



SUNWAY 申威

申威 421 处理器 数据手册

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 421 处理器数据手册》主要描述了申威 421 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，并介绍了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 421 处理器数据手册
	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	接口概述.....	1
1.2	封装说明.....	1
1.3	焊接说明.....	4
2	引脚说明	5
2.1	引脚分类.....	5
2.2	引脚列表.....	5
2.3	时钟接口引脚.....	19
2.4	存储器接口引脚.....	19
2.5	PCIe 接口引脚.....	20
2.6	维护/中断/配置接口引脚.....	20
2.7	测试接口引脚.....	22
2.8	备份引脚.....	23
2.9	温度传感器引脚.....	23
2.10	电地引脚.....	24
3	电地及特殊信号接口	25
3.1	数字电源地.....	25
3.2	模拟电源地.....	25
3.3	特殊引脚.....	26
4	时钟接口	27
4.1	概述.....	27
4.2	工作模式.....	27
5	存储器接口	28
5.1	概述.....	28
5.2	存储器接口配置.....	28
5.2.1	1T/2T 模式.....	28
5.2.2	地址镜像.....	29
5.2.3	颗粒/内存类型.....	30
5.3	信号连接.....	30
5.3.1	选通信号.....	30
5.3.2	ECC 校验码.....	31
5.3.3	双内存模组连接方式.....	31
5.4	初始化.....	32
5.5	测试.....	32
6	PCI-E 接口	33

6.1	概述.....	33
6.2	信号连接.....	33
6.3	初始化.....	33
6.4	测试.....	34
7	维护、中断和配置接口	35
7.1	维护接口	35
7.1.1	概述	35
7.1.2	接口说明.....	35
7.1.3	维护功能.....	36
7.2	配置接口	36
7.2.1	初始化配置.....	36
7.2.2	时钟配置.....	37
7.3	中断接口	38
8	调试测试接口	40
8.1	概述.....	40
8.2	TAP 控制器.....	40
8.3	测试寄存器.....	41
8.4	边界扫描.....	43
8.5	调试操作.....	43
8.5.1	调试接口寄存器.....	43
8.5.2	调试操作流程.....	44
9	复位与初始化	47
9.1	复位.....	47
9.1.1	复位类型.....	47
9.1.2	复位状态机.....	47
9.2	配置.....	49
9.2.1	配置引脚.....	49
9.2.2	配置寄存器.....	50
9.3	初始化.....	51
9.3.1	加关电顺序.....	51
9.3.2	初始化流程.....	51
9.3.3	初始化程序.....	52
10	电气特性	53
10.1	工作条件.....	53
10.2	直流参数.....	53
10.2.1	差分时钟输入信号.....	53

10.2.2	存储器接口信号.....	54
10.2.3	1.8V LVCMOS 信号.....	54
10.3	交流参数.....	55
10.3.1	差分时钟输入信号.....	55
10.3.2	存储器接口信号.....	55
10.3.3	维护接口信号.....	56
10.3.4	测试接口信号.....	58
10.3.5	PCI-E 接口.....	58
10.4	功耗.....	59
10.4.1	封装热特性.....	59
10.4.2	热设计功耗.....	59
10.4.3	功耗管理.....	59
附录 A	维护接口的命令与响应.....	61
A.1	维护命令.....	61
A.1.1	维护命令包格式.....	61
A.1.2	维护命令编码.....	61
A.1.3	维护命令目标编码.....	62
A.1.4	维护命令详细说明.....	63
A.2	维护响应.....	72
A.2.1	维护响应包格式.....	72
A.2.2	维护响应编码.....	72
A.2.3	有数据维护响应.....	73
A.2.4	无数据维护响应.....	73
附录 B	边界扫描链说明.....	74
B.1	GPIO BSR 链说明.....	74
B.2	PCI-E BSR 链说明.....	76
B.3	DDR3 BSR 链说明.....	77

1 概述

申威 421 处理器为 64 位字长的 RISC 架构高性能通用处理器，单芯片中集成了 4 个同构的新一代申威 Core3A 核心、8MB 的三级共享 Cache、两路 64 位 DDR3 存储控制器、两路第三代标准 PCI-E 接口、支持调试和管理的维护接口以及符合 IEEE1149.1 标准的测试接口。

1.1 接口概述

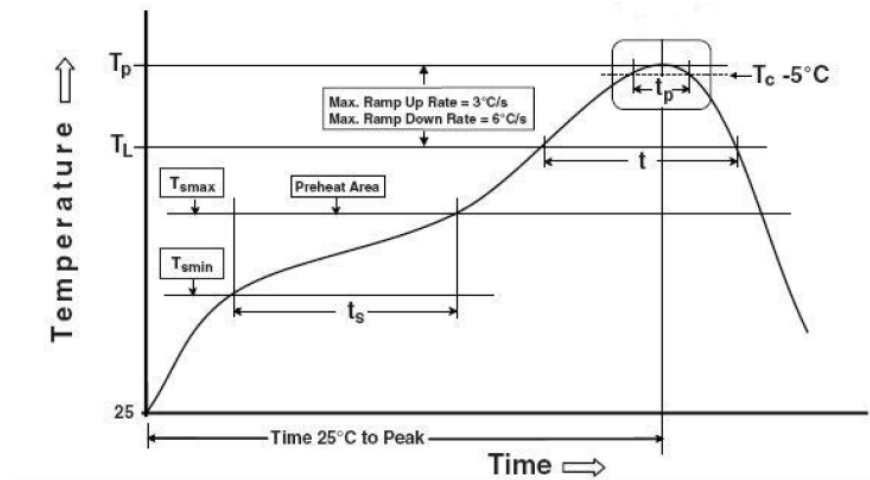
申威 421 处理器的外部硬件接口由六部分组成，具体如下：

- 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 3.0 规范，两路接口，每路接口支持 8 条链路，链路速率为 8Gbps；
- 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，为处理器提供模拟电源等特殊信号。

1.2 封装说明

申威 421 处理器采用 FC-BGA1156 封装，封装引脚数量 1144，芯片四角各预留三个 Ball。申威 421 处理器外观如图 1-1、图 1-2 和图 1-3 所示，图 1-1 为顶视图，图 1-2 为底视图，图 1-3 为侧视图。

1.3 焊接说明



Reflow Profile	T_{smin}	T_{smax}	t_s	T_L	t (time above T_L)	Max. T_p	t_p	Time to T_p
Std. Sn-Pb	100°C	150°C	60-120s	183°C	60-150s	220-235°C	10-20s	4-6min

图 1-4: 焊接曲线

申威 421 处理器的锡球是含铅设计，具体的焊接曲线如上图 1-4 所示。

2 引脚说明

2.1 引脚分类

申威 421 处理器采用 FC-BGA1156 封装，封装引脚数量 1144，芯片四角各预留三个 Ball。表 2-1 为 FCBGA 封装引脚的分类与数量。

表 2-1: FCBGA 封装引脚分类列表

类型	数量
信号引脚	378
电地引脚	677
备份引脚	85
温度传感器引脚	4
合计	1144

表 2-2 为申威 421 处理器信号引脚的分类与数量。

表 2-2: FCBGA 封装信号引脚分类列表

名称	数量	信号电平类型
存储器接口	265	SSTL (1.5V、1.35V 或 1.25V)
PCI-E 接口	70	除 PCI[1:0]_RESREF(双向引脚)连接参考电阻外,其他信号是 LVDS;
时钟接口	2	LVDS (1.8V)
测试接口	7	LVC MOS (1.8V)
维护接口	18	LVC MOS (1.8V)
配置接口	14	LVC MOS (1.8V)
中断接口	2	LVC MOS (1.8V)
合计	378	

注：信号引脚的信号电平主要是 SSTL (1.5V、1.35V 或 1.25V)、LVDS (1.8V) 和 LVC MOS (1.8V) 三大类，其中 SSTL 电平的 I/O 电源为 1.5V、1.35V 或 1.25V，其它为 1.8V。

2.2 引脚列表

FCBGA 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 2-1 所指示的字母和数字坐标组成，图 2-1 中，以左上角为原点，从左到右横坐标编号依次为数字 1~38；从上到下纵坐标编号依次为：A~Y（除 I、O、Q、S、X），AA~AV（除 AI、AO、AQ、AS）。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）。芯片有两路 DDR3 存储器接口，用前缀 MM0[H/L]_表示；芯片有 2 组 PCI-E 接口，用 PCI[1:0]_表示。图 2-1 是芯片的 TopView 视图，TopView 图例见图 2-2。

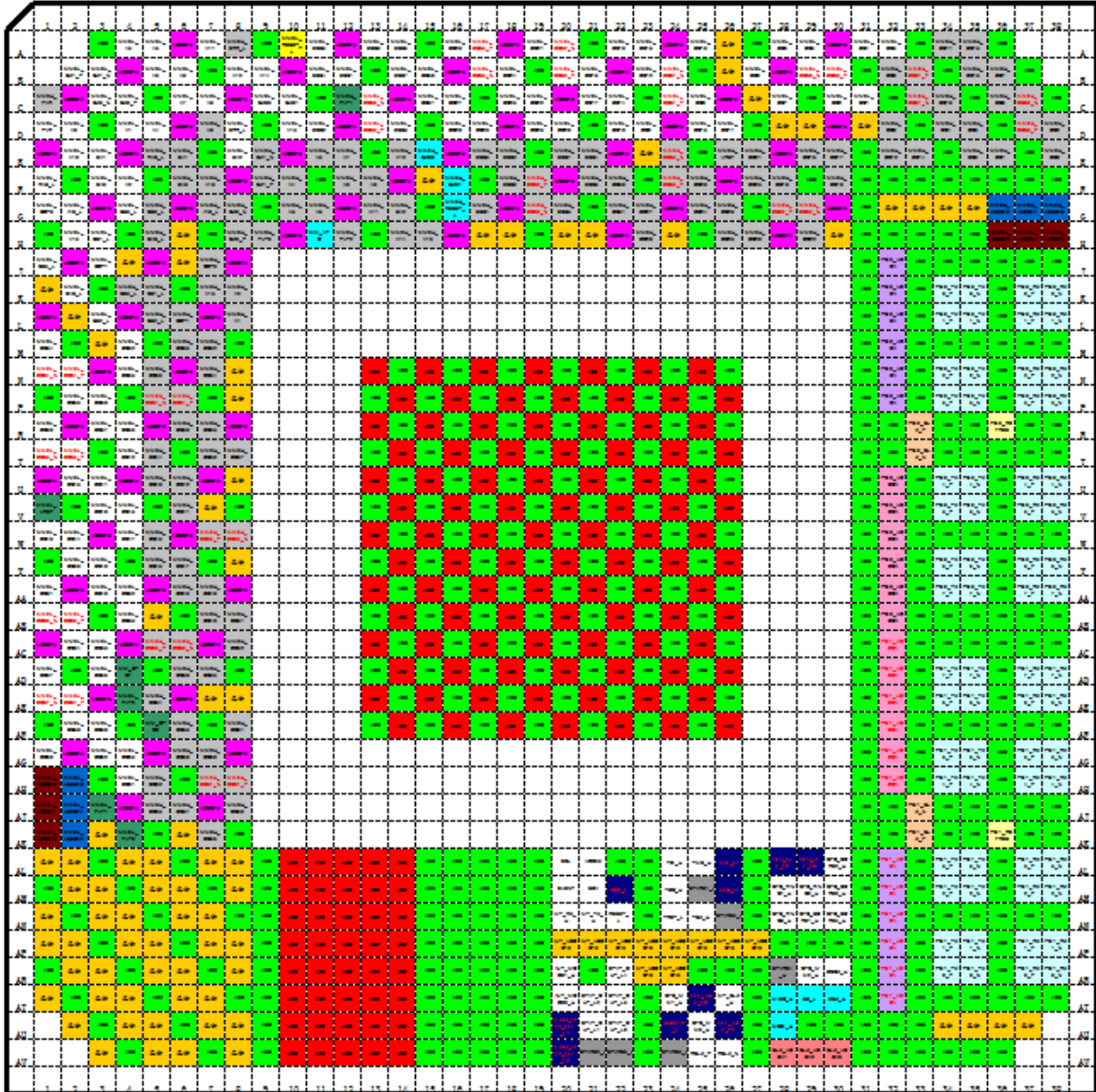


图 2-1: FCBGA 引脚排列图 (TopView) 摘自 V0.95 20161107

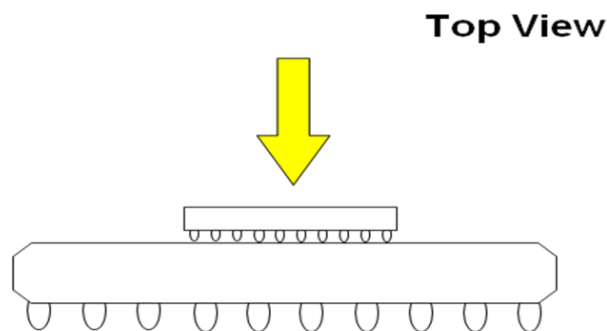


图 2-2: TopView 图例

表 2-3: FCBGA 封装引脚位置与名称对应表

C1	MM0H_PAR	A14	MM0L_ECC6	C27	备份
D1	MM0L_PAR	B14	MM0L_ECC7	D27	VSS
E1	VDDQ15	C14	VDDQ15	E27	MM0H_DQ21
F1	MM0L_RAS_L	D14	MM0L_ECC5	F27	MM0H_DQ23
G1	MM0L_ODT0	E14	MM0H_A12	G27	VSS
H1	VSS	F14	VDDQ15	H27	MM0H_DQ20
J1	MM0L_CS3_L	G14	MM0H_BA2	AL27	VSS
K1	备份	H14	MM0H_A14	AM27	VSS
L1	VDDQ15	N14	VSS	AN27	VSS
M1	MM0L_DQ34	P14	VDD	AP27	MT_VDDQ18
N1	MM0L_DQS4_N	R14	VSS	AR27	VSS
P1	VSS	T14	VDD	AT27	VSS
R1	MM0L_DQ38	U14	VSS	AU27	VSS
T1	MM0L_DQS5_N	V14	VDD	AV27	VSS
U1	VDDQ15	W14	VSS	A28	MM0L_DQ6
V1	MM0L_VREF	Y14	VDD	B28	VDDQ15
W1	MM0L_DQ46	AA14	VSS	C28	MM0L_DQ7
Y1	VSS	AB14	VDD	D28	备份
AA1	MM0L_DQ51	AC14	VSS	E28	VDDQ15
AB1	MM0L_DQS6_N	AD14	VDD	F28	MM0H_DQ19
AC1	VDDQ15	AE14	VSS	G28	MM0H_DQS2_P
AD1	MM0L_DQ57	AF14	VDD	H28	VDDQ15
AE1	MM0L_DQS7_N	AL14	VDD	AL28	CFG_SEL_H1
AF1	VSS	AM14	VDD	AM28	CFG_RING1_H
AG1	MM0L_DQ60	AN14	VDD	AN28	CFG_RING0_H
AH1	MM0L_VDDQA0	AP14	VDD	AP28	VSS
AJ1	MM0L_VDDQA1	AR14	VDD	AR28	SPARE4_H
AK1	MM0L_VDDQA2	AT14	VDD	AT28	MOSI_H
AL1	备份	AU14	VDD	AU28	MISO_H
AM1	VSS	AV14	VDD	AV28	CLK_VDDA1
AN1	备份	A15	VSS	A29	MM0L_DQ5
AP1	备份	B15	MM0L_DQ30	B29	MM0L_DQS0_N

AR1	VSS	C15	MM0L_DQ31	C29	VSS
AT1	备份	D15	VSS	D29	备份
B2	MM0L_CK1_P	E15	MM0H_CKE0	E29	MM0H_DQ16
C2	VDDQ15	F15	备份	F29	VSS
D2	MM0L_A0	G15	VSS	G29	MM0H_DQS2_N
E2	MM0L_A10	H15	MM0H_A15	H29	MM0H_DQ22
F2	VSS	N15	VDD	AL29	CFG_SEL_H0
G2	MM0L_WE_L	P15	VSS	AM29	CFG_RING2_H
H2	MM0L_A13	R15	VDD	AN29	CFG_CORE1_H
J2	VDDQ15	T15	VSS	AP29	VSS
K2	MM0L_CAS_L	U15	VDD	AR29	CFG_MM2_H
L2	备份	V15	VSS	AT29	SS_L
M2	VSS	W15	VDD	AU29	VSS
N2	MM0L_DQS4_P	Y15	VSS	AV29	CLK_VDDA2
P2	MM0L_DQ35	AA15	VDD	A30	VDDQ15
R2	VDDQ15	AB15	VSS	B30	MM0L_DQS0_P
T2	MM0L_DQS5_P	AC15	VDD	C30	MM0L_DQ2
U2	MM0L_DQ43	AD15	VSS	D30	VDDQ15
V2	VSS	AE15	VDD	E30	MM0H_DQ17
W2	MM0L_DQ44	AF15	VSS	F30	MM0H_DQ18
Y2	MM0L_DQ50	AL15	VSS	G30	VDDQ15
AA2	VDDQ15	AM15	VSS	H30	备份
AB2	MM0L_DQS6_P	AN15	VSS	AL30	CFG_CORE2_H
AC2	MM0L_DQ54	AP15	VSS	AM30	CFG_CORE0_H
AD2	VSS	AR15	VSS	AN30	CFG_CORE3_H
AE2	MM0L_DQS7_P	AT15	VSS	AP30	VSS
AF2	MM0L_DQ59	AU15	VSS	AR30	DCOK_H
AG2	VDDQ15	AV15	VSS	AT30	SCK_H
AH2	MM0L_VSSQA0	A16	MM0L_DQ28	AU30	VSS
AJ2	MM0L_VSSQA1	B16	VDDQ15	AV30	CLK_VDDA0
AK2	MM0L_VSSQA2	C16	MM0L_DQ27	A31	MM0L_DQ4
AL2	备份	D16	MM0L_DQ29	B31	VSS
AM2	备份	E16	VDDQ15	C31	MM0L_DQ1

AN2	VSS	F16	MM0H_CKE1	D31	备份
AP2	备份	G16	MM0H_RESET_L	E31	VSS
AR2	备份	H16	VDDQ15	F31	VSS
AT2	VSS	N16	VSS	G31	VSS
AU2	备份	P16	VDD	H31	VSS
A3	VSS	R16	VSS	J31	VSS
B3	MM0L_CK1_N	T16	VDD	K31	VSS
C3	MM0L_CK0_N	U16	VSS	L31	VSS
D3	VSS	V16	VDD	M31	VSS
E3	MM0L_BA1	W16	VSS	N31	VSS
F3	MM0L_BA0	Y16	VDD	P31	VSS
G3	VDDQ15	AA16	VSS	R31	VSS
H3	MM0L_CS1_L	AB16	VDD	T31	VSS
J3	MM0L_ODT1	AC16	VSS	U31	VSS
K3	VSS	AD16	VDD	V31	VSS
L3	MM0L_CS2_L	AE16	VSS	W31	VSS
M3	备份	AF16	VDD	Y31	VSS
N3	VDDQ15	AL16	VSS	AA31	VSS
P3	MM0L_DQ39	AM16	VSS	AB31	VSS
R3	MM0L_DQ37	AN16	VSS	AC31	VSS
T3	VSS	AP16	VSS	AD31	VSS
U3	MM0L_DQ41	AR16	VSS	AE31	VSS
V3	MM0L_DQ40	AT16	VSS	AF31	VSS
W3	VDDQ15	AU16	VSS	AG31	VSS
Y3	MM0L_DQ49	AV16	VSS	AH31	VSS
AA3	MM0L_DQ48	A17	MM0L_DQS3_P	AJ31	VSS
AB3	VSS	B17	MM0L_DQS3_N	AK31	VSS
AC3	MM0L_DQ53	C17	VSS	AL31	VSS
AD3	MM0L_DQ56	D17	MM0L_DQ26	AM31	VSS
AE3	VDDQ15	E17	MM0H_ECC3	AN31	VSS
AF3	MM0L_DQ58	F17	VSS	AP31	VSS
AG3	MM0L_DQ62	G17	MM0H_ECC1	AR31	VSS
AH3	VSS	H17	备份	AT31	VSS

AJ3	MM0L_PVT1	N17	VDD	AU31	VSS
AK3	备份	P17	VSS	AV31	VSS
AL3	VSS	R17	VDD	A32	MM0L_DQ3
AM3	备份	T17	VSS	B32	MM0H_DQ9
AN3	备份	U17	VDD	C32	VSS
AP3	VSS	V17	VSS	D32	MM0H_DQ8
AR3	备份	W17	VDD	E32	MM0H_DQ12
AT3	备份	Y17	VSS	F32	VSS
AU3	VSS	AA17	VDD	G32	备份
AV3	备份	AB17	VSS	H32	VSS
A4	MM0L_A3	AC17	VDD	J32	PCIO_VDDA
B4	VDDQ15	AD17	VSS	K32	PCIO_VDDA
C4	MM0L_CK0_P	AE17	VDD	L32	PCIO_VDDA
D4	MM0L_A1	AF17	VSS	M32	PCIO_VDDA
E4	VDDQ15	AL17	VSS	N32	PCIO_VDDA
F4	MM0L_A2	AM17	VSS	P32	PCIO_VDDA
G4	MM0L_CS0_L	AN17	VSS	R32	VSS
H4	VSS	AP17	VSS	T32	VSS
J4	备份	AR17	VSS	U32	PCIO_VDDQA
K4	MM0H_CS3_L	AT17	VSS	V32	PCIO_VDDQA
L4	VDDQ15	AU17	VSS	W32	PCIO_VDDQA
M4	MM0L_DQ32	AV17	VSS	Y32	PCIO_VDDQA
N4	MM0L_DQ33	A18	VDDQ15	AA32	PCIO_VDDQA
P4	VSS	B18	MM0L_DQ24	AB32	PCIO_VDDQA
R4	MM0L_DQ36	C18	MM0L_DQ25	AC32	PCI1_VDDQA
T4	MM0L_DQ42	D18	VDDQ15	AD32	PCI1_VDDQA
U4	VDDQ15	E18	MM0H_ECC2	AE32	PCI1_VDDQA
V4	MM0L_DQ45	F18	MM0H_ECC0	AF32	PCI1_VDDQA
W4	MM0L_DQ47	G18	VDDQ15	AG32	PCI1_VDDQA
Y4	VSS	H18	备份	AH32	PCI1_VDDQA
AA4	MM0L_DQ52	N18	VSS	AJ32	VSS
AB4	MM0L_DQ55	P18	VDD	AK32	VSS
AC4	VDDQ15	R18	VSS	AL32	PCI1_VDDA

AD4	MM DTO1	T18	VDD	AM32	PCI1_VDDA
AE4	MM0L_PVT2	U18	VSS	AN32	PCI1_VDDA
AF4	VSS	V18	VDD	AP32	PCI1_VDDA
AG4	MM0L_DQ63	W18	VSS	AR32	PCI1_VDDA
AH4	MM0L_DQ61	Y18	VDD	AT32	PCI1_VDDA
AJ4	VDDQ15	AA18	VSS	AU32	VSS
AK4	MM0L_PVT0	AB18	VDD	AV32	VSS
AL4	备份	AC18	VSS	A33	VSS
AM4	VSS	AD18	VDD	B33	MM0H_DQS1_P
AN4	备份	AE18	VSS	C33	MM0H_DQS1_N
AP4	备份	AF18	VDD	D33	VSS
AR4	VSS	AL18	VSS	E33	MM0H_DQ14
AT4	备份	AM18	VSS	F33	VSS
AU4	备份	AN18	VSS	G33	备份
AV4	VSS	AP18	VSS	H33	VSS
A5	MM0L_A5	AR18	VSS	J33	VSS
B5	MM0L_A6	AT18	VSS	K33	VSS
C5	VSS	AU18	VSS	L33	VSS
D5	MM0L_A4	AV18	VSS	M33	VSS
E5	MM0H_RAS_L	A19	MM0L_DQ21	N33	VSS
F5	VSS	B19	VSS	P33	VSS
G5	MM0H_CS0_L	C19	MM0L_DQ22	R33	PCIO_CLK_P
H5	MM0H_CAS_L	D19	MM0L_DQ23	T33	PCIO_CLK_N
J5	VDDQ15	E19	VSS	U33	VSS
K5	MM0H_CS1_L	F19	MM0H_DQS8_P	V33	VSS
L5	MM0H_CS2_L	G19	MM0H_DQS8_N	W33	VSS
M5	VSS	H19	VSS	Y33	VSS
N5	MM0H_DQ36	N19	VDD	AA33	VSS
P5	MM0H_DQS4_N	P19	VSS	AB33	VSS
R5	VDDQ15	R19	VDD	AC33	VSS
T5	MM0H_DQ44	T19	VSS	AD33	VSS
U5	MM0H_DQ40	U19	VDD	AE33	VSS
V5	VSS	V19	VSS	AF33	VSS

W5	MM0H_DQ43	W19	VDD	AG33	VSS
Y5	MM0H_DQ45	Y19	VSS	AH33	VSS
AA5	VDDQ15	AA19	VDD	AJ33	PCI1_CLK_N
AB5	备份	AB19	VSS	AK33	PCI1_CLK_P
AC5	MM0H_DQS6_P	AC19	VDD	AL33	VSS
AD5	VSS	AD19	VSS	AM33	VSS
AE5	MM0H_DQ54	AE19	VDD	AN33	VSS
AF5	MM_DTO0	AF19	VSS	AP33	VSS
AG5	VDDQ15	AL19	VSS	AR33	VSS
AH5	MM0H_DQ62	AM19	VSS	AT33	VSS
AJ5	MM0H_DQ63	AN19	VSS	AU33	VSS
AK5	VSS	AP19	VSS	AV33	VSS
AL5	备份	AR19	VSS	A34	MM0H_DQ11
AM5	备份	AT19	VSS	B34	VSS
AN5	VSS	AU19	VSS	C34	MM0H_DQ10
AP5	备份	AV19	VSS	D34	MM0H_DQ4
AR5	备份	A20	MM0L_DQS2_N	E34	VSS
AT5	VSS	B20	MM0L_DQS2_P	F34	VSS
AU5	备份	C20	VDDQ15	G34	备份
AV5	备份	D20	MM0L_DQ20	H34	VSS
A6	VDDQ15	E20	MM0H_ECC7	J34	VSS
B6	MM0L_A8	F20	VDDQ15	K34	PCIO_TX0_P
C6	MM0L_A7	G20	MM0H_ECC5	L34	PCIO_TX1_P
D6	VDDQ15	H20	备份	M34	VSS
E6	MM0H_BA1	N20	VSS	N34	PCIO_TX2_P
F6	MM0H_BA0	P20	VDD	P34	PCIO_TX3_P
G6	VDDQ15	R20	VSS	R34	VSS
H6	备份	T20	VDD	T34	VSS
J6	备份	U20	VSS	U34	PCIO_TX4_P
K6	VSS	V20	VDD	V34	PCIO_TX5_P
L6	MM0H_ODT1	W20	VSS	W34	VSS
M6	MM0H_DQ33	Y20	VDD	Y34	PCIO_TX6_P
N6	VDDQ15	AA20	VSS	AA34	PCIO_TX7_P

P6	MM0H_DQS4_P	AB20	VDD	AB34	VSS
R6	MM0H_DQ38	AC20	VSS	AC34	VSS
T6	VSS	AD20	VDD	AD34	PCI1_TX7_P
U6	MM0H_DQ42	AE20	VSS	AE34	PCI1_TX6_P
V6	MM0H_DQ41	AF20	VDD	AF34	VSS
W6	VDDQ15	AL20	SCL	AG34	PCI1_TX5_P
Y6	MM0H_DQ47	AM20	EVENT	AH34	PCI1_TX4_P
AA6	MM0H_DQ46	AN20	MT_RX_H	AJ34	VSS
AB6	VSS	AP20	MT_VDDQ18	AK34	VSS
AC6	MM0H_DQS6_N	AR20	INIT_MODE1_H	AL34	PCI1_TX3_P
AD6	MM0H_DQ50	AT20	INIT_MODE0_H	AM34	PCI1_TX2_P
AE6	VDDQ15	AU20	STAT_OUT_SEL_L1	AN34	VSS
AF6	MM0H_DQ55	AV20	STAT_OUT_SEL_L0	AP34	PCI1_TX1_P
AG6	MM0H_DQ59	A21	VSS	AR34	PCI1_TX0_P
AH6	VSS	B21	MM0L_DQ16	AT34	VSS
AJ6	MM0H_DQ57	C21	MM0L_DQ17	AU34	备份
AK6	备份	D21	VSS	AV34	VSS
AL6	VSS	E21	MM0H_ECC4	A35	MM0H_DQ13
AM6	备份	F21	MM0H_ECC6	B35	MM0H_DQ15
AN6	备份	G21	VSS	C35	VSS
AP6	VSS	H21	备份	D35	MM0H_DQ5
AR6	备份	N21	VDD	E35	MM0H_DQ3
AT6	备份	P21	VSS	F35	VSS
AU6	VSS	R21	VDD	G35	备份
AV6	备份	T21	VSS	H35	VSS
A7	MM0L_A11	U21	VDD	J35	VSS
B7	VSS	V21	VSS	K35	PCI0_TX0_N
C7	MM0L_A9	W21	VDD	L35	PCI0_TX1_N
D7	MM0H_A0	Y21	VSS	M35	VSS
E7	VSS	AA21	VDD	N35	PCI0_TX2_N
F7	MM0H_A10	AB21	VSS	P35	PCI0_TX3_N
G7	MM0H_WE_L	AC21	VDD	R35	VSS
H7	VSS	AD21	VSS	T35	VSS

J7	MM0H_ODT0	AE21	VDD	U35	PCIO_TX4_N
K7	MM0H_A13	AF21	VSS	V35	PCIO_TX5_N
L7	VDDQ15	AL21	VDD33	W35	VSS
M7	MM0H_DQ32	AM21	SDA	Y35	PCIO_TX6_N
N7	MM0H_DQ34	AN21	MT_TX_H	AA35	PCIO_TX7_N
P7	VSS	AP21	MT_VDDQ18	AB35	VSS
R7	MM0H_DQ35	AR21	VSS	AC35	VSS
T7	MM0H_DQ39	AT21	STAT_OUT_L2	AD35	PCI1_TX7_N
U7	VDDQ15	AU21	STAT_OUT_L1	AE35	PCI1_TX6_N
V7	备份	AV21	SPARE2_H	AF35	VSS
W7	MM0H_DQS5_P	A22	MM0L_DQ18	AG35	PCI1_TX5_N
Y7	VSS	B22	VDDQ15	AH35	PCI1_TX4_N
AA7	MM0H_DQ48	C22	MM0L_DQ14	AJ35	VSS
AB7	MM0H_DQ49	D22	MM0L_DQ19	AK35	VSS
AC7	VDDQ15	E22	VDDQ15	AL35	PCI1_TX3_N
AD7	MM0H_DQ53	F22	MM0H_DQ30	AM35	PCI1_TX2_N
AE7	备份	G22	MM0H_DQ31	AN35	VSS
AF7	VSS	H22	VDDQ15	AP35	PCI1_TX1_N
AG7	MM0H_DQ60	N22	VSS	AR35	PCI1_TX0_N
AH7	MM0H_DQS7_N	P22	VDD	AT35	VSS
AJ7	VDDQ15	R22	VSS	AU35	备份
AK7	MM0H_DQ56	T22	VDD	AV35	VSS
AL7	备份	U22	VSS	A36	VSS
AM7	VSS	V22	VDD	B36	MM0H_DQ7
AN7	备份	W22	VSS	C36	MM0H_DQ6
AP7	备份	Y22	VDD	D36	VSS
AR7	VSS	AA22	VSS	E36	MM0H_DQ1
AT7	备份	AB22	VDD	F36	VSS
AU7	备份	AC22	VSS	G36	MM0H_VSSQA2
AV7	VSS	AD22	VDD	H36	MM0H_VDDQA2
A8	MM0H_ERR_L	AE22	VSS	J36	VSS
B8	MM0L_A12	AF22	VDD	K36	VSS
C8	VDDQ15	AL22	VSS	L36	VSS

D8	MM0L_ERR_L	AM22	NMI_L	M36	VSS
E8	MM0L_BA2	AN22	RESET_L	N36	VSS
F8	VDDQ15	AP22	MT_VDDQ18	P36	VSS
G8	MM0H_CK0_N	AR22	STAT_OUT_L4	R36	PCI0_REFRES
H8	MM0H_CK0_P	AT22	STAT_OUT_L3	T36	VSS
J8	VDDQ15	AU22	STAT_OUT_L0	U36	VSS
K8	MM0H_A2	AV22	SPARE6_H	V36	VSS
L8	MM0H_A1	A23	MM0L_DQ15	W36	VSS
M8	VSS	B23	MM0L_DQ12	Y36	VSS
N8	备份	C23	VSS	AA36	VSS
P8	备份	D23	MM0L_DQ9	AB36	VSS
R8	VDDQ15	E23	备份	AC36	VSS
T8	MM0H_DQ37	F23	VSS	AD36	VSS
U8	备份	G23	MM0H_DQ27	AE36	VSS
V8	VSS	H23	MM0H_DQ29	AF36	VSS
W8	MM0H_DQS5_N	N23	VDD	AG36	VSS
Y8	备份	P23	VSS	AH36	VSS
AA8	VDDQ15	R23	VDD	AJ36	VSS
AB8	MM0H_DQ51	T23	VSS	AK36	PCI1_REFRES
AC8	MM0H_DQ52	U23	VDD	AL36	VSS
AD8	VSS	V23	VSS	AM36	VSS
AE8	备份	W23	VDD	AN36	VSS
AF8	MM0H_DQ61	Y23	VSS	AP36	VSS
AG8	VDDQ15	AA23	VDD	AR36	VSS
AH8	MM0H_DQS7_P	AB23	VSS	AT36	VSS
AJ8	MM0H_DQ58	AC23	VDD	AU36	备份
AK8	VSS	AD23	VSS	AV36	VSS
AL8	备份	AE23	VDD	B37	VSS
AM8	备份	AF23	VSS	C37	MM0H_DQS0_N
AN8	VSS	AL23	VSS	D37	MM0H_DQS0_P
AP8	备份	AM23	VSS	E37	VSS
AR8	备份	AN23	VSS	F37	VSS
AT8	VSS	AP23	MT_VDDQ18	G37	MM0H_VSSQA1

AU8	备份	AR23	MT_VDDQ18	H37	MM0H_VDDQA1
AV8	备份	AT23	VSS	J37	VSS
A9	VSS	AU23	VSS	K37	PCIO_RX0_P
B9	MM0L_A14	AV23	VSS	L37	PCIO_RX1_P
C9	MM0L_CKE0	A24	VDDQ15	M37	VSS
D9	VSS	B24	MM0L_DQS1_N	N37	PCIO_RX2_P
E9	MM0H_CK1_N	C24	MM0L_DQS1_P	P37	PCIO_RX3_P
F9	MM0H_CK1_P	D24	VDDQ15	R37	VSS
G9	VSS	E24	MM0H_DQS3_N	T37	VSS
H9	MM0H_PVT0	F24	MM0H_DQS3_P	U37	PCIO_RX4_P
AL9	VSS	G24	VDDQ15	V37	PCIO_RX5_P
AM9	VSS	H24	备份	W37	VSS
AN9	VSS	N24	VSS	Y37	PCIO_RX6_P
AP9	VSS	P24	VDD	AA37	PCIO_RX7_P
AR9	VSS	R24	VSS	AB37	VSS
AT9	VSS	T24	VDD	AC37	VSS
AU9	VSS	U24	VSS	AD37	PCI1_RX7_P
AV9	VSS	V24	VDD	AE37	PCI1_RX6_P
A10	MM0L_RESET_L	W24	VSS	AF37	VSS
B10	VDDQ15	Y24	VDD	AG37	PCI1_RX5_P
C10	MM0L_CKE1	AA24	VSS	AH37	PCI1_RX4_P
D10	MM0L_A15	AB24	VDD	AJ37	VSS
E10	VDDQ15	AC24	VSS	AK37	VSS
F10	MM0H_A4	AD24	VDD	AL37	PCI1_RX3_P
G10	MM0H_A3	AE24	VSS	AM37	PCI1_RX2_P
H10	VDDQ15	AF24	VDD	AN37	VSS
AL10	VDD	AL24	TDI_H	AP37	PCI1_RX1_P
AM10	VDD	AM24	TDO_H	AR37	PCI1_RX0_P
AN10	VDD	AN24	TRST_L	AT37	VSS
AP10	VDD	AP24	MT_VDDQ18	AU37	备份
AR10	VDD	AR24	MT_VDDQ18	C38	VSS
AT10	VDD	AT24	CFG_MM1_H	D38	MM0H_DQ2
AU10	VDD	AU24	WAKEUP_L	E38	MM0H_DQ0

AV10	VDD	AV24	SPARE5_H	F38	VSS
A11	MM0L_ECC3	A25	MM0L_DQ10	G38	MM0H_VSSQA0
B11	MM0L_ECC4	B25	VSS	H38	MM0H_VDDQA0
C11	VSS	C25	MM0L_DQ8	J38	VSS
D11	MM0L_ECC2	D25	MM0L_DQ13	K38	PCI0_RX0_N
E11	MM0H_A5	E25	VSS	L38	PCI0_RX1_N
F11	VSS	F25	MM0H_DQ28	M38	VSS
G11	MM0H_A6	G25	MM0H_DQ24	N38	PCI0_RX2_N
H11	MM_ATO	H25	VSS	P38	PCI0_RX3_N
AL11	VDD	N25	VDD	R38	VSS
AM11	VDD	P25	VSS	T38	VSS
AN11	VDD	R25	VDD	U38	PCI0_RX4_N
AP11	VDD	T25	VSS	V38	PCI0_RX5_N
AR11	VDD	U25	VDD	W38	VSS
AT11	VDD	V25	VSS	Y38	PCI0_RX6_N
AU11	VDD	W25	VDD	AA38	PCI0_RX7_N
AV11	VDD	Y25	VSS	AB38	VSS
A12	VDDQ15	AA25	VDD	AC38	VSS
B12	MM0L_ECC1	AB25	VSS	AD38	PCI1_RX7_N
C12	MM0H_PVT1	AC25	VDD	AE38	PCI1_RX6_N
D12	VDDQ15	AD25	VSS	AF38	VSS
E12	MM0H_A7	AE25	VDD	AG38	PCI1_RX5_N
F12	MM0H_A8	AF25	VSS	AH38	PCI1_RX4_N
G12	VDDQ15	AL25	TMS_H	AJ38	VSS
H12	MM0H_PVT2	AM25	SPARE1_H	AK38	VSS
AL12	VDD	AN25	TCK_H	AL38	PCI1_RX3_N
AM12	VDD	AP25	MT_VDDQ18	AM38	PCI1_RX2_N
AN12	VDD	AR25	VSS	AN38	VSS
AP12	VDD	AT25	SEL_TAP_H0	AP38	PCI1_RX1_N
AR12	VDD	AU25	CFG_MM0_H	AR38	PCI1_RX0_N
AT12	VDD	AV25	RCLK_P	AT38	VSS
AU12	VDD	A26	备份		
AV12	VDD	B26	备份		

A13	MM0L_ECC0	C26	VDDQ15		
B13	VSS	D26	MM0L_DQ11		
C13	MM0L_DQS8_N	E26	MM0H_VREF		
D13	MM0L_DQS8_P	F26	VDDQ15		
E13	VSS	G26	MM0H_DQ26		
F13	MM0H_A9	H26	MM0H_DQ25		
G13	MM0H_A11	N26	VSS		
H13	VSS	P26	VDD		
N13	VDD	R26	VSS		
P13	VSS	T26	VDD		
R13	VDD	U26	VSS		
T13	VSS	V26	VDD		
U13	VDD	W26	VSS		
V13	VSS	Y26	VDD		
W13	VDD	AA26	VSS		
Y13	VSS	AB26	VDD		
AA13	VDD	AC26	VSS		
AB13	VSS	AD26	VDD		
AC13	VDD	AE26	VSS		
AD13	VSS	AF26	VDD		
AE13	VDD	AL26	SCL_H		
AF13	VSS	AM26	SDA_H		
AL13	VDD	AN26	SPARE3_H		
AM13	VDD	AP26	MT_VDDQ18		
AN13	VDD	AR26	VSS		
AP13	VDD	AT26	MT_CLK_H		
AR13	VDD	AU26	SEL_TAP_H1		
AT13	VDD	AV26	RCLK_N		
AU13	VDD	A27	VSS		
AV13	VDD	B27	MM0L_DQ0		

注：名称为“SPARE*”的引脚为空引脚。在使用申威 421 处理器时，应保持这些引脚悬空，不连接任何信号。

2.3 时钟接口引脚

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			

2.4 存储器接口引脚

信号名称	位宽	I/O	引脚类型	描述	
信号引脚					
MM0[L/H]_RESET_L	1	输出	1.5V/1.35V /1.25V SSTL	DDR3 存储器异步复位信号。	
MM0[L/H]_CK_P[1:0]	2	输出		DDR3 存储器差分输入参考时钟。	
MM0[L/H]_CK_N[1:0]	2	输出		对应每个 Rank 的时钟使能信号。	
MM0[L/H]_CKE_H[1:0]	2	输出		对应每个 Rank 的内部端接使能信号。	
MM0[L/H]_ODT_H[1:0]	2	输出		对应每个 Rank 的片选信号。	
MM0[L/H]_CS_L[3:0]	4	输出		行地址 RAS#命令。	
MM0[L/H]_RAS_L	1	输出		列地址 CAS#命令。	
MM0[L/H]_CAS_L	1	输出		写 WE#命令。	
MM0[L/H]_WE_L	1	输出		存储器体 (Bank) 地址。	
MM0[L/H]_BA_H[2:0]	3	输出		存储器地址。	
MM0[L/H]_A_H[15:0]	16	输出		校验位	
MM0[L/H]_PAR_IN	1	输出		校验错	
MM0[L/H]_ERROUT_L	1	输入		以字节为单位的差分读/写数据选通信号。其中[8]对应 ECC 所在 Datx8 (按 DIMM 插槽引脚命名规范), 但引脚位置在九个 Datx8 的中间。	
MM0[L/H]_DQS_P[8:0]	9	双向		9	读/写数据。
MM0[L/H]_DQS_N[8:0]		双向			
MM0[L/H]_DQ_H[63:0]	64	双向			

MM0[L/H]_ECC_H[7:0]	8	双向		读/写数据的 ECC 码。	
总计	254	—			
参考电压、PVT 补偿电阻和测试引脚					
MM0[L/H]_VREF	1	—	参考电压	参考电压。	
MM0[L/H]_PVT[2:0]	3	—	PVT 电阻	PVT 补偿电阻。	
MM_ATO	1	输出	1.5V/1.35V /1.25V SSTL	ATO 引脚。	测试引脚， 可悬空。
MM.DTO[1:0]	2	输出		DTO 引脚。	
总计	11	—			

注1：申威421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：MM0[L/H]对应两路存控，分别是 0L 和 0H。

2.5 PCIe 接口引脚

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			

2.6 维护/中断/配置接口引脚

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

MT_CLK_H	1	输入	1.8V LVCMOS	维护时钟输入，频率为 5~25MHz。 注：该时钟需要先提供给 CPU 使用，即上电之后需优先提供（可以工作在 Standby 电源域）。
DCOK_H	1	输入	1.8V LVCMOS	电源就绪，低电平时作为上电复位。
RESET_L	1	输入	1.8V LVCMOS	冷复位，低电平有效。
MT_TX_H	1	输出	1.8V LVCMOS	维护接口串行输出数据，同步于 MT_CLK_H，在时钟的下降沿输出。
MT_RX_H	1	输入	1.8V LVCMOS	维护接口串行输入数据，同步于 MT_CLK_H，在时钟的上升沿接收。 注：如果维护口不使用时（默认不使用），需要接下拉电阻到地。
STAT_OUT_L [4:0]	5	输出	1.8V LVCMOS	芯片测试输出。
STAT_OUT_SEL_L [1:0]	2	输入	1.8V LVCMOS	芯片测试输出选择。
SS_L	1	输出	1.8V LVCMOS	SPI 使能信号，低有效。
SCK_H	1	输出	1.8V LVCMOS	SPI 时钟信号，频率与维护输入时钟相同。
MOSI_H_OUT	1	输出	1.8V LVCMOS	SPI 串行移出。
MISO_H_IN	1	输入	1.8V LVCMOS	SPI 串行移入。
SCL_M_H	1	双向	1.8V LVCOMS	I2C 时钟线。
SDA_M_H	1	双向	1.8V LVCOMS	I2C 数据线。
总计	18	—	—	—

信号名称	位宽	I/O	引脚类型	描述
CFG_CORE_H[3:0]	4	输入	1.8V	核心时钟 PLL 配置。

CFG_MM_H[2:0]	3	输入	LVCMOS	存储器接口时钟 PLL 配置。
CFG_XBX_H[2:0]	3	输入		互连时钟 PLL 配置。
INIT_MODE_H[1:0]	2	输入		初始化模式。应设置值为“11”，其它设置值用于处理器的测试。
CFG_SEL_H[1:0]	2	输入		<p>芯片配置说明。</p> <p>“2'b00”：采用引脚配置方式，根据时钟控制引脚和初始化模式引脚来设置初始化寄存器 INT_CTL，并可以通过维护命令进行配置修改。</p> <p>“2'b01”：采用寄存器默认值配置方式，即配置引脚无效，采用寄存器缺省值，并可以通过维护命令进行配置修改。</p> <p>“2'b1x”：芯片自启动方式，即自动读取 SPI Flash 获得配置参数及初始引导程序。</p>
共计	14			

信号名称	位宽	I/O	引脚类型	描述
WAKEUP_L	1	输入	1.8V LVCMOS	唤醒中断，默认低电平有效（有效方式可通过维护寄存器进行配置）。唤醒中断引脚的使用说明见表 7-9。
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 7-9。
共计	2			

2.7 测试接口引脚

信号名称	位宽	I/O	引脚类型	描述
------	----	-----	------	----

TCK_H	1	输入	1.8V LVCMOS	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'b10: 保留; 2'b11: 表示使用 TBOX 的 TAP。
共计	7			

2.8 备份引脚

引脚名称	位宽	I/O	引脚类型	描述
备份	79	—	空引脚	空引脚, 保持此类引脚悬空, 不连接任何信号。
SPARE*	6	—	空引脚	空引脚, 保持此类引脚悬空, 不连接任何信号。
共计	85			

2.9 温度传感器引脚

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	—	—	—

2.10 电地引脚

引脚名称	位宽	I/O	引脚类型	描述
VSS	421	—	地	地。
VDD	138	—	电源	内核电源。
VDDQ15	69	—	电源	DDR3 存储器接口 1.5V/1.35/1.25V I/O 电源。
MM0[L/H]_VDDQA[2:0]	6	—	模拟电源	DDR3 存储器接口 1.8V 模拟电源。
MM0[L/H]_VSSQA[2:0]	6	—	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/1.5V 接口模拟电源。
PCI[1:0]_VDDA	12	—	模拟电源	PCI-E 接口 0.95V 核心模拟电源。
CLK_VDDA[2:0]	3	—	模拟电源	时钟接口 1.8V 模拟电源。
MT_VDDQ18	10	—	电源	GPIO/LVDS 1.8V 电源
总计	677	—	—	—

3 电地及特殊信号接口

3.1 数字电源地

申威 421 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。三种电源的电压范围见表 3-1。三种数字电源对应的数字地都是 VSS。

表 3-1: FCBGA 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 Vcore	0.91	1.05	1.07	V	
VDDQ15	1.5V I/O 电源电压 V _{IO15}	1.425	1.5	1.575	V	
	1.35V I/O 电源电压 V _{IO15}	1.283	1.35	1.45	V	
	1.25V I/O 电源电压 V _{IO15}	1.19	1.25	1.31	V	
MT_VDDQ18	1.8V I/O 电源电压 V _{IO18}	1.71	1.8	1.89	V	

3.2 模拟电源地

申威 421 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 3-3。

表 3-2: 申威 421 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[L/H]_VDDQA[2:0]	6	DDR3 存储器接口 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。
MM0[L/H]_VSSQA[2:0]	6	DDR3 存储器接口 1.8V 模拟地。	在系统 PCB 上连接数字地 VSS。
PCI[1:0]_VDDQA	12	PCI-E 接口 1.8V/1.5V 接口模拟电源。	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。
PCI[1:0]_VDDA	12	PCI-E 接口 0.95V 核心模拟电源。	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_VDDA[2:0]	3	时钟接口 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。
---------------	---	-----------------	---

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。

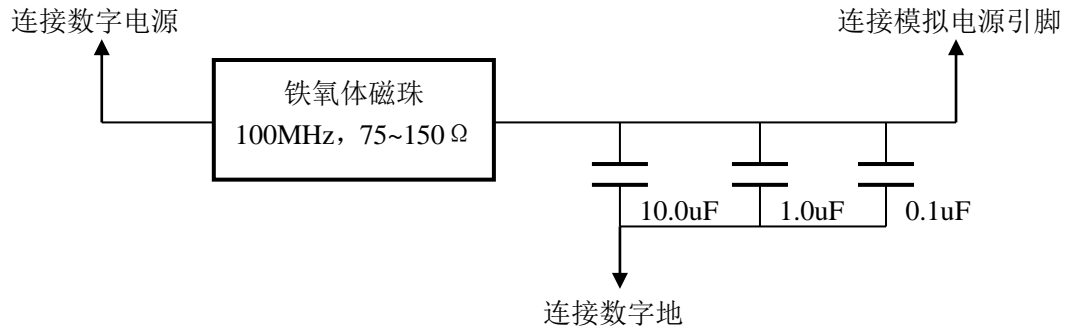


图 3-1: 连接模拟电源引脚的滤波器

3.3 特殊引脚

申威 421 处理器有一些特殊引脚，具体如表 3-3 所示。

表 3-3: 特殊引脚及其连接要求

引脚名称	数量	说明	连接方式
MM_ATO	1	DDR3 存储器接口模拟信号，用于 DDR3 PHY PLL 测试。	用于示波器观测；
MM.DTO[1:0]	2	DDR3 存储器接口数字信号，用于 DDR3 PHY PLL 测试。	用于示波器观测；
MM0[L/H]_VREF	2	DDR3 存储器接口参考电压。	连接带滤波的电阻分压网络，产生的参考电压为 VDDQ15 电源电压的 50%。
MM0[L/H]_PVT[2:0]	6	用于连接 DDR3 存储器接口补偿电阻。	分别连接 $240\Omega \pm 1\%$ 的高精度电阻，电阻另一端连接 VSS。
PCI[1:0]_RESREF	2	PCI-E 接口参考电阻。	参考电阻；每个 PCI-E 接口共用一根；连接 $200\Omega \pm 1\%$ 的高精度电阻，电阻另一端接数字地 VSS。

4 时钟接口

4.1 概述

申威 421 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。

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。

4.2 工作模式

申威 421 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 421 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 421 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，时钟频率与输入参考时钟 RCLK 频率相同。

5 存储器接口

5.1 概述

申威 421 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、16GB 或 32GB；
- 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、双 Rank 或四 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 UDIMM 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 存储器接口配置

5.2.1 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 为例进行说明，且图中的命令是针对 Rank0）。

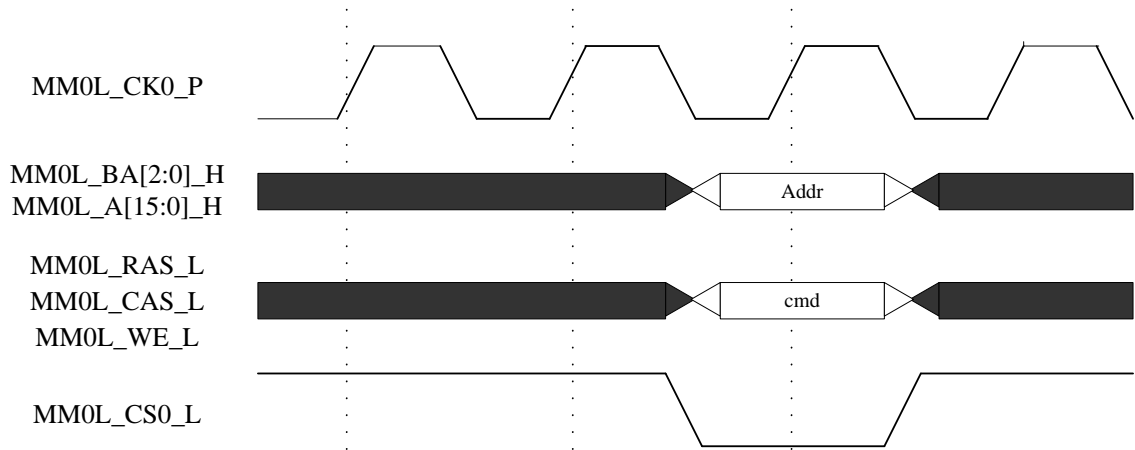


图 5-1: 1T 传输模式示意图

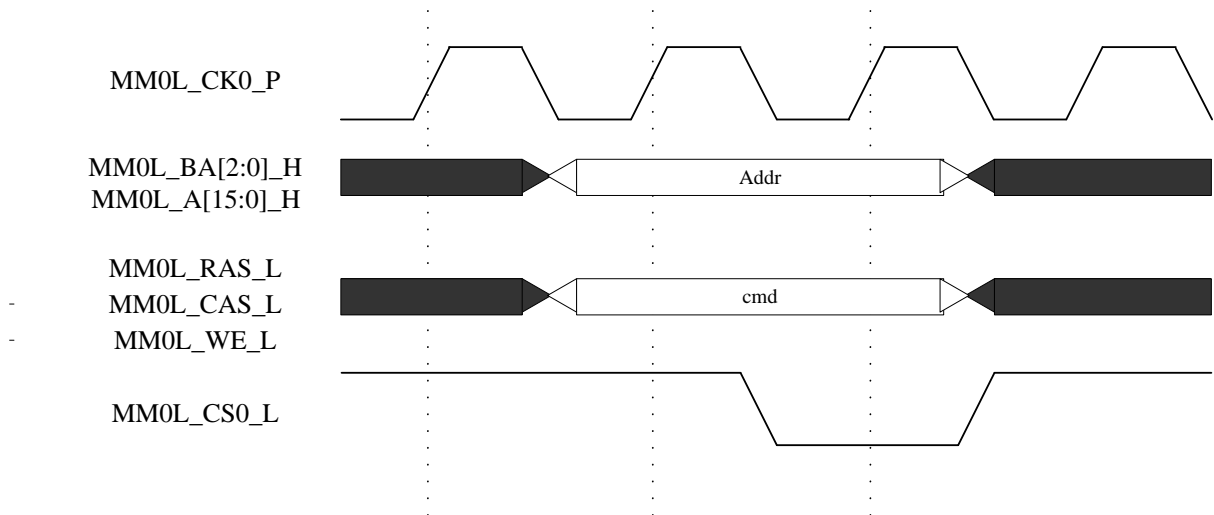


图 5-2: 2T 传输模式示意图

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。

5.2.2 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 421 处理器支持的地址镜像如表 5-1 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 5-1: 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
MM0L_A_H3	MM0L_A_H3	MM0L_A_H4
MM0L_A_H4	MM0L_A_H4	MM0L_A_H3
MM0L_A_H5	MM0L_A_H5	MM0L_A_H6
MM0L_A_H6	MM0L_A_H6	MM0L_A_H5
MM0L_A_H7	MM0L_A_H7	MM0L_A_H8
MM0L_A_H8	MM0L_A_H8	MM0L_A_H7
MM0L_BA_H0	MM0L_BA_H0	MM0L_BA_H1
MM0L_BA_H1	MM0L_BA_H1	MM0L_BA_H0

5.2.3 颗粒/内存类型

申威 421 处理器支持的存储容量及颗粒类型如表 5-2:

表 5-2: 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	四 Rank 容量
64bit	x8	1Gb	1GB	2GB	4GB
		2Gb	2GB	4GB	8GB
		4Gb	4GB	8GB	16GB
		8Gb	8GB	16GB	32GB
	x16	1Gb	---	1GB	2GB
		2Gb	1GB	2GB	4GB
		4Gb	2GB	4GB	8GB
		8Gb	4GB	8GB	16GB

5.3 信号连接

5.3.1 选通信号

申威 421 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、MM0L_DQS[8:0]_P, 以存储器接口 0 为例), 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 进行下拉, 对 MM0_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, 通过数据通用

配置寄存器 DXCCR[DQSNRES]控制 MM0_DQS[8:0]_L。

5.3.2 ECC 校验码

每个存储器接口数据宽度为 64 位，实际信号宽度为 72 位，其中 8 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 5-3 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 5-3: 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[L/H]_DQS[8]_P、 MM0[L/H]_DQS[8]_L	DIMM0/1.dqs[8]	8 位 ECC 校验位
MM0[L/H]_ECC_H[7:0]	DIMM0/1.cb[7:0]	
MM0[L/H]_DQS[7:0]_P、 MM0[L/H]_DQS[7:0]_L	DIMM0/1.dqs[7:0]	64 位数据
MM0[L/H]_DQ_H[63:0]	DIMM0/1.dq[63:0]	

5.3.3 双内存模组连接方式

为了支持大容量内存的需求，针对每路存储器接口可采用双内存模组的方式，表 5-4 给出了双内存模组插接方式。

表 5-4: 双内存模组插接方式说明

	DIMM0	DIMM1
配置 1	单 Rank UDIMM	单 Rank UDIMM
配置 2	双 Rank UDIMM	双 Rank UDIMM
配置 3	单 Rank UDIMM	无
配置 4	双 Rank UDIMM	无
配置 5	单 Rank RDIMM	单 Rank RDIMM
配置 6	双 Rank RDIMM	双 Rank RDIMM
配置 7	单 Rank RDIMM	无
配置 8	双 Rank RDIMM	无
配置 9	四 Rank RDIMM	Disable

5.4 初始化

申威 421 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 421 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。

5.5 测试

申威 421 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 421 处理器集成了两个符合 3.0 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 8Gbps，8 条链路并行传输的双向有效带宽可达到 16GB/s。

6.2 信号连接

申威 421 处理器集成符合 3.0 标准的 PCI-E 接口，该接口作为根端口（Root Complex），可以连接 Endpoint、SWITCH、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）。

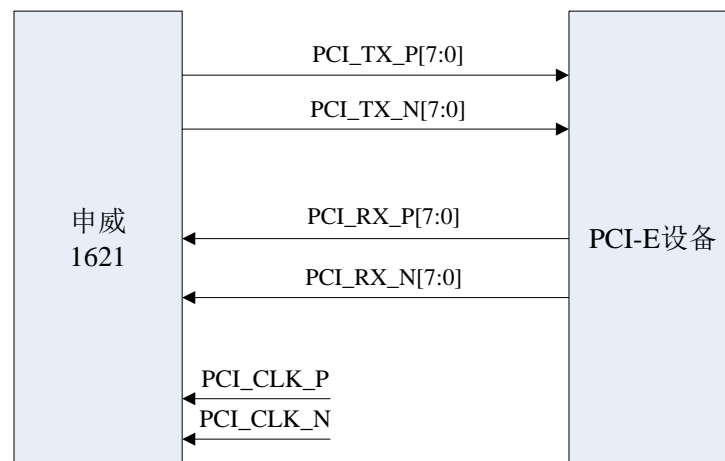


图 6-1: PCI-E 接口信号连接示意图

每个 PCI-E 接口包含 8 条链路（单链路带宽 8Gbps），链路宽度和速率可以进行自协商。PCI-E 接口向下兼容，可以连接支持 2.0 标准和 1.0 标准的 PCI-E 设备。

6.3 初始化

申威 421 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。

6.4 测试

申威 421 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 421 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 接口说明

7.1.2.1 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 421 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 421 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。

7.1.2.2 I2C 接口

引脚 SCL_M_H 和 SDA_M_H 是 I2C 接口。该 I2C 接口被用于读取存储器模组的 SPD (Serial Presence

Detect) 信息，实现申威421处理器存储控制器的自动配置。

7.1.2.3 SPI 串口

引脚 SS_L、SCK_H、MOSI_H_OUT 和 MISO_H_IN 是 SPI 接口。申威 421 处理器作为 SPI Master，连接支持 24 位地址的 SPI Flash。当配置引脚 CFG_SEL_H[1]为 1'b1 时，申威 421 处理器则基于 SPI Flash 实现自启动。

7.1.3 维护功能

7.1.3.1 故障监测

申威 421 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，维护接口会通过总错 (STAT_OUT_SEL_L 配置为 2'b00 下的 STAT_OUT_L[4]表示系统总错) 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。

7.1.3.2 调试支持

申威 421 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 初始化配置

申威 421 处理器分为初始化模式：

- 1) 芯片自引导模式（推荐模式）。此时 CFG_SEL_H[1]配置为 1'b1，芯片启动读取 Flash 的配置信息加载、存储器自测试和 ICache 加载程序，并启动运行（第九章）。
- 2) 交互配置方式（兼容老版本）。此时通过维护接口或 JTAG 接口配置内部寄存器，完成配置

信息加载和 ICache 加载程序，并最终启动运行（第九章）。

7.2.2 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。核心时钟配置引脚的具体定义如表 7-5 所示。

表 7-5: 核心时钟配置表

CFG_CORE_H[3:0]	核心时钟工作频率 (MHz)
0	旁路 (200MHz)
1	800
2	1000
3	1200
4	1400
5	1500
6	1600
7	1700
8	1750
9	1800
10	1850
11	1900
12	1950
13	2000
14	2050
15	2100

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。存储控制器时钟配置引脚的具体定义如表 7-6 所示。

表 7-6: 存储控制器时钟配置表

CFG_MM_H[2:0]	存控时钟频率 (MHz)
0	旁路 (200MHz)
1	266

2	300
3	333
4	366
5	400
6	433
7	466

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。互连时钟配置引脚的具体定义如表 7-7 所示。

表 7-7: 互连时钟配置表

CFG_XBX_H[2:0]	互连时钟工作频率 (MHz)
0	旁路 (200MHz)
1	800
2	1000
3	1050
4	1150
5	1250
6	1300
7	1333

7.3 中断接口

中断引脚的使用说明如表 7-9 所示。

表 7-9: 中断引脚使用说明

中断类型	配置寄存器	备注
WAKEUP_L	WAKEUP_CTL[MASK_EN]作为唤醒中断使能, 该位为“1”表示使能唤醒中断。 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。	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	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。	为防止抖动, 有效电平 (高电平或低电

	<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>	<p>平) 要至少维持 16 个维护时钟周期。</p>
--	---	-----------------------------

8 调试测试接口

8.1 概述

申威 421 处理器的调试测试接口由符合 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) 支持边界扫描测试;
- 2) 支持维护调试操作。

8.2 TAP 控制器

申威 421 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。

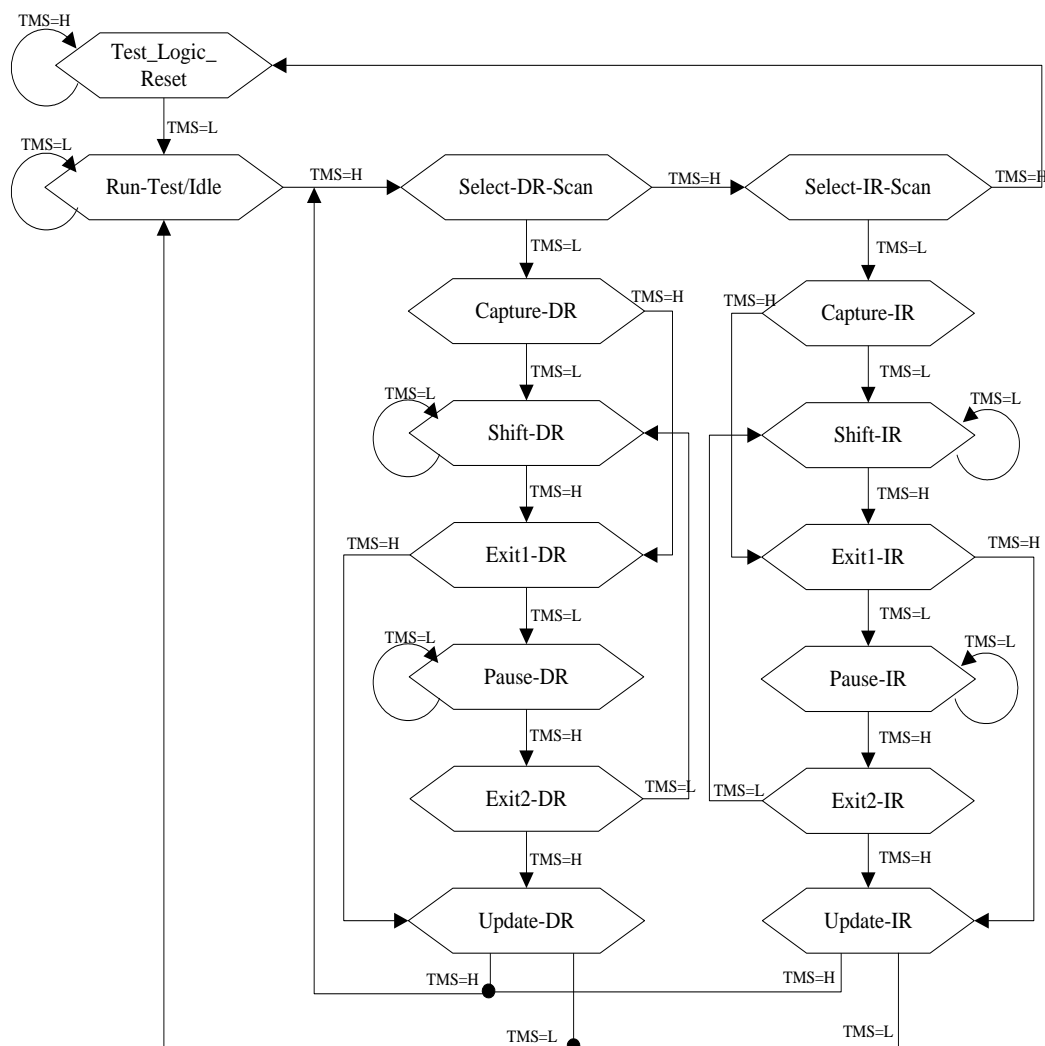


图 8-1: TAP 控制器状态机

8.3 测试寄存器

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 421 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 (二进制)	含义
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。

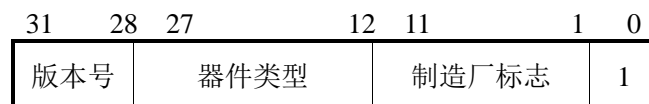


图 8-2: 标志寄存器

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 0110 1010”。
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	描述
ModuleID	[2:0]	表示芯片内各主要Module。 0: 表示GPIO BSR; 1: 表示PCI-E0; 2: 表示PCI-E1; 3: 表示DDR3PHY0/1 (CG0) ; 4: 表示DDR3PHY2/3 (CG1) ; 5: 表示DDR3PHY4/5 (CG2) ; 6: 表示DDR3PHY6/7 (CG3) ; 7: 表示MCU (JTAG可访问的MCU接口寄存器) ;
SubTypeID	[4:3]	当ModuleID为DDR3PHYx时有意义; 0: BSR; 1: JTAG; 2: ATPG。 其他保留。
ChainID	[8:6]	当 ModuleID 和 SubTypeID 表示是 DDR3PHY 的 ATPG测试时，表示ATPG的链号。其它ModuleID和SubTypeID，该域无意义。 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) 其他保留;
DDR3PHYID	[9]	当 ModuleID 表示 DDR3PHYx 时有意义，DDR3PHYID为零则表示DDR3PHY0，否则表示DDR3PHY1。
RSV	[11:10]	保留

8.4 边界扫描

申威 421 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 5 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）	59
PCI-E0 接口的边界扫描链（BSR）	24
PCI-E1 接口的边界扫描链（BSR）	24
DDR3 存储器接口 0 的边界扫描链（BSR）	150
DDR3 存储器接口 1 的边界扫描链（BSR）	150

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 421 处理器的 DDR3 的 IO 引脚都是双向引脚，申威 421 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 调试操作

申威 421 处理器支持基于 JTAG 接口的芯片调试以及运行监测。

8.5.1 调试接口寄存器

JTAG（TAP 控制器）通过接口寄存器与 MCU 进行交互。

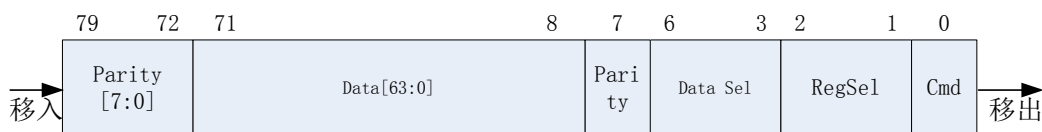


图 8-2：调试接口寄存器

- 1) 位宽：80bit。

- a) Cmd 表示操作类型，0 表示读，1 表示写；读写相对此次操作对应的寄存器（RegSel[1:0] 选择）。
 - b) RegSel[1:0]表示寄存器选择，0 表示命令寄存器，1 表示数据寄存器。2 或 3 表示状态寄存器；
 - c) DataSel[3:0]表示数据 Wrap 选择，只对写数据或读响应数据有意义，表示本次数据 Wrap 地址（按 64 位而言）。
 - d) Data[63:0]表示本次读写操作（Cmd）的内容，如果操作命令寄存器则该数据为命令本身。
 - e) Parity 表示偶校验位。其中 bit7 是 bit[6:0]的偶校验位，bit[79:72]是 bit[71:8]的偶校验位，每个校验位对应 8 位数据。
- 2) 串行扫入格式要求：按照字节 1（低位到高位）+字节 2（低位到高位）+……+字节 10（低位到高位）的顺序进行。
 - 3) 串行扫出格式要求：按照字节 1（低位到高位）+字节 2（低位到高位）+……+字节 10（低位到高位）的顺序进行。

8.5.2 调试操作流程

- 1) 测试地址选择
 - a) 对 TBOX 指令寄存器进行 Update，写入 CHAINADDR 命令；
 - b) 对 TBOX 数据寄存器进行 Update，由于当前指令寄存器是 CHAINADDR 命令，所以此移入的数据是 ScanAddrReg，选择调试接口寄存器；
 - c) 对 TBOX 指令寄存器进行 Update，写入 SAMPLE/PRELOD 或 EXTEST 命令；
 - d) 此时可以操作调试接口寄存器。
- 2) 调试接口状态寄存器获取。
 - a) 对调试接口寄存器进行 Update 操作（Cmd 为零表示读取操作，RegSel[1:0]选择状态寄存器作为读取对象，DataSel[3:0]为零，Data[63:0]为零），选择状态寄存器并进行读操作；
 - b) 等待 16 个 TCLK，用于异步交接（测试时钟和维护时钟之间两次异步交接的延时），以保证接口寄存器内容稳定。
 - c) 对调试接口寄存器进行 Capture 操作（此时调试接口寄存器数据域为状态寄存器内容），

得到状态信息，如果允许发送命令（表 8-6 的 STATE，下同）则发送命令，否则反复查询。

表 8-6: TBXI 模块状态寄存器

描述符	域	意义
MFSM	[19:16]	维护主状态机编码。
RES_LNT	[15:8]	响应长度。 0: 0 字节; 8: 8 字节; 128: 128 字节; 其它保留。
RES_CMD	[7:2]	响应编码[5:0]。
STATE	[1:0]	记录基于 JTAG 接口的维护请求处理状态。 2'b00: 空闲状态。 2'b01: 维护请求发送状态。 2'b10: 维护请求处理状态。 2'b11: 维护命令处理结束状态。 对于维护命令（扫入维护命令寄存器）必须查询该状态标志，等待该标志为空闲状态。

3) 调试操作读命令操作流程

- a) 对调试接口寄存器进行 Update 操作（Cmd 为“1”，RegSel[1:0]选择命令寄存器，Data[63:0]用于指定维护请求格式包，DataSel[3:0]为零），选择命令寄存器并进行写操作；
- b) 对调试接口寄存器进行 Update 操作（update 数据详见步骤 2），读取状态寄存器内容，如果响应已经准备好且有响应数据（STATE 为 2'b11，RES_CMD 表示响应类型、RES_LNT 表示响应长度），则准备得到本次读响应数据（子步骤 c~e）；
- c) 对接口寄存器进行 Update（Cmd 为“0”，RegSel[1:0]选择数据寄存器，DataSel[3:0]按长度从 0 开始选择 8B 数据，Data[63:0]无意义），选择数据寄存器并进行读操作；
- d) 等待 16 个 TCLK，用于异步交接（测试时钟和维护时钟之间两次异步交接的延时），

以保证接口寄存器内容稳定。

- e) 对调试接口寄存器进行 Capture (此时调试接口寄存器数据域为数据寄存器内容), 得到数据寄存器中响应数据; 如果响应数据为 1024 位, 则反复步骤 3 中的 c~e 子步骤。
- f) 上层软件根据维护响应长度指示得到全部的响应数据后, 对接口寄存器进行 Update 操作 (Cmd 为“1”, RegSel[1:0]选择状态寄存器, DataSel[3:0]无意义, Data[63:0]无意义), 选择状态寄存器并进行写操作, 该动作仅仅用于交互。

4) 调试操作写命令操作流程

- a) 对调试接口寄存器进行 Update 操作 (Cmd 为“1”, RegSel[1:0]选择命令寄存器, Data[63:0]用于指定维护请求格式包, DataSel[3:0]为零), 选择命令寄存器并进行写操作;
- b) 对调试接口寄存器进行 Update 操作 (Cmd 为“1”, RegSel[1:0]选择数据寄存器, Data[63:0]为写数据, DataSel[3:0]按长度从 0 开始选择 8B 数据), 选择数据寄存器进行写操作; 如果写数据为 128 位或 1024 位, 则反复该子步骤。
- c) 对调试接口寄存器进行 Update 操作 (update 数据详见步骤 2), 读取状态寄存器内容, 如果响应已经准备好 (STATE 为 2'b11, RES_CMD 表示响应类型、RES_LNT 一定为零), 此时已经得到响应;
- d) 对接口寄存器进行 Update 操作 (Cmd 为“1”, RegSel[1:0]选择状态寄存器, DataSel[3:0]无意义, Data[63:0]无意义), 选择状态寄存器并进行写操作, 该动作仅仅用于交互。

5) 维护操作注意事项:

- a) 系统通过读状态寄存器 STAT_REG[19:16]可确定维护主状态机所处的状态。系统可在配置 1 状态、配置 2 状态、配置 3 状态、初始化加载状态、运行状态下发送维护命令;
- b) 当要发送维护命令时, 系统通过 JTAG 端口读状态寄存器 STAT_REG[1:0], 如果此时 JTAG 接口的维护请求处理状态处在空闲状态, 则系统通过 JTAG 端口发送维护命令;
- c) 系统循环查询状态寄存器, 当确定状态寄存器 STAT_REG[1:0]=2'b11 时, 说明此时响应已经收齐, 根据 STAT_REG[15:2]判断命令类型和响应数据长度。无论是否有读响应数据, 系统都需要写状态寄存器, 通知状态机切换到空闲状态。

9 复位与初始化

申威 421 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 421 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 421 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 421 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 421 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 421 处理器通过复位状态机来控制各种复位的处理，外部系统通过读取内部寄存器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；

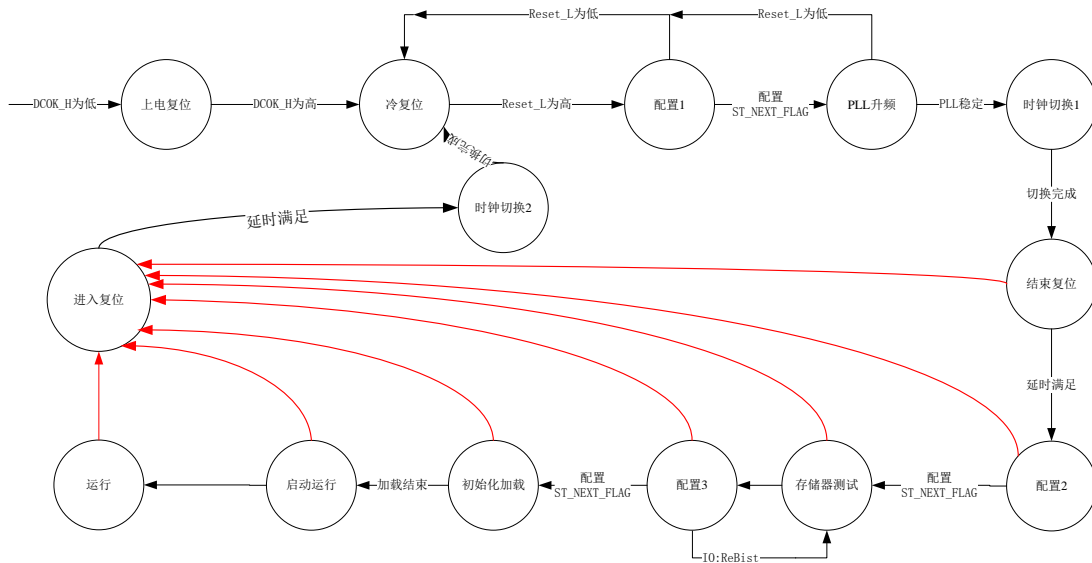


图 9-1: 复位状态机状态转换图

- 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): 如果配置 1 状态使能寄存器 CFG1_EN (INIT_CTL) 配置为零则直接转向 PLL 升频状态, 否则等待初始化相关的控制寄存器完成配置后, 通过写 “ST_NEXT_FLAG” 寄存器的维护命令, 则转向 PLL 升频状态。当引脚 CFG_SEL_L 为 “2'b1x” 时, 则自动读取 Flash 并配置初始化相关的控制寄存器后自动转入 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 1 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;
- 5) 时钟切换 1 (4'b0100): 按顺序进行工作时钟的高低频切换。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 421 处理器的复位, 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 CFG2_EN (INIT_CTL) 配置为零则成跳过此状态, 则直接转向存储器自测试状态, 否则外部先使用维护命令设置与存储器自测试相关的扫描链寄存器, 然后向申威 421 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;
- 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；当引脚 CFG_SEL_L 为“2'b1x”时，则自动读取 Flash 并完成 Icache 加载后自动转入启动运行状态；

- 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 配置引脚

申威 421 处理器的时钟接口和维护接口的部分引脚用于配置申威 421 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]	核心工作频率配置（7.2.3 节）。
CFG_MM_H[2:0]	存储控制器工作频率配置（7.2.3 节）。
CFG_XBX_H[2:0]	互连工作频率（7.2.3 节）。
CFG_SEL_H[1:0]	芯片配置说明。 “2'b00”：采用引脚配置方式，根据时钟控制引脚和初始化

	<p>模式引脚来设置初始化寄存器 INT_CTL，并可以通过维护命令进行配置修改。</p> <p>“2'b01”：采用寄存器默认值配置方式，即配置引脚无效，采用寄存器缺省值，并可以通过维护命令进行配置修改。</p> <p>“2'b1x”：芯片自启动方式，即自动读取 SPI Flash 获得配置参数及初始引导程序。</p>
--	---

9.2.2 配置寄存器

申威 421 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。

9.3 初始化

9.3.1 加关电顺序

为保证申威 421 处理器的正常工作，需要按照一定顺序完成对申威 421 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；
- 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 421 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。

注：要求在对 I/O 电源、内核电源加电同时提供维护时钟。

9.3.2 初始化流程

申威 421 处理器初始化流程具体如下：

1) 上电复位状态

该状态下的具体操作包括：

- a) 设置初始化模式 INIT_MODE_H[1:0]=2' h3、配置方式 CFG_SEL_L[1:0]=2' b10。
- b) 设置 DCOK_H=1' b0、Reset_L=1' b0；表示上电复位；
- c) 等待电源稳定后，设置 DCOK_H=1' b1，上电复位结束，进入冷复位状态。

2) 等待冷复位结束状态

该状态下的具体操作包括：

- a) 设置 Reset_L : 0 → 1，冷复位结束，进入配置 1 状态；
- b) 此状态 FlashCtrl (SPI Master 接口用于访问 SPI Flash) 已经结束复位。主状态机进入配置 1 状态时准备读取 Flash 信息。

3) 配置 1 状态

该状态下的具体操作包括（可配置为直接跳过该状态）：

- a) 读取 SPI Flash 信息，根据读取信息配置初始化控制和时钟配置寄存器 INIT_CTL，

PLL 升频等待计数器 PLL_CHG_CNT、PIUPLL_CNT, 处理器信息寄存器 r_CPUINFO 和 Flash 信息寄存器 FLASHINFO;

- b) 自启动模式 INIT_CTL 信息指示跳过配置 1 状态, 进入 PLL 升频状态。
- 4) PLL 升频状态
当 PLL 处于稳定工作状态, 进入时钟切换 1 状态
- 5) 时钟切换 1 状态
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;
- 6) 结束复位状态
暂时状态, 自动进入配置 2 状态;
- 7) 配置 2 状态
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;
- 9) 配置 3 状态
自启动模式 INIT_CTL 信息指示跳过配置 3 状态, 进入 SROM 加载状态。
- 10) SROM 加载状态
自启动模式启动读取 Flash 的加载数据, 加载结束后转入预备运行状态。
- 11) 预备运行状态
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, 如果进入运行状态。
- 12) 运行状态
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 421 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 421 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 421 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 421 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 421 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 421 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}	内核电源电压 V _{core}	0.91	1.05	1.07	V
V _{VDDQ15}	1.5V I/O 电源电压 V _{IO15}	1.425	1.5	1.575	V
	1.35V I/O 电源电压 V _{IO15}	1.283	1.35	1.45	V
	1.25V I/O 电源电压 V _{IO15}	1.19	1.25	1.31	V
V _{VDDQ18}	1.8V I/O 电源电压 V _{IO18}	1.71	1.8	1.89	V
V _{VREF}	1.5V I/O 参考电压 V _{REF15}	0.49*V _{VDDQ15}	0.5*V _{VDDQ15}	0.51*V _{VDDQ15}	V
T _J	结温	0		125	°C
T _A	环境温度	0 (商业级) -40 (工业级)		70 (商业级) 85 (工业级)	°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_{VDDQ15}, 其它信号为 V_{VDDQ18}。

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 _{ID}	输入差分电压	100		600	mV

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 _{VDDQ15}			V	1
V _{OL}	输出低电平电压			0.2*V _{VDDQ15}	V	1
t _{STEP}	输出调整一级延迟大小		5		ps	
R _{TT}	R _{TT} 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})	0.76	1.26	2.73	mW	3
P _{RCV1}	输入模式下直流功耗 (V _{VDD})	0.07	0.13	13.57	uW	3
P _{DRV0}	输出模式下直流功耗 (V _{VDD15})	0.76	1.26	2.73	mW	4
P _{DRV1}	输出模式下直流功耗 (V _{VDD})	0.06	0.11	13.33	uW	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)	1.62	1.8	1.98	V
V _{IMAX}	最大输入电压			1.98	V
V _{IH}	输入高电平	1.17		1.98	V
V _{IL}	输入低电平	-0.3		0.63	V
V _T	阈值点	0.81	0.89	0.97	V
V _{TPU}	带上拉电阻使能的阈值点	0.8	0.87	0.96	V
V _{TPD}	带下拉电阻使能的阈值点	0.82	0.89	0.98	V
R _{PU}	上拉电阻	47K	69K	106K	欧
R _{PD}	下拉电阻	49K	85K	159K	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)	12.9	28.4	50.5	mA
I _{OL}	低电平输出电流 (V _{OL} =0.45V)	16.5	26.8	37.1	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	45	—	55	%	
J _{max}	输入时钟 Jitter	—	—	1	%	
F _{req}	输入时钟频率	10	—	600	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 值	最大值	单位	说明
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	

P_{RCV0}	输入模式下交流功耗 (V_{VDD15})	0.37	0.37	0.45	uW/MHz	1,2
P_{RCV1}	输入模式下交流功耗 (V_{VDD})	0.06	0.07	0.09	uW/MHz	1,2
P_{DRV0}	输出模式下交流功耗 (V_{VDD15})	10.47	10.57	12.57	uW/MHz	2,3
P_{DRV1}	输出模式下交流功耗 (V_{VDD})	0.31	0.39	0.52	uW/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	偏斜	单位	说明
Skew_ck	差分时钟对儿之间的 skew	11	ps	1
Skew_cktoac	时钟和 AC (Address Command) 之间的 skew	30	ps	1
Skew_DQS	DQS 差分对儿之间的 skew	10	ps	1
Skew_x8_inter	X8 内的 skew	20	ps	1
Skew_x8_Cross	X8 间的 skew	33	ps	1

注 1: 该 skew 包括 Die 和封装两个方面。

10.3.3 维护接口信号

维护接口信号都是电源为 1.8V 的 LVCMOS 电平, 其交流参数如表 10-8 所示, 相应的时序见图 10-1。

表 10-8: 维护接口的交流参数

引脚名称	I/O	延迟 (ns)	说明	频率指标
MT_CLK_H	输入	0.576ns	Pin 到片内部触发器的延时。	25MHz
MT_TX_H	输出	3.194ns	内部触发器到 pin 的延时。	25MHz
MT_RX_H	输入	0.986ns	Pin 到片内部触发器的延时。	25MHz

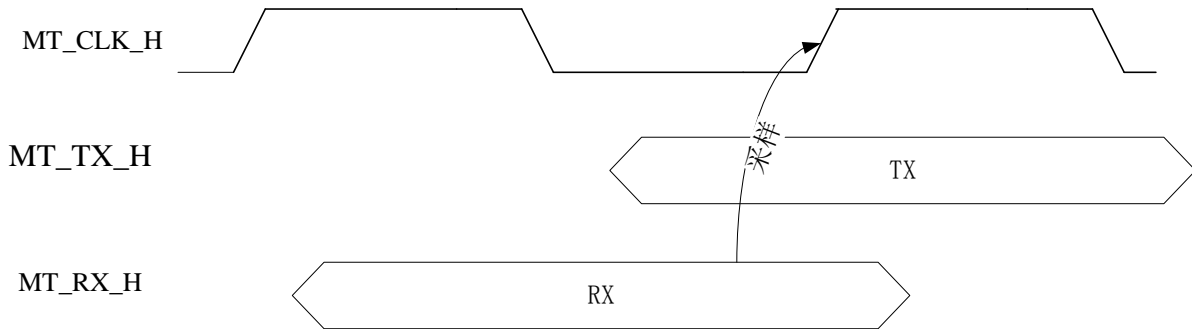


图 10-1: 维护接口延时图

申威 421 处理器在维护时钟 MT_CLK_H 的上升沿对串行输入 MT_RX_H 进行采样，在维护时钟 MT_CLK_H 的下降沿沿输出 MT_TX_H。根据图 10-1 时序和表 10-8 实际延时，并考虑一定的余量，建议板级设计要按 MT_RX_H 在处理器内部用时 3ns、MT_TX_H 在处理器内部用时 5ns 考虑，此外还有考虑板级的延时。为了减少低频信号跨片传输偏斜 (skew) 对 Hold 的影响，建议使用上升沿接收、下降沿输出进行容偏斜，即外部系统在下降沿输出 MT_RX_H、在上升沿接收 MT_TX_H。

SPI 接口信号都是电源为 1.8V 的 LVCMOS 电平，其交流参数如表 10-9 所示，相应的时序见图 10-2。

表 10-9: SPI 接口的交流参数

引脚名称	I/O	延迟 (ns)	说明	频率指标
SCK_H	输出	3.233ns	内部触发器到 pin 的延时	25MHz
SS_L	输出	3.249ns	内部触发器到 pin 的延时	25MHz
MISO_H_IN	输入	0.861ns	Pin 到内部触发器的延时	25MHz
MOSI_H_OUT	输出	3.263ns	内部触发器到 pin 的延时	25MHz

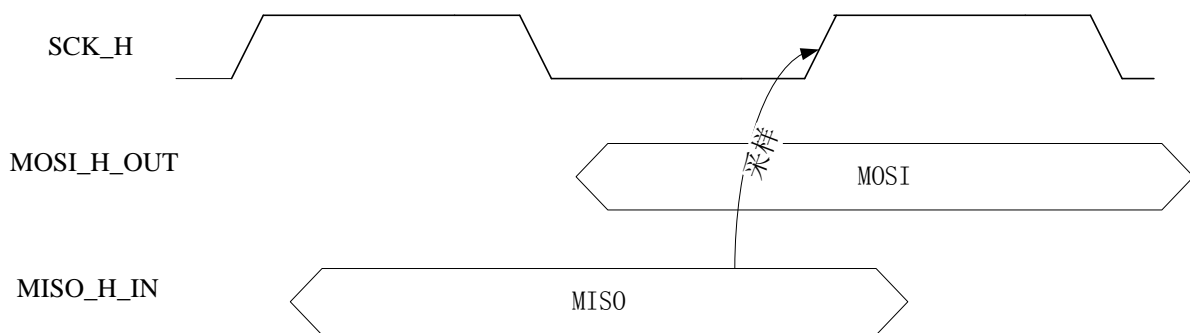


图 10-2: SPI 接口延时图

申威 421 处理器在 SCK_H 的上升沿对串行输入 MISO_H_IN 进行采样，在 SCK_H 的下降沿沿输出 MOSI_H_OUT。根据 SPI 的传输时序，采样 MISO_H_IN 要考虑处理器芯片与 Flash 芯片的往返延时。根据图 10-2 时序和表 10-9 实际延时，并考虑一定的余量，建议板级设计要按 MISO_H_IN 在处理器内部用时 10ns、MOSI_H_OUT 在处理器内部用时 5ns 考虑，此外还有考虑板级的延时。

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 10-10 所示。

表 10-10: 测试接口的交流参数

引脚名称	I/O	传输延时 (ns)	说明	频率指标
TCK_H	输入	0.462ns	Pin 到内部触发器的延时。	10MHz
TMS_H	输入	0.808ns	Pin 到内部触发器的延时。	10MHz
TRST_L	输入	1.118ns	Pin 到内部触发器的延时。	10MHz
TDI_H	输入	2.349ns	Pin 到内部触发器的延时。 TCK_H 时钟上升沿采样。	10MHz
TDO_H	输出	9.585ns	内部触发器到 Pin 的延时。 TCK_H 时钟下降沿输出。	10MHz

10.3.5 PCI-E 接口

申威 421 处理器的 PCI-E 接口符合 PCI-E 3.0 标准,其接口信号的交直流参数可参考《PCI Express base specification Version3.0》。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 10-4 所示。相关特性与要求如表 10-11 所示。

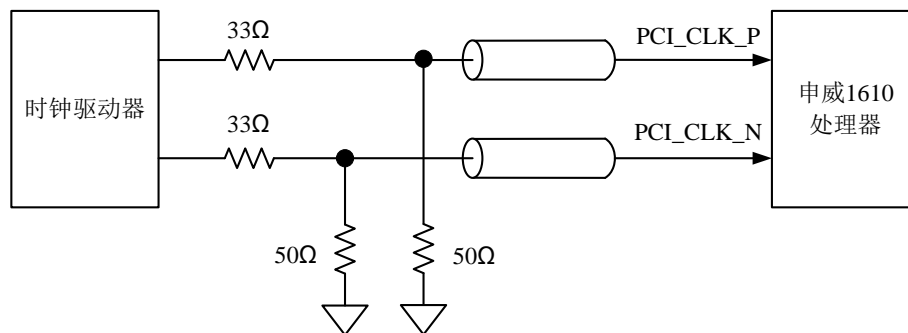


图 10-4: PCI-E 接口差分时钟输入连接参考电路

表 10-11: 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

申威 421 处理器封装的散热特性如表 10-12 所示。

表 10-12: 申威 421 处理器封装的热阻参数

风速 V(m/s)	θ_{ja} (°C/W)	θ_{jc} (°C/W)
0.0	1.958	7.3E-02
1.0	0.683	
2.5	0.416	

说明:

θ_{ja} : 芯片内硅片到芯片工作环境之间的热传导系数;

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=芯片壳温+ θ_{jc} *功耗。

备注: 封装金属盖中集成了铜镍材质的散热器。该数据是带外置散热片条件下的仿真结果。

10.4.2 热设计功耗

建议的申威 421 处理器热设计功耗如下:

- 1) 2.0GHz 核心工作频率下, 热设计功耗 (TDP) 为 35W, 典型运行功耗 25W;
- 2) 1.8GHz 核心工作频率下, 热设计功耗 (TDP) 为 32W, 典型运行功耗 23W;
- 3) 1.6GHz 核心工作频率下, 热设计功耗 (TDP) 为 29W, 典型运行功耗 20W;
- 4) 1.4GHz 核心工作频率下, 热设计功耗 (TDP) 为 26W, 典型运行功耗 18W;

10.4.3 功耗管理

申威 421 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 421 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 421 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 421 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 421 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 421 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	[7:6]	[5:0]	备注
1	维护读存储器	固定为 2'b11	6'b000000	带 CC（一致性处理标志）的存储器读（每次读 128 字节）。
2			6'b000100	不带 CC（一致性处理标志）的存储器读（每次读 128 字节）。
3	维护写存储器		6'b100000	带 CC（一致性处理标志）的存储器写（写数据 16 字节）。

4		6'b100101	带 CC（一致性处理标志）的存储器写（写数据 128 字节）。
5		6'b101000	不带 CC（一致性处理标志）的存储器写（写数据 16 字节）。
6		6'b101001	不带 CC（一致性处理标志）的存储器写（写数据 128 字节）。
7	维护接口复位	6'b110000	维护接口复位。
8	维护读寄存器	6'b001000	每次读 8 字节。
9	维护写寄存器	6'b100100	每次写 8 字节。
10	扫出或监测状态	6'b001100	每次扫出 128 字节。
11	扫入状态	6'b101101	串行扫入（16 字节）。
12		6'b101111	串行扫入（128 字节）。
13	初始化加载	6'b111100	初始化加载（128 字节）。
14		6'b111111	初始化加载结束。

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	1000	存储控制器中 I/O 空间。
3	1100	PCI-E 0 接口中 I/O 空间。
4	1101	维护接口中 I/O 空间。
5	1110	系统接口部件中 I/O 空间。
6	1111	PCI-E 1 接口中 I/O 空间。
	其它	保留

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 421 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 4 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	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	数据有效位[7:0]	
字节 7	8'b00000000	
字节 8	最高 4 位为全“0”，最低 4 位指示中断的目标核心，	
字节 9	最高 6 位为全“0”，最低 2 位指示中断类型编码	
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.1.4.2 存储器读

存储器读命令是对申威 421 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	2'b11	6'b000000/000100 (带 CC 标志/不带 CC 标志)
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

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 421 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	2'b11	6'b100000/100101/101000/101001 (带 CC 标志字节写/带 CC 标志整块写/ 不带 CC 标志字节写/不带 CC 标志整块写)
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 0
	
字节 7+n		数据 n-1 (n=16 或 128)

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 421 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 421 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/ 3'h1/ 3'h2/ 3'h3	CG0/ CG1/ CG2/ CG3	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)	<table border="1" style="display: inline-table; vertical-align: middle;"> <tr> <td>CPM([17]=0)</td> <td>1KB, 即地址[9:0]</td> </tr> <tr> <td>TCDATA([17]=1)</td> <td>2KB, 即地址[10:0]</td> </tr> </table>	CPM([17]=0)	1KB, 即地址[9:0]	TCDATA([17]=1)	2KB, 即地址[10:0]
	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]					
		保留([20:18]=3'b111)						
3'h4	CLU		32B, 即地址[4:0]					
3'h5	ION& INTPU	INTPU([20:19]=2'b0x)	1KB, 即地址[9:0]					
		ION0([20:19]=2'b10)	1KB, 即地址[9:0]					
		ION1([20:19]=2'b11)	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 421 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）。

初始化加载命令的地址（初始化加载结束命令地址没有任何意义）仅作地址检查使用，并不影响申威处理器内部生产的 ICache 加载地址。芯片内部的地址检查仅针对 Srom 加载的“重传”和“漏传”。**芯片默认 Srom 加载地址从 0x0 地址开始，每个 Cache 块加 0x80，且必须按顺序加载。**申威处理器内部实际加载时会根据地址判断该顺序，如果加载地址与期望地址不同（加载地址小于期望地址意味着维护系统发生“重传”，加载地址大于期望地址意味着维护系统发生“漏传”），则发生“重传”或“漏传”，两种情况仍然返回正常响应（从 5A FPGA 模块的情况看，主要是命令“重传”，**注意不能返回错误响应，否则会反复“重传”。**申威处理器会对接收到的 Srom 加载命令数量、实际加载 Icache 的命令数量和维护接口复位命令数量进行计数，并设置只读寄存器，共维护 and 核心读取。

注 1: 如果芯片内部检查到进入芯片的维护命令包有错（校验错或格式错等等），则丢弃该包不加载，且加载地址保持不变，并向芯片外部返回错误响应，由外部进行重传。重传的最大次数由芯片外部确定。如果芯片内部正常完成命令包的加载，加载地址递增，并向芯片外部返回正常响应。

注 2: 对于判断出的“重传”或“漏传”情况，只要该命令包没有检查到校验错或格式错，都会向芯片外部返回正常响应。

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 421 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 421 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 421 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.1.4.11 Flash 块擦除

表 A-17: 块擦除

字节 0	11	6'b100100 (4K/32K/64K 字节擦除)
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
字节 8		数据 0
	
字节 7+n		数据 n-1(n=8)

维护命令携带的地址表示 I0 寄存器地址，4K/32K/64K 字节擦除分别对应三个寄存器，写数据对应 4K/32K/64K 字节擦除所对应的 Flash 地址（该地址是 Flash 域地址，**假设 Flash 的第 2 个 8B 位，即 Flash 域地址 0x8，该写数据内容就为 0x8**）。写相应的三个寄存器将触发 4K/32K/64K 字节擦除操作。擦除后，该 Flash 块内容为全 1。

A.1.4.11 Flash 全片擦除

表 A-18: Flash 全片擦除

字节 0	11	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		00000000
字节 7		00000000
字节 8		数据 0

字节 7+n	数据 n-1(n=8)

维护命令携带的地址表示 I0 寄存器地址，全片擦除对应一个寄存器，写数据没有意义（全片擦除不需要 Flash 地址）。写该寄存器将触发全片擦除操作。擦除后，Flash 内容为全 1。

A.2 维护响应

A.2.1 维护响应包格式

申威 421 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-19。

表 A-19：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 421 支持 11 种维护响应，具体的响应编码如表 A-20 所示。

表 A-20：维护响应编码及含义

编号	维护响应	编码[7:6]	编码[5:0]	备注
1	带数据读响应 (正常读)	固定为 2'b11	6'b000000	带数据（1024 位或 64 位），具体如下： 读存储器：1024 位； 读寄存器是：64 位； 读监测信息或扫出：1024 位。
2	带数据读响应 (含有 ECC 多错)		6'b001000	
3	带数据读响应 (含有控制错)		6'b010000	
4	非法地址读响应		6'b000001	不带数据
5	写结束		6'b000010	
6	非法地址写结束		6'b000011	

7	带控制错写结束		6'b001011	
8	串行维护接口偶校验 错响应		6'b000100	
9	串行维护接口非法维 护命令响应		6'b000101	
10	对应接口关闭错误响 应		6'b100000	
11	串行处理错误响应		6'b101011	
	保留		其它	

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-21。

表 A-21：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-22。

表 A-22：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
59	INIT_MODE_H<0>	in	
58	INIT_MODE_H<1>	in	
57	MT_TX_H	out	
56	MT_RX_H	in	
55	STAT_OUT_L<0>	out	
54	STAT_OUT_SEL_L<0>	in	
53	STAT_OUT_L<1>	out	
52	STAT_OUT_SEL_L<1>	in	
51	STAT_OUT_L<2>	out	
50	STAT_OUT_L<3>	out	
49	RESET_L	in	
48	STAT_OUT_L<4>	out	
47	NMI_L	in	
46	WAKEUP_L	in	
45	SS_N	in	
44	SCK_H	in	
43	NC_Spare1_EN	En	
42	NC_Spare1	Inout-in	
41	NC_Spare1	Inout-out	
40	NC_Spare2_EN	En	
39	NC_Spare2	Inout-in	
38	NC_Spare2	Inout-out	
37	SDA_OUT_EN	En	
36	SDA	Inout-in	
35	SDA	Inout-out	

34	SCL_OUT_EN	En	
33	SCL	Inout-in	
32	SCL	Inout-out	
31	NC_Spare3_EN	En	
30	NC_Spare3	Inout-in	
29	NC_Spare3	Inout-out	
28	MISO_H_IN	In	
27	MOSI_H_OUT	out	
26	MT_CLK_H	in	
25	RSV	RSV	
24	RSV	RSV	
23	RSV	RSV	
22	NC_Spare6_EN	En	
21	NC_Spare6	Inout-in	
20	NC_Spare6	Inout-out	
19	CFG_SEL_L<1>	in	
18	CFG_XBX_H<2>	in	
17	CFG_XBX_H<1>	in	
16	CFG_XBX_H<0>	in	
15	CFG_SEL_L<0>	in	
14	CFG_MM_H<2>	in	
13	DCOK_H	in	
12	CFG_MM_H<1>	in	
11	CFG_MM_H<0>	in	
10	NC_Spare5_EN	En	
9	NC_Spare5	Inout-in	
8	NC_Spare5	Inout-out	
7	CFG_CORE_H<3>	in	
6	CFG_CORE_H<2>	in	
5	CFG_CORE_H<1>	in	
4	CFG_CORE_H<0>	in	
3	NC_Spare4_EN	En	
2	NC_Spare4	Inout-in	

1	NC_Spare4	Inout-out	
---	-----------	-----------	--

B.2 PCI-E BSR 链说明

申威 421 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
24	PCIEx_tx_p[0]	输出	差分输出。
23	PCIEx_rx_p[0]	输入	差分输入。
22	PCIEx_rx_m[0]	输入	差分输入。
21	PCIEx_tx_p[1]	输出	差分输出。
20	PCIEx_rx_p[1]	输入	差分输入。
19	PCIEx_rx_m[1]	输入	差分输入。
18	PCIEx_tx_p[2]	输出	差分输出。
17	PCIEx_rx_p[2]	输入	差分输入。
16	PCIEx_rx_m[2]	输入	差分输入。
15	PCIEx_tx_p[3]	输出	差分输出。
14	PCIEx_rx_p[3]	输入	差分输入。
13	PCIEx_rx_m[3]	输入	差分输入。
12	PCIEx_tx_p[4]	输出	差分输出。
11	PCIEx_rx_p[4]	输入	差分输入。
10	PCIEx_rx_m[4]	输入	差分输入。
9	PCIEx_tx_p[5]	输出	差分输出。
8	PCIEx_rx_p[5]	输入	差分输入。
7	PCIEx_rx_m[5]	输入	差分输入。
6	PCIEx_tx_p[6]	输出	差分输出。
5	PCIEx_rx_p[6]	输入	差分输入。
4	PCIEx_rx_m[6]	输入	差分输入。
3	PCIEx_tx_p[7]	输出	差分输出。
2	PCIEx_rx_p[7]	输入	差分输入。

1	PCIEx_rx_m[7]	输入	差分输入。
---	---------------	----	-------

B.3 DDR3 BSR 链说明

申威 421 集成了两路存储控制器。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
150	dto[0]	输出	DTO 引脚。
149	dto[1]	输出	DTO 引脚。
148	ram_rst_n	输出	复位信号。
147	ck[0]	输出	时钟。
146	ck[1]	输出	时钟。
145	ck_n[0]	输出	时钟。
144	ck_n[1]	输出	时钟。
143	cke[0]	输出	CKE。
142	cke[1]	输出	CKE。
141	cke[2]	输出	CKE。
140	cke[3]	输出	CKE。
139	odt[0]	输出	ODT。
138	odt[1]	输出	ODT。
137	odt[2]	输出	ODT。
136	odt[3]	输出	ODT。
135	cs_n[0]	输出	CS_N。
134	cs_n[1]	输出	CS_N。
133	cs_n[2]	输出	CS_N。
132	cs_n[3]	输出	CS_N。
131	ras_n	输出	RAS_N。
130	cas_n	输出	CAS_N。
129	we_n	输出	WE_N。
128	ba[0]	输出	BA 地址。
127	ba[1]	输出	BA 地址。
126	ba[2]	输出	BA 地址。

125	a[0]	输出	A 地址。
124	a[1]	输出	A 地址。
123	a[2]	输出	A 地址。
122	a[3]	输出	A 地址。
121	a[4]	输出	A 地址。
120	a[5]	输出	A 地址。
119	a[6]	输出	A 地址。
118	a[7]	输出	A 地址。
117	a[8]	输出	A 地址。
116	a[9]	输出	A 地址。
115	a[10]	输出	A 地址。
114	a[11]	输出	A 地址。
113	a[12]	输出	A 地址。
112	a[13]	输出	A 地址。
111	a[14]	输出	A 地址。
110	a[15]	输出	A 地址。
109	error_out	输入	error_out#
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。