# ADVANCED FIFO DATA TRANSMISSION WITH SHARED BUFFER ROUTERS IN BI-DIRECTIONAL NOC

**Y.YASODHA KUMARI [1] SK.JOHN SYDHA [2]**

1 E.C.E Student, Dept of E.C.E, RK College Of Engineering, Kethanakonda, A.P., India

2 Assistant Professor, Dept of E.C.E, RK College Of Engineering, Kethanakonda, A.P., India

## ABSTRACT

Router architecture plays a central role in the performance of a Network on Chip (NoC). Router consists of buffers which are dedicated to their input or output ports for temporarily storing packets during the times of congestion. Unfortunately, significant portion of the router area and power is consumed by the buffers alone. While running some tested traffic patterns, however, not all input ports of routers have incoming packets needed to be transferred simultaneously. Therefore, a large number of buffer queues in the network are empty whereas the other queues are mostly busy.

This observation has led to the design of Router Architecture with shared queues which maximize the buffer utilization by sharing the multiple buffer queues among input ports. Buffers become more efficient by sharing queues, hence the router is able to achieve higher throughput when the network load becomes heavy. On the other side, at light traffic load, this router is designed to achieve low latency by allowing packets to effectively bypass these shared queues. A new adaptive weighted XY routing algorithm for eight port router Architecture is proposed in order to decrease the latency of the network on chip router.

**Keywords:** Network on Chip (NOC), Router, Data Transmission, Data Buffer utilization.

## INTRODUCTION

System on chip (SOC) is a complex interconnection of various functional elements. It creates communication bottleneck in the gigabit communication due to its bus-based architecture. Thus, there was need of system that explicit modularity and parallelism, network on chip possess many such attractive properties and solve the problem of communication bottleneck. It basically works on the idea of interconnection of cores using on chip network.

The communication on network on chip is carried out by means of router, so for implementing better NOC, the router should be efficiently design. This router supports four parallel connections at the same time. It uses store and forward type of flow control and Fsm Controller deterministic routing which improves the performance of router. The switching mechanism used here is packet switching which is generally used on network on chip. In packet switching the data the data transfers in the form of packets between cooperating routers and independent routing decision is taken. The store and forward flow mechanism is best because it does not reserve channels and thus does not lead to idle physical channels. The arbiter is of rotating priority scheme so that every channel once get chance to transfer its data. In this router both input and output buffering is used so that congestion can be avoided at both sides.

A router is a device that forwards data packets across computer networks. Routers perform the data "traffic direction" functions on the Internet. A router is a microprocessor- controlled device that is connected to two or more data lines from different networks. When a data packet comes in on one of the lines .The router reads the address information in the packet to determine its ultimate destination. Then, using information in its routing table, it directs the packet to the next network on its journey. The router is a" Four Port Network Router" has a one input port from which the packet enters. It has three output ports where the packet is driven out. Packet contains 3 parts. They are Header, data and frame check sequence. Packet width is 8 bits and the length of the packet can be between 1 bytes to 63 bytes. Packet header contains three fields DA and length. Destination address (DA) of the packet is of 8 bits. The switch drives the packet to respective ports based on this destination address of the packets. Each output port has 8-bit unique port address. If the destination address of the packet matches the port address, then switch drives the packet to the output port, Length of the data is of 8 bits and from 0 to 63. Length is measured in terms of bytes. Data should be in terms of bytes and can take anything. Frame check sequence contains the security check of the packet. It is calculated over the header and data. A data packet is typically passed from router to router through the networks of the Internet until it gets to its destination computer. Routers also perform other tasks such as translating the data transmission protocol of the packet to the appropriate protocol of the next network.

## LITERATURE SURVEY

A router is a device that forwards data packet between computer networks, creating an overlay internetwork. A router is connected to two or more data lines from different networks. When a data packet comes in one of the lines, the router reads the address information in the packet to determine its ultimate destination. Then, using information in its routing table or routing policy, it directs the packet to the next network on its journey. Routers perform the "traffic directing" functions on the Internet. A data packet is typically forwarded from one router to another through the networks that constitute the internetwork until it reaches its destination node.

Routers may also be used to connect two or more logical groups of computer devices known as subnets, each with a different sub-network address. The subnets addresses recorded in the router do not necessarily map directly to the physical interface connections. Forwarding an IP datagram generally requires the router to choose the address and relevant interface of the next-hop router or (for the final hop) the destination host.

In Transmission Control Protocol/Internet Protocol (TCP/IP) networking, routers are used to interconnect the hardware and software used on different physical network segments called subnets. Routers are also used to forward IP packets between each of the subnets. Determine the physical layout of your network, including the number of routers and subnets you need, before proceeding with the instructions in this guide. Routers may provide connectivity within enterprises, between enterprises and the Internet, and between internet service providers (ISPs) networks. The largest routers (such as the CiscoCRS-1 or JuniperT1600) interconnect the various ISPs, or may be used in large enterprise networks. Smaller routers usually provide connectivity for typical home and office networks. Other networking solutions may be provided by a backbone Wireless Distribution System (WDS), which avoids the costs of introducing networking cables into buildings. All sizes of routers may be found inside enterprises. The most powerful routers are usually found in ISPs, academic and research facilities. Large businesses may also need more powerful routers to cope with ever increasing demands of internet data traffic.

**ROUTER DESIGN SPECIFICATION**

Router is a packet based protocol. Router drives the incoming packet which comes from the input port to output ports based on the address contained in the packet. The router has a one input port from which the packet enters. It has three output ports where the packet is driven out. The router has an active low synchronous input resetn which resets the router.

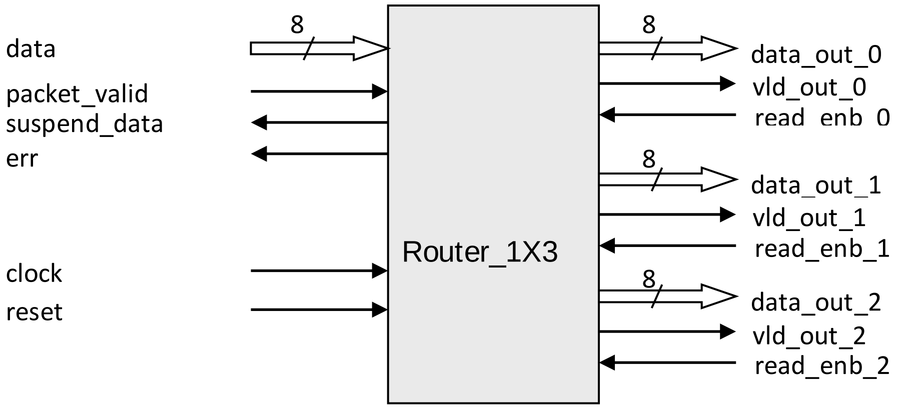


Figure 3.1- Block Diagram of Four Port Router

Data packet moves in to the input channel of one port of router by which it is forwarded to the output channel of other port. Each input channel and output channel has its own decoding logic which increases the performance of the router. Buffers are present at all ports to store the data temporarily. The buffering method used here is store and forward. Control logic is present to make arbitration decisions. Thus communication is established between input and output ports. According to the destination path of data packet, control bit lines of FSM are set. The movement of data from source to destination is called switching mechanism The packet switching mechanism is used here, in which the flit size is 8 bits .Thus the packet size varies from 0 bits to 8 bits. A detailed explanation of Design is as follow.

**Packet Format:**

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.

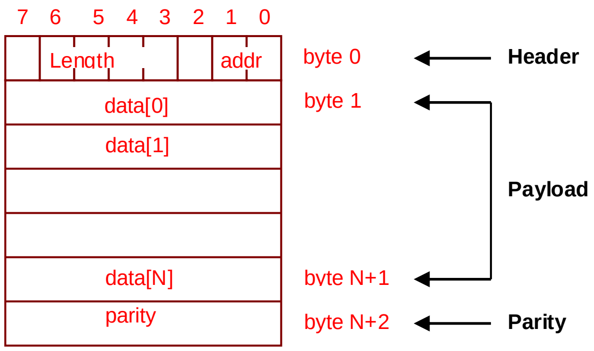


Figure 3.2- Data Packet Format

## FOUR PORT ROUTER ARCHITECTURE

The Four Router Design is done by using of the three blocks .the blocks are 8-Bit Register, Router controller and output block. the router controller is design by using FSM design and the output block consists of threefifo's combined together the fifo's are store packet of data and when u want to data that time the data read from the FIFO's. In this router design has three outputs that is 8-Bit size and one 8_bit data port it using to drive the data into router we are using the global clock and reset signals, and the err signal and suspended data signals are output's of the router .the FSM controller gives the err and suspended_data_insignals .this functions are discussed clearly in below FSM description.

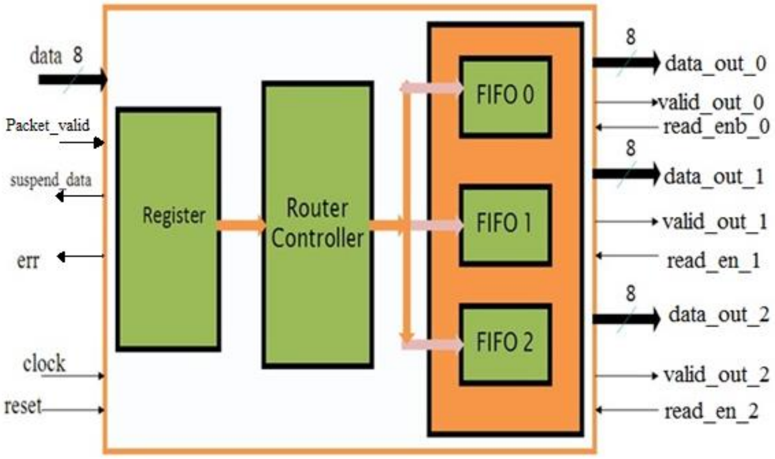


Figure-4.1 Four Port Router Architecture

The router_reg module contains the status, data and parity registers for the Network router_1x3.

These registers are latched to new status or input data through the control signals provided by the fsm_router.

There are 3 FIFO for each output port, which stores the data coming from input port based on the control signals provided by fsm_router module.

The fsm_router block provides the control signals to the fifo, and router_reg module. The Router blocks Diagram shown below fig…

Router blocks are

- Register
- Router controller(FSM)
- FIFO Output Block

First data byte i.e., header is latched inside the internal register first_byte when detect_add and packet_valid signals are high, So that it can be latched to output dout when lfd_state signal goes high.

Then the input data i.e., payload is latched to output dout if ld_state signal is high and fifo_full is low.

Then the input data i.e., parity is latched to output dout if ld_state signal is high and fifo_full is low.

The input data is latched to internal register full_state_byte when ld_state and fifo_full are high; this full_state_byte data is latched inside the output dout when laf_state goes high.

Internal parity register stores the parity calculated for packet data, when packet is transmitted fully, the internal calculated parity is compared with parity byte of the packet. An error signal is generated if packet parity is not equal to the calculated parity.

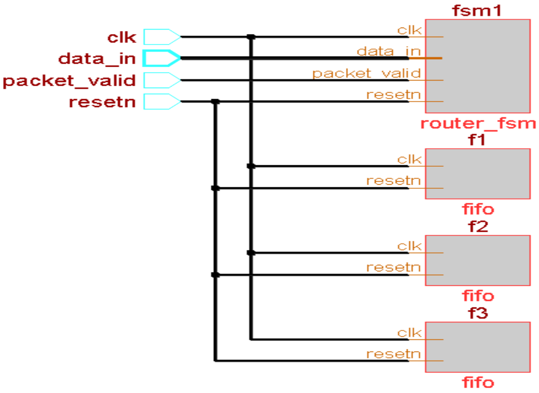


Figure 4.3-Register block synchronization

In the above figure register block is synchronize with the fsm to latch input data to it.Here,clk,resetn signals are synchronous with the entire module.

Eg: We are giving packet data as input to it and making read single (re1, re2, re3) as high w.r.t input first data byte of the packet. Thereceiving data is driven to the Router Controller for reaching its destination port. Which has 11 input pins (data_in [7:0],packet_valid ,clk, reset).

Eg: data_in=8’b10101010, clk,reset,packet_valid are HIGH

## DESIGN ASPECTS AND APPROACH

For designing it we need finite state machine which controls all signals and we need states for controlling.The states should be for initializing data,wait for data until empty, loading data to respective port,iffifo is full w.r.t port it have to wait until empty,for parity byte loading and parity calculation.For initializing decode_state,for data waiting till empty wait_till_empty,for loading data load_data0, load_data1, load_data2,for checking fifo is full fifo_full0, fifo_full1, fifo_full2,for loading parity load_parity0, load_parity1, load_parity2 and for parity calculation check_parity0, check_parity01,check_parity2 totally we need 14 states having 4-bit width.But 4-bit it will give 16 states i.e full case.So, we are using only 14 states i.e parallel case. States represent as parameter keyword by this state value can’t change throughout the design.For this design block we written in verilog code in behavioral model.

First we verifying whether it is reseting or not and when packet_valid signal is the data(10101010) is driving to w.r.t output port(fifo2) when it making low data is loading to load parity for parity calculation.And also verified whether it is switching to another port with another data(10101000) when reseting and also driven some bytes of data to the port.Testbench is written in verilog code.

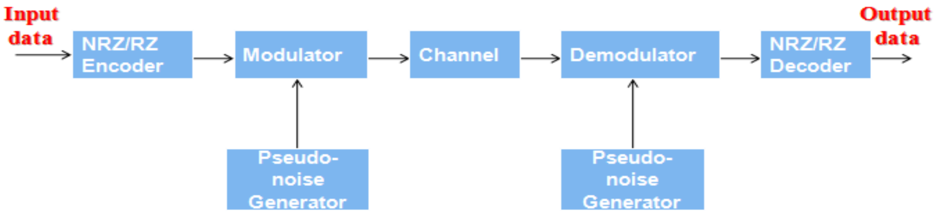
**FIFO**:

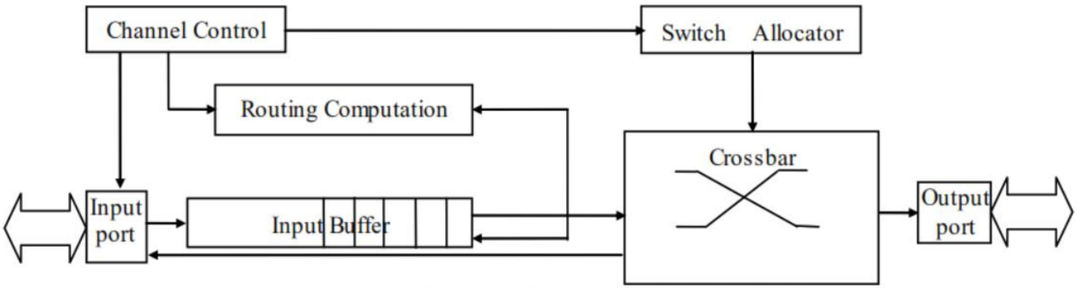
It is 8-bit width and 16-bit depth.Forfifo full or empty we are taking fifo_full and fifo_emptysignals.For the status of full or empty of fifo we need a internal counter for counting it locations upto 16 locations it mean it is 4-bit wide.Input signals are data_in(8-bit),we,re,clk,resetn and output signals are data_ou(8-bit),t,fifo_empty,fifo_full.Data is driven when write and not fifo full and it read when read and not fifoempty.RTL code it is written in verilog code in behavioral model.

It is verified by giving 16 bytes of data in data_in ,we is high then fifo_full becomes high.When it is high data can't be written into it.We get output in data_out and re is high it given all 16 bytes of data which we had driven after that fifo_empty is high then we can't read data and we also verified when both we and re signals are high it is written in verilog code.

**Top module**:

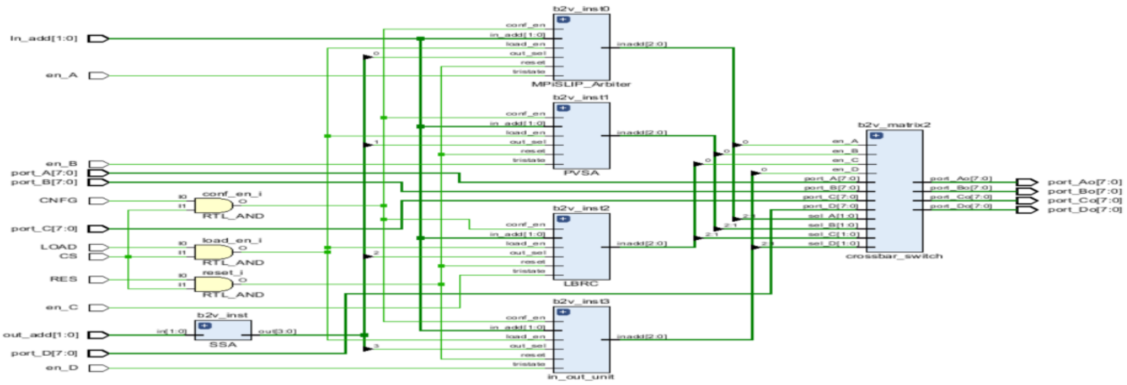
In this module we synchronized register,routercontroller,fifo blocks for that we calling all the modules with .name instance declaration and developing a design plan with these blocks as data_in is driven to register block, routing data to respective port decision taken by router controller and we are considering three fifo as output port. Same clock is given to all blocks i.e clock is synchronous. Code it is written in verilog code.



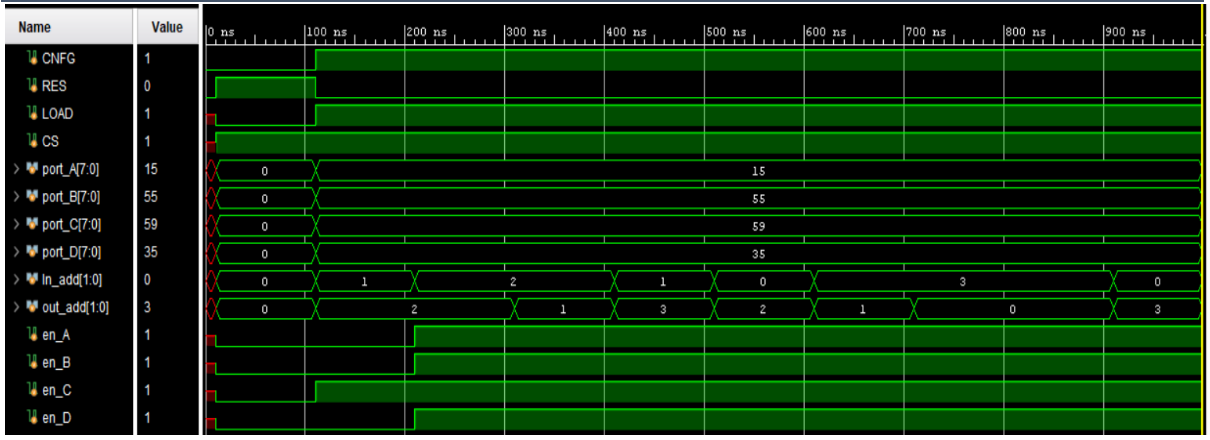


Typical structure of Bi-directional NoC

**SIMULATION AND SYNTHSIS RESULTS**

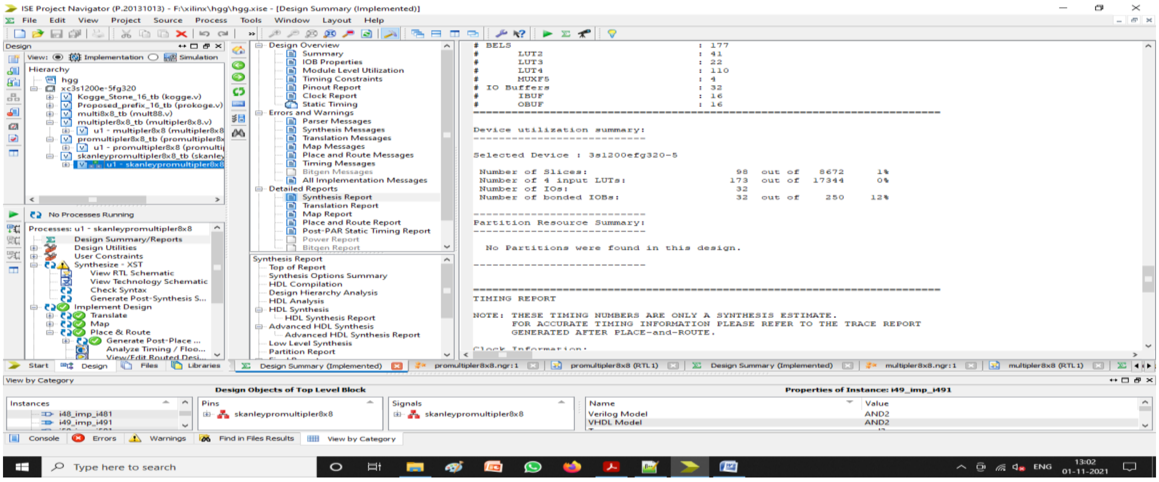


RTL Schematic diagram



**Simulation Result**

**Estimation of power Analysis**



```
#  BELS                                   : 177
#       LUT2                              : 41
#       LUT3                              : 22
#       LUT4                              : 110
#       MUXF5                             : 4
#  IO Buffers                             : 32
#       IBUF                              : 16
#       OBUF                              : 16
=========================================================================

Device utilization summary:
---------------------------

Selected Device : 3s1200efg320-5

 Number of Slices:                      98  out of   8672     1%
 Number of 4 input LUTs:               173  out of  17344     0%
 Number of IOs:                         32
 Number of bonded IOBs:                 32  out of    250    12%

---------------------------
Partition Resource Summary:
---------------------------

  No Partitions were found in this design.

---------------------------


=========================================================================
TIMING REPORT

NOTE: THESE TIMING NUMBERS ARE ONLY A SYNTHESIS ESTIMATE.
      FOR ACCURATE TIMING INFORMATION PLEASE REFER TO THE TRACE REPORT
      GENERATED AFTER PLACE-and-ROUTE.
```

| Name | Value |
|---|---|
| Verilog Model | AND2 |
| VHDL Model | AND2 |

**Area of Utilization Report**

## CONCLUSION

An advanced FIFO structure based NoC is simulated and synthesized in Xilinx 14.7 ISE and implemented Vertex-6 FPGA device to analyze the performance in terms of occupied area, latency, power consumption and throughput. Single router is designed initially and then designed mesh based NoC to realize the memory utilization of FPGA. Fig.4 indicates that Register Transfer Level (RTL) schematic of single NoC router which is composed with input and output ports, arbiter, crossbar and channel control modules. The figure also describes the utilizations in terms of memory units each component individually. Each module of NoC designed using Verilog Hardware Description Language (HDL) separately and integrated as one module. An advanced queued buffer is designed both typical NoC and Bi-directional NoC thereby comparing both designs easily. The simulation results are analyzed area utilization in terms of occupied number of slices registers, LUT-FF pairs and slice registers), latency in terms of delay, Maximum operating frequency, power consumption in terms of dynamic power dissipation, memory utilization in terms of number of RAMs, and finally, throughput in terms of flits per sec., node.describes the performance of NoC router in terms area, delay and power consumption which are obtained by implemented proposed in FPGA configuration.

## REFERENCES

[1] Anh T. Tran and Bevan M. Bass, “Achieving High Performance On-Chip Networks With Shared Buffer Routers”, IEEE Transactions on VLSI Systems,Vol 22, No.6,June 2014

[2] K.Priyanka, D.Gopinath& G. Rajaiah, Design and Implementation of Network Topology for Digitally Enhanced Networks, IJMETMR, Volume No: 2 (2015), Issue No: 7 (July) ,

[3] William J Dally and B.Towels, “Route Packets, not Wires: On-Chip interconnection Networks,” DAC2001

[4] William J Dally, “Virtual channel Flow Control”, IEEE Transactions on Parallel Distribution. Systems., Vol 3, No 2, Mar 1992

[5] T.Moscibroda and O.Mutlu, “A case for bufferless routing in Onchip Networks” ISCA, June 2009

[6]A.T. Tran, D.N. Truong and B.M.Baas, “A GALS many-core heterogenous DSP platform with sourcesynchronous on-chip interconnection network,” in Proc. May 2009

[7] E.Begine, “An asynchronous power aware and adaptive NoC based circuit”, IEEE J. Solid State Circuits, vol 44, Apr. 2009.