International Journal of Emerging Engineering Research and Technology
Volume 2, Issue 2, May 2014, PP 155-159

Design and Verification Point-to-Point Architecture of WISHBONE Bus for System-on-Chip

Chandrala Brijesh A.[1], Mahesh T. Kolte[2]

[1]Department of electronics and telecommunication, MIT College of Engineering, Pune, India
[2]Department of electronics and telecommunication, MIT College of Engineering, Pune, India

Abstract: *SOC design is an integration of multi million transistors in a single chip for time to market and reducing the cost of the design. This paper presents point-to-point Architecture of WISHBONE bus. Main aim of the WISHBONE bus Architectures are to transfer the data from master to slave in different application without redesign of IP core. And that one will increase the productivity with reduction in design time. Standard interface bus protocol should do the plug and play.*

Keywords: *SOC, WISHBONE, Point-to-Point, Shared Bus interconnection, Data Flow interface architecture, Crossbar switch interconnection.*

1. INTRODUCTION

Recent advancement in technology allows integration of logic functions of multimillion transistors into a single chip. To keep pace with the levels of integration, design engineers have developed new methodologies and techniques to manage the increased complexity in these large chips [1]. System-on-Chip (SOC) design is proposed as an extended methodology to this problem where intellectual property (IP) cores of embedded processors, interface blocks, analog blocks and memory blocks are combined on a single chip for specific applications [2].

The WISHBONE System-on-Chip (SoC) Interconnection Architecture for Portable IP Cores is a flexible design methodology for use with semiconductor IP cores. Its purpose is to reduce design time by alleviating SOC integration problems. This is achieved by creating a common interface between IP cores. This improves the reliability and portability of the system, and that one will give product faster to the end user.

Previously, IP cores used non-standard interconnection schemes that made them difficult to integrate. That can't be used in different application. So, this required the creation of custom logic to connect each of the cores together. With the use of standard interconnection scheme, the cores may be integrated more easily and quickly by the end user. This specification may be used for firm core or hard core and soft core IP. Since hard cores and firm are generally conceived as soft cores.

The evolution of shrinking process technologies and increasing design sizes [3][4] has encouraged the manufacturers to undertake and do this kind of multiple component integration on a single chip. The innovative use of the key technology of SoC design (IP reuse) is quite helpful in meeting the above requirements. The advantage of adopting an efficient IP reuse strategy implies that some system modules can be directly implemented by using IP core without designing all modules conventionally. IP reuse design method can effectively improve the design efficiency, reduce market risk and research costs and shorten the development cycle [5].

This specification does not require the use of specific hardware or development tools. Furthermore, it is fully compliant with virtually all logic synthesis tools. However, the examples presented in the specification do use the VHDL language.

The WISHBONE architects were strongly influenced by three factors. First, there was a need for a common interface specification to facilitate structured design methodologies on large project teams. Second, there was a need for a good, reliable SOC integration solution. Third, they were impressed by the traditional system integration solutions afforded by microcomputer buses such as PCI bus and VMEbus.[6]

The rest of this paper is compiled as follows. A brief background of WISHBONE interface basics is discussed in section 2. Proposed system architecture is presented in section 3. Verification of result is presented in section 4. And conclusion is drawn in section 5.

2. WISHBONE BASICS

In this section background of WISHBONE bus interface and its specifications explained. WISHBONE utilizes "Master" and "Slave" architectures which are connected to each other through an interface called "Intercon". Master is an IP core that initiates the data transaction to the Slave IP core. Master starts transaction by providing an address and control signal to Slave. Slave is responds to the data transaction with the Master with the specified address range. The Intercon is the medium consists of wires and logics that help in data transfer between Master and Slave. The interconnection can be described using hardware description languages as VHDL and Verilog, and the system integrator can modify the interconnection according to the requirement of the design. It makes WISHBONE interface different from traditional microcomputer buses. WISHBONE interface supports variable interconnection. Master and Slave interface may use four types of interconnections such as, point to point, dataflow, shared bus and crossbar switch interconnection [4][6].

The point-to-point interconnection is the simplest one that allows a single Master interface to connect to a slave interface. The dataflow interconnection is needed for sequential data processing. In the shared bus interconnection two or more Masters can be connected with one or more Slaves. An arbiter is used to allow the master to gain access to the shared data bus. Crossbar switch interconnection also allows two or more WISHBONE masters to access two or more slaves at the same time. More than one master can use the interconnection as long as two masters don't access the same slave at the same time.

WISHBONE supports all the popular data transfer bus protocols such as block Read/Write, single Read/Write and Read- Modify-Write (RMW). Big Endian and Little Endian data ordering are also supported by WISHBONE.[6]

Other bus protocols available in market are AMBA, OPB, and Core Connect. WISHBONE offers almost free royalty, hence reducing the overall cost of the system design. WISHBONE Intercon can be designed to operate over an infinite frequency range. This is called as variable time specification [6-8]. The speed of the operation is only limited by the technology of the integrated circuits.

3. PROPOSED SYSTEM ARCHITECTURE

Point-to-Point Interconnection

The point-to-point interconnection is the simplest way to connect two WISHBONE IPcores together. As shown in Fig. 1, the point-to-point interconnection allows a single MASTER interface to connect to a single SLAVE interface. For example, the MASTER interface could be on a microprocessor IP core, and the SLAVE interface could be on a serial I/O port.

Fig. 1 shows the architecture of system design using point to point interconnection that includes SYSCON, DMA and MEMORY Cores. DMA transfers data to memory and from the memory using block transfer cycles. These cores are available in the WISHBONE public domain library for VHDL [6][8].

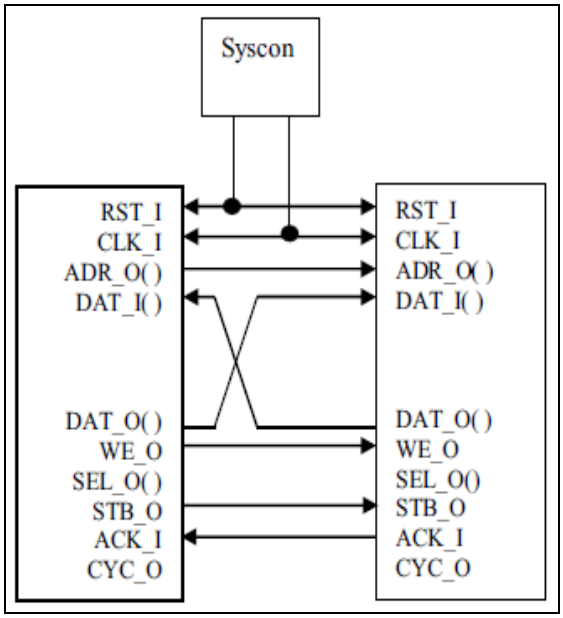

Fig.1. *Point to point interconnection*

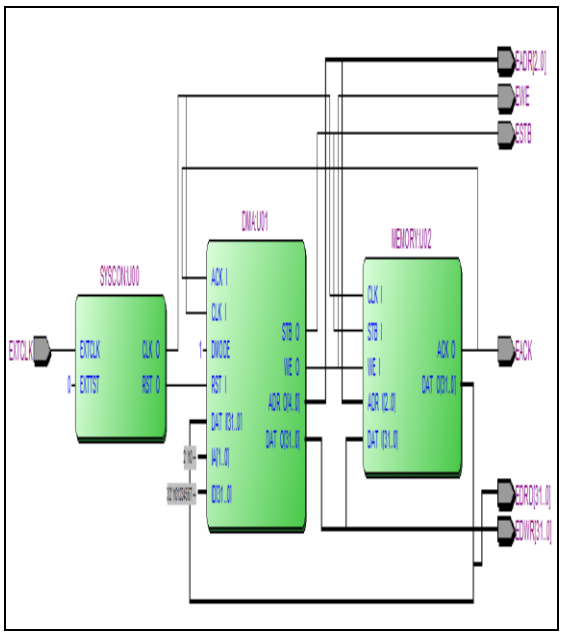

Fig.2. *RTL schematic of Point to Point interconnection*

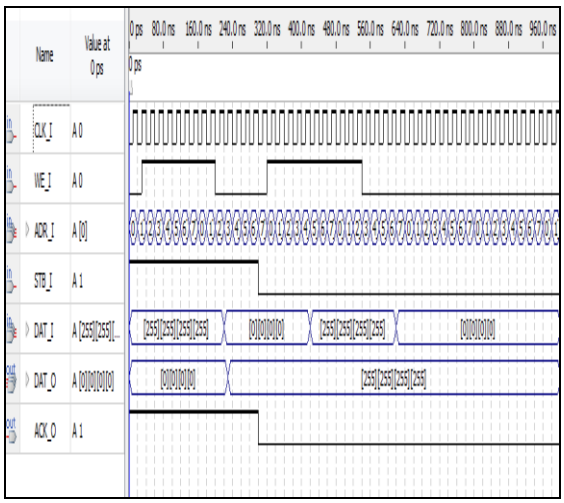

Fig.3 *Simulation result of Memory block*

The DMA core is a simple 32-bit DMA unit with a WISHBONE master interface. Two methods of data transfers are supported such as, single Read/Write cycles and block Read/Write cycles. This type of cycle is selected by a non-WISHBONE signal input DMODE. If DMODE input is negated, the DMA generates single read/write cycles, if it is asserted the DMA generates block read/write cycles. In block read/write mode, the DMA initiates eight phases of block write cycle, and then DMA generates a similar kind of block read cycle.

4. VERIFICATION RESULTS

The verification of the proposed system architectures were done using Quartus II.As shown in fig.2 RTL schematic of proposed architecture that includes SYSCON , DMA and MEMORY blocks.

The SYSCON also called as system controller is used to generate WISHBONE compatible clock and reset signals for the system. The clock output is fed directly from an external clock signal called EXTCLK. The reset generator produces a single reset signal RST in accordance with the WISHBONE reset timing.

The Memory is a simple, 8x32-bit size memory module with WISHBONE Slave interface designed for Quartus II. When data input is given to memory block. First it will check for WE and STB signals if both the signals are asserted then only it will write the data on memory otherwise memory's data remain unchanged. That simulation result is shown in fig.3.

DMA access the memory block and if strobe is asserted then it indicate slave is selected. A slave should give response other wishbone

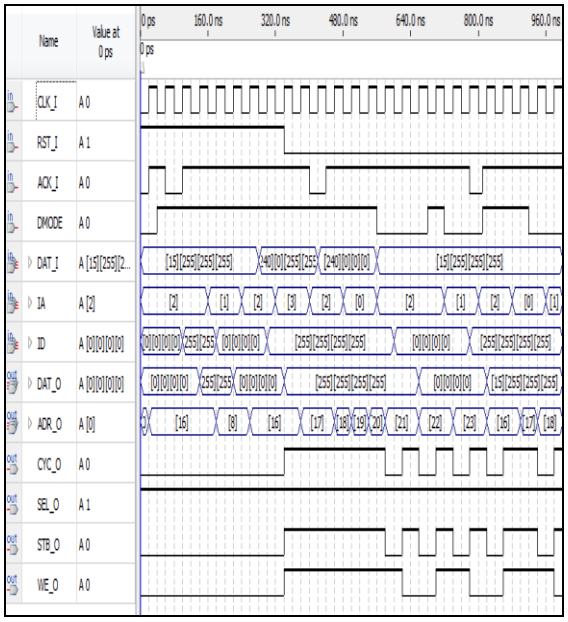

Fig.4 *Simulation result of DMA Block*

Signals only when strobe signal asserted. When reset is asserted then slave should not take input data. DMODE, WE and STB signals are asserted then only input data can be writing on memory block as shown in fig.4.

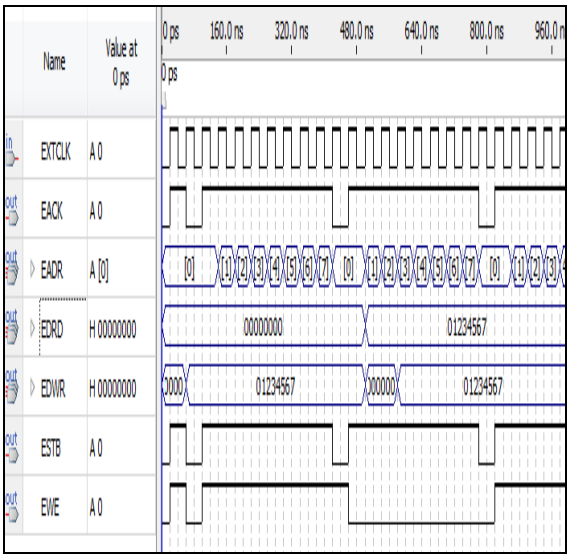

Fig. 5 *Simulation result of Point to Point Architecture*

The verification of the proposed system architecture was done using Quartus II. The point to point interconnection result is shown in Fig. 5 demonstrates that DMA is reading and writing 01234567 data continuously to the memory and from the memory.

Fig.6 *Verification waveform from FPGA kit using Signal tap analyzer*

In Quartus II with the help of Signal Tap analyzer implemented point-to-point Wishbone bus architecture on cyclone III FPGA kit which is shown in fig.6.

5. CONCLUSIONS

A 32-bit point-to-point system is designed and related issues were discussed. The verification of the design is done using Quartus II simulation and signal tap analyzer. The following conclusions are made from the above discussions: The total logic element requires for implementing point-to-point interconnection system is 74 and total memory bits are 256 bits. WISHBONE interface requires a very little logic overhead to implement the entire interface and gives rise to a highly portable system design that works with standard logic primitives available in most of the FPGA and ASIC devices. Hence, Low cost, portable and time to market SOC can be designed successfully using WISHBONE bus interface.

REFERENCES

[1] Resve Saleh and Steve Wilton, "System-on-chip: Reuse and integration," Proceedings of the IEEE, vol. 94, No.6, pp. 1050-1069, June 2006.

[2] A. K. Swain and K. K. Mahapatra, "Low Cost System on Chip design for Audio Processing, " in Proceedings of International MultiConference of Engineers and Computer Scientists -DATICSIMECS, 10, vol. II, pp. 1380-1385, Hong Kong, 17-19 March, 2010.

[3] [3] Milica Mitić and Mile Stojčv, "A Survey of Three System-on-Chip Buses: AMBA, CoreConnect and Wishbone," ICEST 2006.

[4] Chandrala Brijesh, M. T. kolte "A survey on WISHBONE Bus Architectures" National Conference on SITACS ,December 2013.

[5] Gajski, D.D; Wu, A.C-H; Chaiyakul, V.; Mori, S.; Nukiyama, T. and Bricaud, P. "Essential issues for IP reuse," Design Automation Conference 2000 Proceedings of the ASP-DAC, Asia and South Pacific, pp. 111-114, 2000.

[6] Richard Herveille, WISHBONE System-on-Chip (SoC) Interconnection Architecture for Portable IP Cores, rev. version: B4, 2010. By Open Cores Organization, p.7, 2010.

[7] Mohandeep Sharma, Dilip Kumar "Design and Synthesis of Wishbone Bus Dataflow Interface Architecture for SoC Integration" Annual IEEE India Conference (INDICON) pp.813 - 818 December 2012.

[8] Ayas Kanta Swain, KamalaKanta Mahapatra "Design and Verification of WISHBONE Bus Interface for System-on-Chip Integration" Annual IEEE India Conference (INDICON) pp.1 - 4 December 2010.