



(12) 发明专利

(10) 授权公告号 CN 103559156 B

(45) 授权公告日 2016. 04. 06

(21) 申请号 201310556229. 9

卷 (第 4 期), 第 215-219 页.

(22) 申请日 2013. 11. 11

审查员 张琳

(73) 专利权人 北京大学

地址 100871 北京市海淀区颐和园路 5 号

(72) 发明人 王韬 龚健

(74) 专利代理机构 北京万象新悦知识产权代理

事务所 (普通合伙) 11360

代理人 朱红涛

(51) Int. Cl.

G06F 13/28(2006. 01)

G06F 13/38(2006. 01)

(56) 对比文件

CN 102411553 A, 2012. 04. 11,

CN 102521182 A, 2012. 06. 27,

US 2010/0262738 A1, 2010. 10. 14,

单天昌等. 基于 FPGA 的 PCI 接口 DMA 传输的设计与实现. 《计算机技术与发展》. 2010, 第 20

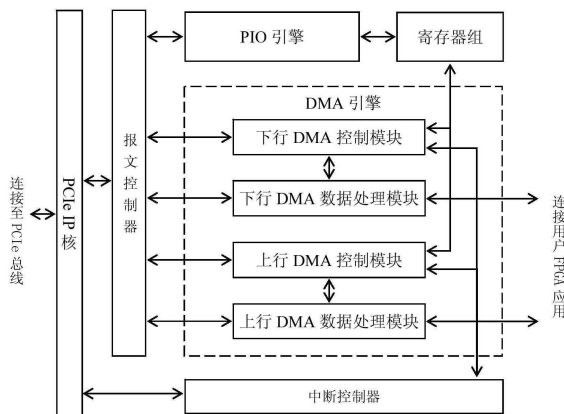
权利要求书 2 页 说明书 7 页 附图 2 页

(54) 发明名称

一种 FPGA 与计算机之间的通信系统

(57) 摘要

本发明提供了一种 FPGA 与计算机之间的通信系统, 包括: DMA 引擎、PIO 引擎、寄存器组、报文控制器、中断控制器和 PCIe IP 核。通过 FPGA 硬件中独立的发送和接收模块实现全双工、高吞吐率的 DMA 数据传输, 通过发送或接收模块中对数据流和控制流的单独处理实现低延迟的数据传输, 通过上述 DMA 数据传输, 将 FPGA 和计算机连接在一起。利用本发明提供的技术方案, 可以提高 FPGA 应用与计算机集成的效率, 加快 FPGA 应用的研发、测试及推向市场速度, 节省了 FPGA 应用与计算机集成所花费的时间与人力投入。



1. 一种 FPGA 与计算机之间的通信系统,其特征是,所述通信系统包括 :DMA 引擎、PIO 引擎、寄存器组、报文控制器、中断控制器和 PCIe IP 核 ;其中,

(1) 所述 DMA 引擎包括下行 DMA 控制模块、下行 DMA 数据处理模块、上行 DMA 控制模块、上行 DMA 数据处理模块,其中,

(1.1) 下行 DMA 控制模块用于对计算机到 FPGA 方向的 DMA 进行控制 ;

(1.2) 下行 DMA 数据处理模块用于对计算机到 FPGA 方向的 DMA 数据进行处理 ;

(1.3) 上行 DMA 控制模块用于对 FPGA 到计算机方向的 DMA 进行控制 ;

(1.4) 上行 DMA 数据处理模块用于对 FPGA 到计算机方向的 DMA 数据进行处理 ;

(2) 所述 PIO 引擎对 PIO 的请求进行处理,对于 PIO 读请求,PIO 引擎将向寄存器组给出 PIO 读地址,当寄存器组返回读取的数据时,PIO 引擎通过报文控制器和 PCIe IP 核向 PCIe 总线上回复读取到的数据 ;对于 PIO 写请求,PIO 引擎向寄存器组给出 PIO 写地址和数据,以使数据被写入寄存器中 ;

(3) 所述寄存器组由一系列的状态与控制寄存器构成,用于对系统进行控制,包括 DMA 读开始寄存器、DMA 写开始寄存器、DMA 读操作描述符地址寄存器、DMA 写操作描述符地址寄存器 ;

(4) 所述报文控制器通过 PCIe IP 核与 PCIe 总线连接,控制数据的发送和接收 ;

(4.1) 报文控制器将通过 PCIe IP 核接收到的不同类别的数据分发到 PIO 引擎或 DMA 引擎中的下行 DMA 控制模块、下行 DMA 数据处理模块、上行 DMA 控制模块、上行 DMA 数据处理模块中 ;

(4.2) 报文控制器从 PIO 引擎或 DMA 引擎中的下行 DMA 控制模块、下行 DMA 数据处理模块、上行 DMA 控制模块、上行 DMA 数据处理模块接收数据,并通过 PCIe IP 核将数据发送到 PCIe 总线 ;

(5) 所述中断控制器通过 PCIe IP 核与 PCIe 总线连接,控制中断请求的发送 ;

(5.1) 中断控制器接收 DMA 引擎中下行 DMA 控制模块、上行 DMA 控制模块的中断请求,当 DMA 完成或出现错误时,中断控制器将收到 DMA 读或 DMA 写的完成中断请求,或中断控制器将收到 DMA 读或 DMA 写的出错中断请求 ;

(5.2) 中断控制器通过 PCIe IP 核将中断请求发送到 PCIe 总线上,进而引发计算机系统的中断过程 ;

(6) 所述 PCIe IP 核用于和 PCIe 总线进行交互,将 PCIe 总线信号中的数据提取出来。

2. 如权利要求 1 所述的 FPGA 与计算机之间的通信系统,其特征是,所述通信系统还包括 PIO 接口模块,该模块为用户 FPGA 应用提供寄存器接口,该模块从 PIO 引擎获得 PIO 请求,完成对用户寄存器的读写操作。

3. 如权利要求 1 所述的 FPGA 与计算机之间的通信系统,其特征是,所述通信系统还包括中断接口模块,该模块为用户 FPGA 应用提供中断接口,该模块将用户 FPGA 应用的中断请求传递到中断控制器模块中,并由中断控制器通过 PCIe IP 核产生 PCIe 总线上的中断。

4. 如权利要求 1 所述的 FPGA 与计算机之间的通信系统,其特征是,所述通信系统还包括排序模块,该模块对下行 DMA 数据进行重排序,并为用户 FPGA 应用提供连续的数据。

5. 如权利要求 1 所述的 FPGA 与计算机之间的通信系统,其特征是,所述 DMA 引擎包括下行 DMA 控制模块、下行 DMA 数据处理模块、上行 DMA 控制模块、上行 DMA 数据处理模块,其

中：

(1.1) 下行 DMA 控制模块用于对计算机到 FPGA 方向的 DMA 进行控制, 该方向的 DMA 也叫 DMA 读, 依据寄存器组给出的 DMA 读开始、DMA 读描述符地址信号对 DMA 描述符进行 DMA 操作, 然后通过 DMA 描述符中的 DMA 地址和 DMA 长度信息控制下行 DMA 数据处理模块进行 DMA 读操作；

(1.2) 下行 DMA 数据处理模块用于对计算机到 FPGA 方向的 DMA 数据进行处理, 依据下行 DMA 控制模块输出的 DMA 地址和 DMA 长度信息, 对数据进行 DMA 操作, 并将 DMA 的数据输出给用户 FPGA 应用；

(1.3) 上行 DMA 控制模块用于对 FPGA 到计算机方向的 DMA 进行控制, 该方向的 DMA 也叫 DMA 写, 依据寄存器组给出的 DMA 写开始、DMA 写描述符地址信号对 DMA 描述符进行 DMA 操作, 然后通过 DMA 描述符中的 DMA 地址和 DMA 长度信息控制上行 DMA 数据处理模块进行 DMA 写操作；

(1.4) 上行 DMA 数据处理模块用于对 FPGA 到计算机方向的 DMA 数据进行处理, 从用户 FPGA 应用获得将要被 DMA 的数据, 并依据上行 DMA 控制模块输出的 DMA 地址和 DMA 长度信息, 对数据进行 DMA 写操作。

## 一种 FPGA 与计算机之间的通信系统

### 技术领域

[0001] 本发明提供一种 FPGA（现场可编程门阵列, Field-Programmable Gate Array）与计算机之间的通信系统, 具体涉及一种高效、易用的 FPGA 与计算机之间的 PCIe 通信系统。

### 背景技术

[0002] 现场可编程门阵列(FPGA)是一种可以编程的电路器件。随着 FPGA 技术的发展, FPGA 芯片中的资源不断增加, 功能和运算能力也有了很大的提升。很多研究者采用 FPGA 对算法进行加速。因此 FPGA 加速器和计算机之间高效、易用的通信就成为了一种需求。

[0003] PCIe (PCI-Express, 快捷外设互联标准) 总线是一种高速的计算机总线, 它使用串行的、点对点的方式来连接计算机的外部设备。PCIe 总线单条链路的单个方向可以达到 2Gbps (1.0 协议)、4Gbps (2.0 协议) 或近 8Gbps (3.0 协议) 的数据传输速率。一个 PCIe 设备还可以同时使用多条串行链路进行数据传输, 因此能够达到很高的数据传输速率。

[0004] MPRace (G. Marcus, W. Gao, A. Kugel, and R. Manner. The mprace framework: An open source stack for communication with custom fpga-based accelerators. In Programmable Logic (SPL), 2011 VII Southern Conference on, pages 155 - 160, 2011.) 是一个使用 PCIe 连接 FPGA 和计算机的系统。该系统使用 DDR RAM (double data rate random access memory) 接口作为和用户硬件的数据接口, 并提供若干寄存器作为用户硬件的控制接口。在软件方面, 该系统提供的软件接口暴露了很多数据传输过程中的细节, 抽象层次较低。这种方法只能对于采用 DDR 接口的用户硬件提供较大的方便, 且由于软件抽象层次较低, 使用比较麻烦。

[0005] Speedy Bus Mastering PCI Express (R. Bittner. Speedy bus mastering pci express. In Field Programmable Logic and Applications (FPL), 2012 22nd International Conference on, pages 523 - 526, 2012.) 是一个将 PCIe 总线信号转化为局部总线的系统。该系统的传输控制都需要用户自己完成, 并且没有提供对应的软件支持。该系统提供了一个带有 DDR RAM 的示例, 并对这个示例提供了对应的软件程序, 但是该示例不具有通用性。

[0006] 一种 PCIe 多功能设备和硬件加速算法集成装置 (实用新型专利申请号 201220109341.9) 是一个具有片上总线结构的加速算法装置。该装置虽然可以被用于 FPGA 与计算机的通信, 但只能支持一个算法加速逻辑设备, 降低了系统的可用性。且没有在软件端对系统提供支持, 不易于用户使用。且该装置的数据通路中存在其它装置, 增加了数据传输延迟, 降低了数据传输性能。

### 发明内容

[0007] 为了便于说明, 本文约定: “DMA” 表示直接内存访问, 即 Directly memory access。“PIO” 表示 programmed input/output, “IP 核” 表示已验证、可重用并具有某种确定功能

的硬件电路模块,即 Intellectual Property Core。

[0008] 本发明的目的是提供一种基于 PCIe 总线的通信系统,用以解决 FPGA 与计算机之间高效、易用的数据通信的问题。

[0009] 本发明的原理是:通过 FPGA 硬件中独立的发送和接收模块实现全双工、高吞吐率的 DMA 数据传输,通过发送或接收模块中对数据流和控制流的单独处理实现低延迟的数据传输,通过上述 DMA 数据传输,将 FPGA 和计算机连接在一起。

[0010] 本发明提供的技术方案如下:

[0011] 一种 FPGA 与计算机之间的通信系统,包括:DMA 引擎、PIO 引擎、寄存器组、报文控制器、中断控制器和 PCIe IP 核;其中,

[0012] (1) 所述 DMA 引擎包括下行 DMA 控制模块、下行 DMA 数据处理模块、上行 DMA 控制模块、上行 DMA 数据处理模块,其中,

[0013] (1.1) 下行 DMA 控制模块用于对计算机到 FPGA 方向的 DMA 进行控制,该方向的 DMA 也叫 DMA 读,依据寄存器组给出的 DMA 读开始、DMA 读描述符地址信号对 DMA 描述符进行 DMA 操作,然后通过 DMA 描述符中的 DMA 地址和 DMA 长度信息控制下行 DMA 数据处理模块进行 DMA 读操作;

[0014] (1.2) 下行 DMA 数据处理模块用于对计算机到 FPGA 方向的 DMA 数据进行处理,依据下行 DMA 控制模块输出的 DMA 地址和 DMA 长度信息,对数据进行 DMA 操作,并将 DMA 的数据输出给用户 FPGA 应用;

[0015] (1.3) 上行 DMA 控制模块用于对 FPGA 到计算机方向的 DMA 进行控制,该方向的 DMA 也叫 DMA 写,依据寄存器组给出的 DMA 写开始、DMA 写描述符地址信号对 DMA 描述符进行 DMA 操作,然后通过 DMA 描述符中的 DMA 地址和 DMA 长度信息控制上行 DMA 数据处理模块进行 DMA 写操作;

[0016] (1.4) 上行 DMA 数据处理模块用于对 FPGA 到计算机方向的 DMA 数据进行处理,从用户 FPGA 应用获得将要被 DMA 的数据,并依据上行 DMA 控制模块输出的 DMA 地址和 DMA 长度信息,对数据进行 DMA 写操作;

[0017] (2) 所述 PIO 引擎对 PIO 的请求进行处理,对于 PIO 读请求,PIO 引擎将向寄存器组给出 PIO 读地址,当寄存器组返回读取的数据时,PIO 引擎通过报文控制器和 PCIe IP 核向 PCIe 总线上回复读取到的数据;对于 PIO 写请求,PIO 引擎向寄存器组给出 PIO 写地址和数据,以使数据被写入寄存器中;

[0018] (3) 所述寄存器组由一系列的状态与控制寄存器构成,用于对系统进行控制,包括 DMA 读开始寄存器、DMA 写开始寄存器、DMA 读操作描述符地址寄存器、DMA 写操作描述符地址寄存器;

[0019] (4) 所述报文控制器通过 PCIe IP 核与 PCIe 总线连接,控制数据的发送和接收;

[0020] (4.1) 报文控制器将通过 PCIe IP 核接收到的不同类别的数据分发到 PIO 引擎或 DMA 引擎中的下行 DMA 控制模块、下行 DMA 数据处理模块、上行 DMA 控制模块、上行 DMA 数据处理模块中;

[0021] (4.2) 报文控制器从 PIO 引擎或 DMA 引擎中的下行 DMA 控制模块、下行 DMA 数据处理模块、上行 DMA 控制模块、上行 DMA 数据处理模块接收数据,并通过 PCIe IP 核将数据发送到 PCIe 总线;

- [0022] (5)所述中断控制器通过与PCIe IP核与PCIe总线连接,控制中断请求的发送;
- [0023] (5.1)中断控制器接收DMA引擎中下行DMA控制模块、上行DMA控制模块的中断请求,当DMA完成或出现错误时,中断控制器将收到DMA读或DMA写的完成或出错中断请求;
- [0024] (5.2)中断控制器通过PCIe IP核将中断请求发送到PCIe总线上,进而引发计算机系统的中断过程;
- [0025] (6)所述PCIe IP核用于和PCIe总线进行交互,将PCIe总线信号中的数据提取出来。
- [0026] 所述的FPGA与计算机之间的通信系统,还包括PIO接口模块,该模块为用户FPGA应用提供寄存器接口,该模块从PIO引擎获得PIO请求,完成对用户寄存器的读写操作。
- [0027] 所述的FPGA与计算机之间的通信系统,还包括中断接口模块,该模块为用户FPGA应用提供中断接口,该模块将用户FPGA应用的中断请求传递到中断控制器模块中,并由中断控制器通过PCIe IP核产生PCIe总线上的中断。
- [0028] 所述的FPGA与计算机之间的通信系统,还包括排序模块,该模块对下行DMA数据进行重排序,并为用户FPGA应用提供连续的数据。
- [0029] 本发明的有益效果:利用本发明提供的技术方案,可以提高FPGA应用与计算机集成的效率,加快FPGA应用的研发、测试及推向市场速度,节省了FPGA应用与计算机集成所花费的时间与人力投入。

## 附图说明

- [0030] 图1本发明的系统结构图
- [0031] 图2扩展的系统结构图

## 具体实施方式

- [0032] 本发明的具体实施方式如下:
- [0033] 一种FPGA与计算机之间的通信系统,包括:DMA引擎、PIO引擎、寄存器组、报文控制器、中断控制器和PCIe IP核(参见图1);其中,
- [0034] A. DMA引擎包括下行DMA控制模块、下行DMA数据处理模块、上行DMA控制模块、上行DMA数据处理模块,其中,
- [0035] a) 下行DMA控制模块用于对计算机到FPGA方向的DMA(该方向的DMA也叫DMA读)进行控制,依据寄存器组给出的DMA读开始、DMA读描述符地址信号对DMA描述符进行DMA操作,然后通过DMA描述符中的DMA地址和DMA长度信息控制下行DMA数据处理模块进行DMA读操作;
- [0036] b) 下行DMA数据处理模块用于对计算机到FPGA方向的DMA(该方向的DMA也叫DMA读)数据进行处理,依据下行DMA控制模块输出的DMA地址和DMA长度信息,对数据进行DMA操作,并将DMA的数据输出给用户FPGA应用;
- [0037] c) 上行DMA控制模块用于对FPGA到计算机方向的DMA(该方向的DMA也叫DMA写)进行控制,依据寄存器组给出的DMA写开始、DMA写描述符地址信号对DMA描述符进行DMA操作,然后通过DMA描述符中的DMA地址和DMA长度信息控制上行DMA数据处理模块进行DMA写操作;

[0038] d) 上行 DMA 数据处理模块用于对 FPGA 到计算机方向的 DMA (该方向的 DMA 也叫 DMA 写) 数据进行处理, 从用户 FPGA 应用获得将要被 DMA 的数据, 并依据上行 DMA 控制模块输出的 DMA 地址和 DMA 长度信息, 对数据进行 DMA 写操作;

[0039] B. PIO 引擎对 PIO 的请求进行处理, 对于 PIO 读请求, PIO 引擎将向寄存器组给出 PIO 读地址, 当寄存器组返回读取的数据时, PIO 引擎通过报文控制器和 PCIe IP 核向 PCIe 总线上回复读取到的数据; 对于 PIO 写请求, PIO 引擎向寄存器组给出 PIO 写地址和数据, 以使数据被写入寄存器中;

[0040] C. 寄存器组由一系列的状态与控制寄存器构成, 用于对系统进行控制, 包括 DMA 读开始寄存器、DMA 写开始寄存器、DMA 读操作描述符地址寄存器、DMA 写操作描述符地址寄存器(可选的, 寄存器组还可以包括其它寄存器, 比如当前系统的状态寄存器, 硬件版本号寄存器等等);

[0041] D. 报文控制器通过 PCIe IP 核与 PCIe 总线连接, 控制数据的发送和接收;

[0042] a) 报文控制器将通过 PCIe IP 核接收到的不同类别的数据分发到 PIO 引擎或 DMA 引擎中的下行 DMA 控制模块、下行 DMA 数据处理模块、上行 DMA 控制模块、上行 DMA 数据处理模块中;

[0043] b) 报文控制器从 PIO 引擎或 DMA 引擎中的下行 DMA 控制模块、下行 DMA 数据处理模块、上行 DMA 控制模块、上行 DMA 数据处理模块接收数据, 并通过 PCIe IP 核将数据发送到 PCIe 总线;

[0044] E. 中断控制器通过与 PCIe IP 核与 PCIe 总线连接, 控制中断请求的发送;

[0045] a) 中断控制器接收 DMA 引擎中下行 DMA 控制模块、上行 DMA 控制模块的中断请求, 当 DMA 完成或出现错误时, 中断控制器将收到 DMA 读或 DMA 写的完成或出错中断请求;

[0046] b) 中断控制器通过 PCIe IP 核将中断请求发送到 PCIe 总线上, 进而引发计算机系统的中断过程;

[0047] F. PCIe IP 核用于和 PCIe 总线进行交互, 将 PCIe 总线信号中的数据提取出来;

[0048] 优选的, 所述 FPGA 与计算机之间的通信系统还可以包括(参见图 2):

[0049] G. PIO 接口模块, 为用户 FPGA 应用提供寄存器接口。PIO 接口模块从 PIO 引擎获得 PIO 请求, 完成对用户寄存器的读写操作;

[0050] H. 中断接口模块, 为用户 FPGA 应用提供中断接口。中断接口模块可以将用户 FPGA 应用的中断请求传递到中断控制器模块中, 并由中断控制器通过 PCIe IP 核产生 PCIe 总线上的中断;

[0051] I. 排序模块, 对下行 DMA 数据进行重排序。下行 DMA 数据在经过 PCIe 总线过程中, 可能会产生先请求的数据后到达的现象。排序模块可以将数据依据请求的顺序进行排序, 并为用户 FPGA 应用提供连续的数据。

[0052] 下面通过实例对本发明做进一步说明:

[0053] 实施例 1: 计算机到 FPGA 方向的 DMA 数据传输

[0054] 假定本发明所述的 FPGA 与计算机之间的通信系统收到了用户软件的 DMA 请求, 需要将内存 0x45000004 位置开始的字符串“1234”通过 DMA 数据传输, 传递给用户 FPGA 应用。而软件已经将描述上述 DMA 操作的 DMA 描述符放在内存 0x43000000 位置。

[0055] 本发明所述的 FPGA 与计算机之间的通信系统的 PIO 引擎将收到两次 PIO 写请

求,第一次写入寄存器组中 DMA 读操作描述符地址寄存器,写入的值为 0x43000000;第二次写入寄存器组中 DMA 读开始寄存器。此时下行 DMA 控制模块检测到 DMA 读开始,因此按照 DMA 读操作描述符地址寄存器中的值 0x43000000 进行数据请求,该请求会依次经过报文控制器模块和 PCIe IP 核,并传递到 PCIe 总线上。PCIe 总线将响应该请求,并将位于内存 0x43000000 的数据传递给 PCIe IP 核。DMA 描述符会依次经过 PCIe IP 核和报文控制器,被传递到下行 DMA 控制模块。

[0056] 下行 DMA 控制模块通过解析 DMA 描述符,获得 DMA 的地址为 0x45000004,DMA 数据长度为 4 字节,并将 DMA 的地址和长度信息传递给下行 DMA 数据处理模块。

[0057] 下行 DMA 数据处理模块请求从 0x45000004 地址开始的 4 字节的数据,该请求会经过报文控制器模块和 PCIe IP 核模块,并传递到 PCIe 总线上。PCIe 总线将响应该请求,并将从内存 0x45000004 地址开始的 4 字节数据传递给 PCIe IP 核,该数据即为包含有“1234”这一字符串的 PCIe 报文。该数据会依次经过 PCIe IP 核模块和报文控制器模块,并被传递到下行 DMA 数据处理模块。下行 DMA 数据处理模块将对上述包含着“1234”这一字符串的 PCIe 报文进行解包处理,并将其中的数据“1234”这一字符串传递给用户 FPGA 应用。

[0058] 上述 DMA 过程完成后,下行 DMA 控制模块将请求中断控制器模块,发送 DMA 读完成中断,以告知计算机本次 DMA 读操作已经完成。

[0059] 实施例 2:FPGA 到计算机方向的 DMA 数据传输

[0060] 假定本发明所述的 FPGA 与计算机之间的通信系统收到了用户软件的 DMA 请求,需要将 FPGA 中的字符串“5678”传递到计算机中,具体而言是 DMA 到内存 0x46000004 开始的 4 个字节中。软件已经将描述上述 DMA 操作的 DMA 描述符放在内存的 0x44000000 位置。

[0061] 本发明所述的 FPGA 与计算机之间的通信系统的 PIO 引擎将收到两次 PIO 写请求,第一次写入寄存器组中 DMA 写描述符地址寄存器,写入的值为 0x44000000;第二次写入寄存器组中 DMA 写开始寄存器。此时上行 DMA 控制模块检测到 DMA 写开始,因此按照 DMA 写操作描述符地址寄存器中的值 0x44000000 进行数据请求,该请求会依次经过报文控制器模块和 PCIe IP 核,并传递到 PCIe 总线上。PCIe 总线将响应该请求,并将位于内存 0x44000000 的数据传递给 PCIe IP 核。DMA 描述符会依次经过 PCIe IP 核和报文控制器,被传递到上行 DMA 控制模块。

[0062] 上行 DMA 控制模块通过解析 DMA 描述符,获得 DMA 的地址为 0x46000004,DMA 数据长度为 4 字节,并将 DMA 的地址和长度信息传递给上行 DMA 数据处理模块。

[0063] 上行 DMA 数据处理模块将用户 FPGA 应用传递过来的头 4 个字节进行打包,生成 PCIe 总线中的存储器写请求报文,并通过报文控制器与 PCIe IP 核模块将该请求报文发送到 PCIe 总线上。随后再发送对 0x46000004 地址的读请求报文。该读请求报文用于确定上述写请求报文已经被处理。当对应于 0x46000004 地址的读请求报文的回复报文依次经过 PCIe IP 核和报文控制器模块后,上行 DMA 数据处理模块确认 DMA 数据已经被处理。

[0064] 上述过程完成后,上行 DMA 控制模块将请求中断控制器模块,发送 DMA 写完成中断,以告知计算机本次 DMA 写操作已经完成。

[0065] 实施例 3:计算机到 FPGA 方向的 PIO 数据传输

[0066] 在附带有 PIO 接口模块的系统中,假定本发明所述的 FPGA 与计算机之间的通信系统收到了用户软件的 PIO 写请求,需要将 0 号寄存器写入 0x00030002。



[0067] 软件的PIO写请求会通过PCIe总线传递到PCIe IP核,并经过报文控制器被传输到PIO引擎。PIO引擎判断写入的地址后,发现软件的PIO写请求写入的寄存器地址位于用户自定义寄存器的地址范围,因此将该PIO请求交由PIO接口模块处理。PIO接口模块判断后,得知软件的PIO写请求需要对0号寄存器写入0x00030002。于是PIO接口模块发送向0号寄存器写入0x00030002的请求。当PIO接口模块收到写0号寄存器请求对应的回复报文后,本次PIO数据传输完成。

[0068] 实施例4:FPGA到计算机方向的PIO数据传输

[0069] 在附带有PIO接口模块的系统中,假定本发明所述的FPGA与计算机之间的通信系统收到用户软件的PIO读请求,需要读取0号寄存器的值。而当前0号寄存器的值为0x00010002。

[0070] 软件的PIO读请求会通过PCIe总线传递到PCIe IP核,并经过报文控制器被传输到PIO引擎。PIO引擎判断读取的地址后,发现软件的PIO读请求的寄存器地址位于用户自定义寄存器地址范围,因此将该PIO请求交给PIO接口模块进行处理。PIO接口模块判断后,得知软件的PIO读请求需要对0号寄存器进行读操作。因此PIO接口模块发送读取0号寄存器请求。当PIO接口模块收到读取0号寄存器请求对应的回复后,会从回复中得到0号寄存器的值为0x00010002。

[0071] 此时PIO接口模块会回复PIO引擎读取到的0号寄存器的值为0x00010002,PIO引擎会依据0号寄存器的值生成回复报文,并将该报文经由报文控制器和PCIe IP核发送到PCIe总线上,从而使软件获得0号寄存器的值为0x00010002。

[0072] 实施例5:用户自定义中断

[0073] 在附带有中断接口模块的系统中,假定本发明所述的FPGA与计算机之间的通信库系统收到用户FPGA应用给出的用户自定义中断请求。

[0074] 如果中断被使能,中断接口模块会将该请求传递给中断控制器模块,中断控制器模块将通过PCIe IP核在PCIe总线上发送中断请求。计算机中的用户软件收到中断信号,即可进行相应的中断处理。

[0075] 实施例6:排序模块的使用

[0076] 在附有排序模块的系统中,假定本发明所述的FPGA与计算机之间的通信库系统接收到软件的DMA读请求,需要将内存中0x45000000开始的256字节DMA到FPGA中。

[0077] 和实施例1中描述的过程类似,本发明所述的系统将向PCIe总线发送存储器读请求报文,假设发送了两个存储器读请求报文,分别请求128字节的数据。即请求报文1请求了从地址0x45000000开始的128字节的数据,请求报文2请求了从地址0x45000080开始的128字节的数据。假设对于请求报文1,PCIe总线上的回复报文为回复报文1-1和回复报文1-2,它们各包含了64字节的数据;对于请求报文2,PCIe总线上的回复报文为回复报文2-1和回复报文2-2,它们各包含了64字节的数据。用户FPGA应用需要获得的数据顺序应该是:回复报文1-1中的数据,回复报文1-2中的数据,回复报文2-1中的数据,回复报文2-2中的数据。

[0078] 依据PCIe总线的协议,PCIe IP核收到上述回复报文的可能顺序为:回复报文2-1,回复报文1-1,回复报文1-2,回复报文2-2。这时,排序模块会将打乱顺序的四个报文重新进行排序,并向用户FPGA应用传输以下数据:回复报文1-1中的数据,回复报文1-2中

的数据, 回复报文 2-1 中的数据, 回复报文 2-2 中的数据。

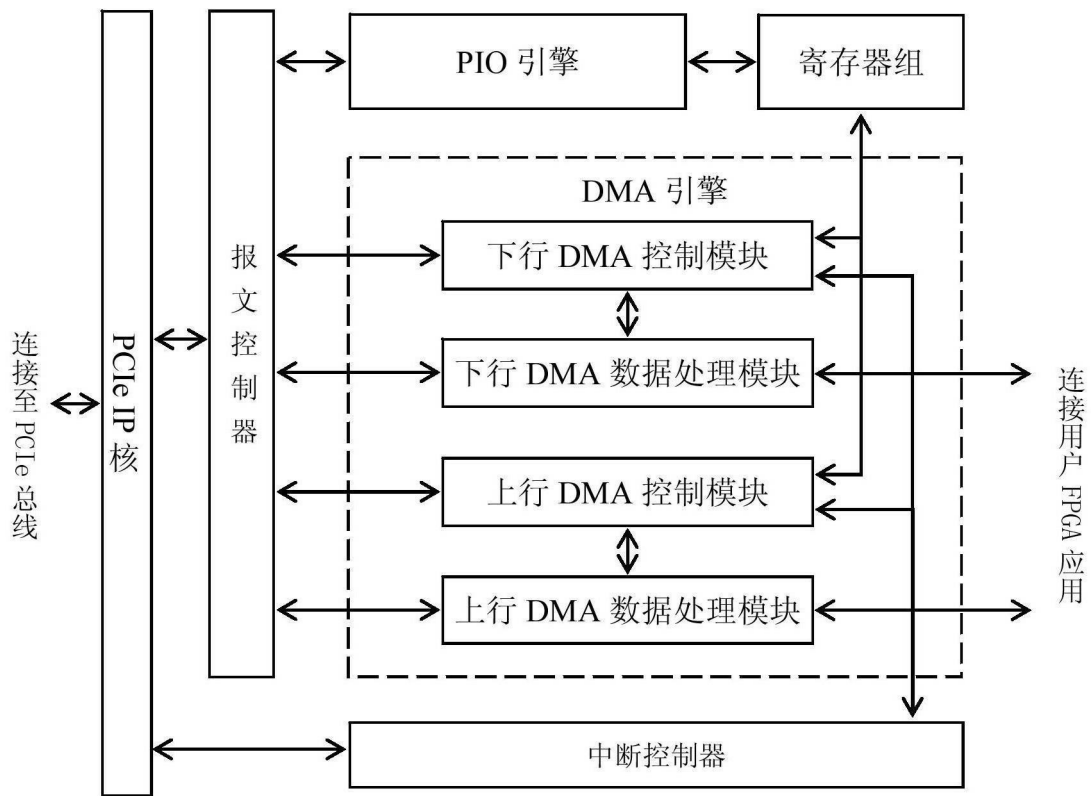


图 1

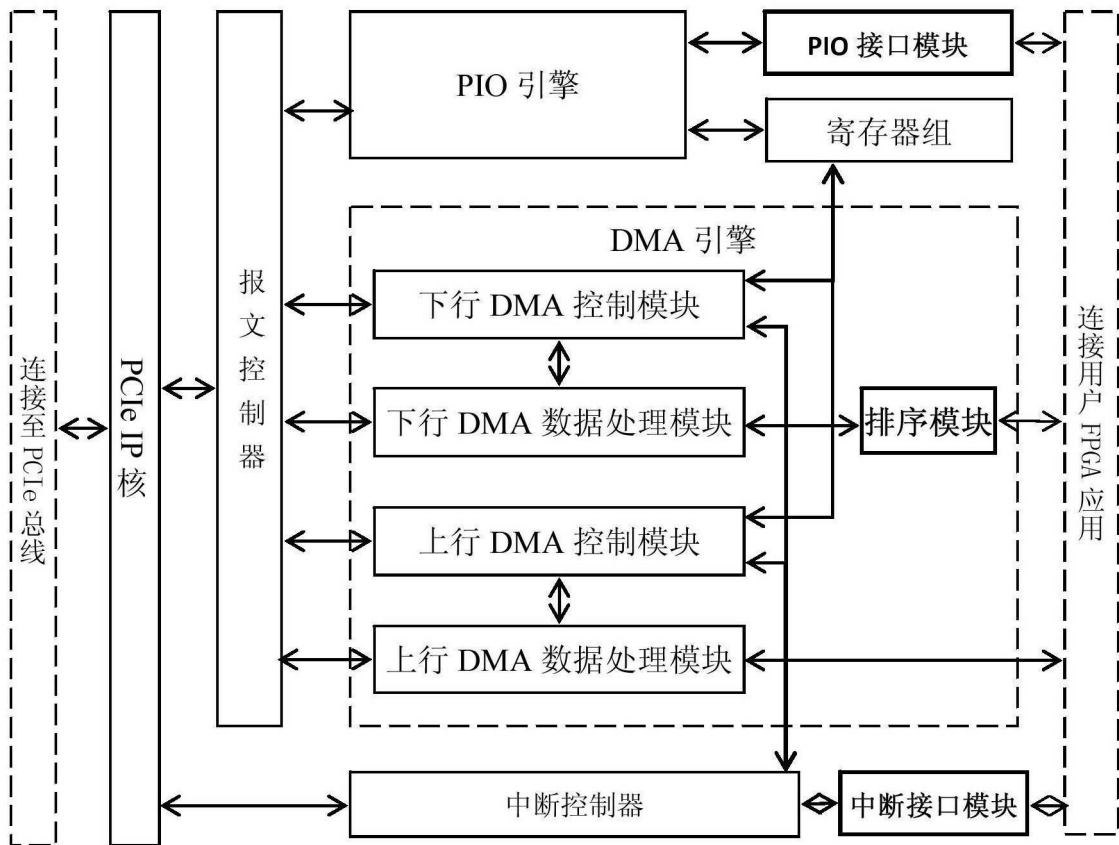


图 2