EXPLORING ARBITRATION SCHEMES OF SoC BUS ARCHITECTURES USING INTERACTING GENERALIZED SEMI MARKOV PROCESS MODEL

ULHAS DESHMUKH*, PRAFULL AGARWAL†, VINEET SAHULA, *Senior Member, IEEE*†
*Govt. Polytechnic College, Dhule, India.
Email: deshmukhur@gmail.com
†Department of Electronics & Comm. Engineering
Malaviya National Institute of Technology, Jaipur, India 302017.
Email: prafagr@gmail.com, sahula@ieee.org

Abstract: ***Diverse and complex communication between ever increasing component counts of System-on-Chip makes communication architecture a major performance determining candidate. This paper proposes a formal technique for system level performance analysis that can help the designer to select the appropriate arbitration scheme for a chosen bus-based communication architecture. We formulate the interacting generalized semi Markov process based model for a bus with arbitration. We mainly focus on building model for single shared bus architecture and explore arbitration along with various priority schemes viz. (i) fixed, (ii) lottery based and (iii) round robin. We describe the model of bus architecture using these arbitration schemes in the Stateflow component of MATLAB. Our modeling approach provides an evaluation and comparison of performance parameters viz. memory bandwidth, processing element utilization, average queue length at the memory and average waiting time seen by processing element for aforementioned arbitration schemes.***

Keywords: Interacting GSMP, arbitration, SoC communication

I. INTRODUCTION

A System-on-Chip (SoC) paradigm allows an integration of system components on a single silicon, and has widely been employed in diverse and emerging application areas like telecommunications and multimedia providing significant improvement in performance. The SoC design methodology addresses two major issues- (i) allocation and mapping of system computation to the set of high performance computing elements, and (ii) selection of appropriate communication architecture that provides communication between these computing elements. The computing elements, which are often called Intellectual Property (IP) blocks, are pre-designed and pre-verified in the form of either hardware or software. Hardware IPs may include embedded processor (DSP, CPU), memory blocks, Application Specific Integrated Circuits (ASICs), analog blocks and interface blocks. Software IPs may include Real Time Operating System (RTOS) and device drivers. In order to satisfy demands for higher performance and greater functionality, designer tend to use heterogeneous IPs. Therefore, communication requirement becomes not only diverse but also complex and hard to satisfy.

In the scenario, communication architecture has emerged as a major performance determining component of an SoC design. Hence, an early performance evaluation of communication architecture is the key to reduce design time, time-to-market and consequently the cost of the system. Bus-based and Network-on-Chip (NoC) communication architectures are the two major contemporary communication architectures employed for SoC and embedded applications. In this paper, we are concerned about bus-based communication architectures since they are still the most widely used in low end high volume SoC and embedded applications, due to their simplicity and popularity. However, even after selection of suitable architecture, trade off between parameters such as, number of buses, topologies, mapping of Processing Elements (PEs) and arbitration policy is crucial. Each of these factors has remarkable impact on system performance.

A. Outline of the paper

In this paper, we propose a formal technique for system level performance analysis of a single shared bus communication architecture. In particular, we explore various arbitration methods along with representative priority schemes viz. fixed, lottery based and round robin. In Section II, we present an overview of bus based architecture. In Section III, we briefly review the work related to performance estimation techniques for SoC communication. While in Section IV, analytical formulation based on Generalized Semi Markov Process (GSMP) for performance evaluation for single shared bus is presented. We propose model formulation using Interacting GSMP (IGSMP) for bus architecture with heterogeneous

processing elements in Section V. We discuss the results in Section VI and conclude in Section VII.

II. Background: SoC communication architecture

Communication architecture allows exchange of data and control signals between various system components. These architectures can be broadly classified as either bus-based or Network-on-Chip (NoC) based [1] or Hybrid bus-NoC architectures [2].

In SoC and embedded applications, bus based architectures have been in use for a long time as they are simple, require less area, cheaper and consume less power. Bus based architectures have different variants including single shared bus, network of shared buses or even dedicated buses. Moreover, bus based architectures not only meet communication requirements but also lead to cheaper design of low end portable applications, which are generally manufactured in high volume. Commercial bus based architectures such as Advanced Micro-controller Bus Architecture (AMBA) from ARM [3], CoreConnect from IBM [4] and Wishbone bus are widely used as Infrastructure IP (I^2P) [5]. In addition, custom bus architectures such as Bidirectional FIFO (First-In First-Out) Bus Architecture (BFBA), Global Bus Architecture version I (GBAVI), GBA version III (GBAVIII) amongst others have also been proposed in [6].

A. Single Shared Bus (SSB) architecture

The system bus is a simple shared communication architecture topology, which is commonly found in many commercial SoCs, where all the components of a system are mapped to a single shared bus. A bus arbiter periodically examines accumulated requests from the multiple master interfaces and grants access to a master as per the arbitration scheme specified in the bus protocol. A limitation of a single shared bus architecture is its lower bus bandwidth due to increasing load on global bus lines. Fig. 1 illustrates various bus based architectures. A single bus architecture as in Fig. 1(b), may have all the masters behaving identically or differently, and accordingly they are termed as homogeneous bus or heterogeneous bus architecture, respectively. Our present discussion is focused on such architectures only.

B. Hierarchical Bus Bridge (HBB) architecture

This topology consists of multiple shared buses interconnected by bridges to form a hierarchy as depicted in Fig. 1(c). This architecture provides higher bandwidth as compared to a single shared bus, providing concurrent transfers in various local bus domains. Commercial examples of such an architecture include the AMBA bus architecture and the CoreConnect architecture. Transactions across the bridge involves additional overhead, and during the transfer both buses remain inaccessible to other components. However, multiple word communication can proceed across the bridge in a pipelined manner. Such topologies typically offer large throughput improvements over the shared topology due to decrease in load per bus and the potential for parallel transactions on different buses.

1) Arbitration Schemes : Arbitration protocols determine the right to access shared resources viz. the shared bus and the shared memory. Some of the arbitration schemes employed in shared bus architectures to resolve the bus access conflicts, are indicated as follows. These arbitration schemes are also used in NoC architectures to allocate shared channels and buffers of the network switch, where these schemes are generally referred to as flow control policies.

1) **Fixed arbitration scheme-** Fixed or static priority based arbitration is a common arbitration policy used in the shared bus architectures. In this scheme, each master is assigned a unique priority level. When more than one master requests for bus access, the arbiter resolves conflict and grants, the access of the bus, to the master with the highest priority among the requesting ones.
2) **Lottery based-** In this arbitration scheme, priority levels to the masters are assigned randomly.
3) **Round robin-** Round robin arbitration can be classified as Fixed Slot Allocation (FSA) arbitration and Idle Recovery Slot Allocation (IRSA) arbitration. In FSA arbitration, the slot time for a master is reserved regardless of whether the master uses the time slot or not. In contrast, the IRSA round robin arbitration allows some other master to use the time allocated when it is no longer required to be used by the current master.

III. Related work

The objective of proposing performance model of an SoC communication architecture is to be able to predict performance parameters of that class of architectures, early in the design cycle. There are various metrics that can be used to evaluate performance of communication architectures during design space exploration. The performance of communication architecture has a significant impact on the performance of the entire system. One of the most vital performance parameters is throughput i.e. the amount of processing-work done per unit time. Nevertheless, this measure depends on many factors such as (a) the type of programs executed; (b) the characteristics of the processors and their interactions in terms of data dependencies and synchronization;

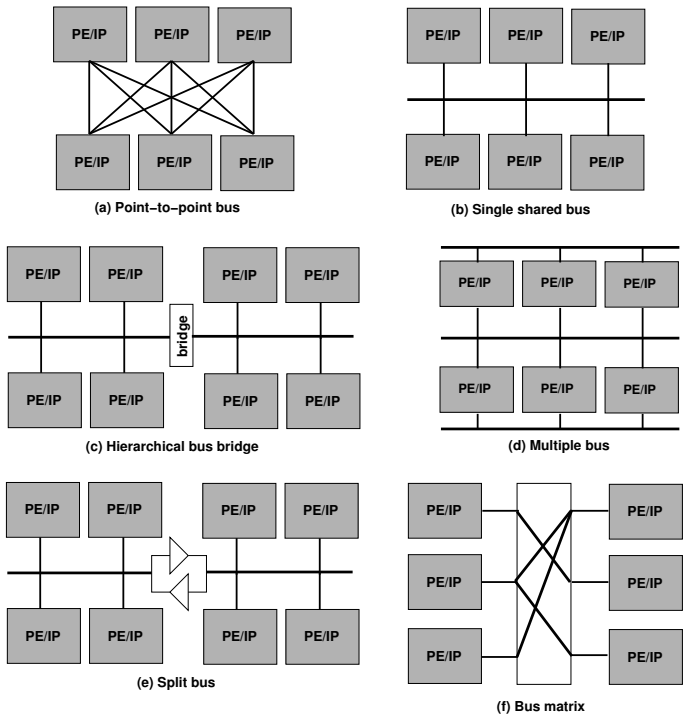


Figure 1. Various bus based architectures [7]

and (c) communication architecture & arbitration protocol employed. Relationship between these measures makes it difficult to isolate the influence of a particular parameter on overall system performance. Hence, more manageable measures have been used and reported in the literature viz. the average number of instructions executed per time unit, the average number of memory modules accessed per time unit, the average processor waiting time and average proportionate time during which memory is being accessed i.e. memory access bandwidth. All of these measures are related and the one that has been most preferred for design space exploration in past studies is effective memory bandwidth. In general, the effective bandwidth of a multiprocessor SoC architecture depends on the distribution of memory requests, communication time with the memory, communication delay, conflicts in the communication architecture and distribution of inter-arrival time of a processor requests. In the proposed work, we also use memory bandwidth (BW) as one of the performance measures for bus-based communication architectures under consideration. Besides, the other performance parameters, such as utilization factor of a PE, average waiting time seen by a processing element for accessing memory ($\overline{W}$) and average queue length at the memory ($\overline{L}$) are also evaluated.

The area of communication architecture synthesis and performance modeling has been widely explored and reported in the literature. Although, we have attempted to review many of the major approaches as possible from the literature, it is by no means comprehensive. Performance evaluation of SoC communication architecture is generally carried out after hardware-software partitioning has been performed, and is then followed by architecture allocation and mapping. However, authors in their work in [8], [9] has considered communication architecture selection simultaneously during the synthesis of the computation parts of a system and the mapping step. In their approach, they have demonstrated that communication overhead has significant influence on the mapping decision, which otherwise ends up to sub-optimum system architecture. The performance evaluation approaches for communication architectures can generally be classified into three categories-analytical approaches [8]–[11], simulation based approaches [12], [13] and hybrid approaches [14], [15]. Simulation approach uses communication models at various levels of abstraction. Pasricha et al. in [13] present simulation by abstracting the system at cycle count accurate at transaction boundaries. Zhu et al. in [12] propose formal concurrent modeling approach based on Operation State Machine (OSM) for entire system comprising of computation and communication. Dey et al. in [10] introduce worst case static performance analysis of the system comprising concurrent communicating processes. Deshmukh et al. in [16], [17] have proposed formal modeling approach based on GSMP model and evaluation based on Analytical Formulation of Model Equations (AFOME) to estimate performance metrics of a shared bus architecture. Two phase hybrid approach encompasses both simulation and analytical approaches [14], [15] to exploit benefits of both. In

[14], Lahiri et al. perform initial co-simulation with the abstracted communication and then analytical analyzed by specifying communication architecture. Queuing analysis in [15] uses analytical approach to prune the design space and then simulate entire system of the selected architectures. Application of interacting Markov chains for modeling cellular signal processing in biological cell have been presented in [18]. Asavathiratham et al. [19] propose influence model comprising network of interacting Markov chains. Stochastic Automata Network (SAN) model described by Plateau and Atif in [20] is quite similar to interacting Markov chains.

A. *Contribution of the paper*

The main contribution of the paper lies in the proposal for system level framework for performance estimation of communication architecture, based on IGSMP model. We mainly focus on building model for homogeneous shared bus architecture and explore arbitration methods along with various priority schemes viz. fixed, lottery based and round robin. We present high level simulation model of each of the aforementioned arbitration schemes based on IGSMP model in the Stateflow component of MATLAB. Our modeling approach provides estimation of performance parameters viz. estimation of memory BandWidth (BW), Processing element Utilization factor (PU), average queue length ($\overline{L}$) at memory and average waiting time ($\overline{W}$) seen by a PE. The model parameters and other input parameters to the model are shown in Table I.

IV. GSMP MODEL FOR HOMOGENEOUS SHARED BUS ARCHITECTURE

In this section, we review GSMP model of a PE mapped to shared bus architecture, assuming that all PEs are homogeneous. We present brief discussion of Analytical Formulation of the Model Equations (AFOME) approach [16] for evaluation of performance metrics. In a Markovian process, the probability distribution of time spent in each state (also called delay) is assumed to be exponential. A Semi Markov process (SMP) is generalization of a Markov process, which allows random delay for the states. A semi Markov process has a non-negative state space $\{0, 1, 2, ..., K\}$, and it can be in any one of the states. Each time the SMP enters a state i ($0 \leq i \leq K$), it spends mean sojourn time η_i before making a transition to the state j with probability p_{ij} [21]. If the SMP has an underlying irreducible embedded Markov chain that consist of ergodic states, then limiting probability P_i of being found in state i is intuited as weighted average of π_i, where π_i is the limiting probability of reaching state i of the underlying embedded Markov chain, given by (1).

$$P_i = \frac{\pi_i \, \eta_i}{\sum_{j=0}^{N} \pi_j \, \eta_j}. \quad (1)$$

A Generalized Semi Markov Process (GSMP) is thus a generalization of an SMP with an arbitrarily distributed time. The shared bus architecture consisting of N processing elements (PEs), PE_1, PE_2,...,PE_N competing for the use of a bus (BUS), for accessing shared memory (MEM) is depicted in Fig. 2. Communication between PEs and shared memory of the system is assumed to be synchronous with system bus cycle. Arbiter of N-user one-server type is employed to resolve the bus access conflict among N number of PEs. All the PEs are assumed to be identical and hence SMP model of a PE suffices for functional modeling [11].

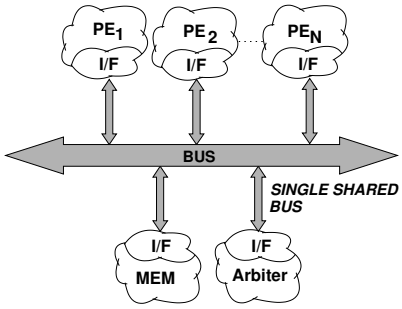


Figure 2. Shared bus communication architecture: Homogeneous PEs

Let's consider a scenario when a PE mapped to the BUS attempts to access MEM. Fig. 3 shows GSMP model of a PE in this scenario. It has four states. In computing state (*state 0*) a PE performs computation. Accessing state labeled as *state 1*, corresponds to the situation when a PE is accessing MEM. In full waiting state labeled as *state 2*, a PE waits for MEM for full connection time of another PE, which is currently accessing MEM; while in residual waiting state labeled as *state 3*, a PE waits for MEM for residual connection time of that another PE currently accessing MEM. In each state, model spends random amount of time with mean value η_i, called mean sojourn time of i^{th} state ($i = 0, 1, 2, 3$).

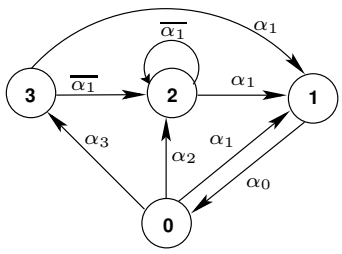


Figure 3. GSMP model of a PE in the homogeneous shared bus architecture as in Fig. 2

The input parameters to the model are number of PEs (N), the mean think time of PE ($\overline{T}$) and the

Table I
DESCRIPTION OF PARAMETERS AND NOTATIONS USED

Input Parameters	Description
N	Number of processing elements
$\overline{T}$	Mean value of computing time
$\overline{C}$	Mean value of connection time
$\overline{C^2}$	Second moment of connection time

Notation	Description
R	Probability that a PE generates a request
η_i	Mean sojourn time of i^{th}state
p	Probability of at least one request
$BUSY$	Probability of bus or memory busy
WIN	Probability of wining request
α_i	Transition probability to i^{th}state

first as well as second moment of connection time between a PE and the memory ($\overline{C}$, $\overline{C^2}$). For computation of transition probabilities, we define following parameters. The probability of leaving i^{th} state is μ_i. The probability that a PE generates a request is $R = \mu_0 + \mu_2 + \mu_3$. Probability of at least one request is $p = [1-(1-R)^N]$. Sojourn time of the states are $\eta_0 = \overline{T}$, $\eta_1 = \eta_2 = \overline{C}$ and $\eta_3 = \frac{(\overline{C^2}-\overline{C})}{(2(\overline{C}-1)}$. The probability of wining arbitration is $WIN = \frac{p}{NR}$. The term $BUSY = (N-1)(\overline{C}-1)\mu_1$ is the probability that a PE finds bus or memory busy.

GSMP has transition from *state 0* to *state 1* when MEM and BUS both are idle with the probability $(1-BUSY)^2$ and request wins arbitration with the probability WIN. Alternatively, GSMP has transition from *state 0* to *state 2* when MEM and BUS are idle and local request does not win arbitration with the probability $(1-WIN)$. Similarly, GSMP makes transition from *state 0* to *state 3* if at the time of request either (i) MEM is busy or (ii) MEM is idle but BUS is being used by PEs mapped to other bus. Pending requests of *state 2* and *state 3* are reconsidered after their mean sojourn times and and if selected, the GSMP enters in *state 1*, otherwise it enters in the *state 2*. From *state 1*, the process always returns to *state 0*. State transition probabilities are deduced from discussion of the model and are given in (2).

$$\alpha_j = \begin{cases} 1 & j=0 \\ (1-BUSY)^2\, WIN & j=1 \\ (1-BUSY)^2(1-WIN) & j=2 \\ BUSY(2-BUSY) & j=3 \end{cases} \quad (2)$$

We iteratively solve model equations to compute steady state probabilities of the model viz. $P_0 = \eta_0\alpha_1 R$, $P_1 = \eta_1\alpha_1 R$, $P_2 = \eta_2\, R(\alpha_2 + \alpha_3\overline{\alpha}_1)$, $P_3 = \eta_3 R\alpha_3\alpha_1$, where $\overline{\alpha_1} = (1-\alpha_1)$. Performance parameters of a shared bus architecture are deduced in the (3).

$$\begin{aligned} BW &= NP_1 & \overline{W} &= (\eta_2\alpha_2 + \eta_3\alpha_3)/\alpha_1 \\ PU &= P_0 + P_1 & \overline{L} &= N(P_2+P_3) \end{aligned} \quad (3)$$

V. MODEL FORMULATION FOR HETEROGENEOUS SHARED BUS ARCHITECTURE

The concept of interaction in various forms has been previously presented in the literature. Depending on applications, interaction models have different structures such as, influence model [19] and interacting model [18]. In this section, applying the same modeling concept, we propose IGSMP model, for evaluating performance of the shared bus architecture with heterogeneous PEs. We use bottom up approach to formulate IGSMP model from the underlying interacting embedded Markov model [18]. Functional characteristic of the abstract system shown in Fig. 2, may be described in terms of its constituent components, that are PEs referred to as modules or nodes. These modules are heterogeneous and behave concurrently to execute system functionality, while communication between them and the memory is sequential, carried over the bus.

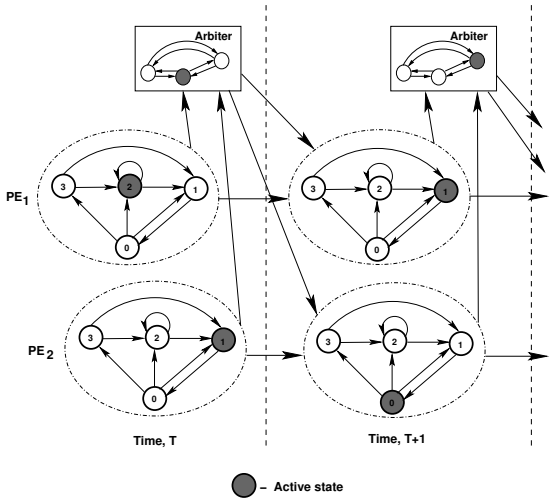


Figure 5. Bus architecture with heterogeneous PEs: IGSMP model unrolled in time T and T+1

A. IGSMP model

We consider an IGSMP model consisting of a network of N interacting GSMPs, X_1, X_2,..., X_N corresponding to the dynamics of PE_1, PE_2,...,PE_N respectively. Fig 4 shows IGSMP model for shared bus architecture.

The IGSMP model when unrolled in time viz. for time n and $n+1$ instances, is shown in Fig. 5. GSMP model of each of the PEs has four states, as discussed in Section IV.

GSMP model at each module has an underlying embedded Markov model with Markovian property, that is the probability that the module X_p at time n is in state $X_p[n]$ is given by: $P(X_p[n] = j | X_p[n-1] =$

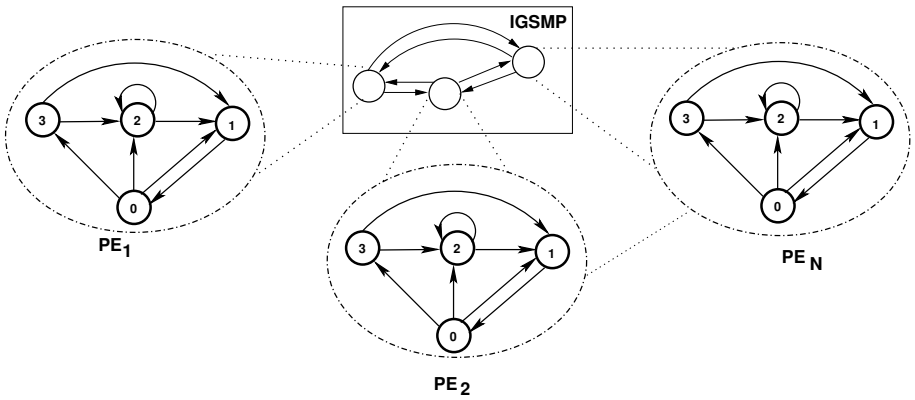


Figure 4. Interactive GSMP model for heterogeneous bus architecture

$i, ..., X_p[0] = m) = P(X_p[n] = j | X_p[n-1] = i) = p_{ij}^{X_p}[n]$. In the absence of interaction, each module behaves independently and transition probability of embedded Markov model X_p is written as: $p_{ij}^{X_p}[n] = q_{ij}^{X_p}[n]$. Here, $q_{ij}^{X_p}[n]$ is independent of all the states of other modules in the network. So, limiting probability P_i of being in state i of the IGSMP model, which has independent GSMP model is given in terms of limiting probabilities, π_i of the underlying model and sojourn times, η_i as:

$$P_i = \frac{\pi_i \, \eta_i}{\sum_{j=0}^{N} \pi_j \, \eta_j}. \quad (4)$$

But, as shared memory MEM and BUS behavior of PEs have influences on each other, the interaction between different modules are defined by influences of states in one module onto the transition probabilities in another module. Two types of influences are possible, either positive or negative. In our IGSMP model, only negative influences are present. In particular, if state ℓ of module X_r influences the transition probability $p_{ij}^{X_p}$ from state i to the state j in the module X_p, then $p_{ij}^{X_p}$ at time n due to influence of state ℓ in X_r, $\{p_{ij}^{X_p}[n]\}_{[\ell, X_r]}$ is written as follows [18].

$$\{p_{ij}^{X_p}[n]\}_{[\ell, X_r]} = \alpha_{r,\ell} f\{P(X_r[n-1] = \ell)\}^{\beta_{r,\ell}} \quad (5)$$

Here, $P(X_r[n-1] = \ell)$ is the probability of the module X_r being in state ℓ at time $n-1$, $0 \leq \alpha_{r,\ell} \leq 1$ and $0 < \beta_{r,\ell}$ are constants, and $f\{x\} = x$ if the influence is positive, otherwise $f\{x\} = 1 - f\{x\}$ if the influence is negative [18]. Equation (5) is only the influence of state ℓ of module X_r on the transition probability $p_{ij}^{X_p}$ of the module X_p. Similar equations can also be written for influences of other states of all the modules, if any. These influences are otherwise called functional transitions in the SAN formulation elsewhere [20]. We adopt *fading model* [18] to combine all the influences.

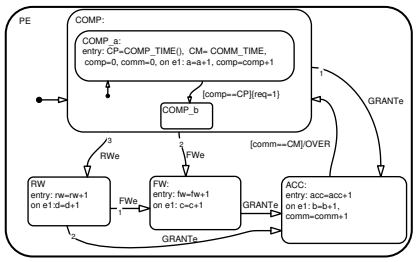


Figure 6. Stateflow representation of GSMP structure of a PE, of Fig. 2

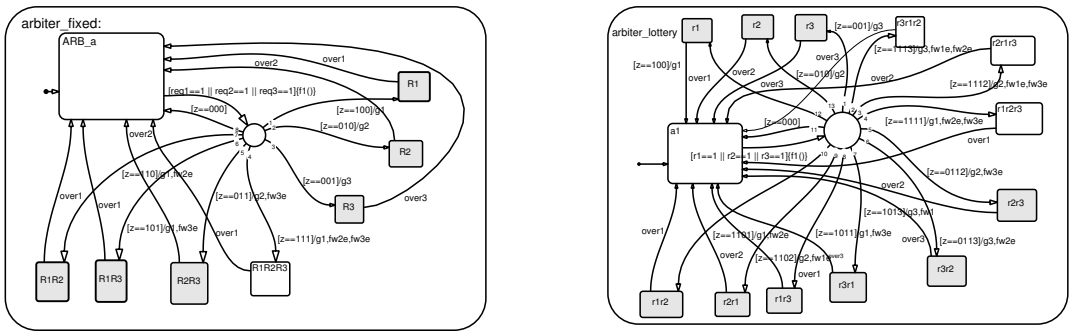
Thus, limiting probability P_i of being in state i of the IGSMP model under the influence of other GSMP models is given by equation (6), where, π_i^* and η_i^* are limiting probability and sojourn time of the underlying interacting model.

$$P_i^* = \frac{\pi_i^* \, \eta_i^*}{\sum_{j=0}^{N} \pi_j^* \, \eta_j^*} \quad (6)$$

B. Simulation methodology

In this section, we describe methodology to specify, simulate and analyze IGSMP model of a shared bus architecture with various arbitration schemes in the Stateflow component of MATLAB [22]. We construct a library of generic blocks that can be combined in a bottom up fashion, to model communication architectures with fixed priority, lottery and round robin arbitration policies. The generic building blocks model different types of resources in the system, e.g. PEs and arbiters with aforementioned arbitration policies. To integrate the system, we instantiate building blocks from the library and provide interface between them. This makes exploration of alternative arbitration schemes possible in less amount of time.

In Fig. 6, we depict generic building block showing state structure of a PE in the Stateflow. It has



(a) (b)

Figure 7. State structure of arbiter in StateFlow- (a) fixed priority scheme (b) Lottery based scheme

four states as explained in the Section IV. State *COMP* is the computing state of a PE, which lasts for random computing time $\tilde{T}$. The syntax of the state consist of two major elements- the name of the state and various state actions (*entry*, *during* and *exit* state action and on *event* action). The state entry action of *COMP* state of all the PEs generates non-deterministic computing time $\tilde{T}$ and communication time $\tilde{C}$ in each communication transaction. For generation of $\tilde{T}$ and $\tilde{C}$, we invoke MATLAB m-function having predefined distribution, through graphical function *COMP_TIME()* and *COMM_TIME()*. The GSMP model of a PE transits from *COMP* state to accessing state labeled as *ACC*, when arbiter model generate *GRANTe* event. Arbiter model generates events viz. *GRANTe*, *FWe* and *RWe* depending on the priori status of the global system. Similarly, we develop generic blocks for arbiters with different arbitration schemes- fixed priority as well as lottery based arbiter in the Stateflow as are shown in Fig. 7(a) and (b).

Fig. 7(a), depicts state structure of fixed priority arbiter with three PEs. It goes through sequence of three states- (i) ARB_a is idle state; (ii) arbiter enters 'pre-grant' state upon receiving requests from one or more PEs; (iii) depending upon how many PEs have requested, arbiter switches from 'pre-grant' state to any one of R1, R2, R3 (when only single PE has requested), or R1R2, R2R3 or R1R3 (when two PEs raise simultaneous requests) or R1R2R3 (when all three PEs have raised requests). The three bits of vector $z = z_1 z_2 z_3$ are reserved for three PEs, where a '1' in place indicates that that-numbered PE has raised request; and a '0' indicates absence of such request. We have presumed that highest priority is assigned to requests from PE_1. Thus, whenever, higher priority PE is grated access to the bus, other PEs are made to wait for full-waiting-time. In Stateflow representation, this is indicated by events 'fw2e' (fw3e) meaning that $PE_2(PE_3)$ will be kept in full-waiting state.

In lottery based scheme for arbitration, Fig. 7 (b), the vector $z = z_1 z_2 z_3 z_i$ has one extra bit z_i at the tail-end indicating which of the PEs has been dynamically assigned access priority over the others using lottery based arbitration.

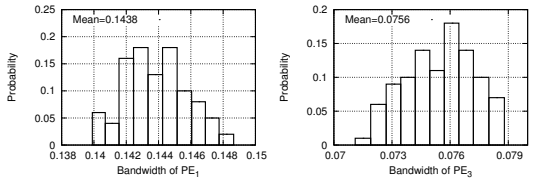
VI. Results

We capture IGSMP model of heterogeneous shared bus architecture with fixed priority, lottery based and round robin arbitration schemes in Stateflow. An architecture has three PEs- PE_1, PE_2 and PE_3. In case of fixed arbitration scheme, we assign the highest priority to PE_1 then PE_2 and the lowest to PE_3. Input parameters to the model are- (i) random communication times of PEs viz. $\tilde{C_1}$, $\tilde{C_2}$ and $\tilde{C_3}$; and (ii) random computation times namely, $\tilde{T_1}$, $\tilde{T_2}$ and $\tilde{T_3}$. These times are generated by using MATLAB m-functions with generalized distribution. Simulation has been performed on a Linux-workstation with Intel Pentium-HT processor having 1 GB RAM.

Various performance parameters of PEs viz. Band-Width (BW), Processing element Utilization (PU) and queue length at the memory ($\overline{L}$) have been estimated for aforementioned arbitration schemes. Table II shows results obtained from simulation of IGSMP model in terms of mean values of these parameters. The mean values of the parameters are equivalent fraction corresponding to relative value with respect to the total value and is displayed in percentage, viz. the bandwidth available to PE_3 in fixed priority arbitration scheme is 0.076 out of total available memory bandwidth of 0.34, which amounts to 8.43%. The fact that total available memory bandwidth is not 100% can be attributed to the fact that the latency increases with decreasing priority of PEs in fixed arbitration scheme. The latency of arbitration is constant with round robin scheme.

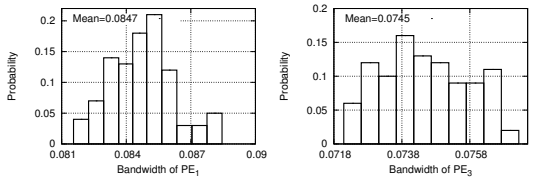
Table II
PERFORMANCE METRICS COMPARISON [BW- MEMORY BANDWIDTH, PU- PROCESSOR UTILIZATION, $\overline{L}$ - AVERAGE QUEUE LENGTH]

	Fixed			**Lottery**			**Round robin**		
	BW (%)	PU (%)	$\overline{L}$ (%)	BW (%)	PU (%)	$\overline{L}$(%)	BW (%)	PU (%)	$\overline{L}$ (%)
PE_1	48.45	48.21	15.86	24.54	24.49	47.66	33.21	33.27	33.01
PE_2	43.29	43.32	21.99	42.47	42.51	23.89	33.27	33.43	32.79
PE_3	8.43	8.46	62.13	32.98	32.99	28.43	33.50	33.29	34.18



(a) (b)

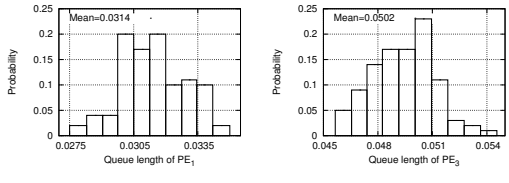
Figure 8. Fixed priority scheme: histogram of available BW (a) for PE_1 (b) for PE_3



(a) (b)

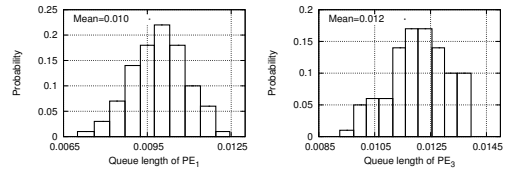
Figure 9. Round robin scheme: histogram of available BW (a) for PE_1 (b) for PE_3

The columns denote performance metrics along with arbitration schemes, whereas rows list PEs. From the table we make following observations. (1) Memory bandwidth of PE_3, under fixed priority scheme is minimum (8.43 %), being the lowest priority, while the highest priority PE_1 receives maximum bandwidth (48.45 %), 5 times larger than PE_3. In column 4, we observer that bandwidth allocated in lottery based arbitration scheme is better than fixed arbitration, but not guaranteed. Under round robin arbitration scheme, all PEs get guaranteed equal share of bandwidth. Moreover, maximum total bandwidth is only 42 % achieved in lottery based arbitration scheme from which, we infer that more PEs can be mapped to the bus to utilize idle time of the memory (58 %). Probability mass function (PMF) of bandwidth for PE_3 under fixed and round robin arbitration schemes are shown in Fig 9(a) and 9(b), with mean observed 0.02 and 0.12 respectively. Similar results are obtained for PU also. (2) Queue length of PE_3 (62.13 %) in fixed arbitration scheme indicates that PE_3 spends comparatively more time in full and residual waiting states, before getting access to the memory. This is quite intuitive that, latency of lower priority masters is higher as compared to their high priority counterparts. In round robin arbitration scheme, all PEs spend approximately equal amount of time in waiting states. Fig 10(a) and 10(b) depict



(a) (b)

Figure 10. Fixed priority scheme: histogram of average queue length $\overline{L}$ (a) for PE_1 (b) for PE_3



(a) (b)

Figure 11. Round robin scheme: histogram of average queue length $\overline{L}$ (a) for PE_1 (b) for PE_3

PMF of queue length for PE_3 in fixed and round robin arbitration schemes, with mean observed 0.30 and 0.20 respectively.

VII. CONCLUSIONS

We have presented a framework for faster evaluation of performance of bus based SoC communication architectures with variety of arbitration Schemes. Traditionally, such Markov chain formulation for concurrent systems would result into a large state-space, solution of which would have been very time-intensive. Whereas we have modeled interaction among concurrent PEs using formal modeling techniques such as IGSMP model, thus avoiding the computation based on flattened large-Markov state-space. The methodology can be used for efficient design space exploration using different buses, alternate mapping of PEs and arbitration schemes to available buses etc.

The results obtained demonstrate that the round robin arbitration scheme provides guaranteed equal share of bandwidth allocation to all the system components (PEs) as compared with fixed and lottery based arbitration schemes, albeit at the cost of increased latency. Performance metrics of lottery based arbitration schemes are also comparable with those of round robin scheme, but not guaranteed. In case of fixed priority scheme, starvation of lower priority components is observed. Starvation becomes worse

for lower priority PEs. Although, higher priority masters observe lower latency. Thus, round robin arbitration scheme offers an attractive alternative compared with other schemes considered here.

ACKNOWLEDGMENT

Authors gratefully acknowledge the financial support received for this research from Ministry of Comm. & IT, Govt. of India under project *SMDP-VLSI-2*.

REFERENCES

[1] L. Benini and G. D. Micheli, "Networks on chips: A new SoC paradigm," *IEEE Computer Magazine*, vol. 35, pp. 70–78, Jan. 2002.

[2] T. Dumitras, S. Kerner, and R. Marculescu, "Enabling on-chip diversity through architectural communication design." in *ASP-Design Automation Conf.*, 2004, pp. 799–805.

[3] AMBA 2.0 Specification [Online] Available: http://www.arm.com/armtech/AMBA.

[4] CoreConnect Bus Architecture, [Online] Available: http://www.chips.ibm.com/products/coreconnect/.

[5] P. P. Pande, C. Grecu, M. Jones, A. Ivanov, and R. Saleh, "Performance evaluation and design trade-offs for network-on-chip interconnect architectures," *IEEE Transaction on Computers*, vol. 54, no. 8, pp. 1025–1040, Aug. 2005.

[6] K. K. Ryu and V. Mooney-III, "Automated bus generation for multiprocessor SoC design," *IEEE Transaction on Computer-Aided Design of ICs and Systems*, vol. 23, no. 11, pp. 1531–1549, Nov. 2004.

[7] N. Dutt, K. Banerjee, L. Ben, K. Lahiri, and S. Pasricha, "Tutorial 5: SoC communication architectures: Technology, current practice, research, and trends," in *IEEE* 20^{th} *International Conference on VLSI Design*, Jan. 2007, pp. 8–8.

[8] P. V. Knudsen and J. Madsen, "Integrating communication protocol selection with partitioning in hardware/software codesign," in *Proceedings of 11th International Symposium on System Synthesis*, Dec 1998, pp. 111–116.

[9] P. Voigt Knudsen and J. Madsen, "Communication estimation for hardware/software codesign," in *Proceedings of the Sixth International Workshop on Hardware/Software Codesign. (CODES/CASHE)*, Mar 1998, pp. 55–59.

[10] S. Dey and S. Bommu, "Performance analysis of a system of communicating processes," in *Proc. IEEE/ACM Int. Conf. Computer-Aided Design(ICCAD)*, 1997, pp. 590–597.

[11] T. N. Mudge and H. B. Al-Sadoun, "A semi Markov model for the performance of multiple-bus systems," *IEEE Trans Computer*, vol. C-34, no. 10, pp. 768–783, Oct 1985.

[12] X. Zhu, W. Qin, and S. Malik, "Modeling operation and microarchitecture concurrency for communication architectures with application to retargetable simulation," *IEEE Trans. VLSI Syst.*, vol. 14, no. 7, pp. 707–716, July 2006.

[13] S. Pasricha, N. Dutt, and M. B. Romdhane, "Using TLM for exploring bus based SoC communication architectures," in *Proc.Int.Conf. ASAP*, 2005.

[14] K. Lahiri, A. Raghunathan, and S. Dey, "System-level performance analysis for designing on-chip communication architecture," *IEEE Tran. Computer Aided Design*, vol. 20, no. 6, pp. 768–783, June 2001.

[15] S. Kim, C. Im, and S.Ha, "Schedule-aware performance estimation of communication architecture for efficient design space exploration," *IEEE Trans.VLSI Sys.*, vol. 13, no. 5, pp. 539–552, May 2005.

[16] U. Deshmukh and V. Sahula, "Analytical performance estimation from generalized semi Markov process model for hierarchical bus bridge based SoC communication architecture," in *Proceedings of 20th IEEE International Conference on Microelectronics (ICM)*, Dec. 2008.

[17] ——, "HCFG based approach for evaluation of SMP model for System-on-Chip communication," in *Proceedings of 12th VLSI Design And Test Symposium (VDAT)*, July 2008, pp. 432–433.

[18] M. R. Said, A. V. Oppenheim, and D. A. Lauffenburger, "Modeling cellular signal processing using interacting Markov chains," in *IEEE Int.Conf. on ASSP*, vol. 6, April 2003, pp. 41–44.

[19] C. Asavathiratham, S. Roy, B. Lesieutre, and G. Verghese, "The Influence Model," *IEEE Control System Magazine*, pp. 52–64, December 2001.

[20] B. Plateau and K. Atif, "Stochastic automata network for modeling parallel systems," *IEEE Trans. Software Eng.*, vol. 17, no. 10, pp. 1093–1108, 1991.

[21] S. M. Ross, *Applied Probability Models with Optimization Applications*. San Francisco CA: Holden-Day, 1970.

[22] Stateflow, "MathWorks, Inc. (2007). [online]. available: http://www.mathworks. com/products/stateflow."

[23] G. Varatkar and R. Marculescu, "On-chip traffic modeling and synthesis for MPEG-2 video applications," *IEEE Transaction on Very Large Scale Integrated (VLSI) Systems*, vol. 12, no. 1, pp. 108–119, Jan. 2004.

[24] K. Sekar, K. Lahiri, A. Raghunathan, and S. Dey, "Dynamically configurable bus topologies for high-performance on-chip communication," *IEEE Transaction on Very Large Scale Integrated (VLSI) Systems*, vol. 16, no. 10, pp. 1413–1426, Oct. 2008.

Ulhas Deshmukh was born in Maharashtra, India in 1969. He obtained a Bachelor of Engineering in Electronics with distinction in 1991, from the S.S.V.P.S. College of Engineering, Dhule, India. He obtained Master of Engineering in Instrumentation with distinction from the S.G.G.S. College of Engineering and Technology, Nanded, India in 1996. He is currently on the board of faculty in the Department of Electronics and Communication Engineering at Govt. Polytechnic, Dhule. During 2005-2009, he was a research scholar at Department of Electronics and Communication Engineering, Malaviya National Institute of Technology Jaipur, through the "Quality Improvement Program" scheme of Ministry of Human Resource Development (MHRD), Govt. of India. His research interests are in the areas of VLSI Design.

Prafull Agarwal was born in U.P., India in 1986. He obtained a Bachelor of Technology in Electronics and Communication Engineering in 2007, from U.P. Technical University, India. He obtained Master of Technology in VLSI Design with Gold-medal from the National Institute of Technology, Jaipur, India in 2009. He is currently on the board of faculty in the Department of Electronics and Communication Engineering at educational institute in Agra, UP. His research interests are in the areas of VLSI Design especially system level design & modeling.

V. Sahula received his Bachelor of Engineering degree in Electronics with honors from Malaviya National Institute of Technology, Jaipur in 1987. He obtained Master of Technology degree in Integrated Electronics & Circuits in 1989 and Ph.D. in Electrical engineering in 2001, both from Indian Institute of Technology, Delhi. He joined as faculty member at Department of Electronics and Communications Engineering, National Institute of Technology Jaipur, India where he is currently an Associate Professor. His research interests are in the areas related to high level design, modeling & synthesis of analog & digital systems, and CAD for VLSI. He has more than 30 journal and refereed conferences papers to his credit. He has served on the Technical programme committee of the VLSI Design and Test Symposium held in India (1998-2010). He has also served on organizing committee as fellowship-chair of 22^{nd} IEEE International Conference on VLSI Design, 2009 India. He is a senior member of IEEE, Life Fellow of IETE, Life member of IMAPS and member of VLSI Society of India and ACM SIGDA.