International Journal of Applied Engineering Research
ISSN 0973-4562 Volume 10, Number 3 (2015) pp. 5339-5349
© Research India Publications
http://www.ripublication.com

Design And Implementation Of High Throughput Asynchronous Router For Multi-Core On Board Execution

M. Kamaraju, Raghunath Mandla

*Professor & HOD: Department of ECE, M.Tech: Department of ECE,
Gudlavalleru Engineering College, Gudlavalleru Engineering College,
Gudlavalleru, Gudlavalleru, India,India,
madduraju@yahoo.com, raghunath.mandla@gmail.com*

Abstract

In this paper, introduces a new architecture which deals with development of permutation network for multi core on-board execution is proposed, the network employs fixed priority permutation network and variable priority permutation network with different modes of transmission, which has been designed to provide the flexibility of the network. Counter is used in order to provide the possible combination of the priorities and it provides the control signals for input module which choose the transmitter whose data has to be transmitted. In previous work, regarding permutation network only variable priority permutation network is designed in which the priority will be allocated dynamically and it cannot give the optimal solution as each transmitter execution time is limited. Therefore, it has to wait for longer instance of time. Timing controller is used in the architecture to generate control signals for the clock time for each transition of the transmitter which can optimize the performance of the system. This architecture results in completing each work in single chance which can be observed in simulation results. This architecture has been designed by using Xilinx 13.2 and implemented using FPGAs like virtex4 and Spartan3E. Compared the resulting frequency with different FPGA devices. At the cost of area and power, this design optimizes the performance of the system.

Keywords*:* multi core on-board execution, permutation network, FPGA, clock controller, counter.

Introduction

Multi processor system on chip is developed mainly with the intention of parallel processing of a System with many number of processors connected through a network and uses same clock for all components of the system. CLOS network is one of the three stage network interconnections used for ATM’s developed by Charles clos [3].

As the sizes of the chip is decreasing, and the internal chip utilization increases, the interconnects becoming loss media to transmit data. Higher incidence of data errors is caused due to Crosstalk, electro-magnetic interference, and switching noise. Critical path delays have become very long as compared to gate delays which cause synchronization problems between different processors.

Packet-switching approaches is used in previous methods to obtain guaranteed through-put in permuted traffic under random permutation, employing circuit-switching mechanism in on-chip network with a dynamic path-setup scheme gives better results. It can manage the objective of arranging the path dynamically for crash-free permuted data. It can avoid the excessive amount of required queuing buffer.

A. *Backtracking probing path-setup scheme for end-to-end flow control scheme:* End-to-end communication, used in the circuit-switching approach, is discussed to give a clear idea of the backtracking probing path-setup scheme used with the BW switch. In this approach communication takes place in three phases: path-setup (or probing), transmission, and release phases respectively. In path-setup phase, with the help of destination port address communication path establishes between input and output modules. When overhead occurs, the probe header needs to backtrack and has to inform or send control signal that indicates the expected line is busy and search for a new path to establish communication, this increases the efficiency of the data transfer. i.e., the path is set up by backtracking probing path-setup scheme. When the probe header reaches its destination, the source will receive an acknowledgement signal. In transmitter phase, source synchronous data transmits by the source via the set up path to destination. To represent answer (ans) signal two bits are used. If ans is 01, it denotes that receiver is ready to accept data from the source, if ans is 10, it denotes tat the receiver path is busy and forces the probe header to backtrack to find possible paths. If the ans is 11, it denotes that the receiver is not able to receive data (e.g., due to being busy, or cause of overflow at the receiving buffer). During the setup and the transmission phases, Req is set to 1 When ans is 00 the probe header continuously advances until it reaches the destination. Then, the destination returns 01 to the source wrapper. When the source wrapper receives ―01, it immediately starts to transmit the pipelined data.

B. *Automatic track setup scheme:* Automatic track setup scheme with circuit switching mechanism is better idea with reduces the overhead and improves the latency. It can arrange the track through which the data has to be transmitted; there will be conflicts in arranging the track as the track might be busy or it cannot transmit any data due to suspend state caused by two or more request arrival at a time. Therefore with the help of previously available algorithm which is named as exhausted profitable backtracking (EPB) is used to track the probe in the network. This can find the required track other than the tracks with blocked or busy states [2].

C. *Timing controller:* The relevant pipelined circuit switching schemes with automatic track setup scheme is not the ultimate design as it not flexible and the data transmission in this method cannot satisfy the applications. Therefore a design which gives the optimal result for the industrial applications, the proposed design with timing controller plays major role in this project and by using circuit switching approach fixed priority permutation network is also designed in such a way one can switch the design from fixed priority permutation network to variable priority permutation network and vice versa.

D. The remaining paper is organized as follows: section 2 deals with proposed system, Working is discussed in section 3, section 4 describing about Results, and finally Conclusion and References are presented.

Proposed Design

The main motto of the proposed architecture is to design circuit switching for both fixed priority and variable priority to facilitate the flexibility of path arrangement for the network, and timing controller is added for round robin scheduling algorithm in switch unit which provides the resource to complete each work in single chance.

Flexibility of the path arrangement: As per the requirement of flexibility of path arrangement in the proposed method, it is designed in such a way that the path arrangement scheme can use either the fixed priority, i.e., making all the request of transmitters to be high or variable priority, i.e., using a counter which can go through all the possible combinations of requests and selects the required combination in each transition. In the existing work, circuit switching approach is applied to dynamic path arrangement scheme and left over the fixed priority permutation. Therefore to provide the flexibility, circuit switching is applied for both the styles and designed as an optional case.

Variable for round robin scheduling algorithm: Timer is used to provide timing control for each transition to complete its work in single chance. In the switch unit, it is included with request checker which accepts the timing values and gives OFF time, ON time, grant signals as output, these signals are used to control the timer which accepts data from each transmitter and according to the control signals it transmit data. Enable is used to accept destination port through multiplexer and according to the control signal from request checker through time checker transmits data to the receivers whose request is logic high.

Input module for flexibility: In the Fig 1, circuit switching mechanism for fixed priority permutation network is applied, the input module in this architecture will be accepting inputs of transmitting data from each transmitter and internally it generating requests as all the requests are high for fixed priority. In the next block, when the Rst=1, Ena=1, the data inputs from different transmitters is sent via multiplexer and the requests are sent to round robin scheduled queue where transmitting data will be stored from different transmitters. The 3-input control signal to multiplexer will decide which transmitter's data to be transferred to Round -robin queue. Three data transmission modes (Transmitting mode, Block mode and Disable mode) are designed. Each transmitter will use its own time slice according to the priority queue.

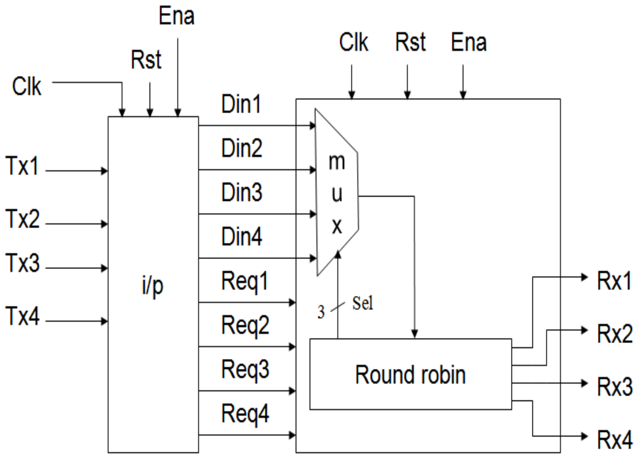


Fig 1: Input module for flexibility

Switch module for flexibility: The Fig 2 show the operation of the Switch module for fixed priority, this block receives the data inputs, respective requests, its destination ports, transmitters. The three-input control signal to multiplexer will decide which transmitter's data to be transferred to Round -robin queue. Three data transmission modes (Transmitting mode, Block mode and Disable mode) are designed. Each transmitter will use its own time slice according to the priority queue.

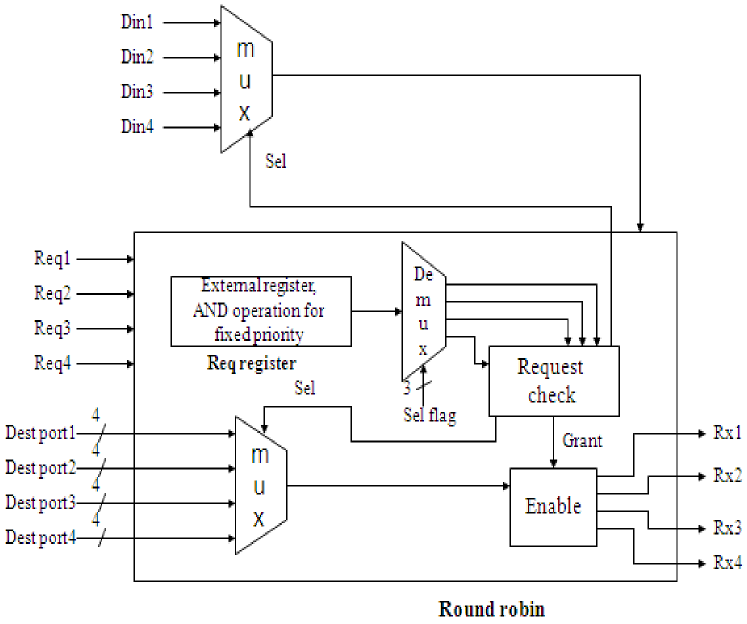


Fig 2: Switch module for flexibility

Data inputs are passed through a multiplexer which is controlled by a selection signal from request checker which indicates the request status. Requests are stored in external register and AND operation is performed with internal designed 4-bit variable which is fixed as all 1’s, if the request is logic-1 that request is going to be high and the 4-bit variable is sent to request checker through de-multiplexer which is controlled by selection flag and the request checker is generating grant signal for enabling the receivers and one more signal to control both multiplexers used for data input and destination port(data select request, destination select). The grant signal to enable sends the data of a transmitter. Destination ports from one end of enable through multiplexer which is controlled by request checker. The enable sends the data to receivers.

Switch module for variable priority with time controller: Fig 3 shows the block diagram of switch module with time controller which optimizes the performance of the network. As explained regarding Fig 2, in this also requests are transferred from external register but there is no need of AND operator from there through multiplexer requests are sent to request checker according to the control signal from request checker block.

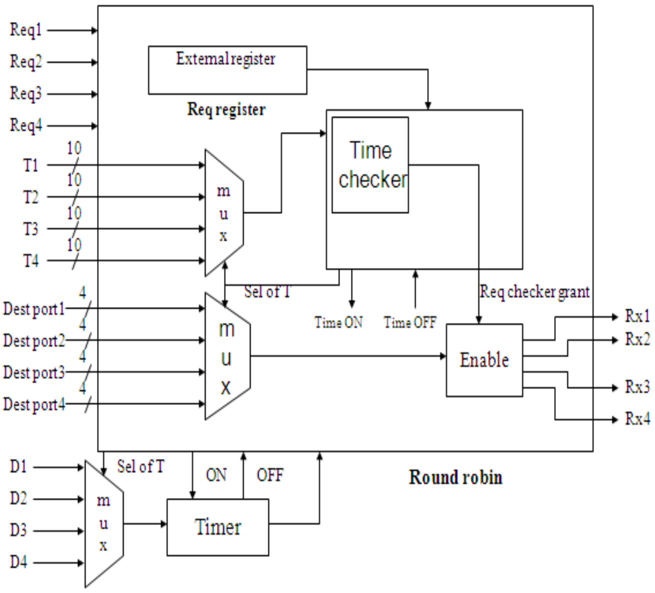


Fig 3: Switch module for variable priority with time controller

Time checker is used in request checker which sends the grant signal to enable block and sends the control signals to timer(ON time & OFF time). Timer sends the data to internal block and as per the destination port bits the data will be transmitted till OFF time. The main advantage in this method is to complete each transmitter work in single chance. (i.e., For example: if a transmitter has to transmit 40 bits of data to destination receivers the Time ON will be logic high till 4 clock cycles as each cycle is assumed to transmit 10 bits of data). This variable timer helps in varying the clock time and increases the efficiency.

Result Analysis

Fig 4 shows the simulation results for input module for transmitting mode (i.e., ans=01). In this one can see the signal for fixed priority which is mentioned as logic high (fixed=1 & variable=0) all the requests are enabled. In this mode all the transmitter's data will be transmitted.

The Fig 5 shows the simulation results for input module for block mode (i.e., ans=10). In this we can see the signal for fixed which is mentioned as logic high (fixed=1 & variable=0) all the requests are enabled. In this mode even all the requests are high but the there will be no data transfer.

Fig 6 shows the simulation results for input module for disabled mode (i.e., Ans=11). In this we can see the signal for fixed which is mentioned as logic high (fixed=1 & variable=0) all the requests are disabled. In this mode all the requests and data transfers are disabled.

Fig 7 shows the simulation results for input module for transmission mode (i.e., ans=01). In this we can see the signal for variable priority which is mentioned as logic high (fixed=0 & variable=1) different requests are enabled. Similarly other modes as shown for fixed priority are designed.

Fig 8 shows the simulation results for switch module in variable priority without timing control. In this we can see according to the requests and destinations, the data transfer takes places whose requests are enabled in such a way it follows the priority (when data select signal datasel=111, there will be no data transfer. when datasel=001 or 010 or 011, there will be data transfer).

Fig 9 shows the Simulation results for switch module in variable priority with timing control in which the data transfer takes place according to the time provided for each transmitter, which is variable it will vary according to the size of data to transmitted.

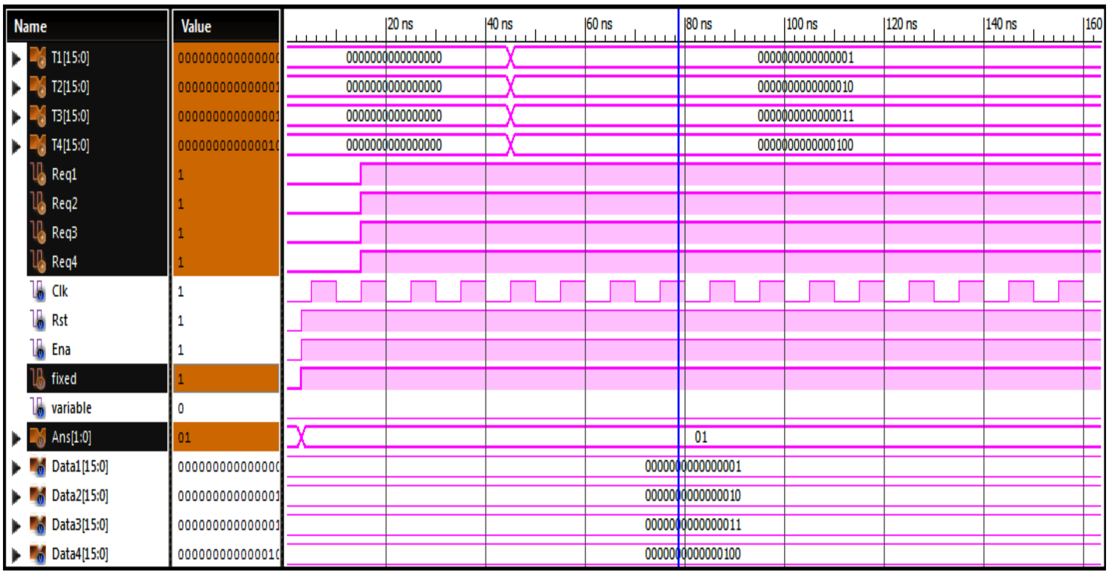


Fig 4: Simulation result for input module with transmission mode

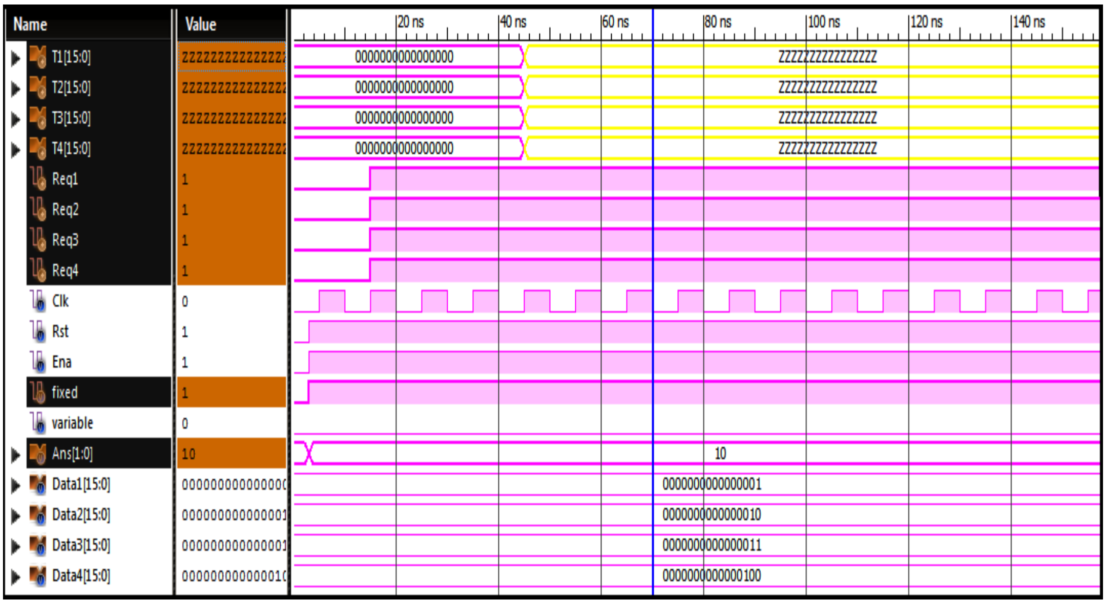


Fig 5: Simulation result for input module with block mode

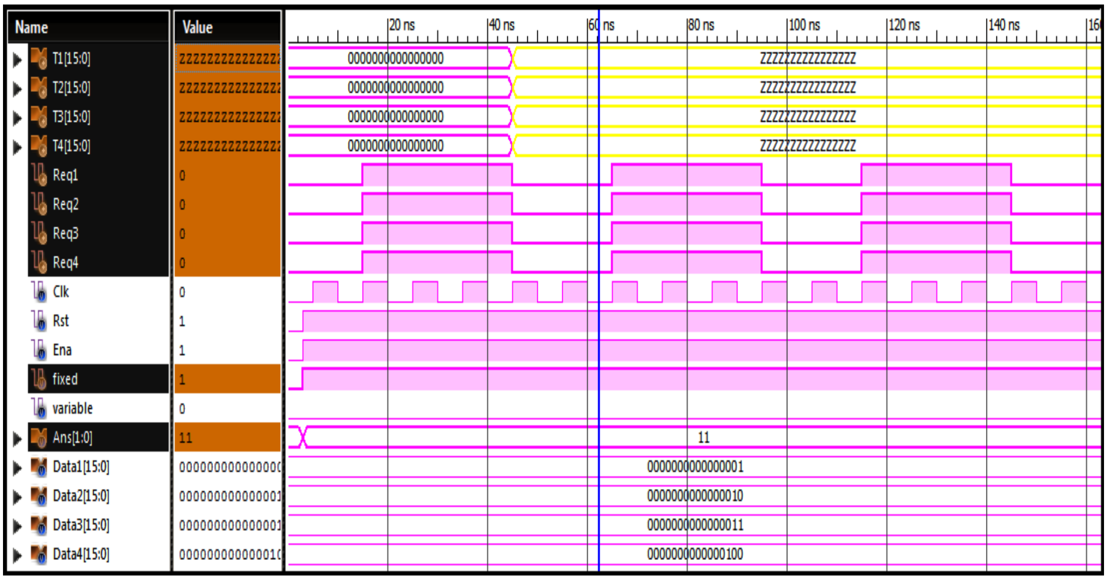


Fig 6: Simulation result for input module with disable mode

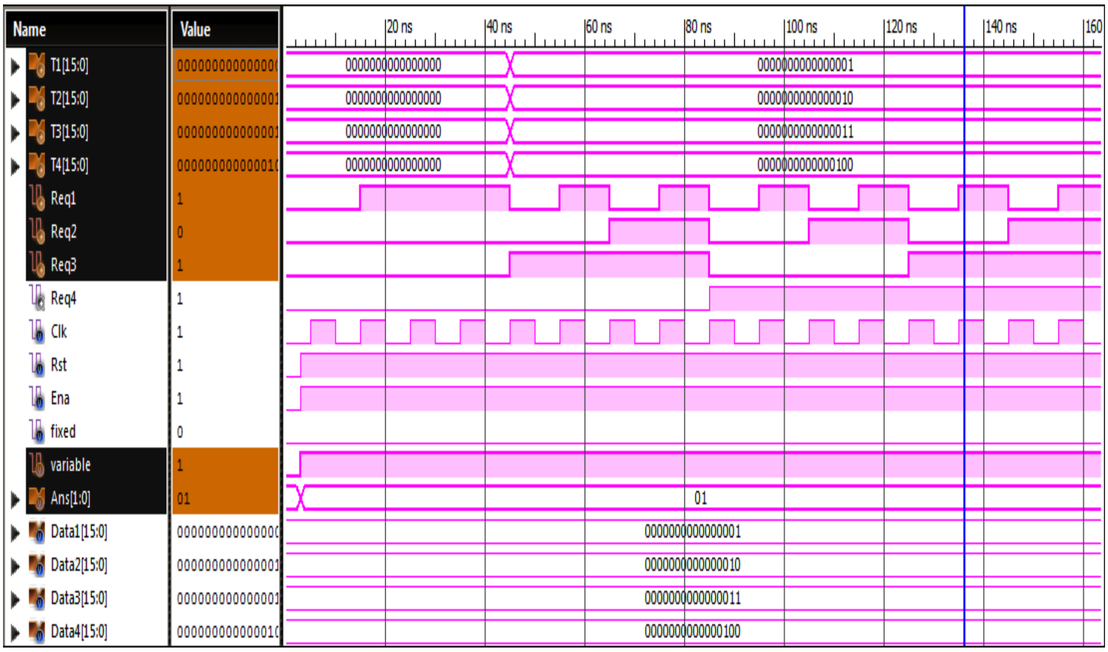


Fig 7: Simulation result for input module for transmission mode with variable priority

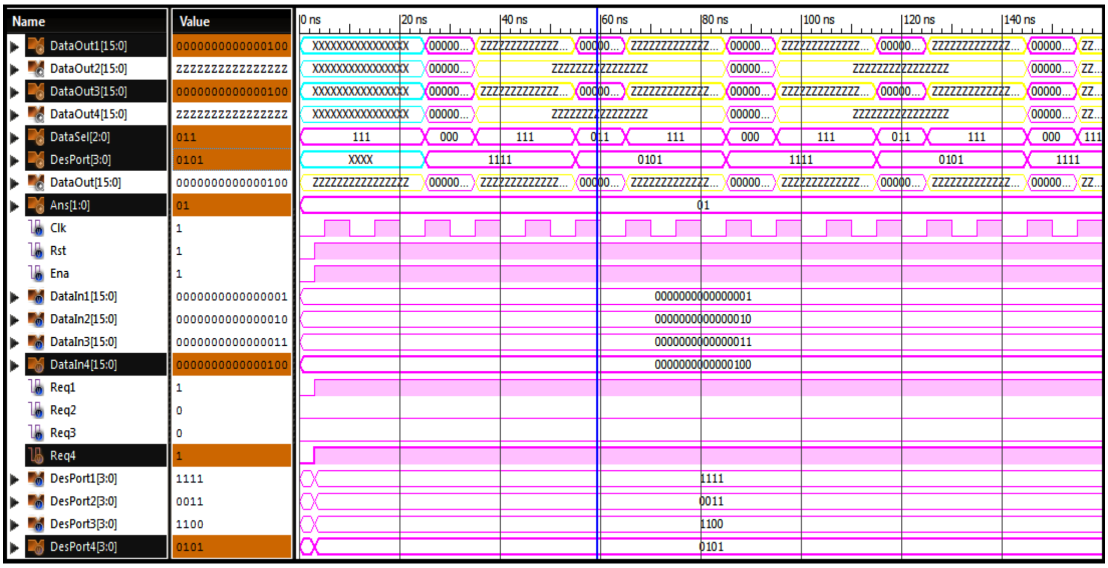
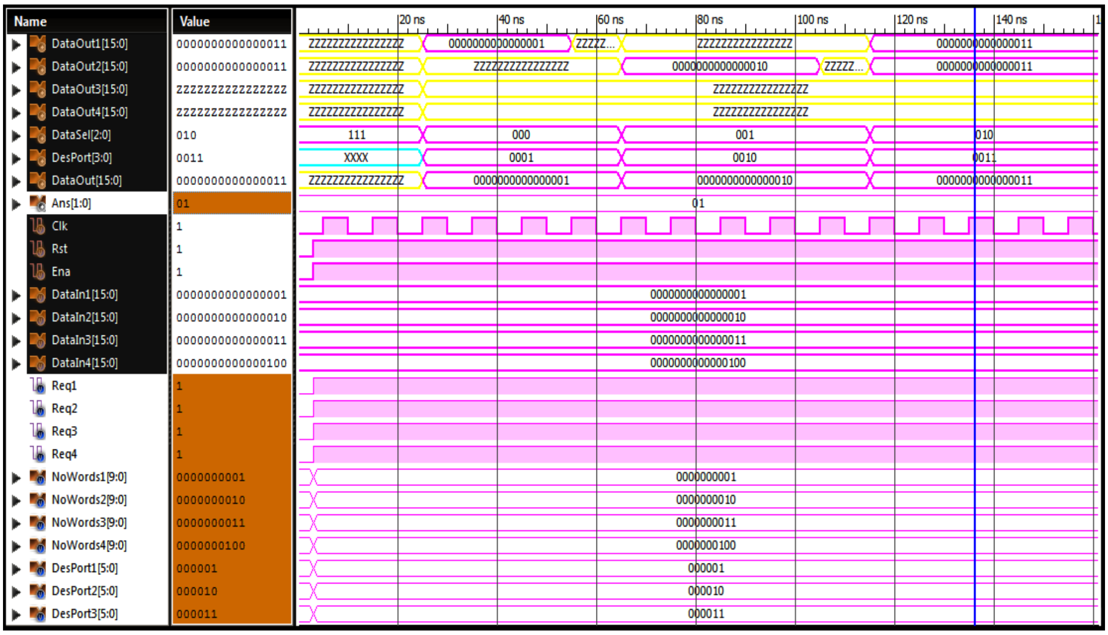


Fig 8: Simulation results for switch module in variable priority without timing control



Name	Value
DataOut1[15:0]	0000000000000011
DataOut2[15:0]	0000000000000011
DataOut3[15:0]	zzzzzzzzzzzzzzzz
DataOut4[15:0]	zzzzzzzzzzzzzzzz
DataSel[2:0]	010
DesPort[3:0]	0011
DataOut[15:0]	0000000000000011
Ans[1:0]	01
Clk	1
Rst	1
Ena	1
DataIn1[15:0]	0000000000000001
DataIn2[15:0]	0000000000000010
DataIn3[15:0]	0000000000000011
DataIn4[15:0]	0000000000000100
Req1	1
Req2	1
Req3	1
Req4	1
NoWords1[9:0]	0000000001
NoWords2[9:0]	0000000010
NoWords3[9:0]	0000000011
NoWords4[9:0]	0000000100
DesPort1[5:0]	000001
DesPort2[5:0]	000010
DesPort3[5:0]	000011

Fig 9: Simulation results for switch module in variable priority with timing control

Table 1 shows the time periods (clock pulses) used by each transmitter to complete its work. Considered, transmitter T1 need 4 words to be transmitted, transmitter T2 need 5 words to be transmitted, transmitter T3 need 6 words to be transmitted. As observed, the increase in number of words to be transmitted can show the increase in difference between existing method and proposed method for total time taken in completing execution.

Table 2 shows the analysis report for this design in varies devices, here it shows the frequency of the design in different devices where virtex6 gives the highest frequency.

Table 1: Analysis Report For Clock Pulses Used By Each Transmitter

Method	Time taken to execute T1 work (nseconds)	Waiting time (nseconds)	Time taken to execute T2 work (nseconds)	Waiting time (nseconds)	Time taken to execute T3 work (nseconds)	Waiting time (nseconds)	Total time taken for completing execution (nseconds)
Existing	40	120	50	150	60	170	590
Proposed	60	0	70	70	80	140	420

Table 2: Analysis Report For Frequency In Different Devices

Parameter	Spartan 3E	Spartan 6	Virtex 4	Virtex 6
Speed grade	5	12	3	2
Minimum period	4.388ns (227.9 MHz)	2.236ns (447.2MHz)	3.757ns (266.16MHz)	1.949ns (513.08MHz)
Minimum input arrival time before clock	4.318ns	2.432ns	3.798ns	1.254ns
Maximum output required time after clock	7.088ns	5.671ns	6.470ns	2.055ns
Maximum combinational path delay	6.834ns	5.532ns	6.430ns	1.482ns

Conclusion

Dynamic path setup scheme using Circuit switching approach can reduce the over head, but it is still required to optimize the performance of the system. So, by a new design which brings out the feature of timing controller for clock, the performance of the system is increased much compared to the existing techniques. And also gives the flexibility of changing the priority of choosing the process. The increase in performance of data transfer is shown through simulation results. The resulting frequency with different devices for this design is tabulated. Finally the proposed design is better for using high speed applications.

References

[1] Manjunath, Dhana Selvi, “Design and Implementation of an On-Chip Permutation Network for Multiprocessor System-On-Chip,” International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 3 Issue 5, May 2014, pp.1974-1979.

[2] N. Michael, M. Nikolov, A. Tang, G. E. Suh, and C. Batten, “Analysis of application-aware on-chip routing under traffic uncertainty,” in Proc. IEEE/ACM Int. Symp. Netw. Chip (NoCS), 2011, pp. 9–16.

[3] Zahra sadat GhandrizP, Pand Esmaeil Zeinali Kh.P, “A New Routing Algorithm for a Three-Stage Clos Interconnection Networks,” IJCSI International Journal Of Computer Science Issues, Vol. 8, Issue 5, No 2, September 2011 ISSN (Online): 1694-0814, pp.309-313.

[4] S. Talapatra, H. Rahaman, and J. Mathew, “Low complexity digit serial systolic montgomery multipliers for special class of GF(2m),” IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 18, no. 5, pp. 847–852, May 2010.

[5] C.-Y. Lee, “Error-correcting codes for concurrent error correction in bit-parallel systolic and scalable multipliers for shifted dual basis of GF (2m),” in Proc. Intern. Sym. Para. Dis. Process. Appl.,2010, pp.405–412.

[6] D. Ludovici, F. Gilabert, S. Medardoni, C. Gomez, M. E. Gomez, P. Lopez, G. N. Gaydadjiev, and D. Bertozzi, “Assessing fat-tree topologies for regular network-on-chip design under nano scale technology constraints,” in Proc. Design, Autom. Test Euro. Conf. Exhib. (DATE), 2009, pp. 562–565.

[7] C.-W. Chiou, C. C. Chang, C. Y. Lee, T. W. Hou, and J. M. Lin, “Concurrent error detection and correction in Guassian normal basis multiplier over GF(2m),” IEEE Trans. Computers, vol. 58, no. 6, pp. 851–857, 2009.

[8] P. K. Meher, “Systolic and non-systolic scalable modular designs of finite field multipliers for Reed-Solomon Codec,” IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 17, no. 6, pp. 747–757, Jun. 2009.

[9] Holmberg, “Optimization Models for Routing in Switching Networks of Clos Type with Many Stages“, An Electronic International Journal - AMO Advanced Modeling and Optimization, Volume 10, Number 1, 2008, pp. 1841-4311.

[10] H. Moussa, A. Baghdadi, and M. Jezequel, “Binary de Bruijn on-chipnetwork for a flexible multiprocessor LDPC decoder,” in Proc. ACM/IEEE Design Autom. Conf. (DAC), 2008, pp. 429–434.