

引用格式:陈雅泽,景乃锋,王琴. 基于 Rocket-IO 的串行 DDR 实现和评估[J]. 微电子学与计算机, 2019, 36(6): 84-87.
[CHEN Y Z, JING N F, WANG Q. Implementation and evaluation of the rocket-IO based serial DDR memory [J]. Micro-
electronics & Computer, 2019, 36(6): 84-87.]

基于 Rocket-IO 的串行 DDR 实现和评估

陈雅泽, 景乃锋, 王 琴

(上海交通大学 电子信息与电气工程学院, 上海 200240)

摘 要: 由于 FPGA 的 IO 端口数量有限, 导致 FPGA 往往只能搭载少量 DDR 存储器, 限制了大数据应用的内存容量. 本文通过对传统 DDR 存储器接口的改进, 利用多种串行接口协议实现了存储数据的高速传输, 有效地提升了存储容量. 实验中, 我们利用少量的 IO 资源实现了 4 倍的容量扩展, 并且对比评估了不同高速串行接口协议的实现, 为基于串行接口的 DDR 设计实现提供了有益的参考.

关键词: 高速串行总线; DDR 接口; FPGA 设计

中图分类号: TP336

文献标识码: A

文章编号: 1000-7180(2019)06-0084-04

Implementation and evaluation of the rocket-IO based serial DDR memory

CHEN Ya-ze, JING Nai-feng, WANG Qin

(School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong
University, Shanghai 200240, China)

Abstract: Due to the amount limitation of FPGA IO, there is little DDR storage can be used. Furthermore, the memory capacity is the restriction in the big data application. In this paper, we modify the DDR interface design, using various protocols to accomplish the high speed data transaction, improving the memory capacity efficiently. In the experiment we get 4 times more storage capacity by using little IO resource, and compare several protocols implementation in the design, providing a useful reference for serial interface-based DDR design.

Key words: High speed serial link; DDR interface; FPGA evaluation board

1 引言

FPGA 开发板由于存储容量有限且受 IO 端口限制, 存在存储容量不易扩展等问题, 使其在面对大数据应用原型验证开发时存在着诸多的限制. Xilinx 提出了 FPGA 和 HBM(high bandwidth memory)或者 HMC(hybrid memory cube)三维结构提高数据的吞吐量以及存储容量, 但是功耗和成本十分高昂^[1]. 本文在 FPGA 串行总线扩展基础上, 利用低成本的 FPGA 开发板, 搭建了一款 DDR 串行扩展系统, 并利用多种高速串行接口协议进行了实现,

分析了多种高速串行传输协议所消耗的面积、功耗和延时等指标. 实验表明, 利用串行总线接口可以实现内存容量的大幅扩展, 增加的延时和一次内存读取或写入的时间接近.

2 系统结构设计

本文采用的整体系统如图 1 所示. 使用的是 Xilinx 的 KC705 开发板, 其开发板采用的是 XC7K325T-2FFG900C, 它具有一个 DDR3 的 DIMM 接口, 和 16 个 GTX 高速串行传输收发器, 通过配置可以实现不同的协议. 其中 16 个 GTX 每

一个通道可以提供 10 Gbps 的传输速度,总共可以提供 160 Gbps 的传输速度,完全可以满足 DDR 峰值传输速度的要求^[2].

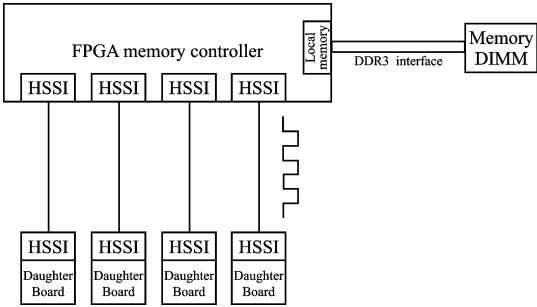


图 1 整体系统结构图

由于 KC705 只有一个 DDR3 接口,而为了实现内存容量的扩展,我们选择了 Xilinx Virtex6 XC6VLX240T-1FFG1156,它有一个 DDR3 接口,同时,还有 24 个高速串行收发器,每个收发器可以支持 6.25 Gbps 的传输速度,在接收到由 KC705 高速串行收发器的数据后,可以将数据存储在 Virtex6 的内存中,从而实现 DDR 容量的扩展.

2.1 DDR3 内存接口设计

本文的 DDR3 接口利用 Xilinx 的提供的 IP 核生成器 CORE Generator 中集成的存储器接口生成器 (MIG) 进行设计. DDR3 的型号选择了 Micron MT8JTF12864HZ 的 1 GB 内存,有 1 个 RANK,最高支持的时钟速度为 800 MHz. MIG 的 IP 提供了用户接口,用于接收和发送数据,在本实验中,从 DDR3 读出来的数据会通过串行的收发器发送至接收端. 写入 DDR3 的数据也是从收发器接收到的数据. 由于串行收发器的参考时钟通常是 156.25 MHz, MIG 用户接口读取和写入数据的时钟是 200 MHz,在这中间需要加入 FIFO 进行数据的缓冲.

MIG 的结构图如下:

在用户接口中, app_addr 用于传输 DDR3 SDRAM 的地址,通常是 bank 地址 + Row 地址 + column 地址; app_cmd 用于传输控制位,为 3 位宽;当 app_cmd 为 3 比特 001 时,表明当前是读指令,当 app_cmd 为全零时,表明当前是写指令; app_en 为地址的使能信号; app_wdf_data 和 app_rd_data 分别是写入和读出的数据; app_wdf_wren 为写使能信号; app_rdy 是标志位,表示当前用户接口可以接收指令. 读写的时序需要满足文献^[3]中要求的时序.

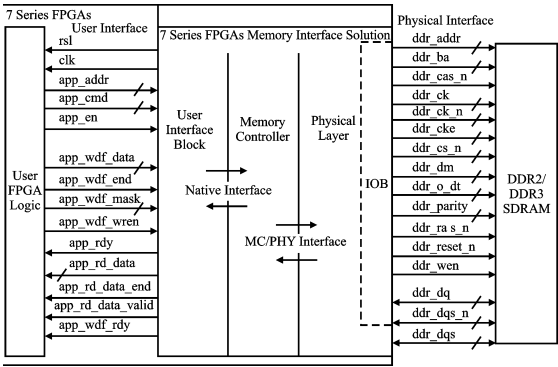


图 2 MIG 用户接口结构图

在本实验中, MIG 的用户接口的数据会从串行总线传输过来,在不同接口的实现中, DDR 接口的设计是相同的. 在接收端接收到字节同步位之后,第一个时钟周期的数据会给 app_addr,第二个时钟周期的数据会给一个计数器实现用户接口地址的控制. 第三个时钟周期的数据会给 app_cmd,用于读写的控制.

2.2 高速串行接口设计

本文选取的开发板 KC705 拥有 16 个 10 Gbps 串行传输的高速串行收发器. 能够支持的高速串行协议包括: PCIe、XAUI、RapidIO 以及 Aurora 等.

XAUI、Rapid IO 和 Aurora 是常用到的基于相似的 SerDes 高速串行总线,具有一定的相似性,通过在 FPGA 上实现这些高速串行接口,评估其消耗的资源以及在串行内存中带来的传输延时^[4-5].

2.2.1 基于 Xilinx FPGA 串行收发器设计

一个收发器的结构如图 3 所示. 它包含两层,一层是物理编码层 (PCS),另一层是物理媒介层 (PMA). PCS 主要包含 8b/10b 编码、同步控制等等,在 PCS 层,可以选择 PCS 近端回环和远端回环的测试,以及伪随机数发生器,方便测试以及在发生问题时,用于隔离问题. PMA 主要包含并串转换和发送端驱动强度的控制,比如预加重、去加重;以及接收端的 PMA 会包含时钟恢复模块等等.

在使用 Xilinx 的收发器的接口时,用户定义的数据从 FPGA 发送端接口发出,经过 8 b/10 b 编码,通过发送端 FIFO,经过并串转换,最后到差分数据的端口. 在接收时,从 FPGA 的差分接收端接收数据,通过做时钟恢复,得到用于接收端的串并转换、8 b/10 b 解码以及通道同步等等功能^[6].

在本文的设计中,由于是采用的 DMA 的方式进行传输,在串行传输的数据中,首先发送的是 8

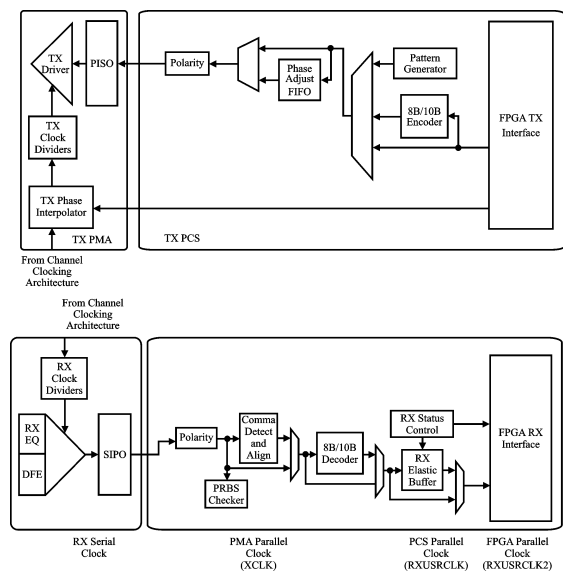


图 3 收发器结构框图

b/10 b 中的 K28. 5, 用于字节同步, 第二个时钟周期数据发送内存地址, 第三个时钟周期数据发送 DMA 需要读取或写入的数据容量, 第四个时钟周期数据发送其余数据, 在没有数据传输时, 会发送 I-DLE 数据。

在接收端, 收到字节同步的数据之后, 会根据地址、数据容量、以及读写指令对 DDR 进行操作。

2.2.2 基于 Xilinx FPGA 的协议接口设计

以 Xilinx 提供的 XAUI IP 为例, 它包含了 HSST 的 wrapper, 以及 XAUI 的 core, 如下图所示。

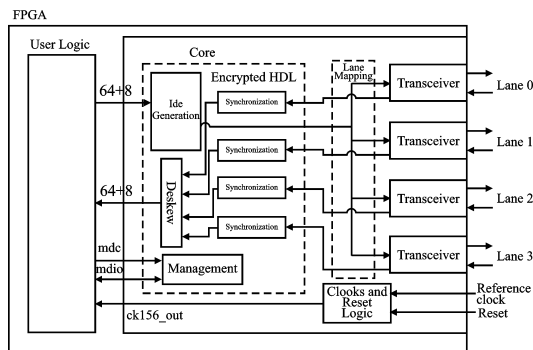


图 4 XAUI 结构图

XAUI 从用户逻辑接口处收到 64 位宽数据, 以及 8 位控制位, 通过 core, 分给 4 个收发器, 并继续传输至 PCB 上, 在接收时, 首先从收发器接收, core 接收到数据后, 对 4 个通道的数据进行同步, 再发给用户逻辑。用户可以通过 mdio 接口, 对 IP 进行配置。没有数据需要发送接收时, 由空闲数据发生器产生数据, 防止总线时钟丢失^[7]。

Aurora 8 b/10 b 的实现与 XAUI IP 类似, 不同的是 Aurora 在发送起始和终止数据时, 需要一个低速的信号 SOF/EOF 进行控制。

而 RapidIO 则在接收和发送之前需要发送读或者写事务请求。

3 实验结果分析

为了避免跨板传输时带来的传输衰减以及提高资源利用率, 本文将系统设计中的 Xilinx V6 FPGA 部分也放在主板的系统中。在选择协议时, 由于 Xilinx 提供的 PCIe 中采用了 AXI 接口, 会消耗额外的资源, 所以我们只选择了 XAUI、RapidIO、Aurora 8b/10b 以及可以自定义协议的 RocketIO。在开发环境的选择上, 采用 vivado 2016. 4 和 Xilinx FPGA KC705 开发板。

3.1 基于串行总线的 DDR3 存储系统

在资源消耗方面, 基于传串行总线的 DDR3 存储系统消耗的资源情况如表 1 所示。

表 1 总工程消耗的 FPGA 资源

资源	使用	可用	百分比%
LUT	18 103	203 800	8. 88
FF	20 615	407 600	5. 06
IO	121	500	24. 20
GT	4	16	25. 00

由于既用到了 MIG, 又用到高速串行总线接口, 所以在 IO 的使用率会比较高。XC7K325T 的 FPGA 一共有 16 个高速传输接口, 每个通道可以支持最大 10 Gbps 的数据传输。在这里, 协议选择了多通道的 RapidIO 协议, 每个通道 6. 25 Gbps, 两个通道用于发送, 两个通道用于接收。所以在这里, 高速串行接口的资源利用率为 25%。

在使用 XAUI 协议时, 由于协议为每个通道的传输速度 3. 125 Gbps, 要实现与 RapidIO 同样的性能, 则需要四组通道发送数据, 以及四组通道接收数据。

Aurora 8 b/10 b 和 RapidIO 类似, 单个通道都可以实现 6. 25 Gbps 速度的传输^[8]。

3.2 不同协议的实现结果对比

在配置收发器的工作状态时, 可以选择数据位宽为 32 bits 或者 64 bits, 但实验结果辨明, 不同参数实验结果影响不大, 资源以及功耗差别并不明显。

通过在 FPGA 上实现不同的协议, 可以看到, 在资源消耗上, 由于 XAUI 协议的单个通道的速度

只有其他协议的一半,所以其消耗的高速传输收发器是其他协议的两倍,在 LUT 和 FF 的资源中,四种串行传输协议的资源利用率比较相似.

针对不同协议的功耗,我们采用功耗计测量收发器一直工作时的功耗. 结果表明,为收发器相同的传输速率,XAUI 需要使用更多的收发器,尽管收发器的工作频率没有其他三者高,但是功耗最高. 由于不同的协议在实现时,其物理层主要是由收发器实现,而这部分的功耗占协议实现消耗的功耗一半以上,所以在不同的协议实现中,功耗差别不明显,如表 2 中功耗一行显示.

对于延时,我们利用接收端数据与发送端数据相差的时钟周期数得到. 在 MIG 中,DDR3 在读取数据时的延时是 20 ns,由高速串行收发器带来的延时也如下表 2 所示,XAUI、RapidIO 和 Aurora8b/10b 由于有对应的 core 设计,用于 4 个通道的同步,以及收发器没有数据传输时,发送 IDLE 数据,所以这三个协议的延时比其他的延时大.

表 2 不同协议实现结果对比

	XAUI	RapidIO	Aurora 8b/10b	Raw data
LUT	3 462	2 067	2 245	2 055
FF	10 382	14 739	15 191	15 638
GT	8	4	4	4
功耗(W)	2. 76	1. 64	1. 63	1. 63
延时(ns)	59	55	52	25

4 结束语

本文利用 FPGA 实现了基于高速串行总线的 DDR 存储系统扩展实现. 本文以 Xilinx KC705 为例,利用 HSSI 实现了 4 倍的内存容量的增加,带来的延时也在可以接受的范围. 实验结果表明,选择不同的传输协议,在同样的数据传输速度下,消耗的资源,增加的延时均有所不同.

参考文献:

[1] LI W, ZHAO Y Y, LIU Y H, et al. SMEFF: A scalable memory extension fabric for FPGA. [C]//International Conference on Field Programmable Technology (ICFPT), 2017, :40-47.

[2] Xilinx. Ug576-ultrascale -gth-transceivers[OL](2018) [2018-9]. https://www.xilinx.com/support/documentation/user_guides/ug576-ultrascale-gth-transceivers.pdf.

[3] Xilinx,UG586 - 7series - MIS[OL],(2018). [2018-9]. https://www.xilinx.com/support/documentation/ip_documentation/mig_7series/v4_1/ug586_7Series_MIS.pdf.

[4] IEEEStandards Association. IEEE 802. 3ae-2002 IEEE Standard for Information technology [S]. 2002.

[5] 刘军伟,多种高速串行总线的对比研究与分析[J]. 电子测试. 2016(3):43-45.

[6] Xilinx. PG053 - XAUI v12. 3 Product Guide[OL], (2017) [2018-9]. https://www.xilinx.com/support/documentation/ip_documentation/xaui/v12_3/pg053-xaui.pdf.

[7] Xilinx,PG046 - Aurora 8B/10B v11. 1 Product Guide [OL], (2018). [2018-9]. https://www.xilinx.com/support/documentation/ip_documentation/aurora_8b10b/v11_1/pg046-aurora-8b10b.pdf.

[8] 苏秀妮,陈建春,那彦. 基于 RocketIO 自定义传输协议在高速串行通信中的设计与实现[J]. 微电子学与计算机, 2013, 30(9):93-90.

作者简介:

陈雅泽 女,(1991-),硕士研究生. 研究方向为集成电路设计. E-mail:chenyaze1991@sitll.edu.cn.

景乃锋 男,(1982-),博士,副研究员. 研究方向为计算机体系结构.

王 琴 女,(1975-),博士,副教授. 研究方向为大规模集成电路设计、先进集成电路设计方法.