UDC 004.72:519.21 DOI: [10.33042/2522-1809-2023-6-180-27-32](10.33042/2522-1809-2023-6-180-27-32)

A. Litvinov

O.M. Beketov National University of Urban Economy in Kharkiv, Ukraine

PROBABILISTIC ANALYSIS OF MULTICHANNEL BUS ARBITRATORS OF COMPUTER NETWORK INTERFACES

Probabilistic analysis was conducted on the functioning algorithms of bus arbiters within the 'Common Bus' interface of computer networks. The research employed the theory of random impulse flows as its apparatus. Analytical dependencies were derived to calculate the average waiting time for service requests from network subscribers. The first-come-first-serve and round-robin service algorithms provided better service uniformity than priority algorithms.

Keywords: *algorithm, service, request–impulse, interface, bus arbiter, common bus.*

Introduction

One of the interface types used in computer networks is the 'Common Bus' (Unibus) interface. When transmitting information over such an interface, the problem of conflict resolution arises due to the simultaneous access of several subscribers to the communication channel and the need to service only one of them. To ensure high speed, these conflicts are mainly resolved in hardware according to a specific algorithm that uses bus arbiters or multi-channel priority devices. The bus arbiter, operating under conditions of random flows of events, significantly influences the characteristics of the information transfer process. In this regard, choosing the most rational scheme for constructing and operating a bus arbiter and its development are essential tasks when designing a computer network that must be solved, accounting for all factors affecting the information transfer processes. It is possible to analyse various schemes for constructing bus arbiters and algorithms for their functioning at the logical level only through simulation. **The research aims** to develop and analyse bus arbiter models considering the specifics of the 'Common Bus' interface.

Literature Review

In the study [1], the authors propose a high-level formal model of multiple-bus multiprocessor architecture seen as a component-based system. The proposed model offers a way to implement, analyse, and compare different arbitration protocols using a rich concept of connectors between components. There are no completed analytical dependencies in the study. The paper [2] is devoted to developing a model that provides a way to describe several existing arbitration protocols in a distributed and abstract manner so that their properties and performance can be easily compared and analysed. In [3, 4], the authors developed mathematical models that make it possible to assess the quality of modern computer networks functioning, considering the number of channels, waiting places in network nodes, and the number of network nodes. The research does not account for the spatial nature of the receipt of service requests. In articles [5, 6], analysis of the functioning and efficiency of different bus arbitration schemes is also carried out, but they only focused on a centralised approach.

The paper [7] analyses analytical solutions for traffic prioritisation in packet communication networks. A comparative analysis of known solutions for traffic prioritisation is conducted. The advantages and disadvantages of the described solutions, their impact on the quality of service characteristics in packet communication networks, application features, and general recommendations for using particular solutions are considered. The study [8] proposes a method for calculating the average waiting times and stay times of applications in n-channel systems with relative, absolute, and mixed priorities with arbitrary and different distributions of pure service durations depending on the type of request. The accuracy of the calculations is illustrated by comparison with the results of simulation modelling and recalculations based on 'relational invariants'.

Discussion of Results

Information is transferred through the 'Common Bus' interface based on requests from network subscribers, and the main task of the bus arbiter is to select only one request to use a communication channel from several received from individual computers or network subscribers. In the simplest case, a bus arbiter is built as a 'daisy chain' where subscribers are implicitly assigned relative priorities that depend on the location of the connection to the interface and are higher the closer the connection to the network server. Fig. 1 shows a logical scheme of such a bus arbiter, developed by the author and protected by the copyright certificate [9].

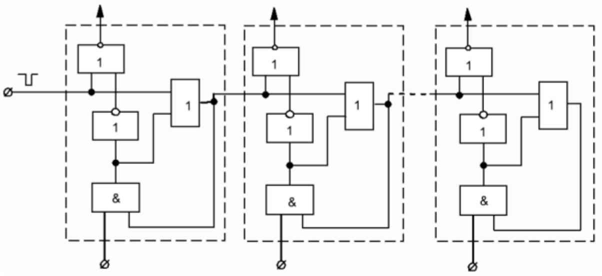

Fig. 1. Logical scheme of a bus arbiter with the ‘daisy chain’ algorithm

When service requests are received, the network server sends a specific polling signal that sequentially passes through the network subscribers and, having found the first subscriber who submitted the request, connects it to the interface. With this algorithm, there is a danger of malfunction of subscribers connected to the interface at the end of the polling line. Therefore, there is a need to use a bus arbiter with a more complex operating algorithm. To date, many bus arbiters have been developed with various operating algorithms: cyclic service, ‘first come, first served’, with arbitrary priority. In this regard, the problem arises of choosing the most rational algorithm for the operation of the bus arbiter and its implementation according to selected efficiency criteria, considering limitations, which are ease of implementation, speed, and a minimum of interface lines used for the operation of the bus arbiter.

To mathematically formalise the processes associated with the functioning of bus arbiters, we will assume that requests for the transmission of information arrays in the form of frames form a Poisson flow with the average value of the time interval between adjacent requests *T*, the duration of information transmission τ. The coefficient of variation of the transmission time of one frame is small (0.25 [10]), and, in this case, for analysis purposes, the transmission time can be considered constant. Thus, incoming requests and the time interval for their service form a rectangular-pulse Poisson random flow. The consequence is the non-Markovian nature of random processes in bus arbiters. Using classical queuing theory to study such processes concerning various service disciplines is extremely difficult. Since we will be mainly interested in the characteristics of the delay in the information frames’ transmission over the interface, the theory of pulsed random flows will be used as a mathematical research tool [11].

Let the number of subscribers served by the communication channel with the MPU be *n*. Intensity of requests received from the *i*-th subscriber is $\lambda_i \ (i = 1,2,..,n)$. Information transfer time for any request is τ. For a stationary state to exist, the following condition must be met:

$$\tau \sum_{i=1}^{n} \lambda_i < 1. \tag{1}$$

For analysis, we will assume that $\lambda_1 \neq \lambda_2 \neq \ldots \neq \lambda_n$, but $\tau_1 = \tau_2 = \ldots = \tau_n = \tau$.

Let us conduct a study for the first-in, first-out algorithm – FIFO. After fixing the beginning of the *i*-th request–impulse, the following combinations are then possible in matching impulse requests:

1. The *i*-th request matches only one of the requests in the flow: either the 1st, the 2nd, or the *k*-th ($k \neq i$) (Fig. 2).

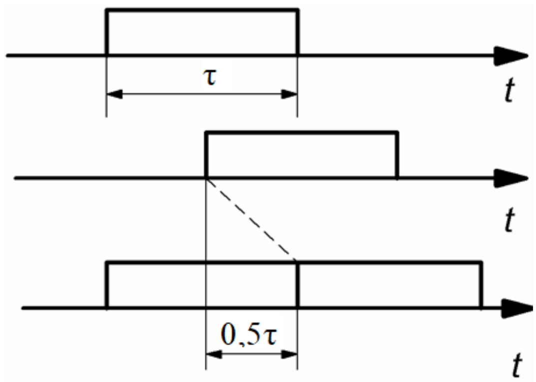

Fig. 2. Coincidence of requests–impulses in time

The probability of this based on [11] will be:

$$p_1^i = \frac{d}{dr} \prod_{\substack{s=1 \\ s \neq i}}^{n} (q_s + r p_s)|_{r=0} = \tau \sum_{\substack{s=1 \\ s \neq i}}^{n} \frac{\prod_{k=1}^{n} (1 - \lambda_k \tau)}{(1 - \lambda_i \tau)(1 - \lambda_s \tau)}, \tag{2}$$

where $p_s = \lambda_s \tau$ is the probability that the request–impulse will fall into the base of the *s*-th request–impulse, and $q_s = 1 - p_s = 1 - \lambda_s \tau$ is the probability that the request–impulse will not fall into the base of the *s*-th request–impulse. In this case, the *i*-th request receives an average delay of $0{,}5\tau$.

2. The *i*-th request matches with two of (*n* – 1) requests of the flows: either the 1st and 2nd, or the 1st and 3rd, and so on. In this case, the request receives a delay of $0{,}5\tau + \tau = 1{,}5\tau$ units of time, and the probability of this equals:

$$p_2^i = \frac{1}{2!} \frac{d^2}{dr^2} \prod_{\substack{s=1 \\ s \neq i}}^{n} (q_s + r p_s)|_{r=0}. \tag{3}$$

Let us introduce the notation:

$$R(k,n,i)=\frac{1}{k!}\frac{d^k}{dr^k}\prod_{\substack{s=1\\ s\neq i}}^{n}\left(q_s+rp_s\right)|_{r=0}. \qquad (4)$$

Then $p_2^i=R(2,n,i)$.

Continuing in the same way, we can derive all other expressions for the probability of coincidence of the *i*-th request–impulse with three, four, and all request–impulse. Accordingly, the *i*-th request–impulse will receive a delay of $2{,}5\tau$, $3{,}5\tau$, $(n-1{,}5)\tau$.

The average delay W_i that the *i*-th request will receive is determined by the expression:

$$W_i=\tau\sum_{s=1}^{n-1}\left(s-0{,}5\right)R(s,n,i). \qquad (5)$$

It should be noted that if $\lambda_1=\lambda_2=\ldots=\lambda_n$ (subscribers are homogeneous, which most often happens in practice), then expression (5) will be simplified:

$$W_i=\tau\sum_{s=1}^{n-1}\left(s-0{,}5\right)\binom{n-1}{s}(\lambda\tau)^s(1-\lambda\tau)^{n-1-s}. \qquad (6)$$

For convenience, we will number the bus arbiter operating algorithms under study: FIFO – number 1, with relative priority – 2, with cyclic service discipline – 3, and the superscript in the expression for the delay time W_i^k will indicate belonging to one or another bus arbiter algorithm.

Next, we will study the functioning of a bus arbiter with an algorithm for servicing requests via the ‘daisy chain’. For this algorithm, each request from the *i*-th subscriber is associated with the *i*-th relative priority determined by the connection to the interface location. The lower the number of the subscriber from whom the application was received, the higher the priority. We will study this mode of operation of the bus arbiter under the same premises as the FIFO algorithm.

Let us denote by $p_{i,k}$ the probability that the moment of receipt of the request–impulse from the *i*-th subscriber will be at the base of the request–impulse from the *k*-th subscriber. From the properties of Poisson flows: $p_{i,k}=\lambda_k\tau$ [12]. Respectively, $q_{i,k}=1-\lambda_k\tau$ is the probability that the *i*-th request–impulse will not overlap the *k*-th impulse. For a request–impulse from the first subscriber, the average delay in service is 0.5τ due to the possibility of its beginning hitting the base of any request–impulse from the *j*-th subscriber $(j=\overline{2,n})$. The probability of this is equal to $\tau\sum_{j=2}^{n}\lambda_j$. The average delay time in servicing requests from the first subscriber equals $W_1^2=0{,}5\tau^2\sum_{j=2}^{n}\lambda_j$.

For a request–impulse from a second subscriber, two types of service delays are possible:

1) the first type with an average delay time of 0.5τ, caused by a single superposition of the beginning of the request–impulse from the second subscriber on any request–impulse from the *j*-th subscriber $(j=1,3,4,\ldots,n)$; the likelihood of this is $\tau[\lambda_1+q_{2,k}(\lambda_3+\lambda_4+\ldots+\lambda_n)]$;

2) the second type with an average delay time of 1.5τ due to the double overlap of the beginning of the second subscriber’s request–impulse into the base of the *j*-th subscriber’s request–impulse $(j=3,4,\ldots,n)$ and simultaneous application of the request–impulse from the first subscriber to the basis of the *j*-th request–impulse. The probability of this, taking into account [11] after appropriate transformations, is equal to $0{,}5(p_{2,3}p_{1,3}+p_{2,4}p_{1,4}+p_{2,n}p_{1,n})=0{,}5\tau\sum_{j=3}^{n}\lambda_j^2$.

As a result, the average delay time in servicing requests from the second subscriber is determined by the expression:

$$W_2^2=0{,}5\tau^2[\lambda_1+(1-\tau\lambda_1)\sum_{j=3}^{n}\lambda_j]+0{,}75\tau^3\sum_{j=3}^{n}\lambda_j^2. \qquad (7)$$

Having made similar calculations regarding requests from the third subscriber, we get:

$$W_3^2=0{,}5\tau^2[\lambda_1q_{2,1}+\lambda_1q_{3,1}+q_{3,1}q_{3,2}\sum_{j=4}^{n}\lambda_j]+ \\ +0{,}75\tau^3\left[\lambda_1\lambda_2+\tau^3(2-\lambda_3\tau)\sum_{j=4}^{n}\lambda_j^2\right]+\frac{2{,}5\tau^4}{3!}\sum_{j=4}^{n}\lambda_j^3. \qquad (8)$$

In the same way, we can obtain an expression for the delay time of servicing requests for the fourth, fifth, and *n*-th subscribers.

In general, the probability that a request–impulse from the *i*-th subscriber will wait in the first, second, or *k*-th queue is determined by the following expression:

$$\theta_{i,k}=\frac{d}{dr^k}\left\{\prod_{j=1}^{i-1}\left(q_j+rp_j\right)\left[1+\sum_{j=i+1}^{n}p_j\right]\right\}\Bigg|_{r=0}. \qquad (9)$$

The average delay time in servicing a request from this subscriber, taking into account (9), will be equal to:

$$W_i^2=\tau\sum_{k=1}^{i}\left(k-0{,}5\right)\theta_{i,k}. \qquad (10)$$

In cases that are important for practice when $\lambda_i = \lambda$, expression (9) will be simplified:

$$\theta_{i,k} = \frac{(i-1)!}{(i-1-k)!} p^k q^{i-1-k} \left[1 + \frac{k(n-i)}{i-k} q\right], \qquad (11)$$

and the expression for the average delay time in servicing requests from the *i*-th subscriber will have the form:

$$W_1^2 = 0{,}5(n-1)\lambda\tau_2;$$
$$W_i^2 = \tau \sum_{k=1}^{i} \frac{(i-1)!(k-0{,}5)(\lambda\tau)^k (1-\lambda\tau)^{i-k-1}}{(i-k-1)!} \times \qquad (12)$$
$$\times \left(1 + \frac{k(n-i)(1-\lambda\tau)}{i-k}\right), i = \overline{2,n}.$$

We can see that the duration of request–impulses significantly affects the delay time in service.

Let us analyse the work of a bus arbiter with a cyclic servicing algorithm. A specific feature of this algorithm is the order in which requests are served in the bus arbiter. At the initial moment, all requests are serviced along a daisy chain, i.e., they are initially assigned priorities: first – first, second – second, and so on. However, the priorities of requests and subscribers, accordingly, change after servicing the current request. If a request from the *k*-th subscriber was serviced, then the highest priority will be assigned to the subscriber with the number *k*+1. Thus, it is necessary to know the distribution of priorities for requests from subscribers, that is, to know with what probability a request from the *i*-th subscriber has priority k $(k = \overline{1,n})$. Let us denote this probability at time *t* by $R_i^k(t)$.

To determine the probabilities $R_i^k(t)$, we will take advantage of the fact that requests are generated by subscribers according to the Poisson law and that the request's service time $\tau << 1/\lambda_i$. Therefore, we can consider that the source's priority changes immediately after the request arrives from it. Using these properties, we can create a system of equations for $R_i^k(t)$, which at $t \to \infty$ will go over to a system of linear equations concerning R_i^k $\left(R_i^k = \lim_{t\to} R_i^k(t)\right)$:

$$\begin{aligned}
&-(\lambda_1+\lambda_2+\lambda_3+\ldots+\lambda_{n-2}+\lambda_{n-1}-\lambda_n)R_i^1+\lambda_n R_i^n=0,\\
&-(\lambda_2+\lambda_3+\lambda_4+\ldots+\lambda_{n-1}+\lambda_n-\lambda_1)R_i^2+\lambda_1 R_i^1=0,\\
&-(\lambda_3+\lambda_4+\lambda_5\ldots+\lambda_n+\lambda_1-\lambda_2)R_i^3+\lambda_2 R_i^2=0,\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\\
&-(\lambda_{n-1}+\lambda_n+\lambda_1+\ldots+\lambda_{n-4}+\lambda_{n-3}-\lambda_{n-2})R_i^{n-1}+\lambda_{n-2}R_i^{n-2}=0,\\
&-(\lambda_n+\lambda_1+\lambda_2+\ldots+\lambda_{n-3}+\lambda_{n-4}-\lambda_{n-1})R_i^n+\lambda_{n-1}R_i^{n-1}=0.
\end{aligned} \qquad (13)$$

This system is complemented by the usual normalisation condition:

$$R_i^1 + R_i^2 + \ldots + R_i^{n-1} + R_i^n = 1. \qquad (14)$$

By replacing one of the system (13) equations with the normalisation condition (14), we can determine all the probabilities of states R_i^k. For small values of *n*, this is easy to do analytically; for large values of *n*, we can use standard routines for solving systems of linear equations included in computer software. Knowing all the probabilities R_i^k and using the average waiting time for each priority obtained for the priority algorithm, the average waiting time for a request from the *i*-th subscriber during cyclic service can be finally determined:

$$W_i^3 = \sum_{k=1}^{n} R_i^k W_k^2 . \qquad (15)$$

Let us consider the most significant case for practice when all subscribers connected to the interface are homogeneous. In this case, the solution to the system of equations (13) and (14) has a simple form $R_i^k = 1/n$, and expression (15) will look like:

$$W_i^3 = \frac{\tau}{n} \sum_{k=1}^{n} \sum_{j=1}^{k} \frac{(k-1)!(k-\lambda j)(j-0{,}5)\lambda^j (1-\lambda\tau)^{i-j-1}}{(k-j)!} . \qquad (16)$$

Fig. 3 shows graphs of the dependence of the waiting time for requests on the intensity of requests received from one source and the type of algorithm.

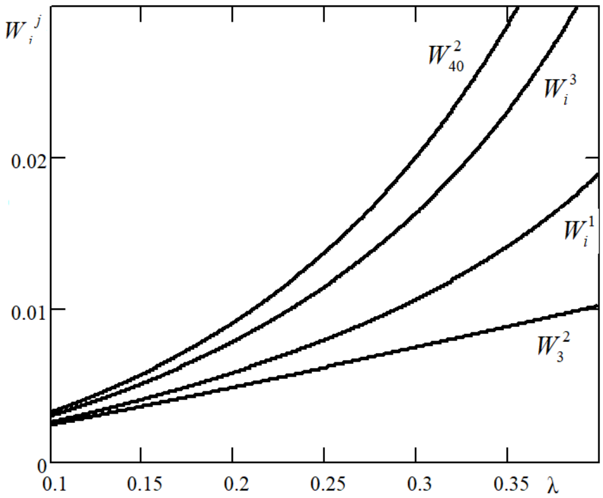

Fig. 3. Dependence of the waiting time for requests on the intensity of requests received from one source and the type of algorithm

With an algorithm with relative priority, there is a spread in the waiting time for requests from different sources, and this spread is greater the greater the difference in the numbers of subscribers from whom the requests are received and the higher the intensity of requests received. FIFO and round-robin service algorithms ensure complete uniformity in service delay time for all subscribers. From this point of view, they are preferable to the priority algorithm on the 'daisy chain'.

Conclusions

An analysis of specific implementation schemes for bus arbiters [9, 13, 14] shows that the hardware implementation of the 'daisy chain' algorithm (with relative priority) is the simplest of the three considered. The most difficult one is based on the FIFO algorithm. From the analysis of dependencies in Fig. 3, we can draw conclusions as follows. If the intensity of requests from their sources is low, preference should be given to implementing a bus arbiter according to the 'daisy chain' scheme. With a high intensity of requests from subscribers and the requirement for uniformity in their service, preference to implement should be given to a more complex FIFO scheme.

References

1. Ben-Hafaiedh, I., & Slimane, M. B. (2016). Model-based Design and Formal Analysis of Arbitration Protocols on Multiple-Bus Architecture. In M. Ghazel, & M. Jmaiel (Eds.), *Proceedings of the 10th Workshop on Verification and Evaluation of Computer and Communication System (VECoS 2016)* (vol. 1689, pp. 177–192). CEUR Workshop Proceedings. Retrieved from https://ceur-ws.org/Vol-1689/paper15.pdf
2. Ben-Hafaiedh, I., Graf, S., & Jaber, M. (2011). Model-based design and distributed implementation of bus arbiter for multiprocessors. *Proceedings of the 18th IEEE International Conference on Electronics, Circuits, and Systems (ICECS 2011)* (pp. 65–68). Institute of Electrical and Electronics Engineers (IEEE). DOI: 10.1109/ICECS.2011.6122215
3. Huseynov, Z. N., Mammadov, M. S., & Ismayilov, T. A. (2023). Modeling and analysis of the characteristics of multichannel and multi-node computer networks with priority service. *IAPGOŚ*, 13(2), 74–77. DOI: 10.35784/iapgos.3394
4. Huseynov, Z. N. (2022). Optimization of the characteristics of single-channel and multi-node computer networks providing priority services. *German International Journal of Modern Science*, (30), 51–57. Retrieved from https://www.calameo.com/read/006382151f80237fa4a38
5. Poletti, F., Bertozzi, D., Benini, L. & Bogliolo, A. (2003). Performance Analysis of Arbitration Policies for SoC Communication Architectures. *Design Automation for Embedded Systems*, 8, 189–210. DOI: 10.1023/B:DAEM.0000003962.54165.5c
6. Doifode, N., Padole, D., & Bajaj, P. R. (2008). Design and Performance analysis of efficient bus arbitration schemes for on-chip shared bus Multi-processor SoC. *International Journal of Computer Science and Network Security (IJCSNS)*, 8(9), 250–255. Retrieved from https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=2babc8a6cdb8fcd6d1b5d9e810ec887a21332ec3
7. Golubenko, V. V., & Lozhkovskii, A. G. (2018). Analysis of analytical decisions on traffic prioritization in packet communication networks. *Proceedings of the O. S. Popov ONAT*, (2), 89–93. Retrieved from http://nbuv.gov.ua/UJRN/Nponaz_2018_2_13 [in Ukrainian]
8. Ryzhikov, Yu. I. (2006). Average waiting and dwell times in multi-channel priority systems. *Information and control systems*, 6, 43–49.
9. Lysenko, E. V., Lytvynov, A. L., & Dikmarov, Ye. F. (1982). *Multichannel priority device* (USSR Patent No. 970372). USSR State Committee for Inventions and Discoveries. Retrieved from https://patents.su/3-970372-mnogokanalnoe-prioritetnoe-ustrojjstvo.html
10. Litvinov, A. L. (2006). Mathematical models of computer networks. *Proceedings of the XXXIX scientific and practical conference of scientific and pedagogical workers, scientists, postgraduate students and employees of the academy* (pp. 106–108). Ukrainian Engineering and Pedagogical Academy (UEPA) [in Ukrainian]
11. Sediakin, M. M. (1965). *Elements of the theory of random impulse flows*. Soviet Radio.
12. Litvinov, A. L. (2018). *Theory of mass service systems: study guide*. O. M. Beketov NUUE. Retrieved from https://eprints.kname.edu.ua/50287/1/2017_ПЕЧ_29Н_Теорія СистМасОбслугов Литвинов.pdf [in Ukrainian]
13. Lytvyn, O. M., Lytvynov, A. L., & Nechuiviter, O. P. (2012). *Multichannel device for connection of subscribers to common bus with cyclic priority service* (Ukraine Patent No. 69115). The State System of Intellectual Property of Ukraine, State Enterprise "Ukrainian Institute of Intellectual Property". Retrieved from https://uapatents.com/4-69115-bagatokanalnijj-pristrijj-dlya-pidklyuchennya-abonentiv-do-zagalno-magistrali-iz-ciklichnim-prioritetnim-obslugovuvannyam.html [in Ukrainian]
14. Lytvynov, A. L., Vasiutin, O. S., & Lavrov, R. V. (2014). *Multichannel device for connection of subscribers to common bus with first-come, first-served service algorithm* (Ukraine Patent No. 89694). The State System of Intellectual Property of Ukraine, State Enterprise "Ukrainian Institute of Intellectual Property". Retrieved from https://uapatents.com/8-89694-bagatokanalnijj-pristrijj-dlya-pidklyuchennya-abonentiv-do-zagalno-magistrali-z-algoritmom-obslugovuvannya-pershim-prijjshov-pershim-obsluzhenijj.html [in Ukrainian]

Література

1. Ben-Hafaied I. Model-based Design and Formal Analysis of Arbitration Protocols on Multiple-Bus Architecture / I. Ben-Hafaiedh, M. B. Slimane // Proceedings of the 10th Workshop on Verification and Evaluation of Computer and Communication System (VECoS 2016), Tunis (Tunisia), October 6–7, 2016 yr. / [ed. by M. Ghazel, M. Jmaiel]. – CEUR Workshop Proceedings, 2016. – Vol. 1689. – P. 177–192. – Regime of access: https://ceur-ws.org/Vol-1689/paper15.pdf, free (date of the application: 05.12.2023).
2. Ben-Hafaiedh I. Model-based design and distributed implementation of bus arbiter for multiprocessors / I. Ben-Hafaiedh, S. Graf, M. Jaber // Proceedings of the 18th IEEE International Conference on Electronics, Circuits, and Systems (ICECS 2011), Beirut (Lebanon), December 11–14, 2011 yr. – New Jersey (USA) : Institute of Electrical and Electronics Engineers (IEEE), 2011. – P. 65–68. – DOI: 10.1109/ICECS.2011.6122215.
3. Huseynov Z. N. Modeling and analysis of the characteristics of multichannel and multi-node computer networks with priority service / Z. N. Huseynov, M. S. Mammadov, T. A. Ismayilov // IAPGOŚ. – 2023. – Vol. 13, No. 2. – P. 74–77. – DOI: 10.35784/iapgos.3394.
4. Huseynov Z. N. Optimization of the characteristics of single-channel and multi-node computer networks providing priority services / Z. N. Huseynov // German International Journal of Modern Science. – 2022. – No. 30. – P. 51–57. – Regime of access: https://www.calameo.com/read/006382151f80237fa4a38, free (date of the application: 05.12.2023).
5. Performance Analysis of Arbitration Policies for SoC Communication Architectures / F. Poletti, D. Bertozzi,

L. Benini, A. Bogliolo // Design Automation for Embedded Systems. – 2003. – Vol. 8. – P. 189–210. – DOI: 10.1023/B:DAEM.0000003962.54165.5c.

6. Doifode N. Design and Performance analysis of efficient bus arbitration schemes for on-chip shared bus Multi-processor SoC / N. Doifode, D. Padole, P. R. Bajaj // International Journal of Computer Science and Network Security (IJCSNS). – 2008. – Vol. 8, No. 9. – P. 250–255. – Regime of access: https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=2babc8a6cdb8fcd6d1b5d9e810ec887a21332ec3, free (date of the application: 05.12.2023).

7. Голубенко В. В. Аналіз аналітичних рішень з пріоритезації трафіка в пакетних мережах зв'язку / В. В. Голубенко, А. Г. Ложковський // Наукові праці ОНАЗ ім. О. С. Попова. – 2018. – № 2. – С. 89–93. – Режим доступу: http://nbuv.gov.ua/UJRN/Nponaz_2018_2_13, вільний (дата звернення: 05.12.2023).

8. Рижиков Ю. І. Середні часи очікування і перебування в багатоканальних пріоритетних системах / Ю. І. Рижиков // Інформаційно-управляючі системи. – 2006. – № 6. – С. 43–49.

9. Багатоканальний пріоритетний пристрій : пат. 970372 СРСР, МПК G06F 9/46 / Е. В. Лисенко, А. Л. Литвинов, Є. Ф. Дікмаров (УРСР) ; Харків. авіац. ін-т ім. М. Є. Жуковського. – № 3268828/18-24 ; заявл. 02.04.81 ; опубл. 30.10.82, Бюл. № 40. – Режим доступу: https://patents.su/3-970372-mnogokanalnoe-prioritetnoe-ustrojjstvo.html, вільний (дата звернення: 05.12.2023).

10. Литвинов А. Л. Математичні моделі мереж ЕОМ / А. Л. Литвинов // XXXIX науково-практична конференція науково-педагогічних працівників, науковців, аспірантів та співробітників академії : зб. тез доп., Харків, 26–27 січ. 2006 р. / Укр. інж.-пед. акад. – Харків : УІПА, 2006. – С. 106–108.

11. Седякін М. М. Елементи теорії випадкових імпульсних потоків / М. М. Седякін. – М. : Радянське радіо, 1965. – 263 р.

12. Литвинов А. Л. Теорія систем масового обслуговування : навч. посіб. / А. Л. Литвинов ; Харків. нац. ун-т міськ. госп-ва ім. О. М. Бекетова. – Харків : ХНУМГ ім. О. М. Бекетова, 2018. – 141 с. – Режим доступу: https://eprints.kname.edu.ua/50287/1/2017_ПЕЧ_29Н_ТеоріяСистМасОбслугов_Литвинов.pdf, вільний (дата звернення: 05.12.2023).

13. Багатоканальний пристрій для підключення абонентів до загальної магістралі із циклічним пріоритетним обслуговуванням : пат. 69115 Україна, МПК G06F 13/37 / О. М. Литвин, А. Л. Литвинов, О. П. Нечуйвітер (Україна) ; Укр. інж.-пед. акад. – № u201110110 ; заявл. 16.08.11 ; опубл. 25.04.12, Бюл. № 8. – Режим доступу: https://uapatents.com/4-69115-bagatokanalnijj-pristrijj-dlya-pidklyuchennya-abonentiv-do-zagalno-magistrali-iz-ciklichnim-prioritetnim-obslugovuvannyam.html, вільний (дата звернення: 05.12.2023).

14. Багатоканальний пристрій для підключення абонентів до загальної магістралі з алгоритмом обслуговуванням "першим прийшов-першим обслужений" : пат. 89694 Україна, МПК G06F 13/37 / А. Л. Литвинов, О. С. Васютін, Р. В. Лавров (Україна) ; Укр. інж.-пед. акад. – № u201314425 ; заявл. 09.12.13 ; опубл. 25.04.14, Бюл. № 8. – Режим доступу: https://uapatents.com/8-89694-bagatokanalnijj-pristrijj-dlya-pidklyuchennya-abonentiv-do-zagalno-magistrali-z-algoritmom-obslugovuvannya-pershim-prijjshov-pershim-obsluzhenijj.html, вільний (дата звернення: 05.12.2023).

Reviewer: Doctor of Technical Sciences, Professor, Head of the Department of Computer Science and Information Technology Oleh Fedorovych, National Aerospace University "Kharkiv Aviation Institute", Ukraine.

Author: LITVINOV Anatolii
Doctor of Technical Sciences, Professor
O.M. Beketov National University of Urban Economy in Kharkiv
E-mail – litan6996@gmail.com
ID ORCID: http://orcid.org/0000-0001-7063-7814

ІМОВІРНІСНИЙ АНАЛІЗ БАГАТОКАНАЛЬНИХ ШИННИХ АРБІТРІВ ІНТЕРФЕЙСІВ ОБЧИСЛЮВАЛЬНИХ МЕРЕЖ

А.Л. Литвинов
Харківський національний університет міського господарства імені О.М. Бекетова, Україна

Під час передачі інформації з інтерфейсу виникає проблема вирішення конфліктів через одночасне звернення кількох абонентів до одного каналу зв'язку та необхідності обслуговування лише одного з них. З метою забезпечення високої швидкості ці конфлікти переважно вирішуються апаратно за певним алгоритмом на основі використання шинних арбітрів. Шинний арбітр, функціонуючи за умов випадкових потоків подій, істотно впливає на характеристики процесу передачі інформації. У зв'язку з цим вибір найбільш раціональної схеми побудови та функціонування шинного арбітра та його розробка є важливим завданням при проєктуванні комп'ютерної мережі, яку необхідно вирішувати з урахуванням усіх факторів, що впливають на процеси передачі інформації. На вибір схеми побудови шинного арбітра та алгоритму його функціонування впливає багато факторів. Проаналізувати різні схеми побудови шинних арбітрів та алгоритми їх функціонування на логічному рівні можна лише за допомогою моделювання. Метою досліджень є розробка та аналіз моделей шинних арбітрів з урахуванням специфіки інтерфейсу «Загальна шина». Шинний арбітр функціонує за умов випадкових потоків подій. Як апарат досліджень обрано теорію випадкових імпульсних потоків, що дозволяє врахувати специфіку передачі інформації комп'ютерною мережею – передача по кадрам. Аналіз функціонування шинних арбітрів проведений у припущенні, що запити на передачу масивів інформації утворюють пуасонівський потік подій, а запити, що надходять, і інтервал часу їх обслуговування утворюють прямокутно-імпульсний пуасонівський випадковий потік. Проведено дослідження трьох основних алгоритмів, які використовуються при арбітражі в інтерфейсі: «перший прийшов – першим обслужений» (FIFO), з відносним пріоритетом та з циклічним алгоритмом обслуговування. Виведено вирази для часу затримки заявки від джерела запитів залежно від інтенсивності надходження та часу передачі інформації.

***Ключові слова**: алгоритм, обслуговування, заявка-імпульс, інтерфейс, шинний арбітр, загальна шина.*