

一种位宽可配置的对角交叉校验码算法

卜佑军,汪斌强,郭 虹,扈红超

(国家数字交换系统工程技术研究中心,河南 郑州 450002)

摘 要: 局部总线标准 SPI-4.2 已经成为在物理层设备和链路层设备之间进行分组或信元传输的最通用的接口标准.详细介绍了 SPI-4.2 标准中对角交叉校验的工作原理、设计思想和电路结构,并详细讨论了一种可扩展为 $16X$ ($X \geq 1$) 数据宽度的适合硬件 FPGA 实现的 DIP-4 码算法,该算法在实际应用中功能正确且实现简单方便.

关键词: 对角交叉校验;SPI-4.2;可配置;FPGA

中图分类号: TP393

文献标识码: A

文章编号: 1000-7180(2010)08-0112-04

A Configurable Bit Width Algorithm of Diagonal Interleaved Parity Code

BU You-jun, WANG Bin-qiang, GUO Hong, HU Hong-chao

(National Digital Switching System Engineering & Technology Researching Center, Zhengzhou 450002, China)

Abstract: The local high speed bus standard SPI-4.2 has already been the universal interface standard for signal transmission between the PHY device and the LINK device. The paper describes the working principle, designing idea and circuit structure of the diagonal interleaved parity code in the SPI-4.2. And it presents a algorithm of DIP-4 which can be configured to be $16X$ bit width. The algorithm has been proved to be right in practice and implemented easily.

Key words: diagonal interleaved parity; SPI-4.2; reconfigurable; FPGA

1 引言

随着 Internet 宽带化、高速化的发展,10Gb/s 速率的接口已经成为骨干网设备的常见配置. SPI-4.2(System Packet Interface)是光互联网论坛 OIF(Optical Internetworking Forum)定义的局部高速总线标准^[1],支持多种物理层协议,用于 PHY 层芯片到链路层芯片的 10Gb/s 信号传输.主要应用有 OC-192 ATM、Packet over SONET/SDH(POS)、10Gb/s 以太网等,已经迅速成为在物理层设备和链路层设备之间进行分组或信元传输的最通用的接口标准.与其他的 10Gbit/sec 接口相比, SPI-4.2 接口协议更易于工程实现,减少了管脚数量和功耗,增强了信号完整性,被业界广泛采用^[2].在 SPI-4.2 接口协议

中,规定数据传输时采用对角交叉校验(Diagonal Interleaved Parity, DIP-4),以便及时发现高速传输中的错误,增强高速数据传输的可靠性.

2 SPI-4.2 接口标准简介

图 1 为 SPI-4.2 接口参考模型,它采用 16 位的基于 LVDS I/O 的数据总线,数据总线的数据速率可高达 840Mb/s,每个方向的累计带宽可达 13.44Gb/s,采用双沿时钟采样.提供带内控制字(control word)和用于流量控制的带外状态通道.带内控制字指示各个通道的地址、分组的起始和结束,以及校验和;状态通道采用 2 位状态总线(RSTAT、TSTAT),提供接收方的通道状态信息,发送方根据这些状态信息对各通道的数据进行轮询调度.

收稿日期: 2010-05-25; 修回日期: 2010-06-25

基金项目: “八六三”国家重点基金项目(2008AA01Z214, 2007AA01Z2a1, 2007AA01Z212); 河南省基础与前沿技术研究计划项目(072300410010)

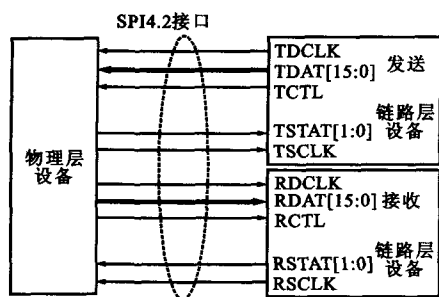


图 1 SPI4.2 接口参考模型

如图 2 所示为 SPI-4.2 将数据包分片映射的图示。图中 PCW 为载荷控制字 (payload control word), ICW 为空闲控制字 (idle control word), 载荷 (payload) 是 SPI-4.2 传输的有效数据。传输时要对数据包分片, 一片称为一个数据突发 (data burst) (两个控制字之间部分), 除了数据包的最后一部分, 数据包必须分割成 16 字节整数倍大小的数据突发。

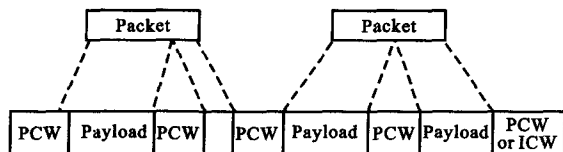


图 2 数据包分片封装

在数据通道, 被传输的数据首尾都要添加一个两字节控制字, 用来描述被传输的数据帧或分组的目的接口、头尾信息和必要的校验码, 没有数据要传输时, 数据总线上为空闲控制字。控制字的结构见表 1。

表 1 控制字的结构

字段	15	14~13	12	11~4	3~0
功能说明	控制字类型	分组结尾状态	分组起始	接口地址	DIP-4 校验码

控制字的采用, 相当于对被传输的数据进行了一次封装, 用较小的带宽作代价, 减少了管脚和高速信号线的数量, 降低了 PCB 设计的难度。

在数据传输过程中, 由于时钟漂移, 时钟相位发生变化, 原本已经同步的采样时钟和数据的相位关系也会被打破。为了及时发现这种传输错误, 在 SPI-4.2 接口中, 对数据总线上的数据都要做对角交叉校验, 图 3 为 DIP-4 码的校验范围。如果出现校验错误, 那么就要停止数据传输并发送训练序列, 重新获取时钟同步后再进行数据传输。

3 对角交叉校验码 (DIP-4) 检错性能分析

假设随机比特错误存在, 对于突发传输长度为 64 字节的情况下, 对 DIP-4 码的检错能力进行评估。

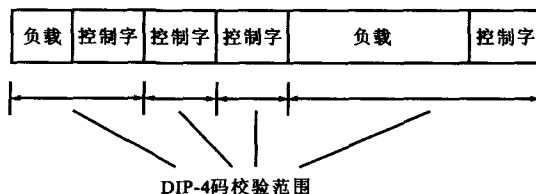


图 3 DIP-4 码的校验范围

使用 DIP-4 码, 每一个校验比特是对 $66 \times 8/4 - 1$ 的信息位进行校验计算。这些信息位和这些校验位组成了一个简单的校验码字。在假设奇校验的情况下, 这个校验位的值是通过码字中 1 的个数是否为奇数来计算的。很容易看出任何奇数的错误能够通过简单的校验码字检测出来。因此对于给定的码字来说只有当偶数个错误出现在一个码字的范围内时, 这些错误才检测不出来。因此对于一个给定的突发传输来说, 至少在一个码字中出现未检测到错误的情况时, 整个突发传输才会有未检测到错误的情况出现。

让 P_{UC} 和 P_{UT} 分别表示在一个单一校验码字和一个 DIP-4 突发传输中未检测到的错误概率, 用 L 来表示一个码字中的比特数目, 用 P 来表示出现一比特错误的概率。列举在一个码字中偶数错误的所有组合情况, 可以得到:

$$P_{UC} = \sum_{i=2, i \text{ even}}^L \binom{L}{i} p^i (1-p)^{L-i}$$

因为在一个突发传输中一个未检测到的错误就对应着至少在一个码字中未检测到的错误, 因此有

$$P_{UT} = 1 - (1 - P_{UC})^4$$

在表 2 中列举了对于 DIP-4 码来说在一个突发传输中相应的未检测到错误的概率, 相应的也对没有错误检测码时对应的概率情况做了一个对比。可以看出 DIP-4 码可以减少未检测到的错误概率几个量级。因为在一个设计精良的通信系统中, 比特错误率已经非常低了, 所以 DIP-4 码能够把未检测到的错误概率降低到可以忽略不计的程度。

表 2 未检测到错误情况对比

比特错误概率	无检错机制时未检测到错误的概率	使用 DIP-4 码时未检测到错误的概率
1.0E-6	5.28E-4	3.458E-8
1.0E-7	5.28E-5	3.458E-10
1.0E-8	5.28E-6	3.459E-12
1.0E-9	5.28E-7	3.464E-14
1.0E-10	5.28E-8	3.441E-16
1.0E-11	5.28E-9	(tiny)

4 对角交叉校验码(DIP-4)编码规则

由于标准的 SPI-4.2 接口是 16 位的数据总线, 首先介绍分析 16 位数据总线宽度的对角交叉校验编码规则.

由图 3 可知, DIP-4 校验码仅对控制字及其前面的载荷数据进行计算, 如果控制字的前面仍然是控制字, 那么仅计算控制字本身. 计算时控制字的低 4 位按全为 1 计算, 计算的结果填写在控制字的低 4 位. 图 4 对 DIP-4 校验码进行了功能型描述. 第一次到达的 16 比特数据循环右移一位后, 与第二次到达的 16 比特数据进行模 2 加运算, 运算的结果再循环右移一位, 依此类推, 遇到控制字停止, 将得到的 16 位校验和的高 8 位和低 8 位进行模 2 加运算, 得到 8 位校验和, 再将高 4 位与低 4 位模 2 加, 得到 4 位的 DIP-4 校验码, 并填写到控制字的低 4 位.

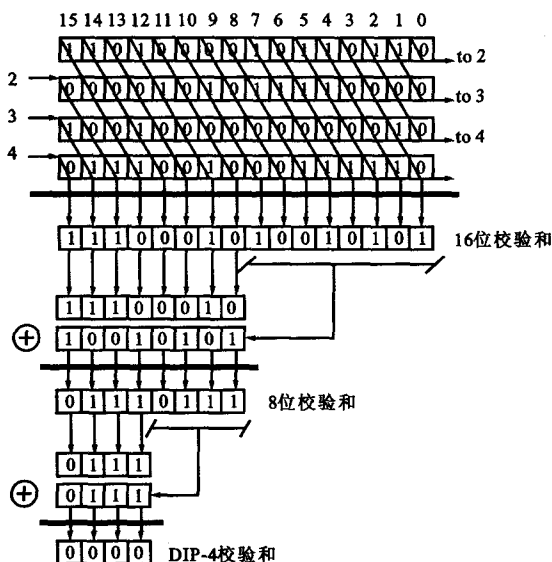


图 4 数据位宽为 16 比特的对角交叉校验算法

对 16 位数据总线, 要达到 10Gb/s 的数据处理带宽, 编码器的时钟频率就要到 640MHz, 这在专用集成电路中是可以实现的. 然而在商用 FPGA 中, 一般来说时钟频率是难以超过 250MHz 的, 因此, 必须找到一种能对对位宽更宽的数据进行编码的方法, 将数据变换为 64 位甚至 128 位再进行编码, 从而使我们的设计能在 FPGA 中实现.

5 位宽可配置的对角交叉校验码编码方法

为了在硬件 FPGA 中实现, 必须实现高位宽的对角交叉校验编码算法. 在上述 16 位数据对角交叉校验码算法的基础上, 我们可以得出数据位宽为 $16X (X \geq 1)$ 的数据的对角交叉校验码的算法.

在对数据位宽为 $16X$ 比特的数据 $DATA[16X, 0]$ 进行校验编码时, 以 16 比特数据为一段, 将数据 $DATA[16X, 0]$ 分成 X 个段, 从高位到低位依次为 $Seg[i, X], Seg[i, X-1], \dots, Seg[i, 1]$, 其中 i 为数据到达的次序. 把每次到达的 $16X$ 比特数据中处于相同段的 16 比特数据, 按照与图 4 中类似的方法进行运算, 不同之处在于, 16 比特数据每次循环右移 X 位后与下一次到达的数据中处于相同段的数据进行模 2 加运算, 而不是循环右移一位. 将所有位于 n 段的数据进行模 2 运算后得到 X 组 16 比特校验和 $sum[n]$, 其中 $1 \leq n \leq X$, 循环右移 $n-1$ 位, 得到 X 组新的 16 比特校验和 $sum[n]$, 再将 X 组校验和 $sum[n]$ 进行模 2 加运算, 得到一组 16 比特校验和 $Checksum16$, 将校验和 $Checksum16$ 的高 8 位与低 8 位进行模 2 加运算, 得到 8 比特校验和 $Checksum8$, 将 $Checksum8$ 的高 4 位与低 4 位模 2 加运算, 得到最后的 4 比特对角交叉校验码 DIP-4.

图 5 中 $16X$ 比特数据的最后一行中, 如果段 $X-1$ 即段 $Seg[4, X-1]$ 为控制字, 则本段载荷数据的 DIP-4 校验码的计算范围包括载荷数据和段 $Seg[4, X-1]$ 的控制字, 其后填充的控制字全部按 0 计算, 并且要对计算的结果进行修正. 修正的方法是, 在最后一行中, 如果本行第一个控制字在段 n 上, 那么将计算得到的 4 位 DIP-4 校验码循环左移位. 这是由于虽然段 $Seg[4, n]$ 之后全按 0 计算, 但前面的数据与这些全为 0 的数据进行模 2 加时, 仍然要作循环右移, 所以我们通过循环左移来修正, 从而得到正确的校验和.

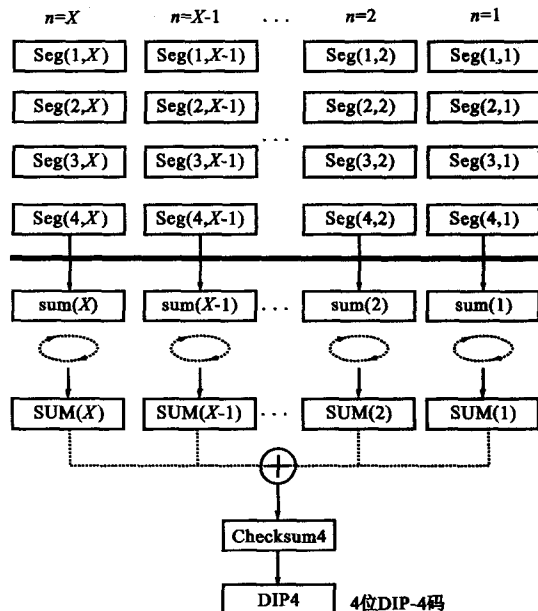


图 5 数据位宽为 $16X$ 的对角交叉校验算法

这样做看起来似乎有些麻烦和多余,但用硬件实现时是简单高效的.图 6 给出了位宽可配置为 16X 的对角交叉校验码的门级电路图.

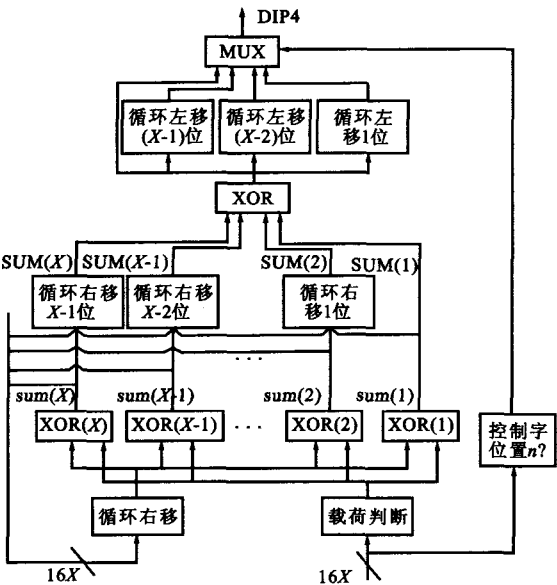


图 6 数据位宽为 16X 比特的对角交叉校验码的门级电路图

6 128 位数据的对角交叉校验码的 FPGA 实现

在我们的 FPGA^[3]设计中,数据处理的总线宽度设置为 128 位,因此,将上面介绍的数据位宽可配置的对角交叉校验算法配置为 128 位,即 $k=8$. 此时校验码编码器和解码器的时钟频率为 80MHz,处理能力为 10.24Gb/s,能够满足 10Gb/s 千兆以太网接口的数据处理需求.

在实际实现时,对每个数据包计算 DIP-4 校验码,DIP-4 校验的数据是每个数据包中的数据和其后的 16bit 控制数据,在输入时这 16bit 的最后 4bit 预置为“1111”,硬件计算后的 DIP-4 校验结果会替代这 4bit 数据,封装在控制字中输出.在接收端可以依此来检测数据包是否出现错误.表 3 是本模块实现时的输入输出列表.图 7 是本模块实现后用 modelsim 的仿真结果,图 8 是实际在 FPGA 内部用 chipscope 采样的结果.

7 结束语

本文提出了 SPI-4.2 接口标准中对角交叉校验

表 3 dip-128 模块接口说明

信号名称	方向	位宽	信号功能
reset	In	1	置‘1’复位
clk	In	1	Spi 同步时钟
data_in	In	128	需要进行 DIP 校验计算的数据
request_to_write	Out	1	用来指示有效的数据
data_out	Out	128	带有 DIP 校验结果的数据



图 7 dip-128 模块 modelsim 仿真结果

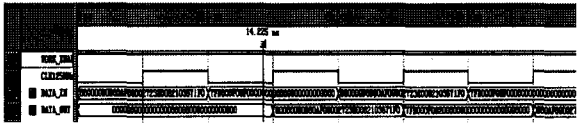


图 8 dip-128 模块相关信号在 FPGA 中采样结果

算法的工程实现方法.此方法适宜于 FPGA 编程实现,数据位宽方便扩展.在实际的电路设计中验证了其正确性,并且比较简单、高效,具有较高的工程应用价值.

参考文献:

[1] Optical Internetworking Forum. System packet interface level 4 (SPI-4)phase 2 revision 1: OC-192 system interface for physical and link layer devices[EB/OL]. (2003-10-15). www.oiforum.com.

[2] 王守军. 高速串行通信技术的发展、设计及应用[EB/OL]. (2006-03-15). http://www.eetchina.com/ART_8800344699_640279_17f79aob200408.HTM.

[3] Jose S. Virtex-4 user guide u9070 (v2.0) [EB/OL]. (2007-01-04). http://agata.pa.inf.nu.it/LLP_Carrier/Ne w_ATCA_Carrier_web/DataSheets/Xilinx/Xilinx-4_User_Guide u9070.pdf.

作者简介:

卜佑军 男,(1978-),博士研究生,讲师.研究方向为宽带信息网络、路由与交换技术.

汪斌强 男,(1963-),博士,教授,博士生导师.

郭虹 女,(1975-),博士研究生.

唐红超 男,(1982-),博士研究生.