International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-3 Issue-1, June 2013

Design of Efficient SOC Bus Based on WISHBONE

Bharath. S. V, Ashwini. S. Shivannavar, M. Z. Kurian

Abstract*—In this paper wishbone bus is used to interconnect variety of devices. SOC designs are usually based on FPGA and ASIC which are widely used in embedded systems. In SOC design flexible interconnection between variety of devices is crucial to get maximum performance. Usually, in SOC design variety of devices such as high performance units like CPU, DMA, RAM ext., low performance devices like UART, GPIO's are connected to a single bus. The interconnecting bus runs at the speed of low speed device. An extra logic needs to be used in SOC to increase the performance of low speed devices, but this increases overall system power consumption. This paper proposes double bus architecture to interconnect the different devices according to the speed of the devices. High speed devices are connected to first level wishbone bus and low speed devices are connected to second level bus. This architecture shows that double bus design is feasible in low power SOC design.

***Index Terms*—Double bus, IP Core, SOC, Wishbone.**

I. INTRODUCTION

The rapid development in the field of mobile communication, digital signal processing (DSP) motivated the design engineer to integrate complex system of million transistors in a single chip. The integration of the transistors in a single chip greatly increases the performance of the system while reduction in system size. There is a considerable increase in the application front in recent time. Moore's law states that integration density gets doubled every two years so the complexity of the integrated system also increases by keeping the used chip area constant. In order to keep the phase with the level of integration available, engineers have developed new methodologies and techniques to manage the increased complexity in these large chips.

System-On-Chip (SOC) design is proposed as an extended methodology to this problem where IP cores of embedded processors, analog blocks, interface blocks and memory blocks are combined on a single chip targeting a specific application. These chips may have one or more processors on chip, a large amount of memory, co-processors, bus based architectures, I/O channels and peripherals. These chips integrates systems far more similar to the boards designed ten years ago that to the chips of even a few years ago. Figure 1 shows idea of SOC the a system of several ICs out of a printed circuit board is being integrated into a single chip while maintaining the overall structure the same.

Manuscript published on 30 June 2013.
*****Correspondence Author(s)
Mr. Bharath.S.V, Dept of E&C, SSIT, Tumkur, India.
Mrs. Ashwini.S.Shivannavar, Dept of E&C, SSIT, Tumkur, India.
Dr. M.Z.Kurian, Dept of E&C, SSIT, Tumkur, India.

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an open access article under the CC-BY-NC-ND license http://creativecommons.org/licenses/by-nc-nd/4.0/

Whole electronic system can be integrated in a chip with the idea of SOC design concept. On chip bus is used to connect the different IP modules. Interconnecting the IP cores has become the bottleneck of system performance.

Performance of the device can be increased by using efficient bus. By using a standard interface for connecting IP cores, the devices can be connected easily. AMBA from ARM, Avalon of Altra, IBM Core connect and wishbone of opencores are several on chip bus standards proposed by industry.

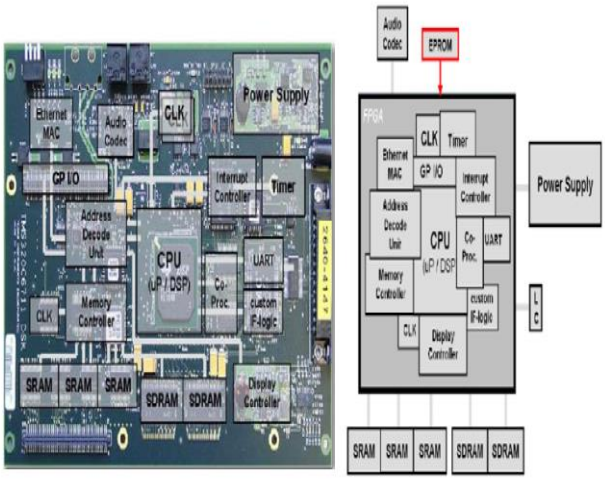

Fig. 1. Idea of SOC Design.

Wishbone bus is used to design SOC system because it is open source and wishbone-based IP's are freely provided by opencores organization.

II. WISHBONE BUS

The wishbone SOC Interconnection is a method for connecting IP cores together to form integrated circuits. Open core SOC design methodology utilizes wishbone bus interface to faster design reuse by alleviating system-on-chip integration problems. With use of this standardize bus interface it is much easier to connect the cores, and therefore custom SOC can be created easily. This way of SOC design improves the portability and reliability of the system, and results in faster time-to-market. The objective behind wishbone is to create a portable interface that supports both FPGA and ASIC that is independent of the semiconductor technology and wishbone interfaces should be independent of logic signaling levels. Another important reason is to create a flexible interconnection scheme that is independent of the type of IP core delivery (hard, soft IP) method. The next reasons are to have a standard interface that can be written using any hardware description language such as VHDL and VERILOG®.

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)
© Copyright: All rights reserved.

It supports a variety of bus transfer cycle in which the data transaction is independent of the application specific functions of the IP cores. It also supports different types of interconnection architectures with theoretically infinite range of operating frequency [12]. The final objective of wishbone bus is that it is absolutely free to use by developers without paying any fee for the cores available.

A. Wishbone basics

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.

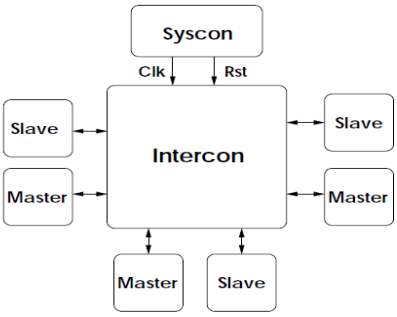

Fig. 2. WISHBONE Intercon system.

Master starts transaction providing an address and control signal to slave. Slave in turn responds to the data transaction with the master with the specified address range. The Intercon is the medium consists of wires and logics which help in data transfer between master and slave. The Intercon also requires a "syscon" module which generates wishbone reset and clock signal for the proper functioning of the system. Figure 2 shows the wishbone Intercon system which consists of masters and slaves and syscon modules. Wishbone Intercon can be designed to operate over an infinite frequency range. This is called as variable time specification. The speed of the operation is only limited by the technology of the integrated circuits. The interconnection can be described using hardware description languages like VHDL and Verilog®, and the system integrator can modify the interconnection according to the requirement of the design. Hence wishbone interface is different from traditional microcomputer buses such as PCI, VME bus and ISA bus.

B. Wishbone interconnections

Wishbone interface supports variable interconnection. It does not put any constraint on the type of interconnection the master/slave interface should use for communicating with each other as long as the wishbone specification signals and cycles are followed. Master and slave interface may use four types of interconnections such as, *Point to point, Dataflow, Shared bus and crossbar Switch interconnection* [12].

Two wishbone IP cores can be interconnected using point to point inter connection. This type of connection is the simplest of all. A single master is connected to single slave using point to point interconnection is shown in figure 3.

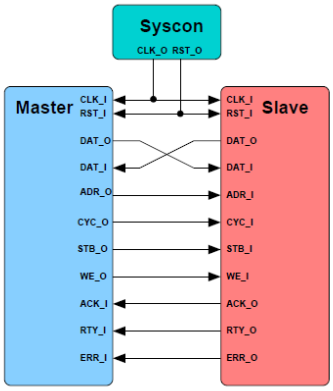

Fig. 3. point-to-point interconnection.

The data flow interconnection is used when data is processed in a sequential manner. As shown in Figure 4, each IP core in the data flow architecture has both a master and slave interface. Data flows from core to core. Sometimes this process is called pipelining.

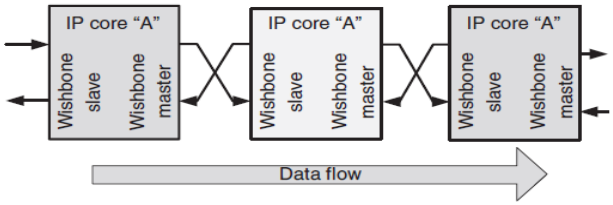

Fig. 4. Dataflow interconnection.

The shared bus interconnection is useful for connecting two or more masters with one or more slaves.

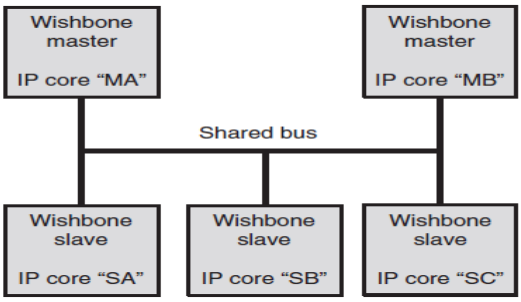

Fig. 5. Shared bus interconnection.

A block diagram is shown in Figure 5. In this topology a master initiates a bus cycle to a target slave. The target slave then participates in one or more bus cycles with the master.

The crossbar switch interconnection is used when connecting two or more wishbone masters together so that each can access two or more slaves. A block diagram is shown in Figure 6. In the crossbar interconnection, a master initiates an addressable bus cycle to a target slave. An arbiter (not shown in the diagram) determines when each master may gain access to the indicated slave. Unlike the shared bus interconnection, the crossbar switch allows more than one master to use the interconnection (as long as two masters don't access the same slave at the same time).

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)
© Copyright: All rights reserved.

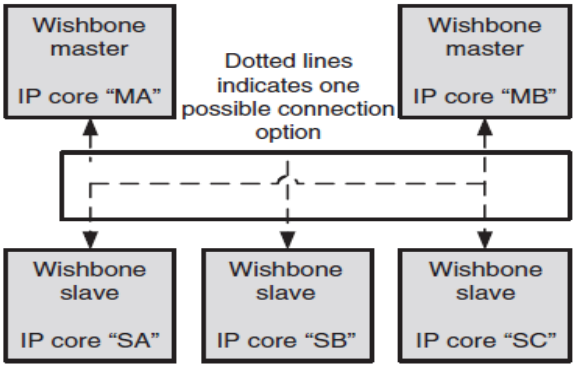

Fig. 6. Crossbar switch interconnection.

C. *Existing problem*

To design SOC, designers normally use interconnect IP as mentioned above, but interconnection between master and slave devices will made using the same bus. During a bus cycle, data transfer rate can be adjusted by master or the slave interface, as handshaking mechanism is used. I.e. the speed of the bus can be adjusted. This will results in a slower data transfer rate for all the wishbone bus cycles.

In systems comprising both high-speed and low-speed devices, to regulate the system frequency of high speed devices, corresponding logic needs to be increased, which will leads to two severe problems.

a) Enhancement of hardware area, as it requires increase in hardware logic resources.

b) Overall system power consumption will increase because of the system frequency enhancement.

In systems which are sensitive to power consumption, Low-power requirements cannot be met.

III. DESIGN OF DUAL ON CHIP BUS BASED ON WISHBONE

Double bus designed is implemented in this paper using wb_conmax and wb shared bus. High speed devices are connected to wb_conmax bus and low speed devices are connected to wb slave bus. Figure 7 shows the basic structure.

Wb_conmax IP is directly adopted in our design provided by opencores organization. Wb shared bus handles address decoding on the basis of wb_conmax. The devices interconnect through the standard wishbone bus signal, unlike the AMBA which consist of AHB and APB.

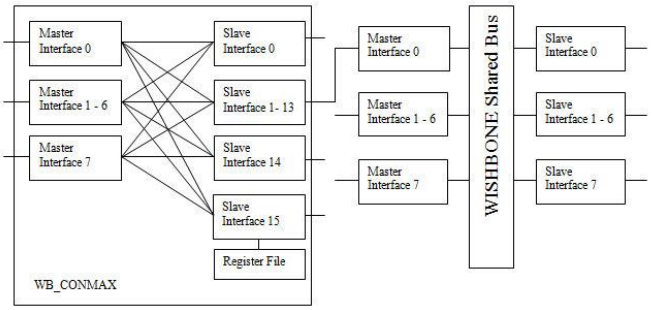

Fig. 7. Basic structure of the double WISHBONE bus.

A. *Master interface*

External wishbone master can connect to an wishbone slave using master interface. Appropriate slave interface is selected by master interface based on wishbone address bits [MSB:MSB-3].

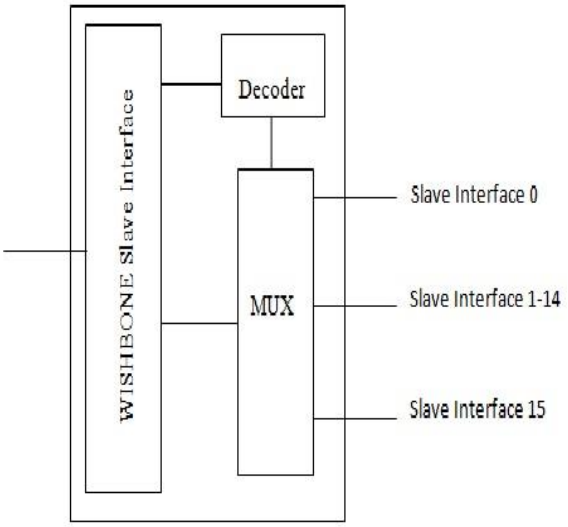

Fig. 8. Master interface.

B. *Slave interface*

External wishbone slave can connect to a wishbone master using master interface. Appropriate master interface is selected by slave interface based on internal arbitration.

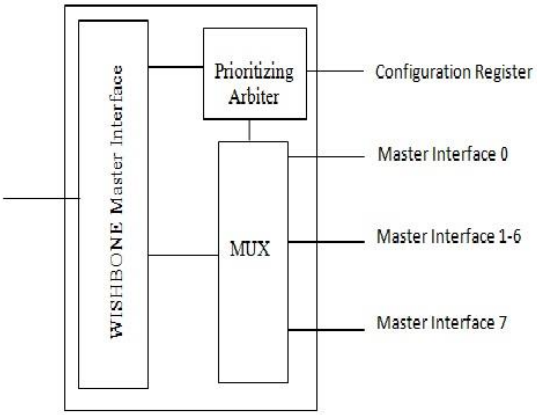

Fig. 9. Slave interface.

IV. IMPLEMENTATION RESULTS

Double wishbone bus is implemented using wb_conmax and wb shared bus. A master can communicate to the slave according to the MSB bits of address. Xilinx design suit is used to analyze the two different SOC system. Resource utilization results of single and double bus is tabulated. Resource utilization of double bus is more compared to single bus.

Table 1 the analysis of SOC of the two different architecture

		Single bus SOC	Double bus SOC
Design Utilization Summary	Number of Slices	6806	6982
	Number of Slice Flip Flops	1259	1942
	Number of Input LUTs	12816	13154

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)
© Copyright: All rights reserved.

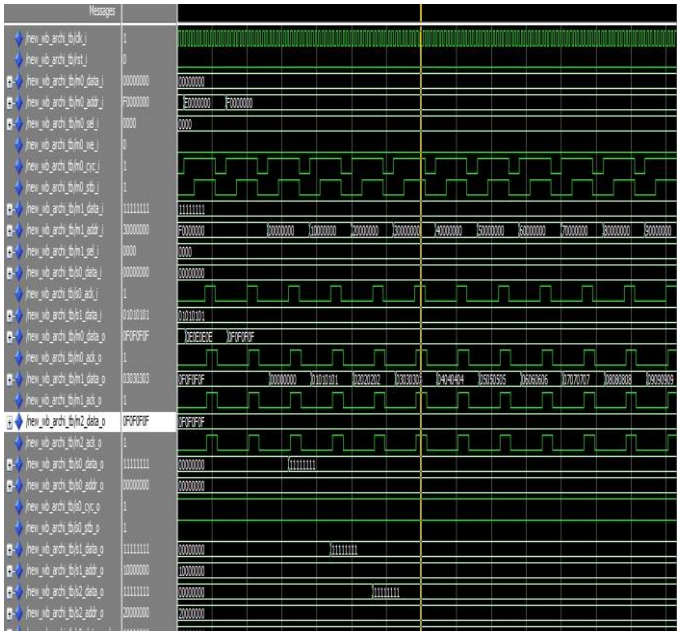

Fig. 10. *Simulation result.*

V. ADVANTAGES

The data transfer rate of high speed devices like CPU and memory will increase. Eliminating the corresponding logic to regulate the system frequency for some low-speed devices decreases the overall system power consumption.

VI. CONCLUSION

Wb_conmax and wb shared bus is used to design double wishbone bus. Wb_conmax IP is used in our design released by opencores organization. The two level bus is linked with two different type of IP cores, "design-reuse" concept can be improved from this architecture and also it gives better reduction of power consumption.

REFERENCES

1. Rudolf Usselmen. (2001, January) "Opencores SOC Bus Review" [Available]. http://www.opencores.org
2. Richard Herveille. "*WISHBONE System-on-Chip(SOC) Interconnection Architecture for Portable IP Cores*".Revision: B. 3, pp:4- 32,September 2002.
3. Raul Fajardo. (2010, September). *"MInimal OpenRISC System on Chip, Rev.1.1"* [Online]. Available: http://www.opencores.org
4. Rudolf Usselmann. (2001, October). "*WISHBONE Conmax IP Core*", Rev.1.1 [Online]. Available: http://www.opencores.org/project
5. Wayne Wolf, "*FPGA-Based System Design*", Prentice Hall, 2005.
6. Hartwig Jeschke, "*Efficiency measures for SOC concepts*", Journal of System Architecture, vol 54, pp1039-1045, April 2008.1
7. Ayas Kanta swain, Kamalakanta Mahapatra, "*Design and verification of WISHBONE bus interface for System-On-Chip integration",* Annual IEEE India Conference (INDICON), 389-1-4244-9073-8/10, 2010.
8. Rajendra Prasad, Dhanabal R, *"Low Power ASIC Design for Automation in Various Industrial Applications",* (IJAEST) INTERNATIONAL JOURNAL OF ADVANCED ENGINEERING SCIENCES AND TECHNOLOGIES Vol No. 6, Issue No. 1, 144 – 149, 2011.
9. Resve Seleh, Steve Wilton, Shahriar Mirabbasi, et al, *"System-onchip: Reuse and integration"*, Procedings of the IEEE,Vol 94(6), pp. 1050-1069, June 2006.
10. Wade Peterson, *"An introduction to WISHBONE: A chip-level microcomputer bus",* Reprinted from VMEbus Systems / February 2004.
11. S.Titri, N.Izebdjen, L.Sahli, D.Lazib, F.Louiz, *"Open Cores based System on Chip Platform for Telecommunication Applications: VOIP"*, IEEE conference 2007
12. "*Wishbone Specification*", Rev B3 [Online]. Available: www.opencores.org/downloads/wbspec_b3.pdf
13. Henry Chang, Merrill Hunt, Larry Cooke, *"Surviving the SOC Revolution",* Pub.Date: July 2008, Publisher: Springer-Verlag New York, LLC, ISBN: 0792386795
14. Daniel Akerlund, Master's thesis, *"Implementation of 2x2 NoC with Wishbone Interface",* Royal Institute of Technology (KTH), Sweden, Nov-2005.

AUTHOR PROFILE

Bharath.S.V currently pursuing his Post graduate degree in Digital Electronics from Sri Siddhartha Institute of Technology, he received his Bachelor degree in Electronics and Communication Engineering from Visvesvaraya Technological University, Belgaum, Karnataka, India. His research interests are in the areas of VLSI Design.

Ashwini S Shivannavar working as a Asst. Professor in the department of electronics and communication, Sri Siddhartha Institute of Technplogy.Since five years . Her research interests are in the field of Digital electronics and Signal Processing. She has published and presented papers in various international journals and several national and international conferences.

Dr.M.Z.Kurian received his Bachelor from Bangalore University and Post graduate degree in Industrial Electronics from Mysore University, and Ph.D degree in Software Engineering from Dr.MGR University, Chennai, Tamil Nadu, India. He has more than 30 Years of Teaching in the field of Electronics & Communication Engineering. Published several papers in peer reviewed international journals including IEEE, and several conference papers.

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)
© Copyright: All rights reserved.