

LOONGSON

龙芯2H处理器用户手册

V1.6

2016年3月

龙芯中科技术有限公司

自主决定命运, 创新成就未来

北京市海淀区中关村环保科技示范园龙芯产业园 100095
Loongson Industrial Park, Zhongguancun Environmental Protection Park,
Haidian District, Beijing 100095. P. R. China



www.loongson.cn

版权声明

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

免责声明

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

龙芯中科技术有限公司

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

电话 (Tel)：010-62546668

传真 (Fax)：010-62600826

阅读指南

《龙芯 2H 处理器用户手册》主要介绍龙芯 2H 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 2H 处理器所集成的 LS464 高性能处理器核的相关资料，请参阅《龙芯 LS464 处理器核用户手册》

修订历史

序号	更新日期	版本号	更新内容
1	2012-11	V1.0	初稿, 第一次正式发布
2	2013-11	V1.1	修正若干拼写错误, 更正 CLKSEL1 描述、EJTAG_DROP 寄存器配置
3	2014-03	V1.2	更新到 2H3 版本, 增加 CHIPID、USB PHY 配置, PCIE 增加总线错屏蔽和插入状态检测, 增加 HPET 和 PWM 章节
4	2014-08	V1.3	文档排版, 修正若干文字错误
5	2014-09	V1.4	修正 LPC/PCIE 地址空间错误, 去除 PCIE 控制器软复位的描述
6	2015-06	V1.5	增加分频器配置方法 (5.4节), 修正若干文字错误
7	2016-03	V1.6	修订 pcie.link_init 函数

手册信息反馈: service@loongson.cn

客户可通过问题反馈网站 <http://bugs.loongnix.org/> 向我司提交芯片产品使用过程中的问题, 并获取技术支持。

目 录

目录	i
第一章 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 媒体处理器	3
1.2.3 GPU	3
1.2.4 显示控制器	3
1.2.5 DDRII 控制器	3
1.2.6 SATA 控制器	4
1.2.7 USB2.0 控制器	4
1.2.8 GMAC 控制器	4
1.2.9 LPC 控制器	4
1.2.10 HDA 控制器	5
1.2.11 NAND 控制器	5
1.2.12 SPI 控制器	5
1.2.13 UART	5
1.2.14 I2C 总线	5
1.2.15 PWM	6
1.2.16 HPET	6
1.2.17 RTC	6
1.2.18 Watchdog	6
1.2.19 中断控制器	6
1.2.20 ACPI 功耗管理	6
1.2.21 PCIe 接口	7
1.2.22 HT 接口	7
第二章 引脚定义	9
2.1 约定	9
2.2 引脚列表	9

第三章 时钟结构	19
3.1 系统参考时钟	19
3.2 CPU 时钟	19
3.3 DDR 时钟	20
3.4 SYS 时钟	20
3.5 显示时钟	21
3.6 时钟信号说明	22
第四章 电源管理	25
4.1 电源管理模块介绍	25
4.2 电源级别	25
4.3 控制引脚说明	26
第五章 芯片配置与控制	27
5.1 芯片工作模式	27
5.1.1 独立 SoC	27
5.1.2 HT 桥片	27
5.1.3 PCIE 桥片	28
5.2 芯片初始化信号	29
5.3 地址空间分配	30
5.4 时钟与复位控制	32
5.4.1 时钟配置概要	32
5.4.2 高频 PLL 配置	32
5.4.3 展频 PLL 配置	33
5.4.4 复位控制	33
5.5 芯片配置寄存器	34
5.5.1 消息中断写端口(msiport)	40
5.5.2 中断状态寄存器(intisr)	41
5.5.3 中断使能寄存器(intien)	41
5.5.4 中断设置寄存器(intset)	41
5.5.5 中断清除寄存器(intclr)	41
5.5.6 中断极性寄存器(intpol)	41
5.5.7 中断模式寄存器(intedge)	41

5.5.8	通用输入输出配置(gpiocfg)	42
5.5.9	通用输入输出方向(gpiooe)	42
5.5.10	通用输入采样(gpioin)	42
5.5.11	通用输出设置(gpioout)	42
5.5.12	DMA 命令控制寄存器(dma_order)	43
5.5.13	芯片配置寄存器 0(chip_config0)	43
5.5.14	芯片配置寄存器 1(chip_config1)	44
5.5.15	芯片配置寄存器 2(chip_config2)	45
5.5.16	芯片配置寄存器 3(chip_config3)	45
5.5.17	芯片采样寄存器 0(chip_sample0)	46
5.5.18	芯片采样寄存器 1(chip_sample1)	47
5.5.19	芯片采样寄存器 2(chip_sample2)	47
5.5.20	芯片采样寄存器 3(chip_sample3)	47
5.5.21	芯片采样寄存器 4(chip_sample4)	47
5.5.22	芯片采样寄存器 5(chip_sample5)	47
5.5.23	时钟配置寄存器 0(clock_ctrl0)	48
5.5.24	时钟配置寄存器 1(clock_ctrl1)	48
5.5.25	时钟配置寄存器 2(clock_ctrl2)	49
5.5.26	时钟配置寄存器 3(clock_ctrl3)	49
5.5.27	显示时钟配置寄存器 0(pixclk0/1_ctrl0)	50
5.5.28	显示时钟配置寄存器 1(pixclk0/1_ctrl1)	50
5.5.29	USB PHY0 配置寄存器(usbphy0_ctrl)	51
5.5.30	USB PHY1 配置寄存器(usbphy1_ctrl)	51
5.6	中断配置及路由	52
第六章	DDR2/3 控制器	55
6.1	功能概述	55
6.2	读操作协议	56
6.3	写操作协议	56
6.4	参数配置格式	56

第七章 HT 控制器	89
7.1 HT 硬件设置及初始化	89
7.2 HT 协议支持	89
7.3 HT 配置寄存器	91
7.3.1 Bridge Control	92
7.3.2 Capability Registers	92
7.3.3 自定义寄存器	95
7.3.4 接收地址窗口配置寄存器	96
7.3.5 中断向量寄存器	97
7.3.6 中断使能寄存器	99
7.3.7 Interrupt Discovery & Configuration	101
7.3.8 POST 地址窗口配置寄存器	102
7.3.9 可预取地址窗口配置寄存器	103
7.3.10 UNCACHE 地址窗口配置寄存器	104
第八章 PCIE 控制器	107
8.1 使用说明	107
8.2 地址空间划分和地址转换机制	107
8.2.1 RC 模式下的地址空间划分	107
8.2.2 EP 模式下的地址空间划分	109
8.2.3 PCI 地址空间到物理地址的转换	110
8.3 龙芯 2H 处理器 PCIE 控制器内部寄存器定义	110
8.3.1 PCIE 端口控制寄存器 0	111
8.3.2 PCIE 端口控制寄存器 1	112
8.3.3 PCIE 端口状态寄存器 0	113
8.3.4 PCIE 端口状态寄存器 1	114
8.3.5 用户定义消息 ID 寄存器	116
8.3.6 PCIE 端口中断状态寄存器	116
8.3.7 PCIE 端口中断状态清除寄存器	119
8.3.8 PCIE 端口中断掩码寄存器	119
8.3.9 PCIE 端口对外配置访问参数寄存器	120
8.3.10 PCIE 端口控制和状态寄存器	120

8.3.11	PHY 状态寄存器	120
8.3.12	用户定制消息寄存器 0	121
8.3.13	用户定制消息寄存器 1	121
8.3.14	用户定制消息数据发送寄存器 0	121
8.3.15	用户定制消息数据发送寄存器 1	122
8.3.16	MSI 参数寄存器	122
8.3.17	地址译码掩码寄存器 0	122
8.3.18	地址译码掩码寄存器 1	122
8.3.19	地址译码转换地址寄存器 0	122
8.3.20	地址译码转换地址寄存器 1	122
8.3.21	接收消息数据负载寄存器 0	123
8.3.22	接收消息数据负载寄存器 1	123
8.4	PCIE 配置头空间	123
8.4.1	Ack Latency Timer and Replay Timer Register	124
8.4.2	Vendor Specific DLLP Register	125
8.4.3	Port Force Link Register	126
8.4.4	Ack frequency and L0-L1 ASPM control Register	126
8.4.5	Port Link Control Register	127
8.4.6	Lane Skew Register	128
8.4.7	Symbol Number Register	129
8.4.8	Symbol Timer Register and Filter Mask Register 1	129
8.4.9	Filter Mask Register 2	131
8.4.10	AMBA Multiple Outbound Decomposed NP Sub-Requests Control Register	131
8.4.11	Transmit Posted FC Credit Status Register	132
8.4.12	Transmit Non-Posted FC Credit Status Register	132
8.4.13	Transmit Completion FC Credit Status Register	132
8.4.14	Queue Status Register	133
8.4.15	VC0 Posted Receive Queue Control	133
8.4.16	VC0 Non-Posted Receive Queue Control	134
8.4.17	VC0 Completion Receive Queue Control	134

8.4.18	VC0 Posted Buffer Depth	134
8.4.19	VC0 Non-Posted Buffer Depth.....	134
8.4.20	VC0 Completion Buffer Depth.....	134
8.4.21	Gen2 Control Register	135
8.4.22	Master Response Composer Control Register 0	136
8.4.23	Master Response Composer Control Register 1	136
8.4.24	MSI Controller Address	136
8.4.25	MSI Controller Upper Address	137
8.5	常用例程	137
第九章	显示控制器	141
9.1	概述.....	141
9.2	寄存器定义	141
9.2.1	帧缓冲配置寄存器	141
9.2.2	帧缓冲地址寄存器 0	141
9.2.3	帧缓冲地址寄存器 1	142
9.2.4	帧缓冲跨度寄存器	142
9.2.5	帧缓冲初始字节寄存器.....	142
9.2.6	颜色抖动配置寄存器.....	142
9.2.7	颜色抖动查找表低位寄存器	142
9.2.8	颜色抖动查找表高位寄存器	143
9.2.9	液晶面板配置寄存器	143
9.2.10	水平显示宽度寄存器	143
9.2.11	行同步配置寄存器	143
9.2.12	垂直显示高度寄存器	144
9.2.13	场同步配置寄存器	144
9.2.14	伽玛校正目录寄存器	144
9.2.15	伽玛校正值寄存器	144
9.2.16	光标配置寄存器	145
9.2.17	光标存储地址寄存器	145
9.2.18	光标显示位置寄存器	145
9.2.19	光标背景色寄存器	145

9.2.20	光标前景色寄存器	146
9.2.21	中断寄存器	146
9.2.22	DAC 控制寄存器	146
9.2.23	DAC Sense 值寄存器	147
9.2.24	Sense 配置寄存器	147
9.2.25	DVO 输出模式寄存器	147
9.3	相关功能说明	147
9.3.1	颜色抖动	147
9.3.2	输出定序	148
9.3.3	伽玛校正	148
9.3.4	硬件光标	149
第十章	GMAC 控制器	151
10.1	DMA 寄存器描述	151
10.1.1	Register0 (Bus Mode Register)	151
10.1.2	Register1 (Transmit Poll Demand Register)	152
10.1.3	Register2 (Receive Poll Demand Register)	152
10.1.4	Register3 (Receive Descriptor List Address Register)	152
10.1.5	Register4 (Transmit Descriptor List Address Register)	153
10.1.6	Register5 (Status Register)	153
10.1.7	Register6 (Operation Mode Register)	154
10.1.8	Register7 (Interrupt Enable Register)	156
10.1.9	Register8 (Missed Frame and Buffer Overflow Counter Register)	157
10.1.10	Register9 (Receive Interrupt Watchdog Timer Register)	157
10.1.11	Register10 (AXI Bus Mode Register)	158
10.1.12	Register11 (AXI Status Register)	159
10.1.13	Register18 (Current Host Transmit Descriptor Register)	159
10.1.14	Register19 (Current Host Receive Descriptor Register)	159
10.1.15	Register20 (Current Host Transmit Buffer Address Register)	160
10.1.16	Register21 (Current Host Receive Buffer Address Register)	160
10.2	GMAC 控制器寄存器描述	160
10.2.1	Register0 (MAC Configuration Register)	160

10.2.2 Register1 (MAC Frame Filter).....	161
10.2.3 Register2 (Hash Table High Register)	162
10.2.4 Register3 (Hash Table Low Register)	162
10.2.5 Register4 (GMII Address Register)	163
10.2.6 Register5 (GMII Data Register)	163
10.2.7 Register6 (Flow Control Register)	163
10.2.8 Register7 (VLAN Tag Register)	164
10.2.9 Register8 (Version Register)	164
10.2.10 Register14 (Interrupt Status Register)	165
10.2.11 Register15 (Interrupt Mask Register)	165
10.2.12 Register16 (MAC Address0 High Register)	166
10.2.13 Register17 (MAC Address0 Low Register).....	166
10.2.14 Register18 (MAC Address1 High Register)	166
10.2.15 Register19 (MAC Address1 Low Register).....	167
10.2.16 Register48 (AN Control Register)	167
10.2.17 Register49 (AN Status Register)	167
10.2.18 Register50 (Auto-Negotiation Advertisement Register)	168
10.2.19 Register51 (Auto-Negotiation Link Partner Ability Register).....	168
10.2.20 Register52 (Auto-Negotiation Expansion Register).....	169
10.2.21 Register54 (SGMII/RGMII Status Register).....	169
10.2.22 Register448 (Time Stamp Control Register)	170
10.2.23 Register449 (Sub-Second Increment Register).....	171
10.2.24 Register 450 (System Time - Seconds Register)	171
10.2.25 Register 451 (System Time - Nanoseconds Register)	172
10.2.26 Register 452 (System Time - Seconds Update Register)	172
10.2.27 Register 453 (System Time - Nanoseconds Update Register)	172
10.2.28 Register 454 (Time Stamp Addend Register)	172
10.2.29 Register 455 (Target Time Seconds Register)	172
10.2.30 Register 456 (Target Time Nanoseconds Register)	173
10.2.31 Register 457 (System Time - Higher Word Seconds Register).....	173
10.2.32 Register 458 (Time Stamp Status Register)	173

10.2.33 Register 459 (PPS Control Register)	174
10.2.34 Register 460 (PPS Auxiliary Time Stamp - Nanoseconds Register)	174
10.2.35 Register 461 (PPS Auxiliary Time Stamp - Seconds Register)	174
10.3 DMA 描述符	175
10.3.1 TDES0	175
10.3.2 TDES1	177
10.3.3 TDES2	177
10.3.4 TDES3	177
10.3.5 TDES6	178
10.3.6 TDES7	178
10.3.7 RDES0	178
10.3.8 RDES1	179
10.3.9 RDES2	180
10.3.10 RDES3	180
10.3.11 RDES4	180
10.3.12 RDES6	181
10.3.13 TDES7	181
10.4 软件编程向导	182
10.4.1 DMA 初始化	182
10.4.2 MAC 初始化	182
10.4.3 发送和接收的一般过程	183
第十一章 SATA 控制器	185
11.1 SATA 总体描述	185
11.2 SATA 控制器内部寄存器描述	185
第十二章 USB 控制器	187
12.1 总体描述	187
12.2 USB 主机控制器寄存器	187
12.2.1 EHCI 相关寄存器	187
12.2.2 Capability 寄存器	187
12.2.3 Operational 寄存器	187
12.2.4 EHCI 实现相关寄存器	188

12.3 OHCI 相关寄存器	190
12.3.1 Operational 寄存器	190
12.3.2 OHCI 实现相关寄存器	191
第十三章 OTG 控制器	193
13.1 概述	193
13.2 寄存器列表	193
13.2.1 全局控制与状态寄存器 (Global CSR Map)	193
13.2.2 模式控制与状态寄存器 (Host Mode CSR)	195
13.2.3 模式控制与状态寄存器 (Device Mode CSR)	195
13.2.4 数据 FIFO 访问寄存器组 (DFIFO Access Register MAP)	196
13.2.5 功耗控制与门控时钟寄存器组 (Power and Clock Gating CSR Map)	197
13.3 寄存器描述	197
13.3.1 寄存器访问特性	197
13.3.2 全局寄存器	197
13.3.3 Host 模式寄存器	218
13.4 Device 模式寄存器	228
13.4.1 功耗与门控时钟寄存器	242
第十四章 HDA 控制器	245
14.1 功能概述	245
14.2 寄存器描述	246
14.2.1 协议定义的音频控制器寄存器集	246
14.2.2 自定义的调试寄存器	249
14.3 内存数据结构	249
第十五章 AC97 控制器	251
15.1 概述	251
15.2 AC97 控制器寄存器	251
15.2.1 配置状态寄存器 (CSR)	252
15.2.2 输出通道配置寄存器 (OCC)	252
15.2.3 输入通道配置寄存器 (ICC)	253
15.2.4 声道格式说明	253
15.2.5 Codec 寄存器访问命令 (Codec)	253

15.2.6 中断状态寄存器 / 中断掩膜寄存器	254
15.2.7 中断状态 / 清除寄存器	254
15.2.8 OC 中断清除寄存器	254
15.2.9 IC 中断清除寄存器	255
15.2.10 CODEC WRITE 中断清除寄存器	255
15.2.11 CODEC READ 中断清除寄存器	255
第十六章 LPC 控制器	257
第十七章 SPI 控制器	259
17.1 SPI 控制器结构	259
17.2 配置寄存器	259
17.3 接口时序	261
17.3.1 SPI 主控制器接口时序	261
17.3.2 SPI Flash 访问时序	261
17.4 使用指南	261
17.4.1 SPI 主控制器的读写操作	261
17.4.2 硬件 SPI Flash 读	263
17.4.3 混合访问 SPI Flash 和 SPI 主控制器	264
第十八章 I2C 控制器	265
18.1 概述	265
18.2 I2C 控制器结构	265
18.3 I2C 控制器寄存器说明	265
18.3.1 分频锁存器低字节寄存器 (PRERlo)	266
18.3.2 分频锁存器高字节寄存器 (PRERhi)	266
18.3.3 控制寄存器 (CTR)	266
18.3.4 发送数据寄存器 (TXR)	266
18.3.5 接受数据寄存器 (RXR)	267
18.3.6 命令控制寄存器 (CR)	267
18.3.7 状态寄存器 (SR)	267
第十九章 UART 控制器	269
19.1 概述	269
19.2 控制器结构	269

19.3 寄存器描述	269
19.3.1 数据寄存器 (DAT)	270
19.3.2 中断使能寄存器 (IER)	270
19.3.3 中断标识寄存器 (IIR)	271
19.3.4 FIFO 控制寄存器 (FCR)	271
19.3.5 线路控制寄存器 (LCR)	271
19.3.6 MODEM 控制寄存器 (MCR)	272
19.3.7 线路状态寄存器 (LSR)	273
19.3.8 MODEM 状态寄存器 (MSR)	273
19.3.9 分频锁存器	274
第二十章 NAND 控制器	275
20.1 NAND 控制器结构描述	275
20.2 NAND 寄存器配置描述	275
20.2.1 命令寄存器 NAND_CMD	275
20.2.2 页内偏移地址寄存器 ADDR_C	276
20.2.3 页地址寄存器 ADDR_R	276
20.2.4 时序寄存器 NAND_TIMING	276
20.2.5 ID 寄存器 ID_L	277
20.2.6 ID 和状态寄存器 STATUS & ID_H	277
20.2.7 参数配置寄存器 NAND_PARAMETER	277
20.2.8 操作数量寄存器 NAND_OP_NUM	277
20.2.9 DMA 读写数据寄存器 DMA_ADDRESS	278
20.3 NAND ADDR 说明	278
20.4 NAND-flash 读写操作举例	280
20.5 NAND ECC 说明	280
20.6 NAND 启动说明	281
第二十一章 PWM 控制器	283
21.1 概述	283
21.2 寄存器描述	283
21.3 功能说明	284
21.3.1 脉宽调制功能	284

21.3.2 脉冲测量功能	285
21.3.3 防死区功能	285
第二十二章 HPET 控制器	287
22.1 概述	287
22.2 寄存器描述	287
第二十三章 DMA 控制器	291
23.1 DMA 控制器结构描述	291
23.2 DMA 控制器与 APB 设备的交互	291
23.3 DMA 控制器	291
23.3.1 下一个描述符地址寄存器 (DMA_ORDER_ADDR)	291
23.3.2 内存地址寄存器 (DMA_SADDR)	292
23.3.3 设备地址寄存器 (DMA_DADDR)	292
23.3.4 长度寄存器 (DMA_LENGTH)	292
23.3.5 间隔长度寄存器 (DMA_STEP_LENGTH)	293
23.3.6 循环次数寄存器 (DMA_STEP_TIMES)	293
23.3.7 控制寄存器 (DMA_CMD)	293
第二十四章 电源管理模块	295
24.1 概述	295
24.2 寄存器描述	295
24.2.1 PMCON_SOC : SOC General PM Configuration Register	295
24.2.2 PMCON_RESUME : RESUME General PM Configuration Register	296
24.2.3 PMCON_RTC : RTC General PM Configuration Register	297
24.2.4 PM1_STS : Power Management 1 Status Register	298
24.2.5 PM1_EN : Power Management 1 Enable Register	299
24.2.6 PM1_CNT : Power Management 1 Control Register	299
24.2.7 PM1_TMR : Power Management 1 Timer	300
24.2.8 P_CNT : Processor Control Register	300
24.2.9 P_LVL2 : Processor LVL_2 Register	300
24.2.10 P_LVL3 : Processor LVL_3 Register	300
24.2.11 GPE0_STS : General Purpose Event0 Status Register	301
24.2.12 GPE0_EN : General Purpose Event0 Status Register	301

24.2.13 RST_CNT : Reset Control Register	302
24.2.14 WD_SET : Watch Dog Set Register	302
24.2.15 WD_Timer : Watch Dog Timer Register	303
24.2.16 DVFS_CNT : Dynamic Voltage Frequency Scaling Control Register	303
24.2.17 DVFS_STS : Dynamic Voltage Frequency Scaling Status Register ·	303
24.2.18 MS_CNT : Media Subsystem Power Control Register	304
24.2.19 MS_THT : Media Subsystem Throttling Register	304
24.2.20 THSENS_CNT : CPU Thermal Sensor Control Register	305
24.2.21 GEN_RTC_1 : General RTC Register 1.....	305
24.2.22 GEN_RTC_2 : General RTC Register 2.....	305
第二十五章 RTC	307
25.1 概述.....	307
25.2 寄存器描述	307
25.2.1 寄存器地址列表	307
25.2.2 TOY 计数器低 32 位数值 (SYS_TOYWRITE0).....	308
25.2.3 TOY 计数器高 32 位数值 (SYS_TOYWRITE1).....	308
25.2.4 TOY 计数器低 32 位数值 (SYS_TOYREAD0)	308
25.2.5 TOY 计数器高 32 位数值 (SYS_TOYREAD1)	308
25.2.6 TOY 计数器中断寄存器 0/1/2(SYS_TOYMATCH0/1/2)	309
25.2.7 RTC 定时器中断寄存器 0/1/2(SYS_RTCCTRL)	309
25.2.8 RTC 计数器写入端口 (SYS_RTCWRITE).....	310
25.2.9 RTC 计数器写入端口 (SYS_RTCREAD)	310
25.2.10 RTC 定时器中断寄存器 0/1/2(SYS_RTCMATCH0/1/2)	310

表 目 录

2.1 信号类型代码	9
2.2 DDR2 SDRAM 控制器接口信号	9
2.3 HT 总线信号	10
2.4 PCIE 总线信号	10
2.5 VGA 接口信号	11
2.6 DVO 接口信号	11
2.7 GMAC 接口信号	11
2.8 SATA 接口信号	11
2.9 USB 接口信号	12
2.10 AC97/HDA 接口信号	12
2.11 LPC 接口信号	12
2.12 SPI 接口信号	12
2.13 I2C 接口信号	12
2.14 UART 接口信号	13
2.15 NAND 接口信号	13
2.16 GPIO 信号	13
2.17 电源接口	14
2.18 电源管理接口	15
2.19 测试接口	16
2.20 EJTAG 接口	16
2.21 时钟信号	16
2.22 其它信号	16
3.1 龙芯 2H 时钟信号	23
4.1 ACPI 状态说明	25
4.2 处理器 Cx 状态	26
4.3 媒体处理器状态 Dx	26

4.4	控制引脚说明	26
5.1	配置信号	29
5.2	地址空间分配之 CPU 视角	30
5.3	地址空间分配之 DMA 视角	31
5.4	高频 PLL 工作条件	33
5.5	展频 PLL 工作条件	34
5.6	芯片配置寄存器列表	34
5.7	消息中断写端口	40
5.8	中断状态寄存器	41
5.9	中断使能寄存器	41
5.10	中断设置寄存器	41
5.11	中断清除寄存器	41
5.12	中断极性寄存器	41
5.13	中断模式寄存器	42
5.14	通用输入输出配置	42
5.15	通用输入输出方向	42
5.16	通用输入采样	42
5.17	通用输出设置	42
5.18	DMA 命令控制寄存器	43
5.19	芯片配置寄存器 0	43
5.20	串口复用配置	44
5.21	芯片配置寄存器 1	45
5.22	芯片配置寄存器 2	45
5.23	芯片配置寄存器 3	45
5.24	芯片采样寄存器 0	46
5.25	芯片采样寄存器 1	47
5.26	芯片采样寄存器 2	47
5.27	芯片采样寄存器 3	47
5.28	芯片采样寄存器 4	47
5.29	芯片采样寄存器 5	47
5.30	时钟配置寄存器 0	48

5.31 时钟配置寄存器 1	48
5.32 时钟配置寄存器 2	49
5.33 时钟配置寄存器 3	49
5.34 显示时钟配置寄存器 0	50
5.35 显示时钟配置寄存器 1	50
5.36 USB PHY0 配置寄存器	51
5.37 USB PHY1 配置寄存器	52
5.38 中断源映射关系	52
5.39 中断控制器配置	53
6.1 DDR2 SDRAM 配置参数寄存器格式	57
7.1 HyperTransport 接收端可接收的命令	90
7.2 HyperTransport 发送端会向外发送的命令	90
7.3 HT 配置寄存器列表	91
8.1 mem0 空间划分	108
8.2 mem1 空间划分	108
8.3 RC 模式下 cfg&ctrl 空间划分	108
8.4 EP 模式下 cfg&ctrl 空间划分	109
8.5 寄存器属性类型	111
8.6 PCIE 端口控制寄存器 0	111
8.7 PCIE 端口控制寄存器 1	112
8.8 PCIE 端口状态寄存器 0	113
8.9 PCIE 端口状态寄存器 1	114
8.10 用户定义消息 ID 寄存器	116
8.11 PCIE 端口中断状态寄存器	116
8.12 PCIE 端口对外配置访问参数寄存器	120
8.13 PCIE 端口控制和状态寄存器	120
8.14 PHY 状态寄存器	121
8.15 用户定制消息寄存器 0	121
8.16 用户定制消息寄存器 1	121
8.17 MSI 参数寄存器	122

8.18 PCI Standard Capability Structures	123
8.19 Port Logic 寄存器地址划分	123
8.20 Ack Latency Timer and Replay Timer Register	124
8.21 Vendor Specific DLLP Register	126
8.22 Port Force Link Register	126
8.23 Ack frequency and L0-L1 ASPM control Register	127
8.24 Port Link Control Register	128
8.25 Lane Skew Register	128
8.26 Symbol Number Register	129
8.27 Symbol Timer Register and Filter Mask Register 1	130
8.28 Filter Mask Register 2	131
8.29 AMBA Multiple Outbound Decomposed NP Sub-Requests Control Register	131
8.30 Transmit Posted FC Credit Status Register	132
8.31 Transmit Non-Posted FC Credit Status Register	132
8.32 Transmit Completion FC Credit Status Register	133
8.33 Queue Status Register	133
8.34 VC0 Posted Receive Queue Control	133
8.35 VC0 Non-Posted Receive Queue Control	134
8.36 VC0 Completion Receive Queue Control	134
8.37 VC0 Posted Buffer Depth	134
8.38 VC0 Non-Posted Buffer Depth	134
8.39 VC0 Completion Buffer Depth	135
8.40 Gen2 Control Register	135
8.41 Master Response Composer Control Register 0	136
8.42 Master Response Composer Control Register 1	136
8.43 MSI Controller Address	136
8.44 MSI Controller Upper Address	137
9.1 帧缓冲配置寄存器	141
9.2 帧缓冲地址寄存器 0	142
9.3 帧缓冲地址寄存器 1	142
9.4 帧缓冲跨度寄存器	142

9.5 帧缓冲初始字节寄存器	142
9.6 颜色抖动配置寄存器	142
9.7 颜色抖动查找表低位寄存器	142
9.8 颜色抖动查找表高位寄存器	143
9.9 液晶面板配置寄存器	143
9.10 水平显示宽度寄存器	143
9.11 行同步配置寄存器	144
9.12 垂直显示高度寄存器	144
9.13 场同步配置寄存器	144
9.14 伽玛校正目录寄存器	144
9.15 伽玛校正值寄存器	144
9.16 光标配置寄存器	145
9.17 光标存储地址寄存器	145
9.18 光标显示位置寄存器	145
9.19 光标背景色寄存器	145
9.20 光标前景色寄存器	146
9.21 中断寄存器	146
9.22 DAC 控制寄存器	146
9.23 DAC Sense 值寄存器	147
9.24 Sense 配置寄存器	147
9.25 DVO 输出模式寄存器	147
9.26 单色光标颜色配置	149
10.1 Register0 (Bus Mode Register)	151
10.2 Register1 (Transmit Poll Demand Register)	152
10.3 Register2 (Receive Poll Demand Register)	152
10.4 Register3 (Receive Descriptor List Address Register)	152
10.5 Register4 (Transmit Descriptor List Address Register)	153
10.6 Register5 (Status Register)	153
10.7 Register6 (Operation Mode Register)	154
10.8 Register7 (Interrupt Enable Register)	156
10.9 Register8 (Missed Frame and Buffer Overflow Counter Register)	157

10.10 Register9(Receive Interrupt Watchdog Timer Register)	158
10.11 Register10 (AXI Bus Mode Register)	158
10.12 Register11 (AXI Status Register)	159
10.13 Register18 (Current Host Transmit Descriptor Register)	159
10.14 Register19 (Current Host Receive Descriptor Register)	159
10.15 Register20 (Current Host Transmit Buffer Address Register)	160
10.16 Register21 (Current Host Receive Buffer Address Register)	160
10.17 Register0 (MAC Configuration Register)	160
10.18 Register1 (MAC Frame Filter)	161
10.19 Register2 (Hash Table High Register)	162
10.20 Register3 (Hash Table Low Register)	162
10.21 Register4 (GMII Address Register)	163
10.22 Register5 (GMII Data Register)	163
10.23 Register6 (Flow Control Register)	164
10.24 Register7 (VLAN Tag Register)	164
10.25 Register8 (Version Register)	165
10.26 Register14 (Interrupt Status Register)	165
10.27 Register15 (Interrupt Mask Register)	165
10.28 Register16 (MAC Address0 High Register)	166
10.29 Register17 (MAC Address0 Low Register)	166
10.30 Register18 (MAC Address1 High Register)	166
10.31 Register19 (MAC Address1 Low Register)	167
10.32 Register48 (AN Control Register)	167
10.33 Register49 (AN Status Register)	167
10.34 Register50 (Auto-Negotiation Advertisement Register)	168
10.35 Register51 (Auto-Negotiation Link Partner Ability Register)	168
10.36 Register52 (Auto-Negotiation Expansion Register)	169
10.37 Register54 (SGMII/RGMII Status Register)	169
10.38 Register448 (Time Stamp Control Register)	170
10.39 Register449 (Sub-Second Increment Register)	171
10.40 Register 450 (System Time - Seconds Register)	171

10.41 Register 451 (System Time - Nanoseconds Register)	172
10.42 Register 452 (System Time - Seconds Update Register)	172
10.43 Register 453 (System Time - Nanoseconds Update Register)	172
10.44 Register 454 (Time Stamp Addend Register)	172
10.45 Register 455 (Target Time Seconds Register)	173
10.46 Register 456 (Target Time Nanoseconds Register)	173
10.47 Register 457 (System Time - Higher Word Seconds Register)	173
10.48 Register 458 (Time Stamp Status Register)	173
10.49 Register 459 (PPS Control Register)	174
10.50 Register 460 (PPS Auxiliary Time Stamp - Nanoseconds Register)	174
10.51 Register 461 (PPS Auxiliary Time Stamp - Seconds Register)	175
10.52 TDES0	176
10.53 TDES1	177
10.54 TDES2	177
10.55 TDES3	178
10.56 TDES6	178
10.57 TDES7	178
10.58 RDES0	178
10.59 RDES1	179
10.60 RDES2	180
10.61 RDES3	180
10.62 RDES4	180
10.63 RDES6	181
10.64 TDES7	182
11.1 SATA 控制器内部寄存器	185
16.1 LPC 控制器地址空间分布	257
16.2 LPC 寄存器 0	258
16.3 LPC 寄存器 1	258
16.4 LPC 寄存器 2	258
16.5 LPC 寄存器 3	258

17.1 SPI 配置寄存器列表	259
17.2 控制寄存器 (SPCR)	260
17.3 状态寄存器 (SPSR)	260
17.4 数据寄存器 (TxFIFO/RxFIFO)	260
17.5 外部寄存器 (SPER)	260
17.6 SPI 分频系数	260
17.7 参数控制寄存器 (SFC_PARAM)	261
17.8 片选控制寄存器 (SFC_SOFTCS)	261
17.9 时序控制寄存器 (SFC_TIMING)	261
21.1 四路 PWM 控制器	283
21.3 PWM 控制寄存器设置	283
22.1 HPET 寄存器列表	287
22.2 General Capabilities and ID Register	288
22.3 General Configuration Register	288
22.4 General Interrupt Status Register	288
22.5 Main Counter Value Register	289
22.6 Timer N Configuration and Capabilities Register	289
22.7 Timer N Comparator Value Register	289

图 目 录

1.1 龙芯 2H 结构图	2
3.1 时钟结构	19
3.2 系统参考时钟	19
3.3 CPU 时钟结构	20
3.4 DDR 时钟结构	20
3.5 SYS 时钟结构	21
3.6 显示时钟结构	21
5.1 龙芯 2H 单芯片系统	27
5.2 龙芯 3A+2H 双芯片系统	28
5.3 高频 PLL 概念性结构	32
5.4 展频 PLL 概念性结构	33
5.5 中断结构示意	54
6.1 DDR2 SDRAM 行列地址与 CPU 物理地址的转换	55
6.2 DDR2 SDRAM 读操作协议	56
6.3 DDR2 SDRAM 写操作协议	57
8.1 PCIE 控制器结构	107
8.2 PCIE 配置空间划分	125
9.1 显示控制器输出定序图	148
10.1 增强型发送描述符	175
10.2 增强型接收描述符	184
13.1 OTG CSRs 地址映射	194
14.1 HDA 模块的整体设计框图	245
15.1 AC97 应用系统	251

17.1 SPI 控制器结构	259
17.2 SPI 主控制器接口时序	262
17.3 SPI Flash 标准读时序	262
17.4 SPI Flash 快速读时序	263
17.5 SPI Flash 双向 I/O 读时序	263
18.1 I2C 主控制器结构	265
19.1 UART 控制器结构	270
21.1 防死区功能例图	285

第一章 概述

龙芯 2H 是龙芯 2G 处理器与龙芯 1A(2F 南桥) 的后继产品, 其目标是为安全适用计算机提供单片解决方案。龙芯 2H 采用 65nm 工艺实现, 主频达 1GHz 以上。片内集成定点处理器、浮点处理器、流媒体处理和图形图像处理功能, 以及南桥、北桥等配套芯片组功能。

龙芯 2H 具有以下关键特性:

- 集成一个 GS464 四发射龙芯处理器核, L1 Cache(I/D) 64KB, L2 Cache 512KB
- 集成 GS232v 媒体处理器, 支持 H264/AVS/VC-1 解码, 720p
- 集成 3D GPU, 兼容 OpenGL ES2.0
- 集成两路 DC 控制器, 最大分辨率可支持到 1920*1080@60Hz/24bit
- 集成 8 位 HT 控制器, 支持从模式
- 集成 PCIE gen2 控制器, 支持 1x4 和 4x1, 支持主从模式
- 集成 2 个 10M/100M/1000M 自适应 GMAC
- 集成 2 个 SATA2
- 集成 1 个 64 位 DDR2/3 控制器
- 集成 6 个 USB HOST 接口, 其中 1 个可配置为 OTG 接口
- 集成 1 个 8 位 NAND FLASH 控制器, 支持 MLC, 支持系统启动
- 集成中断控制器, 支持灵活的中断设置
- 集成 1 个 SPI 控制器, 支持系统启动
- 集成 AC97/HDA 控制器
- 集成 1 个 LPC 控制器
- 集成 4 路 UART 串口
- 集成 2 路 I2C 控制器, 兼容 SMBUS
- 集成 16 路 GPIO 端口
- 集成 1 个 RTC
- 集成 4 路 PWM 控制器
- 集成 ACPI, 支持 USB/ 网络唤醒
- 集成看门狗电路

1.1 体系结构框图

龙芯 2H 内部采用多级总线结构。处理器核、内存控制器、图形媒体模块、PCIE 和南桥使用交叉开关互连。南桥内为共享总线, 连接 GMAC、USB、SATA、HDA、

DMA 等 IO 设备。低速外设（I2C/UART 等）作为一个集合加在南桥总线上。

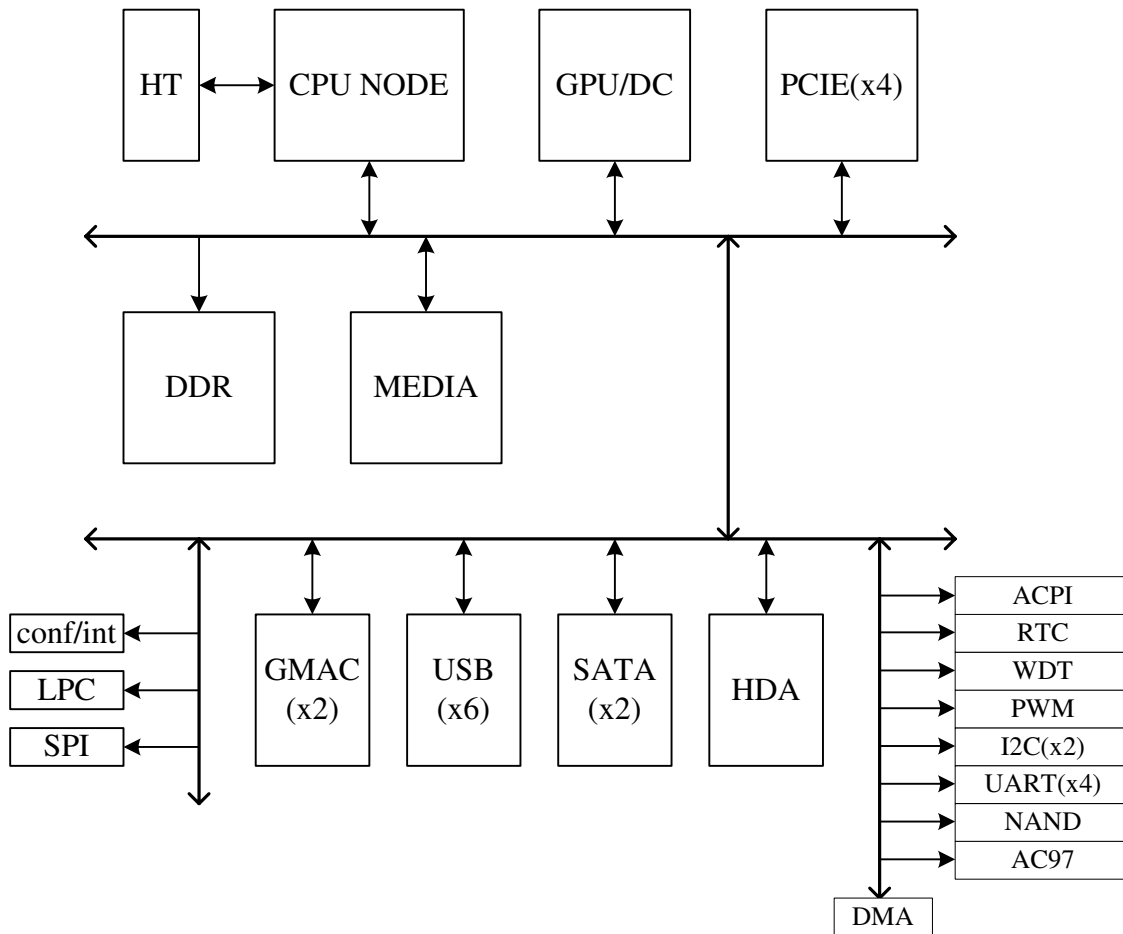


图 1.1: 龙芯 2H 结构图

1.2 芯片主要功能

1.2.1 处理器核

- MIPS64 R2 体系结构兼容
- 包括 2 个全流水的 64 位双精度浮点乘加部件
- 64KB 数据 Cache 和 64KB 的指令 Cache
- 512K 二级 Cache
- 通过目录协议维护 I/O DMA 访问的 Cache 一致性
- x86 虚拟机支持
- EJTAG 支持

1.2.2 媒体处理器

- 支持 AVS、H.264、VC-1 解码
- 支持 720p 高清

1.2.3 GPU

- 支持 OpenGL ES2.0, OpenGL ES 1.1
- 支持 OpenVG
- 通过 Futuremark 认证
- 动态电源管理
- 支持 BitBLT 和 Stretch BLT
- 矩形填充
- 硬件画线
- 单色字体渲染
- ROP2, ROP3, ROP4
- Alpha 混合
- 32Kx32K 坐标系统
- 90 度旋转
- 透明支持
- YUV 色域空间转换
- 高质量缩放

1.2.4 显示控制器

- 双显示输出 (CRT/DVO)
- 硬件光标
- 伽玛校正
- 输出抖动
- 最高像素时钟 (CRT 200MHz, DVO165MHz)
- 支持线性显示缓冲
- 上电序列控制
- 低功耗管理
- VGA 电源管理

1.2.5 DDRII 控制器

- 64 位 DDRII/III 控制器

- 可配置为 32/16 位模式
- 接口上命令、读写数据全流水操作
- 内存命令合并、排序提高整体带宽
- 内建动态延迟补偿电路（DCC），用于数据的可靠发送和接收
- ECC 功能可以对数据通路上的 1 位和 2 位错误进行检测，并能对 1 位错误进行自动纠错

1.2.6 SATA 控制器

- 2 个独立 SATA 端口
- 支持 SATA 1.5Gbps 和 SATA2 代 3Gbps 的传输
- 兼容串行 ATA 2.6 规范和 AHCI 1.1 规范
- 低功耗设计

1.2.7 USB2.0 控制器

- 6 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 和 USB2.0
- 内部 EHCI 控制和实现高速传输可达 480 Mbps
- 内部 OHCI 控制和实现全速和低速传输
- 低功耗管理

1.2.8 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 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
- 支持网络开机

1.2.9 LPC 控制器

- 兼容 LPC Rev1.1 标准
- 支持系统启动

1.2.10 HDA 控制器

- 支持 16，18 和 20 位采样精度支持可变速率
- 最高达 192KHz
- 7.1 频道环绕立体声输出
- 三路音频输入

1.2.11 NAND 控制器

- 最大支持 4GB NAND Flash
- 支持 MLC
- 支持系统启动
- 支持 512/2K/4K/8K 页

1.2.12 SPI 控制器

- 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
- 支持标准读、连续地址读、快速读、双路 I/O 等 SPI Flash 读模式

1.2.13 UART

- 1 个全功能 UART
- 在寄存器与功能上兼容 NS16550A
- 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
- 可配置为 4 个两线串口 (TXD,RXD)

1.2.14 I2C 总线

- 兼容 SMBUS (100Kbps)
- 与 PHILIPS I2C 标准相兼容
- 履行双向同步串行协议

- 只实现主设备操作
- 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
- 支持时钟延伸和等待状态

1.2.15 PWM

- 32 位计数器
- 支持脉冲生成及捕获
- 4 路控制器

1.2.16 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

1.2.17 RTC

- 计时精确到 0.1 秒
- 可产生 3 个计时中断
- 支持定时开机功能

1.2.18 Watchdog

- 32 比特计数器及初始化寄存器

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

1.2.20 ACPI 功耗管理

- 处理器核动态频率电压调节
- 媒体处理器可关断

- 全芯片时钟门控
- PHY 可关断
- USB/GMAC 可唤醒
- 来电可自动启动

1.2.21 PCIe 接口

- 兼容 PCIe 2.0
- 外部连接支持 X4*1 或者 X1*4
- 支持作为 PCIe 桥片

1.2.22 HT 接口

- 支持从模式，可作为 HT 桥片
- 兼容 HyperTransport 1.03
- 接口频率支持 200/400/800Mhz
- 接口宽度支持 8 位模式

第二章 引脚定义

2.1 约定

本章对龙芯 2H 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 结尾，高有效信号则不带 n。如无特别说明，以 ACPI/GMAC/USB 开头的信号位于 RSM 域；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 引脚列表

表 2.2: DDR2 SDRAM 控制器接口信号

信号名称	类型	描述
DDR_DQ[63:0]	I/O	DDR2/3 SDRAM 数据总线信号
DDR_CB[7:0]	I/O	DDR2/3 SDRAM 数据总线 ECC 信号
DDR_DQSp[8:0] DDR_DQSn[8:0]	DIFF I/O	DDR2/3 SDRAM 数据选通（包括 ECC）
DDR_DQM[8:0]	O	DDR2/3 SDRAM 数据屏蔽（包括 ECC）
DDR_A[14:0]	O	DDR2/3 SDRAM 地址总线信号
DDR_BA[2:0]	O	DDR2/3 SDRAM 逻辑 Bank 地址信号
DDR_WEn	O	DDR2/3 SDRAM 写使能信号
DDR_CASn	O	DDR2/3 SDRAM 列地址选择信号
DDR_RASn	O	DDR2/3 SDRAM 行地址选择信号
DDR_CS[3:0]	O	DDR2/3 SDRAM 片选信号
DDR_CKE[3:0]	O	DDR2/3 SDRAM 时钟使能信号

信号名称	类型	描述
DDR_CKp[5:0] DDR_CKn[5:0]	DIFF OUT	DDDR2/3 SDRAM 差分时钟输出信号 1,3,5为一组 DIMM 时钟, 0,2,4为另一组 DIMM 时钟
DDR_ODT[3:0]	O	DDR2/3 SDRAM ODT 信号
DDR_RESETh	O	DDR2/3 SDRAM 复位控制信号
ACPI_CKE[3:0]	O	DDR2/3 SDRAM 时钟使能信号 (RSM 域 3.3V)

表 2.3: HT 总线信号

信号名称	类型	描述
HTCLKp HTCLKn	DIFF OUT	HT 总线参考时钟 (200MHz)
HT_POWEROK	I/O	HT 总线 PowerOK 信号
HT_RSTn	I/O	HT 总线 Reseth 信号
HT_LDT_STOPn	I/O	HT 总线 Ldt_Stopn 信号
HT_LDT_REQn	I/O	HT 总线 Ldt_Reqn 信号
HT_TX_CADp[7:0] HT_TX_CADn[7:0]	DIFF OUT	HT 总线发送数据命令总线
HT_TX_CTLp[0] HT_TX_CTLn[0]	DIFF OUT	HT 总线发送控制信号
HT_TX_CLKp[0] HT_TX_CLKn[0]	DIFF OUT	HT 总线发送时钟总线
HT_RX_CADp[7:0] HT_RX_CADn[7:0]	DIFF IN	HT 总线接收数据命令总线
HT_RX_CTLp[0] HT_RX_CTLn[0]	DIFF IN	HT 总线接收控制信号
HT_RX_CLKp[0] HT_RX_CLKn[0]	DIFF IN	HT 总线接收时钟信号

表 2.4: PCIE 总线信号

信号名称	类型	描述
PCIE_XTAL[2:1]	DIFF IN	PCIE 参考时钟输入
PCIE[3:0]_CLKp PCIE[3:0]_CLKn	DIFF OUT	PCIE 参考时钟输出
PCIE_REFRES	A	外部参考电阻 通过 487ohm(+/-1%) 电阻连至 PLL_PCIE_DVDD 电源
PCIE[3:0]_TXp PCIE[3:0]_TXn	DIFF OUT	PCIE 差分数据输出
PCIE[3:0]_RXp PCIE[3:0]_RXn	DIFF IN	PCIE 差分数据输入
PCIE[3:0]_PRSNT	I	PCIE 插卡检测
PCIE_RSTn	O	PCIE 复位

表 2.5: VGA 接口信号

信号名称	类型	描述
VGA_ROUT	A	VGA 红色通道输出
VGA_GOUT	A	VGA 绿色通道输出
VGA_BOUT	A	VGA 蓝色通道输出
VGA_HSYNC	O	VGA 水平同步
VGA_VSYNC	O	VGA 垂直同步
VGA_IDUMP	G	模拟地
VGA_REXTP VGA_REXTN	A	外部参考电阻 4.12k : 满幅电流 18.7mA 7.81k : 满幅电流 10mA

表 2.6: DVO 接口信号

信号名称	类型	描述
DVO_CLKp	O	DVO 时钟输出
DVO_CLKn	O	DVO 时钟输出
DVO_HSYNC	O	DVO 水平同步
DVO_VSYNC	O	DVO 垂直同步
DVO_DE	O	DVO 数据有效
DVO_D[23:0]	O	DVO 显示数据

表 2.7: GMAC 接口信号

信号名称	类型	描述
GMAC[1:0]_TXCK	O	RGMII 发送时钟
GMAC[1:0]_TCTL	O	RGMII 发送控制
GMAC[1:0]_TXD[3:0]	O	RGMII 发送数据
GMAC[1:0]_RXCK	I	RGMII 接收时钟
GMAC[1:0]_RCTL	I	RGMII 接收控制
GMAC[1:0]_RXD[3:0]	I	RGMII 接收数据
GMAC[1:0]_MDCK	O	SMA 接口时钟
GMAC[1:0]_MDIO	I/O	SMA 接口数据

表 2.8: SATA 接口信号

信号名称	类型	描述
SATA_XTAL[2:1]	I/O	参考时钟晶体
SATA_REFRES	A	外部参考电阻 通过 487ohm(+/-1%) 电阻连至 PLL_SATA_DVDD 电源
SATA[1:0]_TXp SATA[1:0]_TXn	DIFF OUT	SATA 差分数据输出
SATA[1:0]_RXp SATA[1:0]_RXn	DIFF IN	SATA 差分数据输入
SATA[1:0]_LEDn	OD	SATA 工作状态, 低表示有数据传输

表 2.9: USB 接口信号

信号名称	类型	描述
USB_XI[1:0] USB_XO[1:0]	I/O	参考时钟晶体
USB_TXRTUNE[1:0]	A	参考电阻, 分别经 43.2ohm 电阻连接到地
USB[5:0]_DP	I/O	USB D+
USB[5:0]_DM	I/O	USB D-
USB[5:0]_OC	I	USB 过流检测 USB0_OC 复用为 OTG DRVVBUS 输出
USB0.ID	I	USB0 OTG ID 输入
USB0.VBUS	A	USB0 OTG VBUS 输入

表 2.10: AC97/HDA 接口信号

信号名称	类型	描述
AC97.BITCLK	I/O	AC97 BITCLK 输入 HDA BITCLK 输出
AC97.SDATAI	I	AC97/HDA 数据输入
AC97.SDATAO	O	AC97/HDA 数据输出
AC97.SYNC	O	AC97/HDA 同步
AC97.RESET	O	AC97/HDA 复位

表 2.11: LPC 接口信号

信号名称	类型	描述
LPC.CLKIN	I	LPC 时钟输入
LPC.CLK[1:0]	O	相位相同的两路 33MHz 时钟输出
LPC.RESETn	O	LPC 复位输出, 需外部下拉
LPC.FRAMEn	I/O	LPC FRAME 控制信号 主控: 输出 ROM: 输入
LPC.AD[3:0]	I/O	LPC 总线数据
LPC.SIRQ	I/O	LPC 中断线

表 2.12: SPI 接口信号

信号名称	类型	描述
SPLSCK	O	SPI 时钟输出
SPLCSn	O	SPI 片选 0
SPLSDO	O	SPI 数据输出
SPLSDI	I	SPI 数据输入

表 2.13: I2C 接口信号

信号名称	类型	描述
IIC[1:0]_SCL	O	I2C 时钟
IIC[1:0]_SDA	I/O	I2C 数据

表 2.14: UART 接口信号

信号名称	类型	描述
UART0_TXD	O	UART0 数据发送
UART0_RXD	I	UART0 数据接收

表 2.15: NAND 接口信号

信号名称	类型	描述
NAND_CLE	O	NAND 命令锁存
NAND_ALE	O	NAND 地址锁存
NAND_RD	O	NAND 读信号
NAND_WR	O	NAND 写信号
NAND_CE	O	NAND 片选 0
NAND_RDY	I	NAND 准备好 0
NAND_D[7:0]	I/O	NAND 数据线

表 2.16: GPIO 信号

信号名称	类型	描述
GPIO00	I/O	通用输入输出
HDA_SDATAI1	I	HDA 数据输入, 连接第二个 codec
GPIO01	I/O	通用输入输出
HDA_SDATAI2	I	HDA 数据输入, 连接第三个 codec
GPIO02	I/O	通用输入输出
UART0_RTS	O	8 线全功能串口信号, 更多复用见表 5 19
GPIO03	I/O	通用输入输出
UART0_DTR	O	8 线全功能串口信号
GPIO04	I/O	通用输入输出
UART0_RI	I/O	8 线全功能串口信号
GPIO05	I/O	通用输入输出
UART0_CTS	I	8 线全功能串口信号
GPIO06	I/O	通用输入输出
UART0_DSR	I	8 线全功能串口信号
GPIO07	I/O	通用输入输出
UART0_DCD	I	8 线全功能串口信号
GPIO08	I/O	通用输入输出
SPL_CS _{n1}	O	SPI 片选 1
GPIO09	I/O	通用输入输出
SPL_CS _{n2}	O	SPI 片选 2
GPIO10	I/O	通用输入输出
NAND_CE1	O	NAND 片选 1
GPIO11	I/O	通用输入输出
NAND_RDY1	I	NAND 准备好 1
GPIO12	I/O	通用输入输出
PWM0	O	PWM0 输出

信号名称	类型	描述
GPIO13	I/O	通用输入输出
PWM1	O	PWM1 输出
GPIO14	I/O	通用输入输出
PWM2	I/O	PWM2 输入 / 输出
GPIO15	I/O	通用输入输出
PWM3	I/O	PWM3 输入 / 输出

表 2.17: 电源接口

信号名称	类型	描述
PLL.CORE_AVDD	P	Core PLL 1.8V 模拟电源
PLL.CORE_AGND	G	Core PLL 模拟地
PLL.CORE_DVDD	P	Core PLL 1.1V 数字电源
PLL.CORE_DGND	G	Core PLL 数字地
PLL.SYS_AVDD	P	System PLL 1.8V 模拟电源
PLL.SYS_AGND	G	System PLL 模拟地
PLL.SYS_DVDD	P	System PLL 1.1V 数字电源
PLL.SYS_DGND	G	System PLL 数字地
PLL.HT_AVDD	P	HT PLL 1.8V 模拟电源
PLL.HT_AGND	G	HT PLL 模拟地
PLL.HT_DVDD	P	HT PLL 1.1V 数字电源
PLL.HT_DGND	G	HT PLL 数字地
PLL.DDR_AVDD	P	DDR PLL 2.5V 模拟电源
PLL.DDR_AGND	G	DDR PLL 模拟地
PLL.DDR_DVDD	P	DDR PLL 1.2V 数字电源
PLL.DDR_DGND	G	DDR PLL 数字地
PLL.VGA_AVDD	P	VGA PLL 2.5V 模拟电源
PLL.VGA_AGND	G	VGA PLL 模拟地
PLL.VGA_DVDD	P	VGA PLL 1.2V 数字电源
PLL.VGA_DGND	G	VGA PLL 数字地
PLL.DVO_AVDD	P	DVO PLL 2.5V 模拟电源
PLL.DVO_AGND	G	DVO PLL 模拟地
PLL.DVO_DVDD	P	DVO PLL 1.2V 数字电源
PLL.DVO_DGND	G	DVO PLL 数字地
PLL.PCIE_AVDD	P	PCIE PLL 2.5V 模拟电源
PLL.PCIE_DVDD	P	PCIE PLL 1.2V 模拟电源
PLL.PCIE_VSS	G	PCIE PLL 地
PLL.SATA_AVDD	P	SATA PLL 2.5V 模拟电源
PLL.SATA_DVDD	P	SATA PLL 1.2V 模拟电源
PLL.SATA_VSS	G	SATA PLL 地
THSENS_AVDD	P	Thermal sensor 2.5V 电源
THSENS_AGND	G	Thermal sensor 地
SENSE_VCPU	O	CPU 域电源测试点
SENSE_VSOC	O	SOC 域电源测试点
SENSE_VSS	O	CPU/SOC 地测试点

信号名称	类型	描述
VDD_CPU	P	CPU 域电源 (0.95V 1.3V)
VDD_SOC	P	SOC 域电源 (1.15V)
VDD_RSM	P	RSM 域电源 (1.15V)
VSS	G	核心地
RTC_VDD	P	RTC 域电源 (1.3 2.5V)
RTC_GND	G	RTC 域地
VDDE3V3	P	SOC 域 3.3V IO 电源
RSM3V3	P	RSM 域 3.3V IO 电源
RSM2V5	P	RSM 域 2.5V IO 电源
VSSE	G	IO 地
DDR_VDDQ	P	DDR IO 1.8/1.5V 电源
DDR_VREF	P	DDR 参考电压
HT_VDD	P	HT PHY CORE 电源 (1.1V)
HT_VSS	G	HT PHY CORE 地
HT_1V8	P	HT PHY IO 电源 (1.8V)
HT_VSSE	G	HT PHY IO 地
USB_A3V3	P	USB 3.3V 模拟电源
USB_A2V5	P	USB 2.5V 模拟电源
USB_AVSS	G	USB 模拟地
VGA_A2V5	P	VGA 2.5V 模拟电源
VGA_AVSS	G	VGA 模拟地
PEST_1V2T	P	PCIE/SATA 发送端 1.2V 电源
PCIE_VSST	G	PCIE 发送端地
SATA_VSST	G	SATA 发送端地
PCIE_1V2R	P	PCIE 接收端 1.2V 电源
PCIE_VSSR	G	PCIE 接收端地
SATA_1V2R	P	SATA 接收端 1.2V 电源
SATA_VSSR	G	SATA 接收端地

表 2.18: 电源管理接口

信号名称	类型	描述
ACPLSYSRSTn	I	系统复位
ACPLRSMRSTn	I	RSM 域复位 (RTC 电压域)
ACPLRSTCRSTn	I	RTC 域复位 (RTC 电压域)
ACPLRINGn	I	振铃唤醒
ACPLWAKEn	I	PCIE 唤醒
ACPLLID	I	屏盖状态
ACPLPWRTYPE	I	供电来源
ACPLBATLOWn	I	电源电量低
ACPLSUSSTATn	O	低功耗状态
ACPLS3n	O	S3 状态
ACPLS4n	O	S4 状态
ACPLS5n	O	S5 状态

信号名称	类型	描述
ACPLVID[5:0]	O	调压控制
ACPLPLTRSTn	O	平台复位
ACPLSLPLANn	O	网络电源控制
ACPLPWRBTNn	I	电源开关
ACPLPWROK	I	电源有效

表 2.19: 测试接口

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

表 2.20: EJTAG 接口

信号名称	类型	描述
EJTAG_SEL	I	JTAG 选择 (0: JTAG, 1: EJTAG)
EJTAG_TCK	I	JTAG 时钟
EJTAG_TDI	I	JTAG 数据输入
EJTAG_TMS	I	JTAG 模式
EJTAG_TRST	I	JTAG 复位
EJTAG_TDO	O	JTAG 数据输出

表 2.21: 时钟信号

信号名称	类型	描述
SYS.BAKCLK	I	100MHz 备份参考时钟
SYS.TESTCLK	I	测试时钟
SATA_XTAL[2:1]	I/O	参考时钟晶体 (25MHz)
RTC_XI RTC_XO	I/O	RTC 参考时钟晶体 (32.768KHz)

表 2.22: 其它信号

信号名称	类型	描述
SYS.CLKSEL0	I	系统参考时钟选择 0: SATA_XTAL 1: SYS.BAKCLK
SYS.CLKSEL1	I	PCIE 参考时钟选择 0: 从 PCIE_XTAL 输入 1: 内部 100MHz 时钟
ACPL.CLKSEL2	I	USB PHY1 时钟选择 0: 内部时钟 1: 在 USB_XI1/XO1 上的 12MHz 晶体

信号名称	类型	描述
ACPI_CLKSEL3	I	ACPI 使能 0: 不使用 ACPI 控制芯片 1: 使用 ACPI 控制
SYS_NMI _n	I/O	不可屏蔽中断 桥片模式下为输出, 否则为输入
SYS_INT _n	I/O	普通中断 桥片模式下为输出, 否则为输入

第三章 时钟结构

龙芯 2H 的时钟结构如图3.1所示，片内的时钟主要由一个系统参考时钟经由多个时钟复位生成器 (RCG) 产生。本章将详细介绍时钟的结构及其配置情况。

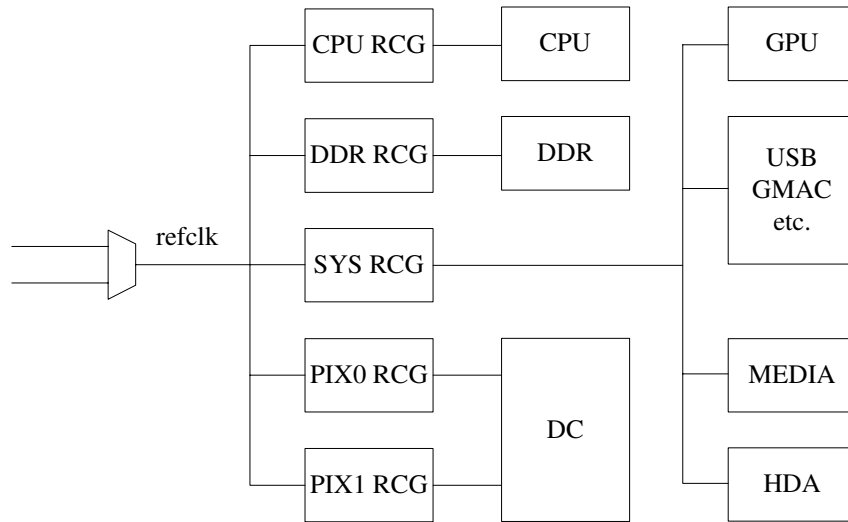


图 3.1: 时钟结构

3.1 系统参考时钟

系统参考时钟有两种来源，分别为 SATA_XTAL 和 SYS_BAKCLK。前者不需要有源晶振，可降低系统成本，因而推荐使用。SYS_BAKCLK 为后备参考时钟，建议在测试系统中连接，若使用 PCIE 参考时钟输出则必须连接并选用。

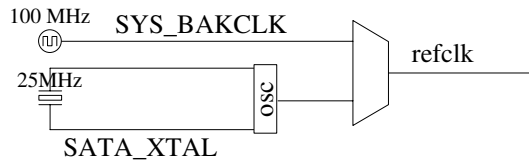


图 3.2: 系统参考时钟

3.2 CPU 时钟

CPU 时钟有硬件跳线和软件控制两种配置方式。硬件跳线使用 NAND_D[2:0] 作为配置信号 (bootcfg[6:4])，只能有少数几种频率选择。硬件配置模式下，参考时钟必须

为 100MHz。软件控制方式可以对所有的参数进行控制，通过 clock_ctrl0 寄存器进行配置，其定义见表5.30。CPU 时钟由高频 PLL(HPLL) 产生，其倍频后频率输出应当在 700MHz 3GHz 之间。

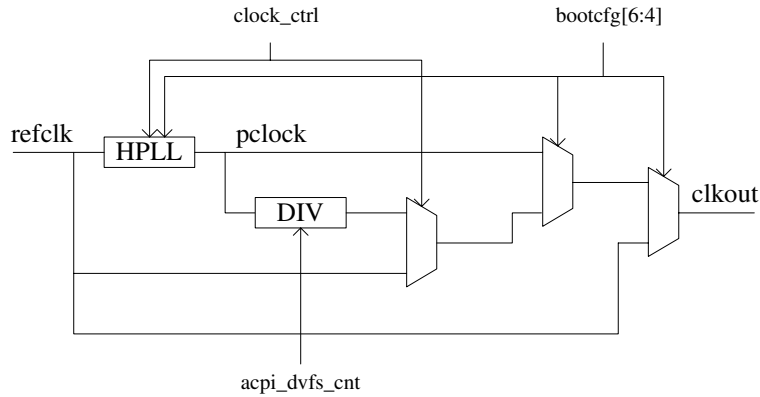


图 3.3: CPU 时钟结构

3.3 DDR 时钟

DDR 时钟有硬件跳线和软件控制两种配置方式。硬件跳线使用 NAND_D[5:3] 作为配置信号 (bootcfg[9:7])，只能有少数几种频率选择。硬件配置模式下，参考时钟必须为 100MHz。软件控制方式可以对所有的参数进行控制，通过 clock_ctrl0 和 clock_ctrl1 寄存器进行配置，其定义见表5.30和表5.31。DDR 时钟由展频 PLL(SPLL) 产生，其倍频后频率输出应当在 600MHz 1.8GHz 之间。

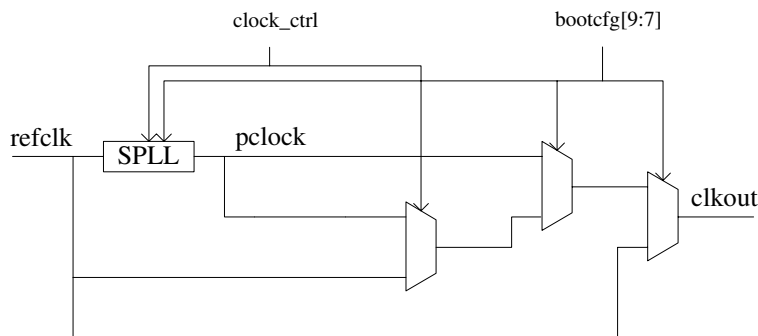


图 3.4: DDR 时钟结构

3.4 SYS 时钟

龙芯 2H 多数片上模块的工作时钟由 SYS 时钟复位发生器产生，包括 GMAC 的 125MHz 时钟，HDA 的 24MHz 时钟，GPU、HDV 以及 PCIe 参考时钟等等。SYS_RCG

由一个高频 PLL 和一组分频器组成，如图 3.5 所示。SYS_RCG 有硬件跳线和软件控制两种配置方式。硬件跳线使用 NAND_D[7:6] 作为配置信号 (bootcfg[11:10])，只有一种固定频率：HPLL 输出 3GHz 时钟，分频器采用默认分频（见图3.5）。硬件配置模式下，参考时钟必须为 100MHz。软件控制方式可以对所有的参数进行控制，通过 sys_pllcfg 寄存器配置 HPLL，通过 clock_ctrl2/3 配置分频器，其定义见表5.32和表5.33。默认分频见图3.5。

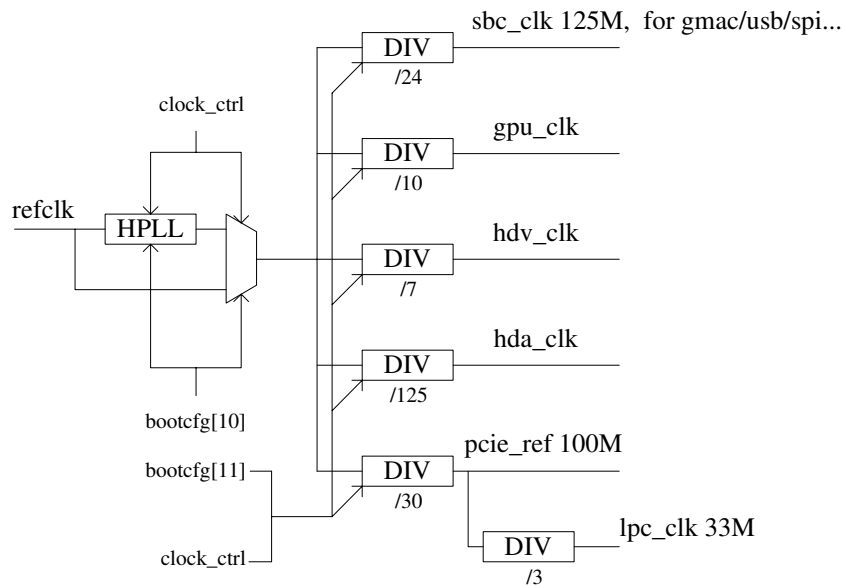


图 3.5: SYS 时钟结构

3.5 显示时钟

显示时钟由 SPLL 产生，需要根据目标显示模式进行设置。其结构如图3.6所示，由软件通过 pix*_pllcfg 寄存器进行配置，其定义见表5.34和表5.35。为保证时钟正确，软件应当先切换到 refclk，然后按选参考时钟—预分频器—SPLL—后分频器的次序配置时钟通路，最后将输出切换到后分频器输出。

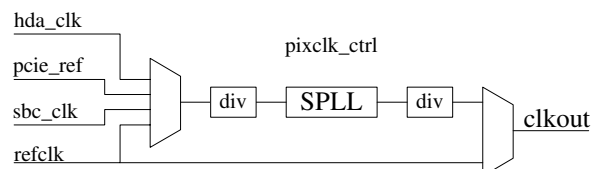


图 3.6: 显示时钟结构

3.6 时钟信号说明

表3.1统一介绍了龙芯 2H 的所有时钟引脚。

表 3.1: 龙芯 2H 时钟信号

信号名称	频率 (MHz)	类型	描述
SATA_XTAL[2:1]	25	I/O	SATA 参考时钟晶体 如果没有选择 BAKCLK，则该 25M 晶体必须连接。
RTC_XI RTC_XO	32.768K	I/O	RTC 参考时钟晶体
USB_XI[1:0] USB_XO[1:0]	12	I/O	USB 参考时钟晶体 根据软件配置，第 2 个晶体可不接。
LPC_CLKIN	33	I	LPC 时钟输入
EJTAG_TCK	33	I	JTAG 时钟
AC97_BITCLK	12.288 24	I O	AC97 时钟输入 HDA 时钟输出
SYS_BAKCLK	100	I	100MHz 时钟
SYS_TESTCLK	-	I	测试时钟，不需要连接
GMAC[1:0]_TXCK	125/25/2.5	O	RGMII 发送时钟
GMAC[1:0]_RXCK	125/25/2.5	I	RGMII 接收时钟
LPC_CLK[1:0]	33	O	相位相同的两路 33MHz 时钟输出，可为两个设备 (含 2H) 的 LPC 总线提供时钟。
PCIE_XTAL[2:1]	100	DIFF IN	PCIE 参考时钟输入 如果使用内部 100M 时钟，则可悬空。
HT_TX_CLKp[0] HT_TX_CLKn[0]	200/800	DIFF OUT	HT 总线发送时钟
HT_RX_CLKp[0] HT_RX_CLKn[0]	200/800	DIFF IN	HT 总线接收时钟
HTCLKp HTCLKn	200	DIFF OUT	HT 总线参考时钟
PCIE[3:0]_CLKp PCIE[3:0]_CLKn	100	DIFF OUT	PCIE 参考时钟输出
DDR_CKp[5:0] DDR_CKn[5:0]	400	DIFF OUT	DDR2/3 SDRAM 差分时钟输出 1,3,5为一组 DIMM 时钟， 0,2,4为另一组 DIMM 时钟

第四章 电源管理

本章对电源管理的大的方面进行介绍，寄存器和使用方法放到第二十四章

4.1 电源管理模块介绍

- 龙芯 2H 电源管理模块提供系统功耗管理实现机制。
- 支持 Advanced Configuration and Power Interface, Version 4.0a(ACPI), 提供相应的功耗管理功能。
- 系统休眠与唤醒，支持 ACPI S3（待机到内存），ACPI S4（待机到硬盘），ACPI S5（软关机），并且支持电源失效检测和自动系统恢复。支持多种唤醒方式（USB，GMAC，电源开关等）
- 动态性能功耗控制，支持处理器核 DVFS 控制，支持动态关闭媒体解码协处理器电源。
- 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 ACPI 状态及其相关说明。

表4.2显示了处理器相关状态 C_x，处理器运行时状态表示为 ACPI 的 P 状态，P 状态由软件决定，P_x 对应于每一组电压和频率值。

媒体处理器状态 D_x 如表4.3所示：

表 4.1: ACPI 状态说明

状态	描述
G0/S0	全部工作，该模式下系统全部工作，处理器子状态可由 C _x 或 P _x 决定，媒体处理器由状态 D _x 决定
G1/S1	暂不支持
G1/S3	Suspend to RAM(STR)，上下文保存到内存
G1/S4	Suspend to Disk(STD)，保存到硬盘，除唤醒电路全部掉电
G2/S5	Soft off，只有唤醒电路上电
G3	Mechanical off，所有供电失效

表 4.2: 处理器 Cx 状态

状态	描述
C0	Full on
C2	Stop Grant. 处理器时钟停止
C3	Deep Sleep. 处理器时钟停止, 处理器工作电压降低, 恢复时间较长, 电压由软件控制

表 4.3: 媒体处理器状态 Dx

状态	描述
D0	媒体处理器工作, 其性能功耗可以通过工作频率调节
D2	媒体处理器时钟关断
D3	媒体处理器电源关闭

4.3 控制引脚说明

表 4.4 为电源管理部分的 IO 信号描述。

表 4.4: 控制引脚说明

名称	类型	描述	供电
SYSRESETn	I	系统复位	RSM3V3
RSMRSTn	I	复位 Resume 域逻辑, 该信号需在 resume 域上电稳定后保持一段时间有效 (推荐: 5ms)	RTC2V5
RTCSTn	I	电池更换后, 重启 RTC 逻辑	RTC2V5
PLTRSTn	O	对系统平台其它设备进行复位	RSM3V3
PWROK	I	主供电电源上电稳定, 如有多个供电, 该信号表示最后一个电源稳定	RSM3V3
PWRBTNn	I	电源按钮	RSM3V3
RINGn	I	Modem 唤醒信号	RSM3V3
PCIWAKEn	I	PCIE 边带唤醒信号	RSM3V3
BATLOWn	I	电池电量低	RSM3V3
LID	I	显示器开关信号	RSM3V3
PWRTYPE	I	识别电池供电和电源供电 1: AC Power 0: System Battery	RSM3V3
SUS_STATn	O	指示系统将要进入低功耗状态	RSM3V3
SLP_S3n	O	STR, 待机到内存指示信号	RSM3V3
SLP_S4n	O	STD, 待机到硬盘指示信号	RSM3V3
SLP_S5n	O	Soft off	RSM3V3
SLP_LANn	O	以太网 PHY 休眠指示信号	RSM3V3
VID[5:0]	O	电压值指示信号	RSM3V3
VCORE_SENSE	O/A	CPU 供电电源采样	VDD_CPU
GND_SENSE	O/A	CPU 供电地信号采样	VDD_CPU

第五章 芯片配置与控制

5.1 芯片工作模式

龙芯 2H 可以作为独立的 SoC 也可以作为 HT 或者 PCIE 的桥片使用。本节就对这几种模式分别进行介绍。

5.1.1 独立 SoC

龙芯 2H 有丰富外围接口，配合若干外围器件，可简便地搭建一个单芯片计算机系统。一个常见的应用如图5.1 所示。

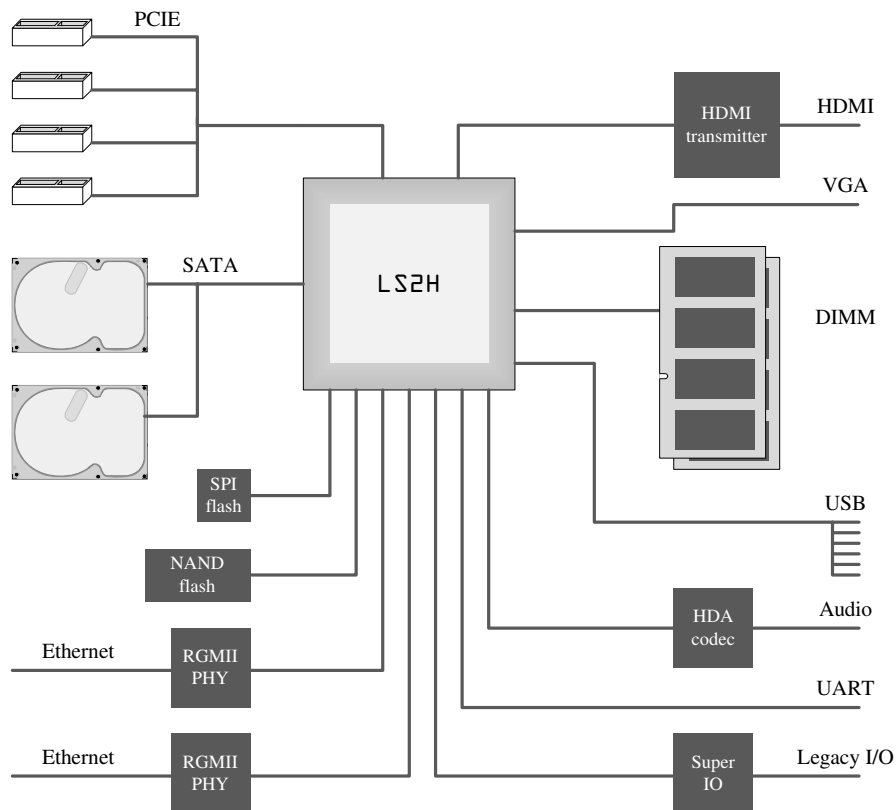


图 5.1: 龙芯 2H 单芯片系统

5.1.2 HT 桥片

龙芯 3 号多核处理器使用 HT 总线接口，可通过 2H 连接各种外设，形成两片系统。以龙芯 3A 为例，图5.2给出一个搭建的样例。

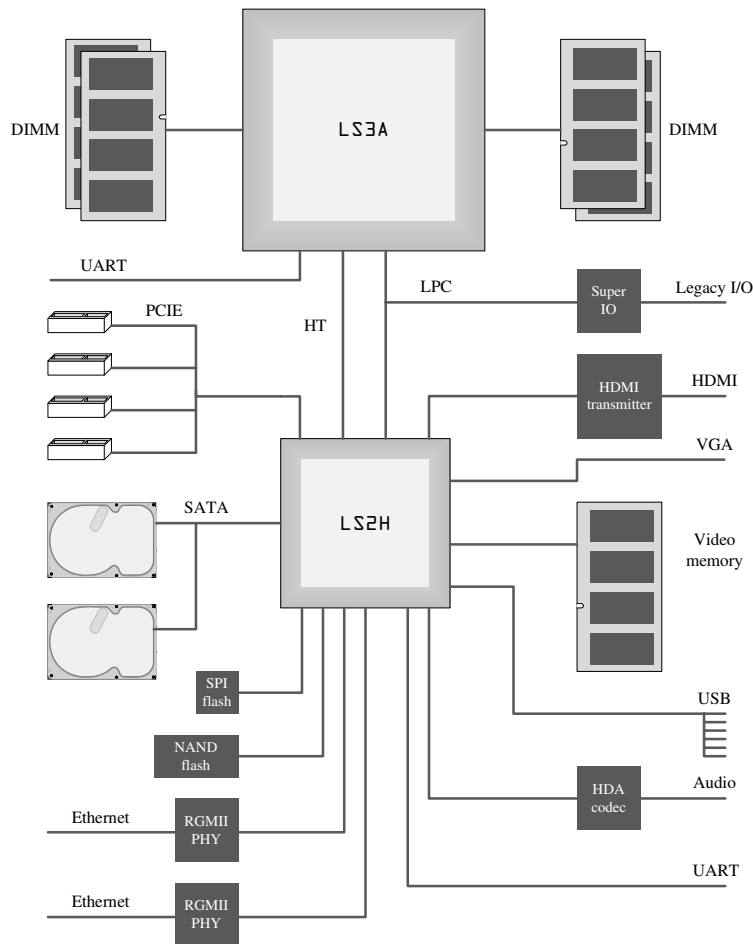


图 5.2: 龙芯 3A+2H 双芯片系统

为降低板级时钟成本，龙芯 3A 的一些参考时钟可以从龙芯 2H 得到，如 HT 和 PCI 时钟。如果要保留时钟的灵活性，这些时钟也可在板级实现。

为方便龙芯 3A 的 LPC 启动，龙芯 2H 的 LPC 接口可配置为 ROM 模式 (板级上拉 NAND_ALE)，支持 3A 的启动。LPC ROM 大小 2KB，只能装载 512 条初始化指令，后续指令应当从 HT 接口得到。需注意的是，3A 的系统复位应当连接 2H 的 LPC_RSTn，因为 2H 初始化 ROM 的内容需要一段时间，复位是软件可控制的。

在桥片模式下，龙芯 2H 的中断由 SYS_INTn 和 SYS_NMI_n 两引脚输出，送往 3A。龙芯 3A 需查询 2H 的中断控制器得到更详细的中断信息。

5.1.3 PCIE 桥片

龙芯 2H 的 PCIE 接口支持从模式 (endpoint)，实现桥片功能。该模式下，PCIE 只能作为 x4 使用，其系统连接情况与 HT 桥片类似，在此不再赘述。

5.2 芯片初始化信号

龙芯 2H 有两类初始化信号：一类从专用的配置引脚 (CLKSEL[3:0]) 输入，控制参考时钟；其它的复用功能引脚，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(见表 5.24)，供软件判定上电状态。

表 5.1: 配置信号

信号名称	bootcfg	描述
SYS_CLKSEL0	0	系统参考时钟选择 0: SATA_XTAL 1: SYS_BAKCLK
SYS_CLKSEL1	1	PCIE 参考时钟选择 0: 从 PCIE_XTAL 输入 1: 内部 100MHz 时钟
ACPI_CLKSEL2	2	USB PHY1 时钟选择 0: 内部时钟 1: 在 USB_XI1/XO1 上的 12MHz 晶体
ACPI_CLKSEL3	3	ACPI 使能 0: 不使用 ACPI 控制芯片 1: 使用 ACPI 控制
NAND_D1 NAND_D0	5:4	硬件控制模式下 CPU PLL 倍频配置 00: *4 01: *6 10: *8 11: bypass
NAND_D2	6	CPU PLL 纯硬件控制选择 0: 关闭, CPU PLL 频率由软件配置 1: 打开, CPU PLL 频率由 D1/D0 两位配置
NAND_D4 NAND_D3	8:7	硬件控制模式下 DDR PLL 倍频配置 00: *5/3 01: *8/3 10: *10/3 11: bypass
NAND_D5	9	DDR PLL 纯硬件控制选择 0: 关闭, DDR PLL 频率由软件配置 1: 打开, DDR PLL 频率由 D4/D3 两位配置
NAND_D6	10	SYS PLL 纯硬件控制选择 0: 关闭, SYS PLL 频率由软件配置 1: 打开, SYS PLL 固定振荡在 3GHz
NAND_D7	11	分频器后备模式选择 0: 关闭, 分频器软件可配 1: 打开, 分频器使用固定分频
NAND_CLE	12	PCIE 从模式选择 0: 关闭, PCIE 为 1x4 或者 4x1 的 root complex 1: 打开, PCIE 为 x4 的 endpoint

信号名称	bootcfg	描述
NAND_ALE	13	LPC ROM 模式使能 0: 关闭, LPC 接口为主控模式 1: 打开, LPC 接口可接受 BOOT ROM 访问
NAND_WR	14	NAND 启动
NAND_RD	15	NAND 是否打开读校验 0: 关闭 1: 打开
SPL_SCK SPLCSn	17:16	启动源选择 00: NAND 启动 (不建议使用) ¹ 10: LPC 启动 x1: SPI 启动
SPLSDO	18	NAND 启动类型选择 {SPLSDO, NAND_WR} 00: <256Mbit 01: 512Mbit 10: 1Gbit 11: >2Gbit
ACPL_VID[5:0]	-	VID 初始值 用于初始化电源管理模块中 VID 寄存器, 使得复位结束后在没有软件干预的情况下 ACPL_VID 能输出有效的编码。

5.3 地址空间分配

龙芯 2H 的地址空间分为 CPU 和 DMA 两个视角: 所有 CPU 可访问的设备编址在 CPU 地址空间上, 谓之 CPU 视角; 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。表5.2和表5.3分别给出了这两个视角的具体定义。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。

CPU 可发生多种访问类型, 包括字节 (B)、半字 (H)、字 (W)、双字 (D)、四字 (Q) 和块式 (C) 等。每个设备所支持的访问类型有限制, 如果超出其范围同样会导致不可预知的后果。

表 5.2: 地址空间分配之 CPU 视角

地址空间	模块	说明	访问
0x0000,0000 - 0x0fff,fff	DDR	256MB	BHWDQC
0x1000,0000 - 0x17ff,fff	PCIE	PCIE MEM	BHW
0x1800,0000 - 0x18ff,fff	PCIE	PCIE IO PCIE CFG PCIE MEM-Lo	B BHW BHW
0x1c00,0000 - 0x1dff,fff	LPC	LPC MEM	BHWDQC

¹由于 SPLSDO 复位时方向为输出, 且输出值不确定, NAND 启动时类型可能会出错, 从而使启动失败。如必须从 NAND 启动, 可使用 GPIO 控制一个可关断的强上拉或者强下拉驱动 SPLSDO 引脚。

地址空间	模块	说明	访问
0x1e00,0000 - 0x1eff,ffff	SPI	SPI Flash	BHWDQC
0x1fc0,0000 - 0x1fcf,ffff	Boot	spi nand lpc	BHWDQC W BHWDQC
0x1fd0,0000 - 0x1fd7,ffff	creg	中断与全局配置	BHW
0x1fd8,0000 - 0x1fdf,ffff	cbus	片上配置寄存器	BHWD
0x1fe0,0000 - 0x1fe0,ffff	USB	32KB EHCI 32KB OHCI	W
0x1fe1,0000 - 0x1fe1,ffff	GMAC	32KB GMAC0 32KB GMAC1	W
0x1fe2,0000 - 0x1fe2,ffff	HDA		BHW
0x1fe3,0000 - 0x1fe3,ffff	SATA		W
0x1fe4,0000 - 0x1fe4,ffff	GPU		W
0x1fe5,0000 - 0x1fe5,ffff	DC		W
0x1fe6,0000 - 0x1fe6,ffff	OTG		W
0x1fe7,0000 - 0x1fe7,ffff	SPI	SPI regs	B
0x1fe8,0000 - 0x1fe8,ffff	uart		B
0x1fe9,0000 - 0x1fe9,ffff	i2c		W
0x1fea,0000 - 0x1fea,ffff	pwm		W
0x1fec,0000 - 0x1fec,ffff	hpet		W
0x1fed,0000 - 0x1fed,ffff	ac97		W
0x1fee,0000 - 0x1fee,ffff	nand		W
0x1fef,0000 - 0x1fef,7fff	acpi		W
0x1fef,8000 - 0x1fef,ffff	rtc		W
0x1ff0,0000 - 0x1ff0,ffff	LPC	LPC IO	B
0x1ff1,0000 - 0x1ff1,00ff	LPC	LPC regs	W
0x4000,0000 - 0x7fff,ffff	PCIE	PCIE MEM-Hi	BHW
0x8000,0000 - 0xffff,ffff	DDR	2GB	BHWDQC

龙芯 2H 内部可发起 DMA 的主设备包括 GPU、DC、PCIE、MEDIA、GMAC、USB、SATA、HDA、DMA 等。除了 GPU、DC 只能访问 DDR 之外，其它主设备的请求可以访问路由到表 5.3 中的所有空间。

通过 HT 实现桥片功能前，HT 总线需要进行若干配置，详见第七章。

表 5.3: 地址空间分配之 DMA 视角

地址空间	目标	说明
0x0000,0000 - 0x3fff,ffff	DDR	不维护 IO 一致性
0x4000,0000 - 0x7fff,ffff	DDR	维护 IO 一致性
0x8000,0000 - 0xbfff,ffff	HT	跨 HT 总线送到 3A 系统内存

5.4 时钟与复位控制

5.4.1 时钟配置概要

龙芯 2H 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. 进入关断模式，*_pd 置位
2. 修改 PLL 参数 (idf/ldf/odf etc.)
3. 满足最短持续时间后离开关断模式，*_pd 清零
4. 等待锁定信号 (*_lock)
5. 切换时钟 (*_sel)

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。

有些时钟通路中还包含软件可配置的分频器，应按以下流程进行配置：

1. 确认 *_set 为零
2. 设置分频系数 *_df 和关断模式 (*_pd 置 1)
3. 设置 *_set 为 1，然后清零
4. 退出关断模式 (*_pd 置 0)
5. 设置 *_set 为 1，然后清零

当输入时钟停止时分频器无法设置，所以应当从参考时钟开始对时钟通路中的模块逐个进行配置（比如显示时钟，详见 3.5 节）。

5.4.2 高频 PLL 配置

高频 PLL 概念性结构如图5.3所示，输入的参考时钟 Fref 送到倍频器，得到 Fvco，然后在输出前除以一个分频系数。

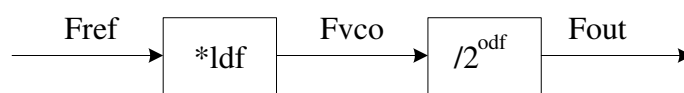


图 5.3: 高频 PLL 概念性结构

该 PLL 的频率转换公式为 $F_{out} = F_{ref} * ldf / 2^{odf}$ ，其工作条件见表5.4。

表 5.4: 高频 PLL 工作条件

参数	最小值	最大值	说明
ldf	7	127	
odf	0	7	
Fref	25MHz	200MHz	
Fvco	700MHz	3000MHz	
Fout	-	3000MHz	
Tpd	10us	-	关断持续时间
Tlock	-	200us	锁定时间

5.4.3 展频 PLL 配置

展频 PLL 概念性结构如图5.4所示，输入时钟 F_{in} 经过输入分频器得到 F_{ref} 送到倍频器，倍频器送出 F_{vco} ，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。

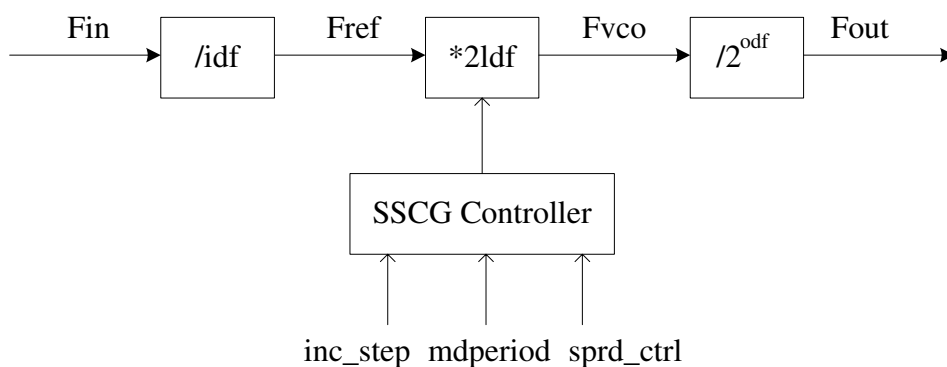


图 5.4: 展频 PLL 概念性结构

该 PLL 的频率转换公式为 $F_{out} = F_{in}/ldf * 2 * ldf / 2^{odf}$ ，其工作条件见表5.5。

展频配置所涉及的参数计算方法如下

$$mdperiod = round\left(\frac{F_{ref}}{4 * F_{mod}}\right)$$

$$inc_step = round\left(\frac{65534 * md * ldf}{500 * mdperiod}\right)$$

5.4.4 复位控制

龙芯 2H 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/HDV/GPU 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** 由看门狗或者电源管理模块发起，详见相关章节。

表 5.5: 展频 PLL 工作条件

参数	最小值	最大值	说明
idf	1	7	
ldf	8	225	
odf	0	3	
Fin	4MHz	525MHz	
Fref	4MHz	75MHz	
Fvco	600MHz	1800MHz	
Fout	75MHz	1800MHz	
Tpd	10us	-	关断持续时间
Tlock	-	100us	锁定时间
md center-spread down-spread	+/-0.25% -0.5%	+/-2.0% -4.0%	调制深度, 根据展频模式有所不同
Fmod 16<LDF<148 150<LDF<450 inc_step * mdperiod		3700/LDF KHz 25KHz 2 ¹⁵ -1	调制频率 两输入参数的积

5.5 芯片配置寄存器

龙芯 2H 有大量的配置寄存器, 多数分布于各个功能模块中, 本节介绍芯片级的配置寄存器。

表 5.6: 芯片配置寄存器列表

地址	名称	描述
0x1fd0,0000	msiport	MSI 中断写端口
0x1fd0,0040	intisr0	中断状态
0x1fd0,0044	intien0	中断使能
0x1fd0,0048	intset0	中断设置 (边沿模式)
0x1fd0,004c	intclr0	中断清除 (边沿模式)
0x1fd0,0050	intpol0	中断有效极性
0x1fd0,0054	intedge0	边沿模式
0x1fd0,0058	intisr1	中断状态
0x1fd0,005c	intien1	中断使能
0x1fd0,0060	intset1	中断设置 (边沿模式)
0x1fd0,0064	intclr1	中断清除 (边沿模式)
0x1fd0,0068	intpol1	中断有效极性
0x1fd0,006c	intedge1	边沿模式
0x1fd0,0070	intisr2	中断状态
0x1fd0,0074	intien2	中断使能
0x1fd0,0078	intset2	中断设置 (边沿模式)
0x1fd0,007c	intclr2	中断清除 (边沿模式)
0x1fd0,0080	intpol2	中断有效极性
0x1fd0,0084	intedge2	边沿模式

地址	名称	描述
0x1fd0,0088	intisr3	中断状态
0x1fd0,008c	intien3	中断使能
0x1fd0,0090	intset3	中断设置 (边沿模式)
0x1fd0,0094	intclr3	中断清除 (边沿模式)
0x1fd0,0098	intpol3	中断有效极性
0x1fd0,009c	intedge3	边沿模式
0x1fd0,00a0	intisr4	中断状态
0x1fd0,00a4	intien4	中断使能
0x1fd0,00a8	intset4	中断设置 (边沿模式)
0x1fd0,00ac	intclr4	中断清除 (边沿模式)
0x1fd0,00b0	intpol4	中断有效极性
0x1fd0,00b4	intedge4	边沿模式
0x1fd0,00c0	gpiocfg	GPIO 功能选择
0x1fd0,00c4	gpiooe	GPIO 输出使能
0x1fd0,00c8	gpioin	GPIO 输入
0x1fd0,00cc	gpioout	GPIO 输出
0x1fd0,00d0	-	保留
0x1fd0,00d4	-	保留
0x1fd0,00d8	chip_sample4	芯片采样
0x1fd0,00dc	-	保留
0x1fd0,00e0	-	保留
0x1fd0,00e4	-	保留
0x1fd0,00e8	chip_sample5	芯片采样
0x1fd0,00ec	-	保留
0x1fd0,0100	dma_order	DMA 命令寄存器
0x1fd0,0180	hdv_scratch0	保留
0x1fd0,0184	hdv_scratch1	保留
0x1fd0,01fc	key	保留
0x1fd0,0200	chip_config0	芯片配置
0x1fd0,0204	chip_config1	芯片配置
0x1fd0,0208	chip_config2	芯片配置
0x1fd0,020c	chip_config3	芯片配置
0x1fd0,0210	chip_sample0	芯片采样
0x1fd0,0214	chip_sample1	芯片采样
0x1fd0,0218	chip_sample2	芯片采样
0x1fd0,021c	chip_sample3	芯片采样
0x1fd0,0220	clock_ctrl0	时钟控制
0x1fd0,0224	clock_ctrl1	时钟控制
0x1fd0,0228	clock_ctrl2	时钟控制
0x1fd0,022c	clock_ctrl3	时钟控制
0x1fd0,0230	pixclk0_ctrl0	显示时钟控制
0x1fd0,0234	pixclk0_ctrl1	显示时钟控制
0x1fd0,0238	pixclk1_ctrl0	显示时钟控制
0x1fd0,023c	pixclk1_ctrl1	显示时钟控制

地址	名称	描述
0x1fd00240	chipid0	芯片编号 0
0x1fd00244	chipid1	芯片编号 1
0x1fd00248	chipid2	芯片编号 2
0x1fd0024c	chipid3	芯片编号 3 该寄存器读出后 chipid[0..3] 将变为 0
0x1fd8,0000	m0_win0_base	保留
0x1fd8,0008	m0_win1_base	保留
0x1fd8,0010	m0_win2_base	保留
0x1fd8,0018	m0_win3_base	保留
0x1fd8,0020	m0_win4_base	保留
0x1fd8,0028	m0_win5_base	保留
0x1fd8,0030	m0_win6_base	保留
0x1fd8,0038	m0_win7_base	保留
0x1fd8,0040	m0_win0_mask	保留
0x1fd8,0048	m0_win1_mask	保留
0x1fd8,0050	m0_win2_mask	保留
0x1fd8,0058	m0_win3_mask	保留
0x1fd8,0060	m0_win4_mask	保留
0x1fd8,0068	m0_win5_mask	保留
0x1fd8,0070	m0_win6_mask	保留
0x1fd8,0078	m0_win7_mask	保留
0x1fd8,0080	m0_win0_mmap	保留
0x1fd8,0088	m0_win1_mmap	保留
0x1fd8,0090	m0_win2_mmap	保留
0x1fd8,0098	m0_win3_mmap	保留
0x1fd8,00a0	m0_win4_mmap	保留
0x1fd8,00a8	m0_win5_mmap	保留
0x1fd8,00b0	m0_win6_mmap	保留
0x1fd8,00b8	m0_win7_mmap	保留
0x1fd8,0100	m1_win0_base	保留
0x1fd8,0108	m1_win1_base	保留
0x1fd8,0110	m1_win2_base	保留
0x1fd8,0118	m1_win3_base	保留
0x1fd8,0120	m1_win4_base	保留
0x1fd8,0128	m1_win5_base	保留
0x1fd8,0130	m1_win6_base	保留
0x1fd8,0138	m1_win7_base	保留
0x1fd8,0140	m1_win0_mask	保留
0x1fd8,0148	m1_win1_mask	保留
0x1fd8,0150	m1_win2_mask	保留
0x1fd8,0158	m1_win3_mask	保留
0x1fd8,0160	m1_win4_mask	保留
0x1fd8,0168	m1_win5_mask	保留
0x1fd8,0170	m1_win6_mask	保留

地址	名称	描述
0x1fd8,0178	m1_win7_mask	保留
0x1fd8,0180	m1_win0_mmap	保留
0x1fd8,0188	m1_win1_mmap	保留
0x1fd8,0190	m1_win2_mmap	保留
0x1fd8,0198	m1_win3_mmap	保留
0x1fd8,01a0	m1_win4_mmap	保留
0x1fd8,01a8	m1_win5_mmap	保留
0x1fd8,01b0	m1_win6_mmap	保留
0x1fd8,01b8	m1_win7_mmap	保留
0x1fd8,0200	m2_win0_base	保留
0x1fd8,0208	m2_win1_base	保留
0x1fd8,0210	m2_win2_base	保留
0x1fd8,0218	m2_win3_base	保留
0x1fd8,0220	m2_win4_base	保留
0x1fd8,0228	m2_win5_base	保留
0x1fd8,0230	m2_win6_base	保留
0x1fd8,0238	m2_win7_base	保留
0x1fd8,0240	m2_win0_mask	保留
0x1fd8,0248	m2_win1_mask	保留
0x1fd8,0250	m2_win2_mask	保留
0x1fd8,0258	m2_win3_mask	保留
0x1fd8,0260	m2_win4_mask	保留
0x1fd8,0268	m2_win5_mask	保留
0x1fd8,0270	m2_win6_mask	保留
0x1fd8,0278	m2_win7_mask	保留
0x1fd8,0280	m2_win0_mmap	保留
0x1fd8,0288	m2_win1_mmap	保留
0x1fd8,0290	m2_win2_mmap	保留
0x1fd8,0298	m2_win3_mmap	保留
0x1fd8,02a0	m2_win4_mmap	保留
0x1fd8,02a8	m2_win5_mmap	保留
0x1fd8,02b0	m2_win6_mmap	保留
0x1fd8,02b8	m2_win7_mmap	保留
0x1fd8,0300	m3_win0_base	保留
0x1fd8,0308	m3_win1_base	保留
0x1fd8,0310	m3_win2_base	保留
0x1fd8,0318	m3_win3_base	保留
0x1fd8,0320	m3_win4_base	保留
0x1fd8,0328	m3_win5_base	保留
0x1fd8,0330	m3_win6_base	保留
0x1fd8,0338	m3_win7_base	保留
0x1fd8,0340	m3_win0_mask	保留
0x1fd8,0348	m3_win1_mask	保留
0x1fd8,0350	m3_win2_mask	保留

地址	名称	描述
0x1fd8,0358	m3_win3_mask	保留
0x1fd8,0360	m3_win4_mask	保留
0x1fd8,0368	m3_win5_mask	保留
0x1fd8,0370	m3_win6_mask	保留
0x1fd8,0378	m3_win7_mask	保留
0x1fd8,0380	m3_win0_mmap	保留
0x1fd8,0388	m3_win1_mmap	保留
0x1fd8,0390	m3_win2_mmap	保留
0x1fd8,0398	m3_win3_mmap	保留
0x1fd8,03a0	m3_win4_mmap	保留
0x1fd8,03a8	m3_win5_mmap	保留
0x1fd8,03b0	m3_win6_mmap	保留
0x1fd8,03b8	m3_win7_mmap	保留
0x1fd8,0400	m4_win0_base	保留
0x1fd8,0408	m4_win1_base	保留
0x1fd8,0410	m4_win2_base	保留
0x1fd8,0418	m4_win3_base	保留
0x1fd8,0420	m4_win4_base	保留
0x1fd8,0428	m4_win5_base	保留
0x1fd8,0430	m4_win6_base	保留
0x1fd8,0438	m4_win7_base	保留
0x1fd8,0440	m4_win0_mask	保留
0x1fd8,0448	m4_win1_mask	保留
0x1fd8,0450	m4_win2_mask	保留
0x1fd8,0458	m4_win3_mask	保留
0x1fd8,0460	m4_win4_mask	保留
0x1fd8,0468	m4_win5_mask	保留
0x1fd8,0470	m4_win6_mask	保留
0x1fd8,0478	m4_win7_mask	保留
0x1fd8,0480	m4_win0_mmap	保留
0x1fd8,0488	m4_win1_mmap	保留
0x1fd8,0490	m4_win2_mmap	保留
0x1fd8,0498	m4_win3_mmap	保留
0x1fd8,04a0	m4_win4_mmap	保留
0x1fd8,04a8	m4_win5_mmap	保留
0x1fd8,04b0	m4_win6_mmap	保留
0x1fd8,04b8	m4_win7_mmap	保留
0x1fd8,0500	m5_win0_base	保留
0x1fd8,0508	m5_win1_base	保留
0x1fd8,0510	m5_win2_base	保留
0x1fd8,0518	m5_win3_base	保留
0x1fd8,0520	m5_win4_base	保留
0x1fd8,0528	m5_win5_base	保留
0x1fd8,0530	m5_win6_base	保留

地址	名称	描述
0x1fd8,0538	m5_win7_base	保留
0x1fd8,0540	m5_win0_mask	保留
0x1fd8,0548	m5_win1_mask	保留
0x1fd8,0550	m5_win2_mask	保留
0x1fd8,0558	m5_win3_mask	保留
0x1fd8,0560	m5_win4_mask	保留
0x1fd8,0568	m5_win5_mask	保留
0x1fd8,0570	m5_win6_mask	保留
0x1fd8,0578	m5_win7_mask	保留
0x1fd8,0580	m5_win0_mmap	保留
0x1fd8,0588	m5_win1_mmap	保留
0x1fd8,0590	m5_win2_mmap	保留
0x1fd8,0598	m5_win3_mmap	保留
0x1fd8,05a0	m5_win4_mmap	保留
0x1fd8,05a8	m5_win5_mmap	保留
0x1fd8,05b0	m5_win6_mmap	保留
0x1fd8,05b8	m5_win7_mmap	保留
0x1fd8,0600	qos_cfg0	保留
0x1fd8,0608	qos_cfg1	保留
0x1fd8,0610	qos_cfg2	保留
0x1fd8,0618	qos_cfg3	保留
0x1fd8,0620	qos_cfg4	保留
0x1fd8,0628	qos_cfg5	保留
0x1fd8,0630	qos_cfg6	保留
0x1fd8,0638	qos_cfg7	保留
0x1fd8,0660	hdv_cmdstat	保留
0x1fd8,0670	amon_cfg	保留
0x1fd8,0678	amon_pcnt	保留
0x1fd8,0680	amon_stat0	保留
0x1fd8,0688	amon_stat1	保留
0x1fd8,0690	amon_stat2	保留
0x1fd8,0698	amon_stat3	保留
0x1fd8,06a0	amon_stat4	保留
0x1fd8,06a8	amon_stat5	保留
0x1fd8,06b0	amon_stat6	保留
0x1fd8,06b8	amon_stat7	保留
0x1fd8,06c0	amon_stat8	保留
0x1fd8,06c8	amon_stat9	保留
0x1fd8,06d0	amon_stat10	保留
0x1fd8,06d8	amon_stat11	保留
0x1fd8,06e0	amon_stat12	保留
0x1fd8,06e8	amon_stat13	保留
0x1fd8,0700	ds_cpu	保留
0x1fd8,0720	ds_ddr	保留

地址	名称	描述
0x1fd8,0740	ds_sb	保留
0x1fd8,0800	usbphy0_cfg	USB PHY0 配置
0x1fd8,0808	usbphy1_cfg	USB PHY1 配置
0x1fd8,2600	m6_win0_base	保留
0x1fd8,2608	m6_win1_base	保留
0x1fd8,2610	m6_win2_base	保留
0x1fd8,2618	m6_win3_base	保留
0x1fd8,2620	m6_win4_base	保留
0x1fd8,2628	m6_win5_base	保留
0x1fd8,2630	m6_win6_base	保留
0x1fd8,2638	m6_win7_base	保留
0x1fd8,2640	m6_win0_mask	保留
0x1fd8,2648	m6_win1_mask	保留
0x1fd8,2650	m6_win2_mask	保留
0x1fd8,2658	m6_win3_mask	保留
0x1fd8,2660	m6_win4_mask	保留
0x1fd8,2668	m6_win5_mask	保留
0x1fd8,2670	m6_win6_mask	保留
0x1fd8,2678	m6_win7_mask	保留
0x1fd8,2680	m6_win0_mmap	保留
0x1fd8,2688	m6_win1_mmap	保留
0x1fd8,2690	m6_win2_mmap	保留
0x1fd8,2698	m6_win3_mmap	保留
0x1fd8,26a0	m6_win4_mmap	保留
0x1fd8,26a8	m6_win5_mmap	保留
0x1fd8,26b0	m6_win6_mmap	保留
0x1fd8,26b8	m6_win7_mmap	保留
0x1fd8,4200	sc_lock0_addr	二级缓存锁地址
0x1fd8,4208	sc_lock1_addr	二级缓存锁地址
0x1fd8,4210	sc_lock2_addr	二级缓存锁地址
0x1fd8,4218	sc_lock3_addr	二级缓存锁地址
0x1fd8,4240	sc_lock0_mask	二级缓存锁掩码
0x1fd8,4248	sc_lock1_mask	二级缓存锁掩码
0x1fd8,4250	sc_lock2_mask	二级缓存锁掩码
0x1fd8,4258	sc_lock3_mask	二级缓存锁掩码

5.5.1 消息中断写端口(msiport)

MSI 中断的写端口，可用于提供 PCIe 的 MSI 中断。

表 5.7: 消息中断写端口

位域	名称	访问	初值	描述
4:0	int_vec	W	-	中断向量的编号。写入的值将译码为 32 位中断设置信号，送往第 3 组中断控制器

5.5.2 中断状态寄存器(intisr)

中断状态寄存器。

表 5.8: 中断状态寄存器

位域	名称	访问	初值	描述
0~31	intisr[i]	R	-	中断状态, 1 为有中断

5.5.3 中断使能寄存器(intien)

中断使能寄存器。

表 5.9: 中断使能寄存器

位域	名称	访问	初值	描述
0~31	intien[i]	R/W	0	中断使能, 1 为使能

5.5.4 中断设置寄存器(intset)

在边沿触发的中断模式下, 往 intset 位写 1 可以将置起该位所对应的中断状态。

表 5.10: 中断设置寄存器

位域	名称	访问	初值	描述
0~31	intset[i]	W	-	中断设置, 写 1 为置起中断

5.5.5 中断清除寄存器(intclr)

在边沿触发的中断模式下, 往 intclr 位写 1 可以将清除该位所对应的中断状态。

表 5.11: 中断清除寄存器

位域	名称	访问	初值	描述
0~31	intclr[i]	W	-	中断清除, 写 1 为清除中断

5.5.6 中断极性寄存器(intpol)

用于设置中断线的有效极性。

表 5.12: 中断极性寄存器

位域	名称	访问	初值	描述
0~31	intpol[i]	R/W	0	电平有效模式下, 1 为高电平; 边沿有效模式下, 1 为上升沿

5.5.7 中断模式寄存器(intedge)

用于设置中断线的触发模式。

表 5.13: 中断模式寄存器

位域	名称	访问	初值	描述
0~31	intedge[i]	R/W	0	0 为电平触发, 1 为边沿触发

5.5.8 通用输入输出配置(gpiocfg)

gpiocfg 用于选择 GPIO 引脚的功能。

表 5.14: 通用输入输出配置

位域	名称	访问	初值	描述
18	pwm3_dir	R/W	0	复用 GPIO 的 PWM3 引脚方向控制 0: 输出 1: 输入
17	pwm2_dir	R/W	0	复用 GPIO 的 PWM2 引脚方向控制 0: 输出 1: 输入
16	gpio_led	R/W	0	SATA 状态指示灯复用控制 0: 两个 led 灯分别对应两个 SATA 1: 使能 gpio 点灯, 两 SATA 状态汇集后由 SATA0_LEDn 输出, 由 gpio 控制 SATA1_LEDn 的输出。
15:0	gpiocfg[i]	R/W	0	对应 16 个 GPIO 引脚。置 1 则选择 GPIO 引脚的第二功能。

5.5.9 通用输入输出方向(gpiooe)

gpiooe 用于控制 GPIO 的方向。

表 5.15: 通用输入输出方向

位域	名称	访问	初值	描述
0~15	gpiooe[i]	R/W	1	0 为输出, 1 为输入

5.5.10 通用输入采样(gpioin)

GPIO 引脚的值由 gpioin 寄存器读出。

表 5.16: 通用输入采样

位域	名称	访问	初值	描述
0~15	gpioin[i]	R	-	直接反映 GPIO 引脚的值

5.5.11 通用输出设置(gpioout)

方向为输出的 GPIO 引脚的值由 gpioout 寄存器控制。

表 5.17: 通用输出设置

位域	名称	访问	初值	描述
0~15	gpioout[i]	R/W	0	0 输出低电平, 1 输出高电平

位域	名称	访问	初值	描述
16	sata1_led	R/W	0	gpio_led 为高时控制 SATA1_LEDn 0: 高阻 1: 驱动为低

5.5.12 DMA 命令控制寄存器(dma_order)

DMA 控制器的命令控制寄存器，存放。

表 5.18: DMA 命令控制寄存器

位域	名称	访问	初值	描述
31:6	ask_addr	R/W	0	32 位地址的高 26 位
5	-	-	0	保留
4	dma_stop	R/W	0	停止 DMA 操作。DMA 控制器完成当前数据读写后停止。
3	dma_start	R/W	0	开始 DMA 操作。DMA 控制器读取描述符地址 (ask_addr) 后将些位清零。
2	ask_valid	R/W	0	DMA 工作寄存器写回到 (ask_addr) 所指向的内存，完成后清零。
1:0	dev_num	R/W	0	DMA 设备选择 00: NAND 01: AC97 record 10: AC97 replay

5.5.13 芯片配置寄存器 0(chip_config0)

芯片配置寄存器提供一般性的全局配置。

表 5.19: 芯片配置寄存器 0

位域	名称	访问	初值	描述
31:30	-	-	0	保留
29	hdv_rstn	R/W	0	媒体解码器复位，低有效
28	-	-	0	保留
27	gpu_rst	R/W	0	GPU 复位，高有效
26	usb_rstn	R/W	0	USB 复位，低有效。访问 USB 寄存器前须置位。
25	lpc_rst	R/W	0/1	LPC 总线复位，高有效。在桥片模式下复位值为 1，否则为 0。
24	thsens_pdn	R/W	0	温度传感器关断，低有效。在使用温度传感器前需置位。
23	-	-	0	保留
22	dc_shut	R/W	0	DC 时钟关断，高有效
21	otg_shut	R/W	0	OTG 时钟关断，高有效
20	gmac1_shut	R/W	0	GMAC1 时钟关断，高有效
19	gmac0_shut	R/W	0	GMAC0 时钟关断，高有效
18	sata_shut	R/W	0	SATA 时钟关断，高有效
17	hda_shut	R/W	0	HDA 时钟关断，高有效
16	usb_shut	R/W	0	USB 时钟关断，高有效
15	macptp_en	R/W	0	MAC 控制器 IEEE1588 时间计数器使能。该位控制一个 64 位时间计数器，在不使用时关闭有助于降低功耗。

位域	名称	访问	初值	描述
14	otg_sel	R/W	0	USB 端口 0 模式选择 0: Host 1: OTG, 端口 0 由 OTG 控制器接管 在初始化 USB/OTG 控制器前须配置好
13	ddrcnf_off	R/W	0	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
12	ht_addr_m	R/W	0	HT 地址过滤 置位后 3A 通过 HT 发出地址的 31 和 30 位将被抹零
11	ht_fscale	R/W	0	保留
10	ht_pll_byp	R/W	0	HT PLL 旁路选择, 高有效
9	ht_pll_pd	R/W	0	HT PLL 关断, 高有效 在不使用 HT 时建议置位。
8:7	ejtag_drop	R/W	0	ejtag 链选择 00: tdi→LS464 →tdo 10: tdi→LS464 →HDV →tdo 11: tdi→HDV →tdo
6	set_nmi	R/W	0	桥片模式下 NMI 输出控制, 高为置位 (NMI _n 引脚输出低电平)
5	br_mode	R/W	0	桥片模式使能, 高有效 桥片模式下 INT _n 和 NMI _n 引脚方向
4	hda_sel	R/W	0	AC97/HDA 模式选择 0: AC97 模式 1: HDA 模式
3	hda_clk_sel	R/W	0	HDA 比特时钟选择 0: SYS_RCG 分频器输出 1: USB 时钟分频
2	ddr32to16_en	R/W	0	DDR 总线 16 位模式选择 0: 64/32 位 1: 16 位 将此位配置为 1 的前提是 DDR 控制器工作在 32 位模式下, 且完成初始化, 不再访问 DDR 配置空间
1:0	uart_split	R/W	0	8 线串口复用选择, 见表 5.20

表 5.20: 串口复用配置

uart_split	TXD	RTS	DTR	RI	RXD	CTS	DSR	DCD
00	TXD0	RTS0	DTR0	RI0	RXD0	CTS0	DSR0	DCD0
01	TXD0	RTS0	TXD1	RTS1	RXD0	CTS0	RXD1	CTS1
10	TXD0	RTS0	TXD1	TXD2	RXD0	CTS0	RXD1	RXD2
11	TXD0	TXD1	TXD2	TXD3	RXD0	RXD1	RXD2	RXD3

5.5.14 芯片配置寄存器 1(chip_config1)

芯片配置寄存器提供一般性的全局配置。

表 5.21: 芯片配置寄存器 1

位域	名称	访问	初值	描述
31:24	phy_ctrl	-	0	保留
23:20	-	-	0	保留
19	pe_pwrfail	R/W	0	PCIe 电源失效
18	pe_sleep	R/W	0	PCIe 睡眠控制
17	pe_wake	R/W	0	PCIe 唤醒控制
16	pe_retryn	R/W	0	PCIe 请求重试
15	gpu_clkdiv	R/W	0	GPU 接口时钟二分频 0: 与 DDR 时钟同频 1: DDR 时钟的二分频
14	gpu_ramcgn	R/W	0	GPU RAM 时钟门控使能 0: 允许 GPU RAM 的时钟门控 1: 禁止 GPU RAM 的时钟门控
13:8	ccio_mask	R/W	0	检测一致性访存的使能位 如果需要使用 ACPI 的 C2/C3 BML 唤醒功能, 此域应置为 0x3f。
7:2	-	-	0	保留
1	dreset	R/W	0	软件全系统复位 除芯片配置、时钟控制寄存器以及 RSM/RTC 电压域外, 所有的逻辑都将被复位。
0	-	-	0	保留

5.5.15 芯片配置寄存器 2(chip_config2)

芯片配置寄存器 2 主要提供 HT IO 的特性配置。

表 5.22: 芯片配置寄存器 2

位域	名称	访问	初值	描述
31:26	-	-	0	保留
25:19	ht_asrc	R/W	0	
18	ht_freeze	R/W	0	
17	ht_comptq	R/W	0	
16	ht_compen	R/W	0	
15:0	scratch	R/W	0	保存 16 位用户自定义的信息, 可用于在热复位时保存状态。

5.5.16 芯片配置寄存器 3(chip_config3)

芯片配置寄存器 3 提供 RGMII IO 和普通 3.3V IO 的特性设置。

表 5.23: 芯片配置寄存器 3

位域	名称	访问	初值	描述
31	mac_update	R/W	0	更新配置先置位后清零, 29:16 的配置才会生效。
30	-	-	0	保留
29:23	mac_asrc	R/W	0	
22	mac_freeze	R/W	0	

位域	名称	访问	初值	描述
21	mac_comptq	R/W	0	
20	mac_compen	R/W	0	
19:18	mac_drive	R/W	0	RGMII 接口驱动强度选择 00: 4mA 01: 6mA 10: 8mA 11: 10mA
17:16	mac_volt	R/W	0x2	RGMII 接口电压选择 10: 2.5V 11: 3.3V
15	pad_update	R/W	0	更新配置 先置位后清零, 14:0 的配置才会生效
14	dvo_drive2	R/W	0	DVO_CLKp、DVO_CLKn 的驱动强度选择 0: 2mA 1: 8mA
13	dvo_drive1	R/W	0	DVO_HSYNC、DVO_VSYNC、DVO_DE 的驱动强度选择 0: 2mA 1: 8mA
12	dvo_drive0	R/W	0	DVO_DAT 的驱动硬件选择 0: 2mA 1: 8mA
11	-	-	0	保留
10:4	phy_asrc	R/W	0	
3	-	-	0	保留
2	phy_freeze	R/W	0	
1	phy_comptq	R/W	0	
0	phy_compen	R/W	0	

5.5.17 芯片采样寄存器 0(chip_sample0)

表 5.24: 芯片采样寄存器 0

位域	名称	访问	初值	描述
31:16	bootcfg_lo	R	-	上电配置状态 bootcfg[15:0]
15:14	-	-	-	保留
13	ht_lock	R	-	ht pll 锁定
12	pix1_lock	R	-	pix1 pll 锁定
11	pix0_lock	R	-	pix0 pll 锁定
10	sys_lock	R	-	sys pll 锁定
9	ddr_lock	R	-	ddr pll 锁定
8	cpu_lock	R	-	cpu pll 锁定
7	thsens_ov	R	-	温度传感器溢出
6:0	thsens_val	R	-	温度传感器读数 7 位整数, 单位为摄氏度

5.5.18 芯片采样寄存器 1(chip_sample1)

表 5.25: 芯片采样寄存器 1

位域	名称	访问	初值	描述
31:4	--	-	保留, 读为 0	
3:0	bootcfg_hi	R	-	上电配置状态 bootcfg[19:16]

5.5.19 芯片采样寄存器 2(chip_sample2)

表 5.26: 芯片采样寄存器 2

位域	名称	访问	初值	描述
31:0	--	-	保留, 读为 0	

5.5.20 芯片采样寄存器 3(chip_sample3)

表 5.27: 芯片采样寄存器 3

位域	名称	访问	初值	描述
31:0	--	-	保留, 读为 0	

5.5.21 芯片采样寄存器 4(chip_sample4)

表 5.28: 芯片采样寄存器 4

位域	名称	访问	初值	描述
31	cmptl_ok	R	-	
30:24	cmptl_asrc	R	-	
23:16	-	-	-	保留, 读为 0
15:0	version	R	-	芯片版本号 0: 2H2 1: 2H3

5.5.22 芯片采样寄存器 5(chip_sample5)

表 5.29: 芯片采样寄存器 5

位域	名称	访问	初值	描述
31	cmppd_ok	R	-	
30:24	cmppd_asrc	R	-	
23	cmpgm_ok	R	-	
22:16	cmpgm_asrc	R	-	
15	cmpbt_ok	R	-	
14:8	cmpbt_asrc	R	-	

位域	名称	访问	初值	描述
7	cmpht_ok	R	-	
6:0	cmpht_asrc	R	-	

5.5.23 时钟配置寄存器 0(clock_ctrl0)

时钟寄存器提供 ddr 和 cpu 两个 pll 的配置。

表 5.30: 时钟配置寄存器 0

位域	名称	访问	初值	描述
31:24	ddrpll_ldf	R/W	0	ddr pll 倍频系数
23:22	ddrpll_odf	R/W	0	ddr pll 输出分频
21:19	ddrpll_idf	R/W	0	ddr pll 输入分频
18	ddrpll_pd	R/W	0	ddr pll 关断
17	ddrpll_sel	R/W	0	ddr pll 输出选择 0: 参考时钟旁路 1: pll 输出
16	ddrpll_set	R/W	0	ddr pll 配置更新 在配好 pd/idf/odf/ldf/clock_ctrl1 后再置位并清零, 从而更新到 PLL 中
15:13	-	-	-	保留
12	cpupll_sel	R/W	0	cpu pll 输出选择 0: 参考时钟旁路 1: pll 输出分频后时钟
11	cpupll_pd	R/W	0	cpu pll 关断
10:8	cpupll_odf	R/W	0	cpu pll 输出分频
7:1	cpupll_ldf	R/W	0	cpu pll 倍频系数
0	cpupll_set	R/W	0	cpu pll 配置更新 在配好 pd/odf/ldf 后再置位并清零

5.5.24 时钟配置寄存器 1(clock_ctrl1)

时钟配置寄存器 1 提供 ddr pll 的展频配置。

表 5.31: 时钟配置寄存器 1

位域	名称	访问	初值	描述
31	-	-	-	保留
30	strobe	R/W	0	ddr pll 展频参数更新 如使用展频模式, 则应置位
29	sscg_en	R/W	0	展频模式使能 0: 普通模式 1: 展频模式
28	sprd_ctrl	R/W	0	展频模式选择 0: 中心展频 1: 下展频
27:13	inc_step	R/W	0	调制深度
12:0	mdperiod	R/W	0	调制周期

5.5.25 时钟配置寄存器 2(clock_ctrl2)

时钟配置寄存器 2 提供 sys pll 参数以及 sbc、hdv 分频配置。

表 5.32: 时钟配置寄存器 2

位域	名称	访问	初值	描述
31	hdvdiv_set	R/W	0	hdv 分频器参数设置 先置位后清零, 以使设置生效
30	hdvdiv_pd	R/W	0	hdv 分频器关断, 高有效
29:24	hdvdiv_df	R/W	0	hdv 分频器分频系数 (1~31)
23	sbcdiv_set	R/W	0	sbc 分频器参数设置 先置位后清零, 以使设置生效
22	sbcdiv_pd	R/W	0	sbc 分频器关断, 高有效
21	sbcdiv_2	R/W	0	sbc 时钟减半运行, 该位不影响 GMAC 125M 时钟的生成。 当要求功耗较低且 IO 性能不要求时可配置为 1
20:16	sbcdiv_df	R/W	0	sbc 分频器分频系数 (1~31)
15:13	-	-	-	保留
12	syspll_sel	R/W	0	sys pll 输出选择 0: 参考时钟旁路 1: pll 输出分频后时钟
11	syspll_pd	R/W	0	sys pll 关断
10:8	syspll_odf	R/W	0	sys pll 输出分频
7:1	syspll_ldf	R/W	0	sys pll 倍频系数
0	syspll_set	R/W	0	sys pll 配置更新 在配好其它参数后再置位并清零

5.5.26 时钟配置寄存器 3(clock_ctrl3)

时钟配置寄存器 3 提供 pcie 参考时钟和 hda、gpu 时钟分频的配置。

表 5.33: 时钟配置寄存器 3

位域	名称	访问	初值	描述
31:28	-	-	0	保留
27	pe3ref_en	R/W	0	PCIE3_CLKp/n 输出使能, 高有效
26	pe2ref_en	R/W	0	PCIE2_CLKp/n 输出使能, 高有效
25	pe1ref_en	R/W	0	PCIE1_CLKp/n 输出使能, 高有效
24	pe0ref_en	R/W	0	PCIE0_CLKp/n 输出使能, 高有效
23	pediv_set	R/W	0	pcie_ref 分频器参数设置 先置位后清零, 以使设置生效
22	pediv_pd	R/W	0	pcie ref 分频器关断, 高有效
21:16	pediv_df	R/W	0	pcie ref 分频器分频系数 (1~31)
15	hdadiv_set	R/W	0	hda 分频器参数设置 先置位后清零, 以使设置生效
14	hdadiv_pd	R/W	0	hda 分频器关断, 高有效
13:8	hdadiv_df	R/W	0	hda 分频器分频系数 (1~31)
7	gpudiv_set	R/W	0	gpu 分频器参数设置 先置位后清零, 以使设置生效
6	gpudiv_pd	R/W	0	gpu 分频器关断, 高有效

位域	名称	访问	初值	描述
5:0	gpudiv_df	R/W	0	gpu 分频器分频系数 (1~31)

5.5.27 显示时钟配置寄存器 0(pixclk0/1_ctrl0)

显示时钟配置寄存器提供 pixclk0/1 的频率配置。

表 5.34: 显示时钟配置寄存器 0

位域	名称	访问	初值	描述
31	pstdiv_set	R/W	0	输出分频器参数设置 先置位后清零, 以使设置生效
30	pstdiv_pd	R/W	0	输出分频器关断, 高有效
29:24	pstdiv_df	R/W	0	输出分频器分频系数 (1~31)
23:16	pll_ldf	R/W	0x10	pll 倍频系数
15	prediv_set	R/W	0	输入分频器参数设置 先置位后清零, 以使设置生效
14	prediv_pd	R/W	0	输入分频器关断, 高有效
13:8	prediv_df	R/W	0	输入分频器分频系数 (1~31)
7	pll_pd	R/W	0	pll 关断, 高有效 在改变 pll 配置前必须清零, 完成修改后再置位。
6:5	pll_odf	R/W	0x2	pll 输出分频 00: 1 分频 01: 2 分频 10: 4 分频 11: 8 分频
4:2	pll_idf	R/W	0x2	pll 输入分频 1~7: 1 到 7 分频
1:0	ref_sel	R/W	0	参考时钟选择 00: refclk 01: sbc_clk 10: pcie_ref 11: hda_clk

5.5.28 显示时钟配置寄存器 1(pixclk0/1_ctrl1)

显示时钟配置寄存器提供 pixclk0/1 的展频配置。

表 5.35: 显示时钟配置寄存器 1

位域	名称	访问	初值	描述
31	strobe	R/W	0	pll 展频参数更新 如使用展频模式, 则应置位
30	sscg_en	R/W	0	展频模式使能 0: 普通模式 1: 展频模式
29	sprd_ctrl	R/W	0	展频模式选择 0: 中心展频 1: 下展频
28:14	inc.step	R/W	0	调制深度

位域	名称	访问	初值	描述
13:1	mdperiod	R/W	0	调制周期
0	pll_sel	R/W	0	时钟输出选择 0: 参考时钟旁路 1: pll 输出

5.5.29 USB PHY0 配置寄存器(usbphy0_ctrl)

配置 USB 接口 0/1/2 相关的电气特性。

表 5.36: USB PHY0 配置寄存器

位域	名称	访问	初值	描述
63:61	dppulldown	R/W	0	硬件默认配置 0
60:58	dmpulldown	R/W	0	硬件默认配置 0
57:56	txhsxvtune2	R/W	0	硬件默认配置 0
55	txpreemtune2	R/W	0	硬件默认配置 1
54	txrisetune2	R/W	0	硬件默认配置 0
53:50	txvreftune2	R/W	0	硬件默认配置 0x5
49:46	txfslstune2	R/W	0	硬件默认配置 0x3
45:43	sqrxtune2	R/W	0	硬件默认配置 0x4
42:40	compdistune2	R/W	0	硬件默认配置 0x4
39:38	txhsxvtune1	R/W	0	硬件默认配置 0
37	txpreemtune1	R/W	0	硬件默认配置 1
36	txrisetune1	R/W	0	硬件默认配置 0
35:32	txvreftune1	R/W	0	硬件默认配置 0x5
31:28	txfslstune1	R/W	0	硬件默认配置 0x3
27:25	sqrxtune1	R/W	0	硬件默认配置 0x4
24:22	compdistune1	R/W	0	硬件默认配置 0x4
21:20	txhsxvtune1	R/W	0	硬件默认配置 0
19	txpreemtune0	R/W	0	硬件默认配置 1
18	txrisetune0	R/W	0	硬件默认配置 0
17:14	txvreftune0	R/W	0	硬件默认配置 0x5
13:10	txfslstune0	R/W	0	硬件默认配置 0x3
9:7	sqrxtune0	R/W	0	硬件默认配置 0x4
6:4	compdistune0	R/W	0	硬件默认配置 0x4
3:1	otgtune	R/W	0	硬件默认配置 0x5
0	cfg_en	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

5.5.30 USB PHY1 配置寄存器(usbphy1_ctrl)

配置 USB 接口 3/4/5 相关的电气特性。

表 5.37: USB PHY1 配置寄存器

位域	名称	访问	初值	描述
63:61	dppulldown	R/W	0	硬件默认配置 0
60:58	dmpulldown	R/W	0	硬件默认配置 0
57:56	txhsxvtune2	R/W	0	硬件默认配置 0
55	txpreemtune2	R/W	0	硬件默认配置 1
54	txrisetune2	R/W	0	硬件默认配置 0
53:50	txvrefune2	R/W	0	硬件默认配置 0x5
49:46	txfslstune2	R/W	0	硬件默认配置 0x3
45:43	sqrxtune2	R/W	0	硬件默认配置 0x4
42:40	compdistune2	R/W	0	硬件默认配置 0x4
39:38	txhsxvtune1	R/W	0	硬件默认配置 0
37	txpreemtune1	R/W	0	硬件默认配置 1
36	txrisetune1	R/W	0	硬件默认配置 0
35:32	txvrefune1	R/W	0	硬件默认配置 0x5
31:28	txfslstune1	R/W	0	硬件默认配置 0x3
27:25	sqrxtune1	R/W	0	硬件默认配置 0x4
24:22	compdistune1	R/W	0	硬件默认配置 0x4
21:20	txhsxvtune1	R/W	0	硬件默认配置 0
19	txpreemtune0	R/W	0	硬件默认配置 1
18	txrisetune0	R/W	0	硬件默认配置 0
17:14	txvrefune0	R/W	0	硬件默认配置 0x5
13:10	txfslstune0	R/W	0	硬件默认配置 0x3
9:7	sqrxtune0	R/W	0	硬件默认配置 0x4
6:4	compdistune0	R/W	0	硬件默认配置 0x4
1	ls_mod	R/W	0	低速模式兼容性优化 0: 关闭 1: 打开 (建议配置, 不受 cfg_en 影响)
0	cfg_en	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

5.6 中断配置及路由

龙芯 2H 内置简单、灵活的中断控制器，管理内部中断事件以及外部中断输入。中断控制器结构如图 5.5 所示，由 5 组中断控制寄存器组成。每组寄存器有 32 个中断控制位，对应内外中断源。每个中断控制位有一组寄存器加以控制，包括 intien、intpol、intedge、intset、intclr 等，中断状态可由 intisr 读出，详见表 5.9~5.11

中断控制器共五个中断输出连接处理器核，分别对应 INT0, INT1, INT2, INT3, INT4。芯片支持 64 个内部中断和 16 个 GPIO 中断，如表 5.38 所示。

表 5.38: 中断源映射关系

	INT0	INT1	INT2	INT3	INT4
31	保留	保留	保留	保留	保留

	INT0	INT1	INT2	INT3	INT4
30	保留	保留	保留	保留	保留
29	保留	保留	保留	保留	保留
28	保留	保留	保留	保留	保留
27	保留	保留	保留	保留	保留
26	保留	保留	保留	保留	保留
25	保留	HDA	保留	保留	保留
24	保留	SATAPHY	保留	保留	保留
23	SYS_INTn	PCIE3	保留	保留	保留
22	NAND	PCIE2	保留	保留	保留
21	TOY_TICK	PCIE1	保留	保留	保留
20	RTC_TICK	PCIE0	保留	保留	保留
19	TOY_INT2	HT7	保留	保留	保留
18	TOY_INT1	HT6	保留	保留	保留
17	TOY_INT0	HT5	保留	保留	保留
16	RTC_INT2	HT4	保留	保留	保留
15	RTC_INT1	HT3	GPIO15	保留	保留
14	RTC_INT0	HT2	GPIO14	保留	保留
13	LPC	HT1	GPIO13	保留	保留
12	DMA2	HT0	GPIO12	保留	保留
11	DMA1	PWM3	GPIO11	保留	保留
10	DMA0	PWM2	GPIO10	保留	保留
9	AC97	PWM1	GPIO09	保留	保留
8	I2C1	PWM0	GPIO08	保留	保留
7	I2C0	DC	GPIO07	保留	保留
6	SPI	GPU	GPIO06	保留	保留
5	UART3	SATA	GPIO05	保留	保留
4	UART2	GMAC1	GPIO04	保留	保留
3	UART1	GMAC0	GPIO03	保留	保留
2	URAT0	OTG	GPIO02	保留	保留
1	HPET	OHCI	GPIO01	保留	保留
0	ACPI	EHCI	GPIO00	保留	保留

INT3 所对应的中断控制器用于接收 MSI 中断，往消息中断写端口写入的 5 位向量号将译码成 32 位的向量，产生与写向 intset 寄存器相同的结果。在使用中断前，软件需根据表 5.39 的值，从左至右逐列配置中断控制器。其中 intien 设为使能所有中断，用户可根据需要配置。

表 5.39: 中断控制器配置

	intedge	intpol	intclr	intien
INT0	0x00000000	0xff7fffff	0x00000000	0x00ffffff
INT1	0x00000000	0xfeffffff	0x00000000	0x03ffffff
INT2	0x00000000	0xffffffff	0x00000000	0x00000000
INT3	0xffffffff	0xffffffff	0xffffffff	0xffffffff
INT4	0xffffffff	0xffffffff	0xffffffff	0x00000000

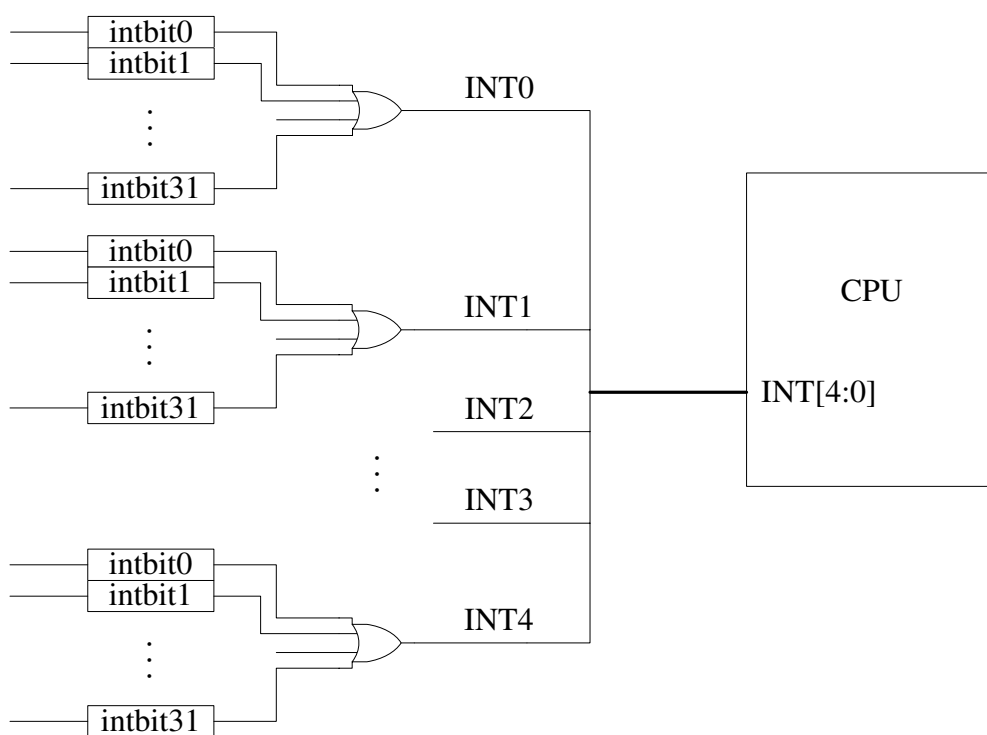


图 5.5: 中断结构示意图

第六章 DDR2/3 控制器

龙芯 2H 处理器内部集成的 DDR2/3 SDRAM 内存控制器的设计遵守 DDR2/3 SDRAM 的行业标准（JESD79-2 和 JESD79-3），所实现的所有内存读 / 写操作都遵守 JESD79-2B 及 JESD79-3 的规定。

6.1 功能概述

龙芯 2H 处理器支持最大 4 个 CS（由 4 个 DDR2 SDRAM 片选信号实现，即两个双面内存条），一共含有 18 位的地址总线（即：15 位的行列地址总线和 3 位的逻辑 Bank 总线）。

龙芯 2H 处理器在具体选择使用不同内存芯片类型时，可以调整 DDR2/3 控制器参数设置进行支持。其中，支持的最大片选（CS_n）数为 4，行地址（RAS_n）数为 15，列地址（CAS_n）数为 14，逻辑体选择（BANK_n）数为 3。最大支持的地址空间为 128GB（2³⁷）。

CPU 发送的内存请求物理地址将按照图 6.1 所示的方法进行行列地址转换：

以 4GB 地址空间为例，按照下面的配置：

片选 = 4 Bank 数 = 8
行地址数 = 12 列地址数 = 12

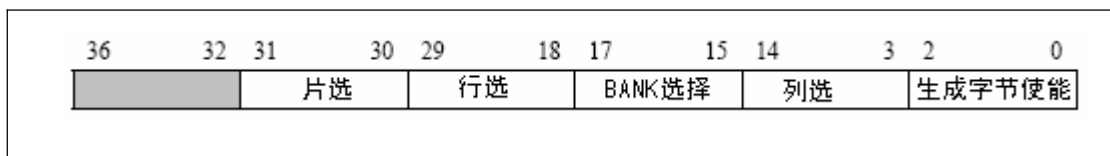


图 6.1: DDR2 SDRAM 行列地址与 CPU 物理地址的转换

龙芯 2H 处理器所集成的内存控制电路接受来自处理器或者内部设备的内存读 / 写请求，在所有的内存读 / 写操作中，内存控制电路处于从设备状态（Slave State）。

龙芯 2H 处理器中内存控制器具有如下特征：

- 接口上命令、读写数据全流水操作
- 内存命令合并、排序提高整体带宽
- 配置寄存器读写端口，可以修改内存设备的基本参数
- 内建动态延迟补偿电路（DCC），用于数据的可靠发送和接收
- ECC 功能可以对数据通路上的 1 位和 2 位错误进行检测，并能对对 1 位错误进行自动纠错

- 支持 133-400MHZ 工作频率

6.2 读操作协议

DDR2/3 SDRAM 读操作的协议如图6.2所示。在图中命令（Command，简称 CMD）由 RAS_n，CAS_n 和 WE_n，共三个信号组成。对于读操作，RAS_n=1，CAS_n=0，WE_n=1。

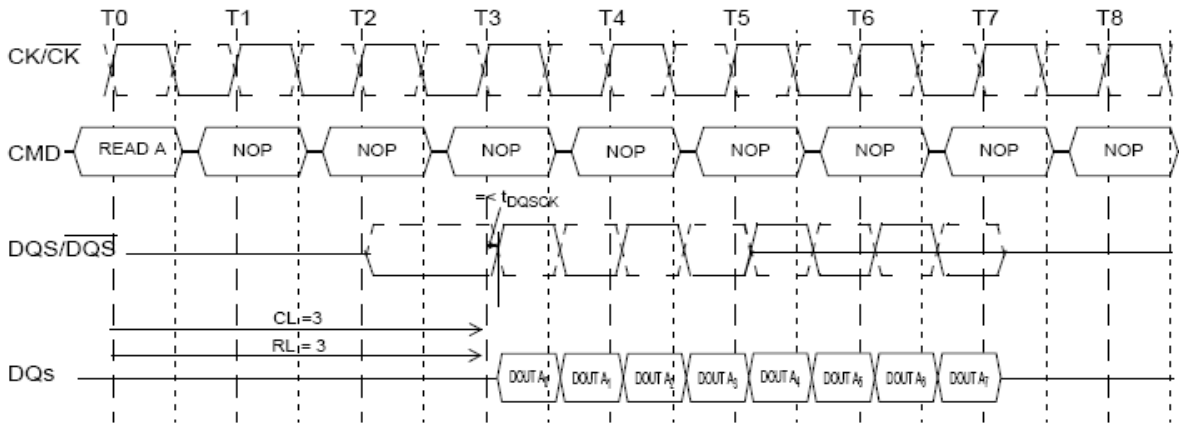


图 6.2: DDR2 SDRAM 读操作协议

其中，Cas Latency (CL) = 3，Read Latency (RL) = 3，Burst Length = 8。

6.3 写操作协议

DDR2/3 SDRAM 写操作的协议如图6.3所示。在图中命令 CMD 是由 RAS_n，CAS_n 和 WE_n，共三个信号组成的。对于写操作，RAS_n=1，CAS_n=0，WE_n=0。另外，与读操作不同，写操作需要 DQM 来标识写操作的掩码，即需要写入的字节数。DQM 与图中 DQs 信号同步。其中，Cas Latency (CL) = 3，Write Latency (WL) = Read Latency (RL) - 1 = 2，Burst Length = 4。

6.4 参数配置格式

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。

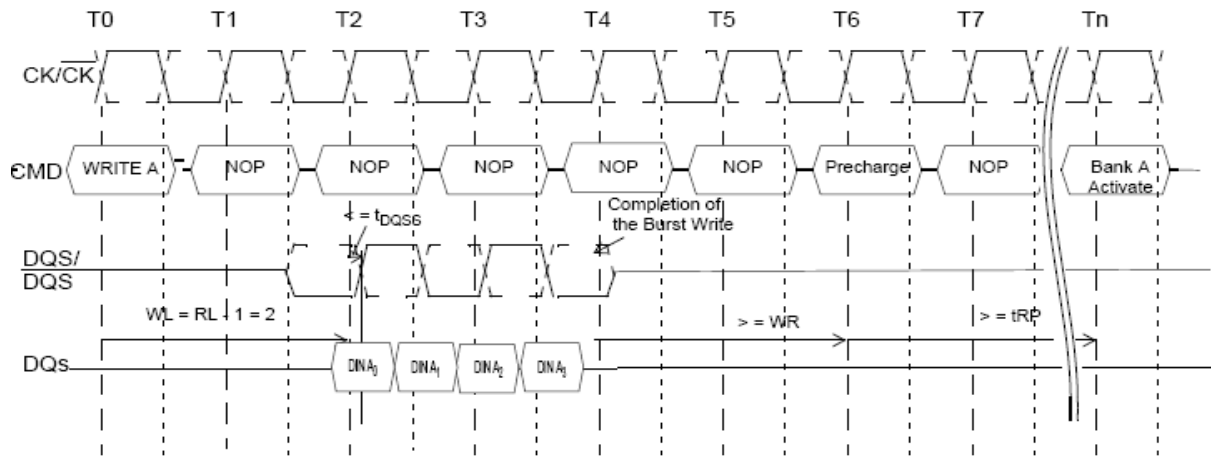


图 6.3: DDR2 SDRAM 写操作协议

3. 向配置寄存器地址发 64 位写指令，配置所有 180 个配置寄存器。此时如果写 CTRL_03，应将其中参数 START 设为 0。所有寄存器都必须正确配置才可以正常工作。
4. 向配置寄存器 CTRL_03 中发 64 位写指令。此时应将参数 START 设为 1。结束后内存控制器将自动对内存发起初始化指令。

在龙芯 2H 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 0x0000 0000 0ff0 0000 相对应的 180 个 64 位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），表中还给出了基于 DDR2 667 的一种寄存器配置方式，具体的配置可以根据实际情况再决定：

表 6.1: DDR2 SDRAM 配置参数寄存器格式

参数名称	位	初值	范围	描述
CONF_CTL_00[63:0] Offset: 0x00 DDR2 667 : 0x0000010000000101				
CONCURRENTAP	48:48	0x0	0x0-0x1	是否允许控制器对一个 bank 进行 auto precharge 时，对另外一个 bank 发出命令。注：部分内存条不支持
BANK_SPLIT_EN	40:40	0x0	0x0-0x1	是否允许命令队列重排序逻辑对 bank 进行拆分（split）
AUTO_REFRESH_MODE	32:32	0x0	0x0-0x1	设置 auto-refresh 是在下一个 burst 还是下一个命令边界发出
AREFRESH	24:24	0x0	0x0-0x1	根据 auto_refresh_mode 参数的设置，向内存发起自动刷新命令（只写）
AP	16:16	0x0	0x0-0x1	是否使能内存控制器自动刷新功能，置 1，表示内存访问为 CLOSE PAGE 方式。

参数名称	位	初值	范围	描述
ADDR_CMP_EN	8:8	0x0	0x0-0x1	是否允许命令队列重排序逻辑对地址冲突进行检测
CONF_CTL_01[63:0] Offset: 0x10 DDR2 667 : 0x0000010100010000				
FWC	56:56	0x0	0x0-0x1	是否强制进行写检查, 当这个参数设置后, 内存控制器将用 xor.check.bits 参数指定的数与数据进行异或写入内存(只写)
FAST_WRITE	48:48	0x0	0x0-0x1	是否允许控制器打开快速写功能。打开快速写功能后, 控制器在未收到全部写数据后即向内存模块发出写命令。
ENABLE_QUICK_SREFRESH	40:40	0x0	0x0-0x1	是否使能快速自刷新。当这个参数使能后, 内存的初始化未进行完就进入自刷新状态
EIGHT_BANK_MODE	32:32	0x0	0x0-0x1	指示内存模块是否有 8 个 bank
ECC_DISABLE_W_UC_ERR	24:24	0x0	0x0-0x1	当检测到不可恢复的错误时, 是否将 ECC 关闭
DQS_N_EN	16:16	0x0	0x0-0x1	设置 DQS 信号为单端还是差分信号。
DLLLOCKREG	0:0	0x0	0x0-0x1	指示 DLL 是否已锁定(只读), 只有在 DLL 锁定之后, 对内存发起的读写操作才能有效到达内存, 所以, 可以用本位判断第一次写内存的时机。
CONF_CTL_02[63:0] Offset: 0x20 DDR2 667 : 0x0100010100000000				
PRIORITY_EN	56:56	0x0	0x0-0x1	是否使能命令队列重排序逻辑使用优先级
POWER_DOWN	48:48	0x0	0x0-0x1	当使能这个参数时, 内存控制器将用 pre-charge 命令关闭内存模块的所有页面, 使时钟使能信号为低, 不发送收到的所有命令, 直到这个参数重新设置为 0
PLACEMENT_EN	40:40	0x0	0x0-0x1	是否使能命令重排序逻辑
ODT_ADD_TURN_CLK_EN	32:32	0x0	0x0-0x1	在对不同片选的快速背对背读或者写命令中间是否插入一个 turn-around 时钟。通常情况下, 插入一个这样的周期是对内存需要的。
NO_CMD_INIT	24:24	0x0	0x0-0x1	在内存初始化过程中, 是否禁止在内存模块的 tDLL 时间内发出其它命令
INTRPTWRITENA	16:16	0x0	0x0-0x1	是否允许用 autoprecharge 命令加上对同一 bank 的其它写命令打断前一个写命令
INTRPTREADA	8:8	0x0	0x0-0x1	是否允许用 autoprecharge 命令加上对同一 bank 的其它读命令打断前一个读命令
INTRPTAPBURST	0:0	0x0	0x0-0x1	是否允许对另一 bank 的其它命令打断当前的 auto-precharge 命令
CONF_CTL_03[63:0] Offset: 0x30 DDR2 667 : 0x0101010001000000				
SWAP_PORT_RW_SAME_EN	56:56	0x0	0x0-0x1	当 swap_en 使能时, 该参数决定是否将同一端口上的类似命令进行交换
SWAP_EN	48:48	0x0	0x0-0x1	在使能命令队列重排序逻辑时, 当高优先级命令到达时, 是否将正在执行的命令与新命令交换
START	40:40	0x0	0x0-0x1	是否开始内存的初始化工作。需要在所有的参数配置完成之后, 再设置该位, 让内存进入初始化配置。在没有完成其它位的配置之前就配置该位, 很可能导致内存访问错误。
SREFRESH	32:32	0x0	0x0-0x1	内存模块是否进入自刷新工作模式
RW_SAME_EN	24:24	0x0	0x0-0x1	在命令队列重排序逻辑中是否考虑对同一 bank 读写命令的重组
REG_DIMM_EN	16:16	0x0	0x0-0x1	是否使能 registered DIMM 内存模组

参数名称	位	初值	范围	描述
REDUC	8:8	0x0	0x0-0x1	是否只使用 32 位位宽的内存数据通道，通常情况下，不应设置该位
PWRUP_SREFRESH_EXIT	0:0	0x0	0x0-0x1	是用 self-refresh 命令而不是用正常的内存初始化命令来脱离下电模式
CONF_CTL_04[63:0] Offset: 0x40 DDR2 667 : 0x0102010100010101				
RTT_0	57:56	0x0	0x0-0x3	使能内存模块的片上终端电阻。 00 - disable 其它 - enable，电阻大小由 mrs_data 中的值决定
CTRL_RAW	49:48	0x0	0x0-0x3	设置 ECC 的检错和纠错模式 2'b00 - 不使用 ECC 2'b01 - 只报错，不纠错 2'b10 - 没有使用 ECC 设备 2'b11 - 使用 ECC 报错纠错
AXI0_W_PRIORITY	41:40	0x0	0x0-0x3	设置 AXI0 端口写命令优先级
AXI0_R_PRIORITY	33:32	0x0	0x0-0x3	设置 AXI0 端口读命令优先级
WRITE_MODEREG	24:24	0x0	0x0-0x1	是否写内存模块的 EMRS 寄存器（只写），每次写 1 时控制器就将配置参数中 emrs_data 与 mrs_data 发往内存。
WRITEINTERP	16:16	0x0	0x0-0x1	定义是否能用一个读命令取打断一个写突发
TREF_ENABLE	8:8	0x0	0x0-0x1	是否使能控制器内部的自动刷新功能，通常的情况下，应该将该位置 1
TRAS_LOCKOUT	0:0	0x0	0x0-0x1	是否在 tRAS 时间到期之前发出 auto-precharge 命令
CONF_CTL_05[63:0] Offset: 0x50 DDR2 667 : 0x0700000404050100				
Q_FULLNESS	58:56	0x0	0x0-0x7	定义内存控制器命令队列中有多少命令时认为命令队列满
PORT_DATA_ERROR_TYPE	50:48	0x0	0x0-0x7	定义内存控制器端口上数据错误类型（只读） 位 0 - 突发数据个数大于 16 位 1 - 写数据交错 位 2 - ECC 2 位错
OUT_OF_RANGE_TYPE	42:40	0x0	0x0-0x7	定义发生越界访问时的错误类型（只读）
MAX_CS_REG	34:32	0x4	0x0-0x4	定义控制器所用片选个数（只读）
COLUMN_SIZE	26:24	0x0	0x0-0x7	设置实际列地址数和最大列地址数 (14) 之间的差值，应该根据具体的内存颗粒进行配置。 内存所用列地址数 = 14 - COLUMN_SIZE
CASLAT	18:16	0x0	0x0-0x7	设置 CAS latency 值。应当根据具体的内存颗粒在不同的运行频率下进行配置。
ADDR_PINS	10:8	0x0	0x0-0x7	设置实际地址引脚数和最大地址数 (15) 之间的差值 内存所用地址线数 = 15 - ADDR_PINS
CONF_CTL_06[63:0] Offset: 0x60 DDR2 667 : 0x0a04040603040003				
APREBIT	59:56	0x0	0x0-0xf	定义用哪位地址线向内存发出 autoprecharge 命令，一般为 bit 10。

参数名称	位	初值	范围	描述
WRLAT	50:48	0x0	0x0-0x7	写操作时写命令发出到接收到第一个数据的时间（按时钟周期数），同时决定何时使对应的 ODT 信号有效。 注：当 $WRLAT = (CASLAT_LIN / 2)$ 时，会在不同 CS 读写之间加入一拍额外延迟。
TWTR	42:40	0x0	0x0-0x7	定义从写命令切换到读命令所需要的时钟周期数，需要根据具体内存颗粒及运行频率进行配置。
TWR_INT	34:32	0x0	0x0-0x7	定义内存模组的写恢复时间，需要根据具体内存颗粒及运行频率进行配置。
TRTP	26:24	0x0	0x0-0x7	定义内存模组的读命令到 precharge 周期数，需要根据具体内存颗粒及运行频率进行配置。
TRRD	18:16	0x0	0x0-0x7	定义到不同 bank 的 active 命令时间间隔，需要根据具体内存颗粒及运行频率进行配置。
TCKE	2:0	0x0	0x0-0x7	定义 CKE 信号最小脉宽
CONF_CTL_07[63:0] Offset: 0x70 DDR2 667 : 0x0f0e020000f0a0a				
MAX_ROW_REG	59:56	0xf	0x0-0xf	系统最大行地址个数（只读）
MAX_COL_REG	51:48	0xe	0x0-0xe	系统最大列地址个数（只读）
INITAREF	43:40	0x0	0x0-0xf	定义系统初始化时所需要执行的 autorefresh 命令个数。DDR2 时设为 2，DDR3 时设为 0。
CS_MAP	19:16	0x0	0x0-0xf	定义可用片选信号，本参数应当根据实际使用的片选个数进行正确的配置，不正确的配置将会导致错误的内存访问。该参数的四位分别对应于 CS0-CS3
CASLAT_LIN	3:0	0x0	0x0-0xf	当板上走线延迟为 DDR2 时钟周期的 0.5~1.5 倍： $CASLAT_LIN = CASLAT \times 2$ 小于 0.5 倍： $CASLAT_LIN = CASLAT \times 2-1$ 大于 1.5 倍： $CASLAT_LIN = CASLAT \times 2+1$ (以半个时钟周期为单位)
CONF_CTL_08[63:0] Offset: 0x80 DDR2 667 : 0x0804020108040201				
ODT_WR_MAP_CS3	59:56	0x0	0x0-0xf	定义 CS3 有写命令时，将指定的 CS 的 ODT 终端电阻有效，具体的配置应当参考相应的内存颗粒手册对于 ODT 配置的要求。该参数的四位分别对应于 CS0- CS3
ODT_WR_MAP_CS2	51:48	0x0	0x0-0xf	定义 CS2 有写命令时，将指定的 CS 的 ODT 终端电阻有效，具体的配置应当参考相应的内存颗粒手册对于 ODT 配置的要求。该参数的四位分别对应于 CS0- CS3
ODT_WR_MAP_CS1	43:40	0x0	0x0-0xf	定义 CS1 有写命令时，将指定的 CS 的 ODT 终端电阻有效，具体的配置应当参考相应的内存颗粒手册对于 ODT 配置的要求。该参数的四位分别对应于 CS0- CS3
ODT_WR_MAP_CS0	35:32	0x0	0x0-0xf	定义 CS0 有写命令时，将指定的 CS 的 ODT 终端电阻有效，具体的配置应当参考相应的内存颗粒手册对于 ODT 配置的要求。该参数的四位分别对应于 CS0- CS3

参数名称	位	初值	范围	描述
ODT_RD_MAP_CS3	27:24	0x0	0x0-0xf	定义 CS3 有读命令时, 将指定的 CS 的 ODT 终端电阻有效, 具体的配置应当参考相应的内存颗粒手册对于 ODT 配置的要求。该参数的四位分别对应于 CS0- CS3
ODT_RD_MAP_CS2	19:16	0x0	0x0-0xf	定义 CS2 有读命令时, 将指定的 CS 的 ODT 终端电阻有效, 具体的配置应当参考相应的内存颗粒手册对于 ODT 配置的要求。该参数的四位分别对应于 CS0- CS3
ODT_RD_MAP_CS1	11:8	0x0	0x0-0xf	定义 CS1 有读命令时, 将指定的 CS 的 ODT 终端电阻有效, 具体的配置应当参考相应的内存颗粒手册对于 ODT 配置的要求。该参数的四位分别对应于 CS0- CS3
ODT_RD_MAP_CS0	3:0	0x0	0x0-0xf	定义 CS0 有读命令时, 将指定的 CS 的 ODT 终端电阻有效, 具体的配置应当参考相应的内存颗粒手册对于 ODT 配置的要求。该参数的四位分别对应于 CS0- CS3
CONF_CTL_09[63:0] Offset: 0x90 DDR2 667 : 0x0000070d00000000				
OCD_ADJUST_PUP_CS0	60:56	0x0	0x0-0x1f	设置内存模组片选 0 OCD 上拉调整值。内存控制器将在初始化时根据这个参数的值向内存模组发出 OCD 调整命令
OCD_ADJUST_PDN_CS0	52:48	0x0	0x0-0x1f	设置内存模组片选 0 OCD 下拉调整值。内存控制器将在初始化时根据这个参数的值向内存模组发出 OCD 调整命令
TRP	43:40	0x0	0x0-0xf	定义内存模组执行 pre-charge 所需要的时钟周期数, 需要根据具体内存颗粒及运行频率进行配置。
TDAL	35:32	0x0	0x0-0xf	当 auto-precharge 参数设置后, 该参数定义了 auto-precharge 和 write recovery 时钟周期数。 TDAL = auto-precharge + write recovery 该参数仅在设置了 AP 之后才生效。
PORT_CMD_ERROR_TYPE	19:16	0x0	0x0-0xf	端口上发生命令错误的类型 (只读) 位 0 - 数据位宽过大 位 1 - 关键字优先操作地址未对齐 位 2 - 关键字优先操作字数不是 2 幂 位 3 - narrow transform 出错
CONF_CTL_10[63:0] Offset: 0xa0 DDR2 667 : 0x0000003f3f140612				
COMMAND_AGE_COUNT	37:32	0x0	0x0-0x3f	定义命令队列重排序逻辑使用 aging 算法时每个命令的 aging 初始值
AGE_COUNT	29:24	0x0	0x0-0x3f	定义命令队列重排序逻辑使用 aging 算法时每个命令的 aging 初始值
TRC	20:16	0x0	0x0-0x1f	定义对内存模组同一 bank 的 active 命令之间的时钟周期数, 需要根据具体内存颗粒及运行频率进行配置。
TMRD	12:8	0x0	0x0-0x1f	定义配置内存模组模式寄存器需要的时钟周期数, 通常为 2 个周期
TFAW	4:0	0x0	0x0-0x1f	定义内存模组 tFAW 参数, 8 个逻辑 bank 时使用
CONF_CTL_12[63:0] Offset: 0xc0 DDR2 667 : 0x00002c0511000000				
TRFC	47:40	0x0	0x0-0xff	定义内存模组刷新操作需要的时钟周期数, 需要根据具体内存颗粒及运行频率进行配置。

参数名称	位	初值	范围	描述
TRCD_INT	39:32	0x0	0x0-0xff	定义内存模组 RAS 到 CAS 之间的时钟周期数, 需要根据具体内存颗粒及运行频率进行配置。
TRAS_MIN	31:24	0x0	0x0-0xff	定义内存模组行地址有效命令的最小时钟周期数
OUT_OF_RANGE_LENGTH	23:16	0x0	0x0-0xff	发生越界访问时的命令长度 (只读)
ECC_U_SYND	15:8	0x0	0x0-0xff	发生 2bit 不可纠错错误时的原因 (只读)
ECC_C_SYND	7:0	0x0	0x0-0xff	发生 1bit 可纠错错误时的原因 (只读)
CONF_CTL_17[63:0] Offset: 0x110 DDR2 667 : 0x000000000000c2d				
TREF	13:0	0x0	0x0-0x3ff	定义内存模组两次刷新命令的时钟间隔, 需要根据具体内存颗粒及运行频率进行配置。
CONF_CTL_18[63:0] Offset: 0x120 DDR2 667 : 0x001c000000000000				
AXIO_EN_LT_WIDTH_INSTR	63:48	0x0	0x0-0xffff	定义 AXIO 端口是否接收小于 64 位位宽的内存访问
CONF_CTL_19[63:0] Offset: 0x130 DDR2 667 : 0x6d56000302000000				
TRAS_MAX	63:48	0x0	0x0-0xffff	定义内存模组行有效命令的最大时钟周期数, 需要根据具体内存颗粒及运行频率进行配置。
TPDEX	47:32	0x0	0x0-0xffff	定义内存模组掉电退出命令的时钟周期数
TDLL	31:16	0x0	0x0-0xffff	定义内存模组 DLL 锁定需要的时钟周期数
TCPD	15:0	0x0	0x0-0xffff	定义内存模组时钟有效到 precharge 之间的时钟周期数, 需要根据具体内存颗粒及运行频率进行配置。
CONF_CTL_20[63:0] Offset: 0x140 DDR2 667 : 0x0000204002000030				
XOR_CHECK_BITS	63:48	0x0	0x0-0xffff	当 fwc 参数设定时, 下次写操作的 check bit 将会与该参数进行异或后写入内存
VERSION	47:32	0x2040	0x2040	定义内存控制器版本号 (只读)
TXSR	31:16	0x0	0x0-0xffff	定义内存模组自刷新退出需要的时钟周期数
TXSNR	15:0	0x0	0x0-0xffff	定义内存模组 tXSNR 参数
CONF_CTL_21[63:0] Offset: 0x150 DDR2 667 : 0x0000000000000000				
ECC_C_ADDR[36:8]	60:32	0x0	0x0-0xffffffff	记录发生 1bit ECC 错误时的地址信息 (只读)
ECC_C_ADDR[7:0]	31:24	0x0	0x0-0xffffffff	记录发生 1bit ECC 错误时的地址信息 (只读)
TINIT	23:0	0x0	0x0-0xffff	定义内存模组初始化需要的时钟周期数, 需要根据具体内存颗粒及运行频率进行配置。一般为 200us。
CONF_CTL_22[63:0] Offset: 0x160 DDR2 667 : 0x0000000000000000				
ECC_U_ADDR[36:32]	36:32	0x0	0x0-0xffffffff	记录发生 2bit ECC 错误时的地址信息 (只读)
ECC_U_ADDR[31:0]	31:0	0x0	0x0-0xffffffff	记录发生 2bit ECC 错误时的地址信息 (只读)
CONF_CTL_23[63:0] Offset: 0x170 DDR2 667 : 0x0000000000000000				
OUT_OF_RANGE_ADDR[36:32]	36:32	0x0	0x0-0xffffffff	记录发生越界访问时的地址信息 (只读)
OUT_OF_RANGE_ADDR[31:0]	31:0	0x0	0x0-0xffffffff	记录发生越界访问时的地址信息 (只读)
CONF_CTL_24[63:0] Offset: 0x180 DDR2 667 : 0x0000000000000000				
PORT_CMD_ERROR_ADDR[36:32]	36:32	0x0	0x0-0xffffffff	记录端口发生命令错误时的地址信息 (只读)

参数名称	位	初值	范围	描述
PORT_CMD_ERROR_ADDR[31:0]	31:0	0x0	0x0-0x1fffffff	记录端口发生命令错误时的地址信息（只读）
CONF_CTL_25[63:0] Offset: 0x190 DDR2 667 : 0x0000000000000000				
ECC_C_DATA[63:32]	63:32	0x0	0x0-0x1fffffff	记录发生 1bit ECC 错误时的数据信息（只读）
ECC_C_DATA[31:0]	31:0	0x0	0x0-0x1fffffff	记录发生 1bit ECC 错误时的数据信息（只读）
CONF_CTL_26[63:0] Offset: 0x1a0 DDR2 667 : 0x0000000000000000				
ECC_U_DATA[63:32]	63:32	0x0	0x0-0x1fffffff	记录发生 2bit ECC 错误时的数据信息（只读）
ECC_U_DATA[31:0]	31:0	0x0	0x0-0x1fffffff	记录发生 2bit ECC 错误时的数据信息（只读）
CONF_CTL_27[63:0] Offset: 0x1b0 DDR2 667 : 0x0000000000000000				
CKE_DELAY	2:0	0x0	0x0-0x7	CKE 有效延迟。 注：用于控制内部 srefresh_enter 命令的响应时间，对于龙芯 2 号无效。
CONF_CTL_29[63:0] Offset: 0x1d0 DDR2 667 : 0x0103070400000101				
TDFI_PHY_WRLAT_BASE	59:56	0x0	0x0-0xf	设置 DDR PHY 中写数据需加入的延迟。对于龙芯 2 号这个值应为 2
TDFI_PHY_WRLAT	51:48	0x0	0x0-0xf	用于显示实际从写命令发出到写数据发出间隔的周期数（只读）
TDFI_PHY_RDLAT	44:40	0x0	0x0-0xf	设置读命令发出到读数据返回间隔的周期数
TDFI_CTRLUPD_MIN	35:32	0x4	0x0-0xf	保存 DFI Tctrlup_min 时间参数（只读）
DRAM_CLK_DISABLE	19:16	0x0	0x0-0xf	设置是否输出 DRAM 时钟信号，每位对应一个片选信号。0：输出时钟信号；1：禁止输出时钟信号。
ODT_ALT_EN	8:8	0x0	0x0-0x1	是否支持 CAS = 3 时的 ODT 信号。 注：对于龙芯 2 号，无效
DRIVE_DQ_DQS	0:0	0x0	0x0-0x1	设置当控制器空闲时是否驱动数据总线
CONF_CTL_30[63:0] Offset: 0x1e0 DDR2 667 : 0x0c2d0c2d0c2d0205				
TDFI_PHYUPD_TYPE0	61:48	0x0	0x0-0x3fff	这个值等于 TREF（只读）
TDFI_PHYUPD_RESP	45:32	0x0	0x0-0x3fff	这个值等于 TREF（只读）
TDFI_CTRLUPD_MAX	29:16	0x0	0x0-0x3fff	这个值等于 TREF（只读）
TDFI_RDDATA_EN_BASE	12:8	0x0	0x0-0x1f	DDR PHY 内部读命令发出到读返回的基本时间。对于龙芯 2 号这个值为 2
TDFI_RDDATA_EN	4:0	0x0	0x0-0x1f	用于显示从读命令发出到读数据返回的实际周期数
CONF_CTL_31[63:0] Offset: 0x1f0 DDR2 667 : 0x0020008000000000				
DLL_CTRL_REG_0_0	63:32	0x0	0x0-0xffffffff	第 0 数据组（DQ7-DQ0）DLL 控制信号 24：控制内部 DLL 的使能信号，为 0 时 DLL 有效 23:16：控制写数据（DQ）与 DQS 之间的相位关系，每个数值表示为 (1/精度) * 360。在龙芯 2 号中，这个值一般为 1/4，即 8'h20 7:0：控制内部 DLL 的精度，在龙芯 2 号中，这个值一般为 8'h80

参数名称	位	初值	范围	描述
DFT_CTRL_REG	7:0	0x0	0x0-0xff	测试使能信号, 0x0 为正常工作模式
CONF_CTL_32[63:0] Offset: 0x200 DDR2 667 : 0x0020008000200080				
DLL_CTRL_REG_0_2	63:32	0x0	0x0-0xffffffff	第 2 数据组 (DQ23-DQ16) DLL 控制信号 24: 控制内部 DLL 的使能信号, 为 0 时 DLL 有效 23:16: 控制写数据 (DQ) 与 DQS 之间的相位关系, 每个数值表示为 (1/精度) * 360。在龙芯 2 号中, 这个值一般为 1/4, 即 8' h20 7:0: 控制内部 DLL 的精度, 在龙芯 2 号中, 这个值一般为 8' h80
DLL_CTRL_REG_0_1	31:0	0x0	0x0-0xffffffff	第 1 数据组 (DQ15-DQ8) DLL 控制信号 24: 控制内部 DLL 的使能信号, 为 0 时 DLL 有效 23:16: 控制写数据 (DQ) 与 DQS 之间的相位关系, 每个数值表示为 (1/精度) * 360。在龙芯 2 号中, 这个值一般为 1/4, 即 8' h20 7:0: 控制内部 DLL 的精度, 在龙芯 2 号中, 这个值一般为 8' h80
CONF_CTL_33[63:0] Offset: 0x210 DDR2 667 : 0x0020008000200080				
DLL_CTRL_REG_0_4	63:32	0x0	0x0-0xffffffff	第 4 数据组 (DQ39-DQ32) DLL 控制信号 24: 控制内部 DLL 的使能信号, 为 0 时 DLL 有效 23:16: 控制写数据 (DQ) 与 DQS 之间的相位关系, 每个数值表示为 (1/精度) * 360。在龙芯 2 号中, 这个值一般为 1/4, 即 8' h20 7:0: 控制内部 DLL 的精度, 在龙芯 2 号中, 这个值一般为 8' h80
DLL_CTRL_REG_0_3	31:0	0x0	0x0-0xffffffff	第 3 数据组 (DQ31-DQ24) DLL 控制信号 24: 控制内部 DLL 的使能信号, 为 0 时 DLL 有效 23:16: 控制写数据 (DQ) 与 DQS 之间的相位关系, 每个数值表示为 (1/精度) * 360。在龙芯 2 号中, 这个值一般为 1/4, 即 8' h20 7:0: 控制内部 DLL 的精度, 在龙芯 2 号中, 这个值一般为 8' h80
CONF_CTL_34[63:0] Offset: 0x220 DDR2 667 : 0x0020008000200080				
DLL_CTRL_REG_0_6	63:32	0x0	0x0-0xffffffff	第 6 数据组 (DQ55-DQ48) DLL 控制信号 24: 控制内部 DLL 的使能信号, 为 0 时 DLL 有效 23:16: 控制写数据 (DQ) 与 DQS 之间的相位关系, 每个数值表示为 (1/精度) * 360。在龙芯 2 号中, 这个值一般为 1/4, 即 8' h20 7:0: 控制内部 DLL 的精度, 在龙芯 2 号中, 这个值一般为 8' h80
DLL_CTRL_REG_0_5	31:0	0x0	0x0-0xffffffff	第 5 数据组 (DQ47-DQ40) DLL 控制信号 24: 控制内部 DLL 的使能信号, 为 0 时 DLL 有效 23:16: 控制写数据 (DQ) 与 DQS 之间的相位关系, 每个数值表示为 (1/精度) * 360。在龙芯 2 号中, 这个值一般为 1/4, 即 8' h20 7:0: 控制内部 DLL 的精度, 在龙芯 2 号中, 这个值一般为 8' h80

参数名称	位	初值	范围	描述
CONF_CTL_35[63:0] Offset: 0x230 DDR2 667 : 0x0020008000200080				
DLL_CTRL_REG_0.8	63:32	0x0	0x0-0xffffffff	第 8 数据组 (DQ71-DQ64) DLL 控制信号 24: 控制内部 DLL 的使能信号, 为 0 时 DLL 有效 23:16: 控制写数据 (DQ) 与 DQS 之间的相位关系, 每个数值表示为 (1/精度) * 360。在龙芯 2 号中, 这个值一般为 1/4, 即 8' h20 7:0: 控制内部 DLL 的精度, 在龙芯 2 号中, 这个值一般为 8' h80
DLL_CTRL_REG_0.7	31:0	0x0	0x0-0xffffffff	第 7 数据组 (DQ63-DQ56) DLL 控制信号 24: 控制内部 DLL 的使能信号, 为 0 时 DLL 有效 23:16: 控制写数据 (DQ) 与 DQS 之间的相位关系, 每个数值表示为 (1/精度) * 360。在龙芯 2 号中, 这个值一般为 1/4, 即 8' h20 7:0: 控制内部 DLL 的精度, 在龙芯 2 号中, 这个值一般为 8' h80
CONF_CTL_36[63:0] Offset: 0x240 DDR2 667 : 0x00001e0000001e00				
DLL_CTRL_REG_1.1	63:32	0x0	0x0-0xffffffff	第 1 数据组 DLL 控制信号 15:8: 读数据返回时, DQS _n 的相位延迟。 5:0: DLL 测试控制信号, 正常情况下为 8' h0
DLL_CTRL_REG_1.0	31:0	0x0	0x0-0xffffffff	第 0 数据组 DLL 控制信号 15:8: 读数据返回时, DQS _n 的相位延迟。 5:0: DLL 测试控制信号, 正常情况下为 8' h0
CONF_CTL_37[63:0] Offset: 0x250 DDR2 667 : 0x00001e0000001e00				
DLL_CTRL_REG_1.3	63:32	0x0	0x0-0xffffffff	第 3 数据组 DLL 控制信号 15:8: 读数据返回时, DQS _n 的相位延迟。 5:0: DLL 测试控制信号, 正常情况下为 8' h0
DLL_CTRL_REG_1.2	31:0	0x0	0x0-0xffffffff	第 2 数据组 DLL 控制信号 15:8: 读数据返回时, DQS _n 的相位延迟。 5:0: DLL 测试控制信号, 正常情况下为 8' h0
CONF_CTL_38[63:0] Offset: 0x260 DDR2 667 : 0x00001e0000001e00				
DLL_CTRL_REG_1.5	63:32	0x0	0x0-0xffffffff	第 5 数据组 DLL 控制信号 15:8: 读数据返回时, DQS _n 的相位延迟。 5:0: DLL 测试控制信号, 正常情况下为 8' h0
DLL_CTRL_REG_1.4	31:0	0x0	0x0-0xffffffff	第 4 数据组 DLL 控制信号 15:8: 读数据返回时, DQS _n 的相位延迟。5:0: DLL 测试控制信号, 正常情况下为 8' h0
CONF_CTL_39[63:0] Offset: 0x270 DDR2 667 : 0x00001e0000001e00				
DLL_CTRL_REG_1.7	63:32	0x0	0x0-0xffffffff	第 7 数据组 DLL 控制信号 15:8: 读数据返回时, DQS _n 的相位延迟。5:0: DLL 测试控制信号, 正常情况下为 8' h0.
DLL_CTRL_REG_1.6	31:0	0x0	0x0-0xffffffff	第 6 数据组 DLL 控制信号 15:8: 读数据返回时, DQS _n 的相位延迟。 5:0: DLL 测试控制信号, 正常情况下为 8' h0
CONF_CTL_40[63:0] Offset: 0x280 DDR2 667 : 0x0000000000001e00				
DLL_OBS_REG_0.0	33:32	0x0	0x0-0x3	测试模式下的第 0 数据组 DLL 输出 (只读)

参数名称	位	初值	范围	描述
DLL_CTRL_REG_1.8	31:0	0x0	0x0-0xffffffff	第 8 数据组 DLL 控制信号 15:8 : 读数据返回时, DQSn 的相位延迟。 5:0 : DLL 测试控制信号, 正常情况下为 8' h0
CONF_CTL_41[63:0] Offset: 0x290 DDR2 667 : 0x0000000000000000				
DLL_OBS_REG_0.2	33:32	0x0	0x0-0x3	测试模式下的第 2 数据组 DLL 输出 (只读)
DLL_OBS_REG_0.1	1:0	0x0	0x0-0x3	测试模式下的第 1 数据组 DLL 输出 (只读)
CONF_CTL_42[63:0] Offset: 0x2a0 DDR2 667 : 0x0x0000000000000000				
DLL_OBS_REG_0.4	33:32	0x0	0x0-0x3	测试模式下的第 4 数据组 DLL 输出 (只读)
DLL_OBS_REG_0.3	1:0	0x0	0x0-0x3	测试模式下的第 3 数据组 DLL 输出 (只读)
CONF_CTL_43[63:0] Offset: 0x2b0 DDR2 667 : 0x0x0000000000000000				
DLL_OBS_REG_0.6	33:32	0x0	0x0-0x3	测试模式下的第 6 数据组 DLL 输出 (只读)
DLL_OBS_REG_0.5	1:0	0x0	0x0-0x3	测试模式下的第 5 数据组 DLL 输出 (只读)
CONF_CTL_44[63:0] Offset: 0x2c0 DDR2 667 : 0x0000000000000000				
DLL_OBS_REG_0.8	33:32	0x0	0x0-0x3	测试模式下的第 8 数据组 DLL 输出 (只读)
DLL_OBS_REG_0.7	1:0	0x0	0x0-0x3	测试模式下的第 7 数据组 DLL 输出 (只读)
CONF_CTL_45[63:0] Offset: 0x2d0 DDR2 667 : 0xf30029470000019d				
PHY_CTRL_REG_0.0	63:32	0x0	0x0-0xffffffff	第 0 数据组时延控制。 28 : 是否对读 DQS 使用去毛刺电路, 指 gate 信号是否通过 PAD_feedback 延迟 27 : 使用读 FIFO 有效信号自动控制读数据返回采样 (1), 还是使用 26:24 中的固定时间采样 (0) 26:24 : 读数据返回采样完成时机, 从内部时钟域采样的延迟。 21 : 在 Read Leveling 模式下, 采样数据总线的电平高低 20 : 数据有效控制信号的电平, 龙芯 2 号中为 0 19 : 是否将写数据延迟再增加一周期 18 : 读 DQS 采样是否提前 1/4 周期 (与 clk_wr 同步) 17 : 写数据 /DQS 延迟是否增加半周期延迟 16 : CAS 延迟是否为半周期 15:12 : 写 DQS 有效的起始时间, 对于 DDR3 应该比 DDR2 提前一个周期打开, 提供颗粒要求的 Preamble DQS 11:8 : 写 DQS 有效的结束时间 6:4 : 写数据有效的起始时间 2:0 : 写数据有效的结束时间

参数名称	位	初值	范围	描述																
PAD_CTRL_REG_0	25:0	0x0	0x0- 0x3ffff	<p>引脚控制信号</p> <p>25:22 : 对应 COMPZCP_dig</p> <p>21:18 : 对应 COMPZCN_dig</p> <p>17 : 对应引脚的 TQ1v8</p> <p>16 : 对应内部反馈引脚的使能信号, 低有效</p> <p>15 : 对应内部反馈引脚的输出使能信号, 低有效</p> <p>14 : 对应数据选通引脚的输出使能信号, 低有效</p> <p>13 : 对应数据屏蔽引脚的输出使能信号, 低有效</p> <p>12 : 对应数据引脚的输出使能信号, 低有效</p> <p>11 : 对应引脚的 USEPAD</p> <p>0 : 使用内部参考电压;</p> <p>1 : 使用外部参考电压。</p> <p>8 : 对应时钟引脚1,3,5的使能信号, 高有效</p> <p>7 : 对应时钟引脚0,2,4的使能信号, 高有效</p> <p>6 : 对应地址引脚的使能信号, 低有效</p> <p>5 : 对应引脚的 PROGB1v8</p> <p>4 : 对应引脚的 PROGA1v8</p> <p>用于控制引脚驱动能力</p> <p>3 : 对应引脚的 ODTB</p> <p>2 : 对应引脚的 ODTA</p> <p>用于控制引脚 ODT 阻值大小</p> <table border="1"> <thead> <tr> <th>ODTA</th> <th>ODTB</th> <th>DDRII</th> <th>DDRIII</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>0</td> <td>150</td> <td>120</td> </tr> <tr> <td>1</td> <td>1</td> <td>5</td> <td>60</td> </tr> <tr> <td>0</td> <td>0</td> <td>Disable</td> <td>Disable</td> </tr> </tbody> </table> <p>1 : 对应引脚的 MODEZI1v8</p> <p>对于龙芯 2 号应该设为 0。</p> <p>0 : 对应引脚的 DDR1v8</p> <p>0 : 对应 DDRII 的 1.8v 模式</p> <p>1 : 对应 DDRIII 的 1.5v 模式</p>	ODTA	ODTB	DDRII	DDRIII	1	0	150	120	1	1	5	60	0	0	Disable	Disable
ODTA	ODTB	DDRII	DDRIII																	
1	0	150	120																	
1	1	5	60																	
0	0	Disable	Disable																	
CONF_CTL_46[63:0] Offset: 0x2e0 DDR2 667 : 0xf3002947f3002947																				

参数名称	位	初值	范围	描述
PHY_CTRL_REG_0_2	63:32	0x0	0x0- 0xffffffff	<p>第 2 数据组时延控制.</p> <p>28 : 是否对读 DQS 使用去毛刺电路, 指 gate 信号是否通过 PAD_feedback 延迟</p> <p>27 : 使用读 FIFO 有效信号自动控制读数据返回采样 (1), 还是使用 26:24 中的固定时间采样 (0)</p> <p>26:24 : 读数据返回采样完成时机, 从内部时钟域采样的延迟。</p> <p>21 : 在 Read Leveling 模式下, 采样数据总线的电平高低</p> <p>20 : 数据有效控制信号的电平, 龙芯 2 号中为 0</p> <p>19 : 是否将写数据延迟再增加一周期</p> <p>18 : 读 DQS 采样是否提前 1/4 周期 (与 clk.wr 同步)</p> <p>17 : 写数据 /DQS 延迟是否增加半周期延迟</p> <p>16 : CAS 延迟是否为半周期</p> <p>15:12 : 写 DQS 有效的起始时间, 对于 DDR3 应该比 DDR2 提前一个周期打开, 提供颗粒要求的 Preamble DQS</p> <p>11:8 : 写 DQS 有效的结束时间</p> <p>6:4 : 写数据有效的起始时间</p> <p>2:0 : 写数据有效的结束时间</p>
PHY_CTRL_REG_0_1	31:0	0x0	0x0- 0xffffffff	<p>第 1 数据组时延控制.</p> <p>28 : 是否对读 DQS 使用去毛刺电路, 指 gate 信号是否通过 PAD_feedback 延迟</p> <p>27 : 使用读 FIFO 有效信号自动控制读数据返回采样 (1), 还是使用 26:24 中的固定时间采样 (0)</p> <p>26:24 : 读数据返回采样完成时机, 从内部时钟域采样的延迟。</p> <p>21 : 在 Read Leveling 模式下, 采样数据总线的电平高低</p> <p>20 : 数据有效控制信号的电平, 龙芯 2 号中为 0</p> <p>19 : 是否将写数据延迟再增加一周期</p> <p>18 : 读 DQS 采样是否提前 1/4 周期 (与 clk.wr 同步)</p> <p>17 : 写数据 /DQS 延迟是否增加半周期延迟</p> <p>16 : CAS 延迟是否为半周期</p> <p>15:12 : 写 DQS 有效的起始时间, 对于 DDR3 应该比 DDR2 提前一个周期打开, 提供颗粒要求的 Preamble DQS</p> <p>11:8 : 写 DQS 有效的结束时间</p> <p>6:4 : 写数据有效的起始时间</p> <p>2:0 : 写数据有效的结束时间</p>
CONF_CTL_47[63:0] Offset: 0x2f0 DDR2 667 : 0xf3002947f3002947				

参数名称	位	初值	范围	描述
PHY_CTRL_REG_0_4	63:32	0x0	0x0- 0xffffffff	<p>第 4 数据组时延控制。</p> <p>28：是否对读 DQS 使用去毛刺电路，指 gate 信号是否通过 PAD_feedback 延迟</p> <p>27：使用读 FIFO 有效信号自动控制读数据返回采样（1），还是使用 26:24 中的固定时间采样（0）</p> <p>26:24：读数据返回采样完成时机，从内部时钟域采样的延迟。</p> <p>21：在 Read Leveling 模式下，采样数据总线的电平高低</p> <p>20：数据有效控制信号的电平，龙芯 2 号中为 0</p> <p>19：是否将写数据延迟再增加一周期</p> <p>18：读 DQS 采样是否提前 1/4 周期（与 clk_wr 同步）</p> <p>17：写数据 /DQS 延迟是否增加半周期延迟</p> <p>16：CAS 延迟是否为半周期</p> <p>15:12：写 DQS 有效的起始时间，对于 DDR3 应该比 DDR2 提前一个周期打开，提供颗粒要求的 Preamble DQS</p> <p>11:8：写 DQS 有效的结束时间</p> <p>6:4：写数据有效的起始时间</p> <p>2:0：写数据有效的结束时间</p>
PHY_CTRL_REG_0_3	31:0	0x0	0x0- 0xffffffff	<p>第 3 数据组时延控制。</p> <p>28：是否对读 DQS 使用去毛刺电路，指 gate 信号是否通过 PAD_feedback 延迟</p> <p>27：使用读 FIFO 有效信号自动控制读数据返回采样（1），还是使用 26:24 中的固定时间采样（0）</p> <p>26:24：读数据返回采样完成时机，从内部时钟域采样的延迟。</p> <p>21：在 Read Leveling 模式下，采样数据总线的电平高低</p> <p>20：数据有效控制信号的电平，龙芯 2 号中为 0</p> <p>19：是否将写数据延迟再增加一周期</p> <p>18：读 DQS 采样是否提前 1/4 周期（与 clk_wr 同步）</p> <p>17：写数据 /DQS 延迟是否增加半周期延迟</p> <p>16：CAS 延迟是否为半周期</p> <p>15:12：写 DQS 有效的起始时间，对于 DDR3 应该比 DDR2 提前一个周期打开，提供颗粒要求的 Preamble DQS</p> <p>11:8：写 DQS 有效的结束时间</p> <p>6:4：写数据有效的起始时间</p> <p>2:0：写数据有效的结束时间</p>
CONF_CTL_48[63:0] Offset: 0x300 DDR2 667 : 0xf3002947f3002947				

参数名称	位	初值	范围	描述
PHY_CTRL_REG_0_6	63:32	0x0	0x0- 0xffffffff	<p>第 6 数据组时延控制.</p> <p>28 : 是否对读 DQS 使用去毛刺电路, 指 gate 信号是否通过 PAD_feedback 延迟</p> <p>27 : 使用读 FIFO 有效信号自动控制读数据返回采样 (1), 还是使用 26:24 中的固定时间采样 (0)</p> <p>26:24 : 读数据返回采样完成时机, 从内部时钟域采样的延迟。</p> <p>21 : 在 Read Leveling 模式下, 采样数据总线的电平高低</p> <p>20 : 数据有效控制信号的电平, 龙芯 2 号中为 0</p> <p>19 : 是否将写数据延迟再增加一周期</p> <p>18 : 读 DQS 采样是否提前 1/4 周期 (与 clk_wr 同步)</p> <p>17 : 写数据 /DQS 延迟是否增加半周期延迟</p> <p>16 : CAS 延迟是否为半周期</p> <p>15:12 : 写 DQS 有效的起始时间, 对于 DDR3 应该比 DDR2 提前一个周期打开, 提供颗粒要求的 Preamble DQS</p> <p>11:8 : 写 DQS 有效的结束时间</p> <p>6:4 : 写数据有效的起始时间</p> <p>2:0 : 写数据有效的结束时间</p>
PHY_CTRL_REG_0_5	31:0	0x0	0x0- 0xffffffff	<p>第 5 数据组时延控制.</p> <p>28 : 是否对读 DQS 使用去毛刺电路, 指 gate 信号是否通过 PAD_feedback 延迟</p> <p>27 : 使用读 FIFO 有效信号自动控制读数据返回采样 (1), 还是使用 26:24 中的固定时间采样 (0)</p> <p>26:24 : 读数据返回采样完成时机, 从内部时钟域采样的延迟。</p> <p>21 : 在 Read Leveling 模式下, 采样数据总线的电平高低</p> <p>20 : 数据有效控制信号的电平, 龙芯 2 号中为 0</p> <p>19 : 是否将写数据延迟再增加一周期</p> <p>18 : 读 DQS 采样是否提前 1/4 周期 (与 clk_wr 同步)</p> <p>17 : 写数据 /DQS 延迟是否增加半周期延迟</p> <p>16 : CAS 延迟是否为半周期</p> <p>15:12 : 写 DQS 有效的起始时间, 对于 DDR3 应该比 DDR2 提前一个周期打开, 提供颗粒要求的 Preamble DQS</p> <p>11:8 : 写 DQS 有效的结束时间</p> <p>6:4 : 写数据有效的起始时间</p> <p>2:0 : 写数据有效的结束时间</p>
CONF_CTL_49[63:0] Offset: 0x310 DDR2 667 : 0xf3002947f3002947				

参数名称	位	初值	范围	描述
PHY_CTRL_REG_0_8	63:32	0x0	0x0- 0xffffffff	<p>第 8 数据组时延控制。</p> <p>28：是否对读 DQS 使用去毛刺电路，指 gate 信号是否通过 PAD_feedback 延迟</p> <p>27：使用读 FIFO 有效信号自动控制读数据返回采样（1），还是使用 26:24 中的固定时间采样（0）</p> <p>26:24：读数据返回采样完成时机，从内部时钟域采样的延迟。</p> <p>21：在 Read Leveling 模式下，采样数据总线的电平高低</p> <p>20：数据有效控制信号的电平，龙芯 2 号中为 0</p> <p>19：是否将写数据延迟再增加一周期</p> <p>18：读 DQS 采样是否提前 1/4 周期（与 clk_wr 同步）</p> <p>17：写数据 /DQS 延迟是否增加半周期延迟</p> <p>16：CAS 延迟是否为半周期</p> <p>15:12：写 DQS 有效的起始时间，对于 DDR3 应该比 DDR2 提前一个周期打开，提供颗粒要求的 Preamble DQS</p> <p>11:8：写 DQS 有效的结束时间</p> <p>6:4：写数据有效的起始时间</p> <p>2:0：写数据有效的结束时间</p>
PHY_CTRL_REG_0_7	31:0	0x0	0x0- 0xffffffff	<p>第 7 数据组时延控制。</p> <p>28：是否对读 DQS 使用去毛刺电路，指 gate 信号是否通过 PAD_feedback 延迟</p> <p>27：使用读 FIFO 有效信号自动控制读数据返回采样（1），还是使用 26:24 中的固定时间采样（0）</p> <p>26:24：读数据返回采样完成时机，从内部时钟域采样的延迟。</p> <p>21：在 Read Leveling 模式下，采样数据总线的电平高低</p> <p>20：数据有效控制信号的电平，龙芯 2 号中为 0</p> <p>19：是否将写数据延迟再增加一周期</p> <p>18：读 DQS 采样是否提前 1/4 周期（与 clk_wr 同步）</p> <p>17：写数据 /DQS 延迟是否增加半周期延迟</p> <p>16：CAS 延迟是否为半周期</p> <p>15:12：写 DQS 有效的起始时间，对于 DDR3 应该比 DDR2 提前一个周期打开，提供颗粒要求的 Preamble DQS</p> <p>11:8：写 DQS 有效的结束时间</p> <p>6:4：写数据有效的起始时间</p> <p>2:0：写数据有效的结束时间</p>
CONF_CTL_50[63:0] Offset: 0x320 DDR2 667 : 0x07c0000007c00000				

参数名称	位	初值	范围	描述
PHY_CTRL_REG_1_1	63:32	0x0	0x0- 0xffffffff	<p>第 1 数据组中 PAD 的终端电阻控制，发起读操作时，才会启用</p> <p>31:28：终端电阻关闭时机控制，每个值表示半周期</p> <p>27:24：终端电阻开启时机控制，从发送读命令后 4 拍开始计算</p> <p>23：终端电阻的有效电平控制，对于龙芯 2 号应为 1</p> <p>22：终端电阻的使能信号，为 1 时，使用动态方式控制终端电阻的使能；为 0 时，可以通过第 23 位 PAD 上的终端电阻永远有效（置 0）或永远无效（置 1）</p> <p>21：测试用信号，正常应为 0</p> <p>20:16：测试用信号，正常应为 0</p> <p>14:12：测试用信号，正常应为 0</p> <p>11:8：读采样延时 1，其中只能 1 位有效，用于控制读 DQS 采样窗口关闭时机</p> <p>7:0：读采样延时 0，其中只能 1 位有效，用于控制读 DQS 采样窗口打开时机</p>
PHY_CTRL_REG_1_0	31:0	0x0	0x0- 0xffffffff	<p>第 0 数据组中 PAD 的终端电阻控制，发起读操作时，才会启用</p> <p>31:28：终端电阻关闭时机控制，每个值表示半周期</p> <p>27:24：终端电阻开启时机控制，从发送读命令后 4 拍开始计算</p> <p>23：终端电阻的有效电平控制，对于龙芯 2 号应为 1</p> <p>22：终端电阻的使能信号，为 1 时，使用动态方式控制终端电阻的使能；为 0 时，可以通过第 23 位 PAD 上的终端电阻永远有效（置 0）或永远无效（置 1）</p> <p>21：测试用信号，正常应为 0</p> <p>20:16：测试用信号，正常应为 0</p> <p>14:12：测试用信号，正常应为 0</p> <p>11:8：读采样延时 1，其中只能 1 位有效，用于控制读 DQS 采样窗口关闭时机</p> <p>7:0：读采样延时 0，其中只能 1 位有效，用于控制读 DQS 采样窗口打开时机</p>
CONF_CTL_51[63:0] Offset: 0x330 DDR2 667 : 0x07c0000007c00000				

参数名称	位	初值	范围	描述
PHY_CTRL_REG_1.3	63:32	0x0	0x0-0xffffffff	<p>第 3 数据组中 PAD 的终端电阻控制，发起读操作时，才会启用</p> <p>31:28：终端电阻关闭时机控制，每个值表示半周期</p> <p>27:24：终端电阻开启时机控制，从发送读命令后 4 拍开始计算</p> <p>23：终端电阻的有效电平控制，对于龙芯 2 号应为 1</p> <p>22：终端电阻的使能信号，为 1 时，使用动态方式控制终端电阻的使能；为 0 时，可以通过第 23 位 PAD 上的终端电阻永远有效（置 0）或永远无效（置 1）</p> <p>21：测试用信号，正常应为 0</p> <p>20:16：测试用信号，正常应为 0</p> <p>14:12：测试用信号，正常应为 0</p> <p>11:8：读采样延时 1，其中只能 1 位有效，用于控制读 DQS 采样窗口关闭时机</p> <p>7:0：读采样延时 0，其中只能 1 位有效，用于控制读 DQS 采样窗口打开时机</p>
PHY_CTRL_REG_1.2	31:0	0x0	0x0-0xffffffff	<p>第 2 数据组中 PAD 的终端电阻控制，发起读操作时，才会启用</p> <p>31:28：终端电阻关闭时机控制，每个值表示半周期</p> <p>27:24：终端电阻开启时机控制，从发送读命令后 4 拍开始计算</p> <p>23：终端电阻的有效电平控制，对于龙芯 2 号应为 1</p> <p>22：终端电阻的使能信号，为 1 时，使用动态方式控制终端电阻的使能；为 0 时，可以通过第 23 位 PAD 上的终端电阻永远有效（置 0）或永远无效（置 1）</p> <p>21：测试用信号，正常应为 0</p> <p>20:16：测试用信号，正常应为 0</p> <p>14:12：测试用信号，正常应为 0</p> <p>11:8：读采样延时 1，其中只能 1 位有效，用于控制读 DQS 采样窗口关闭时机</p> <p>7:0：读采样延时 0，其中只能 1 位有效，用于控制读 DQS 采样窗口打开时机</p>
CONF_CTL_52[63:0] Offset: 0x340 DDR2 667 : 0x07c0000007c00000				

参数名称	位	初值	范围	描述
PHY_CTRL_REG_1_5	63:32	0x0	0x0-0xffffffff	<p>第 5 数据组中 PAD 的终端电阻控制，发起读操作时，才会启用</p> <p>31:28：终端电阻关闭时机控制，每个值表示半周期</p> <p>27:24：终端电阻开启时机控制，从发送读命令后 4 拍开始计算</p> <p>23：终端电阻的有效电平控制，对于龙芯 2 号应为 1</p> <p>22：终端电阻的使能信号，为 1 时，使用动态方式控制终端电阻的使能；为 0 时，可以通过第 23 位 PAD 上的终端电阻永远有效（置 0）或永远无效（置 1）</p> <p>21：测试用信号，正常应为 0</p> <p>20:16：测试用信号，正常应为 0</p> <p>14:12：测试用信号，正常应为 0</p> <p>11:8：读采样延时 1，其中只能 1 位有效，用于控制读 DQS 采样窗口关闭时机</p> <p>7:0：读采样延时 0，其中只能 1 位有效，用于控制读 DQS 采样窗口打开时机</p>
PHY_CTRL_REG_1_4	31:0	0x0	0x0-0xffffffff	<p>第 4 数据组中 PAD 的终端电阻控制，发起读操作时，才会启用</p> <p>31:28：终端电阻关闭时机控制，每个值表示半周期</p> <p>27:24：终端电阻开启时机控制，从发送读命令后 4 拍开始计算</p> <p>23：终端电阻的有效电平控制，对于龙芯 2 号应为 1</p> <p>22：终端电阻的使能信号，为 1 时，使用动态方式控制终端电阻的使能；为 0 时，可以通过第 23 位 PAD 上的终端电阻永远有效（置 0）或永远无效（置 1）</p> <p>21：测试用信号，正常应为 0</p> <p>20:16：测试用信号，正常应为 0</p> <p>14:12：测试用信号，正常应为 0</p> <p>11:8：读采样延时 1，其中只能 1 位有效，用于控制读 DQS 采样窗口关闭时机</p> <p>7:0：读采样延时 0，其中只能 1 位有效，用于控制读 DQS 采样窗口打开时机</p>
CONF_CTL_53[63:0] Offset: 0x350 DDR2 667 : 0x07c0000007c00000				

参数名称	位	初值	范围	描述
PHY_CTRL_REG_1.7	63:32	0x0	0x0-0xffffffff	<p>第 7 数据组中 PAD 的终端电阻控制，发起读操作时，才会启用</p> <p>31:28：终端电阻关闭时机控制，每个值表示半周期</p> <p>27:24：终端电阻开启时机控制，从发送读命令后 4 拍开始计算</p> <p>23：终端电阻的有效电平控制，对于龙芯 2 号应为 1</p> <p>22：终端电阻的使能信号，为 1 时，使用动态方式控制终端电阻的使能；为 0 时，可以通过第 23 位 PAD 上的终端电阻永远有效（置 0）或永远无效（置 1）</p> <p>21：测试用信号，正常应为 0</p> <p>20:16：测试用信号，正常应为 0</p> <p>14:12：测试用信号，正常应为 0</p> <p>11:8：读采样延时 1，其中只能 1 位有效，用于控制读 DQS 采样窗口关闭时机</p> <p>7:0：读采样延时 0，其中只能 1 位有效，用于控制读 DQS 采样窗口打开时机</p>
PHY_CTRL_REG_1.6	31:0	0x0	0x0-0xffffffff	<p>第 6 数据组中 PAD 的终端电阻控制，发起读操作时，才会启用</p> <p>31:28：终端电阻关闭时机控制，每个值表示半周期</p> <p>27:24：终端电阻开启时机控制，从发送读命令后 4 拍开始计算</p> <p>23：终端电阻的有效电平控制，对于龙芯 2 号应为 1</p> <p>22：终端电阻的使能信号，为 1 时，使用动态方式控制终端电阻的使能；为 0 时，可以通过第 23 位 PAD 上的终端电阻永远有效（置 0）或永远无效（置 1）</p> <p>21：测试用信号，正常应为 0</p> <p>20:16：测试用信号，正常应为 0</p> <p>14:12：测试用信号，正常应为 0</p> <p>11:8：读采样延时 1，其中只能 1 位有效，用于控制读 DQS 采样窗口关闭时机</p> <p>7:0：读采样延时 0，其中只能 1 位有效，用于控制读 DQS 采样窗口打开时机</p>
CONF_CTL_54[63:0] Offset: 0x360 DDR2 667 : 0x0800c00507c00000				

参数名称	位	初值	范围	描述
PHY_CTRL_REG_2	63:32	0x0	0x0- 0xffffffff	读写数据延迟控制 27：选择读数据缓冲类型，默认为 0 26：用于清除读返回缓冲区的数据，正常为 0 25：高速引脚使能，为 1 时，所有信号通过引脚向外传输的延迟减小 1 周期 16:13：设置读数据有效时机，从 FIFO 中收集数据返回控制器的延迟。如果从引脚到 FIFO 的延迟增加，这个值也必须增加 8：设置 DQS 信号输出是否为 DDR3 模式，DDR3 模式下，写 DQS 的 Preamble 将含有一个脉冲 5：测试模式信号，正常为 0 4：测试模式信号，正常为 0
PHY_CTRL_REG_1_8	31:0	0x0	0x0- 0xffffffff	第 8 数据组中的终端电阻控制 PAD 的终端电阻控制，发起读操作时，才会启用 31:28：终端电阻关闭时机控制，每个值表示半周期 27:24：终端电阻开启时机控制，从发送读命令后 4 拍开始计算 23：终端电阻的有效电平控制，对于龙芯 2 号应为 1 22：终端电阻的使能信号，为 1 时，使用动态方式控制终端电阻的使能；为 0 时，可以通过第 23 位 PAD 上的终端电阻永远有效（置 0）或永远无效（置 1） 21：测试用信号，正常应为 0 20:16：测试用信号，正常应为 0 14:12：测试用信号，正常应为 0 11:8：读采样延时 1，其中只能 1 位有效，用于控制读 DQS 采样窗口关闭时机 7:0：读采样延时 0，其中只能 1 位有效，用于控制读 DQS 采样窗口打开时机
CONF_CTL_55[63:0] Offset: 0x370 DDR2 667 : 0x0000000000000000				
PHY_OBS_REG_0_1	63:32	0x0	0x0- 0xffffffff	第 1 数据组测试用观测信号（只读）
PHY_OBS_REG_0_0	31:0	0x0	0x0- 0xffffffff	第 0 数据组测试用观测信号（只读）
CONF_CTL_56[63:0] Offset: 0x380 DDR2 667 : 0x0000000000000000				
PHY_OBS_REG_0_3	63:32	0x0	0x0- 0xffffffff	第 3 数据组测试用观测信号（只读）
PHY_OBS_REG_0_2	31:0	0x0	0x0- 0xffffffff	第 2 数据组测试用观测信号（只读）
CONF_CTL_57[63:0] Offset: 0x390 DDR2 667 : 0x0000000000000000				
PHY_OBS_REG_0_5	63:32	0x0	0x0- 0xffffffff	第 5 数据组测试用观测信号（只读）
PHY_OBS_REG_0_4	31:0	0x0	0x0- 0xffffffff	第 4 数据组测试用观测信号（只读）
CONF_CTL_58[63:0] Offset: 0x3a0 DDR2 667 : 0x0000000000000000				

参数名称	位	初值	范围	描述
PHY_OBS_REG_0.7	63:32	0x0	0x0-0xffffffff	第 7 数据组测试用观测信号 (只读)
PHY_OBS_REG_0.6	31:0	0x0	0x0-0xffffffff	第 6 数据组测试用观测信号 (只读)
CONF_CTL_59[63:0] Offset: 0x3b0 DDR2 667 : 0x0000000000000000				
PHY_OBS_REG_0.8	31:0	0x0	0x0-0xffffffff	第 8 数据组测试用观测信号 (只读)
CONF_CTL_114[63:0] Offset: 0x720 DDR2 667 : 0x0000000000000000				
RDLVL_GATE_REQ	56	0x0	0x0-0x1	用户请求读选通采样训练功能。(只写)
RDLVL_GATE_PREAMBLE_CHECK_EN	55	0x0	0x0-0x1	开启读选通采样训练时的前导采样检查
RDLVL_GATE_EN	40	0x0	0x0-0x1	使能 Read Leveling 时读选通采样训练, 在初始化完成后会向颗粒发送命令, 进行读 DQS 采样窗口的训练
RDLVL_EN	32	0x0	0x0-0x1	使能 Read Leveling 功能
RDLVL_BEGIN_DELAY_EN	24	0x0	0x0-0x1	使能 Read Leveling 寻找数据采样点功能
SWLVL_OP_DONE	8	0x0	0x0-0x1	用于指示软件 Leveling 是否完成 (只读)
CONF_CTL_115[63:0] Offset: 0x730 DDR2 667 : 0x0000000000000000				
RDLVL_OFFSET_DIR_7	56	0x0	0x0-0x1	第 7 数据组 Read Leveling 时中点的调整方向。为 0 时, 中点计算为减去 rdlvl_offset_delay, 为 1 则加。
RDLVL_OFFSET_DIR_6	48	0x0	0x0-0x1	第 6 数据组 Read Leveling 时中点的调整方向。为 0 时, 中点计算为减去 rdlvl_offset_delay, 为 1 则加。
RDLVL_OFFSET_DIR_5	40	0x0	0x0-0x1	第 5 数据组 Read Leveling 时中点的调整方向。为 0 时, 中点计算为减去 rdlvl_offset_delay, 为 1 则加。
RDLVL_OFFSET_DIR_4	32	0x0	0x0-0x1	第 4 数据组 Read Leveling 时中点的调整方向。为 0 时, 中点计算为减去 rdlvl_offset_delay, 为 1 则加。
RDLVL_OFFSET_DIR_3	24	0x0	0x0-0x1	第 3 数据组 Read Leveling 时中点的调整方向。为 0 时, 中点计算为减去 rdlvl_offset_delay, 为 1 则加。
RDLVL_OFFSET_DIR_2	16	0x0	0x0-0x1	第 2 数据组 Read Leveling 时中点的调整方向。为 0 时, 中点计算为减去 rdlvl_offset_delay, 为 1 则加。
RDLVL_OFFSET_DIR_1	8	0x0	0x0-0x1	第 1 数据组 Read Leveling 时中点的调整方向。为 0 时, 中点计算为减去 rdlvl_offset_delay, 为 1 则加。
RDLVL_OFFSET_DIR_0	0	0x0	0x0-0x1	第 0 数据组 Read Leveling 时中点的调整方向。为 0 时, 中点计算为减去 rdlvl_offset_delay, 为 1 则加。
CONF_CTL_116[63:0] Offset: 0x740 DDR2 667 : 0x0100000000000000				
AXI1_PORT_ORDERING	57:56	0x0	0x0-0x3	内部端口 1 是否可乱序执行, 对于龙芯 2 号无效
AXI0_PORT_ORDERING	49:48	0x0	0x0-0x3	内部端口 0 是否可乱序执行
WRLVL_REQ	40	0x0	0x0-0x1	用户请求开始 Write Leveling 训练功能。(只写)
WRLVL_INTERVAL_CT_EN	32	0x0	0x0-0x1	使能 Write Leveling 时间间隔功能
WEIGHTED_ROUND_ROBIN_HIGH_PRIORITY	24	0x0	0x0-0x1	Per-port pair shared arbitration for WRR

参数名称	位	初值	范围	描述
WEIGHTED_ROUND_ROBIN_LATENCY_CONTROL	16:0	0x0	0x0-0x1	Free-running or limited WRR latency counters.
RDLVL_REQ	8	0x0	0x0-0x1	用户请求开始 Read Leveling 训练功能。(只写)
RDLVL_OFFSET_DIR_8	0	0x0	0x0-0x1	第 8 数据组 Read Leveling 时中点的调整方向。为 0 时, 中点计算为减去 rdlvl_offset_delay, 为 1 则加。
CONF_CTL_117[63:0] Offset: 0x750 DDR2 667 : 0x0100000101020101				
WRLVL_CS	57:56	0x0	0x0-0x3	指示当前 Write Leveling 操作的片选信号
SW_LEVELING_MODE	49:48	0x0	0x0-0x3	定义软件 Leveling 操作的模式
RDLVL_CS	41:40	0x0	0x0-0x3	指示当前 Read Leveling 操作的片选信号
AXI2_W_PRIORITY	33:32	0x0	0x0-0x3	内部端口 2 的写操作优先级, 对于龙芯 2 号无效
AXI2_R_PRIORITY	25:24	0x0	0x0-0x3	内部端口 2 的读操作优先级, 对于龙芯 2 号无效
AXI2_PORT_ORDERING	17:16	0x0	0x0-0x3	内部端口 2 是否可乱序执行, 对于龙芯 2 号无效
AXI1_W_PRIORITY	9:8	0x0	0x0-0x3	内部端口 1 的写操作优先级, 对于龙芯 2 号无效
AXI1_R_PRIORITY	1:0	0x0	0x0-0x3	内部端口 1 的读操作优先级, 对于龙芯 2 号无效
CONF_CTL_118[63:0] Offset: 0x760 DDR2 667 : 0x0303030000020002				
AXI0_PRIORITY2_RELATIVE_PRIORITY	39:30	0x0	0x0-0xf	内部端口 0 优先级 2 的命令的相对优先级
AXI0_PRIORITY1_RELATIVE_PRIORITY	31:22	0x0	0x0-0xf	内部端口 0 优先级 1 的命令的相对优先级
AXI0_PRIORITY0_RELATIVE_PRIORITY	23:14	0x0	0x0-0xf	内部端口 0 优先级 0 的命令的相对优先级
ADDRESS_MIRRORING	35:32	0x0	0x0-0xf	指示哪个片选支持 Address mirroring 功能
TDFI_DRAM_CLK_DISABLE	26:24	0x0	0x0-0x7	从内部时钟关闭到外部时钟关闭的延迟设置
BSTLEN	18:16	0x0	0x0-0x7	设置控制器上向内存模块发送的 Burst 长度值
ZQ_REQ	9:8	0x0	0x0-0x3	用户请求开始 ZQ 调整功能
ZQ_ON_SREF_EXIT	1:0	0x0	0x0-0x3	定义在退出自刷新模式时 ZQ 调整功能的模式
CONF_CTL_119[63:0] Offset: 0x770 DDR2 667 : 0x0101010202020203				
AXI2_PRIORITY2_RELATIVE_PRIORITY	39:30	0x0	0x0-0xf	内部端口 2 优先级 2 的命令的相对优先级, 对于龙芯 2 号无效
AXI2_PRIORITY1_RELATIVE_PRIORITY	31:22	0x0	0x0-0xf	内部端口 2 优先级 1 的命令的相对优先级, 对于龙芯 2 号无效
AXI2_PRIORITY0_RELATIVE_PRIORITY	23:14	0x0	0x0-0xf	内部端口 2 优先级 0 的命令的相对优先级, 对于龙芯 2 号无效
AXI1_PRIORITY3_RELATIVE_PRIORITY	31:22	0x0	0x0-0xf	内部端口 1 优先级 3 的命令的相对优先级, 对于龙芯 2 号无效
AXI1_PRIORITY2_RELATIVE_PRIORITY	23:14	0x0	0x0-0xf	内部端口 1 优先级 2 的命令的相对优先级, 对于龙芯 2 号无效
AXI1_PRIORITY1_RELATIVE_PRIORITY	15:06	0x0	0x0-0xf	内部端口 1 优先级 1 的命令的相对优先级, 对于龙芯 2 号无效
AXI1_PRIORITY0_RELATIVE_PRIORITY	07:00	0x0	0x0-0xf	内部端口 1 优先级 0 的命令的相对优先级, 对于龙芯 2 号无效
AXI0_PRIORITY3_RELATIVE_PRIORITY	31:22	0x0	0x0-0xf	内部端口 0 优先级 3 的命令的相对优先级
CONF_CTL_120[63:0] Offset: 0x780 DDR2 667 : 0x0102020400040c01				
TDFI_DRAM_CLK_ENABLE	59:56	0x0	0x0-0xf	从内部时钟有效到输出时钟有效的延迟
TDFI_CTRL_DELAY	51:48	0x0	0x0-0xf	从时钟有效到输出命令之间的延迟
RDLVL_GATE_DQ_ZERO_COUNT	43:40	0x0	0x0-0xf	设置读选通采样训练时, 表求由 1 到 0 的 0 的个数
RDLVL_DQ_ZERO_COUNT	35:32	0x0	0x0-0xf	设置读 Read Leveling 时, 表求由 1 到 0 的 0 的个数

参数名称	位	初值	范围	描述
LOWPOWER_REFRESH_ENABLE	27:24	0x0	0x0-0xf	使能低功耗模式下的刷新功能
DRAM_CLASS	19:16	0x0	0x0-0xf	定义控制器外接内存类型
110 : DDR3 100 : DDR2				
BURST_ON_FLY_BIT	11:8	0x0	0x0-0xf	对 DRAM 发出的模式配置中的 burst-on-fly 位
AXI2_PRIORITY3_RELATIVE_PRIORITY	10:0	0x0	0x0-0xf	内部端口 2 优先级 3 的命令的相对优先级, 对于龙芯 2 号无效
CONF_CTL_121[63:0] Offset: 0x790 DDR2 667 : 0x281900000f000303				
WLMRD	61:56	0x0	0x0-0x3f	从对 DRAM 发送模式配置到 Write Leveling 的延迟
WLDQSEN	53:48	0x0	0x0-0x3f	从对 DRAM 发送模式配置到 Write Leveling 的选通数据采样延迟
LOWPOWER_CONTROL	44:40	0x0	0x0-0x1f	低功耗模式使能 Bit 4: power down Bit 3: power down external Bit 2: self refresh Bit 1: external Bit 0: internal
LOWPOWER_AUTO_ENABLE	36:32	0x0	0x0-0x1f	使能当控制器内闲时自动进入低功耗模式控制位与 LOWERPOWER_CONTROL 相同
ZQCS_CHIP	27:24	0x0	0x0-0xf	定义下次 ZQ 时的有效片选
WRR_PARAM.VALUE.ERR	19:16	0x0	0x0-0xf	Errors/warnings related to the WRR parameters. (只读)
TDFI_WRLVL_DLL	15:8	0x0	0x0-0xff	读操作到 Write Leveling 更新延迟线数目的最小周期
TDFI_RDLVL_DLL	7:0	0x0	0x0-0xff	读操作到 Read Leveling 更新延迟线数目的最小周期
CONF_CTL_122[63:0] Offset: 0x7a0 DDR2 667 : 0x0000000000000000				
SWLVL_RESP_6	63:56	0x0	0x0-0xff	第 6 数据组的 Leveling 响应
SWLVL_RESP_5	55:48	0x0	0x0-0xff	第 5 数据组的 Leveling 响应
SWLVL_RESP_4	47:40	0x0	0x0-0xff	第 4 数据组的 Leveling 响应
SWLVL_RESP_3	39:32	0x0	0x0-0xff	第 3 数据组的 Leveling 响应
SWLVL_RESP_2	31:24	0x0	0x0-0xff	第 2 数据组的 Leveling 响应
SWLVL_RESP_1	23:16	0x0	0x0-0xff	第 1 数据组的 Leveling 响应
SWLVL_RESP_0	15:8	0x0	0x0-0xff	第 0 数据组的 Leveling 响应
CONF_CTL_123[63:0] Offset: 0x7b0 DDR2 667 : 0x0000000000000000				
OBSOLETE	63:16			
SWLVL_RESP_8	15:8	0x0	0x0-0xff	第 8 数据组的 Leveling 响应
SWLVL_RESP_7	7:0	0x0	0x0-0xff	第 7 数据组的 Leveling 响应
CONF_CTL_124[63:0] Offset: 0x7c0 DDR2 667 : 0x0000000000000000				
OBSOLETE				
CONF_CTL_125[63:0] Offset: 0x7d0 DDR2 667 : 0x0000000000000000				
RDLVL_GATE_CLK_ADJUST_3	63:56	0x0	0x0-0xff	第 3 数据组中, 读采样训练的起始值
RDLVL_GATE_CLK_ADJUST_2	55:48	0x0	0x0-0xff	第 2 数据组中, 读采样训练的起始值
RDLVL_GATE_CLK_ADJUST_1	47:40	0x0	0x0-0xff	第 1 数据组中, 读采样训练的起始值
RDLVL_GATE_CLK_ADJUST_0	39:32	0x0	0x0-0xff	第 0 数据组中, 读采样训练的起始值
CONF_CTL_126[63:0] Offset: 0x7e0 DDR2 667 : 0x0000000000000000				

参数名称	位	初值	范围	描述
RDLVL_GATE_CLK_ADJUST_8	89:32	0x0	0x0-0xff	第 8 数据组中, 读采样训练的起始值
RDLVL_GATE_CLK_ADJUST_7	81:24	0x0	0x0-0xff	第 7 数据组中, 读采样训练的起始值
RDLVL_GATE_CLK_ADJUST_6	73:16	0x0	0x0-0xff	第 6 数据组中, 读采样训练的起始值
RDLVL_GATE_CLK_ADJUST_5	65:8	0x0	0x0-0xff	第 5 数据组中, 读采样训练的起始值
RDLVL_GATE_CLK_ADJUST_4	57:0	0x0	0x0-0xff	第 4 数据组中, 读采样训练的起始值
CONF_CTL_127[63:0] Offset: 0x7f0 DDR2 667 : 0x0000000000000000				
OBSOLETE				
CONF_CTL_128[63:0] Offset: 0x800 DDR2 667 : 0x0000000000000000				
OBSOLETE				
CONF_CTL_129[63:0] Offset: 0x810 DDR2 667 : 0x0000000000000000				
OBSOLETE				
CONF_CTL_130[63:0] Offset: 0x820 DDR2 667 : 0x0420000c20400000				
TDFL_WRLVL_RESPLAT	63:56	0x0	0x0-0xff	Write Leveling 选通到响应有效的周期数
TDFL_RDLVL_RESPLAT	39:32	0x0	0x0-0xff	Read Leveling 选通到响应有效的周期数
REFRESH_PER_ZQ	23:16	0x0	0x0-0xff	自动 ZQCS 命令之间的刷新命令数目
CONF_CTL_131[63:0] Offset: 0x830 DDR2 667 : 0x00000000000000c0a				
TMOD	15:8	0x0	0x0-0xff	DRAM 模式配置后需空闲的周期数
CONF_CTL_132[63:0] Offset: 0x840 DDR2 667 : 0x0000640064000000				
AXI1_PRIORITY_RELAX	49:40	0x0	0x0-0x3ff	内部端口 1 上触发优先控制放松的计数器值, 对于龙芯 2 号无效
AXI0_PRIORITY_RELAX [9:8]	33:32	0x0	0x0-0x3	内部端口 0 上触发优先控制放松的计数器值
AXI0_PRIORITY_RELAX [7:0]	31:24	0x0	0x0-0xff	内部端口 0 上触发优先控制放松的计数器值
CONF_CTL_133[63:0] Offset: 0x850 DDR2 667 : 0x0000000000000064				
OUT_OF_RANGE_SOURCE_ID	57:48	0x0	0x0-0x3ff	访问地址溢出请求的 ID 号 (只读)
ECC_U_ID	41:32	0x0	0x0-0x3ff	访问出现 2 位错请求的 ID 号 (只读)
ECC_C_ID	25:16	0x0	0x0-0x3ff	访问出现 1 位错请求的 ID 号 (只读)
AXI2_PRIORITY_RELAX	9:0	0x0	0x0-0x3ff	内部端口 2 上触发优先控制放松的计数器值, 对于龙芯 2 号无效
CONF_CTL_134[63:0] Offset: 0x860 DDR2 667 : 0x0000004000000000				
ZQCS	43:32	0x0	0x0-0xffff	ZQCS 命令需要的周期数
PORT_DATA_ERROR_ID	25:16	0x0	0x0-0x3ff	内部端口数据错请求的 ID 号 (只读)
PORT_CMD_ERROR_ID	9:0	0x0	0x0-0x3ff	内部端口命令错请求的 ID 号 (只读)
CONF_CTL_135[63:0] Offset: 0x870 DDR2 667 : 0x0000000000000000				
OBSOLETE				
CONF_CTL_136[63:0] Offset: 0x880 DDR2 667 : 0x0000000000000000				
OBSOLETE				
CONF_CTL_137[63:0] Offset: 0x890 DDR2 667 : 0x0000000000000000				
OBSOLETE				
CONF_CTL_138[63:0] Offset: 0x8a0 DDR2 667 : 0x00000000001c001c				
LOWPOWER_INTERNAL_CNTR	63:48	0x0	0x0-0xffff	Counts idle cycles to self-refresh with memory and controller clk gating.
LOWPOWER_EXTERNAL_CNTR	47:32	0x0	0x0-0xffff	Counts idle cycles to self-refresh with memory clock gating.

参数名称	位	初值	范围	描述
AXI2_EN_SIZE_LT_WIDTH_INSTR6	5:6	0x0	0x0-0xffff	使能内部端口 2 上的各种窄访问, 对于龙芯 2 号无效
AXI1_EN_SIZE_LT_WIDTH_INSTR6	5:6	0x0	0x0-0xffff	使能内部端口 1 上的各种窄访问, 对于龙芯 2 号无效
CONF_CTL_139[63:0] Offset: 0x8b0 DDR2 667 : 0x0000000000000000				
LOWPOWER_POWER_DOWN_CNT	16:16	0x0	0x0-0xffff	进入 Power Down 模式前的空闲周期数
LOWPOWER_REFRESH_HOLD_CNT	16:16	0x0	0x0-0xffff	在时钟门控模式下, 内存控制器 re-lock DLL 前的空闲周期数
LOWPOWER_SELF_REFRESH_HOLD_CNT	16:16	0x0	0x0-0xffff	进入内存自刷新模式前的空闲周期数
CONF_CTL_140[63:0] Offset: 0x8c0 DDR2 667 : 0x0004000000000000				
OBSOLETE				
CONF_CTL_141[63:0] Offset: 0x8d0 DDR2 667 : 0x00000000c8000000				
CKE_INACTIVE[31:8]	55:32	0x0	0x0-0xffffffff	从输出 DDR_RESET 有效到 CKE 有效的时间间隔高位
CKE_INACTIVE [7:0]	31:24	0x0	0x0-0xff	从输出 DDR_RESET 有效到 CKE 有效的时间间隔低位
WRLVL_STATUS	17:0	0x0	0x0-0x3ffff	最近一次 Write Leveling 操作状态 (只读)
CONF_CTL_142[63:0] Offset: 0x8e0 DDR2 667 : 0x0000000000000050				
TRST_PWRON	31:0	0x0	0x0-0xffffffff	从 start 有效 500 拍后到 DDR_RESET 有效之间的延迟
CONF_CTL_143[63:0] Offset: 0x8f0 DDR2 667 : 0x0000000020202080				
DLL_CTRL_REG_2 [32]	32:32	0x0	0x0-0x1	输出时钟 DLL 使能信号, 高有效
DLL_CTRL_REG_2 [31:0]	31:0	0x0	0x0-0xffffffff	输出时钟 DLL 控制 31:24 : 输出时钟 DLL 上 CLK4 与 CLK5 的延迟 23:16 : 输出时钟 DLL 上 CLK2 与 CLK3 的延迟 15:8 : 输出时钟 DLL 上 CLK0 与 CLK1 的延迟 7:0 : 输出时钟 DLL 上精度值
CONF_CTL_144[63:0] Offset: 0x900 DDR2 667 : 0x0000000000000000				
RDLVL_ERROR_STATUS[37:32]	37:32	0x0	0x0-0x3f	指示 RDLVL 发生错误时的状态
RDLVL_ERROR_STATUS[31:0]	31:0	0x0	0x0-0xffffffff	指示 RDLVL 发生错误时的状态
CONF_CTL_145[63:0] Offset: 0x910 DDR2 667 : 0x0000000000000000				
RDLVL_GATE_RESP_MASK[63:32]	63:32	0x0	0x0-0xffffffff	采样训练中读返回屏蔽
RDLVL_GATE_RESP_MASK[31:0]	31:0	0x0	0x0-0xffffffff	采样训练中读返回屏蔽
CONF_CTL_146[63:0] Offset: 0x920 DDR2 667 : 0x0000000000000000				
RDLVL_GATE_RESP_MASK[71:64]	71:64	0x0	0x0-0xff	采样训练中读返回屏蔽
CONF_CTL_147[63:0] Offset: 0x930 DDR2 667 : 0x0000000000000000				
RDLVL_RESP_MASK[63:32]	63:32	0x0	0x0-0xffffffff	Read Leveling 中读返回屏蔽
RDLVL_RESP_MASK[31:0]	31:0	0x0	0x0-0xffffffff	Read Leveling 中读返回屏蔽
CONF_CTL_148[63:0] Offset: 0x940 DDR2 667 : 0x0301010000050500				
TDFI_RDLVL_EN	59:56	0x0	0x0-0xf	Read Leveling 使能到 Read Leveling 读之间的最小周期数
W2R_SAMECS_DLY	50:48	0x0	0x0-0x7	对同一个片选信号, 写到读之间的附加延迟

参数名称	位	初值	范围	描述
W2R_DIFFCS_DLY	42:40	0x0	0x0-0x7	对不同片选信号，写到读之间的附加延迟
LVL_STATUS	34:32	0x0	0x0-0x7	Write Leveling， Read Leveling 与采样训练请求的状态，用于 LVL_REQ 中断（只读）
RDLVL_EDGE	24	0x0	0x0-0x1	Read Leveling 操作中，指明 DQS 上升沿有效或下降沿有效
CKSRX	19:16	0x0	0x0-0x0	退出自刷新模式的时钟周期延迟
CKSRE	11:8	0x0	0x0-0x0	进入自刷新模式的时钟周期延迟
RDLVL_RESP_MASK[71:64]	7:0	0x0	0x0-0xff	Read Leveling 中读返回屏蔽
CONF_CTL_149[63:0] Offset: 0x950 DDR2 667 : 0x000000000000a03				
INT_MASK[17:0]	41:24	0x0	0x0-0x3fff	中断屏蔽
TXPDLL	23:8	0x0	0x0-0xffff	DRAM TXPDLL parameter in cycles.
TDFL_WRLVL_EN	3:0	0x0	0x0-0xf	Write Leveling 使能到 Write Leveling 读操作最小周期数
CONF_CTL_150[63:0] Offset: 0x960 DDR2 667 : 0x0604000000000000				
RDLAT_ADJ	60:56	0x0	0x0-0x1f	PHY 读延迟周期
WRLAT_ADJ	51:48	0x0	0x0-0xf	PHY 写延迟周期
SWLVL_START	40	0x0	0x0-0x1	软件 Leveling 模式下开始操作（只写）
SWLVL_LOAD	32	0x0	0x0-0x1	软件 Leveling 模式下装入操作（只写）
SWLVL_EXIT	24	0x0	0x0-0x1	软件 Leveling 模式下退出操作（只写）
INT_STATUS	18:0	0x0	0x0-0x7fff	中断状态（只读）
CONF_CTL_151[63:0] Offset: 0x970 DDR2 667 : 0x000000000003e805				
CONCURRENTAP_WR_ONLY	56	0x0	0x0-0x1	写操作之后读操作之前是否通过等待写恢复时间来阻止并发的 auto-precharge 操作
CKE_STATUS	48	0x0	0x0-0x1	指示 CKE.STATUS（只读）
INT_ACK [16:0]	40:24	0x0	0x0-0x1fff	中断清除（只写）
DLL_RST_DELAY	23:8	0x0	0x0-0xffff	DLL 复位最小周期数
DLL_RST_ADJ_DLY	7:0	0x0	0x0-0xff	配置 DLL 精度到 DLL 复位结束的最小周期数
CONF_CTL_152[63:0] Offset: 0x980 DDR2 667 : 0x0001010001000101				
ZQ_IN_PROGRESS	56	0x0	0x0-0x1	指示 ZQ 操作正在进行（只读）
ZQCS_ROTATE	48	0x0	0x0-0x1	使能 ZQCS（short ZQ）轮流校正。当该位为 0 时，每次 ZQCS 请求命令会对系统中所有的片选进行校正，当该位置为 1 时，系统在每个 ZQCS 命令到来时只对一个片选进行校正，系统会轮流校正所有的片选。ZQCS 和 REFRESH_PER_ZQ 参数的设置应该与该位一致
WRLVL_REG_EN	40	0x0	0x0-0x1	使能写 wrlvl_delay 寄存器
WRLVL_EN	32	0x0	0x0-0x1	使能控制器的 Write Leveling 功能
RESYNC_DLL_PER_AREF_EN	24	0x0	0x0-0x1	使能在每个刷新命令之后 DLL 自动同步
RESYNC_DLL	16	0x0	0x0-0x1	发起一个 DLL 同步命令（只写）
RDLVL_REG_EN	8	0x0	0x0-0x1	使能写 rdlvl_delay 寄存器
RDLVL_GATE_REG_EN	0	0x0	0x0-0x1	使能写 rdlvl_gate_delay 寄存器
CONF_CTL_153[63:0] Offset: 0x990 DDR2 667 : 0x0101020202010100				

参数名称	位	初值	范围	描述
W2W_SAMECS_DLY	58:56	0x0	0x0-0x7	对同一个片选的写命令到写命令的附加延时时钟周期数
W2W_DIFFCS_DLY	50:48	0x0	0x0-0x7	对不同片选的写命令到写命令的附加延时时钟周期数
TBST_INT_INTERVAL	42:40	0x0	0x0-0x7	DRAM burst 中断间隔周期数
R2W_SAMECS_DLY	34:32	0x0	0x0-0x7	对同一个片选的读命令到写命令的附加延时时钟周期数
R2W_DIFFCS_DLY	26:24	0x0	0x0-0x7	对不同片选的读命令到写命令的附加延时时钟周期数
R2R_SAMECS_DLY	18:16	0x0	0x0-0x7	对同一个片选的读命令到读命令的附加延时时钟周期数
R2R_DIFFCS_DLY	10:8	0x0	0x0-0x7	对不同片选的读命令到读命令的附加延时时钟周期数
AXI_ALIGNED_STROBE_DISABLE	10:10	0x0	0x0-0x7	当 AXI 端口的事务具有以下特征之一时：一、事务的起始地址和结束地址按用户字对齐；二、事务长度为一个用户字（128 位），禁止 AXI Strobe，每位对应一个 AXI 端口。 当设为 0 时，写操作会按照读-修改-写的顺序执行； 当设为 1 时，写操作作为一个标准的写操作（非读-修改-写的顺序）
CONF_CTL_154[63:0] Offset: 0x9a0 DDR2 667 : 0x0707040200060100				
TDFI_WRLVL_LOAD	63:56	0x0	0x0-0xff	写 Leveling 延时数有效到第一个写 Leveling Load 命令的最小时钟周期数
TDFI_RDLVL_LOAD	55:48	0x0	0x0-0xff	读 Leveling 延时数有效到第一个读 Leveling Load 命令的最小时钟周期数
TCKESR	44:40	0x0	0x0-0x1f	自刷新进入到退出时 CKE 保持为低电平的最小时钟周期数
TCCD	36:32	0x0	0x0-0x1f	CASn 到 CASn 命令的延时
ADD_ODT_CLK_DIFFTYPE_DIFFCS	28:24	0x0	0x0-0x1f	为了满足 ODT 时序，对不同片选的不同命令之间插入的附加时钟周期数
TRP_AB	19:16	0x0	0x0-0xf	对所有 bank 的 trp 时间
ADD_ODT_CLK_SAMETYPE_DIFFCS	11:8	0x0	0x0-0xf	为了满足 ODT 时序，对不同片选的同类型命令之间插入的附加时钟周期数
ADD_ODT_CLK_DIFFTYPE_SAMECS	3:0	0x0	0x0-0xf	为了满足 ODT 时序，对同一片选的不同命令之间插入的附加时钟周期数
CONF_CTL_155[63:0] Offset: 0x9b0 DDR2 667 : 0x0200010000000000				
ZQINIT	59:48	0x0	0x0-0xffff	DRAM 初始化过程中 ZQ 命令需要的时钟周期数
ZQCL	43:32	0x0	0x0-0xffff	正常的 ZQCL 命令需要的时钟周期数，它应等于 ZQINIT 的一半
TDFI_WRLVL_WW	25:16	0x0	0x0-0x3ff	连续两次写 leveling 命令之间的最小时钟周期数
TDFI_RDLVL_RR	9:0	0x0	0x0-0x3ff	连续两次读 leveling 命令之间的最小时钟周期数
CONF_CTL_156[63:0] Offset: 0x9c0 DDR2 667 : 0x0a52000000000000				
MR0_DATA_0	62:48	0x0	0x0-0x7fff	对应片选 0 的模式寄存器 0 配置值
TDFI_PHYUPD_TYPE3	45:32	0x0	0x0-0x3fff	保存 DFI Tphyupd_type3 参数（只读）

参数名称	位	初值	范围	描述
TDFL.PHYUPD_TYPE2	29:16	0x0	0x0-0x3fff	保存 DFI Tphyupd_type2 参数 (只读)
TDFL.PHYUPD_TYPE1	13:0	0x0	0x0-0x3fff	保存 DFI Tphyupd_type1 参数 (只读)
CONF_CTL_157[63:0] Offset: 0x9d0 DDR2 667 : 0x00440a520a520a52				
MR1_DATA_0	62:48	0x0	0x0-0x7fff	对应片选 0 的模式寄存器 1 配置值
MR0_DATA_3	46:32	0x0	0x0-0x7fff	对应片选 3 的模式寄存器 0 配置值
MR0_DATA_2	30:16	0x0	0x0-0x7fff	对应片选 2 的模式寄存器 0 配置值
MR0_DATA_1	14:0	0x0	0x0-0x7fff	对应片选 1 的模式寄存器 0 配置值
CONF_CTL_158[63:0] Offset: 0x9e0 DDR2 667 : 0x0000004400440044				
MR2_DATA_0	62:48	0x0	0x0-0x7fff	对应片选 0 的模式寄存器 2 配置值
MR1_DATA_3	46:32	0x0	0x0-0x7fff	对应片选 3 的模式寄存器 1 配置值
MR1_DATA_2	30:16	0x0	0x0-0x7fff	对应片选 2 的模式寄存器 1 配置值
MR1_DATA_1	14:0	0x0	0x0-0x7fff	对应片选 1 的模式寄存器 1 配置值
CONF_CTL_159[63:0] Offset: 0x9f0 DDR2 667 : 0x0000000000000000				
MR3_DATA_0	62:48	0x0	0x0-0x7fff	对应片选 0 的模式寄存器 3 配置值
MR2_DATA_3	46:32	0x0	0x0-0x7fff	对应片选 3 的模式寄存器 2 配置值
MR2_DATA_2	30:16	0x0	0x0-0x7fff	对应片选 2 的模式寄存器 2 配置值
MR2_DATA_1	14:0	0x0	0x0-0x7fff	对应片选 1 的模式寄存器 2 配置值
CONF_CTL_160[63:0] Offset: 0xa00 DDR2 667 : 0x00ff000000000000				
DFI.WRLVL.MAX_DELAY	63:48	0x0	0x0-0xffff	Hardware Write leveling 会使用的延迟线的最大级数
MR3_DATA_3	46:32	0x0	0x0-0x7fff	对应片选 3 的模式寄存器 3 配置值
MR3_DATA_2	30:16	0x0	0x0-0x7fff	对应片选 2 的模式寄存器 3 配置值
MR3_DATA_1	14:0	0x0	0x0-0x7fff	对应片选 1 的模式寄存器 3 配置值
CONF_CTL_161[63:0] Offset: 0xa10 DDR2 667 : 0x0000000000000000				
RDLVL.BEGIN_DELAY_3	63:48	0x0	0x0-0xffff	第 3 数据组中, Read Leveling 时从第一个 1 到 0 的延迟单元数目
RDLVL.BEGIN_DELAY_2	47:32	0x0	0x0-0xffff	第 2 数据组中, Read Leveling 时从第一个 1 到 0 的延迟单元数目
RDLVL.BEGIN_DELAY_1	31:16	0x0	0x0-0xffff	第 1 数据组中, Read Leveling 时从第一个 1 到 0 的延迟单元数目
RDLVL.BEGIN_DELAY_0	15:0	0x0	0x0-0xffff	第 0 数据组中, Read Leveling 时从第一个 1 到 0 的延迟单元数目
CONF_CTL_162[63:0] Offset: 0xa20 DDR2 667 : 0x0000000000000000				

参数名称	位	初值	范围	描述
RDLVL_BEGIN_DELAY_7	63:48	0x0	0x0-0xffff	第 7 数据组中, Read Leveling 时从第一个 1 到 0 的延迟单元数目
RDLVL_BEGIN_DELAY_6	47:32	0x0	0x0-0xffff	第 6 数据组中, Read Leveling 时从第一个 1 到 0 的延迟单元数目
RDLVL_BEGIN_DELAY_5	31:16	0x0	0x0-0xffff	第 5 数据组中, Read Leveling 时从第一个 1 到 0 的延迟单元数目
RDLVL_BEGIN_DELAY_4	15:0	0x0	0x0-0xffff	第 4 数据组中, Read Leveling 时从第一个 1 到 0 的延迟单元数目
CONF_CTL_163[63:0] Offset: 0xa30 DDR2 667 : 0x0000000000000000				
RDLVL_DELAY_2	63:48	0x0	0x0-0xffff	第 2 数据组中, Read Leveling 使用的延迟单元数目
RDLVL_DELAY_1	47:32	0x0	0x0-0xffff	第 1 数据组中, Read Leveling 使用的延迟单元数目
RDLVL_DELAY_0	31:16	0x0	0x0-0xffff	第 0 数据组中, Read Leveling 使用的延迟单元数目
RDLVL_BEGIN_DELAY_8	15:0	0x0	0x0-0xffff	第 8 数据组中, Read Leveling 时从第一个 1 到 0 的延迟单元数目
CONF_CTL_164[63:0] Offset: 0xa40 DDR2 667 : 0x0000000000000000				
RDLVL_DELAY_6	63:48	0x0	0x0-0xffff	第 6 数据组中, Read Leveling 使用的延迟单元数目
RDLVL_DELAY_5	47:32	0x0	0x0-0xffff	第 5 数据组中, Read Leveling 使用的延迟单元数目
RDLVL_DELAY_4	31:16	0x0	0x0-0xffff	第 4 数据组中, Read Leveling 使用的延迟单元数目
RDLVL_DELAY_3	15:0	0x0	0x0-0xffff	第 3 数据组中, Read Leveling 使用的延迟单元数目
CONF_CTL_165[63:0] Offset: 0xa50 DDR2 667 : 0x0000000000000000				
RDLVL_END_DELAY_1	63:48	0x0	0x0-0xffff	第 1 数据组中, Read Leveling 时从第一个 0 到 1 的延迟单元数目
RDLVL_END_DELAY_0	47:32	0x0	0x0-0xffff	第 0 数据组中, Read Leveling 时从第一个 0 到 1 的延迟单元数目
RDLVL_DELAY_8	31:16	0x0	0x0-0xffff	第 8 数据组中, Read Leveling 使用的延迟单元数目
RDLVL_DELAY_7	15:0	0x0	0x0-0xffff	第 7 数据组中, Read Leveling 使用的延迟单元数目
CONF_CTL_166[63:0] Offset: 0xa60 DDR2 667 : 0x0000000000000000				
RDLVL_END_DELAY_5	63:48	0x0	0x0-0xffff	第 5 数据组中, Read Leveling 时从第一个 0 到 1 的延迟单元数目
RDLVL_END_DELAY_4	47:32	0x0	0x0-0xffff	第 4 数据组中, Read Leveling 时从第一个 0 到 1 的延迟单元数目
RDLVL_END_DELAY_3	31:16	0x0	0x0-0xffff	第 3 数据组中, Read Leveling 时从第一个 0 到 1 的延迟单元数目
RDLVL_END_DELAY_2	15:0	0x0	0x0-0xffff	第 2 数据组中, Read Leveling 时从第一个 0 到 1 的延迟单元数目
CONF_CTL_167[63:0] Offset: 0xa70 DDR2 667 : 0x0000000000000000				
RDLVL_GATE_DELAY_0	63:48	0x0	0x0-0xffff	第 0 数据组中, 采样时机到选通信号上升沿的延迟单元个数
RDLVL_END_DELAY_8	47:32	0x0	0x0-0xffff	第 8 数据组中, Read Leveling 时从第一个 0 到 1 的延迟单元数目

参数名称	位	初值	范围	描述
RDLVL_END_DELAY_7	31:16	0x0	0x0-0xffff	第 7 数据组中, Read Leveling 时从第一个 0 到 1 的延迟单元数目
RDLVL_END_DELAY_6	15:0	0x0	0x0-0xffff	第 6 数据组中, Read Leveling 时从第一个 0 到 1 的延迟单元数目
CONF_CTL_168[63:0] Offset: 0xa80 DDR2 667 : 0x0000000000000000				
RDLVL_GATE_DELAY_4	63:48	0x0	0x0-0xffff	第 4 数据组中, 采样时机到选通信号上升沿的延迟单元个数
RDLVL_GATE_DELAY_3	47:32	0x0	0x0-0xffff	第 3 数据组中, 采样时机到选通信号上升沿的延迟单元个数
RDLVL_GATE_DELAY_2	31:16	0x0	0x0-0xffff	第 2 数据组中, 采样时机到选通信号上升沿的延迟单元个数
RDLVL_GATE_DELAY_1	15:0	0x0	0x0-0xffff	第 1 数据组中, 采样时机到选通信号上升沿的延迟单元个数
CONF_CTL_169[63:0] Offset: 0xa90 DDR2 667 : 0x0000000000000000				
RDLVL_GATE_DELAY_8	63:48	0x0	0x0-0xffff	第 8 数据组中, 采样时机到选通信号上升沿的延迟单元个数
RDLVL_GATE_DELAY_7	47:32	0x0	0x0-0xffff	第 7 数据组中, 采样时机到选通信号上升沿的延迟单元个数
RDLVL_GATE_DELAY_6	31:16	0x0	0x0-0xffff	第 6 数据组中, 采样时机到选通信号上升沿的延迟单元个数
RDLVL_GATE_DELAY_5	15:0	0x0	0x0-0xffff	第 5 数据组中, 采样时机到选通信号上升沿的延迟单元个数
CONF_CTL_170[63:0] Offset: 0xaa0 DDR2 667 : 0x0000ffff00000010				
RDLVL_MIDPOINT_DELAY_0	63:48	0x0	0x0-0xffff	当 Hardware read leveling 模块使能时, 等于 rdlvl_begin_delay_0 和 rdlvl_end_delay_0 的中间值, 否则, 等于 rdlvl_delay_0(只读)
RDLVL_MAX_DELAY	47:32	0x0	0x0-0xffff	Read Leveling 延迟线的最大数目
RDLVL_GATE_REFRESH_INTERVAL	15:0	0x0	0x0-0xffff	两次自动 Gate Training 之间的最大刷新命令数(应设为 0)
RDLVL_GATE_MAX_DELAY	15:0	0x0	0x0-0xffff	采样延迟线的最大数目
CONF_CTL_171[63:0] Offset: 0xab0 DDR2 667 : 0x0000000000000000				
RDLVL_MIDPOINT_DELAY_4	63:48	0x0	0x0-0xffff	当 Hardware read leveling 模块使能时, 等于 rdlvl_begin_delay_4 和 rdlvl_end_delay_4 的中间值, 否则, 等于 rdlvl_delay_4(只读)
RDLVL_MIDPOINT_DELAY_3	47:32	0x0	0x0-0xffff	当 Hardware read leveling 模块使能时, 等于 rdlvl_begin_delay_3 和 rdlvl_end_delay_3 的中间值, 否则, 等于 rdlvl_delay_3(只读)
RDLVL_MIDPOINT_DELAY_2	31:16	0x0	0x0-0xffff	当 Hardware read leveling 模块使能时, 等于 rdlvl_begin_delay_2 和 rdlvl_end_delay_2 的中间值, 否则, 等于 rdlvl_delay_2(只读)
RDLVL_MIDPOINT_DELAY_1	15:0	0x0	0x0-0xffff	当 Hardware read leveling 模块使能时, 等于 rdlvl_begin_delay_1 和 rdlvl_end_delay_1 的中间值, 否则, 等于 rdlvl_delay_1(只读)
CONF_CTL_172[63:0] Offset: 0xac0 DDR2 667 : 0x0000000000000000				
RDLVL_MIDPOINT_DELAY_8	63:48	0x0	0x0-0xffff	当 Hardware read leveling 模块使能时, 等于 rdlvl_begin_delay_8 和 rdlvl_end_delay_8 的中间值, 否则, 等于 rdlvl_delay_8(只读)

参数名称	位	初值	范围	描述
RDLVL_MIDPOINT_DELAY_7	47:32	0x0	0x0-0xffff	当 Hardware read leveling 模块使能时, 等于 rdlvl_begin_delay_7 和 rdlvl_end_delay_7 的中间值, 否则, 等于 rdlvl_delay_7(只读)
RDLVL_MIDPOINT_DELAY_6	31:16	0x0	0x0-0xffff	当 Hardware read leveling 模块使能时, 等于 rdlvl_begin_delay_6 和 rdlvl_end_delay_6 的中间值, 否则, 等于 rdlvl_delay_6(只读)
RDLVL_MIDPOINT_DELAY_5	15:0	0x0	0x0-0xffff	当 Hardware read leveling 模块使能时, 等于 rdlvl_begin_delay_5 和 rdlvl_end_delay_5 的中间值, 否则, 等于 rdlvl_delay_5(只读)
CONF_CTL_173[63:0] Offset: 0xad0 DDR2 667 : 0x0000000000000000				
RDLVL_OFFSET_DELAY_3	63:48	0x0	0x0-0xffff	第 3 数据组中, 到 Read Leveling 中点的偏移
RDLVL_OFFSET_DELAY_2	47:32	0x0	0x0-0xffff	第 2 数据组中, 到 Read Leveling 中点的偏移
RDLVL_OFFSET_DELAY_1	31:16	0x0	0x0-0xffff	第 1 数据组中, 到 Read Leveling 中点的偏移
RDLVL_OFFSET_DELAY_0	15:0	0x0	0x0-0xffff	第 0 数据组中, 到 Read Leveling 中点的偏移
CONF_CTL_174[63:0] Offset: 0xae0 DDR2 667 : 0x0000000000000000				
RDLVL_OFFSET_DELAY_7	63:48	0x0	0x0-0xffff	第 7 数据组中, 到 Read Leveling 中点的偏移
RDLVL_OFFSET_DELAY_6	47:32	0x0	0x0-0xffff	第 6 数据组中, 到 Read Leveling 中点的偏移
RDLVL_OFFSET_DELAY_5	31:16	0x0	0x0-0xffff	第 5 数据组中, 到 Read Leveling 中点的偏移
RDLVL_OFFSET_DELAY_4	15:0	0x0	0x0-0xffff	第 4 数据组中, 到 Read Leveling 中点的偏移
CONF_CTL_175[63:0] Offset: 0xaf0 DDR2 667 : 0x0000000000000000				
WRLVL_DELAY_1	63:48	0x0	0x0-0xffff	第 1 数据组中, 控制写 DQS 经 DLL 延迟数
WRLVL_DELAY_0	47:32	0x0	0x0-0xffff	第 0 数据组中, 控制写 DQS 经 DLL 延迟数
RDLVL_REFRESH_INTERVAL	31:16	0x0	0x0-0xffff	两次自动 Read Leveling 之间的最大刷新命令数(应设为 0)
RDLVL_OFFSET_DELAY_8	15:0	0x0	0x0-0xffff	第 8 数据组中, 到 Read Leveling 中点的偏移
CONF_CTL_176[63:0] Offset: 0xb00 DDR2 667 : 0x0000000000000000				
WRLVL_DELAY_5	63:48	0x0	0x0-0xffff	第 5 数据组中, 控制写 DQS 经 DLL 延迟数
WRLVL_DELAY_4	47:32	0x0	0x0-0xffff	第 4 数据组中, 控制写 DQS 经 DLL 延迟数
WRLVL_DELAY_3	31:16	0x0	0x0-0xffff	第 3 数据组中, 控制写 DQS 经 DLL 延迟数
WRLVL_DELAY_2	15:0	0x0	0x0-0xffff	第 2 数据组中, 控制写 DQS 经 DLL 延迟数
CONF_CTL_177[63:0] Offset: 0xb10 DDR2 667 : 0x0000000000000000				
WRLVL_REFRESH_INTERVAL	63:48	0x0	0x0-0xffff	两次自动 Write Leveling 之间的最大刷新命令数(应设为 0)
WRLVL_DELAY_8	47:32	0x0	0x0-0xffff	第 8 数据组中, 控制写 DQS 经 DLL 延迟数
WRLVL_DELAY_7	31:16	0x0	0x0-0xffff	第 7 数据组中, 控制写 DQS 经 DLL 延迟数
WRLVL_DELAY_6	15:0	0x0	0x0-0xffff	第 6 数据组中, 控制写 DQS 经 DLL 延迟数
CONF_CTL_178[63:0] Offset: 0xb20 DDR2 667 : 0x00000c2d00000c2d				
TDFI_RDLVL_RESP	63:32	0x0	0x0-0xffff	保存 DFI Trdlvl_resp 时间参数
TDFI_RDLVL_MAX	31:0	0x0	0x0-0xffff	保存 DFI Trdlvl_max 时间参数
CONF_CTL_179[63:0] Offset: 0xb30 DDR2 667 : 0x00000c2d00000c2d				
TDFI_WRLVL_RESP	63:32	0x0	0x0-0xffff	保存 DFI Twrlvl_resp 时间参数
TDFI_WRLVL_MAX	31:0	0x0	0x0-0xffff	保存 DFI Twrlvl_max 时间参数

第七章 HT 控制器

龙芯 2H 中，HyperTransport 总线用于连接 3A 芯片。用于外设连接时，用户程序可以自由选择是否支持 IO Cache 一致性（由地址窗口 Uncache 窗口设置，见 7.4.2 节）。在 Cache 一致性支持模式下，IO 设备对内 DMA 访问对于 Cache 层次透明，即不需要通过程序 Cache 指令维护一致性，而是由硬件自动维护其一致性。

HyperTransport 控制器在龙芯 2H 中只支持双向 8 位宽度，最高运行频率为 800Mhz。在系统自动初始化建立连接后，用户可通过修改协议中的配置寄存器对需要的运行频率与宽度进行修改，重新初始化，详细方法见 7.1 节。

龙芯 2H HyperTransport 控制器的主要特征如下：

- 支持 200/400/800Mhz
- 支持 8 位宽度
- 总线控制信号（包括 PowerOK，Rstn，LDT_Stopn）方向可配置
- 外设 DMA 空间 Cache/Uncache 可配置
- HT0 控制器只能用于连接 3A

注：以下的控制器配置及初始化中所有的 Hi 信号和控制位宽的寄存器都写成定值了，不可配置与访问。

7.1 HT 硬件设置及初始化

HyperTransport 的初始化在每次复位完成后自动开始，冷启动后 HyperTransport 总线将自动工作在最低频率 (200Mhz) 与最小宽度 (8bit)，并尝试进行总线初始化握手。初始化是否完成的状态可以由寄存器“Init Complete”（见 7.3.2 节）读出。初始化完成后，总线的宽度可以由寄存器“Link Width Out”与“Link Width In”（见 7.3.2 节）读出。在初始化完成后，用户可以重新对寄存器“Link Width Out”，“Link Width In”以及“Link Freq”进行编程，同时需要对对方设备的相应寄存器也进行编程，编程完成后需要重新热复位总线或是通过 HT_Ldt_Stopn 信号对总线进行重新初始化操作，以使编程后的值在重新初始化后生效。在重新初始化后 HyperTransport 总线将工作在新的频率和宽度。需要注意的是，HyperTransport 两端的设备的配置需要一一对应，否则将使得 HyperTransport 接口不能正常工作。

7.2 HT 协议支持

HyperTransport 总线支持 1.03 协议中的大部分命令，并且在多片互联支持的扩展一致性协议中加入了一些扩展指令，对于两种模式下，HyperTransport 接收端可接收

的命令如下表所示。需要注意的是，不支持 HyperTransport 总线的原子操作命令。

表 7.1: HyperTransport 接收端可接收的命令

编码	通道	命令	标准模式	扩展（一致性）
000000	-	NOP	空包或流控	
000001	NPC	FLUSH	无操作	
x01xxx	NPC or PC	Write	bit 5: 0 - Nonposted 1 - Posted bit 2: 0 - Byte 1 - Doubleword bit 1: Don't Care bit 0: Don't Care	bit 5: 必为 1, POSTED bit 2: 0 - Byte 1 - Doubleword bit 1: Don't Care bit 0: 必为 1
01xxxx	NPC	Read	bit 3: Don't Care bit 2: 0 - Byte 1 - Doubleword bit 1: Don't Care bit 0: Don't Care	bit 3: Don't Care bit 2: 0 - Byte 1 - Doubleword bit 1: Don't Care bit 0: 必为 1
110000	R	RdResponse	读操作返回	
110011	R	TgtDone	写操作返回	
110100	PC	WrCoherent	—	写命令扩展
110101	PC	WrAddr	—	写地址扩展
111000	R	RespCoherent	—	读响应扩展
111001	NPC	RdCoherent	—	读命令扩展
111010	PC	Broadcast	无操作	
111011	NPC	RdAddr	—	读地址扩展
111100	PC	FENCE	保证序关系	
111111	-	Sync/Error	Sync/Error	

对于发送端，在两种模式下会向外发送的命令如下表所示。

表 7.2: HyperTransport 发送端会向外发送的命令

编码	通道	命令	标准模式	扩展（一致性）
000000	-	NOP	空包或流控	
x01x0x	NPC or PC	Write	bit 5: 0 - Nonposted 1 - Posted bit 2: 0 - Byte 1 - Doubleword bit 0: 必为 0	bit 5: 必为 1, POSTED bit 2: 0 - Byte 1 - Doubleword bit 0: 必为 1
010x0x	NPC	Read	bit 2: 0 - Byte 1 - Doubleword bit 0: Don't Care	bit 2: 0 - Byte 1 - Doubleword bit 0: 必为 1
110000	R	RdResponse	读操作返回	
110011	R	TgtDone	写操作返回	
110100	PC	WrCoherent	—	写命令扩展
110101	PC	WrAddr	—	写地址扩展
111000	R	RespCoherent	—	读响应扩展
111001	NPC	RdCoherent	—	读命令扩展

编码	通道	命令	标准模式	扩展（一致性）
111011	NPC	RdAddr	—	读地址扩展
111111	-	Sync/Error	只会转发	

7.3 HT 配置寄存器

每个寄存器的具体含义如下节如示：

表 7.3: HT 配置寄存器列表

偏移地址	名称	描述
0x30		
0x34		
0x38		
0x3c	Bridge Control	Bus Reset Control
0x40	Capability Registers	Command , Capabilities Pointer , Capability ID
0x44		Link Config , Link Control
0x48		Revision ID , Link Freq , Link Error , Link Freq Cap
0x4c		Feature Capability
0x50	自定义寄存器	MISC
0x54		
0x58		
0x5c		
0x60	接收地址窗口配置寄存器	HT 总线接收地址窗口 0 使能（外部访问）
0x64		HT 总线接收地址窗口 0 基址（外部访问）
0x68		HT 总线接收地址窗口 1 使能（外部访问）
0x6c		HT 总线接收地址窗口 1 基址（外部访问）
0x70		HT 总线接收地址窗口 2 使能（外部访问）
0x74		HT 总线接收地址窗口 2 基址（外部访问）
0x78		
0x7c		
0x80	中断向量寄存器	HT 总线中断向量寄存器[31:0]
0x84		HT 总线中断向量寄存器[63:32]
0x88		HT 总线中断向量寄存器[95:64]
0x8c		HT 总线中断向量寄存器[127:96]
0x90		HT 总线中断向量寄存器[159:128]
0x94		HT 总线中断向量寄存器[191:160]
0x98		HT 总线中断向量寄存器[223:192]
0x9C		HT 总线中断向量寄存器[255:224]
0xA0	中断使能寄存器	HT 总线中断使能寄存器[31:0]
0xA4		HT 总线中断使能寄存器[63:32]
0xA8		HT 总线中断使能寄存器[95:64]
0xAC		HT 总线中断使能寄存器[127:96]
0xB0		HT 总线中断使能寄存器[159:128]
0xB4		HT 总线中断使能寄存器[191:160]

偏移地址	名称	描述
0xB8		HT 总线中断使能寄存器[223:192]
0xBC		HT 总线中断使能寄存器[255:224]
0xC0	Interrupt Discovery & Configuration	Interrupt Capability
0xC4		DataPort
0xC8		IntrInfo[31:0]
0xCC		IntrInfo[63:32]
0xD0		POST 地址窗口配置寄存器
0xD4	HT 总线 POST 地址窗口 0 基址 (内部访问)	
0xD8	HT 总线 POST 地址窗口 1 使能 (内部访问)	
0xDC	HT 总线 POST 地址窗口 1 基址 (内部访问)	
0xE0	可预取地址窗口配置寄存器	HT 总线可预取地址窗口 0 使能 (内部访问)
0xE4		HT 总线可预取地址窗口 0 基址 (内部访问)
0xE8		HT 总线可预取地址窗口 1 使能 (内部访问)
0xEC		Ht 总线可预取地址窗口 1 基址 (内部访问)
0xF0	Uncache 地址窗口配置寄存器	HT 总线 Uncache 地址窗口 0 使能 (内部访问)
0xF4		HT 总线 Uncache 地址窗口 0 基址 (内部访问)
0xF8		HT 总线 Uncache 地址窗口 1 使能 (内部访问)
0xFC		HT 总线 Uncache 地址窗口 1 基址 (内部访问)

7.3.1 Bridge Control

偏移量: 0x3C

复位值: 0x00200000

名称: Bus Reset Control

位域	名称	位宽	初值	访问	描述
31:23	Reserved	4	0x0		保留
22	Reset	12	0x0	R/W	总线复位控制: 0→1: HT_RSTn 置 0, 总线复位 1→0: HT_RSTn 置 1, 总线解复位
21:0	Reserved	5	0x0		保留

7.3.2 Capability Registers

偏移量: 0x40

复位值: 0x20010008

名称: Command, Capabilities Pointer, Capability ID

位域	名称	位宽	初值	访问	描述
31:29	HOST/Sec	3	0x1	R	Command 格式为 HOST/Sec
28:27	Reserved	2	0x0	R	保留

位域	名称	位宽	初值	访问	描述
26	Act as Slave	1	0x0/0x1	R/W	HOST/SLAVE 模式初始值由引脚 HOST-MODE 决定 HOSTMODE 上拉: 0 HOSTMODE 下拉: 1
25	Reserved	1	0x0		保留
24	Host Hide	1	0x0	R/W	是否禁止来自 HT 总线的寄存器访问
23	Reserved	1	0x0		保留
22:18	Unit ID	5	0x0	R/W	HOST 模式时: 可用于记录使用 ID 个数 SLAVE 模式时: 记录自身 Unit ID
17	Double Ended	1	0x0	R	不采用双 HOST 模式
16	Warm Reset	1	0x1	R	Bridge Control 中 reset 采用热复位方式
15:8	Capabilities Pointer	8	0xa0	R	下一个 Cap 寄存器偏移地址
7:0	Capability ID	8	0x08	R	HyperTransport capability ID

偏移量: 0x44

复位值: 0x00112000

名称: Link Config, Link Control

位域	名称	位宽	初值	访问	描述
31	Reserved	1	0x0		保留
30:28	Link Width Out	3	0x0	R/W	发送端宽度 冷复位后的值为当前连接的最大宽度, 写入此寄存器的值将会在下次热复位或是 HT Disconnect 之后生效 000: 8 位方式 001: 16 位方式
27	Reserved	1	0x0		保留
26:24	Link Width In	3	0x0	R/W	接收端宽度 冷复位后的值为当前连接的最大宽度, 写入此寄存器的值将会在下次热复位或是 HT Disconnect 之后生效
23	Dw Fc out	1	0x0	R	发送端不支持双字流控
22:20	Max Link Width out	3	0x1	R	HT 总线发送端最大宽度: 16bits
19	Dw Fc In	1	0x0	R	接收端不支持双字流控
18:16	Max Link Width In	3	0x1	R	HT 总线接收端最大宽度: 16bits
15:14	Reserved	2	0x0		保留
13	LDTSTOP# Tristate Enable	1	0x1	R/W	当 HT 总线进入 HT Disconnect 状态时, 是否关闭 HT PHY 1: 关闭 0: 不关闭
12:10	Reserved	3	0x0		保留
9	CRC Error (hi)	1	0x0	R/W	高 8 位发生 CRC 错
8	CRC Error (lo)	1	0x0	R/W	低 8 位发生 CRC 错

位域	名称	位宽	初值	访问	描述
7	Trans off	1	0x0	R/W	HT PHY 关闭控制 处于 16 位总线工作方式时 1：关闭高 / 低 8 位 HT PHY 0：使能低 8 位 HT PHY， 高 8 位 HT PHY 由 bit 0 控制
6	End of Chain	0	0x0	R	HT 总线末端
5	Init Complete	1	0x0	R	HT 总线初始化是否完成
4	Link Fail	1	0x0	R	指示连接失败
3:2	Reserved	2	0x0		保留
1	CRC Flood Enable	1	0x0	R/W	发生 CRC 错误时，是否 flood HT 总线
0	Trans off (hi)	1	0x0	R/W	使用 16 位 HT 总线运行 8 位协议时， 高 8 位 PHY 关闭控制 1：关闭高 8 位 HT PHY 0：使能高 8 位 HT PHY

偏移量：0x48

复位值：0x80250023

名称： Revision ID， Link Freq， Link Error， Link Freq Cap

位域	名称	位宽	初值	访问	描述
31:16	Link Freq Cap	16	0x0025	R	支持的 HT 总线频率 200Mhz， 400Mhz， 800Mhz，
15:14	Reserved	2	0x0		保留
13	Over Flow Error	1	0x0	R	HT 总线包溢出
12	Protocol Error	1	0x0	R/W	协议错误， 指 HT 总线上收到不可识别的命令
11:8	Link Freq	4	0x0	R/W	HT 总线工作频率 写入此寄存器的值后将在下次热复位或是 HT Disconnect 之后生效 0000: 200M 0010: 400M 0101: 800M
7:0	Revision ID	8	0x23	R/W	版本号： 1.03

偏移量：0x4C

复位值：0x00000002

名称： Feature Capability

位域	名称	位宽	初值	访问	描述
31:9	Reserved	25	0x0		保留
8	Extended Register	1	0x0	R	没有
7:4	Reserved	3	0x0		保留
3	Extended CTL Time	1	0x0	R	不需要

位域	名称	位宽	初值	访问	描述
2	CRC Test Mode	1	0x0	R	不支持
1	LDTSTOP#	1	0x1	R	支持 LDTSTOP#
0	Isochronous Mode	1	0x0	R	不支持

7.3.3 自定义寄存器

偏移量: 0x50

复位值: 0x00904321

名称: MISC

位域	名称	位宽	初值	访问	描述
31	Reserved	1	0x0		保留
30	Ldt Stop Gen	1	0x0	R/W	使总线进入 LDT DISCONNECT 模式 正确的方法是: 0 → 1
29	Ldt Req Gen	1	0x0	R/W	从 LDT DISCONNECT 中唤醒 HT 总线, 设置 LDT_REQ_n 正确的方法是先置 0 再置 1: 0 → 1 除此之外, 直接向总线发出读写请求也可以自动唤醒总线
28:24	Interrupt Index	5	0x0	R/W	将除了标准中断之外的其它中断重定向到哪个中断向量中 (包括 SMI, NMI, INIT, INTA, INTB, INTC, INTD) 总共 256 个中断向量, 本寄存器表示的是中断向量的高 5 位, 内部中断向量如下: 000: SMI 001: NMI 010: INIT 011: Reserved 100: INTA 101: INTB 110: INTC 111: INTD
23	Dword Write	1	0x1	R/W	对于 32/64/128/256 位的写访问, 是否采用 Dword Write 命令格式 1: 使用 Dword Write 0: 使用 Byte Write (带 MASK)
22	Coherent Mode	1	0x0	R	是否是处理器一致性模式 由引脚 ICCEN 决定
21	Not Care Seqid	1	0x0	R/W	是否不关心 HT 序关系
20	Not Axi2Seqid	1	0x1	R	是否把 Axi 总线上的命令转换成不同的 SeqID, 如果不转换, 则所有的读写命令都会采用 Fixed Seqid 中的固定 ID 号 1: 不转换 0: 转换

位域	名称	位宽	初值	访问	描述
19:16	Fixed Seqid	4	0x0	R/W	当 Not Axi2Seqid 有效时, 配置 HT 总线发出的 Seqid
15:12	Priority Nop	4	0x4	R/W	HT 总线 Nop 流控包优先级
11:8	Priority NPC	4	0x3	R/W	Non Post 通道读写优先级
7:4	Priority RC	4	0x2	R/W	Response 通道读写优先级
3:0	Priority PC	4	0x1	R/W	Post 通道读写优先级 0x0 : 最高优先级 0xF : 最低优先级 对于各个通道的优先级均采用根据时间变化提高的优先级策略, 该组寄存器用于配置各个通道的初始优先级

7.3.4 接收地址窗口配置寄存器

本控制器中的地址窗口命中公式如下:

$$\text{hit} = (\text{BASE} \& \text{MASK}) == (\text{ADDR} \& \text{MASK})$$

$$\text{addr.out} = \text{TRANS_EN} ? \text{TRANS} | \text{ADDR} \& \text{MASK} : \text{ADDR}$$

值得一提的是, 配置地址窗口寄存器时, MASK 高位应全为 1, 低位应全为 0。MASK 中 0 的实际位数表示的就是地址窗口的大小。本窗口的地址为 HT 总线上接收的地址。落在本窗口内的 HT 地址将被发往 CPU 内, 其它地址的命令将作为 P2P 命令被转发回 HT 总线。

偏移量: 0x60

复位值: 0x00000000

名称: HT 总线接收地址窗口 0 使能 (外部访问)

位域	名称	位宽	初值	访问	描述
31	ht_rx_image0_en	1	0x0	R/W	HT 总线接收地址窗口 0, 使能信号
30	ht_rx_image0_trans_en	1	0x0	R/W	HT 总线接收地址窗口 0, 映射使能信号
29:23	Reserved	14	0x0		保留
15:0	ht_rx_image0_trans[39:24]	16	0x0	R/W	HT 总线接收地址窗口 0, 映射后地址的[39:24]

偏移量: 0x64

复位值: 0x00000000

名称: HT 总线接收地址窗口 0 基址 (外部访问)

位域	名称	位宽	初值	访问	描述
31:16	ht_rx_image0_base[39:24]	16	0x0	R/W	HT 总线接收地址窗口 0, 地址基址的[39:24]
15:0	ht_rx_image0_mask[39:24]	16	0x0	R/W	HT 总线接收地址窗口 0, 地址屏蔽的[39:24]

偏移量: 0x68

复位值: 0x00000000

名称: HT 总线接收地址窗口 1 使能 (外部访问)

位域	名称	位宽	初值	访问	描述
31	ht_rx_image1_en	1	0x0	R/W	HT 总线接收地址窗口 1, 使能信号
30	ht_rx_image1_trans_en	1	0x0	R/W	HT 总线接收地址窗口 1, 映射使能信号
29:23	Reserved	14	0x0		保留
15:0	ht_rx_image1_trans[39:24]	16	0x0	R/W	HT 总线接收地址窗口 1, 映射后地址的[39:24]

偏移量: 0x6c

复位值: 0x00000000

名称: HT 总线接收地址窗口 1 基址 (外部访问)

位域	名称	位宽	初值	访问	描述
31:16	ht_rx_image1_base[39:24]	16	0x0	R/W	HT 总线接收地址窗口 1, 地址基址的[39:24]
15:0	ht_rx_image1_mask[39:24]	16	0x0	R/W	HT 总线接收地址窗口 1, 地址屏蔽的[39:24]

偏移量: 0x70

复位值: 0x00000000

名称: HT 总线接收地址窗口 2 使能 (外部访问)

位域	名称	位宽	初值	访问	描述
31	ht_rx_image2_en	1	0x0	R/W	HT 总线接收地址窗口 2, 使能信号
30	ht_rx_image2_trans_en	1	0x0	R/W	HT 总线接收地址窗口 2, 映射使能信号
29:23	Reserved	14	0x0		保留
15:0	ht_rx_image2_trans[39:24]	16	0x0	R/W	HT 总线接收地址窗口 2, 转译后地址的[39:24]

偏移量: 0x74

复位值: 0x00000000

名称: HT 总线接收地址窗口 2 基址 (外部访问)

位域	名称	位宽	初值	访问	描述
31:16	ht_rx_image2_base[39:24]	16	0x0	R/W	HT 总线接收地址窗口 2, 地址基址的[39:24]
15:0	ht_rx_image2_mask[39:24]	16	0x0	R/W	HT 总线接收地址窗口 2, 地址屏蔽的[39:24]

7.3.5 中断向量寄存器

中断向量寄存器共 256 个, 其中除去 HT 总线上 Fixed 与 Arbitrated, PIC 中断直接映射到此 256 个中断向量之中, 其它的中断, 如 SMI, NMI, INIT, INTA, INTB

，INTC，INTD 可以通过寄存器 0x50 的 [28:24] 映射到任何一个 8 位中断向量上去，映射的顺序为 INTD，INTC，INTB，INTA，1'b0，INIT，NMI，SMI。此时中断向量对应值为 Interrupt Index, 内部向量 [2:0]。

偏移量: 0x80

复位值: 0x00000000

名称: HT 总线中断向量寄存器[31:0]

位域	名称	位宽	初值	访问	描述
31:0	Interrupt_case[31:0]	32	0x0	R/W	HT 总线中断向量寄存器[31:0]， 对应中断线 0 /HT HI 对应中断线 4

偏移量: 0x84

复位值: 0x00000000

名称: HT 总线中断向量寄存器[63:32]

位域	名称	位宽	初值	访问	描述
31:0	Interrupt_case[63:32]	32	0x0	R/W	HT 总线中断向量寄存器[63:32]， 对应中断线 0 /HT HI 对应中断线 4

偏移量: 0x88

复位值: 0x00000000

名称: HT 总线中断向量寄存器[95:64]

位域	名称	位宽	初值	访问	描述
31:0	Interrupt_case[95:64]	32	0x0	R/W	HT 总线中断向量寄存器[95:64]， 对应中断线 1 /HT HI 对应中断线 5

偏移量: 0x8c

复位值: 0x00000000

名称: HT 总线中断向量寄存器[127:96]

位域	名称	位宽	初值	访问	描述
31:0	Interrupt_case[127:96]	32	0x0	R/W	HT 总线中断向量寄存器[127:96]， 对应中断线 1 /HT HI 对应中断线 5

偏移量: 0x90

复位值: 0x00000000

名称: HT 总线中断向量寄存器[159:128]

位域	名称	位宽	初值	访问	描述
31:0	Interrupt_case[159:128]	32	0x0	R/W	HT 总线中断向量寄存器[159:128] , 对应中断线 2 /HT HI 对应中断线 6

偏移量: 0x94

复位值: 0x00000000

名称: HT 总线中断向量寄存器[191:160]

位域	名称	位宽	初值	访问	描述
31:0	Interrupt_case[191:160]	32	0x0	R/W	HT 总线中断向量寄存器[191:160] , 对应中断线 2 /HT HI 对应中断线 6

偏移量: 0x98

复位值: 0x00000000

名称: HT 总线中断向量寄存器[223:192]

位域	名称	位宽	初值	访问	描述
31:0	Interrupt_case[223:192]	32	0x0	R/W	HT 总线中断向量寄存器[223:192] , 对应中断线 3 /HT HI 对应中断线 7

偏移量: 0x9c

复位值: 0x00000000

名称: HT 总线中断向量寄存器[255:224]

位域	名称	位宽	初值	访问	描述
31:0	Interrupt_case[255:224]	32	0x0	R/W	HT 总线中断向量寄存器[255:224] , 对应中断线 3 /HT HI 对应中断线 7

7.3.6 中断使能寄存器

中断使能寄存器共 256 个, 与中断向量寄存器一一对应。置 1 为对应中断打开, 置 0 则为中断屏蔽。

偏移量: 0xa0

复位值: 0x00000000

名称: HT 总线中断使能寄存器[31:0]

位域	名称	位宽	初值	访问	描述
31:0	Interrupt_mask[31:0]	32	0x0	R/W	HT 总线中断使能寄存器[31:0] , 对应中断线 0 /HT HI 对应中断线 4

偏移量: 0xa4

复位值: 0x00000000

名称: HT 总线中断使能寄存器[63:32]

位域	名称	位宽	初值	访问	描述
31:0	Interrupt_mask[63:32]	32	0x0	R/W	HT 总线中断使能寄存器[63:32] , 对应中断线 0 /HT HI 对应中断线 4

偏移量: 0xa8

复位值: 0x00000000

名称: HT 总线中断使能寄存器[95:64]

位域	名称	位宽	初值	访问	描述
31:0	Interrupt_mask[95:64]	32	0x0	R/W	HT 总线中断使能寄存器[95:64] , 对应中断线 1 /HT HI 对应中断线 5

偏移量: 0xac

复位值: 0x00000000

名称: HT 总线中断使能寄存器[127:96]

位域	名称	位宽	初值	访问	描述
31:0	Interrupt_mask[127:96]	32	0x0	R/W	HT 总线中断使能寄存器[127:96] , 对应中断线 1 /HT HI 对应中断线 5

偏移量: 0xb0

复位值: 0x00000000

名称: HT 总线中断使能寄存器[159:128]

位域	名称	位宽	初值	访问	描述
31:0	Interrupt_mask[159:128]	32	0x0	R/W	HT 总线中断使能寄存器[159:128] , 对应中断线 2 /HT HI 对应中断线 6

偏移量: 0xb4

复位值: 0x00000000

名称: HT 总线中断使能寄存器[191:160]

位域	名称	位宽	初值	访问	描述
31:0	Interrupt_mask[191:160]	32	0x0	R/W	HT 总线中断使能寄存器[191:160] , 对应中断线 2 /HT HI 对应中断线 6

偏移量: 0xb8

复位值: 0x00000000

名称: HT 总线中断使能寄存器[223:192]

位域	名称	位宽	初值	访问	描述
31:0	Interrupt_mask[223:192]	32	0x0	R/W	HT 总线中断使能寄存器[223:192] , 对应中断线 3 /HT HI 对应中断线 7

偏移量: 0xbc

复位值: 0x00000000

名称: HT 总线中断使能寄存器[255:224]

位域	名称	位宽	初值	访问	描述
31:0	Interrupt_mask[255:224]	32	0x0	R/W	HT 总线中断使能寄存器[255:224] , 对应中断线 3 /HT HI 对应中断线 7

7.3.7 Interrupt Discovery & Configuration

偏移量: 0xc0

复位值: 0x80000008

名称: Interrupt Capability

位域	名称	位宽	初值	访问	描述
31:24	Capabilities Pointer	8	0x80	R	Interrupt discovery and configuration block
23:16	Index	8	0x0	R/W	读寄存器偏移地址
15:8	Capabilities Pointer	8	0x0	R	Capabilities Pointer
7:0	Capability ID	8	0x08	R	Hypertransport Capablity ID

偏移量: 0xc4

复位值: 0x00000000

名称: Dataport

位域	名称	位宽	初值	访问	描述
31:0	Dataport	32	0x0	R/W	当上一寄存器 Index 为 0x10 时, 本寄存器读写 结果为 0xa8 寄存器, 否则为 0xac

偏移量: 0xc8

复位值: 0xF8000000

名称: IntrInfo[31:0]

位域	名称	位宽	初值	访问	描述
31:24	IntrInfo[31:24]	32	0xF8	R	保留
23:2	IntrInfo[23:2]	22	0x0	R/W	IntrInfo[23:2]，当发出 PIC 中断时，IntrInfo 的值用来表示中断向量
1:0	Reserved	2	0x0	R	保留

偏移量：0xcc

复位值：0x00000000

名称： IntrInfo[63:32]

位域	名称	位宽	初值	访问	描述
31:0	IntrInfo[63:32]	32	0x0	R	保留

7.3.8 POST 地址窗口配置寄存器

本控制器中的地址窗口命中公式如下：

$$\text{hit} = (\text{BASE} \& \text{MASK}) == (\text{ADDR} \& \text{MASK})$$

值得一提的是，配置地址窗口寄存器时，MASK 高位应全为 1，低位应全为 0。MASK 中 0 的实际位数表示的就是地址窗口的大小。本窗口的地址是 AXI 总线上接收到的地址。落在本窗口的所有写访问将立即在 AXI B 通道返回，并以 POST WRITE 的命令格式发给 HT 总线。而不在本窗口的写请求则以 NONPOST WRITE 的方式发送到 HT 总线，并等待 HT 总线响应后再返回 AXI 总线。

偏移量：0xd0

复位值：0x00000000

名称： HT 总线 POST 地址窗口 0 使能（内部访问）

位域	名称	位宽	初值	访问	描述
31	ht_post0_en	1	0x0	R/W	HT 总线 POST 地址窗口 0，使能信号
30:23	Reserved	15	0x0		保留
15:0	ht_post0_trans[39:24]	16	0x0	R/W	HT 总线 POST 地址窗口 0，转译后地址的[39:24]

偏移量：0xd4

复位值：0x00000000

名称： HT 总线 POST 地址窗口 0 基址（内部访问）

位域	名称	位宽	初值	访问	描述
31:16	ht_post0_base[39:24]	16	0x0	R/W	HT 总线 POST 地址窗口 0，地址基址的[39:24]
15:0	ht_post0_mask[39:24]	16	0x0	R/W	HT 总线 POST 地址窗口 0，地址屏蔽的[39:24]

偏移量: 0xd8

复位值: 0x00000000

名称: HT 总线 POST 地址窗口 1 使能 (内部访问)

位域	名称	位宽	初值	访问	描述
31	ht_post1_en	1	0x0	R/W	HT 总线 POST 地址窗口 1, 使能信号
30:23	Reserved	15	0x0		保留
15:0	ht_post1_trans[39:24]	16	0x0	R/W	HT 总线 POST 地址窗口 1, 转译后地址的[39:24]

偏移量: 0xdc

复位值: 0x00000000

名称: HT 总线 POST 地址窗口 1 基址 (内部访问)

位域	名称	位宽	初值	访问	描述
31:16	ht_post1_base[39:24]	16	0x0	R/W	HT 总线 POST 地址窗口 1, 地址基址的[39:24]
15:0	ht_post1_mask[39:24]	16	0x0	R/W	HT 总线 POST 地址窗口 1, 地址屏蔽的[39:24]

7.3.9 可预取地址窗口配置寄存器

本控制器中的地址窗口命中公式如下:

$$\text{hit} = (\text{BASE} \& \text{MASK}) == (\text{ADDR} \& \text{MASK})$$

值得一提的是, 配置地址窗口寄存器时, MASK 高位应全为 1, 低位应全为 0。MASK 中 0 的实际位数表示的就是地址窗口的大小。本窗口的地址是 AXI 总线上接收到的地址。落在本窗口的取指指令, CACHE 访问才会被发往 HT 总线, 其它的取指或是 CACHE 访问将不会被发往 HT 总线, 而是立即返回, 如果是读命令, 则会返回相应个数的无效读数据。

偏移量: 0xe0

复位值: 0x00000000

名称: HT 总线可预取地址窗口 0 使能 (内部访问)

位域	名称	位宽	初值	访问	描述
31	ht_prefetch0_en	1	0x0	R/W	HT 总线可预取地址窗口 0, 使能信号
30:23	Reserved	15	0x0		保留
15:0	ht_prefetch0_trans[39:24]	16	0x0	R/W	HT 总线可预取地址窗口 0, 转译后地址的[39:24]

偏移量: 0xe4

复位值: 0x00000000

名称： HT 总线可预取地址窗口 0 基址（内部访问）

位域	名称	位宽	初值	访问	描述
31:16	ht_prefetch0_base[39:24]	16	0x0	R/W	HT 总线可预取地址窗口 0，地址基址的[39:24]位地址
15:0	ht_prefetch0_mask[39:24]	16	0x0	R/W	HT 总线可预取地址窗口 0，地址屏蔽的[39:24]

偏移量： 0xe8

复位值： 0x00000000

名称： HT 总线可预取地址窗口 1 使能（内部访问）

位域	名称	位宽	初值	访问	描述
31	ht_prefetch1_en	1	0x0	R/W	HT 总线可预取地址窗口 1，使能信号
30:23	Reserved	15	0x0		保留
15:0	ht_prefetch1_trans[39:24]	16	0x0	R/W	HT 总线可预取地址窗口 1，转译后地址的[39:24]

偏移量： 0xec

复位值： 0x00000000

名称： HT 总线可预取地址窗口 1 基址（内部访问）

位域	名称	位宽	初值	访问	描述
31:16	ht_prefetch1_base[39:24]	16	0x0	R/W	HT 总线可预取地址窗口 1，地址基址的[39:24]
15:0	ht_prefetch1_mask[39:24]	16	0x0	R/W	HT 总线可预取地址窗口 1，地址屏蔽的[39:24]

7.3.10 UNCACHE 地址窗口配置寄存器

本控制器中的地址窗口命中公式如下：

$$\text{hit} = (\text{BASE} \& \text{MASK}) == (\text{ADDR} \& \text{MASK})$$

$$\text{addr_out} = \text{TRANS_EN} ? \text{TRANS} | \text{ADDR} \& \text{MASK} : \text{ADDR}$$

值得一提的是，配置地址窗口寄存器时，MASK 高位应全为 1，低位应全为 0。MASK 中 0 的实际位数表示的就是地址窗口的大小。本窗口的地址是 HT 总线上接收到的地址。落在本窗口地址的读写命令，将不会被送往二级 CACHE，也不会使得一级 CACHE 发生失效，而是被直接送至内存，或是其它的地址空间。这也就是说这个地址窗口中的读写命令将不会维持 IO 的 CACHE 一致性。这一窗口主要针对一些不会在 CACHE 中命中而可以提高存问效率的操作，如显存的访问等。

偏移量： 0xf0

复位值： 0x00000000

名称： HT 总线 Uncache 地址窗口 0 使能（内部访问）

位域	名称	位宽	初值	访问	描述
31	ht_uncache0_en	1	0x0	R/W	HT 总线 uncache 地址窗口 0, 使能信号
30	ht_uncache0_trans_en	1	0x0	R/W	HT 总线 uncache 地址窗口 1, 映射使能信号
29:23	Reserved	14	0x0		保留
15:0	ht_uncache0_trans[39:24]	16	0x0	R/W	HT 总线 uncache 地址窗口 0, 转译后地址的[39:24]

偏移量: 0xf4

复位值: 0x00000000

名称: HT 总线 Uncache 地址窗口 0 基址 (内部访问)

位域	名称	位宽	初值	访问	描述
31:16	ht_uncache0_base[39:24]	16	0x0	R/W	HT 总线 uncache 地址窗口 0, 地址基址的[39:24]
15:0	ht_uncache0_mask[39:24]	16	0x0	R/W	HT 总线 uncache 地址窗口 0, 地址屏蔽的[39:24]

偏移量: 0xf8

复位值: 0x00000000

名称: HT 总线 Uncache 地址窗口 1 使能 (内部访问)

位域	名称	位宽	初值	访问	描述
31	ht_uncache1_en	1	0x0	R/W	HT 总线 uncache 地址窗口 1, 使能信号
30	ht_uncache1_trans_en	1	0x0	R/W	HT 总线 uncache 地址窗口 1, 映射使能信号
29:23	Reserved	14	0x0		保留
15:0	ht_uncache1_trans[39:24]	16	0x0	R/W	HT 总线 uncache 地址窗口 1, 转译后地址的[39:24]

偏移量: 0xfc

复位值: 0x00000000

名称: HT 总线 Uncache 地址窗口 1 基址 (内部访问)

位域	名称	位宽	初值	访问	描述
31:16	ht_uncache1_base[39:24]	16	0x0	R/W	HT 总线 uncache 地址窗口 1, 地址基址的[39:24]
15:0	ht_uncache1_mask[39:24]	16	0x0	R/W	HT 总线 uncache 地址窗口 1, 地址屏蔽的[39:24]

第八章 PCIE 控制器

8.1 使用说明

龙芯 2H 在 PCIE 总线上既可以充当 RC（root complex）也可以充当 EP（end point）。在充当 RC 时，龙芯 2H 的 PCIE 接口既可以充当一个 X4 的 PCIE 端口也可以充当 4 个独立的 X1 的 PCIE 端口；在充当 EP 时，龙芯 2H 的 PCIE 接口可充当 1 个 X4/X1 的 PCIE 接口。

图 1 为龙芯 2H 的 PCIE 控制器结构示意。龙芯 2H 的 PCIE 控制器包含 0 ~ 3 号，共 4 个 PCIE 端口。0 号端口可以以 X4/X1 的方式工作，1 ~ 3 号端口仅能以 X1 的方式工作。作为 RC 时 0 ~ 3 号端口可用，但在 X4 模式下仅 0 号端口可用。作为 EP 使用时，仅 0 号端口可用。每个 PCIE 端口均有自己独立的 PCI 地址空间。

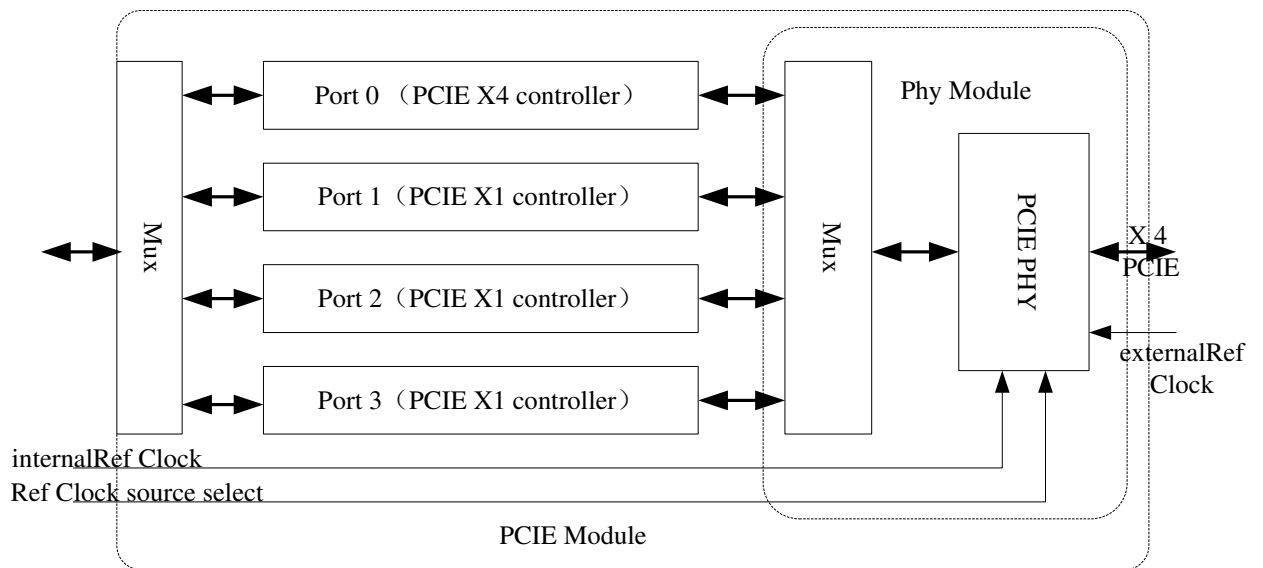


图 8.1: PCIE 控制器结构

8.2 地址空间划分和地址转换机制

龙芯 2H 处理器的 PCIE 控制器的地址空间划分在 RC 模式和 EP 模式下被区别对待。

8.2.1 RC 模式下的地址空间划分

作为 RC 时所占用的空间分为 3 部分：mem0、cfg&ctrl、mem1。

Mem0 从 0x10000000 起始，大小为 128MB。在这段空间中处理器的物理地址和 PCIE 端口上的 PCI 地址相同。这段空间被用于访问设备中被映射为 MEM 空间的资源。这段空间可以由一个 X4 端口独享，也可以由 4 个 X1 端口分享。Mem0 空间被 X1 端口分享时，每个端口均有 32MB 大小的空间。

表 8.1: mem0 空间划分

起始地址	大小	所属端口号
0x10000000	32MB	0
0x12000000	32MB	1 (4X1) /0 (X4)
0x14000000	32MB	2 (4X1) /0 (X4)
0x16000000	32MB	3 (4X1) /0 (X4)

Mem1 从 0x40000000 起始，大小为 1GB。这段空间处理器的物理地址和 PCIE 端口上的 PCI 地址相同。这段空间被用于访问需要映射较大 MEM 空间的设备。这段空间可以由一个 X4 端口独享，也可以由 4 个 X1 端口分享。Mem1 空间被 X1 端口分享时，每个端口均有 256MB 大小的空间。

表 8.2: mem1 空间划分

起始地址	大小	所属端口号
0x40000000	256MB	0
0x50000000	256MB	1 (4X1) /0 (X4)
0x60000000	256MB	2 (4X1) /0 (X4)
0x70000000	256MB	3 (4X1) /0 (X4)

Cfg&ctrl 空间从 0x18000000 起始，大小为 16MB。这个空间由 0 ~ 3 号 4 个 PCIE 端口分享，每个端口均有 4MB 空间。当 PCIE 控制器在 X4 模式工作时，1 ~ 3 号所属的 cfg&ctrl 空间不可访问。表 8.3 给出了这段空间的划分，其中未列出的地址空间为保留空间。

龙芯 2H 处理器 PCIE 模块的内部寄存器在 cfg&ctrl 段中。龙芯 2H 的 PCIE 控制器内部的寄存器包含 3 部分：每个端口都有的 PCIE 头控制寄存器、每个端口都有的端口控制寄存器、phy 控制寄存器。其中对 phy 控制寄存器的访问以字节访问的形式进行，而其它的寄存器则以 32 位的 word 方式进行。当 PCIE 控制器以 X4 方式工作时，仅能访问端口 0 所属寄存器。

表 8.3: RC 模式下 cfg&ctrl 空间划分

起始地址	大小	用途	备注
0x18000000	1MB	对 PCIE 控制器 port 0 的 PCI 空间起始地址为 0x00000000 的最低 1MB 空间进行 mem 访问	
0x18100000	64KB	对 PCIE 控制器 port 0 的 PCI I/O 访问	
0x18110000	16KB	对 PCIE 控制器的 phy 进行控制和配置访问	以字节访问的形式进行

起始地址	大小	用途	备注
0x18114000	8KB	对 PCIE 控制器 port0 本地的 PCIE 配置头空间进行访问	
0x18116000	4KB	对 PCIE 控制器 port0 所连接的 PCIE 设备发起配置访问	
0x18118000	4KB	对配置和控制 PCIE 控制器 port0 的内部寄存器进行访问	
0x18400000	1MB	对 PCIE 控制器 port 1 的 PCI 空间起始地址为 0x00000000 的最低 1MB 空间进行 mem 访问	X4 模式时, 此段空间保留
0x18500000	64KB	对 PCIE 控制器 port 1 的 PCI I/O 访问	X4 模式时, 此段空间保留
0x18514000	8KB	对 PCIE 控制器 port1 本地的 PCIE 配置头空间进行访问	X4 模式时, 此段空间保留
0x18516000	4KB	对 PCIE 控制器 port1 所连接的 PCIE 设备发起配置访问	X4 模式时, 此段空间保留
0x18518000	4KB	对配置和控制 PCIE 控制器 port1 的内部寄存器进行访问	X4 模式时, 此段空间保留
0x18800000	1MB	对 PCIE 控制器 port 2 的 PCI 空间起始地址为 0x00000000 的最低 1MB 空间进行 mem 访问	X4 模式时, 此段空间保留
0x18900000	64KB	对 PCIE 控制器 port 2 的 PCI I/O 访问	X4 模式时, 此段空间保留
0x18914000	8KB	对 PCIE 控制器 port2 本地的 PCIE 配置头空间进行访问	X4 模式时, 此段空间保留
0x18916000	4KB	对 PCIE 控制器 port2 所连接的 PCIE 设备发起配置访问	X4 模式时, 此段空间保留
0x18918000	4KB	对配置和控制 PCIE 控制器 port2 的内部寄存器进行访问	X4 模式时, 此段空间保留
0x18c00000	1MB	对 PCIE 控制器 port 3 的 PCI 空间起始地址为 0x00000000 的最低 1MB 空间进行 mem 访问	X4 模式时, 此段空间保留
0x18d00000	64KB	对 PCIE 控制器 port 3 的 PCI I/O 访问	X4 模式时, 此段空间保留
0x18d14000	8KB	对 PCIE 控制器 port3 本地的 PCIE 配置头空间进行访问	X4 模式时, 此段空间保留
0x18d16000	4KB	对 PCIE 控制器 port3 所连接的 PCIE 设备发起配置访问	X4 模式时, 此段空间保留
0x18d18000	4KB	对配置和控制 PCIE 控制器 port3 的内部寄存器进行访问	X4 模式时, 此段空间保留

8.2.2 EP 模式下的地址空间划分

在 EP 模式下, 除了 mem0、cfg&ctrl、mem1 这 3 段空间外的所有访问都被直接映射为 PCI 地址空间上的 mem 访问。Mem0 和 mem1 在 EP 模式的使用和 RC 模式相同, cfg&ctrl 段地址的使用由表 8.4 给出。

表 8.4: EP 模式下 cfg&ctrl 空间划分

起始地址	大小	用途	备注
0x18000000	1MB	保留	
0x18100000	64KB	保留	
0x18110000	16KB	对 PCIE 控制器的 phy 进行控制和配置访问	以字节访问的形式进行
0x18114000	8KB	对 PCIE 控制器 port0 本地的 PCIE 配置头空间进行访问	
0x18116000	4KB	保留	

起始地址	大小	用途	备注
0x18118000	4KB	对配置和控制 PCIE 控制器 port0 的内部寄存器进行访问	
0x18400000	12MB	保留	

8.2.3 PCI 地址空间到物理地址的转换

龙芯 2H 的 PCIE 控制器在 RC 模式和 EP 模式下拥有不同的 PCI 地址空间到处理器的物理地址空间的地址转换机制。

在 RC 模式下，来自 PCIE 总线的请求需要从 PCI 地址空间转换到处理器内部的物理地址空间。这个转换过程由 `cfg&ctrl` 地址段中的一组寄存器控制：`at_en`、`addr_decoder_mask`、`addr_decoder_trans`。当 `at_en`（`0x18118000` 的第 24 位，此位上电值为 0）为 0 时，PCI 地址被直接作为物理地址使用。当 `at_en` 为 1 时，物理地址[63:12] = (PCI 地址[63:12]& `addr_decoder_mask`)— `addr_decoder_trans`，物理地址[11:0] = PCI 地址[11:0]。

在 EP 模式下，来自 PCIE 总线的请求先经过 3 个 64 位的 PCI base 译码，译码后的地址根据所命中的不同 PCI base 进行地址变换。龙芯 2H 在作为 EP 时有 3 个 64 位的 PCI base。

Base 0 的大小为 128MB。命中这个地址窗口的请求的[63:32]位被赋值为 0，[31:27]位被赋值为 5'b00011，[26:0]位保持不变。这个窗口对应处理器内部物理地址从 `0x18000000` ~ `0x1ffffff` 的范围。

Base1 的大小为 256MB。命中这个地址窗口的请求的[63:28]位被赋值为 0，[27:0]位保持不变。这个窗口被用来映射 DDR 内存的低 256MB 空间。

Base2 的大小可配，命中这个地址窗口的请求的地址不做变化。这个窗口在上电后是关闭的。需要通过编程配置其大小并将其使能后才能使用。Base2 的使用需要 `base2` 所属寄存器进行本地配置。下面的 c 代码例子将说明如何将 `base2` 配置为一个大小为 2GB 的可以预取的 `mem` 窗口。

```
//set bar4 to 2GB bar and enable bar4
*(volatile unsigned int *)0xb8115020 = 0x80000001;//[0] set to 1 to enable bar4,
the upper bit is mask
*(volatile unsigned int *)0xb8115024 = 0xffffffff; //set upper 32 bit mask
//set bar4 to 64 bit memory prefetchable bar
*(volatile unsigned int *)0xb8114020 = 0x0000000c;
```

8.3 龙芯 2H 处理器 PCIE 控制器内部寄存器定义

龙芯 2H 的 4 个 PCIE 端口都有自己的端口控制寄存器。每个端口的控制寄存器都有自己的基址。端口 0 的端口控制寄存器基址为 `0x18118000`，端口 1 的端口控制寄存

器基址为 0x18158000，端口 2 的端口控制寄存器基址为 0x18198000，端口 3 的端口控制寄存器基址为 0x181d8000。由于 0 ~ 3 号端口的端口控制寄存器在命名和使用方法上相同，为了节省篇幅，我们在此仅介绍端口 0 的端口控制寄存器定义。端口 1 的控制寄存器地址为端口 0 的控制寄存器地址+ 0x400000。端口 2 的控制寄存器地址为端口 0 的控制寄存器地址+ 0x800000。端口 3 的控制寄存器地址为端口 0 的控制寄存器地址+ 0xc00000。对只占 1 位的寄存器，如非特别说明均为高有效。

表 8.5: 寄存器属性类型

属性名称	描述
RO	只读
RW	可读可写
RW1C	只读，对应位写入 1 则将该位清 0
ROS	只读，并且寄存器的值不会因为热复位而发生变化
RWS	可读可写，并且寄存器的值不会因为热复位而发生变化
RW1P	可读可写，并且如果向寄存器写入 1，寄存器的值下 1 个时钟周期会自动清 0。因此读此寄存器仅能获得 0 值

8.3.1 PCIE 端口控制寄存器 0

地址: 0x18118000

表 8.6: PCIE 端口控制寄存器 0

位域	初值	属性	名称	描述
0	0	RW	Rx_lane_flip.en	PCIE 接收线反转 when set to 1, Performs lane reversal for receive lanes. for use when automatic lane reversal does not occur because lane 0 is not detected.
1	0	RW	Tx_lane_flip.en	PCIE 发送线反转 when set to 1, initiate lane reversal for transmit lanes. for use when automatic lane reversal does not occur because lane 0 is not detected.
2	1	RW	Sys_aux_pwr_det	指示拥有辅助电源 (Vaux) Auxiliary power detected. indicates that auxiliary power(Vaux) is present.
3	0(RC) 1(EP)	RW	App.ltssm.enable	PCIE 端口链路建立使能 When set to 1, enables PCIE link train start.
11:4	0x4	RW		Reserved Should be set to 0x0 after reset
12	0	RW	App.req_enter_L1	要求 PCIE 链路进入 L1 Application request to enter L1. When set to 1, indicates request from the application to enter ASPM state L1. only effective if L1 is enabled.

位域	初值	属性	名称	描述
13	0	RW	App_ready_enter_L23	已经准备好让 PCIE 链路进入 L23 Application ready to enter L23. When set to 1, indicates from the application that it is ready to enter L23 state.
14	0	RW	App_req_exit_L1	要求 PCIE 端口退出 L1 Application request to exit L1. When set to 1, indicates request from the application to exit ASPM state L1, only effective if L1 is enabled.
15	0	RW	Soft_reset_en	软复位使能 When set to 1, enables soft reset on this PCIE port
23:16	0xff	RW		Reserved
24	0	RW	at_en	.PCIE 端口在 RC 工作模式下入站地址转换使能 When set to 1, enables PCI address to physical address translate in RC mode.
25	0	RW	bus_error_en	PCIE 读返回总线错误的处理方式 0：读数据返回全 1，不产生总线错例外 1：产生总线错例外 当 PCIE 控制器完成总线扫描和初始化后，此位可以修改为 1
27:26	0x0	RO		Reserved
31:28	由芯片的 PCIE[3:0]_PRSNT 引脚的值确定	RO		设备挂载状态 28 到 31 位分别对应插槽 0 3 上设备的挂载状态，每一位的含义如下： 1：插槽中有设备 0：插槽中无设备

8.3.2 PCIE 端口控制寄存器 1

地址：0x18118004

通过对相应位写入 1 产生一个时钟周期的脉冲，控制 PCIE 接口进行相应的操作。写此寄存器时，一次仅能有 1 位被置 1。

表 8.7: PCIE 端口控制寄存器 1

位域	初值	属性	名称	描述
0	0	RW1P	App_unlock_msg	在 PCIE 端口上发送解锁消息 When set to 1, indicates the application request an unlock message. This bits will be automatically cleared.
1	0	RW1P	Apps_pm_xmt_turnoff	在 PCIE 端口上发送 PM_Turn_Off 消息 When set to 1, indicates the application generate a PM_Turn_Off message. This bits will be automatically cleared.

位域	初值	属性	名称	描述
2	0	RW1P	App_init_rst	在 PCIE 端口上发送热复位消息 when set to 1, the Hot Reset request is sent to the downstream device. this bits will be automaticly cleared.
3	0	RW1P	Soft_reset	对 PCIE 端口进行软复位 When set to 1, generate a soft reset on this PCIE port, only effective if soft reset of this port is enabled
4	0	RW1P	Apps_pm_xmt_pme	将 PCIE 端口从 D1 或者 D2 或者 D3 状态中唤醒, 并发送 PM_PME 消息 Set to 1 to wake up the PMC state from a D1, D2 or D3 power state. Upon wake-up, this port sends a PM.PME message. this bits will be automaticly cleared.
31:5	0x0	RO		Reserved

8.3.3 PCIE 端口状态寄存器 0

地址: 0x18118008

只读寄存器

表 8.8: PCIE 端口状态寄存器 0

位域	名称	描述
1:0	Cfg_pwr_ind	系统电源状态指示 the system power indicator(from bits [9:8] of the Slot Control register) 00b: Reserved 01b: On 10b: Blink 11b: Off
3:2	Cfg_atten_ind	Attention 按钮状态指示 Indicates the state of the system attention indicator(from bits [7:6] of the Slot Control register) 00b: Reserved 01b: On 10b: Blink 11b: Off
4	Cfg_pwr_ctrl	系统电源控制器状态 Indicates the state of the system power controller(from bits 10 of the Slot Control register) 0: Power On 1: Power Off

位域	名称	描述
5	Pm_xtlh_block_tlp	禁止发出请求指示 When set to 1, indicates that the application must stop generating new outgoing request TLPs due to the current power management state. The application can continue to generate completion TLPs.
6	Cfg_bus_master_en	PCI 主设备使能 The state of the Bus Master Enable bit in the PCI-compatible command register. 1: enabled 0: disabled
7	Cfg_mem_space_en	内存映射访问使能 The state of the Memory Space Enable bit in the PCI-compatible command register. 1: enabled 0: disabled
10:8	Cfg_max_rd_req_size	最大读请求大小 The value of the Max_Read_Request_Size field in the Device Control register.
13:11	Cfg_max_payload_size	最大数据负载 The value of the Max_Payload_Size field in the Device Control register.
14	Cfg_rcb	RCB 位 The value of the RCB bit in the Link Control register.
15	Rdlh_link_up	数据链路层状态 Data Link Layer up/down indicator 1: Link is up 0: Link is down
18:16	Pm_curnt_state	当前电源状态 Indicates the current power state.
23:19	Cfg_aer_int_msg_num	根错误状态寄存器的 31:21 位 Form bits [31:27] of the Root Error Status register, used when MSI is enabled.
28:24	Cfg_pcie_cap_int_msg_num	PCIE 能力寄存器 13:9 位 Form bits [13:9] of the PCI Express Capabilities register, used when MSI is enabled.
31:29		Reserved

8.3.4 PCIE 端口状态寄存器 1

地址: 0x1811800c

只读寄存器

表 8.9: PCIE 端口状态寄存器 1

位域	名称	描述
5:0	Xmlh_ltssm_state	LTSSM 状态机的当前状态 current state of the LTSSM

位域	名称	描述
6	Xmlh_link_up	物理链路状态指示 PHY link up/down indicator 1: Up 0: Down
7	Rtlh_rfc_upd	数据链路层收到流控包指示 Indicates that the port received a flow control update DLLP. none useful for user.
8	Cfg_eml_control	Electromechanical interlock control. The state of the Electromechanical interlock control bit in the Slot Control register.
16:9	Cfg_pbus_num	设备所属总线号 The primary bus number assigned to the function.
21:17	Cfg_pbus_dev_num	设备号 The device number assigned to the function.
24:22	Pm_dstate	电源管理状态指示 The current power management D-state of the function: 000b: D0 001b: D1 010b: D2 011b: D3 100b: Uninitialized Other values: Not applicable
25	Pm_status	电源管理状态位 PME status bit from PMCSR
26	Pm_pme_en	PME 使能指示 PME enable bit in the PMCSR
27	Aux_pm_en	辅助电源使能指示 Auxiliary power enable bit in the Device Control register
28	Cfg_link_auto_bw_int	链路自治带宽状态寄存器更新指示 A notification when the Link Autonomous Bandwidth Status register (Link Status register bit 15) is updated and the Link Autonomous Bandwidth interrupt enable(Link Control register bit 11) is set. This bit is not applicable to, and is reserved, for EP devices and upstream ports of switch.
29	Cfg_bw_mgt_int	链路带宽管理状态寄存器更新指示 A notification when the Link Bandwidth Management Status register (Link Status register bit 14) is updated and the Link Bandwidth Management interrupt enable(Link Control register bit 10) is set. This bit is not applicable to, and is reserved, for EP devices and upstream ports of switch.
31:30		Reserved

8.3.5 用户定义消息 ID 寄存器

地址：0x18118010

只读

表 8.10: 用户定义消息 ID 寄存器

位域	名称	描述
15:0	Radm_msg_req_id	用户定义消息所携带的 ID The requester ID of vender-defined message.
31:16		Reserved

8.3.6 PCIE 端口中断状态寄存器

地址：0x18008018

只读。对应的位为 1，表示有该位代表的中断。

表 8.11: PCIE 端口中断状态寄存器

位域	名称	描述
0	aer_rc_err_int	当 RC 产生或收到错误消息并且 Root Error Command 寄存器中对应的错误报告使能被打开时置位。此位仅在 RC 工作模式下使用。 This bit is set when a reported error condition causes a bit to be set in the Root Error Status register and the associated error message reporting enable bit is set in the Root Error Command register. This bit is set when an internally generated error message is generated/received by the RC.
1	aer_rc_err_msi	当 RC 产生或收到错误消息并且 MSI 被使能且 Root Error Command 寄存器中对应的错误报告使能被打开时置位。此位仅在 RC 工作模式下使用。 This bit is set when MSI is enabled and a reported error condition causes a bit to be set in the Root Error Status register and the associated error message reporting enable bit is set in the Root Error Command register. This bit is set when an internally generated error message is generated/received by the RC.
2	sys_err_rc	此位报告检测到系统错误。当 Root Control 寄存器的对应位使能被打开并且 PCIE 总线上如何设备产生：可修复错误、致命错误、非致命错误时，或者 RC 产生内部错误时置位。 System error detected. This bit is set to indicate if any device in the hierarchy reports internal error and any of the following errors and the associated enable bit is set in the Root Control register: ERR_COR, ERR_FATAL, ERR_NONFATAL

位域	名称	描述
3	pme_int	<p>收到 PME 中断。当下列条件满足时，此位置位：</p> <ol style="list-style-type: none"> 1. Command 寄存器中 INTx 未被禁止； 2. Root Control 寄存器中 PME 中断使能位被打开； 3. 收到 PME 消息（Root Status 寄存器中 PME 状态位被置位）。 <p>This bit is set when all of the following conditions are true:</p> <p>The INTx assertion disable bit in the Command register is 0.</p> <p>The PME interrupt enable bit in the Root Control register is set to 1.</p> <p>The PME status bit in the Root Status register is set to 1.</p>
4	pme_msi	<p>收到 PME 中断。当下列条件满足时，此位置位：</p> <ol style="list-style-type: none"> 1. Command 寄存器中 INTx 被禁止, MSI 被使能； 2. Root Control 寄存器中 PME 中断使能位被打开； 3. 收到 PME 消息（Root Status 寄存器中 PME 状态位被置位）。 <p>This bit is set when all of the following conditions are true:</p> <p>The MSI is enabled.</p> <p>The PME interrupt enable bit in the Root Control register is set to 1.</p> <p>The PME status bit in the Root Status register is set to 1.</p>
5	vendor_msg	<p>收到用户定义消息</p> <p>This bit is set when the PCIE port received a vendor-defined message.</p>
6	rc_core_wake	<p>从电源管理中唤醒</p> <p>Wake up from power management unit.</p>
7	INTA	<p>INTA 中断</p> <p>This bit is set when INTA asserted message is received.</p>
8	INTB	<p>INTB 中断</p> <p>This bit is set when INTB asserted message is received.</p>
9	INTC	<p>INTC 中断</p> <p>This bit is set when INTC asserted message is received.</p>
10	INTD	<p>INTD 中断</p> <p>This bit is set when INTD asserted message is received.</p>
11	radm_correctable_err	<p>此 PCIE 端口收到可修复错误消息</p> <p>received an ERR_COR message</p>
12	radm_nonfatal_err	<p>此 PCIE 端口收到非致命错误消息</p> <p>received an ERR_NONFATAL message</p>
13	radm_fatal_err	<p>此 PCIE 端口收到致命错误消息</p> <p>received an ERR_FATAL message</p>
14	pm_pme	<p>收到 PM_PME 消息</p> <p>received a PM_PME message</p>

位域	名称	描述
15	pm_to_ack	收到 PM_TO_Ack 消息 received a PM_TO_Ack message
16	hp_pme	当下列条件满足时，此位置位： 1. PCI 电源管理控制和状态寄存器中 PME 使能被打开； 2. Slot Status 寄存器中的任意 1 位由 0 变 1 并且对应位的通知使能被打开。 This bit is set when all of the following conditions are true The PME enable bit in the Power Management Control and Status register is set to 1 Any bit in the Slot Status register transitions from 0 to 1 and the associated event notification is enabled in the Slot Control register.
17	hp_int	当下列条件满足时，此位置位： 1. Command 寄存器中 INTx 未被禁止； 2. Slot Control 寄存器中热插拔中断使能被打开； 3. Slot Status 寄存器中的任意 1 位为 1，并且对应位的通知使能被打开。 This bit is set when all of the following conditions are true The INTx assertion disable bit in the Command register is 0. Hot-Plug interrupts are enabled in the Slot Control register. Any bit in the Slot Status register is equal to 1, and the associated event notification is enabled in the Slot Control register.
18	hp_msi	当下列条件满足时，此位置位： 1. MSI 使能被打开； 2. Slot Control 寄存器中热插拔中断使能被打开； 3. Slot Status 寄存器中的任意 1 位从 0 变 1，并且对应位的通知使能被打开。 This bit is set when all of the following conditions are true The MSI is enabled. Hot-Plug interrupts are enabled in the Slot Control register. Any bit in the Slot Status register transitions from 0 to 1, and the associated event notification is enabled in the Slot Control register.

位域	名称	描述
19	link_auto_bw_int	当链路的自治带宽状态寄存器被更新并且链路的自治带宽中断被使能时置位仅在 RC 模式下使用。 This bit is set when the link Autonomous Bandwidth Status register is updated and the Link Autonomous Bandwidth Interrupt enable is set. Only used by RC mode
20	bw_mgt_int	当链路的带宽状态寄存器被更新并且链路的带宽中断被使能时置位。仅在 RC 模式下使用。 This bit is set when the link Bandwidth Management Status register is updated and the Link Bandwidth Interrupt enable is set. Only used by RC mode
21	gm_composer_lookup_err	此位被置位表示此 PCIE 端口在向外发送数据响应时溢出。这通常表示此 PCIE 端口接收端收到的 Non-Posted 请求的数量超出了端口流控限制。 This bit is set when the an overflow that occurred in a lookup table of the outbound responses. This indicates that there was a violation for the number of outstanding Non-Posted requests issued for the inbound direction.
22	radmx_composer_lookup_err	此位被置位表示此 PCIE 端口接收送数据响应时溢出。这通常表示此 PCIE 端口发送端发送的 Non-Posted 请求的数量超出了端口流控限制。 This bit is set when the an overflow that occurred in a lookup table of the inbound responses. This indicates that there was a violation for the number of outstanding Non-Posted requests issued for the outbound direction.
23	phy_int	PHY 中断
24		Reserved
25	ltssm_l2_to_detect	LTSSM 状态从 L2 状态退出到设备检测状态 LTSSM state exit L2 and enter detect state
26	pm_turn_off	收到 PM_Turn_Off 消息 received PM_Turn_Off message.
27	Link_req_rst_not_fall	PCIE 端口物理链路断裂
31:28		Reserved

8.3.7 PCIE 端口中断状态清除寄存器

地址：0x1811801c

RW1C。在某一位写入 1 则清除 PCIE 端口中断状态寄存器的对应位。

8.3.8 PCIE 端口中断掩码寄存器

地址：0x18118020

RW。PCIE 端口中断状态寄存器对应的中断掩码。哪 1 位置 1 并且中断状态寄存器的相应位也为 1 时，此 PCIE 端口产生中断。

8.3.9 PCIE 端口对外配置访问参数寄存器

地址：0x18118024

表 8.12: PCIE 端口对外配置访问参数寄存器

位域	名称	描述
0	cfg_type	0：发起 type 0 访问 1：发起 type 1 访问
7:1		Reserved
23:8	cfg_addr_high	PCI 配置访问的 31:16 位
31:24		Reserved

8.3.10 PCIE 端口控制和状态寄存器

地址：0x18118028

表 8.13: PCIE 端口控制和状态寄存器

位域	初值	属性	名称	描述
0	0	RW	func_bypass	置 1 时，取消内部接口请求间的先后顺序限制（读可能会越过写，写也可能越过读）
7:1				Reserved
8	0	RW	ssc_en	让 PHY 进入 SSC 模式
15:9				Reserved
16	0	RW	aux_clk_en	PCIE 端口辅助时钟使能
23:17				Reserved
24	1	RW	tx_lspd	PHY 发送端在 PCIE gen1 工作模式下使用低功耗模式工作
25	1	RW	rx_lspd	PHY 接收端在 PCIE gen1 工作模式下使用低功耗模式工作
26		RO	X4_mode	PCIE 端口在 X4 模式工作
27		RO	Is_rc	PCIE 端口在 RC 模式工作
28		RO	L0s	PCIE 端口处于 L0s 功耗状态
29		RO	L1	PCIE 端口处于 L1 功耗状态
30		RO	L2	PCIE 端口处于 L2 功耗状态
31		RO	L2_exit	PCIE 端口退出 L2 功耗状态

8.3.11 PHY 状态寄存器

地址：0x1811802c

RO

表 8.14: PHY 状态寄存器

位域	名称	描述
3:0	Phy_ready	0 到 3 位分别表示 PHY 的 0 到 3 号端口是否准备好。在 X1 模式下，仅第 0 位有效。
31:4		Reserved

8.3.12 用户定制消息寄存器 0

地址: 0x18118038

RW。此寄存器用于在发送用户定制 PCIE 消息前存储 PCIE 传输层协议消息包头的头 4 个字节的信息。

表 8.15: 用户定制消息寄存器 0

位域	名称	描述
1:0	ven_msg_fmt	PCIE 协议中传输层协议包头中的 Fmt 域
6:2	ven_msg_type	PCIE 协议中传输层协议包头中的 Type 域
9:7	ven_msg_tc	PCIE 协议中传输层协议包头中的 TC 域
10	ven_msg_td	PCIE 协议中传输层协议包头中的 TD 域
11	ven_msg_ep	PCIE 协议中传输层协议包头中的 EP 域
13:12	ven_msg_attr	PCIE 协议中传输层协议包头中的 Attr 域
23:14	ven_msg_len	PCIE 协议中传输层协议包头中的 Length 域
31:24		Reserved

8.3.13 用户定制消息寄存器 1

地址: 0x1811803c

RW。此寄存器用于在发送用户定制 PCIE 消息前存储 PCIE 传输层协议消息包头的第 4 到第 7 字节的消息。

表 8.16: 用户定制消息寄存器 1

位域	名称	描述
2:0	ven_msg_fun_num	PCIE 协议中传输层协议包头中的 Function Number 域
10:3	ven_msg_tag	PCIE 协议中传输层协议包头中的 Tag 域
18:11	ven_msg_code	PCIE 协议中传输层协议包头中的 Message Code 域
22:19		Reserved
23	ven_msg_req_valid	此位置 1 表示当前用户定制消息的所有参数均已经配置完毕，要求 PCIE 端口发送此用户定制消息。消息发送完毕后，此位自动清 0。
31:24		Reserved

8.3.14 用户定制消息数据发送寄存器 0

地址: 0x18118040

RW。用于存储待发送用户定制消息所携带数据的低 32 位。

8.3.15 用户定制消息数据发送寄存器 1

地址：0x18118044

RW。用于存储待发送用户定制消息所携带数据的高 32 位。

8.3.16 MSI 参数寄存器

地址：0x1811805c

RW。用于存储 MSI 消息的参数。

表 8.17: MSI 参数寄存器

位域	名称	描述
4:0	ven_msi_vector	PCI 协议 MSI 包中的 Vector 域
7:5	ven_msg_tc	PCIE 协议中传输层协议包头中的 TC 域
10:8	ven_msg_func_num	PCIE 协议中传输层协议包头中的 Function Number 域
11	ven_msi_valid	此位置 1 表示当前 MSI 消息的所有参数均已经配置完毕, 要求 PCIE 端口发送此 MSI 消息。消息发送完毕后, 此位自动清 0。
31:12		Reserved

8.3.17 地址译码掩码寄存器 0

地址：0x18118068

RW。用于存储此 PCIE 端口作为 RC 时入站地址转换的地址掩码的低 32 位。

8.3.18 地址译码掩码寄存器 1

地址：0x1811806c

RW。用于存储此 PCIE 端口作为 RC 时入站地址转换的地址掩码的高 32 位。

8.3.19 地址译码转换地址寄存器 0

地址：0x18118070

RW。用于存储此 PCIE 端口作为 RC 时入站地址转换的转换地址的低 32 位。

8.3.20 地址译码转换地址寄存器 1

地址：0x18118074

RW。用于存储此 PCIE 端口作为 RC 时入站地址转换的转换地址的高 32 位。

8.3.21 接收消息数据负载寄存器 0

地址：0x18118078

RW。用于存储此接收到的消息所携带数据的低 32 位。

8.3.22 接收消息数据负载寄存器 1

地址：0x1811807c

RW。用于存储此接收到的消息所携带数据的高 32 位。

8.4 PCIE 配置头空间

龙芯 2H 的 4 个 PCIE 端口都有自己的 PCIE 配置头空间。每个端口的 PCIE 配置头空间都有自己的基址。端口 0 的 PCIE 配置头空间基址为 0x18114000，端口 1 的 PCIE 配置头空间基址为 0x18154000，端口 2 的 PCIE 配置头空间基址为 0x18194000，端口 3 的 PCIE 配置头空间基址为 0x181d4000。由于 0 ~ 3 号端口的 PCIE 配置头空间在命名和使用方法上相同，为了节省篇幅，我们在此仅介绍端口 0 的 PCIE 配置头空间定义。端口 1 的 PCIE 配置头空间寄存器地址 = 端口 0 的 PCIE 配置头空间寄存器地址 + 0x400000。端口 2 的 PCIE 配置头空间寄存器地址 = 端口 0 的 PCIE 配置头空间寄存器地址 + 0x800000。端口 3 的 PCIE 配置头空间寄存器地址 = 端口 0 的 PCIE 配置头空间寄存器地址 + 0xc00000。

表 8.18: PCI Standard Capability Structures

起始地址偏移	区域
0x00	PCI-Compatible Header
0x40	PCI Power Management
0x50	Message Signaled Interrupt(MSI)
0x70	PCI Express Capabilities
0xd0	VPD

图 8.2 给出了每个 PCIE 端口的 PCIE 配置空间的划分。其中的 PCI Configuration Header Space、PCI Standard Capability Structures、PCIe Extended Capability Structures 均为 PCIE 协议定义。本小节将主要介绍和端口控制相关的 Port Logic Registers。

表 8.19 中的地址偏移是指，寄存器地址相对 Port Logic 寄存器段起始地址的偏移。

表 8.19: Port Logic 寄存器地址划分

地址偏移	寄存器命名
0x0	Ack Latency Timer and Replay Timer Register
0x4	Vendor Specific DLLP Register

地址偏移	寄存器命名
0x8	Port Force Link Register
0xc	Ack frequency and L0-L1 ASPM control Register
0x10	Port Link Control Register
0x14	Lane Skew Register
0x18	Symbol Number Register
0x1c	Symbol Timer Register and Filter Mask Register 1
0x20	Filter Mask Register 2
0x24	AMBA Multiple Outbound Decomposed NP Sub-Requests Control Register
0x30	Transmit Posted FC Credit Status Register
0x34	Transmit Non-Posted FC Credit Status Register
0x38	Transmit Completion FC Credit Status Register
0x3c	Queue Status Register
0x48	VC0 Posted Receive Queue Control
0x4c	VC0 Non-Posted Receive Queue Control
0x50	VC0 Completion Receive Queue Control
0xa8	VC0 Posted Buffer Depth
0xac	VC0 Non-Posted Buffer Depth
0xb0	VC0 Completion Buffer Depth
0x10c	Gen2 Control Register
0x118	Master Response Composer Control Register 0
0x11c	Master Response Composer Control Register 1
0x120	MSI Controller Address
0x124	MSI Controller Upper Address

8.4.1 Ack Latency Timer and Replay Timer Register

地址: 0x18114700

表 8.20: Ack Latency Timer and Replay Timer Register

位域	初值	属性	描述
15:0	2071	RW	往返延迟计数器 当 Ack/Nak 延迟计数器达到此数值时将产生超时错误。当 PCIE 端口在 GEN2 状态工作时，此计数器的缺省值为原值 + 25。 Round Trip Latency Time Limit The Ack/Nak latency timer expires when it reaches this limit. If operating at 5 Gb/s, then an additional 25 is added.
31:0	6214	RW	重发延迟计数器 当 PCIE 端口收到 Nack 或者此计数器超时，PCIE 端口将进行重发操作。当 PCIE 端口在 GEN2 状态工作时，此计数器的缺省值为原值 + 76。 Replay Time Limit The replay timer expires when it reaches this limit. The PCIE port initiates a replay upon reception of a Nack or when the replay timer expires. If operating at 5 Gb/s, then an additional 76 is added.

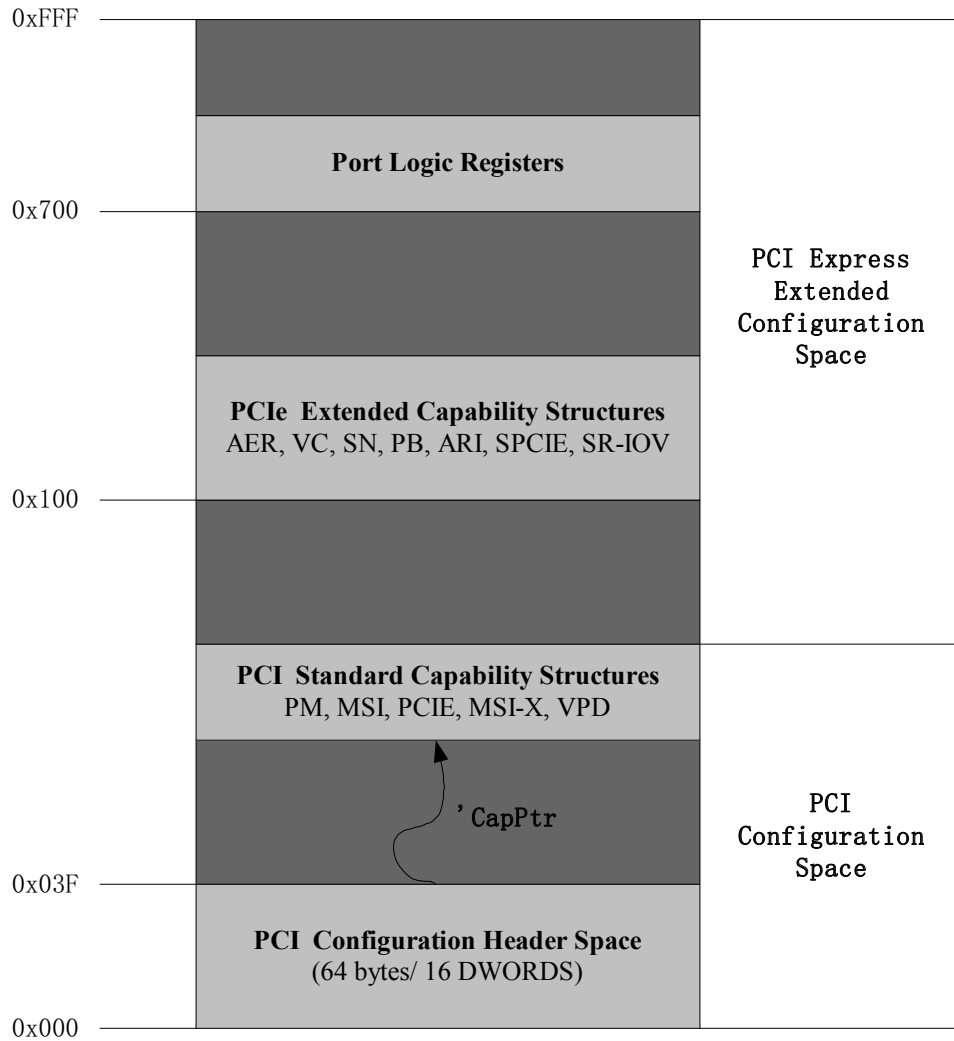


图 8.2: PCIE 配置空间划分

8.4.2 Vendor Specific DLLP Register

地址: 18114704

表 8.21: Vendor Specific DLLP Register

位域	初值	属性	描述
31:0	0xffffffff	RW	<p>特殊 DLLP (数据链路层包) 寄存器 用于在此 PCIE 端口上发送一个特殊的 DLLP。用户将 DLLP 的 8 位类型、24 位数据存入此寄存器后, 通过将 Port Link Control register 的第 0 位置 1 来发送此特殊 DLLP。</p> <p>Vendor Specific DLLP Register Used to send a specific PCI Express DLLP. The application write the 8-bit DLLP Type and 24-bits of Payload data into this register, then sets bit 0 of the Port Link Control register to send the DLLP.</p>

8.4.3 Port Force Link Register

地址: 0x18114708

此寄存器仅在调试时使用。

表 8.22: Port Force Link Register

位域	初值	属性	描述
7:0		RW	<p>链路号 仅在 RC 模式下使用 Link Number. Not used for EP.</p>
14:8	0x00		Reserved
15	0	RW	<p>链路强制控制寄存器 强制链路状态机进入 Link State 域指示的状态。此位置位将导致链路重新进行握手。此位为自动清 0 寄存器。读此位将得到 0 值。</p> <p>Force Link Forces the link to the state specified by the Link State field. The Force Link pulse will trigger Link re-negotiation. Reading from this self-clearing register field always returns a 0.</p>
21:16	0x00	RW	<p>待进入的链路状态域 Link State The Link state that the PCIE port will be forced to when bit 15 is set.</p>
23:22	0		Reserved
31:24	0x7	RW	Reserved

8.4.4 Ack frequency and L0-L1 ASPM control Register

地址: 0x1811470c

表 8.23: Ack frequency and L0-L1 ASPM control Register

位域	初值	属性	描述
7:0	0	RWS	响应频率 PCIE 端口在收集的 Ack 达到此数值时才会发送数据链路层 Ack 包 Ack Frequency The PCIE port accumulates the number of pending Ack ' s specified here before sending Ack DLLP
15:8	15	RWS	从 L0s 到 L0 状态转换时需要传送的快速训练序列数。 The number of Fast Training Sequence ordered sets to be transmitted when transitioning from L0s to L0. The maximum number of FTS ordered-sets that a component can request is 255
23:16	15	RW	从 L0s 到 L0 状态转换时需要传送的快速训练序列数。 This is the N_FTS when common clock is used. The number of Fast Training Sequence ordered sets to be transmitted when transitioning from L0s to L0.
26:24	0x3	RWS	L0s 状态进入延迟 L0s Entrance Latency Values correspond to : 000 : 1 μ s 001 : 2 μ s 010 : 3 μ s 011 : 4 μ s 100 : 5 μ s 101 : 6 μ s 110 or 111 : 7 μ s
29:27	0x3	RWS	L1 状态进入延迟 L1 Entrance Latency Values correspond to : 000 : 1 μ s 001 : 2 μ s 010 : 4 μ s 011 : 8 μ s 100 : 16 μ s 101 : 32 μ s 110 or 111 : 64 μ s
30	0	RWS	允许 ASPM 在 PCIE 链路的另一端尚未进入 L0s 时进入 L1 状态。 Enter ASPM L1 without receive in L0s Allow PCIE port to enter ASPM L1 even when link partner did not go to L0s(receive is not in L0s). When not set, PCIE port goes to ASPM L1 only after idle period during which both receive and transmit are in L0s.
31	0		Reserved

8.4.5 Port Link Control Register

地址: 0x18114710

表 8.24: Port Link Control Register

位域	初值	属性	描述
0	0	RW	用户定义 DLLP 发送请求 当软件在此位写入 1 时, 此 PCIE 端口将发送 Vendor Specific DLLP Register 中指定的 DLLP。此寄存器写入后自动清 0。 Vendor Specific DLLP Request When software writes a '1' to this bit, the core transmits the DLLP contained in the Vendor Specific DLLP Register on 4.2.2 Reading from this self-clearing register field always returns a 0.
1	0	RWS	Scramble Disable Turns off data scrambling
2	0	RWS	Loopback Enable Turns on loopback.
3	0	RWS	置热复位 仅在 RC 模式下使用 Reset Assert Triggers a recovery and forces the LTSSM to the HOT Reset state(RC mode only)
4	0	0	Reserved
5	1	RWS	数据链路初始化使能 当此位为 0 时, 此 PCIE 端口将不会传送数据链路层的流控初始化信息, 也不会建立链路。 DLL Link Enable Enables Link initialization. If DLL Link Enable = 0, the PCIE port does not transmit InitFC DLLPs and does not establish a Link.
6	0		Reserved
7	0		Reserved
15:8	0		Reserved
21:16	PCIE 端口 0 : 000111 其它 PCIE 端口: 000001	RWS	Link Mode Enable 000001 : x1 000011 : x2 000111 : x4 others : not supported
22	1	RWS	Crosslink Enable
23	0	RO	Crosslink Active. Indicates a change from upstream to downstream or downstream to upstream.
31:24			Reserved

8.4.6 Lane Skew Register

地址: 0x18114714

表 8.25: Lane Skew Register

位域	初值	属性	描述
23:0	0	RWS	Insert Lane Skew for Transmit There are three bits per Lane. The value is in units of one symbol time

位域	初值	属性	描述
24	0	RWS	禁止流控 禁止发送数据链路层的流控包 Flow Control Disable Prevents the PCIE port from sending DLLPs
25	0	RWS	禁止应答 禁止在数据链路层发送应答包 Ack/Nack Disable Prevents the PCIE port from sending ACK and Nack DLLPs
30:26	0		Reserved
31	0	RWS	Disable Lane-to-Lane Deskew Causes the PCIE port to disable the internal Lane-toLane deskew.

8.4.7 Symbol Number Register

地址: 0x18114718

表 8.26: Symbol Number Register

位域	初值	属性	描述
3:0	0xA	RWS	Number of TS Symbols Sets the number of TS identifier symbols that are sent in TS1 and TS2 ordered sets.
7:4	0		Reserved
10:8	0x3	RWS	Number of SKP Symbols
13:11	0		Reserved
18:14	1	RWS	Timer Modifier for Replay Timer Increases the timer value for the replay timer, in increments of 64 clock cycles.
23:19	0	RWS	Timer Modifier for Ack/Nack Latency Timer Increases the timer value for the Ack/Nack latency timer, increments of 64 clock cycles.
28:24	0	RWS	Timer Modifier for Flow Control Watchdog Timer Increases the timer value for the Flow Control watchdog timer, increments of 16 clock cycles.
31:29	1	RWS	Configuration Requests targeted at function numbers above this value will be returned with UR (unsupported request).

8.4.8 Symbol Timer Register and Filter Mask Register 1

地址: 0x1811471c

表 8.27: Symbol Timer Register and Filter Mask Register

1

位域	初值	属性	描述
10:0	640	RWS	SKP 间隔值 当传送的符号数为此寄存器值 $\times 2 + 1$ 时, 传送一次 SKP。 SKP interval Value The number of symbol times to wait between transmitting SKP ordered sets. Note that the PCIE port actually waits the number of symbol times in this register plus 1 between transmitting SKP ordered sets. The user must program this register accordingly. the value programmed to this register is actually clock ticks and not symbol times. For example, if 768(1536/2) we're programmed into this register, then the PCIE port will actually transmit Skp ordered sets once every 1537 symbol times.
14:11	0		Reserved
15	0	RWS	禁流控看门狗 Disable FC Watchdog Timer
16	0		允许接收设备功能号不匹配的请求。 Treat Function MisMatched TLPs as Supported。
17	0	RWS	允许接收被污染的请求。 Treat poisoned TLPs as Supported Requests
18	0	RWS	允许接收未命中地址窗口的请求。 Treat out-of-bar TLPs as Supported Requests
19	0	RWS	作为 RC 工作时, 此寄存器值为 0, 允许接收类型 1 的配置访问请求。 作为 EP 工作时, 此寄存器值为 1 则允许接收类型 1 的配置访问。 Treat CFG type 1 TLPs as Supported for EP; UR for RC.
20	0	RWS	作为 RC 工作时, 此寄存器值为 0, 接收读加锁请求。 作为 EP 工作时, 此寄存器值为 1 则允许接收读加锁请求。 Treat locked Read TLPs as Supported for EP; UR for RC
21	0	RWS	忽略数据响应包请求标识号的错误 此寄存器值为 0 时, 数据响应包的请求标识号出错将被丢弃 Mask Tag Error Rules for received CPL TLPs.
22	0	RWS	忽略数据响应包请求者标识的不匹配 此寄存器值为 0 时, 与请求的请求者标识不匹配的数据响应包将被丢弃 Mask Req.Id match for received CPL TLPs.
23	0	RWS	忽略数据响应包设备功能号的不匹配 此寄存器值为 0 时, 与请求设备功能号不匹配的数据响应包将被丢弃 Mask function match for received CPL TLPs.
24	0	RWS	忽略数据响应包 TC 的不匹配 此寄存器值为 0 时, 与请求 TC 不匹配的数据响应包将被丢弃 Mask Traffic Class match for received CPL TLPs.
25	0	RWS	忽略数据响应包属性的不匹配 此寄存器值为 0 时, 与请求属性不匹配的数据响应包将被丢弃 Mask attribute match for received CPL TLPs.
26	0	RWS	忽略数据响应包长度的不匹配 此寄存器值为 0 时, 与请求长度不匹配的数据响应包将被丢弃 Mask length match for received CPL TLPs.

位域	初值	属性	描述
27	0	RWS	允许向用户提交 ECRC 出错的数据请求包。 当此寄存器值为 0 时，ECRC 校验出错的数据请求包将被丢弃。 Allow TLPs with ECRC errors to be passed up
28	0	RWS	允许向用户提交 ECRC 出错的数据响应包。 当此寄存器值为 0 时，ECRC 校验出错的数据响应包将被丢弃。 Allow TLPs with ECRC errors to be passed up for CPL type
29	0	RWS	Reserved
30	0	RWS	在 RC 模式下允许接收 IO 访问 Allow IO transaction being received in RC mode
31	0	RWS	在 RC 模式下允许接收配置访问 Allow CFG transaction being received in RC mode

8.4.9 Filter Mask Register 2

地址: 0x18114720

表 8.28: Filter Mask Register 2

位域	初值	属性	描述
0	0	RWS	向用户提交类型 0 的消息包 Vendor MSG Type 0 not dropped
1	0	RWS	向用户提交类型 1 的消息包 Vendor MSG Type 1 not dropped
2	0	RWS	此寄存器值为 0 时，不期望收到的 CPL 包将在数据链路层被丢弃 Do not enable DLLP abort for unexpected CPL
3	0	RWS	允许 PCIE 端口处理数据刷新请求 Enable core filter to handle flush request
31:4	0	RWS	Reserved

8.4.10 AMBA Multiple Outbound Decomposed NP Sub-Requests Control Register

地址: 0x18114724

表 8.29: AMBA Multiple Outbound Decomposed NP Sub-Requests Control Register

位域	初值	属性	描述
0	1	RWS	允许此 PCIE 端口将用户发出的读请求被分解为更小的读请求。 用户通常不需要改变此寄存器的初始值。 Enable multiple outbound decomposed NP sub-requests. This bit when set to '0' disables the possibility of having multiple outstanding non-posted requests that were derived from decomposition of an outbound request. User should not clear this register unless you are requesting an amount of read data greater than Max_Read_Request_Size, and the remote device is reordering completions that have different tags Vendor MSG Type 0 not dropped

位域	初值	属性	描述
31:1	0		Reserved

8.4.11 Transmit Posted FC Credit Status Register

地址: 0x18114730

表 8.30: Transmit Posted FC Credit Status Register

位域	属性	描述
11:0	RO	Posted 写包数据流控信用 链路另一端设备的 Posted 写包数据信用。此寄存器的值在收到数据链路层的流控更新包后会被更新。 Transmit Posted Data FC Credits The Posted Data credits advertised by the receiver at the other end of the Link, updated with each UpdateFC DLLP.
19:12	RO	Posted 写包报头流控信用 链路另一端设备的 Posted 写包报头信用。此寄存器的值在收到数据链路层的流控更新包后会被更新。 Transmit Posted Header FC Credits The Posted Header credits advertised by the receiver at the other end of the Link, updated with each UpdateFC DLLP.
31:20		Reserved

8.4.12 Transmit Non-Posted FC Credit Status Register

地址: 0x18114734

表 8.31: Transmit Non-Posted FC Credit Status Register

位域	属性	描述
11:0	RO	Non-Posted 写包数据流控信用 链路另一端设备的 Non-Posted 写包数据信用。此寄存器的值在收到数据链路层的流控更新包后会被更新。 Transmit Non-Posted Data FC Credits The Non-Posted Data credits advertised by the receiver at the other end of the Link, updated with each UpdateFC DLLP.
19:12	RO	Non-Posted 写包报头流控信用 链路另一端设备的 Non-Posted 写包报头信用。此寄存器的值在收到数据链路层的流控更新包后会被更新。 Transmit Non-Posted Header FC Credits The Non-Posted Header credits advertised by the receiver at the other end of the Link, updated with each UpdateFC DLLP.
31:20		Reserved

8.4.13 Transmit Completion FC Credit Status Register

地址: 0x18114738

表 8.32: Transmit Completion FC Credit Status Register

位域	属性	描述
11:0	RO	数据响应包数据流控信用 链路另一端设备的数据响应包数据信用。此寄存器的值在收到数据链路层的流控更新包后会被更新。 Transmit Completion Data FC Credits The Completion Data credits advertised by the receiver at the other end of the Link, updated with each UpdateFC DLLP.
19:12	RO	数据响应包报头流控信用 链路另一端设备的数据响应包报头信用。此寄存器的值在收到数据链路层的流控更新包后会被更新。 Transmit Completion Header FC Credits The Completion Header credits advertised by the receiver at the other end of the Link, updated with each UpdateFC DLLP.
31:20		Reserved

8.4.14 Queue Status Register

地址: 0x181473c

表 8.33: Queue Status Register

位域	属性	描述
0	ROS	已接收传输协议层包的流控信用未被返回 标识此 PCIE 端口发送了一个传输协议层包, 但是此包所占用的流控信用未被接收者返回 Received TLP FC Credits Not Returned Indicates that the PCIE port has sent a TLP but has not yet received an UpdateFC DLLP indicating that the credits for that TLP have been restored by the receiver at the other end of the Link.
1	ROS	重试缓存非空 Transmit Retry Buffer Not Empty Indicates that there is data in the transmit retry buffer
2	ROS	接收队列非空 Received Queue Not Empty Indicates there is data in one or more of the receive buffers.
31:3		Reserved

8.4.15 VC0 Posted Receive Queue Control

地址: 0x18114748

表 8.34: VC0 Posted Receive Queue Control

位域	初值	属性	描述
11:0	55	ROS	VC0 Posted Data Credits
19:12	29	ROS	VC0 Posted Header Credits
31:20			Reserved

8.4.16 VC0 Non-Posted Receive Queue Control

地址: 0x1811474c

表 8.35: VC0 Non-Posted Receive Queue Control

位域	初值	属性	描述
11:0	12	ROS	VC0 Non-Posted Data Credits
19:12	29	ROS	VC0 Non-Posted Header Credits
31:20			Reserved

8.4.17 VC0 Completion Receive Queue Control

地址: 0x18114750

表 8.36: VC0 Completion Receive Queue Control

位域	初值	属性	描述
11:0	0	ROS	VC0 Completion Data Credits
19:12	0	ROS	VC0 Completion Header Credits
31:20			Reserved

8.4.18 VC0 Posted Buffer Depth

地址: 0x181147a8

表 8.37: VC0 Posted Buffer Depth

位域	初值	属性	描述
13:0	134	ROS	VC0 Posted Data Queue Depth
15:14			Reserved
25:16	58	ROS	VC0 Posted Header Queue Depth
31:26			Reserved

8.4.19 VC0 Non-Posted Buffer Depth

地址: 0x181147ac

表 8.38: VC0 Non-Posted Buffer Depth

位域	初值	属性	描述
13:0	0	ROS	VC0 Non-Posted Data Queue Depth
15:14			Reserved
25:16	0	ROS	VC0 Non-Posted Header Queue Depth
31:26			Reserved

8.4.20 VC0 Completion Buffer Depth

地址: 0x181147b0

表 8.39: VC0 Completion Buffer Depth

位域	初值	属性	描述
13:0	0	ROS	VC0 Completion Data Queue Depth
15:14			Reserved
25:16	0	ROS	VC0 Completion Header Queue Depth
31:26			Reserved

8.4.21 Gen2 Control Register

地址: 0x1811480c

表 8.40: Gen2 Control Register

位域	初值	属性	描述
7:0	15	RWS	快速训练序列数目 在切换到 Gen2 时传送的快速训练序列的数目。 Sets the Number of Fast Training Sequences(N_FTS) that the PCIE port advertises as its N_FTS during Gen2 Link training. This value is used to inform the Link partner about the PHY's ability to recover synchronization after a low power state.
16:8	0	RWS	确定有效的传输线数目 Predetermined Number of Lanes Used to limit the effective link width to ignore "broken" lanes that detect a receiver. Indicates the number of lanes to check for exit from Electrical Idle in POLLING.ACTIVE and L2.IDLE. It is possible that the LTSSM may detect a Receiver on a "bad" or "broken" lane during the Detect Sub-state. However, it is also possible that such a lane may also fail to exit Electrical Idle and therefore prevent a valid link from being configured. This value is referred as the 'Predetermined Number of Lanes' in section 4.2.6.2.1 of the PCI Express Base Specification Revision 2.1. Encoding is as follows: 0x01 = 1 lane, 0x02 = 2 lanes, 0x03 = 3 lanes and so on.
17	0	RWS	主动发起到 Gen2 转换 Directed Speed Change Indicates to the LTSSM whether or not to initiate a speed change to Gen2 after the link is initialized at Gen1 speed.
18	0	RWS	配置 PHY 发送端电平波动水平 Config PHY Tx Swing Indicates the voltage level the PHY should drive. When set to 1, indicates Full Swing. When set to 0, indicates Low Swing.
19	0	RWS	Config Tx Compliance Receive Bit When set to 1, signals LTSSM to transmit TS ordered sets with the compliance receive bit assert
20	0	RWS	Used to set the de-emphasis level for upstream ports
31:21			Reserved

8.4.22 Master Response Composer Control Register 0

地址: 0x18114818

表 8.41: Master Response Composer Control Register 0

位域	初值	属性	描述
2:0	0	RWS	请求者最大读请求数据长度 (字节) Remote Read Request Size Specifies the largest amount of data (bytes) that will ever be requested (via an inbound MemRd TLP) by a remote device. Encoding is as follows: 000: 128 001: 256 010: 512 011: 1024 100: 2048 101: 4096 default: 128
7:3			Reserved
15:8	15	ROS	Remote Max Bridge Tag Specifies the maximum number (-1) of Non-Posted request inbound.
31:16			Reserved

8.4.23 Master Response Composer Control Register 1

地址: 0x1811481c

表 8.42: Master Response Composer Control Register 1

位域	初值	属性	描述
0	0	RWS	向此寄存器中写入 1 将使对 Master Response Composer Control Register 0 的修改生效。读此寄存器仅能获得 0。 Writing '1' to this (self-clearing register) causes any changes in the Master Response Composer Control Register 0 to take place in the hardware.
31:1			Reserved

8.4.24 MSI Controller Address

地址: 0x18114820

表 8.43: MSI Controller Address

位域	初值	属性	描述
31:0	0	RW	接收 MSI 的地址 MSI Controller Address System specified address for MSI memory write transaction termination.

8.4.25 MSI Controller Upper Address

地址: 0x18114824

表 8.44: MSI Controller Upper Address

位域	初值	属性	描述
31:0	0	RW	接收 MSI 的地址的高 32 位 MSI Controller Upper Address System specified Upper address for MSI memory write transaction termination.

8.5 常用例程

本节给出龙芯 2H 的 PCIE 控制器的常用例程。

当龙芯 2H 以 PCIE 总线上的 EP 方式工作时龙芯 2H 需要执行下面例程中的 `pcie_link_init` ; 而以方式 RC 工作时, 龙芯 2H 需要先执行下面示例中的 `pcie_link_init` , 再执行下面示例中的 `pcie_header_init` 。随后, 龙芯 2H 可以通过 `cfg_device_read` 和 `cfg_device_write` 这两个函数对端口 0 上的设备的 PCI Header 进行初始化。

```

unsigned int tmp_var;
unsigned char * pcie_base = 0xffffffff00000000;
unsigned int * header_base = 0xb8114000;
unsigned int * ctrl_base = 0xb8118000;
unsigned int * cfg_base = 0xb8116000;

void pcie_link_init(unsigned int port_id)
{
    unsigned int port_base = pcie_base + port_id*0x400000;
    // set port logic register of port 0
    // initiate speed change to PCIE Gen2 and set Tx to Low Swing
    tmp_var = *(volatile unsigned int *)(port_base + header_base + 0x80c);
    *(volatile unsigned int *)(port_base + header_base + 0x80c) = (tmp_var | 0
        x20000)&0xffffbfff;

    //start link training
    *(volatile unsigned int *)(port_base + ctrl_base) = 0xff000c;

    //wait link train end
    tmp_var = *(volatile unsigned int *)( port_base + ctrl_base +0xc);
    while((tmp_var&0x1f)!=0x11)
    {
        tmp_var = *(volatile unsigned int *)( port_base + ctrl_base +0xc);
    }
    printf("now_PCIE_port_0_link_is_start_up\n");
}

```

```

void pcie_hot_reset(unsigned int port_id)
{
    unsigned int port_base = pcie_base + port_id*0x400000;
    tmp_var = *(volatile unsigned int *) ( port_base + ctrl_base);
    //enable soft reset
    *(volatile unsigned int *) ( port_base + ctrl_base) = tmp_var |0x1000;
    //trigger hot reset
    *(volatile unsigned int *) ( port_base + ctrl_base +0x4) = 0x4;
}

void pcie_header_init(unsigned int port_id)
{ //only used in RC mode
    unsigned int port_base = pcie_base + port_id*0x400000 + header_base;
    //set master enable, io enable, mem enable, perr enable, serr enable
    *(volatile unsigned int *) ( port_base + 0x4) = 0x147;

    //clear master abort, serr and perr status
    //set IO space to be 16-bit address
    //set 64 KB IO space: 0x0000 ~ 0xffff
    *(volatile unsigned int *) ( port_base + 0x1c) = 0xf100f000;

    //set IO limit and IO base up 16 bit address
    *(volatile unsigned int *) ( port_base + 0x30) = 0x0;

    //set memory limit and memory base
    //the memory window: 0x0 ~ 0x17ffffff
    //but only two memory window bellow is valid
    //1MB lowest memory space: 0x0 ~0xfffff
    //128 MB memory space : 0x10000000 ~ 0x17ffffff
    *(volatile unsigned int *) ( port_base + 0x20) = 0x17f00000;

    //set prefetchable memory limit and base
    //2 GB prefetchable memory space : 0x40000000 ~ 0x7fffffff
    *(volatile unsigned int *) ( port_base + 0x24) = 0x7ff04000;

    //set prefetchable base up 32 bit
    *(volatile unsigned int *) ( port_base + 0x28) = 0x0;
    //set prefetchable limit up 32 bit
    *(volatile unsigned int *) ( port_base + 0x2c) = 0x0;

    //enable serr
    *(volatile unsigned int *) ( port_base + 0x3c) = 0x20000;

    //enable system error on correctalbe error, non-fatal error and fatal error
    //enable PME interrupt
    *(volatile unsigned int *) ( port_base + 0x8c) = 0xf;
}

```

```

//enable correctalbe error, non-fatal error and fatal error
*(volatile unsigned int *)( port_base + 0x12c) = 0x7;

//enable ASPM L0s and L1

*(volatile unsigned char *)( port_base + 0x80) = 0x3;
}
void cfg_device_read(
    unsigned int type, unsigned int bus_num,
    unsigned int dev_num, unsigned int func_num,
    unsigned int reg_id, unsigned int * read_data,
    unsigned int port_id
)
{
    unsigned int port_base = pcie_base + port_id*0x400000;
    *(volatile unsigned char *) ( port_base + ctrl_base + 0x24) = (type&0x1) | (
        bus_num << 24) | (dev_num <<19) | (func_num <<16);
    *(read_data) = *(volatile unsigned int *) ( port_base + cfg_base + (reg_id<<2)
    );
}
void cfg_device_write(
    unsigned int type, unsigned int bus_num,
    unsigned int dev_num, unsigned int func_num,
    unsigned int reg_id, unsigned int write_data,
    unsigned int port_id
)
{
    unsigned int port_base = pcie_base + port_id*0x400000;
    *(volatile unsigned char *) ( port_base + ctrl_base + 0x24) = (type&0x1) | (
        bus_num << 24) | (dev_num <<19) | (func_num <<16);
    *(volatile unsigned int *) ( port_base + cfg_base + (reg_id<<2)) = write_data;
}

```

在上面的 pcie.header.init 例程中，我们将端口对外的 IO、Memory、Prefetchable Memory 窗口皆设为我们所支持的最大窗口。当龙芯 2H 仅在端口 0 接有设备时，Memory 和 Prefetchable Memory 窗口所设的最大值可由端口 0 所独享。当龙芯 2H 在端口 0 以外的端口也接有设备时，Memory 和 Prefetchable Memory 窗口所设的最大值在每个端口上除了最低 1MB 的空间外其它的窗口仅为最大值的四分之一。

第九章 显示控制器

9.1 概述

显示控制器从内存中取帧缓冲和光标信息输出到外部显示接口上。

龙芯 2H 的显示控制器支持的特性包括：

- 双路显示，一路 DVO 接口，一路 VGA 接口
- 每路显示最大支持至 2048x2048@60Hz
- Monochrome 、 ARGB8888 两种模式硬件光标
- RGB444 , RGB555 , RGB565 , RGB888 四种色深
- 输出抖动和伽马校正
- 可切换的双路线性帧缓冲
- 中断和软复位
- DAC 控制和内存调度支持
- DVO 接口线序可配置

9.2 寄存器定义

9.2.1 帧缓冲配置寄存器

表 9.1: 帧缓冲配置寄存器

名称	位域	初值	描述
帧缓冲配置寄存器 地址：0xbfe51240/0xbfe51250 (DVO/VGA)			
Reset	20	0	从值 1 变为 0 时软复位
Gamma	12	0	写 1 使能伽玛校正
FB_num	11	0	指示当前正在使用的缓冲区号，只读
Switch_panel	9	0	写 1 表示使用另一路显示输出，两路配合可实现两路显示交换、拷贝
Output_enable	8	0	写 1 使能显示输出
FB_switch	7	0	写 1 使能缓冲区切换，下一帧完成切换
Format	2:0	0	色深格式： 0:none 1:RGB444 2:RGB555 3:RGB565 4:RGB888

9.2.2 帧缓冲地址寄存器 0

表 9.2: 帧缓冲地址寄存器 0

名称	位域	初值	描述
帧缓冲地址寄存器 0 地址: 0xbfe51260/0xbfe51270 (DVO/VGA)			
Address	31:0	0	缓冲区 0 在内存中的物理地址

9.2.3 帧缓冲地址寄存器 1

表 9.3: 帧缓冲地址寄存器 1

名称	位域	初值	描述
帧缓冲地址寄存器 1 地址: 0xbfe51580/0xbfe51590 (DVO/VGA)			
Address	31:0	0	缓冲区 1 在内存中的物理地址

9.2.4 帧缓冲跨度寄存器

表 9.4: 帧缓冲跨度寄存器

名称	位域	初值	描述
帧缓冲跨度寄存器 地址: 0xbfe51280/0xbfe51290 (DVO/VGA)			
Stride	31:0	0	显示屏一行的字节数, 按 256 字节向上取整

9.2.5 帧缓冲初始字节寄存器

表 9.5: 帧缓冲初始字节寄存器

名称	位域	初值	描述
帧缓冲初始字节寄存器 地址: 0xbfe51300/0xbfe51310 (DVO/VGA)			
Origin	31:0	0	显示屏左侧原有字节数, 一般配 0 即可

9.2.6 颜色抖动配置寄存器

表 9.6: 颜色抖动配置寄存器

名称	位域	初值	描述
颜色抖动配置寄存器 地址: 0xbfe51360/0xbfe51370 (DVO/VGA)			
Enable	31	0	写 1 使能颜色抖动功能
RedSize	19:16	0	红色域宽度
GreenSize	11:8	0	绿色域宽度
BlueSize	3:0	0	蓝色域宽度

9.2.7 颜色抖动查找表低位寄存器

表 9.7: 颜色抖动查找表低位寄存器

名称	位域	初值	描述
颜色抖动查找表低位寄存器 地址: 0xbfe51380/0xbfe51390 (DVO/VGA)			
Y1_X3	31:28	0	坐标 (3,1) 处的比较值

名称	位域	初值	描述
Y1_X2	27:24	0	坐标 (2,1) 处的比较值
Y1_X1	23:20	0	坐标 (1,1) 处的比较值
Y1_X0	19:16	0	坐标 (0,1) 处的比较值
Y0_X3	15:12	0	坐标 (3,0) 处的比较值
Y0_X2	11:8	0	坐标 (2,0) 处的比较值
Y0_X1	7:4	0	坐标 (1,0) 处的比较值
Y0_X0	3:0	0	坐标 (0,0) 处的比较值

9.2.8 颜色抖动查找表高位寄存器

表 9.8: 颜色抖动查找表高位寄存器

名称	位域	初值	描述
颜色抖动查找表高位寄存器 地址: 0xbfe513a0/0xbfe513b0 (DVO/VGA)			
Y3_X3	31:28	0	坐标 (3,3) 处的比较值
Y3_X2	27:24	0	坐标 (2,3) 处的比较值
Y3_X1	23:20	0	坐标 (1,3) 处的比较值
Y3_X0	19:16	0	坐标 (0,3) 处的比较值
Y2_X3	15:12	0	坐标 (3,2) 处的比较值
Y2_X2	11:8	0	坐标 (2,2) 处的比较值
Y2_X1	7:4	0	坐标 (1,2) 处的比较值
Y2_X0	3:0	0	坐标 (0,2) 处的比较值

9.2.9 液晶面板配置寄存器

表 9.9: 液晶面板配置寄存器

名称	位域	初值	描述
液晶面板配置寄存器 地址: 0xbfe513c0/0xbfe513d0 (DVO/VGA)			
ClockPol	9	0	时钟极性, 写 1 取反
ClockEn	8	1	时钟使能, 写 1 使能
DEPol	1	0	数据使能极性, 写 1 取反
DE	0	1	数据使能, 写 1 使能

9.2.10 水平显示宽度寄存器

表 9.10: 水平显示宽度寄存器

名称	位域	初值	描述
水平显示宽度寄存器 地址: 0xbfe51400/0xbfe51410 (DVO/VGA)			
Total	26:16	0	显示屏一行的总像素数 (包括非显示区)
Display	10:0	0	显示屏一行中显示区的像素数

9.2.11 行同步配置寄存器

表 9.11: 行同步配置寄存器

名称	位域	初值	描述
行同步配置寄存器 地址: 0xbfe51420/0xbfe51430 (DVO/VGA)			
Pol	31	0	行同步极性, 写 1 取反
Pulse	30	1	行同步使能, 写 1 使能
End	26:16	0	行同步结束时的像素数
Start	10:0	0	行同步结束时的像素数

9.2.12 垂直显示高度寄存器

表 9.12: 垂直显示高度寄存器

名称	位域	初值	描述
垂直显示高度寄存器 地址: 0xbfe51480/0xbfe51490 (DVO/VGA)			
Total	26:16	0	显示屏一列的总像素数 (包括非显示区)
Display	10:0	0	显示屏一列中显示区的像素数

9.2.13 场同步配置寄存器

表 9.13: 场同步配置寄存器

名称	位域	初值	描述
场同步配置寄存器 地址: 0xbfe514a0/0xbfe514b0 (DVO/VGA)			
Pol	31	0	场同步极性, 写 1 取反
Pulse	30	1	场同步使能, 写 1 使能
End	26:16	0	场同步结束时的像素数
Start	10:0	0	场同步结束时的像素数

9.2.14 伽玛校正目录寄存器

表 9.14: 伽玛校正目录寄存器

名称	位域	初值	描述
伽玛校正目录寄存器 地址: 0xbfe514e0/0xbfe514f0 (DVO/VGA)			
Index	7:0	0	表示从 0-255 颜色值之间的哪一项开始进行 Gamma 调整, 一般设 0。只需配一次, 此后该值硬件会自增

9.2.15 伽玛校正寄存器

表 9.15: 伽玛校正寄存器

名称	位域	初值	描述
伽玛校正寄存器 地址: 0xbfe51500/0xbfe51510 (DVO/VGA)			
Red	23:16	0	Gamma 调整的红色域, 将 Index 指示的值调整为当前域的值
Green	15:8	0	Gamma 调整的绿色域, 将 Index 指示的值调整为当前域的值

名称	位域	初值	描述
Blue	7:0	0	Gamma 调整的蓝色域, 将 Index 指示的值调整为当前域的值

9.2.16 光标配置寄存器

表 9.16: 光标配置寄存器

名称	位域	初值	描述
光标配置寄存器 地址: 0xbfe51520			
HotSpotX	20:16	0	光标的作用点的横坐标 (在光标 32*32 的图案中的横坐标)
HotSpotY	12:8	0	光标的作用点的纵坐标 (在光标 32*32 的图案中的纵坐标)
Display	4	0	指示光标存在于哪个显示单元中, 0 表示在 DVO 中, 1 表示在 VGA 中
Format	1:0	0	光标类型: 0:disabled 1:monochrome 2:ARGB8888

9.2.17 光标存储地址寄存器

表 9.17: 光标存储地址寄存器

名称	位域	初值	描述
光标存储地址寄存器 地址: 0xbfe51530			
Address	31:0	0	光标数据在内存中的物理地址

9.2.18 光标显示位置寄存器

表 9.18: 光标显示位置寄存器

名称	位域	初值	描述
光标显示位置寄存器 地址: 0xbfe51540			
Y	26:16	0	光标作用点在整个屏幕的纵坐标
X	10:0	0	光标作用点在整个屏幕的横坐标

9.2.19 光标背景色寄存器

表 9.19: 光标背景色寄存器

名称	位域	初值	描述
光标背景色寄存器 地址: 0xbfe51550			
Red	23:16	0	monochrome 模式下光标背景色的红色域
Green	15:8	0	monochrome 模式下光标背景色的绿色域
Blue	7:0	0	monochrome 模式下光标背景色的蓝色域

9.2.20 光标前景色寄存器

表 9.20: 光标前景色寄存器

名称	位域	初值	描述
光标前景色寄存器 地址: 0xbfe51550			
Red	23:16	0	monochrome 模式下光标前景色的红色域
Green	15:8	0	monochrome 模式下光标前景色的绿色域
Blue	7:0	0	monochrome 模式下光标前景色的蓝色域

9.2.21 中断寄存器

表 9.21: 中断寄存器

名称	位域	初值	描述
中断寄存器 地址: 0xbfe51570			
En	26:16	0	中断时能信号, 对应于 10:0 位中断, 写 1 使能, 写 0 屏蔽
DB0_FUF	10	0	DVO 内部数据缓冲区致命下溢, 只读
DB1_FUF	9	0	VGA 内部数据缓冲区致命下溢, 只读
DB0_UF	8	0	DVO 内部数据缓冲区下溢, 只读
DB1_UF	7	0	VGA 内部数据缓冲区下溢, 只读
FB0_End	6	0	DVO 帧缓冲读取结束, 只读
FB1_End	5	0	VGA 帧缓冲读取结束, 只读
Cursor_End	4	0	光标数据读取结束, 只读
Display0_Hsync	3	0	DVO 产生了 Hsync, 只读
Display0_Vsync	2	0	DVO 产生了 Vsync, 只读
Display1_Hsync	1	0	VGA 产生了 Hsync, 只读
Display1_Vsync	0	0	VGA 产生了 Vsync, 只读

9.2.22 DAC 控制寄存器

表 9.22: DAC 控制寄存器

名称	位域	初值	描述
DAC 控制寄存器 地址: 0xbfe51600			
fscsel	15	0	控制 DAC 的 fscsel 端口
cmdr	14	0	控制 DAC 的 cmdr 端口
cmds	13	0	控制 DAC 的 cmds 端口
ftb	12:10	0	控制 DAC 的 ftb 端口
ftg	9:7	0	控制 DAC 的 ftg 端口
ftr	6:4	0	控制 DAC 的 ftr 端口
hzb	3	0	控制 DAC 的 hzb 端口
hzg	2	0	控制 DAC 的 hzg 端口
h zr	1	0	控制 DAC 的 h zr 端口

9.2.23 DAC Sense 值寄存器

表 9.23: DAC Sense 值寄存器

名称	位域	初值	描述
DAC Sense 值寄存器 地址: 0xbfe51610			
DAC_Sense	2:0	0	DAC 的 sense 端口, 当 RGB 端口有翻转时, sense 会相应改变。配合 Sense 配置寄存器可以检测接口上是否有设备

9.2.24 Sense 配置寄存器

表 9.24: Sense 配置寄存器

名称	位域	初值	描述
Sense 配置寄存器 地址: 0xbfe51620			
SenseEn	25	0	写 1 使能, 使能时 VGA 通路的 R、G、B、DE 将由此寄存器操纵
DE	24	0	受控制的 DE
Red	23:16	0	受控制的红色域
Green	15:8	0	受控制的绿色域
Blue	7:0	0	受控制的蓝色域

9.2.25 DVO 输出模式寄存器

表 9.25: DVO 输出模式寄存器

名称	位域	初值	描述
DVO 输出模式寄存器 地址: 0xbfe51630			
Clk_Pol	15	0	DVO 时钟相位, 写 1 反向
-	14:10	0	保留, 必须写 0
p_BGR	9	0	RGB 三个总线输出 BGR 颜色信息
p_BRG	8	0	RGB 三个总线输出 BRG 颜色信息
p_GBR	7	0	RGB 三个总线输出 GBR 颜色信息
p_GRB	6	0	RGB 三个总线输出 GRB 颜色信息
p_RBG	5	0	RGB 三个总线输出 RBG 颜色信息
p_RGB	4	1	RGB 三个总线输出 RGB 颜色信息
B_Inv	3	0	DVO 接口蓝色域按位倒置
G_Inv	2	0	DVO 接口绿色域按位倒置
R_Inv	1	0	DVO 接口红色域按位倒置
-	0	0	保留, 必须写 0

9.3 相关功能说明

9.3.1 颜色抖动

颜色抖动功能用于根据一定规则来增强像素值, 下面的这个例子将解释颜色抖动功

能的实现步骤。

首先，确定是对哪一数据位进行增强（即该数据位加 1），为此需要配置颜色抖动配置寄存器。比如配置 RedSize 为 6（1 到 8 之间，包含 1 和 8），则表明对从 MSB 位数起第 6 位进行增强，即 RedColor[7:0] 的 RedColor[2] 位增强，GreenSize 和 BlueSize 同理。

其次，需要建立查找表，即配置颜色抖动查找表寄存器。查找表包含 16 个条目，即 16 个阈值，每个条目位宽为 4。查找表通过屏幕像素计数器的横坐标 x 的最低两位 x[1:0] 和纵坐标 y 的最低两位 y[1:0] 进行索引，得到一个阈值 U[3:0]。

对应屏幕 (x, y) 位置的像素值的最低四位被拿来跟查到的这个阈值比较。如果 RedColor[3:0] > U[3:0]，并且 RedColor[7:2] 不为 6'b111111，则 RedColor[2] 位加 1，实现颜色增强。

9.3.2 输出定序

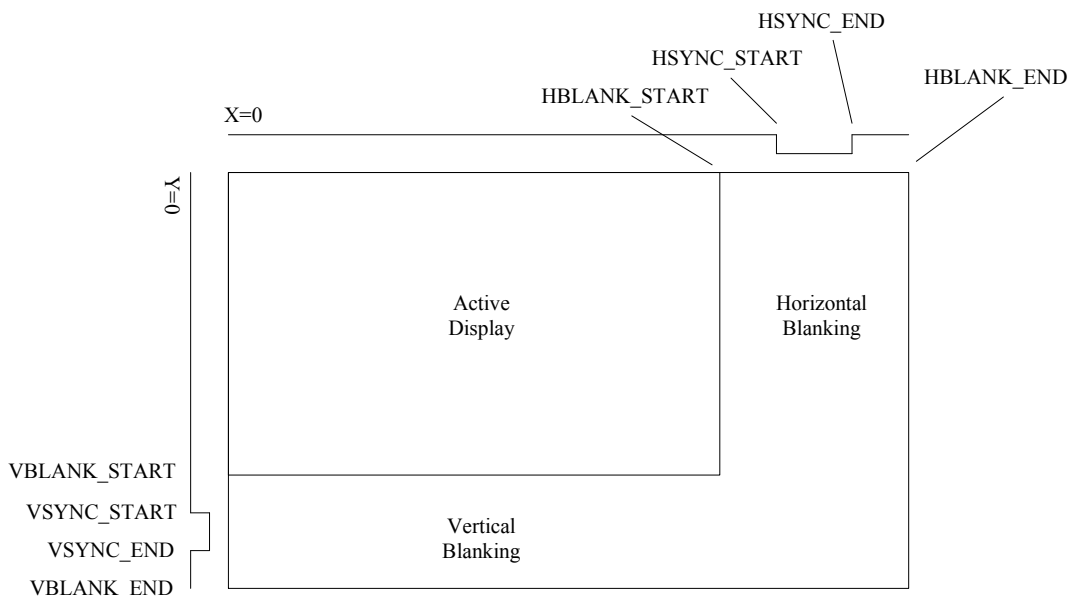


图 9.1: 显示控制器输出定序图

9.3.3 伽玛校正

伽玛校正模块包含三个查找表，分别负责红绿蓝三色，查找表可以通过寄存器改写。

考虑一个伽玛校正配对如：(原色，校正色)。当设置伽玛颜色查找表时，先将 Index 设为 0（即原色从 0 开始配置），然后依次配置三种原色对应的校正色。

当使用伽玛校正功能时，控制器将自动访问查找表并完成原色到校正色的转换。

9.3.4 硬件光标

硬件光标包含两种格式：Monochrome 和 ARGB8888。

在 Monochrome 格式下，光标每个像素点占用两位数据，分别为 mask 和 xor，两个位共同决定了光标的形状和颜色，如表9.26所示。

表 9.26: 单色光标颜色配置

mask	XOR	颜色
0	0	背景色
0	1	前景色
1	0	透明
1	1	屏幕反色

在 ARGB 模式下，红绿蓝三色各有 8 位色深，A 表示透明度，即光标与底色的插值系数。

指针有两个重要的点：左上点（top-left point）和作用点（hot spot）。左上点用来作为指针地址的参考点，作用点用来将鼠标按下动作确定到一个像素上。

第十章 GMAC 控制器

10.1 DMA 寄存器描述

龙芯 2H 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 0x1fe1.0000；GMAC0 的 DMA 寄存器的起始地址是 0x1fe1.1000。

GMAC1 的 GMAC 寄存器的起始地址是 0x1fe1.8000；GMAC0 的 DMA 寄存器的起始地址是 0x1fe1.9000。

下面分别介绍 DMA 寄存器和 GMAC 寄存器的意义。

10.1.1 Register0 (Bus Mode Register)

表 10.1: Register0 (Bus Mode Register)

名称	位域	初值	描述
Register0 (Bus Mode Register) Offset: 0x00			
Reserved 保留	31:27	0x0	保留，只读
MB: Mixed Burst 混合突发访问	26	0x0	当此位为高，FB 位为低时，AXI master 在突发访问长度大于 16 时采用 INCR 访问模式，当突发访问长度为 16 或者小于 16 时采用 FIX 访问模式。用户不用关心此位设置。
AAL: Address-Aligned Beats 地址对齐节拍	25	0x0	当此位和 FB 位同时为高时，AXI 接口的所有访问将对齐到起始地址的 LS 位。如果 FB 位为 0，首次访问地址访问不对齐，剩余的访问地址对齐。用户不用关心此位设置。
8XPBL Mode 是否使能 PBLX8 模式	24	0x0	此位为高时，GMAC DMA 的最大突发数据传输长度为 8,16,32,64,128 或者 256。最大突发长度取决于 PBL。用户不用关心此位设置。
USP: Use Separate PBL 使用分离的 PBL 值	23	0x0	此位为高时，PBL 值只应用于 TxDMA。此位为低时，PBL 值应用于 TxDMA 和 RxDMA。用户不用关心此位设置。
RPBL: RxDMA PBL RxDMA 突发传输长度	22:17	0x01	表示一次 RxDMA 传输的最大突发传输长度。只能为 1,2,4,8,16 和 32，其它值无效。
FB: Fixed Burst 定长突发传输长度使能	16	0x0	指定 AXI Master 接口是否采用 FIX 突发传输模式。用户不用关心此位设置。

名称	位域	初值	描述
PR: Rx:Tx priority ratio RxDMA 与 TxDMA 优先级比例	15:14	0x0	在 DA 位为 0 时起作用。 00 : 1 : 1 01 : 2 : 1 10 : 3 : 1 11 : 4 : 1
PBL:Programmable Burst Length 可编程突发传输长度	13:8	0x1	用户不用关心此设置。
ATDS:Alternate Descriptor size 是否使用 32 字节大小描述符	7	0x0	此位为 1 时使用 32 字节大小的描述符 此位为 0 时使用 16 字节大小的描述符
DSL: Descriptor Skip Length 描述符间隔距离	6:2	0x00	设置 2 个描述符间的距离。但此值为 0 时，默认为 DMA 描述符大小。
DA: DMA Arbitration scheme DMA 传输仲裁策略	1	0x0	0: 在 RxDMA 和 TxDMA 间采用轮转仲裁机制 1: RxDMA 优先级高于 TxDMA 优先级。具体比值见 PR 值。
SWR:Software Reset 软件复位	0	0x1	此位置高 DMA 控制器将复位 GMAC 内部寄存器和逻辑。当复位结束时该位自动清零。

10.1.2 Register1 (Transmit Poll Demand Register)

表 10.2: Register1 (Transmit Poll Demand Register)

名称	位域	初值	描述
Register1 (Transmit Poll Demand Register) Offset: 0x04			
TPD: Transmit Poll Demand 传输轮询使能	31:0	0x0	向此值写入任意值，发送 DMA 控制器将会读取寄存器 18 对应的描述符。如果该描述符无效，DMA 传输将会停止。如果该描述符有效，DMA 传输将会继续。

10.1.3 Register2 (Receive Poll Demand Register)

表 10.3: Register2 (Receive Poll Demand Register)

名称	位域	初值	描述
Register2 (Receive Poll Demand Register) Offset: 0x08			
RPD: Receive Poll Demand 接收轮询使能	31:0	0x0	向此值写入任意值，接收 DMA 控制器将会读取寄存器 18 对应的描述符。如果该描述符无效，DMA 传输将会停止。如果该描述符有效，DMA 传输将会继续。

10.1.4 Register3 (Receive Descriptor List Address Register)

表 10.4: Register3 (Receive Descriptor List Address Register)

名称	位域	初值	描述
Register3 (Receive Descriptor List Address Register) Offset: 0x0C			
Start of Receive List 接收描述符起始地址	31:0	0x0	指向接收描述符首地址。

10.1.5 Register4 (Transmit Descriptor List Address Register)

表 10.5: Register4 (Transmit Descriptor List Address Register)

名称	位域	初值	描述
Register4 (Transmit Descriptor List Address Register) Offset: 0x10			
Start of Transmit List 发送描述符起始地址	31:0	0x0	指向发送描述符首地址

10.1.6 Register5 (Status Register)

表 10.6: Register5 (Status Register)

名称	位域	初值	描述
Register5 (Status Register) Offset: 0x14			
Reserved	31:30		保留, 只读
TTI: Time-Stamp Trigger Interrupt 时间戳触发中断	29	0x0	时间戳模块触发中断。只读。
GPI:GMAC PMT Interrupt 电源管理模块触发中断	28	0x0	电源管理模块触发中断。只读。
GMI:GMAC MMC Interrupt MMC 模块触发中断	27	0x0	MMC 模块触发中断。只读。
GLI:GMAC Line interface Interrupt GMAC 模块线路触发中断	26	0x0	GMAC 模块的 PCS 或者 RGMII 模块触发中断。只读。
EB: Error Bits 错误位	25:23	0x0	23 : 1'b1 TxDMA 数据传输过程中发生错误 1'b0 RxDMA 数据传输过程中发生错误 24 : 1'b1 读传输错误 1'b0 写传输错误 25 : 1'b1 描述符访问错误 1'b0 数据缓存访问错误
TS:Transmit Process State 传输过程状态	22:20	0x0	3'b000: 传输停止; 复位或者停止命令发送 3'b001: 正在进行; 获取传输描述符 3'b010: 正在进行; 等待传输状态 3'b011: 正在进行; 从发送缓存读取数据并发送到传输 FIFO(TxFIFO) 3'b100: 写入时间戳状态 3'b101: 保留 3'b110: 挂起; 传输描述符不可用或者传输缓存下溢。 3'b111: 运行; 关闭传输描述符。

名称	位域	初值	描述
RS:Receive Process State 接收过程状态	19:17	0x0	3'b000: 停止; 复位或者接收到停止命令 3'b001: 运行; 获取接收描述符。 3'b010: 保留; 3'b011: 运行; 等待接收包。 3'b100: 暂停; 接收描述符不可用。 3'b101: 运行; 关闭接收描述符。 3'b110: 时间戳写状态。 3'b111: 运行; 将包内容从接收缓存传输到系统内存。
NIS:Normal Interrupt Summary 正常中断汇总	16	0x0	提示系统是否存在正常中断。
AIS:Abnormal Interrupt Summary 异常中断汇总	15	0x0	提示系统是否存在异常中断。
ERI:Early Receive Interrupt 提前接收中断	14	0x0	提示 DMA 控制器已经把包的第一个数据写入接收缓存
FBI:Fatal Bus Error Interrupt 总线错误中断	13	0x0	提示总线错误, 具体信息见[25:23]。当此位设置后 DMA 引擎停止总线访问操作。
Reserved	12:11	0x0	保留
ETI:Early Transmit Interrupt 提前发送中断	10	0x0	提示需要传输的以太网帧已经完全传输到 MTL 模块中的传输 FIFO
RWT:Receive Watchdog Timeout 接收看门狗超时	9	0x0	提示接收到一个大小超过 2048 字节的以太网帧。(当巨帧使能时, 提示接收到大小超过 10240 字节的以太网帧)
RPS:Receive Process Stopped 接收过程停止	8	0x0	指示接收过程停止
RU:Receive Buffer Unavailable 接收缓存不可用	7	0x0	指示接收缓存不可用
RI:Receive Interrupt 接收中断	6	0x0	指示帧接收完成。帧接收的状态信息已经写入接收描述符。接收处于运行状态。
UNF:Transmit Underflow 传输缓存下溢	5	0x0	指示帧发送过程中产生接收缓存下溢。
OVF:Receive Overflow 接收缓存上溢	4	0x0	指示帧接收过程中接收缓存上溢。
TJT:Transmit Jabber Timeout	3	0x0	
TU:Transmit Buffer Unavailable 传输缓存不可用	2	0x0	提示传输列表中的下一个描述符不能被 DMA 控制器访问。
TPS:Transmit Process Stopped 传输过程停止	1	0x0	提示传输过程停止
TI:Transmit Interrupt 传输完成中断	0	0x0	提示帧传输完成并且第一个描述符的 31 位置位。

10.1.7 Register6 (Operation Mode Register)

表 10.7: Register6 (Operation Mode Register)

名称	位域	初值	描述
Register6 (Operation Mode Register)	Offset: 0x18		
Reserved 保留	31:27	0x0	保留

名称	位域	初值	描述
DT 关闭丢弃 TCP/IP Checksum 错误以太网帧的功能	26	0x0	此位为 1 时 GMAC 将不丢弃 checksum 错误的以太网帧。
RSF:Receive Store and Forward 接收存储转发	25	0x0	此位为 1 时 MTL 模块只接收已经全部存储在接收 FIFO 中的以太网帧。
DFD:Disable Flushing of Received Frames 关闭冲刷接收的以太网帧的功能	24	0x0	此位为 1 时，接收 DMA 在接收描述符或者接收缓存不可用时不冲刷任何以太网帧。
RFA[2]:MSB of Threshold for Activating Flow Control 激活流控阈值	23	0x0	100：最大值减去 5KB 101：最大值减去 6KB 110：最大值减去 7KB 111：保留 (注：最大值为 8KB)
RFD[2]:MSB of Threshold for Deactivating Flow Control 关闭流控阈值	22	0x0	100：最大值减去 5KB 101：最大值减去 6KB 110：最大值减去 7KB 111：保留 (注：最大值为 8KB)
TSF:Transmit Store and Forward 发送存储转发	21	0x0	此位为 1 时，帧的发送只在帧的内容已经全部进入 MTL 的传输 FIFO 中。
FTF:Flush Transmit FIFO 冲刷传输 FIFO	20	0x0	此位为 1 时，传输控制逻辑复位为默认值，并且会导致发送 FIFO 里面的数据全部丢失。
Reserved	19:17	0x0	保留
TTC:Transmit Threshold Control 传输阈值控制	16:14	0x0	当帧大小超过此值时 MTL 将会传输该帧。 000: 64 字节 001: 128 字节 010: 192 字节 011: 256 字节 100: 40 字节 101: 32 字节 110: 24 字节 111: 16 字节
ST:Start/Stop Transmission Command 开始 / 停止传输命令	13	0x0	此位为 1，传输进入运行状态。 此位为 0，传输进入停止状态。
RFD:Threshold for deactivating flow control 关闭流控阈值	12:11	0x0	00: 最大值减去 1KB 01: 最大值减去 2KB 10: 最大值减去 3KB 11: 最大值减去 4KB (最大值为 8KB)
RFA:Threshold for Activating flow control 激活流控阈值	10:9	0x0	00: 最大值减去 1KB 01: 最大值减去 2KB 10: 最大值减去 3KB 11: 最大值减去 4KB (最大值为 8KB)
EFC:Enable HW flow control 使能硬件流控	8	0x0	此位为 1 时，基于接收 FIFO 利用率的硬件流控电路生效。

名称	位域	初值	描述
FEF:Forward Error Frames 传输错误帧	7	0x0	此位为 1 时, 接收错误帧 (错误帧包括: CRC 错误, 冲突错误, 巨帧, 看门狗超时, 溢出等)
FUF:Forward Undersized Good Frames 接收无错误的小帧	6	0x0	此位为 1 时, 接收 FIFO 将会接收没有错误但小于 64 字节的以太网帧。
Reserved	5	0x0	保留
RTC:Receive Threshold Control 接收阈值控制	4:3	0x0	MTL 传输接收 FIFO 中帧内容已经超过此项设置大小。 00: 64 字节 01:32 字节 10: 96 字节 11: 128 字节
OSF:Operate on Second Frame 是否操作第二个以太网帧	2	0x0	此位为高时, DMA 在第一个以太网帧的状态尚未写回时即可以开始处理第二个以太网帧。
SR:Start/Stop Receive 开始 / 停止接收	1	0x0	此位设置为高时, 接收进入运行状态。 此位设置为低时, 接收进入停止状态。
Reserved 保留	0	0x0	保留

10.1.8 Register7 (Interrupt Enable Register)

表 10.8: Register7 (Interrupt Enable Register)

名称	位域	初值	描述
Register7 (Interrupt Enable Register) Offset: 0x1C			
Reserved 保留	31:17	0x0	保留
NIE:Normal Interrupt Summary Enable 正常中断汇总使能	16	0x0	此位为 1 时: 正常中断使能 此位为 0 时: 正常中断不使能
AIE:Abnormal Interrupt Summary Enable 非正常中断汇总使能	15	0x0	此位为 1 时: 非正常中断使能。 此位为 0 时: 非正常中断不使能。
ERE : Early Receive Interrupt Enable 早期接收中断使能	14	0x0	此位为高时: 早期接收中断使能
FBE:Fatal Bus Error Enable 总线致命错误中断使能	13	0x0	此位为高时: 总线致命错误中断使能。
Reserved 保留	12:11	0x0	保留
ETE:Early Transmit Interrupt Enable 早期传输中断使能	10	0x0	此位为高时: 使能早期传输中断
RWE:Receive Watchdog Timeout Enable 接收看门狗超时中断使能	9	0x0	此位为高时: 使能接收看门狗超时中断
RSE:Receive Stopped Enable 接收停止中断使能	8	0x0	此位为高时: 使能接收停止中断。

名称	位域	初值	描述
RUE:Receive Buffer Unavailable Enable 接收缓冲区不可用中断使能	7	0x0	此位为高时：使能接收缓冲区不可用中断。
RIE:Receive Interrupt Enable 接收中断使能	6	0x0	此位为高时：使能接收完成中断
UNE:Underflow Interrupt Enable 传输 FIFO 下溢中断使能	5	0x0	此位为高时：使能传输 FIFO 下溢中断
OVE:Overflow Interrupt Enable 接收 FIFO 上溢中断使能	4	0x0	此位为高时：使能接收 FIFO 上溢中断。
TJE:Transmit Jabber Timeout Enable 传输 Jabber 超时中断使能	3	0x0	此位为高时：使能 Jabber 超时中断。
TUE:Transmit Buffer Unavailable Enable 传输缓存不可用中断使能	2	0x0	此位为高时：使能传输缓存不可用中断。
TSE:Transmit Stopped Enable 传输停止中断使能	1	0x0	此位为高时：使能传输停止中断。
TIE:Transmit Interrupt Enable 传输完成中断使能	0	0x0	此位为高时：使能传输完成中断。

10.1.9 Register8 (Missed Frame and Buffer Overflow Counter Register)

表 10.9: Register8 (Missed Frame and Buffer Overflow Counter Register)

名称	位域	初值	描述
Register8 (Missed Frame and Buffer Overflow Counter Register) Offset: 0x20			
Reserved 保留	31:29	0x0	保留
Overflow bit for FIFO Overflow Counter FIFO 溢出指示位	28	0x0	FIFO 溢出指示位
Indicates the number of frames missed by the application 应用程序丢失的帧个数	27:17	0x0	指示应用程序丢失帧的个数
Overflow bit for Missed Frame Counter 丢失帧个数溢出指示	16	0x0	提示丢失帧个数已经超过计数的最大值。
Indicates the number of frames missed by the controller due to the Host Receive Buffer being unavailable 因为主机接收缓存不可用导致帧丢失的个数	15:0	0x0	指示因为主机接收缓存不可用导致帧丢失个数的计数。

10.1.10 Register9(Receive Interrupt Watchdog Timer Register)

表 10.10: Register9(Receive Interrupt Watchdog Timer Register)

名称	位域	初值	描述
Register9(Receive Interrupt Watchdog Timer Register) Offset: 0x24			
Reserved 保留	31:8	0x0	
RIWT: RI Watchdog Timer count 接收看门狗时间计数	7:0	0x0	当看门狗设置后,表示以时钟周期 x256 的时间为单 单位计时。当 DMA 接收到数据包且 status 寄存器中 RI 位为 0 时开始计时,当看门狗计数超时后 RI 位 置 1。当 RI 位为 1 后该域复位

10.1.11 Register10 (AXI Bus Mode Register)

表 10.11: Register10 (AXI Bus Mode Register)

名称	位域	初值	描述
Register10 (AXI Bus Mode Register) Offset: 0x28			
EN_LPI: Enable LPI (Low Power Interface) 启用低功耗接口	31	0x0	该位为 1 时,启用低功耗接口 (Low Power Interface) ,系统接受来自 AXI 系统时钟控制器的 LPI 请求。 该位为 0 时,不启用低功耗接口,系统永远拒绝 LPI 请求
UNLCK_ON_MGK_RWK: Unlock on Magic Packet or Remote Wake Up 解锁魔法包或远程唤醒包	30	0x0	当该位为 1 时, GMAC 处于低功耗 (Low Power) 状态下时只能通过魔法包或者远程唤醒包来重新回 到工作状态;当该位为 0 时, GMAC 处于低功耗 (Low Power) 状态下时只能通过任意包重新回到工 作状态
Reserved 保留	29:23	0x0	
WR_OSR_LMT: AXI Maximum Write Out Standing Request Limit AXI 最大 outstanding 写请求	22:20	0x0	该位设置了 AXI 接口写操作发出最大的 outstand- ing 请求数
Reserved 保留	19	0x0	
WR_OSR_LMT: AXI Maximum Read Out Standing Request Limit AXI 最大 outstanding 读请求	18:16	0x0	该位设置了 AXI 接口读操作发出最大的 outstand- ing 请求数
Reserved 保留	15:13	0x0	
AXI_AAL: Address-Aligned Beats AXI 地址对齐	12	0x0	这位是一个只读位,与 AAL 位 0 相同
Reserved 保留	11:8	0x0	
BLEN256: AXI Burst Length 256 AXI Burst 长度 256	7	0x0	该位只有当 AXI_BURST_LENGTH 配置 256 时有 效,为 1 时表示 AXI 支持 Burst 长度为 256 的请求; 否则保留
BLEN256: AXI Burst Length 128 AXI Burst 长度 128	6	0x0	该位只有当 AXI_BURST_LENGTH 配置 128 时有 效,为 1 时表示 AXI 支持 Burst 长度为 128 的请求; 否则保留
BLEN256: AXI Burst Length 64 AXI Burst 长度 64	5	0x0	该位只有当 AXI_BURST_LENGTH 配置 64 时有 效,为 1 时表示 AXI 支持 Burst 长度为 64 的请求; 否则保留

名称	位域	初值	描述
BLEN256: AXI Burst Length 32 AXI Burst 长度 32	4	0x0	该位只有当 AXI BURST_LENGTH 配置 32 时有效, 为 1 时表示 AXI 支持 Burst 长度为 32 的请求; 否则保留
BLEN256: AXI Burst Length 16 AXI Burst 长度 16	3	0x0	该位只有当 AXI BURST_LENGTH 配置 16 时有效, 为 1 时表示 AXI 支持 Burst 长度为 16 的请求; 否则保留
BLEN256: AXI Burst Length 8 AXI Burst 长度 8	2	0x0	该位只有当 AXI BURST_LENGTH 配置 8 时有效, 为 1 时表示 AXI 支持 Burst 长度为 8 的请求; 否则保留
BLEN256: AXI Burst Length 4 AXI Burst 长度 4	1	0x0	该位只有当 AXI BURST_LENGTH 配置 4 时有效, 为 1 时表示 AXI 支持 Burst 长度为 4 的请求; 否则保留
UNDEF: AXI Undefined Burst Length 为定义 AXI Burst 长度	0	0x0	该位为 1 时表示 AXI 接口可以发出任意 Burst 长度的请求, 为 0 时表示 AXI 支持以上 bit7-1 定义的长度为 256/128/64/32/16/8/4 或者 1 的请求

10.1.12 Register11 (AXI Status Register)

表 10.12: Register11 (AXI Status Register)

名称	位域	初值	描述
Register11 (AXI Status Register) Offset: 0x2C			
Reserved 保留	31:2	0x0	
	1	0x0	该位为 1 时表示当前 AXI 正在发送读请求
	0	0x0	该位为 1 时表示当前 AXI 正在发送写请求

10.1.13 Register18 (Current Host Transmit Descriptor Register)

表 10.13: Register18 (Current Host Transmit Descriptor Register)

名称	位域	初值	描述
Register18 (Current Host Transmit Descriptor Register) Offset: 0x48			
Host Transmit Descriptor Address Pointer 当前发送描述符主机地址指针	31:0	0x0	只读

10.1.14 Register19 (Current Host Receive Descriptor Register)

表 10.14: Register19 (Current Host Receive Descriptor Register)

名称	位域	初值	描述
Register19 (Current Host Receive Descriptor Register) Offset: 0x4C			
Host Receive Descriptor Address Pointer 当前接收描述符主机地址指针	31:0	0x0	只读

10.1.15 Register20 (Current Host Transmit Buffer Address Register)

表 10.15: Register20 (Current Host Transmit Buffer Address Register)

名称	位域	初值	描述
Register20 (Current Host Transmit Buffer Address Register) Offset: 0x50			
Host Transmit Buffer Address Pointer 当前传输缓冲区主机地址指针	31:0	0x0	只读

10.1.16 Register21 (Current Host Receive Buffer Address Register)

表 10.16: Register21 (Current Host Receive Buffer Address Register)

名称	位域	初值	描述
Register21 (Current Host Receive Buffer Address Register) Offset: 0x54			
Host Receive Buffer Address Pointer 当前接收缓冲区主机地址指针	31:0	0x0	只读

10.2 GMAC 控制器寄存器描述

10.2.1 Register0 (MAC Configuration Register)

表 10.17: Register0 (MAC Configuration Register)

名称	位域	初值	描述
Register0 (MAC Configuration Register) Offset: 0x0000			
Reserved 保留	31:26	0x0	保留
TC: Transmit Configuration in RGMII 使能 RGMII 链路信息传输	24	0x0	此位为高时, 将会把双工模式, 链路速度, 链路以及链路连接 / 断开等信息通过 RGMII 接口传输给 PHY。
WD: Watchdog Disable 关闭看门狗	23	0x0	此位为高时, GMAC 将关闭接收端的看门狗定时器, 可以接收最大 16384 字节的以太网帧。
JD: Jabber Disable 关闭 Jabber 定时器	22	0x0	此位为高时, GMAC 关闭发送过程中的 Jabber 定时器, 可以发送最大 16384 字节的以太网帧。
BE: Frame Burst Enable 帧突发传输使能	21	0x0	此位为高时, GMAC 使能传输过程中的帧突发传输模式。
JE: Jumbo Frame Enable 巨帧使能	20	0x0	此位为高时, GMAC 使能巨帧 (最大 9018 字节) 的接收。
IFG: Inter-Frame Gap 最小帧间距	19:17	0x0	设置传输过程中的最小帧间距。 000: 96 位时间 001: 88 位时间 010: 80 位时间 ... 111: 40 位时间

名称	位域	初值	描述
DCRS: Disable Carrier Sense During Transmission 传输过程中关闭载波冲突检测	16	0x0	此位为高时，MAC 忽略半双工模式下 CRS 信号的检测。
PS: Port Select 端口选择	15	0x0	0: GMII (1000Mbps) 1: MII (10/100Mbps)
FES: Speed 快速以太网速度提示	14	0x0	0: 10Mbps 1: 100Mbps
DO: Disable Receive Own 关闭接收自己发出的以太网帧	13	0x0	此位为高时，GMAC 不接收半双工模式下 gmii_txen_o 有效的以太网帧。
LM: Loopback Mode 使能环回模式	12	0x0	此位为高时，GMII/MII 工作在环回模式下。
DM: Duplex Mode 使能全双工模式	11	0x0	此位为高时，GMAC 工作在全双工模式下，在全双工模式下可以同时发送和接收以太网帧。
IPC: Checksum Offload 校验和卸载使能	10	0x0	此位为高时，GMAC 硬件计算接收到以太网帧的负载 (payload)。还检查 IPV4 头的校验和是否正确。
DR: Disable Retry 关闭重传	9	0x0	此位为高时，GMAC 在遇到冲突时不重传发送冲突的以太网帧，而只报告冲突错误。
LUD: Link Up/Down 链路连接 / 链路断开	8	0x0	0: 链路断开 1: 链路连接
ACS: Automatic Pad/CRC Stripping 以太网帧 Pad/CRC 自动去除	7	0x0	此位为 1 时，GMAC 中去除接收到的以太网帧的 Pad 和 FCS。
BL: Back-Off Limit 回退限制	6:5	0x0	回退限制决定基于 slot 的延迟时间。 00: k=min(n,10) 01: k=min(n,8) 10: k=min(n,4) 11: k=min(n,1)
DC: Deferral Check Deferral 检查	4	0x0	此位为 1 时，使能 deferral 检测功能。
TE: Transmitter Enable 传输使能	3	0x0	此位为 1 时，使能 GMAC 传输功能。
RE: Receiver Enable 接收使能	2	0x0	此位为 1 时，使能 GMAC 接收功能。
Reserved	1:0	0x0	保留。

10.2.2 Register1 (MAC Frame Filter)

表 10.18: Register1 (MAC Frame Filter)

名称	位域	初值	描述
Register1 (MAC Frame Filter) Offset: 0x0004			
RA: Receive All 接收全部	31	0x0	此位为 1 时，GMAC 接收模块把接收到的所有帧都发给应用程序，忽略源地址 / 目标地址过滤机制。
Reserved 保留	30:11	0x0	保留

名称	位域	初值	描述
HPF: Hash or Perfect Filter 哈希或者完全过滤	10	0x0	此位为 1 时，在哈希 / 完全过滤机制中匹配的以太网帧发送给应用。 此位为 0 时，只有在哈希过滤机制中匹配的以太网帧才发送给应用。
SAF: Source Address Filter Enable 源地址过滤使能	9	0x0	GMAC CORE 比较接收到的以太网帧的源地址域和在 SA 寄存器中的值，如果匹配，接收状态寄存器中的 SAMatch 位设置为高。如果此位为 1，源地址匹配失败，GMAC CORE 将丢弃该以太网帧。 如果此位为 0，不管源地址匹配结果 GMAC CORE 都接收此帧，而匹配结果写入接收状态寄存器。
SAIF: SA Inverse Filtering 源地址反转过滤	8	0x0	此位为 1 时，和 SA 寄存器中源地址匹配的以太网帧将会标记为源地址匹配失败。 此位为 0 时，和 SA 寄存器中源地址不匹配的以太网帧将会标记为源地址匹配失败。
PCF: Pass Control Frames 接收控制帧	7:6	0x0	00: GMAC 过滤所有控制帧 01: GMAC 接收除了 pause 帧以外的所有控制帧。 10: GMAC 接收所有控制帧。 11: GMAC 根据地址过滤情况接收控制帧
DBF: Disable Broadcast Frames 关闭广播帧	5	0x0	此位为 1 时，过滤所有接收的广播帧。 此位为 0 时，接收所有广播帧。
PM: Pass All Multicast 接收所有多播帧	4	0x0	此位为 1 时，接收所有多播帧。 此位为 0 时，过滤所有多播帧。
DAIF: DA Inverse Filtering 目标地址反转过滤	3	0x0	此位为 1 时，对单播和多播帧进行反向目标地址匹配。 此位为 0 时，对单播和多播帧进行正常目标地址匹配。
HMC: Hash Multicast 哈希多播过滤	2	0x0	此位为 1 时，对接收到的多播帧根据哈希表的内容进行目标地址过滤。
HUC: Hash Unicast 哈希单播过滤	1	0x0	此位为 1 时，对接收到的单播帧根据哈希表的内容进行目标地址过滤。
PR: Promiscuous Mode 混杂模式	0	0x0	接收所有以太网帧。

10.2.3 Register2 (Hash Table High Register)

表 10.19: Register2 (Hash Table High Register)

名称	位域	初值	描述
Register2 (Hash Table High Register) Offset: 0x0008			
HTH: Hash Table High 哈希表高位	31:0	0x0	哈希表的高 32 位。

10.2.4 Register3 (Hash Table Low Register)

表 10.20: Register3 (Hash Table Low Register)

名称	位域	初值	描述
Register3 (Hash Table Low Register) Offset: 0x000C			

名称	位域	初值	描述
HTL: Hash Table Low 哈希表低位	31:0	0x0	哈希表的低 32 位。

10.2.5 Register4 (GMII Address Register)

表 10.21: Register4 (GMII Address Register)

名称	位域	初值	描述
Register4 (GMII Address Register) Offset: 0x0010			
Reserved 保留	31:16	0x0	保留
PA: Physical Layer Address PHY 地址	15:11	0x0	此域选择需要访问 32 个 PHY 中的哪个。
GR: GMII Register 需要访问的 PHY 设备中的寄存器	10:6	0x0	此域选择需要访问的的 PHY 的哪个 GMII 配置寄存器。
Reserved 保留	5	0x0	保留
CR: CSR Clock Range CSR 时钟范围	4:2	0x0	此域决定 MDC 时钟是 clk_csr_i 时钟频率比例。 0000 clk_csr_i/42 0001 clk_csr_i/62 0010 clk_csr_i/16 0011 clk_csr_i/26 0100 clk_csr_i/102 0101 clk_csr_i/124 0110, 0111 Reserved
GW: GMII Write GMII 写	1	0x0	此位为 1 时, 通过 GMII 数据寄存器对 PHY 进行写操作 此位为 0 时, 通过 GMII 数据寄存器对 PHY 进行读操作。
GB: GMII Busy GMII 忙	0	0x0	对寄存器 4 和寄存器 5 写之前, 此位应为 0。在写寄存器 4 之前此位必须先置 0。在访问 PHY 的寄存器时, 应用程序需要将此位设置为 1, 表示 GMII 接口上有写或者读操作正在进行。

10.2.6 Register5 (GMII Data Register)

表 10.22: Register5 (GMII Data Register)

名称	位域	初值	描述
Register5 (GMII Data Register) Offset: 0x0014			
Reserved 保留	31:16	0x0	保留
GD: GMII Data GMII 数据	15:0	0x0	此域保存了对 PHY 进行管理读访问操作的 16 位数据, 或者对 PHY 进行管理写访问的 16 位数据。

10.2.7 Register6 (Flow Control Register)

表 10.23: Register6 (Flow Control Register)

名称	位域	初值	描述
Register6 (Flow Control Register) Offset: 0x0018			
PT: Pause Time 暂停时间	31:16	0x0	此域保存了需要填入传输控制帧中的暂停时间域。
Reserved 保留	15:8	0x0	保留
DZPQ: Disable Zero-Quanta Pause 禁止零时间片暂停帧	7	0x0	此位为 1 时, 禁止自动零时间片的暂停控制帧的产生。
Reserved 保留	6	0x0	保留
PLT: Pause Low Threshold 暂停帧的低阈值	5:4	0x0	此域用于设置暂停时间的阈值。 00: 暂停时间减少 4 个时间槽 01: 暂停时间减少 28 个时间槽 10: 暂停时间减少 144 个时间槽 11: 暂停时间减少 256 个时间槽 (一个时间槽为在 GMII/MII 接口上传输 512 比特或者 64 字节的时间)
UP: Unicast Pause Frame Detect 单播的暂停帧探测	3	0x0	此位为 1 时, GMAC 将会根据 MAC 地址 0 指定的本站单播地址来探测暂停帧。
RFE: Receive Flow Control Enable 接收流控使能	2	0x0	此位为 1 时, GMAC 将会解析接收到的暂停帧, 并且按照暂停帧指定的时间暂停帧的发送。
TEF: Transmit Flow Control Enable 发送流控使能	1	0x0	在全双工模式下, 此位为 1 时, GMAC 使能暂停帧的发送。 在半双工模式下, 此位为 1 时, GMAC 使能反压操作。
FCB/BPA: Flow Control Busy/Backpressure Activate 流控忙 / 反压激活	0	0x0	此位为 1 时, 在全双工模式下发起暂停控制帧的发送或在全双工模式下启动反压操作。

10.2.8 Register7 (VLAN Tag Register)

表 10.24: Register7 (VLAN Tag Register)

名称	位域	初值	描述
Register7 (VLAN Tag Register) Offset: 0x001C			
Reserved 保留	31:17	0x0	保留
ETV: Enable 12-Bit VLAN Tag Comparison 使能 12 位 VLAN Tag 比较	16	0x0	此位为 1 时, 使用 12 位 VLAN Tag 而不是使用 16 位 VLAN Tag 用于以太网帧比较和过滤。
VL: VLAN Tag Identifier for Receive Frames 帧接收的 VLAN Tag 标识	15:0	0x0	此域保存 802.1Q 格式的 VLAN Tag, 用于比较接收到的以太网帧的位于第 15 和第 16 个字节的 VLAN Tag。

10.2.9 Register8 (Version Register)

表 10.25: Register8 (Version Register)

名称	位域	初值	描述
Register8 (Version Register) Offset: 0x0020			
Reserved 保留	15:8	0x0	保留
Version 版本号	7:0	0x0	0X35

10.2.10 Register14 (Interrupt Status Register)

表 10.26: Register14 (Interrupt Status Register)

名称	位域	初值	描述
Register14 (Interrupt Status Register) Offset: 0x0038			
Reserved 保留	15:8	0x0	保留
MMC Receive Checksum Offload Interrupt Status MMC 接收校验和卸载状态中断	7	0x0	MMC 校验和卸载寄存器产生任何中断产生时，此位设置为 1。
MMC Transmit Interrupt Status MMC 传输中断	6	0x0	MMC 传输中断寄存器产生任何中断时，此位设置为 1。
MMC Receive Interrupt Status MMC 接收中断状态	5	0x0	MMC 接收中断寄存器产生任何中断时，此位设置为 1。
MMC Interrupt Status MMC 中断状态	4	0x0	7:5 的任何位为高时，此位设置为 1。
PMT Interrupt Status 电源管理中断状态	3	0x0	在 Power Down 状态下，收到 magic 帧或在 Wake-on-LAN 帧时，此位设置为 1。
PCS Auto-Negotiation Complete PCS 自动协商完成	2	0x0	RGMII PHY 接口自动协商完成时，此位设置为 1。
PCS Link Status Changed PCS 链路状态变化	1	0x0	RGMII PHY 接口的链路状态发生任何变化时，此位设置为 1。
RGMII Interrupt Status RGMII 中断状态	0	0x0	RGMII 接口的链路状态发生任何变化时，此位设置为 1。

10.2.11 Register15 (Interrupt Mask Register)

表 10.27: Register15 (Interrupt Mask Register)

名称	位域	初值	描述
Register15 (Interrupt Mask Register) Offset: 0x003C			
Reserved 保留	15:10	0x0	保留
Time Stamp Interrupt Mask 时间戳中断使能	9	0x0	此位为 1 时，禁止时间戳发生的中断
Reserved 保留	8:4	0x0	保留
PMT Interrupt Mask 电源管理中断使能	3	0x0	此位为 1 时，禁止电源管理引起的中断。

名称	位域	初值	描述
PCS AN Completion Interrupt Mask PCS 自动协商完成中断使能	2	0x0	此位为 1 时，禁止 PCS 自动协商完成中断。
PCS Link Status Interrupt Mask PCS 链路状态中断使能	1	0x0	此位为 1 时，禁止由于 PCS 链路状态变化引起的中断。
RGMII Interrupt Mask RGMII 中断使能	0	0x0	此位为 1 时，禁止 RGMII 引起的中断。

10.2.12 Register16 (MAC Address0 High Register)

表 10.28: Register16 (MAC Address0 High Register)

名称	位域	初值	描述
Register16 (MAC Address0 High Register) Offset: 0x0040			
MO: Always 1 保留	31	0x0	保留
Reserved 保留	30:16	0x0	保留
MAC Address0[47:32] MAC 地址高 16 位	15:0	0x0	存放用于接收地址过滤和传输流控帧的 MAC 地址。

10.2.13 Register17 (MAC Address0 Low Register)

表 10.29: Register17 (MAC Address0 Low Register)

名称	位域	初值	描述
Register17 (MAC Address0 Low Register) Offset: 0x0044			
MAC Address0[31:0] MAC 地址低 32 位	31:0	0x0	存放用于接收地址过滤和传输流控帧的 MAC 地址。

10.2.14 Register18 (MAC Address1 High Register)

表 10.30: Register18 (MAC Address1 High Register)

名称	位域	初值	描述
Register18 (MAC Address1 High Register) Offset: 0x0048			
AE: Address Enable 地址使能	31	0x0	此位为 1 时，地址过滤模块使用第 2 个 MAC 地址用于完全地址过滤。此位为 0 时，地址过滤模块不使用第 2 个 MAC 地址用于地址过滤。
SA: Source Address 源 MAC 地址	30	0x0	此位为 1 时，MAC 地址 1 用于比较接收帧的源 MAC 地址。 此位为 0 时，MAC 地址 1 用于比较接收帧的目标 MAC 地址。
MBC: Mask Byte Control 掩模字节控制	29:24	0x0	此域用于比较每个 MAC 地址的字节掩模控制位。比如第 29 位用于掩码寄存器 18 的[15:8]这个字节。
Reserved 保留	23:16	0x0	保留。

名称	位域	初值	描述
MAC Address1[47:32] 第 2 个 MAC 地址的高 16 位	15:0	0xFFFF	

10.2.15 Register19 (MAC Address1 Low Register)

表 10.31: Register19 (MAC Address1 Low Register)

名称	位域	初值	描述
Register19 (MAC Address1 Low Register) Offset: 0x004C			
MAC Address1[31:0] 第 2 个 MAC 地址的低 32 位	31:0	0x0	

10.2.16 Register48 (AN Control Register)

表 10.32: Register48 (AN Control Register)

名称	位域	初值	描述
Register48 (AN Control Register) Offset: 0x00C0			
Reserved 保留	31:19	0x0	保留
SGMII RAL Control 保留	18	0x0	保留
LR: Lock to Reference 锁定到参考时钟	17	0x0	此位为 1 时, PHY 将其锁相环锁定到 125MHz 的参考时钟。
ECD: Enable Comma Detect 使能停顿探测	16	0x0	此位为 1 时, 使能 PHY 的停顿探测和字重同步。
Reserved 保留	15	0x0	保留
ELE: External Loopback Enable 外部环回使能	14	0x0	此位为 1 时, 使能 PHY 进入环回模式。
Reserved 保留	13	0x0	保留
ANE: Auto-Negotiation Enable 自动协商使能	12	0x0	此位为 1 时, GMAC 将会和链路对方进行自动协商。
Reserved 保留	11:10	0x0	保留
RAN: Restart Auto-Negotiation 重新进行自动协商	9	0x0	此位为 1 时, 重新进行自动协商。
Reserved 保留	8:0	0x0	保留

10.2.17 Register49 (AN Status Register)

表 10.33: Register49 (AN Status Register)

名称	位域	初值	描述
Register49 (AN Status Register) Offset: 0x00C4			
Reserved 保留	31:9	0x0	保留

名称	位域	初值	描述
ES: Extended Status 扩展状态	8	0x0	只读, 因为 GMAC 支持扩展状态信息。
Reserved 保留	7:6	0x0	保留
ANC: Auto-Negotiation Complete 自动协商完成	5	0x0	只读, 指示自动协商完成。
Reserved 保留	4	0x0	保留
ANA: Auto-Negotiation Ability 自动协商能力	3	0x0	只读, 因为 GMAC 支持自动协商。
LS: Link Status 链路状态	2	0x0	此位为 1 时, 指示链路连接上。 此位为 0 时, 指示链接未连接。
Reserved 保留	1:0	0x0	保留。

10.2.18 Register50 (Auto-Negotiation Advertisement Register)

表 10.34: Register50 (Auto-Negotiation Advertisement Register)

名称	位域	初值	描述
Register50 (Auto-Negotiation Advertisement Register) Offset: 0x00C8			
Reserved 保留	31:16	0x0	保留
NP: Next Page Support 下一页面支持	15	0x0	只读为 0, 因为 GMAC 不支持下一页面。
Reserved 保留	14	0x0	保留
RFE: Remote Fault Encoding 远端错误编码	13:12	0x0	此 2 位指示链路对端发生错误, 具体编码将 IEEE 802.3z 第 37.2.1.5 小节。
Reserved 保留	11:9	0x0	保留
PSE: Pause Encoding Pause 位编码	8:7	0x0	见 IEEE 802.3z 第 37.2.1.4 小节
HD: Half-Duplex 半双工	6	0x0	此位为 1 时, 指示 GMAC 支持半双工。
FD: Full-Duplex 全双工	5	0x0	此位为 1 时, 指示 GMAC 支持全双工。
Reserved 保留	4:0	0x0	保留

10.2.19 Register51 (Auto-Negotiation Link Partner Ability Register)

表 10.35: Register51 (Auto-Negotiation Link Partner Ability Register)

名称	位域	初值	描述
Register51 (Auto-Negotiation Link Partner Ability Register) Offset: 0x00CC			
Reserved 保留	31:16	0x0	保留

名称	位域	初值	描述
NP: Next Page Support 下一页面支持	15	0x0	此位为 1 时, 指示有更多下一页面信息可用 此位为 0 时, 指示下一页面交换不可用。
ACK: Acknowledge 确认	14	0x0	指示在自动协商中, 链路对端成功接收到 GMAC 的基本页面。
RFE: Remote Fault Encoding 远端错误编码	13:12	0x0	见 IEEE 802.3z 第 37.2.1.5 小节。
Reserved 保留	11:9	0x0	保留
PSE: Pause Encoding 对端 pause 状态编码	8:7	0x0	见 IEEE 802.3z 第 37.2.14 小节。
HD: Half-Duplex 半双工	6	0x0	指示对端可以运行在半双工模式。
FD: Full-Duplex 全双工	5	0x0	指示对端可以运行在全双工模式。
Reserved 保留	4:0	0x0	保留

10.2.20 Register52 (Auto-Negotiation Expansion Register)

表 10.36: Register52 (Auto-Negotiation Expansion Register)

名称	位域	初值	描述
Register52 (Auto-Negotiation Expansion Register) Offset: 0x00D0			
Reserved 保留	31:3	0x0	保留
NPA: Next Page Ability 下一页面能力	2	0x0	只读为 0, 因为 GMAC 不支持下一页面。
NPR: New Page Received 接收到新页面	1	0x0	此位为 1 时, 指示 GMAC 接收到新页面。
Reserved 保留	0	0x0	保留

10.2.21 Register54 (SGMII/RGMII Status Register)

表 10.37: Register54 (SGMII/RGMII Status Register)

名称	位域	初值	描述
Register54 (SGMII/RGMII Status Register) Offset: 0x00D8			
Reserved 保留	31:4	0x0	保留
Link Status 链路状态	3	0x0	此位为 1 时, 指示链路连接上。 此位为 0 时, 指示链路未连接上。
Link Speed 链路速度	2:1	0x0	指示链路当前速度 00: 2.5MHz 01: 25MHz 10: 125MHz
Link Mode 链路模式	0	0x0	0: 半双工 1: 全双工

IEEE1588 寄存器:

10.2.22 Register448 (Time Stamp Control Register)

表 10.38: Register448 (Time Stamp Control Register)

名称	位域	初值	描述
Register448 (Time Stamp Control Register) Offset: 0x0700			
Reserved 保留	31:20	0x0	保留
ATSFC: Auxiliary Snapshot FIFO Clear Auxiliary 快照 FIFO 清除	19	0x0	该位为 1 时重置 Auxiliary Snapshot FIFO 的指针, FIFO 清空, 完成后为 0
TSENMADDR: Enable MAC address for PTP frame filtering 启用 PTP 帧的 MAC 地址过滤	18	0x0	此位为 1 时, 接收到的 PTP 包会经过 GMAC 的 MAC 目的地址过滤器 (DA Fiter), 不匹配的 PTP 包会被丢弃
TSCLKTYPE: Select the type of clock node 选择时间节点类型	17:16	0x0	该域按照以下类型编码: 00: 普通时钟 (Ordinary clock) 01: 边界时钟 (Boundary clock) 10: 端到端的透明时钟 (End-to-End Transparent clock) 11: 点到点透明时钟 (Peer-to-peer Transparent clock)
TSMSTRENA: Enable Snapshot for Messages Relevant to Master 启用主机相关信息快照	15	0x0	该位为 1 时快照 (Snapshot) 记录主机相关信息 (Relevant to Master); 为 0 时快照记录从机相关信息 (Relevant to Slave)
TSEVENTENA: Enable Time Stamp Snapshot for Event Messages 启用事件信息时间戳快照	14	0x0	该位为 1 时快照 (Snapshot) 只记录事件相关信息 (Event Message); 为 0 时快照记录所有信息
TSIPV4ENA: Enable Time Stamp Snapshot for IPv4 frames 启用 IPv4 时间戳快照	13	0x0	该位为 1 时, IPv4 帧的时间戳快照 (Time Stamp Snapshot) 启用
TSIPV6ENA: Enable Time Stamp Snapshot for IPv6 frames 启用 IPv6 时间戳快照	12	0x0	该位为 1 时, IPv6 帧的时间戳快照 (Time Stamp Snapshot) 启用
TSIPENA: Enable Time Stamp Snapshot for PTP over Ethernet frames 启用以太网 PTP 包时间戳快照	11	0x0	该位为 1 时, 以太网格式的 PTP 包 (PTP over Ethernet) 启用时间戳快照 (Time Stamp Snapshot); 为 0 时 UDP-IP-Ethernet 格式的 PTP 包启用时间戳快照
TSVER2ENA: Enable PTP packet snooping for version 2 format 基于第二版本格式启用 PTP 包快照	10	0x0	该位为 1 时, 以太网格式的 PTP 包快照为第二版本格式 (version 2 format); 为 0 时为第一版本格式 (version 1 format)
TSCTRLSSR: Time Stamp Digital or Binary rollover control 时间戳数字或二进制更新控制	9	0x0	该位为 1 时, 时间戳低位寄存器 (Time Stamp Low register) 在值超过 0x3B9A_C9FF 后更新且增加时间戳高位寄存器的秒; 该位为 0 时, 时间戳低位寄存器复位为 0x7FFF_FFFF

名称	位域	初值	描述
TSENALL: Enable Time Stamp for All Frames 对所有帧启用时间戳	8	0x0	该位为 1 时, 表示对所有 GMAC 接收到的帧启用时间戳
Reserved 保留	7:6	0x0	保留
TSADDREG: Addend Reg Update 被加数寄存器	5	0x0	该位为 1 时, 时间戳被加数寄存器 (Time Stamp Addend register) 的值被用来更新 PTP 块来做 fine correction. 当更新结束时该位为 0
TSTRIG: Time Stamp Interrupt Trigger Enable 启用时间戳中断触发	4	0x0	该位为 1 时, 当系统时间大于目标时间寄存器 (Target Time register) 中的值时, 产生一个时间戳中断; 中断产生后该位重置为 0
TSUPDT: Time Stamp Update 时间戳更新	3	0x0	该位为 1 时, 系统根据时间戳更新高位寄存器和时间戳低位寄存器的值 (Time Stamp High Update and Time Stamp Low Update registers) 更新当前系统时间, 完成后该位为 0
TSINIT: Time Stamp Initialize 时间戳初始化	2	0x0	该位为 1 时, 系统根据时间戳更新高位寄存器和时间戳低位寄存器的值 (Time Stamp High Update and Time Stamp Low Update registers) 初始化当前系统时间, 完成后该位为 0
TSCFUPDT: Time Stamp Fine or Coarse Update 时间戳 Fine or Coarse 更新	1	0x0	该位为 1 时, 表示时间戳更新将会用 Fine Update 的方式, 为 0 时表示时间戳更新将会用 Coarse 的方式
TSENA: Time Stamp Enable 时间戳启用	1	0x0	该位为 1 时, 表示当前对于接收和发送帧系统将会启用时间戳记录; 为 0 时表示当前停止向接收和发送帧进行时间戳记录, 并且暂停时间戳发生器; 该为置 1 后应该接着进行时间戳初始化 (bit 2)

10.2.23 Register449 (Sub-Second Increment Register)

表 10.39: Register449 (Sub-Second Increment Register)

名称	位域	初值	描述
Register449 (Sub-Second Increment Register) Offset: 0x0704			
Reserved 保留	31:8	0x0	保留
SSINC: Sub-second increment value 亚秒增加值	7:0	0x0	此域中的值将会被加到亚秒寄存器 (sub-second register)

10.2.24 Register 450 (System Time - Seconds Register)

表 10.40: Register 450 (System Time - Seconds Register)

名称	位域	初值	描述
Register 450 (System Time - Seconds Register) Offset: 0x0708			
TSS: Time Stamp Second 时间戳秒	31:0	0x0	该域的值表示当前系统时间的秒级计数

10.2.25 Register 451 (System Time - Nanoseconds Register)

表 10.41: Register 451 (System Time - Nanoseconds Register)

名称	位域	初值	描述
Register 451 (System Time - Nanoseconds Register) Offset: 0x070C			
PSNT: Positive or Negative Time 正或负时间	31	0x0	该域表示正或负的时间, 当为 1 时表示时间为正数, 为 0 时表示为负数
TSSS: Time Stamp Sub Seconds 时间戳亚秒	30:0	0x0	该域的值表示当前系统时间的亚秒级计数, 默认时间精度为 0.46 纳秒 (当 TSCTRLSSR 置起时精度为 1ns, 最大值为 0x3B9A_C9FF)

10.2.26 Register 452 (System Time - Seconds Update Register)

表 10.42: Register 452 (System Time - Seconds Update Register)

名称	位域	初值	描述
Register 452 (System Time - Seconds Update Register) Offset: 0x0710			
TSS: Time Stamp Second 时间戳秒	31:0	0x0	该域的值在系统初始化时的初值, 或是更新时的增加值

10.2.27 Register 453 (System Time - Nanoseconds Update Register)

表 10.43: Register 453 (System Time - Nanoseconds Update Register)

名称	位域	初值	描述
Register 453 (System Time - Nanoseconds Update Register) Offset: 0x0714			
ADDSUB: Add or subtract time 加或减时间	31	0x0	该域表示加或减时间, 当为 1 时表示更新时减去更新寄存器的值, 为 0 时表示增加更新寄存器的值
TSSS: Time Stamp Sub Seconds 时间戳亚秒	30:0	0x0	该域的值表示当前系统时间更新时增加或减少的亚秒级计数, 默认时间精度为 0.46 纳秒 (当 TSCTRLSSR 置起时精度为 1ns, 最大值为 0x3B9A_C9FF)

10.2.28 Register 454 (Time Stamp Addend Register)

表 10.44: Register 454 (Time Stamp Addend Register)

名称	位域	初值	描述
Register 454 (Time Stamp Addend Register) Offset: 0x0718			
TSAR: Time Stamp Addend 时间戳加数寄存器	31:0	0x0	该域的值表示在系统时间同步时需要被增加的 32 位时间值

10.2.29 Register 455 (Target Time Seconds Register)

表 10.45: Register 455 (Target Time Seconds Register)

名称	位域	初值	描述
Register 455 (Target Time Seconds Register) Offset: 0x071C			
TSTR: Target Time Seconds Register 目标时间秒寄存器	31:0	0x0	该域保存了以秒为单位的时间, 当时间戳时间匹配或超过了该域的值及低位寄存器 (reg 456) 的值且包 MAC 地址匹配, 系统会产生一个中断 (中断必须事先启用)

10.2.30 Register 456 (Target Time Nanoseconds Register)

表 10.46: Register 456 (Target Time Nanoseconds Register)

名称	位域	初值	描述
Register 456 (Target Time Nanoseconds Register) Offset: 0x0720			
Reserve 保留	31	0x0	
TSTR: Target Time Stamp Low Register 目标时间戳低寄存器	30:0	0x0	该域保存了以纳秒为单位的时间, 当时间戳时间匹配或超过了该域的值及目标时间寄存器 (reg 455) 的值且包 MAC 地址匹配, 系统会产生一个中断 (中断必须事先启用)

10.2.31 Register 457 (System Time - Higher Word Seconds Register)

表 10.47: Register 457 (System Time - Higher Word Seconds Register)

名称	位域	初值	描述
Register 457 (System Time - Higher Word Seconds Register) Offset: 0x0724			
Reserve 保留	31:16	0x0	
TSHWR: Time Stamp Higher Word Register 时间戳高位寄存器	15:0	0x0	该域保存高 16 位 (most significant bit) 的时间戳值

10.2.32 Register 458 (Time Stamp Status Register)

表 10.48: Register 458 (Time Stamp Status Register)

名称	位域	初值	描述
Register 458 (Time Stamp Status Register) Offset: 0x0728			
Reserve 保留	31:28	0x0	
ATSNS: Auxiliary Time Stamp Number of Snapshots 附属时间戳快照个数	27:25	0x0	该域表示了 FIFO 中快照的个数, 4(3'b100) 表示 FIFO 满, 0(3'000) 表示 FIFO 空
ATSSTM: Auxiliary Time Stamp Snapshot Trigger Missed 附属时间戳快照触发缺失	24	0x0	该位为 1 时表示当前 FIFO 满且设置了外部触发, 表示最后一个快照因为 FIFO 满被丢失了; 否则为 0

名称	位域	初值	描述
Reserve 保留	23:3	0x0	
Auxiliary Time Stamp Trigger Snapshot 附属时间戳快照触发	2	0x0	该位为 1 时表示有快照 (snapshot) 写入了 FIFO
TSTARGET: Time Stamp Target Time Reached 达到时间戳目标时间	1	0x0	该位为 1 时表示系统时间已经大于或等于目标时间戳寄存器中的值
TSSOVF: Time Stamp Seconds Overflow 时间戳秒溢出	0	0x0	该位为 1 时表示当前时间戳的时间已经溢出 (大于 0xFFFF_FFFF) ; 反之为 0

10.2.33 Register 459 (PPS Control Register)

表 10.49: Register 459 (PPS Control Register)

名称	位域	初值	描述
Register 459 (PPS Control Register)	Offset: 0x072C		
Reserve 保留	31:4	0x0	
PPSCTRL: Control the duration between 2-pulses of PPS signal output 2 个 PPS 信号输出的时间间隔控制	15:0	0x0	该域设置 2 个 PPS 信号输出的时间间隔, 编码如下: 0000: 1.0 秒 (Binary & Digital Rollover) 0001: 0.5 秒 (Binary Rollover), 0.536 秒 (Digital Rollover) 0010: 0.25 秒 (Binary Rollover), 0.26 秒 (Digital Rollover) 0011: 0.125 秒 (Binary Rollover), 0.13 秒 (Digital Rollover) ... 1111: 15.28 μ 秒 (Binary Rollover), 32.77 μ 秒 (Digital Rollover)

10.2.34 Register 460 (PPS Auxiliary Time Stamp - Nanoseconds Register)

表 10.50: Register 460 (PPS Auxiliary Time Stamp - Nanoseconds Register)

名称	位域	初值	描述
Register 460 (PPS Auxiliary Time Stamp - Nanoseconds Register)	Offset: 0x0730		
	31:0	0x0	包含了附属时间戳 (Auxiliary Time Stamp) 的低 (纳秒)32 位值

10.2.35 Register 461 (PPS Auxiliary Time Stamp - Seconds Register)

表 10.51: Register 461 (PPS Auxiliary Time Stamp - Seconds Register)

名称	位域	初值	描述
Register 461 (PPS Auxiliary Time Stamp - Seconds Register) Offset: 0x0730			
	31:0	0x0	包含了附属时间戳 (Auxiliary Time Stamp) 的高 (秒)32 位值

10.3 DMA 描述符

DMA 描述符是 GMAC 驱动和硬件的交互接口，记录了数据包的内存地址和传输状态。在此分别定义了发送描述符 (Tx Descriptor) 和接收描述符 (Rx Descriptor) 两种数据结构。两种描述符可以自由选择分别以环式 (ring mode) 或者链式 (chain mode) 相连，以供 GMAC 使用。本芯片集成的 GMAC 控制器采用增强型描述符格式 (enhanced descriptor formate)，具体介绍如下：

增强型发送描述符 (enhanced transmit descriptor): 描述符大小为 32 字节，依次为 TDES0-7，其整体结构如图10.1。

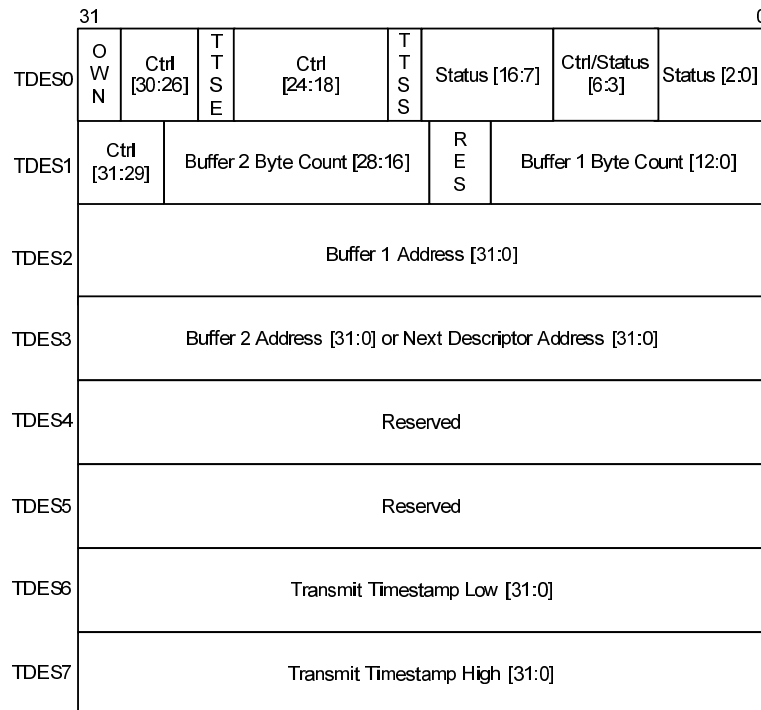


图 10.1: 增强型发送描述符

其中每一位具体意义如下表：

10.3.1 TDES0

表 10.52: TDES0

名称	位域	描述
OWN 所属模式	31	该位为 1 时表示描述符当前属于 DMA 控制, 0 表示属于主机控制。当 DMA 模块完成一次传输时, 会将该位主动清 0
IC: Interruption on Complete 完成时中断	30	该位为 1 时表示该帧接发送完成后将会置起 STATUS 寄存器中 TI 位 (CSR5[0])
LS: Last Segment 最后段	29	该位为 1 时表示当前 buffer 包含的是一帧数据的最后一段 (如果帧分为多个段)
FS: First Segment 第一段	28	该位为 1 时表示当前 buffer 包含的是一帧数据的第一段 (如果帧分为多个段)
DC: Disable CRC 禁止 CRC 校验	27	该位为 1 时 GMAC 硬件不在每个发送帧的结尾添加 CRC 校验数据
DP: Dissable Pading 禁止填充	26	该位为 1 时表示 GMAC 将不会对长度小于 64 字节的数据包进行空数据填充
TTSE: Transmit Time Stamp Enable 启用发送时间戳	25	该位为 1 时表示将启用内部模块计算 IEEE1588 硬件时间戳计算, 在 TDES0[28] 为 1 时有效
Reserved 保留	24	
CIC: Checksum Insertion Control 校验数据填充控制	23:22	该域控制内部模块是否在发送帧中填充校验数据。 2'b00: 不填充校验数据 2'b01: 填充 IP 头校验数据 2'b10: 填充 IP 头和负载 (payload) 的校验数据, 硬件不计算伪头数据 (pseudo-header) 校验 2'b11: 填充 IP 头和负载 (payload) 的校验数据, 硬件计算伪头数据 (pseudo-header) 校验
TER: Transmit End of Ring 环形描述符结尾	21	该位为 1 时表示该描述符为环型描述符链表的最后一个, 下一个描述符的地址为发送描述符链的基址
TCH: Second Address Chained 第二个 buffer 地址指向下个链式 描述符	20	该位为 1 时表示描述符中的第二个 buffer 地址指向的是下一个描述符的地址, 为 0 时表示该地址指向第二个 buffer 地址 当该位为 1 时, TDES1[28-16] 的值将没有意义, TDES0[21] 比 TDES0[20] 具有更高优先级 (代表环型而不是链型)
Reserved 保留	19:18	
TTSS: Tx Time Stamp Status 发送时间戳状态	17	该位为 1 表示 TDES6 和 TDES7 中保存了该发送帧的时间戳信息。该位在 TDES0[29] 为 1 时有效
IHE: IP Header Error IP 头错误	16	该位为 1 时表示内部校验模块发现该发送帧的 IP 头出错, 并且不会对该域做任何修改
ES: Error Summary 总体错误信息	15	指示当前帧是否出错, 其值为 TDES[1]、TDES[2]、TDES[8]、TDES[9]、TDES[10]、TDES[11]、TDES[12]、TDES[13]、TDES[14]、TDES[16] 各位作或运算 (OR) 的结果
JT: Jabber Timeout Jabber 超时	14	该位为 1 时表示 GMAC 发送模块遇到了 Jabber 超时
FF: Frame Flushed 帧刷新	13	该位为 1 时表示软件发出了一个刷新命令导致 DMA/MTL 将其内部的帧刷新掉
IPE: IP Payload Error IP 负载错误	12	该位为 1 时表示 GMAC 检测到所发出的 TCP/UDP/ICMP 包的负载数据出错。GMAC 比较协议头中的负载长度和实际发出的数据负载长度, 当二者不匹配时发出一个错误状态。

名称	位域	描述
LC: Loss of Carrier 载波丢失	11	该位为 1 时表示在发送该帧过程中载波丢失 (gmii_lcrs 信号多个周期未置起)
NC: No Carrier 载波无效	10	该位为 1 时表示在发送过程中, PHY 的载波信号一直未置起
LC: Late Collision 后期冲突	9	当该位为 1 时表示在半双工模式下, 当前帧接收时发生了一个后期冲突
EC: Excessive Collison 连续冲突	8	当该位为 1 时表示在发送当前帧的时候连续出现了 16 次冲突
VF: VLAN Frame VLAN 帧	7	该位为 1 时表示当前发送帧为一个 VLAN 帧
CC: Collison Count 冲突计数	6:3	该域表示当前帧在成功发送之前所遇到冲突次数的总数。当 TDES0[8] 为 1 时有效。
ED: Excessive Deferral 连续 Deferral	2	该位为 1 时表示当前帧传输结束
UF: Underflow Error 溢出错误	1	该位为 1 时表示当前帧传输时发生了溢出错误, 即数据传输 buffer 过小或不可用
DB: Deferred Bit 帧刷新	0	该位为 1 时表示此次发送被延迟, 只有在半双工模式下有效

10.3.2 TDES1

表 10.53: TDES1

名称	位域	描述
Reserved 保留	31:29	
TBS2: Transmit Buffer Size 2 发送 buffer2 大小	28:16	该域表示数据 buffer2 的大小。当 TDES0[20] 为 1 时, 该域无效
Reserved 保留	15:13	
TBS1: Transmit Buffer Size 1 发送 buffer1 大小	12:0	该域表示数据 buffer1 的大小。该域一直有效。如果该域值为 0, DMA 则会自动访问 buffer2 或者下一个接收描述符

10.3.3 TDES2

表 10.54: TDES2

名称	位域	描述
Buffer1 Address Pointer 发送 buffer1 地址	31:0	该域记录了数据接收 buffer1 的 32 位物理地址。该物理地址没有默认的对齐要求。当 GMAC DMA 内部实现了总线数据 32/64/128 位对齐, 则该地址的低 2/3/4 位会被忽略

10.3.4 TDES3

表 10.55: TDES3

名称	位域	描述
Buffer2 Address Pointer 发送 buffer2 地址	31:0	该域记录了数据接收 buffer2 的 32 位物理地址。该物理地址没有默认的对齐要求。当 GMAC DMA 内部实现了总线数据 32/64/128 位对齐, 则该地址的低 2/3/4 位会被忽略。如果描述符是以链式连接, 则该域记录的是下一个描述符的地址。

10.3.5 TDES6

记录了时间戳低 32 位值

表 10.56: TDES6

名称	位域	描述
TTSL: Transmit Frame Time Stamp Low 发送帧时间戳低位	31:0	该域记录了 GMAC 所捕捉的发送帧时间戳低 32 位 (LSB) 有效值, 由 DMA 模块负责更新。该位只有当 LS(TDES0[29]) 和 TTSE(TDES0[25]) 同时为 1 时有效。

10.3.6 TDES7

记录了时间戳高 32 位值

表 10.57: TDES7

名称	位域	描述
TTSL: Transmit Frame Time Stamp Low 发送帧时间戳低位	31:0	该域记录了 GMAC 所捕捉的发送帧时间戳高 32 位 (MSB) 有效值, 由 DMA 模块负责更新。该位只有当 LS(TDES0[29]) 和 TTSE(TDES0[25]) 同时为 1 时有效。

增强型接收描述符 (enhanced receive descriptor): 描述符大小为 32 字节, 依次为 RDES0-7, 其整体结构图10.2: 其中每一位具体意义如下表:

10.3.7 RDES0

表 10.58: RDES0

名称	位域	描述
OWN 所有模式	31	该位为 1 时表示描述符当前属于 DMA 控制, 0 表示属于主机控制。当 DMA 模块完成一次传输时, 会将该位主动清 0
AFM: Destination Address Filter Fai 目标地址过滤错误 1	30	当该位为 1 时, 表示当前数据帧目标地址不符合 GMAC 内部的帧目标地址过滤器
FR: Frame length 帧长度	29:16	表示接收当前帧的长度, 当 ES 位为 0 时有效
ES: Error Summary 总体错误信息	15	指示当前帧是否出错, 其值为 RDES[0]、RDES[1]、RDES[3]、RDES[4]、RDES[6]、RDES[7]、RDES[11]、RDES[14] 各位作或运算 (OR) 的结果
DE: Descriptor Error 描述符错误	14	当该位为 1 时表示, 当前描述符所指向的 buffer 与帧不相符或者 OWN 为 0(主机控制)

名称	位域	描述
SAF: Source Address Filter Fail 源地址过滤错误	13	当该位为 1 时, 表示当前数据帧的源地址不符合 GMAC 内部的帧源地址过滤器
LE: Length Error 长度错误	12	当该位为 1 时, 表示当前接收帧长度与默认长度不符。当 Frame Type 位为 1 且 CRC Error 位为 0 时有效
OE: Over Flow Error 溢出错误	11	当该位为 1 时, 表示接收该帧时 GMAC 内部 RxFIFO 溢出
VLAN: VLAN Tag VLAN 标志	10	当该位为 1 时, 表示该帧的类型为 VLAN
FS: First Descriptor 第一个描述符	9	当该位为 1 时, 表示当前描述符所指向的 buffer 为当前接收帧的第一个保存 buffer
LS: Last Descriptor 最后一个描述符	8	当该位为 1 时, 表示当前描述符所指向的 buffer 为当前接收帧的最后一个保存 buffer
IPC Checksum Error/Giant Frame 校验错误 / 超长帧	7	当该位为 1 时, 如果 IPC 校验功能启用则表示当前帧的 IPv4 头校验值与帧内部校验域的值不相符。如果未启用则表示当前帧为一个超长帧 (长度大于 1518 字节)
LC: late collision 后期冲突	6	当该位为 1 时, 表示在半双工模式下, 当前帧接收时发生了一个后期冲突
FT: Frame Type 帧类型	5	当该位为 1 时, 表示当前帧为一个以太网格式帧, 为 0 时表示当前帧为一个 IEEE802.3 格式帧
RWT: Receive Watchdog Timeout	4	当该位为 1 时, 表示当前时钟值超过了接收模块看门狗电路时钟的值, 既接收帧超时
RE: Receive Error 接收错误	3	当该位为 1 时, 表示接收当前帧时内部模块出错。内部信号 rxer 置 1 且 rxdv 置 1
DE: Dribble bit Error 奇数位错误	2	当该位为 1 时, 表示接收帧长度不是整数, 即总长度为奇数位, 该位只有在 mii 模式下有效
CE: CRC Error 接收 CRC 校验错误	1	当该位为 1 时, 表示接收当前帧时内部 CRC 校验出错。该位只有在 last descriptor(RDES0[8]) 为 1 时有效
Extended Status Available/Rx MAC Address 扩展状态可用 / 接收 MAC 地址	0	当高级时间戳 (Advanced Time Stamp) 或者 ip 校验去载 (IP Checksum Offload) 功能启用时, 该位为 1 表示 RDES4 中记录的扩展状态是有效的。该位仅在 LS(RDES0[8]) 为 1 时有效。 当高级时间戳 (Advanced Time Stamp) 和 ip 完全校验去载功能未被启用时 (IPC Full Offload), 该位表示接收 MAC 地址状态。当该位为 1 时表示接收帧的目的 MAC 地址与 DA(Destination Filter) 寄存器的值匹配, 反之为不匹配。

10.3.8 RDES1

表 10.59: RDES1

名称	位域	描述
Disable Intr in Completion 禁止完成后发中断	31	该位为 1 时表示该帧接收完成后将不会置起 STATUS 寄存器中 RI 位 (CSR5[6]), 这将会使得主机无法检测到该中断
Reserved 保留	30:29	
RBS2: Receive Buffer Size 2 接收 buffer2 大小	28:16	该域表示数据 buffer2 的大小。根据系统总线的宽度 32/64/128, Buffer2 的大小应该为 4/8/16 的整数倍。如果不满足则会导致未知的结果。该域在 RDES1[14] 为 0 时有效
RER: Receive End of Ring 环型描述符结尾	15	该位为 1 时表示该描述符为环型描述符链表的最后一个, 下一个描述符的地址为接收描述符链的基址

名称	位域	描述
RCH: Second Address Chained 第二个 buffer 地址指向下个链式描述符	14	该位为 1 时表示描述符中的第二个 buffer 地址指向的是下一个描述符的地址，为 0 时表示该地址指向第二个 buffer 地址 当该位为 1 时，RDES1[21-11] 的值将没有意义，RDES1[25] 比 RDES1[24] 具有更高优先级（代表环型而不是链型）
Reserved 保留	13	
RBS2: Receive Buffer Size 1 接收 buffer1 大小	12:0	该域表示数据 buffer1 的大小。根据系统总线的宽度 32/64/128，Buffer1 的大小应该为 4/8/16 的整数倍。如果不满足则会导致未知的结果。该域一直有效。如果该域值为 0，DMA 则会自动访问 buffer2 或者下一个接收描述符

10.3.9 RDES2

表 10.60: RDES2

名称	位域	描述
Buffer1 Address Pointer 接收 buffer1 地址	31:0	该域记录了数据接收 buffer1 的 32 位物理地址。该物理地址没有默认的对齐要求。当 GMAC DMA 内部实现了总线数据 32/64/128 位对齐，则该地址的低 2/3/4 位会被忽略

10.3.10 RDES3

表 10.61: RDES3

名称	位域	描述
Buffer2 Address Pointer 接收 buffer2 地址	31:0	该域记录了数据接收 buffer2 的 32 位物理地址。该物理地址没有默认的对齐要求。当 GMAC DMA 内部实现了总线数据 32/64/128 位对齐，则该地址的低 2/3/4 位会被忽略。如果描述符是以链式连接，则该域记录的是下一个描述符的地址

10.3.11 RDES4

表 10.62: RDES4

名称	位域	描述
保留	31:14	
PTP Version PTP 版本	13	该位为 1 时表示接收到的 ptp 同步信息 (ptp message) 符合 IEEE1588 第二版 (version 2) 的规定。为 0 时，表示符合 IEEE1588 第一版 (version 1) 的规定
PTP Frame Type	12	该位为 1 时表示 ptp 包直接通过以太网 (ethernet) 发送，为 0 时且信息类型 (RDES4[11:8]) 不为 0 时，表示 ptp 包通过 UDP-ipv4 或 UDP-ipv6 发送

名称	位域	描述
Message Type 信息类型	11:8	表示当前信息类型 (ptp message type) , 按以下值编码: 0000: 没有 ptp 包收到 0001: SYNC 包 (all clock types) 0010: Follow_Up 包 (all clock types) 0011: Delay_Req 包 (all clock types) 0100: Delay_Resp 包 (all clock types) 0101: Pdelay_Req 包 (in peer-to-peer transparent clock) or Announce (in ordinary or boundary clock) 0110: Pdelay_Resp 包 (in peer-to-peer transparent clock) or Management (in ordinary or boundary clock) 0111: Pdelay_Resp_Follow_Up 包 (in peer-to-peer transparent clock) or Signaling (for ordinary or boundary clock) 1xxx - 保留
IPv6 Packet Received 收到 IPV6 包	7	该位为 1 时表示当前接收到的包是一个 ipv6 包
IPv4 Packet Received 收到 IPV4 包	6	该位为 1 时表示当前接收到的包是一个 ipv4 包
IP Checksum Bypassed IP 校验忽略	5	该位为 1 时表示校验去载引擎 (checksum offload engine) 被忽略 (bypassed)
IP Payload Error IP 负载错误	4	该位为 1 时表示 GMAC 检测到所接收的 TCP/UDP/ICMP 包的负载数据出错。GMAC 比较协议头中的负载长度和实际发出的数据负载长度, 当二者不匹配时发出一个错误状态。
IP Header Error IP 头错误	3	该位为 1 时表示内部校验模块发现该接收帧的 IP 头出错
IP Payload Type IP 负载类型	2:0	该 3 位表示经过接收校验去载引擎 (receive checksum offload engine) 计算后的包负载类型, 编码如下: 3'b000: 未知类型或未启用收校验去载引擎 3'b001: UDP 3'b010: TCP 3'b011: ICMP 3'b1xx: 保留

10.3.12 RDES6

记录时间戳低 32 位值

表 10.63: RDES6

名称	位域	描述
RTSL: Receive Frame Time Stamp Low 接收帧时间戳低位	31:0	该域记录了 GMAC 所捕捉的接收帧时间戳低 32 位 (LSB) 有效值, 由 DMA 模块负责更新。该位只有当 LS(TDES0[29]) 和 TTSE(TDES0[25]) 同时为 1 时有效

10.3.13 TDES7

记录时间戳高 32 位值

表 10.64: TDES7

名称	位域	描述
RTSL: Receive Frame Time Stamp Low 接收帧时间戳低位	31:0	该域记录了 GMAC 所捕捉的接收帧时间戳高 32 位 (MSB) 有效值, 由 DMA 模块负责更新。该位只有当 LS(TDES0[29]) 和 TTSE(TDES0[25]) 同时为 1 时有效

10.4 软件编程向导

10.4.1 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

10.4.2 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

10.4.3 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

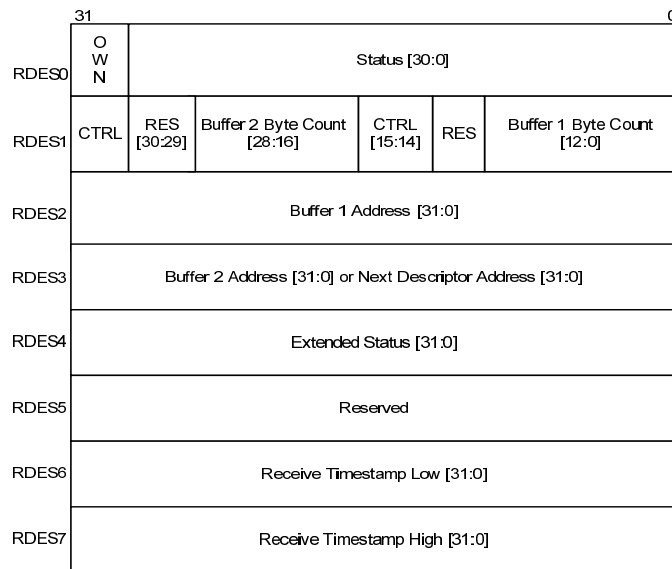


图 10.2: 增强型接收描述符

第十一章 SATA 控制器

11.1 SATA 总体描述

SATA 的特性包括：

- 支持 SATA 1 代 1.5Gbps 和 SATA2 代 3Gbps 的传输
- 兼容串行 ATA 2.6 规范和 AHCI 1.1 规范

11.2 SATA 控制器内部寄存器描述

SATA 的基地址是 0xbfe30000，寄存器的定义和协议标准定义完全一致。

表 11.1: SATA 控制器内部寄存器

地址	位宽	名称	描述
0xbfe3,0000	32	CAP	HBA 特性寄存器
0xbfe3,0004	32	GHC	全局 HBA 控制寄存器
0xbfe3,0008	32	IS	中断状态寄存器
0xbfe3,000c	32	PI	端口寄存器
0xbfe3,0010	32	VS	AHCI 版本寄存器
0xbfe3,0014	32	CCC_CTL	命令完成合并控制寄存器
0xbfe3,0018	32	CCC_PORTS	命令完成合并端口寄存器
0xbfe3,0024	32	CAP2	HBA 特性扩展寄存器
0xbfe3,00A0	32	BISTAFR	BIST 激活 FIS
0xbfe3,00A4	32	BISTCR	BIST 控制寄存器
0xbfe3,00A8	32	BISTCTR	BIST FIS 计数寄存器
0xbfe3,00AC	32	BISTSR	BIST 状态寄存器
0xbfe3,00B0	32	BISTDECR	BIST 双字错计数寄存器
0xbfe3,00BC	32	OOBR	OOB 寄存器
0xbfe3,00E0	32	TIMER1MS	1ms 计数寄存器
0xbfe3,00E8	32	GPARAM1R	全局参数寄存器 1
0xbfe3,00EC	32	GPARAM2R	全局参数寄存器 2
0xbfe3,00F0	32	PPARAMR	端口参数寄存器
0xbfe3,00F4	32	TESTR	测试寄存器
0xbfe3,00F8	32	VERIONR	版本寄存器
0xbfe3,00FC	32	IDR	ID 寄存器
0xbfe3,0100	32	P0_CLB	命令列表基地址低 32 位
0xbfe3,0104	32	P0_CLBU	命令列表基地址高 32 位
0xbfe3,0108	32	P0_FB	FIS 基地址低 32 位
0xbfe3,010c	32	P0_FBU	FIS 基地址高 32 位
0xbfe3,0110	32	P0_IS	中断状态寄存器
0xbfe3,0114	32	P0_IE	中断使能寄存器

地址	位宽	名称	描述
0xbfe3,0118	32	P0_CMD	命令寄存器
0xbfe3,0120	32	P0_TFD	任务文件数据寄存器
0xbfe3,0124	32	P0_SIG	签名寄存器
0xbfe3,0128	32	P0_SSTS	SATA 状态寄存器
0xbfe3,012C	32	P0_SCTL	SATA 控制寄存器
0xbfe3,0130	32	P0_SERR	SATA 错误寄存器
0xbfe3,0134	32	P0_SACT	SATA 激活寄存器
0xbfe3,0138	32	P0_CI	命令发送寄存器
0xbfe3,013C	32	P0_SNTF	SATA 命令通知寄存器
0xbfe3,0170	32	P0_DMACR	DMA 控制寄存器
0xbfe3,0178	32	P0_PHYCR	PHY 控制寄存器
0xbfe3,017C	32	P0_PHYSR	PHY 状态寄存器
0xbfe3,0180	32	P1_CLB	命令列表基地址低 32 位
0xbfe3,0184	32	P1_CLBU	命令列表基地址高 32 位
0xbfe3,0188	32	P1_FB	FIS 基地址低 32 位
0xbfe3,018c	32	P1_FBU	FIS 基地址高 32 位
0xbfe3,0190	32	P1_IS	中断状态寄存器
0xbfe3,0194	32	P1_IE	中断使能寄存器
0xbfe3,0108	32	P1_CMD	命令寄存器
0xbfe3,01a0	32	P1_TFD	任务文件数据寄存器
0xbfe3,01a4	32	P1_SIG	签名寄存器
0xbfe3,01a8	32	P1_SSTS	SATA 状态寄存器
0xbfe3,01aC	32	P1_SCTL	SATA 控制寄存器
0xbfe3,01b0	32	P1_SERR	SATA 错误寄存器
0xbfe3,01b4	32	P1_SACT	SATA 激活寄存器
0xbfe3,01b8	32	P1_CI	命令发送寄存器
0xbfe3,01bC	32	P1_SNTF	SATA 命令通知寄存器
0xbfe3,01f0	32	P1_DMACR	DMA 控制寄存器
0xbfe3,01f8	32	P1_PHYCR	PHY 控制寄存器
0xbfe3,01fC	32	P1s_PHYSR	PHY 状态寄存器

第十二章 USB 控制器

12.1 总体描述

2H 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
- 支持六个端口，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。同时 USB 控制器内部集成了 DMA 控制器，用来和内存 / 应用程序之间通信。

12.2 USB 主机控制器寄存器

12.2.1 EHCI 相关寄存器

EHCI 的相关寄存器包括 Capability 寄存器、Operational 寄存器和，EHCI 实现相关寄存器。2H 的 USB 主机控制器兼容 EHCI Rev 1.0 协议，Capability 寄存器和 Operational 寄存器的详细信息参照 Enhanced Host Controller Interface Rev 1.0 Specification 。EHCI 寄存器的基址为 0xbfe00000 ，以下寄存器的地址为基址加偏移。

12.2.2 Capability 寄存器

名称	偏移	宽度	访问	说明
HCCAPBASE	0x00	32	RO	默认值为32'h01000010
HCSPARAMS	0x04	32	RO	默认值为32'h00001116
HCCPARAMS	0x08	32	RO	默认值为32'h0000A010

12.2.3 Operational 寄存器

名称	偏移	宽度	访问	说明
USBCMD	0x10	32	R/W 、 RO	USB 主机控制器的命令寄存器
USBSTS	0x14	32	R/W 、 RO	USB 主机控制器的状态寄存器
USBINTR	0x18	32	R/W	USB 主机控制器的中断设置寄存器
FRINDEX	0x1c	32	R/W	USB 主机控制器的帧索引寄存器

名称	偏移	宽度	访问	说明
CTRLDSSEGMENT	0x20	32	R/W	存放 EHCI 控制数据结构的地址
PERIODICLISTBASE	0x24	32	R/W	存放周期数据帧表的起始地址
ASYNCLISTADDR	0x28	32	R/W	存放下一个要被执行的异步队列的起始地址
CONFIGFLAG	0x50	32	R/W	配置模式寄存器
PORTSC 1	0x54	32	R/W, RO	端口 1 状态和控制寄存器
PORTSC 2	0x58	32	R/W, RO	端口 2 状态和控制寄存器
PORTSC 3	0x5c	32	R/W, RO	端口 3 状态和控制寄存器
PORTSC 4	0x60	32	R/W, RO	端口 4 状态和控制寄存器
PORTSC 5	0x64	32	R/W, RO	端口 5 状态和控制寄存器
PORTSC 6	0x88	32	R/W, RO	端口 6 状态和控制寄存器

12.2.4 EHCI 实现相关寄存器

EHCI 实现相关寄存器的详细描述如下。

名称	偏移	宽度	访问	说明
INSNREG00	0x90	32	R/W	帧的长度配置寄存器
INSNREG01	0x94	32	R/W	数据包缓冲区 OUT/IN 阈值寄存器
INSNREG02	0x98	32	RO	数据包缓冲深度寄存器
INSNREG03	0x9c	32	RO, R/W	参照寄存器详细描述
INSNREG04	0xa0	32	R/W	用于 Debug
INSNREG05	0xa4	32	RO, R/W	UTMI 配置 (默认配置), 控制和状态寄存器
INSNREG06	0xa8	32	RO	AHB 错误状态寄存器
INSNREG07	0xac	32	RO	AHB Master 错误地址寄存器
INSNREG08	0xb0	32	RO	HSIC 使能寄存器

12.2.4.1 INSNREG01 寄存器

位域	访问	初值	描述
31:16	R/W	16'h0020	OUT 阈值 (单位是 4 bytes), 一旦从系统内存中取出的数据量达到 OUT 阈值, 就开始 USB 传输, 最小为 16bytes
15:0	R/W	16'h0020	IN 阈值 (单位是 4 bytes), 一旦 Packet Buffer 里的数据量达到 IN 阈值, 就开始向内存传输, 最小为 16bytes

12.2.4.2 INSNREG02 寄存器

位域	访问	初值	描述
31:12	-	20'h0	保留
11:0	RO	12'h0020	数据包缓冲深度 (单位是 4 bytes)

12.2.4.3 INSNREG03 寄存器

位域	访问	初值	描述
31:13	-	19'h0	保留
12:10	RO	3'h0	这个字段指定 phy_clks 的额外延时, 这个延时被添加到 “Tx-Tx turnaround Delay” 中。
9	RO	1'h0	置 1: 将迫使主机控制器在一帧的每一微帧中获取周期数据帧表, 置 0: 主机控制器在一帧的微帧 0 中获取周期数据帧表
8:1	R/W	8'h0	时间可容忍偏移, 这个字段用来指明为了容忍计算可用时间而要附加的字节数。计算可用时间为以后的传输弹性增加的, 用户程序默认不需要修改这个字段。
0	RO	1'h0	Break Memory Transaction 模式 置 1: 使能此功能 置 0: 禁止此功能

12.2.4.4 INSNRE04 寄存器 (仅用于调试, 软件不必更改此寄存器)

位域	访问	初值	描述
31:6	-	26'h0	保留
5	R/W	1'h0	置 1: 禁止 automatic 功能, 即当软件清除 Run/Stop 位时, USB 主机控制器会把挂起 (Suspend) 的端口唤醒 置 0: 启用 automatic 功能, 当 reset Run/Stop 位时, Suspend 信号会置为 1
4	R/W	1'h0	置 1: 禁止 NAK reload 修复 置 0: 启用 NAK reload 修复
3	-	1'h0	保留
2	R/W	1'h0	置 1: 缩短端口枚举 (enumeration) 时间 (仿真)
1	R/W	1'h0	置 1: HCCPARAMS 寄存器的第 17、15:4、2:0 位均可写
0	R/W	1'h0	置 1: HCSPARAMS 寄存器可写

12.2.4.5 INSNRE05 寄存器

位域	访问	初值	描述
31:18	-	14'h0	保留
17	RO	1'h0	置 1: 表示对这个寄存器进行了一个写操作, 硬件正在执行 置 0: 表示硬件已经执行完操作
16:13	R/W	5'h0	端口号
12	R/W	4'h1	VControlLoadM 置 1: NOP 置 0: Load
11:8	R/W	4'h0	VControl
7:0	RO	4'h0	VStatus

12.2.4.6 INSNREG06 寄存器

位域	访问	初值	描述
31	R/W	1'h0	一旦 AHB 出错即被捕获并置 1，写 0 清除该字段
30:12	-	19'h0	保留
11:9	RO	3'h0	AHB 出错时控制段 HBURST 的值
8:4	RO	5'h0	AHB 出错的 burst 的预计节拍数
3:0	RO	4'h0	在当前 burst 下，AHB 出错前完成的节拍数

12.2.4.7 INSNREG07 寄存器

位域	访问	初值	描述
31:0	RO	32'h0	AHB 出错时控制段的地址

12.2.4.8 INSNREG08 寄存器

位域	访问	初值	描述
31:0	RO	1'b0	HSIC 使能

12.3 OHCI 相关寄存器

OHCI 的相关寄存器包括 Operational 寄存器和 OHCI 实现相关寄存器。2H 的 USB 主机控制器兼容 OHCI Rev 1.0 协议，Operational 寄存器的详细信息参照 Open Host Controller Interface Rev 1.0 Specification。OHCI 寄存器的基址为 0xbfe08000，以下寄存器的地址为基址加偏移。

12.3.1 Operational 寄存器

名称	偏移	宽度	访问	说明
HcRevision	0x00	32	-	控制和状态
HcControl	0x04	32	-	
HcCommonStatus	0x08	32	-	
HcInterruptStatus	0x0C	32	-	
HcInterruptEnable	0x10	32	-	
HcInterruptDisable	0x14	32	-	
HcHCCA	0x18	32	-	内存指针
HcPeriodCurrentED	0x1C	32	-	
HcControlHeadED	0x20	32	-	
HcControlCurrentED	0x24	32	-	
HcBulkHeadED	0x28	32	-	
HcBulkCurrentED	0x2C	32	-	
HcDoneHead	0x30	32	-	
HcRmInterval	0x34	32	-	帧计数器
HcFmRemaining	0x38	32	-	

名称	偏移	宽度	访问	说明
HcFmNumber	0x3C	32	-	
HcPeriodicStart	0x40	32	-	
HcLSThreshold	0x44	32	-	
HcRhDescriptorA	0x48	32	-	根集线器
HcRhDescriptorB	0x04C	32	-	
HcRhStatus	0x50	32	-	
HcRhPortStatus1	0x54	32	-	
HcRhPortStatus2	0x58	32	-	

12.3.2 OHCI 实现相关寄存器

除了标准的 OHCI 操作寄存器，还实现了两个额外寄存器（寄存器偏移 0x98 和 0x9C）用来报告总线的错误状态。

名称	偏移	宽度	访问	说明
INSNREG06	0x98	32	RO	错误状态寄存器
INSNREG07	0x9c	32	RO	错误地址寄存器

12.3.2.1 INSNREG06 寄存器

位域	访问	初值	描述
31	R/W	1'h0	一旦总线出错即被捕获并置 1，写 0 清除该字段
30:12	RO	19'h0	保留
11:9	RO	3'h0	出错时控制段 HBURST 的值
8:4	RO	5'h0	出错的 burst 的预计节拍数
3:0	RO	4'h0	在当前 burst 下，出错前完成的节拍数

12.3.2.2 INSNREG07 寄存器

位域	访问	初值	描述
31:0	RO	32'h0	总线出错时控制段的地址

第十三章 OTG 控制器

13.1 概述

2H 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 6 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；

13.2 寄存器列表

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。

CSR 的地址映射是固定的，其中 Host 模式下的寄存器与 Device 模式下的寄存器处于不同的地址空间，图13.1显示了 CSR 的地址映射关系：

以下章节分组列出了 OTG 所有的寄存器，及其偏移地址。

13.2.1 全局控制与状态寄存器 (Global CSR Map)

不管 OTG 工作在 host 模式下还是 Device 模式下都可访问这组寄存器，其对应的偏移地址如下表所示：

0000h	Core Global CSRs(1KB)
0400h	Host Mode CSRs(1KB)
0800h	Device Mode CSRs(1.5KB)
0E00h	Power and Clock Gating CSRs (1.5KB)
1000h	Device EP 0/Host Channel 0 FIFO (4KB)
2000h	Device EP 1/Host Channel 1 FIFO (4KB)
3000h	
7000h	Device EP 6/Host Channel 6 FIFO (4KB)
8000h	Reserved
20000h	Direct Access to Data FIFO RAM for Debugging (128KB)
3FFFFh	

图 13.1: OTG CSRs 地址映射

寄存器缩略名	偏移地址	寄存器名
GOTGCTL	000h	Control and Status Register
GOTGINT	004h	Interrupt Register
GAHBCFG	008h	AHB Configuration Register
GUSBCFG	00Ch	USB Configuration Register
GRSTCTL	010h	Reset Register
GINTSTS	014h	Interrupt Status Register
GINTMSK	018h	Interrupt Mask Register
GRXSTSR	01Ch	Receive Status Debug Read/Status Read and Pop Registers
GRXSTSP	020h	
GRXFSIZ	024h	Receive FIFO Size Register
GNPTXFSIZ	028h	Non-Periodic Transmit FIFO Size Register
GNPTXSTS	02Ch	Non-Periodic Transmit FIFO/Queue Status Register
GSNPSID	040h	Synopsys ID Register
GHWCFG1	044h	User HW Config1 Register
GHWCFG2	048h	User HW Config2 Register
GHWCFG3	04Ch	User HW Config3 Register
GHWCFG4	050h	User HW Config4 Register

寄存器缩略名	偏移地址	寄存器名
GDFIFOCFG	05Ch	DFIFO Software Config Register
HPTXFSIZ	100h	Host Periodic Transmit FIFO Size Register
DIEPTXF _n	104h-3FFh	Device IN Endpoint Transmit FIFO Size Register

13.2.2 模式控制与状态寄存器 (Host Mode CSR)

一旦 OTG 工作在 Host 模式下，这组寄存器必须被重新配置。

寄存器缩略名	偏移地址	寄存器名
1 HCFCG	400h	Host Configuration
HFIR	404h	Host Frame Interval Register
HFNUM	408h	Host Frame Number/Frame Time Remaining Register
	40ch	Reserved
HPTXSTS	410h	Host Periodic Transmit FIFO/Queue Status Register
HAINT	414h	Host All Channels Interrupt Register
HAINTMSK	418h	Host All Channels Interrupt Mask Register
HPRT	440h	Host Port Control and Status Register
	444h-4FCh	Reserved
HCCHAR _n	500h	Moving the Host Core to Test Mode
HCSPLT _n	504h	Host Channel-n Split Control Register (HCSPLT _n)
HCINT _n	508h	Host Channel-n Interrupt Register (HCINT _n)
HCINTMSK _n	50Ch	Host Channel-n Interrupt Mask Register (HCINTMSK _n)
HCTSIZ _n	510h	Host Channel-n Transfer Size Register (HCTSIZ _n)
HCDMA _n	514h	Host Channel-n DMA Address Register
	518h	Reserved
HCDMAB _n	51Ch	Host Channel-n DMA Buffer Address Register
HCCHAR _n	520h - 53Ch 540h-55h ... 6C0h - 6DCh 6E0h-6FCh 6FDh-7FFh	Host Channel-n Characteristics Register

13.2.3 模式控制与状态寄存器 (Device Mode CSR)

一旦 OTG 工作在 Device 模式下，这组寄存器必须被重新配置。

寄存器缩略名	偏移地址	寄存器名
1 DCFG	800h	Device Configuration Register
DCTL	804h	Device Control Register
DSTS	808h	Device Status Register
	80Ch	Reserved
DIEPMSK	810h	Device IN Endpoint Common Interrupt Mask Register
DOEPMSK	814h	Device OUT Endpoint Common Interrupt Mask Register
DAINT	818h	Device All Endpoints Interrupt Register

寄存器缩略名	偏移地址	寄存器名
DAINTMSK	81Ch	Device All Endpoints Interrupt Mask Register
DVBUSDIS	828h	Device VBUS Discharge Time Register
DVBUSPULSE	82Ch	Device VBUS Pulsing Time Register
DTHRCTL	830h	Device Threshold Control Register
DIEPEPMSK	834h	Device IN Endpoint FIFO Empty Interrupt Mask Register
DIEPCTL0	900h	Device Control IN Endpint0 Control Register
	904h	Reserved
DIEPCTLn	920-AE0h	Device Endpoint-n Control Register
DIEPINTn	908h	Device Endoint-n Interrupt Register
	90Ch	Reserved
DIEPTSIZE0/DOEPTSIZE0	910h	Device Endpoint0 Transfer Size Register
DIEPTSIZE _n /DOEPTSIZE _n	910h	Device Endpoint _n Transfer Size Register
DIEPDMA _n	914h	Device Endpoint-n DMA Address Register
DTXFSTSn	918h	Device IN Endpoint Transmit FIFO Status Register
DIEPDMA _{B0}	91Ch	Device Endpoint-n DMA Buffer Address Register
DOEPCTL0	B00h	Device Control OUT Endpint0 Control Register
	B04h	Reserved
DOEPCTLn	B20-BE0h	Device Endpoint-n Control Register
DOEPINTn	B08h	Device Endoint-n Interrupt Register
	B0Ch	Reserved
DOEPTSIZE0	B10h	Device Endpoint0 Transfer Size Register
DOEPTSIZE _n	B10h	Device Endpoint _n Transfer Size Register
DOEPDMA _n	B14h-CF4h	Device Endpoint-n DMA Address Register
DOEPDMA _{B0}	B1Ch-CFCh	Device Endpoint-n DMA Buffer Address Register

13.2.4 数据 FIFO 访问寄存器组 (DFIFO Access Register MAP)

不管 OTG 工作在 host 模式下还是 Device 模式下，都可访问这组寄存器。这组寄存器用作读写某个给定方向的 endpoint 或 channel 的 FIFO。如果 host 模式下 channel 的方向为 IN，那么只能读这个 channel 上的 FIFO；相类似地，若 host 模式下 channel 的方向为 OUT，那么只能写这个 channel 上的 FIFO。

寄存器名	偏移地址	读写特性
Device IN Endpoint 0/Host OUT Channel 0: DFIFO Write Access Device OUT Enpoint 0/Host IN Channel 0 : DFIFO Read Access	1000h-1FFCh	WO/RO
Device IN Endpoint 0/Host OUT Channel 0: DFIFO Write Access Device OUT Enpoint 0/Host IN Channel 0 : DFIFO Read Access	2000h-2FFCh	WO/RO
Device IN Endpoint 14/Host OUT Channel 14: DFIFO Write Access Device OUT Enpoint 14/Host IN Channel 14 : DFIFO Read Access	F000h-FFFCh	WO/RO
Device IN Endpoint 15/Host OUT Channel 15: DFIFO Write Access Device OUT Enpoint 15/Host IN Channel 15: DFIFO Read Access	10000h-10FFCh	WO/RO

13.2.5 功耗控制与门控时钟寄存器组 (Power and Clock Gating CSR Map)

不管 OTG 工作在 host 模式下还是 Device 模式下，都可访问这组寄存器。这组寄存器用作功耗控制与门控时钟。寄存器缩略名 偏移地址 寄存器名

寄存器缩略名	偏移地址	寄存器名
PCGCCTL	E00h	Power and Clock Gating Control Register

13.3 寄存器描述

13.3.1 寄存器访问特性

用来标示这些寄存器的读写特性，会有一栏访问特性，描述寄存器时在以下章节。下面的内容将列出这些访问特性的具体含义。

Read Only (RO)	只读
Write Only(WO)	只写
Read and Write(R_W)	可读可写
Read, Write, and Self Clear(R_W_SC)	可读可写；且 OTG 控制器自身可清 0，具体清 0 的条件在以下章节的各个域中有详细解释
Read, Write, Self Set and Self Clear (R_W_SS_SC)	可读可写；且 OTG 控制器自身可置位，可清 0，具体置位与清 0 的条件在以下章节的各个域中有详细解释
Read, Self Set, and Write Clear (R_SS_WC)	可读；且 OTG 控制器自身可置位；且当软件往这位写 1 完成清 0 效果，往这位写 0 不会产生效果。OTG 置位的条件在以下章节的各个域中有详细解释
Read, Write Set, and Self Clear (R_WS_SC)	可读；且当软件往这位写 1 完成置位效果；且 OTG 控制器自身可清 0，软件不可清 0，即软件往这位写 1 不会产生效果。OTG 清 0 的条件在以下章节的各个域中有详细解释
Read, Self Set, and Self Clear or Write Clear (R_SS_SC_WC)	可读；且 OTG 控制器自身可置位，可清 0；当软件往这位写 1 完成清 0 效果，往这位写 0 不会产生效果。OTG 具体置位与清 0 的条件在以下章节的各个域中有详细解释

13.3.2 全局寄存器

不管 OTG 工作在 host 模式下还是 Device 模式下都可访问这组寄存器，当 OTG 从一种工作模式转换到另一种工作模式时，无需重新例化这些寄存器。

13.3.2.1 控制与状态寄存器 (GOTGCTL)

偏移地址：000h

OTG 控制与状态寄存器 (GOTGCTL) 控制 OTG 的功能以及反映其状态。

域	说明	工作模式	复位值	访问特性
31:21	保留域	Host ; Device		RO

域	说明	工作模式	复位值	访问特性
20	OTG 版本 1'b0: 版本 1.3 ; 此版本 OTG 支持 Data line pulsing 与 VBus pulsing 的 SRP 1'b1: 版本 2.0 ; 此版本的 OTG 仅支持 Data line pulsing 的 SRP	Host ; Device	1'b0	RW
19	B-session valid(BSesVld) 指定 device 模式下 transceiver 的状态。 1'b0: B-session 非法 1'b1: B-session 合法。 在 OTG 模式下, 用这位来决定设备是否连接上	Device only	1'b1	RO
18	A-session valid(ASesVld) 指定 host 模式下 transceiver 的状态。 1'b0: A-session 非法 1'b1: A-session 合法。 在 device 模式下, 此位为保留位	Host only	1'b0	RO
17	Long/Short Debounce Time(DbncTime) 指定一个发现连接的 debounce 时间 1'b0: long debounce time , 用作物理连接 (100ms + 2.5us) 1'b1: short debounce time, 用作软连接 (2.5us)	Host only	1'b0	RO
16	连接器 ID 的状态 在一次连接事件中, 指示 ID 的状态 1'b0: OTG 工作在 A 设备模式下 1'b1: OTG 工作在 B 设备模式下	Host ; Device	1'b1	RO
15:12	保留域	Host ; Device		RO
11	设备 HNP 使能位 (DevHNPEn) 1'b0: HNP 未被使能 1'b1: HNP 使能	Device Only	1'b0	R_W
10	Host Set HNP 使能位 1'b0: Host Set HNP 未被使能 1'b1: Host Set HNP 使能	Host Only	1'b0	R_W
9	HNP 请求 软件将这位置位从而发起一个 HNP 请求给 host 。当 GOTGINT.HstNegSucStsChng 位置位时, 软件通过往这位写 0 来清 0 。当 HstNegSucStsChng 位为 0 , OTG 控制器将这位清 0。 1'b0: 无 HNP 请求 1'b1: HNP 请求	Device Only	1'b0	R_W
8	Host Negotiation Success(HstNegScs) 当 host 互换 (Host Negotiation) 成功后, OTG 控制器将这位置起。 1'b0: host 互换失败 1'b1: host 互换成功	Device only	1'b0	RO

域	说明	工作模式	复位值	访问特性
7	B-Peripheral Session Valid 覆盖值 (BvalidOvVal) 当 GOTGCTL.BvalidOvEn 置位时, BValidOvVal 位用来重新设置 Bvalid 信号.. 1'b0: 当 GOTGCTL.BvalidOvEn = 1 时, Bvalid 信号的值为 0 ; 1'b1: 当 GOTGCTL.BvalidOvEn = 1 时, Bvalid 信号的值为 1 ;	Device only	1'b0	R_W
6	B-Peripheral Session Valid 重置使能 (BvalidOvEn) 这位用来使能软件通过 GOTGCTL.BValidOvVal 位来覆盖 Bvalid 信号。 1'b1: 从 PHY 接收到的 Bvalid 信号值被 BvalidOv-Val 重新设置。 1'b0:Bvalid 信号重置功能关闭, OTG 控制器内部使用的 Bvalid 信号的值即位从 PHY 接收到的值	Device only	1'b0	R_W
5	A-Peripheral Session Valid 重置值 (AvalidOvVal) 当 GOTGCTL.AvalidOvEn 置位时, AValidOvVal 位用来重新设置 Avalid 信号 1'b0: 当 GOTGCTL.AvalidOvEn = 1 时, Avalid 信号的值为 0 ; 1'b1: 当 GOTGCTL.AvalidOvEn = 1 时, Avalid 信号的值为 1 ;	Host only	1'b0	R_W
4	A-Peripheral Session Valid 重置使能 (AvalidOvEn) 这位用来使能软件通过 GOTGCTL.AValidOvVal 位来 Override Avalid 信号。 1'b1: 从 PHY 接收到的 Avalid 信号值被 AvalidOv-Val 重新设置。 1'b0:Avalid 信号重置功能关闭, OTG 控制器内部使用的 Avalid 信号的值即位从 PHY 接收到的值	Host only	1'b0	R_W
3	VBUS Valid 重置值 (VbvalidOvVal) 当 GOTGCTL.VbvalidOvEn 置位时, VbvalidOvVal 位用来重新设置 vbus valid 信号 1'b0: 当 GOTGCTL.VbvalidOvEn = 1 时, Vbus valid 信号的值为 0 ; 1'b1: 当 GOTGCTL.VbvalidOvEn = 1 时, vbus valid 信号的值为 1 ;	Host only	1'b0	R_W
2	VBUS Valid 重置使能 (AvalidOvEn) 这位用来使能软件通过 GOTGCTL.VbValidOvVal 位来 Override vbus valid 信号。 1'b1: 从 PHY 接收到的 vbus valid 信号值被 VbvalidOvVal 重新设置。 1'b0:vbus valid 信号重置功能关闭, OTG 控制器内部使用的 vbus valid 信号的值即位从 PHY 接收到的值	Host only	1'b0	R_W

域	说明	工作模式	复位值	访问特性
1	会话请求 (Session Request, SesReq) 软件通过置位 SesReq 位在 USB 总线上发起一次会话请求。当 OTG 中断寄存器的 Host Negotiation Success Status Change 位置位时, 软件通过往这位写 0 来清 0。当 HstNegSucStsChng 位为 0, OTG 控制器将这位清 0。 1'b0: 无会话请求 1'b1: 请求会话	Device only	1'b0	R_W
0	会话请求成功 (Session Request Success, SesReqScs) 1'b0: 会话请求失败 1'b1: 会话请求成功	Device only	1'b0	R_W

13.3.2.2 中断寄存器 (GOTGINT)

偏移地址: 004h

域	说明	工作模式	复位值	访问特性
31:20	保留域	Host ; Device		RO
19	Debounce Done (DbnceDone) 当设备连接上的 debounce 完成后, OTG 控制器置起这位。当应用看到这个中断后, 可以开始 USB 复位操作。	Host only	1'b0	R_SS_WC
18	A 设备超时改变 (ADevTOUTChg) 当 A 设备等待 B 设备连接超时, OTG 控制器置起这位。	Host ; Device	1'b0	R_SS_WC
17	检测到角色互换 (HstNegDet) : 当在 USB 总线上发现角色互换请求 (Host Negotiation Req) 时, OTG 控制器置起这位。	Host ; Device	1'b0	R_SS_WC
16:10	保留域	Host and Device		RO
9	角色互换成功状态改变位 (HstNegSucStsChng) : Host Negotiation Success Status Change 当 USB 总线上的角色请求成功或失败时, OTG 控制器将这位置起。软件可通过查询 GOT-GCTL.HstNegSus 位来确定角色互换请求是否成功	Host and Device	1'b0	R_SS_WC
8	会话请求成功状态改变位 (SesReqSusStsChng) : Session Request Success Status Change 当 USB 总线上的会话请求成功或失败时, OTG 控制器将这位置起。软件可通过查询 GOT-GCTL.SesReqScs 位来确定会话请求是否成功	Host and Device	1'b0	R_SS_WC
7:3	保留域	Host and Device	1'b0	RO

域	说明	工作模式	复位值	访问特性
2	会话结束检测位 (SesEndDet) : Session End Detected 当 utmisrp_bvalid 信号拉低时, OTG 控制器置起这位	Device	1'b0	R_SS_WC
1:0	保留域	Host and Device		RO

13.3.2.3 AHB 配置寄存器 (GAHBCFG)

偏移地址: 008h

这个寄存器用来在上电或工作模式改变时配置 OTG 控制器, 配置的内容主要包括 AHB 相关的系统参数。在初始配置完成后, 不能再修改这个寄存器的值。不管是在 AHB 总线还是 USB 总线上开始一次事务 (transaction) 时, 都需要重新配置这个寄存器。

域	说明	工作模式	复位值	访问特性
31:23	保留域	Host ; Device		RO
22	通知所有的 DMA 写事务 (NotiAllDmaWrit): Notify All Dma Write Transactions 用来使能对所有的 DMA 写事务 (to endpoint/Channel) 的 DMA Done 功能。只有当 GAHBCFG.RemMemSupp 位为 1'b1 时, 这位才意义。 1'b1: 对于所有的 DMA 写事务, OTG 控制器都会在 AHB 总线上置起 (assert) dma_req, int_dma_done, chep_last_transact 以及 chep_number 信号。为了完成对某个 channel/endpoint 的传输, OTG 控制器需要等待所有写事务的 sys_dma_done 信号。 1'b0: 仅对写传输 (transfer) 的最后一个事务 (transaction), OTG 控制器才会置起 int_dma_done 信号。同样, 为了完成对某个 channel/endpoint 的传输, OTG 控制器仅需要等待那个 DMA 写事务的 sys_dma_done 信号。	Host; Device	1'b0	R_W
21	Remote Memory Support (RemMemSupp) 使能为 DMA 写传输等待系统 DMA DONE 信号功能 1'b1: 当 OTG DMA 开始对外部存储空间的传输时, 置起输出信号 int_dma_req。当完成这个传输, OTG 控制器置起 int_dma_done 信号用来指示从 OTG 控制器来的 DMA 写完成。然后 OTG 控制器等待从系统来 sys_dma_done 信号, 以便完成对某个 channel/endpoint 的数据传输。 无需置起 int_dma_req 与 int_dma_done 信号; 一旦 DMA 写传输完成, XferComp 中断之气, OTG 控制器继续工作。且无需等待 sys_dma_done 信号以完成数据传输。	Host and Device	1'b0	R_W

域	说明	工作模式	复位值	访问特性
8	<p>周期性 Tx FIFO Empty Level (PTxFIFOEmpLvl) : Periodic Tx FIFO Empty Level 指示周期性 Tx FIFO 空中断位 (GINTSTS.PTxFEMp) 何时置起。这位仅用在 slave 模式下。 1'b0: GINTSTS.PTxFEMp 中断位指示当周期性 Tx-FIFO 半空 1'b1: GINTSTS.PTxFEMp 中断位指示周期性 Tx-FIFO 全空</p>	Host only	1'b0	R_W
7	<p>非周期性 Tx FIFO Empty Level (NPTxFIFOEmpLvl) : Non-Periodic Tx FIFO Empty Level 这位仅用在 slave 模式下。 指示 IN 端点的 Tx FIFO 空中断位 (DIEPINTn.TxFEmp) 何时置起。 Host 模式: 1'b0: GINTSTS.PTxFEMp 中断位指示非周期性 Tx FIFO 半空; 1'b1: GINTSTS.PTxFEMp 中断位指示非周期性 Tx FIFO 全空; device 模式: 1'b0: DIEPINTn.TxFEmp 中断位指示 IN 端点的 Tx FIFO 半空; 1'b1: DIEPINTn.TxFEmp 中断位指示 IN 端点的 Tx FIFO 全空;</p>	Host and device	1'b0	R_W
6	保留域	Host ; Device		RO
5	<p>DMA 使能 (DMAEn) 1'b0: OTG 工作在 Slave 模式 1'b1: OTG 工作在 DMA 模式</p>	Host ; Device	1'b0	R_W
4:1	<p>Burst 长度 (HBstLen) AHB Master 的 burst 类型 4'b0000: Single 4'b0001: INCR 4'b0000: INCR4 4'b0000: INCR8 4'b0000: INCR16 其他: 保留</p>	Host; Device	4'b0	R_W
0	<p>全局中断屏蔽位 (GlbIntrMsk) 用来屏蔽对软件的中断。不管是否置起这位, OTG 控制器总是更新中断状态寄存器。 1'b0: 中断对软件不可见 1'b0 : 中断对软件可见</p>	Host ; Device	1'b0	R_W

13.3.2.4 USB 配置寄存器 (GUSBCFG)

偏移地址: 00Ch

这个寄存器用来在上电或工作模式改变时配置 OTG 控制器，配置的内容主要包括 USB 与 USB-PHY 相关的参数。不管是在 AHB 总线还是 USB 总线上开始一次事务 (transaction) 时，都需要配置这个寄存器。在初始配置完成后，不能再修改这个寄存器的值。

域	说明	工作模式	复位值	访问特性
31	Corrupt Tx package () 仅用来做调试使用，不可写 1.	Host; Device	1'b0	WO
30	强制 device 模式 (ForceDevMod) 写 1 意味着 OTG 控制器将工作在 device 模式下，而不管 utmiotg_iddig 输入管脚的值。 1'b0: 正常工作模式 1'b1: 强制 device 模式 将这位写 1 后，软件必须等待至少 25ms 的生效时间。 当在 scale down 模式下的仿真环境下，500us 已经足够。	Host ; Device	1'b0	R_W
29	强制 Host 模式 (ForceHostMod) 写 1 意味着 OTG 控制器将工作在 Host 模式下，而不管 utmiotg_iddig 输入管脚的值。 1'b0: 正常工作模式 1'b1: 强制 Host 模式 将这位写 1 后，软件必须等待至少 25ms 的生效时间。 当在 scale down 模式下的仿真环境下，500us 已经足够。	Host; Device	1'b0	R_W
28	Tx End 延时 (TxEndDelay) 写 1 使能 TxEndDelay 计时器。 1'b0: 正常模式 1'b1: 引入 Tx end 延时计时器	Device only	1'b0	R_W
27 : 23	保留	Host ; Device		RO
22	TermSel DLine Pulsing 选择 (TermSelDLPulse) 用来在 SRP 过程中，选择 utmi_termselect 驱动数据脉冲 (data line pulse) 。 1'b0: 用 utmi_txvalid 驱动数据脉冲 (默认) 1'b1: 用 utmi_termselect 驱动数据脉冲	Device	1'b0	R_W
21 : 16	保留	Host ; Device		RO
15	PHY 的低功耗时钟选择 (PhyLpwClkSel) 选择使用 480MHz 或 48Mhz(低功耗)PHY 模式。在 FS 与 LS 模式下，PHY 通常工作在 48MHz 下以节省功耗。 1'b0: 480MHz 内部 PLL 时钟 1'b1: 48MHz 外部时钟	Host; Device	1'b0	R_W
14	保留位	Host ; Device		RO

域	说明	工作模式	复位值	访问特性
13:10	USB 周期时间 (USBTrdTime): USB Turnaround Time 设置 PHY 时钟位单位的周期时间。 即指定 MAC 请求 PFC(数据 FIFO 控制器) 去数据 FIFO 中获取数据的响应时间。其值必须如下设置: 4'h5: 当 MAC 接口位 16 位的 UTMI+ 4'h6: 当 MAC 接口位 8 位的 UTMI+	Device	4'h5	R_W
9	HNP-Capable(HNPCap) 软件用此控制 OTG 控制器的 HNP 功能。 1'b0: HNP 功能不可用 1'b1: HNP 功能可用 注意, 如果 HNP 功能被软件禁掉, 那么 PHY 域相应的 OTG 信号必须固定为确定的值(?)。	Host, Device	1'b0	R_W
8	SRP-Capable(SRPCap) 软件用此控制 OTG 控制器的 SRP 功能。 1'b0: SRP 功能不可用 1'b1: SRP 功能可用 注意, 如果 SRP 功能被软件禁掉, 那么 PHY 域相应的 OTG 信号必须固定为确定的值	Host; Device	1'b0	R_W
7 : 4	保留位	Host ; Device		RO
3	PHY 接口 (PHYIf) 用来配置 UTMI+PHY 的接口。 1'b0: 8bits 1'b1: 16bits 若选择 ULPI 接口, 则仅能配置位 8bits 。	Host; Device	可配	RO
2:0	HS 超时校准 (TOutCal): HS Timeout Calibration 单位为一个 PHY 时钟。用来修正因为 PHY 引起的包之间超时时间。不同的 PHY 引入的延时不同。 对于 HS 而言, USB 标准的超时值为 736 - 816 bit times ; 软件必须基于枚举的速度来配置这个域。每个 PHY 时钟的 bits 时间如下解释: HS : 一个 30MHz 的 PHY 时钟= 16bit times 一个 60MHz 的 PHY 时钟= 8bit times	Host; Device	3'h0	R_W

13.3.2.5 复位寄存器 (GRSTCTL)

偏移地址: 010h

域	说明	工作模式	复位值	访问特性
31	.AHB Master Idle(AHBIdle) 指示 AHB Master 状态机处于 IDLE 状态	Host; Device	1'b1	RO
30	DMA 请求信号 (DMAReq) 指示现在总线上有个 DMA 请求; 调试使用	Host ; Device	1'b0	RO

域	说明	工作模式	复位值	访问特性
29:11	保留域	Host ; Device		RO
10:6	<p>TxFIFO 号 (TxFNum) 指示必须用 TxFIFO Flush 位清空的 FIFO 号。 直到控制器清空了 TxFIFO 位, TxFNum 才可修改。</p> <p>5'h0: - host 模式下清空非周期性 TxFIFO - device 模式下清空 TxFIFO 0</p> <p>5'h1: - host 模式下清空周期性 TxFIFO - device 模式下清空 TxFIFO 1</p> <p>5'h2: - device 模式下清空 TxFIFO 2</p> <p>5'h3: - device 模式下清空 TxFIFO 3 ...</p> <p>5'H10: - 清空所有的 TxFIFO (不管是 host 模式下还是 device 模式下)</p>	Host ; Device	5'h0	R_W
5	<p>TxFIFO 清空 (TxFFlsh) 用来选择性地清空一个或所有的 TxFIFO。但是有事务正在进行时不可进行此操作。 软件必须在确认 OTG 控制器没有读或写 TxFIFO 时才可写此位。确认方式如下: 读 - NAK 有效中断能确保 OTG 控制器没有读 FIFO ; 写 - GRSTCTL.AHBIdle 能确保 OTG 控制器没有写 FIFO 清空 FIFO 的操作可以在以下情形下使用, 比如 FIFO 需要重新配置或在共享 FIFO 模式与专用 FIFO 模式下转换以及禁用某个 device endpoint。 软件必须等待 OTG 控制器将此位清 0 后才能继续其他操作。这大概需要 8 个时钟, 且是使用 phy_clk 与 hclk 中比较慢的那个时钟。</p>	Host ; Device	1'b0	R_WS_SC
4	<p>RxFIFO 清空 (RxFFlsh) 用来清空整个 RxFIFO, 同样有事务正在进行时不可进行此操作。 软件必须在确认 OTG 控制器没有读或写 RxFIFO 时才可写此位。 软件必须等待 OTG 控制器将此位清 0 后才能继续其他操作。这大概需要 8 个时钟, 且是使用 phy_clk 与 hclk 中比较慢的那个时钟。</p>	Host ; Device	1'b0	R_WS_SC
3	保留位	Host ; Device		RO

域	说明	工作模式	复位值	访问特性
2	Host Frame Counter Reset (FrmCntrRst) 软件通过写此位来重置 OTG 控制器内的帧数。一旦帧数被重置, 下一个 OTG 控制器发出的 SOF 的帧号为 0.	Host	1'b0	R_WS_SC
1	保留域	Host; Device		RO
0	OTG 控制器软重启 (CSftRst) 复位 hclk 与 phy_clock 时钟域的内容如下: 清空除以下列以外的所有中断以及 CSR 寄存器 - PCGCCTL.RstPdown - PCGCCTL.GateHclk - PCGCCTL.PwrClmp - GUSBCFG.DDRSel - GUSBCFG.PHYSel - GUSBCFG.FSIntf - GUSBCFG.ULPLUTMLSel - GUSBCFG.PHYIf - HCFG _j FSLSPclkSel - DCFG.DevSpd - GGPIIO - GPWRDN - GADPCTL 所有模块的状态机(除 AHB Slave 单元)都被复位至 IDLE 状态; 清空所有的 TxFIFO 与 RxFIFO 在平滑地完整最后一个 AHB 传输的数据相后, 立即结束所有的 AHB Master 事务。立即结束所有的 USB 事务 PMU 模块(功耗管理单元)不会被复位 软件在任意时刻都可软复位 OTG 控制器。这位是自清 0 的, 即在所有必要的逻辑复位后控制器会清 0 此位。根据控制器当时的状态不同, 这将花费数个时钟。一旦此位被清 0, 软件必须等待 3 个 PHY 时钟后才能操作 PHY 域(同步延时)。软件在开始任何动作之前, 必须确保 AHBIidle 位位 1。	Host ; Device	1'b0	R_WS_SC

13.3.2.6 中断寄存器 (GINTSTS)

偏移地址: 014h

域	说明	工作模式	复位值	访问特性
31	系统恢复 / 远程唤醒中断 (WkUpInt):Resume/Remote Wakeup Detected Interrupt 处于挂起 (L2) 状态时的唤醒中断。 - Device 模式: 仅当在 USB 总线上发现 Host 发起的唤醒操作时, 发生此中断 - Host 模式: 仅当在 USB 总线上发现 device 发起的远程唤醒操作时, 发生此中断	Host; Device	1'b0	R_SS_WC
30	会话请求 / 发现新会话中断 (SessReqInt) :Session Request/New Session Detected Interrupt. - 在 Host 模式下, 当发现从 device 来的会话请求时, 置 1. - 在 device 模式下, 当 utmisrp_bvalid 信号拉高时, 置 1	Host; Device	1'b0	R_SS_WC
29	断开中断 (DisconnInt) 当 device 断开时, 置 1.	Host	1'b0	R_SS_WC
28	连接器 ID 状态变化 (ConIDStsChng) 当连接器 ID 状态有变化时, 置 1.	Host ; Device	1'b1	R_SS_WC
27	保留位	Host ; Device		RO
26	周期性 TxFIFO 空 (PTxFEmp) 当周期性 TxFIFO 是半空或全空, 且在周期性请求队列中至少有一项请求时, 此位置 1. 是半空还是全空时置 1 由 GAHBCFG.PTxFEmpLvl 来决定。	Host	1'b1	RO
25	Host Channels Interrupt (HChInt) 在 host 模式下, 当 OTG 控制器里某一个通道上有一个等待处理的中断时, 置 1. 软件必须读 HAINTE 寄存器来确定有未处理中断的 channel 号, 然后读取对应的 HCINTn 寄存器来确定发生中断的原因。为了将这位清 0, 软件必须先将 HCINTn 寄存器中对应状态位清 0.	Host	1'b0	RO
24	Host 端口中断 (PrtInt) 在 Host 模式下, 若 OTG 控制器的某个端口状态发生变化时, 置 1. 软件必须读 HPRT 寄存器来确定引起这个中断的真正原因。为了将这位清 0, 软件必须先将 HPRT 寄存器中对应状态位清 0.	Host	1'b0	RO
23	复位中断 (ResetDet) : Reset Detected Interrupt 在 device 模式下, 处于挂起状态下的 OTG 控制器发下一复位操作时, 置 1. 仅当 OTG 控制器处于 device 模式, 且工作在 partial Power-down, 或是门控挂起状态下才效。 当 OTG 控制器工作在 Hibernation 挂起状态时, 这位无效。	Device	1'b0	R_SS_WC
22	数据获取挂起 (FetSusp) : Data Fetch Suspended 仅在 DMA 模式下有效。当 TxFIFO 或请求队列没有空间时 OTG 会停止为 IN endpoint 取数据, 这时置 1. 软件在 endpoint 不匹配算法中使用此中断。	Device	1'b0	R_SS_WC

域	说明	工作模式	复位值	访问特性
21	未完成周期性传输 (incomplIP) Host 模式下, 当有一未完成的周期性事务仍等待在当前帧时间内处理时, 置 1. 未完成的实时 OUT 传输 (incomplSOOUT) device 模式下, 若在当前帧时间内, 至少有一个实时 OUT endpoint 上有一个未完成的传输时, 置 1. 同时置 EOPF 位	Host Device	1'b0	R_SS_WC
20	未完成的实时 IN 传输 (incomplSOIN) 若在当前帧时间内, 至少有一个实时 IN endpoint 上有一个未完成的传输时, 置 1. 同时置 EOPF 位	Device	1'b0	R_SS_WC
19	OUT endpoint 中断 (OEPInt) 当在 OUT endpoint 上有一等待处理的中断时, OTG 控制器置 1. 软件必须读 DAINt 寄存器从而确定具体是哪个 endpoint 上有等待处理的中断; 然后读 DOEPINTn 寄存器从而确定引起这个中断的原因. 为了将这位清 0, 软件必须先将 DOEPINTn 寄存器中对应状态位清 0.	Device	1'b0	RO
18	IN endpoint 中断 (IEPInt) 当在 IN endpoint 上有一等待处理的中断时, OTG 控制器置 1. 软件必须读 DAINt 寄存器从而确定具体是哪个 endpoint 上有等待处理的中断; 然后读 DOEPINTn 寄存器从而确定引起这个中断的原因. 为了将这位清 0, 软件必须先将 DOEPINTn 寄存器中对应状态位清 0.	Device	1'b0	RO
17	保留位	Host ; Device		RO
16	保留位	Host ; Device		RO
15	周期性帧结束 (EOPF) 当在当前帧时间内, DCFG.PerFrInt 指示的阈值达到时, 置 1.	Device	1'b0	R_SS_WC
14	实时的 OUT 包结束 (ISOOutDrop) 若 RxFIFO 没有足够的空间来为实时 OUT endpoint 存放一 maximum package size packet, 则 OTG 控制器将无法将一个实时 OUT 包写入到 RxFIFO. 当发生这种情况时, 置此位为 1.	Device	1'b0	R_SS_WC
13	枚举结束 (EnumDone): Enumeration 当速度枚举结束是, 置 1. 软件读取 DSTS 寄存器获取枚举的确切速度	Device	1'b0	R_SS_WC
12	USB 复位 (USB Rst) 当 USB 总线上有一复位操作时, 置 1	Device	1'b0	R_SS_WC
11	USB 挂起 (USBSusp) 当 OTG 控制器检测到 USB 总线上有一挂起操作时, 置 1. 若在一段较长的时间内 utmi.linestate 信号线上没有活动, 那么 OTG 控制器将进去挂起状态	Device	1'b0	R_SS_WC

域	说明	工作模式	复位值	访问特性
10	Early Suspend(ErlySusp) 当检测到 USB 已进入空闲状态超过 3ms 时，OTG 控制器置此位为 1.	Device	1'b0	R_SS_WC
9:8	保留位	Host ; Device		RO
7	全局 OUT NAK 有效 (GOUTNakEff) 软件置 DCTL.SGOUTNak 位后其开始有效时，置 1. 通过清 0 DCTL.SGOUTNak 位达到清 0 此位的效果	Device	1'b0	RO
6	全局 IN 非周期性 NAK 有效 (GINNakEff) 软件置 DCTL.SGNPInNak 位后其开始有效时，置 1. 通过清 0 DCTL. SGNPInNak 位达到清 0 此位的效果	Device	1'b0	RO
5	保留位	Host ; Device		RO
4	RxFIFO 非空 (RxFLvl) 当 RxFIFO 中至少有一个包时，置 1.	Host ; Device1	1'b0	RO
3	帧开始 (Sof) Host 模式下，当 SOF 在 USB 总线上传输时，OTG 控制器将此位置 1. 软件写 1 将此位清 0. Device 模式下，当 OTG 控制器收到 SOF 标志 (token) 时，置 1. 软件可通过读 Device status 寄存器来获取当前帧的帧号。	Host; Device	1'b0	R_SS_WC
2	OTG 中断 (OTGInt) 置 1 指示发生了 OTG 协议事件。软件可读 GOTGINT 寄存器来确定发生中断的原因。为了将这位清 0，软件必须清空 GOTGINT 寄存器相应的状态位。	Host; Device	1'b0	RO
1	模式不匹配中断 (ModeMis) 当软件试图进行一下操作时，置 1. 当 OTG 工作在 device 模式下时，试图访问 host 模式下的寄存器。 当 OTG 工作在 Host 模式下时，试图访问 device 模式下的寄存器。	Host ; Device	1'b0	R_SS_WC
0	当前工作模式 (CurMod) 指示当前 OTG 控制器的工作模式 1'b0: device 模式 1'b1: host 模式	Host ; Device	1'b0	RO

13.3.2.7 中断屏蔽寄存器 (GINTMSK)

偏移地址: 018h

这个寄存器与 GINTSTS 寄存器共同使用。当某个中断被屏蔽，那么这个中断不会被软件看到。当然，GINTSTS 对应的位仍旧置位。

- 1: 中断使能
- 0: 中断屏蔽

域	说明	工作模式	复位值	访问特性
31	系统恢复 / 远程唤醒中断屏蔽位 (WkUpIntMsk):Resume/Remote Wakeup Detected Interrupt Mask	Host; Device	1'b0	R_SS_WC
30	会话请求 / 发现新会话中断屏蔽位 (SessReqIntMsk):Session Request/New Session Detected Interrupt Mask	Host; Device	1'b0	R_SS_WC
29	断开中断屏蔽位 (DisconnIntMsk)	Host	1'b0	R_SS_WC
28	连接器 ID 状态变化屏蔽位 (ConIDStsChngMsk)	Host ; Device	1'b1	R_SS_WC
27	保留位	Host ; Device		RO
26	周期性 Tx FIFO 空屏蔽位 (PTxFEmpMsk)	Host	1'b1	RO
25	Host Channels Interrupt 屏蔽位 (HChIntMsk)	Host	1'b0	RO
24	Host 端口中断屏蔽位 (PrtIntMsk)	Host	1'b0	RO
23	复位中断屏蔽位 (ResetDetMsk) : Reset Detected Interrupt Mask	Device	1'b0	R_SS_WC
22	数据获取挂起屏蔽位 (FetSuspMsk) : Data Fetch Suspended Mask	Device	1'b0	R_SS_WC
21	未完成周期性传输屏蔽位 (incomplIPMsk) 未完成的实时 OUT 传输屏蔽位 (incomplSOOUTMsk)	Host Device	1'b0	R_SS_WC
20	未完成的实时 IN 传输屏蔽位 (incomplSOINMsk)	Device	1'b0	R_SS_WC
19	OUT endpoint 中断屏蔽位 (OEPIntMsk)	Device	1'b0	RO
18	IN endpoint 中断屏蔽位 (IEPIntMsk)	Device	1'b0	RO
17:16	保留位	Host ; Device		RO
15	周期性帧结束屏蔽位 (EOPFMsk)	Device	1'b0	R_SS_WC
14	实时的 OUT 包结束屏蔽位 (ISOOutDropMsk)	Device	1'b0	R_SS_WC
13	枚举结束屏蔽位 (EnumDoneMsk) : Enumeration Mask	Device	1'b0	R_SS_WC
12	USB 复位屏蔽位 (USBRstMsk)	Device	1'b0	R_SS_WC
11	USB 挂起屏蔽位 (USBSuspMsk)	Device	1'b0	R_SS_WC
10	Early Suspend 屏蔽位 (ErlySuspMsk)	Device	1'b0	R_SS_WC
9:8	保留位	Host ; Device		RO
7	全局 OUT NAK 有效屏蔽位 (GOUTNakEffMsk)	Device	1'b0	RO
6	全局 IN 非周期性 NAK 有效屏蔽位 (GIN-NakEffMsk)	Device	1'b0	RO
5	保留位	Host ; Device		RO
4	RxFIFO 非空屏蔽位 (RxFLvlMsk)	Host ; Device1	1'b0	RO
3	帧开始屏蔽位 (SofMsk)	Host; Device	1'b0	R_SS_WC
2	OTG 中断屏蔽位 (OTGIntMsk)	Host; Device	1'b0	RO

域	说明	工作模式	复位值	访问特性
1	模式不匹配中断屏蔽位 (ModeMisMsk)	Host ; Device	1'b0	R_SS_WC
0	当前工作模式 (CurModMsk)	Host ; Device	1'b0	RO

13.3.2.8 接收状态调试寄存器 / 状态 read 以及 pop 寄存器 (GRXSTSR/-GRXSTSP)

读偏移地址：01Ch

Pop 偏移地址：020h

对接收状态调试寄存器的读操作返回的是 RxFIFO 的第一项内容。对状态 read 以及 pop 寄存器的读操作额外还会将 RxFIFO 的第一项内容 pop 出来。接收状态寄存器的内容在 host 模式与 device 模式下有着不同的解释。若 RxFIFO 空，那么 OTG 控制器将忽略对这个寄存器的读以及 pop 操作，且返回 32'h0000_0000。只有当 GINTSTS.RxFLvl 位为 1 时，软件才可以进行 pop 操作。NOTE：OTG 在不同的工作模式下，此寄存器的各个域有着不同的解释

Host 模式

域	说明	复位值	访问特性
31:21	保留域	1	RO
20:17	包状态 (PktSts) 指示接收包的状态： 4'b0010: 接收到 IN 数据包 4'b0011: IN 传输结束 (触发中断) 4'b0101: 数据 toggle 错误 (触发中断) 4'b0111: 通道 halt (触发中断) 其他：保留	4'b0	RO
16:15	数据 PID (DPID) 指示接收包的数据 PID 2'b00:DATA0 2'b10:DATA1 2'b01:DATA2 2'b11:MDATA	2'b0	RO
14:4	字节数 (BCnt) 指示接收到的 IN 数据包的字节数	11'h0	RO
3:0	通道号 (ChNum) 指示当前接收到的数据包所属的通道号	4'h0	RO

Device 模式

域	说明	复位值	访问特性
31:25	保留域		RO

域	说明	复位值	访问特性
24:21	帧号 (FN) 接收包所在帧的帧号最低四位。 仅当支持实时 OUT 端点时, 此域有效	4'h0	RO
20:17	包状态 (PktSts) 指示接收包的状态: 4'b0001: 全局 OUT NAK (触发中断) 4'b0010: 接收到 OUT 数据包 4'b0011: OUT 传输结束 (触发中断) 4'b0100: SETUP 事务结束 (触发中断) 4'b0110: 接收到 SETUP 数据包 其他: 保留	4'b0	RO
16:15	数据 PID (DPID) 指示接收到的 OUT 包的数据 PID 2'b00:DATA0 2'b10:DATA1 2'b01:DATA2 2'b11:MDATA	2'b0	RO
14:4	字节数 (BCnt) 指示接收到的 IN 数据包的字节数	11'h0	RO
3:0	端点号 (EPNum) 指示当前接收到的数据包所属的端点号	4'h0	RO

13.3.2.9 RxFIFO 容量寄存器 (GRXFSIZ)

读偏移地址: 024h

软件用此寄存器分配分配给 RxFIFO 的 RAM 大小。

域	说明	复位值	访问特性
31:16	保留域		RO
15:0	RxFIFO 深度 (RxFDep) 以 32-bits 的字为单位。 - 最小值为 16 - 最大值为 32768	533	R_W

13.3.2.10 非周期性 Tx FIFO 容量寄存器 (GNPTXFSIZ)

读偏移地址: 028h

软件用此寄存器分配分配给非周期性 Tx FIFO 的 RAM 大小以及开始地址。

Note: OTG 在 host 工作模式下, 还是在 device 工作模式下, 此寄存器有不同的解释。

Host 模式下

域	说明	复位值	访问特性
31:16	非周期性 TxFIFO 深度 (NPTxDep) 以 32-bits 的字为单位。 - 最小值为 16 - 最大值为 32768	256	R_W
15:0	非周期性 TxRAM 开始地址 (NPTxStAddr)	533	R_W

Device 模式下

域	说明	复位值	访问特性
31:16	IN 端点 TxFIFO 0 的深度。仅当 Device 模式下，且 OTG_EN_DED_TX_FIFO = 1 时有效。 - 最小值为 16 - 最大值为 32768	32	R_W
15:0	IN 端点 FIFO0 TxRAM 开始地址	533	R_W

13.3.2.11 Synopsys ID 寄存器 (GSNPSID)

读偏移地址：040h

包含 OTG 控制器的版本号

域	说明	复位值	访问特性
31: 0	Synopsys ID (SynopsysID) OTG 控制器的版本号	32'h4F54	RO

13.3.2.12 User HW Config1 寄存器

读偏移地址：044h

用户硬件配置寄存器 2, 指定 endpoint 的方向。

域	说明	复位值	访问特性
31: 14	保留域	18'b0	RO
13: 0	Bits[13:12]: endpoint 6 方向 Bits[11:10]: endpoint 5 方向 Bits[9:8]: endpoint 4 方向 Bits[7:6]: endpoint 3 方向 Bits[5:4]: endpoint 2 方向 Bits[3:2]: endpoint 1 方向 Bits[1:0]: endpoint 0 方向 2'b00: 双向 endpoint 2'b01: IN endpoint 2'b10: OUT endpoint 2'b11: 保留	14'b0	RO

13.3.2.13 User HW Config2 寄存器

读偏移地址：048h

用户硬件配置寄存器 2

域	说明	复位值	访问特性
31 : 26	保留域	6'b0	RO
25: 24	Host 模式下, 周期性请求队列的深度 2'b00: 2 2'b01: 4 2'b10: 8 2'b11: 16	2'b10	RO
23: 22	非周期性请求队列的深度 2'b00: 2 2'b01: 4 2'b10: 8 2'b11: 16	2'b10	RO
21	保留域		RO
20	使能多个处理器中断 1'b0: 否 1'b1: 是	1'b0	RO
19	使能动态 FIFO 大小 1'b0: 否 1'b1: 是	1'b1	RO
18	使能周期性 OUT channel 1'b0: 否 1'b1: 是	1'b1	RO
17:14	Host 模式下的 Channel 数	4'd12	RO
13:10	Device 模式下的端点数, 除 endpoint 0 外	4'd6	RO
9: 8	全速 PHY 接口类型 2'b00: 无 2'b01: FS 有专用 pin 2'b10: 与 UTMI+ 共用 2'b11: 与 ULPI 共用	2'b00	RO
7:6	HS PHY 接口类型 2'b00: 无 2'b01: UTMI+ 2'b10: ULPI 2'b11: UTMI+ 与 ULPI	2'b01	RO
5	点对点 1'b0: 否 1'b1: 是	1'b1	RO
4 : 3	结构 2'b00: 仅能做从设备 2'b01: 外部 DMA 2'b10: 内嵌 DMA 2'b11: 保留	2'b10	RO

域	说明	复位值	访问特性
2:0	操作模式 3'b000: 支持 HNP 与 SRP 的 OTG (host 与 device) 3'b001: 仅支持 SRP 的 OTG (host 与 device) 3'b010: 不支持 HNP 与 SRP 的 OTG (host 与 device) 3'b011: 支持 SRP 的 device 3'b100: 非 OTG 的 device 3'b101: 支持 SRP 的 device 3'b110: 非 OTG 的 host 其他: 保留	3'b000	RO

13.3.2.14 User HW Config3 寄存器

读偏移地址: 04Ch

用户硬件配置寄存器 3

域	说明	复位值	访问特性
31 : 16	DFIFO 的深度 - 最小值: 32 - 最大值: 32768	16'd3072	RO
15	OTG_ENABLE_LPM 指定 OTG 是否支持 LPM 模式 1'b0: 否 1'b1: 是	1'b0	RO
14	OTG_BC_SUPPOT 指定是否支持 USB 充电功能 1'b0: 否 1'b1: 是	1'b0	RO
13	OTG_ENABLE_HSIC 指定是否支持 HSIC 1'b0: 否 1'b1: 是	1'b0	RO
12	OTG_ADP_SUPPORT 指定是否支持 ADP 1'b0: 否 1'b1: 是	1'b0	RO
11	OTG_SYNC_RESET_TYPE 指定是否支持同步复位 1'b0: 否 1'b1: 是	1'b0	RO
10	去掉一些非必须的特性 (OptFeature) 1'b0: 否 1'b1: 是	1'b1	RO
9	是否支持 vendor control 接口 (VndctlSupt) 此接口用于访问 PHY 的内部寄存器。 1'b0: 否 1'b1: 是	1'b0	RO

域	说明	复位值	访问特性
8	是否支持 I2C 接口 (I2CIntSel) 1'b0: 否 1'b1: 是	1'b0	RO
7	OTG 功能使能 (OtgEn) 指定是否使能 OTG 功能 1'b0: 否 1'b1: 是	1'b1	RO
6: 4	Packet Counters 的宽度 (PktSizeWidth) 表示在一次传输中 USB 控制器 send/receive 的 Packet 的最大个数. 3'b000: 4bits 3'b001: 5bits 3'b010: 6bits 3'b011: 7bits 3'b100: 8bits 3'b101: 9bits 3'b110: 10bits 其他: 保留	3'b110	RO
3: 0	Transfer Size Counters 的宽度 (XferSizeWidth), 表示最大传输数据大小 4'b0000: 11bits 4'b0001: 12bits ... 4'b1000: 19bits 其他: 保留	4'b1000	RO

13.3.2.15 User HW Config4 寄存器

读偏移地址: 050h

用户硬件配置寄存器 4

域	说明	复位值	访问特性
31	保留		RO
30	是否支持 Scatter/Gather DMA 1'b0: 否 1'b1: 是	1'b1	RO
29:26	Device 模式下, IN endpoint 的个数, 包括 endpoint0 (INEps) 0: 1 个 IN endpoint 1: 2 个 IN endpoint ... 16: 16 个 IN endpoint	4	RO
25	对 Device In endpoint 使能专用的 Tx FIFO 1'b0: 否 1'b1: 是	1'b1	RO
24	是否加一个对 session_end 信号的去抖逻辑 1'b0: 否 1'b1: 是	1'b1	RO

域	说明	复位值	访问特性
23	是否加一个对 b_valid 信号的去抖逻辑 1'b0: 否 1'b1: 是	1'b1	RO
22	是否加一个对 a_valid 信号的去抖逻辑 1'b0: 否 1'b1: 是	1'b1	RO
21	是否加一个对 vbus_valid 信号的去抖逻辑 1'b0: 否 1'b1: 是	1'b1	RO
20	是否加一个对 iddig_valid 信号的去抖逻辑 1'b0: 否 1'b1: 是	1'b1	RO
19:16	除去 endpoint0, device 模式下控制 endpoint 的个数	4'b0	RO
15:14	UTMI+ PHY 数据宽度 2'b00: 8bits 2'b01: 16bits 2'b10: 8/16 bits 软件可选 2'b11: 保留	2'b10	RO
13:7	保留域		RO
6	使能 Hibernation 1'b0: 否 1'b1: 是	1'b0	RO
5	AHB 频率是否最小不能低于 60MHA(AhbFreq) 1'b0: 否 1'b1: 是	1'b0	RO
4	是否使能降低部分功耗 1'b0: 否 1'b1: 是	1'b1	RO
3:0	保留域		RO

13.3.2.16 DFIFO 软件配置寄存器 (GDFIFOCFG)

偏移地址: 05Ch

域	说明	工作模式	复位值	访问特性
31:16	EPInfoBaseAddr Endpoint 控制器的起始地址	Host ; Device	EPINFO_BASEADDR	R_W
15:0	GDFIFOCfg 动态配置 DFIFO 的大小	Host ; Device	3072	R_W

13.3.2.17 Host 周期性 Tx FIFO 容量寄存器 (HPTXFSIZ)

读偏移地址: 0100h

域	说明	复位值	访问特性
31:16	PTxFSize 周期性 Tx FIFO 深度, 以 32-bits 的字位单位。 - 最小值: 16 - 最大值: 32768	512	R_W
15:0	PTxFStAddr Host 模式下周期性 Tx FIFO 的起始地址	789	R_W

13.3.2.18 Device 模式下 IN 端点 Tx FIFO2 容量寄存器 (DIEPTXF1)

读偏移地址: 0104h

域	说明	复位值	访问特性
31:16	INEPnTxFDep IN 端点 Tx FIFO1 深度, 以 32-bits 的字位单位。 - 最小值: 16 - 最大值: 32768	256	R_W
15:0	INEPnTxFStAddr IN 端点 FIFO 1 开始地址	565	R_W

13.3.2.19 Device 模式下 IN 端点 Tx FIFO2 容量寄存器 (DIEPTXF1)

读偏移地址: 0108h

域	说明	复位值	访问特性
31:16	INEPnTxFDep IN 端点 Tx FIFO2 深度, 以 32-bits 的字位单位。 - 最小值: 16 - 最大值: 32768	256	R_W
15:0	INEPnTxFStAddr IN 端点 FIFO 2 开始地址	821	R_W

13.3.2.20 Device 模式下 IN 端点 Tx FIFO3 容量寄存器 (DIEPTXF1)

读偏移地址: 010Ch

域	说明	复位值	访问特性
31:16	INEPnTxFDep IN 端点 Tx FIFO3 深度, 以 32-bits 的字位单位。 - 最小值: 16 - 最大值: 32768	512	R_W
15:0	INEPnTxFStAddr IN 端点 FIFO 3 开始地址	1077	R_W

13.3.3 Host 模式寄存器

这组寄存器仅在 Host 模式下有效; 在 Device 模式下不可访问这组寄存器。

13.3.3.1 Host 模式配置寄存器 (HCFG)

读偏移地址：400h

域	说明	复位值	访问特性
31:28	保留域		RO
27	ModeChTimEn 是否使能模式转换准备计时器。 用来决定是否使能 host 控制器在 Resume 状态结束时，等待 200PHY 时钟以改变对 PHY 的 opmode 信号从 2'b10 到 00 1'b0: 否 1'b1: 是	1'b0	R_W
26	PerSchedEna 是否使能周期调度。 1'b0: 否 1'b1: 是	1'b0	R_W
25:24	FrListEn Frame list 项数 2'b00: 8 2'b01: 16 2'b00: 32 2'b00: 64	2'b00	R_W
23	DescDMA Host 模式下，使能 Scatter/gather DMA 注意，在复位之后，这位仅能背修改一次。以下配置是可用的： GAHBCFG.DMAEn = 0，HCFG.DescDMA = 0 = _i slave 模式； GAHBCFG.DMAEn = 0，HCFG.DescDMA = 1 = _i 非法配置； GAHBCFG.DMAEn = 1，HCFG.DescDMA = 0 = _i Buffered DMA 模式； GAHBCFG.DMAEn = 0，HCFG.DescDMA = 1 = _i Scatter/-Gather 模式；	1'b0	RO
15:0	保留域		RO

13.3.3.2 Host 帧间隔寄存器 (HFIR)

读偏移地址：404h

域	说明	复位值	访问特性
31:17	保留域		RO
16	Reload 控制 是否允许动态重新载入 HFIR 寄存器。 1'b0: 否 1'b1: 是	1'b0	R_W

域	说明	复位值	访问特性
15:0	帧间隔 (FrInt) 软件用这个值来制定两个连续 micro-SOFs 之间的间隔。以 PHY 时钟位单位。仅当 HPRT.PrtEnaPort 置 1 后才能更改此位。在最初配置后, 不能再更改。 125us * (PHY 时钟频率)	16'd60000	R_W

13.3.3.3 Host 帧号 / 剩余帧时间寄存器 (HFNUM)

读偏移地址: 408h

域	说明	复位值	访问特性
31:16	剩余帧时间 指示当前 micorFrame 的剩余时间, 以 PHY 时钟为单位。当为 0 时, 重新载入 HFIR.FrInt 的值, 并在 USB 总线上发送一个新的 SOF	16'h0	RO
15:0	帧号 (FrNum) 当一个新的 SOF 传输时, 此域加 1, 当达到 16'h3FFF 时复位为 0.	16'h3FFF	RO

13.3.3.4 Host 周期性 Tx FIFO/Queue 状态寄存器 (HPTXSTS)

读偏移地址: 410h

域	说明	复位值	访问特性
31:24	HPTXSTS 周期性 Tx FIFO 请求队列的第一项 (TOP) .用作调试。 Bits[31]: 奇偶帧 - 1'b0: 在偶帧发送 - 1'b1: 在奇帧发送 Bits[30:27]: channel 号 Bits[26:25]: 类型 - 2'b00: IN/OUT - 2'b01:0 长度包 - 2'b10:CSPLIT - 2'b11 : 无效 channel 命令 Bits[24]: 终止 (对选定的 channel 来讲为最后一项)	8'h0	RO
23:16	PTxQSpAvali 周期性 Tx 请求队列可用项数. 指示周期性请求队列中的空项数, 请求队列中包含 IN 与 OUT 8'b0: 队列满 8'b1: 1 个空项 8'b2: 2 个空项 ... 8 : 8 个空项 其他: 保留	8	RO

域	说明	复位值	访问特性
15:0	PTxFSpCAvail 周期性 TxFIFO 可用项数。 以 32-bits 的字为单位。 16'h0: FIFO 满 16'h1: 1 个空项 ... 16'h200: 512 个空项	512	RO

13.3.3.5 Host 所有 channel 中断寄存器 (HAINT)

读偏移地址：414h

域	说明	复位值	访问特性
31:16	保留域		RO
15:0	Channel 中断 (HAINT) 一个 bit 对应一个 channel : Bits 11 : channel 11 ... Bits0 : channel0	16'h0	RO

13.3.3.6 Host 所有 channel 中断屏蔽寄存器 (HAINTMSK)

读偏移地址：414h

域	说明	复位值	访问特性
31:16	保留域		RO
15:0	Channel 中断屏蔽位 (HAINTMSK) 一个 bit 对应一个 channel : Bits 11 : channel 11 ... Bits0 : channel0	16'h0	R_W

13.3.3.7 Host 端口控制与状态寄存器 (HPRT)

读偏移地址：440h

域	说明	复位值	访问特性
31:19	保留域		RO
18:17	PrtSpd 端口速度 2'b00: 高速 2'b01: 全速 2'b10: 低速 2'b11: 保留	2'b0	RO

域	说明	复位值	访问特性
16:13	PrtTstCtl 端口测试控制 当软件往此域写非 0 值时, 使此端口进入测试模式。相应的 Pattern 出现端口上。 4'b0000: 非测试模式 4'b0001: Test_J 模式 4'b0010: Test_K 模式 4'b0011: Test_SE0_NAK 模式 4'b0100: Test_Packet 模式 4'b0101: Test_Force 模式 其他: 保留	4'h0	R_W
12	PrtPwr 端口供电。 1'b0: power off 1'b1: power on	1'b0	R_W_SC
11:10	Port Line Status(PrtLnSts) Bits[10]: D+ Bits[11]: D-	1'b0	RO
9	保留		RO
8	端口复位 (PrtRst)	1'b0	R_W
7	端口挂起 (PrtSup)	1'b0	R_WS_SC
6	端口恢复 (PrtRes)	1'b0	R_W_SS_SC
5	端口过载改变 (PrtOvrCurrChng) 当 PrtOvrCurrAct 位改变时, OTG 控制器将这位置 1	1'b0	R_SS_WC
4	端口过载 (PrtOvrCurrAct) 1'b0: 无 1'b1: 有	1'b0	RO
3	端口使能状态改变 (PrtEnChng) 当 PrtEna 位改变时, OTG 控制器将这位置 1	1'b0	R_SS_WC
2	端口使能位 (PrtEna) 1'b0: 否 1'b1: 是	1'b0	R_SS_SC_WC
1	发现端口连接 (PrtConnDet) 当发现设备连接时, 将通过 GINTSTS.PrtInt 触发中断。软件写 1 将此位清 0。	1'b0	R_SS_WC
0	端口连接状态 (PrtConnSts) 无设备连接到此端口 有设备连接到此端口	1'b0	RO

13.3.3.8 Host Channel-n 特性寄存器 (HCCHARn)

Channel_num: $0 \leq n \leq 11$

读偏移地址: $500h + (\text{Channel_num} * 20h)$

域	说明	复位值	访问特性
31	Channel 使能 (ChEna) 当 Scatter/Gather 模式使能时 1'b0: 指示描述符结构还未准备好 1'b1: 指示描述符结构以及数据缓冲已经准备好, 这个通道可以访问描述符 当 Scatter/Gather 模式未使能时: 1'b0: 通道失效 1'b1: 通道使能	1'b0	R_WS_SC
30	通道失效 (ChDis) 即使此通道上的传输还未完成, 软件也可通过将这位置 1 来停止在此通道上的数据发送与接收。软件必须等待通道失效中断发生后, 才能认为此通道已失效。	1'b0	R_WS_SC_SS
29	奇帧 (OddFrm) 软件通过将此位置 1 来告知 OTG host 必须在奇数 micro Frame 内完成一个传输。此域仅对周期性事务 (实时与中断) 有效 1'b0: 偶帧 1'b1: 奇帧	1'b0	R_W
28:22	设备地址 (DevAddr)	7'h0	R_W
21:20	Multi Count (MC) /Error Count(EC) 当 HCSPLTn.SpltEna 为 1'b0 时, 此域指示对于每个周期性的 endpoint, 在每个 microFrame 时间内完成的事务数。对于非周期性传输, 此域尽在 DMA 模式下使用, 用来指定在 DMA 改变仲裁之前, host 必须为此通道获取的包数。 2'b00: 保留 2'b01: 1 个事务 2'b10: 2 个事务 2'b11: 3 个事务 当 CSPLTn.SpltEna 为 1'b1 时, 此域指定对于一个周期性 split 事务出错时, 必须重传的次數。至少设置为 2'b01	2'b0	R_W
19:18	端点类型 (EPType) 2'b00: 控制 2'b01: 实时 2'b10: 批量 2'b11: 中断	2'b0	R_W
17:16	保留		RO
15	端点方向 (EPDir) 1'b0: OUT 1'b1: IN	1'b0	R_W
14:11	端点号	4'b0	R_W
10:0	最大的包容量 (MPS)	11'h0	R_W

13.3.3.9 Host Channel-n Split 控制寄存器 (HCSPLTn)

Channel_num: $0 \leq n \leq 11$

读偏移地址: $504h + (\text{Channel_num} * 20h)$

域	说明	复位值	访问特性
31	Split 使能	1'b0	R_W
30:17	保留		RO
16	Do Complete Split 指示 OTG host 必须完成一个完整的 split 传输	1'b0	R_W
15:14	事务位置 (XactPos) 2'b00: all, 此事务的整个数据 (少于或等于 188B) 2'b01: begin, 此事务的开始数据 (少于或等于 188B) 2'b10: Mid, 此事务的中间数据 (少于或等于 188B) 2'b01: End, 此事务的最后数据 (少于或等于 188B)	2'h0	R_W
13:7	Hub 地址 (HubAddr)	7'b0	R_W
6 : 0	端口地址 (PrtAddr)	7'h0	R_W

13.3.3.10 Host Channel-n 中断寄存器 (HCINTn)

Channel_num: $0 \leq n \leq 11$

读偏移地址: $508h + (\text{Channel_num} * 20h)$

域	说明	复位值	访问特性
31:14	保留		RO
13	描述符 rollover 中断 (DESC_LST_ROLLIntr) 仅当 Scatter/Gather DMA 模式使能时, 此域有效。当对应通道的描述符列表 roll over 时置 1。 对于非 Scatter/Gather DMA 模式, 此域位保留域	1'b0	R_SS_WC
12	过多事务错 (XCS_XACT_ERR) 当 Scatter/Gather DMA 模式使能时, 此域有效。当 3 个连续的事务出错时时置 1。 对于非 Scatter/Gather DMA 模式, 此域位保留域	1'b0	R_SS_WC
11	BNA(buffer 不可用) 中断 (BNAIntr) 当 Scatter/Gather DMA 模式使能时, 此域有效。当描述符访问还未准备好时置 1。 对于非 Scatter/Gather DMA 模式, 此域位保留域	1'b0	R_SS_WC
10	数据反转位错 (DataTglErr)	1'b0	R_SS_WC
9	帧 overrun (FrmOvrnun)	1'b0	R_SS_WC
8	空泡错 (BblErr)	1'b0	R_SS_WC
7	事务错 (XactErr) 当 USB 总线上发生了以下错误时置 1 : - CRC 验证失败 - 超时 - 位填充错 - 错误的 EOP	1'b0	R_SS_WC
6	收到 NYET 握手 (NYET)	1'b0	R_SS_WC
5	收到 ACK 握手 (ACK)	1'b0	R_SS_WC
4	收到 NAK 握手 (NAK)	1'b0	R_SS_WC
3	收到 STALL 握手 (STALL)	1'b0	R_SS_WC
2	AHB 错 (AHBErr)	1'b0	R_SS_WC

域	说明	复位值	访问特性
1	通道停止 (ChHltd) 对于非 Scatter/Gather DMA 模式, 由于 USB 传输错或软件置无效请求而引起传输的非正常结束, 或正常结束时置 1 对于 Scatter/Gather DMA 模式, 若因为以下原因导致传输结束, 则置 1: - 描述符中 EOL 置位 - AHB 错 - 过多的事务错 - 软件置无效请求 - 空泡 - Stall	1'b0	R_SS_WC
0	传输结束 (XferCompl) 对于 Scatter/Gather DMA 模式, 指示完成对当前描述符的处理, 其描述符的 IOC 位置 1。 对于非 Scatter/Gather DMA 模式, 表示没有任何错误的传输完成。	1'b0	R_SS_WC

13.3.3.11 Host Channel-n 中断屏蔽寄存器 (HCINTMSK_n)

Channel_num: 0 ≤ n ≤ 11

读偏移地址: 50Ch + (Channel_num * 20h)

域	说明	复位值	访问特性
31:14	保留		RO
13	描述符 rollover 中断屏蔽位 (DESC_LST_ROLLIntrMsk) 仅当 Scatter/Gather DMA 模式使能时, 此域有效。 对于非 Scatter/Gather DMA 模式, 此域位保留域	1'b0	R_W
12	保留		RO
11	保留		RO
10	数据反转位错中断屏蔽位 (DataTglErrMsk)	1'b0	R_W
9	帧 overrun 中断屏蔽位 (FrmOvrMsk)	1'b0	R_W
8	空泡错中断屏蔽位 (BblErrMsk)	1'b0	R_W
7	事务错中断屏蔽位 (XactErrMsk)	1'b0	R_W
6	收到 NYET 握手中断屏蔽位 (NYETMsk)	1'b0	R_W
5	收到 ACK 握手中断屏蔽位 (ACKMsk)	1'b0	R_W
4	收到 NAK 握手中断屏蔽位 (NAKMsk)	1'b0	R_W
3	收到 STALL 握手中断屏蔽位 (STALLMsk)	1'b0	R_W
2	AHB 错中断屏蔽位 (AHBErrMsk)	1'b0	R_W
1	通道停止中断屏蔽位 (ChHltdMsk)	1'b0	R_W
0	传输结束中断屏蔽位 (XferComplMsk)	1'b0	R_W

13.3.3.12 Host Channel-n 传输大小寄存器 (HCTSIZ_n)

Channel_num: 0 ≤ n ≤ 11

读偏移地址：510h + (Channel_num * 20h)

在 Scatter/Gather DMA 模式下，此寄存器各个域定义如下

域	说明	复位值	访问特性
31	Do ping(DoPng) 仅对 OUT 传输有效，置 1 时，指示 Host 做 PING 协议。 若对 IN 传输将此位置 1，将使此通道无效	1'b0	R_W
30:29	PID(Pid) 2'b00:DATA0 2'b01:DATA2 2'b10:DATA1 2'b11:MDATA (非控制)	2'b00	R_W
28:16	保留域	9'b0	RO
15:8	NTD(传输描述符的数目) (非实时传输) 0 - 1 个描述符 63 - 64 个描述符 (实时传输) 7 - 8 个描述符 15 - 16 个描述符 31 - 32 个描述符 63 - 64 个描述符 127 - 128 个描述符 255 - 256 个描述符	8'h0	R_W
7:0	SCHEID_INFO (调度信息) 对应的每一位代表着在那个 microFrame 中角度。Bit0 代表在 1st microFrame 中调度。Bits7 代表在 8th 中调度。 8'b11111111 代表着对应的中断通道被调度，在对应帧时间内每个 microFrame 都发送一个标记。8'b10101010 代表对应的中断通道被调度，在对应帧时间内每隔一个 microFrame 发送一个标记。 注意，此域仅对周期性传输有效。	8'h0	R_W

在非 Scatter/Gather DMA 模式下，此寄存器各个域定义如下

域	说明	复位值	访问特性
31	Do ping(DoPng) 仅对 OUT 传输有效，置 1 时，指示 Host 做 PING 协议。 若对 IN 传输将此位置 1，将使此通道无效	1'b0	R_W
30:29	PID(Pid) 2'b00:DATA0 2'b01:DATA2 2'b10:DATA1 2'b11:MDATA (非控制)	2'b00	R_W

域	说明	复位值	访问特性
28:19	包数目 (PktCnt). 软件写入传输的包数据。 每当一个传输完成, host 将此域减 1. 一旦为 0, 软件将接收到中断表示正常结束。	10'h0	R_W
18:0	传输大小 (XferSize) 对于 OUT 传输, 指定 host 将发送的数据字节数; 对于 IN 传输, 指定 host 将接收到的数据字节数。	19'h0	R_W

13.3.3.13 Host Channel-n DMA 地址寄存器 (HCDMA_n)

Channel_num: 0 ≤ n ≤ 11

读偏移地址: 514h + (Channel_num * 20h)

在 IN/OUT 传输中, 此寄存器用来维持当前 buffer 的地址。DMA 传输的起始地址必须双字对齐。

域	说明	复位值	访问特性																				
DMA 模式																							
31:0	DMA 地址 (DMAAddr) 每个 AHB 事务后累加。	X	R_W																				
描述符 DMA 模式																							
31 : N (同步) 31:9 (非同步)	DMA 地址 (DMAAddr) 非同步传输: 保存 512 字节页的起始地址。描述符列表的第一个描述符即在此地址上。OTG 控制器从 CTD 开始处理描述符列表 同步传输: 保存 2*(nTD+1) 个字节的地址, 此地址为实时描述符所在位置。其中 N 的值基于 nTD 而来, 具体如下表: <table border="1" style="margin: 5px 0;"> <tr> <td>31 : N</td> <td>N-1 : 3</td> <td>2:0</td> </tr> <tr> <td>基址</td> <td>偏移</td> <td>000</td> </tr> </table> <table border="1" style="margin: 5px 0;"> <tr> <td>nTD</td> <td>N</td> </tr> <tr> <td>7</td> <td>6</td> </tr> <tr> <td>15</td> <td>7</td> </tr> <tr> <td>31</td> <td>8</td> </tr> <tr> <td>63</td> <td>9</td> </tr> <tr> <td>127</td> <td>10</td> </tr> <tr> <td>255</td> <td>11</td> </tr> </table>	31 : N	N-1 : 3	2:0	基址	偏移	000	nTD	N	7	6	15	7	31	8	63	9	127	10	255	11	23'h0	R/W
31 : N	N-1 : 3	2:0																					
基址	偏移	000																					
nTD	N																						
7	6																						
15	7																						
31	8																						
63	9																						
127	10																						
255	11																						
N-1 : 3 (实时) 8: (非实时)	当前传输描述符 (CTD) 非实时传输: 以描述符为单位, 范围 0 - 63. 指定当前正在处理的描述符。 实时传输: 基于当前 micorFrame 的值, 需被软件清 0	6'h0	R_W																				
2:0	保留	3'h0	RO																				

13.3.3.14 Host Channel-n DMA Buffer 地址寄存器 (HCDMBA_n)

Channel_num: 0 ≤ n ≤ 11

读偏移地址：518h + (Channel_num * 20h)

此仅在 Scatter/Gather DMA 模式下有效

域	说明	复位值	访问特性
31:0	当前 buffer 的地址。	X	RO

13.3.3.15 Host Frame List 基地址寄存器 (HFLBAddr)

Offset : 41Ch

此仅在 Scatter/Gather DMA 模式下有效

域	说明	复位值	访问特性
31:0	Frame list 的起始地址	32'h0	RO

13.4 Device 模式寄存器

这组寄存器仅在 device 模式下有效，host 模式下不可访问。

13.4.0.16 Device 配置寄存器 (DCFG)

Offset : 800h

在初始配置过后，不能修改此寄存器的值

域	说明	复位值	访问特性
31:26	Resume 有效期 (ResValid) 此域控制 OTG 控制器从挂起状态到 resume 的时间。以时钟周期为单位。仅当 DCFG.Ena32KHzSusp 置 1 时，此位有效	6'd2	R_W
25:24	周期性调度间隔 (PerSchIntvl) 仅在 Scatter/Gather DMA 模式下有效。 用来指定 DMA 引擎给周期性 IN 端点获取数据分配的时间。基于周期性 IN 端点的数目，其值必须设定为 75% microFrame。 - 当任意周期性端点活动时，DMA 引擎给周期性 IN 端点获取数据分配一定的时间 - 当无周期性端点活动时，忽略此域 - 在指定时间后，DMA 引擎开始给非周期性端点获取数据。 2'b00: 25% microFrame 2'b01: 50% microFrame 2'b10: 75% microFrame 2'b11: 保留	2'b00	R_W

域	说明	复位值	访问特性
23	在 device 模式下使能 Scatter/Gather DMA 模式 (DescDMA) - GAHBCFG.DMAEn=0, DCFG.DescDMA=0 = ζ 从模式 - GAHBCFG.DMAEn=0, DCFG.DescDMA=1 = ζ 非法 - GAHBCFG.DMAEn=1, DCFG.DescDMA=0 = ζ buffer DMA 模式 - GAHBCFG.DMAEn=0, DCFG.DescDMA=0 = ζ Scatter/-Gather DMA 模式	1'b0	R_W
22:14	保留		RO
13	是否使能 device OUT NAK。 仅对描述符 DMA 模式有效。 1'b0: 在批量 OUT 传输结束时, OTG 控制器不送出 NAK 握手信号 1'b0: 在批量 OUT 传输结束时, OTG 控制器送出 NAK 握手信号	1'b0	R_W
12:11	周期帧的间隔 (PerFrInt) 指示在一个 microFrame 内何时用 EOP 中断来通知软件。此域可用来决定当前 microFrame 内是否所有的实时传输已经完成。 2'b00: 80% 2'b01: 85% 2'b00: 90% 2'b00: 95%	2'h0	R_W
10:4	Device 地址 (DevAddr)	7'h0	R_W
3	保留	1'b0	RO
2	非 0 长度的 Status OUT 握手 (NZStsOUTHShk) 当控制传输的 Status 阶段的 OUT 事务中, OTG 接收到一个非 0 的数据包, 此域用来选着 OTG 控制送出的不同的握手信号。 1'b1: STALL 1'b0: 送出接收到的 OUT 数据包, 并基于 device endpoint 控制寄存器中 NAK 与 STALL 位, 送出握手信号	1'b0	R_W
1:0	设备速度 (DevSpd) 2'b00: HS(USB2.0 PHY) 2'b01: FS(USB2.0 PHY) 2'b10: LS(USB1.1 PHY) 2'b11: FS(USB1.1 PHY)	2'b0	R_W

13.4.0.17 Device 控制寄存器 (DCTL)

Offset : 804h

域	说明	复位值	访问特性
31:17	保留		RO
16	遇到 babble 发送 NAK 握手。	1'b0	R_W

域	说明	复位值	访问特性
15	对实时端节点忽略帧号 (IgnrFrmNum) GAHBCFG.DMAEn=0, 此域无效 GAHBCFG.DMAEn=1, DCFG.DescDMA = 0: 当 threshold 模式有效时, 此位无效。否则, 软件用此位来使能周期性传输中断。 - 1'b0 周期性传输中断无效。 - 1'b1 当接收到 ISOC IN 标记时, 无需清空包。OTG 控制器忽略帧号, 一旦包准备好且接收到对应的标记, 就将这个包发送出去。此域也用来使能周期性传输中断 GAHBCFG.DMAEn=1, DCFG.DescDMA = 1 : 对高速的传输无效	1'b0	R_W
14:13	全局 Multi Count(GMC) 仅在 Scatter/Gater DMA 模式下有效。用来指示在转移到下一个结束点之前, 接收或发送的包个数。仅对非周期性端节点有效。 2'b00: 非法; 2'b01: 1 个包; 2'b00: 2 个包; 2'b00: 3 个包; 当 DCFG.DescDMA =1 时, 此域自动改变成 2'h1	2'h0	R_W
12	保留域		RO
11	Powr-on 编程完成 (PWROnPrgDone) 从 power_down 模式恢复后, 寄存器编程完成时, 此位置 1.	1'b0	R_W
10	清空全局 OUT NAK(CGOUTNak) 对此位的写, 将清空全局 OUT NAK	1'b0	WO
9	设置全局 OUT NAK(SGOUTNak) 对此位的写, 将设置全局 OUT NAK	1'b0	WO
8	清空全局的非周期性 IN NAK (CGNPInNak) 此位的写, 将清空全局非周期性 IN NAK	1'b0	WO
7	设置全局的非周期性 IN NAK (SGNPInNak) 此位的写, 将设置全局非周期性 IN NAK	1'b0	WO
6 : 4	测试控制 (TstCtl) 3'b000: 无效 Test 模式 3'b001: Test_J 模式 3'b010: Test_K 模式 3'b011: Test_SE0_NAK 模式 3'b100: Test_Packet 模式 3'b101: Test_Force.Enable 模式 其他: 保留	3'b0	R_W
3	全局 OUT NAK 状态 (GOUTNakSts) 1'b0: 基于 FIFO 的状态以及 NAK, STALL 位的设置情况来发送相应的握手 1'b1: 不管 Rx FIFO 是否有空项, 都无数据写入 Rx FIFO. 除 SETUP 事务以外, 对所有的包都发送 NAK 握手。终止所有的实时 OUT 包。	1'b0	RO

域	说明	复位值	访问特性
2	全局非周期性 IN NAK 状态位 1'b0: 基于 TxFIFO 的有效数据发送相应的握手; 1'b1: 对于所有的非周期性 IN 端节点, 发送 NAK 握手信号, 不管 TxFIFO 是否有有效数据	1'b0	RO
1	软断开 (SftDisCon) 一旦此位置 1, host 将看不到 device 的连接, device 也将接收不到 USB 上的信号。 1'b0: 正常操作 1'b1: OTG 控制器将 utmi+ 接口上的 phy_opmode_o 信号置为 2'b01, 这将给 USB host 产生一个断开事件	1'b0	R_W
0	远程唤醒信号 (RmtWkUpSig) 当次位置 1, OTG 控制器将发送一个远程唤醒信号去唤醒 USB host。	1'b0	R_W

13.4.0.18 Device 状态寄存器 (DSTS)

Offset : 808h

域	说明	复位值	访问特性
31 : 22	保留		RO
21:8	接收到的 SOF 的帧号 (SOFN)	14'h0	RO
7 : 4	保留		RO
3	严重错 (ErrticErr) OTG 控制器通过此域来报告发生了严重错误 (由于 PHY 出错, phy_rxvalid.i/phy_rxvldh.i 或 phy_rxactive.i 置位超过 2ms), 一旦发生严重错误, OTG 控制器进入挂起模式, GINTSTS.ErlySusp 位置 1. 且软件仅能通过软断开来恢复		RO
2:1	枚举速度 (EnumSpd) 2'b00: HS(USB2.0 PHY) 2'b01: FS(USB2.0 PHY) 2'b10: LS(USB1.1 PHY) 2'b11: FS(USB1.1 PHY)		RO
0	挂起状态 (SuspSts) Device 模式下, 只要在 USB 总线上发现挂起状态, 则置 1. 当 utmi_linestate 信号在一定事件内无活动, 则 OTG 控制器进入挂起状态。一旦以下条件发生, 则 OTG 控制器退出挂起状态: 当 utmi_linestate 信号有活动 当软件往 DCTL.RmtWkUpSig 位写 1 时。	1'b0	RO

13.4.0.19 Device IN 端点共用的中断屏蔽寄存器 (DIEPMSK)

Offset : 810h

域	说明	复位值	访问特性
31 : 14	保留		RO
13	NAK 中断屏蔽位 (NAKMsk)	1'h0	R_W

域	说明	复位值	访问特性
12:10	保留		RO
9	BNA 中断屏蔽位 (BNAInIntrMsk) 仅当描述符 DMA 模式时有效	1'h0	R_W
8	FIFO 下溢中断屏蔽位屏蔽 (TxFifoUndrnMSK)	1'b0	R_W
7	保留	1'b0	RO
6	IN 端节点 NAK 有效中断屏蔽位 (INEPNakEffMsk)	1'b0	R_W
5	接收到 EP 不匹配的 IN 标记中断屏蔽位 (INTknEPMisMsk)	1'b0	R_W
4	当 TxFIFO 空时接收到 IN 标记中断屏蔽位 (INTknTXFEmpMsk)	1'b0	R_W
3	超时中断屏蔽位 (TImeOUTMsk) 仅对非实时端节点有效	1'b0	R_W
2	AHB 错误中断屏蔽位 (AHBErrMsk)	1'b0	R_W
1	端节点失效中断屏蔽位 (EPDisbldMsk)	1'b0	R_W
0	传输完成中断屏蔽位 (XferCompIMsk)	1'b0	R_W

13.4.0.20 device OUT 端点共用的中断屏蔽寄存器 (DOEPMSK)

Offset : 810h

域	说明	复位值	访问特性
31 : 15	保留		RO
14	NYET 中断屏蔽位 (NYETMsk)	1'b0	R_W
12	空泡中断屏蔽位 (BbleErrMsk)	1'b0	R_W
11:10	保留		RO
9	BNA 中断屏蔽位 (BnaOutIntrMsk)	1'b0	R_W
8	OUT 包错误中断屏蔽位 (OutPktErrMsk)	1'b0	R_W
7	保留		RO
6	接收到背对背的 SETUP 包中断屏蔽位 (Back2BackSETup)	1'b0	R_W
5	保留		RO
4	当端节点失效时接收到 OUT 标记中断屏蔽位 (OUTTknEPdisMsk)	1'b0	R_W
3	SETUP 阶段完成中断屏蔽位 (SetUPMsk)	1'b0	R_W
2	AHB 错误中断屏蔽位 (AHBErrMsk)	1'b0	R_W

域	说明	复位值	访问特性
1	端节点失效中断屏蔽位 (EPDisbldMsk)	1'b0	R_W
0	传输完成中断屏蔽位 (XferComplMsk)	1'b0	R_W

13.4.0.21 Device 所有端点共用的中断寄存器 (DAINT)

Offset : 818h

域	说明	复位值	访问特性
31 : 16	OUT 端节点中断位 (OutEPInt) 一个 OUT 端节点对应一位。 位 16 对应端节点 0	16'h0	RO
15:0	IN 端节点中断位 (InEPInt) 一个 IN 端节点对应一位。 位 0 对应端节点 0	16'h0	RO

13.4.0.22 Device 所有端点共用的中断屏蔽寄存器 (DAINTMSK)

Offset : 81Ch

域	说明	复位值	访问特性
31 : 16	OUT 端节点中断屏蔽位 (OutEPIntMsk) 一个 OUT 端节点对应一位。 位 16 对应端节点 0	16'h0	R_W
15:0	IN 端节点中断屏蔽位 (InEPIntMsk) 一个 IN 端节点对应一位。 位 0 对应端节点 0	16'h0	R_W

13.4.0.23 Device VBUS 放电寄存器 (DVBUSDIS)

Offset : 828h

域	说明	复位值	访问特性
31:16	保留		RO
15:0	Device Vbus 放电时间 (DVBUSDIs) SRP 期间, 在 Vbus 脉冲后 Vbus 放电的时间。以 PHY 时钟为单 位。	30MHZ: 16'h0B8F 60MHZ: 16'h17D7	R_W

13.4.0.24 Device VBUS 脉冲寄存器 (DVBUSPULSE)

Offset : 82Ch

域	说明	复位值	访问特性
31:12	保留		RO

域	说明	复位值	访问特性
11:0	Device Vbus 脉冲时间 (DVBUSPulse) SRP 期间, 在 Vbus 脉冲的时间。以 PHY 时钟为单位。	30MHZ: 12'h2c6 60MHZ: 12'h5b8	R_W

13.4.0.25 Device 阈值控制寄存器 (DTHRCTL)

Offset : 830h

域	说明	复位值	访问特性
31:28	保留		RO
27	仲裁器 parking 使能	1'b1	R_W
26	保留		RO
25:17	接收阈值长度 (RxThrLen)	9'h8	R_W
16	接收阈值使能 (RxTHrEn)	1'b0	R_W
15:13	保留		RO
12:11	AHB 阈值率 (AHBThrRatio)	2'b0	R_W
10:2	发送阈值长度 (TxThrLen)	9'h8	R_W
1	实时 IN 端节点阈值使能 (ISOThrEn)	1'b0	R_W
0	非实时 IN 端节点阈值使能 (NonISOThrEn)	1'b0	R_W

13.4.0.26 Device IN 端节 FIFO 空中断屏蔽寄存器 (DIEPEMPSK)

Offset : 834h

域	说明	复位值	访问特性
31:16	保留		RO
15:0	IN 端节点 Tx FIFO 空中断屏蔽位 (InEpTxfEmpMsk) 作为 DIEPINTn 的中断屏蔽位, 一位对应一个端节点。Bit0 对应端节点 0, 以此类推	16'b0	R_W

13.4.0.27 Device 控制 IN 端节 0 控制寄存器 (DIEPCTL0)

Offset : 900h

域	说明	复位值	访问特性
31	端节点使能 (EPEna) - 对于 Scatter/Gather DMA 模式, 对于 IN 端节点来讲, 此域置 1 意味着描述符以及数据缓冲都已经准备好。 - 对于非 Scatter/Gather DMA 模式, 此域置 1 意味着数据缓冲已经准备好。 当 OTG 设置以下中断时, 将此位清 0 : - 端节点失效 - 传输完成	1'b0	R_WS_SC

域	说明	复位值	访问特性
30	端节点失效 (EPDis) 软件将此位置 1，即使此端节点上的传输未完成，也将停止在此端节点上发送数据。软件必须等待端节点失效中断发生后，才能认为此端节点已经失效。当 OTG 控制器设置端节点失效中断时，将此位清 0。软件仅当在 EPEna 位为 1 时才能置此位为 1。 此域仅在 DMA 模式下有效。	1'b0	R_WS_SC
29:28	保留		RO
27	设置 NAK(SNAK) 置 1, 设置此端节点对应的 NAK 位为 1.	1'b0	WO
26	清空 NAK (CNAK) 置 1, 设置此端节点对应的 NAK 位为 0.	1'b0	WO
25:22	TxFIFO 号 (TxFNum) 设置为 IN 端点 0 对应的 FIFO 号	4'h0	R_W
21	STALL 握手 (Stall) 当收到 SETUP 包时，软件置 1，OTG 控制器清 0。 如果同时 NAK 位，全局非周期性 IN NAK，全局 OUT NAK 位都置 1，那么 STALL 位优先级最高。	1'b0	R_WS_SC
20	保留		RO
19:18	端节点类型 (EPType) 对于控制节点为 2'b00	2'h0	RO
17	NAK 状态 (NAKSts) 1'b0: 基于 FIFO 的状态，OTG 控制器发送非 Non-NAK 握手 1'b1: OTG 控制器发送 NAK 握手	1'b0	RO
16	保留		RO
15	USB 活动端节点 (USBActEP) 一直为 1，表示控制端节点 0 总是可用	1'b1	RO
14:2	保留		RO
1:0	最大的包大小 (MPS) 2'b00 : 64B 2'b01 : 32B 2'b10 : 16B 2'b11 : 8B	2'h0	R_W

13.4.0.28 Device 控制 OUT 端节 0 控制寄存器 (DOEPCTL0)

Offset : B00h

域	说明	复位值	访问特性
31	端节点使能 (EPEna) - 对于 Scatter/Gather DMA 模式, 对于 OUT 端节点来讲, 此域置 1 意味着描述符以及数据缓冲都已经准备好接收数据。 - 对于非 Scatter/Gather DMA 模式, 此域置 1 意味着数据缓冲已经准备好。 当 OTG 设置以下中断时, 将此位清 0 : - SETUP 时相完成 - 端节点失效 - 传输完成 在 DMA 模式下, 若 OTG 控制器传输 SETUP 数据包时, 此位必须置 1.	1'b0	R_WS_SC
30	端节点失效 (EPDis) 软件不能使 OUT 控制节点 0 失效	1'b0	RO
29:28	保留		RO
27	设置 NAK(SNAK) 置 1, 设置此端节点对应的 NAK 位为 1.	1'b0	WO
26	清空 NAK (CNAK) 置 1, 清空此端节点对应的 NAK 位为 1.	1'b0	WO
25:22	保留		RO
21	STALL 握手 当收到 SETUP 包时, 软件置 1, OTG 控制器清 0. 如果同时 NAK 位, 全局非周期性 NAK, 全局 OUT NAK 位都置 1, 那么 STALL 位优先级最高。但是不管此位是否置 1, OTG 控制器对 SETUP 数据包的握手信号总是 ACK	1'b0	R_WS_SC
20	Snoop 模式 (Snp) 在 snoop 模式下, 在传输数据给内存时, OTG 控制器不检查 OUT 数据包的正确性	1'b0	R_W
19:18	端节点类型 (EPType) 对于控制节点为 2'b00	2'h0	RO
17	NAK 状态 (NAKSts) 1'b0: 基于 FIFO 的状态, OTG 控制器发送非 Non-NAK 握手 1'b1: OTG 控制器发送 NAK 握手	1'b0	RO
16	保留		RO
15	USB 活动端节点 (USBActEP) 一直为 1, 表示控制端节点 0 总是可用	1'b1	RO
14:2	保留		RO
1:0	最大的包大小 (MPS) 2'b00 : 64B 2'b01 : 32B 2'b10 : 16B 2'b11 : 8B	2'h0	R_W

13.4.0.29 Device 端节- n 控制寄存器 (DIEPCTLn/DOEPCTLn)

Endpoint_num : $1 \leq n \leq 6$

Offset for IN : $900h + (\text{Endpoint_num} * 20h)$

Offset for OUT : B00h + (Endpoint_num*20h)

域	说明	复位值	访问特性
31	<p>端节点使能 (EPEna)</p> <p>应用于 IN/OUT 端节点。</p> <ul style="list-style-type: none"> - 对于 Scatter/Gather DMA 模式, 对于 IN 端节点来讲, 此域置 1 意味着描述符以及数据缓冲都已经准备好; 对于 OUT 端节点来讲, 此域置 1 意味着描述符以及数据缓冲都已经准备好接收数据。 - 对于非 Scatter/Gather DMA 模式, 对于 IN 端节点来讲, 此域置 1 意味着数据缓冲已经准备好发送数据; 对于 IN 端节点来讲, 此域置 1 意味着数据缓冲已经准备好接收数据; <p>当 OTG 设置以下中断时, 将此位清 0 :</p> <ul style="list-style-type: none"> - SETUP 时相完成 - 端节点失效 - 传输完成 	1'b0	R_WS_SC
30	<p>端节点失效 (EPDis)</p> <p>应用于 IN/OUT 端节点</p> <p>软件将此位置 1, 即使此端节点上的传输未完成, 也将停止在此端节点上发送数据。软件必须等待端节点失效中断发生后, 才能认为此端节点已经失效。当 OTG 控制器设置端节点失效中断时, 将此位清 0. 软件仅当在 EPEna 位为 1 时才能置此位为 1.</p>	1'b0	R_WS_SC
29	<p>设置 DATA1 PID (SetD1PID)</p> <p>仅对中断 / 批量 IN 以及 OUT 端节点有效。</p> <p>对此位写 1, 则置 DPID 位 DATA1。</p> <p>对描述符 DMA 与非描述符 DMA 模式都有效</p>	1'b0	WO
28	<p>设置 DATA0 PID (SetD0PID)</p> <p>仅对中断 / 批量 IN 以及 OUT 端节点有效。</p> <p>对此位写 1, 则置 DPID 位 DATA0。</p> <p>对描述符 DMA 与非描述符 DMA 模式都有效。</p>	1'b0	WO
27	<p>设置 NAK(SNAK)</p> <p>应用于 IN/OUT 端节点</p> <p>置 1, 设置此端节点对应的 NAK 位为 1.</p>	1'b0	WO
26	<p>清空 NAK (CNAK)</p> <p>应用于 IN/OUT 端节点</p> <p>置 1, 清空此端节点对应的 NAK 位为 1.</p>	1'b0	WO
25:22	<p>TxFIFO 号 (TxFNum)</p> <p>仅对 IN 端节点有效;</p> <p>设置为 IN 端对应的 FIFO 号</p>	4'h0	R_W

域	说明	复位值	访问特性
21	<p>STALL 握手</p> <p>仅对非控制，非实时的 IN/OUT 端节点有效。</p> <p>通过设置此位，来 stall 所有从 host 来的标记。如果同时 NAK 位，全局非周期性 NAK，全局 OUT NAK 位都置 1，那么 STALL 位优先级最高。但是不管此位是否置 1，OTG 控制器对 SETUP 数据包的握手信号总是 ACK</p> <p>仅对控制端节点有效。</p> <p>当收到 SETUP 包时，软件置 1，OTG 控制器清 0。</p> <p>如果同时 NAK 位，全局非周期性 NAK，全局 OUT NAK 位都置 1，那么 STALL 位优先级最高。但是不管此位是否置 1，OTG 控制器对 SETUP 数据包的握手信号总是 ACK</p>	1'b0	R_W
20	<p>Snoop 模式 (Snp)</p> <p>仅对 OUT 端节点有效</p> <p>在 snoop 模式下，在传输数据给内存时，OTG 控制器不检查 OUT 数据包的正确性</p>	1'b0	R_W
19:18	<p>端节点类型 (EPType)</p> <p>2'b00: 控制</p> <p>2'b01: 实时</p> <p>2'b10: 批量</p> <p>2'b11: 中断</p>	2'h0	R_W
17	<p>NAK 状态 (NAKSts)</p> <p>应用于所有的 IN/OUT 端节点</p> <p>1'b0: 基于 FIFO 的状态，OTG 控制器发送非 Non-NAK 握手</p> <p>1'b1: OTG 控制器发送 NAK 握手</p>	1'b0	RO
16	<p>端节点数据 PID (DPID)</p> <p>应用于中断 / 批量 IN 以及 OUT 端节点。</p> <p>包括在此端节点接收或发送的数据包的 PID。当此端节点活动后，软件必须设置此端节点接收或发送的第一个包的 PID。使用此寄存器的 SetD0PID 以及 SetD1PID 域来设置 DATA0 或 DATA1 PID</p> <p>1'b0: DATA0</p> <p>1'b1: DATA1</p> <p>奇偶帧 (EO_FrNum)</p> <p>在非描述符 DMA 模式下，应用于实时的 IN/OUT 端节点。</p> <p>指定为此端节点发送或接收实时数据所用地帧号。通过此寄存器的 SetEvnFr 与 SetOddFr 域来设置。</p> <p>1'b0: 偶帧</p> <p>1'b1: 奇帧</p> <p>在描述符 DMA 模式下，此域位保留域</p>	1'b0	RO
15	<p>USB 活动端节点</p> <p>应用于 IN 或 OUT 端节点。</p> <p>指示在当前配置与接口下，端节点是否活动。</p>	1'b0	R_W_Sc
14:11	保留		RO
10:0	<p>最大包大小 (MPS)</p> <p>应用于 IN 或 OUT 端节点。设置最大的包大小，以字节为单位</p>	11'b0	R_W

13.4.0.30 Device 端节- n 中断寄存器 (DIEPCTLn /DOEPCTLn)

Endpoint_num : 1 <= n <= 6

Offset for IN : 908h + (Endpoint_num*20h)

Offset for OUT : B08h + (Endpoint_num*20h)

域	说明	复位值	访问特性
31:15	保留		RO
14	NYET 中断 (NYETIntrpt)	1'b0	R_SS_WC
13	NAK 中断 (NAKIntrpt)	1'b0	R_SS_WC
12	空泡错中断 (BbleErrIntrpt)	1'b0	R_SS_WC
11	包 drop 状态 (PktDrpSts)	1'b0	R_SS_WC
10	保留		RO
9	BNA (缓冲不可用) 中断 (BNAIntr)	1'b0	R_SS_WC
8	FIFO 下溢 (TxFifoUndrn)	1'b0	R_SS_WC
7	TxFIFO 空 (TxFEmp)	1'b0	R_SS_WC
6	IN 端节点 NAK 有效 (INEPNakEff)	1'b0	R_SS_WC
5	接收到 EP 不匹配的 IN 标记中断 (INTknEPMis)	1'b0	R_SS_WC
4	当 TxFIFO 空时接收到 IN 标记中断 (INTknTXFEmp)	1'b0	R_SS_WC
3	超时中断 (TImeOUT) 仅对非实时端节点有效	1'b0	R_SS_WC
2	AHB 错误中断 (AHBErr)	1'b0	R_SS_WC
1	端节点失效中断 (EPDisbld)	1'b0	R_SS_WC
0	传输完成中断 (XferCompI)	1'b0	R_SS_WC

13.4.0.31 Device 端节 0 传输大小寄存器 (DIEPTSIZ0/DOEPTSIZ0)

Offset for IN : 910h

Offset for OUT : B10h

对于 IN 端节点:

域	说明	复位值	访问特性
31:15	保留		RO
20:19	包数量 (PktCnt) 指示在此传输大小数量下的包数目。 每当从 TxFIFO 中读取一个包, 此域减 1。	2'b0	R_W
18:7	保留域	7'h0	R_W

域	说明	复位值	访问特性
6:0	传输大小 指示端节点 0 的传输大小 (以字节为单位), 当超出这个值, 则将引发中断。可被设置为此端节点的 MPS。每次一个包写入 TxFIFO, 此域减 1	7'h0	R_W

对于 OUT 端节点:

域	说明	复位值	访问特性
31	保留		RO
30:29	SETUP 包数量 (SUPPktCnt) 指示此端节点能接受到的背对背 SETUP 包的数量 2'b01: 1 个包 2'b10: 2 个包 2'b11: 3 个包	2'b0	R_W
28:20	保留域	7'h0	R_W
19	包数量 (PktCnt) 每当一个包写入 RxTxFIFO, 此域减到 0.	1'b0	R_W
18:7	保留		RO
6:0	传输大小 (XferSize) 指示端节点 0 的传输大小 (以字节为单位), 当超出这个值, 则将引发中断。可被设置为此端节点的 MPS。每次从 RxTxFIFO 中读取一个包, 此域减 1	7'h0	R_W

13.4.0.32 Device 端节- n 传输大小寄存器 (DIEPSIZn /DOEPSIZn)

Endpoint_num : $1 \leq n \leq 6$

Offset for IN : $910h + (\text{Endpoint_num} * 20h)$

Offset for OUT : $B10h + (\text{Endpoint_num} * 20h)$

域	说明	复位值	访问特性
31:29	保留		RO

域	说明	复位值	访问特性
30:29	<p>MC (MultiCount)</p> <p>仅对 IN 端节点有效。</p> <p>对周期性 IN 端节点, 此域指定在一个 microFrame 时间内传输的包个数。OTG 控制器使用此域为实时 IN 端节点计算数据 PID。</p> <p>2'b01: 1 个包; 2'b10: 2 个包; 2'b11: 3 个包</p> <p>对于非周期性 IN 端节点, 指定 OTG 控制器必须为此端节点获取的包数, 在转换到下由 DIEPCTLn.Nextep 指定的下一个端节点时。</p> <p>RxPID (接收到的数据 PID)</p> <p>仅对实时 OUT 端节点有效</p> <p>此端节点接收到的最后一个包的数据 PID</p> <p>2'b00: DATA0 2'b01: DATA2 2'b10: DATA1 ; 2'b11: MDATA ;</p> <p>SETUP 包数 (SUPCnt)</p> <p>仅对控制 OUT 端节点有效</p> <p>指示此端节点能接受到的背对背 SETUP 包的数量</p> <p>2'b01: 1 个包 2'b10: 2 个包 2'b11: 3 个包</p>	2'b0	R_W
28:19	<p>包数量 (PktCnt)</p> <p>指示在此传输大小数量下的包数目 (PktCnt = XferSize/MPS)</p> <ul style="list-style-type: none"> - IN 端节点: 每当从 TxFIFO 中读取一个包, 此域减 1。 - OUT 端节点: 每当一个包写入 Rx FIFO, 此域减到 1 	10'h0	R_W
18:0	<p>传输大小 (XferSize)</p> <p>指示端节点的传输大小 (以字节为单位)。</p> <ul style="list-style-type: none"> - 每次从 Rx FIFO 中读取一个包, 此域减 1 - 每次一个包写入 TxFIFO, 此域减 1 	19'b0	R_W

13.4.0.33 Device 端节- n DMA 地址寄存器 (DIEPDMA_n/DOEPDMA_n)

Endpoint_num : 1 <= n <= 6

Offset for IN : 914h + (Endpoint_num*20h)

Offset for OUT : B14h + (Endpoint_num*20h)

域	说明	复位值	访问特性
31:0	<p>DMA 地址 (DMAAddr)</p> <p>保存分配给此 endpoint 的外部存储的起始地址</p> <ul style="list-style-type: none"> - 非 Scatter/Gather DMA 模式下, 为起始地址 - Scatter/Gather DMA 模式下, 为描述符表的基准地址 	X	R_W

13.4.0.34 Device 端节- n DMA Buffer 地址寄存器 (DIEPDMABn/DOEPDMABn)

Endpoint_num : $1 \leq n \leq 6$

Offset for IN : $91Ch + (Endpoint_num * 20h)$

Offset for OUT : $B1Ch + (Endpoint_num * 20h)$

域	说明	复位值	访问特性
31:0	DMA buffer 地址 (DMABufferAddr) 保存当前 buffer 的地址。当对应此端节点的数据传输在进行时，此地址将实时更新。 仅在 Scatter/Gather DMA 模式下有效。	X	RO

13.4.0.35 Device IN 端节点- nTx FIFO 状态寄存器 (DTXSTS_n)

Endpoint_num : $1 \leq n \leq 6$

Offset $918h + (Endpoint_num * 20h)$

域	说明	复位值	访问特性
31:16	保留		RO
15 : 0	IN 端节点 Tx FIFO 可用空间 (INEPTxFSpcAvail) 以 32 字节为单位。 16'h0: 端节点 Tx FIFO 满 16'h1: 一个字可用 ... 16'hn: n 个字可用 其他: 保留		RO

13.4.1 功耗与门控时钟寄存器

13.4.1.1 功耗与门控时钟控制寄存器 (PCGCCTL)

Offset E00h

域	说明	复位值	访问特性
31:9	保留		RO
8	挂起后复位 (ResetAfterSusp) 如果 host 需要在挂起后发出复位，那么在 clamp 移除之前，软件需置 1 此位。如果此位未置 1，那么 host 在挂起之后发出复位信号	1'b1	R_W
7	深度睡眠 (L1 Suspended) 当处于 L1 状态时，PHY 进入深度睡眠。	1'b0	RO
6	PHY 睡眠 (PhySleep) PHY 进入睡眠状态。	1'b0	RO
5	使能睡眠门控 (EnblL1Gating) 如果 OTG 控制器不能置位 utmi.l1.suspend.n 时，设置此位后，在睡眠状态下能对内部时钟做门控	1'b0	R_W

域	说明	复位值	访问特性
4	保留		RO
3	复位低功耗模块 (RstPdownModule) 当断电时, 软件置 1. 在上电以及 PHY 时钟起来后, 软件清 0	1'b0	R_W
2	供电隔离 (PwrClmp) 在关电之前, 软件将此位置 1 用来隔离 power-on 与 power-off 模块。在上电之前, 软件清 0	1'b0	R_W
1	门控 Hclk(GateHclk) 置 1, 则当 USB 处于挂起状态时, 对 Hclk 使用门控时钟。当 USB 恢复或新的对话开始时, 软件清 0	1'b0	R_W
0	停止 Pclk(StopPclk) 置 1, 则当 USB 处于挂起状态时, 停止 PHY 时钟。当 USB 恢复或新的对话开始时, 软件清 0	1'b0	R_W

第十四章 HDA 控制器

14.1 功能概述

HDA 控制器兼容 High Definition Audio Specification Revision 1.0a，主要的功能包括各种输入、输出流组合，对 48KHZ, 和 44.1KHZ 的采样频率的支持，初始化序列，命令控制通道等。HDA 控制器的整体设计框架包括了 5 个大的模块，分别为 SDI，SDO，axi_master, axi_slave, 和 reg config。其中 axi_master 和 axi_slave 分别控制了 HDA 中 DMA 的读写通道和对 HDA 进行配置时的 AXI 总线控制情况。Reg config 主要的作用就是对 HDA 中的寄存器进行配置，控制 SDI SDO 的参数和运行情况。SDI 和 SDO 主要是对输入输入流的控制，包括 4 个输入流和 4 个输入流。

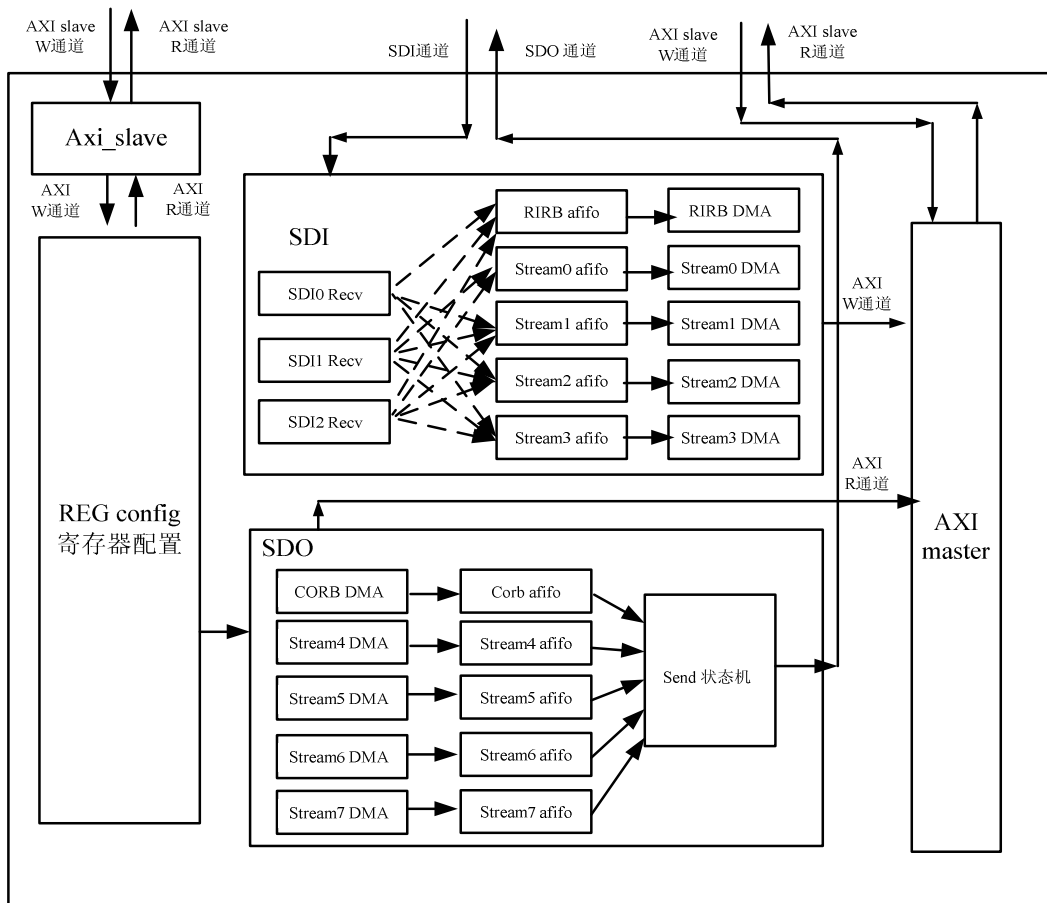


图 14.1: HDA 模块的整体设计框图

14.2 寄存器描述

寄存器的设计也是完全按照 HD audio Rev 1.0 规范进行设计的，下表列举了主要的寄存器的参数信息，具体的参考 HD audio Rev 1.0 手册。

14.2.1 协议定义的音频控制器寄存器集

偏移开始	偏移结束	名称	描述
00	01	GCAP	全局能力
02	02	VMIN	次要版本号
03	03	VMAJ	主要版本号
04	05	OUTPAY	输出的载荷能力
06	07	INPAY	输入的载荷能力
08	0B	GCTL	全局控制
0C	0D	WAKEEN	唤醒启用使能
0E	0F	WAKESTS	唤醒启用状态标志
10	11	GSTS	全局状态
12	17	Rsvd	保留
18	19	OUTSTRMPAY	输出流负载能力
1A	1B	INSTRMPAY	输入流负载能力
1C	1F	Rsvd	保留
20	23	INTCTL	中断控制
24	27	INTSTS	中断状态
28	2F	Rsvd	保留
30	33	WALCLK	时钟计数器
34	37	Rsvd	保留
38	3B	SSYNC	系统同步
3C	3F	Rsvd	保留
40	43	CORBIBASE	CORB 基地址的低位
44	47	CORBUBASE	CORB 基地址的高位
48	49	CORBWP	CORB 写指针
4A	4B	CORBRP	CORB 读指针
4C	4C	CORBCTL	CORB 控制寄存器
4D	4D	CORBSTS	CORB 状态寄存器
4E	4E	CORBSIZE	CORB 大小寄存器
4F	4F	Rsvd	保留
50	53	RIRBIBASE	RIRB 基地址的低位
54	57	RIRBUBASE	RIRB 基地址的高位
58	59	RIRBWP	RIRB 写指针
5A	5B	RIRBINTCNT	RIRB 读指针
5C	5C	RIRBCTL	RIRB 控制寄存器
5D	5D	RIRBSTS	RIRB 状态寄存器
5E	5E	RIRBSIZE	RIRB 大小寄存器
5F	5F	Rsvd	保留

偏移开始	偏移结束	名称	描述
60	63	ICOI	立即命令输出接口
64	67	ICII	立即命令输入接口
68	69	ICIS	立即命令状态寄存器
6A	6F	Rsvd	保留
70	73	DPIBLBASE	DMA 在 buf 中的位置低地址
74	77	DPIBUBASE	DMA 在 buf 中的位置高地址
78	7F	Rsvd	保留
80	82	SD0CTL	输入流控制寄存器
83	83	SD0STS	输入流转台寄存器
84	87	SD0LPIB	输入流 link 在 buf 中的位置寄存器
88	8B	SD0CBL	输入流循环 buf 的长度
8C	8D	SD0LVI	输入流最后一个有效的位置
8E	8F	Rsvd	保留
90	91	SD0FIFOD	输入流 FIFO 的大小
92	93	SD0FMT	输入流格式
94	97	Rsvd	保留
98	9B	SD0BDPL	输入流 BDL 的低地址
9C	9F	SD0BDPU	输入流 BDL 的高地址
A0	A2	SD1CTL	输入流控制寄存器
A3	A3	SD1STS	输入流转台寄存器
A4	A7	SD1LPIB	输入流 link 在 buf 中的位置寄存器
A8	AB	SD1CBL	输入流循环 buf 的长度
AC	AD	SD1LVI	输入流最后一个有效的位置
AE	AF	Rsvd	保留
B0	B1	SD1FIFOD	输入流 FIFO 的大小
B2	B3	SD1FMT	输入流格式
B4	B7	Rsvd	保留
B8	BB	SD1BDPL	输入流 BDL 的低地址
BC	BF	SD1BDPU	输入流 BDL 的高地址
C0	C2	SD2CTL	输入流控制寄存器
C3	C3	SD2STS	输入流转台寄存器
C4	C7	SD2LPIB	输入流 link 在 buf 中的位置寄存器
C8	CB	SD2CBL	输入流循环 buf 的长度
CC	CD	SD2LVI	输入流最后一个有效的位置
CE	CF	Rsvd	保留
D0	D1	SD2FIFOD	输入流 FIFO 的大小
D2	D3	SD2FMT	输入流格式
D4	D7	Rsvd	保留
D8	DD	SD2BDPL	输入流 BDL 的低地址
DC	DF	SD2BDPU	输入流 BDL 的高地址
E0	E2	SD3CTL	输入流控制寄存器
E3	E3	SD3STS	输入流转台寄存器
E4	E7	SD3LPIB	输入流 link 在 buf 中的位置寄存器
E8	EB	SD3CBL	输入流循环 buf 的长度

偏移开始	偏移结束	名称	描述
EE	ED	SD3LVI	输入流最后一个有效的位置
EE	EF	Rsvd	保留
F0	F1	SD3FIFOD	输入流 FIFO 的大小
F2	F3	SD3FMT	输入流格式
F4	F7	Rsvd	保留
F8	FF	SD3BDPL	输入流 BDL 的低地址
FC	FF	SD3BDPU	输入流 BDL 的高地址
100	102	SD4CTL	输入流控制寄存器
103	103	SD4STS	输入流转台寄存器
104	107	SD4LPIB	输入流 link 在 buf 中的位置寄存器
108	10B	SD4CBL	输入流循环 buf 的长度
10C	10D	SD4LVI	输入流最后一个有效的位置
10E	10F	Rsvd	保留
110	111	SD4FIFOD	输入流 FIFO 的大小
112	113	SD4FMT	输入流格式
114	117	Rsvd	保留
118	11B	SD4BDPL	输入流 BDL 的低地址
11C	11F	SD4BDPU	输入流 BDL 的高地址
120	122	SD5CTL	输入流控制寄存器
123	123	SD5STS	输入流转台寄存器
124	127	SD5LPIB	输入流 link 在 buf 中的位置寄存器
128	12B	SD5CBL	输入流循环 buf 的长度
12C	12D	SD5LVI	输入流最后一个有效的位置
12E	12F	Rsvd	保留
130	131	SD5FIFOD	输入流 FIFO 的大小
132	133	SD5FMT	输入流格式
134	137	Rsvd	保留
138	13B	SD5BDPL	输入流 BDL 的低地址
13C	13F	SD5BDPU	输入流 BDL 的高地址
140	142	SD6CTL	输入流控制寄存器
143	143	SD6STS	输入流转台寄存器
144	147	SD6LPIB	输入流 link 在 buf 中的位置寄存器
148	14B	SD6CBL	输入流循环 buf 的长度
14C	14D	SD6LVI	输入流最后一个有效的位置
14E	14F	Rsvd	保留
150	151	SD6FIFOD	输入流 FIFO 的大小
152	155	SD6FMT	输入流格式
154	157	Rsvd	保留
158	15B	SD6BDPL	输入流 BDL 的低地址
15C	15F	SD6BDPU	输入流 BDL 的高地址
160	162	SD7CTL	输入流控制寄存器
163	163	SD7STS	输入流转台寄存器
164	167	SD7LPIB	输入流 link 在 buf 中的位置寄存器
168	16B	SD7CBL	输入流循环 buf 的长度

偏移开始	偏移结束	名称	描述
16C	16D	SD7LVI	输入流最后一个有效的位置
16E	16F	Rsvd	保留
170	171	SD7FIFOD	输入流 FIFO 的大小
172	177	SD7FMT	输入流格式
174	177	Rsvd	保留
178	17B	SD7BDPL	输入流 BDL 的低地址
17C	17F	SD7BDPU	输入流 BDL 的高地址

14.2.2 自定义的调试寄存器

SDO	16'h3000	16'h0,sdo_sendState	32 位	输出流的状态
	16'h3004	16'h0,sdoDma4_State	32 位	输出流 4DMA 状态机的状态
	16'h3008	16'h0,sdoDma5_State	32 位	输出流 5DMA 状态机的状态
	16'h300c	16'h0,sdoDma6_State	32 位	输出流 6DMA 状态机的状态
	16'h3010	16'h0,sdoDma7_State	32 位	输出流 7DMA 状态机的状态
SDI	16'h3014	16'h0,sdi0_RecvState	32 位	输入流 0 接受状态机的状态
	16'h3018	16'h0,sdi1_RecvState	32 位	输入流 1 接受状态机的状态
	16'h301c	16'h0,sdi2_RecvState	32 位	输入流 2 接受状态机的状态
	16'h3020	16'h0,sdi_rirbState	32 位	输入流的状态

14.3 内存数据结构

内存数据结构和 HD audio Rev 1.0 规范是完全一致的，主要包括以下几个部分，详细参见 HD audio Rev 1.0 文档。

DMA Position in Current Buffer	在当前缓冲区的 DMA 的位置
Buffer Descriptor List	缓冲区描述符列表
Buffer Descriptor List Entry	缓冲区描述符列表条目
Command Output Ring Buffer	命令的输出环形缓冲区
Response Input Ring Buffer	响应输入环形缓冲区
Codec Verb and Response Structures	编解码器的命令和响应结构
Stream Format Structure	流格式结构

第十五章 AC97 控制器

15.1 概述

在系统里一个 AC97 应用系统如图15.1所示。在一个片上系统中,与 AC97 控制器相连的有 3 部分：一是外设总线,接收来自微处理器的控制信息以及配置信息;二是 AC97 Codec, 多媒体数字信号编解码器,该解码器对 PCM 信号进行调制, 输出人耳接受的模拟声音或者把真实的声音转换为 PCM 信号,转换通过 D/A 转换器实现;三是 DMA 引擎,通过 DMA 的方式写或读 AC97 控制器内部的 FIFO, 实现 PCM 音频数据的不间断操作。DMA 是通过微处理器配置的,从处理器设定的内存区域搬运数据给 FIFO 或者把 FIFO 的数据搬运到设定的内存区域。

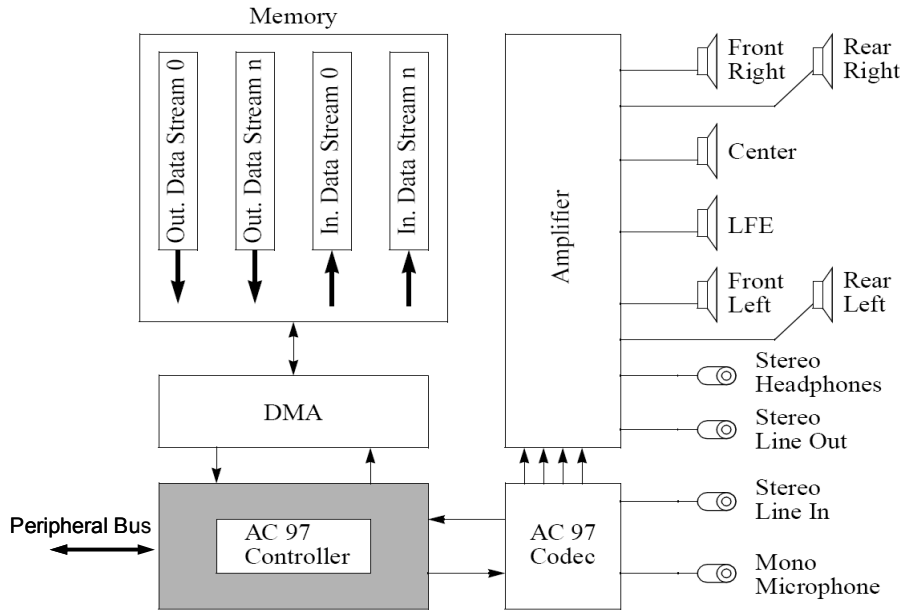


图 15.1: AC97 应用系统

15.2 AC97 控制器寄存器

本模块寄存器物理地址基址为 0x1fed0000。

寄存器名	宽度	偏移量	描述
CSR	2	0x00	配置状态寄存器
OCC0	24	0x04	输出通道配置寄存器 0
OCC1	24	0x08	保留

寄存器名	宽度	偏移量	描述
OCC2	24	0x0c	保留
ICC	24	0x10	输入通道配置寄存器
CODEC_ID	32	0x14	Codec ID 寄存器
CRAC	32	0x18	Codec 寄存器访问命令
OC0	20	0x20	输出声道 0
OC1	20	0x24	输出声道 1
OC2	20	0x28	保留
OC3	20	0x2c	保留
OC4	20	0x30	保留
OC5	20	0x34	保留
OC6	20	0x38	保留
OC7	20	0x3c	保留
OC10	20	0x40	保留
IC0	20	0x44	保留
IC1	20	0x48	保留
IC2	20	0x4c	输入声道 2
INTRAW	32	0x54	中断状态寄存器
INTM	32	0x58	中断掩膜
INTS	32	0x5c	保留

15.2.1 配置状态寄存器 (CSR)

偏移量: 0x00

复位值: 0x00000000

位域	名称	位宽	访问	描述
31:2	Reserved	30	RO	保留
1	RESUME	1	R/W	挂起, 读此位返回现在 AC97 子系统的状态 1: AC97 子系统挂起 0: 正常工作状态 在挂起状态下, 写入 1 到该位, 将会开始恢复操作。
0	RST_FORCE	1	W	AC97 冷启动 写入 1 会导致 AC97 Codec 冷启动

15.2.2 输出通道配置寄存器 (OCC)

偏移量: 0x04

复位值: 0x00004141

位域	名称	位宽	访问	描述
31:24	Reserved	10	R/W	保留
23:16	Reserved	10	R/W	保留

位域	名称	位宽	访问	描述
15:10	OC1_CFG_R	10	R/W	输出通道 1：右声道配置。
7:0	OC0_CFG_L	10	R/W	输出通道 0：左声道配置。

15.2.3 输入通道配置寄存器 (ICC)

偏移量：0x10

复位值：0x00410000

位域	名称	位宽	访问	描述
31:24	Reserved	10	R/W	保留
23:16	IC_CFG_MIC	10	R/W	输入通道 2：MIC 声道配置。
15:10	Reserved	10	R/W	保留
7:0	Reserved	10	R/W	保留

15.2.4 声道格式说明

位域	位域名称	位宽	访问	描述															
7	Reserved	1	R/W	保留															
6	DMA_EN	1	R/W	DMA 使能 1：DMA 打开 0：DMA 关闭															
5:4	FIFO_THRES	2	R/W	FIFO 门限 <table border="0" style="margin-left: 20px;"> <tr> <td></td> <td>输出通道</td> <td>输入通道</td> </tr> <tr> <td>00</td> <td>FIFO 1/4 空</td> <td>FIFO 1/4 满</td> </tr> <tr> <td>01</td> <td>FIFO 1/2 空</td> <td>FIFO 1/2 满</td> </tr> <tr> <td>10</td> <td>FIFO 3/4 空</td> <td>FIFO 3/4 满</td> </tr> <tr> <td>11</td> <td>FIFO 全空</td> <td>FIFO 全满</td> </tr> </table>		输出通道	输入通道	00	FIFO 1/4 空	FIFO 1/4 满	01	FIFO 1/2 空	FIFO 1/2 满	10	FIFO 3/4 空	FIFO 3/4 满	11	FIFO 全空	FIFO 全满
	输出通道	输入通道																	
00	FIFO 1/4 空	FIFO 1/4 满																	
01	FIFO 1/2 空	FIFO 1/2 满																	
10	FIFO 3/4 空	FIFO 3/4 满																	
11	FIFO 全空	FIFO 全满																	
3:2	SW	2	R/W	采样位数 00：10 位 10：16 位															
1	VSR	1	R/W	采样率 1：采样率可变 0：采样率固定（410KHz）															
0	CH_EN	1	R/W	通道使能 1：通道打开 0：通道关闭（或者进入节能状态）															

15.2.5 Codec 寄存器访问命令 (Codec)

偏移量：0x110

复位值：0x00000000

位域	名称	位宽	访问	描述
31	CODEC_WR	1	R/W	读 / 写选择 1: 读, 读取数据时, 先设置 CODEC_WR 为读方式, 并在 CODEC_ADR 设置欲访问的寄存器地址; 等到返回数据完成中断时再读 CODEC_DAT 寄存器读取值。 0: 写
30:23	Reserved	10	R	保留
22:16	CODEC_ADR	7	R/W	Codec 寄存器地址
15:0	CODEC_DAT	16	R/W	Codec 寄存器数据

15.2.6 中断状态寄存器 / 中断掩膜寄存器

偏移量: 0x54/510

复位值: 0x00000000

位域	名称	位宽	访问	描述
31	IC_FULL	1	R/W	输入通道 2: FIFO 满
30	IC_TH_INT	1	R/W	输入通道 2: FIFO 达到门限
29:10	Reserved	22	R/W	保留
7	OC1_FULL	1	R/W	输出通道 1: FIFO 满
6	OC1_EMPTY	1	R/W	输出通道 1: FIFO 空
5	OC1_TH_INT	1	R/W	输出通道 1: FIFO 达到门限
4	OC0_FULL	1	R/W	输出通道 0: FIFO 满
3	OC0_EMPTY	1	R/W	输出通道 0: FIFO 空
2	OC0_TH_INT	1	R/W	输出通道 0: FIFO 达到门限
1	CW_DONE	1	R/W	Codec 寄存器写完成
0	CR_DONE	1	R/W	Codec 寄存器读完成

15.2.7 中断状态 / 清除寄存器

偏移量: 0x5c

复位值: 0x00000000

位域	名称	位宽	访问	描述
31:0	INT_CLR	32	RO	屏蔽后的中断状态寄存器, 对本寄存器的读操作将清除寄存器 0x54 中的所有中断状态

15.2.8 OC 中断清除寄存器

偏移量: 0x60

复位值: 0x00000000

位域	名称	位宽	访问	描述
31:0	INT_OC_CLR	32	RO	对本寄存器的读操作将清除寄存器 0x54 中的所有 output channel 的中断状态对应的 bit[7:2]

15.2.9 IC 中断清除寄存器

偏移量：0x64

复位值：0x00000000

位域	名称	位宽	访问	描述
31:0	INT_IC_CLR	32	RO	对本寄存器的读操作将清除寄存器 0x54 中的所有 input channel 的中断状态对应的 bit[31:30]

15.2.10 CODEC WRITE 中断清除寄存器

偏移量：0x610

复位值：0x00000000

位域	名称	位宽	访问	描述
31:0	INT_CW_CLR	32	RO	对本寄存器的读操作将清除寄存器 0x54 中的中 bit[1]

15.2.11 CODEC READ 中断清除寄存器

偏移量：0x6c

复位值：0x00000000

位域	名称	位宽	访问	描述
31:0	INT_CR_CLR	32	RO	对本寄存器的读操作将清除寄存器 0x54 中的中 bit[0]

第十六章 LPC 控制器

LPC 控制器具有以下特性：

- 符合 LPC1.1 规范
- 支持 LPC 访问超时计数器
- 支持 Memory Read 、 Memory write 访问类型
- 支持 Firmware Memory Read 、 Firmware Memory Write 访问类型（单字节）
- 支持 I/O read 、 I/O write 访问类型
- 支持 Memory 访问类型地址转换
- 支持 SerIALIZED IRQ 规范，提供 17 个中断源

LPC 控制器内部的地址空间分布见表 16.1：

表 16.1: LPC 控制器地址空间分布

地址空间	名称	大小
0x1fc0,0000 - 0x1fc7,fff	LPC Boot	512KB
0x1c00,0000 - 0x1dff,fff	LPC Memory	32MB
0x1ff0,0000 - 0x1ff0,fff	LPC I/O	64KB
0x1ff1,0000 - 0x1ff1,00ff	LPC regs	256B

LPC Boot 地址空间是系统启动时处理器最先访问的地址空间。这个地址空间支持 LPC Memory 访问类型，访问 4Mbit 的 Flash，映射到 LPC 总线后的地址为 0xfff8,0000 ~ 0xffff,fff。

LPC Memory 地址空间是系统用 Memory/Firmware Memory 访问的地址空间。LPC 控制器发出哪种类型的 Memory 访问，由 LPC 控制器的配置寄存器 LPC_MEM_IS_FWH 决定。处理器发往这个地址空间的地址可以进行地址转换。转换后的地址由 LPC 控制器的配置寄存器 mem_trans 设置。地址转换方法为：

$$lpc_addr = \{mem_trans_en ? mem_trans_addr[6:0] : 7'b0, mem_addr[24:0]\}$$

处理器发往 LPC I/O 地址空间的访问按照 LPC I/O 访问类型发往 LPC 总线。地址为地址空间低 16 位。

LPC 控制器配置寄存器共有 4 个 32 位寄存器，顺序放置在 0x1ff10000 开始的 4 个字。配置寄存器的含义见下表：

表 16.2: LPC 寄存器 0

位域	名称	访问	初值	描述
31	sirq_en	R/W	0	SIRQ 使能, 高有效
23	mem_trans_en	R/W	0	LPC Memory 空间地址转换使能, 高有效
22:16	mem_trans_addr	R/W	0	LPC Memory 空间转换目标地址
15:0	timeout	R/W	0	LPC 访问超时计数 当小于 64 时硬件将当作 63 处理

表 16.3: LPC 寄存器 1

位域	名称	访问	初值	描述
31	mem_fwh	R/W	0	LPC Memory 空间访问类型控制 0: Memory 1: Firmware Memory
17:0	int_en	R/W	0	中断使能, 为 1 的位使能对应的中断源 17: timeout 16 0: sirq

表 16.4: LPC 寄存器 2

位域	名称	访问	初值	描述
17:0	int_src	R/W	0	中断状态

表 16.5: LPC 寄存器 3

位域	名称	访问	初值	描述
17	int_clr	W	-	写 1 清除超时中断状态

第十七章 SPI 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。

17.1 SPI 控制器结构

本系统集成的 SPI 控制器仅可作为主控端，所连接的是从设备。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 0x1fc00000，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。SPI 的 IO 寄存器的基地址 0x1fe70000，外部存储地址空间是 0x1e000000 - 0x1effffff 共 16MB。其结构如图 17.1 所示，由 AXI 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）。

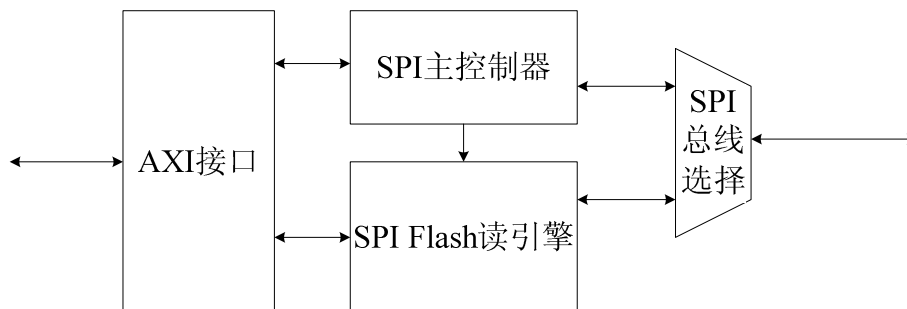


图 17.1: SPI 控制器结构

17.2 配置寄存器

表 17.1: SPI 配置寄存器列表

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

表 17.2: 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

表 17.3: 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	wfull	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

表 17.4: 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

表 17.5: 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 17.6: 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

表 17.7: 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_lio	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, 否则由硬件管理。 为 1 时, 发往 SPI Flash 读引擎的读请求可以转换为 SPI Flash 的访问; 为 0 时, SPI Flash 读引擎无法访问 SPI 总线, 将返回不确定的值。

表 17.8: 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 时对应位的 csn 线由 7:4 位控制

表 17.9: 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: 下沿采样, 间隔一个 SPI 周期 如果提高 SPI Flash 的工作频率后无法正常工作, 可以尝试将此位置 1。使用支持双 I/O 的 Flash 可以获得更好的性能。
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

17.3 接口时序

17.3.1 SPI 主控制器接口时序

17.3.2 SPI Flash 访问时序

17.4 使用指南

17.4.1 SPI 控制器的读写操作

模块初始化

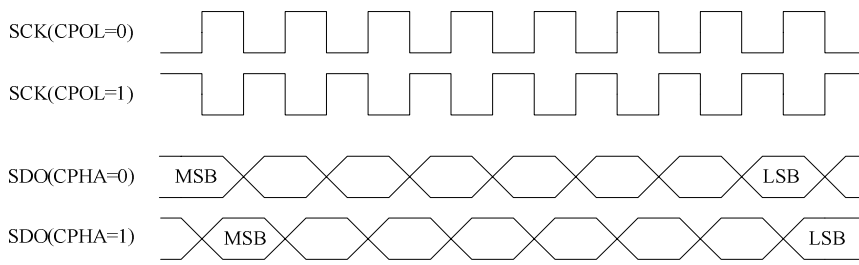


图 17.2: SPI 主控制器接口时序

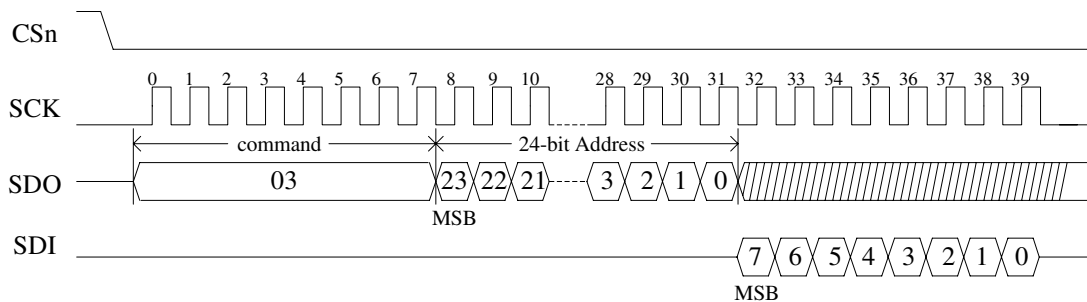


图 17.3: SPI Flash 标准读时序

- a) 停止 SPI 控制器工作，对控制寄存器 `spcr` 的 `spe` 位写 0
- b) 重置状态寄存器 `spsr`，对寄存器写入 `8'b1100_0000`
- c) 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
- d) 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。

e) 配置中断使能，`spcr` 的 `spie` 位

f) 启动 SPI 控制器，对控制寄存器 `spcr` 的 `spe` 位写 1

模块的发送 / 传输操作

a) 往数据传输寄存器写入数据

b) 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。

中断处理

a) 接收到中断申请

b) 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

c) 读或写数据传输寄存器

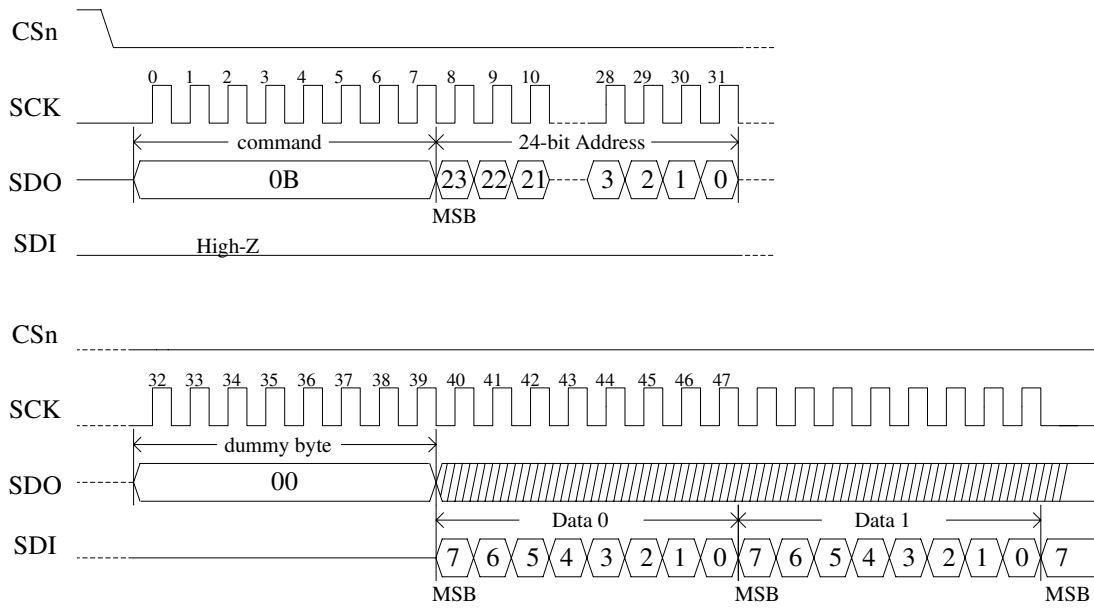


图 17.4: SPI Flash 快速读时序

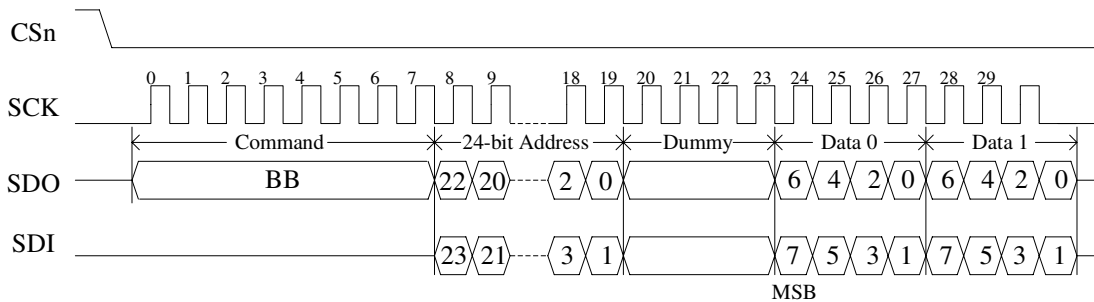


图 17.5: SPI Flash 双向 I/O 读时序

- d) 往状态寄存器 spsr 的 spif 位写 1，清除控制器的中断申请

17.4.2 硬件 SPI Flash 读

初始化

- a) 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。
- b) 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。

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。

17.4.3 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。

第十八章 I2C 控制器

18.1 概述

本芯片集成了 I2C 接口，主要用于实现两个器件之间数据的交换。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。

18.2 I2C 控制器结构

I2C 主控制器的结构如图 18.1 所示。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。

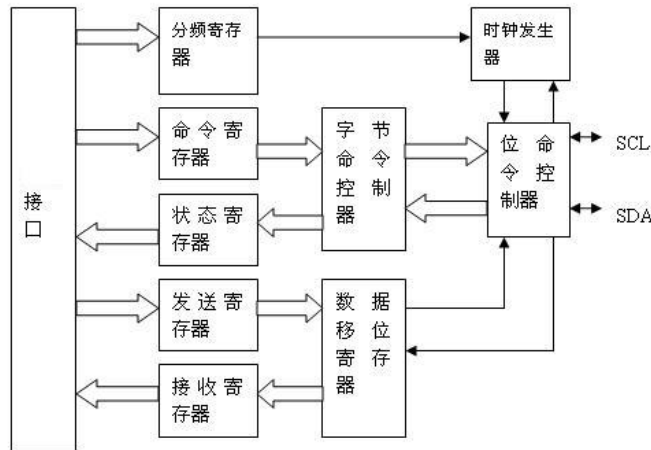


图 18.1: I2C 主控制器结构

18.3 I2C 控制器寄存器说明

本芯片集成了两个 I2C 控制器 I2C-0、I2C-1。

I2C-0 模块寄存器物理地址基址为：0x1fe90000，地址空间 16KB。

I2C-1 模块寄存器物理地址基址为：0x1fe91000，地址空间 16KB。

18.3.1 分频锁存器低字节寄存器（PRERlo）

偏移量：0x00

复位值：0xff

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

18.3.2 分频锁存器高字节寄存器（PRERhi）

偏移量：0x01

复位值：0xff

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 LPB 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： $Prescale = \text{clock}_a / (5 * \text{clock}_s) - 1$

18.3.3 控制寄存器（CTR）

偏移量：0x02

复位值：0x00

位域	名称	位宽	访问	描述
7	EN	1	RW	模块工作使能位为 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
5:0	Reserved	6	RW	保留

18.3.4 发送数据寄存器（TXR）

偏移量：0x03

复位值：0x00

位域	名称	位宽	访问	描述
7:1	DATA	7	W	存放下个将要发送的字节
0	DRW	1	W	当数据传送时，该位保存的是数据的最低位；

当地址传送时，该位指示读写状态

18.3.5 接受数据寄存器（RXR）

偏移量：0x03

复位值：0x00

位域	名称	位宽	访问	描述
7:0	RXR	8	R	存放最后一个接收到的字节

18.3.6 命令控制寄存器（CR）

偏移量：0x04

复位值：0x00

位域	名称	位宽	访问	描述
7	STA	1	W	产生 START 信号
6	STO	1	W	产生 STOP 信号
5	RD	1	W	产生读信号
4	WR	1	W	产生写信号
3	ACK	1	W	产生应答信号
2:1	Reserved	2	W	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。

18.3.7 状态寄存器（SR）

偏移量：0x04

复位值：0x00

位域	名称	位宽	访问	描述
7	RxACK	1	R	收到应答位 1 没收到应答位 0 收到应答位
6	Busy	1	R	I2c 总线忙标志位 1 总线在忙 0 总线空闲
5	AL	1	R	当 I2C 核失去 I2C 总线控制权时，该位置 1
4:2	Reserved	3	R	保留
1	TIP	1	R	指示传输的过程 1 表示正在传输数据 0 表示数据传输完毕

位域	名称	位宽	访问	描述
0	IF	1	R	中断标志位，一个数据传输完，或另外一个器件发起数据传输，该位置 1

第十九章 UART 控制器

19.1 概述

2H 集成了四个 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。

19.2 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如图 19.1 所示。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。

19.3 寄存器描述

龙芯 2H 中有 4 个 UART 接口，其功能寄存器完全一样，只是访问基址不一样。

- UART0 寄存器物理地址基址为 0x1fe80000。
- UART1 寄存器物理地址基址为 0x1fe81000。
- UART2 寄存器物理地址基址为 0x1fe82000。
- UART3 寄存器物理地址基址为 0x1fe83000。

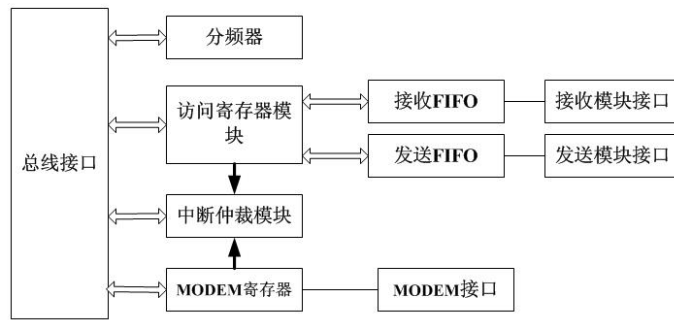


图 19.1: UART 控制器结构

19.3.1 数据寄存器 (DAT)

偏移量: 0x00

复位值: 0x00

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

19.3.2 中断使能寄存器 (IER)

偏移量: 0x01

复位值: 0x00

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

19.3.3 中断标识寄存器 (IIR)

偏移量: 0x02

复位值: 0xc1

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

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	1st	接收线路状态	奇偶、溢出或帧错误, 或打断中断	读 LSR
0	1	0	2nd	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	2nd	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	3rd	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	4th	Modem 状态	CTS, DSR, RI or DCD.	读 MSR

19.3.4 FIFO 控制寄存器 (FCR)

偏移量: 0x02

复位值: 0xc0

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

19.3.5 线路控制寄存器 (LCR)

偏移量: 0x03

复位值：0x03

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	bcb	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	spb	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

19.3.6 MODEM 控制寄存器 (MCR)

偏移量：0x04

复位值：0x00

位域	名称	位宽	访问	描述
7:5	Reserved	3	W	保留
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

位域	名称	位宽	访问	描述
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

19.3.7 线路状态寄存器 (LSR)

偏移量：0x05

复位值：0x00

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
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。

19.3.8 MODEM 状态寄存器 (MSR)

偏移量：0x06

复位值：0x00

位域	名称	位宽	访问	描述
7	CDCD	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

19.3.9 分频锁存器

偏移量: 0x00

复位值: 0x00

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

偏移量: 0x01

复位值: 0x00

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

第二十章 NAND 控制器

20.1 NAND 控制器结构描述

NAND FLASH 控制器最大支持 32GB FLASH 的容量，芯片最多支持 2 个片选和 2 个 RDY 信号，控制器支持 SLC 和 MLC 两种类型 FLASH 的操作，NAND FLASH 控制器支持系统启动，启动模式包括 ECC 模式启动和普通模式启动。

系统启动模式选择包括两种，如下表所示：

启动模式	配置	说明
ECC 模式启动	外部 SPLCK、SPLCS 下拉 NAND_RD 上拉	外部 NAND FLASH 中的第一个 page 的内容必须是原始数据经过 RS(204,188) 编码后生成的数据
普通启动	外部 SPLCK、SPLCS、 NAND_RD 全部下拉	外部 NAND FLASH 第一个 page 的数据普通原始数据

20.2 NAND 寄存器配置描述

NAND 内部的寄存器的设置如下：

地址	寄存器名称
0x1fee_0000	NAND_CMD
0x1fee_0004	ADDR_C
0x1fee_0008	ADDR_R
0x1fee_000C	NAND_TIMING
0x1fee_0010	ID_L
0x1fee_0014	STATUS & ID_H
0x1fee_0018	NAND_PARAMETER
0x1fee_001C	NAND_OP_NUM
0x1fee_0040	DMA access address

20.2.1 命令寄存器 NAND_CMD

位域	名称	访问	描述
31	DMA_REQ	R/-	非 ECC 模式下 NAND 发出 DMA 请求
30	ECC_DMA_REQ	R/-	ECC 模式下 NAND 发出 DMA 请求
29:25	STATUS	R/-	内部状态机（供测试用）
24		R/-	Reserved
23 : 20	NAND_CE	R/-	外部 NAND 芯片片选情况，四位分别对应片外四个片选，0 表示选中
19 : 16	NAND_RDY	R/-	外部 NAND 芯片 RDY 情况，对应关系和 NAND_CE 一致，1 表示准备好
15			Reserved
14	wait_ecc	R/W	为 1 表示等待 ECC 读完成（用于 ECC 读）

位域	名称	访问	描述
13	INT_EN	R/W	NAND 中断使能信号, 为 1 表示使能中断
12	RS_WR	R/W	为 1 表示写操作时候 ECC 功能开启
11	RS_RD	R/W	为 1 表示读操作时候 ECC 功能开启
10	done	R/W	为 1 表示操作完成, 需要软件清零
9	Spare	R/W	为 1 表示操作发生在 NAND 的 SPARE 区
8	Main	R/W	为 1 表示操作发生在 NAND 的 MAIN 区
7	Read status	R/W	为 1 表示读 NAND 的状态操作
6	Reset	R/W	为 1 表示 Nand 复位操作
5	read id	R/W	为 1 表示读 ID 操作
4	blocks erase	R/W	连续擦除标志, 缺省 0; 1 有效, 连续擦除块的数目由 nand_op_num 决定
3	erase operation	R/W	为 1 表示擦除操作
2	write operation	R/W	为 1 表示写操作
1	read operation	R/W	为 1 表示读操作
0	command valid	R/W	为 1 表示命令有效, 操作完成后硬件自动清零

20.2.2 页内偏移地址寄存器 ADDR_C

位域	名称	访问	描述
31:14		R/-	Reserved
13:0	Nand_Col	R/W	读、写、擦除操作起始地址页内地址 (必须以字对齐, 为 4 的倍数), 和页大小对应关系如下: 512Bytes : 只需要填充 [8:0] 2K : 需要填充 [11:0], [11] 表示 spare 区, [10:0] 表示页内偏移地址 4K : 需要填充 [12:0], [12] 表示 spare 区, [11:0] 表示页内偏移地址 8K : 需要填充 [13:0], [13] 表示 spare 区, [12:0] 表示页内偏移地址

20.2.3 页地址寄存器 ADDR_R

位域	名称	访问	描述
31:25		R/-	Reserved
24:0	Nand_Row	R/W	读、写、擦除操作起始地址页地址, 地址组成如下: {片选, 页数} 其中片选固定为 2 位, 页数根据实际的单片颗粒容量确定, 如 1M 页则为 [19:0], [21:20] 用于选择 4 片中的第几片

20.2.4 时序寄存器 NAND_TIMING

位域	名称	访问	描述
31:16		R/-	Reserved
15 : 8	Hold cycle	R/W	NAND 命令有效需等待的周期数, 缺省 4
7 : 0	Wait cycle	R/W	NAND 一次读写所需总时钟周期数, 缺省 18 ECC 模式下配置为 8' hb

20.2.5 ID 寄存器 ID_L

位域	名称	访问	描述
31:0	ID[31:0]	R/-	ID[31:0]

20.2.6 ID 和状态寄存器 STATUS & ID_H

位域	名称	访问	描述
31:24		R/-	Reserved
23:16	STATUS	R/-	NAND 设备当前的读写完成状态
15:0	ID[47:32]	R/-	ID 高 16 位

20.2.7 参数配置寄存器 NAND_PARAMETER

位域	名称	访问	描述
31:20		R/-	Reserved
29:16	op_scope	R/W	每次能操作的范围，配置如下： 1. 操作 main 区，配置为一页的 main 区大小 2. 操作 spare 区，配置为一页的 spare 区大小 3. 操作 main 加 spare 区，配置为一页的 main 区加上 spare 区大小
15		R/-	Reserved
14:12	ID_number	R/W	ID 号的字节数
11:8	外部颗粒容量大小	R/W	0: 1Gb (2K 页) 1: 2Gb (2K 页) 2: 4Gb (2K 页) 3: 8Gb (2K 页) 4: 16Gb (4K 页) 5: 32Gb (8K 页) 6: 64Gb(8K 页) 7: 128Gb(8K 页) 9: 64Mb(512B 页) a:128Mb(512B 页) b:256Mb(512B 页) c:512Mb(512B 页) d:1Gb(512B 页) (bit)
7:0		R/-	Reserved

20.2.8 操作数量寄存器 NAND_OP_NUM

位域	名称	访问	描述
31:0	NAND_OP_NUM	R/W	NAND 读写操作 Byte 数 (必须以字对齐, 为 4 的倍数); 擦除为块数

20.2.9 DMA 读写数据寄存器 DMA_ADDRESS

位域	名称	访问	描述
31 : 0	DMA_DATA	R/W	NAND 数据 DMA 传输端口。其地址应配置到 DMA 控制器的描述符中

20.3 NAND ADDR 说明

以 2K 页的 NAND flash 为例，定义如下：

每一页的 main 区大小为 2KB，spare 区大小为 64B

main_op = NAND_CMD[8];

spare_op = NAND_CMD[9];

addr_in_page = A11, A10.. A2, A1, A0=ADDR_C

page_number = ... A30,A29,A28,A27 ... A13,A12= ADDR_R

NAND flash 的 main 区总容量计算公式为

$$N = 2^{(ADDR_C-1)} \times 2^{(ADDR_R)} \times 8bit = 2K \times 2^{(ADDR_R)} \times 8bit$$

NAND 地址空间示例如下表：

	I/O	0	1	2	3	4	5	6	7
Column1	1st Cycle	A0	A1	A2	A3	A4	A5	A6	A7
Column2	2nd Cycle	A8	A9	A10	A11	*L	*L	*L	*L
Row1	3rd Cycle	A12	A13	A14	A15	A16	A17	A18	A19
Row2	4th Cycle	A20	A21	A22	A23	A24	A25	A26	A27
Row3	5th Cycle	A28	A29	A30	A31	A32	A33

(注：2K 页的 1Gb 容量 NAND flash 对应的 Row 最大值为 A27，发送地址给 NAND flash 时只用发 Column1 2 和 Row1 2，不用发 Row3。配置 NAND 参数时，注意不要配错型号，否则可能会读不出数据并且控制器会死等)

对系统板上 NAND 颗粒来说，如果仅仅操作 spare 区，A11=1 是唯一标志。所以软件配置内部寄存器时，需要配置 A11 和 spare_op 均为 1(见 Examples5)，错误的示例见 Examples2。

对系统板上 NAND 颗粒来说，如果仅仅操作 main 区，A11=0 是唯一标志。；所以软件配置内部寄存器时，需要配置 A11 和 spare_op 均为 0 (见 Examples1)，错误的示例见 Examples4。

对系统板上 NAND 颗粒来说，如果操作 main+spare 区，A11 可以为 0 (见 Examples3)；也可以为 1 (见 Examples6)。

Examples1: (非 ECC 模式下。NAND 颗粒中一个 page 的数据只能位于 0x0-0x83f，第一个 op 表示读写开始的数据，接下来的 op 表示随后的读写数据；NO_op 表示不能被本次 NAND 配置读写的数据)

(spare_op =0 & main_op =0) equal to (spare_op =0 & main_op =1); ADDR_C =0x30

Data in a page	0	0x30	...	0x7ff	0x800	0x830	0x83f
Page 0	NO_op	op	op	op	NO_op	NO_op	NO_op
Page 1	op	op	op	op	NO_op	NO_op	NO_op
Page 2	op	op	op	op	NO_op	NO_op	NO_op

Examples2:

spare_op=1 & main_op=0; ADDR_C = 0x30 (配置出错!! 开始操作不在 spare 区, 下图是可能的错误访问顺序)

Data in a page	0	0x30	...	0x7ff	0x800	0x830	0x83f
Page 0	NO_op	op	op	op	Op	op	op
Page 1	NO_op	NO_op	NO_op	NO_op	Op	op	op
Page 2	NO_op	NO_op	NO_op	NO_op	Op	op	op
Page 3	NO_op	NO_op	NO_op	NO_op	Op	op	op

Examples3:

spare_op = 1 & main_op =1; ADDR_C = 0x30

Data in a page	0	0x30	...	0x7ff	0x800	0x830	0x83f
Page 0	NO_op	op	op	op	op	op	op
Page 1	op	op	op	op	op	op	op
Page 2	op	op	op	op	op	op	op

Examples4:

(spare_op=0 & main_op=0), (equal to spare_op=0 & main_op=1); ADDR_C =0x830: (配置出错!! 开始操作在 spare 区, 下图是可能的错误访问顺序)

Data in a page	0	0x30	...	0x7ff	0x800	0x830	0x83f
Page 0	NO_op	NO_op	NO_op	NO_op	NO_op	NO_op	NO_op
Page 1	NO_op	op	op	op	op	NO_op	NO_op
Page 2	op	op	op	op	op	NO_op	NO_op
Page 3	op	op	op	op	op	NO_op	NO_op

Examples5:

spare_op = 1 and main_op =0; ADDR_C = 0x830

Data in a page	0	0x30	...	0x7ff	0x800	0x830	0x83f
Page 0	NO_op	NO_op	NO_op	NO_op	NO_op	op	op
Page 1	NO_op	NO_op	NO_op	NO_op	op	op	op
Page 2	NO_op	NO_op	NO_op	NO_op	op	op	op

Examples6:

spare_op = 1 & main_op =1; ADDR_C = 0x830

Data in a page	0	0x30	...	0x7ff	0x800	0x830	0x83f
Page 0	NO_op	NO_op	NO_op	NO_op	NO_op	op	op
Page 1	op	op	op	op	op	op	op
Page 2	op	op	op	op	op	op	op
Page 3	op	op	op	op	op	op	op

512B 页大小的 NAND flash 和 2KB 页大小配置类似，但在以下几个地方会有不同，需要注意：

每一页的 main 区大小为 512B，spare 区大小为 16B。其中 main 区分为两个 256B 区，每个 256B 区通过 A0 A7 来寻址。读写操作时，通过发送命令 0x00、0x01 和 0x50 来选择是在哪个 256B 区或者 spare 区（软件不必关心，硬件自动选择，如配置 NAND 控制器写 0x100 时，硬件会自动发送到高 256B 区）。

发送地址命令顺序如下：

	I/O	0	1	2	3	4	5	6	7
Column1	1st Cycle	A0	A1	A2	A3	A4	A5	A6	A7
Row1	2nd Cycle	A9	A10	A11	A12	A13	A14	A15	A16
Row2	3rd Cycle	A17	A18	A19	A20	A21	A22	A23	A24
Row3 ¹	4th Cycle	A25	A26	*L	*L	*L	*L	*L	*L

4K/8K 页大小的配置和 2K 页配置一样，4K 页的 main 区大小为 4KB，spare 区大小为 128B；8K 页的 main 区大小为 8KB，spare 区大小为 640B。都需要发送五次地址命令。

20.4 NAND-flash 读写操作举例

命令寄存器的 ‘command valid’ 位不能与其他读写使能位同时置位，要先设置好要进行的操作，最后才置 ‘command valid’ 位。

例如：现在要读 Main 区的数据，那么操作分成以下两步：

- a. 先 NAND_CMD = 0x102
- b. 再 NAND_CMD = 0x103

20.5 NAND ECC 说明

硬件集成 ECC 功能，ECC 采用 RS(204,188) 方法进行编码和解码，在配置软件过程中需要注意以下几点：

1. 每次读写 NAND 的时候，推荐配置 PAGE 的页内部地址（ADDR_C）为 0；
2. NAND 每个 PAGE 有 2048Bytes，采用 RS(204,188) 方式编解码后，只会用到前面的 2040Bytes，会有 8 个 Bytes 不用；采用 ECC 后 NAND 利用率为 188/204；
3. 在配置操作数的时候，如果每次操作一个页面，请配置 NAND 里面的 op_num 为 204 的倍数 (byte 为单位)；在配置 DMA 控制器时候，操作数为 47(188/4) 的倍数 (word 为单位)。

¹Nand flash 容量为 64Mb、128Mb 和 256Mb 时，对应的最大列地址 ADDR_R 分别为 A22、A23 和 A24，只用发送三次地址命令 Column1 和 Row1~2，不用发送 Row3；容量为 512Mb 和 1Gb 时，需要发送 Row3

4. ECC 操作和 OOB 操作可以分开，比如对一个页完成 ECC 读 / 写后可以对其 OOB 进行操作。

可以在 ECC 操作完成后通过普通方式读回所有内容，包括原始数据和 ECC 校验增加的数据（此时配置操作数 op_num 和 DMA 相同）。

校验能力说明：最多可以纠错 8 个 Bytes，这些 Bytes 内部出错的位数可以是 1-8 个。

第一行数据共出错 64bit，恰好是 8 个 Bytes，可以纠错；最后一行数据虽然错 9bits，分散在 9Bytes 中，无法纠错。

原始数据 (204Bytes)	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	...	
数据 1	00	00	00	00	00	00	00	00	ff	ff	ff	ff	...	可纠错
数据 2	1	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	...	可纠错
数据 3	fe	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	...	可纠错
数据 4	1	2	3	4	5	6	07	08	ff	ff	ff	ff	...	可纠错
数据 5	fe	fe	fe	fe	fe	fe	fe	fe	fe	ff	ff	ff	...	不可纠错

20.6 NAND 启动说明

NAND 启动支持 512B、2K 页、4K 页和 8K 页 NAND flash 启动。根据 FLASH 的容量大小和页大小配置好 NAND_WR 和 SPLSDO 的上下拉电阻（需要对应好，否则可能会出现死等的情况）。

NAND 启动时会取出第 0 页开始的 1K 字节，进行芯片的初始化。如果是 512B 页则会读两页，如果是 2K 页或者更大，则会读取 1K 字节。故系统初始化代码需要放在从第 0 页 0 字节开始。

NAND 启动时可以通过 NAND_RD 的上下拉来选择 ECC 和非 ECC 方式启动。两者的区别在于：ECC 方式采用了 RS(204,188) 编码，ECC 启动时读取 1020 字节，有效数据为 940 字节，其余为编码数据。（注：ECC 启动时不支持 512B 页。）

第二十一章 PWM 控制器

21.1 概述

2H 芯片里实现了四路脉冲宽度调节 / 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。其中 PWM0 和 PWM1 只能作为输出，PWM2/PWM3 的方向可配置（见 gpiocfg）。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。四路 PWM 控制器系统的基地址具体如下：

表 21.1: 四路 PWM 控制器

名称	基地址 (Base)
PWM0	0x1fea0000
PWM1	0x1fea0010
PWM2	0x1fea0020
PWM3	0x1fea0030

21.2 寄存器描述

每路控制器共有五个寄存器，具体描述如下：

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 21.3: PWM 控制寄存器设置

位域	名称	访问	初值	描述
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
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

位域	名称	访问	初值	描述
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

21.3 功能说明

21.3.1 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 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。

21.3.2 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。

21.3.3 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）：

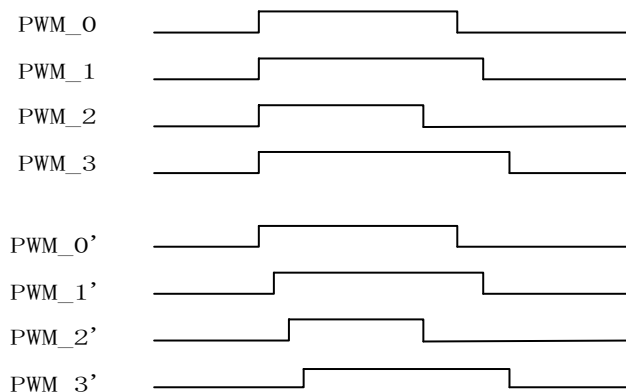


图 21.1: 防死区功能例图

第二十二章 HPET 控制器

22.1 概述

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。这个计时器以固定的频率 (125MHz) 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。

22.2 寄存器描述

下表列出了 HPET 的寄存器:

表 22.1: HPET 寄存器列表

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
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，硬件不保存这些寄存器的值。

表 22.2: General Capabilities and ID Register

位域	名称	描述	访问
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, 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

表 22.3: General Configuration Register

位域	名称	描述	访问
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

表 22.4: General Interrupt Status Register

位域	名称	描述	访问
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	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: 如果是电平触发模式: 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。 如果边沿触发模式: 软件将忽略这位。软件通常往这位写 0。	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。

表 22.5: Main Counter Value Register

位域	名称	描述	访问
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

表 22.6: Timer N Configuration and Capabilities Register

位域	名称	描述	访问
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	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 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。	R/W
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		

表 22.7: Timer N Comparator Value Register

位域	名称	描述	访问
63 : 32	Reserved		

位域	名称	描述	访问
31 : 0	Tn_Com_VAL	<p>Tn.Comparator value (N 为 0-2): 定时器比较器的值;</p> <p>当对应的定时器配置为非周期性模式时:</p> <ul style="list-style-type: none"> • 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 <p>若对应的定时器配置为周期性模式时:</p> <ul style="list-style-type: none"> • 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 , <ol style="list-style-type: none"> 1. 那么当主计时器的值为 0x123h 时, 产生中断; 2. 比较器的值被硬件修改为 0x246h ; 3. 当主计时器的值达到 0x246h 时, 产生另外一个中断; 4. 比较器的值被硬件修改为 0x369h 。 • 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 。 	R/W

第二十三章 DMA 控制器

23.1 DMA 控制器结构描述

龙芯 2H 中 DMA 用来实现内存与 APB 设备之间数据搬移，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：

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

本 DMA 控制器是一个基于 AXI 总线的单通道、可配置的 DMA 控制器 IP 核，主要功能就是在芯片上集成了 DMA 功能，专门负责在内存与 APB 设备间搬运数据。本 DMA 控制器限定为以字 (4Byte) 为单位的数据搬运。CPU 通过一个通用寄存器 (dma_order) 向 DMA 发命令。DMA 根据命令内容，从内存读取描述符启动 DMA 直接操作，或者将 DMA 状态写入内存，或者停止 DMA。

23.2 DMA 控制器与 APB 设备的交互

在 2H 中，使用 DMA 的 APB 设备包括 NAND 和 AC97，每个设备都有单独的 DMA 控制器。在 DMA 控制器的设计中有一小部分代码是用来处理 AC97 写操作。如果是写 AC97 的操作，需要在配置 DMA 描述符时，将 DMA_DADDR[31] 配置为 1，将 DMA_DADDR[30:28] 配置为需要的模式。

23.3 DMA 控制器

23.3.1 下一个描述符地址寄存器 (DMA_ORDER_ADDR)

偏移地址：0x0

复位值：0x00000000

位域	名称	位宽	访问	描述
31:1	dma_order_addr	31	R/W	存储器内部下一个描述符地址寄存器
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。

23.3.2 内存地址寄存器 (DMA_SADDR)

偏移地址：0x4

复位值：0x00000000

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

说明：DMA 操作分为：从内存中读数据，保存在 DMA 控制器的缓存中，由 APB 发请求来访问 DMA 缓存中的数据，该寄存器指定了读 ddr2 的地址；从 APB 设备读数据保存在 DMA 缓存中，当 DMA 缓存中的字超过一定数目，就往内存中写，该寄存器指定了写内存的地址。

23.3.3 设备地址寄存器 (DMA_DADDR)

偏移地址：0x8

复位值：0x00000000

位域	名称	位宽	访问	描述
31		1	R/W	AC97 写使能，“1”表示是写操作
30		1	R/W	0:mono 1: 2 stero
29:28		2	R/W	AC97 写模式, 0: 1byte,1: 2byte,2: 4byte
27:0	dma_daddr	28	R/W	DMA 操作的 APB 设备地址

说明：从内存中读数据，保存在 DMA 控制器的缓存中，由 APB 发请求来访问 DMA 缓存中的数据，该寄存器指定了写 APB 设备的地址；从 APB 设备读数据保存在 DMA 缓存中，当 DMA 缓存中的字超过一定数目，就往内存中写，该寄存器指定了读 APB 设备的地址。

23.3.4 长度寄存器 (DMA_LENGTH)

偏移地址：0xc

复位值：0x00000000

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。

23.3.5 间隔长度寄存器 (DMA_STEP_LENGTH)

偏移地址：0x10

复位值：0x00000000

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。

23.3.6 循环次数寄存器 (DMA_STEP_TIMES)

偏移地址：0x14

复位值：0x00000000

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。

23.3.7 控制寄存器 (DMA_CMD)

偏移地址：0x18

复位值：0x00000000

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：

状态	值	描述
Write_idle	4'h0	写状态正处于空闲状态
W_dds_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_dds	4'h2	内存接收了 dma 写请求，但是还没有执行完写操作
Write_dds_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：

状态	值	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_dds_wait	4'h5	dma 向内存发出读数据请求，等待内存应答
Read_dds	4'h6	内存接收 dma 读数据请求，正在执行读数据操作
Read_dds_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

第二十四章 电源管理模块

24.1 概述

龙芯 2H 电源管理模块提供系统功耗管理实现机制。支持 Advanced Configuration and Power Interface, Version 4.0a(ACPI), 提供相应的功耗管理功能。

- 系统休眠与唤醒，支持 ACPI S3（待机到内存），ACPI S4（待机到硬盘），ACPI S5（软关机），并且支持电源失效检测和自动系统恢复。支持多种唤醒方式（USB，GMAC，电源开关等）
- 动态性能功耗控制，支持处理器核 DVFS 控制，支持动态关闭媒体解码协处理器电源。
- 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。

24.2 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。

电源管理控制的物理基地址为：0x1fef0000 — 0x1fef7fff (32KB)

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

24.2.1 PMCON_SOC : SOC General PM Configuration Register

地址偏移	电压域	属性
0x00	SOC	R/W, RO

位域	描述
25	PWRBTN_LVL - RO 该位指示当前 PWRBTNn 信号状态。
24	PWRTYP - RO 该位指示供电模式 1: AC (适配器) 0: Battery (电池)
23:22	保留

位域	描述
21:18	MS_DELAY - R/W 媒体解码器上电延时，确保上电电压达到稳定值，0b0000-0b1111（0-15us）
17	MS_DELAY_SEL - R/W 该位选择媒体上电等待机制 1：内部 ACK 信号确定上电状态 0：延时方式（MS_DELAY 确定延时值）
16	MS_CNT_EN - R/W 媒体子系统功耗管理开关 0：媒体功耗管理无效 1：使能媒体功耗管理
15:14	DVFS_CNT_1 (DVFS Control 1) - R/W DVFS 转换时何时进行保护操作 00：不进行保护操作。 01：电压转换阶段进行保护操作。 10：频率变换是进行保护操作。 11：电压和频率转换阶段都进行保护操作。
13:12	DVFS_CNT_0 (DVFS Control 2) - R/W DVFS 转换时保护操作类型 00：停时钟。 01：使用备份时钟。 10：保留。 11：对时钟设置不做操作。
11	DVFS_EN - R/W 该位使能处理器核 DVFS 功能 0：无效 DVFS 使能 1：使能 DVFS 功能
10	PWROK_MASK_EN - R/W 如果有效，在 DVFS 电压转换阶段时屏蔽 PWROK 信号
9	DVFS_FORCE_BYPASS - R/W 强制使用备份时钟，用户配置 DVFS 时将其写 0，其它情况下不应该使用该位。
7:0	TEMP_NOW - RO. CPU 内部温度传感器温度值，第 7 为为溢出位。

24.2.2 PMCON_RESUME : RESUME General PM Configuration Register

地址偏移	电压域	属性
0x04	RESUME	R/W，RO，R/WC

位域	描述
31:9	保留
8	USBPHY_LP_EN - R/W 进入 G2/S5 状态时使能 USB PHY 进入低功耗状态。
7	USB_GMAC_OK - R/W 如果 RSMRSTn 有效过，该位为 0，表示 USB 和 GMAC 没有配置，不能唤醒系统。重新上电后由系统配置该位。
6	CTT_STS - R/WC 系统在 S0 状态时发生 temperature trip，系统进入 G2/S5 状态，该位为重新上电后系统检测记录事件状态
5	CTT_EN - R/W 使能 temperature trip 保护机制

位域	描述
4	LID_OPEN - RO 显示屏状态检测位 1: 显示屏打开 0: 显示屏合上
3	保留
2	SRS (System Reset Status) - R/WC 0: SYS_RESETE _n 没有被按下 1: SYS_RESETE _n 被按下过, 系统重新复位后需检查此位并作出相应清除操作。
1	PWROK_FLR (PWROK Failure) - R/WC 当系统在 S0 状态时, PWROK 信号变无效该位置 1, 软件通过写 1 将该位清除。
0	DRAM_INIT - R/W 该位不影响硬件功能, PMON 在进行 DRAM 初始化前将该位置 1, 结束 DRAM 初始化后将该位写 0, 软件可通过此位检查 DRAM 初始化是否被打断过。

24.2.3 PMCON_RTC : RTC General PM Configuration Register

地址偏移	电压域	属性
0x08	RTC	R/W, R/WC

位域	描述
31:9	保留
8	WOL_EN - R/W enable wake on LAN don't shut off SLP_LANn, use with WOL_BAT_EN 使能 wake on LAN, 与 WOL_BAT_EN 共同使用
7	WOL_BAT_EN - R/W 如果系统使用电池供电, 如果该位置 1, 使能 WOL, 如果该位置 0, 不论 WOL_EN, WOL 无效
6:5	S3_ASSERTION_WIDTH - R/W 这 2 bit 值代表 S3n 信号从有效到重新无效的最小时间间隔。 11: 2s 10: 125ms 01: 2ms 00: 120us
4:3	S4_ASSERTION_WIDTH - R/W 这 2 bit 值代表 S4n 信号从有效到重新无效的最小时间间隔。 11: 4 seconds 10: 3 seconds 01: 2 seconds 00: 1 seconds
2	S4_ASSERTION_EN - R/W 0: S4n 信号从有效到重新无效的间隔为 64RTC 周期。 1: S4n 信号从有效到重新无效的间隔为 S4_ASSERTION_WIDTH 决定。
1	PWR_FLR (Power Failure) - R/WC. 该位在 RTC 域, 只能被 RTC_RST _n 复位。 如果置 1 表示系统发生过电源失效 (进入 G3 状态), 即除 RTC 以外所有供电失效过 (RSMRST _n 有效过), 软件通过写 1 将该位清除。

位域	描述
0	<p>AFTERG3_EN - R/W</p> <p>该位决定系统进入 G3 状态后重新供电后的动作。</p> <p>0：系统在供电恢复后将自动回复到 S0 状态。</p> <p>1：系统将恢复到 S5 状态，如果发生电源失效时系统在 S4 状态，重新供电后系统恢复到 S4 状态。</p> <p>该位会被 power button override 和 thermal trip 事件置 1。</p>

24.2.4 PM1_STS : Power Management 1 Status Register

地址偏移	电压域	属性
0x0C	RESUME/RTC/SOC	R/WC

位域	描述
31:16	保留
15	<p>WAK_STS (Wake Status) - R/WC - Resume</p> <p>0：软件写 1 将该位清除。</p> <p>1：如果系统从任何一个休眠状态被唤醒事件唤醒，硬件将该位置 1。</p>
14	<p>PCIEXP_WAKE_STS - R/WC - Resume</p> <p>1：PCIe 唤醒事件发生</p> <p>0：软件写 1 将该位清除</p>
13:12	保留
11	<p>PRBTNOR_STS (Power Button Override Status) - R/WC - RTC</p> <p>0：软件写 1 将该位清除</p> <p>1：当 power button override 发生时，该位置 1，系统无条件进入 G2/S5 状态，同时将 AFTERG3_EN 位置 1。</p>
10	<p>RTC_STS (RTC Status) - R/WC - Resume</p> <p>0：软件写 1 将该位清除</p> <p>1：当 RTC 产生 alarm 时该位置 1。此外当 RTC_EN 有效时，该位产生唤醒事件。</p>
9	保留
8	<p>PWRBTN_STS (Power Button Status) - R/WC - Resume</p> <p>0：软件写 1 将该位清除。Thermal trip 会清除该位。</p> <p>1：当按下 PWRBTNn 保持 16ms 以上（4s 以下）时，该位会置 1。</p> <p>在 S0 状态时，当 PWRBTN_EN 和 PWRBTN_STS 同时有效时，将产生中断。</p> <p>在 S3-S5 任何休眠状态时，如果 PWRBTN_STS 置位，系统将恢复。</p>
7:5	保留
4	<p>BM_STS (Bus Master Status) - R/WC - SOC</p> <p>0：软件写 1 将该位清除</p> <p>1：该位监视 bus master 上的非一致性访问。</p>
3:1	保留
0	<p>TMROF_STS (PM Timer Overflow Status) - R/WC - SOC</p> <p>0：软件写 1 将该位清除</p> <p>1：当 24bit 计数器（周期 8ns）的最高位翻转时，该位置 1，该计时功能推荐使用 HPET 完成。</p>

24.2.5 PM1_EN : Power Management 1 Enable Register

地址偏移	电压域	属性
0x10	RESUME/RTC/SOC	R/W

位域	描述
31:15	保留
14	PCIEXP_WAKE_DIS - R/W - resume 当置位时, 不产生 PCIE 唤醒事件, 但是该位的值不影响 PCIEXP_WAKE_STS 的值。
13:11	保留
10	RTC_EN (RTC Event Enable) - R/W - RTC RTC 唤醒和中断使能
9	保留
8	PWRBTN_EN (Power Button Enable) - R/W - Resume PWRBTN 中断事件产生使能, 该位不影响 PWRBTN 唤醒事件产生。
7:1	保留
0	TMROF_EN (PM Timer Overflow Enable) - R/W - SOC 如果该位置位, TMROF_STS 将产生中断。

24.2.6 PM1_CNT : Power Management 1 Control Register

地址偏移	电压域	属性
0x14	RESUME/RTC/SOC	R/W

位域	描述
31:14	保留
13	SLP_EN (Sleep Enable) - R/W - Resume 该位写 1 将会使系统进入 SLP_TYP 声明的休眠状态, 进入相关休眠状态后该位自动恢复为 0。
12:10	SLP_TYP (Sleep Type) - R/W - RTC 该 3bit 表示系统的休眠状态。 000: 表示 S0 状态。 001: Reserved. 010: Reserved. 011: Reserved. 100: Reserved. 101: Suspend-to-RAM. S3n 信号有效, 进入 S3 状态。 110: Suspend-to-Disk. S3n, S4n 信号有效, 进入 S4 状态。 111: Soft Off. S3n, S4n, S5n 信号有效, 进入 S5 状态。
9:2	保留
1	BM_RLD (Bus master Reload) - R/W - SOC 如果置位, 使能 BM_STS 产生打断事件, 使得系统从 C2/C3 状态恢复。
0	INT_EN - R/W - SOC 中断使能开关, 使能电源管理控制器中断信号的产生。

24.2.7 PM1_TMR : Power Management 1 Timer

地址偏移	电压域	属性
0x18	SOC	RO

位域	描述
31:24	保留
23:0	TMR_VAL (Timer Value) - RO. 计数器计数, 周期 8ns。当 23 位翻转时, 置位 TNROF_STS 位。 推荐使用 HPET。

24.2.8 P_CNT : Processor Control Register

地址偏移	电压域	属性
0x1C	SOC	R/W, RO

位域	描述
31:5	保留
4	THTL_EN - R/W 当系统处于 C0 状态时, 使能该位可以通过 clock throttling 对 CPU 时钟进行控制。
3:1	THTL_DTY - R/W 该 3-bit 确定 throttling 的百分比, throttling 周期是 1024 APB 时钟。 000: no throttling 001: 87.5% throttling 010: 75% throttling 011: 62.5% throttling 100: 50% throttling 101: 37.5% throttling 110: 25% throttling 111: 12.5% throttling
0	保留

24.2.9 P_LVL2 : Processor LVL_2 Register

地址偏移	电压域	属性
0x20	SOC	RO

位域	描述
31:0	读该寄存器是 CPU 进入 C2 状态直到 break event 发生, 写该寄存器无效。

24.2.10 P_LVL3 : Processor LVL_3 Register

地址偏移	电压域	属性
0x24	SOC	RO

位域	描述
31:0	读该寄存器是 CPU 进入 C3 状态直到 break event 发生, 写该寄存器无效。

24.2.11 GPE0_STS : General Purpose Event0 Status Register

地址偏移	电压域	属性
0x28	RESUME	R/WC

位域	描述
31:16	保留
15:10	USB[6:1]_STS - R/WC. 只有第 10 位有意义，15:11 位暂无意义。 0：软件写 1 将该位清除。 1：当 USB 发生 wake 事件时这些位被置位，当 USBx_EN 位有效时，产生唤醒事件或中断。
9	保留
8	RI_STS - R/WC. 0：软件写 1 将该位清除。 1：当 RIn 信号有效时被置位。
7	BATLOW_STS - R/WC. 0：软件写 1 将该位清除。 1：当 BATLOWn 信号有效时被置位 如果 BATLOW_EN 有效，BATLOW_STS 将产生中断。该位不产生唤醒事件。
6	GMAC1_STS - R/WC. 0：软件写 1 将该位清除。 1：当 GMAC1 发生 wake 事件时这些位被置位，当 GMAC1_EN 位有效时，产生唤醒事件或中断。
5	GMAC0_STS - R/WC. 0：软件写 1 将该位清除。 1：当 GMAC0 发生 wake 事件时这些位被置位，当 GMAC0_EN 位有效时，产生唤醒事件或中断。
4	LID_STS - R/WC. 0：软件写 1 将该位清除。 1：当 LID_EN 位有效时，产生唤醒事件。
3	CTW_STS - R/WC. CPU thermal warning 发生
2	CTA_STS - R/WC. CPU thermal alert 发生
1	PWRSWITCH_STS - R/WC. 供电状态发生变化，PWRTYP 变化。该位会产生中断。
0	保留

24.2.12 GPE0_EN : General Purpose Event0 Status Register

地址偏移	电压域	属性
0x2C	RESUME/RTC	R/W

位域	描述
31:16	保留
15:10	USB[6 :1]_EN - R/W. 0：无效 1：使能 USBx_STS 产生唤醒事件，当回到 S0 将产生中断信号。

位域	描述
9	保留
8	RI_EN - R/W - RTC 0: 无效 1: 使能 RI _n _STS 产生唤醒事件, 当回到 S0 将产生中断信号。
7	BATLOW_EN - R/W - RTC 0: 无效 1: 使能 BATLOW _n 产生中断事件。
6	GMAC1_EN - R/W - RTC 0: 无效 1: 使能 GMAC1_STS 产生唤醒事件, 当回到 S0 将产生中断信号。
5	GMAC0_EN - R/W - RTC 0: 无效 1: 使能 GMAC0_STS 产生唤醒事件, 当回到 S0 将产生中断信号。
4	LID_EN - R/W. 0: 无效 1: 使能 LID_STS 产生唤醒事件, S0 状态时将产生中断信号。
3	CTW_EN - R/W 使能 CPU THERMAL WARNING 产生中断。
2	CTA_EN - R/W 使能 CPU THERMAL ALERT 产生中断。
1	PWRSWITCH_EN - R/W 使能 PWRSWITCH_STS 产生中断。
0	LID_POL - R/W 该位设置 LID 的极性。 0: LID 为低时置位 LID_STS bit. 1: LID 为高时置位 LID_STS bit.

24.2.13 RST_CNT : Reset Control Register

地址偏移	电压域	属性
0x30	SOC	R/W

位域	描述
31:2	保留
1	WD_EN - R/W Watch dog 功能使能
0	OS_RST - R/W 软件写该位使系统复位。

24.2.14 WD_SET : Watch Dog Set Register

地址偏移	电压域	属性
0x34	SOC	WO

位域	描述
31:1	保留
0	写该位将重填 watch dog 计数器

24.2.15 WD_Timer : Watch Dog Timer Register

地址偏移	电压域	属性
0x38	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。

24.2.16 DVFS_CNT : Dynamic Voltage Frequency Scaling Control Register

地址偏移	电压域	属性
0x3C	SOC	R/W

位域	描述
28	VID_UPDATE_EN R/W 如果不想改变电压 VID 值，将该位置 0（即只进行频率转换）
27:18	V_DELAY - R/W 该部分值表示 DVFS 电压转换时的延时值，该延时值期间可进行保护操作。 范围：0 - 1000us。
17:9	FEQ_TGT - R/W 修改 CPU 分频器分频值。 17: 保留 16:update en 15:DIV enable 14:9 DIV_number
8:3	VID_TGT - R/W 设置目标电压值（目标 VID）
2	DVFS_POL - R/W 写 1 表示需提升性能，升压升频。 写 0 表示降低性能节约功耗，降压降频。
1	DVFS_START - R/W 写 1 开始一个 DVFS 转换。
0	VID_VALID - R/W 1：输出 VID 值有效 0：输出 VID 高阻，VID 由外部上下拉决定。

24.2.17 DVFS_STS : Dynamic Voltage Frequency Scaling Status Register

地址偏移	电压域	属性
0x40	SOC	R/W, RO

位域	描述
18:10	FEQ_STS - R/W 当前分频器值
9:4	VID_STS - R/W 当前 VID 值
3:2	保留

位域	描述
1	CPU_DVFS_STS(DVFS status) - RO. 0: DVFS 控制器空闲, 可进行 DVFS 操作 1: 系统正在进行 DVFS 转换。
0	保留

24.2.18 MS_CNT : Media Subsystem Power Control Register

地址偏移	电压域	属性
0x44	SOC	R/W, RO

位域	描述
31:5	保留
4:3	MS_TGT - R/W 设置媒体处理器的目标状态 00: 工作 01: (保留) 10: 停时钟 11: 关闭电源
2:1	MS_STS - RO 0: 控制器空闲, 可以进行操作。 1: 媒体控制器正在进行操作。
0	MS_RUNNING_STS - RO 媒体处理器的状态 00: 工作 01: (保留) 10: 停时钟 11: 关闭电源

24.2.19 MS_THT : Media Subsystem Throttling Register

地址偏移	电压域	属性
0x48	SOC	R/W

位域	描述
31:5	保留
4	MS_THT_EN - R/W. 使能媒体处理器的时钟控制

位域	描述
3:1	MS_THT_DTY - R/W. 该 3-bit 确定 throttling 的百分比，throttling 周期是 1024 APB 时钟。 000: no throttling 001: 87.5% throttling 010: 75% throttling 011: 62.5% throttling 100: 50% throttling 101: 37.5% throttling 110: 25% throttling 111: 12.5% throttling
0	保留

24.2.20 THSENS_CNT : CPU Thermal Sensor Control Register

地址偏移	电压域	属性
0x4C	SOC	R/W

位域	描述
31:27	保留
26:20	WARNING_TMP - R/W CPU 温度超过该值，如果被使能，将产生中断。如果温度位降低到该值以下，不会重复产生中断。 推荐操作：系统采取降低功耗操作。
19:13	ALERT_TMP - R/W CPU 温度超过该值，如果被使能，将产生中断。如果温度位降低到该值以下，不会重复产生中断。 推荐操作：采取正常关闭系统操作
12:6	TRIP_TMP - R/W CPU 温度超过该值，如果被使能，系统无条件进入 G2/S5 状态。
5:0	OFFSET - R/W 设置温度传感器的校正偏移（软件），最高位为符号位。

24.2.21 GEN_RTC_1 : General RTC Register 1

地址偏移	电压域	属性
0x50	RTC	R/W

位域	描述
31:0	RTC 通用寄存器

24.2.22 GEN_RTC_2 : General RTC Register 2

地址偏移	电压域	属性
0x54	RTC	R/W

位域	描述
31:0	RTC 通用寄存器

第二十五章 RTC

25.1 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。

25.2 寄存器描述

RTC 模块寄存器位于 0x1fef8000 – 0x1feffff 的 32KB 地址空间内，其基地址为 0x1fef8000，所有寄存器位宽均为 32 位。

25.2.1 寄存器地址列表

名称	地址	位宽	RW	描述
sys_toytrim	0x1fef8020	32	RW	软件必须初始化为 0
sys_toywrite0	0x1fef8024	32	W	TOY 低 32 位数值写入
sys_toywrite1	0x1fef8028	32	W	TOY 高 32 位数值写入
sys_toyread0	0x1fef802C	32	R	TOY 低 32 位数值读出
sys_toyread1	0x1fef8030	32	R	TOY 高 32 位数值读出
sys_toymatch0	0x1fef8034	32	RW	TOY 定时中断 0
sys_toymatch1	0x1fef8038	32	RW	TOY 定时中断 1
sys_toymatch2	0x1fef803C	32	RW	TOY 定时中断 2
sys_rtctrl	0x1fef8040	32	RW	TOY 和 RTC 控制寄存器 软件必须初始化
sys_rtctrim	0x1fef8060	32	RW	软件必须初始化为 0
sys_rtctime0	0x1fef8064	32	W	RTC 定时计数写入
sys_rtctime0	0x1fef8068	32	R	RTC 定时计数读出
sys_rtctime0	0x1fef806C	32	RW	RTC 时钟定时中断 0
sys_rtctime1	0x1fef8070	32	RW	RTC 时钟定时中断 1
sys_rtctime2	0x1fef8074	32	RW	RTC 时钟定时中断 2

25.2.2 TOY 计数器低 32 位数值 (SYS_TOYWRITE0)

偏移量: 0x20

复位值: 0x00000000

位域	名称	访问	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

25.2.3 TOY 计数器高 32 位数值 (SYS_TOYWRITE1)

偏移量: 0x24

复位值: 0x00000000

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

25.2.4 TOY 计数器低 32 位数值 (SYS_TOYREAD0)

偏移量: 0x28

复位值: 0x00000000

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

25.2.5 TOY 计数器高 32 位数值 (SYS_TOYREAD1)

偏移量: 0x2c

复位值: 0x00000000

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

25.2.6 TOY 计数器中断寄存器 0/1/2(SYS_TOYMATCH0/1/2)

偏移量: 0x34/38/3C

复位值: 0x00000000

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

25.2.7 RTC 定时器中断寄存器 0/1/2(SYS_RTCCTRL)

偏移量: 0x40

复位值: 无

位域	名称	访问	描述
31:24	保留	R	保留, 置 0
23	ERS	R	REN(bit13) 写状态
22:21	保留	R	保留, 置 0
20	RTS	R	Sys_rtctrim 写状态
19	RM2	R	Sys_rtcmatch2 写状态
18	RM2	R	Sys_rtcmatch2 写状态
17	RM0	R	Sys_rtcmatch0 写状态
16	RS	R	Sys_rtcwrite 写状态
15	保留	R	保留, 置 0
14	保留	R	保留, 置 0
13	REN	R/W	RTC 使能, 高有效。需要初始化为 1
12	保留	R	保留, 置 0
11	TEN	R/W	TOY 使能, 高有效。需要初始化为 1
10	保留	R	保留, 置 0
9	保留	R	保留, 置 0
8	EO	R/W	0: 32.768k 晶振禁止; 1: 32.768k 晶振使能
7	保留	R	保留, 置 0
6	保留	R	保留, 置 0
5	32S	R	0: 32.768k 晶振不工作; 1: 32.768k 晶振正常工作。
4	保留	R	保留, 置 0
3	TM2	R	Sys_toymatch2 写状态
2	TM1	R	Sys_toymatch1 写状态
1	TM0	R	Sys_toymatch0 写状态
0	TS	R	Sys_toywrite 写状态

25.2.8 RTC 计数器写入端口 (SYS_RTCWRITE)

偏移量: 0x64

复位值: 0x00000000

位域	名称	访问	描述
31:0	RTC	W	

25.2.9 RTC 计数器写入端口 (SYS_RTCREAD)

偏移量: 0x68

复位值: 0x00000000

位域	名称	访问	描述
31:0	RTC	R	

25.2.10 RTC 定时器中断寄存器 0/1/2(SYS_RTCMATCH0/1/2)

偏移量: 0x6C/70/74

复位值: 0x00000000

位域	名称	访问	描述
31:26	RTC	RW	