Stochastic Automata Network for Performance Evaluation of Heterogeneous SoC Communication

Ulhas Deshmukh
Lectrurer in ECE, Govt. Polytechnic, Dhule, India &
Research Scholar MNIT, Jaipur, India
Email: deshmukhur@gmail.com

Vineet Sahula, *SMIEEE*
Associate Professor, Deptt. of Electronics & Comm. Engg.
Malaviya National Institute of Technology, Jaipur, India
Email: sahula@ieee.org

***Abstract*— To meet ever increasing demand for performance of emerging System-on-Chip (SoC) applications, designer employ techniques for concurrent communication between components. Hence communication architecture becomes complex and major performance bottleneck. An early performance evaluation of communication architecture is the key to reduce design time, time-to-market and consequently cost of the system. Moreover, it helps to optimize system performance by selecting appropriate communication architecture. However, performance model of concurrent communication is complex to describe and hard to solve. In this paper, we propose methodology for performance evaluation of bus based communication architectures, modeling for which is based on modular Stochastic Automata Network (SAN). We employ Generalized Semi Markov Process (GSMP) model for each module of the SAN that emulates dynamic behavior of a Processing Element (PE) of an SoC architecture. The proposed modeling approach provides an early estimation of performance parameters viz. memory bandwidth, average queue length at memory and average waiting time seen by a processing element; while we provide parameters viz. number of processing elements, the mean computation time of processing elements and the first and second moments of connection time between processing elements and memories, as input to the model.**

I. Introduction

Modern-day SoC platforms use a large number of embedded processors and application specific hardware components [1]. An integration of these heterogeneous components into a single chip makes communication among them critical. Besides, these components are pre-verified and optimized. Hence, communication architecture emerges as a key performance determining component of these multiprocessor SoC (MP-SoC) platforms. Furthermore, availability of several commercial communication architectures such as, AMBA, CoreConnect and their customization facilitate the designer with variety of design alternatives. Therefore, system level performance estimation is essential for selection of optimum communication architecture from a wide design space at an early stage of design cycle.

System-on-Chip applications use different types of communication architectures viz. bus-based, Network-on-Chip (NoC) based, hybrid bus-NoC architecture and cross-bar architecture. Bus based architectures are further classified as dedicated buses, single shared bus and network of shared buses. In SoCs and embedded applications, bus based architectures are popular because these are simple, consume less power and area. Moreover, performance of bus based architectures not only suffices for low end and high volume applications but also results in cheaper design. This has been motivation for our efforts for estimating performance of bus based communication architectures at the system level.

In this paper, we propose system level performance estimation of bus based communication architectures based on Stochastic Automata Network (SAN). Mainly, we focus on formulation of SAN model for a Single Shared Bus (SSB) architecture and its enhancement for Hierarchical Bus Bridge (HBB) architecture. The approach has been proposed as an extension of GSMP based performance model of these architectures [2]. In Section II, we present basic concepts and terminology of SAN, related work and our contribution. In Section III, we propose the SAN framework of SSB architecture for performance estimation. Section IV contains enhancement of the SAN formulation for HHB architecture. We present the results in Section V. We conclude in Section VI.

II. Background

A. Stochastic Automata Network: an overview

A stochastic automata network consist of a number of modules or stochastic automata. A module is modeled by a set of states and; a set of transitions which determines dynamic behavior of a component of the parallel system. The state of one module is called *local state*, while *global* or *system state* is the collection of local states of all modules. In short, the SAN model is modular representation of parallel system. The modules of a SAN model interact with each other using *local* and *synchronizing events*. Local event changes the state of a single component module by triggering local transition. Synchronizing event modifies the states of more than one modules by simultaneous transitions in those modules. Probabilities of local and synchronizing transition can be *functional* or *non-functional*. In functional transition, transition probability is the function of the states of other modules whereas it is constant in non-functional transition.

For formal description, let us consider a SAN model with N component modules and a set of events E. The i^{th} automaton, $A^{(i)}$ (where $i = 1, 2, ..., N$) with a set of states $S^{(i)} = \{a^{(i)}, ..., z^{(i)}\}$ having cardinality n_i. Local state variable of $A^{(i)}$ is denoted by $x^{(i)}$. Hence, global state of the SAN is the collection of all local states i.e. a vector $\tilde{x} = (x^{(1)}, x^{(2)}, ..., x^{(N)})$ whereas $S = S^{(1)}$ x $S^{(2)}$ x ... x

1-4244-2493-1/08/$20.00 ©2008 IEEE

$S^{(N)}$ is called the global state space . The details of SAN can be found in [3] and references there in.

B. Related Work

Work reported in [4], uses static performance estimation technique for allocation of communication channels. Our previous work [2], proposes an analytical performance evaluation of SSB and HBB architectures based on GSMP model. Analytical approach as in [5], estimates communication overhead in the pipelined communication path, which considers an impact of various protocol parameters on data transfer. Work in [6] proposes simulation based approach based on Operation State Machine for performance estimation of the system. Authors in [7] have proposed two phase hybrid performance estimation approach which first performs initial co-simulation with abstract communication and then analyses time inaccurate communication graph by specifying communication architecture. A large body of work dealing SAN formalization has been published by B. Plateau and her colleagues [3] [8]. Authors in [9] use SAN model for performance analysis in platform based design.

C. Contribution of the paper

Main contribution of the paper lies in the proposal for system level performance estimation of SSB architecture and HBB architecture. The formulation is based on the SAN model of communication architectures. We present high level simulation model of these architectures in the Stateflow component of MATLAB.

Proposed modeling approach provides an early estimation of memory BandWidth (BW), average queue length ($\overline{L}$) and average waiting time ($\overline{W}$) for SSB architecture; whereas in case of HBB architecture, we estimate local bandwidth (BW_ℓ), local average queue length ($\overline{L}_\ell$), local average waiting time ($\overline{W}_\ell$), global memory bandwidth (BW_g), global average queue length ($\overline{L}_g$) and global average waiting time ($\overline{W}_g$). The input parameters to the model are number of Processing Elements (PEs) (N), the mean computation time ($\overline{T}$) and first and second moment of connection time of PEs ($\overline{C}$, $\overline{C^2}$). Additional input parameters for HBB architecture are: probability of local or global request (X_ℓ or X_g), first and second moment of local and global connection times ($\overline{C}_\ell$, $\overline{C_\ell}^2$, $\overline{C}_g$, $\overline{C_g}^2$).

III. SAN BASED MODEL FOR SSB ARCHITECTURE

In this section, we propose the SAN model of a heterogeneous SSB architecture for evaluating performance metrics. The model has been proposed as an extension of GSMP based performance model of homogeneous SSB architecture [2]. Two types of abstract communication models are being used in SoC platforms- (i) massage passing communication model and (ii) shared memory communication model. Our formulation is based on the later model, in which SoC function involves communication of the PEs with the memories. Figure 1 shows synchronous SSB architecture which consists of N heterogeneous processing elements, PE_1, PE_2,...,PE_N competing for the use of a bus. We assume that a bus arbitration is based on the fixed priorities of PEs. The lowest priority is assigned to PE_1 while the highest to PE_N. The bus access is assumed to be non-preemptive. Arbiter of N-user one-server type resolves the bus access conflict.

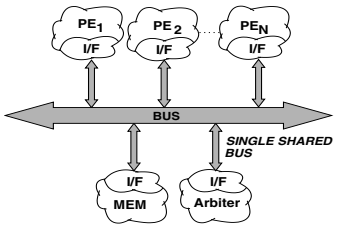

Fig. 1. A single shared bus communication architecture.

A. Model formulation

Stochastic automata network of a heterogeneous SSB architecture is modeled as a collection of interacting modules of PEs. We employ GSMP model [2] for each module which represents dynamic behavior of a PE. We use functional and synchronizing transitions to describe an interactions among these modules. Figure 2 depicts SAN model of a SSB architecture, whereas Fig. 3 shows details of one automaton $A^{(i)}$ that represents GSMP model of PE_i. *Computing state* labelled as ID^i, corresponds to the situation when the PE_i is computing. In *Accessing state* AC^i, the PE_i accesses MEM. In *full waiting state* labelled as FW^i, the PE_i waits for MEM for full connection time of another PE which is accessing MEM; while in *residual waiting state* labelled as RW^i, the PE_i waits for MEM for residual connection time of a accessing PE. In each state, model spends random amount of time with mean value η_k, called mean sojourn time of k^{th} state ($k = ID^i, AC^i, FW^i, RW^i$).

We express state transition probabilities of the SAN model in terms of transition probabilities of GSMP model of a homogeneous SSB architecture [2]. These are explained as follows. (i) α^*_{0i}- a local transition involves only $A^{(i)}$, with constant probability α_{0i}. (ii) α^*_{1i}- the functional transition which depends on the global state of the system. This transition takes place if all high priorities PEs are in computing states. (iii) α^*_{2i}- a synchronizing transition which synchronizes with event e_j (any α_1 transitions of higher priority PEs) with probability p_e and alternate probability 1. (iv) α^*_{3i}- a functional transition which takes place if any one of the PEs is in accessing state.

$$\alpha^*_{0i} = \alpha_{0i} = 1$$

$$\alpha^*_{1i} = f(x^j) = \begin{cases} 1 & \text{if, } \; x^j = ID^j,\ j = i+1, ..., N \\ 0 & \text{otherwise} \end{cases}$$

$$\alpha^*_{2i} = (e_j, p_e, 1),\ j = i+1, ..., N$$

$$\alpha^*_{3i} = f(x^j) = \begin{cases} 1 & \text{if, } \; x^j = AC^j,\ j = 1, 2, ..., N \\ 0 & \text{otherwise} \end{cases}$$

$$\overline{\alpha^*_{1i}} = 1 - \alpha^*_{1i}$$

Performance parameters of the PEs in a SSB architecture viz. $BW = P_{AC}$ and $\overline{L} = (P_{FW} + P_{RW})$ are computed

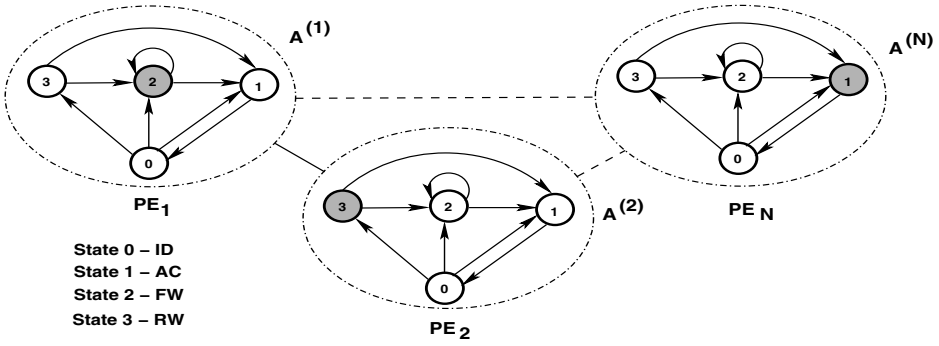

Fig. 2. The SAN model for a heterogeneous SSB communication architecture.

from steady state probabilities [2] (where P_k is steady state probability of the k^{th} state).

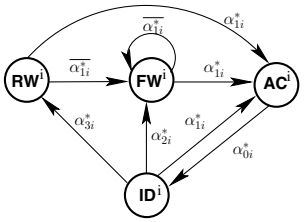

Fig. 3. An automaton $A^{(i)}$ representing GSMP model of PE_i

IV. SAN BASED MODEL FOR HBB ARCHITECTURE

In this section, we extend SAN modeling approach for HBB architecture. HBB architecture is composed of two shared buses BUS_1 and BUS_2, and connected by a bus bridge as shown in Fig. 4. Here, N number of PEs on each bus, compete to access shared memories MEM_1 or MEM_2. At the bridge level communications on two buses are concurrent whereas at bus level behavior of PEs are concurrent. For simplicity, let us consider a scenario when a PE mapped to BUS_1 generates a request to access either MEM_1 or MEM_2. With reference to this PE, parameters of MEM_1 and MEM_2 are referred to as local and global parameters, respectively. In HBB architecture, each PE can generate two requests. Let X_ℓ be the probability of local request, implying only BUS_1 would be used to access MEM_1 and arbitration of BUS_1 is sufficient. Whereas, let X_g be the probability of global request where both BUS_1 and BUS_2 would be used to access MEM_2 and two stage arbitration of BUS_1 and BUS_2 is essential. Similar explanation can also be given for a PE mapped to BUS_2.

A. Model formulation

We propose two level SAN model for HBB architecture. At bridge level the SAN consist of two automata correspond to BUS_1 and BUS_2 and are similar to the Fig. 2. At bus level, each module is composed of automata of PEs. At bridge level two automata of buses interact with each other while at bus level interaction among automata of PEs is modeled.

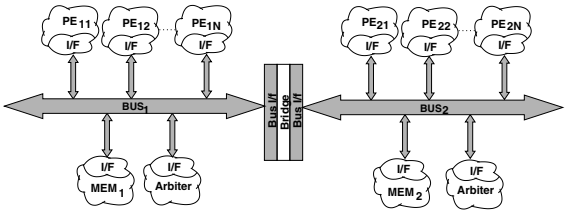

Fig. 4. Hierarchical bus bridge communication architecture.

Automata of the PE_{1i} in aforementioned scenario (mapped to BUS_1) is depicted in Fig. 5. *State* lAC_i, *State* lFW_i and *State* lRW_i correspond to local memory MEM_1 and are similar to the states of automata of a PE_i of SSB architecture (Fig. 3). Global accessing state labelled as *State* gAC_i, global full waiting state labelled as *State* gFW_i and global residual waiting state labelled as *State* gRW_i are analogous states when a PE attempts to access MEM_2. Detail discussion of model equations and performance parameters is omitted.

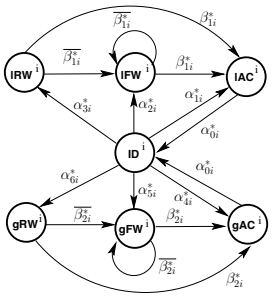

Fig. 5. An automaton $A^{(i)}$ of PE_i in HBB architecture

V. RESULTS

In this section, we present performance evaluation results of SSB and HBB architectures obtained using the proposed modeling approach. We have captured the SAN model of both architectures with fixed arbitration scheme in Stateflow component of MATLAB. Simulation was performed on on P-IV, 1 GB Linux-workstation. In both examples, random

computation and communication times of PEs were generated by using MATLAB m-functions with generalized distribution.

As first example, we have considered a SSB architecture with three PEs- PE_1, PE_2 and PE_3. We assigned the lowest priority to PE_1 and the highest to PE_3. We assigned mean values of computation times of PEs as: $\overline{T}_1 = \overline{T}_2 = \overline{T}_3$= 2 cycles. We varied mean communication time ($\overline{C}_1$) of PE_1 with $\overline{C}_2$ and $\overline{C}_3$ as parameters. Various performance parameters of the PEs viz. BW, $\overline{L}$ and $\overline{W}$ have been estimated. For brevity, we present results of BW_1 and $\overline{L}_1$ of PE_1, as shown in Fig. 6(a) and 6(b).

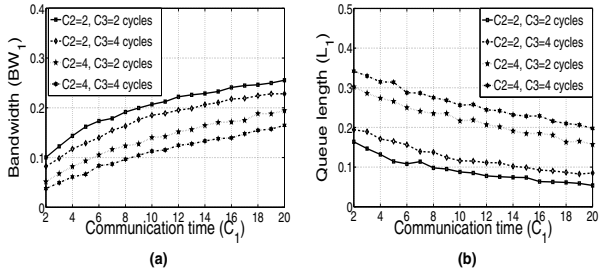

Fig. 6. Variation of (a) BW_1 and (b) $\overline{L}_1$, with C_1

As observed from the Fig. 6(a), bandwidth increases with communication time which is due to increase in mean sojourn time of AC^1 state. The figure also shows influence of $\overline{C}_2$ and/or $\overline{C}_3$ on BW_1. Reduction in bandwidth is observed when we changed $\overline{C}_2$ and/or $\overline{C}_3$ from two cycles to four cycles, since PE_1 has to wait more time for MEM in waiting states. PE_1 received maximum bandwidth (25 %) when $\overline{C}_2$=$\overline{C}_3$=2 cycles, $\overline{C}_1$=20 cycles and minimum bandwidth (3 %) when $\overline{C}_2$ = $\overline{C}_3$ = 4 cycles, $\overline{C}_1$=2 cycles. Figure 6(b) reveals converse observations for queue length, $\overline{L}_1$. For higher values of $\overline{C}_2$ and/or $\overline{C}_3$, PE_2 and/or PE_3 access MEM for more time than PE_1. As a consequence PE_1 spends more time in waiting states. Hence, higher value of $\overline{L}_1$ is noted for $\overline{C}_2$ = $\overline{C}_3$ = 4 cycles.

In second example, we have considered a HBB architecture with two PEs mapped to each bus. Processing elements, PE_{11} and PE_{12} are mapped to BUS_1 while PE_{21} and PE_{22} are mapped to BUS_2. We assigned descending priorities from global requests of PE_{22}, PE_{21}, PE_{12} and PE_{11} and then local requests in the same order. Various model input parameters are assigned values as follows- $X_{\ell 11}$=0.7; $X_{\ell 12}$=0.8; $X_{\ell 21}$=0.7; $\overline{T}_{11}$=$\overline{T}_{12}$=$\overline{T}_{21}$=$\overline{T}_{22}$= 2 cycles; $\overline{C}_{\ell 11}$=$\overline{C}_{\ell 12}$=$\overline{C}_{\ell 21}$= 2 cycles and $\overline{C}_{g11}$=$\overline{C}_{g12}$=$\overline{C}_{g21}$= 2 cycles (ℓ and g denote local and global parameters followed by PE number). From various performance parameters of PEs, we present local and global bandwidth ($BW_{\ell 22}$,BW_{g22}) of PE_{22}. We varied $\overline{C}_{\ell 22}$ for local bandwidth and $\overline{C}_{g22}$ for global bandwidth. Figure 7(a) and 7(b) show these parameters with probability of local request, $X_{\ell 22}$ with $\overline{C}_{\ell 22}$ and $\overline{C}_{g22}$ as parameters.

We observe that local bandwidth, $BW_{\ell 22}$ increases with increase in $X_{\ell 22}$ as well as with $\overline{C}_{\ell 22}$. At higher values of $X_{\ell 22}$, $BW_{\ell 22}$ is more sensitive to $\overline{C}_{\ell 22}$. An influence of $\overline{C}_{g11}$

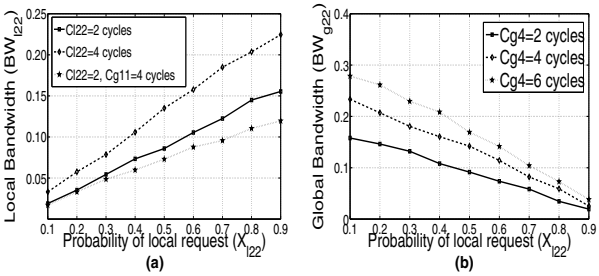

Fig. 7. Effect of $X_{\ell 22}$ on (a) $BW_{\ell 22}$ and (b) BW_{g22}

on $BW_{\ell 22}$ is clearly noted from the Fig. 7(a). Share of local bandwidth declined as we increased $\overline{C}_{g11}$ from two cycles to four cycles. In case of global bandwidth, BW_{g22} gradual decrease is observed with increase in $X_{\ell 22}$. At the same value of $X_{\ell 22}$, the PE_{22} received more bandwidth with higher $\overline{C}_{g22}$. Variations in BW_{g22} with $\overline{C}_{\ell 22}$ at higher values of $X_{\ell 22}$ are not significant.

VI. Conclusions

This paper presents SAN based modeling approach for system level performance evaluation of SSB and HBB architectures. We have evaluated performance metric viz. bandwidth, queue length and waiting time with communication times of processing elements for SSB architecture. For HB architecture performance parameters for local and global memories are evaluated with local or global requesting probabilities. Proposed approach provides an early estimation of performance metrics that can help the designer to select the appropriate communication architecture for SoC and embedded applications.

References

[1] International Technology Roadmap for Semiconductor (ITRS) 2007 Edition, "[online] available: http://public.itrs.net."

[2] U.Deshmukh and V. Sahula, "Interactive generalized semi Markov process model for evaluating arbitration schemes of SoC bus architectures," in *Int. Sym. on European Computer Modeling & Simulation*. IEEE Computer Society, 2008, Accepted.

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

[4] J.-M. Daveau, T. B. Ismail, and A. A. Jerraya, "Synthesis of system-level communication by an allocation-based approach," in *Proc. of the 8th Int. Sym. on System Synthesis (ISSS 1995)*, 1995, pp. 150–155.

[5] P. Knudsen and J. Madsen, "Integrating communication protocol selection with partitioning in hardware/software codesign," in *in Proc. Int. Symp. on Syst. Level Synthesis*, Dec. 1998, pp. 111–116.

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

[7] K.Lahiri, A. Raghunathan, and S. Dey, "System-level performance analysis for designing on-chip communication architecture," *IEEE Trans. on CAD of ICs & Syst.*, vol. 20, no. 6, pp. 768–783, June 2001.

[8] W. J. Stewart, K. Atif, and B. Plateau, "The nuemerical solution of stochastic automata networks," *European Journal of Operation research*, vol. 86, no. 3, pp. 503–525, 1995.

[9] A. Nandi and R. Marculescu, "System-level power/performance analysis for embedded systems design," in *DAC*, 2001, pp. 599–604.