International Journal of Research and Reviews in Applied Sciences And Engineering (IJRRASE)
Vol 8. No.1 –2016 Pp.197-203

available at : www.ijcns.com
ISSN: 2231-0061

AN EFFICIENT DELAY MINIMIZATION IN SYSTEM DESIGN USING MICRO BLAZE WITH BRAM

S. Subathradevi
Research Scholar (PT),
A.P., Department of ECE,
Anna University,BIT Campus,
Tiruchirappalli, Tamil nadu, India .

C.Vennila
Professor
Department of ECE
Saranathan College of Engineering
Tiruchirappalli, TamilNadu, India

ABSTRACT - In VLSI System design connecting internally between more numbers of sub modules is a needed and essential one to build the system. There the optimizing speed may not be an important issue, since all the sub blocks need to be connected. But the speed gets determined by the interconnect delay of the system which is organized and contributed by interconnect. In the delay of the design the design delay is contributed by submodule delay and interconnects i.e. path delay. The path delay is mainly contributed by the buses which are used to transfer data. It is essential to concentrate more towards routing delay of the system design to get the optimized delay or desired speed of the design in trade of with area. In this work, by studying sub modules in the architectures used to construct the system, comparison of their working with different basic module for different bus architecture towards the routing delay was studied, realized and implemented using FPGA which results in reduced delay.

KEYWORDS: BRAM, System Design Architecture, VLSI Design, interconnect delay, Micro Blaze, delay optimization, IP core.

I. INTRODUCTION

For any system design, the IP sub modules are much important to make it as a successful design. IP sub modules and peripheral IPs are taking part to play a vital role since system design does its computation using those IP sub modules only. For any system design there is a need to use bus architecture. Without bus architecture none other system design can does data or control signal transfer. Based on only the data transfer delay, speed of the overall system gets determined. So, in High level synthesis designs, so as to achieve greater speed fast accessibility of data and faster transfer of data through the buses is very important. So, if the speed is achieved by minimizing the delay, in the delay of the architecture of the architecture of system and architecture of bus will surely lead to a better performance in the speed of the design of the high speed system. So, various architectures are realized, implemented and their delay is being compared. In modern VLSI system due to rapid switching of internal signals, power dissipation is high. So, we use optimized choice of path using the bus along with BRAM to reduce power dissipation as well as delay. The fundamental building blocks in many of

the system design is in the form of logic core or IP core units which should be integrated and interconnected by the bus architecture. It provides the easy way of design so as to reduce the design time in digital system to function faster and also to perform faster calculations. One of the major problems with this architecture is integration and scheduling of operation. According to Moore's law, design size will be halved in every 18 months.

This paper is organized as follows. Section - I says about the Introduction. Section - II deals about System Design using Micro Blaze. Section - III deals with BRAM and Memory. Section -IV says about Literature Survey with Existing Architecture and comparison with the proposed Architecture. Section - V deals with experimental results. Section - VI brief about Conclusion and Discussion over the future work.

II. SYSTEM DESIGN USING MICROBLAZE

In terms of its instruction set architecture MicroBlaze is similar to the RISC based DLX architecture. The MicroBlaze can issue a new instruction every cycle, maintaining single-cycle throughput under most circumstances.

The MicroBlaze has a versatile interconnect system to support a variety of embedded applications. MicroBlaze's primary I/O bus, the Core Connect PLB bus, is a traditional system-memory mapped transaction bus with master/slave capability. A newer version of the MicroBlaze supported in both Spartan-6 and Virtex-6 implementations, as well as the 7-series, supports the AXI specification. The majority of vendor-supplied and third-party IP interface to PLB directly (or through a PLB to OPB Bridge). For access to local – memory (FPGA BRAM), MicroBlaze uses a dedicated LMB bus, which reduces loading on the other buses. User-defined coprocessors are supported through a dedicated FIFO-style connection called FSL (Fast Simplex Link). The coprocessors interface can accelerate computationally intensive algorithms by offloading parts or the entirety of the computation to a user-designed hardware module.

Many aspects of the Micro Blaze can be user configured: cache size, pipeline depth(3-stage or 5-stge), embedded peripherals, memory management unit and bus-interfaces can be customized. The area-optimized version of Micro Blaze, which uses a 3-stage pipeline, sacrifices clock frequency for reduced logic area. The performance- optimized version expands the execution pipeline to 5 stages, allowing top speeds of 210 MHz (on Virtex-5 FPGA family). Also, key processor instructions which are rarely used but more expensive to implement in hardware can be selectively added/removed (i.e. multiply, divide and floating point operations). The customization enables a developer to make the appropriate design trade-offs for a specific set off host hardware paging and protection, such as the Linux kernel. Otherwise it is limited to operating systems with a simplified protection and virtual memory model, e.g. FreeRTOS or Linux without MMU support. Micro Blaze's overall throughput is substantially less than a comparable hardened CPU core (such as the PowerPC440 in the Virtex-5)

EDK

Xilinx's EDK (Embedded Development Kit) is the development package for building Micro Blaze (and PowerPC) embedded processor systems in Xilinx FPGAs. Hosted in the Eclipse IDE, the project manager consists of two separate environments: XPS and SDK.

Designers use XPS (Xilinx Platform Studio) to configure and build the hardware specification of their embedded system (processor core, memory controller, i/o peripherals, etc.) The XPS converts the designer's platform specification into a synthesizable RTL description (Verilog or VHDL), and writes a set of scripts to automate

the implementation of the embedded system (from RTL to the bit stream-file). For the Micro Blaze core the EDK normally generates an encrypted (non human-readable) net lists, but the processor description (written in VHDL) can be purchased from Xilinx.

The SDK handles the software that will execute on the embedded system. Powered by the GNU tool chain (GNU Compiler Collection, GNU Debugger), the SDK enables programmer to write, compile and debug c/c++ applications for their embedded system. Xilinx includes a cycle-accurate instruction set simulator (ISS), giving programmers the choice of testing their software in simulation or using a suitable FPGA-board to download and execute on the actual system.

Purchasers of EDK or ISE Design Suite Embedded Edition (IDS) are granted a perpetual license to use Micro Blaze in Xilinx FPGAs with no recurring royalties. The license does not grant the right to use Micro Blaze outside of Xilinx devices, which must be negotiated directly with Xilinx. Alternative compilers and development tools have been made available from Altium but an EDK Installation and license is still required. Micro Blaze became the first soft-CPU architecture to be merged into the mainline Linux kernel source tree. This work was performed by Michal Simek and supported by Petal oglx and Xilinx. As of September 2009, MicroBlaze GNU tools support is also being contributed to the free software Foundations mainline repositories. Support for Micro Blaze is included in GCC releases starting with version 4.6a.

III – BRAM AND MEMORY

The architecture plays a vital role in the design of any digital system. Novelty in how to add a bank of Block RAM to the On-Chip Peripheral Bus (OPB) within EDK/XPS. Block RAM comes in the form of 512X32 bit blocks of SRAM cells embedded within the Virtex-II Pro FPGA. It has very fast access time approx. 1 clock cycle for writes and 2 clock cycles for read.

The basic steps in adding BRAM to the OPB are: Add an OPB BRAM controller. This controller has two port connections: 1 OPB bus connection, 2. Memory connection (PORT A). Add a BRAM memory block to the system; This IP core has 2 port connections, 1. Memory Port A, Memory Port B, Connect the OPB BRAM controller to the OPB bus. Connect the OPB bus connection to the OPB bus. Connect the OPB BRAM controller to the BRAM memory block. Connect the controller's memory connection to one of the ports on the BRAM block (either PORT A or PORT B). Set the memory size and base address for the BRAM controller. This effectively sets the memory range automatically.(from MicroBlaze Guide)

AXI

Disclosed is a method of processing a read/write request conforming to the PLB bus protocol and a bus bridge from PLB bus to AXI bus, the method comprising of receiving the read/write request conforming to the PLB bus protocol without waiting for an acknowledgement of successful execution of a previous read/write request conforming to the PLB bus protocol; buffering the read/write request conforming to the PLB bus protocol, buffering the read/write request conforming to the PLB bus protocol.

Buffering the read/write request conforming to the PLB bus protocol. Mapping the buffered read/write request conforming to the PLB bus protocol to a read/write request conforming to a AXI bus protocol. Outputting the mapped read/write request conforming to the AXI bus protocol. The method of above claim, wherein buffering the read/write request conforming to the PLB bus protocol is full or not. And if the buffer is full, waiting until the buffer has become free spaces in the buffer to read/write the request. The method of claim1,

wherein mapping the buffered read/write request conforming to the PLB bus protocol to a read/write request conforming to a AXI bus protocol. Judging whether the received read/write request conforming to the PLB bus protocol is a read request or a write request.

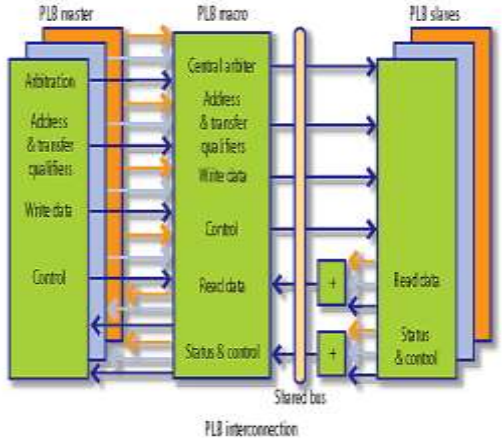


Fig.1 PLB interconnection

If the received read/write request conforming to the PLB bus protocol is a write request, storing a start address and end address of the write request as a start address and end address of a valid write request. The method of claim 3, wherein after outputting the mapped read/write request conforming to the AXI bus protocol further comprising of, if an acknowledgement of successful execution of the valid write request is received, updating the stored start address.

In Fig.1 it was clearly expressed that the way how Processor Local Bus is making connect between the MicroBlaze with peripherals and memory.Fig.2 ellobrates the same. But while using Multiprocessor system, one processor is working as a master and others as slave. The interconnect between the master and slave is also channelized by the bus PLB.

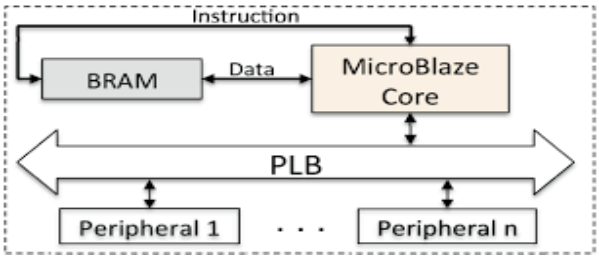


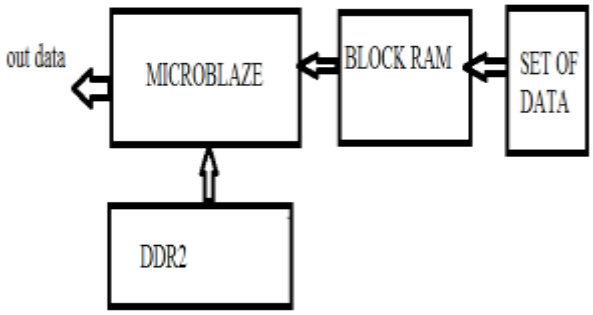
Fig.2 MicroBlaze with PLB and BRAM

In this paper the following buses are discussed to result in optimum delay for system design. They are PLB, OPB, AXI. In order to achieve the high speed and low power, there is a demand in the design of ICs for system design applications.(AXI reference guide)

IV- LITERATURE SURVEY AND PROPOSED ARCHITECTURE

LITERATURE SURVEY

The work proposes this architecture which consists of accessing a set of data stored was retrieved for utility of comparison using look up table and using BRAM with DDR2 results with the reduced interconnect delay. This makes many IP core to communicate with one another based on a bus which was interconnecting the architecture. The proposed architecture four core MicroBlaze soft-core processors are interconnected using PLB /AXI bus. Among the four processors one is designed as a master , the rest were designed as slave processors



And that a bus should being the only one device writing into it, although it can have many devices reading from it. Since many devices always produce output (such as registers) and these devices are hooked to a bus, and a need of way to control what gets on to the bus and doesn't. BRAM with DDR2 through AXI may provide high speed interconnect in system design.

V-EXPERIMENTAL RESULTS

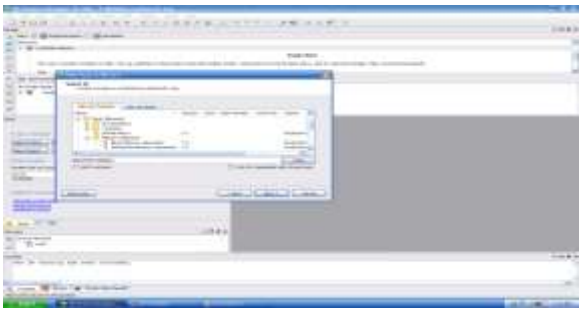


Fig.3. Utility of BRAM from tool- diagram

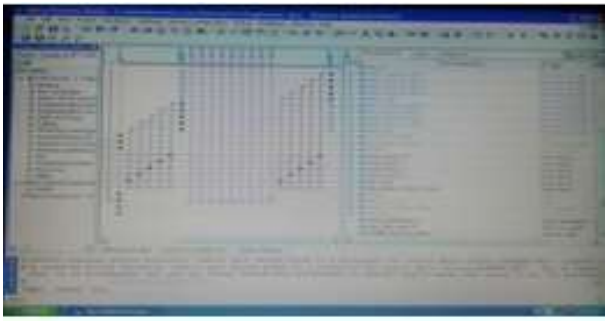


Fig4. System design using MicroBlaze-4 core

Fig5. Utility of DDR2 IP core

In the proposed work an improved architecture for the system design using Micro Blaze with look-up table, BRAM and DDR2 is being arrived and implemented by using the target device as Spartan-3E and Vertex-6 which was provided an optimized path delay for the design of the architecture. Even though various sub-modules are used in the construction of architecture as peripherals (IP), it provides a 4% less in delay compared to other sub-module architecture with look-up table. By analysis, it was concluded that the reduced path delay was obtained for architecture. Due to this it will find an applications in high speed system design like system on-chip and Network on chip.

Fig.4. BRAM IP core from IP List.

III. CONCLUSION AND DISCUSSION

Although this work is focused towards the minimization path delay using BRAM in the architecture of system design, the same can be achieved by implementing the design using some other sub module i.e IP core or IP along with DDR2 and with DCM, which can be controlled by routing algorithm which would be realized in C++.

REFERENCES

1. Nitesh Chandrakant, G.U.Kharat, "FPGA Realization of Multi-port Memory Controller for Communication with DDR2 Memory" International Journal of Emerging Research in Management & Technology

ISSN:2278-9359 Vlume-4, Issue-6, June 2015.

2. Adarian Cosoroaba Marketing Manager Xilinx, "Implementing Low-Cost DDR2 Interfaces with Spartan-3A FPGAs Xilinx provides complete memory interface solutions to help you get to market faster - MEMORY INTERFACES".
3. Elvira Olaso, Kazuo Ikeda, Francis J.Eng, Lieming Xu, Li-Hsien Wang, "DDR2 receptor promotes MMP-2–mediated proliferation and invasion by hepatic stellate cells" The Journal of Clinical Investigation Volume 108| Number 9 1369, November 2001.
4. by Adrian Cosoroaba Marketing Manager Xilinx, Inc.
5. M.Rajendra, N.Suresh Babu, "Speed and Area optimized Design of DDR3 SDRAM (Double Data Rate3 Synchronously Dynamic RAM) Controller for Digital TV Decoders"International Journal of Engineering Trends and Technology Volume6,ISSN:2231-5381, Page 204 Number 4- Dec 2013.
6. Nishanthi G, Yasha Jothi M Shirur, Ramudu B, "Functional Coverage for Low Power DDR2 Memory Controller in UVM"International Journal of Computer Applications Technology and Research Volume 3– Issue 5, 292 - 295, ISSN: 2319–8656, 2014.
7. Vijaykumar,R, K.Karunavathi, VijayPrakash.V "Design of Low Power Double Data Rate 3 Memory Controller with AXI compliant" International Journal of Engineering and Advanced Technology ISSN: 2249 – 8958, Volume-1, Issue-5, June 2012.
8. Andrey Malkov, Dmitry Vasiounin, Oleg Semenov, "A Review of PVT Compensation Circuits for Advanced CMOS Technologies" Circuits and Systems, 2011, 2, 162-169 July 2011.
9. Ms.Seema Sinha, Md.Tariq Anwar, "Design and Verification of DDR3 Memory Controller" International Journal of Advanced Technology in EngineeringandScience,Volume No.02, Issue No. 05, ISSN (online): 2348 – 7550, May2014.
10. Wim Vanderbau whede, Chalamalasetti and M.Margala, "Throughtput Analysis for a High Performance FPGA Accelerated Real Time Search Application", Hindawi Publishing Corporation, International Journal of Reconfigurable Computing Article ID 507173, 2012.
11. Shabaa Aqueel, Kavita Khare."Design and FPGA Implementation of DDR3 SDRAM Conrtroller for High Performance", International journal of Computer Science & Information Technology (IJCSIT) Vol 3,DOI: 10.5121/ijcsit.2011.3408101, No 4, August 2011.
12. Sonam Verma, Akanksha Srivastava,"Comparisons among DDR SDRAM, DDR2 SDRAM and DDR3 SDRAM", International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, ISSN (Print): 2320 – 3765, Vol. 4, Issue 7, July 2015.
13. Ajay Kumar, V.Sulochana Verma, Mohali,"Simulative Investigations of DDR2(SDRAM) Model in HDL", International Journal of Advanced Technology& Engineering Research(IJATER), ISSN No. 2250-3536, Vol.2, Issue 2, page N. 150, March 2012.
14. Sequential operations in digital picture processing Journal of ACM 13(4), 471-494, October 1966.
15. JeanPierre Derutin, Romuald Aufrere and Roland Chapui,"Parallel embedded processor Architecture for FPGA-based image processing using parallel

software skeletons" EURASIP Journal on Advances in Sigal Processing, 2013.
16. Faroudja Abid, Nourna Izeboudjen, Dalila Lazib,"SoCs Based openRISC and MicroBlaze softprocessors Comparison study cases:Audio implementation and Network Implementaton based SoCs", Internatonal Journal of Embedded Systems and Applications, Vol.4, No.1, March 2014.
17. S.Chari"FPGA based complex test pattern generatio for high speed fault diagnosisin memory blocks",International Journal of Advancements in Research & Technology, Volume 2, Issue 10, October-2013, ISSN 2278-7763.
18. Huy Nguyen, "RAPID Prototyping".
19. Kevin D.Jones, Victor Konrad and Dejan Nickovic,"Analog Property Checkers: A DDR2bCase Study".

BIOGRAPHY

Dr.C.Vennila B.E., M.E.(VLSI Systems), Ph.D. is a Professor in the Department of Electronics and Communication Engineering, Saranathan College of Engineering, Trichy. Her area of research interest is towards reconfigurable architecture, cognitive radio and VLSI design. vennila.nitt@gmail.com

S.Subathradevi B.E., M.Tech. (VLSI Systems) is an Assistant Professor in the Department of Electronics and Communication Engineering, Anna University- BIT Campus, Trichy. Her area of research interest is towards VLSI design, Digital System design and CAD VLSI. saminathan.sruthi@gmail.com

AXI, the third generation of AMBA interface defined in the AMBA 3 specification, is targeted at high performance, high clock frequency system designs and includes features that make it suitable for high speed sub-micrometer interconnect:

- separate address/control and data phases
- support for unaligned data transfers using byte strobes
- burst based transactions with only start address issued
- issuing of multiple outstanding addresses with out of order responses
- easy addition of register stages to provide timing closure.