

# NoC Performance Parameters Estimation at Design Stage

Nadezhda Matveeva, Elena Suvorova

Saint-Petersburg State University of Aerospace Instrumentation

Saint-Petersburg, Russian Federation

nadezhda.matveeva@guap.ru, suvorova@aanet.ru

**Abstract**—Nowadays different types of communication systems are used in designing of data transmission systems. Performance and operating characteristics of communication systems are crucial. System-on-chip (SoC) communication system can be built based on a bus, switch or network-on-chip (NoC). Type of communication system is selected according to user requirements for bandwidth, time delays, hardware cost of communication systems implementation and technology limitations. In this paper we consider the problem of different communication systems characteristics estimation. Formulas for average and maximum data transmission time calculation of different flows will be presented for different types of communication systems. Load estimation for each transfer point will be presented also. Proposed network calculator includes mechanisms based on the queuing systems to calculate the parameters of communication system. Attention will be paid to NoC communication system characteristics calculation.

## I. INTRODUCTION

Stringent requirements are imposed for modern embedded systems [1]. Weight, power consumption of Network-on-Chip (NoC) should be as low as possible. Performance, processing speed should be as high as possible. These requirements and limitations must be considered. The task of designing effective systems is difficult. Different methods of performance evaluation can be used as the structure of the system is designed. Systems modeling techniques or analytical methods for calculating the characteristics relate to it. In this paper we consider analytical methods.

## II. METHODOLOGY TO EVALUATE THE PERFORMANCE OF COMMUNICATION SYSTEMS

Currently, the most common group of methods is called network calculators. This group of methods is based on the use of analytical methods for the calculation of the timing characteristics of NoC [2, 3, 4]. It includes tools that allow to perform evaluation of maximum and average data transmission time. Some of these techniques allow to estimate the additional characteristics: area and power consumption. The advantage of this approach is that the preliminary values of the characteristics can be obtained in the early stages of design. NoC with unsuitable for the tasks characteristics should be excluded from further consideration. Present network calculators can be divided

into the following groups: deterministic; physical; based on queuing system; probabilistic.

### A. Deterministic network calculators

The graph theory is basis for deterministic network calculators. Usually it uses dataflow graphs. This approach [5] is only possible in cases where the designer has detailed information on the flow of data between nodes and switches in embedded system.

### B. "Physical" network calculators

"Physical" network calculators are based on usage of calculator models built for various physical processes to calculate various characteristics of a network-on-chip or its various fragments. In one research study [6] it is assumed that the traffic is self-similar. Further analogy was drawn between the processes of network traffic transmission and the processes occurring in a thermodynamic system.

Communication system represents a set of buffers between which data is transmitted by some rules. Rules are determined by logic of a switch. Under this approach, packets transmission between buffers is associated with migration of elementary particles between different levels of energy in a thermodynamic system. State function is associated with each buffer. This function shows the number of packages entering a buffer. It is associated with the fitness function, which characterizes the number of elementary particles at a certain energy level in thermodynamics.

This approach can be applied to NoCs with different topology and traffic patterns. However, when it is used, it is impossible to take into account the effects, which occur in the network due to the fact that the data packets are transmitted not instantly. These effects are particularly relevant for networks in which length of the data packets and the time of transmission by the physical channel can vary considerably. In addition, the process of constructing a network model has a significant computational complexity. It grows exponentially with the size of a network, number of applications. To reduce the computational complexity of the method developers use reduction of detail in the study

of the network. Network with 10 - 20 switches can be investigated without reducing detail.

### C. Network calculators based on queuing system

Examples of use of flow models to calculate the characteristics of NoC are presented in the book [3, 4]. This approach focuses on the definition of boundary values of temporal system characteristics such as transmission delay from the source to the receiver. This approach uses Min-plus algebra as a mathematical basis.

Characterization of the network is performed in accordance with traffic patterns. Service curves and arrival curves are used to characterize the data flow. System is divided into components, depending on degree of detail with which to perform research.

Individual components are switches generally. However, when a detailed study of the network is not required, network fragment, which consist of several switches are chosen as the component. For more detailed studies, components may be individual components of the switch (for example, blocks corresponding to different virtual channels).

Data arrival curve is constructed for each input part of a component. This curve is given by a non-decreasing function of time,  $A(t)$ . It characterizes the maximum possible amount of data that can reach to this input part of component at time  $t$ . Each component is determined by the service curve. This curve is given by a non-decreasing function  $B(t)$ . It defines the boundary value of operation data time in this component. Deconvolution  $A(t)$  and  $B(t)$  is used in order to determine the type of curve data in output component. System model development is performed iteratively. Propagation data paths are analyzed step by step, from the beginning to the end.

Arrival and service curves for the first components at propagation paths are formed at the beginning. Then curves describing output data streams for these components are formed. They are arrival curves for the next components. The procedure is being terminated when all paths are fully specified.

Delay processing based on arrival curves and bandwidth of each component is calculated for each component. Data transmission time between the source and the receiver is defined as the sum of the delays in each component, which is part of the way.

This approach can be used to determine the buffer size of a switch. Computational complexity of this approach is highly dependent on a number of terminal nodes and switches, number of data flows, package service rules at switches and classes of service supported by the network. Arrival and service curves have linear form when service classes with guaranteed processing time are used and

bandwidth for each data flow is limited. Equations become much more complicated with other service types, for example best effort.

### D. Probabilistic calculators

Most probabilistic network calculators based on the theory of queuing systems. This approach does not require detailed information about the data flows.

Traffic intensity, the amount of transmitted data objects is sufficient enough. Such information is available from the developer at an early stage of system development. These calculators are most widely. Different level of detail can be applied in the theory of queuing systems to NoC. It depends on the accuracy of the results. Switches are most often considered as serving partings. Individual service partings may correspond, for example, a separate virtual channels, which is part of the switches. Different units of data (flit, frame, package) can be used as requests. It depends on the desired granularity.

The average packet operation time in the switches, load switches and communication channels, queue lengths can be determined using the queueing theory. Average transmission packet time can be identified based on it. Currently, there are many papers [7,8,9,10,11], that describe the various methods of this class. Some of these calculators allow to perform calculations only for communication systems of the particular standard. Some calculators are not restricted to a specific standard. They are focused on the characteristics calculation for a particular class of graphs links, for example - mesh, torus, tree structures. Typically, they assumed that the data streams have a Poisson distribution.

These calculators are focused on calculation of the characteristics of individual switches. Sum of the average packet delay for all transit switches is the packet transmission time over the network. This approach leads to inaccuracy characteristics obtained when using wormhole routing. The information about the package can be simultaneously across multiple switches is ignored.

## III. PROPOSED METHODOLOGY

Proposed network calculator includes mechanisms based on the queuing system to calculate the average data transmission time, as well as additional mechanisms for calculating the maximum data transmission. This method includes mechanisms for measuring the load port of the communications system.

Proposed network calculator can be used to evaluate the performance of communication systems based on different topology (mesh, torus, hypercube, tree and topology with irregular structure). Unlike similar existing type calculators it can be used for networks with an exponential distribution of packet flows and with other distributions. Also this

calculator can be used for networks with wormhole routing and with buffering. It allows to more accurately taking into account effects associated with simultaneous location packet across multiple switches.

Open-loop stochastic network are used to perform evaluations. Communication system considered as a set of servers and queues of requests for servers. Requests are transaction between applications running on the master and slave devices. Service time per transaction depends on the length of the transaction, the bit line of data, processing time of one data word in the master and slave devices. Type of communication system is taken into account for the calculation of system characteristics.

There are several types of service disciplines, with priority and without. Type of service discipline also taken into account. Time during which the request is waiting in the queue depends on the type of service discipline.

TABLE I. PARAMETERS FOR CALCULATING

Parameter	Description	Unit
$T_c$	Clock	ns
$B$	Bit data word	Byte
$TW_m$	Operation time for one data word in master device	clock
$TW_s$	Operation time for one data word in slave device	clock
$TW$	Time during which a one data word occupies a channel communication system	clock
$S$	Length transaction	word
$T_s$	Transaction transmission time	clock
$T_t$	Time between requests from application	clock
$X_j^i$	Definition of transaction between application $i$ and $j$	-
$N$	Number of all switches	-
$\{P_j^i\}$	Set of data path between application $i$ and $j$	-
$l$	Data path from $\{P\}$	-
$\{CL\}$	Set of transit switches	-
$k$	Serial number of switch	-
$R$	Load of switch port	-
$T_f$	Data transmission time for 1 byte over a point-to-point connection	clock
$Lh_j^i$	Length header of packet between application $i$ and $j$	Byte
$Th_k$	Operation header time in $k$ switch	clock
$Lp_j^i$	Length data packet between application $i$ and $j$	Byte
$Trech_j^i$	Header packet transmission time from the source port to receiver over a point-to-point connection	clock
$Trec_j^i$	Packet data transmission time of packet between application $i$ and $j$ from input port	clock
$Ttrans_j^i$	Packet data transmission time to output port	clock
$Tavij_k$	Average transmission delay of packet between application $i$ and $j$ through transit $k$ switch	clock
$Tmaxij_k$	Maximum transmission delay of packet between application $i$ and $j$ through transit $k$ switch	clock
$Tavij_l$	Average transmission delay of packet between application $i$ and $j$ through $l$ data path	clock
$Tmaxij_l$	Maximum transmission delay of packet between application $i$ and $j$ through $l$ data path	clock
$Tav_j^i$	Average transmission delay of packet between application $i$ and $j$	clock
$Tmax_j^i$	Maximum transmission delay of packet between application $i$ and $j$	clock

If the system has slaves that cannot perform reading and writing transactions in parallel, read and write transaction time will be calculated taking into account the expectations of the previous transaction execution reading or writing, respectively.

IV. DEFINITION AND NOTATION

Introduce some notation before presenting the calculations. Parameters and its units are given in the table. A data path is a collection of devices, such as terminal nodes and switches that perform data transmission operations.

Let us consider that the transactions from different masters to different slave devices may have different timing values.

$$TW = \max(TW_m, TW_s)$$

V. PERFORMANCE EVALUATION FOR NOC

Information about the network architecture, data flows, buffering type and restrictions must be provided to evaluate the performance of the communication system NoC. Also packet transmission path between the sources and receivers application may be defined.

**Algorithm 1** System Network Calculator

**Step 0**

At the initial stage data path between communicating applications  $i$  and  $j$  is defined. Also number of data flow is defined for each switch.

**Step 1**

Each switch has a unique number. Data transmission time for each flow is computed for each transit switches. Total load of the output port is determined.

**Step 2**

Average and maximum time for each flow is calculated along all data path.

The following formulas were used to calculate the characteristics. Loading switch port depends on transaction transmission time and time between requests from application.

$$R = \frac{T_s}{T_t}$$

Header packet transmission time from the source port to receiver over a point-to-point connection depends on data transmission time for 1 byte over a point-to-point connection and length header of packet between application  $i$  and  $j$ .

$$Trech_j^i = Lh_j^i \cdot T_f$$

Packet data transmission time of packet between application  $i$  and  $j$  from input port and packet data transmission time to output port depend on length data

packet between application  $i$  and  $j$  and data transmission time for 1 byte over a point-to-point connection.

$$Trec_j^i = Ttrans_j^i = Lp_j^i \cdot Tf$$

If we use communication system without buffering, than average transmission delay of packet between application  $i$  and  $j$  through transit  $k$  switch is calculated as follows:

$$Tavij_k = Trech_j^i + Th_k + W + Ttrans_j^i$$

If we use communication system with full buffering, than average transmission delay of packet between application  $i$  and  $j$  through transit  $k$  switch is calculated as follows:

$$Tavij_k = Trec_j^i + Th_k + W + Ttrans_j^i$$

Maximum transmission delay of packet between application  $i$  and  $j$  through transit  $k$  switch is different for systems without buffering and with full buffering. Formulas are presented below:

$$Tmax_j^i = Trech_j^i + Th_k + Tsum + Ttrans_j^i$$

$$Tmax_j^i = Trec_j^i + Th_k + Tsum + Ttrans_j^i$$

$Tsum$  is total values transmission delay of packets, which are transmitted between different applications, except for applications between application  $i$  and  $j$ , through considered port.  $W$  is average waiting transaction time.

Average/maximum transmission delay of packet between application  $i$  and  $j$  through  $l$  data path is calculated as the sum of average/maximum transmission delay of packet between applications through all transit switches.

$$Tavij_l = \sum_{k \in \{Cl\}} Tavij_k$$

$$Tmaxij_l = \sum_{k \in \{Cl\}} Tmaxij_k$$

In some cases, for transmission data between application  $i$  and  $j$  several data paths can be used in some communication systems. In this instance, average/maximum transmission delay of packet between application  $i$  and  $j$  is calculated taking into account all data paths.

$$Tav_j^i = \frac{\sum_{l \in \{P_j^i\}} Tavij_l}{|\{P_j^i\}|}$$

$$Tmax_j^i = \max_{l \in \{P_j^i\}} (Tmaxij_l)$$

## VI. CREATION DATA PATH

If the architecture of the data path is not specified, then it can be formed. Data are constructed by the following rules. Paths do not contain cycles. Initially paths are the shortest. If several the shortest paths with equal length are possible to form, then it is valid.

The process of constructing paths (Algorithm 2) presented at Fig. 1.

### Algorithm 2 Build data path

#### Step 0

Represent the system in the graph form. Device of system is vertex. Link between adjacent nodes is edge. All vertexes have the "unmarked" status.

#### Step 1

Lists "Front1" and "Front2" are empty. Specify the source and receiver vertex.

#### Step 2

Front1 = source vertex

#### Step 3

While the receiver vertex is not reached or the list Front2 is empty perform *step 4 - step 5*

#### Step 4

Front2 = Front2 + adjacent vertex from set of vertex with "unmarked" and "marked" status.

#### Step 5

For all vertexes from Front1 status is "viewed". Clear Front1. Copy Front2 to Front1. For all vertexes from Front1 status is "marked".

#### Step 6

If receiver vertex is achievable, then data path is build.

If receiver vertex is not achievable, then data path cannot be built.

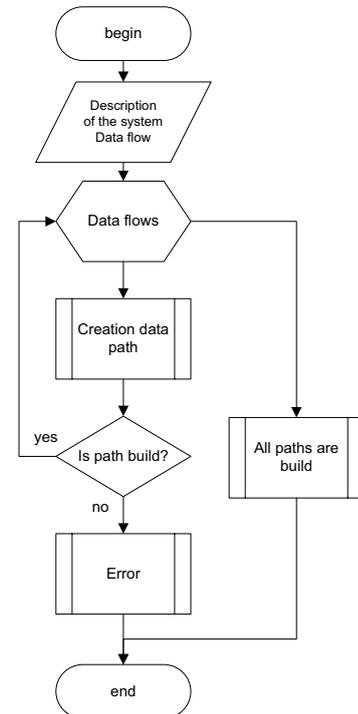


Fig. 1. Build path

The algorithm for generating the data paths is designed so that each individual path can't contain a cycle. However, the set of all paths may include cycles. This fact will cause interlocks data packets.

There are several approaches to removing cycles. One of them is to create paths without cycles. It is most commonly used for regular topologies [12, 13]. Dimension-ordered router is the simplest method of removing deadlocks in such networks [14]. Using this method, a packet is first transmitted by one of the directions (for example, horizontal) until it reaches the column, wherein the receiver node is located. Further, packets are transmitted on another (in this case, the vertical) direction. For irregular topologies, typically used other methods [15, 16]. Several of these methods are based on imposing restrictions on the presence/absence of direct links between network nodes or routing algorithms. These limits are designed to eliminate the cyclic transmission of data packets and thus remove the problem of deadlock. Another group of methods based on the use of virtual channels or groups of physical links [17].

## VII. EXAMPLE PERFORMANCE EVALUATION

In this section we present some calculation examples for different communication systems. Also simulation results of communication systems will be presented in this section. The adapted DCNSimulator [18] model was used to simulate the operation of the network. Modeling System is based on Qt and SystemC.

Simulation time characteristics are measured in time unit (s, ms, us, ns). Main calculation parameters in this paper are measured in clock unit. They are easily converted to nanoseconds through  $Tc$  multiplier. Therefore simulation transmission delay and theoretical maximum transmission delay of a packet between application source and destination are presented in nanoseconds.

### A. Example 1

NoC with regular topology will be considered as the first example. Mesh (3-ary 2-mesh) is network topology. It is presented on Fig. 2. Communication elements are denoted squares. Terminal nodes are denoted by circles. Terminal nodes contain applications that receive and transmit data. The heavy lines show data paths from sources to receivers.

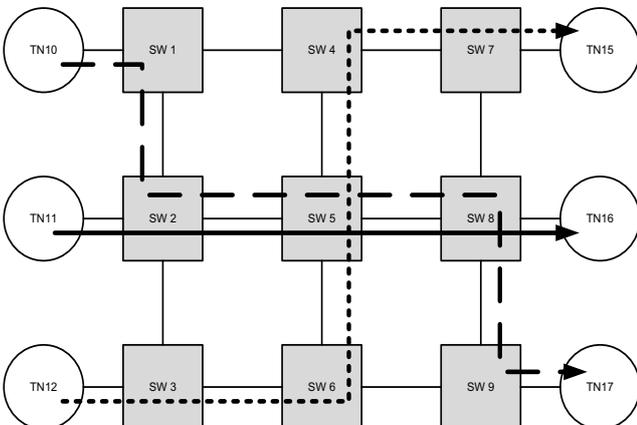


Fig. 2. Architecture of communication system 1

For this example, paths were configured initially. Characteristics for each data point will be calculated using the proposed method. Data point is a single switch or terminal node port. Calculated characteristics are load of switch port, average and maximum delay for each flow which passes through the port. Also for each data flow is calculated average and maximum transmission delay along each path. Let packet header processing time will be the same for all switches. As a single transmission path for each flow, then the following formulas are identical.

$$Tav_j^i = Tavi_j^i$$

$$Tmax_j^i = Tmaxij_j^i$$

Input system parameter values presented in the Table II. Calculations results are presented in Table III.

TABLE II. INPUT PARAMETERS

Parameter	Value	Unit
$Tc$	10	ns
$B$	1	Byte
$Tf$	1	clock
$Tf$	1	clock
$Lh_{17}^{10} = Lh_{16}^{11} = Lh_{15}^{12}$	2	Byte
$Th_k$	35	clock
$Lp_{17}^{10}, Lp_{16}^{11}, Lp_{15}^{12}$	126, 254, 510	Byte
$\{P_{17}^{10}\}, \{P_{16}^{11}\}, \{P_{15}^{12}\}$	$\{10,1,2,5,8,9,17\}, \{11,2,5,8,16\}, \{12,3,6,5,4,7,15\}$	-

TABLE III. RESULT PARAMETERS

Characteristic	Value	Unit
$Trech_j^i$ (for all flows)	2	clock
$Trec_{17}^{10}, Trec_{16}^{11}, Trec_{15}^{12}$	256, 512, 1024	Byte
$Ttrans_{17}^{10}, Ttrans_{16}^{11}, Ttrans_{15}^{12}$	126, 254, 510	clock

Theoretical maximum transmission delay for flows is 10000 ns for source TN10 and destination TN17, 9200 ns for source TN11 and destination TN16, 16300 ns for source TN12 and destination TN15. The link bandwidth in the model is set to 400 Mbit/s. Terminal nodes generate packets in a simultaneously. Fig. 3 shows the simulation results of communication system 1. Theoretical maximum delay is more than simulation delay.

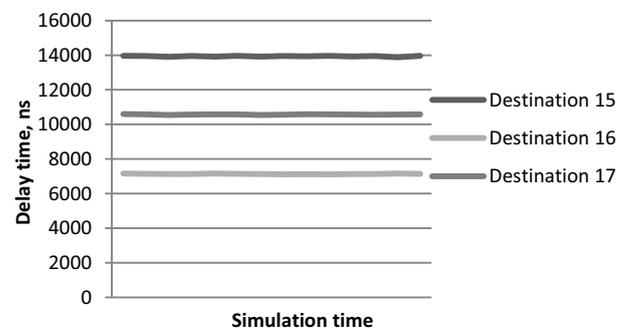


Fig. 3. Simulation transmission delay of communication system 1

**B. Example 2**

Let's take switch communication system as the second example. This system consists of RISC, memory blocks (MEM1, MEM2, MEM3), two DMASpW and ConfSpW, MPORT and ConfMPORT. Interconnection of blocks connected to the switch is shown in Fig. 4.

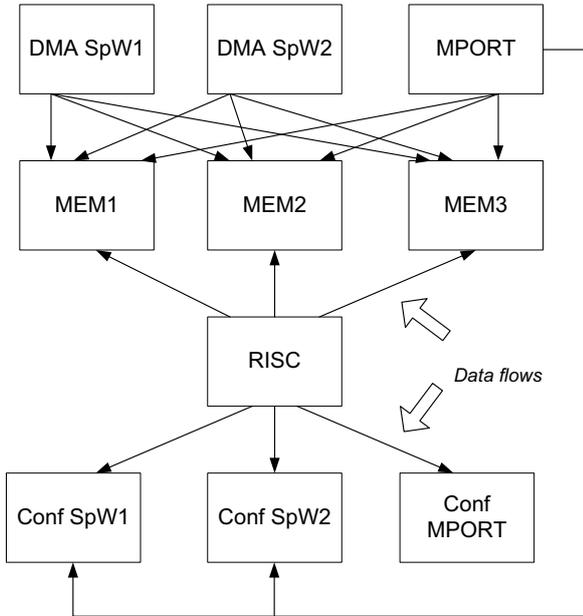
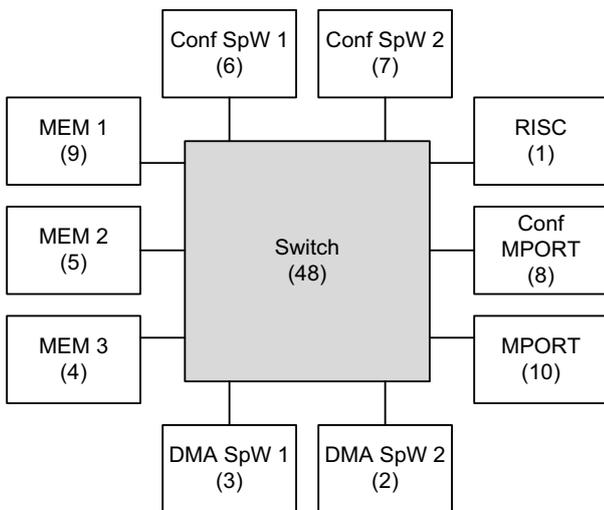


Fig. 4. Interconnection of blocks

Calculation of the communication system characteristics is carried out to determine how much the switch performance satisfies the required constraints. It will help to determine whether or not sufficient to use this type of communication system or need to move more complex communication system. Architecture of communication system for example 2 is presented on Fig. 5.



5. Architecture of communication system 2

Fig.

Input system parameter values presented in the Table IV. Calculations results are presented in Table V and in the text below.

TABLE IV. INPUT PARAMETERS

Parameter	Value	Unit
$T_c$	10	ns
$B$	1	Byte
$TW$	3	clock
$T_f$	1	clock
$S$	128	word
$L_h(\text{for all flows})$	2	Byte
$Th_k$	35	clock
$L_p(\text{for all flows})$	126	Byte

TABLE V. RESULT PARAMETERS

Characteristic	Value	Unit
$Trech_i^j(\text{for all flows})$	2	clock
$Trec_i^j(\text{for all flows})$	126	clock
$Ttrans_i^j(\text{for all flows})$	126	clock

Header and data packet length for each data flow is equal, thus  $Trech$ ,  $Trec$ ,  $Ttrans$  are the same for all flows. Maximum packet transmission delays for various applications differ. Theoretical maximum transmission delay for flows between RISC–MEM3, MPORT–MEM3, DMA SpW2– MEM3, DMA SpW1–MEM3, MPORT–MEM1, DMA SpW1–MEM1, RISC–MEM1, DMA SpW2–MEM1 = 20700 ns; between RISC–MEM2, RISC–ConfSpW2, MPORT–MEM2, DMA SpW1–MEM2, DMA SpW2–MEM2 = 24200 ns, between RISC–ConfSpW1, MPORT–ConfSpW1 = 13700 ns, between RISC–ConfMPORT, MPORT–ConfSpW2 = 10200 ns. Fig. 6–Fig. 9 show the simulation results for different sources of communication system 2. The link bandwidth in the model is set to 400 Mbit/s. Terminal nodes generate packets in a simultaneous time moments.

Analyzing these figures, we can conclude, that simulation transmission delay is no more theoretical maximum transmission delay.

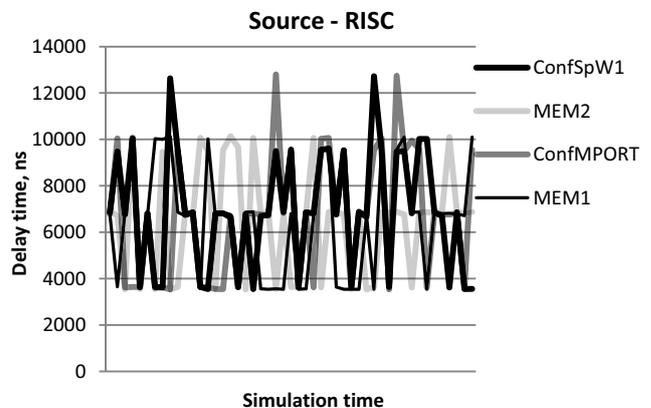


Fig. 6. Simulation transmission delay for RISC source

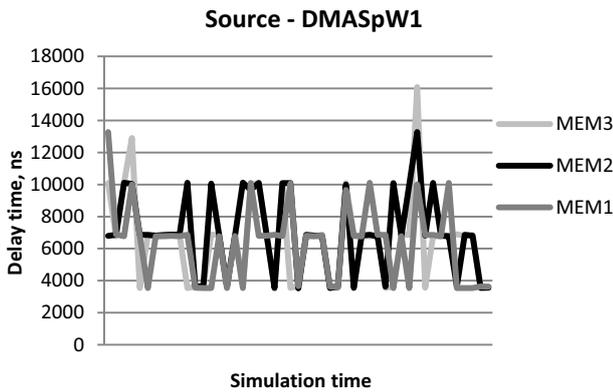


Fig. 7. Simulation transmission delay for DMASpW1 source

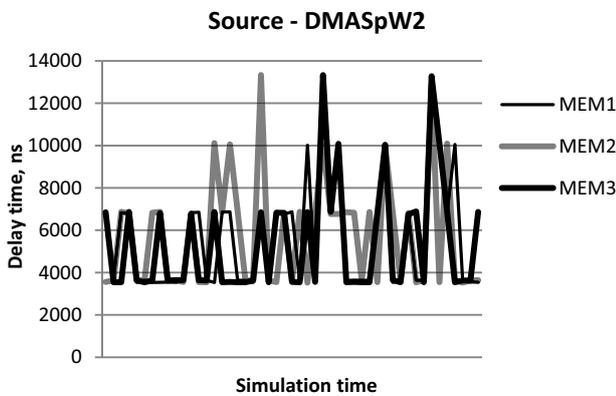


Fig. 8. Simulation transmission delay for DMASpW2 source

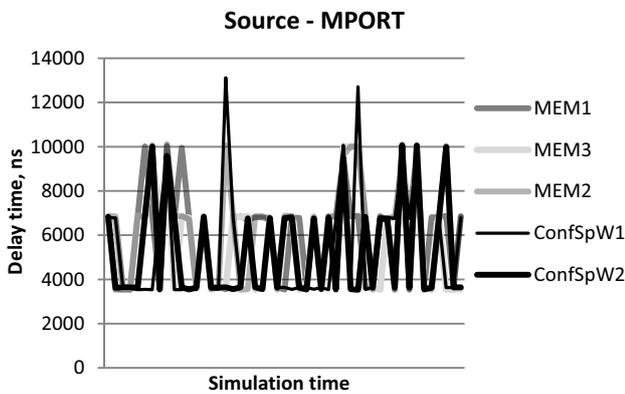


Fig. 9. Simulation transmission delay for MPORT source

VIII. CONCLUSION

In this article, we proposed an approach to evaluate the performance and operating characteristic of different communication systems. Proposed method can be used for different types of communication systems (bus, switch, NoC). Also it can be used for NoC with different network topologies. Topology can be both regular and irregular. Using the obtained values of system characteristics

designer can evaluate the system performance at the stage of architectural design. It allows identifying the bottlenecks in the system and verifies that the system corresponds requirements under which the architecture was developed. Also proposed method can be used for networks with wormhole routing and with buffering.

ACKNOWLEDGMENT

The research leading to these results has received funding from the Ministry of Education and Science of the Russian Federation under state assignment – Scientific Research Project C6.

REFERENCES

- [1] S. Balandin, M. Gillet, "Embedded Network in Mobile Devices", *International Journal of Embedded and Real-Time Communication Systems (IJERTCS)*, vol.1, No.1, 2010, pp. 22-36.
- [2] M. Bakhouya, S. Suboh, J. Gaber, T. El-Ghazawi, S. Niar, "Performance Evaluation and Design Tradeoffs of On-Chip Interconnect Architectures, Simulation Modeling Practices and Theory", *Simulation Modelling Practice and Theory*, vol. 19, № 6, June 2011, pp. 1496-1505.
- [3] M. Bakhouya, S. Suboh, J. Gaber, T. El-Ghazawi, "Analytical modeling and evaluation of on-chip interconnects using network calculus", in *Proc. of the 3rd ACM/IEEE International Symposium on Networks-on-Chip*, May 2009, pp. 74-79.
- [4] J.-Y. L. Boudec, P. Thiran, *Network calculus: A theory of deterministic queuing systems for the internet*. Online Version of the Book Springer Verlag - LNCS 2050, 2012.
- [5] A. Hansson, M. Wiggers, A. Moonen, K. Goossens, M. Bekooij, "Applying dataflow analysis to dimension buffers for guaranteed performance in networks on chip", in *Proc. of the 2nd ACM/IEEE International Symposium on Networks-on-Chip*, April 2008, pp. 211-212.
- [6] P. Bogdan, R. Marculescu, "Quantum-like effects in network-on-chip buffers behavior", in *Proc. of the 44th Design Automation Conference*, June 2007, pp. 266-267.
- [7] M. Moadeli, A. Shahrabi, W. Vanderbauwhede, M. Ould-Khaoua, "An analytical performance model for the spidergon NoC", in *Proc. of 21st International Conference AINA*, May 2007, pp. 1014-1021.
- [8] R. Marculescu, P. Bogdan, "The chip is the network: Toward a science of network-on-chip design", *Foundations and Trends in Electronic Design Automation*, vol. 2, issue 4, 2007, pp. 371-461.
- [9] G. Varatkar, R. Marculescu, "Trac analysis for on-chip networks design of multimedia applications", *DAC '02 Proceedings of the 39th annual Design Automation Conference*, 2002, pp. 795-800.
- [10] H. J. Kim, D. Park, C. Nicopoulos, V. Narayanan, C. Das, "Design and analysis of an NoC architecture from performance, reliability and energy perspective", in *Proc. of Architecture for networking and communications systems (ANCS) Symposium*, Oct. 2005, pp. 173-182.
- [11] U. Y. Ogras, J. Hu, R. Marculescu, "Key research problems in NoC design: A holistic perspective", in *Proc. of International Conference on Hardware/Software Codesign and System Synthesis, Third IEEE/ACM/IFIP International Conferenc*, Sept. 2005, pp. 69-74.
- [12] G. De Micheli, L. Benini, *Networks on Chips: Technology and Tools*. MorganKaufmann, First Edition, July, 2006.
- [13] M. Palesi, G. Longo, S. Signorino, R. Holsmark, S. Kumar, V. Catania, "Design of bandwidth aware and congestion avoiding efficient routing algorithms for Networks-on-Chip Platforms", in *Proc. of Second ACM/IEEE International Symposium*, April 2008, pp. 97-106.

- [14] J. Duato, S. Yalamanchili, L. Ni. *Interconnection Networks, an Engineering Approach*. MorganKaufman, 2003.
- [15] D. Starobinski, L. A. Zakrevski, M. Karpovsky, "Application of network calculus to general topologies using turn-prohibition", *IEEE/ACM Transactions on Networking (TON)*, vol. 11, issue 3, June 2003, pp. 411-421.
- [16] S. Murali, C. Seiculescu, L. Benini, G. De Micheli, "Synthesis of Networks on Chips for 3D Systems on Chips", in *Proc. of Design Automation Conference ASP-DAC*, Jan. 2009, pp. 242-247.
- [17] K. Srinivasan, K.S. Chatha, "A low complexity heuristic for design of custom network-on-chip architectures", in *Proc. of DATE'06*, March 2006, pp. 1-6.
- [18] A. Eganyan, E. Suvorova, Y. Sheynin, A. Khakhulin, I. Orlovsky, "DCNSimulator – Software Tool for SpaceWire Networks Simulation", in *Proc. of International SpaceWire Conference 2013*, June 2013, pp.216-221.