iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 5 **Issue:** IV **Month of publication:** April 2017

DOI: http://doi.org/10.22214/ijraset.2017.4216

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

Volume 5 Issue IV, April 2017
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

Low Latency NoC Router Micro Architecture using Dynamic Virtual Channel Organization

Praveen M [1], Mothi. R[2]
[1]M.E Vlsi Design, [2]Assistant Professor, ECE, Mahendra Engineering College, Namakkal, India

Abstract: ***The number of cores on a chip is rapidly increased the on chip need an efficient communication structure as network on chip (NoC). The channel buffer organization of NoC uses virtual channels (VCs) to improve data flow and performance of the NoC system. Dynamically allocated multi queues are an good mechanism to reach VC flow control with maximum no of buffer utilization. In this design, VCs employ variable number of buffer slots depending on the network traffic. we propose a new input port micro architecture to cooperate our efficient dynamic VC (EDVC) approach that is built on DAMQ buffers. To show the advantages of EDVC, we compare its micro-architecture with that of the traditional dynamic VC (CDVC), it's also employs link-list tables for buffer organization. In terms an hardware, EDVC input-port organization consumes on average 51% less power for ASIC design when compared with the CDVC input port. The saving is even good when compared with VC regulator methodology. This EDVC approach can improve NoC latency by 38%–48% and throughput by 100% on average as compared with the CDVC mechanism. The proposed design parameters are analyzed using Xilinx 9.1.***
Keywords: NOC, CDVC, DAMQ, EDVC

I. INTRODUCTION

In the term Network on chip or network on a chip NoC or NOC is a system for communication on an integrated circuit (commonly called a chip), typically intellectual property (IP) cores in a system on a chip (SOC). Network on chips can span synchronous, asynchronous clock areas or use unlocked asynchronous logic system. Network on chip technology uses networking theory to on-chip communication and brings good improvements over conventional bus and crossbar interconnections. Network on chip improves the scalability of System on Chips, and the power efficiency of complex System on Chips compared to other designs. Traditionally, Integrated Circuits are designed with dedicated point-to-point connections, with one wire dedicated to every signal. For big designs, this has no. of limitations from a physical design viewpoint. The wires occupy much of the area of the chip, and in nanometer CMOS technology, interconnects dominate both performance and dynamic power dissipation, as signal propagation in wires across the chip requires multiple clock cycles. Always network on a chip's can borrow concepts and techniques from general computer networking, it's impractical to blindly reuse features of general computer networks and symmetric multiprocessors. In particular, network on a chip switches size is small, energy-efficient, and fast. . The routing algorithms should be implemented by simple logic, and the number of data buffers should be minimize the Network topology and execution properties may be application-specific on MPSOCs Some designers think that NOCs need to support QOS, namely achieve the various needs in terms of efficiency, end-to-end delays, fairness,[1] and deadlines.

II. SCHEDULING (COMPUTING)

Scheduling in the computing a method by which work specified by some means is assigned to resources that did the work. The resources may be virtual computation elements such as threads, processes or data flows, which are in turn scheduled onto hardwares as processors, network links or expansion cards. A scheduler is carries out the scheduling activity. Scheduling is fundamental to computation itself, and an intrinsic part of the execution model of a computer system; the concept of scheduling makes it possible to have computer multitasking with a single central processing unit (CPU).A scheduler may aim at one of many goals, for example, maximizing throughput (the total amount of work completed per time unit), minimizing response time (time from work becoming enabled until the first point it begins execution on resources), or minimizing latency (the time between work becoming enabled and its subsequent completion), maximizing fairness (equal CPU time to each process, or more generally appropriate times according to the priority and workload of each process). In practice, these goals often conflict (e.g. throughput versus latency), thus a scheduler will implement a suitable compromise.

©IJRASET: All Rights are Reserved

III. PROPOSED SYSTEM

The architecture of the EDVC input port joined with the NoC router micro architecture is shown in Fig. 3. Our new d EDVC router consists of five input ports, arbiter, crossbar switch as shown in Fig. 3(a). However, the architecture of the EDVC input port is much simpler in terms of less and efficient hardware and buffering as shown Buffers in NoC routers can be placed at three locations: 1) input ports; 2) output ports; or 3) both input and output ports We employ asynchronous communication in our EDVC mechanism for NoCs. The working detail of EDVC is .Flit Arrival a credit-in signal causes the incoming flit and its VC-ID to be saved in a slot pointed by the write-pointer. the corresponding bit of the slot-state table is set.

Then the Request Signal read pointer points to a slot and its slot state bit is set, a request signal is issued to the VC-ID. Then the arbiter read the flit information and do the arbitration. Then If the requested output port is open, the arbiter allocates the correct address for the crossbar switch and VC-ID before issuing a grant signal. Flit Departure grant signal causes the flit to leave the buffer. The corresponding bit of the slot state table is reset. The high level of grant at the –ve clock edge causes the credit out and grant signals to be set and reset, respectively. Blocking: If the requested output port of a Virtual Channel is closed, the arbiter issues the Virtual Channel block signal to close the corresponding Virtual Channel. Closing a Virtual channel means that a request is not issued and no flit enters the buffer for the Virtual Channel. In Fig. 4, the EDVC working process is showed above is contrasted with the CDVC working process.

IV. FPGA IMPLEMENTATION

In this part, the optimization techniques for NoC based FPGA implementation are discribed. The analyses on the implementation results using Xilinx FPGA design boards are discussed.

A. Synthesis Results

The NoC programming code is written in Verilog HDL and synthesized using Xilinx ISE 9.2i software. The code was verified using Modelsim Simulator version. when it is complemented with two and four VCs. In overall an Network on Chip router with four Virtual Channel's utilize twice LCs compared to the one with two VCs. Spartan FPGA device has been used in this analysis. We also generated two CONNECT NoC routers, one with one- and another with two-clock-cycle latency using the online CONNECT NoC router generator tools [12]. The proposed NoC router and the two CONNECT NoC routers are configured with the same parameter of 4 VCs with capacity of 4 flits per eachVC and flit payloadwidth of 32 bits. With the same router parameters, CONNECT is twice costly in term of LCs utilization due to implementation of input memory buffers as LCs. Our proposed two-clock-cycle NoC architecture is able to work with two times faster operating frequency compared to CONNECT NoC when configured as one-clock-cycle latency router. The comparisonof to the two-clock latency CONNECT router, our proposed NoC router is 37% and 21% faster when implemented in Spartan3E and Vertex V FPGA, respectively.

B. Performance Results

In order to compare network performance results, we generate cycle-accurate behavioral model of the one-clock-cycle latency CONNECT and our proposed architecture using Verilator . Both NoC are configured in a 5 × 5 mesh topology having 4 VCs on each port with the size of 4 flits per each VC and the flit payload width of 32 bits. As CONNECT only supports dimension order routing (DoR),we useDoR for both routers.

All NoC endpoints are jointed to the custom traffic generator blocks. The traffic generators are response for injecting network packets to the NoC routers and collecting performance statistics as the packets are received by destination cores. For a fixed injection ratio, each endpoint injects packets of 5 flits into randomly selected destination router. Adopting the same operating frequency, as expected CONNECT-one-clk has lower average packet latency, since our design is two clk cycles' latency router. However our experimental results showed when the load injection ratio is above 42% our proposed architecture outperforms CONNECT-one-clk. illustrate an example when each router is running at its own maximum frequency (CONNECT with 40MHz and proposed design with 80MHz). In this figure, the load and the delay are scaled based on Gbit/second and nanosecond, respectively. Our proposed architecture offers significantly lower delay as injection ratio increases

C. Non-Uniform Virtual Channel Allocation

Implementing VCs needs extra buffer space and control logic to handle the flits transmission. Our basic idea for VC planning is to allocate the VC resources only when necessary. More precisely, we allocate the VCs at these router ports where the congestion is most likely to occur. From our experimental results, the effect of this non-uniform (or "on-demand") VC allocation can greatly

©IJRASET: All Rights are Reserved

Volume 5 Issue IV, April 2017
ISSN: 2321-9653

improve the network throughput, while only allowing a limited number of VCs in the network. In what follows, we present the problem formulation and discuss its analytical solution.

Table 1
Device Utilization Summary

Logic Utilization	Used	Available	Utilization
Number of Slice Flip Flops	11	9,312	1%
Number of 4 input LUTs	367	9,312	3%
Logic Distribution			
Number of occupied Slices	197	4,656	4%
Number of Slices containing only related logic	197	197	100%
Number of Slices containing unrelated logic	0	197	0%
Total Number of 4 input LUTs	367	9,312	3%
Number of bonded IOBs	100	190	52%
Number of GCLKs	1	24	4%
Total equivalent gate count for design	2,368		
Additional JTAG gate count for IOBs	4,800		

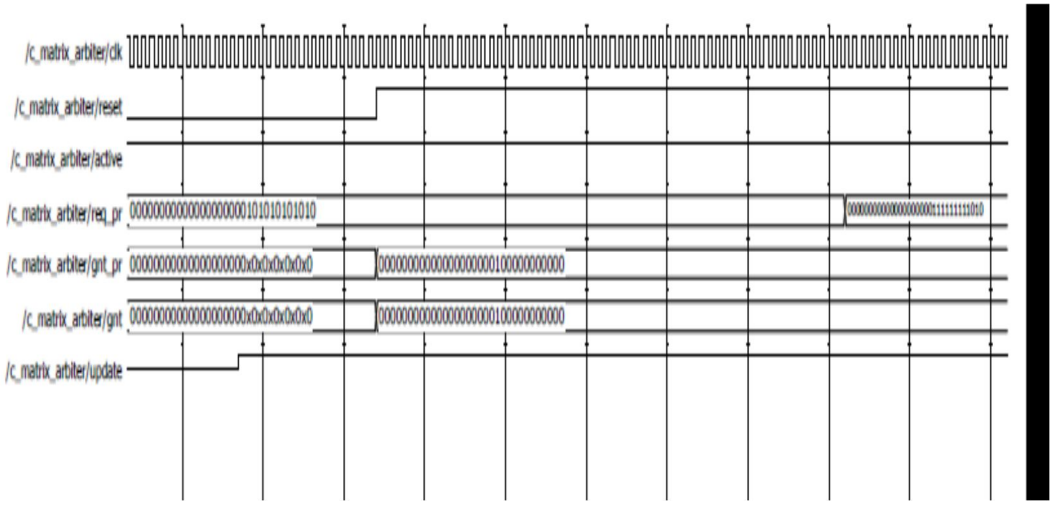


Fig 1.Simultation Output

V. CONCLUSION

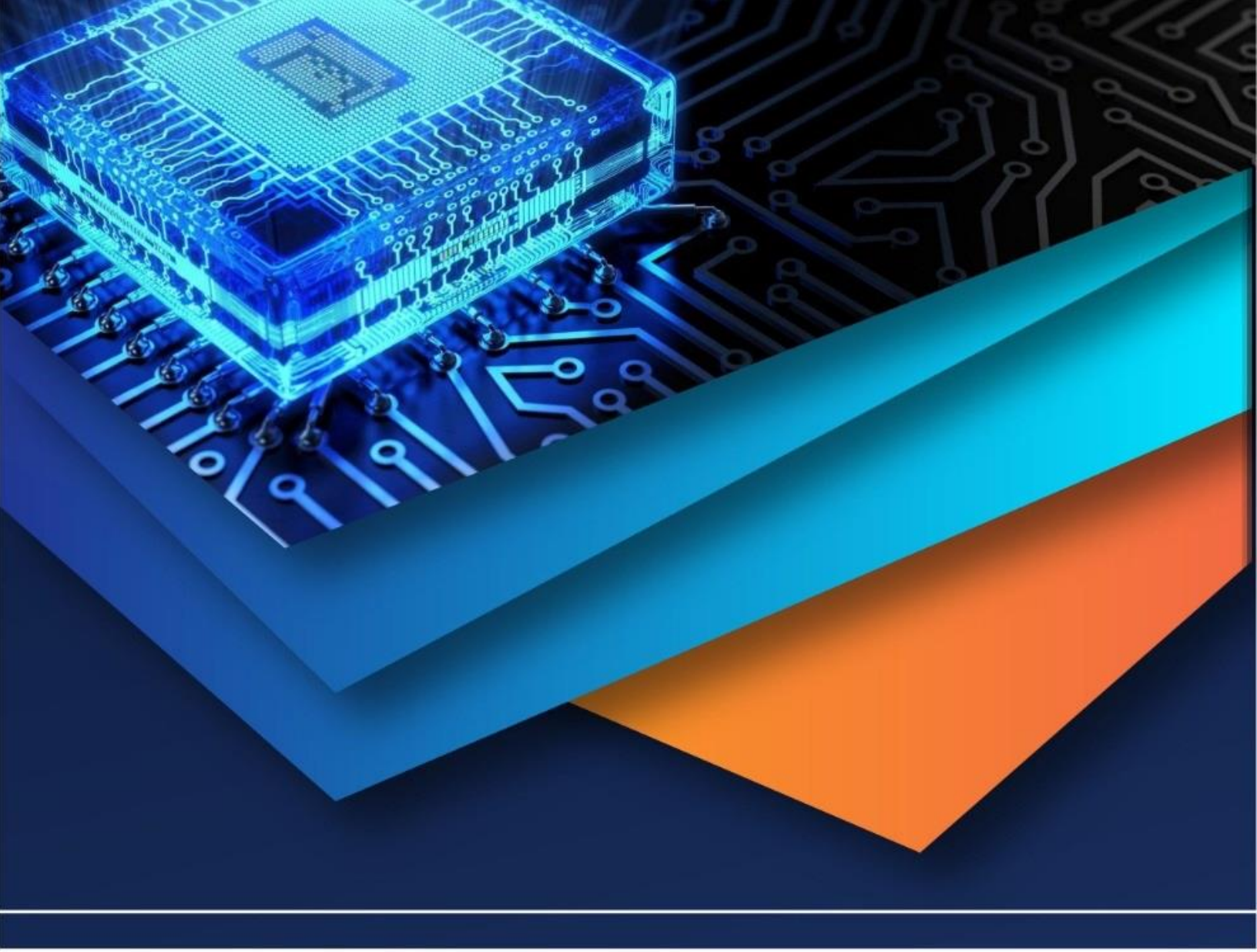
In this work, a two-clock-cycle pipeline wormhole virtual channel NoC router architecture was proposed. The router first pipeline stage is linked to the parallel look-ahead route computation with a no speculative VC/switch allocation. The second stage is crossbar switch traversal. The proposed router micro architecture is splited in three main criteria, which are hardware cost, maximum operating frequency, and QoS compared to existing technique. Compared to the proposed no speculative allocation removes the need for prioritizing any IVC request and reduces unused time slots, which result in better QoS.The Broadcasting only two bits of OVC status to the input ports instead of applying proxy credit counters results in lesser hardware cost. Our analysis shows that the proposed masking technique has negligible influence on NoC critical path delay. Compared to CONNECT, our NoC router has 40% lesser LCs utilization. The proposed router works with two times higher operating frequency when CONNECT is configured as one-clock-cycle latency and is 35%~20% faster when CONNECT is configured as two-clock-cycle latency router. In the results shows that our two clock-cycle NoC architecture outperforms CONNECT-one clk by 2.6 times in terms of performance.

©IJRASET: All Rights are Reserved

REFERENCES

[1] W.Dally and B.Towles, "Route packets, not wires: on-chip interconnection networks," in Proceedings of the Design Automation Conference, pp. 684–689, 2001.
[2] R.Mullins, A.West, and S.Moore, "The design and implementation of a low-latency on-chip network," in Proceedings of the Asia and South Pacific Design Automation Conference, pp. 164– 169, January 2006.
[3] D. U. Becker, Efficient microarchitecture for network-on-chip routers [Ph.D. thesis], Stanford University, Stanford, Calif, USA, 2012.
[4] S. T. Nguyen and S. Oyanagi, "The design of on-the-fly virtual channel allocation for low cost high performance on-chip routers," in Proceedings of the 1st International Conference on Networking and Computing (ICNC '10), pp. 88–94, IEEE, November 2010.
[5] Y. Lu, J. McCanny, and S. Sezer, "Exploring virtual-channel architecture in FPGA based networks-on-chip," in Proceedings of the 24th IEEEInternational SystemonChipConference (SOCC'11), pp. 302–307, September 2011.
[6] R. Mullins, A. West, and S. Moore, "Low-latency virtualchannel routers for on-chip networks," in Proceedings of the 31st Annual International Symposium on Computer Architecture, vol. 32, pp. 188–197, ACM, June 2004.
[7] M. K. Papamichael and J. C. Hoe, "CONNECT: re-examining conventional wisdom for designing nocs in the context of FPGAs," in Proceedings of the 2012 ACM/SIGDA International Symposium on Field Programmable Gate Arrays (FPGA '12), pp. 37–46, ACM, February 2012.
[8] J. D. Owens, W. J. Dally, R. Ho, D. N. Jayashima, S.W. Keckler, and L.-S. Peh, "Research challenges for on-chip interconnection networks," IEEE Micro, vol. 27, no. 5, pp. 96–108, 2007.
[9] A. Kumar, P. Kundu, A. P. Singh, L.-S. Peh, and N. K. Jha, "A 4.6 Tbits/s 3.6GHz single-cycle NoC router with a novel switch allocator in 65nm CMOS," in Proceedings of the 25th IEEE International Conference on Computer Design (ICCD'07),pp. 63–70, October 2007.
[10] S. Kumar, A. Jantsch, J.-P. Soininen et al., "A network on chip architecture and design methodology," in Proceedings of the IEEE Computer Society Annual Symposium on VLSI, pp. 105– 112, Pittsburgh, Pa, USA, 2002.

©IJRASET: All Rights are Reserved



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)