

**МИКРОПРОЦЕССОРНЫЕ СИСТЕМЫ  
РЕАЛЬНОГО ВРЕМЕНИ**

УДК 681.32

**В. А. Колосков, В. С. Титов***(Курск)***УПРАВЛЯЮЩАЯ СИСТЕМА С САМООРГАНИЗУЮЩИМ СЛОЕМ**

Предлагается метод самоорганизации отказоустойчивой мультимикроконтроллерной сети, обеспечивающей оперативное восстановление ее логической структуры и перенастройку микроконтроллеров при их отказах. Сформулирована задача самоорганизации, и рассмотрены способы преобразования логической структуры сети и реализация алгоритмов настройки сети.

**Введение.** В работе рассматривается проблема эффективной организации перестройки логической структуры (самоорганизации) мультимикроконтроллерной управляющей системы (МУС) при отказах ее элементов. Микроконтроллер (МК) МУС представляется композицией коммуникационного устройства (КУ) и устройства обработки данных. В настоящее время средства восстановления исходной логической структуры управляющей системы строятся по централизованному или распределенному (децентрализованному) принципу [1]. Перспективным представляется подход к построению децентрализованных средств самоорганизации, связанный с перераспределением функций между схемным и алгоритмическим уровнями. В этом случае аппаратно-алгоритмические средства, представляющие слой самоорганизации, позволяют существенно ускорить процесс «стабилизации» логической структуры системы.

Известны методы распределенной самоорганизации для линейных [2] и матричных мультипроцессоров [3—5], позволяющие восстанавливать их логическую структуру при отказах. Перестройка матричных мультипроцессоров сводится к реконфигурации связей между процессорными элементами за счет распределенных средств коммутации. Данный подход в управляющих системах, представленный сетью МК, приводит к увеличению времени восстановления вследствие перераспределения алгоритмов и неизбежного усложнения и перепроектирования связей МУС при встраивании между входами и выходами микроконтроллеров коммутационных средств реконфигурации при отказах.

Сохранение прежних физических связей между МК, служащих для обмена данными, достигнуто в самоорганизующейся МУС с настраиваемыми КУ (НКУ), обеспечивающими поиск приемника данных при отказах МК с помощью дополнительных средств реконфигурации маршрута сообщения. Для поиска МК-приемника НКУ использует логический адрес, получаемый от средств самоорганизации. В целом действия по самоорганизации состоят в перенумерации логических адресов и перераспределении алгоритмов, а также в формировании маршрутов сообщений в МУС с учетом отказов МК.

В данной работе предложена оперативная перестройка управляющей системы из настраиваемых многоалгоритмических микроконтроллеров (НМК) с НКУ за счет слоя самоорганизации. Самоорганизующий слой (СОС) обеспечивает вычисление новых логических адресов НМК и перенастройку каж-

дого из работоспособных НМК на один из множества закрепленных за ним алгоритмов в соответствии с ситуацией отказа. СОС не требует перестройки связей между НМК, перезагрузки алгоритмов, а перепроектирование системы при увеличении направлений обхода отказавших НМК сводится к перепрограммированию слоя самоорганизации. В связи с этим разработка СОС сводится к определению методов вычисления логических адресов микроконтроллеров, настройки каждого из них в соответствии с новым логическим адресом и состоянием управляющего процесса и аппаратурной реализации слоя. Рассмотрено формирование маршрутов сообщений между НМК с логическими адресами, перенумерованными СОС, при допущении, что отказ НМК не приводит к отказу коммуникационного устройства [6], которое продолжает осуществлять транзитные передачи между работоспособными НМК.

1. **Постановка задачи.** Представим сеть НМК в виде множества  $M$  микроконтроллеров, отображенных в поле ячеек, каждая из которых характеризуется своим физическим адресом места расположения микроконтроллеров. Поле ячеек опишем матрицей  $S$ , в которой физический адрес каждой ячейки определяется номерами строки и столбца.

Каждому частному алгоритму общего алгоритма управления  $U$ , отображаемому на множество микроконтроллеров, поставим в соответствие физический адрес ячейки, в которой находится НМК. Получим  $U \rightarrow M$ ,  $U = \{U_1 \dots U_j \dots U_n\}$ ,  $M = \{m_1 \dots m_i \dots m_r\}$ ,  $M \rightarrow S$ ,  $S = \{S_{00} \dots S_{ab} \dots S_{AB}\}$ , где  $U_j$  —  $j$ -й частный алгоритм;  $m_i$  —  $i$ -й НМК;  $S_{ab}$  — ячейка  $S$  с физическим адресом  $ab$ ;  $A, B$  — максимальное число строк и столбцов соответственно. Для восстановления логической структуры сети матрица  $S$  дополняется множеством избыточных столбцов и (или) строк  $S_r$ . Тогда избыточная матрица  $S_i$  есть  $S_i = \{S, S_r\}$ .

При известных отказавших НМК задача самоорганизации состоит в поиске отображения общего алгоритма  $U$  в области ячеек  $S_p$  ( $|S_p| = S$ ), соответствующей множеству работоспособных НМК. Поиск эффективного отображения общего алгоритма  $U$  можно осуществить с учетом ограничений на затраты памяти НМК, аппаратуры и число его настроек (число частных алгоритмов). Требуется выбрать метод самоорганизации  $F_{so}$ , позволяющий осуществить отображение  $U \rightarrow S_p$  такое, что

$$P_{OF} = \max\{P_{of}\},$$

$$\left\{ \begin{array}{l} P_{of} = \sum_t Q_t I_{tf}, \\ \Pi_g + K_f \Pi_h \leq \Pi_{gr}, \\ \sum_g \Pi_g \leq \Pi_u, \\ \sum_g E_g \leq E_r, \end{array} \right.$$

где  $P_{OF}$  — максимальная отказоустойчивость, соответствующая  $F$ -му методу самоорганизации  $F_{so}$  среди множества методов  $f$ ;  $t$  — кратность отказа НМК;  $Q_t$  — вероятность появления отказов  $t$ -й кратности;  $I_{tf}$  — вероятность исправления отказов  $t$ -й кратности  $f$ -м методом;  $\Pi_g$  — емкость памяти для хранения частного алгоритма  $g$ -м НМК;  $K_f$  — число настроек  $f$ -м методом при отказах в сети;  $\Pi_h$  — емкость памяти для хранения частного алгоритма  $h$ -го НМК;  $\Pi_{gr}$  — граничная емкость памяти  $g$ -го НМК;  $\Pi_u$  — требуемая емкость для хранения всех частных алгоритмов общего алгоритма управления;  $E_g$  — дополнительные аппаратурные затраты для  $g$ -го микроконтроллера;  $E_r$  — граничные аппаратурные затраты для сети.

Каждый из методов самоорганизации характеризуется своим значением коэффициента настройки  $K_f$ , т. е. числом хранимых частных алгоритмов соседних НМК в памяти каждого из микроконтроллеров. Множество исследуе-

ных методов самоорганизации ограничивается только теми, для которых  $K_f$  удовлетворяет второму из приведенных ограничений. Определение граничного значения  $K_f$  для одного из методов перестройки логической структуры управляющей системы обосновано в разд. 3.

2. Решение задачи. Решение задачи самоорганизации ( $A_{so}$ ) включает несколько этапов ( $A_{sd}, A_{sr}, A_{ss}, A_{sn}$ ), которые связаны с определением дефектных ситуаций сети  $A_{sd}$ , т. е. самодиагностированием при оперативном методе самоорганизации, преобразованием  $A_{sr}$  матрицы  $S$ , обеспечивающим размещение алгоритма  $U$  в области  $S_p$ , синхронизацией  $A_{ss}$  управляющего и управляемого процессов после исправления ситуации отказа, настройкой  $A_{sn}$  НМК на соответствующий частный алгоритм. Указанные этапы решения задачи  $A_{so}$  сводятся к следующим преобразованиям:  $A_{sd}: D_c \rightarrow D_e$ ,  $A_{sr}: S_{pc} \rightarrow S_{pe}$ ,  $A_{ss}: T_c \rightarrow T_e$ ,  $A_{sn}: LA_{se} \rightarrow FAP_e$ , где  $D_c, D_e$  — двоичные векторы состояний работоспособности МУС с числом компонентов «исправен—отказал», равным числу НМК МУС, до и после ситуации отказа;  $S_{pc}, S_{pe}$  — матрицы работоспособных НМК, отображающие общий алгоритм  $U$ , до и после исправления отказа соответственно;  $T_c, T_e$  — состояние управляющего процесса в сети до и после отказа соответственно;  $LA_{se}$  — логический адрес ячейки  $S$ , соответствующий номеру алгоритма, на который настраивается НМК;  $FAP_e$  — физический адрес поля памяти, где находится указанный алгоритм.

Структура СОС, в которой каждому этапу преобразований поставлен в соответствие блок, включает распределенные системы вычисления логических адресов (РСВЛА), настройки микроконтроллеров (РСНМК) и самосинхронизации (РССС) (рис. 1). Отказоустойчивая управляющая система с самоорганизующим слоем состоит из двух взаимодействующих блоков, т. е. непосредственно мультимикроконтроллерной управляющей системы (ММКУС) с резервными НМК и СОС. По состояниям  $R$  НМК и сигналу перестройки  $Z$  формируются  $LA$  и  $FAP$  для каждого НМК, после чего ММКУС с восстановленной логической структурой включается по сигналу  $P$  на выполнение управления. Блоки СОС взаимодействуют между собой по сигналам  $LA, FAP, T$ , характеризующим настройку сети НМК, алгоритмов и управляющего процесса. Оперативные средства самодиагностирования [11] включены в ММКУС и в дальнейшем не рассматриваются.

Распределение общего алгоритма  $U$  выполняется в избыточной матрице  $S_i$ , содержащей ячейки с резервными и отказавшими НМК. Обычно в качестве резерва в матричные структуры вводятся дополнительные строки и (или) столбцы ячеек. После проведения преобразования каждая ячейка имеет два номера или адреса: исходный физический адрес, а также номер алгоритма (логический адрес), на который настраивается НМК ячейки. Номер алгоритма соответствует физическому адресу ячейки, в которой размещен НМК, выпол-

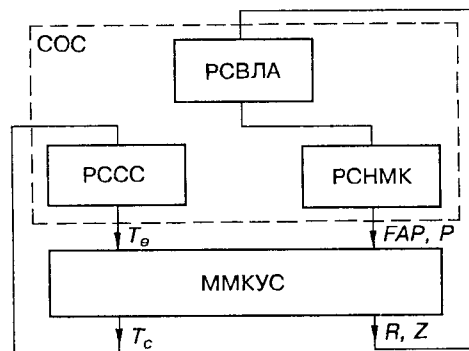


Рис. 1. Структура отказоустойчивой управляющей системы

нявший алгоритм до преобразования. Рассмотрим особенности преобразования логической структуры управляющей системы.

3. **Распределение алгоритмов в логической структуре сети.** Решение задачи преобразования логической структуры управляющей системы заключается в поиске не только метода логической перестройки сети микроконтроллеров, позволяющего в дальнейшем вычислить новые логические адреса НМК, но и пути распределения частных алгоритмов между микроконтроллерами, обеспечивающего возможность их настройки на соответствующий алгоритм по новому логическому адресу.

Распределение  $\pi$  частных алгоритмов между известным числом работоспособных микроконтроллеров осуществляется таким образом, чтобы при любой допустимой комбинации отказавших НМК вычисленные логические адреса  $LA_{ij}$  каждого из работоспособных микроконтроллеров соответствовали одному из номеров  $ij, e_1, \dots, e_{kf}$  ограниченного множества  $\pi_{ij}$  хранимых в нем алгоритмов  $U_{ij}, U_{e_1}, \dots, U_{e_{kf}}$ :

$$\pi = \{\pi_{11}, \dots, \pi_{ij}, \dots, \pi_{A+1, B+1}\}, \quad \pi_{ij} = \{U_{ij}, U_{e_1}, \dots, U_{e_{kf}}\},$$

$$LA_{ij} \in \{ij, e_1, \dots, e_{kf}\}.$$

Стратегия распределения частных алгоритмов определяется особенностями преобразования логической структуры управляющей системы. Метод глобальной перестройки, позволяющий изменять логический адрес ячейки матрицы многократно и тем самым существенно увеличивать число возможных конфигураций работоспособных ячеек в сравнении с локальной перестройкой [4], дает возможность восстанавливать логическую структуру управляющей системы при многократных отказах НМК. В связи с этим для поиска распределения алгоритмов и вычисления логических адресов используется метод глобальной перестройки.

Существо глобальных преобразований поясним на примере так называемой прямой или непосредственной перестройки матрицы  $S_i$  с избыточными строкой и столбцом. Выполнение преобразований осуществляется в четыре этапа:

1. Определение ячеек матрицы, изменяющих адреса строк в логическом адресе.

2. Определение ячеек матрицы, изменяющих номера столбцов в логическом адресе.

3. Сдвиг на одну ячейку физических номеров ячеек, выделенных на втором этапе, в горизонтальном направлении с заменой номера ячейки с отказавшим НМК номером ячейки резервного столбца.

4. Сдвиг на одну ячейку номеров ячеек, выделенных на первом этапе, в вертикальном направлении с заменой номера ячейки с отказавшим НМК номером ячейки резервной строки.

Так как каждый логический адрес ячейки соответствует алгоритму НМК, на который он должен настраиваться, то тем самым общий алгоритм управления  $U$  отображается в области ячеек, соответствующих работоспособным НМК ( $U \rightarrow S_p$ ), т. е. выполняется переход от одной области работоспособных НМК  $S_{pc}$  к другой  $S_{pe}$ .

Определение множества частных алгоритмов для каждого из микроконтроллеров при преобразовании логической структуры в соответствии с указанными этапами основывается на следующем утверждении.

**Утверждение.** Работоспособный микроконтроллер ячейки  $S_{ij}$  при реализации метода глобальной прямой перестройки сети настраивается только на один из четырех алгоритмов управления: непосредственно на алгоритм  $U_{ij}$  и алгоритмы  $U_{i-1, j}, U_{i, j-1}, U_{i-1, j-1}$ .

**Д о к а з а т е л ь с т в о.** При глобальной перестройке каждый номер матрицы может участвовать в сдвигах или только на третьем этапе преобразования, или на четвертом, или последовательно на обоих указанных этапах. Предположим произвольный номер ячейки участвует только в горизонталь-

ном сдвиге. Тогда соседняя ячейка по горизонтали с большим номером, например  $i, j$ , получает номер  $i, j - 1$ .

Если номер ячейки участвует в вертикальном сдвиге, то соседняя ячейка по вертикали, например с текущим номером  $i, j$ , принимает номер  $i - 1, j$ .

При сдвиге номера ячейки последовательно на третьем и четвертом этапах соседняя по диагонали ячейка, например с номером  $i, j$ , захватывает номер  $i - 1, j - 1$ .

Отсюда следует, что произвольная ячейка с номером  $i, j$  может принимать номера  $i - 1, j; i, j - 1; i - 1, j - 1$ . Тогда НМК, размещаемый в ячейке  $S_{ij}$ , настраивается, наряду с алгоритмом  $U_{ij}$ , только на алгоритмы  $U_{i-1,j}; U_{i,j-1}$  и  $U_{i-1,j-1}$ , т. е.  $K_f = 3$ .

**4. Вычисление и поиск логических адресов в сети микроконтроллеров.** В связи с тем, что настройка может быть осуществлена на один из трех соседних алгоритмов, алгоритм вычисления логического адреса НМК обеспечивает выполнение операций горизонтального ( $0 \rightarrow$ ), вертикального ( $0 \downarrow$ ) и последовательно вертикального и горизонтального ( $0 \downarrow \rightarrow$ ) сдвигов. Названные операции сдвига осуществляются следующим образом для  $ij$ -го НМК:  $0 \rightarrow_{ij} : LA_j := FA_j - 1, LA_i := FA_i; 0 \downarrow_{ij} : LA_i := FA_i - 1, LA_j := FA_j; 0 \downarrow \rightarrow_{ij} : LA_i := FA_i - 1, LA_j := FA_j - 1; LA_{ij} = LA_i \cdot LA_j; FA_{ij} = FA_i \cdot FA_j$ , где  $LA_i, LA_j$  — логические адреса строки и столбца ячейки  $S_{ij}$  с физическим адресом  $FA_i$  по строке и  $FA_j$  по столбцу; « $\cdot$ » — операция соединения. Условия выполнения операций сдвига формируются в соответствии с полученными выражениями:

$$\begin{aligned} 0 \downarrow_{ij} &= X \downarrow_{i-1,j} \bar{R}_{ij}, & 0 \downarrow \rightarrow_{ij} &= \bar{X} \downarrow_{i-1,j} X \downarrow_{i-1,j} \bar{R}_{ij}, \\ 0 \rightarrow_{ij} &= \bar{R}_{ij} \bar{X} \downarrow_{i-1,j} \bar{X} \downarrow_{i-1,j} (\bar{X} \rightarrow_{i,j-1} X \downarrow_{i,j-1} \vee X \rightarrow_{i,j-1}), \end{aligned} \quad (1)$$

где  $X \downarrow_{i-1,j}$  — условие вертикального сдвига номера ячейки  $i - 1, j$ ;

$$R_{ij} = \begin{cases} 1, & \text{если отказ НМК } ij; \\ 0, & \text{если НМК } ij \text{ работоспособен;} \end{cases}$$

$X \downarrow \rightarrow_{i-1,j}, X \downarrow \rightarrow_{i,j-1}$  — условия сдвига горизонтального, а затем вертикального номеров ячеек  $i - 1, j$  и  $i, j - 1$  соответственно;  $X \rightarrow_{i,j-1}$  — условие горизонтального сдвига номера ячейки  $i, j - 1$ .

Вычисление условий направления сдвига алгоритмов микроконтроллеров осуществляется в соответствии с логическими выражениями, приведенными в [5]. Операции сдвига выполняются параллельно для всех НМК. При этом физические адреса НМК фиксируются заранее, резервный столбец получает номер  $B + 1$ , где  $B$  — число столбцов избыточной матрицы  $S_p$ , а резервная строка —  $A + 1$ , где  $A$  — число строк избыточной матрицы  $S_p$ . Ячейки отказавших НМК получают нулевой адрес.

Поиск адресата в перенумерованной МУС осуществлен путем модификации формата сообщения и путевой функции [7], используемых в средствах коммутации мультипроцессорных систем [8]. Формат передаваемого сообщения, помимо адреса и данных, включает метку, которая характеризует этап поиска. На первом этапе поиска адрес сообщения сравнивается с физическим и логическим адресами в каждом НМК маршрута и определяется одно из четырех возможных направлений («вверх», «вниз», «влево», «вправо») трансляции. Выбор направления организуется таблично по результатам сравнения физического адреса и адреса в сообщении. В случае совпадения адреса в сообщении с логическим адресом поиск заканчивается и данные передаются получателю. При нахождении НМК с физическим адресом, указанным в сообщении, и нулевым логическим адресом изменяется состояние метки и начинается второй этап поиска. Этот этап заключается в определении адресата среди трех соседних НМК. Маршрут движения сообщения относительно источника инициации второго этапа поиска проходит последовательно через правый («восточный») НМК, далее через находящийся по диагонали внизу («юго-вос-

точный») и заканчивается в расположенном снизу («южном») НМК. Окончание поиска и передача данных возможны в любом из трех НМК при совпадении адреса в сообщении и логического адреса НМК.

5. Настройка микроконтроллера сети. Переключение алгоритмов сети НМК, т. е. настройка ( $A_{sn}$ ) каждого из них на соответствующий алгоритм, осуществляется путем преобразования, реализуемого с помощью таблицы соответствия логических ( $LA_{ij}$ ) и физических ( $FAP_{ij}$ ) адресов областей памяти микроконтроллеров:  $A_{sn}: LA_{ij} \rightarrow FAP_{ij}$ . Таблица соответствия программируется в период записи программ алгоритмов соседних микроконтроллеров в память НМК.

Синхронизация управляющего и управляемого процессов после восстановления логической структуры сети выполняется путем перезапуска алгоритмов НМК с их точек возврата и переключения на текущие состояния параллельных взаимодействующих алгоритмов в соответствии с перенастройкой НМК микроконтроллерной сети. Для этого точки возврата в момент их изменения передаются в соседние НМК. Каждый  $ij$ -й микроконтроллер содержит данные точек возврата  $i, j - 1$ -го,  $i - 1, j - 1$  и  $i - 1, j - 1$ -го соседних НМК:  $TV_{i-1,j}$ ;  $TV_{i,j-1}$ ;  $TV_{i-1,j-1}$ . Поиск места в выбранной области памяти с адресом  $FAP_{ij}$ , соответствующей одному из алгоритмов  $U_{i-1,j}$ ;  $U_{i,j-1}$ ;  $U_{i-1,j-1}$ , осуществляется по адресу смещения, характеризующемуся соответствующей точкой возврата. В соответствии с изложенным преобразование  $A_{ss}$  в зависимости от нового логического адреса можно представить таким образом:

$$A_{ss}: LA_{ij} = FA_{i-1,j}; FAP_{ij} = FAU_{i-1,j}; TU_{i-1,j} = FAU_{i-1,j} \cdot TV_{i-1,j},$$

$$A_{ss}: LA_{ij} = FA_{i,j-1}; FAP_{ij} = FAU_{i,j-1}; TU_{i,j-1} = FAU_{i,j-1} \cdot TV_{i,j-1},$$

$$A_{ss}: LA_{ij} = FA_{i-1,j-1}; FAP_{ij} = FAU_{i-1,j-1}; TU_{i-1,j-1} = FAU_{i-1,j-1} \cdot TV_{i-1,j-1},$$

где  $FAU_{i-1,j}$ ,  $FAU_{i,j-1}$ ,  $FAU_{i-1,j-1}$  — адреса алгоритмов  $U_{i-1,j}$ ,  $U_{i,j-1}$ ,  $U_{i-1,j-1}$  в памяти  $ij$ -го НМК;  $TU_{i-1,j}$ ,  $TU_{i,j-1}$ ,  $TU_{i-1,j-1}$  — адреса перезапуска алгоритмов  $U_{i-1,j}$ ,  $U_{i,j-1}$ ,  $U_{i-1,j-1}$  в памяти  $ij$ -го НМК.

6. Реализация алгоритма настройки. На основании доказанного утверждения, а также известных условий сдвига номеров соседних ячеек разработаны две реализации алгоритма получения логических адресов НМК: алгоритм конвейерного формирования логических адресов, рассмотренный авторами в [9, 10], и алгоритм параллельного вычисления логических адресов. Для реали-

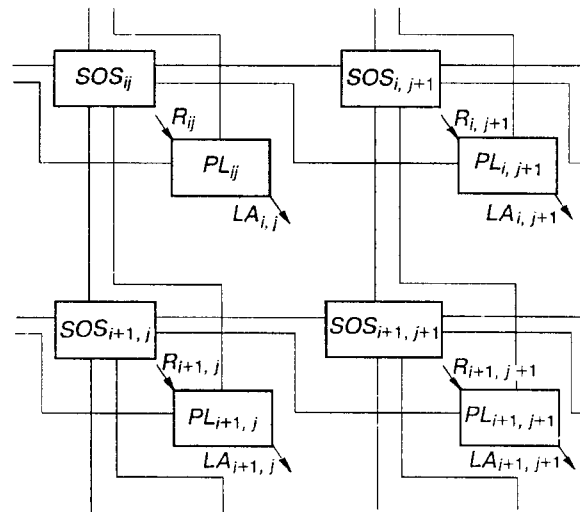


Рис. 2. Структура распределенных средств вычисления логических адресов

зации алгоритма параллельных вычислений необходимо заранее зафиксировать физический адрес каждого НМК.

Аппаратурные средства параллельного вычисления логических адресов, фрагмент которых представлен на рис. 2, включают конвейерный матричный процессор вычисления условий сдвига, состоящий из процессоров  $SOS_{ij}, \dots, SOS_{i+1, j+1}$ , и параллельный мультипроцессор формирования логических адресов сети, включающий процессоры  $PL_{ij}, \dots, PL_{i+1, j+1}$  ( $i = 1, \dots, A + 1$ ;  $j = 1, \dots, B + 1$ ). Для вычисления логических адресов в соответствии с выражением (1) каждый процессор  $PL_{ij}$  связан с соседними процессорами  $SOS_{i-1, j}$  и  $SOS_{i, j-1}$ . Введение процессоров  $SOS_{ij}$  и  $PL_{ij}$  и связей их с соседями для каждого  $ij$ -го НМК позволило в сравнении с чисто программной реализацией вычисления логических адресов в сети исключить трансляцию адресов отказавших микроконтроллеров по сети, хранение информации о них и последующее вычисление логического адреса в каждом из микроконтроллеров. В связи с этим удалось снизить время перестройки, которое для формирования логических адресов в сети определяется временем вычисления адреса в одном процессоре.

**Заключение.** Рассмотренный метод самоорганизации управляющей системы, представленной сетью микроконтроллеров, дает возможность использовать каждый из них в качестве резервного для одного из трех соседних микроконтроллеров. Это обеспечивает высокую гибкость в перестройке системы при многократных отказах микроконтроллеров и ее отказоустойчивость.

В качестве микроконтроллеров для реализации метода самоорганизации предложены НМК, позволившие заменить перераспределение алгоритмов микроконтроллеров их перенастройкой и снизить время восстановления. Для перенастройки системы для каждого НМК определены множества хранимых в их памяти частных алгоритмов.

Слежение каждого НМК за соседними микроконтроллерами и регулярное получение от них в точках возврата необходимых для перезапуска данных обеспечивают синхронизацию после отказа управляющего процесса в мультимикроконтроллерной сети и управляемого процесса, протекающего в объекте.

Рассмотренные алгоритмы определения условий направления сдвига размещенных в НМК алгоритмов и параллельных вычислений логических адресов для каждого НМК позволяют построить оперативные однородные распределенные средства самоорганизации сети при отказах, не требующие перестройки связей в управляющей системе. При этом отпадает необходимость в использовании сложных коммутационных элементов, встраиваемых в сеть связи управляющей системы.

Оперативные однородные распределенные средства самоорганизации позволяют строить системы управления технологическими процессами и техническими машинами с автоматической адаптацией при отказах и обеспечивать непрерывность их функционирования.

#### СПИСОК ЛИТЕРАТУРЫ

1. Мамедли Э. М., Соболев Н. А. Механизмы операционных систем, обеспечивающие отказоустойчивость в управляющих многомашинных вычислительных системах // Автоматика и телемеханика. 1995. № 8. С. 3.
2. Автоматное управление асинхронными процессами в ЭВМ и дискретных системах / Под ред. В. И. Варшавского. М.: Наука, 1986.
3. Воробьев В. А., Лаходынова Н. В. Процессорная матрица с перестраиваемой структурой и перестраиваемым резервом // Автометрия. 1994. № 5. С. 90.
4. Кун С. Матричные процессоры на СБИС. М.: Мир, 1991.
5. Сами М., Стефанелли Р. Перестраиваемые архитектуры матричных процессорных СБИС // ТИИЭР. 1986. № 5. С. 107.
6. Каравай М. Ф. Применение теории симметрии к анализу и синтезу отказоустойчивых систем // Автоматика и телемеханика. 1996. № 6. С. 159.

7. Корнеев В. В. Архитектура вычислительных систем с программируемой структурой. Новосибирск: Наука, 1985.
8. А. с. 1793436 СССР. Модуль матричного коммутатора /В. А. Мельников, А. В. Галицкий, В. В. Копылов и др. Оpubл. 07.02.93, Бюл. № 5.
9. Колосков В. А., Титов В. С. Распределенная самоорганизация однородной управляющей системы // Изв. вузов. Приборостроение. 1994. № 11, 12. С. 20.
10. Колосков В. А., Титов В. С. Матричный процессор самоорганизации управляющей сети // Алгоритмы и структуры систем обработки информации. Тула: ТГТУ, 1994. С. 104.
11. Сокомонян Е. С., Слабаков Е. В. Самопроверяемые устройства и отказоустойчивые системы. М.: Радио и связь, 1989.

*Поступила в редакцию 26 августа 1996 г.*

---