Design and implementation of deadlock free NoC Router Architecture

Rohini¹, Dr.G.R.Udupi², G.A.Bidkar³

1 - Student of M. Tech in Industrial Electronics, 2-Principal, 3- Asst.Prof & HOD E&C Dept

KLS’s VDRIT, Haliyal affiliated to VTU, Belgaum, state: Karnataka, country:India

Abstract

As the feature size is continuously decreasing and chip integration is increasing, bus connections have become a dominating factor in determining the overall quality of a System on Chip. Long global wires also cause many design problems, such as routing traffic, scalability, latency and throughput. Network-on-Chip (NoCs) are an evolving architecture to be used in future systems, due to its increased performance, reusability and scalability. A NoC is a set of interconnected switches, with IP cores[1] connected to these switches. Routing plays an important role in determining latency and delay of router in NoC.

In this paper mesh based router architecture using XY Routing Algorithm and Pseudo XY|YX Routing Algorithm is presented. In this design we have used Verilog as the Hardware Descriptive Language and a single router is created. The delay and latency performance of interconnected routers have been analyzed through simulation and synthesis.

Keywords:ASIC, DSP, FIFO, FSM, NI, NoC, PE, RTL, SoC

1. INTRODUCTION

NoCs are an attempt to miniaturize concepts of large networks, and apply them to the system-on-chip (SoC) domain. NoCs use packets to send data from the source to the destination component, via an architecture that consists of routers and interconnection links. An example of a NoC system is illustrated in fig.1. The figure shows a NoC interconnection architecture with a mesh topology, consisting of several processing elements (PEs) connected together via routers and regular sized wires.

![Fig1.NoC architecture](image)

A PE in this case may be any component such as a microcontroller, DSPs, or a block of memory. A network interface (NI) at the boundary of each PE is used to packetize any data generated by the PE. This NI is connected to a router, which has buffers at its input to accept data packets from a PE or
from other routers. A crossbar switch inside the router is used to route the data packets from the input buffers to the appropriate output link, based on the address in the packet header.

This network can be modeled as a graph wherein nodes, processing elements and edges are the connective links of the processing elements.

In this paper designing and implementing NoC router are presented. In our design dead-lock free routing algorithm (XY routing) is applied to a 2D mesh topology and wormhole switching[2] is used along with data link layer flow control.

2. RELATED RESEARCH

In this section, we mention some literature analysis of various contributions to the NOC domain.

2.1 Topology

From the view of communication we know many topologies for NOC architecture. These include mesh, torus, ring, buttery, octagon and irregular interconnection networks [2], [3]. Various researchers have exploited these different NOC topologies for their NOC implementations.

- Adriahtante naina et al. proposed a tree based implementation of NOC [4], where each node in the tree behaves as a router in NOC;
- Pande et al. compared various network topologies for interconnection networks in terms of latency, throughput, and energy dissipation [5].

2.2 Router architecture

NOC architectures are based on packet-switching. It has led to efficient principles for design of routers for NOC [7]. Various designs and implementations of router architectures based on different routing strategies have been proposed in the literature.

- Wolkotte et al. proposed a circuit switched router architecture for NOC [8] but Dally and Towles proposed a packet switched router architecture.

2.3 Routing protocol

Routing algorithms can also be defined based on their implementation: lookup table and Finite State Machine (FSM). Lookup table routing algorithms are more popular in implementing. They are executed in software, where a lookup table is stored in every node. FSM based routing algorithms may be designed either in software or in hardware.

In [10] Evaluation of Routing Algorithms on Mesh Based NoCs is illustrated that Deterministic (dimension-order) and adaptive routing algorithms different topologies was presented.

Mello et al. researched the performance of minimal routing protocol in NOC [11]. They concluded that the minimal routing provided better results than adaptive routing for on-chip communications, as the adaptive routing concentrates on the traffic in the center of the NOC.
2.4 Switching technique
Al-Tawil et al. provided a well-structured survey of Wormhole Routing techniques and its comparison with other switching techniques [12].

2.5 Flow control
Zeferino et al. used handshaking signals in their SoCIN NOC implementation [11].

3. UTILIZED ROUTING ALGORITHM

The XY algorithm is a distributed routing, where each packet carries the address destination and routing decisions are made in each router by looking up the destination addresses in a routing table. When a packet arrives at the input port of a router, the routing logic is executed to determine the packets output port based on its destination address.

This XY algorithm as shown in fig 2 uses current coordinates and computes the offset address. Depending upon whether the offset is positive or negative the channel direction is incremented or decremented.

It is deterministic algorithm which packet takes routing in one dimension and continues till this packet attains desired coordinate in that dimension. After that, routing is continued to do the same procedure in the other dimension. This method warrants no deadlock to occur.

Algorithm: XY Routing for 2-D Meshes

Inputs: Coordinates of current node \((X_{current}, Y_{current})\)

and destination node \((X_{dest}, Y_{dest})\)

Output: Selected output \(Channel\)

Procedure:

\[X_{offset} := X_{dest} - X_{current};\]

\[Y_{offset} := Y_{dest} - Y_{current};\]

if \(X_{offset} < 0\) then

\(Channel := X-;\)
endif

if \(X_{offset} > 0\) then

\(Channel := X+;\)
endif

if \(X_{offset} = 0\) and \(Y_{offset} < 0\) then

\(Channel := Y-;\)
endif

if \(X_{offset} = 0\) and \(Y_{offset} > 0\) then

\(Channel := Y+;\)
endif

if \(X_{offset} = 0\) and \(Y_{offset} = 0\) then

\(Channel := Internal;\)
endif

Fig 2. XY algorithm for 2D meshes

4. UTILIZED SWITCHING TECHNIQUE

The NoC switching strategy determines how data flows through the routers in the network. PEs generate messages that are partitioned into possibly several data packets. A packet is further divided into multiple flits (flow control unit). Each flit is made up of one or more phits (physical units). A phit is a unit of data that is transferred on a link in one clock cycle as shown in figure 3.
In this design we have used packet switching to transfer data, particularly wormhole switching where a flit from a packet is forwarded to the receiving router. If there is insufficient space in the next router to store the entire packet, parts of the packet are distributed among two or more routers.

5. FLOW CONTROL

In Handshaking Signal Based Flow Control, a VALID in_val signal is sent whenever a sender transmits any it. The receiver acknowledges by asserting a VALID signal after consuming the data it.

In this design T-Error is used that can detect faults by making use of a second delayed clock at every buffer stage. The delayed clock re samples the input data to detect any inconsistencies and then emits a VALID control signal. A resynchronzation stage is added between the end of the link and the receiving switch, to handle the offset between the original and delayed clocks.

6. CLOCKING SCHEMES

In NoCs, several different clocking schemes are possible, such as synchronous, mesochronous, pleisochronous, and asynchronous. In the fully synchronous case, a single global clock is distributed to synchronize the entire chip. The clock signal arrives simultaneously at the local flip-flops of routers, nodes, and buffered links all over the chip. To overcome this problem, multiple clock domains are used. In the mesochronous case, local clocks are derived from a global clock that has been distributed all across the chip. All synchronous modules in a mesochronous system use the same clock source, but the phase between clock signals in different modules may differ due to an unbalanced global clock network. In this design we have used Globally Asynchronous and Locally Synchronous clocking scheme.

7. ROUTER DESIGN

The mesh based router[3] consists of multi ports such as east, west, north, south and local port. It also has a central cross point matrix. Inside each port there are two channels input and output. Data packet is sent from one port moves in to the input channel of router by which it is forwarded to the output channel of the other port.

Each input channel and output channel has its own decoding logic which increases the performance of the router. Buffers are used at all ports to store the data for a short time span. The store and forward method is used here for data transmission. Control logic is present to make decisions to grant access to a port request. In this way communication is established between input and output ports. The transfer of data from source to destination is called packet switching mechanism where the flit size is 8 bits.
Fig 4. Block diagram of Mesh based router

7.1 input channel

Fig 5. Input channel block diagram

As shown in fig 5 each router has one input channel at each port and each has its control and decoding logic. It consists of main three parts

- FIFO
- FSM and
- Routing logic.

FIFO is used as input buffer as temporary data storage. The FIFO is of 8 bits and depth is of 16 bits. The first 8 bits are the header which consists of coordinates of destination path.

The status of FIFO decides the communication can start or not. If the FIFO is empty the data can be written in it and communication can start.

If FIFO is full, data can be read or can be forwarded to its destination router. Inside the router Grant/acknowledgement signals are used to access the FIFO. The read and write operation of FIFO is controlled by FSM. FSM controls the read and write operation of FIFO according to its status.

If FIFO is empty and having space to store, then FSM will send a request to output channel of other port, if grant signal is received by it then read operation starts and continues until grant signal goes low. Thus empty status of indicates the end of communication. XY logic is the deterministic logic which analyses the header of data and sends it to its destination port. The first four bits of the header are the coordinates of destination port.

In XY logic a comparator is used which compares the header of the data to the locally stored X and Y coordinate and sends the packet according to its destination address. Let the coordinates stored in header be Hx and Hy and locally stored coordinates be X and Y. So according to XY logic if Hx > X then packet will move to east port otherwise move to west port. If Hx = X then Y coordinate is compared, If Hy > Y then packet will move to north port otherwise move to south port, when Hy = Y then packet will move to local port. Thus in this way XY logic will send the packet to the output channel of its destination port.
7.2 Output channel

As shown in fig 6 each Output channel consists of three parts i.e.
- FIFO,
- FSM and
- Arbiter

FIFO and FSM are same as in input channel but in place of XY logic, arbiter is used in output channel. FIFO in output channels used as output buffer to store the data temporarily.

If the FIFO is empty the data can be written and communication can start. If FIFO is full, data can be read or can be forwarded to its destination router. Inside the router the Grant/Acknowledgement signals are used to access the FIFO. The read and write operation of FIFO is controlled by FSM. FSM controls the read and write operation of FIFO according to its status. If FIFO is empty, FSM will give acknowledgement signal with respect to the request sent by the input channel of the router.

Using XILINX ISE 13.1 simulator with Verilog as the HDL we created the modules for each blocks of the input and output channel. Each module is verified for its correct operation. Then we combined input and output channels and is called a **Router**, the same code is simulated and we got the output of it. Then we duplicated the module of router for 4X4 topology and the data packet is sent from west port of one router to north, south and east port of other routers.

8. EXPECTED RESULT

Two router architectures are implemented in structural Register-Transfer-Level (RTL) Verilog and then synthesized in a Xilinx 13.1.

Router design is simulated in ISIM 13.1 by creating a dummy environment around its surroundings.

8.1 Synthesized result

The synthesized result [4] of two router module is obtained in terms of input module and output module and then integrated two modules to combine complete router.

1. Input module of Router 2
2. Minimum period: 4.612ns
3. Output module of router 1 Minimum period: 4.612ns
4. Output module of router 2 Minimum period: 5.223ns
5. Router Minimum period: 5.223ns

8.2 Simulation result:

The purpose of the simulation is to determine the router latency and delay in terms of clock cycles.

8.2.1 Test Cases
1. with zero loads: Only one input port receives flits from core or router.
2. with load (without conflict): More than one input port receive different flits from core.
   - Testing with zero load is performed by sending flits to only one input port of any direction. Suppose if flits are sent to north...
port then there will be no flit at any other Input port.

We can see from Simulation results in the form of clock cycles were data packet (head flit, body flit and tail flit) is sent from west port to east port are shown in the fig 7. It has latency of 7 clock cycle.

![Fig 7: data packet is sent from west port to east port](image1.png)

Testing with load and no output port conflict is performed by sending flits to more than one Input Port but different flits access different output ports. Suppose if one it is sent to north input port and another it is sent to south input port then there will be no chance of getting same output port by these flits. In another way both have different destination output ports. Simulation results in the form of clock cycles were data packet (head flit, body flit and tail flit) is sent from west port to east port and south to north port are shown in the fig 8. It has latency of 7 clock cycle at each port.

![Fig 8: data packet is sent from west port to east port and south to north port](image2.png)

### 9. CONCLUSION

In this paper, router architecture using XY algorithm is implemented using HDL called Verilog at RTL level. Architecture is synthesized using Xilinx and simulated using ISIM 13.1 for evaluating latency and delay of two routers. The simulation result show that both has same latency there is same latency but in zero load case and with load (no Conflict), but using XY-YX algorithm zero clock delay is achieved and hardware usage is 9.569.56

### 10. REFERENCES


[14] “Interconnection Networks” By Jose DUATO, Sudhakar YALAMANCKILLI