Indian Journal of Communication Engineering and Systems (IJCES)
Vol.2.No.2 2014 pp 83-87.
Available at: www.goniv.com
Paper Received: 04-04-2014
Paper Accepted: 14-04-2014
Paper Reviewed by: 1. R. Venkatakrishnan 2. R. Marimuthu
Editor: Prof. P.Muthukumar

NETWORK ON CHIP ADDRESS BASED NETWORK ROUTER (NANR) DESIGN FOR SOC APPLICATIONS

B.Silambarasi[1], Mr.P.S.Jayakumar[2]
[1]Post Graduate Student, [2]Associate professor
Department of Electronics and Communication,
Dhanalakshmi Srinivasan College of Engineering and Technology,
Affiliated to Anna University, Chennai-25
[1] silambarasi1991@gmail.com, [2]psjayakumar@gmail.com

ABSTRACT

The Network-on-Chip (NoC) is a specific architecture and represents the third layer working in the OSI model. NoC is integrated into the MPSoC (Multi Processor System on Chip) system to reduce the routing Complexity. The heart of an on-chip network is the router that carries out the critical task of coordinating the data flow. In this paper, the design of NoC router to transfer the data from source to destination without fault is presented. Conventional router is designed based on the priority. Highest Priority data will be routed first. This is done using Round Robin Arbiter (RRA). But it consumes more area and delay. Proposed router will be designed using specific packet format. The address based routing technique is proposed to reduce the complexity of NoC router for SoC application. Proposed novel NANR routing technique provides less area, delay and high speed than the conventional routing techniques. Simulation, Synthesis and implementation are performed by Modelsim6.3C, Xilinx10.1 and FPGA Spartan3.

Index terms – Network on Chip, Round Robin Arbitration (RRA), Network router, Synchronizer.

I. INTRODUCTION

System on a chip (SoC) is one of the design methodology recently used by VLSI designers, based on widespread IP core reuse. Cores do not make up SoCs alone; they must include an interconnection structural design and interfaces to side-line devices. Usually, the interconnection architectures based on dedicated wires or shared busses. Dedicated wires are effective only for systems with a small number of cores, since the number of wires in the system increases radically as the number of cores grows. Therefore, dedicated wires have poor reusability and flexibility [1].

A shared bus is a set of wires universal to various cores. This move towards is more scalable and reusable, when compared to dedicated wires. However, busses permit only one communication transaction at a time. Thus, every cores share the same communication bandwidth in the system and scalability is restricted to a few dozens IP cores. Using separate busses interconnected by bridges or hierarchical bus architectures may decrease some of these constraints, since different busses may account for different bandwidth needs, protocols and also increase communication parallelism. Nonetheless, scalability remains a problem for hierarchical bus architectures [4].A NoC appears as a probably better solution to apply prospect on-chip interconnection architectures. In the most commonly establish organization, a NoC is a set of interconnected switches, with IP cores connected to these switches. NoCs offer better performance, bandwidth, and scalability than shared busses [5]. The router operation revolves around two essential systems: (1) the data path and (2) the associated control logic. The data path consists of number of input and output channels to facilitated packet switching and traversal. Generally 5 input X 5 output routers is used. Out of five ports four ports are in cardinal direction (North, South, East, and West) and one port is attached to its local processing element [2].

II. CONVENTIONAL RRA BASED ROUTER

In the conventional method, Round Robin Arbitration (RRA) technique is used find the priority of the different data, which comes from different input channel. Arbiter is generally used to find the priority. Different arbiter is available like Matrix arbiter and Round Robin Arbiter. Matrix arbiter consumes more area and delay. And also very difficult to design Matrix arbiter. If more than one channel request for same time, arbiter set the priority based on the highest number of ones. During the contention situation Round Robin Arbiter will execute cyclic manner like RANDC. Hence every data transfer within one cycles [9].

The round robin arbiter design requirement can be summed up in a list:

1. One cycle calculation, so the arbiter can grant different requestors in each cycle.
2. Wraparound functionality, meaning that the arbiter does not loose cycles at the end of each round when moving from a grant to the last active requestor, back to the first one.

Work conserving functionality, so no cycles are lost on requestors that are inactive

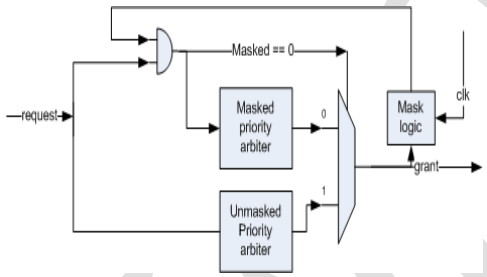

Fig.1 General working Principle of Round Robin Arbiter.

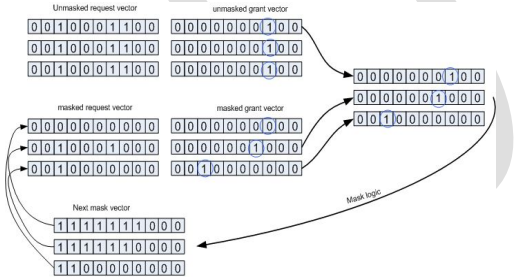

Fig.2 Examples of Round Robin Arbitration.

The wrap around functionality, results from the two priority arbiters. Once the masked request vector has no active requests, the mux will cause the grant to be generated from the non-masked priority, which will naturally start at the first requestor. The two priority arbiters implement simple find-first-set priority encoding, returning the first bit which is set in their respective input vectors [7]. In the first cycle, the masked grant vector is all 0, so the MUX selects the unmasked grant and therefore grants the first requestor (bit #2). The mask logic calculates the mask for the next cycle, by masking all requestors below the selected one as well as the selected requestor itself, so bits 0-2 are masked and all the rest are enabled. At the second cycle, the masked request vector which is the result of an AND between the first cycle "Next mask vector" and the request vector is not all 0, so the resulting grant is bit #3 which is the lowest bit set in the "masked request vector". This arbitration technique consumes more area and less delay. So processing time of this conventional router is more [5].

III. PROPOSED ADDRESS BASED ROUTING TECHNIQUE

In packet switching, the message to be transmitted is partitioned and transmitted as Fixed-length packets. Routing and control is handled on a per packet basis. The packet header includes routing and other control information needed for the packet to reach. Packet switching increases network resource utilization as communication channels share resources along the path. Buffers and arbitration units in routers manage resource conflicts and storage demands in communication paths. Packet switching networks aid IP block re-use and are scalable. Packet-switching is more flexible than circuit switching though it requires buffering and introduces unpredictable latency (jitter).

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 [8]. 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 1 bytes. If Length = 2, it means data length is 2 bytes. If Length = 63, it means data length is 63 bytes.

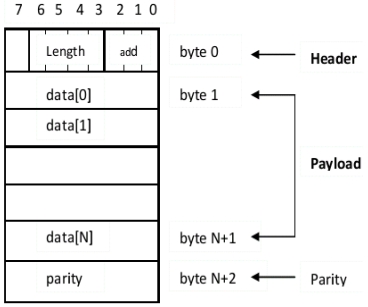

Fig.3 Block diagram of Proposed Specific packet format.

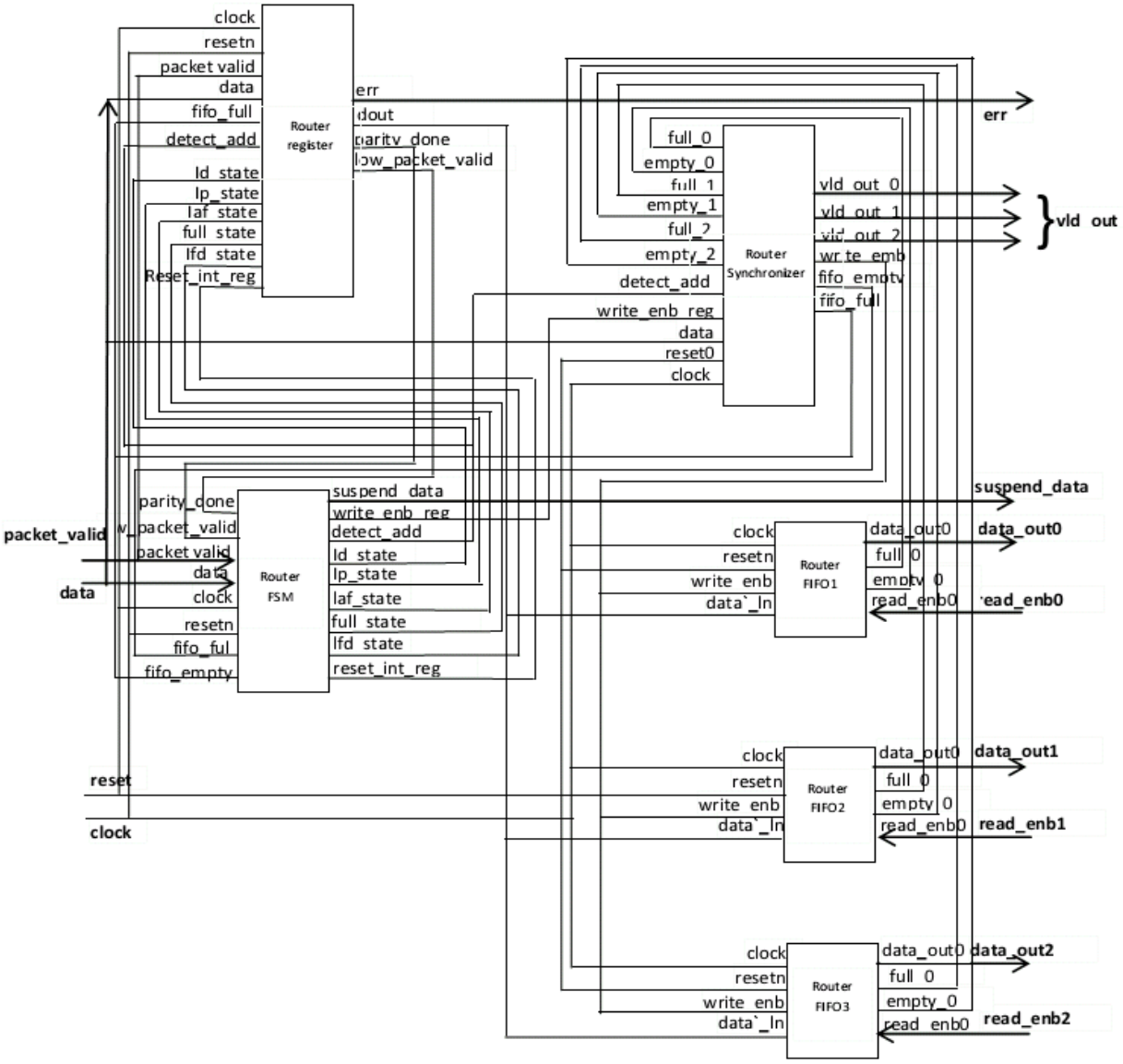

Fig.4 Architecture of Proposed NoC Address based Network Router (NANR).

The proposed NANR router consists of register, synchronizer, FSM and three FIFO. The 'fsm_router' module is the controller circuit for the router. This module generates all the control signals when new packet is sent to router. These control signals are used by other modules to send data at output, writing data into the FIFO. There are 3 FIFOs used in the router design. Each FIFO is of 8 bit width and 16 bit depth. The FIFO works on system clock. It has synchronous input signal reset.

Write operation: The data from input data_in is sampled at rising edge of the clock when input write_enb is high and FIFO is not full.

Read Operation: The data is read from output data_out at rising edge of the clock, when read_enb is high and FIFO is not empty. Read and Write operation can be done simultaneously. Full indicates that all the locations inside FIFO have been written. Empty indicates that all the locations of FIFO are empty.

Router Synchronizer provides synchronization between FSM and FIFO modules. It provides faithful

communication between single input port and three output ports. It will detect the address of channel and will latch it till packet_valid is asserted; address and write_enb_sel will be used for latching the incoming data into the FIFO of that particular channel.

Router register contains status, data and parity registers required by router. All the registers in this module are latched on rising edge of the clock. Data registers latches the data from data input based on state and status control signals, and this latched data is sent to the FIFO for storage. Apart from it, data is also latched into the parity registers for parity calculation and it is compared with the parity byte of the packet. An error signal is generated if packet parity is not equal to the calculated parity. Parity check method is used to find the error in the proposed router [3].

IV. RESULT AND DISCUSSIONS

Conventional priority based network router and proposed Address based network router are designed using Verilog and implemented in Xilinx Spartan 3 XC3S400 (package: pQ208, speed grade: -5) FPGA.

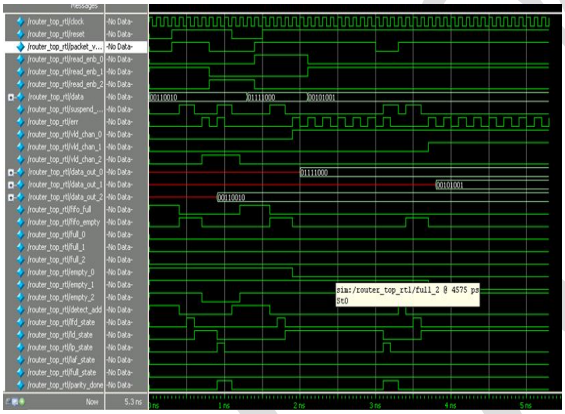

Fig.5 Simulation of proposed address based network router.

Total equivalent Slices in case of priority based router is 1204 and that is improved to 113 using Address based router. Delay used in priority based router is also improved. In case of priority based router it is 13.604ns and in Address based router it is 8.841ns. So throughput of the proposed router is also increased.

Table1. Comparison of area and delay between Existing and Proposed router.

Methods	Slices	Frequency (MHz)	Delay(ns)
Conventional Router	1242	73.076	13.604
Proposed router	113	113.116	9.824

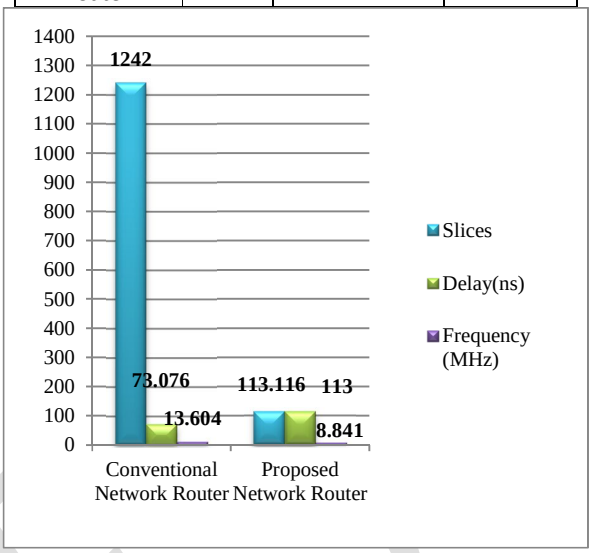

Fig.5 Performance comparison of Existing and Proposed network router.

V.CONCLUSIONS

This paper, proposed an address based network router for System on Chip application. In the Conventional method, number of input and output channel is equal. Conventional design is based on RRA (Round Robin Arbitration) technique. This method route the data to the corresponding channel like input channel1 route the data to output channel1. But it cannot route the data to all the output channel. So novel proposed address based routing technique is designed to transfer the data from one channel to the entire output channel. This technique is very useful to select the alternative path, when one of the path is damaged. Proposed network routing technique consumes less area, delay and high frequency than the conventional RRA network routing technique.

route the data to the corresponding channel like input channel1 route the data to output channel1. But it cannot route the data to all the output channel. So novel proposed address based routing technique is designed to transfer the data from one channel to the entire output channel. This technique is very useful to select the alternative path, when one of the path is damaged. Proposed network routing technique consumes less area, delay and high frequency than the conventional RRA network routing technique.

REFERENCES

[1]. Alireza Ejlali, Bashir M. Al-Hashimi, Paul Rosinger, Seyed Ghassem Miremadi and Luca Benini, "Performability/Energy Tradeoff in Error-Control Schemes for On-Chip Networks". IEEE Transactions On Very Large Scale Integration (Vlsi) Systems, Vol. 18, No. 1, January 2010.

[2]. Cédric Killian, Camel Tanougast, Fabrice Monteiro, and Abbas Dandache, "Smart Reliable Network-on-Chip". IEEE Transactions On Very Large Scale Integration (VLSI) Systems". IEEE Transactions On Very Large Scale Integration (Vlsi) Systems 2013 IEEE.

[3]. Erica cota, Fernanta Lima kasteinsmidt and Maico castel, "High fault coverage approach for the test of data, control and handshake interconnect in Mesh network on Chip".2008 IEEE Published by the IEEE Computer Society.

[4]. Leandro Fiorin and Mariagiovanna Sami, "Fault-Tolerant Network Interfaces for Networks-on-Chip". IEEE Transactions On Dependable and Secure Computing, Vol. 11, NO. 1, JANUARY/FEBRUARY 2014. IEEE Published by the IEEE Computer Society.

[5]. Maurizio Palesi, Giuseppe Ascia, Fabrizio Fazzino and Vincenzo Catania, "Data Encoding Schemes in Networks on Chip". IEEE Transactions On Computer-Aided Design Of Integrated Circuits And Systems, Vol. 30, No. 5, May 2011.

[6]. Mehdi Modarressi, Arash Tavakkol and Hamid Sarbazi-Azad, "Application-Aware Topology Reconfiguration for On-Chip Networks". IEEE Transactions On Very Large Scale Integration (Vlsi) Systems, Vol. 19, No. 11, November 2011.

[7]. Mohammad Hosseinabady, Mohammad Reza Kakoee, Jimson Mathew, and Dhiraj K. Pradhan, Low Latency and Energy Efficient Scalable Architecture for Massive NoCs Using Generalized de Bruijn Graph". IEEE Transactions On Very Large Scale Integration (Vlsi) Systems, Vol. 19, No. 8, August 2011.

[8]. Po-An Tsai, Yu-Hsin Kuo, En-Jui Chang, Hsien-Kai Hsin and An-Yeu (Andy) Wu, "Hybrid Path-Diversity-Aware Adaptive Routing with Latency Prediction Model in Network-on-Chip Systems".2013 IEEE.

[9]. Si Qing Zheng and Mei Yang, "Algorithm-Hardware Codesign of Fast Parallel Round-Robin Arbiters". IEEE Transactions On Parallel And Distributed Systems, Vol. 18, No. 1, January 2007. 2007 Ieee Published by the IEEE Computer Society.

[10]. Soo Yun Hwang, Dong Soo Kang, Hyeong Jun Park, and Kyoung Son Jhang, Member, IEEE Transaction on "Implemention of Self Motivated Arbitration Scheme for the Multilayer AHB Busmatrix.