

Logos2 系列 FPGA 配置（configuration） 用户指南

(UG040005, V1.3)

（2023.4.24）

深圳市紫光同创电子有限公司

版权所有 侵权必究

文档版本修订记录

版本号	发布日期	修订记录
V1.0	2019/12/15	初始版本
V1.1	2020/3/19	<ol style="list-style-type: none"> 1. 修改位流加密中软件设置。 2. 修改图 22 Slave Parallel 时序，CS_N 拉低前需要至少 8 个时钟。 3. 修改图 23，去掉 BUSY 信号。 4. 修改Slave Parallel 接口回读描述，新增回读流程的指令。 5. 修改表 5 SPI FLASH 支持列表。 6. 链接SEU IP。 7. 新增图 25 并行菊花链连接图。 8. 新增表 4 配置管脚定义。 9. 删除 BPI 模式。 10. 修改内部从并行接口“RST N”信号、UID 接口参数“UDID WIDTH”备注仅用于仿真。 11. 修改表 1、2，新增备注，MODE 不可设置为 3'b000。 12. 增加描述，JTAG 最高速率在不同配置电平标准下不同。 13. 修改 Slave Serial、Slave Parallel 时钟要求的描述。
V1.2	2020/5/20	<ol style="list-style-type: none"> 1. 修改配置速率描述。 2. 增加SCBV 软硬件设置描述。
V1.3	2023/4/24	<ol style="list-style-type: none"> 1. 新增eFuse 功能位锁定描述。 2. 新增从并非连续加载描述。 3. 新增从串级联时序要求描述。 4. 更新图 2-12、图 2-16、图 2-23，新增 INIT_FLAG_N、RSTN、CFG_DONE、MODE 的连接。 5. 更新图 3-3 唤醒时序，删除图中时钟信号，新增唤醒时序设置描述。 6. 更新表 2-4 SCBV 描述。 7. 新增JTAG 边界扫描结构介绍。 8. 更新表 2-4 中“第…位”描述，统一为“D[…]”；以及 D[3:2]连接描述。 9. 更新表 2-4 中 MODE、IO_STATUS_C、CFG_DONE 管脚上下拉描述。 10. 更新多启动回退条件，新增认证失败。

名词术语解释

Abbreviations 缩略语	Full Spelling 英文全拼	Chinese Explanation 中文解释
JTAG	Joint Test Action Group	联合测试工作组
SED	Soft Error Detection	软错误检测
CCS	Configuration Control System	配置控制系统
SPI	Serial Peripheral Interface	串行外围接口
AES	Advanced Encryption Standard	高级加密标准
SEU	Single Event Upsets	单粒子翻转
ECC	Error Control Coding	差错控制编码
SECDED	Single Error Correcting and Double Error Detecting	单比特纠错双比特检错
UID	Unique ID	唯一的 ID
TAPC	Test Access Port Controler	测试访问端口控制器
IR	Instruction Register	指令寄存器
TDR	Test Data Register	测试数据寄存器

目 录

文档版本修订记录	1
名词术语解释	2
目 录	3
表目录	6
图目录	8
1 概述	10
2 配置模式描述	11
2.1 JTAG 配置模式	16
2.1.1 JTAG 级联模式	17
2.1.2 JTAG 边界扫描结构	17
2.2 Master SPI 配置模式	18
2.2.1 Master SPI 串行菊花链	22
2.3 Slave Serial 配置模式	23
2.3.1 Slave Serial 串行菊花链	25
2.4 Slave Parallel 配置模式	26
2.4.1 Slave Parallel 并行菊花链	30
3 下载配置	31
3.1 配置过程	31
3.1.1 建立	31
3.1.2 位流加载	32
3.1.3 唤醒	35
3.2 配置文件	37
3.2.1 位流生成	37
3.2.2 位流大小	39
3.2.3 不同接口位宽下的位序	39
3.3 配置数据包	40
3.3.1 包格式	40

3.3.2 配置寄存器.....	40
3.4 位流格式.....	49
3.5 下载线速率.....	51
3.6 配置速率.....	52
3.6.1 JTAG 模式.....	52
3.6.2 Master SPI 模式.....	52
3.6.3 Slave Parallel 模式.....	52
3.6.4 Slave Serial 模式.....	53
3.7 多启动.....	53
3.7.1 黄金位流初始化系统.....	53
3.7.2 应用位流初始化系统.....	54
3.7.3 无黄金位流.....	55
3.7.4 热启动.....	56
3.7.5 看门狗.....	57
4 回读操作.....	58
4.1 JTAG 接口回读.....	58
4.2 Slave Parallel 接口回读.....	58
4.2.1 回读配置存储器.....	58
4.2.2 回读配置寄存器.....	59
5 SEU 检测.....	61
6 设计保护.....	62
6.1 位流加密.....	62
6.1.1 不带位流加密保护的场景.....	62
6.1.2 带位流加密保护的场景.....	62
6.1.3 加密算法.....	63
6.1.4 位流加密保护的使用流程.....	63
6.2 回读保护.....	65
6.3 位流认证.....	65
6.4 DPA 保护.....	66

6.5 UID	67
6.6 JTAG 安全管理	67
6.7 eFuse.....	67
7 用户逻辑接口	69
7.1 内部从并行接口	69
7.1.1 端口列表.....	69
7.1.2 参数定义.....	70
7.1.3 接口时序	70
7.1.4 例化模板.....	70
7.2 UID 接口	71
7.2.1 端口列表.....	71
7.2.2 参数列表.....	72
7.2.3 接口时序.....	72
7.2.4 例化模板.....	72
7.3 用户 JTAG 接口	73
7.3.1 端口列表.....	73
7.3.2 参数列表.....	73
7.3.3 例化模板.....	73
7.4 用户 Fuse 接口	74
7.4.1 端口列表.....	74
7.4.2 参数列表.....	74
7.4.3 例化模板.....	74
免责声明	75

表目录

表 2-1 Logos2 系列 FPGA 配置模式.....	11
表 2-2 串行模式配置管脚.....	11
表 2-3 并行模式配置管脚.....	12
表 2-4 配置管脚功能定义.....	13
表 2-5 JTAG 常用指令集.....	18
表 2-6 Master SPI 接口信号描述.....	18
表 2-7 支持的 SPI Flash 型号.....	22
表 2-8 Slave Serial 配置接口信号描述.....	23
表 2-9 Slave Parallel 配置接口信号描述.....	27
表 3-1 GOUTEN 逻辑功能.....	36
表 3-2 PDS 配置文件描述.....	37
表 3-3 Logos2 系列各器件的位流大小.....	39
表 3-4 不同位宽下同步字发送顺序.....	39
表 3-5 类型 1 包头格式.....	40
表 3-6 类型 2 包头格式.....	40
表 3-7 配置寄存器说明.....	40
表 3-8 Logos2 系列 FPGA 器件型号.....	41
表 3-9 CCS 命令及其描述.....	41
表 3-10 控制寄存器 0 说明.....	42
表 3-11 控制寄存器 1 说明.....	43
表 3-12 选项寄存器 0 说明.....	43
表 3-13 选择寄存器 1 说明.....	44
表 3-14 状态寄存器说明.....	45
表 3-15 看门狗寄存器说明.....	46
表 3-16 帧地址寄存器说明.....	46
表 3-17 SBPI 寄存器说明.....	46
表 3-18 SEU 控制寄存器说明.....	47

表 3-19 SEU 状态寄存器说明	47
表 3-20 SEU 帧地址寄存器说明	47
表 3-21 SEU 下帧地址寄存器说明.....	48
表 3-22 热启动控制寄存器说明	48
表 3-23 热启动地址寄存器说明	48
表 3-24 历史状态寄存器说明	48
表 3-25 普通位流格式.....	49
表 3-26 热启动流程.....	56
表 4-1 回读配置存储器流程	58
表 4-2 回读配置寄存器流程	59
表 7-1 GTP_IPAL_E2 端口列表.....	69
表 7-2 GTP_IPAL_E2 参数列表.....	70
表 7-3 GTP_UDID 端口列表	71
表 7-4 GTP_UDID 参数列表	72
表 7-5 GTP_JTAGIF 端口列表	73
表 7-6 GTP_JTAGIF 参数列表	73
表 7-7 GTP_EFUSECODE 端口列表	74
表 7-8 GTP_EFUSECODE 参数列表	74

图目录

图 2-1 JTAG 模式应用示意图.....	16
图 2-2 JTAG 编程典型时序图.....	16
图 2-3 JTAG 级联应用示意图.....	17
图 2-4 JTAG 边界扫描系统框图.....	17
图 2-5 Master SPI 配置接口	18
图 2-6 Master SPI 配置模式 X1 应用示意图	20
图 2-7 Master SPI 配置模式 X2 应用示意图	20
图 2-8 Master SPI 配置模式 X4 应用示意图	21
图 2-9 Master SPI 配置模式 X8 应用示意图	21
图 2-10 1bit Master SPI 编程典型时序图	21
图 2-11 2/4bit Master SPI 编程典型时序图	22
图 2-12 Master SPI 串行菊花链应用示意图	23
图 2-13 Slave Serial 配置接口	23
图 2-14 Slave Serial 模式应用示意图	24
图 2-15 Slave Serial 编程典型时序图	25
图 2-16 Slave Serial 配置模式串行菊花链连接	25
图 2-17 调整时序举例-后级	26
图 2-18 Slave Parallel 配置接口	26
图 2-19 Slave Parallel 模式应用示意图	28
图 2-20 Slave Parallel 配置时序图	29
图 2-21 Slave Parallel 非连续配置时序图	29
图 2-22 Slave Parallel 回读时序图	30
图 2-23 Slave Parallel 配置模式并行菊花链连接	30
图 3-1 下载配置流程图	31
图 3-2 下载配置时序	31
图 3-3 唤醒时序	36
图 3-4 生成位流示意图	38

图 3-5 Configure Generate Bitstream	38
图 3-6 Generate Flash Programing File	39
图 3-7 Fabric Configuration	51
图 3-8 Connect To Cable	52
图 3-9 USB Cable 速率设置	52
图 3-10 黄金位流加载流程	54
图 3-11 应用位流加载流程	55
图 3-12 无黄金位流时加载流程	56
图 6-1 无位流加密场景	62
图 6-2 位流加密场景	62
图 6-3 Configure 界面	63
图 6-4 Configure 设置选项	64
图 6-5 回读禁止设置	65
图 6-6 认证过程示意图	66
图 7-1 GTP_IPAL_E2 端口图	69
图 7-2 GTP_IPAL_E2 内部从并行接口读写接口时序	70
图 7-3 读 UID 时序	72
图 7-4 扩展 UID 位宽	72
图 7-5 GTP_EFUSECODE 端口图	74

1 概述

配置（configuration）是把用户的设计数据（位流）写入FPGA的内部存储器中的过程。配置数据可以由芯片主动从外部Flash获取，也可通过外部处理器/控制器将配置数据下载到芯片中。Logos2系列FPGA使用SRAM单元存储配置数据，掉电后配置数据丢失，所以每次上电需要重新对FPGA进行配置。

Logos2系列FPGA支持如下四种配置模式：

- JTAG模式，符合IEEE 1149.1、IEEE 1149.6和IEEE 1532标准
- Master SPI模式，支持数据位宽1/2/4/8bit
- Slave Parallel模式，支持数据位宽8/16/32bit
- Slave Serial模式，支持数据位宽1bit

此外，Logos2系列FPGA还提供以下功能：

- 看门狗，支持超时检测
- 支持通过内部从并行接口进行SEU 1-bit纠错和2-bit检错
- 在Master SPI模式下，支持配置位流版本回退功能
- 配置位流压缩可有效减小位流的大小、存储空间和编程时间
- 配置位流加密可防止恶意抄袭，有效保护客户设计知识产权
- JTAG模式提供专用接口，支持在线调试和边界扫描测试
- 每个器件在出厂前被写入与之对应的唯一编码 — 96bits的UID保护
- 支持SHA摘要、RSA-2048认证、AES256-GCM自认证对位流进行数字签名校验，验证位流的完整性
- 密钥存储方式支持eFuse和BB-RAM，其中BB-RAM可提供芯片级的安全防护
- 支持防位流反向读取的安全防护技术
- 支持JTAG安全管理，可永久或临时关闭JTAG功能
- 支持防DPA攻击，防止加密密钥被黑客破解

2 配置模式描述

用户可以通过设置MODE[2:0]的值来选择Logos2系列FPGA的配置模式，如表 2-1所示。其中JTAG配置模式优先级最高，MODE[2:0]设置为任意值（除了3'b000）都能进行配置。配置时钟CFG_CLK的方向由配置模式决定，主模式时，CFG_CLK为输出，由FPGA输出给外部存放位流的设备，如Flash；从模式时，CFG_CLK为输入，由外部设备（如微处理器，CPLD或其他FPGA）输出给FPGA。

表 2-1 Logos2 系列 FPGA 配置模式

序号	配置模式	MODE[2:0]	数据位宽	CFG_CLK 方向
1	JTAG	xxx ⁽¹⁾	1	输入（TCK）
2	Master SPI	001	1,2,4,8	输出
3	Slave Parallel	110	8,16,32	输入
4	Slave Serial	111	1	输入

注：

（1）禁止设置 MODE[2:0]=3'b000，此模式任何接口均无法使用。除了 3'b000 之外，设置为任意值均可使用 JTAG 模式，若只需要使用 JTAG 模式，可设置为 MODE[2:0]=3'b101，该设置下仅 JTAG 可用。

不同配置模式使用的管脚如表 2-2所示，有些管脚会在不同配置模式下有不同功能。

表 2-2 串行模式配置管脚

管脚名称	Bank	类型	JTAG	Master SPI				Slave Serial
				X1	X2	X4	X8	
SCBV	VCCIOCFG	专用	SCBV	SCBV	SCBV	SCBV	SCBV	SCBV
MODE[2:0]	VCCIOCFG	专用	xxx ⁽¹⁾	001	001	001	001	111
TCK	VCCIOCFG	专用	TCK	TCK	TCK	TCK	TCK	TCK
TMS	VCCIOCFG	专用	TMS	TMS	TMS	TMS	TMS	TMS
TDI	VCCIOCFG	专用	TDI	TDI	TDI	TDI	TDI	TDI
TDO	VCCIOCFG	专用	TDO	TDO	TDO	TDO	TDO	TDO
RSTN	VCCIOCFG	专用	RSTN	RSTN	RSTN	RSTN	RSTN	RSTN
INIT_FLAG_N	VCCIOCFG	专用	INIT_FLAG_N	INIT_FLAG_N	INIT_FLAG_N	INIT_FLAG_N	INIT_FLAG_N	INIT_FLAG_N
CFG_DONE	VCCIOCFG	专用	CFG_DONE	CFG_DONE	CFG_DONE	CFG_DONE	CFG_DONE	CFG_DONE
CFG_CLK	VCCIOCFG	专用	-	CFG_CLK	CFG_CLK	CFG_CLK	CFG_CLK	CFG_CLK
IO_STATUS_C	VCCIO5	复用	IO_STATUS_C	IO_STATUS_C	IO_STATUS_C	IO_STATUS_C	IO_STATUS_C	IO_STATUS_C
ECCLKIN	VCCIO5	复用	-	ECCLKIN	ECCLKIN	ECCLKIN	ECCLKIN	-
FCS_N	VCCIO5	复用	-	FCS_N	FCS_N	FCS_N	FCS_N	-
FCS2_N	VCCIO4	复用	-	-	-	-	FCS2_N	-
CS_N	VCCIO5	复用	-	-	-	-	-	-
RWSEL	VCCIO5	复用	-	-	-	-	-	-
D[31:16]	VCCIO5	复用	-	-	-	-	-	-

管脚名称	Bank	类型	JTAG	Master SPI				Slave Serial
				X1	X2	X4	X8	
D[15:8]	VCCIO15	复用	-	-	-	-	-	-
D[7:4]	VCCIO15	复用	-	-	-	-	D[7:4]	-
D[3:2]	VCCIO15	复用	-	-	-	D[3:2]	D[3:2]	-
MISO_D1_DI	VCCIO15	复用	-	MISO	D[1]	D[1]	D[1]	DI
MOSI_D0	VCCIO15	复用	-	MOSI	D[0]	D[0]	D[0]	-
CSO_DOUT	VCCIO15	复用	-	CSO_DOUT	-	-	-	CSO_DOUT

注：（1）禁止设置 MODE[2:0]=3' b000，此模式任何接口均无法使用。除了 3' b000 之外，设置为任意值均可使用 JTAG 模式，若只需要使用 JTAG 模式，可设置为 MODE[2:0]=3' b101，该设置下仅 JTAG 可用。

表 2-3 并行模式配置管脚

管脚名称	Bank	类型	Slave Parallel		
			X8	X16	X32
SCBV	VCCIOCFG	专用	SCBV	SCBV	SCBV
MODE[2:0]	VCCIOCFG	专用	110	110	110
TCK	VCCIOCFG	专用	TCK	TCK	TCK
TMS	VCCIOCFG	专用	TMS	TMS	TMS
TDI	VCCIOCFG	专用	TDI	TDI	TDI
TDO	VCCIOCFG	专用	TDO	TDO	TDO
RSTN	VCCIOCFG	专用	RSTN	RSTN	RSTN
INIT_FLAG_N	VCCIOCFG	专用	INIT_FLAG_N	INIT_FLAG_N	INIT_FLAG_N
CFG_DONE	VCCIOCFG	专用	CFG_DONE	CFG_DONE	CFG_DONE
CFG_CLK	VCCIOCFG	专用	CFG_DONE	CFG_DONE	CFG_DONE
IO_STATUS_C	VCCIO15	复用	IO_STATUS_C	IO_STATUS_C	IO_STATUS_C
ECCLKIN	VCCIO15	复用	-	-	-
FCS_N	VCCIO15	复用	-	-	-
FCS2_N	VCCIO14	复用	-	-	-
CS_N	VCCIO15	复用	CS_N	CS_N	CS_N
RWSEL	VCCIO15	复用	RWSEL	RWSEL	RWSEL
D[31:16]	VCCIO15	复用	-	-	D[31:16]
D[15:8]	VCCIO15	复用	-	D[15:8]	D[15:8]
D[7:4]	VCCIO15	复用	D[7:4]	D[7:4]	D[7:4]
D[3:2]	VCCIO15	复用	D[3:2]	D[3:2]	D[3:2]
MISO_D1_DI	VCCIO15	复用	D[1]	D[1]	D[1]
MOSI_D0	VCCIO15	复用	D[0]	D[0]	D[0]
CSO_DOUT	VCCIO15	复用	CSO_DOUT	CSO_DOUT	CSO_DOUT

配置管脚功能定义如表 2-4所示：

表 2-4 配置管脚功能定义

管脚名称	Bank	类型	方向	管脚说明
SCBV	VCCIOCFG	专用	输入	选择复用配置管脚所在的 Bank 电压。使用该管脚来确定 BANKCFG 与包含复用配置管脚的 BANK 电压之间的压差, 以及这些 Bank 的 I/O 电压工作范围。该管脚对 BANKCFG 始终起作用, 但只在配置期间对包含复用配置管脚的 Bank 起作用。当 VCCIOCFG 的电压是 2.5V 或者 3.3V, 该管脚必须接到高电平 (可以直接接到 VCCIOCFG)。当 VCCIOCFG 的电压为 1.8V 或者更低时, 该管脚必须接到低电平 (可以直接接到地)。注意: 该管脚需与软件配合使用, 即位流设置中 SCBV 的选择需与硬件设置一致! SCBV 管脚上下拉电平与配置相关 Bank 电源的对应情况参见《UG040012_Logos2 单板硬件设计指南》;
MODE[2:0]	VCCIOCFG	专用	输入	配置专用输入管脚, 用于配置模式选择; 内置弱上拉电阻; 确定模式时可以使用一个不大于 1K Ω 的电阻上拉到 VCCIOCFG 或者使用一个不大于 1K Ω 的电阻下拉到 VSS。
TCK	VCCIOCFG	专用	输入	符合 IEEE STD 1149.1 的测试时钟输入管脚; 为 FPGA 的 JTAG 链提供时钟, 测试时钟输入管脚; 内置弱上拉电阻 (始终都是使能), 上拉到 VCCIOCFG;
TMS	VCCIOCFG	专用	输入	专用 JTAG, 测试模式选择输入管脚; 内置弱上拉电阻 (始终都是使能), 上拉到 VCCIOCFG;
TDI	VCCIOCFG	专用	输入	专用 JTAG, 测试数据输入管脚; 内置弱上拉电阻 (始终都是使能), 上拉到 VCCIOCFG
TDO	VCCIOCFG	专用	输出	专用 JTAG, 测试数据输出管脚; 内置弱上拉电阻 (始终都是使能), 上拉到 VCCIOCFG
RSTN	VCCIOCFG	专用	输入	专用的配置复位管脚, 内置弱上拉电阻, 且弱上拉一直有效; 用于复位配置逻辑及配置存储器, 低有效。 当该管脚被置为低, FPGA 配置存储器被清空并且开始新的配置过程。配置逻辑复位开始于该管脚的下降沿, 配置过程开始于该管脚紧接的上升沿。 该管脚需要外接不大于 4.7K Ω 的上拉电阻到 VCCIOCFG。 在上电时保持该管脚为低电平并不会让 FPGA 配置逻辑处于复位状态。 配置完成之后, 用户可选配置为弱上拉或者浮空。
INIT_FLAG_N	VCCIOCFG	专用	双向(开漏)	初始化及配置状态专用管脚。当输出为低的时, 表示 FPGA 芯片正在执行初始化操作 (清空配置存储器) 或者发生了配置错误。 内置弱上拉电阻, 在配置过程中, 使能弱上拉; 当 FPGA 上电完成, 驱动该管脚为低电平。当 FPGA 初始化完成后, 释放对该管脚的驱动。在上电以及初始化过程中, 该管脚可以由外部输入低电平, 用于延迟初始化结束之后的配置过程。当芯片在初始化完成后检测到这个管脚上的高电平输入, 芯片开始启动配置过程。在配置过程中, 该管脚作为配置错误的指示状态输出, 为低表示配置过程中出现错误; 该管脚需要外接一个弱上拉电阻到 VCCIOCFG (建议电阻不大于 4.7K Ω) 配置完成之后, 用户可选配置为弱上拉或者浮空。
CFG_DONE	VCCIOCFG	专用	双向(开漏)	配置状态专用管脚。内置弱上拉电阻 (大约 10K Ω); 作为配置完成指示状态输出, 为高表示配置完成。默认为开漏输出。当 FPGA 上电完成, 在配置之前或者配置过程中该管脚被驱动为低电平。一旦所有的配置数据被正确的接收并且启动了唤醒时序, 释放对该管脚的控制。 在配置完成后, 该管脚可以由外部继续驱动为低电平, 一旦内部唤醒时序发现外部 DONE 管脚为低电平, 那么内部的唤醒电路继续保持直到外部管脚变为高电平才会继续执行唤醒过程。 该管脚需要外接一个弱上拉电阻到 VCCIOCFG (建议电阻不大于 4.7K Ω) 配置完成之后, 用户可选配置为弱上拉或者浮空。
CFG_CLK	VCCIOCFG	专用	输入/输出	配置时钟管脚。除了 JTAG 配置模式外, 其他模式都是使用该时钟来同步 FPGA 的配置过程。

管脚名称	Bank	类型	方向	管脚说明
				在 Slave Serial, Slave Parallel 模式, 该管脚作为时钟输入以从外部获取配置数据; 在 Master SPI 配置模式, 该管脚作为时钟输出以从外部获取配置数据, 在此模式, 需外接 1 K Ω 上拉电阻; 不需要该时钟时 (比如 JTAG 模式), 处于高阻态。 配置完成之后, 用户可选配置为弱上拉或者浮空。
IO_STATUS_C	VCCIO15	复用	输入	复用管脚, 输入信号, 控制从上电完成后到进入用户模式之前中所有用户 IO (包括复用 IO) 的弱上拉电阻是否使能。 (1) “0”, 使能所有用户 IO 内部上拉电阻。 (2) “1”, 不使能所有用户 IO 内部上拉电阻。 (3) 建议该管脚外部加强上拉电阻到对应的 VCCIO。 (4) 该管脚可以直接或者通过一个不大于 1K Ω 的电阻接到对应的 VCCIO 或者 VSS。 (5) 在配置之前或者配置过程中, 该管脚不允许悬空;
ECCLKIN	VCCIO15	复用	输入	外部输入的主模式配置时钟。这是一个可选的外部时钟输入到配置逻辑。 (1) 在主模式下 (Master SPI), FPGA 可以选择这个时钟输入作为配置逻辑的配置时钟。该时钟可以被分频 (取决于位流中的设置) 并从 CFG_CLK 输出。 (2) 对于其他配置模式、用户模式以及初始化期间, 该管脚不会被使用, 作为普通用户 I/O, 为高阻态或者弱上拉。
FCS_N	VCCIO15	复用	输出	复用配置管脚, 外部 Master SPI 配置模式使用。 (1) Master SPI X1, X2, X4 模式下, 低有效输出片选信号给外部 flash; 同时需要外部上拉电阻 (不大于 4.7K Ω) 到其所在 Bank 的 VCCIO (2) 其他配置模式下以及初始化期间, 该管脚作为普通用户 I/O, 为高阻态或者弱上拉。 (3) 配置完成后, 可以作为用户 I/O 使用。
FCS2_N	VCCIO14	复用	输出	复用配置管脚, 外部 Master SPI X8 配置模式使用。 (1) Master SPI X8 模式下, 低有效输出片选信号给外部 flash; 需要外接上拉电阻不大于 4.7K Ω 到 VCCIO (2) 对于其他配置模式、使用模式以及初始化期间, 该管脚不会被使用, 作为普通用户 I/O, 为高阻态或者弱上拉。 (3) 配置完成后, 可以作为用户 I/O 使用。
CS_N	VCCIO15	复用	输入	复用配置管脚。片选输入, 低电平有效。 (1) 为低电平使能 Slave Parallel 模式配置接口。在 Slave Parallel 配置模式下, 外部的控制器可通过控制该管脚来选择即将工作的 FPGA 的 Slave Parallel 总线, 或者在 Slave Parallel 配置链中连接到前一个 FPGA 的 CSO_DOUT 管脚。 (2) 对于其他配置模式以及初始化期间, 该管脚不会被使用, 作为普通用户 I/O, 为高阻态或者弱上拉。 (3) 在配置结束后, 可以作为用于 I/O 使用。
RWSEL	VCCIO15	复用	输入	复用配置管脚。Slave Parallel 配置模式读写选择输入, 高为读, 低为写。 (1) 为高电平, Slave Parallel 配置模式从数据总线读取数据; (2) 为低电平, Slave Parallel 配置模式向数据总线写数据; (3) 读写之间的切换只能在 CS_N 为高电平期间才可以切换。 (4) 在配置结束后, 可以作为用户 I/O 使用。 (5) 对于其他配置模式以及初始化期间, 该管脚不会被使用, 作为普通用户 I/O, 为高阻态或者弱上拉。
D[31:16]	VCCIO15	复用	输入/输出	复用配置数据管脚 (1) 对于 Slave Parallel, X32 位宽, 作为数据总线的 D[31: 16]。 (2) 对于其他配置模式、位宽不够的以及初始化期间, 该管脚不会被使用, 作为普通用户 I/O, 为高阻态或者弱上拉。 (3) 配置完成后, 可以作为用户 I/O 使用。
D[15:8]	VCCIO15	复用	输入/输出	复用配置数据管脚 (1) 对于 Slave Parallel 模式, X16 或 X32 位宽, 作为数据总线的 D[15: 8]。 (2) 对于其他配置模式、位宽不够的以及初始化期间, 该管脚不

管脚名称	Bank	类型	方向	管脚说明
				会被使用，作为普通用户 I/O，为高阻态或者弱上拉。 (3) 配置完成后，可以作为用户 I/O 使用。
D[7:4]	VCCIO15	复用	输入/输出	复用配置数据管脚： (1) Master SPI X8 模式，以和 D[3:0]相同的方式连接到第二片 flash (2) 对于 Slave Parallel 模式，作为数据总线的 D[7:4]。 (3) 对于其他配置模式以及初始化期间，该管脚不会被使用，作为普通用户 I/O，为高阻态或者弱上拉。 (4) 配置完成后，可以作为用户 I/O 使用。
D[3:2]	VCCIO15	复用	输入/输出	复用配置数据管脚 (1) Master SPI 模式，X4/X8，作为数据的输入连接到 SPI flash 的数据输出端。“D2”连接到外部 SPI flash 的第 3 位数据输出（如 DQ2，W#，WP#，IO2），“D3”连接到外部 SPI flash 的第 4 位数据输出（如 DQ3，HOLD#，IO3 等），该管脚需要外部弱上拉一个 4.7KΩ 的电阻到其所对应的 VCCIO。 (2) 对于 Slave Parallel 模式，作为数据总线的 D[3:2]。 (3) 对于其他配置模式以及初始化期间，该管脚不会被使用，作为普通用户 I/O，为高阻态或者弱上拉。 (4) 配置完成后，可以作为用户 I/O 使用。
MISO_D1_DI	VCCIO15	复用	输入/输出	复用配置数据管脚 (1) Master SPI 模式，X1，“MISO”作为数据输入连接到 SPI flash 的数据输出端（如 DQ1，Q，SO，IO1 等）。 (2) Master SPI X2、X4、X8 模式，“D1”连接到 SPI flash 的第 2 个串行数据输出端口（如：DQ1，Q，SO，IO1 等）。 (3) 对于 Slave Parallel 模式，作为数据总线的 D[1]。 (4) 对于 Slave Serial 模式，“DI”作为数据输入管脚。 (5) 其他配置模式下以及初始化期间，该管脚作为普通用户 I/O，为高阻态或者弱上拉。在其他配置模式下如 JTAG，该管脚上的状态会被忽略。 (6) 配置完成后，可以作为用户 I/O 使用。
MOSI_D0	VCCIO15	复用	输入/输出	复用配置数据管脚： (1) “MOSI”，Master SPI X1 模式串行数据输出，连接到外部 SPI flash 的数据输入管脚（如 DQ0，D，SI，IO0 等）。当命令和地址被发送到外部 SPI flash 后，该管脚输出为高阻态或者弱上拉 (2) Master SPI X2/X4/X8 模式，双向数据端口，作为命令和地址输出给到外部的 SPI flash。最低位数据输入从外部 SPI flash 接收数据。连接到外部 SPI flash 的双向数据管脚（如 DQ0，D，SI，IO0 等） (3) “D0”，对于 Slave Parallel 模式，作为数据总线的 D[0]； (4) 对于其他配置模式以及初始化期间，该管脚不会被使用，作为普通用户 I/O，为高阻态或者弱上拉。 (5) 配置完成后，可以作为用户 I/O 使用。
CSO_DOUT	VCCIO15	复用	输出(开漏)/输出	复用配置管脚。只在级联时才需要使用。 (1) Master SPI，X1 模式，可作为级联数据输出；其他情况下，对于其他配置模式以及初始化期间，该管脚不会被使用，作为普通用户 I/O，为高阻态或者弱上拉。 (2) Slave Serial 模式配置，可作为级联数据输出；其他情况下，对于其他配置模式以及初始化期间，该管脚不会被使用，作为普通用户 I/O，为高阻态或者弱上拉。 (3) 在任意配置模式，当选中了位流调试选项（安全级别最低），其可以输出数据。 (4) Slave Parallel 级联模式配置下，可作为片选信号开漏输出；连接到下游器件的 CS_N 管脚，并且外部需要加入一个 330Ω 的上拉电阻到其所对应的 VCCIO。其他情况下，对于其他配置模式以及初始化期间，该管脚不会被使用，作为普通用户 I/O，为高阻态或者弱上拉。 (5) 对于其他配置模式以及初始化期间，该管脚不会被使用，作为普通用户 I/O，为高阻态或者弱上拉。

2.1 JTAG 配置模式

JTAG配置模式应用接口示意图如图 2-1所示。

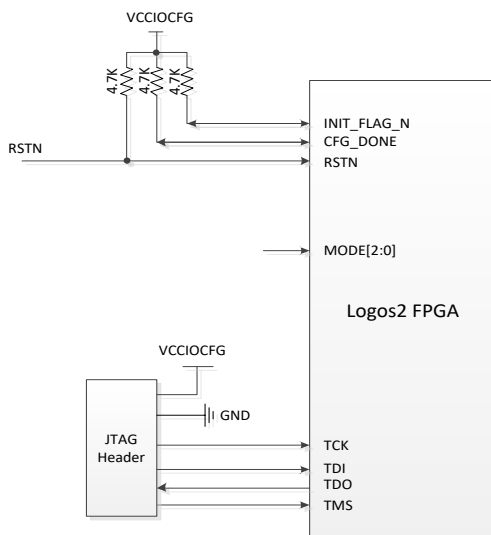


图 2-1 JTAG 模式应用示意图

RSTN 低电平将复位配置逻辑，建议外部使用4.7K电阻上拉至VCCIIOCFG（3.3V）。初始化过程中，INIT_FLAG_N作为输入时，如果连接低电平将使Logos2 FPGA停留在初始化阶段，建议外部采用4.7K的电阻上拉至VCCIIOCFG。INIT_FLAG_N作为输出时，输出高电平指示芯片初始化结束。CFG_DONE输出高电平表示芯片进入用户模式，外部输入低电平将使Logos2 FPGA停留在配置阶段，建议外部采用4.7K的电阻上拉至VCCIIOCFG。

JTAG模式下TCK测试时钟需要由外部提供；外部可以通过改变TMS的状态来控制JTAG内部TAP状态机的跳变，以此选择配置位流的写入（TDI）或者片内数据回读（TDO）。除了配置编程，JTAG接口还常用于在线调试和边界扫描测试。JTAG模式下，建议将TDI、TCK、TMS用12~15K电阻上拉至VCCIIOCFG，以提供稳定的初始输入电平。

JTAG编程模式典型时序如图 2-2所示。

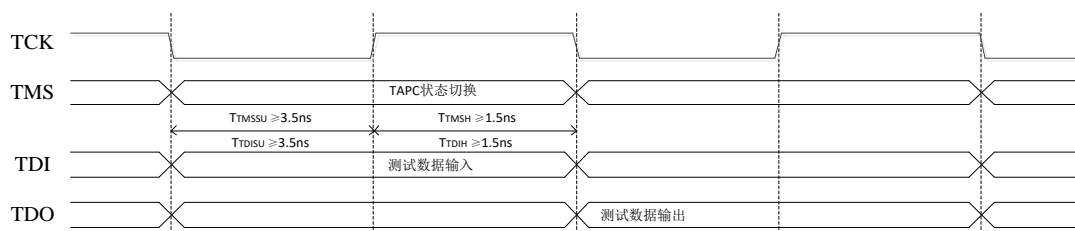


图 2-2 JTAG 编程典型时序图

由于JTAG接口可用于编程eFuse，因此为了避免JTAG接口上可能由于环境引发的干扰信

号导致eFuse被误写入，建议用户在确认芯片应用场景后，将eFuse所有功能位锁定，详见eFuse介绍。

2.1.1 JTAG 级联模式

可用JTAG菊花链配置多个器件，连接如图 2-3所示。

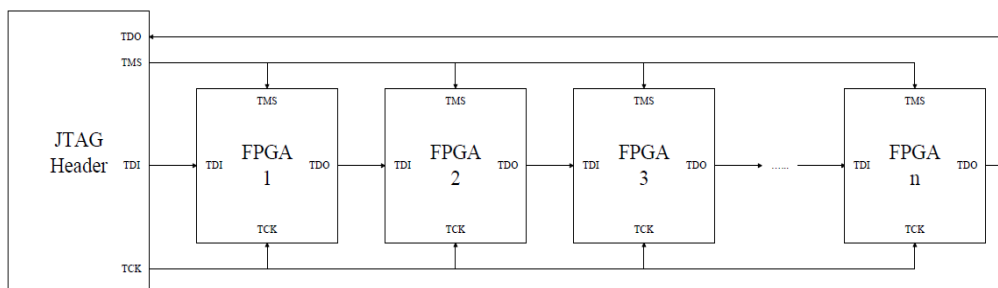


图 2-3 JTAG 级联应用示意图

此种方式，应将模式管脚设为JTAG模式。连接下载线之后，主机上的PDS软件会扫到在JTAG链上的所有器件，用户可选择对相应的FPGA进行编程下载。

TCK和TMS信号连接了所有JTAG链上的器件，所以这两者的信号质量会影响JTAG配置的最大频率及可靠性。注意，JTAG级联时，仅支持对第一级FPGA的外部Flash进行操作。

2.1.2 JTAG 边界扫描结构

器件支持 IEEE1149.1 标准，JTAG 边界扫描结构可概括如图 2-4 所示。

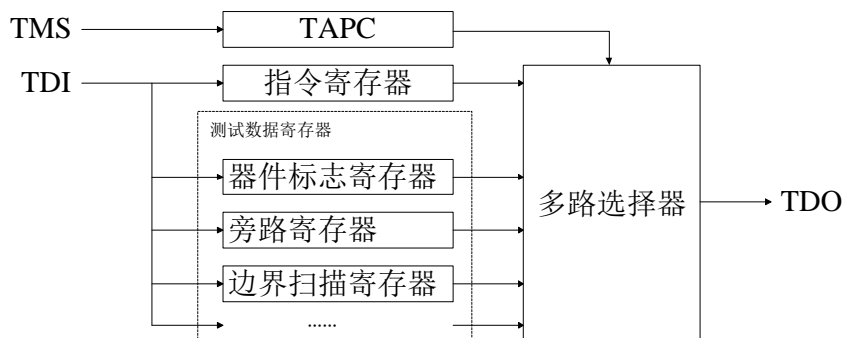


图 2-4 JTAG 边界扫描系统框图

边界扫描硬件结构主要包含：测试访问端口控制器（TAPC）、指令寄存器（IR）、测试数据寄存器（TDR）。

其中 TAPC 实现 IEEE1149.1 标准定义的状态跳转控制。指令寄存器及测试数据寄存器的功能和用法、以及 FPGA 各器件边界寄存器的分布请参见 BSDL (*.bsm) 文件，该文件可在 PDS 软件安装目录下获取。

指令寄存器支持的部分常用指令及说明如表 2-5 所示。

表 2-5 JTAG 常用指令集

指令	类型	操作码	描述
BYPASS	1149.1 非测试指令	111111111	旁路指令
SAMPLE/PRELOAD	1149.1 非测试指令	1010000000	采样/预装指令
EXTEST	1149.1 测试指令	1010000001	外测试指令
IDCODE	1149.1 非测试指令	1010000011	标识指令
HIGHZ	1149.1 测试指令	1010000101	高阻指令
JRST	设计专有	1010001010	复位指令
CFGI	设计专有	1010001011	配置指令
CFGO	设计专有	1010001100	回读指令
JWAKEUP	设计专有	1010001101	唤醒指令

2.2 Master SPI 配置模式

Master SPI配置接口如图 2-5所示：

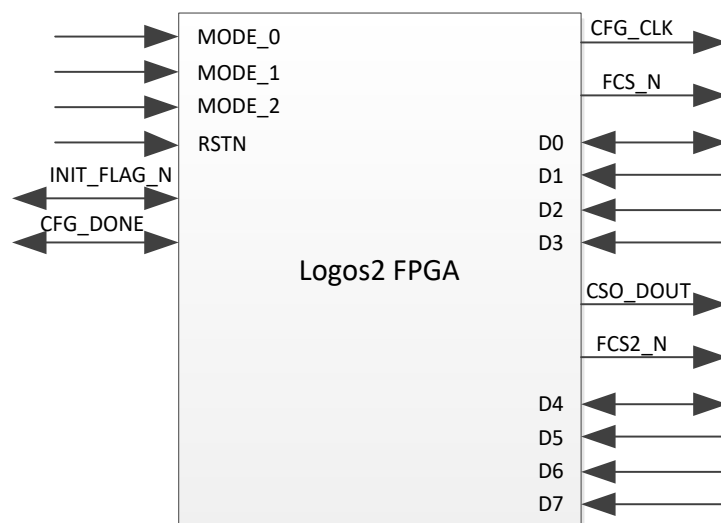


图 2-5 Master SPI 配置接口

Master SPI配置接口信号描述如表 2-6所示。

表 2-6 Master SPI 接口信号描述

管脚名称	方向	类型	管脚说明
RSTN	输入	专用	异步全芯片复位信号，低有效
CFG_CLK	输出	专用	配置时钟
MODE[2:0]	输入	专用	配置模式管脚 001: Master SPI 配置模式(x1/x2/x4/x8)

管脚名称	方向	类型	管脚说明
INIT_FLAG_N	双向（开漏）	专用	采样 MODE[2:0]管脚之前，INIT_FLAG_N 是输入， 可以通过保持低电平来延迟配置。 采样 MODE[2:0]管脚之后，INIT_FLAG_N 是开漏， 指示配置过程是否出错。 0: 错误 1: 正确
FCS_N	输出	复用	SPI Flash 芯片选择信号，低有效
FCS2_N	输出	复用	x8 模式第二片 SPI Flash 芯片选择信号，低有效
D[7: 0]	双向	复用	串行输入输出数据总线，CFG_CLK 上升沿或下降沿采样 D[7:4]为 x8 模式第二片 SPI Flash 的串行输入输出数据总线
CSO_DOUT	输出	复用	菊花链数据输出，CFG_CLK 下降沿发送
CFG_DONE	双向（开漏）	专用	指示配置完成 0: FPGA 未配置 1: FPGA 已配置

Master SPI模式下，位流通常保存在外部SPI Flash中。要选择Master SPI模式，建议将MODE[2:1]通过下拉电阻连接到地，将MODE[0]通过上拉电阻连接到VCCIOCFG电源。通过上电或给RSTN一个低电平脉冲来启动编程，Logos2系列FPGA会主动从外部Flash读取位流，CCS自动使用fast read X1模式，上升沿采样数据，从串行NOR FLASH的0地址开始取数据。取到位流开头部分模式设置指令后，切换到指令规定的模式、上升沿还是下降沿采样。位流全部从Flash取出后，FCS_N（FCS2_N）置1，Master SPI模式操作结束。

若设定的CFG_CLK频率不高于15.38MHz，可使用上升沿采样。若高于15.38MHz，应设置使用下降沿采样。

CFG_CLK由芯片内部产生并输出；通过监控管脚INIT_FLAG_N和CFG_DONE来判断编程是否结束；Master SPI支持1/2/4/8bit三种数据位宽模式，可以通过对位流的解析来选择数据位宽，针对不同数据位宽，数据bit位选择、数据输入/输出方向会不一样，如下描述：

X1位宽时，FPGA的管脚D[0]作为命令输出连接到SPI Flash的数据输入端，管脚D[1]作为数据输入连接到SPI Flash的数据输出端。应用示意图如图 2-6所示：

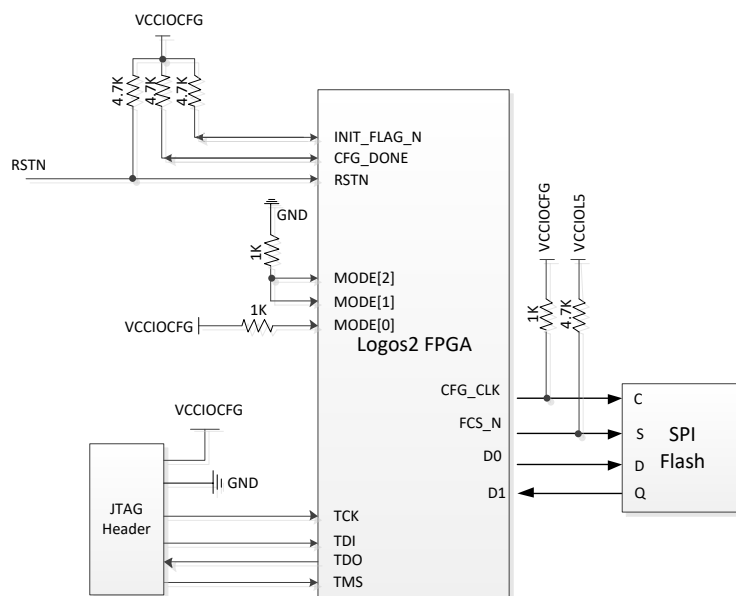


图 2-6 Master SPI 配置模式 X1 应用示意图

X2 位宽时，管脚 D[1:0]为数据总线。应用示意图如图 2-7 所示：

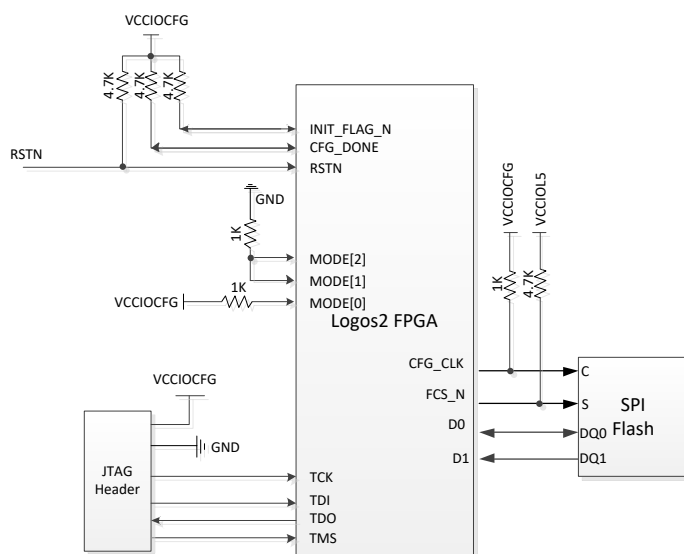


图 2-7 Master SPI 配置模式 X2 应用示意图

X4位宽时，管脚D[3:0]为数据总线。应用示意图如图 2-8所示：DQ[3:2]必须有上拉电阻。

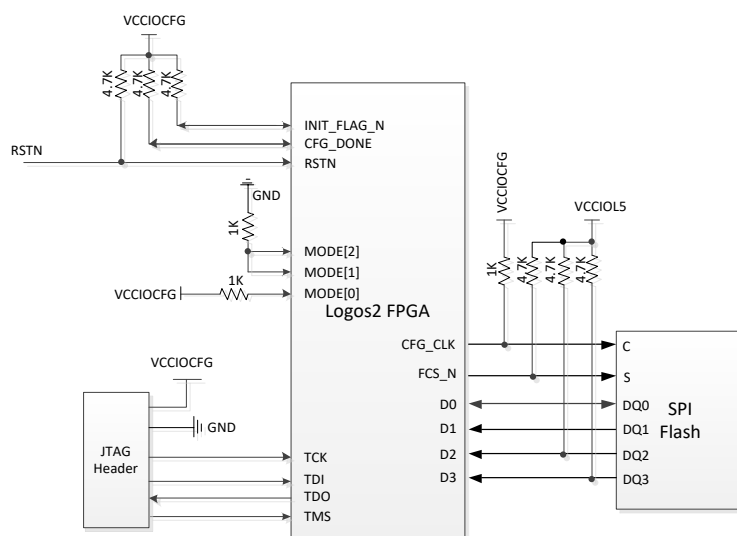


图 2-8 Master SPI 配置模式 X4 应用示意图

X8 位宽时，管脚 D[3:0] 为第 1 片 SPI Flash 的数据总线，管脚 D[7:4] 为第 2 片 SPI Flash 的数据总线。应用示意图如图 2-9 所示：

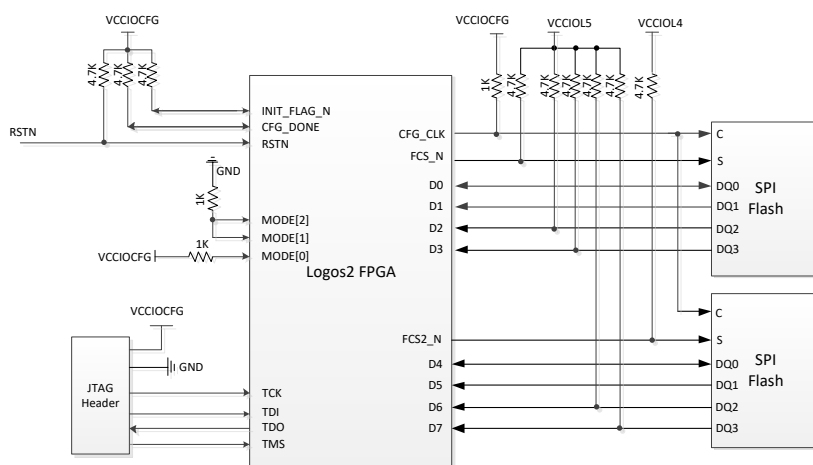


图 2-9 Master SPI 配置模式 X8 应用示意图

Master SPI 编程模式典型时序如图 2-10 所示：

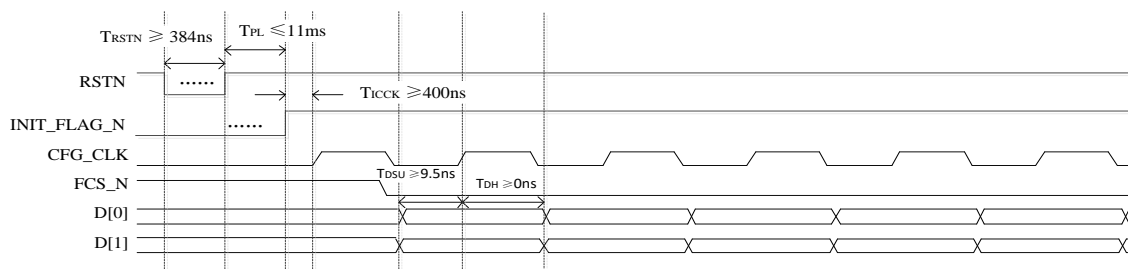


图 2-10 1bit Master SPI 编程典型时序图

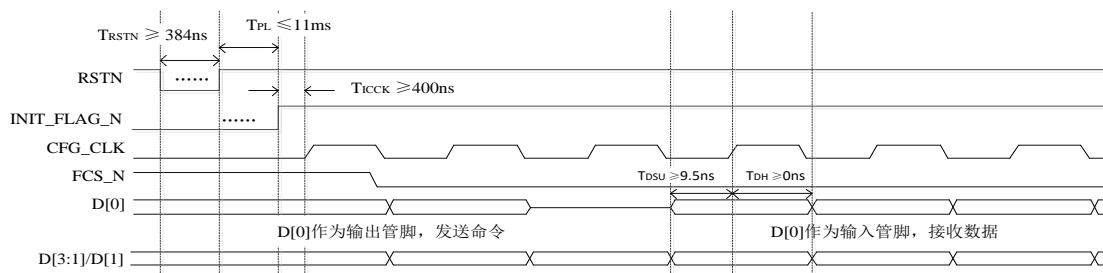


图 2-11 2/4bit Master SPI 编程典型时序图

Logos2系列FPGA支持的部分SPI Flash型号如表 2-7所示。

表 2-7 支持的 SPI Flash 型号

型号	厂家	容量
N25Q32	Micron	32Mb
N25Q64	Micron	64Mb
N25Q128	Micron	128Mb
N25Q256	Micron	256Mb
N25Q512	Micron	512Mb
W25Q80	WINBOND	8Mb
W25Q16	WINBOND	16Mb
W25Q32	WINBOND	32Mb
W25Q64	WINBOND	64Mb
W25Q128	WINBOND	128Mb
W25Q256	WINBOND	256Mb

注：选用 Flash 时需根据实际存储空间需求（如实际位流大小，Logos2 器件普通位流大小见表 3-3）进行考虑。以上 Flash 为经过测试的型号，若选择其他 Flash，需经过测试。

2.2.1 Master SPI 串行菊花链

Master SPI配置模式支持使用Master SPI模式和Slave Serial模式进行SPI串行菊花链配置。SPI串行菊花链中，第一片器件使用Master SPI模式，其它器件使用Slave Serial模式。位流需要使用 Fabric Configuration 工具合成（使用 Fabric Configuration 工具生成位置：Operations->Generate Daisy Chain File）。SPI串行菊花链配置时，从最下级器件开始配置，最后配置第一片器件。逻辑框图如图 2-12所示：

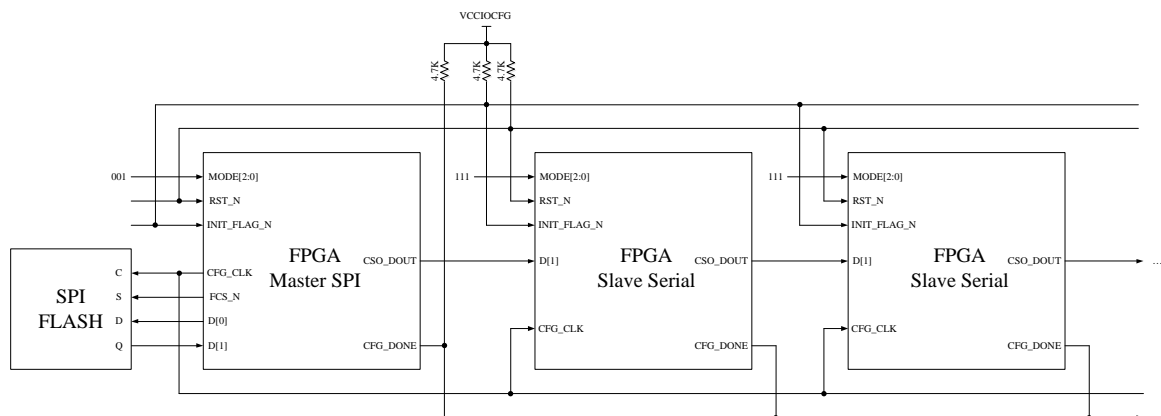


图 2-12 Master SPI 串行菊花链应用示意图

2.3 Slave Serial 配置模式

Slave Serial 配置接口如图 2-13 所示：

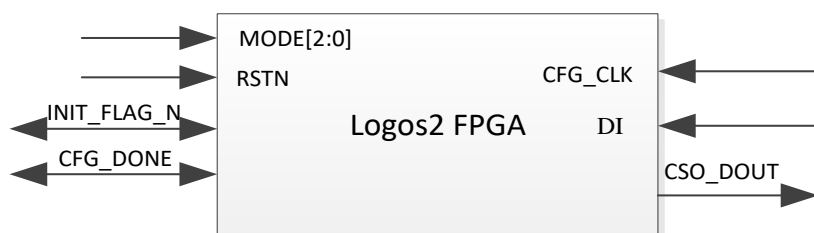


图 2-13 Slave Serial 配置接口

Slave Serial 配置接口信号描述如表 2-8 所示。

表 2-8 Slave Serial 配置接口信号描述

管脚名称	方向	类型	管脚说明
RSTN	输入	专用	异步全芯片复位信号，低有效
CFG_CLK	输入	专用	配置时钟
MODE[2:0]	输入	专用	配置模式管脚 111: Slave Serial 模式
INIT_FLAG_N	双向（开漏）	专用	采样 MODE[2:0]管脚之前，INIT_FLAG_N 是输入， 可以通过保持低电平来延迟配置。 采样 MODE[2:0]管脚之后，INIT_FLAG_N 是开漏， 指示配置过程是否出错。 0: 错误 1: 正确
CFG_DONE	双向（开漏）	专用	指示配置完成 0: FPGA 未配置 1: FPGA 已配置
DI	输入	复用	串行输入数据总线，CFG_CLK 上升沿采样用于自身的配置，级联时下降沿采样用于下一级的配置。
CSO_DOUT	输出	复用	菊花链数据输出，CFG_CLK 下降沿发送

要选择Slave Serial模式，建议将MODE[2:0]通过上拉电阻连接VCCIOCFG。Slave Serial配置模式的应用示意图如图 2-14所示：

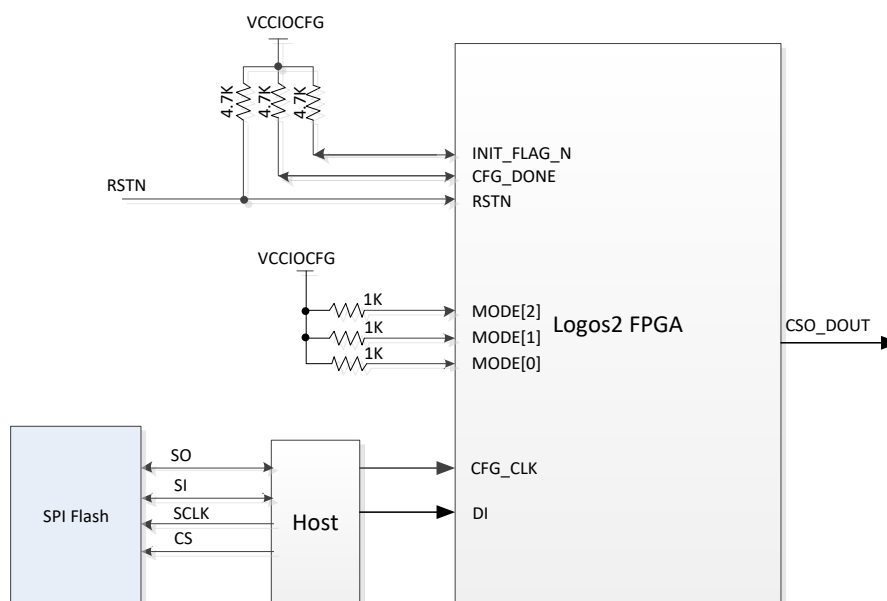


图 2-14 Slave Serial 模式应用示意图

在Slave Serial配置模式下，可以通过一个主控芯片（Host）控制板上多个芯片的上电启动以及数据加载。Host可以是微处理器、CPLD或者其他FPGA。该模式下可以通过上电或给RSTN一个低电平脉冲来启动编程；通过监控管脚INIT_FLAG_N和CFG_DONE来判断编程是否结束。

Host向Logos2器件发送位流时，若不使能等待PLL Lock（默认不使能），器件会在位流结尾的100个无操作类型1包头之间某个时钟周期释放对CFG_DONE的控制，由于外部上拉电阻的存在，CFG_DONE会在此时拉高。而CFG_DONE拉高之后的位流部分用于为器件唤醒提供时钟，因此需要确保将位流发完整后才能终止时钟。若使能等待，则在CFG_DONE拉高前不能终止时钟，并且在CFG_DONE拉高后，需多给至少100个时钟。

当使用串行菊花链配置时，每一级器件除了会使用CFG_CLK上升沿采样DI用于自身配置之外，还会使用下降沿采样DI用于输出配置下一级。因此级联配置时，Host需要同时考虑上升沿和下降沿满足建立保持时间要求。Slave Serial编程模式典型时序如图 2-15所示：

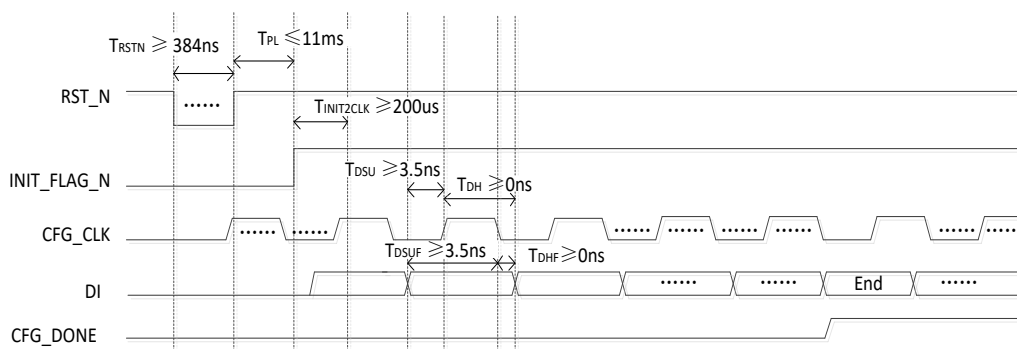


图 2-15 Slave Serial 编程典型时序图

2.3.1 Slave Serial 串行菊花链

Slave Serial配置模式的典型应用是用于串行菊花链。所有器件可以全使用Slave Serial模式进行串行级联配置。级联配置时，从最末级器件（最远端）开始配置，最后配置第一片器件。位流需要使用Fabric Configuration工具合成（使用Fabric Configuration工具生成位置：Operations->Generate Daisy Chain File）。

串行级联可以通过一个主控芯片（Host）控制板上多个级联芯片的配置。连接如图 2-16 所示。

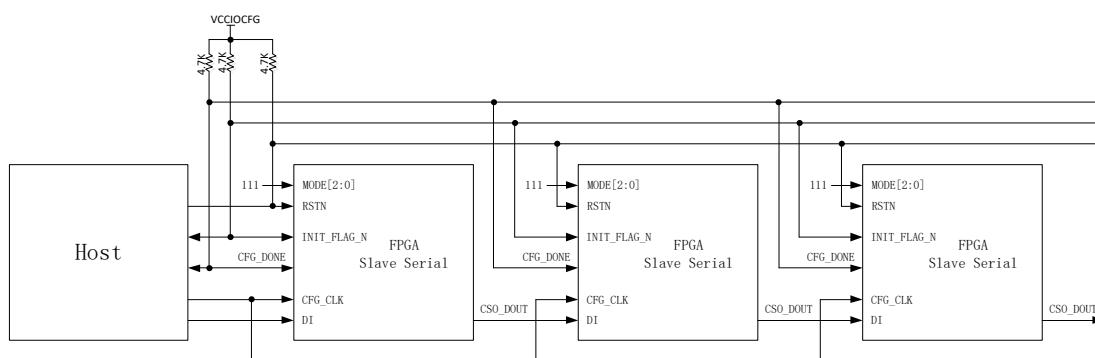


图 2-16 Slave Serial 配置模式串行菊花链连接

使用串行菊花链时，需要注意：

1. 满足首级时序

如图 2-15所示，Host发送的时序需满足CFG_CLK上升沿、下降沿采样DI的建立保持时间要求。

2. 满足后级时序

首级时序可以通过Host调整输出来满足，而后级输入均来自前级，可以调整的主要有PCB

走线等引入的延时。如图 2-17 所示，需要调整延时参数 $T_{CLKDELAY}$ 和 $T_{DOUTDELAY}$ 来满足后级器件两个沿的建立保持时间要求。

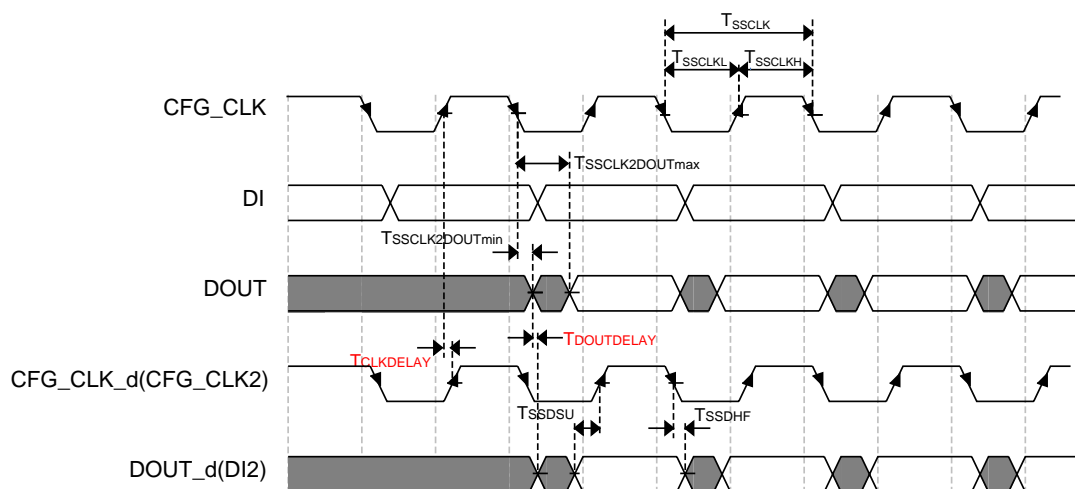


图 2-17 调整时序举例-后级

注： $T_{SSCLK2DOUT}$ 、 T_{SSCLKL} 、 T_{SSCLKH} 、 T_{SSDSU} 、 T_{SSDHF} 见《DS04001_Logos2 系列 FPGA 器件数据手册》。

由上图可以得到，下降沿发送数据的情况下，若满足上升沿的建立时间，则必定满足下降沿的建立时间，满足下降沿的保持时间则必满足上升沿的保持时间。因此 $T_{CLKDELAY}$ 和 $T_{DOUTDELAY}$ 需满足如下公式（单位：ns）：

$$\begin{cases} T_{SSDSU} = (T_{SSCLK} + T_{CLKDELAY}) - (T_{SSCLKH} + T_{SSCLK2DOUTmax} + T_{DOUTDELAY}) \geq 3.5 \\ T_{SSDHF} = T_{SSCLK2DOUTmin} + T_{DOUTDELAY} - T_{CLKDELAY} \geq 0 \end{cases}$$

即：

$$10.5 - T_{SSCLKL} \leq T_{CLKDELAY} - T_{DOUTDELAY} \leq 2$$

2.4 Slave Parallel 配置模式

Slave Parallel配置接口如图 2-18所示：

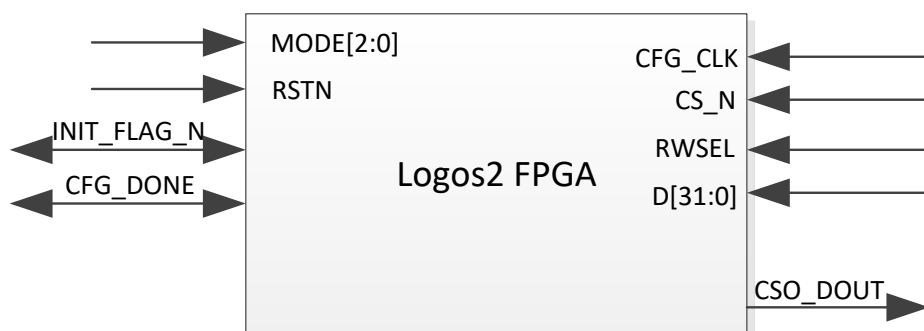


图 2-18 Slave Parallel 配置接口

Slave Parallel配置接口信号描述如表 2-9所示。

表 2-9 Slave Parallel 配置接口信号描述

管脚名称	方向	类型	管脚说明
RSTN	输入	专用	异步全芯片复位信号，低有效
CFG_CLK	输入	专用	配置时钟
MODE[2:0]	输入	专用	配置模式管脚 110: Slave Parallel 模式
INIT_FLAG_N	双向（开漏）	专用	采样 MODE[2:0]管脚之前，INIT_FLAG_N 是输入， 可以通过保持低电平来延迟配置。 采样 MODE[2:0]管脚之后，INIT_FLAG_N 是开漏， 指示配置过程是否出错。 0: 错误 1: 正确
CFG_DONE	双向（开漏）	专用	指示配置完成 0: FPGA 未配置 1: FPGA 已配置
CS_N	输入	复用	外部并行接口芯片选择信号，低有效。CFG_CLK 上升沿采样。 (1)
RWSEL	输入	复用	外部并行接口读写控制信号。CFG_CLK 上升沿采样。 0: 写 1: 读
D[31:0]	双向	复用	并行数据总线，CFG_CLK 上升沿采样和发送
CSO_DOUT	输出	复用	菊花链数据输出，CFG_CLK 下降沿发送

注：

(1) CS_N 为同步信号，在 INIT_FLAG_N 拉高后，对芯片进行配置前，需要保持 CS_N 为高电平并给 CFG_CLK 至少 8 个时钟周期，以确保接口处于初始工作状态。

Slave Parallel模式下，可以通过一个主控芯片（Host）控制板上多个芯片的上电启动以及数据加载。Host可以是微处理器、CPLD或者其他FPGA。要选择Slave Parallel模式，建议将MODE[0]通过下拉电阻连接到地，将MODE[2:1]通过上拉电阻连接到VCCIOCFG电源。应用示意图如图 2-19所示：

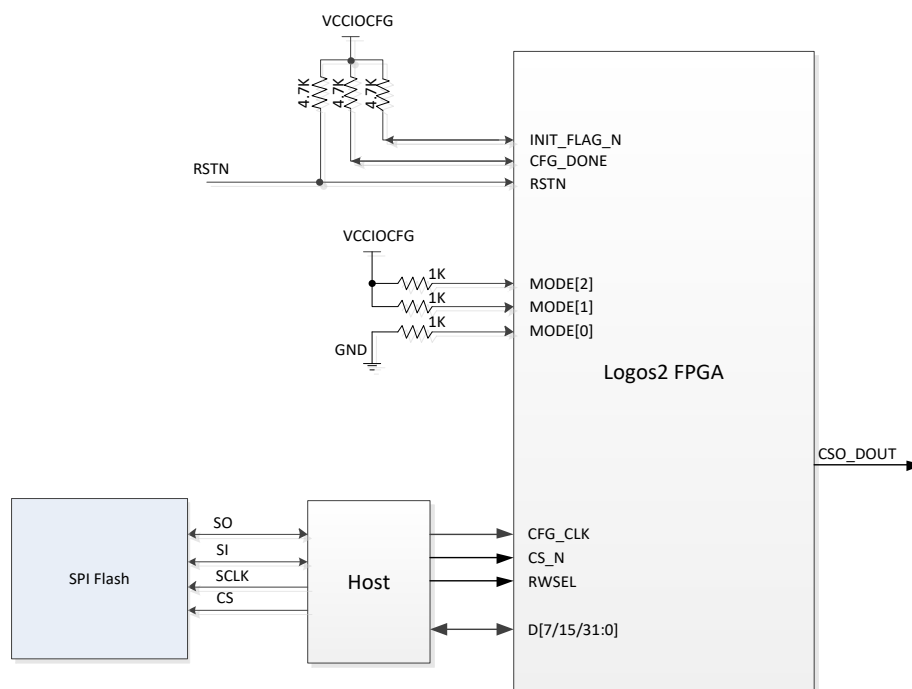


图 2-19 Slave Parallel 模式应用示意图

该模式下可以通过上电或给 RSTN 一个低电平脉冲来启动编程；

通过监控管脚 INIT_FLAG_N 和 CFG_DONE 来判断编程是否结束；Host 向 Logos2 器件发送位流时，若不使能等待 PLL Lock（默认不使能），器件会在位流结尾的 100 个无操作类型 1 包头之间某个时钟周期释放对 CFG_DONE 的控制，由于外部上拉电阻的存在，CFG_DONE 会在此时拉高。而 CFG_DONE 拉高之后的位流部分用于为器件唤醒提供时钟，因此需要确保将位流发完整后才能拉高 CS_N。若使能等待，则在 CFG_DONE 拉高前不能终止时钟，在 CFG_DONE 拉高后拉高 CS_N。拉高 CS_N 之后需多给至少 100 个时钟。

Slave Parallel 模式支持 8/16/32bit 三种数据位宽模式，可以通过对位流的解析来选择数据位宽。详情参见[总线位宽自动检测](#)。

Logos2 器件 Slave Parallel 接口支持 x8/x16/x32 几种位宽，使用 Slave Parallel 接口进行配置时，应注意数据总线与数据的位序对应，参考[不同接口位宽下的位序](#)。

Slave Parallel 配置接口典型时序图如图 2-20 所示。

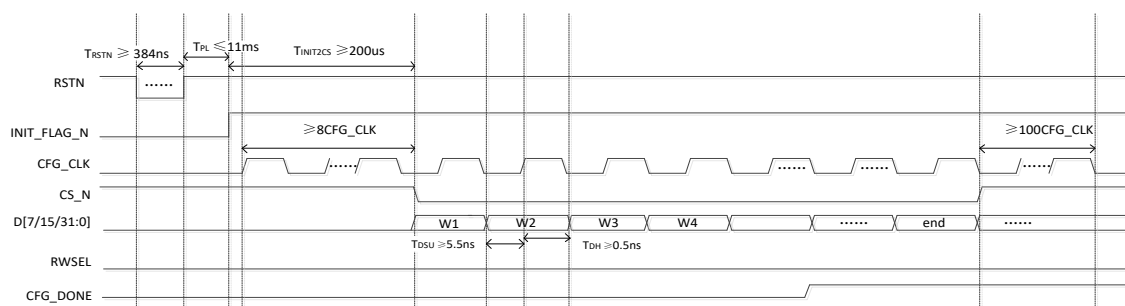


图 2-20 Slave Parallel 配置时序图

当Host无法连续地发送数据流时，可通过控制Slave Parallel配置接口的CS_N或CFG_CLK来实现非连续加载，如图 2-21所示。

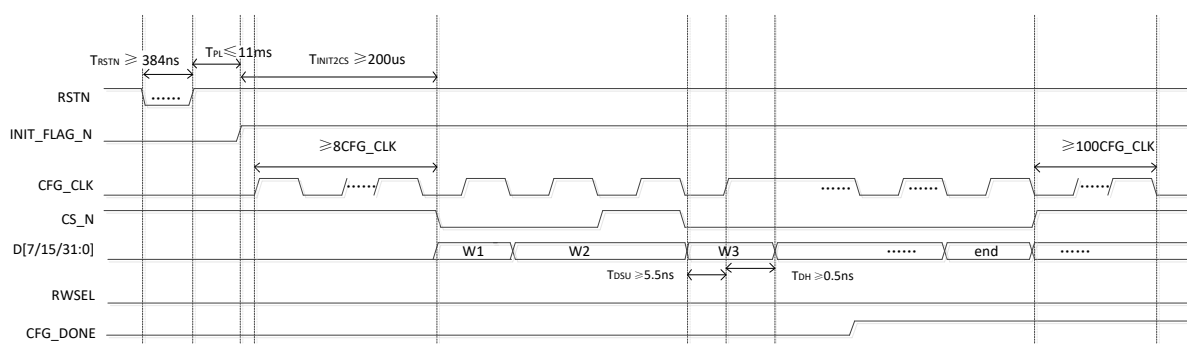


图 2-21 Slave Parallel 非连续配置时序图

其中发送W2之后，通过拉高CS_N来暂停数据加载，需要特别注意，拉高CS_N来暂停数据加载时，需要保持数据不变，即D[31:0]在CS_N为高时保持不变，否则可能导致加载失败；发送W3之后，通过停止CFG_CLK的翻转，同样可以暂停数据加载，这种情况下不需要保持D[31:0]不变。

另外，Slave Parallel接口也支持回读，典型时序如图 2-22所示，具体回读流程参考Slave Parallel接口回读。

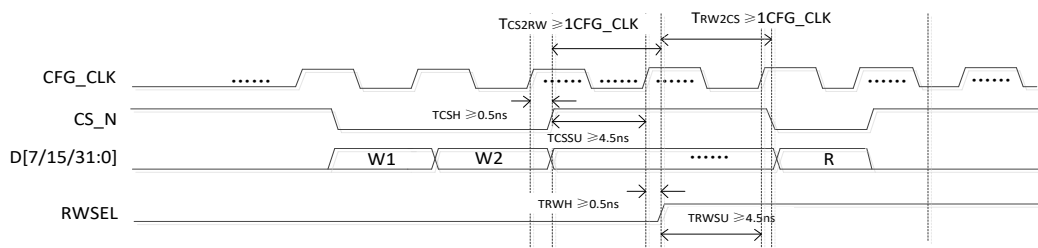


图 2-22 Slave Parallel 回读时序图

2.4.1 Slave Parallel 并行菊花链

Slave Parallel配置模式的典型应用是用于并行菊花链。所有器件可以全使用Slave Parallel模式进行并行级联配置。级联配置时，从最末级器件（最远端）开始配置，最后配置第一片器件。位流需要使用Fabric Configuration工具合成（使用Fabric Configuration工具生成位置：Operations->Generate Daisy Chain File）。

并行级联可以通过一个主控芯片（Host）控制板上多个级联芯片的配置。连接如图 2-23 所示。

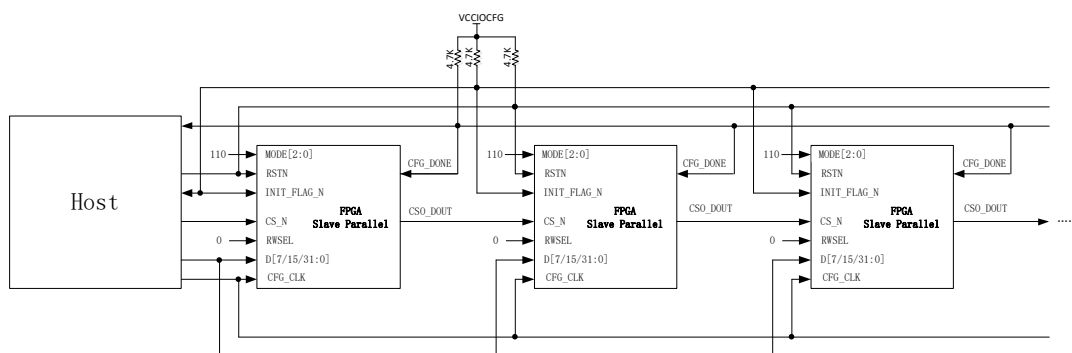


图 2-23 Slave Parallel 配置模式并行菊花链连接

3 下载配置

3.1 配置过程

Logos2系列FPGA的下载配置过程如图 3-1所示：

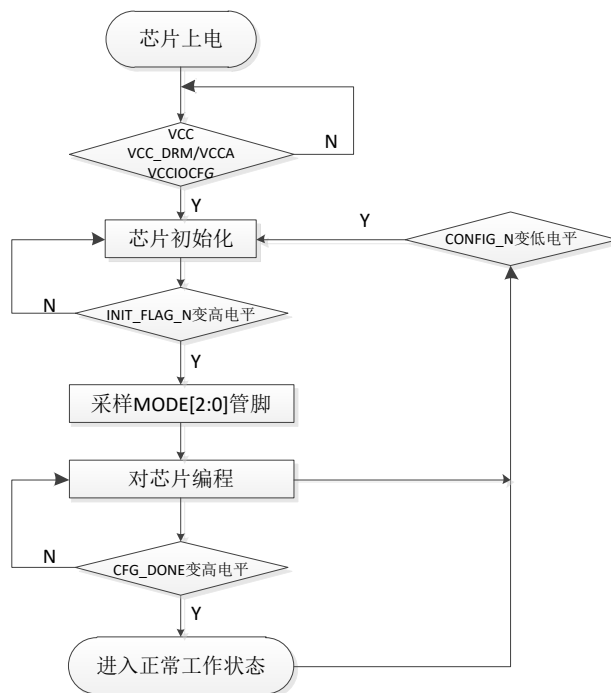


图 3-1 下载配置流程图

对于所有的配置接口，基本的配置步骤是相同的，包括建立，位流加载以及唤醒。

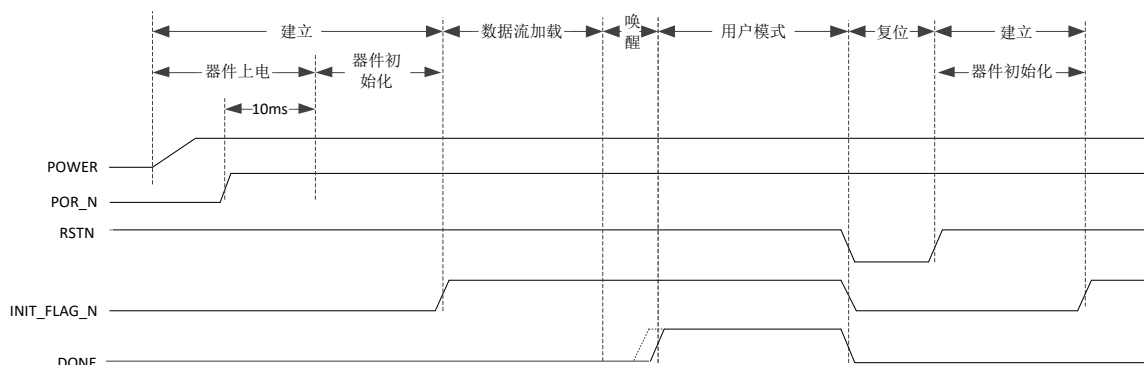


图 3-2 下载配置时序

Logos2 FPGA的配置过程包括如下几个阶段。

3.1.1 建立

建立完成FPGA器件初始化和配置模式确定的操作，包括器件上电、器件初始化和配置模式选择。

3.1.1.1 器件上电

器件上电后，配置控制系统的上电复位电路部分开始工作，其它电路部分仍处于复位状态。上电复位电路利用器件内部晶振产生的配置控制系统主时钟CLK进行计数，计数10ms后，表示内部晶振产生的时钟已经稳定，此时配置控制系统上电复位释放，配置控制系统开始工作。

3.1.1.2 器件初始化

器件上电，配置控制系统复位释放，配置控制系统开始工作后，首先进行器件初始化操作。

器件初始化完成两个操作：清空配置存储器和读eFuse内容。初始化完成后，标志信号INIT_COMPLETE变高。

可以通过维持外部的INIT_FLAG_N管脚为低电平，来维持配置控制系统继续处于器件初始化过程。

当复位，热启动，版本回退中的一种功能触发时，配置控制系统重新进入器件初始化过程。

3.1.1.3 配置模式选择

当INIT_FLAG_N信号为高后，采样配置模式管脚MODE[2:0]，选择器件的配置模式。模式采样完成后，CCS选择复用管脚的配置功能。

3.1.2 位流加载

不同配置模式具有不同配置接口。配置模式选择完成后，相应的模式所对应的管脚被设置为配置管脚。本阶段包括总线位宽检查、同步、器件标志检查、加载前选项设置、数据加载、CRC、加载后选项设置等操作。

3.1.2.1 总线位宽自动检测

Slave Parallel配置模式时，数据总线位宽由CCS自动进行检测，通过检测由两个字组成的总线位宽自动检测序列000000AA 08100020，进行位宽匹配。

总线位宽自动检测序列在位流的开头部分，在同步字之前。CCS只检测并行数据总线的低8位。CCS进行检测时，先在数据总线低八位检测0xAA字节，检测到0xAA后，继续检测下一个字节是否是0x08(x8)，0x10 (x16)或0x20(x32)。如果下一个字节不是上述三个中的一个，CCS会重新检测0xAA，直到检测到0xAA + 0x08/0x10/0x20序列。之后，CCS设置相关的数据管脚为配置数据引脚，数据总线切换到检测到的位宽，开始寻找同步字。

8位总线

FF	填充字
FF	
FF	
FF	
00	总线位宽自动检测序列
00	
00	
AA	
08	
10	
00	
20	
FF	填充字
FF	
FF	
FF	
...	

16位总线

FFFF	填充字
FFFF	
0000	总线位宽自动检测序列
00AA	
0810	
0020	
FFFF	填充字
FFFF	
...	

32位总线

FFFFFFFF	填充字
000000AA	总线位宽自动检测序列
08100020	
FFFFFFFF	填充字
...	

3.1.2.2 同步

同步字0x01332D94用于32位字边界对齐。同步后，才能进行后续解帧操作。

当接收到正确的同步字后，CCS才认为数据是有效数据；CCS的包处理器才进行解包操作；除了总线位宽自动检测序列，其它任何在同步字之前的数据都将被忽略。

3.1.2.3 主模式读操作选择

Master SPI配置模式时，对读操作进行选择。

Master SPI配置模式时，选择完成后，器件向SPI Flash重新发送读命令，并根据读命令重新调整数据总线位宽。

3.1.2.4 复位 CRC

复位CRC寄存器。

3.1.2.5 器件标识检查

检查位流中的器件标识是否与硬件匹配，如果不匹配，INIT_COMPLETE信号变低，ID错误标志存入状态寄存器，跳出配置过程。

3.1.2.6 加载前选项设置

加载配置数据前，在配置寄存器中指定器件的工作状态：

看门狗设置

解密初始向量设置

密钥选择

解密选择

设置是否保留复用的配置端口在配置完成后继续作为配置端口使用

唤醒时钟选择

唤醒时序设置

唤醒时是否等待PLL

版本回退设置

主模式时钟频率设定

3.1.2.7 加载配置数据

将配置数据帧写入配置存储器。

3.1.2.8 加载后 CRC 检查

配置数据加载完成后，进行CRC校验。软件生成的CRC值存放在CRC寄存器中，与CCS计算的CRC值进行比较。如果两个值一致，表示CRC检查通过，所有配置信息正确写入了配置存储器中。如果两个值不一致，表示CRC检查不同过，那么INIT_N信号变为低，CRC错误

标志存入状态寄存器。

3.1.2.9 加载后选项设置

配置数据加载完成后，在配置寄存器中指定器件的工作状态：

设置是否允许用户逻辑关断OSC

外部端口安全级别设置

3.1.3 唤醒

位流加载完毕并通过CRC后，FPGA进入唤醒阶段。配置系统先使能FPGA内部所有功能模块的逻辑输出。唤醒前再次CRC校验，然后进行去同步操作，表示配置操作结束，后续操作需要重新同步。最后完成唤醒，逐步释放相应的全局信号。

3.1.3.1 全局逻辑使能

加载后CRC检查通过后，使能芯片内部所有功能模块的逻辑输出。

3.1.3.2 唤醒启动

全局逻辑使能后，启动唤醒操作。

3.1.3.3 唤醒前 CRC 检查

唤醒操作执行前，进行CRC校验。软件生成的CRC值存放在CRC寄存器中，与CCS计算的CRC值进行比较。如果两个值一致，表示CRC检查通过，唤醒前准备工作完成。如果两个值不一致，表示CRC检查不同过，那么INIT_FLAG_N信号变为低，CRC错误标志存入状态寄存器。

3.1.3.4 去同步

表示配置过程结束。后续操作需要重新同步。

3.1.3.5 唤醒

去同步完成后，唤醒电路按照设定的时序开始工作，逐步释放相应的全局信号。PDS中可以设置唤醒时序（Project->Project Setting->Generate Bitstream->Startup，具体操作见文档《Pango_Design_Suite_User_Guide》），下文对各个全局信号进行介绍（基于该唤醒顺序：DONE-T1、GOUTEN-T2、GWEN-T3、GRS_N-T4 进行介绍，具体默认唤醒时序以PDS为准）。

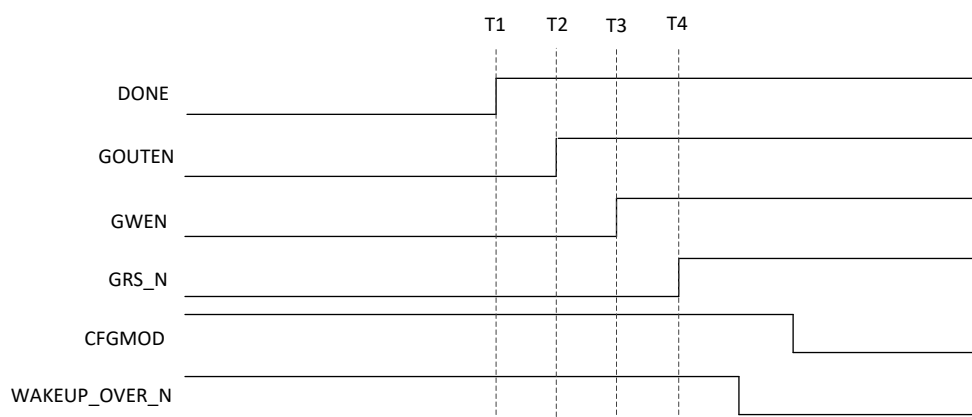


图 3-3 唤醒时序

全局逻辑使能信号GLOGEN输出到fabric，使能各逻辑块；在最远端反馈回CCS。CCS接收到反馈信号GLOGEN_FB后，进入唤醒时序。

在配置完成之前，全局寄存器置位复位信号GRS_N为低电平，控制全芯片各模块的GRS_N端口，使各模块的寄存器处于置位复位状态；配置完成后，全芯片各模块的GRS_N信号释放，使各模块的寄存器进入非全局置位复位状态。

在配置完成之前，全局IO输出使能信号GOUTEN为低电平，所有IO的输出处于高阻状态。在配置完成后，使能所有IO的输出。

表 3-1 GOUTEN 逻辑功能

GOUTEN	0	1
用户使用的 IO	High-Z	用户逻辑控制
用户未使用的 IO	High-Z	位流设置 ⁽¹⁾

注：(1)PDS 工具中设置位置为 Project->Project Setting->Generate Bitstream->General->Unused IO Status, 默认值为 UNUSED, 即 High-Z。

全局写使能信号GWEN是FPGA芯片内部存储资源的全局写使能。在配置完成之前，该信号为低电平，禁止FPGA存储资源的写操作。

复用管脚控制信号CFGMOD将配置过程中使用的复用管脚释放给用户使用。

唤醒完成指示信号WAKEUP_OVER_N指示唤醒过程成功结束。

唤醒结束后，FPGA由配置模式进入用户模式。

如果用户需要重新进行位流加载，可以把RSTN拉低一段时间后释放，FPGA会重复进行上诉过程。

3.1.3.6 延迟唤醒

全局唤醒信号GOUTEN, GRS_N, GWEN和DONE置1的时刻用户可配, 可在时刻T1, T2, T3或T4置1。

两个时刻之间的周期数用户可配。

配置完成指示信号DONE输出驱动CFG_DONE管脚。CFG_DONE置1后, 解除对该管脚的驱动控制; CFG_DONE管脚是开漏输出管脚, 通常情况下外部接上拉电阻。如果DONE信号变高后, CFG_DONE管脚没有变高(外部驱动为低), 那么唤醒过程暂停, 直到CFG_DONE管脚变高后, 才继续完成后续的唤醒过程。

DONE置1后, 需要等待外部信号CFG_DONE, 外部CFG_DONE驱动为高后才能进行后续的操作。在DONE置1后置1的信号需要等待CFG_DONE为高, 在DONE置1前以及和DONE同时置1的信号不受CFG_DONE外部驱动的影响。

比如设置GOUTEN在T1置1, DONE在T2置1, GWEN在T3置1, GRSN在T4置1。那么GOUTEN, DONE按照设定的时序置1, 在T3时刻时, 如果CFG_DONE外部驱动为1, 那么GWEN置1, 否则GWEN延迟一个时隙(两个时刻之间的周期数可配), 在下一个时隙继续检测CFG_DONE。如果CFG_DONE为高, 则GWEN置1, 否则继续延迟一个时隙。GRSN的置1时刻根据GWEN的置1时刻延迟顺延。

3.2 配置文件

3.2.1 位流生成

PDS设计软件会产生不同扩展名的配置文件来适应不同的配置方案, 具体描述见[表 3-2](#)。

表 3-2 PDS 配置文件描述

配置文件扩展名	描述
.sbit	包含头信息(位流名称、日期等)的二进制配置数据, 配置工具会识别头信息, 但不会把头信息写入FPGA。用 Fabric Configuration 工具通过 Cable 可直接把.sbit 文件写入FPGA
.bin	无头信息的二进制配置数据(纯位流), 适用于用户配置方案, 如微处理器对FPGA进行配置。
.sfc	写入Flash的位流文件, 由.sbit文件转换而来

.sbit文件是PDS工具编译完成后生成的默认位流文件, 如[图 3-4](#)所示, 运行Generate Bitstream即可生成.sbit文件。

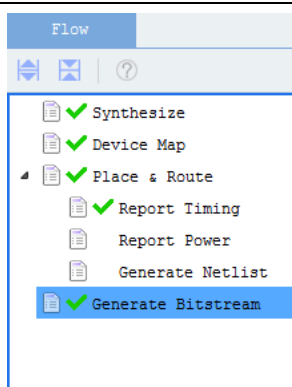


图 3-4 生成位流示意图

图 3-4 中鼠标右击 Generate Bitstream，选择 configure，弹出如图 3-5 所示，勾选 Create Bin File，点击 ok 后，重新运行 Generate Bitstream，则同时生成 .sbit 和 .bin 文件。

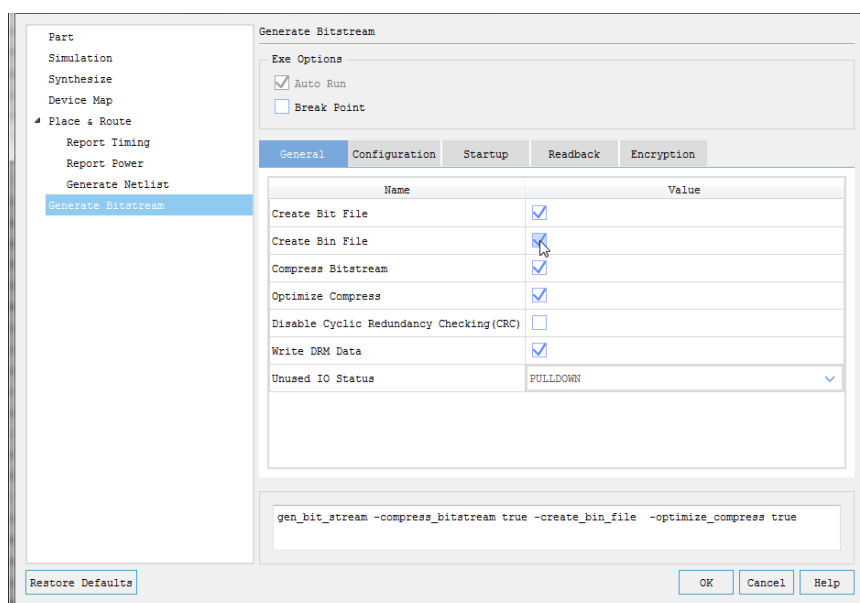


图 3-5 Configure Generate Bitstream

在 Fabric Configuration 工具界面，点击菜单栏的 Operations，单击下拉菜单中的 Generate Flash Programming File，弹出如图 3-6 所示界面。

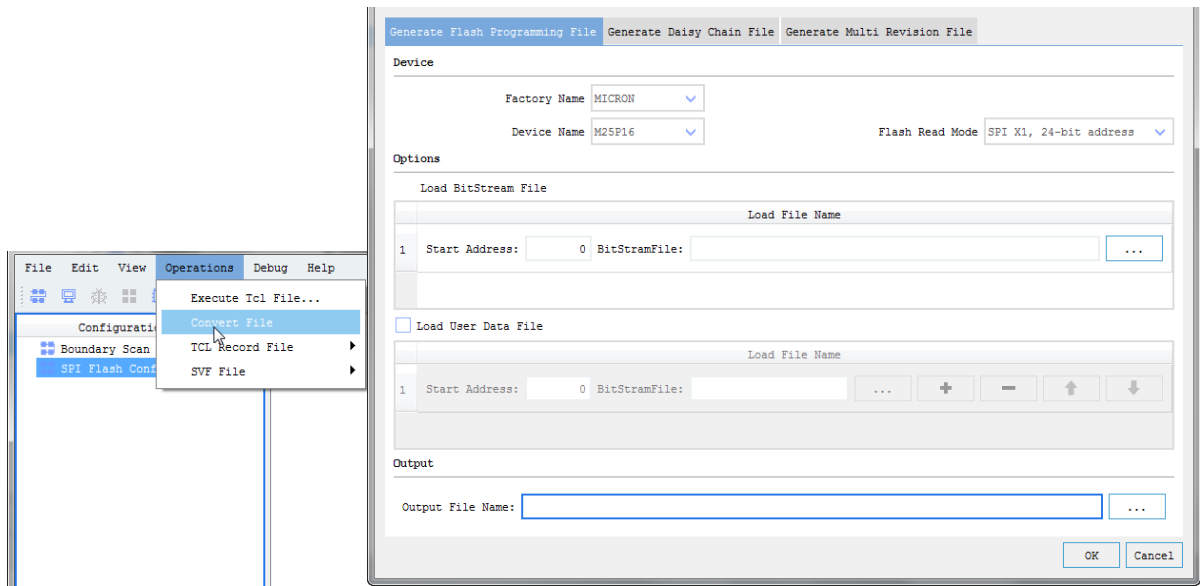


图 3-6 Generate Flash Programming File

用户根据实际应用情况，选择对应的Flash Device，在Flash Read Mode下拉框中选择Flash数据位宽和地址位宽，然后填入sbit文件路径和sfc文件存放路径，点击ok即可生成Flash配置文件—sfc文件。

3.2.2 位流大小

Logos2系列各器件的位流大小如表 3-3所示。压缩后的位流文件比压缩前更小，压缩文件的大小跟设计有关，这里不做说明。

表 3-3 Logos2 系列各器件的位流大小

器件	文件名	非压缩文件大小（MByte）
PG2L100H	*.sbit	3.61

3.2.3 不同接口位宽下的位序

Logos2系列FPGA配置支持多种位宽，同一位宽的不同配置模式，其位序，即数据总线的高低位与位流顺序的对应关系一致。如Master SPI（x8）与Slave Parallel（x8）模式的位序一样。以位流中的同步字（32'h01332D94）为例，数据总线的高低位与位流顺序如表 3-4所示。

表 3-4 不同位宽下同步字发送顺序

时钟周期	进制	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
D[1:0](x2)	2'b	00	00	00	01	00	11	00	11	00	10	11	01	10	01	01	00
D[3:0](x4)	4'h	0	1	3	3	2	D	9	4								
D[7:0](x8)	8'h	01	33	2D	94												
D[15:0](x16)	16'h	0133	2D94														
D[31:0](x32)	32'h	01332D94															

注：表中 D[x:0]仅表示数据总线高低位，具体对应管脚需参考不同接口模式下的端口列表。

3.3 配置数据包

3.3.1 包格式

Logos2系列FPGA配置位流的包有两种类型：类型1和类型2。

类型1包用于寄存器读写。类型1包由包头部分和数据部分组成。

类型1包的包头是一个32位字。包头后面跟数据。数据以32位字为单位。

如果包头部分指定的包头后面跟的32位字的数量为0，那么这个包只有包头部分没有数据部分，是空包。

类型1包头格式如表 3-5所示。

表 3-5 类型 1 包头格式

[31:29]	[28:27]	[26:22]	[21:0]
包头类型	操作码	寄存器地址	32 位字个数
101	00:无操作；01:写；10:读；11:保留	寄存器的地址	包头后面跟的 32 位字的数量

类型2包用于读写配置存储器。类型2包由包头部分和数据部分组成。

类型2包的包头是一个32位字。包头后面跟数据。数据以32位字为单位。

类型2包必须跟在类型1空包之后。类型2包不指定寄存器地址，用前面的类型1空包指定的地址。

类型2包头格式如表 3-6所示。

表 3-6 类型 2 包头格式

[31:29]	[28:27]	[26:0]
包头类型	操作码	32 位字个数
010	00:无操作；01:写；10:读；11:保留	包头后面跟的 32 位字的数量

3.3.2 配置寄存器

Logos2系列FPGA的配置寄存器如表 3-7所列。

表 3-7 配置寄存器说明

名称	R/W	地址	描述
CRCR	R/W	00000	CRC 寄存器
IDR	R/W	00001	器件标识寄存器
CMDR	R/W	00010	命令寄存器
CTRL0R	R/W	00011	控制寄存器 0
CTRL1R	R/W	00100	控制寄存器 1
CMEMIR	W	00101	帧数据输入寄存器
MFWRITER	W	00110	多帧写寄存器
CMEMOR	R	00111	帧数据输出寄存器

名称	R/W	地址	描述
IVR	W	01000	初始向量寄存器
STATUSR	R	01001	状态寄存器
CHAINR	W	01010	级联寄存器
ADRR	R/W	01011	帧地址寄存器
SBPIR	R/W	01100	SBPI 寄存器
SEUR	R/W	01101	SEU 控制寄存器
SEUSTATUSR	R	01110	SEU 状态寄存器
IRSTCTRLR	R/W	01111	热启动控制寄存器
IRSTADDR	R/W	10000	热启动地址寄存器
WATCHDOGR	R/W	10001	看门狗寄存器
HSTATUSR	R	10010	历史状态寄存器
CMASKR	R/W	10111	控制屏蔽寄存器
OPTION0R	R/W	11001	选项寄存器 0
OPTION1R	R/W	11010	选项寄存器 1
SEUADDR	R	11101	SEU 帧地址寄存器
SEUNADDR	R	11111	SEU 下帧地址寄存器

各配置寄存器的详细描述如下。

3.3.2.1 CRC 寄存器（CRCR）

每写一次 CRC 寄存器，对应一次位流的 CRC 检查。如果写入 CRC 寄存器的值与 CCS 计算出的当前 CRC 值相同，则 CRC 检查通过。否则 `init_complete` 拉低。

3.3.2.2 器件标识寄存器（IDR）

器件标识。载入位流之前要向 ID 寄存器写入 IDCODE。写入的 IDCODE 与器件 IDCODE 匹配，后续操作才能继续进行。每写一次 ID 寄存器，对应一次器件 ID 检查。如果写入 ID 寄存器低 28 位的值与器件的 ID 低 28 位的值相同，则 ID 检查通过。否则 `init_complete` 拉低。

表 3-8 Logos2 系列 FPGA 器件型号

器件型号	器件标识
PG2L100H	X0602899

3.3.2.3 命令寄存器（CMDR）

命令寄存器决定配置控制系统接下来进行的操作。

每次操作最多向命令寄存器中写一条命令。不能一个包头后跟多条命令。

表 3-9 CCS 命令及其描述

命令	编码	描述
NOP	00000	无操作

命令	编码	描述
RSTCRC	00001	复位 CRC 复位 CRC 寄存器
SWITCH	00010	切换主模式配置时钟频率 启动更新主模式配置时钟频率操作，数值由 CTRL0R 中的 oscfsel 决定
WCMEM	00100	写配置数据 用于经 CMEMIR 写入配置存储器配置数据之前
MFWRITE	00101	多帧写 将当前写入的一帧数据写入到接下来的连续帧中。帧的数量由 MFWRITER 决定
RCMEM	00110	读配置数据 用于经 CMEMOR 回读配置存储器配置数据之前
SWAKEUP	00111	开始唤醒操作
SWAKEDOWN	01000	关闭唤醒操作
GUP	01001	内部逻辑使能
GDOWN	01010	内部逻辑不使能
DESYNC	01011	去同步 用于配置结束时
RWD	01100	重起看门狗
RRBCRC	01101	复位回读 CRC
RBCRC	01110	回读 CRC 检查
IRST	01111	热启动 版本回退时，热启动无效
WCMEMDIS	10000	写配置存储器禁止 用于配置数据写完后，禁止配置存储器写
RCMEMDIS	10001	读配置存储器禁止 用于配置数据读完后，禁止配置存储器读

3.3.2.4 控制寄存器 0 (CTRL0R)

表 3-10 控制寄存器 0 说明

位	名称	初始值	描述
[31:23]	保留		
[22]	otsd_en	1'b0	过温度关断使能 0: 不使能 1: 使能 过温度时(ADC 给 CCS 的 over_temperature 信号由 0 变为 1)，如果过温度关断使能，那么 CCS 使用内部主配置时钟 imclk 进行计数（如果时钟频率是 50M，那么计数时间为 10ms），用户要在计数的这段时间内结束所有配置接口（包括内部从并行接口）的操作。计数完成后，启动关断流程（先根据关断时序关断全局信号，再关断全局逻辑使能信号 glogen）。后续温度降到正常后(ADC 给 CCS 的 over_temperature 信号由 1 变为 0)，启动唤醒流程（先将全局逻辑使能信号 glogen 置 1，再根据唤醒时序唤醒全局信号）。关断和唤醒流程都工作在内部主配置时钟 imclk 的时钟域。
[21:5]	保留		
[4]	fallback_en	1'b1	版本回退使能

位	名称	初始值	描述
[3]	保留		
[2]	persist	1'b0	用户模式下配置接口（Master SPI 接口，Slave Parallel 接口，Slave Serial 接口）保留选择 1'b0：释放给用户使用 1'b1：保留继续用于配置接口 用户模式下，内部从并行接口使能时，Slave Parallel 接口，Slave Serial 接口不能保留继续用于配置接口
[1]	保留		
[0]	dec_en	1'b0	解密使能

3.3.2.5 控制寄存器 1（CTRL1R）

表 3-11 控制寄存器 1 说明

位	名称	初始值	描述
[31:5]	保留		
[4]	mask_en	1'b0	可变存储单元回读屏蔽使能 0：不屏蔽，可回读 1：屏蔽
[3]	mfg_por_off	1'b0	0：正常监控 VCC、VCCA、VCCIOCFG、VCC_DRM 电压 1：需要测试 SRAM retention 电压时，当 gwen 为 1 时，关闭 VCC 和 VCC_DRM 的细探测功能。此时，VCC<0.65V 或 VCC_DRM<0.5V 时，POR 才会重新 assert。
[2]	osc_off_en	1'b0	控制是否允许用户逻辑关断 OSC 0：不允许 1：允许
[1:0]	wrctrl	2'b00	外部端口安全级别。控制关断重配，部分重配，局部动态重配，回读配置存储器。配置为禁止后，不能再更改为使能，除非复位 00：重配使能，回读使能（默认） 01：重配使能，回读禁止 1x：重配禁止，回读禁止

3.3.2.6 控制屏蔽寄存器（CMASKR）

控制屏蔽寄存器用来屏蔽控制寄存器0和控制寄存器1中的对应位，默认值为32'd0。0为屏蔽。例如控制屏蔽寄存器的值为32'h0000_0003时，控制寄存器0和控制寄存器1的低2位能够写入数据，其它位不能写入数据。

3.3.2.7 选项寄存器 0（OPTION0R）

表 3-12 选项寄存器 0 说明

位	名称	初始值	描述
[31:24]	保留		
[23]	done_syn	1'b0	外部 done 同步使能。指示外部输入 done 是否是输出 done 的同步信号。如果是同步信号，CCS 的唤醒模块直接使用。如果是异步信号，使用前需要同步，通常用于延迟 done 功能。 0:不同步

位	名称	初始值	描述
			1: 同步
[22:21]	t_sel	2'd0	唤醒周期长度选择, 即 T1, T2, T3 之间的时间间隔 2'b00: 1 2'b01: 2 2'b10: 4 2'b11: 1
[20:19]	done_sel	2'd0	done 信号拉高时刻选择 2'd0: T3 2'd1: T1 2'd2: T2 2'd3: T4
[18:17]	gwen_sel	2'd0	gwen 信号拉高时刻选择 2'd0: T2 2'd1: T1 2'd2: T3 2'd3: T4
[16:15]	grsn_sel	2'd0	grsn 信号拉高时刻选择 2'd0: T2 2'd1: T1 2'd2: T3 2'd3: T4
[14:13]	gouten_sel	2'd0	gouten 信号拉高时刻选择 2'd0: T1 2'd1: T2 2'd2: T3 2'd3: T4
[12]	保留	1'b0	
[11]	wait_pll	1'b0	唤醒时等待 PLL 锁定使能 0: 不需要等待 PLL 锁定 1: 需要等待 PLL 锁定
[10]	保留	1'b0	
[9]	startup_uclk_en	1'b0	用户时钟唤醒使能 0: 使用配置时使用的时钟进行唤醒 1: 使用用户时钟进行唤醒
[8:7]	保留	3'd0	
[6:0]	oscfssel	7'd0	主模式配置时钟 CFG_CLK 频率选择

3.3.2.8 选项寄存器 1 (OPTION1R)

表 3-13 选择寄存器 1 说明

位	名称	初始值	描述
[31:2]	保留		
[1]	rbcrc_disable	1'b0	禁止回读 CRC 错误指示 0: 使能 1: 禁止
[0]	crc_disable	1'b0	禁止 CRC 检查 0: 使能 1: 禁止

3.3.2.9 帧数据输入寄存器（CMEMIR）

配置存储器配置数据接口。数据经 CMEMIR 写入配置存储器。

3.3.2.10 多帧写寄存器（MFWRITER）

压缩的帧数目，表示后面连续几帧的数据内容与刚写完的一帧数据内容相同。比如配置完一帧数据后，如果后面连续三帧的数据内容与这一帧相同，那么多帧写寄存器内容需要写为 3。

3.3.2.11 帧数据输出寄存器（CMEMOR）

配置存储器回读数据接口。数据经 CMEMOR 从配置存储器回读出。

3.3.2.12 初始向量寄存器（IVR）

内容为加密数据的初始向量。

3.3.2.13 状态寄存器（STATUSR）

表 3-14 状态寄存器说明

位	名称	描述
[31:30]	保留	
[29]	prcfg_over	局部重配完成标志
[28]	prcfg_err	局部重配错误标志
[27]	over_temp	过温度标志
[26]	flg_x32	Slave Parallel 模式 32 位数据位宽指示
[25]	flg_x16	Slave Parallel 模式 16 位数据位宽指示
[24]	flg_x8	Slave Parallel 模式 8 位数据位宽指示
[23:22]	ipal_m[1:0]	内部从并行接口数据位宽选择
[21]	fallback	回退指示标志
[20]	保留	
[19]	pll_lock	PLL 锁定标志
[18]	gwen	全局写使能
[17]	grsn	全局寄存器置位复位
[16]	gouten	全局 IO 输出使能
[15]	glogen_fb	全局逻辑使能反馈
[14]	glogen	全局逻辑使能
[13]	done_i	DONE 管脚输入
[12]	done	器件唤醒成功标志
[11]	init_n	INIT_FLAG_N 管脚输入
[10]	init_complete	初始化完成和配置错误指示
[9:7]	m[2:0]	模式选择
[6]	wakedown_over	唤醒关断结束

位	名称	描述
[5]	wakeup_over	唤醒结束
[4]	timeout	看门狗超时
[3]	rbcrc_err	回读 CRC 检测结果 0: CRC 正确 1: CRC 错误
[2]	aut_err	认证结果 0: 认证通过 1: 认证失败
[1]	crc_err	CRC 检测结果 0: CRC 正确 1: CRC 错误
[0]	id_err	ID 检测结果 0: 正确 1: 错误

3.3.2.14 看门狗寄存器（WATCHDOGR）

用于配置模式和用户模式下的超时检测。

表 3-15 看门狗寄存器说明

位	名称	初始值	描述
[31]	wd_user_en	1'b0	用户模式下看门狗使能
[30]	wd_cfg_en	1'b0	配置模式下看门狗使能
[29:0]	wd_value	30'h3FFF_FFFF	看门狗超时数值

3.3.2.15 帧地址寄存器（ADRR）

提供配置和回读数据的起始地址。只能发有效地址，不能超范围。

表 3-16 帧地址寄存器说明

位	名称	初始值	描述
[31: 27]	保留		
[26: 25]	type	2'd0	配置存储器内容类型 00: 非 DRM 内容 01: DRM 内容 10: 保留 11: 保留
[24: 20]	addr_region	5'd0	区域地址
[19: 18]	保留		
[17: 10]	addr_column	8'd0	列地址
[9: 8]	保留		
[7: 0]	addr_frame	8'd0	帧地址

3.3.2.16 SBPI 寄存器（SBPIR）

提供 Master SPI 模式控制和选项信息。

表 3-17 SBPI 寄存器说明

位	名称	初始值	描述
[31]	emclk_en	1'b0	外部主时钟使能 1'b0: 主模式配置时钟使用内部时钟 1'b1: 主模式时钟使用外部时钟

位	名称	初始值	描述
[30]	sbpi_rf_sel	1'b0	数据采样时钟沿选择 0: 上升沿 1: 下降沿 Master SPI 模式时, 下降沿采样为快速模式
[29: 11]	保留		
[10]	addrwidth	1'b0	SPI 地址位宽 0: 24 位地址 1: 32 位地址
[9: 8]	datawidth	2'd0	SPI 数据位宽 00: X1 01: X2 10: X4 11: X8
[7: 0]	opcode	8'h0B	SPI 主模式操作码

3.3.2.17 SEU 控制寄存器 (SEUR)

提供 SEU 检错纠错和回读 CRC 控制。

表 3-18 SEU 控制寄存器说明

位	名称	初始值	描述
[31: 2]	保留		
[1]	seu_en	1'b0	SEU 使能
[0]	rbcrc_en	1'b0	回读 CRC 使能

3.3.2.18 SEU 状态寄存器 (SEUSTATUSR)

表 3-19 SEU 状态寄存器说明

位	名称	描述
[31:17]	保留	
[16]	drcfg_over	动态重配完成标志
[15]	drcfg_err	动态重配错误标志
[14:3]	seu_index	SEU 单比特错误在帧中的位置指示
[2]	seu_ded	SEU 双比特错误标志
[1]	seu_sec	SEU 单比特错误标志
[0]	seu_over	SEU 完成标志

3.3.2.19 SEU 帧地址寄存器 (SEUADDR)

表 3-20 SEU 帧地址寄存器说明

位	名称	描述
[31:21]	保留	
[20:16]	seu_region_naddr	SEU 当前帧的区域地址
[15:8]	seu_column_naddr	SEU 当前帧的列地址
[7:0]	seu_frame_naddr	SEU 当前帧的帧地址

3.3.2.20 SEU 下帧地址寄存器（SEUNADDR）

表 3-21 SEU 下帧地址寄存器说明

位	名称	描述
[31:21]	保留	
[20:16]	seu_region_naddr	SEU 下一帧的区域地址
[15:8]	seu_column_naddr	SEU 下一帧的列地址
[7:0]	seu_frame_naddr	SEU 下一帧的帧地址

3.3.2.21 热启动控制寄存器（IRSTCTRLR）

表 3-22 热启动控制寄存器说明

位	名称	初始值	描述
[31:1]	保留		
[0]	vback_en	1'b0	版本回退使能。控制回退时，回退到黄金位流还是上一版本位流 0：禁止 1：使能

3.3.2.22 热启动地址寄存器（IRSTADDR）

热启动时 FLASH 首地址，字节寻址。

表 3-23 热启动地址寄存器说明

位	名称	初始值	描述
[31:24]	FLASH 首地址	8'd0	FLASH 首地址[31:24]位，用于 4-Byte 模式
[23:0]	FLASH 首地址	24'd0	FLASH 首地址

3.3.2.23 历史状态寄存器（HSTATUSR）

表 3-24 历史状态寄存器说明

位	名称	描述
[31:15]	保留	
[14]	timeout1	上个状态 timeout 0：看门狗未超时 1：看门狗超时
[13]	rbcrc_err1	上个状态回读 CRC 错误 0：正确 1：错误
[12]	crc_err1	上个状态 CRC 错误 0：正确 1：错误
[11]	id_err1	上个状态 ID 错误 0：正确 1：错误
[10]	irst1	上个状态的版本为热启动升级版本指示 0：非热启动升级版本 1：热启动升级版本

位	名称	描述
[9]	fallback1	指示上个状态是否为 fallback 操作 0: 否 1: 是
[8]	valid1	上个状态有效 0: 无效 1: 有效
[7]	1'b0	保留
[6]	timeout0	当前状态 timeout 0: 看门狗未超时 1: 看门狗超时
[5]	rbcrc_err0	当前状态回读 CRC 错误 0: 正确 1: 错误
[4]	crc_err0	当前状态 CRC 错误 0: 正确 1: 错误
[3]	id_err0	当前状态 ID 错误 0: 正确 1: 错误
[2]	irst0	当前状态的版本为热启动升级版本指示 0: 非热启动升级版本 1: 热启动升级版本
[1]	fallback0	指示当前状态是否为 fallback 操作 0: 否 1: 是
[0]	valid0	当前状态有效 0: 无效 1: 有效

3.4 位流格式

本节列举普通位流格式。

表 3-25 普通位流格式

内容	含义
FFFFFFFF	填充字（100 个）
.....	
FFFFFFFF	
000000AA	总线位宽自动检测
08100020	
FFFFFFFF	填充字（10 个）
.....	
FFFFFFFF	
01332D94	同步
AB000001	类型 1 包头：写 SBPIR 寄存器

内容	含义
XXXXXXXX	数据：SBPI 主模式配置控制选项
A0000000	无操作类型 1 包头（10 个）
.....	
A0000000	
ABC00001	类型 1 包头：写 IRSTCTRLR 寄存器
XXXXXXXX	数据：IRSTCTRLR 的内容
AC000001	类型 1 包头：写 IRSTADRR 寄存器
XXXXXXXX	数据：IRSTADRR 的内容
A8800001	类型 1 包头：写 CMDR 寄存器
00000000	数据：NOP 命令
A8800001	类型 1 包头：写 CMDR 寄存器
00000001	数据：RSTCRC 命令
A8400001	类型 1 包头：写 IDR 寄存器
XXXXXXXX	数据：IDCODE
AC400001	类型 1 包头：写 WATCHDOG 寄存器
XXXXXXXX	数据：WATCHDOG
ADC00001	类型 1 包头：写 CMASKR 寄存器
XXXXXXXX	数据：CMASKR 的内容
A8C00001	类型 1 包头：写 CTRL0R 寄存器
XXXXXXXX	数据：CTRL0R 的内容
AE400001	类型 1 包头：写 OPTION0R 寄存器
XXXXXXXX	数据：OPTION0R 的内容
AE800001	类型 1 包头：写 OPTION1R 寄存器
XXXXXXXX	数据：OPTION1R 的内容
A8800001	类型 1 包头：写 CMDR 寄存器
00000002	数据：SWITCH 命令
A0000000	无操作类型 1 包头（10 个）
.....	
A0000000	
AAC00001	类型 1 包头：写 ADRR 寄存器
XXXXXXXX	数据：ADRR 的内容
A8800001	类型 1 包头：写 CMDR 寄存器
00000004	数据：WCMEM 命令
A9400000	类型 1 包头：写 CMEMIR 寄存器
{3'b010, 2'b01, 27'dx}	类型 2 包头：写操作
XXXXXXXX	数据：写入配置存储器的内容
.....	
XXXXXXXX	
A0000000	无操作类型 1 包头（50 个）

内容	含义
.....	
A0000000	
A8800001	类型 1 包头：写 CMDR 寄存器
00000010	数据：WCMEMDIS 命令
A8800001	类型 1 包头：写 CMDR 寄存器
00000012	数据：VDDTUP 命令
A8000001	类型 1 包头：写 CRCR 寄存器
xxxxxxxx	数据：CRC 值
ADC00001	类型 1 包头：写 CMASKR 寄存器
xxxxxxxx	数据：CMASKR 的内容
A9000001	类型 1 包头：写 CTRL1R 寄存器
xxxxxxxx	数据：CTRL1R 的内容
A8800001	类型 1 包头：写 CMDR 寄存器
00000009	数据：GUP 命令
A8800001	类型 1 包头：写 CMDR 寄存器
00000007	数据：SWAKEUP 命令
A8000001	类型 1 包头：写 CRCR 寄存器
xxxxxxxx	数据：CRC 值
A8800001	类型 1 包头：写 CMDR 寄存器
0000000B	数据：DESYNC 命令
A0000000	无操作类型 1 包头（100 个）
.....	
A0000000	

3.5 下载线速率

当使用USB Cable进行下载时，包括JTAG下载以及对SPI Flash烧录，默认速率是10MHz。

下载速率可以在Fabric Configuration中设置，点击图标Connect To Server。

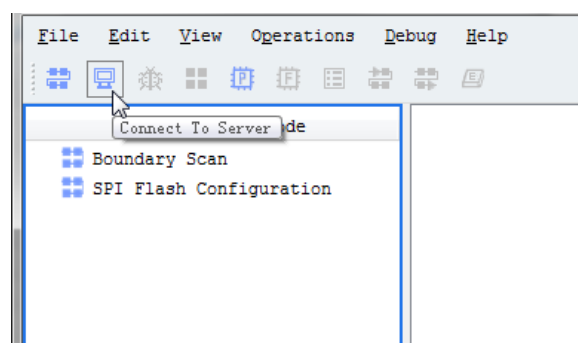


图 3-7 Fabric Configuration

若未连接，则先连接，如图 3-8左图所示，若已连接，则点击Next，如图 3-8右图所示：

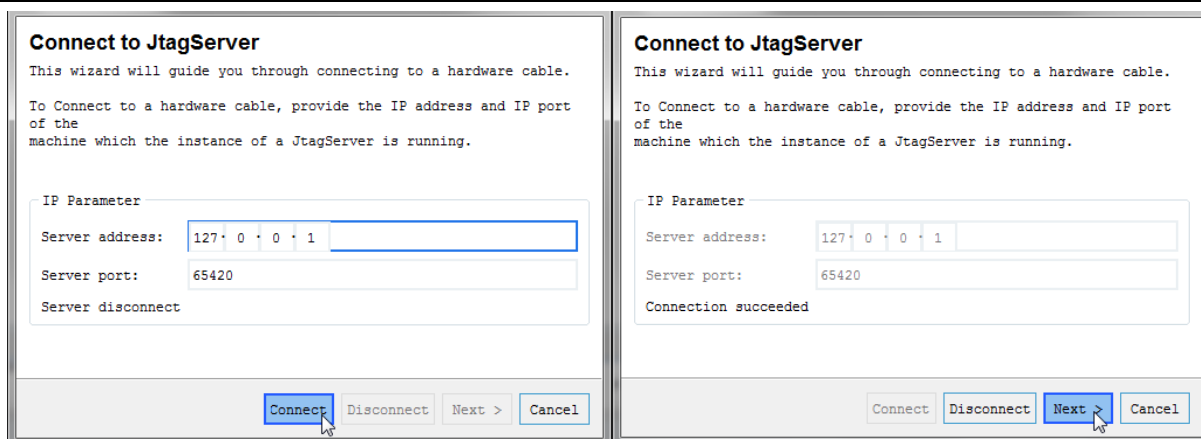


图 3-8 Connect To Cable

在TCK Frequency下拉框中可以选择USB Cable频率。

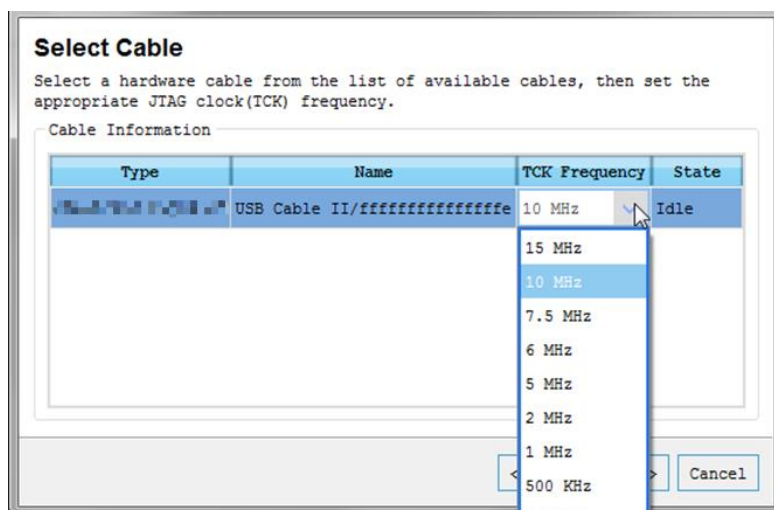


图 3-9 USB Cable 速率设置

3.6 配置速率

3.6.1 JTAG 模式

JTAG模式下载，速率最高可为50MHz。

由于PANGO USB CABLE II下载线速率限制，使用下载线下载时，最高配置频率为15MHz。

3.6.2 Master SPI 模式

Master SPI模式，CFG_CLK时钟频率可配置，默认为2.99MHz。低速模式支持最大配置频率15.38MHz，高速模式支持最大配置频率40MHz。Master SPI串行菊花链最高支持22.22MHz。

3.6.3 Slave Parallel 模式

Slave Parallel模式下，下载时钟来自外部Host设备，时钟速率最高可为80MHz。

3.6.4 Slave Serial 模式

Slave Serial模式下，下载时钟来自外部Host设备，最高可设为80MHz。Slave Serial菊花链最高速率为50MHz。

3.7 多启动

使用多启动功能时，外部Flash最少需要支持存放2套位流。位流0为黄金位流版本或应用位流版本，起始地址为0，其它位流为应用位流。配置过程中出现任何错误，复位版本回退电路以外的电路，重新载入黄金位流或上个版本应用位流。如果黄金位流或上一版本应用位流也有错误，那就不再复位了，而是在INIT_FLAG_N置0的同时，FCS_N置1，Master SPI模式操作结束。

使用多启动功能时，需要打开看门狗。

配置过程中，以下错误会触发版本回退。

- 1、器件 ID 错误
- 2、CRC 错误
- 3、看门狗超时
- 4、认证失败

3.7.1 黄金位流初始化系统

该应用场景下，FPGA会首先从黄金位流启动并进入用户模式，然后通过JTAG或者用户逻辑完成Flash的位流更新以及热启动跳转到应用位流操作。在软件上生成时，与下文应用位流初始化系统差异既是没有勾选插入内嵌热启动。（插入内嵌热启动设置：Fabric Configuration->Operations->Convert File->Generate Multi Revision File->Data Stream Type:Multi Boot Data Stream->Insert IRST CMD to Data Stream）

- 1、从Flash 0地址载入黄金位流。
- 2、JTAG或用户进行Flash操作。（可选）如果用于多启动的应用位流已经在Flash中，可以不进行Flash操作。
- 3、JTAG热启动。
- 4、从IRSTCTRLR寄存器指定的Flash地址载入应用位流。
- 5、如果载入应用位流过程中有错误，从Flash 0地址载入黄金位流或回退到上一版本应用位流。

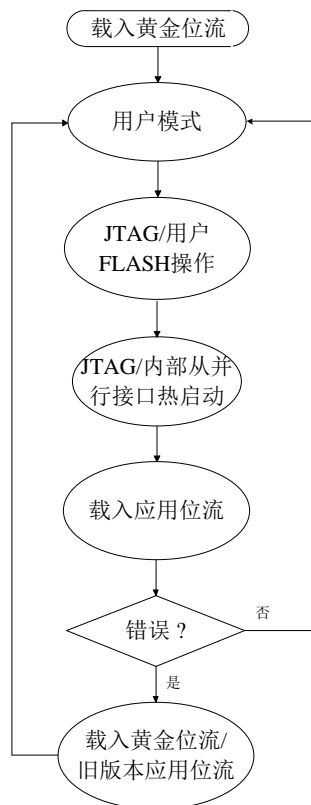


图 3-10 黄金位流加载流程

3.7.2 应用位流初始化系统

该应用场景下，FPGA会首先从指定的应用位流启动并进入用户模式，相比黄金位流初始化系统，不需要额外操作即可从应用位流启动，较为方便。在软件上生成时，与黄金位流初始化系统差异既是有勾选插入内嵌热启动。（插入内嵌热启动设置：Fabric

Configuration->Operations->Convert File->Generate Multi Revision File->Data Stream Type:Multi Boot Data Stream->Insert IRST CMD to Data Stream）

- 1、从Flash 0地址载入黄金位流。
- 2、黄金位流内嵌热启动使从IRSTCTRLR指定的Flash地址载入应用位流。
- 3、如果载入应用位流过程中有错误，从Flash 0地址载入黄金位流或回退到上一版本应用位流。

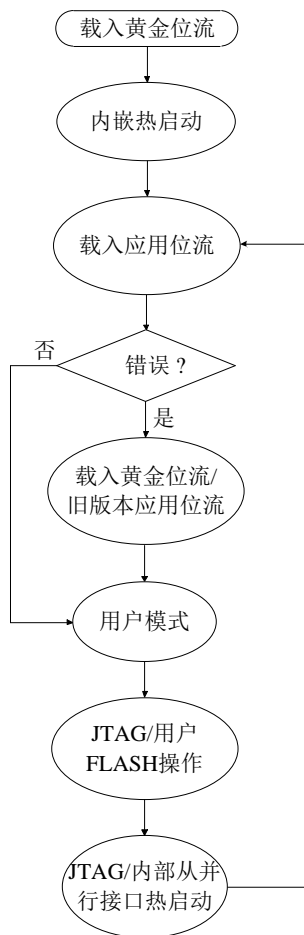


图 3-11 应用位流加载流程

3.7.3 无黄金位流

如果Flash最多支持存放2套位流，使能版本回退功能时，两套位流都为应用位流。

- 1、从Flash 0地址载入应用位流。
- 2、JTAG或用户进行Flash操作。（可选）如果用于多启动的应用位流已经在Flash中，可以不进行Flash操作。
- 3、JTAG热启动或内部从并行接口热启动。
- 4、从IRSTCTRLR寄存器指定的Flash地址载入应用位流。
- 5、如果载入应用位流过程中有错误，回退到上一版本应用位流。

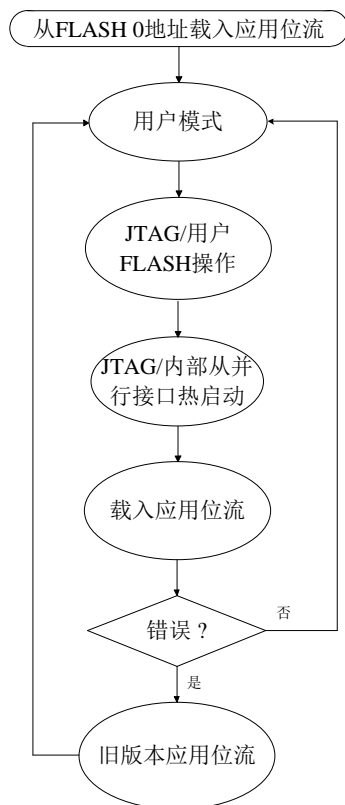


图 3-12 无黄金位流时加载流程

3.7.4 热启动

向器件发送指令前，需先同步，确保器件能正确解析指令。若通过Slave Parallel接口发送指令，还需发送总线位宽检测指令。在多启动应用中，黄金位流包含写IRSTADDR寄存器指令，写入地址为应用位流起始地址，热启动时无需包含写IRSTADDR寄存器操作。若多启动中不包含黄金位流，则需要用户在热启动流程中往IRSTADDR寄存器写入热启动时载入位流的起始地址。若黄金位流中内嵌热启动指令，即在应用位流初始化系统中，不可通过热启动升级到黄金位流。

热启动也可用于清空配置存储器，如解除JTAG临时禁止时使用，此时不需写IRSTADDR寄存器。热启动完整流程如表 3-26所示。

表 3-26 热启动流程

内容	含义
FFFFFFFF	填充字（100 个）
.....	
FFFFFFFF	
000000AA	总线位宽自动检测
08100020	

内容	含义
FFFFFFFF	填充字（10 个）
.....	
FFFFFFFF	
01332D94	同步
AC000001	类型 1 包头：写 IRSTADRR 寄存器（可选）
xxxxxxxx	数据：IRSTADRR 的内容（可选）
A8800001	类型 1 包头：写 CMDR 寄存器
0000000F	数据：IRST 命令
A8800001	类型 1 包头：写 CMDR 寄存器
0000000B	数据：DESYNC 命令
A0000000	无操作类型 1 包头（100 个）
.....	
A0000000	

3.7.5 看门狗

看门狗用于检测配置过程的超时。

配置模式下，看门狗时钟为配置时钟。每个时钟周期，看门狗计数减1。看门狗计数减为0时，产生看门狗超时标志。

若位流设置中使能版本回退（Project Setting->Generate Bitstream->Configuration->Enable Watchdog In Configuration Mode），并设置合适的看门狗数值（Load Watchdog），则在主模式（Master SPI）配置过程中，看门狗超时会引发版本回退。

版本回退过程中和版本回退结束后，看门狗被禁止。版本回退成功结束后，热启动可以取消对看门狗的禁止。

看门狗数值

看门狗在不同的配置模式下都有效，使用不同的配置模式时，看门狗也使用相应的时钟，因此，可通过实际位流大小（ $L_{\text{bitstream}}$ ）以及配置位宽（ W ）得到看门狗数值的估算。

即：

$$\text{Watchdog (Hex)} > L_{\text{bitstream}}/W$$

4 回读操作

Logos2 FPGA支持通过JTAG和Slave Parallel接口回读普通位流和压缩位流。

4.1 JTAG 接口回读

JTAG接口回读操作前需要检测IR中的唤醒结束信号wakeup_over或状态寄存器中的唤醒结束信号wakeup_over，确保唤醒后才能回读。通常唤醒结束很长时间以后才会回读。

4.2 Slave Parallel 接口回读

使用Slave Parallel接口进行回读，需确保该接口被使能或保留。

如在配置模式下使用该接口进行回读，应确保当前生效的配置模式为Slave Parallel模式，即MODE[2:0] = 3'b110。

如在用户模式下使用该接口进行回读，应确保当前生效的位流中设置了保留配置接口（设置位置为PDS中的Project Setting->Generate Bitstream->Readback->Persist Pin，默认不保留），并且是通过Slave Parallel接口进行配置的。并且回读使用的数据位宽必须与配置时的位宽一致，即若使用Slave Parallel x8进行配置，并保留了配置接口，那么可以通过Slave Parallel x8进行回读，而不能以x16/x32位宽进行回读。

Slave Parallel接口回读操作前需要检测状态寄存器中的唤醒结束信号wakeup_over，确保唤醒后才能回读。通常唤醒结束很长时间以后才会回读。

使用Slave Parallel接口进行回读时，需要首先按照Slave Parallel接口时序要求，向器件发送一些指令，见[Slave Parallel接口回读](#)，其中指令流以32bits形式呈现，需根据[不同接口位宽下的位序](#)发送。器件在正确接收到指令后，会在数据准备好后，在时钟上升沿将数据发出，用户即可采样数据。回读配置寄存器和配置存储器流程不同，如下文所述。回读需要满足的时序要求参考第2章的[Slave Parallel配置接口典型时序图](#)。

4.2.1 回读配置存储器

表 4-1 回读配置存储器流程

流程	指令流（Hex）/操作
100 个填充字	FFFFFFFF FFFFFFFF
总线位宽自动检测	000000AA 08100020

流程	指令流 (Hex) /操作
10 个填充字	FFFFFFFF FFFFFFFF
同步字	01332D94
类型 1 包的包头：空操作	A0000000
写 ADDR 寄存器	AAC00001 00000000 ⁽¹⁾
写 RCMEM 命令到 CMDR 寄存器	A8800001 00000006
写类型 1 包的包头，用于读 CMEMOR 寄存器	B1C00000
写类型 2 包的包头，用于读数据	5xxxxxxx ⁽²⁾
暂停操作至少 40 个配置时钟周期 (CS_N 为 1，时钟 CFG_CLK 持续有)	拉高 CS_N，并在 CS_N 为 1 时改变 RWSEL 为读 ⁽³⁾
读 CMEMOR 寄存器	拉低 CS_N，采样数据，读完后拉高 CS_N，改变 RWSEL 为写 ⁽³⁾ 。
写 RCMEMDIS 命令到 CMDR 寄存器	A8800001 00000011
写 NOP 命令到 CMDR 寄存器	A8800001 A0000000
写 DESYNC 命令到 CMDR 寄存器	A8800001 0000000B
100 个无操作类型 1 包头	A0000000 A0000000

注：

- (1) 向 ADDR 寄存器写入回读起始地址。
- (2) 回读的 32bits 字数。
- (3) RWSEL 改变需满足的时序要求见 [Slave Parallel 回读时序](#)。

4.2.2 回读配置寄存器

回读配置寄存器的流程如下所示：

表 4-2 回读配置寄存器流程

流程	指令流 (Hex) /操作
100 个填充字	FFFFFFFF FFFFFFFF
总线位宽自动检测	000000AA 08100020
10 个填充字	FFFFFFFF FFFFFFFF
同步	01332D94
类型 1 包的包头：空操作	A0000000
写 NOP 命令到 CMDR 寄存器	A8800001 A0000000
写类型 1 包的包头，用于读指定寄存器	Bxxxxxxx ⁽¹⁾
等待最少 20 个配置时钟周期	拉高 CS_N，并在 CS_N 为 1 时改变 RWSEL 为读 ⁽²⁾

流程	指令流 (Hex) / 操作
读指定寄存器	拉低 CS_N，采样数据，读完后拉高 CS_N，改变 RWSEL 为写 ⁽²⁾ 。
写 DESYNC 命令到 CMDR 寄存器	A8800001 0000000B
100 个无操作类型 1 包头	A0000000 A0000000

注：

- (1) 按照[包格式](#)，填入需要回读的[配置寄存器](#)地址，以及回读的 32bits 字数。
- (2) RWSEL 改变需满足的时序要求见 [Slave Parallel 回读时序](#)。

5 SEU 检测

Logos2 FPGA支持SEU 1-bit纠错和2-bit检错。ECC算法使用SECDED算法，按帧分组。

SEU检错和纠错可以通过内部从并行接口进行。

在SEU检错纠错过程中，用户逻辑应保存当前回读帧，当出现单比特错误时，用户逻辑将错误比特进行翻转后，动态重配这一帧进行纠错。

当出现双比特错误时，无法检测错误位置，只能通过动态重配正确的位流帧，进行纠错。由于错误可能出现在任意帧，若用户需要双比特纠错，则需要保证正确位流可获取，如保存在SPI FLASH中，在出现双比特错误时，读取正确的位流帧动态重配进行纠错。

Logos2 SEU功能可使用IP进行实现，详情参考文档《UG042001_SEU_IP》。

6 设计保护

6.1 位流加密

用户设计被竞争对手恶意抄袭是 FPGA 使用过程中的常见问题之一。为解决该问题，Logos2 系列 FPGA 中内置了位流加密保护的功能，用于防止用户的设计被恶意克隆。

6.1.1 不带位流加密保护的场景

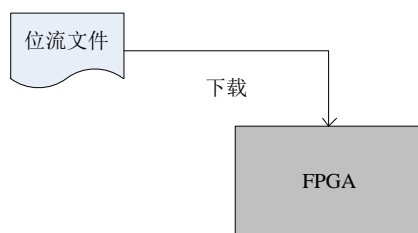


图 6-1 无位流加密场景

如图 6-1 所示，Logos2 系列 FPGA 是基于 SRAM 的 FPGA，每次上电都需要从外部加载位流文件。用户的位流文件通常存储于 FPGA 外部的存储单元，例如 Flash 等。

恶意克隆者可以有各种方法获取这个位流文件。获取位流文件后，恶意克隆者把这个位流文件直接加载于他们自己采购的同款 FPGA，即可实现与用户完全相同的 FPGA 功能。

6.1.2 带位流加密保护的场景

为了有效保护用户的知识产权，Logos2 系列 FPGA 提供了位流加密的功能。

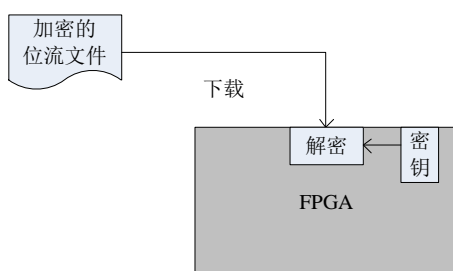


图 6-2 位流加密场景

如图 6-2 所示，位流文件本身是加密的，加密后的位流文件加载到 Logos2 系列 FPGA 时，必须首先结合预先存储在芯片内部的密钥进行解密。这里，加密的位流文件的密钥和储存在 Logos2 FPGA 内部的密钥必须匹配，否则 Logos2 FPGA 无法正常进入工作状态。

Logos2 FPGA 内部的密钥是由用户通过 JTAG 提前写入的，可以写入的位置有 eFuse 和电源供电 RAM (BBRAM)，或用于测试的临时密钥寄存器。eFuse 是一次可编程 (OTP) 的，也是非易失的，所以说该密钥信息一经编程会永久保留。此外，Logos2 FPGA 的密钥提供了读保护功能，使能读保护功能后写入的密钥无法回读，将密钥写入 eFuse 或 BBRAM 时，PDS

下载工具会默认使能密钥读保护。

使用了位流加密保护后，用户不再受恶意抄袭的困扰。恶意抄袭者即使拿到了位流文件（加密的），但是他不知道密钥是什么，所以无法将该位流文件加载到未包含对应密钥的 Logos2 FPGA 中。

6.1.3 加密算法

Logos2 系列 FPGA 采用 AES256-GCM 加密算法对位流进行加密，并且非压缩位流还支持分块加密功能，即将位流分成多块，每一块的加密都使用不同的密钥，并且除了第一块的密钥正常存储在 FPGA 中之外，每一块的密钥都跟前一块一起加密。分块加密提供了更强的安全性，能有效防止旁路攻击，如 DPA 攻击。分块加密分块数可通过 PDS 软件进行设置，越多的分块数能提供更强的安全性，但也会增加位流大小，从而影响配置时间。

AES256-GCM 加密算法还支持自认证。打开自认证时，如果不知道密钥，那么位流不会被加载。

6.1.4 位流加密保护的使用流程

位流加密保护的主要使用流程包括：

- 1) 选定密钥，注意用户需要妥善保护密钥；
- 2) 用户 PDS 生成位流文件时，使用选定的密钥，产生加密的位流文件；
- 3) 用户用 PDS 内置的 Fabric configuration，把密钥烧写到 Logos2 系列产品中。之后，用户可以放心的使用加密的位流文件来加载经过 3) 处理后的 Logos2 FPGA。

生成加密位流文件（sbit）的步骤如下，在 PDS 的 Generate Bitstream 选项点鼠标右键，选择 Configure 项进行设置，如图 6-3 所示。也可以通过菜单中的 Process 进行选择。

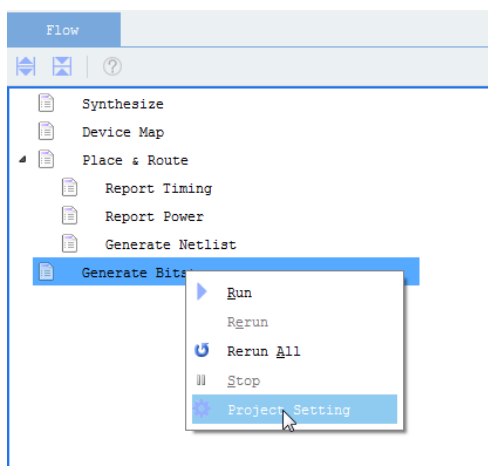


图 6-3 Configure 界面

弹出窗口如图 6-4 所示：

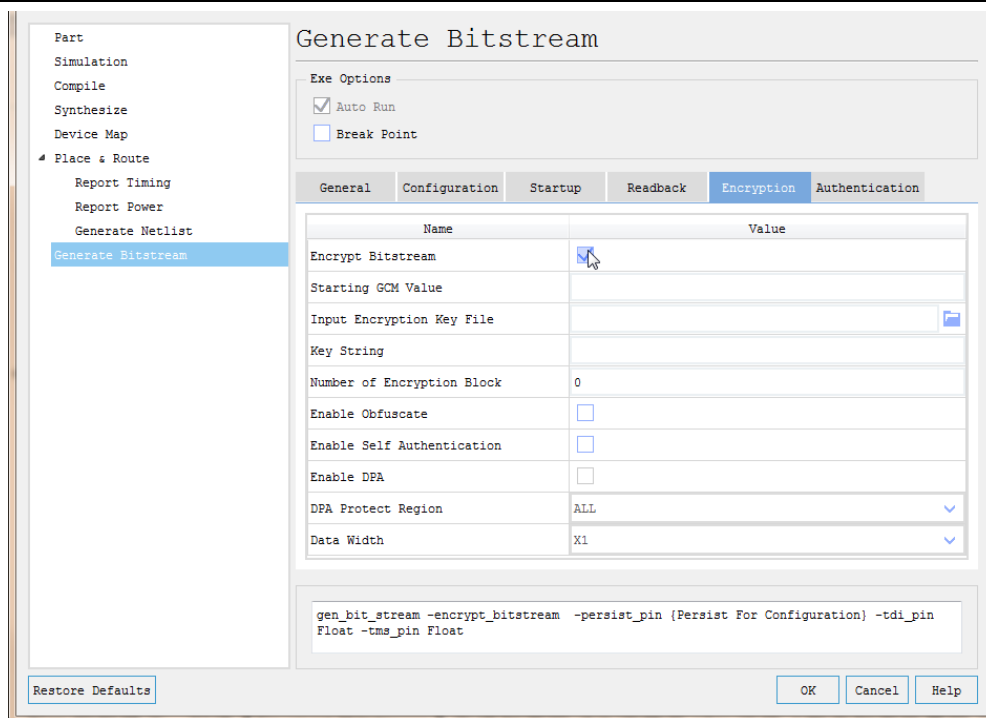


图 6-4 Configure 设置选项

其中，【Encrypt Bitstream】选择是否加密位流文件，在【General】的【Create Bit File】配置为 Yes 的前提下，勾选【Encrypt Bitstream】生成加密的 sbit 文件及 nky 文件（包含加密所需的 96 比特的初始 GCM 值和 256 比特的密钥）。用户可以编辑 GCM 和密钥字符串或选择 nky 文件，若用户不指定，则随机生成 nky 文件。

【Starting GCM Value】用户手动输入初始的 GCM 字符串；

【Input Encryption Key File】选择 nky 文件，如果在此选择了密钥文件，则软件选择以密钥文件进行加密，而不管用户在选项中是否输入了密钥字符串；

【Key String】用户手动输入密钥字符串；

设置完成后，点击 OK 退出配置窗口。再运行 Generate Bitstream 即生成加密的位流文件。

【Enable Obfuscate】由于 AES 算法标准的公开性，为了增强数据流的安全性，在满足业界 AES256 算法标准的基础上，进行一些优化处理。

【Number of Encryption Block】分块加密的块大小，以 128 帧（413696bits）为单位。默认值为 0，即不分块。该数值越大，则分块数越小。

【Enable Self Authentication】AES256-GCM 加密算法提供的自认证使能；

【Data Width】加密算法以 128 比特为单位，当使用不同配置位宽的配置方式进行配置时，需根据实际情况设置这里的数值，以确保器件能正常解密。

6.2 回读保护

默认情况下，Logos2 器件在用户模式下可通过 JTAG 接口进行重配和回读，若用户设计中例化了内部从并行接口，或者未使用内部从并行接口，并且保留 Slave Parallel 接口时，则可通过内部从并行接口或 Slave Parallel 接口进行重配或回读。为了避免配置存储器被非法访问，用户可设置不保留外部配置接口，并且不将内部从并行接口连接到器件管脚上。另一方面，Logos2 器件还提供回读保护设置。开启回读保护后，只有通过复位器件，才能再次使能回读。

回读禁止设置见图 6-5。

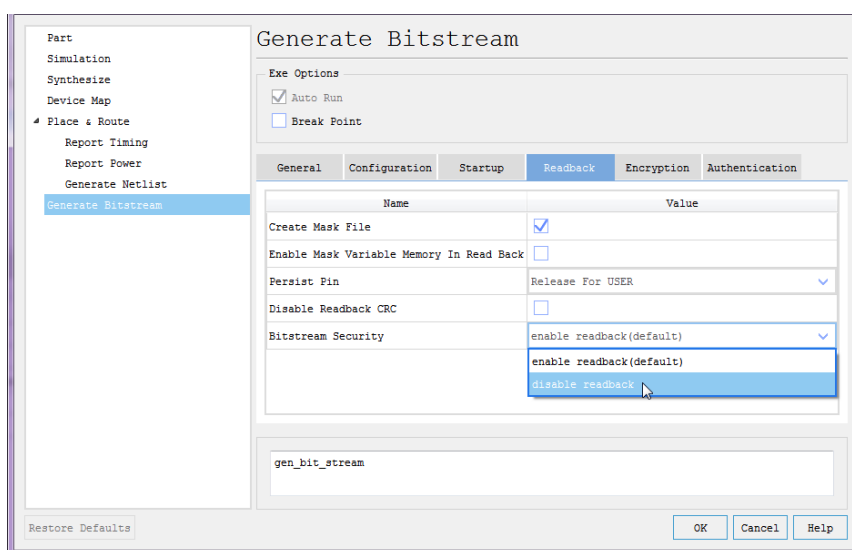


图 6-5 回读禁止设置

6.3 位流认证

除了 AES256-GCM 加密算法提供的自认证，Logo2 还支持 RSA 认证，采用认证算法 RSA-2048。

RSA-2048 公钥摘要算法为 SHA3-384。RSA-2048 公钥摘要长度为 384 位。

位流摘要算法为 SHAKE256。位流摘要长度为 2047 位。

认证位流支持普通位流、加密位流和分块加密位流。

认证从写完认证寄存器后开始，到去同步结束。

认证位流进行认证的同时，净荷存入配置存储器。认证通过后，才能够进行后续的唤醒操作。认证失败后，产生认证失败标识，同时将 INIT_FLAG_N 置 0。

认证加密位流和认证分块加密位流进行认证的同时进行解密，净荷存入配置存储器。认证通过后，才能够进行后续的唤醒操作。认证失败后，产生认证失败标识，同时将

INIT_FLAG_N 置 0。

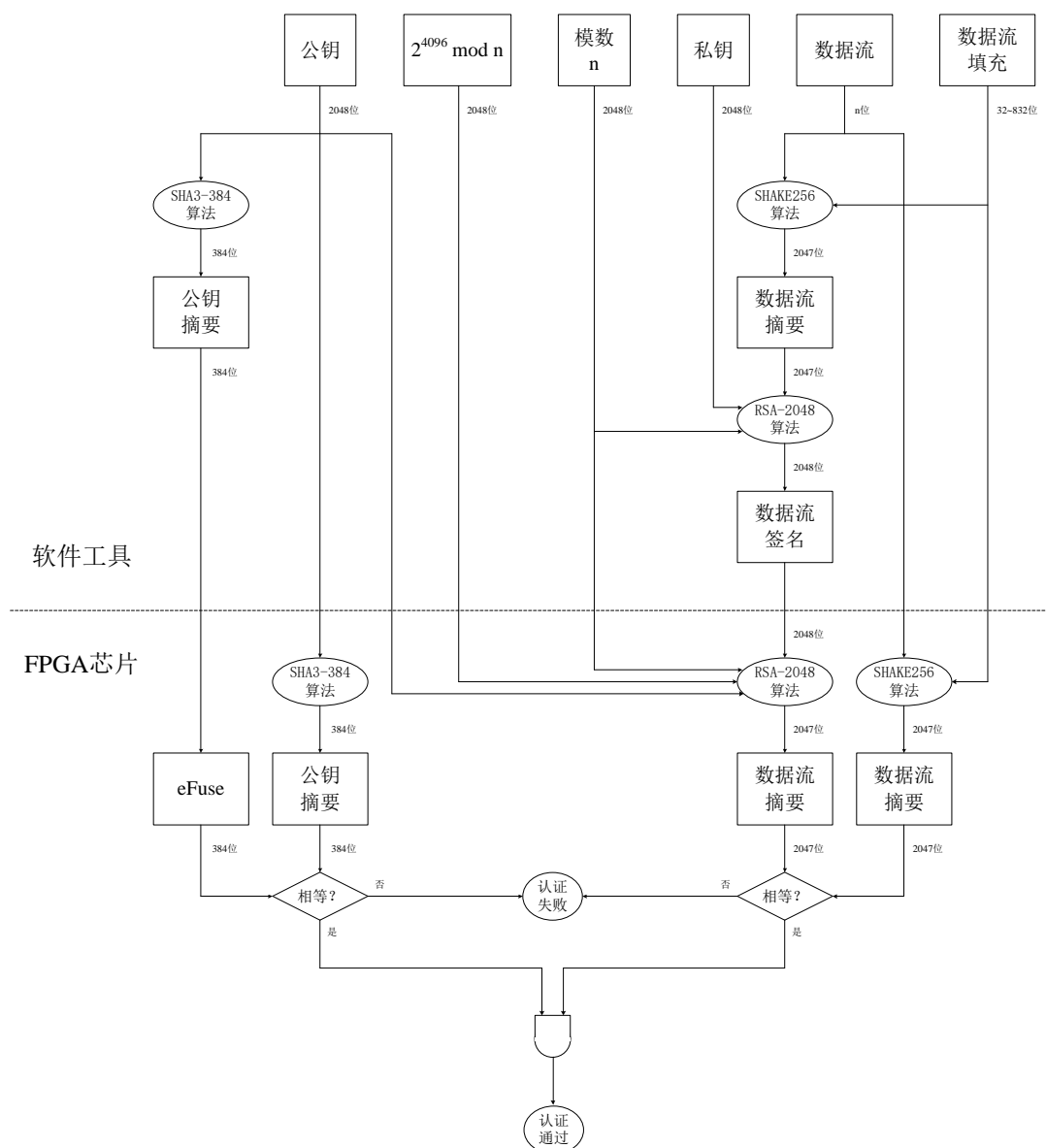


图 6-6 认证过程示意图

6.4 DPA 保护

Logos2 支持的认证方式有 AES256-GCM 的自认证，以及 RSA 认证。使用这两种认证加密位流会在认证的同时进行解密，为了进一步防止 DPA 攻击，一种有效的方式是认证的时候先将位流写入配置存储器，不进行解密，认证通过之后才将位流解密并配置。

Logos2 提供 DPA 保护认证支持。当 DPA 保护使能时，DPA 保护认证加密位流进行认证的同时，将加密净荷写入配置存储器。认证通过后，将加密净荷逐帧回读出，解密后写回配置存储器，然后进行唤醒操作。认证失败后，产生认证失败标识，同时将 INIT_FLAG_N 置 0。

6.5 UID

可重构器件行业存在很大的一个利益安全危害即是设计的抄袭和非法阔产。那么为了避免此类非法行为，UID 即应运而生。每一个器件设备都有一个与之对应的唯一编号，该编号在器件出厂的时候已经唯一确定。用户可以通过 UID 接口（例化 GTP_UDID）和 JTAG 接口读取，并且以自己特有的加密算法处理后将得到的结果并入编程位流。每一次重载位流后，设备进入用户模式，用户逻辑都会先读取该 UID 以用户独特的加密算法处理后与之前编程位流中的结果相比对，若有不同。则设备无法正常工作。

96 位 UID 存储在 eFuse 中，在器件出厂时统一编程。每次设备上电，自动将 eFuse 中的 UID 读取到寄存器中，以便用户随时取出。

6.6 JTAG 安全管理

JTAG 接口为用户提供了方便且全面的芯片配置、回读、测试的方法。在用户希望芯片处于稳定工作状态不受恶意干扰时，可将 JTAG 接口禁止。

Logos2 提供 JTAG 接口安全管理支持。用户可以通过在用户逻辑中例化用户 JTAG 接口（对应 GTP_JTAGIF）对 JTAG 临时禁止，临时禁止可以通过对芯片重新上电或其他配置接口重新打开。或者通过写 eFuse 中的 JTAG 禁止位，对 JTAG 进行永久禁止。

JTAG 临时禁止后，可通过其他接口重新打开。若用户逻辑中例化了 GTP_IPAL_E2，则用户可通过内部从并行接口发送热启动指令，使芯片的配置存储器复位，重新打开外部 JTAG 接口。若用户逻辑未例化 GTP_IPAL_E2，并且保留了外部 Slave Serial、Slave Parallel 接口，可通过外部接口发送热启动指令，来打开外部 JTAG。

6.7 eFuse

eFuse 是一次性可编程非易失存储器，在 Logos2 系列器件中用于存储一些器件设置信息、UID 以及其他用户可编程位：

- 1、256 位 AES-GCM 加密密钥
- 2、384 位公钥摘要
- 3、32 位用户 Fuse

安全设置（认证使能，DPA 保护使能，密钥锁定标志，公钥摘要锁定标志，用户 Fuse 锁定标志）。

未编程的 Fuse 值为 0，编程后的 Fuse 值为 1。所有用户可编程位均可通过 PDS 进行配置，

具体操作见文档《Fabric Configuration 用户手册》。

由于 JTAG 接口可用于编程 eFuse，用户应避免 JTAG 接口受热插拔等操作影响而出现干扰信号，导致 eFuse 被误写入。若 JTAG 接口无法避免干扰，可在确定芯片应用场景后，锁定 eFuse 所有功能位。在 Fabric Configuration 工具中使用 TCL 指令“`cfg_efuse_lock -lock FF -password 000000`”完成 eFuse 所有功能位锁定。

96 位 UID 在器件出厂时统一编程，可由 JTAG 接口或 UID 接口读出，见 [UID](#)。32 位用户 Fuse 可通过 JTAG 接口或用户 Fuse 接口进行读取，见[用户 Fuse 接口](#)。

7 用户逻辑接口

7.1 内部从并行接口

GTP_IPAL_E2 提供内部从并行接口，以及回读 CRC、SEU 接口。用户可以通过例化该 GTP 进行内部从并行接口操作，也可以使用回读 CRC、SEU 接口很方便地进行回读 CRC 和 SEU 操作。这里主要对内部从并行接口进行介绍，SEU 功能可使用 IP 进行实现，详情参考文档《UG042001_SEU_IP》。

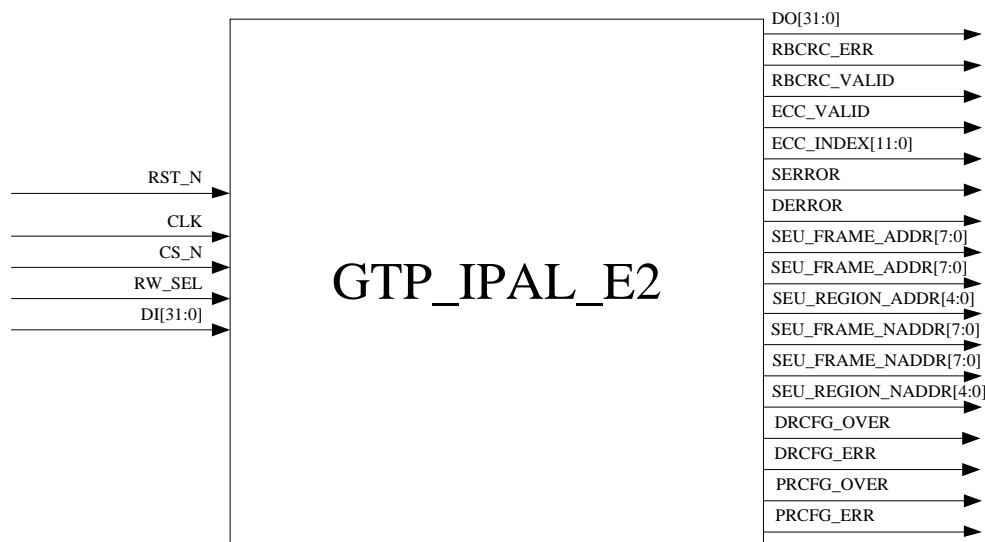


图 7-1 GTP_IPAL_E2 端口图

7.1.1 端口列表

表 7-1 GTP_IPAL_E2 端口列表

端口名称	I/O	位宽	描述
CLK	I	1	内部从并行接口时钟
RST_N	I	1	复位信号，低电平有效，仅用于仿真 ⁽¹⁾
内部从并行接口			
CS_N	I	1	片选信号，低电平有效
RW_SEL	I	1	读写选择：0 写，1 读；
DI	I	32	数据输入
DO	O	32	数据输出
回读 CRC 接口			
RBCRC_ERR	O	1	回读 CRC 错误标志，高有效。
RBCRC_VALID	O	1	回读 CRC 有效标志，高有效，持续 1 个时钟周期。
SEU 检测接口			
ECC_VALID	O	1	ECC 有效标志，高有效。
ECC_INDEX	O	12	单比特错误的地址索引

端口名称	I/O	位宽	描述
SERROR	O	1	SEU 检测的单比特错误标志
DERROR	O	1	SEU 检测的双比特错误标志
SEU_FRAME_ADDR	O	8	SEU 检测的当前帧地址
SEU_COLUMN_ADDR	O	8	SEU 检测的当前列地址
SEU_REGION_ADDR	O	5	SEU 检测的当前区地址
SEU_FRAME_NADDR	O	8	SEU 检测的下一帧地址
SEU_COLUMN_NADDR	O	8	SEU 检测的下一列地址
SEU_REGION_NADDR	O	5	SEU 检测的下一区地址
PRCFG_OVER	O	1	局部重配结束标志，为 1 时指示局部重配完成。
PRCFG_ERR	O	1	局部重配错误标志，在 PRCFG_OVER 有效时指示局部重配结果。
DRCFG_OVER	O	1	动态重配结束标志，为 1 时指示动态重配完成。
DRCFG_ERR	O	1	动态重配错误标志，在 DRCFG_OVER 有效时指示动态重配结果。

注：（1）该信号仅用于仿真时使接口恢复初始状态，实际硬件上无此端口，对该端口操作即无必要也不会起任何作用。

7.1.2 参数定义

表 7-2 GTP_IPAL_E2 参数列表

参数名称	参数类型	有效值	功能描述
IDCODE	Binary	0~32'hffffff	器件 IDCODE，仅用于仿真
DATA_WIDTH	String	"X8"、"X16"、"X32"	并行数据位宽

7.1.3 接口时序

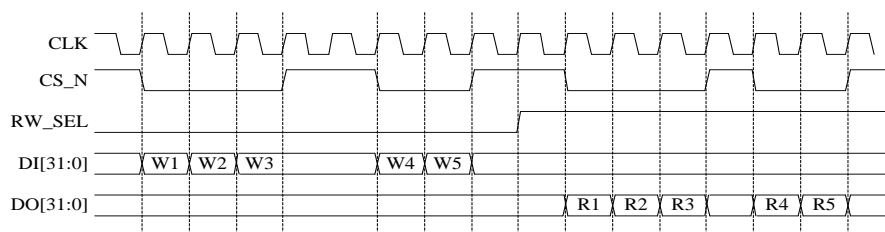


图 7-2 GTP_IPAL_E2 内部从并行接口读写接口时序

注:RW_SEL 信号跳变只能在 CS_N 为高时进行。

7.1.4 例化模板

```
GTP_IPAL_E2 #(
    .DATA_WIDTH("X8"),
    .IDCODE('b101010101010100101010101010101)
) <InstanceName> (
    .DO(),
```

```
.ECC_INDEX(),
.DI(),
.DERROR(),
.ECC_VALID(),
.RBCRC_ERR(),
.RBCRC_VALID(),
.SERROR(),
.CLK(),
.CS_N(),
.RST_N(),
.RW_SEL(),
.SEU_FRAME_ADDR(),
.SEU_COLUMN_ADDR(),
.SEU_REGION_ADDR(),
.SEU_FRAME_NADDR(),
.SEU_COLUMN_NADDR(),
.SEU_REGION_NADDR(),
.PRCFG_OVER(),
.PRCFG_ERR(),
.DRCFG_OVER(),
.DRCFG_ERR()
);
```

7.2 UID 接口

7.2.1 端口列表

表 7-3 GTP_UDID 端口列表

端口名称	I/O	位宽	功能描述
DI	I	1	串行数据输入
DO	O	1	串行数据输出
SE	I	1	数据移位使能
LOAD	I	1	数据寄存器并行加载 UID CODE
CLK	I	1	时钟

7.2.2 参数列表

表 7-4 GTP UDID 参数列表

[illegible]

注：（1）不同系列的 UID 长度不同，为了保持仿真与上板一致，用于 Logos2 器件时应改为 96。

7.2.3 接口时序

UID 长度为 96 位。

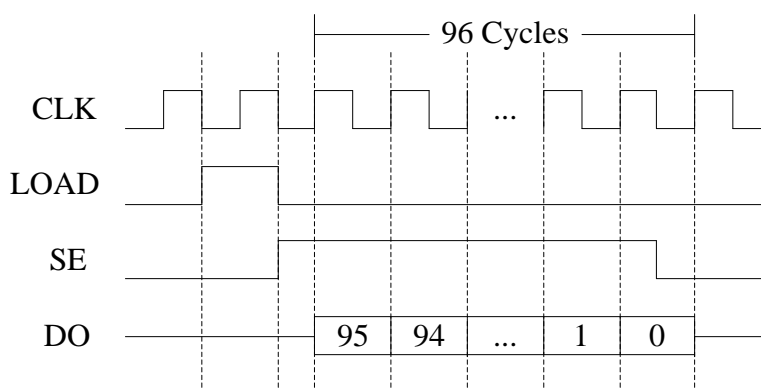


图 7-3 读 UID 时序

用户可以扩展 UID 的位宽。

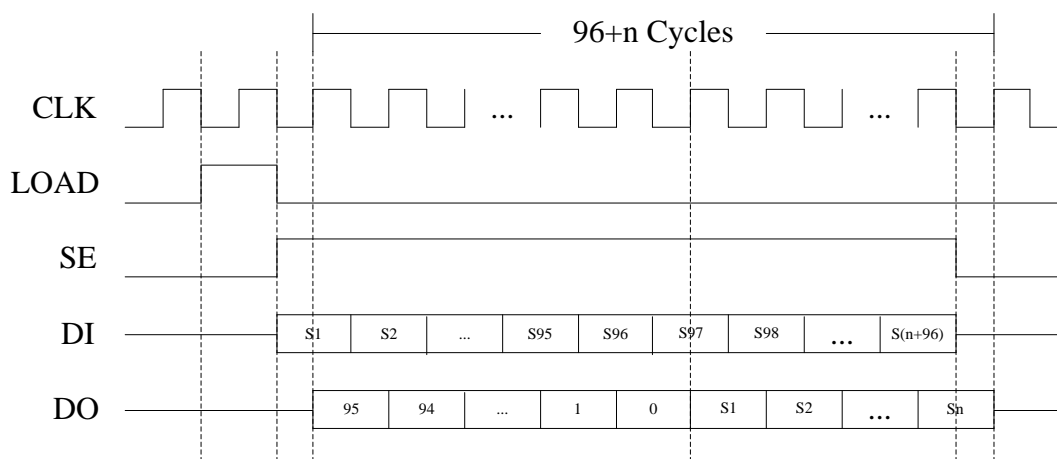


图 7-4 扩展 UID 位宽

7.2.4 例化模板

GTP UDID 的 Verilog 例化模板如下所示。

GTP UDID #(

.UID WIDTH(96)

[illegible]

```

000000)
) <InstanceName> (
    .DO(),
    .CLK(),
    .DI(),
    .LOAD(),
    .SE()
);

```

7.3 用户 JTAG 接口

当用户逻辑中例化用户 JTAG 接口时，用户逻辑完全接管外部 JTAG 接口。接口时序、操作方法与外部 JTAG 一致。

7.3.1 端口列表

表 7-5 GTP_JTAGIF 端口列表

端口名称	I/O	位宽	功能描述
TCK	I	1	用户 JTAG 时钟，最高频率为 50M
TMS	I	1	用户 JTAG 测试模式选择。时钟 TCK 上升沿时，控制 TAP 控制器的状态跳转
TDI	I	1	用户 JTAG 测试数据输入。时钟 TCK 上升沿时，做为 JTAG 指令和数据寄存器的串行输入
TDO	O	1	用户 JTAG 测试数据输出。时钟 TCK 下降沿时，做为 JTAG 指令和数据的串行输出

7.3.2 参数列表

表 7-6 GTP_JTAGIF 参数列表

参数名称	参数类型	有效值	功能描述
USERCODE	Binary	32'h0~32'hFFFF_FFFF	USERCODE

7.3.3 例化模板

GTP_JTAGIF 的 Verilog 例化模板如下所示。

```

GTP_JTAGIF #(
    .USERCODE(b11111111111111111111111111111111)
) <InstanceName> (
    .TDO(),

```

```

        .TCK(),

        .TDI(),

        .TMS()

    );
    
```

7.4 用户 Fuse 接口

用户逻辑可通过例化用户 Fuse 接口（GTP_EFUSECODE）对用户 Fuse 进行读取，该接口以异步并行方式输出 32 位用户 Fuse 值。

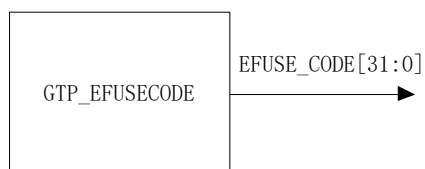


图 7-5 GTP_EFUSECODE 端口图

7.4.1 端口列表

表 7-7 GTP_EFUSECODE 端口列表

端口名称	I/O	位宽	功能描述
EFUSE_CODE	O	32	Efusecode 输出数据总线

7.4.2 参数列表

表 7-8 GTP_EFUSECODE 参数列表

参数名称	参数类型	有效值	功能描述
SIM_EFUSE_VALUE	Binary	32'h0~32'hFFFF_FFFF	EFUSE_CODE 仿真值

7.4.3 例化模板

GTP_EFUSECODE 的 Verilog 例化模板如下所示。

```

GTP_EFUSECODE #(
    .SIM_EFUSE_VALUE('b00010010001101000101011001111000)
) <InstanceName> (
    .EFUSE_CODE()
);
    
```

免责声明

版权声明

本文档版权归深圳市紫光同创电子有限公司所有，并保留一切权利。未经书面许可，任何公司和个人不得将此文档中的任何部分公开、转载或以其他方式披露、散发给第三方。否则，公司必将追究其法律责任。

免责声明

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

2、本文档按现状提供，不负任何担保责任，包括对适销性、适用于特定用途或非侵权性的任何担保，和任何提案、规格或样品在他处提到的任何担保。本文档在此未以禁止反言或其他方式授予任何知识产权使用许可，不管是明示许可还是暗示许可。

3、公司保留任何时候在不事先声明的情况下对公司系列产品相关文档的修改权利。