ISSN: 2319-5967
ISO 9001:2008 Certified
International Journal of Engineering Science and Innovative Technology (IJESIT)
Volume 3, Issue 4, July 2014

Design and Implementation of Arbiter schemes for SDRAM on FPGA

Priyanka C. Sankpal, Arun S. Tigadi, Dr. Hansraj Guhilot

Abstract*— Memories are the storage devices, which typically work with single processing system. Sometimes, multiple systems require accessing the same memory for a number of different processes. This is when the Arbiter comes into picture. Since, a shared memory is one of the fastest techniques for inter-process communication; the design of an arbiter plays a very important role as it acts as a control system for various processes and it also avoids data corruption. With this said, an arbiter is designed and implemented in this work. It is implemented using VHDL (hardware description language) in the Xilinx ISE (Integrated Software Environment) Design suite 14.2. The simulation results obtained verified the correct functioning of the arbiter.

***Index Terms*— Arbiter, FPGA, Round Robin, shared memory, VHDL, Xilinx 14.2.**

I. INTRODUCTION

Random-access memory (RAM) is a form of computer data storage. The two main forms of modern RAM are static RAM (SRAM) and dynamic RAM (DRAM). Both static and dynamic RAM is considered volatile, as their state is lost or reset when power is removed from the system. Synchronous dynamic random access memory (SDRAM) is dynamic random access memory (DRAM) that is synchronized with the system bus.

The memory controller is a digital circuit which manages the flow of data going to and from the main memory. Memory controllers contain the logic necessary to read and write to DRAM, and to "refresh" the DRAM. Without constant refreshes, DRAM will lose the data written to it as the capacitors leak their charge within a fraction of a second.

Even though the memory allocated for specific processes is normally isolated, processes sometimes need to be able to share information. Shared memory is one of the fastest techniques for inter-process communication.

A. Problem statement

Multiprocessor systems can solve problems many times faster than a single processor alone. However, these processors must communicate in order to efficiently divide and solve a problem. Fast volatile memory is designed for use with a single processing system. In order for multiprocessor systems to access a common resource such as a memory, there must be some facilitation between the processing systems and the memory, in order to avoid any data corruption or malfunctioning of the system. [1] This is where the arbiter comes into play. An arbiter can be considered as a controlling system, which can be incorporated in between the multiprocessor systems and the shared memory. The arbiter follows a set of rules to pass the communication between the various processing systems.

B. Objective of the work

Arbiters can be used for different purposes and in a variety of applications. However, in this case it will be implemented on a Field Programmable Gate Array (FPGA) between some processing systems and a shared memory. Here, each system can be considered to be different and perform different tasks. A memory arbiter must be designed in such a way that it should consider which system is granted access in order to fairly share the access.

ISSN: 2319-5967
ISO 9001:2008 Certified
International Journal of Engineering Science and Innovative Technology (IJESIT)
Volume 3, Issue 4, July 2014

II. LITERATURE SURVEY

A. Concept of Arbiter

The arbiters are an important piece of the scheduler design. It works on three elements of a process: Request, Grant and Accept. [2]

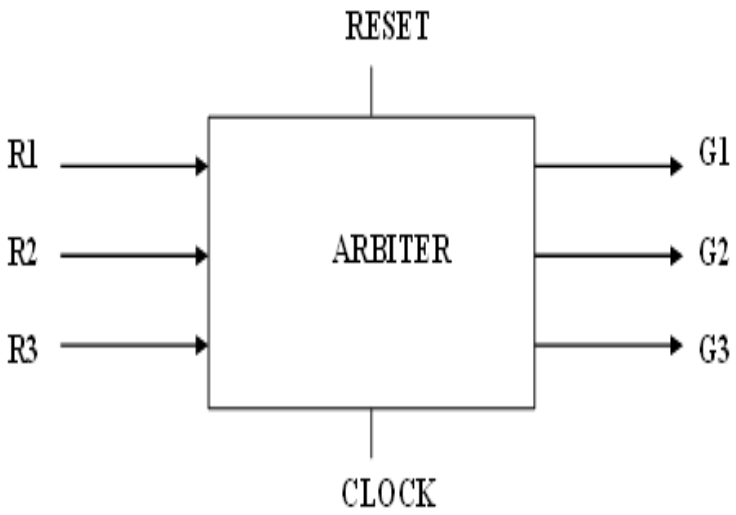

Fig 1: Block diagram of Synchronous Arbiter [2]

Step1 Request

Each unmatched input (R1, R2 & R3 in this case) request the arbiter to grant the access of the memory.

Step2 Grant

The arbiter grants the access depending upon the scheduling algorithm used. It chooses the one that appears next in a fixed priority or round-robin schedule starting from the highest priority element. The output notifies each input whether or not its request was granted.

Step 3 Accept

If an unmatched input receives a grant, it accepts the grant and does its work of data transfer or data access from the memory.

B. Round robin scheduling scheme

In this type of arbiter scheme, priority is cyclic in nature. It keeps on changing at every clock cycle. Here no process is considered the most important and so each process gets equal chance for getting their request completed. *Advantages* of Round Robin Arbiter are

- Maximum delay in granting client's request is C-1 where C is the number of clients/requests.
- Starving of clients does not take place.
- No user biasing is prevalent.

Disadvantages of Round Robin Arbiter are

- Request of prime importance can be missed to due to cyclic priority.
- Complexity of the design is higher than fixed priority. [3]

III. IMPLEMENTATION

This section describes the design of Arbiter. It has been divided into two parts: - In the first part, a single system interface to the memory is described and in the next part a four system arbiter design using round robin scheme is described.

ISSN: 2319-5967
ISO 9001:2008 Certified
International Journal of Engineering Science and Innovative Technology (IJESIT)
Volume 3, Issue 4, July 2014

A. Single System interface showing Read and Write operations

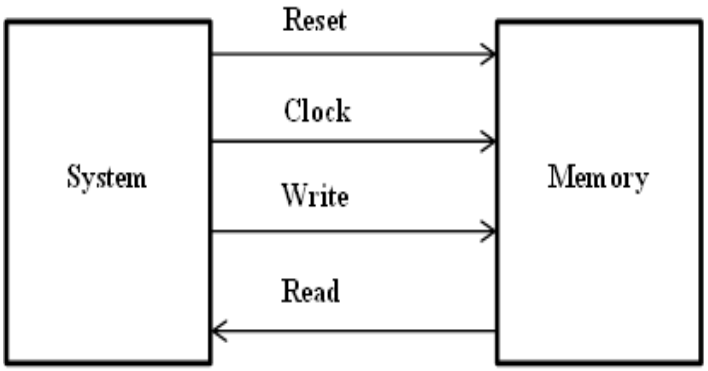

Fig. 2: Single system interface to memory

In this case, a single system is interfaced with the memory. Reset and Clock are the inputs and they should be enabled for any operation to take place. Write is the input to the memory and the data can be written into the memory only when the write is enabled. Read is the output from the memory and the data can be read only when read is enabled.

B. Four Systems Arbiter design using Round Robin scheduling algorithm

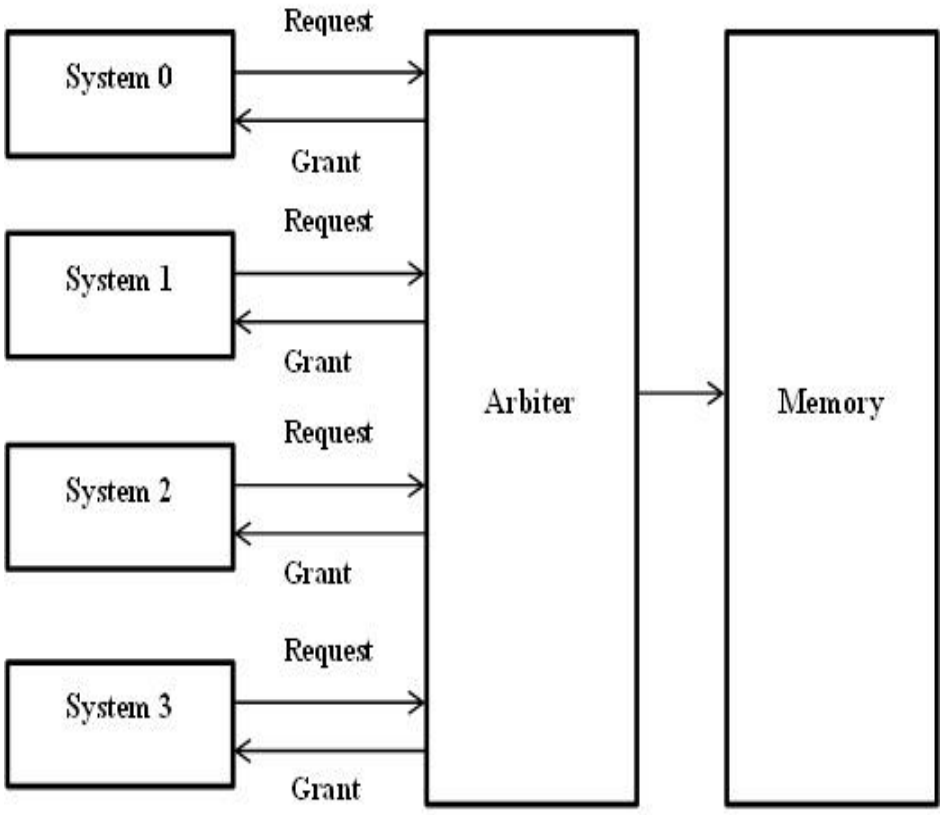

Fig. 3: Four systems interfacing with Arbiter.

In this case, the four systems, namely s0, s1, s2 and s3 want to access the memory. The Arbiter which is the control system has been designed in such a way that it follows the Round Robin scheduling algorithm to provide the access of the memory to the four systems. Each of the systems can access the memory for a particular duration of time, i.e. the scheduling is based upon fixed time slice for each system.

ISSN: 2319-5967
ISO 9001:2008 Certified
International Journal of Engineering Science and Innovative Technology (IJESIT)
Volume 3, Issue 4, July 2014

IV. RESULTS

A. Single system Read & Write operations

Fig. 4: Write operation

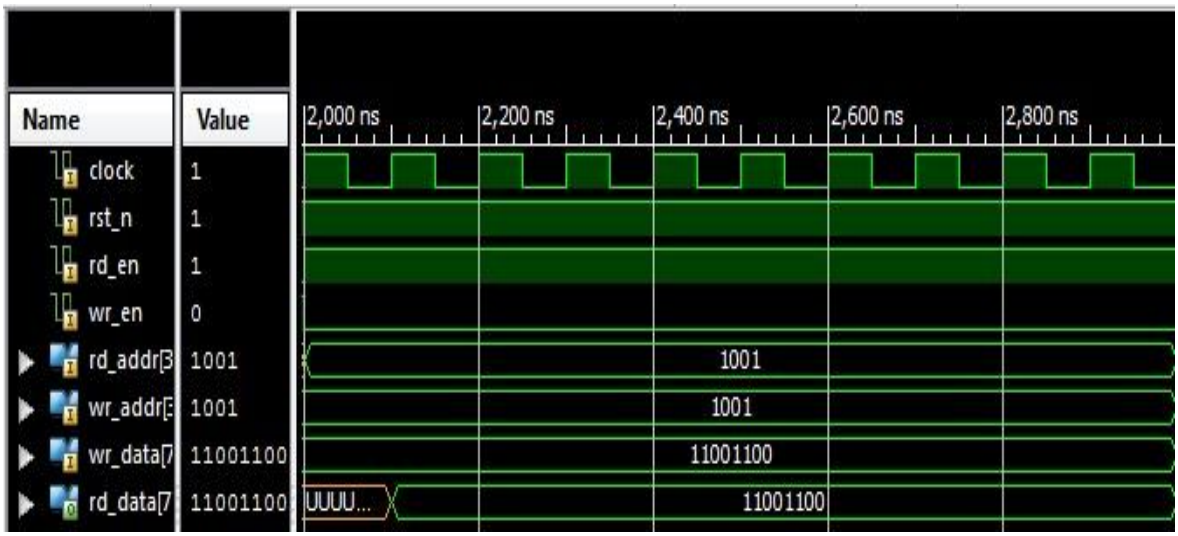

Fig. 5: Read operation

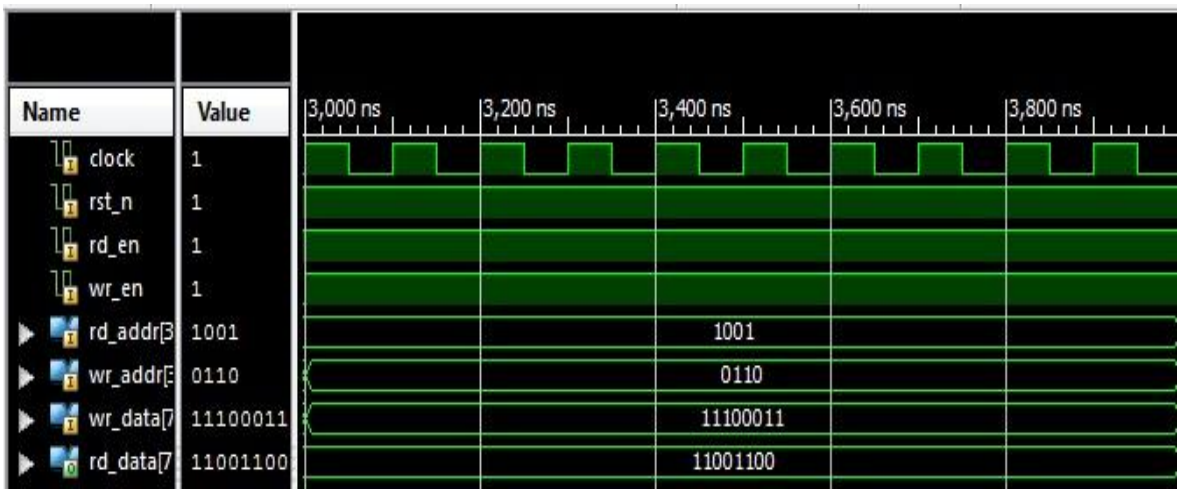

Fig. 6: Read and Write operation

ISSN: 2319-5967
ISO 9001:2008 Certified
International Journal of Engineering Science and Innovative Technology (IJESIT)
Volume 3, Issue 4, July 2014

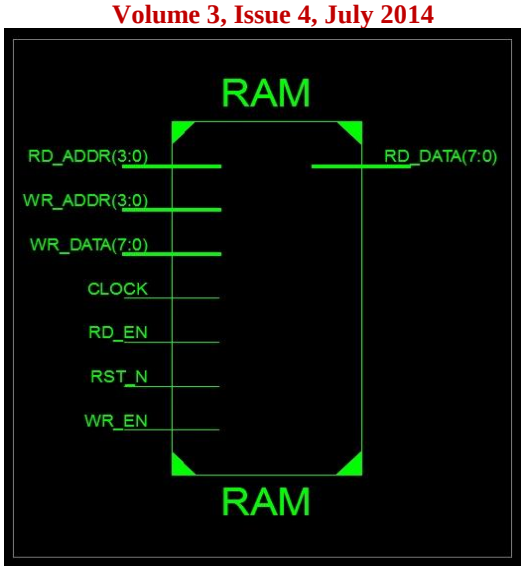

Fig. 7: RTL schematic of single system interfacing to RAM showing Top view

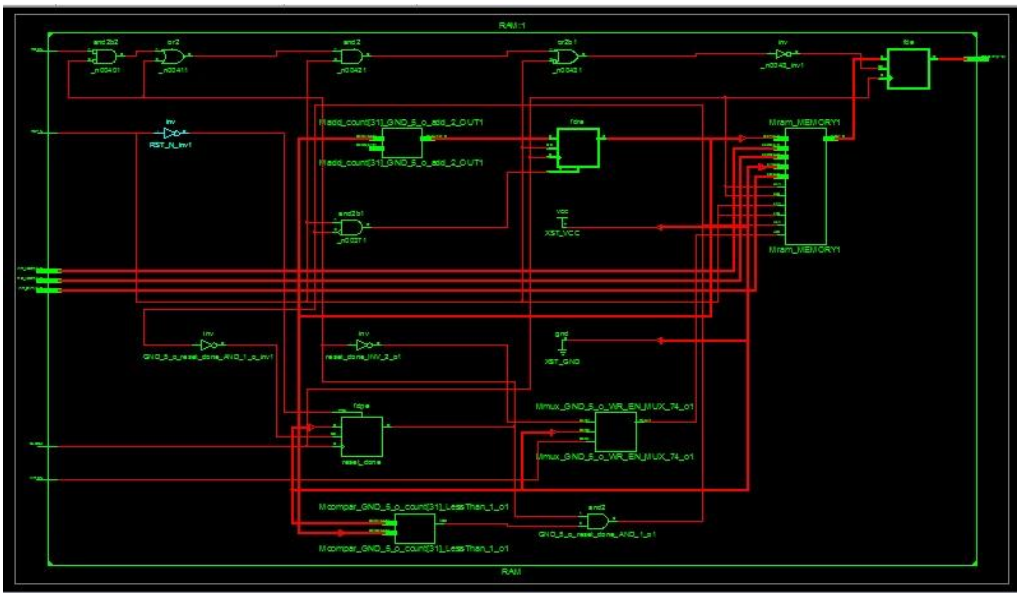

Fig. 8: RTL schematic of single system interfacing to RAM showing detailed view

B. Arbiter Design using Round Robin scheduling scheme with four systems/requestors interface

Fig. 9: Idle case

ISSN: 2319-5967
ISO 9001:2008 Certified
International Journal of Engineering Science and Innovative Technology (IJESIT)
Volume 3, Issue 4, July 2014

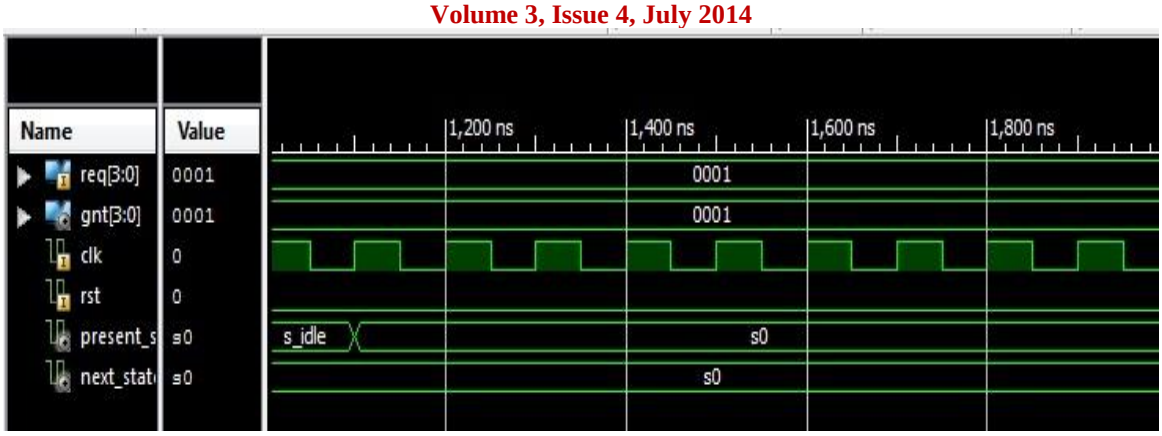

Fig. 10: Single system i.e. only s0 requests

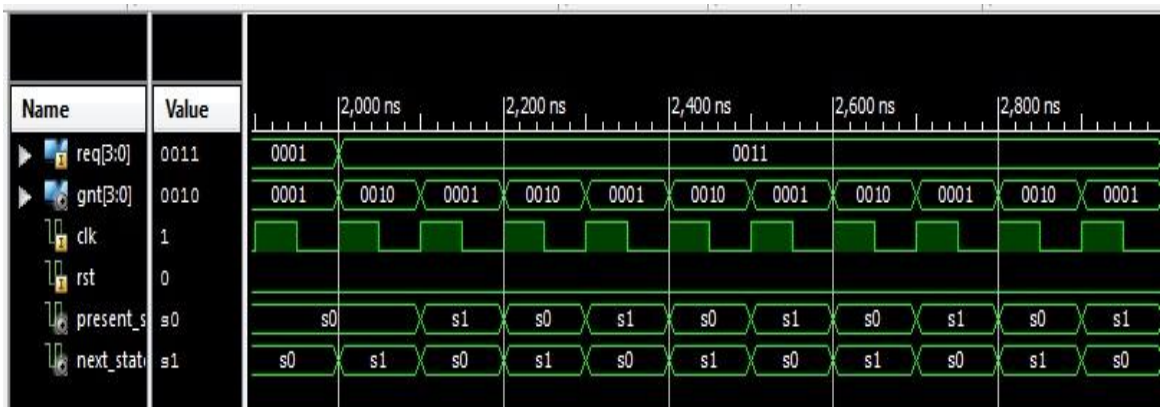

Fig. 11: Two systems i.e. s0 and s1 request

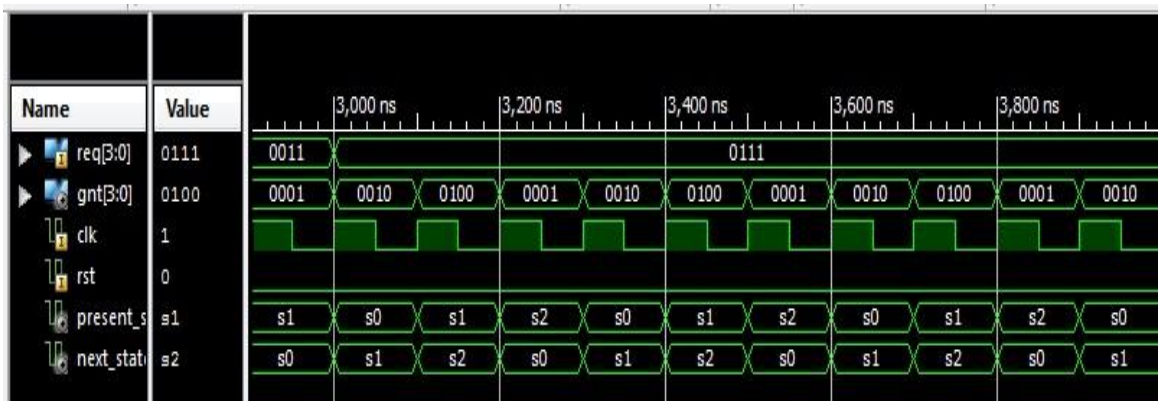

Fig. 12: Three systems i.e. s0, s1 and s2 request

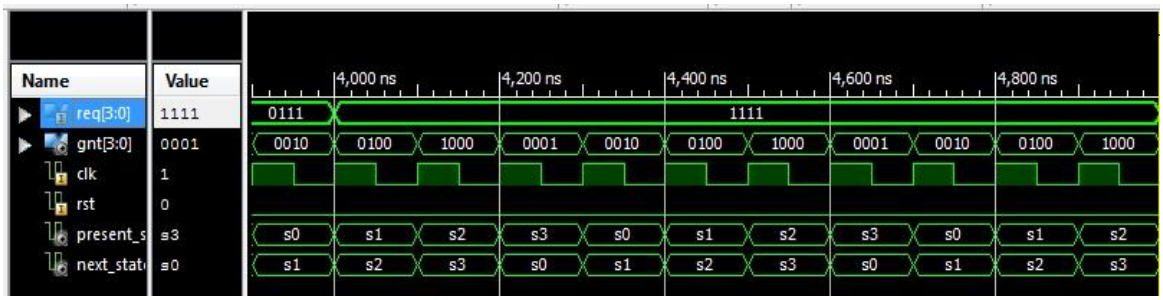

Fig. 13: All four i.e. s0, s1, s2, & s3 request at the same time

ISSN: 2319-5967
ISO 9001:2008 Certified
International Journal of Engineering Science and Innovative Technology (IJESIT)
Volume 3, Issue 4, July 2014

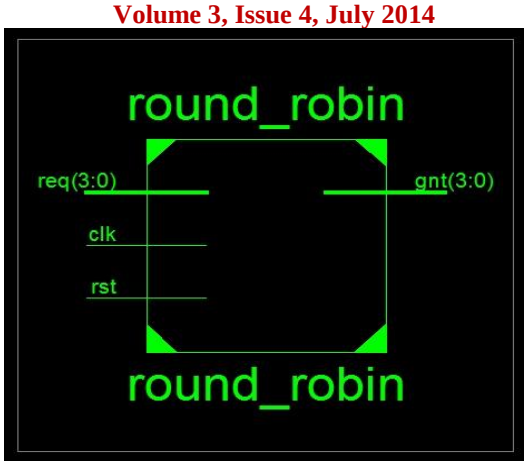

Fig. 14: RTL schematic of four requestors interfacing to Round Robin Arbiter showing Top view

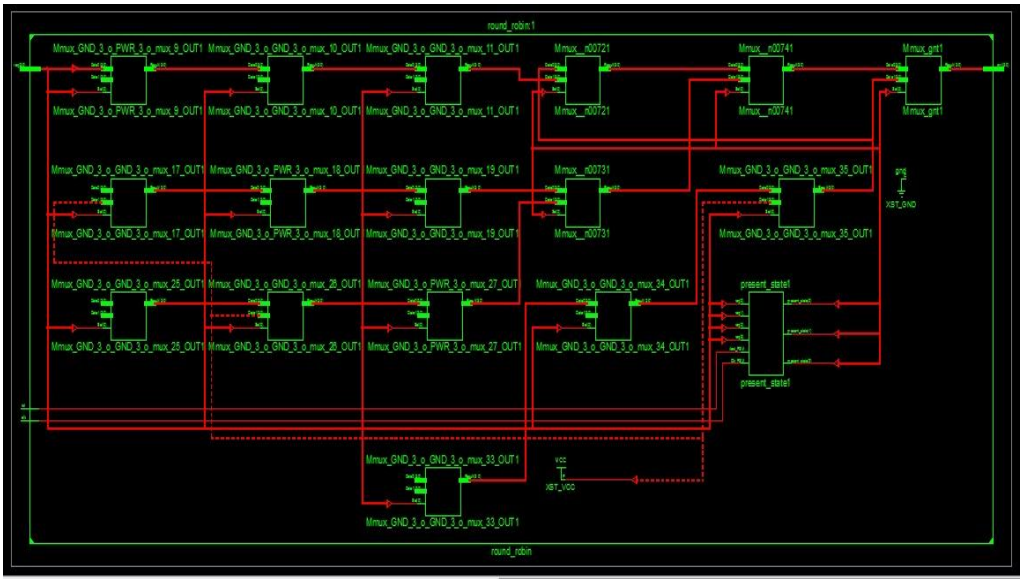

Fig. 15: RTL schematic of four requestors interfacing to Round Robin Arbiter showing detailed view

V. CONCLUSIONS AND FUTURE SCOPE

In this work, the basic structure of the arbiter was designed and implemented. This design can be extended for more number of systems and also can be implemented using various other scheduling algorithms. Different scheduling algorithms can be implemented, and the arbiter could be designed in such a way as to choose between any one of the scheduling schemes.

The arbiter has been designed in a generalized way, showing simple read and write operations of the data. It can be implemented considering the real time applications. The data in the form of images or video can be stored into or accessed from a shared memory by the multiprocessors.

ACKNOWLEDGMENT

First author, Ms. Priyanka C. Sankpal thanks Prof. Arun S. Tigadi and Dr. Hansraj Guhilot for their constant support and guidance. The author also thanks the institute KLE Dr. M.S. Sheshgiri College of Engineering and Technology for all the facilities provided. The Author also expresses sincere gratitude towards all the teaching and non-teaching faculties, dear friends and family for their support.

REFERENCES

[1] ArtenEsa and Bryan Myers, “Design of an Arbiter for DDR3 Memory”, WPI, Worcester, April 25, 2013.

ISSN: 2319-5967
ISO 9001:2008 Certified
International Journal of Engineering Science and Innovative Technology (IJESIT)
Volume 3, Issue 4, July 2014

[2] Tarun Kumar, Gauttam, Rekha Agrawal, Sandhya Sharma, "Arbiter Design Using Verilog for Switching to Communicate in Between Multiple Resources", International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-3, Issue-3, August 2013.

[3] Kedar Trivedi, Nandish Thaker, "HARD IP CORE OF MEMORY ARBITER" International Journal of Advanced Research in Computer and Communication Engineering Vol. 2, Issue 5, May 2013.

[4] M. Weber, "Arbiters: Design Ideas and Coding Styles," Synopsys Users Group, Boston, 2001.

[5] Peter Manoilov, Plamena Krivoshieva, "Shared Memory Design for Multicore Systems" International Scientific Conference Computer Science, 2008.

[6] Jonathan Chao, "Saturn: A Terabit packet Switch Using Dual Round-Robin", IEEE Communications Magazine, vol. 38, no. 12 December 2000, pp78-84.

AUTHOR BIOGRAPHY

Ms. Priyanka C. Sankpal received B.E. degree in Electrical and Electronics Engineering from KLS Gogte Institute of technology (KLSGIT), Belgaum, India in 2012. She is currently pursuing M.Tech in VLSI Design and Embedded systems from KLE Dr. M S Sheshgiri College of Engineering and Technology (KLECET) affiliated to Visvesvaraya Technological University (VTU), Belgaum. Her research interests are VLSI design and embedded systems, FPGA Design, Memory controllers and arbiters, renewable energy resources and smart grid technology.

Prof. Arun S. Tigadi is an Assistant Professor in the Dept. of E&C, K.L.E DR. M.S.Sheshgiri College of Engineering and Technology, Belgaum. He received his U.G Degree in E&C from S.D.M CET Dharwad in the year 2006 and P.G Degree in VLSI Design and Embedded systems from K.L.E CET Belgaum in the year 2008. His fields of interest are Low power VLSI design, FPGA Design, Memory controllers and arbiters, Real time system design, Operating systems. He published a paper entitled "*E-Governance in India*" in a National conference at PIMR Indore (MP) 11th and 12th September 2009, National IT Conference as "*Innovative IT practices for Organizational Excellence*" and another paper entitled "*Implementation of reconfigurable synchronous Ram controller on FPGA for flexible platforms*" International conference on communication VLSI design and signal processing 2013 held at SIT Tumkur.

Dr. Hansraj Guhilot held many academic and R&D positions over career span of 28 years, currently working as the Principal, K. C. College of engineering and management studies and research. He worked as Dean (R&D) and Professor of EC at KLE Dr. M. S. Sheshgiri College of Engineering and Technology, Belgaum, Karnataka. He has teaching experience spanning 28 years with a Ph.D. in Electronics , having thesis titled "*Design and Development of CMOS Mixed Mode Integrated Circuit for Chloroplast Measurement*" and Research work published in IEEE Sensors Journal. He is an IEEE Technical Paper Reviewer at IEEE International Conference on Recent Trends in Information, Telecommunication and Computing (ITC), 2010. He is a member of Entrepreneur Development Cell (EDC) in Visveswaraya Technological University (VTU), Belgaum. He is a subject expert in CMOS VLSI, Edusat Program, VTU, Belgaum. He has published 36 papers, delivered 10 invited technical talks and is awarded with one US Patent and nine international patents. Worked as Director (R&D), Paradigm Industries Inc. USA and Consultant for N&N Allied Energy Services Inc. USA.