National Research University Higher School of Economics

*as a manuscript*

Lezhnev Evgeny Vladimirovich

# Automation of the low-level network-on-chip modeling

Dissertation summary
for the purpose of obtaining academic degree
Doctor of Philosophy in Engineering

Academic supervisor:
Candidate of Technical Sciences,
associate professor
Romanov Aleksandr Yurievich

Moscow – 2022

**The relevance of research**

The ever-increasing complexity of computing tasks, as well as the increase in the amount of data required for computing, are becoming important factors influencing the computing system development industry. This is reflected in the transition from single-core to multi-core processors, as well as from single-processor computing systems to multiprocessor ones. In the past few years, the development of processors and systems-on-chip (SoCs), as well as multiprocessor systems-on-chip (MPSoCs) has reached a new qualitative level, which is a combination on one a chip a large number of processors (from 100 or more) and their connection into a single computer network-on-chip (NoC). This required the developing new approaches to the design of computing systems.

In the classical process of designing computing systems, the following stages can be distinguished: development of technical specifications; building high-level models in high-level languages; division of the system into functional blocks, and specification of their characteristics; development of behavioral models in hardware description languages; prototyping and verification of the developed system; adaptation of low-level models to the requirements of the chip manufacturer; assessment of the physical characteristics of the implemented system; chip production; device enclosure. The emergence of new approaches to the design of computing systems leads to changes in the design process itself, in which the stage of modeling the system being developed becomes the most important.

In the simulation phase of the NoC design, there are two types of simulation that allow full exploration: high-level and low-level simulation. Thanks to them, it is possible at an early stage to evaluate the features of the system being developed. In most cases, high-level models explore the process of data transfer to the NoC in a general way and cannot take into account all the features of its hardware implementation. In the high-level modeling, as a rule, no changes are made to the description of the NoC model, since the model is hardware independent. In low-level modeling, when the entire system is represented as an RTL block diagram, the study

of the system can occur due to architectural changes. Such changes include the division of NoC into structural blocks that describe its individual components, the change in the method of exchanging information packets between the NoC components, the change in the descriptions of the components themselves, etc.

When implementing the NoC design, a separate important task is the development of its communication subsystem. In general, it is the determining of many different characteristics of such elements as: router connection topology, routing algorithm, router structure, methods for managing and arbitrating data flows in the network. The general structure of the NoC also includes the following components: computing nodes and external peripherals. But their influence on the network operation at the initial stage can be neglected because they do not affect the communication subsystem but only generate data that the communication subsystem must transmit. The NoC elements are connected to the communication subsystem only between the computing node and the network router.

To analyze the impact on the performance of the designed NoC and adoption of certain architectural decisions, modeling is required. The modeling stage is the most labor-intensive in the design – both in terms of describing modules and in the time spent on the obtaining modeling results (the time spent is much more than in case of high-level modeling). Depending on the type of data to be obtained, various types of modeling can be applied at this stage: mathematical and behavioral. Model preparation usually is the development of a large amount of the same type of program code in the hardware description languages, during which the behavior of the NoC components is described, and various model parameters are set. The division of the original model into several models of less complexity, which explore the individual components of the NoC, can significantly reduce the time for obtaining simulation results.

**The degree of research topic elaboration**

A significant contribution to the development of NoCs was made by such well-known foreign scientists as W.J. Dally, D. Debb, I.H. Wang. Among the works

of domestic researchers, one should single out the works of S.O. Bykov, N.V. Varaskin, E.A. Kichin, A.S. Kozhin, V.V. Korneev, V.O. Kostenko, A.V. Lavrov, E.A. Monakhova, Yu.A. Nedbaylo, N.Yu. Polyakov, Yu.Kh Sakhin, V.V. Tikhorsky, S.R. Tumkovskiy A.Yu. Romanov, D.I. Shpagilev, and others.

Already now, NoCs have become widespread and are used in many projects (for example, projects from such companies as Intel using the core ring connections in the processors of the Ivy Bridge family and in the 6th generation Xeon processors; MCST uses the NoCs to connect computing cores and organize data transfer between them; Tilera introduces the NoC technologies into its designs of large multi-core chips). It is also worth highlighting the developments of Arteris, which creates tools for designing and implementing the NoCs for device development, which has found application in the processor chips of Qualcomm and Samsung. One of the latest trends in the processor development is the creation of large-scale chips, for example, the Cerebras Wafer Scale Engine, which contains more than 850,000 computing cores, connected using a mesh-based communication subsystem, where the use of NoCs is the only solution that ensures the necessary speed transferring data between the computing cores. Also, it is worth mentioning the Esperanto Technologies project, in within 1088 energy-efficient 64-bit RISC-V ET-Minion vector/tensor cores were implemented in the ET-SoC-1 chip.

There are also numerous academic research groups, and the number of studies and publications on STNC is increasing every year. At the same time, it should be noted that due to the novelty of the direction, most of the studies are diverse and unrelated. This problem is raised in the works of Ahmed Ben Achballah, in which a review of low-level models is carried out according to the criteria for the synthesis of a hardware description, according to the studied components of StnC and the statement is made that the results of modeling using different models are not consistent with each other in data presentation formats. There is a lack of specialized CAD systems for the development of STNC and low-level modeling in particular. This problem is demonstrated in the works of Wim Meeus, who analyze the StnK

modeling tools according to various criteria: the language of the model implementation, the possibility of adding third-party components, the ability to generate an RTL description, the complexity of working with the modeling tool. The author describes a number of shortcomings of known modeling tools that complicate their use. For example, there is often no standardization when entering the description of the system under study, so that additional training is required for developers to develop using a specific STNC modeling tool. Often, to study the designed STNC and optimize its architecture, a deep modification of the source code is required. Some tools are designed for a single application (for example, only for describing data flows or control logic). This is also a significant disadvantage, since developers need to use several modeling tools and manually adapt the data received from them to fit their needs. The problems described above determine the object and subject of this study.

The **object** of research are NoCs.

The **subject** of research is the automation of the low-level network modeling on a chip.

**The purpose and research problems**

The purpose of research is to reduce the labor costs for preparing a low-level NoC model, reduce modeling time, and increase the size of the studied NoC by automating the processes of model synthesis and analysis of the results obtained via developing CAD for low-level NoC modeling.

In order to achieve this purpose, the following tasks are carried out:

− analysis of the principles of organization and operation of low-level NoC models, their typical structures, and tasks they solve;

− development of a computer modeling method, which is creating specialized low-level NoC models ensuring the performance of separate modeling and estimation of the resources occupied by both the communication subsystem (separately from the rest of the NoC components) and the network as a whole;

− implementation of the universal, i.e. requiring minimal configuration, an interface for connecting StnK components to its communication subsystem to reduce labor costs for modifying the connected components, as well as to coordinate the data format between the connected component and the StnK communication subsystem;

− development of a NoC model reduction method for studying networks with a large number of nodes;

− development of a specialized code translator for NoC routing algorithms from C# to HDL;

− development of a methodology for automated end-to-end NoC

− design based on a low-level model, in order to harmonize the results of NoC modeling at all stages of the design and automation of the modeling;

− development of a methodology for automated end-to-end design of STNK, which consists in automatic parameterization of a low-level model, in order to coordinate the results of modeling STNK among themselves at all stages of design and automation of the modeling process;

− development of a new CAD architecture, which differs from existing ones in combining synthesis and analysis processes in relation to the NoC design area, due to the possibility of the automated parametric modification of the model core and further analysis of the results obtained in comparison with the description of the model in Java;

− development of the CAD algorithmic support for the synthesis of complex low-level models of the NoC communication subsystems for various design tasks, as well as the automation of generation of such models;

− development of the application software that implements CAD, including information, mathematical, linguistic, methodological, technical, and software support for the development of low-level NoC models;

− approbation of the proposed methodology exemplified by solving problems of assessing the influence of the topology, routing algorithm, and other parameters

on the NoC performance as a whole to conduct its comparative analysis with the classical low-level modeling cycle.

**Research methodology and methods**

The methods of HDL modeling and simulation, methods of structural and functional programming; methods of statistical processing, analysis and interpolation of data were used.

**Author's personal contribution**

All results and provisions submitted for of the dissertation were obtained by the author personally. Based on the review and analysis of the subject area, the study of literature sources and the features of creating of the modern low-level NoC models, the author identified the problem of parametric generation of the low-level NoC models for multiple modeling in terms of the lack of automation systems for creating the same type of the low-level models.

The author personally formulated the object of research as a process in which the problem was identified, and the subject of research was disclosed as the development of tools for solving the scientific problem of automating the creation of low-level models of the NoC communication subsystem. The purpose of the work was set and the logically connected set of tasks for its achievement was defined, in the solution of which the author personally obtained the new scientific results being of great applied and practical importance for the NoC design industry.

The author's personal contribution is also reflected in a sufficient number of publications in peer-reviewed and indexed journals, in some of which the applicant is the main author.

**The main results of research**

1. A new approach to low-level modeling of the NoC communication subsystem which consists in separate modeling of the communication subsystem and computing IP-blocks, as well as in the reduction of computing IP-blocks in the

complex modeling of NoC which made it possible to study the NoCs with up to 200 nodes using any topologies and routing algorithms.

2. A new methodology for the synthesis of low-level NoC models based on the prototype of the core of the low-level model of the NoC communication subsystem which made it possible to automatically synthesize the parametric NoC models, due to which the process of debugging the NoC communication subsystem performance accelerated up to 15 times.

3. The developed method of reducing the NoC model made it possible to simulate the NoC communication subsystem without the need to use the computational IP cores to generate data packets.

4. The developed CAD made it possible to automate the parametric synthesis of the low-level models of the NoC communication subsystem, facilitated the connection of NoC components to the communication subsystem, provided a unified use of specialized algorithms (both in high-level and low-level models), processing and storage of the modeling data, and also ensured the continuity of the NoC design;

5. New approaches and methods for automating the end-to-end NoC design (from parametric description to prototype on FPGA) made it possible to speed up the process of developing the NoC communication subsystem up to 15 times and ensure the consistency of modeling results at all stages of the design.

The **reliability** and **validity** of the results obtained is confirmed by the correct formulation of the problem and the applied research methods, the consistency of the experimental results using the modern and widely tested design tools and mathematical modeling.

The reliability is also confirmed by approbation of the main results of the work over a number of years at many all-Russian and international conferences and publications in the peer-reviewed journals indexed in the international and domestic citation databases, such as WoS, Scopus, and RSCI.

**Scientific novelty** of research is in follows:

– a new approach to low-level modeling of the NoC communication subsystem was developed, which differs from the known ones in that separate modeling of NoC components is performed, as well as reduction of computing IP-blocks in the complex modeling of NoC which allows estimating the hardware costs for the implementation of NoCs based on any topologies and routing algorithms with up to 200 nodes;

– a new methodology for the synthesis of low-level NoC models was proposed, which differs from the known ones in that it uses automated parametric synthesis of NoC models for specified requirements based on the use of a prototype kernel of a low-level model of the NoC communication subsystem which made it possible to automate the parametric synthesis of NoC models for the specified requirements based on the use of a prototype core of a low-level model of the NoC communication subsystem, as well as speed up the debugging process of the NoC communication subsystem up to 15 times;

– a method for reducing the low-level NoC model was developed, which made it possible, by replacing computing IP cores with a data packet generation module, to model the NoC communication subsystem without taking into account the computing IP cores and increase the number of nodes in the network by 2.5 times, up to 200 nodes.

**Theoretical significance and practical usefulness**

The theoretical significance of this study lies in the development of the theory of automation systems for the design of the NoC communication subsystem and end-to-end NoC design methods.

The practical significance of the results obtained lies in the fact that:

− based on the review and analysis of literary sources, a review of NoC modeling methods, types of models, their structure and characteristics studied, an analysis of the subject area was carried out, which made it possible to formulate the scientific task of the research;

− a low-level model of the NoC communication subsystem was developed, which made it possible to simulate data transmission in the NoC with up to 200 nodes;

− a universal communication interface for connecting NoC components was developed, which made it possible to automate the process of data coordination between the communication subsystem and new elements added to the NoC;

− taking into account the peculiarities of the structure of the description of the components of the NoC communication subsystem, a translator to translate high-level descriptions of the specialized algorithms into a low-level representation (in HDL) was developed, which made it possible to carry out a comparative analysis of the influence of various topologies and routing algorithms on the performance of the NoC communication subsystem;

− based on the developed methodology for prototyping the NoC communication subsystem on the FPGA, a new CAD architecture was created, which made it possible to automate the parametric synthesis of low-level NoC models, due to which it was possible to speed up the debugging of the communication subsystem up to 15 times.

**Provisions to be defended**

− a new approach to low-level modeling of the NoC communication subsystem, which made it possible to conduct the NoC research with up to 200 nodes using any topologies and routing algorithms;

− a new methodology for the synthesis of low-level NoC models based on the prototype of the core of the low-level model of the NoC communication subsystem, which made it possible to automatically synthesize the parametric NoC models, due to which the process of debugging the operation of the NoC communication subsystem accelerated up to 15 times;

− a NoC model reduction method, which allows modeling the NoC communication subsystem without the need to use computational IP cores to generate the data packets;

− the CAD developed allows automating the parametric synthesis of the low-level models of the NoC communication subsystem, facilitates the connection of the NoC components to the communication subsystem, provides a unified use of the specialized algorithms, both in high-level and low-level models, allows processing and storage of the modeling data, and also ensures the continuity of the NoC design;

− new approaches and methods for automating the end-to-end NoC design (from parametric description to the FPGA prototype) made it possible to speed up the development of the NoC communication subsystem up to 15 times and ensured consistency of the simulation results at all design stages.

**Work approbation**

The new results, proposed in the dissertation, were introduced into the educational process at MIEM HSE University, applied in the MIEM HSE University design work, in the research projects of the CFS of the HSE University, as well as in the RSF grant, which is confirmed by the relevant acts:

− the act of introduction into the educational process the Department of Computer Engineering of the Moscow Institute of Electronics and Mathematics of the HSE University. The implementation of the results of the dissertation work made it possible to use modern theoretical and practical developments in the field of modeling networks-on-chip (NoCs) in the educational and scientific activities of students of the Department of Computer Engineering. The developed low-level models enabled students to study the NoCs of various configurations, apply theoretical knowledge in the field of graph theory in the practice of research and development of routing algorithms for various topologies, use tools for analyzing RTL circuits and device speed in CAD Quartus Prime. The communication subsystem model, developed as part of the dissertation, provides an opportunity for students to study cosimulation methods, work with TCL scripts, programmatic code generation on Verilog, and work with CAD Quartus Prime without using a graphical interface.;

− the act on the use of the results of the dissertation in the design work "Hardware and software complex for training in remote access to laboratory equipment". The results of the dissertation are used in part of laboratory work adapted to be performed on CAD laboratory equipment in a remote mode and without the use of debug boards using virtualization, which opens up new opportunities for studying NoCs.

− the research project of the CFS of the HSE University “Synthesis of circulant topologies for application in networks-on-chip”, reg. No. R&D AAAA-A18-118051690145-1, 01.02.2018–29.12.2018;

− the research project of the CFS of the HSE University “Modeling networks-on-chip with a communication subsystem based on circulant topologies”, reg. No. R&D AAAA-A19-119061490099-1, 01.02.2019–31.12.2019;

− the research project of the CFS of the HSE University “Development of a hybrid model for the design and simulation of networks-on-chip”, reg. No. R&D AAAA-A20-120070390136-2, 03.02.2020-31–31.12.2020;

− the research project of the CFS of the HSE University “Development of routing algorithms in networks-on-chip”, reg. No. R&D 121051100322-4, 01.02.2021–31.12.2021;

− the RSF grant “Self-organization in networks-on-chip: principles, models, routing algorithms, programs, production technologies”, agreement No. 22-29-00979, 29.12.2021 – 29.12.2023.

The main provisions of the dissertation and its sections were presented in the form of reports and discussed at 8 international conferences:

− Scientific and technical conference of students, graduate students and young professionals of HSE University named after E.V. Armensky, Moscow, Higher School of Economics, 2019 (title: “Razrabotka HDL modeli seti na kristalle na osnove marshrutizatora s odnoj ochered'yu”);

− International conference “Actual problems of system and software engineering (APSSE)”, Moscow, HSE University, 2019 (title: “Modification of the

BookSim simulator for modeling networks-on-chip based on two dimensional circulant topologies");

- International conference "Computer Simulation in Physics and Beyond (CSP)", Moscow, HSE University, 2018 (title: "Routing in Networks on Chip with Multiplicative Circulant Topology");
- International conference "Computer Simulation in Physics and Beyond (CSP)", Moscow, HSE University, 2019 (title: "Development of multiprocessor system-on-chip based on soft processor cores schoolMIPS");
- Moscow Workshop on Electronic and Networking Technologies (MWENT), Moscow, HSE University, 2020 (title: "Analytical Routing Algorithm for Networks-on-Chip with the Three-dimensional Circulant Topology");
- International scientific and technical conference "Automation" (RusAutoCon), Sochi, 2019 (title: "Routing in Networks-on-Chip with Circulant Topology with Three Generatrices of Type C(N;S1,S2,S3)");
- International scientific and technical conference "Automation" (RusAutoCon), Sochi, 2021 (title: "Development of Automation System for HDL Modeling of the Communication Subsystem for Networks-on-Chip");
- All-Russian scientific and technical conference "Problems of advanced micro- and nanoelectronic systems development (MES)", Zelenograd, IDPM RAS, 2021 (title: "Automation Low-Level Modeling Network-on-Chips").

**List of published articles reflecting the main scientific findings of research**

The results of the dissertation are reflected in 14 publications, 10 of them – in the journals included in international scientometric databases (WoS, Scopus).

*The author's works published in the peer-reviewed scientific journals included in the international citation system Web of Science and Scopus:*

1. Monakhov, O.G. Adaptive Dynamic Shortest Path Search Algorithm in Networks-on-Chip Based on Circulant Topologies / O.G. Monakhov,

E.A. Monakhova, A.Yu. Romanov, A.M. Sukhov, E.V. Lezhnev // IEEE Access. – IEEE, 2021. – Vol. 9. – P. 160836–160846. (Q1, WoS).

2. Monakhova, E.A. Shortest Path Search Algorithm in Optimal Two-Dimensional Circulant Networks: Implementation for Networks-on-Chip / E.A. Monakhova, A.Yu. Romanov, E.V. Lezhnev // IEEE Access. – IEEE, 2020. – Vol. 8. – P. 215010–215019. (Q1, WoS).

3. Romanov, A.Yu. Development of routing algorithms in networks-on-chip based on two-dimensional optimal circulant topologies / A.Yu. Romanov, E.V. Lezhnev, A.Yu. Glukhikh, A.A. Amerikanov // Heliyon. – Elsevier, 2020. – Vol. 6. – No. 1. – P. 1–8. (Q1, Scopus, WoS).

4. Romanov, A.Yu. Analysis of Approaches for Synthesis of Networks-on-chip by Using Circulant Topologies / A.Yu. Romanov, A.A. Amerikanov, E.V. Leghnev // Journal of Physics: Conference Series. – IOP Publishing, 2018. – Vol. 1050. – No. 1. – P. 1–12. (Q4, Scopus).

5. Romanov, A.Yu. Modification of the BookSim simulator for modeling networks-on-chip based on two dimensional circulant topologies / A.Yu. Romanov, E.V. Lezhnev, A.A. Amerikanov // Proceedings of the 6th International Conference Actual Problems of System and Software Engineering (APSSE). – Moscow: HSE University, 2019. – Vol. 2514. – Ch. 107. – P. 182–192. (Scopus).

6. Romanov, A.Yu. Routing in Networks-on-Chip with Circulant Topology with Three Generatrices of Type C(N;S1,S2,S3) / A.Yu. Romanov, M.V. Sidorenko, E.V. Lezhnev // International Russian Automation Conference (RusAutoCon). – IEEE, 2019. – P. 1–6. (Scopus).

7. Schegoleva, M.A. Routing in Networks on Chip with Multiplicative Circulant Topology / M.A. Schegoleva, A.Yu. Romanov, E.V. Lezhnev, A.A. Amerikanov // Journal of Physics: Conference Series. – IOP Publishing, 2019. – Vol. 1163. – No. 1. – P. 1–7. (Q4, Scopus).

8. Ryazanova, A.E. Development of multiprocessor system-on-chip based on soft processor cores schoolMIPS / A.E. Ryazanova, A.A. Amerikanov,

E.V. Lezhnev // Journal of Physics: Conference Series. – IOP Publishing, 2019. – Vol. 1163. – No. 1. – P. 1–7. (Q4, Scopus).

9. Lezhnev, E.V. Development of Automation System for HDL Modeling of the Communication Subsystem for Networks-on-Chip / E.V. Lezhnev // International Russian Automation Conference (RusAutoCon). – IEEE, 2021. – P. 780–784. (Scopus).

10. Monakhova, E.A. Analytical Routing Algorithm for Networks-on-Chip with the Three-dimensional Circulant Topology / E.A. Monakhova, O.G. Monakhov, A.Yu. Romanov, E.V. Lezhnev // Moscow Workshop on Electronic and Networking Technologies (MWENT). – Moscow: 2020. – No. 9067418. – P. 1–6. (Scopus).

*The author's works published in the other journals*

11. Prilepko, P.M. Modification of the high-level model NoCModel 2.0 for modeling networks-on-chip with circulant topologies / P.M. Prilepko, A.Yu. Romanov, E.V. Lezhnev // Problems of development of advanced micro- and nanoelectronic systems (MES-2020). – Moscow: IDPM RAS, 2020. – No. 4. – P. 23–30.

12. Lezhnev, E.V. Automation of low-level modeling of networks-on-chip / E.V. Lezhnev // Problems of advanced micro- and nanoelectronic systems development (MES-2021). – Moscow: IDPM RAS, 2021. – No. 1. – P. 46–50.

*The author's other works*

13. Zavyalov A.N. Development of an HDL model of a network- on-chip based on a router with one queue / A.N. Zavyalov, E.V. Lezhnev // Interuniversity scientific and technical conference of students, graduate students, and young specialists named after. E.V. Armensky. – Moscow, HSE University. – 2019. – P. 95–96. (RSCI).

***Copyrights and Patents***

14. Certificate 2021616623 Russian Federation. Certificate of state registration of the computer program. Verilog Code Generator of Communication

Subsystem for Networks-on-Chip (HDLNoCGen) / A.Yu. Romanov, E.V. Lezhnev; applicant and copyright holder Federal State Autonomous Educational Institution of Higher Education National Research University Higher School of Economics (RU). – No. 2021615675; dec. 04/20/21; publ. 04/23/21, Register of computer programs.

**Contents**

The dissertation consists of an introduction, five chapters, a conclusion, a list of references, and an appendix with the acts of implementation of the research results.

The **introduction** substantiates the relevance of the dissertation topic, formulates the purpose and objectives of the study, determines the scientific novelty and practical value of the results obtained, and provides the data on their approbation.

**Chapter 1** analyzes the main approaches to NoC modeling, stipulated by choosing the level of abstraction of the model. Depending on the required tasks, the model can be implemented in hardware description languages (HDL) or high-level languages (HLL). The characteristics of the main types of modeling are determined due to the level of abstraction of the model, the data processed, and the purposes of modeling. One of the consequences of choosing a method for implementing the model is the speed of modeling, as well as the accuracy of the results. Although simulators, developed in hardware description languages, are inherently more accurate models, they have the disadvantage of long simulation times. High-level models have significantly less average simulation time, but their accuracy may be worse.

The chapter describes the structure and main components of NoC models. We can distinguish the following NoC components used in the models: routers, the structure of connections between them, and computational IP cores.

The NoC structure and tasks solved by its components were compared with the OSI network model (Fig. 1). The application layer, presentation layer, and session layer are implemented by an IP computational block that generates the data

for transmission over the network and indicates the destination IP core where this data should be transferred. The remaining layers are implemented by the NoC communication subsystem, which receives the data from the IP core, converts it into a format suitable for transmission between routers in a network with a predetermined topology, transmits it along the path calculated by the router, and, upon reaching the target node, performs the reverse transformation for the transmission to the IP core.

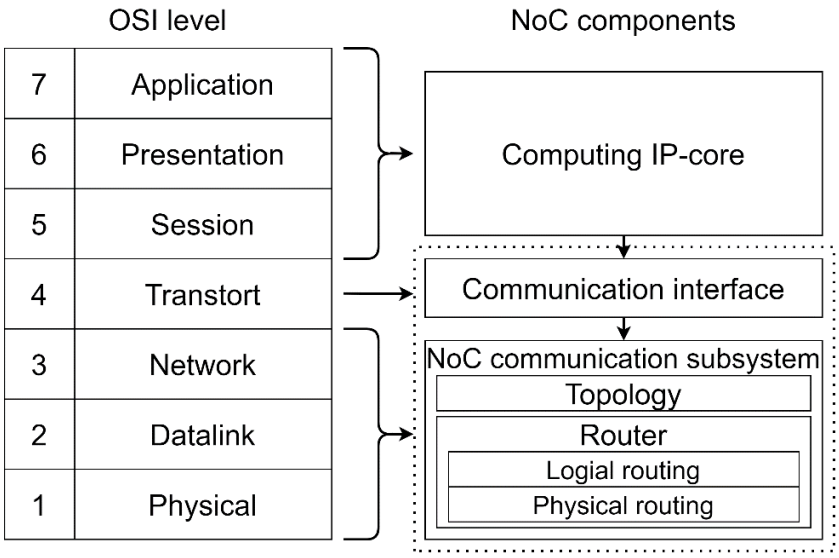



Fig. 1. OSI model in the NoC.

Necessity of low-level modeling in the NoC design is substantiated. A comparison of low-level modeling with high-level and behavioral modeling is given; it is shown that only low-level models can be synthesized into a real NoC and provide the highest modeling accuracy, since a network prototype that is closest to the real NoC implementation is researched. High-level models implement a NoC in a general form and cannot take into account all the features of its implementation.

The chapter presents a classification of low-level NoC models, their structure, the studied characteristics and network components (Fig. 2).

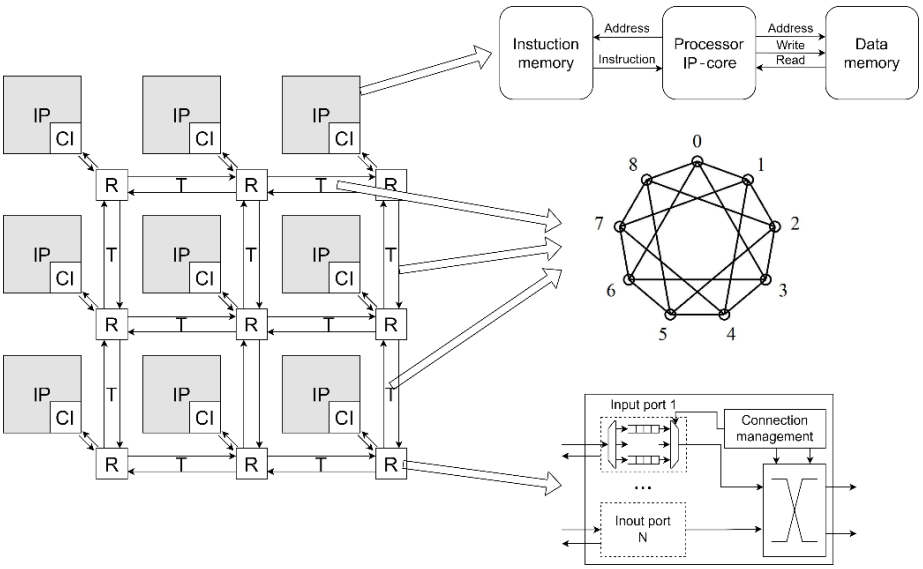



Fig. 2. Structure of the low-level NoC model.

This chapter also provides an overview of low-level NoC models, according to which most of them allow exploring the individual characteristics of the NoC and implement its individual components. As a result of the review, it is shown that the results of modeling of different models cannot be shared because they are presented in various formats. In addition, modification of models is usually not possible because in most implementations, there is no parametric tuning of the models.

**Chapter 2** shows that not all cases require simulation of the entire NoC. Often it is necessary to simulate the certain parts of it, such as a communication subsystem, routing algorithm, and traffic control system. The modeling of individual NoC components allows evaluating their impact on the network separately, as well as (based on the data obtained) making a parametric selection of optimal settings for the operation of the selected NoC component. Also, separate modeling of the NoC components makes it possible to speed up the process of modeling both the network as a whole and the components under study separately.

The chapter substantiates the importance of the topological approach in the NoC design and modeling. Structurally, NoC can be divided into 2 main parts: the computing IP core and the communication subsystem. These parts of the NoC interact with each other only when transferring the data between the IP core and the communication subsystem and do not actually affect the operation of each other.

The consequence of this is the possibility of separate modeling of these parts while maintaining the correctness of the results obtained. Depending on the chosen topology of the connection of routers, as well as the chosen routing algorithm, the characteristics of the network change. Therefore, when NoC designing, the modeling of the communication subsystem is the most important step.

**Chapter 3** presents the developed low-level model of the NoC communication subsystem (Fig. 3). The model consists of 4 components: the core of the model is generated files in the Verilog language that implement the communication subsystem of the studied NoC; model setting module, which receives the necessary data for the parametric generation of the NoC communication subsystem; data processing module, which is necessary to obtain the results of the model and generate the test data for submission to the model in real time; testing infrastructure, which contains files for conducting automated event modeling of the model operation on the pre-prepared data.

The core of the model consists of three main files: the router module, topology module, and test data packet generator auxiliary file. The router module implements the routing algorithm under study and actually represents the state machine on the basis of which the router operates in the network. Using the information about the topology of the communication subsystem stored in the router, as well as service information stored in forwarded data packets, the port is selected to which the packet must be sent so that it reaches the destination node. The topology module implements the topology of the communication subsystem under study. It connects all the routers to each other, transfers the data between them and outputs the information to the computing node. The packet generator is auxiliary and in fact is not a structural element of the communication subsystem; it is needed to reduce the NoC model (the replacement of computational IP cores). It emulates the generation of the data packets when testing the network. All ports of the routers for communication with computing nodes are connected to the packet generator module. This makes it possible to conduct the NoC research with a larger number of nodes

than with the real IP computing blocks. For testing, an add-on was developed with testbench files and for using DPI tools.

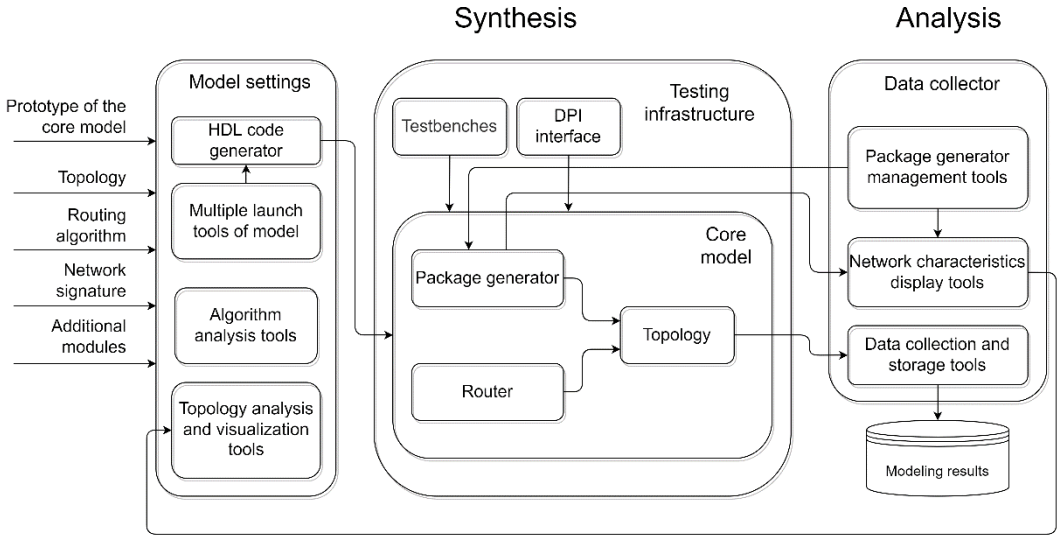



Fig. 3. Structure of the low-level model of the NoC communication subsystem.

The high-level part of CAD was developed in C#. It includes the tools for collecting data from the model core, as well as tools for managing the generation of the data packets.

An important part of CAD is the model setup module, which allows generating the core code of the model, as well as running the model multiple times with different parameters.

The remaining parts of the CAD system are the results analysis tools. Some of them (the tools for displaying network characteristics) are located in the data collector module to provide access to the raw data. The remaining parts are located in the model setting block to perform the data processing from the model and allow them to be compared with the results obtained at the high-level modeling stage.

The following set of parameters is fed to the CAD input: the network topology, routing algorithm, network signature (which contains the information about the number of nodes and the topology of their connection), additional modules that need to be connected to the network. Optionally, one can set the location of the model core prototype.

The developed model allows modeling the parameterized NoC communication subsystem to obtain an estimate of the number of occupied arithmetic logic blocks (ALM) and registers (REG, memory blocks) required for prototyping the communication subsystem.

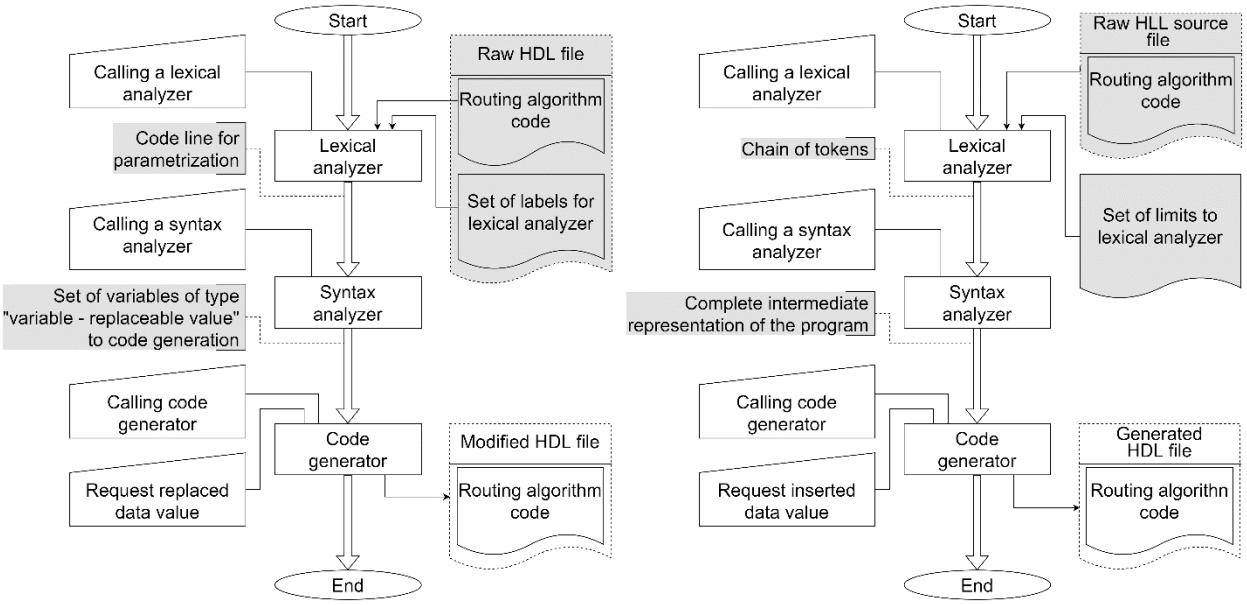
All components of the communication subsystem are implemented as separate modules, due to which it is possible to add the necessary components for the study. This is possible due to the created universal interface, to which all connected components transmit the data in their own format, after which they are converted for further use by other components. With the help of a universal interface, the ability to add the additional components to the model for the NoC modeling as a whole is implemented.

In accordance with paragraph 3.3.2 of GOST 25301.101-87 “Computer-aided design systems”, CAD should include the following set of support: mathematical, algorithmic, informational, technical, software, methodical, organizational. Algorithmic support includes the HDL code generator block, which is a code translator that generates the model core. Information support includes the input data, on the basis of which the parametric synthesis of the model takes place, as well as the results of the model performance, such as the occupied chip resources, frequency of the synthesized network project, and efficiency of the component placement. The hardware includes a workstation on the Windows operating system, which runs the developed CAD, CAD Intel Quartus Prime Lite, which creates a network description file for the FPGA chip, as well as the FPGA chip itself as part of the debug board.

**Chapter 4** discusses the ways to automate preparing the HDL model of the NoC communication subsystem. For each analyzed configuration of the communication subsystem, it is necessary to create its own model core files, which is a routine and time-consuming task. To automate this process in C#, a module was developed for setting up the model, which includes multiple launch tools, as well as an HDL code generator. Multiple launch tools allow the user to specify multiple

datasets for model synthesis and run all necessary software modules for the model synthesis and execution.

The main component of this subsystem is the HDL code generator. It is used to perform the parametric synthesis of the core of the model. Structurally, the HDL code generator is a translator that can operate in two modes, depending on the data supplied to the CAD. The operation modes of the translator are shown in Fig. 4.




a) b)

Fig. 4. Operation modes of the HDL code generator: a) based on the prototype of the model core; b) in C#-to-Verilog translator mode.

In the first mode, the translator's operation is based on the use of the model core prototype (Fig. 4a). The developer creates a prototype of the model core once. During further operation of the system, it is required to change only the file in which the routing algorithm is implemented. In this file, the developer notes the lines of the code that need to be changed to create a parameterized model. The translator receives this file as input, reads it line by line, and passes it to the lexical analyzer. The lexical analyzer in each line searches for a label to change it. The found string is passed to the parser, which parses the string, determines the parameter to be changed, and returns the name of the parameter to be set. A set of variables, consisting of a variable and a value to be replaced, is passed to the code generator,

which requests all the necessary data from the model and creates a new file with the routing algorithm.

The second mode of the generator is C#-to-Verilog translation (Fig. 4b). In this operation mode, the generator does not need a model core prototype. The input of the translator is a file with a routing algorithm developed in C#. A file with restrictions for the lexical analyzer is also supplied. Based on these inputs, the code generator generates a file with a routing algorithm in the Verilog language, as well as all the other necessary files.

The developed translator not only makes it possible to automatically synthesize a parametric model of a communication subsystem, but also provides an opportunity to coordinate the study of the routing algorithm with a high-level model.

The NoC model implements the ability to connect the external computing IP cores to test the network as a whole. To coordinate the data and bring them into the required format, a communication interface between the IP cores and the communication subsystem that converts the data is implemented. The communication interface is a separate module that the developer can configure to connect various IP cores.

Using real computing IP cores to generate data for NoC operation, it is possible to explore the networks with a small number of nodes, which was shown by calculating the resources consumed by networks based on schoolMIPS cores and NIOS II cores. Based on the DE1-SoC FPGA development board, it is possible to implement a NoC with 50 NIOS II nodes, and with 127 schoolMIPS nodes. It should be noted that the schoolMIPS core is not suitable for the real use due to its limited capabilities. Thus, when using the real computing IP cores, it is problematic to explore a NoC with more than a hundred nodes.

The solution to this problem is to reduce the model. This task is handled by the packet generator module. It creates the test data packets to simulate the operation of the communication subsystem. The developed low-level model uses a simplified packet structure (Fig. 5): the payload in the packet occupies only 1 bit, which is the

packet presence flag. The rest of the packet is auxiliary information for the operation of the routing algorithm. The size of the auxiliary data depends on the routing algorithm. To test the work of the reduced model, the pair exchange algorithm as a routing algorithm was used.

| Data packet presence flag | Routing information | |
|---|---|---|
| | Source address | Destination address |

Fig. 5. Structure of the packet generated by the packet generator model reduction module.

Using the packet generator as a replacement for the computing IP cores, it was possible, with the same limitations of the debug board, to increase the network by 37% compared to the network based on the schoolMIPS cores and by 248% compared to the network based on the NIOS II cores. The maximum network size, based on the reduced model, was 174 nodes. Using other routing algorithms, it was possible to increase the size of the NoC to 200 nodes.

The reduced model presented has several additional modules that are not architectural components of the communication subsystem and the NoC as a whole, but also participate in modeling and introduce some distortions into the results. The assessment of the impact of these modules is also discussed in Chapter 4. Formulas, on the basis of which it is possible to pre-calculate the introduced distortions in the simulation results using additional modules, are presented; they can be used to correct the simulation results.

The structure of the packet generator module is constant and depends linearly on the parameters of the NoC communication subsystems. Its basis is the dimension vector $L = N \cdot BIT$, where $N$ – number of nodes, and $BIT$ – the length of the packet entering the router. Further, this vector is divided into $N$ vectors of the length $BIT$, which are transmitted to the routers. Thus, it is possible to obtain a theoretical estimate of the redundancy introduced into the model of the communication subsystem of the memory registers (Fig. 6).

$$U_{reg} = N \cdot BIT + 16 + 2 \cdot (BIT - 1),$$

where $N$ – number of nodes in the network;

$BIT$ – packet length.

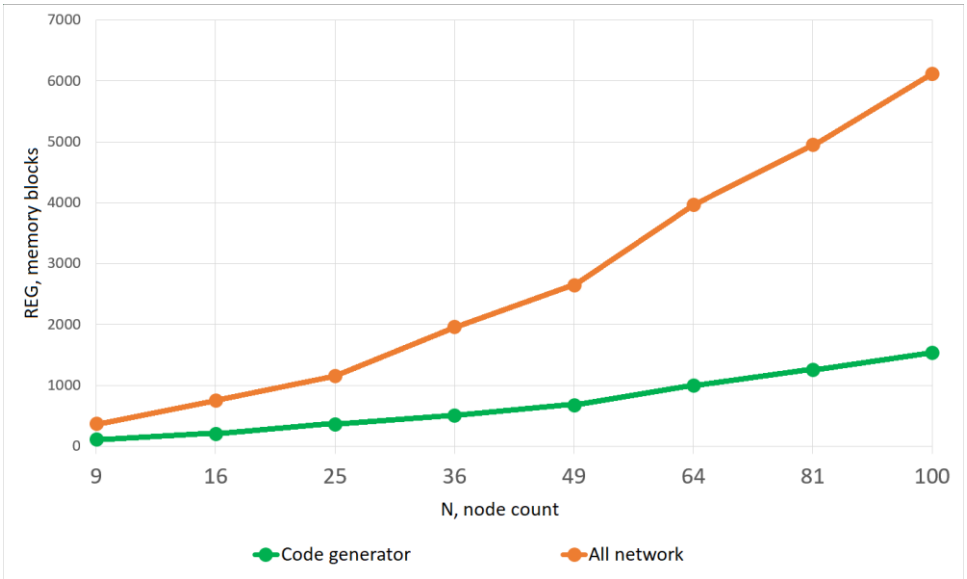



Fig. 6. Dependence of the use of FPGA registers for the auxiliary modules on the number of nodes in the network.

Redundancy assessment in the logical blocks (ALM) (Fig. 7)

$$U_{alm} = 0{,}0006 \cdot N^2 + 8{,}546 \cdot N + 25{,}101,$$

where $N$ – number of nodes in the network.

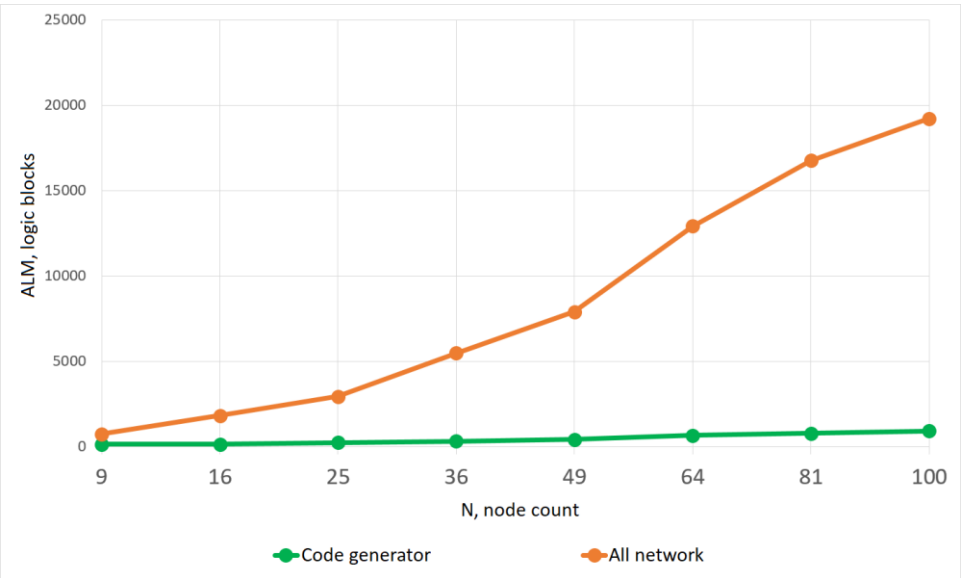



Fig. 7. Dependence of the use of the FPGA logical resources for the entire network and auxiliary modules on the number of nodes in the network $N$.

For memory registers, the resource estimate exactly matches the theoretically calculated values. The estimation of logical resources can be expressed by a

polynomial function of the second degree presented above with reliability coefficients $R^2 = 0{,}991$. It also follows from the results obtained that (compared with the communication subsystem) the packet generator module takes up very few resources, and for a larger number of nodes this redundancy can be neglected.

Based on the obtained formulas for estimating the resources used by non-architectural NoC components, a graph for calculating the resources used for a larger number of nodes using the data extrapolation method was obtained (Fig. 8).

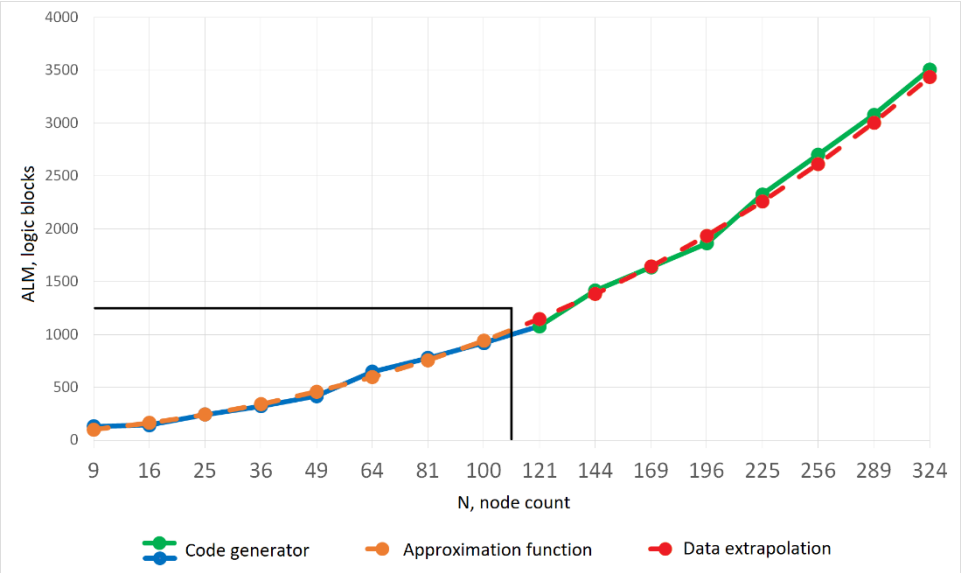



Fig. 8. Dependence of the use of FPGA logical resources for the entire network and auxiliary modules on the number of nodes in the network $N$.

The polynomial-based data for NoCs from 121 to 324 nodes were compared with the experimental data, and the extrapolated data proved to be consistent with the measured data.

All the modeling tools created made it possible to speed up the process of obtaining the characteristics of the network. Fig. 9 shows the dependence of the speed of obtaining network operation data on the number of nodes using the developed CAD system and using low-level modeling (in ModelSim). To assess the speed of obtaining results, the modeling of the passage of one data packet was carried out. Compared to the ModelSim, the modeling acceleration was several orders (up to 15 times).

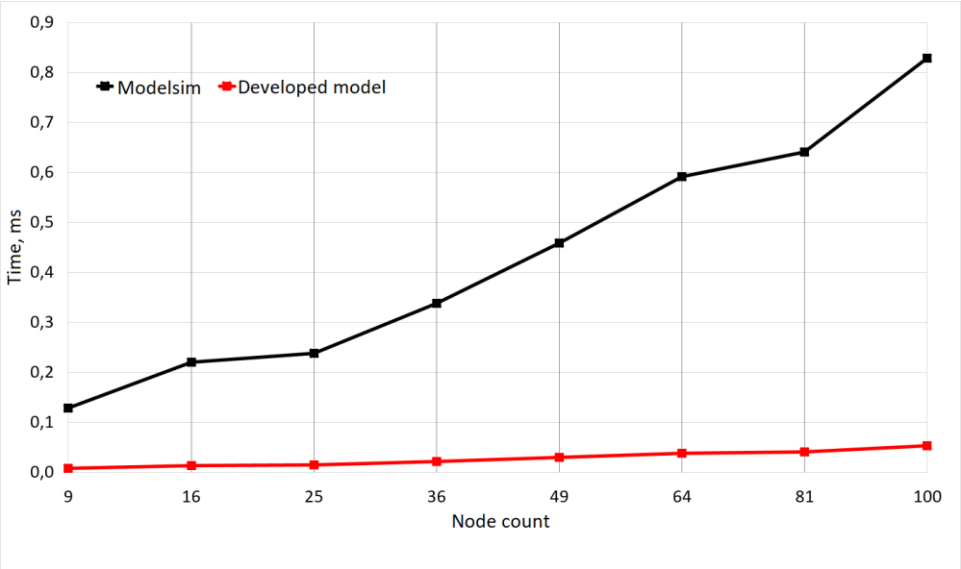



Fig. 9. Comparison of the speed of obtaining the NoC results using the developed tools and the ModelSim tools.

**Chapter 5** is devoted to testing the model developed. The developed CAD and modeling methods were applied in the studies of the operation of algorithms and the influence of the topology on the NoC performance, which were carried out as part of the research projects of the HSE University and CFS research projects. The data on the NoC occupied resources for the networks with different parameters were obtained. Based on these results, the graphs and dependencies were constructed.

The modeling was carried out using the non-architectural elements in the structure of the NoC communication subsystem (in comparison with the same network but in which schoolMIPS and NIOS II processor cores were used as computing nodes). Fig. 11 shows the structure of the schoolMIPS software processor modified for the use in the developed model. 2 modules were added to the processor structure to coordinate the data format between the processor and the communication subsystem. The first module generates a packet of a given format for its transmission to the router, and the second module is engaged in the inverse data transformation to transfer the packet to the schoolMIPS processor.

Thus, thanks to testing on various tasks, it was shown that the developed CAD can be used for practical application for NoC modeling with the number of nodes

reaching hundreds to study various aspects of its functioning: from routing algorithms to a traffic control method, or their combinations.

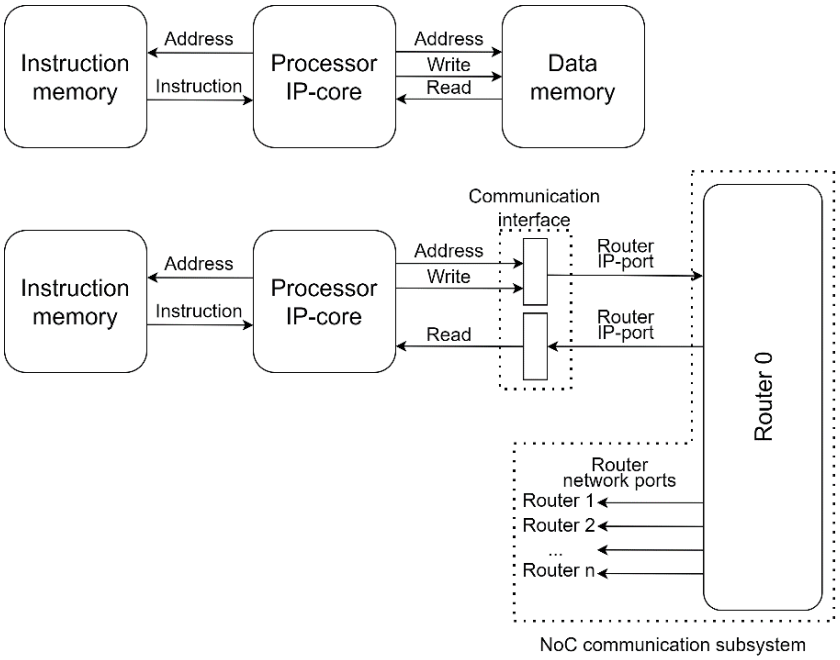



Fig. 10. Modified structure of the SchoolMIPS soft processor for the use in the communication subsystem model

In **conclusion**, the main results of the research are listed, the most important of which is the solution of an actual scientifically applied problem related to automating the development of the low-level models of NoC communication subsystems and increasing the size of the simulated NoC by creating a new low-level model of its communication subsystem.