http://www.jctjournals.com

ISSN(ONLINE):2278 - 3814

Volume-11, Issue 11, November 2022

JOURNAL OF COMPUTING TECHNOLOGIES (JCT)

International Journal

Page Number: 01-06

A LITERATURE REVIEW ON DIFFERENT SYSTEM-ON-CHIPS (SOC)

Raj Kamal Kishor Bharti[1], Prof. Devendra Patle [2],
[1]Research Scholar of VLSI [2]Department of Electronics and Communication
[1,2] School of Engineering
[1,2]Sri Satya Sai University of Technology & Medical Sciences, Sehore (M. P.) INDIA,
devendrapatlebgt@gmail.com

***Abstract*—The application domain of System-On-Chips (SoC) includes mobile devices, end terminals, multimedia terminals, automotive, set-top-boxes, games, processors etc. The SoC design paradigm relies heavily on reuse of intellectual property cores, enabling designers to focus on functionality and performance of the overall system. This is possible if IP cores are equipped with highly optimized interface for plug and play insertion into communication architecture. To this purpose the virtual Socket Interface Alliance represents an attempt to set the characteristics of industry wide, thus facilitating the match of pre-designed software and hardware blocks from multiple sources. The SoC interconnect must be designed and optimized to support a heterogeneous mix of data paths which may inherently have widely varying performance characteristics. The fabric must reliably deliver the required throughput and hide latency for performance critical paths while simultaneously managing the flow of traffic for slower paths and ports requiring lower bandwidth. Thus the system bus as a whole must strike the appropriate between latency and throughput for the collection data paths. Optimizing around this balance is essential to minimizing power, performance, area (PPA) costs and avoiding an inefficient, over-designed SoC.**

***Keywords*—Heat Sink, Review, Heat Transfer, Performance Parameters**.

I. INTRODUCTION

Heat Deep submicron technologies are making the integration of large IP blocks on same silicon blocks on same silicon die technically feasible. As a consequence, several heterogeneous cores can be combined through sophisticated communication architectures on same integrated circuit, leading to the development of flexible hardware platforms able to accommodate highly parallel computation. The application domain of these System-On-Chips (SoC). Includes mobile devices, end terminals, multimedia terminals, automotive, set-top-boxes, games, processors etc.

The SoC design paradigm relies heavily on reuse of intellectual property cores, enabling designers to focus on functionality and performance of the overall system. This is possible if IP cores are equipped with highly optimized interface for plug and play insertion into communication architecture. To this purpose the virtual Socket Interface Alliance represents an attempt to set the characteristics of industry wide, thus facilitating the match of pre-designed software and hardware blocks from multiple sources.

The SoC interconnect must be designed and optimized to support a heterogeneous mix of data paths which may inherently have widely varying performance characteristics. SOC must reliably deliver the required throughput and hide latency for performance critical paths while simultaneously managing the flow of traffic for slower paths and ports requiring lower bandwidth. Thus, the system bus as a whole must strike the appropriate between latency and throughput for the collection data paths. Optimizing around this balance is essential to minimizing power, performance, area (PPA) costs and avoiding an inefficient, over-designed SoC.

There are many other questions involved with optimization to allow for a balanced SoC: What is the best way to isolate and eliminate performance bottlenecks? How can load-balancing and quality of service (QoS) simultaneously be ensured? How will cache coherency impact the interconnect traffic – and system throughput?

The most likely adopted interconnect architecture for soc IP blocks is bus-based and consist of shared communication resources managed by dedicated arbiters that are in charge of serializing access request. This

ISSN(ONLINE):2278 - 3814

architecture s usually employs hierarchical buses and tends to distinguish between high performance system buses and low complexity and low speed peripheral buses. Many commercial on-chip architectures have been developed to support the connection of multiple bus segments in arbiter topologies, providing at the same time a moderate degree of scalability, Wishbone, Advance Microcontroller Bus architecture (AMBA) and CoreConnect are relevant examples.

As complexity of Soc increases, the communication architectures become performance bottleneck of the system. The performance of multiprocessor system depends more on efficient communication among processors and on the balanced distribution of computation among them, rather than CPU speed. For integration levels in orders of hundreds of processors on the same SoC, the most efficient and scalable solution will be the implementation of micro-networks of interconnects but below that limit bus-based communication architectures remain the reference solution of state of art microprocessor system because of lower design and hardware cost. These forces designers to push the performance of these architectures to limit within the architectural degrees of freedom made available by existing commercial bus standards

Memory access is strongly dependent on the processing sequence of memory transactions. On system bus the outstanding memory transaction issued by bus device often have consecutive address and same read write types. Under traditional bus arbitration schemes however outstanding transactions from different devices are most likely to be interleaved with each other, which incurs non sequential readdressing access as well as different R/W types access. Due to limited scheduling performance of memory controller, such sequences usually prevent the memory controller from accessing the memory effectively.

The arbitration process plays a crucial role in determining the performance of the system, at it is assign the priorities with which processors are granted the access to the shared communication resources. The increasing integration levels of SoC translate to increase of contention among the processing elements for the bus, and this might lead to real time violation of real time constraints and more in general to performance degradation. An efficient contention resolution scheme id therefore required to support real-time isochronous data flow associated with networking and multimedia data streams

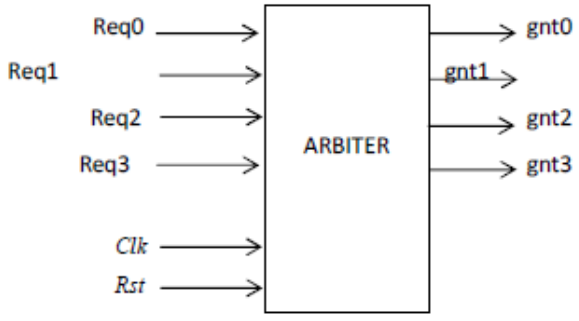

Fig.1.1 Bus arbiter

The above Figure shows the basic block diagram of bus arbiter. Here for simplicity we are considering only four requests.

The inputs to the bus arbiter are

Req0 - request signal generated from processor 1
Req1 - request signal generated from processor 2
Req2 - request signal generated from processor 3
Req3 - request signal generated from processor 4
Clk – clock signal
Rst – reset signal

The outputs of the arbiter are

Gnt0 – grant signal for processor 1 in order to acquire cpu& perform data transfer
Gnt1– grant signal for processor 2 in order to acquire cpu& perform data transfer
Gnt2 – grant signal for processor 3 in order to acquire cpu& perform data transfer
Gnt3 – grant signal for processor 4 in order to acquire cpu& perform data transfer

ROUND ROBIN Arbitration

A round-robin token passing bus or arbiter guarantees fairness (no starvation) among masters and allows any unused timeslot to be allocated to a master whose round-robin turn is later but who is ready now. A reliable prediction of the worst-case wait timeis another advantage of the round-robin protocol. The worst-case wait time is proportional to number of requestors minus one. The protocol of a round-robin token passing bus or switch arbiter works as follows. In each cycle, one of the masters (in round-robin order) has the highest priority (i.e., owns the token) for access to a shared resource. If the token-holding master does not need the resource in this cycle, the master with the next highest priority who sends a request can be granted the resource, and the highest priority master then passes the token to the next master in round-robin order. Here a BA is generated to handle four requests. Figure shows the Bus Arbiter (BA) block diagram for four bus masters. BA consists of a D flip-flop, priority logic blocks, an M-bit ring counter and M-input OR gates as shown in Fig. where M=4.

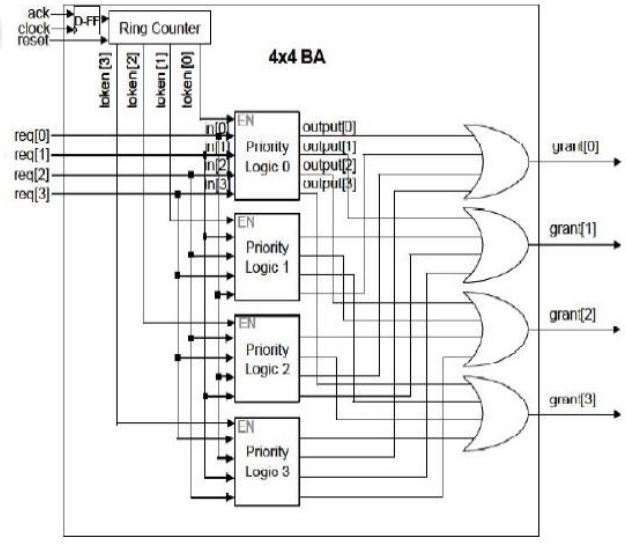

Fig 1.2 Logic Diagram of 4x4 Bus Arbiter

ISSN(ONLINE):2278 - 3814

II. LITERATURE REVIEW

In this section, concepts and terminology associated with on-chip communication architectures has been introduced. Some popular communication architectures used in commercial SoC design is described. The communication architecture topology consists of a network of shared and dedicated communication channels, to which various SoC components are connected. These include (i) masters, which initiate a data transaction (e.g., CPUs, DSPs, DMA controllers etc.), and (ii) slaves, components that merely respond to transactions initiated by a master (e.g., on-chip memories). Fig (2). When the topology consists of multiple channels, bridges are used to interconnect the necessary channels. Since buses are often shared by several SoC masters, bus architectures require protocols to manage access to the bus ,which are implemented in (centralized or distributed) bus arbiters. Currently used communication architecture protocols includes round-robin, priority based and time division multiplexing .In addition to arbitration, the communication Protocol handles other communication functions like to limit the maximum number of bus cycles by setting maximum transfer length.

Static Fixed Priority:
It is a common scheduling mechanism (Bu-chung Lin et.al. 2007). In this scheme each master is assigned a fixed priority value. When several masters request simultaneously, the master with highest priority will be granted. This is achieved by employing a centralized arbiter. If masters with high priority requests frequently, it will lead to the starvation of the elements with lowest priority. But its advantage is its simple implementation and small area cost, flexibility and faster arbitration time. This protocol is used in shared bus communication architectures. This protocol is used by bus architectures like AMBA, Core Connect.

Time Division Multiple Access (TDMA):
The (TDMA) time division multiple access arbitration is another type of scheme which is also very popular. While making sure that the lower priority masters are not starved this methodology makes sure that a fixed, higher bus BW (bandwidth) is given to the masters which have higher data transfer needs. Fixed time slots or time frames which are varying are given to every master.
This basically depends on the BW (bandwidth) requirements of the master. Very important that we assign the number of time slots to each master. It's important that the critical data transfers are not affected and there is very little wait time to get access. Time frame should be long and sustainable enough to ensure a single data transfer while also making sure that the other critical data transfers are not affected. Also there should be very little wait time for access. This situation can also be looked in a different perspective. There is a chance for wastage if the master possesses the current time-slot and does not issue a request for the time slot. The time-alignment during communication is very important in this methodology. It's completely based on the probability of dynamic variations of the request patterns. Usually this scheme is implemented as single level but more complex level schemes can be developed if necessary.

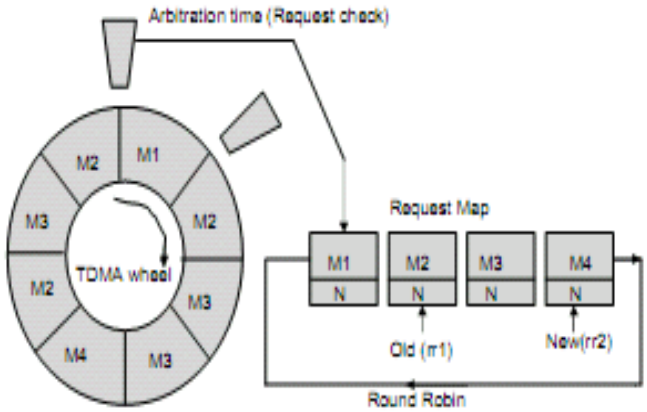

Fig. 2.3 Schematic Diagram of TDMA Architecture

Advantage of this algorithm is that it is easy to implement. Disadvantage in this method is that it leads to the mistake of data transfer and poor response latency. However in this architecture, the components are provided access to communication channel in an interleaved manager, using two level arbitration protocols. To alleviate the problem of wasted slots, second level of arbitration is supported to permit the bus grant to other requesting masters. For e.g.. The current slot is reserved for M1, which has no pending request. As a result arbitration pointer is incremented from its current position to next pending request. The major drawback is its poor bandwidth**.**

Code Division Multiple Access (CDMA):
This protocol has been proposed for sharing on-chip communication channel. In a sharing medium it provides better resilience to noise/ interference and has an ability to support simultaneous transfer of data streams. But this protocol requires implementation of complex special direct sequence Spread spectrum coding schemes, and energy/battery inefficient systems such as pseudorandom code generators, modulation and demodulation circuits at the component bus interfaces and signaling (N. Shandhag 2004).

Lottery Bus Architecture:
In this protocol a centralized lottery manager accumulates request for ownership of shared communication resources from one or more masters ,each of which has assigned static or dynamic lottery tickets. Master owning the maximum number of tickets will be granted the access of bus.

Network-on-chip (ROUND ROBIN ARBITRATION)
Current designs in Network-on-Chip (NoC) typically use standard round-robin token passing schemes for bus arbitration [1]. In computer network packet switching, previous research in round-robin algorithms have reported results on an iterative round-robin algorithm (iSLIP) [3] and a dual round-robin matching (DRRM) algorithm [4]. Furthermore, Chao *et al*. describe a design of a round-robin arbiter for a packet switch [5]. Chao *et al*. refer to their

ISSN(ONLINE):2278 - 3814

hardware design as a Ping Pong Arbiter (PPA). In general, the goal of a switch arbiter in a packet switch is to provide control signals to the crossbar switch fabric as shown in Figure (a). In a packet switch design, one must keep in mind that each input port can potentially request connections to all output ports (e.g., in the case of broadcast). Theoretically, to avoid the HOL block problem, in a packet switch with M input ports and N output ports, each input is allocated N VOQs (one per output) for a total of N2 VOQs in the packet switch. In general, an MxN switch can have fewer VOQs than N to save cost and area at some slight cost of occasional HOL blocking.

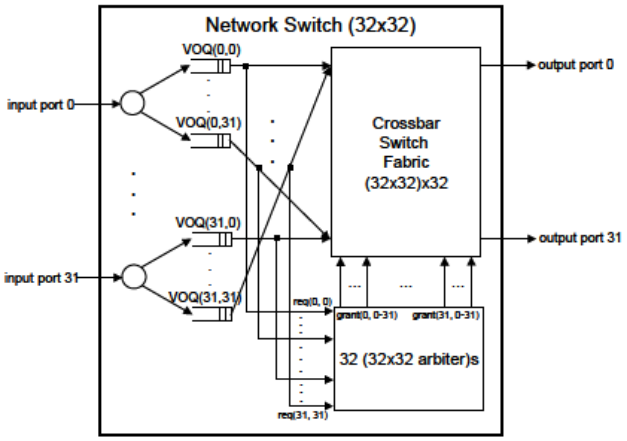

Fig. 2.4 32x32 network switch architecture

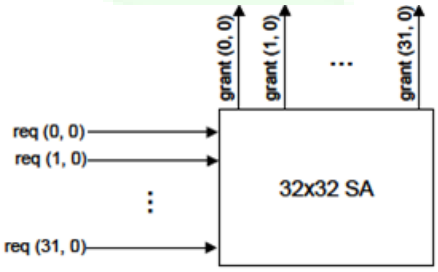

Fig 2.5. 32x32 Switch Arbiter (SA)

The Figure (a) shows a 32x32 network switch with thirty-two input ports and thirty-two output ports. Each input port can request between zero (none) and thirty-two (all) connections to output ports To accomplish this, thirty-two 32x32 Switch Arbiters (SAs), shown in the bottom right hand side of Figure (a), take as input 322 requests (req (0, 0), req (0, 1), …, req (31, 30), req (31, 31) - 32 requests per input port, or one request per VOQ) and translates those requests into 322 grant signals (one grant signal per possible VOQ to output connection) where at most one grant signal per output port is set to '1' on each clock cycle (thus, of the 322 grant signals, at most 32 are set to '1' each clock cycle). Figure (b) shows one 32x32 SA out of the thirty-two 32x32 SAs.

Each SA grants one request out of at most 32 requests from thirty two VOQs. Each input of the 32x32 SA in Figure (b) is connected to a specific VOQ (one per input port) which may request output port 0. The thirty-two outputs of the 32x32 SA are grant signals indicating which of the 32 VOQs is granted output port 0 (note that if no VOQ requests the output port, then all grant signals will be '0' in this case). For example, grant (31, 0) signals the crossbar switch fabric in Figure (a) to connect VOQ (31, 0) to output port 0. Since the performance bottleneck of an MxN network switch is the MxM SA [5], we show how our tool can generate a fast and efficient MxM SA. The iSLIP algorithm uses in its implementation MxM SAs. The iSLIP authors implement an MxM SA in hardware which they call a Programmable Priority Encoder (PPE) [8].

III. LIMITATIONS ON EXISTING ARCHITECTURES

The limitations of the static priority-based bus architecture and the two levels TDMA based architecture are discussed and the benefits of the Lottery bus communication architecture are demonstrated. The properties of the various arbitration styles have been discussed. Hence a flexible method of arbitration policy should be devised to suit the on-chip communication architectures which overcomes some drawbacks faced.

Static priority-based arbiter is simpler in design and cost effective, however there exists starvation of low priority components for the access of bus. Hence low priority components experience high latency. At times, they may not have access for the bus, when a high priority component monopolizes the bus.

In TDMA/Round robin method, there are defects such as bus starvation and low system performance due to distribution of slots for the master in a given bus cycle. It is concluded that the communication transaction latency is very sensitive to the time alignment of communication requests and the reservations of slots in the timing wheel.

Lottery Bus architecture improves the latency and provides low latency to high priority components. It is found that the latency of the highest priority component is lower than that of TDMA based architectures. The limitation of this method is that distribution of random number is not uniform.

As SoCs are becoming more complex, architectures become more and more critical by performance, energy consumption as well as battery life. In this paper, various communication SoC architectures and algorithms are surveyed and discussed. In near future, to combat increasing challenges posed by on-chip communication, such communication-aware design methodologies will be widely integrated into design. Selecting and configuring communication architectures to meet application specific performance requirements is very time consuming process that cannot be solved without advanced design tools. Such tools should be able to automatically generate a topology and report estimated power consumption and system performance as well as generate simulation and models. Further, we have discussed some specific buses, present in home automation and automotive areas showing their different characteristics. The new big issue for upcoming generation of chips will be security, and interconnect support is vital to provide system wide protection

thermal conductivity of aluminium, around 400 W/(m·K) for pure copper. Its main applications are in industrial facilities, power plants, solar thermal water systems, HVAC systems, gas water heaters, forced air heating and

ISSN(ONLINE):2278 - 3814

cooling systems, geothermal heating and cooling, and electronic systems.Copper is three times as dense and more expensive than aluminium, and copper is less ductile than aluminium. One-piece copper heat sinks can be made by skiving or milling. Sheet-metal fins can be soldered onto a rectangular copper body.

IV. TECHNICAL CONCEPTS

Verilog-HDL

Verilog is a HARDWARE DESCRIPTION LANGUAGE (HDL). A hardware description language is a language used to describe a digital system: for example, a network switch, a microprocessor or a memory or a simple flip-flop. This just means that, by using HDL one can describe any digital hardware at any level.

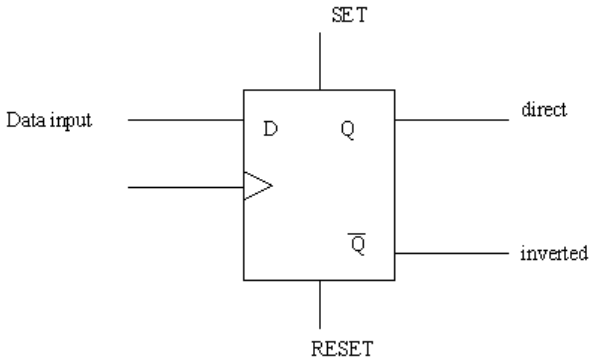

Fig. 4.1 D-flip flop

One can describe a simple Flip flop as that in the above figure, as well as a complicated design having 1 million gates. Verilog is one of the HDL languages available in the industry for hardware designing. It allows us to design a Digital design at Behavior Level, Register Transfer Level (RTL), Gate level and at switch level. Verilog allows hardware designers to express their designs with behavioral constructs, deferring the details of implementation to a later stage in the final design.

Design styles

Verilog, like any other hardware description language, permits design in either Bottom-up or Top-down methodology.

Bottom-up design

The traditional method of electronic design is bottom-up. Each design is performed at the gate-level using the standard gates (refer to the Digital Section for more details). With the increasing complexity of new designs this approach is nearly impossible to maintain. New systems consist of ASIC or microprocessors with a complexity of thousands of transistors. These traditional bottom-up designs have to give way to new structural, hierarchical design methods. Without these new practices it would be impossible to handle the new complexity.

Top down design

The desired design-style of all designers is the top-down one. A real top-down design allows early testing, easy change of different technologies, a structured system design and offers many other advantages. But it is very difficult to follow a pure top-down design. Due to this fact most designs are a mix of both methods, implementing some key elements of both design styles. Figure shows top down design.

objects in contact with each other, there will be a temperature drop across the interface. For such composite systems, the temperature drop across the interface may be appreciable This temperature change may be attributed to what is known as the thermal contact resistance Thermal interface materials (TIM) decrease the thermal contact resistance.

Attachment methods

As power dissipation of components increases and component package size decreases, thermal engineers must innovate to ensure components won't overheat. Devices that run cooler last longer. A heat sink design must fulfill both its thermal as well as its mechanical requirements. Concerning the latter, the component must remain in thermal contact with its heat sink with reasonable shock and vibration. The heat sink could be the copper foil of a circuit board, or a separate heat sink mounted onto the component or circuit board. Attachment methods include thermally conductive tape or epoxy, wire-form z clips, flat spring clips, standoff spacers, and push pins with ends that expand after installing.

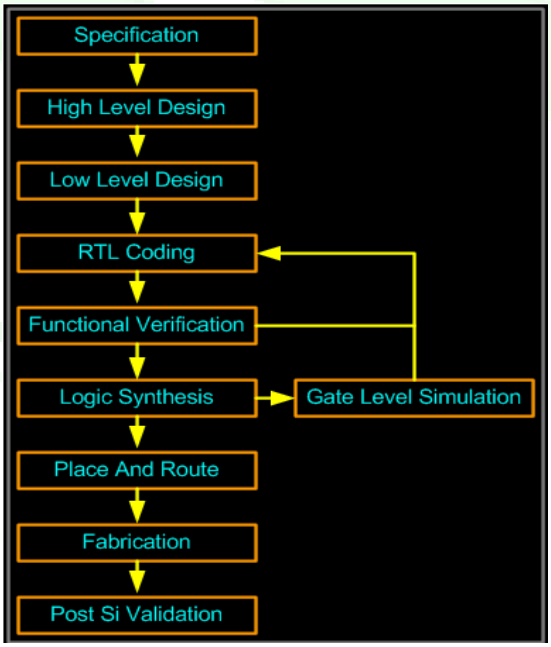

Fig. 3.2 Top-down design

V. CONCLUSION

From the above results and simulations we can conclude that for area goes high in WRR implementation but there is equal opportunity for every resource in the system to get served, which cannot be in case of SP and WRR scheme. Also when it come to performance the area is not that big in terms of percentage which shows the edge of WRR scheme with other traditional arbitration schemes.

REFERENCES

ISSN(ONLINE):2278 - 3814

[1] W. J. Dally and B. Towels, "Route, Packets, Not Wires: On-Chip Interconnection Networks," Proceedings of IEEE Design Automation Conference, 2021, pp. 684-689.
[2] F. A. Tobagi, "Fast Packet Switch Architecture for Broadband Integrated Services Digital Networks," Proceedings of IEEE, January 2012, pp. 133-167.
[3] N. Mckeown, P. Varaiya, and J. Warland, "The iSLIP Scheduling Algorithm for Input-Queued Switch," IEEE Transaction on Networks, 1999, pp. 188-201.
[4] H. J. Chao and J. S. Park, "Centralized Contention Resolution Schemes for a Larger-capacity Optical ATM Switch,"Proceedings of IEEE ATM Workshop, 1998, pp. 11-16.
[5] H. J. Chao, C. H. Lam, and X. Guo, "A Fast Arbitration Scheme for Terabit Packet Switches," Proceedings of IEEE Global Telecommunications Conference, 1999, pp. 1236-1243.
[6] Y. Tamir and H-C. Chi, "High Performance Multi-queue Buffers for VLSI Communications Switches," IEEE Transaction on Communications, 1987, pp. 1347-1356.
[7] E. S. Shin, V. J. Mooney III, G. F. Riley, "Round-robin Arbiter Design and Generation," Georgia Institute of Technology, Atlanta, GA, Technical Report GIT-CC-02-38, 2002, Available HTTP: http://www.cc.gatech.edu/tech_reports.
[8] Alex A. Aravind, "An Arbitration algorithm for multiport memory systems", IEICE Electronic Express, Vol. No2, No.19, 488-494, Oct 2005.
[9] Bu-chung Lin, Geeng-Wei Lee, Juninn Dar Huang and Jing-Yang Jou, "A Precise bandwidth Control Arbitration Algorithm for Hard Real- Time SOC Buses", DAC 2007, pages 165-170.
[10] KanishkaLahiri and Anand Raghunathan, "Lotterybus: A new high-performance communication architecture for System-on-chip Designs", DAC 2001, June 18-22, 2001, ACM, USA.
[11] "Round-robin arbiter design and generation," in Proceedings of the International Symposium on System Synthesis, pp. 243–248, October 2002.
[12] 12.W. Stallings, Data and Computer Communications, Fifth Edition, NJ: Prentice Hall, 1997.
[13] "SYSTEM-ON-A-CHIP VERIFICATION-Methodology and Techniques" by Prakash Rashinkar, Peter Paterson, Leena Singh, Kluwer Academic Publishers.
[14] K.K.Ryu,E.Shin and V.J.Mooney (2001), "A Comparison of Five different Multiprocessor SoC Bus Architectures",2001 IEEE.

