TERNARY TREE ASYNCHRONOUS INTERCONNECT NETWORK FOR GALS' SOC

Vivek E. Khetade[1], Dr. S. S. Limaye[2]

[1]Department of Electronic Design Technology
Shri Ramdeobaba College of Engineering & Management,
Rashtrasant Tukdoji Maharaj Nagpur university Nagpur, India
khetadev@rknec.edu

[2]Department of electronics engineering,
Jhulelal Institute of Technology
Rashtrasant Tukdoji Maharaj Nagpur university Nagpur, India
shyam_limaye@hotmail.com

ABSTRACT

Interconnect fabric requires easy integration of computational block operating with unrelated clocks. This paper presents asynchronous interconnect with ternary tree asynchronous network for Globally Asynchronous Locally Synchronous (GALS) system-on-chip (SOC). Here architecture is proposed for interconnection with ternary tree asynchronous network where ratio of number NOC design unit and number of router is 4:1,6:2, 8:3,10:4 etc .It is scalable for any number of NOC design unit. It offers an easy integration of different clock domain with low communication overhead .NOC design unit for GALS 'SOC is formulated by wrapping synchronous module with input port along with input port controller, output port along with output port controller and local clock generator. It creates the interface between synchronous to asynchronous and asynchronous to synchronous. For this purpose four port asynchronous routers is designed with routing element and output arbitration and buffering with micro-pipeline. This interconnect fabric minimizes silicon area, minimize Latency and maximize throughput. Here functional model is made for TTAN and application MPEG4 is mapped on the Network .Desired traffic pattern is generated and performance of the network is evaluated. Significant improvement in the network performance parameter has been observed.

KEYWORDS

GALS, NOC, asynchronous design, ternary tree network, data synchronization

1. INTRODUCTION

Network-on-chip focused on research challenge of maintaining the scalability of interconnection network[1].Current techniques when extrapolated to future technologies will face significant shortcoming in several key areas .Power consumption is expected to exceed the budget for commercial chip multiprocessor by a factor of 10 x by year 2015 .There are less quantifiable support for scalability, reliability and ease of integration of complex heterogeneous system .These issues becomes an important requirement for implementing future system ,specially handling synchronous block with arbitrary unrelated clock frequencies. An ongoing efforts to develop low cost and flexible NOC for high performance where power consumption and ease of integration challenges serves as motivation for incorporating low power asynchronous network[2]. Network provides fine grained related integration of synchronous component in a globally asynchronous locally synchronous style architecture [3][4][5][6]. The synchronous component may be processing cores, function unit or memory modules .To support scalability synchronous component may have arbitrary unrelated clock rates i.e. the goal to integrate

DOI : 10.5121/vlsic.2013.4106

hetrochronous system. Ternary tree asynchronous Network which offers the good ratio between router and number of the NoC design unit connected to it. In the proposed system with three routers, it is possible to connect eight synchronous module /NOC unit. With four router ,it is possible to connect ten synchronous module ,with 5-12,7-16 and so on .We tried to reduce the interconnect overhead by reducing number of router for the required synchronous module. Here four port asynchronous routers is designed for routing the packet among NOC unit.Three port are used for connecting the NOC unit. Synchronous module is wrapped with asynchronous wrapper. Wrapper consists of input port along with input port controller and output port along with output port controller. Clock is provided with local clock generator .Local clock generator is the part of asynchronous wrapper. Packetisation and Depacketisation of data takes place inside the synchronous module. Ideally interconnection fabric should provide zero overhead delay and explicit concurrency. But practically there is delay of synchronization during transit of data from synchronous to asynchronous and from asynchronous to synchronous and delay offered by router. Effort is made to reduce delay by clustering the synchronous module at one router where there is more traffic. Globally asynchronous and locally synchronous (GALS) architectures aid such integration by allowing the synchronous blocks to operate independently with other synchronous blocks through asynchronous communication channels [7].

This paper is organized as follows. The paper begins with architecture of ternary tree asynchronous network for GALS SOC. In section 3 four ports asynchronous router architecture is described. In section 4 architecture of NOC unit is described with synchronizer for data synchronization. In section 5, communication task graph of MPEG4 is presented. In section6 mapping of application MPEG4 on TTAN is analysed. Network performance analysis is presented in section 7 under result analysis. Conclusion is presented in section 8.

2. ARCHITECTURE OF TERNARY TREE ASYNCHRONOUS NETWORK

Ternary tree asynchronous interconnect network consists of four port asynchronous router as node and noc unit as a leaf .There is a link between each router and noc unit, and also between two routers. Each leaf node has three NOC unit distinguished as left ,mid and right . Node with children is parent nodes and child node may contain reference to their parent .Outside the tree there is often reference to the root node.

TTAN is a heterogeneous interconnect network which consists of non- regular structure and where communication flows are certainly localized & non–uniformly distributed[2]. It is highly possible that the optimal topology is TTAN. Whereas Mesh network is a homogeneous topology. It consists of regular structures with uniform communication flow.

The main feature of Ternary Tree Asynchronous Network

1) *Scalable*: - TTAN is scalable for any number of NOC design unit. As per the application requirement, structure will grow i.e number of leaf node and router node.

2) *Interconnect overhead*: - Interconnect overhead is reduced by reducing the number of router for connecting the required number of NOC unit. The ratio of number of noc unit and router is 4:1, 6:2, 8:3 etc. Here TTN is given which requires three routers for eight NOC unit as compared to mesh network which requires eight router. TTAN is best of Binary tree network and Quaternary tree network with respect to interconnect overhead and complexity of router.

3) *Clustering*: - Clustering of NOC units to one of the node or nearby leaf node of one hop toward root node is very easily accommodate for the application where there is a traffic

exist.. Localizations of flow of data can achieved through clustering. It helps in facing heavy traffic scenario.

4) *Implicit Parallelism*: - Concurrent operation is possible as network is asynchronous.

The interconnection of node depends on number of synchronous block .Always leaf node will consists of three synchronous block and root node will depends on the structure of network. The philosophical concepts of clustering synchronous block can be easily achieved with this ternary tree network. Its architectural impact is more with respect to latency, throughput, and communication overhead. Clustering of synchronous block can help in facing heavy traffic scenario between synchronous block. Such cluster can be connected to leaf node or adjacent leaf node. This network structure is scalable. Figure 1, shows possible network interconnection. Routing of packet is highly simplified. Routing between adjacent router leaf nodes is always one hop, two hop depending on the number of synchronous block in the NOC. It is possible to route the packet according to wormhole or xy protocols. They implement the routing strategy. Links connect the nodes. NOC unit is attached to a local router which connects the core to the neighbouring nodes via a NOC.

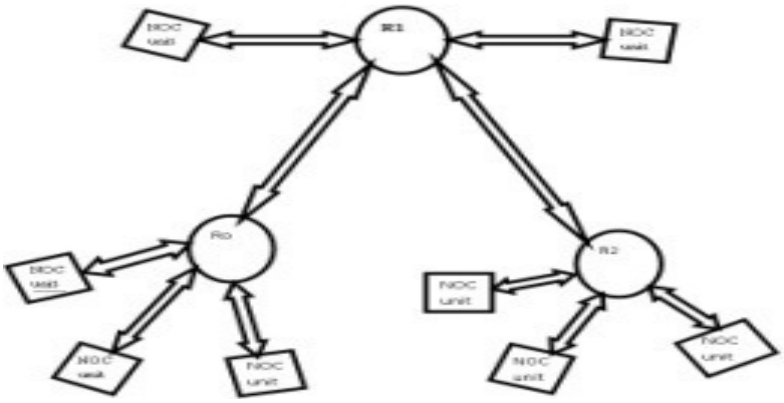

Fig. 1 Ternary tree asynchronous network for eight NOC unit

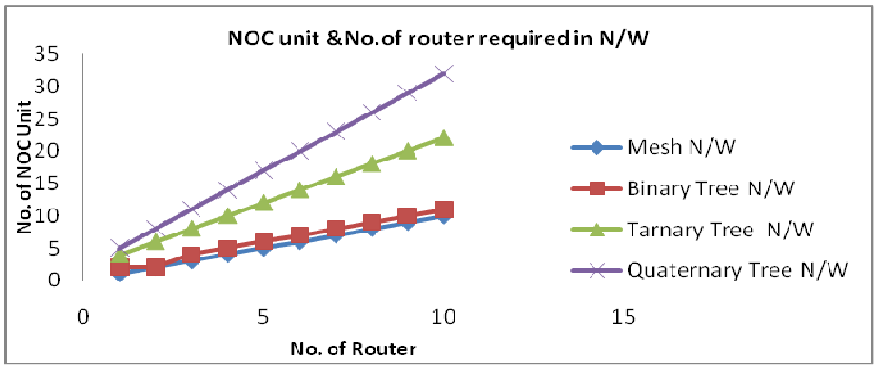

Fig 2 Comparisons of Network

3. FOUR PORT ASYNCHRONOUS ROUTER ARCHITECTURE

Proposed router architecture consist of four port labeled north (N), south(S) ,east(E),and west(W) separately for input port and output port along with handshake signal. Buffering facility at the input side is provided with muller-c micropipeline . Routing element is responsible for routing incoming packet to one of the output port of remaining three ports. Asynchronous arbitration accepts data from exactly one of three port and forward it to single output port. In one router there is four micro-pipeline , four routing element and four arbitration element. Adjacent router or attached NOC design unit communicate using request (req) and acknowledge (ack) signals following four phase signaling. It is push channel type. 'ReqEs' is the request from east port as a source and 'AckEs' is the acknowledge to east port from micro-pipeline. Similarly ReqWs,AckWs , ReqNs, AckNs, and ReqSs, AckSs are request and acknowledge of west, north and south respectively.' ReqEd' is the request from micro pipeline to destination router or port of NOC unit,and AckEd is the acknowledge to micro pipeline from destination router or port of NOC unit. Similarly ReqWd, AckWd ,ReqNd,AckNd,and ReqSd,AckSd are request and acknowledge of west, north and south respectively. B signal is part of the destination address for the packet .The full destination address is encoded in each flit by the source terminal and may be removed after the flit is routed to the subsequent stage.

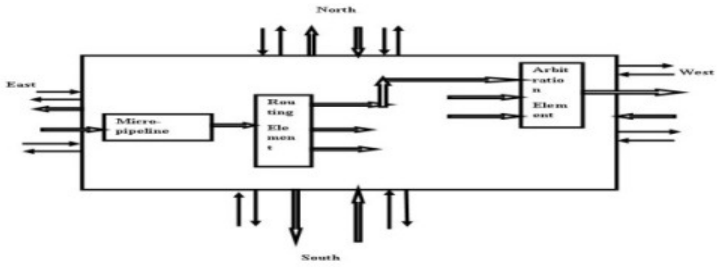

Fig. 3**.** Block diagram of 4-port router.

3.1ROUTING ELEMENT

The routing Element performs demultiplexing operation. This architecture is taken from [2]with little modification i.e. in mesh-of-tree network routing elements consist of only one input and two output .Here we have taken one input and three output. Basic structure is same of [2].Each routing element consists of register and latch controller. Each register is standard cell level-sensitive D-type Latch register that is normally opaque, preventing data from passing through it. Detail implementation for latch controller is shown in Fig 4. Each latch controller is responsible for controlling two signal which enables data storage and inter-stage communication :1)the corresponding data register enable signal(En) 2)the corresponding request output to the next stage 3)three request output are combined by XOR2 to produce single acknowledgement to input stage. B signal is part of the destination address for the packet. After the data input are stable and valid, a single request transition on Req occurs at the input. The latch controller selected by decoded routing signal, enables the corresponding latch register and data advances to the selected output channel. Each toggle elements converts an input Req transition to an output transition on the appropriate port.

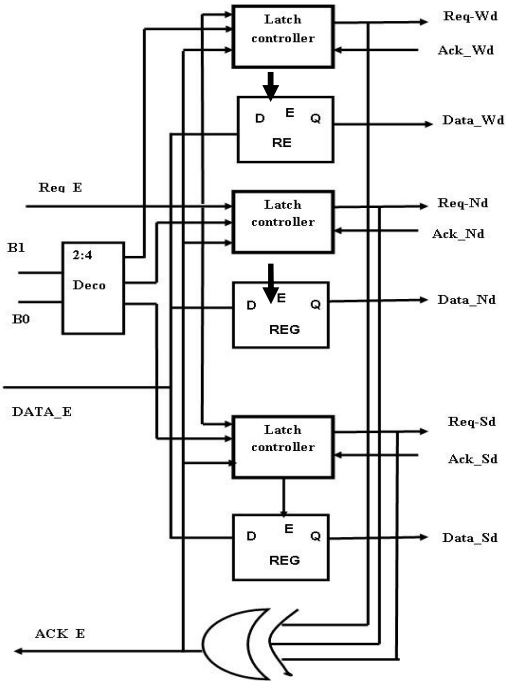

Fig.4 Routing Element

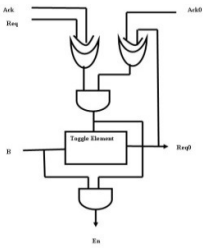

Fig 5 Latch controller

3.2ARBITRATION ELEMENT

The arbitration element accepts data from exactly one of the three ports from output of routing element and forwards it to single output port. It provides the function complementary to routing element. This architecture is taken from [2]with little modification that is in datapath there is three input ,Data0,Data1,Data2 and latch controller controls the three 'req' and three 'ack'.Fig3.2 shows structure of the arbitration element. It include 10 (ten) level sensitive D-type transparent latch (numbered L1 through L10).Latches L1, L2 and L3 are normally opaque. Latches L4, L5 and L6 are normally transparent, selectively propagating or blocking acknowledgement on the input channel .Latches L7 through L10 are normally transparent, selectively propagating or blocking request on the output channel. The XOR gate at the input of latch L7 function as merge element joining three mutually –exclusive transition signaling signal ReqEs, ReqNs and ReqSs on to a single signal Req. Finally there is one multiplexer and one data register for the entire data path. The arbitration is performed by mutual exclusion element.

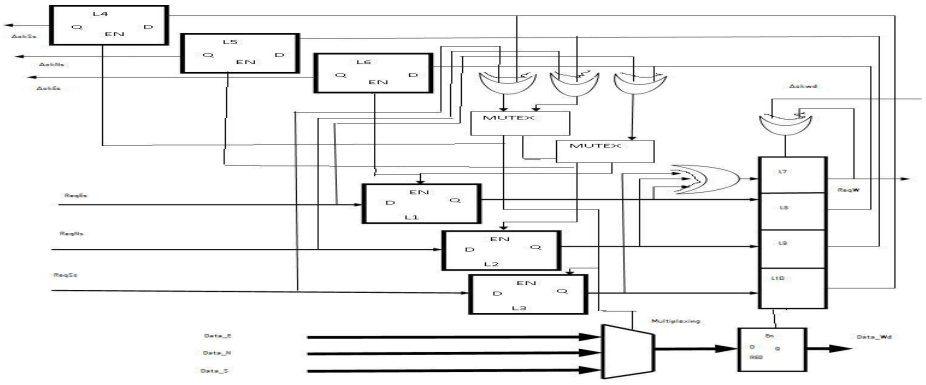

Fig. 6 Arbitration Element

The operation begins when flit arrives on one of the input channel. The flow control unit arbitrates the request through the Mutex component and performs two action: setting the correct multiplexer select signal for the data path and forwarding the winning request transition to the latch controller (i.e. enabling the corresponding latch L1 or L2 or L3). Once the Mutex is resolved the acknowledgement latches L4, L5 and L6 become opaque any acknowledgement on the input channel. If there is competition i.e. the other input channel concurrently receives a new request Mutex grants one request and the loosing request remains blocked until the winner has been processed.

3.3 MULLER-C MICRO PIPELINE

A micro pipeline has event-controlled registers for data path and Muller C-elements for control. The control circuit for a micro pipeline is a string of Muller C-elements [8][9][10]. In this figure one of four identical stages is shaded and alternate stages have been drawn upside down. At the input and output to each stage there are request, R(n), and acknowledge, A(n), signals. Inverters in the acknowledge paths are represented by "bubbles" at one input of each Muller C-element. Cross bar has four input port and four output port with two or more control lines for simply routing the packet toward desired port. Control and decoding circuit is for controlling the flow of data to desired port.

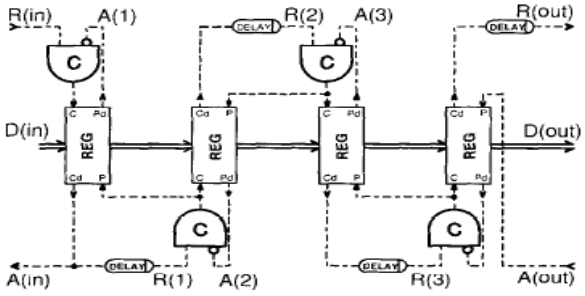

Fig7 Muller-C Micropipeline

4. NOC Design unit

This proposed architecture of NOC unit for GALS'SOC is as shown in fig.8 Data synchronization between synchronous to asynchronous and asynchronous to synchronous domains is accomplished with port controller and synchronizer respectively. NOC unit is operating at different frequency. This asynchronous wrapper facilitate the asynchronous data transfer between NoC unit and router .Asynchronous wrapper consists of local clock generator , output port along with output port controller ,and input port along with input port controller. Two flop synchronizer is used for synchronization of signal crossing from asynchronous to synchronous domain [11]. Synchronizer prevents the system to enter into metastability state .It support safe data transfer between the two different frequency domains. The packetisation and depacketisation takes place in NOC unit.

Packet will consist of header flit and body flit as shown in fig 9. Several consecutive flits are gathered into single packets. This packet is transmitted without interruption from node to node in a worm hole data flow. The packet protocol requires a specific signalling in order to identify the first flit and last flit of the packet. Two additional flits are added to the 8-bit data path coding respectively BOP(Beginning of packet) and EOP(End of packet). The

routing information(path target) is provided in the first flit which is called header flit of packet[15].

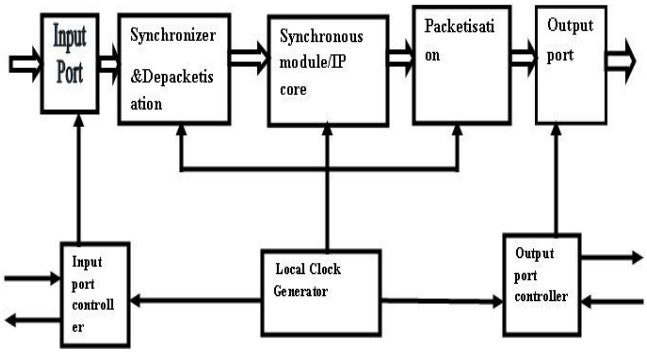

Fig. 8 NOC design Unit

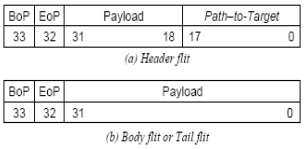

Fig 9. Packet format

5. COMMUNICATION TASK GRAPH OF MPEG4

A commonly use abstraction in the literature has been titled a communication trace graph (CTG) [12] or a core graph. A path describes pair of source and destination cores and the particular link and routers a packet traverses[13]. The CTG has a n-tuple of values per path that often includes average expected traffic rate per path and sometimes a latency requirement of a packet.

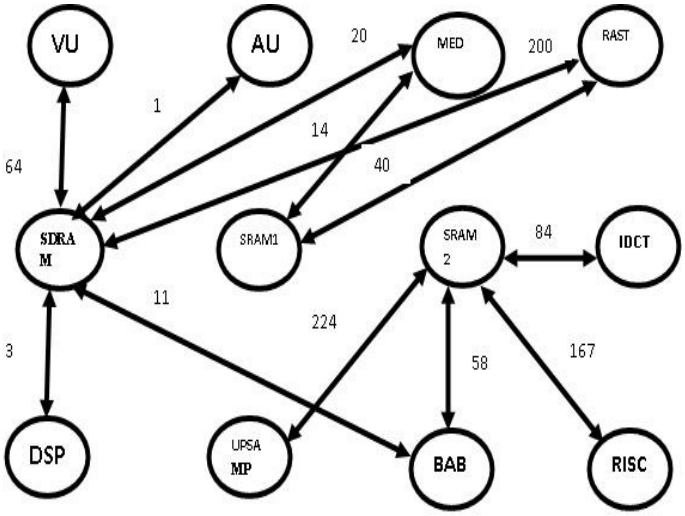

Fig10 CTG of MPEG4 (Mbytes/sec)

6. MAPPING OF MPEG4 APPLICATION ON TTAN

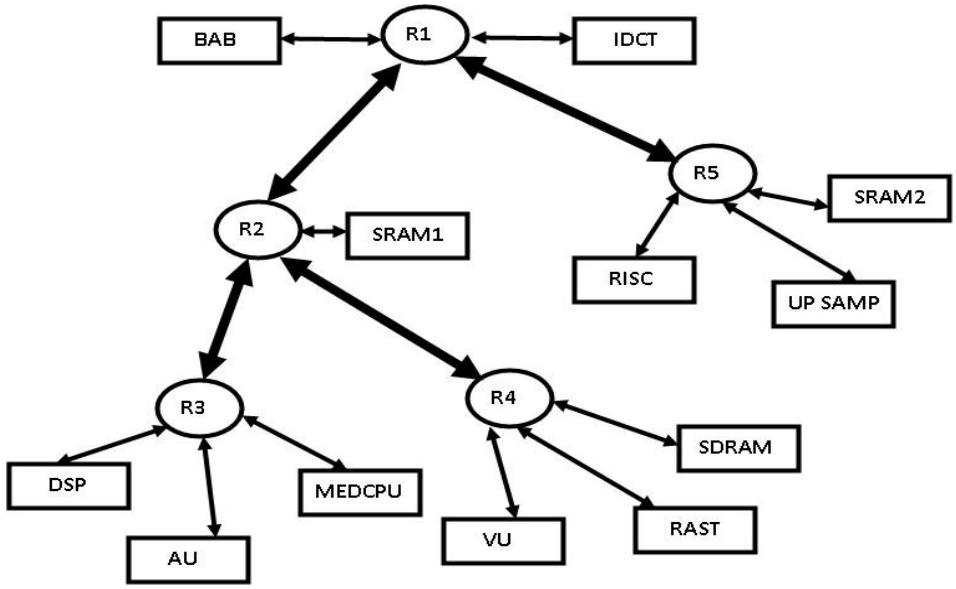

Fig 11 Mapping of MPEG4 on TTAN

Here we mapped the application on TTAN .During mapping the application on the TTAN, we have consider the traffic between modules. SDRAM, RAST & VU is placed on the same router R4 as the traffic between these module is 200Mbytes,64 Mbytes .Localisation of the traffic takes place. Other core AU,MEDCPU,DSP are placed two hope SDRAM as traffic is comparatively less. SRAM2 ,UPSAMP ,RISC is connected to router 'R5'as data rate is 224 Mbytes/sec and 167 Mbytes/sec. It is clustering of heavy traffic core. Core IDCT and BAB are placed one hope above the SRAM2 where traffic comparatively less i.e.84 Mbytes/sec and 58Mbytes/sec.

7. RESULT ANALYSIS

Functional model is constructed with Verilog for TTAN. Five routers are instantiated for mapping MPEG4 application. The packet of 8-bit are generated at each port at random interval following an exponential distribution with the mean corresponding to the desired input traffic rate. Each NOC module is assumed to be operated at the frequency listed in the Table7.1 with data rate among various core of MPEG4.

Table.1 Frequency and data rate of core of MPEG4

Core module	Frequency of operation	Data rate (Mb/sec)
VU	667 MHz- 1000MHz	64
SDRAM	100-533MHz	3/11/200/ 20/1/64
AU	100-600 MHz	1
MEDCPU	200Mhz-300Mhz	20
DSP	300MHz-700MHz	3
SRAM1	300MHz-700Mhz	14/40

RAST	300-400Mhz	200
SRAM2	300MHz-700MHz	84/167/ 58/224
RISC	200-250Mhz	162
BAB	100-150Mhz	58
UPSAMP	250-350Mhz	224
IDCT	50MHz-1000 MHz	84

Throughput is the output data rate of the network during the measurement phase of simulation (GB/s) . Latency is measured as the time from creation of packet until it reaches its destination Table -7.2 shows result of latency and throughput for routing primitives ,arbiter and micro pipeline for conducted experiment. Throughput is evaluated for single steady state pattern Giga flit per seconds (GFPS)

Table 2 Latency and throughput of Component

Component	Latency (ps)	Throughput (GFPS)
Routing Element	557	0.6
Arbitration Element	504	0.7
Micro pipeline	478	0.6

System level performance evaluation of latency for the asynchronous network is shown in table below based on the interaction specified as per application MPEG4 and experiment conducted neglecting the delay of synchronization. Latency depends on the way how we have mapped the application MPEG4Latency of the packet generated at core SDRAM and reaching at destination core BAB is certainly highest. In the table 'X' indicates the no interaction and number indicates the latency in picoseconds. The minimum latency of network is 1535 ps and maximum latency is 4605ps.

Table.3 System level latency w.r.t each communication

CORES	VU	SDRAM	AU	MEDCPU	DSP	SRAM1	RAST	SRAM2	RISC	BAB	UPSAMP	IDCT
VU	X	1535	X	X	X	X	X	X	X	X	X	X
SDRAM	1535	X	4605	4605	4605	X	3070	X	X	4605	X	X
AU	X	4605	X	X	X	X	X	X	X	X	X	X
MEDCPU	X	4605	X	X	X	3070	X	X	X	X	X	X
DSP	X	4605	X	X	X	X	X	X	X	X	X	X
SRAM1	X	X	X	3070	X	X	3070	X	X	X	X	X
RAST	X	1535	X	X	X	3070	X	X	X	X	X	X
SRAM2	X	X	X	X	X	X	X	X	1535	3070	1535	1535
RISC	X	X	X	X	X	X	X	1535	X	X	X	X
BAB	X	4605	X	X	X	X	X	3070	X	X	X	X
UPSAMP	X	X	X	X	X	X	X	1535	X	X	X	X
IDCT	X	X	X	X	X	X	X	3070	X	X	X	X

8. CONCLUSIONS

Ternary tree asynchronous interconnect network is a low overhead asynchronous interconnect network best suited for GALS 'SOC. It is hetrochronous, scalable structure. It provides localisation of traffic by creating the cluster of NOC unit having heavy traffic NOC unit is articulated by wrapping IP core with asynchronous wrapper along with packetisation and depacketisation circuit. Asynchronous routing primitives, arbitration and micro pipeline is used for creating the router. Application MPEG4 is mapped on TTAN which requires only five routers. The required data rate is generated inside the NOC module and injected into the network from the conducted experiment network latency and throughput is observed. Comparable improvement in performance is observed.

REFERENCES

[1] J. Owen, W. Dally , R Ho,D.Jayasimha,S.Keckler and L.Peh, "Research challenges for on-chip interconnection networks", IEEE Micro, vol 27,no 5,pp.96-108,sep-oct2007.

[2] M.N.Horak, S.M.Nowick,M.Carlberg and U.Vishkin,"A low overhead asynchronous interconnection network for GALS chip multiprossor", in IEEE Int Symp. NOCS,May 2010,pp 43-50.

[3] Y.Thonnart,P Vivet and F.Clermidy, "A full asynchronous low power framework for GALS NoC integration",in Proc Conf . DATE 2010, pp 33-38.

[4] Hyung Gyu , Radu morculescu , "on chip communication architecture exploration:A quantitative evaluation of Point To Point ,Bus , NOC approaches" ACM Transaction design Automation of Electronic system Vol. 12, No. 3,Article 23Aug 2007.

[5] S. Moore, G. Taylor, R. Mullins, and P. Robinson, "Point to point GALS interconnect," in Proc. ASYNC, 2002, pp. 69–75

[6] Denial Gebhardt, Junbok you and Kenneth S. Stevens, "Design of an energy –efficient Asynchronous NOC and its optimization tool for Heterogeneous SOCs"IEEE transaction on computer added Design of integrated circuit and system, Vol 30, No-9,sept 2011

[7] Rostislav (Reuven) Dobkin, Ran Ginosar, and Chris.tos P.Sotiriou, "High Rate Data Synchronization in GALS SoCs", IEEE Transac. on VLSI Systems, Vol. 14, No. 10, Oct 2006

[8] I.E. Sutherland, "Micropipelines",Commication ,ACM, vol 32,no6,pp720-738,1989 .

[9] J. Sparsø and S.B. Furber. Principles of asynchronous circuit design – A systems perspective. Kluwer Academic Publishers, 2001.

[10] J. cortadella, M kishinevskey etl, "Petrify: A tool for manipulating concurrent specification and synthesis of asynchronous controler" IECE Trans Inf.Syst vol E80-D,no.3 pp315-325,1990

[11] Ran Ginosar, "Metastability and synchronizer: A Tutorial", IEEE Design &Test of computer,September/October 2011.

[12] Davide Berfozzi, Antinine Jalbert, Shrinivasan Murali,Rutuparna Tamhankar ,stergios stergiou, Luca Benini, Giovanni DeMicheli, "NOC synthesis Flow for customized Domain Specific Multiprocessor system –on-chip",IEEE transaction on Parallel and distributed system ,vol-16,No-2, February 2005.

[13] Denial Gebhardt, Junbok you and Kenneth S. Stevens, "Design of an energy –efficient Asynchronous NOC and its optimization tool for Heterogeneous SOCs" IEEE transaction on computer aided Design of integrated circuit and system, Vol 30, No-9,sept 2011.

Authors

Prof. Vivek E. Khetade is pursuing PhD from Rashtrasant Tukdoji Maharaj Nagpur University, India, in the field of "Data synchronization in GALS SOC". He is an Assistant Professor in the Department of Electronic Design Technology of Ramdeobaba college of Engineering & Management for the last 10 years. He has been associated with switching theory and finite Automata, Asynchronous circuit related subject's right from the start of his career. His Masters was in the field of Electronic Design Technology.

Dr. Shyam S. Limaye received his PhD from Nagpur University in the faculty of Electronics Engineering. Currently he is working with JIT as an Principal. He has 33 years of experience in teaching as well as in industry. He also carried out a number of consultancy projects at DCM Data products, Delhi, PSI Data Systems Bangalore, and Zen and Art New York. His areas of specialization include Asynchronous VLSI design, interconnect design, Network on chip with emphasis on modelling and verification. He is a recognized supervisor and Guide to PhD scholars at Nagpur University.

.