

**Борисенко Юлия Васильевна**

Юго-Западный государственный университет, кафедра «Вычислительная техника»

Аспирант

*Borisenko Julia Vasilievna*

*South-West State University*

*Graduate student*

E-Mail: [jv.sokolova@mail.ru](mailto:jv.sokolova@mail.ru)

Элементы и устройства вычислительной техники и систем управления

## **Акселератор перераспределения подпрограмм в отказоустойчивых мультимикомпьютерах**

Reallocation routines in the accelerator fault tolerance multicomputers

**Аннотация:** В статье рассмотрена проблема отказов в мультипроцессорных системах, обоснована необходимость оперативной реакции на отказ в системах высокой готовности. Рассмотрены отказы процессоров и межпроцессорных связей, для которых предложены соответствующие аппаратные средства, показана временная и аппаратная сложность.

**Abstract:** The article deals with the problem of failures in multiprocessor systems, the necessity of rapid reaction to failures in the systems highly available. Considered failures of processors and interprocessor communications, which suggest appropriate hardware, see the timing and complexity of the hardware.

**Ключевые слова:** Мультипроцессорная систем; процессор; отказ; межпроцессорная связь; размещение; система высокой готовности.

**Keywords:** Multiprocessor system; processor; failure; interprocess communications; accommodation; high-availability systems.

\*\*\*

В настоящее время все большее распространение получают *отказоустойчивые мультимикомпьютерные системы (ОМС)* [1]. При этом к ним повышаются требования по быстродействию, и качеству решения задач. При использовании ОМС в системах высокой готовности (системы мониторинга, слежения за высокоскоростными предметами, бортовой аппаратуры авиации т.д.) в случае отказа в них процессора или межпроцессорной связи необходимо быстрое восстановление правильности функционирования. Возможна реконфигурация структуры с отключением неисправного процессора и заменой его резервным, расположенным обычно вне поля обрабатывающих процессоров либо поиска альтернативного маршрута пути обхода [2]. С целью уменьшения общего времени восстановления ОМС, необходимо многократно снизить затраты на замену отказавшего процессорного модуля резервным, последующие перераспределение подпрограмм и/или поиск маршрута обхода отказавшей межпроцессорной связи. Работа является продолжением исследований, начатых в [3,4]. Из анализа, проведенного в [3,4] следует необходимость разработки устройства, реализующее предложенные алгоритмы.

Формализованная постановка задачи отказоустойчивого перераспределения в мультимикомпьютерных системах представлена в [3,4]. Исходная задача представляется в виде графа  $G = \langle X, E \rangle$ , вершины  $x_{q,v}$  которого соответствуют задачам (подпрограммам), а дуги

$e_{ij} \in E$  связям между ними, которые передаются между подпрограммами и сведены в матрицу смежности (МС)  $M = \|m_{ij}\|_{N \times |E|}$ , где  $N = |X|$ .

Топология мультимодульного компьютера задается графом  $H = \langle P1, V \rangle$ , где  $P1$  соответствует процессорным модулям, а  $V$  – межмодульным связям. Множество  $P1$  разбивается на два непересекающихся подмножества  $P1 = P \cup L$ , где  $\{P\}$  – множество основных процессоров, а  $\{L\}$  – множество резервных процессоров. Идентификаторы множества  $P$  упорядочим в виде матрицы  $P = \|p_{ij}\|_{n \times n}$ , где  $n = |P|$ . Множество резерва  $L$  представим в виде матрицы  $L = \|l_{ij}\|_{n \times n}$ .

Размещение пакета подпрограмм, описываемых графом  $G$ , в мультимодульном компьютере может быть аналитически описано отображением:

$$\beta_s = X_s \rightarrow P_1 = \left\{ \begin{array}{cccc} x_{s1.1} & x_{s1.2} & \dots & x_{s1.v} & \dots & x_{s1.n} \\ x_{s2.1} & x_{s2.2} & \dots & x_{s2.v} & \dots & x_{s2.n} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ x_{sq.1} & x_{sq.1} & \dots & x_{sq.v} & \dots & x_{sq.n} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ x_{sn.1} & x_{sn.2} & \dots & x_{sn.v} & \dots & x_{sn.n} \end{array} \right\} \rightarrow \left\{ \begin{array}{cccc} p_{1.1}l_{1.1} & p_{1.2}l_{1.2} & \dots & p_{1.v}l_{1.v} & \dots & p_{1.n}l_{1.n} \\ p_{2.1}l_{2.1} & p_{2.2}l_{2.2} & \dots & p_{2.v}l_{2.v} & \dots & p_{2.n}l_{2.n} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ p_{q.1}l_{q.1} & p_{q.2}l_{q.2} & \dots & p_{q.v}l_{q.v} & \dots & p_{q.n}l_{q.n} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ p_{n.1}l_{n.1} & p_{n.2}l_{n.2} & \dots & p_{n.v}l_{n.v} & \dots & p_{n.n}l_{n.n} \end{array} \right\}, \quad (1)$$

где  $s$  – номер варианта размещения задач  $\{x_{qk}\}$  по процессорным модулям  $\{P_{qv}\}$ ,  $s = \overline{1, N!}$ , символ  $\rightarrow$  – это отображение одной из вершин графа  $G$  на один из процессоров  $P$ . Мощность множества всевозможных отображений  $\Psi = \{\beta_s\}$  равна числу перестановок задач  $\{x_{qv}\}$  в матрице  $X$ :  $|\Psi| = N!$ .

Пусть  $\Psi$  это множество всевозможных отображений вида (1), тогда задачу размещения можно сформулировать как поиск такого отображения  $\beta^* \in \Psi$ , что

$$T_{\beta^*} = \min_{\Psi} \left\{ \max_{\beta \in \Psi} \left\{ T_{\beta}(p_{a,b}, p_{x,y}) \right\} \right\}, \quad (2)$$

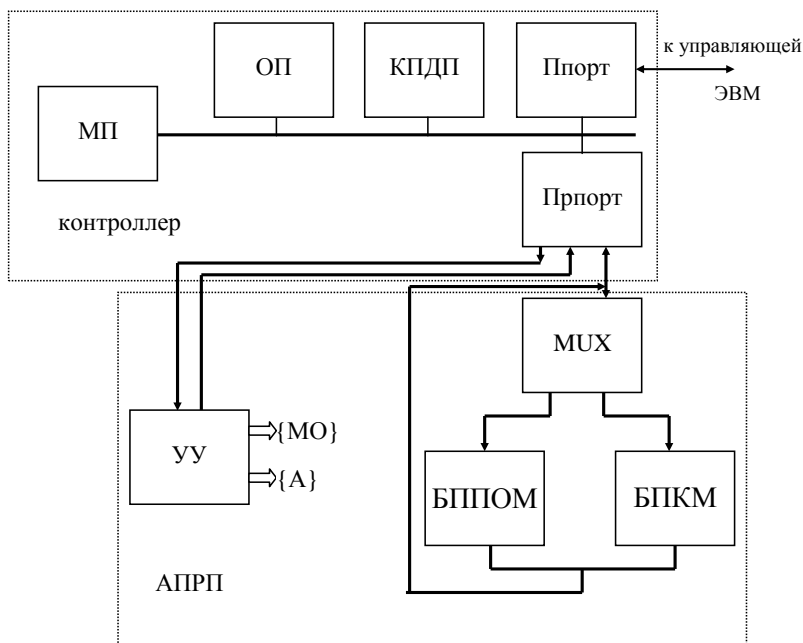
где  $T_{\beta}(p_{a,b}, p_{x,y})$  – коммуникационная задержка, определяемая временем обработки принятых данных и/или подготовки пакета передаваемых данных между процессорными модулями  $p_{a,b}$  и  $p_{x,y}$ , соответствующих отображению  $\beta^*$ . При этом первоначально вычисляется недостижимая минимальная оценка  $L_{\inf}$  из предположения, что дуги графа  $G$  назначаются без учета влияния топологии исходной подпрограммы. Она вычисляется как:

$$T_{\beta}(p_{a,b}, p_{x,y}) = \sum_{i=1, j=1}^N d_{ij} \cdot m_{ij} \quad (3)$$

В случае отказа процессора  $p_{\alpha, \beta}$  ( $\alpha = \overline{1, n}$ ,  $\beta = \overline{1, n}$ ), описываемых графом  $G$  или межпроцессорной связи, применяется процедура отказоустойчивого перераспределения подробно изложенная в [4], основанная на целенаправленных поисковых перестановках строк и столбцов МС и сравнении полученной с помощью (1), (2) величины коммуникационной

задержки с  $L_{inf}$ , вычисленной на основе критерия (3).

Для решения поставленной задачи применен параллельный отказоустойчивый акселератор перераспределения подпрограмм, структурная схема которого приведена на рис. 1.



**Рис. 1.** Структурная организация акселератора отказоустойчивого перераспределения

Ведущая ЭВМ мультипроцессорной системы передает в акселератор планирования перераспределения подпрограмм (АПРП) исходные данные, необходимые для оперативной реакции системы на либо на отказ процессорного модуля системы, либо на отказ межпроцессорной связи, а именно: матрицу смежности, матрицу расстояний, матрицу исправности процессоров, матрицу работоспособности и занятости резервных процессоров и матрицу исправности межпроцессорных связей.

Акселератор планирования перераспределения состоит из двух независимых блоков: блока перераспределения отказавших процессорных модулей (БППОМ) и блока поиска кратчайшего маршрута (БПКМ) в случае отказа межпроцессорной связи. Такое разделение акселератора на индивидуальные блоки позволяет независимо реагировать на отказ внутреннего процессорного модуля и/или отказ межпроцессорной связи.

АПРП представляет собой аппаратный комплекс, на базе микропроцессорного контроллера и функционирует на основе алгоритмов планирования перераспределения, описанных в [3,4].

Результатом работы акселератора планирования размещения задач является файл конечной матрицы смежности, матрицы расстояний и матрицы исправности межпроцессорных связей, соответствующей найденному варианту размещения и/или кратчайшего маршрута обхода.

Для подключения блока АПРП к контроллеру используется порт параллельного интерфейса типа LPT (Line PrinTer). [5].

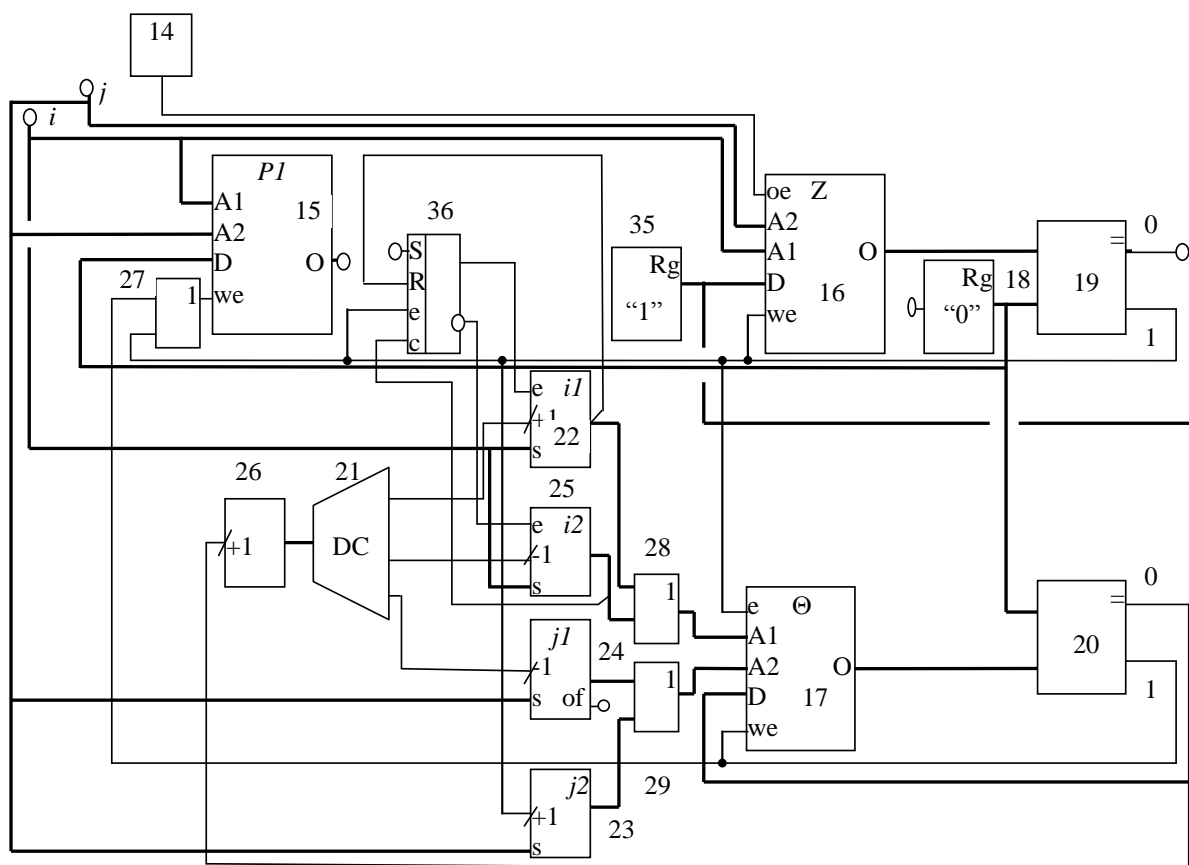
На рисунке 1 приняты следующие обозначения блоков и узлов: АПРП – акселератор планирования перераспределения подпрограмм; МП – микропроцессор; ОП – оперативная память; КПДП – контроллер прямого доступа в память; Ппорт – последовательный порт; Прпорт – параллельный порт; УУ – устройство управления; БППОМ - блока перераспределения отказавших процессорных модулей; БПКМ - блока поиска кратчайшего маршрута; МО –

микрооперации; A – адрес.

МП контроллера работает в соответствии с программой, записанной в ОП и отслеживает работу системы высокой готовности. В случае возникновения отказа процессорного модуля мультипроцессорной системы, либо сбоя межпроцессорной связи соответствующие исходные данные передается в один из блоков.

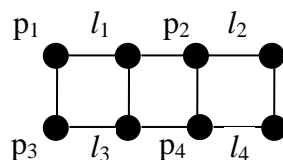
В случае поступления через последовательный порт с управляющей ЭВМ сигнала об отказе процессорного модуля либо межпроцессорной связи соответствующий сигнал подается в контроллер и далее в соответствии с записанной в МП программой через параллельный порт поступает в специализированный мультиплексор MUX, который в зависимости от отказоустойчивой ситуации (отказал процессорный модуль либо межпроцессорная связь) передает необходимые для оперативной реакции данные либо в блок БППОМ либо в БПКМ.

Функциональная схема устройства оперативной замены отказавшего процессорного модуля резервным представлена на рисунке 2, а устройства поиска кратчайшего пути – на рисунке 4.



**Рис. 2.** Функциональная схема устройства оперативной замены отказавшего процессорного модуля

В ОЗУ 15 хранится матрица  $P1$  процессорных модулей и матрица  $L$  резервных процессоров, так как показано на рисунке 3.



**Рис. 3.** Матричная организация мультикомпьютера

Нам рисунке 3 буквами  $p$  обозначены основные процессорные модули, а буквами  $l$  – резервные процессоры.

Работа устройства происходит следующим образом. В случае появления на входах  $j$  и  $i$  двоичных кодов происходит внештатная ситуация в *системе логического управления (СЛУ)*, вызванная сбоем процессора. В этом случае необходима оперативная реакция системы на отказ, связанная с заменой отказавшего процессорного модуля резервным. Это происходит следующим образом.

Первоначально происходит подготовка к замене отказавшего процессора. Для этого импульс с выхода генератора 14 импульсов поступает на вход разрешения выдачи ОЗУ 16, а так как на адресных входах  $A1$  и  $A2$  присутствуют соответствующие коды, то значение с выхода ОЗУ 16 подается на первый вход элемента 19 сравнения, на втором входе которого присутствует код нуля с выхода регистра 18. В результате сравнения происходит анализ исправности процессора. В случае если результат сравнения положительный (процессор исправен), он должен быть отмечен как неисправный, поэтому единичный импульс со второго выхода элемента сравнения 19 подается на вход  $we$  разрешения записи ОЗУ 16. Поэтому единица с выхода регистра 34 поступает на вход данных  $D$  ОЗУ 16 и заносит туда код единицы, как признак неисправности данного процессора.

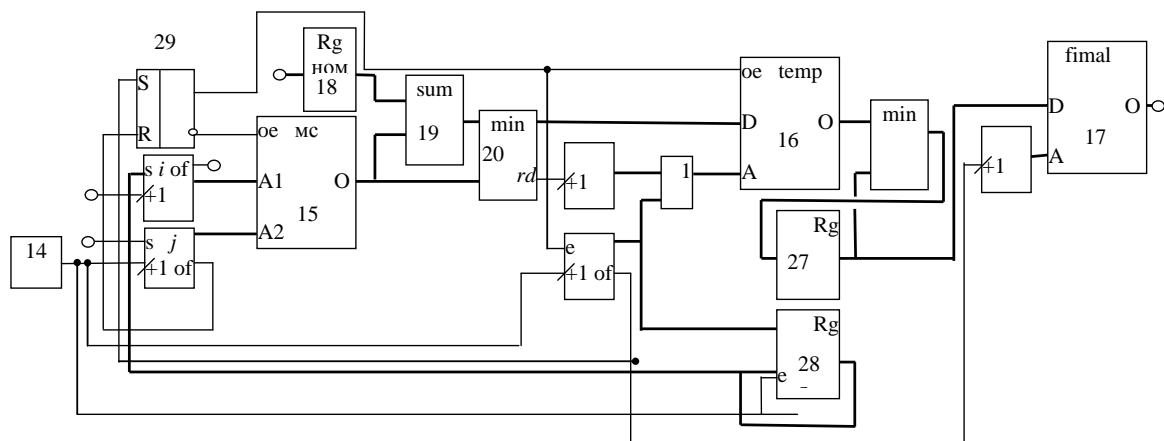
Одновременно единичный потенциал с выхода элемента сравнения 19 подается на второй вход элемента 27 ИЛИ и счетный вход счетчика 23. Единица поступает на вход  $we$  разрешения записи ОЗУ 15, чем разрешает запись нулевого кода с выхода регистра 16, что означает признак неисправности процессора с данным адресом в матрице  $P1$ .

Далее выполняется замена неисправного процессора резервным, что происходит следующим образом. Счетчики 22-25 выполняют роль соседних резервных процессоров для потенциального отказавшего.

При появлении единичного потенциала на выходе элемента 19 сравнения, он поступает на счетный вход счетчика 24 и увеличивает его значение по переднему фронту на единицу. Этот код поступает на вход  $A2$  ОЗУ 17. Соответствующий код с выхода ОЗУ 17 подается на второй вход элемента 20 сравнения для сравнения с кодом нуля, поступающим с выхода регистра 18. В результате сравнения, если резервный процессор не занят, то единичный потенциал со второго выхода элемента 20 сравнения поступает на  $we$  вход ОЗУ 17 для единичного кода, означающего его занятость, в результате замены на отказавший процессор.

В случае занятости резервного процессора в следующих тактах выполняются аналогичные сравнения для определения свободного резервного процессора и фиксация свободного как занятого вместо основного отказавшего.

Появление сигнала переполнения на выходе переполнения счетчика 34 означает, что необходимость полной замены матрицы процессоров.



**Рис. 4.** Функциональная схема устройства поиска кратчайшего пути

Работа устройства, представленного на рисунке 4 происходит следующим образом.

В МС 15 хранится матрица смежности исходной задачи, в регистре 27 хранится код FFFF, как максимально возможное для данной применяемой разрядности. В счетчике 22 и 23 установлен код адреса, хранящийся в ОЗУ 15, что фактически является координатами отказавшей межпроцессорной связи.

Так входах A1 и A2 присутствуют коды отказавшей процессорной связи, то соответствующий код из ОЗУ 15 поступает на второй вход сумматора 19 и на вход блока нахождения минимума 20. В результате, в сумматоре 19 находится сумма кодов, которая поступает на вход блока нахождения минимума 20. В результате, полученное значение поступает на D-вход ОЗУ 16. В это время с разрешающего rd выхода блока нахождения минимума 20 поступил единичный импульс на вход счетчика 25 и увеличил его содержимое на единицу, которое поступает на A вход ОЗУ 16. В результате, значение из ОЗУ 16 поступает на вход блока нахождения минимума 21, на другом входе которого присутствует код FFFF. В результате сравнения минимальный код заносится в регистр 27.

Появление единичного импульса на выходе переполнения счетчика 26 означает, что первый цикл поиска кратчайшего пути закончен, и полученный результат сохраняется в ОЗУ 17, как первое «звено» в искомом кратчайшем пути.

Аналогично продолжается до тех пор, пока на выходе переполнения счетчика 22 не появится единичный импульс, который означает, что все коды в ОЗУ 15 исследованы и в ОЗУ 17 записаны все кратчайшие пути для исходного графа G. Соответственно, коды длин кратчайших путей могут быть поданы на ВУУ для анализа через D-выход ОЗУ 17.

Для анализа производительности предложенного акселератора воспользуемся серией 1533 интегральных микросхем. При этом при подсчете быстродействия и производительности учитывались временные характеристики отдельных внутренних элементов предложенного устройства (таблица 1) согласно серии микросхем ТТЛ 1533 [6].

**Таблица 1**

**Временные характеристики элементов микросхем серии 1533**

Тип элемента	$t_{зд.р.}, нс$
ИЛИ	12
Сумматор	36
Элемент сравнения	18
RS-триггер	20
Регистр	21
Счетчик	18
ОЗУ	30

На основе функциональной схемы (рис. 2) подсчитано количество использованных внутренних микросхем, представленных в табл. 2.

**Таблица 2**

**Количество микросхем устройства оперативной замены отказавшего модуля**

Тип элемента	Количество
ИЛИ	1
Сумматор	1
Элемент сравнения	2
RS-триггер	1
Регистр	2
Счетчик	5
ОЗУ	3

На основе данных таблиц 1 и 2 можно подсчитаны временные характеристики предложенного устройства, на основе чего была составлена временная диаграмма зависимости скорости работы устройства от размерности задачи (рис. 5).



**Рис. 5.** Зависимость скорости работы устройства от размерности матрицы процессоров

Из анализа представленной диаграммы можно сделать вывод, что при относительно небольшой размерности матрицы процессоров ( $3 \times 3 - 5 \times 5$ ) время работы устройства составляет несколько нс. С ростом размерности количество вариантов обхода резко увеличивается, что приводит к росту времени работы устройства. Отсюда в свою очередь можно сделать вывод о целесообразности применения предложенного устройства для систем

высокой готовности, в которых размерность матрицы процессоров от  $50 \times 50$  и выше, в которых применение программных средств неприемлемо из-за большого времени поиска.

Для оценки аппаратной сложности устройства использовалось представление внутренних модулей устройства в виде набора эквивалентных вентилях (под эквивалентным вентилях здесь будем понимать логический элемент И) [6], отраженных в таблице 3

Таблица 3

Количество внутренних элементов устройства

Тип элемента	Количество	Количество эквивалентных вентилях
ИЛИ	1	1
Сумматор	1	16
Элемент сравнения	2	10
RS-триггер	1	2
Регистр	2	32
Счетчик	5	17
ОЗУ	3	16 на 1 ячейку

В результате расчетов был получен график зависимость роста аппаратной сложности размерности матрицы процессоров (рис. 6).



Рис. 6. Зависимость роста аппаратной сложности от размерности матрицы процессоров

Из представлено диаграммы можно сделать вывод об увеличении аппаратной сложности устройства в зависимости от размерности матрицы процессоров. Из рис. 6 видно, что при относительно большой размерности матрицы процессоров ( $9 \times 9$  и выше) необходимо 10988 элементарных вентилях И, что при современном уровне развития технологий является небольшим количеством.

Из анализа рисунков 5 и 6 можно сделать вывод о целесообразности применения предложенного метода и разработанной процедуры отказоустойчивого перераспределения в мультикомпьютерных системах программным способом для относительно небольших размерностей матрицы процессоров (в пределах 10). Однако с увеличением размерности задачи, например в случае использования систем высокой готовности, (размерность в пределах 50-100), целесообразно использования предложенных акселераторов.



## ЛИТЕРАТУРА

1. Воеводин В.В., Воеводин Вл.В., Параллельные вычисления // БХВ– Петербург. – Санкт-Петербург, 2002.– 608 с.
2. Зотов И.В. Организация и синтез микропрограммных мультимикроконтроллеров. / И.В. Зотов // Курск.: Изд-во «Курск», 1999. – 368 с.
3. Соколова, Ю.В. [Переразмещение подпрограмм в отказоустойчивых мультипроцессорных системах](#) [Текст] / Ю.В. Соколова, Д.Б. Борзов, В.В. Минайлов // Известия вузов. Приборостроение. – Санкт-Петербург, – 2013, – №6, С. 39-44.
4. Борисенко, Ю.В. Метод оперативного переразмещения подпрограмм в мультимикроконтроллерах с учётом отказов линков [Текст] / Ю.В. Борисенко, Д.Б. Борзов, А.С. Сизов // Известия ЮЗГУ, – 2012, – №6(45), С. 50-54.
5. Гук М. Аппаратные интерфейсы ПК. – СПб.: Питер, 2003.– 528 с.
6. Петровский И.И. и др. Логические ИС КР1533, КР1554: Справочник в 2-х ч. – М.: ТОО «Бином», 1993.

**Рецензент:** А.В. Кониченко, д.т.н., с.н.с., ГНС НИЦ (г. Курск) ФГУП «18 ЦНИИ» МО

РФ