International Journal for Modern Trends in Science and Technology, 9(06): 184-194, 2023

ISSN: 2455-3778 online
DOI: https://doi.org/10.46501/IJMTST0906027
Available online at: http://www.ijmtst.com/vol9issue06.html

Design and Verification of 1*3 Router

Pratiksha Musale[1], Pooja Thakre[2], Yogita Raut[2]

[1]PG Scholar, Department of Electronic and Telecommunication Engineering, Swaminarayan Siddhanta Institute of Technology, Nagpur, Maharashtra, India.
[2]Assistant Professor, Department of Electronic and Telecommunication Engineering, Swaminarayan Siddhanta Institute of Technology, Nagpur, Maharashtra, India

To Cite this Article
Pratiksha Musale, Pooja Thakre, Yogita Raut. Design and Verification of 1*3 Router. International Journal for Modern Trends in Science and Technology 2023, 9(06), pp. 184-194. https://doi.org/10.46501/IJMTST0906027

Article Info
Received: 23 May 2023; Accepted: 13 June 2023; Published: 19 June 2023.

ABSTRACT

Routing is the process of moving a packet of data from source to destination and enables messages to pass from one computer to another and eventually reach the target machine. A router is a networking device that forwards data packets between computer networks. It is connected to two or more data lines from different networks (as opposed to a network switch, which connects data lines from one single network). This project, mainly emphasizes the study of router devices, their top-level architecture, and how various sub-modules of routers i.e. Register, FIFO, FSM, and Synchronizer are synthesized, simulated, and finally connected to their top module. The router is a device that transfers information in the form of packets. The typical protocol has been defined for proper communication of information which validates that the data reaches a destination as defined by the source. TCP/ IP protocol is widely used in communication networks. Router and routing are defined as part of a network layer of TCP/IP. The design of the 1X3 router can be extended without extensive changes in the code. Code coverage and functional coverage are used to check the functionality of code and how efficiently the code is written. Functional coverage is user defined matrix whereas code coverage is tool specified matrix. So, Verification of code is important before it is implemented in hardware as it can prevent loss of money, time, and resources of an organization.

Keywords: Router, Synchronizer, Protocol, Computer Networks, TCP/IP, Register, FIFO, FSM

1. INTRODUCTION

Information gathering, processing and distribution are the key to technological communication networks. A communication networks can be defined as a collection of several hardware and software devices. It provides various technology which makes the transmission easier and prominent. The basic examples can be the LANs and WANs. Routing is one way to transmit data from source to destination in form of packets. For multiple users, to ensure proper communication worldwide, certain standards are being followed which include Open System Interconnect (OSI) model and Transmission Control Protocol/ Internet Protocol (TCP/IP) model. OSI model is not much in use but still valid. On the other hand, the TCP/IP model is widely used. Also, OSI model is reference for TCP/IP model. This report focuses on the architecture of TCP/IP model, which includes The Internet Layer, The Transport Layer, The Application Layer and the Host-to-network Layer. From the layers of TCP/IP model, the router belongs to The Internet layer. Router forwards data packets between computer networks. The internet layer helps in driving an arriving

packet to an output destination. It reads the report information in the form of packets and decides the client location. Similarly, it forwards the packet to the next destination. This transmission requires at least two networks, one for source and other for the client. The router provides the direction in the internet traffic.

TCP/IP Model Computer networks are a big trap for proper communication worldwide. To control the network, there is a need for a system which can regulate the transmission of information and messages. These messages are traveling in the form of data packets. So, it is necessary to connect those messy networks in a flawless way. This is the foremost concern of designing any protocol [2]. Also, data/message should be safe and intact, even if the source and destination machines are damaged or not working properly.This design can be achieved with the TCP/IP reference model.

The Internet Layer: The work of this layer is to make sure that the host injects data information into any network and let them drive independently to the destination. If the arrival of data is not the same as it has been sent, it is job of higher layers to arrange them. Internet Layer defines the proper format i.e. INTERNET PROTOCOL. So, it can be said that the main functionality of the internet layer is to transport IP packets to the end point. Proper transmission or routing of data helps in avoiding data congestion.

The Transport Layer: This layer provides peer connection between source and destination to carry out the conversation. Two different protocols are provided in this layer which is Transmission Control Protocol which is used in connection-oriented communications and other is User Datagram Protocol which is used in connectionless communications.

The Application Layer: On top of the transport layer the next layer which comes is application layer. It contains all the high-level protocol, few of them are: -

a) Virtual terminal (TELNET): It allows the user to login to the different machine and worksthere.
b) File transfer protocol: It allows the data to move from one system to other effectively.
c) Electronic mail: It is one kind of file transfer.
d) Domain Name System: DNS is used for mapping the host names onto their network addresses.

The Host- to- Network Layer: There is very less discussion in papers about this layer except that the source must connect to a network using some protocol.

Routeris a device that helps in transmitting information through the networks in the form of data packets. It is connected to at least two networks, commonly two LANs or WANs or a LAN 4 and its ISPs network and is located at gateways, the place where two or more networks connect. The incoming packets reach to the output channel by determining address fields present in the header of the packet. To regulate further transmission and path determination of forwarding packets, it uses header and forwarding tables. This resultsin controlling internet traffic which helps in finding the best path communication between several hosts

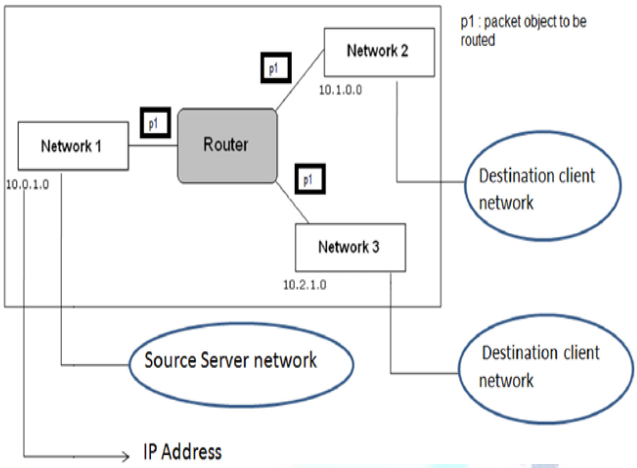


Fig. 1.2a. Router network

Router 1*3 Features

Packet Routing: The packet is driven from the input port and it routed to any one output port, based on the address of the destination network.

Parity Checking: An error detection technique that tests the integrity of digital data being transmitted between server & client. This technique ensure that the data transmitted by the server network is received by the client network without getting corrupted.

Reset: It is an active low synchronous input that resets the router. Under reset condition, the router FIFOs are made empty and the valid out signals goes low indicating that no valid packet is detected on the output data bus.

Sending packet:

The packet consists of 3 parts Header, Payload, and Parity such that each id of 8 bit width and length of payload can be extended between 1 byte to 63 byte.

Reading packet:

Active high input signal are reading the packet through output data bus .

2. LITERATURE REVIEW

The sheer volume of data available on the internet is staggering. It's fascinating to watch how data gets transferred from one computer to the next. It doesn't have to follow a set course; in fact, it could change in the middle of a computer discussion. A router is a computer networking device that sends data packets from one network to the next. The router scans the address information in a data packet to determine its eventual destination when it arrives on one of the lines. The packet is then directed to the next network on its path using information from its routing table or routing policy. As a result, overlay internetwork is constructed. On the Internet, routers provide "traffic directing" responsibilities. A router is a computer networking device that sends data packets from one network to the next. If you have a large image that you wish to email to a friend or post to a website, it could be made up of tens of millions of bits of 1s and 0s, which are too many to transfer in a single packet. Because the image is data on a computer, it may be instantly broken down into hundreds or even thousands of tiny pieces known as packets. The origin and destination internet addresses are included in each packet. Routers, which are internet-connected machines that operate as traffic supervisors, keep packets moving efficiently over networks. Individual packets may follow a different path over the internet if one route is clogged. To deal with all of these scenarios, we'll need a device with a more advanced design. And to check that design, we'll need a reliable verification mechanism.

- Most home users may want to set up a LAN (Local Area Network) or WLAN (wireless LAN) and connect all computers to the Internet without having to pay a full broadband subscription service to their ISP for each computer on the network.
- In many instances, an ISP will allow you to use a router and connect multiple computers to a single Internet connection and pay a nominal fee for each additional computer sharing the connection. This is when home users will want to look at smaller routers, often called broadband routers that enable two or more computers to share an Internet connection. Within a business or organization, you may need to connect multiple computers to the Internet, but also want to connect multiple private networks, not all routers are created equal since their job will differ slightly from network to network.
- Additionally, you may look at a piece of hardware and not even realize it is a router. What defines a router is not its shape, colour, size, or manufacturer, but its job function of routing data packets between computers. A cable modem, which routes data between your PC and your ISP can be considered a router.

In its most basic form, a router could simply be one of two computers running the Windows 98 (or higher) operating system connected using ICS (Internet Connection Sharing). In this scenario, the computer that is connected to the Internet is acting as the router for the second computer to obtain its Internet connection. Going a step up from ICS, we have a category of hardware routers that used to perform the same basic task as ICS, albeit with more features and functions often called broadband or Internet connection sharing routers, these routers allow you to share one Internet connection with multiple computers. Broadband or ICS routers will look a bit different depending on the manufacturer or brand, but wired routers are generally small box-shaped hardware devices with ports on the front or back into which you will plug each computer along with a port to plug in your broadband modem.

3. PROPOSED METHODOLOGY

As discuss above router is a device that helps in traffic controlling in the internet network. A message packet is forwarded after one router to another through the systems that constitute the internetwork till it reaches its destination nodes. Fig show the Top-Level Block.

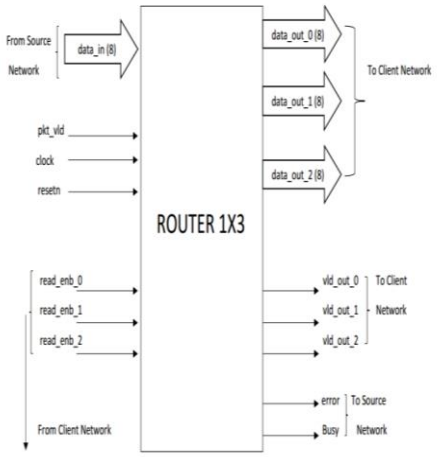


Fig. router Top Model

Packet routing is done on bases of destination details, the packet is sent to output port from output port. - Parity Checking: After decoding the address of client, it is important that client should receive the pure form of data send the pure. Instead of checking each bit, parity is checked to give information about corrupted or pure data. - Reset: - Reset is active low input signal. It helps in resetting the FIFOs and indicate no valid data is available at output buses. –

Sending Packet. Design under test input procedure

• Clock, pkt_vld, busy and error are active high signals.

• Pkt_vld signal indicates that new packet has been started and asserted with header byte in input data bus.

• Header byte is the first byte of data which also has the address of FIFO. With the help of header byte router gets to know where packet should be routed.

• before the parity byte started pkt_vld will be deasserted, which indicates the packet is complete.

• Busy signal will be asserted at next rising edge of clock after pkt_vld and data is asserted.

• Testbench cannot started new packet when busy signal is asserted.

• Error is there to indicate the packet is corrupted. i.e. packet which is sent by server is not the same what client has received. Packet sending mechanism - Reading Packet: **Output procedure details**:

• Vld_out read_enb and data_out is active high signals

• Vld_out signals will be asserted according to FIFO address in header byte.

• It will be asserted at next rising edge of clock after payload byte starts on data_out.

• It helps in identifying that data is present in output bus.

• data_out signals are also asserted according to FIFO selection.

• For each FIFO different data_out signals are present to show its data.

• Read_enb signals reads data from data_out signals and assert accordingly.

• Read_enb signals are also connected to internal reset which indicates data is late.

• 30 clock cycles are assigned for internal reset.

• If it doesn't receive any packet which in 30 clock cycle.

• That will be the case time out and that resets the FIFO.

• Data_out bus will be in high impedance state when time out condition occurs. Packet reading mechanism .

Router Interface Below is the table for router interface signals Table 1: Router Interface Signals .

Clock	Active High Clocking Event
Pkt_valid	Pkt_valid is an active high input signal that detects an arrival of a new packet from a source network
resetn	Active low synchronous reset
Data_in	8 bit input data bus that transmits the packet from source network to router
Read_enb_0	Active high input signal for reading the packet through output data bus data_out_0
Read_enb_1	Active high input signal for reading the packet through output data bus data_out_0
Read_enb_2	Active high input signal for reading the packet through output data bus data_out_0
Data_out_0	8 bit output data bus that transmits the packet from the router to destination client network 1

Data_out_1	8 bit output data bus that transmits the packet from the router to destination client network 2
Data_out_2	8 bit output data bus that transmits the packet from the router to destination client network 3
Vld_out_0	Active high signal that detects a valid byte is available for destination client network 1
Vld_out_1	Active high signal that detects a valid byte is available for destination client network 2
Vld_out_2	Active high signal that detects a valid byte is available for destination client network 3
busy	Active high signal that detects a busy state for the router that stops accepting any new byte
error	Active high signal that detects the mismatch between packet parity and internal parity

Router- Output Protocol

Test Bench Note:

All output signals are active high and are synchronized to the rising edge of the clock.

Each output port data_out_X (data_out_0, data_out_1, data_out_2) is internally buffered by a FIFO of sixe 16*9.

The router asserts the vld_out_X (vld_out_0, vld_out_1, vld_out_2) signal when vaild data appears on the vld_out_X(data_out_0, data_out_1, data_out_2) output bus. This is a signal to the receiver's client which indicates that data is available on a particular output data bus.

The packet reciver will then wait until it has enough space to hold the butes of the packet and then respond with the assertion of the read_enb_X (read_enb_0, read_enb_1, read_enb_2) signal.

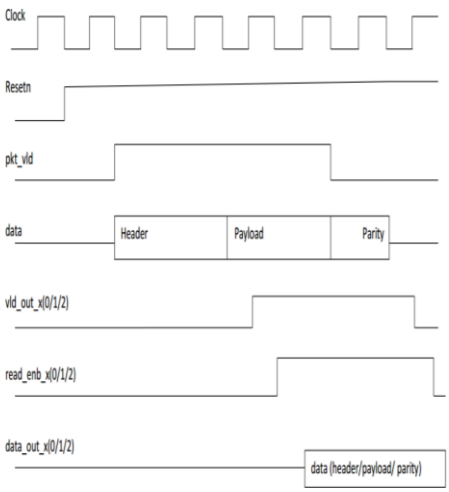
The read_enb_X (read_enb_0, read_enb_1, read_enb_2) input signal can be asserted on the falling clock edge in when data are read from the data_out_X (data_out_0, data_out_1, data_out_2) bus.

The read_enb_X (read_enb_0, read_enb_1 or read_enb_2) must be asserted within 30 clock cycles of vld_out_0, vld_out_1,vld_out_2)being asserted else time out occurs, which resets like the FIFO

The data_out_X bus will be tri-stated during a scenario when a packets byte is lost due to time out condition.

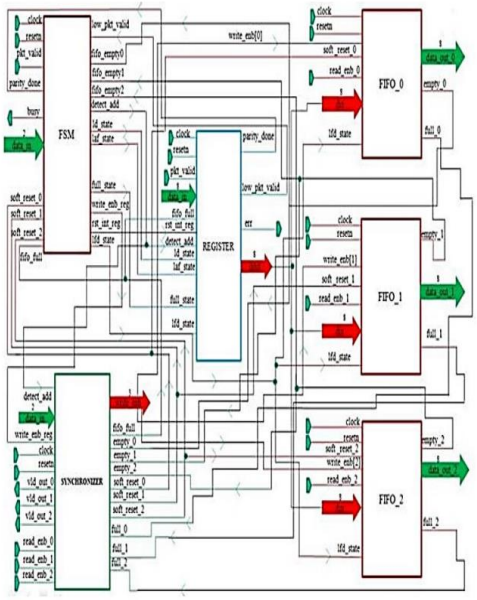
Vld_out read_enb and data_out is active high signals

- Vld_out signals will be asserted according to FIFO address in header byte.
- It will be asserted at next rising edge of clock after payload byte starts on data_out.
- It helps in identifying that data is present in output bus.
- data_out signals are also asserted according to FIFO selection.
- For each FIFO different data_out signals are present to show its data.
- Read_enb signals reads data from data_out signals and assert accordingly.
- Read_enb signals are also connected to internal reset which indicates data is late.
- 30 clock cycles are assigned for internal reset.
- If it doesn't receive any packet which in 30 clock cycle.
- That will be the case time out and that resets the FIFO.
- Data_out bus will be in high impedance state when time out condition occurs.



Router Architecture

The top-level block module consists of 6 sub blocks: a) 3 FIFO b) Synchronizer c) Register d) Finite State Machine



1

Router: FIFO

- FIFO helps in sending the packet through buses. As our design is 1X3 ,3 FIFOs of 16X9 have been used, which means each FIFO has depth of 16 bytes and width of 9 bits. The operation of FIFO is connected to clock and is reset according to resetn signal.
- There are 2 FIFOs used in the router design. Each FIFO is of 9 bits width and of 16 locations as depth. The FIFO Works on the system clock and is reset with a synchronous active low reset.
- The FIFO is also internally reset by an internal reset signal soft_reset. Soft_reset is an active high signal which is generated by the SYNCHRONIZER block during the time out state of the ROUTER.
- IF reset is low then full =0, empty=1 and data-out=0.
- The FIFO memory size is 16*9. The extra bit in the data width is appended in order to detect the header byte.
- The lfd_state detects the header byte of packet. The 9th bit is 1 for header byte and 0 for remaining bytes.
- WRITE OPERATION :
- Signal data_in is sampled at the rising edge of the clock when write_enb is high.
- Write operation only takes place when FIFO is not full in order to avoid over_run condition.
- **READ OPERATION:**
- The data is read from data_out at rising edge of the clock, when read_enb is high .
- Read operation only takes place when FIFO is not empty in order to avoid under_run condition.
- During the read operation when a header byte is read , an internal counter is loaded with the payload length of the packet plus "1" (parity byte) and starts decrementing every clock cycle till it reaches 0.the counter holds 0 till it is reloaded back with a new packet payload length.
- During the time out state, full = 0, empty =1.
- Data_out driven to HIGH impedance state under 2 scenarios:
- When the FIFO memory is read completely (Header + Payload + Parity)
- Under the time out state of the router.
- Full –FIFO status which indicates that all the locations inside FIFO have been written.
- Empty –FIFO status which indicates that the locations of FIFO have been read and made empty.
- Read and write operation can be done simultaneously.

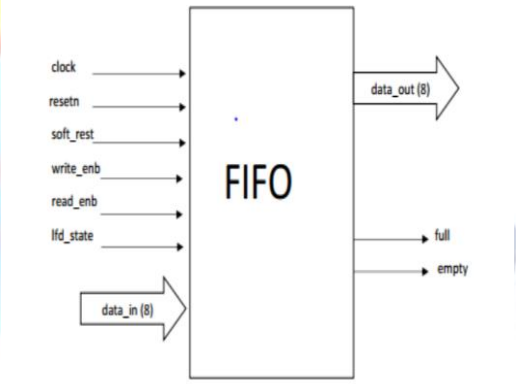
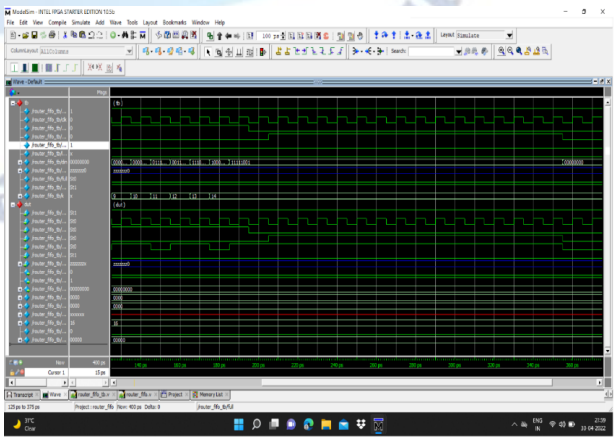
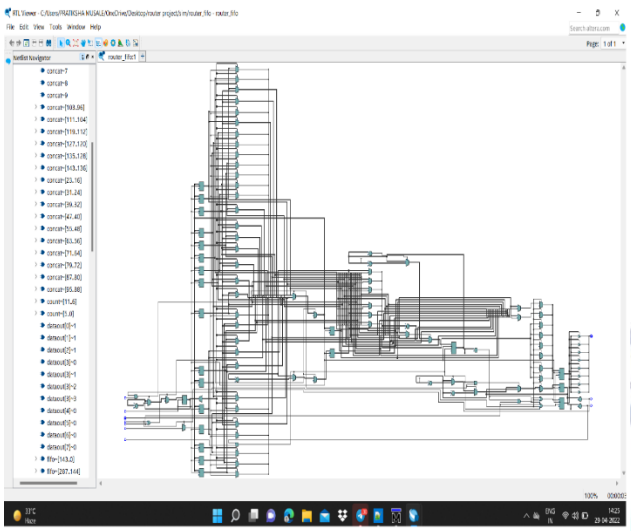
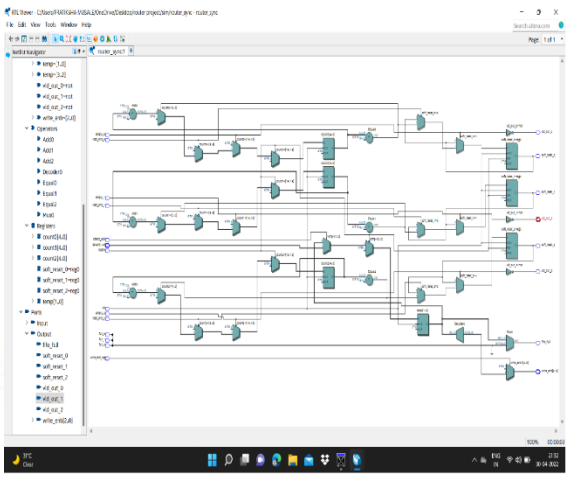


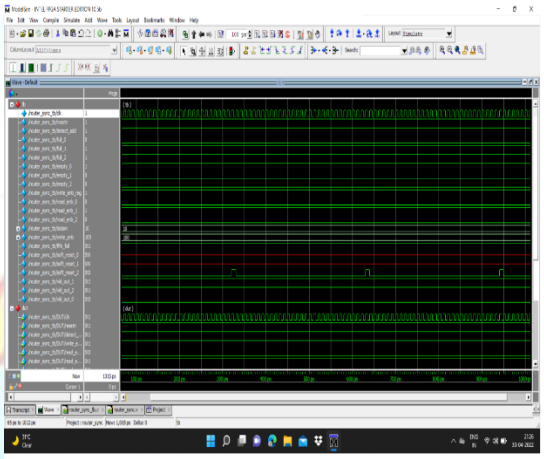
Fig. FIFO

FIFO Waveform









ROUTER SYNCHRONIZER

Functionality:

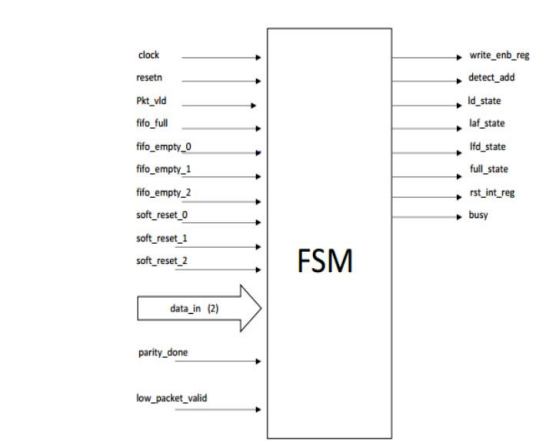
This module Provides synchronization between router FSM and router FIFO modules. It provides faithful communication between the single input port and three output ports.

- Detect_and and data_in signals are used to select a FIFO till a packet outing is over for the selected FIFO.
- Signal fifo_full signal is asserted based on full status of FIFO_0 or FIFO_1 or FIFO_2.
- If data_in =2'b00 then fifo_full = full_0
- If data_in = 2'b01 then fifo_full = full_1
- If data_in = 2'b10 then fifo_full = full_2 else fifo_full=0
- The signal vld_out_x signal is generated based on empty status of the FIFO as shown below
- Vld_out_0 = ~empty_0
- Vld_out_1 = ~empty_1
- Vld_out_2 = ~empty_2
- The write_enb_reg signal is used to generate write_enb signal for the write operation of the selected FIFO.
- There are 3 internal reset signals (soft_reset_0, soft_reset_1, soft_reset_2) for each of the FIFO respectively. The respective internal reset signal goes high if read_enb_x (read_enb_0, read_enb_1 or read_enb_2) is not asserted within 30 clock cycles of the vld_out_x (vld_out_0, vld_out_1, vld_out_2) being asserted respectively.

ROUTER- FSM (CONTROLLER)

Functionality:

The FSM module acts as the controller unit for the router. The module generates all the regulating signals when any new packet is received by the router. These control signals are used by other design components in order to transfer the packet to the output port.



STATE-DECODE_ADDRESS

- This is the initial reset state.
- Signal detect_add is asserted in this state which is used to detect an incoming packet. It is also used to latch the first byte as a header byte.

STATE-LOAD_FIRST_DATA

- Signal lfd_state is asserted in this state which is used to load the first data byte to the FIFO.
- Signal busy is also asserted in this state so that header byte that is already latched doesn't update to a new value for the current packet.
- This state is changed to LOAD_DATA state unconditionally in the next clock cycle.

STATE-LAOD_DATA

- In this state the signal ld_state is asserted which is used to load the payload data to the FIFO.
- Signal busy is deassrted in this state, so that Router can receive new data from input source every clock cycle.
- Signal write_enb_reg is asserted in this state in order to write the packet information (Header+ payload+ parity) to the selected FIFO.
- This state transits to LOAD_PARITY state when pkt_valid goes low and to FIFO_FULL_STATE when FIFO is full.

STATE-LOAD_PARITY

- In this state last byte is latched which is the parity byte.
- It goes unconditionally to the state CHECK_PARITY_ERROR
- Signal busy is asserted so that router doesn't accepts any new data.
- Write_enb_reg is made high for laching the parity byte to FIFO.

STATE-FIFO_FULL_STATE

- Busy signal is made high and write _enb_reg signal is made low.
- Signal full_state is asserted which detects the FIFO full state.

STATE-LOAD_AFTER_FULL

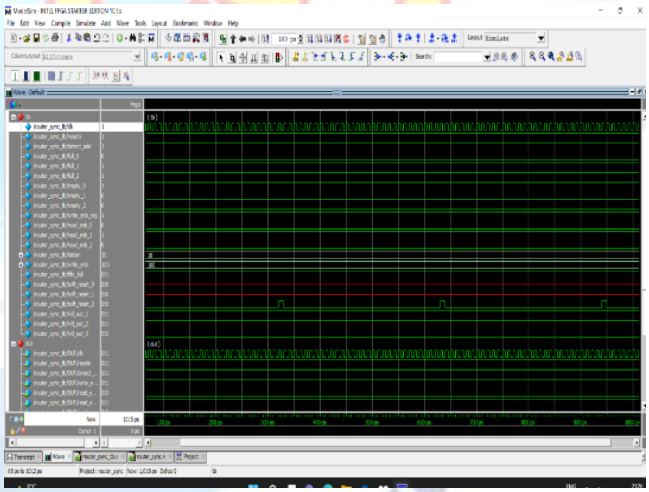
- In this state laf_state signal is asserted which is used to latch the data after FIFO_FULL_STATE.
- Signal busy & write_enb_reg is asserted.
- In check for parity_done signal and if it is high shows that LOAD_PARITY state has been detected and it goes to the state DECODE_ADDRESS.
- If low_pkt_valid is high it goes to LOAD_PARITY state otherwise it goes back to the LOAD_DATA state.

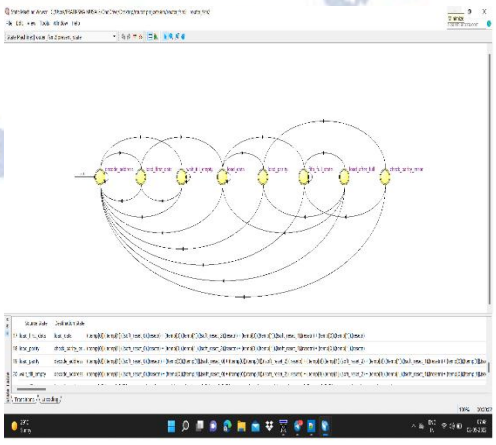
STATE-WAIT_TILL_EMPTY

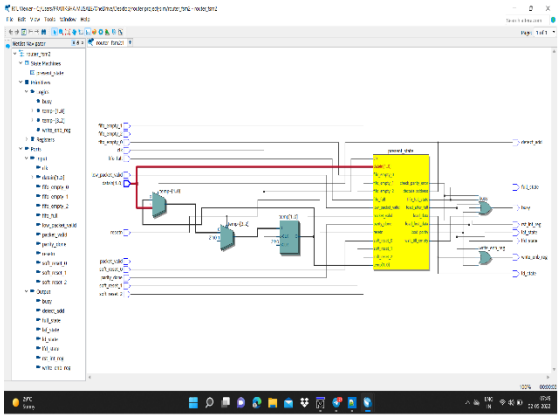
- Busy signal is made high and write_enb_reg signal is made low.

STATE-CHECK_PARITY_ERROR

- In this state rst_int_reg signal is generated, which is used to reset low_pkt_valid signal.
- This state changes to DECODE_ADDRESS when FIFO is not full and to FIFO_FULL_STATE when FIFO is full.
- Busy is asserted in this state.
- The soft-reset signal should be used in the FSM in such a way that the current state should change back to "DECODE_ADDRESS" state only for timeout situation of the current transmitted packet.



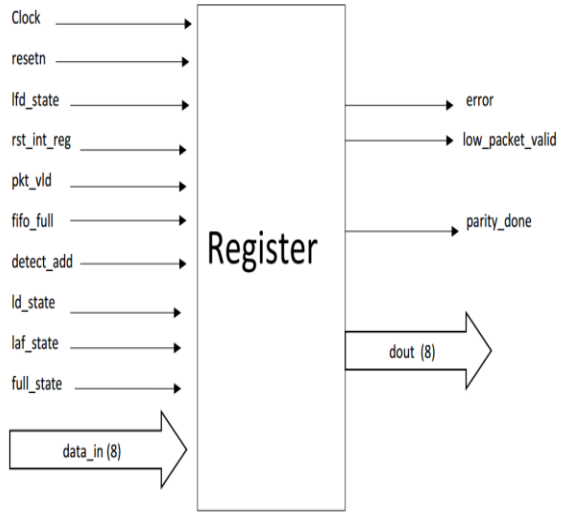




ROUTER-REGISTER

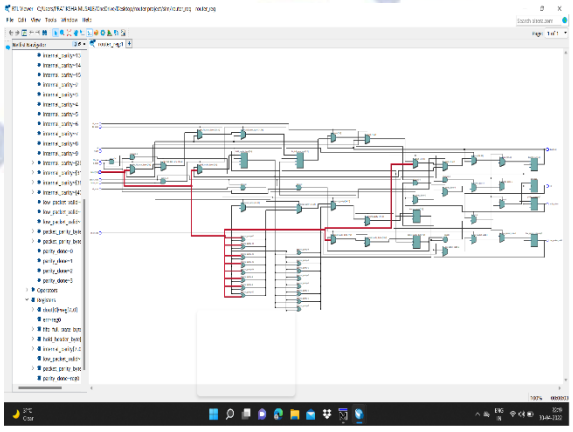
Functionality:

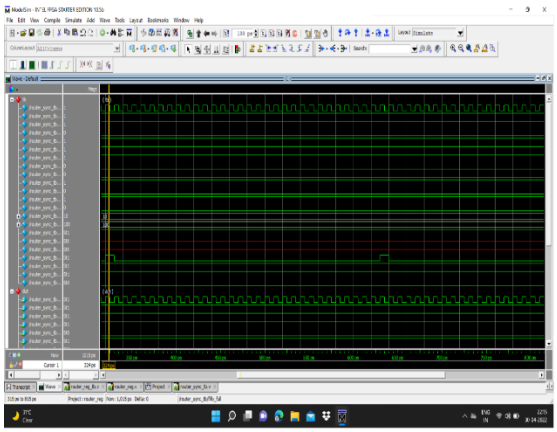
This module implements 4 internal registers in order to hold header byte, FIFO fill state byte, internal parity and packet parity byte.

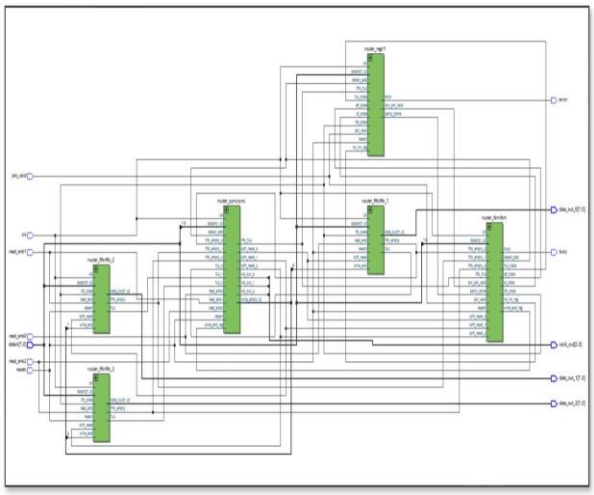


All the register in this module are latched on the rising edge of clock.

- If resetn is low then the signal (dout, err, parity_done and low_pkt_valid) are made low.
- The signal parity_done is high under the following conditions
- When signal ld_state is high and signal (fifo_full and pkt_valid) are low.
- When signal laf_state and low_pkt_valid both are high and the previous value of parity_done is low.
- Rst_int_reg signal is used to reset low_pkt_valid signal.
- Detect_add signal is used to reset parity_done signal.
- Signal low_pkt_valid is high when ld_state is high and pkt_valid is low. Low_pkt_valid shows that pkt_valid for current packet has been deasserted.
- First data byte i.e. Header is latched inside an internal register when detect_add and pkt_valid signals are high. This dada is latched to the output dout when lfd_state signal goes high.
- Then signal data_in i.e. payload is latched to dout if ld_state and fifo _full are high. This data is latched to output dout when laf_state goes high.
- Full_state is used to calculate internal parity.
- Another internal register is used to store internal parity for parity matching. internal parity is calculated using the bit-wise xor operation between header byte, payload byte and previous parity values as shown below:
- Parity_reg = parity_reg_previos ^ header_byte-----t1 clock cycle.
- Parity_reg = parity_reg_previos ^ header_byte 1-----t2 clock cycle
- Parity_reg = parity_reg_previos ^ header_byte 2-----t3 clock cycle
- Parity_reg = parity_reg_previos ^ header_byte n-----tn clock cycle (last payload byte)
- The error is calculated only after packet parity is loaded and goes high if the packet parity doesn't match with the internal parity.



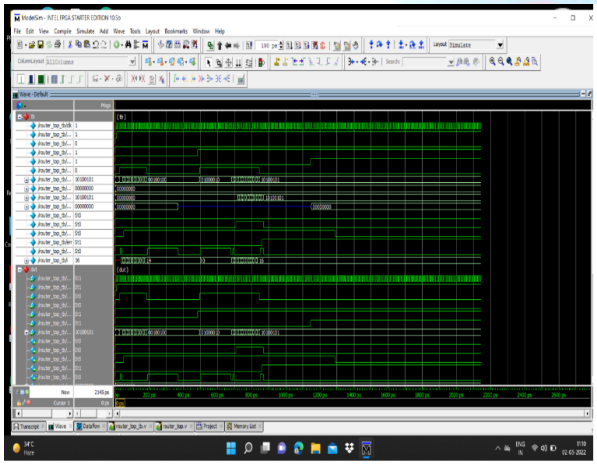




Router TOP MODULE

Top module which is integration of sub modules. IN Top module we have to use structural modelling.

Below is the output waveform in which first two data packets are shown. In this upon getting new packet pkt_valid signal is getting high and on next posedge of clock, signal busy is asserted with high. Signal error is not high as internal parity is matching with packet parity in these packets. Signal read_enb for FIFO 2 is getting high which means FIFO 2 will read the first packet and for second packet, read_enb signal of FIFO 1 is high



4. RESULTS & DISCUSSION

I have designed network ROUTER and I have verified the functionality of the ROUTER. The Verilog code for all the modules is written and instantiated in the top module and simulation is done using the modelsim and different packet lengths of size 14 and 16 are observed in waveforms and it is synthesized and the rtl schematic is obtained. The Router1X3 is a successfully designed by using the verilog hardware description language. During the simulation in the modelsim and synthesis in the quartus prime software, many coding flaws ,errors and warnings are discovered and rectified .Packets with a long & different payload length like 4 byte packet length,14 bytes packet length,16 bytes packet length are generated and Full FIFO state i.e (observing busy signal) & which is a good packet packet and observed read a read/write operation are happening at the same time ,different bad packets or a packet that has been corrupted were derived from the testbench in order to assess the design's reliability. And according to Questasim's coverage report, it was working flawlessly, with 100% FSM state coverage, 82.81 percent FSM transition coverage, 95.4 percent toggle coverage, and 94.31 percent statement coverage. It is proved that the router 1x3 is designed very efficiently with lower constant by using the available resources.

Conflict of interest statement

Authors declare that they do not have any conflict of interest.

REFERENCES

[1] M. Pirretti, G. M. Link, R. R. Brooks, N. Vijaykrishnan, M. Kandemir, and M. J. Irwin, "F@ult tolerant algorithms$ for network-on-chip interconnect," Proceedin$. IEEE Computer society Annual Symposium on VLSI, pp.46-51, Feb. 2004.

[2] Salminen, E., Kulmala, A., & Hamalainen, T. D. (2008).

[3] Cota, É., de Morais Amory, A., & Lubaszew$ki, M. S. (2012). NoC Basics. In Reliability, Availability and Serviceability of Networks-on-Chip (pp. 11-24). $pringer US.

[4] Swapna, S., Swain, A. K., & Mahapatra, K. K. (2012, December). Design and analysis of f5 port router for network on chip. In Microelectronics and Electronics (PrimeA$ia), 2012 Asia Pacific Conference on Postgraduate Re$earch in (pp. 51-55). IEEE.

[5] Choudhari, E. M., & Dakhole, P. K. (2014, April). Design and verification of five port router for network on chip. In Communications and Signal Processing (ICCSP), 2014 International Conference on (pp. 607- 611). IE

[6] Gopal N., Router 1x3 RTL Design Journal of Engineering.

[7] Graham, B. (2001). TCP/IP addressing. 2nd ed. San Diego, Calif.: Morgan Kaufmann.

[8] www.google.com/fifo-theory-Verilog

[9] www.wikipedia.com

[10] Verilog HDL: - Samir Palnitkar

[11] Supriyareddy Patil, Dr.Biswaraj Gadgay "Design of Router 1*3 using HDL", International Journal for Research in Applied science & Engineering Technology, Vol. 7, issue x, pp. 329-332, oct. 2019

[12] Bhaskar. J, A Verilog HDL Primer, Publisher: Syndicate Publications