

一种 SRIO 交换器内部网络设计

胡孔阳, 韩琼磊, 顾大晔

(中国电子科技集团公司 第三十八研究所, 安徽 合肥 230088)

摘 要: 针对 SRIO 交换器数据通路的特点, 提出了一种交换器内部的互连网络设计. 重点解决了实现交换网络的三个问题: 1. SRIO 端点与端点的跨时钟域转换; 2. SRIO 传输层包的组合与拆分; 3. 多个源端点向同一个目的端点请求时的仲裁. 使用 VerilogHDL 对设计进行了描述, 同时设计了 SRIO 端点模型并与之互连, 对交叉网络进行随机测试, 通过自动化比对收、发端点的数据包验证设计的正确性. 经过验证, 本设计可以利用 SRIO 端点控制器 IP 基于 ASIC 或 FPGA 构建 SRIO 交换器.

关键词: SRIO 交换器; 交换网络; 传输层包; 跨时钟域; 仲裁

中图分类号: TP302

文献标识码: A

文章编号: 1000—7180(2018)09—0010—04

ASRIO Switch Inside Network Design

HU Kong-yang, HAN Qiong-lei, GU Da-ye

(No. 38 Research Institute, China Electronics Technology Group Corporation, Hefei 230088, China)

Abstract: According to the characteristics of SRIO switch data path, we proposed one design to build switch inside network. Focus on solving three problems to make switch network; 1. Clock domain crossing between SRIO endpoint to endpoint; 2. SRIO transmission line's package merging and splitting; 3. Arbitration between multi-source endpoint to the same endpoint. We used VerilogHDL to describe this design, and also design the SRIO endpoint model to connect with the network, to make the random test, we check the input and output package to verify the correctness automatic. The verification proves that this design can use SRIO end point controller in ASIC or FPGA to build SRIO switch.

Key words: coupling transitions; low power encoding; data bus; system on chip

1 引言

SRIO 是由 Motorola 等公司率先倡导的一种高性能、低引脚数、基于数据^[1]包交换的互连体系结构. 从 2001 年 3 月最初的 1.1 规范延续至今, 已持续更新至 4.0 规范, 速率亦由 1.25Gbps/lane 提升至 25.78125Gbps/lane. 其高效的传输方式和易扩展的互连结构可以成功的应用于芯片到芯片、板间和背板互连, 满足多种场景下的通讯需求.

SRIO 协议由逻辑层、传输层和物理层组成. 逻辑层定义了数据包的格式; 传输层在逻辑层包之上进一步添加了设备 ID 信息, 为 SRIO 包在系统中传

输、路由提供源和目的地址; 物理层则定义了设备之间接口的电气特性、流量控制以及错误管理. 依据 SRIO 协议定义, 端点与端点之间进行非直连传输时, 需要增加 SRIO 交换器(Switch), 由交换器负责包的转发和路由, 换句话说, 交换器是组成 SRIO 复杂拓扑结构的重要部分^[2-3].

目前国内外 IP 市场上, SRIO 端点 IP 可以从 IDT、Praesum、Mobiveil 几家厂商采购, 而交换器 IP 则几乎没有可直接利用的资源. 本文在分析 Mobiveil 端点 IP 传输层信号的基础上, 提出一种互连结构设计, 基于该设计可以实现由多个端点 IP 构建 SRIO 交换器^[4-5].

2 SRIO 交换器结构

SRIO 协议没有定义交换器的具体结构,分析多家交换器产品的用户手册,一般的结构均如图 1 所示。交换器内部一般包括多个端点设备、路由表和一个复杂的交换网络。

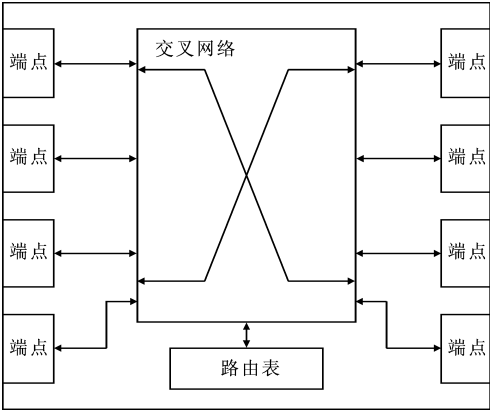


图 1 SRIO 交换器典型结构

端点设备负责与交换器相连的 SRIO 设备对接,因此具备物理层与传输层的功能;路由表记录 SRIO 网络中各个节点的设备 ID 信息,查询路由表可以知道传输层包转发给哪个端点,路由表可以由 SRIO 网络中的节点进行动态的配制,也可以由交换器内部的慢速外设做静态配制;交叉网络负责端点与端点之间包的传输,由于端点个数一般较多,端点至端点的数据通路与端点的个数成平方关系,因此交叉网络的设计尤为重要。

交叉网络作为本文所研究的重点,一般要解决如下三个问题:1、SRIO 端点与端点的跨时钟域转换;2、SRIO 传输层包的组合与拆分;3、多个源端点向同一个目的端点请求时的仲裁电路设计。

问题 1 描述:每个端点均包含完整的物理层与传输层,物理层由 Serdes PHY 实现。Serdes PHY 除了实现数据的并串转换外,另一个重要功能是由内部的 PLL 向与之相连的上层控制器提供收、发数据时钟。而每个端点所使用的 Serdes PHY 是独立的,这样会导致端点之端点所使用的时钟完全是异步的,甚至会出现一个端点时钟工作在 1.25 GHz,另一个工作在 25.781 25 GHz 的情况。

解决问题 1 的基本方案是使用异步 FIFO 进行数据的跨时钟域处理。

问题 2 描述:在使用异步 FIFO 进行跨时钟域处理时会出现另一个问题,端点传输层数据包已被解析成包头、数据负载、优先级等信号,并且以连续

的多个时钟周期的形式出现,如图 2 所示。而异步 FIFO 读写均发生在一个时钟周期之内,这就要求设计一个逻辑实现将端点输出的信号进行组合,将组合后的包写入异步 FIFO;在 FIFO 的外一端读出组合后的数据包,再将其按原来写入的格式进行拆分,后发送给接收端点。

这里隐藏着一个将时间信息(连续有效的数据)转化成空间(FIFO 内部的存储单元)的原理。

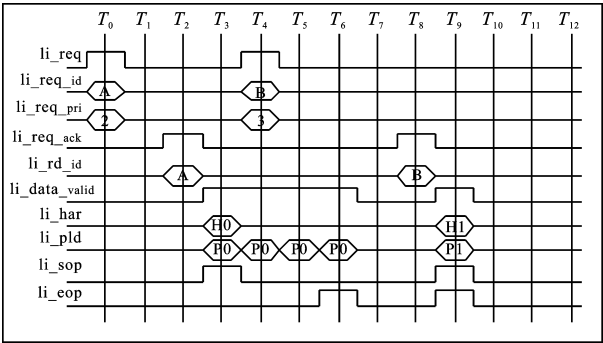


图 2 端点传输层发送信号

问题 3 描述:由于每个端点在交换器内部都是对等的,所以两两之间都存在数据通路;另一个观点是,端点接收到外部的请求是随机的,所以极可能出现多个端点向同一个端点进行请求的现象。

根据 SRIO 协议 QOS 相关内容,在出现请求竞争时,基于下面原则处理:高优先级比低优先级优先获得响应;相同优先级之间进行 Round-Robin 仲裁。

3 端点传输层信号时序

在介绍交叉网络设计之前,有必要分析一下端点传输层信号的时序。端点传输层信号按 SRIO 全双工工作方式分为收、发两组,两组信号都需要进行握手处理。

分析图 2 发送端信号。T₀ 时刻发送端点发出有效请求(li_req),同时还有请求 ID(li_req_id)和优先级(li_req_pri);T₂ 时刻交叉网络响应 T₀ 的请求;T₃ 时刻发送端点开始发送包头(li_hdr)和第一个负载(li_pld),同时包头标志(li_sop)有效;T₄、T₅、T₆ 分别输出后续的负载,并且在 T₆ 时刻输出包结束标志(li_eop);T₄ 时刻端点发出另一个请求;T₈ 时刻交叉网络开始响应 T₄ 时刻的请求;紧接着 T₉ 时刻开始传输第二个包,该包只有一个有效负载。

分析图 3 接收端信号。T₀ 时刻接收端点缓存指示可以接收优先级为(0、1、2、3)的有效请求(lo_pri_

status 等于 4'b1111); T_1 时刻由交叉网络发起图 2 中第一次包传输,共需要四个时钟周期完成,同时 T_1 时刻接收端点缓存指示优先级为 2 的缓存空间已满(第一次包传输优先级为 2);类似的, T_6 时刻交叉网络发起第二次包传输,该次传输只需要一个时钟周期.

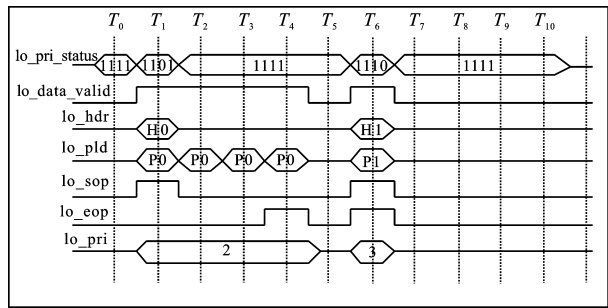


图 3 端点传输层接收信号

4 交叉网络设计

根据上文提出的三个问题结合端点传输层收、发信号时序,设计交叉网络结点结构框图如图 4 所示.

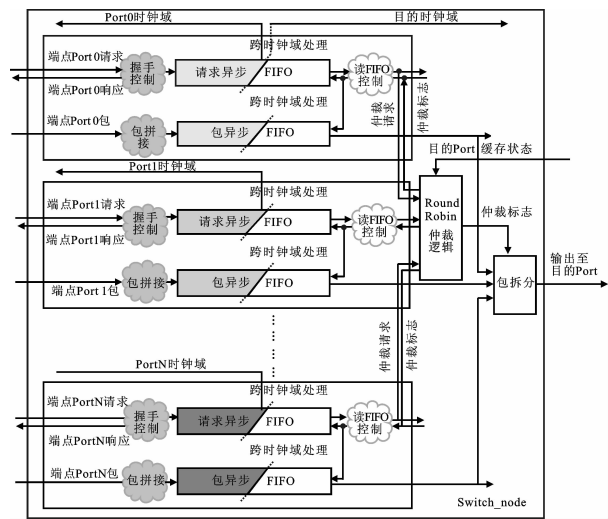


图 4 交叉网络结点结构框图

图 4 中深色部分为不同发送端点的时钟域逻辑,白色部分为接收端点时钟域逻辑,跨时钟域处理集中在请求异步 FIFO 和包异步 FIFO 之中.关于异步 FIFO 的原理在这里不做进一步研究.

在请求异步 FIFO 写端口之前设计控制逻辑,用于控制 FIFO 的写入时机和与请求端点握手;包异步 FIFO 写端口之前设计包拼接逻辑,主要结构包括一个计数器,该计数器在包头出现后开始自增记数,在包结束标志出现后停止记数,并作为包的

一部分一同写入异步 FIFO.此过程如图 5 所示,ply_num 即为计数器,write_fifo_data 为写入 FIFO 的数据,write_fifo 为 FIFO 写信号.

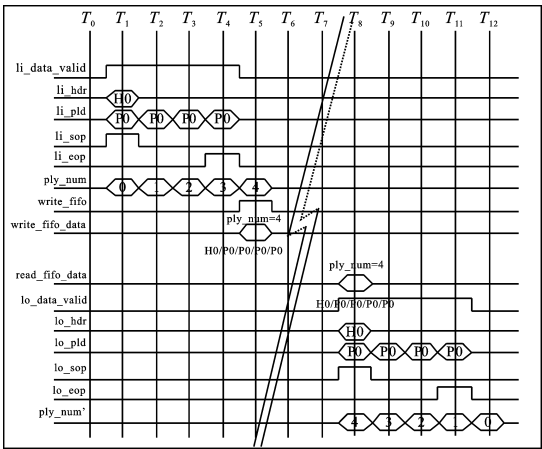


图 5 计数器记录时间信息

在异步 FIFO 读端口处设计控制逻辑,根据包拆分模块输出的完成标志和仲裁逻辑输出的仲裁结果控制 FIFO 的读动作.

包拆分模块将获得仲裁权的数据包重新拆分成包头、负载的形式,分多周期发送至接收端口,这里将利用数据包中的 ply_num 进行自减,当 ply_num 减为 0 时表示包发送结束.

Round-Robin 仲裁逻辑结构如图 6 所示.主要由优先级索引寄存器、优先级索引选择、独热码生成、优先级索引选回和排序逻辑组成.

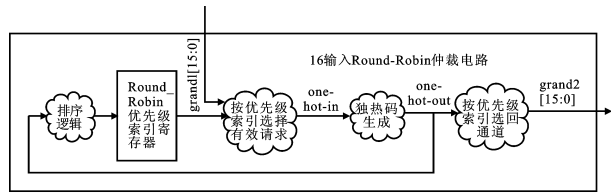


图 6 Round-Robin 仲裁逻辑

优先级索引寄存器用以存储输入请求的优先级信息.当优先级发生变化时,重新排列各个寄存器对应的索引号值;优先级索引选择是按索引寄存器的值将输入的有效请求按索引号重新排列,并输出至独热码生成模块;独热码生成模块用于选择输出优先级最高的输入请求;优先级索引选回逻辑将独热码生成模块输出值按优先级索引寄存器的内容选回至对应的输入请求.

以上介绍的交叉网络结点是基于每个端点前安放的逻辑,在每个端点前例化一个交叉网络结点就可以构建一个完整的交叉网络,如图 7 所示.

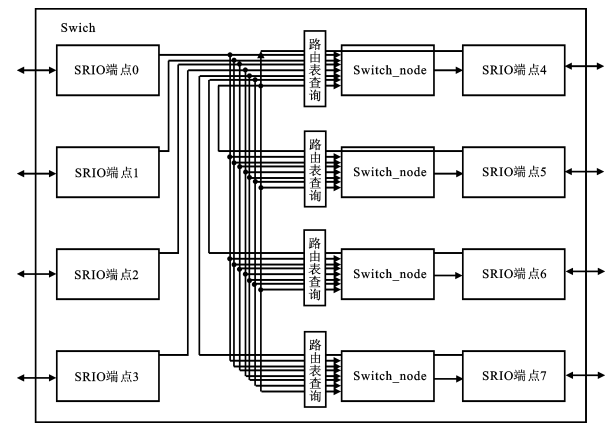


图 7 使用交叉结点构建交叉网络

图 7 画出了端点 4、5、6、7 前的交叉结点,出于画面清晰的考虑,端点 0、1、2、3 前并没有画出,端点 0、1、2、3 也是类似的。

5 验证平台介绍

根据 SRIO 端点传输层的信号时序设计出端点的行为模型并与交叉结点输入和输出接口互连,发送端口的行为模型会自动产生随机长度、随机优先级的数据包,接收端点的行为模型会随机产生缓存的状态标志,记录发送和接口两端的数据包,并在仿真完成后进行数据的自动比对,可以充分模拟 SRIO 交换器的行为,保证设计的正确性,如图 8 所示。

6 结束语

本文基于对 SRIO 端点控制器 IP 传输层行为和 SRIO 交换器数据通路的特点,提出了一种交换器内部的交换网络设计。设计的基本思想是端点与端点之间的点对点传输,通过实例化多个交叉结点可以构建任意端口数目的 SRIO 交换器件。该结构经过改造也可以适用于其它高速串行协议标准的交

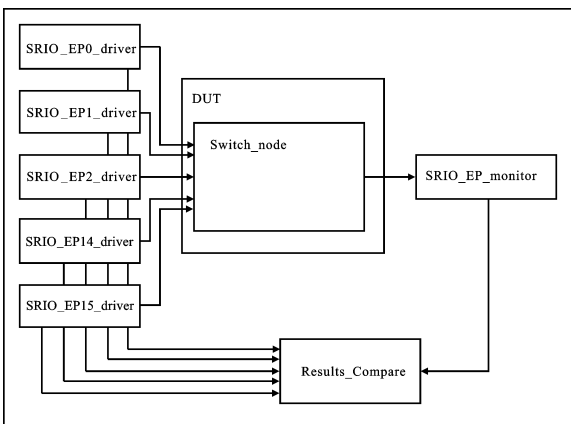


图 8 验证平台示意

换器设备,例如 PCIe 和 10GKR 等。

参考文献:

[1] 谢丽斌,刘凯,尚煜. 串行 RapidIO 交换技术[J]. 电子科技,2012,25(9):129-135.

[2] 翟彦彬,蒋志焱,张保宁. 大规模 RapodIO 协议交换的 FPGA 实现[J]. 现代雷达,2011,33(12):33-35.

[3] 高毅,刘永强,梁小虎. 基于串行 RapidIO 协议的包交换模块的设计与实现[J]. 航空计算技术,2010,40(3):123-126.

[4] 石海洋. 一种 RapidIO 交换网络配置方法的设计与实现[J]. 航空计算技术,2012,42(2):132-134.

[5] 左颜,柴小丽,顾燕飞. 万兆以太网与 RapidIO 网络的互联与传输[J]. 重庆理工大学学报(自然科学),2017,8(2):145-147.

作者简介:

胡孔阳 男,(1986-),硕士,工程师. 研究方向为数字集成电路设计与验证. E-mail:hukongy123@163.com.

韩琼磊 男,(1983-),硕士,工程师. 研究方向为数字集成电路设计与验证.

顾大晔 男,(1987-),硕士,工程师. 研究方向为数字集成电路设计与验证.

(上接第 9 页)

[9] 吕天阳,谢文艳,郑纬民,等. 加权复杂网络社团的评价指标及其发现算法分析[J]. 物理学报,2012,61(21):210511:1-10.

[10] 贾珺,胡晓峰,贺筱媛. 基于节点动态连接度的网络社团划分算法[J]. 复杂系统与复杂性科学,2016,13(4):56-61.

作者简介:

秦怀斌 男,(1980-),硕士,副教授. 研究方向为软件工程、复杂网络等. E-mail:qhb_inf@shzu.edu.cn.

郑瑶 女,(1976-),硕士,副教授. 研究方向为软件工程.

郭理 男,(1968-),硕士,副教授. 研究方向为软件工程、人工智能等.