Verification of 10 Gigabit Ethernet controllers

M.V. Petrochenkov <petroch_m@mcst.ru>
R.E. Mushtakov <mushtakov_r@mcst.ru>
I.A. Stotland <stotl_i@mcst.ru>
MCST, 1 Nagatinskaya st., Moscow, 117105, Russia

Abstract. This article proposes approaches used to verify 10 Gigabit Ethernet controllers developed by MCST. We present principles of the device operation – they provide a set of memory-mapped registers and use direct memory access, and their characteristics. We describe a set of approaches used to verify such devices – prototype based verification, system and stand-alone verification. We provide the motivation for the chosen approach – combination of system verification with stand-alone verification of its single component. The structure of the test systems that we used to verify devices and their components are presented. Test system of the controller transmits Ethernet frames to the network and receives frames from it. Algorithms to transfer packet to representation used by the device were implemented. Stand-alone test system was developed for a connector module between internal device buses and its external interface. Test systems were developed using UVM. This methodology and structure of test systems allowed to reuse components in a different systems. A set of test scenarios used to verify the device is described. The examination of network characteristics of the controller is very important in the verification process. Some approaches and techniques for throughput measuring and modes of device operations for the measurement are described. We present measured throughput in different modes. In conclusion, we provide a list of found errors and their distribution by different types of functionality they affected.

1. Introduction

Development of modern computer networks provides the demand for high-speed communication without sacrificing reliability. The evolution of Ethernet standard

(IEEE 802.3 [1]) is an example of ever-rising demand for higher speed networks. Network interface controller (NIC) is the device that connects a computer to the network. Reliability and performance of the controller is very important for organization of modern networks. Network performance and accuracy of its work as a whole depends on the quality of implementation of NICs. To ensure that the controller satisfies all requirements for performance and reliability, it should be thoroughly verified.

Various methods of verification are used at all phases of NIC design flow. Common approaches for the device verification are physical prototype verification, system verification and stand-alone verification.

Process of physical prototype verification uses the device implemented in FPGA as a NIC in a "real" machine. Characteristics of the approach:

- Test stimuli are generated using operating system network drivers and signals from physical network (in our case - third-party 10 Gigabit Ethernet controllers).
- The fastest approach by a wide margin.
- Ability to execute "real life" scenarios and gather information to improve the device performance for most important use cases.
- Ability to debug network drivers.
- Difficulty in localization of detected errors.
- Slow iteration cycle due to slow recompilation to FPGA process.

In the system, verification approach a NIC is simulated as a part of whole System on a Chip (SoC). NIC is configured according to required settings and then it executes network transactions. Characteristics of the system verification approach for the Ethernet controller:

- Test stimuli are memory access operation to the device registers and received Ethernet packets - very similar to typical mode of device operation.
- Simpler localization of detected errors and better error detection tools than using physical prototyping.
- Ability to implement directed scenarios from a physical prototype.
- Difficult and time consuming to cover all possible situations, especially for complex internal components.
- Requires all device components to be in working state.

In stand-alone verification, a single device component is simulated. Typically, it is used for components with (1) high internal complexity and (2) which reliability is crucial for the device [2]. Properties of the approach:

- Test stimuli are transactions on the external interfaces of the component.
- Could be start as soon as only RTL-model of device component is ready (not the whole device).

- Faster simulation for smaller device subset.
- It is easier to create specific test cases and more complex test scenarios for component under test.
- Information about internal interfaces of the device is required.
- Cannot eliminate the need of system verification completely and thus always requires extra labor and other resources.

In our previous projects, we used only physical prototyping and system verification to verify NICs. However, there are types of errors that are hard to find using only these approaches. In this regard, all aforementioned methods we used during verification process of 10 Gigabit Ethernet controllers. Separate team used physical prototype verification approach, and further discussion of it is beyond the scope of this article.

In this paper, we present a case study for functional verification of 10 Gigabit Ethernet controllers developed by MCST. The paper addresses the problem and methods of stand-alone verification of 10 Gigabit Ethernet controllers.

The rest of the paper is organized as follows. In the Section 2, we describe the devices under test: RTL-models of the 10 Gigabit Ethernet controllers, their features and intended methods of implementation. Section 3 presents different test systems developed for components of the controller and the complete controller. In Section 4 we give further insight into the process of examination of the device network properties - most importantly its throughput. Section 5 presents the results of verification and plans of future work.

2. Device Under Test for Different 10 Gigabit Ethernet Controller Implementation

Model of the 10 Gigabit Ethernet Controller is implemented using Verilog Hardware Description Language (HDL). It is RTL (register transfer level) description that is used in different implementation (Device Under Test, DUT):

- FPGA-based network controller (based on Altera Cyclone V [3]). This FPGA provides a set of components that were used in the device: PCI Express Hardware IP module that implements physical and data link layer of the protocol, and a set of configuration space registers, and XAUI Hardware IP module to transform 10 Gigabit Media Independent Interface (XGMII) signals. It is connected to the other parts of the system using standard interface Avalon [4].
- ASIC-based network controller - a part of a currently developed Elbrus-16C System on Chip (SOC). Controller is connected to the rest of the system using in-house interface (SLink) to transfer packets based on PCI Express transaction layer packets.

General schemes of both DUT are presented in figure 1. Both types of the DUT share the Ethernet Control Module and implement the same programming interface. This interface is typical for PCI and PCI-Express devices. A set of memory-mapped

259

registers are used to control device behavior. Those registers can be separated into four groups:

1. PCI Express registers - common set of registers of PCI Express devices. They are used to control access to internal memory of the devices and from the device to system memory. It also implements basic interrupt control.

2. Media Access Control (MAC) registers allow to control the Ethernet physical layer. They are used to control pause frames, control sum (CRC) calculation, non standard-compliant frames and limit the speed of packet transmission.

3. Transceiver (TX) registers control the transmission of packets from the system to the network, calculation of CRCs for higher-level protocols supported by the controller (IPv4 and IPv6, TCP and UDP).

4. Receiver (RX) registers control the reception of packets from the network, packet filtering and control sum checking for IP, TCP/IP and UDP/IP packets.
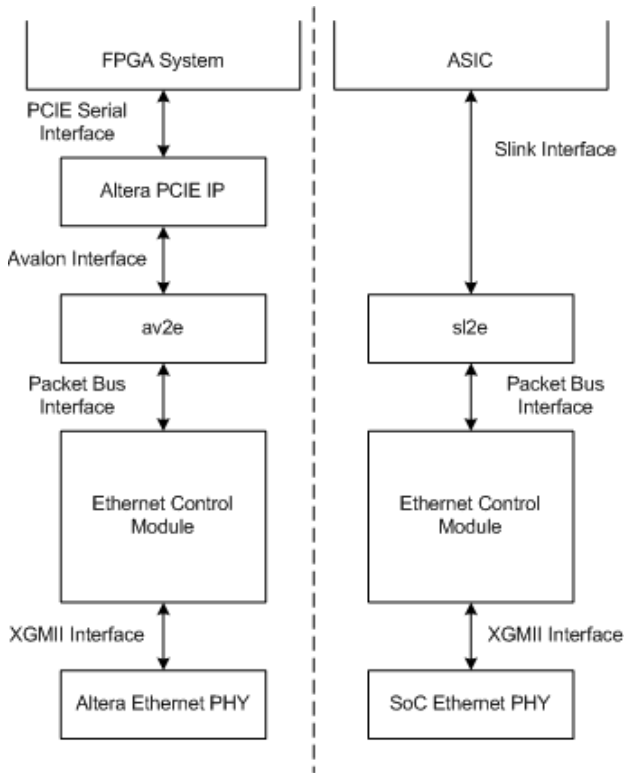

Fig. 1. Devices under test for FPGA and ASIC-based in SoC implementation.

To allow multiple processes to work in parallel with a single controller TX and RX registers contain several identical groups of registers (descriptor queues).

Ethernet Control module uses universal packet bus interface. This interface provides convenient access to large continuous areas of memory where Ethernet packet data are stored. Altera PCI Express and SLink interfaces work with packets of size up to 64 bytes. To increase the rate of data transmission and reduce the CPU involvement into device operation it used direct memory access (DMA). Devices use different modules to connect packet bus to Avalon and SLink interfaces. Internal names of those connector modules are av2e (avalon to everything) and sl2e (SLink to everything). Those connectors implement DMA by splitting packet bus transactions and transforming them into memory access operations.

3. Test Systems for 10 Gigabit Ethernet Controller Verification

As stand-alone verification could be started as soon as RTL-model of device component is ready, without waiting RTL-model of the whole device. Verification of the 10 Gigabit Ethernet Controller process was started at the same time as the development of the FPGA-based controller and system on chip (SoC). This approach allowed identifying errors earlier, and reducing total development time of the device. To check correctness of the controller model, it is included in a test system — a program that generates test stimuli, checks validity of reactions and determines verification quality.

There are several verification methodologies in order to develop constraint-random coverage-driven verification test systems. A verification methodology provides guidelines, class libraries and macros libraries. The Universal Verification Methodology (UVM) [5] is currently the most widespread verification methodology. UVM allows automating test system design process and makes it easier to add new components and collecting the functional coverage [6]. In paper [7] the approach to UVM test system developing for Gigabit Ethernet is presented. However, Gigabit Ethernet has some differences in protocol and interfaces from 10 Gigabit Ethernet. Moreover, in our case we have to verify in-house Slink interface communication.

For stand-alone verification of the 10 Gigabit Ethernet Controller, we developed two stand-alone UVM test systems based on two different DUTs for FPGA and ASIC-based implementation. The structures of the test systems and approaches used to process verification both of DUTs are presented below.

3.1 Test Systems for FPGA-based 10 Gigabit Ethernet Controller Verification

The top-level module of the 10 Gigabit Ethernet Controller is called XGBE (10 GigaBit Ethernet). The structure of the test system for XGBE stand-alone verification is provided in figure 2.

In the controller, packet is represented as one or multiple (split) descriptors and a payload stored in the system memory. Each transmit and receive descriptor queue in

the device works with continuous area of memory where descriptors are stored. The controller implements head and tail pointer registers used to determine descriptors currently in use. Each descriptor contains a payload pointer. XGBE driver transforms Ethernet packet between representation used by the controller and representation in the test system - UVM-base transaction class.
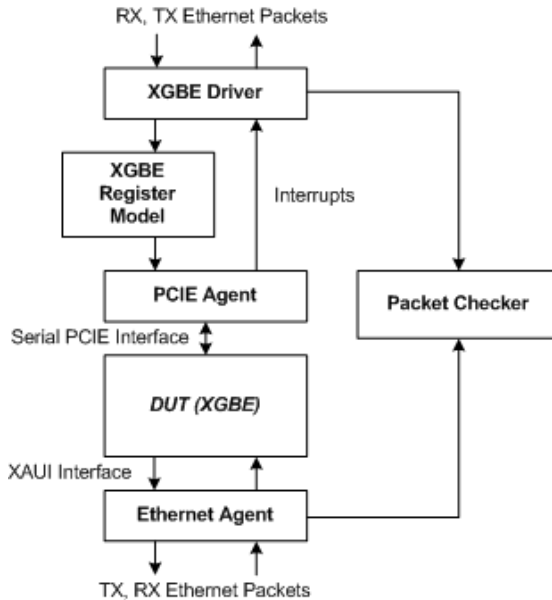

Fig. 2. Structure of XGBE test system.

Algorithm for packet transmission:
- Allocate memory for payload.
- Form corresponding packet descriptor.
- Write descriptor(s) to first free memory location in a queue.
- Change queue head pointer value.
- Wait until tail value becomes equal to head.
- Collect packet transmission information and free used memory resources.

Packet reception algorithm works in a similar way, but because we do not have an information on expected packets sizes, algorithm works in two threads: descriptor preparation and packet reception.

Descriptor preparation works as follows:
- Wait when test system requests additional space for packet reception (conditions of this request are generally test-specific and determined by test system settings).

- Allocate memory for the number of descriptors and fill corresponding descriptor memory.
- Increase RX queue head value.

Packet reception algorithm:

- Wait for change of RX queue tail value.
- Collect received packet data from descriptor and payload.
- Free resources used allocated in descriptor preparation routine.

To simplify access to device registers and abstract away details of register access operations, UVM register model (XGBE Register Model) of the controller was developed. This model uses a bus adapter to transform generic register access operations to the required bus format (in our case - transactions for PCIE agent). Other features of PCIE agent used by device driver are direct access to system memory and interrupt notifications.

Test system was used to verify the device on various test sequences, directed at different device functions. Test can be separated into four large groups: data flow tests, filtering tests, packet parsing tests and throughput tests. The general goal of the first group is to ensure that data processed by the controller will stay correct. At first, maximum possible packet flow through the device was tested. Later we started introducing different bottlenecks (by means of Ethernet Pause frames, PCI Express credits, limiting the size of transmission and reception buffers, available amount of receive descriptors etc...) to achieve different events in the internal components of the controller. The goal of the second group is to ensure correctness of filtering capabilities of device. A set of packets are generated in a way to be test all available packet filters. For the third group, higher-level protocol packets are encapsulated in the basic Ethernet frame and the ability of the device to handle them correctly (packet type detection, automatic checksum calculation etc.) was tested. Throughput tests will be discussed separately later in the article.

It also was decided that the stand-alone verification was necessary for a single type of module in the device - connector between multiple internal packet buses and the external PCI Express like interface (Altera Avalon Interface of PCI Express module), due to several reasons:

- These modules are relatively independent from the rest of the system, its early completion allowed for early verification start.
- High complexity of this module is due to complex rules of transaction splitting.
- Different interactions between all packet bus requesters are difficult to achieve in a complete system.

Its structure is provided in a figure 3. Connector module communicates through Altera PCIE module with the memory inside the PCI Express agent. Information about Ethernet packets is stored in this memory. These data can be separated into two groups: packet descriptors (which are used by the controller to facilitate data

transfer) and payload of packets themselves. Connector module solves several problems. First, it transforms requests from packet bus to PCIE memory access transactions. Second, it transforms the responses to those requests, to format convenient for the rest of the devices. It was decided to include third party PCIE module that (we presume) is well-tested and bug-free because we have access to PCI Express agent, but do not have one for the Avalon interface. Additional performance gained through exclusion of this module is compensated by the time and other resources needed for development of Avalon agent.
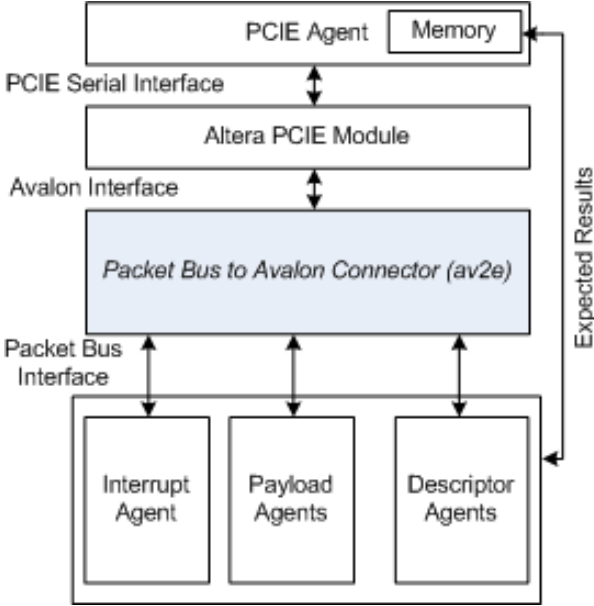

Fig. 3. Structure of av2e module test system.

To verify av2e module (connector to Avalon interface) as a part of 10 Gigabit Ethernet controller a test system was designed. The test system is also designed using UVM and consists of the set of components, which are inherited from standard classes of UVM library.

The exchange between these components is carried out by transactions that facilitates scaling and configurability of system. Developed components could be adapted for use in system for verification of a whole controller to speed up its development.

3.2 Test System for ASIC-based 10 Gigibit Ethernet Controller Verification

Test system for connector between packet buses and system SLink interface module (sl2e) is similar to one used for verification of av2e module. PCI Express agent and

(and Altera PCI Express module) were replaced with SLink agent which provides similar interface for other parts of test system:

- Transformation of transaction-level PCI Express operations into interface signals.
- Access to internal memory of the agent.
- Automatic generation of the completions for upstream requests.
- Notification mechanism for special requests.

The test system for ASIC-based 10 Gigabit Ethernet controller was developed by replacing PCIE Agent with SLink agent.

Usage of various test system components by different test systems is summarized in table 1. Only limited part of device register model (PCI Express Configuration space registers) was actively used in av2e and sl2e test systems.

Table. 1. Test system component reuse.

	FPGA XGBE	av2e	ASIC XGBE	sl2e
Bus Agent	-	+	-	+
PCIE Agent	+	+	-	-
SLink Agent	-	-	+	+
Register Model	+	+/-	+	+/-
XGBE Driver	+	-	+	-

3. Throughput Analyzing

Throughput is one of the most important characteristics of any network controller. In our case, the controller should support throughput of 10 Gigabit per second for packet transmission and reception. Therefore, tests system must support the development of special scenarios for throughput testing. It is implemented in the test system by limiting the test system and device configuration in a way that it will not introduce new "bottleneck".

To achieve necessary controller throughput it is essential to ensure that every component satisfy the requirement. In the controller, PCI Express Gen2 x4 bus was used as a connection to the system. Maximum possible value of throughput for this bus is 16 Gigabit per second. Thus, this bus satisfies the requirements. Throughput of av2e module was measured after its verification was complete. To do that, the throughput analyzer was developed. It executes all necessary calculations using the information about the start time of first packet`s data transmission and the end time of the last one. Initial value of av2e module was ~11.2 Git/s in both directions. This value is higher than maximum packet flow from the network, so this module will not serve as a "bottleneck".

Measurements of throughput of whole controller started after the verification of single packet transfer. Separate packet analyzers were designed for transmission to

network and reception from it. Principle of measurement is similar to the one described above: analyzer collects the information on transmission start time, end time and size of transferred data. A set of tests were developed to determine throughput for different packet flows: (1) transmission, (2) reception, (3) mixed and (4) loopback. In addition, for each of these tests it is possible to select mode of operation: either raw Ethernet packets of mixed UDP/TCP flow to ensure that packet parsing will not slow down the device.

The first measured value of throughput was 0.36 GBit/s in loopback mode. This, of course, does not satisfy the requirements. After multiple performance improvements, the goal to achieve maximum possible throughput for separate transmission and reception packet flow was achieved. At this moment, throughput value for mixed mode 10Gbit/s for reception and 4GBit/s for transmission and 7 GBit/s for both values in loopback mode. Work to further improve the performance is ongoing.

4. Results

To verify the 10 Gigabit Ethernet Controllers four separate test systems were designed using a set of components. Test sequences were developed to test the correctness of the device for each test system.

A total of 49 test scenarios were used to verify all functions of av2e module: read and write operations with different parameters, sequentially and in parallel. Total number of bugs detected by the av2e test system is 14. Found errors are the corruption of transmitted or received data and complete loss of packets by the components.

Number of test for the whole 10 Gigabit Ethernet controller is 31.They thoroughly check that the device works as described in the specification. Different test scenarios check different modes of operation: transmission of packets from system to network, reception of packets from network, mixed flow and loopback mode. The check proper handling of different types of payload (Raw Ethernet, or encapsulated IPv4, IPv6, UDP, TCP, Runt Frames, PTP packets), working with packets with different priorities, working with packets with vlan tags, pause frames, checking of packet filtering capabilities and automatic calculation of checksums. Different ways of interaction with the system are also checked: correctness of interrupts and mirroring of certain device registers in memory.

As a result of verification of the controller, 74 errors were discovered and corrected. Those can be divided into 3 groups:

- Errors in data transmission is the biggest group of 49 errors. Those errors caused the transmission of incorrect data in packets by the controller. This group includes such errors as: partial loss of data, duplication of received data, "merging" of different packets into one and incorrect calculation of checksum.
- Number of errors in packets parsing and filtering is 9. They caused incorrect detection of types of packets, incorrect placement of CRCs in packet and incorrect filtering of received packets.

266

- Group of 16 miscellaneous errors have not caused errors in packet transfer. Those errors appeared during accessing internal registers of the device or caused suboptimal utilization of the device resources.

All above-mentioned errors were corrected. ASIC-based version of the device and its test system are currently under active development. Our future works is aimed at further verification of ASIC-based version of the 10 Gigabit Ethernet Controller, developing UVM-based reusable components (UVC) for PCIE, Avalon, Slink interfaces for using in test system for other network controllers.

References

[1]. IEEE Standard for Ethernet. IEEE Std 802.3-2012. 1983 p.
[2]. Petrochenkov M., Stotland I., Mushtakov R. Approaches to Stand-alone Verification of Multicore Microprocessor Caches. Trudy ISP RAN/Proc. ISP RAS, vol. 28, issue 3, pp. 161-172. DOI: 10.15514/ISPRAS-2016-28(3)-10
[3]. Cyclon V – Overview. URL: https://www.altera.com/products/fpga/cyclone-series/cyclone-v/overview.html (accessed 09.04.2017).
[4]. Avalon Interface Specification. Altera. MNL-AVABUSREF. 2015.12.10. 101 Innovation Drive. San Jose, CA 95134. URL: https://www.altera.com/content/dam/altera-www/global/en_US/pdfs/literature/manual/mnl_avalon_spec.pdf (accessed 09.04.2017).
[5]. Standard Universal Verification Methodology. URL: http://accellera.org/downloads/standards/uvm (accessed 09.04.2017).
[6]. Stotland I., Shpagilev D., Petrochenkov M. Osobennosti funkcional'noj verifikacii kontrollerov vysokoskorostnyh kanalov obmena mikroprocessornyh sistem semejstva "Elbrus" [Features of High Speed Communication Controllers Standalone Verification of "Elbrus" Microprocessor Systems]. Voprosy radioelektroniki, seriya EVT [Issues of Radioelectronics, the series EVT], 2017, 3, pp. 69-75.
[7]. S. Chitti, P. Chandrasekhar, M. Asha Rani. "Gigabit Ethernet Verification using Efficient Verification Methodology". Proc. of International Conference on Industrial Instruments and Control (ICIC), College of Enginnering Pune, India. May 28-30, 2015, pp.1231-1235.

Верификация контроллеров 10 гигабитного Ethernet

М.В. Петроченков <petroch_m@mcst.ru>
Р.Е. Муштаков <mushtakov_r@mcst.ru>
И.А. Стотланд <stotl_i@mcst.ru>
АО «МЦСТ», 117105, Россия, г. Москва, ул. Нагатинская, д.1, стр.1

Аннотация. В статье приведены подходы, использовавшиеся в процессе верификации контроллеров 10 гигабитного Ethernet, разработанных в АО «МЦСТ». Описаны принципы работы устройств – они предоставляют программисту набор регистров, отображаемых в память, а также используют прямой доступ к памяти. Представлен набор подходов, применяемых при верификации подобных устройств – верификация физического прототипа, системная и автономная верификация. Описана мотивация выбора подхода – комбинации системной верификации целого устройства и

автономной верификации одного из его компонентов. В статье дано описание тестовых систем, использовавшихся для верификации устройств. Тестовая система всего устройства осуществляет передачу Ethernet пакетов в сеть и их прием из сети. Разработаны и описаны алгоритмы преобразования пакетов в представление, используемое устройством. Для модулей связи между внутренними пакетными шинами и внешним интерфейсом была разработана автономная тестовая система. При разработке тестовых систем использовалась методология UVM. Выбранная методология, а также предложенные структуры тестовых систем позволили использовать одни и те же компоненты в различных тестовых системах. Приведен набор тестовых сценариев, разработанных для тестовых систем. Особо важным является процесс верификации пропускной способности устройства. Описаны методы, использовавшиеся для измерения пропускной способности устройства, а также режимы работы контроллера, в которых проводилось измерение. Представлены текущие значения пропускной способности устройства, которых удалось достигнуть в различных режимах. В заключение приведен список найденных ошибок, а также описаны те функции устройства, на которые они влияли, а также направления дальнейшей работы.

Ключевые слова: 10 гигабитный Ethernet; контроллер сетевых интерфейсов; верификация; пропускная способность; UVM; тестовая система

Список литературы

[1]. IEEE Standard for Ethernet. IEEE Std 802.3-2012. 1983 p.

[2]. Petrochenkov M., Stotland I., Mushtakov R. Approaches to Stand-alone Verification of Multicore Microprocessor Caches. Trudy ISP RAN, vol. 28, 3, pp. 161-172. DOI: 10.15514/ISPRAS-2016-28(3)-10

[3]. Cyclon V – Overview. URL: https://www.altera.com/products/fpga/cyclone-series/cyclone-v/overview.html (дата обращения 09.04.2017).

[4]. Avalon Interface Specification. Altera. MNL-AVABUSREF. 2015.12.10. 101 Innovation Drive. San Jose, CA 95134. URL: https://www.altera.com/content/dam/altera-www/global/en_US/pdfs/literature/manual/mnl_avalon_spec.pdf (дата обращения 09.04.2017).

[5]. Standard Universal Verification Methodology. URL: http://accellera.org/downloads/standards/uvm (дата обращения 09.04.2017).

[6]. Стотланд И.А., Шпагилев Д.И., Петроченков М.В. Особенности функциональной верификации контроллеров высокоскоростных каналов обмена микропроцессорных систем семейства «Эльбрус». Вопросы радиоэлектроники, серия ЭВТ, 2017, 3, стр. 69-75.

[7]. S. Chitti, P. Chandrasekhar, M. Asha Rani. "Gigabit Ethernet Verification using Efficient Verification Methodology". Proc. of International Conference on Industrial Instruments and Control (ICIC), College of Enginnering Pune, India. May 28-30, 2015, pp.1231-1235.