International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 03 Issue: 11 | Nov -2016 www.irjet.net p-ISSN: 2395-0072

FPGA Implementation of Four Port Router for Network on Chip

Sagar Latti

PG Scholar, Dept. of VLSI & ES
Visvesvaraya Technological University-Belagavi
Karnataka-590018, India
Email: latti.sagar@gmail.com

Abstract - *Nowadays with technological advancement large number of electronic components can be integrated in a single system and such systems are controlled by single chip called system on chip (SoC).The system on chip integrates multiple Intellectual property Cores (IP Cores) on a single chip. Because of the integration complexity communications on System on Chip (SoC) became vital. The network on chip (NoC) is the technology used to provide lossless communication between these IP cores. The router is the fundamental component of NOC which uses store and forward mechanism to transfer data packets from source to destination. In this paper the design and synthesis of four port router is explained. The proposed router designed with HDL verilog using Xilinx 14.7 ISE tools.*

Key Words: **Network on Chip, Four Port Router, Router Packet, Router Register, Router FIFO, Parity Check,**

1. INTRODUCTION

The system on chip is an integrated circuit that integrates all the electronic components on a single chip. In this system on chip multiple Intellectual property (IP) cores are integrated, which makes the chip more complex. Hence the major problem is to provide the communication between these multiple IP cores. The on-chip physical interconnections will lead to energy consumption and time wastage. It is required to provide proper on chip communication for these components. Network on chip (NoC) is an approach to design communication subsystem between theses intellectual property cores. The dedicated buses can provide the communication between the resources, but this will not give any flexibility. Every time the bus architecture has to be changed when the design is modified, this reduces the device time to market hence NOC approach can be used. NoC architecture better supports the integration of SoC consisting of on switch packet switched network. These cores on SoC receive and forward the data packets coming from different components.

Router is the fundamental unit of NoC which receives the incoming packets from different cores on SoCs and forward them to specific destination based on the address contained on the header of data. The proposed four port router has one input port which receives the data and has three output ports from which the packets are driven out. The router in the NOC should be flexible and supports parallel connections at the same time. It uses store and forward mechanism. It does not reserve any channels hence the network traffic can be avoided.

The routers are used to overcome the congestion and communications bottleneck. The router supports four port parallel connections at the same time to provide better implementation for NoC, It uses store and forward type of control which reduces the data loss during the packet forward from source to destination. The switching mechanism is used here to improve the performance of the router.

In packet switching the data transfers in the form of packets between cooperating routers and the decisions on direction of the packet are taken independently. Whenever the data comes from the source to the input port on the router the router holds the data and checks the address information on the data and send to particular output port. In this proposed router the design consists of three main blocks.

The router uses a register to hold the incoming data and router FSM controls the operations to be performed by the router and the three FIFOs are used to hold the data to be driven out at the output ports. The router receives 8-bit of packets of data followed by the header and data 8-bits of data information and the parity data at the end of the packet to determine the success of data received at the output ports. The router can store up to 64 bytes of data information. The length of the data information is measured interns of payload which determine the number of data bytes in the packet.

In this paper the section I cover the introduction of router. In the section II the router design with each block operations are explained. In the section III simulation results are explained.

2. FOUR PORT ROUTER DESIGN

A. Router packet Format

In router the data information is given in the form of packets. The packets are transferred from source to destination based on the address information provided at the header at the beginning of packet. Then the packet is provided with data bytes. The router packet format is as shown in the fig 1.

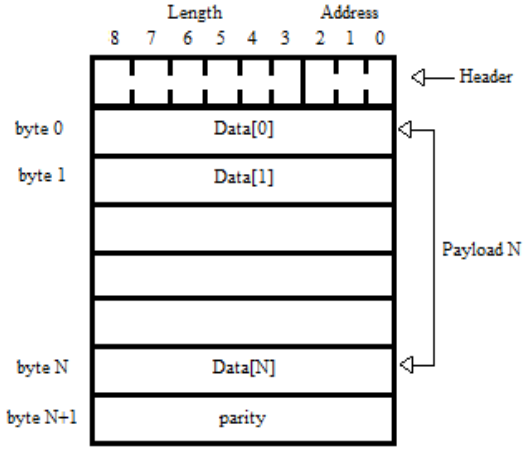


Fig -1: Router Packet Format.

1. Router Input Protocol.

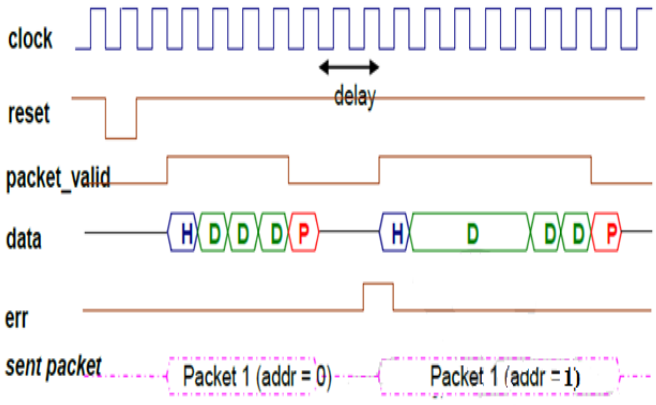


Fig -2 Router Input Protocol

1. Router Output Protocol.

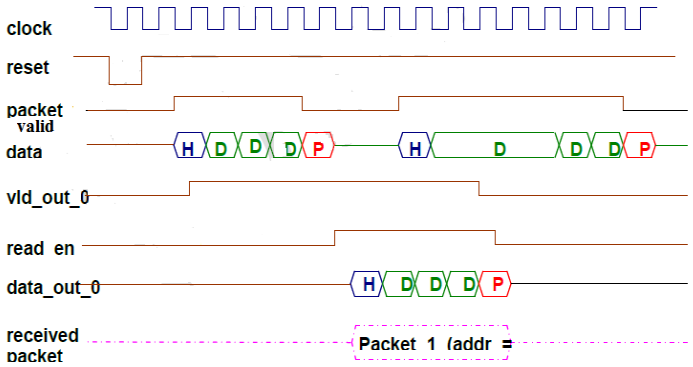


Fig -3 Router Output Ptotocol

Each packet consists of header byte, data-bytes, parity byte.

Header byte: The header byte consists of two blocks address and length. The address block consists of the destination address where the data has to be routed and the length of the data bytes indicated by payload.

Payload: The payload is the number of data bytes in the packet.

Parity: The parity byte is added at the end of the last data byte, the parity byte is used at the receiver to acknowledge that the byte received at the output port is same as of the input port.

The router designed in this paper is a four port router. It consists of one input port through which the packet enters and three output ports through which the data is driven out to its destination the block diagram of proposed router design is as show in the fig 4.

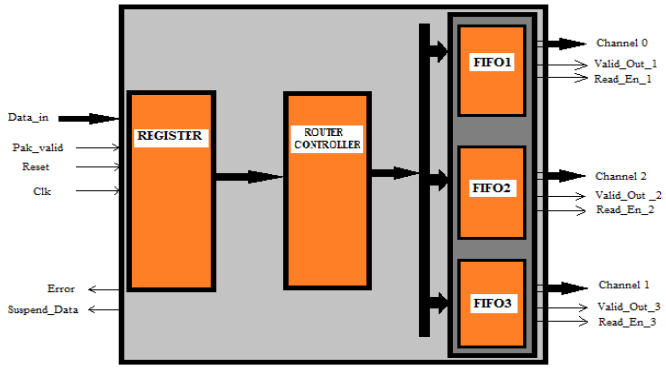


Fig - 4: Block diagram of four port router.

The proposed router consists of Register, Router controller, Three FIFO's.

Register: The Register is used in the router to latch the packet coming at the input port. The register holds the address and the length information in the header byte. It also holds the parity byte present at the end of the packet to check the parity at the output. The register consists of different signals at the input to load the header byte, data bytes and the parity byte. The register determines the parity of the each byte entered at the input port. And compares the parity with the parity byte loaded at the input port. The Router register block diagram is as shown in fig 5.

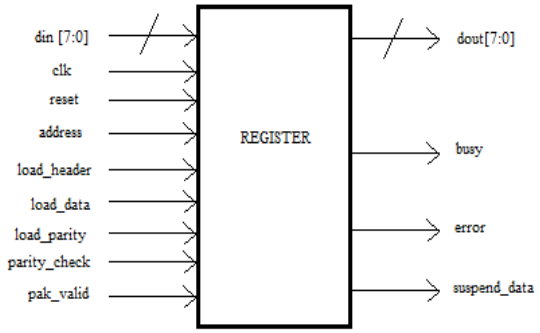
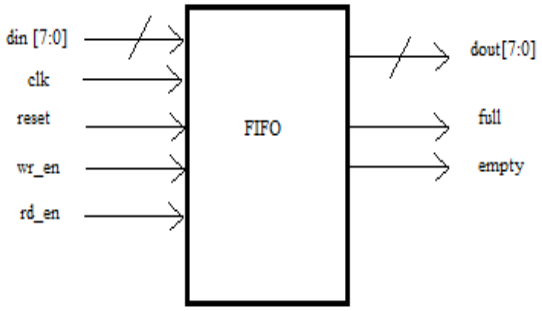


Fig - 5: Router Register Block

Router Controller: The router controller is used to determine the packet output port based on the address provided at the header byte of the packet. Based on the address the controller loads the data bytes in to the particular output port.

Router FIFO: The proposed router uses three FIFOs at the output ports to hold the packets driven out by the controller at the output port. Then by asserting the read_en signal at the output the data packet can be read at the output port. The router block diagram is as shown in the fig 6.



Fg - 6: Router FIFO.

3. SIMULATION RESULTS

The router design is synthesized and simulated using Xilinx 14.7 ISE tools.

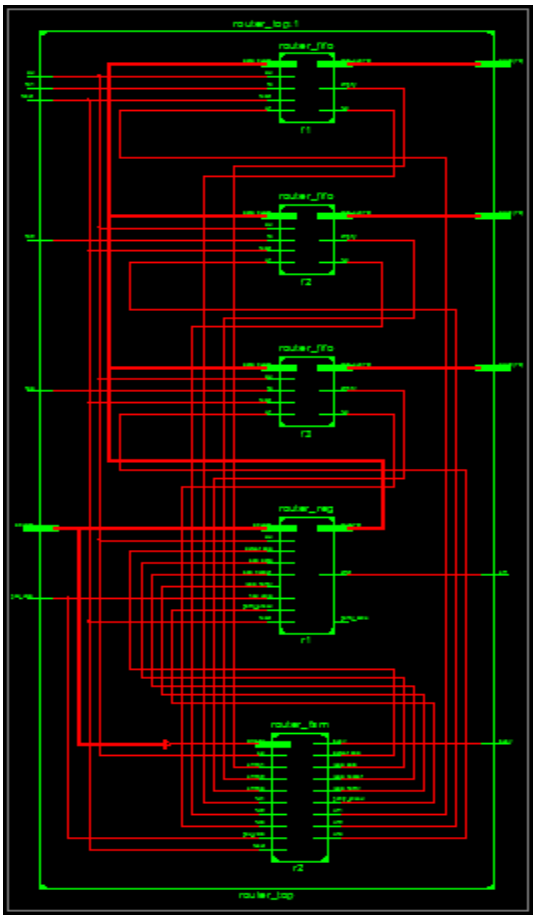


Fig -7: RTL Schematic of Four ort Router.

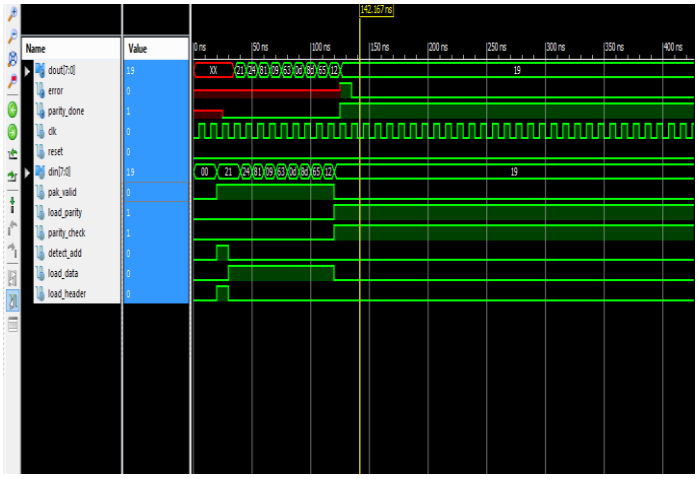


Fig - 8: Simulation results of Router Register.

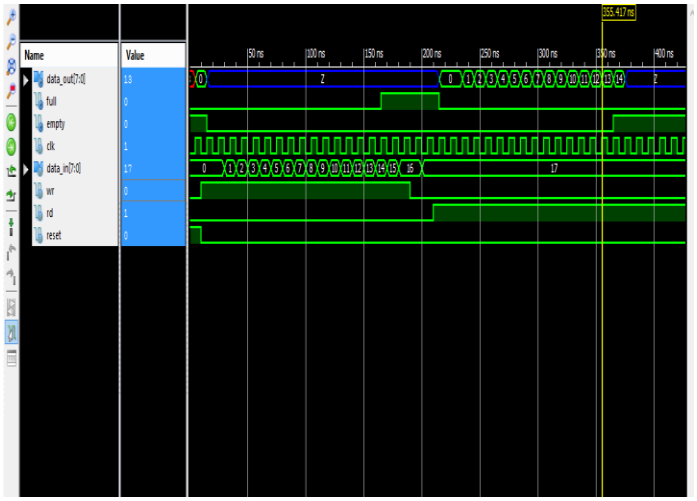


Fig - 9: Simulation results of FIFO.

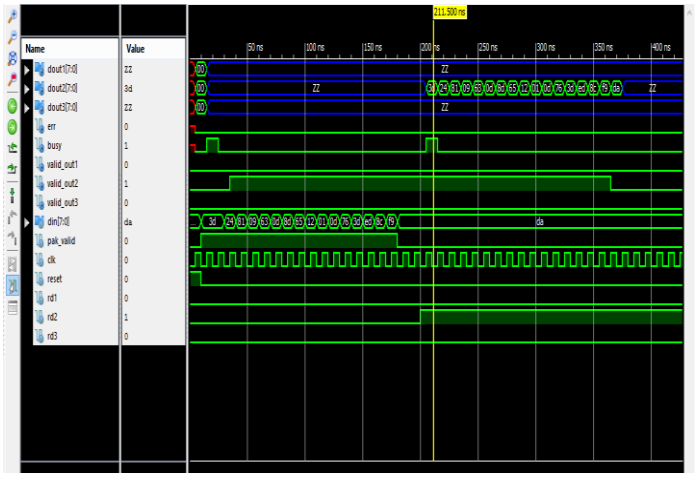


Fig -10: Simulations results of Four Port Router.

CONCLUSIONS

In this paper the efficient design and synthesis of router for Network on Chip is explained. The proposed router uses the hold and forward mechanism to receive and forward the data to the address specified on the header of the packet. The parity check is done at the end to determine the errors during the transfer. The proposed router can be used to provide high speed data transfers on SoCs.

IV. REFERENCES

[1] S. Vangal, J. Howard, G. Ruhl, S. Dighe, H. Wilson, J. Tschanz,D. Finan, A. Singh, T. Jacob, S. Jain, V. Erraguntla, C. Roberts,Y. Hoskote, N. Borkar, and S. Borkar, —An 80-tile sub-100-WTeraFLOPS processor in 65-nm CMOS,|| IEEE Journal of Solid-State Circuits, vol. 43, no. 1, pp. 29 –41, Jan. 2008.

[2] P. Golani, G. Dimou, M. Prakash, and P. Beerel, —Design of a highspeed asynchronous turbo decoder,|| Proc. 13th IEEE Int. Symp. on Asynchronous Circuits and Systems, pp. 49 - 59, Mar. 2007.

[3] N. Onizawa, V. Gaudet, and T. Hanyu, —Low-energy asynchronous interleaver for clockless fully parallel LDPC decoding,|| IEEE Trans. on Circuits and Systems I: Regular Papers, vol. 58, no. 8, pp. 1933 –1943, Aug. 2011.

[4] M. Horak, S. Nowick, M. Carlberg, and U. Vishkin, —A lowoverhead asynchronous interconnection network for gals chip multiprocessors,|| IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems, vol. 30, no. 4, pp. 494 –507, April 2011

[5] A. Lines, —Asynchronous interconnect for synchronous SoC design,|| IEEE Micro, vol. 24, no. 1, pp. 32 – 41, Jan.-Feb. 2004.

[6] Y. Thonnart, P. Vivet, and F. Clerisy, —A fully-asynchronous low power framework for GALS NoC integration,|| in Proc. Design, Automation Test in Europe Conference Exhibition 2010, pp. 33 - 38, Mar. 2010.