International Journal of Advanced Information Science and Technology (IJAIST) ISSN: 2319:2682
Vol.4, No.6, June 2015 DOI:10.15693/ijaist/2015.v4i6.35-40

# A detailed study and implementation of efficient multi-ported memories for Xilinx FPGA device

***Arun S. Tigadi[1], Deepti .S.P. [2], Hansraj Guhilot[3]***

[1]*Department of Electronics & communication, K.L.E. Society's Dr. M.S.Sheshgiri college of Engineering and technology, Belagavi,*

[2]*Department of Electronics& communication, K.L.E. Society's Dr. M.S.Sheshgiri college of Engineering and technology, Belagavi*

[3]*K. C. College of Engineering and Management Studies and Research,,Thane,Maharashtra,India*

**ABSTARCT-** ***As the block RAMs included in the fabrication typically have only two ports, Multi-ported memories have become challenging to implement with FPGAs. If memory with more than two ports is required by any design, then it should therefore be built of logic elements or by multiple block RAMs combination. Here in this paper a new design is introduced called Live Value Table(LVT) that combines block RAMs into multi-ported memories with more number of read and write ports and compared to other conventional methods, this method provides the most significant operating frequency. Another method is also presented here which is based on XOR operation which gives multi-ported memories that uses less logic but with more RAM blocks than the LVT method.***

***Index terms – FPGA, memory, multi-port, LVT, XOR***

## I . INTRODUCTION

In recent years, for designing any application, the FPGA technology plays the most important role on the integrated circuit area in order to minimize the time to market and cost. Multi-ported memories have become challenging to implement with FPGAs as the block RAMs provided typically have only two ports. Here in this paper a thorough exploration of the design of FPGA based soft multi-ported memories is presented by evaluating conventional solutions to this problem and a new design is introduced here that combines block RAMs into multi-ported memories with more number of read and write ports called Live Value Table. Along with this LVT, other methods such as Replication, Banking and multi-pumping are designed and implemented.[1]

These methods are implemented in Xilinx integrated software environment (ISE) using verilog hardware description language.

## II .IMPLEMENTATION

### 2.1. Replication :

Replication is nothing but process of making a replica or copy of something. Here in this project it includes only one external write port with any number of external read ports. But the single write port is connected to one of the four ports of each replicated RAM. In this method , the memory can be accessed between ports as only one write port is routed to all the ports. Hence this technique does not support more than one write port. The design for this replication is shown in figure 1(a). Here 32*8 memory bank is considered. For each port the 32*8 bit does not double or does not get shared , instead it goes on replicating the same 32*8 bit memory simply requiring the large area space. This requiring the more area space is one of the disadvantage of this replication method.

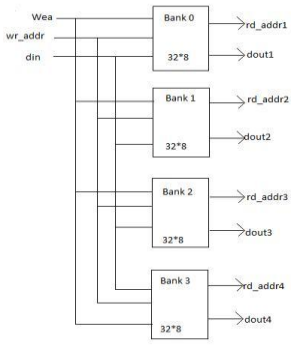



Figure 1(a) Replication

### 2.2. Banking :

Here in the banking the 32*8 memory location is divided among all multiple banks. As the four ports are considered, the 32*8 is divided into four 8*8. That is

each bank will have memory location of 8*8 as shown in the figure 1(b). These additional banks formed will support the additional read and write port. But like replication, this method cannot access memory between the memory banks. Only the corresponding memory bank is accessed by its particular read or write port. Hence this design do not support the sharing between the ports, this is the disadvantage of this method.

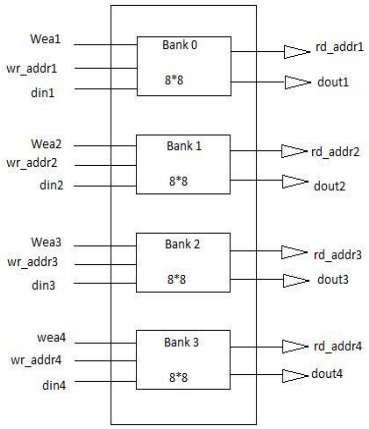



Figure 1(b) Banking

### 2.3. Multipumping :

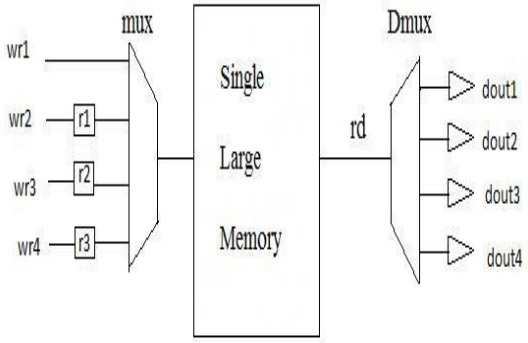



Figure 1(b) multipumping

Here the multipumping design requires multiplexers, registers to hold the address and data temporarily of the pending reads and writes. In this method there is no doubling or sharing of memory, there is only a single large memory. The write address are being written in memory, before that the write address are multiplexed using the select lines. The registers holds the temporary address and data of the pending ports. The address which are stored in memory are read using the select lines of the demux and read at the corresponding output. The drawback of this method is with increase in number of ports the external operating frequency get reduced.

### 2.4. Live Value Table :

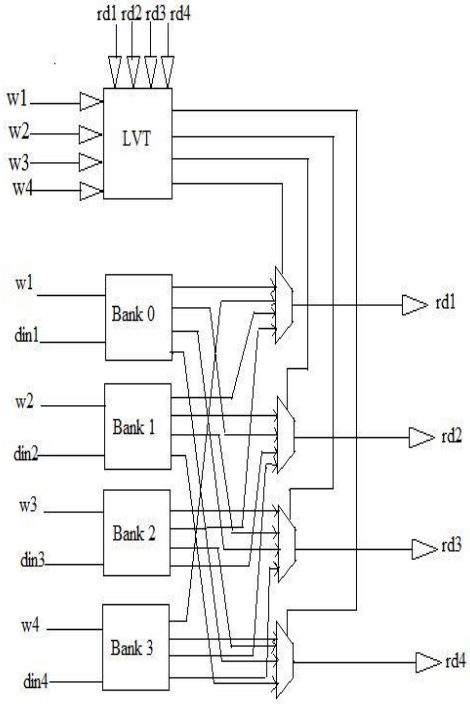



Figure 1(d) LVT

The Live Value Table is somewhat similar to replication. Writing is exactly like replication. But this method has a separate LVT module which stores the latest values. Whenever the particular write enable is one, its corresponding write address is written into its corresponding bank. And this value is updated in LVT and stores the recent values that is it stores the newly updated address along with its bank number. Multiplexers controls the read enables and the

corresponding read address are read by select lines of mux and are read at the corresponding output. The advantage of this method is that it can simultaneously do read and writes and is faster. The disadvantage is that it requires more area space.

### 2.5. XOR Based approach:

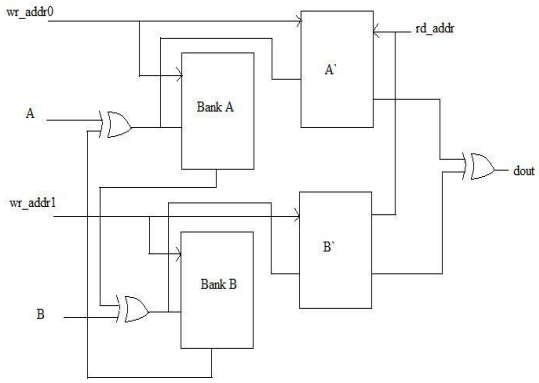
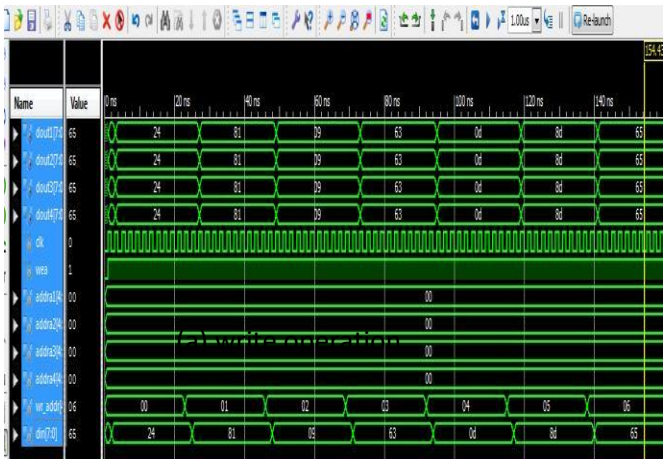


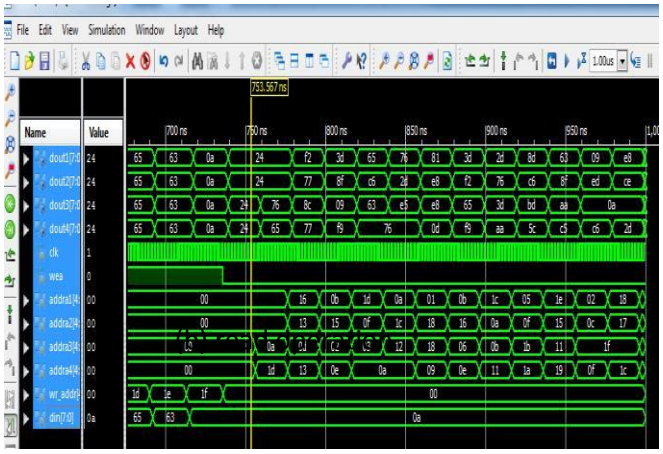

Figure 1(e) XOR

This method is mainly based on XOR and this is an alternative to the LVT based approach. This method is mainly designed to overcome the challenges faced by LVT method. As used in LVT, the need of LVT module to direct reads and output multiplexers are not used in this XOR method. Instead of that, this method makes use of the logic of read ports and reads the XOR values from the memory banks. This XOR is also faster as like LVT method. Here a simple memory 2W/1R is constructed as shown in figure(e). As illustrated in the figure, there are two BRAMs as the write ports each has its own bank. And to both the BRAMs each writes are copied that is the same value is written in all the BRAMs of the corresponding locations. The value A is stored by the Wo write port to some location, now this value A is XORed with the value of bank B of the corresponding location as of bank A. Similarly, the value B is stored by W1 write port to lower location of bank B, this value B is XORed with the value of the corresponding location of bank A. The main goal of this design for the read port is that it should only consist of the XOR values of the memory bank outputs. And this goal is achieved by storing the XORed values in their respective separate banks that is A` and B` as shown in the above design. And at the end, the stored values are XORed again. So that should result in recovering the value of A.

## III.SIMULATION RESULTS :

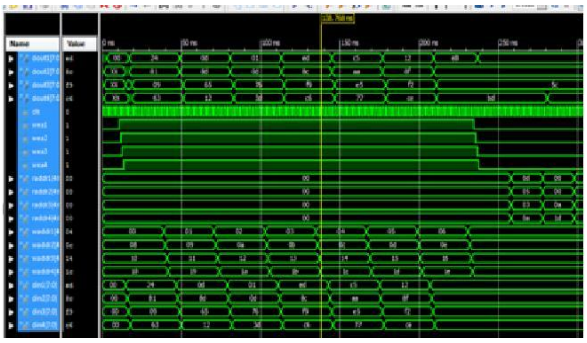
### 3.1. Replication :

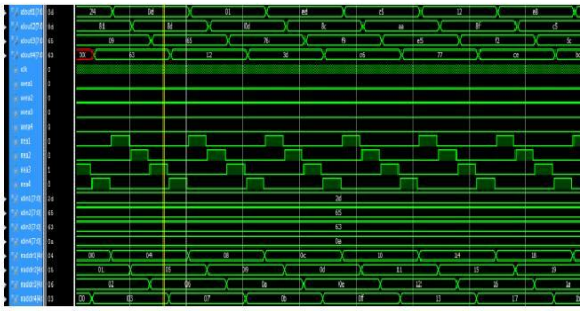


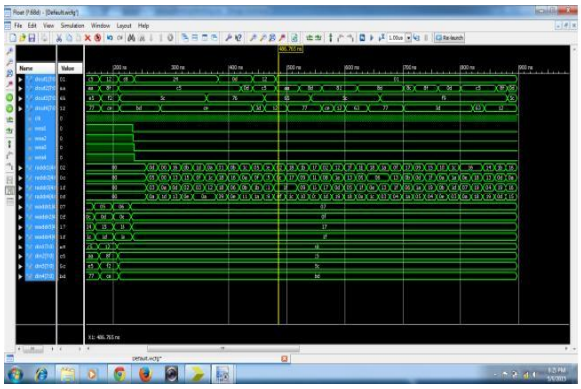


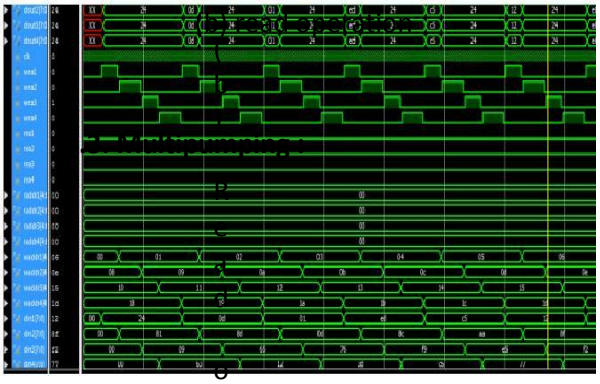
The above screenshots are the simulation results of the replication which includes write and read operations. When the write enable is 1, the data din is written in to write address as shown in figure. And when write enable becomes zero, the address are read in output i.e dout. The same read and write operations are shown for all methods below in the screenshots.

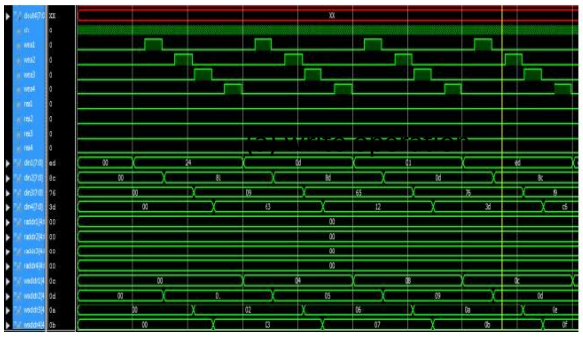
### 3.2. Banking:





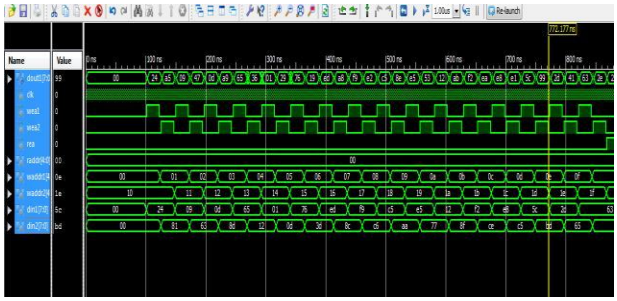






Live Valu

**.4. XOR**



(a) Write operation

(b) Read operation

The above screenshots are the read and write operations of the XOR. If we consider an example like the write address 01 has a data din1 as 09. This 09 is XORed with 01th location of data din2, i.e 81 as shown in screenshot. And in read operation the 01 read address should recover a value of data din1 i.e it should read as 09.

## IV. CONCLUSION

The efficient block RAMs are provided by FPGA systems but not with more than two ports. The conventional approaches like replication, banking, multipumping with large number of ports are either have inefficient area or slow. So the fastest method called Live Value Table has been introduced which coordinates read and write access such that this design behaves like a true multi-ported design.

## REFERENCES

[1] Implementing Multi-Port Memories in ProASICPLUSDevices.http://www.actel.com/document s/APA_MultiPort_AN.pdf, July 2003. Application Note AC176, Accessed Sept. 2009.

[2] Mercury Programmable Logic Device Family Data Sheet.http://www.altera.com/literature/ds/ dsmercury.pdf, Jan 2003. Version 2.2, Accessed Sept. 2009

[3] Stratix III Device Handbook Volume 1, Chapter 4: TriMatrix Embedded Memory Blocks in Stratix III Devices.http://www.altera.com/literature/hb/stx3/stx3 _siii5 1004.pdf,May 2008. Version 1.8, Accessed Sept. 2009.

[4].Advanced Synthesis Cookbook: A Design Guide for Stratix II, Stratix III, and Stratix IV Devices.http://www.altera.com/literature/manual/stx_ co okbook.pdf, July 2009. Version 5.0, Accessed Nov. 2009.

**Authors Profile**

**Arun S. Tigadi**, Assistant professor Department of E & C , K.L.E DR. M.S. Sheshgiri College of Engineering and Technology. Have a working experience of 6 years in the department of E & C. Received the B.E. 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. Fields of interest are Low power VLSI design, FPGA design, Memory controllers, arbiters, multiport memory design, Real time system design and operating systems. Published 7 international journal papers and presented 2 papers in international conferences. Life member of ISTE

**Deepti. S. P** received **the B.E.** degree in Electrical and Electronics engineering from the Bapuji Institute of Engineering and Technology, Davanagere, VTU university, Karnataka, India, in 2013. Currently doing **Mtech** final year in VLSI design and embedded systems in K.L.E CET Belgaum in the year 2015. Research interest includes multi-ported memories on FPGA.

**Dr. Hansraj Guhilot** held many academic and R&D positions over career span of 30 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. Email: hansraj.g@gmail.coms