



SUNWAY 申威

申威 411 处理器 数据手册

2017 年 1 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。如因文档使用不当造成的直接或间接损失，本公司不承担任何责任。

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 411 处理器有两种封装版本，LGA1156 封装和 CBGA1144 封装。

《申威 411 处理器数据手册》主要描述了申威 411 处理器的封装、引脚定义、对外接口、配置和复位与初始化流程，详细说明处理器的维护命令，并介绍处理器的电气特性等内容。

文档修订

文档更新记录	文档名	申威 411 处理器数据手册
	版本号	V1.0
	创建人	研发部
	创建日期	2017-1-8

版本更新

版本号	更新内容	更新日期
V1.0	初稿	2017-1-8

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	核心结构.....	1
1.2	芯片结构.....	2
1.3	硬件接口.....	2
2	封装及引脚说明	4
2.1	LGA 封装结构及尺寸.....	4
2.2	LGA 封装引脚分类及列表.....	6
2.3	CBGA 封装结构及尺寸.....	18
2.4	CBGA 封装引脚分类及列表.....	20
3	电地及特殊信号接口	34
3.1	电地引脚列表.....	34
3.1.1	数字电源地说明.....	35
3.1.2	模拟电源地说明.....	35
3.2	温度传感器引脚列表.....	36
3.3	空引脚列表.....	37
4	时钟接口	38
4.1	引脚列表.....	38
4.2	工作模式.....	38
5	存储器接口	39
5.1	概述.....	39
5.2	引脚列表.....	39
5.3	存储器接口配置.....	41
5.3.1	连接方式.....	41
5.3.2	1T/2T 模式.....	41
5.3.3	地址镜像.....	42
5.3.4	颗粒/内存类型.....	43
5.4	信号连接.....	43
5.4.1	片选信号.....	43
5.4.2	选通信号.....	44
5.4.3	ECC 校验码.....	44
5.4.4	连接存储器芯片.....	45
5.4.5	连接存储器条.....	45
5.5	初始化.....	46
5.6	测试.....	46

6	PCI-E 接口	47
6.1	概述	47
6.2	引脚列表	47
6.3	信号连接	47
6.4	初始化	48
6.5	测试	48
7	维护、中断和配置接口	49
7.1	维护接口	49
7.1.1	概述	49
7.1.2	引脚列表	49
7.1.3	维护功能	50
7.2	配置接口	53
7.2.1	引脚列表	53
7.2.2	初始化配置	54
7.2.3	时钟配置	54
7.3	中断接口	58
8	测试接口	60
8.1	概述	60
8.2	引脚列表	60
8.3	TAP 控制器	61
8.4	测试功能	61
8.5	维护功能	64
9	复位与初始化	67
9.1	复位	67
9.1.1	复位类型	67
9.1.2	复位状态机	67
9.2	配置	69
9.2.1	引脚配置	69
9.2.2	I/O 寄存器配置	70
9.3	初始化	71
9.3.1	加关电顺序	71
9.3.2	初始化流程	72
9.3.3	初始化程序	73
10	电气特性	74
10.1	工作条件	74
10.2	直流参数	74

10.2.1	差分时钟输入信号.....	74
10.2.2	存储器接口信号.....	75
10.2.3	1.8V LVCMOS 信号.....	75
10.3	交流参数.....	76
10.3.1	差分时钟输入信号.....	76
10.3.2	存储器接口信号.....	76
10.3.3	维护接口信号.....	78
10.3.4	测试接口信号.....	80
10.3.5	PCI-E 接口.....	81
10.4	功耗.....	82
10.4.1	封装热特性.....	82
10.4.2	热设计功耗.....	82
10.4.3	功耗管理.....	82
附录 A	维护接口的命令与响应.....	84
A.1	维护命令.....	84
A.1.1	维护命令包格式.....	84
A.1.2	维护命令编码.....	84
A.1.3	维护命令目标编码.....	85
A.1.4	维护命令详细说明.....	86
A.2	维护响应.....	93
A.2.1	维护响应包格式.....	93
A.2.2	维护响应编码.....	94
A.2.3	有数据维护响应.....	94
A.2.4	无数据维护响应.....	94
附录 B	边界扫描链说明.....	96
B.1	GPIO BSR 链说明.....	96
B.2	PCI-E BSR 链说明.....	97
B.3	DDR3 BSR 链说明.....	98

1 概述

申威 411 为 64 位字长的高性能通用处理器，采用多核架构和片上系统（SoC: System on Chip）技术，单芯片集成了 4 个对称的新一代申威 Core3 核心，还集成了 6MB 的共享三级 Cache、两路 DDR3 存储控制器接口、两路 PCI-E 接口、一路维护接口和一路测试接口。

申威 411 处理器提供两种封装，即有机基板的 LGA1156 封装和陶瓷基板的 CBGA1144 封装。

1.1 核心结构

申威 411 处理器采用新一代申威 Core3 核心，基本结构如图 1-1 所示：

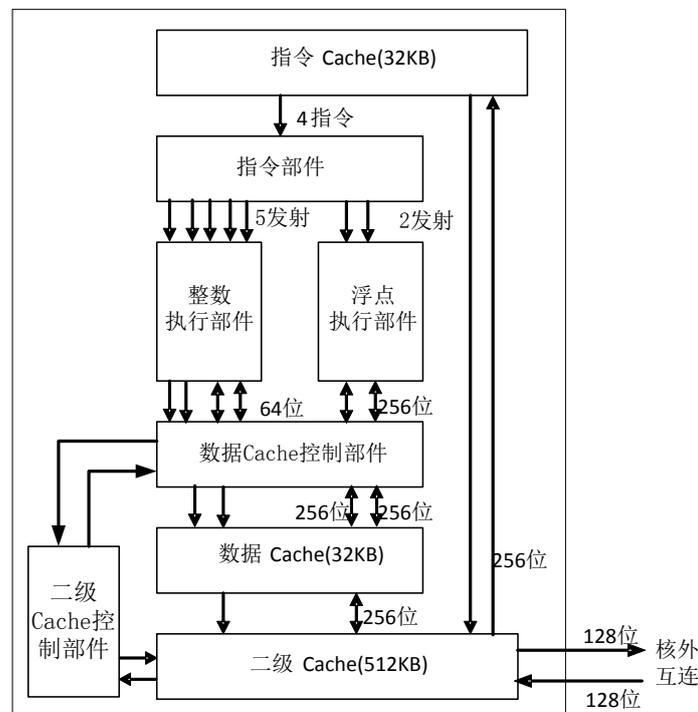


图 1-1: 申威 411 处理器核心结构图

Core3 由指令部件、整数执行部件、浮点执行部件、数据 Cache 控制部件、二级 Cache 控制部件以及一级指令 Cache、一级数据 Cache 和二级 Cache 组成。其技术特征如下：

- 1) 核心为采用并行发射、乱序发射、乱序执行和推测执行技术的 4 译码 7 发射超标量结构；
- 2) 采用短向量加速计算技术提高整数和浮点运算性能，支持浮点双 256 位 SIMD 流水线、整数单 256 位 SIMD 流水线，每个时钟周期可产生 11 个整数运算结果或 16 个浮点运算结果；
- 3) 一级指令 Cache 容量为 32KB，采用四路组相联结构，虚地址访问方式，Cache 行大小为 128 字节，采用可容错的偶校验；
- 4) 一级数据 Cache 容量为 32KB，采用四路组相联结构，物理地址访问方式，Cache 行大小为

128 字节，采用可纠错的 ECC 校验；

- 5) 二级 Cache 容量为 512KB，采用八路组相联结构，物理地址访问方式，Cache 行大小为 128 字节，为指令和数据混合 Cache，采用可纠错的 ECC 校验；
- 6) 一级数据 Cache 与二级 Cache 为严格的包含关系，一级指令 Cache 与二级 Cache 为既不包含，也不互斥关系，硬件自动支持指令与数据的 Cache 一致性。

1.2 芯片结构

申威 411 处理器芯片包含四个 Core3 核心、6MB 的三级共享 Cache 和两路 DDR3 存储器接口、两路 PCI-E 接口以及维护、测试接口，其结构图如下所示：

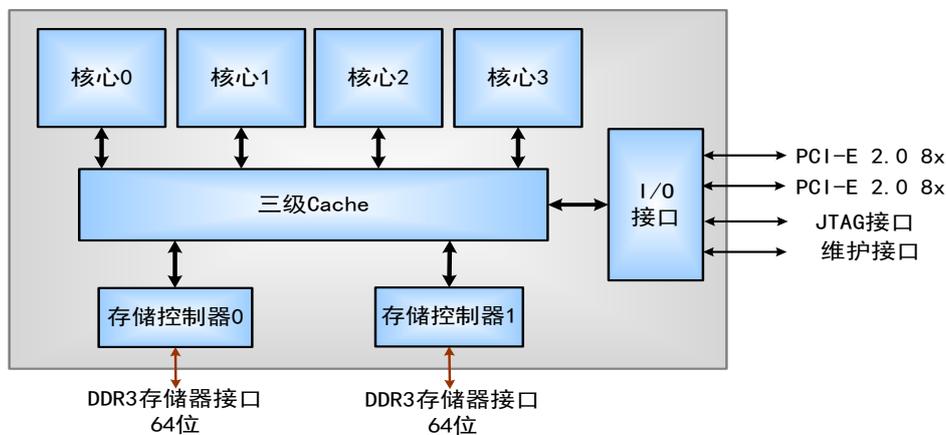


图 1-2: 申威 411 处理器芯片结构图

其特点有：

- 1) 单芯片集成 4 个核心，最高工作频率 1.6GHz；
- 2) 采用对称式共享存储（SMP）结构和 SoC 技术；
- 3) 集成 6MB 的三级共享 Cache；
- 4) 集成两路 DDR3 存储器控制器和 DDR3 存储器接口；
- 5) 集成两路 PCI-E 2.0 标准接口，分别采用 8x 链路，支持 I/O 虚拟化功能；
- 6) 提供两种倒装焊接（Flip-chip）的封装产品，分别是基于有机基板的 LGA 封装和基于陶瓷基板的 CBGA 封装，其中 LGA 封装与申威 410 引脚兼容。

1.3 硬件接口

申威 411 处理器的外部硬件接口由六部分组成，具体如下：

- 1) 时钟接口：为处理器提供时钟源，通过配置产生处理器内部各种工作时钟；

- 2) 存储器接口：两路各 64 位，支持 ECC 校验，最高数据传输率为 1600Mbps；
- 3) PCI-E 接口：符合 PCI-E 2.0 规范，两路接口，每路接口支持 8 条链路，链路速率为 5Gbps；
- 4) 维护、中断和配置接口：支持对处理器的维护调试，支持对处理器的复位和外部中断，支持对处理器的配置；
- 5) 测试接口：支持兼容 IEEE1149.1 标准的 JTAG 测试接口；
- 6) 电地和特殊信号接口：为处理器提供内核电源和 I/O 电源，
- 7) 为处理器提供模拟电源等特殊信号。

2 封装及引脚说明

申威 411 处理器提供两种封装，即有机基板的 LGA1156 封装和陶瓷基板的 CBGA1144 封装。

2.1 LGA 封装结构及尺寸

采用 LGA (Land Grid Array) 封装的申威 411 处理器，封装引脚数量为 1156。其外观如图 2-1、图 2-2 和图 2-3 所示，图 2-1 为顶视图，图 2-2 为底视图，图 2-3 为侧视图。

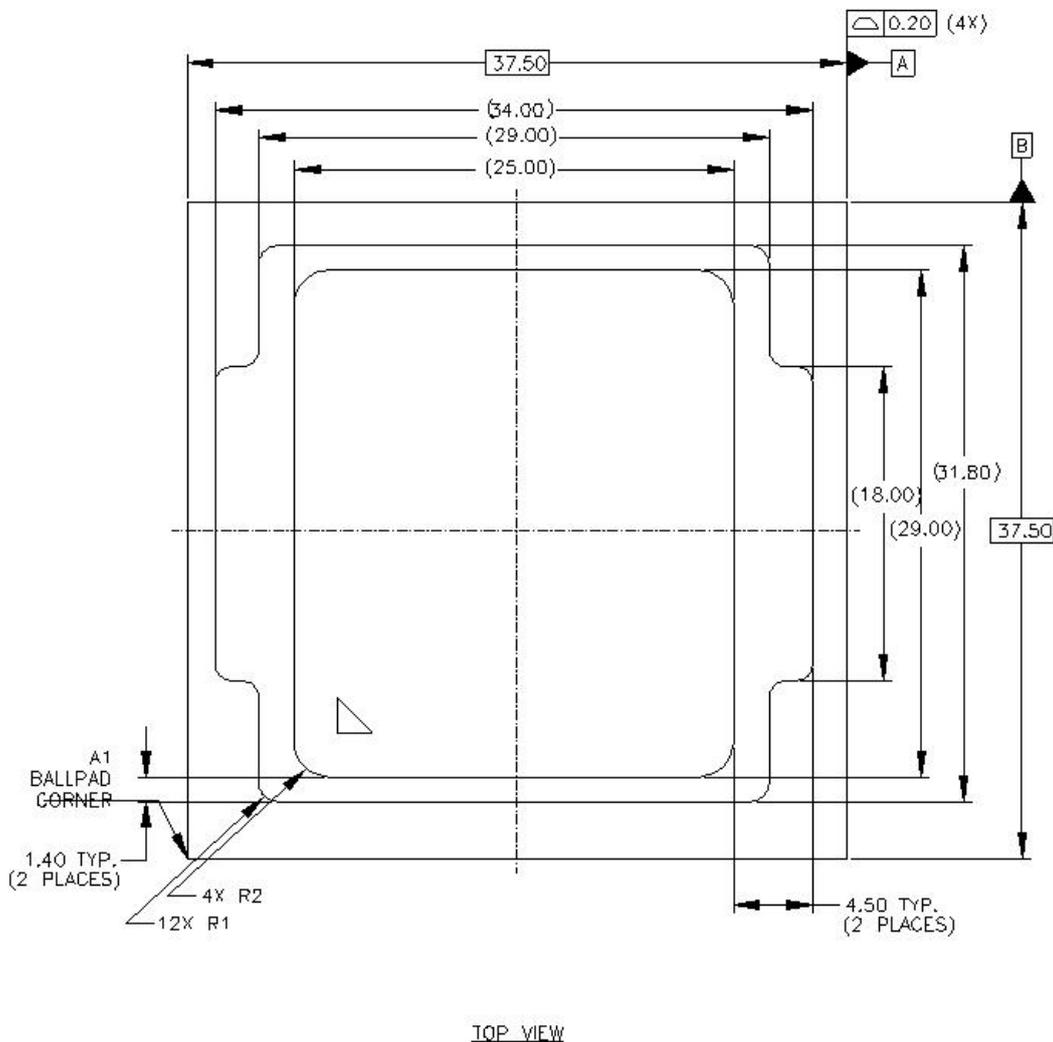


图 2-1: LGA 封装顶视图

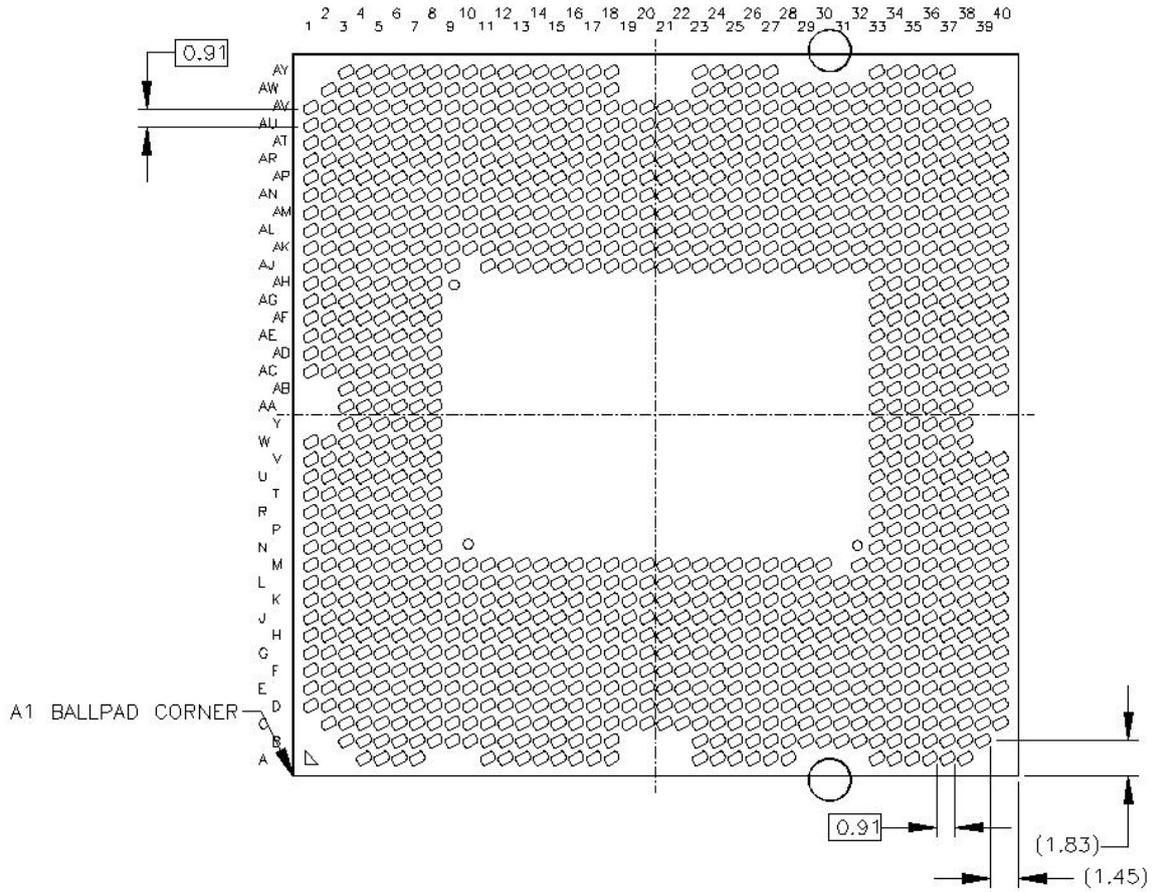


图 2-2: LGA 封装底视图

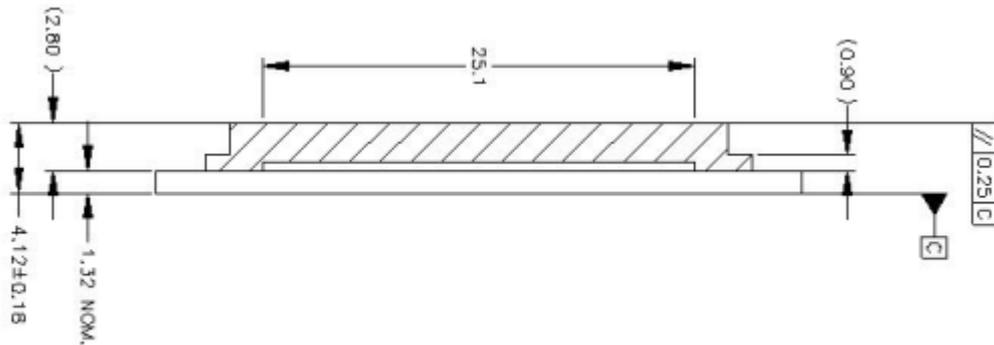


图 2-3: LGA 封装侧视图

LGA 封装的封装特征如下:

- 1) 封装尺寸 (长×宽×高): 37.5mm×37.5mm×4.12mm (高度误差: ±0.18mm);
- 2) 引脚数量: 1156;
- 3) 引脚为 40×40 阵列分布, 引脚编号参见图 2-2 (图中垂直方向的字母和水平方向的数字组成每个引脚的位置)。

2.2 LGA 封装引脚分类及列表

LGA 封装的引脚总数为 1156 根，表 2-1 为 LGA 封装引脚的分类与数量。

表 2-1: LGA 封装引脚分类列表

类型	数量
信号引脚	363
电地引脚	783
备份引脚（保持悬空，不连接）	6
温度传感器引脚	4
合计	1156

表 2-2 为申威 411 处理器信号引脚的分类与数量。

表 2-2: LGA 封装信号引脚分类列表

名称	数量	输入引脚	输出引脚	双向引脚	信号电平类型
存储器接口	257	0	69	188	SSTL (1.5V 或 1.35V)
PCI-E 接口	70	36	32	2	除 PCI[1:0]_RESREF(双向引脚)连接参考电阻外, 其他信号是 LVDS;
维护接口	17	11	6	0	LVC MOS (1.8V)
时钟接口	12	12	0	0	一对差分输入为 LVDS, 其它为 LVC MOS (1.8V)
测试接口	7	6	1	0	LVC MOS (1.8V)
合计	363	65	108	190	

注：信号引脚的信号电平主要是 SSTL (1.5V 或 1.35V)、LVDS (1.8V) 和 LVC MOS (1.8V) 三大类，其中 SSTL 电平的 I/O 电源为 1.5V 或 1.35V，其它为 1.8V。

LGA 封装的引脚位置与引脚名称对应关系如表 2-3 所示。其中位置是根据图 2-2 所指示的字母和数字组成，引脚名称尾缀为“_L”的信号为低电平有效的信号，尾缀为“_H”的信号为高电平有效的信号；引脚名称相同且尾缀分别为“_P”和“_N”的一对信号为配对的差分信号，其中尾缀为“_P”信号为“正”信号（高电平有效）。

表 2-3: LGA 封装引脚位置与名称对应表

位置	引脚	位置	引脚	位置	引脚
A38	MM1_DQ33_H	L29	VSS	AK4	VDD
A37	VSS	L28	VDD	AK3	VDD
A36	MM1_DQ39_H	L27	VSS	AK2	VDD
A35	MM1_DQS4_N	L26	VDDQ15	AK1	VDD

A34	VDDQ15	L25	VSS	AL40	VDDQ15
A33	MM1_ODT0_H	L24	VDD	AL39	MM0_DQ7_H
A28	VDDQ15	L23	VSS	AL38	MM0_DQ14_H
A27	MM1_CK1_N	L22	VDD	AL37	VDDQ15
A26	MM1_CK1_P	L21	VSS	AL36	MM0_DQ13_H
A25	VSS	L20	VDDQ15	AL35	VSS
A24	MM1_DQ42_H	L19	VSS	AL34	VSS
A23	MM1_DQ41_H	L18	VDD	AL33	VSS
A18	MM1_DQ51_H	L17	VSS	AL32	VSS
A17	MM1_DQS6_N	L16	VDD	AL31	VDD
A16	VDDQ15	L15	VSS	AL30	VSS
A15	MM1_ECC7_H	L14	VDDQ15	AL29	VDD
A14	MM1_ECC6_H	L13	VSS	AL28	VSS
A13	VSS	L12	VDD	AL27	VDD
A12	MM1_ECC1_H	L11	VSS	AL26	VSS
A11	MM1_DQS8_N	L10	VDD	AL25	VDD
A7	SEL_TAP1_L	L9	VSS	AL24	VSS
A6	CFG_CORE0_H	L8	PCI1_VTT	AL23	VDD
A5	CFG_CORE3_H	L7	VSS	AL22	VSS
A4	CFG_XBX1_H	L6	PCI1_VDDQA	AL21	VDD
B39	VSS	L5	PCI1_TX2_P	AL20	VSS
B38	MM1_DQ35_H	L4	PCI1_TX2_N	AL19	VDD
B37	MM1_DQ32_H	L3	VSS	AL18	VSS
B36	VDDQ15	L2	PCI1_RX2_P	AL17	VDD
B35	MM1_DQS4_P	L1	PCI1_RX2_N	AL16	VSS
B34	MM1_DQ38_H	M40	MM1_DQS2_N	AL15	VSS
B33	VSS	M39	MM1_DQS2_P	AL14	VDD
B32	MM1_BA0_H	M38	VSS	AL13	VDD
B31	MM1_CK0_P	M37	MM0_DQ37_H	AL12	VSS
B30	VDDQ15	M36	MM0_DQ39_H	AL11	VSS
B29	MM1_A1_H	M35	VSS	AL10	VDD
B28	MM1_A7_H	M34	VSS	AL9	VDD
B27	VSS	M33	VSS	AL8	VDD

B26	MM1_A11_H	M32	VDDQ15	AL7	VDD
B25	MM1_A15_H	M30	VDD	AL6	VSS
B24	VDDQ15	M29	VSS	AL5	VSS
B23	MM1_DQ43_H	M28	VDD	AL4	VDD
B18	VDDQ15	M27	VSS	AL3	VDD
B17	MM1_DQS6_P	M26	VDDQ15	AL2	VSS
B16	MM1_DQ50_H	M25	VSS	AL1	VSS
B15	VSS	M24	VDD	AM40	MM0_DQ5_H
B14	MM1_ECC5_H	M23	VSS	AM39	VSS
B13	MM1_ECC4_H	M22	VDD	AM38	VSS
B12	VDDQ15	M21	VSS	AM37	VSS
B11	MM1_DQS8_P	M20	VDDQ15	AM36	VSS
B10	MM1_ECC2_H	M19	VSS	AM35	PCI0_VTT
B9	CLK_VDDQ18	M18	VDD	AM34	PCI0_VTT
B8	CFG_MM0_H	M17	VSS	AM33	PCI0_VTT
B7	CLK_VDDA0	M16	VDD	AM32	PCI0_VTT
B6	CLK_VDDA1	M15	VSS	AM31	PCI0_VTT
B5	VSS	M14	VDDQ15	AM30	PCI0_VTT
B4	CFG_XBX2_H	M13	VSS	AM29	PCI0_VTT
B3	CFG_XBX0_H	M12	VDD	AM28	PCI0_VTT
C40	MM1_DQ36_H	M11	VSS	AM27	VSS
C39	MM1_DQ34_H	M10	VDD	AM26	VSS
C38	VDDQ15	M9	VSS	AM25	VSS
C37	MM1_WE_L	M8	PCI1_VTT	AM24	VSS
C36	MM1_DQ37_H	M7	VSS	AM23	VSS
C35	VSS	M6	PCI1_VDDQA	AM22	VSS
C34	MM1_CS0_L	M5	PCI1_TX3_P	AM21	VSS
C33	MM1_ODT1_H	M4	PCI1_TX3_N	AM20	VSS
C32	VDDQ15	M3	VSS	AM19	VSS
C31	MM1_CK0_N	M2	PCI1_RX3_P	AM18	VSS
C30	MM1_A2_H	M1	PCI1_RX3_N	AM17	VSS
C29	VSS	N40	VDDQ15	AM16	VSS
C28	MM1_A6_H	N39	MM1_DQ16_H	AM15	VSS

C27	MM1_A8_H	N38	MM1_PVT1	AM14	VDD
C26	VDDQ15	N37	VDDQ15	AM13	VDD
C25	MM1_A12_H	N36	MM0_DQ36_H	AM12	VSS
C24	MM1_A14_H	N35	MM0_DQ38_H	AM11	VSS
C23	VSS	N34	VDD	AM10	VDD
C22	MM1_DQ47_H	N33	VDD	AM9	VDD
C21	MM1_DQS5_N	N8	PCI1_VTT	AM8	VSS
C20	VDDQ15	N7	VSS	AM7	VDD
C19	MM1_DQ53_H	N6	PCI1_CLK_P	AM6	VDD
C18	MM1_DQ52_H	N5	VSS	AM5	VSS
C17	VSS	N4	VSS	AM4	VSS
C16	MM1_DQ48_H	N3	PCI1_REFRES	AM3	VDD
C15	MM1_DQ60_H	N2	VSS	AM2	VDD
C14	VDDQ15	N1	VSS	AM1	VSS
C13	MM1_DQ57_H	P40	MM1_DQ17_H	AN40	MM0_DQS0_N
C12	MM1_ECC3_H	P39	VSS	AN39	MM0_DQS0_P
C11	VSS	P38	MM1_DQ19_H	AN38	VSS
C10	MM1_ECC0_H	P37	MM0_DQ33_H	AN37	VSS
C9	CLK_VDDQ18	P36	VSS	AN36	VSS
C8	CFG_MM1_H	P35	MM0_DQ35_H	AN35	VSS
C7	CLK_VSSA0	P34	VSS	AN34	VSS
C6	CLK_VSSA1	P33	VSS	AN33	VSS
C5	CFG_CORE1_H	P8	PCI1_VTT	AN32	VSS
C4	CLK_VDDQ18	P7	VSS	AN31	VSS
C3	VSS	P6	PCI1_CLK_N	AN30	VSS
C2	NC	P5	VSS	AN29	VSS
D40	VDDQ15	P4	VSS	AN28	VSS
D39	MM1_DQ26_H	P3	VSS	AN27	VSS
D38	MM1_RESET_L	P2	VSS	AN26	VSS
D37	VSS	P1	VSS	AN25	VSS
D36	MM1_CAS_L	R40	MM1_DQ20_H	AN24	MT_VDDQ18
D35	MM1_RAS_L	R39	MM1_DQ18_H	AN23	VSS
D34	VDDQ15	R38	VDDQ15	AN22	VSS

D33	MM1_CKE0_H	R37	MM0_DQS4_N	AN21	MT_VDDQ18
D32	MM1_BA1_H	R36	MM0_DQS4_P	AN20	VSS
D31	VSS	R35	VDDQ15	AN19	VSS
D30	MM1_BA2_H	R34	VDDQ15	AN18	MT_VDDQ18
D29	MM1_A0_H	R33	VDDQ15	AN17	VSS
D28	VDDQ15	R8	PCI1_VTT	AN16	VSS
D27	MM1_A5_H	R7	VSS	AN15	MT_VDDQ18
D26	MM1_A9_H	R6	PCI1_VDDA	AN14	VDD
D25	VSS	R5	PCI1_TX4_P	AN13	VDD
D24	MM1_A13_H	R4	PCI1_TX4_N	AN12	VSS
D23	MM1_DQ44_H	R3	VSS	AN11	VSS
D22	VDDQ15	R2	PCI1_RX4_P	AN10	VDD
D21	MM1_DQS5_P	R1	PCI1_RX4_N	AN9	VDD
D20	MM1_DQ45_H	T40	VSS	AN8	VDD
D19	VSS	T39	MM1_DQ10_H	AN7	VSS
D18	MM1_DQ54_H	T38	MM1_VSSQA0	AN6	VDD
D17	MM1_DQ49_H	T37	VSS	AN5	VDD
D16	VDDQ15	T36	MM0_DQ32_H	AN4	VSS
D15	MM.DTO1	T35	MM0_PVT0	AN3	VSS
D14	MM1_PVT2	T34	VSS	AN2	VDD
D13	VSS	T33	VSS	AN1	VDD
D12	MM1_DQ63_H	T8	PCI1_VTT	AP40	VSS
D11	MM1_DQS7_N	T7	VSS	AP39	MM0_DQ6_H
D10	VDDQ15	T6	PCI1_VDDA	AP38	PCIO_VDDQA
D9	CLK_VDDQ18	T5	PCI1_TX5_P	AP37	PCIO_VDDQA
D8	CLK_VDDQ18	T4	PCI1_TX5_N	AP36	PCIO_VDDQA
D7	VSS	T3	VSS	AP35	PCIO_VDDQA
D6	CLK_VDDQ18	T2	PCI1_RX5_P	AP34	PCIO_VDDQA
D5	CLK_VDDA2	T1	PCI1_RX5_N	AP33	PCIO_VDDQA
D4	CFG_CORE2_H	U40	MM1_DQ8_H	AP32	PCIO_CLK_P
D3	LVDS_VSSQA	U39	VDDQ15	AP31	PCIO_CLK_N
D2	VSS	U38	MM1_VDDQA0	AP30	PCIO_VDDA
D1	RCLK_N	U37	MM0_DQ26_H	AP29	PCIO_VDDA

E40	MM1_DQ25_H	U36	VDDQ15	AP28	PCIO_VDDA
E39	VSS	U35	MM0_DQ34_H	AP27	PCIO_VDDA
E38	MM1_DQ24_H	U34	VDD	AP26	PCIO_VDDA
E37	MM1_CS1_L	U33	VDD	AP25	PCIO_VDDA
E36	VDDQ15	U8	PCI1_VTT	AP24	MT_VDDQ18
E35	MM1_VDDQA1	U7	VSS	AP23	VSS
E34	MM1_VSSQA1	U6	PCI1_VDDA	AP22	VSS
E33	VSS	U5	VSS	AP21	NC
E32	MM1_CKE1_H	U4	VSS	AP20	VSS
E31	MM1_PVT0	U3	VSS	AP19	VSS
E30	VDDQ15	U2	VSS	AP18	MT_VDDQ18
E29	MM1_A3_H	U1	VSS	AP17	VSS
E28	MM1_A4_H	V40	MM1_DQ9_H	AP16	VSS
E27	VSS	V39	MM1_DQ11_H	AP15	MT_VDDQ18
E26	MM_ATO	V38	VSS	AP14	VDD
E25	MM1_A10_H	V37	MM0_DQ27_H	AP13	VDD
E24	VDDQ15	V36	MM0_DQ28_H	AP12	VSS
E23	MM1_VREF	V35	VSS	AP11	VSS
E22	MM1_DQ40_H	V34	VSS	AP10	VDD
E21	VSS	V33	VSS	AP9	VDD
E20	MM1_DQ46_H	V8	VSS	AP8	VDD
E19	MM1_DQ55_H	V7	VSS	AP7	VDD
E18	VDDQ15	V6	PCI1_VDDA	AP6	VSS
E17	MM1_VDDQA2	V5	PCI1_TX6_P	AP5	VDD
E16	MM1_VSSQA2	V4	PCI1_TX6_N	AP4	VDD
E15	VSS	V3	VSS	AP3	VSS
E14	MM1_DQ58_H	V2	PCI1_RX6_P	AP2	VSS
E13	MM.DTO0	V1	PCI1_RX6_N	AP1	VDD
E12	VDDQ15	W38	MM1_DQS1_P	AR40	MM0_DQ4_H
E11	MM1_DQS7_P	W37	VDDQ15	AR39	VDDQ15
E10	MM1_DQ62_H	W36	MM0_DQ25_H	AR38	VSS
E9	CLK_VDDQ18	W35	MM0_VREF	AR37	VSS
E8	CFG_MM2_H	W34	VDD	AR36	VSS

E7	NC	W33	VDD	AR35	VSS
E6	NC	W8	VSS	AR34	VSS
E5	CLK_VSSA2	W7	VSS	AR33	VSS
E4	NMI_L	W6	PCI1_VDDA	AR32	VSS
E3	LVDS_VDDQA	W5	PCI1_TX7_P	AR31	VSS
E2	SEL_TAP0_L	W4	PCI1_TX7_N	AR30	PCIO_TX4_P
E1	RCLK_P	W3	VSS	AR29	PCIO_TX5_P
F40	MM1_DQS3_N	W2	PCI1_RX7_P	AR28	VSS
F39	MM1_DQS3_P	W1	PCI1_RX7_N	AR27	VSS
F38	VDDQ15	Y38	MM1_DQS1_N	AR26	VSS
F37	MM0_ODT1_H	Y37	MM0_DQ31_H	AR25	VSS
F36	MM0_BA2_H	Y36	VSS	AR24	MT_VDDQ18
F35	VSS	Y35	MM0_DQ24_H	AR23	VSS
F34	MM0_CKE0_H	Y34	VSS	AR22	CFG_SEL_L
F33	MM0_A2_H	Y33	VSS	AR21	MT_VDDQ18
F32	VDDQ15	Y8	VDD	AR20	TCK_H
F31	MM0_A6_H	Y7	VDD	AR19	TMS_H
F30	MM0_A8_H	Y6	PCI1_VDDA	AR18	MT_VDDQ18
F29	VSS	Y5	VSS	AR17	VSS
F28	MM0_A12_H	Y4	VSS	AR16	VSS
F27	MM0_A14_H	Y3	VSS	AR15	MT_VDDQ18
F26	VDDQ15	AA38	VDDQ15	AR14	VDD
F25	MM0_DQ41_H	AA37	MM0_DQS3_N	AR13	VDD
F24	MM0_DQS5_N	AA36	MM0_DQS3_P	AR12	VSS
F23	VSS	AA35	VDDQ15	AR11	VSS
F22	MM0_DQ54_H	AA34	VDDQ15	AR10	VDD
F21	MM0_DQ55_H	AA33	VDDQ15	AR9	VDD
F20	VDDQ15	AA8	VDD	AR8	VSS
F19	MM0_DQ49_H	AA7	VDD	AR7	VDD
F18	MM0_DQ50_H	AA6	VDD	AR6	VDD
F17	VSS	AA5	VDD	AR5	VSS
F16	MM0_DQ60_H	AA4	VDD	AR4	VDD
F15	MM0_DQ62_H	AA3	VDD	AR3	VDD

F14	VDDQ15	AB40	VSS	AR2	VSS
F13	MM1_DQ59_H	AB39	MM1_DQ12_H	AR1	VSS
F12	MM1_DQ56_H	AB38	MM1_DQ14_H	AT40	MM0_DQ1_H
F11	VSS	AB37	VSS	AT39	MM0_DQ3_H
F10	MM1_DQ61_H	AB36	MM0_DQ30_H	AT38	VSS
F9	CLK_VDDQ18	AB35	MM0_DQ29_H	AT37	PCIO_TX0_P
F8	CLK_VDDQ18	AB34	VSS	AT36	PCIO_TX1_P
F7	CLK_VDDQ18	AB33	VSS	AT35	VSS
F6	CLK_VDDQ18	AB8	VDD	AT34	PCIO_TX2_P
F5	CLK_VDDQ18	AB7	VDD	AT33	PCIO_TX3_P
F4	CLK_VDDQ18	AB6	VDD	AT32	VSS
F3	CLK_VDDQ18	AB5	VDD	AT31	VSS
F2	CLK_VDDQ18	AB4	VDD	AT30	PCIO_TX4_N
F1	CLK_VDDQ18	AB3	VDD	AT29	PCIO_TX5_N
G40	VDDQ15	AC40	MM1_DQ13_H	AT28	VSS
G39	MM1_DQ27_H	AC39	VDDQ15	AT27	PCIO_TX6_P
G38	MM0_CKE1_H	AC38	MM1_DQ15_H	AT26	PCIO_TX7_P
G37	VSS	AC37	MM0_DQ21_H	AT25	VSS
G36	MM0_ODT0_H	AC36	VDDQ15	AT24	MT_VDDQ18
G35	MM0_CK0_P	AC35	MM0_DQ23_H	AT23	VSS
G34	VDDQ15	AC34	VDD	AT22	INIT_MODE1_H
G33	MM0_A3_H	AC33	VDD	AT21	STAT_OUT_SEL1_L
G32	MM0_CK1_P	AC8	VSS	AT20	STAT_OUT3_L
G31	VSS	AC7	VSS	AT19	STAT_OUT4_L
G30	MM0_A5_H	AC6	VSS	AT18	NC
G29	MM0_A11_H	AC5	VSS	AT17	VSS
G28	VDDQ15	AC4	VSS	AT16	VSS
G27	MM0_A15_H	AC3	VSS	AT15	MT_VDDQ18
G26	MM0_DQ42_H	AC2	VSS	AT14	VDD
G25	VSS	AC1	VSS	AT13	VDD
G24	MM0_DQS5_P	AD40	MM1_DQ6_H	AT12	VSS
G23	MM0_DQ45_H	AD39	MM1_DQ5_H	AT11	VSS
G22	VDDQ15	AD38	VSS	AT10	VDD

G21	MM0_DQ48_H	AD37	MM0_DQS2_N	AT9	VDD
G20	MM0_DQS6_N	AD36	MM0_DQS2_P	AT8	VSS
G19	VSS	AD35	VSS	AT7	VSS
G18	MM0_DQ58_H	AD34	VSS	AT6	VDD
G17	MM0_DQ57_H	AD33	VSS	AT5	VDD
G16	VDDQ15	AD8	VSS	AT4	VSS
G15	MM0_DQ61_H	AD7	VSS	AT3	VDD
G14	MM0_ECC6_H	AD6	VSS	AT2	VDD
G13	VSS	AD5	VSS	AT1	VSS
G12	MM0_ECC7_H	AD4	VSS	AU40	VDDQ15
G11	MM0_ECC4_H	AD3	VSS	AU39	MM0_DQ0_H
G10	VDDQ15	AD2	VSS	AU38	VSS
G9	CLK_VDDQ18	AD1	VSS	AU37	PCI0_TX0_N
G8	VSS	AE40	VDDQ15	AU36	PCI0_TX1_N
G7	VSS	AE39	MM1_DQ7_H	AU35	VSS
G6	PCI1_VDDQA	AE38	MM0_DQ19_H	AU34	PCI0_TX2_N
G5	VSS	AE37	VDDQ15	AU33	PCI0_TX3_N
G4	VSS	AE36	MM0_DQ20_H	AU32	VSS
G3	VSS	AE35	MM0_DQ22_H	AU31	VSS
G2	VSS	AE34	VDD	AU30	VSS
G1	VSS	AE33	VDD	AU29	VSS
H40	MM1_DQ28_H	AE8	VDD	AU28	VSS
H39	VSS	AE7	VDD	AU27	PCI0_TX6_N
H38	MM1_DQ30_H	AE6	VDD	AU26	PCI0_TX7_N
H37	MM0_CS0_L	AE5	VDD	AU25	VSS
H36	VDDQ15	AE4	VDD	AU24	MT_VDDQ18
H35	MM0_CK0_N	AE3	VDD	AU23	MT_VDDQ18
H34	MM0_BA0_H	AE2	VDD	AU22	VSS
H33	VSS	AE1	VDD	AU21	WAKEUP_L
H32	MM0_CK1_N	AF40	MM1_DQ0_H	AU20	VSS
H31	MM0_A4_H	AF39	VSS	AU19	VSS
H30	VDDQ15	AF38	MM0_DQ18_H	AU18	MT_VDDQ18
H29	MM0_A9_H	AF37	MM0_DQ16_H	AU17	NC

H28	MM0_A10_H	AF36	VSS	AU16	MT_VDDQ18
H27	VSS	AF35	MM0_DQ17_H	AU15	MT_VDDQ18
H26	MM0_DQ40_H	AF34	VSS	AU14	VDD
H25	MM0_DQ43_H	AF33	VSS	AU13	VDD
H24	VDDQ15	AF8	VDD	AU12	VSS
H23	MM0_DQ47_H	AF7	VDD	AU11	VSS
H22	MM0_DQ53_H	AF6	VDD	AU10	VDD
H21	VSS	AF5	VDD	AU9	VDD
H20	MM0_DQS6_P	AF4	VDD	AU8	VDD
H19	MM0_DQ51_H	AF3	VDD	AU7	VSS
H18	VDDQ15	AF2	VDD	AU6	VSS
H17	MM0_DQ59_H	AF1	VDD	AU5	VDD
H16	MM0_DQS7_N	AG40	MM1_DQS0_N	AU4	VDD
H15	VSS	AG39	MM1_DQS0_P	AU3	VSS
H14	MM0_ECC5_H	AG38	VDDQ15	AU2	VDD
H13	MM0_DQS8_N	AG37	MM0_DQ12_H	AU1	VDD
H12	VDDQ15	AG36	MM0_DQ10_H	AV39	MM0_DQ2_H
H11	MM0_ECC1_H	AG35	VDDQ15	AV38	VSS
H10	MM0_ECC2_H	AG34	VDDQ15	AV37	VSS
H9	CLK_VDDQ18	AG33	VDDQ15	AV36	VSS
H8	VSS	AG8	VSS	AV35	VSS
H7	VSS	AG7	VSS	AV34	VSS
H6	PCI1_VDDQA	AG6	VSS	AV33	VSS
H5	PCI1_TX0_P	AG5	VSS	AV32	PCIO_REFRES
H4	PCI1_TX0_N	AG4	VSS	AV31	VSS
H3	VSS	AG3	VSS	AV30	PCIO_RX4_P
H2	PCI1_RX0_P	AG2	VSS	AV29	PCIO_RX5_P
H1	PCI1_RX0_N	AG1	VSS	AV28	VSS
J40	MM1_DQ29_H	AH40	VSS	AV27	VSS
J39	MM1_DQ31_H	AH39	MM1_DQ1_H	AV26	VSS
J38	VDDQ15	AH38	MM0_DQ8_H	AV25	VSS
J37	MM0_RAS_L	AH37	VSS	AV24	MT_VDDQ18
J36	MM0_CAS_L	AH36	MM0_DQ9_H	AV23	INIT_MODE0_H

J35	VSS	AH35	MM0_DQ11_H	AV22	STAT_OUT1_L
J34	MM0_BA1_H	AH34	VSS	AV21	STAT_OUT2_L
J33	MM0_A0_H	AH33	VSS	AV20	RESET_L
J32	VDDQ15	AH8	VSS	AV19	TDI_H
J31	MM0_A1_H	AH7	VSS	AV18	MT_CLK_H
J30	MM0_A7_H	AH6	VSS	AV17	TRST_L
J29	VSS	AH5	VSS	AV16	DCOK_H
J28	MM0_PVT2	AH4	VSS	AV15	MT_VDDQ18
J27	MM0_A13_H	AH3	VSS	AV14	VDD
J26	VDDQ15	AH2	VSS	AV13	VDD
J25	MM0_DQ44_H	AH1	VSS	AV12	VSS
J24	MM0_DQ46_H	AJ40	MM1_DQ3_H	AV11	VSS
J23	VSS	AJ39	VDDQ15	AV10	VDD
J22	MM0_VDDQA2	AJ38	MM0_DQS1_N	AV9	VDD
J21	MM0_VSSQA2	AJ37	MM0_DQS1_P	AV8	VDD
J20	VDDQ15	AJ36	VDDQ15	AV7	VDD
J19	MM0_DQ52_H	AJ35	MM0_PVT1	AV6	VSS
J18	MM0_DQ56_H	AJ34	VSS	AV5	VSS
J17	VSS	AJ33	VSS	AV4	VDD
J16	MM0_DQS7_P	AJ32	VSS	AV3	VDD
J15	MM0_DQ63_H	AJ31	VDD	AV2	VSS
J14	VDDQ15	AJ30	VSS	AV1	VDD
J13	MM0_DQS8_P	AJ29	VDD	AW38	VSS
J12	MM0_ECC0_H	AJ28	VSS	AW37	PCI0_RX0_P
J11	VSS	AJ27	VDD	AW36	PCI0_RX1_P
J10	MM0_ECC3_H	AJ26	VSS	AW35	VSS
J9	CLK_VDDQ18	AJ25	VDD	AW34	PCI0_RX2_P
J8	VSS	AJ24	VSS	AW33	PCI0_RX3_P
J7	VSS	AJ23	VDD	AW32	VSS
J6	PCI1_VDDQA	AJ22	VSS	AW31	VSS
J5	PCI1_TX1_P	AJ21	VDD	AW30	PCI0_RX4_N
J4	PCI1_TX1_N	AJ20	VSS	AW29	PCI0_RX5_N
J3	VSS	AJ19	VDD	AW28	VSS

J2	PCI1_RX1_P	AJ18	VSS	AW27	PCI0_RX6_P
J1	PCI1_RX1_N	AJ17	VDD	AW26	PCI0_RX7_P
K40	VSS	AJ16	VSS	AW25	VSS
K39	MM1_DQ22_H	AJ15	VSS	AW24	MT_VDDQ18
K38	MM0_WE_L	AJ14	VDD	AW23	MT_VDDQ18
K37	VSS	AJ13	SCL	AW18	TDO_H
K36	MM0_CS1_L	AJ12	VDD33	AW17	VSS
K35	MM0_VSSQA1	AJ11	VSS	AW16	MT_VDDQ18
K34	VSS	AJ9	VDD	AW15	MT_VDDQ18
K33	VSS	AJ8	VDD	AW14	VDD
K32	VDDQ15	AJ7	VDD	AW13	VDD
K31	VSS	AJ6	VDD	AW12	VSS
K30	VDD	AJ5	VDD	AW11	VSS
K29	VSS	AJ4	VDD	AW10	VDD
K28	VDD	AJ3	VDD	AW9	VDD
K27	VSS	AJ2	VDD	AW8	VSS
K26	VDDQ15	AJ1	VDD	AW7	VDD
K25	VSS	AK40	MM1_DQ4_H	AW6	VDD
K24	VDD	AK39	MM1_DQ2_H	AW5	VSS
K23	VSS	AK38	VSS	AW4	VSS
K22	VDD	AK37	MM0_DQ15_H	AW3	VDD
K21	VSS	AK36	MM0_VDDQA0	AW2	VDD
K20	VDDQ15	AK35	MM0_VSSQA0	AY37	PCI0_RX0_N
K19	VSS	AK34	VSS	AY36	PCI0_RX1_N
K18	VDD	AK33	VSS	AY35	VSS
K17	VSS	AK32	VSS	AY34	PCI0_RX2_N
K16	VDD	AK31	VDD	AY33	PCI0_RX3_N
K15	VSS	AK30	VSS	AY27	PCI0_RX6_N
K14	VDDQ15	AK29	VDD	AY26	PCI0_RX7_N
K13	VSS	AK28	VSS	AY25	VSS
K12	VDD	AK27	VDD	AY24	MT_VDDQ18
K11	VSS	AK26	VSS	AY23	STAT_OUT0_L
K10	VDD	AK25	VDD	AY18	MT_RX_H

K9	VSS	AK24	VSS	AY17	MT_TX_H
K8	PCI1_VTT	AK23	VDD	AY16	STAT_OUT_SEL0_L
K7	VSS	AK22	VSS	AY15	MT_VDDQ18
K6	PCI1_VDDQA	AK21	VDD	AY14	VDD
K5	VSS	AK20	VSS	AY13	VDD
K4	VSS	AK19	VDD	AY12	VSS
K3	VSS	AK18	VSS	AY11	VSS
K2	VSS	AK17	VDD	AY10	VDD
K1	VSS	AK16	VSS	AY9	VDD
L40	MM1_DQ21_H	AK15	VSS	AY8	VSS
L39	VDDQ15	AK14	VDD	AY7	VSS
L38	MM1_DQ23_H	AK13	EVENT	AY6	VDD
L37	MM0_RESET_L	AK12	SDA	AY5	VDD
L36	VDDQ15	AK11	VSS	AY4	VSS
L35	MM0_VDDQA1	AK10	VDD	AY3	VSS
L34	VSS	AK9	VDD		
L33	VSS	AK8	VDD		
L32	VDDQ15	AK7	VDD		
L31	VSS	AK6	VDD		
L30	VDD	AK5	VDD		

注：名称为“NC”的引脚为空引脚。在使用申威 411 处理器时，应保持这些引脚悬空，不连接任何信号。

2.3 CBGA 封装结构及尺寸

采用 CBGA 陶瓷封装 CBGA 封装，封装引脚数量 1144，其外观如图 2-4、图 2-5 和图 2-6 所示，图 2-4 为顶视图，图 2-5 为底视图，图 2-6 为侧视图。

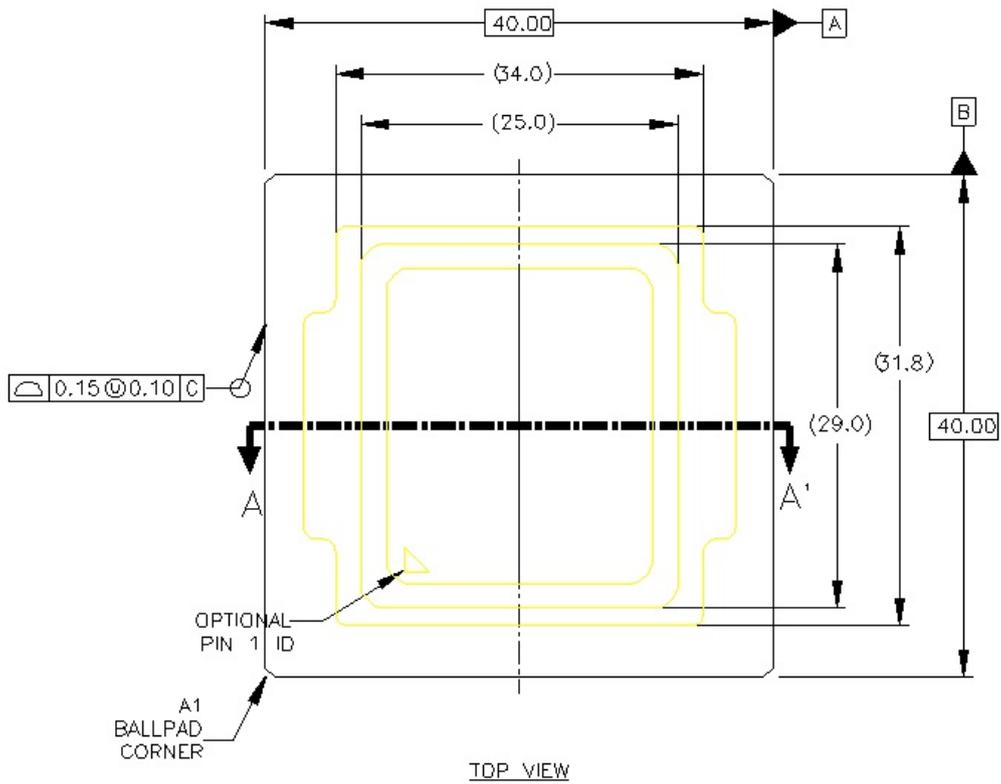


图 2-4: CBGA 封装顶视图

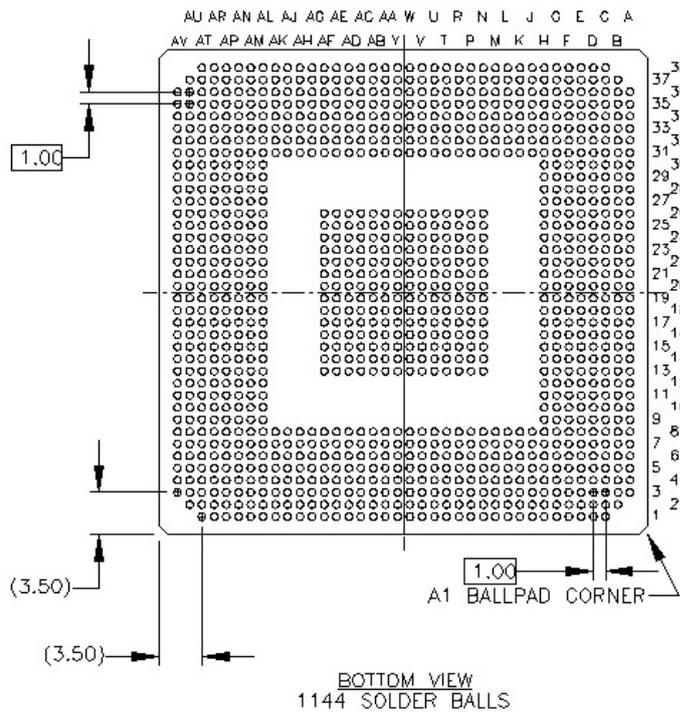


图 2-5: CBGA 封装底视图

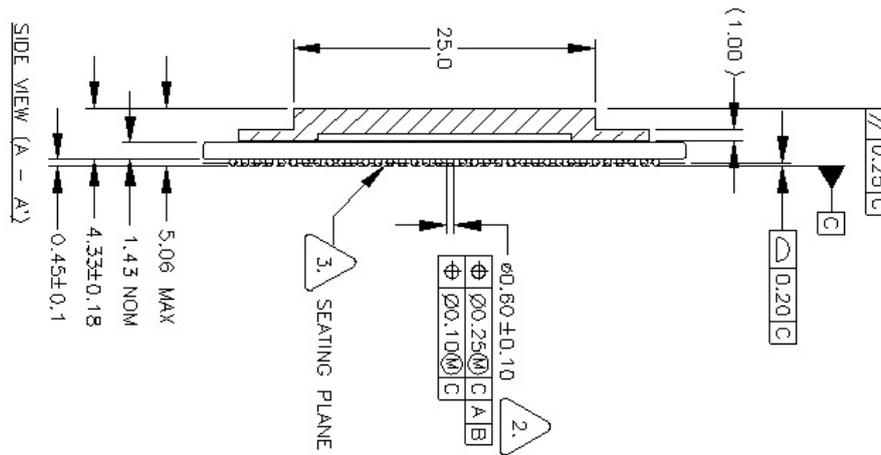


图 2-6: CBGA 封装侧视图

CBGA 封装的封装特征如下:

- 1) 封装尺寸 (长×宽×高): 40mm×40mm×4.78mm (含焊球高度误差: $\pm 0.28\text{mm}$);
- 2) 引脚数量: 1144;
- 3) 引脚为 38×38 阵列, 引脚分布参见图 2-5 (图中垂直方向的字母和水平方向的数字组成每个引脚的位置)。

2.4 CBGA 封装引脚分类及列表

申威 411 CBGA 封装的引脚总数为 1144 根, 表 2-4 为申威 411 CBGA 处理器引脚的分类与数量。

表 2-4: 申威 411 CBGA 处理器引脚分类列表

类型	数量
信号引脚	363
电地引脚	744
备份引脚 (保持悬空, 不连接)	37
合计	1144

表 2-5 为申威 411 处理器信号引脚的分类与数量。

表 2-5: CBGA 封装信号引脚分类列表

名称	数量	输入引脚	输出引脚	双向引脚	信号电平类型
存储器接口	257	0	69	188	SSTL (1.5V 或 1.35V)
PCI-E 接口	70	36	32	2	除 PCI[1:0]_RESREF(双向引脚)连接参考电阻外, 其他信号为 LVDS 类型;

维护接口	17	11	6	0	LVC MOS (1.8V)
时钟接口	12	12	0	0	一对差分输入为 LVDS, 其它为 LVC MOS (1.8V)
测试接口	7	6	1	0	LVC MOS (1.8V)
合计	363	65	108	190	

注：信号引脚的信号电平主要是 SSTL (1.5V 或 1.35V)、LVDS (1.8V) 和 LVC MOS (1.8V) 三大类，其中 SSTL 电平的 I/O 电源为 1.5V 或 1.35V，其它为 1.8V。

申威 411 CBGA 处理器的引脚位置与引脚名称对应关系如表 2-6 所示。其中位置是根据图 1-5 所指示的字母和数字组成，引脚名称尾缀为“_L”的信号为低电平有效的信号，尾缀为“_H”的信号为高电平有效的信号；引脚名称相同且尾缀分别为“_P”和“_N”的一对信号为配对的差分信号，其中尾缀为“_P”信号为“正”信号（高电平有效）。

表 2-6: CBGA 封装引脚位置与名称对应表

位置	引脚	位置	引脚	位置	引脚
A36	MM0_A3_H	N5	PCI1_TX2_P	AF8	VDD
A35	MM0_A5_H	N4	PCI1_TX2_N	AF7	VDD
A34	VDDQ15	N3	VSS	AF6	VDD
A33	MM0_A7_H	N2	PCI1_RX2_P	AF5	VDD
A32	MM0_A12_H	N1	PCI1_RX2_N	AF4	VDD
A31	VSS	P38	MM0_DQS4_N	AF3	VDD
A30	MM0_A14_H	P37	MM0_DQS4_P	AF2	VDD
A29	MM0_A15_H	P36	VDDQ15	AF1	VDD
A28	VDDQ15	P35	MM0_DQ32_H	AG38	VSS
A27	MM0_DQ40_H	P34	MM1_DQ33_H	AG37	MM0_DQ7_H
A26	MM0_DQS5_N	P33	VDDQ15	AG36	MM0_DQ5_H
A25	VSS	P32	MM1_DQS4_N	AG35	VSS
A24	MM0_DQ47_H	P31	MM1_DQS4_P	AG34	MM1_DQ7_H
A23	MM0_DQS6_N	P26	VSS	AG33	MM1_DQ6_H
A22	VDDQ15	P25	VDD	AG32	VSS
A21	MM1_VREF	P24	VSS	AG31	MM1_DQ11_H
A20	MM0_DQ53_H	P23	VDD	AG8	VDD
A19	VSS	P22	VSS	AG7	VDD
A18	MM0_DQ60_H	P21	VDD	AG6	VDD

A17	MM0_DQS7_N	P20	VSS	AG5	VDD
A16	VDDQ15	P19	VDD	AG4	VDD
A15	MM0_ECC1_H	P18	VSS	AG3	VDD
A14	MM0_DQS8_N	P17	VDD	AG2	VDD
A13	VSS	P16	VSS	AG1	VDD
A12	MM0_ECC4_H	P15	VDD	AH38	MM0_DQ4_H
A11	MM1_VDDQA0	P14	VSS	AH37	VDDQ15
A10	MM1_VDDQA1	P13	VDD	AH36	MM0_DQ6_H
A9	MM1_VDDQA2	P8	PCI1_VTT	AH35	MM1_DQ5_H
A8	LVDS_VDDQA	P7	VSS	AH34	VDDQ15
A7	NC	P6	PCI1_VDDQA	AH33	MM1_DQ4_H
A6	CLK_VDDA2	P5	PCI1_TX3_P	AH32	MM1_DQ3_H
A5	CLK_VDDA1	P4	PCI1_TX3_N	AH31	VDDQ15
A4	CLK_VDDA0	P3	VSS	AH8	VDD
A3	VSS	P2	PCI1_RX3_P	AH7	VDD
B37	MM0_RESET_L	P1	PCI1_RX3_N	AH6	VDD
B36	VDDQ15	R38	VSS	AH5	VDD
B35	MM0_A4_H	R37	MM0_DQ31_H	AH4	VDD
B34	MM0_CK1_P	R36	MM0_DQ30_H	AH3	VDD
B33	VSS	R35	VSS	AH2	VDD
B32	MM0_ODT0_H	R34	MM1_DQ32_H	AH1	VDD
B31	MM0_A11_H	R33	MM0_PVT0	AJ38	MM0_DQ3_H
B30	VDDQ15	R32	VSS	AJ37	MM0_DQ2_H
B29	MM0_A13_H	R31	MM1_DQ34_H	AJ36	VSS
B28	NC	R26	VDD	AJ35	MM0_DQ0_H
B27	VSS	R25	VSS	AJ34	MM1_DQ1_H
B26	MM0_DQS5_P	R24	VDD	AJ33	VSS
B25	MM0_DQ45_H	R23	VSS	AJ32	MM1_DQS0_N
B24	VDDQ15	R22	VDD	AJ31	MM1_DQS0_P
B23	MM0_DQS6_P	R21	VSS	AJ8	VSS
B22	MM0_DQ48_H	R20	VDD	AJ7	VSS
B21	VSS	R19	VSS	AJ6	VSS
B20	MM0_DQ54_H	R18	VDD	AJ5	VSS

B19	MM0_DQ56_H	R17	VSS	AJ4	VSS
B18	VDDQ15	R16	VDD	AJ3	VSS
B17	MM0_DQS7_P	R15	VSS	AJ2	VSS
B16	MM0_DQ63_H	R14	VDD	AJ1	VSS
B15	VSS	R13	VSS	AK38	VDDQ15
B14	MM0_DQS8_P	R8	PCI1_VTT	AK37	MM0_DQ1_H
B13	MM0_ECC3_H	R7	VSS	AK36	NC
B12	VDDQ15	R6	PCI1_CLK_P	AK35	VDDQ15
B11	MM1_VSSQA0	R5	VSS	AK34	MM1_DQ0_H
B10	MM1_VSSQA1	R4	VSS	AK33	VSS
B9	MM1_VSSQA2	R3	PCI1_REFRES	AK32	VSS
B8	LVDS_VSSQA	R2	VSS	AK31	VSS
B7	CLK_VDDQ18	R1	VSS	AK8	VSS
B6	CLK_VSSA2	T38	MM0_DQ29_H	AK7	VSS
B5	CLK_VSSA1	T37	VDDQ15	AK6	VSS
B4	CLK_VSSA0	T36	MM0_DQ28_H	AK5	VSS
B3	CLK_VDDQ18	T35	MM0_DQ27_H	AK4	VSS
B2	VSS	T34	VDDQ15	AK3	VSS
C38	VSS	T33	MM1_DQ30_H	AK2	VSS
C37	MM0_CS1_L	T32	MM1_DQ29_H	AK1	VSS
C36	MM0_CAS_L	T31	VDDQ15	AL38	NC
C35	VSS	T26	VSS	AL37	VSS
C34	MM0_CK1_N	T25	VDD	AL36	MM0_PVT1
C33	MM0_CKE1_H	T24	VSS	AL35	MM1_DQ2_H
C32	VDDQ15	T23	VDD	AL34	VSS
C31	MM0_A9_H	T22	VSS	AL33	PCI0_VTT
C30	MM0_A10_H	T21	VDD	AL32	PCI0_VTT
C29	VSS	T20	VSS	AL31	PCI0_VTT
C28	NC	T19	VDD	AL30	PCI0_VTT
C27	NC	T18	VSS	AL29	PCI0_VTT
C26	VDDQ15	T17	VDD	AL28	PCI0_VTT
C25	MM0_DQ43_H	T16	VSS	AL27	PCI0_VTT
C24	MM0_DQ46_H	T15	VDD	AL26	PCI0_VTT

C23	VSS	T14	VSS	AL25	VSS
C22	MM0_DQ50_H	T13	VDD	AL24	VSS
C21	MM0_DQ52_H	T8	PCI1_VTT	AL23	VSS
C20	VDDQ15	T7	VSS	AL22	VSS
C19	MM0_DQ57_H	T6	PCI1_CLK_N	AL21	CFG_SEL_L
C18	MM0_DQ59_H	T5	VSS	AL20	VSS
C17	VSS	T4	VSS	AL19	STAT_OUT_SEL_L1
C16	MM0_DQ62_H	T3	VSS	AL18	VSS
C15	MM0_ECC0_H	T2	VSS	AL17	NC
C14	VDDQ15	T1	VSS	AL16	VSS
C13	MM0_ECC2_H	U38	MM0_DQS3_N	AL15	NC
C12	MM0_ECC5_H	U37	MM0_DQS3_P	AL14	VSS
C11	VSS	U36	VSS	AL13	VSS
C10	NC	U35	MM0_DQ26_H	AL12	VDD
C9	NC	U34	MM1_DQ27_H	AL11	VDD
C8	VSS	U33	VSS	AL10	VDD
C7	CFG_MM0_H	U32	MM1_DQS3_N	AL9	VDD
C6	VSS	U31	MM1_DQS3_P	AL8	VSS
C5	SEL_TAP_L1	U26	VDD	AL7	VSS
C4	VSS	U25	VSS	AL6	VSS
C3	NC	U24	VDD	AL5	VSS
C2	VSS	U23	VSS	AL4	VSS
C1	VSS	U22	VDD	AL3	VSS
D38	MM0_WE_L	U21	VSS	AL2	VSS
D37	VDDQ15	U20	VDD	AL1	VSS
D36	MM0_A2_H	U19	VSS	AM38	NC
D35	MM0_A1_H	U18	VDD	AM37	MM0_VREF
D34	VDDQ15	U17	VSS	AM36	VSS
D33	MM0_A6_H	U16	VDD	AM35	VSS
D32	NC	U15	VSS	AM34	VSS
D31	VSS	U14	VDD	AM33	VSS
D30	MM_ATO	U13	VSS	AM32	VSS
D29	MM0_A8_H	U8	PCI1_VTT	AM31	VSS

D28	VDDQ15	U7	VSS	AM30	VSS
D27	MM0_DQ42_H	U6	PCI1_VDDA	AM29	VSS
D26	MM0_DQ41_H	U5	PCI1_TX4_P	AM28	VSS
D25	VSS	U4	PCI1_TX4_N	AM27	VSS
D24	MM0_DQ44_H	U3	VSS	AM26	VSS
D23	MM0_DQ49_H	U2	PCI1_RX4_P	AM25	VSS
D22	VDDQ15	U1	PCI1_RX4_N	AM24	VSS
D21	MM0_DQ51_H	V38	VDDQ15	AM23	VSS
D20	MM0_DQ55_H	V37	MM0_DQ24_H	AM22	VSS
D19	VSS	V36	MM0_DQ25_H	AM21	MT_VDDQ18
D18	MM0_DQ58_H	V35	VDDQ15	AM20	TDI_H
D17	MM0_DQ61_H	V34	MM1_DQ24_H	AM19	MT_VDDQ18
D16	VDDQ15	V33	MM1_DQ28_H	AM18	STAT_OUT_SEL_L0
D15	MM.DTO1	V32	VDDQ15	AM17	MT_VDDQ18
D14	MM1_PVT2	V31	MM1_DQ31_H	AM16	NC
D13	VSS	V26	VSS	AM15	MT_VDDQ18
D12	MM0_ECC6_H	V25	VDD	AM14	VSS
D11	MM0_ECC7_H	V24	VSS	AM13	VSS
D10	VDDQ15	V23	VDD	AM12	VDD
D9	NC	V22	VSS	AM11	VDD
D8	SPARE6_H	V21	VDD	AM10	VDD
D7	CLK_VDDQ18	V20	VSS	AM9	VDD
D6	CFG_MM2_H	V19	VDD	AM8	VSS
D5	CLK_VDDQ18	V18	VSS	AM7	VSS
D4	NC	V17	VDD	AM6	VSS
D3	CLK_VDDQ18	V16	VSS	AM5	VSS
D2	NC	V15	VDD	AM4	VSS
D1	CLK_VDDQ18	V14	VSS	AM3	VSS
E38	MM0_CS0_L	V13	VDD	AM2	VSS
E37	MM0_RAS_L	V8	PCI1_VTT	AM1	VSS
E36	VSS	V7	VSS	AN38	NC
E35	MM0_CKE0_H	V6	PCI1_VDDA	AN37	VSS
E34	MM1_CKE0_H	V5	PCI1_TX5_P	AN36	PCI0_VDDQA

E33	VSS	V4	PCI1_TX5_N	AN35	PCI0_VDDQA
E32	MM1_A7_H	V3	VSS	AN34	PCI0_VDDQA
E31	MM1_A6_H	V2	PCI1_RX5_P	AN33	PCI0_VDDQA
E30	VDDQ15	V1	PCI1_RX5_N	AN32	PCI0_VDDQA
E29	MM1_CS0_L	W38	MM0_DQ23_H	AN31	PCI0_VDDQA
E28	NC	W37	VSS	AN30	PCI0_CLK_P
E27	VSS	W36	MM0_DQ22_H	AN29	PCI0_CLK_N
E26	NC	W35	MM0_DQ21_H	AN28	PCI0_VDDA
E25	NC	W34	VSS	AN27	PCI0_VDDA
E24	VDDQ15	W33	MM1_DQ20_H	AN26	PCI0_VDDA
E23	MM1_DQ42_H	W32	MM1_DQ25_H	AN25	PCI0_VDDA
E22	MM1_DQ43_H	W31	VSS	AN24	PCI0_VDDA
E21	VSS	W26	VDD	AN23	PCI0_VDDA
E20	MM1_DQ48_H	W25	VSS	AN22	VSS
E19	MM1_DQ52_H	W24	VDD	AN21	TDO_H
E18	VDDQ15	W23	VSS	AN20	VSS
E17	MM1_DQ54_H	W22	VDD	AN19	TCK_H
E16	MM1_DQ61_H	W21	VSS	AN18	VSS
E15	VSS	W20	VDD	AN17	TMS_H
E14	MM1_DQ62_H	W19	VSS	AN16	VSS
E13	MM_DTO0	W18	VDD	AN15	NC
E12	VDDQ15	W17	VSS	AN14	VSS
E11	MM1_ECC3_H	W16	VDD	AN13	VSS
E10	MM1_ECC4_H	W15	VSS	AN12	VDD
E9	VSS	W14	VDD	AN11	VDD
E8	VSS	W13	VSS	AN10	VDD
E7	CFG_XBX2_H	W8	PCI1_VTT	AN9	VDD
E6	VSS	W7	VSS	AN8	VDD
E5	CFG_MM1_H	W6	PCI1_VDDA	AN7	VDD
E4	VSS	W5	VSS	AN6	VDD
E3	NC	W4	VSS	AN5	VDD
E2	VSS	W3	VSS	AN4	VDD
E1	RCLK_N	W2	VSS	AN3	VDD

F38	VDDQ15	W1	VSS	AN2	VDD
F37	MM0_ODT1_H	Y38	MM0_DQS2_N	AN1	VDD
F36	MM0_CK0_P	Y37	MM0_DQS2_P	AP38	MM0_VDDQA2
F35	VDDQ15	Y36	VDDQ15	AP37	MM0_VSSQA2
F34	MM1_CKE1_H	Y35	MM0_DQ20_H	AP36	VSS
F33	MM1_A3_H	Y34	MM1_DQ21_H	AP35	PCI0_TX0_P
F32	VDDQ15	Y33	VDDQ15	AP34	PCI0_TX1_P
F31	MM1_A5_H	Y32	MM1_DQ23_H	AP33	VSS
F30	MM1_A8_H	Y31	MM1_DQ26_H	AP32	PCI0_TX2_P
F29	VSS	Y26	VSS	AP31	PCI0_TX3_P
F28	MM1_A11_H	Y25	VDD	AP30	VSS
F27	MM1_A12_H	Y24	VSS	AP29	VSS
F26	VDDQ15	Y23	VDD	AP28	PCI0_TX4_P
F25	NC	Y22	VSS	AP27	PCI0_TX5_P
F24	MM1_DQ41_H	Y21	VDD	AP26	VSS
F23	VSS	Y20	VSS	AP25	PCI0_TX6_P
F22	MM1_DQ45_H	Y19	VDD	AP24	PCI0_TX7_P
F21	MM1_DQ47_H	Y18	VSS	AP23	VSS
F20	VDDQ15	Y17	VDD	AP22	DCOK_H
F19	MM1_DQ51_H	Y16	VSS	AP21	MT_VDDQ18
F18	MM1_DQ53_H	Y15	VDD	AP20	INIT_MODE1_H
F17	VSS	Y14	VSS	AP19	MT_VDDQ18
F16	MM1_DQ59_H	Y13	VDD	AP18	TRST_L
F15	MM1_DQ60_H	Y8	VSS	AP17	MT_VDDQ18
F14	VDDQ15	Y7	VSS	AP16	SPARE2_H
F13	MM1_DQ63_H	Y6	PCI1_VDDA	AP15	MT_VDDQ18
F12	MM1_ECC1_H	Y5	PCI1_TX6_P	AP14	VSS
F11	VSS	Y4	PCI1_TX6_N	AP13	VSS
F10	MM1_ECC5_H	Y3	VSS	AP12	VDD
F9	NC	Y2	PCI1_RX6_P	AP11	VDD
F8	SPARE4_H	Y1	PCI1_RX6_N	AP10	VDD
F7	CLK_VDDQ18	AA38	VSS	AP9	VDD
F6	CFG_XBX0_H	AA37	MM0_DQ17_H	AP8	VDD

F5	CLK_VDDQ18	AA36	MM0_DQ19_H	AP7	VDD
F4	CFG_CORE2_H	AA35	VSS	AP6	VDD
F3	CLK_VDDQ18	AA34	MM1_DQ16_H	AP5	VDD
F2	NC	AA33	MM1_DQ19_H	AP4	VDD
F1	RCLK_P	AA32	VSS	AP3	VDD
G38	MM0_BA0_H	AA31	MM1_DQ22_H	AP2	VDD
G37	VSS	AA26	VDD	AP1	VDD
G36	MM0_CK0_N	AA25	VSS	AR38	MM0_VDDQA1
G35	MM1_CK0_P	AA24	VDD	AR37	MM0_VSSQA1
G34	VSS	AA23	VSS	AR36	VSS
G33	MM1_BA1_H	AA22	VDD	AR35	PCI0_TX0_N
G32	MM1_A1_H	AA21	VSS	AR34	PCI0_TX1_N
G31	VSS	AA20	VDD	AR33	VSS
G30	MM1_A4_H	AA19	VSS	AR32	PCI0_TX2_N
G29	NC	AA18	VDD	AR31	PCI0_TX3_N
G28	VDDQ15	AA17	VSS	AR30	VSS
G27	MM1_A13_H	AA16	VDD	AR29	VSS
G26	NC	AA15	VSS	AR28	PCI0_TX4_N
G25	VSS	AA14	VDD	AR27	PCI0_TX5_N
G24	MM1_DQ40_H	AA13	VSS	AR26	VSS
G23	MM1_DQS5_N	AA8	VSS	AR25	PCI0_TX6_N
G22	VDDQ15	AA7	VSS	AR24	PCI0_TX7_N
G21	MM1_DQ46_H	AA6	PCI1_VDDA	AR23	VSS
G20	MM1_DQS6_N	AA5	PCI1_TX7_P	AR22	VSS
G19	VSS	AA4	PCI1_TX7_N	AR21	INIT_MODE0_H
G18	MM1_DQ50_H	AA3	VSS	AR20	VSS
G17	MM1_DQ55_H	AA2	PCI1_RX7_P	AR19	WAKEUP_L
G16	VDDQ15	AA1	PCI1_RX7_N	AR18	VSS
G15	MM1_DQ58_H	AB38	MM0_DQ16_H	AR17	MT_CLK_H
G14	MM1_DQS7_N	AB37	VDDQ15	AR16	VSS
G13	VSS	AB36	MM0_DQ14_H	AR15	NC
G12	MM1_ECC2_H	AB35	MM0_DQ18_H	AR14	VSS
G11	MM1_DQS8_N	AB34	VDDQ15	AR13	VSS

G10	VDDQ15	AB33	MM1_DQ17_H	AR12	VDD
G9	MM1_ECC6_H	AB32	MM1_DQ18_H	AR11	VDD
G8	VSS	AB31	VDDQ15	AR10	VDD
G7	CFG_XBX1_H	AB26	VSS	AR9	VDD
G6	VSS	AB25	VDD	AR8	VDD
G5	CFG_CORE0_H	AB24	VSS	AR7	VDD
G4	VSS	AB23	VDD	AR6	VDD
G3	SEL_TAP_L0	AB22	VSS	AR5	VDD
G2	VSS	AB21	VDD	AR4	VDD
G1	CLK_VDDQ18	AB20	VSS	AR3	VDD
H38	MM0_BA1_H	AB19	VDD	AR2	VDD
H37	MM0_A0_H	AB18	VSS	AR1	VDD
H36	VDDQ15	AB17	VDD	AT38	MM0_VDDQA0
H35	MM1_CK0_N	AB16	VSS	AT37	MM0_VSSQA0
H34	MM1_ODT0_H	AB15	VDD	AT36	VSS
H33	VDDQ15	AB14	VSS	AT35	VSS
H32	MM1_A0_H	AB13	VDD	AT34	VSS
H31	MM1_A2_H	AB8	VSS	AT33	VSS
H30	VDDQ15	AB7	VSS	AT32	VSS
H29	MM1_A9_H	AB6	PCI1_VDDA	AT31	VSS
H28	MM1_A10_H	AB5	VSS	AT30	PCI0_REFRES
H27	VSS	AB4	VSS	AT29	VSS
H26	MM1_A14_H	AB3	VSS	AT28	VSS
H25	MM1_A15_H	AB2	VSS	AT27	VSS
H24	VDDQ15	AB1	VSS	AT26	VSS
H23	MM1_DQS5_P	AC38	MM0_DQS1_N	AT25	VSS
H22	MM1_DQ44_H	AC37	MM0_DQS1_P	AT24	VSS
H21	VSS	AC36	VSS	AT23	VSS
H20	MM1_DQS6_P	AC35	MM0_DQ15_H	AT22	RESET_L
H19	MM1_DQ49_H	AC34	MM1_DQ8_H	AT21	MT_VDDQ18
H18	VDDQ15	AC33	VSS	AT20	STAT_OUT1_H
H17	MM1_DQ56_H	AC32	MM1_DQS2_N	AT19	MT_VDDQ18
H16	MM1_DQ57_H	AC31	MM1_DQS2_P	AT18	SPARE3_H

H15	VSS	AC26	VDD	AT17	MT_VDDQ18
H14	MM1_DQS7_P	AC25	VSS	AT16	NC
H13	MM1_ECC0_H	AC24	VDD	AT15	MT_VDDQ18
H12	VDDQ15	AC23	VSS	AT14	VSS
H11	MM1_DQS8_P	AC22	VDD	AT13	VSS
H10	MM1_ECC7_H	AC21	VSS	AT12	VDD
H9	VSS	AC20	VDD	AT11	VDD
H8	SPARE5_H	AC19	VSS	AT10	VDD
H7	CLK_VDDQ18	AC18	VDD	AT9	VDD
H6	VSS	AC17	VSS	AT8	VDD
H5	CFG_CORE1_H	AC16	VDD	AT7	VDD
H4	CFG_CORE3_H	AC15	VSS	AT6	VDD
H3	CLK_VDDQ18	AC14	VDD	AT5	VDD
H2	VSS	AC13	VSS	AT4	VDD
H1	NMI_L	AC8	VSS	AT3	VDD
J38	VSS	AC7	VSS	AT2	VDD
J37	MM1_PVT0	AC6	VSS	AT1	VDD
J36	MM1_CK1_P	AC5	VSS	AU37	VSS
J35	VSS	AC4	VSS	AU36	VSS
J34	MM1_BA2_H	AC3	VSS	AU35	PCI0_RX0_P
J33	MM1_BA0_H	AC2	VSS	AU34	PCI0_RX1_P
J32	VSS	AC1	VSS	AU33	VSS
J31	MM1_RAS_L	AD38	VDDQ15	AU32	PCI0_RX2_P
J8	VSS	AD37	MM0_DQ9_H	AU31	PCI0_RX3_P
J7	VSS	AD36	MM0_DQ12_H	AU30	VSS
J6	PCI1_VDDQA	AD35	VDDQ15	AU29	VSS
J5	VSS	AD34	MM1_DQ12_H	AU28	PCI0_RX4_P
J4	VSS	AD33	MM1_DQ9_H	AU27	PCI0_RX5_P
J3	VSS	AD32	VDDQ15	AU26	VSS
J2	VSS	AD31	MM1_DQ15_H	AU25	PCI0_RX6_P
J1	VSS	AD26	VSS	AU24	PCI0_RX7_P
K38	MM0_BA2_H	AD25	VDD	AU23	VSS
K37	VDDQ15	AD24	VSS	AU22	VSS

K36	MM1_CK1_N	AD23	VDD	AU21	STAT_OUT4_H
K35	MM0_PVT2	AD22	VSS	AU20	VSS
K34	VDDQ15	AD21	VDD	AU19	SPARE1_H
K33	MM1_RESET_L	AD20	VSS	AU18	VSS
K32	MM1_CS1_L	AD19	VDD	AU17	MT_TX_H
K31	VDDQ15	AD18	VSS	AU16	VSS
K8	VSS	AD17	VDD	AU15	NC
K7	VSS	AD16	VSS	AU14	VSS
K6	PCI1_VDDQA	AD15	VDD	AU13	VSS
K5	PCI1_TX0_P	AD14	VSS	AU12	VSS
K4	PCI1_TX0_N	AD13	VDD	AU11	VSS
K3	VSS	AD8	VSS	AU10	VSS
K2	PCI1_RX0_P	AD7	VSS	AU9	VSS
K1	PCI1_RX0_N	AD6	VSS	AU8	VSS
L38	MM0_DQ37_H	AD5	VSS	AU7	VSS
L37	MM0_DQ36_H	AD4	VSS	AU6	VSS
L36	VSS	AD3	VSS	AU5	VSS
L35	MM0_DQ38_H	AD2	VSS	AU4	VSS
L34	MM1_ODT1_H	AD1	VSS	AU3	VSS
L33	VSS	AE38	MM0_DQ8_H	AU2	VSS
L32	MM1_WE_L	AE37	VSS	AV36	VSS
L31	MM1_CAS_L	AE36	MM0_DQ10_H	AV35	PCI0_RX0_N
L8	VSS	AE35	MM0_DQ13_H	AV34	PCI0_RX1_N
L7	VSS	AE34	VSS	AV33	VSS
L6	PCI1_VDDQA	AE33	MM1_DQ14_H	AV32	PCI0_RX2_N
L5	PCI1_TX1_P	AE32	MM1_DQ13_H	AV31	PCI0_RX3_N
L4	PCI1_TX1_N	AE31	VSS	AV30	VSS
L3	VSS	AE26	VDD	AV29	VSS
L2	PCI1_RX1_P	AE25	VSS	AV28	PCI0_RX4_N
L1	PCI1_RX1_N	AE24	VDD	AV27	PCI0_RX5_N
M38	VDDQ15	AE23	VSS	AV26	VSS
M37	MM0_DQ35_H	AE22	VDD	AV25	PCI0_RX6_N
M36	MM1_PVT1	AE21	VSS	AV24	PCI0_RX7_N

M35	VDDQ15	AE20	VDD	AV23	VSS
M34	MM1_DQ37_H	AE19	VSS	AV22	STAT_OUT3_H
M33	MM1_DQ39_H	AE18	VDD	AV21	MT_VDDQ18
M32	VDDQ15	AE17	VSS	AV20	STAT_OUT0_H
M31	MM1_DQ38_H	AE16	VDD	AV19	MT_VDDQ18
M8	PCI1_VTT	AE15	VSS	AV18	STAT_OUT2_H
M7	VSS	AE14	VDD	AV17	MT_VDDQ18
M6	PCI1_VDDQA	AE13	VSS	AV16	MT_RX_H
M5	VSS	AE8	VDD	AV15	MT_VDDQ18
M4	VSS	AE7	VDD	AV14	VSS
M3	VSS	AE6	VDD	AV13	VSS
M2	VSS	AE5	VDD	AV12	VSS
M1	VSS	AE4	VDD	AV11	VSS
N38	MM0_DQ39_H	AE3	VDD	AV10	VSS
N37	VSS	AE2	VDD	AV9	VSS
N36	MM0_DQ33_H	AE1	VDD	AV8	VSS
N35	MM0_DQ34_H	AF38	MM0_DQS0_N	AV7	VSS
N34	VSS	AF37	MM0_DQS0_P	AV6	VSS
N33	MM1_DQ36_H	AF36	VDDQ15	AV5	VSS
N32	MM1_DQ35_H	AF35	MM0_DQ11_H	AV4	VSS
N31	VSS	AF34	MM1_DQ10_H	AV3	VSS
N26	VDD	AF33	VDDQ15		
N25	VSS	AF32	MM1_DQS1_N		
N24	VDD	AF31	MM1_DQS1_P		
N23	VSS	AF26	VSS		
N22	VDD	AF25	VDD		
N21	VSS	AF24	VSS		
N20	VDD	AF23	VDD		
N19	VSS	AF22	VSS		
N18	VDD	AF21	VDD		
N17	VSS	AF20	VSS		
N16	VDD	AF19	VDD		
N15	VSS	AF18	VSS		

N14	VDD	AF17	VDD		
N13	VSS	AF16	VSS		
N8	PCI1_VTT	AF15	VDD		
N7	VSS	AF14	VSS		
N6	PCI1_VDDQA	AF13	VDD		

3 电地及特殊信号接口

3.1 电地引脚列表

LGA 封装电地引脚如表 3-1 所示。CBGA 封装电地引脚如表 3-2 所示。

表 3-1: LGA 封装电地引脚列表

引脚名称	位宽	I/O	引脚类型	描述
VSS	401	—	地	地。
VDD	191	—	电源	内核电源。
VDDQ15	86	—	电源	DDR3 存储器接口 1.5V I/O 电源。
MM0_VDDQA[2:0]	3	—	模拟电源	DDR3 存储器接口 1.8V 模拟电源。
MM1_VDDQA[2:0]	3	—	模拟电源	DDR3 存储器接口 1.8V 模拟电源。
MM0_VSSQA[2:0]	3	—	模拟地	DDR3 存储器接口 1.8V 模拟地。
MM1_VSSQA[2:0]	3	—	模拟地	DDR3 存储器接口 1.8V 模拟地。
PCI[1:0]_VDDQA	12	—	模拟电源	PCI-E 接口 1.8V 模拟电源。
PCI[1:0]_VDDA	12	—	模拟电源	PCI-E 接口 0.9V 模拟电源。
PCI[1:0]_VTT	16	—	端接电源	PCI-E 接口端接电源。
CLK_VDDA[2:0]	3	—	模拟电源	时钟接口 0.9V 模拟电源。
CLK_VSSA[2:0]	3	—	模拟地	时钟接口 0.9V 模拟地。
LVDS_VDDQA	1	—	电源	LVDS 1.8V 模拟电源
LVDS_VSSQA	1	—	地	LVDS 模拟地
MT_VDDQ18	26	—	电源	GPIO 1.8V 电源
CLK_VDDQ18	19	—	电源	GPIO 1.8V 电源
总计	783	—	—	—

表 3-2: CBGA 封装电地引脚列表

引脚名称	位宽	I/O	引脚类型	描述
VSS	402	—	地	地。
VDD	186	—	电源	内核电源。
VDDQ15	68	—	电源	DDR3 存储器接口 1.5V I/O 电源。
MM0_VDDQA[2:0]	3	—	模拟电源	DDR3 存储器接口 1.8V 模拟电源。
MM1_VDDQA[2:0]	3	—	模拟电源	DDR3 存储器接口 1.8V 模拟电源。

MM0_VSSQA[2:0]	3	—	模拟地	DDR3 存储器接口 1.8V 模拟地。
MM1_VSSQA[2:0]	3	—	模拟地	DDR3 存储器接口 1.8V 模拟地。
PCI[1:0]_VDDQA	12	—	模拟电源	PCI-E 接口 1.8V 模拟电源。
PCI[1:0]_VDDA	12	—	模拟电源	PCI-E 接口 0.9V 模拟电源。
PCI[1:0]_VTT	16	—	端接电源	PCI-E 接口端接电源。
CLK_VDDA[2:0]	3	—	模拟电源	时钟接口 0.9V 模拟电源。
CLK_VSSA[2:0]	3	—	模拟地	时钟接口 0.9V 模拟地。
LVDS_VDDQA	1	—	电源	LVDS 1.8V 模拟电源
LVDS_VSSQA	1	—	地	LVDS 模拟地
MT_VDDQ18	16	—	电源	GPIO 1.8V 电源
CLK_VDDQ18	12	—	电源	GPIO 1.8V 电源
总计	744	—	—	—

3.1.1 数字电源地说明

申威 411 处理器需要外部系统提供三种数字电源，分别是内核电源 VDD、存储器接口 I/O 电源 VDDQ15 和 GPIO 接口电源 VDDQ18。LGA 封装和 CBGA 封装的电压范围相同，三种电源的电压范围见表 3-3。三种数字电源对应的数字地都是 VSS。

表 3-3: LGA 封装和 CBGA 封装数字电源及其电压范围

电源类型	参数	最小值	正常值	最大值	单位	数量
VDD	内核电源电压 V _{core}	0.91	0.95	0.99	V	191
VDDQ15	1.5V I/O 电源电压 V _{IO15}	1.425	1.5	1.575	V	86
MT_VDDQ18 CLK_VDDQ18	1.8V I/O 电源电压 V _{IO18}	1.71	1.8	1.89	V	45

3.1.2 模拟电源地说明

申威 411 处理器内部 PLL、存储器接口和 PCI-E 接口都需要外部提供模拟电源和模拟地，具体见表 3-4。

表 3-4: 申威 411 处理器模拟电源引脚及其连接要求

电源类型	数量	说明	连接方式
MM0_VDDQA[2:0]	3	DDR3 存储器接口 1.8V 模	每个引脚都必须分别通过图 3-1 所示

MM1_VDDQA[2:0]	3	拟电源。	的滤波器连接到数字电源 MT_VDDQ18 或 CLK_VDDQ18。
MM0_ASSQA[2:0]	3	DDR3 存储器接口 1.8V 模拟地。	在系统 PCB 上连接数字地 VSS。
MM1_ASSQA[2:0]	3		
PCI[1:0]_VDDQA	12	PCI-E 接口 1.8V 模拟电源。	每个引脚都必须分别通过图 3-1 所示的滤波器连接到数字电源 MT_VDDQ18 或 CLK_VDDQ18。
PCI[1:0]_VDDA	12	PCI-E 接口 0.9V 模拟电源。	每个引脚都必须分别通过图 3-1 所示的滤波器连接到数字电源 VDD。
PCI[1:0]_VTT	16	PCI-E 接口端接电源，电压可以是 1.5V 或 1.8V	PCI-E 通道发送方向的端接电源，可选择连接 1.5V 或 1.8V 电源。
CLK_VDDA[2:0]	3	时钟接口 0.9V 模拟电源。	每个引脚都必须分别通过图 3-1 所示的滤波器连接到数字电源 VDD。
CLK_VSSA[2:0]	3	时钟接口 0.9V 模拟地。	连接数字地 VSS。
LVDS_VDDQA	1	LVDS 1.8V 模拟电源	每个引脚都必须分别通过图 3-1 所示的滤波器连接到数字电源 MT_VDDQ18 或 CLK_VDDQ18。
LVDS_VSSQA	1	LVDS 1.8V 模拟地	在系统 PCB 上连接数字地 VSS。

注：PCI-E 接口两种模拟电源没有对应的模拟地，而是直接采用数字地 VSS 作为模拟地。

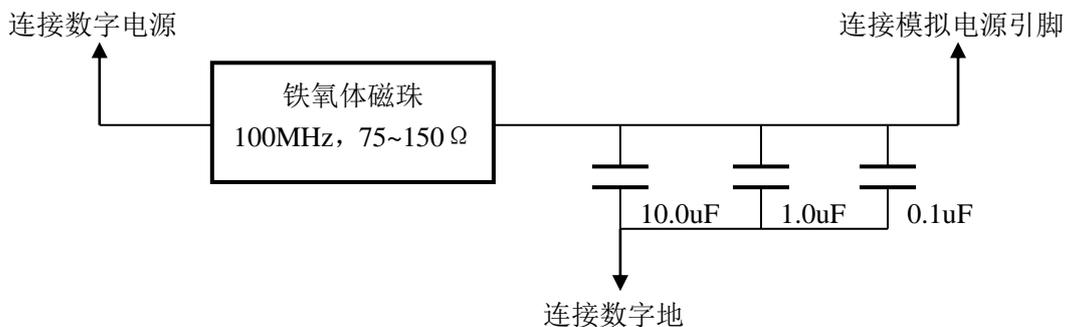


图 3-1：连接模拟电源引脚的滤波器

3.2 温度传感器引脚列表

温度传感器引脚仅限 LGA 封装，CBGA 封装没有该类引脚。

表 3-5：LGA 封装温度传感器引脚列表

引脚名称	位宽	I/O	引脚类型	描述
------	----	-----	------	----

VDD33	1	—	电源	温度传感器的 3.3V 电源。
SCL	1	输入	Open-drain	I2C 标准时钟信号，需要外接上拉电阻。
SDA	1	双向	Open-drain	I2C 标准数据信号，需要外接上拉电阻。
EVENT	1	输出	Open-drain	温度报警信号，需要外接上拉电阻。
总计	4	—	—	—

3.3 空引脚列表

LGA 封装备份引脚如表 3-7 所示。

表 3-6: LGA 封装空引脚列表

引脚名称	位宽	I/O	引脚类型	描述
NC	6	—	空引脚	空引脚，使用申威 411 处理器时，保持此类引脚悬空，不连接任何信号。

CBGA 封装备份引脚如表 3-8 所示。

表 3-7: CBGA 封装空引脚列表

引脚名称	位宽	I/O	引脚类型	描述
NC	37	—	空引脚	空引脚，使用申威 411 处理器时，保持此类引脚悬空，不连接任何信号。

4 时钟接口

4.1 引脚列表

申威 411 处理器的时钟接口为内部 PLL 提供参考时钟。处理器内部集成了三个时钟锁相环 PLL，分别根据外部输入的参考时钟，产生内核时钟、存储控制器时钟（包括 DDR3 存储器接口时钟）和互连时钟。核心处于内核时钟域，存储器控制器接口处于存控时钟域，一致性处理部件和系统接口处理部件处于互连时钟域。时钟接口相关的引脚信号如表 4-1 所示。

维护接口有独立的时钟域，时钟由引脚 MT_CLK_H 直接输入。PCI-E 接口也有独立的时钟域，时钟由芯片引脚输入的参考时钟（PCI[1:0]_CLK_P/PCI[1:0]_CLK_N）通过 PCI-E 接口内的 PLL 产生。

表 4-1：时钟复位接口引脚信号

信号名称	位宽	I/O	引脚类型	描述
RCLK_P	1	输入	LVDS	LVDS 电平的差分参考时钟输入，典型工作频率为 200MHz，为 1.8V LVDS 电平信号。
RCLK_N	1			
共计	2			

4.2 工作模式

申威 411 处理器的时钟接口存在两种工作模式，一种是正常工作模式，另一种为 PLL 旁路模式。旁路模式主要用于处理器的测试。

在正常工作模式下，输入参考时钟 RCLK 的频率典型值为 200MHz，通过内部三个独立的 PLL 以及时钟配置引脚 CFG_CORE[3:0]_H、CFG_MM[2:0]_H 和 CFG_XBX[2:0]_H 以及相应配置寄存器 INIT_CTL，分别产生核心时钟、存储器控制器时钟（包括 DDR3 存储器接口时钟）和互连时钟。PCI-E 接口时钟则是通过 PCI-E 接口输入参考时钟 PCI_CLK（频率固定为 100MHz）和 PCI-E 接口内部的 PLL 产生。为保证申威 411 处理器正常工作，必须在各电源稳定、输入参考时钟稳定和相关配置引脚电平固定情况下，才能结束申威 411 的复位（包含上电复位和冷复位），进入正常工作状态。

配置引脚 CFG_CORE[3:0]_H、CFG_MM[2:0]_H 或 CFG_XBX[2:0]_H 设置为“全零”，分别使得核心、存储控制器及 DDR3 存储器接口、互连时钟处于 PLL 旁路模式。此模式下，核心时钟频率与输入参考时钟 RCLK 频率相同，维护接口时钟频率仍与输入的维护时钟 MT_CLK_H 相同，可根据 MIU IOR: PCIE_x_CLK_SEL_x 选择 PCI-E 接口时钟频率与维护时钟 MT_CLK_H 相同。

5 存储器接口

5.1 概述

申威 411 处理器内部集成了两个 DDR3 存储控制器, 每个存储控制器对应 64 位数据宽度的 DDR3 存储器接口, 该接口支持“64 位数据+8 位校验码”的 ECC 校验 (可纠正单错、检测双错), 最高数据传输率可达 1600Mbps, 主要特点如下:

- 1) 支持 DDR3-800/1066/1333/1600Mbps 等数据传输速率;
- 2) 支持的存储器容量为 2、4、8 或 16GB;
- 3) 支持直接连接×8 和×16 位结构的 DDR3 SDRAM 存储器芯片, 也支持连接单 Rank 或双 Rank 的 DDR3 UDIMM 或 RDIMM 存储器条;
- 4) 连接 DDR3 SDRAM 存储器芯片或 UDIMM 存储器条时, 支持 1T 和 2T 模式;
- 5) 支持双 Rank DIMM 存储器条的地址镜像功能;
- 6) 支持对 RDIMM 存储器条的配置寄存器 RC (Register Control) 访问;
- 7) 支持 JESD79-3A 标准定义的各种延迟参数动态调节功能;
- 8) 支持 DDR3 SDRAM 存储器芯片的浅睡眠 (Precharge PowerDown) 和深睡眠 (Self Refresh);
- 9) 支持对 DDR3 SDRAM 存储器芯片的模式寄存器 MR0~MR3 进行实时配置;
- 10) 支持突发长度为 8 (BL8) 的存储器读写访问;
- 11) 支持交错的 Wrap 地址访问方式;
- 12) 支持链路训练 (Write Leveling) 功能;
- 13) 支持读写通路的 DQS/DQ 延迟调节功能;
- 14) 支持环回测试 (LoopBack)。

5.2 引脚列表

申威 411 处理器的 DDR3 存储器接口引脚信号共 257 根, 其中双向信号 188 根, 输出信号 69 根, 均采用 SSTL (1.5V 或 1.35V) 电平信号。具体如表 5-1 所示。

表 5-1: DDR3 存储器接口引脚信号

信号名称	位宽	I/O	引脚类型	描述
信号引脚				
MM0_RESET_L	1	输出	1.5V/1.35V	DDR3 存储器异步复位信号。
MM0_CK_P[1:0]	2	输出	SSTL	DDR3 存储器差分输入参考时钟。

MM0_CK_N[1:0]	2	输出		
MM0_CKE_H[1:0]	2	输出		对应每个 Rank 的时钟使能信号。
MM0_ODT_H[1:0]	2	输出		对应每个 Rank 的内部端接使能信号。
MM0_CS_L[1:0]	2	输出		对应每个 Rank 的片选信号。
MM0_RAS_L	1	输出		行地址 RAS#命令。
MM0_CAS_L	1	输出		列地址 CAS#命令。
MM0_WE_L	1	输出		写 WE#命令。
MM0_BA_H[2:0]	3	输出		存储器体 (Bank) 地址。
MM0_A_H[15:0]	16	输出		存储器地址。
MM0_DQS_P[8:0]	9	双向		以字节为单位的差分读/写数据选通信号。
MM0_DQS_N[8:0]	9	双向		
MM0_DQ_H[63:0]	64	双向		读/写数据。
MM0_ECC_H[7:0]	8	双向		读/写数据的 ECC 码。
MM1_RESET_L	1	输出		DDR3 存储器异步复位信号。
MM1_CK_P[1:0]	2	输出		DDR3 存储器差分输入参考时钟。
MM1_CK_N[1:0]	2	输出		
MM1_CKE_H[1:0]	2	输出		对应每个 Rank 的时钟使能信号。
MM1_ODT_H[1:0]	2	输出		对应每个 Rank 的内部端接使能信号。
MM1_CS_L[1:0]	2	输出		对应每个 Rank 的片选信号。
MM1_RAS_L	1	输出		行地址 RAS#命令。
MM1_CAS_L	1	输出		列地址 CAS#命令。
MM1_WE_L	1	输出		写 WE#命令。
MM1_BA_H[2:0]	3	输出		存储器体 (Bank) 地址。
MM1_A_H[15:0]	16	输出		存储器地址。
MM1_DQS_P[8:0]	9	双向		以字节为单位的差分读/写数据选通信号。
MM1_DQS_N[8:0]	9	双向		
MM1_DQ_H[63:0]	64	双向		读/写数据。
MM1_ECC_H[7:0]	8	双向		读/写数据的 ECC。
总计	246	—		
参考电压、PVT 补偿电阻和测试引脚				
MM0_VREF	1	—	参考电压	参考电压。
MM1_VREF	1	—	参考电压	参考电压。
MM0_PVT[2:0]	3	—	PVT 电阻	PVT 补偿电阻。
MM1_PVT[2:0]	3	—	PVT 电阻	PVT 补偿电阻。

MM_ATO	1	输出	1.5V/1.35V	ATO 引脚。	测试引脚，可悬空。
MM.DTO[1:0]	2	输出	SSTL	DTO 引脚。	
总计	11	—			

注1：申威411处理器在连接DDR3 SDRAM存储器芯片或DDR3 DIMM存储器条时，存储器芯片或存储器条的DM引脚应接地。

注2：申威411处理器在连接RDIMM存储器条时，存储器条的PAR_IN引脚接地（下拉）、QCSEN_N引脚接VDDQ15（上拉），ERROUT_N引脚悬空。

5.3 存储器接口配置

5.3.1 连接方式

申威 411 处理器的 DDR3 存储器接口支持多种连接配置，既支持单 Rank 存储器，也支持双 Rank 存储器，既支持连接 DDR3 SDRAM 存储器芯片，也支持连接 DDR3 DIMM 存储器条，具体的连接配置如表 5-2 所示。

表 5-2：存储器接口的连接配置方式

MC_CTRL [RANKSEL]	CfgCR [RDIMM]	编号	说明
0	0	配置 1	用于连接 DDR3 SDRAM 存储器芯片或单 Rank 的 UDIMM 存储器条。
0	1	配置 2	用于连接单 Rank 的 RDIMM 存储器条。
1	0	配置 3	用于连接双 Rank 的 UDIMM 存储器条。
1	1	配置 4	用于连接双 Rank 的 RDIMM 存储器条。

上表中 MC_CTRL、CfgCR 是申威 411 处理器内部存储控制器中的 I/O 寄存器，可用于对存储器接口的配置。

5.3.2 1T/2T 模式

DDR3 存储器接口工作在高负载和高频率条件下，为提高信号传输的可靠性，可以选择将存储器访问的地址与命令加长为两个周期，但片选信号仍维持一个时钟周期，即 2T 模式，以增加命令和地址的建立时间。可通过配置 IOR: CfgCR[2T]为“1”来使能 2T 模式。1T 和 2T 模式下的信号传输如图 5-1 和图 5-2 所示（图中以存储器接口 0 的 PHY0 为例进行说明，且图中的命令是针对 Rank0）。

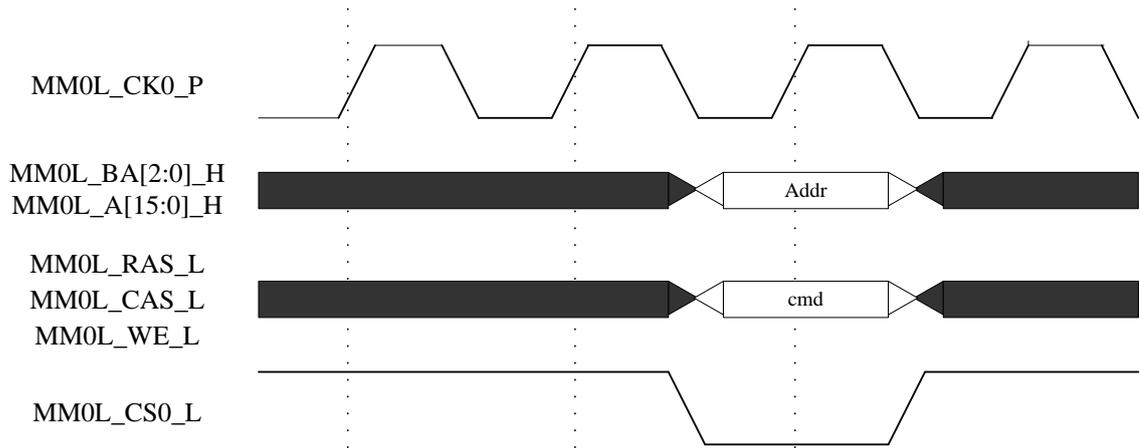


图 5-1: 1T 传输模式示意图

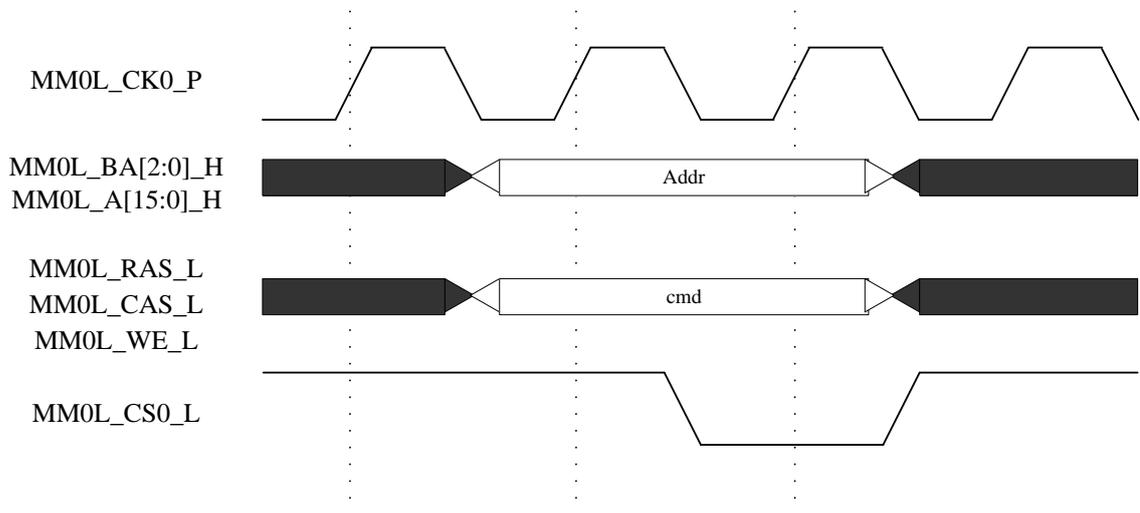


图 5-2: 2T 传输模式示意图

2T 传输模式可以增强 DDR3 存储器接口信号的完整性，但存储器访问命令之间的间隔增加，可能会降低存储器访问的性能。

5.3.3 地址镜像

采用地址镜像，可以缩短两个 Rank 的 DDR3 SDRAM 存储器芯片在 PCB 上双面贴装时 PCB 布线的长度，使得 PCB 上正面与反面贴装的 DDR3 SDRAM 存储器芯片使用不同排列顺序的地址、命令信号。申威 411 处理器支持的地址镜像如表 5-3 所示，其它存储器接口信号则直接连接。可配置 CfgCR[AddressMirror]为“1”来使能地址镜像功能。

表 5-3: 地址镜像时存储器接口信号引脚连接方式

存储器接口信号引脚	连接的 DDR3 SDRAM 存储器芯片引脚	
	Rank 0	Rank 1
MM0/1_A_H3	MM0/1_A_H3	MM0/1_A_H4

MM0/1_A_H4	MM0/1_A_H4	MM0/1_A_H3
MM0/1_A_H5	MM0/1_A_H5	MM0/1_A_H6
MM0/1_A_H6	MM0/1_A_H6	MM0/1_A_H5
MM0/1_A_H7	MM0/1_A_H7	MM0/1_A_H8
MM0/1_A_H8	MM0/1_A_H8	MM0/1_A_H7
MM0/1_BA_H0	MM0/1_BA_H0	MM0/1_BA_H1
MM0/1_BA_H1	MM0/1_BA_H1	MM0/1_BA_H0

5.3.4 颗粒/内存类型

申威 411 处理器支持的存储容量及颗粒类型如表 5-4:

表 5-4: 存储器接口支持的主存类型 (容量、Rank 数量、颗粒类型)

SDRAM 位宽	颗粒类型	颗粒容量	单 Rank 容量	双 Rank 容量
64bit/ 72bit(ECC)	x8	1Gb	1GB	2GB
		2Gb	2GB	4GB
		4Gb	4GB	8GB
		8Gb	8GB	--
	x16	1Gb	--	1GB
		2Gb	1GB	2GB
		4Gb	2GB	4GB
		8Gb	4GB	8GB

5.4 信号连接

5.4.1 片选信号

申威 411 处理器的每路存储器接口片选信号 (MM0_CS_L[1:0]和 MM1_CS_L[1:0]), 其中 MM0_CS_L[1:0]对应 DIMM0, MM1_CS_L[1:0]对应 DIMM1。图 5-3 给出了片选的连接方式。

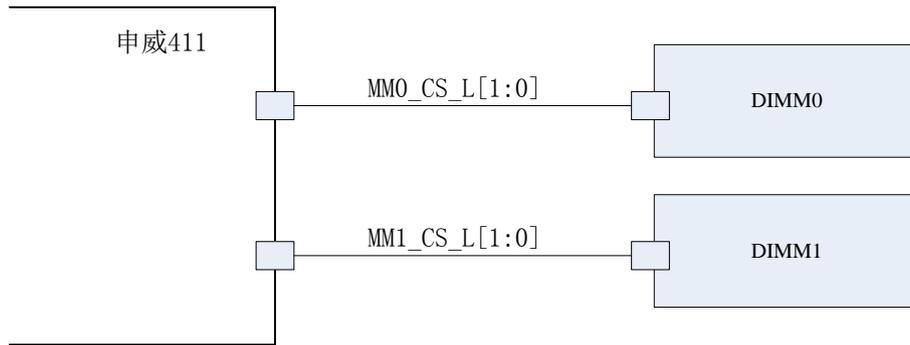


图 5-3: 片选信号连接示意图

5.4.2 选通信号

申威 411 处理器存储器接口数据选通信号为差分的双向三态信号（MM0_DQS[8:0]_L、MM0_DQS[8:0]_P、MM1_DQS[8:0]_L、MM1_DQS[8:0]_P），为避免噪声干扰，芯片内部采用带上/下拉电阻的 IO 单元（PDQSR_VSSQ）对 MM0_DQS[8:0]_P、MM1_DQS[8:0]_P 进行下拉，对 MM0_DQS[8:0]_L、MM1_DQS[8:0]_L 进行上拉。阻值在 344~688 欧姆之间，通过数据通用配置寄存器 DXCCR[DQSRES]控制 MM0_DQS[8:0]_P、MM1_DQS[8:0]_P，通过数据通用配置寄存器 DXCCR[DQSNRES]控制 MM0_DQS[8:0]_L、MM1_DQS[8:0]_L。

5.4.3 ECC 校验码

存储器接口数据宽度为 128 位，实际信号宽度为 144 位，其中 16 位为 ECC 校验位。为支持带 ECC 校验的 DDR3 DIMM 存储器条，推荐的连接方式如表 5-5 所示（表中 dqs[8:0]为 DIMM 存储器条的数据选通信号引脚，cb[7:0]为 DIMM 存储器条的 ECC 校验码引脚，dq[63:0]为 DIMM 存储器条的读写数据信号引脚）。

表 5-5: 带 ECC 校验的 DDR3 DIMM 存储器条连接方式

存储器接口引脚信号	DIMM 存储器条引脚	连接说明
MM0_DQS_N[8]	DIMM0.dqs[8]	存储器接口 0 的 64 位数据对应的 ECC 校验位
MM0_ECC_H[7:0]	DIMM0.cb[7:0]	
MM1_DQS_N[8]	DIMM1.dqs[8]	存储器接口 1 的 64 位数据对应的 ECC 校验位
MM1_ECC_H[7:0]	DIMM1.cb[7:0]	
MM0_DQS_N[7:0]	DIMM0.dqs[7:0]	存储器接口 0 的 64 位数据
MM0_DQ_H[63:0]	DIMM0.dq[63:0]	
MM1_DQS_N[7:0]	DIMM1.dqs[7:0]	存储器接口 1 的 64 位数据
MM1_DQ_H[63:0]	DIMM1.dq[63:0]	

5.4.4 连接存储器芯片

申威 411 处理器的存储器接口可以直接连接 DDR3 SDRAM 存储器芯片，每个存储器接口驱动各自的存储器颗粒，连接关系可参照存储器条的连接方式。

5.4.5 连接存储器条

申威 411 处理器的存储器接口可连接单 Rank 和双 Rank 的 DDR3 DIMM 存储器条，既可以连接 UDIMM 存储器条，也可以连接 RDIMM 存储器条。

图 5-4 为连接双 Rank 的 DIMM 存储器条的示意图。单 Rank 的 DIMM 存储器条的连接方式与双 Rank 相同。

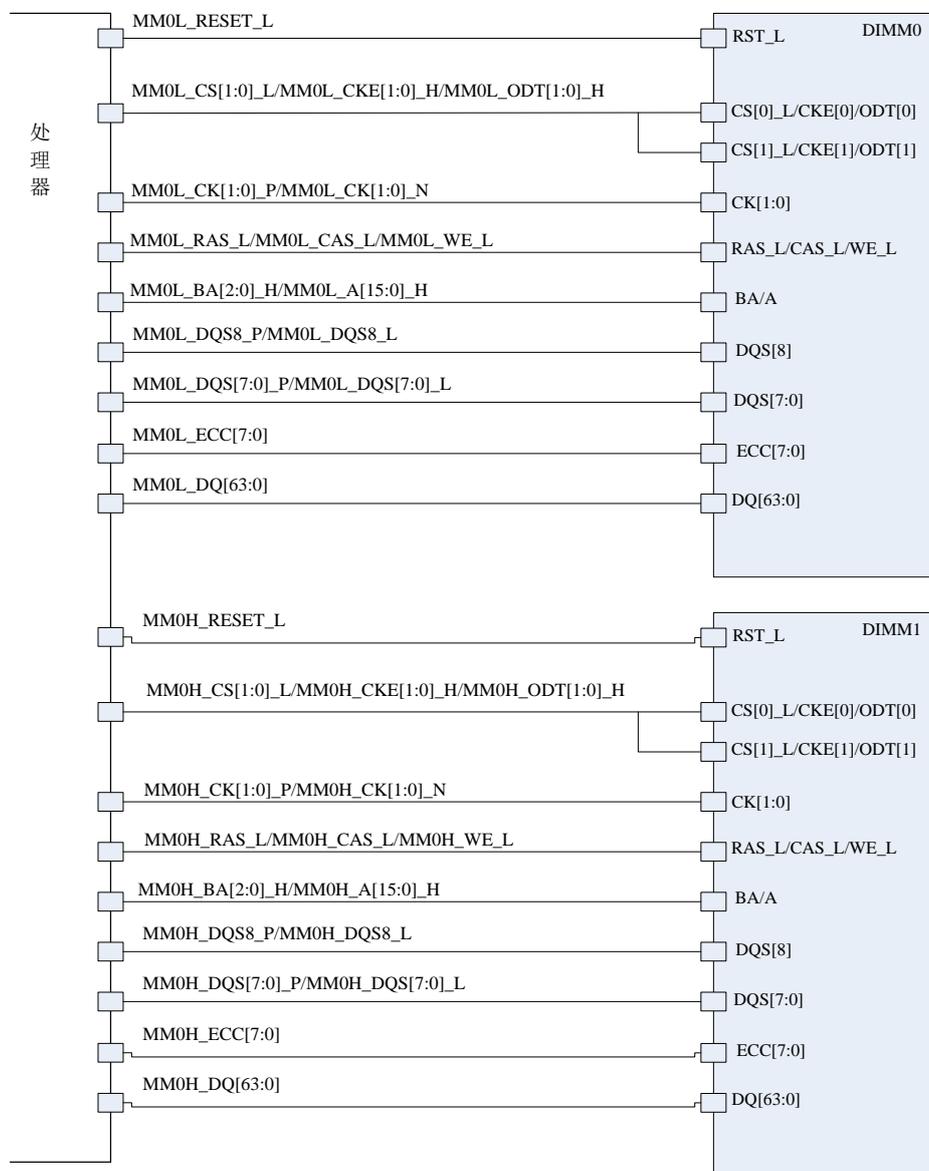


图 5-4: 双 Rank 的 DIMM 存储器条信号引脚连接方式

5.5 初始化

申威 411 处理器在上电复位和冷复位结束后，需要先根据系统中所使用的 DDR3 SDRAM 存储器或 DDR3 DIMM 存储器条类型，通过维护接口或初始化程序配置存储控制器中对应的 I/O 寄存器，然后配置 DDR3 PHY 内的 IO 寄存器，启动 DDR3 存储器接口进行初始化。

申威 411 处理器的存储器接口在初始化结束之后，需要进行存储器接口链路训练（Data Training），链路训练是对存储器接口的数据总线以 8 位为单位与对应的差分数据选通信号（DQS）进行校准、字节内部逐位对齐的过程，通过链路训练可以提高存储器接口输出信号的完整性，提高抗干扰能力。处理器内部的四个核心共享一路存储器接口，可以选择任何一个核心在初始化加载程序运行时进行链路训练。具体链路训练方法可参考《申威 411 处理器管理员手册》。

5.6 测试

申威 411 处理器的存储器接口支持环回测试，主要用于测试内部存储控制器的读写数据通路是否正常。测试时，设置相应的 I/O 寄存器，配置具体的环回测试模式和数据模式，启动测试，读相应的 I/O 寄存器判断环回测试是否成功。

6 PCI-E 接口

6.1 概述

申威 411 处理器集成了两个符合 2.0 标准的 PCI-E 接口。每个接口都具备极性翻转、通道反转、链路自协商等特性，所有符合 PCI-E 标准的设备均可与处理器的 PCI-E 接口连接。该接口作为根端口 (Root Complex)，同时支持 256 个 MSI-X 类型中断及 4 个 INTx 中断，兼容 32 位和 64 位 PCI-E 地址空间的访问，单向差分链路传输速率可达 5Gbps，8 条链路并行传输的双向有效带宽可达到 8GB/s。

6.2 引脚列表

申威 411 处理器的 PCI-E 接口采用 LVDS 电平，接口信号共有 70 根，其中 36 根输入信号、32 根输出信号以及 2 根双向引脚，具体如表 6-1 所示。

表 6-1: PCI-E 接口引脚信号

信号名称	位宽	I/O	引脚类型	描述
PCI[1:0]_TX_P[7:0]	16	输出	LVDS	差分输出的 8 对数据信号。
PCI[1:0]_TX_N[7:0]	16			
PCI[1:0]_RX_P[7:0]	16	输入	LVDS	差分输入的 8 对数据信号。
PCI[1:0]_RX_N[7:0]	16			
PCI[1:0]_CLK_P	2	输入	LVDS	差分输入时钟信号,正常工作频率为 100MHz。
PCI[1:0]_CLK_N	2			
PCI[1:0]_RESREF	2	—	参考电阻	参考电阻; 每个 PCI-E 接口共用一根; 连接 200Ω±1% 的高精度电阻, 电阻另一端接数字地 VSS。
总计	70			

6.3 信号连接

申威 411 处理器集成符合 2.0 标准的 PCI-E 接口，该接口作为根端口 (Root Complex)，可以连接 Endpoint、SWICTH、PCI-E TO PCI/PCI-X Bridge 等类型的 PCI-E 设备，连接示意图如图 6-1 所示 (以一个 PCI-E 接口为例)。

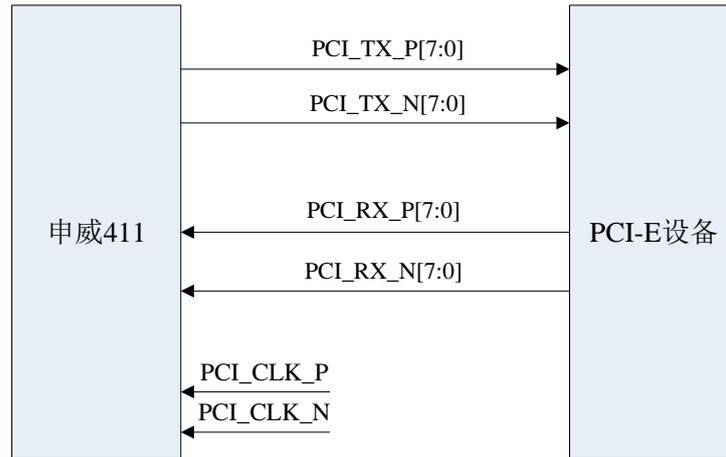


图 6-1: PCI-E 接口信号连接示意图

每个 PCI-E 接口包含 8 条通道（单通道带宽 5Gbps），可以支持单链路连接和 4 链路连接。同时 PCI-E 接口向下兼容 1.0 标准（单通道带宽 2.5Gbps），可以连接支持 1.0 标准的 PCI-E 设备。

6.4 初始化

申威 411 处理器可以连接标准的 PCI-E 端点（EP）或路由（SWITCH）设备。在差分引脚（PCI[1:0]_CLK_P/PCI[1:0]_CLK_N）输入 100MHz 的参考时钟稳定后，结束对处理器的上电复位和冷复位，处理器通过 PCI-E 接口开始检测连接的 PCI-E 设备，并自动与连接的设备交互 TS 序列，协商链路宽度与速度。可通过读取 PCI-E 接口内部的“RC Debug 信息寄存器”，判断链路建立是否成功。

软件检测到链路协商成功之后，还需对 PCI-E 接口进行一些初始化配置，主要流程如下：

- 1) 进行 PCI 兼容性配置，如打开 I/O 空间、存储器空间和总线的使能等；
- 2) 配置设备的工作模式，如请求最大负载（Max_Payload_Size），保序模式等；
- 3) 配置中断相关寄存器，以保证 PCI-E 接口产生的 MSI 以及 INTx 中断能正确产生与传递。

6.5 测试

申威 411 处理器的 PCI-E 接口支持低频模式测试。

设置维护接口 IOR: PCIE_x_CLK_SEL_x，可以使 PCI-E 接口工作在维护时钟频率下，此时可以通过维护接口访问 PCI-E 接口内部 I/O 寄存器，判断低频模式下基本的 I/O 寄存器访问通路是否正确。

7 维护、中断和配置接口

7.1 维护接口

7.1.1 概述

申威 411 处理器的维护接口实现复位、初始化加载和调试支持以及运行监测等功能，一部分功能是通过维护接口的引脚信号直接实现，其它则是通过维护接口支持的维护命令来实现。维护接口的功能和特点如下：

- 1) 实现对处理器的复位和初始化加载；
- 2) 支持处理器的运行状态监测及错误报告；
- 3) 支持读写处理器连接的存储器和内部各部件的 I/O 寄存器；
- 4) 支持对处理器各核心上运行程序的单步调试、断点调试等调试支持功能；
- 5) 支持对处理器各核心、存储控制器的断连，支持核心的睡眠和唤醒。

7.1.2 引脚列表

申威 411 处理器的维护接口有 12 根引脚信号，其中 6 根输入信号，6 根输出信号，均为 1.8V LVC MOS 电平，具体如表 7-1 所示。

表 7-1：维护接口引脚信号

信号名称	位宽	I/O	引脚类型	描述
MT_CLK_H	1	输入	1.8V LVC MOS	维护接口工作时钟,由系统提供, 频率为 5~25MHz。
MT_TX_H	1	输出		维护串口输出数据, 同步于 MT_CLK_H, 在维护时钟的下降沿输出。
MT_RX_H	1	输入		维护串口输入数据, 同步于 MT_CLK_H, 在维护时钟的上升沿接收。
RESET_L	1	输入		冷复位, 低电平有效。
DCOK_H	1	输入		电源就绪, 低电平时作为上电复位。
STAT_OUT_L[4:0]	5	输出		芯片测试输出。
STAT_OUT_SEL_L[1:0]	2	输入		芯片测试输出选择。
共计	27			

其中, STAT_OUT_L[4:0]为维护接口的状态输出信号,其输出受 STAT_OUT_SEL_L[1:0]的控制。具体含义如表 7-2 所示:

表 7-2: STAT_OUT_L[4:0]具体定义

STAT_OUT_SEL_L	域	STAT_OUT_L[4:0]输出含义
2'b00、2'b01		未定义。
2'b10	[4]	自定义串口维护命令错标志, 低电平有效 。
	[3:0]	维护主状态机状态(在维护时钟的下降沿输出),具体如下: 4'b0000 DCOKRST 表示上电复位状态(等待 DCOK_H 有效); 4'b0001 COLDRST 表示冷复位状态(等待 RESET_L 无效); 4'b0010 CONFIG1 表示配置 1 状态(MCU 配置); 4'b0011 WAITPLL 表示 PLL 升频状态; 4'b0100 WAITUPCLK 表示时钟切换 1 状态; 4'b0101 SOCRSTEND 表示结束复位状态; 4'b0110 CONFIG2 表示配置 2 状态(存储器自测试配置); 4'b0111 MEMBIST 表示存储器自测试状态; 4'b1000 CONFIG3 表示配置 3 状态(其它接口配置); 4'b1001 SROMLD 表示初始化程序加载状态; 4'b1010 STARTRUN 表示启动运行状态; 4'b1011 RUN 表示运行状态; 4'b110x ENTERRST 表示进入复位状态; 4'b111x WAITDOWNCLK 表示时钟切换 2 状态。
2'b11	[4]	芯片总错标志(包括维护接口错), 低电平有效 。
	[3]	中断完成标志, 低电平有效 。
	[2]	存储器自测试完成, 低电平有效 。
	[1:0]	存储器自测试结果,含义如下: 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。

7.1.3 维护功能

7.1.3.1 复位

维护接口支持对申威 411 处理器的复位，包括上电复位、冷复位和热复位（睡眠/唤醒）。其中上电复位由引脚信号 DCOK_H 为低电平引起，冷复位由引脚信号 RESET_L 为低电平引起，热复位通过睡眠唤醒流程来产生。

维护接口复位命令是一种特殊复位，只复位处理器的维护接口部分逻辑，不影响处理器的其它逻辑，主要是在维护接口自身出现故障而无法支持维护命令处理时，此信号引脚可以使得维护接口恢复正常，继续进行正常的维护功能。

7.1.3.2 维护串口

引脚信号 MT_TX_H 和 MT_RX_H 组成一对同步的串行接口，同步于维护时钟 MT_CLK_H，称之为维护串口。外部系统可通过 MT_RX_H 向申威 411 处理器发送维护命令，并从 MT_TX_H 回收维护命令处理的维护响应。维护命令只能串行处理，维护命令都遵循发送、处理和返回响应的串行处理流程，外部系统必须在收到前一个维护命令返回的响应后，才能发出新的维护命令。串行输入的维护命令以数据包格式传输，称之为维护命令包。串行输出的维护响应也以数据包格式传输，称之为维护响应包。维护命令包和维护响应包都由若干字节组成，字节数与具体的命令和响应类型相关。具体可参见附录 A。

维护命令中包含维护中断，用于外部系统向申威 411 处理器内部的核心发送各种维护中断，以实现更多的维护功能，包括对核心上运行程序的调试支持。

7.1.3.3 主要维护命令功能

初始化程序加载

申威 411 处理器结束复位后，可通过维护接口向每个核心的指令 Cache 加载初始化程序，由初始化程序完成进一步初始化设置和相关测试。所有处于连接状态的核心（最多 4 个）一起进行指令 Cache 加载，且加载的内容完全相同。

初始化程序加载以 Cache 行（128 字节）为基本单位，加载数据自动从指令 Cache（两路联想）的第 0 路的第 0 个 Cache 行开始装填，地址依次递增，直到接收到初始化程序加载结束命令为止。当第 0 路加载满后，再从第 1 路的第 0 个 Cache 行开始装填。因此，初始化程序的大小不能超过指令 Cache 的容量（32KB）。

存储器自测试

申威 411 处理器内部含有大量的存储器（内部逻辑用到的缓冲、Cache 等资源），在上电复位/

冷复位、睡眠/唤醒过程中，可以对这些支持存储器自测试，自测试包括 BIST、BISR 和 ScanBISR 三种测试模式。

- 1) 自动迭代的存储器可修复自测试 (ScanBISR)：遍历灵敏放大器的配置值，对存储器进行自测试，发现错误，自动使用冗余的存储器单元对存储器进行替换，并根据选择策略选择灵敏放大器的值，以保证存储器正确性；
- 2) 指定灵敏放大器配置值的存储器可修复自测试 (BISR)：指定灵敏放大器的配置值，对存储器进行自测试，发现错误，自动使用冗余的存储器单元对存储器进行替换，以保证存储器正确性；
- 3) 指定灵敏放大器配置值的存储器自测试 (BIST)：指定灵敏放大器的配置值，对存储器进行自测试，发现错误，不支持冗余替换。

上电复位、冷复位会启动片内所有存储器阵列的存储器自测试，核心睡眠唤醒会启动相应核心内所有存储器阵列的存储器自测试。存储器自测试结束以后，测试结果会登记在维护接口 IOR：BIST_STAT、CG_BIST_STAT 和 PC_BIST_STAT，可以通过维护命令读取该测试结果。对于自测试类型，可通过配置维护 IOR：BIST_CTL 进行指定。建议处理器使用指定灵敏放大器的值为“1”进行自测试。

处理器内部的存储器及其支持的存储器自测试类型如表 7-3 所示。

表 7-3：存储器自测试类型

存储器阵列名称	容量	存储器自测试类型
指令 Cache 数据存储单元	32KB	ScanBISR/BISR/BIST
指令 Cache 标记存储器	10Kb	无
指令 Cache 标记副本存储器	5.6Kb	无
数据 Cache 数据存储单元	32KB	ScanBISR/BISR/BIST
数据 Cache 标记存储器	6.75Kb	无
二级 Cache 数据存储单元	512KB	ScanBISR/BISR/BIST
二级 Cache 标记存储器	15.5KB	ScanBISR/BISR/BIST
三级 Cache 数据存储单元	156KB	ScanBISR/BISR/BIST
三级 Cache 标记存储器	6MB	ScanBISR/BISR/BIST
Page Cache 数据存储单元	9.75Kb	无
Page Cache 标记存储器	32KB	BISR/BIST

故障监测

申威 411 处理器内部主要的数据通路和存储器阵列都设置了校验，基本实现了数据传输的连续

校验。当发现不可纠正的错误，或可纠正的错误达到一定数量时，维护接口会通过 STAT_OUT_L[4] 引脚变为低电平来通知外部系统，此时外部系统可以通过 I/O 寄存器读维护命令来获得故障的详细信息，进行故障分析、诊断，协助进行系统错误处理和容错。

调试支持

申威 411 处理器的维护接口支持系统的软硬件调试，一方面，通过维护命令可以直接读出内部 I/O 寄存器以及连接的主存储器中的信息；另一方面，通过维护命令可以向内部某个核心发出维护中断，实现更为复杂的维护功能，如设置和清除断点、进行单步调试和断点调试。

7.2 配置接口

7.2.1 引脚列表

申威 411 处理器的配置接口有 13 根输入引脚信号，均为 1.8V LVCMOS 电平，具体如表 7-4 所示。

表 7-4: 配置接口引脚信号

信号名称	位宽	I/O	引脚类型	描述
CFG_CORE_H[3:0]	4	输入	1.8V LVCMOS	核心时钟 PLL 配置，为 1.8V LVCMOS 电平信号。
CFG_MM_H[2:0]	3	输入		存储器接口时钟 PLL 配置，为 1.8V LVCMOS 电平信号。
CFG_XBX_H[2:0]	3	输入		互连时钟 PLL 配置，为 1.8V LVCMOS 电平信号。
INIT_MODE_H[1:0]	2	输入		初始化模式，选择复位后初始化方式（具体内容见表 7-2）。
CFG_SEL_L	1	输入		配置方式选择。 “0”：（初始化配置和时钟配置）采用引脚配置方式； “1”：（初始化配置和时钟配置）采用维护配置方式相关引脚信号无效，可以采用缺省配置，也可以通过维护命令进行重新配置。

7.2.2 初始化配置

配置接口可实现对申威 411 处理器进行初始化配置，同时也能反映处理器内部的主要状态。

引脚信号 INIT_MODE[1:0]_H 用来选择申威 411 处理器在结束上电复位或冷复位后，进行初始化的方式，正常工作模式下只能设置为“11”，其它设置主要用于对处理器的测试或调试支持。

表 7-5: INIT_MODE 模式说明

INIT_MODE_H[1:0]	2'b00	2'b01	2'b10	2'b11
初始化状态机状态				
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 寄存器和存储器自测试相关的扫描链寄存器	
存储器自测试状态	可在配置 2	无操作	进行存储器自测试	
配置 3 状态	状态下修改	无操作	可读写所有 I/O 寄存器	
初始化加载状态	参数配置寄	无操作	进行初始化程序加载	
启动运行状态	存器来选择	停在此状态	延时等待	
进入复位状态	无操作			延时等待
应用场景	测试模拟	程序模拟	运行模拟	正常运行

说明：在引脚 CFG_SEL_L 为“0”时，引脚 INIT_MODE_H[1:0]决定初始化的方式，其中 2'b00~2'b10 为模拟测试模式，分别用于测试模拟（如存储器自测试）、程序模拟（运行模拟验证程序）和运行模拟（仿真实际运行模式），采用缺省配置加快模拟速度，并缩短部分状态处理的等待时间，如时钟切换间隔在快速模式下为 2 个维护时钟周期，否则缺省为 65535 个维护时钟周期（可通过维护命令设置）。

7.2.3 时钟配置

1) 核心时钟配置：根据参考时钟输入引脚 (RCLK) 的时钟频率和核心时钟配置引脚 CFG_CORE [3:0]_H 或者维护配置寄存器 INIT_CTL[CORE_FREQ]，通过核心 PLL 产生核心时钟。在 RCLK 输入时钟频率为 200MHz 的条件下，可配置核心时钟频率为 300~1725MHz。在该 PLL 旁路模式下，

核心时钟频率即为输入参考时钟 RCLK 的频率。核心时钟配置引脚的具体定义如表 7-6 所示。

表 7-6: 核心时钟配置表 (引脚)

CFG_CORE_H[3:0]	核心工作频率 (MHz)
0	旁路 (200MHz)
1	400
2	600
3	800
4	1000
5	1100
6	1200
7	1300
8	1350
9	1400
10	1450
11	1500
12	1550
13	1600
14	1650
15	1700

2) 存储控制器时钟配置: 根据参考时钟输入引脚 (RCLK) 和存储器接口配置引脚 CFG_MM [2:0]_H 或者维护寄存器 INIT_CTL[MM_FREQ], 通过存控 PLL 产生存控时钟。在 RCLK 输入时钟频率为 200MHz 的条件下, PLL 输出频率 233MHz~583MHz。在该 PLL 旁路模式下, 存储控制器使用为 200MHz。存储控制器时钟配置引脚的具体定义如表 7-7 所示。

表 7-7: 存储控制器时钟配置表 (引脚)

CFG_MM_H[2:0]	MC 时钟频率 (MHz)
0	旁路 (200MHz)
1	266
2	300
3	333
4	366
5	400
6	466
7	533

3) 互连时钟配置: 根据参考时钟输入引脚 (RCLK) 的时钟频率和互连时钟配置引脚 CFG_XBX

[3:0]_H 或者维护寄存器 INIT_CTL[XBX_FREQ]，通过互联 PLL 产生互连时钟。在 RCLK 输入时钟频率为 200MHz 的条件下，可配置互连时钟频率为 300~1350MHz。在该 PLL 旁路模式下，互连时钟频率即为 200MHz。互连时钟配置引脚的具体定义如表 7-8 所示。

表 7-8: 互连时钟配置表（引脚）

CFG_XBX_H[2:0]	CPM 及 IPU 工作频率 (MHz)
0	旁路 (200MHz)
1	600
2	900
3	1050
4	1150
5	1250
6	1300
7	1325

4) 为能提供更多的时钟频率配置，MCU 内部设置配置寄存器 INIT_CTL，其中核心时钟配置为 5 位，CPM 时钟配置为 4 位，MC 时钟配置为 4 位，核心、存控和互连时钟配置的寄存器配置表如表 7-9、表 7-10 和表 7-11 所示。

表 7-9: 核心时钟配置表（寄存器）

IOR_CFG_CORE[4:0]	核心工作频率 (MHz)
0	旁路 (200MHz)
1	300
2	400
3	500
4	600
5	700
6	800
7	900
8	1000
9	1050
10	1100
11	1150
12	1200
13	1250
14	1300
15	1325

16	1350
17	1375
18	1400
19	1425
20	1450
21	1475
22	1500
23	1525
24	1550
25	1575
26	1600
27	1625
28	1650
29	1675
30	1700
31	1725

表 7-10: 存储控制器时钟配置表 (寄存器)

IOR_CFG_MM[3:0]	MC 时钟频率 (MHz)
0	旁路
1	233
2	266
3	283
4	300
5	316
6	333
7	350
8	366
9	383
10	400
11	433
12	466
13	500
14	533

15	583
----	-----

表 7-11: 互连时钟配置表 (寄存器)

IOR_CFG_XBX[3:0]	CPM 及 IPU 工作频率 (MHz)
0	旁路 (200MHz)
1	400
2	600
3	800
4	900
5	1000
6	1050
7	1100
8	1150
9	1200
10	1250
11	1275
12	1300
13	1312.5
14	1325
15	1350

5) PCI-E 接口时钟配置: PCI-E 参考时钟 PCI_CLK (差分输入引脚 PCI_CLK_P/PCI_CLK_N, 频率为 100MHz), 通过 PLL 产生 PCI-E 接口时钟, PCI-E 接口时钟频率为 250MHz。可根据维护接口 IOR: PCIE_x_CLK_SEL_x 选择 PCI-E 接口时钟频率与维护时钟 MT_CLK_H 相同, 以降低功耗。

7.3 中断接口

申威 411 处理器的中断接口有 2 根输入引脚信号, 均为 1.8V LVCMOS 电平, 具体如表 7-12 所示。

表 7-12: 中断接口引脚信号

信号名称	位宽	I/O	引脚类型	描述
WAKEUP_L	1	输入	1.8V	唤醒中断, 默认低电平有效 (有效方式可

			LVC MOS	通过维护寄存器进行配置)。唤醒中断引脚的使用说明见表 7-4。
NMI_L	1	输入		不可屏蔽中断，默认低电平有效（有效方式可通过维护寄存器进行配置）。不可屏蔽中断引脚的使用说明见表 7-4。

中断引脚的使用说明如表 7-13 所示。

表 7-13: 中断引脚使用说明

中断类型	配置寄存器	备注
WAKEUP_L	<p>WAKEUP_CTL[MASK_EN]作为唤醒中断使能，该位为“1”表示使能唤醒中断。</p> <p>WAKEUP_CTL[TYPE_SEL]作为有效类型选择，该位为“0”表示低电平有效，该位为“1”表示高电平有效。</p>	为防止抖动，有效电平（高电平或低电平）要至少维持 16 个维护时钟周期。
NMI_L	<p>WAKEUP_CTL[MASK_EN]作为不可屏蔽中断使能，该位为“1”表示使能不可屏蔽中断。</p> <p>WAKEUP_CTL[NMI_CTL]作为有效类型选择：</p> <p>2`b00: 表示上升沿有效；</p> <p>2`b01: 表示下降沿有效；</p> <p>2`b10: 表示高电平有效；</p> <p>2`b11: 表示低电平有效。</p>	为防止抖动，有效电平（高电平或低电平）要至少维持 16 个维护时钟周期。

8 测试接口

8.1 概述

申威 411 处理器的测试接口由符合 IEEE1149.1 标准的 JTAG 测试信号组成，工作频率为 5~10MHz，具体功能包括：

- 1) 支持维护接口和时钟接口（统称为 GPIO 接口）的边界扫描测试；
- 2) 支持 DDR3 存储器接口的边界扫描测试；
- 3) 支持 PCI-E 接口的边界扫描测试；
- 4) 支持维护操作。

8.2 引脚列表

申威 411 处理器的测试接口有 7 根接口信号，其中 6 根输入信号，1 根输出信号，均为 1.8V LVC MOS 电平，具体如表 8-1 所示。

表 8-1: 测试接口信号

信号名称	位宽	I/O	引脚类型	描述
TCK_H	1	输入	1.8V LVC MOS	测试时钟。
TMS_H	1	输入		测试模式选择。
TRST_L	1	输入		测试复位。
TDI_H	1	输入		测试数据输入。
TDO_H	1	输出		测试数据输出。
SEL_TAP_L[1:0]	2	输入		TAP 控制器配置。 2'b00: 表示使用 PCI-E0 的 TAP; 2'b01: 表示使用 PCI-E1 的 TAP; 2'b1x: 表示使用 TBOX 的 TAP。
共计	7			

8.3 TAP 控制器

申威 411 处理器测试接口设置有符合 IEEE1149.1 标准的 TAP 控制器，用于测试接口的控制。该 TAP 控制器的状态转换如图 8-1 所示。

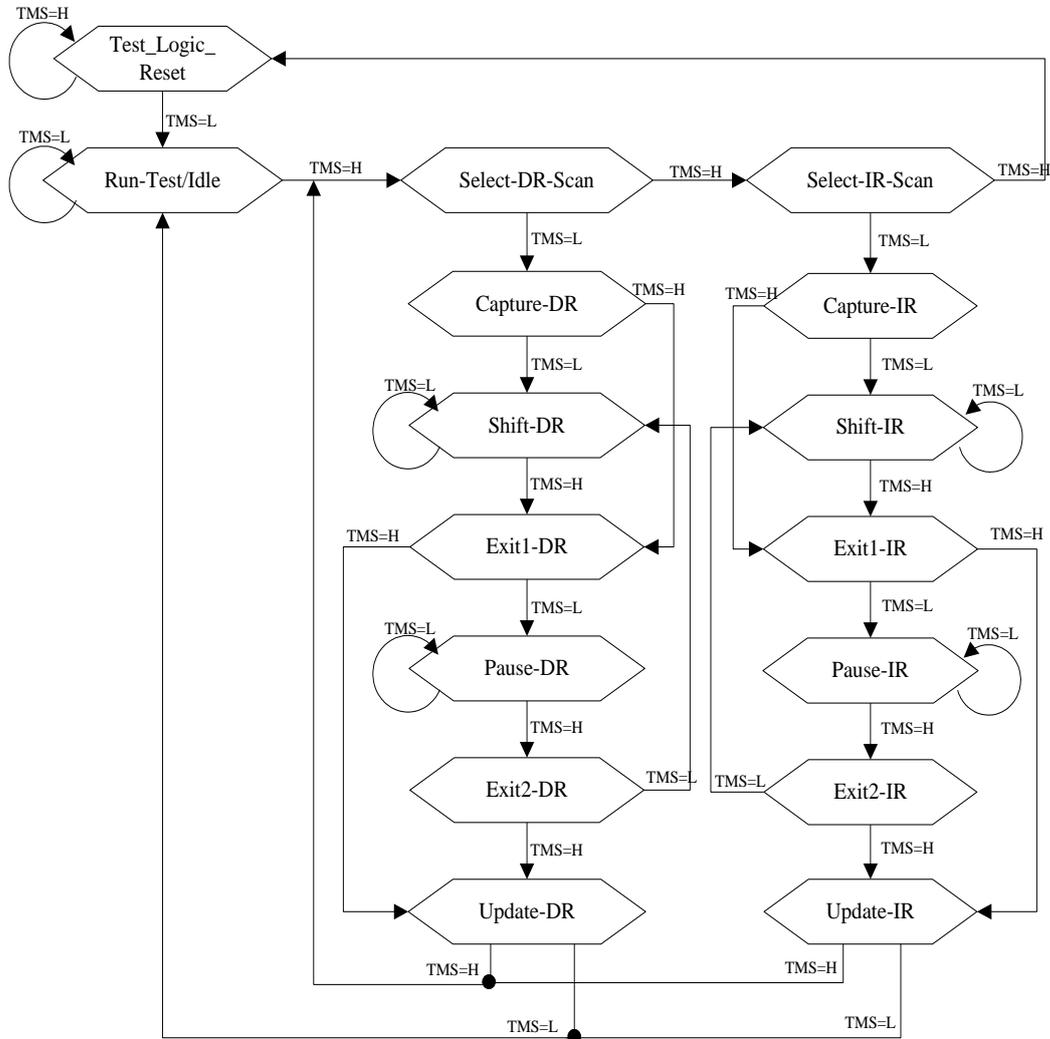


图 8-1: TAP 控制器状态机

8.4 测试功能

测试功能相关的寄存器有：

1) 指令寄存器：测试接口包含一个 8 位的指令寄存器，该寄存器为具有输出锁存和输入锁存功能的移位寄存器。除了支持 IEEE1149.1 标准强制需要的测试指令外，还提供在申威 411 处理器中专用的测试指令，为处理器提供调试与测试支持。主要测试指令见表 8-2 所示。

表 8-2: 指令寄存器定义的主要测试指令

序号	类型	指令名称	指令编码	含义
----	----	------	------	----

			(二进制)	
1	标准 指令	EXTEST	00000000	实现在系统中的互连测试。
2		SAMPLE/PRELOAD	00000001	实现对信号引脚的采样与预加载。
3		IDCODE	00000010	访问标志寄存器。
4		EXTEST_PULSE	00000011	AC boundary Scan 测试指令 1
5		EXTEST_TRAIN	00000100	AC boundary Scan 测试指令 2
6		BYPASS	11111111	设置旁路。
7	专用 测试指令	CHAINADDR	00001010	设置扫描链地址寄存器，选择指定的扫描链
8		RUNSCAN	00001001	开启 DDR3PHY 的 ATPG 测试

2) 旁路寄存器: 1 位, IEEE1149.1 标准中规定的寄存器之一。当 BYPASS 指令有效的时候, 将该寄存器接在 TDI_H 和 TDO_H 之间。在默认情况下, TAP 控制器选择旁路寄存器作为从 TDI_H 到 TDO_H 的串行移位数据通路。

3) 标志寄存器: 32 位, IEEE1149.1 标准中规定了该寄存器的格式, 具体如图 8-2 所示。标志寄存器的最低位总是设置为“1”, [11:1]位制造商代码, [27:12]位为器件类型代码, [31:28]位为同一器件不同的版本号, 具体定义如表 8-3 所示。

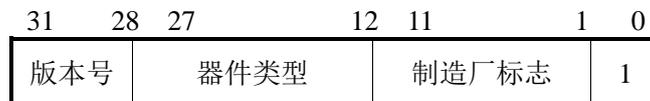


图 8-2: 标志寄存器

表 8-3: 标志寄存器说明

IDCODE[0]	固定为“1”。
IDCODE[11:1]	固定为“000 0000 0000”。
IDCODE[27:12]	固定为“0000 0000 0000 0111”。
IDCODE[31:28]	固定为“0000”。

4) 测试地址寄存器: 12 位, 自定义寄存器, 复位时清“0”, 可通过 CHAINADDR 指令对该寄存器进行设置。测试地址寄存器的具体含义如表 8-4 所示。

表 8-4: 测试地址寄存器说明

名称	范围	类型	描述
ChainID	[8:6]	WO,0x0	只在 ModuleID 和 SubTypeID 表示是 DDR3PHY 的 ATPG 测试时有意义, 表示 ATPG 的链号。 0: AC 的 Positive 链 (211) 1: AC 的 Negative 链 (112) 2: DATX0-2 的 Positive 链 (999)

			3: DATX3-5 的 Positive 链 (999) 4: DATX6-8 的 Positive 链 (999) 5: DATX0-8的Negative链 (783) 其它保留;
SubTypeID	[4:3]	WO,0x0	只在ModuleID为DDR3PHY或MCU时有意义; 当ModuleID为DDR3PHY时: 0: BSR; 1: JTAG; 2: ATPG 其它保留。 当ModuleID为MCU时: 0: 维护命令寄存器; 1: 维护数据寄存器; 2: 状态寄存器; 3: 保留。
ModuleID	[2:0]	WO,0x0	选择测试模块。 0: GPIO; 1: PCI-E0; 2: PCI-E1; 3: DDR3-0; 4: DDR3-1; 5: 维护接口; 6: 保留; 7: 保留。
RSV	其他	—	保留。

申威 411 处理器支持三类接口的边界扫描测试，一类是维护接口和时钟接口（统称为 GPIO 接口），第二类为 PCI-E 接口，第三类为 DDR3 存储器接口，共计 7 条边界扫描链，扫描链地址寄存器用于指定这些链的链号，每条链的链长信息如表 8-5 所示，每条链的具体信息见附录 B。

表 8-5: 边界扫描链长度列表

扫描链类型	链长
GPIO 接口的边界扫描链 (BSR)	27
PCI-E0 接口的边界扫描链 (BSR)	32
PCI-E1 接口的边界扫描链 (BSR)	32

DDR3 存储器接口 0 的边界扫描链 (BSR)	143
DDR3 存储器接口 1 的边界扫描链 (BSR)	143

边界扫描测试的具体流程是：使用 CHAINADDR 命令，设定需要进行扫描测试的边界扫描链；使用边界扫描命令 (SAMPLE/PRELOAD/EXTEST)，根据扫描链地址寄存器的值，选择对应的边界扫描链进行扫出和扫入测试。

特殊说明：申威 411 处理器的 DDR3 的 IO 引脚都是双向引脚，申威 411 根据实际属性默认配置成输出或双向。DDR3 的双向引脚的 BSR 测试不需要借助板级引脚信号控制，只通过 JTAG 进行移位控制。具体流程如下：先移入 (shift)，然后更新 (update)，接下来采样 (capture)，再移出 (shift) 判断正确性。该方法是利用双向引脚环回特性，对于双向引脚移出值与移入值一致，而 OE 控制和单向引脚，其移出值为 0。

8.5 维护功能

通过 JTAG 接口可访问的维护接口寄存器，实现对芯片的维护操作。图 8-2 给出了基于 JTAG 接口的维护控制的结构图，主要包括维护接口寄存器和维护处理逻辑。

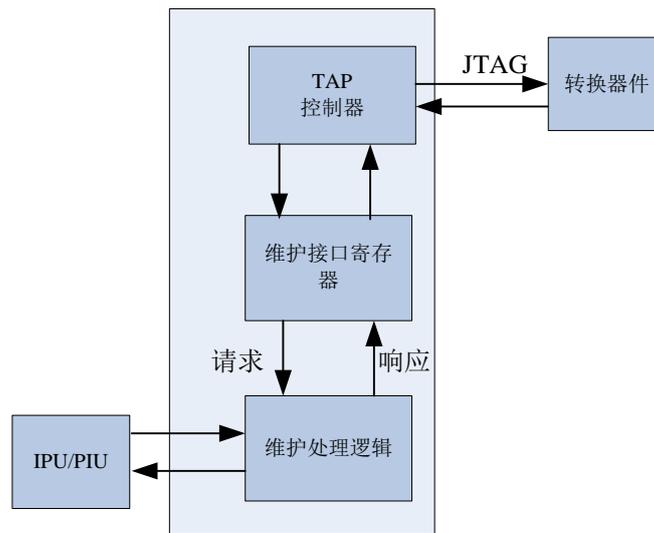


图 8-2: 基于 JTAG 接口的维护控制结构图

维护功能相关的寄存器有：

1) 命令寄存器

为了简化命令扫入时的“左对齐”设计（此时硬件不知道命令长度），维护命令统一按照最大长度（64 位）设置。



图 8-3: 命令寄存器

- a) 串行扫入格式要求: 按照字节 0 (低位到高位) → 字节 1 (低位到高位) → → 字节 7 (低位到高位) 的顺序进行。命令寄存器要求 64 位全部扫入。
- b) 串行扫出格式要求: 按照字节 0 (低位到高位) → 字节 1 (低位到高位) → → 字节 7 (低位到高位) 的顺序进行。命令寄存器要求 64 位全部扫出。

2) 数据寄存器



图 8-4: 数据寄存器 DATA_REG

- a) 串行扫入格式要求: 按照字节 0 (低位到高位) → 字节 1 (低位到高位) → → 字节 127 (低位到高位) 的顺序进行。软件只要扫入足够完整的数据即可 Update, 移入数据进行“左对齐”。
- b) 串行扫出格式要求: 按照字节 0 (低位到高位) → 字节 1 (低位到高位) 的顺序进行。软件只要扫出足够完整的数据即可, 移出数据进行“右对齐”。

3) 状态寄存器

表 8-6: 状态寄存器

描述符	域	意义
RSV	[63:38]	保留。
OBS	[37:33]	观测输出 (仅选择能稳定输出的信号)。 [37]: 核心 PLL 时钟锁定。 [36]: 互连 PLL 时钟锁定。 [35]: 存控 PLL 时钟锁定。 [34]: PCIe-0 接收方链路连接建立。 [33]: PCIe-1 接收方链路连接建立。
FLAG	[32:28]	[32]: 保留; [31:28]: 对应 IOR: FlagReg[3:0]。该寄存器核心可读写, 通过编码表示各类状态, 具体状态信息意义由软件控制。
BIST_INF	[27:24]	[27]: BIST 测试完成, 高电平有效。 [26:25]: 测试结果。

		<p>2'b00, 表示测试无错;</p> <p>2'b01, 表示有错可修复;</p> <p>2'b10, 表示有错不可修复;</p> <p>2'b11, 表示测试时间超时。</p> <p>[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。</p>
RES_ERR	[23]	JTAG 接口维护控制错, 高电平有效。(在收到下一个基于 JTAG 接口的维护请求或“维护接口复位命令”时被清除)。
SYS_ERR	[22]	系统错标志, 高电平有效。
INT_DONE	[21]	中断完成标志, 高电平有效。
RSV	[20]	保留
MFSM	[19:16]	维护主状态机编码。
RES_LNT	[15:8]	<p>响应长度。</p> <p>0: 0 字节;</p> <p>8: 8 字节;</p> <p>128: 128 字节;</p> <p>其它保留。</p>
RES_CMD	[7:2]	响应编码[5:0]。
STATE	[1:0]	<p>记录基于 JTAG 接口的维护请求处理状态。</p> <p>2'b00: 空闲状态。</p> <p>2'b01: 维护处理状态。</p> <p>2'b10: 保留。</p> <p>2'b11: 维护命令处理结束状态。</p> <p>对于维护命令(扫入维护命令寄存器)必须查询该状态标志, 等待在标志为空闲。</p>

9 复位与初始化

申威 411 处理器在加电后，需要进行复位与初始化，才能进入正常工作状态。在复位与初始化期间，需要对申威 411 处理器进行配置，以满足系统的需要。

9.1 复位

9.1.1 复位类型

申威 411 处理器有三种复位方式，即上电复位、冷复位和睡眠唤醒，前两种复位由引脚信号启动，是对整个处理器的复位；第三种睡眠唤醒可以由维护接口的维护命令启动，也可以由内部的核心启动。

- 1) 上电复位：由引脚 `DCOK_H` 为低电平引起的复位，为优先级最高的复位。在申威 411 处理器加电时应维持 `DCOK_H` 为低电平，使得处理器处于上电复位状态，待提供给处理器的各种电源稳定、输入给处理器的各种时钟稳定，才能将 `DCOK_H` 信号变为高电平，结束上电复位。上电复位是非常彻底的复位，包括复位处理器内部的 PLL 和维护接口；
- 2) 冷复位：由引脚 `RESET_L` 为低电平来引起的复位，优先级低于上电复位（即冷复位时 `DCOK_H` 应有效）。冷复位与上电复位功能相同；
- 3) 睡眠唤醒：申威 411 处理器的核心可以处于睡眠状态，此状态下的核心处于复位状态，且工作频率非常低（维护时钟频率的八分之一），因此睡眠的核心功耗很低。通过核心运行程序或者维护接口的维护命令，可以使得任意的核心处于睡眠状态。同样通过核心运行程序或维护接口的维护命令，可以唤醒处于睡眠状态的核心。核心唤醒后，首先将核心工作时钟频率恢复到正常状态，然后结束复位，恢复正常工作状态。

9.1.2 复位状态机

申威 411 处理器通过复位状态机来控制各种复位的处理，外部系统通过引脚 `STAT_OUT[3:0]_L` 可以直接了解复位状态机的状态。复位状态机的状态转换如图 9-1 所示。

- 1) 上电复位(4'b0000)：若引脚 `DOCK_H` 为“0”，则无条件处于此状态。此状态下，若 `DCOK_H` 变为“1”，则转向冷复位状态；

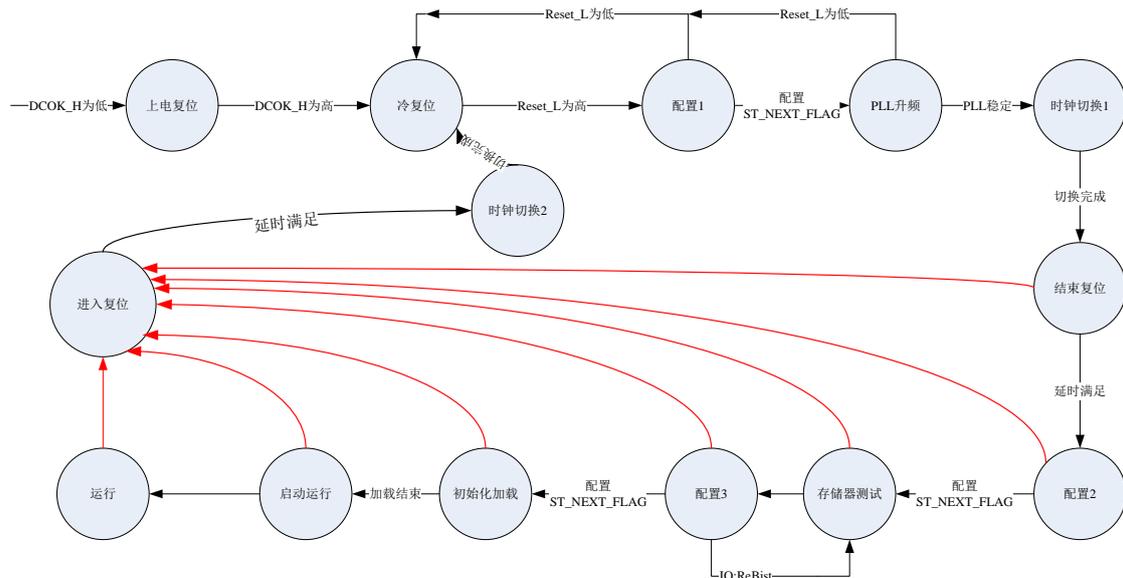


图 9-1: 复位状态机状态转换图

- 2) 冷复位 (4'b0001): 若引脚 **RESET_L** 为“0”，则维持在此状态，否则转向配置 1 状态；
- 3) 配置 1 (4'b0010): 如果引脚 **CFG_SEL_L** 为“1”，外部可使用维护命令，通过自定义串行维护接口，修改维护控制部件内部的 I/O 寄存器，外部向申威 411 处理器发出写“**ST_NEXT_FLAG**”寄存器的维护命令，则转向 PLL 升频状态。若 **CFG_SEL_L** 为“0”，则根据配置接口引脚设置参数配置寄存器内容，然后转向 PLL 升频状态；
- 4) PLL 升频 (4'b0011): 如果参数配置寄存器设置成 PLL 旁路模式，则直接转入时钟切换状态；否则通过计数器计数方式等待 PLL 升频稳定，计数器溢出再转入时钟切换 1 状态（由于存储器测试状态增加 Page Cache 测试，而 Page Cache 的 Clock 来自 PCI-E PHY，所以此处需要等待 PCI-E PHY 的 Clock 稳定。硬件实现时在维护接口设置计数器，同时监测来自 PCI-E PHY 的 Clock Stable 信号）；
- 5) 时钟切换 1 (4'b0100): 按顺序进行工作时钟的高低频切换。按照“系统接口→存控→互连→核心 0→核心 1”的顺序进行正常工作时钟切换（被切割的核心维持低频工作时钟状态）。完成频率切换后，转向结束复位状态；
- 6) 结束复位 (4'b0101): 结束申威 411 的复位，等待一段时间后，转向配置 2 状态；
- 7) 配置 2 (4'b0110): 如果参数配置寄存器设置成跳过此状态，则直接转向存储器自测试状态，否则外部可使用维护命令，通过自定义串行维护接口，设置与存储器自测试相关的扫描链寄存器，外部向申威 411 处理器发出写“**ST_NEXT_FLAG**”寄存器的维护命令，则转向存储器自测试状态；
- 8) 存储器自测试 (4'b0111): 如果参数配置寄存器设置成跳过此状态，则直接转向配置 3 状态，否则对内部存储器进行自测试，存储器自测试结束转向配置 3 状态；如果等待指定时间尚未收到存储器自测试结束标志，则作为超时而转向配置 3 状态；
- 9) 配置 3 (4'b1000): 如果参数配置寄存器设置成跳过此状态，则直接转向初始化加载状态，

否则外部可使用维护命令，通过自定义串行维护接口，设置 DDR3 存储器接口以及 PCI-E 接口等相关 I/O 寄存器，外部向处理器发出写“ST_NEXT_FLAG”寄存器的维护命令，则转向初始化加载状态，外部向处理器发出“重新存储器自测试”维护命令，则转向存储器自测试状态；

- 10) 初始化加载 (4'b1001): 如果参数配置寄存器设置成跳过此状态，则直接转向启动运行状态，否则外部可使用维护命令，通过自定义串行维护接口，将初始化程序加载到核心的指令 Cache 中，外部向处理器发出“初始化加载结束”的维护命令，则转向启动运行状态；
- 11) 启动运行 (4'b1010): 如果参数配置寄存器设置成不运行状态，则停留在该状态，否则向各个核心发出启动运行信号，并转向运行状态；
- 12) 运行 (4'b1011): 此状态下，启动各核心开始运行（被切割的核心维持低频工作时钟下的复位状态）；
- 13) 进入复位 (4'b110x): 在结束复位状态、配置 2 状态、存储器自测试状态、配置 3 状态、初始化加载状态、启动运行状态和运行状态下，若引脚 Reset_L 变为“0”，则转向此状态。在该状态下，使得处理器除维护控制部件之外的所有部件进入复位状态，等待一段时间好后，转向时钟切换 2 状态；
- 14) 时钟切换 2 (4'b111x): 按顺序进行各模块的高低频工作时钟切换，即按照“系统接口 → 存控 → 互连 → 核心 0 → 核心 1”的顺序切换为低频工作时钟，然后转向冷复位状态。

注：配置 1 状态下外部只能读写维护控制部件的 I/O 寄存器，配置 2 状态和配置 3 状态下可以修改任意的 I/O 寄存器，但配置 2 状态主要用于修改与存储器自测试相关的扫描链寄存器。

9.2 配置

9.2.1 引脚配置

申威 411 处理器的时钟接口和维护接口的部分引脚用于配置申威 411 处理器，这些引脚需要在引脚 DCOK_H 有效之前设定相应的值，在 DCOK_H 有效后，不得改变，否则会产生不可预测的结果。这些配置引脚情况如表 9-1 所示。

表 9-1: 与配置相关的引脚信号

配置引脚	配置要求
INIT_MODE_H[1:0]	应设置值为“11”，其它设置值用于处理器的测试。
CFG_CORE_H[3:0]	根据表 4-2 设置核心工作频率，一般应设置为非全“0”值（全“0”值为 PLL 旁路方式，用于处理器的测试）。

CFG_MM_H[2:0]	根据表 4-3 设置存储控制器工作频率, 一般应设置为非全“0”值 (全“0”值为 PLL 旁路方式, 用于处理器的测试)
CFG_XBX_H[2:0]	根据表 4-4 设置交叉开关工作频率, 一般应设置为非全“0”值 (全“0”值为 PLL 旁路方式, 用于处理器的测试)。
CFG_SEL_L	配置方式选择。 “0”: 采用引脚配置方式, 即采用上述 12 根信号来配置申威 411; “1”: 采用维护配置方式, 即上述 12 根信号无效, 采用缺省寄存器配置, 并可以通过维护命令对默认值进行修改。

9.2.2 I/O 寄存器配置

申威 411 处理器在结束上电复位和冷复位后, 进入初始化。在初始化过程中, 外部系统需要通过维护接口的维护命令, 修改处理器内部与配置相关的 I/O 寄存器, 以进一步配置处理器, 满足系统的需求。与此相关的 I/O 寄存器如表 9-2 所示。

表 9-2: 配置相关的 I/O 寄存器

复位状态机状态	I/O 寄存器名称	含义
配置 1 状态	核心在位使能寄存器 CORE_ONLINE	用于确定处理器的 4 个核心中哪些核心在位, CORE_ONLINE[i]为“0”, 表示该核心为断开状态, 断开的核心始终处于复位和低功耗状态。
	存储在位使能寄存器 MC_ONLINE	用于确定处理器的 2 个存储器控制器中哪个在位, MC_ONLINE[i]为“0”, 表示该存控被隔离, 对该存控的 I/O 寄存器访问以及对该存控连接的存储器访问都将产生非法地址响应。
	I/O 接口使能寄存器 IO_START	用于处理器内部的 PCI-E 接口的故障隔离和容错, IO_START[i]为“0”, 表示 PCI-E 接口 0 或 1 被隔离, 对该接口的 I/O 访问将产生非法地址响应。
	时钟切换间隔寄存器 PLL_CHG_CNT	用于在处理器复位结束和复位开始时, 控制不同核心、存控、和系统接口的时钟在低频和高频工作时钟间切换的时间间隔, 该间隔默认为 65536 个维护时钟周期。

	初始化控制寄存器 INIT_CTL	用于控制处理器的复位初始化流程（时钟配置以及状态机状态控制）。
	BIST 控制寄存器 BIST_CTL	用于控制处理器的复位初始化流程的存储器自测试。
配置 2,3 状态	存储器接口控制寄存器 CfgCR	用于控制处理器存储器接口的配置。
	存储控制器接口控制寄存器 MC_CTRL	用于控制处理器内部的存储器控制器的配置。
	存储控制器接口初始化控制寄存器 INIT_CTL	用于控制处理器内部的存储器控制器初始化过程。
初始化加载	存储器自测试状态寄存器： BIST_STAT	用于存放处理器内部的每个存储器阵列进行存储器自测试后的总结果。
	存储器自测试结果寄存器： CG_BIST_STAT	用于存放处理器内部的每个存储器阵列进行存储器自测试后的详细信息。
	Page Cache 存储器自测试结果寄存器：PC_BIST_STAT	用于存放处理器 Page Cache 的每个存储器阵列进行存储器自测试后的详细信息。

9.3 初始化

9.3.1 加关电顺序

为保证申威 411 处理器的正常工作，需要按照一定顺序完成对申威 411 处理器的加电，具体如下：

- 1) 将处理器的引脚信号 DCOK_H、RESET_L 设置为低电平；引脚信号 STAT_OUT_SEL[1:0]_L 设置为 2'b10；
- 2) 对处理器的 I/O 电源进行加电；
- 3) 对处理器的内核电源进行加电；
- 4) 将处理器的引脚信号 DCOK_H 变为高电平的有效状态；
- 5) 将处理器的引脚信号 RESET_L 变为高电平的无效状态；
- 6) 处理器进入初始化流程。

申威 411 处理器关电也必须维持一定的顺序，必须按“RESET_L 有效→DCOK_H 无效→关闭内

核电源→关闭 I/O 电源”的顺序进行。

9.3.2 初始化流程

申威 411 处理器初始化流程具体如下：

1) 上电复位状态 (STAT_OUT[3:0]_L=4'b0000)

该状态下的具体操作包括：

- a) 设置引脚 INIT_MODE_H[1:0]为 2'b11；
- b) 设置引脚 DCOK_H=1'b0、RESET_L=1'b0、STAT_OUT_SEL[1:0]_L=2'b10 (默认值，选择主状态机状态)，表示上电复位，且状态输出选择主状态机状态；
- c) 配置芯片时钟频率：设置引脚 CFG_CORE[3:0]_H、CFG_MM[2:0]_H、CFG_XBX[2:0]_H；
- d) 配置模式选择：设置引脚 CFG_SEL_L。

2) 等待冷复位结束状态 (STAT_OUT[3:0]_L=4'b0001)

该状态下的具体操作包括：

- a) 设置引脚 RESET_L: 0→1，冷复位结束，进入配置状态 1，STAT_OUT[3:0]_L: 4'b0001→4'b0010。

3) 配置 1 状态 (STAT_OUT[3:0]_L=4'b0010)

该状态下的具体操作包括 (可配置为直接跳过该状态)：

- a) 根据需要设置 IOR: MC_ONLINE、CORE_ONLINE 和 IO_START；
- b) 设置 IOR: INIT_CTL，根据需要配置时钟配置以及状态机配置。
- c) 设置 IOR: BIST_CTL，根据需要配置存储自测试类型。
- d) 设置 IOR: NEXT_STAGE，进入 PLL 升频状态，STAT_OUT[3:0]_L: 4'b0010→4'b0011。

4) PLL 升频状态 (STAT_OUT[3:0]_L=4'b0011)

当 PLL 处于稳定工作状态，进入时钟切换 1 状态，STAT_OUT[3:0]_L: 4'b0011→4'b0100；

5) 时钟切换 1 状态 (STAT_OUT[3:0]_L=4'b0100)

芯片自动执行，所有时钟升频都切换完成后，进入结束复位状态，STAT_OUT[3:0]_L: 4'b0100→4'b0101；

6) 结束复位状态 (STAT_OUT[3:0]_L=4'b0101)

暂时状态，自动进入结束复位状态，STAT_OUT[3:0]_L: 4'b0101→4'b0110；

7) 配置 2 状态 (STAT_OUT[3:0]_L=4'b0110)

该状态下的具体操作包括 (可配置为直接跳过该状态)：

- a) 根据需要对 BISR 修复信息进行扫入操作；
- b) 设置 IOR: NEXT_STAGE，进入 PLL 升频状态，STAT_OUT[3:0]_L: 4'b0110→4'b0111。

8) 存储器自测试状态 (STAT_OUT[3:0]_L=4'b0111)

根据 IOR: BIST_CTL, 进行存储器自测试, 自测试结果记录在 IOR:BIST_STAT、CG_BIST_STAT 和 PC_BIST_STAT 寄存器中。存储器自测试完成后, 进入配置 3 状态, STAT_OUT[3:0]_L: 4'b0111→4'b1000;

9) 配置 3 状态 (STAT_OUT[3:0]_L=4'b1000)

配置必要的 I/O 寄存器。可以在该状态进行存储器的初始化, 也可以利用 SROM 加载进行存储器的初始化。完成配置后, 通过写 IOR: ST_NEXT_FLAG, 进入 SROM 加载状态, STAT_OUT[3:0]_L: 4'b1000→4'b1001。

10) SROM 加载状态 (STAT_OUT[3:0]_L=4'b1001)

该状态下的具体操作如下:

- a) 维护读 IOR: BIST_STAT、CG_BIST_STAT 和 PC_BIST_STAT, 判断存储器测试是否成功, 如果成功跳转到 b), 如果某个核心的存储器阵列自测试失败, 则要设置 IOR: CORE_ONLINE 以关闭该核心; 然后设置 RESET_L: 1→0, 进入等待冷复位状态, STAT_OUT[3:0]_L: 4'b1001→4'b1100→4'b0001;
- b) 根据 IOR: INIT_CTL[ICLD_EN], 决定是否将 SROM 程序加载到 Icache 中, 如果执行加载, 则每个维护加载命令加载一个 Cache 行数据;
- c) SROM 程序的所有数据加载完后, 系统发送一个维护加载结束的维护命令, 状态机进入预备运行状态, STAT_OUT[3:0]_L: 4'b1001→4'b1010,

11) 预备运行状态 (STAT_OUT[3:0]_L=4'b1010)

根据 IOR: INIT_CTL[STRUN_EN]决定是否进入运行状态, 如果不进入则在该状态保持, 如果进入则 STAT_OUT[3:0]_L: 4'b1010→4'b1011,

12) 运行状态 (STAT_OUT[3:0]_L=4'b1011)

此时芯片正常启动执行, 可以运行加载进指令 Cache 的 SROM 程序或者从主存取值令, 对申威 411 处理器进行进一步的测试和控制, 然后开始进行操作系统引导, 使得申威 411 处理器进入正常工作状态。

9.3.3 初始化程序

初始化程序包含对申威 411 处理器进行进一步的初始化, 设置相关的控制寄存器, 进行进一步的测试, 为后续启动操作系统建立运行环境。初始化程序正确运行完成后, 通知外部系统可以向申威 411 处理器控制的存储器中加载操作系统, 外部系统完成操作系统加载以后, 通知申威 411 处理器启动加载的操作系统。

10 电气特性

10.1 工作条件

申威 411 的工作条件如表 10-1 所示。

表 10-1: 工作环境参数

符号	参数	最小值	正常值	最大值	单位
$V_{VDD} \pm 40mV$	内核电源电压	0.91	0.95	0.99	V
$V_{VDD15} \pm 5\%$	1.5V I/O 电源电压 (支持 1.35V 电压)	1.425	1.5	1.575	V
$V_{VDD135} \pm 5\%$	1.35V I/O 电源电压	1.283	1.35	1.45	V
$V_{VDD18} \pm 5\%$	1.8V I/O 电源电压	1.71	1.8	1.89	V
V_{PAD}	信号引脚电压	-0.3	-	$V_{VDDQ} + 0.09$	V
V_{REF15}	1.5V I/O 参考电压	$0.49 * V_{VDD15}$	$0.5 * V_{VDD15}$	$0.51 * V_{VDD15}$	V
T_J	结温	0	70	125	°C
T_A	环境温度	-40	25	85	°C
ESD 静电保护电压	Human Body Model	2000			V
	Machine Model	200			V
	Charge Device Model	500			V

注: V_{VDDQ} 为信号引脚对应的 I/O 电源电压, 对 SSTL 信号为 V_{VDD15} , 其它信号为 V_{VDD18} 。

10.2 直流参数

10.2.1 差分时钟输入信号

差分时钟输入信号 RCLK_P/RCLK_N 为 LVDS 电平, 其直流参数如表 10-2 所示。

表 10-2: 差分时钟的直流参数

参数	描述	最小值	正常值	最大值	单位
V_{INPUT}	输入电压范围	0	0.9	1.8	V
$ V_{IDTH} $	输入差分门槛电压	50			mV
$ V_{ID} $	输入差分电压	50		1800	mV

V_{IA}, V_{IB}	输入电压	0	0.9	1.8	V
$ I_{IA} , I_{IB} $	输入漏电流		0.01	5	μA

10.2.2 存储器接口信号

存储器接口信号都是电源为 1.5V 或 1.35V 的 SSTL 电平，其直流参数如表 10-3 所示。

表 10-3: 存储器接口信号直流参数

符号	参数	最小值	正常值	最大值	单位	说明
V_{IH}	输入高电平电压	$V_{REF15}+0.1$		V_{VDD15}	V	
V_{IL}	输入低电平电压	$V_{SS}-0.3$		$V_{REF15}-0.1$	V	
V_{IHT}	输入高电平阈值电压			$V_{REF15}+0.025$	V	
V_{ILT}	输入低电平阈值电压	$V_{REF15}-0.025$			V	
V_{OH}	输出高电平电压	$0.8*V_{VDD15}$			V	1
V_{OL}	输出低电平电压			$0.2*V_{VDD15}$	V	1
t_{STEP}	输出调整一级延迟大小		5		ps	
R_{TT}	RTT 有效阻抗值	-12%	40/60/120	+31%	欧姆	2
R_{serdrv}	串行输出电阻	-10%	34	+10%	欧姆	
P_{RCV0}	输入模式下直流功耗 (V_{VDD15})	0.78	1.32	2.58	mW	3
P_{RCV1}	输入模式下直流功耗 (V_{VDD})	0.90	3.00	25.15	μW	3
P_{DRV0}	输出模式下直流功耗 (V_{VDD15})	0.81	1.33	2.49	mW	4
P_{DRV1}	输出模式下直流功耗 (V_{VDD})	0.90	3.00	25.15	μW	4

说明:

- 1) 表中的直流参数是没有端接电阻的情况下负载为 1pf 的值;
- 2) 端接电阻可选择为 40 欧姆、60 欧姆或 120 欧姆;
- 3) 输入没有端接 (ODT 不使能), 总功耗需将直流功耗和交流功耗相加;
- 4) 输出没有端接, 总功耗需将直流功耗和交流功耗相加。

10.2.3 1.8V LVCMOS 信号

时钟接口 (除差分参考时钟输入)、维护接口和测试接口信号都是采用 1.8V LVCMOS 电平。这类信号的直流参数如表 10-4 所示:

表 10-4: 1.8V LVCMOS 电平信号的直流参数

参数	描述	最小值	正常值	最大值	单位
V _{DD}	前驱电压 (核心电压)	0.91	0.95	0.99	V
V _{DDPST}	后驱电压 (IO 电压)	1.71	1.8	1.89	V
V _{IMAX}	最大输入电压			2.75	V
V _{IH}	输入高电平	1.17		2.75	V
V _{IL}	输入低电平	-0.3		0.63	V
V _T	阈值点	0.78	0.84	0.89	V
V _{TPU}	带上拉电阻使能的阈值点	0.77	0.83	0.88	V
V _{TPD}	带下拉电阻使能的阈值点	0.8	0.85	0.9	V
R _{PU}	上拉电阻	38K	57K	92K	欧
R _{PD}	下拉电阻	37K	53K	99K	欧
V _{OH}	输出高电平	1.35			V
V _{OL}	输出低电平			0.45	V
I _{OH}	高电平输出电流 (V _{OH} =1.35V)	9.0	20.2	36.5	mA
I _{OL}	低电平输出电流 (V _{OL} =0.45V)	10.4	17.9	26.5	mA

10.3 交流参数

10.3.1 差分时钟输入信号

差分时钟输入信号 RCLK_P/RCLK_N 为 LVDS 电平, 其交流参数如表 10-5 所示。

表 10-5: 差分时钟的交流参数

符号	参数	最小值	典型值	最大值	单位	说明
t _{dc}	输入时钟占空比	40	—	60	%	
J _{max}	输入时钟 Jitter	—	—	2	%	
F _{req}	输入时钟频率	100	200	300	MHz	

10.3.2 存储器接口信号

存储器接口信号为 1.5V 或 1.35V SSTL 电平, 其交流参数和延时信息如表 10-6 和表 10-7 所示。表 10-7 中的延时信息是针对 72 位数据和 9 位 DQS 而言。存储器接口信号在 PCB 板级的长度要求 (推荐值) 见表 10-8, 该表中的 Clock、Data、Control 和 Command 的具体分类见表 10-9。

表 10-6: DDR3 存储器接口交流参数

符号	参数	最小值	正常	最大值	单位	说
----	----	-----	----	-----	----	---

			值			明
V_{IH} (AC)	交流输入逻辑高电压阈值	$V_{REF15}+175mV$			V	
V_{IL} (AC)	交流输入逻辑低电压阈值			$V_{REF15}-175mV$	V	
F_{MAX}	最大工作频率			1066	MHz	
D_{MAX}	最大数据传输率			2133	Mbps	
P_{RCV0}	输入模式下交流功耗 (V_{VDD15})	0.76	0.82	0.88	$\mu W/MHz$	1,2
P_{RCV1}	输入模式下交流功耗 (V_{VDD})	0.06	0.08	0.10	$\mu W/MHz$	1,2
P_{DRV0}	输出模式下交流功耗 (V_{VDD15})	7.74	8.70	10.05	$\mu W/MHz$	2,3
P_{DRV1}	输出模式下交流功耗 (V_{VDD})	0.36	0.45	0.73	$\mu W/MHz$	2,3

说明:

- 1) 输入没有端接 (ODT 不使能);
- 2) 总功耗需将直流功耗和交流功耗相加;
- 3) 假设负载为 5pf, 并有 25 欧姆端接电阻, 端接电压为 V_{VDD15} 的二分之一。

表 10-7: DDR3 存储器接口延时信息

符号	参数	边沿	最小值	最大值	偏斜	单位	说明
t_{pd-CLK}	时钟信号输出延时	上升沿	1.324	1.325	0.001	ns	1
		下降沿	1.348	1.348	0.000	ns	1
t_{pd-AC}	地址命令信号输出延时	上升沿	1.344	1.375	0.031	ns	1
		下降沿	1.314	1.343	0.029	ns	1
$t_{pd-DQS-out}$	数据选通信号输出延时	上升沿	1.356	1.362	0.006	ns	1
		下降沿	1.371	1.377	0.006	ns	1
$t_{pd-DQ-out}$	数据信号输出延时	上升沿	1.381	1.417	0.036	ns	1
		下降沿	1.396	1.432	0.036	ns	1
$t_{pd-DQS-in}$	数据选通信号输入延时	上升沿	0.353	0.354	0.001	ns	2
		下降沿	0.353	0.354	0.001	ns	2
$t_{pd-DQ-in}$	数据信号输入延时	上升沿	0.307	0.311	0.004	ns	2
		下降沿	0.309	0.313	0.004	ns	2

说明:

- 1) 输出延时都是内部触发器到芯片引脚。

- 2) 输入延时都是芯片引脚到内部触发器。

表 10-8: DDR3 存储器接口推荐长度

组	最短长度	最长长度
Control 相对与 Clock	Clock-12.7mm	Clock+12.7mm
Command 相对与 Clock	Clock-12.7mm	Clock+12.7mm
DQS 相对于 Clock	Clock-12.7mm	Clock+12.7mm
DQ 相对于 DQS	DQS-0.25mm	DQS+0.25mm

说明:

- 1) 长度是指从芯片引脚到 DIMM 的连接引脚的总长度。

表 10-9: DDR3 存储器接口分类

组	信号	描述
Clock	MM[3:0]L_CK[1:0]_P	差分输入时钟。
	MM[3:0]L_CK[1:0]_L	
Data	MM[3:0]L_DQ[63:0]_H	数据位。
	MM[3:0]L_ECC[7:0]_H	ECC 位。
	MM[3:0]L_DQS[8:0]_P	数据选通信号 (差分)。
	MM[3:0]L_DQS[8:0]_L	
Control	MM[3:0]L_CKE[1:0]_H	对应每个 Rank 的时钟使能信号。
	MM[3:0]L_CS[1:0]_L	对应每个 Rank 的内部端接使能信号。
	MM[3:0]L_ODT[1:0]_H	对应每个 Rank 的片选信号。
Command	MM[3:0]L_RAS_L	行地址 RAS#命令。
	MM[3:0]L_CAS_L	列地址 CAS#命令。
	MM[3:0]L_WE_L	写 WE#命令。
	MM[3:0]L_BA[2:0]_H	存储器体 (Bank) 地址。
	MM[3:0]L_A[15:0]_H	存储器地址。

10.3.3 维护接口信号

维护接口信号都是电源为 1.8V 的 LVC MOS 电平, 其交流参数如表 10-10 所示, 相应的延时见图 10-1。图中的延迟 1 和延迟 2 分别对应输入引脚 MT_CLK_H 和 MT_RX_H 到内部触发器的延迟,

延迟 3 是对应输出引脚 MT_TX_H 从内部触发器到引脚的延迟，延迟 4 是输出引脚 MT_TX_H 相对于 MT_CLK_H 引脚上升沿的间隔（1/2 个 MT_CLK_H 周期-延迟 1-延迟 3）。输出引脚 STAT_OUT[4:0]_H 与输出引脚 MT_TX_H 类似。

表 10-10: 维护接口的交流参数

引脚名称	I/O	频率范围 (MHz)	传输延迟 (ns)	说明
MT_CLK_H	输入	5~25	3	
MT_TX_H	输出	同步于 MT_CLK_H	5	MT_CLK_H 时钟下降沿输出。
MT_RX_H	输入	同步于 MT_CLK_H	3	MT_CLK_H 时钟上升沿采样。
WAKEUP_L	输入	同步于 MT_CLK_H	3	MT_CLK_H 时钟上升沿输出。
NMI_L	输入	同步于 MT_CLK_H	3	MT_CLK_H 时钟上升沿输出。
RESET_L	输入	电平信号	—	有效宽度应大于 40ms。
DCOK_H	输入	电平信号	—	为低电平时的宽度应大于 40ms。
INIT_MODE[1:0]_H	输入	电平信号	—	应在上电复位之前设置且保持不变。
STAT_OUT[4:0]_H	输出	同步于 MT_CLK_H	5	MT_CLK_H 时钟下降沿输出。
STAT_OUT_SEL_L[1:0]	输入	电平信号	—	有效宽度应大于 40ms。
CFG_SEL_L	输入	电平信号	—	应在上电复位之前设置且保持不变。

注 1: 输入延时为从引脚输入到内部触发器的延时，输出延时是内部触发器到输出引脚延时。

注 2: STAT_OUT[4:0]_L 根据选择信号输出不同的内容，在输出申威 411 处理器状态时是同步于 MT_CLK_H，且下降沿输出。

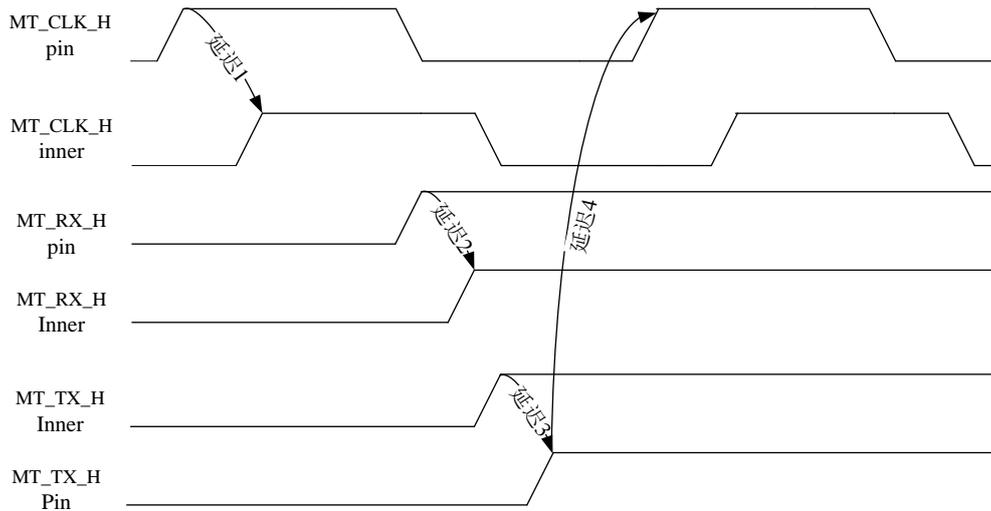


图 10-1: 维护接口延时图

10.3.4 测试接口信号

测试接口信号都是电源为 1.8V 的 LVCMOS 电平，其交流参数如表 10-11 所示。

表 10-11: 测试接口的交流参数

引脚名称	I/O	频率范围 (MHz)	传输延时 (ns)	说明
TCK_H	输入	5~10	3	输入延时。
TMS_H	输入	同步于 TCK_H	3	输入延时，TCK_H 时钟上升沿采样。
TRST_L	输入	电平信号	—	有效宽度应大于 200us。
TDI_H	输入	同步于 TCK_H	3	输入延时，TCK_H 时钟上升沿采样。
TDO_H	输出	同步于 TCK_H	10	输出延时，TCK_H 时钟下降沿输出。

注：输入延时为从引脚输入到内部触发器的延时，输出延时是内部触发器到输出引脚延时。

测试接口的延时如图 10-2。图中的延迟 1、延迟 2 和延迟 3 分别对应输入引脚 TCK_H、TMS_H 和 TDI_H 到内部触发器的延迟，延迟 4 是输出引脚 TDO_H 从内部触发器到引脚的延迟，延迟 5 是输出引脚 TDO_H 相对于 TCK_H 引脚上升沿的间隔（1/2 个 TCK_H 周期-延迟 1-延迟 4）。

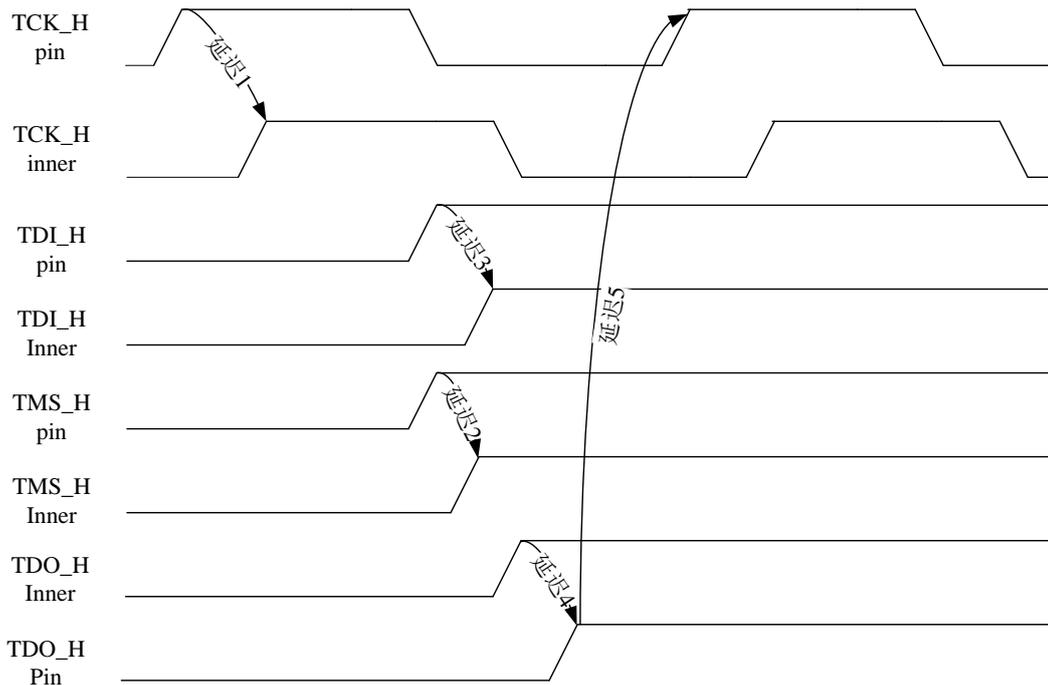


图 10-2: 测试接口延时图

10.3.5 PCI-E 接口

申威 411 处理器的 PCI-E 接口符合 PCI-E 2.0 标准,其接口信号的交直流参数可参考《PCI Express base specification Version2.0》。PCI-E 接口差分时钟输入 PCI_CLK_P/PCI_CLK_N 在系统中建议的连接方式如图 10-3 所示。相关特性与要求如表 10-12 所示。

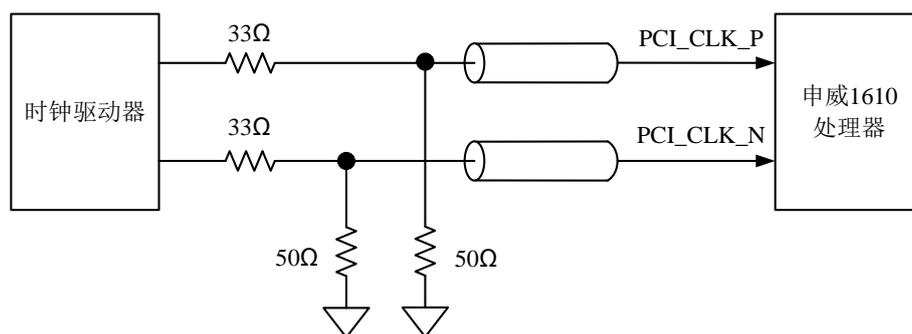


图 10-3: PCI-E 接口差分时钟输入连接参考电路

表 10-12: PCI-E 接口差分时钟输入特性要求

参数	最小值	典型值	最大值	单位
频率		100		MHz
占空比	40		60	%
直流输入电压	0		1.8	V

交流输入电压	0.150		1.44	V
差分输入摆幅	0.300		0.900	V
输入边沿速度	0.6			V/ns

10.4 功耗

10.4.1 封装热特性

在无额外散热器的条件下，申威 411 处理器封装的散热特性如表 10-13 所示。

表 10-13: 申威 411 处理器封装的热阻参数

风速 V(m/s)	θ_{ja} (°C/W)	θ_{jb} (°C/W)	θ_{jc} (°C/W)
0.0	2.92	2.80	0.10
1.0	0.82		
2.0	0.48		

说明:

θ_{ja} : 芯片内硅片到芯片工作环境之间的热传导系数，与封装以及环境温度和冷却风速相关；

θ_{jb} : 芯片内硅片到芯片基板之间的热传导系数；

θ_{jc} : 芯片内硅片到芯片封装管壳之间的热传导系数；

芯片结温=环境温度+ θ_{ja} *功耗=基板温度+ θ_{jb} *芯片壳温+ θ_{jc} *功耗。

10.4.2 热设计功耗

建议的申威 411 处理器热设计功耗如下:

- 1) 1.6GHz 核心工作频率下，热设计功耗 (TDP) 为 35W，典型运行功耗 20~30W；
- 2) 1.4GHz 核心工作频率下，热设计功耗 (TDP) 为 30W，典型运行功耗 20~25W；
- 3) 1.2GHz 核心工作频率下，热设计功耗 (TDP) 为 25W，典型运行功耗 15~20W。

10.4.3 功耗管理

申威 411 处理器支持多种功耗管理策略，可在操作系统配合下实现多种低功耗工作模式，具体如下:

- 1) 深睡眠: 申威 411 处理器内含 4 个完全相同的核心，若某些核心处于无运行负载的状态，应将这些核心设置在睡眠状态。核心或者外部系统通过维护命令可向任意核心发出睡眠中断，使得这些无负载的核心处于睡眠状态。处于睡眠状态的核心处于复位状态，且维持较

低的时钟频率（仅为维护时钟频率的八分之一）。一旦需要睡眠的核心恢复工作状态，可由非睡眠的核心或外部系统向睡眠核心发送唤醒中断，即可使得睡眠的核心恢复正常工作频率，结束复位，重新进行初始化，并引导操作系统，恢复正常工作状态。睡眠状态的核心几乎没有动态功耗。

- 2) 浅睡眠：申威 411 处理器支持一种浅睡眠功能，一旦核心无可运行的进程，处于“空闲”状态，可执行“停机”指令，使得核心处于浅睡眠状态。此状态下核心内部的各流水线处于停顿状态，因此只有少量的时钟动态功耗。对浅睡眠的核心发送任何中断，都可以快速使其恢复工作状态。
- 3) 关闭部件：申威 411 处理器核心中的浮点部件和 SIMD 部件可以单独关闭，当核心运行的程序不需要浮点或 SIMD 功能时，可关闭浮点部件或者 SIMD 部件，以节省运行功耗。浮点部件和 SIMD 部件可以动态打开或关闭。
- 4) 降低指令处理速度：申威 411 处理器核心的指令流水线流水速度可以动态调节，一旦核心需要的性能较低时，可以在满足系统性能需求基础上，降低指令流水线流水速度，从而降低核心的动态运行功耗。
- 5) 存储控制器和 PCI-E 接口控制器也可控制进入低功耗状态或者直接关闭。

附录 A 维护接口的命令与响应

A.1 维护命令

A.1.1 维护命令包格式

维护命令包的格式如表 A-1。通过引脚信号 MT_RX_H 串行传输时，按照先低字节后高字节的顺序传输，传输每个字节时，从高位开始传输。

表 A-1: 维护命令包的格式

字节 0	11	命令编码[5:0]
字节 1		地址[7:0]
字节 2		地址[15:8]
字节 3		地址[23:16]
字节 4		地址[31:24]
字节 5		地址[39:32]
字节 6		数据有效位[7:0]
字节 7		数据有效位[15:8]
字节 8		数据字节 0
	
字节 7+n		数据字节 n-1

A.1.2 维护命令编码

申威 411 支持 14 种维护命令，具体的维护命令和含义如表 A-2 所示。

表 A-2: 维护命令编码及其含义

编号	维护命令	命令编码	含义
1	Cache 一致性的存储器读	000000	读存储器中 128 字节数据，支持 Cache 一致性。
2	存储器直接读	000100	读存储器中 128 字节数据，不支持 Cache 一致性，直接读出存储器中数据。
3	I/O 寄存器读	001000	读 I/O 寄存器，数据长度为 8 字节。
4	SCAN 扫出或监测状态	001100	读出指定状态链中 1024 位数据。

5	Cache 一致性的存储器字节写	100000	写存储器中 128 位数据的任意字节，支持 Cache 一致性。
6	Cache 一致性的存储器长字写	100101	写存储器中 128 字节数据，支持 Cache 一致性。
7	存储器直接写字节	101000	写存储器中 128 位数据的任意字节，不支持 Cache 一致性，数据直接写入存储器。
8	存储器直接写长字	101001	写存储器中 128 字节数据，不支持 Cache 一致性，数据直接写存储器。
9	I/O 寄存器写	100100	写 I/O 寄存器，数据长度为 8 字节。
10	SCAN 扫入字节	101101	将 8 字节数据写入指定控制链。
11	SCAN 扫入长字	101111	将 128 字节数据写入指定控制链。
12	初始化程序加载	111100	将 128 字节的初始化程序数据加载到每个核心的指令 Cache 中。
13	初始化程序加载结束	111111	指示初始化程序加载结束，无数据。
14	维护接口复位	110000	维护接口复位。

A.1.3 维护命令目标编码

维护命令包中定义的地址为 40 位物理地址 (PA[39:0])。40 位物理地址对应的维护命令目标如表 A-3 所示。

表 A-3: 维护命令目标编码

编号	维护命令包中地址 PA[39:36]	维护命令目标
1	000x	存储控制器控制的存储器空间。
2	001x	保留。
3	010x	保留。
4	011x	保留。
5	1000	存储控制器中 I/O 空间。
6	1001	保留。
7	1010	保留。
8	1011	保留。

9	1100	PCI-E 0 接口中 I/O 空间。
10	1101	维护接口中 I/O 空间。
11	1110	系统接口部件中 I/O 空间。
12	1111	PCI-E 1 接口中 I/O 空间。

A.1.4 维护命令详细说明

A.1.4.1 维护中断

维护中断命令是对申威 411 内指定的某个核心发送维护中断，地址为维护接口中维护中断控制寄存器地址（IOR: MT_INT）。第 8 字节的最低 2 位指示中断的目标核心（编码表示），第 9 字节的最低 2 位表示中断类型编码，“0x”指示正常维护中断，“10”指示维护睡眠中断，“11”指示维护唤醒中断。维护中断命令包的格式见表 A-4。

表 A-4: 维护中断命令包

字节 0	11	100100
字节 1		地址[7:0]
字节 2		地址[15:8]
字节 3		地址[23:16]
字节 4		地址[31:24]
字节 5		地址[39:32]
字节 6		数据有效位 (V[7:0] = 8'b1111_1111)
字节 7		数据有效位 (V[15:8]=0)
字节 8		“000000”，中断目标[1:0]
字节 9		“000000”，中断类型编码[1:0]
字节 10		00000000
字节 11		00000000
字节 12		00000000
字节 13		00000000
字节 14		00000000
字节 15		00000000

A.1.4.2 存储器读

存储器读命令是对申威 411 控制的存储器进行读操作，读数据的长度固定为 128 字节，并根据读命令编码确定是否对所读数据执行 Cache 一致性操作，“000000”表示执行 Cache 一致性操作，“000100”表示直接读存储器，不执行 Cache 一致性操作。存储器读命令包的格式见表 A-5。

表 A-5: 存储器读命令包

字节 0	11	000000/000100
字节 1	地址[7:0]	
字节 2	地址[15:8]	
字节 3	地址[23:16]	
字节 4	地址[31:24]	
字节 5	地址[39:32]	
字节 6	00000000	
字节 7	00000000	

对存储器读命令，读响应的数据长度固定为 128 字节。要求地址必须是 128 字节对界。

A.1.4.3 存储器写

存储器写命令是对申威 411 控制的存储器进行写操作，根据写命令编码决定写数据长度以及是否对所写数据执行 Cache 一致性操作。存储器字节写命令（包括 Cache 一致性的存储器字节写和存储器直接写字节）中的数据有效位有意义，其它存储器写命令中数据有效位无意义，总是写入地址对界的 128 字节。当写数据长度小于或等于 16 字节时，用存储器字节写命令（命令编码“100000”和“101000”），数据有效位表示的粒度为字节；当写数据长度超过 16 字节且不超过 128 字节时，需要分拆成若干存储器字节写命令来处理。存储器写命令包的格式见表 A-6，写命令编码见表 A-7。

表 A-6: 写存储器命令包

字节 0	11	100000/100101/101000/101001
字节 1	地址[7:0]	
字节 2	地址[15:8]	
字节 3	地址[23:16]	
字节 4	地址[31:24]	
字节 5	地址[39:32]	
字节 6	数据有效位 (V[7:0])	

字节 7	数据有效位 (V[15:8])
字节 8	数据 0

字节 7+n	数据 n-1

表 A-7: 写存储器命令编码与含义

编码	含义
100000	执行 Cache 一致性操作的存储器写, 写数据粒度为字节。
100101	执行 Cache 一致性操作的存储器写, 写数据粒度为长字。
101000	不执行 Cache 一致性操作的存储器直接写, 写数据粒度为字节。
101001	不执行 Cache 一致性操作的存储器直接写, 写数据粒度为长字。

A.1.4.4 I/O 寄存器读

I/O 寄存器读命令是对与申威 411 内 I/O 寄存器的读操作, I/O 寄存器读的数据长度为 64 位, 读数据粒度为字节。I/O 寄存器读命令包的格式见表 A-8。

表 A-8: I/O 寄存器读命令包

字节 0	11	001000
字节 1	地址[7:0]	
字节 2	地址[15:8]	
字节 3	地址[23:16]	
字节 4	地址[31:24]	
字节 5	地址[39:32]	
字节 6	数据有效位 (V[7:0])	
字节 7	00000000	

对一般的 I/O 寄存器, 读响应的数据长度固定为 64 位。

PCI-E 接口部件 (EP) 中的 I/O 寄存器支持 8b、16b、32b 和 64b 四种粒度。四种粒度与低位地址的对应关系如表 A-9:

表 A-9: 粒度与数据有效位关系

粒度	数据有效位	地址[2:0]
64b	8'b1111_1111	3'b000
32b	8'b0000_1111	3'b000
	8'b1111_0000	3'b100

16b	8'b0000_0011	3'b000
	8'b0000_1100	3'b010
	8'b0011_0000	3'b100
	8'b1100_0000	3'b110
8b	8'b0000_0001	3'b000
	8'b0000_0010	3'b001
	8'b0000_0100	3'b010
	8'b0000_1000	3'b011
	8'b0001_0000	3'b100
	8'b0010_0000	3'b101
	8'b0100_0000	3'b110
	8'b1000_0000	3'b111

维护接口的读响应数据为 64 位的存储器格式（8b/16b/32b 的响应数据在各自对应位置），维护系统根据请求数据有效位得到相应数据。

A.1.4.5 I/O 寄存器写

I/O 寄存器写命令是对申威 411 内 I/O 寄存器的写操作，I/O 寄存器写的数据长度为 64 位，写数据粒度为字节。I/O 寄存器写命令包的格式见表 A-10。

表 A-10：写 I/O 寄存器命令包

字节 0	11	100100
字节 1	地址[7:0]	
字节 2	地址[15:8]	
字节 3	地址[23:16]	
字节 4	地址[31:24]	
字节 5	地址[39:32]	
字节 6	数据有效位 (V[7:0])	
字节 7	8'b00000000	
字节 8	数据 0	
	
字节 7+n	数据 7	

对一般的 I/O 寄存器，读响应的数据长度固定为 64 位。

PCI-E 接口部件 (EP) 中的 I/O 寄存器支持 8b、16b、32b 和 64b 四种粒度。四种粒度与低位地址的对应关系如表 A-9。

维护接口的读响应数据为 64 位的存储器格式 (8b/16b/32b 的响应数据在各自对应位置), 维护系统根据请求数据有效位得到相应数据。

A.1.4.6 状态扫出

状态监测和状态扫描采用统一编址 (24 位地址), 全芯片共 16MB 的地址空间。状态扫出命令统一按 1024 位操作, 要求地址[6:0]为全零。状态信息位少于 1024 位时, 维护接口自动用“0”补齐。读状态链命令包的格式见表 A-11, 具体编址见表 A-12。

表 A-11: 读状态链命令包

字节 0	11	001100
字节 1	地址[7:0]	
字节 2	地址[15:8]	
字节 3	地址[23:16]	
字节 4	地址[31:24]	
字节 5	地址[39:32]	
字节 6	00000000	
字节 7	00000000	

表 A-12: SCAN 编址说明

地址[23:21]	模块	子模块	实际可用	
3'h0	CG0	核心 0([20:18]=3'b000)	8KB, 即地址[12:0]	
		核心 1([20:18]=3'b001)	8KB, 即地址[12:0]	
		核心 2([20:18]=3'b010)	8KB, 即地址[12:0]	
		核心 3([20:18]=3'b011)	8KB, 即地址[12:0]	
		CPM([20:18]=3'b100)	CPM([17]=0)	1KB, 即地址[9:0]
			TCDATA([17]=1)	2KB, 即地址[10:0]
		MC0([20:18]=3'b101)	2KB, 即地址[10:0]	
MC1([20:18]=3'b110)	2KB, 即地址[10:0]			
3'h5	IPU		1KB, 即地址[9:0]	
3'h6	PIU0	PIU0([20:18]=3'bxx0)	1KB, 即地址[9:0]	
		PageCache0([20:18]=3'bxx1)	256B, 即地址[7:0]	
3'h7	PIU1&MCU	PIU1([20:18]=3'b0x0)	1KB, 即地址[9:0]	

		PageCache1([20:18]=3'b0x1)	256B, 即地址[7:0]
		MCU([20:19]=2'b10)	1KB, 即地址[9:0]
其它	—	—	保留

A.1.4.7 状态扫入

状态扫描采用统一编址（24 位地址），全芯片共 16MB 的地址空间。状态扫入分为字节写（16 字节）和长字写（128 字节）两种，两种命令的地址都要求对界，即字节写地址[3:0]为零，长字写地址[6:0]为零。只有字节写命令支持数据有效位，即写入可精确到哪个字节。数据长度只有 128 位和 1024 位两种。当扫入的长度小于或等于 128 位时，用字节扫入，数据有效位表示对应字节的数据有效。如果扫入数据长度小于 1 个 Cache 行且大于 128 位时，需要由软件分拆成若干个 128 位。对不支持写入的 SCAN 寄存器写入不产生任何影响。

状态扫入命令包的格式见表 A-13，具体编址见表 A-12。

表 A-13: 状态扫入命令包

字节 0	11	101101/101111
字节 1		地址[7:0]
字节 2		地址[15:8]
字节 3		地址[23:16]
字节 4		地址[31:24]
字节 5		地址[39:32]
字节 6		数据有效位 (V[7:0])
字节 7		数据有效位 (V[15:8])
字节 8		数据 0
	
字节 7+n		数据 n-1 (n=16 或 128)

A.1.4.8 初始化程序加载

初始化程序加载命令实现对申威 411 内所有核心的指令 Cache 进行初始化程序加载。当命令编码为“111100”时，表示向核心的指令 Cache 加载初始化程序，每次加载 1 个 Cache 行，数据长度固定为 128 字节，数据有效位为全“1”。当命令编码为“111111”时，表示初始化程序加载结束，

数据有效位为全“0”，表示不含加载数据。初始化程序加载命令包的格式见表 A-14。

表 A-14: 初始化程序加载命令包

字节 0	11	111100/111111
字节 1	地址[7:0]	
字节 2	地址[15:8]	
字节 3	地址[23:16]	
字节 4	地址[31:24]	
字节 5	地址[39:32]	
字节 6	数据有效位 (V[7:0])	
字节 7	数据有效位 (V[15:8])	
字节 8	数据 0 (只针对命令“111100”)	
	
字节 7+n	数据 n-1 (只针对命令“111100”)	

初始化加载结束命令不带数据（命令包的长度为 8 字节）。

A.1.4.9 维护接口复位命令

执行维护接口复位命令，硬件自动对 4 内部的维护控制部件的部分逻辑进行复位，便于处理维护命令超时等特殊情况，但对申威 411 内部核心运行不产生直接影响。维护接口复位命令包的格式见表 A-15。

表 A-15: 维护接口复位命令包

字节 0	2'b11	6'b110000
字节 1	8'b00000000	
字节 2	8'b00000000	
字节 3	8'b00000000	
字节 4	8'b00000000	
字节 5	8'b00000000	
字节 6	8'b00000000	
字节 7	8'b00000000	

A.1.4.10 重新进行存储器自测试命令

重新存储器自测试是通过对 IOR: BIST_GOON 的写来实现。该寄存器只写，写该寄存器将触

发申威 411 内部重新进行存储器自测试。命令格式中的地址为该寄存器的地址。重新进行存储器自测试命令包的格式见表 A-16。

表 A-16: 维护接口复位命令包

字节 0	2'b11	6'b100100
字节 1		地址[7:0]
字节 2		地址[15:8]
字节 3		地址[23:16]
字节 4		地址[31:24]
字节 5		地址[39:32]
字节 6		8'b00000000
字节 7		8'b00000000
字节 8		8'b00000000
字节 9		8'b00000000
字节 10		8'b00000000
字节 11		8'b00000000
字节 12		8'b00000000
字节 13		8'b00000000
字节 14		8'b00000000
字节 15		8'b00000000

A.2 维护响应

A.2.1 维护响应包格式

申威 411 处理维护命令，产生维护响应，从维护接口发出的维护响应包（即维护命令处理结果）的格式见表 A-17。

表 A-17: 维护响应包

字节 0	11	响应编码[5:0]
字节 1	数据长度（字节数，只有 0x00、0x08 和 0x80 三种）	
字节 2	数据字节 0	
	
字节 1+n	数据字节 n-1	

A.2.2 维护响应编码

申威 411 支持 9 种维护响应，具体的响应编码如表 A-18 所示。

表 A-18: 维护响应编码及含义

编号	维护命令响应	编码
1	正常读响应（带数据，不含 ECC 多错）	000000
2	正常读响应（带数据，含 ECC 多错）	001000
3	错误读响应（带数据，含控制错）	010000
4	非法地址读响应（不带数据）	000001
5	写结束	000010
6	非法地址写结束	000011
7	带控制错写结束	001011
8	维护串口校验错响应（不带数据）	000100
9	维护串口非法命令响应（不带数据）	000101

A.2.3 有数据维护响应

读类命令处理返回有数据的维护响应（非法地址读响应没有数据），根据不同的读类命令，返回不同数据长度的维护响应，其中存储器读命令的响应数据长度为 1 个 Cache 行（128 字节），I/O 寄存器读命令的响应数据长度为 8 字节，读状态链命令的响应数据长度为 1024 位。有数据维护响应包的格式见表 A-19。

表 A-19: 有数据维护响应命令包

字节 0	11	000000/000001
字节 1	数据长度（字节数，只有 0x00、0x08 和 0x80 三种）	
字节 2	数据字节 0	
	
字节 1+n	数据字节 n-1	

A.2.4 无数据维护响应

这类响应不包含响应数据，包括写结束响应、写错误响应、校验错响应、非法命令响应和非法地址读响应，无数据维护响应包的格式见表 A-20。

表 A-20: 无数据维护响应包

字节 0	11	命令编码[5:0]
------	----	-----------

字节 1	数据长度 (0)
------	----------

附录 B 边界扫描链说明

B.1 GPIO BSR 链说明

表 B-1: GPIO BSR 链说明

链位置	端口名称	端口属性	端口描述
1	STAT_OUT_L[0]	输出	状态输出。
2	INIT_MODE_H[0]	输入	初始化模式。
3	INIT_MODE_H[1]	输入	初始化模式。
4	STAT_OUT_L[1]	输出	状态输出。
5	DCOK_H	输入	电源正常。
6	CFG_SEL_L	输入	配置方式选择
7	STAT_OUT_L[2]	输出	状态输出。
8	RESET_L	输入	复位信号。
9	STAT_OUT_SEL_L[0]	输入	测试输出选择。
10	STAT_OUT_L[3]	输出	状态输出。
11	MT_RX_H	输入	维护串行输入。
12	STAT_OUT_SEL_L[1]	输入	测试输出选择。
13	STAT_OUT_L[4]	输出	测试输出。
14	WAKEUP_L	输入	唤醒中断
15	MT_CLK_H	输入	维护时钟。
16	MT_TX_H	输出	维护串行输出。
17	CFG_MM_H[2]	输入	存控时钟配置。
18	CFG_MM_H[1]	输入	存控时钟配置。
19	CFG_MM_H[0]	输入	存控时钟配置。
20	NMI_L	输入	不可屏蔽中断。
21	CFG_CORE_H[3]	输入	核心时钟配置。
22	CFG_CORE_H[2]	输入	核心时钟配置。
23	CFG_CORE_H[1]	输入	核心时钟配置。
24	CFG_CORE_H[0]	输入	核心时钟配置。
25	CFG_XBX_H[2]	输入	互连时钟配置。

26	CFG_XBX_H[1]	输入	互连时钟配置。
27	CFG_XBX_H[0]	输入	互连时钟配置。

说明:

- 1) TAP_SEL_L[1:0]信号涉及 JTAG 的 TAP 控制器选择, 不放在 BSR 链上。

B.2 PCI-E BSR 链说明

申威 411 共有 2 个 PCI-E 接口, 每个 PCI-E 接口有 2 个 PHY。每个 PCI-E 接口将两个 PHY 的 BSR 串为一条链, 从 JTAG 的角度看, 每个 PCI-E 接口只有一条 BSR 链。每个 PCI-E 接口的 BSR 链顺序如下:

表 B-2: PCI-E BSR 链说明

链位置	端口名称	端口属性	端口描述
1	*		PCIEx_tx_p[0]输出使能控制。
2	PCIEx_tx_p[0]	输出	差分输出。
3	PCIEx_rx_p[0]	输入	差分输入。
4	PCIEx_rx_m[0]	输入	差分输入。
5	*		PCIEx_tx_p[1]输出使能控制。
6	PCIEx_tx_p[1]	输出	差分输出。
7	PCIEx_rx_p[1]	输入	差分输入。
8	PCIEx_rx_m[1]	输入	差分输入。
9	*		PCIEx_tx_p[2]输出使能控制。
10	PCIEx_tx_p[2]	输出	差分输出。
11	PCIEx_rx_p[2]	输入	差分输入。
12	PCIEx_rx_m[2]	输入	差分输入。
13	*		PCIEx_tx_p[3]输出使能控制。
14	PCIEx_tx_p[3]	输出	差分输出。
15	PCIEx_rx_p[3]	输入	差分输入。
16	PCIEx_rx_m[3]	输入	差分输入。
17	*		PCIEx_tx_p[4]输出使能控制。
18	PCIEx_tx_p[4]	输出	差分输出。
19	PCIEx_rx_p[4]	输入	差分输入。
20	PCIEx_rx_m[4]	输入	差分输入。
21	*		PCIEx_tx_p[5]输出使能控制

22	PCIEx_tx_p[5]	输出	差分输出。
23	PCIEx_rx_p[5]	输入	差分输入。
24	PCIEx_rx_m[5]	输入	差分输入。
25	*		PCIEx_tx_p[6]输出使能控制
26	PCIEx_tx_p[6]	输出	差分输出。
27	PCIEx_rx_p[6]	输入	差分输入。
28	PCIEx_rx_m[6]	输入	差分输入。
29	*		PCIEx_tx_p[7]输出使能控制
30	PCIEx_tx_p[7]	输出	差分输出。
31	PCIEx_rx_p[7]	输入	差分输入。
32	PCIEx_rx_m[7]	输入	差分输入。

B.3 DDR3 BSR 链说明

申威 411 集成了两个存储控制器。每个 DDR3 存储控制器包括控制器和 PHY。每个 PHY 的 BSR 链具体如下：

表 B-3: DDR3 BSR 链说明

链位置	端口名称	端口属性	意义
143	dto[0]	输出	DTO 引脚。
142	dto[1]	输出	DTO 引脚。
141	ram_rst_n	输出	复位信号。
140	ck[0]	输出	时钟。
139	ck[1]	输出	时钟。
138	ck_n[0]	输出	时钟。
137	ck_n[1]	输出	时钟。
136	cke[0]	输出	CKE。
135	cke[1]	输出	CKE。
134	odt[0]	输出	ODT。
133	odt[1]	输出	ODT。
132	cs_n[0]	输出	CS_N。
131	cs_n[1]	输出	CS_N。
130	ras_n	输出	RAS_N。

129	cas_n	输出	CAS_N。
128	we_n	输出	WE_N。
127	ba[0]	输出	BA 地址。
126	ba[1]	输出	BA 地址。
125	ba[2]	输出	BA 地址。
124	a[0]	输出	A 地址。
123	a[1]	输出	A 地址。
122	a[2]	输出	A 地址。
121	a[3]	输出	A 地址。
120	a[4]	输出	A 地址。
119	a[5]	输出	A 地址。
118	a[6]	输出	A 地址。
117	a[7]	输出	A 地址。
116	a[8]	输出	A 地址。
115	a[9]	输出	A 地址。
114	a[10]	输出	A 地址。
113	a[11]	输出	A 地址。
112	a[12]	输出	A 地址。
111	a[13]	输出	A 地址。
110	a[14]	输出	A 地址。
109	a[15]	输出	A 地址。
108	dq[0]	双向	dq[0]。
107	dq[1]	双向	dq[1]。
106	dq[2]	双向	dq[2]。
105	dq[3]	双向	dq[3]。
104	dq[4]	双向	dq[4]。
103	dq[5]	双向	dq[5]。
102	dq[6]	双向	dq[6]。
101	dq[7]	双向	dq[7]。
100	*		dq[7:0]的输出使能。
99	dqs[0]	双向	dqs[0]。
98	dqs_n[0]	双向	dqs_n[0]。
97	*		dqs[0]和 dqs_n[0]的输出使能。

96	dq[8]	双向	dq[8]。
95	dq[9]	双向	dq[9]。
94	dq[10]	双向	dq[10]。
93	dq[11]	双向	dq[11]。
92	dq[12]	双向	dq[12]。
91	dq[13]	双向	dq[13]。
90	dq[14]	双向	dq[14]。
89	dq[15]	双向	dq[15]。
88	*		dq[15:8]的输出使能。
87	dqs[1]	双向	dqs[1]。
86	dqs_n[1]	双向	dqs_n[1]。
85	*		dqs[1]和 dqs_n[1]的输出使能。
84	dq[16]	双向	dq[16]。
83	dq[17]	双向	dq[17]。
82	dq[18]	双向	dq[18]。
81	dq[19]	双向	dq[19]。
80	dq[20]	双向	dq[20]。
79	dq[21]	双向	dq[21]。
78	dq[22]	双向	dq[22]。
77	dq[23]	双向	dq[23]。
76	*		dq[23:16]的输出使能。
75	dqs[2]	双向	dqs[2]。
74	dqs_n[2]	双向	dqs_n[2]。
73	*		dqs[2]和 dqs_n[2]的输出使能。
72	dq[24]	双向	dq[24]。
71	dq[25]	双向	dq[25]。
70	dq[26]	双向	dq[26]。
69	dq[27]	双向	dq[27]。
68	dq[28]	双向	dq[28]。
67	dq[29]	双向	dq[29]。
66	dq[30]	双向	dq[30]。
65	dq[31]	双向	dq[31]。
64	*		dq[31:24]的输出使能。

63	dqs[3]	双向	dqs[3]。
62	dqs_n[3]	双向	dqs_n[3]。
61	*		dqs[3]和 dqs_n[3]的输出使能。
60	dq[32]	双向	dq[32]。
59	dq[33]	双向	dq[33]。
58	dq[34]	双向	dq[34]。
57	dq[35]	双向	dq[35]。
56	dq[36]	双向	dq[36]。
55	dq[37]	双向	dq[37]。
54	dq[38]	双向	dq[38]。
53	dq[39]	双向	dq[39]。
52	*		dq[39:32]的输出使能。
51	dqs[4]	双向	dqs[4]。
50	dqs_n[4]	双向	dqs_n[4]。
49	*		dqs[4]和 dqs_n[4]的输出使能。
48	dq[40]	双向	dq[40]。
47	dq[41]	双向	dq[41]。
46	dq[42]	双向	dq[42]。
45	dq[43]	双向	dq[43]。
44	dq[44]	双向	dq[44]。
43	dq[45]	双向	dq[45]。
42	dq[46]	双向	dq[46]。
41	dq[47]	双向	dq[47]。
40	*		dq[47:40]的输出使能。
39	dqs[5]	双向	dqs[5]。
38	dqs_n[5]	双向	dqs_n[5]。
37	*		dqs[5]和 dqs_n[5]的输出使能。
36	dq[48]	双向	dq[48]。
35	dq[49]	双向	dq[49]。
34	dq[50]	双向	dq[50]。
33	dq[51]	双向	dq[51]。
32	dq[52]	双向	dq[52]。
31	dq[53]	双向	dq[53]。

30	dq[54]	双向	dq[54]。
29	dq[55]	双向	dq[55]。
28	*		dq[55:48]的输出使能。
27	dqs[6]	双向	dqs[6]。
26	dqs_n[6]	双向	dqs_n[6]。
25	*		dqs[6]和 dqs_n[6]的输出使能。
24	dq[56]	双向	dq[56]。
23	dq[57]	双向	dq[57]。
22	dq[58]	双向	dq[58]。
21	dq[59]	双向	dq[59]。
20	dq[60]	双向	dq[60]。
19	dq[61]	双向	dq[61]。
18	dq[62]	双向	dq[62]。
17	dq[63]	双向	dq[63]。
16	*		dq[63:56]的输出使能。
15	dqs[7]	双向	dqs[7]。
14	Dqs_n[7]	双向	dqs_n[7]。
13	*		dqs[7]和 dqs_n[7]的输出使能。
12	dq[64]	双向	dq[64]。
11	dq[65]	双向	dq[65]。
10	dq[66]	双向	dq[66]。
9	dq[67]	双向	dq[67]。
8	dq[68]	双向	dq[68]。
7	dq[69]	双向	dq[69]。
6	dq[70]	双向	dq[70]。
5	dq[71]	双向	dq[71]。
4	*		dq[71:64]的输出使能。
3	dqs[8]	双向	dqs[8]。
2	dqs_n[8]	双向	dqs_n[8]。
1	*		dqs[8]和 dqs_n[8]的输出使能。