录相应消息定义，可进行不同系统间消息传递
31:0	mlb0_message5	RW	0x0	0号Mailbox消息寄存器5，记录相应消息定义，可进行不同系统间消息传递
31:0	mlb0_message4	RW	0x0	0号Mailbox消息寄存器4，记录相应消息定义，可进行不同系统间消息传递
31:0	mlb0_message3	RW	0x0	0号Mailbox消息寄存器3，记录相应消息定义，可进行不同系统间消息传递
31:0	mlb0_message2	RW	0x0	0号Mailbox消息寄存器2，记录相应消息定义，可进行不同系统间消息传递
31:0	mlb0_message1	RW	0x0	0号Mailbox消息寄存器1，记录相应消息定义，可进行不同系统间消息传递
31:0	mlb0_message0	RW	0x0	0号Mailbox消息寄存器0，记录相应消息定义，可进行不同系统间消息传递



### 5.4.70 扫描系统 1 号 MAILBOX 中断状态寄存器

扫描系统1号 MAILBOX 中断状态寄存器，记录中断触发状态。

寄存器地址：0x15202830。

表 5- 79 扫描系统 1 号 MAILBOX 中断状态寄存器

位域	名称	访问	缺省值	描述
31:0	mlbl_ipi_status	RO	0x0	1号Mailbox中断状态寄存器，记录被触发中断状态位，任何一位被置1且对应中断使能情况下，路由至对应处理器核中断位将被置位

### 5.4.71 扫描系统 1 号 MAILBOX 中断使能寄存器

扫描系统1号 MAILBOX 中断使能寄存器，使能相应中断位。

寄存器地址：0x15202834。

表 5- 80 扫描系统 1 号 MAILBOX 中断使能寄存器

位域	名称	访问	缺省值	描述
31:0	mlbl_ipi_enable	RW	0x0	1号Mailbox中断使能寄存器，使能对应中断位，高电平有效

### 5.4.72 扫描系统 1 号 MAILBOX 中断触发寄存器

扫描系统1号 MAILBOX 中断触发寄存器，触发相应中断位。

寄存器地址：0x15202838。

表 5- 81 扫描系统 1 号 MAILBOX 中断触发寄存器

位域	名称	访问	缺省值	描述
31:0	mlbl_ipi_set	W	-	1号Mailbox中断触发寄存器，触发对应中断位，高电平有效。当相应中断使能位有效时，该位置1将触发相应中断，并将路由至对应处理器核中断位置位

### 5.4.73 扫描系统 1 号 MAILBOX 中断清除寄存器

扫描系统1号 MAILBOX 中断清除寄存器，清除相应中断状态位。

寄存器地址：0x1520283c。

表 5- 82 扫描系统 1 号 MAILBOX 中断清除寄存器

位域	名称	访问	缺省值	描述
31:0	mlbl_ipi_clear	W	-	1号Mailbox中断清除寄存器，清除对应中断位，写1清除。对应位写1将清除相应中断状态位，取消相应中断触发，中断使能位不做清除

### 5.4.74 扫描系统 1 号 MAILBOX 消息寄存器 0~7

扫描系统1号 MAILBOX 消息寄存器0~7，传递记录相应消息，可由用户自由定义。

寄存器地址：0x15202840~0x1520285c。

表 5- 83 扫描系统 1 号 MAILBOX 消息寄存器 0~7

位域	名称	访问	缺省值	描述
31:0	mlbl_message7	RW	0x0	1号Mailbox消息寄存器7，记录相应消息定义，可进行不同系统间消息传递



位域	名称	访问	缺省值	描述
31:0	mlb1_message6	RW	0x0	1号Mailbox消息寄存器6, 记录相应消息定义, 可进行不同系统间消息传递
31:0	mlb1_message5	RW	0x0	1号Mailbox消息寄存器5, 记录相应消息定义, 可进行不同系统间消息传递
31:0	mlb1_message4	RW	0x0	1号Mailbox消息寄存器4, 记录相应消息定义, 可进行不同系统间消息传递
31:0	mlb1_message3	RW	0x0	1号Mailbox消息寄存器3, 记录相应消息定义, 可进行不同系统间消息传递
31:0	mlb1_message2	RW	0x0	1号Mailbox消息寄存器2, 记录相应消息定义, 可进行不同系统间消息传递
31:0	mlb1_message1	RW	0x0	1号Mailbox消息寄存器1, 记录相应消息定义, 可进行不同系统间消息传递
31:0	mlb1_message0	RW	0x0	1号Mailbox消息寄存器0, 记录相应消息定义, 可进行不同系统间消息传递

#### 5.4.75 扫描系统 2 号 MAILBOX 中断状态寄存器

扫描系统2号 MAILBOX 中断状态寄存器, 记录中断触发状态。

寄存器地址: 0x15202860。

表 5- 84 扫描系统 2 号 MAILBOX 中断状态寄存器

位域	名称	访问	缺省值	描述
31:0	mlb2_ipi_status	RO	0x0	2号Mailbox中断状态寄存器, 记录被触发中断状态位, 任何一位被置1且对应中断使能情况下, 路由至对应处理器核中断位将被置位

#### 5.4.76 扫描系统 2 号 MAILBOX 中断使能寄存器

扫描系统2号 MAILBOX 中断使能寄存器, 使能相应中断位。

寄存器地址: 0x15202864。

表 5- 85 扫描系统 2 号 MAILBOX 中断使能寄存器

位域	名称	访问	缺省值	描述
31:0	mlb2_ipi_enable	RW	0x0	2号Mailbox中断使能寄存器, 使能对应中断位, 高电平有效

#### 5.4.77 扫描系统 2 号 MAILBOX 中断触发寄存器

扫描系统2号 MAILBOX 中断触发寄存器, 触发相应中断位。

寄存器地址: 0x15202868。

表 5- 86 扫描系统 2 号 MAILBOX 中断触发寄存器

位域	名称	访问	缺省值	描述
31:0	mlb2_ipi_set	W	-	2号Mailbox中断触发寄存器, 触发对应中断位, 高电平有效。当相应中断使能位有效时, 该位置1将触发相应中断, 并将路由至对应处理器核中断位置位

#### 5.4.78 扫描系统 2 号 MAILBOX 中断清除寄存器

扫描系统2号 MAILBOX 中断清除寄存器, 清除相应中断状态位。

寄存器地址: 0x1520286c。

表 5- 87 扫描系统 2 号 MAILBOX 中断清除寄存器

位域	名称	访问	缺省值	描述
31:0	mlb2_ipi_clear	W	-	2号Mailbox中断清除寄存器，清除对应中断位，写1清除。对应位写1将清除相应中断状态位，取消相应中断触发，中断使能位不做清除

#### 5.4.79 扫描系统 2 号 MAILBOX 消息寄存器 0~7

扫描系统2号 MAILBOX 消息寄存器0~7，传递记录相应消息，可由用户自由定义。

寄存器地址：0x15202870~0x1520288c。

表 5- 88 扫描系统 0 号 MAILBOX 消息寄存器 0~7

位域	名称	访问	缺省值	描述
31:0	mlb2_message7	RW	0x0	2号Mailbox消息寄存器7，记录相应消息定义，可进行不同系统间消息传递
31:0	mlb2_message6	RW	0x0	2号Mailbox消息寄存器6，记录相应消息定义，可进行不同系统间消息传递
31:0	mlb2_message5	RW	0x0	2号Mailbox消息寄存器5，记录相应消息定义，可进行不同系统间消息传递
31:0	mlb2_message4	RW	0x0	2号Mailbox消息寄存器4，记录相应消息定义，可进行不同系统间消息传递
31:0	mlb2_message3	RW	0x0	2号Mailbox消息寄存器3，记录相应消息定义，可进行不同系统间消息传递
31:0	mlb2_message2	RW	0x0	2号Mailbox消息寄存器2，记录相应消息定义，可进行不同系统间消息传递
31:0	mlb2_message1	RW	0x0	2号Mailbox消息寄存器1，记录相应消息定义，可进行不同系统间消息传递
31:0	mlb2_message0	RW	0x0	2号Mailbox消息寄存器0，记录相应消息定义，可进行不同系统间消息传递

#### 5.4.80 扫描系统 MAILBOX 中断路由寄存器

扫描系统 MAILBOX 中断路由寄存器，配置0~2号 MAILBOX 中断路由。

寄存器地址：0x15202890。

表 5- 89 打印系统 MAILBOX 中断路由寄存器

位域	名称	访问	缺省值	描述
31:11	Reserved	RO	0x0	-
10:8	mlb2_ipi_map	RW	0x0	2号Mailbox中断路由配置： 001:2号Mailbox触发中断路由至主系统中断； 010:2号Mailbox触发中断路由至打印系统中断； 100:2号Mailbox触发中断路由至扫描系统中断
7	Reserved	RO	0x0	-
6:4	mlb1_ipi_map	RW	0x0	1号Mailbox中断路由配置： 001:1号Mailbox触发中断路由至主系统中断； 010:1号Mailbox触发中断路由至打印系统中断； 100:1号Mailbox触发中断路由至扫描系统中断
3	Reserved	RO	0x0	-
2:0	mlb0_ipi_map	RW	0x0	0号Mailbox中断路由配置： 001:0号Mailbox触发中断路由至主系统中断； 010:0号Mailbox触发中断路由至打印系统中断； 100:0号Mailbox触发中断路由至扫描系统中断

## 5.5 中断配置及路由

龙芯2P0500芯片中断根据不同系统应用需求，分为主系统、打印系统、扫描系统中断。

### 5.5.1 各系统中断概述

其中，主系统分为传统、扩展中断两种方式：

#### 一、主系统传统中断方式

传统中断方式最多支持64个中断源，以统一方式进行管理，如下图所示，任意一个IO中断源可以被配置为是否使能、触发的方式、以及被路由的目标处理器核中断脚。

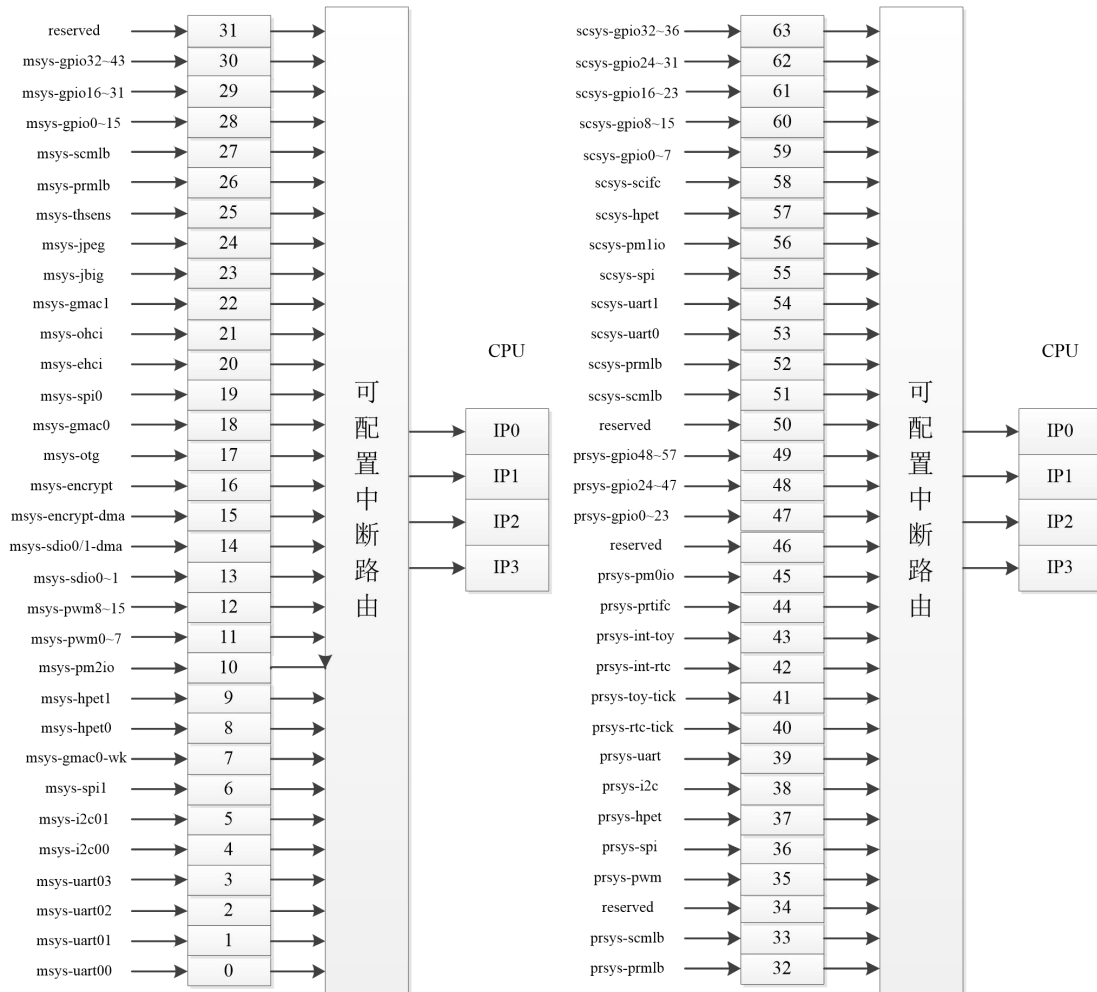


图5- 2 主系统传统 IO 中断路由示意图

传统中断相关配置寄存器都是以位的形式对相应的中断线进行控制，中断控制位连接及属性配置见表 5-94。中断使能(Enable)的配置有三个寄存器：**Intenset**、**Intenclr** 和 **Inten**。**Intenset** 设置中断使能，**Intenset** 寄存器写 1 的位对应的中断位被使能。**Intenclr** 清除中断使能，**Intenclr** 寄存器写 1 的位对应的中断状态和中断使能同时被清除。**Inten** 中断使能寄存器(只读)，读取当前各中断使能的情况。脉冲形式的中断信号由 **Intedge** 配置寄存器来选择，写 1 表示脉冲触发，写 0 表示电平触发。**Intpol** 中断极性寄存器，电平触发模式下，写 0 对

应中断源为高电平触发中断，写 1 对应中断源为低电平触发中断；边沿触发模式下，写 0 对应中断源为上升沿触发中断，写 1 对应中断源为下降沿触发中断。中断处理程序可以通过 Intenclr 的相应位来清除脉冲记录,在中断被清除后，由于中断使能位同时被清除，需要重新配置相应的 Intenset，使能相应中断位才能采集到该中断的下一次中断触发。

表 5- 90 主系统传统 IO 中断寄存器列表

地址	名称	描述
0x14001040	CORE_INTISRO	路由给CORE的低32位中断状态
0x14001044	INTISRO	低32位中断状态寄存器
0x14001048	CORE_INTISR1	路由给CORE的高32位中断状态
0x1400104c	INTISR1	高32位中断状态寄存器
0x14001400	ENTRY0_0	8位中断路由寄存器[0--7]
0x14001408	ENTRY8_0	8位中断路由寄存器[8--15]
0x14001410	ENTRY16_0	8位中断路由寄存器[16--23]
0x14001418	ENTRY24_0	8位中断路由寄存器[24--31]
0x14001420	INTISR_0	低32位中断状态寄存器
0x14001424	INTIEN_0	低32位中断使能状态寄存器
0x14001428	INTSET_0	低32位设置使能寄存器
0x1400142c	INTCLR_0	低32位中断清除寄存器，清除使能寄存器和脉冲触发的中断
0x14001430	INTPOL_0	低32位极性设置寄存器(电平中断)
0x14001434	INTEDGE_0	低32位触发方式寄存器（1：脉冲触发；0：电平触发）
0x14001440	ENTRY0_1	8位中断路由寄存器[32--39]
0x14001448	ENTRY8_1	8位中断路由寄存器[40--47]
0x14001450	ENTRY16_1	8位中断路由寄存器[48--55]
0x14001458	ENTRY24_1	8位中断路由寄存器[56--63]
0x14001460	INTISR_1	高32位中断状态寄存器
0x14001464	INTIEN_1	高32位中断使能状态寄存器
0x14001468	INTSET_1	高32位设置使能寄存器
0x1400146c	INTCLR_1	高32位中断清除寄存器，清除使能寄存器和脉冲触发的中断
0x14001470	INTPOL_1	高32位极性设置寄存器(电平中断)
0x14001474	INTEDGE_1	高32位触发方式寄存器（1：脉冲触发；0：电平触发）
0x14001500	Thsens_int_ctrl_Hi0	温度传感器高温中断控制寄存器0
0x14001504	Thsens_int_ctrl_Hi1	温度传感器高温中断控制寄存器1
0x14001508	Thsens_int_ctrl_Lo0	温度传感器低温中断控制寄存器0
0x1400150c	Thsens_int_ctrl_Lo1	温度传感器低温中断控制寄存器1
0x14001510	Thsens_int_status/c lr	温度传感器中断状态寄存器
0x14001514	Thsens_value	温度传感器测量值
0x14001520	Thsens_scale_hi0	温度传感器高阈值寄存器0
0x14001524	Thsens_scale_hi1	温度传感器高阈值寄存器1

## 二、主系统扩展中断方式

龙芯2P0500主系统除了支持上述传统IO中断方式外，还增加了扩展IO中断方式，即：将2P0500芯片中所有IO设备中断，全部映射至全新的扩展IO中断向量，并增加相应扩展中断使能、中断状态、中断清除及路由等功能。该扩展IO中断向量最多支持128个IO设备，IO设备与扩展中断号对应关系如下图所示，该扩展中断可独立于传统IO中断处理方式之外，填补原传统IO中断方式仅64个有限中断源的处理限制，提升IO中断使用的灵活性。

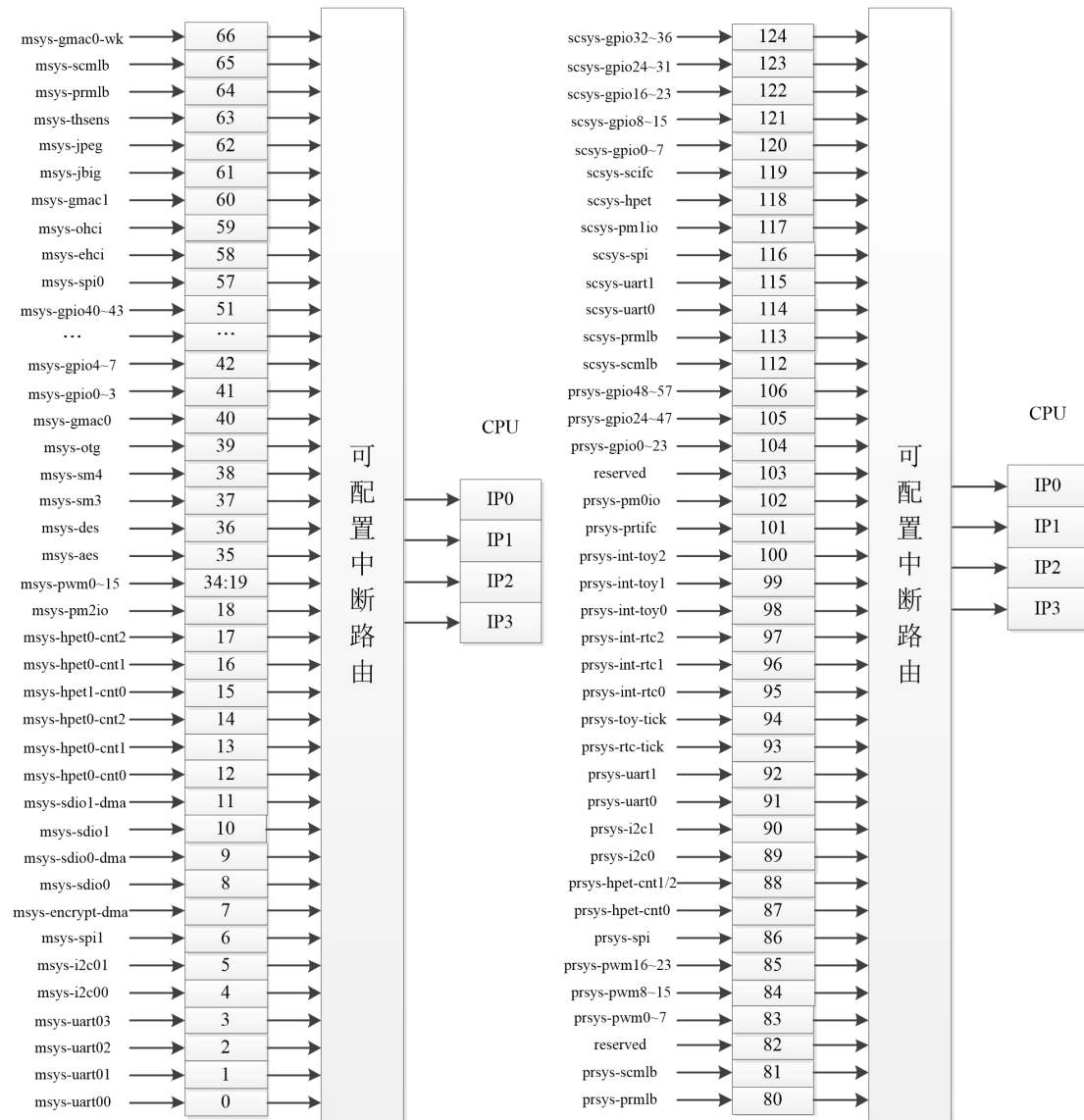


图5- 3 主系统扩展 IO 中断路由示意图

2P0500 芯片主系统在使用扩展 IO 中断前，需要使能“主系统通用配置寄存器 0”中的扩展 IO 中断使能位。该寄存器配置地址为 0x14000100，对应配置位如下表：

位域	名称	访问	缺省值	描述
15	extioi_enable	RW	0x0	扩展中断使能控制位： 1: 打印扩展中断(传统/扩展中断同时有效)；0: 关闭扩展中断(仅传统中断有效)

扩展 IO 中断相关配置寄存器与传统中断寄存器类似，都是以位的形式对相应的中断线进行控制，中断控制位连接及属性配置见表 5-96。扩展中断配置寄存器主要有三种寄存器：扩展中断使能 EXTINT\_IEN、扩展中断状态 EXTINT\_ISR 和扩展中断清除 EXTINT\_ICLR。EXTINT\_IEN 设置扩展中断使能，相应寄存器写 1 的位对应的中断被使能。EXTINT\_ISR 扩展中断状态，寄存器相应位为 1 表示对应 IO 中断有效(注：该状态不依赖于中断使能位是否被置起)。EXTINT\_ICLR 扩展中断清除寄存器，与扩展中断状态寄存器共用，对应位写 1 清除对应中断状态。处理器核增加扩展中断状态 CORE\_EXTISR 和中断清除寄存器 CORE\_EXTICLR，CORE\_EXTISR 处理器核中断状态寄存器，当对应扩展 IO 中断使能位有效且 IO 中断有效时，该寄存器对应中断状态位被置起，表示对应 IO 中断可被处理器核接收。CORE\_EXTICLR 中断清除寄存器与状态寄存器共用，对应位写 1 清除对应中断状态。

此外，扩展 IO 中断增加路由配置寄存器 EXTINT\_MAP，该寄存器与传统 IO 中断路由类似，可将所有扩展 IO 中断按组分类路由至处理器核中断引脚向量，对应 CP0\_Status 寄存器的 IP0 到 IP3。

表 5- 91 主系统 IO 扩展中断寄存器列表

地址	名称	描述
0x14001148	EXTIOI_ACK	扩展中断设备反馈寄存器
0x140014c0	EXTIOI_MAP	扩展中断设备路由寄存器
0x14001600	EXTIOI_IEN0	扩展中断设备使能寄存器0
0x14001604	EXTIOI_IEN1	扩展中断设备使能寄存器1
0x14001608	EXTIOI_IEN2	扩展中断设备使能寄存器2
0x1400160c	EXTIOI_IEN3	扩展中断设备使能寄存器3
0x14001640	EXTIOI_POLO	扩展中断电平配置寄存器0
0x14001644	EXTIOI_POL1	扩展中断电平配置寄存器1
0x14001648	EXTIOI_POL2	扩展中断电平配置寄存器2
0x1400164c	EXTIOI_POL3	扩展中断电平配置寄存器3
0x14001700	EXTIOI_ISR0/ICLR0	扩展中断状态/清除寄存器0
0x14001704	EXTIOI_ISR1/ICLR1	扩展中断状态/清除寄存器1
0x14001708	EXTIOI_ISR2/ICLR2	扩展中断状态/清除寄存器2
0x1400170c	EXTIOI_ISR3/ICLR3	扩展中断状态/清除寄存器3
0x14001800	EXTIOI_CORE_ISR0/ICLR0	路由至CORE扩展中断状态/清除寄存器0
0x14001804	EXTIOI_CORE_ISR1/ICLR1	路由至CORE扩展中断状态/清除寄存器1
0x14001808	EXTIOI_CORE_ISR2/ICLR2	路由至CORE扩展中断状态/清除寄存器2
0x1400180c	EXTIOI_CORE_ISR3/ICLR3	路由至CORE扩展中断状态/清除寄存器3



### 三、打印系统中断方式

其中，打印系统中断方式最多支持96个中断源，以统一方式进行管理，如下图所示，任意一个IO中断源可以被配置为是否使能、触发的方式、以及被路由的目标处理器核中断脚。

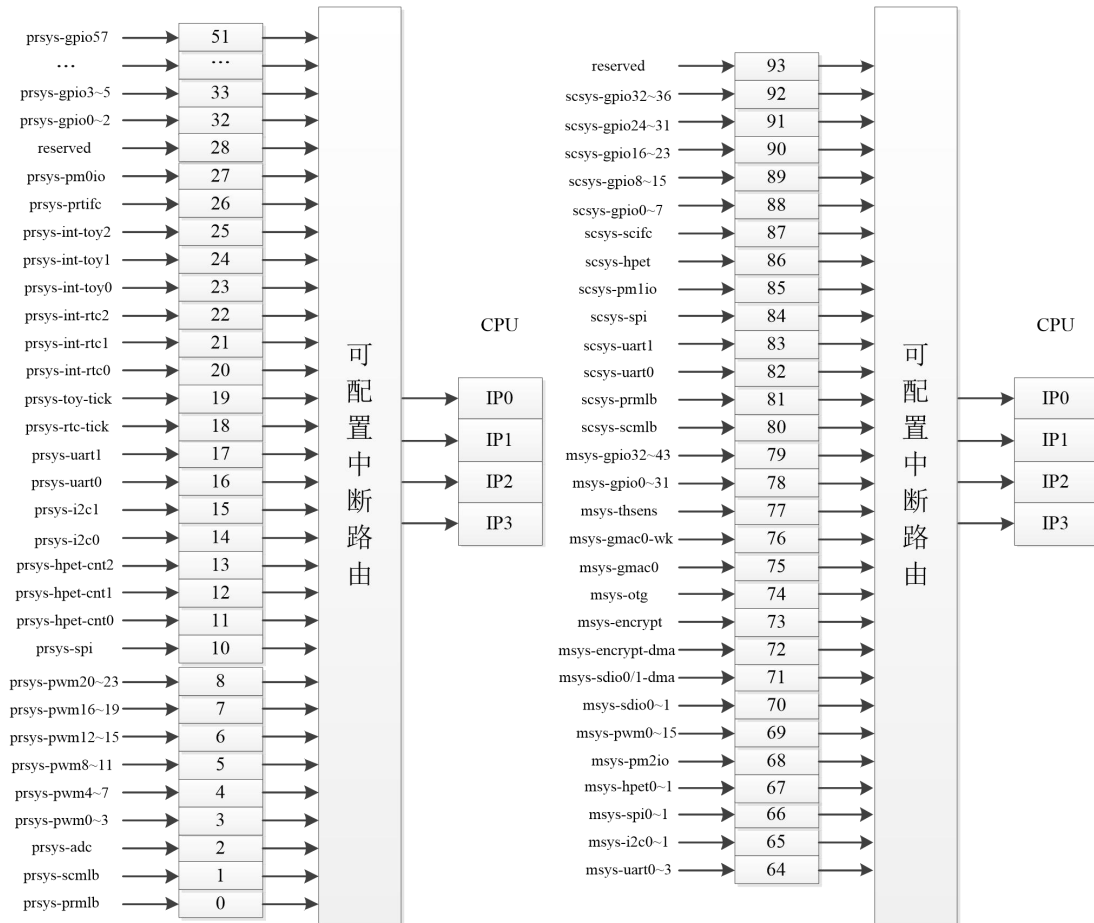


图5- 4 打印系统 IO 中断路由示意图

打印系统中断相关配置寄存器都是以位的形式对相应的中断线进行控制，中断控制位连接及属性配置见表 5-97。中断使能（Enable）的配置 INT\_IEN 寄存器，INT\_IEN 设置中断使能，INT\_IEN 寄存器写 1 的位对应的中断位被使能。中断清除 INT\_CLR 寄存器，清除中断状态位，INT\_CLR 寄存器写 1 的位对应的中断状态被清除（注：仅针对边沿触发形式，且中断使能位不清除）。脉冲形式的中断信号由 INT\_EDGE 配置寄存器来选择，写 1 表示脉冲触发，写 0 表示电平触发。当中断触发配置为脉冲形式后，中断电平配置 INT\_POL 寄存器可选择相应触发沿，0 为下降沿触发，1 为上升沿触发；中断双沿 INT\_DUAL 寄存器可选择是否上升沿、下降沿均触发中断，高电平有效。当中断触发配置为电平触发形式时，中断电平 INT\_POL 可选择触发电平形式，0 为低电平触发，1 为高电平触发。中断处理程序可以通过 INT\_CLR 的相应位来清除脉冲中断记录，在中断被清除后，不需要重新使能相应中断位，即可采集到该中断位的下一次脉冲中断触发。

表 5- 92 打印系统 IO 中断寄存器列表

地址	名称	描述
0x15103400	PRT_INT_ISR0	打印设备中断状态0
0x15103404	PRT_INT_IEN0	打印设备中断使能配置0
0x1510340c	PRT_INT_CLR0	打印设备中断清除配置0
0x15103410	PRT_INT_POLO	打印设备中断电平配置0
0x15103414	PRT_INT_EDGE0	打印设备中断边沿配置0
0x15103418	PRT_INT_DUAL0	打印设备中断双沿配置0
0x15103420	PRT_INT_ISR1	打印设备中断状态1
0x15103424	PRT_INT_IEN1	打印设备中断使能配置1
0x1510342c	PRT_INT_CLR1	打印设备中断清除配置1
0x15103430	PRT_INT_POL1	打印设备中断电平配置1
0x15103434	PRT_INT_EDGE1	打印设备中断边沿配置1
0x15103438	PRT_INT_DUAL1	打印设备中断双沿配置1
0x15103440	PRT_INT_ISR2	打印设备中断状态2
0x15103444	PRT_INT_IEN2	打印设备中断使能配置2
0x1510344c	PRT_INT_CLR2	打印设备中断清除配置2
0x15103450	PRT_INT_POL2	打印设备中断电平配置2
0x15103454	PRT_INT_EDGE2	打印设备中断边沿配置2
0x15103458	PRT_INT_DUAL2	打印设备中断双沿配置2
0x15103460	PRT_INT_MAP0	打印设备中断路由配置0
0x15103464	PRT_INT_MAP1	打印设备中断路由配置1
0x15103468	PRT_INT_MAP2	打印设备中断路由配置2
0x1510346c	PRT_INT_MAP3	打印设备中断路由配置3
0x15103470	PRT_INT_MAP4	打印设备中断路由配置4
0x15103474	PRT_INT_MAP5	打印设备中断路由配置5



#### 四、扫描系统中断方式

其中，扫描系统中断方式最多支持96个中断源，以统一方式进行管理，如下图所示，任意一个IO中断源可以被配置为是否使能、触发的方式、以及被路由的目标处理器核中断脚。

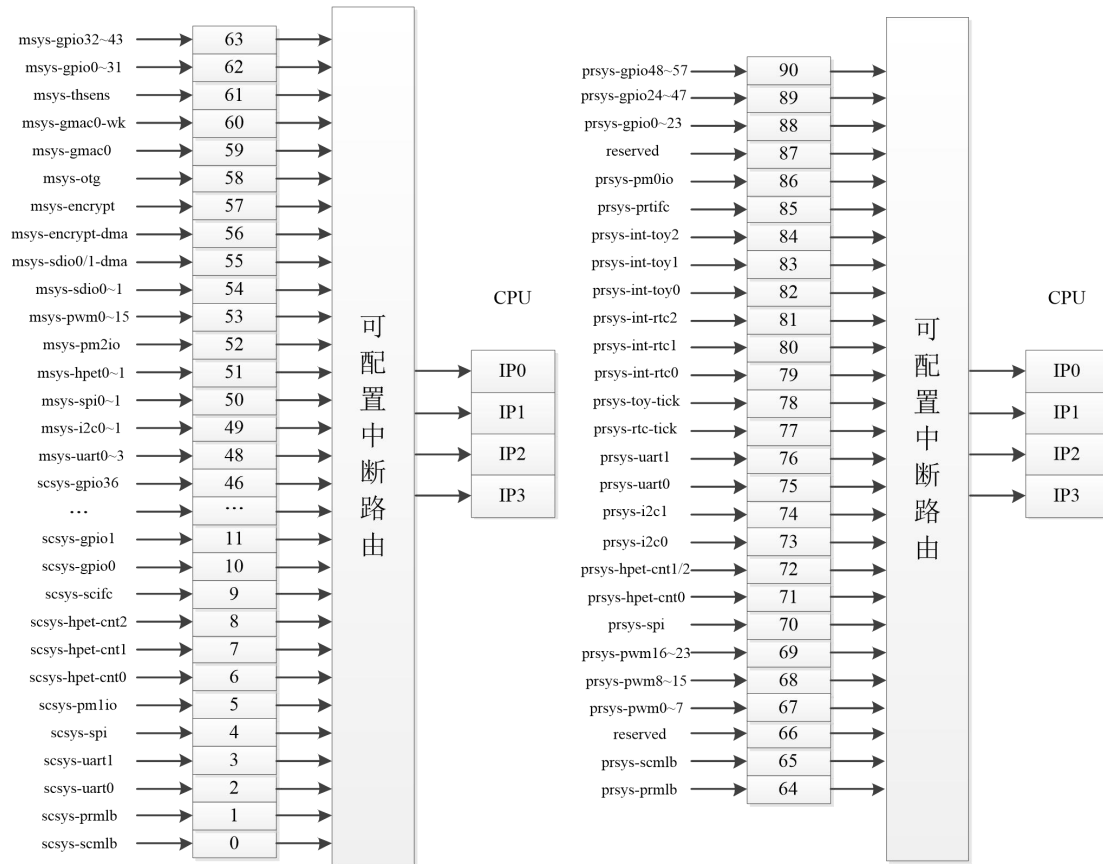


图5- 5 扫描系统 IO 中断路由示意图

扫描系统中断相关配置寄存器都是以位的形式对相应的中断线进行控制，中断控制位连接及属性配置见表 5-97。中断使能（Enable）的配置 INT\_IEN 寄存器，INT\_IEN 设置中断使能，INT\_IEN 寄存器写 1 的位对应的中断位被使能。中断清除 INT\_CLR 寄存器，清除中断状态位，INT\_CLR 寄存器写 1 的位对应的中断状态被清除（注：仅针对边沿触发形式，且中断使能位不清除）。脉冲形式的中断信号由 INT\_EDGE 配置寄存器来选择，写 1 表示脉冲触发，写 0 表示电平触发。当中断触发配置为脉冲形式后，中断电平配置 INT\_POL 寄存器可选择相应触发沿，0 为下降沿触发，1 为上升沿触发；中断双沿 INT\_DUAL 寄存器可选择是否上升沿、下降沿均触发中断，高电平有效。当中断触发配置为电平触发形式时，中断电平 INT\_POL 可选择触发电平形式，0 为低电平触发，1 为高电平触发。中断处理程序可以通过 INT\_CLR 的相应位来清除脉冲中断记录，在中断被清除后，不需要重新使能相应中断位，即可采集到该中断位的下一次脉冲中断触发。

表 5- 93 扫描系统 IO 中断寄存器列表

地址	名称	描述
0x15202400	SCA_INT_ISR0	扫描设备中断状态0
0x15202404	SCA_INT_IEN0	扫描设备中断使能配置0
0x1520240c	SCA_INT_CLR0	扫描设备中断清除配置0
0x15202410	SCA_INT_POLO	扫描设备中断电平配置0
0x15202414	SCA_INT_EDGE0	扫描设备中断边沿配置0
0x15202418	SCA_INT_DUAL0	扫描设备中断双沿配置0
0x15202420	SCA_INT_ISR1	扫描设备中断状态1
0x15202424	SCA_INT_IEN1	扫描设备中断使能配置1
0x1520242c	SCA_INT_CLR1	扫描设备中断清除配置1
0x15202430	SCA_INT_POL1	扫描设备中断电平配置1
0x15202434	SCA_INT_EDGE1	扫描设备中断边沿配置1
0x15202438	SCA_INT_DUAL1	扫描设备中断双沿配置1
0x15202440	SCA_INT_ISR2	扫描设备中断状态2
0x15202444	SCA_INT_IEN2	扫描设备中断使能配置2
0x1520244c	SCA_INT_CLR2	扫描设备中断清除配置2
0x15202450	SCA_INT_POL2	扫描设备中断电平配置2
0x15202454	SCA_INT_EDGE2	扫描设备中断边沿配置2
0x15202458	SCA_INT_DUAL2	扫描设备中断双沿配置2
0x15202460	SCA_INT_MAP0	扫描设备中断路由配置0
0x15202464	SCA_INT_MAP1	扫描设备中断路由配置1
0x15202468	SCA_INT_MAP2	扫描设备中断路由配置2
0x1520246c	SCA_INT_MAP3	扫描设备中断路由配置3
0x15202470	SCA_INT_MAP4	扫描设备中断路由配置4
0x15202474	SCA_INT_MAP5	扫描设备中断路由配置5

## 5.5.2 中断触发类型

对于 2P0500 来说，内部各控制器中断分为脉冲、电平触发类型，gpio 中断根据需要可以配置成电平触发或者脉冲触发。

## 5.5.3 中断相关寄存器描述

表 5- 94 主系统传统中断控制寄存器属性

位域	访问属性/缺省值						中断源
	Intedge	Inten	Intenset	Intenclr	Intpol	Intentry	
0	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_UART00
1	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_UART01
2	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_UART02
3	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_UART03
4	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_I2C0
5	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_I2C1
6	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_SPI1
7	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_GMAC0_WK
8	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_HPETO

位域	访问属性/缺省值						中断源
	Intedge	Inten	Intenset	Intenc1r	Intpol	Intentry	
9	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_HPET1
10	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_PM2IO
11	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_PWM0~7
12	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_PWM8~15
13	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_SDIO0~1
14	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_SDIO0/1-DMA
15	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_ENCRYPT-DMA
16	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_ENCRYPT-DEV
17	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_OTG
18	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_GMAC0
19	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_SPIO
20	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_EHCI
21	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_OHCI
22	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_GMAC1
23	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_JBIG
24	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_JPEG
25	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_THSENS
26	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_PRMLB
27	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_SCMLB
28	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_GPIO0~15
29	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_GPIO16~31
30	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	MSYS_GPIO32~43
31	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	Reserved
32	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	PRSYS_PRMLB
33	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	PRSYS_SCMLB
34	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	Reserved
35	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	PRSYS_PWM
36	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	PRSYS_SPI
37	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	PRSYS_HPET
38	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	PRSYS_I2C
39	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	PRSYS_UART
40	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	PRSYS_RTC_TICK
41	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	PRSYS_TOY_TICK
42	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	PRSYS_INT_RTC
43	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	PRSYS_INT_TOY
44	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	PRSYS_PRTIFC

位域	访问属性/缺省值						中断源
	Intedge	Inten	Intenset	Intenclr	Intpol	Intentry	
45	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	PRSYS_PMOIO
46	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	Reserved
47	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	PRSYS_GPIO0~23
48	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	PRSYS_GPIO24~47
49	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	PRSYS_GPIO48~57
50	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	Reserved
51	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	SCSYS_SCMLB
52	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	SCSYS_PRMLB
53	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	SCSYS_UART0
54	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	SCSYS_UART1
55	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	SCSYS_SPI
56	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	SCSYS_PM1IO
57	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	SCSYS_HPET
58	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	SCSYS_SCIFC
59	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	SCSYS_GPIO0~7
60	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	SCSYS_GPIO8~15
61	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	SCSYS_GPIO16~23
62	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	SCSYS_GPIO24~31
63	RW / 0	R / 0	W / 0	W / 0	RW / 0	RW / 0	SCSYS_GPIO32~36

表 5- 95 传统中断控制寄存器地址

名称	地址偏移	访问属性	缺省值	描述
Intisr_0	0x14001420	RO	NA	低 32 位中断状态寄存器
Inten_0	0x14001424	RO	NA	低 32 位中断使能状态寄存器
Intenset_0	0x14001428	WO	NA	低 32 位设置使能寄存器
Intenclr_0	0x1400142c	WO	NA	低 32 位清除使能寄存器和脉冲触发的中断
Intpol_0	0x14001430	WO	0x0	低 32 位中断极性选择寄存器（1：极性取反，0：极性相同）
Intedge_0	0x14001434	WO	0x0	低 32 位触发方式寄存器（1：脉冲触发；0：电平触发）
Intisr_1	0x14001460	RO	NA	高 32 位中断状态寄存器
Inten_1	0x14001464	RO	NA	高 32 位中断使能状态寄存器
Intenset_1	0x14001468	WO	NA	高 32 位设置使能寄存器
Intenclr_1	0x1400146c	WO	NA	高 32 位清除使能寄存器和脉冲触发的中断
Intpol_1	0x14001470	WO	0x0	高 32 位中断极性选择寄存器（1：极性取反，0：极性相同）
Intedge_1	0x14001474	WO	0x0	高 32 位触发方式寄存器（1：脉冲触发；0：电平触发）
CORE_IPISR	0x14001000	RO	NA	处理器核的 IPI_Status 寄存器

名称	地址偏移	访问属性	缺省值	描述
CORE_IPIEN	0x14001004	RW	0x0	处理器核的 IPI_Enalbe 寄存器
CORE_IPISET	0x14001008	WO	NA	处理器核的 IPI_Set 寄存器
CORE_IPI_CLR	0x1400100c	WO	NA	处理器核的 IPI_Clear 寄存器
CORE_INTISRO	0x14001040	RO	NA	路由给 CORE 的低 32 位中断状态
CORE_INTISR1	0x14001048	RO	NA	路由给 CORE 的高 32 位中断状态

表 5- 96 扩展中断控制寄存器属性

位域	访问属性/缺省值			中断源	位域	访问属性/缺省值			中断源
	IEN	ICLR	POL			IEN	ICLR	POL	
0	R / 0	W / 0	RW / 0	MSYS_UART00	82	R / 0	W / 0	RW / 0	Reserved
1	R / 0	W / 0	RW / 0	MSYS_UART01	83	R / 0	W / 0	RW / 0	PRSYS_PWM0~7
2	R / 0	W / 0	RW / 0	MSYS_UART02	84	R / 0	W / 0	RW / 0	PRSYS_PWM8~15
3	R / 0	W / 0	RW / 0	MSYS_UART03	85	R / 0	W / 0	RW / 0	PRSYS_PWM16~23
4	R / 0	W / 0	RW / 0	MSYS_I2C0	86	R / 0	W / 0	RW / 0	PRSYS_SPI
5	R / 0	W / 0	RW / 0	MSYS_I2C1	87	R / 0	W / 0	RW / 0	PRSYS_HPET_CNT0
6	R / 0	W / 0	RW / 0	MSYS_SPI1	88	R / 0	W / 0	RW / 0	PRSYS_HPET_CNT1/2
7	R / 0	W / 0	RW / 0	MSYS_ENCRYPT-DMA	89	R / 0	W / 0	RW / 0	PRSYS_I2C0
8	R / 0	W / 0	RW / 0	MSYS_SDIO0	90	R / 0	W / 0	RW / 0	PRSYS_I2C1
9	R / 0	W / 0	RW / 0	MSYS_SDIO0_DMA	91	R / 0	W / 0	RW / 0	PRSYS_UART0
10	R / 0	W / 0	RW / 0	MSYS_SDIO1	92	R / 0	W / 0	RW / 0	PRSYS_UART1
11	R / 0	W / 0	RW / 0	MSYS_SDIO1_DMA	93	R / 0	W / 0	RW / 0	PRSYS_RTC_TICK
12	R / 0	W / 0	RW / 0	MSYS_HPETO_CNT0	94	R / 0	W / 0	RW / 0	PRSYS_TOY_TICK
13	R / 0	W / 0	RW / 0	MSYS_HPETO_CNT1	95	R / 0	W / 0	RW / 0	PRSYS_INT_RTC0
14	R / 0	W / 0	RW / 0	MSYS_HPETO_CNT2	96	R / 0	W / 0	RW / 0	PRSYS_INT_RTC1
15	R / 0	W / 0	RW / 0	MSYS_HPET1_CNT0	97	R / 0	W / 0	RW / 0	PRSYS_INT_RTC2
16	R / 0	W / 0	RW / 0	MSYS_HPET1_CNT1	98	R / 0	W / 0	RW / 0	PRSYS_INT_TOY0
17	R / 0	W / 0	RW / 0	MSYS_HPET1_CNT2	99	R / 0	W / 0	RW / 0	PRSYS_INT_TOY1
18	R / 0	W / 0	RW / 0	MSYS_PM2IO	100	R / 0	W / 0	RW / 0	PRSYS_INT_TOY2
19 20 ~ 34	R / 0	W / 0	RW / 0	MSYS_PWM0 MSYS_PWM1 ~ MSYS_PWM15	101	R / 0	W / 0	RW / 0	PRSYS_PRTIFC
35	R / 0	W / 0	RW / 0	MSYS_AES	102	R / 0	W / 0	RW / 0	PRSYS_PM0IO
36	R / 0	W / 0	RW / 0	MSYS_DES	103	R / 0	W / 0	RW / 0	Reserved
37	R / 0	W / 0	RW / 0	MSYS_SM3	104	R / 0	W / 0	RW / 0	PRSYS_GPI00~23
38	R / 0	W / 0	RW / 0	MSYS_SM4	105	R / 0	W / 0	RW / 0	PRSYS_GPI024~47
39	R / 0	W / 0	RW / 0	MSYS_OTG	106	R / 0	W / 0	RW / 0	PRSYS_GPI048~57

位域	访问属性/缺省值			中断源	位域	访问属性/缺省值			中断源
	IEN	ICLR	POL			IEN	ICLR	POL	
40	R / 0	W / 0	RW / 0	MSYS_GMAC0	112	R / 0	W / 0	RW / 0	SCSYS_SCMLB
41	R / 0	W / 0	RW / 0	MSYS_GPIO00~03	113	R / 0	W / 0	RW / 0	SCSYS_PRMLB
42				MSYS_GPIO04~07					
~				~					
51				MSYS_GPIO40~43					
57	R / 0	W / 0	RW / 0	MSYS_SPIO	114	R / 0	W / 0	RW / 0	SCSYS_UART0
58	R / 0	W / 0	RW / 0	MSYS_EHCI	115	R / 0	W / 0	RW / 0	SCSYS_UART1
59	R / 0	W / 0	RW / 0	MSYS_OHCI	116	R / 0	W / 0	RW / 0	SCSYS_SPI
60	R / 0	W / 0	RW / 0	MSYS_GMAC1	117	R / 0	W / 0	RW / 0	SCSYS_PM1IO
61	R / 0	W / 0	RW / 0	MSYS_JBIG	118	R / 0	W / 0	RW / 0	SCSYS_HPET
62	R / 0	W / 0	RW / 0	MSYS_JPEG	119	R / 0	W / 0	RW / 0	SCSYS_SCIFC
63	R / 0	W / 0	RW / 0	MSYS_THSENS	120	R / 0	W / 0	RW / 0	SCSYS_GPIO0~7
64	R / 0	W / 0	RW / 0	MSYS_PRMLB	121	R / 0	W / 0	RW / 0	SCSYS_GPIO8~15
65	R / 0	W / 0	RW / 0	MSYS_SCMLB	122	R / 0	W / 0	RW / 0	SCSYS_GPIO16~23
66	R / 0	W / 0	RW / 0	MSYS_GMAC0_WK	123	R / 0	W / 0	RW / 0	SCSYS_GPIO24~31
80	R / 0	W / 0	RW / 0	PRSYS_PRMLB	124	R / 0	W / 0	RW / 0	SCSYS_GPIO32~36
81	R / 0	W / 0	RW / 0	PRSYS_SCMLB					

表 5- 97 打印系统 IO 中断控制寄存器属性

位域	访问属性/缺省值					中断源
	Intedge	Inten	Intclr	Intpol	Intdual	
0	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_PRMLB
1	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_SCMLB
2	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	Reserved
3	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_PWM00~03
4	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_PWM04~07
5	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_PWM08~11
6	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_PWM12~15
7	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_PWM16~19
8	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_PWM20~23
10	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_SPI
11	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_HPET_CNT0
12	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_HPET_CNT1
13	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_HPET_CNT2
14	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_I2C0
15	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_I2C1
16	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_UART0

位域	访问属性/缺省值					中断源
	Intedge	Inten	Intclr	Intpol	Intdual	
17	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_UART1
18	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_RTC_TICK
19	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_TOY_TICK
20	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_INT_RTC0
21	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_INT_RTC1
22	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_INT_RTC2
23	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_INT_TOY0
24	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_INT_TOY1
25	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_INT_TOY2
26	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_PRTIFC
27	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_PMOIO
28	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	Reserved
32 33 ~ 51	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_GPI000~02 PRSYS_GPI003~05 ~ PRSYS_GPI057
52~63	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	Reserved
64	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_UART00~03
65	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_I2C0~1
66	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_SPI0~1
67	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_HPET
68	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_PM2IO
69	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_PWM0~15
70	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_SDIO0~1
71	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_ SDIO0/1_DMA
72	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_ENCRYPT_DMA
73	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_ ENCRYPT_DEV
74	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_OTG
75	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_GMAC0
76	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_GMAC0_WK
77	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_THSENS
78	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_GPI00~31
79	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_GPI032~43
80	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	SCSYS_SCMLB
81	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	SCSYS_PRMLB
82	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	SCSYS_UART0

位域	访问属性/缺省值					中断源
	Intedge	Inten	Intclr	Intpol	Intdual	
83	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	SCSYS_UART1
84	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	SCSYS_SPI
85	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	SCSYS_PMIIO
86	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	SCSYS_HPET
87	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	SCSYS_SCIFC
88	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	SCSYS_GPI00~7
89	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	SCSYS_GPI08~15
90	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	SCSYS_GPI016~23
91	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	SCSYS_GPI024~31
92	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	SCSYS_GPI032~36
93	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	Reserved

表 5- 98 打印系统中断控制寄存器描述

中断寄存器名称	地址偏移	读写属性	初始值	寄存器描述
int0_src	0x0	R	0	0号中断状态寄存器，高电平有效
int0_en	0x4	R/W	0	0号中断使能寄存器，高电平有效
int0_clr	0xc	W	-	0号中断清除寄存器，写1清零，仅清除中断状态，不清除中断使能位
int0_pol	0x10	R/W	0	0号中断极性配置寄存器，0-低电平触发，1-高电平触发
int0_edge	0x14	R/W	0	0号中断边沿配置寄存器，0-电平中断，1-边沿中断
int0_dual	0x18	R/W	0	0号中断双边沿使能配置寄存器，高电平有效，仅在选择边沿触发时有效
Int1_src	0x20	R	0	1号中断状态寄存器，高电平有效
Int1_en	0x24	R/W	0	1号中断使能寄存器，高电平有效
Int1_clr	0x2c	W	-	1号中断清除寄存器，写1清零，仅清除中断状态，不清除中断使能位
Int1_pol	0x30	R/W	0	1号中断极性配置寄存器，0-低电平触发，1-高电平触发
Int1_edge	0x34	R/W	0	1号中断边沿配置寄存器，0-电平中断，1-边沿中断
Int1_dual	0x38	R/W	0	1号中断双边沿使能配置寄存器，高电平有效，仅在选择边沿触发时有效
Int2_src	0x40	R	0	2号中断状态寄存器，高电平有效
Int2_en	0x44	R/W	0	2号中断使能寄存器，高电平有效
Int2_clr	0x4c	W	-	2号中断清除寄存器，写1清零，仅清除中断状态，不清除中断使能位
Int2_pol	0x50	R/W	0	2号中断极性配置寄存器，0-低电平触发，



中断寄存器名称	地址偏移	读写属性	初始值	寄存器描述
				1-高电平触发
Int2_edge	0x54	R/W	0	2号中断边沿配置寄存器, 0-电平中断, 1-边沿中断
Int2_dual	0x58	R/W	0	2号中断双边沿使能配置寄存器, 高电平有效, 仅在选择边沿触发时有效
Int_map0 ~ Int_map5	0x60 ~ 0x74	R/W	0	96个中断位路由对应处理器核4个中断号路由配置寄存器, 192位配置, 每2位对应1个中断资源: 00:路由至处理器核0中断号; 01:路由至处理器核1中断号; 10:路由至处理器核2中断号; 11:路由至处理器核3中断号。

表 5- 99 扫描系统 IO 中断控制寄存器属性

位域	访问属性/缺省值					中断源
	Intedge	Inten	Intclr	Intpol	Intdual	
0	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	SCSYS_SCMLB
1	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	SCSYS_PRMLB
2	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	SCSYS_UART0
3	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	SCSYS_UART1
4	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	SCSYS_SPI
5	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	SCSYS_PMIIO
6	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	SCSYS_HPET_CNT0
7	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	SCSYS_HPET_CNT1
8	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	SCSYS_HPET_CNT2
9	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	SCSYS_SCIFC
10 11 ~ 46	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	SCSYS_GPI00 SCSYS_GPI01 ~ SCSYS_GPI036
47	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	Reserved
48	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_UART00~03
49	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_I2C0~1
50	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_SPI0~1
51	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_HPET
52	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_PM2IO
53	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_PWM0~15
54	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_SDIO0~1
55	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_SDIO0/1_DMA

位域	访问属性/缺省值					中断源
	Intedge	Inten	Intclr	Intpol	Intdual	
56	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_ENCRYPT_DMA
57	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_ENCRYPT_DEV
58	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_OTG
59	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_GMAC0
60	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_GMAC0_WK
61	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_THSENS
62	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_GPIO0~31
63	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	MSYS_GPIO32~43
64	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_PRMLB
65	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_SCMLB
66	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	Reserved
67	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_PWM0~7
68	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_PWM8~15
69	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_PWM16~23
70	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_SPI
71	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_HPET_CNT0
72	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_HPET_CNT1/2
73	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_I2C0
74	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_I2C1
75	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_UART0
76	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_UART1
77	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_RTC_TICK
78	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_TOY_TICK
79	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_INT_RTC0
80	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_INT_RTC1
81	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_INT_RTC2
82	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_INT_TOY0
83	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_INT_TOY1
84	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_INT_TOY2
85	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_PRTIFC
86	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_PMOIO
87	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	Reserved
88	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_GPIO0~23
89	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_GPIO24~47
90	RW / 0	RW / 0	W / 0	RW / 0	RW / 0	PRSYS_GPIO48~57

表 5- 100 扫描系统中断控制寄存器描述

中断寄存器名称	地址偏移	读写属性	初始值	寄存器描述
int0_src	0x0	R	0	0号中断状态寄存器，高电平有效
int0_en	0x4	R/W	0	0号中断使能寄存器，高电平有效
int0_clr	0xc	W	-	0号中断清除寄存器，写1清零，仅清除中断状态，不清除中断使能位
int0_pol	0x10	R/W	0	0号中断极性配置寄存器，0-低电平触发，1-高电平触发
int0_edge	0x14	R/W	0	0号中断边沿配置寄存器，0-电平中断，1-边沿中断
int0_dual	0x18	R/W	0	0号中断双边沿使能配置寄存器，高电平有效，仅在选择边沿触发时有效
Int1_src	0x20	R	0	1号中断状态寄存器，高电平有效
Int1_en	0x24	R/W	0	1号中断使能寄存器，高电平有效
Int1_clr	0x2c	W	-	1号中断清除寄存器，写1清零，仅清除中断状态，不清除中断使能位
Int1_pol	0x30	R/W	0	1号中断极性配置寄存器，0-低电平触发，1-高电平触发
Int1_edge	0x34	R/W	0	1号中断边沿配置寄存器，0-电平中断，1-边沿中断
Int1_dual	0x38	R/W	0	1号中断双边沿使能配置寄存器，高电平有效，仅在选择边沿触发时有效
Int2_src	0x40	R	0	2号中断状态寄存器，高电平有效
Int2_en	0x44	R/W	0	2号中断使能寄存器，高电平有效
Int2_clr	0x4c	W	-	2号中断清除寄存器，写1清零，仅清除中断状态，不清除中断使能位
Int2_pol	0x50	R/W	0	2号中断极性配置寄存器，0-低电平触发，1-高电平触发
Int2_edge	0x54	R/W	0	2号中断边沿配置寄存器，0-电平中断，1-边沿中断
Int2_dual	0x58	R/W	0	2号中断双边沿使能配置寄存器，高电平有效，仅在选择边沿触发时有效
Int_map0 ~ Int_map5	0x60 ~ 0x74	R/W	0	96个中断位路由对应处理器核4个中断号路由配置寄存器，192位配置，每2位对应1个中断资源： 00:路由至处理器核0中断号； 01:路由至处理器核1中断号； 10:路由至处理器核2中断号； 11:路由至处理器核3中断号。

### 5.5.4 中断路由寄存器描述

龙芯 2P0500 中断源可以选择路由到处理器核中断 INT0 到 INT3 中的任意一个，即对

应 CP0\_Status 的 IP0 到 IP3。主系统传统中断中 64 个 I/O 中断源中每一个都对应一个 8 位的路由控制器，其格式和地址如下表所示。路由寄存器采用向量的方式进行路由选择，如 0x40 表示路由到处理器的 INT2 上。

表 5- 101 传统中断路由寄存器的说明

位域	说 明
3:0	保留
7:4	路由的处理器核中断引脚向量号： 0001: 主系统处理器核 0 中断号； 0010: 主系统处理器核 1 中断号； 0100: 主系统处理器核 2 中断号； 1000: 主系统处理器核 3 中断号。

表 5- 102 传统中断路由寄存器地址

名称	地址偏移	描述	名称	地址偏移	描述
Entry0	0x14001400	MSYS_UART00	Entry32	0x14001440	PRSYS_PRMLB
Entry1	0x14001401	MSYS_UART01	Entry33	0x14001441	PRSYS_SCMLB
Entry2	0x14001402	MSYS_UART02	Entry34	0x14001442	Reserved
Entry3	0x14001403	MSYS_UART03	Entry35	0x14001443	PRSYS_PWM
Entry4	0x14001404	MSYS_I2C0	Entry36	0x14001444	PRSYS_SPI
Entry5	0x14001405	MSYS_I2C1	Entry37	0x14001445	PRSYS_HPET
Entry6	0x14001406	MSYS_SPI1	Entry38	0x14001446	PRSYS_I2C
Entry7	0x14001407	MSYS_GMAC0_WK	Entry39	0x14001447	PRSYS_UART
Entry8	0x14001408	MSYS_HPETO	Entry40	0x14001448	PRSYS_RTC_TICK
Entry9	0x14001409	MSYS_HPET1	Entry41	0x14001449	PRSYS_TOY_TICK
Entry10	0x1400140a	MSYS_PM2IO	Entry42	0x1400144a	PRSYS_INT_RTC
Entry11	0x1400140b	MSYS_PWM0~7	Entry43	0x1400144b	PRSYS_INT_TOY
Entry12	0x1400140c	MSYS_PWM8~15	Entry44	0x1400144c	PRSYS_PRTIFC
Entry13	0x1400140d	MSYS_SDI00~1	Entry45	0x1400144d	PRSYS_PM0IO
Entry14	0x1400140e	MSYS_SDI00/1-DMA	Entry46	0x1400144e	Reserved
Entry15	0x1400140f	MSYS_ENCRYPT-DMA	Entry47	0x1400144f	PRSYS_GPIO0~23
Entry16	0x14001410	MSYS_ENCRYPT-DEV	Entry48	0x14001450	PRSYS_GPIO24~47
Entry17	0x14001411	MSYS_OTG	Entry49	0x14001451	PRSYS_GPIO48~57
Entry18	0x14001412	MSYS_GMAC0	Entry50	0x14001452	Reserved
Entry19	0x14001413	MSYS_SPI0	Entry51	0x14001453	SCSYS_SCMLB
Entry20	0x14001414	MSYS_EHCI	Entry52	0x14001454	SCSYS_PRMLB

名称	地址偏移	描述	名称	地址偏移	描述
Entry21	0x14001415	MSYS_OHCI	Entry53	0x14001455	SCSYS_UART0
Entry22	0x14001416	MSYS_GMAC1	Entry54	0x14001456	SCSYS_UART1
Entry23	0x14001417	MSYS_JBIG	Entry55	0x14001457	SCSYS_SPI
Entry24	0x14001418	MSYS_JPEG	Entry56	0x14001458	SCSYS_PMIIO
Entry25	0x14001419	MSYS_THSENS	Entry57	0x14001459	SCSYS_HPET
Entry26	0x1400141a	MSYS_PRMLB	Entry58	0x1400145a	SCSYS_SCIFC
Entry27	0x1400141b	MSYS_SCMLB	Entry59	0x1400145b	SCSYS_GPI00~7
Entry28	0x1400141c	MSYS_GPI00~15	Entry60	0x1400145c	SCSYS_GPI08~15
Entry29	0x1400141d	MSYS_GPI016~31	Entry61	0x1400145d	SCSYS_GPI016~23
Entry30	0x1400141e	MSYS_GPI032~43	Entry62	0x1400145e	SCSYS_GPI024~31
Entry31	0x1400141f	Reserved	Entry63	0x1400145f	SCSYS_GPI032~36

表 5- 103 扩展中断路由寄存器的说明

位域	EXTIOI_MAP 寄存器说明
31:28	保留
27:24	EXT_IOI_ISR[127:96]统一路由的处理器核中断引脚向量号： 0001：主系统处理器核 0 中断号； 0010：主系统处理器核 1 中断号； 0100：主系统处理器核 2 中断号； 1000：主系统处理器核 3 中断号。
23:20	保留
19:16	EXT_IOI_ISR[95:64]统一路由的处理器核中断引脚向量号： 0001：主系统处理器核 0 中断号； 0010：主系统处理器核 1 中断号； 0100：主系统处理器核 2 中断号； 1000：主系统处理器核 3 中断号。
15:12	保留
11:8	EXT_IOI_ISR[63:32]统一路由的处理器核中断引脚向量号： 0001：主系统处理器核 0 中断号； 0010：主系统处理器核 1 中断号； 0100：主系统处理器核 2 中断号； 1000：主系统处理器核 3 中断号。
7:4	保留
3:0	EXT_IOI_ISR[31:0]统一路由的处理器核中断引脚向量号： 0001：主系统处理器核 0 中断号； 0010：主系统处理器核 1 中断号； 0100：主系统处理器核 2 中断号； 1000：主系统处理器核 3 中断号。

表 5- 104 打印系统中断路由寄存器的说明

位域	INT_MAP 寄存器说明
PRSYS_INT_MAP4[31:0]	打印系统共 96 个 IO 中断位，路由对应打印系统处理器核 4 个中断号路由配置寄存器，共计 5 组路由由配置寄存器 192 位路由配置，每 2 位对应 1 个中断资源，路由选择： 00: 路由至打印系统处理器核 0 中断号； 01: 路由至打印系统处理器核 1 中断号； 10: 路由至打印系统处理器核 2 中断号； 11: 路由至打印系统处理器核 3 中断号。
PRSYS_INT_MAP3[31:0]	
PRSYS_INT_MAP2[31:0]	
PRSYS_INT_MAP1[31:0]	
PRSYS_INT_MAP0[31:0]	

表 5- 105 扫描系统中断路由寄存器的说明

位域	INT_MAP 寄存器说明
SCSYS_INT_MAP4[31:0]	扫描系统共 96 个 IO 中断位，路由对应扫描系统处理器核 4 个中断号路由配置寄存器，共计 5 组路由由配置寄存器 192 位路由配置，每 2 位对应 1 个中断资源，路由选择： 00: 路由至扫描系统处理器核 0 中断号； 01: 路由至扫描系统处理器核 1 中断号； 10: 路由至扫描系统处理器核 2 中断号； 11: 路由至扫描系统处理器核 3 中断号。
SCSYS_INT_MAP3[31:0]	
SCSYS_INT_MAP2[31:0]	
SCSYS_INT_MAP1[31:0]	
SCSYS_INT_MAP0[31:0]	

# 6 DDR3 控制器

龙芯 2P0500 处理器内部集成的 DDR3 SDRAM 内存控制器。

## 6.1 访问地址

DDR3 控制器包括两个地址空间，分别如下：

表 6- 1 内存控制器地址空间分配

起始地址	结束名称	名称	说明
0x0FF0_0000	0x0FFF_FFFF	配置空间	当 mc_default_reg =1 时， 或 mc_default_reg = 0 且 mc_disable_reg = 0 时，为配置空间。 其它情况下为内存空间
其它		内存空间	使用 X2 的窗口配置路由至 DDR 的所有地址

具体的 mc\_default\_reg 和 mc\_disable\_reg 配置请参考主系统通用配置 0。功能概述

龙芯 2P0500 处理器支持 1 个 CS（由 1 个片选信号实现），一共含有 19 位的地址总线（即：16 位的行列地址总线和 3 位的逻辑 Bank 总线）。

在具体选择使用不同内存芯片类型时，可以调整 DDR3 控制器参数设置进行支持。其中，支持的最大片选（CS\_n）数为 1，行地址（RAS\_n）数为 16，列地址（CAS\_n）数为 16，逻辑体地址（BANK\_n）数为 3。

CPU 发送的内存请求物理地址可以根据控制器内部不同的配置进行多种不同的地址映射。

龙芯 2P0500 处理器中内存控制器具有如下特征：

1. 接口上命令、读写数据全流水操作
2. 内存命令合并、排序提高整体带宽
3. 配置寄存器读写端口，可以修改内存设备的基本参数
4. 内建动态延迟补偿电路（DCC），用于数据的可靠发送和接收
5. 支持 133-400MHZ 工作频率

## 6.2 DDR3 控制器寄存器

由于系统中可能使用不同类型的 DDR2/3 SDRAM，因此，在系统上电复位以后，需要对 DDR2/3 SDRAM 进行配置。在 JESD79-2B 和 JESD79-3 中规定了详细的配置操作和配置过程，在没有完成 DDR2/3 的内存初始化操作之前，DDR2/3 不可用。内存初始化操作执行顺序如下：

- 1) 系统复位，此时控制器内部所有寄存器内容将被清除为初始值。
- 2) 系统解复位。
- 3) 向配置寄存器地址发写指令，配置所有 DDR3 配置寄存器。所有寄存器都必须正确配置才可以正常工作。

4) 配置结束后内存控制器将自动对内存发起初始化指令。

在龙芯 2P0500 处理器设计中，DDR2/3 SDRAM 的配置在系统主板初始化完成以后，需要使用内存之前，进行内存类型的配置。具体的配置操作是对物理地址 0x0ff0 0000 相对应的配置寄存器写入相应的配置参数。一个寄存器可能会包括多个、一个、部分参数的数据。这些配置寄存器及其包含的参数意义如下表（寄存器中未使用的位均为保留位），具体的配置可以根据实际情况再决定：

表 6- 2 DDR3 SDRAM 配置参数寄存器

	63:56	55:48	47:40	39:32	31:24	23:16	15:8	7:0
0x000	Dll_value_0(RD)/Dll_adj_cnt/dll_sync_disable/dll_close_disable		Dll_value_ck(RD)		Dll_init_done(RD)		Version(RD)	
0x008	Dll_value_4(RD)		Dll_value_3(RD)		Dll_value_2(RD)		Dll_value_1(RD)/capability(RD)	
0x010	Dll_value_8(RD)		Dll_value_7(RD)		Dll_value_6(RD)		Dll_value_5(RD)	
0x018	Dll_ck_3	Dll_ck_2	Dll_ck_1	Dll_ck_0	Dll_increment	Dll_start_point	Dll_bypass	Init_start
0x020	Dq_oe_end_0	Dq_oe_begin_0	Dq_stop_edge_0	Dq_start_edge_0	Rddata_delay_0	Rddqs_lt_half_0	Wrdqs_lt_half_0	Wrdq_lt_half_0
0x028	Rd_oe_end_0	Rd_oe_begin_0	Rd_stop_edge_0	Rd_start_edge_0	Dqs_oe_end_0	Dqs_oe_begin_0	Dqs_stop_edge_0	Dqs_start_edge_0
0x030	Enzi_end_0	Enzi_begin_0	Wrcrk_sel_0	Wrdq_clkdelay_0	Odt_oe_end_0	Odt_oe_begin_0	Odt_stop_edge_0	Odt_start_edge_0
0x038	Enzi_stop_0	Enzi_start_0	Dll_oe_shorten_0	Dll_rddqs_n_0	Dll_rddqs_p_0	Dll_wrdqs_0	Dll_wrdata_0	Dll_gate_0
0x040	Dq_oe_end_1	Dq_oe_begin_1	Dq_stop_edge_1	Dq_start_edge_1	Rddata_delay_1	Rddqs_lt_half_1	Wrdqs_lt_half_1	Wrdq_lt_half_1
0x048	Rd_oe_end_1	Rd_oe_begin_1	Rd_stop_edge_1	Rd_start_edge_1	Dqs_oe_end_1	Dqs_oe_begin_1	Dqs_stop_edge_1	Dqs_start_edge_1
0x050	Enzi_end_1	Enzi_begin_1	Wrcrk_sel_1	Wrdq_clkdelay_1	Odt_oe_end_1	Odt_oe_begin_1	Odt_stop_edge_1	Odt_start_edge_1
0x058	Enzi_stop_1	Enzi_start_1	Dll_oe_shorten_1	Dll_rddqs_n_1	Dll_rddqs_p_1	Dll_wrdqs_1	Dll_wrdata_1	Dll_gate_1
0x060	Dq_oe_end_2	Dq_oe_begin_2	Dq_stop_edge_2	Dq_start_edge_2	Rddata_delay_2	Rddqs_lt_half_2	Wrdqs_lt_half_2	Wrdq_lt_half_2
0x068	Rd_oe_end_2	Rd_oe_begin_2	Rd_stop_edge_2	Rd_start_edge_2	Dqs_oe_end_2	Dqs_oe_begin_2	Dqs_stop_edge_2	Dqs_start_edge_2
0x070	Enzi_end_2	Enzi_begin_2	Wrcrk_sel_2	Wrdq_clkdelay_2	Odt_oe_end_2	Odt_oe_begin_2	Odt_stop_edge_2	Odt_start_edge_2
0x078	Enzi_stop_2	Enzi_start_2	Dll_oe_shorten_2	Dll_rddqs_n_2	Dll_rddqs_p_2	Dll_wrdqs_2	Dll_wrdata_2	Dll_gate_2
0x080	Dq_oe_end_3	Dq_oe_begin_3	Dq_stop_edge_3	Dq_start_edge_3	Rddata_delay_3	Rddqs_lt_half_3	Wrdqs_lt_half_3	Wrdq_lt_half_3
0x088	Rd_oe_end_3	Rd_oe_begin_3	Rd_stop_edge_3	Rd_start_edge_3	Dqs_oe_end_3	Dqs_oe_begin_3	Dqs_stop_edge_3	Dqs_start_edge_3
0x090	Enzi_end_3	Enzi_begin_3	Wrcrk_sel_3	Wrdq_clkdelay_3	Odt_oe_end_3	Odt_oe_begin_3	Odt_stop_edge_3	Odt_start_edge_3
0x098	Enzi_stop_3	Enzi_start_3	Dll_oe_shorten_3	Dll_rddqs_n_3	Dll_rddqs_p_3	Dll_wrdqs_3	Dll_wrdata_3	Dll_gate_3
0x0A0	Dq_oe_end_4	Dq_oe_begin_4	Dq_stop_edge_4	Dq_start_edge_4	Rddata_delay_4	Rddqs_lt_half_4	Wrdqs_lt_half_4	Wrdq_lt_half_4
0x0A8	Rd_oe_end_4	Rd_oe_begin_4	Rd_stop_edge_4	Rd_start_edge_4	Dqs_oe_end_4	Dqs_oe_begin_4	Dqs_stop_edge_4	Dqs_start_edge_4
0x0B0	Enzi_end_4	Enzi_begin_4	Wrcrk_sel_4	Wrdq_clkdelay_4	Odt_oe_end_4	Odt_oe_begin_4	Odt_stop_edge_4	Odt_start_edge_4
0x0B8	Enzi_stop_4	Enzi_start_4	Dll_oe_shorten_4	Dll_rddqs_n_4	Dll_rddqs_p_4	Dll_wrdqs_4	Dll_wrdata_4	Dll_gate_4
0x0C0	Dq_oe_end_5	Dq_oe_begin_5	Dq_stop_edge_5	Dq_start_edge_5	Rddata_delay_5	Rddqs_lt_half_5	Wrdqs_lt_half_5	Wrdq_lt_half_5
0x0C8	Rd_oe_end_5	Rd_oe_begin_5	Rd_stop_edge_5	Rd_start_edge_5	Dqs_oe_end_5	Dqs_oe_begin_5	Dqs_stop_edge_5	Dqs_start_edge_5
0x0D0	Enzi_end_5	Enzi_begin_5	Wrcrk_sel_5	Wrdq_clkdelay_5	Odt_oe_end_5	Odt_oe_begin_5	Odt_stop_edge_5	Odt_start_edge_5
0x0D8	Enzi_stop_5	Enzi_start_5	Dll_oe_shorten_5	Dll_rddqs_n_5	Dll_rddqs_p_5	Dll_wrdqs_5	Dll_wrdata_5	Dll_gate_5
0x0E0	Dq_oe_end_6	Dq_oe_begin_6	Dq_stop_edge_6	Dq_start_edge_6	Rddata_delay_6	Rddqs_lt_half_6	Wrdqs_lt_half_6	Wrdq_lt_half_6
0x0E8	Rd_oe_end_6	Rd_oe_begin_6	Rd_stop_edge_6	Rd_start_edge_6	Dqs_oe_end_6	Dqs_oe_begin_6	Dqs_stop_edge_6	Dqs_start_edge_6
0x0F0	Enzi_end_6	Enzi_begin_6	Wrcrk_sel_6	Wrdq_clkdelay_6	Odt_oe_end_6	Odt_oe_begin_6	Odt_stop_edge_6	Odt_start_edge_6
0x0F8	Enzi_stop_6	Enzi_start_6	Dll_oe_shorten_6	Dll_rddqs_n_6	Dll_rddqs_p_6	Dll_wrdqs_6	Dll_wrdata_6	Dll_gate_6
0x100	Dq_oe_end_7	Dq_oe_begin_7	Dq_stop_edge_7	Dq_start_edge_7	Rddata_delay_7	Rddqs_lt_half_7	Wrdqs_lt_half_7	Wrdq_lt_half_7



	63:56	55:48	47:40	39:32	31:24	23:16	15:8	7:0
0x108	Rd_oe_end_7	Rd_oe_begin_7	Rd_stop_edge_7	Rd_start_edge_7	Dqs_oe_end_7	Dqs_oe_begin_7	Dqs_stop_edge_7	Dqs_start_edge_7
0x110	Enzi_end_7	Enzi_begin_7	Wrclk_sel_7	Wrclk_sel_7	Wrclk_sel_7	Wrclk_sel_7	Wrclk_sel_7	Wrclk_sel_7
0x118	Enzi_stop_7	Enzi_start_7	Dll_oe_shorten_7	Dll_rddqs_n_7	Dll_rddqs_p_7	Dll_wrdds_7	Dll_wrdata_7	Dll_gate_7
0x120	Dq_oe_end_8	Dq_oe_begin_8	Dq_stop_edge_8	Dq_start_edge_8	Rddata_delay_8	Rddqs_lt_half_8	Wrdds_lt_half_8	Wrddq_lt_half_8
0x128	Rd_oe_end_8	Rd_oe_begin_8	Rd_stop_edge_8	Rd_start_edge_8	Dqs_oe_end_8	Dqs_oe_begin_8	Dqs_stop_edge_8	Dqs_start_edge_8
0x130	Enzi_end_8	Enzi_begin_8	Wrclk_sel_8	Wrclk_sel_8	Wrclk_sel_8	Wrclk_sel_8	Wrclk_sel_8	Wrclk_sel_8
0x138	Enzi_stop_8	Enzi_start_8	Dll_oe_shorten_8	Dll_rddqs_n_8	Dll_rddqs_p_8	Dll_wrdds_8	Dll_wrdata_8	Dll_gate_8
0x140	Pad_ocd_clk	Pad_ocd_ctl	Pad_ocd_dqs	Pad_ocd_dq	Pad_enzi		Pad_en_ctl	Pad_en_clk
0x148	Pad_adj_code_dqs	Pad_code_dqs	Pad_adj_code_dq	Pad_code_dq		Pad_vref_internal	Pad_odt_se	Pad_modezi1v8
0x150		Pad_reset_po	Pad_adj_code_clk	Pad_code_lk	Pad_adj_code_cmd	Pad_code_cmd	Pad_adj_code_addr	Pad_code_addr
0x158		Pad_comp_code_o	Pad_comp_okn		Pad_comp_code_i	Pad_comp_mode	Pad_comp_tm	Pad_comp_pd
0x160	Rdfifo_empty(RD)		Overflow(RD)		Dram_init(RD)	Rdfifo_valid	Cmd_timming	Ddr3_mode
0x168	Ba_xor_row_offset	Addr_mirror	Cmd_delay	Burst_length	Bank/Cs_resync	Cs_zq	Cs_mrs	Cs_enable
0x170	Odt_wr_cs_map		Odt_wr_length	Odt_wr_delay	Odt_rd_cs_map		Odt_rd_length	Odt_rd_delay
0x178								
0x180	Lvl_resp_0(RD)	Lvl_done(RD)	Lvl_ready(RD)		Lvl_cs	tLVL_DELAY	Lvl_req(WR)	Lvl_mode
0x188	Lvl_resp_8(RD)	Lvl_resp_7(RD)	Lvl_resp_6(RD)	Lvl_resp_5(RD)	Lvl_resp_4(RD)	Lvl_resp_3(RD)	Lvl_resp_2(RD)	Lvl_resp_1(RD)
0x190	Cmd_a		Cmd_ba	Cmd_cmd	Cmd_cs	Status_emd(RD)	Cmd_req(WR)	Command_mode
0x198			Status_sref(RD)	Srefresh_req	Pre_all_done(RD)	Pre_all_req(RD)	Mrs_done(RD)	Mrs_req(WR)
0x1A0	Mr_3_cs_0		Mr_2_cs_0		Mr_1_cs_0		Mr_0_cs_0	
0x1A8	Mr_3_cs_1		Mr_2_cs_1		Mr_1_cs_1		Mr_0_cs_1	
0x1B0	Mr_3_cs_2		Mr_2_cs_2		Mr_1_cs_2		Mr_0_cs_2	
0x1B8	Mr_3_cs_3		Mr_2_cs_3		Mr_1_cs_3		Mr_0_cs_3	
0x1C0	tRESET	tCKE	tXPR	tMOD	tZQCL	tZQ_CMD	tWLDQSEN	tRDDATA
0x1C8	tFAW	tRRD	tRCD	tRP	tREF	tRFC	tZQCS	tZQperiod
0x1D0	tODTL	tXSRD	tPHY_RDLAT	tPHY_WRLAT	tRAS_max			tRAS_min
0x1D8	tXPDLL	tXP	tWR	tRTP	tRL	tWL	tCCD	tWTR
0x1E0	tW2R_diffCS	tW2W_diffCS	tR2P_sameBA	tW2P_sameBA	tR2R_sameBA	tR2W_sameBA	tW2R_sameBA	tW2W_sameBA
0x1E8	tR2R_diffCS	tR2W_diffCS	tR2P_sameCS	tW2P_sameCS	tR2R_sameCS	tR2W_sameCS	tW2R_sameCS	tW2W_sameCS
0x1F0	Power_up	Age_step	tCPDED	Cs_map	Bs_config	Nc	Pr_r2w	Placement_en
0x1F8	Hw_pd_3	Hw_pd_2	Hw_pd_1	Hw_pd_0	Credit_16	Credit_32	Credit_64	Selection_en
0x200	Cmdq_age_16		Cmdq_age_32		Cmdq_age_64		tCKESR	tRDPDEN
0x208	Wfifo_age		Rfifo_age		Power_stat3	Power_stat2	Power_stat1	Power_stat0
0x210	Active_age		Cs_place_0	Addr_win_0	Cs_diff_0	Row_diff_0	Ba_diff_0	Col_diff_0
0x218	Fastpd_age		Cs_place_1	Addr_win_1	Cs_diff_1	Row_diff_1	Ba_diff_1	Col_diff_1
0x220	Slowpd_age		Cs_place_2	Addr_win_2	Cs_diff_2	Row_diff_2	Ba_diff_2	Col_diff_2
0x228	Selfref_age		Cs_place_3	Addr_win_3	Cs_diff_3	Row_diff_3	Ba_diff_3	Col_diff_3
0x230	Win_mask_0				Win_base_0			
0x238	Win_mask_1				Win_base_1			
0x240	Win_mask_2				Win_base_2			
0x248	Win_mask_3				Win_base_3			
0x250		Cmd_monitor	Axi_monitor		Ecc_code(RD)	Ecc_enable	Int_vector	Int_enable

	63:56	55:48	47:40	39:32	31:24	23:16	15:8	7:0
0x258								
0x260	Ecc_addr(RD)							
0x268	Ecc_data(RD)							
0x270	Lpbk_ecc_mask(RD)	Prbs_init			Lpbk_error(RD)	Prbs_23	Lpbk_start	Lpbk_en
0x278	Lpbk_ecc(RD)		Lpbk_data_mask(RD)		Lpbk_correct(RD)		Lpbk_counter(RD)	
0x280	Lpbk_data_r(RD)							
0x288	Lpbk_data_f(RD)							
0x290	Axi0_bandwidth_w				Axi0_bandwidth_r			
0x298	Axi0_latency_w				Axi0_latency_r			
0x2A0	Axi1_bandwidth_w				Axi1_bandwidth_r			
0x2A8	Axi1_latency_w				Axi1_latency_r			
0x2B0	Axi2_bandwidth_w				Axi2_bandwidth_r			
0x2B8	Axi2_latency_w				Axi2_latency_r			
0x2C0	Axi3_bandwidth_w				Axi3_bandwidth_r			
0x2C8	Axi3_latency_w				Axi3_latency_r			
0x2D0	Axi4_bandwidth_w				Axi4_bandwidth_r			
0x2D8	Axi4_latency_w				Axi4_latency_r			
0x2E0	Cmdq0_bandwidth_w				Cmdq0_bandwidth_r			
0x2E8	Cmdq0_latency_w				Cmdq0_latency_r			
0x2F0	Cmdq1_bandwidth_w				Cmdq1_bandwidth_r			
0x2F8	Cmdq1_latency_w				Cmdq1_latency_r			
0x300	Cmdq2_bandwidth_w				Cmdq2_bandwidth_r			
0x308	Cmdq2_latency_w				Cmdq2_latency_r			
0x310	Cmdq3_bandwidth_w				Cmdq3_bandwidth_r			
0x318	Cmdq3_latency_w				Cmdq3_latency_r			
0x320	tRESYNC_length	tRESYNC_shift	tRESYNC_max	tRESYNC_min	Pre_predict		tXS	tREF_low
0x328								tRESYNC_delay
0x330	Stat_en	Rdbuffer_max	Retry	Wr_pkg_num	Rwq_rb	Stb_en	Addr_new	tRDQidle
0x338				Rd_fifo_depth	Retry_cnt			
0x340	tREFretention					Ref_num	tREF_IDLE	Ref_sch_en
0x348								
0x350	Lpbk_data_en							
0x358						Lpbk_ecc_mask_en	Lpbk_ecc_en	Lpbk_data_mask_en
0x360			Int_ecc_cnt_fatal	Int_ecc_cnt_error	Ecc_cnt_cs_3	Ecc_cnt_cs_2	Ecc_cnt_cs_1	Ecc_cnt_cs_0
0x368								
0x370	Prior_age3		Prior_age2		Prior_age1		Prior_age_0	
0x378							No_dead_inorder	Row_hit_place
0x380	Zq_cnt_1				Zq_cnt_0			
0x388	Zq_cnt_3				Zq_cnt_2			
0x390								Nc16_map
0x398								

# 7 GMAC 控制器

## 7.1 寄存器描述

龙芯 2P0500 集成了两个 GMAC 控制器，即 GMAC0 和 GMAC1，二者在逻辑结构上完全相同。以下文档部分在未区分二者的情况下，表示该部分说明对二者均适用。

GMAC 控制器寄存器包括 GMAC 寄存器部分和 DMA 寄存器部分。

GMAC0 的 GMAC 寄存器的起始地址是 0x1414\_0000；GMAC0 的 DMA 寄存器的起始地址是 0x1414\_1000。

GMAC1 的 GMAC 寄存器的起始地址是 0x1402\_0000；GMAC1 的 DMA 寄存器的起始地址是 0x1402\_0000。

## 7.2 软件编程向导：

### DMA 初始化：

1. 软件重置(reset)GMAC
2. 等待重置完成(查询 DMA reg0[0])
3. 对 DMA reg0 的以下域进行编程
  - a) MIX-BURST 和 AAL(DMA reg0[26]、[25])
  - b) Fixed-burst 或者 undefined-burst(DMA reg0[16])
  - c) Burst-length 和 Burst-mode
  - d) Descriptor Length(只有当环形格式时有效)
  - e) Tx 和 Rx 仲裁调度
4. 对 AXI Bus Mode Reg 进行编程
  - a) 如果选择了 Fixed-burst，则需要在该寄存器内设置最大 burst length
5. 分别创建发送、接收描述符链，可以分别选择环形模式或者链型模式进行连接，并将接收描述符的 OWN 位设为 1(DMA 拥有)
6. 在软件启用 DMA 描述符之前，必须保证至少发送/接收描述符链中有三个描述符
7. 将发送、接收描述符链表的首地址写入 DMA reg3、4
8. 对 DMA reg6(DMA mode operation)中的以下位进行配置
  - a) 接收/发送的 Store and Forward
  - b) 接收/发送的阈值因子(Threshold Control)
  - c) 启用流控制(hardware flow control enable)
  - d) 错误帧和未识别的正确帧略过(forwarding enable)

- e) OSF 模式
- 9. 向 DMA reg6(Status reg)写 1. 清除所有中断请求
- 10. 向 DMA reg7(interrupt enable reg)写 1, 启用所有中断
- 11. 向 DMA reg6[1]、[13]中写 1, 启用发送和接收 DMA。

**MAC 初始化:**

1. 正确配置配套 PHY 芯片
2. 对 GMAC reg4(GMII Address Register)进行正确配置, 使其能够正常访问 PHY 相关寄存器
3. 读取 GMAC reg5(GMII Data Register)获取当前 PHY 的链接(link)、速度(speed)、模式(双工)等信息
4. 配置 MAC 地址
5. 如果启用了 hash filtering, 则需要对 hash filtering 进行配置
6. 对 GMAC reg1(Mac Frame filter)以下域进行配置, 来进行帧过滤
  - a) 接收所有
  - b) 混杂模式(promiscuous mode)
  - c) 哈希或完美过滤(hash or perfect filter)
  - d) 组播、多播过滤设置等等
7. 对 GMAC reg6(Flow control register)以下域进行配置
  - a) 暂停时间和其他暂停控制位
  - b) 接收和发送流控制位
  - c) 流控制忙/后压力启用
8. 对中断掩码寄存器(Mac reg15)进行配置
9. 基于之前得到的线路信息(link, speed, mode)对 GMAC reg0 进行正确的配置
10. 设置 GMAC reg0[2]、[3]来启用 GMAC 中的发送、接收模块。

**发送和接收的一般过程:**

1. 检测到发送或接收中断后, 查寻相应描述符来判断其是否属于主机, 并读取描述符中的数据
2. 完成对描述符中数据的读取后, 将描述符各位清 0 并设置其 OWN 位, 使其继续发送/接收数据
3. 如果当前发送或接收描述符不属于 DMA (OWN=0), 则 DMA 模块会进入挂起状态。当有数据需要被发送或接收时, 向 DMA Tx/Rx POLL 寄存器写 1 重新使能 DMA 模块。需要注意的是接收描述符在空闲时应该总是属于 DMA (OWN=1)
4. 发送和接收描述符及对应 buffer 地址的实时信息可以通过查寻 DMA reg18、19、20、21 获得。

# 8 USB 控制器

## 8.1 总体概述

2P0500 的 USB 主机端口特性如下：

- 兼容 USB Rev 1.1、USB Rev 2.0 协议
- 兼容 OHCI Rev 1.0、EHCI Rev 1.0 协议
- 支持 LS (Low Speed)、FS (Full Speed) 和 HS (High Speed) 的 USB 设备
- 支持三个 USB2.0 端口，每个端口都可挂 LS、FS 或 HS 设备

USB 主机控制器模块包括一个支持高速设备的 EHCI 控制器，一个支持全速与低速设备的 OHCI 控制器。其中 EHCI 控制器处于主控地位，只有当挂上的设备是全速或低速设备时，才将控制权转交给 OHCI 控制器；当全速或低速设备拔掉时，控制权返回 EHCI 控制器。

## 8.2 控制器寄存器

表 8- 1 USB 控制器地址空间分布

地址空间	名称	大小
0x1403,0000 - 0x1403,7fff	EHCI 寄存器	32KB
0x1403,8000 - 0x1403,ffff	OHCI 寄存器	32KB

# 9 OTG 控制器

## 9.1 概述

2P0500 的 OTG 支持特性如下:

- 支持 HNP 与 SRP 协议;
- 内嵌 DMA, 无需占用处理器带宽即可在 OTG 与外部存储之间移动数据;
- 在 device 模式下, 为高速设备 (480Mbps);
- 在 host 模式下, 仅能支持高速设备 (480Mbps);
- 在 device 模式下, 支持 10 个双向的 endpoint, 其中仅有默认的 endpoint0 支持控制传输;
- 在 device 模式下, 最多同时支持 4 个 IN 方向的传输;
- 在 host 模式下, 支持 12 个 channel, 且软件可配置每个 channel 的方向;
- 在 host 模式下, 支持 periodic OUT 传输;

## 9.2 寄存器描述

应用程序通过 AHB slave 接口来读写 OTG 控制器里的控制与状态寄存器 (CSRs), 这些寄存器都是 32 位宽, 寄存器地址为 32 位对齐。

这里需要注意的是, 在 host 模式与 device 模式下都能访问的寄存器组仅包括全局寄存器组 (Core Global), 功耗与门控时钟 (Power and Clock Gating) 寄存器组, 数据 FIFO 访问 (Data FIFO Access) 寄存器组, 端口 (Host Port) 寄存器组。当 OTG 控制器处于 host 或 device 模式下, 不能访问另一个模式下的寄存器。如果发生非法的寄存器读写, 将发生模式不匹配 (Mode Mismatch) 中断, 这个中断将反映在中断寄存器中 (Core Interrupt Register)。

当 OTG 从一个模式转换到另一个模式时, 必须重新配置这个模式下的寄存器, 因为这些寄存器在转换后的状态与上电重启时是一样的。

# 10 SPI-FLASH 控制器

串行外围设备接口 SPI 总线技术是 Motorola 公司推出的多种微处理器、微控制器以及外围设备之间的一种全双工、同步、串行数据接口标准。

## 10.1 SPI 控制器结构

本系统集成的 SPI0-flash 控制器仅可作为主控端，所连接的是从设备。2P0500 共集成 4 个 SPI 控制器，其中 SPI0 支持 IO/memory 两种空间访问，SPI1~3 仅支持 IO 空间访问。对于软件而言，SPI 控制器除了有若干 IO 寄存器外还有一段映射到 SPI Flash 的只读 memory 空间。如果将这段 memory 空间分配在 0x1c000000，复位后不需要软件干预就可以直接访问，从而支持处理器从 SPI Flash 启动，该启动功能仅 SPI0 控制器支持。2P0500 芯片 4 个 SPI 控制器空间分布，如下表。

表 10- 1 SPI 控制器地址空间分布

地址空间	名称	大小
0x1c00,0000 - 0x1c0f,ffff	SPI0 Boot	1MB
0x1000,0000 - 0x13ff,ffff	SPI0 MEM	64MB
0x1401,0000 - 0x1401,ffff	SPI0 IO/CFG	64KB
0x1420,3000 - 0x1420,3fff	SPI1 IO	4KB
0x1510,9000 - 0x1510,9fff	SPI2 IO	4KB
0x1520,3000 - 0x1520,3fff	SPI3 IO	4KB

对于 SPI 接口模块，使用时要注意将对应的芯片复用引脚设置为相应的接口功能。

与 SPI 接口相关的引脚复用设置可查询节 2.19 中的外设功能引脚复用关系表，并根据复用配置寄存器配置相应的 GPIO 引脚复用关系，实现对应设备功能引脚。

其结构如图10- 1所示，由 AXI/APB 内部总线接口、简单的 SPI 主控制器、SPI Flash 读引擎和总线选择模块组成。根据访问的地址和类型，来自内部总线接口上的合法请求转发到 SPI 主控制器或者 SPI Flash 读引擎中(非法请求被丢弃)。

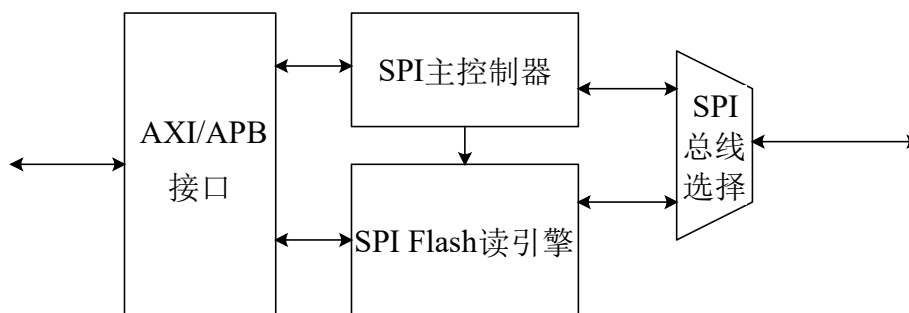


图10- 1 SPI 控制器结构

## 10.2 配置寄存器

表 10- 2 SPI0 配置寄存器列表

偏移	名称	描述	备注
0	SPCR	控制寄存器	SPI0 控制器支持
1	SPSR	状态寄存器	SPI0 控制器支持
2	TxFIFO/RxFIFO	数据寄存器	SPI0 控制器支持
3	SPER	外部寄存器	SPI0 控制器支持
4	SFC_PARAM	参数控制寄存器	SPI0 控制器支持
5	SFC_SOFTCS	片选控制寄存器	SPI0 控制器支持
6	SFC_TIMING	时序控制寄存器	SPI0 控制器支持

### 10.2.1 控制寄存器(SPCR)

表 10- 3 SPI 控制寄存器 (SPCR)

位域	名称	访问	初值	描述
7	spie	R/W	0	中断输出使能信号 高有效
6	spe	R/W	0	系统工作使能信号高有效
5	-	-	0	保留
4	mstr	-	1	master 模式选择位, 此位一直保持 1
3	cpol	R/W	0	时钟极性位
2	cpha	R/W	0	时钟相位位 1 则相位相反, 为 0 则相同
1:0	spr	R/W	0	sclk_o 分频设定, 需要与 sper 的 spre 一起使用

### 10.2.2 状态寄存器(SPSR)

表 10- 4 SPI 状态寄存器 (SPSR)

位域	名称	访问	初值	描述
7	spif	R/W	0	中断标志位 1 表示有中断申请, 写 1 则清零
6	wcol	R/W	0	写寄存器溢出标志位 为 1 表示已经溢出, 写 1 则清零
5:4	-	-	0	保留
3	wffull	R	0	写寄存器满标志 1 表示已经满
2	wfempty	R	1	写寄存器空标志 1 表示空
1	rffull	R	0	读寄存器满标志 1 表示已经满
0	rfempty	R	1	读寄存器空标志 1 表示空

### 10.2.3 数据寄存器(TxFIFO/RxFIFO)

表 10- 5 SPI 数据寄存器 (TxFIFO/RXFIFO)

位域	名称	访问	初值	描述
7:0	TxFIFO	W	-	数据发送端口
	RxFIFO	R		数据接收端口

### 10.2.4 外部寄存器(SPER)



表 10- 6 SPI 外部寄存器 (SPER)

位域	名称	访问	初值	描述
7:6	icnt	R/W	0	传输完多少个字节后发中断 00: 1 01: 2 10: 3 11: 4
5:3	-	-	-	保留
2	mode	R/W	0	spl 接口模式控制 0: 采样与发送时机同时 1: 采样与发送时机错开半周期
1:0	spre	R/W	0	与 spr 一起设定分频的比率

表 10- 7 SPI 分频系数

spre	00	00	00	00	01	01	01	01	10	10	10	10
spr	00	01	10	11	00	01	10	11	00	01	10	11
分频系数	2	4	16	32	8	64	128	256	512	1024	2048	4096

### 10.2.5 参数控制寄存器(SFC\_PARAM)

表 10- 8 SPI 参数控制寄存器 (SFC\_PARAM)

位域	名称	访问	初值	描述
7:4	clk_div	R/W	2	时钟分频数选择 分频系数与 {spre, spr} 组合相同
3	dual_io	R/W	0	双 I/O 模式, 优先级高于快速读
2	fast_read	R/W	0	快速读模式
1	burst_en	R/W	0	SPI flash 支持连续地址读模式
0	memory_en	R/W	1	SPI flash 读使能, 无效时 csn[0] 可由软件控制。

### 10.2.6 片选控制寄存器(SFC\_SOFTCS)

表 10- 9 SPI 片选控制寄存器 (SFC\_SOFTCS)

位域	名称	访问	初值	描述
7:4	csn	R/W	0	csn 引脚输出值
3:0	csen	R/W	0	为 1 时对应的 csn 线由 7:4 位控制

### 10.2.7 时序控制寄存器(SFC\_TIMING)

表 10- 10 SPI 时序控制寄存器 (SFC\_TIMING)

位域	名称	访问	初值	描述
7:3	-	-	-	保留
2	tFAST	R/W	0	SPI flash 读采样模式 0: 上沿采样, 间隔半个 SPI 周期 1: 上沿采样, 间隔一个 SPI 周期
1:0	tCSH	R/W	3	SPI Flash 的片选信号最短无效时间, 以分频后时钟周期 T 计算 00: 1T 01: 2T 10: 4T 11: 8T

## 10.3 接口时序

### 10.3.1 SPI 主控制器接口时序

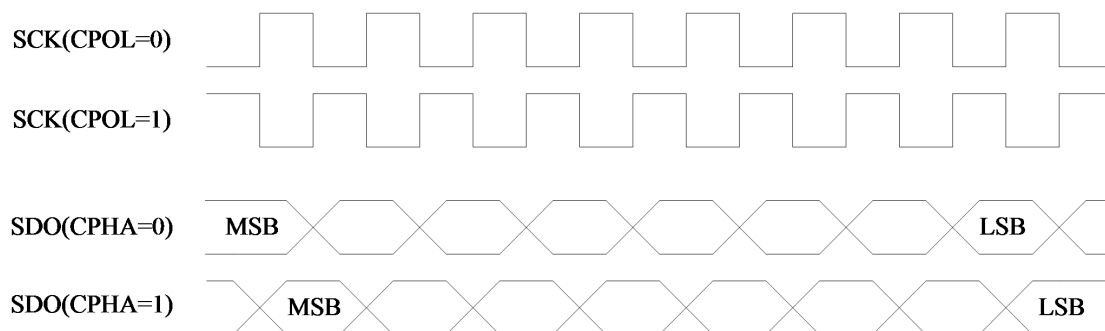


图10- 2 SPI 主控制器接口时序

### 10.3.2 SPI Flash 访问时序

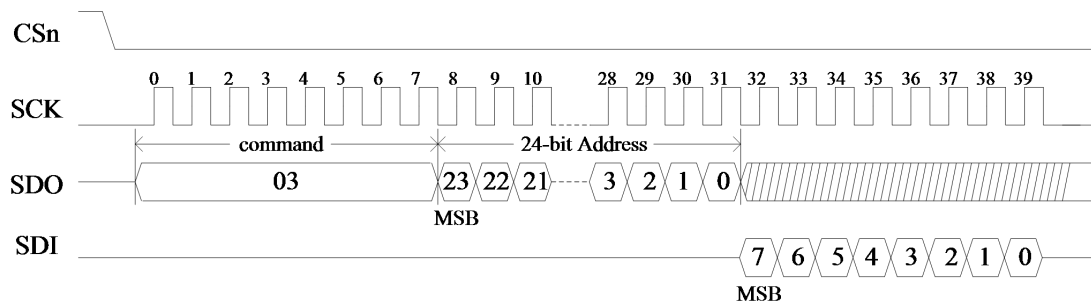


图10- 3 SPI Flash 标准读时序

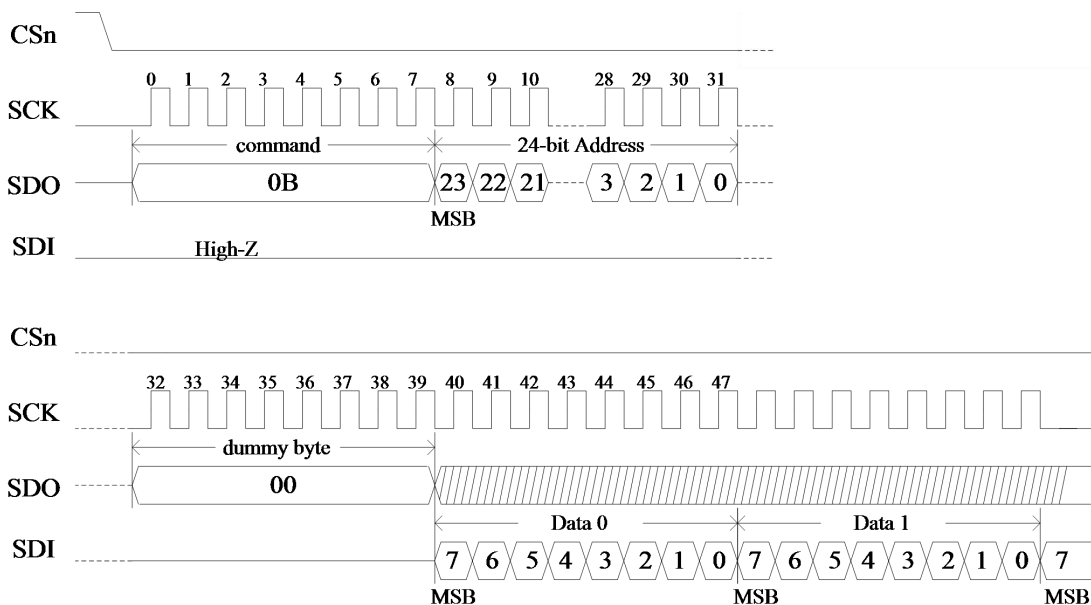


图10- 4 SPI Flash 快速读时序

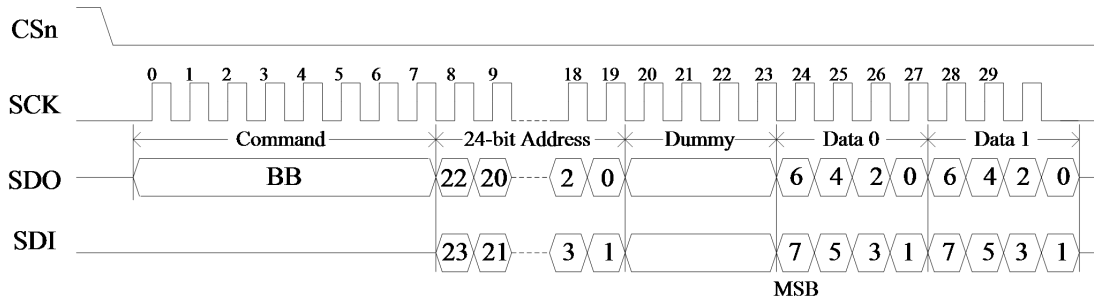


图10- 5 SPI Flash 双向 I/O 读时序

## 10.4 使用指南

### 10.4.1 SPI 主控制器的读写操作

#### 1. 模块初始化

- 停止 SPI 控制器工作，对控制寄存器 `sPCR` 的 `spe` 位写 0
- 重置状态寄存器 `sPSR`，对寄存器写入 `8'b1100_0000`
- 设置外部寄存器 `sPER`，包括中断申请条件 `sPER[7:6]` 和分频系数 `sPER[1:0]`，具体参考寄存器说明
- 配置 SPI 时序，包括 `sPCR` 的 `cpol`、`cpha` 和 `sPER` 的 `mode` 位。`mode` 为 1 时是标准 SPI 实现，为 0 时为兼容模式。
- 配置中断使能，`sPCR` 的 `spie` 位
- 启动 SPI 控制器，对控制寄存器 `sPCR` 的 `spe` 位写 1。

#### 2. 模块的发送/传输操作

- 往数据传输寄存器写入数据
- 传输完成后从数据传输寄存器读出数据。由于发送和接收同时进行，即使 SPI 从设备没有发送有效数据也必须进行读出操作。

#### 3. 中断处理

- 接收到中断申请
- 读状态寄存器 `sPSR` 的值，若 `sPSR[2]` 为 1 则表示数据发送完成，若 `sPSR[0]` 为 1 则表示已经接收数据
- 读或写数据传输寄存器
- 往状态寄存器 `sPSR` 的 `spif` 位写 1，清除控制器的中断申请。

### 10.4.2 硬件 SPI Flash 读

#### 1. 初始化

- 将 `SFC_PARAM` 的 `memory_en` 位写 1。当 SPI 被选为启动设备时此位复位为 1

- 设置读参数(时钟分频、连续地址读、快速读、双 I/O、tCSH 等)。这些参数复位值均为最保守的值。

## 2. 更改参数

如果所使用的 SPI Flash 支持更高的频率或者提供增强功能，修改相应参数可以大大加快 Flash 的访问速度。参数的修改不需要关闭 SPI Flash 读使能(memory\_en)。具体参考寄存器说明。

### 10.4.3 混合访问 SPI Flash 和 SPI 主控制器

#### 1. 对 SPI Flash 进行读以外的访问

将 SPI Flash 读使能关闭后，软件就可直接控制 csn[0]，并通过 SPI 主控制器访问 SPI 总线。这意味着在进行此操作时，不能从 SPI Flash 中取指。

除了读以外，SPI Flash 还实现了很多命令(如擦除、写入)，具体参见相关 Flash 的文档。

# 11 SPI-IO 控制器

串行外围设备接口 SPI 总线技术是 Motorola 公司推出的多种微处理器、微控制器以及外围设备之间的一种全双工、同步、串行数据接口标准。

## 11.1 SPI 控制器结构

2P0500 共集成 4 个 SPI 控制器，其中 SPI0 支持 IO/memory 两种空间访问，SPI1~3 仅支持 IO 空间访问。2P0500 芯片 3 个 SPI-IO 控制器空间分布，如下表。

表 11- 1 SPI 控制器地址空间分布

地址空间	名称	大小
0x1420,3000 - 0x1420,3fff	SPI1 IO	4KB
0x1510,9000 - 0x1510,9fff	SPI2 IO	4KB
0x1520,3000 - 0x1520,3fff	SPI3 IO	4KB

## 11.2 配置寄存器

表 11- 2 SPI-IO 配置寄存器列表

偏移	名称	描述
0x00	CR1	控制寄存器 1
0x04	CR2	控制寄存器 2
0x08	CR3	控制寄存器 3
0x0c	CR4	控制寄存器 4
0x10	IER	中断寄存器
0x14	SR1	状态寄存器 1
0x18	SR2	状态寄存器 2
0x20	CFG1	配置寄存器 1
0x24	CFG2	配置寄存器 2
0x28	CFG3	配置寄存器 3
0x30	CRC1	CRC 寄存器 1
0x34	CRC2	CRC 寄存器 2
0x40	DR	数据寄存器

### 11.2.1 控制寄存器 1(CR1)

表 11- 3 SPI 控制寄存器 1 (CR1)

位域	名称	访问	初值	描述
31: 9	-	-	0	保留
8	SSREV	R/W	0	SS REVerse- 片选信号翻转 0: 片选 (SS) 输入输出正常 1: 片选 (SS) 输入输出翻转

位域	名称	访问	初值	描述
7:3	-	-	0	保留
2	AUTOSUS	R/W	0	AUTOSUSpend- 自动挂起 0: 在当前传输结束后挂起 1: 在当前传输结束后重新向 CTSIZE 载入 TSIZE, 继续传输 注: 仅在主模式下使用。当前传输结束即: CTSIZE=0 且当前帧结束
1	CSTART	R/W	0	Control START of transfer- 传输开始控制 0: 在当前帧结束后挂起 1: 开始传输 注: 仅在主模式下使用。当 AUTOSUS=1 时, 会在 CTSIZE=0 时清 0
0	SPE	R/W	0	SPi Enable - SPI 使能。 0: 禁止传输, 此时 DR无法访问; 1: 可进行SPI 传输, 此时 CFGx 和 CRCx无法写入。

### 11.2.2 控制寄存器 2(CR2)

表 11- 4 SPI 控制寄存器 2(CR2)

位域	名称	访问	初值	描述
31: 16	-	-	0	保留
15	TXDMAEN	R/W	0	TX DMA ENable - 发送侧 dma 使能 0: 数据通过 apb 写入DR 传输 1: 数据通过 dma自动传输, 此时 DR无法写入
14: 10	-	-	0	保留
9: 8	TXFTHLV	R/W	0	TX THreshold LeVel - 发送侧阈值 当 Txfifo 内的空余空间不少于 (TXFTHLV+1) 时, 发出 TXA 标志 (这个值会自动向上与 DSIZE 对齐)
7	RXDMAEN	R/W	0	Receive Side DMA ENable - 接收侧 dma 使能 0: 数据通过 apb 读取 DR 传输 1: 数据通过 dma自动传输, 此时 DR无法读取
6: 2	-	-	0	保留
1: 0	RXFTHLV	R/W	0	RX THreshold LeVel - 接收侧阈值 当 Rxfifo 内的数据不少于 (RXFTHLV+1) 时, 发出 RXA 标志。(这个值会自动向上与 DSIZE 对齐)

### 11.2.3 控制寄存器 3(CR3)

表 11- 5 SPI 控制寄存器 3(CR3)

位域	名称	访问	初值	描述
31: 16	-	-	0	保留
15: 0	TSIZE	R/W	0	Transfer SIZE of frame - 单次传输帧数 当上一次传输结束 (CTSIZE=0), 继续下一次传输则会载入 TSIZE。 即单次传输的帧数为 (TSIZE + 1)

## 11.2.4 控制寄存器 4(CR4)

表 11- 6 SPI 控制寄存器 4(CR4)

位域	名称	访问	初值	描述
31: 16	-	-	0	保留
15: 0	CTSIZE	R/W	0	Counter of TSIZE - TSIZE 计数器 每次传输开始时会减 1, 表示本次传输剩余需要传输的帧数 (不包括当前帧)。 减成 0 后会重新载入 TSIZE 中的值。

## 11.2.5 中断寄存器(IER)

表 11- 7 SPI 中断寄存器 (IER)

位域	名称	访问	初值	描述
31: 16	-	-	0	保留
15	EOTIE	R/W	0	EOT Int Enable - EOT 中断使能 使能后, 会在 SR1 中的 EOT 为 1 时产生中断
14: 12	-	-	0	保留
11	MODFIE	R/W	0	MODF Int Enable - MODF 中断使能 使能后, 会在 SR1 中的 MODF 为 1 时产生中断
10	CRCEIE	R/W	0	CRCE Int Enable - CRCE 中断使能 使能后, 会在 SR1 中的 CRCE 为 1 时产生中断
9	UDRIE	R/W	0	UDR Int Enable - UDR 中断使能 使能后, 会在 SR1 中的 UDR 为 1 时产生中断
8	OVRIE	R/W	0	OVR Int Enable - OVR 中断使能 使能后, 会在 SR1 中的 OVR 为 1 时产生中断
7	SUSPIE	R/W	0	SUSP Int Enable - SUSP 中断使能 使能后, 会在 SR1 中的 SUSP 为 1 时产生中断
6	-	-	0	保留
5	TXEIE	R/W	0	TXE Int Enable - TXE 中断使能 使能后, 会在 SR1 中的 TXE 为 1 时产生中断
4	RXEIE	R/W	0	RXE Int Enable - RXE 中断使能 使能后, 会在 SR1 中的 RXE 为 1 时产生中断
3	-	-	0	保留
2	DXAIE	R/W	0	DXA Int Enable - DXA 中断使能 使能后, 会在 SR1 中的 DXA 为 1 时产生中断
1	TXAIE	R/W	0	TXA Int Enable - TXA 中断使能 使能后, 会在 SR1 中的 TXA 为 1 时产生中断
0	RXAIE	R/W	0	RXA Int Enable - RXA 中断使能 使能后, 会在 SR1 中的 RXA 为 1 时产生中断

## 11.2.6 状态寄存器 1(SR1)

表 11- 8 SPI 状态寄存器 1(SR1)

位域	名称	访问	初值	描述
31: 16	-	-	0	保留
15	EOT	R	0	End Of Transfer - 发送结束 若一次发送结束 (即 CTSIZE=0 且当前帧结束), 置 1。通过向该位写 1 来清零 硬件置位软件清零
14: 12	-	-	0	保留

位域	名称	访问	初值	描述
11	MODF	R	0	MODE Fault - 模式错误 若在多主模式下出现多主冲突, 置1。通过向该位写1来清零 硬件置位软件清零
10	CRCE	R	0	CRC Err - CRC 错误 若接收侧收到的 crc比较错误, 置1。通过向该位写1来清零 硬件置位软件清零
9	UDR	R	0	UnDer Run - 发送侧数据下溢 若 txfifo 已空后仍然发送数据时, 置1。通过向该位写1来清零 硬件置位软件清零
8	OVR	R	0	Over Run - 接收侧数据上溢 若 rxfifo 已满后仍然接收数据时, 置1。通过向该位写1来清零 硬件置位软件清零
7	SUSP	R	0	SUSPend - 挂起状态 仅在主模式下, 挂起时置1 硬件置位硬件清零
6	-	-	0	保留
5	TXE	R	0	TX fifo Empty - 发送侧 fifo 为空 当 txfifo 为空时置1 硬件置位硬件清零
4	RXE	R	0	RX fifo Empty - 接收侧 fifo 为空 当 rxfifo 为空时置1 硬件置位硬件清零
3	-	-	0	保留
2	DXA	R	0	DX dr Available - 两侧数据寄存器可访问 $dxa = rxa \& txa$ 硬件置位硬件清零
1	TXA	R	0	TX dr Available - 发送侧数据寄存器可访问 $txa = (txflv + (txfthlv + 1)) \leq 4$ 硬件置位硬件清零
0	RXA	R	0	RX dr Available - 接收侧数据寄存器可访问 $rxr = (rxflv - (rxfthlv + 1)) \geq 0$ 硬件置位硬件清零

### 11.2.7 状态寄存器 2(SR2)

表 11- 9 SPI 状态寄存器 2 (SR2)

位域	名称	访问	初值	描述
31: 11	-	-	0	保留
10: 8	TXFLV	R	0	TX Fifo LeVel - 发送侧 fifo 级别 表示txfifo 中储存的字节数, 0x0~0x4
7: 3	-	-	0	保留
2: 0	RXFLV	R	0	RX Fifo LeVel - 接收侧 fifo 级别 表示rxfifo 中储存的字节数, 0x0~0x4



## 11.2.8 配置寄存器 1(CFG1)

表 11- 10 SPI 配置寄存器 1 (CFG1)

位域	名称	访问	初值	描述
31: 13	-	-	0	保留
12: 8	DSIZE	R/W	7	Data SIZE of frame - 单帧数据大小 单帧数据大小为(DSIZE + 1) bits 注: 至少为 3, 即单帧数据大小>=4bits
7	LSBFRST	R/W		LSB transfer FiRST - LSB 先发模式 0: 每一帧从 MSB 开始先发 1: 每一帧从 LSB 开始先发
6: 2	-	-	0	保留
1	CPHA	R/W	0	Clock PHAse - 时钟相位 0: 从 SCK 第一个沿开始采样数据 1: 从 SCK 第二个沿开始采样数据
0	CPOL	R/W	0	Clock POLaRity - 时钟极性 0: 空闲时 SCK 为低电平 1: 空闲时 SCK 为高电平

## 11.2.9 配置寄存器 2(CFG2)

表 11- 11 SPI 配置寄存器 2 (CFG2)

位域	名称	访问	初值	描述
31: 16	-	-	0	保留
15: 8	BRINT	R/W	2	Baut Rate ratio INTEger - 波特率系数整数部分 主时钟频率 / 波特率 = (BRINT + BRDEC*2 <sup>(-6)</sup> ) 该比率应不小于 2.0
7:2	BRDEC	R/W	0	Baut Rate ratio DECimal - 波特率系数小数部分 注: 详见BRINT
1: 0	-	-	0	保留

## 11.2.10 配置寄存器 3(CFG3)

表 11- 12 SPI 配置寄存器 3 (CFG3)

位域	名称	访问	初值	描述
31: 10	-	-	0	保留
9: 8	SSMODE	R/W	0	SS MODE - 片选模式 0: 一般模式--主模式下 SS 做输出: SS = (!SPE); 从模式下 SS 做输入: 收到 SS=0 时有效 1: 软件模式--主模式下 SS 无效: PIN_SS 可用作其他 IO; 从模式下 SS 无效: 默认收到内部 SS=0 2: 多主模式--主模式下 SS 做输入: 当输入为 0 时产生MODF 错误; 从模式下 SS 做输入: 收到 SS=0 时有效 注: SS 输出和输入时可用SSREV 反向, 软件模式的从模式下, SSREV 置 1 可使内部 SS=1
7: 4	-	-	0	保留

位域	名称	访问	初值	描述
3	DOE	R/W	1	Data Output Enable - 数据输出使能 0: 数据输出禁止, 输出的 pin 脚可做其他 IO 1: 数据输出使能
2	DIE	R/W	1	Data Input Enable - 数据输入使能 0: 数据输入禁止, 输入的 pin 脚可做其他 IO 1: 数据输入使能
1	DIOSWP	R/W	0	Data In/Output SWaP - 数据输入输出交换 0: 主模式下 MISO 输入, MOSI 输出; 从模式下 MOSI 输入, MISO 输出 1: 主模式下 MOSI 输入, MISO 输出; 从模式下 MISO 输入, MOSI 输出
0	MSTR	R/W	0	MaSTeR - 主模式 0: SPI 从模式 1: SPI 主模式

### 11.2.11 CRC 寄存器 1(CRC1)

表 11- 13 SPI CRC 寄存器 1(CRC1)

位域	名称	访问	初值	描述
31: 1	CRCPOLY	R/W	0	CRC POLYnomial - crc 多项式 表示crc 多项式的高位, 即抹除了最低位的 crccpoly, 而实际的多项式为 {CRCPOLY, 1'b1} 目的是使用最低位作为 CRCEN, 因为只有 在多项式最低位为 1 时 crc 才有意义
0	CRCEN	R/W	0	CRC ENable - crc 使能 表示本次传输使用crc 模式

### 11.2.12 CRC 寄存器 2(CRC2)

表 11- 14 SPI CRC 寄存器 2(CRC2)

位域	名称	访问	初值	描述
31: 13	-	-	0	保留
12: 8	CRCSIZE	R/W	0	CRC SIZE - crc大小 表示一帧 crc 的数据大小, 可以与数据大小不 同 为(CRCSIZE+1)
7: 2	-	-	0	保留
1	TCRCINI	R/W	0	Tx CRC INItial value - 发送侧 crc 初始值 0: 表示txcrc 的初始值为全 0; 1: 表示txcrc 的初始值为全 1;
0	RCRCINI	R/W	0	Rx CRC INItial value - 接收侧 crc 初始值 0: 表示rxcrc 的初始值为全 0; 1: 表示rxcrc 的初始值为全 1;

### 11.2.13 数据寄存器(DR)

表 11- 15 SPI 数据寄存器 (DR)

位域	名称	访问	初值	描述
31: 0	DR	R/W	0	Data Reg - fifo 的读写接口: 读此寄存器将读出 RXFIFO 的数据, 读出的数据 字节数由 RXFTHLV 决定; 写此寄存器将把数据写入 TXFIFO, 写入的数据 字节数由 apb_wstrb 决定。

## 11.3 DMA 寄存器

表 11- 16 SPI-IO DMA 寄存器列表

偏移	名称	描述
0x100	ISR	状态寄存器
0x104	IFCR	状态清除寄存器
0x110	CCR1	控制寄存器 1
0x114	CNDTR1	计数寄存器 1
0x11c	CMAR1	地址寄存器 1
0x120	CCR2	控制寄存器 2
0x124	CNDTR2	计数寄存器 2
0x12c	CMAR2	地址寄存器 2

### 11.3.1 DMA 中断状态寄存器(ISR)

表 11- 17 SPI DMA 中断状态寄存器 (ISR)

位域	名称	访问	初值	描述
31: 8	-	-	0	保留
7	TEIF2	R	0	通道 2 传输错误标志
6	HTIF2	R	0	通道 2 传输过半标志
5	TCIF2	R	0	通道 2 传输完成标志
4	GIF2	R	0	通道 2 全局中断标志
3	TEIF1	R	0	通道 1 传输错误标志
2	HTIF1	R	0	通道 1 传输过半标志
1	TCIF1	R	0	通道 1 传输完成标志
0	GIF1	R	0	通道 1 全局中断标志

### 11.3.2 DMA 中断状态清除寄存器(IFCR)

表 11- 18 SPI DMA 中断状态清除寄存器 (IFCR)

位域	名称	访问	初值	描述
31: 8	-	-	0	保留
7	CTEIF2	W	0	清除通道 2 传输错误标志
6	CHTIF2	W	0	清除通道 2 传输过半标志
5	CTCIF2	W	0	清除通道 2 传输完成标志
4	CGIF2	W	0	清除通道 2 全局中断标志
3	CTEIF1	W	0	清除通道 1 传输错误标志
2	CHTIF1	W	0	清除通道 1 传输过半标志
1	CTCIF1	W	0	清除通道 1 传输完成标志
0	CGIF1	W	0	清除通道 1 全局中断标志

### 11.3.3 DMA 控制寄存器 n(CCRn)

表 11- 19 SPI DMA 控制寄存器 n(CCRn)

位域	名称	访问	初值	描述
31: 14	-	-	0	保留
13: 12	MSIZE_n	R/W	0	Memory SIZE of n-channel - n 通道内存访问数据大小 指 AXI 访问时的数据大小 0: msize = 8 bits 1: msize = 16 bits 2: msize = 32 bits
11: 10	-	-	0	保留
9: 8	PSIZE_n	R/W	0	Peripheral SIZE of n-channel - n 通道外围访问数据大小 指访问 spi 的 XFIFO 时的数据大小 0: psize = 8 bits 1: psize = 16 bits 2: psize = 32 bits
7	MINC_n	R/W	0	Memory INCrement Mode of n-channel - n 通道内存地址递进模式 0: n 通道关闭内存地址递进模式: AXI 访问的地址不随着 NDT 变化而变化 1: n 通道关闭内存地址递进模式: AXI 访问的地址随着 NDT 变化而递进, 递进的位数与 MSIZE 相关
6	-	-	0	保留
5	CIRC_n	R/W	0	CIRCular Mode of n-channel - n 通道循环模式 0: n 通道关闭循环模式: 当 NDT 即将为 0 时, 停止传输 1: n 通道开启循环模式: 当 NDT 即将为 0 时重新载入原值, 然后继续发送 注: NDT 即将为 0, 即 NDT=1 并前且此次 AXI 访问完成
4	-	-	0	保留
3	TEIE_n	R/W	0	TE Interrupt Enable of n-channel - n 通道 TE 中断使能 0: n 通道 TE 中断使能 1: n 通道 TE 中断禁止
2	HTIE_n	R/W	0	HT Interrupt Enable of n-channel - n 通道 HT 中断使能 0: n 通道 HT 中断使能 1: n 通道 HT 中断禁止
1	TCIE_n	R/W	0	TC Interrupt Enable of n-channel - n 通道 TC 中断使能 0: n 通道 TC 中断禁止 1: n 通道 TC 中断使能
0	EN_n	R/W	0	Enable of n-channel - n 通道使能 0: dma 的 n 通道禁止 1: dma 的 n 通道使能

注: 当 n=1 时, 为 RX; 当 n=2 时, 为 TX。

### 11.3.4 DMA 计数寄存器(CNDTRn)

表 11- 20 SPI DMA 计数寄存器 (CNDTRn)

位域	名称	访问	初值	描述
31: 16	-	-	0	保留
15: 0	NDT_n	R/W	0	Number of Data to Transfer n-channel - n 通道待传输数据量 每次访问完成时该数减一，为 0 时停止传输 若 CIRC_n=1，则会在即将减成 0 时重新载入最近一次写 NDT_n 的值

注：当 n=1 时，为 RX；当 n=2 时，为 TX。

### 11.3.5 DMA 地址寄存器(CMARn)

表 11- 21 SPI DMA 地址寄存器 (CMARn)

位域	名称	访问	初值	描述
31: 16	-	-	0	保留
15: 0	MA_n	R/W	0	Memory Address n-channel - n 通道目标内存地址 axi 访问的内存地址，若设置了 MINC_n，则会根据 NDT 累加访问地址

注：当 n=1 时，为 RX；当 n=2 时，为 TX。

# 12 I2C 控制器

## 12.1 概述

本章给出 I2C 的详细描述和配置使用。本系统芯片集成了 I2C 接口，支持主设备和从设备模式，主要用于实现两个器件之间数据的交换，芯片默认配置为从设备模式。主设备模式时，控制器可通过轮询或中断方式工作；从设备模式时，控制器可通过中断方式工作。I2C 总线是由数据线 SDA 和时钟 SCL 构成的串行总线，可发送和接收数据。器件与器件之间进行双向传送，最高传送速率 400kbps。

控制器支持特性：(1) Standard\_Mode, East\_Mode；(2)Clock stretch as slave；(3)Clock Synchronization as master。不支持特性：(1) Hs\_mode；(2)10-bit addressing；(3) 同时作为主/从设备。

## 12.2 I2C 控制器结构

I2C 主控制器的结构，主要模块有，时钟发生器（Clock Generator）、字节命令控制器（Byte Command Controller）、位命令控制器（Bit Command controller）、数据移位寄存器（Data Shift Register）。其余为 LPB 总线接口和一些寄存器。

- 1) 时钟发生器模块：产生分频时钟，同步位命令的工作。
- 2) 字节命令控制器模块：将一个命令解释为按字节操作的时序，即把字节操作分解为位操作。
- 3) 位命令控制器模块：进行实际数据的传输，以及位命令信号产生。
- 4) 数据移位寄存器模块：串行数据移位。

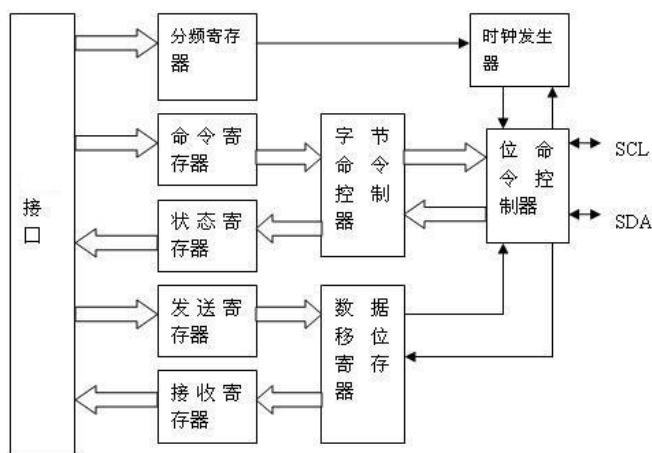


图12- 1 I2C 主控制器结构

## 12.3 I2C 控制器寄存器说明

本芯片集成了四个 I2C 控制器 I2C-0~3。

I2C-0 模块寄存器物理地址基址为：0X14201000，地址空间2KB。

I2C-1 模块寄存器物理地址基址为：0X14201800，地址空间2KB。

I2C-2 模块寄存器物理地址基址为：0X15101000，地址空间2KB。

I2C-3 模块寄存器物理地址基址为：0X15101800，地址空间2KB。

I2C 接口模块配置寄存器，如下表所示。

表 12- 1 I2C 配置寄存器列表

地址偏移	寄存器名称	描述
0x00	PRERlo	分频系数低字节寄存器
0x01	PRERhi	分频系数高字节寄存器
0x02	CTR	控制寄存器
0x03	TXD/RXD	发送/接收数据寄存器
0x04	CR/SR	命令/状态寄存器
0x05	BLTOP	总线死锁时间寄存器
0x07	SADDR	从设备地址寄存器

对于 I2C 接口模块，使用时要注意将对应的芯片复用引脚设置为相应的接口功能。

与 I2C 接口相关的引脚复用设置可查询节 2.19 中的外设功能引脚复用关系表，并根据复用配置寄存器配置相应的 GPIO 引脚复用关系，实现对应设备功能引脚。

### 12.3.1 分频锁存器低字节寄存器（PRERlo）

中文名： 分频锁存器低字节寄存器

寄存器位宽： [7: 0]

偏移量： 0x00

复位值： 0xff

表 12- 2 分频锁存器低字节寄存器

位域	位域名称	位宽	访问	描述
7:0	PRERlo	8	RW	存放分频锁存器的低8位

### 12.3.2 分频锁存器高字节寄存器（PRERhi）

中文名： 分频锁存器高字节寄存器

寄存器位宽： [7: 0]

偏移量： 0x01

复位值： 0xff

表 12- 3 分频锁存器高字节寄存器

位域	位域名称	位宽	访问	描述
7:0	PRERhi	8	RW	存放分频锁存器的高8位

假设分频锁存器的值为  $prescale$ ，从 APB 总线 PCLK 时钟输入的频率为  $clock\_a$ ，SCL 总线的输出频率为  $clock\_s$ ，则应满足如下关系：

$$clock\_s = clock\_a / 4 * (prescale + 1)$$

### 12.3.3 控制寄存器 (CTR)

中文名： 控制寄存器

寄存器位宽： [7: 0]

偏移量： 0x02

复位值： 0x00

表 12- 4 控制寄存器

位域	位域名称	位宽	访问	描述
7	EN	1	RW	模块工作使能位 1 正常工作模式; 0 对分频寄存器进行操作
6	IEN	1	RW	中断使能位 为1 则打开中断
5	ms	1	RW	主从模式选择位 (默认从设备): 0: 从设备模式; 1: 主设备模式
4	txrok	1	RW	从设备发送数据准备好: 从设备模式时, 当要发送的数据已写入 DR 时, 此位置为 1, 该位自动清零
3	rxrok	1	RW	从设备接收数据以读出: 从设备模式时, 当 DR 收到的数据已经被读出时, 此位置为 1, 该位自动清零
2	Reserved	-	-	保留
1	Buslock_check_en	1	RW	总线死锁状态检查使能: 使能后, 依据 buslock_top 寄存器规定的时间检查总线是否死锁, 死锁状态持续周期达到 {buslock_top, 16' b0} 后认定为产生死锁
0	Slv_autoreset_en	1	RW	总线死锁时从设备自动复位状态机使能: 使能时总线产生死锁后从设备会复位自身状态机, 从而解除死锁, 需要 buslock_check_en 使能

### 12.3.4 发送/接收数据寄存器 (TXD/RXD)

中文名： 发送/接收寄存器

寄存器位宽： [7: 0]

偏移量： 0x03

复位值： 0x00

表 12- 5 发送/接收寄存器

位域	位域名称	位宽	访问	描述
7:0	TXD/RXD	8	R/W	发送/接收数据寄存器 写入时为总线待发送的数据; 读出时为总线收到的数据



### 12.3.5 命令控制寄存器 (CR)

中文名： 命令寄存器

寄存器位宽： [7: 0]

偏移量： 0x04

复位值： 0x00

表 12- 6 命令寄存器

位域	位域名称	位宽	访问	描述
7	STA	1	W	开始信号： 该位置 1 时，作为主设备下一次传输产生开始波形
6	STO	1	W	结束信号： 该位置 1 时，作为主设备下一次传输产生结束波形
5	RD	1	W	读信号： 该位置 1 时，作为主设备下一次传输为总线读请求
4	WR	1	W	写信号： 该位置 1 时，作为主设备下一次传输为总线写请求
3	ACK	1	W	主设备应答： 该位写 1 表示下一次读数据返回时应答 NACK, 此时连续读请求结束
2	Recover	1	W	总线死锁恢复命令： 该位置 1 时作为主设备，看到总线死锁状态后，执行此命令接出死锁
1	Reserved	-	-	保留
0	IACK	1	W	产生中断应答信号

都是在 I2C 发送数据后硬件自动清零。对这些位读操作时候总是读回 ‘0’。bit 3 为 1 时表示此次传输结束时控制器不发送 ack，反之结束时发送 ack。

### 12.3.6 状态寄存器 (SR)

中文名： 状态寄存器

寄存器位宽： [7: 0]

偏移量： 0x04

复位值： 0x00

表 12- 7 状态寄存器

位域	位域名称	位宽	访问	描述
7	RxACK	1	R	收到的应答位： 0: 收到应答; 1: 收到 NACK
6	Busy	1	R	总线忙状态： 0: 总线空闲; 1: 总线忙
5	AL	1	R	主设备失去仲裁： 该位为 1 表示主设备时序总线控制权
4	Slave_addressed	1	R	从设备被寻址： 该位为 1 表示从设备已被寻址成功

位域	位域名称	位宽	访问	描述
3	Slave_rw	1	R	从设备读写： 0：从设备被读；1：从设备被写
2	BUSLOCK	1	R	总线死锁： 该位为 1 表示总线出现死锁
1	TIP	1	R	传输进行状态： 作为主设备时，该位为 1 表示传输正在进行中
0	IF	1	R	中断标志位： 当传输完一个字节或主设备丢失仲裁时，中断标志为 1

### 12.3.7 总线死锁时间寄存器（BLTOP）

中文名： 总线死锁时间寄存器

寄存器位宽： [7: 0]

偏移量： 0x05

复位值： 0xff

表 12- 8 总线死锁时间寄存器

位域	位域名称	位宽	访问	描述
7:0	Buslock_top	8	RW	总线死锁时间： 死锁状态持续周期达到 {buslock_top, 16' b0}， 认为产生死锁

### 12.3.8 从设备地址寄存器（SADDR）

中文名： 从设备地址寄存器

寄存器位宽： [7: 0]

偏移量： 0x07

复位值： 0x00

表 12- 9 从设备地址寄存器

位域	位域名称	位宽	访问	描述
7	Reserved	-	-	保留
6:0	saddr	7	RW	从设备地址： 作为从设备模式时，存放总线地址

# 13 打印接口

## 13.1 概述

打印接口模块集成在芯片打印系统下，并可共享给扫描系统与主系统。主要用于将内存中的打印数据搬运并转换成指定格式（串/并）输出至打印机芯，支持黑白与彩色（至多4通道）打印。其中，主要包含以下功能模块：

- DMA 控制器，用于从内存中搬运打印数据；
- 像素数据控制器，负责将像素数据转换为指定格式（1, 2, 4, 8 比特色深）；
- 调制单元，实现并行数据的串行解码与 sync 信号的处理；
- 交换台单元，主要用于 sync 信号与输出数据的来源选择。

打印接口模块基地址为 0x1510c000。

## 13.2 配置寄存器

表 13-1 打印控制器地址空间分布

寄存器地址	寄存器名称	寄存器大小
0x0000 - 0x03ff	打印控制器 0 (Channel 0)	1KB
0x0400 - 0x07ff	打印控制器 1 (Channel 1)	1KB
0x0800 - 0x0bff	打印控制器 2 (Channel 2)	1KB
0x0c00 - 0x0fff	打印控制器 3 (Channel 3)	1KB
0x2000 - 0x20ff	打印接口配置	256B

# 14 扫描接口

## 14.1 概述

扫描接口模块集成在芯片扫描系统下，并可共享给打印系统与主系统。主要用于生成扫描仪中 CIS、LED 和 AFE 的控制时序，并将采回的数据存至指定内存区域。扫描接口模块支持灰度与彩色扫描，同时支持分段式 (Segment) CIS 的数据拼接。支持的 AFE 数据输入宽度有 4、8 和 16 位，并可根据不同 AFE 特性对输入数据进行对齐。支持至多 12 个分段（单色 12 个，彩色 4 个），并可在 DMA 搬运时将不同颜色不同分段的数据进行重组，便于后续图像处理。其中，主要包含以下功能模块：

- CIS 控制单元，用于生成 CIS 及 LED 所需时序；
- AFE 控制单元，主要用于生成 AFE 控制时序并接收像素数据；
- 数据拼接单元，将接收到的像素数据以颜色和分段进行重组并添加 tag；
- DMA 控制器，根据 tag 将像素数据发送至指定内存区域；
- 中断处理单元，负责根据各模块状态产生中断。

扫描接口模块基地址为 0x15206000。

## 14.2 配置寄存器

表 14-1 SCAN 控制器地址空间分布

寄存器地址	寄存器名称	寄存器大小
0x0000 - 0x01ff	CIS控制单元配置	512B
0x0200 - 0x02ff	AFE控制单元配置	256B
0x0300 - 0x03ff	数据处理单元配置	256B
0x0400 - 0x04ff	扫描接口配置	128B

# 15 PMIO

## 15.1 概述

龙芯 2P0500 集成 3 个可编程多功能 IO (Programmable Multifunction IO, PMIO), 其中含有多个功能模块, 可通过 SoC 内部或外部输入产生输出。打印与扫描系统各集成了一个全功能 PMIO, 主系统包含一个精简版 PMIO。

PMIO 的核心功能模块, 可由软件或其他 PMIO 功能模块控制, 并产生复杂的输出。其中包含:

核心功能模块: (精简版 PMIO 仅支持 PWM 功能)

1. PWM 输出: 包含 Timer、Clock 和 PWM Generator, 在触发后生成相应波形;
2. FIFO 功能: 在触发后将寄存器中的值串行输出;
3. 其他功能: Access (读写对应寄存器), Bypass (输入直连输出)。

触发控制:

1. PMIO 内部, 其他 PMIO, 或外部输入事件均可作为触发信号。
2. 触发信号的采样/屏蔽控制

每个 PMIO 由两个相同的 Bank 构成 (精简版 PMIO 仅有一个 Bank), 两个 Bank 的基地址分别为 0x000 和 0x800。每个 Bank 含有 24 个功能模块 (Function Unit), 其中 16 个为可输出功能模块, 8 个为内部功能模块。PMIO 配置寄存器基地址为 0xf00。

## 15.2 配置寄存器

表 15-1 各系统 PMIO 控制器地址空间分布

各系统PMIO基地址	寄存器名称	寄存器大小
0x1420_6000	主系统Bank0寄存器	2KB
0x1420_6f00	主系统PMIO寄存器	2KB
0x1510_8000	打印系统Bank0寄存器	2KB
0x1510_8800	打印系统Bank1寄存器	2KB
0x1510_8f00	打印系统PMIO寄存器	2KB
0x1520_4000	扫描系统Bank0寄存器	2KB
0x1520_4800	扫描系统Bank1寄存器	2KB
0x1520_4f00	扫描系统PMIO寄存器	2KB

# 16 JPEG 编码器

## 16.1 概述

一个以 RGB 为输入，JPEG 为输出的编码器，该编码器直接生成 JPEG 文件格式的 SOI 到 EOI 部分字段。编码器内部集成了 DMA 读写接口，作为输入的 RGB 原始图像需要按照 Zero[31:24], R[23:16], G[15:8], B[7:0] 的格式按 32 位对齐放入内存，供 Y 分量使用的 DC 量化表和供 Cb, Cr 分量使用的 AC 量化表各存放了 64 个量化值 (Q)，可以将 4096/Q 的值按照 32 位对齐放入内存。需要注意的是，其中只有 [12:0] 是有效的。编码器采用固定的哈夫曼编码值，支持 4:4:4, 4:2:2, 4:1:1 和灰度四种采样形式。

控制器的基址为 0x14050000。

## 16.2 寄存器描述

表 16-1 JPEG 编码器寄存器列表

偏移	名称	描述
0x00	CMD	命令寄存器
0x04	STS	状态寄存器
0x08	Q_TABLE_ADDR	量化表寄存器
0x0c	BLOCK_OF_IMAGE	块数量寄存器
0x10	RGB_IN_ADDR	输入地址寄存器
0x14	JPEG_OUT_ADDR	输出地址寄存器
0x18	ENCODING_BYTE	编码字节数寄存器
0x1c	END_BIT_CNT	末尾字节位数寄存器
0x20	TRAILING	跟随字节寄存器

### 16.2.1 命令寄存器 (CMD)

偏 移： 0x00

复位值： 32'h0

表 16-2 命令寄存器

位域	名称	访问	描述
31	soft_reset	RW	软复位 高有效，复位后需要手动解除
31: 17	-	-	保留
16	finish_int_en	RW	结束中断使能 置 1 使能中断
15: 5	-	-	保留
4	need_load_q_tbl	RW	需要装载量化表 1: 装载内存中的量化表 0: 默认使用全 1 量化表

位域	名称	访问	描述
3	bit_bound	RW	末尾按位边界生产 1: 编码末尾按位边界生成 0: 末尾编码在字节边界前自动补若干个 1
2: 1	sample_type	RW	采样形式 00: 4: 4: 4 01: 4: 2: 2 10: 4: 1: 1 11: 灰度
0	start	RW	开始编码 全部配置完成后, 置 1 开始编码

### 16.2.2 状态寄存器 (STS)

偏 移: 0x04

复位值: 32'h0

表 16-3 状态寄存器

位域	名称	访问	描述
31: 1	-	-	保留
0	jpeg_enc_done	RW1C	JPEG 编码完成 编码完成后此位变成 1, 写 1 清零

### 16.2.3 量化表地址寄存器 (Q\_TABLE\_ADDR)

偏 移: 0x08

复位值: 32'h0

表 16-4 量化表地址寄存器

位域	名称	访问	描述
31: 4	addr	RW	地址 量化表在内存的地址, 按 16 字节对齐
3: 0	-	-	保留

### 16.2.4 块数量寄存器 (BLOCK\_OF\_IMAGE)

偏 移: 0x0c

复位值: 32'h0

表 16-5 块数量寄存器

位域	名称	访问	描述
31: 0	amount	RW	数量 原始 RGB 图像中包含 8*8 块的数量

### 16.2.5 输入地址寄存器 (RGB\_IN\_ADDR)

偏 移: 0x10

复位值: 32'h0

表 16-6 输入地址寄存器

位域	名称	访问	描述
31: 4	addr	RW	地址 原始 RGB 图像在内存的地址，按 16 字节对齐
3: 0	-	-	保留

### 16.2.6 输出地址寄存器 (JPEG\_OUT\_ADDR)

偏 移: 0x14

复位值: 32'h0

表 16-7 输出地址寄存器

位域	名称	访问	描述
31: 4	addr	RW	地址 输出 jpeg 编码在内存的地址，按 16 字节对齐
3: 0	-	-	保留

### 16.2.7 编码字节数寄存器 (ENCODING\_BYTE)

偏 移: 0x18

复位值: 32'h0

表 16-8 编码字节数寄存器

位域	名称	访问	描述
31: 0	amount	RW	数量 输出 JPEG 编码的字节数

### 16.2.8 末尾字节位数寄存器 (ENF\_BIT\_CNT)

偏 移: 0x1c

复位值: 32'h0

表 16-9 末尾字节位数寄存器

位域	名称	访问	描述
31: 3	-	-	保留
2: 0	amount	RW	数量 编码末尾字节包含有效位个数

### 16.2.9 跟随字节寄存器 (TRAILING)

偏 移: 0x20

复位值: 32'h0

表 16-10 跟随字节寄存器

位域	名称	访问	描述
31: 11	-	-	保留
10: 8	bit_cnt	RW	跟随字节位数 输入上一次编码的末尾字节有效位数, 用于连续编码
7: 0	byte	RW	跟随字节 输入上一次编码的末尾字节, 用于连续编码



## 16.3 编程示例

JPEG 编码配置操作流程:

- (1) 命令寄存器配置采用格式: `CMD = 0x10;` //装载量化表, 按照 4: 4: 4 采样
- (2) 状态寄存器清除原始状态: `STS = 0x1;` //清空相应状态位
- (3) 配置量化表地址: `Q_TABLE_ADDR = 0x0;` //量化表地址为 0x0
- (4) 配置图像块大小: `BLOCK_OF_IMAGE = 0x1000;` //原始 RGB 图像包含 4096 个 8\*8 块
- (5) 配置原图像大小: `RGB_IN_ADDR = 0x200;` //原始 RGB 图像为 0x200
- (6) 配置 JPEG 格式文件存放地址: `JPEG_OUT_ADDR = 0x40000;` //输出 JPEG 编码地址为 0x40000
- (7) 启动 JPEG 编码: `CMD = CMD | 0x1;` //开始编码
- (8) 等待 JPEG 编码结束标志: `while(~(STS & 0x1));` //等待编码结束状态位
- (9) 读取 JPEG 格式文件...

# 17 JBIG85 解码器

## 17.1 概述

JBIG85 解码模块可实现对由 jbig85 压缩算法压缩过的图片数据进行解码处理，可负责同时对青、粉、黄、黑四个颜色进行对应数据的解码工作。

JBIG85 解码单元包括 4 个模块，分别负责对青粉黄黑四个颜色进行对应数据的解码工作。每个模块包括主要包括两个子模块：主控制模块，JBIG85 算法解码模块。结构框图如图 17-1 所示：

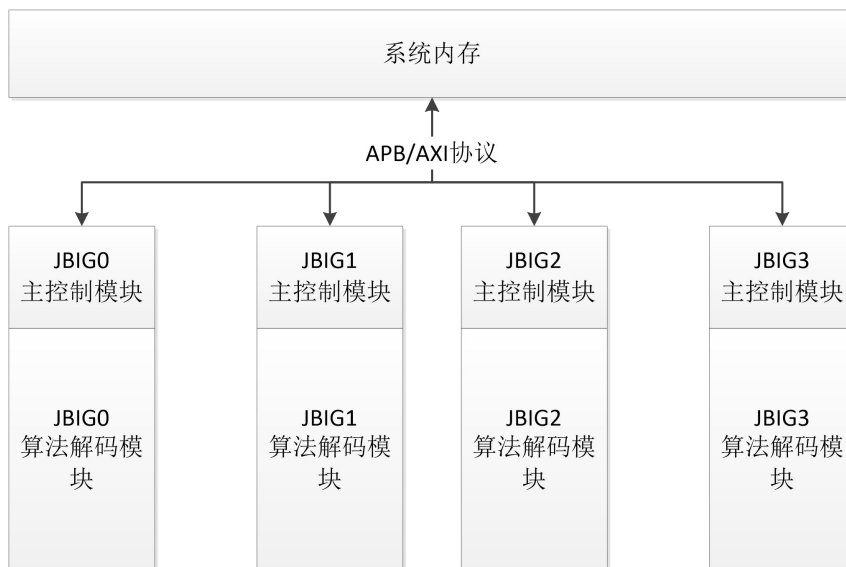


图17- 1 JBIG85 解码结构示意图

JBIG85 主控制模块包括三个功能：寄存器配置、输入数据读取，输出数据写回。配置寄存器通过配置总线进行寄存器读写，寄存器项包括青粉黄黑各模块使能寄存器、输入数据起始地址、输入数据长度、输出数据起始地址、多模块的中断传递类型、解码错误状态寄存器。输入数据保存在内存中，数据地址由 APB 进行配置，主控制模块通过 AXI 协议对数据进行读取，并将读取的数据发送至 JBIG85 解码模块进行解码。输出数据通过 AXI 协议进行写回，写回基地址同样由 APB 进行配置，并且所有写回数据按照图片行数据进行 256 像素单位对齐。

JBIG85 算法解码模块主要实现对经由 jbig85 压缩算法压缩过的图片数据。

控制器的基址为 0x14040000。

## 17.2 寄存器描述

JBIG85 模块有 4 个相同的 JBIG 解码单元，四个模块均具有相同的配置寄存器格式。

表 17- 1 JBIG 配置地址

物理地址	设备	备注
0x0000	JBIG0	四个模块寄存器配置方式相同， 区别在于寄存器基地址偏移不同
0x0800	JBIG1	
0x1000	JBIG2	
0x1800	JBIG3	

JBIG85 每个模块包括 10 个可以通过配置总线进行读写的寄存器。

表 17- 2 JBIG 配置寄存器

偏移地址	位宽	寄存器名称	读写	说明
0x0	32	jbig_top_cfg	R/W	JBIG 模块的整体配置位，
0x4	32	pscd_addr	R/W	当前 JBIG 模块待解码原始数据的存储地址
0x8	32	pscd_data_length	R/W	当前 JBIG 模块待解码数据的长度，单位为字节
0xc	32	output_addr	R/W	当前 JBIG 模块解码后数据存储地址
0x10	1	start	R/W	开始当前 JBIG 模块的解码，高电平有效
0x14	1	error_processed_reg	R/W	错误清除位，出现错误并处理结束后，写 1 清除
0x18	32	error_code_reg	R	错误码状态寄存器
0x1c	1	apb_clear	W	清空当前 JBIG 模块的所有配置项，高电平有效
0x20	1	int_reg_clear	W	清除当前 JBIG 模块的中断位，高电平有效
0x24	1	int_reg	R	读取当前 JBIG 模块的中断位

(1) jbig\_top\_cfg 寄存器

偏移地址：0x0。

表 17- 3 jbig\_top\_cfg 寄存器

位域	名称	复位值	访问	说明
31:9	Reserved	-	RO	保留
8	int_oen	1' b0	R_W	4 个 JBIG 模块的中断位合并为 1 位输出， 0: 将 4 位中断位做 AND 操作，所有四个模块均为中断，才输出中断，可用作没有数据错误的正常打印方式。 1: 将 4 位中断位做 OR 操作，所有模块有一个中断，则立即输出中断，可用作有数据错误的调试方式。
7:4	int_en	4' b0	R_W	配置 4 个 JBIG 模块的中断使能位 1: 打开使能 0: 关闭使能 [7]表示 JBIG3 [6]表示 JBIG2 [5]表示 JBIG1 [4]表示 JBIG0

位域	名称	复位值	访问	说明
3:0	jbig_en	4' b00	R/W	配置 4 个 JBIG 模块的使能位 1: 打开使能 0: 关闭使能 [3]表示 JBIG3 [2]表示 JBIG2 [1]表示 JBIG1 [0]表示 JBIG0

(2) error\_code\_reg 寄存器

偏移地址: 0x18。

JBIG 错误码, 数据 s->buffer 为 jbig85 数据前 20 字节数据, 为当前 jbig85 数据的描述符, x0/y0/10/mx/options 为通过 buffer 计算得来的数据。

表 17- 4 error\_code\_reg 寄存器

位域	名称	复位值	访问	说明
31:24	jbig3_err_code	8' b0	R_W	JBIG3 的错误码, 位域含义等同[7:0]
23:16	jbig2_err_code	8' b0	R_W	JBIG2 的错误码, 位域含义等同[7:0]
15:8	jbig1_err_code	8' b0	R_W	JBIG1 的错误码, 位域含义等同[7:0]
7:0	jbig0_err_code	8' b0	R/W	JBIG0 的错误码, 数据 s 为 jbig85 数据的描述符, 具体表示参考 jbig85_c_code [7:4] == 4' d6, 输入数据包含错误输入 [3:0] == 4' d1: s->buffer[1] < s->buffer[0] [3:0] == 4' d2: s->buffer[3] != 0 [3:0] == 4' d3: s->buffer[18]&0xf0 != 0 [3:0] == 4' d4: s->buffer[19]&80 != 0 [3:0] == 4' d5: s->buffer[2] == 0 [3:0] == 4' d6: s->x0 == 0 [3:0] == 4' d7: s->y0 == 0 [3:0] == 4' d8: s->10 == 0 [3:0] == 4' d9: s->mx > 127 [7:4] == 4' d7, 输入数据不符合 JBIG85 格式 [3:0] == 4' d8: s->buffer[0] != 0 [3:0] == 4' d9: s->buffer[1] != 0 [3:0] == 4' d10: s->buffer[2] != 1 [3:0] == 4' d11: s->buffer[17] != 0 [3:0] == 4' d12: s->buffer[18] != 0 [3:0] == 4' d13: s->options &0x17 != 0

## 17.3 编程说明

### 操作 1: 正常打印配置方式

1. 逐个配置四个 JBIG 模块的 apb\_clear 寄存器，清空所有 jbig 配置项；
2. 逐个配置四个 JBIG 模块的 jbig\_cfg 寄存器；
3. 配置需要启用的 JBIG 模块的 pscd\_addr 寄存器，将待解压缩的 jbig 数据地址写入

该寄存器；

4. 配置需要启用的 JBIG 模块的 pscd\_length 寄存器，将待解压的 jbig 数据长度写入该寄存器；

5. 配置需要启用的 JBIG 模块的 output\_addr 寄存器，将解压后的数据地址写入该寄存器；

6. 配置需要启用的 JBIG 模块的 start 寄存器，开始解压缩。

### 操作 2: 打印结束后获取状态

1. 读取 int\_reg 寄存器，如果值为 1，表示打印结束或者出现错误；
2. 读取 error\_code\_reg 寄存器，如果值为 0，表示没有错误，为正常结束；
3. 配置 int\_reg\_clear 寄存器；
4. 重复正常打印配置方式，继续下一页打印。

### 操作 3: 打印错误及状态获取

1. 读取 int\_reg 寄存器，如果值为 1，表示打印结束或者出现错误；
2. 读取 error\_code\_reg 寄存器，如果值不为 0，表示有错误，对照错误码表，确定错误类型；

3. 配置 int\_reg\_clear 寄存器；
4. 修改错误，重复正常打印配置方式，继续下一页打印。

# 18 UART 控制器

## 18.1 概述

2P0500 集成了 8 个 UART 控制器,通过 APB 总线与总线桥通信。UART 控制器提供与 MODEM 或其他外部设备串行通信的功能,例如与另外一台计算机,以 RS232 为标准使用串行线路进行通信。该控制器在设计上能很好地兼容国际工业标准半导体设备 16550A。

## 18.2 控制器结构

UART 控制器有发送和接收模块 (Transmitter and Receiver)、MODEM 模块、中断仲裁模块 (Interrupt Arbitrator)、和访问寄存器模块 (Register Access Control), 这些模块之间的关系见下图所示。

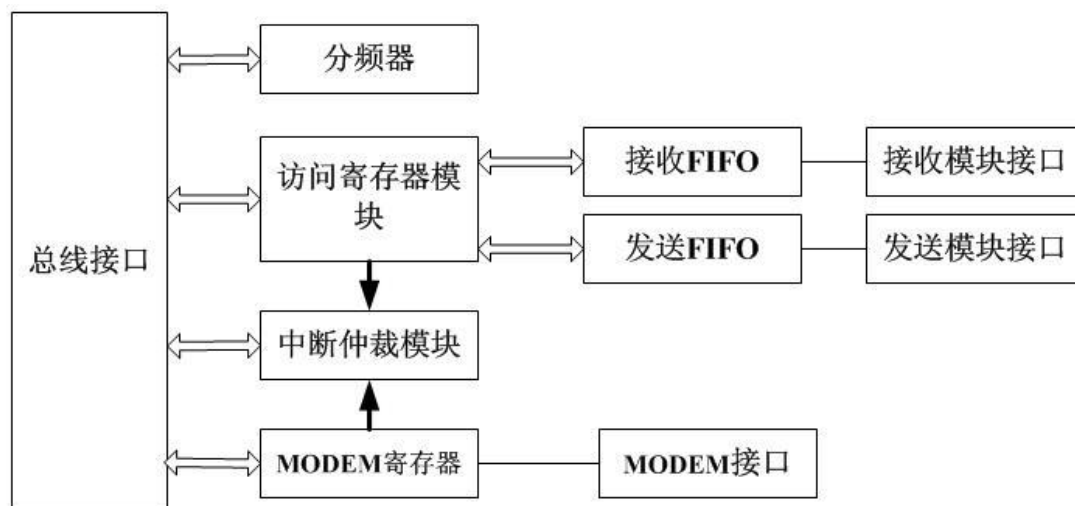


图18- 1 UART 控制器结构

主要模块功能及特征描述如下:

1) 发送和接收模块: 负责处理数据帧的发送和接收。发送模块是将 FIFO 发送队列中的数据按照设定的格式把并行数据转换为串行数据帧, 并通过发送端口送出去。接收模块则监视接收端信号, 一旦出现有效开始位, 就进行接收, 并实现将接收到的异步串行数据帧转换为并行数据, 存入 FIFO 接收队列中, 同时检查数据帧格式是否有错。UART 的帧结构是通过行控制寄存器 (LCR) 设置的, 发送和接收器的状态被保存在行状态寄存器 (LSR) 中。

2) MODEM 模块: MODEM 控制寄存器 (MCR) 控制输出信号 DTR 和 RTS 的状态。MODEM 控制模块监视输入信号 DCD, CTS, DSR 和 RI 的线路状态, 并将这些信号的状态记录在 MODEM 状态寄存器 (MSR) 的相对应位中。

3) 中断仲裁模块: 当任何一种中断条件被满足, 并且在中断使能寄存器 (IER) 中相应位置 1, 那么 UART 的中断请求信号 UAT\_INT 被置为有效状态。为了减少和外部软件的交

互，UART 把中断分为四个级别，并且在中断标识寄存器（IIR）中标识这些中断。四个级别的中断按优先级级别由高到低的排列顺序为，接收线路状态中断；接收数据准备好中断；传送拥有寄存器为空中断；MODEM 状态中断。

4) 访问寄存器模块：当 UART 模块被选中时，CPU 可通过读或写操作访问被地址线选中的寄存器。

## 18.3 寄存器描述

龙芯2P0500中有8个UART，每个UART只是访问基址不一样，其功能寄存器保持一致。

- UART0寄存器物理地址基址为 0x14200000。
- UART1寄存器物理地址基址为 0x14200400。
- UART2寄存器物理地址基址为 0x14200800。
- UART3寄存器物理地址基址为 0x14200c00。
- UART4寄存器物理地址基址为 0x15100000。
- UART5寄存器物理地址基址为 0x15100800。
- UART6寄存器物理地址基址为 0x15200000。
- UART7寄存器物理地址基址为 0x15200800。

UART 接口模块配置寄存器，如下表所示。

表 18- 1 UART 配置寄存器列表

地址偏移	寄存器名称	描述
0x00	DAT/DL_L	数据寄存器/分频系数低 8 位
0x01	IER/DL_H	中断使能寄存器/分频系数高 8 位
0x02	IIR	中断标识寄存器
0x02	FCR/DL_D	FIFO 控制寄存器/分频系统小数位
0x03	LCR	线路控制寄存器
0x04	MCR	Modem 控制寄存器
0x05	LSR	线路状态寄存器
0x06	MSR	Modem 状态寄存器

对于 UART 接口模块，使用时要注意将对应的芯片复用引脚设置为相应的接口功能。

与 UART 接口相关的引脚复用设置可查询 2.20 章节外设功能引脚复用关系表，配置相应的 GPIO 引脚复用关系，实现对应设备功能引脚。

### 18.3.1 数据寄存器（DAT）

中文名： 数据传输寄存器

寄存器位宽： [7: 0]

偏移量： 0x00

复位值： 0x00

表 18- 2 数据传输寄存器

位域	位域名称	位宽	访问	描述
7:0	Tx FIFO	8	W	数据传输寄存器

### 18.3.2 中断使能寄存器 (IER)

中文名： 中断使能寄存器

寄存器位宽： [7: 0]

偏移量： 0x01

复位值： 0x00

表 18- 3 中断使能寄存器

位域	位域名称	位宽	访问	描述
7:4	Reserved	4	RW	保留
3	IME	1	RW	Modem 状态中断使能 ‘0’ - 关闭 ‘1’ - 打开
2	ILE	1	RW	接收器线路状态中断使能 ‘0’ - 关闭 ‘1’ - 打开
1	ITxE	1	RW	传输保存寄存器为空中断使能 ‘0’ - 关闭 ‘1’ - 打开
0	IRxE	1	RW	接收有效数据中断使能 ‘0’ - 关闭 ‘1’ - 打开

### 18.3.3 中断标识寄存器 (IIR)

中文名： 中断源寄存器

寄存器位宽： [7: 0]

偏移量： 0x02

复位值： 0xc1

表 18- 4 中断源寄存器

位域	位域名称	位宽	访问	描述
7:4	Reserved	4	R	保留
3:1	II	3	R	中断源表示位，详见下表
0	INTp	1	R	中断表示位

表 18- 5 中断控制功能表

Bit 3	Bit 2	Bit 1	优先级	中断类型	中断源	中断复位控制
0	1	1	1 <sup>st</sup>	接收线路状态	奇偶、溢出或帧错误，或打断中断	读 LSR
0	1	0	2 <sup>nd</sup>	接收到有效数据	FIFO 的字符个数达到 trigger 的水平	FIFO 的字符个数低于 trigger 的值
1	1	0	2 <sup>nd</sup>	接收超时	在 FIFO 至少有一个字符，但在 4 个字符时间内没有任何操作，包括读和写操作	读接收 FIFO
0	0	1	3 <sup>rd</sup>	传输保存寄存器为空	传输保存寄存器为空	写数据到 THR 或者多 IIR
0	0	0	4 <sup>th</sup>	Modem 状态	CTS, DSR, RI or DCD.	读 MSR



### 18.3.4 FIFO 控制寄存器 (FCR)

中文名: FIFO 控制寄存器

寄存器位宽: [7: 0]

偏移量: 0x02

复位值: 0xc0

表 18- 6 FIFO 控制寄存器

位域	位域名称	位宽	访问	描述
7:6	TL	2	W	接收FIFO 提出中断申请的trigger 值 ‘00’ - 1 字节 ‘01’ - 4 字节 ‘10’ - 8 字节 ‘11’ - 14 字节
5:3	Reserved	3	W	保留
2	Txset	1	W	‘1’ 清除发送FIFO 的内容, 复位其逻辑
1	Rxset	1	W	‘1’ 清除接收FIFO 的内容, 复位其逻辑
0	Reserved	1	W	保留

### 18.3.5 线路控制寄存器 (LCR)

中文名: 线路控制寄存器

寄存器位宽: [7: 0]

偏移量: 0x03

复位值: 0x03

表 18- 7 线路控制寄存器

位域	位域名称	位宽	访问	描述
7	dlab	1	RW	分频锁存器访问位 ‘1’ - 访问操作分频锁存器 ‘0’ - 访问操作正常寄存器
6	bcbl	1	RW	打断控制位 ‘1’ - 此时串口的输出被置为 0(打断状态). ‘0’ - 正常操作
5	spbl	1	RW	指定奇偶校验位 ‘0’ - 不用指定奇偶校验位 ‘1’ - 如果 LCR[4] 位是 1 则传输和检查奇偶校验位为 0。如果 LCR[4] 位是 0 则传输和检查奇偶校验位为 1。
4	eps	1	RW	奇偶校验位选择 ‘0’ - 在每个字符中有奇数个 1) 包括数据和奇偶校验位 ( ) ‘1’ - 在每个字符中有偶数个 1
3	pe	1	RW	奇偶校验位使能 ‘0’ - 没有奇偶校验位 ‘1’ - 在输出时生成奇偶校验位, 输入则判断奇偶校验位
2	sb	1	RW	定义生成停止位的位数 ‘0’ - 1 个停止位 ‘1’ - 在 5 位字符长度时是 1.5 个停止位, 其他长度是 2 个停止位
1:0	bec	2	RW	设定每个字符的位数 ‘00’ - 5 位 ‘01’ - 6 位 ‘10’ - 7 位 ‘11’ - 8 位

### 18.3.6 MODEM 控制寄存器 (MCR)

中文名: Modem 控制寄存器

寄存器位宽: [7: 0]

偏移量: 0x04

复位值: 0x00

表 18- 8 Modem 控制寄存器

位域	位域名称	位宽	访问	描述
7	Infrared	1	W	红外载波使能: 0: 关闭; 1: 使能
6	Rx_pol	1	W	载波使能时 RX 极性 0: 正常, 低电平有效; 1: 反转, 高电平有效
5	Infrared_pol	1	W	载波使能时 TX 极性 0: 正常, 低电平有效; 1: 反转, 高电平有效
4	Loop	1	W	回环模式控制位 ‘0’ - 正常操作 ‘1’ - 回环模式。在在回环模式中, TXD 输出一直为 1, 输出移位寄存器直接连到输入移位寄存器中。其他连接如下。 DTR DSR RTS CTS Out1 RI Out2 DCD
3	OUT2	1	W	在回环模式中连到 DCD 输入
2	OUT1	1	W	在回环模式中连到 RI 输入
1	RTSC	1	W	RTS 信号控制位
0	DTRC	1	W	DTR 信号控制位

### 18.3.7 线路状态寄存器 (LSR)

中文名: 线路状态寄存器

寄存器位宽: [7: 0]

偏移量: 0x05

复位值: 0x00

表 18- 9 线路状态寄存器

位域	位域名称	位宽	访问	描述
7	ERROR	1	R	错误表示位 ‘1’ - 至少有奇偶校验位错误, 帧错误或打断中断的一个。 ‘0’ - 没有错误
6	TE	1	R	传输为空表示位 ‘1’ - 传输 FIFO 和传输移位寄存器都为空。 给传输 FIFO 写数据时清零 ‘0’ - 有数据
5	TFE	1	R	传输 FIFO 位空表示位 ‘1’ - 当前传输 FIFO 为空, 给传输 FIFO 写数据时清零 ‘0’ - 有数据

位域	位域名称	位宽	访问	描述
4	BI	1	R	打断中断表示位 ‘1’ - 接收到 起始位+数据+奇偶位+停止位都是 0, 即有打断中断 ‘0’ - 没有打断
3	FE	1	R	帧错误表示位 ‘1’ - 接收的数据没有停止位 ‘0’ - 没有错误
2	PE	1	R	奇偶校验位错误表示位 ‘1’ - 当前接收数据有奇偶错误 ‘0’ - 没有奇偶错误
1	OE	1	R	数据溢出表示位 ‘1’ - 有数据溢出 ‘0’ - 无溢出
0	DR	1	R	接收数据有效表示位 ‘0’ - 在 FIFO 中无数据 ‘1’ - 在 FIFO 中有数据

对这个寄存器进行读操作时, LSR[4:1]和 LSR[7]被清零, LSR[6:5]在给传输 FIFO 写数据时清零, LSR[0]则对接收 FIFO 进行判断。

### 18.3.8 MODEM 状态寄存器 (MSR)

中文名: Modem 状态寄存器

寄存器位宽: [7: 0]

偏移量: 0x06

复位值: 0x00

表 18- 10 Modem 状态寄存器

位域	位域名称	位宽	访问	描述
7	CD CD	1	R	DCD 输入值的反, 或者在回环模式中连到Out2
6	CRI	1	R	RI 输入值的反, 或者在回环模式中连到OUT1
5	CDSR	1	R	DSR 输入值的反, 或者在回环模式中连到DTR
4	CCTS	1	R	CTS 输入值的反, 或者在回环模式中连到RTS
3	DDCD	1	R	DDCD 指示位
2	TERI	1	R	RI 边沿检测。RI 状态从低到高变化
1	DDSR	1	R	DDSR 指示位
0	DCTS	1	R	DCTS 指示位

### 18.3.9 分频锁存器

中文名: 分频锁存器低 8 位

寄存器位宽: [7: 0]

偏移量: 0x00

复位值: 0x00

表 18- 11 分频锁存器低 8 位寄存器

位域	位域名称	位宽	访问	描述
7:0	DL_L	8	RW	存放分频锁存器的低 8 位数值

中文名： 分频锁存器高 8 位

寄存器位宽： [7: 0]

偏移量： 0x01

复位值： 0x00

表 18- 12 分频锁存器高 8 位寄存器

位域	位域名称	位宽	访问	描述
7:0	DL_H	8	RW	存放分频锁存器的高 8 位数值

中文名： 分频锁存器小数位

寄存器位宽： [7: 0]

偏移量： 0x02

复位值： 0x00

表 18- 13 分频锁存器小数位寄存器

位域	位域名称	位宽	访问	描述
7:0	DL_D	8	RW	存放分频锁存器的小数部分二进制值 例如,分频小数值为 0.45,DL_D=0x73 (即: 0x100*0.45)

UART 分频由 DL\_H、DL\_L、DL\_D 三个寄存器组成分频值寄存器 DL，则 UART 波特率为： $CLK_{in}/16*DL$ 。例如，输入为 10MHz 时钟，目标波特率为 115200，则  $DL=5.42535$ ，故， $DL_H=0x0$ ， $DL_L=0x5$ ， $DL_D=0x6c$  (即： $0x100*0.42535$ )。波特率配置值与期望值的误差在 5%以内，否则无法识别所有数据位，将导致串口显示乱码。

# 19 加解密

## 19.1 DES

### 19.1.1 DES 功能概述

DES 控制器采用 32 位的 APB 接口，支持使用 DES/TDEA 算法进行加/解密，并支持使用 APB 接口进行 DMA 操作。DES 控制器采用了 OpenCore 的面积节约方案的 DES3 加/解密单元作为实现加/解密的运算单元。这个运算单元每 16 个时钟周期完成一次 DES 加/解密，每 48 个时钟周期完成一次 TDEA 加/解密。为了减少加/解密的等待时间，DES 控制器使用 2 个时钟域分别处理 APB 接口的操作和进行 DES 加/解密计算（DES 加解密使用的时钟的频率更高）。两个不同的时钟域通过 2 个 4 项的 64bit 位宽异步 FIFO 交换加/解密运算前后的数据。

DES 控制器在使用前需要先配置密钥以及 Command 寄存器。控制器有 3 个使用 64bit 格式存储的密钥：Key0、Key1、Key2。其中，Key1、Key2 仅在 TDEA 算法是需要进行配置。控制器不检查密钥中的校验位。待运算（加密还是解密由 Command[1]的值确定）的数据通过 Data\_low 和 Data\_high 写入运算数据 FIFO。DES 运算模块从运算数据 FIFO 读取数据后进行加/解密运算迭代，迭代的结果被送入运算结果 FIFO。通过 APB 接口查询 Command[4]可以获知运算结果是否就绪。当 Command[4]的值为 0 时，可以从 APB 接口通过 Data\_low 和 Data\_high 读取运算结果 FIFO 中的运算结果。

### 19.1.2 DES 访问地址

DES 的 PCI 设备编号为(dev29,func1)，可以通过配置总线设置 DES 寄存器的基地址。物理地址的低 6 位作为偏移访问配置寄存器。

### 19.1.3 DES 寄存器描述

DES 的寄存器列表及寄存器说明如下：

表 19- 1 DES 寄存器列表

地址偏移	名称	说明
0x0	Key0_low	DES 密钥的低 32 位/TDEA 密钥 0 的低 32 位
0x4	Key0_high	DES 密钥的高 32 位/TDEA 密钥 0 的高 32 位
0x8	Key1_low	TDEA 密钥 1 的低 32 位
0xc	Key1_high	TDEA 密钥 1 的高 32 位
0x10	Key2_low	TDEA 密钥 2 的低 32 位
0x14	Key2_high	TDEA 密钥 2 的高 32 位

地址偏移	名称	说明
0x18	Data_low	非 DMA 模式 (Command[2] = 1'b0) : 待加/解密数据低 32 位的写入端口; 加/解密后数据低 32 位的读出端口。 DMA 模式 (Command[2] = 1'b1) : Data_low 和 Data_high 不做区分。先写入/读取的是数据的低 32 位, 后写入/读取的是数据的高 32 位。
0x1c	Data_high	非 DMA 模式 (Command[2] = 1'b0) : 待加/解密数据高 32 位的写入端口; 加/解密后数据高 32 位的读出端口。 DMA 模式 (Command[2] = 1'b1) : Data_low 和 Data_high 不做区分。先写入/读取的是数据的低 32 位, 后写入/读取的是数据的高 32 位。
0x20	Command	命令和状态控制寄存器
0x24	Rev	保留
0x28		
0x2c		

Comand 寄存器位域说明:

表 19- 2 Comand 寄存器

位域	复位值	名称	属性	说明
0	0	des3	RW	0: 使用 DES 算法 1: 使用 TDEA 算法
1	0	decrypt	RW	0: 加密操作 1: 解密操作
2	0	dma_start	RW	写入 1 启动 DMA 操作, 写入 0 无影响 在 DMA 操作完成前此位保持为 1 当 DMA 操作完成时此位自动清零
3	0	dma_done	RW1C	当 DMA 操作完成时, 此位置 1 向此位写入 1, 则清零
4	1	dout_empty	RO	0: 数据读 FIFO 非空 1: 数据读 FIFO 为空
5	0	din_full	RO	0: 数据写 FIFO 未 1: 数据写 FIFO 已满
7:6	0	Rev	RO	保留
31:8	0	dma_count	RW	需要进行 DMA 加/解密的 64 位数的个数

## 19.2 AES

### 19.2.1 AES 功能概述

AES 控制器采用 32 位的 APB 接口, 支持使用 128-bit KEY、192-bit KEY、256-bit KEY

进行 AES 算法加/解密，并支持使用 APB 接口进行 DMA 操作。在使用 DMA 功能时，支持 CTR 模式进行加/解密。AES 控制器有 3 个主要模块：KEY 扩展模块、加密模块、解密模块。加/解密运算模块每 11 个时钟周期完成一次 128-bit KEY 的 AES 加/解密，每 13 个时钟周期完成一次 192-bit KEY 的 AES 加/解密，每 15 个时钟周期完成一次 256-bit KEY 的 AES 加/解密。为了减少加/解密的等待时间，AES 控制器使用 2 个时钟域分别处理 APB 接口的操作和进行 AES 加/解密计算（AES 加解密使用的时钟的频率更高）。KEY 扩展模块工作在速度较慢的 APB 时钟域。两个不同的时钟域通过 2 个 4 项的 128bit 位宽异步 FIFO 交换加/解密运算前后的数据。

AES 控制器在使用前需要先配置密钥以及 Command 寄存器。AES 的密钥、明文和密文均以小尾端的格式进行存储。控制器使用 8 个 32bit 寄存器以小尾端的格式存储密钥：Key0、Key1、Key2、key3、key4、key5、key6、key7。其中，Key4、Key5 仅在 192-bit KEY 和 256-bit KEY 时需要进行配置；Key6、Key7 仅在 256-bit KEY 时需要进行配置。待运算（加密还是解密由 Command[1]的值确定）的数据通过 4 个 32 位寄存器地址（Data0、Data1、Data2、Data3）写入运算数据 FIFO。AES 运算模块从运算数据 FIFO 读取数据后进行加/解密运算迭代，迭代的结果被送入运算结果 FIFO。通过 APB 接口查询 Command[4]可以获知运算结果是否就绪。当 Command[4]的值为 0 时，可以从 APB 接口通过 Data0、Data1、Data2、Data3 读取运算结果 FIFO 中的运算结果。

### 19.2.2 AES 访问地址

AES 的 PCI 设备编号为(dev29,func0)，可以通过配置总线设置 AES 寄存器的基地址。物理地址的低 6 位作为偏移访问配置寄存器。

### 19.2.3 AES 寄存器描述

AES 的寄存器列表及寄存器说明如下：

表 19- 3 AES 寄存器列表

地址偏移	名称	说明
0x0	Key0	AES 密钥的 Key [31:0]，以小尾端形式存储的 AES 密钥的最低 32 位
0x4	Key1	AES 密钥的 Key [63:32]
0x8	Key2	AES 密钥的 Key [95:64]
0xc	Key3	AES 密钥的 Key [127:96]，在使用 128-bit Key 时为以小尾端形式存储的 AES 密钥的最高 32 位
0x10	Key4	AES 密钥的 Key [159:128]，仅在使用 192/256-bit 的 Key 时使用
0x14	Key5	AES 密钥的 Key [191:160]，仅在使用 192/256-bit 的 Key 时使用，在使用 192-bit Key 时为以小尾端形式存储的 AES 密钥的最高 32 位
0x18	Key6	AES 密钥的 Key [223:192]，仅在使用 256-bit 的 Key 时使用

地址偏移	名称	说明
0x1c	Key7	AES 密钥的 Key[255:224]，仅在使用 256-bit 的 Key 时使用，在使用 256-bit Key 时为以小尾端形式存储的 AES 密钥的最高 32 位
0x20	Data0	非 DMA 模式 (Command[2]=1' b0): 待加/解密数据最低 32 位的写入端口和加/解密后数据最低 32 位的读出端口 DMA 模式 (Command[2]=1' b1): Data0~Data3 不做区分，数据按照从低到高的尾端顺序写入或读出
0x24	Data1	非 DMA 模式 (Command[2]=1' b0): 待加/解密数据 63~32 位的写入端口和加/解密后数据高 63~32 位的读出端口 DMA 模式 (Command[2]=1' b1): Data0~Data3 不做区分，数据按照从低到高的尾端顺序写入或读出
0x28	Data2	非 DMA 模式 (Command[2]=1' b0): 待加/解密数据 95~64 位的写入端口和加/解密后数据高 95~64 位的读出端口 DMA 模式 (Command[2]=1' b1): Data0~Data3 不做区分，数据按照从低到高的尾端顺序写入或读出
0x2c	Data3	非 DMA 模式 (Command[2]=1' b0): 待加/解密数据的最高 32 位 (127~96) 的写入端口和加/解密后数据最高 32 位 (127~96) 的读出端口 DMA 模式 (Command[2]=1' b1): Data0~Data3 不做区分，数据按照从低到高的尾端顺序写入或读出
0x30	Ctr_init_val	使用 CTR 模式时 CTR 计数器的初始值。要使此寄存器的值发生作用需要先配置此寄存器的值，再向 cammand[0] 写入 0。
0x34	Command	命令和状态控制寄存器
0x38	Rev	保留
0x3c		

Comand 寄存器位域说明:

表 19- 4 Comand 寄存器

位域	复位值	名称	属性	说明
0	0	Ctr_mode	RW	0: 使用普通的 AES 算法进行加/解密 1: 使用 CTR 模式进行 AES 算法的加/解密
1	0	decrypt	RW	0: 加密操作 1: 解密操作
2	0	dma_start	RW	写入 1 启动 DMA 操作，写入 0 无影响 在 DMA 操作完成前此位保持为 1 当 DMA 操作完成时此位自动清零



位域	复位值	名称	属性	说明
3	0	dma_done	RW1C	当 DMA 操作完成时，此位置 1，中断输出信号变为高电平 向此位写入 1，则清零，中断输出信号变为低电平
4	1	dout_empty	RO	0: 数据读 FIFO 非空 1: 数据读 FIFO 为空
5	0	din_full	RO	0: 数据写 FIFO 未滿 1: 数据写 FIFO 已滿
7:6	0	Kr_mode	RW	0: 128-bit KEY 1: 192-bit KEY 2: 256-bit KEY 3: 保留
31:8	0	dma_count	RW	需要进行 DMA 加/解密的 128 位数的个数

## 20 eMMC 控制器

### 20.1 功能概述

龙芯 2P0500 集成了两个 SDIO/eMMC 控制器。eMMC 控制器特性如下：

- 支持eMMC启动
- 8位预分频逻辑（频率=系统时钟/(p+1)）
- DMA数据传输模式
- 专用独立DMA通道
- 1位/4位/8位的总线模式

### 20.2 访问地址及引脚复用

SDIO/eMMC 控制器内部寄存器的物理地址构成如下：

表 20- 1 eMMC 内部寄存器物理地址构成

地址	设备	备注
0x1421_0000	SDIO0/eMMC0	32KB 大小寄存器配置空间
0x1421_8000	SDIO1/eMMC1	32KB 大小寄存器配置空间

对于 eMMC 模块，使用时要注意将对应的引脚设置为相应的功能。

与 eMMC 相关的引脚复用设置可查询节 2.19 中的 eMMC 功能引脚复用关系，并配置相应 GPIO 引脚复用配置寄存器实现。

### 20.3 寄存器描述

eMMC 控制器的寄存器详细说明如下：

表 20- 2 EMMC\_CON 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
EMMC_CON	0x00	R/W	eMMC 控制寄存器	0x0

表 20- 3 EMMC\_CON 寄存器位域描述

EMMC_CON	位	缺省值	描述
Reserved	31:9	0x0	-
soft_rst	8	0x0	软件复位，整个模块复位。复位完成后硬件自动清零
Reserved	7:1	0x0	-
enclk	0	0x0	SD 时钟输出使能

表 20- 4 EMMC\_PRE 寄存器

寄存器名称	地址	读/写(R/W)	功能描述	复位值
EMMC_PRE	0x04	R/W	eMMC 预分频寄存器	0x1

表 20- 5 EMMC\_PRE 寄存器位域描述

EMMC_PRE	位	缺省值	描述
emmc_clk_rev_en	31	0x1	SDR 模式时，该位为 1，表示控制器输出的 eMMC 数据与 eMMC 时钟下降沿对齐；该位为 0，表示控制器输出的 eMMC 数据与 eMMC 时钟上升沿对齐。DDR 模式，该位必须置为 1。
Reserved	30:8	0x0	-
emmc_pre	7:0	0x1	eMMC 时钟预分频值，输出频率=PCLK/预分频值

表 20- 6 EMMC\_CMD\_ARG 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
EMMC_CMD_ARG	0x08	R/W	eMMC 命令参数寄存器	0x0

表 20- 7 EMMC\_CMD\_ARG 寄存器位域描述

EMMC_CMD_ARG	位	缺省值	描述
sdi_cmd_arg	31:0	0x0	命令参数

表 20- 8 EMMC\_CMD\_CON 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
EMMC_CMD_CON	0x0c	R/W	eMMC 命令控制寄存器	0x0

表 20- 9 EMMC\_CMD\_CON 寄存器位域描述

EMMC_CMD_CON	位	缺省值	描述
Reserved	31:18	0x0	
func_num_abort	17:15	0x0	eMMC 卡时中断的功能号，用于多块读写时，硬件自动发送停止命令。如果 auto_stop_en 为 0，则此位无效
emmc_en	14	0x0	eMMC 使能信号。用于多块读写时，硬件自动发送停止命令，为 1 时发送 CMD52，为 0 是发送 CMD12。如果 auto_stop_en 为 0，则此位无效
check_on	13	0x0	是否检查 CRC，为 1 时有效
Auto_stop_en	12	0x0	硬件自动发送停止命令，多块读写时，是否硬件自动发送停止命令，为 1 时有效
Reserved	11	0x0	-
long_rsp	10	0x0	是否为 136 位长响应，为 1 时表示长消息回复
Wait_rsp	9	0x0	决定是否主机等待响应，为 1 是表示等待消息回复
CMST	8	0x0	命令开始，置 1 时开始，命令结束后硬件自动清零
cmd_index	7:0	0x0	带开始 2 位的命令索引（共 8 位）

表 20- 10 EMMC\_CMD\_STA 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
EMMC_CMD_STA	0x10	RO	eMMC 命令状态寄存器	0x0

表 20- 11 EMMC\_CMD\_STA 寄存器位域描述

EMMC_CMD_STA	位	缺省值	描述
Reserved	31:13	0x0	-
cmd_sent_fin	14	0x0	命令发送完成（包含响应）标志位，为 1 表示命令发送完成及响应完成
auto_stop	13	0x0	硬件自动发送停止命令标志位，为 1 表示硬件自动发送停止命令，为 0 则没有
rsp_crc_err	12	0x0	响应 CRC 错误，接收到的响应 CRC 错误。为 1 时表示响应 CRC 错误，为 0 时未发现
cmd_end	11	0x0	命令发送完成（不关心响应）。为 1 时表示命令发送完成，为 0 时未完成。
cmd_tout	10	0x0	命令超时。命令响应超时（64 个时钟周期），或者 R1b 类型的命令，忙等待超时，为 1 时表示响应超时，为 0 时未超时。
rsp_fin	9	0x0	响应结束，接收完成从设备的返回信息。为 1 时表示响应结束，为 0 时未完成。
cmd_on	8	0x0	命令传输标志位。为 1 时表示传输进行中，为 0 表示结束。
rsp_index	7:0	0x0	从设备返回的带开始 2 位的响应索引（共 8 位）

表 20- 12 EMMC\_RSP0 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
EMMC_RSP0	0x14	RO	eMMC 命令响应寄存器 0	0x0

表 20- 13 EMMC\_RSP0 寄存器位域描述

EMMC_RESP0	位	缺省值	描述
sdi_resp0	31:0	0x0	卡状态[31:0]（短），卡状态[127:96]（长）长响应的配置间 sdi_cmd_con[10]

表 20- 14 EMMC\_RSP1 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
EMMC_RSP1	0x18	RO	eMMC 命令响应寄存器 1	0x0

表 20- 15 EMMC\_RESP1 寄存器位域描述

EMMC_RESP1	位	缺省值	描述
sdi_resp1	31:0	0x0	未使用（短），卡状态[95:64]（长）长响应的配置间 sdi_cmd_con[10]

表 20- 16 EMMC\_RSP2 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
EMMC_RSP2	0x1c	RO	eMMC 命令响应寄存器 2	0x0

表 20- 17 EMMC\_RSP2 寄存器位域描述

EMMC_RESP2	位	缺省值	描述
sdi_resp2	31:0	0x0	未使用（短），卡状态[63:32]（长）长响应的配置 间 sdi_cmd_con[10]

表 20- 18 EMMC\_RSP3 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
EMMC_RSP3	0x20	RO	eMMC 命令响应寄存器 3	0x0

表 20- 19 EMMC\_RSP3 寄存器位域描述

EMMC_RESP3	位	缺省值	描述
sdi_resp3	31:0	0x0	未使用（短），卡状态[31:0]（长）长响应的配置 间 sdi_cmd_con[10]

表 20- 20 EMMC\_DTIMER 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
EMMC_DTIMER	0x24	R/W	eMMC 命令数据超时寄存器	0x0

表 20- 21 EMMC\_DTIMER 寄存器位域描述

EMMC_DTIMER	位	缺省值	描述
Reserved	31:24	0x0	
sdi_dtimer	23:0	0x0	数据超时计数值，用分频后的时钟计数

表 20- 22 EMMC\_BSIZE 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
EMMC_BSIZE	0x28	R/W	EMMC 块大小寄存器	0x0

表 20- 23 EMMC\_BSIZE 寄存器位域描述

EMMC_BSIZE	位	缺省值	描述
Reserved	31:12	0x0	
sdi_bsize	11:0	0x0	块大小值（0~4095）

表 20- 24 EMMC\_DAT\_CON 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
EMMC_DAT_CON	0x2c	R/W	eMMC 数据控制寄存器	0x0

表 20- 25 EMMC\_DAT\_CON 寄存器位域描述

EMMC_DAT_CON	位	缺省值	描述
Reserved	31:21	0x0	-
wide_mode_8b	26	0x0	wide_mode_8b 为 1, wide_mode 为 0, 表示八线模式 (eMMC 模式有效)
resume_rw	20	0x0	eMMC 挂起回复读写标志位。为 1 时, eMMC 挂起后恢复之前的写操作; 为 0 时, 恢复之前的读操作
IO_resume	19	0x0	eMMC 恢复请求。在 EMMC 设备进入挂起状态后, 将此位写 1, 并且 IO_suspend 位写 0 后, EMMC 设备恢复之前的操作。
IO_suspend	18	0x0	eMMC 挂起请求。写 1 后控制器会在合适的时机发送 CMD52 命令, 通知 eMMC 设备进入挂起状态。恢复操作时需要将此位写 0.
RwaitReq	17	0x0	读等待请求。写 1 后控制器会在合适的时机将 DAT2 拉低, 通知 eMMC 设备进入读等待状态。写 0 后恢复之前的读操作。
wide_mode	16	0x0	位宽选择位。为 1 表示 4 线模式, 为 0 表示单线模式。
DMA_en	15	0x0	DMA 使能。为 1 时表示使能 DMA, 为 0 表示禁止 DMA
DTST	14	0x0	数据传输开始, 写 1 时数据传输开始, 数据传输结束后硬件清零。
Reserved	13:12	0x0	-
Blk_num	11:0	0x0	读写操作的块数。

表 20- 26 EMMC\_DAT\_CNT 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
EMMC_DAT_CNT	0x30	R/W	eMMC 数据计数寄存器	0x0

表 20- 27 EMMC\_DAT\_CNT 寄存器位域描述

EMMC_DAT_CNT	位	缺省值	描述
Reserved	31:24	0x0	
blk_num_cnt	23:12	0x0	当前传输块的字节数
blk_cnt	11:0	0x0	当前传输的块数

表 20- 28 EMMC\_DAT\_STA 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
EMMC_DAT_STA	0x34	RO	eMMC 数据状态寄存器	0x0

表 20- 29 EMMC\_DAT\_STA 寄存器位域描述

EMMC_DAT_STA	位	缺省值	描述
Reserved	31:17	0x0	-
suspend_on	16	0x0	为 1 时表示正在挂起状态
rst_suspend	15	0x0	为 1 表示正在挂起复位。用于 eMMC 设备挂起后，控制器复位 FIFO 和 DMA 请求
R1b_tout	14	0x0	为 1 表示 R1b 类型命令超时
data_start	13	0x0	为 1 表示数据传输开始
R1b_fin	12	0x0	检测到带 busy 状态的命令完成。当发送带 busy 状态的命令时，此位为 0；当 busy 状态结束时变成 1
auto_stop	11	0x0	为 1 时表示硬件正在自动发送停止命令
Reserved	10	0x0	-
r_wait_req	9	0x0	读等待发生。发送读等待请求信号到 EMMC 卡
EMMC_int	8	0x0	eMMC 中断标志位。为 1 表示检测到中断
crc_sta	7	0x0	数据发送后，从设备返回 CRC 错误
dat_crc	6	0x0	数据接收 CRC 错误
dat_tout	5	0x0	数据传输超时。为 1 时表示数据超时。
dat_fin	4	0x0	数据传输结束标志位（比如编程时）。为 1 时标志忙结束
busy_fin	3	0x0	编程错误标志位（比如编程时）。为 1 时标志忙结束
prog_err	2	0x0	编程错误标志位，为 1 时表示编程错误
tx_dat_on	1	0x0	Tx 数据发送中，为 1 时表示正在发送，为 0 时发送完成
rx_dat_on	0	0x0	Rx 数据接收中，为 1 时表示正在接收，为 0 时发送完成。

表 20- 30 EMMC\_FIFO\_STA 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
EMMC_FIFO_STA	0x38	RO	eMMC FIFO 状态寄存器	0x0

表 20- 31 EMMC\_FIFO\_STA 寄存器位域描述

EMMC_FIFO_STA	位	缺省值	描述
Reserved	31:12	0x0	-
tx_full	11	0x0	Tx FIFO 满标志位
tx_empty	10	0x0	Tx FIFO 空标志位
Reserved	9	0x0	-
rx_full	8	0x0	Rx FIFO 满标志
rx_empty	7	0x0	Rx FIFO 空标志位
Reserved	6:0	0x0	-

表 20- 32 EMMC\_INT\_MASK 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
EMMC_INT_MASK	0x3c	R/W	eMMC 中断寄存器	0x0

表 20- 33 EMMC\_INT\_MASK 寄存器位域描述

EMMC_INT_MASK	位	缺省值	描述
Reserved	31:10	0x0	-
R1b_fin_int	9	0x0	检测到 busy 结束中断，写 1 清零
rsp_crc_int	8	0x0	命令响应 CRC 错误中断，写 1 清零
cmd_tout_int	7	0x0	命令超时中断，写 1 清零
cmd_fin_int	6	0x0	发送完成中断，硬件清零
EMMC_int	5	0x0	检测到 eMMC 中断，写 1 清零
prog_err_int	4	0x0	编程错误中断，写 1 清零
crc_sta_int	3	0x0	数据发送后从设备返回 CRC 错误中断，写 1 清零
dat_crc_int	2	0x0	数据接收 CRC 错误中断，写 1 清零
dat_tout_int	1	0x0	数据超时中断，写 1 清零
dat_fin_int	0	0x0	数据完成中断，硬件清零

表 20- 34 EMMC\_DAT 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
EMMC_DAT	0x40	RO	eMMC 命令数据寄存器	0x0

表 20- 35 EMMC\_DAT 寄存器位域描述

EMMC_DAT	位	缺省值	描述
emmc_dat	31:0	0x0	eMMC 控制器发送或者接收的数据(用于 DMA 操作)

表 20- 36 EMMC\_INT\_EN 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
EMMC_INT_EN	0x64	R/W	eMMC 中断寄使能寄存器	0x0

表 20- 37 EMMC\_INT\_EN 寄存器位域描述

EMMC_INT_EN	位	缺省值	描述
Reserved	31:10	0x0	
R1b_fin_int_en	9	0x0	Busy 结束中断使能，为 1 时有效
rsp_crc_int_en	8	0x0	命令响应 CRC 错误中断使能，为 1 时有效
cmd_tout_int_en	7	0x0	命令超时中断使能，为 1 时有效
cmd_fin_int_en	6	0x0	命令发送完成中断使能，为 1 时有效
EMMC_int_en	5	0x0	eMMC 中断使能，为 1 时有效
prog_err_int_en	4	0x0	SD 卡编程错误中断使能，为 1 时有效
crc_sta_int_en	3	0x0	数据发送后从设备返回 CRC 错误中断使能，为 1 时有效
dat_crc_int_en	2	0x0	数据接收 CRC 错误中断使能，为 1 时有效
dat_tout_int_en	1	0x0	数据超时中断使能，为 1 时有效
dat_fin_int_en	0	0x0	数据完成中断使能，为 1 时有效



表 20- 38 DLL\_MASTER\_VAL 寄存器

寄存器名称	地址	读/写(R/W)	功能描述	复位值
dll_master_val	0xf0	R	DLL master 锁定值	0x0

表 20- 39 DLL\_MASTER\_VAL 寄存器位域描述

dll_master_val	位	缺省值	描述
reserved	31:4	0x0	-
dll_init_done	8	0x0	DLL master 锁定完成标志位
pm_dll_value	7: 0	0x0	DLL master 锁定值

表 20- 40 DLL\_CON 寄存器

寄存器名称	地址	读/写(R/W)	功能描述	复位值
dll_con	0xf4	R/W	DLL 控制寄存器	0x0

表 20- 41 DLL\_CON 寄存器位域描述

dll_con	位	缺省值	描述
reserved	31:30	0x0	-
resync_dll_rd	29	0x0	内部采样时钟 DLL 重同步使能位
dll_bypass_rd	28	0x0	采样时钟 DLL bypass。该位置 1，DLL 控制器将不对 DLL 参数值进行微调。
resync_dll_pad	27	0x0	pad 时钟 DLL 重同步使能位
dll_bypass_pad	26	0x0	pad 时钟 DLL bypass。该位置 1，DLL 控制器将不对 DLL 参数值进行微调。
pm_init_start	25	0x0	DLL master 初始化开始位
pm_dll_lock_mode	24	0x0	DLL master 锁定模式。0，锁定一个周期；1，锁定半个周期
pm_dll_start_point	23:16	0x0	DLL master 初始化的起点值。该值应该低于一个周期的延迟值
pm_dll_increment	15:8	0x0	DLL master 初始化的步进值
pm_dll_adj_cnt	7:0	0x0	刷新锁定值的时间间隔

表 20- 42 PARAM\_DELAY 寄存器

寄存器名称	地址	读/写(R/W)	功能描述	复位值
param_delay	0xf8	R/W	DLL 延迟参数寄存器。理想情况下，DLL 一级延迟为 100ps，共 256 级	0x0

表 20- 43 PARAM\_DELAY 寄存器位域描述

param_delay	位	缺省值	描述
reserved	31:16	0x0	-
clk_rd_delay	15:8	0x0	内部采样时钟延迟参数，用于调整控制器采样数据的采样点。DDR 模式下，该值一般比 clk_pad_delay 大。
clk_pad_delay	7:0	0x0	时钟延迟参数。DDR 模式下，此时的时钟与内部参考时钟的相位关系应该为 90°。例如，内部参考时钟为 50MHz(20ns)，此时的 clk_pad_delay 值 应该为 50 (50*100ps)。

表 20- 44 SDIO\_EMMC\_SEL 寄存器

寄存器名称	地址	读/写(R/W)	功能描述	复位值
sdio_emmc_sel	0xfc	R/W	总线模式选择	0x0

表 20- 45 SDIO\_EMMC\_SEL 寄存器位域描述

sdio_emmc_sel	位	缺省值	描述
reserved	31:4	0x0	-
bus_sel	1	0x0	SDIO 与 eMMC 总线模式选择。0 表示 eMMC 总线模式；1 表示 SDIO 总线模式
data_mode	0	0x0	数据模式选择。0，表示 SDR 数据模式；1，表示 DDR 数据模式

## 20.4 专用 DMA 控制器

### 20.4.1 结构描述

eMMC 中包含 2 个 DMA 控制器，用来实现内存与 eMMC 之间数据搬移，可以节省资源提高系统数据传输的效率。

DMA 的传送数据的过程由三个阶段组成：

1. 传送前的预处理：由 CPU 配置 DMA 描述符相关的寄存器。
2. 数据传送：在 DMA 控制器的控制下自动完成。
3. 传送结束处理：发送中断请求。

该 DMA 控制器限定为以字（4Byte）为单位的数据搬运。

DMA 控制器支持 64 位地址空间，这主要通过 dma\_64bit 来控制，当该位设置为 1 时表示 DMA 控制器工作在 64 位地址空间，反之为 32 位地址空间。在 64 位地址模式下，需要扩展 DMA\_ORDER\_ADDR 和 DMA\_SADDR 为 64 位寄存器。

### 20.4.2 DMA 描述符

#### DMA\_ORDER\_ADDR\_LOW

偏移地址： 0x0

复位值： 0x00000000

表 20- 46 DMA\_ORDER\_ADDR\_LOW 寄存器

位域	位域名称	位宽	访问	描述
31:1	dma_order_addr	31	R/W	存储器内部下一描述符地址寄存器（低 32 位）
0	Dma_order_en	1	R/W	描述符是否有效信号

说明：存储下一个 DMA 描述符的地址，dma\_order\_en 是下个 DMA 描述符的使能位，如果该位为 1 表示下个描述符有效，该位为 0 表示下个描述符无效，不执行操作，地址 16 字节对齐。在配置 DMA 描述符时，该寄存器存放的是下个描述符的地址，执行完该次 DMA 操作

后，通过判断 dma\_order\_en 信号确定是否开始下次 DMA 操作。在 64 位地址模式下，该寄存器存储低 32 位地址。

### DMA\_SADDR

偏移地址：0x4

复位值：0x00000000

表 20- 47 DMA\_SADDR 寄存器

位域	位域名称	位宽	访问	描述
31:0	dma_saddr	32	R/W	DMA 操作的系统内存地址（低 32 位）

说明：DMA 操作分为：内存读：从内存中读数据，保存在 DMA 控制器的缓存中，然后写入 I2S 设备，该寄存器指定了读内存的地址；内存写：从 I2S 设备读数据保存在 DMA 缓存中，当 DMA 缓存中的数据超过一定数目，就往内存中写，该寄存器指定了写内存的地址。在 64 位地址模式下，该寄存器存储低 32 位地址。

### DMA\_DADDR

偏移地址：0x8

复位值：0x00000000

表 20- 48 DMA\_DADDR 寄存器

位域	位域名称	位宽	访问	描述
27:0	dma_daddr	28	R/W	DMA 操作的 I2S 设备地址

### DMA\_LENGTH

偏移地址：0xc

复位值：0x00000000

表 20- 49 DMA\_LENGTH 寄存器

位域	位域名称	位宽	访问	描述
31:0	dma_length	32	R/W	传输数据长度寄存器

说明：代表一块被搬运内容的长度，单位是字。当搬运完 length 长度的字之后，开始下个 step 即下一个循环。开始新的循环，则再次搬运 length 长度的数据。当 step 变为 1，单个 DMA 描述符操作结束，开始读下个描述符。

### DMA\_STEP\_LENGTH

偏移地址：0x10

复位值：0x00000000

表 20- 50 DMA\_STEP\_LENGTH 寄存器

位域	位域名称	位宽	访问	描述
31:0	dma_step_length	32	R/W	数据传输间隔长度寄存器

说明：间隔长度说明两块被搬运内存数据块之间的长度，前一个 step 的结束地址与后一个 step 的开始地址之间的间隔。

### DMA\_STEP\_TIMES

偏移地址：0x14

复位值：0x00000000

表 20- 51 DMA\_STEP\_TIMERS 寄存器

位域	位域名称	位宽	访问	描述
31:0	dma_step_times	32	R/W	数据传输循环次数寄存器

说明：循环次数说明在一次 DMA 操作中需要搬运的块的数目。如果只想搬运一个连续的数据块，循环次数寄存器的值可以赋值为 1。

### DMA\_CMD

偏移地址：0x18

复位值：0x00000000

表 20- 52 DMA\_CMD 寄存器

位域	位域名称	位宽	访问	描述
14:13	Dma_cmd	2	R/W	源、目的地址生成方式
12	dma_r_w	1	R/W	DMA 操作类型，“1”为读 ddr2 写设备，“0”为读设备写 ddr2
11:8	dma_write_state	4	R/W	DMA 写数据状态
7:4	dma_read_state	4	R/W	DMA 读数据状态
3	dma_trans_over	1	R/W	DMA 执行完被配置的所有描述符操作
2	dma_single_trans_over	1	R/W	DMA 执行完一次描述符操作
1	dma_int	1	R/W	DMA 中断信号
0	dma_int_mask	1	R/W	DMA 中断是否被屏蔽掉

说明：dma\_single\_trans\_over=1 指一次 DMA 操作执行结束，此时 length=0 且 step\_times=1，开始取下个 DMA 操作的描述符。下个 DMA 操作的描述符地址保存在 DMA\_ORDER\_ADDR 寄存器中，如果 DMA\_ORDER\_ADDR 寄存器中 dma\_order\_en=0，则 dma\_trans\_over=1，整个 dma 操作结束，没有新的描述符要读；如果 dma\_order\_en=1，则 dma\_trans\_over 置为 0，开始读下个 dma 描述符。dma\_int 为 DMA 的中断，如果没有中断屏蔽，在一次配置的 DMA 操作结束后发生中断。CPU 处理完中断后可以直接将其置低，也可以等到 DMA 进行下次传输时自动置低。dma\_int\_mask 为对应 dma\_int 的中断屏蔽。

dma\_read\_state 说明了 DMA 当前的读状态。dma\_write\_state 说明了 DMA 当前的写状态。

DMA 写状态(WRITE\_STATE[3:0])描述，DMA 包括以下几个写状态：

表 20- 53 DMA 写状态描述

Write_state	[3:0]	描述
Write_idle	4' h0	写状态正处于空闲状态
W_ddr_wait	4' h1	Dma 判断需要执行读设备写内存操作，并发起写内存请求，但是内存还没准备好响应请求，因此 dma 一直在等待内存的响应
Write_ddr	4' h2	内存接收了 dma 写请求，但是还没有执行完写操作

Write_state	[3:0]	描述
Write_ddr_end	4' h3	内存接收了 dma 写请求, 并完成写操作, 此时 dma 处于写内存操作完成状态
Write_dma_wait	4' h4	Dma 发出将 dma 状态寄存器写回内存的请求, 等待内存接收请求
Write_dma	4' h5	内存接收写 dma 状态请求, 但是操作还未完成
Write_dma_end	4' h6	内存完成写 dma 状态操作
Write_step_end	4' h7	Dma 完成一次 length 长度的操作 (也就是说完成一个 step)

DMA 读状态(READ\_STATE[3:0])描述, DMA 包括以下几个读状态:

表 20- 54 DMA 读状态描述

Read_state	[3:0]	描述
Read_idle	4' h0	读状态正处于空闲状态
Read_ready	4' h1	接收到开始 dma 操作的 start 信号后, 进入准备好状态, 开始读描述符
Get_order	4' h2	向内存发出读描述符请求, 等待内存应答
Read_order	4' h3	内存接收读描述符请求, 正在执行读操作
Finish_order_end	4' h4	内存读完 dma 描述符
R_ddr_wait	4' h5	Dma 向内存发出读数据请求, 等待内存应答
Read_ddr	4' h6	内存接收 dma 读数据请求, 正在执行读数据操作
Read_ddr_end	4' h7	内存完成 dma 的一次读数据请求
Read_dev	4' h8	Dma 进入读设备状态
Read_dev_end	4' h9	设备返回读数据, 结束此次读设备请求
Read_step_end	4' ha	结束一次 step 操作, step times 减 1

### DMA\_ORDER\_ADDR\_HIGH

偏移地址: 0x20

复位值: 0x00000000

表 20- 55 DMA\_ADDR\_HIGH 寄存器

位域	位域名称	位宽	访问	描述
31:0	dma_order_addr	32	R/W	存储器内部下一个描述符地址寄存器(高 32 位)

### DMA\_SADDR\_HIGH

偏移地址: 0x24

复位值: 0x00000000

表 20- 56 DMA\_SADDR\_HIGH 寄存器

位域	位域名称	位宽	访问	描述
31:0	dma_saddr	32	R/W	DMA 操作的内存地址(高 32 位)

## 20.5 软件配置流程

### 20.5.1 eMMC 正常读写流程

eMMC 正常读写, 软件配置流程如下

1. 配置 emmc\_con, 使能时钟;
2. 配置 emmc\_pre, 输出预设频率的时钟;
3. 配置 emmc\_bsize, 设置一块数据的大小, 单位为字节;
4. 配置 emmc\_dtimer, 设置超时计数, 最大为 100ms;
5. 配置 emmc\_int\_en, 设置中断使能, 设置读写完成及错误中断使能;
6. 配置 emmc\_dat\_con, 设置线宽数据模式;
7. 配置 bus\_sel, 设置 DDR 或 SDR 模式;
8. eMMC 设备初始化: 初始化流程需要对命令通道进行操作, 先配置 emmc\_cmd\_arg, 填写命令参数, 再 emmc\_cmd\_con, 开始发送命令, 通过检测 emmc\_int\_msk 检查命令发送完成中断; 命令完成后读 emmc\_rsp0~3, 读 EMMC 发回来的回复;
9. 初始化完成后, 可发送数据相关命令进行数据操作。配置 DMA (注: eMMC 控制器基址加上 0x800 为 DMA 读, 基址加上 0x400 为 DMA 写) 与 eMMC 控制器。命令发送同样时通过配置 emmc\_cmd\_arg 和 emmc\_cmd\_con 来完成。数据收发是否完成, 通过检测 emmc\_int\_msk 来判断。

### 20.5.2 eMMC 初始化流程

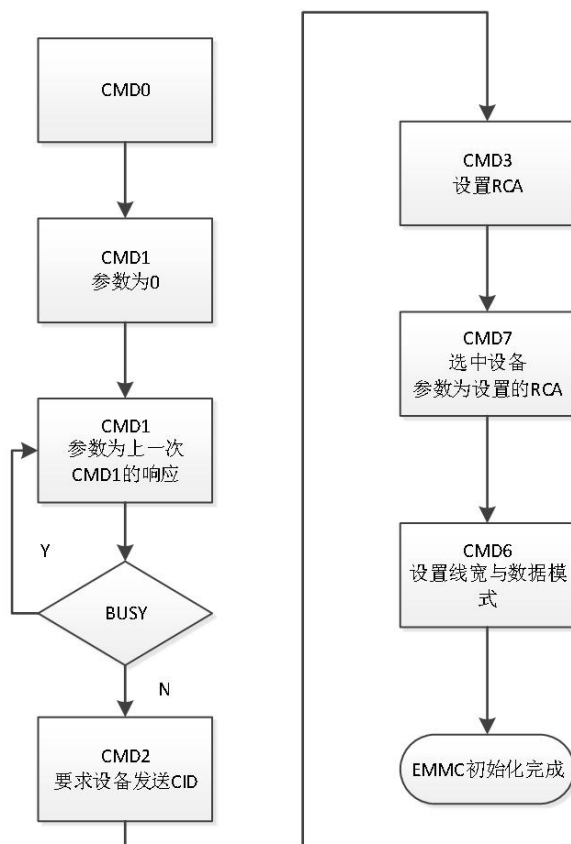


图20- 1 eMMC 初始化流程图

### 20.5.3 DDR 模式设置

在开启 DDR 模式前，需要先初始化 eMMC 设备，同时设置 eMMC 设备为 DDR 模式；然后初始化 DLL，设置延迟值，最后将控制器设置为 DDR 模式。流程如下：

1. DLL 设置：pm\_dll\_lock\_mode 置 1 锁定半个周期时钟；pm\_dll\_start\_point 置为 0x10（该值应该大于 0 小于半周期）；pm\_dll\_adj\_cnt 置为 0xff；pm\_dll\_increment，pm\_dll\_bypass，pm\_dll\_resync 置为 1；pm\_init\_start 置 1 开始 DLL 初始化；

2. DLL 锁定：DLL 设置完成后，检测 dll\_init\_done 寄存器，该值为 1 表示 DLL 完成锁定；

3. 配置延迟值：将 DLL 锁定值 pm\_dll\_value 除以 2 后的值写入 clk\_pad\_delay；clk\_rd\_delay 应该比 pm\_dll\_value/2 大，具体值视不同芯片和环境条件（PCB 走线，工作温度等）而定，软件需要根据实际情况对 clk\_rd\_delay 进行调整；

4. data\_mode 置 1，开启 DDR 模式。

## 21 SDIO 控制器

### 21.1 功能概述

龙芯 2P0500 集成了两个 SDIO/eMMC 控制器，用于 SD Memory 和 SDIO 卡的读写，支持 SD Memory 卡启动。SDIO 控制器特性如下：

- 8字（32字节）数据发送/接收FIFO
- 扩展的256位SD卡状态寄存器
- 8位预分频逻辑（频率=系统时钟/(p+1)）
- DMA数据传输模式
- 专用独立DMA通道
- 1位/4位（宽总线）的SD模式

### 21.2 访问地址及引脚复用

SDIO 控制器内部寄存器的物理地址构成如下：

表 21- 1 SDIO 内部寄存器物理地址构成

地址	设备	备注
0x1421_0000	SDI00	32KB 大小寄存器配置空间
0x1421_8000	SDI01	32KB 大小寄存器配置空间

对于 SDIO 模块，使用时要注意将对应的引脚设置为相应的功能。

与 SDIO 相关的引脚复用设置可查询节 2.19 中的 SDIO 功能引脚复用关系，并配置相应 GPIO 引脚复用配置寄存器实现。

### 21.3 寄存器描述

SDIO 控制器的寄存器详细说明如下：

表 21- 2 SDI\_CON 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
SDI_CON	0x00	R/W	SDIO 控制寄存器	0x0

表 21- 3 SDI\_CON 寄存器位域描述

SDI_CON	位	缺省值	描述
Reserved	31:9	0x0	-
soft_rst	8	0x0	软件复位，整个模块复位。复位完成后硬件自动清零
Reserved	7:1	0x0	-
enclk	0	0x0	SD 时钟输出使能



表 21- 4 SDI\_PRE 寄存器

寄存器名称	地址	读/写(R/W)	功能描述	复位值
SDI_PRE	0x04	R/W	SDIO 预分频寄存器	0x1

表 21- 5 SDI\_PRE 寄存器位域描述

SDI_PRE	位	缺省值	描述
sdio_clk_rev_en	31	0x1	SDR 模式时, 该位为 1, 表示控制器输出的 sdio 数据与 sdio 时钟下降沿对齐; 该位为 0, 表示控制器输出的 sdio 数据与 sdio 时钟上升沿对齐。DDR 模式, 该位必须置为 1。
Reserved	30:8	0x0	
Sdi_pre	7:0	0x1	SDIO 时钟预分频值, 输出频率=PCLK/预分频值

表 21- 6 SDI\_CMD\_ARG 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
SDI_CMD_ARG	0x08	R/W	SDIO 命令参数寄存器	0x0

表 21- 7 SDI\_CMD\_ARG 寄存器位域描述

SDI_CMD_ARG	位	缺省值	描述
sdio_cmd_arg	31:0	0x0	命令参数

表 21- 8 SDI\_CMD\_CON 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
SDI_CMD_CON	0x0c	R/W	SDIO 命令控制寄存器	0x0

表 21- 9 SDI\_CMD\_CON 寄存器位域描述

SDI_CMD_CON	位	缺省值	描述
Reserved	31:18	0x0	-
func_num_abort	17:15	0x0	SDIO 卡时中断的功能号, 用于多块读写时, 硬件自动发送停止命令。如果 auto_stop_en 为 0, 则此位无效
sdio_en	14	0x0	SDIO 使能信号。用于多块读写时, 硬件自动发送停止命令, 为 1 时发送 CMD52, 为 0 是发送 CMD12。如果 auto_stop_en 为 0, 则此位无效
check_on	13	0x0	是否检查 CRC, 为 1 时有效
Auto_stop_en	12	0x0	硬件自动发送停止命令, 多块读写时, 是否硬件自动发送停止命令, 为 1 时有效
Reserved	11	0x0	
long_rsp	10	0x0	是否为 136 位长响应, 为 1 时表示长消息回复
Wait_rsp	9	0x0	决定是否主机等待响应, 为 1 是表示等待消息回复
CMST	8	0x0	命令开始, 置 1 时开始, 命令结束后硬件自动清零
cmd_index	7:0	0x0	带开始 2 位的命令索引 (共 8 位)

表 21- 10 SDI\_CMD\_STA 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
SDI_CMD_STA	0x10	RO	SDIO 命令状态寄存器	0x0

表 21- 11 SDI\_CMD\_STA 寄存器位域描述

SDI_CMD_STA	位	缺省值	描述
Reserved	31:13	0x0	-
cmd_sent_fin	14	0x0	命令发送完成（包含响应）标志位，为 1 表示命令发送完成及响应完成
auto_stop	13	0x0	硬件自动发送停止命令标志位，为 1 表示硬件自动发送停止命令，为 0 则没有
rsp_crc_err	12	0x0	响应 CRC 错误，接收到的响应 CRC 错误。为 1 时表示响应 CRC 错误，为 0 时未发现
cmd_end	11	0x0	命令发送完成（不关心响应）。为 1 时表示命令发送完成，为 0 时未完成。
cmd_tout	10	0x0	命令超时。命令响应超时（64 个时钟周期），或者 R1b 类型的命令，忙等待超时，为 1 时表示响应超时，为 0 时未超时。
rsp_fin	9	0x0	响应结束，接收完成从设备的返回信息。为 1 时表示响应结束，为 0 时未完成。
cmd_on	8	0x0	命令传输标志位。为 1 时表示传输进行中，为 0 表示结束。
rsp_index	7:0	0x0	从设备返回的带开始 2 位的响应索引（共 8 位）

表 21- 12 SDI\_RSP0 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
SDI_RSP0	0x14	RO	SDIO 命令响应寄存器 0	0x0

表 21- 13 SDI\_RSP0 寄存器位域描述

SDI_RESP0	位	缺省值	描述
sdi_resp0	31:0	0x0	卡状态[31:0]（短），卡状态[127:96]（长）长响应的配置间 sdi_cmd_con[10]

表 21- 14 SDI\_RSP1 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
SDI_RSP1	0x18	RO	SDIO 命令响应寄存器 1	0x0

表 21- 15 SDI\_RSP1 寄存器位域描述

SDI_RESP1	位	缺省值	描述
sdi_resp1	31:0	0x0	未使用（短），卡状态[95:64]（长）长响应的配置间 sdi_cmd_con[10]

表 21- 16 SDI\_RSP2 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
SDI_RSP2	0x1c	RO	SDIO 命令响应寄存器 2	0x0

表 21- 17 SDI\_RSP2 寄存器位域描述

SDI_RESP2	位	缺省值	描述
sdi_resp2	31:0	0x0	未使用（短），卡状态[63:32]（长）长响应的配置间 sdi_cmd_con[10]

表 21- 18 SDI\_RSP3 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
SDI_RSP3	0x20	RO	SDIO 命令响应寄存器 3	0x0

表 21- 19 SDI\_RSP3 寄存器位域描述

SDI_RESP3	位	缺省值	描述
sdi_resp3	31:0	0x0	未使用（短），卡状态[31:0]（长）长响应的配置间 sdi_cmd_con[10]

表 21- 20 SDI\_DTIMER 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
SDI_DTIMER	0x24	R/W	SDIO 命令数据超时寄存器	0x0

表 21- 21 SDI\_DTIMER 寄存器位域描述

SDI_DTIMER	位	缺省值	描述
Reserved	31:24	0x0	-
sdi_dtimer	23:0	0x0	数据超时计数值，用分频后的时钟计数

表 21- 22 SDI\_BSIZE 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
SDI_BSIZE	0x28	R/W	SDIO 块大小寄存器	0x0

表 21- 23 SDI\_BSIZE 寄存器位域描述

SDI_BSIZE	位	缺省值	描述
Reserved	31:12	0x0	
sdi_bsize	11:0	0x0	块大小值（0~4095）

表 21- 24 SDI\_DAT\_CON 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
SDI_DAT_CON	0x2c	R/W	SDIO 数据控制寄存器	0x0

表 21- 25 SDI\_DAT\_CON 寄存器位域描述

SDI_DAT_CON	位	缺省值	描述
Reserved	31:21	0x0	-
resume_rw	20	0x0	SDIO 挂起回复读写标志位。为 1 时，SDIO 挂起后恢复之前的写操作；为 0 时，恢复之前的读操作
IO_resume	19	0x0	SDIO 恢复请求。在 SDIO 设备进入挂起状态后，将此位写 1，并且 IO_suspend 位写 0 后，SDIO 设备恢复之前的操作。
IO_suspend	18	0x0	SDIO 挂起请求。写 1 后控制器会在合适的时机发送 CMD52 命令，通知 SDIO 设备进入挂起状态。恢复操作时需要将此位写 0。
RwaitReq	17	0x0	读等待请求。写 1 后控制器会在合适的时机将 DAT2 拉低，通知 SDIO 设备进入读等待状态。写 0 后恢复之前的读操作。
wide_mode	16	0x0	位宽选择位。为 1 表示 4 线模式，为 0 表示单线模式。
DMA_en	15	0x0	DMA 使能。为 1 时表示使能 DMA，为 0 表示禁止 DMA
DTST	14	0x0	数据传输开始，写 1 时数据传输开始，数据传输结束后硬件清零。
Reserved	13:12	0x0	-
Blk_num	11:0	0x0	读写操作的块数。

表 21- 26 SDI\_DAT\_CNT 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
SDI_DAT_CNT	0x30	R/W	SDIO 数据计数寄存器	0x0

表 21- 27 SDI\_DAT\_CNT 寄存器位域描述

SDI_DAT_CNT	位	缺省值	描述
Reserved	31:24	0x0	-
blk_num_cnt	23:12	0x0	当前传输块的字节数
blk_cnt	11:0	0x0	当前传输的块数

表 21- 28 SDI\_DAT\_STA 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
SDI_DAT_STA	0x34	RO	SDIO 数据状态寄存器	0x0

表 21- 29 SDI\_DAT\_STA 寄存器位域描述

SDI_DAT_STA	位	缺省值	描述
Reserved	31:17	0x0	-
suspend_on	16	0x0	为 1 时表示正在挂起状态

SDI_DAT_STA	位	缺省值	描述
rst_suspend	15	0x0	为 1 表示正在挂起复位。用于 SDIO 设备挂起后，控制器复位 FIFO 和 DMA 请求
R1b_tout	14	0x0	为 1 表示 R1b 类型命令超时
data_start	13	0x0	为 1 表示数据传输开始
R1b_fin	12	0x0	检测到带 busy 状态的命令完成。当发送带 busy 状态的命令时，此位为 0；当 busy 状态结束时变成 1
auto_stop	11	0x0	为 1 时表示硬件正在自动发送停止命令
Reserved	10	0x0	-
r_wait_req	9	0x0	读等待发生。发送读等待请求信号到 SDIO 卡
SDIO_int	8	0x0	SDIO 中断标志位。为 1 表示检测到中断
crc_sta	7	0x0	数据发送后，从设备返回 CRC 错误
dat_crc	6	0x0	数据接收 CRC 错误
dat_tout	5	0x0	数据传输超时。为 1 时表示数据超时。
dat_fin	4	0x0	数据传输结束标志位（比如编程时）。为 1 时标志忙结束
busy_fin	3	0x0	编程错误标志位（比如编程时）。为 1 时标志忙结束
prog_err	2	0x0	编程错误标志位，为 1 时表示编程错误
tx_dat_on	1	0x0	Tx 数据发送中，为 1 时表示正在发送，为 0 时发送完成
rx_dat_on	0	0x0	Rx 数据接收中，为 1 时表示正在接收，为 0 时发送完成。

表 21- 30 SDI\_FIFO\_STA 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
SDI_FIFO_STA	0x38	RO	SDIO FIFO 状态寄存器	0x0

表 21- 31 SDI\_FIFO\_STA 寄存器位域描述

SDI_FIFO_STA	位	缺省值	描述
Reserved	31:12	0x0	-
tx_full	11	0x0	Tx FIFO 满标志位
tx_empty	10	0x0	Tx FIFO 空标志位
Reserved	9	0x0	-
rx_full	8	0x0	Rx FIFO 满标志
rx_empty	7	0x0	Rx FIFO 空标志位
Reserved	6:0	0x0	-

表 21- 32 SDI\_INT\_MASK 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
SDI_INT_MASK	0x3c	R/W	SDIO 中断寄存器	0x0

表 21- 33 SDI\_INT\_MASK 寄存器位域描述

SDI_INT_MASK	位	缺省值	描述
Reserved	31:10	0x0	
R1b_fin_int	9	0x0	检测到 busy 结束中断，写 1 清零
rsp_crc_int	8	0x0	命令响应 CRC 错误中断，写 1 清零
cmd_tout_int	7	0x0	命令超时中断，写 1 清零
cmd_fin_int	6	0x0	发送完成中断，硬件清零
SDIO_int	5	0x0	检测到 SDIO 中断，写 1 清零
prog_err_int	4	0x0	SD 卡编程错误中断，写 1 清零
crc_sta_int	3	0x0	数据发送后从设备返回 CRC 错误中断，写 1 清零
dat_crc_int	2	0x0	数据接收 CRC 错误中断，写 1 清零
dat_tout_int	1	0x0	数据超时中断，写 1 清零
dat_fin_int	0	0x0	数据完成中断，硬件清零

表 21- 34 SDI\_DAT 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
SDI_DAT	0x40	RO	SDIO 命令数据寄存器	0x0

表 21- 35 SDI\_DAT 寄存器位域描述

SDI_DAT	位	缺省值	描述
sdi_dat	31:0	0x0	SDIO 控制器发送或者接收的数据（用于 DMA 操作）

表 21- 36 SDI\_INT\_EN 寄存器

寄存器名称	偏移地址	读/写(R/W)	功能描述	复位值
SDI_INT_EN	0x64	R/W	SDIO 中断寄使能寄存器	0x0

表 21- 37 SDI\_INT\_EN 寄存器位域描述

SDI_INT_EN	位	缺省值	描述
Reserved	31:10	0x0	-
R1b_fin_int_en	9	0x0	Busy 结束中断使能，为 1 时有效
rsp_crc_int_en	8	0x0	命令响应 CRC 错误中断使能，为 1 时有效
cmd_tout_int_en	7	0x0	命令超时中断使能，为 1 时有效
cmd_fin_int_en	6	0x0	命令发送完成中断使能，为 1 时有效
SDIO_int_en	5	0x0	SDIO 中断使能，为 1 时有效
prog_err_int_en	4	0x0	SD 卡编程错误中断使能，为 1 时有效
crc_sta_int_en	3	0x0	数据发送后从设备返回 CRC 错误中断使能，为 1 时有效
dat_cec_int_en	2	0x0	数据接收 CRC 错误中断使能，为 1 时有效
dat_tout_int_en	1	0x0	数据超时中断使能，为 1 时有效
dat_fin_int_en	0	0x0	数据完成中断使能，为 1 时有效

表 21- 38 dll\_master\_val 寄存器

寄存器名称	地址	读/写(R/W)	功能描述	复位值
dll_master_val	0xf0	R	DLL master 锁定值	0x0

表 21- 39 dll\_master\_val 寄存器位域描述

dll_master_val	位	缺省值	描述
reserved	31:4	0x0	
dll_init_done	8	0x0	DLL master 锁定完成标志位
pm_dll_value	7: 0	0x0	DLL master 锁定值

表 21- 40 dll\_con 寄存器

寄存器名称	地址	读/写(R/W)	功能描述	复位值
dll_con	0xf4	R/W	DLL 控制寄存器	0x0

表 21- 41 dll\_con 寄存器位域描述

dll_con	位	缺省值	描述
reserved	31:30	0x0	-
resync_dll_rd	29	0x0	内部采样时钟 DLL 重同步使能位
dll_bypass_rd	28	0x0	采样时钟 DLL bypass。该位置 1，DLL 控制器将不对 DLL 参数值进行微调。
resync_dll_pad	27	0x0	pad 时钟 DLL 重同步使能位
dll_bypass_pad	26	0x0	pad 时钟 DLL bypass。该位置 1，DLL 控制器将不对 DLL 参数值进行微调。
pm_init_start	25	0x0	DLL master 初始化开始位
pm_dll_lock_mode	24	0x0	DLL master 锁定模式。0，锁定一个周期；1，锁定半个周期
pm_dll_start_point	23:16	0x0	DLL master 初始化的起点值。该值应该低于一个周期的延迟值
pm_dll_increment	15:8	0x0	DLL master 初始化的步进值
pm_dll_adj_cnt	7:0	0x0	刷新锁定值的时间间隔

表 21- 42 param\_delay 寄存器

寄存器名称	地址	读/写(R/W)	功能描述	复位值
param_delay	0xf8	R/W	DLL 延迟参数寄存器。理想情况下，DLL 一级延迟为 100ps，共 256 级	0x0

表 21- 43 param\_delay 寄存器位域描述

param_delay	位	缺省值	描述
reserved	31:16	0x0	-
clk_rd_delay	15:8	0x0	内部采样时钟延迟参数，用于调整控制器采样数据的采样点。DDR 模式下，该值一般比 clk_pad_delay 大。
clk_pad_delay	7:0	0x0	时钟延迟参数。DDR 模式下，此时的时钟与内部参考时钟的相位关系应该为 90°。例如，内部参考时钟为 50MHz(20ns)，此时的 clk_pad_delay 值 应该为 50（50*100ps）。

表 21- 44 sdio\_emmc\_sel 寄存器

寄存器名称	地址	读/写(R/W)	功能描述	复位值
sdio_emmc_sel	0xfc	R/W	总线模式选择	0x0

表 21- 45 sdio\_emmc\_sel 寄存器位域描述

sdio_emmc_sel	位	缺省值	描述
reserved	31:4	0x0	-
bus_sel	1	0x0	SDIO 与 EMMC 总线模式选择。0 表示 SDIO 总线模式；1 表示 EMMC 总线模式。切换至 EMMC 模式时，EMMC 最多只能支持四线模式。
data_mode	0	0x0	数据模式选择。0，表示 SDR 数据模式；1，表示 DDR 数据模式

## 21.4 软件编程指南

### 21.4.1 SD Memory 卡软件编程说明

SD Memory 卡要想正常工作，必须先初始化。初始化的过程需要发送不同的命令序列来配置从设备。

初始化完成之后就可以正常工作了。

配置寄存器的流程如下：

1. 配置 sdi\_con，使能输出时钟。
2. 配置 sdi\_pre，设置一个分频系数，如果时序不满足，可以设置输出反向时钟来调整时序。

3. 配置 sdi\_int\_en，使能命令、数据完成及其他中断。

4. 按照上图初始化流程初始化控制器

发送命令的配置寄存器过程如下：

- 根据发送的命令，配置 cmd\_arg 寄存器
- 配置 cmd\_con 寄存器，发送命令
- 读 sdi\_int\_msk 寄存器，检查是否传输完成，是否有错误



- 如果需要，读 sdi\_rsp 寄存器

初始化的流程如下：

CMD0 → CMD8 → ACMD41(即 CMD55 → CMD41) → CMD2 → CMD3

→ CMD7 → ACMD6（用于配置是否用 4bit 数据线传输）。

5. 进行数据操作之前需要配置 Bsize 寄存器，Dtimer 寄存器。

6. 数据的操作必须要配置 DMA，配置 dat\_con 寄存器并配置 DMA（注：SDIO 控制器基址加上 0x800 为 DMA 读，基址加上 0x400 为 DMA 写，寄存器与 eMMC 专用 DMA 相同）。

7. 读 sdi\_int\_msk 寄存器，检测是否传输完成，是否有错误。

8. 没有错误则完成一次数据传输，不需要软件发送停止命令。

## 21.4.2 SDIO 卡软件编程说明

SDIO 卡的初始化流程和 SD memory 卡不同，其初始化流程如下：

1. 配置 sdi\_con，使能输出时钟。
2. 配置 sdi\_pre，设置一个分频系数，如果时序不满足，可以设置输出反向。
3. 时钟来调整时序。
4. 配置 sdi\_int\_en，使能命令、数据完成及其他中断。
5. 初始化流程如下：

发送命令的配置寄存器过程如下：

- 根据发送的命令，配置 cmd\_arg 寄存器
- 配置 cmd\_con 寄存器，发送命令
- 读 sdi\_int\_msk 寄存器，检查是否传输完成，是否有错误
- 如果需要，读 sdi\_rsp 寄存器

初始化的流程如下（对 CCCR 的操作）：

CMD52（复位） CMD5（等待上电完成） CMD3(获取 RCA) CMD7（选择相应 RCA 的卡） CMD52(配置是否用 4bit 数据线传输) CMD52(配置读写数据的块大小) CMD52（打开 IO 中断使能）进行数据操作之前需要配置 Bsize 寄存器，Dtimer 寄存器。

6. 数据的操作必须要配置 DMA，配置 dat\_con 寄存器并配置 DMA（注：读操作时先配置 DMA，写操作时先配置 dat\_con）。

7. 发送读写数据的命令时，如果需要硬件自动发送停止命令，需要配置 auto\_stop 和 sdio\_en。读写数据时需要先读写支持的 Function，配置相应的 FBR 的指针寄存器，再发送多块读写（CMD53）或者单块读写（CMD52）命令进行读写。

8. 读 sdi\_int\_msk 寄存器，检测是否传输完成，是否有错误。

9. 没有错误则完成一次数据传输，不需要软件发送停止命令。

10. 如果检测到 IO 中断，控制器会置起响应的中断，但是不会停止当前的操作。

11. 对于读等待，控制器会在合适的时机将 DAT2 拉低，通知 SDIO 卡停止发送数据。所以如果当前正在传输数据过程中，控制器可能会在当前一块数据传输结束后，发出读等待信号，这时才不会接收下一块数据。

12. 对于挂起和恢复操作。有可能出现挂起嵌套情况，比如说操作 1 被操作 2 中断而挂起，然后操作 2 被操作 3 中断而挂起。挂起时中断的现场需要软件保存（如当前的读写标志位，当前传输的块数，地址等），进行入栈操作。恢复时需要软件再按相应的顺序出栈。

### 21.4.3 DDR 模式设置

在开启 DDR 模式前，需要先初始化 sdio 设备，同时设置 sdio 设备为 DDR 模式；然后初始化 DLL，设置延迟值，最后将控制器设置为 DDR 模式。流程如下：

1. DLL 设置：pm\_dll\_lock\_mode 置 1 锁定半个周期时钟；pm\_dll\_start\_point 置为 0x10（该值应该大于 0 小于半周期）；pm\_dll\_adj\_cnt 置为 0xff；pm\_dll\_increment，pm\_dll\_bypass，pm\_dll\_resync 置为 1；pm\_init\_start 置 1 开始 DLL 初始化；

2. DLL 锁定：DLL 设置完成后，检测 dll\_init\_done 寄存器，该值为 1 表示 DLL 完成锁定；

3. 配置延迟值：将 DLL 锁定值 pm\_dll\_value 除以 2 后的值写入 clk\_pad\_delay；clk\_rd\_delay 应该比 pm\_dll\_value/2 大，具体值视不同芯片和环境条件（PCB 走线，工作温度等）而定，软件需要根据实际情况对 clk\_rd\_delay 进行调整；

4. data\_mode 置 1，开启 DDR 模式。

## 21.5 支持 SDIO 型号

本节列出经过验证的可支持的 SDIO 卡型号，其它类型的 SDIO 卡未经验证，不保证与本控制器兼容。

- Mem 卡：Kingston SD-CO2G SDC/2GB
- IO 卡（wifi）：maxwell sd8686

## 22 PWM 控制器

### 22.1 概述

2P0500 芯片实现了 40 路脉冲宽度调节/计数控制器，以下简称 PWM。每一路 PWM 工作和控制方式完全相同。每路 PWM 有一路脉冲宽度输出信号和一路待测脉冲输入信号。系统时钟高达 125MHz，计数寄存器和参考寄存器均 32 位数据宽度。

### 22.2 访问地址及引脚复用

PWM 控制器内部寄存器的物理地址构成如下：

表 22- 1 PWM 寄存器地址

地址	设备	备注
0x1420_7000	MAIN_PWM0~15	每个 PWM 占用 16B 寄存器配置空间： 0x1420_7000-pwm0, 0x1420_7010-pwm1, ...
0x1510_7000	PRT_PWM0~23	每个 PWM 占用 16B 寄存器配置空间： 0x1510_7000-pwm0, 0x1510_7010-pwm1, ...

对于 PWM 模块，使用时要注意将对应的引脚设置为相应的功能。与 PWM 相关的引脚复用设置可查询 PWM 功能引脚复用关系，并配置相应 GPIO 引脚复用配置寄存器实现。

### 22.3 寄存器描述

每路控制器共有五个寄存器，具体描述如下：

表 22- 2 PWM 寄存器列表

名称	地址	宽度	访问	说明
Low_buffer	Base + 0x4	32	R/W	低脉冲缓冲寄存器
Full_buffer	Base + 0x8	32	R/W	脉冲周期缓冲寄存器
CTRL	Base + 0xC	11	R/W	控制寄存器

表 22- 3 PWM 控制寄存器设置

位域	名称	访问	复位值	说明
0	EN	R/W	0	计数器使能位 置 1 时：CNTR 用来计数 置 0 时：CNTR 停止计数（输出保持）
2: 1		Reserved	2'b0	预留
3	OE	R/W	0	脉冲输出使能控制位,低有效 置 0 时：脉冲输出使能 置 1 时：脉冲输出屏蔽
4	SINGLE	R/W	0	单脉冲控制位 置 1 时：脉冲仅产生一次 置 0 时：脉冲持续产生

位域	名称	访问	复位值	说明
5	INTE	R/W	0	中断使能位 置 1 时：当 full_pulse 到 1 时送中断 置 0 时：不产生中断
6	INT	R/W	0	中断位 读操作：1 表示有中断产生,0 表示没有中断 写入 1：清中断
7	RST	R/W	0	使得 Low_level 和 full_pulse 计数器重置 置 1 时：计数器重置（从 buffer 读，输出低电平） 置 0 时：计数器正常工作
8	CAPTE	R/W	0	测量脉冲使能 置 1 时：测量脉冲模式 置 0 时：非测量脉冲模式（一般而言则是脉冲输出模式）
9	INVERT	R/W	0	输出翻转使能 置 1 时：使脉冲在输出出去发生信号翻转（周期以高电平开始） 置 0 时：使脉冲保持原始输出（周期以低电平开始）
10	DZONE	R/W	0	防死区功能使能 置 1 时：该计数模块需要启用防死区功能 置 0 时：该模块无需防死区功能

## 22.4 功能说明

### 22.4.1 脉宽调制功能

Low\_buffer 和 Full\_buffer 寄存器可以由系统编程写入获得初始值。系统编程写入完毕后，模块内部的 low\_level 和 full\_pulse 寄存器分别从 Low\_buffer 和 Full\_buffer 缓冲寄存器中读取初值，之后在系统时钟驱动下不断自减（初始输出低电平）。当 low\_level 寄存器到达 1 之后，输出变为高电平，此时 full\_pulse 仍在自减。当 full\_pulse 寄存器到达 1 之后，输出变为低电平，low\_level 和 full\_pulse 又分别从 Low\_buffer 和 Full\_buffer 缓冲寄存器中读取初值，然后重新开始不断自减，控制器就产生连续不断的脉冲宽度输出。当 full\_pulse 寄存器的值等于 1 的时候，可以配置产生一个中断，从而作为定时器使用。

例：如果要产生宽度为系统时钟周期 50 倍的高脉宽和 90 倍的低脉宽，在 low\_buffer 中应该配置初始值 90，在 full\_buffer 寄存器中配置初始值(50+90)=140。

值得说明的是，由于两个缓冲寄存器的写入有先后之分，在某些特殊的情况下（比如写入时刻刚好是旧脉冲结束时）会使得输出脉冲有异于预期。推荐的做法是在向缓冲寄存器写入新数前，将控制寄存器 EN 位写 0，在写入新数之后再写 EN 位写 1。值得说明的是，即使没有重写 EN 位，紊乱的脉冲输出最多只会维持一个周期。

如果对两个缓冲寄存器都写 0，则输出为低电平；如果对 low\_buffer 写 0，对 full\_buffer 写 1，则输出高电平；如果写入 Low\_buffer 的值不小于 full\_buffer，则输出低电平。但这三类数值都是不推荐的。

此外，缓冲寄存器的数值写入应当先于 CTRL 控制寄存器。

### 22.4.2 脉冲测量功能

待测脉冲信号连在 PWM 输入信号接口上，在设置完 CTRL 控制寄存器后，在系统时钟的驱动下，Low\_level 和 full\_pulse 寄存器开始不断自增。当检测到输入脉冲信号上跳变时，将 Low\_level 寄存器的值传送到 low\_buffer 寄存器中；当检测到输入脉冲信号下跳变时，将 full\_pulse 寄存器的值传送到 full\_buffer 寄存器中，并将 Low\_level 和 full\_pulse 寄存器置 1，重新开始计数。

例：如果要输入脉冲为系统时钟 50 倍的高脉宽和 90 倍的低脉宽，在 low\_buffer 中最终读出的值为 90，在 full\_buffer 寄存器中读出的值为(50+90)=140。

待测脉冲应当是周期信号，且脉冲周期不应超出 32 位计数器能计量的范围。

每次测量均是从下跳变开始，到下一个下跳变结束。由于测量及缓冲的需要，在连续测量两个脉冲周期后，low\_buffer 和 full\_buffer 寄存器中存储的才是正确的脉冲参数。

若出现持续的周期超过 0xFFFF\_FFF9 的脉冲，控制寄存器 INT 位会被置 1，表示待测脉冲超出了计量范围。

### 22.4.3 防死区功能

四路 PWM 都配备了防死区功能，可以防止四路脉冲输出同时发生跳变。

将四路模块分别标记为 PWM\_0、PWM\_1、PWM\_2、PWM\_3，它们的优先级为 0>1>2>3，即若要同时产生跳变，在 PWM\_0 跳变之后 PWM\_1 才能跳变（低优先级的信号被“抹去”一个或多个系统时钟），依此类推。该优先级是固化的，不可配置。

一个典型的防死区示例如下（PWM\_\*为未开防死区的输出，PWM\_\*'为打开防死区后的输出）：

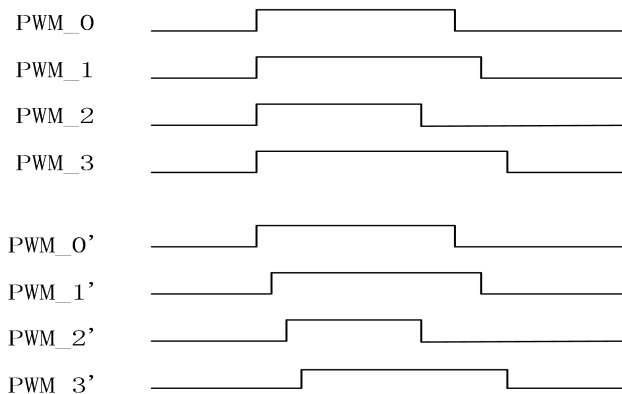


图22- 1 防死区功能

## 23 HPET 控制器

### 23.1 概述

2P0500 芯片实现了 4 个 HPET 定时控制器，HPET（High Precision Event Timer，高精度事件定时器）定义了一组新的定时器，这组定时器被操作系统使用，用来给线程调度，内核以及多媒体定时器服务器等产生中断。操作系统可以将不同的定时器分配给不同的应用程序使用。通过配置，每个定时器都能独立产生中断。

这组定时器由一个向上累加的主计时器（up-counter）以及一组比较器构成。这个计时器以各系统下 APB 总线频率向上累加，因此当软件两次读取计时器的值时，除非遇到计时器溢出，否则第二次读取的值总是比第一次读取的值大。而每个定时器都包含一个 match 寄存器以及一个比较器。当 match 寄存器的值与主计时器相等时，那么定时器产生中断。部分定时器可产生周期性中断。

HPET 模块包括一个主计数器（main count）以及三个比较器（comparator），且他们的宽度都是 32 位。在这三个比较器中，有且仅有一个比较器支持周期性中断（periodic-capable）；这三个比较器都支持非周期性中断。

### 23.2 访问地址

HPET 控制器内部寄存器的物理地址构成如下：

表 23- 1 HPET 寄存器地址

地址	设备	备注
0x1420_4000	HPET0	2KB 寄存器配置空间
0x1420_4800	HPET1	2KB 寄存器配置空间
0x1510_b000	HPET2	4KB 寄存器配置空间
0x1520_5000	HPET3	4KB 寄存器配置空间

### 23.3 寄存器描述

下表列出了 HPET 的寄存器：

表 23- 2 HPET 寄存器

寄存器偏移地址	寄存器	类型
000-007h	General Capabilities and ID Register	只读
008-00Fh	Reserved	
010-017h	General Configuration Register	读/写
018-01Fh	Reserved	R/WC
020-027h	General Interrupt Status Register	R/W
028-0EFh	Reserved	
0F0-0F7h	Main Counter Value Register	R/W

寄存器偏移地址	寄存器	类型
100-107h	Timer 0 Configuration and Capability Register	R/W
108-10Fh	Timer 0 Comparator Value Register	R/W
110-11Fh	Reserved	
120-127h	Timer 1 Configuration and Capability Register	R/W
128-12Fh	Timer 1 Comparator Value Register	R/W
130-13Fh	Reserved	
140-147h	Timer 2 Configuration and Capability Register	R/W
148-14Fh	Timer 2 Comparator Value Register	R/W
150-15Fh	Reserved	

若系统在状态转换过程中需要保存这些寄存器的值以便随后恢复，那么操作系统负责保存这些寄存器的值，硬件无需保存这些寄存器的值。因此当系统处于 S3, S4, S5 状态时，这些寄存器无需维持。

### 23.3.1 General Capabilities and ID Register

表 23- 3 General Capabilities ID 寄存器

位	名称	描述	读写特性
63: 32	COUNTER_CLK_PERIOD	Main Counter Tick Period: 这个域标示了主计时器的计时频率，以 fps ( $10^{-15}$ s) 为单位。这个值必须大于 0，且小于或等于 05F5E100 (100ns, 即 10MHz)	RO
31: 16	VENDOR_ID		RO
15: 14	Reserved		
13	COUNT_SIZE_CAP	Counter Size:主计时器的宽度; 0: 32 bits 1: 64 bits	RO
12:8	NUM_TIM_CAP	Num of Timer: 定时器的个数; 这个域的值指示最后一个定时器的编号, GS 南桥芯片的 HPET 有三个定时器, 因此这个域的值是 2。	RO
7:0	REV_ID	版本号; 不可为 0	RO

### 23.3.2 General Configuration Register

表 23- 4 General Configuration 寄存器

位	名称	描述	读写特性
63: 1	Reserved		
0	ENABLE_CNF	Overall Enable; 用来使能所有定时器产生中断。如果为 0, 主计时器停止计时且所有的定时器都不再产生中断。 0: 主计时器停止计时且所有的定时器都不再产生中断; 1: 主计时器计时且允许定时器产生中断;	R/W

### 23.3.3 General Interrupt Status Register

表 23- 5 General Interrupt Status 寄存器

位	名称	描述	读写特性
63: 3	Reserved		
2	T2_INT_STS	Timer 2 Interrupt Active:功能同 T0_INT_STS	R/WC
1	T1_INT_STS	Timer 1 Interrupt Active:功能同 T0_INT_STS	R/WC
0	T0_INT_STS	<p>Timer 0 Interrupt Active:功能依赖于这个定时器的中断触发模式是电平触发还是边沿触发:</p> <p>如果是电平触发模式:</p> <p>    这位默认是 0。当对应的定时器发生中断, 那么有硬件将其置 1. 一旦被置位, 软件往这位写 1 将会清空这位。往这位写 0, 则无意义。</p> <p>如果边沿触发模式:</p> <p>    软件将忽略这位。软件通常往这位写 0.</p>	R/WC

各个定时器的中断触发模式由各自 Configuration and Capability 寄存器的 Tn\_TYPE\_CNF 位确定。

### 23.3.4 Main Counter Value Register

表 23- 6 Main Counter Value 寄存器

位	名称	描述	读写特性
63: 32	Reserved		
31: 0	Main_Counter_Val	主计时器的值; 只有当主计时器停止计时时, 才允许修改这个寄存器的值。	R/W

### 23.3.5 Timer N Configuration and Capabilities Register

表 23- 7 Timer N Configuration Capabilities 寄存器

位	名称	描述	读写特性
63: 9	Reserved		
8	Tn_32MODE_CNF	Timer n 32-bit 模式 (N 为 0-2)。当定时器为 32 位时, 这位为 0, 且只读	RO
7	Reserved		RO
6	Tn_VAL_SET_CNF	<p>Timer N Value Set (N 为 0-2): 只有能产生周期性中断的定时器才会使用这个域。通过对这位写 1, 软件能直接修改周期性定时器的累加器。软件无需对这位清 0</p> <p>GS 南桥芯片中只有 Timer 0 能产生周期性中断, 因此对 Timer0 来讲, 这位是可读可写。而对于 Timer1, Timer2, 这位默认为 0, 且为只读。</p>	R/W



位	名称	描述	读写特性
5	Tn_SIZE_CAP	Timer N Size; Timer N 的宽度 (N 为 0-2)。 0: 32 位宽。	RO
4	Tn_PER_INT_CAP	Timer N Periodic Interrupt Capable (N 为 0-2) : 1: 定时器能产生周期性中断; 0: 定时器不能产生周期性中断;	RO
3	Tn_TYPE_CNF	Timer N type (N 为 0-2) : 如果对应的 Tn_PER_INT_CAP 位为 0, 那么这位为只读, 且默认为 0。 若对应的 Tn_PER_INT_CAP 位为 1, 那么这位可读可写。 用作使能相应的定时器产生周期性中断。 1: 使能定时器产生周期性中断 0: 使能定时器产生非周期性中断	R/W
2	Tn_INT_ENB_CNF	Timer N interrupt Enable (N 为 0-2) :使能定时器产生中断	R/W
1	Tn_INT_TYPE_CNF	Timer N Interrupt Type (N 为 0-2) : 0: 定时器的中断触发模式为边沿触发; 这意味着对应的定时器将产生边沿触发中断。若另外的的中断产生, 那么将产生另外的边沿。 1: 定时器的中断触发模式为电平触发; 这意味着对应的定时器将产生电平触发中断。这个中断将一直有效直到被软件清掉 (General Interrupt Status Register)。	
0	Reserved		

### 23.3.6 Timer N Comparator Value Register

表 23- 8 Timer N Comparator Value 寄存器

位	名称	描述	读写特性
63: 32	Reserved		
31: 0	Tn_Com_VAL	Tn_Comparator value (N 为 0-2) : 定时器比较器的值; 当对应的定时器配置为非周期性模式时: ◆ 这个寄存器的值将与主计时器寄存器的值做比较; ◆ 若主计时器的值与比较器的值相等时, 则产生定时中断 (如果; 对应的中断使能打开)。 ◆ 比较器的值不会因为中断的产生而发生变化 若对应的定时器配置为周期性模式时: 当主计时器的值域比较器的值相等时, 产生中断 (如果对应的中断使能被打开); 如果产生中断, 那么比较器的值将累加最后一次软件写入比较器的值。比如当比较器的值被写入 0x0123h,	R/W

位	名称	描述	读写特性
		<p>那么当主计时器的值为 0x123h 时，产生中断； 比较器的值被硬件修改为 0x246h； 当主计时器的值达到 0x246h 时，产生另外一个中断； 比较器的值被硬件修改为 0x369h。</p> <p>◆ 只要产生中断，那么比较器的值都会累加；直到比较器的值达到最大 (0xffffffff)，那么累加器的值将会继续累加。比如当比较器的值是 FFFF0000h，而最后一次由软件写入比较器的值是 20000。当中断发生后，比较器的值变为 00010000h。</p>	

## 24 DA 控制器

### 24.1 概述

DA 电路是一种数字模拟信号转换电路，包含 4 路模拟信号输出通道，支持 8 位数字转换码，转换速率最高为 0.5MSPS，模拟电压输出范围 0~3.3V。

### 24.2 寄存器描述

DA 模块寄存器位于 0x15105000——0x15105fff 的 4KB 地址空间内，其基地址为 0x15105000，所有寄存器位宽均为 32 位。

#### 24.2.1 DAC 使能寄存器 (DACEN)

偏移 : 0x80  
复位值 : 32' 0

表 24- 1 DACEN 寄存器

位域	名称	访问	描述
31:26	Reserved	-	
25:16	DAC_CD	RW	<b>DAC clock divider</b> DAC 时钟分频系数, N+1
15:5	Reserved	-	
4	DAC_EN	RW	<b>DAC enable</b> DAC 使能
3	DCH_EN3	RW	<b>DAC channel 3 enable</b> DAC 通道 3 使能
2	DCH_EN2	RW	<b>DAC channel 2 enable</b> DAC 通道 2 使能
1	DCH_EN1	RW	<b>DAC channel 1 enable</b> DAC 通道 1 使能
0	DCH_EN0	RW	<b>DAC channel 0 enable</b> DAC 通道 0 使能

#### 24.2.2 DAC 数值寄存器 (DACDATA)

偏移 : 0x84  
复位值 : 32' 0

表 24- 2 DACDATA 寄存器

位域	名称	访问	描述
31:24	CH3	RW	<b>DAC 3 output data</b> DAC 3 输出数值
23:16	CH2	RW	<b>DAC 2 output data</b> DAC 2 输出数值

位域	名称	访问	描述
15:8	CH1	RW	<b>DAC 1 output data</b> DAC 1 输出数值
7:0	CH0	RW	<b>DAC 0 output data</b> DAC 0 输出数值

## 25 电源管理模块

### 25.1 概述

龙芯 2P0500 电源管理模块提供系统功耗管理实现机制。

- 系统待机、休眠与唤醒，支持多种唤醒方式（网络，OTG，电源开关等）
- 支持 Dynamic Power Management（DPM），动态性能功耗控制，支持动态关闭 NODE（CORE+SCACHE）、DDR、IMAGE、SCAN、SYS 等电源域电源。
- 支持 Dynamic Frequency Scaling（DFS），处理器核 DFS 控制，由片内打印 LA132 处理器核独立控制。
- 系统时钟控制，模块时钟门控，多种方式调节频率。
- 提供温度管理控制功能。支持 3 级报警机制。

### 25.2 动态电源管理

龙芯 2P0500 支持芯片内部各大功能模块动态时钟门控、电源开关管理功能，主要包括动态频率调节 (DFS) 和动态电源管理 (DPM) 技术。

芯片内部各电源域划分，及各电源域包含功能模块，如下表：

表 25- 1 芯片各电源域划分

LS2P05xx			
各电源域		包含主要功能模块	
1	常开域(Always-On)	打印系统	PRINT-LA132、PRT-SRAM、 PRT-DEVs(UART、I2C、GPIO、DA、PWM、PM0IO、SPI、 HPET、PRINTer)
		唤醒模块	OTG、GMAC0
2	NODE 电源域	LA364、Scache、IODMA、L1XBAR	
3	DDR 电源域	DDR3-Controller、DDR3-PHY	
4	IMAGE 电源域	JBIG、JPEG	
5	SCAN 电源域	SCAN-LA132、 SCA-DEVs(UART、GPIO、PM1IO、SPI、HPET、SCANNer)	
6	SYS+NETWORK 电源域	系统主互联	NETWORK
		主系统设备	BOOT(SPI0+CONFBUS)、 GMAC1、USB1/2、 UART、I2C、GPIO、SPI、HPET、PM2IO、PWM、SDIO、 EMMC、ENCRYPT

其中，具体电源域划分示意图如下图：

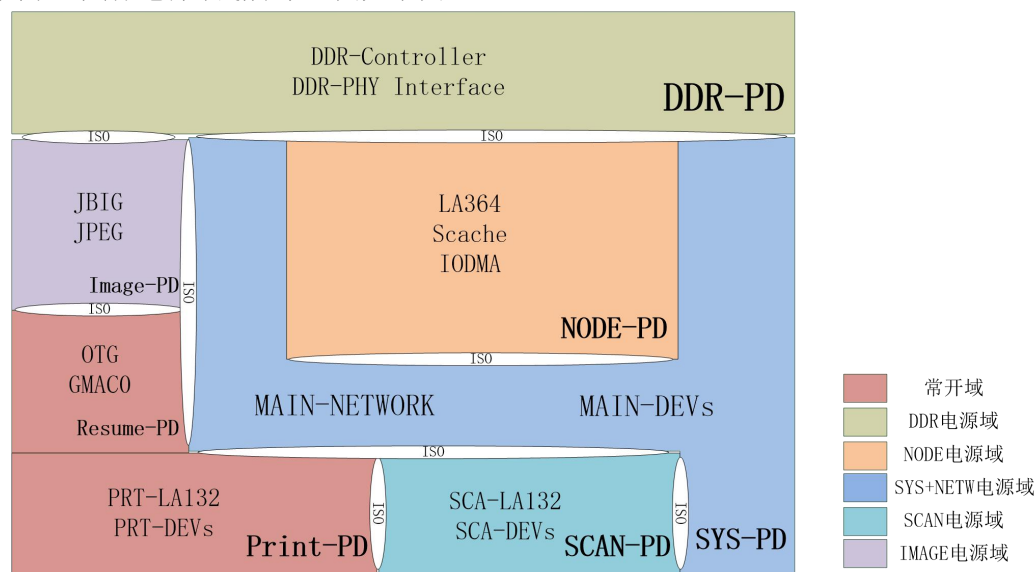


图25- 1 芯片电源域划分示意图

其中，具体电源域功能划分如下表：

表 25- 2 芯片各功能模块电源管理功能划分

功能模块	DFS	动态时钟门控	动态电源管理	PHY 低功耗	说明
NODE (LA364+SCACHE)	支持	支持	支持-	-	独立电源域，通过打印核 LA132 调节 NODE 时钟频率和电源关断
PRT-LA132	支持	支持	-	-	常开域，管理其他电源域电源关断
SCA-LA132	支持	支持	支持	-	独立电源域
DDR	-	支持	支持	支持	独立电源域，DDR 模块电源关断，PHY 接口自动进入低功耗模式
IMAGE	-	支持	支持	-	独立电源域
打印设备	-	支持	-	-	常开域
扫描设备	-	支持	支持	-	独立电源域
主互联	-	支持	支持	-	独立电源域
GMAC0	-	支持	-	-	常开域，支持 GMAC0 唤醒
GMAC1	-	支持	支持	-	独立电源域
OTG	-	支持	-	-	常开域，支持 OTG 唤醒
USB1/2	-	支持	支持	支持	独立电源域，USB 模块电源关断，PHY 接口自动进入低功耗模式
主系统设备	-	支持	支持	-	

### 25.2.1 DFS 功能描述

2P0500 芯片支持主系统处理器核(LA364+SCACHE)、打印处理器核(LA132)、扫描处理器核(LA132)动态频率调节(DFS)控制,由片内打印核 LA132 处理器核控制实现,控制结构框图如下图所示:

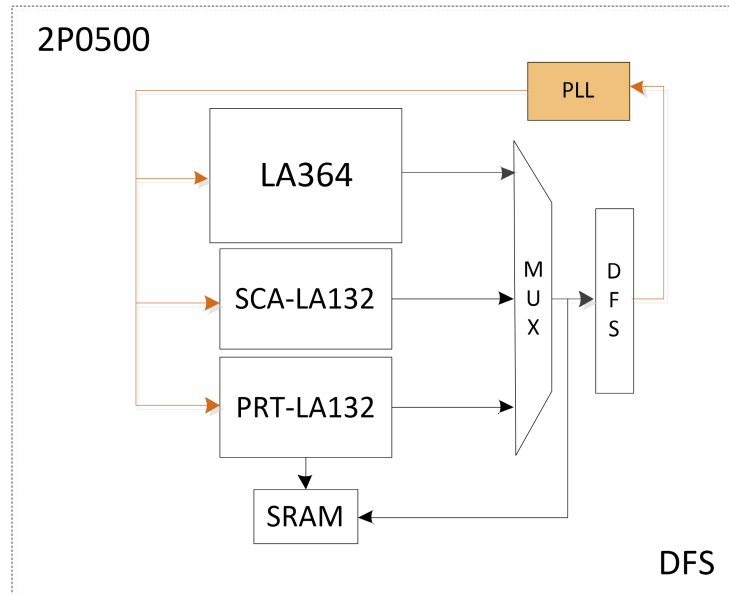


图25- 2 处理器核 DFS 控制结构框图

芯片内部各处理器核动态调频均由打印核 LA132 控制实现,芯片内部配有 128KB 大小 SRAM 供打印核 LA132 启动运行,启动入口地址可由打印启动入口配置寄存器(地址:0x15103100)配置选择。

2P0500 各处理器核动态调频控制具体实现:

(1) LA364 处理器核**频率调节**,主要通过打印核 LA132 配置 LA364-DFS 控制寄存器(LA364\_DFS\_CTRL)实现,打印核 LA132 通过访问 LA364\_DFS 控制寄存器,首先,配置 DFS 使能位有效,此时可通过配置 CLKBYP 位先将 LA364 处理器核时钟 BYPASS 为系统参考时钟(100MHz),再配置 DFS\_CLK\_ODIV、DFS\_FREQSCALE 分频参数,对应处理器核时钟分频参数,进行时钟分频配置,待选择合适分频参数后,解除 CLKBYP 配置,处理器核恢复正常时钟运行。

(2) LA132 处理器核**频率调节**,主要通过打印核 LA132 配置 LA32-DFS 控制寄存器(LA132\_DFS\_CTRL)实现,打印核 LA132 通过访问 LA132\_DFS 控制寄存器,首先,配置 DFS 使能位有效,此时可通过配置 CLKBYP 位先将各 LA132 处理器核时钟 BYPASS 为系统参考时钟(100MHz),再配置 DFS\_CLK\_ODIV、DFS\_FREQSCALE 分频参数,对应处理器核时钟分频参数,进行时钟分频配置,待选择合适分频参数后,解除 CLKBYP 配置,处理器核恢复正常时钟运行。

2P0500 芯片主处理器核(LA364+SCACHE)、打印核、扫描核动态频率调节(DFS)具体操作

流程图如下图所示：

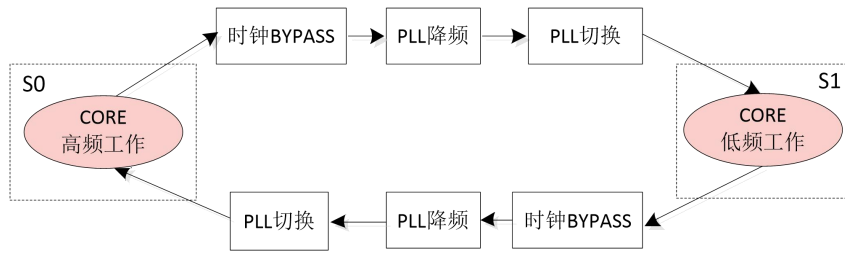


图25- 3 处理器核 DFS 操作流程

**软件操作：**

首先，2P0500 处理器核进入低压工作模式，首先需将处理器核进行降频处理，具体操作为：1) 使能 DFS，配置 dfs\_en 有效；2) 将处理器核时钟 bypass 为系统参考时钟，配置 dfs\_clkbyp 有效；3) 配置 dfs\_clkodiv 时钟输出分频参数(最高位需为 1)；4) 切换 PLL 时钟，配置 dfs\_clkbyp 无效；完成 2P0500 处理器核低频工作模式切换。

与此类似，2P0500 处理器核进入高频高性能工作模式，需对其工作频率进行升频处理，并进行相应时钟切换，完成 2P0500 处理器核高频模式切换。

**25. 2. 2 DPM 功能描述**

2P0500 芯片支持电源域模块动态电源调节 (DPM) 控制，通过 DPM 模块实现各个功能模块动态时钟门控、电源开关状态管理，其控制结构框图如下图所示：

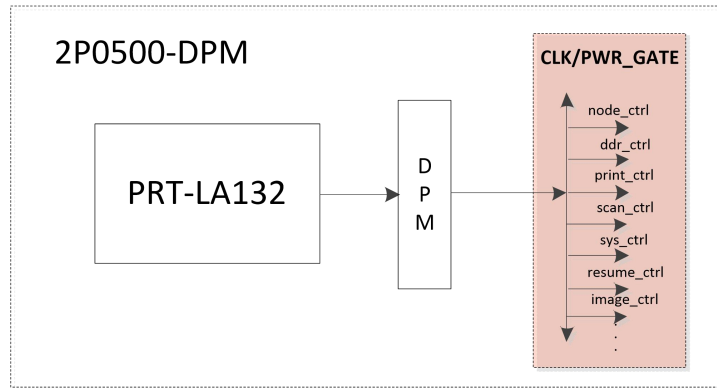


图25- 4 设备 DPM 控制结构框图

2P0500 芯片各功能模块动态电源调节 (DPM)，模块时钟、电源状态动态切换流程图如下图所示：



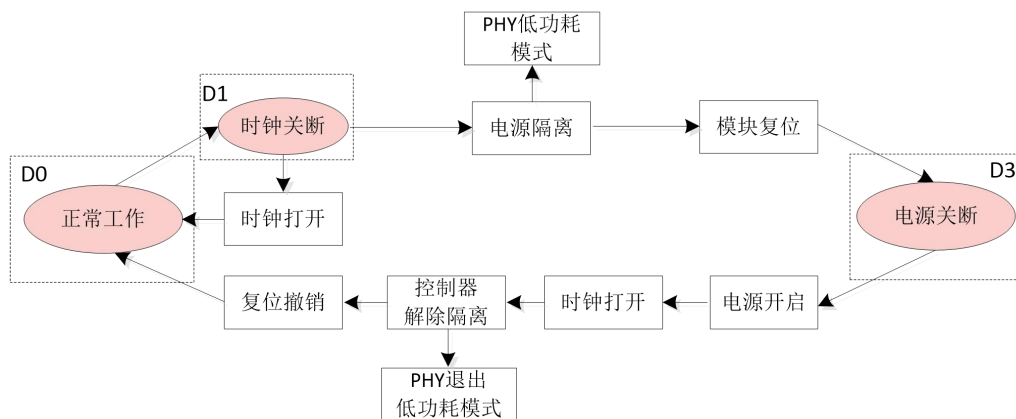


图25- 5 功能模块 DPM 状态切换流程图

其中，D0 状态为模块时钟、电源全开正常工作模式；D1 状态为模块时钟关闭、电源开启静态模式；D3 状态为模块时钟、电源全部关闭(若功能模块包括 PHY 接口，PHY 处于低功耗模式)。

**软件操作：**2P0500 芯片特定功能模块正常工作模式(D0 状态)下，首先，使能 DPM 模式调节，配置 DPM\_EN 相应使能位有效；然后，选择目标状态，若进行时钟关闭、电源关断操作直接选择 D1、D3 状态，配置 DPM\_TGT 对应模块目标状态值，完成模块时钟、电源关断切换，且可由 DPM\_STS 进行当前状态软件确认。

与此类似，若需解除当前关断状态，时钟关断状态 D1 下，配置 DPM\_TGT 目标状态值完成状态恢复，并由 DPM\_STS 进行软件确认；若解除电源关断状态，首先需配置 WAIT\_TIME 上电、复位等待时间参数，并确认 PWRUP\_SEL 应答模式，再配置 DPM\_TGT 目标状态值完成状态恢复，并由 DPM\_STS 进行软件确认。

### 25.2.3 设备唤醒功能描述

2P0500 芯片支持多种工作模式：正常工作、待机、休眠等模式，其中待机、休眠工作模式下均支持多种设备唤醒功能，包括 OTG 唤醒、网络唤醒、WIFI 唤醒及按键唤醒等。工作模式切换、各个设备唤醒操作均由打印系统统一控制维护。

其中，芯片待机唤醒操作控制流程，如下图所示：

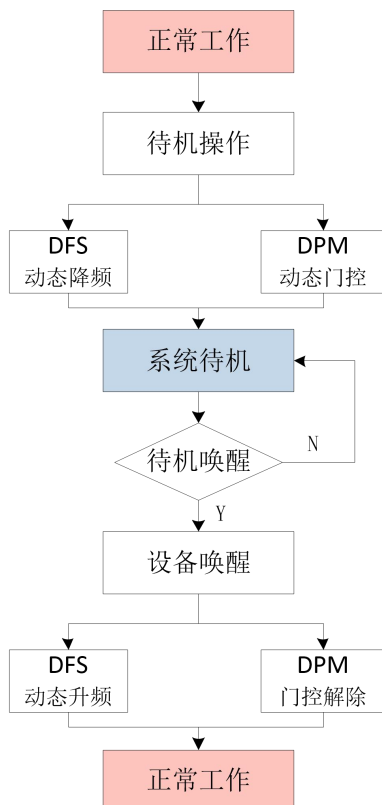


图25- 6 芯片待机唤醒操作流程

芯片正常工作状态下，在没有打印等任务需求时，可进行待机操作控制，该操作任务由打印系统统一完成，主要包括：DFS 动态降频，将芯片内各处理器核进行动态降频操作；DPM 动态时钟、电源门控，将芯片内各个电源域根据待机任务需求进行独立时钟门控、电源门控动态控制，待以上操作完成后芯片进入系统待机模式。系统待机模式下，若芯片设备需进行唤醒操作，主要包括 OTG 唤醒、网络唤醒、WIFI 及按键等唤醒，打印系统接收并维护各个设备唤醒事件，检测到以上设备有效唤醒事件后，将进行待机唤醒操作，主要包括：DFS 动态升频，将芯片内各处理器核进行动态升频恢复操作；DPM 动态时钟、电源门控，将芯片内各个电源域由待机状态下门控状态进行动态解除恢复，待以上操作完成后芯片进入正常工作模式，可响应打印等任务需求。

其中，芯片休眠唤醒操作控制流程，如下图所示：

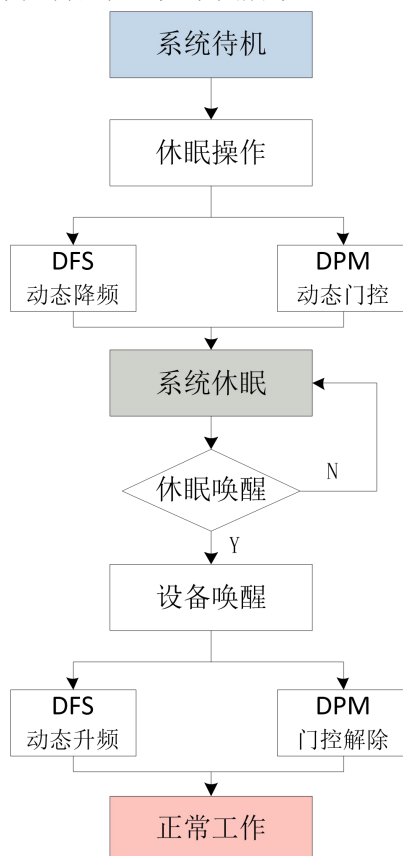


图25- 7 芯片休眠唤醒操作流程

芯片系统待机状态下，一段时间内没有打印等任务需求时，可进一步进行休眠操作控制，该操作任务仍由打印系统统一完成，主要包括：DFS 动态降频，将芯片内各处理器核进一步进行动态降频操作；DPM 动态时钟、电源门控，将芯片内各个电源域根据休眠任务需求进行独立时钟门控、电源门控动态控制，待以上操作完成后芯片进入系统休眠模式。系统休眠模式下，若芯片设备需进行唤醒操作，主要包括 OTG 唤醒、网络唤醒、WIFI 及按键等唤醒，打印系统接收并维护各个设备唤醒事件，检测到以上设备有效唤醒事件后，将进行休眠唤醒操作，主要包括：DFS 动态升频，将芯片内各处理器核进行动态升频恢复操作；DPM 动态时钟、电源门控，将芯片内各个电源域由休眠状态下门控状态进行动态解除恢复，待以上操作完成后芯片进入正常工作模式，可响应打印等任务需求。

## 25.3 寄存器描述

本节介绍电源管理控制器相关寄存器，使用方法可参见下一节描述。

动态电源管理(DPM/DFS)的物理基地址为：0x1510\_4000 — 0x1510\_4FFF (4KB)；

寄存器电压域表示寄存器的该位所属电压域。

寄存器属性简写包括：

R/W（可读可写），RO（只读），

R/WC（可读，写清除），WO（只写，读无效）

### 25.3.1 DPM 寄存器描述

DPM 配置寄存器基址为：0x1510\_4000。

#### 25.3.1.1 DPM\_EN : DPM Enable Register

表 25- 3 DPM\_EN 寄存器描述

地址偏移		电压域	属性
0x00		SOC	R/W
位域	描述		电压域
31:7	保留		-
6	<b>IMAGE_EN - R/W</b> IMAGE 图像模块动态电源管理(DPM)使能位,使能 IMAGE 图像模块动态电源管理, DPM 可对 IMAGE 图像模块进行动态时钟或电源关断控制。	SOC	
5	<b>RESUME_EN - R/W</b> RESUME 设备模块动态电源管理(DPM)使能位,使能 RESUME 设备模块动态电源管理, DPM 可对 RESUME 模块进行动态时钟或电源关断控制。	SOC	
4	<b>SYS_EN - R/W</b> 主系统设备模块动态电源管理(DPM)使能位,使能主系统设备模块动态电源管理, DPM 可对主系统设备模块进行动态时钟或电源关断控制。	SOC	
3	<b>SCAN_EN - R/W</b> SCAN 系统模块动态电源管理(DPM)使能位,使能 SCAN 系统模块动态电源管理, DPM 可对 SCAN 系统模块进行动态时钟或电源关断控制。	SOC	
2	<b>PRINT_EN - R/W</b> PRINT 系统模块动态电源管理(DPM)使能位,使能 PRINT 系统模块动态电源管理, DPM 可对 PRINT 系统模块进行动态时钟或电源关断控制。	SOC	
1	<b>DDR_EN - R/W</b> DDR 模块动态电源管理(DPM)使能位,使能 DDR 模块动态电源管理, DPM 可对 DDR 模块进行动态时钟或电源关断控制。	SOC	
0	<b>NODE_EN - R/W</b> NODE 模块动态电源管理(DPM)使能位,使能 NODE 模块动态电源管理, DPM 可对 NODE 模块进行动态时钟或电源关断控制。	SOC	

#### 25.3.1.2 PWRUP\_SEL : POWERUP Select Register

表 25- 4 PWRUP\_SEL 寄存器描述

地址偏移		电压域	属性
0x04		SOC	R/W
位域	描述		电压域
31:7	保留		
6	<b>IMAGE_SEL - R/W</b> IMAGE 图像模块上电完成应答选择: 1: 选择 IMAGE 图像模块本身上电完成标志; 0: 选择 image_pwrup_time 计数结束标志。	SOC	

5	<b>RESUME_SEL - R/W(不支持)</b> RESUME 模块上电完成应答选择： 1：选择 RESUME 模块本身上电完成标志；0：选择 resume_pwrup_time 计数结束标志。	SOC
4	<b>SYS_SEL - R/W</b> 主系统设备模块上电完成应答选择： 1：选择主系统设备模块本身上电完成标志；0：选择 sys_pwrup_time 计数结束标志。	SOC
3	<b>SCAN_SEL - R/W</b> SCAN 系统模块上电完成应答选择： 1：选择 SCAN 系统模块本身上电完成标志；0：选择 scan_pwrup_time 计数结束标志。	SOC
2	<b>PRINT_SEL - R/W(不支持)</b> PRINT 系统模块上电完成应答选择： 1：选择 PRINT 系统模块本身上电完成标志；0：选择 print_pwrup_time 计数结束标志。	SOC
1	<b>DDR_SEL - R/W</b> DDR 模块上电完成应答选择： 1：选择 DDR 模块本身上电完成标志；0：选择 ddr_pwrup_time 计数结束标志。	SOC
0	<b>NODE_SEL - R/W</b> NODE 模块上电完成应答选择： 1：选择 NODE 模块本身上电完成标志；0：选择 node_pwrup_time 计数结束标志。	SOC

### 25.3.1.3 DPM\_TGT : DPM Target Register

表 25- 5 DPM\_TGT 寄存器描述

地址偏移		电压域	属性
0x08		SOC	R/W
位域	描述		电压域
31:14	保留		
13:12	<b>IMAGE_TGT - R/W</b> IMAGE 图像模块动态电源目标状态选择： 00：全部开启工作状态；01：IMAGE 图像模块时钟关断状态； 10：保留；11：IMAGE 图像模块电源关断状态。		SOC
11:10	<b>RESUME_TGT - R/W</b> RESUME 模块动态电源目标状态选择： 00：全部开启工作状态；01：RESUME 模块时钟关断状态； 10：保留；11：RESUME 模块电源关断状态(不支持)。		SOC
9:8	<b>SYS_TGT - R/W</b> 主系统设备模块动态电源目标状态选择： 00：全部开启工作状态；01：主系统设备模块时钟关断状态； 10：保留；11：主系统设备模块电源关断状态。		SOC
7:6	<b>SCAN_TGT - R/W</b> SCAN 系统模块动态电源目标状态选择： 00：全部开启工作状态；01：SCAN 系统模块时钟关断状态； 10：保留；11：SCAN 系统模块电源关断状态。		SOC
5:4	<b>PRINT_TGT - R/W</b> PRINT 系统模块动态电源目标状态选择： 00：全部开启工作状态；01：PRINT 系统模块时钟关断状态； 10：保留；11：PRINT 系统模块电源关断状态(不支持)。		SOC
3:2	<b>DDR_TGT - R/W</b> DDR 模块动态电源目标状态选择： 00：全部开启工作状态；01：DDR 模块时钟关断状态； 10：保留；11：DDR 模块电源关断状态。		SOC
1:0	<b>NODE_TGT - R/W</b> NODE 模块动态电源目标状态选择： 00：全部开启工作状态；01：NODE 模块时钟关断状态； 10：保留；11：NODE 模块电源关断状态。		SOC

### 25.3.1.4 DPM\_STS : DPM Status Register

表 25- 6 DPM\_STS 寄存器描述

地址偏移		电压域	属性
0x0c		SOC	RO
位域	描述		电压域
31:14	保留		
13:12	<b>IMAGE_STS - RO</b> IMAGE 模块动态电源当前状态： 00: 全部开启状态；01: IMAGE 模块时钟关断状态； 10: 保留；11: IMAGE 模块电源关断状态。		SOC
11:10	<b>RESUME_STS - RO</b> RESUME 模块动态电源当前状态： 00: 全部开启状态；01: RESUME 模块时钟关断状态； 10: 保留；11: RESUME 模块电源关断状态(不支持)。		SOC
9:8	<b>SYS_STS - RO</b> 主系统设备模块动态电源当前状态： 00: 全部开启状态；01: 主系统设备模块时钟关断状态； 10: 保留；11: 主系统设备模块电源关断状态。		SOC
7:6	<b>SCAN_STS - RO</b> SCAN 系统模块动态电源当前状态： 00: 全部开启状态；01: SCAN 系统模块时钟关断状态； 10: 保留；11: SCAN 系统模块电源关断状态。		SOC
5:4	<b>PRINT_STS - RO</b> PRINT 系统模块动态电源当前状态： 00: 全部开启状态；01: PRINT 系统模块时钟关断状态； 10: 保留；11: PRINT 系统模块电源关断状态(不支持)。		SOC
3:2	<b>DDR_STS - RO</b> DDR 模块动态电源当前状态： 00: 全部开启状态；01: 模块时钟关断状态； 10: 保留；11: 模块电源关断状态。		SOC
1:0	<b>NODE_STS - RO</b> NODE 模块动态电源当前状态： 00: 全部开启状态；01: 模块时钟关断状态； 10: 保留；11: 模块电源关断状态。		SOC

### 25.3.1.5 WAIT\_TIME0 : Wait Time0 Register

表 25- 7 WAIT\_TIME0 寄存器描述

地址偏移		电压域	属性
0x10		SOC	R/W
位域	描述		电压域
31:24	<b>DDR_PWRUP_TIME - R/W</b> DDR 模块上电完成等待计数值：0~255。		SOC
23:16	<b>DDR_RST_WTIME - R/W</b> DDR 模块复位完成等待计数值：0~255。		SOC
15:8	<b>NODE_PWRUP_TIME - R/W</b> NODE 模块上电完成等待计数值：0~255。		SOC
7:0	<b>NODE_RST_WTIME - R/W</b> NODE 模块复位完成等待计数值：0~255。		SOC

### 25.3.1.6 WAIT\_TIME1 : Wait Time1 Register

表 25- 8 WAIT\_TIME1 寄存器描述

地址偏移		电压域	属性
0x14		SOC	R/W
位域	描述		电压域
31:24	<b>SCAN_PWRUP_TIME - R/W</b> SCAN 系统模块上电完成等待计数值：0~255。		SOC
23:16	<b>SCAN_RST_WTIME - R/W</b> SCAN 系统模块复位完成等待计数值：0~255。		SOC
15:8	<b>PRINT_PWRUP_TIME - R/W(不支持)</b> PRINT 系统模块上电完成等待计数值：0~255。		SOC
7:0	<b>PRT_RST_WTIME - R/W(不支持)</b> PRINT 系统模块复位完成等待计数值：0~255。		SOC

### 25.3.1.7 WAIT\_TIME2 : Wait Time2 Register

表 25- 9 WAIT\_TIME2 寄存器描述

地址偏移		电压域	属性
0x18		SOC	R/W
位域	描述		电压域
31:24	<b>RESUME_PWRUP_TIME - R/W(不支持)</b> RESUME 系统模块上电完成等待计数值：0~255。		SOC
23:16	<b>RESUME_RST_WTIME - R/W(不支持)</b> RESUME 系统模块复位完成等待计数值：0~255。		SOC
15:8	<b>SYS_PWRUP_TIME - R/W</b> 主系统设备模块上电完成等待计数值：0~255。		SOC
7:0	<b>SYS_RST_WTIME - R/W</b> 主系统设备模块复位完成等待计数值：0~255。		SOC

### 25.3.1.8 WAIT\_TIME3 : Wait Time3 Register

表 25- 10 WAIT\_TIME3 寄存器描述

地址偏移		电压域	属性
0x1c		SOC	R/W
位域	描述		电压域
31:16	保留		
15:8	<b>IMAGE_PWRUP_TIME - R/W</b> IMAGE 图像设备模块上电完成等待计数值：0~255。		SOC
7:0	<b>IMAGE_RST_WTIME - R/W</b> IMAGE 图像设备模块复位完成等待计数值：0~255。		SOC

### 25.3.1.9 CG\_CFG : Clkgate Config Register

表 25- 11 CG\_CFG 寄存器描述

地址偏移		电压域	属性
0x20		SOC	R/W
位域	描述		电压域
31:23	保留		-
22	<b>DDR.CG_EN - R/W</b> DDR 模块设备时钟关断控制使能位(DDR 域 DPM 动态时钟门控), 高电平使能时钟关断控制, 默认值为 0: 0: 屏蔽时钟关断控制; 1: 使能时钟关断控制。		SOC
21	<b>DDR_RST_EN - R/W</b> DDR 模块设备复位控制使能位(DDR 域 DPM 动态电源门控), 高电平使能复位控制,		SOC



	默认值为 0: 0: 屏蔽复位控制; 1: 使能复位控制。	
20	<b>DDR_ISO_EN - R/W</b> DDR 模块设备隔离控制使能位(DDR 域 DPM 动态电源门控), 高电平使能隔离关断控制, 默认值为 0: 0: 屏蔽隔离关断控制; 1: 使能隔离关断控制。	SOC
19	<b>JPEG_CG_EN - R/W</b> JPEG 模块设备时钟关断控制使能位(IMAGE 域 DPM 动态时钟门控), 高电平使能时钟关断控制, 默认值为 0: 0: 屏蔽时钟关断控制; 1: 使能时钟关断控制。	SOC
18	<b>JBIG_CG_EN - R/W</b> JBIG 模块设备时钟关断控制使能位(IMAGE 域 DPM 动态时钟门控), 高电平使能时钟关断控制, 默认值为 0: 0: 屏蔽时钟关断控制; 1: 使能时钟关断控制。	SOC
17	<b>OTG_CG_EN - R/W</b> OTG 模块设备时钟关断控制使能位(RESUME 域 DPM 动态时钟门控), 高电平使能时钟关断控制, 默认值为 0: 0: 屏蔽时钟关断控制; 1: 使能时钟关断控制。	SOC
16	<b>GMACO_CG_EN - R/W</b> GMACO 模块设备时钟关断控制使能位(RESUME 域 DPM 动态时钟门控), 高电平使能时钟关断控制, 默认值为 0: 0: 屏蔽时钟关断控制; 1: 使能时钟关断控制。	SOC
15:13	保留	-
12	<b>APBDEVS_CG_EN - R/W</b> 主系统 APB 模块设备时钟关断控制使能位(SYS 域 DPM 动态时钟门控), 高电平使能时钟关断控制, 默认值为 0: 0: 屏蔽时钟关断控制; 1: 使能时钟关断控制。	SOC
11	<b>SDIO_CG_EN - R/W</b> 主系统 SDIO0~1 模块设备时钟关断控制使能位(SYS 域 DPM 动态时钟门控), 高电平使能时钟关断控制, 默认值为 0: 0: 屏蔽时钟关断控制; 1: 使能时钟关断控制。	SOC
10	<b>USB_CG_EN - R/W</b> 主系统 USB1~2 模块设备时钟关断控制使能位(SYS 域 DPM 动态时钟门控), 高电平使能时钟关断控制, 默认值为 0: 0: 屏蔽时钟关断控制; 1: 使能时钟关断控制。	SOC
9	<b>GMAC1_CG_EN - R/W</b> 主系统 GMAC1 模块设备时钟关断控制使能位(SYS 域 DPM 动态时钟门控), 高电平使能时钟关断控制, 默认值为 0: 0: 屏蔽时钟关断控制; 1: 使能时钟关断控制。	SOC
8	<b>BOOT_CG_EN - R/W</b> 主系统 BOOT 模块设备时钟关断控制使能位(SYS 域 DPM 动态时钟门控), 高电平使能时钟关断控制, 默认值为 0: 0: 屏蔽时钟关断控制; 1: 使能时钟关断控制。	SOC
7:6	保留	-
5	<b>SCADEVS_CG_EN - R/W</b> 扫描系统设备时钟关断控制使能位(SCAN 域 DPM 动态时钟门控), 高电平使能时钟关断控制, 默认值为 0: 0: 屏蔽时钟关断控制; 1: 使能时钟关断控制。	SOC
4	<b>SCALA132_CG_EN - R/W</b> 扫描核 LA132 时钟关断控制使能位(SCAN 域 DPM 动态时钟门控), 高电平使能时钟关断控制, 默认值为 0: 0: 屏蔽时钟关断控制; 1: 使能时钟关断控制。	SOC
3	<b>PRTXBAR_CG_EN - R/W</b> 打印系统互联时钟关断控制使能位(PRINT 域 DPM 动态时钟门控), 高电平使能时钟关断控制, 默认值为 0: 0: 屏蔽时钟关断控制; 1: 使能时钟关断控制。	SOC
2	<b>PRTLAI32_CG_EN - R/W</b> 打印核 LA132 时钟关断控制使能位(PRINT 域 DPM 动态时钟门控), 高电平使能时钟关断控制, 默认值为 0:	SOC



	0: 屏蔽时钟关断控制; 1: 使能时钟关断控制。	
1	<b>IODMA_CG_EN - R/W</b> IODMA 时钟关断控制使能位 (NODE 域 DPM 动态时钟门控), 高电平使能时钟关断控制, 默认值为 0: 0: 屏蔽时钟关断控制; 1: 使能时钟关断控制。	SOC
0	<b>LA364_CG_EN - R/W</b> LA364 时钟关断控制使能位 (NODE 域 DPM 动态时钟门控), 高电平使能时钟关断控制, 默认值为 0: 0: 屏蔽时钟关断控制; 1: 使能时钟关断控制。	SOC

### 25.3.2 DFS 寄存器描述

DFS 配置寄存器基址为: 0x1510\_4000。

#### 25.3.2.1 NODE\_DVFS\_CTRL : NODE DVFS Control Register

表 25- 12 NODE DVFS Control 寄存器描述

地址偏移	电压域	属性
0x30	SOC	R/W
位域	描述	电压域
31:20	保留	
19:15	<b>DVFS_FREQSCALE - R/W</b> NODE 模块 PLL 时钟输出 Freqscale 分频参数配置: 对应 PLL 时钟分频配置参数 freqscale。	SOC
14:8	<b>DVFS_CLK_ODIV - R/W</b> NODE 模块 PLL 时钟输出分频参数配置, 最高位(odiv[14])为软件参数配置使能位(高有效): 对应 PLL 时钟配置参数 ODIV[5:0]: 0~63	SOC
7:3	保留	-
2	<b>DVFS_CLK_OFF - R/W</b> NODE 模块时钟关断配置位, 高电平时钟关断, 默认值为 0。	SOC
1	<b>DVFS_CLKBYP - R/W</b> NODE 模块时钟 BYPASS 配置位, 默认值为 0: 1: NODE 时钟 BYPASS 为系统参考时钟; 0: NODE 时钟为 PLL 输出时钟。	SOC
0	<b>DVFS_EN - R/W</b> NODE 模块动态调频调压(DFS)使能位, 高电平有效, 默认值为 0。	SOC

#### 25.3.2.2 LA132\_DVFS\_CTRL : LA132 DVFS Control Register

表 25- 13 LA132 DVFS Control 寄存器描述

地址偏移	电压域	属性
0x34	SOC	R/W
位域	描述	电压域
31:24	保留	-
23:20	<b>SCALA132_DVFS_FREQSCALE - R/W</b> LA132(扫描核)模块 PLL 时钟输出 Freqscale 分频参数配置: 对应 PLL 时钟分频配置参数 freqscale。	SOC
19:16	<b>PRTL132_DVFS_FREQSCALE - R/W</b> LA132(打印核)模块 PLL 时钟输出 Freqscale 分频参数配置: 对应 PLL 时钟分频配置参数 freqscale。	SOC
15	保留	-
14:8	<b>LA132_DVFS_CLK_ODIV - R/W</b> LA132(打印、扫描核)模块 PLL 时钟输出分频参数配置, 最高位(odiv[14])为软件参数配置使能位(高有效): 对应 PLL 时钟配置参数 ODIV[5:0]: 0~63	SOC

7	<b>SCALA132_RSTN_EN - R/W</b> LA132(扫描核)复位门控关断配置位，默认值为0： 0-复位有效，1-复位解除。	SOC
6	<b>SCALA132_CLK_EN - R/W</b> LA132(扫描核)时钟门控关断配置位，默认值为0： 0-时钟关闭，1-时钟开启。	SOC
5	<b>PRTL132_RSTN_EN - R/W</b> LA132(打印核)复位门控关断配置位，默认值为0： 0-复位有效，1-复位解除。	SOC
4	<b>PRTL132_CLK_EN - R/W</b> LA132(打印核)时钟门控关断配置位，默认值为0： 0-时钟关闭，1-时钟开启。	SOC
3	保留	-
2	<b>LA132_DVFS_CLK_OFF - R/W</b> LA132(打印、扫描核)时钟关断配置位，高电平时钟关断，默认值为0。	SOC
1	<b>LA132_DVFS_CLKBYP - R/W</b> LA132(打印、扫描核)时钟 BYPASS 配置位，默认值为0： 1: LA132 时钟 BYPASS 为系统参考时钟；0: LA132 时钟为 PLL 输出时钟。	SOC
0	<b>LA132_DVFS_EN - R/W</b> LA132(打印、扫描核)动态调频调压(DFS)使能位，高电平有效，默认值为0。	SOC

### 25.3.3 WDT 寄存器描述

芯片内部看门狗(WDT)共三组，分别置于主系统、打印、扫描系统中，每组看门狗功能均一致，各个看门狗寄存器基址如下表：

表 25- 14 WDT 寄存器地址

各系统 WDT 基址	设备	备注
0x1420_5000	MAIN_WDT	每组 WDT 独立控制
0x1510_3900	PRT_WDT	每组 WDT 独立控制
0x1520_2900	SCA_WDT	每组 WDT 独立控制

#### 25.3.3.1 WD\_EN : Watch Dog Enable Register

表 25- 15 WD\_EN 寄存器描述

地址偏移	电压域	属性
0x00	SOC	R/W
位域	描述	
31:1	保留	
1	<b>WD_EN - R/W</b> Watch dog 功能使能，高电平有效	
0	<b>OS_RST - R/W</b> 系统软复位，该位软件写‘1’使系统复位	

#### 25.3.3.2 WD\_SET : Watch Dog Set Register

表 25- 16 WD\_SET 寄存器描述

地址偏移	电压域	属性
0x04	SOC	WO
位域	描述	
31:1	保留	
0	写该位将重填 watch dog 计数器，配置该位需首先配置 WDT 使能位。	

### 25.3.3.3 WD\_Timer : Watch Dog Timer Register

表 25- 17 WD\_TIMER 寄存器描述

地址偏移	电压域	属性
0x08	SOC	R/W
位域	描述	
31:0	该寄存器的值为 watch dog 重填的值，复位后为全 1。	

### 25.3.3.4 PLTRST\_SET : Plantrst Set Register

表 25- 18 PLTRST\_SET 寄存器描述

地址偏移	电压域	属性
0x0c	SOC	R/W
位域	描述	
31:1	保留	
0	Platform Reset 信号输出选择位： 0: 选择将内部看门狗、系统软复位由 platform reset 引脚同步输出； 1: 内部看门狗、系统软复位不输出，仅内部复位有效（默认配置，复位引脚不输出）。	

## 26 RTC

### 26.1 概述

实时时钟（RTC）单元可以在主板上电后进行配置，当主板断电后，该单元仍然运作，可以仅靠板上的电池供电就正常运行。RTC 单元运行时电流仅几个微安。

RTC 包含振荡器，结合外部 32.768KHZ 晶体产生工作时钟。该时钟用于时间信息的维护以及产生各种定时和计数中断。

RTC 模块中包含两个计数器，分别为 TOY（Time of Year）计数器和 RTC 计数器。其中 TOY 计数器按年月日时分秒计数，精度为以 0.1 秒；RTC 计数器以 32.768KHz 时钟计数，宽度为 32 位。

### 26.2 寄存器描述

RTC 模块寄存器位于 0x1510a000——0x1510afff 的 4KB 地址空间内，其基地址为 0x1510a000，所有寄存器位宽均为 32 位。

#### 26.2.1 寄存器地址列表

表 26- 1 RTC 寄存器列表

名称	偏移地址	位宽	RW	描述
sys_toytrim	0x20	32	RW	软件必须初始化为 0
sys_toywrite0	0x24	32	W	TOY 低 32 位数值写入
sys_toywritel	0x28	32	W	TOY 高 32 位数值写入
sys_toyread0	0x2C	32	R	TOY 低 32 位数值读出
sys_toyreadl	0x30	32	R	TOY 高 32 位数值读出
sys_toymatch0	0x34	32	RW	TOY 定时中断 0
sys_toymatch1	0x38	32	RW	TOY 定时中断 1
sys_toymatch2	0x3C	32	RW	TOY 定时中断 2
sys_rtctrl	0x40	32	RW	TOY 和 RTC 控制寄存器 软件必须初始化
sys_rtctrim	0x60	32	RW	软件必须初始化为 0
sys_rtcwrite0	0x64	32	W	RTC 定时计数写入
sys_rtcread0	0x68	32	R	RTC 定时计数读出
sys_rtcmatch0	0x6C	32	RW	RTC 时钟定时中断 0
sys_rtcmatch1	0x70	32	RW	RTC 时钟定时中断 1
sys_rtcmatch2	0x74	32	RW	RTC 时钟定时中断 2

#### 26.2.2 SYS\_TOYWRITE0

中文名： TOY 计数器低 32 位数值

寄存器位宽： [31： 0]

偏移量： 0x24

复位值： 0x00000000

表 26- 2 TOY 计数器低 32 位写入寄存器

位域	位域名称	访问	缺省	描述
31:26	TOY_MONTH	W	-	月, 范围 1~12
25:21	TOY_DAY	W	-	日, 范围 1~31
20:16	TOY_HOUR	W	-	小时, 范围 0~23
15:10	TOY_MIN	W	-	分, 范围 0~59
9:4	TOY_SEC	W	-	秒, 范围 0~59
3:0	TOY_MILLISEC	W	-	0.1 秒, 范围 0~9

### 26.2.3 SYS\_TOYWRITE1

中文名: TOY 计数器高 32 位数值

寄存器位宽: [31: 0]

偏移量: 0x28

复位值: 0x00000000

表 26- 3 TOY 计数器高 32 位写入寄存器

位域	位域名称	访问	缺省	描述
31:0	TOY_YEAR	W	-	年, 范围 0~16383

### 26.2.4 SYS\_TOYREAD0

中文名: TOY 计数器低 32 位数值

寄存器位宽: [31: 0]

偏移量: 0x2C

复位值: 0x00000000

表 26- 4 TOY 计数器低 32 位读出寄存器

位域	位域名称	访问	缺省	描述
31:26	TOY_MONTH	R	0	月, 范围 1~12
25:21	TOY_DAY	R	0	日, 范围 1~31
20:16	TOY_HOUR	R	0	小时, 范围 0~23
15:10	TOY_MIN	R	0	分, 范围 0~59
9:4	TOY_SEC	R	0	秒, 范围 0~59
3:0	TOY_MILLISEC	R	0	0.1 秒, 范围 0~9

### 26.2.5 SYS\_TOYREAD1

中文名: TOY 计数器高 32 位数值

寄存器位宽: [31: 0]

偏移量: 0x30

复位值: 0x00000000

表 26- 5 TOY 计数器高 32 位读出寄存器

位域	位域名称	访问	缺省	描述
31:0	TOY_YEAR	R	0	年, 范围 0~16383

## 26.2.6 SYS\_TOYMATCH0/1/2

中文名: TOY 计数器中断寄存器 0/1/2

寄存器位宽: [31: 0]

偏移量: 0x34/38/3C

复位值: 0x00000000

表 26- 6 TOY 计数器中断寄存器 0/1/2

位域	位域名称	访问	缺省	描述
31:26	YEAR	RW	0	年, 范围 0~16383
25:22	MONTH	RW	0	月, 范围 1~12
21:17	DAY	RW	0	日, 范围 1~31
16:12	HOUR	RW	0	小时, 范围 0~23
11:6	MIN	RW	0	分, 范围 0~59
5:0	SEC	RW	0	秒, 范围 0~59

## 26.2.7 SYS\_RTCCTRL

中文名: RTC 定时器中断寄存器 0/1/2

寄存器位宽: [31: 0]

偏移量: 0x40

复位值: 无

表 26- 7 RTC 定时器中断寄存器 0/1/2

位域	位域名称	访问	缺省	描述
31:24	保留	R	0	保留, 置 0
23	ERS	R	0	REN(bit13)写状态
22:21	保留	R	0	保留, 置 0
20	RTS	R	0	Sys_rtctrim 写状态
19	RM2	R	0	Sys_rtcmatch2 写状态
18	RM2	R	0	Sys_rtcmatch2 写状态
17	RM0	R	0	Sys_rtcmatch0 写状态
16	RS	R	0	Sys_rtcwrite 写状态
15	保留	R	0	保留, 置 0
14	保留	R	0	保留, 置 0
13	REN	R/W	0	RTC 使能, 高有效。需要初始化为 1
12	保留	R	0	保留, 置 0
11	TEN	R/W	0	TOY 使能, 高有效。需要初始化为 1
10	保留	R	0	保留, 置 0
9	保留	R	0	保留, 置 0

位域	位域名称	访问	缺省	描述
8	EO	R/W	0	0: 32.768k 晶振禁止; 1: 32.768k 晶振使能
7	保留	R	0	保留, 置 0
6	保留	R	0	保留, 置 0
5	32S	R	0	0: 32.768k 晶振不工作; 1: 32.768k 晶振正常工作。
4	保留	R	0	保留, 置 0
3	TM2	R	0	Sys_toymatch2 写状态
2	TM1	R	0	Sys_toymatch1 写状态
1	TM0	R	0	Sys_toymatch0 写状态
0	TS	R	0	Sys_toywrite 写状态

### 26.2.8 SYS\_RTCWRITE

中文名: RTC 计数器写入端口

寄存器位宽: [31: 0]

偏移量: 0x64

复位值: 0x00000000

表 26- 8 RTC 计数器写入寄存器

位域	位域名称	访问	缺省	描述
31:0	RTCWRITE	W	-	RTC 计数器写入寄存器

### 26.2.9 SYS\_RTCREAD

中文名: RTC 计数器读出端口

寄存器位宽: [31: 0]

偏移量: 0x68

复位值: 0x00000000

表 26- 9 RTC 计数器读出寄存器

位域	位域名称	访问	缺省	描述
31:0	RTCREAD	R	0	RTC 计数器读出寄存器

### 26.2.10 SYS\_RTCMATCH0/1/2

中文名: RTC 定时器中断寄存器 0/1/2

寄存器位宽: [31: 0]

偏移量: 0x6C/70/74

复位值: 0x00000000

表 26- 10 RTC 定时器中断寄存器 0/1/2

位域	位域名称	访问	缺省	描述
31:26	RTCMATCH0/1/2	RW	0	RTC 定时比较寄存器 0/1/2

# 27 GPIO

## 27.1 概述

芯片共有 139 个 GPIO 引脚，全部与其他功能引脚复用。其中，主系统 GPIO 引脚资源共 44 个，打印系统 GPIO 引脚资源共 58 个，扫描系统 GPIO 引脚资源共 37 个。

## 27.2 寄存器描述

主系统、打印系统、扫描系统下 GPIO 寄存器基地址分别为：

表 27- 1 GPIO 寄存器地址

各系统基地址	设备	备注
0x1420_2000	MAIN_GPIO0~43	每个 GPIO 引脚均由一组寄存器控制
0x1510_2000	PRT_GPIO0~57	每个 GPIO 引脚均由一组寄存器控制
0x1520_1000	SCA_GPIO0~36	每个 GPIO 引脚均由一组寄存器控制

每个系统下，GPIO 引脚均由一组寄存器控制，包括：GPIO 方向控制（GPIO\_OEN）、GPIO 输出值（GPIO\_O）、GPIO 输入值（GPIO\_I）、GPIO 输入中断使能控制（GPIO\_INT\_EN）、GPIO 输入中断极性控制（GPIO\_INT\_POL）、GPIO 输入中断边沿性控制（GPIO\_INT\_EDGE）、GPIO 输入中断双沿控制（GPIO\_INT\_DUAL）、GPIO 输入中断清除（GPIO\_INT\_CLR）、GPIO 输入中断状态（GPIO\_INT\_STS）。

表 27- 2 GPIO 控制寄存器

寄存器	大小（按位）	描述															
GPIO_OEN	1	GPIO 输出使能，低有效。															
GPIO_O	1	GPIO 输出值。															
GPIO_I	1	GPIO 输入值。															
GPIO_INT_EN	1	GPIO 中断使能。															
GPIO_INT_POL	1	GPIO 中断极性。															
GPIO_INT_EDGE	1	GPIO 中断边沿性。与 GPIO 中断极性配合控制 GPIO 中断状态的产生。															
		<table border="1"> <thead> <tr> <th>POL</th> <th>EDGE</th> <th>描述</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>低电平触发中断</td> </tr> <tr> <td>1</td> <td>0</td> <td>高电平触发中断</td> </tr> <tr> <td>0</td> <td>1</td> <td>下降沿触发中断</td> </tr> <tr> <td>1</td> <td>1</td> <td>上升沿触发中断</td> </tr> </tbody> </table>	POL	EDGE	描述	0	0	低电平触发中断	1	0	高电平触发中断	0	1	下降沿触发中断	1	1	上升沿触发中断
		POL	EDGE	描述													
		0	0	低电平触发中断													
		1	0	高电平触发中断													
0	1	下降沿触发中断															
1	1	上升沿触发中断															
GPIO_INT_CLR	1	GPIO 中断状态清除															
GPIO_INT_STS	1	GPIO 中断状态															
GPIO_INT_DUAL	1	GPIO 中断双沿控制，仅在边沿中断模式下有效															



## 27.3 访问地址

GPIO 的寄存器访问地址等于各系统下的基地址加寄存器偏移，各寄存器访问地址均需以字节形式访问。

芯片提供了两种方式来控制 GPIO 引脚：一种是按位控制每个 GPIO 引脚，一种是按字节控制每个 GPIO 引脚。芯片通过提供两个地址空间来映射 GPIO 控制寄存器实现该功能。一种是按位映射，一种是按字节来索引控制寄存器的每个比特位。对应的 GPIO 内部的地址空间也分为两部分。以上两种地址读写均需按照字节形式进行访问，推荐使用后一种方式来控制器 GPIO 引脚。

GPIO 模块的内部寄存器物理地址构成如下：

表 27- 3 GPIO 模块内部寄存器物理地址

地址空间	说明
0x800-0xF80	按字节控制寄存器地址（需按字节形式访问，以字节为单位读写）
0x0 - 0x80	按位控制寄存器地址（需按字节形式访问，按位读写）

表 27- 4 按位控制 GPIO 配置寄存器地址

地址偏移	寄存器	大小（位）	描述
0x00	GPIO_OEN	各系统区分	GPIO 输出使能，低有效。每位控制一个 GP IO 引脚。
0x10	GPIO_0	各系统区分	GPIO 输出值。每位控制一个 GPIO 引脚。
0x20	GPIO_I	各系统区分	GPIO 输入值。每位控制一个 GPIO 引脚。
0x30	GPIO_INT_EN	各系统区分	GPIO 中断使能。每位控制一个 GPIO 引脚。
0x40	GPIO_INT_POL	各系统区分	GPIO 中断极性。每位控制一个 GPIO 引脚。
0x50	GPIO_INT_EDGE	各系统区分	GPIO 中断边沿性。每位控制一个 GPIO 引脚。
0x60	GPIO_INT_CLR	各系统区分	GPIO 中断清除。每位控制一个 GPIO 引脚。
0x70	GPIO_INT_STS	各系统区分	GPIO 中断状态。每位控制一个 GPIO 引脚。
0x80	GPIO_INT_DUAL	各系统区分	GPIO 中断双沿模式。每位控制一个 GPIO 引脚。

表 27- 5 按字节控制 GPIO 配置寄存器地址

地址偏移	寄存器	大小（字节）	描述
0x800	GPIO_OEN	各系统区分	GPIO 输出使能，低有效。每个字节控制一个 GPIO 引脚。
0x900	GPIO_0	各系统区分	GPIO 输出值。每个字节控制一个 GPIO 引脚。
0xA00	GPIO_I	各系统区分	GPIO 输入值。每个字节控制一个 GPIO 引脚。
0xB00	GPIO_INT_EN	各系统区分	GPIO 中断使能。每个字节控制一个 GPIO 引脚。
0xC00	GPIO_INT_POL	各系统区分	GPIO 中断极性。每个字节控制一个 GPIO 引脚。
0xD00	GPIO_INT_EDGE	各系统区分	GPIO 中断边沿性。每个字节控制一个 GPIO 引脚。
0xE00	GPIO_INT_CLR	各系统区分	GPIO 中断清除。每个字节控制一个 GPIO 引脚。
0xF00	GPIO_INT_STS	各系统区分	GPIO 中断状态。每个字节控制一个 GPIO 引脚。
0xF80	GPIO_INT_DUAL	各系统区分	GPIO 中断双沿模式。每个字节控制一个 GPIO 引脚。

## 27.4 控制寄存器

### 27.4.1 GPIO 方向控制

地址偏移：00-07h 属性：R/W

默认值：FFFFFFFh 大小：8

表 27- 6 GPIO 方向按位控制寄存器

位域	名称	访问	描述
63:0	GPIO_OEN	R/W	每位对应于 GPIO[63:0] (依各系统 GPIO 资源) 的方向控制。 0: 输出 1: 输入

地址偏移：800-83fh 属性：R/W

默认值：FFFFFFFh 大小：64

表 27- 7 GPIO 方向按字节控制寄存器

位域	名称	访问	描述
7:0	GPIO_OEN	R/W	每字节对应于每一位 GPIO[63:0] (依各系统 GPIO 资源) 的方向控制。 0: 输出 1: 输入

### 27.4.2 GPIO 输出值

地址偏移：10-17h 属性：R/W

默认值：00000000h 大小：8

表 27- 8 GPIO 按位输出寄存器

位域	名称	访问	描述
63:0	GPIO_O	R/W	每位对应于 GPIO[63:0] (依各系统 GPIO 资源) 的输出值。

地址偏移：900-93fh 属性：R/W

默认值：00000000h 大小：64

表 27- 9 GPIO 按字节输出寄存器

位域	名称	访问	描述
7:0	GPIO_O	R/W	每字节对应于每一位 GPIO[63:0] (依各系统 GPIO 资源) 的输出值。

### 27.4.3 GPIO 输入值

地址偏移：20-27h 属性：RO

默认值：N/A 大小：8

表 27- 10 GPIO 按位输入寄存器

位域	名称	访问	描述
63:0	GPIO_I	RO	每位对应于 GPIO[63:0] (依各系统 GPIO 资源) 的输入值。

地址偏移：a00-a3fh                      属性：RO  
默认值：00000000h                      大小：64

表 27- 11 GPIO 按字节输入寄存器

位域	名称	访问	描述
7:0	GPIO_I	RO	每字节对应于每一位 GPIO[63:0] (依各系统 GPIO 资源)的输入值。

#### 27.4.4 GPIO 中断使能

地址偏移：30-37h                      属性：R/W  
默认值：00000000h                      大小：8

表 27- 12 GPIO 按位中断使能寄存器

位域	名称	访问	描述
63:0	GPIO_INT_EN	R/W	每位对应于 GPIO[63:0]的输入中断使能。 0: 关闭中断 1: 使能中断

地址偏移：b00-b3fh                      属性：R/W  
默认值：00000000h                      大小：64

表 27- 13 GPIO 按字节中断使能寄存器

位域	名称	访问	描述
7:0	GPIO_INT_EN	R/W	每字节对应于每一位 GPIO[63:0] (依各系统 GPIO 资源)的输入中断使能。 0: 关闭中断 1: 使能中断

#### 27.4.5 GPIO 中断极性

地址偏移：40-47h                      属性：R/W  
默认值：00000000h                      大小：8

表 27- 14 GPIO 按位中断极性寄存器

位域	名称	访问	描述
63:0	GPIO_INT_POL	R/W	每位对应于 GPIO[63:0]的中断极性。 与中断边沿性配合，组成四种中断触发方式，见下文 GPIO 中断边沿。

地址偏移：c00-c3fh                      属性：R/W  
默认值：00000000h                      大小：64

表 27- 15 GPIO 按字节中断极性寄存器

位域	名称	访问	描述
7:0	GPIO_INT_POL	R/W	每字节对应于每一位 GPIO[63:0] (依各系统 GPIO 资源)的中断极性。 与中断边沿性配合，组成四种中断触发方式，见下文 GPIO 中断边沿。

### 27.4.6 GPIO 中断边沿

地址偏移：50-57h 属性：R/W

默认值：00000000h 大小：8

表 27- 16 GPIO 按位中断边沿寄存器

位域	名称	访问	描述		
63:0	GPIO_INT_EDGE	R/W	每位对应于 GPIO[63:0] 的中断边沿。 与中断极性配合，组成四种中断触发方式。		
			POL	EDGE	描述
			0	0	低电平触发中断
			1	0	高电平触发中断
			0	1	下降沿触发中断
1	1	上升沿触发中断			

地址偏移：d00-d3fh 属性：R/W

默认值：00000000h 大小：64

表 27- 17 GPIO 按字节中断边沿寄存器

位域	名称	访问	描述		
7:0	GPIO_INT_EDGE	R/W	每字节对应于每一位 GPIO[63:0] (依各系统 GPIO 资源) 的中断边沿。 与中断极性配合，组成四种中断触发方式。		
			POL	EDGE	描述
			0	0	低电平触发中断
			1	0	高电平触发中断
			0	1	下降沿触发中断
1	1	上升沿触发中断			

### 27.4.7 GPIO 中断清除

地址偏移：60-67h 属性：R/W

默认值：00000000h 大小：8

表 27- 18 GPIO 按位中断清除寄存器

位域	名称	访问	描述
63:0	GPIO_INT_CLR	R/W	对应于 GPIO[63:0] 的中断清除。 写 1 清除相应 GPIO 位上的中断，随后硬件会自动置 0 中断清除寄存器相应位，不需软件再作处理。

地址偏移：e00-e3fh 属性：R/W

默认值：00000000h 大小：64

表 27- 19 GPIO 按字节中断清除寄存器

位域	名称	访问	描述
7:0	GPIO_INT_CLR	R/W	每字节对应于每一位GPIO[63:0] (依各系统GPIO资源)的中断极性。 写1清除相应GPIO位上的中断, 随后硬件会自动置0中断清除寄存器相应位, 不需软件再作处理。

### 27.4.8 GPIO 中断状态

地址偏移: 70-77h 属性: R/W

默认值: 00000000h 大小: 8

表 27- 20 GPIO 按位中断状态寄存器

位域	名称	访问	描述
63:0	GPIO_INT_STS	R/W	对应于GPIO[63:0]的中断状态。 1: 有中断 0: 无中断

地址偏移: f00-f3fh 属性: R/W

默认值: 00000000h 大小: 64

表 27- 21 GPIO 按字节中断状态寄存器

位域	名称	访问	描述
7:0	GPIO_INT_STS	R/W	每字节对应于每一位GPIO[63:0] (依各系统GPIO资源)的中断状态。 1: 有中断 0: 无中断

### 27.4.9 GPIO 中断双沿模式

地址偏移: 80-87h 属性: R/W

默认值: 00000000h 大小: 8

表 27- 22 GPIO 按位中断双沿寄存器

位域	名称	访问	描述
63:0	GPIO_INT_DUAL	R/W	每位对应于GPIO[63:0]的中断双沿模式。 1: 双沿中断触发 0: 单沿中断触发

地址偏移: f80-fbfh 属性: R/W

默认值: 00000000h 大小: 64

表 27- 23 GPIO 按字节中断双沿寄存器

位域	名称	访问	描述
7:0	GPIO_INT_DUAL	R/W	每字节对应于每一位GPIO[63:0] (依各系统GPIO资源)的中断双沿模式。 1: 双沿中断触发 0: 单沿中断触发