Open access Journal International Journal of Emerging Trends in Science and Technology

Efficient Flow Control And Congestion Averting Mechanism In NOC Router Architecture

Chandralekha D[1], Dr.X.Susan Christina M.E., Ph.D., [2]

1. M.E in VLSI Design, M.I.E.T Engineering College, Trichy, India. Chandra.kings.ece@gmail.com

2. Professor and Head, Department of ECE, M.I.E.T Engineering College, Trichy, India.

Abstract - Network on chip is a communication subsystem in integrated circuits and also scalable communication structure. In NOC router architecture, input ports are used to store and forward the data. Conventional input port contains efficient dynamic virtual channel uses logic circuits to transfer the data. In this input port, data is stored in SRAM and forward to the destination. In proposed system, a new input port contains buffering logic to store and forward the data. It is based on packet based wormhole routing mechanism. This is used to transmit the messages as multiple packets and each packet contains number of flits. The wormhole routing used to avoid the head of line blocking which increases the throughput. In input ports, FIFO buffers are used to store and forward the data. A FIFO buffer has four write pointers and only one read pointer. It provides simultaneous read and write operations. No wastage of clock cycles during data transmission and it increases the performance of NOC system. Buffering the input port used to reduce the latency by 27% as compared with efficient dynamic virtual channel method and also increases the system performance. This buffering method is used for efficient data flow control and congestion avoidance. Input buffering used to reduce complexity and lower cost of implementation.

Keywords: On-chip communication, Router architecture, Buffering, Flow control.

1. INTRODUCTION

Network on Chip (NOC) is an approach to designing communication subsystem between intelligent property (IP) cores in a system on chip. Typical elements of Network-on-Chip are processing elements, network interface card and router. The Processing Elements (PEs) can be various processors, memory elements and dedicated hardware like audio cores, video cores, wireless transceivers etc. The network interface (NIC) is used for communication between router and the processing elements (PE). The Network Interface is used to packetize data before using the router backbone to transverse the NOC. The routers are used to connect the various elements in the chip. Routers are mainly used for communication, packets routing and safe transfer from the source to the destination in the sub-system and also prevented the congestion.

A. 5 *X 5 Router Architecture*

A router is the key component and called as the communication backbone in NOC. A router is responsible for routing information from a source port to its destination port. The router is shown in Figure 1 consists of five ports east, west, north, south and local port and a central cross point matrix. The data is transferred from input port to output port. Each input port and output port has its own decoding logic which increases the performance of the router.

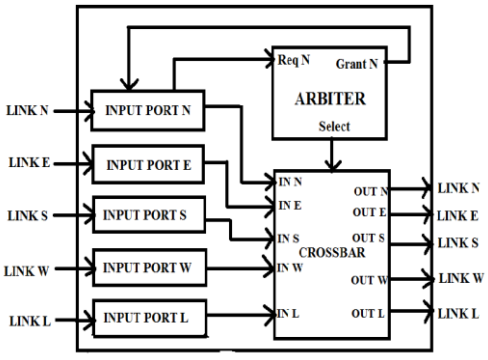

Figure 1. 5 x 5 Router architecture

The buffering method is used to store and forward the data. Control logic is present to make arbitration decisions. According to the destination path of data packet, control bit lines of crosspoint matrix are set. Packet-based wormhole routing is the most viable communication mechanism employed in NOCs. In wormhole routing, the messages are organized as multiple packets, where each packet consists of varying number of flits. The header flit holds the information about the packet destination. When it passes through routers, the route path is kept reserved until all the packet flits pass through it. This packet based method used to avoid the head of line blocking problem.

II. PREVIOUS RESEARCH WORK

A new type of buffer called dynamically allocated multiqueue (DAMQ) buffer which is used in router architecture and n x n switches. A new buffer design compared to several alternative designs in the context of a multistage interconnection networks was proposed by Frazier G.L and Tamir Y [4]. This new buffer design provide non FIFO message handling and efficient storage allocation for variable size packets using linked lists managed by a simple on chip controller. This dynamically allocated multiqueue buffers provide features such as non FIFO handling of packets and dynamic partitioning of buffer storage. It has lower latencies, higher throughput and maximum buffer storage capacity.

Virtual channel regulator is a centralized buffer architecture which was designed by Nicopoulos C.A et al [3]. A novel unified buffer structure also called the dynamic virtual channel regulator. This UBS allows a router to use a dynamically assigned virtual channel management scheme. This is used to dynamically allocate virtual channels and buffer resources according to network traffic conditions.

The virtual channel regulator (ViChaR) module is a very compact unit operating at the granularity of one router input and output port. The main difference between a generic and ViChaR router is that ViChaR uses a novel, tabled-based design which provides single-clock operation without overhead. A novel shared self compacting buffer scheme for SOC was introduced by Delgado-Frias et al [5]. It is based on dynamically allocated multiqueue self compacting buffer. In this buffer two virtual channels shared the same buffer space. DAMQ allocates buffer space when a packet is received. In this system, duato's routing methodology and adaptive routing protocols used for routing the data. Duato's routing methodology provides dimension order path selection function.

In on chip routers, simple virtual channel allocation design used to obtain high throughput and high frequency. This simple virtual channel allocation was designed by Zhao B and Zhang Y [2]. The design proposes two new virtual channel allocations (VA) mechanisms, termed Fixed VC Assignment with Dynamic VC Allocation (FVADA) and Adjustable VC Assignment with Dynamic VC Allocation (AVADA). A buffer is used as a set of virtual channels (VCs) to store incoming or outgoing packets.

A high-Throughput Distributed Shared-Buffer NOC Router was designed by Lin B and Soteriou V [7]. Buffers are needed in routers to house incoming flits which cannot be immediately forwarded due to contention. This buffering can be done at the inputs or the outputs of a router, corresponding to an input buffered router (IBR) or an output-buffered router (OBR). Virtualizing Virtual Channels for increasing robustness and Upgradeability in NOC router architecture was dealt by Evripidou .M and Kim .J [3]. The buffering resources orchestrate the flow-control mechanism of the network and facilitate the so called Virtual Channels (VC), which enable the multiplexing of several packets onto a single physical channel.

Efficient dynamic virtual channel (EDVC) buffer organization in NOC router architecture was presented by Masoud Oveis-Gharan and Gul N. Khan [8]. In this method, input microarchitecture used to store and forward data by using dynamic virtual channels. This microarchitecture used to reduce the complexity of conventional dynamic virtual channel and also improve the performance of NOC systems. This microarchitecture contains efficient dynamic virtual channel (EDVC) uses logic circuits instead of lookup tables. EDVC used to reduce the hardware in the architecture.

III. PROPOSED METHODOLOGY

Routers are used to transfer the data from source to destination. Buffering method provide efficient flow control and congestion avoidance. In proposed methodology, input ports, an arbiter

and crossbar used to transmit the data from source to destination. Arbiter used to provide the priority for data during transmission. Each input port contains FIFO and control logic. FIFO buffer used to store the data temporarily. Crossbar used to connect the input ports and output ports.

A. Schematic of 5 X 5 router architecture

The schematic diagram of the proposed router architecture is shown in Figure 2 which consists of three blocks arbiter, buffer, and crossbar. The FIFO input buffer stores the input data temporarily. The arbiter receives requests from input buffers and allocates input data to requests and then gives grant signals to request initiators.

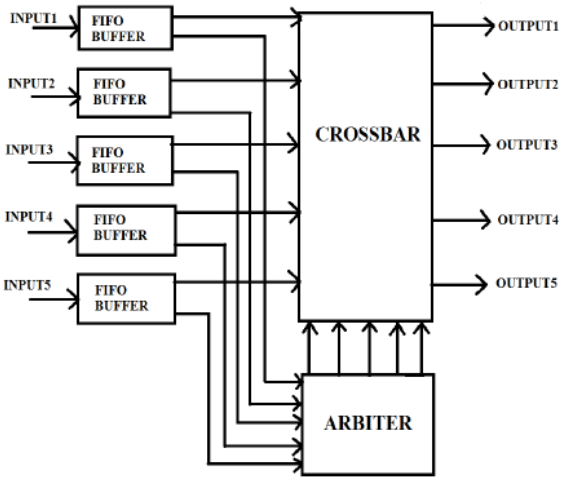

Figure 2. Schematic diagram of 5x5 Router architecture

This schematic diagram of the router architecture is used to establish the data transfer from source to destination. The connection or configuration is made between both with the central crosspoint matrix. The request signals from the arbiter are sending to the crossbar. This is used to select the priority signal for output port.

B. *Router architecture using buffering method*

The proposed NOC router architecture using buffering method is shown in Figure 3 Buffer used to store the data in flits. The data is transmitted in the form of packets. When the data in the form of packet is moved from source to its destination, it is sent on the network based on the routing decision taken by each router. At each router the packet is first collected and stored in buffer then the routing decisions are taken and channel arbitration is made by the control logic. Finally the granted packet crosses through the crossbar and reach to the next router.

This process repeats until the packet reaches to its destination. The proposed system shown in Fig.3 contains input ports, an arbiter and crossbar which are used to transmit the data from source to destination. Input ports contain FIFO buffer used to store and forward the data.

The output of the buffer is given to the arbiter as request signals. This is based on packet based wormhole routing method. The data is transmitted as multiple packets which are divided into number of flits. Each packet contains header flits, data flits and tail flits.

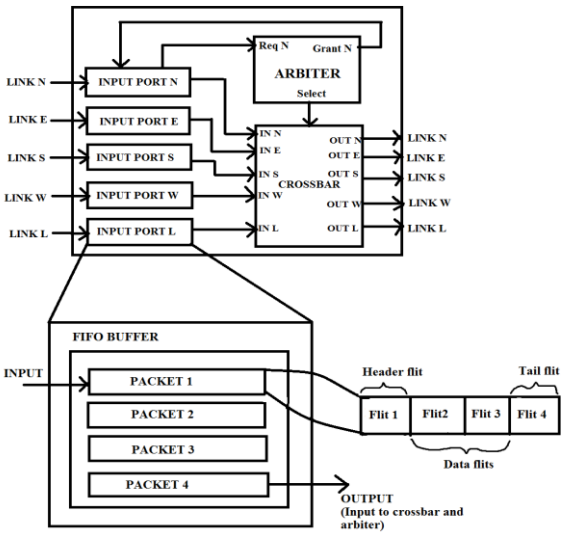

Figure 3. 5x5 Router architecture using buffering method

In FIFO logic, the write and read signal are used to transfer the data between sources to destination. In this FIFO, four write pointer and one read pointer used for simultaneous reading and writing operations. When the write signal is low, the data is written in to the buffer and if it is high, the read signal is enable. If the packet1 is filled, the read pointer starts the read operations even other packets are not filled. Simultaneous read and write operations are done in these packets.

C. *Wormhole packet based routing*

Routing methods are used to routing the data path from source to destination. These methods are used to transfer the data from source to destination without any collapse. If the head of packet is blocked during data transmission, the

router cannot transfer the packet any more. So the packet based wormhole routing method used to remove the blocking problem. The wormhole router splits the packet into several flits which can be transferred in a single transmission.

1) Packet format:

A message is divided into number of packets used to store the data. A packet format is shown in Figure 4. A packet is the basic unit of routing and the packet is divided into flits. A flit (flow control digit) is the basic unit of bandwidth and storage allocation. Therefore, flits do not contain any routing or sequence information and have to follow the route for the whole packet.

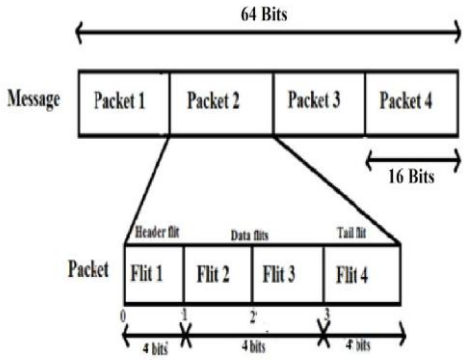

Figure 4. Packet format

A packet is composed of a head flit, body flits (data flits), and a tail flit. A head flit contains data and also allocates channel state for a packet, and a tail flit de-allocates it. The typical value of flits is between 16 bits to 512 bits. In proposed method, each packet has 16 bits and each flit has 4 bits. Total length of the message is 64 bits.

2) Buffering in packets

Buffering is necessary to store packet because the packets which arrive at nodes are unscheduled and should be multiplexed by control information. Three buffering cases happen in a NOC router. The first buffering condition is the output port can receive only one packet at a time when two packets arrive at the same output port at the same time. The second buffering condition is that the next stage of network is blocked and the packet in the previous stage cannot be routed into next router. Finally, a packet has to wait for arbitration time to get route path in a current router. The current router must store this packet.

IV. FIFO BUFFER IMPLEMENTAION

High performance parallel interfaces between independent clocks domains are implemented with first-in first-out memory called FIFO. This method helps to avoid deterioration and obsolescence. In proposed system, synchronous FIFO used to store and transfer the data from source to destination. In synchronous FIFO, write operation to the buffer and read operation from a same buffer are occurring in same clock domain.

A. Read and write operations

A FIFO can receive data until it is full and can be read until it is empty. A FIFO has Separate address pointers and data paths for reading and writing data. A FIFO can allow simultaneous reading and writing of data. The FIFO has a single clock port for both read and writes operations. For reset, both read pointer and write pointer set to zero. When write signal enable which increments the write pointer and the read signal enable which increments the read pointer. In write operation, data is write in to the buffer and in read operation data is read from the buffer. Read pointer and write pointer used to write and read the data in buffer. Packet contains number of flits which is used to store the data. Read and write pointer contains read and write addresses.

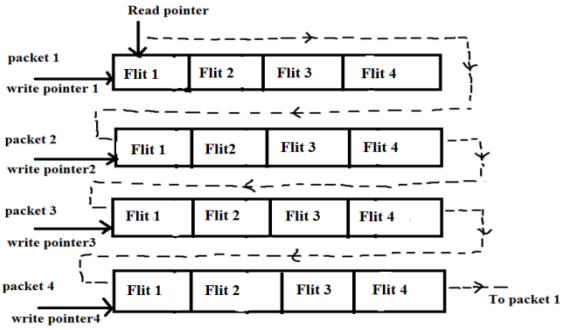

Figure 5. Read and write operations in FIFO

In proposed router architecture, FIFO buffer contains four write pointers and only one read pointer which is shown in Fig. 5. These four write pointers are used to write the data in the flits and read pointer read the data from the buffer. So the write and read operations are done simultaneously.

This method is used for efficient flow control and averting congestion. When the write signal is low, the data is written in to the buffer and if it is high, the read signal is enable. If the packet1 is filled, the read pointer starts the read

operations even other packets are not filled. Simultaneous read and write operations are done in these packets.

B. Flow control mechanisms

Flow control can be examined with the same method as the switching technique. A role of flow control mechanism is to decide which data is serviced first when a physical channel has many data to be transferred. In this proposed system, flow control is achieved by using read and writes operations in FIFO. First in First out (FIFO) method used to buffering the input ports.

By using wormhole packet based routing method, each packet is divided into four flits. Flit means flow control digit which is mainly used for efficient flow control. This is used to control the flow of data from source to destination. In this method, continuous data is transferred from source to destination by using four write pointers and one read pointer.

C. Congestion averting mechanism

Congestion control used to keep a set of senders from sending two much data into the network. In proposed system, wormhole routing method has efficient flow control and avoids congestion when compared with store & forward method and virtual cut through method. In router architecture, FIFO has four write pointers and only one read pointer which reduces the delay time. This is used to avoid the congestion during data transmission from source to destination.

In conventional router architecture, SRAM used to transfer the data with one write pointer and read pointer. In proposed router architecture, FIFO buffering method contains four write pointer and only one read pointer. Simultaneous read and write operations are done in this buffering method. This method mainly used to avoid congestion and produce efficient flow control.

D. Arbiter

Arbiters are electronic devices that allocate access to shared resources. Arbiter generates the grant signal based on input signal. It transmits the packet from input port to output port with priority. In proposed router architecture, arbiter is used to trap the source and destination address of input and output port. Arbiter generates the control signal according to priority so that crossbar switch transmits data from source to destination. This arbitration mechanism schedules a flit for transmission on the output path.

E. Crossbar

Crossbar switch is used to connect multiple inputs to multiple outputs in a matrix manner. Crossbar switch is central and important component in router architecture. It switches the data from the input port to the output port doing the essence of the router function. By making the speed of the crossbar faster more data packets can be sent.

V. RESULTS

The design of router architecture was coded by using verilog code. This design was compiled and simulated using Xilinx 14.2 software.

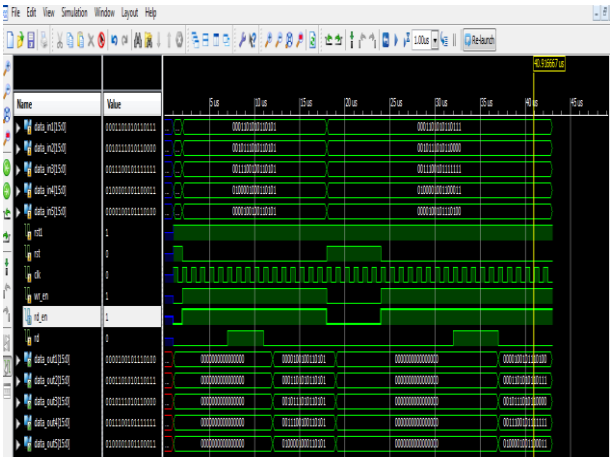

Figure 6. Simulation results of router architecture

The NOC router architecture is implemented by generic standard cell library. The design of 5x5 router architecture is synthesized in Application Specific Integrated Circuit (ASIC) design using cadence software with TSMC 18 1.0 (Taiwan Semiconductor Manufacturing Company) library (180 nm library). Area, power and timing report of router architecture is estimated by using cadence software.

Table 1 Comparison of existing and proposed router architectures

Methods	Area	Internal power	Switching power
For Existing Method	155269.7	13335808.3	15877262
For Proposed Method	75859.55	7213807.44	8394924.6
Percentage	68.7%	59%	61.6%

The latency can be described as the time passing from a source node to a destination node through intermediate nodes. The difference between first stage and second stage of the delay used to estimate the latency. The design of 5x5 router architecture has 27% less latency when compared with conventional router architecture. Latency is measured by using timing report of the router architecture.

The 5x5 router architecture consumes 69% less area when compared with existing router architecture. This is also consumes on average 59% less internal power and 62% less switching power. Proposed router reduces the latency up to 27% and also increases the throughput.

VI. CONCLUSION

FIFO buffering method used to transmit the data from source to destination in router architecture. FIFO buffering method has four write pointers and only one read pointer. This proposed router architecture used to transfer the data with efficient flow control and also avoid congestion.

This FIFO based router used to transfer the data efficiently without any interruption. The 5x5 router architecture consumes on average 69% less area, 59% less internal power and 62% less switching power for application specific integrated circuit design using cadence software with180 nm library when compared with conventional router architecture. FIFO buffering method indicated the better latency by 27% as compared with efficient dynamic virtual channel method. This buffering method used to increases throughput and also increases the performance of NOC router architecture.

REFERENCES

[1] Dumitriu V. and Khan G.N. (2009), 'Throughput-oriented NOC topology generation and analysis for high performance SOCs', IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 17, no. 10, pp. 1433–1446.

[2] Evripidou M. Nicopoulos C. Soteriou V. and Kim J. (2012), 'Virtualizing virtual channels for increased network-on-chip robustness and upgradeability', Proc. IEEE Comput. Soc. Annu. Symp. VLSI, Amherst, MA, USA, pp. 21–26.

[3] Liu J. and Delgado-Frias G.J. (2006), 'A shared self-compacting buffer for network-on-chip systems', Proc. 49th IEEE Int. Midwest Symp. Circuits Syst., San Juan, Puerto Rico, pp. 26–30.

[4] Masoud Oveis-Gharan and Khan N. (2015), 'Efficient dynamic virtual channel organization and Architecture for NoC Systems', IEEE transactions on very large scale integration (VLSI) systems.

[5] Nicopoulos C.A. Park D. J. Vijaykrishnan N. Yousif M.S. and Das C.R. (2006), 'ViChaR: A dynamic virtual channel regulator for network-on-chip routers', Proc. 39th Annu. IEEE/ACM Int. Symp. Microarchitecture, Orlando, FL, USA, pp. 333–346.

[6] Ramanujam R.S. Soteriou V. Lin B. and Peh L.S. (2010), 'Design of a high-throughput distributed shared-buffer NOC router', Proc. 4th IEEE/ACM Int. Symp. NOCS, Grenoble, France, pp. 69–78.

[7] Tamir Y. and Frazier G.L. (1992), 'Dynamically-allocated multi-queue buffers for VLSI communication switches', IEEE Trans. Comput., vol. 41, no. 6, pp. 725–737.

[8] Xu Y. Yang J. Zhao B. and Zhang Y. (2010), 'Simple virtual channel allocation for high throughput and high frequency on-chip routers', Proc. IEEE 16th Int. Symp. High Perform. Comput. Archit., Bengaluru, India, pp. 1–1.