International Research Journal of Education and Technology
Peer Reviewed Journal
ISSN 2581-7795

IMPLEMENTATION OF FIFO ROUTER DESIGN TO AVOID THE DATA LOSSES DURING TRANSMISSION

Dr .N. Srinivas, Associate Professor, Dept of ECE, Guru Nanak Institute of Technology, Hyderabad

P. Pradeep, Dept of ECE, Guru Nanak Institute of Technology, Hyderabad,

B. Mallesh, Dept of ECE, Guru Nanak Institute of Technology, Hyderabad

M. Premalatha, Associate Professor, Dept of ECE, Guru Nanak Institute of Technology, Hyderabad

K. Swetha, Dept of ECE, Guru Nanak Institute of Technology, Hyderabad,

ABSTRACT

This paper introduces FIFOs are used generally in electronic circuits for buffering and flux control. In attack form a FIFO primarily consists of a set of read and write pointers, storage and control sense. Storage may be SRAM, flip- movables , latches or any other suitable form of storage. For FIFO of nontrivial size a double- harborage SRAM is generally used where one harborage is used for notation and the other is used for reading. FIFO is used for synchronization purposes. It's constantly executed as a circular line, and thus has two pointers 1. Read Pointer/ Read Address Register 2. Write Pointer Write Address Register.

Read and write addresses are firstly both at the first memory position and the FIFO line is empty. FIFO Empty When read address register reaches to write address register, the FIFO triggers the Empty signal. FIFO FULL When write address register reaches to read address register, the FIFO triggers the FULL signal.

Keywords Routing, Router, Data Packets, Verilog, Xilinx VIVADO.

Overall, our paper presents a comprehensive result for avoid the data loss during transmission..

I.INTRODUCTION

Memories Memories are one of the most useful VLSI building blocks. Semiconductor memory is a digital electronic semiconductor device used for digital data storage.

Therefore stored data can be reused for realizing real time operations like routing a data through a router switch.

The router is a ” Network Router ” has a one input harborage from which the packet enters. It has three affair anchorages where the packet is driven out. Packet contains two corridor. They're address, and data. Packet range is 8 bits. Destination address(DA) of the packet is of 8 bits, among those 2 bits are for address and the remaining 6 bits are for data.

The switch drives the packet to separate anchorages grounded on this destination address of the packets. Each affair harborage has 2- bit unique harborage address.However, also switch drives the packet to the affair harborage, If the destination address of the packet matches the harborage address.

In conclusion, the 3port router was the proposed design. The proposed Router structure functionality is enforced in Verilog HDL and proven that this armature gave lower detention and high frequence, so speed of operation will be increases. In this paper the Xilinx ISE 14.7 EDA Tool is used for conflation and simulation. The data which can be shoot through the router is reached the destination with 3.935 ns quiescence and the frequence was 360.101 MHz.

II.EXISTING SYSTEM

Accessible, complex network of connected computer networks that uses the standard Internet Protocol(IP) to transmit data by dividing it into preface The Internet is a global, intimately lower units, called packets.

These packets pass through routers, that connect separate networks, to reach their destinations. Whenever a packet reaches a router, that router has to decide on the coming router on the path toward the destination of that packet. Indeed though it takes just a simple lookup operation in the ‘ router table’, it becomes significant when the appearance rate of packets is larger than the speed of a lookup operation. This is exactly the case in the Internet backbone where high- speed routers are employed to handle this situation.

The thing of this design is to establish an applicable model to address this issue and to perform an adaptive analysis of the performance of these high- speed routers grounded on the ‘ graciousness’ of the input packet sequence.

We say an input sequence is ‘ nicer’ than another input sequence if the former bone has smaller number of changes in packet destinations in successive packets of the sequence i.e. the total number of lookup operations demanded is smaller.

In Field Test for Permanent Faults in FIFO Buffers of NoC Routers Bibhas Ghoshal, Kanchan Manna, Santanu Chattopadhyay, and Indranil Sengupta This brief proposes an on- line transparent test fashion for discovery of idle hard faults which develop in first input first affair buffers of routers during field operation of NoC. Grecu, P. Pande, B. Wang, A. Ivanov, and R. Saleh, “ Methodologies and algorithms for testing switch- grounded NoC interconnects, ” In this paper, we present a new methodology for testing similar NoC infrastructures.

The proposed methodology offers a dicker between test time and on- chip tone- test coffers. The fault models used are specific to semiconductor technology.

check on the design styles of low power SRAM cell L Saranya This paper proposes arrival of movable electronics in our day moment life as made Power Optimization as one of the major challenges in the ultramodern VLSI technologies.

stationary arbitrary- access memory(SRAM) has been extensively used in the recent days due to its high performance in VLSI design ways which operates in the range of submicron or nano range.

III.PROPOSED SYSTEM

In the SRAM cell, the scaling of transistor will increase the stability of the cell at the time of read and write operation.

FIFOs are used commonly in electronic circuits for buffering and flow control. In hardware form a FIFO primarily consists of a set of read and write pointers, storage and control logic. Storage may be SRAM, flip-flops, latches or any other suitable form of storage. FIFO is used for synchronization purposes. It is often implemented as a circular queue, and thus has two pointers: 1. Read Pointer/Read Address Register 2. Write Pointer/Write Address Register.

FIFO Empty: When read address register reaches to write address register, the FIFO triggers the Empty signal.

FIFO FULL: When write address register reaches to read address register, the FIFO triggers the FULL signal.

In conclusion, the proposed system router structure functionality is implemented in Verilog HDL and proven that this architecture gave less delay and high frequency, so speed of operation will Increases

IV.METHODOLOGY

Block diagram of Router

The methodology Router is a Packet based protocol. Router drives the incoming packet which comes from the input port to output ports based on the address contained in the packet. The router has a one input port from which the packet enters. It has three output ports where the packet is driven out. The router has an active low synchronous input reset which resets the router.

Packet

Packet contains 3 parts. They are Header, payload and parity. Packet width is 8 bits and the length of the packet can be between 1 bytes to 63 bytes

Packet header contains two fields DA and length.

DA: Destination address of the packet is of 2 bits. The router drives the packet to respective ports based on this destination address of the packets. Each output port has 2-bit unique port address. If the destination address of the packet matches the port address, then router drives the packet to the output port. The address "3" is invalid.

Length: Length of the data is of 6 bits and from 1 to 63. It specifies the number of data bytes. A packet can have a minimum data size of 1 byte and a maximum size of 63 bytes.

If Length = 1, it means data length is 1bytes

If Length = 2, it means data length is 2bytes

If Length = 63, it means data length is 63bytes

Packet-Payload:

Data: Data should be in terms of bytes and can take anything.

Packet-Parity:

Parity: This field contains the security check of the packet. It should be a byte of even, bitwise parity, calculated over the header and data bytes of the packet

IMPLEMENTATION

Functionality: On reset router FSM(finite state machine) will be in DECODE ADDRESS State**.**

DECODE ADDRESS State:

- In this state if packet valid = 1, data in<3, and memory empty = 1 then circuit jumps to LOAD_FIRST_DATA.
- In DECODE ADDRESS State if packet valid =1, data in<3, and memory empty = 0 then circuit goes to WAIT_TILL_EMPTY.

WAIT_TILL_EMPTY State:

- In this state if memory empty = 0 then circuit remains in the same state i.e WAIT_TILL_EMPTY State.
- If memory empty = 1 then circuit goes to the LOAD_FIRST_DATA State.

LOAD_FIRST_DATA State:

- By default the circuit will be in LOAD_ DATA State.
- If memory full=0,and packet valid = 0,then the circuit goes to LOAD_PARITY state.
- If memory full = 1, then the circuit goes to MEMORY_FULL_ STATE.

LOAD_ PARITY State:

- If memory full = 0 then the circuit goes to CHECK_PARITY_ERROR.
- If memory full = 1 then the circuit goes to MEMORY_FULL_ STATE.

MEMORY_FULL_ STATE State:

- If memory full = 1 then the circuit remains in the same state i.e MEMORY_FULL_ STATE.
- If memory full = 0 then the circuit goes to the state called as i,e LOAD_AFTER_FULL STATE.

CHECK_PARITY_ERROR state:

- If the circuit reaches to unconditional criteria then the circuit goes to DECODE ADDRESS State.

LOAD_AFTER_FULL state:

- In this state if parity done = 0,and low packet valid = 1 then the circuit goes to a state called as LOAD_PARITY state.
- If parity done = 0,and low packet valid = 0 then the circuit goes to a state called LOAD_DATA stat e.

International Research Journal of Education and Technology
Peer Reviewed Journal
ISSN 2581-7795

V.FIGURES

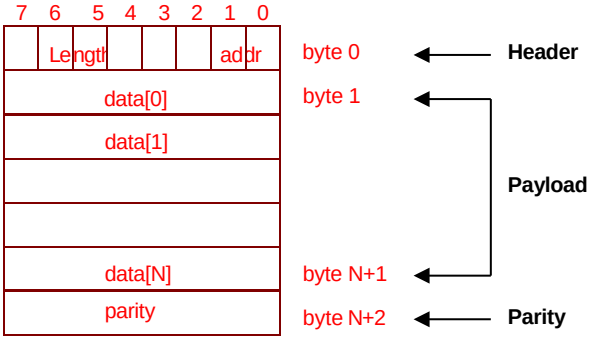


Fig 1. Data Packet Format

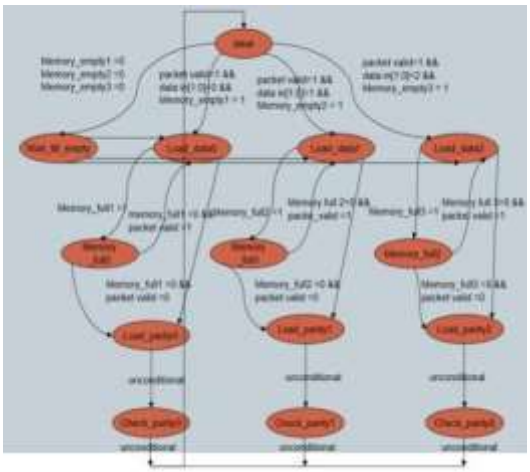


Fig 2. FSM State diagram

VI.BLOCK DIAGRAMS

Fig 3: Block Diagram of Router

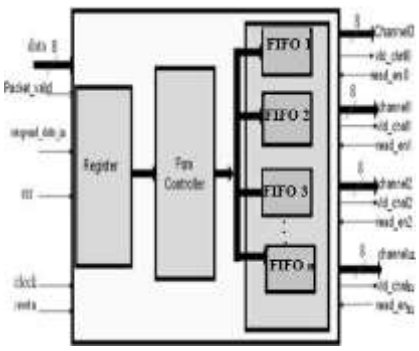


Fig 4: Internal Structure of Protocol

Applications

- Data packets transmission
- Streaming Services
- Telecommunications
- Message Queuing Systems.
- Electronic system in cars.
- Digital electronics control VCRs
- Transaction processing system, ATM

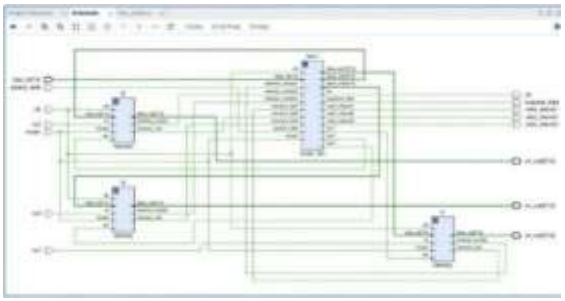
VII.DESCRIPTION OF SOFTWARE

SOFTWARE USED: Xilinx
Xilinx software is used by the VHDL/VERILOG designers for performing Synthesis operation. Any simulated code can be synthesized and configured on FPGA. Synthesis is the transformation of VHDL code into gate level net list. It is an integral part of current design flows

VIII.SIMULATION RESULT

Fig 5. Simulation output

IX.SCHEMATIC DIAGRAMS



X.CONCLUSION

In this project, the 3port router was the proposed design. The proposed router structure functionality is implemented in Verilog HDL and proven that this architecture gave less delay and high frequency, so speed of operation will increases .In this paper the Xilinx ISE14.7 EDA Tool is used for synthesis and simulation. The

data which can be send through the router is reached the destination with 3.935ns latency and the frequency was 360.101MHz.

We would like to express our deep sense of gratitude to Dr. H.S. Saini, Managing Director, Guru Nanak Group of Institutions for his tremendous support, encouragement and inspiration. Lastly, We would like to thank all the other staff members, both teaching and non-teaching, which have extended their timely help and eased my work.

In All Sincerity,

K. Swetha -21831A0458

P. Pradeep - 22835A0407

B. Mallesh- 21831A0413

XI.REFERENCES

[1] Robert G. Gallager Abhay K. Parekh. AY. Mansour, and A. Zhu. Competitive queueing policies for QoS switches. In *Proceedings of the 14th ACM-SIAM Symposium on Discrete Algorithms(SODA'03)*, pages 761–770, 2003.

[2] Y. Azar and Y. Richter. The zero-one principle for switching networks. In *Proceedings of ACMSymposium on Theory of Computing(STOC'04)*, pages 64–71, 2004.

[3] Y. Azar and Y. Richter. Management of multi-queue switches in QoS networks. *Algorithmica*, 43:81–96, 2005.

[4] Nikhil Bansal, Lisa K Fleischer, Tracy Kimbrel, Mohammad Mahdian, Baruch Schieber, and Maxim Sviridenko. Further improvements in competitive guarantees for QoS buffering. In *Proceedings of the 31st International Colloquium on Automata, Languages and Programming (ICALP'04)*, pages 196–207, 2004

[5] Y. Bartal, F. Chin, M. Chrobak, S. Fung, W. Jawor, R. Lavi, J. Sgall, and T. Tichy. Online competitive algorithms for maximizing weighted throughput of unit jobs. In *Proceedings of the 21st Symposium on Theoretical Aspects of Computer Science (STACS'04)*, 2004.

[6] J. C. R. Bennet and H. Zhang. *WF ^{2}Q*: Worst-case fair weighted queueing. In *Proceedings of IEEE INFOCOM'96*, pages 120–128, 1996.

[7] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss. An architecture for differentiated service, 1998.

[8] Bogdan Caprita, Jason Nieh, and Wong Chun Chan. Group round robin: Improving the fairness and complexity of packet scheduling. In *Proceedings of the 2005 Symposium on Ar- chitecture for Networking and Communications Systems*, pages 29–40, 2005.

[9] Marek Chrobak, Wojciech Jawor, Jiri Sgall, and Tomas Tichy. Improved online algorithms for buffer management in QoS switches. In *Proceedings of the 12th Annual Europian Symposium on Algorihtms(ESA'04)*, pages 204–215, 2004.

[10] A. Demers, S. Keshav, and S. Shenkar. Analysis and simulation of a fair queuing algorithm. *Journal of Internetworking Research and Experience*, pages 3–26, 1990.

[11] M. Englert and M. Westermann. Lower and upper bounds on FIFO buffer management in QoS switches. In *Proceedings of the 14th Annual European Symposium on Algorithms(ESA'06)*, pages 352–363, 2006.

[12] Toshiya Itoh and Noriyuki Takahashi. Competitive analysis of multi-queue preemptive qos algorithms for general priorities. *IEICE Transaction Fundamentals*, E89-A(5):1186–1197, 2006.

[13] Alexander Kesselman, Zvi Lotker, Yishay Mansour, Boaz Patt-Shamir, Baruch Schieber, and Maxim Sviridenko. Buffer overflow management in QoS switches. In *Proceedings of ACM Symposium on Theory of Computing(STOC'01)*, pages 520–529, 2001.

[14] Alexander Kesselman, Yishay

Mansour, and Rob Van Stee. Improved competitive guaran- tees for QoS buffering. In *Proceedings of the 11th Annual Europian Symposium on Algori- htms(ESA'03)*, pages 361– 372, 2003.

[15] Fei Li, Jay Sethuraman, and Clifford Stein. An optimal online algorithm for packet schedulingwith agreeable deadlines. In *Proceedings of the Sixteenth Annual ACM-SIAM Symposium on Discrete Algorithms(SODA '05)*, pages 801– 802, 2005.

[16] J. B. Nagle. On packet switches with infinite storage. *IEEE Transactions on Communications*, 35(4):435– 438, 1987.

[17] A. Sen, L. Mohammed, R. Samprathi, and S. Bandyopadhyay. Fair queuing with round robin: A new packet scheduling algorithm for routers. In *Proceedings of the Seventh International Symposium on Computers and Communications*, pages 120–128, 2002.

[18] M. Shreedhar and G. Varghese. Efficient fair queuing using deficit round robin. *IEEE/ACM Transactions on Networking*, 4(3):375–385, 1996.