



SUNway 申威

申威 421 处理器 IO 寄存器手册

2017 年 10 月

成都申威科技有限责任公司



免责声明

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

成都申威科技有限责任公司

Chengdu Sunway Technology Corporation Limited

地址：成都市华府大道四段电子科大科技园 D22 栋

Building D22, National University Science and technology park,
Section 4, Huafu Avenue, Chengdu

Mail: sales@swcpu.cn

Tel : 028-68769016

Fax: 028-68769019



阅读指南

《申威 421 处理器 IO 寄存器手册》主要描述了申威 421 处理器的寄存器编制，寄存器功能介绍等内容。

文档修订

文档更新记录	文档名	申威 421 处理器 IO 寄存器手册
	版本号	V1.0
	创建人	研发部
	创建日期	2017-10-8

版本更新

版本号	更新内容	更新日期
V1.0	初稿	2017-10-8

技术支持

可通过邮箱或问题反馈网站向我司提交产品使用的问题，并获取技术支持。

售后服务邮箱：sales@swcpu.cn

问题反馈网址：<http://www.swcpu.cn/>

目 录

1	概述	1
1.1	术语与约定.....	1
1.2	IO 寄存器的编址.....	2
1.2.1	特殊说明.....	2
1.2.2	IO 寄存器使用的 IO 空间.....	2
1.2.3	存储器接口 IO 寄存器编址.....	3
1.2.4	IO 路由部件与中断处理部件 IO 寄存器编址.....	10
1.2.5	维护接口 IO 寄存器编址.....	13
1.2.6	PCI-E 接口的 IO 空间划分与寄存器编址.....	19
1.3	IO 寄存器的访问机制.....	24
2	存储器接口的 IO 寄存器	25
2.1	一致性处理部件的 IO 寄存器.....	25
2.1.1	基地址说明.....	25
2.1.2	CPM_MC0Disable / CPM_MC1Disable.....	25
2.1.3	CPM_MCCrossMode.....	25
2.1.4	CPM_TCWayMask.....	26
2.1.5	CPM_FRQCmprEn.....	26
2.1.6	CPM_PerfCtrl.....	27
2.1.7	CPM_SerrThres.....	27
2.1.8	CPM_SerrCnt.....	27
2.1.9	CPM_Serr.....	28
2.1.10	CPM_FaultSpot.....	28
2.1.11	CPM_Fault.....	29
2.1.12	CPM_FaultEn.....	31
2.1.13	RN_Fault.....	31
2.1.14	RN_FaultEn.....	32
2.1.15	CPM_TCReqCnt.....	33
2.1.16	CPM_TCHitCnt.....	33
2.1.17	CPM_TCMissCnt.....	33
2.1.18	CPM_AddrCfiCnt.....	34
2.1.19	CPM_FRQCmprCnt.....	34
2.1.20	RN_BlockFlag.....	34
2.2	存储控制器内部的 IO 寄存器.....	35
2.2.1	基地址说明.....	35
2.2.2	CfgTR0.....	36
2.2.3	CfgTR1.....	36
2.2.4	CfgTR2.....	37
2.2.5	CfgTR3.....	38
2.2.6	CfgTR4.....	38
2.2.7	CfgTR5.....	39
2.2.8	CfgTR6.....	39
2.2.9	CfgTR7.....	40
2.2.10	CfgDec.....	40
2.2.11	CfgCR.....	42
2.2.12	ODT_CTRL.....	45
2.2.13	MC_CTRL.....	46
2.2.14	INIT_CTRL.....	49
2.2.15	ClosePage.....	50
2.2.16	DTUR.....	50
2.2.17	DTUW.....	51
2.2.18	DTUWD0.....	51
2.2.19	DTUWD1.....	51
2.2.20	DTURD0.....	52

2.2.21	DTURD1	52
2.2.22	DTUCTL	52
2.2.23	DTUPREA	53
2.2.24	WL_CTRL	53
2.2.25	MemSERR_P	54
2.2.26	CfgErr	54
2.2.27	FSMSTAT	55
2.2.28	CfgTest	56
2.2.29	RC	56
2.2.30	RCMRD	56
2.2.31	6U	57
2.2.32	POWERCTRL	57
2.2.33	PUB_INTERFACE	58
2.2.34	CfgMR0	59
2.2.35	CfgMR1	60
2.2.36	CfgMR2	61
2.2.37	CfgMR3	62
2.2.38	PREF_CTRL	63
2.2.39	MC_PERF0	64
2.2.40	MC_PERF1	64
2.2.41	RWCNTOFREF	64
2.2.42	MEMSERR_TH	65
2.2.43	MEMSERR_CNT	65
2.2.44	WRDATASERR_CNT	66
2.2.45	MEMMERRDATAx	66
2.2.46	MEMMERRADDR	67
2.3	存储器地址/命令/数据接口相关 IO 寄存器	68
2.3.1	基地址说明	68
2.3.2	RIDR	68
2.3.3	PIR	69
2.3.4	PGCR0-2	71
2.3.5	PGSR0-1	76
2.3.6	PLLCR	78
2.3.7	PTR0-4	80
2.3.8	ACMDLR	81
2.3.9	ACBDLR	82
2.3.10	ACIOCR	82
2.3.11	DXCCR	83
2.3.12	DSGCR	86
2.3.13	DCR	88
2.3.14	DTPR0-2	88
2.3.15	MR0-3	90
2.3.16	ODTCR	94
2.3.17	DTCR	95
2.3.18	DTAR0-3	96
2.3.19	DTDR0-1	97
2.3.20	DTEDR0-1	98
2.3.21	RDIMMGCR0-1	99
2.3.22	RDIMMMCR0-1	100
2.3.23	DCUAR	103
2.3.24	DCUDR	103
2.3.25	DCURR	104
2.3.26	DCULR	104
2.3.27	DCUGCR	105
2.3.28	DCUTPR	105
2.3.29	DCUSR0-1	106
2.3.30	BISTR	106

2.3.31	BISTWCR.....	108
2.3.32	BISTMSKR0-2.....	108
2.3.33	BISTLSR.....	109
2.3.34	BISTAR0-2.....	109
2.3.35	BISTUDPR.....	110
2.3.36	BISTGSR.....	111
2.3.37	BISTWER.....	111
2.3.38	BISTBER0-3.....	112
2.3.39	BISTWCSR.....	112
2.3.40	BISTFWR0-2.....	113
2.3.41	AACR.....	114
2.3.42	ZQnCR0-1.....	114
2.3.43	ZQnSR0-1.....	116
2.3.44	DXnGCR.....	117
2.3.45	DXnGSR0-2.....	120
2.3.46	DXnBDLR0-4.....	122
2.3.47	DXnLCDLR0-2.....	123
2.3.48	DXnMDLR.....	125
2.3.49	DXnGTR.....	126
3	IO 路由部件与中断处理部件的 IO 寄存器.....	128
3.1	IO 路由部件的 IO 寄存器.....	128
3.1.1	基地址说明.....	128
3.1.2	IRU_CTRL.....	128
3.1.3	IRU_ERR.....	129
3.1.4	IRU_ERREN.....	130
3.1.5	SERR_CNTTH.....	130
3.1.6	MCUSERR_CNT.....	131
3.1.7	PIUSERR_CNT.....	131
3.1.8	INTPUSERR_CNT.....	131
3.1.9	LOCKEN.....	132
3.1.10	IRU_STAT.....	132
3.2	中断处理部件的 IO 寄存器.....	133
3.2.1	基地址说明.....	133
3.2.2	CORE_SLEEP_STAT.....	134
3.2.3	DEVINT_MIS.....	134
3.2.4	SHORT_CLK_START.....	135
3.2.5	SHORT_CLK_MSK.....	135
3.2.6	MT_INDX.....	136
3.2.7	SHORT_INDX.....	136
3.2.8	INTEN.....	136
3.2.9	AMBA_INDX.....	137
3.2.10	FMT_ERR.....	137
3.2.11	SYSERR_INDX.....	138
3.2.12	NMI_INDX.....	138
3.2.13	SERR_CNTTH.....	138
3.2.14	MCUSERR_CNT.....	139
3.2.15	IRUSERR_CNT.....	139
3.2.16	ERRRPT_EN.....	139
3.2.17	SYSERR_NMI_INT_DST.....	140
3.2.18	PCIE0_INTx_INDX.....	141
3.2.19	PCIE1_INTx_INDX.....	141
3.2.20	IINT_MIS.....	142
3.2.21	IINTMIS_RPTEN.....	142
3.2.22	DEVINT_MIS_RPTEN.....	143
3.2.23	ECCSERR.....	143
3.2.24	ECCSERR_RPTEN.....	144
3.2.25	ECCMERR.....	144

3.2.26	ECCMERR_RPTEN	144
3.2.27	DEVINT_WKEN	145
3.2.28	DEVINTWK_INTEN (新增)	145
3.2.29	MAIL_BOX_i	146
3.2.30	MAIL_STAT_i	146
3.2.31	PCIE0_MSI[i]_INDX	147
3.2.32	PCIE1_MSI[i]_INDX	147
4	维护接口的 IO 寄存器.....	148
4.1	维护接口的 IO 寄存器.....	148
4.1.1	基地址说明	148
4.1.2	CG_ONLINE.....	148
4.1.3	MC_ONLINE.....	149
4.1.4	CORE_ONLINE	150
4.1.5	CGx_FAULT_STAT	150
4.1.6	CGx_FAULT_EN.....	152
4.1.7	CGx_FAULT_INT_EN	153
4.1.8	INIT_CTL	155
4.1.9	BIST_CTL.....	156
4.1.10	BIST_OVERTIME_TH.....	159
4.1.11	MT_STATE0/1	159
4.1.12	MT_INT	162
4.1.13	MT_INT_END	162
4.1.14	CPUID.....	163
4.1.15	SOFT_INFO~15.....	163
4.1.16	LONG_TIME	163
4.1.17	LONG_PRESET	164
4.1.18	TESTSEL	164
4.1.19	IO_START.....	164
4.1.20	PERST_N_PCIE0~1	164
4.1.21	BUTTON_RST_N_PCIE0~1	165
4.1.22	MTRSPMISS.....	165
4.1.23	SLEEP_DONE0~15	165
4.1.24	PLL_CHG_CNT	166
4.1.25	CGx_BIST_STAT.....	166
4.1.26	BIST_STAT	167
4.1.27	TAPSEL.....	169
4.1.28	ST_NEXT_FLAG	169
4.1.29	SERR_CNTTH.....	169
4.1.30	IRU0SERR_CNT	170
4.1.31	INTPUSERR_CNT	170
4.1.32	IRU1SERR_CNT	170
4.1.33	PIUxSERR_CNT.....	171
4.1.34	ERRRPT_EN.....	171
4.1.35	ERR_INF.....	172
4.1.36	SCAN_WAIT_CFG.....	174
4.1.37	PC_BIST_STAT	175
4.1.38	Corex_ERRRST_N	176
4.1.39	SI_FAULT_STAT	176
4.1.40	SI_FAULT_EN.....	177
4.1.41	SI_FAULT_INT_EN	178
4.1.42	CLK_SEL_PCIE0~1	179
4.1.43	FIFO_SYNSEL	179
4.1.44	CPU_INFO.....	180
4.1.45	FlagReg	180
4.1.46	OP_MASK	180
4.1.47	BIST_GOON.....	181
4.1.48	WAKEUP_CTL.....	181

4.1.49	NMI_CTL.....	181
4.1.50	PIUPLL_CNT	182
4.1.51	FlashINFO.....	183
4.1.52	RTPUSROMCNT.....	183
4.1.53	SICMDCNT	183
4.1.54	FLASHERASE4K.....	184
4.1.55	FLASHERASE32K.....	184
4.1.56	FLASHERASE64K.....	184
4.1.57	FLASHERASEALL.....	185
4.1.58	INTERFACE_EN.....	185
4.1.59	MCU_DEBUG0~7.....	185
4.1.60	MC_CONFIG.....	185
4.1.61	CoreSleepCnt.....	187
4.1.62	DeepSleepFlag.....	187
4.1.63	flash_wrprotect_data	187
4.1.64	CHIP_ID 寄存器.....	188
4.1.65	App_Ltssm_En.....	189
4.2	I2C 接口的 IO 寄存器.....	190
4.2.1	地址说明.....	190
4.2.2	IIC IC_CON 控制寄存器	190
4.2.3	IIC IC_TAR 目标地址寄存器	191
4.2.4	IIC IC_SAR 从设备地址寄存器.....	191
4.2.5	IIC IC_HS_MADDR 高速 master 模式编码地址寄存器	192
4.2.6	IIC IC_DATA_CMD 发送和接收数据命令寄存器.....	192
4.2.7	IIC IC_SS_HCNT 标准速度 SCL 高电平计数寄存器	193
4.2.8	IIC IC_SS_LCNT 标准速度 SCL 低电平计数寄存器.....	193
4.2.9	IIC IC_FS_HCNT 快速 SCL 高电平计数寄存器	193
4.2.10	IIC IC_FS_LCNT 快速 SCL 低电平计数寄存器.....	194
4.2.11	IIC IC_HS_HCNT 高速 SCL 高电平计数寄存器.....	194
4.2.12	IIC IC_HS_LCNT 高速 SCL 低电平计数寄存器.....	194
4.2.13	IIC IC_INTR_STAT 中断状态寄存器	195
4.2.14	IIC IC_INTR_MASK 中断掩码寄存器.....	195
4.2.15	IIC IC_RAW_INTR_STAT 原始中断状态寄存器	196
4.2.16	IIC IC_RX_TL 接收 FIFO 阈值寄存器.....	198
4.2.17	IIC IC_TX_TL 发送 FIFO 阈值寄存器.....	198
4.2.18	IIC IC_CLR_INTR 中断清除寄存器.....	198
4.2.19	IIC IC_CLR_RX_UNDER 清除 RX_UNDER 中断寄存器.....	199
4.2.20	IIC IC_CLR_RX_OVER 清除 RX_OVER 中断寄存器	199
4.2.21	IIC IC_CLR_TX_OVER 清除 TX_OVER 中断寄存器.....	199
4.2.22	IIC IC_CLR_RD_REQ 清除 RD_REQ 中断寄存器	199
4.2.23	IIC IC_CLR_TX_ABRT 清除 RD_REQ 中断寄存器	200
4.2.24	IIC IC_CLR_RX_DONE 清除 RX_DONE 中断寄存器.....	200
4.2.25	IIC IC_CLR_ACTIVITY 清除 ACTIVITY 中断寄存器.....	200
4.2.26	IIC IC_CLR_STOP_DET 清除 STOP_DET 中断寄存器	200
4.2.27	IIC IC_CLR_START_DET 清除 START_DET 中断寄存器.....	201
4.2.28	IIC IC_CLR_GEN_CALL 清除 GEN_CALL 中断寄存器	201
4.2.29	IIC IC_ENABLE 总线有效寄存器	201
4.2.30	IIC IC_STATUS 总线状态寄存器	201
4.2.31	IIC IC_TXFLR 发送 FIFO 标记寄存器.....	202
4.2.32	IIC IC_RXFLR 接收 FIFO 标记寄存器	202
4.2.33	IIC IC_SDA_HOLD SDA 保持时间寄存器.....	203
4.2.34	IIC IC_TX_ABRT_SOURCE 传输中断源寄存器	203

4.2.35	IIC IC_SLV_DATA_NACK_ONLY 产生从设备数据 NACK 寄存器	204
4.2.36	IIC IC_DMA_CR DMA 控制寄存器	205
4.2.37	IIC IC_DMA_TDLR DMA 发送数据标记寄存器	205
4.2.38	IIC IC_DMA_RDLR DMA 接收数据标记寄存器	205
4.2.39	IIC IC_SDA_SETUP 数据线建立时间寄存器	205
4.2.40	IIC IC_ACK_GENERAL_CALL 响应广播寄存器	206
4.2.41	IIC IC_ENABLE_STATUS 有效状态寄存器	206
4.2.42	IIC IC_COMP_PARAM_1 参数寄存器 1	206
4.2.43	IIC IC_COMP_VERSION 版本寄存器	208
4.2.44	IIC IC_COMP_TYPE 组成方式寄存器	208
5	PCI-E 接口 IO 寄存器	209
5.1	PCI-E 接口控制部件内部的 IO 寄存器	209
5.1.1	PIU 内部 IOR0 部分寄存器	210
5.1.2	PIU 内部 IOR1 部分寄存器	225
5.2	PCI-E 根部件内部的 IO 寄存器	242
5.2.1	RC 寄存器空间布局	243
5.2.2	PCI 兼容配置头寄存器	247
5.2.3	PCI 标准能力结构寄存器	255
5.2.4	PCI-E 扩展能力寄存器	270
5.2.5	端口逻辑寄存器	275
附录 A	PCI-E 术语表	287
附录 B	PCI-E 链路状态编码	289

1 概述

IO 寄存器 (IOR) 是申威 421 处理器处理器内部存储器接口和 PCIe 接口等部件中用于控制存储器访问和 PCIe 接口访问的控制与状态寄存器，这些寄存器可以通过核心进行读写访问，也可以通过维护接口的维护命令进行读写访问。

申威 421 处理器处理器的 IO 寄存器按照所在部件的不同，分成五大类，分别是：

- 1) 存储器接口 IO 寄存器：每个存储控制器及其对应的存储器接口，各有一组 IO 寄存器；
- 2) IO 路由部件 (IRU) IO 寄存器：IO 路由部件内的一组 IO 寄存器；
- 3) 中断处理部件 (INTPU) IO 寄存器：中断处理部件内的一组 IO 寄存器；
- 4) 维护接口 IO 寄存器：维护接口部件内的一组 IO 寄存器；
- 5) PCI-E 接口 IO 寄存器：每个 PCI-E 接口 (包括 PCI-E 控制部件和 PCI-E 根部件)，各有一组 IO 寄存器。

1.1 术语与约定

本手册对 IO 寄存器内容的软件访问特性和复位状态进行描述时，遵循如下符号约定：

- 1) “W” 指示可写；
- 2) “WO” 指示只写；
- 3) “R” 指示可读；
- 4) “RO” 指示只读；
- 5) “RC” 指示读清除；
- 6) “WC” 指示写清除；
- 7) “RW” 指示可读写；
- 8) “RWC” 指示可读且写清除；
- 9) “WIC” 指示写 “1” 清除，写 “0” 无影响；
- 10) “RO,x” 指示只读且复位时初值为 “x”；
- 11) “WO,x” 指示只写且复位时初值为 “x”；
- 12) “RW,x” 指示可读写且复位时初值为 “x”；
- 13) “SEXT (x)” 指示将 “x” 扩展指定位数；
- 14) PCI-E 的 RC 配置空间寄存器的属性描述，遵循表 5-1 的约定。

以上 “x” 可以为 “0”，也可以为 “非 0 的确定值”，没有指定复位设置值的 IO 寄存器在复位后是一个不确定的值，需要在复位后进行初始化设置。读保留位时，返回 “0”；写保留位时，写入的值被忽略。

本手册中数字的表示方法如下：

- 1) 十六进制：数字前缀为“0x”或者“n’h”（其中 n 为 16 进制数的位数）；
- 2) 十进制：没有前缀的数字；
- 3) 二进制：数字前缀为“0b”或者“n`b”（其中 n 为 2 进制数的位数）。

1.2 IO 寄存器的编址

1.2.1 特殊说明

申威 421 处理器芯片使用 48 位物理地址空间，其中物理地址的最高位 PA[47]用于区分是存储器空间（PA[47]位为“0”）还是 IO 空间（PA[47]位为“1”）。对于存储器空间物理地址 PA[46]位作为可 Cache 属性标志位，为“0”表示可 Cache 访问，为“1”表示不可 Cache 访问；PA[45:0]位作为真正的存储器空间物理地址，用于访问核外存储器空间。实际上，申威 421 处理器芯片存储器接口最大支持 256GB（PA[37:0]），PA[45:38]是便于以后扩展。

【特殊说明】：核心视角和维护接口视角的物理地址空间略有不同。核心视角是 48 位物理地址，但维护接口协议中请求包地址域仍维持 40 位，即维护视角仍是 40 位物理地址（即维护视角的 Addr[39]作为 PA[47]、Addr[38:0]作为 PA[38:0]）。核心视角，IO 寄存器对应的物理地址 PA[47]固定为“1”。维护接口视角，IO 寄存器对应的物理地址 PA[39]固定为“1”。

1.2.2 IO寄存器使用的IO空间

物理地址 PA[38:36]区分不同部件的 IO 寄存器使用的 IO 空间（要求核心视角时物理地址[46:40]为全零），物理地址 PA[35:7]作为 IO 寄存器的索引地址。

不同部件的 IO 寄存器对应的物理地址 PA[38:36]位如下：

- 1) PA[38:36]=“0b000”，对应存储器接口 IO 寄存器；
- 2) PA[38:36]=“0b100”，对应 PCI-E 0 接口 IO 寄存器；
- 3) PA[38:36]=“0b101”，对应维护接口 IO 寄存器；
- 4) PA[38:36]=“0b110”，对应 IRU0、IRU1 和 INTPU 的 IO 寄存器（原 IPU 地址空间）；
- 5) PA[38:36]=“0b111”，对应 PCI-E 1 接口 IO 寄存器。
- 6) 其它保留。

1.2.3 存储器接口IO寄存器编址

PA[35:26]为全零，PA[25:23]表示一致性处理部件或存储器接口的 IOR，具体如下：

PA[25:23]= “0b000”，对应一致性处理部件的 IO 寄存器；

PA[25:23]= “0b100”，对应存储控制器 0 的 IO 寄存器；

PA[25:23]= “0b101”，对应存储器地址/命令/数据接口 0 的 IO 寄存器；

PA[25:23]= “0b110”，对应存储控制器 1 的 IO 寄存器；

PA[25:23]= “0b111”，对应存储器地址/命令/数据接口 1 的 IO 寄存器；

其它保留。

1.2.3.1 一致性处理部件 IO 寄存器编址

一致性处理部件（CPM）IO 寄存器编址如表 1-1 所示，其中地址[6:0]要求为全零。

表 1-1：一致性处理部件 IO 寄存器编址

序号	名称	符号	索引地址 PA[22:7]	存取特性
1	MC#0 不使能寄存器	CPM_MC0Disable	0x0000	RW
2	MC#1 不使能寄存器	CPM_MC1Disable	0x0001	RW
3	核组内 MC 交叉模式寄存器	CPM_CrossMode	0x0002	RW
4	TCache 路屏蔽寄存器	CPM_TCWayMask	0x0003	RW
5	FRQ 压缩使能寄存器	CPM_FRQCmprEn	0x0004	RW
6	性能计数器控制寄存器	CPM_PerfCtrl	0x0005	RW
7	单错计数阈值寄存器	CPM_SerrThres	0x0008	RW
8	CPM 单错计数器	CPM_SerrCnt	0x0009	RWC
9	CPM 单错寄存器	CPM_Serr	0x000A	RW1C
10	CPM 故障现场寄存器	CPM_FaultSpot	0x000B	RWC
11	CPM 故障状态寄存器	CPM_Fault	0x000C	RW1C
12	CPM 故障报错使能寄存器	CPM_FaultEn	0x000D	RW
13	环网故障状态寄存器	RN_Fault	0x000E	RW1C
14	环网故障报错使能寄存器	RN_FaultEn	0x000F	RW
15	可 Cache 请求计数器	CPM_TCReqCnt	0x0010	RO
16	TCache 直接命中计数器	CPM_TCHitCnt	0x0011	RO
17	TCache 脱靶计数器	CPM_TCMissCnt	0x0012	RO

18	地址冲突计数器	CPM_AddrCflCnt	0x0013	RO
19	FRQ 压缩计数器	CPM_FRQCmprCnt	0x0014	RO
20	环网阻塞超时标志	RN_BlockFlag	0x0015	RO

注 1：“黑体标识的部分触发器”为申威 421 处理器 CPM 在申威 411 处理器的基础上新增的寄存器。此外，删除了申威 411 处理器 CPM 中与核心睡眠功能相关的“核心睡眠使能寄存器 CPM_CoreSleep”和“CPM 外部 Probe 完成标志 CPMProbeOk”以及 TMAF 最大使用情况寄存器“CPM_TMAFMax”。

注 2：此外，CPM_CrossMode、CPM_Fault 和 CPM_FaultEn 寄存器相对于申威 411 处理器有更新。

1.2.3.2 存储控制器的 IO 寄存器编址

存储控制器的 IO 寄存器编址如表 1-2 所示，其中地址[6:0]要求为全零。

表 1-2: 存储控制器 IO 寄存器编址

序号	名称	符号	索引地址 PA[22:7]	存取 特性
1	时序寄存器 0	CfgTR0	0x0000	RW
2	时序寄存器 1	CfgTR1	0x0001	RW
3	时序寄存器 2	CfgTR2	0x0002	RW
4	时序寄存器 3	CfgTR3	0x0003	RW
5	时序寄存器 4	CfgTR4	0x0004	RW
6	时序寄存器 5	CfgTR5	0x0005	RW
7	时序寄存器 6	CfgTR6	0x0006	RW
8	时序寄存器 7	CfgTR7	0x0007	RW
9	配置译码寄存器	CfgDec	0x0008	RO
10	配置寄存器	CfgCR	0x0009	RW
11	存储控制器控制寄存器	MC_CTRL	0x000a	RW
12	存控初始化控制寄存器	INIT_CTRL	0x000b	RW
13	页关闭策略控制寄存器	ClosePage	0x000c	RW
14	数据通路信号调节读测试寄存器	DTUR	0x000d	RW
15	数据通路信号调节写测试寄存器	DTUW	0x000e	RW
16	数据通路信号调节写测试数据寄存器 0	DTUWD0	0x000f	RW

17	数据通路信号调节写测试数据寄存器 1	DTUWD1	0x0010	RW
18	保留	—	0x0011	—
19	存储控制器 ODT 控制寄存器	ODT_CTRL	0x0012	RW
20	数据通路信号调节读测试数据寄存器 0	DTURD0	0x0013	RO
21	数据通路信号调节读测试数据寄存器 1	DTURD1	0x0014	RO
22	保留	—	0x0015	—
23	数据通路信号调节测试控制寄存器	DTUCTL	0x0016	RW
24	数据通路信号调节全局预充电寄存器	DTUPREA	0x0017	RW
25	Write Leveling 控制寄存器	WL_CTRL	0x0018	RW
26	主存数据单错验证码寄存器	MEMSERR_P	0x0019	RW1C
27	配置错误寄存器	CfgErr	0x001a	RO
28	保留	—	0x001b	—
29	初始化状态寄存器	FSMSTAT	0x001c	RO
30	RDIMM 控制字配置寄存器	RC	0x001d	RW
31	RDIMM 控制字配置延时控制寄存器 0	RCMRD	0x001e	RW
32	RDIMM 控制字配置延时控制寄存器 1	6U	0x001f	RW
33	存储器地址/命令/数据接口控制寄存器	PUB_INTERFACE	0x0020	RW
34	功耗控制寄存器	POWERCTRL	0x0021	RW
35	测试寄存器	CfgTest	0x0022	RW
36-37	保留	—	0x0023~ 0x0024	—
38	SDRAM 模式寄存器 0	CfgMR0	0x0025	RW
39	SDRAM 模式寄存器 1	CfgMR1	0x0026	RW
40	SDRAM 模式寄存器 2	CfgMR2	0x0027	RW
41	SDRAM 模式寄存器 3	CfgMR3	0x0028	RW
42	性能检测控制寄存器	PERF_CTRL	0x0029	RW
43	性能检测寄存器 0	MC_PERF0	0x002a	RO
44	读主存数据单错寄存器	MEMSERR_CNT	0x002b	RW1C
45	单错阈值寄存器	MEMSERR_TH	0x002c	RW
46	写主存数据单错寄存器	WRDATASERR_CNT	0x002d	RW1C
47	性能检测寄存器 1	MC_PERF1	0x002e	RO
48	刷新周期性能计数寄存器	RWCNTOFREF	0x002f	RO
49	存储器数据多错数据寄存器 0	MEMMERRDATA0	0x0030	RO

50	存储器数据多错数据寄存器 1	MEMMERRDATA1	0x0031	RO
51	存储器数据多错数据寄存器 2	MEMMERRDATA2	0x0032	RO
52	存储器数据多错数据寄存器 3	MEMMERRDATA3	0x0033	RO
53	存储器数据多错数据寄存器 4	MEMMERRDATA4	0x0034	RO
54	存储器数据多错地址寄存器	MEMMERRADDR	0x0035	RO

注 1：“黑体标识的补充触发器”为申威 421 处理器在申威 411 处理器的基础上新增寄存器。

注 2：此外，CfgCR、MC_CTRL、DTUR、DTUW、DTUPREA、WL_CTRL、FSMSTAT 寄存器相对于申威 411 处理器有更新。

1.2.3.3 存储器地址/命令/数据接口 IO 寄存器编址

存储器地址/命令/数据接口的 IO 寄存器编址如表 1-3 所示，其中地址[6:0]要求为全零。

表 1-3：存储器地址/命令/数据接口的 IO 寄存器编址

序号	名称	符号	索引地址 PA[22:7]	存取 特性
1	版本信息寄存器	RIDR	0x0000	RO
2	初始化寄存器	PIR	0x0001	RW
3~4	通用控制寄存器	PGCR0-1	0x0002~0x0003	RW
5~6	通用状态寄存器	PGSR0-1	0x0004~0x0005	RO
7	PLL 控制寄存器	PLLCR	0x0006	RW
8~12	延时参数寄存器	PTR0-4	0x0007~0x000B	RW
13	地址/命令主延时寄存器	ACMDLR	0x000C	RW
14	地址/命令位延时寄存器	ACBDLR	0x000D	RW
15	地址/命令 IO 配置寄存器	ACIOCR	0x000E	RW
16	数据通用配置寄存器	DXCCR	0x000F	RW
17	系统通用配置寄存器	DSGCR	0x0010	RW
18	SDRAM 配置寄存器	DCR	0x0011	RW
19~21	SDRAM 延时参数寄存器	DTPR0-2	0x0012~0x0014	RW
22	SDRAM 模式寄存器 0	MR0	0x0015	RW
23	SDRAM 模式寄存器 1	MR1	0x0016	RW
24	SDRAM 模式寄存器 2	MR2	0x0017	RW
25	SDRAM 模式寄存器 3	MR3	0x0018	RW
26	ODT 配置寄存器	ODTCR	0x0019	RW
27	Data Training 配置寄存器	DTCR	0x001A	RW

28~31	Data Training 地址寄存器	DTAR0-3	0x001B~0x001E	RW
32~33	Data Training 数据寄存器	DTDR0-1	0x001F~0x0020	RW
34~35	Data Training 眼图数据寄存器	DTEDR0-1	0x0021~0x0022	RW
36	通用控制寄存器 2	PGCR2	0x0023	RW
37~46	保留	—	0x0024~0x002B	—
47	RDIMM 通用控制寄存器 0-1	RDIMMGCR0-1	0x002C~0x002D	RW
48	RDIMM 控制寄存器 0-1	RDIMMCR0-1	0x002E~0x002F	RW
49	SDRAM 命令单元地址寄存器	DCUAR	0x0030	RW
50	SDRAM 命令单元数据寄存器	DCUDR	0x0031	RW
51	SDRAM 命令单元运行寄存器	DCURR	0x0032	RW
52	SDRAM 命令单元循环寄存器	DCULR	0x0033	RW
53	SDRAM 命令单元通用配置寄存器	DCUGCR	0x0034	RW
54	SDRAM 命令单元时间延时寄存器	DCUTPR	0x0035	RW
55~56	SDRAM 命令单元状态寄存器	DCUSR0-1	0x0036~0x0037	RO
57~64	保留	—	0x0038~0x003F	—
65	BIST 运行寄存器	BISTRR	0x0040	RW
66	BIST 字计数寄存器	BISTWCR	0x0041	RW
67~69	BIST 屏蔽寄存器	BISTMSKR0-2	0x0042~0x0044	RW
70	BIST 随机测试种子寄存器	BISTLSR	0x0045	RW
71~73	BIST 地址寄存器	BISTAR0-2	0x0046~0x0048	RW
74	BIST 用户数据向量寄存器	BISTUDPR	0x0049	RW
75	BIST 通用状态寄存器	BISTGSR	0x004A	RO
76	BIST 字错误寄存器	BISTWER	0x004B	RO
77~80	BIST 位错误寄存器	BISTBER0-3	0x004C~0x004F	RO
81	BIST 字计数状态寄存器	BISTWCSR	0x0050	RO
82~84	BIST 失败错误字寄存器	BISTFWR0-2	0x0051~0x0053	RO
85~93	保留	—	0x0054~0x005C	—
94	Anti-aging 控制寄存器	AACR	0x005D	RW
95~96	保留	—	0x005E~0x005F	—
97~98	ZQ0 电阻控制寄存器	ZQ0CR0-1	0x0060~0x0061	RW
99~100	ZQ0 电阻状态寄存器	ZQ0SR0-1	0x0062~0x0063	RW
101~102	ZQ1 电阻控制寄存器	ZQ1CR0-1	0x0064~0x0065	RW
103~104	ZQ1 电阻状态寄存器	ZQ1SR0-1	0x0066~0x0067	RW

105~106	ZQ2 电阻控制寄存器	ZQ2CR0-1	0x0068~0x0069	RW
107~108	ZQ2 电阻状态寄存器	ZQ2SR0-1	0x006A~0x006B	RW
109~110	保留	—	0x006C~0x006D	RW
111~112	保留	—	0x006E~0x006F	RW
113~128	DATX0 通用配置寄存器	DX0GCR	0x0070	RW
	DATX0 通用状态寄存器	DX0GSR0-1	0x0071~0x0072	RW
	DATX0 位延时寄存器	DX0BDLR0-4	0x0073~0x0077	RW
	DATX0 本地校准延时寄存器	DX0LCDLR0-2	0x0078~0x007A	RW
	DATX0 主延时寄存器	DX0MDLR	0x007B	RW
	DATX0 通用延时寄存器	DX0GTR	0x007C	RW
	保留	—	0x007D~0x007F	—
129~144	DATX1 通用配置寄存器	DX1GCR	0x0080	RW
	DATX1 通用状态寄存器	DX1GSR0-1	0x0081~0x0082	RO
	DATX1 位延时寄存器	DX1BDLR0-4	0x0083~0x0087	RW
	DATX1 本地校准延时寄存器	DX1LCDLR0-2	0x0088~0x008A	RW
	DATX1 主延时寄存器	DX1MDLR	0x008B	RW
	DATX1 通用延时寄存器	DX1GTR	0x008C	RW
	保留	—	0x008D~0x008F	—
145~160	DATX2 通用配置寄存器	DX2GCR	0x0090	RW
	DATX2 通用状态寄存器	DX2GSR0-1	0x0091~0x0092	RO
	DATX2 位延时寄存器	DX2BDLR0-4	0x0093~0x0097	RW
	DATX2 本地校准延时寄存器	DX2LCDLR0-2	0x0098~0x009A	RW
	DATX2 主延时寄存器	DX2MDLR	0x009B	RW
	DATX2 通用延时寄存器	DX2GTR	0x009C	RW
	保留	—	0x009D~0x009F	—
161~176	DATX3 通用配置寄存器	DX3GCR	0x00A0	RW
	DATX3 通用状态寄存器	DX3GSR0-1	0x00A1~0x00A2	RO
	DATX3 位延时寄存器	DX3BDLR0-4	0x00A3~0x00A7	RW
	DATX3 本地校准延时寄存器	DX3LCDLR0-2	0x00A8~0x00AA	RW
	DATX3 主延时寄存器	DX3MDLR	0x00AB	RW
	DATX3 通用延时寄存器	DX3GTR	0x00AC	RW
	保留	—	0x00AD~0x00AF	—
177~192	DATX4 通用配置寄存器	DX4GCR	0x00B0	RW

	DATX4 通用状态寄存器	DX4GSR0-1	0x00B1~0x00B2	RO
	DATX4 位延时寄存器	DX4BDLR0-4	0x00B3~0x00B7	RW
	DATX4 本地校准延时寄存器	DX4LCDLR0-2	0x00B8~0x00BA	RW
	DATX4 主延时寄存器	DX4MDLR	0x00BB	RW
	DATX4 通用延时寄存器	DX4GTR	0x00BC	RW
	保留	—	0x00BD~0x00BF	—
193~208	DATX5 通用配置寄存器	DX5GCR	0x00C0	RW
	DATX5 通用状态寄存器	DX5GSR0-1	0x00C1~0x00C2	RO
	DATX5 位延时寄存器	DX5BDLR0-4	0x00C3~0x00C7	RW
	DATX5 本地校准延时寄存器	DX5LCDLR0-2	0x00C8~0x00CA	RW
	DATX5 主延时寄存器	DX5MDLR	0x00CB	RW
	DATX5 通用延时寄存器	DX5GTR	0x00CC	RW
保留	—	0x00CD~0x00CF	—	
209~224	DATX6 通用配置寄存器	DX6GCR	0x00D0	RW
	DATX6 通用状态寄存器	DX6GSR0-1	0x00D1~0x00D2	RO
	DATX6 位延时寄存器	DX6BDLR0-4	0x00D3~0x00D7	RW
	DATX6 本地校准延时寄存器	DX6LCDLR0-2	0x00D8~0x00DA	RW
	DATX6 主延时寄存器	DX6MDLR	0x00DB	RW
	DATX6 通用延时寄存器	DX6GTR	0x00DC	RW
保留	—	0x00DD~0x00DF	—	
225~240	DATX7 通用配置寄存器	DX7GCR	0x00E0	RW
	DATX7 通用状态寄存器	DX7GSR0-1	0x00E1~0x00E2	RO
	DATX7 位延时寄存器	DX7BDLR0-4	0x00E3~0x00E7	RW
	DATX7 本地校准延时寄存器	DX7LCDLR0-2	0x00E8~0x00EA	RW
	DATX7 主延时寄存器	DX7MDLR	0x00EB	RW
	DATX7 通用延时寄存器	DX7GTR	0x00EC	RW
保留	—	0x00ED~0x00EF	—	
241~256	DATX8 通用配置寄存器	DX8GCR	0x00F0	RW
	DATX8 通用状态寄存器	DX8GSR0-1	0x00F1~0x00F2	RO
	DATX8 位延时寄存器	DX8BDLR0-4	0x00F3~0x00F7	RW
	DATX8 本地校准延时寄存器	DX8LCDLR0-2	0x00F8~0x00FA	RW
	DATX8 主延时寄存器	DX8MDLR	0x00FB	RW
	DATX8 通用延时寄存器	DX8GTR	0x00FC	RW

	保留	—	0x00FD~0x00FF	—
--	----	---	---------------	---

注 1：“黑体标识的部分触发器”为申威 421 处理器在申威 411 处理器基础上增加的寄存器。

注 2：申威 421 处理器采用 28nm 工艺，与申威 411 处理器相比，很多寄存器进行了更新，具体见 2.3 节。

1.2.4 IO路由部件与中断处理部件IO寄存器编址

2 个 IO 路由部件与中断处理部件使用原申威 411 处理器 IPU 的地址空间 (PA[38:36]=“0b110”), 其中 INTPU 通过 IO 路由部件 0 进行扩展, 3 个部件的地址空间使用 PA[35:32]进行区分, 具体如下:

PA[35:32]= “0b0000”, 对应中断处理部件 (INTPU) 的 IO 寄存器;

PA[35:32]= “0b0010”, 对应 IO 路由部件 0 (IRU0) 的 IO 寄存器;

PA[35:32]= “0b0011”, 对应 IO 路由部件 1 (IRU1) 的 IO 寄存器;

其它保留。

1.2.4.1 IO 路由部件 IO 寄存器编址

IO 路由部件 (IRU) 内的 IO 寄存器编址如表 1-4 所示, 其中地址[6:0]要求为全零。

表 1-4: IO 路由部件内的 IO 寄存器编址

序号	名称	符号	索引地址 PA[31:7]	存取 特性
1	保留	—	0x000,0000	—
2	IRU 控制寄存器	IRU_CTRL	0x000,0001	RW
3	IRU 错误寄存器	IRU_ERR	0x000,0002	RWC
4	IRU 报错使能	IRU_ERREN	0x000,0003	RW
5	单错阈值寄存器	SERR_CNTTH	0x000,0004	RW
6	MCU 接口单错数量	MCUSERR_CNT	0x000,0005	RWC
7	PIU 接口单错数量	PIUSERR_CNT	0x000,0006	RWC
8	INTPU 接口单错数量	INTPUSERR_CNT	0x000,0007	RWC
9	LOCKEN 寄存器	IRU_LOCKEN	0x000,0008	RW
10	IRU_STAT 寄存器	IRU_STAT	0x000,0009	RO

注 1: IRU 和 INTPU 独立, 申威 421 处理器的 IRU 寄存器重新编址。

注 2: 由于每个 IRU 对应一个 PCIe 接口, 同时 SOC 采用双向环网, IRU_CTRL、IRU_ERR 和 IRU_STAT 等寄存器都相对于申威 411 处理器有所更新。

1.2.4.2 中断处理部件 IO 寄存器编址

中断处理部件（INTPU）内的 IO 寄存器编址如表 1-5 所示，其中地址[6:0]要求为全零。

表 1-5：中断处理部件内的 IO 寄存器编址

序号	名称	符号	索引地址 PA[31:7]	存取 特性
1	保留	—	0x000,0000	—
2	核心睡眠状态寄存器	CORE_SLEEP_STAT	0x000,0001	RO
3	设备中断丢失寄存器	DEVINT_MIS	0x000,0002	RW
4	保留	—	0x000,0003	—
5	短时钟中断计数起始值寄存器	SHORT_CLK_START	0x000,0004	RW
6	短时钟中断屏蔽寄存器	SHORT_CLK_MSK	0x000,0005	RW
7	维护中断 MAP 表	MT_INDX	0x000,0006	RW
8	短时钟中断 MAP 表	SHORT_INDX	0x000,0007	RW
9	中断使能	INTEN	0x000,0008	RW
10~13	保留	—	0x000,0009~ 0x000,000c	—
14	AMBA 设备中断 MAP 表	AMBA_INDEX	0x000,000d	RW
15	包格式错误寄存器	FMT_ERR	0x000,000e	RW
16	系统故障中断 MAP 表	SYSERR_INDX	0x000,000f	RW
17	保留	—	0x000,0010	—
18	INTPU 单错预警阈值寄存器	SERR_CNTTH	0x000,0011	RW
19	INTPU 接收 MCU 方向请求或响应 包包头 ECC 单错次数统计寄存器	MCUSERR_CNT	0x000,0012	RW
20~21	保留	—	0x000,0013~ 0x000,0014	—
22	INTPU 接收 IRU 方向请求或响应 包包头 ECC 单错次数统计寄存器	IRUSERR_CNT	0x000,0015	RW
23	INTPU 报错使能寄存器	ERRRPT_EN	0x000,0016	RW
24	故障中断目标配置寄存器	SYSERRINT_DST	0x000,0017	RW
25	保留	—	0x000,0018	—
26~29	PCI-E0 INT _x 中断 MAP 表	PCIE0_INT _x _INDX	0x000,0019~ 0x000,001c	RW
30~33	PCI-E1 INT _x 中断 MAP 表	PCIE1_INT _x _INDX	0x000,001d~	RW

			0x000,0020	
34~35	保留	—	0x000,0021~ 0x000,0022	—
36	核间中断丢失状态寄存器	IINT_MIS	0x000,0023	RW
37	核间中断丢失报错使能寄存器	IINT_MIS_RPTEN	0x000,0024	RW
38	PCI-E 和维护中断丢失寄存器	DEVINT_MIS_RPTEN	0x000,0025	RW
39	ECC 单错状态寄存器	ECCSERR	0x000,0026	RW
40	ECC 单错报错使能寄存器	ECCSERR_RPTEN	0x000,0027	RW
41	ECC 多错状态寄存器	ECCMERR	0x000,0028	RW
42	ECC 多错报错使能寄存器	ECCMERR_RPTEN	0x000,0029	RW
43	设备中断唤醒使能寄存器	DEVINT_WKEN	0x000,002a	RW
44	不可屏蔽中断 NMI MAP 表	NMI_INDX	0x000,002b	RW
45	设备中断唤醒再次中断使能寄存器	DEVINTWK_INTEN	0x000,002c	RW
46 ~ 128	保留	—	0x000,002d~ 0x000,007f	—
129 ~ 133	异步消息队列	MAIL_BOX_i	0x000,0080~ 0x000,0083 (这 4 个地址由 低到高对应目标 核心 0~3; 当某 个核心断连时, 对应的地址为非 法地址)	RW
134 ~ 144	保留	—	0x000,0084~ 0x000,008f	—
145 ~ 148	异步消息队列状态寄存器	MAIL_STAT_i	0x000,0090~ 0x000,0093 (这 4 个地址由 低到高对应目标 核心 0~3; 当某 个核心断连时, 对应的地址为非	RO

			法地址)	
149 ~ 160	保留	—	0x000,0094~ 0x000,009f	—
161~ 256	保留	—	0x000,00a0~ 0x000,00ff	—
257~ 512	PCI-E0 MSI 中断 MAP 表	PCIE0_MSI[i]_INDX	0x000,0100~ 0x000,01ff	RW
513~ 768	PCI-E1 MSI 中断 MAP 表	PCIE1_MSI[i]_INDX	0x000,0200~ 0x000,02ff	RW

注 1: IRU 和 INTPU 独立, 申威 421 处理器的 INTPU 寄存器重新编址。“黑体标识的部分触发器”为申威 421 处理器在申威 411 处理器基础上增加的寄存器。

1.2.5 维护接口IO寄存器编址

维护接口的 IO 寄存器包括三个部分, 通过物理地址[35:34]加以区分:

PA[35:34]= “0b00”, 对应维护接口的 IO 寄存器;

PA[35:34]= “0b01”, 对应 I2C IP 的 IO 寄存器;

PA[35:34]= “0b1x”, 对应 Flash 空间;

1.2.5.1 维护接口的 IO 寄存器编址

维护接口部件 (MCU) 内的 IO 寄存器编址如表 1-6 所示, 其中地址[6:0]要求为全零。

表 1-6: 维护接口内的 IO 寄存器编址

序号	名称	符号	索引地址 PA[33:7]	存取特 性
1	核组在位使能寄存器	CG_ONLINE	0x000,0000	RO
2	计算核组总错状态寄存器	CG0_FAULT_STAT	0x000,0001	RW
3~5	保留	保留	0x000,0002~ 0x000,0004	保留
6	计算核组总错状态报错使能寄存器	CG0_FAULT_EN	0x000,0005	RW
7~9	保留	保留	0x000,0006~ 0x000,0008	保留

10	计算核组故障中断报错使能寄存器	CG0_FAULT_INT_EN	0x000,0009	RW
11~13	保留	保留	0x000,000a~ 0x000,000c	保留
14	初始化控制寄存器	INIT_CTL	0x000,000d	RW
15	芯片状态寄存器	MT_STATE0	0x000,000e	RO
16	核心在位使能寄存器	CORE_ONLINE	0x000,000f	RW
17	维护中断寄存器	MT_INT	0x000,00010	WO
18	维护中断结束寄存器	MT_INT_END	0x000,00011	RW
19	CPU 号寄存器	CPUID	0x000,00012	RW
20~35	核心软件信息寄存器	SOFT_INF0~15	0x000,0013~ 0x000,0022	RW
36	长时钟计数寄存器	LONG_TIME	0x000,0023	RW
37	长时钟计数间隔寄存器	LONG_PRESET	0x000,0024	RW
38	测试选择控制寄存器	TESTSEL	0x000,0025	RW
39	外围接口使能寄存器	IO_START	0x000,0026	RW
40	PCI-E0 接口 PERST_N 复位寄存器	PERST_N_PCIE0	0x000,0027	RW
41	PCI-E0 接口 BUTTON_RST_N 复位寄存器	BUTTON_RST_N_PCIE0	0x000,0028	RW
42	BIST 测试控制寄存器	BIST_GOON	0x000,0029	WO
43	维护响应不匹配计数器	MTRSPMISS	0x000,002a	RO
44~59	中断睡眠完成寄存器	SLEEP_DONE0~15	0x000,002b~ 0x000,002e	WO
48~59	保留	保留	0x000,002f~ 0x000,003a	保留
60	时钟切换时间间隔	PLL_CHG_CNT	0x000,003b	RW
61	核组 0 BIST 结果信息	CG0_BIST_STAT	0x000,003c	RO
62~64	保留	保留	0x000,003d~ 0x000,003f	保留
65	芯片 BIST 结果信息寄存器	BIST_STAT	0x000,0040	RO
66	JTAG TAP 控制选择寄存器	TAPSEL	0x000,0041	RO
67	PCI-E1 接口 PERST_N 复位寄存器	PERST_N_PCIE1	0x000,0042	RW

68	PCI-E1 接口 BUTTON_RST_N 复位寄存器	BUTTON_RST_N_PCIE1	0x000,0043	RW
69	存储器自测试控制寄存器	BIST_CTL	0x000,0044	RW
70	存储器自测试超时阈值寄存器	BIST_OVERTIME_TH	0x000,0045	RW
71	状态机控制寄存器	ST_NEXT_FLAG	0x000,0046	WO
72	维护操作屏蔽寄存器	OP_MASK	0x000,0047	RW
73	MCU 单错预警阈值寄存器	SERR_CNTTH	0x000,0048	RW
74	MCU 接收 IRU0 方向请求或响 应单错次数统计寄存器	IRU0SERR_CNT	0x000,0049	RW
75~76	MCU 接收 PCI-E 方向响应单 错次数统计寄存器	PIU0SERR_CNT~ PIU1SERR_CNT	0x000,004a~ 0x0000,004b	RW
77	MCU 报错使能寄存器	ERRRPT_EN	0x000,004c	RW
78	MCU 错误信息寄存器	ERR_INF	0x000,004d	RW
79	保留	—	0x000,004e	—
80	SCAN 间隔控制寄存器	SCAN_WAIT_CFG	0x000,004f	RW
81	保留	—	0x000,0050	—
82	PCACHE BIST 结果寄存器	PC_BIST_STAT	0x000,0051	RO
83~86	核心错误信息清除寄存器	Core0_ERRRST_N~ Core15_ERRRST_N	0x000,0052~ 0x000,0055	RW
87~98	保留	—	0x000,0056~ 0x000,0061	—
99	系统接口总错状态寄存器	SI_FAULT_STAT	0x000,0062	RW
100	系统接口总错状态报错使能寄 存器	SI_FAULT_EN	0x000,0063	RW
101	系统接口故障中断报错使能寄 存器	SI_FAULT_INT_EN	0x000,0064	RW
102~ 103	PCI-E 时钟选择寄存器	CLK_SEL_PCIE0~1	0x000,0065~ 0x000,0066	RW
104	保留	—	0x000,0067	—
105	异步 FIFO 读写指针同步的级 数配置寄存器	FIFO_SYNSEL	0x000,0068	RW
106	CPU 信息寄存器	CPUINFO	0x000,0069	RW
107	外部唤醒中断控制寄存器	WAKEUP_CTL	0x000,006a	RW

108	标志寄存器	FLAGREG	0x000,006b	RW
109	外部故障中断控制寄存器	NMI_CTL	0x000,006c	RW
110	PCI-E PHY PLL 等待计数器	PIUPLL_CNT	0x000,006d	RW
111	保留	—	0x000,006e	—
112	核组在位使能寄存器	MC_ONLINE	0x000,006f	RW
113	FLASHINFO 寄存器	FLASHINFO	0x000,0070	RO
114	SROM 计数器	RTPUSROMCNT	0x000,0071	RO
115	维护复位及 SROM 计数器	SICMDCNT	0x000,0072	RO
116	FLASH 4K 擦除	FLASHERASE4K	0x000,0073	WO
117	FLASH 32K 擦除	FLASHERASE32K	0x000,0074	WO
118	FLASH 64K 擦除	FLASHERASE64K	0x000,0075	WO
119	FLASH 全片擦除	FLASHERASEALL	0x000,0076	WO
120~ 133	保留	—	0x000,0077~ 0x000,0084	—
134	INTERFACE_EN	INTERFACE_EN	0x000,0085	RW
135	保留	—	0x000,0086	—
136	保留	—	0x000,0089	—
137	调试寄存器 0	MCU_DEBUG0	0x000,008a	RW
138	调试寄存器 1	MCU_DEBUG1	0x000,008b	RW
139	调试寄存器 2	MCU_DEBUG2	0x000,008c	RW
140	调试寄存器 3	MCU_DEBUG3	0x000,008d	RW
141	调试寄存器 4	MCU_DEBUG4	0x000,008e	RW
142	调试寄存器 5	MCU_DEBUG5	0x000,008f	RW
143	调试寄存器 6	MCU_DEBUG6	0x000,0090	RW
144	调试寄存器 7	MCU_DEBUG7	0x000,0091	RW
145	存控配置寄存器	MC_CONFIG	0x000,0092	RW
146	睡眠等待阈值寄存器	CoreSleepCnt	0x000,0093	RW
147	深度睡眠寄存器	DeepSleepFlag	0x000,0094	RW
148	Flash 取消写保护数据寄存器	flash_wrprotect_data	0x000,0095	RW
149	保留	—	0x000,0096	—
150	MCU 接收 INTPU 方向请求或 响应单错次数统计寄存器	INTPUSERR_CNT	0x000,0097	RW

151	MCU 接收 IRU1 方向请求或响应单错次数统计寄存器	IRU1SERR_CNT	0x000,0098	RW
152	芯片状态寄存器	MT_STATE1	0x000,0099	RO
153	CHIP_ID 寄存器	CHIP_CODE	0x000,009a	RO
154	App_Ltssm_En	App_Ltssm_En	0x000,009b	RW

注 1：“黑体标识的部分触发器”为申威 421 处理器在申威 411 处理器基础上增加的寄存器。此外，与申威 411 处理器相比，修改 BIST_CTL、MT_STATE、CG_ONLINE、MC_ONLINE、CORE_ONLINE、BIST_STAT。

注 2：此外以下寄存器相当于申威 411 处理器有所更新：

- CORE_ONLINE、MC_ONLINE；
- INT_CTRL；
- BIST_CTRL、BIST_OVERTIME_TH、BIST_STAT；
- MT_STAT、ERR_INF、SI_FAULT_STAT。

1.2.5.2 I2C 接口的 IO 寄存器

I2C 接口的 IO 寄存器编址如表 1-7 所示，其中地址[6:0]要求为全零，地址[33:23]要求为全零。

表 1-7：I2C 接口的 IO 寄存器编址

序号	名称	符号	索引地址 PA[22:7]	存取特性
1	控制寄存器	IC_CON	0x00	RW
2	目标地址寄存器	IC_TAR	0x04	RW
3	从设备地址寄存器	IC_SAR	0x08	RW
4	高速 master 模式编码地址寄存器	IC_HS_MADDR	0x0c	RW
5	发送和接收数据命令寄存器	IC_DATA_CMD	0x10	RW
6	标准速度 SCL 高电平计数寄存器	IC_SS_HCNT	0x14	RW
7	标准速度 SCL 低电平计数寄存器	IC_SS_LCNT	0x18	RW
8	快速 SCL 高电平计数寄存器	IC_FS_HCNT	0x1c	RW
9	快速 SCL 低电平计数寄存器	IC_FS_LCNT	0x20	RW

	器			
10	高速 SCL 高电平计数寄存器	IC_HS_HCNT	0x24	RW
11	高速 SCL 低电平计数寄存器	IC_HS_LCNT	0x28	RW
12	中断状态寄存器	IC_INTR_STAT	0x2c	RO
13	中断掩码寄存器	IC_INTR_MASK	0x30	RW
14	原始中断状态寄存器	IC_RAW_INTR_STAT	0x34	RO
15	接收 FIFO 阈值寄存器	IC_RX_TL	0x38	RW
16	发送 FIFO 阈值寄存器	IC_TX_TL	0x3c	RW
17	中断清除寄存器	IC_CLR_INTR	0x40	RC
18	清除 RX_UNDER 中断寄存器	IC_CLR_RX_UNDER	0x44	RC
19	清除 RX_OVER 中断寄存器	IC_CLR_RX_OVER	0x48	RC
20	清除 TX_OVER 中断寄存器	IC_CLR_TX_OVER	0x4c	RC
21	清除 RD_REQ 中断寄存器	IC_CLR_RD_REQ	0x50	RC
22	清除 RD_REQ 中断寄存器	IC_CLR_TX_ABRT	0x54	RC
23	清除 RX_DONE 中断寄存器	IC_CLR_RX_DONE	0x58	RC
24	清除 ACTIVITY 中断寄存器	IC_CLR_ACTIVITY	0x5c	RC
25	清除 STOP_DET 中断寄存器	IC_CLR_STOP_DET	0x60	RC
26	清除 START_DET 中断寄存器	IC_CLR_START_DET	0x64	RC
27	清除 GEN_CALL 中断寄存器	IC_CLR_GEN_CALL	0x68	RC
28	总线有效寄存器	IC_ENABLE	0x6c	RW
29	总线状态寄存器	IC_STATUS	0x70	RW
30	发送 FIFO 标记寄存器	IC_TXFLR	0x74	RO
31	接收 FIFO 标记寄存器	IC_RXFLR	0x78	RO
32	数据线保持时间寄存器	IC_SDA_HOLD	0x7c	RW
33	传输中断源寄存器	IC_TX_ABRT_SOURCE	0x80	RO
34	产生从设备数据 NACK 寄	IC_SLV_DATA_NACK	0x84	RW

	寄存器	_ONLY		
35	DMA 控制寄存器	IC_DMA_CR	0x88	RW
36	DMA 发送数据标记寄存器	IC_DMA_TDLR	0x8c	RW
37	DMA 接收数据标记寄存器	IC_DMA_RDLR	0x90	RW
38	数据线建立时间寄存器	IC_SDA_SETUP	0x94	RW
39	响应广播寄存器	IC_ACK_GENERAL_C ALL	0x98	RW
40	有效状态寄存器	IC_ENABLE_STATUS	0x9c	RO
41	参数寄存器 1	IC_COMP_PARAM_1	0xf4	RO
42	版本寄存器	IC_COMP_VERSION	0xf8	RO
43	组成方式寄存器	IC_COMP_TYPE	0xfc	RO

注：该寄存器为申威 421 处理器新增。

1.2.5.3 Flash 空间

申威 421 处理器芯片集成 16MB 的 Flash 空间，编址在 MCU 的 IO 空间上。

SPI Flash 空间的每个 64 位对应一个 64 位 IO 寄存器（SPI Flash 空间仅支持对界的 8B 访问），由于 MCU 的 IO 空间按 Cache 块编址，所以物理地址 {PA[27:7], 3'b000}（共 24 位地址）作为 SPI Flash 空间的访问地址（SPI Flash 空间最大支持 16MB）。

假设要访问 SPI Flash 的地址 Addr_a（Flash 域），需要把 Addr_a 左移四位（假设 0x8 这个 Flash 域地址，左移四位为 0x80 作为核心或维护访问的地址），并且 PA[38:35] 拼上 4'b1011、PA[47] 拼上 1'b1（核心视角，详见 1.2.1 节）或者 PA[39] 拼上 1'b1（维护视角，详见 1.2.1 节）得到 IO 访问地址。无论核心 IO 访问或者维护请求都需要按照该方式生成地址 [33:0]。

1.2.6 PCI-E接口的IO空间划分与寄存器编址

申威 421 处理器处理器共集成了两个 PCI-E 接口，这两个接口的 IO 地址空间以 PA[38:36] 进行区分，PCI-E0 接口的 IO 空间对应 PA[38:36] 为 0x4，PCI-E1 接口的 IO 空间对应 PA[38:36] 为 0x7。本文档中 PCI-E 接口中寄存器编址以 PCI-E0 接口为例，PCI-E1 接口类似。每个 PCI-E 接口内部，再按照 PA[35:0] 可将该空间划分为以下 6 个子空间：

- 1) PCI-E 终端设备（EP, End Point）的 32 位存储器空间；
- 2) PCI-E 终端设备的 64 位存储器空间；
- 3) PCI-E 接口控制部件（PIU, PCI-E Interface Unit）的 IO 空间；
- 4) PCI-E 终端设备的 IO 空间；

- 5) PCI-E 根部件 (RC, Root Complex) 的配置空间;
- 6) PCI-E 终端设备的配置空间。

其中 PIU 和 RC 集成在申威 421 处理器内部, EP 是通过 PCI-E 链路与 RC 相连的外部设备。具体划分如表 1-8 所示。

表 1-8: PCI-E 接口内部 IO 空间划分

PA[35:34]	其它	
2'b00	PA[33:32]为“0”	EP 的 32 位存储器空间。
	PA[33:32]为“1”、“2”、“3”	EP 的 64 位存储器空间。
2'b01	PA[33]为“0”	PIU 内部 IO 空间。该空间又划分成两个部分——PA [32] 为 0 为 IOR0 地址空间; PA [32]为 1 为 IOR1 地址空间。寄存器索引地址为 PA[13:7], PA[6:0]为全 0。 PIU 内部 IO 空间按 1024 位编址 (128B 对界)。
	PA[33]为“1”	EP 的 IO 空间。PA[32]取值被忽略, 因此有效空间只有 4GB。EP IO 空间按 32 位编址 (4B 对界)。
2'b10	PA[33]为“0”	RC 配置空间。寄存器索引地址为 PA[23:7], 内部划分如下: PA[23:21]: 功能号; PA[16:7]: 寄存器号; PA[6:0]: 全 0; 其它: 保留。 RC 配置空间按 1024 位编址 (128B 对界)。
	PA[33]为“1”	EP 配置空间。内部地址编址为: PA[32]: 保留; PA[31:24]: 总线号; PA[23:19]: 设备号; PA[18:16]: 功能号; PA[15:12]: 保留; PA[11:8]: 扩展寄存器号; PA[7:2]: 寄存器号。 EP 配置空间按 32 位编址 (4B 对界)。

说明:

- 1) EP 的存储器空间划分在连续的 16GB 空间范围内;
- 2) 当访问 EP 的配置空间时, 采用 TYPE0 配置请求访问 “非桥设备”, 采用 TYPE1 配置请求访问“交叉开关”或“桥设备”。配置请求类型由 PCIe 接口硬件根据访问总线号(PA[31:24])

判断后自动生成：如果 PA[31:24]等于 RC 的二级总线号，则生成 Type0 配置请求；如果 PA[31:24]大于 RC 的二级总线号，小于等于从属总线号，则生成 Type1 配置请求。

对于 EP 的地址空间，共有以下 5 类子空间，其地址范围（包括申威 421 处理器编址和 PCI-E 标准编址的对应关系）、空间大小如表 1-9 所示。

表 1-9: EP 的地址空间划分

EP 地址空间类型	申威 421 处理器编址范围(MCU 视角)	PCI-E 标准编址范围
32 位存储器空间	0xC0,0000,0000~0xC0,FFFF,FFFF (4GB)	0x0,0000,0000~0x0,FFFF,FFFF
64 位存储器空间	0xC1,0000,0000~0xC3,FFFF,FFFF (12GB)	0x1,0000,0000~0x3,FFFF,FFFF
EP IO 空间	0xC6,0000,0000~0xC6,FFFF,FFFF (4GB)	0x0000,0000~0xFFFF,FFFF
EP 配置空间	0xCA,0000,0000~0xCA,FFFF,FFFF (4GB)	0x0000,0000~0xFFFF,FFFF

说明:

- 1) EP 的 32 位存储器空间共 4GB 大小，发给 EP 的有效地址截取 PA[31:0]即可；
- 2) EP 的 64 位存储器空间共 12GB 大小，发给 EP 的有效地址截取 PA[33:0]即可；
- 3) EP 的 IO 空间范围为 8GB，但由于其最大只能扩展到 4GB（受 PCI-E 请求数据包中地址域的位宽所限），因此发给 EP 的有效地址只需截取 PA[31:0]即可，PA[32]被忽略；
- 4) 申威 421 处理器编址和 PCI-E 标准编址的转换由 PIU 实现。
- 5) 该表格中的申威 421 处理器编址范围是按 MCU 视角给出，对于核心视角而言，地址高位（PA[47]和 PA[39]的含义）有所区别。详见 1.2.1 节。

PIU 和 RC 内部的寄存器都是按 1024 位编址的。其中，PIU 内部的 IOR0 寄存器、IOR1 部分寄存器编址如表 1-10 和表 1-11 所示。RC 内部的 IO 寄存器编址及访问方式详见 5.2 节。

表 1-10: PIU 内部 IOR0 部分寄存器编址

序号	名称	符号	索引地址 [15:7]	存取特性
1	PIU 配置寄存器 0	PIUConfig0	0x00	RWS
2	EP DMA 请求基址寄存器	EPDMABAR	0x01	RW
3~ 66	IOMMU 段式代换段表寄存器 0~63	IOMMUSegItem~63	0x02~0x41 (由低到高对应寄存器 0~63)	RW

67	IOMMU 地址代换异常控制寄存器	IOMMUExcpt_Ctrl	0x42	RW
68	MSI 地址寄存器	MSIAddr	0x43	RW
69~ 324	MSI 中断配置寄存器 0~255	MSIConfig0~255	0x44~0x143 (由低到高对 应寄存器 0~255)	RW
325	INTA 中断配置寄存器	INTAConfig	0x144	RW
326	INTB 中断配置寄存器	INTBConfig	0x145	RW
327	INTC 中断配置寄存器	INTCConfig	0x146	RW
328	INTD 中断配置寄存器	INTDConfig	0x147	RW
329	高级错误报告错 INT 中断配置寄存器	aerErrIntConfig	0x148	RW
330	高级错误报告错 MSI 中断配置寄存器	aerErrMsiConfig	0x149	RW
331	PME INT 中断配置寄存器	pmeIntConfig	0x14a	RW
332	PME MSI 中断配置寄存器	pmeMsiConfig	0x14b	RW
333	热插拔 INT 中断配置寄存器	hpIntConfig	0x14c	RW
334	热插拔 MSI 中断配置寄存器	hpMsiConfig	0x14d	RW
335	链路自动带宽调节 INT 中断配置寄存器	linkAutoBwIntConfig	0x14e	RW
336	链路带宽管理 INT 中断配置寄存器	bwMgtIntConfig	0x14f	RW
337	DMAW 请求/响应计数寄存器	DMAWReqCplCnt	0x150	RW
338	DMAR 请求/响应计数寄存器	DMARReqCplCnt	0x151	RW
339	MSI 中断请求/处理计数寄存器	MSIReqClrCnt	0x152	RW
340	INTX 中断请求/处理计数寄存器	INTXReqClrCnt	0x153	RW
341	页式代换访存请求/响应计数寄存器	PageMemReqCplCnt	0x154	RW
342~ 352	保留	---	0x155~0x15f	---
353	页式代换设备表基址寄存器	DTBaseAddr	0x160	RW
354	设备表 TLB 全刷新寄存器	DTLB_FlushAll	0x161	RW
355	指定设备号设备 TLB 表刷新寄存器	DTLB_FlushDev	0x162	RW
356	页表 TLB 全刷新寄存器	PTLB_FlushAll	0x163	RW
357	指定设备号页表 TLB 刷新寄存器	PTLB_FlushDev	0x164	RW
358	指定设备和虚地址页表 TLB 刷新寄存器	PTLB_FlushVAddr	0x165	RW
359	页表 Cache 全刷新寄存器	PCache_FlushAll	0x166	RW
360	指定设备和索引页表 Cache 刷新寄存器	PCache_FlushDev	0x167	RW

361	指定物理地址页表 Cache 刷新寄存器	PCache_FlushPAddr	0x168	RW
362	超时配置寄存器	Timeout_Config	0x169	RW
363	IOMMU 地址代换异常状态寄存器	IOMMUExcpt_Status	0x16a	RW
364	IOMMU 页式地址代换地址寄存器 1	IOMMUPage_PAddr1	0x16b	RO
365	IOMMU 页式地址代换地址寄存器 2	IOMMUPage_PAddr2	0x16c	RO
366	IOMMU 页式地址代换地址寄存器 3	IOMMUPage_PAddr3	0x16d	RO

表 1-11: PIU 内部 IOR1 部分寄存器编址

序号	名称	符号	索引地址 [15:7]	存取 特性
1	PIU 配置寄存器 1	PIUConfig1	0x00	RWS
2	PIO 请求/响应计数器	PIOReqCplCnt	0x01	RW
3	RC 请求/响应计数器	RCReqCplCnt	0x02	RW
4	设备 NP 类请求/响应计数器	EPNPReqCplCnt	0x03	RW
5	IOR0 请求/响应计数器	IOR0ReqCplCnt	0x04	RW
6	IOR1 请求/响应计数器	IOR1ReqCplCnt	0x05	RW
7~ 11	LTSSM 链路训练最近状态寄存器 0~4	NewLTSSMState0~4	0x06~0xa	RO
12~ 16	LTSSM 链路训练最旧状态寄存器 0~4	OldLTSSMState0~4	0x0b~0xf	RO
17	错误登记寄存器	ErrReport	0x10	RW
18	错误使能寄存器	ErrEnable	0x11	RW
19	DMA 响应 ECC 单错计数器	DMACpl_SErrCnt	0x12	RWC
20	PIO 请求 ECC 单错计数器	PIOReq_SErrCnt	0x13	RWC
21	MCU 请求 ECC 单错计数器	MCUReq_SErrCnt	0x14	RWC
22	ECC 单错计数阈值寄存器	ECCSErrLimit	0x15	RW
23	RC 控制寄存器	RCCControl	0x16	RW
24	RC 电源管理寄存器	RCPowerManage	0x17	RW
25	RC 调试信息寄存器 0	RCDebugInf0	0x18	RO
26	RC 调试信息寄存器 1	RCDebugInf1	0x19	RO
27	RC 电气特性控制寄存器	RCElectroControl	0x1a	RW
28	RC 电气特性状态寄存器	RCElectroStatus	0x1b	RO
29	RC 复位寄存器	RCHotRst	0x1c	RW
30	MSG 控制寄存器 0	MSGControl0	0x1d	RW

31	MSG 控制寄存器 1	MSGControl1	0x1e	RW
32	链路复位延迟控制寄存器	LinkDownDelay	0x1f	RW
33	PHY 控制寄存器 0	PMAControl0	0x20	RWS
34	页表 Cache 数据单错计数寄存器	PCData_SErrCounter	0x21	RWC
35	页表 Cache TAG 偶校验错计数寄存器	PCTag_PErrCounter	0x22	RWC
36	PHY 控制寄存器 1	PMAControl1	0x23	RWC
37	PHY 控制寄存器 2	PMAControl2	0x24	RWC
38	PHY 控制寄存器 3	PMAControl3	0x25	RWC
39	PHY 控制寄存器 4	PMAControl4	0x26	RWC

注 1：“黑体标识的部分触发器”为申威 421 处理器在申威 411 处理器基础上增加的寄存器。

注 2：此外，与申威 411 处理器相比，以下寄存器进行了更新：

- PIUConfig0、PIUConfig1、RCDebugInf0/1。

1.3 IO 寄存器的访问机制

IO 寄存器编址在申威 421 处理器的 IO 空间上，软件可以通过 IO 空间的 Load、Store 指令显式访问 IO 寄存器。维护接口也可以通过维护命令显式访问 IO 寄存器。对维护接口内的 IO 寄存器而言，维护接口的权限通常比软件高，其它 IO 寄存器的属性则对维护接口和软件访问权限相同。

申威 421 处理器保证每个核心对 IO 空间的读写操作不能推测进行，这就保证了软件显式读写 IO 寄存器之间的顺序。维护命令只能串行执行，这也保证了维护接口显式读写 IO 寄存器之间的顺序。申威 421 处理器不保证软件对 IO 寄存器的显式读写与维护接口对 IO 寄存器的显式读写之间的顺序，需要软件和系统人员在高层进行控制。

申威 421 处理器不保证显式写 IO 寄存器和隐式读 IO 寄存器之间的顺序关系，需要软件通过同步方式保证显式写完成之后，IO 寄存器的新值能对后续请求的处理产生影响。如对存控的配置寄存器进行写操作前，首先软件必须保证没有访存操作正在执行，显示写执行完毕后，才能开始后续的访存操作，如果软件无法同步 4 个核心和 PCI-E0、PCI-E1、维护接口的访存行为，则对相关 IO 寄存器寄存器的配置只能在系统引导（Boot）之前通过维护接口或者初始化加载到 ICache 中的程序进行配置，配置完成后不允许再修改。

2 存储器接口的 IO 寄存器

2.1 一致性处理部件的 IO 寄存器

2.1.1 基地址说明

表 2-1: 一致性处理部件 IO 寄存器基地址说明

模块	核心视角(PA[47:0])	维护视角(PA[39:0])	备注
核组 0	0x8000,0000,0000	0x80,0000,0000	
核组 1	0x8010,0000,0000	0x90,0000,0000	
核组 2	0x8020,0000,0000	0xa0,0000,0000	
核组 3	0x8030,0000,0000	0xb0,0000,0000	

注: 各寄存器的偏移地址(相对于零地址)见具体寄存器说明。

2.1.2 CPM_MC0Disable / CPM_MC1Disable

偏移地址: 16'h0000/16'h0080

CPM_MC0Disable 和 CPM_MC1Disable 是一致性处理部件的 MC 不使能配置寄存器, 维护和核心可读写, 复位为默认配置值。

表 2-2: CPM_MC*Disable 寄存器域的描述

名称	范围	类型	描述
MCDisable	[0]	WR,0	禁止存储器访问标志, 为“1”表示不允许接收访问主存的请求, 如果接收到访问主存的请求, 存控将直接返回 ReadDataError, 初始值为“0”。该标志不限制对 MC 的 IO 访问。 要求软件在 SROM 加载程序一开始, 就将该位写为“1”, 等存控进行 Data training 结束后, 再将该位写为“0”。
—	其它	—	保留。

2.1.3 CPM_MCCrossMode

偏移地址: 16'h0100

CPM_MCCrossMode 是同一个核组的两个 MC 之间的地址交叉模式配置寄存器, 维护和核心可

读写，复位为默认配置值。单个 MC 工作时，该寄存器不起作用，CPM 固定将按“最高位地址交叉”模式处理请求。

表 2-3: CPM_MCCrossMode 寄存器域的描述

名称	范围	类型	描述
MCCrossMode	[0]	WR,0	两个 MC 的地址交叉模式，分为 2 种： <ul style="list-style-type: none"> • 0: 按奇/偶 Cache 块(Addr[7])划分两个 MC; (默认值) • 1: 按最高地址划分两个 MC, 即两个 MC 连续编址。 该位地址与 MC#0 的容量有关;
—	其它	—	保留。

注：与申威 411 处理器相比，减少了“8KB 页面交叉模式”和“1MB 页面交叉模式”。

2.1.4 CPM_TCWayMask

偏移地址：16'h0180

CPM_TCWayMask 是本核组中 32 路三级 Cache(TCache)的按路屏蔽使能配置寄存器，维护和核心可读写，复位为“不屏蔽”。TCache Bist 逻辑在系统启动时会对 TCache 阵列进行读写测试后，而后再将检测结果加载到该寄存器中，用于关闭无法修复的 TCache 路。

表 2-4: CPM_TCWayMask 寄存器域的描述

名称	范围	类型	描述
TCWayMask	[31:0]	WR,0	TCWayMask[i]对应第 i 路 TCache <ul style="list-style-type: none"> • '0': 打开第#i 路 TCache (默认值) • '1': 关闭第#i 路 TCache
—	其它	—	保留。

2.1.5 CPM_FRQCmprEn

偏移地址：16'h0200

CPM_FRQCmprEn 是一致性处理部件的访存一次请求队列 (FRQ) 压缩使能配置寄存器，维护和核心可读写，复位为默认配置值。

表 2-5: CPM_FRQCmprEn 寄存器域的描述

名称	范围	类型	描述
FRQCmprEn	[0]	WR,1	<ul style="list-style-type: none"> • '0': 不允许对 FRQ 进行队列压缩; • '1': 允许对 FRQ 进行有限的队列压缩; (默认值)
—	其它	—	保留。

2.1.6 CPM_PerfCtrl

偏移地址：16'h0280

CPM_PerfCtrl 是一致性处理部件中多个性能计数器的控制寄存器，维护和核心可读写，复位为默认配置值。

表 2-6: CPM_PerfCtrl 寄存器域的描述

名称	范围	类型	描述
PerfCtrl	[1:0]	WR,1	根据写入值控制 CPM 各个性能计数器的操作： <ul style="list-style-type: none"> • 0: 清除性能计数器； • 1: 启动性能计数器；（默认值） • 2: 停止性能计数器； • 其他值: 停止性能计数器
—	其它	—	保留。

CPM 性能计数器的使用流程：

- 1) 写“性能计数器控制寄存器”为 0，清除性能计数器。如果刚复位结束，可省略此步骤；
- 2) 写“性能计数器控制寄存器”为 1，启动性能计数器。如果刚复位结束，可省略此步骤；
- 3) 写“性能计数器控制寄存器”为 2，停止性能记数。

2.1.7 CPM_SerrThres

偏移地址：16'h0400

CPM_SerrTh 是单错计数阈值。

表 2-7: CPM_SerrThres 寄存器域的描述

名称	范围	类型	描述
SerrThres	[15:0]	RW,0xffff	CPM单错预警阈值。如果单错统计计数器的计数值大于该阈值，则CPM向MCU报SERR错误。
—	其它	—	保留。

2.1.8 CPM_SerrCnt

偏移地址：16'h0480

CPM_SerrCnt 是 CPM 内部检测到的各种单错的累计计数值，软件写该寄存器（任意值）时清 0。

表 2-8: CPM_SerrCnt 寄存器域的描述

名称	范围	类型	描述
----	----	----	----

TData_SerrCnt	[15:0]	RWC,0	对 TData 阵列的输出数据进行 ECC 单错计数，到达最大值时停止计数。 软件写该寄存器时清 0。
CPM_SerrCnt	[47:32]	RWC,0	对 CPM 内部检测到的 ECC 单错进行计数，不含 TData 阵列的 ECC 单错。到达最大值时停止计数。 软件写该寄存器时清 0。
—	其它	—	保留。

2.1.9 CPM_Serr

偏移地址：16'h0500

CPM_Serr 是一致性处理部件 ECC 单错状态寄存器，记录 CPM 发现的阵列或数据通路的 ECC 单错。该寄存器按位写'1'清'0'。

表 2-9: CPM_Serr 寄存器域的描述

名称	范围	类型	描述
CPM_SErr	[10]	RW1C, 0	DRspBufA/DRspBufB/R2LFRD 读数据 ECC 多错； 错误原因：可能是核心回答数据、IPU 写数据、数据环网或数据环网接收缓冲(DRspBuf*/R2LFRD)出现单错。
	[9]	RW1C, 0	IO 请求数据阵列(IODQ)ECC 单错
	[8]	RW1C, 0	TMAD1 ECC 单错
	[7:4]	RW1C, 0	核心#3~ #0 回答数据 ECC 单错(目前接 0)
	[3:2]	RW1C, 0	MC#1/ MC#0 读数据 ECC 单错
	[1]	RW1C, 0	TTag 阵列 ECC 单错
	[0]	RW1C, 0	TData 阵列 ECC 单错
—	其它	—	保留。

2.1.10 CPM_FaultSpot

偏移地址：16'h0580

CPM_FaultSpot 登记了一致性处理部件检测到的部分故障的错误现场，比如一致性协议错、IO 地址非法等。该寄存器只登记第一个故障的错误现场，软件写任意值时将有效位清 0。

表 2-10: CPM_FaultSpot 寄存器域的描述

名称	范围	类型	描述
SpotValid	[63]	RWC, 0	故障现场有效位，复位时清0;

FaultType	[62:60]	RO, X	故障类型，在SpotValid='1'时有意义： <ul style="list-style-type: none"> • 3'h1: 独占核心发送可Cache请求(一致性状态错) • 3'h2: 请求命中多个TTag条目错(一致性状态错) • 3'h3: 在流水线ST4站台发现控制错 • 3'h5: IO请求地址非法 • 3'h6: IO请求发生控制错 • 其余编码: 保留
FaultSpot	[59:0]	RO, X	各种故障类型对应的错误现场，在SpotValid='1'时有意义。 <ul style="list-style-type: none"> • [51:50]: 命中TTag状态(一致性状态错) • [49:40]: 出错请求的SrcInf[9:0] • [39:36]: AckType(一致性状态错)/Addr[39:36](IO请求错) • [35:7]: Addr[35:7] • [5:0]: RQ0Type

请求源信息 SrcInf[9:0]:

SrcInf	[9:7]	[6:0]
含义	<ul style="list-style-type: none"> • 0~3: 核组#0~#3 • 4: PCIE#0 • 5: MCU • 6: INTPU • 7: PCIE#1 	<ul style="list-style-type: none"> • 对核组而言，SrcInf[6:5]为核心号，[4:0]为 SMAF 号； • 对 PCIE/MCU/INTPU 而言，SrcInf[6:0]为悬挂号

2.1.11 CPM_Fault

偏移地址: 16'h0600

CPM_Fault 是一致性处理部件的故障状态寄存器，包括 CPM 发现的不可纠错（ECC 多错或偶校验错）、硬件错误或协议错误。该寄存器按位写'1'清'0'，写'0'保持相应位不变。

表 2-11: CPM_Fault 寄存器域的描述

名称	范围	类型	描述
CPM 硬件故障			
CPM_CtrlErr	[63]	RW1C, 0	IOQ 收到不期望的数据回答
	[62]	RW1C, 0	IOQ 收到无数据回答(IOQ 只会向核心发送“取数释放缓冲”请求)
	[61]	RW1C, 0	IPU 到 CPM 的写数据缓冲(R2LFRD)的二次请求队列溢出
	[60]	RW1C, 0	核心组二次请求队列 (SRQ) 溢出错
	[59]	/	保留

	[58]	RW1C, 0	TData 端口发生读写冲突;
	[57]	RW1C, 0	TTag 端口发生读写冲突;
	[56]	RW1C, 0	二次请求目标错;(非置无效类二次请求的目标核心有多个)
	[55]	RW1C, 0	请求命中多个 TTag 条目; 故障现场见 CPM_FaultSpot;
	[54]	RW1C, 0	请求命中 TTag 状态错(脏独占 Local 核心向 CPM 发送指令流读请求或可 Cache 读写请求); 故障现场见 CPM_FaultSpot;
	[53]	RW1C, 0	TCache 淘汰请求地址非法(含 EvictTCacheBlk 和 TCache 挤占淘汰的地址);
	[52]	RW1C, 0	多个 TMAF 条目全地址相同(非 EvictECBlk 请求命中多个 TMAF1 条目的全地址相同);
	[51]	RW1C, 0	TMAF2 接收到不期望的回答或响应;
	[50]	RW1C, 0	TMAF2 满溢出;
	[49]	RW1C, 0	TMAF1 接收到不期望的回答或响应;
	[48]	RW1C, 0	TMAF1 满溢出;
	[47]	RW1C, 0	MAQ#1 满溢出;
	[46]	RW1C, 0	CPM-MC#1 读响应协议错(主存读数据不是 4 个 flit, 或 IO 读数据不是 1 个 flit, 或 4 个主存读数据携带的 CPM ID 号不一致);
	[45]	RW1C, 0	MAQ#0 满溢出
	[44]	RW1C, 0	CPM-MC#0 读响应协议错
	[43:40]	RW1C, 0	睡眠核心#3-#0 收到非置无效类二次请求
	[39:36]	RW1C, 0	核心#3~ #0 二次请求信用上溢;
	[35:32]	RW1C, 0	核心#3~ #0 一次请求类型错;
CPM 单错预警			
CPMSErrWarn	[31]	RW1C, 0	CPM(不含 TData)单错预警, CPM_SerrCnt[47:32]超过阈值;
TDSErrWarn	[30]	RW1C, 0	TData 阵列单错预警, CPM_SerrCnt[15:0]超过阈值;
CPM 偶校验错(不可纠错)			
CPM_ParityEr	[29:28]	/	保留
r	[27]	RW1C, 0	TDATA 修复地址 Latch 偶校验错

	[26]	RW1C, 0	TMAD2 偶校验错
	[25:24]	RW1C, 0	MC#1/ MC#0 读控制偶校验错
	[23:20]	RW1C, 0	核心#3~ #0 回答控制偶校验错
	[19:16]	RW1C, 0	核心#3~ #0 请求控制偶校验错
CPM ECC 多错 (不可纠错)			
CPM_MErr	[15:11]	/	保留
	[10]	RW1C, 0	DRspBufA/DRspBufB/R2LFRD 读数据 ECC 多错; 错误原因: 可能是核心回答数据、IPU 写数据、数据环网或数据环网接收缓冲(DRspBuf*/R2LFRD)出现不可纠错。
	[9]	RW1C, 0	IO 请求数据阵列(IODQ)ECC 多错
	[8]	RW1C, 0	TMAD1 ECC 多错
	[7:4]	RW1C, 0	核心#3~ #0 回答数据 ECC 多错 (目前接 0)
	[3:2]	RW1C, 0	MC#1/ MC#0 读数据 ECC 多错
	[1]	RW1C, 0	TTag 阵列 ECC 多错
	[0]	RW1C, 0	TData 阵列 ECC 多错

2.1.12 CPM_FaultEn

偏移地址: 16'h0680

CPM_FaultEn 是一致性处理部件的故障使能寄存器, 按位屏蔽 CPM 内的各种错误(CPM_Fault 寄存器)是否报告给 MCU。

表 2-12: CPM_FaultEn 寄存器域的描述

名称	范围	类型	描述
CPMFAULTEN	[63:0]	RW,全 0	CPM 内各个报错信号的使能标志, 默认关闭报错使能, 按位对应 CPM 故障状态寄存器 CPM_Fault;

2.1.13 RN_Fault

偏移地址: 16'h0700

RN_Fault 是核组环网节点的故障状态寄存器, 包括 CPM 与环网节点接口发现的不可纠错、硬件错误或协议错误。该寄存器按位写'1'清'0', 写'0'保持相应位不变。

注: 对于连续 2 位“顺/逆指针环网”域, 低位表示“顺时针”, 高位表示“逆时针”。

表 2-13: RN_Fault 寄存器说明

名称	范围	类型	描述
数据环网故障			
RN_DTErr	[22]	RW1C, 0	来自 CT 环网的核心 ReqComplete 响应计数器上溢
	[21]	RW1C, 0	DT 环网释放无效响应数据缓冲错
	[20]	RW1C, 0	DT 环网数据回答信用计数器上溢
	[19]	RW1C, 0	响应数据缓冲 AckDBuf 写操作错
	[18]	RW1C, 0	IPU 到 CPM 的写请求数据缓冲(R2LFRD)写操作错
	[17:16]	RW1C, 0	顺/逆时针 DT 环网回答数据接收缓冲 DRspBuf 上溢
一致性环网故障			
RN_SAErr	[15:14]	/	保留
	[13]	RW1C, 0	CPM 向不在位核组发送二次请求或响应
	[12]	RW1C, 0	SA 环网“二次请求&响应控制”信用计数器上溢
	[11:10]	RW1C, 0	顺/逆时针 SA 环网接收缓冲 SABUF 上溢
	[9:8]	RW1C, 0	顺/逆时针 SA 环网接收 flit 偶校验错
请求环网故障			
RN_RQErr	[7:5]	/	保留
	[4]	RW1C, 0	RQ 环网请求信用计数器上溢
	[3:2]	RW1C, 0	顺/逆时针 RQ 环网接收缓冲 RQBUF 上溢
	[1:0]	RW1C, 0	顺/逆时针 RQ 环网接收 flit 偶校验错
—	其它	—	保留, 不可写, 读出数据固定为0。

2.1.14 RN_FaultEn

偏移地址: 16h0780

RN_FaultEn 是核组环网节点的故障使能寄存器, 按位屏蔽 RN_Fault 寄存器的各类错误是否报告给 MCU。

表 2-14: RN_FaultEn 寄存器域的描述

名称	范围	类型	描述
----	----	----	----

RNFaultEn	[22:0]	RW, 0	环网节点内各个报错信号的使能标志, 默认关闭报错使能, 按位对应环网故障状态寄存器 RN_Fault;
—	其它	—	保留, 不可写, 读出数据固定为0。

2.1.15 CPM_TReqCnt

偏移地址: 16'h0800

CPM_TReqCnt 是一致性处理部件记录可 Cache 请求数量的计数器, 核心和维护只读。该计数器的启动、停止和清除操作需要通过 IO 写“CPM 性能计数器控制寄存器 (CPM_PerfCtrl)”来控制。

表 2-15: CPM_TReqCnt 寄存器域的描述

名称	范围	类型	描述
TReqCnt	[31:0]	RO	记录核心发向 CPM 的可 Cache 请求(可 Cache 读、可 Cache 写、置脏、原子操作) 的数量, 用于统计 TCache 命中率。该计数值到达最大值时, 可 Cache 请求计数器、TCache 直接命中计数器、TCache 脱靶计数器停止计数。
—	其它	—	保留, 读出数据固定为0

2.1.16 CPM_TCHitCnt

偏移地址: 16'h0880

CPM_TCHitCnt 是一致性处理部件记录 TCache 命中次数的计数器, 核心和维护只读。该计数器的启动、停止和清除操作需要通过 IO 写“CPM 性能计数器控制寄存器 (CPM_PerfCtrl)”来控制。

表 2-16: CPM_TCHitCnt 寄存器域的描述

名称	范围	类型	描述
TCHitCnt	[31:0]	RO	记录核心的可 Cache 请求直接命中 TCache (从 TCache 读取数据) 的次数。
—	其它	—	保留, 读出数据固定为0

2.1.17 CPM_TCMissCnt

偏移地址: 16'h0900

CPM_TCMissCnt 是一致性处理部件记录 TCache 脱靶次数的计数器, 核心和维护只读。该计数器的启动、停止和清除操作需要通过 IO 写“CPM 性能计数器控制寄存器 (CPM_PerfCtrl)”来控制。

表 2-17: CPM_TCMissCnt 寄存器域的描述

名称	范围	类型	描述
----	----	----	----

TCMissCnt	[31:0]	RO	记录核心的可 Cache 请求发生 TCache 脱靶（从主存读取数据）的次数。 TCache 命中率 = 1 - (TCache 脱靶计数/可 Cache 请求计数)
—	其它	—	保留，读出数据固定为0

2.1.18 CPM_AddrCflCnt

偏移地址：16'h0980

CPM_AddrCflCnt 是一致性处理部件记录一次请求在一致性流水线上发生地址冲突的次数（一次请求与 TMAF 条目的全地址相同，或与 FRQ 年老条目的 TCache 索引地址相同），核心和维护只读。该计数器的启动、停止和清除操作需要通过 IO 写“CPM 性能计数器控制寄存器（CPM_PerfCtrl）”来控制。

表 2-18: CPM_AddrCflCnt 寄存器域的描述

名称	范围	类型	描述
AddrCflCnt	[23:0]	RO	记录一致性流水线上的请求发生 TCache 索引冲突的次数。该计数值到达最大值时停止计数，维持最大值不变；
—	其它	—	保留，读出数据固定为0

2.1.19 CPM_FRQCmprCnt

偏移地址：16'h0A00

CPM_FRQCmprCnt 是一致性处理部件记录 FRQ 压缩性能的计数器，核心和维护只读。该计数器的启动、停止和清除操作需要通过 IO 写“CPM 性能计数器控制寄存器（CPM_PerfCtrl）”来控制。

表 2-19: CPM_FRQCmprCnt 寄存器域的描述

名称	范围	类型	描述
FRQCmprCnt	[23:0]	RO	记录 FRQ 尾指针回溯的条目数。该计数值到达最大值时停止计数，维持最大值不变；
—	其它	—	保留，读出数据固定为0

2.1.20 RN_BlockFlag

偏移地址：16'h0A80

RN_BlockFlag 记录核组连接的各套环网是否发生过阻塞操作。环网节点为每个环网设置了一个计时器，当本地上网请求被上游请求阻塞而无法发送时，计时器加 1。当计时器达到最大值时，相应环网将进行流量控制，并在 RN_BlockFlag 寄存器的相应位登记为'1'。该寄存器复位时为 0，不支持软件写。

表 2-20: RN_BlockFlag 寄存器域的描述

名称	范围	类型	描述
CT_BlockFlag	[7:6]	RO, 0	为'1'表示本核组的顺/逆时针信用环网节点曾经发生"阻塞超时";
DT_BlockFlag	[5:4]	RO, 0	为'1'表示本核组的顺/逆时针数据环网节点曾经发生"阻塞超时";
SA_BlockFlag	[3:2]	RO, 0	为'1'表示本核组的顺/逆时针一致性环网节点曾经发生"阻塞超时";
RQ_BlockFlag	[1:0]	RO, 0	为'1'表示本核组的顺/逆时针请求环网节点曾经发生"阻塞超时";
—	其它	—	保留, 读出数据固定为0。

2.2 存储控制器内部的 IO 寄存器

存储控制器的 IO 寄存器中, 有部分信息域用于控制各种命令的发送间隔。如无特殊说明, 这些信息域对应计数器的工作时钟均为“DDR3 存储器接口输出时钟”, 其复位后的初值, 是按照 JEDEC DDR3 标准中, DDR3-1600 (对应的“DDR3 存储器接口输出时钟”频率为 800MHz)、8Gb 存储器颗粒要求的典型配置值, 折算为“DDR3 存储器接口输出时钟”的周期数。在使用申威 421 处理器时, 应该根据主存的实际工作频率和颗粒容量, 参照存储器颗粒的相关手册进行配置。

每个存控对应的 DDR3 接口物理层电路中包含一套存储器地址/命令/数据控制接口, 用于控制 DDR3 接口的命令、地址和数据信息。地址/命令/数据控制接口包括 1 组地址/命令接口 (简称 AC) 和 9 组数据接口 (简称 Datx8), 每组 Datx8 对应 8 位数据。AC/Datx8 子模块统称 PHY, AC/Datx8 的公共控制模块统称 PUB。

2.2.1 基地址说明

表 2-21: 一致性处理部件 IO 寄存器基地址说明

模块	核心视角(PA[47:0])		维护视角(PA[39:0])		备注
	MC0	MC1	MC0	MC1	
核组 0	0x8000,0200,0000	0x8000,0300,0000	0x80,0200,0000	0x80,0300,0000	
核组 1	0x8010,0200,0000	0x8010,0300,0000	0x90,0200,0000	0x90,0300,0000	
核组 2	0x8020,0200,0000	0x8020,0300,0000	0xa0,0200,0000	0xa0,0300,0000	

核组 3	0x8030,0200,0000	0x8030,0300,0000	0xb0,0200,0000	0xb0,0300,0000	
------	------------------	------------------	----------------	----------------	--

注：各寄存器的偏移地址（相对于零地址）见具体寄存器说明。

2.2.2 CfgTR0

偏移地址：0x0000

CfgTR0为时序寄存器0，可读写，该寄存器用于控制DDR3刷新时序。

表 2-22: CfgTR0 寄存器域的描述

名称	范围	类型	描述
CfgREFI	[18:9]	RW,195	两次刷新命令之间的平均间隔周期数。（REFI计数器使用维护时钟进行计数，该时钟频率为25MHz时，初值“195”相当于7.8us）
CfgRFC	[8:0]	RW,280	REF命令到REF/ACT命令之间的最小间隔周期数。（该默认配置按8Gb的颗粒进行设置，即350ns）
—	其它	—	保留。

注：CfgRFC 的设置与存储器颗粒的容量有关，具体见下表：

表 2-23: CfgRFC 与存储器颗粒的关系

存储器颗粒容量	CfgRFC (ns)
512Mb	90
1Gb	110
2Gb	160
4Gb	300
8Gb	350

2.2.3 CfgTR1

偏移地址：0x0080

CfgTR1 为时序寄存器 1，可读写，该寄存器用于控制 DDR3 Speed Bins 时序。

表 2-24: CfgTR1 寄存器域的描述

名称	范围	类型	描述
CfgRAS	[23:18]	RW,28	ACT到PRE命令的最小间隔周期数。
CfgRC	[17:12]	RW,39	ACT到ACT/REF命令的最小间隔周期数。
CfgRP	[11:8]	RW,9	PRE命令最小间隔周期数。

CfgRCD	[7:4]	RW,9	ACT到读写命令（CAS）之间的最小间隔周期数。
CfgAA	[3:0]	RW,9	读命令到其第一个数据上数据总线的最小时间间隔。（相当于11.25ns@800MHz）。 注：该延时等同于CL（即tCfgAA/tCK）。SW1610具体实现时由CL进行延时控制；
—	其它	—	保留。

2.2.4 CfgTR2

偏移地址：0x0100

CfgTR2为时序寄存器2，可读写，该寄存器用于控制DDR3命令和地址时序。

表 2-25：CfgTR2 寄存器域的描述

名称	范围	类型	描述
CfgDQS_WW	[63:61]	RW,2	不同Rank之间写到写切换的额外延迟。
CfgDQS_RR	[60:58]	RW,2	不同Rank之间读到读切换的额外延迟。
CfgDQS_WR	[57:54]	RW,2	不同Rank之间写到读切换的额外延迟。
CfgCCDR	[53:51]	RW,4	RD到RD的最小间隔周期数。
CfgDQS_RW	[50:48]	RW,2	不同Rank之间读到写切换的额外延迟。
CfgRTW	[47:44]	RW,6	读到写切换延时控制。实际切换延时计数方式如下： （1）Rank0->Rank0：4+(RL-WL)+2+CfgRTW； （2）Rank0->Rank1：4+(RL-WL)+CfgDQS； （3）Rank1->Rank1：同（1）； （4）Rank1->Rank0：同（2）。
CfgFAW	[43:38]	RW,32	4个活动Bank的窗口时间周期。
CfgRRD	[37:34]	RW,6	ACT到ACT命令之间的最小间隔周期数（不同Bank）。
CfgCCDW	[33:31]	RW,4	WR到WR的最小间隔周期数。
CfgMOD	[30:26]	RW,12	模式寄存器更新延迟周期数。
CfgMRD	[25:23]	RW,4	模式寄存器设置时间，即2条MRS命令间的最小间隔周期。
CfgWR	[22:18]	RW,12	写恢复（Write Recovery）时间（周期数）。
CfgWTR	[17:14]	RW,6	同一个Bank内的写命令到读命令之间的最小间隔周期。
CfgRTP	[13:10]	RW,6	同一个Bank内的读命令到PRE命令之间的最小间隔周期。
CfgDLLK	[9:0]	RW,512	DLL锁定所需的最短时间（周期数）。
—	其它	—	保留。

2.2.5 CfgTR3

偏移地址：0x00180

CfgTR3 为时序寄存器 3，可读写，该寄存器用于控制端接校准（ZQ Calibraion）和初始化时序。

表 2-26: CfgTR3 寄存器域的描述

名称	范围	类型	描述
CfgXPR	[34:26]	RW,288	复位结束后，从CKE到第一个有效的命令之间的间隔。
CfgZQCS	[25:19]	RW,64	正常操作期间短校准时间（周期数）。
CfgZQoper	[18:10]	RW,256	正常操作期间完全校准时间（周期数）。
CfgZQinit	[9:0]	RW,512	上电和复位过程的校准时间（周期数）。
—	其它	—	保留。

2.2.6 CfgTR4

偏移地址：0x00200

CfgTR4为时序寄存器4，可读写，该寄存器用于控制自刷新时序。

表 2-27: CfgTR4 寄存器域的描述

名称	范围	类型	描述
CfgCKSRX	[30:27]	RW,8	自刷新、PowerDown退出或者复位退出前时钟必须有效最少时间。
CfgCKSRE	[26:23]	RW,8	进入自刷新或PowerDown后时钟仍需要保持有效的最少时间。
CfgCKESR	[22:19]	RW,7	自刷新期间CKE为低的最少周期数。
CfgXSDLL	[18:9]	RW,512	退出自刷新到需要锁定的DLL命令之间的最小间隔周期数。
CfgXS	[8:0]	RW,288	退出自刷新到不需要锁定的DLL命令间的最小间隔周期数。
—	其它	—	保留。

2.2.7 CfgTR5

偏移地址：0x00280

CfgTR5为时序寄存器5，可读写，该寄存器用于控制Power Down时序。

表 2-28: CfgTR5 寄存器域的描述

名称	范围	类型	描述
CfgWRAPDEN	[36:31]	RW,35	WRA命令到进入PowerDown的最小间隔周期。
CfgWRPDEN	[30:25]	RW,34	WR命令到进入PowerDown的最小间隔周期。
CfgRDPDEN	[24:20]	RW,22	RD/RDA命令到PowerDown进入的最小间隔周期。
CfgPD	[19:11]	RW,6	进入PowerDown到退出时间最小间隔周期数。该配置值不能小于CfgCKE。
CfgCKE	[10:8]	RW,6	CKE的最小脉冲宽度（周期数）。
CfgXPDLL	[7:3]	RW,20	退出Precharge PowerDown（DLL frozen）到需要一个锁定的DLL的命令之间的最小间隔周期数。
CfgXP	[2:0]	RW,6	退出PowerDown状态时，DLL打开到一个有效命令的最小间隔周期数，或者是从退出Precharge PowerDown（DLL frozen）到一个不需要锁定的DLL命令间的最小间隔周期数。
—	其它	—	保留。

2.2.8 CfgTR6

偏移地址：0x00300

CfgTR6为时序寄存器6，可读写，该寄存器用于控制片上端接（ODT）时序。

表 2-29: CfgTR6 寄存器域的描述

名称	范围	类型	描述
CfgAOFDPD	[9:7]	RW,1	异步ODT关闭延时（周期数）。未实现异步ODT；
CfgAONPD	[6:4]	RW,1	异步ODT打开延时（周期数）。未实现异步ODT；
CfgODTH8	[3:0]	RW,6	ODT为高的时间（周期数）。
—	其它	—	保留。

2.2.9 CfgTR7

偏移地址：0x00380

CfgTR7为时序寄存器，可读写，该寄存器用于控制其它时序。

表 2-30: CfgTR7 寄存器域的描述

名称	范围	类型	描述
CfgZQI	[49:28]	RW, 2500000	正常运行期间，ZQCL/S命令的产生间隔周期数。（ZQI计数器使用维护时钟进行计数）
CfgRSTH	[27:13]	RW, 15000	初始化阶段复位信号为高电平的时间。（RSTH计数器使用维护时钟进行计数）
CfgRSTL	[12:0]	RW, 6000	初始化阶段复位信号为低电平的时间。（RSTL计数器使用维护时钟进行计数）
—	其它	—	保留。

2.2.10 CfgDec

偏移地址：0x00400

CfgDec 记录了当前存控中部分配置的摘要内容。

表 2-31: CfgDec 寄存器域的描述

名称	范围	类型	描述
IsMPR	[57]	RO,0	为“1”时，指示MPR处于打开状态。
IsDynODTOn	[56]	RO,0	为“1”时，指示动态ODT处于打开状态。
IsRttWRRZQD4	[55]	RO,0	RttWR配置。
IsRttWRRZQD2	[54]	RO,0	RttWR配置。
IsSRTNormal	[53]	RO,1	SRT为normal模式。
IsASREnable	[52]	RO,0	ASR使能。
IsCWL10	[51]	RO,0	为“1”时，指示CWL配置为“10”。
IsCWL9	[50]	RO,0	为“1”时，指示CWL配置为“9”。
IsCWL8	[49]	RO,0	为“1”时，指示CWL配置为“8”。
IsCWL7	[48]	RO,0	为“1”时，指示CWL配置为“7”。
IsCWL6	[47]	RO,1	为“1”时，指示CWL配置为“6”。
IsCWL5	[46]	RO,0	为“1”时，指示CWL配置为“5”。
IsFullArray	[45]	RO,1	全阵列自刷新。

IsHalfArrayL	[44]	RO,0	低段半阵列自刷新。
IsQuarArrayL	[43]	RO,0	低段四分之一阵列自刷新。
Is1by8ArrayL	[42]	RO,0	低段八分之一阵列自刷新。
Is3by4Crray	[41]	RO,0	四分之三阵列自刷新。
IsHalfArrayH	[40]	RO,0	高段半阵列自刷新。
IsQuarArrayH	[39]	RO,0	高段四分之一阵列自刷新。
Is1by8ArrayH	[38]	RO,0	高段八分之一阵列自刷新。
IsOutEnable	[37]	RO,1	输出使能。
IsTDQSEnable	[36]	RO,0	TDQS使能。
IsSynODTOn	[35]	RO,1	同步ODT打开。
IsRttNOMRZQD4	[34]	RO,1	RttNOMR配置为1/4。
IsRttNOMRZQD2	[33]	RO,0	RttNOMR配置为1/2。
IsRttNOMRZQD6	[32]	RO,0	RttNOMR配置为1/6。
IsRttNOMRZQD12	[31]	RO,0	RttNOMR配置为1/12。
IsRttNOMRZQD8	[30]	RO,0	RttNOMR配置为1/8。
IsWLEnable	[29]	RO,0	Write Leveling使能。
IsOutputRZQD6	[28]	RO,1	输出电阻选择为1/6。
IsOutputRZQD7	[27]	RO,0	输出电阻选择为1/7。
IsAL0	[26]	RO,1	为“1”时，指示AL配置为“0”。
IsALCL_1	[25]	RO,0	为“1”时，指示AL配置为“CL-1”。
IsALCL_2	[24]	RO,0	为”1”时，指示AL配置为“CL-2”。
IsDLLEnable	[23]	RO,1	DLL使能。
IsFastExit	[22]	RO,1	快速退出Precharge PowerDown。
IsWR5	[21]	RO,0	为“1”时，指示WR配置为“5”。
IsWR6	[20]	RO,0	为“1”时，指示WR配置为“6”。
IsWR7	[19]	RO,0	为“1”时，指示WR配置为“7”。
IsWR8	[18]	RO,0	为“1”时，指示WR配置为“8”。
IsWR10	[17]	RO,0	为“1”时，指示WR配置为“10”。
IsWR12	[16]	RO,1	为“1”时，指示WR配置为“12”。
IsDLLReset	[15]	RO,1	DLL复位。
IsTestMode	[14]	RO,0	测试模式。
IsCL14	[13]	RO,0	为“1”时，指示CL配置为“14”。
IsCL13	[12]	RO,0	为“1”时，指示CL配置为“13”。

IsCL12	[11]	RO,0	为“1”时，指示CL配置为“12”。
IsCL11	[10]	RO,0	为“1”时，指示CL配置为“11”。
IsCL10	[9]	RO,0	为“1”时，指示CL配置为“10”。
IsCL9	[8]	RO,0	为“1”时，指示CL配置为“9”。
IsCL8	[7]	RO,0	为“1”时，指示CL配置为“8”。
IsCL7	[6]	RO,1	为“1”时，指示CL配置为“7”。
IsCL6	[5]	RO,0	为“1”时，指示CL配置为“6”。
IsCL5	[4]	RO,0	为“1”时，指示CL配置为“5”。
IsInterleave	[3]	RO,1	为“1”时，指示突发类型为交错。
IsBL8	[2]	RO,1	为“1”时，指示突发长度为“8”。
IsBL8orBC4	[1]	RO,0	为“1”时，指示突发长度为“8”或者“4”。
IsBC4	[0]	RO,0	为“1”时，指示突发长度为“4”。
—	其它	—	保留。

2.2.11 CfgCR

偏移地址：0x00480

CfgCR为控制寄存器。该寄存器必须先于其它存储控制器的IO寄存器进行配置。

表 2-32: CfgCR 寄存器域的描述

名称	范围	类型	描述
MCUPDProcTime	[62:57]	RW,24	配置MC发起的Update过程从Update Req有效到PUB的Ack信号撤销的最大等待时间。最大配置63拍。
MCUPDWaitTime	[56:51]	RW,12	配置MC发起的Update过程从仲裁上ZQCL/ZQCS命令到发起Update Req的间隔。最大配置63拍。
UpdatePUBAck	[50]	RW,0	对于MC发起的Update请求，PUB是否给出Ack信号的指示。如果该位为“1”，则表示PUB会给出Ack信号。
MCUpdateEn	[49]	RW,0	MC发起的Update请求使能。该位为“1”，则在仲裁上ZQCL/ZQCS命令时，MC发起Update请求。
UPDResCntEn	[48]	RW,0	Update 状态机从 UpdST_WAIT 跳转到 UpdST_UPD时，是否判断发出去的MEM读都已返回响应使能，为0，表示不对此判断，为1，表

			示需要对此判断。
RankSelCnt	[47:42]	RW,7	Rank之间的命令仲裁按固定时隙优先选择，该域定义固定时隙的大小。
UPDWaitTime	[41:36]	RW,12	配置Update Req到Update Ack的间隔。最大配置63拍。
AtomUPD	[35]	RW,1	该位为“1”，表示Udpate处理遵守与其他原子序列的关系。
UPDForbiddenAll	[34]	RW,1	该位为“1”，表示Update请求需要禁止所有SDRAM命令，否则只禁止读写命令。
UpdateEn	[33]	RW,0	对于PHY发起的Update请求，存控的确认使能。如果该位为“1”，则予以确认。
gr_EnableWrap	[32]	RW,0	存控进行数据训练时起作用，该位为“1”则使能Wrap地址，否则屏蔽Wrap地址（清零）。
ForceResponce	[31]	RW,0	DQS Gate Training的超时强制响应，主要用于调试（存控进行数据训练的子步骤DQS Gate训练）。
RDIMM	[30]	WR,0	RDIMM 指示。
UnDefine	[29]	RW,0	该位保留。 目前设计该位可读写，但无意义。
DisableREF	[28]	RW,0	关闭刷新功能。
2T	[27]	RW,1	2T模式使能。
AddressMirror	[26]	RW,0	地址镜像使能。 在双Rank UDIMM时需要根据实际情况开启地址镜像。 对于TwinDie封装的颗粒（相当于两个Rank，这个两个Rank不需要考虑镜像的问题），如果使用该颗粒组成双面的DIMM条目，双面之间可能需要镜像，此时认为0/1、2/3分别处于DIMM条的不同面。 双Rank时：对Rank1进行镜像； 四Rank时：对Rank2/3、Rank1/3进行镜像，由AddressMirrorType进行控制。
DTUWrSEL	[25:23]	RW,0	存控进行数据训练时，用于选择第（0~7）个Beat写数据。
DTURdSEL	[22:20]	RW,0	存控进行数据训练时，用于选择第（0~7）个Reat

			读数据。
AddressMirrorType	[19]	RW,0	表示在四 Rank 且进行地址镜像时，为1对 Rank2/3、为0对Rank1/3进行镜像。
ForbiddenReceive	[16]	RW,1	禁止将DDR3物理通路接收到的读响应数据转发给一致性处理部件。该功能用于消除DDR3接口初始化过程中可能出现的干扰。 正常访问之前必须把该位清零。
ForbiddenArbWR	[15]	RW,0	禁止仲裁WR命令。用于DEBUG。
UnDefine	[11]	RW,0	该位保留。 该位可读写，但无意义。
ForbiddenArbRD	[10]	RW,0	禁止仲裁RD命令。用于DEBUG。
DisablePD	[9]	RW,0	关闭PD功能。
DisableSR	[8]	RW,0	关闭SR功能。
DisableZQ	[7]	RW,0	关闭ZQ功能。
DisableECC	[6]	RW,0	关闭ECC校验。 在64位数据通路(不带ECC时),该位必须为“1”。
ForbiddenArbACT	[5]	RW,0	禁止仲裁ACT命令。用于DEBUG。
ForbiddenArbPRE	[4]	RW,0	禁止仲裁PRE命令。用于DEBUG。
FastReq	[3]	RW,0	快速产生请求。 该功能仅用于验证，用于缩写刷新间隔、缩短进入Self-Refresh、Power-Down的等待时间。芯片正常使用时，该位必须配置为0。
DFTBypassDTU	[2]	RW,1	数据训练旁路使能。 该信号只在“存控进行数据训练状态”有意义。 如果需要存控进行数据训练，则需要控制状态机进入“存控进行数据训练状态”，并且在此前把该位改为“0”。
DFTZQCLSEL	[1]	RW,0	选择ZQCL命令。 0: ZQCS; 1: ZQCL。 该位只影响定期ZQ命令类型。在存储控制器进行SDRAM初始化过程中按流程使用ZQCL命令。

DFTFastInit	[0]	RW,1	快速初始化(在存储控制器进行SDRAM初始化过程中, 控制Sdram_Reset在High或Low等待的时间)。仅仅用于测试。正常使用时, 该位必须为“0”。
—	其它	—	保留。

注: 该寄存器相对于申威 411 处理器有修改。

2.2.12 ODT_CTRL

偏移地址: 0x00900

ODT_CTRL 为存储控制器内部控制寄存器, 对 ODT 进行控制。

表 2-33: ODT_CTRL 寄存器域的描述

名称	范围	类型	描述
WRODT3	[31:28]	RW,0x8	对 Rank3 进行写操作时, Rank3~Rank0 的 ODT 控制使能。高位对应 Rank3, 低位对应 Rank0。
RDODT3	[27:24]	RW,0x0	对 Rank3 进行读操作时, Rank3~Rank0 的 ODT 控制使能。高位对应 Rank3, 低位对应 Rank0。
WRODT2	[23:20]	RW,0x4	对 Rank2 进行写操作时, Rank3~Rank0 的 ODT 控制使能。高位对应 Rank3, 低位对应 Rank0。
RDODT2	[19:16]	RW,0x0	对 Rank2 进行读操作时, Rank3~Rank0 的 ODT 控制使能。高位对应 Rank3, 低位对应 Rank0。
WRODT1	[15:12]	RW,0x2	对 Rank1 进行写操作时, Rank3~Rank0 的 ODT 控制使能。高位对应 Rank3, 低位对应 Rank0。
RDODT1	[11:8]	RW,0x0	对 Rank1 进行读操作时, Rank3~Rank0 的 ODT 控制使能。高位对应 Rank3, 低位对应 Rank0。
WRODT0	[7:4]	RW,0x1	对 Rank0 进行写操作时, Rank3~Rank0 的 ODT 控制使能。高位对应 Rank3, 低位对应 Rank0。
RDODT0	[3:0]	RW,0x0	对 Rank0 进行读操作时, Rank3~Rank0 的 ODT 控制使能。高位对应 Rank3, 低位对应 Rank0。
—	其它	—	保留。

注: 该寄存器相对于申威 411 处理器有修改。

2.2.13 MC_CTRL

偏移地址：0x00500

MC_CTRL 为存储控制器内部控制寄存器，该寄存器中包含存储控制器的 ODT 控制、主存配置和错误检查使能信息。

表 2-34：MC_CTRL 寄存器域的描述

名称	范围	类型	描述		
DRankLocation	[31]	WR,0	表示双 Rank 时内存条位置。 0: 双 Rank 位于一个插槽； 1: 双 Rank 位于二个插槽。		
MEMWIDTH	[30:28]	WR,0	SDRAM 接口数据宽度，其中： 3'h0: 72bit，即 64bit 数据+8bitECC； 3'h1: 64bit 数据，不带 ECC； 3'h2: 32bit 数据，可以配置支持 ECC； 3'h3: 16bit 数据，不带 ECC； 其它保留		
BANKHASH	[27]	WR,0	BANK 地址 hash 选择。 首先根据 MAPCFG 选择地址映射模式，在根据该位决定是否对地址进行 Hash。		
BankSEL	[26:24]	WR,0	地址映射模式选择。 只有0、4、6三种有意义，其它值默认为0。		
			Bank 配置	Bank 地址	备注
			0 (默认)	9-7	按 Cache 块交叉
			4 (8KB)	15-13	按 8KB 页面交叉
6 (1MB)	22-20	按 1MB 数据块交叉			

MEMCAP	[23:21]	RO,4	<p>表示该存储器接口的主存容量，具体如下：</p> <p>0: 256MB（只在 16 位数据宽度时使用）</p> <p>1: 512MB（只在 16/32 位数据宽度时使用）</p> <p>2: 1GB</p> <p>3: 2GB</p> <p>4: 4GB</p> <p>5: 8GB</p> <p>6: 16GB (申威 421 处理器新增)</p> <p>7: 32GB (申威 421 处理器新增)</p> <p>注1：硬件根据颗粒容量、颗粒个数（即存储器接口位宽）和Rank数目自动确定芯片总容量。</p> <p>注2：上述容量按72位（含ECC）/64位数据位宽计算，0/1编码在72位（含ECC）/64位时保留。</p> <p>注3：软件只需根据SPD信息填写本寄存器的其它域，本域由硬件计算得到。需要保证MCU IOR：MC_CONFIG[MC0CAP]与该配置一致。</p>
DVTYPE	[20]	WR,0	<p>颗粒类型：</p> <p>“0”：x8颗粒；</p> <p>“1”：x16颗粒。</p> <p>不支持x4颗粒，对于x4颗粒每4比特数据对应一个DQS。</p>
DVPAGE	[19:18]	WR,0	<p>表示内存颗粒页面大小。</p> <p>0: 1KB</p> <p>1: 2KB</p> <p>其它：保留</p> <p>页面大小 KB: $(2^{**}ColBits*ORG/8)$，其中 ColBits 为 Col 地址位宽，ORG 为颗粒类型（x8 或 x16）。</p>

DVCAP	[17:15]	WR,2	表示颗粒容量。 0: 1Gb 1: 2Gb 2: 4Gb 3: 8Gb 其它保留 颗粒容量 KB: (2**RowBits*DVPAGE*Bank), 其中 RowBits 为 Row 地址位宽, DVPAGE 为页面大小, Bank 为 Bank 数量; 根据 DDR3 标准, 不支持 8Gb 以上的颗粒容量。对于 TwinDie 的芯片颗粒是通过 Rank 扩展实现 (在一个颗粒中集成两个 Die, 每个 Die 对应一个 Rank)。																																																				
RANKTYPE	[14:13]	WR,0	表示 Rank 类型。 2'b00: 单 Rank; 2'b01: 双 Rank; 2'b1x: 四 Rank;																																																				
RankSEL	[12]	WR,0	0: Rank 按存控的主存容量进行顺序编址。 <table border="1" data-bbox="740 1106 1342 1653"> <thead> <tr> <th>主存容量</th> <th>1R</th> <th>2R</th> <th>4R</th> </tr> </thead> <tbody> <tr> <td>256MB</td> <td>2'b0</td> <td>1'b0, MA[27]</td> <td>MA[27:26]</td> </tr> <tr> <td>512MB</td> <td>2'b0</td> <td>1'b0, MA[28]</td> <td>MA[28:27]</td> </tr> <tr> <td>1GB</td> <td>2'b0</td> <td>1'b0, MA[29]</td> <td>MA[29:28]</td> </tr> <tr> <td>2GB</td> <td>2'b0</td> <td>1'b0, MA[30]</td> <td>MA[30:29]</td> </tr> <tr> <td>4GB</td> <td>2'b0</td> <td>1'b0, MA[31]</td> <td>MA[31:30]</td> </tr> <tr> <td>8GB</td> <td>2'b0</td> <td>1'b0, MA[32]</td> <td>MA[32:31]</td> </tr> <tr> <td>16GB</td> <td>2'b0</td> <td>1'b0, MA[33]</td> <td>MA[33:32]</td> </tr> <tr> <td>32GB</td> <td>2'b0</td> <td>1'b0, MA[34]</td> <td>MA[34:33]</td> </tr> </tbody> </table> 1: Rank 按低位地址进行交叉编址。与主存容量无关, 与 Bank 地址的配置相关 <table border="1" data-bbox="740 1760 1342 1973"> <thead> <tr> <th>Bank 配置</th> <th>1R</th> <th>2 Rank</th> <th>4 Rank</th> </tr> </thead> <tbody> <tr> <td>0 (默认)</td> <td>2'b0</td> <td>1'b0, MA[10]</td> <td>MA[11:10]</td> </tr> <tr> <td>4 (8KB)</td> <td>2'b0</td> <td>1'b0, MA[16]</td> <td>MA[17:16]</td> </tr> <tr> <td>6 (1MB)</td> <td>2'b0</td> <td>1'b0, MA[23]</td> <td>MA[24:23]</td> </tr> </tbody> </table>	主存容量	1R	2R	4R	256MB	2'b0	1'b0, MA[27]	MA[27:26]	512MB	2'b0	1'b0, MA[28]	MA[28:27]	1GB	2'b0	1'b0, MA[29]	MA[29:28]	2GB	2'b0	1'b0, MA[30]	MA[30:29]	4GB	2'b0	1'b0, MA[31]	MA[31:30]	8GB	2'b0	1'b0, MA[32]	MA[32:31]	16GB	2'b0	1'b0, MA[33]	MA[33:32]	32GB	2'b0	1'b0, MA[34]	MA[34:33]	Bank 配置	1R	2 Rank	4 Rank	0 (默认)	2'b0	1'b0, MA[10]	MA[11:10]	4 (8KB)	2'b0	1'b0, MA[16]	MA[17:16]	6 (1MB)	2'b0	1'b0, MA[23]	MA[24:23]
主存容量	1R	2R	4R																																																				
256MB	2'b0	1'b0, MA[27]	MA[27:26]																																																				
512MB	2'b0	1'b0, MA[28]	MA[28:27]																																																				
1GB	2'b0	1'b0, MA[29]	MA[29:28]																																																				
2GB	2'b0	1'b0, MA[30]	MA[30:29]																																																				
4GB	2'b0	1'b0, MA[31]	MA[31:30]																																																				
8GB	2'b0	1'b0, MA[32]	MA[32:31]																																																				
16GB	2'b0	1'b0, MA[33]	MA[33:32]																																																				
32GB	2'b0	1'b0, MA[34]	MA[34:33]																																																				
Bank 配置	1R	2 Rank	4 Rank																																																				
0 (默认)	2'b0	1'b0, MA[10]	MA[11:10]																																																				
4 (8KB)	2'b0	1'b0, MA[16]	MA[17:16]																																																				
6 (1MB)	2'b0	1'b0, MA[23]	MA[24:23]																																																				

ArbCFG	[9:4]	WR,0	读写请求仲裁控制。 [5:4]表示仲裁类型：2'b00 表示选择与上一个请求相同属性；2'b01 表示读优先；2'b10 表示轮转；2'b11 表示按固定时隙； [3]表示按固定时隙，读写时隙比例，该位为“1”表示读写比例是 2:1，否则读写比例是 1:1； [2:0]表示按固定时隙时的时间单位：2*(Value+1)；表示固定时隙的时间单位时，有效配置是0x0~0x5。
WRDATMERR_EN	[3]	WR,0	写数据多错检查使能，该位为“1”时，如果写主存数据发现不可纠正多错，则产生WRDATMERR。
WRDATMSERR_EN	[2]	WR,0	写数据可纠正错记录使能，该位为“1”时，如果写主存数据发现可纠正错，则产生WRDATMSERR。
MEMMERR_EN	[1]	WR,0	主存多错检查使能，该位为“1”时，如果读主存时发现不可纠正多错，则产生MEMMERR。
MEMSERR_EN	[0]	WR,0	主存可纠正错记录使能，该位为“1”时，如果读主存时发现可纠正错，则产生MEMSERR。
—	其它	—	保留。

注：该寄存器相对于申威 411 处理器有修改。

2.2.14 INIT_CTRL

偏移地址：0x00580

INIT_CTRL 是存控初始化流程控制寄存器，可读写。

表 2-35：INIT_CTRL 寄存器域的描述

名称	范围	类型	描述
DATATRAN_MC	[3]	RW,0	该位为“1”表示由用户软件控制存储器控制进行Data Training。该位与WAITPUB_DONE结合使用，一起自清除。
WAITPUB_DONE	[2]	RW,0	表示通过触发PUB已经完成SDRAM初始化和Data Training。用户配置该寄存器进入下一个状态；该位会自清除。 当该存控对应的所有的PHY都完成SDRAM初始化和数据训练后，配置该位进行跳转。
SDRAMINIT_MC	[1]	RW,0	该位为“1”表示SDRAM初始化由用户软件控制存储

			控制器进行，否则由用户触发PUB自动完成。推荐使用触发PUB自动完成的方式。 该位与PHYINIT_DONE结合使用，一起自清除。
PHYINIT_DONE	[0]	RW,0	表示PHY已经自动完成初始化。用户配置该寄存器进入下一个状态；该位会自清除。 当该存控对应的所有的PHY都完成初始化后，配置该位进行跳转。
—	其它	—	保留。

2.2.15 ClosePage

偏移地址：0x00600

ClosePage 是页面管理策略配置寄存器，可读写。默认是 OpenPage 策略。

表 2-36: ClosePage 寄存器域的描述

名称	范围	类型	描述
ClosePage	[0]	RW,0	该位为“1”表示采用ClosePage策略，否则采用OpenPage策略。
—	其它	—	保留。

2.2.16 DTUR

偏移地址：0x00680

DTUR 为数据通路信号调节（Data Training）读命令寄存器，进行数据通路信号调节时，对于该寄存器的写将触发一次存储器读。通过该寄存器可以配置读地址、Bank 和 Rank，通过读 DTURD0~DTURD2 获得读数据。

表 2-37: DTUR 寄存器域的描述

名称	范围	类型	描述
DTURdCAAddr	[32:22]	RW,0	数据通路信号调节时，读请求的列地址。
DTURdRAAddr	[21:6]	RW,0	数据通路信号调节时，读请求的行地址。
DTURdBank	[5:3]	RW,0	数据通路信号调节时，读请求的Bank号。
DTURdRank	[2:1]	RW,0	数据通路信号调节时，读请求的Rank号。
DTURdValid	[0]	RW,0	数据通路信号调节读请求有效。该位具有自清除功能。

—	其它	—	保留。
---	----	---	-----

注：该寄存器相对于申威 411 处理器有修改。

2.2.17DTUW

偏移地址：0x00700

DTUW 为数据通路信号调节（Data Training）写命令寄存器，进行数据通路信号调节时，对于该寄存器的写将触发一次存储器写。通过该寄存器可以配置写地址、Bank 和 Rank，通过 DTUWD0~DTUWD2 指定写数据。

表 2-38：DTUW 寄存器域的描述

名称	范围	类型	描述
DTUWrCAAddr	[32:22]	RW,0	数据通路信号调节时，写请求的列地址。
DTUWrRAddr	[21:6]	RW,0	数据通路信号调节时，写请求的行地址。
DTUWrBank	[5:3]	RW,0	数据通路信号调节时，写请求的Bank号。
DTUWrRank	[2:1]	RW,0	数据通路信号调节时，写请求的Rank号。
DTUWrValid	[0]	RW,0	数据通路信号调节写请求有效。该位可自清除。
—	其它	—	保留。

注：该寄存器相对于申威 411 处理器有修改。

2.2.18DTUWD0

偏移地址：0x00780

DTUWD0 为数据通路信号调节（Data Training）写数据寄存器 0。

表 2-39：DTUWD0 寄存器域的描述

名称	范围	类型	描述
DTUWD0	[7:0]	RW,0	数据通路信号调节写数据对应的ECC。
—	其它	—	保留。

2.2.19DTUWD1

偏移地址：0x00800

DTUWD1 为数据通路信号调节（Data Training）写数据寄存器 1。

表 2-40：DTUWD1 寄存器域的描述

名称	范围	类型	描述
DTUWD1	[63:0]	RW,0	数据通路信号调节写数据。

2.2.20 DTURD0

偏移地址：0x00980

DTURD0 为数据通路信号调节（Data Training）读数据寄存器 0。

表 2-41：DTURD0 寄存器域的描述

名称	范围	类型	描述
DTURD0	[7:0]	RW	数据通路信号调节读数据对应的ECC。
—	其它	—	保留。

2.2.21 DTURD1

偏移地址：0x00a00

DTURD1 为数据通路信号调节（Data Training）读数据寄存器 1。

表 2-42：DTURD1 寄存器域的描述

名称	范围	类型	描述
DTURD1	[63:0]	RW	数据通路信号调节读数据。

2.2.22 DTUCTL

偏移地址：0x00b00

DTUCTL 为数据通路信号调节（Data Training）测试结束控制，用于初始化流程。

表 2-43：DTUCTL 寄存器域的描述

名称	范围	类型	描述
DTUEnd	[0]	RW,0	数据通路信号调节测试完成。对该位写1，可以强制结束数据通路信号调节。具有自清除功能。
—	其它	—	保留。

2.2.23 DTUPREA

偏移地址：0x00b80

DTUPREA 为数据通路信号调节（Data Training）PREA 命令寄存器，进行数据通路信号调节测试时，对于该寄存器的写将触发一次 PREA 命令。通过该寄存器可以配置 PREA 命令的目标 Rank。

表 2-44：DTUPREA 寄存器域的描述

名称	范围	类型	描述
DTUPREARank	[2:1]	RW,0	数据通路信号调节期间产生 PREA 命令对应的 Rank。
DTUPREAVValid	[0]	RW,0	数据通路信号调节期间产生 PREA 命令。对该位写 1，可以产生一条 PREA 命令。具有自清除功能。
—	其它	—	保留。

注：该寄存器相对于申威 411 处理器有修改。

2.2.24 WL_CTRL

偏移地址：0x00c00

WL_CTRL 寄存器用于非全流程的 Write Leveling 控制，可读写；

表 2-45：WL_CTRL 寄存器域的描述

名称	范围	类型	描述
WLRankValid	[3]	RW,0	该位为“1”，表示 MRS 命令和 ODT 都是对指定 Rank 的控制。当 IOR: CfgMR1[WL] 为“1”时，该位有意义。
WLRank	[2:1]	RW,0	当 IOR: CfgMR1[WL] 为“1”，且 WLRankValid 为“1”时该位有意义。该位表示针对的 Rank。注：此时存储控制器发出的 MRS 命令只发向该位指定的 Rank。
WLODTON	[0]	RW,0	当 IOR: CfgMR1[WL] 为“1”，且 WLRankValid 为“1”时，该位有意义，该位为“0”，表示打开 ODT，否则表示关闭 ODT。注：此时存储控制器发出的 ODT 只发向 WLRank 指定的 Rank。
—	其它	—	保留。

注：该寄存器相对于申威 411 处理器有修改。

2.2.25 MemSERR_P

偏移地址：0x00c80

MEMSERR_P 用于记录已纠正校验错的相关信息。复位时为全“0”。

表 2-46：MEMSERR_P 寄存器域的描述

名称	范围	类型	描述
Valid	[8]	RW1C,0	发生主存数据单错错标志。
ECC	[7:0]	RO,0	发生主存数据单错错时的诊断码。
—	其它	—	保留。

当存储器访问发生主存数据单错（MEMSERR），且 IOR: MEMSERR_P[Valid]为“0”时，置 Valid 位为“1”，并在 IOR: MEMSERR_P 中登记发生可纠错的 64 位数据的 16 位诊断码，用于判断单错发生的位置，该诊断码只记录第 1 个已纠正校验错的信息。

2.2.26 CfgErr

偏移地址：0x00d00

CfgErr 用于记录当前配置合法性检查结果。

表 2-47：CfgErr 寄存器域的描述

名称	范围	类型	描述
ErrorASRSRT	[10]	RO,0	ASR和SRT同时有效。
ErrorMPR	[9]	RO,0	MPR配置错。 正常操作模式必须Disable MPR。
ErrorWLeveling	[8]	RO,0	Write Leveling配置错。 正常操作模式必须Disable Write Leveling。 Write Leveling必须关闭动态ODT。
ErrorDLLODT	[7]	RO,0	DLL与ODT组合配置错。 DLL Disable是必须关闭ODT。
ErrorRBT	[5]	RO,0	MRS配置错。 突发类型不能配置为顺序模式。
ErrorBL8	[4]	RO,0	突发长度不是“8”。
ErrorMBZ	[3]	RO,0	必须为“0”的配置实际没有配“0”。 MR0[7]：测试模式必须为零； MR1[8]和MR1[10]必须为零；

			MR1[11]必须为零； MR1[12]在非Write Leveling时必须为零； MR2[8]和MR2[2:0]必须为零； MR3[1:0]必须为零。
ErrorRsv	[2]	RO,0	保留位配置错。 CfgMR0~CfgMR3的配置信息使用了保留值。
ErrorCfg	[1]	RO,0	模式寄存器配置错（总的出错标志）。
ErrorCfgFPGA	[0]	RO,1	FPGA配置错。 RL配置不为6，或者WL配置不为6，或者DLL使能则报FPGA配置错；
—	其它	—	保留。

2.2.27 FSMSTAT

偏移地址：0x00e00

FSMSTAT 用于记录存储控制器内部的状态机状态。

表 2-48：FSMSTAT 寄存器域的描述

名称	范围	类型	描述
SMSDRAM	[7:3]	RO,0	SDRAM初始化状态机（只有在用户软件控制存储控制器进行SDRAM进行初始化时有意义）。 0: 空闲状态； 1: Reset_N为低； 2: Reset_N为高； 3: 等待XPR满足； 4/11/16/17: 发送MR2命令（Rank0~3）； 5/12/18/19: 发送MR3命令（Rank0~3）； 6/13/20/21: 发送MR1命令（Rank0~3）； 7/14/22/23: 发送MR0命令（Rank0~3）； 8/15/24/25: 发送ZQ命令（Rank0~3）； 9: DTU测试； 10: 初始化结束状态。
SMINIT	[2:0]	RO,0	存控初始化状态机。 0: PHY初始化状态；

			1: 存控进行SDRAM初始化状态; 2: PUB进行SDRAM初始化状态; 3: 存控进行Data Training状态; 4: 正常访问状态; 其它: 保留。
—	其它	—	保留。

注：该寄存器相对于申威 411 处理器有修改。

2.2.28 CfgTest

偏移地址：0x01100

CfgTest 为存控接口备份寄存器，该寄存器不控制任何值，可以对该寄存器进行读写，便于测试 IO 通路。

表 2-49: CfgTest 寄存器域的描述

名称	范围	类型	描述
CfgTest	[3:0]	RW,0	数据。
—	其它	—	保留。

2.2.29 RC

偏移地址：0x00e80

RC 为 RDIMM 配置寄存器。外接 RDIMM 存储条时，需要写该寄存器触发 WRC 命令，对 RC 进行配置。

表 2-50: RC 寄存器域的描述

名称	范围	类型	描述
ReqWRCValid	[8]	RW,0	RC配置请求的有效位。
ReqWRCAddr	[7:4]	RW,0	RC配置地址。
ReqWRCDData	[3:0]	RW,0	RC配置数据。
—	其它	—	保留。

2.2.30 RCMRD

偏移地址：0x00f00

RCMRD 用于配置 RDIMM 的扩展模式寄存器加载等待时间。外接 RDIMM 存储条时，使用该寄存器配置 MRD 计数初值。

表 2-51: RCMRD 寄存器域的描述

名称	范围	类型	描述
CfgRCMRD	[4:0]	RW,8	RC配置时的tMRD。
—	其它	—	保留。

2.2.316U

偏移地址: 0x00f80

6U 用于配置 RDIMM 的 RC 寄存器加载等待时间。外接 RDIMM 存储条时，使用该寄存器设置 RC 配置延时。

表 2-52: 6U 寄存器域的描述

名称	范围	类型	描述
Cfg6U	[12:0]	RW,4800	RC配置时的6us延迟。
—	其它	—	保留。

2.2.32POWERCTRL

偏移地址: 0x01080

POWERCTRL为存储器低功耗控制寄存器，通过设置该寄存器可以开启和关闭DDR3存储器的浅睡眠和深睡眠功能，以实现低功耗控制。

表 2-53: POWERCTRL 寄存器域的描述

名称	范围	类型	描述
DSLEEP	[3:2]	RW,0	存储器进入深睡眠状态的时间阈值。该计数器使用维护时钟进行计数，该时钟频率为25MHz时。 0: 关闭; 1: 1ms; 2: 2ms; 3: 3ms。 该时间与维护时钟频率相关。
SSLEEP	[1:0]	RW,0	存储器进入浅睡眠状态的时间阈值。该计数器使

			用维护时钟进行计数，该时钟频率为25MHz时。 0: 关闭; 1: 1us; 2: 2us; 3: 3us。 该时间与维护时钟频率相关。
—	其它	—	保留。

2.2.33 PUB_INTERFACE

偏移地址：0x01000

PUB_INTETFACE 为地址/命令/数据接口信号控制寄存器，可读写。

表 2-54: PUB_INTETFACE 寄存器域的描述

名称	范围	类型	描述
PUB_ctrlupd_ack_error	[32:31]	RW1 C,0	PUB输出的ctrlupd_ack信号错误指示。 [31]:MC在未发起update请求时，ctrlupd_ack信号有效; [32]:MC发起的update请求，在等待MCUPDProcTime时间后，ctrlupd_ack信号仍未撤销;
PUB_rdlvl_gate_mode	[30:29]	RO,0	Pub引脚
PUB_rdlvl_mode	[28:27]	RO,0	Pub引脚
PUB_wrlvl_mode	[26:25]	RO,0	Pub引脚
PUB_parity_error	[24]	RW1 C,0	RDIMM奇偶校验错指示。
PUB_init_complete	[23]	RO,0	Pub引脚
PUB_qcsen_n_di	[22]	RO,0	Pub引脚
PUB_mirror_di	[21]	RO,0	Pub引脚
PUB_ram_rst_n_di	[20]	RO,0	Pub引脚
PUB_dto_di	[19:18]	RO,0	Pub引脚
PUB_byp_mode	[17]	RO,0	Pub引脚，表示时钟旁路。
PUB_err_out_n_dout	[16]	RW,0	只有在使用err_out_n引脚作为输出引脚时有意义。

PUB_dram_clk_disable	[15:13]	RW,0	关闭时钟。每位对应一个CK引脚，共3位。
PUB_data_byte_disalbe	[12:4]	RW,0	关闭Datx8。每位对应一个Datx8，共9位。
PUB_init_start	[3]	RW,0	接口上产生dfi_inti_start信号。
PUB_ret_en_i	[2]	RW,0	配置ret_en_i。
PUB_ret_en	[1]	RW,0	配置ret_en_i。
PUB_poc_dis	[0]	RW,0	IO的Power ON Clear Disable。
—	其它	—	保留。

注：该寄存器相对于申威 411 处理器有修改。

2.2.34 CfgMR0

偏移地址：0x01280

CfgMR0 为 DDR3 模式寄存器 0；写该寄存器会触发存储控制器向存储器发一条 MRS 命令，配置 DDR3 存储器颗粒的 MR0 寄存器。

表 2-55：CfgMR0 寄存器域的描述

名称	范围	类型	描述
PPD	[12]	RW,1	进入带Precharge的关电状态时，DLL控制方式： “0”：慢速退出（DLL off）； “1”：快速退出（DLL On）。
WR	[11:9]	RW,6	自动Precharge的时候，Write Recovery时间，配置值与周期数的对应关系如下： “0”：“16”；（DDR3标准79-3D版本新增） “1”：“5”； “2”：“6”； “3”：“7”； “4”：“8”； “5”：“10”； “6”：“12”； “7”：“14”。（DDR3标准79-3D版本新增）
DLL	[8]	RW,1	DLL复位，该位为“1”表示DLL复位。
TM	[7]	RW,0	测试模式，该位为“1”表示测试模式。
CL	[6:4],[2]	RW,10	CAS 延迟。配置值与周期数的对应关系如下： “0”：保留； “2”：“5”；

			“4” : “6”; “6” : “7”; “8” : “8”; “10” : “9”; “12” : “10”; “14” : “11”(DDR3-1610的可选配置); “1” : “12” ; “3” : “13” ; “5” : “14” 。 “7” : 为后续15保留; “9” : 为后续16保留;
RBT	[3]	RW,1	突发类型。 “0” : 顺序模式; “1” : 缠绕描述。
BL	[1:0]	RW,0	突发长度, 译码为“8”。 00: 固定为8; 01: BC4或8; 10: 固定为4; 11: 保留
—	其它	—	保留。

2.2.35 CfgMR1

偏移地址: 0x01300

CfgMR1 为 DDR3 模式寄存器 1; 写该寄存器会触发存储控制器向存储器发一条 MRS 命令, 配置 DDR3 存储器颗粒的 MR1 寄存器。

表 2-56: CfgMR1 寄存器域的描述

名称	范围	类型	描述
Qoff	[12]	RW,0	该位为“1”表示输出缓冲不使能。
TDQS	[11]	RW,0	TDQS使能。
MBZ	[10],[8]	RW,0	必需配置为“0”。
Level	[7]	RW,0	Write Leveling使能。
AL	[4:3]	RW,0	附加延时:

			“0”：AL=0; “1”：AL=CL-1; “2”：AL=CL-2; “3”：保留。
Rtt_Nom	[9],[6],[2]	RW,1	Rtt_Nom阻值选择，其中RZQ为240欧姆： “0”：关闭Rtt_Nom; “1”：RZQ/4; “2”：RZQ/2; “3”：RZQ/6; “4”：RZQ/12; “5”：RZQ/8; “6”：保留; “7”：保留。
DIC	[5],[1]	RW,0	输出驱动电阻控制： “0”：RZQ/6; “1”：RZQ/7; “2”：RZQ/TBD; “3”：RZQ/TBD。
DLL	[0]	RW,0	DLL使能： “0”：表示使能; “1”：表示不使能。
—	其它	—	保留。

2.2.36 CfgMR2

偏移地址：0x01380

CfgMR2 为 DDR3 模式寄存器 2；写该寄存器会触发存储控制器向存储器发一条 MRS 命令，配置 DDR3 存储器颗粒的 MR2 寄存器。

表 2-57：CfgMR2 寄存器域的描述

名称	范围	类型	描述
Rtt_WR	[10:9]	RW,0	动态ODT端接控制： “0”：关闭动态ODT; “1”：RZQ/4;

			“2”：RZQ/2; “3”：保留。
MBZ	[8]	RW,0	必需配置为“0”。
SRT	[7]	RW,0	自刷新温度范围。 0: 普通温度范围; 1: 扩展温度范围。
ASR	[6]	RW,0	自动自刷新 (ASR, Auto Self-Refresh)。 0: 人工SR; 1: ASR使能 (可选)。
CWL	[5:3]	RW,3	CAS 写延迟。配置值, 与周期数的对应关系如下: “0”：5; “1”：6; “2”：7; “3”：8; “4”：9; “5”：10; “6”：11; “7”：12。
PASR	[2:0]	RW,0	部分阵列自刷新 (可选功能)。 0: Full Array; 1: 1/2 Array; (Bank0~3); 2: 1/4 Array; (Bank0~1); 3: 1/8 Array; (Bank0); 4: 3/4 Array; (Bank2~7); 5: 1/2 Array; (Bank4~7); 6: 1/4 Array; (Bank6~7); 7: 1/8 Array; (Bank7);
—	其它	—	保留。

2.2.37 CfgMR3

偏移地址: 0x01400

CfgMR3 为 DDR3 模式寄存器 3; 写该寄存器会触发存储控制器向存储器发一条 MRS 命令, 配

置 DDR3 存储器颗粒的 MR3 寄存器。

表 2-58: CfgMR3 寄存器域的描述

名称	范围	类型	描述
MPR	[2]	RW,0	MPR。
MPRLoc	[1:0]	RW,0	MPR地址。 0: 预定义向量。 1: RFU。 2: RFU。 3: RFU。 根据申威421处理器应用 (BL8), 地址A[2:0]必须为零。 Burst顺序为: 0,1,2,3,4,5,6,7。 预定义数据格式: [0,1,0,1,0,1,0,1]。 其它配置 (1/2/3) 时, Burst顺序相同, 但数据格式与此不同。
—	其它	—	保留。

2.2.38 PERF_CTRL

偏移地址: 0x01480

PERF_CTRL 寄存器为性能计数控制寄存器, 与 MC_PERF0、MC_PERF1、RWCNTOFREF 寄存器相关。

表 2-59: PERF_CTRL 寄存器域的描述

名称	范围	类型	描述
RWCntClr	[2]	RW,0	读写命令计数器清除控制。 对该位写 1, 将清除 RWCNTOFREF 寄存器。
PerfCtrl	[1:0]	RW,0	存控性能计数器控制信息。 0: 默认值, 不做任何操作动作; 1: 表示清除计数器; 对该位写 1, 对 MC_PERF0、MC_PERF1 进行清零。 2: 表示启动计数器; 该位为 2, 对 MC_PERF0、MC_PERF1 进行计数; 3: 表示停止计数器计数; 该位为 3, 对

			MC_PERF0、MC_PERF1 停止计数;
--	--	--	-------------------------

2.2.39 MC_PERF0

偏移地址：0x01500

MC_PERF0 寄存器记录发出的读写请求数量以及页直接命中的数量。

表 2-60: MC_PERF0 寄存器域的描述

名称	范围	类型	描述
ReqCnt_ALL	[63:32]	RO,0	存控仲裁出去的读写请求数
PDHCnt	[31:0]	RO,0	存控仲裁出去的页直接命中的读写命令数

2.2.40 MC_PERF1

偏移地址：0x01700

MC_PERF1 寄存器记录发出的读写请求页命中和页缺失的数量。

表 2-61: MC_PERF1 寄存器域的描述

名称	范围	类型	描述
PHCnt	[63:32]	RO,0	存控仲裁出去的页命中的读写命令数
PMCnt	[31:0]	RO,0	存控仲裁出去的页缺失的读写命令数

2.2.41 RWCNTOFREF

偏移地址：0x01780

RWCNTOFREF 寄存器记录在一个刷新周期内各种命令的情况。

表 2-62: RWCNTOFREF 寄存器域的描述

名称	范围	类型	描述
RdWrCntOfREF_Max	[63:48]	RO,0	一个刷新周期内仲裁的读写命令数的峰值
ConRdCntOfREF_Max	[47:32]	RO,0	一个刷新周期内连续仲裁读命令数的峰值
ConWrCntOfREF_Max	[31:16]	RO,0	一个刷新周期内连续仲裁写命令数的峰值
RdWrTurnCntOfREF_Max	[15:0]	RO,0	一个刷新周期内读写命令切换次数的峰值

2.2.42 MEMSERR_TH

偏移地址：0x01600

存储控制器会对存储器返回的数据和来自 CPM 的写数据进行 ECC 校验，该寄存器控制 MEMSERR_CNT 和 WRDATASERR_CNT 的计数，当指定的时间窗口内单错计数达到阈值，则向维护接口报错。

表 2-63: MEMSERR_TH 寄存器域的描述

名称	范围	类型	描述
SErrThreshold	[10:6]	RW,0x1F	主存单错计数阈值。
SErrWdsThresholdEn	[5]	RW,1	主存单错时间窗口阈值使能。
SErrWdsThreshold	[4:0]	RW,0x1F	主存单错时间窗口阈值。
—	其它	—	保留。

2.2.43 MEMSERR_CNT

偏移地址：0x01580

存储控制器会对存储器返回的数据进行 ECC 校验，该寄存器记录 ECC 单错的次数。寄存器 MEMSERR_TH 控制该寄存器的计数，当指定的时间窗口内单错计数达到阈值，则向系统报错。同时也会对 ECC 多错进行记录，并报给维护接口。

表 2-64: MEMSERR_CNT 寄存器域的描述

名称	范围	类型	描述
MEMMErr	[33]	RW1C,0	Mem读数据多错标志： [33]: 表示Mem读数据发生ECC多错。 软件写该寄存器时，如果数据部分bit 33为1，则清除该寄存器的[33]位。 存储控制器对所有读主存数据都重新生成ECC，并将重新生成的ECC与数据返回给目标方（各核心或各设备方），当读主存数据发生多错时由存储控制器通知维护接口。
CntMEMSErr	[32:0]	RW1C,0	Mem读数据单错信息： [32]: 表示Mem读数据发生ECC单错预警； [31:0]: 表示Mem读数据时间窗口内发生ECC单错的次数。 软件写该寄存器时，如果数据部分bit 0为1，则清除该寄存器的[31:0]。如果数据部分bit 32为1，则清除该寄存器的[32]

			位。
—	其它	—	保留。

2.2.44 WRDATASERR_CNT

偏移地址：0x01680

存储控制器会对来自一致性处理部件的写数据进行 ECC 校验，对读写请求的控制信息进行偶校验，该寄存器记录 ECC 单错的次数，同时记录 ECC 多错和偶校验错标志。寄存器 MEMSERR_TH 控制该寄存器的计数，当指定的时间窗口内单错计数达到阈值，则向维护接口报错，同时也会对 ECC 多错和偶校验错进行记录，并报给维护接口。

表 2-65: WRDATASERR_CNT 寄存器域的描述

名称	范围	类型	描述
WRDATAMErr	[34:33]	RW1C,0	CPM下行通路多错标志： [34]：表示下行异步FIFO写数据通路发生ECC多错； [33]：表示下行异步FIFO控制信息通路发生偶校验错。 软件写该寄存器时，如果数据部分bit 34为1，则清除该寄存器的[34]位。如果数据部分bit 33为1，则清除该寄存器的[33]位。
CntWRDATASErr	[32:0]	RW1C,0	CPM下行通路单错信息： [32]：表示下行通路发生ECC单错预警； [31:0]：表示下行通路在时间窗口内发生ECC单错的次数。 软件写该寄存器时，如果数据部分bit 0为1，则清除该寄存器的[31:0]。如果数据部分bit 32为1，则清除该寄存器的[32]位。
—	其它	—	保留。

2.2.45 MEMMERRDATAx

偏移地址：0x01800, 0x01880, 0x01900, 0x01980, 0x01a00

MEMMERRDATA 记录 ECC 多错对应的 288 位数据，包括 32 位的 ECC。

表 2-66: MEMMERRDATA 寄存器域的描述

名称	范围	类型	描述
MEMMERRDATA0	[63:0]	RO,0	DATA[63:0]
MEMMERRDATA1	[63:0]	RO,0	DATA[127:64]
MEMMERRDATA2	[63:0]	RO,0	DATA[191:128]
MEMMERRDATA3	[63:0]	RO,0	DATA[255:192]
MEMMERRDATA4	[63:0]	RO,0	{8'h0,ECC[31:24],8'h0,ECC[23:16], 8'h0,ECC[15:8],8'h0,ECC[7:0]}

注：该寄存器为申威 421 处理器新增。

2.2.46 MEMMERRADDR

偏移地址：0x01a80

MEMMERRADDR 记录 ECC 多错对应的地址。

表 2-67: MEMMERRADDR 寄存器域的描述

寄存器名	范围	类型	描述
RankAddr	[36:35]	RO,0	Rank 地址。
BankAddr[2:0]	[34:32]	RO,0	Bank 地址。
RowAddr[15:0]	[31:16]	RO,0	Row 地址。
ColAddr[15:0]	[15:0]	RO,0	Col 地址。
—	其它	—	保留。

注：该寄存器为申威 421 处理器新增。

2.3 存储器地址/命令/数据接口相关 IO 寄存器

AC 和 Datx8 的复位由其内部的 PLL 模块生成，在复位之前，从 IO 寄存器中读出的数据为不定值。复位结束后，PHY 会自动开始初始化过程，该过程完成后，某些寄存器中读到的数据不再是复位值。

2.3.1 基地址说明

表 2-68：一致性处理部件 IO 寄存器基地址说明

模块	核心视角(PA[47:0])		维护视角(PA[39:0])		备注
	MC0	MC1	MC0	MC1	
核组 0	0x8000,0280,0000	0x8000,0380,0000	0x80,0280,0000	0x80,0380,0000	
核组 1	0x8010,0280,0000	0x8010,0380,0000	0x90,0280,0000	0x90,0380,0000	
核组 2	0x8020,0280,0000	0x8020,0380,0000	0xa0,0280,0000	0xa0,0380,0000	
核组 3	0x8030,0280,0000	0x8030,0380,0000	0xb0,0280,0000	0xb0,0380,0000	

注：各寄存器的偏移地址（相对于零地址）见具体寄存器说明。

2.3.2 RIDR

偏移地址：0x0000;

RIDR 是存储器地址/命令/数据接口版本标识寄存器。

表 2-69：RIDR 寄存器域的描述

名称	范围	类型	描述
UDRID	[31:24]	RW,0x0	用户自定义
PHYMJR	[23:20]	RO,0x1	AC/Datx8（统称PHY）大规模修改的标号。
PHYMDR	[19:16]	RO,0x0	AC/Datx8（统称PHY）中规模修改的标号。
PHYMNR	[15:12]	RO,0x0	AC/Datx8（统称PHY）小规模修改的标号。
PUBMJR	[11:8]	RO,0x3	AC/Datx8的总控部分（统称PUB）大规模修改的标号。
PUBMDR	[7:4]	RO,0x1	AC/Datx8的总控部分（统称PUB）中规模修改的标号。
PUBMNR	[3:0]	RO,0x0	AC/Datx8的总控部分（统称PUB）小规模修改的标号。
—	其它	—	保留。

2.3.3 PIR

偏移地址：0x0080

PIR 是存储器 PHY 初始化控制寄存器。配置 PIR 相应的各位，将触发相应的初始化流程，相应位在完成初始流程后进行自清除。

通过读取 PGSR 寄存器读取初始化状态。PIR[INIT]设置为 1 后，PGSR0[IDONE]并非立即清除，需要等待 20 个 Cfg 时钟后再读取 PGSR 寄存器。

表 2-70: PIR 寄存器域的描述

名称	范围	类型	描述
INITBYP	[31]	RW,0x0	初始化旁路：该位为“1”表示初始化阶段被旁路（包括PHY的初始化、DRAM初始化和PHY的Training）或停止初始化。用户可配置PIR寄存器(INIT位和其它相关位)进行人工触发。该位会自清除。
ZCALBYP	[30]	RW,0x0	阻抗校准旁路：该位为“1”表示初始化阶段跳过阻抗校准或停止阻抗校准。用户可配置PIR寄存器（INIT位和ZCAL位）进行人工触发。该位会自清除。
DCALBYP	[29]	RW,0x0	DDL校准Bypass：该位为“1”表示初始化阶段跳过DDL校准或停止DDL校准。用户可配置PIR寄存器（INIT位和DCAL位）进行人工触发。该位会自清除。
LOCKBYP	[28]	RW,0x0	PLL Lock Bypass：该位为“1”表示初始化阶段跳过PLL锁定或停止PLL锁定。用户可配置PIR寄存器（INIT配置成1、PLLINIT配置成1）进行人工触发。该位会自清除。
CLRSR	[27]	RW,0x0	清除状态寄存器：对该位写“1”将清除所有的状态寄存器，包括PGSR和DXnGSR；该位会自清除。
RDIMMINIT	[19]	RW,0x0	执行DRAM初始化之前(CKE已经为高)需要对RDIMM buffer芯片进行初始化。
CTLDINIT	[18]	RW,0x0	Controller DRAM初始化：该位为“1”表示DRAM初始化由Controller完成，否则DRAM初始化由PUB完成（内建）或由软件完成。
PLLBYP	[17]	RW,0x0	PLL Bypass；该位为“1”，AC和DATX8的PLL处于旁路模式。

ICPC	[16]	RW,0x0	<p>初始化完成端口（dfi_init_complete）配置。</p> <p>该位为“0”，该完成端口在完成PHY初始化（DLL锁定且完成阻抗校准）后有效；</p> <p>该位为“1”，该完成端口在完成PHY初始化、SDRAM初始化和PHY Training后有效。</p>
WREYE	[15]	RW,0x0	<p>Write Data Eye Training: 该位为“1”表示需要由PUB进行Write Data Eye Training。该位会进行自清除。</p>
RDEYE	[14]	RW,0x0	<p>Read Data Eye Training: 该位为“1”表示需要由PUB进行Read Data Eye Training。该位会进行自清除。</p>
WRDSKW	[13]	RW,0x0	<p>Write Data Bit Deskew: 该位为“1”表示需要由PUB进行Write Data Bit Deskew。该位会进行自清除。</p>
RDDSKW	[12]	RW,0x0	<p>Read Data Bit Deskew: 该位为“1”表示需要由PUB进行Read Data Bit Deskew。该位会进行自清除。</p>
WLADJ	[11]	RW,0x0	<p>Write Leveling Adjust: 该位为“1”表示需要由PUB进行Write Leveling Adjust。该位会进行自清除。</p>
QSGATE	[10]	RW,0x0	<p>Read DQS Gate Training: 该位为“1”表示需要由PUB进行Read DQS Gate Training。该位会进行自清除。</p>
WL	[9]	RW,0x0	<p>Write Leveling: 该位为“1”表示需要由PUB进行Write Leveling。该位会进行自清除。</p>
DRAMINIT	[8]	RW,0x0	<p>DRAM初始化: 该位为“1”表示需要进行SDRAM初始化。该位会进行自清除。</p>
DRAMRST	[7]	RW,0x0	<p>DRAM复位: 该位为“1”表示需要进行DRAM复位（仅限于DDR3，驱动复位信号），且维持200us。可单独触发复位或者作为SDRAM初始化过程的一部分，对于后者Reset信号要保持200us后才可以开始SDRAM初始化流程。该位会进行自清除。</p>
PHYRST	[6]	RW,0x0	<p>PHY复位: 该位为“1”表示要复位AC和DATX8（驱动AC和DATX8的复位信号）。该位会进行自清除。</p>
DCAL	[5]	RW,0x0	<p>DDL校准: 该位为“1”表示需要进行DDL校准。该位会进行自清除。</p>
PLLINT	[4]	RW,0x0	<p>PLL初始化: 该位为“1”表示需要进行PLL初始化。该过程包括正确的驱动Power-Down、Reset和gear shift信号，且等待PLL稳定。该位会进行自清除。</p>

ZCAL	[1]	RW,0x0	阻抗校准：该位为“1”表示需要进行输出阻抗和输入端接电阻校准。该步和PHY的初始化(PLL初始化+DDL校准+PHY的复位)同步进行。该位会进行自清除。
INIT	[0]	RW,0x0	初始化触发位：对该位写“1”则触发初始化，包括PHY初始化、DRAM初始化和PHY Training。具体的初始化步骤根据[15:1]位进行独立使能（不使能则被跳过）。该位会进行自清除。
—	其它	—	保留。

注：该寄存器相对于申威 411 处理器有修改。

2.3.4 PGCR0-2

偏移地址：0x00100;0x00180;0x01180;

PGCR0-1 是 PUB 的通用控制寄存器。

表 2-71：PGCR0 寄存器域的描述

名称	范围	类型	描述
CKEN	[31:26]	RW,0x2a	CK使能，用于控制给SDRAM的SDR_CLK时钟。每个CK两位控制，具体含义如下： 2'b00：保持为0； 2' 01：反向输出； 2'b10：正常输出； 2'b11：保持为1。
PUBMODE	[25]	RW,0x0	该位为“1”，则由PUB控制PHY与SDRAM接口，此时PUB的dfi接口被忽略。
DTOSEL	[18:14]	RW,0x0	数字测试输出（phy_dto）控制选择； 5'b00000：DATX8 0 PLL数字测试输出； 5'b00001：DATX8 1 PLL数字测试输出； 5'b00010：DATX8 2 PLL数字测试输出； 5'b00011：DATX8 3 PLL数字测试输出； 5'b00100：DATX8 4 PLL数字测试输出； 5'b00101：DATX8 5 PLL数字测试输出； 5'b00110：DATX8 6 PLL数字测试输出； 5'b00111：DATX8 7 PLL数字测试输出； 5'b01000：DATX8 8 PLL数字测试输出；

			5'b01001: AC PLL数字测试输出; 5'b01010~5'b01111: 保留; 5'b10000: DATX8 0 DDL数字测试输出; 5'b10001: DATX8 1 DDL数字测试输出; 5'b10010: DATX8 2 DDL数字测试输出; 5'b10011: DATX8 3 DDL数字测试输出; 5'b10100: DATX8 4 DDL数字测试输出; 5'b10101: DATX8 5 DDL数字测试输出; 5'b10110: DATX8 6 DDL数字测试输出; 5'b10111: DATX8 7 DDL数字测试输出; 5'b11000: DATX8 8 DDL数字测试输出; 5'b11001: AC PLL数字测试输出; 5'b11010~5'b11111: 保留。
OSCWDL	[13:12]	RW,0x3	DLL 环振测试模式Write-Leveling Delay Line选择, 处于inactvie的Delay Line的delay value设置为零。 2'b00: no WL LDCL is active; 2'b01: DDR WL LCDL is active; 2'b10: SDR WL LCDL is active; 2'b11: Both WL LCDL is active。
OSCDIV	[11:9]	RW,0x7	DLL 环振测试模式分频输出。 0: 1分频; 1: 256分频; 2: 512分频; 3: 1024分频; 4: 2048分频; 5: 4096分频; 6: 8192分频; 7: 65536分频。
OSCEN	[8]	RW,0x0	环振模式使能。
DLTST	[7]	RW,0x0	Delay Line Test Start, 对该位写“1”表示触发DLL 环振测试。
DLTMODE	[6]	RW,0x0	Delay Line Test Mode, 该位为“1”表示处于DLL环振测试模式。
RDBVT	[5]	RW,0x1	Read Data BDL VT补偿使能。

			对应DXnBDLR3~DXnBDLR4。
WDBVT	[4]	RW,0x1	Write Data BDL VT补偿使能。 对应DXnBDLR0~DXnBDLR2。
RGLVT	[3]	RW,0x1	Read DQS Gating LCDL VT补偿使能。 对应DXnLCDLR2[RxDQSD]。
RDLVT	[2]	RW,0x1	Read DQS LCDL VT补偿使能。 对应DXnLCDLR1[RDQSD]。
WDLVT	[1]	RW,0x1	Write DQ LCDL VT补偿使能。 对应DXnLCDLR1[WDQD]。
WLLVT	[0]	RW,0x1	Write Leveling LCDL VT补偿使能。 对应DXnLCDLR0[RxWLD]。
—	其它	—	保留。

表 2-72: PGCR1 寄存器域的描述

名称	范围	类型	描述
LBMODE	[31]	RW,0x0	Loopback Mode: 该位为“1”表示处于LoopBack测试模式。
LBGDQS	[30:29]	RW,0x0	Loopback DQS Gating选择。 2'b00: DQS Gate总是开启; 2'b01: DQS Gate由PUB触发; 2'b10: DQS Gate由软件触发; 2'b11: 保留。
LBDQSS	[28]	RW,0x0	LoopBack DQS Shift选择。 0: PUB把Read DQS LCDL设置为0, DQS已经被Write Path移位90度; 1: Read DQS移位通过软件配置。
IOLB	[27]	RW,0x0	IO Loop-Back选择。 0: 在output buffer之后 (pad) 环回, 必须控制输出使能; 1: 在output buffer之前 (core sice) 环回, 可以不关心输出使能。
INHVT	[26]	RW,0x0	禁止VT调节, 该位写“1”则禁止VT调节。在直接配置delay Line寄存器时应该配置该位为“1”。
DXHRST	[25]	RW,0x1	PHY高速逻辑复位; 对该位写“0”会复位AC和DTAX8的高速逻辑; 对该位写“1”结束复位。

ZCKSEL	[24:23]	RW,0x2	电阻时钟驱动选择; 用于选择用于阻抗校准控制逻辑的时钟和CFG_CLK的频率关系; 0: 1分频; 1: 8分频; 2: 32分频; 3: 64分频。
DLDLMT	[22:15]	RW,0x1	延时线上PVT调节阈值(单个方向上), 超出这个阈值将启动VT调节。该位设置为零, 则Disable PVT调节。
FDEPTH	[14:13]	RW,0x2	滤波深度, 指定初始化期间或DCAL被触发后周期测量时的滤波深度。 0: 2; 1: 4; 2: 8; 3: 16。
LPFDEPTH	[12:11]	RW,0x0	低通滤波深度, 指定MDL周期测量时的滤波深度。 0: 2; 1: 4; 2: 8; 3: 16。
LPFEN	[10]	RW,0x1	低通滤波使能。该位为“1”, 则MDL周期测量的低通滤波使能。
MDLEN	[9]	RW,0x0	Master Delay Line使能。该位为“1”, AC的Master delay line在初始周期测量后会执行后续的周期测量。
IODDRM	[8:7]	RW,0x0	IO DDR Mode。
WLSELT	[6]	RW,0x1	Write Leveling Select Type: 0: 90度的Setup margin、90度的hold margin; 1: 135度的Setup margin、45度的hold margin。
ACHRST	[5]	RW,0x1	AC PHY High-Speed Reset 。写零则复位AC Marco, 并不复位PUB。该位会被自清(恢复成1)。

WSLOPT	[4]	RW,0x0	写延时优化。控制是否对dfi接口（给PHY）的AC信号插入一拍的站台，以考虑Write Levling的PipeLine调节（WL-1）。 0: 仅根据每个Rank中的Write Levling的PipeLine训练结果（WL-1）来增加一个站台。 1: 增加一个站台。
WLSTEP	[2]	RW,0x0	Write Leveling步长控制。 0: 1/2 DXnGSR0.WLPRD; 1: 1 step。
WLMODE	[1]	RW,0x0	Write Leveling (software) Mode: 该位为“1”，表示软件Write Leveling模式，为单步执行，由软件触发PIR.WL执行，DQ状态被采样到DXnGSR0.WLDQ中。
PDDISDX	[0]	RW,0x1	Power-Down Disable Byte Enable: 该位为“1”，被Disables的字节的全DLL和IO会PowerDown。
—	其它	—	保留。

表 2-73: PGCR2 寄存器域的描述

名称	范围	类型	描述
DYNACPDD	[31]	RW,0x0	动态AC驱动（输出）Power Down。该位为1，则ADDR和BA的输出会动态PowerDown。该位与ACIOCR[3]进行或操作，共同起作用。
LPMSTRC0	[30]	RW,0x0	Low-Power Master Channel 0。Master Channel 0低功耗。只在shared-AC模式下有效。
ACPDDC	[29]	RW,0x0	AC Power-Down with Dual Channels。只在shared-AC模式下有效。
SHRAC	[28]	RW,0x0	Shared-AC mode。两个数据通道共用AC通道。共用AC通读模式在使用CoreConsultant生成代码时就要指定。
DTPMXTM R	[27:20]	RW,0xf	Data Training PUB Mode Timer Exit。 指定在进行DataTraining时进入和退出PUB Mode的时间。

FXDLAT	[19]	RW,0x0	固定延迟：指定是否所有的读数据按固定延时返回给控制器。使能固定延时将会增加访存延时。 0：读固定延时无效（不使能）。 1：读固定延时有效（使能）。 读延时的计算公式为： $(12 + (\text{maximumDXnGTR.RxDGSL}) / 2) \text{ HDR clock cycles}$
NOBUB	[18]	RW,0x0	无气泡：定义是否允许返回给控制其的读数据之间有气泡。如果使能无气泡配置，则会增加访存延时。 0：允许读之间有气泡 1：不允许读之间有气泡 【当前理解】 ：每个读命令的读数据在PHY->控制器接口传输两拍，气泡是对这两拍而言。
tREFPRD	[17:0]	RW, 0x12480	Refresh Period。 该默认值是按SDRAM时钟1066MHz（DDR3-2133）计算，9个刷新闻隔的时间。 $0x12480 * (1/1.066) / 9 = 7.8\mu s$ 。
—	其它	—	保留。

注：该寄存器相对于申威 411 处理器有修改。

2.3.5 PGSR0-1

偏移地址：0x00200;0x00280

PGSR0-1 是 PUB 的通用状态寄存器。

表 2-74：PGSR0 寄存器域的描述

名称	范围	类型	描述
APLOCK	[31]	RO,0x0	AC PLL锁定信号。
PLDONE_CHN	[29:28]	RO,0x0	PLL Lock Done per Channel. [28]: Channel 0; [29]: Channel 1.
WEERR	[27]	RO,0x0	Write Eye Training错误。
REERR	[26]	RO,0x0	Read Eye Training错误。
WDERR	[25]	RO,0x0	Write Bit Deskew错误。
RDERR	[24]	RO,0x0	Read Bit Deskew错误。

WLAERR	[23]	RO,0x0	Write Leveling PipeLine调节错误。
QSGERR	[22]	RO,0x0	DQS Gate训练错误。
WLERR	[21]	RO,0x0	Write Leveling错误。
ZCERR	[20]	RO,0x0	阻抗校准错误。
WEDONE	[11]	RO,0x0	Write Eye Training完成。
REDONE	[10]	RO,0x0	Read Eye Training完成。
WDDONE	[9]	RO,0x0	Write Data Deskew完成。
RDDONE	[8]	RO,0x0	Read Data Deskew完成。
WLADONE	[7]	RO,0x0	Write Leveling PipeLine完成。
QSGDON	[6]	RO,0x0	DQS Gate训练完成。
WLDONE	[5]	RO,0x0	Write Leveling完成。
DIDONE	[4]	RO,0x0	DRAM初始化完成。
ZCDONE	[3]	RO,0x0	阻抗校准完成。
DCDONE	[2]	RO,0x0	DLL校准完成。
PLDONE	[1]	RO,0x0	PLL锁定完成。
IDONE	[0]	RO,0x0	初始化完成。
—	其它	—	保留。

表 2-75: PGSR1 寄存器域的描述

名称	范围	类型	描述
PARERR	[31]	RO,0x0	RDIMM Parity Error. 该位表示在访问 RDIMM buffer chip 时产生偶校验错。 该位保持有效直到被 PIR.CLRSR 清除。
VTSTOP	[30]	RO,0x0	VT Stop. 如果设置为 1, 则 VT 计算逻辑则停止 VT 补偿计算。 在设置 PGCR[INHVT]后, 写入 DDL 寄存器之前, 应该确认 VTSTOP 以确保 VT 补偿被停止。
DLTCODE	[24:1]	RO,0x0	延迟线测试结果, 返回 AC 延迟线测量 (DLL 环振测试) 得到的周期数。
DLTDONE	[0]	RO,0x0	延迟线测试结束, 表示已经完成了 AC 的延迟线测量 (DLL 环振测试)。
—	其它	—	保留。

注: 该寄存器相对于申威 411 处理器有修改。

2.3.6 PLLCR

偏移地址：0x00300

PLLCR 是 PLL 控制寄存器。

表 2-76: PLLCR 寄存器域的描述

名称	范围	类型	描述
BYP	[31]	RW,0x0	PLL旁路。
PLL_RST	[30]	RW,0x0	PLL复位，会驱动pll_rst。对该写“1”将复位pll。该位不会自清除，需要写零结束复位。
PLL_PD	[29]	RW,0x0	PLL Power Down，会驱动pll_pwrnd。对该写“1”将Power Down PLL。该位不会被自清除，需要写零退出。
FRQSEL	[19:18]	RW,0x0	PLL频率选择，根据PHY的不同，该位信息配置有所不同。 对于可以工作在2133Mbps的PHY： 2'b00：PLL参考时钟在335MHz~533MHz； 2'b01：PLL参考时钟在225MHz~385MHz； 2'b10：保留； 2'b11：PLL参考时钟在166MHz~275MHz。 对于不可以工作在2133Mbps的PHY： 2'b00：PLL参考时钟在250MHz~400MHz； 2'b01：PLL参考时钟在166MHz~300MHz； 2'b10：保留； 2'b11：保留。
QPMODE	[17]	RW,0x0	PLL Quadrature Phase Mode: Enables, if set, the quadrature phase clock outputs. This mode is not used in this version of the PHY. 该模式在该版本的PHY上不使用。
CPPC	[16:13]	RW,0xe	Charge Pump proportional Current Control，调节压控振荡器的控制电流。
CPIC	[12:11]	RW,0x0	Charge Pump integrating Current Control，调节压控振荡器的控制电流。
GSHIFT	[10]	RW,0x0	Gear Shift使能，如果使能则快速锁定模式。
ATOEN	[9:6]	RW,0x0	PLL模拟测试使能。 4'b0000：ALL PLL模拟测试输出为三态；

			4'b0001: AC PLL模拟测试输出; 4'b0010: DATX8 0 PLL模拟测试输出; 4'b0011: DATX8 1 PLL模拟测试输出; 4'b0100: DATX8 2 PLL模拟测试输出; 4'b0101: DATX8 3 PLL模拟测试输出; 4'b0110: DATX8 4 PLL模拟测试输出; 4'b0111: DATX8 5 PLL模拟测试输出; 4'b1000: DATX8 6 PLL模拟测试输出; 4'b1001: DATX8 7 PLL模拟测试输出; 4'b1010: DATX8 8 PLL模拟测试输出; 4'b1011~4'b1111: 保留。 PLL数字测试使能由PGCR0[DTOSEL]进行控制。
ATC	[5:2]	RW,0x0	PLL模拟测试控制选择（通过pll_ato输出）; 4'b0000: 保留; 4'b0001: vdd_ckin; 4'b0010: rfbf; 4'b0011: vdd_cko; 4'b0100: vp_cp; 4'b0101: vpfil; 4'b0111: gd; 4'b1000: vcntrl_atb; 4'b1001: vref_atb; 4'b1010: vpsf_atb; 4'b1011~4'b1111: 保留。
DTC	[1:0]	RW,0x0	PLL数字测试控制选择（通过Pll_dto[1]输出） 2'b00: 不使能，输出为零; 2'b01: PLL x1 clock; 2'b10: PLL参考时钟; 2'b11: PLL feedback clock; 通过 PGCR0[DTOSEL]选择 AC 或 DATX8 的 PLL 或 DDL 测试输出到 DTO[1:0]。对于 DDL 测试只有 DTO[0]有意义。对于 PLL 测试，两位都有意义，且 DTO[0]为反馈时钟，DTO[1]根据 DTC 可控制。
—	其它	—	保留。

注：该寄存器相对于申威 411 处理器有修改。

2.3.7 PTR0-4

偏移地址：0x00380,0x00400,0x00480,0x00500,0x00580

PTR 是延时控制寄存器。

表 2-77: PTR0 寄存器域的描述

名称	范围	类型	描述
tPLLPD	[31:21]	RW, 0x216	PLL Power Down时间。在Power Down状态维持的时间，即CFG复位结束到PLL PowerDown结束的时间。该时间要大于等于1us。默认值是25MHz（CFG时钟）的16us。
tPLLSGS	[20:6]	RW, 0x856	PLL Gear Shift Time。从PLL Reset pin无效到PLL gear Shift pin无效的时间。该时间要大于等于40us，默认值是25MHz（CFG时钟）的80us。
tPHYRST	[5:0]	RW, 0x10	PHY Rest Time。PHY Reset在结束PHY校准之后复位信号必须维持的时间。默认值是25MHz（CFG时钟）的400ns。
—	其它	—	保留。

表 2-78: PTR1 寄存器域的描述

名称	范围	类型	描述
tPLLLOCK	[31:16]	RW, 0xd056	PLL锁定时间。从PLL复位结束到PLL锁定的时间。该时间要大于等于30us。默认值是25MHz（CFG时钟）的480us。
tPLL RST	[12:0]	RW, 0x12c0	PLL复位时间；从PLL PowerDown结束到PLL复位结束的时间。该时间要大于等于3.5us。默认值是25MHz（CFG时钟）的56us。
—	其它	—	保留。

表 2-79: PTR2 寄存器域的描述

名称	范围	类型	描述
tWLDLYS	[19:15]	RW,0x10	Write Leveling Delay Setting Time。Write Leveling Delay设置到DQS为高的节拍数（SDR时钟）。
tCALH	[14:10]	RW,0xf	Calibration Hold Time: Cal_en有效到Cal_clk_en有效的节拍数（SDR时钟）。

tCALS	[9:5]	RW,0xf	Calibration Setup Time: Cal_clk_en无效到cal_en有效的节拍数 (SDR时钟)。
tCALON	[4:0]	RW,0xf	Calibration ON Time: Cal_clk_en有效的节拍数 (SDR时钟)。
—	其它	—	保留。

表 2-80: PTR3 寄存器域的描述

名称	范围	类型	描述
tDINIT1	[28:20]	RW, 384	SDRAM初始化流程, CKE为高到第一个命令的时间, 默认值是800Mhz (DDR3-1600) 的480ns。
tDINIT0	[19:0]	RW, 533334	SDRAM初始化流程, Reset无效但CKE仍需要为低的时间, 默认值是800Mhz (DDR3-1600) 的666us。
—	其它	—	保留。

表 2-81: PTR4 寄存器域的描述

名称	范围	类型	描述
tDINIT3	[27:18]	RW, 683	SDRAM初始化流程, ZQ命令到第一个命令的时间, 默认值是800Mhz (DDR3-1600) 的853ns。
tDINIT2	[17:0]	RW, 213334	SDRAM初始化流程, Rest维护有效的时间, 默认值是800Mhz (DDR3-1600) 的266us。
—	其它	—	保留。

2.3.8 ACMDLR

偏移地址: 0x00600

ACMDLR 为地址和命令接口的主延迟线 (Master Delay Line)。

表 2-82: ACMDLR 寄存器域的描述

名称	范围	类型	描述
MDLD	[23:16]	RW,0x0	MDL 延迟: 采用折半查找的方法, 对 DDR 周期进行测量, 该配置用于指定测量过程中 MDL 延迟值。
TPRD	[15:8]	RW,0x0	目标周期值。主延迟线周期测量值 (主延迟线更新功能使能时, 该值会周期性的更新)。该值用于延迟线校准, 用作比例系数的分子。其它延时线 (BDL) 根据该调节因子 (TPRD/ IPRD) 去计算延时线更新后的值, 如果偏差超过阈值, 则更新其它延迟线 (PHY 发起 Update 操作)。
IPRD	[7:0]	RW,0x0	初始周期值。主延迟线初始测量得到的周期值。该值用于

			延迟线校准，用作比例系数的分母。
—	其它	—	保留。

2.3.9 ACBDLR

偏移地址：0x00680

ACBDLR 为地址和命令接口的位延迟线（Bit Delay Line）。

表 2-83：ACBDLR 寄存器域的描述

名称	范围	类型	描述
ACBD	[23:18]	RW,0x0	AC Bit Delay。
CK2BD	[17:12]	RW,0x0	CK2 Bit Delay。
CK1BD	[11:6]	RW,0x0	CK1 Bit Delay。
CK0BD	[5:0]	RW,0x0	CK0 Bit Delay。
—	其它	—	保留。

2.3.10 ACIOCR

偏移地址：0x00700

ACIOCR 为地址和命令接口的 IO 配置寄存器。

表 2-84：ACIOCR 寄存器域的描述

名称	范围	类型	描述
ACSR	[31:30]	RW,0x0	AC Slew Rate选择（仅限D3F I/O）。 2'b00：非常快速（DDR3的默认配置）； 2'b01：快速； 2'b10：中速； 2'b11：慢速。
RSTIOM	[29]	RW,0x1	SDRAM Reset IO Mode选择。 0：选择SSTL；1：选择CMOS。
RSTPDR	[28]	RW,0x1	SDRM Reset Power Down Receiver。如果设置为“1”，Reset_N输入驱动处于Power-Down。
RSTPDD	[27]	RW,0x0	SDRM Reset Power Down Driver。如果设置为“1”，Reset_N输出驱动处于Power-Down。
RSTODT	[26]	RW,0x0	SDRM Reset ODT控制。如果设置为“1”，则Reset_N设置端接电阻。

RANKPDR	[25:22]	RW,0xf	RANK Power Down Receiver。如果设置为“1”，则CKE[3:0]、ODT[3:0]、CS[3:0]输入驱动处于Power-Down，其中RANKPDR[0]对应Rank0，RANKPDR[1]对应Rank1，依次类推。
CSPDD	[21:18]	RW,0x0	CS Power Down Driver。如果设置为“1”，则CS[3:0]输出驱动处于Power-Down。
RANKODT	[17:14]	RW,0x0	Rank ODT控制。如果设置为“1”，则CKE[3:0]、ODT[3:0]、CS[3:0]设置端接电阻，其中RANKODT[0]对应Rank0，RANKODT[1]对应Rank1，依次类推。
CKPDR	[13:11]	RW,0x7	CK Power Down Receiver。如果设置为“1”，CK[2:0]输入驱动处于Power-Down。
CKPDD	[10:8]	RW,0x0	CK Power Down Driver。如果设置为“1”，CK[2:0]输出驱动处于Power-Down。
CKODT	[7:5]	RW,0x0	CK ODT控制。如果设置为“1”，则CK[2:0]设置端接电阻。
ACPDR	[4]	RW,0x1	AC Power Down Receiver。如果设置为“1”，RAS_N、CAS_N、WE_N、BA[2:0]和A[15:0]输入驱动处于Power-Down。
ACPDD	[3]	RW,0x0	AC Power Down Driver。如果设置为“1”，RAS_N、CAS_N、WE_N、BA[2:0]和A[15:0]输出驱动处于Power-Down。
ACODT	[2]	RW,0x0	AC ODT控制。如果设置为“1”，则RAS_N、CAS_N、WE_N、BA[2:0]和A[15:0]设置端接电阻。
ACOE	[1]	RW,0x1	AC输出使能。
ACIOM	[0]	RW,0x0	AC IO Mode选择。 0: 选择SSTL; 1: 选择CMOS。
—	其它	—	保留。

2.3.11 DXCCR

偏移地址：0x00780

DXCCR 是 DatX8 部分的通用配置寄存器。

表 2-85: DXCCR 寄存器域的描述

名称	范围	类型	描述
----	----	----	----

DDPDRCD0	[31:28]	RW,0x 4	动态数据PowerDown（输入）倒计时阈值。 用于计算接收器上电的时间窗口。
DDPDDCDO	[27:24]	RW,0x 4	动态数据PowerDown（输出）倒计时阈值。 用于计算驱动上电所需的时间窗口。
DYNDXPDR	[23]	RW,0x 0	动态数据PowerDown（输入），该配置影响所有活动Datx8的DQ引脚。只有在DXPDR和DXnGCR.DXPDR不被设置成1时，该配置有意义。接收器在DFI READ指令到来时上电，在此指令之后经过（trddata_en +fixed_read_latency+n）HDR cycles 时间后断电。式中的n由寄存器DXCCR[31:28]来决定。
DYNDXPDD	[22]	RW,0x 0	动态数据PowerDown（输出），该配置影响所有活动Datx8的DQ引脚。只有在DXPDD和DXnGCR.DXPDD不被设置成1时，该配置有意义。驱动器在DFI WRITE指令到来时上电，在此指令之后经过（twrlat +WL2+N）HDR cycles 后断电。式中的n由寄存器DXCCR的[27:24]决定。
UDQIOM	[21]	RW,0x 0	无用比特的I/O类型选择。为0时为无用DQ端口的IO选择SSTL模式，为1时选择CMOS模式
UDQPDR	[20]	RW,0x 1	无用比特的接收（输入）被PowerDown。该位与MSBUDQ联合使用。
UDQPDD	[19]	RW,0x 1	无用比特的驱动（输出）被PowerDown。该位与MSBUDQ联合使用。
UDQODT	[18]	RW,0x 0	无用比特的端接电阻控制，“1”表示使能。该位与MSBUDQ联合使用。
MSBUDQ	[17:15]	RW,0x 0	字节内高段比特无意义标识，用于指定字节内高段的无意义比特。字节被用的位宽为[8-MSBDQ-1:0]。用DXnGCR.DXEN寄存器来使整个字节都无效。
DXSR	[14:13]	RW,0x0	该位控制所有DATX8中的DQ、DM、DQS、DQS_N的Slew rate。 2'b00: 非常快速（DDR3的默认配置）； 2'b01: 快速； 2'b10: 中速； 2'b11: 慢速。

DQSNRES	[12:9]	RW,0xc	<p>对DQS#的上拉、下拉电阻调节。</p> <p>[3]: 0表示上拉电阻调节, 1表示下拉电阻调节。</p> <p>[2:0]表示具体值, 具体如下:</p> <p>0: 没有片上电阻;</p> <p>1: 688欧姆;</p> <p>2: 611欧姆;</p> <p>3: 550欧姆;</p> <p>4: 500欧姆;</p> <p>5: 458欧姆;</p> <p>6: 393欧姆;</p> <p>7: 344欧姆。</p>
DQSRES	[8:5]	RW,0x4	<p>对DQS的上拉、下拉电阻调节。</p> <p>[3]: 0表示上拉电阻调节, 1表示下拉电阻调节。</p> <p>[2:0]表示具体值, 具体如下:</p> <p>0: 没有片上电阻;</p> <p>1: 688欧姆;</p> <p>2: 611欧姆;</p> <p>3: 550欧姆;</p> <p>4: 500欧姆;</p> <p>5: 458欧姆;</p> <p>6: 393欧姆;</p> <p>7: 344欧姆。</p>
DXPDR	[4]	RW,0x0	<p>Data Power Down Receiver。该位控制所有DATX8中的DQ、DM、DQS、DQS_N; 该位和每个独立的DATX8中的IOM配置信息共同起作用(与DXnGCR[DXPDR]“或”操作)。</p>
DXPDD	[3]	RW,0x0	<p>Data Power Down Driver。该位控制所有DATX8中的DQ、DM、DQS、DQS_N; 该位和每个独立的DATX8中的IOM配置信息共同起作用(与DXnGCR[DXPDD]“或”操作)。</p>
MDLEN	[2]	RW,0x1	<p>Master Delay Line使能。该位控制所有DATX8的Master delay Line校准; 该位和每个独立的DATX8中的IOM配置信息共同起作用(与DXnGCR[MDLEN]“或”操作)。</p>

DXIOM	[1]	RW,0x0	Data IO Mode选择。该位控制所有DATX8中的DQ、DM、DQS、DQS_N。 0: 选择SSTL; 1: 选择CMOS。 该位和每个独立的DATX8中的IOM配置信息共同起作用（与DXnGCR[DXIOM]“或”操作）。
DXODT	[0]	RW,0x0	Data ODT控制。该位控制所有DATX8中的DQ、DM、DQS、DQS_N的ODT。该位和每个独立的DATX8中的ODT配置共同起作用（与DXnGCR[DQSODT/DQODT]“或”操作）。
—	其它	—	保留。

注：该寄存器相对于申威 411 处理器有修改。

2.3.12 DSGCR

偏移地址：0x00800

DSGCR 是 DDR System 通用控制寄存器。

表 2-86：DSGCR 寄存器域的描述

名称	范围	类型	描述
CKEOT	[31]	RW,0x1	CKE输出使能。
RSTOE	[30]	RW,0x1	Reset_N输出使能。
ODTOE	[29]	RW,0x1	ODT输出使能。
CKOE	[28]	RW,0x1	CK输出使能。
ODTPDD	[27:24]	RW,0x0	ODT Power Down Driver, 用于关闭ODT引脚的输出功能。
CKEPDD	[23:20]	RW,0x0	CKE Power Down Driver, 用于关闭CKE引脚的输出功能。
SDRMODE	[19]	RW,0x0	SDR模式或HDR模式（4C处理器使用HDR模式）。 0: 表示HDR模式；1: 表示SDR模式。
RRMODE	[18]	RW,0x0	Rist-to-Rise模式。
ATOAE	[17]	RW,0x0	ATO 模拟测试使能: 有效时, 模拟测试在(ATO) I/O输出。
DTOOE	[16]	RW,0x0	DTO输出使能: 当有效时, 在DTP端口进行I/O输出驱动。
DTOIOM	[15]	RW,0x0	DTP引脚的I/O模式: 为0时选择DTP I/O模式为SSTL, 为1时选择CMOS模式。
DTPDR	[14]	RW,0x0	DTP Power Down Receive Powers down. 当该位为1, 则

		1	DTO引脚的接收被PowerDown。
DTOPDD	[13]	RW,0x1	DTO Power Down Driver Powers down。当该位为1, 则DTO引脚的驱动被PowerDown。
DTOODT	[12]	RW,0x0	DTO的端接电阻使能: 当有效时, 使能DTO的端接电阻。
PUAD	[11:8]	RW,0x4	PHY Update Acknowledge delay. Specifies the number of clock cycles that the indication for the completion of PHY update from the PHY to the controller should be delayed. This essentially delays, by this many clock cycles, the de-assertion of dfi_ctrlup_ack and dfi_phyupd_req signals relative to the time when the delay lines or I/Os are updated.
BRRMODE	[7]	RW,0x0	Bypass Rise-to-Rise Mode。该位为1, 则Bypass时是rise-to-rise模式, 否则是rise-to-fall模式。
DQSGX	[6]	RW,0x0	DQS Gate左右各延长1拍。
CUAEN	[5]	RW,0x0	Controller update Acknowledge使能。该位为“1”, PHY应该对来自controller的Update请求予以确认信号。
LPPLLPD	[4]	RW,0x1	Low Power PLL Power Down。与DFI Low Power有关; 如果设置为“1”, 则要响应DFI low power请求, 使PLL处于Power-Down。
LPIOPD	[3]	RW,0x1	Low Power IO Power Down。与DFI Low Power有关, 如果设置为“1”, 则要响应DFI low power请求, 使IO处于Power-Down。
ZUEN	[2]	RW,0x1	阻抗Update使能。该位为“1”, 当发起的Update请求时, PHY会相应的执行阻抗校准。
BDisEN	[1]	RW,0x1	Byte Disable Enable。该位为“1”, PHY会响应controller的Byte Disable请求, 否则被忽略, 此时只能通过DXnGCR寄存器进行Disable。
PUREN	[0]	RW,0x1	PHY Update请求使能。如果使能, 当DDL VT漂移时, PHY发Update请求。
—	其它	—	保留。

注: 该寄存器相对于申威 411 处理器有修改。

2.3.13 DCR

偏移地址：0x00880

DCR 是 DRAM 配置寄存器。

表 2-87: DCR 寄存器域的描述

名称	范围	类型	描述
UDIMM	[29]	RW,0x0	UDIMM地址镜像指示。
DDR2T	[28]	RW,0x0	2T模式指示。
NOSRA	[27]	RW,0x0	该位为“1”，则同一拍命令不能同时发射到两个Rank上。对于带镜像的双Rank UDIMM，该位必须设置为“1”。
BYTEMASK	[17:10]	RW,0x1	在DQS Gate训练时，选择相应的bit，MPR操作时该位要和PDQ一致。
MPRDQ	[7]	RW,0x0	MPR模式读时，指定非主要DQ pin上的值（主要DQ是0/1翻转）。 0：表示非主要DQ上是0； 1：表示非主要DQ上与主要DQ上一致。
PDQ	[6:4]	RW,0x0	MPR模式读时，指定主要DQ管脚，即从DQ[7:0]中进行8选一。
DDR8BNK	[3]	RW,0x1	8 Bank指示。
DDRMD	[2:0]	RW,0x3	SDRAM类型指示。 3'b000：保留； 3'b001：保留； 3'b010：DDR2； 3'b011：DDR3； 3'b100~3'b111：保留。
—	其它	—	保留。

注：该寄存器相对于申威 411 处理器有修改。

2.3.14 DTPR0-2

偏移地址：0x00900,0x00980,0x00a00

DTPR 是 DRAM 延时参数寄存器。

表 2-88: DTPR0 寄存器域的描述

名称	范围	类型	描述
tRC	[31:26]	RW,0x32	ACT命令到ACT命令之间的间隔(同Bank), 有效取值是0x2~0x2a。 根据DDR3-1600要求, 需要修改为0x27。
tRRD	[25:22]	RW,0x7	ACT命令到ACT命令之间的间隔(不同Bank), 有效取值是1~8。 根据DDR3-1600要求, 需要修改为6。
tRAS	[21:16]	RW,0x24	ACT命令到PRE命令之间的间隔, 有效取值是0x2~0x1f。 根据DDR3-1600要求, 需要修改为0x1c。
tRCD	[15:12]	RW,0xe	ACT命令到CAS命令之间的间隔, 有效取值是2~11。 根据DDR3-1600要求, 需要修改为9。
tRP	[11:8]	RW,0xe	PRE命令到ACT命令的间隔, 有效取值是2~11。 根据DDR3-1600要求, 需要修改为9。
tWTR	[7:4]	RW,0x8	WR命令到RD命令之间的间隔, 有效取值是1~6。 根据DDR3-1600要求, 需要修改为6。
tRTP	[3:0]	RW,0x8	RD命令到PRE命令之间的间隔, 有效取值是2~6。 根据DDR3-1600要求, 需要修改为6。
—	其它	—	保留。

表 2-89: DTPR1 寄存器域的描述

名称	范围	类型	描述
tAOND/ tAOFD	[31:30]	RW,0x0	ODT开关。(仅用于DDR2)
tWLO	[29:26]	RW,0x8	Write Leveling输出延时, 即从DQS的上升沿到通过DQ返回的时间。
tWLMRD	[25:20]	RW,0x28	Write Leveling Mode到第1个DQS上升沿的间隔。
tRFC	[19:11]	RW,374	Refresh-to-Refresh或Refresh-to-active的最小时间。 根据DDR3-1600和相应的颗粒类型修改该值。
tFAW	[10:5]	RW,38	四个ACT命令的窗口, 有效取值2~31。 根据DDR3-1600要求, 需要修改为31。
tMOD	[4:2]	RW,0x4	MRS命令与后续指令(非MRS)之间的间隔; 0: 12; 1: 13; 2: 14; 3: 15; 4: 16; 5: 17; 其它保留。

tMRD	[1:0]	RW,0x2	两个MRS命令之间的间隔，实际延时值是该配置值加4。
—	其它	—	保留。

表 2-90: DTPR2 寄存器域的描述

名称	范围	类型	描述
tCCD	[31]	RW,0x0	0表示4拍；1表示5拍。
tRTW	[30]	RW,0x0	Read to Write的总线切换间隔。 0: 标准的读写切换时间； 1: 额外增加1拍。
tRTODT	[29]	RW,0x0	Read ODT Delay: 0: ODT在Read后导后立即开启； 1: ODT在Read后导的下一个时钟才可以开启。 如果ODT使能，且该值配置为“1”，读到写的延时需要加1。
tDLLK	[28:19]	RW, 512	DDL锁定时间，有效取值2~1023。
tCKE	[18:15]	RW,0x6	CKE的最小脉冲宽度，有效取值2~15。 根据DDR3-1600要求，需要修改为6。
tXP	[14:10]	RW,0x1a	Power Down退出延时，有效取值2~31。
tXS	[9:0]	RW, 512	Self Refresh退出的延时，有效取值2~1023。 根据DDR3-1600要求，需要修改为288。
—	其它	—	保留。

2.3.15MR0-3

偏移地址: 0x00a80,0x00b00,0x00b80,0x00c00

MR 是 SDRAM 的模式寄存器。

表 2-91: MR0 寄存器域的描述

名称	范围	类型	描述
RSVD	[15:13]	RW,0x0	JEDEC保留(零);
PD	[12]	RW,0x0	Power Down Control: 0: Slow Exit (DLL off) ; 1: Fast Exit (DLL on) 。
WR	[11:9]	RW,0x5	Write Recovery延时: 0: 16;

			1: 5; 2: 6; 3: 7; 4: 8; 5: 10; 6: 12; 7: 14。
DR	[8]	RW,0x0	DLL Reset; 该位自清除;
TM	[7]	RW,0x0	0: 普通模式; 1: 测试模式;
CL	[6:4],[2]	RW,0xa	读延时: 2: 5; 4: 6; 6: 7; 8: 8; 10: 9; 12: 10; 14: 11; 1: 12; 3: 13; 5: 14。
BT	[3]	RW,0x0	突发类型 (4C要求该位配置为“1”) : 0: 顺序模式; 1: Wrap模式。
BL	[1:0]	RW,0x2	突发长度: 0: 表示固定8; 1: 表示4或8; 2: 表示固定4; 3: 保留。 4C要求该位配置为“0”。
—	其它	—	保留。

表 2-92: MR1 寄存器域的描述

名称	范围	类型	描述
RSVD	[15:13]	RW,0x0	JEDEC保留(零);

QOFF	[12]	RW,0x0	输出使能。 0=所有输出功能正常； 1=所有 SDRAM 输出关闭。
TDQS	[11]	RW,0x0	DQS 端接。 1=提供附加的端接电阻输出； 0=没有附加端接电阻。
RSVD	[10],[8]	RW,0x0	为 JEDEC 保留(零)。
LEVEL	[7]	RW,0x0	Write Leveling 使能信号。置“1”表示 SDRAM 进行 Write Leveling。
AL	[4:3]	RW,0x0	Posted CAS 附加延迟。允许 Read 和 Write 命令比正常更早的让 SDRAM 执行。 2'b00=不使能； 2'b01=CL-1； 2'b10=CL-2； 2'b11=保留。
RTT	[9],[6],[2]	RW,0x0	片上端接：为 SDRAM 选择有效的电阻。 3'b000=无效； 3'b001 = RZQ/4； 3'b010 = RZQ/2； 3'b011 = RZQ/6； 3'b100 = RZQ/12； 3'b101 = RZQ/8； 其它值保留。
DIC	[5],[1]	RW,0x0	输出驱动阻抗控制。 2'b00=RZQ/6； 2'b01=RZQ/7； 2'b10~2'b11=保留。
DE	[0]	RW,0x0	DLL 使能： 1=不使能； 0=使能。
—	其它	—	保留。

表 2-93: MR2 寄存器域的描述

名称	范围	类型	描述
RSVD	[15:11],[8]	RW,0x0	JEDEC保留(零)。

RTTWR	[10:9]	RW,0x0	<p>动态 ODT 选择。</p> <p>2'b00 = 关闭动态 ODT;</p> <p>2'b01 = RZQ/4;</p> <p>2'b10 = RZQ/2;</p> <p>2'b11 = 保留。</p>
SRT	[7]	RW,0x0	<p>自刷新温度范围。在自刷新过程中通过分析温度范围来选择刷新模式。</p>
ASR	[6]	RW,0x0	<p>该位为“1”表示SDRAM根据工作温度自动提供 Self-Refresh电源管理，否则必须根据温度范围通过SRT设置。</p>
CWL	[5:3]	RW,0x0	<p>CAS 写延迟，从 SDRAM 寄存器写命令到写数据有效的周期数。</p> <p>3'b000 = 5 (tCK > 2.5ns);</p> <p>3'b001 = 6 (2.5ns > tCK > 1.875ns);</p> <p>3'b010 = 7 (1.875ns > tCK > 1.5ns);</p> <p>3'b011 = 8 (1.5ns > tCK > 1.25ns);</p> <p>3'b100 = 9 (1.25ns > tCK > 1.07ns);</p> <p>3'b101=10 (1.07ns > tCK > 0.935ns);</p> <p>3'b110=11 (0.935ns>tCK > 0.833ns);</p> <p>3'b111=12 (0.833ns>tCK > 0.75ns);</p>
PASR	[2:0]	RW,0x0	<p>指定部分阵列进行 Self-Refresh。进入 Self Refresh 后，除指定区域外，其它区域的数据将被丢失。</p> <p>4 Bank 有效取值：</p> <p>3'b000 = Full Array;</p> <p>3'b001 = Half Array (BA[1]=0);</p> <p>3'b010 = Quarter Array (BA[1:0]= 2'b00);</p> <p>3'b011 = 没有定义;</p> <p>3'b100 = 3/4 Array (BA[1:0]=2'b00 除外);</p> <p>3'b101 = Half Array (BA[1]=1);</p> <p>3'b110 = Quarter Array (BA[1:0]= 2'b11);</p> <p>3'b111 =没有定义。</p> <p>8 bank 有效取值：</p> <p>3'b000 = Full Array;</p> <p>3'b001 = Half Array (BA[2]=0);</p>

			3'b010 = Quarter Array (BA[2:1]=2'b00); 3'b011 = 1/8 Array (BA[2:0]=3'b000); 3'b100 = 3/4 Array (BA[2:1]=2'b00 除外); 3'b101 = Half Array (BA[2]=1); 3'b110 = Quarter Array (BA[2:1]= 2'b11); 3'b111 = 1/8 Array (BA[2:0]= 3'b111);
—	其它	—	保留。

表 2-94: MR3 寄存器域的描述

名称	范围	类型	描述
RSVD	[15:3]	RW,0x0	JEDEC保留(零)。
MPR	[2]	RW,0x0	多用途寄存器 (MPR) 使能信号。置“1”表示从 MPR 中读取数据, 否则从 DRAM 阵列读取数据。
MPRLOC	[1:0]	RW,0x0	多用途寄存器 (MPR) 位置, 选择 MPR 数据的位置。 00=为系统校准按预定义模式; 其它值不能使用。 Burst顺序为: 0,1,2,3,4,5,6,7。 预定义数据格式: [0,1,0,1,0,1,0,1]。
—	其它	—	保留。

2.3.16 ODTCR

偏移地址: 0x00c80

ODTCR 为 ODT 控制寄存器。

表 2-95: ODTCR 寄存器域的描述

名称	范围	类型	描述
WRODT3	[31:28]	RW,0x8	写 ODT: 当一个 Write 命令发送到某 RANK 时, 是否将各 RANK 的 ODT 使能。 WRODT0~3 分别对 RANK0~3 执行 Write 命令时对各 RANK 的 ODT 的设置。 每个域四个 bit 分别对应四个 RANK。 缺省值时在 Write 命令时只对正在执行 Write 的 RANK 的 ODT 使能;
WRODT2	[27:24]	RW,0x4	
WRODT1	[23:20]	RW,0x2	
WRODT0	[19:16]	RW,0x1	
RDODT3	[15:12]	RW,0x0	读 ODT: 当一个 Read 命令发送到某 RANK 时, 是否将 RANK 的 ODT 使能。
RDODT2	[11:8]	RW,0x0	

RDODT1	[7:4]	RW,0x0	RDODT0~3 分别对 RANK0~3 执行 Read 命令时对各 RANK 的 ODT 的设置。 每个域四个 bit 分别对应四个 RANK。 缺省值时在 Read 命令时全部 ODT 不使能。
RDODT0	[3:0]	RW,0x0	

2.3.17 DTCR

偏移地址: 0x00d00

DTCR 为 Data Training 配置寄存器。

表 2-96: DTCR 寄存器域的描述

名称	范围	类型	描述
RFSHDT	[31:28]	RW,0x9	训练期间 Refresh: 非 0 值表示在每个 Rank 训练之后 (除了最后一个 rank), 将指定数量的刷新命令发送到相应的 SDRAM。
RANKEN	[27:24]	RW,0xF	RANK 使能: 指定 RANK 进行数据训练 (Read DQS Gate 训练), 每位对应一个 RANK。
DTEXG	[23]	RW,0x0	数据训练 DQS Gate 扩展使能。置“1”表示 DQS Gate 训练时, 按 DSGCR[DQSGX]要求, 扩展 DQS Gate。
DTEXD	[22]	RW,0x0	Data Training Extended Write DQS: Enables, if set, an extended write DQS whereby two additional pulses of DQS are added as post-amble to a burst of writes. Generally this should only be enabled when running read bit deskew with the intention of performing read eye deskew prior to running write leveling adjustment. 写 DQS 扩展。该配置有效时, 会在后导 (post-amble) 增加两个额外的脉冲。
DTDSTP	[21]	RW,0x0	Debug 数据训练单步模式, 该位会自清除。
DTDEN	[20]	RW,0x0	数据训练 Debug 使能: 置“1”表示以 Debug 模式进行数据训练。
DTDBS	[19:16]	RW,0x0	数据训练 Debug 字节选择。在数据训练 Debug 模式下, 选择相应字节。
DTWDQMO	[14]	RW,0x0	Data Training WDQ Margin Override: If set, the Training WDQ Margin value specified in DTCR[11:8] (DTWDQM) is used during data training. Otherwise the value is computed as 1/4 of

			the ddr_clk period measurement found during calibration of the WDQ LCDL. WDQ Margin (余量) 选择。为 1 时由 DTCR[11:8]决定, 否则会根据 1/4 的 ddr_clk 测量周期 (在 WDQLCDL 校准期间) 计算。
DTBDC	[13]	RW,0x1	Data Training Bit Deskew Centering: Enables, if set, eye centering capability during write and read bit deskew training. 读写数据的 Bit Skew 调节训练的中心对齐。
DTWBDDM	[12]	RW,0x1	DM 是否参与 Bit Deskew 调节的控制使能。
DTWDQM	[11:8]	RW,0x5	WDQ 训练的 Margin 设置。
DTCMPD	[7]	RW,0x1	数据训练数据比较。置“1”表示在 DQS Gate 训练时检查返回的数据是否正确。否则, 只对返回数据的个数 (DQS 边沿数量) 进行检查。
DTMPR	[6]	RW,0x0	使用 MPR 进行数据训练。置“1”, 指定使用 SDRAM 的 MPR 寄存器进行 DQS Gate 训练, 否则, 将先写入指定 SDRAM 空间, 然后读回。
DTRANK	[5:4]	RW,0x0	数据训练 RANK 号: 用于指定进行数据 bit deskew 和眼图训练的 Rank 号。
DTRPTN	[3:0]	RW,0x7	数据训练重复次数: 指定数据训练时读、写次数, 验证 DDR 的稳定性。
—	其它	—	保留。

注: 该寄存器相对于申威 411 处理器有修改。

2.3.18DTAR0-3

偏移地址: 0x00d80;0x00e00,0x00e80,0x00f00

DTAR 为 Data Training 地址寄存器。

表 2-97: DTAR0 寄存器域的描述

名称	范围	类型	描述
DTBANK	[30:28]	RW,0x0	数据训练 BANK 地址: 用于数据训练时, 指定 SDRAM 的 BANK 地址。
DTROW	[27:12]	RW,0x0	数据训练行地址: 用于数据训练时, 指定 SDRAM 行地址。
DTCOL	[11:0]	RW,0x0	数据训练列地址: 用于数据训练时, 指定 SDRAM 列地址。地址低三位必须为“3'b000”。

—	其它	—	保留。
---	----	---	-----

表 2-98: DTAR1 寄存器域的描述

名称	范围	类型	描述
DTBANK	[30:28]	RW,0x0	数据训练 BANK 地址: 用于数据训练时, 指定 SDRAM 的 BANK 地址。
DTROW	[27:12]	RW,0x0	数据训练行地址: 用于数据训练时, 指定 SDRAM 行地址。
DTCOL	[11:0]	RW,0x0 8	数据训练列地址: 用于数据训练时, 指定 SDRAM 列地址。地址低三位必须为“3'b000”。
—	其它	—	保留。

表 2-99: DTAR2 寄存器域的描述

名称	范围	类型	描述
DTBANK	[30:28]	RW,0x0	数据训练 BANK 地址: 用于数据训练时, 指定 SDRAM 的 BANK 地址。
DTROW	[27:12]	RW,0x0	数据训练行地址: 用于数据训练时, 指定 SDRAM 行地址。
DTCOL	[11:0]	RW,0x1 0	数据训练列地址: 用于数据训练时, 指定 SDRAM 列地址。地址低三位必须为“3'b000”。
—	其它	—	保留。

表 2-100: DTAR3 寄存器域的描述

名称	范围	类型	描述
DTBANK	[30:28]	RW,0x0	数据训练 BANK 地址: 用于数据训练时, 指定 SDRAM 的 BANK 地址。
DTROW	[27:12]	RW,0x0	数据训练行地址: 用于数据训练时, 指定 SDRAM 行地址。
DTCOL	[11:0]	RW,0x1 8	数据训练列地址: 用于数据训练时, 指定 SDRAM 列地址。地址低三位必须为“3'b000”。
—	其它	—	保留。

2.3.19 DTDR0-1

偏移地址: 0x00f80,0x01000

DTDR 为 Data Training 数据寄存器。

表 2-101: DTDR0 寄存器域的描述

名称	范围	类型	描述
----	----	----	----

DTBYTE3	[31:24]	RW,0XDD	数据训练的数据（字节 0~字节 3）。在数据训练期间，每个通道采用相同的数据。缺省单个周期序列步长为“1”。
DTBYTE2	[23:16]	RW,0X22	
DTBYTE1	[15:8]	RW,0XEE	
DTBYTE0	[7:0]	RW,0X11	
—	其它	—	保留。

表 2-102: DTDR1 寄存器域的描述

名称	范围	类型	描述
DTBYTE7	[31:24]	RW,0X77	数据训练的数据（字节 4~字节 7）。在数据训练期间，每个通道采用相同的数据。缺省单个周期序列步长为“1”。
DTBYTE6	[23:16]	RW,0X88	
DTBYTE5	[15:8]	RW,0XBB	
DTBYTE4	[7:0]	RW,0X44	
—	其它	—	保留。

2.3.20 DTEDR0-1

偏移地址：0x01080,0x01100

DTEDR 为 Data Training 眼图数据寄存器。

表 2-103: DTEDR0 寄存器域的描述

名称	范围	类型	描述
DTWBMX	[31:24]	RW,0x0	数据训练写通路 BDL shift 最大值。
DTWBMN	[23:16]	RW,0x0	数据训练写通路 BDL shift 最小值。
DTWLMX	[15:8]	RW,0x0	数据训练 WDQ LCDL 最大值。
DTWLMN	[7:0]	RW,0x0	数据训练 WDQ LCDL 最小值。
—	其它	—	保留。

表 2-104: DTEDR1 寄存器域的描述

名称	范围	类型	描述
DTRBMX	[31:24]	RW,0x0	数据训练读通路 BDL shift 最大值。
DTRBMN	[23:16]	RW,0x0	数据训练读通路 BDL shift 最小值。
DTRLMX	[15:8]	RW,0x0	数据训练 RDQS LCDL 最大值。
DTRLMN	[7:0]	RW,0x0	数据训练 RDQS LCDL 最小值。
—	其它	—	保留。

2.3.21 RDIMMGCR0-1

偏移地址：0x01600,0x01680

表 2-105: RDIMMGCR0 寄存器域的描述

名称	范围	类型	描述
MIRROR	[31]	RW,0x0	RDIMM 镜像。此寄存器位控制着 RDIMM 缓冲芯片 (chip buffer) 的 MIRROR 信号 (可以作为 IP 的输出引脚)。对于标准 RDIMM, 不需要该引脚, RDIMM 会根据实际需求把 chip buffer 的该引脚接死。地址镜像只对双 chip buffer 的 RDIMM 有意义, 为了平衡从 RDIMM 的 pin 到 chip buffer 的连线, 对 Back Side 的 chip buffer 进行地址镜像。
QCSEN	[30]	RW,0x0	Quad CS Enable。此寄存器控制着 RDIMM 缓冲芯片 (chip buffer) 的 QSEN_N 信号 (低有效, 可以作为 IP 的输出引脚)。对于标准 RDIMM, 不需要该引脚, RDIMM 会根据实际需求把 QSEN_N 的引脚接死。
MIRROROE	[29]	RW,0x1	MIRROR 引脚输出使能。
QCSENOE	[28]	RW,0x1	QCSEN 引脚输出使能。
RDIMMIOM	[27]	RW,0x1	RDIMM IO (QCSEN# and MIRROR) 模式选择: 为 0 时, 选择 SSTL, 为 1 时选择 CMOS。
RDIMMPDR	[26]	RW,0x1	RDIMM IO (QCSEN# and MIRROR) 接收 PowerDown。
RDIMMPDD	[25]	RW,0x0	RDIMM IO (QCSEN# and MIRROR) 驱动 PowerDown。
RDIMMODT	[24]	RW,0x0	RDIMM IO (QCSEN# and MIRROR) 端接使能。
ERROUTOE	[23]	RW,0x0	ERROUT#引脚输出使能。
ERROUTIOM	[22]	RW,0x1	ERROUT# IO 模式选择: 为 0 时, 选择 SSTL, 为 1 时选择 CMOS。
ERROUTPDR	[21]	RW,0x0	ERROUT#接收 PowerDown。
ERROUTPDD	[20]	RW,0x1	ERROUT#驱动 PowerDown。
ERROUTODT	[19]	RW,0x0	ERROUT#端接电阻使能。
PARINOE	[18]	RW,0x1	PAR_IN 引脚输出使能。
PARINIOM	[17]	RW,0x0	PAR_IN IO 模式选择: 为 0 时, 选择 SSTL, 为 1 时选择 CMOS。
PARINPDR	[16]	RW,0x1	PAR_IN 接收 PowerDown。
PARINPDD	[15]	RW,0x0	PAR_IN 驱动 PowerDown。
PARINODT	[14]	RW,0x0	PAR_IN 端接电阻使能。

SOPERR	[2]	RW,0x0	校验错停止使能：当被设置时，当遇到校验错错误时，PUB 将停止发送 DRAM 的指令，直到 PIR.CLRSR 状态被清除。
ERRNOREG	[1]	RW,0x0	不记录校验错错误使能：当被设置时，来自于 RDIMM 的校验错 (ERROUT#) 将直接传递给 DFI 的控制器（没有同步或缓冲），当不被设置时，错误将会被同步（SDRAM 时钟到 Ctl_clk）。
RDIMM	[0]	RW,0x0	RDIMM 使能。如果被设置，说明 RDIMM 被使用。在这种情况下，PUB 将 SDRAM 的读写延迟加 1。这仅仅应用于 PUB 内嵌的 SDRAM 操作。由控制器（存控）产生 SDRAM 操作的延时必须满足 RDIMM 要求（不受该配置影响）。当使用标准的 RDIMM 缓冲芯片时 DCR.NOSRA 寄存器位必须被设置为 1（不能同时驱动两个 Rank）。
—	其它	—	保留。

表 2-106: RDIMMGCR1 寄存器域的描述

名称	范围	类型	描述
CRINIT	[31:16]	RW, 0xffff	缓冲芯片中控制寄存器初始化使能，分别对应 RC0..RC15。CRINIT[i]对应位为 1，表示该寄存器 RCi 需要被初始化。
tBCMRD	[14:12]	RW,0x0	WRC 命令间隔。使用 SDRAM 时钟进行计数，该配置值加 8 为实际延时间隔（该参数与缓冲芯片 tMRD 参数一致）。
tBCSTAB	[11:0]	RW, 3200	稳定时间：使用 SDRAM 时钟进行计数（DDR3-2133, 533MHz 的 6us）。该参数和缓冲芯片 tsTAB 参数一致。
—	其它	—	保留。

注：该寄存器为申威 421 处理器新增。

2.3.22 RDIMMMCR0-1

偏移地址：0x01700,0x01780

表 2-107: RDIMMMCR0 寄存器域的描述

名称	范围	类型	描述
[31:28]	RC7	0x0	Control Word 7: 保留，用户自定义。
[27:24]	RC6	0x0	Control Word 6: 保留，用户自定义。
[23:20]	RC5	0x0	Control Word 5: 时钟驱动能力。 RC5[1:0]控制 Y1, Y1#,Y3, Y3#的时钟输出。 RC5[3:2]控制 Y0, Y0#,Y2,Y2#的时钟输出。

			<p>2'b00: 小规模驱动(4 or 5 SRAM Load)。</p> <p>2'b01: 中规模驱动(8 or 10 SRAM Load)。</p> <p>2'b10: 大规模驱动(16 or 20 SRAM Load)。</p> <p>2'b11: 保留。</p>
[19:16]	RC4	0x0	<p>Control Word 4: 控制信号驱动能力。</p> <p>[1:0]: 表示 A 输出驱动设置。</p> <p>[3:2]: 表示 B 输出驱动设置。</p> <p>2'b00: 小规模驱动(4 or 5 SRAM Load)。</p> <p>2'b01: 中规模驱动(8 or 10 SRAM Load)。</p> <p>2'b10: 保留。</p> <p>2'b11: 保留。</p>
[15:12]	RC3	0x0	<p>Control Word 3: 命令地址信号驱动能力。</p> <p>[1:0]: A 输出驱动器设置</p> <p>[3:2]: B 输出驱动器设置</p> <p>2'b00: 小规模驱动(4 or 5 SRAM Load)。</p> <p>2'b01: 中规模驱动(8 or 10 SRAM Load)。</p> <p>2'b10: 大规模驱动(16 or 20 SRAM Load)。</p> <p>2'b11: 保留。</p>
[11:8]	RC2	0x0	<p>Control Word 2: 延时控制。</p> <p>[0]: 0: 标准 pre-launch 时间。1: 受 RC12 控制。</p> <p>[1]: 保留</p> <p>[2]: 0: 端接 100 欧姆。1: 端接 150 欧姆。</p> <p>[3]: 0: 控制频率域 1。1: 测试模式的频率域 2</p>
[7:4]	RC1	0x0	<p>Control Word 1: 时钟驱动使能控制。</p> <p>[0]: 0: Y0/Y0#时钟使能有效。1: Y0/Y0#使能无效。</p> <p>[1]: 0: Y1/Y1#时钟使能有效。1: Y1/Y1#使能无效。</p> <p>[2]: 0: Y2/Y2#时钟使能有效。1: Y2/Y2#使能无效。</p> <p>[3]: 0: Y3/Y3#时钟使能有效。1: Y3/Y3#使能无效。</p>
[3:0]	RC0	0x0	<p>Control Word 0: 全局控制信息。</p> <p>[0]: 0: 反向输出有效; 1: 反向输出无效。</p> <p>[1]: 0: 禁止无效输出; 1: 允许无效输出。</p> <p>[2]: 0: A 输出有效; 1: A 输出无效。</p> <p>[3]: 0: B 输出有效; 1: B 输出无效。</p>
—	其它	—	保留。

表 2-108: RDIMMCR1 寄存器域的描述

名称	范围	类型	描述
RC15	[31:28]	0x0	Control Word 15: Reserved for future use 用户自定义
RC14	[27:24]	0x0	Control Word 14: Reserved for future use 用户自定义
RC13	[23:20]	0x0	Control Word 13: Reserved for future use 用户自定义
RC12	[19:16]	0x0	Control Word 12: Reserved for future use 用户自定义
RC11	[15:12]	0x0	Control Word 11 : 电压控制。 [1:0]: 00: DDR3 1.5V 模式 01: 1.35V 模式 10: 保留 11: 保留 [3:2]: 保留
RC10	[11:8]	0x0	Control Word 10: 速率选择。 [2:0]: 000: DDR3/DDR3L-800 001: DDR3/DDR3L-1066 010: DDR3/DDR3L-1333 011: DDR3/DDR3L-1600 其它保留 [3]保留。
RC9	[7:4]	0x0	Control Word 9: 低功耗配置。 [0]: 0: 输出悬空; 1: 轻驱动有效 [1]: 保留 [2]: 0: CKE 断电, IBT 开启, QxODT 正常功能。 1: CKE 断电, IBT 关闭, QxODT 保持低电平。 仅当[3]被设置为 1 时[2]才能有效。 [3]: 0: CKE 断电模式无效; 1: CKE 断电模式有效
RC8	[3:0]	0x0	Control Word 8: 附加总线端接电阻设置。 [2:0]: 000: IBT 定义为 RC2。 001: 保留。 010: 200 欧姆。 011: 保留。 100: 300 欧姆。 111: 关闭。

			其它保留。 [3]: 0: 当 MIRROR 为高电平时关闭 IBT。 1: 当 MIRROR 为高电平时开启 IBT。
—	其它	—	保留。

注：该寄存器为申威 421 处理器新增。

2.3.23 DCUAR

偏移地址：0x01800

DCUAR 为 DCU 地址寄存器。

表 2-109: DCUAR 寄存器域的描述

名称	范围	类型	描述
ATYPE	[11]	RW,0x0	访问类型。 0=写访问; 1=读访问。
INCA	[10]	RW,0x0	地址加 1。置“1”时，每次访问 Cache 后，WADDR 和 SADDR 的 cache 地址自动加 1。
CSEL	[9:8]	RW,0x0	Cache 选择：选择被访问的 Cache。 2'b00 = 命令 cache; 2'b01 = 期望值数据 cache; 2'b10 = 读数据 cache; 2'b11 = 保留。
CSADDR	[7:4]	RW,0x0	Cache slice 地址：本次访问的 Cache Slice 的地址，对应行内的 Slice 地址；读写接口都是 32 位，如果 Cache 行宽度大于 32 位，则需要通过 Slice 地址进行访问。
CWADDR	[3:0]	RW,0x0	Cache WORD 地址：本次访问的 cache word 的地址，对应 Cache 的行地址。
—	其它	—	保留。

2.3.24 DCUDR

偏移地址：0x01880

DCUDR 为 DCU 数据寄存器。

表 2-110: DCUDR 寄存器域的描述

名称	范围	类型	描述
CDATA	[31:0]	RW,0x0	Cache 数据，从 Cache 读出或写入的数据接口。

2.3.25 DCURR

偏移地址：0x01900

DCURR 为 DCU 运行寄存器。

表 2-111: DCURR 寄存器域的描述

名称	范围	类型	描述
XCEN	[23]	RW,0x0	期望值比较使能。置“1”时，从 SDRAM 读出的数据将与期望值比较；
RCEN	[22]	RW,0x0	读 Cache 使能。置“1”时，从 SDRAM 读出的数据被写入到 Read Data Cache 中；
SCOF	[21]	RW,0x0	读数据采样满后停止标志。置“1”表示对读出数据进行采样时，用于存放采样数据的 Cache 已满，则停止采样。
SNOF	[20]	RW,0x0	在 N 次错误后停止运行标志。该标志与 NFAIL 一起使用。
NFAIL	[19:12]	RW,0x0	失败次数。在执行命令过程中，发生错误的次数达到规定值，且 SNOF 寄存器=1 则停止运行。
EADDR	[11:8]	RW,0x0	结束地址。执行命令 Cache 中命令的结束地址
SADDR	[7:4]	RW,0x0	启动地址。执行命令 Cache 中命令的起始地址
DINST	[3:0]	RW,0x0	DCU 指令。 4'b0000=空操作； 4'b0001=开始执行命令 Cache 中指令； 4'b0010=停止执行命令 Cache 中命令； 4'b0011=停止循环执行命令 Cache； 4'b0100=复位所有 DCU 中运行计时器； 4'b0101~4'b 1111 保留。
—	其它	—	保留。

2.3.26 DCULR

偏移地址：0x01980

DCULR 为 DCU 循环控制寄存器。

表 2-112: DCULR 寄存器域的描述

名称	范围	类型	描述
XLEADDR	[31:28]	RW,0xf	期望数据循环结束地址。
IDA	[17]	RW,0x0	DRAM 地址加 1。每个读/写命令执行后，DRAM 地址加 1。
LINF	[16]	RW,0x0	无限循环。置“1”将循环执行 Cache 中的命令，直到被 STOP 命令停止，否则执行 LCNT 指定的循环次数。
LCNT	[15:8]	RW,0x0	LOOP 次数。如果 LINF 寄存器不为“1”，循环执行指定的次数。
LEADDR	[7:4]	RW,0x0	LOOP 结束地址，指定循环执行的命令 Cache 结束地址。
LSADDR	[3:0]	RW,0x0	LOOP 开始地址，指定循环执行的命令 Cache 开始地址。
—	其它	—	保留。

2.3.27 DCUGCR

偏移地址：0x01a00

DCUGCR 为 DCU 通用配置寄存器。

表 2-113：DCUGCR 寄存器域的描述

名称	范围	类型	描述
RCSW	[15:0]	RW,0x0	读出数据比较启动。在连续读出 RCSW 个数据后，启动对读出数据的比较。
—	其它	—	保留。

2.3.28 DCUTPR

偏移地址：0x01a80

DCUTPR 为 DCU 延时参数寄存器。

表 2-114：DCUTPR 寄存器域的描述

名称	范围	类型	描述
tDCUT3	[31:24]	RW,0x0	DCU 通用时间参数 3。
tDCUT2	[23:16]	RW,0x0	DCU 通用时间参数 2。
tDCUT1	[15:8]	RW,0x0	DCU 通用时间参数 1。
tDCUT0	[7:0]	RW,0x0	DCU 通用时间参数 0。

2.3.29 DCUSR0-1

偏移地址：0x01b00,0x01b80

DCUSR 为 DCU 状态寄存器。

表 2-115: DCUSR0 寄存器域的描述

名称	范围	类型	描述
CFULL	[2]	RO,0x0	Capture 满, 为“1”表示 capture cache 满。
CFAIL	[1]	RO,0x0	Capture 失败, 为“1”表示至少有一个读出数据有错。
RDONE	[0]	RO,0x0	执行结束标志, 在成功执行了 STOP 后, 将置起该标志。
—	其它	—	保留。

表 2-116: DCUSR1 寄存器域的描述

名称	范围	类型	描述
LPCNT	[31:24]	RO,0x0	循环次数: 已经循环执行命令的次数。主要用于因出错停止执行的情况。
FLCNT	[23:16]	RO,0x0	失败次数: 读出失败的次数。
RDCNT	[15:0]	RO,0x0	读计数器: 从 SDRAM 读出次数。

2.3.30 BISTR

偏移地址：0x02000

BISTR 为 BIST 运行寄存器。

表 2-117: BISTR 寄存器域的描述

名称	范围	类型	描述
BCCSEL	[26:25]	RW,0x0	BIST 时钟周期选择。AC 的 LOOPBACK 测试时, 从返回四个节拍中选择哪个写入 FIFO 中。 2'b00 = Clock cycle 0, 4, 8, 12 等; 2'b01 = Clock cycle 1, 5, 9, 13 等; 2'b10 = Clock cycle 2, 6, 10, 14 等; 2'b11 = Clock cycle 3, 7, 11, 15 等。

BCKSEL	[24:23]	RW,0x0	<p>BIST CK 选择。AC 的 LOOPBACK 测试时，从三个 CK 中选择哪个进行测试。取值范围如下：</p> <p>2'b00 = CK[0]; 2'b01 = CK[1]; 2'b10 = CK[2]; 2'b11 = 保留。</p>
BDXSEL	[22:19]	RW,0x0	<p>DATX8 的 BIST 测试使能时有意义，选择哪个 Datx8 进行 BIST 测试，有效取值范围 0~8，对应于 9 个 DatX8。</p>
BDPAT	[18:17]	RW,0x0	<p>BIST 测试数据模式选择：</p> <p>2'b00 = Walking 0; 2'b01 = Walking 1; 2'b10 = LFSR-based pseudo-random; 2'b11 = 用户自定义。</p>
BDMEN	[16]	RW,0x0	<p>BIST Data MASK 使能：该位为“1”表示 BIST 测试包括 DM 测试。</p>
BACEN	[15]	RW,0x0	<p>AC 的 BIST 测试使能。该位和 BDXEN 是互斥的，不能同时置“1”。</p>
BDXEN	[14]	RW,0x0	<p>DATX8 的 BIST 测试使能。该位和 BACEN 是互斥的，不能同时置“1”。</p>
BSONF	[13]	RW,0x0	<p>BIST 测试发生 N 次报错时（次数由 NFAIL 指定），则停止执行。</p>
NFAIL	[12:5]	RW,0x0	<p>失败次数。如果 BSBOF 寄存器置“1”时，执行命令并发现读出数据出错次数达到 NFAIL+1 次，则停止测试。</p>
BINF	[4]	RW,0x0	<p>BIST 无限执行。置“1”表示无限执行 BIST 测试，直到被停止或遇到读出错误（满足退出条件）。置“0”表示按 BISTWCR[BWCNT]寄存器指定次数执行。</p>
BMODE	[3]	RW,0x0	<p>BIST 模式：</p> <p>0=自环模式，地址、命令、数据 IO 上环回，可以选择从 Pad 端或 Core 端换回； 1=DRAM 模式，以正常访存方式。</p>

BINST	[2:0]	RW,0x0	BIST 指令，选择 BIST 测试指令。 3'b000 =空操作； 3'b 001 =启动 BIST 测试； 3'b010 =停止 BIST 测试； 3'b011 =清除所有运行相关的状态寄存器； 3'b100 ~3'b111 保留。
—	其它	—	保留。

2.3.31 BISTWCR

偏移地址：0x2080

BISTWCR 为 BIST 字计数寄存器。

表 2-118: BISTWCR 寄存器域的描述

名称	范围	类型	描述
BWCNT	[15:0]	RW, 0X20	BIST Word 计数：BIST 生成数据计数。这个值应给是突发长度除以 2 的倍数。例如，BL=8，则有效取值是 4、8、12、16、……。
—	其它	—	保留。

2.3.32 BISTMSKR0-2

偏移地址：0x02100,0x02180,0x02200

BISTMSKR 为 BIST 屏蔽寄存器。

表 2-119: BISTMSKR0 寄存器域的描述

名称	范围	类型	描述
ODTMSK	[31:28]	RW,0x0	4 个 ODT 位屏蔽指示。
CSMSK	[27:24]	RW,0x0	4 个 CS_N 位屏蔽指示。
CKEMSK	[23:20]	RW,0x0	4 个 CKE 位屏蔽指示。
WEMSK	[19]	RW,0x0	WE_N 位屏蔽指示。
BAMSK	[18:16]	RW,0x0	3 位 BANK 地址屏蔽指示。
AMSK	[15:0]	RW,0x0	16 位地址屏蔽指示。

表 2-120: BISTMSKR1 寄存器域的描述

名称	范围	类型	描述
----	----	----	----

DMMSK	[31:28]	RW,0x0	DM 位屏蔽指示。
PARMSK	[27]	RW,0x0	PAR_IN 屏蔽指示。仅适用与支持校验的 DIMM。 Mask bit for the PAR_IN. Only for DIMM parity support and only if the design is compiled for less than 3 ranks.
CASMSK	[1]	RW,0x0	CAS_N 位屏蔽指示。
RASMSK	[0]	RW,0x0	RAS_N 位屏蔽指示。
—	其它	—	保留。

表 2-121: BISTMSKR2 寄存器域的描述

名称	范围	类型	描述
DQMSK	[31:0]	RW,0x0	DQ位屏蔽指示。

2.3.33 BISTLSR

偏移地址: 0x02280

BISTLSR 是 BIST 随机测试种子寄存器。

表 2-122: BISTLSR 寄存器域的描述

名称	范围	类型	描述
SEED	[31:0]	RW, 0X1234ABCD	随机bist测试数据时的种子 (LFSR)。

2.3.34 BISTAR0-2

偏移地址: 0x2300,0x02380,0x02400

BISTAR 是 BIST 地址寄存器。

表 2-123: BISTAR0 寄存器域的描述

名称	范围	类型	描述
BBANK	[30:28]	RW,0x0	BIST Bank 地址: 为 BIST 选择 SDRAM 的 BANK 地址。
BROW	[27:12]	RW,0x0	BIST 行地址: 为 BIST 选择 SDRAM 的行地址。
BCOL	[11:0]	RW,0x0	BIST 列地址: 为 BIST 选择 SDRAM 的列地址。
—	其它	—	保留。

表 2-124: BISTAR1 寄存器域的描述

名称	范围	类型	描述
BAINC	[15:4]	RW,0x0	BIST 地址增量: 为每个读/写访问指定地址的增量值。该值要满

			足对界要求。
BMRANK	[3:2]	RW,0x3	BIST 最大 RANK 数：为 BIST 指定最大 SDRAM 的 RANK 数。 0: 1 个 Rank; 1: 2 个 Rank; 2: 3 个 Rank; 3: 4 个 Rank。
BRANK	[1:0]	RW,0x0	BIST RANK 号：为 BIST 指定 RANK 号。 0: Rank 0; 1: Rank 1; 2: Rank 2; 3: Rank 3。
—	其它	—	保留。

表 2-125: BISTAR2 寄存器域的描述

名称	范围	类型	描述
BMBANK	[30:28]	RW, 0x7	BIST 最大 Bank 地址：为 BIST 指定 SDRAM 最大的 BANK 地址。
BMROW	[27:12]	RW, 0xFFFF	BIST 最大行地址：为 BIST 指定 SDRAM 的最大行地址。
BMCOL	[11:0]	RW,0xFFF	BIST 最大列地址：为 BIST 指定 SDRAM 的最大列地址。
—	其它	—	保留。

2.3.35 BISTUDPR

偏移地址：0x02480

BISTUDPR 是 BIST 用户制定测试向量寄存器。

表 2-126: BISTUDPR 寄存器域的描述

名称	范围	类型	描述
BUDP1	[31:16]	RW, 0Xffff	BIST 数据模式 1: 在 BIST 测试时, 该数据依次被发送到奇数的 DQ 引脚上。
BUDP0	[15:0]	RW, 0X0000	BIST 数据模式 0: 在 BIST 测试时, 该数据依次被发送到偶数的 DQ 引脚上。

2.3.36 BISTGSR

偏移地址：0x02500

BISTGSR 是 BIST 通用状态寄存器。

表 2-127: BISTGSR 寄存器域的描述

名称	范围	类型	描述
CASBER	[31:30]	RO,0x0	CAS 位错误：记录 CAS 出错的次数。
RASBER	[29:28]	RO,0x0	RAS 位错误：记录 RAS 出错的次数。
DMBER	[27:20]	RO,0x0	DM 位错错误：Data Mask 位错误次数。 DMBER[1:0]：对应 Beat 0； DMBER[3:2]：对应 Beat 1； DMBER[5:4]：对应 Beat 2； DMBER[7:6]：对应 Beat 3。
PARBER	[17:16]	RO,0x0	PAR_IN Bit Error (DIMM Only): Indicates the number of bit errors on PAR_IN. PAR_IN 错误位：显示 PAR_IN 错误的位数。仅已适用于支持校验的 DIMM。
BDXERR	[2]	RO,0x0	BIST 数据错：对数据通道测试时，出现结果比较不等，则报错。
BACERR	[1]	RO,0x0	BIST 地址/命令错：对地址/命令通道测试时，出现结果比较不等，则报错。
BDONE	[0]	RO,0x0	BIST 测试完成标志。当重新进行 bist 测试时，该位会自动清零；
—	其它	—	保留。

2.3.37 BISTWER

偏移地址：0x02580

BISTWER 是 BIST 字错计数寄存器。

表 2-128: BISTWER 寄存器域的描述

名称	范围	类型	描述
DXWER	[31:16]	RO,0x0	BIST 测试中，DATX8 发生错误的次数。
ACWER	[15:0]	RO,0x0	BIST 测试中，AC 发生错误的次数。

2.3.38 BISTBER0-3

偏移地址：0x02600,0x002680,0x02700,0x02780

BISTBER 是位字错计数寄存器。

表 2-129: BISTBER0 寄存器域的描述

名称	范围	类型	描述
ABER	[31:0]	RO,0x0	地址 A[15:0]的各 bit 位错误计数器。每个地址线两位，例如 A[0]对应于 ABER[1:0]，A[1]对应 ABER[3:2]，依次类推。

表 2-130: BISTBER1 寄存器域的描述

名称	范围	类型	描述
ODTBER	[31:24]	RO,0x0	ODT[3:0]信号位错误计数器，每位信号对应两位。
CSBER	[23:16]	RO,0x0	CS_N[3:0]信号位错误计数器，每位信号对应两位。
CKEBER	[15:8]	RO,0x0	CKE[3:0]信号位错误计数器，每位信号对应两位。
WEBER	[7:6]	RO,0x0	WE_N 信号错误计数器。
BABER	[5:0]	RO,0x0	BANK[2:0]信号位错计数器，每位信号对应两位。

表 2-131: BISTBER2 寄存器域的描述

名称	范围	类型	描述
DQBER0	[31:0]	RO,0x0	DQS 偶数脉冲（第 0 个、第 2 个...）上数据对应的错误计数器。第 1 个 16 位对应上升沿（Beat 0、Beat 4...），第 2 个 16 位对应下降沿（Beat 1、Beat 5...）。每个 16 位对应 8 位数据，每个数据占用两位。

表 2-132: BISTBER3 寄存器域的描述

名称	范围	类型	描述
DQBER1	[31:0]	RO,0x0	DQS 奇数脉冲（第 1 个、第 3 个...）上数据对应的错误计数器。第 1 个 16 位对应上升沿（Beat 2、Beat 6...），第 2 个 16 位对应下降沿（Beat 3、Beat 7...）。每个 16 位对应 8 位数据，每个数据两位。

2.3.39 BISTWCSR

偏移地址：0x2800

BISTWCSR 是 BIST 字计数统计寄存器。

表 2-133: BISTWER 寄存器域的描述

名称	范围	类型	描述
DXWCNT	[31:16]	RO,0x0	数据通道 WORD 统计，从数据通道上接收到的数量。
ACWCNT	[15:0]	RO,0x0	地址/命令通道 WORD 统计，从 ADDR/CMD 通道上接收到的数量。

2.3.40 BISTFWR0-2

偏移地址：0x2880,0x02900,0x02980

BISTFWR 是 BIST 错误字寄存器。

表 2-134: BISTFWR0 寄存器域的描述

名称	范围	类型	描述
ODTWEBS	[31:28]	RO,0x0	4 位 ODT 信号线 WORD 错误状态。
CSWEBS	[27:24]	RO,0x0	4 位 CS# 信号线 WORD 错误状态。
CKEWEBS	[23:20]	RO,0x0	4 位 CKE 信号线 WORD 错误状态。
WEWEBS	[19]	RO,0x0	WE_N 信号线 WORD 错误状态。
BAWEBS	[18:16]	RO,0x0	3 位 Bank 地址线 WORD 错误状态。
AWEBS	[15:0]	RO,0x0	16 位地址线 WORD 错误状态。

表 2-135: BISTFWR1 寄存器域的描述

名称	范围	类型	描述
DMWEBS	[31:28]	RO,0x0	DM 信号 WORD 错误状态。DMWEBS[0]对应于第 1 个 Beat (对应 DQS 的上升沿)，DMWEBS[1]对应于第 2 个 Beat (对应 DQS 的下降沿)，以此类推共四个 Beat。
PARWEBS	[26]	RO,0x0	PAR_IN 信号 WORD 错误状态,仅适用支持校验的 RDIMM。
CASWEBS	[1]	RO,0x0	CAS_N 信号线 WORD 错误状态。
RASWEBS	[0]	RO,0x0	RAS_N 信号线 WORD 错误状态。
—	其它	—	保留。

表 2-136: BISTFWR2 寄存器域的描述

名称	范围	类型	描述
DQWEBS	[31:0]	RO,0x0	通道内8位数据线WORD错误状态。第1个8位对应第1个 Beat (对应DQS的上升沿)，第2个8位对应第2个Beat (对应DQS的下降沿)，依次类推共4个Beat。8位之内每一位对应一位数据。

2.3.41 AACR

偏移地址: 0x02e80

表 2-137: AACR 寄存器域的描述

名称	范围	类型	描述
AAOENC	[31]	RW, 0x0	Anti-Aging PAD Output Enable Control: Enables, if set, anti-aging toggling on the pad output enable signal “ctl_oe_n” going into the DATX8s. This will increase power consumption for the anti-aging feature. Anti-Aging PAD 输出使能控制: 当有效时, 有效使能 ctl_oe_n 进入 DATX8。这将会增加功耗。
AAENC	[30]	RW, 0x0	Anti-Aging Enable Control: Enables, if set, the automatic toggling of the data going to the DATX8 when the data channel from the controller/PUB to DATX8 is idle for programmable number of clock cycles. Anti-Aging 使能控制: 当有效时, Datx8 处于空闲一段时间 (可编程) 后启动 Anti-Aging。
AATR	[29:0]	RW, 0xff	Anti-Aging Toggle Rate: Defines the number of controller clock (ctl_clk) cycles after which the PUB will toggle the data going to DATX8 if the data channel between the controller/PUB and DATX8 has been idle for this long. The default value correspond to a toggling count of 4096 ctl_clk cycles. For a ctl_clk running at 533MHz the toggle rate will be approximately 7.68us. The default value may also be overridden by the macro DWC_AACR_AATR_DFLT. Anti-Aging 空闲时间阈值: 默认值为 4096 个 Ctl_clk 时钟 (533MHz, 约为 7.68us)。默认值也有可能通过宏 DWC_AACR_AATR_DFLT 来定义。

注: 该寄存器为申威 421 处理器新增。

2.3.42 ZQnCR0-1

偏移地址: ZQ0CR0-1:0x03000,0x03080

ZQ1CR0-1:0x03200,0x03280

ZQ2CR0-1:0x03400,0x03480

ZQnCR 是 ZQ 控制寄存器。

表 2-138: ZQnCR0 寄存器域的描述

名称	范围	类型	描述
ZQPD	[31]	RW,0x0	ZQ PowerDown: 置“1”表示 PZQ 单元被 Power Down。
ZCALEN	[30]	RW,0x1	阻抗校准使能, 置“1”将通过 PIR 寄存器的 ZCAL 或 DFI 接口触发 ZQ 控制模块进行阻抗校准。
ZCALBYP	[29]	RW,0x0	旁路阻抗校准, 置“1”表示旁路阻抗校准。ZCALBYP 优先级高于 ZCALEN。
ZDEN	[28]	RW,0x0	阻抗数据使能: 置“1”表示直接使用 ZDATA 数据域控制阻抗。
ZDATA	[27:0]	RW, 0x14A	阻抗数据, 该数据直接驱动阻抗控制。 ZDATA[27:21]用于选择片上端接上拉阻抗; ZDATA[20:14]用于选择片上端接下拉阻抗; ZDATA[13:7]用于输出阻抗上拉阻抗; ZDATA[6:0]用于输出阻抗下拉阻抗。 片上端接控制如下: 0x0a: 120 欧姆; 0x0f: 80 欧姆; 0x14: 60 欧姆; 0x1e: 40 欧姆; 0x28: 30 欧姆; 输出阻抗控制如下: 0x2d: 45 欧姆; 0x32: 40 欧姆; 0x3b: 34 欧姆。

表 2-139: ZQnCR1 寄存器域的描述

名称	范围	类型	描述
DFIPU1	[17]	RW,0x0	DFI PHY发起Update (接口0) I/O阻抗校准使能。仅在AC共享模式下有意义。
DFIPU0	[16]	RW,0x0	DFI PHY发起Update (接口0) I/O阻抗校准使能。
DFICCU	[14]	RW,0x0	DFI控制器发起Update (接口0、接口1) 同时使能。仅在AC共享的模式下。
DFICU1	[13]	RW,0x0	DFI控制器发起Update (接口0) I/O阻抗校准使能。仅在

			AC共享模式下有意义。
DFICU0	[12]	RW,0x1	DFI控制器发起Update（接口0）I/O阻抗校准使能。
ZPROG	[7:0]	RW,0x7B	通过外部端接高精度 240 欧姆电阻比率来设置输出阻抗和片上端接，具体如下： ZPROG[7:4]对应片上端接电阻。 2: 120 欧姆，即外接电阻的 1/2； 5: 60 欧姆，即外接电阻的 1/4； 8: 40 欧姆，即外接电阻的 1/6。 ZPROG[3:0]对应输出阻抗。 11: 40 欧姆，即外接电阻的 1/6； 13: 34 欧姆，即外接电阻的 1/7。
—	其它	—	保留。

该寄存器相对于申威 411 处理器有修改。

2.3.43 ZQnSR0-1

偏移地址：ZQ0SR0-1:0x03100,0x03180

ZQ1SR0-1:0x03300,0x03380

ZQ2SR0-1:0x03500,0x03580

ZQnSR 是 ZQ 状态寄存器。

表 2-140: ZQnSR0 寄存器域的描述

名称	范围	类型	描述
ZDONE	[31]	RO,0x0	阻抗校准结束，为“1”表示已经完成了阻抗校准。
ZERR	[30]	RO,0x0	阻抗校准错，为“1”表示在阻抗校准过程中发生错误。
ZCTRL	[27:0]	RO,0x1 4A	阻抗控制：阻抗控制的当前数值。 ZDATA[27:21]用于选择片上端接上拉阻抗； ZDATA[20:14]用于选择片上端接下拉阻抗； ZDATA[13:7]用于输出阻抗上拉阻抗； ZDATA[6:0]用于输出阻抗下拉阻抗。
—	其它	—	保留。

表 2-141: ZQnSR1 寄存器域的描述

名称	范围	类型	描述
OPU	[7:6]	RO,0x0	片上端接上拉端接校准状态，各值含义如下： 2'b00=没有错；

			2'b01=上溢错; 2'b10=下溢错; 2'b11=在校准处理中。
OPD	[5:4]	RO,0x0	片上端接下拉端接校准状态, 各值含义如下: 2'b00=没有错; 2'b01=上溢错; 2'b10=下溢错; 2'b11=在校准处理中。
ZPU	[3:2]	RO,0x0	输出阻抗上拉校准状态, 各值含义如下: 2'b00=没有错; 2'b01=上溢错; 2'b10=下溢错; 2'b11=在校准处理中。
ZPD	[1:0]	RO,0x0	输出阻抗下拉校准状态, 各值含义如下: 2'b00=没有错; 2'b01=上溢错; 2'b10=下溢错; 2'b11=在校准处理中。
—	其它	—	保留。

2.3.44 DXnGCR

偏移地址: DX0GCR:0x03800

DX1GCR:0x04000

DX2GCR:0x04800

DX3GCR:0x05000

DX4GCR::0x05800

DX5GCR:0x06000

DX6GCR:0x06800

DX7GCR:0x07000

DX8GCR:0x07800

DXnGCR 是 Datx8 的通用控制寄存器。

表 2-142: DXnGCR 寄存器域的描述

名称	范围	类型	描述
----	----	----	----

CALBYP	[31]	RW,0x0	旁路校准：置“1”表示在 Phy 初始化后关闭自动触发校准。
MDLEN	[30]	RW,0x1	主延迟线（MDL）使能：置“1”表示在初始周期测量后，开启 DATX8 的主延迟线周期校准。 该配置将与 DXCCR 寄存器的相关配置信息（MDLEN）进行共同控制（“或”操作）。
WLRKEN	[29:26]	RW,0xF	Rank Write Leveling 使能。指定 RANK 进行 Write Leveling 处理。对于没有指定的 RANK，Write Leveling 的结果被忽略，并直接标识为完成。 WLRKEN[0]对应 Rank0； WLRKEN[1]对应 Rank1； WLRKEN[2]对应 Rank2； WLRKEN[3]对应 Rank3。
PLLBYB	[19]	RW,0x0	PLL 旁路：通过驱动旁路端口使 PLL 进入旁路模式。此位不会自动清除，当不使用旁路时必须写入 0。这位与 PLLCR.BYP 共用起作用（或操作）。
GSHIFT	[18]	RW,0x0	PLL 变速使能。置“1”时，PLL 处于快速 Lock 模式。 该配置将与 PLLCR 寄存器的相关配置信息（GSHIFT）进行共同控制（“或”操作）。
PLLPD	[17]	RW,0x0	PLL PowerDown：通过 PLL PowerDown 端口将 PLL 设置成 PowerDown 模式。该寄存器没有自清功能，必须软件清除。 该配置将与 PLLCR 寄存器的相关配置信息（PLLPD）进行共同控制（“或”操作）。
PLLRST	[16]	RW,0x0	PLL 复位。通过 PLL 复位端口对 DatX8 的 PLL 复位。该寄存器没有自清功能，必须软件清除。 该配置将与 PLLCR 寄存器的相关配置信息（PLLRST）进行共同控制（“或”操作）。
DXOEO	[15:14]	RW,0x0	数据输出强制使能： 2'b00：输出使能由 DFI 控制。 2'b01：输出使能强迫有效 2'b10：输出使能强迫无效 2'b11：保留
RTTOAL	[13]	RW,0x0	ODT 开启附加延迟，表示在 Read 期间，何时开启 ODT（DQS 根据 DQSODT、DQ/DM 根据 DQODT 配置）。 0=在 Read 数据前导之前 2 个周期开启 ODT；

			1=在 Read 数据前导之前 1 个周期开启 ODT。
RTTOH	[12:11]	RW,0x1	ODT 输出保持时间： 表示在读数据后导之后，ODT 仍维持 0~3 个周期（DQS 根据 DQSODT、DQ/DM 根据 DQODT 配置）。如果是动态 ODT 方式，RTTOH 之后会关闭 ODT。
DQRTT	[10]	RW,0x1	DQ 动态 RTT 控制。 1=表示 DQ/DM 的 ODT 在 Read 时根据 DQODT 动态控制，其它时候不使能； 0=表示 DQ/DM 的 ODT 始终根据 DQODT 控制。
DQSRTT	[9]	RW,0x1	DQS 动态 ODT 控制。 1=表示 DQS 的 ODT 在 Read 时根据 DQSODT 动态控制，其它时候不使能； 0=表示 DQS 的 ODT 始终根据 DQSODT 控制。
DSEN	[8:7]	RW,0x1	写 DQS 使能：控制到 SDRAM 的写 DQS 是否使能或者 DQS 反相。有效取值范围如下： 2'b00=DQS 始终驱动为低电平； 2'b01=DQS 极性反相； 2'b10=DQS 正常工作； 2'b11=DQS 始终驱动为高电平。 这些值只有在 DQS/DQS_N 输出使能时才有效，否则 DQS/DQS_N 为三态。
DQSRPD	[6]	RW,0x0	DQSR 单元 PowerDown：置“1”表示 PDQSR 单元设置成 PowerDown。 该配置将与 DXCCR 寄存器的相关配置信息（DXPDR）进行共同控制（“或”操作）。
DXPDR	[5]	RW,0x0	DQS/DQ/DM 接收 PowerDown：置“1”表示 DQ、DM 和 DQS/DQS_N 的输入接收器设置成 PowerDown。 该配置将与 DXCCR 寄存器的相关配置信息（DXPDR）进行共同控制（“或”操作）。
DXPDD	[4]	RW,0x0	DQS/DQ/DM 发送 PowerDown：置“1”表示 DQ、DM 和 DQS/DQS_N 的输出驱动器设置成 PowerDown。 该配置将与 DXCCR 寄存器的相关配置信息（DXPDD）进行共同控制（“或”操作）。
DXIOM	[3]	RW,0x0	DQS/DQ/DM IO 模式选择：

			0=选择 SSTL 模式； 1=选择 CMOS 模式。 该配置将与 DXCCR 寄存器的相关配置信息（DXIOM）进行共同控制（“或”操作）。
DQODT	[2]	RW,0x0	DQ 片上端接：置“1”表示打开 DQ、DM 信号 SSTL IO 的片上端接。该配置将与 DXCCR 寄存器的相关配置信息（DXODT）进行共同控制（“或”操作）。
DQSODT	[1]	RW,0x0	DQS 片上端接：置“1”表示打开 DQS/DQS_N 信号 SSTL IO 的片上端接。该配置将与 DXCCR 寄存器的相关配置信息（DXODT）进行共同控制（“或”操作）。
DXEN	[0]	RW,0x1	数据通道使能： 1=打开该数据通道； 0=关闭通道，此时初始化、数据训练和读写操作均被屏蔽。
—	其它	—	保留。

2.3.45 DXnGSR0-2

偏移地址：

DX0GSR0-1:0x03880,0x03900 DX0GSR2:0x03e80
 DX1GSR0-1:0x04080;0x04100 DX1GSR2:0x04680
 DX2GSR0-1:0x04880,0x04900 DX2GSR2:0x04e80
 DX3GSR0-1:0x05080,0x05100 DX3GSR2:0x05680
 DX4GSR0-1:0x05880,0x05900 DX4GSR2:0x05e80
 DX5GSR0-1:0x06080,0x06100 DX5GSR2:0x06680
 DX6GSR0-1:0x06880,0x06900 DX6GSR2:0x06e80
 DX7GSR0-1:0x07080,0x07100 DX7GSR2:0x07680
 DX8GSR0-1:0x07880,0x07900 DX7GSR2:0x07e80

DXnGSR 是 Datx8 的通用状态寄存器。

表 2-143: DXnGSR0 寄存器域的描述

名称	范围	类型	描述
WLDQ	[28]	RO.0x0	Write Leveling DQ Status: Captures the write leveling DQ status from the DRAM during software write leveling. Write Leveling DQ 状态: 当软件进行 Write Leveling 时从 DQ 上采样到的值。

QSGERR	[27:24]	RO,0x0	DQS Gating 训练错: 为“1”表示在 DQS Gate 训练时发生错误, 每个比特位对应一个 RANK。
GDQSPRD	[23:16]	RO,0x0	DQS Gate Read period: 读 DQS Gate 的 LCDL 延迟线测得的 DDR 时钟周期数, 这个值需要进行 PVT 校准。
DPLOCK	[15]	RO,0x0	DATX8 PLL LOCK: 为“1”表示 DATX8 的 PLL 已经 LOCK。
WLPRD	[14:7]	RO,0x0	Write Leveling Period: Write Leveling LCDL 延迟线测得的 DDR 周期数, 这个值需要进行 PVT 校准。
WLERR	[6]	RO,0x0	Write Leveling Error: 为“1”表示 DATX8 在 Write Leveling 中报错。
WLDONE	[5]	RO,0x0	Write Leveling Done: 为“1”表示 DATX8 已经完成 Write Leveling。
WLCAL	[4]	RO,0x0	Write Leveling 校准: 为“1”表示 DATX8 已经为 Write Leveling LCDL (DXnLCDLR1[RxWLD]) 完成周期测量。
GDQSCAL	[3]	RO,0x0	DQS Gate Read 校准: 为“1”表示 DATX8 已经为 DQS gate 读通路 LCDL (DXnLCDLR1[RxDQSGD]) 完成周期测量。
RDQSNCAL	[2]	RO,0x0	DQS_N 读校准: 为“1”表示 DATX8 已经为 DQS_N 读通路 LCDL (DXnLCDLR1[RDQSND]) 完成周期测量。
RDQSCAL	[1]	RO,0x0	DQS 读校准: 为“1”表示 DATX8 已经为 DQS 读通路 LCDL (DXnLCDLR1[RDQSD]) 完成周期测量。
WDQCAL	[0]	RO,0x0	DQ 写校准: 为“1”表示 DATX8 已经为 DQ 写通路 LCDL (DXnLCDLR1[WDQD]) 完成周期测量。
—	其它	—	保留。

表 2-144: DXnGSR1 寄存器域的描述

名称	范围	类型	描述
DLTCODE	[24:1]	RO,0x0	延迟线测试结果, 返回 DATX8 延迟线测量 (DLL 环振测试) 得到的周期数。
DLTDONE	[0]	RO,0x0	延迟线测试结束, 表示已经完成了 DATX8 的延迟线测量 (DLL 环振测试)。
—	其它	—	保留。

表 2-145: DXnGSR2 寄存器域的描述

名称	范围	类型	描述
ESTAT	[11:8]	RO,0x0	错误码。

WEWN	[7]	RO,0x0	Write Data Eye Training Warning。
WEERR	[6]	RO,0x0	Write Data Eye Training Error。
REWN	[5]	RO,0x0	Read Data Eye Training Warning。
REERR	[4]	RO,0x0	Read Data Eye Training Error。
WDWN	[3]	RO,0x0	Write Bit Deskew Warning。
WDERR	[2]	RO,0x0	Write Bit Deskew Error。
RDWN	[1]	RO,0x0	Read Bit Deskew Warning。
RDERR	[0]	RO,0x0	Read Bit Deskew Error。
—	其它	—	保留。

2.3.46DXnBDLR0-4

偏移地址:

DX0BDLR0-4:0x3980,0x3a00,0x3a80,0x3b00,0x3b80

DX1BDLR0-4:0x4180,0x4200,,4280,0x4300,0x4380

DX2BDLR0-4:0x4980,0x4a00,0x4a80,0x4b00,0x4b80

DX3BDLR0-4:0x5180,0x5200,0x5280,0x5300,0x5380

DX4BDLR0-4:0x5980,0x5a00,0x5a80,0x5b00,0x5b80

DX5BDLR0-4:0x6180,0x6200,0x6280,0x6300,0x6380

DX6BDLR0-4:0x6980,0x6a00,0x6a80,0x6b00,0x6b80

DX7BDLR0-4:0x7180,0x7200,0x7280,0x7300,0x7380

DX8BDLR0-4:0x7980,0x7a00,0x7a80,0x7b00,0x7b80

DXnBDLR 是 Datx8 的位延时的匹配寄存器。

表 2-146: DXnBDLR0 寄存器域的描述

名称	范围	类型	描述
DQ4WBD	[29:24]	RW,0x0	DQ4 写延迟: DQ4 写通路 BDL 延迟值。
DQ3WBD	[23:18]	RW,0x0	DQ3 写延迟: DQ3 写通路 BDL 延迟值。
DQ2WBD	[17:12]	RW,0x0	DQ2 写延迟: DQ2 写通路 BDL 延迟值。
DQ1WBD	[11:6]	RW,0x0	DQ1 写延迟: DQ1 写通路 BDL 延迟值。
DQ0WBD	[5:0]	RW,0x0	DQ0 写延迟: DQ0 写通路 BDL 延迟值。
—	其它	—	保留。

表 2-147: DXnBDLR1 寄存器域的描述

名称	范围	类型	描述
----	----	----	----

DQSWBD	[29:24]	RW,0x0	写 DQS 延迟：写 DQS 通路 BDL 延迟值。
DMWBD	[23:18]	RW,0x0	DM 写延迟：DM 写通路 BDL 延迟值。
DQ7WBD	[17:12]	RW,0x0	DQ7 写延迟：DQ7 写通路 BDL 延迟值。
DQ6WBD	[11:6]	RW,0x0	DQ6 写延迟：DQ6 写通路 BDL 延迟值。
DQ5WBD	[5:0]	RW,0x0	DQ5 写延迟：DQ5 写通路 BDL 延迟值。
—	其它	—	保留。

表 2-148: DXnBDLR2 寄存器域的描述

名称	范围	类型	描述
DSNRBD	[23:18]	RW,0x0	读 DQS_N 延迟：读 DQS_N 通路 BDL 延迟值。
DSRBD	[17:12]	RW,0x0	读 DQS 延迟：读 DQS 通路 BDL 延迟值。
DQOEBD	[11:6]	RW,0x0	DQ 输出使能延迟：DQ/DM 输出使能 BDL 延迟值。
DSOEBD	[5:0]	RW,0x0	DQS 输出使能延迟：DQS 输出使能 BDL 延迟值。
—	其它	—	保留。

表 2-149: DXnBDLR3 寄存器域的描述

名称	范围	类型	描述
DQ4RBD	[29:24]	RW,0x0	DQ4 读延迟：DQ4 读通路 BDL 延迟值。
DQ3RBD	[23:18]	RW,0x0	DQ3 读延迟：DQ3 读通路 BDL 延迟值。
DQ2RBD	[17:12]	RW,0x0	DQ2 读延迟：DQ2 读通路 BDL 延迟值。
DQ1RBD	[11:6]	RW,0x0	DQ1 读延迟：DQ1 读通路 BDL 延迟值。
DQ0RBD	[5:0]	RW,0x0	DQ0 读延迟：DQ0 读通路 BDL 延迟值。
—	其它	—	保留。

表 2-150: DXnBDLR4 寄存器域的描述

名称	范围	类型	描述
DMRBD	[23:18]	RW,0x0	DM 读延迟：DM 读通路 BDL 延迟值。
DQ7RBD	[17:12]	RW,0x0	DQ7 读延迟：DQ7 读通路 BDL 延迟值。
DQ6RBD	[11:6]	RW,0x0	DQ6 读延迟：DQ6 读通路 BDL 延迟值。
DQ5RBD	[5:0]	RW,0x0	DQ5 读延迟：DQ5 读通路 BDL 延迟值。
—	其它	—	保留。

2.3.47 DXnLCDLR0-2

偏移地址：

DX0LCDLR:0x3c00,0x3c80,0x3d00

DX1LCDLR:0x4400,0x4480,0x4500

DX2LCDLR:0x4c00,0x4c80,0x4d00

DX3LCDLR:0x5400,0x5480,0x5500

DX4LCDLR:0x5c00,0x5c80,0x5d00

DX5LCDLR:0x6400,0x6480,0x6500

DX6LCDLR:0x6c00,0x6c80,0x6d00

DX7LCDLR:0x7400,0x7480,0x7500

DX8LCDLR:0x7c00,0x7c80,0x7d00

DXnLCDLR 是 Datx8 的本地主延时的匹配寄存器。

表 2-151: DXnLCDLR0 寄存器域的描述

名称	范围	类型	描述
R3WLD	[31:24]	RW,0x0	Rank3 Write Leveling 延迟: 为 RANK3 指定 LCDL Write Leveling 延迟。该寄存器与 DXnGTR[R3WLSL]共同起作用。
R2WLD	[23:16]	RW,0x0	Rank2 Write Leveling 延迟: 为 RANK2 指定 LCDL Write Leveling 延迟。该寄存器与 DXnGTR[R2WLSL]共同起作用。
R1WLD	[15:8]	RW,0x0	Rank1 Write Leveling 延迟: 为 RANK1 指定 LCDL Write Leveling 延迟。该寄存器与 DXnGTR[R1WLSL]共同起作用。
R0WLD	[7:0]	RW,0x0	Rank0 Write Leveling 延迟: 为 RANK0 指定 LCDL Write Leveling 延迟。该寄存器与 DXnGTR[R0WLSL]共同起作用。

表 2-152: DXnLCDLR1 寄存器域的描述

名称	范围	类型	描述
RDQSD	[23:16]	RW,0x0	读 DQS_N 延迟: 为读 DQS_N 指定 LCDL 延迟, 其默认值是 1600MHz (DDR3-1600 的 DDR 时钟) 的 1/2 拍。
RDQSD	[15:8]	RW,0x.	读 DQS 延迟: 为读 DQS 指定 LCDL 延迟, 其默认值是 1600MHz (DDR3-1600 的 DDR 时钟) 的 1/2 拍。
WDQD	[7:0]	RW,0x0	写数据延迟: 为写数据指定 LCDL 延迟, 其默认值是 1600MHz (DDR3-1600 的 DDR 时钟) 的 1/2 拍。
—	其它	—	保留。

表 2-153: DXnLCDLR2 寄存器域的描述

名称	范围	类型	描述
R3DQSGD	[31:24]	RW,0x0 0	RANK3 读 DQS Gating 延迟: 为 Rank3 读 DQS Gating 指定 LCDL 延迟。该寄存器与 DXnGTR[R3DGSL]共同起作用, 其默认值是 800MHz (DDR3-1600 的 SDR 时钟) 的一拍。

R2DQSGD	[23:16]	RW,0x0 0	RANK2 读 DQS Gating 延迟: 为 Rank2 读 DQS Gating 指定 LCDL 延迟。该寄存器与 DXnGTR[R2DGSL]共同起作用, 其默认值是 800MHz (DDR3-1600 的 SDR 时钟) 的一拍。
R1DQSGD	[15:8]	RW,0x0 0	RANK1 读 DQS Gating 延迟: 为 Rank1 读 DQS Gating 指定 LCDL 延迟。该寄存器与 DXnGTR[R1DGSL]共同起作用, 其默认值是 800MHz (DDR3-1600 的 SDR 时钟) 的一拍。
R0DQSGD	[7:0]	RW,0x0 0	RANK0 读 DQS Gating 延迟: 为 Rank0 读 DQS Gating 指定 LCDL 延迟。该寄存器与 DXnGTR[R0DGSL]共同起作用, 其默认值是 800MHz (DDR3-1600 的 SDR 时钟) 的一拍。

2.3.48 DXnMDLR

偏移地址:

DX0MDLR:0x3d80

DX1MDLR:0x4580

DX2MDLR:0x4d80

DX3MDLR:0x5580

DX4MDLR:0x5d80

DX5MDLR:0x6580

DX6MDLR:0x6d80

DX7MDLR:0x7580

DX8MDLR:0x7d80

DXnMDLR 是 Datx8 的主延时寄存器。

表 2-154: DXnMDLR 寄存器域的描述

名称	范围	类型	描述
MDLD	[23:16]	RW,0x0	MDL 延迟: 采用折半查找的方法, 对 DDR 周期进行测量, 该配置用于指定测量过程中 MDL 延迟值。
TPRD	[15:8]	RW,0x0	目标周期值。主延迟线周期测量值(主延迟线更新功能使能时, 该值会周期性的更新)。该值用于延迟线校准, 用作比例系数的分子。其它延时线(LCDL 或 BDL) 根据该调节因子(TPRD/ IPRD) 去计算延时线更新后的值,

			如果偏差超过阈值,则更新其它延迟线(PHY 发起 Update 操作)。
IPRD	[7:0]	RW,0x0	初始周期值。主延迟线初始测量得到的周期值。该值用于延迟线校准,用作比例系数的分母。
—	其它	—	保留。

2.3.49 DXnGTR

偏移地址:

DX0GTR:0x3e00

DX1GTR:0x4600

DX2GTR:0x4e00

DX3GTR:0x5600

DX4GTR:0x5e00

DX5GTR:0x6600

DX6GTR:0x6e00

DX7GTR:0x7600

DX8GTR:0x7e00

DXnGTR 为 Datx8 通用延时寄存器。

表 2-155: DXnGTR 寄存器域的描述

名称	范围	类型	描述
R3WLSL	[19:18]	RW,0x1	Rank0~3 的 Write Leveling 延迟:该值用于 Write Leveling 之后的 Pipe Line 调整。上电后,缺省值为 01。该值会根据 Data Training 结果进行调整,也可以直接写入来直接配置该值。有效取值范围 0~2: 2'b00: 写延迟=WL-1; 2'b01: 写延迟=WL; 2'b10: 写延迟=WL+1; 2'b11: 保留。
R2WLSL	[17:16]	RW,0x1	
R1WLSL	[15:14]	RW,0x1	
R0WLSL	[13:12]	RW,0x1	
R3DGSL	[11:9]	RW,0x0	Rank0~3 的 DQS Gating system 延迟:为了补偿系统板级延迟、封装延迟、IO 延迟等各种延迟以获得有效的读数数据,使用该寄存器延迟 DQS Gating,最大可达 7 个 SDR Clock 周期。在上电后,缺省值为 00。该值会根据 Data Training 结果进行调整,也可以直接写入来直接配置该
R2DGSL	[8:6]	RW,0x0	
R1DGSL	[5:3]	RW,0x0	
R0DGSL	[2:0]	RW,0x0	

			值。有效范围是 0~7。
—	其它	—	保留。

3 IO 路由部件与中断处理部件的 IO 寄存器

3.1 IO 路由部件的 IO 寄存器

3.1.1 基地址说明

表 3-1: IRU 基地址

模块	核心视角(PA[47:0])	维护视角(PA[39:0])	备注
IRU0	0x8062,0000,0000	0xe2,0000,0000	
IRU1	0x8063,0000,0000	0xe3,0000,0000	

注：各寄存器的偏移地址（相对于零地址）见具体寄存器说明。

3.1.2 IRU_CTRL

IRU0 核心地址：0x8062,0000, 0080

IRU0 维护地址：0xe2,0000, 0080

IRU_CTRL的设置DMA写脉冲的计数方式、PCIE DMA读NB计数控制等。维护和核心可读写，具体定义如下表：

表 3-2: IRU_CTRL 寄存器描述

名称	范围	类型	描述
PCI-E_wrcpl_mode	[1]	RW,1	为1，即使用脉冲信号来作为PCI-E接口 NB计数的减1指示； 为0，则此时IPU使用正常数据通路返回的PCI-E写结束来作为PCI-E0 NB计数的减1指示。
PCI-EDMAR_NBCNTEN	[0]	RW,0	PCI-E DMA读的NB计数控制。当该位为“1”时，DMA读请求发出时，NB计数器加“1”，收到读响应，NB计数器减“1”；该位为“0”，DMA读请求和响应不进行NB计数。默认为“0”。

3.1.3 IRU_ERR

IRU0 核心地址：0x8062,0000, 0100

IRU0 维护地址：0xe2,0000, 0100

IRU_ERR记录收到片上网络或者设备的消息包的格式是否有错误，以及包头和数据ECC单错和多错的情况，维护和核心可读，写清“0”。

表3-3: IRU_ERR寄存器描述

名称	范围	类型	描述
保留	[63:45]	Rsv	保留
保留	[44]	Rsv	保留
CTRingSrcErr	[43]	RWC,0	ION信用环网的消息源节点号错
SARingSrcErr	[42]	RWC,0	ION一致性环网的响应的请求源不是PCIE/MCU
RQRingSrcErr	[41]	RWC,0	ION请求环网的请求消息的来源不是核心
AckDBufWrErr	[40]	RWC,0	ION数据环网的数据写AckDBuf位置错
L2RFRDWrErr	[39]	RWC,0	ION数据环网的数据写L2RFRD位置错
DTRingSrcErr	[38]	RWC,0	ION数据环网的数据来源错
DTRingTypeErr	[37]	RWC,0	ION数据环网的数据类型非法
保留	[36:35]	Rsv	保留
SARingCreditErr	[34]	RWC,0	ION一致性环网响应控制信用计数器上溢
RQRingCreditErr	[33]	RWC,0	ION请求环网信用计数器上溢
SARingParityErr	[32:31]	RWC,0	ION顺/逆时钟一致性环网偶校验错
SABufOverFlow	[30:29]	RWC,0	ION顺/逆时钟一致性环网接收缓冲SABuf上溢
RQRingParityErr	[28:27]	RWC,0	ION顺/逆时钟请求环网偶校验错
RQBufOverFlow	[26:25]	RWC,0	ION顺/逆时钟请求环网接收缓冲RQBuf上溢
IRU_RspIntfErr	[24:23]	RWC,0	IRU#A/#B到ION的响应通路的接口协议错
IRU_RspTypeErr	[22:21]	RWC,0	IRU#A/#B到ION的响应类型错
IRU_ReqIntfErr	[20:19]	RWC,0	IRU#A/#B到ION的请求通路的接口协议错
IRU_ReqTypeErr	[18:17]	RWC,0	IRU#A/#B到ION的请求类型错
NBOverFlow	[16]	RWC,0	NB计数器溢出标志
PIU_ReqHead_FmtErr	[15]	RWC,0	PIU入片请求包头存在格式错
PIU_ReqHead_EccMErr	[14]	RWC,0	PIU入片请求包头存在多错
PIU_ReqData_EccMErr	[13]	RWC,0	PIU入片请求数据部分存在多错
PIU_EccSErr_Report	[12]	RWC,0	PIU入片请求和响应的单错之和超过了最大阈值

PIU_CplHead_FmtErr	[11]	RWC,0	PIU入片响应包头存在格式错
PIU_CplHead_EccMErr	[10]	RWC,0	PIU入片响应包头存在多错
MCU_ReqHead_FmtErr	[9]	RWC,0	MCU入片请求包头存在格式错
MCU_ReqHead_EccMErr	[8]	RWC,0	MCU入片请求包头存在多错
MCU_ReqData_EccMErr	[7]	RWC,0	MCU入片请求数据部分存在多错
MCU_CplHead_FmtErr	[6]	RWC,0	MCU入片响应包头存在格式错
MCU_CplHead_EccMErr	[5]	RWC,0	MCU入片响应包头存在多错
MCU_CplData_EccMErr	[4]	RWC,0	MCU入片响应数据部分存在多错 (DMAW数据)
MCU_EccSErr_Report	[3]	RWC,0	MCU入片响应和请求的单错之和超过了最大阈值
INTPU_CplHead_FmtErr	[2]	RWC,0	INTPU入片响应包头存在格式错
INTPU_CplHead_EccMErr	[1]	RWC,0	INTPU入片响应包头存在多错
INTPU_EccSErr_Report	[0]	RWC,0	INTPU入片请求和响应单错之和超过了最大阈值

3.1.4 IRU_ERREN

IRU0 核心地址: 0x8062,0000, 0180

IRU0 维护地址: 0xe2,0000, 0180

IRU_ERREN为IRU状态报错使能寄存器。维护和主核可读写, 复位为1。

表3-4: IRU_ERREN寄存器描述

名称	范围	类型	描述
保留	[63:44]	Rsv	保留
IRU_ErrEn	[43:0]	RW,0	IRU_ERR对应位的报错使能

3.1.5 SERR_CNTTH

IRU0 核心地址: 0x8062,0000, 0200

IRU0 维护地址: 0xe2,0000, 0200

IRU单错预警阈值, 维护和核心可读写。

表 3-5: SERR_CNTTH 寄存器描述

名称	范围	类型	描述
SerrCntTH	[31:0]	RW,0xffff,	IRU单错预警阈值。如果单错统计计数器的计数

		ffff	值大于该阈值且对应的单错报错使能位打开，则IPU向MCU报SERR错误。 单错统计计数器共有：MCUSERR_CNT、PIUSERR_CNT、INTPUSERR_CNT。
--	--	------	--

3.1.6 MCUSERR_CNT

IRU0 核心地址：0x8062,0000, 0280

IRU0 维护地址：0xe2,0000, 0280

该寄存器统计：IRU接收到MCU方向请求和响应包的包头和数据发生ECC单错的次数。维护和核心可读，“写清“0”。

表 3-6: MCUSERR_CNT 寄存器描述

名称	范围	类型	描述
MCUSERR_CNT	[31:0]	RWC,0x0	每发生一次ECC单错，计数器加“1”。
—	其它	—	保留。

3.1.7 PIUSERR_CNT

IRU0 核心地址：0x8062,0000, 0300

IRU0 维护地址：0xe2,0000, 0300

该寄存器统计：IRU接收到PIU方向请求和响应包的包头和数据发生ECC单错的次数。维护和核心可读，“写清“0”。

表 3-7: PIUSERR_CNT 寄存器描述

名称	范围	类型	描述
PIUSERR_CNT	[31:0]	RWC,0x0	每发生一次ECC单错，计数器加“1”。
—	其它	—	保留。

3.1.8 INTPUSERR_CNT

IRU0 核心地址：0x8062,0000, 0380

IRU0 维护地址：0xe2,0000, 0380

该寄存器统计：IRU接收到INTPU方向请求和响应包的包头和数据发生ECC单错的次数。维护和核心可读，“写清“0”。

表 3-8: INTPUSERR_CNT 寄存器描述

名称	范围	类型	描述
INTPUSERR_CNT	[31:0]	RWC,0x0	每发生一次ECC单错，计数器加“1”。
—	其它	—	保留。

3.1.9 LOCKEN

IRU0 核心地址：0x8062,0000, 0400

IRU0 维护地址：0xe2,0000, 0400

LOCKEN 设置在 IRU 中，用于控制核心 IO 访问是否单步执行，维护和核心可读写。

表 3-9: LOCKEN 寄存器描述

名称	范围	类型	描述
LOCKEN	[3:0]	RW,0x0	如果为LOCKEN[i]为1，则核心i发起的IO请求会单步执行。
—	其它	—	保留。

3.1.10 IRU_STAT

IRU0 核心地址：0x8062,0000, 0480

IRU0 维护地址：0xe2,0000, 0480

记录 IRU 中各接口的缓冲状态和转发处理状态机状态，维护和核心可读。

表 3-10: IRU_STAT 寄存器描述

名称	范围	类型	描述
保留	[63:55]	—	保留
IMRB_Full	[54]	RO,0	MCU接口入片请求缓冲满
IMRB_Empty	[53]	RO,1	MCU接口入片请求缓冲空
OMCB_Full	[52]	RO,0	MCU接口出片响应缓冲满
OMCB_Empty	[51]	RO,1	MCU接口出片响应缓冲空
OMRB_Full	[50]	RO,0	MCU接口出片请求缓冲满

OMRB_Empty	[49]	RO,1	MCU接口出片请求缓冲空
IMCB_Full	[48]	RO,0	MCU接口入片响应缓冲满
IMCB_Empty	[47]	RO,1	MCU接口入片响应缓冲空
OPReq_Full	[46]	RO,0	PIU接口出片请求缓冲满
OPReq_Empty	[45]	RO,1	PIU接口出片请求缓冲空
OPCpl_Full	[44]	RO,0	PIU接口出片响应缓冲满
OPCpl_Empty	[43]	RO,1	PIU接口出片响应缓冲空
IPReq_Full	[42]	RO,0	PIU接口入片请求缓冲满
IPReq_Empty	[41]	RO,1	PIU接口入片请求缓冲空
IPCpl_Full	[40]	RO,0	PIU接口入片响应缓冲满
IPCpl_Empty	[39]	RO,1	INTPU接口入片响应缓冲空
OIRB_Full	[38]	RO,0	INTPU接口出片请求缓冲满
OIRB_Empty	[37]	RO,1	INTPU接口出片请求缓冲空
IICB_Full	[36]	RO,0	INTPU接口入片响应缓冲满
IICB_Empty	[35]	RO,1	INTPU接口入片响应缓冲空
IPRState	[34:31]	RO,1	PIU接口入片请求处理状态机
IPCState	[30:28]	RO,1	PIU接口入片响应处理状态机
OPRState	[27:25]	RO,1	PIU接口出片请求处理状态机
OPCState	[24:22]	RO,1	PIU接口出片响应处理状态机
NBCnt	[21:18]	RO,0	PIU接口NB计数器的最低4位
IMRState	[17:13]	RO,1	MCU接口入片请求处理状态机
IxCState	[12:10]	RO,1	FCU_CB模块中入片响应处理状态机
OxRState	[9:6]	RO,1	FCU_CB模块中出片请求处理状态机
OxCState	[5:0]	RO,1	FCU_CB模块中出片响应处理状态机

3.2 中断处理部件的 IO 寄存器

3.2.1 基地址说明

表 3-11: INTPU 基地址

模块	核心视角(PA[47:0])	维护视角(PA[39:0])	备注
----	----------------	----------------	----

INTPU	0x8060,0000,0000	0xe0,0000,0000	
-------	------------------	----------------	--

注：各寄存器的偏移地址（相对于零地址）见具体寄存器说明。

3.2.2 CORE_SLEEP_STAT

偏移地址： 16'h0080

CORE_SLEEP_STAT为睡眠中断状态寄存器，该寄存器用于记录睡眠中断请求处理状态，该寄存器只读。

表 3-12: CORE_SLEEP_STAT 寄存器域的描述

名称	范围	类型	描述
Sleep	[19:16]	RO,0	Sleep [i]为“1”表示核心已经处于睡眠。 即CORE_Running=0。
Sleepy	[3:0]	RO,0	Sleepy [i]为“1”表示即将睡眠。 即睡眠中断发向核心。
—	其它	—	保留。

对于每一个核心 i 而言，这两位的含义如下：

表 3-13: 状态位描述

Sleep[i]	Sleepy[i]	核心 i 的状态
0	0	核心处于运行状态。
0	1	核心处于运行状态，即将睡眠。
1	0	核心处于睡眠状态。
1	1	非法状态。

3.2.3 DEVINT_MIS

偏移地址： 16'h0100

DEVINT_MIS为AMBA设备中断、PCI-E0中断、维护中断、PCI-E1、短时钟和系统错误中断、不可屏蔽中断丢失寄存器。该寄存器记录是否发生过因中断目标睡眠或INTEN寄存器为全0而产生的中断丢失。维护和核心可读，写清“0”。

表 3-14: DEVINT_MIS 寄存器域的描述

名称	范围	类型	描述
----	----	----	----

SHT_MISS	[24]	RWC,0	短时钟中断丢失标志。
NMI_MISS	[21]	RWC,0	不可屏蔽中断丢失标志
SYSERR_MISS	[20]	RWC,0	故障中断丢失标志。
AMBA_MISS	[19]	RWC,0	AMBA设备中断丢失。
PCIE1_MISS	[16:12]	RWC,0	PCI-E1中断丢失标志。同PCI-E0。
PCIE0_MISS	[8:4]	RWC,0	PCI-E0中断丢失标志，从高到低对应MSI、INTD、INTC、INTB、INTA中断。
MCU_MISS	[0]	RWC,0	维护中断丢失标志。
—	其它	—	保留。

其中不可屏蔽中断丢失、故障中断丢失会通知维护部件中断丢失，AMBA 设备中断丢失、维护中断丢失、PCIE 中断丢失会通知维护部件软件错。

3.2.4 SHORT_CLK_START

偏移地址： 16'h0200

SHORT_CLK_START为短时钟中断计数起始值寄存器。该寄存器用于指定短时钟中断计数器的初值。

表 3-15: SHORT_CLK_START 寄存器域的描述

名称	范围	类型	描述
SHORT_CLK_START	[63:0]	RW, 64'h1000,0000,0000,0000	短时钟中断计数起始值，最小取值范围大于等于64。

3.2.5 SHORT_CLK_MSK

偏移地址： 16'h0280

SHORT_CLK_MSK为以短时钟中断目标核心屏蔽寄存器，可读写，该寄存器用于指定不向哪些核心发送短时钟中断。

表 3-16: SHORT_CLK_MSK 寄存器域的描述

名称	范围	类型	描述
SHORT_CLK_MSK	[15:0]	RW,0	短时钟中断目标核心屏蔽位。
—	其它	—	保留。

3.2.6 MT_INDEX

偏移地址： 16h0300

MT_INDEX为维护中断映射寄存器。该寄存器用于把维护中断映射至64位中断向量中的第几位。

表 3-17: MT_INDEX 寄存器域的描述

名称	范围	类型	描述
MT_INDEX	[7:0]	RW,0x24	维护中断向量号。
—	其它	—	保留。

3.2.7 SHORT_INDEX

偏移地址： 16h0380

SHORT_INDEX为短时钟中断映射寄存器。该寄存器用于把短时钟中断映射至64位中断向量中的第几位。

表 3-18: SHORT_INDEX 寄存器域的描述

名称	范围	类型	描述
SHORT_INDEX	[7:0]	RW,0x25	短时钟中断向量号。
—	其它	—	保留。

3.2.8 INTEN

偏移地址： 16h0400

INTEN 为中断使能寄存器，每一位对应一个核心。该寄存器软件和维护可读写。当某一位为 0 时，向对应核心发送的中断均被丢弃，并进行记录。对该核心的核间中断、异步消息中断悬挂在 INTPU，直到该核心可以接受中断，再进行处理。

表 3-19: INTEN 寄存器域的描述

名称	范围	类型	描述
INTEN	[15:0]	RW,0x0	中断使能位，每一位对应一个核心。如果对应位为1，则允许向该核心发送中断请求；否则禁止中断请求。注意：每个核心的对该寄存器的

			写只能修改自己对应的位，维护可对所有位进行操作。
—	其它	—	保留。

3.2.9 AMBA_INDXX

偏移地址： 16'h0680

AMBA_INDXX为AMBA设备中断映射寄存器。该寄存器用于把AMBA设备中断映射至64位中断向量中的第几位。

表 3-20: AMBA_INDXX 寄存器域的描述

名称	范围	类型	描述
AMBA_INDXX	[7:0]	RW,0x27	AMBA设备中断向量号。
—	其它	—	保留。

注：申威 421 处理器不产生 AMBA 中断。

3.2.10 FMT_ERR

偏移地址： 16'h0700

FMT 记录收到片上网络或者设备的消息包的格式是否有错误，以及包头和数据 ECC 单错和多错的情况，维护和核心可读，写清“0”。

表 3-21: FMT_ERR 寄存器域的描述

名称	范围	类型	描述
MCU_REQ_ERR	[11:8]	RWC,0x0	MCU_REQ_ERR[3]: IO读越界; MCU_REQ_ERR[2]: IO写越界; MCU_REQ_ERR[1]: 收到MCU的IO写请求发生数据ECC多错; MCU_REQ_ERR[0]: 收到MCU的请求发生包格式错误。
IRU_REQ_ERR	[3:0]	RWC,0x0	IRU_REQ_ERR[3]: IO读越界; IRU_REQ_ERR[2]: IO写越界; IRU_REQ_ERR[1]: 收到IRU的IO写请求发生数据ECC多错; IRU_REQ_ERR[0]: 收到IRU的请求发生包格式

			错误。
—	其它	—	保留。

3.2.11 SYSERR_INDX

偏移地址： 16h0780

SYSERR_INDX为系统故障中断映射寄存器。该寄存器用于把系统故障中断映射至64位中断向量中的第几位。

表 3-22: SYSERR_INDX 寄存器域的描述

名称	范围	类型	描述
SYSERR_INDX	[7:0]	RW,0x26	系统故障中断向量号。
—	其它	—	保留。

3.2.12 NMI_INDX

偏移地址： 16h1580

NMI_INDX为不可屏蔽中断映射寄存器。该寄存器用于把不可屏蔽中断映射至64位中断向量中的第几位。

表 3-23: NMI_INDX 寄存器域的描述

名称	范围	类型	描述
NMI_INDX	[7:0]	RW,0x28	不可屏蔽中断向量号。
—	其它	—	保留。

3.2.13 SERR_CNTTH

偏移地址： 16h0880

INTPU 单错预警阈值，维护和核心可读写。

表 3-24: SERR_CNTTH 寄存器域的描述

名称	范围	类型	描述
SerrCntTH	[31:0]	RW,0xffff,ffff	INTPU单错预警阈值。如果单错统计计数器的计数值大于该阈值且对应的单错报错使能位打

			开，则INTPU向MCU报SERR错误。 单错统计计数器共有：MCUSERR_CNT、 IRUSERRCnt 。
—	其它	—	保留。

3.2.14 MCUSERR_CNT

偏移地址： 16'h0900

该寄存器统计：INTPU 接收到 MCU 方向请求包头发生 ECC 单错的次数。维护和核心可读，写清“0”。

表 3-25: MCUSERR_CNT 寄存器域的描述

名称	范围	类型	描述
MCUSERR_CNT	[31:0]	RWC,0x0	每发生一次ECC单错，计数器加“1”。
—	其它	—	保留。

3.2.15 IRUSERR_CNT

偏移地址： 16'h0a80

该寄存器统计：INTPU 接收到 IRU 方向请求包头发生 ECC 单错的次数。维护和核心可读，写清“0”。

表 3-26: IRUSERR_CNT 寄存器域的描述

名称	范围	类型	描述
IRUSERR_CNT	[31:0]	RWC,0x0	每发生一次ECC单错，计数器加“1”。
—	其它	—	保留。

3.2.16 ERRRPT_EN

偏移地址： 16'h0b00

该寄存器和 FMT_ERR 的报错位一一对应，ERRRPT_EN 中的某 bit 为“1”表示 FMT_ERR 中相应的 bit 报错使能。维护和核心可读写，复位为全“0”。

表 3-27: ERRRPT_EN 寄存器域的描述

名称	范围	类型	描述
MCU_REQ_ERREn	[11:8]	RW,0x0	MCU_REQ_ERR[3]: IO读越界报错使能; MCU_REQ_ERR[2]: IO写越界报错使能; MCU_REQ_ERR[1]: 收到MCU的IO写请求发生ECC多错报错使能。 MCU_REQ_ERR[0]: 收到MCU的请求发生包格式错误报错使能。
IRU_REQ_ERREn	[3:0]	RW,0x0	IRU_REQ_ERR[3]: IO读越界报错使能; IRU_REQ_ERR[2]: IO写越界报错使能; IRU_REQ_ERR[1]: 收到IRU的IO写请求发生数据ECC多错报错使能; IRU_REQ_ERR[0]: 收到IRU的请求发生包格式错误报错使能。
—	其它	—	保留。

3.2.17 SYSERR_NMI_INT_DST

偏移地址: 16'h0b80

SYSERR_NMI_INT_DST 为系统故障和不可屏蔽中断的目标配置寄存器。

表 3-28: SYSERR_NMI_INT_DST 寄存器域的描述

名称	范围	类型	描述
NMIINT_DSTEN	[12]	RW,0	复位为“0”; 当该位为“0”时, 中断目标在可接收中断的核心中轮转; 当该位为“1”时, 中断目标为NMIINT_DST所指定的核心, 如果NMIINT_DST所指定的核心不可接收中断, 则从其它可接收中断的核心中选择一个作为中断目标。
NMIINT_DST	[11:8]	RW,0	编码表示不可屏蔽中断目标核心心号, 复位值为4'b0。
SYSERRINT_DSTEN	[4]	RW,0	复位为“0”; 当该位为“0”时, 中断目标在可接收中断的核心中轮转; 当该位为“1”时, 中断目标为SYSERRINT_DST所指定的核心, 如果SYSERRINT_DST所指定的核心不可接收中

			断，则从其它可接收中断的核心中选择一个作为中断目标。
SYSERRINT_DST	[3:0]	RW,0	编码表示故障中断目标核心心号，复位值为4'b0。
—	其它	—	保留。

3.2.18 PCIE0_INTx_INDx

偏移地址： 16'h0c80~16'h0e00

PCIE0_INTx_INDx为PCI-E0的INTx中断映射寄存器。该寄存器用于把INTa—INTd中断映射至64位中断向量中的第几位。

表 3-29: PCIE0_INTx_INDx 寄存器域的描述

名称	范围	类型	描述
INTx_INDx	[7:0]	RW, 0x20, 0x21, 0x22, 0x23	PCI-E0的INTx中断向量号, INTa-d对应复位的初值为0x20-0x23。
—	其它	—	保留。

3.2.19 PCIE1_INTx_INDx

偏移地址： 16'h0e80~16'h1000

PCIE1_INTx_INDx为PCI-E1的INTx中断映射寄存器。该寄存器用于把INTa—INTd中断映射至64位中断向量中的第几位。

表 3-30: PCIE1_INTx_INDx 寄存器域的描述

名称	范围	类型	描述
INTx_INDx	[7:0]	RW, 0x1C, 0x1D, 0x1E, 0x1F	PCI-E1的INTx中断向量号, INTa-d对应复位的初值为0x1C-0x1F。
—	其它	—	保留。

3.2.20 IINT_MIS

偏移地址： 16'h1180

IINT_MIS 设置在 INTPU 中，用于记录目标核心的核间中断丢失情况。该寄存器软件和维护可读写，硬件实现为写清 0。如果核间中断 0、核间中断 1、睡眠中断的目标核心断连，则由硬件隐式写 IINT_MIS 对应位。

表 3-31: IINT_MIS 寄存器域的描述

名称	范围	类型	描述
SlpMIS	[47:32]	RWC,0x0	比特32~47分别对应核心0~15的睡眠中断丢失情况，为“1”表示对应的目标核心发生中断丢失。
CoreInt1MIS	[31:16]	RWC,0x0	比特16~31分别对应核心0~15的核间中断1丢失情况，为“1”表示对应的目标核心发生中断丢失。
CoreInt0MIS	[15:0]	RWC,0x0	比特0~15分别对应核心0~15的核间中断0丢失情况，为“1”表示对应的目标核心发生中断丢失。
—	其它	—	保留。

3.2.21 IINTMIS_RPTEN

偏移地址： 16'h1200

IINTMIS_RPTEN 设置在 INTPU 中，用于 IOR: IINT_MIS 的报错使能。该寄存器软件和维护可读写。

表 3-32: IINTMIS_RPTEN 寄存器域的描述

名称	范围	类型	描述
SlpMIS_RPTEN	[47:32]	RW,0x0	比特32~47分别对应核心0~15的睡眠中断丢失情况的报错使能，为“1”表示如果对应的目标核心发生中断丢失，可以向维护报错。
CoreInt1MIS_RPTEN	[31:16]	RW,0x0	比特16~31分别对应核心0~15的核间中断1丢失情况的报错使能，为“1”表示如果对应的目标

			核心发生中断丢失，可以向维护报错。
CoreInt0MIS_RPTEN	[15:0]	RW,0x0	比特0~15分别对应核心0~15的核间中断0丢失情况的报错使能，为“1”表示如果对应的目标核心发生中断丢失，可以向维护报错。
—	其它	—	保留。

3.2.22 DEVINT_MIS_RPTEN

偏移地址： 16h1280

DEVINT_MIS_RPTEN 为 IOR: DINT_MIS 中 PCI-E0 中断、维护中断、PCI-E1 中断、故障中断、不可屏蔽中断丢失情况的报错使能。维护和核心可读写。

表 3-33: DEVINT_MIS_RPTEN 寄存器域的描述

名称	范围	类型	描述
NMI_MISS_RPTEN	[21]	RW,0	不可屏蔽中断丢失报错使能。
SYSERR_MISS_RPTEN	[20]	RW,0	故障中断丢失报错使能。
AMBA_MISS_RPTEN	[19]	RW,0	AMBA设备中断丢失报错使能。
PCIE1_MISS_RPTEN	[16:12]	RW,0	PCI-E1中断丢失报错使能。同PCI-E0。
PCIE0_MISS_RPTEN	[8:4]	RW,0	PCI-E0中断丢失报错使能，从高到低对应MSI、INTD、INTC、INTB、INTA中断。
MCU_MISS_RPTEN	[0]	RW,0	维护中断丢失报错使能。
—	其它	—	保留。

3.2.23 ECCSERR

偏移地址： 16h1300

ECCSERR为记录INTPU模块发生ECC单错的情况。维护和核心可读写，写清“0”。

表 3-34: ECCSERR 寄存器域的描述

名称	范围	类型	描述
CntMCUSErr	[1]	RWC,0	INTPU收到MCU包头和数据单错达到阈值。
CntIRUSErr	[0]	RWC,0	INTPU收到IRU包头和数据单错达到阈值。
—	其它	—	保留。

3.2.24 ECCSERR_RPTEN

偏移地址： 16'h1380

ECCSERR_RPTEN为记录INTPU模块发生ECC单错的报错使能寄存器。维护和核心可读写。

表 3-35： ECCSERR_RPTEN 寄存器域的描述

名称	范围	类型	描述
CntMCUSErrEn	[1]	RW,0	INTPU收到MCU包头和数据单错达到阈值报错使能。
CntIRUSErrEn	[0]	RW,0	INTPU收到IRU包头和数据单错达到阈值报错使能。
—	其它	—	保留。

3.2.25 ECCMERR

偏移地址： 16'h1400

ECCMERR为记录INTPU模块发生ECC多错的情况。维护和核心可读写，写清“0”。

表 3-36： ECCMERR 寄存器域的描述

名称	范围	类型	描述
CntMCUMErr	[1]	RWC,0	INTPU收到MCU包头发生多错。
CntIRUMErr	[0]	RWC,0	INTPU收到IRU包头发生多错。
—	其它	—	保留。

3.2.26 ECCMERR_RPTEN

偏移地址： 16'h1480

ECCMERR_RPTEN为记录INTPU模块发生ECC多错的报错使能寄存器。维护和核心可读写。

表 3-37： ECCMERR_RPTEN 寄存器域的描述

名称	范围	类型	描述
CntMCUMErrEn	[1]	RW,0	INTPU收到MCU包头和数据发生多错报错使能。
CntIRUMErrEn	[0]	RW,0	INTPU收到IRU包头和数据发生多错报错使能。
—	其它	—	保留。

3.2.27 DEVINT_WKEN

偏移地址： 16'h1500

DEVINT_WKEN为PCI-E0/1中断、维护中断的睡眠唤醒使能寄存器。维护和核心可读写。

表 3-38: DEVINT_WKEN 寄存器域的描述

名称	范围	类型	描述
PCIE1_WKEN	[16:12]	RW,0	PCI-E1中断唤醒使能。含义同PCI-E0。
PCIE0_WKEN	[8:4]	RW,0	[8:4]: PCI-E0中断唤醒使能，从高到低对应MSI、INTD、INTC、INTB、INTA中断。当收到PCI-E0中断，且目标核心睡眠，如果对应位为“1”，则唤醒该目标核心。 触发唤醒时，是否登记 IOR: DEVINT_MIS 还需再判断DEVINTWK_INTEN的值，DEVINTWK_INTEN 不使能则登记 IOR: DEVINT_MIS，如果DEVINTWK_INTEN使能则在目标核心唤醒后再次递交中断（中断未丢），不登记 IOR: DEVINT_MIS。
AMBA_WKEN	[1]	RW,0	AMBA设备中断唤醒使能。当收到AMBA设备中断，且对应目标核心睡眠，如果该位为“1”，则登记IOR: DEVINT_MIS的同时，唤醒该目标核心。
MCU_WKEN	[0]	RW,0	维护中断唤醒使能。当收到普通维护中断，且对应目标核心睡眠，如果该位为“1”，则登记IOR: DEVINT_MIS的同时，唤醒该目标核心。
—	其它	—	保留。

3.2.28 DEVINTWK_INTEN (新增)

偏移地址： 16'h1600

DEVINTWK_INTEN为PCI-E0/1中断唤醒再次中断使能寄存器。维护和核心可读写。

表 3-39: DEVINTWK_INTEN 寄存器域的描述

名称	范围	类型	描述
PCIE1_WKEN	[16:12]	RW,0	PCI-E1中断唤醒再次中断使能。含义同PCI-E0。
PCIE0_WKEN	[8:4]	RW,0	[8:4]: PCI-E0中断唤醒再次中断使能，从高到低对应

			MSI、INTD、INTC、INTB、INTA中断。当收到PCI-E0中断，且目标核心睡眠且DEVINT_WKEN使能，如果对应位为“1”，则在目标核心唤醒后再次形成中断响应递交给目标核心，不登记IOR: DEVINT_MIS。
—	其它	—	保留。

3.2.29 MAIL_BOX_i

偏移地址： 16'h4000~16'h4780

MAIL_BOX_i (i: 0~3) 异步消息队列，该队列为64b×16的FIFO结构，记录其它核心（包括核心i）发给核心i的异步中断短消息。

如果其它核心（包括核心i）对IOR: MAIL_BOX_i进行IO写，会触发对核心i的异步消息中断，同时，IO写携带的数据（中断短消息）缓存入异步消息队列中。核心i在适当时刻读该队列的中断短消息，从而达到核心间通信的目的。

MAIL_BOX_i 只有核心 i 可以读，所有核心可写；维护不可读写。复位时，所有队列为空。

3.2.30 MAIL_STAT_i

偏移地址： 16'h4800~16'h4f80

MAIL_STAT_i (i: 0~3) 为异步消息队列状态寄存器。维护和核心只读。具体定义如下表：

表 3-40: MAIL_STAT_i 寄存器域的描述

名称	范围	类型	描述
MAIL_NUM	[20:16]	RO,0	该目标核心的收到的邮件数量。每个核心最多可接收16个异步消息。
FULL_FLAG	[15:0]	RO,0	某位为“1”则表示某个源核心对该目标核心发送异步消息时，邮箱已满。
—	其它	—	保留。

当核心n想往核心m发异步消息时，可通过写IOR: MAIL_BOX_m发送一个64位的消息，系统接口收到后，如果IOR: MAIL_STAT_m[MAIL_NUM]指示邮箱不满，则将消息保存，系统接口自动向核心m发一个核间异步消息中断请求。如果IOR: MAIL_STAT_m[MAIL_NUM]指示邮箱满，则置IOR: MAIL_STAT_m[FULL_FLAG[n]]为1，指示本次消息丢失。

软件必须保证源核心每发一次消息，都要检查 IOR: MAIL_STAT_m[FULL_FLAG[n]]是否为“1”，

如果为“1”，则说明刚发出的消息不成功，需要重发。

3.2.31 PCIE0_MSI[i]_INDX

偏移地址： 16'h8000~16'hff80

PCIE0_MSI[i]_INDX (i: 0~255) 为PCI-E0的MSI中断映射REG，维护和核心可读写，该寄存器用于把MSI[i]中断映射至中断向量中的第几位。软件在配置PCI-E时需要配置该IO寄存器以指定中断位置，注意不要影响CSR：INT_STAT0~3中已经定义各域。

表 3-41: PCIE0_MSI[i]_INDX 寄存器域的描述

名称	范围	类型	描述
MSI[i]_INDX	[7:0]	RW,0	PCI-E0的MSI中断向量号。
—	其它	—	保留。

3.2.32 PCIE1_MSI[i]_INDX

偏移地址： 20'h10000~20'h17f80

PCIE1_MSI[i]_INDX (i: 0~255) 为PCI-E1的MSI中断映射REG，维护和核心可读写，该寄存器用于把MSI[i]中断映射至中断向量中的第几位。软件在配置PCI-E时需要配置该IO寄存器以指定中断位置，注意不要影响CSR：INT_STAT0~3中已经定义各域。

表 3-42: PCIE1_MSI[i]_INDX 寄存器域的描述

名称	范围	类型	描述
MSI[i]_INDX	[7:0]	RW,0	PCI-E1的MSI中断向量号。
—	其它	—	保留。

4 维护接口的 IO 寄存器

4.1 维护接口的 IO 寄存器

4.1.1 基地址说明

表 4-1: MCU 基地址

模块	核心视角(PA[47:0])	维护视角(PA[39:0])	备注
MCU	0x8050,0000,0000	0xd0,0000,0000	

注：各寄存器的偏移地址（相对于零地址）见具体寄存器说明。

4.1.2 CG_ONLINE

偏移地址： 16'h0000

CG_ONLINE 为核组在位使能寄存器，只读。如果核组不在位，则对应的核心、三级 Cache 和存控都处于复位状态。

表 4-2: CG_ONLINE 寄存器域的描述

名称	范围	类型	描述
CG_ONLINE	[3:0]	RO	<p>CG_ONLINE[i]为“1”，表示对应核组在位。</p> <p>CG_ONLINE根据维护引脚CHIP_MODH_H[2:0]译码得到。</p> <p>合法配置有以下三种：</p> <p>4'b1111：全功能模式</p> <p>4'b0011：八核模式</p> <p>4'b0001：四核模式</p> <p>注1：核组关闭时，CPM和三级处于复位状态，时钟处于低频状态。申威421处理器该域高三位固定为零。</p> <p>注 2：未关闭的核心和存控，可以通过配置寄存器（CORE_ONLINE、MC_ONLINE）以及睡眠中断进行动态关闭！</p>
—	其它	—	保留。

表 4-3: CHIP_MODE_H[2:0]引脚说明

信号名	位宽	属性	含义
CHIP_MODE_H[2:0]	3	输入	芯片模式。 3'b0xx: 四核模式。强制关闭核组 1、核组 2 和核组 3（包括核心与对应的 2 路主存和三级 Cache） 3'b100: 八核经济模式。强制关闭核组 2 和核组 3（包括核心与对应的 2 路主存和三级 Cache）以及核组 0 和核组 1 的 MM1； 3'b101: 八核模式。强制关闭核组 2 和核组 3（包括核心与对应的 2 路主存和三级 Cache）。 3'b110: 16 核经济模式。强制关闭每个核组的 MM1； 3'b111: 16 核全功能模式。

4.1.3 MC_ONLINE

偏移地址: 16'h3780

MC_ONLINE 为存控在位使能寄存器，维护可读写，软件只读。该寄存器用于表示存储控制器是否在位。

表 4-4: MC_ONLINE 寄存器域的描述

名称	范围	类型	描述
MC_ONLINE	[7:0]	RW,DEF	MC_ONLINE[i]为“1”，表示对应存控在位。 MC_ONLINE初值根据维护引脚CHIP_MODH_H[2:0]（见表4-3）译码得到。 合法配置有以下三种： 8'b1111_1111: 全功能模式 8'b0101_0101: 全功能经济模式 8'b0000_1111: 八核模式 8'b0000_0101: 八核经济模式 8'b0000_0011: 四核模式 在该值基础上，维护可以关闭在位的核心，但不能开启被CHIP_MODE_H屏蔽的存控。

			申威421处理器该域高六位固定为零。
—	其它	—	保留。

注 1: MC_ONLINE 对于存控时钟和复位的控制是直接控制，即对该 IOR 的修改直接影响存控时钟的高低频选择和复位。

注 2: 为了简化处理，在降级使用时各个核组的存控控制器要一致，即四个核组都只使能一路存控或者全部使能两路存控。

4.1.4 CORE_ONLINE

偏移地址： 16'h780

CORE_ONLINE 为核心在位使能寄存器，维护可读写，软件只读。该寄存器用于控制核心的在位使能，复位时初值为与 CHIP_MODE_H 有关。

表 4-5: CORE_ONLINE 寄存器域的描述

名称	范围	类型	描述
CORE_ONLINE	[15:0]	RW,DEF	CORE_ONLINE[i]为“1”，表示该核心在位，该位只控制核心。 CORE_ONLINE 初值根据维护引脚 CHIP_MODH_H[2:0]（见表4-3）译码得到。 合法配置有以下三种： 16'hfff: 全功能模式 16'hffff: 全功能经济模式 16'h00ff: 八核模式 16'h00ff: 八核经济模式 16'h000f: 四核模式 在该值基础上，维护可以关闭在位的核心，但不能开启被CHIP_MODE_H屏蔽的存控。 申威421处理器该域高12位固定为零。
—	其它	—	保留。

4.1.5 CGx_FAULT_STAT

偏移地址： 16'h0080~16'h0200

申威 421 处理器只有 CG0_FAULT_STAT 有意义，其它保留。

CGx_FAULT_STAT 为核组总错状态寄存器，维护和软件可读写。该寄存器用于存放各个核组的总错信息。复位时初值为全“0”。采上升沿登记；按位写“1”清除。

表 4-6: CGx_FAULT_STAT 寄存器域的描述

名称	范围	类型	描述
MC1_FAULT	[31:28]	RW1C,0	[28]: MC1主存读数据产生单错预警; [29]: MC1主存读数据有多错; [30]: MC1写主存数据产生单错预警; [31]: MC1写主存数据产生多错, 包括请求通路的偶校验错。
SOFTERR	[27:24]	RW1C,0	分别表示对应4个核心的软件错误。根据相应的IOR: SOFT_INF[63]位挤脉冲置1。
MC0_FAULT	[23:20]	RW1C,0	[20]: MC0主存读数据产生单错预警; [21]: MC0主存读数据有多错; [22]: MC0写主存数据产生单错预警; [23]: MC0写主存数据产生多错, 包括请求通路的偶校验错。
CPM_FAULT	[19:16]	RW1C,0	[16]: CPM产生控制错误、一致性错等硬件故障; [17]: CPM单错预警, 含CTAG和传输通路; [18]: CPM多错, 含CTAG和传输通路; [19]: 保留。

Core_FAULT	[15:0]	RW1C,0	分别记录每个核组内各个核心的错误信息，具体含义如下： [0]: 核心0产生机器检查错故障（Rob检查错、Dbox检查错、Sbox检查错）该位置1；如果软件也设置了相应CSR：HARD_ERR，该位也为1； [1]: 核心0产生单错预警； [2]: 核心0产生多错； [3]: 核心0产生软件故障（写地址越界，或者如果软件设置了相应CSR：SOFT_ERR，则该位也为1）； [4]: 核心1产生机器检查错故障（同上）； [5]: 核心1产生单错预警； [6]: 核心1产生多错； [7]: 核心1产生软件故障（同上）。 [8]: 核心2产生机器检查错故障（同上）； [9]: 核心2产生单错预警； [10]: 核心2产生多错； [11]: 核心2产生软件故障（同上）。 [12]: 核心3产生机器检查错故障（同上）； [13]: 核心3产生单错预警； [14]: 核心3产生多错； [15]: 核心3产生软件故障（同上）。
—	其它	—	保留。

4.1.6 CGx_FAULT_EN

偏移地址： 16'h0280~16'h0400

申威 421 处理器只有 CG0_FAULT_EN 有意义，其它保留。

CGx_FAULT_EN 为核组总错状态报错使能寄存器，维护和软件都可读写。该寄存器用于指示 CGx_FAULT_STAT 中那些错误标志位参与总错信号的生成。复位时初值为全“0”。

表 4-7: CG_FAULT_EN 寄存器域的描述

名称	范围	类型	描述
MC1_FAULT_EN	[31:28]	RW,0	[28]: MC1主存读数据产生单错预警报错使能； [29]: MC1主存读数据有多错报错使能； [30]: MC1写主存数据产生单错预警报错使能；

			[31]: MC1写主存数据产生多错报错使能。
SOFTERR_EN	[27:24]	RW,0	分别表示对应4个核心的软件错误报错使能。
MC0_FAULT_EN	[23:20]	RW,0	[20]: MC0主存读数据产生单错预警报错使能; [21]: MC0主存读数据有多错报错使能; [22]: MC0写主存数据产生单错预警报错使能; [23]: MC0写主存数据产生多错报错使能。
CPM_FAULT_EN	[19:16]	RW,0	[16]: CPM产生控制错误、一致性错等硬件故障报错使能; [17]: CPM单错预警, 含CTAG和传输通路报错使能; [18]: CPM多错, 含CTAG和传输通路报错使能; [19]: 保留。
Core_FAULT_EN	[15:0]	RW,0	分别记录每个核组内各个模块的错误信息, 具体含义如下: [0]: 核心0产生机器检查错故障报错使能; [1]: 核心0产生单错预警报错使能; [2]: 核心0产生多错报错使能; [3]: 核心0产生软件故障报错使能; [4]: 核心1产生机器检查错故障报错使能; [5]: 核心1产生单错预警报错使能; [6]: 核心1产生多错报错使能; [7]: 核心1产生软件故障报错使能。 [8]: 核心2产生机器检查错故障报错使能; [9]: 核心2产生单错预警报错使能; [10]: 核心2产生多错报错使能; [11]: 核心2产生软件故障报错使能。 [12]: 核心3产生机器检查错故障报错使能; [13]: 核心3产生单错预警报错使能; [14]: 核心3产生多错报错使能; [15]: 核心3产生软件故障报错使能。
—	其它	—	保留。

4.1.7 CGx_FAULT_INT_EN

偏移地址: 16'h0480~16'h0600

申威 421 处理器只有 **CG0_FAULT_INT_EN** 有意义，其它保留。

CGx_FAULT_INT_EN 为核组故障中断使能寄存器，维护和软件都可读写。该寄存器用于指示 CG_FAULT_STAT 中那些错误标志位参与故障中断的生成。复位时初值为全“0”。

表 4-8: CG_FAULT_INT_EN 寄存器域的描述

名称	范围	类型	描述
MC1_FAULT_INT_EN	[31:28]	RW,0	[28]: MC1主存读数据产生单错预警中断使能; [29]: MC1主存读数据有多错中断使能; [30]: MC1写主存数据产生单错预警中断使能; [31]: MC1写主存数据产生多错中断使能。
SOFTERR_INT_EN	[27:24]	RW,0	分别表示对应4个核心的软件错误中断使能。
MC0_FAULT_INT_EN	[23:20]	RW,0	[20]: MC0主存读数据产生单错预警中断使能; [21]: MC0主存读数据有多错中断使能; [22]: MC0写主存数据产生单错预警中断使能; [23]: MC0写主存数据产生多错中断使能。
CPM_FAULT_INT_EN	[19:16]	RW,0	[16]: CPM产生控制错误、一致性错等硬件故障中断使能; [17]: CPM单错预警, 含CTAG和传输通路中断使能; [18]: CPM多错, 含CTAG和传输通路中断使能; [19]: 保留。
Core_FAULT_INT_EN	[15:0]	RW,0	分别记录每个核组内各个模块的错误信息, 具体含义如下: [0]: 核心0产生机器检查错故障中断使能; [1]: 核心0产生单错预警中断使能; [2]: 核心0产生多错中断使能; [3]: 核心0产生软件故障中断使能; [4]: 核心1产生机器检查错故障中断使能; [5]: 核心1产生单错预警中断使能; [6]: 核心1产生多错中断使能; [7]: 核心1产生软件故障中断使能。 [8]: 核心2产生机器检查错故障中断使能; [9]: 核心2产生单错预警中断使能; [10]: 核心2产生多错中断使能; [11]: 核心2产生软件故障中断使能。 [12]: 核心3产生机器检查错故障中断使能;

			[13]: 核心3产生单错预警中断使能; [14]: 核心3产生多错中断使能; [15]: 核心3产生软件故障中断使能。
—	其它	—	保留。

4.1.8 INIT_CTL

偏移地址： 16h0680

INIT_CTL 是初始化状态机控制寄存器，维护可读写，软件只读。该寄存器用来控制复位初始化流程，复位时初值为全“0”。

表 4-9: INIT_CTL 寄存器域的描述

名称	范围	类型	描述
RSTENT_EN	[41]	RW,0x1	该位为“1”，表示需要在进入复位状态进行正常延时等待。
STRUN_EN	[40]	RW,0x1	该位为“1”，表示需要在启动运行状态进行运行。
ICLD_EN	[39]	RW,0x1	该位为“1”，表示需要在初始化加载状态进行加载。
CFG3_EN	[38]	RW,0x1	该位为“1”，表示需要在配置3状态进行配置。
BIST_EN	[37]	RW,0x1	该位为“1”，表示需要在存储器自测试状态进行测试。
CFG2_EN	[36]	RW,0x1	该位为“1”，表示需要在配置2状态进行配置。
RSTEND_EN	[35]	RW,0x1	该位为“1”，表示需要在结束复位状态进行正常延时等待。
CLKCHG_EN	[34]	RW,0x1	该位为“1”，表示需要在时钟切换1/2状态进行正常切换。
PLLUP_EN	[33]	RW,0x1	该位为“1”，表示需要在PLL升频状态等待升频结束。
CFG1_EN	[32]	RW,0x1	该位为“1”，表示需要在配置1状态进行配置。
MC_FREQ	[18:16]	RW,0	存控频率配置。在配置1状态下，若引脚CFG_SEL_L为“0”，则修改为引脚CFG_MM_H[2:0]的值
RING_FREQ	[10:8]	RW,10	互连频率配置。在配置1状态下，若引脚CFG_SEL_L为“0”，则修改为引脚CFG_RING_H[2:0]的值
CORE_FREQ	[3:0]	RW,26	核心频率配置。在配置1状态下，若引脚CFG_SEL_L为“0”，则修改为引脚CFG_CORE_H[3:0]的值
—	其它	—	保留。

在引脚 CFG_SEL_L[1:0]为零时，硬件会根据 INIT_MODE_H 对 INIT_CTL 进行隐式写，具体见下表。

表 4-10: INIT_MODE_H 说明

INIT_MODE_H[1:0] 初始化状态机状态	2'b00	2'b01	2'b10	2'b11
配置 1 状态	无操作			可读写维护控制器的 I/O 寄存器
PLL 升频状态	无操作			延时等待
时钟切换状态	快速进行时钟切换			正常进行时钟切换
结束复位状态	无操作			延时等待
配置 2 状态	停在此状态	无操作	可读写维护控制器的 I/O 寄存器和配置存储器自测试相关的扫描链寄存器	
存储器自测试状态	—	无操作	进行存储器自修复测试	
配置 3 状态	—	无操作	可读写所有 I/O 寄存器	
初始化加载状态	—	无操作	进行初始化程序加载	
启动运行状态	—	停在此状态	延时等待	
应用场景	测试模拟	程序模拟	运行模拟	正常运行

表 4-11: INIT_CTL 隐式写说明

INIT_MODE_H INIT_CTL	2'b00	2'b01	2'b10	2'b11
RSTENT_EN	0	0	0	1
STRUN_EN	0	0	1	1
ICLD_EN	0	0	1	1
CFG3_EN	0	0	1	1
BIST_EN	0	0	1	1
CFG2_EN	1	0	1	1
RSTEND_EN	0	0	0	1
CLKCHG_EN	0	0	0	1
PLLUP_EN	0	0	0	1
CFG1_EN	0	0	0	1

4.1.9 BIST_CTL

偏移地址： 16'h2200

BIST_CTL 是 BIST 控制寄存器，维护可读写，软件只读。该寄存器用来控制复位初始化流程，

复位时初值为全“0”。

表 4-12: BIST_CTL 寄存器域的描述

名称	范围	类型	描述
SAEN_SEL_TData	[27:26]	RW,0x0	对应三级Cache的SAEN_SEL值。
SAEN_SEL_CORE	[23:22]	RW,0x0	对应核心的SAEN_SEL的值。
SAEN_SEL_EN	[21]	RW,0x0	<p>指定SAEN值的有效位。</p> <p>该功能只针对BIST测试的新增需求（在不扫入的情况指定SAEN进行BIST或BISR的测试）。</p> <p>即上电复位冷复位时，把该寄存器配置为“1”，并配置SAEN_SEL_CORE/SAEN_SEL_CPM，且BistMode配置为BIST或BISR模式，则相应的阵列（可统一测试）进行指定SAEN的测试。</p> <p>注1：各阵列的SAEN的值指定相同；</p> <p>注2：正常流程时该位“0”；</p> <p>注3：配置该寄存器其测试类型必须是BIST或BISR，否则为非法配置。</p>
SAEN_SEL_CTRL	[20]	RW,0x0	<p>指示SAEN选择策略：为“0”则按照可靠性优先原则，否则为性能优先原则。按照可靠性选择原则时，灵敏放大器配置值从3、2、1到0的顺序选择可通过测试的配置；否则从0、1、2到3的顺序选择可通过测试的配置。</p>
MEMIDValid	[19]	RW,0x0	<p>当进行存储器自测试时有意义，表示对指定阵列进行测试。阵列编号由CGID和MEMID指定，BIST类型由BistMode指定。主要用于对指定阵列的debug测试。</p>
CGID	[18:16]	RW,0x0	<p>CG编号；当指定阵列测试时有意义。</p> <p>3'b000: 表示核组0；</p> <p>3'b001: 保留；</p> <p>3'b010: 保留；</p> <p>3'b011: 保留；</p> <p>3'b100: PageCache0；</p> <p>3'b101: PageCache1；</p> <p>其它保留。</p>
MEMID	[15:8]	RW,0x0	<p>指定阵列编号；当指定阵列测试时有意义。</p> <p>特别关注该位对应不同芯片、不同封装的不同含义。</p> <p>[15:14]没有意义；</p>

			<p>当 CGID 表示核组时，</p> <p>[13]=1'b0，表示对核心进行 Bist 测试；核心号由[12:11]指定；</p> <p>2'b00: 核心 0；</p> <p>2'b01: 核心 1；</p> <p>2'b10: 核心 2；</p> <p>2'b11: 核心 3；</p> <p>核心内的阵列号由[10:8]指定；</p> <p>3'b001: 表示 ICache；</p> <p>3'b011: 表示 DCache；</p> <p>3'b100: 表示 Stag；</p> <p>3'b101: 表示 SCache；</p> <p>其它值：保留。</p> <p>[13]=1'b1，表示对三级 Cache 进行 Bist 测试；具体阵列由[10:8]指定：</p> <p>3'b000: 表示 TTAG；</p> <p>3'b001: 表示 TDATA；</p> <p>其它保留</p> <p>当 CGID 表示 PageCache 时，具体阵列由[10:8]指定。</p> <p>3'b001: 表示 Pache；</p> <p>其它保留。</p>
BistMode	[6:4]	RW,0x6	<p>Bist测试类型；当指定阵列测试和全阵列测试都有意义。</p> <p>“3'b0xx”：Debug 测试（特殊的 BIST 自测试，当发现错误时，暂停测试，将错误信息锁存到一个可扫描的寄存器中，此模式下只能对指定的存储器进行），[1:0]为敏感放大器设置值；</p> <p>“3'b100”：使用之前保留的灵敏放大器配置进行 BIST 初始化（该配置可以通过 Scan 扫入配置）；</p> <p>“3'b101”：使用之前保留的灵敏放大器配置进行 BISR 自修复测试（该配置可以通过 Scan 扫入配置）；</p> <p>“3'b11x”：自动迭代的灵敏放大器配置进行BISR自修复测试。</p> <p>注：Debug测试只有在指定阵列测试时有意义；非指定阵列测试时，debug模式强制成BIST、BISR或ScanBISR（即</p>

			直接把BistMode[2]强制成“1”);
HOTRST_CTL[1:0]	[3:2]	RW,0x0	0x: 表示核心在唤醒以及热复位进行保留原SAEN的BIST测试; 10: 表示核心在唤醒以及热复位时进行保留原SAEN的BISR测试; 11: 表示核心在唤醒以及热复位时进行SAEN自动迭代的BISR测试。 注: 睡眠唤醒流程根据该域指定测试类型, 默认是BIST测试。
REBIST	[1]	RW,0x0	重新BIST测试指示, 在配置3状态判断该标志, 如果该位有效, 则重新转入BIST测试, 同时对该位进行自清除。
—	其它	—	保留。

4.1.10 BIST_OVERTIME_TH

偏移地址: 16'h2280

存储器测试超时阈值寄存器 BIST_OVERTIME_TH, 维护可读写, 软件只读。该寄存器用于存储器自测试时的超时阈值设置。

表 4-13: BIST_OVERTIME_TH 寄存器域的描述

名称	范围	类型	描述
BIST_OVERTIME_EN	[4]	RW,0x1	存储器自测试超时机制使能。
BIST_OVERTIME_TH	[3:0]	RW,0xf	存储器自测试时的超时阈值。 MTBOX 进行如下处理: assign c_BistOT = 32'h1 << MCU2MT_BistOT[3:0] *2 即Bist超时阈值最大可配置为: 2^{30} , 最小可配置为1。
—	其它	—	保留。

4.1.11 MT_STATE0/1

偏移地址: 16'h700、16'h4c80

MT_STATE0/1 是状态寄存器，维护和软件只读，复位为“0x0”；

表 4-14: MT_STATE0 寄存器域的描述

名称	范围	类型	描述
SFSM3_STATE	[19:17]	RO,0x0	核心3的子状态机状态（同核心0）。
SFSM2_STATE	[16:14]	RO,0x0	核心2的子状态机状态（同核心0）。
SFSM1_STATE	[13:11]	RO,0x0	核心1的子状态机状态（同核心0）。
SFSM0_STATE	[10:8]	RO,0x0	核心0的子状态机状态。 0: 空闲状态； 1: 运行状态； 2: 降频状态； 3: 睡眠状态； 4: 升频状态； 5: 存储器自测试状态； 6: 切换等待状态（在升频状态到存储器自测试状态之间插入该状态以保证热复位能包住时钟使能信号）； 7: 深度睡眠状态（该状态为申威421处理器芯片新增，用于在所有核心都睡眠以后进一步降低功耗，此时会把一致性处理部件、三级Cache进行复位且时钟切换到低频状态。任意核心唤醒时，首先要判断芯片是否处于深睡眠，先从中恢复。）
MFMSM_STATE	[7:4]	RO,0x0	主状态机状态。 0: 上电复位状态； 1: 冷复位结束状态； 2: 配置1状态； 3: PLL升频状态； 4: 时钟切换1状态； 5: 结束复位状态； 6: 配置2状态； 7: BIST测试状态； 8: 配置3状态； 9: 初始化加载状态； 10: 预备运行状态； 11: 运行状态；

			12/13: 进入复位状态; 14/15: 时钟切换2状态。
RTPU_STATE	[3:0]	RO,0x0	串行接口状态机状态。 0: 空闲状态; 1: 请求接收状态; 2: 等待维护响应状态; 4: 发送维护响应状态; 8: 处理维护读命令状态; 9: 处理维护写命令状态; 10: 处理维护SCAN读命令状态; 11: 处理维护SCAN写命令状态; 12: 处理维护SROM加载命令状态; 其他保留。
—	其它	—	保留。

表 4-15: MT_STATE1 寄存器域的描述

名称	范围	类型	描述
OBS	[37:31]	RO,0x0	观测输出（仅选择稳定输出信号）。 [37]: 保留。 [36]: 保留。 [35]:核心 PLL 时钟锁定。 [34]: 互连 PLL 时钟锁定。 [33]:存控 PLL 时钟锁定。 [32]: PCIe-0 接收方链路连接建立。 [31]: PCIe-1 接收方链路连接建立。
FLAG	[30:27]	RO,0x0	对应 IOR: FlagReg;
BIST_INF	[26:23]	RO,0x0	[27]: BIST 测试完成, 高电平有效。 [26:25]: 测试结果。 2'b00, 表示测试无错; 2'b01, 表示有错可修复; 2'b10, 表示有错不可修复; 2'b11, 表示测试时间超时。 [24]: 指示在存储器 Debug 测试方式下, 发现错误而暂停存储器 Debug 测试, 此时可通过状态扫

			描获得具体的错误信息，高电平有效。
SYS_ERR	[22]	RO,0x0	系统错标志，高电平有效。
INT_DONE	[21]	RO,0x0	中断完成标志，高电平有效。
MT_ERR	[20]	RO,0x0	自定义维护接口错，高电平有效。
—	其它	—	保留。

4.1.12 MT_INT

偏移地址： 16'h800

MT_INT 是维护中断寄存器，维护只写。通过维护请求写该寄存器时，将对目标核心产生一次维护中断。该寄存器硬件实现时不真实存在，直接通过请求包向 IPU 发送（唤醒中断直接在 MCU 根据核心状态处理）。

表 4-16： MT_INT 寄存器域的描述

名称	范围	类型	描述
MT_INT_Type	[9:8]	WO,0	指示中断类型，0x：正常维护中断；10：维护睡眠中断；11：维护唤醒中断。
MT_INT_Tar	[3:0]	WO,0	指示中断目标编码指示。 0~3：核心0~3； 其它保留
—	其它	—	保留。

4.1.13 MT_INT_END

偏移地址： 16'h880

MT_INT_END 是维护中断结束寄存器，维护和软件可读写，用于标记核心已经处理完维护中断。复位为 0x1；

收到维护中断（包括睡眠中断、唤醒中断和一般维护中断）都清除标志位。

对于一般的维护中断由中断处理程序重新置该位；

对于睡眠中断，维护收到对应核心写 SleepDone 的请求时重新置该位；

对于唤醒中断，唤醒请求能够被处理或者被丢弃（已经唤醒或正在唤醒）或者被后面的睡眠请求覆盖，重新置该位；

该标志作为中断完成的结果输出给系统；

表 4-17： MT_INT_END 寄存器域的描述

名称	范围	类型	描述
MT_INT_END	[0]	RW,0x1	“1”时表明已经处理完一次维护中断。
—	其它	—	保留。

4.1.14 CPUID

偏移地址： 16'h900

CPUID 是 CPU 号寄存器，维护和软件都可读写，用于设置 CPU 编号信息。

表 4-18: CPUID 寄存器域的描述

名称	范围	类型	描述
CPUID	[55:0]	RW,0	CPU编号。
—	其它	—	保留。

4.1.15 SOFT_INF0~15

偏移地址： 16'h980~16'h1100

SOFT_INF0~15 是软件信息寄存器（共 5 个寄存器，每个核心对应一个寄存器），维护和软件都可读写，软件可使用该寄存器与维护接口交换信息。其中[63]为软件错误标志，写该位为 1 时，维护接口将置 IOR: GG_FAULT_STAT[SOFTERR]位为 1。

申威 421 处理器只有 SOFT_INF0~3 有意义，其它保留。

表 4-19: SOFT_INF0~15 寄存器域的描述

名称	范围	类型	描述
ERROR	[63]	RW,0	软件错误标志。
INF	[62:0]	RW,0	软件信息位。

4.1.16 LONG_TIME

偏移地址： 16'h1180

LONG_TIME 为长时钟寄存器，维护和软件都可读写，复位时初值为“0”。该寄存器用于长时钟计数。软件可以通过 IO 写设置该寄存器的初值，计数时钟为维护时钟。

表 4-20: LONG_TIME 寄存器域的描述

名称	范围	类型	描述
LONG_TIME	[63:0]	RW,0	长时钟计数器值。

4.1.17 LONG_PRESET

偏移地址： 16'h1200

LONG_PRESET 为长时钟预置寄存器，维护和软件都可读写，上电或冷复位预置为“0x61A8”，即对 25MHz 的维护时钟计数 1ms 的次数。每个维护时钟减“1”，减至“0”后循环至预置值。减至“0”时，将长时钟寄存器 LONG_TIME 加“1”。

表 4-21: LONG_PRESET 寄存器域的描述

名称	范围	类型	描述
LONG_PRESET	[31:0]	RW,0x61A8	长时钟预置值。
—	其它	—	保留。

4.1.18 TESTSEL

偏移地址： 16'h1280

控制 TESOUT 选择输出的选择端；维护和软件都可读，维护只写。

表 4-22: TESTSEL 寄存器域的描述

名称	范围	类型	描述
TESTSEL	[5:0]	RW,0x0	TESTOUT输出选择信号。
—	其它	—	保留。

4.1.19 IO_START

偏移地址： 16'h1300

IO_START 是外围接口使能寄存器，维护和软件都可读写。该寄存器用于故障隔离和容错，IO_START[1:0]分别对应 PCI-E1 和 PCI-E0。此二接口根据对应 IO_START 来完成对接收到的请求返回响应，如果对应位为“1”则按照正常的流程处理；否则直接返回读/写错误响应。

表 4-23: IO_START 寄存器域的描述

名称	范围	类型	描述
IO_START	[1:0]	RW,0x3	外围接口使能。
—	其它	—	保留。

4.1.20 PERST_N_PCIE0~1

偏移地址： 16'h1380、16'h2100

PERST_N_PCIE0~1 用于对 PCI-E 接口 (PCI-E0/PCI-E1) 进行复位, 可以实现由软件发起 PCIE 接口部分非粘连逻辑复位。维护和软件可读写, 软件自己控制复位信号的宽度, 写“0”表示复位, 写“1”表示复位结束。

表 4-24: PERST_N_PCIE0~1 寄存器域的描述

名称	范围	类型	描述
PERST_N	[0]	RW,0x1	PCI-E接口PERST复位。
—	其它	—	保留。

4.1.21 BUTTON_RST_N_PCIE0~1

偏移地址: 16'h1400、16'h2180

BUTTON_RST_N_PCIE0~1 是对 PCI-E 接口 (PCI-E0/PCI-E1) 的所有逻辑进行复位。维护和软件可读写, 软件自己控制复位信号的宽度, 写“0”表示复位, 写“1”表示复位结束。

表 4-25: BUTTON_RST_N_PCIE0~1 寄存器域的描述

名称	范围	类型	描述
BUTTON_RST_N	[0]	RW,0x1	PCI-E接口BUTTON_RST复位。
—	其它	—	保留。

4.1.22 MTRSPMISS

偏移地址: 16'h1500

MTRSPMISS[15:0]用于对维护接口接收到的响应与目前的请求 SMAF 编号是否相等进行记录, 维护和软件只读;

表 4-26: MTRSPMISS 寄存器域的描述

名称	范围	类型	描述
MTRSPMISS	[15:0]	RO,0x0	维护接口接收到的响应与目前的请求编号不匹配则该计数器加1。
—	其它	—	保留。

4.1.23 SLEEP_DONE0~15

偏移地址: 16'h1580~16'h1d00

写 SLEEP_DONE0~15 则指示了睡眠核心已经可以睡眠, 此时 MCU 将启动对应核心的睡眠流程。

SLEEP_DONE_x 只能软件可写（睡眠中断处理程序），维护不可写，并且该寄存器硬件实现时不真实存在，如果来自 IPU 的写操作则需要判断写对应寄存器的请求源核心号是否与寄存器的编号[0~15]对应，如果不对应则返回错误响应。

申威 421 处理器只有 SLEEP_DONE 0~3 有意义，其它保留。

表 4-27: SLEEP_DONE0~15 寄存器域的描述

名称	范围	类型	描述
SLEEP_DONE	[0]	WO,0x0	写该寄存器则MCU对对应核心进入睡眠流程。
—	其它	—	保留。

4.1.24 PLL_CHG_CNT

偏移地址： 16'h1d80

时钟切换时间间隔寄存器 PLL_CHG_CNT，维护可读写，软件只读。该寄存器用于在复位结束和复位开始时，根据该值决定不同核心、存控、和系统接口的时钟在低频和高频间切换的时间间隔。

表 4-28: PLL_CHG_CNT 寄存器域的描述

名称	范围	类型	描述
PLL_CHG_CNT	[31:0]	RW,65536	核心、存控、和系统接口的时钟在低频和高频间切换的时间间隔。
—	其它	—	保留。

4.1.25 CG_x_BIST_STAT

偏移地址： 16'h1e00~16'h1f80

CG_x_BIST_STAT 为 BIST 状态寄存器，维护和软件只读，该寄存器用于存放 CG 内各核心内阵列的 BIST 结果（Fail/Repair），以及 CTAG 的 BIST 结果。复位时初值为全“0”。

该寄存器 MCU 进行隐式写，其中各核心的 BIST 结果在 MCU 和 MTBOX 的接口上采用串行移位的方式；

申威 421 处理器只有 CG0_BIST_STAT 有意义，其它保留。

表 4-29: CG_x_BIST_STAT 寄存器域的描述

名称	范围	类型	描述
CPMbistinfo	[51:48]	RO,0	CPM BIST信息，从高到低对应TCache，TTag（每个阵列2位，高位表示Fail，低

			位表示Repair)。 [Fail、Repair]具体如下： [1:0]=2'b00，表示测试无错； [1:0]=2'b01，表示有错可修复； [1:0]=2'b10，表示有错不可修复； [1:0]=2'b11，表示测试时间超时。
Core3bistinfo	[47:36]	RO,0	Core3 BIST信息（同Core0）。 对于核组1而言，该域没有意义；
Core2bistinfo	[35:24]	RO,0	Core2 BIST信息（同Core0）。 对于核组1而言，该域没有意义；
Core1bistinfo	[23:12]	RO,0	Core1 BIST信息（同Core0）。 对于核组1而言，该域没有意义；
Core0bistinfo	[11:0]	RO,0	Core0 BIST信息，从高到低对应Scache， Stag, Dcache, Dtag（保留为零），Icache， Ctag（对于核组0该位保留） （每个阵列2位，高位表示Fail，低位表 示Repair）。 [Fail、Repair]具体如下： [1:0]=2'b00，表示测试无错； [1:0]=2'b01，表示有错可修复； [1:0]=2'b10，表示有错不可修复； [1:0]=2'b11，表示测试时间超时。
—	其它	—	保留。

4.1.26 BIST_STAT

偏移地址： 16'h2000

BIST_STAT 为 BIST 状态寄存器，维护和软件只读，该寄存器用于存放各核心和 CTAG 的 BIST 结果。复位时初值为全“0”。

表 4-30: BIST_STAT 寄存器域的描述

名称	范围	类型	描述
TC3_BIST_Info	[59:57]	RO,0	指示TC0的BIST结果（同TC0）；
TC2_BIST_Info	[56:54]	RO,0	指示TC0的BIST结果（同TC0）；
TC1_BIST_Info	[53:51]	RO,0	指示TC0的BIST结果（同TC0）；

TC0_BIST_Info	[50:48]	RO,0	<p>指示TC0的BIST结果； 该结果对应三级Cache所有阵列或指定阵列测试：</p> <p>[2]: Done; [1]: Fail; [0]: Repair。</p> <p>[Fail、Repair]具体如下： [1:0]=2'b00，表示测试无错； [1:0]=2'b01，表示有错可修复； [1:0]=2'b10，表示有错不可修复； [1:0]=2'b11，表示测试时间超时。</p>
Core15_BIST_Info	[47:45]	RO,0	指示核心15的BIST结果（同Core0）；
Core14_BIST_Info	[44:42]	RO,0	指示核心14的BIST结果（同Core0）；
Core13_BIST_Info	[41:39]	RO,0	指示核心13的BIST结果（同Core0）；
Core12_BIST_Info	[38:36]	RO,0	指示核心12的BIST结果（同Core0）；
Core11_BIST_Info	[35:33]	RO,0	指示核心11的BIST结果（同Core0）；
Core10_BIST_Info	[32:30]	RO,0	指示核心10的BIST结果（同Core0）；
Core9_BIST_Info	[29:27]	RO,0	指示核心9的BIST结果（同Core0）；
Core8_BIST_Info	[26:24]	RO,0	指示核心8的BIST结果（同Core0）；
Core7_BIST_Info	[23:21]	RO,0	指示核心7的BIST结果（同Core0）；
Core6_BIST_Info	[20:18]	RO,0	指示核心6的BIST结果（同Core0）；
Core5_BIST_Info	[17:15]	RO,0	指示核心5的BIST结果（同Core0）；
Core4_BIST_Info	[14:12]	RO,0	指示核心4的BIST结果（同Core0）；
Core3_BIST_Info	[11:9]	RO,0	指示核心3的BIST结果（同Core0）；
Core2_BIST_Info	[8:6]	RO,0	指示核心2的BIST结果（同Core0）；
Core1_BIST_Info	[5:3]	RO,0	指示核心1的BIST结果（同Core0）；
Core0_BIST_Info	[2:0]	RO,0	<p>指示核心0的BIST结果；该结果对应核心内所有阵列或指定阵列测试：</p> <p>[2]: Done; [1]: Fail; [0]: Repair。</p> <p>[Fail、Repair]具体如下： [1:0]=2'b00，表示测试无错； [1:0]=2'b01，表示有错可修复；</p>

			[1:0]=2'b10, 表示有错不可修复; [1:0]=2'b11, 表示测试时间超时。
—	其它	—	保留。

4.1.27 TAPSEL

偏移地址： 16'h2080

TAPSEL 为 TAP 控制器选择寄存器，维护和软件只读。用于记录芯片引脚的值。

表 4-31: TAPSEL 寄存器域的描述

名称	范围	类型	描述
TAPSEL	[1:0]	RO	用于控制 TAP 控制器的输出： 2'b00: 表示选择 PCI-E 0 的 TAP 控制器； 2'b01: 表示选择 PCI-E 1 的 TAP 控制器。 2'b1x: 表示选择 Tbox 的 TAP 控制器；
—	其它	—	保留。

4.1.28 ST_NEXT_FLAG

偏移地址： 16'h2300

状态机控制寄存器 ST_NEXT_FLAG。该寄存器不真实存在，维护只写，不关心写的内容。写该寄存器作为状态切换的条件，具体包括：配置状态 1 向等待 PLL 稳定状态切换、配置状态 2 向存储器自测试状态切换、配置 3 状态向 SROM 加载状态切换。在写该寄存器之前，维护已经完成对其它寄存器的配置。

表 4-32: ST_NEXT_FLAG 寄存器域的描述

名称	范围	类型	描述
ST_NEXT_FLAG	[0]	WO,0x0	写该寄存器作为主状态机从配置状态 2 向 SROM 状态切换的条件。
—	其它	—	保留。

4.1.29 SERR_CNTTH

偏移地址： 16'h2400

MCU 单错预警阈值，维护和主核可读写，复位时初值是 0xFFFF_FFFF；

表 4-33: SERR_CNTTH 寄存器域的描述

名称	范围	类型	描述
SERR_CNTTH	[31:0]	RW, 0xFFFFFFFF	MCU单错预警阈值。如果单错统计计数器的计数值等于该阈值且对应的单错报错使能位打开，则在SI_FAULT_STAT记录单错预警。 单错统计计数器共有 3 个：IPUSERR_CNT，PIU0SERR_CNT，PIU1SERR_CNT。
—	其它	—	保留。

4.1.30 IRU0SERR_CNT

偏移地址： 16'h2480

该寄存器统计：MCU 接收到 IRU0 方向请求和响应发生 ECC 单错的次数。维护和主核可读写。
该寄存器写清零。

表 4-34：IRU0SERR_CNT 寄存器域的描述

名称	范围	类型	描述
IRU0SERR_CNT	[31:0]	RW,0	来自 IRU0 的请求或响应报发生单错，则计数器加 1。
—	其它	—	保留。

4.1.31 INTPUSERR_CNT

偏移地址： 16'h4b80

该寄存器统计：MCU 接收到 INTPU 方向请求和响应发生 ECC 单错的次数。维护和主核可读写。
该寄存器写清零。

表 4-35：INTPUSERR_CNT 寄存器域的描述

名称	范围	类型	描述
INTPUSERR_CNT	[31:0]	RW,0	来自 INTPU 的请求或响应报发生单错，则计数器加 1。
—	其它	—	保留。

4.1.32 IRU1SERR_CNT

偏移地址： 16'h4c00

该寄存器统计：MCU 接收到 IRU1 方向请求和响应发生 ECC 单错的次数。维护和主核可读写。
该寄存器写清零。

表 4-36：IRU1SERR_CNT 寄存器域的描述

名称	范围	类型	描述
IRU1SERR_CNT	[31:0]	RW,0	来自 IRU1 的请求或响应报发生单错，则计数器加 1。
—	其它	—	保留。

4.1.33 PIUxSERR_CNT

偏移地址： 16'h2500~16'h2580

该寄存器统计：MCU 接收到 PIU 方向响应包发生 ECC 单错的次数。维护和主核可读写。该寄存器写清零。

表 4-37: PIUxSERR_CNT 寄存器域的描述

名称	范围	类型	描述
PIUxSERR_CNT	[31:0]	RW,0	来自 PIUx 的响应报发生单错，则计数器加 1。
—	其它	—	保留。

4.1.34 ERRRPT_EN

偏移地址： 16'h2600

该寄存器控制 MCU 各种报错使能。维护和主核可读写，复位初始为全“0”。具体定义如下表：

表 4-38: ERRRPT_EN 寄存器域的描述

名称	范围	类型	描述
Cnt IRU1SErr	[32]	RW,0	MCU 收到 IRU1 的响应有 ECC 单错预警报错报错使能。
CntINTPUSErr	[31]	RW,0	MCU 收到 INTPU 的响应有 ECC 单错预警报错报错使能。
FromINTPUResMErr	[30]	RW,0	INTPU 发向 MCU 的响应包有多错报错使能；
From IRU1ResMErr	[29]	RW,0	IRU1 发向 MCU 的响应包有多错报错使能；
ToINTPUCtrlErr	[28:27]	RW,0	维护发向 INTPU 的请求的控制错报错使能
ToIRU1CtrlErr	[26:25]	RW,0	维护发向 IRU1 的请求的控制错报错使能
IRU0_ADDRERR	[24]	RW,0	来自 IRU0 的请求携带的地址非法报错使能。
MCU_ADDRERR	[23]	RW,0	来自维护接口的请求携带的地址非法报错使能。
OMRB_FLASHADDRERR	[22]	RW,0	读写 FLASH 空间的地址错误报错使能。
OMCB_NotReadErr	[21]	RW,0	收到的 IRU0 OMCB 响应未能写入缓冲报错使能。

保留	[20]	RW,0	保留。
TBX_TI_TransErr	[19]	RW,0	维护接口错使能 (JTAG)。
MCU_SI_TransErr	[18]	RW,0	维护接口错使能 (自定义串口)。
From IRU0ResMErrEn	[17]	RW,0	IRU0 发向 MCU 的响应包的包头多错报错使能。
From IRU0ResCtrlErrEn	[16]	RW,0	IRU0 发向 MCU 的响应包控制错使能。
From IRU0ReqMErrEn	[15]	RW,0	IRU0 发向 MCU 的请求包多错报错使能。
From IRU0ReqCtrlErrEn	[14:13]	RW,0	IRU0 发向 MCU 的请求的包头和包尾信息不一致使能。
To IRU0ReqCtrlErr	[12]	RW,0	维护发向 IRU0 的请求的控制错使能。
ToMCUCtrlErrEn	[11]	RW,0	维护发向 MCU 的请求的控制错使能。
ToPIU1CtrlErrEn	[10:9]	RW,0	维护发向 PIU1 的请求的控制错使能。
ToPIU0CtrlErrEn	[8:7]	RW,0	维护发向 PIU0 的请求的控制错使能。
CntPIU1MErrEn	[6]	RW,0	MCU 收到 PIU1 ECC 多错报错使能。
CntPIU0MErrEn	[5]	RW,0	MCU 收到 PIU0 ECC 多错报错使能。
Cnt IRU0MErrEn	[4]	RW,0	MCU 收到 IRU0 ECC 多错报错使能。
CntPIU1SErrEn	[3]	RW,0	MCU 收到 PIU1 ECC 单错报错使能。
CntPIU0SErrEn	[2]	RW,0	MCU 收到 PIU0 ECC 单错报错使能。
Cnt IRU0SErrEn	[1]	RW,0	MCU 收到 IRU0 ECC 单错报错使能。
CtrlErrEn	[0]	RW,0	MCU 内部控制错报错使能。
—	其它	—	保留。

4.1.35 ERR_INF

偏移地址： 16'h2680

该寄存器记录 MCU 各种报错。维护和主核可读写，写清零。具体定义如下表：

表 4-39: ERR_INF 寄存器域的描述

名称	范围	类型	描述
CtrlErrCode	[38:33]	RW,0	表示出现控制错时，各源的有效位情况； [5]：表示 INTPU； [4]：表示 IRU1； [3]：表示 PIU1；

			[2]: 表示 IRU0; [1]: 表示 MCU; [0]: 表示 PIU0。
Cnt IRU1SErr	[32]	RW,0	MCU 收到 IRU1 的响应有 ECC 单错预警报错。
CntINTPUSErr	[31]	RW,0	MCU 收到 INTPU 的响应有 ECC 单错预警报错。
FromINTPUResMErr	[30]	RW,0	INTPU 发向 MCU 的响应包有多错;
From IRU1ResMErr	[29]	RW,0	IRU1 发向 MCU 的响应包有多错;
ToINTPUCtrlErr	[28:27]	RW,0	维护发向 INTPU 的请求的控制错,即上一个请求没有串行输出完成又发出一个请求,该错误也可能是维护串行接口复位造成。 维护发向 INTPU 的请求的控制错,即包头和包尾信息不一致(必须是单包)。
ToIRU1CtrlErr	[26:25]	RW,0	维护发向 IRU1 的请求的控制错,即上一个请求没有串行输出完成又发出一个请求,该错误也可能是维护串行接口复位造成。 维护发向 IRU1 的请求的控制错,即包头和包尾信息不一致(必须是单包)。
IRU0_ADDRERR	[24]	RW,0	来自 IRU0 的请求携带的地址非法。
MCU_ADDRERR	[23]	RW,0	来自维护接口的请求携带的地址非法。
OMRB_FLASHADDRERR	[22]	RW,0	读写 FLASH 空间的地址错误
OMCB_NotReadErr	[21]	RW,0	收到的 IPU OMCB 响应未能写入缓冲。
保留	[20]	RW,0	保留
TBX_TI_TransErr	[19]	RW,0	维护接口错(JTAG)。
MCU_SI_TransErr	[18]	RW,0	维护接口错(自定义串口)。
From IRU0ResMErr	[17]	RW,0	IRU0 发向 MCU 的响应包的包头有多错;
From IRU0ResCtrlErr	[16]	RW,0	多包的 OMCB 响应,在传输发数据包时维护接口被复位,此时 MCU 要把数据收齐,该报文被丢弃。
From IRU0ReqMErr	[15]	RW,0	IRU0 发向 MCU 的请求包有多错。
From IRU0ReqCtrlErr	[14:13]	RW,0	IRU0 发向 MCU 的请求的地址[39:36]与 MCU 的编址不匹配。 IRU0 发向 MCU 的请求的包头和包尾信息不一

			致（必须是单包）。
ToIRU0ReqCtrlErr	[12]	RW,0	多包的 IMRB 请求，在传输发数据包时维护接口被复位，此时 MCU 要把数据传完，但不保证数据的正确性。
ToMCU CtrlErr	[11]	RW,0	维护发向 MCU 的请求的控制错，即包头和包尾信息不一致（必须是单包）。
ToPIU1CtrlErr	[10:9]	RW,0	维护发向 PIU1 的请求的控制错，即上一个请求没有串行输出完成又发出一个请求，该错误也可能是维护串行接口复位造成。 维护发向 PIU1 的请求的控制错，即包头和包尾信息不一致（必须是单包）。
ToPIU0CtrlErr	[8:7]	RW,0	维护发向 PIU0 的请求的控制错，即上一个请求没有串行输出完成又发出一个请求，该错误也可能是维护串行接口复位造成。 维护发向 PIU0 的请求的控制错，即包头和包尾信息不一致（必须是单包）。
CntPIU1MErr	[6]	RW,0	MCU 收到 PIU1 的响应有 ECC 多错报错。
CntPIU0MErr	[5]	RW,0	MCU 收到 PIU0 的响应有 ECC 多错报错。
CntIRU0MErr	[4]	RW,0	MCU 收到 IRU0 的响应数据包有 ECC 多错报错。
CntPIU1SErr	[3]	RW,0	MCU 收到 PIU1 的响应有 ECC 单错预警报错。
CntPIU0SErr	[2]	RW,0	MCU 收到 PIU0 的响应有 ECC 单错预警报错。
CntIRU0SErr	[1]	RW,0	MCU 收到 IRU0 的请求或响应（包括数据包）有 ECC 单错预警报错。
CtrlErr	[0]	RW,0	MCU 内部控制错报错，同时收到多个响应。即同时收到来自 MCU、PIU1、PIU0 和 IPU 的响应。
—	其它	—	保留。

4.1.36 SCAN_WAIT_CFG

偏移地址： 16'h2780

SCAN_WAIT_CFG 是 SCAN 间隔控制寄存器，维护可读写，软件只读，用于控制两个 SCAN 操作（对字节而言，最小间隔是 5 拍处理一次）之间的间隔，复位为“0x0”。

表 4-40: SCAN_WAIT_CFG 寄存器域的描述

名称	范围	类型	描述
SCAN_WAIT_CFG	[2:0]	RW,0x0	3'h0: 0拍; 即5拍处理1个; 3'h1: 1拍; 即6拍处理1个; 3'h2: 2拍; 即7拍处理1个; 3'h3: 3拍; 即8拍处理1个; 3'h4: 4拍; 即9拍处理1个; 3'h5: 5拍; 即10拍处理1个; 3'h6: 6拍; 即11拍处理1个; 3'h7: 7拍; 即12拍处理1个。
—	其它	—	保留。

4.1.37 PC_BIST_STAT

偏移地址: 16'h2880

PC_BIST_STAT 为 BIST 状态寄存器, 维护和软件只读, 该寄存器用于存放 Page Cache 阵列的 BIST 结果 (Fail/Repair)。复位时初值为全“0”。

该寄存器 MCU 进行隐式写, 其中 Page Cache 的 BIST 结果在 MCU 和 MTBOX 的接口上采用串行移位的方式;

表 4-41: PC_BIST_STAT 寄存器域的描述

名称	范围	类型	描述
PC1bistinfo	[11:6]	RO,0	Page Cache 1 BIST信息 (同Page Cache 0)。
PC0bistinfo	[5:0]	RO,0	Page Cache 0 BIST信息, 从高到低对应 PCDATA 每个阵列2位, 高位表示Fail, 低位表示Repair)。 [5]: BISTDONE信号 [Fail、Repair]具体如下: [4:3]=2'b00, 表示测试无错; [4:3]=2'b01, 表示有错可修复; [4:3]=2'b10, 表示有错不可修复; [4:3]=2'b11, 表示测试时间超时。
—	其它	—	保留。

4.1.38 Corex_ERRRST_N

偏移地址： 16'h2900~16'h3080

核心错误信息清除寄存器，用于清除核心记录的错误信息，维护、软件均可读写,复位时初值为 0x1。

表 4-42: Corex_ERRRST_N 寄存器域的描述

名称	范围	类型	描述
ERRRST_N	[0]	RW,0x1	为“0”表示要清除核心记录的错误信息。
—	其它	—	保留。

4.1.39 SI_FAULT_STAT

偏移地址： 16'h3100

SI_FAULT_STAT 为系统接口总错状态寄存器，维护和软件可读写。该寄存器用于系统接口总错信息。复位时初值为全“0”。采上升沿登记；按位写“1”清除。

表 4-43: SI_FAULT_STAT 寄存器域的描述

名称	范围	类型	描述
ic_intr	[31]	RW1C,0	IIC发生中断
IRU1ERR	[30:28]	RW1C,0	由高到低分别表示IRU1产生的已纠正错、不可纠正错、控制错。
IPU0ERR	[27:25]	RW1C,0	由高到低分别表示IRU0产生的已纠正错、不可纠正错、控制错。
FlashOFault	[24]	RW1C,0	读取flash中Icache加载程序发生校验和错误
FlashWrFault	[23]	RW1C,0	对flash进行写入或者擦除的时候发生错误
PIU1_LinkRstFault	[22]	RW1C,0	指示PCIe链路因软件配置或链路断掉而发生了复位故障，对应PIU1。
PIU0_LinkRstFault	[21]	RW1C,0	指示PCIe链路因软件配置或链路断掉而发生了复位故障，对应PIU0。
IPU_FaultIntLost	[20]	RW1C,0	表示维护故障中断丢失。
IPU_NMIIntLost	[19]	RW1C,0	表示不可屏蔽故障故障中断丢失。
INTPUERR	[15:12]	RW1C,0	由高到低分别表示INTPU产生的已纠正错、不可纠正错、控制错和软件错。
PIU1ERR	[11:8]	RW1C,0	由高到低分别表示PIU1产生的已纠正错、不可纠正错、控制错和软件错。

PIU0ERR	[7:4]	RW1C,0	由高到低分别表示PIU0产生的已纠正错、不可纠正错、控制错和软件错。
MCUERR	[3:0]	RW1C,0	由高到低分别表示MCU产生的已纠正错、不可纠正错、控制错和软件错。
—	其它	—	保留。

4.1.40 SI_FAULT_EN

偏移地址： 16'h3180

SI_FAULT_EN 为系统接口总错使能寄存器，维护和软件都可读写。该寄存器用于控制系统接口总错信息。复位时初值为全“0”。

表 4-44: SI_FAULT_EN 寄存器域的描述

名称	范围	类型	描述
ic_intr_EN	[31]	RW1C,0	IIC发生中断
IRU1ERR_EN	[30:28]	RW1C,0	由高到低分别表示IRU1产生的已纠正错、不可纠正错、控制错。
IRU0ERR_EN	[27:25]	RW1C,0	由高到低分别表示IRU0产生的已纠正错、不可纠正错、控制错。
FlashOFault_EN	[24]	RW,0	读取flash中Icache加载程序发生校验和错误
FlashWrFault_EN	[23]	RW,0	对flash进行写入或者擦除的时候发生错误
PIU1_LinkRstFault_EN	[22]	RW,0	指示PCIe链路因软件配置或链路断掉而发生了复位故障，该位表示该故障的报错使能。对应PIU1。
PIU0_LinkRstFault_EN	[21]	RW,0	指示PCIe链路因软件配置或链路断掉而发生了复位故障，该位表示该故障的报错使能。对应PIU0。
IPU_FaultIntLost_EN	[20]	RW,0	表示故障中断丢失报错使能。
IPU_NMIIntLost_EN	[19]	RW,0	表示不可屏蔽故障故障中断丢失报错使能。
INTPUERR_EN	[15:12]	RW,0	由高到低分别表示INTPU产生的已纠正错、不可纠正错、控制错和软件错报错使能。
PIU1ERR_EN	[11:8]	RW,0	由高到低分别表示PIU1产生的已纠正错、不可纠正错、控制错和软件错报错使能。
PIU0ERR_EN	[7:4]	RW,0	由高到低分别表示PIU0产生的已纠正错、不可纠正错、控制错和软件错报错使能。
MCUERR_EN	[3:0]	RW,0	由高到低分别表示MCU产生的已纠正错、不可纠正错、控制错和软件错报错使能。

—	其它	—	保留。
---	----	---	-----

4.1.41 SI_FAULT_INT_EN

偏移地址： 16'h3200

SI_FAULT_INT_EN 为系统接口故障中断使能寄存器，维护和软件都可读写。该寄存器用于控制系统接口产生故障中断。复位时初值为全“0”。

表 4-45: SI_FAULT_INT_EN 寄存器域的描述

名称	范围	类型	描述
ic_intr_INT_EN	[31]	RW1C ,0	IIC发生中断
IRU1ERR_INT_EN	[30:28]	RW1C ,0	由高到低分别表示IRU1产生的已纠正错、不可纠正错、控制错。
IRU0ERR_INT_EN	[27:25]	RW1C ,0	由高到低分别表示IRU0产生的已纠正错、不可纠正错、控制错。
FlashOFault_INT_EN	[24]	RW,0	读取flash中Icache加载程序发生校验和错误
FlashWrFault_INT_EN	[23]	RW,0	对flash进行写入或者擦除的时候发生错误
PIU1_LinkRstFault_INT_EN	[22]	RW,0	指示PCIe链路因软件配置或链路断掉而发生了复位故障，该位表示该故障的报中断使能。对应PIU1。
PIU0_LinkRstFault_INT_EN	[21]	RW,0	指示PCIe链路因软件配置或链路断掉而发生了复位故障，该位表示该故障的报中断使能。对应PIU0。
IPU_FaultIntLost_INT_EN	[20]	RW,0	表示故障中断丢失报错使能。
IPU_NMIIntLost_INT_EN	[19]	RW,0	表示不可屏蔽故障故障中断丢失报错使能。
INTPUERR_INT_EN	[15:12]	RW,0	由高到低分别表示INTPU产生的已纠正错、不可纠正错、控制错和软件错故障中断使能。
PIU1ERR_INT_EN	[11:8]	RW,0	由高到低分别表示PIU1产生的已纠正错、不可纠正错、控制错和软件错故障中断使能。
PIU0ERR_INT_EN	[7:4]	RW,0	由高到低分别表示PIU0产生的已纠正错、不可纠正错、控制错和软件错故障中断使能。
MCUERR_INT_EN	[3:0]	RW,0	由高到低分别表示MCU产生的已纠正错、不可纠正错、控制错和软件错故障中断使能。
—	其它	—	保留。

4.1.42 CLK_SEL_PCIE0~1

偏移地址： 16'h3280、16'h3300

PCI-E 时钟选择寄存器，维护和软件可读写；

注：PERST_N 不影响该 IOR，且 BUTTON_RST_N 和 PERST_N 的复位值都是无效（高）；

表 4-46: CLK_SEL_PCIE0~1 寄存器域的描述

名称	范围	类型	描述
CLK_SEL_PCIE	[0]	RW,0x1	该位为“1”PIU选择PCLK，否则选择MCLK。
—	其它	—	保留。

4.1.43 FIFO_SYNSEL

偏移地址： 16'h3400

FIFO_SYNSEL 是 SBOX、CPM 和 IPU 模块中异步 FIFO 读/写地址同步的级数配置寄存器，维护和软件可读写。

表 4-47: FIFO_SYNSEL 寄存器域的描述

名称	范围	类型	描述
SYNSEL1	[7:4]	RW,0x1	[7:6]: 保留； [5:4]: CPM或IPU内异步FIFO读/写地址同步的级数： 2'b00: 2级； 2'b01: 3级； 2'b10: 4级； 2'b11: 5级。 缺省为2'b01。
SYNSEL0	[3:0]	RW,0x2	[3:2]: 保留； [1:0]: SBox内异步FIFO读/写地址同步的级数： 2'b00: 2级； 2'b01: 3级； 2'b10: 4级； 2'b11: 5级。 缺省为2'b10。
—	其它	—	保留。

4.1.44 CPU_INFO

偏移地址： 16'h3480

CPU_INFO 寄存器用于记录产品信息。维护可读写，软件只读。

表 4-48: CPU_INFO 寄存器域的描述

名称	范围	类型	描述
MANUFACTURE_ID	[63:0]	RO,0x0	该字段用于记录生产批次号等信息。

4.1.45 FlagReg

偏移地址： 16'h3580

维护和软件可读写。可由任意核心进行软件读写。

表 4-49: FlagReg 寄存器域的描述

名称	范围	类型	描述
Flag3	[3]	RW,0x0	
Flag2	[2]	RW,0x0	
Flag1	[1]	RW,0x0	
Flag0	[0]	RW,0x0	
—	其它	—	保留。

4.1.46 OP_MASK

偏移地址： 16'h2380

维护操作屏蔽控制寄存器寄存器 OP_MASK。该寄存器维护可读写，软件只读。

表 4-50: OP_MASK 寄存器域的描述

名称	范围	类型	描述
MemWrMask	[3]	WO,0x0	维护写存储器被屏蔽。
MemRdMask	[2]	WO,0x0	维护读存储器被屏蔽。
IOWrMask	[1]	WO,0x0	维护写I0寄存器被屏蔽。
IORdMask	[0]	WO,0x0	维护读I0寄存器被屏蔽。
—	其它	—	保留。

4.1.47 BIST_GOON

偏移地址： 16'h1480

Debug 测试控制寄存器 BIST_GOON。该寄存器不真实存在，维护只写，不关心写的内容。Debug 测试扫出出错信息后，写该寄存器继续进行测试。

表 4-51: BIST_GOON 寄存器域的描述

名称	范围	类型	描述
BIST_GOON	[0]	WO,0x0	Debug测试扫出出错信息后，写该寄存器继续进行测试。
—	其它	—	保留。

4.1.48 WAKEUP_CTL

偏移地址： 16'h1480

外部唤醒中断控制寄存器 WAKEUP_CTL。该寄存器维护可读写，软件只读。

表 4-52: WAKEUP_CTL 寄存器域的描述

名称	范围	类型	描述
WAKEUP_L	[4]	RO	用于观测芯片引脚WAKEUP_L。
TYPE_SEL	[1]	RW,0x0	有效类型选择。 0: 表示低电平有效。 1: 表示高电平有效。
MASK_EN	[0]	RW,0x1	屏蔽使能。该位为“1”不响应中断。
—	其它	—	保留。

4.1.49 NMI_CTL

偏移地址： 16'h3600

外部不可屏蔽中断控制寄存器 NMI_CTL。该寄存器维护可读写，软件只读。

表 4-53: NMI_CTL 寄存器域的描述

名称	范围	类型	描述
NMI_L	[4]	RO	用于观测芯片引脚NMI_L。
NMI_CTL	[3:2]	RW,0x3	有效类型选择。 2`b00: 表示上升沿有效；

			2`b01: 表示下降沿有效; 2`b10: 表示高电平有效; 2`b11: 表示低电平有效。
MASK_EN	[0]	RW,0x1	屏蔽使能。该位为“1”不响应中断。
—	其它	—	保留。

4.1.50 PIUPLL_CNT

偏移地址: 16`h3680

PIU PLL 稳定等待阈值寄存器 PIUPLL_CNT。该寄存器维护可读写, 软件只读。

表 4-54: PIUPLL_CNT 寄存器域的描述

名称	范围	类型	描述
PLL_CHG_CNT	[31:0]	RW,655 36	PIU PLL稳定的等待时间阈值。 状态机在PLL升频状态等待计数器满足(计数阈值由该寄存器设置), 且等待PIU的Clock Stable信号(该信号为电平信号)。
—	其它	—	保留。

4.1.51 FlashINFO

偏移地址： 16'h3800

维护和软件只读，在初始化过程中由 MIU 隐式从 flash 中读出配置信息写入该寄存器。

表 4-55: FlashINFO 寄存器域的描述

名称	范围	类型	描述
CPURWStartAddr	[29:20]	RO,0x1	CPU 即可读又可写区的 flash 空间起始地址（以 32KB 为单位），为 i 表示第 0~i-1 个 32KB 不允许 CPU 写，第 i、i+1...个 32KB 可被 CPU 写。
CPUROStartAddr	[19:10]	RO,0x1	CPU 只可读不可写区的 flash 空间起始地址（以 32KB 为单位），为 i 表示第 0~i-1 个 32KB 不允许 CPU 读，第 i、i+1...个 32KB 可被 CPU 读。
FlashSize	[9:0]	RO,0x1	Flash 存储容量（以 32KB 为单位）
—	其它	—	保留。

4.1.52 RTPUSROMCNT

偏移地址： 16'h3880

该寄存器只读，用于维护实际处理的 Srom 加载命令

表 4-56: RTPUSROMCNT 寄存器域的描述

名称	范围	类型	描述
SromLack	[63]	R,0x0	发生 Srom 漏传的情况； Srom 加载地址大于期望地址；
SromRetry	[62]	R,0x0	发生 Srom 重传的情况； Srom 加载地址小于期望地址；
SromCnt	[15:0]	R,0x0	Srom 加载命令计数
—	其它	—	保留。

4.1.53 SICMDCNT

偏移地址： 16'h3900

该寄存器只读，用于记录维护接口上的 Srom 加载命令和维护接口复位命令；

表 4-57: SICMDCNT 寄存器域的描述

名称	范围	类型	描述
MResetCnt	[47:32]	R,0x0	维护接口复位命令计数
SromCnt	[15:0]	R,0x0	Srom加载命令计数
—	其它	—	保留。

4.1.54 FLASHERASE4K

偏移地址： 16'h3980

非真实存在寄存器，写该地址产生一个 FLASH 4K 擦除操作，该寄存器地址只写，软件和维护均可写。写请求所携带的数据[35:0]位为擦除操作起始地址。

该地址的含义见 Flash 空间说明（1.2.5.3 节）。

注：擦除地址来自于该寄存器，该地址与 IO 访问地址映射不同，直接用 IOR[23:0]作为 Flash 地址。

4.1.55 FLASHERASE32K

偏移地址： 16'h3a00

非真实存在寄存器，写该地址产生一个 FLASH 32K 擦除操作，该寄存器地址只写，软件和维护均可写。写请求所携带的数据[35:0]位为擦除操作起始地址。

该地址的含义见 Flash 空间说明（1.2.5.3 节）。

注：擦除地址来自于该寄存器，该地址与 IO 访问地址映射不同，直接用 IOR[23:0]作为 Flash 地址。

4.1.56 FLASHERASE64K

偏移地址： 16'h3a80

非真实存在寄存器，写该地址产生一个 FLASH 64K 擦除操作，该寄存器地址只写，软件和维护均可写。写请求所携带的数据[35:0]位为擦除操作起始地址。

该地址的含义见 Flash 空间说明（1.2.5.3 节）。

注：擦除地址来自于该寄存器，该地址与 IO 访问地址映射不同，直接用 IOR[23:0]作为 Flash 地址。

4.1.57 FLASH_ERASE_ALL

偏移地址： 16'h3b00

非真实存在寄存器，写该地址产生一个 FLASH 全片擦除操作，该寄存器地址只写，软件和维
护均可写。

4.1.58 INTERFACE_EN

偏移地址： 16'h4280

INTERFACE_EN 控制 jtag 接口，自定义串口的使能信号，当接口关闭时进行访问会返回错误响
应。

表 4-58: INTERFACE_EN 寄存器域的描述

名称	范围	类型	描述
MCU_EN	[1]	RW,0x1	自定义串口使能，为1打开，为0关闭。
JTAG_EN	[0]	RW,0x1	JTAG接口使能，为1打开，为0关闭。
—	其它	—	保留。

4.1.59 MCU_DEBUG0~7

偏移地址： 16'h4500~16'h4880

MCU_DEBUG 用来调试使用，维护和软件可读写。

表 4-59: MCU_DEBUG 寄存器域的描述

名称	范围	类型	描述
MCU_DEBUG	[63:0]	RW,0x0	
—	其它	—	保留。

4.1.60 MC_CONFIG

偏移地址： 16'h4900

MC_CONFIG 用来配置主存容量及交叉模式，维护和软件可读写。

表 4-60: MC_CONFIG 寄存器域的描述

名称	范围	类型	描述
MC0CAP	[7:5]	RW,0x4	该信号表示一路存控接口的主存容量，根据使用约定要

			求每路存控的容量必须相同，所以在 MCU 设置该域。 具体含义如下： 0: 保留； 1: 保留； 2: 1GB； 3: 2GB； 4: 4GB； 5: 8GB； 6: 16GB； 7: 32GB； 该信号由MCU IOR输出同时驱动四个CPM。CPM根据MC_ONLINE自行得到核组主存容量。
CGCrossMode	[1:0]	RW,0x0	核组地址交叉模式。 2'b00: 低中地址+Hash； 2'b01: Cache 行交叉； 2'b10: 8KB 页面交叉； 2'b11: 连续编址。 该信号同时驱动四个CPM。CPM中的具体使用表4-61。 申威421处理器只能配置为连续编址。
—	其它	—	保留。

表 4-61: CrossMode 域含义

CGCrossMode [1:0]	芯片工作模式		
	CGOnline= 4'b0001/4'b0111	CGOnline=4'b0011	CGOnline=4'b1111
0 (默认值, 低中地址哈希交叉)	/	DstCGId[1]=1'b0 DstCGId[0]= Addr[7] xor Addr[13]	DstCGId= Addr[8:7] xor {Addr[18],Addr[13]}
1 (Cache 地址交叉)	/	DstCGId[1]=1'b0 DstCGId[0]= Addr[7]	DstCGId= Addr[8:7]
2 (8KB 页面交叉)	/	DstCGId[1]=1'b0 DstCGId[0]= Addr[13]	DstCGId= Addr[14:13]
3	DstCGId=2'h0	DstCGId[1]=1'b0	DstCGId=地址最高 2 位,

(连续编址)		DstCGId[0]=地址最高位, 与核组主存容量有关	与核组主存容量有关
--------	--	--------------------------------	-----------

4.1.61 CoreSleepCnt

偏移地址： 16'h4980

CoreSleepCnt 用来控制 SI2CGx_CoreRunning[3:0]和 SI2CGx_HotReset_N[3:0]的间隔。维护和软件可读写。

表 4-62: CoreSleepCnt 寄存器域的描述

名称	范围	类型	描述
CoreSleepCnt	[11:0]	RW,0x800	
—	其它	—	保留。

4.1.62 DeepSleepFlag

偏移地址： 16'h4a00

DeepSleepFlag 用来控制芯片是否进入深度睡眠。维护和软件可读写。

表 4-63: DeepSleepFlag 寄存器域的描述

名称	范围	类型	描述
DeepSleepFlag	[0]	RW,0x0	当设置为1时，在16个核心都睡眠时，如果配置为深睡眠，则使三级Cache和CPM进入复位状态，时钟切换到低频状态（时钟降频为同时进行）。
—	其它	—	保留。

4.1.63 flash_wrprotect_data

偏移地址： 16'h4a80

flash_wrprotect_data 用来配置 Flash 写保护指令所携带的写数据。维护和软件可读写。

表 4-64: flash_wrprotect_data 寄存器域的描述

名称	范围	类型	描述
flash_wrprotect_data	[7:0]	RW,0x0	
—	其它	—	保留。

4.1.64 CHIP_ID 寄存器

偏移地址： 16'h4d00

CHIP_ID 寄存器维护和软件可读。

表 4-65: CHIP_ID 寄存器域的描述

名称	范围	类型	描述
芯片架构系列号	[33:30]	RO, 0x6	架构序列号：区分不同系列，16个系列，可以使用30年； 0x00:申威1系列 0x01:申威2系列
芯片封装架构号	[29:26]	RO	区分不同封装或不同裁减的芯片：通过封装基板来设置，增加4根引脚！根据该系列芯片中不同架构类型顺序编号，对于一次流片多种不同封装形态（核心、存控、IO配置）的芯片，可以根据配置引脚或寄存器的值，由MCU自动译码得到芯片架构号。 对应CHIP_MODE_H[2:0]引脚： 3'b0xx：四核模式。强制关闭核组 1、核组 2 和核组 3（包括核心与对应的 2 路主存和三级 Cache） 3'b100：八核模式经济模式。强制关闭核组 2 和核组 3（包括核心与对应的 2 路主存和三级 Cache）以及核组 0 和核组 1 的 MM1； 3'b101：八核模式。强制关闭核组 2 和核组 3（包括核心与对应的 2 路主存和三级 Cache）。 3'b110：16 核经济模式。强制关闭每个核组的 MM1； 3'b111：16 核全功能模式。
主核心系列号	[25:18]	RO, 0x31	高四位为大系列号，低四位为系列微调号： 0x0X 无通用核心 0x1X Core1 0x2X Core2 0x30 Core3 0X31 Core3A
从核心系列号	[17:10]	RO,0x	高四位为大序列号，低四位为系列微调号：

		0	0x00 无从核心 0x1X Score1
流片序列号	[9:2]	RO, 0x6	高四位为流片号(指示此工艺的使用次数以及工艺分支的区别, 如G工艺还是LL工艺等), 低四位为工艺代号: 选择有0.35um、0.18um、0.13um、90nm、65nm、45/40nm、32/28nm、22/20nm、16/14nm、10nm、7nm、5nm、3nm、特殊工艺1、特殊工艺2 0x0 130nm 0x1 90nm 0x2 65nm 0x3 45nm 0x4 40nm 0x5 32nm 0x6 28nm 0x7 16nm 0x8 14nm
流片号	[1:0]	RO, 0x0	区分不同的流片, 为“00 ₂ ”指示第一次流片, 为“01 ₂ ”指示第二次流片, 如此类推。
--	其它	RO	保留

4.1.65 App_Ltssm_En

偏移地址: 16'h4d80

App_Ltssm_En 用来控制 PCIE 接口的 App_ltssm_Enable 信号。维护和软件可读写。

表 4-66: flash_wrprotect_data 寄存器域的描述

名称	范围	类型	描述
PCIE1_App_ltssm_Enable	[1]	RW,0x1	PCI-E1接口App_ltssm_Enable信号控制, 默认值为1
PCIE0_App_ltssm_Enable	[0]	RW,0x1	PCI-E0接口App_ltssm_Enable信号控制, 默认值为1
—	其它	—	保留。

4.2 I2C 接口的 IO 寄存器

4.2.1 基地址说明

表 4-67: I2C 基地址

模块	核心视角(PA[47:0])	维护视角(PA[39:0])	备注
MCU	0x8054,0000,0000	0xd4,0000,0000	

注：各寄存器的偏移地址（相对于零地址）见具体寄存器说明。

4.2.2 IIC IC_CON 控制寄存器

偏移地址：0x0000

表 4-68: IIC IC_CON 寄存器域的描述

名称	范围	类型	描述
IC_SLAVE_DISABLE	[6]	RW,0x1	Slave有效寄存器，指示IP作为Slave是否有效。 0: 有效 1: 无效
IC_RESTART_EN	[5]	RW,0x1	I ² C Restat有效，作为Master是否发送重新启动标志。 0: 无效 1: 有效
IC_10BITADDR_MASTER/ IC_10BITADDR_MASTER_rd_only	[4]	RO,0x1	Master10位地址和7位地址配置。 由于I2C_DYNAMIC_TAR_UPDATE配置为1，这位是只读属性。配置为0为可读写属性。 0: Master寻址7位地址 1: Master寻址10位地址
IC_10BITADDR_SLAVE	[3]	RW,0x1	Slave10位地址和7位地址配置。 0: Slave响应7位地址 1: Slave响应10位地址
SPEED	[2:1]	RW,0x3	I ² C总线工作速度模式。 2'b01: 标准模式（100kb/s） 2'b10: 快速模式（400kb/s） 2'b11: 高速模式（3.4mb/s）
MASTER_MODE	[0]	RW,0x1	Master模式，控制IP是否作为Master使用。

			0: master无效 1: master有效
—	其它	—	保留。

4.2.3 IIC IC_TAR 目标地址寄存器

偏移地址：0x00200

表 4-69: IIC IC_TAR 寄存器域的描述

名称	范围	类型	描述
IC_10BITADDR_MASTER	[12]	RW,0x1	Master 寻址 10 位 / 7 位地址，此位只有当 I2C_DYNAMIC_TAR_UPDATE 为 1 时有效。 0: 7 位地址 1: 10 位地址
SPECIAL	[11]	RW,0x0	是否发送广播。 0: 忽略 GC_OR_START 里的配置，正常使用 IC_TAR 作为寻址对象地址。 1: 根据 GC_OR_START 里的配置执行特殊 I2C 命令。
GC_OR_START	[10]	RW,0x0	当 SPECIAL 设置为 1 时，这位来选择是发送 call 还是发送起始字节。 0: 发送 call，将会一直发送广播，直到 SPECIAL 被清零。 1: 发送起始字节
IC_TAR	[9:0]	RW,0x5 5	目标地址
—	其它	—	保留。

4.2.4 IIC IC_SAR 从设备地址寄存器

偏移地址：0x0400

表 4-70: IIC IC_SAR 寄存器域的描述

名称	范围	类型	描述
IC_SAR	[9:0]	RW,0x5 5	此寄存器保存了 Slave 地址，作为 Slave 时有意义。如果使用 7 位地址，IC_SAR[6:0] 有意义。

			只有当IC_ENABLE为0时才能更新此寄存器。
—	其它	—	保留。

4.2.5 IIC IC_HS_MADDR 高速master模式编码地址寄存器

偏移地址：0x0600

表 4-71：IIC IC_HS_MADDR 寄存器域的描述

名称	范围	类型	描述
IC_HS_MAR	[2:0]	RW,0x1	高速master模式编码： 只有在高速模式下此寄存器的值有意义。高速模式下Master编码为一个八位的编码（{00001, IC_HS_MAR}），也就是最多有8个Master。 当IC_ENABLE为0时可以更新此寄存器。其他时刻修改寄存器值无效。
—	其它	—	保留。

4.2.6 IIC IC_DATA_CMD发送和接收数据命令寄存器

偏移地址：0x0800

表 4-72：IIC IC_DATA_CMD 寄存器域的描述

名称	范围	类型	描述
CMD	[8]	RW,0x0	读写命令，这一位控制是执行读或者写操作。当IP作为Slave时，这位并不控制传输方向；只有当IP作为Master时，这位才指示了传输方向。 1：读 0：写 当IP作为Slave-Receiver时，并不关心这一位； 当IP作为Slave-Transmitter时，为0表示需要传输DAT。
DAT	[7:0]	RW,0x0	数据将要被发送到I2C总线上。当要执行读操作，会忽略此寄存器内的值，将会给出保留在IP接口上的数据。
—	其它	—	保留。

4.2.7 IIC IC_SS_HCNT 标准速度SCL高电平计数寄存器

偏移地址：0x0a00

表 4-73：IIC IC_SS_HCNT 寄存器域的描述

名称	范围	类型	描述
IC_SS_SCL_HCNT	[15:0]	RW,0x1 90	标准模式下SCL高电平计数：当IC_ENABLE为0可以更新此寄存器。 注意： 配置的最小值为6；配置的值不能超过65525，即 $2^{16}-1-10$ 。 需要留10个周期用来控制总线跳转到初始状态。 当IC_HC_COUNT_VALUES为1时为只读
—	其它	—	保留。

4.2.8 IIC IC_SS_LCNT 标准速度SCL低电平计数寄存器

偏移地址：0x0c00

表 4-74：IIC IC_SS_LCNT 寄存器域的描述

名称	范围	类型	描述
IC_SS_SCL_LCNT	[15:0]	RW,0x1 d6	标准模式下SCL低电平计数：当IC_ENABLE为0可以更新此寄存器。 注意： 配置的最小值为8； 当IC_HC_COUNT_VALUES为1时为只读
—	其它	—	保留。

4.2.9 IIC IC_FS_HCNT 快速SCL高电平计数寄存器

偏移地址：0x0e00

表 4-75：IIC IC_FS_HCNT 寄存器域的描述

名称	范围	类型	描述
IC_FS_SCL_HCNT	[15:0]	RW,0x3c	快速模式下SCL高电平计数：当IC_ENABLE为0可以更新此寄存器。

			注意： 配置的最小值为6； 当IC_HC_COUNT_VALUES为1时为只读
—	其它	—	保留。

4.2.10 IIC IC_FS_LCNT 快速SCL低电平计数寄存器

偏移地址：0x1000

表 4-76：IIC IC_FS_LCNT 寄存器域的描述

名称	范围	类型	描述
IC_FS_SCL_LCNT	[15:0]	RW,0x8 2	快速模拟下SCL低电平计数：当IC_ENABLE为0可以更新此寄存器。 注意： 配置的最小值为8； 当IC_HC_COUNT_VALUES为1时为只读
—	其它	—	保留。

4.2.11 IIC IC_HS_HCNT 高速SCL高电平计数寄存器

偏移地址：0x1200

表 4-77：IIC IC_HS_HCNT 寄存器域的描述

名称	范围	类型	描述
IC_HS_SCL_HCNT	[15:0]	RW,0xc	高速模拟下SCL高电平计数：当IC_ENABLE为0可以更新此寄存器。 注意： 配置的最小值为6； 当IC_HC_COUNT_VALUES为1时为只读
—	其它	—	保留。

4.2.12 IIC IC_HS_LCNT 高速SCL低电平计数寄存器

偏移地址：0x1400

表 4-78：IIC IC_HS_LCNT 寄存器域的描述

名称	范围	类型	描述
IC_HS_SCL_LCNT	[15:0]	RW,0x2 0	高速模拟下SCL低电平计数：当IC_ENABLE为0可以更新此寄存器。 注意： 配置的最小值为8； 当IC_HC_COUNT_VALUES为1时为只读
—	其它	—	保留。

4.2.13 IIC IC_INTR_STAT 中断状态寄存器

偏移地址：0x1600

表 4-79：IIC IC_INTR_STAT 寄存器域的描述

名称	范围	类型	描述
R_GEN_CALL	[11]	RO,0x0	此寄存器为IC_RAW_INTR_STAT通过屏蔽寄存器IC_INTR_MASK之后生成的中断信号。对应信号含义同IC_RAW_INTR_STAT。
R_START_DET	[10]	RO,0x0	
R_STOP_DET	[9]	RO,0x0	
R_ACTIVITY	[8]	RO,0x0	
R_RX_DONE	[7]	RO,0x0	
R_TX_ABRT	[6]	RO,0x0	
R_RD_REQ	[5]	RO,0x0	
R_TX_EMPTY	[4]	RO,0x0	
R_TX_OVER	[3]	RO,0x0	
R_RX_FULL	[2]	RO,0x0	
R_RX_OVER	[1]	RO,0x0	
R_RX_UNDER	[0]	RO,0x0	
—	其它	—	

4.2.14 IIC IC_INTR_MASK 中断掩码寄存器

偏移地址：0x1800

表 4-80：IIC IC_INTR_MASK 寄存器域的描述

名称	范围	类型	描述
----	----	----	----

M_GEN_CALL	[11]	RW,0x1	用于屏蔽对应中断。1表示该中断有效；0表示将会屏蔽对应中断。
M_START_DET	[10]	RW,0x1	
M_STOP_DET	[9]	RW,0x1	
M_ACTIVITY	[8]	RW,0x1	
M_RX_DONE	[7]	RW,0x1	
M_TX_ABRT	[6]	RW,0x1	
M_RD_REQ	[5]	RW,0x1	
M_TX_EMPTY	[4]	RW,0x1	
M_TX_OVER	[3]	RW,0x1	
M_RX_FULL	[2]	RW,0x1	
M_RX_OVER	[1]	RW,0x1	
M_RX_UNDER	[0]	RW,0x1	
—	其它	—	

4.2.15 IIC IC_RAW_INTR_STAT 原始中断状态寄存器

IC_INTR_STAT 内的中断状态是包括 IC_RAW_INTR_STAT 被屏蔽之后结果；IC_RAW_INTR_STAT 是原始的中断状态。有的应用需要监测原始的中断状态，具体见第 5 章软件流程。

偏移地址：0x1a00

表 4-81：IIC IC_RAW_INTR_STAT 寄存器域的描述

名称	范围	类型	描述
GEN_CALL	[11]	RO,0x0	当接收到广播地址并给出响应的时候置1，直到以下条件中的某个成立的时候才会被清除： 1. I2C总线关闭即IC_ENABLE为0； 2. 寄存器IC_CLR_GEN_CALL[0]为1； IP将接收到的数据存入Rx buffer。
START_DET	[10]	RO,0x0	指示I2C总线上是否有启动标志或者重新启动标志发生，忽略本IP是作为Master还是Slave；用于debug模式。
STOP_DET	[9]	RO,0x0	指示I2C总线上是否有停止标志发生，忽略本IP是作为Master还是Slave；用于debug模式。
ACTIVITY	[8]	RO,0x0	当监测到总线状态为activity时这一位置1直到清除。有4种方法清除这一位： 1. I2C总线关闭，即IC_ENABLE为0；

			<ol style="list-style-type: none"> 2. 寄存器IC_CLR_ACTIVITY被置1; 3. 寄存器IC_CLR_INTR被置1; 4. 系统复位 <p>这4种方法中的某一种有效就会清除这一位, 否则即使总线进入idle状态也不会清除它。</p>
RX_DONE	[7]	RO,0x0	当IP作为Slave-Transmitter, 在Master给出NACK时这一位会置1, 代表着传输完成。
TX_ABRT	[6]	RO,0x0	当这一位置1, 表示着这次传输发送数据失败。既可能发生在Master也可能发生在Slave, 但它们都作为发送方。发生数据失败的原因会记录进寄存器IC_TX_ABRT_SOURCE。注意: 当这一位置1表示传输失败时IP会刷新或者清空所有的FIFO (无论是接收方还是发送方)。
RD_REQ	[5]	RO,0x0	当IP接收到总线上其他Mater发出的读数据请求时, 这一位被置1 (作为Slave-Transmitter)。通过这一个中断告之软件需要将数据写入寄存器IC_DATA_CMD。RD_REQ通过读寄存器IC_CLR_RD_REQ (IC_CLR_RD_REQ为1时, 表示数据已准备好) 来清零。RD_REQ为1时会保持拉低时钟, 使总线处于等待状态。
TX_EMPTY	[4]	RO,0x0	当发送数据缓冲 (transmit buffer) 中有效条目小于等于阈值 (寄存器IC_TX_TL) 时中断置1; 当有效条目数据大于阈值时, 硬件会自动将它清零。
TX_OVER	[3]	RO,0x0	在数据超过发送缓冲的深度时, 此中断会被置1。当IP的状态机在idle状态且I2C总线关闭的时候会清空此寄存器。
RX_FULL	[2]	RO,0x0	当接收数据缓冲 (receiver buffer) 中有效条目数大于或者等于阈值 (寄存器IC_RX_TL) 时中断置1, 当数据量回到阈值之内时, 硬件会自动将它清零。当总线关闭的时候, RX FIFO会被刷新并保持在复位状态。
RX_OVER	[1]	RO,0x0	当接收缓冲的数据超过接收缓冲的深度时, 此位会被置1。超出的数据部分会丢失。当I2C总线关闭的时候会清空此寄存器。
RX_UNDER	[0]	RO,0x0	当接收缓冲空的时候, 软件去读接收缓冲内的数据, 会将此位置1。即使关闭I2C总线或者IP处于idle状态都会保持。当ic_en为0时清零。

—	其它	—	保留。
---	----	---	-----

4.2.16 IIC IC_RX_TL 接收FIFO阈值寄存器

偏移地址：0x1c00

表 4-82：IIC IC_RX_TL 寄存器域的描述

名称	范围	类型	描述
RX_TL	[0]	RW,0x0	接收FIFO阈值：用于控制发生RX_FULL中断的标准。取值范围0~255，硬件上不允许使阈值大于接收缓冲的深度。超过的话会自动使用缓冲深度作为阈值。
—	其它	—	保留。

4.2.17 IIC IC_TX_TL 发送FIFO阈值寄存器

偏移地址：0x1e00

表 4-83：IIC IC_TX_TL 寄存器域的描述

名称	范围	类型	描述
TX_TL	[0]	RW,0x0	发送FIFO阈值：用于控制发生TX_EMPTY中断的标准。取值范围0~255，硬件上不允许使阈值大于发送缓冲的深度。超过的话会自动使用缓冲深度作为阈值。
—	其它	—	保留。

4.2.18 IIC IC_CLR_INTR 中断清除寄存器

偏移地址：0x2000

表 4-84：IIC IC_CLR_INTR 寄存器域的描述

名称	范围	类型	描述
CLR_INTR	[0]	R,0x0	中断清除：被置一时清除联合中断，所有独立中断，和寄存器IC_TX_ABRT_SOURCE。不用于清除硬件可清除中断，而用于清除软件可清除中断。
—	其它	—	保留。

4.2.19 IIC IC_CLR_RX_UNDER 清除RX_UNDER中断寄存器

偏移地址：0x2200

表 4-85：IIC IC_CLR_RX_UNDER 寄存器域的描述

名称	范围	类型	描述
CLR_RX_UNDER	[0]	R,0x0	读此寄存器，清除IC_RAW_INTR_STAT.RX_UNDER中断
—	其它	—	保留。

4.2.20 IIC IC_CLR_RX_OVER 清除RX_OVER中断寄存器

偏移地址：0x2400

表 4-86：IIC IC_CLR_RX_OVER 寄存器域的描述

名称	范围	类型	描述
CLR_RX_OVER	[0]	R,0x0	读此寄存器，清除IC_RAW_INTR_STAT.RX_OVER中断
—	其它	—	保留。

4.2.21 IIC IC_CLR_TX_OVER 清除TX_OVER中断寄存器

偏移地址：0x2600

表 4-87：IIC IC_CLR_TX_OVER 寄存器域的描述

名称	范围	类型	描述
CLR_TX_OVER	[0]	R,0x0	读此寄存器，清除IC_RAW_INTR_STAT.TX_OVER中断
—	其它	—	保留。

4.2.22 IIC IC_CLR_RD_REQ 清除RD_REQ中断寄存器

偏移地址：0x2800

表 4-88：IIC IC_CLR_RD_REQ 寄存器域的描述

名称	范围	类型	描述
CLR_RD_REQ	[0]	R,0x0	读此寄存器，清除IC_RAW_INTR_STAT.RD_REQ中断
—	其它	—	保留。

4.2.23 IIC IC_CLR_TX_ABRT 清除RD_REQ中断寄存器

偏移地址：0x2a00

表 4-89：IIC IC_CLR_TX_ABRT 寄存器域的描述

名称	范围	类型	描述
CLR_TX_ABRT	[0]	R,0x0	读此寄存器，清除IC_RAW_INTR_STAT.TX_ABRT中断
—	其它	—	保留。

4.2.24 IIC IC_CLR_RX_DONE 清除RX_DONE中断寄存器

偏移地址：0x2c00

表 4-90：IIC IC_CLR_RX_DONE 寄存器域的描述

名称	范围	类型	描述
CLR_RX_DONE	[0]	R,0x0	读此寄存器，清除IC_RAW_INTR_STAT.RX_DONE中断
—	其它	—	保留。

4.2.25 IIC IC_CLR_ACTIVITY 清除ACTIVITY中断寄存器

偏移地址：0x2e00

表 4-91：IIC IC_CLR_ACTIVITY 寄存器域的描述

名称	范围	类型	描述
CLR_ACTIVITY	[0]	R,0x0	读此寄存器，清除IC_RAW_INTR_STAT.ACTIVITY中断
—	其它	—	保留。

4.2.26 IIC IC_CLR_STOP_DET 清除STOP_DET中断寄存器

偏移地址：0x3000

表 4-92：IIC IC_CLR_STOP_DET 寄存器域的描述

名称	范围	类型	描述
CLR_STOP_DET	[0]	R,0x0	读此寄存器，清除IC_RAW_INTR_STAT.STOP_DET中断
—	其它	—	保留。

4.2.27 IIC IC_CLR_START_DET 清除START_DET中断寄存器

偏移地址：0x3200

表 4-93：IIC IC_CLR_START_DET 寄存器域的描述

名称	范围	类型	描述
CLR_START_DET	[0]	R,0x0	读此寄存器，清除IC_RAW_INTR_STAT.START_DET中断
—	其它	—	保留。

4.2.28 IIC IC_CLR_GEN_CALL 清除GEN_CALL中断寄存器

偏移地址：0x3400

表 4-94：IIC IC_CLR_GEN_CALL 寄存器域的描述

名称	范围	类型	描述
CLR_GEN_CALL	[0]	R,0x0	读此寄存器，清除IC_RAW_INTR_STAT.GEN_CALL中断
—	其它	—	保留。

4.2.29 IIC IC_ENABLE 总线有效寄存器

偏移地址：0x3600

表 4-95：IIC IC_ENABLE 寄存器域的描述

名称	范围	类型	描述
ENABLE	[0]	RW,0x0	0: DW_I ² C无效; 1: DW_I ² C有效; 当DW_I ² C无效时, TX FIFO和RX FIFO会被刷新。寄存器IC_INTR_STAT依旧有效, 直到DW_I ² C进入到idle状态。
—	其它	—	保留。

4.2.30 IIC IC_STATUS 总线状态寄存器

偏移地址：0x3800

表 4-96：IIC IC_STATUS 寄存器域的描述

名称	范围	类型	描述
SLV_ACTIVITY	[6]	RW,0x0	Slave启动状态: 作为Slave时, 状态机不处于idle状态。 0: Slave处于idle状态;

			1: Slave不处于idle状态;
MST_ACTIVITY	[5]	RW,0x0	Master启动状态: 作为Master时, 状态机不处于idle状态。 0: Master处于idle状态; 1: Master不处于idle状态;
RFF	[4]	RW,0x0	接收FIFO满: 当缓冲中出现空条目, 这一位会被清零。 0: 接收缓冲不满; 1: 接收缓冲满;
RFNE	[3]	RW,0x0	接收FIFO非空: 当缓冲中只要有1个有效条目时, 这一位会被置1。当缓冲为空会被清零。
TFE	[2]	RW,0x0	发送FIFO空: 当缓冲为空会被置1。当缓冲中有至少1个有效条目会被清零。
TFNF	[1]	RW,0x0	发送FIFO非满: 发送缓冲有至少1个空条目, 会被置1。当缓冲为满会被清零。
ACTIVITY	[0]	RW,0x0	I ² C有效: 总线处于activity状态。
—	其它	—	保留。

4.2.31 IIC IC_TXFLR 发送FIFO标记寄存器

偏移地址: 0x3a00

表 4-97: IIC IC_TXFLR 寄存器域的描述

名称	范围	类型	描述
TXFLR	[3:0]	RO,0x0	发送FIFO标记: 标记发送缓冲中有效条目的数量。
—	其它	—	保留。

4.2.32 IIC IC_RXFLR 接收FIFO标记寄存器

偏移地址: 0x3c00

表 4-98: IIC IC_RXFLR 寄存器域的描述

名称	范围	类型	描述
RXFLR	[3:0]	RO,0x0	接收FIFO标记: 标记接收缓冲中有效条目的数量。
—	其它	—	保留。

4.2.33 IIC IC_SDA_HOLD SDA保持时间寄存器

偏移地址：0x3e00

表 4-99: IIC IC_SDA_HOLD SDA 寄存器域的描述

名称	范围	类型	描述
IC_SDA_HOLD	[15:0]	RW,0x0	根据ic_clk频率SDA保持时间计数值。
—	其它	—	保留。

4.2.34 IIC IC_TX_ABRT_SOURCE 传输中断源寄存器

偏移地址：0x4000

表 4-100: IIC IC_TX_ABRT_SOURCE 寄存器域的描述

名称	范围	类型	描述
ABRT_SLVRD_INT X	[15]	RO,0x0	为1表示处理器要作为Slave发送数据，会将1写入IC_DATA_CMD第8位。（作为Slave-Transmitter）
ABRT_SLV_ARBLO ST	[14]	RO,0x0	为1表示作为Slave在发送数据的时候失去总线，IC_TX_ABRT_SOURCE[12]在同时被置1。（作为Slave-Transmitter）
ABRT_SLVFLUSH_ TXFIFO	[13]	RO,0x0	为1表示作为Slave接收到读数据请求，但是有数据在发送缓冲当中，所以发起TX_ABRT中断将发送缓冲中的旧数据刷新出去。（作为Slave-Transmitter）
ARB_LOST	[12]	RO,0x0	为1表示当前占用总线的Master失去仲裁或者IC_TX_ABRT_SOURCE[14]被置1，作为Slave-Transmitter同样失去仲裁。（作为Master-Transmitter或者Slave-Transmitter）
ABRT_MASTER_DI S	[11]	RO,0x0	为1表示用户在Master模式无效的情况下进行对Master的初始化操作。（作为Master-Transmitter或者Master-Receiver）
ABRT_10B_RD_NO RSTRT	[10]	RO,0x0	为1表示Master通过10位地址寻址的方式发起了一个读数据请求且重新启动标志无效。（作为Master-Receiver）
ABRT_SBYTE_NOR STRT	[9]	RO,0x0	为1表示无法发起重新启动标志（IC_CON[5]即IC_RESTART_EN为0）并且尝试发起起始字节。（作为Master）
ABRT_HS_NORSTR	[8]	RO,0x0	为1表示无法发起重新启动标志（IC_CON[5]即

T			IC_RESTART_EN为0) 并且用户要在高速模式下发起传输。(作为Master-Transmitter或者Master-Receiver)
ABRT_SBYTE_ACK DET	[7]	RO,0x0	为1表示Master发出了一个起始字节并且起始字节得到响应。这是一种错误行为。(作为Master)
ABRT_HS_ACKDE T	[6]	RO,0x0	为1表示Master工作在高速模式并且高速模式Master编码被确认。这是一种错误行为。(作为Master)
ABRT_GCALL_RE AD	[5]	RO,0x0	为1表示IP在Master模式发出一个广播 (General Call) 但是软件将广播之后的那个字节编程为读操作 (IC_DATA_CMD[9]被置1)。(作为Master-Transmitter)
ABRT_GCALL_NO ACK	[4]	RO,0x0	为1表示IP发出一个广播但是没有Slave响应。(作为Master-Transmitter)
ABRT_TXDATA_N OACK	[3]	RO,0x0	为1表示在Master发出地址寻址有Slave响应, 但是当Master发送数据的时候没有Slave回正确的ACK。(作为Master-Transmitter)
ABRT_10ADDR2_N OACK	[2]	RO,0x0	为1表示Master使用10位地址寻址Slave, 但是没有Slave在第二字节结束时回ACK。(作为Master-Transmitter或者Master-Receiver)
ABRT_10ADDR1_N OACK	[1]	RO,0x0	1表示Master使用10位地址寻址Slave, 但是没有Slave在第一字节结束时回ACK。(作为Master-Transmitter或者Master-Receiver)
ABRT_7B_ADDR_N OACK	[0]	RO,0x0	为1表示Master使用7位地址寻址Slave, 但是没有Slave对该地址回ACK。(作为Master-Transmitter或者Master-Receiver)
—	其它	—	保留。

4.2.35 IIC IC_SLV_DATA_NACK_ONLY 产生从设备数据NACK寄存器

偏移地址: 0x4200

表 4-101: IIC IC_SLV_DATA_NACK_ONLY 寄存器域的描述

名称	范围	类型	描述
NACK	[0]	RW,0x0	在发送方发出数据后, 需要接收方回一个NACK表示收到数据。 1: 在数据接收后产生NACK

			0: 正常产生NACK/ACK
—	其它	—	保留。

4.2.36 IIC IC_DMA_CR DMA控制寄存器

偏移地址：0x4400

表 4-102：IIC IC_DMA_CR DMA 寄存器域的描述

名称	范围	类型	描述
TDMAE	[1]	RW,0x0	发送DMA有效：用来打开或则关闭DMA通道。
RDMAE	[0]	RW,0x0	接收DMA有效：用来打开或则关闭DMA通道。
—	其它	—	保留。

4.2.37 IIC IC_DMA_TDLR DMA发送数据标记寄存器

只有在有配置 DMA 时才有此寄存器。

偏移地址：0x4600

表 4-103：IIC IC_DMA_TDLR DMA 寄存器域的描述

名称	范围	类型	描述
DMATDL	[3:0]	RW,0x0	DMA发送数据标记：标记发送数据请求引起的DMA传输。
—	其它	—	保留。

4.2.38 IIC IC_DMA_RDLR DMA接收数据标记寄存器

只有在有配置 DMA 时才有此寄存器。

偏移地址：0x4800

表 4-104：IIC IC_DMA_RDLR DMA 寄存器域的描述

名称	范围	类型	描述
DMARDL	[3:0]	RW,0x0	DMA接收数据标记：标记接收数据请求引起的DMA传输。
—	其它	—	保留。

4.2.39 IIC IC_SDA_SETUP 数据线建立时间寄存器

偏移地址：0x4a00

表 4-105：IIC IC_SDA_SETUP 寄存器域的描述

名称	范围	类型	描述
SDA_SETUP	[7:0]	RW,0x6 4	SDA建立时间
—	其它	—	保留。

4.2.40 IIC IC_ACK_GENERAL_CALL 响应广播寄存器

偏移地址：0x4c00

表 4-106：IIC IC_ACK_GENERAL_CALL 寄存器域的描述

名称	范围	类型	描述
ACK_GEN_CALL	[0]	RW,0x1	1: 在接收到General Call发送ACK; 0: 在接收到General Call不发送ACK;
—	其它	—	保留。

4.2.41 IIC IC_ENABLE_STATUS 有效状态寄存器

偏移地址：0x4e00

表 4-107：IIC IC_ENABLE_STATUS 寄存器域的描述

名称	范围	类型	描述
SLV_RX_DATA_LOST	[2]	RO,0x0	Slave接收数据丢失。
SLV_DISABLED_WHILE_BUSY	[1]	RO,0x0	Slave在工作时失去总线。
IC_EN	[0]	RO,0x0	ic_en的状态： 1: I2C视为处于有效状态； 0: I2C视为处于完全无效状态；
—	其它	—	保留。

4.2.42 IIC IC_COMP_PARAM_1 参数寄存器1

偏移地址：0x7a00

表 4-108：IIC IC_COMP_PARAM_1 寄存器域的描述

名称	范围	类型	描述
TX_BUFFER_DEPT	[23:1]	RO,0x0	发送缓冲的深度配置

H	[6]		0x00: 保留 0x01: 2 0x02: 3 ... 0xFF: 256
RX_BUFFER_DEPT H	[15:8]	RO,0x0	接收缓冲的深度配置 0x00: 保留 0x01: 2 0x02: 3 ... 0xFF: 256
ADD_ENCODED_P ARAMS	[7]	RO,0x0	地址译码参数 0: False 1: True
HAS_DMA	[6]	RO,0x0	是否支持DMA: 0: False 1: True
INTR_IO	[5]	RO,0x0	中断输出: 0: 单个输出 1: 组合输出
HC_COUNT_VALU ES	[4]	RO,0x0	是否使用HC_COUNT_VALUES配置 0: False 1: True
MAX_SPEED_MOD E	[3:2]	RO,0x0	I2C能支持的最大工作速度模式;
APB_DATA_WIDTH	[1:0]	RO,0x0	APB总线的数据宽度: 0x0: 8bits 0x1: 16bits 0x2: 32bits 0x3: 保留
—	其它	—	保留。

4.2.43 IIC IC_COMP_VERSION 版本寄存器

偏移地址：0x7c00

表 4-109：IIC IC_COMP_VERSION 寄存器域的描述

名称	范围	类型	描述
IC_COMP_VERSION	[31:0]	RO	I2C的版本信息
—	其它	—	保留。

4.2.44 IIC IC_COMP_TYPE 组成方式寄存器

偏移地址：0x7e00

表 4-110：IIC IC_COMP_TYPE 寄存器域的描述

名称	范围	类型	描述
IC_COMP_TYPE	[31:0]	RO	由“DW”的ASIC码衍生得到。
—	其它	—	保留。

5 PCI-E 接口 IO 寄存器

PCI-E 接口的 IO 寄存器的属性与 1.1 节的描述不同，将遵循 5-1 中的约定。

表 5-1: RC 配置寄存器各域的属性列表

属性	描述
HwInit	硬件初始化 (Hardware Initialized)。该位由硬件控制，软件只读。
RO	只读 (Read-Only)。软件只读。
RW	可读写 (Read-Write)。
RW1C	只读且写“1”清除 (Read-Only/Write-1-to-Clear)。软件对该位写“1”会清除该位，写“0”则没有任何效果。
ROS	粘性只读 (Sticky Read-Only)。软件只读，且在辅助电源的情况下即便是复位也不会清除该域。
RWS	粘性读写 (Sticky Read-Write)。软件可读写，但是在有辅助电源的情况下复位不会清除该域。
RW1CS	粘性只读且写“1”清除 (Sticky Read-Only/Write-1-to-Clear)。软件对该位写“1”会清除该位，写“0”则没有任何效果；且在辅助电源的情况下即便是复位也不会清除该域。

表 5-2: PCI-E 接口基地址说明

模块	核心视角(PA[47:0])	维护视角(PA[39:0])	备注
PCIE0-IOR0	0x8044,0000,0000	0xC4,0000,0000	
PCIE0-IOR1	0x8045,0000,0000	0xC5,0000,0000	
PCIE0-RC	0x8048,0000,0000	0xC8,0000,0000	
PCIE1-IOR0	0x8074,0000,0000	0xf4,0000,0000	
PCIE1-IOR1	0x8075,0000,0000	0xf5,0000,0000	
PCIE1-RC	0x8078,0000,0000	0xf8,0000,0000	

5.1 PCI-E 接口控制部件内部的 IO 寄存器

PCI-E 接口控制部件 (PIU) 内部的 IO 寄存器按照 128B 对界编址，每个寄存器的有效位宽为 64 位。核心对这些寄存器的访问必须是 8B 粒度的。并且，PIU 内部寄存器划分为 IOR0、IOR1 两部分，其基地址描述见表 5-2。每个寄存器描述部分说明了其偏移地址，软件访问时直接将基址+偏

移地址，生成实际访问地址即可。

5.1.1 PIU内部IOR0部分寄存器

5.1.1.1 PIUConfig0

偏移地址：16'h0000

PIU 配置寄存器 0，配置 PCI-E 接口中 DMA 通路的相关功能参数，可读写。

表 5-3: PIUConfig0 寄存器域的描述

名称	范围	类型	描述
DMAR_RO_En	[17]	RWS,0	DMAR请求的RO标志使能控制位。 该位为0时，根据DMARNBConfig控制NB位的设置。 该位为1时，根据DMAR TLP报文头中的RO标志设置NB标志。RO=0时，NB为1；RO=1时，NB为0。
DMAW_RO_En	[16]	RWS,0	DMAW请求的RO标志使能控制位。 该位为0时，根据DMAWNBConfig控制NB位的设置。 该位为1时，根据DMAW TLP报文头中的RO标志设置NB标志。RO=0时，NB为1；RO=1时，NB为0。
PCBoxAndIOTLB_ClkEn	[15]	RWS,1	PCBox和IOTLB的门控时钟使能控制。在不使用页式IOMMU地址代换模式时，如果希望降低功耗，软件可以设置此位为0，关闭PCBox和IOTLB的时钟。 注意： 关闭时钟后，PCache无法进行BIST。
MaxPendDMARCnt	[14:10]	RWS,5'd16	DMA请求通路上悬挂的DMAR请求的最大数量，有效取值为0~16。 使用建议：根据PCI-E接口的性能模拟结果，建议软件对该域做如下配置——如果应用中的DMAR请求大多是256B粒度的数据包，则将该域配置为8；否则如果请求大多是512B粒度的数据包，则将该域配置为16。这样可以获得

			更好的带宽。
IOMMU_Mode	[9:8]	RWS,0	<p>DMA请求地址代换模式选择:</p> <p>00: 普通地址代换;</p> <p>10: 段式IOMMU地址代换;</p> <p>11: 页式IOMMU地址代换;</p> <p>其他: 保留</p>
DMAWNBConfig	[7:6]	RWS,10	<p>指示提交给系统接口的DMAW请求数据包中的NB位（请求包中NB位为“1”表示需要等前面的DMA请求全部收到响应后才可以提交当前请求）如何设置:</p> <p>0b00: NB 位固定为“1”;</p> <p>0b01: NB 位固定为“0”;</p> <p>0b1x: PIU在接收到DMAW请求时, 会按照一定的对界规则将其拆分成多个子包; 该配置时, 拆分出的第一个子包的NB位为“1”, 其余子包的NB位为“0”;</p> <p>其它: 保留。</p>
DMARNBConfig	[5:4]	RWS,01	<p>指示提交给系统接口的DMAR请求数据包中的NB位如何设置:</p> <p>0b00: NB 位固定为“1”;</p> <p>0b01: NB 位固定为“0”;</p> <p>0b10: PIU 在接收到 DMAR 请求时, 会按照一定的对界原则将其拆分成多个子包; 该配置时, 拆分出的第一个子包的 NB 位为“1”, 其余子包的 NB 位为“0”;</p> <p>0b11: 如果有连续的DMAR请求, 则第一个DMAR请求拆分出的第一个子包的NB位为“1”, 其余子包(包括后面连续的DMAR请求)的为“0”。</p>
CCConfig	[3:2]	RWS,01	<p>指示提交给系统接口的 DMAR/W 请求数据包中的 CC 位（决定是否做 Cache 一致性处理）如何设置:</p> <p>00: 则 CC 位固定给 0;</p> <p>01: 则 CC 位固定给 1;</p>

			1x: 根据 PCI-E 接口上 TLP 报文头中的 ATTR[0]位 (No Snoop) 设置, 即 No Snoop 位取反。
CplRelaxOrder	[1]	RWS,1	对于接收到的请求和响应数据包, 该位为0时表示只有本地发起的NP类请求的响应包收到以后才可以接收DMA请求数据包 (暂停请求包的接收), 否则按顺序接收数据包。
DMAStepByStep	[0]	RWS,0	配置DMAR、DMAW是否单步执行。
—	其它	—	保留。

该寄存器与申威 411 处理器相比有修改。

5.1.1.2 EPDMABAR

偏移地址: 16'h0080

EP 设备 DMA 请求基址寄存器, 指示在普通地址代换模式下 (PIUConfig0 中的 IOMMU_Mode 域为 2'b00), EP 发起的 DMA 请求的地址应该在该基址以上的范围内, 可读写, 复位时初值为 0x4_0000_0000。

表 5-4: EPDMABAR 寄存器域的描述

名称	范围	类型	描述
EPDMABAR	[63:0]	RW, 0x400000000	EPDMA 请求基址, 在PIU配置寄存器的IOMMU_EN关闭时用于DMA请求的地址代换。代换方式是: DMA 请求地址减去该基址。 注: 1) 该寄存器[63:40]位段无意义; 2) 软件在配置时应保证[39:37]位段为全0。

5.1.1.3 IOMMU SegItem0~63

偏移地址: 16'h0100~16'h2080

IOMMU 段式代换段表寄存器 0~63, 共计 64 个, 在段式代换模式时 (PIUConfig0 中的 IOMMU_Mode 域为 2'b10) 用作段表快表, 可读写。

表 5-5: IOMMU SegItem0~63 寄存器域的描述

名称	范围	类型	描述
Valid	[63]	RW,0	为“1”时, 指示该寄存器有效。

BaseAddr	[62:23]	RW,0	该tag对应EP设备DMA空间的40位物理基址。 注：该基址应至少保证[12:0]为0，即8KB页面对界。
Mask	[22:16]	RW,0	该tag对应设备DMA请求允许的空间大小（以256M为单位，DMA允许访问地址范围为：0~Mask*2 ²⁸ ）。
TAG	[15:0]	RW,0	索引，由{8位总线号, 5位设备号, 3位功能号}组成。

5.1.1.4 DTBaseAddr

偏移地址：16'hB000

页式代换设备表基址寄存器，在页式地址代换模式时（PIUConfig0 中的 IOMMU_Mode 域为 2'b11）指定一级设备表在主存中的基址，可读写。

表 5-6: DTBaseAddr 寄存器域的描述

名称	范围	类型	描述
DTBaseAddr[39:13]	[39:13]	RW,0	一级设备表基址。 注：一级设备表基址必须按8KB自然对界。
—	其它	—	保留。

5.1.1.5 IOMMUExcpCtrl

偏移地址：16'h2100

IOMMU 地址代换异常控制寄存器，配置 IOMMU 地址代换（段式或页式时）发生异常时的行为，可读写。

表 5-7: IOMMUExcpCtrl 寄存器域的描述

名称	范围	类型	描述
INT_EN	[63]	RW,1	指示IOMMU地址代换异常是否被允许发送中断。
INT_TYPE	[12:4]	RW,0	发生IOMMU地址代换异常时需要发送的中断类型： 0~255对应MSI0~MSI255；256~259对应INTA/B/C/D。
INT_DST	[3:0]	RW,0	该中断的中断目标核心编号。
—	其它	—	保留。

5.1.1.6 IOMMUExcpStatus

偏移地址：16'hb500

IOMMU 地址代换异常状态寄存器，指示异常的类型、设备号等信息，可读写。

表 5-8: IOMMUExcpStatus 寄存器域的描述

名称	范围	类型	描述
Valid	[63]	W1C,0	为“1”时，指示该寄存器有效，即发生了IOMMU地址代换异常。 软件处理该类异常后，需要将该位写1清0。
Retrans_EN	[62]	RW,1	软件处理 IOMMU 异常过程中，需要同时配置该位，以指示：在异常处理结束后，发生异常的请求是否需要重新代换。 该位在 Valid 位由 1 跳变为 0（即软件清 0）的同时被硬件采样，硬件根据该值决定丢弃发生异常的请求（为‘0’时）还是重新代换（为‘1’时）。
ExcptType	[61:59]	RO,0	表示当前 IOMMU 地址代换异常的类型： 0：主存中一级设备表表项无效； 1：主存中二级设备表表项无效； 2：主存中一级页表表项无效； 3：主存中二级页表表项无效； 4：页式代换越权访问； 5：访问主存中设备表/页表，返回非法响应； 6：IOMMU 段式代换缺失； 7：IOMMU 段式代换越界访问。
SegExcptIndex	[58:53]	RO,0	发生段式代换越界异常时，记录发生越界的段表索引号。
DeviceID	[52:37]	RO,0	记录发生异常的DMA请求的设备ID（{8位总线号,5位设备号,3位功能号}）。
DVA	[31:13]	RO,0	记录发生异常的DMA请求的虚页号，用于页式代换的异常处理。
—	其它	—	保留。

5.1.1.7 IOMMUPage_PAddr1

偏移地址：16’hb580

IOMMU 页式地址代换地址寄存器 1，记录页式代换过程的中间结果，即二级设备表在主存中的基址，只读。

表 5-9: IOMMUPage_PAddr1 寄存器域的描述

名称	范围	类型	描述
----	----	----	----

Addr	[39:13]	RO,0	页式地址代换过程中，访问一级设备表而得到的二级设备表基址。 该寄存器主要用于芯片调试。
—	其它	—	保留。

5.1.1.8 IOMMUPage_PAddr2

偏移地址：16'hb600

IOMMU 页式地址代换地址寄存器 2，记录页式代换过程的中间结果，即一级页表在主存中的基址，只读。

表 5-10: IOMMUPage_PAddr2 寄存器域的描述

名称	范围	类型	描述
Addr	[39:13]	RO,0	页式地址代换过程中，访问主存中的二级设备表或硬件的设备表TLB而得到的一级页表基址。 该寄存器主要用于芯片调试。
—	其它	—	保留。

5.1.1.9 IOMMUPage_PAddr3

偏移地址：16'hb680

IOMMU 页式地址代换地址寄存器 3，记录页式代换过程的中间结果，即二级页表在主存中的基址，只读。

表 5-11: IOMMUPage_PAddr3 寄存器域的描述

名称	范围	类型	描述
Addr	[39:13]	RO,0	页式地址代换过程中，访问主存或页表Cache中的一级页表项而得到的二级页表基址。 该寄存器主要用于芯片调试。
—	其它	—	保留。

5.1.1.10 DTLB_FlushAll

偏移地址：16'hb080

设备表 TLB 全刷新寄存器，写该寄存器即对设备表 TLB 的所有条目进行刷新，可读写，且读返回 0。

5.1.1.11 DTLB_FlushDev

偏移地址：16'hb100

指定设备号设备 TLB 表刷新寄存器，写该寄存器即按照指定设备号对设备表 TLB 所有匹配条目进行刷新，可读写。

表 5-12: DTLB_FlushDev 寄存器域的描述

名称	范围	类型	描述
DevNum	[15:0]	RW,0	指定需要刷新的设备表TLB表项的设备编号： [15:8] 总线号； [7:3] 设备号； [2:0] 功能号。
—	其它	—	保留。

5.1.1.12 PTLB_FlushAll

偏移地址：16'hb180

页表 TLB 全刷新寄存器，写该寄存器即对页表 TLB 的所有条目进行刷新，可读写，且读返回 0。

5.1.1.13 PTLB_FlushDev

偏移地址：16'hb200

指定设备号页表 TLB 刷新寄存器，写该寄存器即按照指定设备号对页表 TLB 所有匹配的条目进行刷新，可读写。

表 5-13: PTLB_FlushDev 寄存器域的描述

名称	范围	类型	描述
DevNum	[15:0]	RW,0	指定需要刷新的页表TLB表项的设备编号： [15:8] 总线号； [7:3] 设备号； [2:0] 功能号。
—	其它	—	保留。

5.1.1.14 PTLB_FlushVAddr

偏移地址：16'hb280

指定设备和虚地址页表 TLB 刷新寄存器，写该寄存器即按照指定设备号、指定虚页号对页表

TLB 中匹配的条目进行刷新，可读写。

表 5-14: PTLB_FlushVAddr 寄存器域的描述

名称	范围	类型	描述
VirAddr	[34:16]	RW,0	指定需要刷新的页表TLB项的虚页号（即Addr[31:13]）；
DevNum	[15:0]	RW,0	指定需要刷新的页表TLB项的设备编号： [15:8] 总线号； [7:3] 设备号； [2:0] 功能号。
—	其它	—	保留。

5.1.1.15 PCache_FlushAll

偏移地址：16'hb300

页表 Cache 全刷新寄存器，写该寄存器即对页表 Cache 的所有条目进行刷新，可读写，且读返回 0。

5.1.1.16 PCache_FlushDev

偏移地址：16'hb380

指定设备和索引页表 Cache 刷新寄存器，写该寄存器即按照指定索引、指定设备号对页表 Cache 所有匹配的条目进行刷新，可读写。

表 5-15: PCache_FlushDev 寄存器域的描述

名称	范围	类型	描述
Index	[21:16]	RW,0	指定需要刷新的页表Cache条目所处的索引位置； 该域的宽度与页表Cache容量相关。
DevNum	[15:0]	RW,0	指定需要刷新的页表Cache条目的设备编号： [15:8] 总线号； [7:3] 设备号； [2:0] 功能号。
—	其它	—	保留。

5.1.1.17 PCache_FlushPAddr

偏移地址：16'hb400

指定物理地址页表 Cache 刷新寄存器，写该寄存器即按照指定物理地址对页表 Cache 匹配的条

目进行刷新，可读写。

表 5-16: PCache_FlushPAddr 寄存器域的描述

名称	范围	类型	描述
PhyAddr	[39:7]	RW,0	指定需要刷新的页表 Cache 条目的物理地址高段 (Addr[39:7]) ;
—	其它	—	保留。

5.1.1.18 Timeout_Config

偏移地址: 16'hb480

超时配置寄存器, 可读写。

表 5-17: Timeout_Config 寄存器域的描述

名称	范围	类型	描述
PTCplTO	[3:0]	RW,0x7	页式IOMMU地址代换的主存响应超时阈值。 单位是128个周期, 并且0x0对应1个超时单位; 默认值是8个超时单位 (即1024个周期)。
—	其它	—	保留。

5.1.1.19 MSIAddr

偏移地址: 16'h2180

MSI 地址寄存器, 配置 MSI 中断请求的地址。

表 5-18: MSIAddr 寄存器域的描述

名称	范围	类型	描述
MSIAddr	[63:0]	RW, 0	MSI中断地址。对该地址的DMA写请求即为MSI请求。 该地址不经代换 (普通的、或IOMMU_EN打开的) 直接使用。

5.1.1.20 MSIConfig0~255

偏移地址: 16'h2200~16'ha180

MSI 中断配置寄存器, 共计 256 个, 配置相应中断的控制、状态信息, 可读写。

表 5-19: MSIConfig0~255 寄存器域的描述

名称	范围	类型	描述
----	----	----	----

Valid	[63]	W1C,0	为“1”时，指示该类型中断有效，即有该类型中断提交到对应核心；软件在处理完该中断后需要将该位写1清0。
EN	[62]	RW,0	指示该中断矢量是否被使能。
MSI_Data	[27:4]	RW,0	保存接收到的 MSI 中断的中断数据的[31:8]位段。
INTDST	[3:0]	RW,0	该中断的中断目标核心编号。
—	其它	—	保留。

5.1.1.21 INTxConfig

偏移地址：16’ha200~16’ha380

INTx 中断配置寄存器，共 INTA/B/C/D 四种类型，配置相应中断的控制、状态信息，可读写。

表 5-20: INTA/B/C/Dconfig 寄存器域的描述

名称	范围	类型	描述
Valid	[63]	RO,0	指示该类型中断是否有效，硬件自动更新，软件只读。
EN	[62]	RW,0	中断提交使能位。只有该位为1时，采可以向核心提交该 INTx 中断。 该位相当于屏蔽位取反。软件收到该类型 INTx 中断时，在进入中断处理程序后，需要先将该域写0；等相关中断事件处理完后，再将该位写1。
INTTYPE	[7:4]	RW,0	对于 PCI-E 接口上收到的该类 INTx 中断请求，该位配置应该将其重新映射成何种类型的中断再提交给核心： 0b1000: INTD; 0b0100: INTC; 0b0010: INTB; 0b0001: INTA; 其他：无效。
INTDST	[3:0]	RW,0	该中断的中断目标核心编号。
—	其它	—	保留。

5.1.1.22 aerErrIntConfig

偏移地址：16’ha400

高级错误报告错 INT 中断配置寄存器，可读写。

表 5-21: aerErrIntConfig 寄存器域的描述

名称	范围	类型	描述
Valid	[63]	W1C,0	为“1”时，指示该类型中断有效，即有该事件引起的中断提交到对应核心；软件在处理完该中断后需要将该位写1清0。
EN	[62]	RW,0	指示该中断事件源是否被允许触发中断。
INTTYPE	[7:4]	RW,0	配置该事件触发的中断类型： 0b1000: INTD; 0b0100: INTC; 0b0010: INTB; 0b0001: INTA; 其他：无效。
INTDST	[3:0]	RW,0	该中断的中断目标核心编号。
—	其它	—	保留。

5.1.1.23 aerErrMsiConfig

偏移地址：16'ha480

高级错误报告错 MSI 中断配置寄存器，可读写。

表 5-22: aerErrMsiConfig 寄存器域的描述

名称	范围	类型	描述
Valid	[63]	W1C,0	为“1”时，指示该类型中断有效，即有该事件引起的中断提交到对应核心；软件在处理完该中断后需要将该位写1清0。
EN	[62]	RW,0	指示该中断事件源是否被允许触发中断。
INTTYPE	[8:4]	RO,0	RC指定的MSI中断向量号，即该事件触发的MSI中断向量号 该中断向量号实际是由RC中的“根错误状态寄存器”中的AERIntMsgNum域指定（见5.2.4.1.10）。
INTDST	[3:0]	RW,0	该中断的中断目标核心编号。
—	其它	—	保留。

5.1.1.24 pmeIntConfig

偏移地址：16'ha500

PME（Power Management Event，电源管理事件）INT 中断配置寄存器，可读写。

表 5-23: pmeIntConfig 寄存器域的描述

名称	范围	类型	描述
Valid	[63]	W1C,0	为“1”时，指示该类型中断有效，即有该事件引起的中断提交到对应核心；软件在处理完该中断后需要将该位写1清0。
EN	[62]	RW,0	指示该中断事件源是否被允许触发中断。
INTTYPE	[7:4]	RW,0	配置该事件触发的中断类型： 0b1000: INTD； 0b0100: INTC； 0b0010: INTB； 0b0001: INTA； 其他：无效。
INTDST	[3:0]	RW,0	该中断的中断目标核心编号。
—	其它	—	保留。

5.1.1.25 pmeMsiConfig

偏移地址：16'ha580

PME MSI 中断配置寄存器，可读写。

表 5-24: pmeMsiConfig 寄存器域的描述

名称	范围	类型	描述
Valid	[63]	W1C,0	为“1”时，指示该类型中断有效，即有该事件引起的中断提交到对应核心；软件在处理完该中断后需要将该位写1清0。
EN	[62]	RW,0	指示该中断事件源是否被允许触发中断。
INTTYPE	[8:4]	RO,0	RC指定的MSI中断向量号，即该事件触发的MSI中断向量号 该中断向量号实际是由RC中的“PCI-E能力列表寄存器”中的IntMsgNum域指定（见5.2.3.3.2）。
INTDST	[3:0]	RW,0	该中断的中断目标核心编号。
—	其它	—	保留。

5.1.1.26 hpIntConfig

偏移地址：16'ha600

热插拔 INT 中断配置寄存器，可读写。

表 5-25: hpIntConfig 寄存器域的描述

名称	范围	类型	描述
Valid	[63]	W1C,0	为“1”时，指示该类型中断有效，即有该事件引起的中断提交到对应核心；软件在处理完该中断后需要将该位写1清0。
EN	[62]	RW,0	指示该中断事件源是否被允许触发中断。
INTTYPE	[7:4]	RW,0	配置该事件触发的中断类型： 0b1000: INTD; 0b0100: INTC; 0b0010: INTB; 0b0001: INTA; 其他：无效。
INTDST	[3:0]	RW,0	该中断的中断目标核心编号。
—	其它	—	保留。

5.1.1.27 hpMsiConfig

偏移地址：16'ha680

热插拔 MSI 中断配置寄存器，可读写。

表 5-26: hpMsiConfig 寄存器域的描述

名称	范围	类型	描述
Valid	[63]	W1C,0	为“1”时，指示该类型中断有效，即有该事件引起的中断提交到对应核心；软件在处理完该中断后需要将该位写1清0。
EN	[62]	RW,0	指示该中断事件源是否被允许触发中断。
INTTYPE	[8:4]	RO,0	RC指定的MSI中断向量号，即该事件触发的MSI中断向量号 该中断向量号实际是由RC中的“PCI-E能力列表寄存器”中的IntMsgNum域指定（见5.2.3.3.2）。
INTDST	[3:0]	RW,0	该中断的中断目标核心编号。

—	其它	—	保留。
---	----	---	-----

5.1.1.28 linkAutoBwIntConfig

偏移地址：16'ha700

链路自动带宽调节 INT 中断配置寄存器，可读写。

表 5-27: linkAutoBwIntConfig 寄存器域的描述

名称	范围	类型	描述
Valid	[63]	W1C,0	为“1”时，指示该类型中断有效，即有该事件引起的中断提交到对应核心；软件在处理完该中断后需要将该位写1清0。
EN	[62]	RW,0	指示该中断事件源是否被允许触发中断。
INTTYPE	[7:4]	RW,0	配置该事件触发的中断类型： 0b1000: INTD; 0b0100: INTC; 0b0010: INTB; 0b0001: INTA; 其他：无效。
INTDST	[3:0]	RW,0	该中断的中断目标核心编号。
—	其它	—	保留。

5.1.1.29 bwMgtIntConfig

偏移地址：16'ha780

链路带宽管理 INT 中断配置寄存器，可读写。

表 5-28: bwMgtIntConfig 寄存器域的描述

名称	范围	类型	描述
Valid	[63]	W1C,0	为“1”时，指示该类型中断有效，即有该事件引起的中断提交到对应核心；软件在处理完该中断后需要将该位写1清0。
EN	[62]	RW,0	指示该中断事件源是否被允许触发中断。

INTTYPE	[7:4]	RW,0	配置该事件触发的中断类型： 0b1000: INTD; 0b0100: INTC; 0b0010: INTB; 0b0001: INTA; 其他：无效。
INTDST	[3:0]	RW,0	该中断的中断目标核心编号。
—	其它	—	保留。

5.1.1.30 DMAWReqCplCnt

偏移地址：16'ha800

DMAW 请求/响应计数寄存器，可读写。

表 5-29: DMAWReqCplCnt 寄存器域的描述

名称	范围	类型	描述
DMAW_Cpl_Cnt	[63:32]	RW,0	从系统接口接收到的DMA写响应数目。
DMAW_Req_Cnt	[31:0]	RW,0	提交给系统接口的DMA写请求数目。

5.1.1.31 DMARReqCplCnt

偏移地址：16'ha880

DMAR 请求/响应计数寄存器，可读写。

表 5-30: DMARReqCplCnt 寄存器域的描述

名称	范围	类型	描述
DMAR_Cpl_Cnt	[63:32]	RW,0	从系统接口接收到的DMA读响应数目。
DMAR_Req_Cnt	[31:0]	RW,0	提交给系统接口的DMA读请求数目。

5.1.1.32 MSIReqClrCnt

偏移地址：16'ha900

MSI 中断请求/处理计数寄存器，可读写。

表 5-31: MSIReqClrCnt 寄存器域的描述

名称	范围	类型	描述
MSI_Process_Cnt	[63:32]	RW,0	软件处理MSI中断时，写1清中断有效位次数的计数。

MSI_Req_Cnt	[31:0]	RW,0	提交给系统接口的MSI中断请求数目。
-------------	--------	------	--------------------

5.1.1.33 INTXReqClrCnt

偏移地址：16'ha980

INTX 中断请求/处理计数寄存器，可读写。

表 5-32: INTXReqClrCnt 寄存器域的描述

名称	范围	类型	描述
INTX_Process_Cnt	[63:32]	RW,0	软件处理INTX中断时，写1清中断有效位次数的计数。
INTX_Req_Cnt	[31:0]	RW,0	提交给系统接口的INTX中断请求数目。

5.1.1.34 PageMemReqCplCnt

偏移地址：16'haa00

页式代换访存请求/响应计数器，可读写。

表 5-33: INTXReqClrCnt 寄存器域的描述

名称	范围	类型	描述
PTransMem_Cpl_Cnt	[63:32]	RW,0	页式IOMMU地址代换时，访问主存中设备表/页表项的响应数目。
PTransMem_Req_Cnt	[31:0]	RW,0	页式IOMMU地址代换时，访问主存中设备表/页表项的请求数目。

5.1.2 PIU内部IOR1部分寄存器

5.1.2.1 PIUConfig1

偏移地址：16'h0000

PIU 配置寄存器 1，配置 PCI-E 接口中 IO 通路的相关功能参数，可读写。

表 5-34: PIUConfig1 寄存器域的描述

名称	范围	类型	描述
----	----	----	----

AllOneRspEn	[1]	RWS,0	在 NP 类配置读访问 EP 的响应为 UR/CA 类型时，该域为“1”时指示需要将响应转化为数据全 1 的正常响应后返回给核心，且不再登记 PIO 响应 UR/CA 错误；为“0”时则将读错误响应直接返回给核心，并登记 PIO 响应 UR/CA 错。
PIOStepByStep	[0]	RWS,0	配置PIO请求是否单步执行。
—	其它	—	保留。

5.1.2.2 PIOReqCplCnt

偏移地址：16'h0080

PIO 请求/响应计数器，对来自 IPU 的 PIO 请求、及返回给 IPU 的 PIO 响应数据包的数目进行计数，可读写。

表 5-35: PIOReqCpl 寄存器域的描述

名称	范围	类型	描述
PIO_Cpl_Cnt	[63:32]	RW,0	提交给系统接口的 PIO 请求（Processor IO Request）的响应计数。
PIO_Req_Cnt	[31:0]	RW,0	从系统接口接收到的的 PIO 请求计数。

5.1.2.3 RCReqCplCnt

偏移地址：16'h0100

RC 请求/响应计数器，对来自 IPU 的对 RC 的访问请求、及 RC 返回的响应数据包进行计数，可读写。

表 5-36: RCReqCplCnt 寄存器域的描述

名称	范围	类型	描述
RC_Cpl_Cnt	[63:32]	RW,0	PIU 收到的 RC 返回的响应计数。
RC_Req_Cnt	[31:0]	RW,0	PIU 发送给 RC 的请求计数。

5.1.2.4 EPNPReqCplCnt

偏移地址：16'h0180

对 EP 的 NP 类请求/响应计数器，对来自 IPU 的对 EP 的 NP 类访问请求及 EP 返回的对应响应数据包进行计数，可读写。

表 5-37: NPReqCplCnt 寄存器域的描述

名称	范围	类型	描述
NP_Cpl_Cnt	[63:32]	RW,0	PIU 收到的 EP 返回的 NP 类请求所对应响应的计数。
Client0_Req_Cnt	[31:0]	RW,0	PIU 发送给 EP 的 NP 类请求计数。

5.1.2.5 IOR0ReqCplCnt

偏移地址: 16'h0200

IOR0 请求/响应计数器, 对来自 IPU 的对 IOR0 部分寄存器的访问请求及 IOR0 部分寄存器返回的响应数据包进行计数, 可读写。

表 5-38: IOR0ReqCplCnt 寄存器域的描述

名称	范围	类型	描述
IOR0_Cpl_Cnt	[63:32]	RW,0	PIU 内部 IOR0 寄存器部分返回的响应计数。
IOR0_Req_Cnt	[31:0]	RW,0	PIU 内部 IOR0 部分寄存器收到的访问请求计数。

5.1.2.6 IOR1ReqCplCnt

偏移地址: 16'h0280

IOR1 请求/响应计数器, 对来自 IPU 的对 IOR1 部分寄存器的访问请求及 IOR1 部分寄存器返回的响应数据包进行计数, 可读写。

表 5-39: IOR1ReqCplCnt 寄存器域的描述

名称	范围	类型	描述
IOR1_Cpl_Cnt	[63:32]	RW,0	PIU 内部 IOR1 寄存器部分返回的响应计数。
IOR1_Req_Cnt	[31:0]	RW,0	PIU 内部 IOR1 部分寄存器收到的访问请求计数。

5.1.2.7 NewLTSSMState0~4

偏移地址: 16'h0300~16'h0500

NewLTSSMState0~4 寄存器记录了 PCI-E 链路的 LTSSM 的最近的 60 个状态 (以移位寄存器的方式), 其中 NewLTSSMState0[4:0]记录的是最新的状态。

表 5-40: NewLTSSMState0~4 寄存器域的描述

名称	范围	类型	描述
----	----	----	----

LTSSMState	[59:0]	RO,0	每 5 位表示一个 LTSSM 状态，状态编码含义见附录 B。
—	其它	—	保留。

5.1.2.8 OldLTSSMState0~4

偏移地址：16'h0580~16'h0780

OldLTSSMState0~4 寄存器记录了 PCI-E 链路初始化链路训练时 LTSSM 的 60 个状态，其中 OldLTSSMState0[4:0]记录的是前 60 个状态中最新的状态。

如果工作期间，PCI-E 链路的状态迁移少于 60 个，则 NewLTSSMState0~4 和 OldLTSSMState0~4 记录的状态完全一样；状态达到 60 个后，OldLTSSMState0~4 即停止记录，保持最初的状态，而 NewLTSSMState0~4 则继续记录最新的状态。

表 5-41: OldLTSSMState0~4 寄存器域的描述

名称	范围	类型	描述
LTSSMState	[59:0]	RO,0	每 5 位表示一个 LTSSM 状态，状态编码含义见附录 B。
—	其它	—	保留。

5.1.2.9 ErrReport

偏移地址：16'h0800

错误登记寄存器，登记 PIU 内部检测到的各种类型错误，可读写。

表 5-42: ErrReport 寄存器域的描述

名称	范围	类型	描述
PCacheTagPErr_OverLimit	[30]	RW,0	IOMMU 页式地址代换时，页表 Cache 的 TAG 阵列发生偶校验错的次数超过阈值。
PCacheDataECCSErr_OverLimit	[29]	RW,0	IOMMU 页式地址代换时，页表 Cache 的数据阵列发生 ECC 单错的次数超过阈值。
PCacheDataECCMErr	[28]	RW,0	IOMMU 页式地址代换时，页表 Cache 的数据阵列发生 ECC 多错。
DMARCPiTimeout	[27]	RW,0	PCI-E 链路提示 DMAR 请求的响应发生超时。
PIOLengthErr	[26]	RW,0	收到对 RC 配置空间、EP I/O 以及配置空间的 I/O 请求长度超过 32 位。
DMAWLengthErr	[24]	RW,0	收到的 DMAW 请求长度超过 512B。

DMARHardWall	[22]	RW,0	收到对不可访问的 CG（核组）的 DMAR 操作，登记错误。
DMAWHardWall	[21]	RW,0	收到对不可访问的 CG（核组）的 DMAW 操作，登记错误。
DMAAddrErr	[20]	RW,0	非 IOMMU 地址代换时，DMA 请求地址错误——请求地址的[36:0]位段低于 EPIOSpace 寄存器 [36:0]位段的值。
MSIErr	[19]	RW,0	MSI 请求错误，即携带数据长度不是 32 位。
PIOCpl_CRS	[18]	RW,0	PIO 请求返回的响应带有 CRS 标志(Configuration Request Retry Status)。
PIOCplTimeout	[17]	RW,0	NP 类 PIO 请求的响应超时。
PIOCpl_UR	[16]	RW,0	PIO 请求返回的响应带有 UR 标志(Unsupported Request)。
PIOCpl_CA	[15]	RW,0	PIO 请求返回的响应带有 CA 标志(Completion Abort)。
PIOReqECCSErr_OverLimit	[14]	RW,0	来自 IPU 的 PIO 请求数据包发生 ECC 单错的次数超过了阈值(见 5.2.14 中 ECCSErrLimit 寄存器)。
PIOReqECCMErr	[13]	RW,0	来自 IPU 的 PIO 请求数据包发生 ECC 多错。
MCUECCSErr_OverLimit	[12]	RW,0	来自 MCU 的请求数据包发生 ECC 单错的次数超过了阈值(见 5.2.14 中 ECCSErrLimit 寄存器)。
MCUECCMErr	[11]	RW,0	来自 MCU 的请求发生 ECC 多错。
PIOReqTypeErr	[10]	RW,0	来自 IPU 的 PIO 请求数据包类型出错。
PIOReqAddrErr	[9]	RW,0	来自 IPU 的 PIO 请求的地址不在 PCI-E 空间内。
PIOWrReqDVLDErr	[8]	RW,0	来自 IPU 的 PIO 写请求携带的数据有效位(DVLD, Data Valid)为全 0。
DMACplECCSErr_OverLimit	[7]	RW,0	来自 IPU 的 DMAR 响应发生 ECC 单错的次数超过了阈值(见 5.2.14 中 ECCSErrLimit 寄存器)。
DMACplECCMErr	[6]	RW,0	来自 IPU 的 DMAR 响应发生 ECC 多错。
IPUDMAWrErrAck	[5]	RW,0	收到来自 IPU 的 DMAW 错误响应。
IPUDMARdErrAck	[4]	RW,0	收到来自 IPU 的 DMAR 错误响应。
DMAReqFifo_OverflowErr	[3]	RW,0	DMA 请求队列发生上溢。
DMAWrDataFifo_OverflowErr	[2]	RW,0	DMAW 数据缓冲发生上溢。
DMAWrDataFifo_UnferflowErr	[1]	RW,0	DMAW 数据缓冲发生下溢。

DMAWrData_Poison	[0]	RW,0	收到了带有数据中毒标志的 DMAW 请求。
—	其它	—	保留。

5.1.2.10 ErrEnable

偏移地址：16'h0880

错误使能寄存器，控制是否使能 PIU 内部相应的错误检测，可读写。

表 5-43: ErrEnable 寄存器域的描述

名称	范围	类型	描述
PCacheTagPErrOverLimit_En	[30]	RW,1	是否使能 PCacheTagPErr_OverLimit 错误登记。
PCacheDataECCSErrOverLimit_En	[29]	RW,1	是否使能 PCacheDataECCSErr_OverLimit 错误登记。
PCacheDataECCMErr_En	[28]	RW,1	是否使能 PCacheDataECCMErr 错误登记。
DMARCplTimeout_En	[27]	RW,1	是否使能 DMARCplTimeout 错误登记。
PIOLengthErr_En	[26]	RW,1	是否使能 PIOLengthErr 错误登记。
DMAWLengthErr_En	[24]	RW,1	是否使能 DMAWLengthErr 错误登记。
DMARHardWall_En	[22]	RW,1	是否使能 DMARHardWall_NotAllOneRsp 错误登记。
DMAWHardWall_En	[21]	RW,1	是否使能 DMAWHardWall 错误登记。
DMAAddrErr_En	[20]	RW,1	是否使能 DMAAddrErr 错误登记。
MSIErr_En	[19]	RW,1	是否使能 MSIErr 错误登记。
PIOCpl_CRs_En	[18]	RW,1	是否使能 PIOCpl_CRs 错误登记。
PIOCplTimeout_En	[17]	RW,1	是否使能 PIOCplTimeout 错误登记。
PIOCpl_UR_En	[16]	RW,1	是否使能 PIOCpl_UR 错误登记。
PIOCpl_CA_En	[15]	RW,1	是否使能 PIOCpl_CA 错误登记。
IPUReqECCSErrOverLimit_En	[14]	RW,1	是否使能 IPUReqECCSErr_OverLimit 错误登记。
IPUReqECCMErr_En	[12]	RW,1	是否使能 IPUReqECCMErr 错误登记。
MCUECCSErrOverLimit_En	[12]	RW,1	是否使能 MCUECCSErr_OverLimit 错误登记。
MCUECCMErr_En	[11]	RW,1	是否使能 MCUECCMErr 错误登记。
PIOReqPTypeErr_En	[10]	RW,1	是否使能 PIOReqPTypeErr 错误登记。
PIOReqAddrErr_En	[9]	RW,1	是否使能 PIOReqAddrErr 错误登记。
PIOWrReqDVLDErr_En	[8]	RW,1	是否使能 PIOWrReqDVLDErr 错误登记。
IPUCplECCSErrOverLimit_En	[7]	RW,1	是否使能 IPUCplECCSErr_OverLimit 错误登记。

IPUCplECCMErr_En	[6]	RW,1	是否使能 IPUCplECCMErr 错误登记。
IPUDMAWrErrAck_En	[5]	RW,1	是否使能 IPUDMAWrErrAck 错误登记。
IPUDMARdErrAck_En	[4]	RW,1	是否使能 IPUDMARdErrAck 错误登记。
DMAReqFifoOverflowErr_En	[3]	RW,1	是否使能 DMAReqFifo_OverflowErr 错误登记。
DMAWrDataFifoOverflowErr_En	[2]	RW,1	是否使能 DMAWrDataFifo_OverflowErr 错误登记。
DMAWrDataFifoUnferflowErr_En	[1]	RW,1	是否使能 DMAWrDataFifo_UnferflowErr 错误登记。
DMAWrData_Poison_En	[0]	RW,1	是否使能 DMAWrData_Poison 错误登记。
—	其它	—	保留。

5.1.2.11 DMACpl_SErrCnt

偏移地址：16'h0900

DMA 响应 ECC 单错计数器，可读写，且写清 0。

表 5-44: DMACpl_SErrCnt 寄存器域的描述

名称	范围	类型	描述
DMACplECCSErrCnt	[31:0]	RWC,0	来自IPU的DMAR响应数据包发生ECC单错计数器。该计数器设定为饱和计数，全1时停止计数并保持。
—	其它	—	保留。

5.1.2.12 PIOReq_SErrCnt

偏移地址：16'h0980

PIO 请求 ECC 单错计数器，可读写，且写清 0。

表 5-45: PIOReq_SErrCnt 寄存器域的描述

名称	范围	类型	描述
PIOReqECCSErrCnt	[31:0]	RWC,0	来自IPU的PIO请求数据包发生ECC单错计数器。该计数器设定为饱和计数，全1时停止计数并保持。
—	其它	—	保留。

5.1.2.13 MCUREq_SErrCnt

偏移地址：16'h0a00

MCU 请求 ECC 单错计数器，可读写，且写清 0。

表 5-46: MCUREq_SErrCnt 寄存器域的描述

名称	范围	类型	描述
MCUReqECCSErrCnt	[31:0]	RWC,0	来自MCU的请求数据包发生ECC单错计数器。该计数器设定为饱和计数，全1时停止计数并保持。
—	其它	—	保留。

5.1.2.14 PCData_SErrCnt

偏移地址：16'h1080

页表 Cache 数据单错计数寄存器，对页式代换过程中发现的页表 Cache 数据阵列 ECC 单错进行计数，可读写，且写清 0。

表 5-47: PCData_SErrCnt 寄存器域的描述

名称	范围	类型	描述
PCDATA_SErrCnt	[31:0]	RWC,0	页表Cache的数据阵列的ECC校验单错计数。 该计数器为饱和计数，全1时停止计数并保持。 该计数器的计数达到ECCSErrLimit指定的阈值后，会报单错预警给MCU。
—	其它	—	保留。

5.1.2.15 PCTag_PErrCnt

偏移地址：16'h1100

页表 Cache TAG 偶校验错计数寄存器，对页式代换过程中发现的页表 Cache Tag 阵列偶校验错进行计数，可读写，且写清 0。

表 5-48: PCTag_PErrCnt 寄存器域的描述

名称	范围	类型	描述
PCTAG_PErrCnt	[31:0]	RWC,0	页表Cache的TAG阵列的偶校验错计数。 该计数器设定为饱和计数，全1时停止计数并保持。 该计数器的计数达到ECCSErrLimit指定的阈值后，会报单错预警给MCU。
—	其它	—	保留。

5.1.2.16 ECCSErrLimit

偏移地址：16'h0a80

ECC 单错计数阈值寄存器，可读写。

表 5-49: ECCSErrLimit 寄存器域的描述

名称	范围	类型	描述
ECCSErrLimit	[31:0]	RW,0xffffffff	ECC单错阈值寄存器。如果来自IPU的PIO请求数据包、或DMAR响应数据包、或来自MCU的请求数据包的ECC单错计数超过了该阈值，则在ErrReport寄存器中登记相应的错误。
—	其它	—	保留。

5.1.2.17 RCCControl

偏移地址：16'h0b00

RC 控制寄存器，配置 RC 的行为参数，可读写。

表 5-50: RCCControl 寄存器域的描述

名称	范围	类型	描述
cfg_l1_clk_removal_en	[3]	RW,0	配置该位为“1”，则链路在L1状态下也可以关闭时钟。
tx_lane_flip_en	[2]	RW,0	控制传输通道手动进行通道翻转 (lane reversal)，可用于硬件调试。在由于lane0没有检测到而不能自动翻转时可以用到。
rx_lane_flip_en	[1]	RW,0	控制接收通道手动进行通道翻转 (lane reversal)，可用于硬件调试。在由于lane0没有检测到而不能自动翻转时可以用到。
—	其它	—	保留。

5.1.2.18 RCPowerManage

偏移地址：16'h0b80

RC 电源管理寄存器，配置 RC 电源管理行为参数，可读写。

表 5-51: RCPowerManage 寄存器域的描述

名称	范围	类型	描述
sys_aux_pwr_det	[3]	RW,0	指示当前系统是否配置有辅助电源。
app_req_entr_l1	[2]	RW,0	应用 (application, 指 PIU) 要求链路进入 L1 状态，需要 ASPM 支持 L1 状态，当核心正在处理传输事务时会忽略该请求。

app_req_exit_l1	[1]	RW,0	应用请求退出 L1 状态, 仅在 ASPM 支持 L1 状态时有用。
app_ready_entr_l23	[0]	RW,1	应用准备好进入L2、L3状态, 当设置该信号为“0”时, 核心推迟发PM_Enter_L23 消息 (对应PM_Turn_Off) 直到变为“1”。
—	其它	—	保留。

5.1.2.19 RCDebugInf0

偏移地址: 16'h0c00

RC 调试信息寄存器 0, 记录 RC 链路传输方面的调试信息, 只读, 对该寄存器的写请求被忽略但不报错。

表 5-52: RCDebugInf0 寄存器域的描述

名称	范围	类型	描述
cxpl_debug_info	[63:0]	RO,0	RC调试信息: [63]: 链路禁止数据缠绕; [62]: 链路训练状态处于禁用 (Disable) 状态, 链路不可用; [61]: 链路正在训练中; [60]: 链路正在测试极性颠倒 (polarity reversal); [59]: 链路需要复位; [54]: PIPE接口的接收器检测、或环回测试请求; [53]: PIPE发出电气空闲请求; [52]: PIPE发出兼容性测试 (compliance) 请求; [51]: 应用申请复位链路; [47:40]: 链路接收方通告/确认的通道数; [37]: 发送了SKP有序集; [36]: PMA报告链路成功 (link up); [35]: 接收器报告收到SKP符号; [34]: 接收器报告收到TS1序列 (脉冲); [33]: 接收器报告收到TS2序列 (脉冲); [32]: 接收器检测到通道反向 (lane reversal); [31:28]: 链路另一端通告的链路控制位; [27]: 协商通道数目时在相应的域为PAD空符号; [26]: 协商链路数目时在相应的域为PAD空符号;

			[25]: 接收方收到逻辑空闲; [24]: 第2个符号依然空闲 (只在Gen2模式下PIPE接口宽度为16时有意义); [23:8]: PIPE接口传输的数据; [7:6]: PIPE接口传输的数据是否为控制字符的指示; [5:0]: LTSSM当前的状态; 其它: 保留。
--	--	--	---

5.1.2.20 RCDebugInf1

偏移地址: 16'h0c80

RC 调试信息寄存器 1, 只读。

表 5-53: RCDebugInf1 寄存器域的描述

名称	范围	类型	描述
cfg_2nd_reset	[17]	RO,0	二级总线热复位
radm_qoverflow	[16]	RO,0	发生过缓冲阵列溢出错误
radm_q_not_empty	[15]	RO,0	指示接收队列中TLP头或数据
aux_pm_en	[14]	RO,0	辅助电源使能。
pm_pme_en	[13]	RO,0	电源管理事件使能。
pm_status	[12]	RO,0	电源管理事件状态。
pm_dstate	[11:9]	RO,0	设备的电源管理状态 (DSTATE)。
xmlh_ltssm_state	[8:4]	RO,0	链路训练状态。
pm_curnt_state	[3:1]	RO,0	指示当前电源管理状态。
rdlh_link_up	[0]	RO,0	接收方链路建立。
—	其它	—	保留。

5.1.2.21 RCElectroControl

偏移地址: 16'h0d00

RC 电气特性控制寄存器, 可读写。

表 5-54: RCElectroControl 寄存器域的描述

名称	范围	类型	描述
sys_atten_button_pressed	[7]	RW,0	指示按了系统警示按钮 (attention button)。
sys_pre_det_state	[6]	RW,0	指示插槽中有无插卡。

sys_mrl_sensor_state	[5]	RW,0	指示MRL (Manually-operated Retention Latch) 传感器的状态。
sys_pwr_fault_det	[4]	RW,0	指示发现插槽供电错误。
sys_mrl_sensor_chged	[3]	RW,0	指示MRL传感器的状态有变化。
sys_pre_det_chged	[2]	RW,0	指示插槽插卡在位检测状态有变化。
sys_cmd_cpled_int	[1]	RW,0	指示热插拔控制器完成了一条命令。
sys_eml_interlock_engaged	[0]	RW,0	指示系统电气互锁设备有无被占用。
—	其它	—	保留。

5.1.2.22 RCElectroStatus

偏移地址：16'h0d80

RC 电气特性状态寄存器，只读。

表 5-55: RCElectroStatus 寄存器域的描述

名称	范围	类型	描述
cfg_eml_control	[0]	RO,0	电气互锁设备控制状态。
—	其它	—	保留。

5.1.2.23 RCHotRst

偏移地址：16'h0e00

RC 复位寄存器，控制 RC 生成热复位，可读写。

表 5-56: RCHotRst 寄存器域的描述

名称	范围	类型	描述
app_init_rst	[0]	RW,0	应用申请复位，以对链路重新进行训练。 链路正常工作后，对该位先配为1再配为0，可以触发链路热复位。
—	其它	—	保留。

5.1.2.24 MSGControl0

偏移地址：16'h0e80

MSG 控制寄存器 0。写该地址即产生一个脉冲信号控制 app_unlock_msg；读返回全 0。

5.1.2.25 MSGControl1

偏移地址：16'h0f00

MSG 控制寄存器 1。写该地址即产生一个脉冲信号控制 apps_pm_xmt_turnoff；读返回全 0。

5.1.2.26 LinkDownDelay

偏移地址：16'h0f80

链路复位延迟控制寄存器，可读写。

表 5-57: LinkDownDelay 寄存器域的描述

名称	范围	类型	描述
LinkdownCnt	[7:0]	RW,0x20	PCI-E链路热复位时，为保证内部逻辑的正确性，需要延迟该热复位操作；该参数即配置了内部生成热复位到真正实施时的延迟周期数。
—	其它	—	保留。

5.1.2.27 PMAControl0

偏移地址：16'1000

PMA 控制寄存器 0，可读写。

表 5-58: PMAControl0 寄存器域的描述

名称	范围	类型	描述
ext_tx_vboost_lvl	[63:61]	RWS,0	设置 TX 端最大电压摆幅
ext_rx_vref_ctrl	[60:56]	RWS,0	设置 RX 端偏置电流
ext_mplla_ssc_range	[55:53]	RWS,0	扩展频谱时钟范围
ext_mplla_ssc_clk_sel	[52:50]	RWS,0	扩展频谱时钟选择
ext_mplla_multiplier	[49:42]	RWS,0	倍频参数
ext_mplla_fracn_ctrl	[41:33]	RWS,0	MPLLA 的部分控制输入
ext_mplla_div_multiplier	[32:26]	RWS,0	倍频参数
ext_mplla_div_clk_en	[25]	RWS,0	mplla_div_clk 的频率为 MPLL 频率的 ext_mplla_div_multiplier 倍
ext_mplla_div8_clk_en	[24]	RWS,0	mplla_work_clk 的频率为 MPLL 频率的 8 倍
ext_mplla_div16p5_clk_en	[23]	RWS,0	mplla_work_clk 的频率为 MPLL 频率的 16.5 倍
ext_mplla_div10_clk_en	[22]	RWS,0	mplla_work_clk 的频率为 MPLL 频率的 10 倍

ext_mplla_bandwidth	[21:15]	RWS,0	MPLLA的带宽
ext_bs_tx_lowswing	[14]	RWS,0	使能ACJTAG的小摆幅模式
ext_bs_rx_level	[13:9]	RWS,0	设置ACJTAG接收器的敏感阈值
ext_bs_rx_bigswing	[8]	RWS,0	使能ACJTAG的大摆幅模式
phy_ext_ctrl_sel	[7]	RWS,0	当 PHY 使用默认的配置值有错的时候,可以尝试将该位置 1, 使用 protocol0 的信号值来配置 PHY, 用于芯片调试。
ext_pclk_req	[6]	RWS,0	当 PCS 处于 POWERDOWN 的状态时, 依旧提供稳定的 pclk
phy_test_powerdown	[5]	RWS,0	所有电路关闭电源的控制。 关闭 PHY 中的所有电阻以进行 IDDQ 测试。 该模式下PHY不再工作; 该信号撤销后需要重新复位。
phy_test_burnin	[4]	RWS,0	激活所有电路。 激活 PHY 的所有电路以进行 burn-in 测试。 该信号在test_powerdown有效时不能有效。
phy_rtune_req	[3]	RWS,0	调节阻抗请求。
phy_res_req_in	[2]	RWS,0	多个PHY共享参考电阻的仲裁信号线 (arbitration lines), 与参考电阻调节有关。
phy_lane_rx2tx_par_lb_en	[1]	RWS,0	环回测试, 进行芯片测试时使用
pipe_tx2rx_loopbk	[0]	RWS,0	从PIU到PCS (Physical Coding Sub-layer, 物理编码子层, 在RC和PMA之间)的环回测试, 进行芯片测试时使用。
—	其它	—	保留。

5.1.2.28 PMAControl1

偏移: 16'h1180

PMA 控制寄存器 1, 可读写

表 5-59:: PMAControl1 寄存器域的描述

名称	范围	类型	描述
ext_rx_eq_att_lv1_g4	[63:61]	RWS,0	均衡衰减级别, 3'b000: -2dB; 3'b111: -6dB
ext_rx_eq_att_lv1_g3	[60:58]	RWS,0	均衡衰减级别, 3'b000: -2dB; 3'b111: -6dB
ext_rx_eq_att_lv1_g2	[57:55]	RWS,0	均衡衰减级别, 3'b000: -2dB; 3'b111: -6dB

ext_rx_eq_att_lvl_g1	[54:52]	RWS,0	均衡衰减级别, 3'b000: -2dB; 3'b111: -6dB
ext_rx_vco_ld_val_g4	[51:39]	RWS,0	VCO刻度值
ext_rx_vco_ld_val_g3	[38:26]	RWS,0	VCO刻度值
ext_rx_vco_ld_val_g2	[25:13]	RWS,0	VCO刻度值
ext_rx_vco_ld_val_g1	[12:0]	RWS,0	VCO刻度值
—	其它	—	保留。

5.1.2.29 PMAControl2

偏移: 16'h1200

PMA 控制寄存器 2, 可读写

表 5-6058: PMAControl2 寄存器域的描述

名称	范围	类型	描述
ext_rx_ref_ld_val_g4	[63:58]	RWS,0	VCO 参考刻度值
ext_rx_ref_ld_val_g3	[57:52]	RWS,0	VCO 参考刻度值
ext_rx_ref_ld_val_g2	[51:46]	RWS,0	VCO参考刻度值
ext_rx_ref_ld_val_g1	[45:40]	RWS,0	VCO参考刻度值
ext_mpll_b_ssc_range	[39:37]	RWS,0	扩展频谱时钟范围
ext_mpll_b_ssc_clk_sel	[36:34]	RWS,0	扩展频谱时钟选择
ext_mpll_b_multiplier	[33:26]	RWS,0	倍频参数
ext_mpll_b_fracn_ctrl	[25:17]	RWS,0	MPLL B的部分控制输入
ext_mpll_b_div_multiplier	[16:10]	RWS,0	倍频参数
ext_mpll_b_div_clk_en	[9]	RWS,0	mpll_b_div_clk 的频率为 MPLL 频率的 ext_mpll_b_div_multiplier倍
ext_mpll_b_div8_clk_en	[8]	RWS,0	mpll_b_work_clk的频率为MPLL频率的8倍
ext_mpll_b_div10_clk_en	[7]	RWS,0	mpll_b_work_clk的频率为MPLL频率的10倍
ext_mpll_b_bandwidth	[6:0]	RWS,0	MPLL B的带宽
—	其它	—	保留

5.1.2.30 PMAControl3

偏移: 16'h1280

PMA 控制寄存器 3, 可读写

表 5-61: PMAControl3 寄存器域的描述

名称	范围	类型	描述
ext_rx_eq_vga2_gain_g4	[63:60]	RWS,0	控制 AFE 的二级可变放大器
ext_rx_eq_vga2_gain_g3	[59:56]	RWS,0	控制 AFE 的二级可变放大器
ext_rx_eq_vga2_gain_g2	[55:52]	RWS,0	控制 AFE 的二级可变放大器
ext_rx_eq_vga2_gain_g1	[51:48]	RWS,0	控制 AFE 的二级可变放大器
ext_rx_eq_vga1_gain_g4	[47:44]	RWS,0	控制 AFE 的一级可变放大器
ext_rx_eq_vga1_gain_g3	[43:40]	RWS,0	控制 AFE 的一级可变放大器
ext_rx_eq_vga1_gain_g2	[39:36]	RWS,0	控制 AFE 的一级可变放大器
ext_rx_eq_vga1_gain_g1	[35:32]	RWS,0	控制 AFE 的一级可变放大器
ext_rx_eq_dfe_tap1_g4	[31:24]	RWS,0	控制 DFE 数据 TAP1 的值
ext_rx_eq_dfe_tap1_g3	[23:16]	RWS,0	控制 DFE 数据 TAP1 的值
ext_rx_eq_dfe_tap1_g2	[15:8]	RWS,0	控制 DFE 数据 TAP1 的值
ext_rx_eq_dfe_tap1_g1	[7:0]	RWS,0	控制 DFE 数据 TAP1 的值
—	其它	—	保留。

5.1.2.31 PMAControl4

偏移：16'h1300

PMA 控制寄存器 4，可读写

表 5-62：PMAControl4 寄存器域的描述

名称	范围	类型	描述
ext_rx_los_threshold	[60:58]	RWS,0	设置接收器的 LOS 阈值，PCIE 推荐 3'b001
ext_rx_los_pwr_up_cnt	[57:47]	RWS,0	phy_ext_ctrl_sel 为 1 时有效
ext_rx_los_lfps_en	[46]	RWS,0	接收端 LFPS 使能，保留，接 0
ext_rx_eq_ctle_pole_g4	[45:43]	RWS,0	控制接收端均衡 CTLE 电极的位置
ext_rx_eq_ctle_pole_g3	[42:40]	RWS,0	控制接收端均衡 CTLE 电极的位置
ext_rx_eq_ctle_pole_g2	[39:37]	RWS,0	控制接收端均衡 CTLE 电极的位置
ext_rx_eq_ctle_pole_g1	[36:34]	RWS,0	控制接收端均衡 CTLE 电极的位置
ext_rx_eq_ctle_boost_g4	[33:29]	RWS,0	控制接收端均衡 CTLE 增强的等级
ext_rx_eq_ctle_boost_g3	[28:24]	RWS,0	控制接收端均衡 CTLE 增强的等级
ext_rx_eq_ctle_boost_g2	[23:19]	RWS,0	控制接收端均衡 CTLE 增强的等级
ext_rx_eq_ctle_boost_g1	[18:14]	RWS,0	控制接收端均衡 CTLE 增强的等级
ext_rx_adapt_dfe_en_g4	[13]	RWS,0	使能接收端适应电路和 DFE 电路(decision

			feedback equalization), 使用以下配置信号: ext_rx_eq_dfe_tap1_g4
ext_rx_adapt_dfe_en_g3	[12]	RWS,0	使能接收端适应电路和 DFE 电路(decision feedback equalization), 使用以下配置信号: ext_rx_eq_dfe_tap1_g3
ext_rx_adapt_dfe_en_g2	[11]	RWS,0	使能接收端适应电路和 DFE 电路(decision feedback equalization), 使用以下配置信号: ext_rx_eq_dfe_tap1_g2
ext_rx_adapt_dfe_en_g1	[10]	RWS,0	使能接收端适应电路和 DFE 电路(decision feedback equalization), 使用以下配置信号: ext_rx_eq_dfe_tap1_g1
ext_rx_adapt_afe_en_g4	[9]	RWS,0	使能接收端适应电路, 使用以下配置信号: ext_rx_eq_att_lvl_g4 ext_rx_eq_vga2_gain_g4 ext_rx_eq_vga1_gain_g4 ext_rx_eq_ctle_pole_g4 ext_rx_eq_ctle_boost_g4
ext_rx_adapt_afe_en_g3	[8]	RWS,0	使能接收端适应电路, 使用以下配置信号: ext_rx_eq_att_lvl_g3 ext_rx_eq_vga2_gain_g3 ext_rx_eq_vga1_gain_g3 ext_rx_eq_ctle_pole_g3 ext_rx_eq_ctle_boost_g3
ext_rx_adapt_afe_en_g2	[7]	RWS,0	使能接收端适应电路, 使用以下配置信号: ext_rx_eq_att_lvl_g2 ext_rx_eq_vga2_gain_g2 ext_rx_eq_vga1_gain_g2 ext_rx_eq_ctle_pole_g2 ext_rx_eq_ctle_boost_g2
ext_rx_adapt_afe_en_g1	[6]	RWS,0	使能接收端适应电路, 使用以下配置信号: ext_rx_eq_att_lvl_g1 ext_rx_eq_vga2_gain_g1 ext_rx_eq_vga1_gain_g1 ext_rx_eq_ctle_pole_g1

			ext_rx_eq_ctle_boost_g1
ext_ref_range	[5:3]	RWS,0	输入的参考时钟频率范围
ext_ref_clk_mpll_div2_en	[2]	RWS,0	用于MPLLb计量和锁定的参考时钟2分频
ext_ref_clk_mplla_div2_en	[1]	RWS,0	用于MPLLA计量和锁定的参考时钟2分频
ext_ref_clk_div2_en	[0]	RWS,0	参考时钟被2分频
—	其它	—	保留。

5.1.2.32 PMAControl5

偏移: 16'h1380

PMA 控制寄存器 5, 可读写

表 5-62: PMAControl5 寄存器域的描述

名称	范围	类型	描述
ext_tx_iboost_lvl	[48:45]	RWS,0	设置 TX 端当前电压摆幅
ext_tx_eq_pre_g3	[44:41]	RWS,0	传输 Pre-Emphasis 等级调节控制
ext_tx_eq_pre_g2	[40:37]	RWS,0	传输Pre-Emphasis等级调节控制
ext_tx_eq_pre_g1	[36:33]	RWS,0	传输Pre-Emphasis等级调节控制
ext_tx_eq_post_g3	[32:29]	RWS,0	传输Post-Emphasis等级调节控制
ext_tx_eq_post_g2	[28:25]	RWS,0	传输Post-Emphasis等级调节控制
ext_tx_eq_post_g1	[24:21]	RWS,0	传输Post-Emphasis等级调节控制
ext_tx_eq_ovrd_g3	[20]	RWS,0	配置值override控制
ext_tx_eq_ovrd_g2	[19]	RWS,0	配置值override控制
ext_tx_eq_ovrd_g1	[18]	RWS,0	配置值override控制
ext_tx_eq_main_g3	[17:12]	RWS,0	传输振幅调节控制
ext_tx_eq_main_g2	[11:6]	RWS,0	传输振幅调节控制
ext_tx_eq_main_g1	[5:0]	RWS,0	传输振幅调节控制
—	其它	—	保留。

5.2 PCI-E 根部件内部的 IO 寄存器

PCI-E 根部件 (RC) 内部的 IO 寄存器遵从 PCI-E 3.0 规范, 按照 4B 对界编址。该节描述的 RC 内部寄存器的偏移以字节为单位且 4 字节对界, 以便与 PCI-E 规范保持一致。

5.2.1 RC寄存器空间布局

申威 421 处理器的 PCI-E RC 实现了一个功能 (function)，其包含 4096 字节的 PCI-E 配置空间。该配置空间可划分为：

- 2) 64 字节的 PCI 3.0 兼容配置空间头；
- 3) PCI 标准能力结构，起始偏移量 0x40；PCI 兼容配置空间头和 PCI 标准能力结构属于 PCI 配置空间；
- 4) PCI-E 扩展能力结构，起始偏移量 0x100；
- 5) 端口逻辑寄存器（申威 421 处理器自定义部分），起始偏移量 0x700。端口逻辑寄存器主要是用于测试。PCI-E 扩展能力结构和端口逻辑寄存器属于 PCI-E 扩展配置空间。

RC 配置空间的布局如图 5-1 所示：

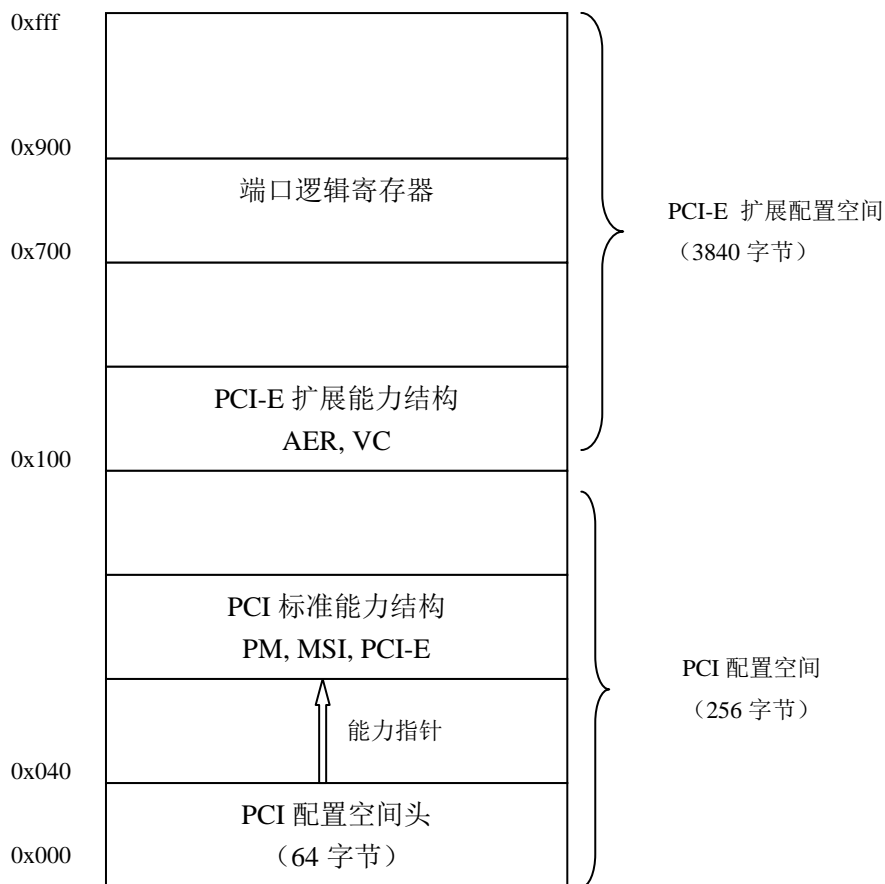


图 5-1 RC 配置空间布局

5.2.1.1 寄存器映射

5.2.1.1.1 PCI 配置空间头

PCI 配置空间头的寄存器域定义如表 5-59 所示。该配置空间在 PCI-E 规范中属于 TYPE1 类型的

配置空间（区别于 EP 的 TYPE0 类型配置空间）。大多数 PCI 兼容的寄存器域在 PCI 和 PCI-E 中有相同的软件含义。

表 5-59: PCI 配置空间头

字节偏移	字节 3	字节 2	字节 1	字节 0
0x00	设备 ID		厂商 ID	
0x04	状态寄存器		命令寄存器	
0x08	类代码			修订版 ID
0x0c	BIST	头类型	主延迟计时器	Cache 行大小
0x10	保留			
0x14	保留			
0x18	二级延迟计时器	从属总线号	二级总线号	主总线号
0x1c	二级状态		I/O 界限	I/O 基址
0x20	存储器界限		存储器基址	
0x24	可预取存储器界限		可预取存储器基址	
0x28	可预取基址高 32 位			
0x2c	可预取界限高 32 位			
0x30	I/O 界限高 16 位		I/O 基址高 16 位	
0x34	保留			能力指针
0x38	扩展 ROM 基址			
0x3c	桥控制		中断引脚	中断线

5.2.1.1.2 PCI 能力结构

申威 421 处理器支持三种 PCI 标准能力结构。每个结构都有自己的能力 ID (Capability ID)，且起始地址都由指针指定，这些指针就形成了一个链表，如表 5-60 所示。其中第一个能力结构为 PCI 电源管理能力，其起始地址在 PCI 兼容配置空间头偏移为 0x34 的寄存器（即能力指针）中指定，并有专门的寄存器指定下一个能力结构的起始地址；这样依次组成链表。

三种能力结构，及其起始地址和链表组织情况如表 5-60 所示，各种能力结构的寄存器列表如表 5-61~表 5-63 所示：

表 5-60: PCI 能力结构偏移和指针

起始偏移地址	能力名称	下一个能力指针
0x00	PCI 兼容头 (类型 1)	0x40
0x40	PCI 电源管理能力	0x50
0x50	MSI 中断能力	0x70

0x70	PCI-E 能力	0xb0
------	----------	------

表 5-61: 电源管理能力结构

字节偏移	字节 3	字节 2	字节 1	字节 0
0x40	电源管理能力		下一能力指针 (0x50)	能力 ID (0x01)
+0x04	电源管理控制状态寄存器			

表 5-62: MSI 能力结构

字节偏移	字节 3	字节 2	字节 1	字节 0
0x50	保留		下一能力指针 (0x70)	能力 ID (0x05)

表 5-63: PCI-E 能力结构

字节偏移	字节 3	字节 2	字节 1	字节 0
0x70	PCI-E 能力寄存器		下一能力指针 (0xB0)	能力 ID (0x10)
+0x4	设备能力			
+0x8	设备状态		设备控制	
+0xC	链路能力			
+0x10	链路状态		链路控制	
+0x14	插槽能力			
+0x18	插槽状态		插槽控制	
+0x1C	根能力			
+0x20	根状态		根控制	

5.2.1.1.3 PCI-E 扩展能力结构

申威 421 处理器支持两种 PCI-E 扩展能力结构。每个结构都有自己的能力 ID (Capability ID)，且组织方式类似于 PCI 标准能力结构。第一个能力结构为高级错误报告能力结构，其在 RC 配置空间中的偏移量是 0x100，并有指针指向下一个 PCI-E 扩展能力结构。

两种能力结构，及其起始地址和链表组织情况如表 5-64 所示，各种能力结构的寄存器列表如表 5-65~表 5-66 所示：

表 5-64: PCI-E 扩展能力结构偏移和指针

起始偏移地址	能力名称	下一个能力指针
0x100	高级错误报告能力	0x140
0x140	虚通道能力	—

表 5-65: 高级错误报告能力结构

字节偏移	字节 3	字节 2	字节 1	字节 0
0x100	PCI-E 扩展能力头			
+0x4	不可纠正错状态寄存器			
+0x8	不可纠正错屏蔽寄存器			
+0xC	不可纠正错严重性寄存器			
+0x10	可纠正错状态寄存器			
+0x14	可纠正错屏蔽寄存器			
+0x18	高级错能力和控制寄存器			
+0x1C ~ +0x28	头记录寄存器			
+0x2C	根错误命令寄存器			
+0x30	根错误状态寄存器			
+0x34	错误源标志寄存器			

表 5-66: 虚通道能力结构

字节偏移	字节 3	字节 2	字节 1	字节 0
0x140	PCI-E 扩展能力头			

注: 申威 421 处理器的 RC 中只实现了 1 个虚通道, 即虚通道 0。

5.2.1.1.4 端口逻辑寄存器

端口逻辑寄存器是申威 421 处理器自定义的寄存器, 其映射关系如表 5-67 所示。

表 5-67: 端口逻辑寄存器映射

序号	名称	索引地址
1	ACK 延迟计时和重发计时寄存器	0x700
2	其它消息寄存器	+0x4
3	端口强制链路寄存器	+0x8
4	ACK 频率寄存器	+0xC
5	端口链路控制寄存器	+0x10
6	通道偏斜寄存器	+0x14
7	计数器控制和最大功能号寄存器	+0x18
8	符号计时寄存器和过滤屏蔽寄存器 1	+0x1C
9	过滤屏蔽寄存器 2	+0x20
10	传输 P 类事务流量控制信用状态寄存器	+0x30
11	传输 NP 类事务流量控制信用状态寄存器	+0x34

12	传输完成事务流量控制信用状态寄存器	+0x38
13	队列状态寄存器	+0x3C
14	Gen2 控制寄存器	+0x10C
15	Gen3 控制寄存器	+0x190
16	均衡控制寄存器	+0x1A8
17	顺序规则寄存器	+0x1B4
18	只读域写使能寄存器	+0x1BC
19	LUT 删除寄存器	+0x1C8
其它	保留	—

5.2.1.2 RC 配置空间寄存器的访问

本章描述 RC 配置空间内部的寄存器偏移是按照 cache 行对界的，在实际访问中，软件可以直接使用表 5-2 中的基址+偏移地址，作为访问地址。本手册所列地址偏移地址与 PCIE 协议规范的地址对应关系为：本手册地址右移 5 位，比如手册偏移地址为 0x700，其协议规范地址为 0x38。

尽管 RC 配置空间中有些寄存器域是只读类型（HwInit、RO、ROS 等），但是软件对这些寄存器的写操作依然会成功，且返回正常响应。

5.2.2 PCI兼容配置头寄存器

5.2.2.1 设备 ID 和厂商 ID 寄存器

偏移：0x0000。

表 5-68：设备 ID 和厂商 ID 寄存器描述

名称	范围	类型	描述
DeviceID	[31:16]	RO,0xABCD	设备 ID。
VendorID	[15:0]	RO,0x16C3	厂商 ID。

5.2.2.2 命令寄存器

偏移：0x0080。

字节：0-1。

表 5-69：命令寄存器描述

名称	范围	类型	描述
INTxDisable	[10]	RW,0	INTx 断言屏蔽 (INTx Assertion Disable)。
FastB2BEn	[9]	RO,0	快速背靠背使能。不适用于 PCI-E, 必须固定为“0”。
SERREn	[8]	RW,0	SERR#使能。
IDSEL	[7]	RO,0	IDSEL Stepping/ Wait Cycle Control。不适用于 PCI-E, 必须固定为“0”。
ParErrRsp	[6]	RW,0	奇偶校验错响应。
VGASnoop	[5]	RO,0	VGA 调色监听。不适用于 PCI-E, 必须固定为“0”。
MemWrInv	[4]	RO,0	存储器写和置无效。不适用于 PCI-E, 必须固定为“0”。
SpecialCycleEn	[3]	RO,0	特殊周期使能。不适用于 PCI-E, 必须固定为“0”。
BusMasterEn	[2]	RW,0	总线主设备使能。
MemSpaceEn	[1]	RW,0	存储器空间使能。
IOSpaceEn	[0]	RW,0	I/O 空间使能。
—	其它	—	保留。

5.2.2.3 状态寄存器

偏移: 0x0080。

字节: 2-3。

表 5-70: 状态寄存器描述

名称	范围	类型	描述
ParityErr	[15]	RW1C,0	检测到奇偶校验错。
SignaledErr	[14]	RW1C,0	信号系统错误 (Signaled System Error)。
RcvMasterAbort	[13]	RW1C,0	接收到主设备异常中止。
RcvTrgtAbort	[12]	RW1C,0	接收到目标方异常中止。
SignalTrgtAbort	[11]	RW1C,0	信号目标异常中止 (Signaled Target Abort)。
DEVSEL	[10:9]	RO,0	DEVSEL Timing。不适用于 PCI-E,
MasterDataParErr	[8]	RW1C,0	主设备数据奇偶校验错。
FastB2BCap	[7]	RO,0	快速背靠背是否有效。不适用于 PCI-E, 硬连线为“0”。
66MhzCap	[5]	RO,0	66 Mhz 是否可用。不适用于 PCI-E, 硬连线为“0”。
CapList	[4]	RO,1	能力列表。表示存在扩展能力, 硬连线为“1”。
INTxStat	[3]	RO,0	INTx 状态。

IMM_READ INESS	[0]	RO,0	Immediate Readiness
—	其它	—	保留。

5.2.2.4 修订版 ID 寄存器

偏移：0x0100。

字节：0。

表 5-71：修订版 ID 寄存器描述

名称	范围	类型	描述
RevisionID	[7:0]	RO,0x01	修订版本。

5.2.2.5 类代码寄存器

偏移：0x0100。

字节：1-3。

表 5-72：类代码寄存器描述

名称	范围	类型	描述
BaseClassCode	[23:16]	RO, 0x00	基本类代码。
SubClassCode	[15:8]	RO, 0x00	子类代码。
ProgramInf	[7:0]	RO, 0x00	编程接口。

5.2.2.6 Cache 行大小寄存器

偏移：0x0180。

字节：0。

表 5-73：Cache 行大小寄存器描述

名称	范围	类型	描述
CacheLineSize	[7:0]	RW, 0x00	Cache行大小。该寄存器是为了兼容PCI设备，并不适用于PCI-E设备；修改该域不影响功能。

5.2.2.7 主延迟计时器寄存器

偏移：0x0180。

字节：1。

表 5-74：主延迟计时器寄存器描述

名称	范围	类型	描述
MasterLatencyTimer	[7:0]	RO,0	主延迟计数器。不适用于PCI-E设备，硬连线为“0”。

5.2.2.8 头类型寄存器

偏移：0x0180。

字节：2。

表 5-75：头类型寄存器描述

名称	范围	类型	描述
MultiFunc	[7]	RO,0	多功能设备标志。
HeaderFmt	[6:0]	RO,0x1	配置头格式。硬连线为 0x01。

5.2.2.9 BIST 寄存器

偏移：0x0180。

字节：3。

表 5-76：BIST 寄存器描述

名称	范围	类型	描述
BIST	[7:0]	RO,0x0	申威 421 处理器不支持 RC 的 BIST 寄存器功能，读该寄存器将返回全“0”。

5.2.2.10 总线号寄存器

偏移：0x0300。

表 5-77：总线号寄存器描述

名称	范围	类型	描述
2ndLatencyTimer	[31:24]	RO, 0x00	二级延迟计时器。不适用于 PCI-E，硬连线为 0x00。
Subordinate	[23:16]	RW, 0x00	从属总线号。
Secondary	[15:8]	RW, 0x00	二级总线号。
Primary	[7:0]	RW, 0x00	主总线号。

5.2.2.11 I/O 基址和界限寄存器

偏移：0x0380。

字节：0-1。

表 5-78：I/O 基址和界限寄存器描述

名称	范围	类型	描述
IOLimit	[15:12]	RW, 0x00	I/O 空间界限。
32bitIO	[8]	RO, 0	32 位 I/O 空间。 “0”：16 位 I/O 编址寻址； “1”：32 位 I/O 编址寻址。
IOBase	[7:4]	RW, 0x00	I/O 空间基址。
32bitIO	[0]	RO, 0	32 位 I/O 空间。 “0”：16 位 I/O 编址寻址； “1”：32 位 I/O 编址寻址。 该位被修改时，第[8]位也一同被修改。
—	其它	—	保留。

5.2.2.12 二级状态寄存器

偏移：0x0380。

字节：2-3。

表 5-79：二级状态寄存器描述

名称	范围	类型	描述
ParityErr	[15]	RW1C, 0	检测到奇偶校验错。
RcvSysErr	[14]	RW1C, 0	接收系统错。
RcvMasterAbort	[13]	RW1C, 0	接收到主设备异常中止。
RcvTrgtAbort	[12]	RW1C, 0	接收到目标方异常中止。
SignalTrgtAbort	[11]	RW1C, 0	信号目标异常中止。
DEVSEL	[10:9]	RO, 0	DEVSEL Timing。不适用于 PCI-E，硬连线为“0”。
MasterDataParErr	[8]	RW1C, 0	主设备数据奇偶校验错。
FastB2BCap	[7]	RO, 0	快速背靠背使能。不适用于 PCI-E，硬连线为“0”。
66MhzCap	[5]	RO, 0	66 Mhz 是否可用。不适用于 PCI-E，硬连线为“0”。

—	其它	—	保留。
---	----	---	-----

5.2.2.13 存储器基址和界限寄存器

偏移：0x040020。

表 5-80：存储器基址和界限寄存器描述

名称	范围	类型	描述
MemLimit	[31:20]	RW, 0x00	存储器界限地址。
MemBase	[15:4]	RW, 0x00	存储器空间基址。
—	其它	—	保留。

5.2.2.14 可预取存储器基址和界限寄存器

偏移：0x0480。

表 5-81：可预取存储器基址和界限寄存器描述

名称	范围	类型	描述
Upper12bitEndAddr	[31:20]	RW,0x000	32 位可预取存储器空间结束地址的高 12 位。
64bitMem	[16]	RO,0	存储器空间编址标志。 “0”：32 位编址寻址； “1”：64 位编址寻址。
Upper12bitStartAddr	[15:4]	RW,0x000	32 位可预取存储器空间起始地址的高 12 位。
64bitMem	[0]	RO,0	存储器空间编址标志。 “0”：32 位编址寻址； “1”：64 位编址寻址。 该位被修改时，第[16]位也一同被修改。
—	其它	—	保留。

5.2.2.15 可预取存储器基址高 32 位寄存器

偏移：0x0500。

表 5-82：可预取基址高 32 位寄存器描述

名称	范围	类型	描述
UpperBaseAddr	[31:0]	RW,0x00000000	可预取存储器空间基址的高 32 位。只有当可预取存储器空间编址为 64 位模式时有效。

5.2.2.16 可预取存储器界限高 32 位寄存器

偏移：0x0580。

表 5-83：可预取存储器界限高 32 位寄存器描述

名称	范围	类型	描述
UpperLimitAddr	[31:0]	RW,0x00000000	可预取存储器空间界限地址的高 32 位。只有当可预取存储器空间编址为 64 位模式时有效。

5.2.2.17 I/O 基址和界限高 16 位寄存器

偏移：0x0600。

表 5-84：I/O 基址和界限高 16 位寄存器描述

名称	范围	类型	描述
UpperLimitAddr	[31:16]	RW,0x0000	I/O 空间界限的高 16 位（如果二级设备支持 32 位 I/O 解码）。
UpperBaseAddr	[15:0]	RW,0x0000	I/O 空间基址的高 16 位（如果二级设备支持 32 位 I/O 解码）。

5.2.2.18 能力指针寄存器

偏移：0x0680。

字节：0。

表 5-85：功能指针寄存器描述

名称	范围	类型	描述
FirstCapPtr	[7:0]	RO,0x40	第一个能力指针。默认指向电源管理能力结构。

5.2.2.19 扩展 ROM 基址寄存器

偏移：0x0700。

表 5-86：扩展 ROM 基址寄存器寄存器描述

名称	范围	类型	描述
Addr	[31:11]	RW,0x00000	扩展 ROM 地址。
En	[0]	RW,0	扩展 ROM 使能。
—	其它	—	保留。

5.2.2.20 中断线寄存器

偏移：0x0780。

字节：0。

表 5-87：中断线寄存器描述

名称	范围	类型	描述
IntLine	[7:0]	RW,0xFF	中断线。其值因系统结构而异，当初始化或配置系统的时候，软件会写入路由信息。

5.2.2.21 中断引脚寄存器

偏移：0x0780。

字节：1。

表 5-88：中断引脚寄存器描述

名称	范围	类型	描述
IntPin	[7:0]	RO,0x01	中断引脚。指定设备（或者设备功能）使用的中断消息。 有效值如下： 0x00：设备（或者功能）没有使用 INTx 中断； 0x01：设备（或者功能）使用了 INTA 中断； 0x02：设备（或者功能）使用了 INTB 中断； 0x03：设备（或者功能）使用了 INTC 中断； 0x04：设备（或者功能）使用了 INTD 中断。

5.2.2.22 桥控制寄存器

偏移：0x0780。

字节：2-3。

表 5-89：桥控制寄存器描述

名称	范围	类型	描述
2ndBusRst	[6]	RW,0	二级总线复位。
MstAbtMode	[5]	RO,0	主设备异常中止模式。不适用于 PCI-E，硬连线为“0”。
VGA16Dec	[4]	RW,0	VGA 16 位解码。
VGAEn	[3]	RW,0	VGA 使能

ISAEn	[2]	RW,0	ISA 使能。
SERREn	[1]	RW,0	SERR 使能。
ParErrRspEn	[0]	RW,0	奇偶校验错响应使能。
—	其它	—	保留。

5.2.3 PCI标准能力结构寄存器

申威 421 处理器支持三种 PCI 标准能力结构：

- 1) PCI 电源管理能力；
- 2) MSI 中断能力；
- 3) PCI-E 能力。

这三种能力结构中，需要重点关注的是 PCI-E 能力结构的寄存器。

5.2.3.1 PCI 电源管理能力

PCI配置头里的能力指针指向PCI电源管理能力结构。该能力结构包含以下寄存器：

5.2.3.1.1 电源管理能力 ID 寄存器

偏移：0x0800。

字节：0。

表 5-90：电源管理能力 ID 寄存器描述

名称	范围	类型	描述
PMCapID	[7:0]	RO,0x01	电源管理能力 ID。

5.2.3.1.2 电源管理下一能力指针

偏移：0x0800。

字节：1。

表 5-91：电源管理下一能力指针寄存器描述

名称	范围	类型	描述
NextCapPtr	[7:0]	RO,0x50	下一能力指针。默认指向 MSI 能力结构。

5.2.3.1.3 电源管理能力寄存器

偏移：0x0800。

字节：2-3。

表 5-92：电源管理能力寄存器描述

名称	范围	类型	描述
PME_Support	[15:11]	RO,0x1B	是否支持电源管理事件（PME，Power Management

			Event)。 指定哪些功耗状态下可以生成 PME 消息。只有为“1”时该位对应的设备电源管理状态才可以产生 PME 消息： [11]: D0 状态是否可产生 PME 消息； [12]: D1 状态是否可产生 PME 消息； [13]: D2 状态是否可产生 PME 消息； [14]: D3hot状态是否可产生PME消息； [15]: D3cold 状态是否可产生 PME 消息。 第[11]、[14]、[15]位必须设置。
D2Support	[10]	RO,1	是否支持 D2 电源管理状态。
D1Support	[9]	RO,1	是否支持 D1 电源管理状态。
AuxCurrent	[8:6]	RO,0x7	AUX 电流。
DSI	[5]	RO,0	设备特定初始化 (DSI, Device Specific Initialization)。
IMM_READI_RETU RN_D0	[4]	RO,0	Immediate Readiness on Return to D0
PMEClock	[3]	RO,0	PME 时钟。硬连线为“0”。
PMVersion	[2:0]	RO,0x3	电源管理规格版本 (Power Management Specification Version)。
—	其它	—	保留。

5.2.3.1.4 电源管理控制和状态寄存器

偏移: 0x0880。

表 5-93: 电源管理控制和状态寄存器描述

名称	范围	类型	描述
DATA_REG	[31:24]	RO,0	电源数据信息寄存器
ClkCtrlEn	[23]	RO,0	总线电源/时钟控制使能。硬连线为“0”。
B23Support	[22]	RO,0	是否支持 B2/B3。硬连线为“0”。
PMEStatus	[15]	RW1CS,0	PME 状态。指示早先使能的 PME 事件是否发生。
DATA_SCALE	[14:13]	RO,0	数据刻度因素
DATA_SELECT	[12:9]	RO,0	数据选择
PMEEEn	[8]	RWS,0	PME 使能。设为“1”表示设备被使能可以产生 PME。
NoSoftRst	[3]	RO,0	无软件复位。
PowerState	[1:0]	RW,0x0	配置设备的电源管理状态： 0b00 : D0 状态，正常工作状态；

			0b01 : D1 状态, 可选, 低功耗状态; 0b10 : D2 状态, 可选, 低功耗状态; 0b11 : D3 状态, 最低功耗状态。 当特定状态不支持时, 写入的值被忽略。
—	其它	—	保留。

5.2.3.2 MSI 能力

5.2.3.2.1 MSI 能力 ID

偏移: 0x0a00。

字节: 0。

表 5-94: MSI 能力 ID 寄存器描述

名称	范围	类型	描述
MSICapID	[7:0]	RO,0x05	MSI 能力 ID。

5.2.3.2.2 MSI 下一能力指针

偏移: 0x0a00。

字节: 1。

表 5-95: MSI 下一能力指针寄存器描述

名称	范围	类型	描述
NextCapPtr	[7:0]	RO,0x70	下一功能指针。默认指向 PCI-E 能力。

5.2.3.2.2.1 MSI 控制寄存器

偏移: 0x0a00。

字节: 2-3。

表 5-96: MSI 控制描述

名称	范围	类型	描述
PVM_SUPPO RT	[8]	R,1	MSI Per Vector Masking Capable
64b AC	[7]	RO, 1	能否使用 64 位地址, DBI 可写。
MultiMsg En	[6:4]	RW, 0x0	多消息使能, 即分配的 MSI 中断的数量。 表示系统软件使能多消息模式。使能的消息数量不能大于 MultiMsg_Cap 的值。
MultiMsg_Cap	[3:1]	RO, 0x0	可发送的消息数目, DBI 可写。
MSI En	[0]	RW, 0	MSI 使能, 设为“1”时, 必须保证 INTx 无效。

—	其它	—	保留。
---	----	---	-----

5.2.3.2.2.2 MSI 低 32 位地址寄存器

偏移：0x0a80。

表 5-97：MSI 低 32 位地址描述

名称	范围	类型	描述
L32b Addr	[31:2]	RW, 0x0	低 32 位地址。
—	其它	—	保留。

5.2.3.2.2.3 MSI 高 32 位地址寄存器

偏移：0x0b00。

表 5-98：MSI 高 32 位地址描述

名称	范围	类型	描述
L32b Addr	[31:0]	RW, 0x0	高 32 位地址。

5.2.3.2.2.4 MSI 数据寄存器

偏移：0x0b80。

表 5-99：MSI 数据描述

名称	范围	类型	描述
MSI Data	[15:0]	RW, 0x0	MSI 数据。由系统软件指定模式，[4:0]为与 MSI_VECTOR 位或，每个功能产生 32MSI 消息。
—	其它	—	保留。

5.2.3.2.2.5 MSI 屏蔽位寄存器

偏移：0x0c00

名称	范围	类型	描述
MSI_MASK_BITS	[31:0]	V,0	MSI 屏蔽位
—	其它	—	保留。

5.2.3.2.2.6 MSI 悬挂位寄存器

偏移：0x0c80

名称	范围	类型	描述
----	----	----	----

MSI_PENDING_BITS	[31:0]	RO,0	MSI 悬挂位
—	其它	—	保留。

5.2.3.3 PCI-E 能力

5.2.3.3.1 PCI-E 能力列表寄存器

偏移：0x0e00。

字节：0-1。

表 5-100: PCI-E 能力列表寄存器描述

名称	范围	类型	描述
NextCapPtr	[15:8]	RO,0xB0	下一能力指针。默认指向 MSI-X 能力，但是申威 421 处理器未实现 MSI-X 能力结构。
PCIECapID	[7:0]	RO,0x10	PCI-E 能力 ID。

5.2.3.3.2 PCI-E 能力寄存器

偏移：0x0e00。

字节：2-3。

表 5-101: PCI-E 能力寄存器描述

名称	范围	类型	描述
IntMsgNum	[13:9]	RW,0x0	电源管理事件、或热插拔事件所触发的 MSI 中断所对应的中断向量号。
Slot	[8]	HwInit,0	是否实现了插槽。
DevPortType	[7:4]	RO,0x4	设备端口类型。
PCIECapVersion	[3:0]	RO,0x2	PCI-E 能力版本。
—	其它	—	保留。

5.2.3.3.3 设备能力寄存器

偏移：0x0e80。

表 5-102: 设备能力寄存器描述

名称	范围	类型	描述
RoleErrRpt	[15]	RO,1	基于角色的错误报告 (Role-Base Error Reporting)。
ExtTagSupport	[5]	RO,0	是否支持扩展标签 (tag) 域。必须固定为“0”。
PhantomFuncSupport	[4:3]	RO,0	是否支持仿真功能 (Phantom Function)。必须固定为“0”。

MaxPayloadSize	[2:0]	RO,0x2	支持的最大有效负载。其编码含义为： 0b000: 128B; 0b001: 256B; 0b010: 512B; 0b011: 1024B; 0b100: 2048B; 0b101: 4096B; 其它: 不支持。
—	其它	—	保留。

5.2.3.3.4 设备控制寄存器

偏移: 0x0f00。

字节: 0-1。

表 5-103: 设备控制寄存器描述

名称	范围	类型	描述
MaxReadReqSize	[14:12]	RW,0x2	最大读请求数据量。
NoSnoopEn	[11]	RW,1	不监听 (No Snoop) 使能。
AuxPwrPMEn	[10]	RW,0	辅助电源电源管理使能。
PhantomFuncEn	[9]	RW,0	仿真功能 (Phantom Function) 使能。
ExtTagEn	[8]	RW,0	扩展 tag 域使能。
MaxPadyloadSize	[7:5]	RW,0x0	最大有效负载数。
ROEn	[4]	RW,1	宽松的序 (Relaxed Ordering) 使能。
URRptEn	[3]	RW,0	不支持的请求 (Unsupported Request) 报告使能。
FatalErrRptEn	[2]	RW,0	致命错误报告使能。
NFatalErrRptEn	[1]	RW,0	非致命错误报告使能。
CErrRptEn	[0]	RW,0	可纠正错误报告使能。
—	其它	—	保留。

5.2.3.3.5 设备状态寄存器

偏移: 0x0f00。

字节: 2-3。

表 5-104: 设备状态寄存器描述

名称	范围	类型	描述
TransPend	[5]	RO,0	事务悬挂。硬连线为“0”。
AuxPwrDtct	[4]	RO,0	检测到AUX电源。

URDtct	[3]	RW1C,0	检测到不支持的请求。
FatalErrDtct	[2]	RW1C,0	检测到致命错误。检测到致命错误后直接登记到该域，而不管高级错误报告有没有使能（设备控制寄存器的 FatalErrRptEn 位）。
NfatalErrDtct	[1]	RW1C,0	检测到非致命错误。检测到非致命错误后直接登记到该域，而不管高级错误报告有没有使能（设备控制寄存器的 NFatalErrRptEn 位）。
CerrDtct	[0]	RW1C,0	检测到可修正错误。检测到可纠正错误后直接登记到该域，而不管高级错误报告有没有使能（设备控制寄存器的 CErrRptEn 位）。
—	其它	—	保留。

5.2.3.3.6 链路能力寄存器

偏移：0x0f80。

表 5-105：链路能力寄存器描述

名称	范围	类型	描述
PortNum	[31:24]	HwInit,0x00	端口数。
ASPMOptionCompliance	[22]	RO,1	ASPM 可选兼容
LinkBWNnote	[21]	RO,1	链路带宽通告能力。硬连线为“1”。
DLLActiveRpt	[20]	RO,1	数据链接层主动报告能力。硬连线为“1”。
SuppriseDownErrRep	[19]	RO,0	意外的下行错误报告能力
ClkPM	[18]	RO,0	时钟电源管理。在指示链路处于L1、L2、L3这些低功耗状态时是否可以移除参考时钟。
L1ExitLatency	[17:15]	RO,0x4	L1状态退出延时。编码含义为： 0x0: 少于1us; 0x1: 1us~2us; 0x2: 2us~4us; 0x3: 4us~8us; 0x4: 8us~16us; 0x5: 16us~32us; 0x6: 32us~64us; 0x7: 多于64us。
L0sExitLatency	[14:12]	RO,0x6	L0s状态退出延时。编码含义为： 0x0: 少于64ns; 0x1: 64ns~128ns;

			0x2: 128ns~256ns; 0x3: 256ns~512ns; 0x4: 512ns~1us; 0x5: 1us~2us; 0x6: 2us~4us; 0x7: 多于4us。
ASPM_Support	[11:10]	RO,0x3	主动链路电源管理（ASPM, Active State Link Power Management）支持。编码含义为： 0x1: 支持L0s状态下的ASPM; 0x3: 支持L0s和L1状态下的ASPM; 其它: 保留。
MaxLinkWidth	[9:4]	RO,0x8	最大链接宽度。
MaxLinkSpeed	[3:0]	RO,0x3	最大链接速度。编码含义： 0b0001: 支持2.5Gbps速率; 0b0010: 支持5.0Gbps和2.5Gbps速率; 0b0011: 支持8.0Gbps及以下速率 其它: 保留。
—	其它	—	保留。

5.2.3.3.7 链路控制寄存器

偏移: 0x1000。

字节: 0-1。

表 5-106: 链路控制寄存器描述

名称	范围	类型	描述
LinkAutoBWIntEn	[11]	RO,0	链路自动调整带宽中断使能。 设置时会使能该类型中断的生成, 来指示链接自动调整带宽状态位已经被设置。
LinkBWMgtIntEn	[10]	RO,0	链路带宽管理中断使能。 设置时会使能该类型中断的生成, 来指示链路带宽管理状态位已经被设置。
ClkPMEn	[8]	RW,0	时钟电源管理使能。 如果链路能力寄存器中 ClkPM 位为“0”, 该位也为“0”。
ExtSsynch	[7]	RW,0	扩展同步。
ComClk	[6]	RW,0	公共时钟配置。
RetrainLink	[5]	RW,0	重新训练链路。

LinkDisable	[4]	RW,0	链路禁用。
RCB	[3]	RO,1	读完成边界（Read Completion Boundary）。编码含义为： 0: 64B； 1: 128B。
ASPMCtrl	[1:0]	RW,0	主动链路电源管理控制。 0b00: Disable 0b01: L0s 进入使能 0b10: L1 进入使能 0b11: L0s 和 L1 进入使能
—	其它	—	保留。

5.2.3.3.8 链路状态寄存器

偏移：0x1000。

字节：2-3。

表 5-107: 链路状态寄存器描述

名称	范围	类型	描述
LinkAutoBWStatus	[15]	RW1C,0	链路自动调整带宽状态。 该位表示硬件自动调节了链路速度或宽度。如果物理层报告速度或者宽度因下游设备发起而发生改变，那么该位一定要置为“1”。
LinkBWMgtStatus	[14]	RW1C,0	链路带宽管理状态。 该位由硬件设置，表示以下情况之一发生： （1）将链路控制寄存器的 RetrainLink 位设为“1”之后，链路重新训练完成； （2）硬件改变链路速度或宽度来纠正不可靠的链路运行。如果物理层报告速度或者宽度因下游设备发起而发生改变，那么该位必须置为“1”。
DLLAactive	[13]	RO,0	数据链接层运作（active）。 如果实现了链路能力寄存器的DLLActiveRpt位，则该位也必须实现。
SlotClkConfig	[12]	HwInit,1	插槽时钟配置。 表示该器件使用的是连接器提供的相同物理参考时钟。
LinkTraining	[11]	RO,0	链路训练中。
NegLinkWidth	[9:4]	RO,0x1	协商的链接宽度。

			在链路初始化之后由硬件自动设置，链路起来（link up）之前该域无意义。 0b 00 0001: x1 0b 00 0010: x2 0b 00 0100: x4 0b 00 1000: x8 0b 00 1100: x12 0b 01 0000: x16 0b 10 0000: x32
LinkSpeed	[3:0]	RO,0x1	链路速率。 在链路初始化之后由硬件自动设置。链路起来（link up）之前该域无意义。编码含义为： 0b0001: Gen 1, 2.5Gbps; 0b0010: Gen 2, 5.0Gbps; 0b0011: Gen 3, 8.0Gbps; 其它：保留。
—	其它	—	保留。

5.2.3.3.9 插槽能力寄存器

偏移：0x1080。

表 5-108: 插槽能力寄存器描述

名称	范围	类型	描述
PhySlotNum	[31:19]	HwInit,0x00	物理插槽数目。
NoCmdCplSupport	[18]	HwInit,0	支持无命令完成。该位为“1”表示热插拔控制器完成了一条命令后，插槽不会产生软件提示。
ElecInterlockPresent	[17]	HwInit,0	存在电气互锁。该位为“1”表示该插槽的底板上实现了电气互锁。
SlotPwrLmtScale	[16:15]	HwInit,0x0	插槽功耗限制比例（Slot Power Limit Scale）。该域的编码含义为： 0b00: 1.0x; 0b01: 0.1x; 0b10: 0.01x; 0b11: 0.001x。
SlotPwrLmtValue	[14:7]	HwInit,0x00	插槽功耗限制值，即物理插槽所支持的最大功耗。
HotPlugCap	[6]	HwInit,0	支持热插拔。为“1”表示插槽支持热插拔。

HotPlugSurprise	[5]	HwInit,0	突然热插拔。为“1”表示插槽允许设备在不提示软件的情况下移除。
PwrIndicatorPresent	[4]	HwInit,0	存在电源指示。为“1”表示插槽底板上实现了电源指示。
AttentionIndicatorPresent	[3]	HwInit,0	存在警告指示。为“1”表示插槽底板上实现了警告指示。
MRLSensorPresent	[2]	HwInit,0	存在手动保持门闩（MRL, Manually-operated Retention Latch）传感器指示。为“1”表示插槽底板上实现了 MRL 传感器。
PwrCtrlPresent	[1]	HwInit,0	存在电源控制器。为“1”表示插槽底板上实现了电源管理器。
AttentionButtonPresent	[0]	HwInit,0	存在警告按钮。为“1”表示插槽底板上实现了警告按钮。

5.2.3.3.10 插槽控制寄存器

偏移：0x1100。

字节：0-1。

表 5-109：插槽控制寄存器描述

名称	范围	类型	描述
DLLStateChangeEn	[12]	RW,0	数据链接层状态变化使能。
ElecInterlockCtrl	[11]	RW,0	电气互锁控制。
PwrCtrlrCtrl	[10]	RW,0	电源控制器控制。
PwrIndicatorCtrl	[9:8]	RW,0x3	电源指示器控制。
AttentionIndicatorCtrl	[7:6]	RW,0x3	警告指示器控制。
HPIntEn	[5]	RW,0	热插拔中断使能。
CmdCplIntEn	[4]	RW,0	命令完成中断使能。
PresenceDtctChangeEn	[3]	RW,0	在位检测变化使能。
MRLSensorChangeEn	[2]	RW,0	手动保持门闩传感器变化使能。
PwrFaultDtctEn	[1]	RW,0	电源故障检测使能。
AttentionButtonPressEn	[0]	RW,0	警告按钮按下使能。
—	其它	—	保留。

5.2.3.3.11 插槽状态寄存器

偏移：0x1100。

字节：2-3。

表 5-110：插槽状态寄存器描述

名称	范围	类型	描述
DLLStateChange	[8]	RW1C,0	数据链接层状态变化。
ElecInterlockStatus	[7]	RO,0	电气互锁状态。
PresenceDtctState	[6]	RO,0	存在检测状态。
MRLSensorState	[5]	RO,0	手动保持门闩传感器状态。
CmdCpl	[4]	RW1C,0	命令完成。
PresenceDtctChange	[3]	RW1C,0	存在检测变化。
MRLSensorChange	[2]	RW1C,0	手动保持门闩传感器变化。
PwrFaultDtct	[1]	RW1C,0	检测到电源故障。
AttentionButtonPress	[0]	RW1C,0	按下警告按钮。
—	其它	—	保留。

5.2.3.3.12 根控制寄存器

偏移：0x1180。

字节：0-1。

表 5-111：根控制寄存器描述

名称	范围	类型	描述
CRSSoftVisibilityEn	[4]	RO,0	CRS（Configuration Request Retry Status，即 EP 的配置返回未成功需要重试的响应）软件可视性使能。不支持，硬连线为“0”。
PMEIntEn	[3]	RW,0	电源管理事件中断使能。
SerrFatalErrEn	[2]	RW,0	致命错误上的系统错误使能。
SerrNFatalErrEn	[1]	RW,0	非致命错误上的系统错误使能。
SerrCErrEn	[0]	RW,0	可纠正错误上的系统错误使能。
—	其它	—	保留。

5.2.3.3.13 根能力寄存器

偏移：0x1180。

字节：2-3。

表 5-112：根能力寄存器描述

名称	范围	类型	描述
----	----	----	----

CRSSoftVisibility	[0]	HwInit,0	对 EP 的配置请求未成功时，软件是否可见。不支持，硬连线为“0”。
—	其它	—	保留。

5.2.3.3.14 根状态寄存器

偏移：0x1200。

表 5-113：根状态寄存器描述

名称	范围	类型	描述
PMEPend	[17]	RO,0	电源管理事件悬挂。
PMESstatus	[16]	RW1C,0	电源管理事件状态。
PMEReqID	[15:0]	RO,0x00	电源管理事件请求 ID。
—	其它	—	保留。

5.2.3.3.15 设备能力 2 寄存器

偏移：0x1280。

表 5-114：设备能力 2 寄存器描述

名称	范围	类型	描述
OBFFSupport	[19:18]	RO,0	Optimized Buffer Flush/fill Supported
LN_SYS_CLS	[15:14]	RO,0	LN System CLS
TPH_CMPLT_Support	[13:12]	RO,0	TPH Completer Supported Bit
LTR_Support	[11]	RO,0	LTR Mechanism Supported
NO_RO_EN_PR2PR _PAR	[10]	RO,0	No Relaxed Ordering Enabled PR-PR Passing
ARI_FWD_Support	[5]	RO,1	支持 ARI 转发
CplTimeoutDisSpt	[4]	RO,1	支持禁用响应超时。
CplTimeoutRangeSpt	[3:0]	HWInit,0xF	支持的响应超时范围。编码含义为： 0bxxx1：支持范围 A； 0bxx1x：支持范围 B； 0bx1xx：支持范围 C； 0b1xxx：支持范围 D。
—	其它	—	保留。

5.2.3.3.16 设备控制 2 寄存器

偏移：0x1300。

表 5-115：设备控制 2 寄存器描述

名称	范围	类型	描述
ARI_FWD_En	[5]	RO,0	ARI 转发使能
CplTimeoutDisable	[4]	RW,0	禁用完成超时。
CplTimeoutValue	[3:0]	RW,0x0	完成超时值。编码含义为： 0b0000: 50us~50ms，默认范围； 0b0001: 50us~100us； 0b0010: 1ms~10ms； 0b0101: 16ms~55ms； 0b0110: 65ms~210ms； 0b1001: 260ms~900ms； 0b1010: 1s~3.5s； 0b1101: 4s~13s； 0b1110: 17s~64s； 其它：保留。
—	其它	—	保留。

5.2.3.3.17 链路能力 2 寄存器

偏移：0x1380

名称	范围	类型	描述
DRS_SUPPORTED	[31]	RO,0	DRC 支持
LINK_SPEED_VEC	[7:1]	RO,7'h7	链路速度向量
—	其它	—	保留。

5.2.3.3.18 链路控制 2 寄存器

偏移：0x1400。

字节：0-1。

表 5-116: 链路控制 2 寄存器描述

名称	范围	类型	描述
CompliancePreset	[15:12]	RWS,0	8Gbps时，在LTSSM的轮询兼容状态设置发送方preset值。 5Gbps时，链路训练状态机处于Polling.Compliance状态时的去加重等级（De-emphasis Level）。编码含义为： 0: -6 dB； 1: -3.5 dB。 当链路工作在 2.5Gbps 时，该位无效。

ComplianceSOS	[11]	RWS,0	兼容性SKP有序集（SOS, SKP Ordered Set）。为“1”表示链路训练状态机需要在兼容性模式中发送SKP有序集。 当链路工作在2.5Gbps时，该位无效。
EnterMdfyCompliance	[10]	RW,0	进入修改兼容性（Modified Compliance）。为“1”表示当链路训练状态机进入 Polling.Compliance 状态时该RC 需要发送兼容性序列。 当链路工作在 2.5GT/s 时，该位无效。
TransmitMargin	[9:7]	RW,0x0	传送差额。该域控制发送端口非去加重电压等级的值（value of the non-de-emphasized voltage level），编码含义为： 0b000：全摆幅 800~1200 mV，半摆幅 400~600mV； 0b001- 0b010：值必须与非 0 偏斜单调一致； 0b011：全摆幅 200~400mV，半摆幅 100~200mV； 其它：保留。 工作在 5.0Gbps 为全摆幅时，去加重率必须维持在规定操作值（-3.5dB 或-6dB）的+/-1dB 之内。
SlctDeemph	[6]	RWS,0	可选择去加重。 链路工作在 5.0Gbps 时，选择去加重的等级： 0： -6 dB； 1： -3.5 dB。 当链路工作在 2.5GT/s 时，该位无效。
HwAutoSpeedDis	[5]	RO,0	禁止硬件自动调整速度。
EnterCompliance	[4]	RWS,0	进入兼容性模式。 软件可以强制链路按照 TrgtLinkSpeed 域指定的速率进入兼容性模式，并将会在链路上产生一个热复位。
TrgtLinkSpeed	[3:0]	RW,0x3	目标链路速率。 通过设置一个工作速率上限，来限制 RC 在链路训练序列（TS, Training Sequence）中的速率通告值。编码含义为： 0b0001： 2.5Gbps 的目标链路速率； 0b0010： 5Gbps 的目标链路速率； 0b0011： 8Gbps 的目标链路速率； 其它：保留。任何其余的值写入，会导致未知结果。

			该域指定了 RC 的最高链接速率值。
—	其它	—	保留。

5.2.3.3.19 链路状态 2 寄存器

偏移：0x1400。

字节：2-3。

表 5-117：链路状态 2 寄存器描述

名称	范围	类型	描述
LinkEqReq	[5]	RW1C,0	链路均衡请求 8GT/s
EqCplP3	[4]	RO,0	8GT/s 均衡第 3 步成功
EqCplP2	[3]	RO,0	8GT/s 均衡第 2 步成功
EqCplP1	[2]	RO,0	8GT/s 均衡第 1 步成功
EQ_CPL	[1]	RO,0	8GT/s 均衡完成
CurDeemphLevel	[0]	RO,1	当前去加重等级。 当链路工作在 5Gbps 或 8Gbps 时, 该位表示去加重等级, 编码含义为: 0: -6 dB; 1: -3.5 dB。 当链路工作在 2.5Gbps 时, 该位无效。
—	其它	—	保留。

5.2.4 PCI-E 扩展能力寄存器

申威 421 处理器处理器支持两种 PCI-E 扩展能力结构：

- (1) 高级错误报告能力；
- (2) 虚通道能力。

5.2.4.1 高级错误报告能力

5.2.4.1.1 PCI-E 扩展能力头

偏移：0x2000。

表 5-118：PCI-E 扩展能力头寄存器描述

名称	范围	类型	描述
NextCapOffset	[31:20]	RO,0x148	下一能力偏移, 指向虚通道能力结构。
CapVersion	[19:16]	RO,0x2	能力版本。
PCIExtCapID	[15:0]	RO,0x1	PCI-E 扩展能力 ID。

5.2.4.1.2 不可纠正错误状态寄存器

偏移：0x2080。

表 5-119：不可纠正错误寄存器描述

名称	范围	类型	描述
PrfxBlockErrStatus	[25]	RW1C,0	TLP 前缀阻塞错误状态
InternalErrStatus	[22]	RW1C,0	不可纠正内部错误状态
URrErrStatus	[20]	RW1CS,0	不支持请求错误状态。
ECRCrErrStatus	[19]	RW1CS,0	ECRC 错误状态。
MTLPStatus	[18]	RW1CS,0	畸形 TLP (Malformed TLP) 错误状态。
RcvOverflowStatus	[17]	RW1CS,0	接收溢出错误状态。
UnexpctCplStatus	[16]	RW1CS,0	非预期完成错误状态。
CASStatus	[15]	RW1CS,0	响应端异常中止错误状态。
CplTimeoutStatus	[14]	RW1CS,0	响应超时错误状态。
FCPrctlErrStatus	[13]	RW1CS,0	流控制协议错误状态。
PoisonTLPStatus	[12]	RW1CS,0	中毒 TLP (posioned TLP) 错误状态。
SurpriseDwnErrS	[5]	RW1C,0	意外下行错误状态
DLPErrStatus	[4]	RW1CS,0	数据链路协议错误状态。
—	其它	—	保留。

5.2.4.1.3 不可纠正错误屏蔽寄存器

偏移：0x2100。

表 5-120：不可纠正错误屏蔽寄存器描述

名称	范围	类型	描述
PrfxBlockErrMask	[25]	RWS,0	TLP 前缀阻塞错误屏蔽
InternalErrMask	[22]	RWS,0	不可纠正内部错误屏蔽
URrErrMask	[20]	RWS,0	不支持请求错误屏蔽。
ECRCrErrMask	[19]	RWS,0	ECRC 错误屏蔽。
MTLPMask	[18]	RWS,0	畸形 TLP 错误屏蔽。
RcvOverflowMask	[17]	RWS,0	接收溢出错误屏蔽。
UnexpctCplMask	[16]	RWS,0	非预期完成错误屏蔽。
CAMask	[15]	RWS,0	响应端异常中止错误屏蔽。
CplTimeoutMask	[14]	RWS,0	响应超时错误屏蔽。
FCPrctlErrMask	[13]	RWS,0	流控制协议错误屏蔽。
PoisonTLPMask	[12]	RWS,0	中毒 TLP 错误屏蔽。

SurpriseDwnErrM	[5]	RW1C,0	以外下行错误屏蔽
DLPerrMask	[4]	RWS,0	数据链路协议错误屏蔽。
—	其它	—	保留。

5.2.4.1.4 不可纠正错严重性寄存器

偏移：0x2180。

表 5-121：不可纠正错严重性寄存器描述

名称	范围	类型	描述
PrfxBlockErrSvrty	[25]	RWS,0	TLP 前缀阻塞错误严重性
InternalErrSvrty	[22]	RWS,0	不可纠正内部错误严重性
URerrSvrty	[20]	RWS,0	不支持请求错误严重性。
ECRCerrSvrty	[19]	RWS,0	ECRC 错误严重性。
MTLPSvrty	[18]	RWS,1	畸形 TLP 错误严重性。
RcvOverflowSvrty	[17]	RWS,1	接收溢出错误严重性。
UnexptCplSvrty	[16]	RWS,0	非预期完成错误严重性。
CASvrty	[15]	RWS,0	响应端异常中止错误严重性。
CplTimeoutSvrty	[14]	RWS,0	响应超时错误严重性。
FCPrctlErrSvrty	[13]	RWS,1	流控制协议错误严重性。
PoisonTLPSvrty	[12]	RWS,0	中毒 TLP 错误严重性。
SurpriseDwnErrSvrty	[5]	RW1C,0	以外下行错误严重性
DLPerrSvrty	[4]	RWS,1	数据链路协议错误严重性。
—	其它	—	保留。

5.2.4.1.5 可纠正错状态寄存器

偏移：0x2200。

表 5-122：可纠正错状态寄存器描述

名称	范围	类型	描述
HeadLogOverFlow	[15]	RW1C,0	头记录溢出错误状态
CorrctIntErrStatus	[14]	RW1C,0	可纠正的内部错误状态
NFfatalErrStatus	[13]	RW1CS,0	报告非致命错误状态。
ReplyTimeoutStatus	[12]	RW1CS,0	重发计时器超时状态。
ReplyNumStatus	[8]	RW1CS,0	重发次数翻转状态。
BadDLLPStatus	[7]	RW1CS,0	错误 DLLP(Data Link Layer Packet, 数据链路层数据包) 状态。

BadTLPStatus	[6]	RW1CS,0	错误 TLP (Transaction Layer Packet) 状态。
RcvErrStatus	[0]	RW1CS,0	接收端错误状态。
—	其它	—	保留。

5.2.4.1.6 可纠正错屏蔽寄存器

偏移: 0x2280。

表 5-123: 可纠正错屏蔽寄存器描述

名称	范围	类型	描述
HeadLogOverFlowMask	[15]	RW1C,0	头记录溢出错误屏蔽
CorrctIntErrMask	[14]	RW1C,0	可纠正的内部错误屏蔽
NFfatalErrMask	[13]	RWS,1	报告非致命错误屏蔽。
ReplyTimeoutMask	[12]	RWS,0	重发计时器超时屏蔽。
ReplyNumMask	[8]	RWS,0	重发次数翻转屏蔽。
BadDLLPMask	[7]	RWS,0	错误 DLLP 屏蔽。
BadTLPMask	[6]	RWS,0	错误 TLP 屏蔽。
RcvErrMask	[0]	RWS,0	接收端错误屏蔽。
—	其它	—	保留。

5.2.4.1.7 高级能力及控制寄存器

偏移: 0x2300。

表 5-124: 高级能力及控制寄存器描述

名称	范围	类型	描述
TLPPrfxLogPrst	[11]	RO,0	TLP 前缀记录
MultiHdrEn	[10]	RO,0	多个头记录使能
MutltiHdrEn	[9]	RO,0	多个头记录能力
ECRCCheckEn	[8]	RWS,0	ECRC 检查使能。
ECRCCheckCap	[7]	RO,1	ECRC 检查能力。
ECRCGenEn	[6]	RWS,0	ECRC 生成使能。
ECRCGenCap	[5]	RO,1	ECRC 生成能力。
FirstErrPtr	[4:0]	ROS,0	第一个错误指针。
—	其它	—	保留。

5.2.4.1.8 头记录寄存器

偏移: 0x2380~0x2500。

头记录寄存器记录检测到错误的 TLP 的包头, 并遵从 PCI-E 2.0 标准。头记录寄存器都是 ROS

类型：默认值为 0x00000000。

表 5-125：头记录寄存器描述

偏移	字节 3	字节 2	字节 1	字节 0
0x11C	头记录寄存器(包头第一个 32 位)。			
0x120	头记录寄存器(包头第二个 32 位)。			
0x124	头记录寄存器(包头第三个 32 位)。			
0x128	头记录寄存器(包头第四个 32 位)。			

5.2.4.1.9 根错误命令寄存器

偏移：0x2580。

表 5-126：根错误命令寄存器描述

名称	范围	类型	描述
FatalErrRptEn	[2]	RW,0	致命错误报告使能。
NfatalErrRptEn	[1]	RW,0	非致命错误报告使能。
CerrRptEn	[0]	RW,0	可纠正错报告使能。
—	其它	—	保留。

5.2.4.1.10 根错误状态寄存器

偏移：0x2600。

表 5-127：根错误状态寄存器描述

名称	范围	类型	描述
AERIntMsgNum	[31:27]	RW, 0	高级错误报告触发的 MSI 中断所对应的中断向量号。
FatalErrMsgRcv	[6]	RW1CS,0	接收到致命错误消息。
NFatalErrMsgRcv	[5]	RW1CS,0	接收到非致命错消息。
FirstUFatal	[4]	RW1CS,0	首个不可纠致命错。
MultiFatalNFatalErrRcv	[3]	RW1CS,0	接收到多个致命错 / 非致命错消息 (ERR_FATAL/NONFATAL)。
FatalNFatalErrRcv	[2]	RW1CS,0	接收到致命错/非致命错消息 (ERR_FATAL/NONTATAL)。
MultiCErrRcv	[1]	RW1CS,0	接收到多个错误纠正消息 (ERR_COR)。
CErrRcv	[0]	RW1CS,0	接收到错误纠正消息 (ERR_COR)。
—	其它	—	保留。

5.2.4.1.11 错误源标志寄存器

偏移: 0x2680。

表 5-128: 错误源标志寄存器描述

名称	范围	类型	描述
FatalNFatalSrc	[31:16]	ROS, 0	致命/非致命错消息 (ERR_FATAL/ NONTATAL) 源标志。
CErrSrc	[15:0]	ROS,0	可纠正错消息 (ERR_COR) 源标志。

5.2.4.1.12 TLP 前缀记录寄存器

偏移: 0x2700~0x2880。

TLP 前缀记录寄存器记录检测到错误的 TLP 的前缀, 并遵从 PCI-E 3.0 标准。TLP 前缀记录寄存器都是 ROS 类型; 默认值为 0x00000000。

偏移	字节 3	字节 2	字节 1	字节 0
0x2700	TLP 前缀记录寄存器(包头第一个 32 位)。			
0x2780	TLP 前缀记录寄存器(包头第二个 32 位)。			
0x2800	TLP 前缀记录寄存器(包头第三个 32 位)。			
0x2880	TLP 前缀记录寄存器(包头第四个 32 位)。			

5.2.4.2 虚通道能力

5.2.4.2.1 虚通道扩展能力包头

偏移: 0x2900。

表 5-129: 虚通道扩展能力包头寄存器域的描述

名称	范围	类型	描述
NextCapPtr	[31:20]	RO,0x000	下一能力偏移。
CapVersion	[19:16]	RO,0x1	能力版本。
PCIExtCapID	[15:0]	RO,0x2	虚通道能力所对应的扩展能力号。

5.2.5 端口逻辑寄存器

5.2.5.1 ACK 延迟计时和重发计时寄存器

偏移: 0xe000。

表 5-130: ACK 延迟计时和重发计时寄存器描述

名称	范围	类型	描述
ReplyTimeLimit	[31:16]	RWS, 129	重发延时上限。当重发计时器达到该上限后，就会超时。该上限值的计算取决于工作频率、链路宽度和最大负载（MaxPayloadSize）。该复位值是 x8 链路、MPS 为 512B 时的值。
ACKLatencyLimit	[15:0]	RWS, 43	ACK 确认回复间隔上限。当 Ack/Nak 延迟计时器达到该上限后，就会超时。该上限值的计算取决于工作频率、链路宽度和最大负载（MaxPadloadSize）。该复位值是 x8 链路、MPS 为 512B 时的值。

5.2.5.2 其它消息寄存器

偏移: 0xe080。

表 5-131: 其它消息寄存器描述

名称	范围	类型	描述
OtherMsg	[31:0]	RW,0xFFFF,FFFF	其它消息寄存器。该寄存器的作用是：要发送一个特定的 PCI-E 消息，可向该寄存器写入消息所携带的数据，然后设置端口链路控制寄存器（偏移为 0x710）的 OtherMsgReq 位（即第 0 位），则可发出该消息。

5.2.5.3 端口强制链路寄存器

偏移: 0xe100。

表 5-132: 端口强制链路寄存器描述

名称	范围	类型	描述
LowPwrEnterCnt	[31:24]	RWS,0x7	低功耗模式进入等待周期数。 不适用于 RC。
ForceLinkState	[21:16]	RWS,0x00	强制进入的链路状态。编码含义见附录 B。
ForceLinkEn	[15]	RW,0	强制链路进入指定状态。强制发送指定命令配置该域为 1 会生成脉冲信息，而导致链路训练状态机重新进行链路训练。读该域总是返回 0。
ForceLinkCmd	[11:8]	RW,0	强制发送的指定链路训练命令编码。
LaneNum	[7:0]	RWS,0x08	通道数目。

—	其它	—	保留。
---	----	---	-----

5.2.5.4 ACK 频率寄存器

偏移: 0xe180。

表 5-133: ACK 频率寄存器描述

名称	范围	类型	描述
EnterL1WithoutL0s	[30]	RWS,0	允许在链路另一端没有进入 L0s 的情况下, 即可发起进入 L1 低功耗状态。
L1EntryLatency	[29:27]	RWS,0x3	L1 进入延时。编码含义为: 0b000: 1 μ s; 0b001: 2 μ s; 0b010: 4 μ s; 0b011: 8 μ s; 0b100: 16 μ s; 0b101: 32 μ s; 其它: 64 μ s。
L0sEntryLatency	[26:24]	RWS,0x3	L0s 进入延时。编码含义为: 0b000: 1 μ s; 0b001: 2 μ s; 0b010: 3 μ s; 0b011: 4 μ s; 0b100: 5 μ s; 0b101: 6 μ s; 其它: 7 μ s。
ComFTSNum	[23:16]	RWS,180	使用共模时钟时, 从 L0s 迁移到 L0 状态时需要发送的 FTS 有序集的数目。
FTSNum	[15:8]	RWS,180	从 L0s 迁移到 L0 状态时需要发送的 FTS 有序集的数目。
AckFreq	[7:0]	RWS,0	指定 RC 累积的 Ack 消息数量, 达到该数量后 RC 会发送一个统一的 Ack 消息。
—	其它	—	保留。

5.2.5.5 端口链路控制寄存器

偏移：0xe200。

表 5-134：端口链路控制寄存器描述

名称	范围	类型	描述
LinkModeEn	[21:16]	RWS,x0F	链路模式使能。 0x1: x1; 0x3: x2; 0x7: x4; 0xF: x8; 0x1F: x16; 0x3F: x32 (不支持)。
FastLinkMode	[7]	RWS,0	快速链路模式。仅在硬件模拟时使用。
DLLLinkEn	[5]	RWS,1	数据链路层链路使能。设置该位，则进行流量控制信用初始化、并最终建立链路。
RstAssert	[3]	RWS,0	复位断言。配置该位为“1”则触发一个恢复并强制 LTSSM 进入热复位状态。
LoopbackEn	[2]	RWS,0	打开环回测试。
ScrambleDis	[1]	RWS,0	缠绕禁用。关闭数据缠绕。
OtherMsgReq	[0]	RWS,0	请求发送消息。 配置该位为“1”可以在 PCI-E 链路上发送消息，消息的内容存放在其它消息寄存器（偏移：0x704）中。

5.2.5.6 通道偏斜寄存器

偏移：0xe280。

表 5-135：通道偏斜寄存器描述

名称	范围	类型	描述
DisLane2LaneDeskew	[31]	RWS,0	禁用内部通道之间的去偏斜（lane-to-lane deskew）逻辑。
AckNakDis	[25]	RWS,0	禁止发送 Ack（Acknowledge，确认）和 Nak（Non Acknowledgement，）DLLP（Data Link Layer Packet，数据链路层数据包）。
FCDIS	[24]	RWS,0	禁止发送流量控制 DLLP。

InstLaneSkew	[23:0]	RWS,0x000000	在传输时插入通道偏斜。测试时使用。每条通道依次对应 3 位，且每个通道最多有 5 个符号时间的偏斜。
—	其它	—	保留。

5.2.5.7 计数器控制和最大功能号寄存器

偏移：0xe300

表 5-136：计数控制和最大功能号寄存器描述

名称	范围	类型	描述
FastLinkScaleFactor	[30:29]	RW,0	当端口链路寄存器中的 FastLinkMode 域为 1 时，设置 LTSSM 的计数器比例： 0: 1024 (1ms=1us) ; 1: 256 (1ms=4us) ; 2: 64 (1ms=16us) ; 3: 16 (1ms=64us)
TimerModAckNak	[23:19]	RW,0	增大 Ack/Nak 延迟计时器的数值，即 ACK 延迟计时和重发计时器（偏移为 0x700）的 RTLLimit 域。该域的单位是 64 个周期。
TimerModReplayTimer	[18:14]	RW,0	增大重发计时器的数值，即 ACK 延迟计时和重发计时器（偏移为 0x700）的 ReplyTimeLimit 域。该域的单位是 64 个周期（当 GEN3 速率时，单位是 256 个周期）。
MaxFuncNum	[7:0]	RW,0	在一个请求中能使用的最大的功能号，如果配置请求的功能号超过该值，会返回一个 UR 响应

5.2.5.8 符号计时寄存器和过滤屏蔽寄存器 1

偏移：0xe380。

表 5-137：符号计时寄存器和过滤屏蔽寄存器 1 描述

名称	范围	类型	描述
MaskFilter1	[31:16]	RWS,0x0000	使用屏蔽位来选择事务过滤和错误处理的规则。为“0”时，

			标识使用相关的过滤规则；为“1”时屏蔽相关的过滤规则。 各位所对应的规则为： [31]: 屏蔽配置请求的过滤； [30]: 屏蔽 I/O 请求的过滤； [29]: 屏蔽除厂商自定义消息以外的消息 TLP 的过滤； [28]: 屏蔽完成事务 ECRC 错误过滤； [27]: 屏蔽 ECRC 错误过滤； [26]: 屏蔽完成事务的长度不匹配错误； [25]: 屏蔽完成事务的属性不匹配错误； [24]: 屏蔽完成事务的业务等级 (Traffic Class) 不匹配错误； [23]: 屏蔽完成事务的功能不匹配错误； [22]: 屏蔽完成事务的请求者 ID 不匹配错误； [21]: 屏蔽完成事务的标签 (tag) 错误检查规则； [20]: 屏蔽锁读请求过滤； [19]: 屏蔽 Type1 类型配置请求过滤； [18]: 屏蔽基址寄存器匹配过滤； [17]: 屏蔽中毒 TLP 的过滤； [16]: 屏蔽请求功能不匹配过滤。
FCTimerDis	[15]	RWS,0	禁用流量控制的监视计时器。
SKPInterval	[10:0]	RWS,1280	SKP 有序集间的传输间隔。
—	其它	—	保留。

5.2.5.9 过滤屏蔽寄存器 2

偏移: 0xe400。

表 5-138: 过滤屏蔽寄存器 2 描述

名称	范围	类型	描述
MaskFilter2	[31:0]	RWS,0x00000000	使用屏蔽位来选择事务过滤和错误处理的规则。为“0”时，标识使用相关的过滤规则；为“1”时屏蔽相关的过滤规则。各位所对应的规则为： [5]: 屏蔽数据中毒的请求 [4]: 屏蔽 LN 消息 [3]: 屏蔽刷新请求 (flush request) 过滤；

			<p>[2]: 对应不期望的完成 TLP, 屏蔽使能 DLLP 异常中止;</p> <p>[1]: 屏蔽丢弃 Type1 类型的厂商自定义消息的处理;</p> <p>[0]: 屏蔽丢弃 Type0 类型的厂商自定义消息并将其当作 UR 的处理;</p> <p>其它: 保留。</p>
--	--	--	---

5.2.5.10 传输 P 类事务流量控制信用状态寄存器

偏移: 0xe600。

表 5-139: 传输 P 类事务流量控制信用状态寄存器描述

名称	范围	类型	描述
PHdrFCCrdt	[19:12]	ROS,0x00	链路另一端接收器所通告的 P 类事务包头信用。初始值为 0, 在流量控制初始化后根据接收缓冲的规模进行设置。
PDataFCCrdt	[11:0]	ROS,0x000	链路另一端接收器所通告的 P 类事务数据信用。初始值为 0, 在流量控制初始化后根据接收缓冲的规模进行设置。
—	其它	—	保留。

5.2.5.11 传输 NP 类事务流量控制信用状态寄存器

偏移: 0xe680。

表 5-140: 传输 NP 类事务流量控制信用状态寄存器描述

名称	范围	类型	描述
NPHdrFCCrdt	[19:12]	ROS,0x00	链路另一端接收器所通告的 NP 类事务包头信用。初始值为 0, 在流量控制初始化后根据接收缓冲的规模进行设置。
NPDataFCCrdt	[11:0]	ROS,0x000	链路另一端接收器所通告的 NP 类事务数据信用。初始值为 0, 在流量控制初始化后根据接收缓冲的规模进行设置。
—	其它	—	保留。

5.2.5.12 传输完成事务流量控制信用状态寄存器

偏移: 0xe700。

表 5-141: 传输完成事务流量控制信用状态寄存器描述

名称	范围	类型	描述
CplHdrFCCrdt	[19:12]	ROS,0x00	链路另一端接收器所通告的完成类事务包头信用。初始值为 0，在流量控制初始化后根据接收缓冲的规模进行设置。
CplDataFCCrdt	[11:0]	ROS,0x000	链路另一端接收器所通告的完成类事务数据信用。初始值为 0，在流量控制初始化后根据接收缓冲的规模进行设置。
—	其它	—	保留。

5.2.5.13 队列状态寄存器

偏移：0xe780。

表 5-142：队列状态寄存器描述

名称	范围	类型	描述
TimrModFlowCtlEn	[31]	RW,0	流量控制延时计数器覆盖使能，当这位为 1 时，流量控制延时计数器覆盖值会覆盖掉原始自动计算的值
TimerModFlowCtl	[28:16]	RW,0	流量控制延时计数器覆盖值
RcvQNotEmpty	[2]	ROS,0	接收队列非空。
TransRBufNotEmpty	[1]	ROS,0	传输重发缓冲非空。
RcvFCCrdtNotRtrn	[0]	ROS,0	事务层数据包信用尚未返回。
—	其它	—	保留。

5.2.5.14 Gen2 控制寄存器

偏移：0x10180。

该部分寄存器在 Gen2 和 Gen3 速率下都存在

表 5-143：Gen2 模式寄存器描述

名称	范围	类型	描述
Gen1EIInfer	[21]	RW,0	在 GEN1 速率下的电气空闲推断模式。当该位为 1 时，在 GEN1 的 Recovery.Speed 或者 Loopback.Active 状态下通过判断 RxElecIdle 是否为 1，而不是判断 RxValid 为 0。
SelDeemphasis	[20]	RW,0	设置去加重等级（De-emphasis Level）。编码含义为：0: -6 dB; 1: -3.5 dB。
TxCompRcv	[19]	RW,0	配置发送器兼容性测试接收位。设置为“1”时，通知 LTSSM 发送 TS 序列。

TxSwing	[18]	RW,0	配置 PMA 发送电压摆幅。编码含义为： 0: 低摆幅； 1: 全摆幅。
DrctSpeedChange	[17]	RW,0	指导进行速率改变。指示 LTSSM 是否要改变速率。
AutoLaneFlipEn	[16]	RW,0	使能自动连接通道
PreDetLane	[15:13]	RW,0	预先设置当自动连接通道的时候，哪个物理通道连接到逻辑通道 0： 000: 逻辑通道 0 连接到物理通道 0 001: 逻辑通道 0 连接到物理通道 1 010: 逻辑通道 0 连接到物理通道 3 011: 逻辑通道 0 连接到物理通道 7 100: 逻辑通道 0 连接到物理通道 15
NumOfLanes	[12:8]	RW,8	通道数
FTSNum	[7:0]	RW,180	设置 FTS (Fast Training Sequence, 快速训练序列) 的数目。在 Gen2 或 Gen3 链路训练时，RC 需要通告其 FTS 的数目，其作用是把链路恢复同步的能力告知链路的另一方。
—	其它	—	保留。

5.2.5.15 Gen3 控制寄存器

偏移: 0x11200

表 5-144: Gen3 控制寄存器描述

名称	范围	类型	描述
InvReqEvalDiff TimeDisable	[23]	RW,0	取消 Eq InvalidRequest 和 RxEqEval 不能同时出现的要求
DCBalnceDisable	[18]	RW,0	取消 DC 平衡的功能
DllpXmtDelay Disable	[17]	RW,0	取消在链路均衡前延迟链路层报文的传输
EqDisable	[16]	RW,0	取消链路均衡
RxEqRgrdlessTS	[15]	RW,0	0: mac_phy_rxeqeval 在收到一个新的 preset 请求过后 1us 并且从对端链路接收到 2 个 TS1 序列过后开始有效 1: mac_phy_rxeqeval 在收到一个新的 preset 请求 500ns 过

			后直接有效，不管 TS 序列是否被接收
RxEqPH01En	[12]	RW,0	上行端在均衡第 0 阶段保持 10ms，下行端在均衡第 1 阶段保持 10ms 以提供足够时间完成均衡
EqReDoDisable	[11]	RW,0	取消自动重新开始均衡
EqEIEOSCnt	[10]	RW,0	取消在均衡过程中请求复位 EIEOS 序列计数器的功能
EqPH23	[9]	RW,0	只在下行端有效，取消均衡的第二和第三个阶段
DisableScrambler	[8]	RW,0	在 Gen3 速率下取消加扰
ZRX-DC_ NonComply	[0]	RW,0	在 Gen3 速率下接收器不遵守 Gen1 速率下的阻抗参数 (40~60Ohms)
—	其它	—	保留。

5.2.5.16 均衡控制寄存器

偏移：0x11500

表 5-145：均衡控制寄存器描述

名称	范围	类型	描述
ReqSendConsec EIEOS	[26]	RW,0	请求 RC 在链路状态的 Recovery.RcvrLock 状态发送连续的 EIEOS 序列，直到 presets to coefficients mapping 完成
EqFOMInitialEval	[24]	RW,0	
EqPsetReqVec	[23:8]	RW,0	Preset 请求向量，编码含义： 0000000000000000: 在 EQ Master 阶段不请求和评估 preset; 000000xxxxxxx1: 在 EQ Master 阶段请求和评估 preset0; 000000xxxxxxx1x: 在 EQ Master 阶段请求和评估 preset1; 000000xxxxxxx1xx: 在 EQ Master 阶段请求和评估 preset2; 000000xxxxxxx1xxx: 在 EQ Master 阶段请求和评估 preset3; 000000xxxxx1xxxx: 在 EQ Master 阶段请求和评估 preset4; 000000xxxx1xxxxx: 在 EQ Master 阶段请求和评估 preset5; 000000xxx1xxxxxx: 在 EQ Master 阶段请求和评估 preset6; 000000xx1xxxxxxx: 在 EQ Master 阶段请求和评估 preset7; 000000x1xxxxxxx: 在 EQ Master 阶段请求和评估 preset8; 0000001xxxxxxx: 在 EQ Master 阶段请求和评估 preset9; 其余编码保留

EqEval2MsDisable	[5]	RW,0	关掉 EQ 第 2 和第 3 阶段的 2ms 超时退出功能。给 PHY 足够的时间完成 EQ。
EqPh23ExitMode	[4]	RW,0	当 24ms 超时未完成 EQ 第 3 阶段时，LTSSM 状态机退出 EQ3 状态的下一个状态： 0: Recovery.Speed 1: Recovery.Equalization.RcvrLock
EqFeedbackMode	[3:0]	RW,0	反馈模式： 0000: 直接反馈 (Direction Change) 0001: Figure Of Merit
—	其它	—	保留。

5.2.5.17 顺序规则寄存器

偏移: 0x11680

表 5-146: 顺序规则寄存器描述

名称	范围	类型	描述
CplPassP	[15:8]	RW,0	决定响应是否能够超越 P 类事务
NPPassP	[7:0]	RW,0	决定 NP 类事务是否可以超越 P 类事务
—	其它	—	保留。

5.2.5.18 只读域写使能寄存器

偏移: 0x11780

表 5-147: 只读域写使能寄存器描述

名称	范围	类型	描述
DbiRoWrEn	[0]	RW,1	是否可以通过 DBI 接口写 RC 中的只读域
—	其它	—	保留。

5.2.5.19 LUT 删除寄存器

偏移: 0x11900

表 5-148: LUT 删除寄存器描述

名称	范围	类型	描述
DeleteEn	[31]	RW1C	对这个寄存器写 1 操作会删除掉 LookUpId 域中所对应 LUT 的条目。该域自动清 0，任何时候读这个寄存器都返回 0 值
LookUpId	[30:0]	RW,0	LUT 中需要删除的条目号
—	其它	—	保留。

附录 A PCI-E 术语表

表 A-0-1: PCI-E 标准术语的缩写、英文全称和中文含义对照表

缩写	英文	含义
PCI-E	Peripheral Component Interconnection Express	设备接口协议增强版。
DM	PCI Express Dual Mode core	双模式 (RC/EP) 核心。
RC	PCI Express Root Complex core	根设备核心。
EP	PCI Express End Point core	端设备核心。
PIPE	PHY Interface for the PCI Express Architecture	PCI-E 结构中的 PHY 接口。
NW	double words	32 位。
DW	Data width	数据宽度, 可为: 32、64、128 位。
NF	Number of functions	功能设备数。
TLP	Transaction Layer Packet	事务层数据包。
DLLP	Data Link Layer Packet	数据链路层数据包。
VC	Virtual Channel	虚通道。
BAR	Base Address Register	基址寄存器。
—	Upstream	上游设备, 在系统中指 CPU。
—	Downstream	下游设备, 在系统中指 EP 设备。
MSI	Message Signalled Interrupt	中断消息。
TC	Traffic Classes	业务级别。
—	Sticky	粘着属性, 即指仅有上电复位才能复位该寄存器。
—	Non_ Sticky	非粘着属性, 即指所有复位都可以复位该寄存器。
PME	Power Management Event	电源管理事件。
—	Payload	请求携带的数据。
DBI	External Data Bus Interface	数据总线接口, 即用来配置读写 RC 内部寄存器的接口。
LTSSM	Link Training and Status State Machine	数据链路状态机。
CRS	Completion Request Retry Status	对 EP 的配置请求未成功, 需要重发。
UR	Unsupported Request	请求是 EP 不支持的类型。
CA	Completion Abort	EP 此时故障无法处理该笔事务。

—	Non-Posted	非邮寄事务，即需要返回响应的事务。
—	Posted	邮寄事务，即不需要返回响应的事务。
ASPM	Active State Power Management	有效状态电源管理策略。
FC	Flow Control	流量控制。

附录 B PCI-E 链路状态编码

申威 421 处理器的 PCI-E 链路状态机遵从 PCI-E2.0 规范, 并自定义了部分状态以方便内部处理。LTSSM 的状态编码及含义具体如下:

表 B-1: PCI-E 标准术语的缩写、英文全称和中文含义对照表

编码	PCI-E 2.0 规范的对应状态	含义
0x00	Detect.Quiet	检测状态进入。
0x01	Detect.Active	检测状态工作期间。
0x02	Polling.Active	查询状态工作期间。
0x03	Polling.Compliance	查询状态兼容性测试。
0x04	Polling.Configuration	查询状态配置。
0x05	PreDetect.Quiet	预检测状态。
0x06	Detect.Wait	检测状态等待。
0x07	Configuration.Linkwidth.Start	配置状态开始进行链路宽度协商。
0x08	Configuration.Linkwidth.Accept	配置状态完成链路宽度协商。
0x09	Configuration.Lanenum.Wait	配置状态进行通道编号协商。
0x0A	Configuration.Lanenum.Accept	配置状态完成通道编号协商。
0x0B	Configuration.Complete	配置状态完成。
0x0C	Configuration.Idle	配置状态保持。
0x0D	Recovery.RcvrLock	恢复状态进行接收器同步 (符号锁定、比特锁定)。
0x0E	Recovery.Speed	恢复状态进行链路速率协商。
0x0F	Recovery.RcvrCfg	恢复状态进行接收器配置。
0x10	Recovery.Idle	恢复状态保持。
0x20	Recovery.Equalization0	恢复状态的均衡化第 0 阶段
0x21	Recovery.Equalization1	恢复状态的均衡化第 1 阶段
0x22	Recovery.Equalization2	恢复状态的均衡化第 2 阶段
0x23	Recovery.Equalization3	恢复状态的均衡化第 3 阶段
0x11	L0	链路训练成功, 正常工作状态。
0x12	L0s	L0s 低功耗状态退出。
0x13	L123_Send_EIDLE	L1/L2/L3 低功耗状态下发送电气空闲 (Electrical Idle)。
0x14	L1.Idle	L1 低功耗保持状态。
0x15	L2.Idle	L2 低功耗保持状态。

0x16	L2.TransmitWake	L2 低功耗唤醒状态。
0x17	Disabled.Entry	禁用状态进入。
0x18	Disabled.Idle	禁用保持状态。
0x19	Disabled	禁用状态。此时链路已关闭 DC 共模电压。
0x1A	Lookback.Entry	环回测试状态进入。
0x1B	Loopback.Active	环回测试状态工作。
0x1C	Loopback.Exit	环回测试状态退出。
0x1D	Loopback.ExitTimeout	环回测试退出超时。
0x1E	HotReset.Entry	热复位状态进入。
0x1F	HotReset	热复位状态。